
Model specification

This chapter is devoted to the description of specific classes of DLMS that, alone or in combinations, are most often used to model univariate or multivariate time series. The additive structure of DLMS makes it easy to think of the observed series as originating from the sum of different components, a long term trend and a seasonal component, for example, possibly subject to an observational error. The basic models introduced in this chapter are in this view elementary building blocks in the hands of the modeler, who has to combine them in an appropriate way to analyze any specific data set. The focus of the chapter is the description of the basic models together with their properties; estimation of unknown parameters will be treated in the following chapter. For completeness we include in Section 3.1 a brief review of some traditional methods used for time series analysis. As we will see, those methods can be cast in a natural way in the DLM framework.

3.1 Classical tools for time series analysis

3.1.1 Empirical methods

There are several forecasting methods that, although originally proposed as empirical tools with no probabilistic interpretation, are quite popular and effective. Exponentially weighted moving average (EWMA) is a traditional method used to forecast a time series. It used to be very popular for forecasting sales and inventory level. Suppose one has observations y_1, \dots, y_t and one is interested in predicting y_{t+1} . If the series is non-seasonal and shows no systematic trend, a reasonable predictor can be obtained as a linear combination of the past observations in the following form:

$$\hat{y}_{t+1|t} = \lambda \sum_{j=0}^{t-1} (1 - \lambda)^j y_{t-j} \quad (0 \leq \lambda < 1). \quad (3.1)$$

For t large, the weights $(1 - \lambda)^j \lambda$ sum approximately to one. From an operational point of view, (3.1) implies the following updating of the forecast at time $t - 1$ when the new data point y_t becomes available:

$$\hat{y}_{t+1|t} = \lambda y_t + (1 - \lambda) \hat{y}_{t|t-1},$$

starting from $\hat{y}_{2|1} = y_1$. This is also known as exponential smoothing or Holt point predictor. It can be rewritten as

$$\hat{y}_{t+1|t} = \hat{y}_{t|t-1} + \lambda(y_t - \hat{y}_{t|t-1}), \quad (3.2)$$

enlightening its “forecast-error correction” structure: the point forecast for y_{t+1} is equal to the previous forecast $\hat{y}_{t|t-1}$, corrected by the *forecast error* $e_t = (y_t - \hat{y}_{t|t-1})$ once we observe y_t . Notice the similarity between (3.2) and the state estimate updating recursion given by the Kalman filter for the local level model (see page 38). At time t , forecasts of future observations are taken to be equal to the forecast of y_{t+1} ; in other words, $\hat{y}_{t+k|t} = \hat{y}_{t+1|t}$, $k = 1, 2, \dots$, and the forecast function is constant.

Extensions of EWMA exist that allow for a linear forecast function. For example, the popular Holt linear predictor extends the simple exponential smoothing to nonseasonal time series that show a local linear trend, decomposing y_t as the sum of a local level and a local growth rate—or slope: $y_t = L_t + B_t$. Point forecasts are obtained by combining exponential smoothing forecasts of the level and the growth rate:

$$\hat{y}_{t+k|t} = \hat{L}_{t+1|t} + \hat{B}_{t+1|t} k,$$

where

$$\begin{aligned} \hat{L}_{t+1|t} &= \lambda y_t + (1 - \lambda) \hat{y}_{t|t-1} = \lambda y_t + (1 - \lambda) (\hat{L}_{t|t-1} + \hat{B}_{t|t-1}) \\ \hat{B}_{t+1|t} &= \gamma (\hat{L}_{t+1|t} - \hat{L}_{t|t-1}) + (1 - \gamma) \hat{B}_{t|t-1}. \end{aligned}$$

The above recursive formulae can be rewritten as

$$\hat{L}_{t+1|t} = \hat{y}_{t|t-1} + \lambda e_t, \quad (3.3a)$$

$$\hat{B}_{t+1|t} = \hat{B}_{t|t-1} + \lambda \gamma e_t, \quad (3.3b)$$

where $e_t = y_t - \hat{y}_{t|t-1}$ is the forecast error. Further extensions to include a seasonal component are possible, such as the popular Holt and Winters forecasting methods; see, e.g., Hyndman et al. (2008).

Although of some practical utility, the empirical methods described in this subsection are not based on a probabilistic or statistical model for the observed series, which makes it impossible to assess uncertainty (about the forecasts, for example) using standard measures like confidence or probability intervals. As they stand, these methods can be used as exploratory tools. In fact, they can also be derived from an underlying DLM, which can in this case be used to provide a theoretical justification for the method and to derive probability intervals. For an in-depth treatment along these lines, the reader can consult Hyndman et al. (2008). Package `forecast` (Hyndman; 2008) contains functions that implement the methods described in the book.

3.1.2 ARIMA models

Among the most widely used models for time series analysis is the class of autoregressive moving average (ARMA) models, popularized by Box and Jenkins (see Box et al.; 2008). For nonnegative integers p and q , a univariate stationary ARMA(p,q) model is defined by the relation

$$Y_t = \mu + \sum_{j=1}^p \phi_j (Y_{t-j} - \mu) + \sum_{j=1}^q \psi_j \epsilon_{t-j} + \epsilon_t, \quad (3.4)$$

where (ϵ_t) is Gaussian white noise with variance σ_ϵ^2 and the parameters ϕ_1, \dots, ϕ_p satisfy a stationarity condition. To simplify the notation, we assume in what follows that $\mu = 0$. When the data appear to be nonstationary, one usually takes differences until stationarity is achieved, and then proceeds fitting an ARMA model to the differenced data. A model for a process whose d th difference follows an ARMA(p,q) model is called an autoregressive integrated moving average process of order (p, d, q) , or ARIMA(p,d,q). The orders p, q can be chosen informally by looking at empirical autocorrelations and partial autocorrelations, or using a more formal model selection criterion like AIC or BIC. Univariate ARIMA models can be fit in R using the function `arima` (see Venables and Ripley (2002) for details on ARMA analysis in R).

ARMA models for m -dimensional vector observations are formally defined by the same formula (3.4), taking (ϵ_t) to be m -dimensional Gaussian white noise with variance Σ_ϵ and the parameters ϕ_1, \dots, ϕ_p and ψ_1, \dots, ψ_q to be $m \times m$ matrices satisfying appropriate stationarity restrictions. Although in principle as simple to define as in the univariate case, multivariate ARMA models are much harder to deal with than their univariate counterpart, in particular for what concerns identifiability issues and fitting procedures. The interested reader can find a thorough treatment of multivariate ARMA models in Reinsel (1997). Functions for the analysis of multivariate ARMA models in R can be found in the contributed package `dse1` (Gilbert; 2008).

It is possible to represent an ARIMA model, univariate or multivariate, as a DLM, as we will show in Sections 3.2.5 and 3.3.7. This may be useful for the evaluation of the likelihood function. However, in spite of the fact that formally an ARIMA model can be considered a DLM, the philosophy underlying the two classes of models is quite different: on the one hand, ARIMA models provide a black-box approach to data analysis, offering the possibility of forecasting future observations, but with a limited interpretability of the fitted model; on the other hand, the DLM framework encourages the analyst to think in terms of easily interpretable, albeit unobservable, processes—such as trend and seasonal components—that drive the observed time series. Forecasting the individual underlying components of the process, in addition to the observations, is also possible—and useful in many applications—within the DLM framework.

3.2 Univariate DLMs for time series analysis

As we have discussed in Chapter 2, the Kalman filter provides the formulae for estimation and prediction for a completely specified DLM, that is, a DLM where the matrices F_t , G_t and the covariance matrices V_t and W_t are known. In practice, however, specifying a model can be a difficult task. A general approach that works well in practice is to imagine a time series as obtained by combining simple elementary components, each one capturing a different feature of the series, such as trend, seasonality, and dependence on covariates (regression). Each component is represented by an individual DLM, and the different components are then combined in a unique DLM, producing a model for the given time series. To be precise, the components are combined in an additive fashion; series for which a multiplicative decomposition is more appropriate can be modeled using an additive decomposition after a log transformation. We detail below the additive decomposition technique in the univariate case, although the same approach carries over to multivariate time series with obvious modifications.

Consider a univariate series (Y_t) . One may assume that the series can be written as the sum of *independent* components

$$Y_t = Y_{1,t} + \cdots + Y_{h,t}, \quad (3.5)$$

where $Y_{i,t}$ might represent a trend component, $Y_{2,t}$ a seasonal component, and so on. The i th component $Y_{i,t}$, $i = 1, \dots, h$, might be described by a DLM as follows:

$$\begin{aligned} Y_{i,t} &= F_{i,t}\theta_{i,t} + v_{i,t}, & v_{i,t} &\sim \mathcal{N}(0, V_{i,t}), \\ \theta_{i,t} &= G_{i,t}\theta_{i,t-1} + w_{i,t}, & w_{i,t} &\sim \mathcal{N}(0, W_{i,t}), \end{aligned}$$

where the p_i -dimensional state vectors $\theta_{i,t}$ are distinct and the series $(Y_{i,t}, \theta_{i,t})$ and $(Y_{j,t}, \theta_{j,t})$ are mutually independent for all $i \neq j$. The component DLMs are then combined in order to obtain the DLM for (Y_t) . By the assumption of independence of the components, it is easy to show that $Y_t = \sum_{i=1}^h Y_{i,t}$ is described by the DLM

$$\begin{aligned} Y_t &= F_t\theta_t + v_t, & v_t &\sim \mathcal{N}(0, V_t), \\ \theta_t &= G_t\theta_{t-1} + w_t, & w_t &\sim \mathcal{N}(0, W_t), \end{aligned}$$

where

$$\theta_t = \begin{bmatrix} \theta_{1,t} \\ \vdots \\ \theta_{h,t} \end{bmatrix}, \quad F_t = [F_{1,t} | \cdots | F_{h,t}],$$

G_t and W_t are the block diagonal matrices

$$G_t = \begin{bmatrix} G_{1,t} & & \\ & \ddots & \\ & & G_{h,t} \end{bmatrix}, \quad W_t = \begin{bmatrix} W_{1,t} & & \\ & \ddots & \\ & & W_{h,t} \end{bmatrix},$$

and $V_t = \sum_{i=1}^j V_{i,t}$. In this chapter, with the exception of Section 3.2.6, we will assume that all the matrices defining a DLM are known; the analysis will be extended to DLMs with unknown parameters in Chapters 4 and 5.

In R, `dlm` objects are created by the functions of the family `dlmMod*`, or by the general function `dlm`. DLMs having a common dimension of the observation vectors can be added together to produce another DLM. For example, `dlmModPoly(2) + dlmModSeas(4)` adds together a linear trend and a quarterly seasonal component. We start by introducing the families of DLMs that are commonly used as basic building blocks in the representation (3.5). In particular, Sections 3.2.1 and 3.2.2 cover trend and seasonal models, respectively. These two component models can be used to carry over to the DLM setting the classical decomposition “trend + seasonal component + noise” of a time series.

3.2.1 Trend models

Polynomial DLMs are the models most commonly used for describing the trend of a time series, where the trend is viewed as a smooth development of the series over time. At time t , the expected trend of the time series can be thought of as the expected behavior of Y_{t+k} for $k \geq 1$, given the information up to time t ; in other words, the expected trend is the forecast function $f_t(k) = E(Y_{t+k}|y_{1:t})$. A polynomial model of order n is a DLM with constant matrices $F_t = F$ and $G_t = G$, and a forecast function of the form

$$f_t(k) = E(Y_{t+k}|y_{1:t}) = a_{t,0} + a_{t,1}k + \cdots + a_{t,n-1}k^{n-1}, \quad k \geq 0, \quad (3.6)$$

where $a_{t,0}, \dots, a_{t,n-1}$ are linear functions of $m_t = E(\theta_t|y_{1:t})$ and are independent of k . Thus, the forecast function is a polynomial of order $n-1$ in k (note that, as we will see, n is the dimension of the state vector and not the degree of the polynomial). Roughly speaking, any reasonable shape of the forecast function can be described or closely approximated by a polynomial, by choosing n sufficiently large. However, one usually thinks of the trend as a fairly smooth function of time, so that in practice small values of n are used. The most popular polynomial models are the random walk plus noise model, which is a polynomial model of order $n=1$, and the linear growth model, that is a polynomial model of order $n=2$.

The local level model

The random walk plus noise, or local level model, is defined by the two equations (2.5) of the previous chapter. As noted there, the behavior of the process

(Y_t) is greatly influenced by the signal-to-noise ratio $r = W/V$, the ratio between the two error variances. Figure 3.1 shows some simulated trajectories of (Y_t) and (μ_t) for different values of the ratio r (see Problem 3.1).

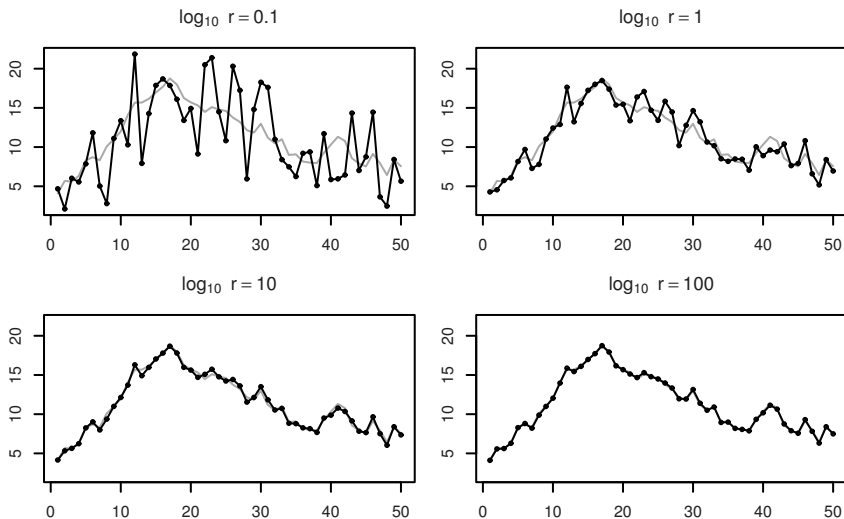


Fig. 3.1. Trajectories of the random walk plus noise, for different values of the signal-to-noise ratio. The trajectory of the state μ_t is shown in gray and is the same in the four plots

The k -steps-ahead predictive distribution for this simple model is

$$Y_{t+k}|y_{1:t} \sim N(m_t, Q_t(k)) \quad , \quad k \geq 1, \tag{3.7}$$

where $Q_t(k) = C_t + \sum_{j=1}^k W_{t+j} + V_{t+k} = C_t + kW + V$. We see that the forecast function $f_t(k) = E(Y_{t,k}|y_{1:t}) = m_t$ is constant (as a function of k). For this reason this model is also referred to as the *steady model*. The uncertainty on the future observations is summarized by the variance $Q_t(k) = C_t + kW + V$, and we clearly see that it increases as the time horizon $t+k$ gets further away.

The controllability and observability matrices of the model are

$$\begin{aligned} \mathcal{C} &= [W^{1/2}], \\ \mathcal{O} &= F = [1], \end{aligned}$$

which are trivially of full rank, as long as $W > 0$. It follows from the results of Section 2.11 that the Kalman filter for this model is asymptotically stable, with R_t, C_t , and the gain matrix K_t converging to limiting values R, C , and K , respectively. It can be shown (see West and Harrison; 1997, Theorem 2.3) that

$$K = \frac{r}{2} \left(\sqrt{1 + \frac{4}{r}} - 1 \right). \quad (3.8)$$

It follows that $C = KV$. This gives an upper bound to the precision attainable in estimating the current value of μ_t . Furthermore, we obtain a limit form of the one-step-ahead forecasts. From (3.7),

$$f_{t+1} = E(Y_{t+1}|y_{1:t}) = m_t = m_{t-1} + K_t(Y_t - m_{t-1}) = m_{t-1} + K_t e_t.$$

For large t , $K_t \approx K$ so that, asymptotically, the one-step-ahead forecast is given by

$$f_{t+1} = m_{t-1} + K e_t. \quad (3.9)$$

A forecast function of the kind (3.9) is used in many popular models for time series. It corresponds to Holt point predictor, see equation (3.2).

It can be shown that Holt point predictor is optimal if (Y_t) is an ARIMA(0,1,1) process. In fact, the steady model has connections with the popular ARIMA(0,1,1) model. It can be shown (Problem 3.3) that, if Y_t is a random walk plus noise, then the first differences $Z_t = Y_t - Y_{t-1}$ are stationary, and have the same autocorrelation function as an MA(1) model. Furthermore, being $e_t = Y_t - m_{t-1}$ and $m_t = m_{t-1} + K_t e_t$, we have

$$\begin{aligned} Y_t - Y_{t-1} &= e_t + m_{t-1} - e_{t-1} - m_{t-2} \\ &= e_t + m_{t-1} - e_{t-1} - m_{t-1} + K_{t-1} e_{t-1} \\ &= e_t - (1 + K_{t-1}) e_{t-1}. \end{aligned}$$

If t is large, so that $K_{t-1} \approx K$,

$$Y_t - Y_{t-1} \approx e_t - (1 - K) e_{t-1}.$$

Since the forecast errors are a white noise sequence (see Chapter 2, page 73), (Y_t) is asymptotically an ARIMA(0,1,1) process.

Example — Annual precipitation at Lake Superior

Figure 3.2 shows annual precipitation in inches at Lake Superior, from 1900 to 1986¹. The series shows random fluctuations about a changing level over time, with no remarkable trend behavior; thus, a random walk plus noise model could be tentatively entertained. We suppose here that the evolution variance W and the observational variance V are known, and we assume that W is much smaller (0.121) than V (9.465) (so $r = 0.0128$). In R a local level model can be set up using the function `d1mModPoly` with first argument `order=1`.

Figure 3.3(a) shows the filtering estimates m_t of the underlying level of the series and Figure 3.3(b) shows the square root of the variances C_t . Recall that

¹ Source: Hyndman (n.d.).

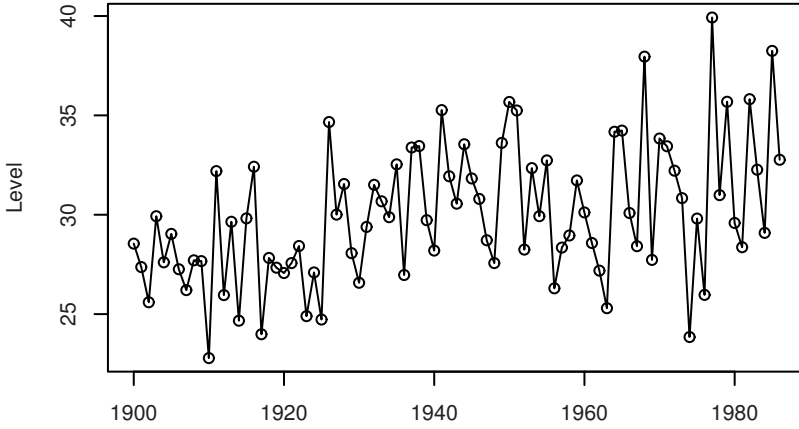


Fig. 3.2. Annual precipitation at Lake Superior

for the local level model C_t has a limiting value as t approaches infinity. The smoothed states s_t and square root of the variances S_t are plotted in Figures 3.3(c) and 3.3(d). The U-shaped behavior of the sequence of variances S_t reflects the intuitive fact that the states around the middle of the time interval spanned by the data are those that can be estimated more accurately.

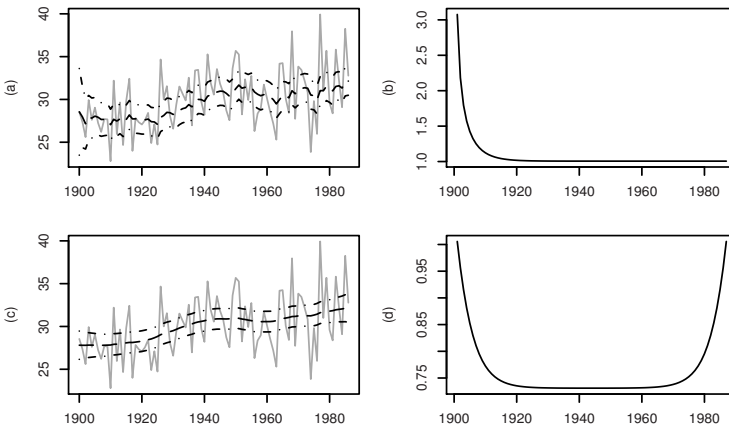


Fig. 3.3. (a): Filtered state estimates m_t with 90% confidence intervals; (b): Square root of filtering variances C_t ; (c): Smoothed state estimates s_t with 90% confidence intervals; (d): Square root of smoothing variances S_t

The one-step-ahead forecasts, for the local level model, are $f_t = m_{t-1}$. The standardized one-step-ahead forecast errors, or standardized innovations, can be computed in R with a call to the `residuals` function, which has a method for `dlmFiltered` objects. The residuals can be inspected graphically (Figure 3.4(a)) to check for unusually large values or unexpected patterns—recall that the standardized innovation process has the distribution of a Gaussian white noise. Two additional very useful graphical tools to detect departures from the model assumptions are the plot of the empirical autocorrelation function (ACF) of the standardized innovations (Figure 3.4(b)) and their normal QQ-plot (Figure 3.4(c)). These can be drawn using the standard R functions `acf` and `qqnorm`. By looking at the plots, there does not seem to be any meaningful departure from the model assumptions.

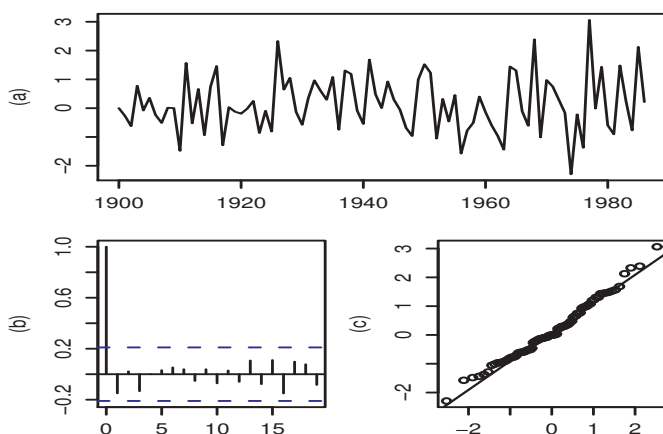


Fig. 3.4. (a): Standardized one-step-ahead forecast errors; (b): ACF of one-step-ahead forecast errors (c): Normal probability plot of standardized one-step-ahead forecast errors

Formal statistical tests may also be employed to assess model assumptions via the implied properties of the innovations. For example, the Shapiro–Wilk test can be used to test the standardized innovations for normality. It is available in R as `shapiro.test`. For the standardized innovations from the Lake Superior precipitation data, the p-value is 0.403, so the null hypothesis of normally distributed standardized innovations cannot be rejected. The Shapiro–Wilk normality test is commonly preferred to the Kolmogorov–Smirnov test, which is also available in R as `ks.test`, as being more powerful against a broad range of alternatives. R functions that perform other normality tests are available in contributed packages `fBasics` (Wuertz; 2008) and `nortest` (Gross; n.d.). For a thorough treatment of normality tests the reader is referred to D’Agostino and Stephens (1986). To test for lack of serial correlation one can

use the Ljung and Box test (Ljung and Box; 1978), which is based on the first k sample autocorrelations, for a prespecified value of k . The test statistic is

$$Q(k) = n(n+2) \sum_{j=1}^k \hat{\rho}^2(j)/(n-j),$$

where n is the sample size and $\hat{\rho}(j)$ is the sample autocorrelation at lag j , defined by

$$\hat{\rho}(j) = \frac{\sum_{t=1}^{n-j} (\tilde{e}_t - \bar{\tilde{e}})(\tilde{e}_{t+j} - \bar{\tilde{e}})}{\sum_{t=1}^n (\tilde{e}_t - \bar{\tilde{e}})^2}, \quad j = 1, 2, \dots$$

What the Ljung–Box test effectively does is test for the absence of serial correlation up to lag k . Using $k = 20$, the p-value of the Ljung–Box test for the standardized innovations of the example is 0.813, confirming that the standardized innovations are uncorrelated. It is also common to compute the p-value of the Ljung–Box test for all the values of k up to a maximum, say 10 or 20. The function `tsdiag`, among other things, does this calculation and plots the resulting p-values versus k for the residuals of a fitted ARMA model. Of course, in this case the calculated p-values should only be taken as an indication since, in addition to the asymptotic approximation of the distribution of the test statistic for any fixed k , the issue of multiple testing would have to be addressed if one wanted to draw a conclusion in a formal way. The display below illustrates how to obtain in R the standardized innovations and perform the Shapiro–Wilk and Ljung–Box tests.

R code

```

> loc <- "http://www.robjhyndman.com/TSDL/roberts/plsuper.dat"
2 > lakeSup <- ts(read.table(loc, skip = 3,
+                       colClasses = "numeric"),
4 +                       start = 1900)
> modLSup <- dlmModPoly(1, dV = 9.465, dW = 0.121)
6 > lSupFilt <- dlmFilter(lakeSup, modLSup)
> res <- residuals(lSupFilt, sd = FALSE)
8 > shapiro.test(res)

10      Shapiro-Wilk normality test

12 data:  res
W = 0.9848, p-value = 0.4033

14 > Box.test(res, lag = 20, type = "Ljung")

16      Box-Ljung test

18
```

```

data: res
20 X-squared = 14.3379, df = 20, p-value = 0.813

22 > sapply(1 : 20, function(i)
+       Box.test(res, lag = i, type = "Ljung-Box")$p.value)
24 [1] 0.1552078 0.3565713 0.2980295 0.4508888 0.5829209 0.6718375
    [7] 0.7590090 0.8148123 0.8682010 0.8838797 0.9215812 0.9367660
26 [13] 0.9143456 0.9185912 0.8924318 0.7983241 0.7855680 0.7971489
    [19] 0.8010898 0.8129607

```

Exponential smoothing one-step-ahead forecasts can be obtained using the function `HoltWinters`. The results for the annual precipitations in Lake Superior are plotted in Figure 3.2.1 (here the smoothing parameter is estimated as $\lambda = 0.09721$). We see that the steady model has, for large t , essentially the same forecast function as the simple exponential smoothing.

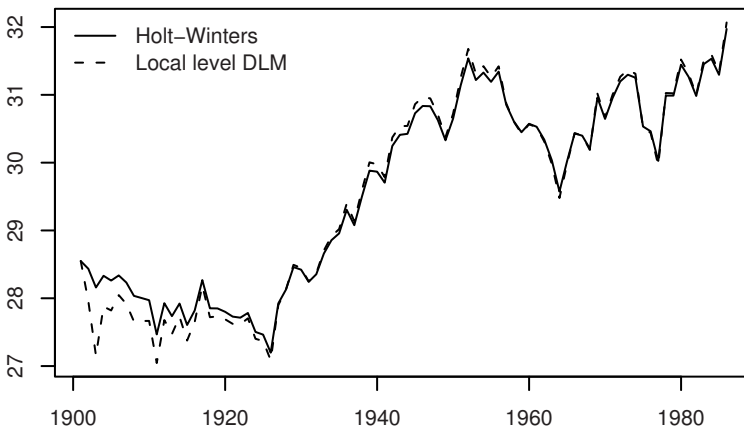


Fig. 3.5. One-step ahead forecasts

R code

```

> HWout <- HoltWinters(lakeSup, gamma = 0, beta = 0)
2 > plot(dropFirst(lSupFilt$f), lty = "dashed",
+       xlab = "", ylab = "")
4 > lines(HWout$fitted[, "level"])
> leg <- c("Holt-Winters", "Local level DLM")
6 > legend("topleft", legend = leg, bty = "n",
+       lty = c("solid", "dashed"))

```

Linear growth model

The linear growth, or local linear trend model, is defined by (2.6). The state vector is $\theta_t = (\mu_t, \beta_t)'$, where μ_t is usually interpreted as the local level and β_t as the local growth rate. The model assumes that the current level μ_t changes linearly through time and that the growth rate may also evolve. It is thus more flexible than a global linear trend model. A good exercise, also for this model, is to simulate trajectories of $(Y_t, t = 1, \dots, T)$, for different values of V and W (see Problem 3.1).

Denoting $m_{t-1} = (\hat{\mu}_{t-1}, \hat{\beta}_{t-1})'$, the one-step-ahead point forecasts and the filtering state estimates are given by

$$a_t = Gm_{t-1} = \begin{bmatrix} \hat{\mu}_{t-1} + \hat{\beta}_{t-1} \\ \hat{\beta}_{t-1} \end{bmatrix} \quad (3.10a)$$

$$f_t = F_t a_t = \hat{\mu}_{t-1} + \hat{\beta}_{t-1}, \quad (3.10b)$$

$$m_t = \begin{bmatrix} \hat{\mu}_t \\ \hat{\beta}_t \end{bmatrix} = a_t + K_t e_t = \begin{bmatrix} \hat{\mu}_{t-1} + \hat{\beta}_{t-1} + k_{t1} e_t \\ \hat{\beta}_{t-1} + k_{t2} e_t \end{bmatrix}. \quad (3.10c)$$

The forecast function is

$$f_t(k) = \hat{\mu}_t + k\hat{\beta}_t,$$

(see Problem 3.6) which is a linear function of k , so the linear growth model is a polynomial DLM of order 2.

The controllability matrix of the linear growth model is

$$\mathcal{C} = \begin{bmatrix} \sigma_\mu & \sigma_\beta & \sigma_\mu & 0 \\ 0 & \sigma_\beta & 0 & \sigma_\beta \end{bmatrix}.$$

The rank of \mathcal{C} is two if and only if $\sigma_\beta > 0$, in which case the model is controllable. The model is always observable, since the observability matrix is always full-rank:

$$\mathcal{O} = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}.$$

Therefore, assuming that $\sigma_\beta > 0$, there are limiting values for R_t , C_t , and K_t . In particular, the Kalman gain K_t converges to a constant matrix $K = [k_1 \ k_2]$ (see West and Harrison; 1997, Theorem 7.2). Therefore, the asymptotic updating formulae for the estimated state vector are given by

$$\begin{aligned} \hat{\mu}_t &= \hat{\mu}_{t-1} + \hat{\beta}_{t-1} + k_1 e_t \\ \hat{\beta}_t &= \hat{\beta}_{t-1} + k_2 e_t. \end{aligned} \quad (3.11)$$

Several popular point predictors' methods use expressions of the form (3.11), such as the Holt linear predictor (compare with (3.3)) and the Box and Jenkins' ARIMA(0,2,2) predictor (see West and Harrison; 1997, p. 221 for a discussion). In fact, the linear growth model is related to the ARIMA(0,2,2)

process. It can be shown (Problem 3.5) that the second differences of (Y_t) are stationary and have the same autocorrelation function as an MA(2) model. Furthermore, we can write the second differences $z_t = Y_t - 2Y_{t-1} + Y_{t-2}$ as

$$z_t = e_t + (-2 + k_{1,t-1} + k_{2,t-1})e_{t-1} + (1 - k_{1,t-2})e_{t-2} \tag{3.12}$$

(see Problem 3.7). For large t , $k_{1,t} \approx k_1$ and $k_{2,t} \approx k_2$, so that the above expression reduces to

$$Y_t - 2Y_{t-1} + Y_{t-2} \approx e_t + \psi_1 e_{t-1} + \psi_2 e_{t-2}$$

where $\psi_1 = -2 + k_1 + k_2$ and $\psi_2 = 1 - k_1$, which is a MA(2) model. Thus, asymptotically, the series (Y_t) is an ARIMA(0,2,2) process.

Example — Spain annual investment

Consider Spain annual investments from 1960 to 2000², plotted in Figure 3.6. The time series shows a roughly linear increase, or decrease, in the level, with a slope changing every few years. In the near future it would not be unreasonable to predict the level of the series by linear extrapolation, i.e., using a linear forecast function. A linear growth model could be therefore

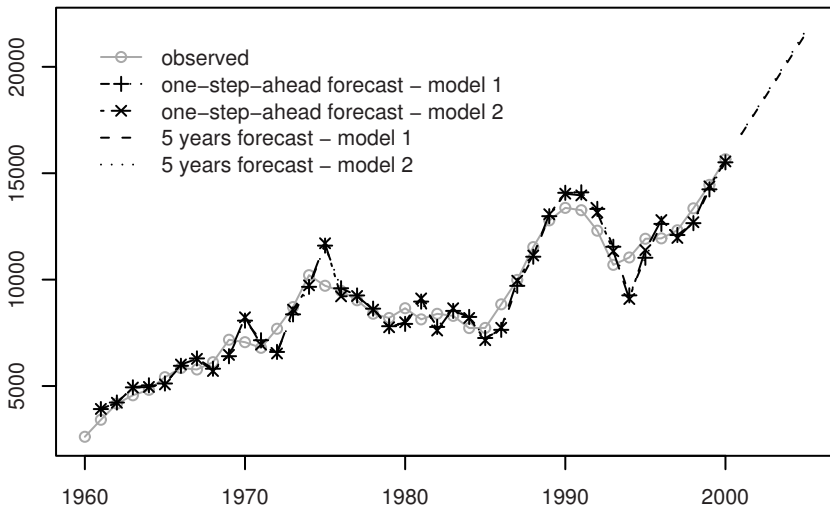


Fig. 3.6. Spain investments

appropriate for these data. We assume that the variances are known (they were actually estimated) and are as follows:

² Source: <http://www.fgn.unisg.ch/eumacro/macrodatta>.

$$W = \text{diag}(102236, 321803), \quad V = 10.$$

The function `dlmModPoly` with argument `order=2` (which is the default) can be used to set up the model in R, see display below. Visual inspection of a QQ-plot and ACF of the standardized innovations (not shown) do not raise any specific concern about the appropriateness of the model. An alternative model (an integrated random walk model, in fact, see page 100) that describes the data almost equally well, with one less parameter, is the linear growth model with the same V and

$$W = \text{diag}(0, 515939).$$

R code

```

> mod1 <- dlmModPoly(dV = 10, dW = c(102236, 321803))
2 > mod1Filt <- dlmFilter(invSpain, mod1)
> fut1 <- dlmForecast(mod1Filt, n = 5)
4 > mod2 <- dlmModPoly(dV = 10, dW = c(0, 515939))
> mod2Filt <- dlmFilter(invSpain, mod2)
6 > fut2 <- dlmForecast(mod2Filt, n = 5)

```

Figure 3.6 shows, together with the data, one-step-ahead forecasts and five years forecasts for the two models under consideration. It is clear that the forecasts, both in sample and out of sample, produced by the two models are very close. The standard deviations of the one-step-ahead forecasts, which can be obtained as `residuals(mod1Filt)$sd`, are also fairly close, 711 for the first model versus 718 for the second at time $t = 41$ (year 2000). The reader can verify that the difference in the forecast variances (`fut1$Q` and `fut2$Q`) grows with the number of steps ahead to be predicted. Several measures of forecasting accuracy can be used to compare the two models more formally. Commonly used criteria are the mean absolute deviation (MAD), the mean square error (MSE), and the mean absolute percentage error (MAPE), defined respectively by the following formulae:

$$\begin{aligned} \text{MAD} &= \frac{1}{n} \sum_{t=1}^n |e_t|, \\ \text{MSE} &= \frac{1}{n} \sum_{t=1}^n e_t^2, \\ \text{MAPE} &= \frac{1}{n} \sum_{t=1}^n \frac{|e_t|}{y_t}. \end{aligned}$$

For the two models under consideration and the Spain investment data, none of the two stands out as a clear winner, as the following display shows.

R code

```

> mean(abs(mod1Filt$f - invSpain))
2 [1] 623.5682

```

```

> mean(abs(mod2Filt$f - invSpain))
4 [1] 610.2621
> mean((mod1Filt$f - invSpain)^2)
6 [1] 655480.6
> mean((mod2Filt$f - invSpain)^2)
8 [1] 665296.7
> mean(abs(mod1Filt$f - invSpain) / invSpain)
10 [1] 0.08894788
> mean(abs(mod2Filt$f - invSpain) / invSpain)
12 [1] 0.08810524

```

An additional statistic that can be used to assess the forecasting performance of a specific model is Theil's U (Theil; 1966). This compares the MSE of the model with the MSE of the trivial “no-change” model that predicts the next observation to be the same as the current one. Formally, the definition is the following:

$$U = \sqrt{\frac{\sum_{t=2}^n (y_t - f_t)^2}{\sum_{t=2}^n (y_t - y_{t-1})^2}}.$$

A value of U less than one means that the entertained model produces better forecasts, on average, than the no-change model. For the Spain investment data the results are reported below.

R code

```

> sqrt(sum((mod1Filt$f - invSpain)[-(1:5)]^2) /
2 + sum(diff(invSpain[-(1:4)])^2))
[1] 0.9245
4 > sqrt(sum((mod2Filt$f - invSpain)[-(1:5)]^2) /
+ sum(diff(invSpain[-(1:4)])^2))
6 [1] 0.9346

```

We left out the first five observations to give the Kalman filter the time to adapt to the data—remember that the prior variance C_0 is very large. In this example, both models perform better than the no-change model, with about a 7% reduction in the square root of the MSE.

n th order polynomial model

The local level and the linear growth models are special cases of the n th order polynomial model. The general n th order polynomial model has an n -dimensional state space and is described by the matrices

$$F = (1, 0, \dots, 0) \quad (3.13a)$$

$$G = \begin{bmatrix} 1 & 1 & 0 & \dots & 0 \\ 0 & 1 & 1 & 0 & \dots & 0 \\ \vdots & & & \ddots & \vdots \\ 0 & \dots & 0 & 1 & 1 \\ 0 & \dots & & 0 & 1 \end{bmatrix} \quad (3.13b)$$

$$W = \text{diag}(W_1, \dots, W_n). \quad (3.13c)$$

In terms of its components, the model can be written in the form

$$\begin{cases} Y_t = \theta_{t,1} + v_t \\ \theta_{t,j} = \theta_{t-1,j} + \theta_{t-1,j+1} + w_{t,j} & j = 1, \dots, n-1 \\ \theta_{t,n} = \theta_{t-1,n} + w_{t,n}. \end{cases} \quad (3.14)$$

So, for $j = 2, \dots, n$, the j th component of the state vector at any time t represents, up to a random error, the increment of the $(j-1)$ st component during the next time interval, while the first component represents the mean response, or the level of the series. The forecast function, $f_t(k)$, is a polynomial of degree $n-1$ in k (Problem 3.6).

The special case that is obtained by setting $W_1 = \dots = W_{n-1} = 0$ is called *integrated random walk model*. The mean response function satisfies for this model the relation $\Delta^n \mu_t = \epsilon_t$ for some white noise sequence (ϵ_t) . The form of the forecast function is again polynomial. With respect to the n th order polynomial model, the integrated random walk model has $n-1$ fewer parameters, which may improve the precision attainable in estimating unknown parameters. On the other hand, having only one degree of freedom in the system noise, it may be slower in adapting to random shocks to the state vector, which may reflect in a lower accuracy in the forecasts.

3.2.2 Seasonal factor models

In this section and the next we present two ways of modeling a time series that shows a cyclical behavior, or “seasonality”: the seasonal factor model, covered below, and the Fourier-form seasonal model, treated in the following section.

Suppose that we have quarterly data $(Y_t, t = 1, 2, \dots)$, for examples on the sales of a store, which show an annual cyclic behavior. Assume for simplicity that the series has zero mean: a non-zero mean, or a trend component, can be modeled separately, so for the moment we consider the series as purely seasonal. We might describe the series by introducing seasonal deviations from the mean, expressed by different coefficients α_i for the different quarters, $i = 1, \dots, 4$. So, if Y_{t-1} refers to the first quarter of the year and Y_t to the second quarter, we assume

$$\begin{aligned} Y_{t-1} &= \alpha_1 + v_{t-1} \\ Y_t &= \alpha_2 + v_t \end{aligned} \tag{3.15}$$

and so on. This model can be written as a DLM as follows. Let $\theta_{t-1} = (\alpha_1, \alpha_4, \alpha_3, \alpha_2)'$ and $F_t = F = (1, 0, 0, 0)$. Then the observation equation of the DLM is given by

$$Y_{t-1} = F\theta_{t-1} + v_{t-1},$$

which corresponds to (3.15). The state equation must “rotate” the components of θ_{t-1} into the vector $\theta_t = (\alpha_2, \alpha_1, \alpha_4, \alpha_3)'$, so that $Y_t = F\theta_t + v_t = \alpha_2 + v_t$. The required permutation of the state vector can be obtained by a permutation matrix G so defined:

$$G = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}.$$

Then the state equation can be written as

$$\theta_t = G\theta_{t-1} + w_t = (\alpha_2, \alpha_1, \alpha_4, \alpha_3)' + w_t.$$

In the static seasonal model, w_t is degenerate on a vector of zeros (i.e., $W_t = 0$). More generally, the seasonal effects might change in time, so that W_t is nonzero and has to be carefully specified.

In general, a seasonal time series with period s can be modeled through an s -dimensional state vector θ_t of seasonal deviations, by specifying a DLM with $F = (1, 0, \dots, 0)$ and G given by a s by s permutation matrix. Identifiability constraints have to be imposed on the seasonal factors $\alpha_1, \dots, \alpha_s$. A common choice is to impose that they sum to zero, $\sum_{j=1}^s \alpha_j = 0$. The linear constraint on the s seasonal factors implies that there are effectively only $s - 1$ free seasonal factors, and this suggests an alternative, more parsimonious representation that uses an $(s - 1)$ -dimensional state vector. In the previous example of quarterly data, one can consider $\theta_{t-1} = (\alpha_1, \alpha_4, \alpha_3)'$ and $\theta_t = (\alpha_2, \alpha_1, \alpha_4)'$, with $F = (1, 0, 0)$. To go from θ_{t-1} to θ_t , assuming for the moment a static model without system evolution errors and using the constraint $\sum_{i=1}^4 \alpha_i = 0$, one has to apply the linear transformation given by the matrix

$$G = \begin{bmatrix} -1 & -1 & -1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}.$$

In general, for a seasonal model with period s , one can consider an $(s - 1)$ -dimensional state space, with $F = (1, 0, \dots, 0)$ and

$$G = \begin{bmatrix} -1 & -1 & \dots & -1 & -1 \\ 1 & 0 & & 0 & 0 \\ 0 & 1 & & 0 & 0 \\ & & \ddots & & \\ 0 & 0 & & 1 & 0 \end{bmatrix}.$$

A dynamic variation in the seasonal components may be introduced via a system evolution error with variance $W = \text{diag}(\sigma_w^2, 0, \dots, 0)$.

A seasonal factor DLM can be specified in R using the function `d1mModSeas`. For example, a seasonal factor model for quarterly data, with $\sigma_w^2 = 4.2$ and observation variance $V = 3.5$ can be specified as follows.

R code

```
mod <- d1mModSeas(frequency = 4, dV = 3.5, dW = c(4.2, 0, 0))
```

3.2.3 Fourier form seasonal models

Any discrete-time periodic function having period s is characterized by the values it takes at time $t = 1, 2, \dots, s$: if g_t is such a function, with $g_t = \alpha_t$, $t = 1, \dots, s$, then after time s the values of g_t simply repeat themselves, and we have $g_{s+1} = \alpha_1$, $g_{s+2} = \alpha_2$ and so forth. Therefore, we can associate the periodic function g_t to the s -dimensional vector $\alpha = (\alpha_1, \dots, \alpha_s)'$. We can think of α as a linear combination of basis vectors:

$$\alpha = \sum_{j=1}^s \alpha_j u_j,$$

where u_j is the s -dimensional vector having j th component equal one and all the remaining components equal zero. The set $\{u_1, \dots, u_s\}$ is customarily referred to as the canonical basis of \mathbb{R}^s ; clearly, any vector in \mathbb{R}^s has a unique representation as a linear combination of basis vectors. For our purposes, however, this representation is not very useful, since it does not allow us to distinguish between smooth, less smooth, and not-so-smooth functions. Fortunately, there is an alternative basis of \mathbb{R}^s that allows this finer distinction to be made. To begin with, suppose that s is even. In practice, this is the most commonly encountered case—think of quarterly or montly data. Define the *Fourier frequencies*

$$\omega_j = \frac{2\pi j}{s}, \quad j = 0, 1, \dots, \frac{s}{2}$$

and consider the following s dimensional vectors:

$$\begin{aligned}
e_0 &= (1, 1, \dots, 1)' \\
c_1 &= (\cos \omega_1, \cos 2\omega_1, \dots, \cos s\omega_1)' \\
s_1 &= (\sin \omega_1, \sin 2\omega_1, \dots, \sin s\omega_1)' \\
&\vdots \\
c_j &= (\cos \omega_j, \cos 2\omega_j, \dots, \cos s\omega_j)' \\
s_j &= (\sin \omega_j, \sin 2\omega_j, \dots, \sin s\omega_j)' \\
&\vdots \\
c_{s/2} &= (\cos \omega_{s/2}, \cos 2\omega_{s/2}, \dots, \cos s\omega_{s/2})'.
\end{aligned} \tag{3.16}$$

Note that $c_{s/2} = (-1, 1, -1, \dots, -1, 1)'$ and we do not consider $s_{s/2}$ since this would be a vector of zeros. The number of vectors defined in (3.16) is $1 + 2(\frac{s}{2} - 1) + 1 = s$ and, by using standard trigonometric identities, one can show that these vectors are orthogonal. In turn, this implies that every vector in \mathbb{R}^s has a unique representation as a linear combination of $e_0, \dots, c_{s/2}$, which we will write as

$$\alpha = a_0 e_0 + \sum_{j=1}^{s/2-1} (a_j c_j + b_j s_j) + a_{s/2} c_{s/2}. \tag{3.17}$$

There are two distinct advantages in using this basis of \mathbb{R}^s over the canonical one. The first is that the extension of the basis vectors to periodic functions can be obtained naturally in terms of the trigonometric functions involved in their definition. Consider for example s_j : for any $1 \leq t \leq s$, its t th component is $s_j(t) = \sin(2\pi t j/s)$. To extend s_j to a periodic function we must define $s_j(t + ks) = s_j(t)$. But, since

$$\sin \frac{2\pi(t + ks)j}{s} = \sin \left(\frac{2\pi t j}{s} + 2\pi k j \right) = \sin \frac{2\pi t j}{s},$$

the extension of $s_j(t)$ to any integer t is achieved simply by plugging the argument t into the trigonometric expression defining s_j . The second and more substantial advantage of using this new basis composed by trigonometric functions is that the basis vectors now go from smoothest to roughest: at the two extremes we have the constant vector e_0 , which corresponds to a constant periodic function, and the maximally oscillating $c_{s/2}$, which corresponds to a periodic function bouncing back and forth between -1 and 1. To understand what happens in between these two extremes, let us focus on the c_j 's; similar considerations can be made regarding the s_j 's. For a fixed j , as t goes from 1 to s , $2\pi t j/s$ increases monotonically from $2\pi j/s$ to $2\pi j$. Furthermore, the trigonometric functions sine and cosine are periodic, with period 2π . So, considering $j = 1$, the values $2\pi t/s$, $t = 1, \dots, s$, consist of s equally spaced points in the interval $(0, 2\pi]$. For $j = 2$, $2\pi t j/s$ runs up to 4π , spanning two times the basic period 2π of the cosine function and therefore being twice as wiggling.

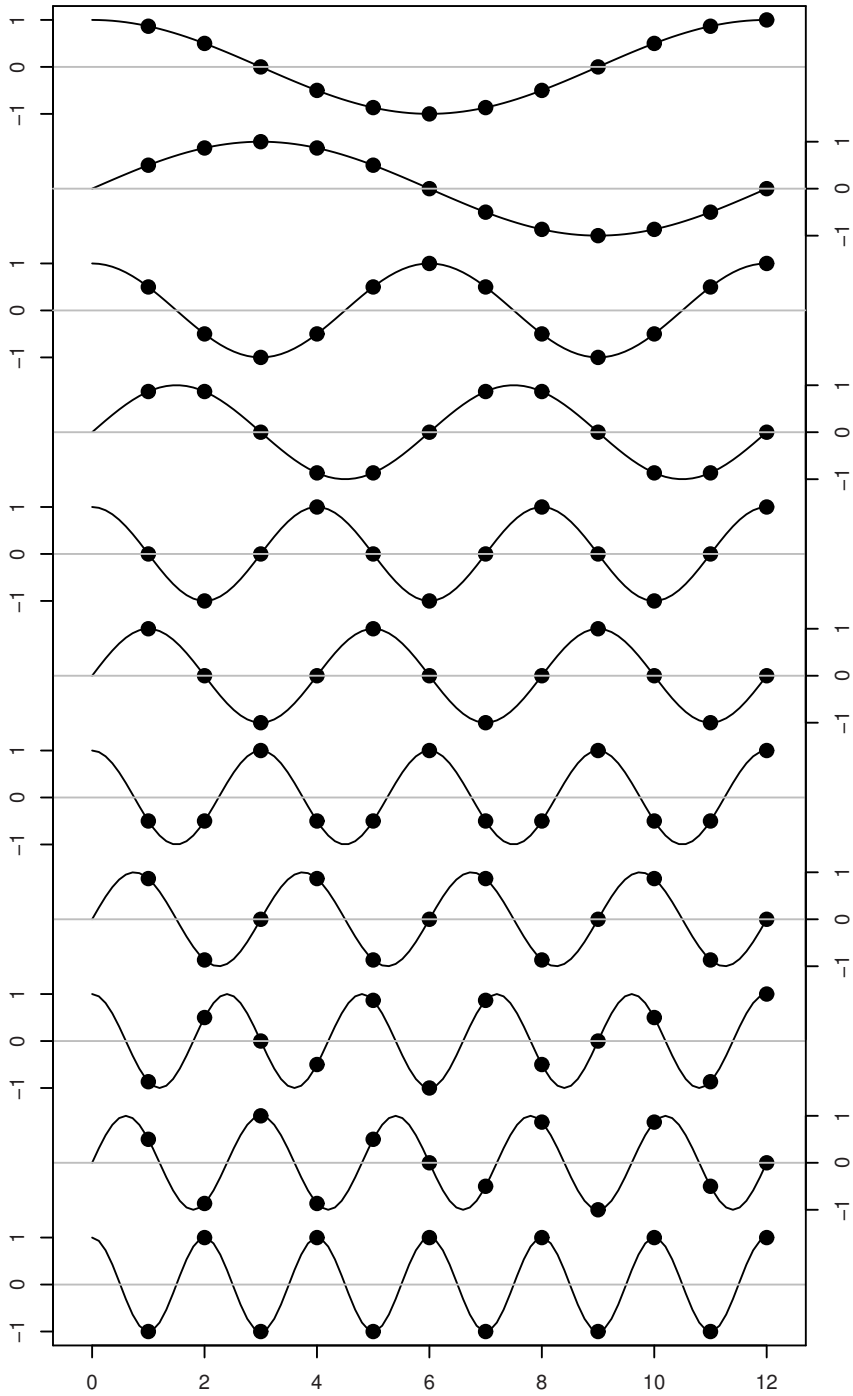


Fig. 3.7. Trigonometric basis functions

In general, for any j , the argument $2\pi tj/s$ spans the interval $(0, 2\pi j]$ and the corresponding cosine function c_j goes through a complete period exactly j times as t goes from 1 to s . Therefore, a higher value of j corresponds to a rougher c_j , in the sense that the function displays more frequent oscillations. To illustrate this feature for $s = 12$, Figure 3.7 shows, from top to bottom, the graphs of $c_1, s_1, \dots, c_5, s_5, c_6$.

It is convenient to group together in the representation (3.17) the terms involving c_j and s_j , which are functions oscillating with the same frequency. For $j = 1, \dots, s/2$, define the j th harmonic of g_t by

$$S_j(t) = a_j \cos(t\omega_j) + b_j \sin(t\omega_j), \quad (3.18)$$

where we take $b_{s/2} = 0$. Moreover, we will assume that $a_0 = 0$, since in the DLM framework—as in the classical time series decomposition—the mean is typically modelled separately from the seasonal component. Note that, in view of the orthogonality of the basis vectors, the sum of any harmonic over an entire period is zero. With these assumptions, we can write

$$g_t = \sum_{j=1}^{s/2} S_j(t). \quad (3.19)$$

The last step, in order to use in the context of DLMS the representation of seasonality in terms of harmonics discussed above, is to study, for a fixed j , the temporal dynamics of $S_j(t)$. The evolution of S_j from time t to time $t + 1$ is given by

$$S_j(t) \mapsto S_j(t + 1) = a_j \cos((t + 1)\omega_j) + b_j \sin((t + 1)\omega_j).$$

If $j < s/2$, it is easy to realize that, from the knowledge of $S_j(t)$ alone, i.e., without knowing a_j and b_j individually, it is impossible to determine the value of $S_j(t + 1)$. However, if in addition to $S_j(t)$ one also knows the conjugate harmonic

$$S_j^*(t) = -a_j \sin(t\omega_j) + b_j \cos(t\omega_j),$$

then one can explicitly compute $S_j(t + 1)$ and also $S_j^*(t + 1)$. In fact,

$$\begin{aligned} S_j(t + 1) &= a_j \cos((t + 1)\omega_j) + b_j \sin((t + 1)\omega_j) \\ &= a_j \cos(t\omega_j + \omega_j) + b_j \sin(t\omega_j + \omega_j) \\ &= a_j (\cos(t\omega_j) \cos \omega_j - \sin(t\omega_j) \sin \omega_j) \\ &\quad + b_j (\sin(t\omega_j) \cos \omega_j + \cos(t\omega_j) \sin \omega_j) \\ &= (a_j \cos(t\omega_j) + b_j \sin(t\omega_j)) \cos \omega_j \\ &\quad + (-a_j \sin(t\omega_j) + b_j \cos(t\omega_j)) \sin \omega_j \\ &= S_j(t) \cos \omega_j + S_j^*(t) \sin \omega_j \end{aligned} \quad (3.20a)$$

and

$$\begin{aligned}
S_j^*(t+1) &= -a_j \sin((t+1)\omega_j) + b_j \cos((t+1)\omega_j) \\
&= -a_j(\sin(t\omega_j)\cos\omega_j + \cos(t\omega_j)\sin\omega_j) \\
&\quad + b_j(\cos(t\omega_j)\cos\omega_j - \sin(t\omega_j)\sin\omega_j) \\
&= (-a_j\sin(t\omega_j) + b_j\cos(t\omega_j))\cos\omega_j \\
&\quad - (a_j\cos(t\omega_j) + b_j\sin(t\omega_j))\sin\omega_j \\
&= -S_j(t)\sin\omega_j + S_j^*(t)\cos\omega_j.
\end{aligned} \tag{3.20b}$$

The two equations (3.20) can be combined in the matrix equation

$$\begin{bmatrix} S_j(t+1) \\ S_j^*(t+1) \end{bmatrix} = \begin{bmatrix} \cos\omega_j & \sin\omega_j \\ -\sin\omega_j & \cos\omega_j \end{bmatrix} \begin{bmatrix} S_j(t) \\ S_j^*(t) \end{bmatrix}.$$

In this form, the j th harmonic fits naturally in the DLM framework by considering the bivariate state vector $(S_j(t), S_j^*(t))'$ with evolution matrix

$$H_j = \begin{bmatrix} \cos\omega_j & \sin\omega_j \\ -\sin\omega_j & \cos\omega_j \end{bmatrix}$$

and observation matrix $F = [1 \ 0]$. The case $j = s/2$ is even simpler, since

$$S_{s/2}(t+1) = \cos((t+1)\pi) = -\cos(t\pi) = -S_{s/2}(t).$$

That is, $S_{s/2}$ simply changes sign at every unit time increment. As a DLM, we can consider this to correspond to a univariate state vector with evolution matrix $H_{s/2} = [-1]$ and observation matrix $F = [1]$.

The DLM representations of the different harmonics can be combined to obtain back (3.19). To this aim, we can consider the state vector

$$\theta_t = (S_1(t), S_1^*(t), \dots, S_{\frac{s}{2}-1}(t), S_{\frac{s}{2}-1}^*(t), S_{\frac{s}{2}}(t))', \quad t = 0, 1, \dots,$$

together with the evolution matrix

$$G = \text{blockdiag}(H_1, \dots, H_{\frac{s}{2}})$$

and observation matrix

$$F = [1 \ 0 \ 1 \ 0 \ \dots \ 0 \ 1].$$

Setting to zero all the evolution and observation variances, the above definitions give a DLM representation of a periodic seasonal component. Non-zero evolution variances may be included to account for a stochastically evolving seasonal component. In this case, strictly speaking, the seasonal component will no longer be periodic. However, the forecast function will, see Problem 3.9. In any case,

$$\theta_0 = (a_1, b_1, \dots, a_{\frac{s}{2}-1}, b_{\frac{s}{2}-1}, a_{\frac{s}{2}})'.$$

By appropriately selecting θ_0 , the representation above can give just any zero-mean periodic function of period s . In most applications, however, we want to

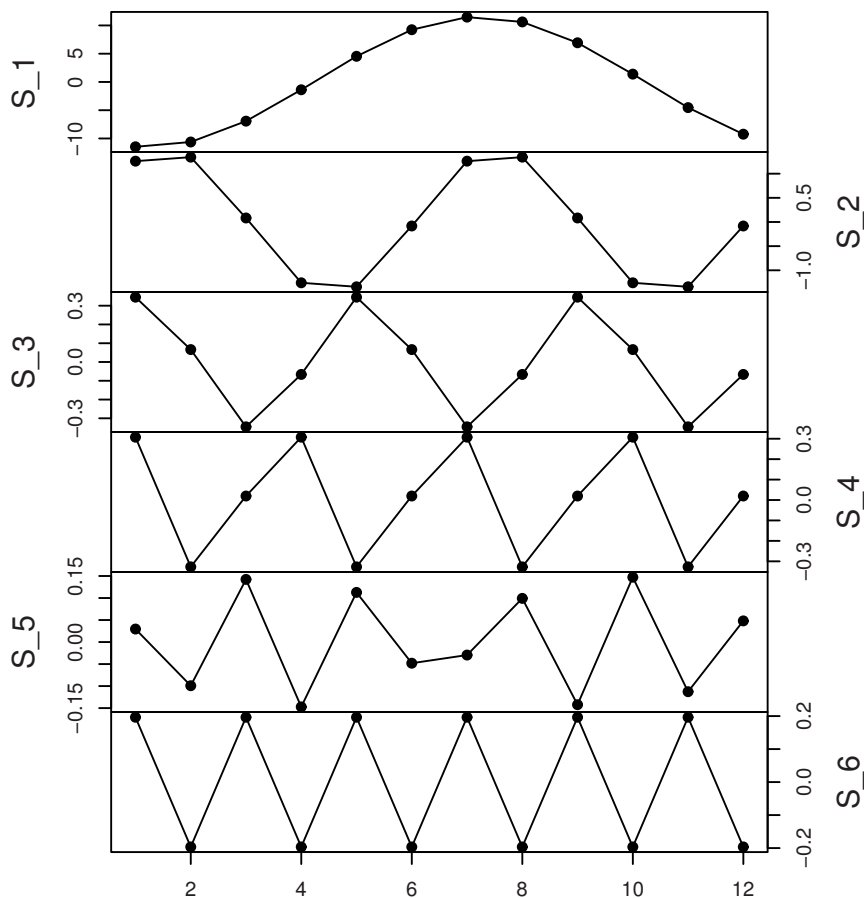


Fig. 3.8. Harmonics of average monthly temperature at Nottingham

model a fairly smooth seasonal component. This can be achieved by discarding a number of high-frequency harmonics and retaining only a few of the first harmonics, which are those that oscillate more slowly (cf. Figure 3.7). In this way we have a parsimonious representation of a periodic component, not available when using the seasonal factors of Section 3.2.2.

To create in R the representation of a Fourier-form seasonal DLM, we can use the function `d1mModTrig`, specifying the period via the argument `s`. In addition, we may specify the number of harmonics to retain in the DLM via the argument `q`. To illustrate how the Fourier form of a seasonal component can be used to obtain a more parsimonious model, consider the data set of average monthly temperatures at Nottingham, `nottem`. As a first model, we use a full 12-month static seasonal component added to a local level. The non-

zero variances were estimated by maximum likelihood. Figure 3.8 shows the six harmonics of the seasonal component over a complete 12-month period.

R code

```

> mod1 <- dlmModTrig(s = 12, dV = 5.1118, dW = 0) +
+   dlmModPoly(1, dV = 0, dW = 81307e-3)
> smoothTem1 <- dlmSmooth(nottem, mod1)
> plot(ts(smoothTem1$s[2 : 13, c(1, 3, 5, 7, 9, 11)],
+   names = paste("S", 1 : 6, sep = "_")),
+   oma.multi = c(2, 0, 1, 0), pch = 16, nc = 1,
+   yax.flip = TRUE, type = 'o', xlab = "", main = "")

```

From the plot we can see that the harmonics after the second have a relatively small amplitude. In view of this, we can try a model containing only the first two harmonics, in addition to the local level representing the process mean. As the code below shows, the MAPE of this reduced model is slightly lower than that of the full model: dropping seven components from the state vector did not affect much the prediction capability of the model—in fact it helped improving it.

R code

```

> mod2 <- dlmModTrig(s = 12, q = 2, dV = 5.1420, dW = 0) +
+   dlmModPoly(1, dV = 0, dW = 81942e-3)
> mean(abs(residuals(dlmFilter(nottem, mod1),
+   type = "raw", sd = FALSE)) / nottem)
[1] 0.08586188
> mean(abs(residuals(dlmFilter(nottem, mod2),
+   type = "raw", sd = FALSE)) / nottem)
[1] 0.05789139

```

An explanation for the lower MAPE of the reduced model is that the higher-order harmonics are basically used to fit the noise in the data, and this fitted noise does not generalize well when it comes to making out-of-sample predictions, in particular one-step-ahead predictions.

So far in our discussion of periodic functions we have considered an even period. The case of an odd period can be treated in essentially the same way. The only difference is in the last harmonic which, instead of being a cosine function only, also has a sine component and has therefore the same form as the previous ones. More specifically, a zero-mean periodic function having odd period s has the representation

$$\alpha(t) = \sum_{j=1}^{(s-1)/2} S_j(t),$$

where the S_j are defined by (3.18). Each harmonic, including the last, has two degrees of freedom, expressed by the coefficients a_j and b_j . The DLM representation parallels the one outlined above in the even-period case, with the difference that the last diagonal block of the evolution matrix G is the 2×2 matrix

$$H_{\frac{s-1}{2}} = \begin{bmatrix} \cos \omega \frac{s-1}{2} & \sin \omega \frac{s-1}{2} \\ -\sin \omega \frac{s-1}{2} & \cos \omega \frac{s-1}{2} \end{bmatrix}$$

and the observation matrix is

$$F = [1 \ 0 \ 1 \ 0 \ \dots \ 0 \ 1 \ 0].$$

3.2.4 General periodic components

The treatment of seasonal components given in the previous section is perfectly appropriate when the period of the process underlying the observations is a multiple of the time between consecutive observations. However, for many natural phenomena this is not the case. Consider, for example, the data set of monthly sunspots available in R as `sunspots`. A plot of the data reveals very clearly some kind of periodicity of about eleven years, but it would be naïf to think that the period consists of an integer number of months. After all, a month is a totally arbitrary time scale to measure something happening on the Sun. In cases like this, it is useful to think of an underlying periodic continuous-time process, $g(t)$, observed at discrete time intervals. A periodic function defined on the real line can be expressed as a sum of harmonics similar to (3.17) or (3.19). In fact, it can be proved that for a continuous periodic function $g(t)$ one has the representation

$$g(t) = a_0 + \sum_{j=1}^{\infty} (a_j \cos(t\omega_j) + b_j \sin(t\omega_j)), \quad (3.21)$$

where $\omega_j = j\omega$ and ω is the so-called *fundamental frequency*. The fundamental frequency is related to τ , the period of $g(t)$, by the relationship $\tau\omega = 2\pi$. In fact, it is easy to see that, if $t_2 = t_1 + k\tau$ for some integer k , then

$$t_2\omega_j = t_2j\omega = (t_1 + k\tau)j\omega = t_1j\omega + kj\tau\omega = t_1\omega_j + kj \cdot 2\pi$$

and, therefore, $g(t_2) = g(t_1)$. Assuming, as we did in Section 3.2.3, that $a_0 = 0$, that is, that $g(t)$ has mean zero, and defining the j th harmonic of $g(t)$ to be

$$S_j(t) = a_j \cos(t\omega_j) + b_j \sin(t\omega_j),$$

we can rewrite (3.21) as

$$g(t) = \sum_{j=1}^{\infty} S_j(t). \quad (3.22)$$

The discrete-time evolution of the j th harmonic from time t to time $t + 1$ can be cast as a DLM in the same way as we did in Section 3.2.3. We just have to consider the bivariate state vector $(S_j(t), S_j^*(t))'$, where $S_j^*(t)$ is the conjugate harmonic defined by

$$S_j^*(t) = -a_j \sin(t\omega_j) + b_j \cos(t\omega_j),$$

and the evolution matrix

$$H_j = \begin{bmatrix} \cos \omega_j & \sin \omega_j \\ -\sin \omega_j & \cos \omega_j \end{bmatrix}.$$

Then we have

$$\begin{bmatrix} S_j(t+1) \\ S_j^*(t+1) \end{bmatrix} = H_j \begin{bmatrix} S_j(t) \\ S_j^*(t) \end{bmatrix}.$$

A nonzero evolution error may be added to this evolution equation to account for a stochastically varying harmonic.

Clearly, in a DLM we cannot keep track of the infinitely many harmonics in the representation (3.22): we have to truncate the infinite series on the RHS to a finite sum of, say, q terms. As already seen in the previous section, the higher-order harmonics are those oscillating more rapidly. Thus, the decision of truncating the infinite sum in (3.22) can be interpreted as a subjective judgement about the degree of smoothness of the function $g(t)$. In practice it is not uncommon to model the periodic function $g(t)$ using only one or two harmonics.

To set up in R a general periodic DLM component, the function `dlmModTrig` introduced in Section 3.2.3 may be used. The user needs to specify the period via the argument `tau` and the number of harmonics, `q`. Alternatively, the fundamental frequency ω can be specified, via the argument `om`, instead of the period τ . The display below shows how to specify a two-harmonic stochastic periodic component added to a local level for the sunspots data, on the square root scale. The period $\tau = 130.51$ as well as the nonzero variances were estimated by maximum likelihood.

R code

```
> mod <- dlmModTrig(q = 2, tau = 130.51, dV = 0,
+               dW = rep(c(1765e-2, 3102e-4), each = 2)) +
+               dlmModPoly(1, dV = 0.7452, dW = 0.1606)
```

The model can be used, for example, to smooth the data, extracting the level (the “signal”) from the data. Figure 3.9, obtained with the code below, shows the data, the level, and the general stochastic periodic component. The level is the last (fifth) component of the state vector, while the periodic component is obtained by adding the two harmonics of the model (first and third components).

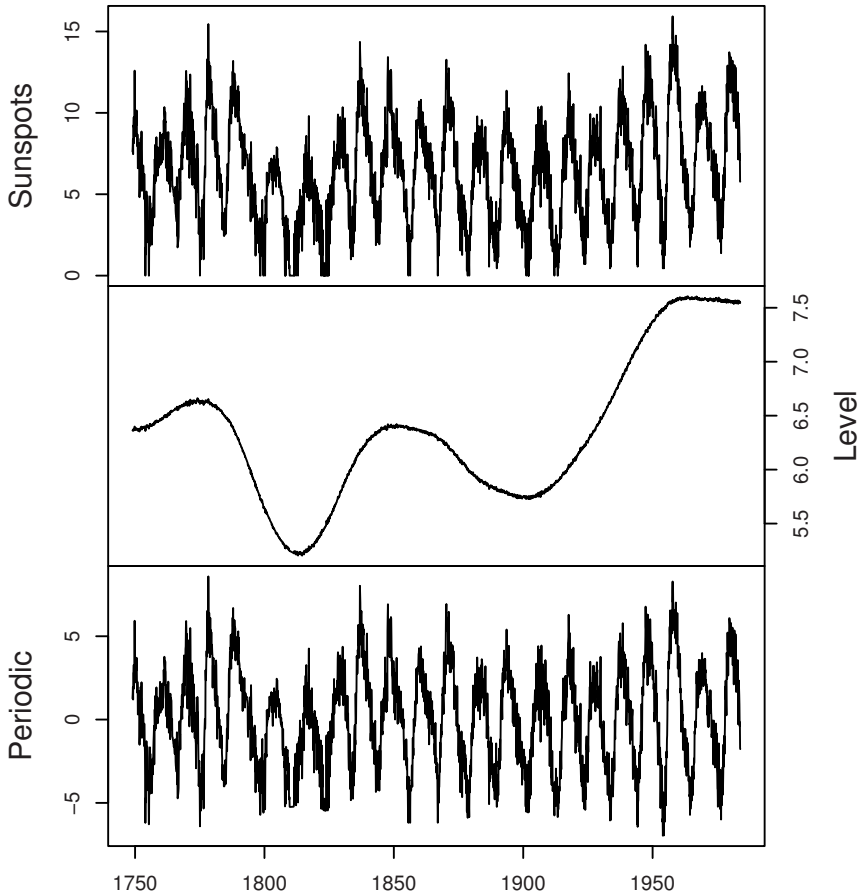


Fig. 3.9. Monthly sunspot numbers

R code

```

> sspots <- sqrt(sunspots)
2 > sspots.smooth <- dlmSmooth(sspots, mod)
> y <- cbind(sspots,
4 +         tcrossprod(dropFirst(sspots.smooth$s[, c(1, 3, 5)]),
+           matrix(c(0, 0, 1, 1, 1, 0), nr = 2,
6 +             byrow = TRUE)))
> colnames(y) <- c("Sunspots", "Level", "Periodic")
8 > plot(y, yax.flip = TRUE, oma.multi = c(2, 0, 1, 0))

```

3.2.5 DLM representation of ARIMA models

Any ARIMA model can be expressed as a DLM. More precisely, for any ARIMA process, it is possible to find a DLM whose measurement process (Y_t) has the same distribution as the given ARIMA. The state space with its dynamics is not uniquely determined: several representations have been proposed in the literature and are in use. Here we will present only one of them, which is probably the most widely used. For alternative representations the reader can consult [Gourieroux and Monfort \(1997\)](#).

Let us start with the stationary case. Consider the ARMA(p, q) process defined by (3.4), assuming for simplicity that μ is zero. The defining relation can be written as

$$Y_t = \sum_{j=1}^r \phi_j Y_{t-j} + \sum_{j=1}^{r-1} \psi_j \epsilon_{t-j} + \epsilon_t,$$

with $r = \max\{p, q + 1\}$, $\phi_j = 0$ for $j > p$ and $\psi_j = 0$ for $j > q$. Define the matrices

$$\begin{aligned} F &= [1 \ 0 \ \dots \ 0], \\ G &= \begin{bmatrix} \phi_1 & 1 & 0 & \dots & 0 \\ \phi_2 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \phi_{r-1} & 0 & \dots & 0 & 1 \\ \phi_r & 0 & \dots & 0 & 0 \end{bmatrix}, \\ R &= [1 \ \psi_1 \ \dots \ \psi_{r-2} \ \psi_{r-1}]'. \end{aligned} \tag{3.23}$$

If one introduces an r -dimensional state vector $\theta_t = (\theta_{1,t}, \dots, \theta_{r,t})'$, then the given ARMA model has the following DLM representation:

$$\begin{cases} Y_t = F\theta_t, \\ \theta_{t+1} = G\theta_t + R\epsilon_t. \end{cases} \tag{3.24}$$

This is a DLM with $V = 0$ and $W = RR'\sigma^2$, where σ^2 is the variance of the error sequence (ϵ_t). To verify this equivalence, note that the observation equation gives $y_t = \theta_{1,t}$ and the state equation is

$$\begin{aligned} \theta_{1,t} &= \phi_1 \theta_{1,t-1} + \theta_{2,t-1} + \epsilon_t \\ \theta_{2,t} &= \phi_2 \theta_{1,t-1} + \theta_{3,t-1} + \psi_1 \epsilon_t \\ &\vdots \\ \theta_{r-1,t} &= \phi_{r-1} \theta_{1,t-1} + \theta_{r,t-1} + \psi_{r-2} \epsilon_t \\ \theta_{r,t} &= \phi_r \theta_{1,t-1} + \psi_{r-1} \epsilon_t. \end{aligned} \tag{3.25}$$

Substituting the expression of $\theta_{2,t-1}$, obtained from the second equation, in the first equation, we have

$$\theta_{1,t} = \phi_1\theta_{1,t-1} + \phi_2\theta_{1,t-2} + \theta_{3,t-2} + \psi_1\epsilon_{t-1} + \epsilon_t$$

and proceeding by successive substitutions we eventually get

$$\theta_{1,t} = \phi_1\theta_{1,t-1} + \dots + \phi_r\theta_{1,t-r} + \psi_1\epsilon_{t-1} + \dots + \psi_{r-1}\epsilon_{t-r-1} + \epsilon_t .$$

Recalling that $r = \max\{p, q + 1\}$ and $y_t = \theta_{1,t}$ we see that this is the ARMA model (3.4).

The DLM representation (3.24) might appear quite artificial. To develop a better understanding, let us look at the simpler case of pure autoregressive models, for example the AR(2) model

$$Y_t = \phi_1 Y_{t-1} + \phi_2 Y_{t-2} + \epsilon_t, \quad \epsilon_t \stackrel{iid}{\sim} \mathcal{N}(0, \sigma^2). \quad (3.26)$$

One might think of a simpler representation of the AR(2) as a DLM with observation equation having $F_t = [Y_{t-1}, Y_{t-2}]$ and $\theta_t = [\phi_{1,t}, \phi_{2,t}]'$ (thus, possibly including a temporal evolution of the AR parameters, otherwise letting $W = 0$). However, the matrix F_t of a DLM cannot depend on past values of the observations: in our case, the above choice of F_t would imply

$$Y_t | y_{t-1}, y_{t-2}, \theta_t \sim N(\phi_1 y_{t-1} + \phi_2 y_{t-2}, \sigma^2),$$

that is, Y_t would not be independent on the past values y_{t-1}, y_{t-2} given θ_t , thus violating assumption (A.2) in the definition of state space model (p.40). In order to consider such a model we would have to extend the definition to include *conditionally* Gaussian state space models (Lipster and Shirayev; 1972). In order to stay within the boundaries of the standard definition used in this book, the first trick that we have used in the DLM representation (3.4) is thus to “shift” the AR(2) dependence from Y_t to the state vector: $Y_t = \theta_{1,t}$. However, the second basic assumption of state space models is that the state process is Markovian, thus we need a second trick for representing the second order dependence. We augmented the state vector by a second component $\theta_{2,t}$, and we chose G and W such that

$$\begin{bmatrix} \theta_{1,t} \\ \theta_{2,t} \end{bmatrix} = \begin{bmatrix} \phi_1 & 1 \\ \phi_2 & 0 \end{bmatrix} \begin{bmatrix} \theta_{1,t-1} \\ \theta_{2,t-1} \end{bmatrix} + \begin{bmatrix} \epsilon_t \\ 0 \end{bmatrix},$$

giving

$$\theta_{1,t} = \phi_1\theta_{1,t-1} + \phi_2\theta_{1,t-2} + \epsilon_t.$$

In this way, we obtain a DLM representation of AR(p) models in the framework of state-space models as defined by assumptions (A.1) and (A.2) in Section 2.3. A further step is needed for expressing the MA(q) component. For example, for the ARMA(1,1) model

$$Y_t = \phi_1 Y_{t-1} + \epsilon_t + \psi_1 \epsilon_{t-1}, \quad \epsilon_t \sim \mathcal{N}(0, \sigma^2), \quad (3.27)$$

$r = q + 1 = 2$ and the matrices of the corresponding DLM are

$$\begin{aligned}
 F &= [1 \ 0], & V &= 0, \\
 G &= \begin{bmatrix} \phi_1 & 1 \\ 0 & 0 \end{bmatrix}, & W &= \begin{bmatrix} 1 & \psi_1 \\ \psi_1 & \psi_1^2 \end{bmatrix} \sigma^2.
 \end{aligned} \tag{3.28}$$

Representing an ARMA model as a DLM is useful mainly for two reasons. The first is that an ARMA component in a DLM can explain residual autocorrelation not accounted for by other structural components such as trend and seasonal. The second reason is technical, and consists in the fact that the evaluation of the likelihood function of an ARMA model can be performed efficiently by applying the general recursion used to compute the likelihood of a DLM.

The DLM representation of an ARIMA(p, d, q) model, with $d > 0$, can be derived as an extension of the stationary case. In fact, if one considers $Y_t^* = \Delta^d Y_t$, then Y_t^* follows a stationary ARMA model, for which the DLM representation given above applies. In order to model the original series (Y_t) we need to be able to recover it from the Y_t^* and possibly other components of the state vector. For example, if $d = 1$, $Y_t^* = Y_t - Y_{t-1}$ and therefore $Y_t = Y_t^* + Y_{t-1}$. Suppose that Y_t^* satisfies the AR(2) model (3.26). Then a DLM representation for Y_t is given by the system

$$\begin{cases} Y_t = [1 \ 1 \ 0] \theta_{t-1}, \\ \theta_t = \begin{bmatrix} 1 & 1 & 0 \\ 0 & \phi_1 & 0 \\ 0 & \phi_2 & 1 \end{bmatrix} \theta_{t-1} + w_t, \quad w_t \sim \mathcal{N}(0, W), \end{cases} \tag{3.29}$$

with

$$\theta_t = \begin{bmatrix} Y_{t-1} \\ Y_t^* \\ \phi_2 Y_{t-1}^* \end{bmatrix} \tag{3.30}$$

and $W = \text{diag}(0, \sigma^2, 0)$. For a general d , set $Y_t^* = \Delta^d Y_t$. It can be shown that the following relation holds:

$$\Delta^{d-j} Y_t = Y_t^* + \sum_{i=1}^j \Delta^{d-i} Y_{t-1}, \quad j = 1, \dots, d. \tag{3.31}$$

Define the state vector as follows:

$$\theta_t = \begin{bmatrix} Y_{t-1} \\ \Delta Y_{t-1} \\ \vdots \\ \Delta^{d-1} Y_{t-1} \\ Y_t^* \\ \phi_2 Y_{t-1}^* + \dots + \phi_r Y_{t-r+1}^* + \psi_1 \epsilon_t = \dots + \psi_{r-1} \epsilon_{t-r+2} \\ \phi_3 Y_{t-1}^* + \dots + \phi_r Y_{t-r+2}^* + \psi_2 \epsilon_t = \dots + \psi_{r-1} \epsilon_{t-r+3} \\ \vdots \\ \phi_r Y_{t-1}^* + \psi_{r-1} \epsilon_t \end{bmatrix}. \tag{3.32}$$

Note that the definition of the last components of θ_t follows from equations (3.25). The system and observation matrices, together with the system variance, are defined by

$$F = [1 \ 1 \ \dots \ 1 \ 0 \ \dots \ 0],$$

$$G = \begin{bmatrix} 1 & 1 & \dots & 1 & 0 & \dots & 0 \\ 0 & 1 & \dots & 1 & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & 1 & 1 & 0 & \dots & 0 \\ 0 & \dots & 0 & \phi_1 & 1 & 0 & \dots & 0 \\ \dots & \dots & \dots & \phi_2 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ \dots & \dots & \phi_{r-1} & 0 & \dots & 0 & 1 \\ 0 & \dots & 0 & \phi_r & 0 & \dots & 0 & 0 \end{bmatrix}, \tag{3.33}$$

$$R = [0 \ \dots \ 0 \ 1 \ \psi_1 \ \dots \ \psi_{r-2} \ \psi_{r-1}]',$$

$$W = RR'\sigma^2.$$

With the above definition the ARIMA model for (Y_t) has the DLM representation

$$\begin{cases} Y_t = F\theta_t, \\ \theta_t = G\theta_{t-1} + w_t, \quad w_t \sim \mathcal{N}(0, W). \end{cases} \tag{3.34}$$

Since in DLM modeling a nonstationary behavior of the observations is usually accounted for directly, through the use of a polynomial trend or a seasonal component for example, the inclusion of nonstationary ARIMA components is less common than that of stationary ARMA components that, as we already mentioned, are typically used to capture correlated noise in the data.

3.2.6 Example: estimating the output gap

Measuring the so-called *output gap* is an important issue in monetary politics. The output gap is the difference between the observed Gross Domestic Product of a country (GDP, or output) and the *potential output* of the economy.

It is this discrepancy that is considered relevant in determining inflationary tendencies. Thus, it is crucial to be able to separate the two components and, since they are unobservable, treating them as latent states in a DLM results particularly attractive.

Variants of the following model for the deseasonalized output have been considered in the econometric literature; see, e.g., Kuttner (1994):

$$\begin{aligned} Y_t &= Y_t^{(p)} + Y_t^{(g)}, \\ Y_t^{(p)} &= Y_{t-1}^{(p)} + \delta_t + \epsilon_t, & \epsilon_t &\sim \mathcal{N}(0, \sigma_\epsilon^2), \\ \delta_t &= \delta_{t-1} + z_t, & z_t &\sim \mathcal{N}(0, \sigma_z^2), \\ Y_t^{(g)} &= \phi_1 Y_{t-1}^{(g)} + \phi_2 Y_{t-2}^{(g)} + u_t, & u_t &\sim \mathcal{N}(0, \sigma_u^2), \end{aligned} \quad (3.35)$$

where Y_t is the logarithm of the output, $Y_t^{(p)}$ represents the log-potential output and $Y_t^{(g)}$ is the log of the output gap. The above model can be seen as a DLM, obtained by adding, in the sense discussed in Section 3.2, a stochastic trend component for the potential output and a stationary AR(2) residual component. More specifically,

- $Y_t^{(p)}$ follows a linear growth model, observed without error. The state vector is $\theta_t^{(p)} = (Y_t^{(p)}, \delta_t)'$, with innovation vector $w_t^{(p)} = (\epsilon_t, z_t)'$. The observation matrix and observation variance are

$$F^{(p)} = [1 \ 0], \quad V^{(p)} = [0],$$

while the system evolution matrix and innovation variance are

$$G^{(p)} = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}, \quad W^{(p)} = \text{diag}(\sigma_\epsilon^2, \sigma_z^2).$$

The error terms ϵ_t and z_t are interpreted as shocks to the output level and to the output growth rate, respectively. Note that the case $\sigma_\epsilon^2 = \sigma_z^2 = 0$ corresponds to the global trend model $y_t^{(p)} = \mu_0 + t\delta_0$, while $\sigma_\epsilon^2 = 0$ gives an integrated random walk model.

- The output gap $Y_t^{(g)}$ is described by an AR(2) model, which can be written as a DLM with $\theta_t^{(g)} = (Y_t^{(g)}, \theta_{t,2}^{(g)})'$ and

$$\begin{aligned} F^{(g)} &= [1 \ 0], & V^{(g)} &= [0], \\ G^{(g)} &= \begin{bmatrix} \phi_1 & 1 \\ \phi_2 & 0 \end{bmatrix}, & W^{(g)} &= \text{diag}(\sigma_u^2, 0). \end{aligned}$$

The order of the AR process allows the residuals, i.e., the departures from the trend, to have a damped cyclic autocorrelation function, which is often observed in economic time series. The DLM representation of model (3.35) for Y_t is obtained by adding the two components, as described in Section 3.2. The matrices of the resulting DLM are:

$$\begin{aligned}
 F &= [1 \ 0 \ 1 \ 0], \\
 G &= \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \phi_1 & 1 \\ 0 & 0 & \phi_2 & 0 \end{bmatrix}, \\
 V &= [0] \\
 W &= \text{diag}(\sigma_\epsilon^2, \sigma_z^2, \sigma_u^2, 0).
 \end{aligned} \tag{3.36}$$

The resulting DLM has five unknown parameters: σ_ϵ^2 and σ_z^2 for the trend component, and ϕ_1, ϕ_2 and σ_u^2 for the AR(2) component. Here, we estimate the unknown parameters by maximum likelihood (see Chapter 4) and then apply the Kalman filter and smoother using their MLE. In Section 4.6.1 we will illustrate a Bayesian approach to make inference both on the unknown parameters of the model and on the unobservable states at the same time.

For a practical application, we consider quarterly deseasonalized GDP of the US economy (source: Bureau of Economic Analysis), measured in billions of chained 2000 US dollars, from 1950.Q1 to 2004.Q4. A logarithmic transform is applied, and the resulting log GDP is plotted in Figure 3.10.

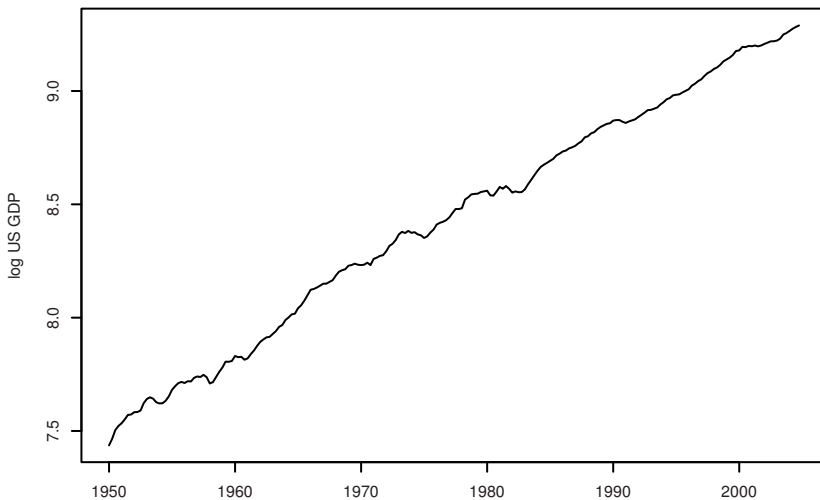


Fig. 3.10. Time plot of log US Gross Domestic Product

R code

```

> gdp <- read.table("Datasets/gdp5004.dat")
2 > gdp <- ts(gdp, frequency = 4, start = 1950)

```

```

> Lgdp <- log(gdp)
4 > plot(Lgdp, xlab = "", ylab = "log US GDP")

```

We first compute the MLE of the unknown parameters $\psi_1 = \log(\sigma_\epsilon^2)$, $\psi_2 = \log(\sigma_z^2)$, $\psi_3 = \log(\sigma_u^2)$, $\psi_4 = \phi_1$, $\psi_5 = \phi_2$. For the moment, we do not impose stationarity restrictions on the AR(2) parameters ϕ_1, ϕ_2 ; we will return to this later. We choose convenient values for the initial mean $m_0^{(p)}$ of the states of the trend component, with a fairly large variance $C_0^{(p)}$, and use the default initial values in the R function `dlmModArma` for the AR(2) component. The MLE fitting function, as well as `dlmFilter`, do not allow singular observation variances, so we take a very small value for V , which can be considered zero for all practical purposes.

R code

```

> level0 <- Lgdp[1]
2 > slope0 <- mean(diff(Lgdp))
> buildGap <- function(u) {
4 +   trend <- dlmModPoly(dV = 1e-7, dW = exp(u[1 : 2]),
+                       m0 = c(level0, slope0),
6 +                       C0 = 2 * diag(2))
+   gap <- dlmModARMA(ar = u[4 : 5], sigma2 = exp(u[3]))
8 +   return(trend + gap)}
> init <- c(-3, -1, -3, .4, .4)
10 > outMLE <- dlmMLE(Lgdp, init, buildGap)
> dlmGap <- buildGap(outMLE$par)
12 > sqrt(diag(W(dlmGap))[1 : 3])
[1] 5.781792e-03 7.637864e-05 6.145361e-03
14 > GG(dlmGap)[3 : 4, 3]
[1] 1.4806246 -0.5468102

```

Thus the MLE estimates are $\hat{\sigma}_\epsilon = 0.00578$, $\hat{\sigma}_z = 0.00008$, $\hat{\sigma}_u = 0.00615$, $\hat{\phi}_1 = 1.481$, $\hat{\phi}_2 = -0.547$. The estimated AR(2) parameters satisfy the stationarity constraints.

R code

```

> Mod(polyroot(c(1, -GG(dlmGap)[3 : 4, 3])))
2 [1] 1.289162 1.418587
> plot(ARMAacf(ar = GG(dlmGap)[3 : 4, 3], lag.max = 20),
4 +   ylim = c(0, 1), ylab = "acf", type = "h")

```

A note of caution is in order. There are clearly identifiability issues in separating the stochastic trend and the AR(2) residuals, and MLE is not stable; as a minimal check, one should repeat the fitting procedure starting from different sets of initial values.

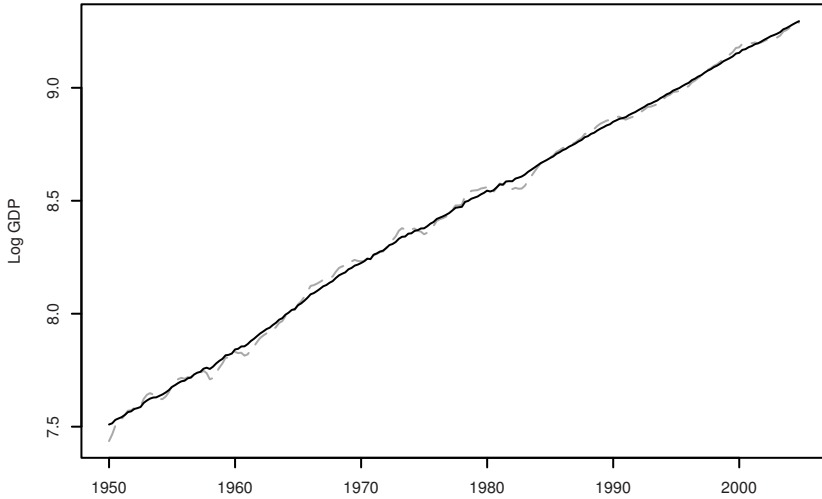


Fig. 3.11. Observed log GDP (grey) and smoothing estimate of the potential output

Plugging in the MLE, smoothing estimates of the model are finally computed. The observed output and the smoothing estimate of the potential output are plotted in Figure 3.11. Figure 3.12 shows the smoothing estimates of the potential output, together with the output gap (note the different scale).

R code

```

> gdpSmooth <- dlmSmooth(Lgdp, dlmGap)
2 > plot(cbind(Lgdp, dropFirst(gdpSmooth$s[, 1])),
+       xlab = "", ylab = "Log GDP", lty = c("longdash", "solid"),
4 +       col = c("darkgrey", "black"), plot.type = "single")
> plot(dropFirst(gdpSmooth$s[, 1:3]), ann = F, yax.flip = TRUE)

```

Finally, let us see how we could introduce stationarity constraints in the MLE, using the R function `ARtransPars`.

R code

```

> buildgapr <- function(u)
2 + {
+   trend <- dlmModPoly(dV = 0.000001,
4 +                 dW = c(exp(u[1]), exp(u[2])),
+                 m0 = c(Lgdp[1], mean(diff(Lgdp))),
6 +                 CO = 2 * diag(2))
+   gap <- dlmModARMA(ar = ARtransPars(u[4 : 5]),
8 +                 sigma2 = exp(u[3]))
+   return(trend + gap)

```

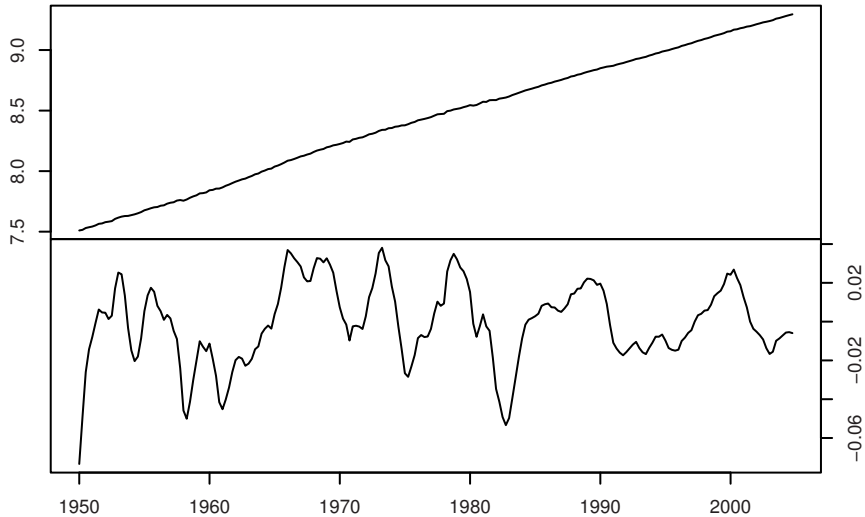


Fig. 3.12. Smoothing estimates of the potential output (top) and output-gap (bottom)

```

10 + }
> init <- c(-3, -1, -3, .4, .4)
12 > outMLER <- dlmMLE(Lgdp,init,buildgapr)
> outMLER$value
14 [1] -896.5234
> parMLER <- c(exp(outMLER$par[1 : 3])^.5,
16 + ARtransPars(outMLER$par[4 : 5]))
> round(parMLER, 4)
18 [1] 0.0051 0.0000 0.0069 1.4972 -0.4972

```

We obtained a slightly better value of the likelihood. Yet, the estimates of the AR(2) parameters are now too close to the nonstationary region. It is anyway a useful exercise to look at the smoothing estimates of the potential output and of the output gap in this case: nonstationarity of the AR(2) residuals gives a nonrealistic estimated pattern for the output gap (figures not shown).

R code

```

> Mod(polyroot(c(1, -ARtransPars(outMLER$par[4 : 5]))))
2 [1] 1.000001 2.011090
> plot(ARMAacf(ar = ARtransPars(outMLER$par[4 : 5]),
4 + lag.max = 10),
+ ylim = c(0, 1), ylab = "acf")
6 > modr <- buildgapr(outMLER$par)

```

```

> outFr <- dlmFilter(Lgdp, modr)
8 > outSr <- dlmSmooth(outFr)
> ts.plot(cbind(Lgdp, outSr$s[-1, 1]), col = 1 : 2)
10 > plot.ts(outSr$s[-1, c(1, 3)])

```

We will return to this example in Chapter 4, showing how we can impose stationarity constraints through a prior distribution on the AR parameters, in a Bayesian approach.

3.2.7 Regression models

Regression components can be quite easily incorporated in a DLM. It is of interest in many applications to include the effect of explanatory variables on a time series (Y_t). For example, in a clinical trial, one may be interested in studying a response Y_t to different drug doses x_t over time. Note that the explanatory variable x_t is nonstochastic; the case of stochastic regression requires a joint model for the bivariate time series $(X_t, Y_t), t \geq 1$ and will be discussed later.

The standard linear regression model for Y on x is defined as

$$Y_t = \beta_1 + \beta_2 x_t + \epsilon_t, \quad \epsilon_t \stackrel{iid}{\sim} \mathcal{N}(0, \sigma^2).$$

However, the assumption of i.i.d. errors is often not realistic if the observations are taken over time. A possible solution is to introduce a temporal dependence for the residuals, e.g., describing $(\epsilon_t : t \geq 1)$ as an autoregressive process. Another choice, which is in fact quite interesting in many problems, is to think that the relationship between y and x evolves over time. That is, to consider a *dynamic* linear regression model of the form

$$Y_t = \beta_{1,t} + \beta_{2,t} x_t + \epsilon_t, \quad \epsilon_t \stackrel{iid}{\sim} \mathcal{N}(0, \sigma^2),$$

and model the temporal evolution of $(\beta_{1,t}, \beta_{2,t})$, e.g., $\beta_{j,t} = \beta_{j,t-1} + w_{j,t}$, $j = 1, 2$, $w_{1,t}$ and $w_{2,t}$ independent. This gives a DLM with $F_t = [1 \ x_t]$, $\theta_t = [\beta_{1,t} \ \beta_{2,t}]'$, $V = \sigma^2$, completed by a state equation for θ_t .

More generally, a dynamic linear regression model is described by

$$\begin{aligned} Y_t &= x_t' \theta_t + v_t, & v_t &\sim \mathcal{N}(0, \sigma_t^2) \\ \theta_t &= G_t \theta_{t-1} + w_t, & w_t &\sim \mathcal{N}_p(0, W_t) \end{aligned}$$

where $x_t' = [x_{1,t} \ \dots \ x_{p,t}]$ are the values of the p explanatory variables at time t . Again, note that x_t is not stochastic; in other terms, this is a conditional model for $Y_t | x_t$. A popular default choice for the state equation is to take the evolution matrix G_t as the identity matrix and W diagonal, which corresponds to modeling the regression coefficients as independent random walks.

A regression DLM, with the above assumptions for G and W , can be created in R through the function `d1mModReg`. The static linear regression model corresponds to the case where $W_t = 0$ for any t , so that θ_t is constant over time, $\theta_t = \theta$, with prior density $\theta \sim \mathcal{N}(m_0, C_0)$. Thus, the function `d1mModReg` can also be used for Bayesian inference in linear regression; the filtering density gives the posterior density of $\theta|y_{1:t}$, and $m_t = E(\theta|y_{1:t})$ is the Bayesian estimate under a quadratic loss function of the regression coefficients. This also suggests that DLM techniques may be used to sequentially update the estimates of the parameters of a (static) regression model as new observations become available.

Example — Capital asset pricing model

The capital asset pricing model (CAPM) is a popular asset pricing tool in financial econometrics; see for example Campbell et al. (1996). In its simplest, univariate version, the CAPM model assumes that the returns on an asset depend linearly on the overall market returns, thus allowing us to study the behavior, in terms of risk and expected returns, of individual assets compared to the market as a whole. Here we consider a dynamic version of the standard, static CAPM model; in Section 3.3.3, it will be extended to a multivariate CAPM for a small portfolio of m assets.

Let r_t , $r_t^{(M)}$ and $r_t^{(f)}$ be the returns at time t of the asset under study, of the market and of a risk-free asset, respectively. Define the *excess returns* of the asset as $y_t = r_t - r_t^{(f)}$ and the market's excess returns as $x_t = r_t^{(M)} - r_t^{(f)}$. A univariate CAPM assumes that

$$y_t = \alpha + \beta x_t + v_t, \quad v_t \stackrel{iid}{\sim} N(0, \sigma^2). \quad (3.37)$$

The parameter β measures the sensitivity of the asset excess returns to changes in the market. A value of β greater than one suggests that the asset tends to amplify changes in the market returns, and it is therefore considered an aggressive investment; while assets having β smaller than one are thought of as conservative investments.

The data for this example consists of monthly returns³ from January 1978 to December 1987 of four stocks (Mobil, IBM, Weyer, and Citicorp), of the 30-day Treasury Bill as a proxy for the risk-free asset, and of the value-weighted average returns for all the stocks listed at the New York and American Stock Exchanges, representing the overall market returns. The data are plotted in Figure 3.13. Let us consider the data for the IBM stock here; a multivariate CAPM will be illustrated later.

Least squares estimates for the static CAPM are obtained in R by the function `lm`. We can also compute Bayesian estimates of the regression coef-

³ The data, originally from Berndt (1991), are available at the time of writing from <http://shazam.econ.ubc.ca/intro/P.txt>.

ficients by the function `dlmModReg`, letting $\text{diag}(W) = (0, 0)$ (while assuming, for simplicity, a known measurement variance).

R code

```

> capm <- read.table("http://shazam.econ.ubc.ca/intro/P.txt",
2 +           header = TRUE)
> capm.ts <- ts(capm, start = c(1978, 1), frequency = 12)
4 > colnames(capm)
[1] "MOBIL" "IBM" "WEYER" "CITCRP" "MARKET" "RKFREE"
6 > plot(capm.ts)
> IBM <- capm.ts[, "IBM"] - capm.ts[, "RKFREE"]
8 > x <- capm.ts[, "MARKET"] - capm.ts[, "RKFREE"]
> outLM <- lm(IBM ~ x)
10 > outLM$coef
      (Intercept)           x
12 -0.0004895937  0.4568207721
> acf(outLM$res)
14 > qqnorm(outLM$res)

```

Despite the data being taken over time, the residuals do not show relevant temporal dependence (plots not shown). Bayesian estimates of the regression coefficients are obtained below. For simplicity, we let $V = \hat{\sigma}^2 = 0.00254$. Prior information on the values of α and β can be introduced through the prior, $\mathcal{N}(m_0, C_0)$; below, we assume vague prior information on α , while the prior guess on β is 1.5 (aggressive investment), with a fairly small variance.

R code

```

> mod <- dlmModReg(x, dV = 0.00254, m0 = c(0, 1.5),
2 +           C0 = diag(c(1e+07, 1)))
> outF <- dlmFilter(IBM, mod)
4 > outF$m[1 + length(IBM), ]
[1] -0.0005232801  0.4615301204

```

The filtering estimates at the end of the observation period are the Bayes estimates, under quadratic loss, of the regression coefficients; here, the results are very close to the OLS estimates. As a matter of fact, it seems more natural to allow the CAPM coefficients α and β to vary over time; for example, a stock might evolve from “conservative” to “aggressive.” A dynamic version of the classical CAPM is obtained as

$$y_t = \alpha_t + \beta_t x_t + v_t, \quad v_t \stackrel{iid}{\sim} N(0, \sigma^2),$$

with a state equation modeling the dynamics of α_t and β_t . As mentioned above, the function `dlmModReg` assumes that the regression coefficients follow independent random walks. Below, the observation and state evolution

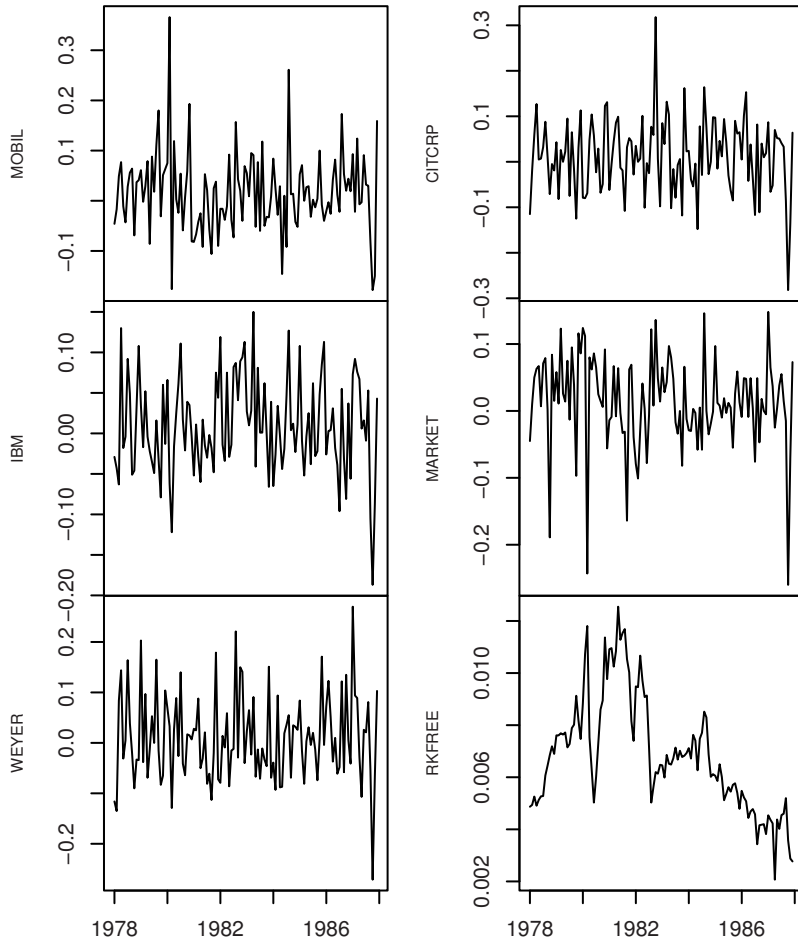


Fig. 3.13. Monthly returns for Mobil, IBM, Weyer and Citicorp stocks; market index; 30-days Treasury bill (January 1978-December 1987)

variances are estimated by maximum likelihood. The smoothing estimates are shown in Figure 3.14. The results can be compared with those obtained in Section 3.3.3, where the model is estimated jointly for the four stocks.

R code

```

> buildCapm <- function(u) {
2 +   dlmModReg(x, dV = exp(u[1]), dW = exp(u[2 : 3]))
+ }
4 > outMLE <- dlmMLE(IBM, parm = rep(0, 3), buildCapm)
> exp(outMLE$par)
6 [1] 2.328397e-03 1.100181e-05 6.496259e-04

```



```

> outMLE$value
8 [1] -276.7014
> mod <- buildCapm(outMLE$par)
10 > outS <- dlmSmooth(IBM, mod)
> plot(dropFirst(outS$s))

```

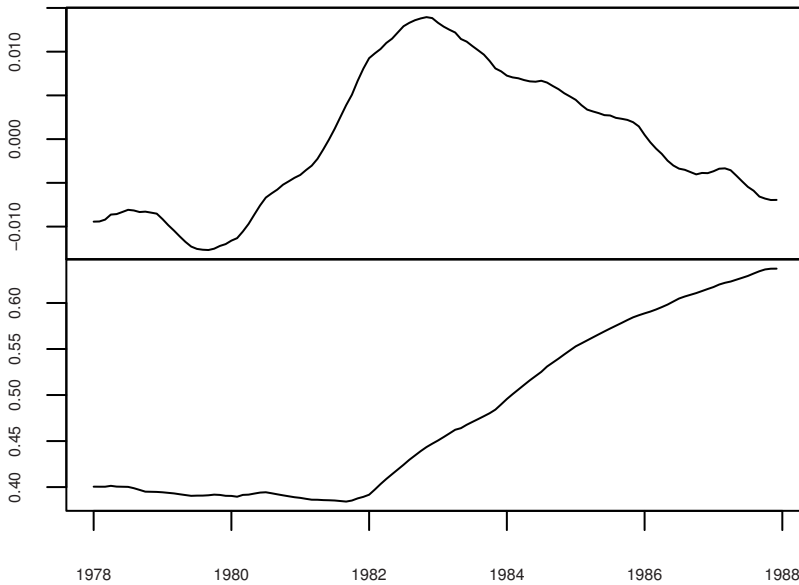


Fig. 3.14. Smoothing estimates of the coefficients of a dynamic CAPM model for the IBM stock returns

Further examples of dynamic regression with DLMs will be presented in the following sections, for more general, multivariate settings, such as longitudinal data, of the kind $(x_{i,t}, Y_{i,t}), t \geq 1, i = 1, \dots, m$, or more simply $(x_t, Y_{i,t})$ (e.g., dose $x_{i,t}$ of a drug and response $Y_{i,t}$ at time t for patient $i, i = 1, \dots, m$; or market returns x_t and excess returns $Y_{i,t}$ for asset i ; see Section 3.3.3); or time series of cross-sectional data, where x and Y are observed for m statistical units, as above, but usually m is large; see Section 3.3.5.

3.3 Models for multivariate time series

Modeling multivariate time series is of course more interesting—and more challenging—than studying univariate models, and also in this case DLMs offer a very flexible framework for the analysis. In this section we just present

a few examples of the extremely large variety of applications of DLMS to multivariate data, hoping that the reader can find tools and ideas for the analysis of her specific problems and models.

One can envision two basic types of data and problems in the analysis of multivariate time series. In many applications, one has data $Y_t = (Y_{1,t}, \dots, Y_{m,t})'$ on one or more variables observed for different units; for example, Y_t could be the gross domestic product observed for m countries over time, or the income and the expenses for a group of m families, or $Y_{i,t}$ could be the historical returns of stock i , $i = 1, \dots, m$, etc. In these cases, the focus of interest is typically in understanding the correlation structure among the time series, perhaps investigating the presence of clusters. These aspects might be of interest in themselves, or for improving the predictive performance of the model. In other contexts, the data are observations on one or more variables of interest Y and on some explanatory variables X_1, \dots, X_k . For example, Y could be the inflation rate and X_1, \dots, X_k be relevant macroeconomic variables for a country. We have again a multivariate time series $(Y_t, X_{1,t}, \dots, X_{k,t})$, but now the emphasis is on explaining or predicting the variable of interest Y_t by means of the explanatory variables $X_{j,t}$, so we are more in a regression framework. Note that in the regression DLM discussed in Section 3.2.7, the covariates were deterministic, while here $X_{1,t}, \dots, X_{k,t}$ are random variables. Of course by a joint model for $(Y_t, X_{1,t}, \dots, X_{k,t})$ one can also study feedback effects and causality relations among all variables.

3.3.1 DLMS for longitudinal data

Suppose that the value of a quantity y is observed for m statistical units over time, so we have a multivariate time series $(Y_t)_{t \geq 1}$, with $Y_t = (Y_{1,t}, \dots, Y_{m,t})'$. In fact, the simplest approach would be to study the m series independently, specifying a univariate model for each of them. This approach might give fairly good forecasts, but it doesn't do what is sometimes called "borrowing strength;" that is, in predicting Y for individual i , say, we do not exploit the information provided by the similar time series $(Y_{j,t})$ for $j \neq i$. In order to use all the available information, we want to introduce dependence across the time series, that is, we want a joint model for the m -variate process $((Y_{1,t}, \dots, Y_{m,t}) : t = 1, 2, \dots)$.

One way of introducing dependence across the time series $(Y_{1,t}), \dots, (Y_{m,t})$ is the following. Suppose that we can reasonably model each of the m time series using the same type of DLM, possibly with different variances but with the same time-invariant matrices G and F ; that is

$$\begin{aligned} Y_{i,t} &= F\theta_t^{(i)} + v_{i,t}, & v_{i,t} &\sim \mathcal{N}(0, V_i) \\ \theta_t^{(i)} &= G\theta_{t-1}^{(i)} + w_t^{(i)}, & w_{i,t} &\sim \mathcal{N}_p(0, W_i), \end{aligned} \quad (3.38)$$

$i = 1, \dots, m$. This corresponds to the qualitative assumption that all series follow the same type of dynamics. It also implies that the components of

the state vectors have similar interpretations across the different DLM, but they can assume different values for each time series ($Y_{i,t}$). Furthermore, as we shall discuss more extensively in Chapter 4, the variance matrices of the individual DLMs (3.38) might not be completely known, but be dependent on some unknown parameters, ψ_i say. For example, in a regression DLM of the kind discussed in Section 3.2.7, we would have

$$\begin{aligned} Y_{i,t} &= \alpha_{i,t} + \beta_{i,t}x_t + v_{i,t}, & v_{i,t} &\stackrel{iid}{\sim} N(0, \sigma_i^2) \\ \alpha_{i,t} &= \alpha_{i,t-1} + w_{1t,i}, & w_{1t,i} &\stackrel{iid}{\sim} N(0, \sigma_{w_{1,i}}^2) \\ \beta_{i,t} &= \beta_{i,t-1} + w_{2t,i}, & w_{2t,i} &\stackrel{iid}{\sim} N(0, \sigma_{w_{2,i}}^2) \end{aligned}$$

so that the model for $Y_{i,t}$ is characterized by its own state vector $\theta_t^{(i)} = (\alpha_{i,t}, \beta_{i,t})$ and by its parameters $\psi_i = (\sigma_i^2, \sigma_{w_{1,i}}^2, \sigma_{w_{2,i}}^2)$. In this framework we can assume that the time series $(Y_{1,t}), \dots, (Y_{m,t})$ are conditionally independent given the state processes $(\theta_t^{(1)}), \dots, (\theta_t^{(m)})$ and the parameters (ψ_1, \dots, ψ_m) , with $(Y_{i,t})$ depending only on its state $(\theta_t^{(i)})$ and parameters ψ_i ; in particular

$$(Y_{1,t}, \dots, Y_{m,t}) | \theta_t^{(1)}, \dots, \theta_t^{(m)}, \psi_1, \dots, \psi_m \sim \prod_{i=1}^m \mathcal{N}(y_{i,t}; F\theta_t^{(i)}, V_i(\psi_i)).$$

Note that this framework is similar to the one studied in Section 1.3. A dependence among $Y_{1,t}, \dots, Y_{m,t}$ can be introduced through the joint probability law of $(\theta_t^{(1)}, \dots, \theta_t^{(m)})$ and/or of the parameters (ψ_1, \dots, ψ_m) . If the states and the parameters are independent across individuals, then the time series $(Y_{1,t}), \dots, (Y_{m,t})$ are independent, and there is no borrowing strength; on the other hand, a dependence across the individual states and parameters implies dependence across the m time series. In the next sections we will provide some examples, where, for simplicity, the parameters, i.e., the matrices F, G, V_i, W_i , are known; their estimate (MLE and Bayesian) will be discussed in Chapter 4. A recent reference for mixtures of DLMs is Frühwirth-Schnatter and Kaufmann (2008). Also related are the articles by Caron et al. (2008) and Lau and So (2008).

3.3.2 Seemingly unrelated time series equations

Seemingly unrelated time series equations (SUTSE) are a class of models which specify the dependence structure among the state vectors $\theta_t^{(1)}, \dots, \theta_t^{(m)}$ as follows. As we said, the model (3.38) corresponds to the qualitative assumption that all series follow the same type of dynamics, and that the components of the state vectors have similar interpretations across the different DLMs. For example, each series might be modeled using a linear growth model, so that for each of them the state vector has a level and a slope component and, although

not strictly required, it is commonly assumed for simplicity that the variance matrix of the system errors is diagonal. This means that the evolution of level and slope is governed by independent random inputs. Clearly, the individual DLMs can be combined to give a DLM for the multivariate observations. A simple way of doing so is to assume that the evolution of the levels of the series is driven by correlated inputs, and the same for the slopes. In other words, at any fixed time, the components of the system error corresponding to the levels of the different series may be correlated, and the components of the system error corresponding to the different slopes may be correlated as well. To keep the model simple, we retain the assumption that levels and slopes evolve in an uncorrelated way. This suggests describing the joint evolution of the state vectors by grouping together all the levels and then all the slopes in an overall state vector $\theta_t = (\mu_{1,t}, \dots, \mu_{m,t}, \beta_{1,t}, \dots, \beta_{m,t})'$. The system error of the dynamics of this common state vector will then be characterized by a block-diagonal variance matrix having a first $m \times m$ block accounting for the correlation among levels and a second $m \times m$ block accounting for the correlation among slopes. To be specific, suppose one has $m = 2$ series. Then $\theta_t = (\mu_{1,t}, \mu_{2,t}, \beta_{1,t}, \beta_{2,t})'$ and the system equation is

$$\begin{bmatrix} \mu_{1,t} \\ \mu_{2,t} \\ \beta_{1,t} \\ \beta_{2,t} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \mu_{1,t-1} \\ \mu_{2,t-1} \\ \beta_{1,t-1} \\ \beta_{2,t-1} \end{bmatrix} + \begin{bmatrix} w_{1,t} \\ w_{2,t} \\ w_{3,t} \\ w_{4,t} \end{bmatrix}, \quad (3.39a)$$

where $(w_{1,t}, w_{2,t}, w_{3,t}, w_{4,t})' \sim \mathcal{N}(0, W)$ and

$$W = \begin{bmatrix} W_\mu & \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \\ \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} & W_\beta \end{bmatrix}. \quad (3.39b)$$

The observation equation for the bivariate time series $((Y_{1,t}, Y_{2,t})' : t = 1, 2, \dots)$ is

$$\begin{bmatrix} Y_{1,t} \\ Y_{2,t} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \theta_t + \begin{bmatrix} v_{1,t} \\ v_{2,t} \end{bmatrix}, \quad (3.39c)$$

with $(v_{1,t}, v_{2,t})' \sim \mathcal{N}(0, V)$. In order to introduce a further correlation between the series, the observation error variance V can be taken nondiagonal.

The previous example can be extended to the general case of m univariate time series. Let Y_t denote the multivariate observation at time t , and suppose that the i th component of Y_t follows a time invariant DLM as in (3.38), where $\theta_t^{(i)} = (\theta_{1,t}^{(i)}, \dots, \theta_{p,t}^{(i)})'$ for $i = 1, \dots, m$. Then a SUTSE model for (Y_t) has the form⁴

⁴ Given two matrices A and B , of dimensions $m \times n$ and $p \times q$ respectively, the Kronecker product $A \otimes B$ is the $mp \times nq$ matrix defined as

$$\begin{bmatrix} a_{1,1}B & \cdots & a_{1,n}B \\ \vdots & \vdots & \vdots \\ a_{m,1}B & \cdots & a_{m,n}B \end{bmatrix}.$$

$$\begin{cases} Y_t = (F \otimes I_m) \theta_t + v_t, & v_t \sim \mathcal{N}(0, V), \\ \theta_t = (G \otimes I_m) \theta_{t-1} + w_t, & w_t \sim \mathcal{N}(0, W), \end{cases} \quad (3.40)$$

with $\theta_t = (\theta_{1,t}^{(1)}, \theta_{1,t}^{(2)}, \dots, \theta_{p,t}^{(m-1)}, \theta_{p,t}^{(m)})'$. When the $w_t^{(i)}$ have diagonal variances, it is common to assume for W a block-diagonal structure with p blocks of size m . An immediate implication of the structure of the model is that forecasts made at time t of $\theta_{t+k}^{(i)}$ or $Y_{i,t+k}$ are based on the distribution of $\theta_t^{(i)}$ given all the observations $y_{1:t}$.

Example — Annual Denmark and Spain investments

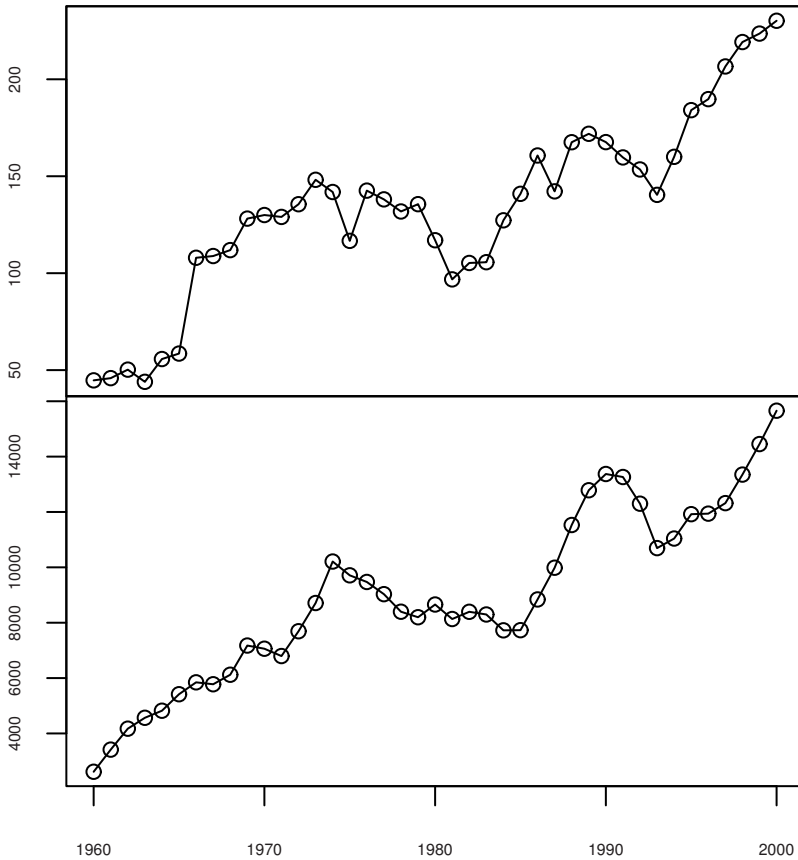


Fig. 3.15. Annual Denmark and Spain investments

Figure 3.15 shows the annual investment in Denmark and Spain from 1960 to 2000⁵. From visual inspection it appears that the two series display the same

⁵ Source: <http://www.fgn.unisg.ch/eumacro/macrodatta>.

type of qualitative behavior, that can be modeled by a linear growth DLM. This is the model we used on page 97 for the investments in Spain series alone. To set up a multivariate model for the two series one can combine the two linear growth models in a comprehensive SUTSE model. This turns out to be exactly of the form described by (3.39). There are six variances and three covariances in the model, for a total of nine parameters that need to be specified—or estimated from the data, as we will see in the next chapter. It is convenient to simplify slightly the model in order to reduce the overall number of parameters. So, for this example, we are going to assume that the two individual linear growth models are in fact integrated random walks. This means that, in (3.39b), $W_\mu = 0$. The MLE estimates of the remaining parameters are

$$W_\beta = \begin{bmatrix} 49 & 155 \\ 155 & 437266 \end{bmatrix}, \quad V = \begin{bmatrix} 72 & 1018 \\ 1018 & 14353 \end{bmatrix}.$$

The display below shows how to set up the model in R. In doing this we start by constructing a (univariate) linear growth model and then redefine the F and G matrices according to (3.40), using Kronecker products (lines 2 and 3). This approach is less subject to typing mistakes than manually entering the individual entries of F and G . The part of the code defining the variances V and W is straightforward.

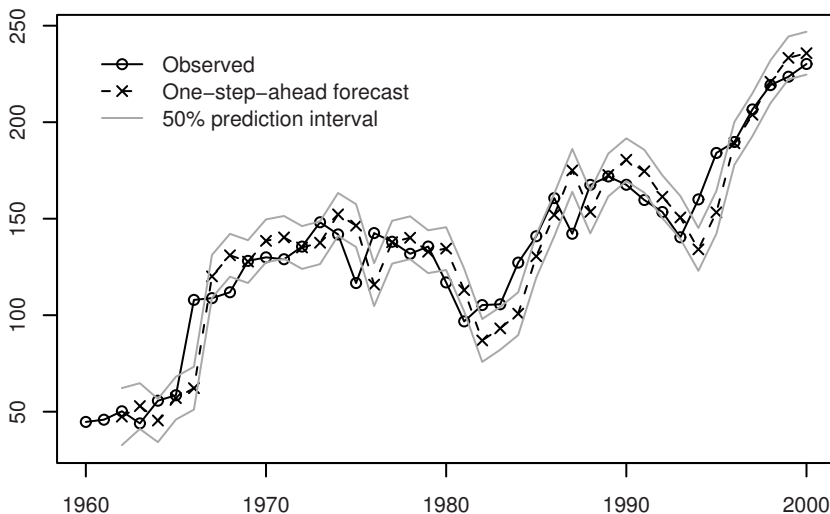


Fig. 3.16. Denmark investments and one step-ahead forecasts

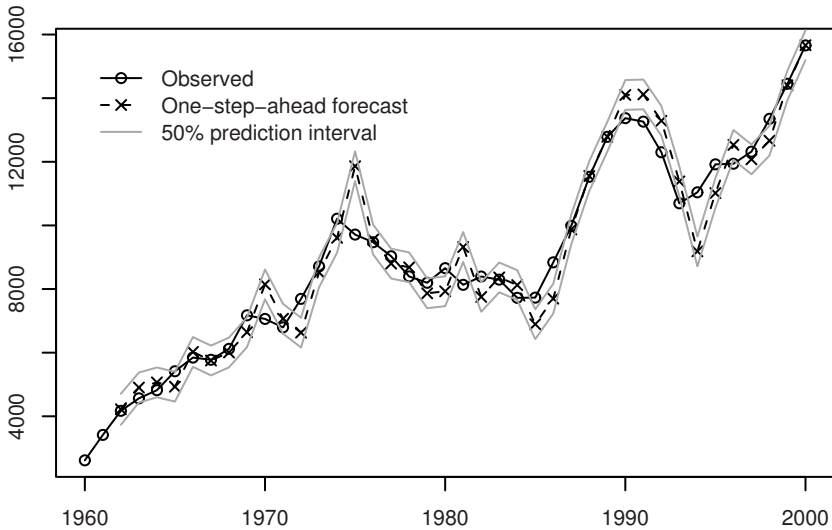


Fig. 3.17. Spain investments and one step-ahead forecasts

R code

```

> mod <- dlmModPoly(2)
2 > mod$FF <- mod$FF %x% diag(2)
> mod$GG <- mod$GG %x% diag(2)
4 > W1 <- matrix(0, 2, 2)
> W2 <- diag(c(49, 437266))
6 > W2[1, 2] <- W2[2, 1] <- 155
> mod$W <- bdiag(W1, W2)
8 > V <- diag(c(72, 14353))
> V[1, 2] <- V[2, 1] <- 1018
10 > mod$V <- V
> mod$m0 <- rep(0, 4)
12 > mod$C0 <- diag(4) * 1e7
> investFilt <- dlmFilter(invest, mod)
14 > sdev <- residuals(investFilt)$sd
> lwr <- investFilt$f + qnorm(0.25) * sdev
16 > upr <- investFilt$f - qnorm(0.25) * sdev

```

The code also illustrates how to compute probability intervals for the one-step-ahead forecasts, shown in Figures 3.16 and 3.17. Note that conditionally on $y_{1:t-1}$, Y_t and e_t have the same variance, see Section 2.8. This justifies the use of the innovation variances in lieu of the one-step-ahead observation forecast variances on line 14.

3.3.3 Seemingly unrelated regression models

As an example of how the idea expressed by SUTSE can be applied to more general DLMS than the basic structural model, we present below a multivariate dynamic regression model.

Let us consider a multivariate version of the dynamic CAPM illustrated on pages 122-125. Again, let $x_t = r_t^{(M)} - r_t^{(f)}$ be the excess market returns, but we have now a vector of excess returns for m assets, $y_{i,t} = r_{i,t} - r_t^{(f)}$, $i = 1, \dots, m$. For each asset, we can define a regression DLM

$$\begin{aligned} y_{i,t} &= \alpha_{i,t} + \beta_{i,t}x_t + v_{i,t} \\ \alpha_{i,t} &= \alpha_{i,t-1} + w_{1i,t} \\ \beta_{i,t} &= \beta_{i,t-1} + w_{2i,t}, \end{aligned}$$

and it is sensible to assume that the intercepts and the slopes are correlated across the m stocks. A *seemingly unrelated regression model* (SUR) is defined as

$$\begin{aligned} y_t &= (F_t \otimes I_m)\theta_t + v_t, & v_t &\stackrel{iid}{\sim} \mathcal{N}(0, V), \\ \theta_t &= (G \otimes I_m)\theta_{t-1} + w_t, & w_t &\stackrel{iid}{\sim} \mathcal{N}(0, W), \end{aligned}$$

with

$$y_t = \begin{bmatrix} y_{1,t} \\ \vdots \\ y_{m,t} \end{bmatrix}, \quad \theta_t = \begin{bmatrix} \alpha_{1,t} \\ \vdots \\ \alpha_{m,t} \\ \beta_{1,t} \\ \vdots \\ \beta_{m,t} \end{bmatrix}, \quad v_t = \begin{bmatrix} v_{1,t} \\ \vdots \\ v_{m,t} \end{bmatrix}, \quad w_t = \begin{bmatrix} w_{1,t} \\ \vdots \\ w_{2m,t} \end{bmatrix},$$

$F_t = [1 \ x_t]$, $G = I_2$, and $W = \text{blockdiag}(W_\alpha, W_\beta)$.

The data we analyze are the monthly returns for the Mobil, IBM, Weyer, and Citicorp stocks, 1978.1-1987.12, described on page 122. We assume for simplicity that the $\alpha_{i,t}$ are time-invariant, which amounts to assuming that $W_\alpha = 0$. The correlation between the different excess returns is explained in terms of the nondiagonal variance matrices V and W_β , estimated from the data:

$$\begin{aligned} V &= \begin{bmatrix} 41.06 & 0.01571 & -0.9504 & -2.328 \\ 0.01571 & 24.23 & 5.783 & 3.376 \\ -0.9504 & 5.783 & 39.2 & 8.145 \\ -2.328 & 3.376 & 8.145 & 39.29 \end{bmatrix}, \\ W_\beta &= \begin{bmatrix} 8.153 \cdot 10^{-7} & -3.172 \cdot 10^{-5} & -4.267 \cdot 10^{-5} & -6.649 \cdot 10^{-5} \\ -3.172 \cdot 10^{-5} & 0.001377 & 0.001852 & 0.002884 \\ -4.267 \cdot 10^{-5} & 0.001852 & 0.002498 & 0.003884 \\ -6.649 \cdot 10^{-5} & 0.002884 & 0.003884 & 0.006057 \end{bmatrix}. \end{aligned}$$

Smoothing estimates of the $\beta_{i,t}$'s, shown in Figure 3.18, can be obtained using the code below.

R code

```

> tmp <- ts(read.table("http://shazam.econ.ubc.ca/intro/P.txt",
2 +
+           header = TRUE),
+           start = c(1978, 1), frequency = 12) * 100
4 > y <- tmp[, 1 : 4] - tmp[, "RKFREE"]
> colnames(y) <- colnames(tmp)[1 : 4]
6 > market <- tmp[, "MARKET"] - tmp[, "RKFREE"]
> rm("tmp")
8 > m <- NCOL(y)
> ### Set up the model
10 > CAPM <- dlmModReg(market)
> CAPM$FF <- CAPM$FF %x% diag(m)
12 > CAPM$GG <- CAPM$GG %x% diag(m)
> CAPM$JFF <- CAPM$JFF %x% diag(m)
14 > CAPM$W <- CAPM$W %x% matrix(0, m, m)
> CAPM$W[-(1 : m), -(1 : m)] <-
16 +   c(8.153e-07, -3.172e-05, -4.267e-05, -6.649e-05,
+     -3.172e-05, 0.001377, 0.001852, 0.002884,
18 +     -4.267e-05, 0.001852, 0.002498, 0.003884,
+     -6.649e-05, 0.002884, 0.003884, 0.006057)
20 > CAPM$V <- CAPM$V %x% matrix(0, m, m)
> CAPM$V[] <- c(41.06, 0.01571, -0.9504, -2.328,
22 +   0.01571, 24.23, 5.783, 3.376,
+   -0.9504, 5.783, 39.2, 8.145,
24 +   -2.328, 3.376, 8.145, 39.29)
> CAPM$m0 <- rep(0, 2 * m)
26 > CAPM$C0 <- diag(1e7, nr = 2 * m)
> ### Smooth
28 > CAPMsmooth <- dlmSmooth(y, CAPM)
> ### Plot
30 > plot(dropFirst(CAPMsmooth$s[, m + 1 : m]),
+   lty = c("13", "6413", "431313", "B4"),
32 +   plot.type = "s", xlab = "", ylab = "Beta")
> abline(h = 1, col = "darkgrey")
34 > legend("bottomright", legend = colnames(y), bty = "n",
+   lty = c("13", "6413", "431313", "B4"), inset = 0.05)

```

Apparently, while Mobil's beta remained essentially constant during the period under consideration, starting around 1980 the remaining three stocks became less and less conservative, with Weyer and Citicorp reaching the status of aggressive investments around 1984. Note in Figure 3.18 how the estimated betas for the different stocks move in a clearly correlated fashion (with the

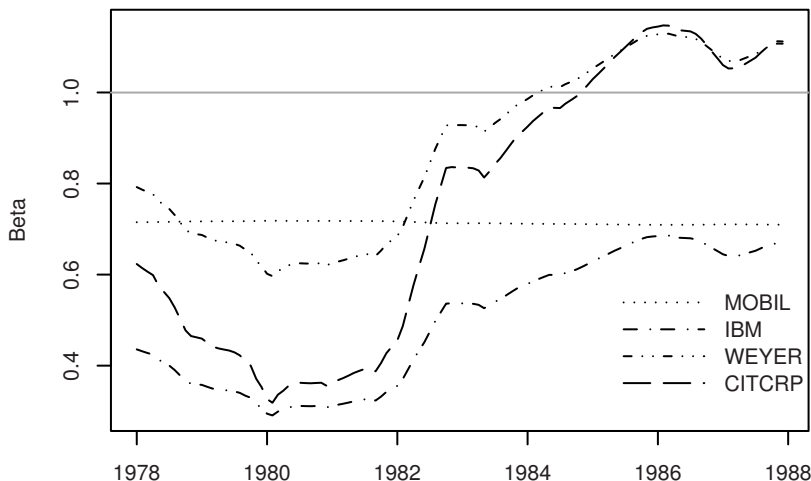


Fig. 3.18. Estimated betas for four stocks

exception of Mobil, that does not move at all)—a consequence of the positive covariances specified in the matrix W_β .

3.3.4 Hierarchical DLMs

Another general class of models for panel data and longitudinal studies is given by *dynamic hierarchical models* (Gamerman and Migon (1993) and references therein), which extend to dynamic systems the hierarchical linear models introduced by Lindley and Smith (1972).

A two-stage hierarchical DLM is specified as follows

$$\begin{aligned}
 Y_t &= F_{y,t}\theta_t + v_t, & v_t &\sim \mathcal{N}_m(0, V_{y,t}), \\
 \theta_t &= F_{\theta,t}\lambda_t + \epsilon_t, & \epsilon_t &\sim \mathcal{N}_P(0, V_{\theta,t}), \\
 \lambda_t &= G_t\lambda_{t-1} + w_t, & w_t &\sim \mathcal{N}_k(0, W_t),
 \end{aligned} \tag{3.41}$$

where the disturbance sequences (v_t) , (ϵ_t) , (w_t) are independent, and the matrices $F_{y,t}$ and $F_{\theta,t}$ are of full rank. Thus, in a two-stage DLM the state vector θ_t is itself modeled by a DLM. A key aspect is the progressive reduction in the dimension of the state parameters as the level becomes higher, that is $P > k$.

One application of hierarchical DLM is in modeling random effects in multivariate time series. Suppose that $Y_t = (Y_{1,t}, \dots, Y_{m,t})'$ are observations of a variable Y for m units at time t , and $Y_{i,t}$ is modeled as

$$Y_{i,t} = F_{1,t}\theta_{i,t} + v_{i,t}, \quad v_{i,t} \sim \mathcal{N}(0, \sigma_{i,t}^2) \tag{3.42a}$$

$i = 1, \dots, m$, with $v_{1,t}, \dots, v_{m,t}$ independent, for any given t . In the previous section, we illustrated SUTSE models for introducing a dependence across the individual time series ($Y_{i,t}$); however, these models require us to specify or estimate blocks of $m \times m$ matrices in the covariance matrix W , which might become complicated if m is large. In many applications it is in fact sufficient to model a simpler dependence, in particular to allow individual *random effects*. In its simpler form, this means assuming that

$$\theta_{i,t} = \lambda_t + \epsilon_{i,t}, \quad \epsilon_{i,t} \sim \mathcal{N}_p(0, \Sigma_t), \quad (3.42b)$$

$$\lambda_t = G\lambda_{t-1} + w_t, \quad w_t \sim \mathcal{N}_p(0, W_t), \quad (3.42c)$$

with $\epsilon_{1,t}, \dots, \epsilon_{m,t}$ independent. In other words, for any t the cross-sectional state vectors $\theta_{1,t}, \dots, \theta_{m,t}$ are a random sample from a $\mathcal{N}_p(\lambda_t, \Sigma_t)$ (i.e., they are conditionally i.i.d. given λ_t , with common distribution $\mathcal{N}_p(\lambda_t, \Sigma_t)$). Thus, we assume the same observation equation for the individual time series $Y_{i,t}$, modeling however heterogeneity across individuals by allowing random effects in the state processes.

Equations (3.42) can be expressed as a hierarchical DLM of the form (3.41), letting $\theta_t = (\theta'_{1,t}, \dots, \theta'_{m,t})'$, $v_t = (v_{1,t}, \dots, v_{m,t})'$, $\epsilon_t = (\epsilon'_{1,t}, \dots, \epsilon'_{m,t})'$, $V_{y,t} = \text{diag}(\sigma^2_{1,t}, \dots, \sigma^2_{m,t})$, $V_{\theta,t} = \text{diag}(\Sigma_t, \dots, \Sigma_t)$, $F_{y,t}$ block-diagonal with m blocks given by $F_{1,t}$, and

$$F_{\theta,t} = [I_p \mid \dots \mid I_p]'$$

A dimensionality reduction is obtained by passing from the mp -dimensional vectors θ_t to their p -dimensional common mean λ_t .

An example are dynamic regression models with random effects. Consider

$$Y_{i,t} = x'_{i,t}\theta_{i,t} + v_{i,t}.$$

Here, $Y_{i,t}$ are individual response variables, explained by the same regressors X_1, \dots, X_p with known value $x_{i,t} = (x_{1,it}, \dots, x_{p,it})'$ for unit i at time t . Again, random effects in the regression coefficients can be modeled by assuming that, for fixed t , the coefficients for the same regressor are exchangeable, more precisely

$$\theta_{1,t}, \dots, \theta_{m,t} | \lambda_t \stackrel{iid}{\sim} \mathcal{N}_p(\lambda_t, \Sigma_t).$$

A dynamics is then specified for (λ_t) , e.g., $\lambda_t = \lambda_{t-1} + w_t$, with $w_t \sim \mathcal{N}_p(0, W_t)$.

One way of obtaining filtering and smoothing estimates for hierarchical DLMs is to substitute the expression for θ_t in the observation equation (3.41), obtaining

$$\begin{aligned} Y_t &= F_{y,t}F_{\theta,t}\lambda_t + v_t^*, & v_t^* &\sim \mathcal{N}_m(0, F_{y,t}V_{\theta,t}F'_{y,t} + V_{y,t}) \\ \lambda_t &= G_t\lambda_{t-1} + w_t, & w_t &\sim \mathcal{N}(0, W_t). \end{aligned}$$

This has the form of a DLM and can be estimated by the usual procedures. In particular, for model (3.42), the observation equation reduces to

$$Y_t = F_{y,t} F_{\theta,t} \lambda_t + v_t^*,$$

$$v_t^* \sim \mathcal{N}(0, \text{diag}(F_{1,t} \Sigma_t F'_{1,t} + \sigma_y^2, \dots, F_{1,t} V_{\epsilon,t} F'_{1,t} + \sigma_y^2)).$$

Recursive formulae for filtering and prediction for hierarchical DLM are further discussed in Gamerman and Migon (1993). Landim and Gamerman (2000) present extensions to multivariate time series.

3.3.5 Dynamic regression

In many applications one wants to study the dependence of a variable Y on one or more explanatory variables x , over time. Suppose that, for each time t , we observe Y for different values of x , so that we have a time series of cross sectional data of the kind $((Y_{i,t}, x_i), i = 1, \dots, m, t \geq 1)$ (the values x_1, \dots, x_m are deterministic, and, for simplicity, we suppose that they are constant over time). For example, in financial applications, $Y_{i,t}$ might be the yield at time t of a zero coupon bond that gives one euro at time-to-maturity x_i . Data of this kind are plotted in Figure 3.19. Here the data are monthly yields from January 1985 to December 2000, with times-to-maturity of 3, 6, 9, 12, 15, 18, 21, 24, 30, 36, 48, 60, 72, 84, 96, 108, and 120 months ⁶.

Several issues arise for data of this nature. From the cross-sectional observations available at time t , we can estimate the regression function of Y_t on x , that is $m_t(x) = E(Y_t|x)$, for example with the aim of interpolating y at a new value x_0 . On the other hand, it is clearly of interest to study the temporal evolution of the m time series $(Y_{i,t} : t \geq 1)$. One might want to predict $Y_{i,t+1}$ based on the data $(Y_{i,1}, \dots, Y_{i,t})$, by a univariate time series model; or specify a multivariate model for the m -variate time series $(Y_{1,t}, \dots, Y_{m,t}), t \geq 1$. However, this approach does not fully exploit all the available information, since it ignores the dependence of the Y_i 's on the covariate x . In fact, estimating the dynamics of the regression curve is often the main objective of the analysis.

To consider both aspects of the problem, that is the cross-sectional and the temporal nature of the data, the proposal we illustrate in this section is a (semiparametric) dynamic regression model, written in the form of a DLM.

Suppose for simplicity that x is univariate. A flexible cross-sectional regression model is obtained by considering a basis functions expansion of the regression function at time t , of the form

$$m_t(x) = E(Y_t|x) = \sum_{j=1}^k \beta_{j,t} h_j(x) \quad (3.43)$$

where $h_j(x)$ are known basis functions (e.g., powers of x : $m_t(x) = \sum_{j=1}^{\infty} \beta_{j,t} x^j$, or trigonometric functions, or splines) and $\beta_t = (\beta_{1,t}, \dots, \beta_{k,t})'$ is the vector

⁶ Source: <http://www.ssc.upenn.edu/~fdiebold/papers/paper49/FBFITTED.txt>

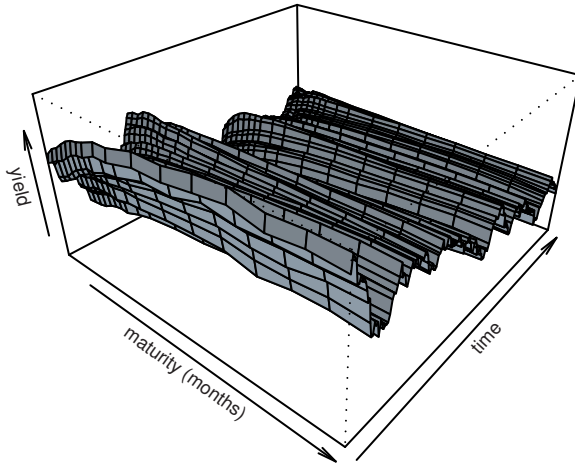


Fig. 3.19. Monthly yields, 1985.01-2000.12, for times-to-maturity 3, 6, 9, 12, 15, 18, 21, 24, 30, 36, 48, 60, 72, 84, 96, 108, and 120 months

of the expansion coefficients. The idea is that, for k large enough (in principle, $k \rightarrow \infty$), (3.43) can approximate any interesting shape of the regression function $m_t(x)$; for example, any continuous function on a closed interval can be approximated by polynomials. Models of the kind (3.43) are nevertheless simple since they are linear in the parameters $\beta_{j,t}$; at a given time t , we have a regression model

$$Y_{i,t} = \sum_{j=1}^k \beta_{j,t} h_j(x_i) + \epsilon_{i,t}, \quad i = 1, \dots, m,$$

with $\epsilon_{i,t} \stackrel{iid}{\sim} \mathcal{N}(0, \sigma^2)$; in matrix notation,

$$Y_t = F\beta_t + \epsilon_t, \quad \epsilon_t \sim \mathcal{N}(0, \sigma^2 I_m), \tag{3.44}$$

where

$$Y_t = \begin{bmatrix} Y_{1,t} \\ \vdots \\ Y_{m,t} \end{bmatrix}, \quad F = \begin{bmatrix} h_1(x_1) & \dots & h_k(x_1) \\ \vdots & & \vdots \\ h_1(x_m) & \dots & h_k(x_m) \end{bmatrix}, \quad \beta_t = \begin{bmatrix} \beta_{1,t} \\ \vdots \\ \beta_{k,t} \end{bmatrix}, \quad \epsilon_t = \begin{bmatrix} \epsilon_{1,t} \\ \vdots \\ \epsilon_{m,t} \end{bmatrix}.$$

Thus, β_t can be easily estimated, e.g., by least squares.

However, the regression curve evolves over time. Clearly, day-by-day cross-sectional estimates do not give a complete picture of the problem. We might have information on the dynamics of the curve that should be included in

the analysis. Modeling the dynamics of the curve $m_t(x)$ is not simple at first, since the curve is infinite-dimensional. However, having expressed $m_t(x)$ as in (3.43), its temporal evolution can be described by the dynamics of the finite-dimensional vector of coefficients $(\beta_{1,t}, \dots, \beta_{k,t})$. Suppose that the series (β_t) is Markovian, with

$$\beta_t = G\beta_{t-1} + w_t, \quad w_t \sim \mathcal{N}_k(0, W_t).$$

Then, we obtain a DLM with observation equation (3.44) and state equation as above. The state equation models the temporal evolution of the regression function. A simple specification is to assume that the $(\beta_{j,t})_{t \geq 1}$ are independent random walks or AR(1) processes; or a joint model might be used for the vector β_t . A word of caution is, however, of worth; the state equation introduces additional information, but it also constrains the dynamics of the curve. So, it is delicate: a poor specification of the dynamics may result in an unsatisfactory fit of the data.

3.3.6 Common factors

Sometimes it is conceptually useful to think of a number of observed series as driven by a small number of common factors. This is a common approach for example in economics, where one assumes that many observable series reflect the current state of the economy, which in turn can be expressed as a lower dimensional unobservable time series. For example, suppose that m observed series depend linearly on p ($p < m$) correlated random walks. The model can be written as

$$\begin{aligned} Y_t &= A\mu_t + v_t, & v_t &\sim \mathcal{N}(0, V), \\ \mu_t &= \mu_{t-1} + w_t, & w_t &\sim \mathcal{N}(0, W), \end{aligned} \tag{3.45}$$

where A is a fixed $m \times p$ matrix of factor loadings. The model can be seen as a dynamic generalization of factor analysis, where the common factors μ_t evolve with time. Note that (3.45) is nothing else than a DLM, with $\theta_t = \mu_t$ and $F_t = A$. One important difference with other DLMs that we have seen in this chapter is that here $p < m$, i.e., the state has a lower dimension than the observation. In addition, the system matrix A does not have any particular structure. As in standard factor analysis, in order to achieve identifiability of the unknown parameters, some constraints have to be imposed. In fact, if H is a $p \times p$ invertible matrix, defining $\tilde{\mu}_t = H\mu_t$ and $\tilde{A} = AH^{-1}$ and multiplying the second equation in (3.45) on the left by H , we obtain the equivalent model

$$\begin{aligned} Y_t &= \tilde{A}\tilde{\mu}_t + v_t & v_t &\sim \mathcal{N}(0, V), \\ \tilde{\mu}_t &= \tilde{\mu}_{t-1} + \tilde{w}_t & \tilde{w}_t &\sim \mathcal{N}(0, HWH'). \end{aligned}$$

Since A and W contain mp and $\frac{1}{2}p(p+1)$ parameters, respectively, but each combination of parameters belongs to a manifold of dimension p^2 (the number of elements of H) of equivalent models, the effective number of free parameters

(not including those in V) is $mp - \frac{1}{2}p(p-1)$. One way to parametrize the model and achieve identifiability is to set W equal to the identity matrix, and to impose that $A_{i,j}$, the (i, j) element of A , be zero for $j > i$. The fact that A is $m \times p$, with $p < m$, implies that A can be written in a partitioned form as

$$A = \begin{bmatrix} T \\ B \end{bmatrix},$$

with T being a $p \times p$ lower triangular matrix, and B an $(m-p) \times p$ rectangular matrix. This clearly shows that with this parametrization there are only $\frac{1}{2}p(p+1) + p(m-p) = mp - \frac{1}{2}p(p-1)$ parameters, which is exactly the number of free parameters of the unrestricted model. An alternative parametrization that achieves identifiability is obtained by assuming that W is a diagonal matrix, that $A_{i,i} = 1$ and $A_{i,j} = 0$ for $j > i$.

The model expressed by (3.45) is related to the notion of cointegrated series, introduced by Granger (1981) (see also Engle and Granger; 1987). The components of a vector time series x_t are said to be cointegrated of order d, b , written $x_t \sim \text{CI}(d, b)$, if (i) all the components of x_t are integrated of order d (i.e., $\Delta^d x_t^{(i)}$ is stationary for any i), and (ii) there exists a nonzero vector α such that $\alpha' x_t$ is integrated of order $d-b < d$. The components of Y_t in (3.45), as linear combinations of independent random walks (assuming for simplicity that the components of μ_0 are independent), are integrated of order 1. The columns of A are p vectors in \mathbb{R}^m , hence there are at least $m-p$ other linearly independent vectors in \mathbb{R}^m that are orthogonal to the columns of A . For any such α we have $\alpha' A = 0$ and, therefore, $\alpha' Y_t = \alpha' v_t$, i.e., $\alpha' Y_t$ is stationary—in fact, white noise. This shows that $Y_t \sim \text{CI}(1, 1)$. In a model where the common factors are stochastic linear trends instead of random walks, one can see that the observable series are $\text{CI}(2, 2)$.

Other DLM components that are commonly used as common factors include seasonal components and cycles, especially in economic applications. Further details on dynamic factor models can be found in Harvey (1989). For more recent developments, see also Forni et al. (2000).

3.3.7 Multivariate ARMA models

ARMA models for multivariate, m -dimensional, observations are formally defined as in the univariate case, through the recursive relation

$$Y_t = \sum_{j=1}^p \Phi_j Y_{t-j} + \epsilon_t + \sum_{j=1}^q \Psi_j \epsilon_{t-j}, \quad (3.46)$$

where (ϵ_t) is an m -variate Gaussian white noise sequence with variance Σ and the Φ_j and Ψ_j are $m \times m$ matrices. Here, without loss of generality, we have taken the mean of the process to be zero. In order for (3.46) to define a stationary process, all the roots of the complex polynomial

$$\det(I - \Phi_1 z - \dots - \Phi_p z^p) \quad (3.47)$$

must lie outside the unit disk. A DLM representation of a multivariate ARMA process can be formally obtained by a simple generalization of the representation given earlier (Section 3.2.5) for univariate ARMA processes. Namely, in the G matrix each ϕ_j needs to be replaced by a block containing the matrix Φ_j ; similarly for the ψ_j in the matrix R , that have to be replaced by Ψ_j blocks. Finally, all the occurrences of a “one” in F , G , and R must be replaced by the identity matrix of order m , and all the occurrences of a “zero” with a block of zeroes of order $m \times m$. For example, let us consider the bivariate ARMA(2,1) process

$$Y_t = \Phi_1 Y_{t-1} + \Phi_2 Y_{t-2} + \epsilon_t + \Psi_1 \epsilon_{t-1}, \quad \epsilon_t \sim \mathcal{N}(0, \Sigma), \quad (3.48)$$

with

$$\Psi_1 = \begin{bmatrix} \Psi_{11} & \Psi_{12} \\ \Psi_{21} & \Psi_{22} \end{bmatrix}, \quad \Phi_i = \begin{bmatrix} \Phi_{11,i} & \Phi_{12,i} \\ \Phi_{21,i} & \Phi_{22,i} \end{bmatrix}, \quad i = 1, 2. \quad (3.49)$$

Then the system and observation matrices needed to define the DLM representation of (3.48) are the following:

$$\begin{aligned} F &= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}, \\ G &= \begin{bmatrix} \Phi_{11,1} & \Phi_{12,1} & 1 & 0 \\ \Phi_{21,1} & \Phi_{22,1} & 0 & 1 \\ \Phi_{11,2} & \Phi_{12,2} & 0 & 0 \\ \Phi_{21,2} & \Phi_{22,2} & 0 & 0 \end{bmatrix}, \\ R &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ \Psi_{11} & \Psi_{12} \\ \Psi_{21} & \Psi_{22} \end{bmatrix}, \quad W = R \Sigma R'. \end{aligned} \quad (3.50)$$

In R, the function `d1mModARMA` can be used to create a DLM representation of an ARMA model also in the multivariate case. The contributed package `dse1` also provides tools for the analysis of multivariate ARMA models.

For a detailed treatment of multivariate ARMA models the reader can consult Reinsel (1997) and Lütkepohl (2005).

In many applications, in particular in econometrics, ARMA models with no moving average part are often considered, as they are more easily interpretable. Multivariate AR models are commonly called *vector autoregressive* (VAR) models. In econometrics, VAR models are widely used both for forecasting and for detecting relationships between macroeconomic variables or groups of variables. Suppose we partition Y_t into a group of k variables and a group of $m - k$ variables, X_t and Z_t say, so that $Y_t' = (X_t', Z_t')$. It is often of

interest to study causal relationships between (X_t) and (Z_t) . There are several ways of rigorously defining this problem. One possibility is to appeal to the notion of *Granger causality*, which is based on assessing the improvement in the forecasting of one variable produced by the inclusion of the other in the information set upon which the forecast is based. More precisely, we say that (X_t) is Granger-causal for (Z_t) if for all t

$$\text{Var}(\mathbb{E}(Z_{t+h}|z_{1:t})) > \text{Var}(\mathbb{E}(Z_{t+h}|z_{1:t}, x_{1:t}))$$

for some $h \geq 1$.

Instantaneous causality is another notion of causal relationship between two groups of variables that is often used in macroeconomic analysis. We say that there is instantaneous causality between (X_t) and (Z_t) if for all t

$$\text{Var}(\mathbb{E}(Z_{t+1}|y_{1:t})) > \text{Var}(\mathbb{E}(Z_{t+1}|y_{1:t}, x_{t+1})).$$

Yet another way of studying relationships among variables is the approach based on impulse response analysis. Loosely speaking, and without entering in the details, in this type of analysis one tries to understand how one variable responds to a shock in another variable in a system that includes other variables as well.

It turns out that also the class of VAR models is too large and the interpretation of a particular model may not be easy. In particular, impulse responses are generally not unique. To overcome this issue, structural restrictions can be imposed on VAR models, leading to the so-called *structural VARs*; see, e.g., Amisano and Giannini (1997). VARs and structural VARs provide also an appropriate setting to apply the notion of cointegration (see p. 139), which has received a lot of attention from econometricians in the last twenty years.

We will not treat VAR models or cointegration in this book; the interested reader can consult Lütkepohl (2005), Canova (2007), or Pfaff (2008a), the latter showing also implementations in R. The contributed package `vars` (Pfaff; 2008b) can be used for the analysis in R of VAR models.

To conclude this section, let us point out that macroeconomic analysis is an area where expert opinion, in the form of informative prior distributions for unknown model parameters, is often available and can be fruitfully incorporated into a model to improve forecasts and inference in general. However, specifying in a meaningful way a prior for the many parameters of a VAR model may not be an easy task, and several simplified ways of doing so have been suggested, starting with the so-called “Minnesota prior” (Litterman; 1986; Doan et al.; 1984). There is now a vast literature on Bayesian VAR’s and we refer the reader to Canova (2007) and references therein. In R, the contributed package `MSBVAR` (Brandt; 2008) can be used for the analysis of Bayesian VAR models.

Problems

- 3.1.** Simulate patterns of the random walk-plus-noise model for different values of V and W .
- 3.2.** Show that, for the local level model, $\lim_{t \rightarrow \infty} C_t = KV$, where K is defined by (3.8).
- 3.3.** Let $(Y_t, t = 1, 2, \dots)$ be described by a random walk plus noise model. Show that the first differences $Z_t = Y_t - Y_{t-1}$ are stationary and have the same autocorrelation function of a MA(1) model.
- 3.4.** Simulate patterns of the linear growth model for different values of V and W_1, W_2 .
- 3.5.** Let $(Y_t, t = 1, 2, \dots)$ be described by a linear growth model. Show that the second differences $Z_t = Y_t - 2Y_{t-1} + Y_{t-2}$ are stationary and have the same autocorrelation function of a MA(2) model.
- 3.6.** Show that the forecast function for the polynomial model of order n is polynomial of order $n - 1$.
- 3.7.** Verify that the second differences of (Y_t) for a linear growth model can be written in terms of the innovations as in (3.12).
- 3.8.** Consider the data studied in Section 3.2.6. Estimate a global trend model, $y_t = \mu_0 + \delta t + \epsilon_t$. Look at the one-step-ahead forecast residuals, and make your comments. Then, compare with the integrated random walk and the linear growth model for these data.
- 3.9.** Show that a Fourier form seasonal DLM has a periodic forecast function, even when W is a general variance matrix.
- 3.10.** Prove that (3.31) holds.
- 3.11.** Simulate paths of a stationary bivariate VAR(1) process. Compare with paths of a nonstationary VAR(1) process.