Giovanni Petris, Sonia Petrone, Patrizia Campagnoli

Dynamic Linear Models with R

SPIN Springer's internal project number, if known

– Monograph –

August 10, 2007

Springer
Berlin Heidelberg New York
Hong Kong London
Milan Paris Tokyo

Contents

1

Introduction: basic notions about Bayesian inference

In the last decades, dynamic linear models, and more generally state-space models, have become a focus of interest in time series analysis. Part of the reason is due to the possibility of solving computational difficulties using modern Monte Carlo methods, in a Bayesian approach. This book introduces to Bayesian modeling and forecasting of time series using dynamic linear models, presenting the basic concepts and techniques, and providing an R-package for their practical implementation.

Before getting started, this chapter briefly reviews some basic notions of Bayesian statistics. Reference books on Bayesian statistics are Bernardo and Smith (1994), DeGroot (1970), Berger (1985), O'Hagan (1994), Robert (2001), Cifarelli and Muliere (1989), or Zellner (1971), Poirier (1995) and Geweke (2005) for a more econometric viewpoint.

1.1 Introduction

In the analysis of real data, in economics, sociology, biology, engineering and in any field, we rarely have perfect information on the phenomenon of interest. Even when an accurate deterministic model describing the system under study is available, there is always something that is not under our control, effects of forgotten variables, measurement errors, imperfections. We always have to deal with some uncertainty. A basic point in Bayesian statistics is that all the uncertainty that we might have on a phenomenon should be described by means of *probability*. In this viewpoint, probability has a *subjective* interpretation, being a way of formalizing the incomplete information that the researcher has about the events of interest. Probability theory prescribes how to assign probabilities coherently, avoiding contradictions and undesirable consequences.

The Bayesian approach to the problem of "learning from experience" on a phenomenon moves from this crucial role recognized to probability. The learning process is solved through the application of probability rules: one

simply has to compute the *conditional probability* of the event of interest, given the experimental information. Bayes theorem is the basic rule to be applied to this aim. Given two events A and B, probability rules say that the joint probability of A and B occurring is given by $P(A \cap B) = P(A|B)P(B) = P(B|A)P(A)$, where $P(A|B)$ is the conditional probability of A given B and $P(B)$ is the (marginal) probability of B. Bayes theorem, or the theorem of inverse probability, is a simple consequence of the above equalities and says that

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}.$$

This is an elementary result, that goes back to Thomas Bayes (who died in 1761). The importance of this theorem in Bayesian statistics is in the interpretation and scope of the inputs of the two sides of the equation, and in the role that consequently Bayes theorem assumes for formalizing the inductive learning process. In Bayesian statistics, A represents the event of interest for the researcher and B an experimental result which she believes can provide information about A. Given $P(A)$ and consequently $P(\bar{A}) = 1 - P(A)$, and having assigned the conditional probabilities $P(B|A)$ and $P(B|\bar{A})$ of the experimental fact B conditionally on A or $\bar{A}$, the problem of learning about A from the "experience" B is solved by computing the conditional probability $P(A|B)$.

The event of interest and the experimental result depend on the problem. In statistical inference, the experimental fact is typically the result of a sampling procedure, and it is described by a random vector Y; usually, we use a parametric model for assigning the probability law of Y, and the quantity of interest is the vector θ of the parameters of the model. Bayesian inference on θ is solved by computing its conditional distribution given the sampling results. More specifically, suppose that, based on his knowledge of the problem, the researcher can assign a density[1] $f(y \mid \theta)$ of Y given θ (*likelihood*), and a *prior density* $\pi(\theta)$ expressing his uncertainty on the parameters θ. In this case we can use a generalization of the elementary Bayes theorem, known as Bayes formula, for computing the conditional density of θ given y:

$$\pi(\theta \mid y) = \frac{f(y \mid \theta)\pi(\theta)}{m(y)},$$

[1] In general, we use the term density in its measure-theoretic sense, with respect to some dominating measure. The reader not accustomed to measure theory can think of a density function in the continuous case, or a probability mass function in the discrete case. In general, to apply Bayes formula we have to assume that the model is dominated, that is, there exists a conditional density $f(y|\theta)$ w.r.t. one dominating measure (the same for any value of θ)

where $m(y)$ is the marginal density of Y[2] Thus, Bayesian inference is solved by computing the relevant conditional distributions, and Bayes formula is a basic tool to this aim. It has an elegant, appealing coherence and simplicity. Differently from Bayesian procedures, frequentist statistical inference does not have a probability distribution for the unknown parameters, and inference on θ is based on the determination of estimators with good properties, confidence intervals, hypothesis testing. The reason is that, since the value of the parameter θ does not "vary", θ is not interpretable as a random "variable" in a frequentist sense, neither the probability that θ takes values in a certain interval can have a frequentist interpretation. Adopting subjective probability, instead, θ is a random quantity simply because its value is uncertain to the researcher, who should formalize the information she has about it by means of probability. This seems indeed quite natural. You might have experienced that expressions such as "the probability that θ has values in an interval (a, b) is 0.9" come generally more naturally in your mind than the notion of confidence level of a frequentist confidence interval; however, they are justified only if θ is a random variable, with a subjective probability law. We have to refer the reader to the literature for a much deeper discussion; to the fundamental work of Bruno de Finetti (de Finetti; 1970a,b, see) or Savage.... Lindley.....

In many applications, the main objective of a statistical analysis is *forecasting*; thus, the event of interest is the value of a future observation Y^*. Again, prediction of a future value Y^* given the data y is simply solved in principle in the Bayesian approach, by computing the conditional density of $Y^* \mid y$, which is called *predictive density*. In parametric models it can be computed as

$$f(y^* \mid y) = \int f(y^*, \theta \mid y) \mathrm{d}\nu(\theta) = \int f(y^*|y, \theta)\pi(\theta|y) \mathrm{d}\nu(\theta).$$

The last expression involves again the posterior distribution of θ. In fact, apart from controversies about frequentist or subjective probability, a difficulty with (prior or posterior) probability distributions on model parameters is that, in some problems, they do not have a clear physical interpretation, so that assigning to them a probability law is debatable, even from a subjective viewpoint. According to de Finetti, one can give probability only to "observable facts"; indeed, the ultimate goal of a statistical analysis is often forecasting the future observations rather than learning on unobservable parameters. Taking a *predictive approach*, the parametric model is to be regarded just as a tool for facilitating the task of specifying the probability law of the observable quantities and, eventually, of the predictive distribution. The choice of the prior should be suggested, in this approach, by predictive considerations, that is taking into account its implications on the probability law of Y. We discuss this point further in the next section.

[2] If θ is continuous, $m(y) = \int f(y|\theta)\pi(\theta)\mathrm{d}\theta$; if θ is discrete, $m(y) = \sum_{\theta_j} f(y \mid \theta_j)\pi(\theta_j))$. The measure-theoretic notation $\int f(y|\theta)\pi(\theta)\mathrm{d}\nu(\theta)$ covers both cases and we will use it throughout the book.

1.1.1 Simple dependence structures

Forecasting is one of the main tasks in time series analysis. A multivariate time series is described probabilistically by a stochastic process $(Y_t; t = 1, 2, \ldots)$, that is, by an ordered sequence of random vectors with the index t denoting time. For simplicity, we will think of equally spaced time points (daily data, monthly data, and so on); for example, (Y_t) might describe the daily prices of m bonds, or monthly observations on the sales of a good, etcetera. One basic problem is to make forecasts about the value of the next observation, Y_{n+1} say, having observed data up to time n, $(Y_1 = y_1, \ldots, Y_n = y_n)$. Clearly, the first step to this aim is to formulate reasonable assumptions about the dependence structure of the process $(Y_t; t = 1, 2, \ldots)$. If we are able to specify the probability law of the process (Y_t), we know the joint densities $f(y_1, \ldots, y_n)$ for any $n \geq 1$, and Bayesian forecasting would be solved by computing the predictive density

$$f(y_{n+1}|y_1, \ldots, y_n) = \frac{f(y_1, \ldots, y_{n+1})}{f(y_1, \ldots, y_n)}.$$

In practice, specifying the densities $f(y_1, \ldots, y_n)$ directly is not easy, and one finds convenient to make use of parametric models; that is, one usually finds simpler to express the probability law of $(Y_1, \ldots, Y_n)$ conditionally on some characteristics θ of the system that generates the data. The relevant characteristics θ can be finite or infinite-dimensional, that is, θ can be a random vector or, as we shall see in the case of state space models, a stochastic process itself. The researcher often finds simpler to specify the conditional density $f(y_1, \ldots, y_n|\theta)$ of $(Y_1, \ldots, Y_n)$ given θ, and a density $\pi(\theta)$ on θ, then obtaining $f(y_1, \ldots, y_n)$ as $f(y_1, \ldots, y_n) = \int f(y_1, \ldots, y_n \mid \theta)\pi(\theta)d\theta$. As we shall see, we will proceed in this fashion when introducing dynamic linear models for time series analysis. But let's first study simpler dependence structures.

Conditional independence

The simplest dependence structure is conditional independence. In particular, in many applications it is reasonable to assume that $Y_1, \ldots, Y_n$ are conditionally independent and identically distributed (i.i.d.) given θ: $f(y_1, \ldots, y_n|\theta) = \prod_{i=1}^{n} f(y_i|\theta)$. For example, if the Y_t's are repeated measurements affected by a random error, we are used to think of a model of the kind $Y_t = \theta + \epsilon_t$, where the ϵ_t's are independent Gaussian random errors, with mean zero and variance σ^2 depending on the precision of the measurement device. This means that, conditionally on θ, the Y_t's are i.i.d., with $Y_t|\theta \sim \mathcal{N}(\theta, \sigma^2)$.

Note that $Y_1, Y_2, \ldots$ are only conditionally independent: the observations $y_1, \ldots, y_n$ provide us information about the unknown value of θ and, through θ, on the value of the next observation Y_{n+1}. Thus, Y_{n+1} depends, in a probabilistic sense, on the past observations $Y_1, \ldots, Y_n$. The predictive density in this case can be computed as

$$f(y_{n+1}|y_1,\ldots,y_n) = \int f(y_{n+1},\theta|y_1,\ldots,y_n)\mathrm{d}\nu(\theta) \tag{1.1}$$

$$= \int f(y_{n+1}|\theta,y_1,\ldots,y_n)\pi(\theta|y_1,\ldots,y_n)\mathrm{d}\nu(\theta)$$

$$= \int f(y_{n+1}|\theta)\pi(\theta|y_1,\ldots,y_n)\mathrm{d}\nu(\theta), \tag{1.2}$$

the last equality following from the assumption of conditional independence, where $\pi(\theta|y_1,\ldots,y_n)$ is the posterior density of θ, conditionally on the data $(y_1,\ldots,y_n)$. As we have seen, the posterior density can be computed by Bayes formula:

$$\pi(\theta|y_1,\ldots,y_n) = \frac{f(y_1,\ldots,y_n|\theta)\pi(\theta)}{m(y_1,\ldots,y_n)} \propto \prod_{t=1}^{n} f(y_t|\theta)\ \pi(\theta)\ . \tag{1.3}$$

Note that the marginal density $m(y_1,\ldots,y_n)$ does not depend on θ, having the role of normalizing constant, so that the posterior is proportional to the product of the likelihood and the prior (the symbol $\propto$ means "proportional to").

It is interesting to note that, with the assumption of conditional independence, the posterior distribution can be computed *recursively*. This means that one does not need all the previous data to be kept in storage and reprocessed every time a new measurement is taken. In fact, at time $(n-1)$, the information available about θ is described by the conditional density

$$\pi(\theta|y_1,\ldots,y_{n-1}) \propto \prod_{t=1}^{n-1} f(y_t|\theta)\pi(\theta),$$

so that this density plays the role of prior at time n. Once the new observation y_n becomes available, we have just to compute the likelihood, which is $f(y_n|\theta,y_1,\ldots,y_{n-1}) = f(y_n|\theta)$ by the assumption of conditional independence, and update the "prior" $\pi(\theta|y_1,\ldots,y_{n-1})$ by Bayes rule, obtaining

$$\pi(\theta|y_1,\ldots,y_{n-1},y_n) \propto \pi(\theta|y_1,\ldots,y_{n-1})f(y_n|\theta) \propto \prod_{t=1}^{n-1} f(y_t|\theta)\pi(\theta)f(y_n|\theta),$$

which is (1.3). The recursive structure of the posterior will be a crucial point when we will study dynamic linear models and Kalman filter in the next chapters.

Example. To fix ideas, let's use a simple example. Suppose that, after a wreck in the ocean, you landed on a small island, and let θ denote your position, the distance from the coast say. When studying dynamic linear models, we will consider the case when θ is subject to change over time (you are on a life boat in the ocean and not on an island! so that you slowly move with the stream

and the waves, being at distance θ_t from the coast at time t). However, for the moment let's consider θ as fixed. Luckily, you can see the coast at times; you have some initial idea of your position θ, but you are clearly interested in learning more about θ based on the measurements y_t that you can take. Let's formalize the learning process in the Bayesian approach.

The measurements Y_t can be modeled as

$$Y_t = \theta + \epsilon_t \ , \quad \epsilon_t \ i.i.d \ \sim \mathcal{N}(0, \sigma^2),$$

where the ϵ_t's and θ are independent and, for simplicity, σ^2 is a known constant. In other words:

$$Y_1, Y_2, \ldots, |\theta \ i.i.d \ \sim \mathcal{N}(\theta, \sigma^2).$$

Suppose you agree to express your prior idea about θ as

$$\theta \sim \mathcal{N}(m_0, C_0),$$

where the prior variance C_0 might be very large if you are very uncertain about your guess m_0. Given the measurements $(y_1, \ldots, y_n)$, you will update your opinion about θ computing the posterior density of θ, using the Bayes formula. We have

$$\begin{aligned}
\pi(\theta|y_1, \ldots, y_n) &\propto \text{ likelihood} \times \text{ prior} \\
&= \prod_{t=1}^{n} \frac{1}{\sqrt{2\pi}\sigma} \exp\{-\frac{1}{2\sigma^2}(y_t - \theta)^2\} \ \frac{1}{\sqrt{2\pi C_0}} \exp\{-\frac{1}{2C_0}(\theta - m_0)^2\} \\
&\propto \exp\{-\frac{1}{2\sigma^2}(\sum_{t=1}^{n} y_t^2 - 2\theta \sum_{t=1}^{n} y_t + n\theta^2) - \frac{1}{2C_0}(\theta^2 - 2\theta m_0 + m_0^2)\} \\
&\propto \exp\{-\frac{1}{2\sigma^2 C_0}((nC_0 + \sigma^2)\theta^2 - 2(nC_0\bar{y} + \sigma^2 m_0)\theta)\}
\end{aligned}$$

The above expression might appear complicated, but in fact it is the kernel of a Normal density. Note that, if $\theta \sim \mathcal{N}(m, C)$, then $\pi(\theta) \propto \exp\{-(1/2C)(\theta^2 - 2m\theta)\}$; so, writing the above expression as

$$\exp\{\frac{1}{2\sigma^2 C_0/(nC_0 + \sigma^2)}(\theta^2 - 2\frac{nC_0\bar{y} + \sigma^2 m_0}{(nC_0 + \sigma^2)}\theta)\},$$

we recognize that

$$\theta|y_1, \ldots, y_n \sim \mathcal{N}(m_n, C_n),$$

where

$$m_n = \mathrm{E}(\theta|y_1, \ldots, y_n) = \frac{C_0}{C_0 + \sigma^2/n}\bar{y} + \frac{\sigma^2/n}{C_0 + \sigma^2/n} m_0 \tag{1.4}$$

and

$$C_n = \mathrm{Var}(\theta|y_1, \ldots, y_n) = (\frac{n}{\sigma^2} + \frac{1}{C_0})^{-1} = \frac{\sigma^2 C_0}{\sigma^2 + nC_0}. \tag{1.5}$$

The posterior *precision* is $1/C_n = n/\sigma^2 + 1/C_0$, and it is the sum of the precision n/σ^2 of the sample mean and the initial precision $1/C_0$. The posterior precision is always bigger than the initial precision: even data of poor quality provide some information. The posterior expectation $m_n = \mathrm{E}(\theta|y_1, \ldots, y_n)$ is a weighted average between the sample mean $\bar{y} = \sum_{i=1}^{n} y_i/n$ and the prior guess $m_0 = \mathrm{E}(\theta)$, with weights depending on C_0 and σ^2. If the prior uncertainty, represented by C_0, is small w.r.t. σ^2, the prior guess receives more weight. If C_0 is very large, then $m_n \simeq \bar{y}$ and $C_n \simeq \sigma^2/n$. Figure .. shows the prior-to-posterior updating.

As we have seen, the posterior distribution can be computed recursively. At time n, the conditional density $\mathcal{N}(m_{n-1}, C_{n-1})$ of θ given the previous data $y_1, \ldots, y_{n-1}$ plays the role of prior; and the likelihood for the current observation is

$$f(y_n \mid \theta, y_1, \ldots, y_{n-1}) = f(y_n \mid \theta) = \mathcal{N}(y_n; \theta, \sigma^2).$$

We can update the prior $\mathcal{N}(m_{n-1}, C_{n-1})$ on the base of the observation y_n using formulas (1.4) and (1.5), with m_{n-1} and C_{n-1} in place of m_0 and C_0. We see that the resulting posterior density is Gaussian, with parameters

$$m_n = \frac{C_{n-1}}{C_{n-1}+\sigma^2}\, y_n + (1 - \frac{C_{n-1}}{C_{n-1}+\sigma^2})\, m_{n-1} = m_{n-1} + \frac{C_{n-1}}{C_{n-1}+\sigma^2}(y_n - m_{n-1}) \tag{1.6}$$

and variance

$$C_n = (\frac{1}{\sigma^2} + \frac{1}{C_{n-1}})^{-1} = \frac{\sigma^2 C_{n-1}}{\sigma^2 + C_{n-1}}. \tag{1.7}$$

Being $Y_{n+1} = \theta + \epsilon_{n+1}$, the *predictive density* of $Y_{n+1}|y_1, \ldots, y_n$ is Normal, with mean m_n and variance $C_n + \sigma^2$; thus, m_n is the posterior expected value of θ and also the one step-ahead "point prediction" $E(Y_{n+1}|y_1, \ldots, y_n)$. Expression (1.6) shows that m_n is obtained by correcting the previous estimate m_{n-1} by a term which takes into account the forecast error $e_n = (y_n - m_{n-1})$, weighted by

$$\frac{C_{n-1}}{C_{n-1}+\sigma^2} = \frac{C_0}{\sigma^2 + nC_0} \tag{1.8}$$

(being, from (1.5), $C_{n-1} = \frac{\sigma^2 C_0}{\sigma^2 + (n-1)C_0}$). As we shall see in chapter 2, this "prediction-error correction" structure is proper, more generally, of the formulas of the Kalman filter for dynamic linear models.

Exchangeability

Exchangeability is the basic dependence structure in Bayesian analysis. Let $(Y_t; t = 1, 2, \ldots)$ be an infinite sequence of random vectors. Suppose that the order in the sequence is not relevant, in the sense that, for any $n \geq 1$, the vector $(Y_1, \ldots, Y_n)$ and any of its permutations, $(Y_{i_1}, \ldots, Y_{i_n})$, have the same

probability law. In this case, we say that the sequence $(Y_t; t = 1, 2, \ldots)$ is *exchangeable*. This is a reasonable assumption when the Y_t's represent the results of experiments repeated under similar conditions. In the example of the previous paragraph, it is quite natural to consider that the order in which the measurements Y_t of the distance from the coast are taken is not relevant. There is an important result, known as de Finetti representation theorem, that shows that the assumption of exchangeability is equivalent to the assumption of conditional independence and identical distribution that we have discussed in the previous paragraph. There is however an important difference. As you can see, here we move from a quite natural assumption on the dependence structure of the observables, that is exchangeability; we have not introduced, up to now, parametric models or prior distributions on parameters. In fact, the hypothetical model, that is the pair likelihood and prior, arises from the assumption of exchangeability, as shown by the representation theorem.

Theorem 1.1. *(***de Finetti representation theorem***). Let* $(Y_t; t = 1, 2, \ldots)$ *be an infinite sequence of exchangeable random vectors. Then*

(a) With probability one, the sequence of empirical d.f.'s

$$F_n(y) = F_n(y; Y_1, \ldots, Y_n) = \frac{1}{n} \sum_{i=1}^{n} I_{(-\infty, y]}(Y_i)$$

converges weakly to a random d.f. F*, as* $n \to \infty$*;*

(b) for any $n \geq 1$*, the d.f. of* $(Y_1, \ldots, Y_n)$ *can be represented as*

$$P(Y_1 \leq y_1, \ldots, Y_n \leq y_n) = \int \prod_{i=1}^{n} F(y_i) d\pi(F)$$

where π *is the probability law of the weak limit* F *of the sequence of the empirical d.f.'s.*

The fascinating aspect of the representation theorem is that the hypothetical model results from the assumptions on the dependence structure of the observable variables $(Y_t; t = 1, 2, \ldots)$. If we assume that the sequence $(Y_t; t = 1, 2, \ldots)$ is exchangeable, then the observations are i.i.d. conditionally on the d.f. F, with common d.f. F. The random d.f. F is the weak limit of the empirical d.f.'s. The prior distribution π (also called, in this context, de Finetti measure) is a probability law on the space $\mathcal{F}$ of all the d.f.'s on the sample space $\mathcal{Y}$ and expresses our beliefs on the limit of the empirical d.f.'s. In many problems we can restrict the support of the prior to a parametric class $\mathcal{P}_\Theta = \{F(\cdot|\theta), \theta \in \Theta\} \subset \mathcal{F}$, where $\Theta \subseteq \mathbb{R}^p$; in this case the prior is said *parametric*. We see that, in the case of a parametric prior, the representation theorem implies that $Y_1, Y_2, \ldots$ are conditionally i.i.d., given θ, with common d.f. $F(\cdot|\theta)$, and θ has a prior distribution $\pi(\theta)$. This is the conditional i.i.d. dependence structure that we have discussed in the previous subsection.

Heterogeneous data

Exchangeability is the simplest dependence structure, which allows to enlighten the basic aspects of Bayesian inference. It is appropriate when we believe that the data are "homogeneous". However, in many problems the dependence structure is more complex. Often, it is appropriate to allow some "heterogeneity" among the data, assuming that

$$Y_1, \ldots, Y_n | \theta_1, \ldots, \theta_n \sim \prod_{t=1}^{n} f_t(y_t|\theta_t),$$

that is, $Y_1, \ldots, Y_n$ are conditionally independent given a vector $\theta = (\theta_1, \ldots, \theta_n)$, with Y_t depending only on the corresponding θ_t. For example, Y_t could be the expense of customer t for some service, and we might assume that each customer has a different average expense θ_t, introducing heterogeneity, or "random effects", among customers. In other applications, t might denote time; for example, each Y_t could represent the average sales in a sample of stores, at time t; and we might assume that $Y_t|\theta_t \sim \mathcal{N}(\theta_t, \sigma^2)$, with θ_t representing the expected sales at time t.

In these cases, the model specification is completed by assigning the probability law of the vector $(\theta_1, \ldots, \theta_n)$. For modeling random effects, a common assumption is that $\theta_1, \ldots, \theta_n$ are i.i.d. according to a distribution G. If there is uncertainty about G, we can model $\theta_1, \ldots, \theta_n$ as conditionally i.i.d. given G, with common distribution function G, and assign a prior on G.

If $(Y_t, t = 1, 2, \ldots)$ is a sequence of observations over time, then the assumption that the θ_t's are i.i.d., or conditionally i.i.d., is generally not appropriate, since we want to introduce a temporal dependence among them. As we shall see in chapter 2, in state space models we assume a Markovian dependence structure among the θ_t's.

We will return on this problem in the next section (example 2).

1.1.2 Synthesis of conditional distributions

We have seen that Bayesian inference is simply solved, in principle, by computing the conditional probability distributions of the quantities of interest: the posterior distribution of the parameters of the model, or the predictive distribution. However, especially when the quantity of interest is multivariate, one might want to present a summary of the posterior or predictive distribution. Consider the case of inference on a multivariate parameter $\theta = (\theta_1, \ldots, \theta_p)$. After computing the joint posterior distribution of θ, if some elements of θ are regarded as nuisance parameters, one can integrate them out to obtain the (marginal) posterior of the parameters of interest. For example, if $p = 2$, we can marginalize the joint posterior $\pi(\theta_1, \theta_2|y)$ and compute the marginal posterior density of θ_1

$$\pi(\theta_1|y) = \int \pi(\theta_1, \theta_2|y)\mathrm{d}\theta_2.$$

We can provide a graphical representation of the marginal posterior distributions, or some summary values, such as the posterior expectations $\mathrm{E}(\theta_i|y)$ or the posterior variances $\mathrm{Var}(\theta_i|y)$, and so on. We can also naturally show intervals (usually centered on $\mathrm{E}(\theta_i|y)$) or bands with high posterior probability.

More formally, the choice of a summary of the posterior distribution (or of the predictive distribution) is regarded as a decision problem. In a statistical decision problem we want to choose an action in a set $\mathcal{A}$, called the action space, on the basis of the sample y. The consequences of an action a are expressed through a loss function $L(\theta, a)$. Given the data y, the Bayesian decision rule selects an action in $\mathcal{A}$ (if there is one) that minimizes the conditional expected loss, $\mathrm{E}(L(\theta, a)|y) = \int L(\theta, a)\pi(\theta|y)\mathrm{d}\nu(\theta)$. Bayesian point estimation is formalized as a decision problem where the action space coincides with the parameter space Θ. The choice of the loss function depends on the problem at hand, and of course different loss functions give rise to different Bayesian estimates of θ. Some forms of the loss function are of particular interest.

(*Quadratic loss*). Let θ be a scalar. A common choice is a quadratic loss function $L(\theta, a) = (\theta - a)^2$. Then the posterior expected loss is $\mathrm{E}((\theta - a)^2|y)$, which is minimized at $a = \mathrm{E}(\theta|y)$. So, the Bayesian estimate of θ with quadratic loss is the posterior expected value of θ. If θ is p-dimensional, a quadratic loss function is expressed as $L(\theta, a) = (\theta - a)'H(\theta - a)$, for a positive definite matrix H. The Bayesian estimate of θ is the vector of posterior expectations $E(\theta|y)$.

(*Linear loss*). If θ is scalar and

$$L(\theta, a) = \begin{cases} c_1|a - \theta| & a \leq \theta \\ c_2|a - \theta| & a > \theta, \end{cases}$$

where c_1 and c_2 are positive constants, then the Bayesian estimate is the $c_1/(c_1 + c_2)$ quantile of the posterior distribution. If $c_1 = c_2$, the estimate is the posterior median.

(*Zero-one loss*). If θ is a discrete random variable and

$$L(\theta, a) = \begin{cases} c & a \neq \theta \\ 0 & a = \theta, \end{cases}$$

the Bayesian estimate is any mode of the posterior distribution.

Similarly, a Bayesian point forecast of Y_{n+1} given $y_1, \ldots, y_n$ is a synthesis of the predictive density with respect to a loss function, which expresses the forecast error in predicting Y_{n+1} with a value $\hat{y}$, say. With a quadratic loss function, $L(y_{n+1}, \hat{y}) = (y_{n+1} - \hat{y})^2$, the Bayesian point forecast is the expected value $\mathrm{E}(Y_{n+1}|y_1, \ldots, y_n)$.

Again, point estimation or forecasting is coherently treated in the Bayesian approach, on the basis of statistical decision theory. However, in practice the

computation of the Bayes solutions can be difficult. If θ is multivariate and the model structure complex, posterior expectations or more generally integrals of the kind $\int g(\theta)\pi(\theta|y)d\theta$ can be analytically untractable. In fact, despite its attractive theoretical and conceptual coherence, the diffusion of Bayesian statistics in applied fields has been hindered, in the past, by computational difficulties, which had restricted the availability of Bayesian solutions to rather simple problems. As we shall see in section 1.3, these difficulties can be overcome by the use of modern simulation techniques.

Example 1. If $Y_1, \ldots, Y_n|\theta$ are i.i.d. with $Y_t|\theta \sim \mathcal{N}(\theta, \sigma^2)$ and $\theta \sim \mathcal{N}(m_0, C_0)$, the posterior density is $\mathcal{N}(m_n, C_n)$, where m_n and C_n are given by (1.4) and (1.5). The Bayesian estimate of θ with quadratic loss is $E(\theta|y_1, \ldots, y_n) = m_n$, a weighted average between the prior guess m_0 and the sample mean $\bar{y}$. Note that, if the sample size is large, then the weight of the prior guess decreases to zero, and the posterior density concentrates around $\bar{y}$, which is the maximum likelihood estimate (MLE) of θ.

This asymptotic behavior of the posterior density holds more generally. Let $Y_1, Y_2, \ldots$ be conditionally i.i.d. given θ, with $Y_t|\theta \sim f(y|\theta)$ and $\theta \in \mathbb{R}^p$ having prior density $\pi(\theta)$. Under general assumptions, it can be proved that the posterior density $\pi(\theta|y_1, \ldots, y_n)$, for n large, can be approximated by a Normal density centered on the MLE $\hat{\theta}_n$. This implies that, in these cases, Bayesian and frequentist estimates tend to agree for a sufficiently large sample size. For a more rigorous discussion of asymptotic normality of the posterior distribution, see Bernardo and Smith (1994) (section 5.3), or Schervish.....

Example 2. A classical problem is estimating the mean of a multivariate Normal distribution. In the classical formulation, the problem is as follows. Suppose that $Y_1, \ldots, Y_n$ are independent r.v.'s, with $Y_t \sim \mathcal{N}(\theta_t, \sigma^2), t = 1, \ldots, n$, where σ^2 is a known constant. As in section 1.1 (heterogeneous data), the Y_t's could be sample means, in n independent experiments; but note that here $\theta = (\theta_1, \ldots, \theta_n)$ is regarded as a vector of unknown constants. Thus we have

$$Y = (Y_1, \ldots, Y_n)' \sim \mathcal{N}_n(\theta, \sigma^2 I_n)$$

and the problem is estimating the mean vector θ. The MLE of θ, which is also the UMVUE estimator, is given by the vector of sample means: $\hat{\theta} = \hat{\theta}(Y) = Y$. However, an important result, which had a great impact when Stein proved it in 1956, shows that the MLE is not optimal with respect to a quadratic loss function: $L(\theta, a) = (\theta - a)'(\theta - a) = \sum_{t=1}^{n}(\theta_t - a_t)^2$, if $n \geq 3$. The overall expected loss, or mean square error, of $\hat{\theta}$ is

$$\mathrm{E}((\theta - \hat{\theta}(Y))'(\theta - \hat{\theta}(Y))) = \mathrm{E}(\sum_{t=1}^{n}(\theta_t - \hat{\theta}_t(Y))^2)$$

where the expectation is w.r.t. the density $f_\theta(y)$, i.e. the $\mathcal{N}_n(\theta, \sigma^2 I_n)$. Stein (1956) proved that, if $n \geq 3$, there exists another estimator $\theta^* = \theta^*(Y)$ which

is more efficient than the MLE $\hat{\theta}$ in the sense that

$$\mathrm{E}((\theta - \theta^*(Y))'(\theta - \theta^*(Y))) < \mathrm{E}((\theta - \hat{\theta}(Y))'(\theta - \hat{\theta}(Y))) \text{ for all } \theta.$$

Stein estimator is given by $\theta^*(Y) = (1-(n-2)/Y'Y)Y$ (for $\sigma^2 = 1$); it shrinks the sample means $Y = (Y_1, \ldots, Y_n)$ towards zero. More generally, *shrinkage estimators* shrink the sample means towards the overall mean $\bar{y} = \sum_{t=1}^{n} y_i$, or towards different values.

Note that the MLE of θ_t, that is $\hat{\theta}_t = y_t$, does not make use of the data y_j, for $j \neq t$, which come from the other independent experiments. Thus, Stein result seems quite surprising, showing that a more efficient estimator of θ_t can be obtained using the information from "independent" experiments. Borrowing strength from different experiments is in fact quite natural in a Bayesian approach. The vector θ is regarded as a random vector, and the Y_t's are *conditionally* independent given $\theta = (\theta_1, \ldots, \theta_n)$, with $Y_t|\theta_t \sim \mathcal{N}(\theta_t, \sigma^2)$, that is

$$Y|\theta \sim \mathcal{N}_n(\theta, \sigma^2 I_n).$$

With a $N_n(m_0, C_0)$ prior density for θ, the posterior density is $\mathcal{N}_n(m_n, C_n)$ where

$$m_n = (C_0^{-1} + \sigma^{-2} I_n)^{-1}(C_0^{-1} m_0 + \sigma^{-2} I_n y)$$

and $C_n = (C_0^{-1} + \sigma^{-2} I_n)^{-1}$. Thus the posterior expectation m_n provides a shrinkage estimate, shrinking the sample means towards the value m_0. Clearly, the shrinkage depends on the choice of the prior; see Lindley and Smith (1972) ....

1.1.3 Choice of the prior distribution

The explicit use of prior information, besides the information from the data, is a basic aspect of Bayesian inference. Indeed, some prior knowledge of the phenomenon under study is always needed: data never speak entirely by themselves. The Bayesian approach allows to explicitly introduce all the information we have (from experts' opinions, from previous studies, from the theory and from the data) in the inferential process. However, the choice of the prior can be a delicate point in practical applications. Here we briefly summarizes some basic notions, but first let us underline a fundamental point, which is clearly enlightened in the case of exchangeable data: the choice of a prior is in fact the choice of the *pair* $f(y|\theta)$ and $\pi(\theta)$. Often, the choice of $f(y|\theta)$ is called *model specification*, but in fact it is part, with the specification of $\pi(\theta)$, of the subjective choices that we have to do for studying a phenomenon, based of our prior knowledge. Anyway, given $f(y|\theta)$, the prior $\pi(\theta)$ should be a honest expression of our beliefs about θ, with no mathematical restrictions on its form.

That said, there are some practical aspects that deserve some consideration. For computational convenience, it is common practice to use *conjugate*

priors. A family of densities on θ is said to be conjugate to the model $f(y|\theta)$ if, when the prior is in that family, so is the posterior. In the example in section 1.1, we used a Gaussian prior density $\mathcal{N}(m_0, C_0)$ on θ, and the posterior resulted still Gaussian, with updated parameters, $\mathcal{N}(m_n, C_n)$; thus, the Gaussian family is conjugate to the model $f(y|\theta) = \mathcal{N}(\theta, \sigma^2)$ (with σ^2 non random). In general, a prior will be conjugate when it has the same analytic form of the likelihood, regarded as a function of θ. Clearly this definition does not determine uniquely the conjugate prior for a model $f(y|\theta)$. For the exponential family, we have a more precise notion of *natural conjugate prior* which is defined from the density of the sufficient statistics; see e.g. Bernardo and Smith (1994), section 5.2; ..... It is worth noting that natural conjugate priors for the exponential family can be quite rigid, and *enriched* conjugate priors have been proposed (see.. Consonni and Veronese...Zellner.....). Furthermore, it can be proved that any prior for an exponential family parameter can be approximated by a mixture of conjugate priors (see Dalal and Hall (1983), Diaconis and Ylvisaker (1985)). We provide some examples below and in the next section. Anyway, computational ease has become less stringent in recent years, due to the availability of simulation-based approximation techniques.

In practice, people quite often use *default priors* or *non-informative priors*, for expressing a situation of "prior ignorance" or vague prior information.appropriately defining the idea of "prior ignorance", or of a prior with "minimal effect" relative to the data on the inferential results, has a long history and is quite delicate; see e.g. Bernardo and Smith (1994), section 5.6.2; O'Hagan (1994), section....; Robert (2001), section..... If the parameter θ takes values in a finite set, $\{\theta_1^*, \ldots, \theta_k^*\}$ say, then the classical notion of a non-informative prior, since Bayes (1763) and Laplace (1814), is of a uniform distribution, $\pi(\theta_j^*) = 1/k$. However, even in this simple case it can be shown that care is needed in defining the quantity of interest; see Bernardo and Smith (1994) section 5.6.2. Anyway, extending the notion of a uniform prior when the parameter space is infinite clearly leads to *improper* distributions, that cannot be regarded as (σ-additive) probability distributions. For example, if $\theta \in (-\infty, +\infty)$, a uniform prior would be a constant and its integral on the real line would be infinite. Furthermore, a uniform distribution for θ implies a non-uniform distribution for any non-linear monotone transformation of θ and thus the Bayes-Laplace postulate is inconsistent in the sense that, intuitively, "ignorance about θ" should also imply "ignorance" about one-to-one transformations of it. Priors based on invariance considerations are Jeffrey's priors (...). Widely used are also *reference priors*, suggested by Bernardo (...), on an information-decisional theoretical base (see e.g. Bernardo and Smith (1994), section 5.4). The use of improper priors is debatable, but often the posterior density from an improper prior returns to be proper, so that improper priors are anyway widely used, also for reconstructing frequentist results in a Bayesian framework. For example, if $Y_t|\theta$ are i.i.d. $\mathcal{N}(\theta, \sigma^2)$, using a noninformative uniform prior $\pi(\theta) = c$ and formally applying the Bayes formula gives

$$\pi(\theta|y_1,\ldots,y_n) \propto \exp\{-\frac{1}{2\sigma^2}\sum_{t=1}^{n}(y_t-\theta)^2\} \propto \exp\{-\frac{n}{2\sigma^2}(\theta^2-2\theta\bar{y})^2\},$$

that is, the posterior is $\mathcal{N}(\bar{y},\sigma^2/n)$. In this case, the Bayes point estimate with quadratic loss corresponds to the MLE $\bar{y}$ of θ. As we noted before, starting with a proper, Gaussian prior would give a posterior density centered around the sample mean only if the prior variance C_0 is very large compared to σ^2, or if the sample size n is large.

Another common practice is to have a hierarchical specification of the prior density. This means that θ has density $\pi(\theta|\lambda)$ conditionally on some hyperparameter λ, and then a prior $h(\lambda)$ is assigned to λ. This is often a way for expressing a kind of uncertainty in the choice of the prior density. Clearly it corresponds to assuming that $\theta \sim \int \pi(\theta|\lambda)\mathrm{d}h(\lambda)$.

For avoiding theoretical and computational difficulties related to the use of improper priors, in this book we will use only proper priors. However, we underline that it is clearly relevant to be aware of the information that we introduce through the choice of the model and the prior density, that is, being aware of the effect of the model specification and of the choice of prior hyperparameters on the inferential results, possibly providing some sensitivity analysis.

Example: Bayesian conjugate inference for univariate Gaussian models

In section 1.1 we considered conjugate Bayesian analysis for the mean of a Gaussian population, with known variance. Let now $Y_1,\ldots,Y_n|\theta,\sigma^2$ i.i.d. $\sim \mathcal{N}(\cdot;\theta,\sigma^2)$, where both θ and σ^2 are unknown. It will be convenient to work with the *precision* $\phi = 1/\sigma^2$ rather than with the variance σ^2. A conjugate prior for (θ,ϕ) can be obtained noting that the likelihood can be written as

$$f(y_1,\ldots,y_n|\theta,\phi) \propto \phi^{(n-1)/2}\exp\{-\frac{1}{2}\phi\, ns^2\}\ \phi^{1/2}\exp\{-\frac{n}{2}\phi(\mu-\bar{y})^2\}$$

(add and subtract $\bar{y}$ in the squared term and note that the cross product is zero), where $\bar{y}$ is the sample mean and $s^2 = \sum_{t=1}^{n}(y_i-\bar{y})^2/n$ is the sample variance. We see that, as a function of (θ,ϕ), the likelihood is proportional to the kernel of a Gamma density in ϕ with parameters $(n/2+1, ns^2/2)$ times the kernel of a Normal density in θ, with parameters $(\bar{y},(n\phi)^{-1})$ (definition and basic properties of the distributions introduced here are provided in the Appendix). Therefore, a conjugate prior for (θ,σ^2) is such that ϕ has a Gamma density with parameters (a,b) and, conditionally on ϕ, θ has a Normal density with parameters $(m_0,(n_0\phi)^{-1})$. The joint prior density is

$$\begin{aligned}\pi(\theta,\phi) &= \pi(\phi)\ \pi(\theta|\phi) = Ga(\phi;a,b)\ \mathcal{N}(\theta;m_0,(n_0\phi)^{-1})\\ &\propto \phi^{a-1}\exp\{-b\phi\}\ \phi^{1/2}\exp\{-\frac{n_0}{2}\phi(\theta-m_0)^2\},\end{aligned}$$

and it is called Normal-Gamma, here with parameters $(m_0,(n_0)^{-1},a,b)$. In particular, $\mathrm{E}(\theta|\phi) = m_0$ and $\mathrm{Var}(\theta|\phi) = (n_0\phi)^{-1} = \sigma^2/n_0$, that is, the variance of θ, given σ^2, is expressed as a proportion $1/n_0$ of σ^2. Marginally,

$\mathrm{E}(\theta) = \mathrm{E}(\mathrm{E}(\theta|\phi)) = m_0$ and $\mathrm{Var}(\theta) = \mathrm{E}(\sigma^2)/n_0 = (b/(a-1))/n_0$ (the variance $\sigma^2 = \phi^{-1}$ has an Inverse-gamma density, with $\mathrm{E}(\sigma^2) = b/(a-1)$). Furthermore, it can be shown that the marginal density of θ is a non-central Student-t with parameters $(m_0, (n_0\, a/b)^{-1})$ and $2a$ degrees of freedom; in symbols, $\theta \sim \mathcal{T}(\theta; m_0, (n_0\, a/b)^{-1}, 2a)$.

With a conjugate Normal-Gamma prior, the posterior of (θ, ϕ) is still Normal-Gamma, with updated parameters. We have to do some computations

$$\pi(\theta, \phi|y_1, \ldots, y_n) \propto$$

$$\phi^{\frac{n}{2}+a-1}\ \exp\{-\frac{1}{2}\phi(ns^2+2b)\}\ \phi^{\frac{1}{2}} \exp\{-\frac{1}{2}\phi n[(\theta-\bar{y})^2 + n_0(\theta_0)^2]\};$$

with some algebra and completing the square that appears in it, the last exponential term can be written as

$$\exp\{-\frac{1}{2}\phi[nn_0\frac{(m_0-\bar{y})^2}{n_0+n} + (n_0+n)(\theta - \frac{n\bar{y}+n_0m_0}{n_0+n})^2]\}$$

so that

$$\pi(\theta, \phi|y_1, \ldots, y_n) \propto$$

$$\phi^{\frac{n}{2}+a-1}\ \exp\{-\frac{1}{2}\phi(ns^2+2b+nn_0\frac{(m_0-\bar{y})^2}{n_0+n})\}\ \phi^{\frac{1}{2}} \exp\{-\frac{1}{2}\phi(n_0+n)(\theta-m_n)^2\}.$$

We see that the parameters of the posterior Normal-Gamma density are

$$\begin{aligned} m_n &= \frac{n\bar{y}+n_0m_0}{n_0+n} \\ n_n &= n_0+n \\ a_n &= a+\frac{n}{2} \\ b_n &= b+\frac{1}{2}ns^2+\frac{1}{2}\frac{nn_0}{n_0+n}(\bar{y}-m_0)^2\,. \end{aligned} \tag{1.9}$$

This means that

$$\phi|y_1, \ldots, y_n \sim Ga(a_n, b_n);$$

$$\theta|\phi, y_1, \ldots, y_n \sim \mathcal{N}(m_n, (n_n\phi)^{-1}).$$

Clearly, conditionally on ϕ, we are back to the case of inference on the mean of a $\mathcal{N}(\theta, \phi^{-1} = \sigma^2)$ with known variance; you can check that the expressions of $E(\theta|\phi, y_1, \ldots, y_n) = m_n$ and $V(\theta|\phi, y_1, \ldots, y_n) = ((n_0+n)\phi)^{-1} = \sigma^2/(n_0+n)$ given above correspond to (1.4) and (1.5), when $C_0 = \sigma^2/n_0$. Here, n_0 has a role of "prior sample size". The marginal density of $\theta|y_1, \ldots, y_n$ is obtained by marginalizing the joint posterior of (θ, ϕ) and results to be a non-central Student-t, with parameters $m_n, (n_n\, a_n/b_n)^{-1}$ and $2a_n$ degrees of freedom.

The predictive density is also Student-t:

$$Y_{n+1}|y_1, \ldots, y_n \sim \mathcal{T}(m_n, \frac{b_n}{a_nn_n}(1+n_n), 2a_n).$$

The recursive formula to update the distribution of (θ, ϕ) when a new observation y_n becomes available is

$$\begin{aligned} m_n &= m_{n-1} + \frac{1}{n_{n-1}+1}(y_n - m_{n-1}) \\ n_n &= n_{n-1} + 1 \\ a_n &= a_{n-1} + \frac{1}{2} \\ b_n &= b_{n-1} + \frac{1}{2}\frac{n_{n-1}}{n_{n-1}+1}(y_n - m_{n-1})^2 \end{aligned}$$

1.2 Bayesian inference in the linear regression model

Dynamic linear models can be regarded as a generalization of the usual linear regression model, where the regression coefficient are allowed to change over time. Therefore for the reader convenience we remind briefly here the basic elements of Bayesian analysis of the static linear regression model.

The linear regression model is the most popular tool for relating the variable Y to explanatory variables x. It is defined as

$$Y_t = x_t'\beta + \epsilon_t, \quad t = 1, \ldots, n, \quad \epsilon_t \sim \mathcal{N}(0, \sigma^2) \tag{1.10}$$

where Y_t is a random variable and x_t and β are $(p \times 1)$ vectors. In its basic formulation, the variables x are considered as deterministic or exogenous; while in stochastic regression x are random variables. In the latter case, we have in fact a random $(p+1) \times 1$ vector (Y_t, X_t) and we have to specify its joint distribution and derive the linear regression model from it. A way for doing this (but more general approaches are possible) is to assume that the joint distribution is Gaussian

$$\begin{pmatrix} Y_t \\ X_t \end{pmatrix} |\beta, \Sigma \sim \mathcal{N}\left(\begin{pmatrix} \mu_y \\ \mu_x \end{pmatrix}, \Sigma\right), \qquad \Sigma = \begin{pmatrix} \Sigma_{yy} & \Sigma_{yx} \\ \Sigma_{xy} & \Sigma_{xx} \end{pmatrix}.$$

From the properties of the multivariate Gaussian distribution (see the Appendix), we can decompose the joint distribution into a marginal model for X_t and a conditional model for Y_t given x_t:

$$X_t|\beta, \Sigma \sim \mathcal{N}(\mu_x, \Sigma_{xx}),$$

$$Y_t|x_t, \beta, \Sigma \sim \mathcal{N}(x_t'\beta, \sigma^2),$$

where

$$\beta = \Sigma_{xx}^{-1}\Sigma_{xy},$$

$$\sigma^2 = \Sigma_{yy} - \Sigma_{yx}\Sigma_{xx}^{-1}\Sigma_{xy}.$$

If the prior distribution on (β, Σ) is such that the parameters of the marginal model and those of the conditional model are independent, then we have a cut

in the distribution of $(Y_t, X_t, \beta, \Sigma)$; in other words, if our interest is mainly on the variable Y, we can restrict our attention to the conditional model. In this case the regression model describes the conditional distribution of Y_t given (β, Σ) and x_t.

With the above remarks, model (1.10) gives

$$Y|X, \beta, V \sim \mathcal{N}_n(X\beta, V), \tag{1.11}$$

where $Y = (Y_1, \ldots, Y_n)'$ and X is the $(n \times p)$ matrix with t-th row x_t'. The covariance matrix V is usually supposed to be diagonal, $V = \sigma^2 I_n$, where I_n is the n-dimensional identity matrix; this means that the Y_t are conditionally independent, with the same variance σ^2. More generally, V is a symmetric positive-definite matrix. In the Bayesian approach, the unknown parameters of the model (the regression coefficients and/or the covariance matrix) are regarded as random quantities, and we have to describe our uncertainty on them through a prior distribution. Inference on the parameters of the model is then solved by computing their posterior distribution.

We describe Bayesian inference with conjugate priors for the regression model, for three cases: inference on the regression coefficients β, assuming that V is known; inference on the covariance matrix V when β is known; inference on β and V.

Inference on the regression coefficients

Here we suppose that V is known and we are interested in inference about the regression coefficients β given the data y. As briefly discussed in the previous section, a conjugate prior for β can be obtained by looking at the likelihood as a function of β. The likelihood for the regression model (1.11) is

$$\begin{aligned} f(y|\beta, \sigma^2, X) &= (2\pi)^{-n/2}|V|^{-n/2} \exp\{-\frac{1}{2}(y - X\beta)'V^{-1}(y - X\beta)\} \qquad (1.12) \\ &\propto |V|^{-n/2} \exp\{-\frac{1}{2}(y'V^{-1}y - 2\beta'X'V^{-1}y + \beta'X'V^{-1}X\beta)\} \end{aligned}$$

where $|V|$ denotes the determinant of V. Now, note that, if $\beta \sim \mathcal{N}(m, C)$ then $\pi(\beta) \propto \exp\{-\frac{1}{2}(\beta - m)'C^{-1}(\beta - m)\} \propto \exp\{-\frac{1}{2}(\beta'C^{-1}\beta - 2\beta'C^{-1}m)\}$. Therefore, we see that the likelihood, as a function of β, is proportional to the kernel of a Normal density, with parameters $((X'V^{-1}X)^{-1}X'Vy, (X'V^{-1}X)^{-1})$. Thus, a conjugate prior for β is the normal density, $\mathcal{N}(m_0, C_0)$ say; as usual, m_0 represent a prior guess about β; the elements on the diagonal of C_0 express prior uncertainty on the prior guess m_0 and (quite relevant) the off-diagonal elements of C_0 model the dependence among the regression coefficients β_t's.

With a conjugate Gaussian prior, the posterior will be Gaussian, too, with updated parameters. For deriving the expression of the posterior parameters, we compute the posterior density by the Bayes formula:

$$\pi(\beta|Y,X,V) \propto \exp\{-\frac{1}{2}(\beta' X'V^{-1}X\beta - 2\beta' X'V^{-1}y\} \ \exp\{-\frac{1}{2}(\beta - m_0)'C_0^{-1}(\beta - m_0)\}$$
$$\propto \exp\{-\frac{1}{2}(\beta'(X'V^{-1}X + C_0^{-1})\beta - 2\beta'(X'V^{-1}y + C_0^{-1}m_0))\}.$$

We recognize the kernel of a p-variate Gaussian density with parameters

$$m_n = C_n(X'V^{-1}y + C_0^{-1}\beta_0)$$
$$C_n = (C_0^{-1} + X'V^{-1}X)^{-1}.$$

The Bayesian point estimate of β, w.r.t. a quadratic loss function, is the posterior expected value $\mathrm{E}(\beta|X,y) = m_n$. Note that it does not require the assumption that $(X'V^{-1}X)^{-1}$ exists, which is instead necessary for computing the classical generalized least square estimate of β, that is $\hat{\beta} = (X'V^{-1}X)^{-1}X'Vy$. However, when $(X'V^{-1}X)$ is non-singular, the Bayes estimate m_n can be written as

$$m_n = (C_0^{-1} + X'V^{-1}X)^{-1}(X'V^{-1}X\hat{\beta} + C_0^{-1}m_0),$$

that is, as a linear combination of the prior guess m_0, with weight proportional to the prior precision matrix C_0^{-1}, and of the generalized least square estimate $\hat{\beta}$, whose weight is proportional to the precision matrix $X'V^{-1}X$ of $\hat{\beta}$. Clearly m_n is a shrinkage estimator of the regression coefficients; see Lindley and Smith (1972).

The posterior precision matrix is the sum of the prior precision C_0^{-1} and of $X'V^{-1}X$. Of course, one can integrate the joint posterior density of β for obtaining the marginal posterior density of one or more coefficients β_j.

For the analysis that we will do in the next chapter, when studying dynamic linear models, it is useful to provide an alternative "recursive" expression of the posterior parameters. It can be proved that the posterior variance can be rewritten as

$$C_n = (X'V^{-1}X + C_0^{-1})^{-1} = C_0 - C_0X'(XC_0X' + V)^{-1}XC_0 \qquad (1.13)$$

(see *problem 1.1*). Using the above identity, it can be shown that the posterior expectation m_n can be expressed as

$$m_n = m_0 + C_0X'(XC_0X' + V)^{-1}(y - Xm_0) \qquad (1.14)$$

(see *problem 1.2*). Note that $Xm_0 = E(Y|\beta, X)$ is the prior point forecast of Y. So, the above expression writes the Bayes estimate of β as the prior guess m_0 corrected by a term which takes into account the forecast error $(y - Xm_0)$.

Inference on the covariance matrix

Suppose now that β is known and we are interested in inference on the covariance matrix V. Analogously to the case of inference on the parameters of the

Gaussian univariate model, it is convenient to work with the precision matrix $\Phi = V^{-1}$. For determining a conjugate prior for Φ, note that we can write the likelihood (1.12) as

$$\begin{aligned} f(y|\beta, \Phi, X) &\propto |\Phi|^{n/2} \exp\{-\frac{1}{2}(y - X\beta)'\Phi(y - X\beta)\} \\ &= |\Phi|^{n/2} \exp\{-\frac{1}{2}tr(y - X\beta)(y - X\beta)'\Phi\}, \end{aligned}$$

where $tr(A)$ denotes the trace of a matrix A, since $(y - X\beta)'\Phi(y - X\beta) = tr((y - X\beta)'\Phi(y - X\beta))$ (being a scalar) and recalling that $tr(AB) = tr(BA)$. We see that, as a function of Φ, the likelihood is proportional to the kernel of a Wishart density with parameters $(n + 1/2, 1/2(y - X\beta)(y - X\beta)')$ (see the Appendix). So, a conjugate prior for the precision Φ is Wishart

$$\Phi \sim Wishart(\nu_0, S_0).$$

The posterior will be Wishart with updated parameters,

$$\Phi|Y, X, \beta \sim Wishart(\nu_n, S_n)$$

and it can be easily checked that

$$\begin{aligned} \nu_n &= \nu_0 + \frac{n}{2} \\ S_n &= \frac{1}{2}(y - X\beta)(y - X\beta)' + S_0 \end{aligned}$$

Inference on (β, V)

Now let both β and V be random. We consider two cases. First, we assume that V has the form $V = \sigma^2 D$, where σ^2 is a random variable and the $(n \times n)$ matrix D is known; a common assumption is $D = I_n$. We then consider the case of a general unknown covariance matrix V.

In the case $V = \sigma^2 D$, let $\phi = \sigma^{-2}$. A conjugate prior for (β, ϕ) is a Normal-Gamma, with parameters $(\beta_0, N_0^{-1}, a, b)$

$$\pi(\beta, \phi) \propto \phi^{a-1} \exp\{-b\phi\}\, \phi^{\frac{p}{2}} \exp\{-\frac{\phi}{2}(\beta - \beta_0)'N_0(\beta - \beta_0)\}$$

that is

$$\begin{aligned} \beta|\phi &\sim \mathcal{N}(\beta_0, (\phi N_0)^{-1}) \\ \phi &\sim Ga(a, b) \end{aligned}$$

Note that, conditionally on ϕ, β has covariance matrix $(\phi N_0)^{-1} = \sigma^2 \tilde{C}_0$ where we let $\tilde{C}_0 = N_0^{-1}$, a symmetric $(p \times p)$ positive-definite matrix which 'rescales' the observation variance σ^2.

It can be shown (see problem 1.3) that the posterior is a Normal-Gamma with parameters

$$\begin{aligned}\beta_n &= \beta_0 + \tilde{C}_0 X'(X\tilde{C}_0 X' + D)^{-1}(y - X\beta_0), \\ \tilde{C}_n &= \tilde{C}_0 - \tilde{C}_0 X'(X\tilde{C}_0 X' + D)^{-1} X\tilde{C}_0 \\ a_n &= a + \frac{n}{2} \\ b_n &= b + \frac{1}{2}(\beta_0' \tilde{C}_0^{-1}\beta_0 + y' D^{-1} y - \beta_n' \tilde{C}_n \beta_n)\end{aligned} \tag{1.15}$$

Furthermore, we can simplify the expression of b_n; in particular, it can be shown that

$$b_n = b + \frac{1}{2}(y - X\beta_0)'(D + X\tilde{C}_0 X')^{-1}(y - X\beta_0), \tag{1.16}$$

(see problem 1.3). These formulas have again the estimation-error correction structure that we have underlined in the simple Gaussian model, see (1.6), and in the regression model with known covariance, compare with (1.14).

1.3 Simulation techniques

In Bayesian inference, it is very often the case that the posterior distribution of the parameters, denoted here by ψ, is analytically intractable. By this we mean that it is impossible to derive in closed form summaries of the posterior, such as its mean and variance, or the marginal distribution of a particular parameter. In fact, most of the time the posterior density is only known up to a normalizing factor. To overcome this limitation, the standard practice is to resort to simulation methods. For example, if one could draw a random sample $\psi_1, \ldots, \psi_N$ (i.i.d.) from the posterior distribution π, then, using the standard Monte Carlo method, the mean of any function $g(\psi)$ having finite posterior expectation can be approximated numerically by a sample average:

$$\mathrm{E}_\pi(g(\psi)) \approx N^{-1} \sum_{j=1}^{N} g(\psi_j) \tag{1.17}$$

Unfortunately, independent samples from the posterior are not easy to obtain. Luckily, however, (1.17) holds more generally for dependent samples. In particular, it holds for certain Markov chains. Monte Carlo methods based on simulating random variables from a Markov chain, called Markov chain Monte Carlo (MCMC) methods, are nowadays the standard way of performing the numerical analysis required by Bayesian data analysis. In the next subsections

we review the main general methods that are commonly employed to simulate a Markov chain such that (1.17) holds for a specific π. References ???

For an irreducible, aperiodic and recurrent Markov chain $\{\psi_t\}_{t\geq 1}$, having invariant distribution π, it can be shown that for every[3] initial value ψ_1, the distribution of ψ_t tends to π as t increases to infinity. Therefore, for M sufficiently large, $\psi_{M+1}, \ldots, \psi_{M+N}$ are all approximately distributed according to π and, jointly, they have statistical properties similar to those enjoyed by an independent sample from π. In particular, the law of large numbers, expressed by (1.17), holds in the form

$$\mathrm{E}_\pi(g(\psi)) \approx N^{-1} \sum_{j=1}^{N} g(\psi_{M+j}) \tag{1.18}$$

We note, in passing, that if the Markov chain is only irreducible and recurrent, but has period $d > 1$, (1.18) still holds, even if in this case the distribution of ψ_t depends on where the chain started, no matter how large t is. In practice it is important to determine how large M should be, i.e., how many iterations of a simulated Markov chain are to be considered *burn-in* and discarded in the calculation of ergodic averages (1.18).

Another issue is the assessment of the accuracy of an ergodic average as an estimator of the corresponding expected value. When the ψ_j's are simulated from a Markov chain, the usual formula for estimating the variance of a sample mean in the i.i.d. case no longer holds. For simplicity, suppose that the burn-in part of the chain has already been discarded, so that we can safely assume that ψ_1 is distributed according to π and $\{\psi_t\}_{t\geq 1}$ is a stationary Markov chain. Let $\bar{g}_N$ denote the right-hand side of (1.18) It can be shown that, for N large,

$$\mathrm{Var}(\bar{g}_N) \approx N^{-1}\mathrm{Var}(g(\psi_1))\tau(g),$$

where $\tau(g) = \sum_{t=-\infty}^{+\infty} \rho_t$ and $\rho_t = \mathrm{corr}(g(\psi_s), g(\psi_{s+t}))$. An estimate of the term $\mathrm{Var}(g(\psi_1))$ is provided by the sample variance of $g(\psi_1), \ldots, g(\psi_N)$. In order to estimate $\tau(g)$, Sokal (1989) suggests to truncate the summation and plug in empirical correlations for theoretical correlations:

$$\hat{\tau}_n = \sum_{|t|\leq n} \hat{\rho}_t,$$

with $n = \min\{k : k \geq 3\hat{\tau}_k\}$.

We now briefly present the most popular MCMC algorithms for simulating from a given distribution π.

[3] We omit here some measure-theoretic details, trying to convey only the main ideas. For rigorous results the reader should consult the suggested references.

1.3.1 Gibbs sampler

Suppose that the unknown parameter is multidimensional, so the posterior distribution is multivariate. In this case we can write $\psi = (\psi^{(1)}, \psi^{(2)})$, where $\psi^{(1)}$ and $\psi^{(2)}$ may be unidimensional or multidimensional. Let $\pi(\psi) = \pi(\psi^{(1)}, \psi^{(2)})$ be the target density. The Gibbs sampler starts from an arbitrary point $\psi_0 = (\psi_0^{(1)}, \psi_0^{(2)})$ in the parameter space and alternates updating $\psi^{(1)}$ and $\psi^{(2)}$ by drawing from the relevant conditional distribution, according to the scheme in Table 1.1

0. Set $j = 1$.
1. Draw $\psi_j^{(1)}$ from $\pi(\psi^{(1)}|\psi^{(2)} = \psi_{j-1}^{(2)})$.
2. Draw $\psi_j^{(2)}$ from $\pi(\psi^{(2)}|\psi^{(1)} = \psi_j^{(1)})$.
3. Set $j = j + 1$.
4. If $j > N$ stop, otherwise go to 1.

Table 1.1. Gibbs sampling

1.3.2 Metropolis-Hastings algorithm

A very flexible method to generate a Markov chain having a prescribed invariant distribution is provided by Metropolis-Hastings algorithm [reference ???]. The method is very general, since it allows to generate the next state of the chain from essentially an arbitrary distribution: the invariance of the target distribution is then enforced by an accept/reject step. Suppose that the chain is currently at ψ. Then a *proposal* $\tilde{\psi}$ is drawn from a density $q(\psi, \cdot)$. Note that the proposal density may depend on the current state ψ. The proposal $\tilde{\psi}$ is accepted as the new state of the chain with probability

$$\alpha(\psi, \tilde{\psi}) = \min \left\{ 1, \frac{\pi(\tilde{\psi}) q(\tilde{\psi}, \psi)}{\pi(\psi) q(\psi, \tilde{\psi})} \right\}.$$

If the proposal is rejected, the chain stays in the current state ψ. Table 1.2 describes in detail the steps involved in the algorithm, assuming an arbitrary value ψ_0 for the initial state of the chain.

The choice of the proposal density is an important practical issue. A proposal leading to a high rejection rate will result in a "sticky" Markov chain, in which the state will tend to stay constant for many iterations. Ergodic averages like (1.18) provide in such a situation poor approximations, unless N is extremely large. On the other hand, a high acceptance rate is not guarantee, *per sé*, of a good behavior of the chain. Consider, for example, a uniform proposal on $(\psi - a, \psi + a)$, where a is a very small positive number, and ψ is the current state. In this case $q(\psi, \tilde{\psi})$ is constant, and hence it cancels out in α.

0. Set $j = 1$.
1. Draw $\tilde{\psi}_j$ from $q(\psi_{j-1}, \cdot)$.
2. Compute $\alpha = \alpha(\psi_{j-1}, \tilde{\psi}_j)$.
3. Draw an independent random variable $U_j \sim \text{Ber}(\alpha)$.
4. If $U_j = 1$ set $\psi_j = \tilde{\psi}_j$, otherwise set $\psi_j = \psi_{j-1}$.
5. Set $j = j + 1$.
6. If $j > N$ stop, otherwise go to 1.

Table 1.2. Metropolis-Hastings algorithm

Moreover, since the proposal $\tilde{\psi}$ will be close to ψ, in most cases one will have $\pi(\tilde{\psi}) \approx \pi(\psi)$ and $\alpha \approx 1$. However, the resulting simulated chain will move very slowly through its state space, exhibiting a strong positive autocorrelation, which in turn implies that in order to obtain good approximations via (1.18), one has to take N very large. Generally speaking, one shoud try to devise a proposal that is a good approximation – possibly local, in a neighborhood of the current state – to the target distribution. In the next section we illustrate a general method to construct such a proposal.

The Gibbs sampler and Metropolis-Hastings algorithm are by no means competing approaches to Markov chain simulation: in fact, they can be combined and used together. When taking a Gibbs sampling approach, it may be unfeasible, or simply not practical, to sample from one or more conditional distributions. Suppose for example that $\pi(\psi^{(1)}|\psi^{(2)})$ does not have a standard form and is therefore difficult to simulate from. In this case one can, instead of drawing $\psi^{(1)}$ from $\pi(\psi^{(1)}|\psi^{(2)})$, update $\psi^{(1)}$ using a Metropolis-Hastings step. It can be shown that this does not alter the invariant distribution of the Markov chain.

1.3.3 Adaptive rejection Metropolis sampling

Rejection sampling is a simple algorithm that allows one to generate a random variable from a target distribution π by drawing from a different proposal distribution f and then accepting with a specific probability. Suppose that there is a constant C such that $\pi(\psi) \leq Cf(\psi)$ for every ψ and define $r(\psi) = \pi(\psi)/Cf(\psi)$, so that $0 \leq r(\psi) \leq 1$. Draw two independent random variables U and V, with U uniformly distributed on $(0, 1)$ and $V \sim f$. If $U \leq r(V)$ set $\psi = V$, otherwise repeat the process. In other words, draw V from $f(\psi)$ and accept V as a draw from $\pi(\psi)$ with probability $r(V)$. In case of rejection, restart the process. It can be shown that if the support of π is included in the support of f, the algorithm terminates in a finite time, i.e. one eventually generates a V that is accepted. To see that the resulting draw has the correct distribution, consider that the proposed V is accepted only if $U \leq r(V)$, so that the distribution of an accepted V is not just f, but f conditional on the event $\{U \leq r(V)\}$. Denoting by Π the cumulative distribution function of the target distribution π, one has:

$$\begin{aligned}\mathrm{P}(V \le v, U \le r(V)) &= \int_{-\infty}^{v} \mathrm{P}(U \le r(V)|V = \zeta) f(\zeta)\mathrm{d}\zeta \\ &= \int_{-\infty}^{v} \mathrm{P}(U \le r(\zeta)) f(\zeta)\mathrm{d}\zeta = \int_{-\infty}^{v} r(\zeta) f(\zeta)\mathrm{d}\zeta \\ &= \int_{-\infty}^{v} \frac{\pi(\zeta)}{C f(\zeta)} f(\zeta)\mathrm{d}\zeta = \frac{1}{C}\Pi(v).\end{aligned}$$

Letting v go to $+\infty$, one obtains $\mathrm{P}(U \le r(V)) = C^{-1}$. Therefore,

$$\mathrm{P}(V \le v | U \le r(V)) = \frac{\mathrm{P}(V \le v, U \le r(V))}{\mathrm{P}(U \le r(V))} = \Pi(v).$$

The most favorable situations, in terms of acceptance probability, are obtained when the proposal distribution is close to the target: in this case C can be taken close to one and the acceptance probability $r(\cdot)$ will also be close to one. It is worth noting the analogy with Metropolis-Hastings algorithm. In both methods one generates a proposal from an instrumental density, and then accepts the proposal with a specific probability. However, while in rejection sampling one keeps on drawing proposals until a candidate is accepted, so that, repeating the process, one can generate a sequence of independent draws exactly from the target distribution, in the Metropolis-Hastings algorithm the simulated random variables are in general dependent and are distributed according to the target only in the limit.

If π is univariate, log-concave[4], and it has bounded support, it is possible to construct a continuous piecewise linear envelope for $\log \pi$, see Figure 1.1, which corresponds to a piecewise exponential envelope for π. Appropriately

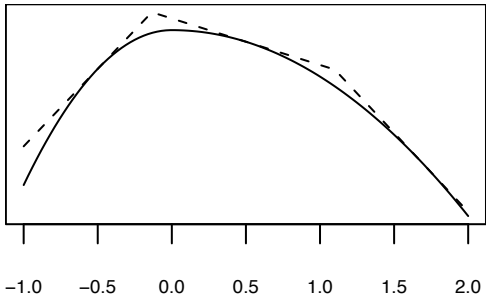

Fig. 1.1. Target log density with a piecewise linear envelope

normalized, this results in a piecewise exponential proposal density, which is easy to sample from using standard random number generators. Moreover,

[4] A function g is concave if it is defined in an interval (a, b) and $g(\alpha x + (1-\alpha)y) \ge \alpha g(x) + (1-\alpha)g(y)$ for every $\alpha \in (0,1)$ and $x, y \in (a, b)$. π is log-concave if $\log \pi(\psi)$ is a concave function.

due to the interplay between C and the normalizing constant of the piecewise exponential density, the target density π needs only to be known up to a normalizing factor. Clearly, the more points one uses in constructing the envelope to the target log density, the closer the proposal density will be to the target, and the sooner a proposal V will be accepted. This suggests an adaptive version of the method, according to which every time a proposal V is rejected, one refines the piecewise linear envelope using the point $(V, \log \pi(V))$, so that the next proposal will be drawn from a density that is closer to π. This algorithm is called adaptive rejection sampling in [ref ???]. If the univariate target π is not logconcave, one can combine adaptive rejection sampling with the Metropolis-Hastings algorithm to obtain a Markov chain having π as invariant distribution. The details are given in ref ???, where the algorithm is termed adaptive rejection Metropolis sampling (ARMS).

Within an MCMC setting, the univariate ARMS algorithm described above can be adapted to work also for a multivariate target distribution using the following simple device. Suppose that the chain is currently at $\psi \in \mathbb{R}^k$. Draw a uniformly distributed unit vector $u \in \mathbb{R}^k$. Then apply ARMS to the univariate density proportional to

$$t \longmapsto \pi(\psi + tu).$$

Up to a normalizing factor, this is the conditional target density, given that the new draw belongs to the straight line through the current ψ and having direction u. The function `arms`, originally written as part of the package `HI`

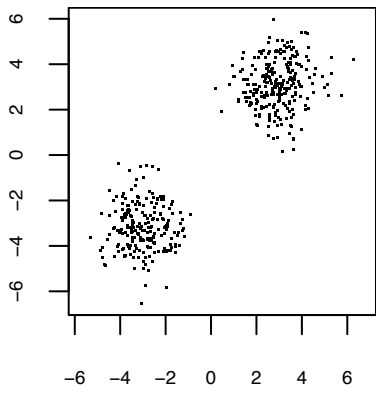

Fig. 1.2. Sample from a mixture of two bivariate normal distributions.

(see Petris and Tardella; 2003) and now included in package `dlm`, performs this kind of multivariate version of ARMS. The function needs the arguments `y.start`, `myldens`, `indFunc`, and `n.sample` for the starting point, a function that evaluates the target logdensity, a function that evaluates the support of the density, and the number of draws to be simulated, respectively. It has also

the additional argument `...` that is passed on to `myldens` and `indFunc`. This is useful when the logdensity and the support depend on additional parameters. Figure 1.2 shows the plot of 500 simulated points from a mixture of two bivariate normal densities with unit variances and independent components and means $(-3, 3)$, $(3, 3)$, respectively. The code below was used to generate the sample.

R code

```
> bimodal <- function(x) log(prod(dnorm(x,mean=3)) +
+                            prod(dnorm(x,mean=-3)))
> supp <- function(x) all(x>(-10)) * all(x<(10))
> y <- arms( c(-2,2), bimodal, supp, 500 )
```

Note that for this target an ordinary Gibbs sampler would very likely get stuck in one of the two modes. This suggests that when one suspects a multivariate posterior distribution to be multimodal, it may be wise to include ARMS in a MCMC, and not to rely solely on a simple Gibbs sampler.

1.4 Appendix. Some useful distributions

Gamma distribution

A random variable X has a Gamma distribution, with parameters (a, b), if it has density

$$Ga(x; a, b) = cx^{a-1} \exp\{-bx\}\, I_{(0,\infty)}(x)$$

where $c = b^a/\Gamma(a)$, $\Gamma(a) = \int_0^\infty x^{a-1}e^{-x}\mathrm{d}x$, and a, b are positive parameters. We find that

$$E(X) = \frac{a}{b} \quad , \quad V(X) = \frac{a}{b^2}\,.$$

If $a > 1$, there is a unique mode at $(a-1)/b$. For $a = 1$, the density reduces to the (negative) exponential distribution with parameter b. For $(a = k/2, b = 1/2)$ it is a chi-square distribution with k degrees of freedom, $\chi^2(k)$.

If $X \sim Ga(a, b)$, the density of $Y = 1/X$ is called Inverse-Gamma, with parameters (a, b), and we have $\mathrm{E}(X) = b/(a-1)$ if $a > 1$ and $\mathrm{Var}(X) = b^2/((a-1)^2(a-2))$ if $a > 2$.

Student-t distribution

If $Z \sim \mathcal{N}(0, 1), U \sim \chi^2(k)$, $k > 0$ and Z and U are independent, then the random variable $T = Z/\sqrt{U/k}$ has a (central) *Student-t distribution* with k degrees of freedom, with density

$$f(t; k) = c\,(1 + \frac{t^2}{k})^{-\frac{k+1}{2}},$$

where $c = \Gamma((k+1)/2)/(\Gamma(k/2)\sqrt{\pi}\sqrt{k})$. We write $T \sim \mathcal{T}(0,1,k)$ or simply $T \sim \mathcal{T}_k$.

It is clear from the definition that the density is positive on the whole real line and symmetric around the origin. It can be shown that, as k increases to infinity, the density converges to a standard Normal density at any point. It can be shown that

$$\mathrm{E}(X) = \mu \text{ if } k > 1$$
$$\mathrm{Var}(X) = \frac{k}{k-2} \text{ if } k > 2$$

If $T \sim \mathcal{T}(0,1,k)$, then $X = \mu + \sigma T$ has a Student-t distribution, with parameters (μ, σ^2) and k degrees of freedom; we write $X \sim \mathcal{T}(\mu, \sigma^2, k)$. Clearly $\mathrm{E}(X) = \mu$ if $k > 1$ and $\mathrm{Var}(X) = \sigma^2 \frac{k}{k-2}$ if $k > 2$.

Normal-Gamma distribution

Let (X, Y) be a bivariate random vector. If $X|Y = y \sim \mathcal{N}(\mu, (n_0 y)^{-1})$, and $Y \sim Ga(a, b)$, then we say that (X, Y) has a Normal-Gamma density with parameters (μ, n_0^{-1}, a, b) (where of course $\mu \in R$, $n_0, a, b \in R^+$). We write $(X, Y) \sim \mathcal{NG}(\mu, n_0^{-1}, a, b)$ The marginal density of X is a Student-t, $X \sim \mathcal{T}(\mu, (n_0 \frac{a}{b})^{-1}, 2a)$.

Multivariate Normal distribution

A continuous random vector $Y = (Y_1, \ldots, Y_k)'$ has a k-variate Normal distribution with parameters $\mu = (\mu_1, \ldots, \mu_k)'$ and Σ, where $\mu \in R^k$ and Σ is a symmetric positive-definite matrix, if it has density

$$\mathcal{N}_k(y; \mu, \Sigma) = |\Sigma|^{-1/2}(2\pi)^{-k/2} \exp\{-\frac{1}{2}(y-\mu)'\Sigma^{-1}(y-\mu)\}, y \in R^k$$

where $|\Sigma|$ denotes the determinant of the matrix Σ. We write

$$Y \sim \mathcal{N}_k(\mu, \Sigma).$$

Clearly, if $k = 1$, so that Σ is a scalar, the $N_k(\mu, \Sigma)$ reduces to the univariate Normal density.

We have $\mathrm{E}(Y_i) = \mu_i$ and, denoting by $\sigma_{i,j}$ the elements of Σ, $\mathrm{Var}(Y_i) = \sigma_{i,i}$ and $Cov(Y_i, Y_j) = \sigma_{i,j}$. The inverse of the covariance matrix Σ, $\Phi = \Sigma^{-1}$ is the precision matrix of Y.

Several results are of interest; their proof can be found in any multivariate analysis textbook (see, e.g. Barra and Herbach; 1981, pp.92,96).

1. If $Y \sim \mathcal{N}_k(\mu, \Sigma)$ and X is a linear transformation of Y, that is $X = AY$ where A is a $n \times k$ matrix, then $X \sim \mathcal{N}_k(A\mu, A\Sigma A')$.

2. Let X and Y be two random vectors, with covariance matrices Σ_X and Σ_Y, respectively. Let Σ_{YX} be the covariance between Y and X, i.e. $\Sigma_{YX} = E((Y - E(Y))(X - E(X))')$. The covariance between X and Y is then $\Sigma_{XY} = \Sigma'_{YX}$. Suppose that Σ_X is nonsingular. Then it can be proved that the joint distribution of (X, Y) is Gaussian if and only if the following conditions are satisfied:
   (i) X has a Gaussian distribution;
   (ii) the conditional distribution of Y given $X = x$ is a Gaussian distribution whose mean is

$$E(Y \mid X = x) = E(Y) + \Sigma_{YX}\Sigma_X^{-1}(x - E(X))$$

   and whose covariance matrix is

$$\Sigma_{Y|X} = \Sigma_Y - \Sigma_{YX}\Sigma_X^{-1}\Sigma_{XY}.$$

Wishart distribution

Let X be a symmetric positive-definite matrix of random variables $x_{i,j}$, $i, j = 1, \ldots, k$. The distribution of X is in fact the distribution of the $k(k-1)/2$-dimensional vector of the distinct entries of X. We say that X has a *Wishart* distribution with parameters α and B (with $\alpha > (k-1)/2$ and B a symmetric, nonsingular matrix), if it has density

$$W(X; \alpha, B) = c|X|^{\alpha-(k+1)/2} \exp\{-tr(BX)\},$$

where $c = |B|^\alpha / \Gamma_k(\alpha)$,

$$\Gamma_k(\alpha) = \pi^{k(k-1)/4} \prod_{i=1}^{k} \Gamma(\frac{2\alpha + 1 - i}{2})$$

is the *generalized gamma function* and $tr(\cdot)$ denotes the trace of a matrix argument. If $k = 1$, so that B is a scalar, then $W(\alpha, B)$ reduces to the Gamma density $Ga(\cdot; \alpha, B)$.

The following properties of the Wishart distribution can be proved.

$$E(X) = \alpha B^{-1} \text{ and } E(X^{-1}) = (\alpha - \frac{k+1}{2})^{-1} B.$$

If $(Y_1, \ldots, Y_n)$, $n > 1$, is a random sample from a multivariate normal distribution $N_k(\cdot; \mu, \Sigma)$ and $\bar{Y} = \sum_{i=1}^n Y_i/n$, then $\bar{Y} \sim N_k(\cdot; \mu, \Sigma/n)$ and

$$S = \sum_{i=1}^{n} (Y_i - \bar{Y})(Y_i - \bar{Y})'$$

is independent of $\bar{Y}$ and has a Wishart distribution $W(\cdot; (n-1)/2, \Sigma^{-1}/2)$.

Multivariate Student-t distribution

If Y is a p-variate random vector with $Y \sim \mathcal{N}_p(0, \Sigma)$ and $U \sim \chi^2(k)$, with Y and U independent, then $X = \frac{Y}{\sqrt{U/k}} + \mu$ has a p-variate Student-t distribution, with parameters (μ, Σ) and $k > 0$ degrees of freedom, with density

$$f(x) = c\,[1 - \frac{1}{k}\,(x-\mu)'\Sigma^{-1}(x-\mu)]^{-(k+p)/2} \quad , x \in \mathbb{R}^p \ ,$$

where $c = \Gamma((k+p)/2)/(\Gamma(k/2)\pi^{p/2}k^{p/2}|\Sigma|^{1/2}$. We write $X \sim \mathcal{T}(\mu, \Sigma, k)$. For $p = 1$ it reduces to the univariate Student-t distribution. We have

$$\mathrm{E}(X) = \mu \text{ if } k > 1$$
$$\mathrm{Var}(X) = \Sigma\frac{k}{k-2} \text{ if } k > 2.$$

Multivariate Normal-Gamma distribution

Let (X, Y) be a random vector, with $X|Y = y \sim \mathcal{N}_m(\mu, (N_0 y)^{-1})$, and $Y \sim Ga(a, b)$. Then we say that (X, Y) has a Normal-Gamma density with parameters (μ, N_0^{-1}, a, b), in symbols $(X, Y) \sim \mathcal{NG}(\mu, N_0^{-1}, a, b)$.

The marginal density of X is a multivariate Student-t, $X \sim \mathcal{T}(\mu, (N_0\,\frac{a}{b})^{-1}, 2a)$, so that $\mathrm{E}(X) = \mu$ and $\mathrm{Var}(X) = N_0^{-1}b/(a-1)$.

Problems

1.1. Verify the identity (1.13).

1.2. Verify the identity (1.14).

1.3. Consider the linear regression model discussed in section 1.2, with $V = \sigma^2 D$ for a known matrix D. Verify that the posterior density for the parameters $(\beta, \phi = \sigma^{-1})$, with a Normal-Gamma prior, in Normal-Gamma, with parameters given by (1.15). Then, verify the identity (1.16).

1.4. *(Shrinkage estimation)*. Consider random variables $Y_1, \ldots, Y_n$ such that

$$Y_1, \ldots, Y_n \mid \theta_1, \ldots, \theta_n \sim \prod_{t=1}^{n} N(y_t \mid \theta_t, \sigma^2),$$

where σ^2 is known.

(a) Verify that, if $\theta_1, \ldots, \theta_n$ are i.i.d. $\sim N(m, \tau)$, then the Y_t are independent. Compute the posterior density $p(\theta_1, \ldots, \theta_n \mid y_1, \ldots, y_n)$. With quadratic loss, the Bayesian estimate of θ_t is $E(\theta_t \mid y_1, \ldots, y_n)$. Comment the expression of $E(\theta_t \mid y_1, \ldots, y_n)$ that you found. What is the posterior variance, $V(\theta_t \mid y_1, \ldots, y_n)$?

(b) Now suppose that

$$\theta_1, \ldots, \theta_n \mid \lambda \text{ i.i.d. } \sim \mathcal{N}(\lambda, \sigma_w^2)$$

$$\lambda \sim N(m, \tau),$$

where m, σ_m^2, τ are known. Compute the posterior density $p(\theta_1, \ldots, \theta_n \mid y_1, \ldots, y_n)$. Comment the expressions of $E(\theta_t \mid y_1, \ldots, y_n)$ and of $V(\theta_t \mid y_1, \ldots, y_n)$ that you found.

1.5. (*Pooling experts opinions).* Let $Y1, \ldots, Y_n$ be i.i.d. random variables conditionally on θ, with $Y_i \mid \theta \sim \mathcal{N}(\theta, \sigma^2)$ with σ^2 known. Suppose that

$$\theta \sim \sum_{j=1}^{k} p_j \mathcal{N}(\mu_j, \tau_j^2).$$

Given $Y_1 = y_1, \ldots, Y_n = y_n$, compute the posterior distribution of θ, and the predictive distribution of Y_{n+1}.

1.6. *ex. on sensitivity to prior specification...*

1.7. *Exercise on Gibbs sampling – e.g. linear model, β and V unknown...*

2

Dynamic linear models

In this chapter we discuss the basic notions about state-space models and their use in time series analysis. Dynamic linear models (DLM) are presented as a special case of general state space models, being linear and Gaussian. For DLM, estimation and forecasting can be obtained recursively by the well known Kalman filter.

2.1 Introduction

In the recent years there has been an increasing interest for the application of state-space models in time series analysis; see for example West and Harrison (1997), Harvey (1989), Durbin and Koopman (2001), the recent overviews by Künsch (2001) and Migon et al. (2005), and the references therein. State-space models consider a time series as the output of a dynamic system perturbed by random disturbances. As we shall see, they allow a natural interpretation of a time series as the result of several components, such as trend, seasonal or regressive components. At the same time, they have an elegant and powerful probabilistic structure, offering a flexible framework for a very wide range of applications. Computations can be implemented by recursive algorithms. The problems of estimation and forecasting are solved by recursively computing the conditional distribution of the quantities of interest, given the available information. In this sense, they are quite naturally treated from a Bayesian approach.

State-space models can be used for modeling univariate or multivariate time series, also in presence of non-stationarity, structural changes, irregular patterns. They include the popular ARMA models as special cases. For having a first idea of their potential use in time series analysis, consider for example the data plotted in figure 2.1. This time series appears fairly predictable, since it repeats quite regularly its behavior over time: we see a trend and a rather regular seasonal component, with a slightly increasing variability. For data of this kind, we would probably be happy with a fairly simple time-series model,

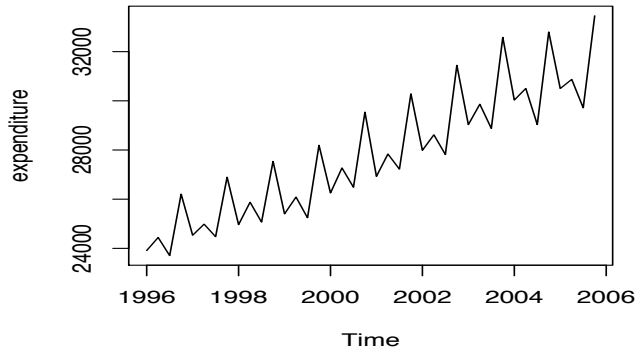

Fig. 2.1. Family food expenditure, quarterly data (1996Q1 to 2005Q4). Data available from `http://con.istat.it`

with a trend and a seasonal component. In fact, basic time series analysis lies on the possibility to find a reasonable regularity in the behavior of the phenomenon under study: forecasting the future behavior is clearly easier if the series tends to repeat a regular path over time.

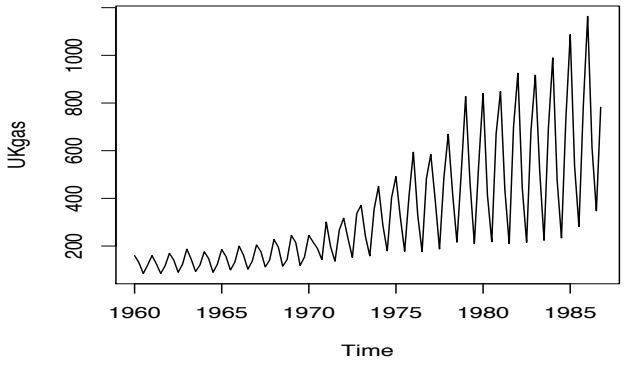

Fig. 2.2. Quarterly UK gas consumption from 1960Q1 to 1986Q4, in millions of therms.

But things get more complex for time series such as the ones plotted in figures 2.2-2.4. Figure 2.2 shows the quarterly UK gas consumption from 1960 to 1986 (data available in R). We clearly see a nonstationary variance. But

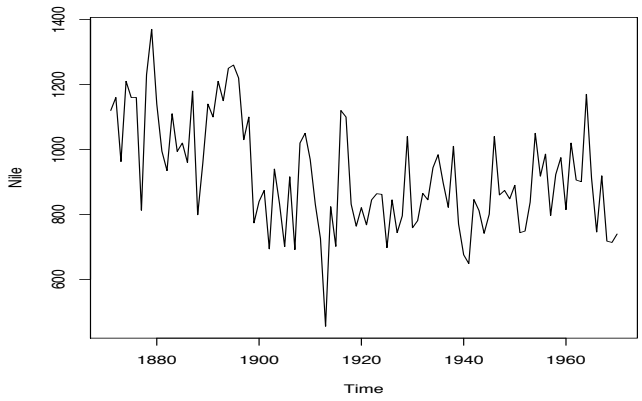

Fig. 2.3. Measurements of the annual flow of the river Nile at Ashwan 1871-1970.

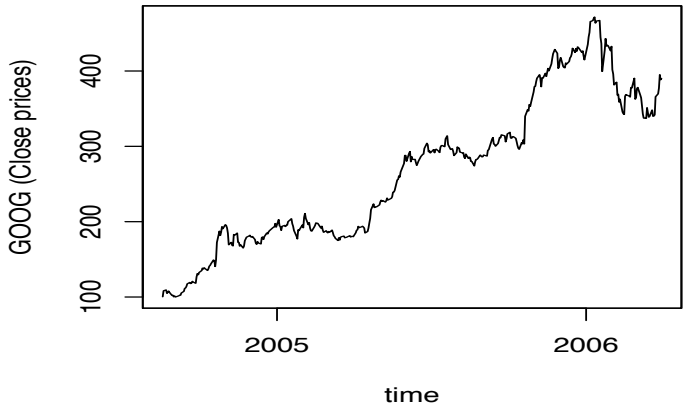

Fig. 2.4. Daily prices for Google Inc (GOOG)

still, an appropriate transformation of the data might succeed in stabilizing the variance and then we might use one of the familiar time-series models. Figure 2.3 shows a well-studied data set, the measurements of the annual flow of the river Nile at Ashwan from 1871 to 1970. The series shows level shifts. We know that the construction of the first dam of Ashwan started in 1898; the second big dam was completed in 1971: if you have ever seen these huge dams, you do understand the enormous changes that they caused on the Nile flow and in the vast surrounding area. Thus, we begin to feel the need for

more flexible time-series models, which do not assume a regular pattern and stability of the underlying system, but can include change points or structural breaks. Possibly more irregular is the series plotted in 2.4, showing daily prices of Google[1](close prices, 2004-8-19 to 2006-3-31). This series looks clearly non-stationary and in fact quite irregular: indeed, we know how unstable the market for the new economy has been! The analysis of non-stationary time series with ARMA models requires at least a preliminary transformation of the data to get stationarity; but we might feel more natural to have models which allow to analyze more directly data which show instability in the mean level and in the variance, structural breaks, sudden jumps. State-space models include ARMA models as special case, but as we shall see, they can be applied to nonstationary time series without requiring a preliminary transformation of the data. But there is a further basic issue. When dealing with economic or financial data, for example, a univariate time series model might appear quite limited. An economist might want to have a more wide comprehension of the economic system, looking for example at relevant macroeconomic variables which influence the variable of specific interest. For the financial example of figure 2.4, a univariate series model might be satisfying for high frequency data (the data in figure 2.4 are daily prices). But even a flexible univariate model (such as a stochastic volatility model, possibly with jumps, see chapter 5), might provide a quite good description of the behavior of the series, *adapting* to irregularities, structural breaks or jumps; but it will be hardly capable of *predicting* sudden changes without a further effort in a deeper and wider study of the economic, socio-political, real variables which have influence on the markets. Even then, forecasting sudden changes is clearly not at all an easy task! But we do feel that it is desirable to include regression terms in our model or use multivariate time series models. Again, including regression terms is quite natural in state space time series models. And state space models can in general be formulated for multivariate time series.

State space models originated in the engineering in the early sixties. In fact, the problem of forecasting has always been a fundamental and fascinating issue in the theory of stochastic processes and time series. Kolmogorov (1941) studied this problem for discrete time stationary stochastic processes, using a representation proposed by Wold (1938). Wiener (1949) studied continuous time stochastic processes, reducing the problem of forecasting to the solution of the so-called Wiener-Hopf integral equation. However, the methods for solving the Wiener problem were subject to several theoretical and practical limitations. A new look to the problem was given by Kalman (1960), using the Bode-Shannon representation of random processes and the "state transition" method of analsyis of dynamic systems. Kalman's solution, known as Kalman filter (Kalman (1960); Kalman and Bucy (1963)), applies to stationary and non-stationary random processes. These methods were immediately

[1] Financial data can be easily downloaded in R using the function `get.hist.quote` in package `tseries`, or the function `priceIts` in package `its`.

widely used by control engineers, but also in an extremely large range of applied contexts, from the determination of the orbits of the Voyager spacecraft to oceanographic problems, from agriculture to economics and speech recognition (see for instance the special issue of the IEEE Transactions on Automatic Control (1983) dedicated to applications of Kalman filter). However, the importance of these methods was recognized by statisticians only later, although the idea of latent variables and recursive estimation can be found in the statistical literature at least as early as Plackett (1950) and Thiele, see Lauritzen (1981). One reason for this delay is due to the fact that the work on Kalman filter was mostly published in the engineering literature. This means not only that the language of these works was not familiar to statisticians, but also that some problems which are crucial in applications in statistics and time series analysis were not sufficiently focussed yet. Kalman himself, in his 1960 paper, underlines that the problem of obtaining the transition model, which is crucial in practical applications, was treated as a separate question and not solved. In the engineering literature, it was common practice to assume the structure of the dynamic system as known, except for the effects of random disturbances, the main problem being to find an optimal estimate of the state of the system, given the model. In time series analysis, the emphasis is somehow different. The physical interpretation of the underlying states of the dynamic system is often less evident than in engineering applications. What we have is the observable process, and even if we can find convenient to think of it as the output of a dynamic system, the problem of forecasting is often the most relevant. In this context, the problem of model building can be more difficult, and even when a state-space representation is obtained, there are usually quantities or parameters in the model that are unknown.

State-space models appeared in the time series literature in the seventies (Akaike (1974a), Harrison and Stevens (1976)) and became established during the eighties (West and Harrison (1997), ...). In the last decades they have become a focus of interest. This is due on one hand to the development of models well suited to time series analysis, but also to an even wider range of applications, including for instance molecular biology or genetics, and on the other hand to the development of computational tools, such as modern Monte Carlo methods, for dealing with more complex nonlinear and non-Gaussian situations.

In the next sections we discuss the basic formulation of state-space models and the structure of the recursive computations for estimation. Then, as a special case, we present the Kalman filter for Gaussian linear dynamic models.

2.2 A simple example

Before presenting the general formulation of state space models, it is useful to give an intuition of the basic ideas and of the recursive computations through a simple, introductory example. Let's think of the problem of determining

the position θ of an object, based on some measurements $(Y_1, Y_2, \ldots)$ affected by random errors. This problem is fairly intuitive, and dynamics can be incorporated into it quite naturally: in the static problem, the object does not move over time, but it is natural to extend the discussion to the case of a moving target. If you prefer, you might think of some economic problem, such as forecasting the sales of a good; in short-term forecasting, the observed sales are often modeled as measurements of the unobservable average sales' level plus a random error; in turn, the average sales are supposed to be constant or randomly evolving over time (this is the so-called random walk plus noise model, see page 42).

The static problem. We have already discussed Bayesian inference in the static problem in chapter 1, page 5. There, you were lost at sea, on a small island, and θ was your unknown position (univariate: distance from the coast, say). The observations were modeled as

$$Y_t = \theta + \epsilon_t \ , \quad \epsilon_t \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \sigma^2),$$

that is, the Y_t's are conditionally i.i.d. $\sim \mathcal{N}(\theta, \sigma^2)$ given θ, with a conjugate Normal prior $\mathcal{N}(m_0, C_0)$ for θ. As we have seen in chapter 1, the posterior for θ is still Gaussian, with updated parameters given by (1.4) and (1.5), or by (1.6) and (1.7) if we compute them recursively, as new data become available.

To be concrete, let us suppose that your prior guess about the position θ is $m_0 = 1$, with variance $C_0 = 2$; the prior density is plotted in the first panel of figure 2.5. Note that m_0 is also your point forecast for the observation: $\mathrm{E}(Y_1) = \mathrm{E}(\theta + \epsilon_1) = \mathrm{E}(\theta) = m_0 = 1$.

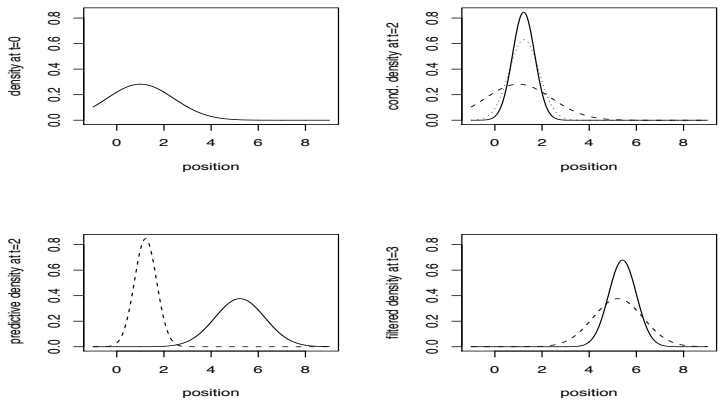

Fig. 2.5. Recursive updating of the density of θ_t

At time $t = 1$, we take a measurement $Y_1 = 1.3$; from (1.6) and (1.7), the parameters of the posterior Normal density of θ are

$$m_1 = m_0 + \frac{C_0}{C_0 + \sigma^2}(Y_1 - m_0) = 1.24,$$

with precision $C_1^{-1} = \sigma^{-2} + C_0^{-1} = 0.4^{-1}$. We see that m_1 is obtained as our best guess at time zero, m_0, corrected by the forecast error $(Y_1 - m_0)$, weighted by a factor $K_1 = C_0/(C_0 + \sigma^2)$. The more precise the observation is, or the more rough our initial information was, the more we "trust the data": in the above formula, the smaller σ^2 is with respect to C_0, the bigger is the weight K_1 of the data-correction term in m_1. When a new observation, $Y_2 = 1.2$ say, becomes available at time $t = 2$, we can compute the density of $\theta|(Y_1, Y_2)$, which is $\mathcal{N}(m_2, C_2)$, with $m_2 = 1.222$ and $C_2 = 0.222$, using again (1.6) and (1.7). The second panel in figure 2.5 shows the updating from the prior density to the posterior density of θ, given (Y_1, Y_2). We can proceed recursively in this manner as new data become available.

The dynamic problem. We solved the static problem. However, suppose we know that at time $t = 2$ the object starts to move, so that its position changes between two consecutive measurements. In this case we need to make a further assumptions on the dynamics. Let us assume a motion of a simple form, say

$$\theta_t = \theta_{t-1} + \nu + w_t, \quad w_t \sim \mathcal{N}(0, \sigma_w^2). \tag{2.1}$$

where ν is a known nominal speed and w_t is a Gaussian random error with mean zero and known variance σ_w^2. [2] Let, for example, $\nu = 4.5$ and $\sigma_w^2 = 0.9$. Now we have a process $(\theta_t, t = 1, 2, \ldots)$ which describes the unknown position of the target at successive time points. The observation equation is now

$$Y_t = \theta_t + \epsilon_t \ , \ \epsilon_t \text{ i.i.d.} \sim \mathcal{N}(0, \sigma^2), \tag{2.2}$$

and we assume that the sequences (θ_t) and (ϵ_t) are independent. For inference, we proceed along the following steps.

Initial step By the previous results, at time $t = 2$ we have

$$\theta_2|Y_1, Y_2 \sim \mathcal{N}(m_2 = 1.222, C_2 = 0.222).$$

[2] Equation (2.1) can be thought of as a discretization of a motion law in continuous time, such as

$$d\theta_t = \nu dt + dW_t$$

where ν is the nominal speed and dW_t is an error term. For simplicity, we consider a discretization in small intervals of time (t_{i-1}, t_i), as follows:

$$\frac{\theta_{t_i} - \theta_{t_{i-1}}}{t_i - t_{i-1}} = \nu + w_{t_i},$$

that is

$$\theta_{t_i} = \theta_{t_{i-1}} + \nu(t_i - t_{i-1}) + w_{t_i}(t_i - t_{i-1}),$$

where we assume that the random error w_{t_i} has density $\mathcal{N}(0, \sigma_w^2)$. With a further simplification, we take unitary time intervals, $(t_i - t_{i-1}) = 1$, so that the above expression is rewritten as the (2.1)

Prediction step. But now the position θ_t changes between two measurements. At time $t = 2$, we predict where the object will be at time $t = 3$, based on the dynamics (2.1). We easily find that

$$\theta_3|Y_1, Y_2 \sim \mathcal{N}(a_3, R_3),$$

with

$$a_3 = \mathrm{E}(\theta_2 + \nu + w_3|Y_1, Y_2) = m_2 + \nu = 5.722$$

and variance

$$R_3 = \mathrm{Var}(\theta_2 + \nu + w_3|Y_1, Y_2) = C_2 + \sigma_w^2 = 1.122.$$

The third plot in figure 2.5 illustrates the prediction step, from the conditional density of $\theta_2|Y_1, Y_2$ to the "predictive" density of $\theta_3|Y_1, Y_2$. Note that even if we were fairly confident about the position of the target at time $t = 2$, we become more uncertain about its position at time $t = 3$, for the effect of the random error w_t in the dynamics of θ_t: the larger σ_w^2 is, the more uncertain we are about the position at the time of the next measurement.
We can also predict the next observation Y_3 given (Y_1, Y_2). Based on the observation equation (2.2), we find easily that

$$Y_3|Y_1, Y_2 \sim \mathcal{N}(f_3, Q_3),$$

where

$$f_3 = \mathrm{E}(\theta_3 + \epsilon_3|Y_1, Y_2) = a_3 = 5.722$$

and

$$Q_3 = \mathrm{Var}(\theta_3 + \epsilon_3|Y_1, Y_2) = R_3 + \sigma^2 = 1.622.$$

The uncertainty about Y_3 depends on the measurement error (the term σ^2 in Q_3) and by the uncertainty about the position at time $t = 3$ (expressed by R_3).

Estimation step (filtering). At time $t = 3$, the new observation $Y_3 = 5$ becomes available. Our point forecast of Y_3 was $f_3 = a_3 = 5.722$, so we have a forecast error $e_t = Y_t - f_t = -0.722$. Intuitively, we have overestimated θ_3 and consequently Y_3; thus, our new estimate $\mathrm{E}(\theta_3|Y_1, Y_2, Y_3)$ of θ_3 will be smaller than $a_3 = \mathrm{E}(\theta_3|Y_1, Y_2)$. For computing the posterior density of $\theta_3|Y_1, Y_2, Y_3$, we use the Bayes formula, where the role of the prior is played by the density $\mathcal{N}(a_3, R_3)$ of θ_3 given (Y_1, Y_2), and the likelihood is the density of Y_3 given (θ_3, Y_1, Y_2). Note that (2.2) implies that Y_3 is independent form the past observations given θ_3 (assuming independence among the error sequences) with

$$Y_3|\theta_3 \sim \mathcal{N}(\theta_3, \sigma^2).$$

Thus, by Bayes formula (see (1.6) and (1.7)), we obtain

$$\theta_3|Y_1, Y_2, Y_3 \sim \mathcal{N}(m_3, C_3),$$

where

$$m_3 = a_3 + \frac{R_3}{R_3 + \sigma^2}(Y_3 - f_3) = 5.568$$

and

$$C_3 = \frac{\sigma^2 R_3}{\sigma^2 + R_3} = R_3 - \frac{R_3}{R_3 + \sigma^2} R_3 = 0.346.$$

We see again the estimation-correction structure of the updating mechanism. Our best estimate of θ_3 given the data (Y_1, Y_2, Y_3) is computed as our previous best estimate a_3, corrected by the forecast error $e_3 = (Y_3 - f_3)$, which has weight $K_3 = R_3/(R_3 + \sigma^2)$. This weight is bigger the more uncertain we are about our forecast a_3 of θ_3 (that is, the larger R_3 is, which in turn depends on C_2 and σ_w^2) and the more precise the observation Y_3 is (i.e., the smaller σ^2 is). From these results we see that a crucial role in determining the effect of the data on estimation and forecasting is played by the relative magnitude of the observation variance σ^2 and of the system variance σ_w^2. The last plot in figure 2.5 illustrates this estimation step.
We can proceed repeating recursively the previous steps for updating our estimates and forecasts as new observations become available.

This simple example illustrates the basic aspects of dynamic linear models:
- the observable process $(Y_t; t = 1, 2, \ldots)$ is thought of as determined by a latent process $(\theta_t; t = 1, 2, \ldots)$, up to Gaussian random errors. If we knew the position of the object at successive time points, the Y_t's would be independent: what remain are only unpredictable measurement errors. Furthermore, the observation Y_t depends only on the position θ_t of the target at time t;
- the latent process (θ_t) has a fairly simple dynamics: θ_t does not depend on the entire past trajectory but only on the previous position θ_{t-1}, through a linear relationship, up to Gaussian random errors;
- estimation and forecasting can be obtained sequentially, as new data become available. The example illustrates the role played by the modeling assumptions (in particular by the observational variance and system variance) in the updating mechanism.

The assumption of linearity and Gaussianity is specific of DLMs, but the dependence structure of the process (Y_t) is what we assume in general state space models.

2.3 State space models

Consider a time series $(Y_t, t = 1, 2, \ldots)$, where Y_t is an observable $(m \times 1)$ random vector; for example, $Y_t = (Y_{1,t}, \ldots, Y_{m,t})'$ are the prices of m bonds in a portfolio at time t. For making inference on the time series, in particular for predicting the next value Y_{t+1} given the observations $(Y_1, \ldots, Y_t)$, we need

to specify the probability law of the process (Y_t), which means giving the dependence structure among the Y_t's variables.

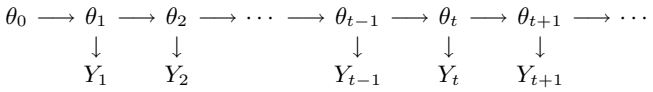

Fig. 2.6. Dependence structure for a state space model

State-space models are based on the idea that the time series (Y_t) is an incomplete and noisy function of some underlying unobservable process $(\theta_t, t = 1, 2, \ldots)$, called the *state* process. In engineering applications, θ_t usually describes the state of a physical system which produces the output Y_t, up to random disturbances. More generally, we might think of (θ_t) as an auxiliary random process which facilitates the task of specifying the probability law of the time series: the observable process (Y_t) depends on the latent state process (θ_t), which has a simpler, Markovian dynamics, and we can reasonably assume that the observation Y_t only depends on the state of the system at the time the measurement is taken, θ_t. Figure 2.6 represents graphically the dependences among variables that we are assuming.

More formally, the assumptions of a state space model are

A.1 $(\theta_t, t = 0, 1, \ldots)$ is a Markov chain; that is, θ_t depends on the past values $(\theta_0, \theta_1, \ldots, \theta_{t-1})$ only through θ_{t-1}. Thus, the probability law of the process $(\theta_t, t = 0, 1, \ldots)$ is specified by assigning the initial density [3] $p_0(\theta_0)$ of θ_0 and the transition densities $p(\theta_t|\theta_{t-1})$ of θ_t conditionally on θ_{t-1}.

A.2 Conditionally on $(\theta_t, t = 0, 1, \ldots)$, the Y_t's are independent and Y_t depends on θ_t only. It follows that, for any $n \geq 1$, $(Y_1, \ldots, Y_n)|\theta_1, \ldots, \theta_n$ have joint conditional density $\prod_{t=1}^{n} f(y_t|\theta_t)$.

The term *state-space model* is used when the state variables are continuous. When they are discrete, one usually calls this model a *hidden Markov model*. The assumptions (A1)-(A2) and the specification of the relevant densities allow to write the probability law of the joint random process $((\theta_t, Y_t), t = 1, 2, \ldots)$, from which we can deduce all the dependences among the variables. The graph in Figure 2.6 may be used to deduce useful conditional independence properties of the random variables occurring in a state space model. In fact, two sets of random variables, A and B, can be shown to be conditionally

[3] As in chapter 1, we mean a density w.r.t. some dominating measure. It can be a probability density function (density w.r.t. Lebesgue measure) or a probability mass function (density w.r.t. the counting measure). Also, in general we shall use the sloppy but convenient notation $p(u|v)$ for the conditional density of a random vector U given another random vector $V = v$. Here we assume that the relevant densities exist.

independent given a third set of variables, C, if and only if C separates A and B, i.e., if any path connecting one variable in A to one in B passes through C. Note that in the previous statement the arrows in Figure 2.6 have to be considered as undirected edges of the graph that can be transversed in both directions. For a proof, see Cowell et al. (1999, Section 5.3). With the help of the graph for understanding the conditional independence relations implied by the model, we find that, for any $n \geq 1$,

$$\begin{aligned}
&(\theta_0, \theta_1, \ldots, \theta_n, Y_1, \ldots, Y_n) \sim p_0(\theta_0) \prod_{t=1}^{n} p(\theta_t, Y_t | \theta_0, \theta_1, \ldots, \theta_{t-1}, Y_1, \ldots, Y_{t-1}) \\
&= p_0(\theta_0) \prod_{t=1}^{n} f(Y_t | \theta_0, \ldots, \theta_t, Y_1, \ldots, Y_{t-1})\, p(\theta_t | \theta_0, \ldots, \theta_{t-1}, Y_1, \ldots, Y_{t-1}) \\
&= p_0(\theta_0) \prod_{t=1}^{n} f(Y_t | \theta_t) p(\theta_t | \theta_{t-1})
\end{aligned} \tag{2.3}$$

In particular, we see that the process $((\theta_t, Y_t), t = 1, 2, \ldots)$ is Markovian. The density of $(Y_1, \ldots, Y_n)$ can be obtained by integrating out all the θ-variables from the joint density (2.3). As we shall see, computations are fairly simple in Gaussian linear state space models; however, in general the density of $(Y_1, \ldots, Y_n)$ is not available in close form and the observable process (Y_t) is not Markovian. However, we can see an important property: Y_t is conditionally independent from the past observations $(Y_1, \ldots, Y_{t-1})$ given the value of θ_t. This gives us an appealing interpretation of the *state* θ_t: it represents some quantitative information which summarizes the past history of the observable process and suffices for predicting its future behavior.

2.3.1 Dynamic linear models.

The first, important class of state-space models is given by Gaussian linear state-space models, also called *dynamic linear models* (DLM). These models are specified by means of two equations

$$\begin{aligned}
Y_t &= F_t \theta_t + v_t, \quad v_t \sim N_m(0, V_t), \\
\theta_t &= G_t \theta_{t-1} + w_t, \; w_t \sim N_p(0, W_t),
\end{aligned} \tag{2.4}$$

where G_t and F_t are known matrices and the (v_t) and (w_t) are two independent white noise sequences (i.e., they are independent, both between them and within each of them), with mean zero and known covariance matrices V_t and W_t respectively. The first equation above is called the *observation equation*, the second *state equation* or *system equation*. Furthermore, it is assumed that θ_0 has a Gaussian distribution,

$$\theta_0 \sim N_p(m_0, C_0), \tag{2.5}$$

for some non-random vector m_0 and matrix C_0, and it is independent on (v_t) and (w_t). One can show that the DLM satisfies the assumptions (A.1) and (A.2) of the previous section, with $Y_t|\theta_t \sim \mathcal{N}(F_t\theta_t, V_t)$ and $\theta_t|\theta_{t-1} \sim \mathcal{N}(G_t\theta_{t-1}, W_t)$ (see problems 2.1 and 2.2).

In contrast to (2.4), the general state space model can be written in the form

$$\begin{aligned} Y_t &= h_t(\theta_t, v_t) \\ \theta_t &= g_t(\theta_{t-1}, w_t) \end{aligned}$$

with arbitrary functions g_t and h_t. It is thus more flexible. *Linear* state space models specify g_t and h_t as linear functions, and *Gaussian* linear models add the assumptions of Gaussian distributions. The assumption of Normality is sensible in many applications, and it can be justified by central limit theorem arguments. However, there are many important extensions, such as heavy tailed errors for modeling outliers, or generalized DLM for treating discrete time series. The price to be paid when removing the assumption of Normality are additional computational difficulties. We will briefly mention some extensions in the following section and in chapter 5.

Example 1. Simple DLM for time series analysis

We introduce here some examples of DLM for time series analysis, which will be treated more extensively in chapter 3. The simplest model for a univariate time series $(Y_t, t = 1, 2, \ldots)$ is the so-called *random walk plus noise* model, defined by

$$\begin{aligned} Y_t &= \mu_t + v_t, & v_t &\sim N(0, V) \\ \mu_t &= \mu_{t-1} + w_t, & w_t &\sim N(0, W), \end{aligned} \tag{2.6}$$

where the error sequences (v_t) and (w_t) are independent, both within them and between them. This is a DLM with $m = p = 1$, $\theta_t = \mu_t$ and $F_t = G_t = 1$. It is the model used in the introductory example in section 2.2 of chapter 2, when there is no speed in the dynamics ($\nu = 0$ in the state equation (2.1)). Intuitively, it is appropriate for time series showing no clear trend or seasonal variation: the observations (Y_t) are modeled as random fluctuations around a level (μ_t); in turn, the level can evolve randomly over time (described by a random walk). This is why the model is also called *local level* model; if $W = 0$, we are back to the constant mean model. Note that Y_t is modeled as a noisy observation of the random walk μ_t, which is non-stationary. Indeed, DLM can be used for modeling non-stationary time series. On the contrary, the usual ARMA models require a preliminary transformation of the data for getting stationarity.

A slightly more elaborated model is the *linear growth*, or local linear trend model, which has the same observation equation as the local level model, but includes a time-varying slope in the dynamics for μ_t

$$\begin{aligned} Y_t &= \mu_t + v_t, & v_t &\sim N(0, V) \\ \mu_t &= \mu_{t-1} + \beta_{t-1} + w_{1,t}, & w_{1,t} &\sim N(0, \sigma^2_{w_1}) \\ \beta_t &= \beta_{t-1} + w_{2,t}, & w_{2,t} &\sim N(0, \sigma^2_{w_2}), \end{aligned} \tag{2.7}$$

with uncorrelated errors. This is a DLM with

$$\theta_t = \begin{pmatrix} \mu_t \\ \beta_t \end{pmatrix}, \quad G = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}, \quad W = \begin{pmatrix} \sigma^2_{w_1} & 0 \\ 0 & \sigma^2_{w_2} \end{pmatrix}, \quad F = \begin{pmatrix} 1 & 0 \end{pmatrix}.$$

The system variances $\sigma^2_{w_i}$ are allowed to be zero. We have used this model in the introductory example of section 2.2; there, we had a constant nominal speed in the dynamics, that is $\sigma^2_{w_2} = 0$.

Note that in these examples the matrices G_t and F_t and the covariance matrices V_t and W_t are constant; in this case the model is said *time invariant*. We will see other examples in chapter 3. In particular, we shall see that the popular Gaussian ARMA models can be obtained as particular cases of DLM; in fact, it can be shown that Gaussian ARMA and DLM models are equivalent in the time-invariant case (see Hannan and Deistler; 1988).

Example 2. Simple dynamic regression

DLM can be regarded as a generalization of the linear regression model, allowing for time varying regression coefficients. The simple, static linear regression model describes the relationship between a variable y and a nonrandom explanatory variable x as

$$Y_t = \theta_1 + \theta_2 x_t + \epsilon_t \ , \quad \epsilon_t \text{ i.i.d} \sim \mathcal{N}(0, \sigma^2);$$

here we think of $(Y_t, x_t), t = 1, 2, \ldots$ as observed through time. Allowing for time varying regression parameters, one can model non-linearity of the functional relationship between x and y, structural changes in the process under study, omission of some variables. A simple dynamic linear regression model assumes

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

with a further equation for describing the system evolution

$$\theta_t = G_t \theta_{t-1} + w_t, \quad w_t \sim \mathcal{N}_2(0, W_t).$$

This is a DLM with $F_t = (1, x_t)$ and states $\theta_t = (\theta_{1,t}, \theta_{2,t})'$. If $G_t = I$, the identity matrix, $\sigma^2_t = \sigma^2$ and $w_t = 0$ for any t, we are back to the simple static linear regression model.

Creating DLM in R

DLM are represented in package `dlm` as lists with a class attribute, which makes them into objects of class `"dlm"`. The required components of a dlm object are `m0`, `C0`, `FF`, `V`, `GG`, `W`, which correspond to the vector/matrices $m_0, C_0, F_t, V_t, G_t, W_t$ in (2.5) and (2.4), assuming that F_t, V_t, G_t, W_t do not vary with t. For such a constant DLM, this information is enough to completely specify the model. The `dlm` function creates a dlm object from its components, performing some sanity checks on the input, such as testing the dimensions of the matrices for consistency. The input may be given as a list with named arguments or as individual arguments. Here is how to use `dlm` to create dlm objects corresponding to the random walk plus noise model and to the linear growth model introduced on page 42. We assume that $V = 1.4$ and $\sigma^2 = 0.2$.

R code

```
> rw <- dlm(m0 = 0, C0 = 10, FF = 1, V = 1.4, GG = 1, W = 0.2)
> unlist(rw)
  m0   C0   FF    V   GG    W
 0.0 10.0  1.0  1.4  1.0  0.2
> lg <- dlm(m0 = rep(0,2), C0 = 10 * diag(2), FF = matrix(c(1,0),nr=1),
+           V = 1.4, GG = matrix(c(1,0,1,1),nr=2), W = diag(c(0,0.2)))
> lg
$FF
     [,1] [,2]
[1,]    1    0

$V
     [,1]
[1,]  1.4

$GG
     [,1] [,2]
[1,]    1    1
[2,]    0    1

    ....

> is.dlm(lg)
[1] TRUE
```

Let us turn now on time-varying DLM and how they are represented in R. Most often, in a time-invariant DLM, only a few entries (possibly none) of each matrix will change over time, while the remaining will be constant. Therefore,

instead of storing the entire matrices F_t, V_t, G_t, W_t for all values of t that one wish to consider, we opted to store a template of each of them, and save the time-varying entries in a separate matrix. This is defined as argument `X`. Taking this approach, one also needs to know to which entry of which matrix each column of `X` corresponds. To this aim one has to specify the arguments `JFF`, `JV`, `JGG`, and `JW`. Let us focus on the first one, `JFF`. This should be a matrix of the same dimension of `FF`, with integer entries: if `JFF[i,j]` is `k`, a positive integer, that means that the value of `FF[i,j]` at time `s` will be set to `X[s,k]`. If `JFF[i,j]` is zero then `FF[i,j]` is taken to be constant in time. `JV`, `JGG`, and `JW` are used in the same way, for `V`, `GG`, and `W`, respectively. Consider, for example, the dynamic regression model introduced on page 43. The only time-varying element is the $(1, 2)$-entry of F_t, therefore `X` will be a one-column matrix (although `X` is allowed to have extra, unused, columns). The following code shows how a dynamic regression model can be defined in R.

R code

```
> x <- rnorm(100) # covariates
> dlr <- dlm(m0 = rep(0,2), C0 = 10 * diag(2), FF = matrix(c(1,0),nr=1),
+            V = 1.3, GG = diag(2), W = diag(c(0.4,0.2)),
+            JFF = matrix(c(0,1),nr=1), X = as.matrix(x))
> dlr
$FF
     [,1] [,2]
[1,]    1    0

$V
     [,1]
[1,]  1.3

$GG
     [,1] [,2]
[1,]    1    0
[2,]    0    1

$W
     [,1] [,2]
[1,]  0.4  0.0
[2,]  0.0  0.2

$JFF
     [,1] [,2]
[1,]    0    1

$X
     [,1]
```

```
30 [1,] -0.5865
   [2,] 0.2031
32 [3,] ...

34 $m0
   [1] 0 0
36
   $C0
38      [,1] [,2]
   [1,]   10    0
40 [2,]    0   10
```

Note that the dots on line 32 of the display above were produced by the `print` method function for objects of class `"dlm"`. If you want the entire `X` component to be printed, you need to use `print.default` or extract it as `dlr$X`.

We have illustrated the usage of the function `dlm` to create a general (constant or time-varying) object of class `"dlm"`. As a matter of fact, the dynamic regression model, as well as the random walk plus noise and the linear growth model, are so common that simplified functions to create them are provided, as we will see in Chapter 3.

2.3.2 Examples of non-linear and non-Gaussian state space models

Specification and estimation of DLM for time series analysis will be treated in chapters 3 and 4. In chapter 5 we will discuss some important classes of non-linear and non-Gaussian state space models, which are briefly introduced here.

Exponential family state space models

Dynamic linear models can be generalized by removing the assumption of Gaussian distributions. This generalization is required for modeling discrete time series; for example, if Y_t represents the presence/absence of a characteristic in the problem under study over time, we would use a Bernoulli distribution; if Y_t are counts, we might use a Poisson model, etc. *Generalized DLM* assume that the conditional distribution $f(Y_t|\theta_t)$ of Y_t given θ_t is a member of the exponential family, with natural parameter $\eta_t = F_t\theta_t$. The state equation is as for Gaussian linear models, $\theta_t = G_t\theta_{t-1} + w_t$. As we shall see in chapter 5, generalized linear models arise computational difficulties, which can however be solved by MCMC techniques.

Hidden Markov models

State space models where the state θ_t is discrete are usually referred as *hidden Markov models*. Hidden Markov models are used extensively in speech recognition (see e.g. Rabiner and Juang (1993)); in economics and finance, they are

often used for modeling a time series with structural breaks. The dynamics of the series and the change points are thought as determined by a latent Markov chain (θ_t), with state space $\{\theta_1^*, \ldots, \theta_k^*\}$ and transition probabilities

$$p(i|j) = P(\theta_t = \theta_i^* | \theta_{t-1} = \theta_j^*).$$

Consequently, Y_t can be in different *regimes* depending on the state of the chain at time t, in the sense that

$$Y_t|\theta_t = \theta_j^* \sim f(Y_t|\theta_j^*)\,, j = 1, \ldots, k.$$

Although state-space models and hidden Markov models have evolved as separate subjects, it is worth noting that the basic assumptions and recursive computations are closely related. MCMC methods for hidden Markov models have been developed, see Rydén and Titterington (1998), Cappé et al. (2005) and the references therein.

Stochastic volatility models

Stochastic volatility models are widely used in financial applications. Let Y_t be the log-return of an asset at time t (i.e., $Y_t = \log P_t/P_{t-1}$, where P_t is the asset price at time t). Under the assumption of efficient markets, the log-returns have null conditional mean: $\mathrm{E}(Y_{t+1}|Y_1, \ldots, Y_t) = 0$. However, the conditional variance, called volatility, varies over time. There are two main classes of models for analyzing volatility of returns. The popular ARCH and GARCH models describe the volatility as a function of the past values of the returns (see .....). Stochastic volatility models, instead, consider the volatility as an exogenous random process. This leads to a state space model where the volatility is (part of) the state vector; see e.g. Shephard (1996). The simplest stochastic volatility model has the following form

$$\begin{aligned} Y_t &= \exp\{\frac{1}{2}\theta_t\} w_t \\ \theta_t &= \eta + \phi\theta_{t-1} + v_t \end{aligned}$$

that is, θ_t is an autoregressive model of order one. These models are non-linear and non-Gaussian and computations are usually more demanding than for ARCH and GARCH models; however, MCMC approximations are available (Jacquier et al. (1994)). On the other hand, stochastic volatility models seem easier to generalize to the case of returns of a collection of assets, while for multivariate ARCH and GARCH models the number of parameters becomes too large. Let $Y_t = (Y_{1,t}, \ldots, Y_{m,t})$ be the log-returns for m assets. A simple multivariate stochastic volatility model might assume that

$$Y_{i,t} = \exp\{z_t + x_{i,t}\} v_{i,t},\ i = 1, \ldots, m,$$

where z_t describes a common market volatility factor and the $x_{i,t}$'s are individual volatilities. The state vector is $\theta_t = (z_t, x_{1,t}, \ldots, x_{m,t})'$ and a simple

state equation might assume that the components of θ_t are independent AR(1) processes.

We will discuss ARCH and stochastic volatility models in Chapter 5.

2.4 State estimation and forecasting

The great flexibility of state-space models is one reason for their extensive application in an enormous range of applied problems. Of course, as in any statistical application, a crucial and often difficult step is a careful model specification. In many problems, the statistician and the experts together can build a state-space model where the states have an intuitive meaning, and expert knowledge can be used for specifying the transition probabilities in the state equation, determine the dimension of the state-space, etc. However, often the model building can be a major difficulty: there might be no clear identification of physically interpretable states, or the state-space representation could be non-unique, or the state-space is too big and poorly identifiable, or the model is too complicated. We will discuss some issues about model building for time series analysis with DLM in chapter 3. Here, to get started, we consider the model as given, that is we assume that the densities $f(y_t|\theta_t)$ and $p(\theta_t|\theta_{t-1})$ have been specified, and we present the basic recursions for estimation and forecasting. In chapter 4, we will let these densities depend on unknown parameters ψ and we will discuss their estimation.

For a given state-space model, the main tasks are to make inference on the unobserved states or predict future observations based on a part of the observation sequence. Estimation and forecasting are solved by computing the conditional distributions of the quantities of interest, given the available information.

For estimating the state vector we compute the conditional densities $p(\theta_s|y_1, \ldots, y_t)$. We distinguish between problems of *filtering* (when $s = t$), *state prediction* ($s > t$) and *smoothing* ($s < t$). It is worth to underline the difference between filtering and smoothing. In the filtering problem, the data are supposed to arrive sequentially in time. This is the case in many applied problems: think for example of the problem of tracking a moving object, or of financial applications where one has to estimate, day by day, the term structure of interest rates, updating the current estimates as new data are observed on the markets the following day, etc. In these cases, we want a procedure for estimating the current value of the state vector, based on the observations up to time t ("now"), and for updating our estimates and forecasts as new data become available at time $t + 1$. For solving the filtering problem, we compute the conditional density $p(\theta_t|y_1, \ldots, y_t)$. In DLM, the Kalman filter provides the formulas for updating our current inference on the state vector

as new data become available, that is for passing from the filtering density $p(\theta_t|y_1,\ldots,y_t)$ to $p(\theta_{t+1}|y_1,\ldots,y_{t+1})$.

The problem of smoothing, or retrospective analysis, consists instead in estimating the state sequence at times $1,\ldots,t$, given the data $y_1,\ldots,y_t$. In many applications, one has observations on a time series for a certain period, and wants to retrospectively study the behavior of the system underlying the observations; for example, in economic studies, the researcher might have the time series of consumption, or of the gross domestic product of a country, for a certain number of years, and she might be interested in retrospectively understanding the socio-economic behavior of the system. The smoothing problem is solved by computing the conditional distribution of $\theta_1,\ldots,\theta_t$ given $Y_1,\ldots,Y_t$; again, this can be done by a recursive algorithm.

In fact, in time series analysis forecasting is often the main task; the state estimation is then just a step for predicting the value of future observations. For one-step-ahead forecasting, that is predicting the next observation Y_{t+1} based on the data $y_1,\ldots,y_t$, one first estimates the next value θ_{t+1} of the state vector, and then, based on this estimates, one computes the forecast for Y_{t+1}. The one-step-ahead state predictive density is $p(\theta_{t+1}|y_1,\ldots,y_t)$ and, as we shall see, it is based on the filtering density of θ_t. From this, one obtains the one-step-ahead predictive density $f(y_{t+1}|y_1,\ldots,y_t)$.

One might be interested in looking a bit further ahead, estimating the evolution of the system, i.e. the state vector θ_{t+k} for some $k \geq 1$, and making k-steps-ahead forecasts for Y_{t+k}. The state-prediction is solved by computing the k-steps-ahead state predictive density $p(\theta_{t+k}|y_1,\ldots,y_t)$; based on this density, one can compute the k-steps-ahead predictive density $f(y_{t+k}|y_1,\ldots,y_t)$ for the future observation at time $t+k$. Of course, forecasts become more and more uncertain as the time horizon $t+k$ gets far away in the future (think of weather forecasts!); but note that we can anyway quantify the uncertainty through a probability density, namely the predictive density of Y_{t+1} given $(Y_1,\ldots,Y_t)$. We will show how to compute the predictive densities in a recursive fashion. In particular, the conditional mean $\mathrm{E}(Y_{t+1} \mid Y_1,\ldots,Y_t)$ provides an optimal one-step-ahead point forecast of the value of Y_{t+1}, minimizing the conditional expected square prediction error. As a function of k, $\mathrm{E}(Y_{t+k} \mid Y_1,\ldots,Y_t)$ is usually called the *forecast function*.

2.4.1 Filtering

We first describe the recursive steps for computing the filtering densities $p(\theta_t|Y_1,\ldots,Y_t)$ in general state space models. Even if we will not make extensive use of these formulae until chapter 5, it is useful to look now at the general recursions for better understanding the role of the conditional independence assumptions that have been introduced. Then we move to the case of DLM for which the filtering problem is solved by the well-known Kalman filter.

Let us denote with $\mathcal{D}_t$ the information provided by the first t observations, $Y_1, \ldots, Y_t$. One of the advantages of state space models is that, due to the Markovian structure of the state dynamics (A.1) and the assumptions on conditional independence for the observables (A.2), the filtered and predictive densities can be computed by a *recursive algorithm*. As we have seen in the introductory example of section 2.2, starting from $\theta_0 \sim p_0(\theta_0) = p(\theta_0|\mathcal{D}_0)$ one can recursively compute, for $t = 1, 2, \ldots$:

(i) the one-step-ahead predictive density for θ_t given $\mathcal{D}_{t-1}$, based on the filtering density $p(\theta_{t-1}|\mathcal{D}_{t-1})$ and the transition model;

(ii) the one-step-ahead predictive density for the next observation;

(iii) the filtering density $p(\theta_t|\mathcal{D}_t)$ using Bayes rule with $p(\theta_t|\mathcal{D}_{t-1})$ as the prior density and the likelihood $f(y_t|\theta_t)$.

More formally, the filtering recursions are as follows.

Proposition 2.1. *(Filtering recursions).*

(i) The one-step-ahead predictive density for the states can be computed from the filtered density $p(\theta_{t-1}|\mathcal{D}_{t-1})$ according to

$$p(\theta_t|\mathcal{D}_{t-1}) = \int p(\theta_t|\theta_{t-1})p(\theta_{t-1}|\mathcal{D}_{t-1})\mathrm{d}\nu(\theta_{t-1}).$$

(ii) The one-step-ahead predictive density for the observations can be computed from the predictive density for the states as

$$f(y_t|\mathcal{D}_{t-1}) = \int f(y_t|\theta_t)p(\theta_t|\mathcal{D}_{t-1})\mathrm{d}\nu(\theta_t).$$

(iii)The filtering density can be computed from the above densities as

$$p(\theta_t|\mathcal{D}_t) = \frac{f(y_t|\theta_t)p(\theta_t|\mathcal{D}_{t-1})}{f(y_t|\mathcal{D}_{t-1})}.$$

Proof. What is interesting to understand is the role played by the assumptions of conditional independence. The graph in figure 2.6 can help again.

(i) Note that $\theta_{t+1} \amalg (Y_1, \ldots, Y_t)|\theta_t$. Therefore

$$\begin{aligned} p(\theta_t|\mathcal{D}_{t-1}) &= \int p(\theta_{t-1}, \theta_t|\mathcal{D}_{t-1})\mathrm{d}\nu(\theta_{t-1}) = \int p(\theta_t|\theta_{t-1}, \mathcal{D}_{t-1})p(\theta_{t-1}|\mathcal{D}_{t-1})\mathrm{d}\nu(\theta_{t-1}) \\ &= \int p(\theta_t|\theta_{t-1})p(\theta_{t-1}|\mathcal{D}_{t-1})\mathrm{d}\nu(\theta_{t-1}). \end{aligned}$$

(ii) From the conditional independence $Y_t \amalg (Y_1, \ldots, Y_{t-1})|\theta_t$, we have

$$\begin{aligned} f(y_t|\mathcal{D}_{t-1}) &= \int f(y_t, \theta_t|\mathcal{D}_{t-1})\mathrm{d}\nu(\theta_t) = \int f(y_t|\theta_t, \mathcal{D}_{t-1})p(\theta_t|\mathcal{D}_{t-1})\mathrm{d}\nu(\theta_t) \\ &= \int f(y_t|\theta_t)p(\theta_t|\mathcal{D}_{t-1})\mathrm{d}\nu(\theta_t). \end{aligned}$$

(iii) Using the Bayes rule:

$$p(\theta_t|\mathcal{D}_t) = \frac{p(\theta_t|\mathcal{D}_{t-1})\, f(y_t|\theta_t, \mathcal{D}_{t-1})}{f(y_t|\mathcal{D}_{t-1})} = \frac{p(\theta_t|\mathcal{D}_{t-1})\, f(y_t|\theta_t)}{f(y_t|\mathcal{D}_{t-1})},$$

by the conditional independence $Y_t \amalg (Y_1, \ldots, Y_{t-1})|\theta_t$. □

The above results can be used for recursively computing the k-steps ahead predictive densities, starting from $k = 1$:

$$p(\theta_{t+k}|\mathcal{D}_t) = \int p(\theta_{t+k}|\theta_{t+k-1})\, p(\theta_{t+k-1}|\mathcal{D}_t) \mathrm{d}\nu(\theta_{t+k-1})$$

and

$$f(y_{t+k}|\mathcal{D}_t) = \int f(y_{t+k}|\theta_{t+k})\, p(\theta_{t+k}|\mathcal{D}_t) \mathrm{d}\nu(\theta_{t+k}).$$

Note that $p(\theta_{t+k}|\mathcal{D}_t)$ summarizes the information contained in the past observation $\mathcal{D}_t$ which is sufficient for predicting Y_{t+k}.

2.4.2 The Kalman filter for DLM

The previous results solve in principle the filtering and the forecasting problems, however in general the actual computation of the relevant conditional densities is not at all an easy task. Dynamic linear models are one relevant case where the general recursions simplify considerably. In this case, using standard results about the multivariate Gaussian distribution, it is easily proved that the random vector $(\theta_0, \theta_1, \ldots, \theta_t, Y_1, \ldots, Y_t)$ has a Gaussian distribution for any $t \geq 1$. It follows that the marginal and conditional distributions are also Gaussian. Since all the relevant distributions are Gaussian, it suffices to compute their means and covariances. The solution of the filtering problem is given by the famous Kalman filter.

Theorem 2.2 (Kalman filter). *For the DLM (2.4), if*

$$\theta_{t-1}|\mathcal{D}_{t-1} \sim \mathcal{N}(m_{t-1}, C_{t-1}),$$

where $t \geq 1$, then

(i) the one-step-ahead state predictive density of θ_t, given $\mathcal{D}_{t-1}$ is Gaussian, with parameters

$$a_t = \mathrm{E}(\theta_t|\mathcal{D}_{t-1}) = G_t m_{t-1}$$
$$R_t = \mathrm{Var}(\theta_t|\mathcal{D}_{t-1}) = G_t C_{t-1} G_t' + W_t;$$

(b) the one-step-ahead predictive density of Y_t given $\mathcal{D}_{t-1}$ is Gaussian, with parameters

$$f_t = \mathrm{E}(Y_t|\mathcal{D}_{t-1}) = F_t a_t$$
$$Q_t = \mathrm{Var}(Y_t|\mathcal{D}_{t-1}) = F_t R_t F_t' + V_t;$$

(c) the filtering density of θ_t given $\mathcal{D}_t$ is Gaussian, with

$$m_t = \mathrm{E}(\theta_t|\mathcal{D}_t) = a_t + R_t F_t' Q_t^{-1} e_t$$
$$C_t = \mathrm{Var}(\theta_t|\mathcal{D}_t) = R_t - R_t F_t' Q_t^{-1} F_t R_t,$$

where $e_t = Y_t - f_t$ is the forecast error.

Proof. The random vector $(\theta_0, \theta_1, \ldots, \theta_t, Y_1, \ldots, Y_t)$ has joint density given by (2.3), where the marginal and conditional densities involved are Gaussian. From standard results on the multivariate Normal distribution (see property (2) reported in the appendix of chapter 1, page 27), it follows that the joint density of $(\theta_0, \theta_1, \ldots, \theta_t, Y_1, \ldots, Y_t)$ is Gaussian, for any $t \geq 1$. Consequently, the distribution of any subvector is also Gaussian, as is the conditional distribution of some components given some other components. Therefore the predictive densities and the filtering densities are Gaussian, and it suffices to compute their means and variances. If $\theta_{t-1} \sim \mathcal{N}(m_{t-1}, C_{t-1})$, then we have the following results.

(i) From the state equation, $\theta_t|\mathcal{D}_{t-1} \sim \mathcal{N}(a_t, R_t)$, with [4]

$$a_t = \mathrm{E}(\theta_t|\mathcal{D}_{t-1}) = \mathrm{E}(\mathrm{E}(\theta_t|\theta_{t-1}, \mathcal{D}_{t-1})|\mathcal{D}_{t-1}) = \mathrm{E}(G_t\theta_{t-1}|\mathcal{D}_t) = G_t m_{t-1};$$

$$R_t = \mathrm{Var}(\theta_t|\mathcal{D}_{t-1}) = \mathrm{E}(\mathrm{Var}(\theta_t|\theta_{t-1}, \mathcal{D}_{t-1})|\mathcal{D}_{t-1}) + \mathrm{Var}(\mathrm{E}(\theta_t|\theta_{t-1}, \mathcal{D}_{t-1})|\mathcal{D}_{t-1})$$
$$= W_t + G_t C_{t-1} G_t'.$$

(ii) From the observation equation, $Y_t|\mathcal{D}_{t-1} \sim \mathcal{N}(f_t, Q_t)$ with

$$f_t = \mathrm{E}(Y_t|\mathcal{D}_{t-1}) = \mathrm{E}(\mathrm{E}(Y_t|\theta_t, \mathcal{D}_{t-1})|\mathcal{D}_{t-1}) = \mathrm{E}(F_t\theta_t|\mathcal{D}_{t-1}) = F_t a_t;$$

$$Q_t = \mathrm{Var}(Y_t|\mathcal{D}_{t-1}) = \mathrm{E}(\mathrm{Var}(Y_t|\theta_t, \mathcal{D}_{t-1})|\mathcal{D}_{t-1}) + \mathrm{Var}(\mathrm{E}(Y_t|\theta_t, \mathcal{D}_{t-1})|\mathcal{D}_{t-1})$$
$$= V_t + F_t R_t F_t' .$$

(iii) As shown in (iii) of the previous proposition, we use the Bayes formula for computing the conditional density of $\theta_t|\mathcal{D}_t$, with the density $\mathcal{N}(a_t, R_t)$ of $\theta_t|\mathcal{D}_{t-1}$ as the prior, and the density $\mathcal{N}(F_t\theta_t, V_t)$ of $Y_t|\theta_t$ as the likelihood (remind that $Y_t \amalg \mathcal{D}_{t-1}|\theta_t$). Note that this problem is the same as Bayesian inference for a linear model

$$Y_t = F_t\theta_t + v_t\,,\ v_t \sim \mathcal{N}(0, V_t)$$

[4] Alternatively, you might exploit the independence properties of the error sequences (see problem 2.1) and use the state equation directly:

$$\mathrm{E}(\theta_t|\mathcal{D}_{t-1}) = \mathrm{E}(G_t\theta_{t-1} + w_t|\mathcal{D}_{t-1}) = G_t m_{t-1}$$

$$\mathrm{Var}(\theta_t|\mathcal{D}_{t-1}) = \mathrm{Var}(G_t\theta_{t-1} + w_t|\mathcal{D}_{t-1}) = G_t C_{t-1} G_t' + W_t.$$

Analogously for (ii).

where the regression parameters θ_t have a conjugate Gaussian prior $\mathcal{N}(a_t, R_t)$ and V_t is known. From the results in section 1.2, we have that

$$\theta_t|\mathcal{D}_t \sim \mathcal{N}(m_t, C_t),$$

where, from (1.14),

$$m_t = a_t + R_t F_t' Q_t^{-1}(Y_t - F_t a_t)$$

and, from (1.13),

$$C_t = R_t + R_t F_t' Q_t^{-1} F_t R_t.$$

□

The Kalman filter allows to compute the predictive and filtering densities recursively, starting from $\theta_0|\mathcal{D}_0 \sim \mathcal{N}(m_0, C_0)$ then computing $p(\theta_1|\mathcal{D}_1)$ and proceeding recursively as new data information becomes available.

The conditional density of $\theta_t|\mathcal{D}_t$ solves the filtering problem. However, in many cases one is interested in a point estimate. As we have discussed in chapter 1, section 1.1.2, the Bayesian point estimate of θ_t given the information $\mathcal{D}_t$, with respect to a quadratic loss function $L(\theta_t, a) = (\theta_t - a)'H(\theta_t - a)$, is the conditional expected value $m_t = \mathrm{E}(\theta_t|\mathcal{D}_t)$; this is the optimal estimate since it minimizes the conditional expected loss $\mathrm{E}((\theta_t - a)'H(\theta_t - a)|\mathcal{D}_{t-1})$. The minimum expected loss corresponds to the conditional covariance matrix $\mathrm{Var}(\theta_t|\mathcal{D}_{t-1})$ if $H = I_p$. As we noted in the introductory example in section 2.2, the expression of m_t has the intuitive estimation-correction form "filter mean equal to the prediction mean a_t plus a correction depending on how much the new observation differs from its prediction". The weight of the correction term is given by the *gain matrix*

$$K_t = R_t F_t' Q_t^{-1} .$$

Thus the weight of current information Y_t depends on the observation covariance matrix V_t (through Q_t) and on $R_t = \mathrm{Var}(\theta_t|D_{t-1}) = G_t C_{t-1} G_t' + W_t$.

Example. For the local level model (2.6), the Kalman filter gives

$$\begin{aligned}
\mu_t|\mathcal{D}_{t-1} &\sim \mathcal{N}(m_{t-1}, R_t = C_{t-1} + W),\\
Y_t|\mathcal{D}_{t-1} &\sim \mathcal{N}(f_t = m_{t-1}, Q_t = R_t + V),\\
\mu_t|\mathcal{D}_t &\sim \mathcal{N}(m_t = m_{t-1} + K_t e_t, C_t = K_t V),
\end{aligned}$$

where $K_t = R_t/Q_t$ and $e_t = Y_t - f_t$. It is worth to underline that the behavior of the process (Y_t) is greatly influenced by the ratio between the two error variances, $r = W/V$, which is usually called the *signal-to-noise* ratio (a good exercise for seeing this is to simulate some trajectories of (Y_t), for different values of V and W). This is reflected in the structure of the estimation and forecasting mechanism. Note that $m_t = K_t Y_t + (1 - K_t)m_{t-1}$, a weighted average

of Y_t and m_{t-1}. The weight $K_t = R_t/Q_t = (C_{t-1}+W)/(C_{t-1}+W)+V)$ of the current observation Y_t is also called *adaptive coefficient*, and it is $0 < K_t < 1$. Given C_0, if the signal-to-noise r is small, K_t is small and Y_t receives little weight; if $V = 0$, we have $K_t = 1$ and $m_t = Y_t$, that is, the one-step-ahead forecast is given by the most recent data point. A practical illustration of how different relative magnitudes of W and V affect the mean of the filtered distribution and the one-step-ahead forecasts is given on pages 55 and 62.

The evaluation of the posterior variances C_t (and consequently also of R_t and Q_t) using the iterative updating formulae contained in Theorem 2.2, as simple as it may appear, suffers from numerical instability that may lead to nonsymmetric and even negative definite calculated variance matrices. Alternative, stabler, algorithms have been developed to overcome this issue. Apparently, the most widely used, at least in the Statistics literature, is the square root filter, which provides formulae for the sequential update of a square root[5] of C_t. References for the square root filter are Morf and Kailath (1975) and Anderson and Moore (1979, Ch. 6)

In our work we have found that occasionally, in particular when the observational noise has a small variance, even the square root filter incurs in numerical stability problems, leading to negative definite calculated variances. A more robust algorithm is the one based on sequentially updating the singular value decomposition[6] (SVD) of C_t. The details of the algorithm can be found in Oshman and Bar-Itzhack (1986) and Wang et al. (1992). Strictly speaking, the SVD-based filter can be seen as a square root filter: in fact if $A = UD^2U'$ is the SVD of a variance matrix, then UD is a square root of A. However, compared to the standard square root filtering algorithms, the SVD-based one is typically more stable (see the references for further discussion).

Kalman filter can be performed in R using the function `dlmFilter`. The arguments are the data, `y`, in the form of a numerical vector, matrix, or time series, and the model, `mod`, an object of class `"dlm"` or a list that can be coherced to a dlm object. For the reasons of numerical stability mentioned above, the calculations are performed on the SVD of the variance matrices C_t and R_t. Accordingly, the output provides, for each t, an orthogonal matrix $U_{C,t}$ and a vector $D_{C,t}$ such that $C_t = U_{C,t}\text{diag}(D^2_{C,t})U'_{C,t}$, and similarly for R_t.

The output produced by `dlmFilter`, a list with class attribute `"dlmFiltered"`, includes, in addition to the original data and the model, `y` and `mod`, the means of the predictive and filtered densities in `a` and `m`, and the SVD of the variances of the predictive and filtered densities, in `U.R`, `D.R`, `U.C`, and `D.C`. For convenience, the component `f` of the output list provides the user with one-

[5] A square root of a matrix A is any matrix N such that $A = NN'$.

[6] The SVD of a symmetric nonnegative definite matrix A consists in an orthogonal matrix U and a diagonal matrix D with nonnegative entries such that $A = UD^2U'$.

step-ahead forecasts. The component `U.C` is a list of matrices, the $U_{C,t}$ above, while `D.C` is a matrix containing, stored by row, the $D_{C,t}$ vectors of the SVD of the C_t. Similarly for `U.R` and `D.R`. The function `dlmSvd2var` can be used to reconstruct the variances from their SVD. In the display below we used a random walk plus noise model with the Nile data (figure 2.3). The variances $V = 15100$ and $W = 1468$ are the maximum likelihood estimates. To set up the model we use, instead of `dlm`, the more convenient `dlmModPoly`, which will be discussed in Chapter 3.

R code

```
> mod <- dlmModPoly(order = 1, dV = 15100, dW = 1468)
> unlist(mod)
      m0       C0       FF        V       GG        W
       0 10000000        1    15100        1     1468
> modFilt <- dlmFilter(Nile, mod)
> str(modFilt,1)
List of 9
 $ y  : Time-Series [1:100] from 1871 to 1970: 1120 1160 963 1210 1160 1160 813 1230 137
 $ mod:List of 10
  ..- attr(*, "class")= chr "dlm"
 $ m  : Time-Series [1:101] from 1870 to 1970:    0 1118 1140 1072 1117 ...
 $ U.C:List of 101
 $ D.C: num [1:101, 1] 3162.3  122.8   88.9   76.0   70.0 ...
 $ a  : Time-Series [1:100] from 1871 to 1970:    0 1118 1140 1072 1117 ...
 $ U.R:List of 100
 $ D.R: num [1:100, 1] 3162.5  128.6   96.8   85.1   79.8 ...
 $ f  : Time-Series [1:100] from 1871 to 1970:    0 1118 1140 1072 1117 ...
 - attr(*, "class")= chr "dlmFiltered"
> with(modFilt, dlmSvd2var(U.C[[101]], D.C[101,]))
         [,1]
[1,] 4031.035
```

The last number in the display is the variance of the filtering distribution of the 100-th state vector. Note that m_0 and C_0 are included in the output, which is the reason why `U.C` has one element more than `U.R`, and `m` and `U.D` one row more than `a` and `D.R`.

As we already noted on page 53, the relative magnitude of W and V is an important factor that enters the gain matrix which, in turn, determines how sensitive the state prior-to-posterior updating is to unexpected observations. To illustrate the role of the signal-to-noise ratio W/V in the local level model, we use here two models, with a significantly different signal-to-noise ratio, to estimate the true level of the Nile river. The filtered values for the two models can then be compared.

R code

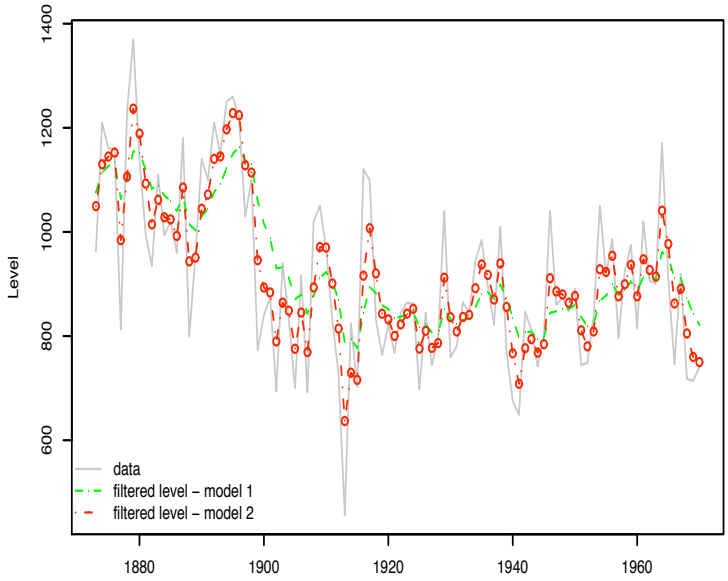

Fig. 2.7. Filtered values of the Nile river level for two different signal-to-noise ratios

```
> mod1 <- dlmModPoly(order = 1, dV = 15100, dW = 0.5 * 1468)
> nileFilt1 <- dlmFilter(Nile, mod1)
> plot(window(cbind(Nile,nileFilt1$m[-1]),start=start(Nile)+1), plot.type='s',
+      type='o', col=c("grey","green"), lty=c(1,2), xlab="", ylab="Level")
> mod2 <- dlmModPoly(order = 1, dV = 15100, dW = 5 * 1468)
> nileFilt2 <- dlmFilter(Nile, mod2)
> lines(window(nileFilt2$m,start=start(Nile)+1), type='o', col="red", lty=4)
> legend("bottomleft", legend=c("data", "filtered level - model 1",
+        "filtered level - model 2"),
+        col=c("grey", "green", "red"), lty=c(1,2,4), pch=1, bty='n')
```

Figure 2.7 displays the filtered levels resulting from the two models. It is appearent that for model 2, which has a signal-to-noise ratio ten times larger than model 1, the filtered values tend to follow more closely the data.

2.4.3 Smoothing

One of the attractive features of state-space models is that estimation and forecasting can be developed sequentially, as new data become available. However, in time series analysis one often has observations on Y_t for a certain period, $t = 1, \ldots, T$ and wants to retrospectively reconstruct the behavior of the system, for studying the socio-economic construct or physical phenomenon

underlying the observations. Again, one has a backward-recursive algorithm for computing the conditional densities of $\theta_t|\mathcal{D}_T$, for $t < T$, starting from the filtering density $p(\theta_T|\mathcal{D}_T)$ and estimating backward all the states' history.

Proposition 2.3. *(Smoothing recursion). (i) Conditional on $\mathcal{D}_T$, the state sequence $(\theta_0, \ldots, \theta_T)$ has backward transition probabilities given by*

$$p(\theta_t|\theta_{t+1}, \mathcal{D}_T) = \frac{p(\theta_{t+1}|\theta_t)p(\theta_t|\mathcal{D}_t)}{p(\theta_{t+1}|\mathcal{D}_t)}.$$

(ii) The smoothing densities of θ_t given D_T can be computed according to the following backward recursion in t (starting from $p(\theta_T|\mathcal{D}_T)$):

$$p(\theta_t|\mathcal{D}_T) = p(\theta_t|\mathcal{D}_t) \int \frac{p(\theta_{t+1}|\theta_t)}{p(\theta_{t+1}|\mathcal{D}_t)} p(\theta_{t+1}|\mathcal{D}_T) d\mu(\theta_{t+1}).$$

Proof. (i) Note that $\theta_{t+1} \amalg (\theta_0, \ldots, \theta_{t-1})|\theta_t, \mathcal{D}_T$ and $\theta_t \amalg (Y_{t+1}, \ldots, Y_T)|\theta_{t+1}$ (you might use the properties of the DAG in Figure 2.6 to show this). Using the Bayes formula

$$p(\theta_t|\theta_{t+1}, \mathcal{D}_T) = p(\theta_t|\theta_{t+1}, \mathcal{D}_t) = \frac{p(\theta_t|\mathcal{D}_t)p(\theta_{t+1}|\theta_t, \mathcal{D}_t)}{p(\theta_{t+1}|\mathcal{D}_t)} = \frac{p(\theta_t|\mathcal{D}_t)p(\theta_{t+1}|\theta_t)}{p(\theta_{t+1}|\mathcal{D}_t)}.$$

(ii) Marginalizing $p(\theta_t, \theta_{t+1}|\mathcal{D}_T)$ we get

$$\begin{aligned}
p(\theta_t|\mathcal{D}_T) &= \int p(\theta_t, \theta_{t+1}|\mathcal{D}_T) d\theta_{t+1} = \int p(\theta_{t+1}|\mathcal{D}_T) p(\theta_t|\theta_{t+1}, \mathcal{D}_T) d\theta_{t+1} \\
&= \int p(\theta_{t+1}|\mathcal{D}_T) p(\theta_t|\theta_{t+1}, \mathcal{D}_t) d\theta_{t+1} \\
&= \int p(\theta_{t+1}|\mathcal{D}_T) \frac{p(\theta_{t+1}|\theta_t, \mathcal{D}_t)p(\theta_t|\mathcal{D}_t)}{p(\theta_{t+1}|\mathcal{D}_t)} d\theta_{t+1} \text{ , using Bayes rule} \\
&= p(\theta_t|\mathcal{D}_t) \int p(\theta_{t+1}|\theta_t) \frac{p(\theta_{t+1}|\mathcal{D}_T)}{p(\theta_{t+1}|\mathcal{D}_t)} d\theta_{t+1}.
\end{aligned}$$

□

For the DLM the above formulae reduce to the following.

Proposition 2.4. *(Smoothing recursion for the DLM). For the DLM 2.4, if $\theta_{t+1}|\mathcal{D}_T \sim \mathcal{N}(s_{t+1}, S_{t+1})$, then $\theta_t|\mathcal{D}_T \sim \mathcal{N}(s_t, S_t)$, where*

$$\begin{aligned}
s_t &= m_t + C_t G'_{t+1} R^{-1}_{t+1}(s_{t+1} - a_{t+1}) \\
S_t &= C_t + C_t G'_{t+1} R^{-1}_{t+1}(S_{t+1} - R_{t+1}) R^{-1}_{t+1} G_{t+1} C_t.
\end{aligned}$$

Proof. From the properties of the multivariate Gaussian distribution, one finds easily that the conditional density of θ_t given $\mathcal{D}_T$ is Gaussian, thus it suffices to compute its expected value and covariance matrix. We have

$$s_t = \mathrm{E}(\theta_t|\mathcal{D}_T) = \mathrm{E}(\mathrm{E}(\theta_t|\theta_{t+1}, \mathcal{D}_T)|\mathcal{D}_T)$$

and

$$S_t = \mathrm{Var}(\theta_t|\mathcal{D}_T) = \mathrm{Var}(\mathrm{E}(\theta_t|\theta_{t+1}, \mathcal{D}_T)|\mathcal{D}_T) + \mathrm{E}(\mathrm{Var}(\theta_t|\theta_{t+1}, \mathcal{D}_T)|\mathcal{D}_T).$$

Now observe that, as from part (i) of the previous proposition, $\theta_t \amalg (Y_{t+1}, \ldots, Y_T)|\theta_{t+1}$ so that $p(\theta_t|\theta_{t+1}, \mathcal{D}_T) = p(\theta_t|\theta_{t+1}, \mathcal{D}_t)$ and we can use Bayes formula for computing it. Here, $p(\theta_{t+1}|\theta_t, \mathcal{D}_t) = p(\theta_{t+1}|\theta_t)$ is described by the state equation

$$\theta_{t+1} = G_{t+1}\theta_t + w_{t+1},\ w_{t+1} \sim N(0, W_{t+1})$$

that is $\theta_{t+1}|\theta_t \sim \mathcal{N}(G_{t+1}\theta_t, W_{t+1})$. The role of prior is played by $p(\theta_t|\mathcal{D}_t)$ which is $\mathcal{N}(m_t, C_t)$. Using (1.14) and (1.13), we find that

$$\begin{aligned}
\mathrm{E}(\theta_t|\theta_{t+1}, \mathcal{D}_t) &= m_t + C_t G'_{t+1}(G_{t+1}C_tG'_{t+1} + W_{t+1})^{-1}(\theta_{t+1} - G_{t+1}m_t) \\
&= m_t + C_tG'_{t+1}R^{-1}_{t+1}(\theta_{t+1} - a_{t+1}) \\
\mathrm{Var}(\theta_t|\theta_{t+1}, \mathcal{D}_t) &= C_t - C_tG'_{t+1}R^{-1}_{t+1}G_{t+1}C_t,
\end{aligned}$$

from which

$$\begin{aligned}
s_t &= \mathrm{E}(\mathrm{E}(\theta_t|\theta_{t+1}, \mathcal{D}_t)|\mathcal{D}_T) = m_t + C_tG'_{t+1}R^{-1}_{t+1}(s_{t+1} - a_{t+1}) \\
S_t &= \mathrm{Var}(\mathrm{E}(\theta_t|\theta_{t+1}, \mathcal{D}_t)|\mathcal{D}_T) + \mathrm{E}(\mathrm{Var}(\theta_t|\theta_{t+1}, \mathcal{D}_t)|\mathcal{D}_T) \\
&= C_t - C_tG'_{t+1}R^{-1}_{t+1}G_{t+1}C_t + C_tG'_{t+1}R^{-1}_{t+1}S_{t+1}R^{-1}_{t+1}G_{t+1}C_t \\
&= C_t + C_tG'_{t+1}R^{-1}_{t+1}(S_{t+1} - R_{t+1})R^{-1}_{t+1}G_{t+1}C_t,
\end{aligned}$$

being $\mathrm{E}(\theta_{t+1}|\mathcal{D}_T) = s_{t+1}$ and $\mathrm{Var}(\theta_{t+1}|\mathcal{D}_T) = S_{t+1}$ by assumption. □

The Kalman smoother allows to compute the densities of $\theta_t|\mathcal{D}_T$, starting from $t = T-1$, in which case $\theta_T|\mathcal{D}_T \sim \mathcal{N}(s_T = m_T, S_T = C_T)$, and then proceeding backward for computing the densities of $\theta_t|\mathcal{D}_T$ for $t = T-2$, $t = T-3$, etcetera.

About the numerical stability of the smoothing algorithm, the same caveat holds as for the filtering recursions. The formulae of Proposition 2.4 are subject to numerical instability, and more robust square root and SVD-based smoothers are available (see Zhang and Li; 1996). The function `dlmSmooth` performs the calculations in R, starting from an object of class `"dlmFiltered"`. The output is a list with components `s`, the means of the smoothing distributions, and `U.S`, `D.S`, their variances, given in terms of their SVD. The following display illustrates the use of `dlmSmooth` on the Nile data.

R code

```
> modSmooth <- dlmSmooth(modFilt)
> str(modSmooth,1)
List of 3
 $ s  : Time-Series [1:101] from 1870 to 1970: 1111 1111 1111 1105 1113 ...
```

```
 $ U.S:List of 101
 $ D.S: num [1:101, 1] 74.1 63.5 56.9 53.1 50.9 ...
> with(modSmooth, drop(dlmSvd2var(U.S[[101]], D.S[101,])))
[1] 4031.035
> with(modFilt, drop(dlmSvd2var(U.C[[51]], D.C[51,])))
[1] 4031.035
> with(modSmooth, drop(dlmSvd2var(U.S[[51]], D.S[51,])))
[1] 2325.985
```

Note, in the above display, how filtering and smoothing variances at time 100, the time of the last observation, are equal, but the smoothing variance at time 50 is much smaller than the filtering variance at the same time. This is due to the fact that in the filtering distribution at time 50 one is conditioning on the first fifty observations only, while in the smoothing distribution the conditioning is with respect to all the one hundred observations available. As the display below illustrates, the variance of the smoothing distribution can be used to construct pointwise probability intervals for the state components[7] – only one in this example. The plot produced by the code below is shown in Figure 2.8

R code

```
> hwid <- qnorm((1-0.95) / 2) *
+     sqrt(with(modSmooth, unlist(dlmSvd2var(U.S, D.S))))
> smooth <- cbind(modSmooth$s, as.vector(modSmooth$s) + hwid %o% c(1,-1))
> plot(cbind(Nile, window(smooth, start=start(Nile))), plot.type='s',
+      col=c("grey", "magenta", "cyan", "cyan"), lty=c(1, 2, 3, 3), type='o',
+      ylab="Level", xlab="")
> legend("bottomleft", legend=c("data", "smoothed level", "95% probability limits"),
+        col=c("grey", "magenta", "cyan"), lty=1:3, pch=1, bty='n')
```

Here is another example. The data consist of a quartely time series of quarterly consumer expenditure on durable goods in UK, in 1958 pounds, from the first quarter of 1957 to the last quarter of 1967[8]. A DLM including a local level plus a quarterly seasonal component was fitted to the data, giving the parameter estimates that we use here to illustrate filtering and smoothing in R. In this model the state vector is 4-dimensional. Two of its components have a particularly relevant interpretation: the first one can be thought of as the true, deseasonalized, level of the series; the second is a dynamic seasonal component. The series is obtained, according to the model, by adding observational noise to the sum of the first and second component of the state vector, as can be deduced from the *FF* matrix. Figure 2.9 shows the data, together with the

[7] At the time of writing the package does not have functions returning directly probability intervals. This may change in future releases.

[8] Data taken from `http://www-personal.buseco.monash.edu.au/~hyndman/TSDL/`

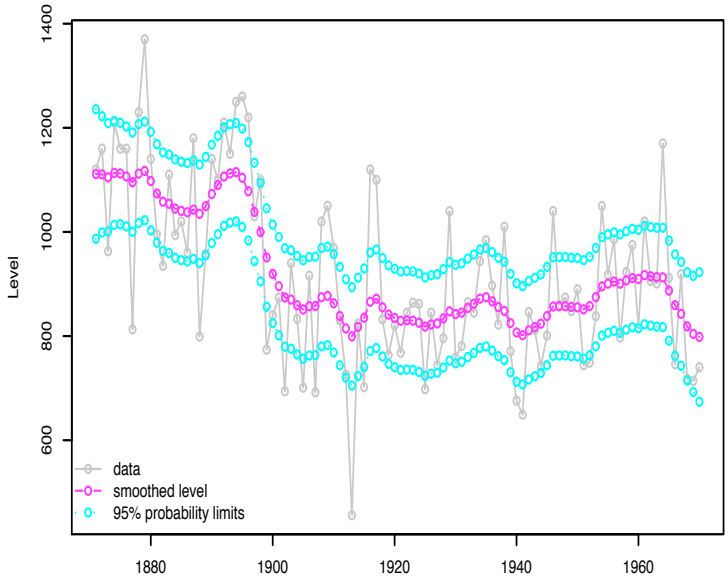

Fig. 2.8. Smoothed values of the Nile river level, with 95% probability limits

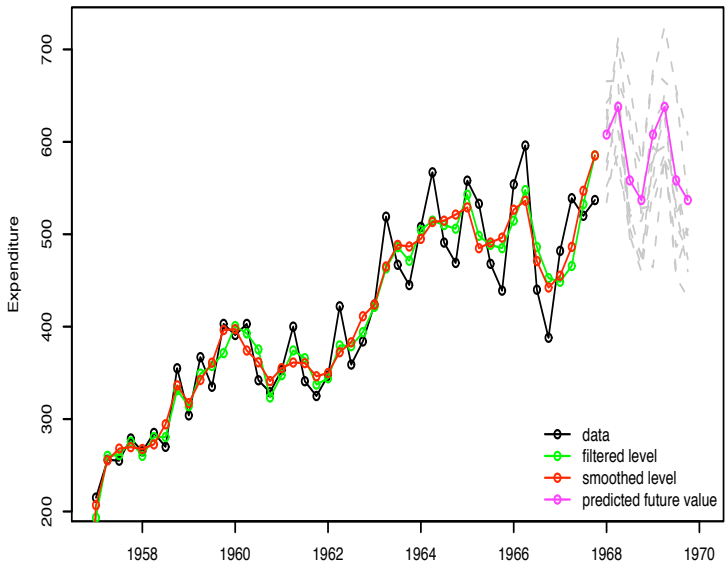

Fig. 2.9. Quarterly expenditure on durable goods, with filtered, smoothed and predicted level

filtered and smoothed level. These values are just the first components of the series of filtered and smoothed state vectors. In addition to the deseasonalized

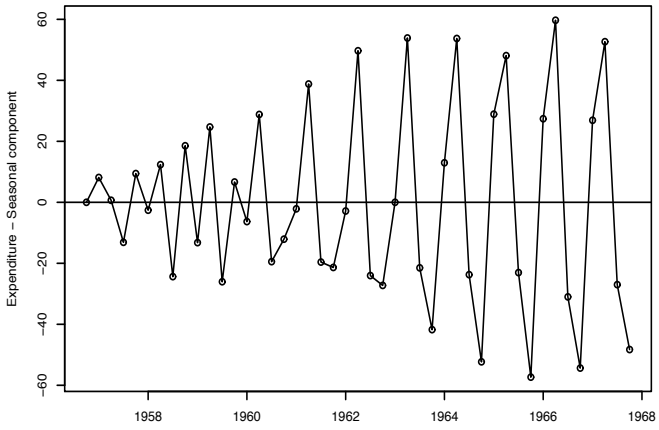

Fig. 2.10. Quarterly expenditure on durable goods: smoothed seasonal component

level of the series, one can also estimate the seasonal component, which is just the second component of the smoothed or filtered state vector. Figure 2.10 shows the smoothed seasonal component. It is worth stressing that the model is dynamic, hence the seasonal component is allowed to vary as time goes by. This is clearly the case in the present example: from an alternating of positive and negative values at the beginning of the observation period, the series moves to a two-positive two-negative pattern in the second half. The display below shows how filtered and smoothed values have been obtained in R, as well as how the plots were created (up to 1967 – the predicted level is explained in the next subsection). The function *bdiag* is a handy function that creates a block diagonal matrix from the individual blocks, or from a list containing the blocks.

R code

```
> expd <- ts(read.table("Datasets/qconsum.dat", skip = 4,
+                       colClasses = "numeric")[,1],
+            start = c(1957,1), frequency = 4)
> mod <- dlm(m0 = rep(0,4), C0 = 1e-8 * diag(4),
+            FF = matrix(c(1,1,0,0), nr=1),
+            V = 1e-3,
+            GG = bdiag(matrix(1),
+            matrix(c(-1,-1,-1,1,0,0,0,1,0),nr=3,byrow=TRUE)),
+            W = diag(c(771.35, 86.48, 0, 0), nr=4))
> modFilt <- dlmFilter(expd, mod)
```

```
> modSmooth <- dlmSmooth(modFilt)
> plot(expd, type='o', xlim=c(1957, 1970), ylim=c(210, 725), ylab="Expenditure")
> lines(modFilt$m[,1], col='green', type='o')
> lines(modSmooth$s[,1], col='red', type='o')
> plot(modSmooth$s[,2], type='o', ylab="Expenditure - Seasonal component")
> abline(h=0)
```

2.5 Forecasting

With $\mathcal{D}_t$ at hand, one can be interested in forecasting future values of the observations, Y_{t+k}, or of the state vectors, θ_{t+k}. For DLM, the recursive form of computations makes it natural to compute the one-step-ahead forecasts and to update them sequentially, as new data become available. This is clearly of interest in applied problems where the data do arrive sequentially, such as in day-by-day forecasting the price of a stock, or in tracking a moving target; but one-step-ahead forecasts are often also computed "in-sample", as a tool for checking the performance of the model.

The one-step-ahead predictive densities, for states and observations, are obtained as a byproduct of the Kalman filter, as presented in Theorem 2.2.

In R, the one-step-ahead forecasts $f_t = \mathrm{E}(Y_t|\mathcal{D}_t)$ are provided in the output of the function `dlmFilter`. Since for each t the one-step-ahead forecast of the observation, f_t, is a linear function of the filtering mean m_{t-1}, the magnitude of the gain matrix plays the same role in determining how sensitive f_t is to an unexpected observation Y_{t-1} as it did for m_{t-1}. In the case of the random walk plus noise model this is particularly evident, since in this case $f_t = m_{t-1}$. Figure 2.11, produced with the code below, contains the one-step-ahead forecasts obtained from the local level models with different signal-to-noise ratio defined in the display on page 55.

R code

```
> plot(window(cbind(Nile, nileFilt1$f, nileFilt2$f), start=1880, end=1920),
+      plot.type='s', type='o', col=c("grey", "green", "red"), lty=c(1,2,4),
+      xlab="", ylab="Level")
> legend("bottomleft", legend=c("data", paste("one-step-ahead forecast - model", 1:2)),
+        col=c("grey", "green", "red"), lty=c(1,2,4), pch=1, bty='n')
```

To elaborate more on the same example, we note that the signal-to-noise ratio need not be constant in time. The construction of the Ashwan dam in 1898, for instance, can be expected to produce a major change in the level of the Nile river. A simple way to incorporate this expected level shift in the model is to assume a system evolution variance W_t larger than usual for that year and the following one. In this way the estimated true level of the river will adapt

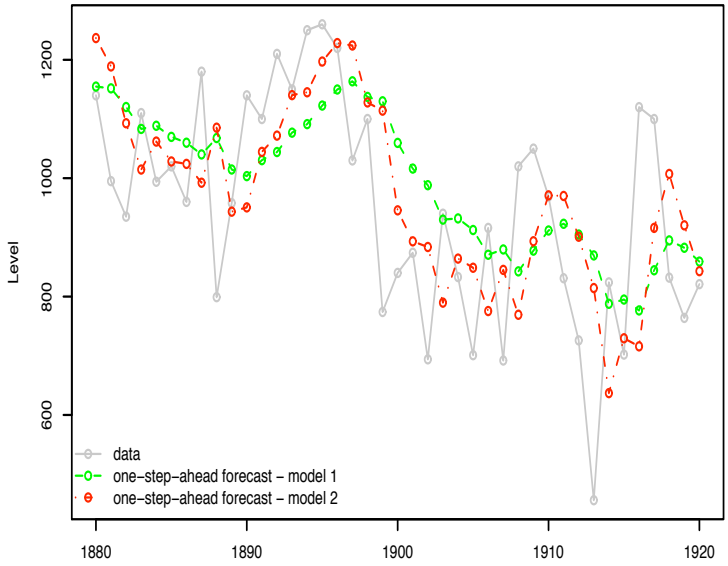

Fig. 2.11. One-step-ahead forecasts for the Nile level using different signal-to-noise ratios

faster to the new regime, leading in turn to more accurate one-step-ahead forecasts. The code below illustrates this idea.

R code

```
> mod0 <- dlmModPoly(order = 1, dV = 15100, dW = 1468)
> X <- ts(matrix(mod0$W, nc=1, nr=length(Nile)), start=start(Nile))
> window(X, 1898, 1899) <- 12 * mod0$W
> modDam <- mod0
> modDam$X <- X
> modDam$JW <- matrix(1,1,1)
> damFilt <- dlmFilter(Nile, modDam)
> mod0Filt <- dlmFilter(Nile, mod0)
> plot(window(cbind(Nile, mod0Filt$f, damFilt$f), start=1880, end=1920),
+      plot.type='s', type='o', col=c("grey", "green", "red"),
+      lty=c(1,2,4), xlab="", ylab="Level")
> abline(v=1898, lty=2)
> legend("bottomleft", col=c("grey", "red", "green"), lty=c(1,4,2), pch=1, bty='n',
+        legend=c("data", paste("one-step-ahead forecast -", c("modDam", "mod0"))))
```

Note in Figure 2.12 how, using the modified model *modDam*, the forecast for the level of the river in 1900 is already around what is the new river level,

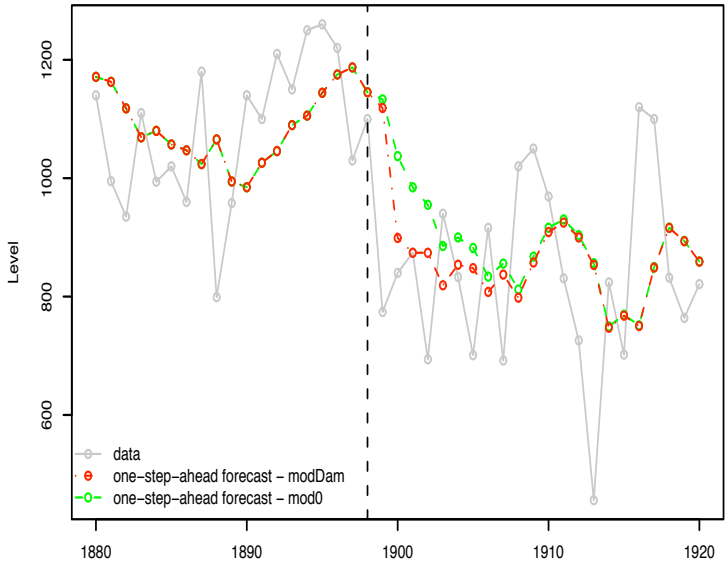

Fig. 2.12. One-step-ahead forecasts of Nile river level with and without change point

while for the other model this happens only around 1907.

In many applications one is interested in looking a bit further in the future, and provide possible scenarios of the behavior of the series for k steps ahead. We present here the recursive formulae for the means and variances of the conditional distributions of states and observations at a future time $t+k$, given the data up to time t. In view of the Markovian nature of the model, the filtering distribution at time t acts like an initial distribution for the future evolution of the model. To be more precise, the joint distribution of present and future states $(\theta_{t+k})_{k\geq 0}$, and future observations $(Y_{t+k})_{k\geq 1}$ is that of a DLM having the relevant system/observation matrices and variances, and initial distribution $p(\theta_t \mid \mathcal{D}_t)$. The information about the future provided by the data is all contained in this distribution. In particular, since the data are only used to obtain m_t, the mean of $p(\theta_t \mid \mathcal{D}_t)$, it follows that m_t provides a summary of the data which is sufficient for predictive purposes.

$$\begin{array}{ccccccc} \theta_t & & \longrightarrow & \theta_{t+1} & \longrightarrow \cdots \longrightarrow & & \theta_{t+k} \\ | & & & & & & | \\ (Y_1, \ldots, Y_t) & & & & & & Y_{t+k} \end{array}$$

Fig. 2.13. Flow of information from $(Y_1, \ldots, Y_t)$ to Y_{t+k}

You can have a further intuition about that looking at the DAG representing the dependence structure among the variables (figure 2.6). We see that the path from $(Y_1, \ldots, Y_t)$ to Y_{t+k} is as in figure 2.13, showing that the data $(Y_1, \ldots, Y_t)$ provide information on θ_t, which gives information about the future state evolution up to θ_{t+k} and consequently on Y_{t+k}. Of course, as k gets larger, more uncertainty enters in the picture, and the forecasts will be less and less precise.

Proposition 2.5 provides recursive formulae to compute first and second moments of the predictive distributions. We need first some notation. For $k \geq 1$, define

$$a_t(k) = \mathrm{E}(\theta_{t+k} \mid \mathcal{D}_t), \tag{2.8a}$$
$$R_t(k) = \mathrm{Var}(\theta_{t+k} \mid \mathcal{D}_t), \tag{2.8b}$$
$$f_t(k) = \mathrm{E}(Y_{t+k} \mid \mathcal{D}_t), \tag{2.8c}$$
$$Q_t(k) = \mathrm{Var}(Y_{t+k} \mid \mathcal{D}_t). \tag{2.8d}$$

Proposition 2.5. *Set $a_t(0) = m_t$ and $R_t(0) = C_t$. Then, for $k \geq 1$, the following hold:*

1. *the distribution of θ_{t+k} given $\mathcal{D}_t$ is Gaussian, with*

$$a_t(k) = G_{t+k} a_{t,k-1},$$
$$R_t(k) = G_{t+k} R_{t,k-1} G'_{t+k} + W_{t+k};$$

2. *the distribution of Y_{t+k} given $\mathcal{D}_t$ is Gaussian, with*

$$f_t(k) = F_{t+k} a_t(k),$$
$$Q_t(k) = F_{t+k} R_t(k) F'_{t+k} + V_t.$$

Proof. As we have already noted, all conditional distributions are Gaussian. Therefore, we only need to prove the formulae giving the means and variances. We proceed by induction. The result holds for $k = 1$ in view of Theorem 2.2. For $k > 1$, [9]

[9] Again, you can alternatively exploit the independence properties of the error sequences (see problem 2.1) and use the state equation directly:

$$a_t(k) = \mathrm{E}(\theta_{t+k} \mid \mathcal{D}_t) = \mathrm{E}(G_{t+k}\theta_{t+k-1} + w_{t+k} \mid \mathcal{D}_t) = G_{t+k} a_{t,k-1},$$
$$R_t(k) = \mathrm{Var}(\theta_{t+k} \mid \mathcal{D}_t) = \mathrm{Var}(G_{t+k}\theta_{t+k-1} + w_{t+k} \mid \mathcal{D}_t) = G_{t+k} R_{t,k-1} G'_{t+k} + W_{t+k}$$

and analogously, from the observation equation:

$$f_t(k) = \mathrm{E}(Y_{t+k} \mid \mathcal{D}_t) = \mathrm{E}(F_{t+k}\theta_{t+k} + v_{t+k} \mid \mathcal{D}_t) = F_{t+k} a_t(k),$$
$$Q_t(k) = \mathrm{Var}(Y_{t+k} \mid \mathcal{D}_t) = \mathrm{Var}(F_{t+k}\theta_{t+k} + v_{t+k} \mid \mathcal{D}_t) = F_{t+k} R_t(k) F'_{t+k} + V_{t+k}.$$

$$
\begin{aligned}
a_t(k) &= \mathrm{E}(\theta_{t+k} \mid \mathcal{D}_t) = \mathrm{E}(\mathrm{E}(\theta_{t+k} \mid \mathcal{D}_t, \theta_{t+k-1}) \mid \mathcal{D}_t) \\
&= \mathrm{E}(G_{t+k}\theta_{t+k-1} \mid \mathcal{D}_t) = G_{t+k}a_{t,k-1}, \\
R_t(k) &= \mathrm{Var}(\theta_{t+k} \mid \mathcal{D}_t) = \mathrm{Var}(\mathrm{E}(\theta_{t+k} \mid \mathcal{D}_t, \theta_{t+k-1}) \mid \mathcal{D}_t) \\
&\quad + \mathrm{E}(\mathrm{Var}(\theta_{t+k} \mid \mathcal{D}_t, \theta_{t+k-1}) \mid \mathcal{D}_t) \\
&= G_{t+k}R_{t,k-1}G'_{t+k} + W_{t+k}, \\
f_t(k) &= \mathrm{E}(Y_{t+k} \mid \mathcal{D}_t) = \mathrm{E}(\mathrm{E}(Y_{t+k} \mid \mathcal{D}_t, \theta_{t+k}) \mid \mathcal{D}_t) \\
&= \mathrm{E}(F_{t+k}\theta_{t+k} \mid \mathcal{D}_t) = F_{t+k}a_t(k), \\
Q_t(k) &= \mathrm{Var}(Y_{t+k} \mid \mathcal{D}_t) = \mathrm{Var}(\mathrm{E}(Y_{t+k} \mid \mathcal{D}_t, \theta_{t+k}) \mid \mathcal{D}_t) \\
&\quad + \mathrm{E}(\mathrm{Var}(Y_{t+k} \mid \mathcal{D}_t, \theta_{t+k}) \mid \mathcal{D}_t) \\
&= F_{t+k}R_t(k)F'_{t+k} + V_{t+k},
\end{aligned}
$$

□

Note that the data only enter the predictive distributions through the mean of the filtering distribution at the time the last observation was taken. The function `dlmForecast` computes the means and variances of the predictive distributions of the observations and the states. Optionally, it can be used to draw a sample of future states and observations. The principal argument of `dlmForecast` is a `dlmFiltered` object. Alternatively, it can be a `dlm` object (or a list with the appropriate named components), where the components `m0` and `C0` are interpreted as being the mean and variance of the state vector at the end of the observation period, given the data, i.e., they are the mean and variance of the last (most recent) filtering distribution. The code below shows how to obtain predicted values of the series for the two years following the last observation, together with a sample from their distribution (dashed lines in Figure 2.9).

R code

```
> fore <- dlmForecast(modFuture, nAhead = 8, sampleNew = 10)
> invisible(lapply(fore$newObs, function(x) lines(x, col='grey', lty=2)))
> lines(fore$f, col="magenta", type='o')
```

The innovation process

As we have seen, for DLM the Kalman filter provides the filtering estimate m_t, given the information $\mathcal{D}_t$, as the previous estimate m_{t-1} corrected by a term which depends on the forecast error

$$e_t = Y_t - \mathrm{E}(Y_t|\mathcal{D}_{t-1}) = Y_t - f_t.$$

The forecast errors can alternatively be written in terms of the estimation errors as follows:

$$\begin{aligned} e_t &= Y_t - F_t a_t = F_t\theta_t + v_t - F_t a_t \\ &= F_t(\theta_t - a_t) + v_t = F_t(\theta_t - G_t m_{t-1}) + v_t \end{aligned}$$

For the sequence $(e_t, t \geq 1)$, some interesting properties hold.

(i) The expected value of e_t is zero, since $\mathrm{E}(e_t) = \mathrm{E}(\mathrm{E}(e_t \mid \mathcal{D}_{t-1})) = 0$.

(ii) The random vector e_t is uncorrelated with any function of $Y_1, \ldots, Y_{t-1}$. In particular, if $s < t$, then e_t and Y_s are uncorrelated. Let $Z = g(Y_1, \ldots, Y_{t-1})$. Then

$$\begin{aligned} \mathrm{Cov}(e_t, Z) &= \mathrm{E}(e_t Z) = \mathrm{E}(\mathrm{E}(e_t Z \mid \mathcal{D}_{t-1})) \\ &= \mathrm{E}(\mathrm{E}(Y_t - f_t \mid \mathcal{D}_{t-1})Z) = 0. \end{aligned}$$

In more abstract terms, this amounts to saying that $\mathrm{E}(Y_t \mid \mathcal{D}_{t-1})$ is the orthogonal projection of Y_t on the linear vector space of random variables that are functions of $Y_1, \ldots, Y_{t-1}$.

(iii) For $s \neq t$, e_s and e_t are uncorrelated. This follows from 2 since, if $s < t$, each component of e_s is a function of $Y_1, \ldots, Y_{t-1}$.

(iv) e_t is a linear function of $Y_1, \ldots, Y_{t-1}$. Since $Y_1, \ldots, Y_{t-1}$ have a joint Gaussian distribution, $\mathrm{E}(Y_t \mid \mathcal{D}_{t-1})$ is a linear function of $Y_1, \ldots, Y_{t-1}$.

(v) $(e_t, t \geq 1)$ is a Gaussian process. From 4 it follows that, for every t, $(e_1, \ldots, e_t)$ is a linear function of $(Y_1, \ldots, Y_t)$ and therefore have a Gaussian distribution. As a consequence, since the e_t's are uncorrelated by 3, they are also independent. Moreover, since $Y_t \mid \mathcal{D}_{t-1} \sim \mathcal{N}_m(f_t, Q_t)$, one has that $e_t \mid \mathcal{D}_{t-1} \sim \mathcal{N}_m(0, Q_t)$. But Q_t does not depend on the data $Y_1, \ldots, Y_{t-1}$, and so neither does the conditional distribution $\mathcal{N}_m(0, Q_t)$, which must therefore be also the unconditional distribution of e_t:

$$e_t \sim \mathcal{N}_m(0, Q_t), \qquad t = 1, 2, \ldots.$$

The forecast errors e_t are also called *innovations*. The representation $Y_t = f_t + e_t$ justifies this terminology, since one can think of Y_t as the sum of a component which is predictable from past observations, f_t, and another component, e_t, which is independent of the past and therefore contains the real new information carried by the observation Y_t.

For a DLM, one sometimes works with the so-called *innovation form* of the model. This is obtained by choosing as new state variables the vectors $a_t = \mathrm{E}(\theta_t | \mathcal{D}_{t-1})$. Then the observation equation is derived from $e_t = Y_t - f_t = Y_t - F_t a_t$:

$$Y_t = F_t a_t + e_t \tag{2.9}$$

and, being $a_t = G_t m_{t-1}$, where m_{t-1} is given by the Kalman filter :

$$a_t = G_t m_{t-1} = G_t a_{t-1} + G_t R_{t-1} F'_{t-1} Q^{-1}_{t-1} e_t,$$

so that the new state equation is

$$a_t = G_t a_{t-1} + w^*_t, \tag{2.10}$$

with $w_t^* = G_t R_{t-1} F'_{t-1} Q_{t-1}^{-1} e_t$. The system (2.9) and (2.10) is the innovation form of the DLM. Note that, in this form, the observation errors and the system errors are no longer independent, that is the dynamics of the states is no longer independent from the observations. The main advantage is that in the innovation form all components of the state vector on which we cannot obtain any information from the observations are automatically removed. It is thus in some sense a minimal model.

2.5.1 Model checking

When the observations are univariate, the standardized innovations, defined by $\tilde{e}_t = e_t/\sqrt{Q_t}$, are a Gaussian white noise, i.e. a sequence of independent identically distributed zero-mean normal random variables. This property can be exploited to check the model assumptions: if the model is correct, the sequence $\tilde{e}_1, \ldots, \tilde{e}_t$ computed from the data should look like a sample of size t from a standard normal distribution. Many statistical tests, several of them readily available in R, can be carried out on the standardized innovations. They fall into two broad categories: those aimed at checking if the distribution of the $\tilde{e}_t$'s is standard normal, and those aimed at checking whether the $\tilde{e}_t$'s are uncorrelated. We will illustrate the use of some of these tests in Chapter 3. However, most of the time we take a more informal approach to model checking, based on the subjective assessment of selected diagnostic plots. The most illuminating are, in the authors' opinion, a QQ-plot and a plot of the empirical autocorrelation function of the standardized innovations. The former is used to assess normality, while the latter reveals departures from uncorrelatedness. A time series plot of the standardized innovations may prove useful in detecting outliers, change points and other unexpected patterns.

For multivariate observations we usually apply the same univariate graphical diagnostic tools component-wise to the innovation sequence. A further step would be to adopt the vector standardization $\tilde{e}_t = Q_t^{-1/2} e_t$. This makes the components of $\tilde{e}_t$ independent and identically distributed according to a standard normal distribution. Using this standardization, the sequence $\tilde{e}_{1,1}, \tilde{e}_{1,2} \ldots, \tilde{e}_{1,p}, \ldots, \tilde{e}_{t,p}$ should look like a sample of size tp from a univariate standard normal distribution. This approach, however, is not very popular in applied work and it will not be used in this book.

2.6 Limiting behavior

xxx Stability, observability, controllability... xxx

Problems

2.1. Show that

(i) w_t and $(Y_1, \ldots, Y_{t-1})$ are independent;
(ii) w_t and $(\theta_1, \ldots, \theta_{t-1})$ are independent;
(iii) v_t and $(Y_1, \ldots, Y_t)$ are independent;
(iv) v_t and $(\theta_1, \ldots, \theta_t)$ are independent.

2.2. Show that the DLM satisfies the conditional independence assumptions A.1 and A.2 of state space models.

2.3. Plot the following data:
$(Y_t, t = 1, \ldots, 10) = (17, 16.6, 16.3, 16.1, 17.1, 16.9, 16.8, 17.4, 17.1, 17)$.
Consider the random walk plus noise model

$$\begin{aligned} Y_t &= \mu_t + v_t \quad , \quad v_t \sim N(0, 0.25) \\ \mu_t &= \mu_{t-1} + w_t \quad , \quad w_t \sim N(0, 25) \end{aligned}$$

with $V = 0.25$, $W = 25$, and $\mu_0 \sim N(17, 1)$.
(a) Compute the filtering states estimates.
(b)Compute the one-step ahead forecasts $f_t, t = 1, \ldots, 10$ and plot them, together with the observations. Comment briefly.
(c) What is the effect of the observation variance V and of the system variance W on the forecasts? Repeat the exercise with different choices of V and W.
(d) Compute the smoothing state estimates and plot them.

2.4. This requires maximum likelihood estimates (see chapter 4). For the data and model of exercise 2.3, compute the maximum likelihood estimates of the variances V and W (since these must be positive, write them as $V = \exp(u_1)$, $W = \exp(u_2)$ and compute the MLE of the parameters (u_1, u_2)). Then repeat exercise 2.3, using the MLE of V and W.

2.5. Repeat the exercise 2.4 with the following data....

using again a random walk plus noise model. Discuss the effects of the choice of the initial distribution.

2.6. Let $R_{t,h,k} = \mathrm{Cov}(\theta_{t+h}, \theta_{t+k} \mid \mathcal{D}_t)$ and $Q_{t,h,k} = \mathrm{Cov}(Y_{t+h}, Y_{t+k} \mid \mathcal{D}_t)$ for $h, k > 0$, so that $R_{t,k,k} = R_t(k)$ and $Q_{t,k,k} = Q_t(k)$, according to definition (2.8b) and (2.8d).

(i) Show that $R_{t,h,k}$ can be computed recursively via the formula:

$$R_{t,h,k} = G_{t+h} R_{t,h-1,k}, \qquad h > k.$$

(ii) Show that $Q_{t,h,k}$ is equal to $F_{t+h} R_{t,h,k} F'_{t+k}$.

(iii) Find explicit formulas for $R_{t,h,k}$ and $Q_{t,h,k}$ for the random walk plus noise model.

2.7. Derive the filter formulae for the DLM with intercepts:

$$v_t \sim \mathcal{N}(\delta_t, V_t),\ w_t \sim \mathcal{N}(\lambda_t, W_t).$$

3

Model specification

This chapter is devoted to the description of specific classes of dynamic linear models that, alone or in combinations, are most often used to model univariate or multivariate time series. The additive structure of dynamic linear models makes it easy, as we will show in more detail, 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 modeller, that 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 dynamic linear model framework.

3.1 Classical tools for time series analysis

3.1.1 Empirical methods

Exponentially weighted moving average

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 $Y_1, \ldots, Y_t$ and she 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:

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

For t large, the weights $(1-\lambda)\lambda^j$ 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

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

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 65). At time t, forecasts of future observations are taken to be equal to the forecast of Y_{t+1}; in other words, $\hat{y}_{t+k|t} = \hat{y}_{t+1|t}$, $k = 1, 2, \ldots$, and the forecast function is constant.

Extensions of EWMA exist that allow for a linear forecast function. For example, the popular Holt-Winters point predictor for non-seasonal time series includes a trend component, decomposing Y_t as the sum of a local level and a local trend: $Y_t = L_t + T_t$. Point forecasts are then obtained by combining exponential smoothing forecasts of the level and the trend:

$$\hat{y}_{t+1|t} = (\hat{L}_{t+1|t} + \hat{T}_{t+1|t}),$$

where

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

The above recursive formulae can be rewritten as

$$\hat{L}_{t+1|t} = \hat{y}_{t|t-1} + \lambda e_t \tag{3.3}$$

$$\hat{T}_{t+1|t} = \hat{T}_{t|t-1} + \lambda\gamma e_t, \tag{3.4}$$

where $e_t = Y_t - \hat{y}_{t|t-1}$ is the forecast error. Further extensions to include a seasonal component are possible, see e.g. Chatfield (2004).

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. They can also be derived from an underlying dynamic linear model, which can in this case be used to provide a theoretical justification for the method and to derive probability intervals.

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.; 1994). For nonnegative integers p and q, a univariate stationary ARMA(p,q) model is defined by the relation

$$Y_t = \mu + \sum_{j=1}^{p} \phi_j (Y_{t-j} - \mu) + \sum_{j=1}^{q} \psi_j \epsilon_{t-j} + \epsilon_t, \tag{3.5}$$

where (ϵ_t) is Gaussian white noise with variance σ_ϵ^2 and the parameters $\phi_1, \ldots, \phi_p$ 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 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 3.5, taking (ϵ_t) to be m-dimensional Gaussian white noise with variance Σ_ϵ^2 and the parameters $\phi_1, \ldots, \phi_p$ and $\psi_1, \ldots, \psi_q$ to be m by m matrices satisfying appropriate stationarity restrictions. Although in principle as simple to define as for univariate data, 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`.

It is possible to represent an ARIMA model, univariate or multivariate, as a DLM, as we will show in 3.2.4 and 3.3.8. 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 very 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 individul 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 DLM 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 together 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

$$Y_t = Y_{1,t} + \cdots + Y_{h,t}, \tag{3.6}$$

where $Y_{i,t}$ might represent a trend component, $Y_{2,t}$ a seasonal component, and so on. The i-th component $Y_{i,t}$, $i = 1, \ldots, h$, 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 $(p_i \times 1)$ state vectors $\theta_{i,t}$ are distinct and $(Y_{i,t}, \theta_{i,t})$ and $(Y_{j,t}, \theta_{j,t})$ are mutually independent for all $i \neq j$. The component DLM's are then combined for obtaining 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

$$\theta_t = \begin{pmatrix} \theta_{1,t} \\ \vdots \\ \theta_{h,t} \end{pmatrix}, \quad F_t = \left(F_{1,t} | \cdots | F_{h,t}\right),$$

G_t and W_t are the block diagonal matrices

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

and $V_t = \sum_{i=1}^{j} V_{i,t}$. In all this section, we assume that the covariance matrices are known, but the analysis can be extended to the case of unknown V_t and W_t (see Chapter 4).

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. More detailed examples will be given later in this chapter, especially in 3.2.5. We start by introducing the families of DLM that are commonly used as basic building blocks in the representation (3.6). 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 DLM 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 $f_t(k) = \mathrm{E}(Y_{t+k}|\mathcal{D}_t)$. A polynomial model of order n is a DLM with constant matrices $F_t = F$ and $G_t = G$, known covariance matrices V_t and W_t, and a forecast function of the form

$$f_t(k) = E(Y_{t+k}|\mathcal{D}_t) = a_{t,0} + a_{t,1}k + \cdots + a_{t,n-1}k^{n-1}, \quad k \geq 0\,, \tag{3.7}$$

where $a_{t,0}, \ldots, a_{t,n-1}$ are linear functions of $m_t = E(\theta_t|\mathcal{D}_t)$ and are independent of k. Thus, the forecast function is a polynomial of order $(n-1)$ in k (in fact, 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 (2.6) 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 3.2.1 shows some simulated trajectories of (Y_t) and (μ_t) for varying values of the ratio r (see Problem 3.1).

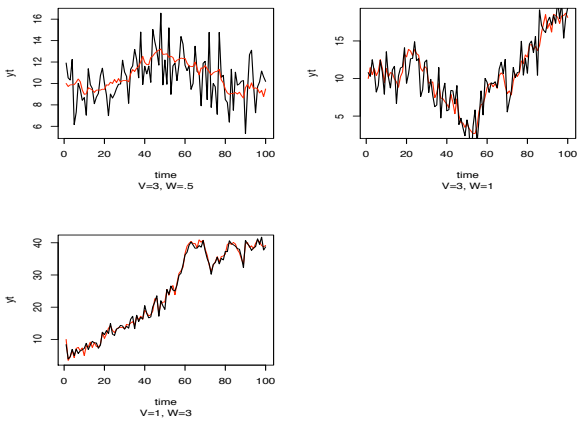

Fig. 3.1. Trajectories of the random walk plus noise, for varying values of the signal-to-noise ratio $r = W/V$ (red μ_t; black Y_t).

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

$$Y_{t+k}|\mathcal{D}_t \sim N(m_t, Q_t(k)) \quad , \quad k \geq 1\,, \tag{3.8}$$

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 $f_t(k) = E(Y_{t,k}|\mathcal{D}_t) = m_t$ 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 farer away.

The random walk plus noise model has an interesting behavior for $t \to \infty$. Note that $K_t = (C_{t-1} + W)Q_t^{-1}$ does not depend on the value of the observations $Y_1, \ldots, Y_t$ but only on t; therefore, exploiting also the fact that V and W are constant, one can study the limit of K_t as $t \to \infty$. It can be proved that

$$\lim_{t\to\infty} K_t = \frac{r}{2}\left(\sqrt{1 + \frac{4}{r}} - 1\right) = K \tag{3.9}$$

(see West and Harrison; 1997, Theorem 2.3). A first implication of this result is that, for t large enough, $C_t \approx KV$. 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 (3.8),

$$f_{t+1} = E(Y_{t+1}|\mathcal{D}_t) = m_t = m_{t-1} + K_t(Y_t - m_{t-1}) = m_{t-1} + K_t e_t\,.$$

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

$$f_{t+1} = m_{t-1} + Ke_t. \tag{3.10}$$

A forecast function of the kind (3.10) is used in many popular models for time series. It corresponds to Holt point predictor, see equation (3.2).

In 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 $K_{t-1} \approx K$,

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

where the forecast errors are a white noise sequence (see Chapter 2, page 66). Therefore, (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 1986[1]. 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 `dlmModPoly` 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 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.

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

[1] Source: `http://www-personal.buseco.monash.edu.au/~hyndman/TSDL` or Hipel and Mcleod (1994)

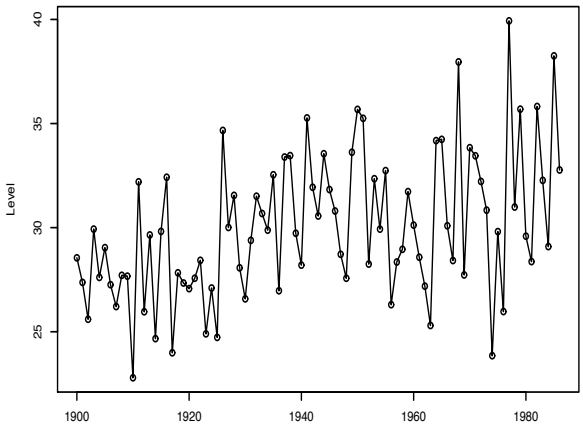

Fig. 3.2. Annual precipitation at Lake Superior

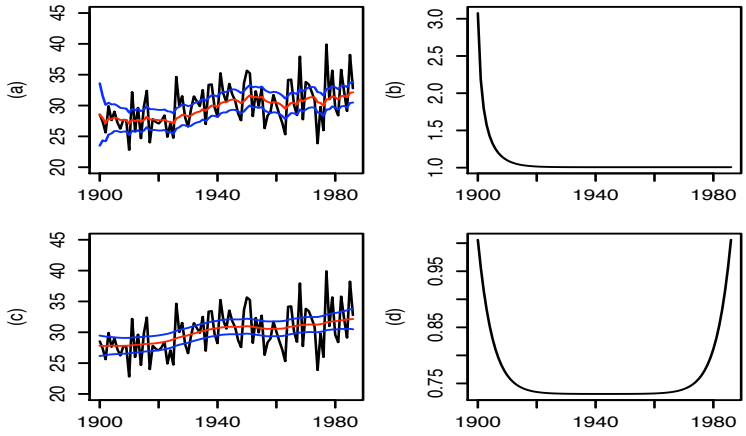

Fig. 3.3. (a): Filtered state estimates m_t with 90% confidence intervals; (b): Square root of filtering variances C_t (c): Smoothed state estimates s_t with 90% confidence intervals;(d): Square root of smoothing variances S_t

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.

Formal statistical tests may also be employed to assess model assumptions via the implied properties of the innovations. For example, Shapiro-Wilk test can be used to test the standardized innovations for normality. It is avail-

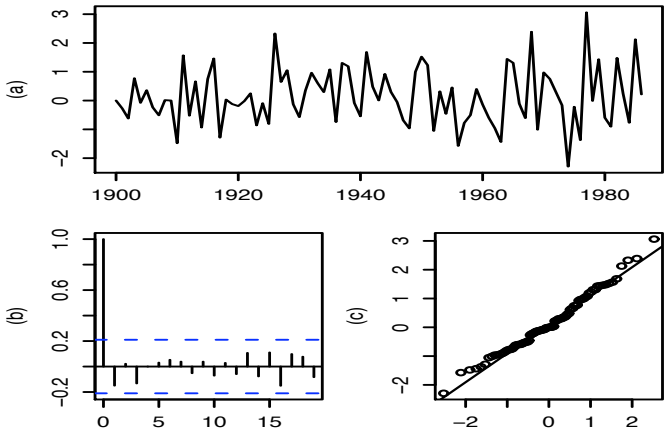

Fig. 3.4. (a): Standardized one-step-ahead forecast errors; (b): ACF of one-step-ahead forecast errors (c): Normal probability plot of standardized one-step-ahead forecast errors

able 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. Shapiro-Wilk normality test is commonly preferred to 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 and nortest. 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 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

$$\hat{\rho}(j) = \sum_{t=1}^{n-j}(\tilde{e}_t - \bar{\tilde{e}})(\tilde{e}_{t+j} - \bar{\tilde{e}}) \Big/ \sum_{t=1}^{n}(\tilde{e}_t - \bar{\tilde{e}})^2, \qquad j = 1, 2, \ldots.$$

What Ljung-Box test effectively does is testing for the absence of serial correlation up to lag k. Using $k = 20$, the p-value of 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 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 illlustrates how to obtain in R the standardized innovations and perform Shapiro-Wilk and Ljung-Box tests.

R code

```
> lakeSup <- ts(read.table("Datasets/lakeSuperior.dat", skip = 3,
+                          colClasses = "numeric")[,2], start = 1900)
> modLSup <- dlmModPoly(1, dV = 9.465, dW = 0.121)
> lSupFilt <- dlmFilter(lakeSup, modLSup)
> res <- residuals(lSupFilt, sd=FALSE)
> shapiro.test(res)

        Shapiro-Wilk normality test

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

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

        Box-Ljung test

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

> sapply(1:20, function(i) Box.test(res, lag=i, type="Ljung-Box")$p.value)
 [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
[13] 0.9143456 0.9185912 0.8924318 0.7983241 0.7855680 0.7971489
[19] 0.8010898 0.8129607
```

Exponential smoothing methods for computing the one-step forecasts (3.3) are provided by the function *HoltWinters* of the *Stats* package. The results for the annual precipitations in Lake Superior are plotted in figure 3.5 (here the smoothing parameter is estimated as $\lambda = 0.09721$). We see that the steady model has, for large t, the same forecast function as the Holt forecasting procedure.

R code

```
> HWout=HoltWinters(lakeSup, gamma=0, beta=0)
> plot(window(lSupFilt$f, start=1901), type="l", lty =1, xlab="", ylab="")
```

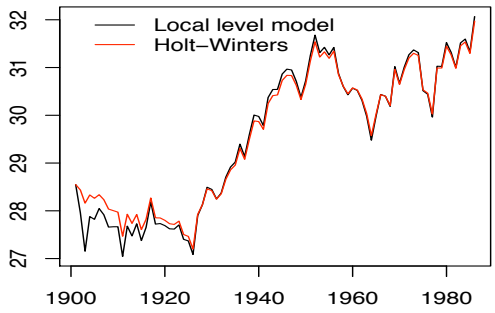

Fig. 3.5. One-step ahead forecasts

```
> lines(HWout$fitted[,1], lty=1, col="red")
> leg <- c("Local level model", "Holt-Winters")
> legend(1901, 32.3, legend=leg, lty = rep(1,2),col=c("black","red"), bty="n")
```

Linear growth model

The linear growth, or local linear trend model, is defined by (2.7). 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 $(Y_t, t = 1, \ldots, T)$, for varying 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

$$a_t = G m_{t-1} = \begin{bmatrix} \hat{\mu}_{t-1} + \hat{\beta}_{t-1} \\ \hat{\beta}_{t-1} \end{bmatrix} \tag{3.11}$$

$$f_t = F_t\, a_t = \hat{\mu}_{t-1} + \hat{\beta}_{t-1}, \tag{3.12}$$

$$m_t = \begin{bmatrix} \hat{\mu}_t \\ \hat{\beta}_t \end{bmatrix} = a_t + K_t e_t = \begin{bmatrix} \hat{\mu}_{t-1} + \hat{\beta}_{t-1} + k_{t1} e_t \\ \hat{\beta}_{t-1} + k_{t2} e_t \end{bmatrix}. \tag{3.13}$$

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.

Exploiting the fact that the variances are constant, we can study the limit behavior of the linear growth process as $t \to \infty$. It can be proved that the gain matrix K_t converges to a constant vector $K = (k_1, k_2)$ as $t \to \infty$ (see West and Harrison; 1997, Theorem 7.2) Therefore, the asymptotic updating formulae for the state vector are given by

$$\begin{aligned} \hat{\mu}_t &= \hat{\mu}_{t-1} + \hat{\beta}_{t-1} + k_1\, e_t \\ \hat{\beta}_t &= \hat{\beta}_{t-1} + k_2\, e_t \end{aligned} \tag{3.14}$$

Several popular point predictors methods use expressions of the form (3.14), such as the Holt and Winters exponential smoothing method (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 ARIMA(0,2,2) processes. 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

$$z_t = e_t + (-2 + k_{1,t-1} + k_{2,t-1})e_{t-1} + (1 - k_{1,t-2})e_{t-2} \tag{3.15}$$

(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 2000, plotted in Figure 3.6 (source: `http://www.fgn.unisg.ch/eumacro/macrodata/macroeconomic-time-series.html`). 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 appropriate for these data. We assume that the variances are known (they were actually estimated) and are as follows:

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

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

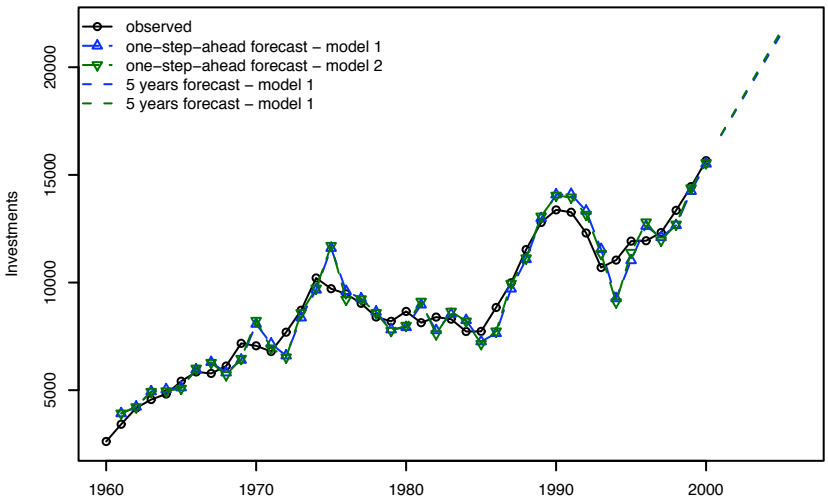

Fig. 3.6. Spain investments

model (an integrated random walk model, in fact, see page 85) 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))
> mod1Filt <- dlmFilter(invSpain, mod1)
> fut1 <- dlmForecast(mod1Filt, n = 5)
> mod2 <- dlmModPoly(dV = 10, dW = c(0, 515939))
> mod2Filt <- dlmFilter(invSpain, mod2)
> 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, that 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. A more formal comparison of the two models in terms of forecast accuracy can be done using the mean absolute deviation (MAD)

$$\text{MAD} = \frac{1}{n}\sum_{t=1}^{n}|e_t|$$

or the mean square error (MSE)

$$\text{MSE} = \frac{1}{n}\sum_{t=1}^{n}e_t^2.$$

Also very common is the mean absolute percentage error (MAPE)

$$\text{MAPE} = \frac{1}{n}\sum_{t=1}^{n}\frac{|e_t|}{Y_t}.$$

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$f - invSpain))
[1] 623.5682
> mean(abs(mod2Filt$f - invSpain))
[1] 610.2621
> mean((mod1Filt$f - invSpain)^2)
[1] 655480.6
> mean((mod2Filt$f - invSpain)^2)
[1] 665296.7
> mean(abs(mod1Filt$f - invSpain) / invSpain)
[1] 0.08894788
> mean(abs(mod2Filt$f - invSpain) / invSpain)
[1] 0.08810524
```

*n*th order polynomial model

The general nth order polynomial model has an n-dimensional state space and is described by the matrices

$$F = (1, 0, \ldots, 0) \tag{3.16}$$

$$G = \begin{bmatrix} 1 & 1 & 0 & & \ldots & 0 \\ 0 & 1 & 1 & 0 & \ldots & 0 \\ \vdots & & & & \ddots & \vdots \\ 0 & & \ldots & 0 & 1 & 1 \\ 0 & & \ldots & & 0 & 1 \end{bmatrix} \tag{3.17}$$

$$W = \text{diag}(W_1, \ldots, W_n). \tag{3.18}$$

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

$$\begin{cases} Y_t = \theta_{t,1} + v_t \\ \theta_{t,j} = \theta_{t-1,j} + \theta_{t-1,j+1} + w_{t,j} \quad j = 1, \ldots, n-1 \\ \theta_{t,n} = \theta_{t-1,n} + w_{t,n} \end{cases} \tag{3.19}$$

So, for $j = 2, \ldots, n$, the jth 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 $W_1 = \cdots = W_{n-1} = 0$ 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 nth 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, the integrated random walk model, having only one degree of freedom in the system noise, may be slower in adapting to random shocks to the state vector, which reflects in a lower accuracy in the forecasts.

3.2.2 Seasonal models

We presents two ways of modelling a time series which shows a cyclical behavior, or "seasonality": the seasonal factor model and the Fourier-form seasonal model.

Seasonal factor models

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

$$\begin{aligned} Y_{t-1} &= \alpha_1 + v_{t-1} \\ Y_t &= \alpha_2 + v_t \end{aligned} \tag{3.20}$$

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

$$Y_t = F\theta_t + v_t,$$

which corresponds to (3.20). The state equation must "rotate" θ_{t-1} into a 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 modelled through an s-dimensional state vector θ_t of seasonal deviations, by specifying a DLM with $F = (1, 0, \ldots, 0)$ and G given by a s by s permutation matrix. Identifiability constraints have to be imposed on the seasonal factors $\alpha_1, \ldots, \alpha_s$. 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. For the example given by (3.20), 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

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

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

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

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

Fourier form representation of seasonality

For modelling the cyclic behavior of a time series, rather than introducing s seasonal effects $(\alpha_1, \ldots, \alpha_s)$, a more parsimonious representation can be obtained by using periodic functions.

A seasonal time series $(Y_t, t = 1, 2 \ldots)$ might be represented as $Y_t = g(t-1) + v_t$, where $g(\cdot)$ is a periodic function. We need a model for the function $g(\cdot)$, which can be estimated from the data. Since any periodic function on the real line can be approximated by a sum of harmonic functions (Fourier sums)

$$g(t) \approx a_0 + \sum_{r=1}^{\nu} [a_r \cos(\omega r t) + b_r \sin(\omega r t)],$$

one can try to model g in this way. Moreover, when t is discrete, we have an exact representation of a periodic function g as a sum of harmonic functions. Let g be a periodic function on the set of nonnegative integers of period s, and let $\omega = 2\pi/s$. The behavior of g is completely determined by the values $g(0), \ldots, g(s-1)$; indeed, for $t = ks + j$ $(0 \leq j < s, k \geq o)$, we have that $g(t) = g(j)$. (For example, for monthly data with period $s = 12$, where $t = 0$ corresponds to January, say, $t = 1$ to February, etc., we have that $g(0) = g(12) = g(2 \cdot 12) = \cdots$, $g(1) = g(12+1) = g(2 \cdot 12 + 1) = \cdots$, and so on). Note the relationship with the seasonal factor $\alpha_1, \ldots, \alpha_s$ discussed in the previous section: in fact, we may let $g(0) = \alpha_1, \ldots, g(s-1) = \alpha_s$.

Given the values $g(0) = \alpha_1, \ldots, g(s-1) = \alpha_s$, we can write the system of s equations

$$g(j) = a_0 + \sum_{r=1}^{\nu} [a_r \cos(\omega r j) + b_r \sin(\omega r j)], \qquad j = 0, \ldots, s-1$$

in the $2\nu + 1$ unknown variables $(a_0, \ldots, a_\nu, b_1, \ldots, b_\nu)$. This system has a unique solution if the number of unknowns is equal to the number of equations, i.e. $2\nu + 1 = s$. This is true if $\nu = \lfloor s/2 \rfloor$, where $\lfloor x \rfloor$ denotes the largest integer less than or equal to x. Indeed, if s is odd, $\nu = \lfloor s/2 \rfloor = (s-1)/2$, so that $2\nu + 1 = s$. For s even, $\nu = s/2$, and we have $s + 1$ unknown coefficients. However, since $\sin(\omega \nu t) = 0$ for every t, b_ν is arbitrary (although it is common to set $b_\nu = 0$), so the number of unknown coefficients is s in this case, too. The constraint $\sum_{j=1}^{s} \alpha_j = 0$ is equivalent to $a_0 = 0$ in view of the trigonometric identities

$$\sum_{j=0}^{s-1} \cos(\omega r j) = 0 \qquad \text{and} \qquad \sum_{j=0}^{s-1} \cos(\omega r j) = 0.$$

In what follows we assume without loss of generality that $a_0 = 0$, i.e. we assume that we are modelling a purely seasonal, zero-mean, time series.

It may seem at first that the representation of seasonality by harmonic functions is more complicated than the use of seasonal factors. However, the

Fourier representation usually allows for a more parsimonious representation of real-world seasonal phenomena, compared to the one that uses seasonal factors. The *rth harmonic* of g, $a_r \cos(\omega rt) + b_r \sin(\omega rt)$, has period s/r. For seasonal time series having a smooth behavior, the high-frequency harmonics, i.e. those corresponding to large values of r, are typically negligible, so one can assume for g a Fourier representation truncated to $q < \nu$ harmonics.

Consider for example monthly data, with period $s = 12$. For $t = 0$, 12, $2 \times 12, \ldots$ (January, say), we have

$$Y_t = + \sum_{r=1}^{q} [a_r \cos(0) + b_r \sin(0)] + v_t = \sum_{r=0}^{q} a_r + v_t$$

and $\sum_{r=1}^{q} a_r = \alpha_1$, say. For $t = 1$, $12 + 1$, $2 \times 12 + 1, \ldots$ (February) we have

$$Y_t = \sum_{r=1}^{q} [a_r \cos(\omega r) + b_r \sin(\omega r)] + v_t = \alpha_2 + v_t.$$

In general, for $t = 12k + j$ $(k = 0, 1, \ldots)$, we have

$$Y_t = \sum_{r=1}^{q} [a_r \cos(\omega rj) + b_r \sin(\omega rj)] + v_t = \alpha_{j+1} + v_t. \tag{3.21}$$

The unknown parameters are the $2q$ parameters a_r, b_r, $r = 1, \ldots, q$ in the Fourier-type representation and the s parameters $\alpha_0, \ldots, \alpha_{s-1}$ in the seasonal factor model. In general, it is enough to consider $q = 1, 2$, so that the Fourier representation has 3 or 5 parameters and is more parsimonious.

We can write model (3.21) in a state-space form, as follows. Let F be a $1 \times 2q$ partitioned matrix

$$F = \left[1\ 0 \mid 1\ 0 \mid \ldots \mid 1\ 0\right],$$

and define the state vector at time $t = 0$ as

$$\theta_0 = \left[\psi_1' \mid \psi_2' \mid \ldots \mid \psi_q'\right],$$

where $\psi_r = (a_r, b_r)'$. Define the *harmonic matrix*

$$H_r = \begin{bmatrix} \cos(\omega r) & \sin(\omega r) \\ -\sin(\omega r) & \cos(\omega r) \end{bmatrix}, \qquad r = 1, \ldots, q,$$

and let G be the $2q$ by $2q$ block diagonal matrix

$$G = \begin{bmatrix} H_1 & 0 & & 0 \\ 0 & H_2 & & \\ \vdots & & \ddots & \vdots \\ 0 & & & H_q \end{bmatrix}.$$

Finally, let W be a $2q$ by $2q$ matrix of zeros. Then model (3.21) can be written as

$$Y_t = F\theta_t + v_t, \qquad v_t \sim \mathcal{N}(0, V_t) \tag{3.22}$$
$$\theta_t = G\theta_{t-1} + w_t, \qquad w_t \sim \mathcal{N}(0, W). \tag{3.23}$$

Indeed, we have

$$\theta_1 = G\,\theta_0 = \begin{pmatrix} H_1\psi_1 \\ \vdots \\ H_q\psi_q \end{pmatrix}$$

Then

$$Y_1 = F\theta_1 + v_1 = \begin{pmatrix} 1 & 0 & \dots & 1 & 0 \end{pmatrix} \begin{pmatrix} H_1\psi_1 \\ \vdots \\ H_q\psi_q \end{pmatrix} + v_1 = \sum_{r=1}^{q} (a_r \cos(\omega r) + b_r \sin(\omega r)) + v_1 \ .$$

Analogously, $\theta_2 = G\theta_1 = G^2\theta_0$. It can be shown that, for $j = 1, 2, \dots, s-1$

$$H_r^j = \begin{pmatrix} \cos(\omega r j) & \sin(\omega r j) \\ -\sin(\omega r j) & \cos(\omega r j) \end{pmatrix},$$

and $H_r^t = H_r^j$ for $t = kp + j$ $(k = 1, 2, \dots)$. Therefore

$$\theta_2 = G^2\theta_0 = \begin{pmatrix} H_1^2\psi_1 \\ \vdots \\ H_q^2\psi_q \end{pmatrix}$$

where

$$H_r^2 = \begin{pmatrix} \cos(\omega r 2) & \sin(\omega r 2) \\ -\sin(\omega r 2) & \cos(\omega r 2) \end{pmatrix},$$

so $Y_2 = F\theta_2 + v_2 = FG^2\theta_0 + v_2 = \sum_{r=1}^{q} (a_r \cos(\omega r 2) + b_r \cos(\omega r 2)) + v_2$, and so on.

As is the case with the representation of periodic components via seasonal factors, one may consider a dynamic version of the Fourier form representation by defining W to be a nontrivial variance matrix, typically a nonsingular diagonal matrix. While the seasonal component in this case is no longer periodic, the forecast function is, see Problem 3.8

We have noticed before that, in case s is even, the last harmonic depends effectively on the parameter a_ν only and b_ν can be set to any value, usually zero. In this case, if one needs to include all the harmonics in the representation of the seasonal DLM component, i.e. $q = \nu$, then for the last harmonic the following modifications are needed: $\psi_\nu = (a_\nu)$, $H_\nu = [-1]$, and the last block of F is composed by a '1' only. The system matrix G has the same block diagonal structure as before – although in this case the number of rows will be $2q - 1 = s - 1$ instead of $2q$.

The Fourier representation of periodic components described above can also be used to model cycles whose period is less obviously related to the frequency at which the observations are taken. In econometrics, for example, a common application of this type is the inclusion of a DLM component representing the business cycle. The period in this case is typically estimated. A DLM component for this kind of cycle can be set up exactly as described above, with ω replaced by ω_c, the appropriate frequency of the cycle. The corresponding period of the cycle is $\tau_c = 2\pi/\omega_c$. Strictly speaking, in this case one may need to sum an infinite number of harmonics to exactly represent a function of period τ_c. In practice, however, only a finite number q of them are used in the DLM, and $q = 1$ or $q = 2$ are not uncommon choices in many applications.

3.2.3 Regression models

One of the interesting aspects in the analysis of time series by DLM is the possibility of easily including explanatory variables in the model. For example, family expenses Y_t might depend on the income x_t according to the relationship

$$Y_t = \alpha_t x_t + v_t, \qquad v_t \sim \mathcal{N}(0, V_t).$$

The usual regression model is a special case where $\alpha_t = \alpha$ is constant over time. More generally, we might want to let the coefficient α_t change over time, introducing a state equation; for example

$$\alpha_t = \alpha_{t-1} + w_t, \qquad w_t \sim \mathcal{N}(0, W_t).$$

In general, the dynamic linear regression model is described by

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

where the X_t is the vector of explanatory variables for period t, $X_t = (x_{1,t}, \ldots, x_{r,t})$, which is assumed known at time t. When the elements of X_t have the same meaning across time (e.g., for any t, $x_{1,t}$ is the GDP of a specific country at time t), a common choice for the evolution matrix G_t is the identity matrix. The static regression linear model corresponds to the case where $W_t = 0$ for any t, so that $\theta_t = \theta$ is constant over time. This observation suggests that DLM techniques may be used to sequentially update the estimates of the parameters of a regression model as new observations become available.

3.2.4 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 (3.5), 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

$$\begin{aligned} F &= \begin{bmatrix} 1 & 0 & \dots & 0 \end{bmatrix}, \\ G &= \begin{bmatrix} \phi_1 & 1 & 0 & \dots & 0 \\ \phi_2 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & & \ddots & \\ \phi_{r-1} & 0 & \dots & 0 & 1 \\ \phi_r & 0 & \dots & 0 & 0 \end{bmatrix}, \\ R &= \begin{bmatrix} 1 & \psi_1 & \dots & \psi_{r-2} & \psi_{r-1} \end{bmatrix}'. \end{aligned} \tag{3.24}$$

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:

$$\begin{cases} Y_t = F\theta_t, \\ \theta_{t+1} = G\theta_t + R\epsilon_t. \end{cases} \tag{3.25}$$

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

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

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

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

Recalling that $r = \max\{p, q+1\}$ and $y_t = \theta_{1,t}$ we see that this is the ARMA model (3.5).

Consider for example the AR(2) model

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

Here $r = p = 2$ and the matrices defining the DLM representation are:

$$\begin{aligned} F &= \begin{bmatrix} 1 & 0 \end{bmatrix}, \qquad V = 0, \\ G &= \begin{bmatrix} \phi_1 & 1 \\ \phi_2 & 0 \end{bmatrix}, \quad W = \begin{bmatrix} \sigma^2 & 0 \\ 0 & 0 \end{bmatrix}. \end{aligned} \tag{3.27}$$

On the other hand, for the ARMA(1,1) model

$$Y_t = \phi_1 Y_{t-1} + \epsilon_t + \psi_1\epsilon_{t-1}, \qquad \epsilon_t \sim \mathcal{N}(0, \sigma^2), \tag{3.28}$$

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

$$\begin{aligned} F &= \begin{bmatrix} 1 & 0 \end{bmatrix}, \qquad V = 0, \\ G &= \begin{bmatrix} \phi_1 & 1 \\ 0 & 0 \end{bmatrix}, \quad W = \begin{bmatrix} 1 & \psi_1 \\ \psi_1 & \psi_1^2 \end{bmatrix} \sigma^2. \end{aligned} \tag{3.29}$$

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 case 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 ARIMA 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

$$\begin{cases} Y_t = \begin{bmatrix} 1 & 1 & 0 \end{bmatrix} \theta_{t-1}, \\ \theta_t = \begin{bmatrix} 1 & 1 & 0 \\ 0 & \phi_1 & 0 \\ 0 & \phi_2 & 1 \end{bmatrix} \theta_{t-1} + w_t, \qquad w_t \sim \mathcal{N}(0, W), \end{cases} \tag{3.30}$$

with

$$\theta_t = \begin{bmatrix} Y_{t-1} \\ Y_t^* \\ \phi_2 Y_{t-1}^* \end{bmatrix} \tag{3.31}$$

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

$$\Delta^{d-j} Y_t = Y_t^* + \sum_{i=1}^{j} \Delta^{d-i} Y_{t-1}, \qquad j = 1, \ldots, d. \tag{3.32}$$

Define the state vector as follows:

$$\theta_t = \begin{bmatrix} Y_{t-1} \\ \Delta Y_{t-1} \\ \vdots \\ \Delta^{d-1} Y_{t-1} \\ Y_t^* \\ \phi_2 Y_{t-1}^* + \cdots + \phi_r Y_{t-r+1}^* + \psi_1 \epsilon_t = \cdots + \psi_{r-1} \epsilon_{t-r+2} \\ \phi_3 Y_{t-1}^* + \cdots + \phi_r Y_{t-r+2}^* + \psi_2 \epsilon_t = \cdots + \psi_{r-1} \epsilon_{t-r+3} \\ \vdots \\ \phi_r Y_{t-1}^* + \psi_{r-1} \epsilon_t \end{bmatrix} \tag{3.33}$$

Note that the definition of the last components of θ_t follows from formula (3.26). The system and observation matrices, together with the system variance are defined by

$$\begin{aligned} F &= \begin{bmatrix} 1\ 1 \ldots 1\ 0 \ldots 0 \end{bmatrix}, \\ G &= \begin{bmatrix} 1 & 1 & \ldots & 1 & 0 \ldots\ldots 0 \\ 0 & 1 & \ldots & 1 & 0 \ldots\ldots 0 \\ \ldots & \ldots & \ldots & \ldots & \ldots\ldots \\ \ldots & 1 & & 1 & 0 \ldots\ldots 0 \\ 0 \ldots & 0 & \phi_1 & 1\ 0 & \ldots 0 \\ \ldots\ldots & & \phi_2 & 0\ 1 & \ldots 0 \\ \vdots & & \vdots & \vdots & \ddots \\ \ldots\ldots & & \phi_{r-1} & 0 \ldots & 0\ 1 \\ 0 \ldots & 0 & \phi_r & 0 \ldots & 0\ 0 \end{bmatrix}, \\ R &= \begin{bmatrix} 0 \ldots 0\ 1\ \psi_1 \ldots \psi_{r-2}\ \psi_{r-1} \end{bmatrix}', \\ W &= RR'\sigma^2. \end{aligned} \tag{3.34}$$

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

$$\begin{cases} Y_t = F\theta_t, \\ \theta_t = G\theta_{t-1} + w_t, \qquad w_t \sim \mathcal{N}(0, W). \end{cases} \tag{3.35}$$

Since in DLM modelling 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.

3.2.5 Combining component models: examples

In the previous sections we have presented some common models for the different components (trend, seasonality, regression) of a time series. These models can be used as "building blocks" for constructing a DLM for a time series with a more complex behavior. The additive structure of the DLM allows to easily combine the different component models, as discussed at the beginning of this section.

Suppose, for example, that a series Y_t is the sum of a trend component $Y_{L,t}$ and a seasonal component $Y_{S,t}$:

$$Y_t = Y_{L,t} + Y_{S,t} + v_t.$$

We can construct a DLM for each component, so that

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

and

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

Define F_t and θ_t as partitioned matrices

$$F_t = \begin{pmatrix} F_{L,t} & F_{S,t} \end{pmatrix}, \quad \theta_t = \begin{pmatrix} \theta_{L,t} \\ \theta_{S,t} \end{pmatrix}$$

and G and W_t as block-diagonal matrices

$$G_t = \begin{pmatrix} G_{L,t} & \\ & G_{S,t} \end{pmatrix}, \quad W_t = \begin{pmatrix} W_{L,t} & \\ & W_{S,t} \end{pmatrix} .$$

Then Y_t is described by a DLM with observation equation

$$Y_t = F_t\theta_t + v_t = F_{L,t}\theta_{L,t} + F_{S,t}\theta_{S,t} + v_t, \quad v_t \sim \mathcal{N}(0, V_t)$$

and state equation

$$\theta_t = G_t\theta_{t-1} + w_t, \qquad w_t \sim \mathcal{N}(0, W_t) .$$

In particular, a DLM obtained by combining a linear growth model and a seasonal component, either in the form of a seasonal factor model or a Fourier form DLM, is known in the econometric literature as *Basic Structural Model*, see Harvey (1989).

Example

Let (Y_t) be a univariate time series. A linear growth model with seasonality for (Y_t) can be constructed as follows. The trend component is described by a linear growth model, with

$$F_{L,t} = (1,0), \quad G_{L,t} = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}, \quad \theta_{L,t} = (\mu_t, \beta_t)' \ .$$

The seasonal component is described by introducing a harmonic component, so by a DLM with

$$F_{S,t} = (1,0), \quad G_{S,t} = \begin{pmatrix} \cos(\omega) & \sin(\omega) \\ -\sin(\omega) & \cos(\omega) \end{pmatrix}$$

and the general model is obtained as illustrated above.

3.3 Models for multivariate time series

Modeling multivariate time series is of course more interesting - and more challenging - than studying univariate models, and again DLM offer a very flexible framework for the analysis. In this section we present some basic problems and models for multivariate time series, which of course represent only some examples of the extremely large variety of applications of DLM for multivariate data.

We will consider two basic types of data and problems in studying multivariate time series. In many applications, one has data $Y_t = (Y_{1,t}, \ldots, Y_{m,t})'$ 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, $i = 1, \ldots, m$, etc. In these cases, the focus of interest is typically understanding the correlation structure among the time series, investigating the possible presence of clusters etc. These aspects might be of interest in themselves, or for improving the predictive ability of the model.

In other contexts, the data are observations on one or more variables of interest Y and on some explanatory variables $X_1, \ldots, X_k$. For example, Y could be the inflation rate and X_i relevant macroeconomic variables for a country. We have again a multivariate time series $(Y_t, X_{1,t}, \ldots, X_{k,t})$, but now the emphasis is on explaining or predicting the variable of interest Y_t by means of the explanatory variables $X_{i,t}$, so we are more in a regression framework. Note

that in the regression DLM discussed in section...., the covariates were deterministic (control variables) while here $X_{1,t}, \ldots, X_{k,t}$ are random variables. Of course by a joint model for $(Y_t, X_{1,t}, \ldots, X_{k,t})$ one can also study feedbacks effects and causality relations among all variables.

3.3.1 Time series of cross sectional data

Consider a multivariate time series $Y_t = (Y_{1,t}, \ldots, Y_{m,t})$ where the $Y_{i,t}$'s are observations of a variable Y for m different units. Of course, the simplest approach would be to study the m series independently, specifying a univariate model for each of them

$$\begin{aligned} y_{i,t} &= F_i\theta_{i,t} + v_{i,t}\,,\ v_{i,t} \sim \mathcal{N}(0, V_i) \\ \theta_{i,t} &= G_i\theta_{i,t-1} + w_{i,t}\,,\ w_{i,t} \sim \mathcal{N}_p(0, W_i), \end{aligned}$$

$i = 1, \ldots, m$ (we take F_i and G_i as time-invariant just for brevity). This approach might give fairly good forecasts for each time series, but in predicting $Y_{i,t+1}$ say, it doesn't exploit the information provided by the similar time series $Y_{j,t}$, $j \neq i$. For using all the available information, clearly we want a joint model for the m-variate process $(Y_{1,t}, \ldots, Y_{m,t})$, that is we want to introduce dependence across the time series.

With this kind of data, it can be reasonable to assume that the m time series can be modeled using the "same" DLM, possibly with different variances but with the same time-invariant system and observation matrices G and F; that is

$$\begin{aligned} y_{i,t} &= F\theta_{i,t} + v_{i,t}\,,\ v_{i,t} \sim \mathcal{N}(0, V_i) \\ \theta_{i,t} &= G\theta_{i,t-1} + w_{i,t}\,,\ w_{i,t} \sim \mathcal{N}_p(0, W_i), \end{aligned} \tag{3.36}$$

$i = 1, \ldots, m$. 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 $(Y_{i,t})$. For simplicity, suppose for the moment that the variances V_i and W_i are known. Thus we are modeling the processes $(Y_{i,t})_{t\geq 1}$, $i = 1, \ldots m$, as conditionally independent given the state processes, with $(Y_{i,t})$ depending only on "its" $(\theta_{i,t})$; in particular

$$Y_{1,t}, \ldots, Y_{m,t} \mid \theta_{1,t}, \ldots, \theta_{m,t} \sim \prod_{i=1}^{m} \mathcal{N}(y_{i,t} \mid F\theta_{i,t}, V_i).$$

Note that this assumption is similar to the framework of Section 1.1.2. A dependence among $Y_{1,t}, \ldots, Y_{m,t}$ can be introduced through the joint probability law of $\theta_{1,t}, \ldots, \theta_{m,t}$. If $\theta_{1,t}, \ldots, \theta_{m,t}$ are independent, then the $Y_{i,t}$ for $i = 1, \ldots, m$ are independent; inference on $(\theta_{i,t})$ only depends on $(Y_{i,t})$. Otherwise, the dependence structure of $\theta_{1,t}, \ldots, \theta_{m,t}$ will be reflected in the dependence across the time series $(Y_{i,t})$. Examples are provided in the next two sections.

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 $\theta_{1,t}, \ldots, \theta_{m,t}$ as follows. As we said, the model (3.36) 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 to describe the joint evolution of the state vectors by grouping together all the levels and then all the slopes in an overall state vector $\theta_t = (\mu_{t1}, \ldots, \mu_{tm}, \beta_{t1}, \ldots, \beta_{tm})'$. 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 by m block accounting for the correlation among levels and a second m by m block accounting for the correlation among slopes. To be specific, suppose one has $m = 2$ series. Then $\theta_t = (\mu_{t1}, \mu_{t2}, \beta_{t1}, \beta_{t2})'$ and the system equation is

$$\begin{bmatrix} \mu_{t1} \\ \mu_{t2} \\ \beta_{t1} \\ \beta_{t2} \end{bmatrix} = \begin{bmatrix} 1\ 0\ 1\ 0 \\ 0\ 1\ 0\ 1 \\ 0\ 0\ 1\ 0 \\ 0\ 0\ 0\ 1 \end{bmatrix} \begin{bmatrix} \mu_{t-1,1} \\ \mu_{t-1,2} \\ \beta_{t-1,1} \\ \beta_{t-1,2} \end{bmatrix} + \begin{bmatrix} w_{t1} \\ w_{t2} \\ w_{t3} \\ w_{t4} \end{bmatrix}, \tag{3.37a}$$

where $(w_{t1}, w_{t2}, w_{t3}, w_{t4})' \sim \mathcal{N}(0, W)$ and

$$W = \left[\begin{array}{cc|cc} & W_\mu & 0 & 0 \\ & & 0 & 0 \\ \hline 0 & 0 & & \\ 0 & 0 & & W_\beta \end{array}\right]. \tag{3.37b}$$

The observation equation for the bivariate time series $\big((y_{t1}, y_{t2}) : t \geq 1\big)$ is

$$\begin{bmatrix} Y_{t1} \\ Y_{t2} \end{bmatrix} = \begin{bmatrix} 1\ 0\ 0\ 0 \\ 0\ 1\ 0\ 0 \end{bmatrix} \theta_t + \begin{bmatrix} v_{t1} \\ v_{t2} \end{bmatrix}, \tag{3.37c}$$

with $(v_{t1}, v_{t2})' \sim \mathcal{N}(0, V)$. 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 ith component of Y_t follows the DLM

$$\begin{aligned} Y_{ti} &= F\theta_t^{(i)} + v_{ti}, \\ \theta_t^{(i)} &= G\theta_{t-1}^{(i)} + w_t^{(i)}, \end{aligned} \tag{3.38}$$

with $\theta_t^{(i)} = (\theta_{t1}^{(i)}, \ldots, \theta_{tp}^{(i)})'$ for $i = 1, \ldots, m$. Then a SUTSE model for (Y_t) has the form

$$\begin{cases} Y_t = (F \otimes I_m)\,\theta_t + v_t, & v_t \sim \mathcal{N}(0, V), \\ \theta_t = (G \otimes I_m)\,\theta_{t-1} + w_t, & w_t \sim \mathcal{N}(0, W), \end{cases} \tag{3.39}$$

with $\theta_t = (\theta_{t1}^{(1)}, \theta_{t1}^{(2)}, \ldots, \theta_{tp}^{(m-1)}, \theta_{tp}^{(m)})'$. 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 $Y_{t+k,i}$ are based only on the distribution of $\theta_t^{(i)}$ given $\mathcal{D}_t$.

Example — Annual Denmark and Spain investments

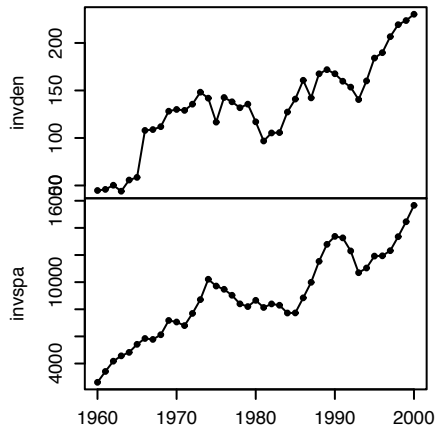

Fig. 3.7. Denmark investment and Spain investment

Figure 3.7 shows the annual investment in Denmark and Spain from 1960 to 2000. From visual inspection it appears that the two series display the same

type of qualitative behavior, that can be modeled by a linear growth DLM. This is the model we used on page 82 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 (3.37a)-(3.37c). 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 (3.37b) $W_\mu = 0$. The MLE estimates of the remaining parameters are

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

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 (3.39), using Kronecker products (lines 4 and 5). 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.

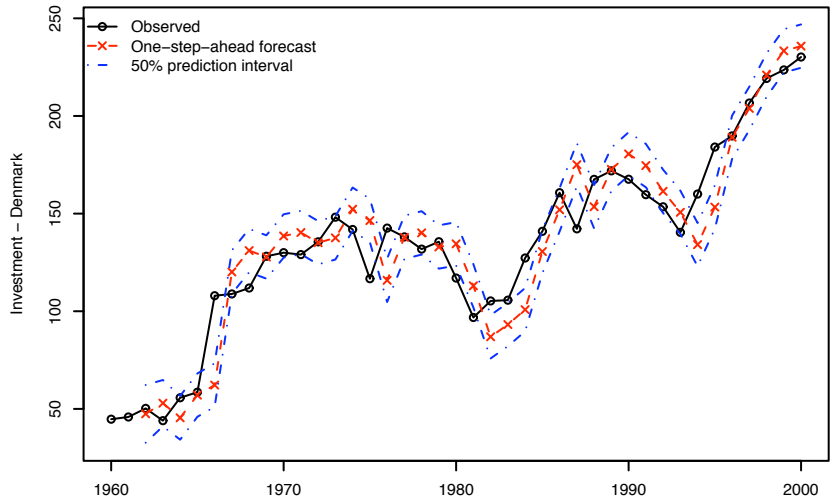

Fig. 3.8. Denmark investment and Spain investment

R code

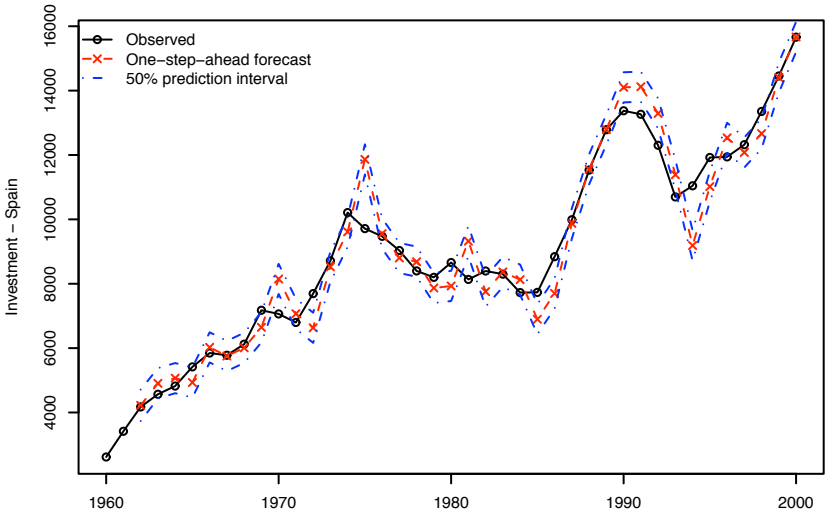

Fig. 3.9. Denmark investment and Spain investment

```
> invest <- ts(matrix(scan("Datasets/invest2.dat"), nc = 2, byrow = TRUE),
+              start = 1960, names = c("Denmark", "Spain"))
> mod <- dlmModPoly(2)
> mod$FF <- mod$FF %x% diag(2)
> mod$GG <- mod$GG %x% diag(2)
> W1 <- matrix(0,2,2)
> W2 <- diag(c(49, 437266))
> W2[1,2] <- W2[2,1] <- 155
> mod$W <- bdiag(W1, W2)
> V <- diag(c(72, 14353))
> V[1,2] <- V[2,1] <- 1018
> mod$V <- V
> mod$m0 <- rep(0,4)
> mod$C0 <- diag(4) * 1e7
> investFilt <- dlmFilter(invest, mod)
> sdev <- residuals(investFilt)$sd
> lwr <- investFilt$f + qnorm(0.25) * sdev
> 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 3.8 and 3.9. Note that conditionally on $\mathcal{D}_{t-1}$, Y_t and e_t have the same variance, see Section 2.5. This justifies the use of the innovation variances in lieu of the one-step-ahead observation forecast variances, line 16.

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.

Example — Capital Asset Pricing Model

In the Capital Asset Pricing Model (CAPM), an important asset pricing tool in financial econometrics, one fits a linear model for the returns on a set of assets in a small portfolio using the overall market return as a covariate. This allows to study the behavior, in terms of risk and expected returns, of individual assets compared to the market as a whole. The reader interested in a deeper treatment from a financial standpoint can consult Campbell et al. (1996), where additional references can be found; here we will illustrate a dynamic version of the classical CAPM. Let $r_t = (r_{t1}, \ldots, r_{tm})'$ be the vector of returns on m assets during period t, and let r_t^M and r_t^f be the market return and the return on a risk-free asset, respectively. Define the vector of *excess returns* on the m assets as

$$y_t = \begin{bmatrix} r_{t1} - r_t^f \\ \vdots \\ r_{tm} - r_t^f \end{bmatrix}.$$

Similarly, the excess market return is defined to be $x_t = r_t^M - r_t^f$. The classical, static, CAPM postulates that y_t follows the linear model

$$y_t = \alpha + \beta x_t + \epsilon_t,$$

where α and β are m-dimensional vectors. β_i measures the sensitivity of asset i to movements of the market. A β_i greater than one suggests that the asset tends to magnify changes in the overall market return. Assets whose β_i's are greater than one are considered aggressive investments, while those whose β_i is less than one are considered conservative investments. It seems natural to allow the β_i's to vary in time. We can use a SUTSE type of model to describe the phenomenon, assuming that the individual series (α_{ti}) and (β_{ti}) have a random walk distribution. In terms of its components, we can write the model as:

$$\begin{aligned} y_{ti} &= \alpha_{ti} + \beta_{ti} x_t + v_{ti}, \\ \alpha_{ti} &= \alpha_{t-1,i} + w_{t1}^{(i)}, \\ \beta_{ti} &= \beta_{t-1,i} + w_{t2}^{(i)} \end{aligned}$$

More concisely, in the usual DLM notation, the model can be written in the form

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

with $y_t = \begin{bmatrix} y_{t1} \\ \vdots \\ y_{tm} \end{bmatrix}$, $\theta_t = \begin{bmatrix} \alpha_{t1} \\ \vdots \\ \alpha_{tm} \\ \beta_{t1} \\ \vdots \\ \beta_{tm} \end{bmatrix}$, $v_t = \begin{bmatrix} v_{t1} \\ \vdots \\ v_{tm} \end{bmatrix}$, $w_t = \begin{bmatrix} w_{t1} \\ \vdots \\ w_{t,2m} \end{bmatrix}$,

$F_t = \begin{bmatrix} 1 & x_t \end{bmatrix}$, $G = I_2$, $W = \text{blockdiag}(W_\alpha, W_\beta)$.

The data we are going to analyze for the present example are monthly returns from January 1978 to December 1987 on the stock of Mobil, IBM, Weyer, and Citicorp. In addition, we will use 30-day Treasury Bill as a proxy for the risk-free asset, and a value-weighted composite monthly market return based on all stocks listed at the New York and American Stock Exchanges to represent the overall market return. We assume for simplicity that the α_{ti} 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:

$$
V = \begin{bmatrix} 41.06 & 0.01571 & -0.9504 & -2.328 \\ 0.01571 & 24.23 & 5.783 & 3.376 \\ -0.9504 & 5.783 & 39.2 & 8.145 \\ -2.328 & 3.376 & 8.145 & 39.29 \end{bmatrix},
$$

$$
W_\beta = \begin{bmatrix} 8.153 \cdot 10^{-7} & -3.172 \cdot 10^{-5} & -4.267 \cdot 10^{-5} & -6.649 \cdot 10^{-5} \\ -3.172 \cdot 10^{-5} & 0.001377 & 0.001852 & 0.002884 \\ -4.267 \cdot 10^{-5} & 0.001852 & 0.002498 & 0.003884 \\ -6.649 \cdot 10^{-5} & 0.002884 & 0.003884 & 0.006057 \end{bmatrix}.
$$

Smoothing estimates of the β_{ti}'s, shown in Figure 3.10, can be obtained using the code below.

R code

```
> tmp <- 100 * ts(read.table("Datasets/capm.dat", header=T),
+            start=c(1978,1), frequency=12)
> y <- tmp[,1:4] - tmp[,"RKFREE"]; colnames(y) <- colnames(tmp)[1:4]
> market <- tmp[,"MARKET"] - tmp[,"RKFREE"]
> rm("tmp")
> m <- NCOL(y)
> CAPM <- dlmModReg(market)
> CAPM$FF <- CAPM$FF %x% diag(m)
> CAPM$GG <- CAPM$GG %x% diag(m)
> CAPM$JFF <- CAPM$JFF %x% diag(m)
> CAPM$W <- CAPM$W %x% matrix(0,m,m)
```

```
> CAPM$W[-(1:m),-(1:m)] <- c(8.153e-07,  -3.172e-05, -4.267e-05, -6.649e-05,
+                            -3.172e-05, 0.001377,   0.001852,   0.002884,
+                            -4.267e-05, 0.001852,   0.002498,   0.003884,
+                            -6.649e-05, 0.002884,   0.003884,   0.006057)
> CAPM$V <- CAPM$V %x% matrix(0,m,m)
> CAPM$V[] <- c(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)
> CAPM$m0 <- rep(0,2 * m)
> CAPM$C0 <- diag(1e7, nr = 2 * m)
> CAPMfilt <- dlmFilter(y, CAPM)
> CAPMsmooth <- dlmSmooth(CAPMfilt)
> plot(window(CAPMsmooth$s[,1:m + m], start=start(y)),
+      plot.type='s', col=1 + 1:4, xlab="", ylab="Beta")
> abline(h=1, lty=2)
> legend("bottomright", lty=1, col=1 + 1:4, legend=colnames(y), bty='n')
```

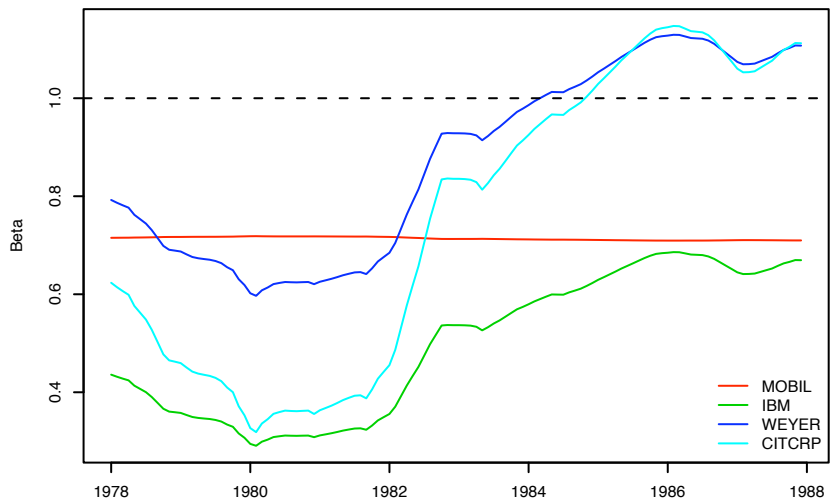

Fig. 3.10. Estimated beta's for four stocks

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 3.10 how the estimated beta's for the different stocks move in a clearly correlated fashion

(with the 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 time series of cross-sectional data (including panel data and longitudinal studies) is given by the so called *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-stages hierarchical DLM is specified as follows

$$\begin{aligned} Y_t &= F_{y,t}\theta_t + v_t, \quad v_t \sim \mathcal{N}_n(0, V_{y,t}); \\ \theta_t &= F_{\theta,t}\lambda_t + \epsilon_t, \quad \epsilon_t \sim \mathcal{N}_{p_1}(0, V_{\theta,t}) \\ \lambda_t &= G_t\lambda_{t-1} + w_t \quad w_t \sim \mathcal{N}_p(0, W_t), \end{aligned} \tag{3.40}$$

where the disturbance sequences $(v_t), (\epsilon_t), (w_t)$ are independent, and the matrices $F_{j,t}$ are of full rank. Again, we assume that the matrices $F_{y,t}$, $F_{\theta,t}$, G_t and the covariance matrices are known, but the more realistic case where they contain unknown parameters will be studied in chapter 4. Thus, in a two-stages 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 $p_1 > p$.

Example. Hierarchical models can be used for allowing random effects in DLM for multivariate time series. Suppose that $Y_t = (Y_{1,t}, \ldots, Y_{n,t})'$ are observations of a variable Y for n units at time t, and $Y_{i,t}$ is modeled as

$$Y_{i,t} = F_{1,t}\theta_{i,t} + v_{i,t}, \quad v_t \sim \mathcal{N}(0, v_{i,t}), i = 1, \ldots, n.$$

(more generally, $Y_{i,t}$ may be a multivariate time series). The observation equation for the n time series can be expressed as in (3.40), with $\theta_t = (\theta_{1,t}, \ldots, \theta_{n,t})'$, $F_{y,t}$ block-diagonal with blocks $F_{1,t}$, $\epsilon_t = (\epsilon_{1,t}, \ldots, \epsilon_{n,t})$, $V_{y,t}$ diagonal with elements $(v_{1,t}, \ldots, v_{n,t})$.

For the state vectors, assume that

$$\begin{aligned} \theta_{i,t} &= F_{2,t}\lambda_t + \epsilon_{i,t}, \quad \epsilon_{i,t} \sim \mathcal{N}(0, V_t), \text{ independent for } i = 1, \ldots, n \\ \lambda_t &= G\lambda_{t-1} + w_t, \quad w_t \sim \mathcal{N}_p(0, W_t), \end{aligned} \tag{3.41}$$

which can be easily written in the form (3.40). This specification corresponds to an assumption of exchangeability for the state vectors $(\theta_{i,t}), i \geq 1$ at a given time t, that is, the cross-sectional state vectors $\theta_{1,t}, \theta_{2,t}, \ldots$ are conditionally i.i.d. given λ_t, with common distribution $\mathcal{N}(F_{2,t}\lambda_t, V_t)$. In other words, we assume the same observation equation for the individual time series $Y_{i,t}$, allowing however random effects in the individual state processes.

As a simple example, consider the model

$$
\begin{aligned}
Y_{i,t} &= \theta_{i,t} + v_{i,t}, \quad v_{i,t} \sim \mathcal{N}(0, V_{y,t}) \text{independent for } i = 1, \ldots, n \\
\theta_{i,t} &= \lambda_t + \epsilon_{i,t}, \quad \epsilon_{i,t} \sim \mathcal{N}(0, V_{\theta,t}) \\
\lambda_t &= \lambda_{t-1} + w_t, \quad w_t \sim \mathcal{N}(0, W_t),
\end{aligned}
$$

which can be used for modeling measurements over time for a collections of units, with individual random effects. Another example are dynamic regression models with random effects. Consider

$$Y_{i,t} = x'_{i,t}\theta_{i,t} + v_{i,t} \quad \text{independent for } i = 1, \ldots, n.$$

Here, $Y_{i,t}$ are individual response variables, explained by the same regressors $X_1, \ldots, X_p$ with known value $x_{i,t}$ 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, i.e.

$$\theta_{i,t} \mid \lambda_t \sim \mathcal{N}_p(\lambda_t, V), \quad \text{independent for } i = 1, \ldots, n.$$

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 can add a further level in the model, obtaining a three-stage hierarchical DLM. Recursive formulae for filtering and prediction for hierarchical DLM are given in Gamerman and Migon (1993). Landim and Gamerman (2000) present further extensions to multivariate time series.

3.3.5 Mixtures of DLMs

In some applications, especially when we have data on a variable Y for a large number m of units, it is of interest to explore the presence of *clusters* among the time series. In general, a basic inferential tool for cluster analysis are mixture models, and for time series one might think of using a mixture of DLM. In Chapter 5, we will give a more detailed analysis of mixtures of DLM, from a Bayesian nonparametric approach; here we only give the basic ideas. We consider a DLM of the kind (3.36), where for brevity we let the time series $(Y_{i,t})$ have the same known variance $V_i = V$. Thus we assume for the moment that individual random effects act on the state process only.

In many applications it is useful to think that there are k clusters, or "species", in the population of the time series. Cluster j is characterized by a state process $(\theta^*_{j,t})$ and $(p_1, \ldots, p_k)$, with $p_j \geq 0$ and $\sum_{j=1}^k p_j = 1$, are the proportions of the k species in the population. We assume that the "typical paths" $(\theta^*_{j,t})$ are i.i.d. Markov processes, with common probability law described by the state equation

$$\theta^*_{j,t} = G\theta^*_{j,t-1} + w_{j,t};, \; w_{j,t} \sim \mathcal{N}_p(0, W), \tag{3.42}$$

with $\theta^*_{j,0} \sim N(m_0, C_0)$ (for simplicity, we are assuming a common covariance matrix W, but the model could be extended to the case of different W_j). Time series in cluster j are described by the DLM

$$
\begin{aligned}
y_{i,t} &= F\theta^*_{j,t} + v_{i,t},\ v_{i,t} \sim \mathcal{N}(0, V) \\
\theta^*_{j,t} &= G\theta^*_{j,t-1} + w_{j,t};,\ w_{j,t} \sim \mathcal{N}_p(0, W).
\end{aligned} \tag{3.43}
$$

It might help to think that there is some latent factor Z with values in $(1, \ldots, k)$ which induces the clusters. That is, if $Z_i = j$, then $(Y_{i,t})$ belongs to cluster j. Thus we can think of the data in terms of the following table

Unit	Observations	Latent States	
		factor	
1	$(Y_{1,t})$	Z_1	$(\theta_{1,t})$
$\vdots$	$\vdots$	$\vdots$	$\vdots$
i	$(Y_{i,t})$	Z_i	$(\theta_{i,t})$
$\vdots$	$\vdots$	$\vdots$	$\vdots$
m	$(Y_{m,t})$	Z_m	$(\theta_{m,t})$

Could we observe the latent factor Z, we would know the clustering structure of the time series $(Y_{i,t})$: if $Z_i = j$, then $(Y_{i,t})$ belongs to cluster j and it is described by the DLM (3.43). With a heuristic notation, let us write

$$
(Y_{i,t}) \mid Z_i = j \sim DLM((\theta_{j,t})^*).
$$

In fact, the latent factor Z is not observable and we assume that, given the weights $(p_1, \ldots, p_k)$,

$$
Z_i \sim \left\{ \begin{array}{ccc} 1 & \cdots & k \\ p_1 & \cdots & p_k. \end{array} \right.
$$

Thus, again using an informal notation

$$
(Y_{i,t}) \sim \sum_{j=1}^{k} Pr((y_{i,t}) \mid Z_j = j) Pr(Z_i = j) = \sum_{j=1}^{k} p_j DLM((\theta^*_{j,t})),
$$

a mixture of DLM, with mixing weights $(p_1, \ldots, p_k)$.

This model provides a possible way of introducing dependence across the time series in the framework of (3.39), focussed on clustering and shrinkage estimation. In fact, from the previous assumptions it follows that each individual state process $(\theta_{i,t})$ in (3.36) can be equal to one of the "typical paths" $(\theta^*_{1,t}), \ldots, (\theta^*_{k,t})$; more precisely we have $(\theta_{i,t}) = (\theta^*_{j,t})$ with probability p_j, $j = 1, \ldots, k$. In other words, $(\theta_{1,t}), \ldots, (\theta_{m,t})$ are a sample from a discrete latent distribution, P say, which gives probability mass p_j to the process $(\theta^*_{j,t})$, $j = 1, \ldots, k$; in formulas

$$
(\theta_{1,t}), \ldots, (\theta_{m,t}) \mid P = \left\{ \begin{array}{ccc} (\theta^*_{1,t}) & \cdots & (\theta^*_{k,t}) \\ p_1 & \cdots & p_k \end{array} \right. \overset{\text{i.i.d.}}{\sim} P.
$$

Note that the individual state processes $(\theta_{1,t}), \ldots, (\theta_{m,t})$ are independent only conditionally on the latent distribution P. As we shall see in section.. of chapter 4, in Bayesian inference the mixing distribution P is random and one

has to assign a prior probability law on it. We have already assumed that the support points $(\theta^*_{1,t}), \ldots, (\theta^*_{k,t})$ of P are i.i.d. Markov processes, described by (3.42), and the prior is usually completed by assuming that the unknown weights $(p_1, \ldots, p_k)$ have a Dirichlet distribution, independent if the $(\theta^*_{j,t})$'s. The equivalence between the model formulated in terms of the latent factor Z or in the mixing distribution P is given by letting $(\theta_{i,t}) = (\theta^*_{j,t})$ if and only if $Z_i = j$. Bayesian inference on the clustering structure of the data is then carried out by computing (by MCMC) the posterior probability law of the mixing distribution, as we shall see in section.... of Chapter 5.

3.3.6 Dynamic regression

Suppose we want to study the dependence of a variable Y on one or more explanatory variables X, and to this aim at times $t = 1, 2, \ldots$, we collect the values $Y_{i,t} = Y_t(x_i)$ of Y at different values $x_1, \ldots, x_m$ of X. Thus we have a time series of cross sectional data of the kind $((Y_{i,t}, x_i), i = 1, \ldots, m), t \geq 1$; note that the x_i are deterministic, while the $Y_{i,t}$ are random. For example, in financial applications, $Y_{i,t}$ might be the price at time t of a zero coupon bond which gives one euro at time-to-maturity x_i. At each time t, we observe the prices $Y_t = (Y_{1,t}, \ldots, Y_{m,t})'$ of bonds with times-to-maturity $x_1, \ldots, x_m$ respectively. Data of this kind are plotted in Figure 3.11.

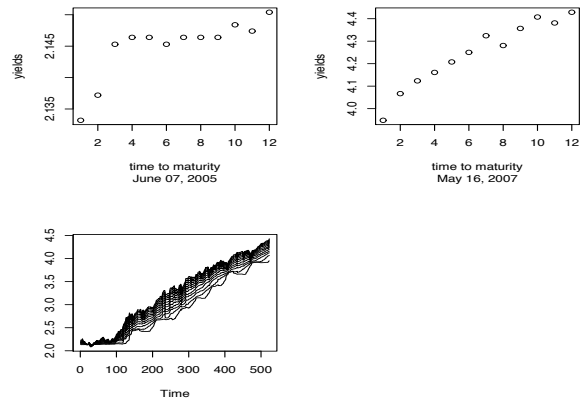

Fig. 3.11. Euribor..... Cross-sectional data (1 to 12 months to maturity), and the 12 time series from 05/16/2005 to 05/16/2007. Source: .....

Here we have two problems. On one hand, we want to estimate the regression function of Y_t on X, that is $m_t(x) = E(Y_t \mid x)$, using the cross-sectional data at time t. This is clearly of interest for understanding the relation between Y and x or for estimating Y for a new value of x (interpolation), and

so on. On the other hand, we have m time series $(Y_{i,t})$, and the interest might be in studying their temporal evolution. A simple approach could be using a univariate time series model for each of them, or a multivariate model, for predicting $Y_{i,t+1}$ given the data up to time t. This approach might have a fairly good predictive performance but treats $(Y_{1,t}, \ldots, Y_{m,t})$ as a random vector, forgetting that they are observations for different values of X and "should lie" on the regression curve $m_t(x)$. In fact, estimating the dynamics of the regression curve is often one main objective of the analysis.

For considering both aspects of the problem, that is the cross-sectional and the time series nature of the data, the proposal in this section is a (nonparametric) dynamic regression model, written in the form of DLM.

Suppose for brevity that X is univariate. Flexible (nonparametric) cross-sectional models for the regression function are often obtained by expressing it through some expansion of the kind

$$m_t(x) = E(Y \mid x) = \sum_{j=1}^{k} \beta_{j,t}\, h_j(x) \tag{3.44}$$

where $h_j(x)$ are given basis functions (e.g. powers of x: $m_t(x) = \sum_{j=1}^{\infty} \beta_{j,t} x^j$, trigonometric functions, wavelets, etc.) and $\beta_t = (\beta_{1,t}, \ldots, \beta_{k,t})'$ a vector of coefficients. Roughly speaking, the idea is that the model is flexible since for k large enough (in principle, $k \to \infty$) it can approximate any interesting shape of the function $m_t(x)$ (for example, any continuous function on a closed interval can be approximated by polynomials). Models of the kind (3.44) are nevertheless simple since they are still linear in the parameters $\beta_{j,t}$; at a given time t, we have an observation equation

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

where

$$Y_t = \begin{pmatrix} Y_{1,t} \vdots Y_{m,t} \end{pmatrix}, \quad F = \begin{pmatrix} h_1(x_1) & \ldots & h_k(x_1) \\ \vdots & & \vdots \\ h_1(x_m) & \ldots & h_k(x_m) \end{pmatrix}, \quad \beta_0 = \begin{pmatrix} \beta_{1,t} \vdots \beta_{k,t} \end{pmatrix},$$

and β_t can be estimates by least squares.

Suppose now that 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 we want to include in the analysis. Note that 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 (3.44), its temporal evolution can be described by the dynamics of the finite-dimensional vector of coefficients $(\beta_{1,t}, \ldots, \beta_{k,t})$.

Thus we obtain a dynamic regression model in the form of a DLM

$$\begin{aligned} Y_t &= F\beta_t + v_t,\ v_t \sim \mathcal{N}_m(0, \sigma^2 I_m) \\ \beta_t &= G\beta_{t-1} + w_t,\ w_t \sim \mathcal{N}_k(0, W_t). \end{aligned}$$

The state equation expresses the temporal evolution of the regression function. A simple specification assumes that $\beta_{j,t}$ are independent random walks or AR(1) processes; or that, jointly, β_t is a VAR processes. A word of caution is however worth; the state equation introduces additional information but also constraints on the dynamics of the curve, so it is delicate: a poor specification of the dynamics may result in an unsatisfactory fit of the data.

Example — Estimating the term structure of interest rates

A relevant problem in financial applications is estimating the term structure of interest rates. Let $P_t(x)$ be the price at time t of a zero-coupon bond which gives 1 euro at time to maturity x. The curve $P_t(x)$, $x \in (0, t)$, is called discount function. Other curves of interest are obtained as one-by-one transformations of the discount function; the yield curve is $\gamma_t(x) = -\log P_t(x)/x$, and the instantaneous (nominal) forward rate curve is $f_t(x) = d(-\log P_t(x)/dx) = (dP_t(x)/dx)/P_t(x)$. The yield curve, or one of its transformations, allows to price any coupon bond as the sum of the present values of future coupon and principal payments. Clearly, the whole curve cannot be observed, but we can estimate it from the bond prices observed for a finite number of times-to-maturity, $x_1, \ldots, x_m$ say. More precisely, at time t we have data $(y_{i,t}, x_i), i = 1, \ldots, m)$, where $y_{i,t}$ is the observed yield corresponding to time-to-maturity x_i. Due to market frictions, the observed yields do not lie exactly on the yield curve but we assume that

$$y_{i,t} = \gamma_t(x_i) + v_{i,t}, \quad v_{i,t} \overset{\text{indep}}{\sim} \mathcal{N}(0, \sigma^2), i = 1, \ldots, m.$$

Several cross sectional models for the yield curve have been proposed in the literature. One of the most popular is the Nelson and Siegel (1987) model. In fact, Nelson and Siegel model the forward rate curve as

$$f_t(x) = \beta_{1,t} + \beta_{2,t} e^{-\lambda x} + \beta_{3,t} \lambda\, x\, e^{-\lambda x},$$

(which is a Laguerre polynomial), from which the yield curve can be obtained as

$$\gamma_t(x) = \beta_{1,t} + \beta_{2,t} \frac{1 - e^{-\lambda x}}{\lambda\, x} + \beta_{3,t} \left(\frac{1 - e^{-\lambda x}}{\lambda\, x} - e^{-\lambda x} \right).$$

This model is not linear in the parameters $(\beta_{1,t}, \beta_{2,t}, \beta_{3,t}, \lambda)$; however the decay parameter λ is usually approximated with a fixed value (Diebold and Li; 2006, see for example), so that the model is more simply treated as a linear model in the unknown parameters $(\beta_{1,t}, \beta_{2,t}, \beta_{3,t})$. Thus, for a fixed value of λ, the model is of the form (3.44), with $k = 3$ and

$$h_1(x) = 1, \quad h_2(x) = \frac{1 - e^{-\lambda x}}{\lambda\, x}, \quad h_3(x) = \left(\frac{1 - e^{-\lambda x}}{\lambda\, x} - e^{-\lambda x} \right).$$

In the literature, Nelson and Siegel model is also regarded as a latent factor model (see Section 3.3.7), with $\beta_{1,t}, \beta_{2,t}, \beta_{3,t}$ playing the role of latent dynamic

factors (long-term, short-term and medium-term factors), also interpreted in terms of level, slope and curvature of the yield curve. In these terms, λ determines the maturity at which the loading on the medium-term factor, or curvature, achieves it maximum (so, for example, Diebold and Li (2006) consider 30-months maturity and fix $\lambda = 0.0609$). Cross-sectional estimates at a given time t can be obtained by ordinary least squares. However, as discussed above, we can also obtain dynamic estimates of the yield curve, adding a state equation for $\beta_t = \beta_{1,t}, \beta_{2,t}, \beta_{3,t}$. For example, Diebold et al. (2006) consider a VAR(1) dynamics for β_t where

$$\beta_t = A\beta_{t-1} + w_t \quad w_t \sim \mathcal{N}(0, W),$$

also studying the effects of macroeconomic variables. More refined dynamics take into account the constraints imposed on the yield curve evolution by the no-arbitrage restrictions; see e.g. Petrone and Corielli (2005).

3.3.7 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

$$Y_t = A\mu_t + v_t \qquad v_t \sim \mathcal{N}(0, V), \tag{3.45}$$
$$\mu_t = \mu_{t-1} + w_t \qquad w_t \sim \mathcal{N}(0, W), \tag{3.46}$$

where A is a fixed m by 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 by 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

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

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 the (i,j) element of A, $A_{i,j}$, is zero for $j > i$. Since A is m by p, with $p < m$, this means that A can be written as a partitioned matrix as

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

with T p by p lower triangular, and B an $m-p$ by 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 co-integrated series, introduced by Granger (1981) (see also Engle and Granger (1987)). The components of a vector time series x_t are said to be co-integrated of order d, b, written $x_t \sim \mathrm{CI}(d,b)$, 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 $Y_t \sim \mathrm{CI}(1,1)$. In a model where the common factors are stochastic linear trends instead of random walks, one can see that the observable series are $\mathrm{CI}(2,2)$.

Other DLM components that are commonly used as common factors include seasonal components and cycles, especially in economic applications. Further details on common factor models can be found in Harvey (1989).

3.3.8 Multivariate ARMA models

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

$$Y_t = \sum_{j=1}^{p} \Phi_j Y_{t-j} + \epsilon_t + \sum_{j=1}^{q} \Psi_j \epsilon_{t-j}, \tag{3.47}$$

where (ϵ_t) is an m-variate Gaussian white noise sequence with variance Σ and the Φ_j and Ψ_j are m by m matrices. Here, without loss of generality, we have taken the mean of the process to be zero. In order for (3.47) to define a stationary process, all the roots of the complex polynomial

$$\det(I - \Phi_1 z - \cdots - \Phi_p z^p) \tag{3.48}$$

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 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 by m. For example, let us consider the bivariate ARMA(2,1) process

$$Y_t = \Phi_1 Y_{t-1} + \Phi_2 Y_{t-2} + \epsilon_t + \Psi_1 \epsilon_{t-1}, \qquad \epsilon_t \sim \mathcal{N}(0, \Sigma), \tag{3.49}$$

with

$$\Psi_1 = \begin{bmatrix} \Psi_{11} & \Psi_{12} \\ \Psi_{21} & \Psi_{22} \end{bmatrix}, \qquad \Phi_i = \begin{bmatrix} \Phi_{11,i} & \Phi_{12,i} \\ \Phi_{21,i} & \Phi_{22,i} \end{bmatrix}, \quad i = 1, 2. \tag{3.50}$$

Then the system and observation matrices needed to define the DLM representation of (3.49) are the following:

$$\begin{aligned} F &= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}, \\ G &= \begin{bmatrix} \Phi_{11,1} & \Phi_{12,1} & 1 & 0 \\ \Phi_{21,1} & \Phi_{22,1} & 0 & 1 \\ \Phi_{11,2} & \Phi_{12,2} & 0 & 0 \\ \Phi_{21,2} & \Phi_{22,2} & 0 & 0 \end{bmatrix}, \\ R &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ \Psi_{11} & \Psi_{12} \\ \Psi_{21} & \Psi_{22} \end{bmatrix}, \qquad W = R \Sigma R'. \end{aligned} \tag{3.51}$$

In R, the function `dlmModARMA` can be used to create a DLM representation of an ARMA model also in the multivariate case.

For a detailed treatment of multivariate ARMA models the reader can consult Reinsel (1997) and Lütkepohl (2005). In fact, multivariate ARMA models can be difficult to interpret and most popular are vector autoregressive models (VAR), which we discuss briefly in the next subsection.

3.3.9 Vector autoregressive models

Vector autoregressive models (VAR) are widely used especially in econometrics; a comprehensive treatment is given by Lütkepohl (2005) and Bayesian VAR are discussed e.g. by Canova (2007). A VAR model of order p is a special case of (3.47), defined by

$$y_t = \nu + \Phi_1 y_{t-1} + \cdots + \Phi_p yt - p + \epsilon_t,$$

where (ϵ_t) is a (Gaussian) white noise, $\epsilon_t \sim \mathcal{N}_m(0, \Sigma)$ and $E(u_t u_s') = 0$ for $t \neq s$. The VAR(p) process is stationary if its reverse characteristic polynomial

$$\det(I_m - \Phi_1 z - \cdots - \Phi_p z^p)$$

has no roots outside the unit circle (more precisely, this is a condition for *stability* of the VAR(p) process, but it can be proved that it implies that (y_t) is stationary). In econometrics, VAR models are widely used for forecasting but also for structural analysis, that is for exploring the relationships between groups of economic variables. Suppose that y_t is partitioned in a group of k variables x_t and $(m-k)$ variables z_t. The VAR(p) model for $y_t = (x_t', z_t')'$ provides the joint probability law of all the variables involved and the covariance matrix Σ of y_t summarizes the correlation structure among them. In principle, from the joint distribution of the process $(y_t, t \geq 1)$ all the conditional distributions can be computed, in particular the predictive density $p(y_{t+1}|y_1, \ldots, y_t)$, or the conditional predictive density $p(z_{t+1}|x_{t+1}, y_1, \ldots, y_t)$, and point or interval forecasts can be obtained as synthesis of the predictive densities or, more formally, with respect to a loss function. We will provide an example of forecasting with a VAR model in Chapter 5, in a Bayesian framework.

In practice, economists are also interested in exploring *causality* relationships: that is, from estimation of the joint model for (y_t), one would like to study what are the effects of the variables x_t, say, on the variables z_t. There are several ways of interpreting this problem (Granger causality, impulse-response functions, forecast error variance decomposition) but a discussion is beyond our scope here; see e.g. Lütkepohl (2005) and references therein. We only note that these studies are affected by identifiability issues, so that causality relationships or impulse-response functions are not uniquely defined. In fact, VAR are reduced form models which capture the dynamic properties of the variables and are useful for forecasting, but for exploring the structural relationships they are often insufficient because different economic theories may be compatible with the same statistical reduced form models. Prior information and constraints have to be included for identifying the relevant innovations and impulse-response; the resulting models are referred as *structural VAR*; see e.g. Amisano and Giannini (1997). As we will illustrate in Chapter 5, the Bayesian approach reveals several advantages in these problems and Bayesian VAR models have become increasingly popular in econometrics and in many other applications.

Problems

3.1. Simulate patterns of the random walk plus noise model for varying values of V and W.

3.2. Show that, for the local level model, $\lim_{t\to\infty} C_t = KV$, where K is defined by (3.9).

3.3. Let $(Y_t, t = 1, 2, \ldots)$ 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 $MA(1)$ process.

3.4. Simulate patterns of the linear growth model for varying values of V and W_1, W_2.

3.5. Let $(Y_t, t = 1, 2, \ldots)$ 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 $MA(2)$ process.

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 (3.15).

Solution From the identity $e_t = Y_t - (\hat{\mu}_{t-1} + \hat{\beta}_{t-1})$ and the expressions of m_t in (3.11), we can write the second differences $z_t = Y_t - 2Y_{t-1} + Y_{t-2}$ as

$$\begin{aligned} z_t &= \hat{\mu}_{t-1} + \hat{\beta}_{t-1} + e_t - 2(\hat{\mu}_{t-2} + \hat{\beta}_{t-2}) - 2e_{t-1} + \hat{\mu}_{t-3} + \hat{\beta}_{t-3} + e_{t-2} \\ &= \hat{\mu}_t - k_{1,t}e_t + e_t - 2\hat{\mu}_{t-1} + 2k_{1,t-1}e_{t-1} - 2e_{t-1} + \hat{\mu}_{t-2} - k_{1,t-2}e_{t-2} + e_{t-2} \\ &= \hat{\beta}_{t-1} - \hat{\beta}_{t-2} + e_t + k_{1,t-1}e_{t-1} - 2e_{t-1} - k_{1,t-2}e_{t-2} + e_{t-2} \\ &= e_t + (-2 + k_{1,t-1} + k_{2,t-1})e_{t-1} + (1 - k_{1,t-2})e_{t-2} \end{aligned}$$

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.

3.8. Show that the DLM (3.22) has a periodic forecast function, even when W is a general variance matrix.

3.9. (*ARMA models*). — qualche esercizio anche di analisi standard - f di autocoorelazione ect., con R - dai lab di Giovanni e mio.... Confronto con stima ARIMA e con dlm? anche se qui non parliamo di stima?

3.10. Prove (3.32).

3.11. (*VAR models*). Simulate and draw paths of a stationary bivariate VAR(1) process. Compare with paths of a nonstationary VAR(p) process.

4

Models with unknown parameters

In the previous chapters we presented some basic DLM for time series analysis, assuming that the system matrices F_t, G_t, V_t and W_t were known, with the aim of understanding their behavior and general properties. In fact, in time series applications the matrices in the DLM are very rarely completely known. In this chapter we let the model matrices depend on a vector of unknown parameters ψ. We will consider examples where ψ is constant over time, or it has a temporal evolution/random fluctuations. The dynamics of ψ is anyway such to maintain the linear, Gaussian structure of the DLM. In chapter 5, we will see examples where more general dynamics are introduced on the unknown parameters, which are then treated as latent states in the more general framework of state space models.

In a classical framework one typically starts by estimating ψ, usually by maximum likelihood. If the researcher is only interested in the unknown parameters, the analysis terminates here; if, on the other hand, he is interested in smoothing or forecasting the values of the observed series or those of the state vectors, the customary way to proceed is to use the estimated value of ψ as if it were a known constant, and apply the relevant techniques of Chapter 2 for forecasting or smoothing.

From a Bayesian standpoint unknown parameters are instead random variables, as we discussed in Chapter 1: therefore, in the context of DLM, the posterior distribution of interest is the joint conditional distribution of the state vectors – or of future measurements – and the unknown parameter ψ, given the observations. As we shall see, Bayesian inference, even if simple in principle, involves computations that are usually not analytically manageable; however, Markov chain Monte Carlo and modern sequential Monte Carlo methods can be quite efficient in providing an approximation of the posterior distributions of interest.

In Section 4.1 we discuss Maximum Likelihood estimation of an unknown parameter occurring in the specification of a DLM, while the rest of the chapter is devoted to Bayesian inference.

4.1 Maximum likelihood estimation

Suppose that we have n random vectors, $Y_1, \ldots, Y_n$, whose distribution depends on an unknown parameter ψ. We will denote the joint density of the observations for a particular value of the parameter, by $p(y_1, \ldots, y_n; \psi)$. The likelihood function is defined to be, up to a constant factor, the probability density of the observed data read as a function of ψ, i.e., denoting the likelihood by L, we can write $L(\psi) = \text{const.} \cdot p(y_1, \ldots, y_n; \psi)$. For a DLM it is convenient to write the joint density of the observations in the form

$$p(y_1, \ldots, y_n; \psi) = \prod_{t=1}^{n} p(y_t | \mathcal{D}_{t-1}; \psi), \tag{4.1}$$

where $p(y_t|\mathcal{D}_{t-1}; \psi)$ is the conditional density of y_t given the data up to time $t-1$, assuming that ψ is the value of the unknown parameter. We know from Chapter 2 that the terms occurring in the RHS of (4.1) are Gaussian densities with mean f_t and variance Q_t. Therefore we can write the loglikelihood as

$$\ell(\psi) = -\frac{1}{2}\sum_{t=1}^{n} \log |Q_t| - \frac{1}{2}\sum_{t=1}^{n} (y_t - f_t)' Q_t^{-1} (y_t - f_t), \tag{4.2}$$

where the f_t and the Q_t depend implicitely on ψ. The expression (4.2) can be numerically maximized to obtain the maximum likelihood estimator (MLE) of ψ:

$$\hat{\psi} = \underset{\psi}{\operatorname{argmax}}\, \ell(\psi). \tag{4.3}$$

Denote by H the Hessian matrix of $-\ell(\psi)$, evaluated at $\psi = \hat{\psi}$. The matrix H^{-1} provides an estimate of the variance of the MLE, $\text{Var}(\hat{\psi})$. Conditions for consistency as well as asymptotic normality of the MLE can be found in Caines (1988) and Hannan and Deistler (1988). See also Shumway and Stoffer (2000) for an introduction. For most of the commonly used DLM, however, the usual consistency and asymptotic normality properties of MLE hold.

A word of caution about numerical optimization is in order. The likelihood function for a DLM may present many local maxima. This implies that starting the optimization routine from different starting points may lead to different maxima. It is therefore a good idea to start the optimizer several times from different starting values and compare the corresponding maxima. A rather flat likelihood is another problem that one may face when looking for a MLE. In this case the optimizer, starting from different initial values, may end up at very different points corresponding to almost the same value of the likelihood. The estimated variance of the MLE will typically be very large. This is a signal that the model is not well identifiable. The solution is usually to simplify the model, eliminating some of the parameters, especially when one is interested in making inference and interpreting the parameters themselves. On the other hand, if smoothing or forcasting is the focus, then

sometimes even a model which is not well identified in terms of its parameters may produce good results.

R provides an extremely powerful optimizer with the function *optim*, which is used inside the function *dlmMLE* in package *dlm*. In the optimization world it is customary to minimize functions, and *optim* is no ecception: by default it seeks a minimum. Statisticians too, when looking for an MLE, tend to think in terms of minimizing the negative loglikelihood. In line with this point of view, the function *dlmLL* returns the *negative* loglikelihood of a specified DLM, for a given data set. In terms of the parameter ψ occurring in the definition of the DLM of interest, one can think of minimizing the compound function obtained in two steps by building a DLM first, and then evaluating its negative loglikelihood, as a function of the matrices defining it. A suggestive graphical representation is the following:

$$\psi \quad \stackrel{\text{build}}{\Longrightarrow} \quad \text{DLM} \quad \stackrel{\text{loglik.}}{\Longrightarrow} \quad -\ell(\psi).$$

That is exacly what *dlmMLE* does: it takes a user-defined function *build* that creates a DLM, defines a new function by composing it with *dlmLL*, and passes the result to *optim* for the actual minimization. Consider for example the annual precipitation data for Lake Superior. By plotting the data (see ???), it seems that a polynomial model of order one can provide an adequate description of the phenomenon. The code below shows how to find the MLE of V and W.

R code

```
> y <- ts(as.matrix(read.table("Datasets/lakeSuperior.dat",
+                              skip=3))[,2],start=c(1900,1))
> build <- function(parm) {
+   dlmModPoly(order=1, dV=exp(parm[1]), dW=exp(parm[2]))
+ }
> fit <- dlmMLE(y, rep(0,2), build)
> fit$convergence
[1] 0
> unlist(build(fit$par)[c("V","W")])
        V         W
9.4654447 0.1211534
```

We have parametrized the two unknown variances in terms of their log, so as to avoid problems in case the optimizer went on to examine negative values of the parameters. The value returned by *dlmMLE* is the list returned by the call to *optim*. In particular, the component *convergenge* needs always to be checked: a nonzero value signals that convergence to a minimum has not been achieved. *dlmMLE* has a `...` argument that can be used to provide additional named arguments to *optim*. For example, a call to *optim* including the argument *hessian=TRUE* forces *optim* to return a numerically evaluated

Hessian at the minimum. This can be used to estimate standard errors of the components of the MLE, or more generally its estimated variance matrix, as detailed above. In the previous example we parametrized the model in terms of $\psi = (\log(V), \log(W))$, so that standard errors estimated from the Hessian refer to the MLE of these parameters. In order to get standard errors for the MLE of V and W, one can apply the delta method. Let us recall the general multivariate form of the delta method. Suppose that ψ is h-dimensional, and $g : \mathbb{R}^h \to \mathbb{R}^k$ is a function which has continuous first derivatives. Write $g(\psi) = (g_1(\psi), \ldots, g_k(\psi))$ for any $\psi = (\psi_1, \ldots, \psi_h) \in \mathbb{R}^h$, and define the derivative of g to be the k by h matrix

$$Dg = \begin{bmatrix} \frac{\partial g_1}{\partial \psi_1} & \cdots & \frac{\partial g_1}{\partial \psi_h} \\ \cdots & \cdots & \cdots \\ \frac{\partial g_k}{\partial \psi_1} & \cdots & \frac{\partial g_k}{\partial \psi_h} \end{bmatrix}, \tag{4.4}$$

that is, the ith row of Dg is the gradient of g_i. If $\widehat{\Sigma}$ is the estimated variance matrix of the MLE $\hat{\psi}$, then the MLE of $g(\psi)$ is $g(\hat{\psi})$, and its estimated variance is $Dg(\hat{\psi})\widehat{\Sigma}Dg(\hat{\psi})'$. In the example, $g(\psi) = (\exp(\psi_1), \exp(\psi_2))$, so that

$$Dg(\psi) = \begin{bmatrix} \exp(\psi_1) & 0 \\ 0 & \exp(\psi_2) \end{bmatrix}. \tag{4.5}$$

We can use the Hessian of the negative loglikelihood at the minimum and the delta method to compute in R standard errors of the estimated variances, as the code below shows.

R code

```
> fit <- dlmMLE(y, rep(0,2), build, hessian=TRUE)
> avarLog <- solve(fit$hessian)
> avar <- diag(exp(fit$par)) %*% avarLog %*%
+   diag(exp(fit$par)) # Delta method
> sqrt(diag(avar)) # estimated standard errors
[1] 1.5059107 0.1032439
```

As an alternative to using the delta method, one can numerically compute the Hessian of the loglikelihood, expressed as a function of the new parameters $g(\psi)$, at $g(\hat{\psi})$. The recommended package **nlme** provides the function **fdHess**, that we put to use in the following piece of code.

R code

```
> avar1 <- solve(fdHess(exp(fit$par), function(x)
+                       dlmLL(y,build(log(x))))$Hessian)
> sqrt(diag(avar1))
[1] 1.5059616 0.1032148 # estimated standard errors
```

In this example one could parametrize the model in terms of V and W, and then use the Hessian returned by *dlmMLE* to compute the estimated standard errors directly. In this case, however, one needs to be careful about the natural restriction of the parameter space, and provide a lower bound for the two variances. Note that the default optimization method, *L-BFGS-B*, is the only method that accepts restrictions on the parameter space, expressed as bounds on the components of the parameter. In the following code, the lower bound 10^{-6} for V reflects the fact that the functions in *dlm* require the matrix V to be nonsingular. On the scale of the data, however, 10^{-6} can be considered zero for all practical purposes.

R code

```
> build <- function(parm) {
+   dlmModPoly(order=1, dV=parm[1], dW=parm[2])
+ }
> fit <- dlmMLE(y, rep(0.23,2), build, lower=c(1e-6,0), hessian=T)
> fit$convergence
[1] 0
> unlist(build(fit$par)[c("V","W")])
        V         W
9.4654065 0.1211562
> avar <- solve(fit$hessian)
> sqrt(diag(avar))
[1] 1.5059015 0.1032355
```

Finally, let us mention the function *StructTS*, in base R. This function can be used to find MLE for the variances occurring in some particular univariate DLM. The argument *type* selects the model to use. The available models are the first order polynomial model (*type="level"*), the second order polynomial model (*type="trend"*), and a second order polynomial model plus a seasonal component (*type="BSM"*). Standard errors are not returned by *StructTS*, nor are easy to compute from its output.

R code

```
> StructTS(y,"level")

Call:
StructTS(x = y, type = "level")

Variances:
  level  epsilon
 0.1212   9.4654
```

4.2 Bayesian inference

The common practice of using the MLE's $\hat{\psi}$ as if they were the true values of the parameters in applying the filtering and smoothing recursions clearly suffers of the difficulties in taking properly into account the uncertainty about ψ. The Bayesian approach offers a more consistent formulation of the problem. The unknown parameters ψ are regarded as a random vector. The general hypotheses of state space models for the processes (Y_t) and (θ_t) (assumptions A.1 and A.2 on page 40) are assumed to hold *conditionally* on the parameters ψ. Prior knowledge about ψ is expressed through a probability law $\pi(\psi)$. Thus, for any $n \geq 1$, we assume that

$$(\theta_0, \theta_1, \ldots, \theta_n, Y_1, \ldots, Y_n, \psi) \sim \pi(\theta_0|\psi)p(\psi) \prod_{t=1}^{n} f(Y_t|\theta_t, \psi)\pi(\theta_t|\theta_{t-1}, \psi) \tag{4.6}$$

(compare with (2.3)). Given the data $\mathcal{D}_t = (y_1, \ldots, y_t)$, inference on the unknown states and parameters is solved by computing the posterior distribution

$$\pi(\theta_s, \psi|D_t) = \pi(\theta|\psi, \mathcal{D}_t)\pi(\psi|\mathcal{D}_t)$$

(*marginal* posterior; as usual, with $s = t$ for filtering, $s > t$ for state prediction, $s < t$ for smoothing problems), or the *joint* posterior distribution of the unknown state history up to time t and of the unknown parameter ψ. It is convenient to use the notation $\theta_{0:t}$ for denoting the vector $(\theta_0, \theta_1, \ldots, \theta_t)$, $t \geq 0$; similarly, from now on we will use the notation $y_{1:t}$ (in place of $\mathcal{D}_t$) for denoting the data $(y_1, \ldots, y_t)$. Thus, given the data $y_{1:t}$, the joint posterior distribution of interest is

$$\pi(\theta_{0:t}, \psi|y_{1:t}) = \pi(\theta_{0:t}|\psi, y_{1:t})\pi(\psi|y_{1:t}). \tag{4.7}$$

The results and the recursion formulae for estimation and forecasting given in chapter 2 hold conditionally on ψ and can be used for computing $\pi(\theta_s|\psi, y_{1:t})$; furthermore, they can be extended for obtaining the joint conditional density $\pi(\theta_{0:t}|\psi, y_{1:t})$ in (4.7). However, they are now weighted according to the posterior distribution of ψ given the data.

In principle, the posterior distribution (4.7) is computed using the Bayes rule. In some simple models and using conjugate priors, it can be computed in closed form; examples are given in the following section. More often, computations are analytically intractable. However, MCMC methods and modern sequential Monte Carlo algorithms provide quite efficient tools for approximating the posterior distributions of interest, and this is one reason of the enormous impulse enjoyed by Bayesian inference for state space models in the recent years.

Posterior distribution. MCMC and in particular Gibbs sampling algorithms can be used for approximating the joint posterior π. Gibbs sampling from $\pi(\theta_{0:t}, \psi|y_{1:t})$ requires to iteratively simulate from the full conditional

distributions $\pi(\theta_{0:t}|\psi, y_{1:t})$ and $\pi(\psi|\theta_{0:t}, |y_{1:t})$. Efficient algorithms for sampling from the full conditional $\pi(\theta_{0:t}|\psi, y_{1:t})$ have been developed, and will be presented in section 4.5. Furthermore, exploiting the conditional independence assumptions of DLM, the full conditional density $\pi(\psi|\theta_{0:T}, y_{1:T})$ is easier to compute than $\pi(\psi|y_{1:T})$. Clearly, this full conditional is problem-specific, but we will provide several examples in the next sections.

We can thus implement Gibbs sampling algorithms for approximating π. Samples from $\pi(\theta_{0:t}, \psi|y_{1:t})$ can also be used for approximating the filtering density $\pi(\theta_t, \psi|y_{0:t})$ and the marginal smoothing densities $\pi(\theta_s, \psi|y_{0:t}), s < t$; and, as we shall see, they also allow to simulate samples from the predictive distribution of the states and observables, $\pi(\theta_{t+1}, y_{t+1}|y_{0:t})$. Thus, this approach solves at the same time the filtering, smoothing and forecasting problems for a DLM with unknown parameters.

The shortcoming is that it is not designed for recursive or on-line inference. If a new observation y_{t+1} becomes available, the distribution of interest becomes $\pi(\theta_{0:t+1}, \psi|y_{1:t+1})$ and one has to run a new MCMC all over again for sampling from it. This can be quite inefficient, especially in applications that require an *on-line* type of analysis, in which new data arrive rather frequently. These problems are best dealt by using sequential Monte Carlo algorithms .

Filtering and on-line forecasting. As discussed in chapter 2, one of the attractive properties of DLM is the recursive nature of the filter formulas, which allows to update the inference efficiently as new data become available. In the case of no unknown parameters in the DLM, one could compute $\pi(\theta_{t+1}|y_{1:t+1})$ from $\pi(\theta_t|y_{1:t})$ by the estimation-error correction formulae given by Kalman filter, without doing all the computations again. Analogously, when there are unknown parameters ψ, one would like to exploit the samples generated from $\pi(\theta_{0:t}, \psi|y_{1:t})$ in simulating from $\pi(\theta_{0:t+1}, \psi|y_{1:t+1})$, without running the MCMC all over again. Modern sequential Monte Carlo techniques, in particular the family of algorithms that go under the name of *particle filters*, can be used to this aim and allow efficient on-line analysis and simulation-based sequential updating of the posterior distribution of states and unknown parameters. These techniques will be described in Section 4.8.

4.3 Conjugate Bayesian inference

In some simple cases, Bayesian inference can be carried out in closed form using conjugate priors. We illustrate an example here.

Clearly, even in simple structural models as presented in chapter 3, where the system matrices F_t and G_t are known, very rarely the covariance matrices V_t and W_t are completely known. Thus, a basic problem is estimating V_t and W_t. Here we consider a simple case where V_t and W_t are known only up to a common scale factor, that is $V_t = \sigma^2\tilde{V}_t$ and $W_t = \sigma^2\tilde{W}_t$, with σ^2 unknown. This specification of the covariance matrices has been discussed

in section .. of chapter 1 for the static linear regression model. A classical example is $V_t = \sigma^2 I_m$; an interesting way of specifying $\tilde{W}_t$ is discussed later, using *discount factors*.

4.3.1 Unknown covariance matrices: conjugate inference

Let $(Y_t, \theta_t), t \geq 1$ be described by a DLM with

$$V_t = \sigma^2 \tilde{V}_t, \qquad W_t = \sigma^2 \tilde{W}_t, \qquad C_0 = \sigma^2 \tilde{C}_0. \tag{4.8}$$

Here all the matrices $\tilde{V}_t$, $\tilde{W}_t$, as well as $\tilde{C}_0$ and all the F_t and G_t are assumed to be known. The scale parameter σ^2, on the other hand, is unknown. As usual in Bayesian inference it is convenient to work with its inverse $\phi = 1/\sigma^2$. The uncertainty therefore is all in the state vectors and in the parameter ϕ. The DLM provides the conditional probability law of (Y_t, θ_t) given ϕ; in particular the model assumes, for any $t \geq 1$,

$$Y_t|\theta_t, \phi \sim \mathcal{N}_m(F_t\theta_t, \phi^{-1}\tilde{V}_t)$$

$$\theta_t|\theta_{t-1}, \phi \sim \mathcal{N}_p(G_t\theta_{t-1}, \phi^{-1}\tilde{W}_t).$$

We have to choose a prior for (ϕ, θ_0), and a convenient choice is a conjugate Normal-Gamma prior (see the Appendix of Chapter 1), that is

$$\phi \sim \mathcal{G}(\alpha_0, \beta_0)$$

and

$$\theta_0|\phi \sim \mathcal{N}(m_0, \phi^{-1}\tilde{C}_0),$$

in symbols $(\theta_0, \phi) \sim \mathcal{NG}(m_0, \tilde{C}_0, \alpha_0, \beta_0)$. Then we have the following recursive formulae for filtering.

Proposition 4.1. *For the DLM described above, if*

$$\theta_{t-1}, \phi|y_{1:t-1} \sim \mathcal{NG}(m_{t-1}, \tilde{C}_{t-1}, \alpha_{t-1}, \beta_{t-1})$$

where $t \geq 1$, then

(i) The one-step-ahead predictive density of $(\theta_t, \phi)|y_{1:t-1}$ is Normal-Gamma with parameters $(a_t, \tilde{R}_t, \alpha_{t-1}, \beta_{t-1})$, where

$$a_t = G_t m_t, \qquad \tilde{R}_t = G_t \tilde{C}_{t-1} G_t' + \tilde{W}_t; \tag{4.9}$$

the one-step-ahead conditional predictive density of $Y_t|\phi, y_{1:t-1}$ is Gaussian, with parameters

$$f_t = F_t a_t, \qquad Q_t = \phi^{-1}\tilde{Q}_t = \phi^{-1} F_t \tilde{R}_t F_t' + \tilde{V}_t. \tag{4.10}$$

(ii) The filtering distribution of $(\theta_t, \phi|y_{1:t})$ *is Normal-Gamma, with parameters*

$$m_t = a_t + \tilde{R}_t F_t \tilde{Q}^{-1}(y_t - f_t) \quad \tilde{C}_t = \tilde{R}_t - \tilde{R}_t F_t' \tilde{Q}_t^{-1} \tilde{R}_t', \tag{4.11}$$
$$\alpha_t = \alpha_{t-1} + \frac{m}{2}, \qquad \beta_t = \beta_{t-1} + \frac{1}{2}(y_t - f_t)' \tilde{Q}_t^{-1}(y_t - f_t).$$

Note the analogy with the recursive formulas valid for a DLM with no unknown parameters. The results (4.9)-(4.10) and the expressions for m_t and C_t in (4.11) are those that we would obtain by Kalman filter (ch 2, theorem 2.2) for a DLM with variance matrices (4.8) and ϕ known.

By the properties of the Normal-Gamma distribution it follows from (ii) that the marginal filtering density of $\theta_t|y_{1:t}$ is a multivariate Student-t, and the density of σ^2 given $y_{1:T}$ is Inverse-Gamma (....)

Proof. (i) Suppose that $\theta_{t-1}, \psi|y_{1:t-1} \sim \mathcal{NG}(m_{t-1}, \tilde{C}_{t-1}, \alpha_{t-1}, \beta_{t-1})$ (this is true for $t = 0$). By (i) of theorem 2.2, we have that

$$\theta_t|\psi, y_{1:t-1} \sim \mathcal{N}_p(a_t, \phi^{-1}\tilde{R}_t)$$

with a_t and $\tilde{R}_t$ given by (4.9). Therefore

$$(\theta_t, \phi)|y_{1:t-1} \sim \mathcal{NG}(m_t, \tilde{C}_t, \alpha_t, \beta_t).$$

It also follows that $(y_t, \phi)|y_{1:t-1}$ has a Normal-Gamma distribution with parameters $(f_t, \tilde{Q}_t, \alpha_{t-1}, \beta_{t-1})$ and from this we have (4.10).

(ii) For a new observation y_t, the likelihood is

$$y_t|\theta_t, \psi \sim \mathcal{N}_m(F_t\theta_t, \phi^{-1}\tilde{V}_t)$$

The theory of linear regression with a Normal-Gamma prior discussed in ch.1 (page 19) applies, and leads to the conclusion that (θ_t, ϕ) given $\mathcal{D}_t$ has again a $\mathcal{NG}(m_t, \tilde{C}_t, \alpha_t, \beta_t)$ defined as in (4.11) (use (1.15) and (1.16)).

As far as smoothing is concerned, note that

$$(\theta_T, \phi|\mathcal{D}) \sim \mathcal{NG}(s_T, \tilde{S}_T, \alpha_T, \beta_T), \tag{4.12}$$

with $s_T = m_T$ and $\tilde{S}_T = \tilde{C}_T$, and write

$$\pi(\theta_t, \phi|\mathcal{D}_T) = \pi(\theta_t|\phi, \mathcal{D}_T)\pi(\phi|\mathcal{D}_T), \qquad t = 0, \ldots, T. \tag{4.13}$$

Conditional on ϕ, the Normal theory of chapter 1 applies, showing that (θ_t, ϕ), conditional on $\mathcal{D}_T$ has a Normal-Gamma distribution. The parameters can be computed using recursive formulas that are the analog of those developed for the Normal case. Namely, for $t = T-1, \ldots, 0$, let

$$s_t = m_t + \tilde{C}_t G_{t+1}' \tilde{R}_{t+1}^{-1}(s_{t+1} - a_{t+1}),$$
$$\tilde{S}_t = \tilde{C}_t - \tilde{C}_t G_{t+1}' \tilde{R}_{t+1}^{-1}(\tilde{R}_{t+1} - \tilde{S}_{t+1})\tilde{R}_{t+1}^{-1} G_{t+1}\tilde{C}_t.$$

Then

$$(\theta_T, \phi|\mathcal{D}) \sim \mathcal{NG}(s_t, \tilde{S}_t, \alpha_T, \beta_T). \tag{4.14}$$

4.3.2 Specification of W_t by discount factors

We briefly discuss a popular technique for specifying the covariance matrices W_t, based on the so-called discount-factors, which has the advantage of being fairly simple and effective; (see West and Harrison; 1997, section 6.3) for an in-depth discussion.

As we have often underlined, the structure and magnitude of the state covariance matrices W_t has a crucial role in determining the role of past observations in state estimation and forecasting. Roughly speaking, if W_t is large there is high uncertainty in the state evolution, and a lot of information is lost in passing from θ_{t-1} to θ_t: the information carried by the past observations $y_{1:t-1}$ about θ_{t-1} is of little relevance in forecasting θ_t and the current observation y_t is what mainly determines the estimate of $\theta_t|y_{1:t}$. In the Kalman filter recursions, the uncertainty about θ_{t-1} given the data $y_{1:t-1}$ is summarized in the conditional covariance matrix $C_{t-1} = V(\theta_{t-1}|y_{1:t-1})$; moving from θ_{t-1} to θ_t via the state equation $\theta_t = G_t\theta_{t-1} + w_t$, the uncertainty increases and we have $V(\theta_t|y_{1:t-1}) = R_t = G_t'C_{t-1}G_t + W_t$. Thus, if $W_t = 0$, i.e. there is no error in the state equation, we have $R_t = V(G_t\theta_{t-1}|y_{1:t-1}) = P_t$, say. Otherwise, P_t is increased in $R_t = P_t + W_t$. In this sense, W_t expresses the loss of information in passing from θ_{t-1} to θ_t due to the stochastic error component in the state evolution, the loss depending on the magnitude of W_t with respect to P_t. One can thus think of expressing W_t as a proportion of P_t:

$$W_t = \frac{1-\delta}{\delta} P_t \tag{4.15}$$

where $\delta \in (0, 1]$. It follows that $R_t = 1/\delta\, P_t$, with $1/\delta > 1$. The parameter δ is called *discount factor* since it "discounts" the matrix P_t that one would have with a deterministic state evolution into the matrix R_t. In practice, the value of the discount factor is usually chosen between 0.9 and 0.99, or it is chosen by model selection diagnostics, e.g. looking at the predictive performance of the model for varying values of δ.

The discount factor specification can be used in the model of the previous section. In (4.8), we can assume that

$$\tilde{W}_t = \frac{1-\delta}{\delta} G_t'\tilde{C}_{t-1}G_t.$$

Given $\tilde{C}_0$ and $\tilde{V}_t$ (e.g., $\tilde{V}_t = I_m$), the value of $\tilde{W}_t$ can be recursively computed for every t. Further refinements consider different discount factors δ_i for the different components of the state vector.

Example. As a simple illustration of the usage of the discount factor, let us consider again the time series of the example on page 77, giving the annual precipitation in inches at Lake Superior from 1900 to 1986 (Figure 3.2) We model this series by a random walk plus noise DLM with unknown variances

V_t and W_t. We suppose that V_t, W_t and C_0 satisfy the (formula 4.8). In particular we assume $\tilde{V}_t = 1$, that is $V_t = \sigma^2$ and we specify $\tilde{W}_t$ by the discount factor δ. Assuming $\phi = 1/\sigma^2$, we choose for (ϕ, θ_0) a Normal-Gamma prior $(\theta_0, \phi) \sim \mathcal{NG}(m_0, \tilde{C}_0, \alpha_0, \beta_0)$.

We create a linear growth DLM by `dlmModPoly` specifying $\tilde{V}_t$ by `dV=1`. The Kalman filter with the matrices W_t defined as in (formula 4.15) and the matrices F_t, G_t and V_t constant can be performed in R using the function `dlmFilter_DF`. The arguments are the data `y`, the model `mod`, and the discount factor `DF` which correspond to the δ value. The output produced by `dlmFilter_DF` is the same as this produced by `dlmFilter`.
Therefore the function `dlmFilter_DF` specifying $\tilde{W}_t$ by the discount factor `DF`, computes m_t, a_t, f_t and the singular value decomposition of $\tilde{C}_t$ and $\tilde{R}_t$ for any t.
Then we choose the parameters α_0 and β_0. The mean

$$E(1/\phi) = \beta_0/(\alpha_0 - 1)$$

is the initial a priori point estimate of the observational variance σ^2. By the recursive formulas for α_t and β_t (formula 4.11) we obtain

$$\alpha_t = \alpha_0 + \tfrac{t}{2}$$
$$\beta_t = \beta_0 + \tfrac{1}{2}\sum_{i=1}^{t}(y_i - f_i)^2 \tilde{Q}_i^{-1} = \beta_0 + \tfrac{1}{2}\sum_{i=1}^{t} \tilde{e}_t^2$$

where the standardized innovations $\tilde{e}_t$ and the standard deviation $\tilde{Q}_t^{1/2}$ can computed with a call to a `residuals` function. Finally, assuming $S_t = \beta_t/\alpha_t$ we compute

$$C_t = Var(\theta_t|\mathcal{D}_t) = \tilde{C}_t S_t \tag{4.16}$$
$$Q_t = Var(Y_t|\mathcal{D}_{t-1}) = \tilde{Q}_t S_{t-1} \tag{4.17}$$

We assume the initial values $m0 = 0$, $\tilde{C}_0 = 10^7$, $\alpha_0 = 2$, $\beta_0 = 20$, so that the initial point estimate of the observational variance σ^2 is $\hat{\sigma}_0^2 = 20$.
We examine four models being defined by discount values of $0.7, 0.8, 0.9$ and 1.0, the latter corresponding to the degenerate static model, $W_t = 0$. The data and the one-step ahead forecasts for the four models appear in Figure 4.1 The loss of information about the level between time $t-1$ and t and therefore the degree of adaptation to new data increases as δ decreases.
To compare the four models in terms of forecast accuracy the following table displays the MAPE, MAD, MSE measures.

DF	MAPE	MAD	MSE
1.0	0.0977	3.0168	21.5395
0.9	0.0946	2.8568	19.9237
0.8	0.0954	2.8706	20.2896
0.7	0.0977	2.9367	20.9730

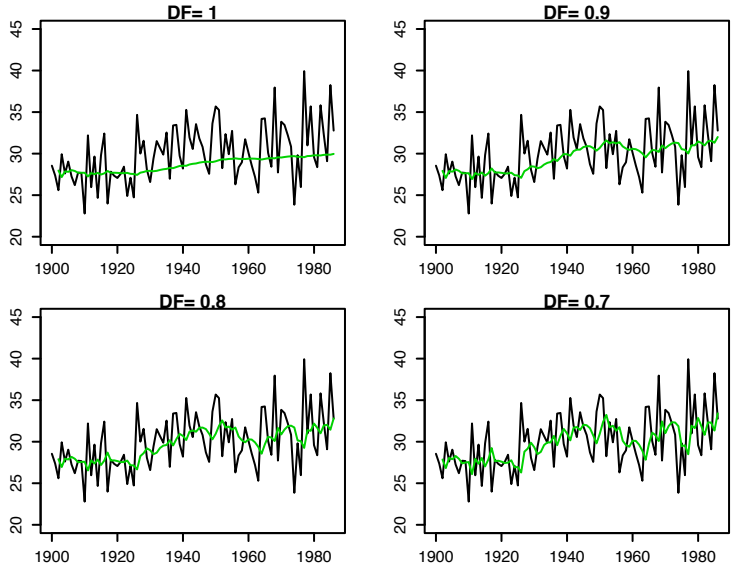

Fig. 4.1. Annual precipitation at Lake Superior and one-step ahead forecasts

Moreover the Figure 4.2 shows the sequence of estimates variances

$$\hat{\sigma}_t^2 = E(1/\phi|\mathcal{D}_t) = \beta_t/(\alpha_t - 1)$$

obtained by introducing a learning mechanism. In particular the final value of the estimated observational variance $\hat{\sigma}_{87}^2$ decreases as δ decreases.

DF	1.0	0.9	0.8	0.7
$\hat{\sigma}_{87}^2$	12.0010	9.6397	8.9396	8.3601

The MAPE, MAD e MSE measures lead to choose $\delta = 0.9$.
The unconditional (on ϕ) distribution of $\mu_t|\mathcal{D}_t$ is Student-t $\mathcal{T}(m_t, C_t, 2\alpha_t)$ where C_t is given by the (4.16). The Figure 4.3 shows the filtered level m_t with 90% equal tails probability intervals.
The unconditional (on ϕ) distribution of $Y_t|\mathcal{D}_{t-1}$ is Student-t $\mathcal{T}(f_t, Q_t, 2\alpha_{t-1})$ where Q_t is given by the (4.17). The Figure 4.3 shows the one-step ahead forecasts f_t with 90% equal tails probability intervals.

R code

```
> y <- read.table("plsuper.txt")
> y <- ts(y, frequency = 1, start = 1900)
> mod <- dlmModPoly(1,dV=1)
> modFilt <- dlmFilter_DF(y, mod, DF=0.9)
```

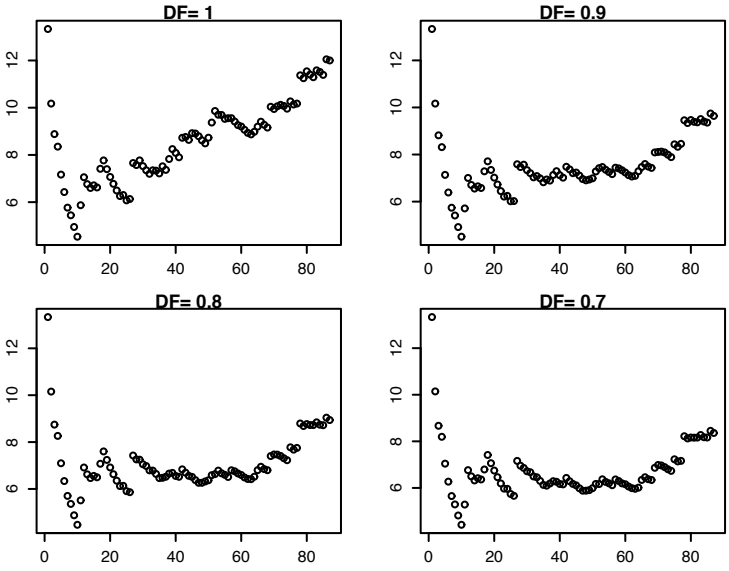

Fig. 4.2. Estimates of the variance σ^2

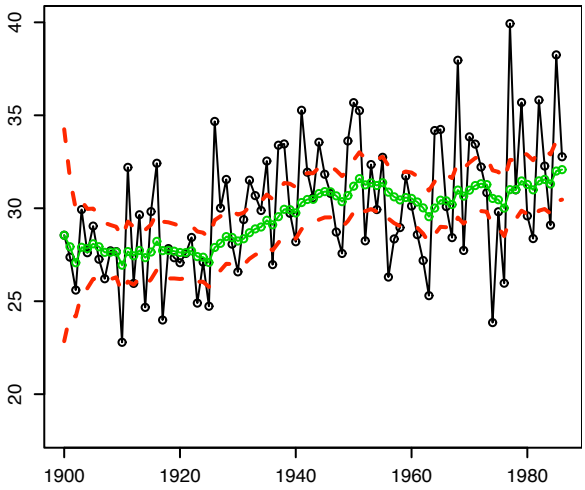

Fig. 4.3. Filtered level and 90% probability intervals

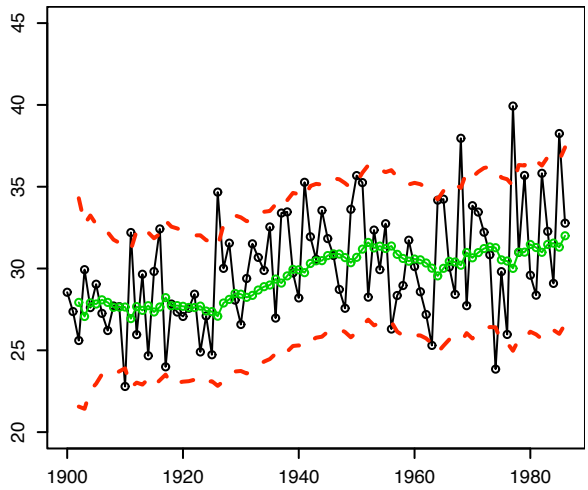

Fig. 4.4. One-step ahead forecasts and 90% probability intervals

```
> out <- residuals(modFilt)
> beta0 <- 20
> alpha0 <- 2
> beta <- beta0 + 1/2 * cumsum(out$res^2)
> alpha <- alpha0 + (1: length(y))*(1/2)
> S <- beta/alpha
> Ctildelist <- dlmSvd2var(modFilt$U.C, modFilt$D.C)
> C11tilde <- unlist(lapply(Ctildelist,function(x)x[1,1])[-1])
> se1 <- sqrt(C11tilde * S)
> conf <- 0.90
> k <- qt((conf+1)/2,df=2*alpha)
> plot(y,ylab="Observed/Level filtered estimates",type="o",ylim=c(18,40))
> lines(window(modFilt$m, start=start(y)),col=3,type="o")
> lines(window(modFilt$m,start=start(y)) - k*se1, lty=3, col="red")
> lines(window(modFilt$m,start=start(y)) + k*se1, lty=3, col="red")
> lagS <- c(beta0/alpha0,beta/alpha)
> lagS <- lagS[-length(lagS)]
> Qt <- out$sd^2 * lagS
> alphalag <- c(alpha0,alpha)
> alphalag <- alphalag[-length(alphalag)]
> kf <- qt((conf+1)/2,df=2*alphalag)
> plot(y,lty=2,ylab="Observed/One step ahead forecasts",type="p",ylim=c(20,45))
> lines(window(modFilt$f, start=1902),col=3, type="o")
> lines(window(modFilt$f,start=1902) - kf*sqrt(Qt), lty=3, col="red")
> lines(window(modFilt$f,start=1902) + kf*sqrt(Qt), lty=3, col="red")
```

4.3.3 A discount factor model for time-varying V_t

In the dlm (4.8), the unknown precision factor ϕ is assumed to be constant over time. Since, for simplicity, the components $\tilde{V}_t$ are often taken as time-invariant too, this implies a constant observation covariance matrix V_t, which is a restrictive assumption in many applications, for example for financial time series which show a stochastic volatility. More complex models will be discussed in later sections; here we apply the technique of discount factors for introducing a fairly simple temporal evolution for the precision ϕ in (4.8) (see West and Harrison; 1997, section 10.8).

Consider the dlm described in section 4.3.1. Suppose again that at time $t-1$

$$\phi_{t-1}|y_{1:t-1} \sim \mathcal{G}(\alpha_{t-1}, \beta_{t-1});$$

however, now ϕ is not constant, but it evolves from time $t-1$ to time t. Consequently, the uncertainty about ϕ_t, given the data $y_{1:t-1}$ will be bigger, that is $V(\phi_t|y_{1:t-1}) > V(\phi_{t-1}|y_{1:t-1})$. Let us *suppose* for the moment that $\phi_t|y_{1:t-1}$ has still a Gamma density, and in particular suppose that

$$\phi_t|y_{1:t-1} \sim \mathcal{G}(\delta\alpha_{t-1}, \delta\beta_{t-1}) \,, \tag{4.18}$$

where $0 < \delta < 1$. Notice that the expected value is not changed: $\mathrm{E}(\phi_t|y_{1:t-1}) = \mathrm{E}(\phi_{t-1}|y_{1:t-1}) = \alpha_{t-1}/\beta_{t-1}$, while the variance is bigger: $V(\phi_t|y_{1:t-1}) = 1/\delta\ V(\phi_{t-1}|y_{1:t-1})$, with $1/\delta > 1$. With this assumption, once a new observation y_t becomes available one can use the updating formulas of proposition 4.1, but starting from (4.18) in place of the $\mathcal{G}(\alpha_{t-1}, \beta_{t-1})$.

It remains to motivate the assumption (4.18), and in fact we have not specified yet the dynamics which leads from ϕ_{t-1} to ϕ_t. It can be proved (Ulhig (1994)) that assumption (4.18) is equivalent to assuming the following multiplicative model for the dynamics of ϕ_t

$$\phi_t = \frac{\gamma_t}{\delta}\phi_{t-1} \,,$$

where γ_t is a random variable independent on ϕ_{t-1}, with a beta density with parameters $(\delta\alpha_{t-1}, (1-\delta)\alpha t - 1)$ (so that $\mathrm{E}(\gamma_t) = \delta$). Therefore, ϕ_t is equivalent to ϕ_{t-1} multiplied by a random impulse with expected value 1 ($\mathrm{E}(\frac{\gamma_t}{\delta}) = 1$).

4.4 Simulation-based Bayesian inference

For a DLM including a possibly multidimensional unknown parameter ψ in its specification, and observations $y_{1:T}$, the posterior distribution of the parameter and unobservable states is

$$\pi(\psi, \theta_{0:T}|y_{1:T}). \tag{4.19}$$

As mentioned in section 4.2, in general it is impossible to compute in closed form this distribution. Therefore, in order to come up with posterior summaries one has to resort to numerical methods, almost invariably stochastic, Monte Carlo methods. The customary MCMC approach to analyze the posterior distribution (4.19) is to generate a (dependent) sample from it and evaluate posterior summaries from the simulated sample. The inclusion of the states in the posterior distribution usually simplifies the design of an efficient sampler, even when one is only interested in the the posterior distribution of the unknown parameter, $\pi(\psi|y_{1:T})$. In fact, drawing a random variable/vector from $\pi(\psi|\theta_{0:T}, y_{1:T})$ is almost invariably much easier than drawing it from $\pi(\psi|y_{1:T})$; in addition, efficient algorithms to generate the states conditionally on the data and the unknown parameter are available, see Section 4.5. This suggests that a sample from (4.19) can be obtained from a Gibbs sampler alternating draws from $\pi(\psi|\theta_{0:T}, y_{1:T})$ and $\pi(\theta_{0:T}|\psi, y_{1:T})$. The simulated sample from the posterior can in turn be used as imput to generate a sample from the predictive distribution of states and observables, $\pi(\theta_{T+1:T+k}, y_{T+1:T+k}|y_{1;T})$. In fact,

$$\pi(\theta_{T+1:T+k}, y_{T+1:T+k}, \psi, \theta_T|y_{1;T}) = \pi(\theta_{T+1:T+k}, y_{T+1:T+k}|\psi, \theta_T)\cdot\pi(\psi, \theta_T|y_{1;T}).$$

Therefore, for every pair $(\psi, \theta - T)$ drawn from $\pi(\psi, \theta_T|y_{1;T})$, one can generate the "future" $\theta_{T+1:T+k}, y_{T+1:T+k}$ from $\pi(\theta_{T+1:T+k}, y_{T+1:T+k}|\psi, \theta_T)$ (see Section 2.5) to obtain a sample from the predictive distribution.

The approach sketched above completely solves the filtering, smoothing and forecasting problems for a DLM with unknown parameters. However, if one needs to update the posterior distribution after one or more new observations become available, then one has to run the Gibbs sampler all over again, and this can be extremely inefficient. As already mentioned, on-line analysis and simulation-based sequential updating of the posterior distribution of states and unknown parameters are best dealt with employing sequential Monte Carlo techniques (Section 4.8).

4.5 Drawing the states given $\mathcal{D}_T$: Forward Filtering Backward Sampling

In a Gibbs sampling from $\pi(\theta_{0:T}, \psi|y_{1:T})$, one needs to simulate from the full conditional densities $\pi(\psi|\theta_{0:T}, y_{1:T})$ and $\pi(\theta_{0:T}|\psi, y_{1:T})$. While the first density is problem specific, the general expression of the latter density and efficient algorithms for sampling from it are available.

In fact, the smoothing recursions provide an algorithm for computing the mean and variance of the distribution of θ_t conditional on $y_{0:T}$ and ψ $(t = 0, 1, \ldots, T)$. Since all the involved distributions are Normal, this completely determines the *marginal* posterior distribution of θ_t given $y_{0:T}$ and ψ. If one is interested in the joint posterior distribution of $(\theta_{0:T})$ given $y_{1:T}, \psi$,

then also the posterior covariances between θ_t and θ_s have to be computed. General formulas to recursively evaluate these covariances are available, see Durbin and Koopman (2001). However, when $\pi(\theta_{0:T}|\psi, y_{1:T})$ has the role of full conditional in a Gibbs sampling from $\pi(\theta_{0:T}, \psi|y_{1:T})$, the main question becomes: how can one generate a draw from the distribution of $(\theta_{0:T})$ given $y_{1:T}, \psi$? We will use a method due to Carter and Kohn (1994), Früwirth-Schnatter (1994), and Shephard (1994), which is now widely known as Forward Filtering Backward Sampling (FFBS) algorithm. By reading the description that follows, the reader will realize that FFBS is essentially a simulation version of the smoothing recursions.

We can write the joint distribution of $(\theta_0, \theta_1, \ldots, \theta_T)$ given $\mathcal{D}_T$ as

$$\pi(\theta_0, \theta_1, \ldots, \theta_T|\mathcal{D}_T) = \prod_{t=0}^{T} \pi(\theta_t|\theta_{t+1}, \ldots, \theta_T, \mathcal{D}_T), \tag{4.20}$$

where the last factor in the product is simply $\pi(\theta_T|\mathcal{D}_T)$, i.e. the filtering distribution of θ_T, which is $\mathcal{N}(m_T, C_T)$. Formula (4.20) suggests that in order to obtain a draw from the distribution on the left-hand side, one can start by drawing θ_T from a $\mathcal{N}(m_T, C_T)$ and then, for $t = T-1, T-2, \ldots, 0$, recursively draw θ_t from $\pi(\theta_t|\theta_{t+1}, \ldots, \theta_T, \mathcal{D}_T)$. We have seen in the proof of Proposition 2.4 that $\pi(\theta_t|\theta_{t+1}, \ldots, \theta_T, \mathcal{D}_T) = \pi(\theta_t|\theta_{t+1}, \mathcal{D}_t)$, and we showed that this distribution is $\mathcal{N}(h_t, H_t)$, with

$$\begin{aligned} h_t &= m_t + C_t G'_{t+1} R^{-1}_{t+1}(\theta_{t+1} - a_{t+1}), \\ H_t &= C_t - C_t G'_{t+1} R^{-1}_{t+1} G_{t+1} C_t. \end{aligned}$$

Therefore, having already $(\theta_{t+1}, \ldots, \theta_T)$, the next step consists in drawing θ_t from a $\mathcal{N}(h_t, H_t)$. Note that h_t explicitly depends on the value of θ_{t+1} already generated.

In summary, the FFBS algorithm can be described as follows;

1. Run Kalman filter;
2. Draw θ_T from a $\mathcal{N}(m_T, C_T)$ distribution;
3. For $t = T-1, T-2, \ldots, 0$, draw θ_t from a $\mathcal{N}(h_t, H_t)$ distribution.

FFBS is commonly used as a building block of a Gibbs sampler, as we will illustrate in many examples in the remaining of the chapter. However, it can be of interest also in DLM when there are no unknown parameters. In this case, the marginal smoothing distribution of each θ_t is usually enough to evaluate posterior probabilities of interest. However, the posterior distribution of a nonlinear function of the states may be difficult or impossible to derive, even when all the parameters of the model are known. In this case FFBS provides an easy way to generate an independent sample from the posterior of the nonlinear function of interest. Note that in this type of application the "forward filtering" part of the algorithm only needs to be performed once.

4.6 General strategies for MCMC

For a completely specified DLM, i.e., one not containing unknown parameters, draws from the posterior distribution of the states, and possibly from the forecast distribution of states and observations, can be obtained using the algorithms described in Section 4.5. In the more realistic situation of a DLM containing an unknown parameter vector ψ, with prior distribution $\pi(\psi)$, in the observation, system, or variance matrices, one typically uses MCMC to obtain posterior summaries of the distributions of interest. Almost all Markov chain samplers for posterior analysis of a DLM fall in one of the following categories: Gibbs samplers which include the states as latent variables, marginal samplers, and hybrid samplers, which combine aspects of both. Note that, depending on the context, the analyst may be interested in making inference about the unobservable states, the unknown parameter, or both. Of the three types of samplers, two (Gibbs and hybrid samplers) generate draws from the joint posterior of the states and the parameter, while the other (marginal samplers) only generates draws from the posterior of the parameter. Keep in mind, however, that once a sample from the posterior distribution of the parameter is available, a sample from the joint posterior of states and parameter can be easily obtained in view of the decomposition

$$\pi(\theta_{0:T}, \psi | y_{1:T}) = \pi(\theta_{0:T} | \psi, y_{1:T}) \cdot \pi(\psi | y_{1:T}).$$

More specifically, for each $\psi^{(i)}$ in the sample ($i = 1, \ldots, N$), it is enough to draw $\theta_{0:T}^{(i)}$ from $\pi(\theta_{0:T} | \psi = \psi^{(i)}, y_{1:T})$ using FFBS, and $\{(\theta_{0:T}^{(i)}, \psi^{(i)}) : i = 1, \ldots, N\}$ will be the required sample from the joint posterior distribution.

The Gibbs sampling approach, consisting in drawing in turn the states from their conditional distribution given the parameter and observations, and the parameter from its conditional distribution given the states and observations, is summarized in Table 4.1. Package `dlm` provides the function

- Initialize: set $\psi = \psi^{(0)}$.
- For $i = 1, \ldots, N$:
  1. Draw $\theta_{0:T}^{(i)}$ from $\pi(\theta_{0:T} | y_{1:T}, \psi = \psi^{(i-1)})$ using FFBS.
  2. Draw $\psi^{(i)}$ from $\pi(\psi | y_{1:T} \theta_{0:T} = \theta_{0:T}^{(i)})$.

Table 4.1. Forward Filtering Backward Sampling in a Gibbs sampler

`dlmBSample` which, in conjunction with `dlmFilter`, can be used to perform step 1. Step 2, on the other hand, depends heavily on the model under consideration, including the prior distribution on ψ. In fact, when ψ is an r-dimensional vector, it is often simpler to perform a Gibbs step for each component of ψ instead of drawing ψ at once. The intermediate approach of drawing blocks of components of ψ together is another option. In any case,

when a full conditional distribution is difficult to sample from, a Metropolis-Hastings step can replace the corresponding Gibbs step. A generic sampler for nonstandard distributions is ARMS (Section 1.3), available in package `dlm` in the function `arms`.

The second approach, marginal sampling, is conceptually straightforward, consisting in drawing a sample from $\pi(\psi|y_{0:T})$. The actual implementation of the sampler depends on the model under consideration. Typically, if ψ is multivariate, one can use a Gibbs sampler, drawing each component, or block of components, from its full conditional distributions, possibly using a Metropolis-Hastings step when the relevant full conditional is not a standard distribution. Again, ARMS can be used in the latter case.

A hybrid sampler can be used when the parameter can be decomposed in two components, that is, when ψ can be written as (ψ_1, ψ_2), where each component may be univariate or multivariate. Table 4.2 gives an algorithmic description of a generic hybrid sampler. As for the previous schemes, Metropolis-Hastings steps, and ARMS in particular, can be substituted for direct sampling in steps 1 and 3. Step 2 can always be performed using `dlmFilter` followed by `dlmBSample`. For the theoretically inclined reader, let us point out a subtle difference between this sampler and a Gibbs sampler. In a Gibbs sampler, each step consists in applying a Markov transition kernel whose invariant distribution is the target distribution, so that the latter is also invariant for the composition of all the kernels. In a hybrid sampler, on the other hand, the target distribution is not invariant for the Markov kernel corresponding to step 1, so the previous argument does not apply directly. However, it is not difficult to show that the composition of step 1 and 2 does preserve the target distribution and so, when combined with step 3, which is a standard Gibbs step, it produces a Markov kernel having the correct invariant distribution.

- Initialize: set $\psi_2 = \psi_2^{(0)}$.
- For $i = 1, \ldots, N$:
  1. Draw $\psi_1^{(i)}$ from $\pi(\psi_1|y_{0:T}, \psi_2 = \psi_2^{(i-1)})$.
  2. Draw $\theta_{0:T}^{(i)}$ from $\pi(\theta_{0:T}|y_{1:T}, \psi_1 = \psi_1^{(i)}, \psi_2 = \psi_2^{(i-1)})$ using FFBS.
  3. Draw $\psi_2^{(i)}$ from $\pi(\psi_2|y_{1:T}\theta_{0:T} = \theta_{0:T}^{(i)}, \psi_1 = \psi_1^{(i)})$.

Table 4.2. Forward Filtering Backward Sampling in a hybrid sampler

The output produced by a Markov chain sampler must always be checked to assess convergence to the stationary distribution and mixing of the chain. Given that the chain has practically reached the stationary distribution, mixing can be assessed by looking at autocorrelation functions of parameters or functions of interest. Ideally, one would like to have as low a correlation as possible between draws. Correlation may be reduced by *thinning* the simulated

chain, i.e., discarding a fixed number of iterations between every saved value. Although this method is very easy to implement, the improvements are usually only marginal, unless the number of discarded simulations is substantial, which significantly increases the time required to run the entire sampler. As far as assessing convergence, a fairly extensive body of literature exists on diagnostic tools for MCMC. In R the package `BOA` provides a suite of functions implementing many of these diagnostics. In most cases, a visual inspection of the output, after discarding a first part of the simulated chain as *burn in*, can reveal obvious departures from stationarity. For an organic treatment of MCMC diagnostics, we refer to Robert and Casella (2004) and references therein.

4.6.1 Example: US Gross National Product

We now illustrate with an example how to implement in R a hybrid sampler using `dlmBSample` and `arms`. The data consist of the quarterly time series of deseasonalized real GNP of the US from 1950 to 2004, on a logarithmic scale. Following standard econometric practice, we assume that GNP can be decomposed into two unobservable components: a stochastic trend and a stationary component. We will estimate the two components as well as the parameters of the underlying model. We assume that the stochastic trend is described by a local linear trend, while the stationary component follows an AR(2) process. The order of the AR process allows the residuals (departures from the trend) to have a (dumped) cyclic autocorrelation function, which is often observed in economic time series. The model, as a DLM, is therefore the sum, in the sense discussed in Section 3.2, of a polynomial model of order two and a DLM representation of a stationary AR(2) process, observed with no error. The matrices of the resulting DLM are:

$$\begin{aligned} F &= \begin{bmatrix} 1 & 0 & 1 & 0 \end{bmatrix}, \\ G &= \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \phi_1 & 1 \\ 0 & 0 & \phi_2 & 0 \end{bmatrix}, \\ V &= \begin{bmatrix} 0 \end{bmatrix}, \\ W &= \operatorname{diag}\left(\sigma_\mu^2, \sigma_\delta^2, \sigma_u^2, 0\right). \end{aligned}$$

At any time t, The first component of the state vector represents the trend, while the third is the AR(2) stationary component. The AR parameters ϕ_1 and ϕ_2 must lie in the stationarity region $\mathcal{S}$ defined by

$$\begin{aligned} \phi_1 + \phi_2 &< 1, \\ \phi_1 - \phi_2 &> -1, \\ |\phi_2| &< 1. \end{aligned}$$

The prior we select for (ϕ_1, ϕ_2) is a product of a $\mathcal{N}(0, (2/3)^2)$ and a $\mathcal{N}(0, (1/3)^2)$, restricted to $\mathcal{S}$. In this way, the prior penalizes those values of the AR parameters close to the boundary of the stationarity region. For the three precisions, i.e., the inverses of the variances σ_μ^2, σ_δ^2, and σ_u^2, we assume independent gamma priors with mean a and variance b:

$$\mathcal{G}\left(\frac{a^2}{b}, \frac{a}{b}\right).$$

In this specific case, we set $a = 1$, $b = 1000$. A hybrid sampler can draw in turn the AR parameters from $\pi(\phi_1, \phi_2 | \sigma_\mu^2, \sigma_\delta^2, \sigma_u^2, y_{0:T})$, the states, and the three precisions from their full conditional distribution given the states and the AR parameters. In the notation used in Table 4.2, $\psi_1 = (\phi_1, \phi_2)$ and $\psi_2 = ((\sigma_\mu^2)^{-1}, (\sigma_\delta^2)^{-1}, (\sigma_u^2)^{-1})$. The precisions, given the states and the AR parameters, are conditionally independent and gamma-distributed. Specifically,

$$\begin{aligned}
(\sigma_\mu^2)^{-1} | \ldots &\sim \mathcal{G}\left(\frac{a^2}{b} + \frac{T}{2}, \frac{a}{b} + \frac{1}{2}\sum_{t=1}^{T}\big(\theta_{t,1} - (G\theta_{t-1})_1\big)^2\right), \\
(\sigma_\delta^2)^{-1} | \ldots &\sim \mathcal{G}\left(\frac{a^2}{b} + \frac{T}{2}, \frac{a}{b} + \frac{1}{2}\sum_{t=1}^{T}\big(\theta_{t,2} - (G\theta_{t-1})_2\big)^2\right), \\
(\sigma_u^2)^{-1} | \ldots &\sim \mathcal{G}\left(\frac{a^2}{b} + \frac{T}{2}, \frac{a}{b} + \frac{1}{2}\sum_{t=1}^{T}\big(\theta_{t,3} - (G\theta_{t-1})_3\big)^2\right).
\end{aligned} \tag{4.21}$$

The AR parameters, given the precisions (but not the states), have a non-standard distribution and we can use ARMS to draw from their joint full conditional distribution. One can write a function to implement the sampler in R. One such function, on which the analysis that follows is based, is available from the book web site. We reproduce below the relevant part of the main loop. In the code, `theta` is a T by 4 matrix of states and `gibbsPhi` and `gibbsVars` are matrices in which the results of the simulation are stored. The states, generated in the loop, can optionally be saved, but they can also be easily generated again, given the simulated values of the AR and variance parameters.

R code

```
for (it in 1:mcmc)
{
    ## generate AR parameters
    mod$GG[3:4,3] <- arms(mod$GG[3:4,3],
                          ARfullCond, AR2support, 1)
    ## generate states - FFBS
    modFilt <- dlmFilter(y, mod, simplify=TRUE)
    theta[] <- dlmBSample(modFilt)
```

```
    ## generate W
10  theta.center <- theta[-1,-4,drop=FALSE] -
        (theta[-(nobs + 1),,drop=FALSE] %*% t(mod$GG))[,-4]
12  SStheta <- drop(sapply( 1 : 3, function(i)
                    crossprod(theta.center[,i])))
14  diag(mod$W)[1:3] <-
        1 / rgamma(3, shape = shape.theta,
16      rate = rate.theta + 0.5 * SStheta)
    ## save current iteration, if appropriate
18  if ( !(it %% every) )
    {
20      it.save <- it.save + 1
        gibbsTheta[,,it.save] <- theta
22      gibbsPhi[it.save,] <- mod$GG[3:4,3]
        gibbsVars[it.save,] <- diag(mod$W)[1:3]
24  }
}
```

The *if* statement on line 18 takes care of the thinning, saving the draw only when the iteration counter `it` is divisible by `every`. The object `SStheta` (line 12) is a vector of length 3 containing the sum of squares appearing in the full conditional distributions of the precisions (equations 4.21). The two functions `ARfullCond` and `AR2support`, which are the main arguments of `arms` (line 5) are defined, inside the main function, as follows.

R code

```
AR2support <- function(u)
2 {
    ## stationarity region for AR(2) parameters
4   (sum(u) < 1) && (diff(u) < 1) && (abs(u[2]) < 1)
}
6 ARfullCond <- function(u)
  {
8   ## log full conditional density for AR(2) parameters
    mod$GG[3:4,3] <- u
10  -dlmLL(y, mod) + sum(dnorm(u, sd = c(2,1) * 0.33,
                               log=TRUE))
12 }
```

The sampler was run using the following call, where `gdp` is a time series object containing the data.

R code

```
outGibbs <- gdpGibbs(gdp, a.theta=1, b.theta=1000, n.sample =
2                    2050, thin = 1, save.states = TRUE)
```

Discarding the first 50 draws as burn in, we look at some simple diagnostic plots. The traces of the simulated variances (Figure 4.5) do not show any particular sign of a nonstationary behavior. We have also plotted the running

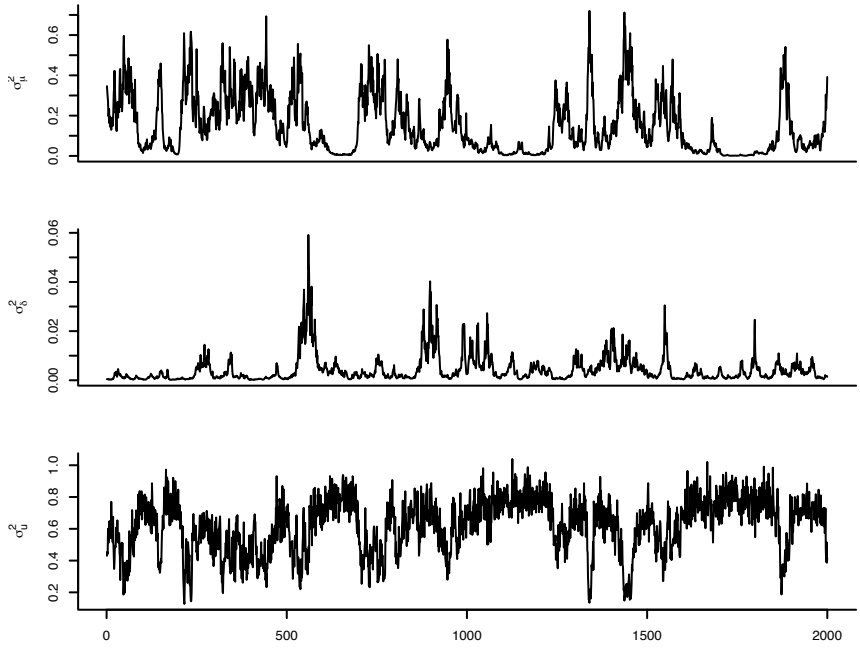

Fig. 4.5. GDP: traces of simulated variances

ergodic means of the simulated standard deviations σ_μ σ_δ, and σ_u (Figure 4.6). The first plot shows $n^{-1}\sum_{i=1}^{n}\sigma_\mu^{(i)}$ versus i, and similarly for the second and third. In other words, this is the MC estimate of σ_μ versus the number of iterations of the sampler. The estimates look reasonaby stable in the last part of the plot. (This impression was also confirmed by the results from a longer run, not shown here). The empirical autocorrelation functions of the three variances (Figure 4.7) give an idea of the degree of autocorrelation in the sampler. In the present case, the decay of the ACF is not very fast; this will reflect in a relatively large Monte Carlo standard error of the Monte Carlo esimates. Clearly, a smaller standard error can always be achieved by running the sampler longer. Similar diagnostic plots can be done for the AR parameters. The reader can find in the display below, for the three standard

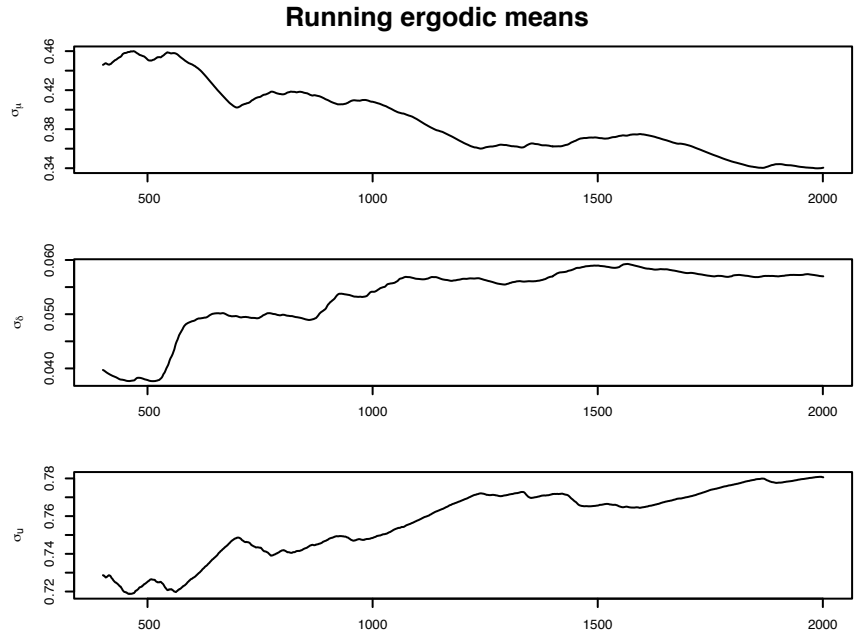

Fig. 4.6. GDP: ergodic means

deviations in the model and the two AR parameters, the estimates of the posterior means and their estimated standard errors, obtained using Sokal's method (see Section 1.3). In addition, equal-tail 90% probability intervals are derived for the five parameters. These probability intervals give an idea of the region where most of the posterior probability is contained.

R code

```
> mcmcMeans(outGibbs$phi[-burn,], names = paste("phi", 1:2))
    phi 1     phi 2
   1.3422   -0.4027
 ( 0.0112) ( 0.0120)
> apply(outGibbs$phi[-burn,], 2, quantile, probs = c(.05,.95))
        [,1]       [,2]
5%  1.174934 -0.5794382
95% 1.518323 -0.2495367
> mcmcMeans(sqrt(outGibbs$vars[-burn,]),
            names = paste("Sigma", 1:3))
   Sigma 1    Sigma 2    Sigma 3
  0.34052    0.05698    0.78059
 (0.03653)  (0.00491)  (0.01766)
> apply(sqrt(outGibbs$vars[-burn,]), 2, quantile,
```

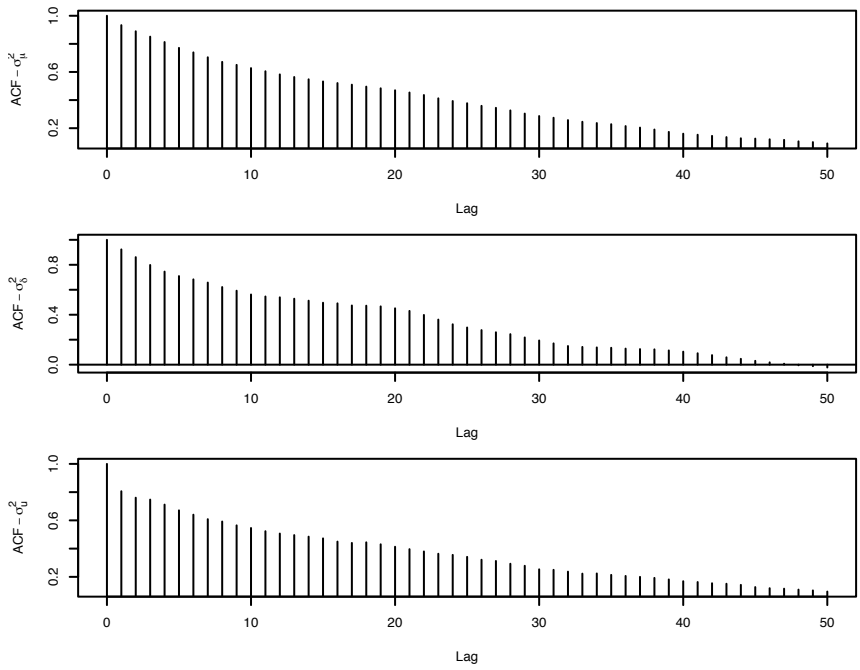

Fig. 4.7. GDP: Autocorrelation functions

```
        probs = c(.05,.95))
16            [,1]       [,2]      [,3]
   5%  0.06792123 0.02057263 0.5596150
18 95% 0.65583319 0.12661949 0.9294306
```

One can also plot histograms based on the output of the sampler, to gain some insight about the shape of posterior distributions of parameters or functions thereof – at least for univariate marginal posteriors. Figure 4.8 displays the histograms of the posterior distributions of the three variances. Scatterplots are sometimes useful to explore the shape of bivariate distributions, especially for pairs of parameters that are highly dependent on each other. Figure 4.9 displays a bivariate scatterplot of (ϕ_1, ϕ_2), together with their marginal histograms. From the picture, it is clear that there is a strong dependence between ϕ_1 and ϕ_2, which, incidentally, confirms that drawing the two at the same time was the right thing to do in order to improve the mixing of the chain.

Finally, since the sampler also included the unobservable states as latent variables, one can obtain posterior distributions and summaries of the states. In particular, in this example it is of interest to separate the trend of the GDP from the (autocorrelated) noise. The posterior mean of the trend at time t

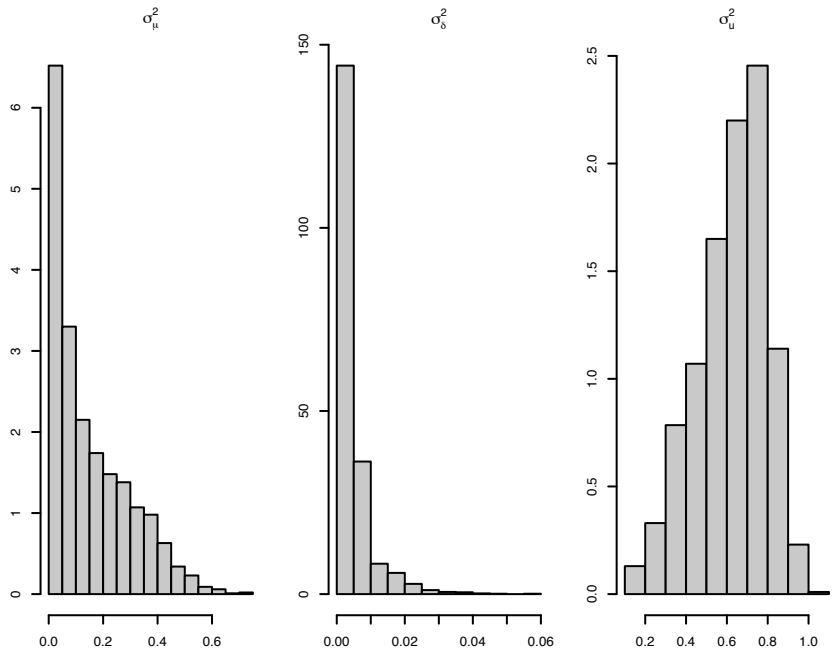

Fig. 4.8. GDP: posterior distributions of the model variances

is estimated by the mean of the simulated values $\theta_{t,1}^{(i)}$. Figure 4.10 displays graphically the posterior mean of the trend, together with the data, and the posterior mean of the AR(2) noise process, represented by $\theta_{t,3}$.

4.7 Unknown variances

In many of the models analyzed in Chapter 3 the system and observation matrices G_t and F_t are set to specific values as part of the model specification. This is the case for polynomial and seasonal factor models, for example. The only possibly unknown parameters are therefore part of the variance matrices W_t and V_t. In this section we will describe several ways to specify a prior for the unknown variances and we will derive algorithms to draw from the posterior distribution of the unknown parameters.

4.7.1 Constant unknown variances: *d* Inverse Gamma prior

This is the simplest model commonly used for unknown variances. We assume for simplicity that the observations are univariate ($m = 1$), although extensions to the multivariate case are easy to devise. The unknown parameters are the precisions $\psi_y, \psi_{\theta,1}, \ldots, \psi_{\theta,p}$. The observation and system variances

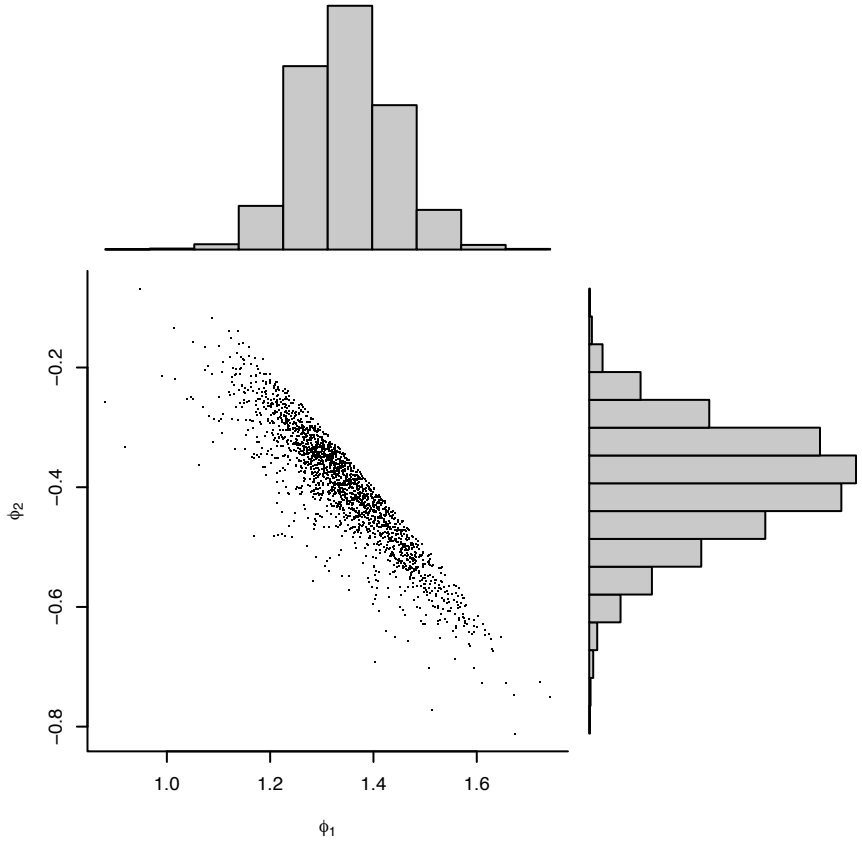

Fig. 4.9. GDP: posterior distribution of AR parameters

are time-invariant and are related to the unknown parameters by the following equalities:

$$V_t = \psi_y^{-1},$$
$$W_t = \mathrm{diag}(\psi_{\theta,1}^{-1}, \ldots, \psi_{\theta,p}^{-1}).$$

The parameters have independent gamma distributions, a priori:

$$\psi_y \sim \mathcal{G}\left(\frac{a_y^2}{b_y}, \frac{a_y}{b_y}\right),$$
$$\psi_{\theta,i} \sim \mathcal{G}\left(\frac{a_{\theta,i}^2}{b_{\theta,i}}, \frac{a_{\theta,i}}{b_{\theta,i}}\right), \qquad i = 1, \ldots, p.$$

As particular cases, this framework includes a Bayesian treatment of nth order polynomial models as well as the Structural Time Series models of Harvey

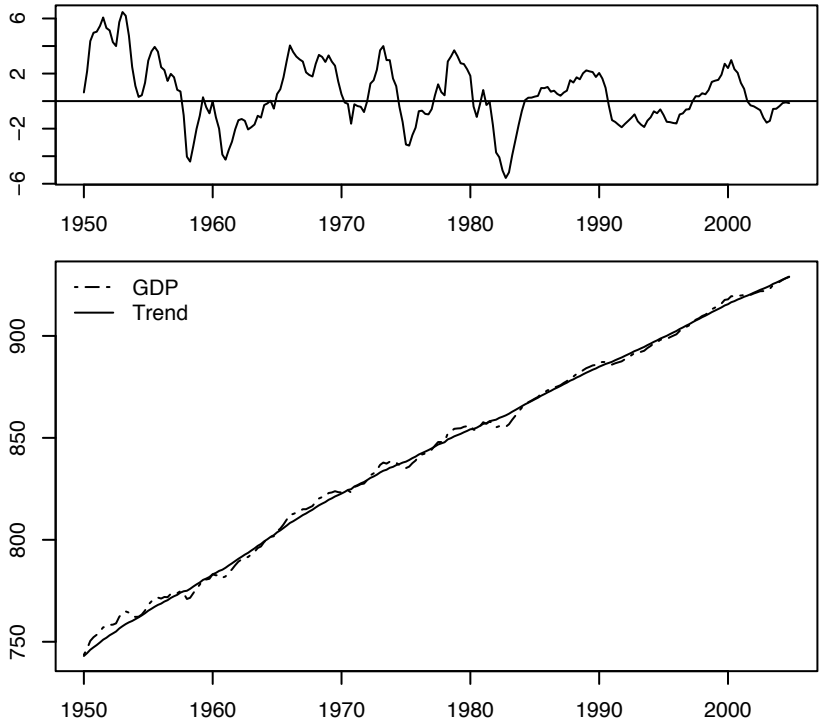

Fig. 4.10. GDP: posterior mean of trend and AR(2) noise

and coauthors. Since each full conditional distribution is proportional to the joint distribution, it is a usful exercise at this point to write down the joint distribution of the observations, states, and unknown parameters.

$$\begin{aligned}
&\pi(y_{1:T}, \theta_{0:T}, \psi_y, \psi_{\theta,1}, \ldots, \psi_{\theta,p}) \\
&\quad = \pi(y_{1:T}|\theta_{0:T}, \psi_y, \psi_{\theta,1}, \ldots, \psi_{\theta,p}) \cdot \pi(\theta_{0:T}|\psi_y, \psi_{\theta,1}, \ldots, \psi_{\theta,p}) \\
&\qquad \cdot \pi(\psi_y, \psi_{\theta,1}, \ldots, \psi_{\theta,p}) \\
&\quad = \prod_{t=1}^{T} \pi(y_t|\theta_t, \psi_y) \cdot \prod_{t=1}^{T} \pi(\theta_t|\theta_{t-1}, \psi_{\theta,1}, \ldots, \psi_{\theta,p}) \cdot \pi(\theta_0) \\
&\qquad \cdot \pi(\psi_y) \cdot \prod_{i=1}^{p} \pi(\psi_{\theta,i}).
\end{aligned}$$

Note that the second product in the factorization can also be written as a product over $i = 1, \ldots, p$, due to the diagonal form of W. This alternative factorization is useful when deriving the full conditional distribution of the $\psi_{\theta,i}$'s. A Gibbs sampler for the d-Inverse-Gamma model draws from the full

conditional distribution of the states and from the full conditional distributions of $\psi_y, \psi_{\theta,1}, \ldots, \psi_{\theta,p}$ in turn. Sampling the states can be done using the FFBS algorithm of Section 4.5. Let us derive the full conditional distribution of ψ_y:

$$\begin{aligned}\pi(\psi_y|\ldots) &\propto \prod_{t=1}^{T} \pi(y_t|\theta_t, \psi_y) \cdot \pi(\psi_y) \\ &\propto \psi_y^{\frac{T}{2}+\frac{a_y^2}{b_y}-1} \exp\left\{-\psi_y \cdot \left[\frac{1}{2}\sum_{t=1}^{T}(y_t - F_t\theta_t)^2 + \frac{a_y}{b_y}\right]\right\}.\end{aligned}$$

Therefore the full conditional of ψ_y is again a gamma distribution,

$$\psi_y|\ldots \sim \mathcal{G}\left(\frac{a_y^2}{b_y} + \frac{T}{2}, \frac{a_y}{b_y} + \frac{1}{2}SS_y\right).$$

with $SS_y = \sum_{t=1}^{T}(y_t - F_t\theta_t)^2$. Similarly, it is easy to show that the full conditionals of the $\psi_{\theta,i}$'s are as follows:

$$\psi_{\theta,i}|\ldots \sim \mathcal{G}\left(\frac{a_{\theta,i}^2}{b_{\theta,i}} + \frac{T}{2}, \frac{a_{\theta,i}}{b_{\theta,i}} + \frac{1}{2}SS_{\theta,i}\right), \qquad i = 1, \ldots, p,$$

with $SS_{\theta,i} = \sum_{t=1}^{T}(\theta_{t,i} - (G_t\theta_{t-1})_i)^2$.

Example. Let us consider again the data on Spain investment (Section 3.2.1). We are going to fit a 2nd-order polynomial model – local linear growth – to the data. The priors for the precisions of the observation and evolution errors are (independent) gamma distributions with means a_y, a_μ, a_β and variances b_y, b_μ, b_β. We decide for the values $a_y = 1$, $a_\mu = a_\beta = 10$, with a common variance equal to 1000, to express a large uncertainty in the prior estimate of the precisions. The function `dlmGibbsDIG` can be called to generate a sample from the posterior distribution of the parameters and the states. The means and variances of the gamma priors are passed to the function via the arguments `a`, `b` (prior mean and variance of observation precision), `alpha`, `beta` (prior mean(s) and variance(s) of evolution precision). Alternatively, the prior distribution can be specified in terms of the usual shape and rate parameters of the gamma distribution. The arguments to pass in this case are `shape.y`, `rate.y`, `shape.theta`, and `rate.theta`. The number of samples from the posterior to generate is determined by the argument `n.sample`, while the logical argument `save.states` is used to determine whether to include the generated unobservable states in the output. In addition, a thinning parameter can be specified via the integer argument `thin`. This gives the number of Gibbs iterations to discard for every saved one. Finally, the data and the model are passed via the arguments `y` and `mod`, respectively. The following display show how `dlmGibbsDIG` works in practice.

R code

```
> invSpain <- ts(read.table("~/Research/DLM/Book/Datasets/invest2.dat",
+                           colClasses = "numeric")[,2]/1000, start = 1960)
> set.seed(5672)
> MCMC <- 12000
> gibbsOut <- dlmGibbsDIG(invSpain, mod = dlmModPoly(2), a = 1, b = 1000,
+                         alpha = 10, beta = 1000, n.sample = MCMC,
+                         thin = 1, save.states = FALSE)
```

Setting *thin = 1* means that the function actually generates a sample of size 24,000 but only kept in the output every other value. In addition, the states are not returned (*save.states = FALSE*). Considering the first 2000 saved iterations as burn in, one can proceed to graphically assess the convergence and mixing properties of the sampler. Figure 4.11 displays a few diagnostic plots obtained from the MCMC output for the variances V, W_{11}, and W_{22}. The first row shows the traces of the sampler, i.e., the simulated values, the second the running ergodic means of the parameters (starting at iteration 500), and the last the estimated autocovariance functions. We obtained the running ergodic means using the function *ergMean*. For example, the plot in the first column and second row was created using the following commands.

R code

```
use <- MCMC - burn
from <- 0.05 * use
plot(ergMean(gibbsOut$dV[-(1:burn)], from), type="l",
     xaxt="n",xlab="", ylab="")
at <- pretty(c(0,use),n=3); at <- at[at>=from]
axis(1, at=at-from, labels=format(at))
```

From a visual assessment of the MCMC output it seems fair to deduce that convergence has been achieved and, while the acf's of the simulated variances do not decay very fast, the ergodic means are nonetheless pretty stable in the last part of the plots. One can therefore go ahead and use the MCMC output to estimate the posterior means of the unknown variances. The function *mcmcMeans* computes the (column) means of a matrix of simulated values, together with an estimate af the Monte Carlo standard deviation, obtained using Sokal's method (Section 1.3).

R code

```
> mcmcMeans(cbind(gibbsOut$dV[-(1:burn),], gibbsOut$dW[-(1:burn),]))
     V.1        V.2        V.3
  0.012197   0.117391   0.329588
 (0.000743) (0.007682) (0.007833)
```

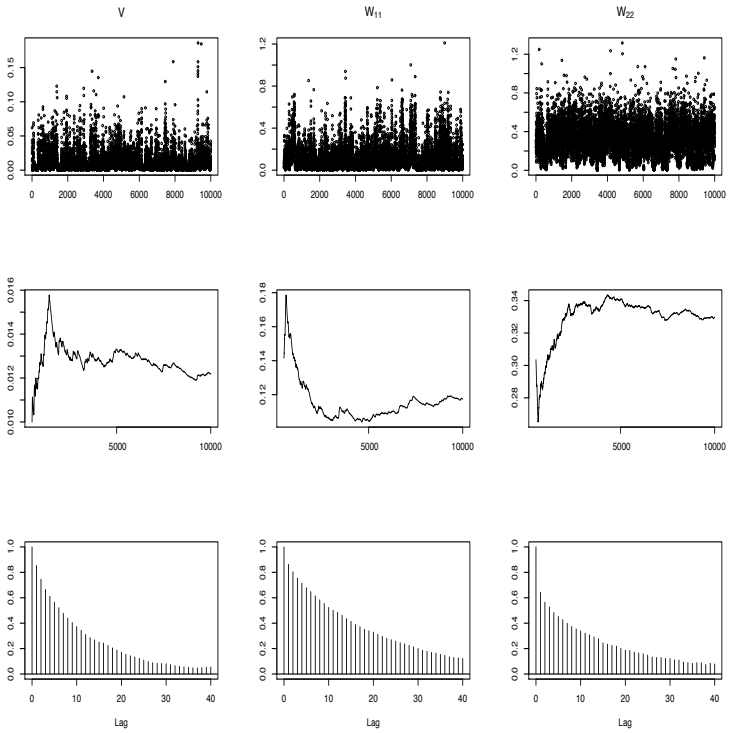

Fig. 4.11. Diagnostic plots for d-inverse-Gamma model applied to Spain investments

Bivariate plots of the simulated parameters may provide additional insight. Consider the plots in Figure 4.12. The joint realizations seem to suggest that

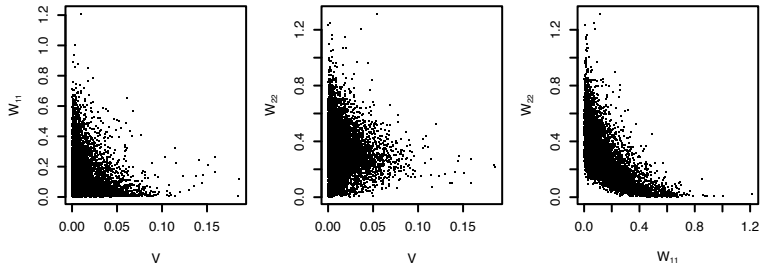

Fig. 4.12. Bivariate plots for d-inverse-Gamma model applied to Spain investments

one, or maybe two, of the model variances may be zero. From the third plot we can see that when W_{11} is close to zero then W_{22} is clearly positive, and vice

versa: W_{11} and W_{22} cannot both be zero. The second plot suggests that this is also true for V and W_{22}. From the first plot it seems that V and W_{11} may be zero, possibly at the same time. In summary, by looking at the bivariate plots, three reduced models come up as alternative to the full model worth exploring: the submodel obtained by setting $V = 0$, the submodel $W_{11} = 0$, the submodel $V = W_{11} = 0$, and the submodel $W_{22} = 0$.

d inverse-Wishart. Multivariate extensions of the d inverse Gamma model can be devised, using independent Wishart priors. Suppose that Y_t is m-variate, $m \geq 1$, and W is block-diagonal with elements $(W_1, \ldots, W_h)$, W_i having dimension $(p_i \times p_i)$. Examples of dlm with a block-diagonal state covariance matrix W include additive compositions of structural models (see section 3....), or SUTSE models (section 3...). Clearly, the case of a general matrix W is obtained by letting $h = 1$.

Again, we parametrize in the precision matrices $\Phi_0 = V^{-1}$ and $\Phi = W^{-1}$, the latter being block-diagonal with elements $\Phi_i = W_i^{-1}$, $i = 1, \ldots, h$. We assume that $\Phi_0, \Phi_1, \ldots, \Phi_h$ have independent Wishart priors, $\Phi_i \sim W(\nu_i, S_i)$, $i = 0, \ldots, h$, where S_i is a symmetric nonsingular *positive definite?* matrix of dimensions $(p_i \times p_i)$, with $p_0 = m$. Then the posterior density $\pi(\theta_{0:T}, \Phi_0, \ldots, \Phi_h | y_{1:T})$ is proportional to

$$\prod_{t=1}^{T} \mathcal{N}(y_t | F_t\theta_t, \Phi_0^{-1})\, \mathcal{N}(\theta_t | G\theta_{t-1}, \Phi^{-1}) \mathcal{N}(\theta_0 | m_0, C_0)\; W(\Phi_0 | \nu_0, S_0) \prod_{i=1}^{h} W(\Phi_i | \nu_i, S_i). \tag{4.22}$$

A Gibbs sampling from π is obtained by iteratively sampling the states $\theta_{0:T}$ (by the FFBS algorithm) and the precisions $\Phi_0, \ldots, \Phi_h$ from their full conditionals. From (4.22) we see that the full conditional density of Φ_i is proportional to

$$\prod_{t=1}^{T} \prod_{j=1}^{h} |\Phi_j|^{1/2} \exp\{-\frac{1}{2}(\theta_t - G_t\theta_{t-1})'\Phi(\theta_t - G_t\theta_{t-1})\}\, |\Phi_i|^{\nu_i - (p_i+1)/2} \exp\{-tr(S_i\Phi_i)\}$$

$$\propto |\Phi_i|^{T/2+\nu_i-(p_i+1)/2} \exp\{-tr(\frac{1}{2}\sum_{t=1}^{T}(\theta_t - G_t\theta_{t-1})(\theta_t - G_t\theta_{t-1})'\Phi) - tr(S_i\Phi_i)\}$$

(see section... page 21). Let

$$SS_t = (\theta_t - G_t\theta_{t-1})(\theta_t - G_t\theta_{t-1})'$$

and partition it in accord with Φ:

$$SS_t = \begin{pmatrix} SS_{11,t} & \cdots & SS_{1h,t} \\ \vdots & \ddots & \vdots \\ SS_{h1,t} & \cdots & SS_{hh,t} \end{pmatrix}.$$

Then $tr(SS_t\Phi) = \sum_{j=1}^{h} tr(SS_{jj,t}\Phi_j)$, so that the full conditional of Φ_i results to be proportional to

$$|\Phi_i|^{T/2+\nu_i-(p_i+1)/2} \exp\{-tr((\frac{1}{2}\sum_{t=1}^{T} SS_{ii,t} + S_i)\Phi_i)\}.$$

That is, for $i = 1, \ldots, h$, the full conditional of Φ_i is Wishart, with parameters $(\nu_i + T/2, 1/2\sum_{t=1}^{T} SS_{ii,t} + S_i)$. In particular, for a dlm obtained by combining components models as in section 3....., we have

$$\theta_t = \begin{pmatrix} \theta_{1,t} \\ \vdots \\ \theta_{h,t} \end{pmatrix}; \quad G_t = \begin{pmatrix} G_{1,t} & \cdots & 0 \\ 0 & \ddots & 0 \\ 0 & \cdots & G_{h,t} \end{pmatrix},$$

with $\theta_{i,t} = (p_i \times 1)$ and $G_{i,t} = (p_i \times p_i)$, so that the full conditional of Φ_i is $W(\nu_i + T/2, 1/2\sum_{t=1}^{T} SS_{ii,t} + S_i)$ where $S_{ii,t} = (\theta_{i,t} - G_{i,t}\theta_{i,t-1})(\theta_{i,t} - G_{i,t}\theta_{i,t-1})'$.

Analogously, one finds that the full conditional of Φ_0 is $W(T/2+\nu_0, 1/2\sum_{t=1}^{T}(y_t - F_t\theta_t)(y_t - F_t\theta_t)' + S_0)$.

4.7.2 λ-ω_t model for outliers and structural breaks

In this subsection we consider a generalization of the d-Inverse-Gamma model that is appropriate to account for outliers and structural breaks. To introduce the model, let us focus on observational outliers first. Structural breaks – or outliers in the state series – will be dealt with in a similar way later on. From the observation equation $y_t = F_t\theta_t + v_t$, we see that a simple way to account for observations that are unusually far from their one-step-ahead predicted value is to replace the Normal distribution of v_t with a heavy-tailed distribution. The Student-t distribution family is particularly appealing in this respect for two reasons. On one hand, it can accommodate, through its degrees-of-freedom parameter, different degrees of heaviness in the tails. On the other hand, the t distribution admits a simple representation as a scale mixture of Normal distributions, which allows one to treat a DLM with t-distributed observation errors as a Gaussian DLM, conditionally on the scale parameters. The obvious advantage is that all the standard algorithms for DLMs – from Kalman filter to FFBS – can still be used, albeit conditionally. In particular, within a Gibbs sampler, one can still draw the states from their full conditional distribution using the FFBS algorithm. We assume that, up to a scale factor λ_y, the v_t have Student-t distributions with $\nu_{y,t}$ degrees of freedom:

$$\lambda_y^{1/2} v_t | \lambda_y, \nu_{y,t} \sim t_{\nu_{y,t}}.$$

Introducing latent variables $\omega_{y,t}$, distributed as $\mathcal{G}(\frac{\nu_{y,t}}{2}, \frac{\nu_{y,t}}{2})$, we can equivalently write:

$$\begin{aligned}\lambda_y^{1/2} v_t|\lambda_y, \omega_{y,t} &\sim \mathcal{N}(0, \omega_{y,t}^{-1}),\\ \omega_{y,t}|\nu_{y,t} &\sim \mathcal{G}\big(\frac{\nu_{y,t}}{2}, \frac{\nu_{y,t}}{2}\big).\end{aligned}$$

The first line can also be written as

$$v_t|\lambda_y, \omega_{y,t} \sim \mathcal{N}(0, (\lambda_y \omega_{y,t})^{-1}).$$

The latent variable $\omega_{y,t}$ in the previous representation can be informally interpreted as the degree of nonnormality of v_t. In fact, taking the $\mathcal{N}(0, \lambda_y^{-1})$ as baseline – corresponding to $\omega_{y,t} = \mathrm{E}(\omega_{y,t}) = 1$, – values of $\omega_{y,t}$ lower than 1 make larger absolute values of v_t more likely. Figure 4.13 shows a plot of the

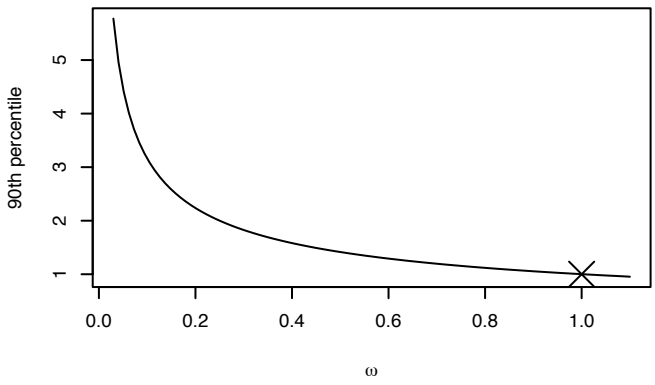

Fig. 4.13. 90th percentile of the conditional distribution of v_t as a function of $\omega_{y,t}$.

90th percentile of the $\mathcal{N}(0, (\lambda_y \omega_{y,t})^{-1})$ as a function of $\omega_{y,t}$ (λ_y is selected so that the percentile is one when $\omega_{y,t}$ is one). From the previous discussion, it follows that the posterior mean of the $\omega_{y,t}$ can be used to flag possible outliers. As a prior for the precision parameter λ_y we choose a Gamma distribution with mean a_y and variance b_y,

$$\lambda_y|a_y, b_y \sim \mathcal{G}\left(\frac{a_y^2}{b_y}, \frac{a_y}{b_y}\right),$$

taking in turn a_y and b_y uniformly distributed on a large, but bounded, interval,

$$\begin{aligned}a_y &\sim \mathit{Unif}(0, A_y),\\ b_y &\sim \mathit{Unif}(0, B_y).\end{aligned}$$

Although the degrees-of-freedom parameter of a Student-t distribution can take any positive real value, we restrict for simplicity the set of possible values to a finite set of integers and set

$$\nu_{y,t}|p_y \sim \mathit{Mult}(1; p_y),$$

where $p_y = (p_{y,1}, \ldots, p_{y,K})$ is a vector of probabilities, the levels of the multinomial distribution are the integers $n_1, \ldots, n_K$, and the $\nu_{y,t}$'s are independent across t. As a convenient, yet flexible choice for $n_1, \ldots, n_K$ we use the set $\{1, 2, \ldots, 10, 20, \ldots, 100\}$. Note that for $\nu_{y,t} = 100$, v_t is approximately normally distributed, given λ_y. As a prior for p_y we adopt a Dirichlet distribution with parameter $\alpha_y = (\alpha_{y,1}, \ldots, \alpha_{y,K})$. A similar hierarchical structure is assumed for each diagonal element of W_t, i.e. for the precision parameters of the state innovations.

In this model the precisions, or, equivalently, the variances, are allowed to be different at different times, although in a way that does not account for a possible correlation in time. In other words, the sequences of precisions at different times are expected to look more like independent, or exchangeable, sequences, rather than time series. For this reason the model is appropriate to account for occasional abrupt changes – corresponding to innovations having a large variance – in the state vector. For example, for polynomial and seasonal factor models an outlier in a component of w_t corresponds to an abrupt change in the corresponding component of the state, such as a jump in the level of the series. However, the modeler do not anticipate this changes to present a clear pattern in time.

Writing $W_{t,i}$ for the ith diagonal element of W_t, the hierarchical prior can be summarized in the following display, in which trivial independence or conditional independence assumptions are not explicitely stated (for example, the $\nu_{\theta,ti}$'s are independent over t and i).

$$
\begin{aligned}
V_t^{-1} &= \lambda_y \omega_{y,t}, \\
W_{t,i}^{-1} &= \lambda_{\theta,i} \omega_{\theta,ti}, \\
\lambda_y | a_y, b_y &\sim \mathcal{G}\left(\frac{a_y^2}{b_y}, \frac{a_y}{b_y}\right), \\
\lambda_{\theta,i} | a_{\theta,i}, b_{\theta,i} &\sim \mathcal{G}\left(\frac{a_{\theta,i}^2}{b_{\theta,i}}, \frac{a_{\theta,i}}{b_{\theta,i}}\right), \\
\omega_{y,t} | \nu_{y,t} &\sim \mathcal{G}\left(\frac{\nu_{y,t}}{2}, \frac{\nu_{y,t}}{2}\right), \\
\omega_{\theta,ti} | \nu_{\theta,ti} &\sim \mathcal{G}\left(\frac{\nu_{\theta,ti}}{2}, \frac{\nu_{\theta,ti}}{2}\right), \\
a_y &\sim \mathit{Unif}(0, A_y), \\
b_y &\sim \mathit{Unif}(0, B_y), \\
a_{\theta,i} &\sim \mathit{Unif}(0, A_{\theta,i}), \\
b_{\theta,i} &\sim \mathit{Unif}(0, B_{\theta,i}), \\
\nu_{y,t} &\sim \mathit{Mult}(1; p_y) \\
\nu_{\theta,ti} &\sim \mathit{Mult}(1; p_{\theta,i}) \\
p_y &\sim \mathit{Dir}(\alpha_y) \\
p_{\theta,i} &\sim \mathit{Dir}(\alpha_{\theta,i}),
\end{aligned}
$$

with $\alpha_{\theta,i} = (\alpha_{\theta,i,1}, \ldots, \alpha_{\theta,i,K})$, $i = 1, \ldots, K$. The levels of all the multinomial distributions are the integers $n_1, \ldots, n_K$.

A Gibbs sampler can be implemented to draw a sample from the posterior distribution of parameters and states of the model specified above. Given all the unknown parameters, the states are generated at once from their joint full conditional distribution using the standard FFBS algorithm. The full conditional distributions of the parameters are easy to derive. We provide here a detailed derivation of the full conditional distribution of λ_y, as an example:

$$
\begin{aligned}
\pi(\lambda_y | \ldots) &\propto \pi(y_{1:T} | \theta_{1:T}, \omega_{y,1:T}, \lambda_y) \cdot \pi(\lambda_y | a_y, b_y) \\
&\propto \prod_{t=1}^{T} \lambda_y^{\frac{1}{2}} \exp\left\{-\frac{\omega_{y,t}\lambda_y}{2}(y_t - F_t\theta_t)^2\right\} \cdot \lambda_y^{\frac{a_y^2}{b_y}-1} \exp\left\{-\lambda_y \frac{a_y}{b_y}\right\} \\
&\propto \lambda_y^{\frac{T}{2}+\frac{a_y^2}{b_y}-1} \exp\left\{-\lambda_y \left[\frac{1}{2}\sum_{t=1}^{T} \omega_{y,t}(y_t - F_t\theta_t)^2 + \frac{a_y}{b_y}\right]\right\}.
\end{aligned}
$$

Therefore,

$$
\lambda_y | \ldots \sim \mathcal{G}\left(\frac{a_y^2}{b_y} + \frac{T}{2}, \frac{a_y}{b_y} + \frac{1}{2} SS_y^*\right),
$$

with $SS_y^* = \sum_{t=1}^{T} \omega_{y,t}(y_t - F_t\theta_t)^2$ The following is a summary of all the full conditional distributions of the unknown parameters.

- λ_y:
$$\lambda_y|\ldots \sim \mathcal{G}\left(\frac{a_y^2}{b_y}+\frac{T}{2}, \frac{a_y}{b_y}+\frac{1}{2}SS_y^*\right),$$
with $SS_y^* = \sum_{t=1}^T \omega_{y,t}(y_t - F_t\theta_t)^2$.
- $\lambda_{\theta,i}, \quad i = 1,\ldots,p$:
$$\lambda_{\theta,i}|\ldots \sim \mathcal{G}\left(\frac{a_{\theta,i}^2}{b_{\theta,i}}+\frac{T}{2}, \frac{a_{\theta,i}}{b_{\theta,i}}+\frac{1}{2}SS_{\theta,i}^*\right),$$
with $SS_{\theta,i}^* = \sum_{t=1}^T \omega_{\theta,ti}(\theta_{ti} - (G_t\theta_{t-1})_i)^2$.
- $\omega_{y,t}, \quad t = 1,\ldots,T$:
$$\omega_{y,t}|\ldots \sim \mathcal{G}\left(\frac{\nu_{y,t}+1}{2}, \frac{\nu_{y,t}+\lambda_y(y_t - F_t\theta_t)^2}{2}\right).$$
- $\omega_{\theta,ti}, \quad i = 1,\ldots,p, \quad t = 1,\ldots,T$:
$$\omega_{\theta,ti}|\ldots \sim \mathcal{G}\left(\frac{\nu_{\theta,ti}+1}{2}, \frac{\nu_{\theta,ti}+\lambda_{\theta,i}(\theta_{ti} - (G_t\theta_{t-1})_i)^2}{2}\right).$$
- (a_y, b_y):
$$\pi(a_y, b_y|\ldots) \propto \mathcal{G}(\lambda_y; a_y, b_y) \qquad \text{on} \quad 0 < a_y < A_y, \quad 0 < b_y < B_y.$$
- $(a_{\theta,i}, b_{\theta,i}), \quad i = 1,\ldots,p$:
$$\pi(a_{\theta,i}, b_{\theta,i}|\ldots) \propto \mathcal{G}(\lambda_{\theta,i}; a_{\theta,i}, b_{\theta,i}) \qquad \text{on} \quad 0 < a_{\theta,i} < A_{\theta,i}, \quad 0 < b_{\theta,i} < B_{\theta,i}.$$
- $\nu_{y,t}, \quad t = 1,\ldots,T$:
$$\pi(\nu_{y,t} = k) \propto \mathcal{G}\left(\omega_{y,t}; \frac{k}{2}, \frac{k}{2}\right) \cdot p_{y,k} \qquad \text{on} \quad \{n_1,\ldots,n_K\}.$$
- $\nu_{\theta,ti}, \quad i = 1,\ldots,p, \quad t = 1,\ldots,T$:
$$\pi(\nu_{\theta,ti} = k) \propto \mathcal{G}\left(\omega_{\theta,ti}; \frac{k}{2}, \frac{k}{2}\right) \cdot p_{\theta,i,k} \qquad \text{on} \quad \{n_1,\ldots,n_K\}.$$
- p_y:
$$p_y|\ldots \sim Dir(\alpha_y + N_y),$$
with $N_y = (N_{y,1},\ldots,N_{y,K})$ and, for each k, $N_{y,k} = \sum_{t=1}^T (\nu_{y,t} = k)$.
- $p_{\theta,i}, \quad i = 1,\ldots,p$:
$$p_{\theta,i}|\ldots \sim Dir(\alpha_{\theta,i} + N_{\theta,i}),$$
with $N_{\theta,i} = (N_{\theta,i,1},\ldots,N_{\theta,i,K})$ and, for each k, $N_{\theta,i,k} = \sum_{t=1}^T (\nu_{\theta,ti} = k)$.

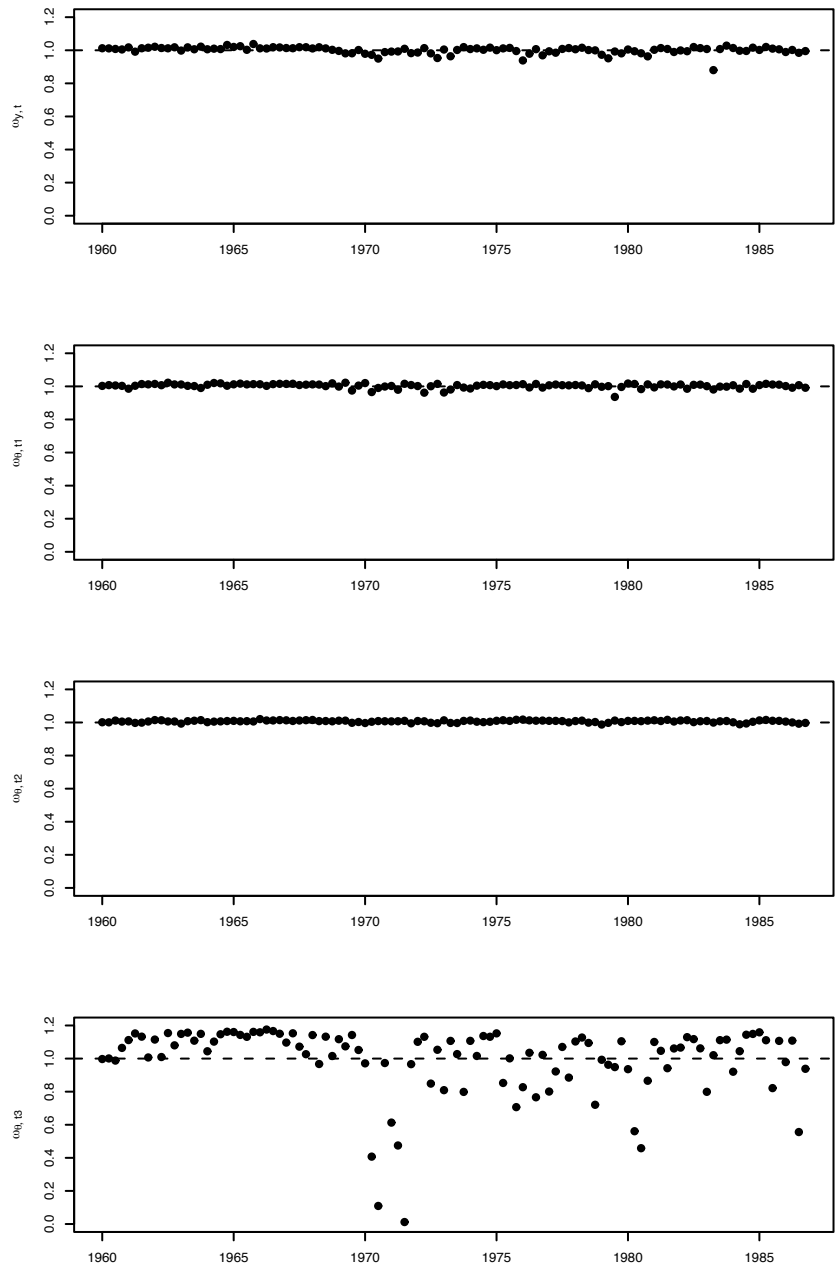

Fig. 4.14. UK gas consumption: posterior means of the ω_t's.

All the full conditional distributions, except for those of $a_y, b_y, a_{\theta,i}, b_{\theta,i}$, are standard. The latter can be drawn from using ARMS. More specifically, we suggest to use ARMS separately on each pair (a, b).

As an example of the use of the $\lambda - \omega_t$ model, consider the time series of quarterly gas consumption in the UK from 1960 to 1986. The data are available in R as *UKgas*. A plot of the data, on the log scale, suggests a possible change in the seasonal factor around the third quarter of 1970. After taking logs, we employ a $\lambda - \omega_t$ model built on a local linear trend plus seasonal component DLM to analyze the data. In this model the five-by-five variance matrix W_t has only three nonzero diagonal elements: the first refers to the level of the series, the second to the slope of the stochastic linear trend, and the third to the seasonal factor. We packaged the entire Gibbs sampler in the function *dlmGibbsDIGt*, available from the book website. The parameters $a_y, b_y, a_{\theta,1}, b_{\theta,1}, \ldots, a_{\theta,3}, b_{\theta,3}$ are taken to be uniform on $(0, 10^5)$, and the parameters of the four Dirichlet distributions of $p_y, p_{\theta,1}, p_{\theta,2}, p_{\theta,3}$ are all equal to 1/19. The posterior analysis is based on 10000 iterations, after a burn-in of 500 iterations. To improve the mixing, two extra sweeps were run between every two saved sweeps, that is, the iterations of the sampler after burn-in were actually 30000.

R code

```
y <- log(UKgas)
set.seed(4521)
MCMC <- 10500
gibbsOut <- dlmGibbsDIGt(y, mod = dlmModPoly(2) + dlmModSeas(4),
                         A_y = 10000, B_y = 10000, p = 3,
                         n.sample = MCMC, thin = 2)
```

Figure 4.14, obtained with the code below, graphically summarizes the posterior means of the $\omega_{y,t}$ and $\omega_{\theta,ti}$, $t = 1, ..108$, $i = 1, 2, 3$.

R code

```
burn <- 1:500
nuRange <- c(1:10, seq(20, 100, by = 10))
omega_theta <- ts(apply(gibbsOut$omega_theta[,,-burn], 1:2, mean),
                  start=start(y), freq=4)
layout(matrix(c(1,2,3,4),4,1))
par(mar = c(5.1, 4.1, 2.1, 2.1))
plot(ts(colMeans(gibbsOut$omega_y[-burn,]), start=start(y), freq=4),
     ylim=c(0,1.2), pch = 16, xlab="",
     ylab=expression(omega[list(y,t)]), type='p')
abline(h=1, lty="dashed")
for (i in 1:3)
{
```

```
    plot(omega_theta[,i], ylim=c(0,1.2), pch = 16, xlab="",
14        ylab=substitute(omega[list(theta,t*i)], list(i=i)),
          type='p')
16    abline(h=1, lty="dashed")
}
```

It is clear that there are no outliers and the trend is fairly stable. The seasonal component, on the other hand, presents several structural breaks, particularly in the first couple of years of the seventies. The most extreme change in the seasonal component happened in the third quarter of 1971, when the corresponding ω_t had an estimated value of 0.012. It can also be seen that after that period of frequent shocks, the overall variability of the seasonal component remained higher than in the first period of observation.

From the output of the Gibbs sampler one can also estimate the unobserved components – trend and seasonal variation – of the series. Figure 4.15 provides a plot of estimated trend and seasonal component, together with 95% probability intervals. An interesting feature of a model with time-specific variances, like the one considered here, is that confidence intervals need not be of constant width – even after accounting for boundary effects. This is clearly seen in the example, where the 95% probability interval for the seasonal component is wider in the period of high instability of the early seventies. The following code was used to obtained the plot.

R code

```
thetaMean <- ts(apply(gibbsTheta,1:2,mean), start=start(y),
2               freq=frequency(y))
LprobLim <- ts(apply(gibbsTheta,1:2,quantile,probs=0.025),
4              start=start(y), freq=frequency(y))
UprobLim <- ts(apply(gibbsTheta,1:2,quantile,probs=0.975),
6              start=start(y), freq=frequency(y))
par(mfrow=c(2,1), mar=c(5.1, 4.1, 2.1, 2.1))
8 plot(thetaMean[,1], xlab="", ylab="Trend")
lines(LprobLim[,1], lty=2); lines(UprobLim[,1], lty=2)
10 plot(thetaMean[,3], xlab="", ylab="Seasonal", type='o')
lines(LprobLim[,3], lty=2); lines(UprobLim[,3], lty=2)
```

4.8 Sequential Monte Carlo

In this Section and the next two we briefly introduce sequential Monte Carlo methods for state space models. Sequential Monte Carlo provides an alternative set of simulation-based algorithms to approximate complicated posterior distributions. It has proved extremely successful when applied to DLMs and

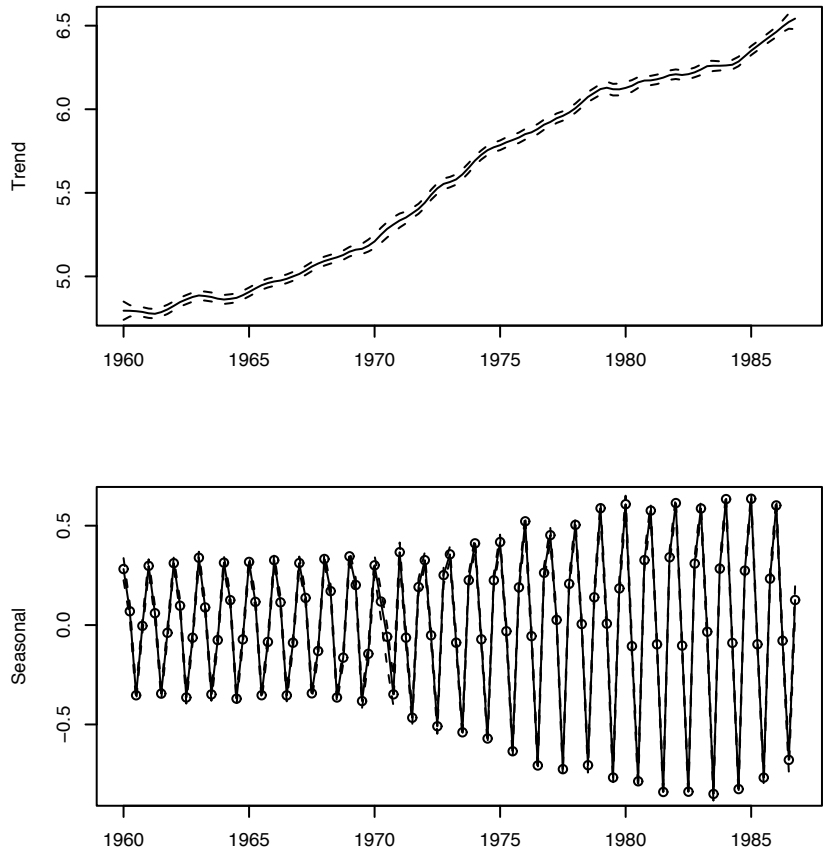

Fig. 4.15. UK gas consumption: trend and seasonal component, with 95% probability intervals.

more general state space models – especially in those applications that require frequent updates of the posterior as new data are observed. Research in sequential Monte Carlo methods is currently very active and we will not try to give here an exhaustive review of the field. Instead, we limit ourselves to a general introduction and a more specific description of a few algorithms that can be easily implemented in the context of DLMs. For more information the interested reader can consult the books by Liu (2001), Doucet et al. (2001), Del Moral (2004), and Cappé et al. (2005).

Particle filtering, which is how sequential Monte Carlo is usually referred to in applications to state space models, is easier to understand when viewed

as an extension of importance sampling. For this reason we open this section with a brief recall of importance sampling.

Suppose one is interested in evaluating the expected value

$$\mathrm{E}_\pi(f(X)) = \int f(x)\pi(x)\mu(dx). \tag{4.23}$$

If g is an *importance density* having the property that $g(x) = 0$ implies $\pi(x) = 0$, then one can write

$$\mathrm{E}_\pi(f(X)) = \int f(x)\frac{\pi(x)}{g(x)}g(x)\mu(dx) = \mathrm{E}_g(f(X)w^\star(X)),$$

where $w^\star(x) = \pi(x)/g(x)$ is the so-called *importance function*. This suggests to approximate the expected value of interest by generating a random sample of size N from g and computing

$$\frac{1}{N}\sum_{i=1}^{N} f(x^{(i)})w^\star(x^{(i)}) \approx \mathrm{E}_\pi(f(X)). \tag{4.24}$$

In Bayesian applications one can typically evaluate the target density only up to a normalizing factor, i.e., only $C \cdot \pi(x)$ can be computed, for an unknown constant C. Unfortunately, this implies that also the importance function can only be evaluated up to the same factor C and (4.24) cannot be used directly. However, letting $\tilde{w}_i = Cw^\star(x^{(i)})$, if one takes $f(x) \equiv C$, then (4.24) yields

$$\frac{1}{N}\sum_{i=1}^{N} Cw^\star(x^{(i)}) = \frac{1}{N}\sum_{i=1}^{N}\tilde{w}_i \approx \mathrm{E}_\pi(C) = C. \tag{4.25}$$

Since the $\tilde{w}_i$'s are available, (4.25) provides a way of evaluating C. Moreover, for the purpose of evaluating (4.23) one does not need an explicit estimate of the constant C: in fact,

$$\begin{aligned}\mathrm{E}_\pi(f(X)) &\approx \frac{1}{N}\sum_{i=1}^{N} f(x^{(i)})w^\star(x^{(i)}) \\ &= \frac{\frac{1}{N}\sum_{i=1}^{N} f(x^{(i)})\tilde{w}_i}{C} \approx \frac{\sum_{i=1}^{N} f(x^{(i)})\tilde{w}_i}{\sum_{i=1}^{N}\tilde{w}_i} \\ &= \sum_{i=1}^{N} f(x^{(i)})w_i,\end{aligned}$$

with $w_i = \tilde{w}_i / \sum_{j=1}^{N}\tilde{w}_j$. Note that: (1) the weights w_i sum to one, and (2) the approximation $\mathrm{E}_\pi(f(X)) \approx \sum_{i=1}^{N} f(x^{(i)})w_i$ holds for every well-behaved function f. Therefore, the sample $x^{(1)}, \ldots, x^{(N)}$ with the associated weights $w_i, \ldots, w_N$ can be viewed as a discrete approximation of the target π. In other

words, writing δ_x for the unit mass at x, and setting $\hat{\pi} = \sum_{i=1}^{N} w_i \delta_{x^{(i)}}$, one has $\pi \approx \hat{\pi}$.

In filtering applications, the target distribution changes every time a new observation is made, moving from $\pi(\theta_{0:t-1}|y_{1:t-1})$ to $\pi(\theta_{0:t}|y_{1:t})$. Note that the former is not a marginal distribution of the latter, even though $\theta_{0:t-1}$ are the first components of $\theta_{0:t}$. The problem then is how to efficiently update a discrete approximation of $\pi(\theta_{0:t-1}|y_{1:t-1})$ when the observation y_t becomes available, in order to obtain a discrete approximation of $\pi(\theta_{0:t}|y_{1:t})$. For every s, let us denote[1] by $\hat{\pi}_s(\theta_{0:s}|y_{1:s})$ the approximation of $\pi(\theta_{0:s}|y_{1:s})$. The updating process consists of two steps: for each point $\theta_{0:t-1}^{(i)}$ in the support of $\hat{\pi}_{t-1}$, (1) draw an additional component $\theta_t^{(i)}$ to obtain $\theta_{0:t}^{(i)}$ and, (2) update its weight $w_{t-1}^{(i)}$ to an appropriate $w_t^{(i)}$. The weighted points $(\theta_t^{(i)}, w_t^{(i)})$, $i = 1, \ldots, N$, provide the new discrete approximation $\hat{\pi}_t$. For every t, let g_t be the importance density used to generate $\theta_{0:t}$. Since at time t the observations $y_{1:t}$ are available, g_t may depend on them and we will write $g_t(\theta_{0:t}|y_{1:t})$ to make the dependence explicit. We assume that g_t can be expressed in the following form:

$$g_t(\theta_{0:t}|y_{1:t}) = g_{t|t-1}(\theta_t|\theta_{0:t-1}, y_{1:t}) \cdot g_{t-1}(\theta_{0:t-1}|y_{1:t-1}).$$

This allows to "grow" sequentially $\theta_{0:t}$ by combining $\theta_{0:t-1}$, drawn from g_{t-1} and available at time $t-1$, and θ_t, generated at time t from $g_{t|t-1}(\theta_t|\theta_{0:t-1}, y_{1:t})$. We will call the functions $g_{t|t-1}$ *importance transition densities*. Note that only the importance transition densities are needed to generate $\theta_{0:t}$. Suggestions about the selection of the importance density are provided at the end of the section. Let us consider next how to update the weights. One has, dropping the superscripts for notational simplicity:

$$\begin{aligned} w_t &\propto \frac{\pi(\theta_{0:t}|y_{1:t})}{g_t(\theta_{0:t}|y_{1:t})} \propto \frac{\pi(\theta_{0:t}, y_t|y_{1:t-1})}{g_t(\theta_{0:t}|y_{1:t})} \\ &\propto \frac{\pi(\theta_t, y_t|\theta_{0:t-1}, y_{1:t-1}) \cdot \pi(\theta_{0:t-1}|y_{1:t-1})}{g_{t|t-1}(\theta_t|\theta_{0:t-1}, y_{1:t}) \cdot g_{t-1}(\theta_{0:t-1}|y_{1:t-1})} \\ &\propto \frac{\pi(y_t|\theta_t) \cdot \pi(\theta_t|\theta_{t-1})}{g_{t|t-1}(\theta_t|\theta_{0:t-1}, y_{1:t})} \cdot w_{t-1}. \end{aligned}$$

Hence, for every i, after drawing $\theta_t^{(i)}$ from $g_{t|t-1}(\theta_t|\theta_{0:t-1}^{(i)}, y_{1:t})$, one can compute the unnormalized weight $\tilde{w}_t^{(i)}$ as

$$\tilde{w}_t^{(i)} = w_{t-1}^{(i)} \cdot \frac{\pi(y_t|\theta_t^{(i)}) \cdot \pi(\theta_t^{(i)}|\theta_{t-1}^{(i)})}{g_{t|t-1}(\theta_t^{(i)}|\theta_{0:t-1}^{(i)}, y_{1:t})}. \tag{4.26}$$

[1] We keep the index s in the notation $\hat{\pi}_s$ because approximations at different times can be in principle unrelated to one another, while the targets are all derived from the unique distribution of the process $\{\theta_i, y_j : i \geq 0, j \geq 1\}$.

- Initialize: draw $\theta_0^{(1)}, \ldots, \theta_0^{(N)}$ independently from $\pi(\theta_0)$ and set $w_0^{(i)} = N^{-1}$, $i = 1, \ldots, N$.
- For $t = 1, \ldots, T$:
  - For $i = 1, \ldots, N$:
    - Draw $\theta_t^{(i)}$ from $g_{t|t-1}(\theta_t | \theta_{0:t-1}^{(i)}, y_{1:t})$ and set

$$\theta_{0:t}^{(i)} = (\theta_{0:t-1}^{(i)}, \theta_t^{(i)})$$

    - Set

$$\tilde{w}_t^{(i)} = w_{t-1}^{(i)} \cdot \frac{\pi(\theta_t^{(i)}, y_t | \theta_{t-1}^{(i)})}{g_{t|t-1}(\theta_t^{(i)} | \theta_{0:t-1}^{(i)}, y_{1:t})}.$$

  - Normalize the weights:

$$w_t^{(i)} = \frac{\tilde{w}_t^{(i)}}{\sum_{j=1}^N \tilde{w}_t^{(j)}}.$$

  - Compute

$$N_{\mathit{eff}} = \left(\sum_{i=1}^N \big(w_t^{(i)}\big)^2 \right)^{-1}.$$

  - If $N_{\mathit{eff}} < N_0$, resample:
    - Draw a sample of size N from the discrete distribution

$$\mathrm{P}\big(\theta_{0:t} = \theta_{0:t}^{(i)}\big) = w_t^{(i)}, \qquad i = 1, \ldots, N,$$

      and relabel this sample

$$\theta_{0:t}^{(1)}, \ldots, \theta_{0:t}^{(N)}.$$

    - Reset the weights: $w_t^{(i)} = N^{-1}$, $i = 1, \ldots, N$.
  - Set $\hat{\pi}_t = \sum_{i=1}^N w_t^{(i)} \delta_{\theta_{0:t}^{(i)}}$.

Table 4.3. Summary of the particle filter algorithm

The fraction on the left-hand side of equation (4.26), or any quantity proportional[2]to it, is called the *incremental weight*. The final step in the updating process consists in scaling the unnormalized weights:

$$w_t^{(i)} = \frac{\tilde{w}_t^{(i)}}{\sum_{j=1}^N \tilde{w}_t^{(j)}}.$$

In practice it is often the case that, after a number of updates have been performed, a few points in the support of $\hat{\pi}_t$ have relatively large weights, while

[2] The proportionality constant may depend on $y_{1:t}$, but should not depend on $\theta_t^{(i)}$ or $\theta_{0:t-1}^{(i)}$ for any i.

all the remaining have negligible weights. This clearly leads to a deterioration in the Monte Carlo approximation. To keep this phenomenon in control, a useful criterion to monitor over time is the *effective sample size*, defined as

$$N_{\mathit{eff}} = \left(\sum_{i=1}^{N} \left(w_t^{(i)} \right)^2 \right)^{-1},$$

which ranges between N (when all the weights are equal) and one (when one weight is equal to one). When N_{eff} falls below a threshold N_0, it is advisable to perform a resampling step. This consists in drawing a random sample of size N from $\hat{\pi}_t$ and using the sampled points, with equal weights, as the new discrete approximation of the target. Table 4.3 contains an algorithmic summary of particle filtering. Let us stress once again the sequential character of the algorithm. Each pass of the outermost "for" loop represents the updating from $\hat{\pi}_{t-1}$ to $\hat{\pi}_t$ following the observation of the new data point y_t. Therefore, at any time $t \leq T$ one has a working approximation $\hat{\pi}_t$ of the current filtering ditribution.

At time t, a discrete approximation of the filtering distribution $\pi(\theta_t|y_{0:t})$ is immediately obtained as a marginal distribution of $\hat{\pi}_t$. More specifically, if $\hat{\pi}_t = \sum_{i=1}^{N} w^{(i)} \delta_{\theta_{0:t}^{(i)}}$, we only need to discard the first t components of each path $\theta_{0:t}^{(i)}$, leaving only $\theta_t^{(i)}$, to obtain

$$\pi(\theta_t|y_{1:t}) \approx \sum_{i=1}^{N} w^{(i)} \delta_{\theta_t^{(i)}}.$$

As a matter of fact, particle filter is most frequently viewed, as the name itself suggests, as an algorithm to update sequentially the filtering distribution. Note that, as long as the transition densities $g_{t|t-1}$ are Markovian, the incremental weights in (4.26) only depend on $\theta_t^{(i)}$ and $\theta_{t-1}^{(i)}$, so that, if the user is only interested in the filtering distribution, the previous components of the path $\theta_{0:t}^{(i)}$ can be safely discarded. This clearly translates into substantial savings in terms of storage space. Another, more fundamental, reason to focus on the filtering distribution is that the discrete approximation provided by $\hat{\pi}_t$ is likely to be more accurate for the most recent components of $\theta_{0:t}$ than for the initial ones. To see why this is the case, consider, for a fixed $s < t$, that the $\theta_s^{(i)}$'s are generated at a time when only $y_{0:s}$ is available, so that they may well be far from the center of their smoothing distribution $\pi(\theta_s|y_{0:t})$, which is conditional on $t - s$ additional observations.

We conclude this section with practical guidelines to follow in the selection of the importance transition densities. In the context of DLM, as well as for more general state space models, two are the most used importance

transition densities. The first is $g_{t|t-1}(\theta_t|\theta_{0:t-1}, y_{1:t}) = \pi(\theta_t|\theta_{t-1})$, i.e., the actual transition density of the Markov chain of the states. It is clear that in this way all the particles are drawn from the prior distribution of the states, without accounting for any information provided by the observations. The simulation of the particles and the calculation of the incremental weights are straightforward. However, most of the times the generated particles will fall in regions of low posterior density. The consequence will be an inaccurate discrete representation of the posterior density and a high Monte Carlo variance for the estimated posterior expected values. For these reasons we discourage the use of the prior as importance density. A more efficient approach, that accounts for the observations in the importance transition densities, consists in generating θ_t from its conditional distribution given θ_{t-1} and y_t. In view of the conditional independence structure of the model, this is the same as the conditional distribution of θ_t given $\theta_{0:t-1}$ and $y_{1:t}$. Therefore, in this way one is generating θ_t from the target (conditional) distribution. However, since θ_{t-1} is not drawn from the target, the particles $\theta_{0:t}^{(i)}$ are not draws from the target distribution[3]and the incremental importance weights need to be evaluated. Applying standard results about Normal models, it is easily seen that the importance transition density $g_{t|t-1}$ is a Normal density with mean and variance given by

$$\begin{aligned} \mathrm{E}(\theta_t|\theta_{t-1}, y_t) &= G_t\theta_{t-1} + W_t F_t' \Sigma_t^{-1}(y_t - F_t G_g \theta_{t-1}), \\ \mathrm{Var}(\theta_t|\theta_{t-1}, y_t) &= W_t - W_t F_t' \Sigma_t^{-1} F_t W_t, \end{aligned}$$

where $\Sigma_t = F_t W_t F_t' + V_t$. Note that for time-invariant DLMs the conditional variance above does not depend on t and can therefore be computed once and for all at the beginning of the process. The incremental weights, using this importance transition density, are proportional to the conditional density of y_t given $\theta_{t-1} = \theta_{t-1}^{(i)}$, i.e., to the $\mathcal{N}(F_t G_t \theta_{t-1}^{(i)}, \Sigma_t)$ density, evaluated at y_t.

4.9 Auxiliary particle filter

For a fully specified DLM, i.e., one containing no unknown paremters, the algorithm discussed in the previous section, employing the optimal transition kernel and resampling whenever the effective sample size falls below a given threshold, typically provides fairly good sequential approximations to the filtering or smoothing distributions. For nonlinear and/or nonnormal state space models the general framework of the previous section still applies, but devising effective importance transition densities is much harder. In fact, the transition density of the optimal kernel may not be available in closed form and one has to devise alternative importance densities. The *auxiliary particle filter* algorithm was proposed by Pitt and Shephard (1999) to overcome

[3] The reason for this apparent paradox is that the target distribution changes from time $t-1$ to time t. When one generates θ_{t-1}, the observation y_t is not used.

this difficulty, in the context of general state space models. While not really needed for fully specified DLMs, an extension of the algorithm, due to Liu and West (2001), turns out to be very useful even in the DLM case when the model contains unknown parameters. For this reason we present Pitt and Shephard's auxiliary particle filter here, followed in the next section by Liu and West's extension to deal with unknown model parameters.

Suppose that at time $t-1$ a discrete approximation $\hat{\pi}_{t-1} = \sum_{i=1}^{N} w_{t-1}^{(i)} \delta_{\theta_{0:t-1}^{(i)}}$ to the joint smoothing distribution $\pi(\theta_{0:t-1}|y_{1:t-1})$ is available. The goal is to update the approximate smoothing distribution when a new data point is observed or, in other words, to obtain a discrete approximation $\hat{\pi}_t$ to the joint smoothing distribution at time t, $\pi(\theta_{0:t}|y_{1:t})$. We have:

$$\begin{aligned}
\pi(\theta_{0:t}|y_{1:t}) &\propto \pi(\theta_{0:t}, y_t|y_{1:t-1}) \\
&= \pi(y_t|\theta_{0:t}, y_{1:t-1}) \cdot \pi(\theta_t|\theta_{0:t-1}, y_{1:t-1}) \cdot \pi(\theta_{0:t-1}|y_{1:t-1}) \\
&= \pi(y_t|\theta_t) \cdot \pi(\theta_t|\theta_{t-1}) \cdot \pi(\theta_{0:t-1}|y_{1:t-1}) \\
&\approx \pi(y_t|\theta_t) \cdot \pi(\theta_t|\theta_{t-1}) \cdot \hat{\pi}_{t-1}(\theta_{0:t-1}) \\
&= \sum_{i=1}^{N} w_{t-1}^{(i)} \pi(y_t|\theta_t) \pi(\theta_t|\theta_{t-1}^{(i)}) \delta_{\theta_{0:t-1}^{(i)}}.
\end{aligned}$$

Note that the last expression is an unnormalized distribution for $\theta_{0:t}$ which is discrete in the first t components and continuous in the last, θ_t. This distribution, which approximates $\pi(\theta_{0:t}|y_{1:t})$, can be taken to be our target for an importance sampling step. The target being a mixture distribution, a standard approach to get rid of the summation is to introduce a latent variable I, taking values in $\{1, \ldots, N\}$, such that:

$$\begin{aligned}
&\mathrm{P}(I = i) = w_{t-1}^{(i)}, \\
&\theta_{0:t}|I = i \sim C\pi(y_t|\theta_t)\pi(\theta_t|\theta_{t-1}^{(i)})\delta_{\theta_{0:t-1}^{(i)}}.
\end{aligned}$$

Thus extended, the target becomes

$$\pi^{\mathrm{aux}}(\theta_{0:t}, i|y_{1:t}) \propto w_{t-1}^{(i)} \pi(y_t|\theta_t)\pi(\theta_t|\theta_{t-1}^{(i)})\delta_{\theta_{0:t-1}^{(i)}}$$

The importance density suggested by Pitt and Shephard for this target is

$$g_t(\theta_{0:t}, i|y_{1:t}) \propto w_{t-1}^{(i)} \pi(y_t|\hat{\theta}_t^{(i)})\pi(\theta_t|\theta_{t-1}^{(i)})\delta_{\theta_{0:t-1}^{(i)}},$$

where $\hat{\theta}_t^{(i)}$ is a central value, such as the mean or the mode, of $\pi(\theta_t|\theta_{t-1} = \theta_{t-1}^{(i)})$. A sample from g_t is easily obtained by iterating, for $k = 1, \ldots, N$, the following two steps.

1. Draw a classification variable I_k, with

$$\mathrm{P}(I_k = i) \propto w_{t-1}^{(i)} \pi(y_t|\hat{\theta}_t^{(i)}), \qquad i = 1, \ldots, N.$$

- Initialize: draw $\theta_0^{(1)}, \ldots, \theta_0^{(N)}$ independently from $\pi(\theta_0)$ and set $w_0^{(i)} = N^{-1}$, $i = 1, \ldots, N$.
- For $t = 1, \ldots, T$:
  - For $k = 1, \ldots, N$:
    - Draw I_k, with $\mathrm{P}(I_k = i) \propto w_{t-1}^{(i)} \pi(y_t | \hat{\theta}_t^{(i)})$.
    - Draw $\theta_t^{(k)}$ from $\pi(\theta_t | \theta_{t-1} = \theta_{t-1}^{(I_k)})$ and set
    $$\theta_{0:t}^{(k)} = \big(\theta_{0:t-1}^{(I_k)}, \theta_t^{(k)}\big).$$
    - Set
    $$\tilde{w}_t^{(k)} = \frac{\pi(y_t | \theta_t^{(k)})}{\pi(y_t | \hat{\theta}_t^{(k)})}.$$
  - Normalize the weights:
  $$w_t^{(i)} = \frac{\tilde{w}_t^{(i)}}{\sum_{j=1}^N \tilde{w}_t^{(j)}}.$$
  - Compute
  $$N_{\mathit{eff}} = \left(\sum_{i=1}^N \big(w_t^{(i)}\big)^2 \right)^{-1}.$$
  - If $N_{\mathit{eff}} < N_0$, resample:
    - Draw a sample of size N from the discrete distribution
    $$\mathrm{P}\big(\theta_{0:t} = \theta_{0:t}^{(i)}\big) = w_t^{(i)}, \qquad i = 1, \ldots, N,$$
    and relabel this sample
    $$\theta_{0:t}^{(1)}, \ldots, \theta_{0:t}^{(N)}.$$
    - Reset the weights: $w_t^{(i)} = N^{-1}$, $i = 1, \ldots, N$.
  - Set $\hat{\pi}_t = \sum_{i=1}^N w_t^{(i)} \delta_{\theta_{0:t}^{(i)}}$.

Table 4.4. Summary of the auxiliary particle filter algorithm

2. Given $I_k = i$, draw
$$\theta_t^{(k)} \sim \pi(\theta_t | \theta_{t-1}^{(i)})$$
and set $\theta_{0:t}^{(k)} = (\theta_{0:t-1}^{(i)}, \theta_t^{(k)})$.

The importance weight of the kth draw from g_t is proportional to

$$\tilde{w}_t^{(k)} = \frac{w_{t-1}^{(I_k)} \pi(y_t | \theta_t^{(k)}) \pi(\theta_t^{(k)} | \theta_{t-1}^{(k)})}{w_{t-1}^{(I_k)} \pi(y_t | \hat{\theta}_t^{(k)}) \pi(\theta_t^{(k)} | \theta_{t-1}^{(k)})} = \frac{\pi(y_t | \theta_t^{(k)})}{\pi(y_t | \hat{\theta}_t^{(k)})}.$$

After normalizing the $\tilde{w}_t^{(k)}$'s and discarding the classification variables I_k's, we finally obtain the discrete approximation to the joint smoothing distribution at time t:

$$\hat{\pi}_t(\theta_{0:t}) = \sum_{i=1}^{N} w_t^{(i)} \delta_{\theta_{0:t}^{(i)}} \approx \pi(\theta_{0:t}|y_{1:t}).$$

As with the standard algorithm of Section 4.8, a resampling step is commonly applied in case the effective sample size drops below a specified threshold. A summary of the auxiliary particle filter is provided in Table 4.4

The main advantage of auxiliary particle filter over the simple direct algorithm described in the previous section consists in the fact that it allows to use the one-step prior distribution $\pi(\theta_t|\theta_{t-1})$ to draw θ_t without loosing much efficiency. Loosely speaking, when drawing from g_t, the role of the first step is to pre-select a conditioning θ_{t-1} that is likely to evolve into a highly plausible θ_t in the light of the new observation y_t. In this way possible conflicts between prior – $\pi(\theta_t|\theta_{t-1})$ – and likelihood – $\pi(y_t|\theta_t)$ – are minimized. While for a DLM deriving the optimal instrumental kernel is straightforward, for a general state space model this is not the case, so that efficiently using the prior distribution, which is almost always available, as importance transition kernel provides a substantial simplification.

4.10 Sequential Monte Carlo with unknown parameters

In real applications the model almost invariably contains unknown parameters that need to be estimated from the data. Denoting again by ψ the vector of unknown parameters, the target distribution at time t for a sequential Monte Carlo algorithm is therefore in this case $\pi(\theta_{0:t}, \psi|y_{1:t})$. As detailed in Section 4.4, a (weighted) sample from the forecast distributions can be easily obtained once a (weighted) sample from the joint posterior distribution is available. On the other hand, the filtering distribution and the posterior distribution of the parameter can be trivially obtained by marginalization. A simple-minded approach to sequential Monte Carlo for a model with an unknown parameter is to extend the state vector to include ψ as part of it, defining the trivial dynamics $\psi_t = \psi_{t-1}$ $(= \psi)$. In this way a relatively simple DLM typically becomes a nonlinear and nonnormal state space model. However, the most serious drawback is that, applying the general algorithm of Section 4.8 (or the auxiliary particle filter of Section 4.9), the values $\psi_t^{(i)}$, $i = 1, \ldots, N$. are those drawn at time $t = 0$, since there is no evolution for this fictitious state. In other words, $\psi_t^{(i)} = \psi_0^{(i)}$ for every i and t, so that the $\psi_t^{(i)}$'s, drawn from the prior distribution, are typically not representative of the posterior distribution at a later time $t > 0$. It is true that, as the particle filter algorithm is sequentially applied, the weights are adjusted to reflect the changes of the target distributions. However, this can only account for the relative weights: if the $\psi_t^{(i)}$'s happen to be all in the tails of the marginal target $\pi(\psi|y_{1:t})$, the discrete approximation provided by the algorithm will always be a poor one. There is, in view of the previous considerations, a need to "refresh" the sampled values of ψ in order to follow the evolution

of the posterior distribution. This can be achieved by discarding the current values of ψ each time the target changes and generating new ones. Among the different available methods, probably the most commonly used is the one proposed by Liu and West (2001) and described below, which extends the auxiliary particle filter. Gilks and Berzuini (2001) and Storvik (2002) propose interesting alternative algorithms.

The idea of Liu and West essentially consists in constructing an approximate target distribution at time t which is continuous not only in θ_t, but also in ψ, so that using importance sampling one draws values of ψ from a continuous importance density, effectively forgetting about the values of ψ used in the discrete approximation at time $t-1$. Consider the discrete approximation available at time $t-1$:

$$\hat{\pi}_{t-1}(\theta_{0:t-1}, \psi) = \sum_{i=1}^{N} w_{t-1}^{(i)} \delta_{(\theta_{0:t-1}^{(i)}, \psi^{(i)})} \approx \pi(\theta_{0:t-1}, \psi | y_{0:t-1}).$$

Marginally,

$$\hat{\pi}_{t-1}(\psi) = \sum_{i=1}^{N} w_{t-1}^{(i)} \delta_{\psi^{(i)}} \approx \pi(\psi | y_{0:t-1}).$$

Liu and West suggest to replace each point mass $\delta_{\psi^{(i)}}$ with a Normal distribution, so that the resulting mixture becomes a continuous distribution. The most natural way of doing so would be to replace $\delta_{\psi^{(i)}}$ with a Normal centered at $\psi^{(i)}$. However, while preserving the mean, this would increase the variance of the approximating distribution. To see that this is the case, let $\bar{\psi}$ and V be the mean vector and variance matrix of ψ under $\hat{\pi}_{t-1}$, and let

$$\tilde{\pi}_{t-1}(\psi) = \sum_{i=1}^{N} w_{t-1}^{(i)} \mathcal{N}(\psi; \psi^{(i)}, \Lambda).$$

Introducing a latent classification variable I for the component of the mixture an observation comes from, we have

$$\begin{aligned} \mathrm{E}(\psi) &= \mathrm{E}(\mathrm{E}(\psi|I)) = \mathrm{E}(\psi^{(I)}) \\ &= \sum_{i=1}^{N} w_{t-1}^{(i)} \psi^{(i)} = \bar{\psi}; \\ \mathrm{Var}(\psi) &= \mathrm{E}(\mathrm{Var}(\psi|I)) + \mathrm{Var}(\mathrm{E}(\psi|I)) \\ &= \mathrm{E}(\Lambda) + \mathrm{Var}(\psi^{(I)}) \\ &= \Lambda + V > V, \end{aligned}$$

where expected values and variances are with respect to $\tilde{\pi}_{t-1}$. However, by changing the definition of $\tilde{\pi}_{t-1}$ to

$$\tilde{\pi}_{t-1}(\psi) = \sum_{i=1}^{N} w_{t-1}^{(i)} \mathcal{N}(\psi; m^{(i)}, h^2 V),$$

with $m^{(i)} = a\psi^{(i)} + (1-a)\bar{\psi}$ for some a in $(0,1)$ and $a^2 + h^2 = 1$, we have

$$\begin{aligned}
\mathrm{E}(\psi) &= \mathrm{E}(\mathrm{E}(\psi|I)) = \mathrm{E}(a\psi^{(I)} + (1-a)\bar{\psi}) \\
&= a\bar{\psi} + (1-a)\bar{\psi} = \bar{\psi}; \\
\mathrm{Var}(\psi) &= \mathrm{E}(\mathrm{Var}(\psi|I)) + \mathrm{Var}(\mathrm{E}(\psi|I)) \\
&= \mathrm{E}(h^2 V) + \mathrm{Var}(a\psi^{(I)} + (1-a)\bar{\psi}) \\
&= h^2 V + a^2 \mathrm{Var}(\psi^{(I)}) = h^2 V + a^2 V = V.
\end{aligned}$$

Thus, ψ has the same first and second moment under $\tilde{\pi}_{t-1}$ and $\hat{\pi}_{t-1}$. Albeit this is true for any a in $(0,1)$, in practice Liu and West recommend to set $a = (3\delta - 1)/(2\delta)$ for a "discount factor" δ in $(0.95, 0.99)$, which corresponds to an a in $(0.974, 0.995)$. The very same idea can be applied even in the presence of $\theta_{0:t-1}$ to the discrete distribution $\hat{\pi}_{t-1}(\theta_{0:t-1}, \psi)$, leading to the extension of $\tilde{\pi}_{t-1}$ to a joint distribution for $\theta_{0:t-1}$ and ψ:

$$\tilde{\pi}_{t-1}(\theta_{0:t-1}, \psi) = \sum_{i=1}^{N} w_{t-1}^{(i)} \mathcal{N}(\psi; m^{(i)}, h^2 V) \delta_{\theta_{0:t-1}^{(i)}}.$$

Note that $\tilde{\pi}_{t-1}$ is discrete in $\theta_{0:t-1}$, but continuous in ψ. From this point onward, the method parallels the development of the auxiliary particle filter. After the new data point y_t is observed, the distribution of interest becomes

$$\begin{aligned}
\pi(\theta_{0:t}, \psi|y_{1:t}) &\propto \pi(\theta_{0:t}, \psi, y_t|y_{1:t-1}) \\
&= \pi(y_t|\theta_{0:t}, \psi, y_{1:t-1}) \cdot \pi(\theta_t|\theta_{0:t-1}, \psi, y_{1:t-1}) \cdot \pi(\theta_{0:t-1}, \psi|y_{1:t-1}) \\
&= \pi(y_t|\theta_t, \psi) \cdot \pi(\theta_t|\theta_{t-1}, \psi) \cdot \pi(\theta_{0:t-1}, \psi|y_{1:t-1}) \\
&\approx \pi(y_t|\theta_t, \psi) \cdot \pi(\theta_t|\theta_{t-1}, \psi) \cdot \tilde{\pi}_{t-1}(\theta_{0:t-1}, \psi) \\
&= \sum_{i=1}^{N} w_{t-1}^{(i)} \pi(y_t|\theta_t, \psi) \pi(\theta_t|\theta_{t-1}^{(i)}, \psi) \mathcal{N}(\psi; m^{(i)}, h^2 V) \delta_{\theta_{0:t-1}^{(i)}}.
\end{aligned}$$

Similarly to what we did in Section 4.9, we can introduce an auxiliary classification variable I such that:

$$\begin{aligned}
\mathrm{P}(I = i) &= w_{t-1}^{(i)}, \\
\theta_{0:t}, \psi | I = i &\sim C\pi(y_t|\theta_t, \psi) \pi(\theta_t|\theta_{t-1}^{(i)}, \psi) \mathcal{N}(\psi; m^{(i)}, h^2 V) \delta_{\theta_{0:t-1}^{(i)}}.
\end{aligned}$$

Note that the conditional distribution in the second line is continuous in θ_t and ψ, and discrete in $\theta_{0:t-1}$ – in fact, degenerate on $\theta_{0:t-1}^{(i)}$. With the introduction of the random variable I, the auxiliary target distribution for the importance sampling update becomes

$$\pi^{\mathrm{aux}}(\theta_{0:t}, \psi, i|y_{1:t}) \propto w_{t-1}^{(i)} \pi(y_t|\theta_t, \psi) \pi(\theta_t|\theta_{t-1}^{(i)}, \psi) \mathcal{N}(\psi; m^{(i)}, h^2 V) \delta_{\theta_{0:t-1}^{(i)}}.$$

As an importance density, a convenient choice is

- Initialize: draw $(\theta_0^{(1)}, \psi^{(1)}), \ldots, (\theta_0^{(N)}, \psi^{(N)})$ independently from $\pi(\theta_0)\pi(\psi)$. Set $w_0^{(i)} = N^{-1}$, $i = 1, \ldots, N$, and

$$\hat{\pi}_0 = \sum_{i=1}^{N} w_0^{(i)} \delta_{(\theta_0^{(i)}, \psi^{(i)})}.$$

- For $t = 1, \ldots, T$:
  - Compute $\bar{\psi} = \mathrm{E}_{\hat{\pi}_{t-1}}(\psi)$ and $V = \mathrm{Var}_{\hat{\pi}_{t-1}}(\psi)$. Set

$$m^{(i)} = a\psi^{(i)} + (1-a)\bar{\psi}, \qquad i = 1, \ldots, N.$$

  - For $k = 1, \ldots, N$:
    - Draw I_k, with $\mathrm{P}(I_k = i) \propto w_{t-1}^{(i)} \pi(y_t | \theta_t = \hat{\theta}_t^{(i)}, \psi = m^{(i)})$.
    - Draw $\psi^{(k)}$ from $\mathcal{N}(m^{(I_k)}, h^2 V)$.
    - Draw $\theta_t^{(k)}$ from $\pi(\theta_t | \theta_{t-1} = \theta_{t-1}^{(I_k)}, \psi = \psi^{(k)})$ and set

$$\theta_{0:t}^{(k)} = \big(\theta_{0:t-1}^{(I_k)}, \theta_t^{(k)}\big).$$

    - Set

$$\tilde{w}_t^{(k)} = \frac{\pi(y_t | \theta_t = \theta_t^{(k)}, \psi = \psi^{(k)})}{\pi(y_t | \theta_t = \hat{\theta}_t^{(k)}, \psi = m^{(I_k)})}.$$

  - Normalize the weights:

$$w_t^{(i)} = \frac{\tilde{w}_t^{(i)}}{\sum_{j=1}^{N} \tilde{w}_t^{(j)}}.$$

  - Compute

$$N_{\mathit{eff}} = \left(\sum_{i=1}^{N} \big(w_t^{(i)}\big)^2 \right)^{-1}.$$

  - If $N_{\mathit{eff}} < N_0$, resample:
    - Draw a sample of size N from the discrete distribution

$$\mathrm{P}\big((\theta_{0:t}, \psi) = (\theta_{0:t}^{(i)}, \psi^{(i)})\big) = w_t^{(i)}, \qquad i = 1, \ldots, N,$$

      and relabel this sample

$$(\theta_{0:t}^{(1)}, \psi^{(1)}), \ldots, (\theta_{0:t}^{(N)}, \psi^{(N)}).$$

    - Reset the weights: $w_t^{(i)} = N^{-1}$, $i = 1, \ldots, N$.
  - Set $\hat{\pi}_t = \sum_{i=1}^{N} w_t^{(i)} \delta_{(\theta_{0:t}^{(i)}, \psi^{(i)})}$.

Table 4.5. Summary of Liu and West's algorithm

$$g_t(\theta_{0:t}, \psi, i|y_{1:t}) \propto w_{t-1}^{(i)} \pi(y_t|\theta_t = \hat{\theta}_t^{(i)}, \psi = m^{(i)}) \pi(\theta_t|\theta_{t-1}^{(i)}, \psi) \mathcal{N}(\psi; m^{(i)}, h^2 V) \delta_{\theta_{0:t-1}^{(i)}},$$

where $\hat{\theta}_t^{(i)}$ is a central value, such as the mean or the mode, of $\pi(\theta_t|\theta_{t-1} = \theta_{t-1}^{(i)}, \psi = m^{(i)})$. A sample from g_t can be obtained by iterating, for $k = 1, \ldots, N$, the following three steps.

1. Draw a classification variable I_k, with
$$\mathrm{P}(I_k = i) \propto w_{t-1}^{(i)} \pi(y_t|\theta_t = \hat{\theta}_t^{(i)}, \psi = m^{(i)}), \qquad i = 1, \ldots, N.$$
2. Given $I_k = i$, draw $\psi \sim \mathcal{N}(m^{(i)}, h^2 V)$ and set $\psi^{(k)} = \psi$.
3. Given $I_k = i$ and $\psi = \psi^{(k)}$, draw
$$\theta_t^{(k)} \sim \pi(\theta_t|\theta_{t-1} = \theta_{t-1}^{(i)}, \psi = \psi^{(k)})$$
and set $\theta_{0:t}^{(k)} = (\theta_{0:t-1}^{(i)}, \theta_t^{(k)})$.

The importance weight of the kth draw from g_t is proportional to

$$\begin{aligned}\tilde{w}_t^{(k)} &= \frac{w_{t-1}^{(I_k)} \pi(y_t|\theta_t = \theta_t^{(k)}, \psi = \psi^{(k)}) \pi(\theta_t^{(k)}|\theta_{t-1}^{(k)}, \psi^{(k)}) \mathcal{N}(\psi^{(k)}; m^{(I_k)}, h^2 V)}{w_{t-1}^{(I_k)} \pi(y_t|\theta_t = \hat{\theta}_t^{(k)}, \psi = m^{(I_k)}) \pi(\theta_t^{(k)}|\theta_{t-1}^{(k)}, \psi^{(k)}) \mathcal{N}(\psi^{(k)}; m^{(I_k)}, h^2 V)} \\ &= \frac{\pi(y_t|\theta_t = \theta_t^{(k)}, \psi = \psi^{(k)})}{\pi(y_t|\theta_t = \hat{\theta}_t^{(k)}, \psi = m^{(I_k)})}.\end{aligned}$$

Renormalizing the weights, we obtain the approximate joint posterior distribution at time t

$$\hat{\pi}_t(\theta_{0:t}, \psi) = \sum_{i=1}^{N} w_t^{(i)} \delta_{(\theta_{0:t}^{(i)}, \psi^{(i)})} \approx \pi(\theta_{0:t}, \psi|y_{1:t}).$$

As usual, a resampling step can be applied whenever the effective sample size drops below a specified threshold. Table 4.5 provides a convenient summary of the algorithm.

As a final remark, let us point out that, in order for the mixture of normals approximation of the posterior distribution at time t to make sense, the parameter ψ has to be expressed in a form which is consistent with such a distribution – in particular, the support of a one-dimensional parameter must be the entire real line. For example, variances can be parametrized in terms of their log, probabilities in terms of their logit, and so on.

5

Further developments and advanced examples

In Chapter 4 we have discussed the basic issues of Bayesian analysis for DLMs, and here we present further examples, in particular for multivariate time series. Indeed, the Bayesian approach is particularly effective in treating multivariate series, allowing to model quite naturally the dependence structure of the data. Furthermore, even in complex models, computations can be developed using modern Monte Carlo techniques. In fact, the development of powerful computational tools allows to move from linear Gaussian models to more general state space models. In the last section, we will briefly discuss some more advanced applications and directions for developments.

5.1 Missing data

Gibbs sampling with missing data

5.2 Model selection/comparison

5.3 Multivariate models

5.3.1 Time series of cross sectional models

Bayesian inference for SUTSE, hierarchical DLM, other examples...

Example. Let us consider again the data on Spain and Denmark investments (Section 3.3.2). We are going to fit a SUTSE model with both W_μ and W_β not zero. The prior for the precisions $\Phi_0 = V^{-1}$,$\Phi_1 = W_\mu^{-1}$ and $\phi_2 = W_\beta^{-1}$ are (independent) Wishart distributions with parameters S_0 and ν_0, S_1 and ν_1, S_2 and ν_2, respectively. In this specific case we assume $\nu_0 = \nu_1 = \nu_2 = 2$ and $S_0 = S_1 = S_2$ with S_0^{-1} equal to sample covariance between $(y_{t1}, t = 1, 2, \cdots)$ and $(y_{t2}, t = 1, 2, \cdots)$.
The full conditional of Φ_0 is $W(\nu_0 + T/2, S_0 + 1/2 \sum_{t=1}^{T} (y_t - F\theta_t)(y_t - F\theta_t)')$

and the full conditional of Φ_i, for $i = 1, 2$, is $W(\nu_i + T/2, S_i + 1/2 \sum_{t=1}^{T} SS_{ii,t})$ where $SS_{ii,t}$ is defined in Section ??. The Gibbs sampling generates in turn the state vectors θ_t, V, W_μ and W_β. We set the number of MCMC samples to 10,000 and we remove the first 1000 iterations as burn-in. The ergodic means of all parameters (not displayed here) show that the convergence has been achieved. In the display below are given for the parameters V, W_μ, W_β the estimates of the posterior means, their estimated standard errors, obtained by using the function *mcmcMeans*.

R code

```
> meanV <- round(cbind(mcmcMeans(gibbsV[-burn,1,1]),
+            mcmcMeans(gibbsV[-burn,2,2]),
+            mcmcMeans(gibbsV[-burn,2,1])),4);meanV
        [,1]       [,2]     [,3]
mean 24.2210 34857.4228 304.7083 sd    0.0904   125.5578   2.3393
> meanWmu <- round(cbind(mcmcMeans(gibbsWmu[-burn,1,1]),
+             mcmcMeans(gibbsWmu[-burn,2,2]),
+             mcmcMeans(gibbsWmu[-burn,2,1])),4);meanWmu
        [,1]       [,2]     [,3]
mean 12.4505 29523.7481 121.1562 sd    0.0450   106.3611   1.5000
> meanWbeta <- round(cbind(mcmcMeans(gibbsWbeta[-burn,1,1]),
+               mcmcMeans(gibbsWbeta[-burn,2,2]),
+               mcmcMeans(gibbsWbeta[-burn,2,1])),4);meanWbeta
        [,1]       [,2]     [,3]
mean 12.9639 50877.1045 131.7372 sd    0.0470   178.6172   2.0368
```

Figures 5.1 and 5.2 show the Bayesian estimate of the level for Denmark investment and Spain investment respectively.

5.3.2 Conditionally Gaussian DLMs

GARCH??

5.3.3 Factor models

Example. We extract the common stochastic trend from the federal funds rate (short rate) and 30-year conventional fixed mortgage rate (long rate), obtained from Federal Reserve Bank of St. Louis (http://research.stlouisfed.org/fred2/). (Y. et al.; 2005, See). These series are sampled at weekly intervals over the period April 7, 1971 trough September 8, 2004, and we transform them by taking the natural logarithm one plus the interest rate. The transformed series are illustrated in Figure 5.3 To extract a common trend in the bivariate time series $((y_{t1}, y_{t2}) : t \geq 1)$, we assume the following factor model:

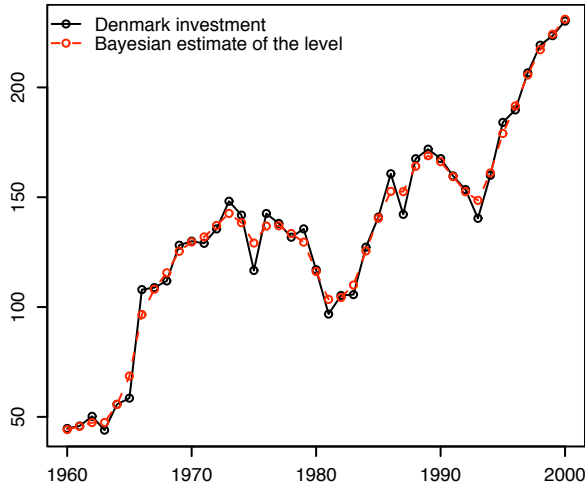

Fig. 5.1. Bayesian estimate of the level for Denmark investment

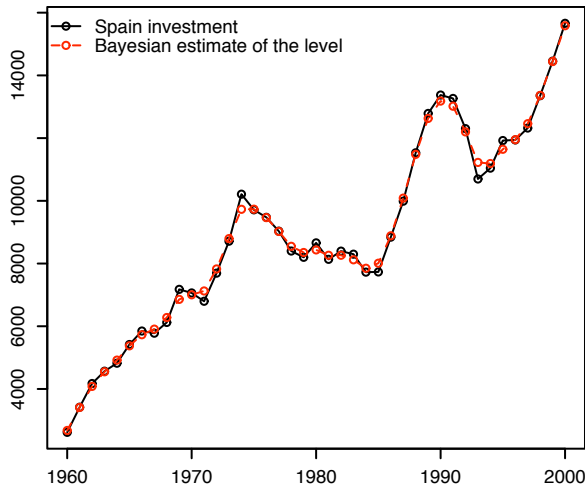

Fig. 5.2. Bayesian estimate of the level for Spain investment

$$\begin{cases} Y_t = A\mu_t + \mu_0 + v_t, & v_t \sim \mathcal{N}(0, V), \\ \mu_t = \mu_{t-1} + w_t, & w_t \sim \mathcal{N}(0, \sigma_w^2), \end{cases} \tag{5.1}$$

where the 2 by 1 matrix A is set to be $A = \begin{bmatrix} 1 & \alpha \end{bmatrix}'$ for ensuring the parameters identification and $\mu_0 = \begin{bmatrix} 1 & \bar{\mu} \end{bmatrix}'$ The latent variable μ_t is defined as a random walk and may be regarded as a common stochastic trend in Y_t.
In the usual DLM notation the model can be written in the form

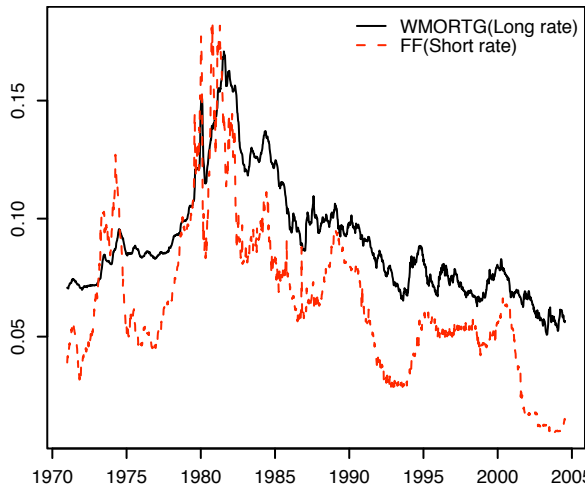

Fig. 5.3. Log of one plus the federal funds rate (FF) and the 30-year mortgage rate (WMORTG)

$$\begin{cases} Y_t = \begin{bmatrix} 1 & 0 \\ \alpha & 1 \end{bmatrix} \theta_t + v_t, \\ \theta_t = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \theta_{t-1} + \begin{bmatrix} w_t \\ 0 \end{bmatrix}, \end{cases} \tag{5.2}$$

with

$$V = \begin{bmatrix} V_{11} & V_{12} \\ V_{12} & V_{22} \end{bmatrix},$$
$$W = \text{diag}(\sigma_\mu^2, 0)$$

The prior we select for α is a $\mathcal{N}(0, 4^2)$. For the precision $(\sigma_\mu^2)^{-1}$ and V^{-1} we assume independent gamma prior distribution $\mathcal{G}(\frac{a^2}{b}, \frac{a}{b})$ and Wishart prior distribution with parameter S_0 and ν_0 degrees of freedom, respectively. In this specific case, we set $a = 1$, $b = 1000$, $\nu_0 = 2$ and S_0^{-1} equal to sample covariance between $(y_{t1}, t = 1, 2, \cdots)$ and $(y_{t2}, t = 1, 2, \cdots)$. A hybrid sampler can draw in turn the α parameter from $\pi(\alpha|\sigma_\mu^2, V, y_{1:T})$, the states, the precision V^{-1} from its full conditional distribution given the states, the observations and α parameter

$$W\left(\nu_0 + \frac{T}{2}, +S_0 + \frac{1}{2}\sum_{t=1}^{T}(y_t - F\theta_t)(y_t - F\theta_t)'\right)$$

and the precision $(\sigma_\mu^2)^{-1}$ from its full conditional distribution given the states and the observations

$$\mathcal{G}\left(\frac{a^2}{b} + \frac{T}{2}, \frac{a}{b} + \frac{1}{2}\sum_{t=1}^{T}(\theta_{t,1} - (G\theta_{t-1})_1)^2\right)$$

Generating a sample of size 5000 and discarding the first 500 draws as burn in, we look some diagnostic plots. We plot the running ergodic means, obtained by using the function ***ergmean***, of the simulated parameter α, of the simulated standard deviation σ_μ, of the simulated standard deviations $V_{11}^{1/2}$ and $V_{22}^{1/2}$ (Figure 5.4). The estimates appear stable in the last part of the plot. In the

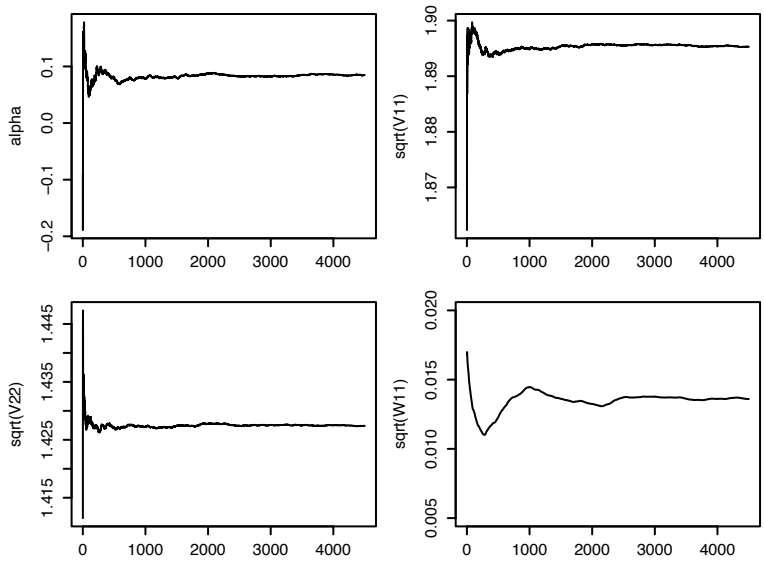

Fig. 5.4. Running ergodic means

display below are given for the parameters α, V and σ_μ^2 the estimates of the posterior means, their estimated standard errors, obtained by using the function *mcmcMeans*, and their equal-tail 90% probability intervals.

R code

```
> round(mcmcMeans(gibbsAlpha[-burn,]),4)
       [,1]
   mean 0.0849
   sd   0.0047
> round(quantile(gibbsAlpha[-burn,], probs=c(0.05,0.95)),4)
     5%     95%
 -0.4474  0.6189
> round(mcmcMeans(sqrt(gibbsV[-burn,1,1])),4)
       [,1]
  mean 1.8953
  sd   0.0005
> round(quantile(sqrt(gibbsV[-burn,1,1]), probs=c(0.05,0.95)),4)
```

```
    5%    95%
  1.8447  1.9476
> round(mcmcMeans(sqrt(gibbsV[-burn,2,2])),4)
       [,1]
  mean 1.4274
  sd   0.0003
> round(quantile(sqrt(gibbsV[-burn,2,2]), probs=c(0.05,0.95)),4)
    5%    95%
  1.3898 1.4676
> round(mcmcMeans(gibbsV[-burn,2,1]),4)
       [,1]
  mean -2.3166
  sd    0.0012
> round(quantile(gibbsV[-burn,2,1], probs=c(0.05,0.95)),4)
     5%     95%
  -2.4567  -2.1828
> round(mcmcMeans(sqrt(gibbsW[-burn,])),4)
       [,1]
  mean 0.0136
  sd   0.0005
> round(quantile(sqrt(gibbsW[-burn,]), probs=c(0.05,0.95)),4)
    5%    95%
0.0099 0.0180
```

The Figure 5.5 displays graphically the posterior mean of the common stochastic trend, together with the data. Note that the common trend is very similar

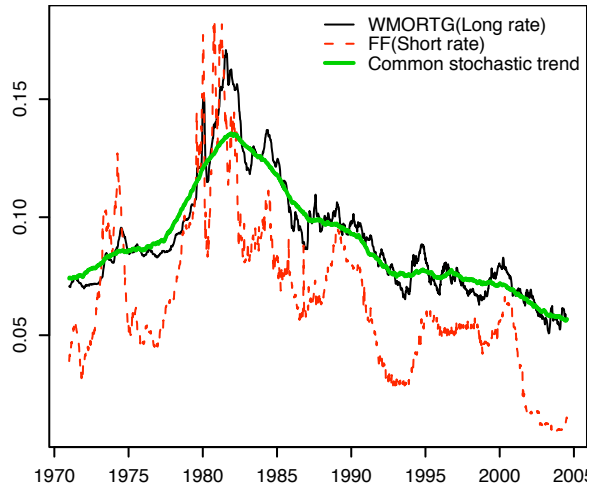

Fig. 5.5. Posterior mean of the common stochastic trend

to the long rate.
Moreover from the equations (5.2) we obtain

$$Y_{t2} = \alpha(Y_{t1} - v_{t1}) + \bar{\mu} + v_{t2}$$

Then the long-run equilibrium relation of the components of Y_t is the stationary component

$$[-\alpha\ 1] \begin{bmatrix} Y_{t1} \\ Y_{t2} \end{bmatrix} = v_{t2} - \alpha v_{t1} + \bar{\mu}$$

where $\beta = \begin{bmatrix} -\alpha \\ 1 \end{bmatrix}$ is the cointegrating vector. The Figure 5.6 displays the mean posterior stationary component. Note that this component resembles the short rate and does not appear to be completely stationary.

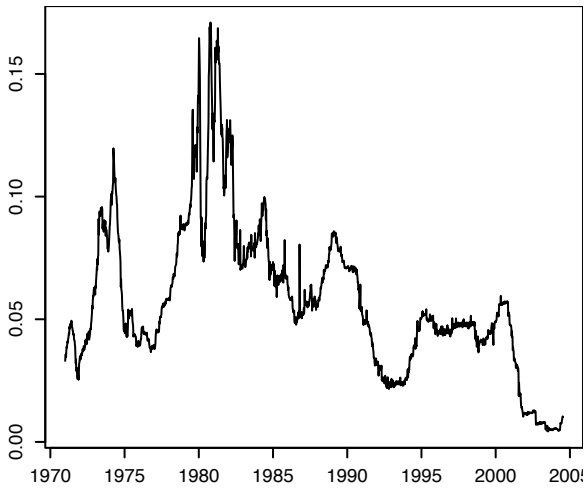

Fig. 5.6. Stationary component

5.3.4 Bayesian VAR

Bayesian inference for VAR models ...

5.4 Further topics...

Here, we just give a brief overview with references – see also Chapter 2.

- Non linear, non gaussian state space models
- Stochastic volatility models
- Hidden Markov models
- Processes in continuous time

References

Akaike, H. (1974a). Markovian representation of stochastic processes and its application to the analysis of autoregressive moving average processes, *Annals of the Institute of Statistical Mathematics* **26**: 363–387.

Akaike, H. (1974b). Stochastic theory of minimal realization, *IEEE Trans. on Automatic Control* **19**: 667–674.

Amisano, G. and Giannini, C. (1997). *Topics in Structural VAR Econometrics*, 2nd edn, Springer, Berlin.

Anderson, B. and Moore, J. (1979). *Optimal Filtering*, Prentice-Hall, Englewood Cliffs.

Barndorff-Nielsen, O., Cox, D. and Klüppelberg, C. (eds) (2001). *Complex stochastic systems*, Chapman & Hall.

Barra, J. and Herbach, L. H. (1981). *Mathematical Basis of Statistics*, Academic Press.

Bawens, L., Lubrano, M. and Richard, J.-F. (1999). *Bayesian inference in dynamic econometric models*, Oxford University Press, N.Y.

Bayes, T. (1763). *An essay towards solving a problem in the doctrin of chances.* Published posthumously in *Phil. Trans. Roy. Stat. Soc. London*, **53**, 370–418 and **54**, 296–325. Reprinted in *Biometrika* **45** (1958), 293–315, with a biographical note by G.A.Barnard. Reproduced in Press (1989), 185–217.

Berger, J. (1985). *Statistical Decision Theory and Bayesian Analysis*, Springer, Berlin.

Bernardo, J. and Smith, A. (1994). *Bayesian Theory*, Wiley, Chichester.

Box, G., Jenkins, G. and Reinsel, G. (1994). *Time Series Analysis, Forecasting and Control*, 3rd edn, Holden-Day, San Francisco.

Caines, P. (1988). *Linear Stochastic Systems*, Wiley, New York.

Campbell, J., Lo, A. and MacKinley, A. (1996). *The econometrics of financial markets*, Princeton University Press.

Cappé, O., Moulines, E. and Rydén, T. (2005). *Inference in Hidden Markov Models*, Springer, New York.

Carmona, R. A. (2004). *Statistical analysis of financial data in S-plus*, Springer-Verlag, New York.

Carter, C. and Kohn, R. (1994). On Gibbs sampling for state space models, *Biometrika* **81**: 541–553.

Chatfield, C. (2004). *The analysis of time series*, sixth edn, CRC-Chapman & Hall, London.

Cifarelli, D. and Muliere, P. (1989). *Statistica bayesiana*, Iuculano Editore, Pavia. (in Italian).

Cowell, R., Dawid, P., Lauritzen, S. and Spiegelhalter, D. (1999). *Probabilistic networks and expert systems*, Springer-Verlag, New York.

D'Agostino, R. and Stephens, M. (eds) (1986). *Goodness-of-fit techniques*, M. Dekker, New York.

Dalal, S. and Hall, W. (1983). Approximating priors by mixtures of conjugate priors, *J. Roy. Statist. Soc. Ser. B* **45**(2): 278–286.

de Finetti, B. (1970a). *Teoria della probabilità I*, Einaudi, Torino. English translation as *Theory of Probability I* in 1974, Wiley, Chichester.

de Finetti, B. (1970b). *Teoria della probabilità II*, Einaudi, Torino. English translation as *Theory of Probability II* in 1975, Wiley, Chichester.

De Finetti, B. (1972). *Probability, Induction and Statistics*, Wiley, Chichester.

DeGroot, M. (1970). *Optimal statistical decisions*, McGraw Hill, New York.

Del Moral, P. (2004). *Feyman-Kac formulae. Genealogical and interacting particle systems with applications*, Springer.

Diaconis, P. and Ylvisaker, D. (1985). Quantifying prior opinion, *in* J. Bernardo, M. deGroot, D. Lindley and A. Smith (eds), *Bayesian statistics 2*, Elsevier Science Publishers B.V. (North Holland), pp. 133–156.

Diebold, F. and Li, C. (2006). Forecasting the term structure of government bond yields, *Journal of Econometrics* **130**: 337–364.

Diebold, F., Rudebuschb, G. and Aruoba, S. (2006). The macroeconomy and the yield curve: a dynamic latent factor approach, *Journal of Econometrics* **131**: 309–338.

Doucet, A., De Freitas, N. and Gordon, N. (eds) (2001). *Sequential Monte Carlo methods in practice*, Springer.

Durbin, J. and Koopman, S. (2001). *Time Series Analysis by State Space Methods*, Oxford University Press, Oxford.

Engle, R. and Granger, C. (1987). Co-integration and error correction: representation, estimation, and testing, *Econometrica* **55**: 251–276.

Früwirth-Schnatter, S. (1994). Data augmentation and dynamic linear models, *Journal of Time Series Analysis* **15**: 183–202.

Gilks, W. and Berzuini, C. (2001). Following a moving target – Monte Carlo inference for dynamic Bayesian models, *J. Royal Statist. Soc. B* **63**: 127–146.

Gourieroux, C. and Monfort, A. (1997). *Time series and dynamic models*, Cambridge University Press, Cambridge.

Granger, C. (1981). Some properties of time series data and their use in econometric model specification, *Journal of Econometrics* **16**: 150–161.

Hannan, E. and Deistler, M. (1988). *The Statistical Theory of Linear Systems*, Wiley, New York.

Harrison, P. and Stevens, C. (1976). Bayesian forecasting (with discussion), *J. Royal Statist. Soc. B* **38**: 205–247.

Harvey, A. (1989). *Forecasting, structural time series models and the Kalman filter*, Cambridge University Press, Cambridge.

Jacquier, E., Polson, N. and Rossi, P. (1994). Bayesian analysis of stochastic volatility models (with discussion), *Journal of Business and Economic Statistics* **12**: 371–417.

Johannes, M. and Polson, N. (2006). .., Vol. .., .., chapter MCMC methods for continuous-time financial econometrics, p. ..

Kalman, R. (1960). A new approach to linear filtering and prediction problems, *Trans. of the AMSE - Journal of Basic Engineering (Series D)* **82**: 35–45.

Kalman, R. and Bucy, R. (1963). New results in linear filtering and prediction theory, *Trans. of the AMSE - Journal of Basic Engineering (Series D)* **83**.

Kolmogorov, A. (1941). Interpolation and extrapolation of stationary random sequences, *Bull. Moscow University, Ser. Math.* **5**.

Künsch, H. (2001). State space and hidden Markov models, *in* O. Barndorff-Nielsen, D. Cox and C. Klüppelberg (eds), *Complex stochastic systems*, Chapman & Hall/CRC, Boca Raton, pp. 109–173.

Laplace, P. (1814). *Essai Philosophique sur les Probabilitiès*, Courcier, Paris. The 5th edition (1825) was the last revised by Laplace. English translation in 1952 as *Philosophical Essay on Probabilities*, Dover, New York.

Lauritzen, S. (1981). Time series analysis in 1880: A discussion of contributions made by T.N. Thiele, *International Statist. Review* **49**: 319–331.

Lauritzen, S. (1996). *Graphical models*, Oxford University Press, Oxford.

Lindley, D. and Smith, A. (1972). Bayes estimates for the linear model, *Journal of the Royal Statistical Society. Series B (Methodological)* **34**: 1–41.

Liu, J. (2001). *Monte Carlo strategies in scientific computing*, Springer.

Liu, J. and West, M. (2001). Combined parameter and state estimation in simulation-based filtering, *in* A. Doucet, N. De Freitas and N. Gordon (eds), *Sequential Monte Carlo methods in practice*, Springer.

Ljung, G. and Box, G. (1978). On a measure of lack of fit in time series models, *Biometrika* **65**: 297–303.

Lütkepohl, H. (2005). *New introduction to multiple time series analysis*, Springer-Verlag, Berlin.

Maybeck, P. (1979). *Stochastic models, estimation and control*, Vol. 1 and 2, Academic Press, New York.

Migon, H., Gamerman, D., Lopez, H. and Ferreira, M. (2005). Bayesian dynamic models, *in* D. Day and C. Rao (eds), *Handbook of Statistics*, Vol. 25, Elsevier B.V., chapter 19, pp. 553–588.

Morf, M. and Kailath, T. (1975). Square-root algorithms for least-squares estimation, *IEEE Trans. Automatic Control* .

Muliere, P. (1984). Modelli lineari dinamici (in italian), *Studi statistici 8*, Istituto di Metodi Quantitativi, Bocconi University.

O'Hagan, A. (1994). *Bayesian Inference*, Kendall's Advanced Theory of Statistics, 2B, Edward Arnold, London.

Oshman, Y. and Bar-Itzhack, I. (1986). Square root filtering via covariance and information eigenfactors, *Automatica* **22**(5): 599–604.

Petris, G. and Tardella, L. (2003). A geometric approach to transdimensional Markov chain Monte Carlo, *Canadian Journal of Statistics* **31**: 469–482.

Pitt, M. and Shephard, N. (1999). Filtering via simulation: auxiliary particle filters, *Journal of the American Statistical Association* **94**: 590–599.

Plackett, R. (1950). Some theorems in least squares, *Biometrika* **37**: 149–157.

Poirier, D. (1995). *Intermediate Statistics and Econometrics: a Comparative Approach*, Cambridge: the MIT Press.

Pole, A., West, M. and Harrison, J. (n.d.). *Applied Bayesian forecasting and time series analysis*, Chapman & Hall, New York.

Prakasa Rao, B. (1999). *Statistical Inference for Diffusion Type Processes*, Oxford University Press, N.Y.

Rabiner, L. and Juang, B. (1993). *Fundamentals of Speech Recognition*, Prentice-Hall, Englewood Cliffs, New Jersey.

Reinsel, G. (1997). *Elements of Multivariate Time Series Analysis*, 2nd edn, Springer-Verlag, New York.

Robert, C. (2001). *The Bayesian choice*, 2nd edn, Springer-Verlag, New York.

Robert, C. and Casella, G. (2004). *Monte Carlo statistical methods*, 2nd edn, Springer, New York.

Rydén, T. and Titterington, D. (1998). Computational bayesian analysis of hidden markov models, *J. Comput. Graph. Statist.* **7**: 194–211.

Shephard, N. (1994). Partial non-Gaussian state space models, *Biometrika* **81**: 115–131.

Shephard, N. (1996). Statistical aspects of ARCH and stochastic volatility, *in* D. Cox, D. Hinkley and O. Barndorff-Nielsen (eds), *Time series models with econometric, finance and other applications*, Chapman and Hall, London, pp. 1–67.

Shumway, R. and Stoffer, D. (2000). *Time Series analysis and its Applications*, Springer-Verlag, New York.

Sokal, A. (1989). *Monte Carlo methods in statistical mechanics: foundations and new algorithms*, Cours de Troisiéme Cycle de la Physique en Suisse Romande, Lausanne.

Stein, C. (1956). Inadmissibility of the usual estimator for the mean of a multivariate normal distribution, *in* J. Neyman (ed.), *Proceedings of the Third Berkeley Symposium on Mathematical Statistics and Probability, Volume 1*, University of California Press, pp. 197–206.

Storvik, G. (2002). Particle filters for State-Space models with the presence of unknown static parameters, *IEEE Transactions on Signal Processing* **50**(2): 281–289.

Uhlig, H. (1994). On singular wishart and singular multivariate beta distributions, *Annals of Statistics* **22**: 395–405.

Venables, W. and Ripley, B. (2002). *Modern Applied Statistics with S*, 4th edn, Springer-Verlag, New York.

Wang, L., Liber, G. and Manneback, P. (1992). Kalman filter algorithm based on singular value decomposition, *Proc. of the 31st conf. on decision and control*, pp. 1224–1229.

West, M. and Harrison, J. (1997). *Bayesian forecasting and dynamic models*, second edition. first edition: 1989 edn, Springer, New York.

Wiener, N. (1949). *The Extrapolation, Intepolation and Smoothing of Stationary Time Series*, John Wiley & Sons, New York.

Wold, H. (1938). *A study in the Analysis of Stationary Time series*, Almquist and Wiksell, Uppsala.

Y., C., J.I., M. and Park, J. (2005). Extracting a common stochastic trend: Theories with some applications, *Technical report*, ???

Zellner, A. (1971). *An Introduction to Bayesian Inference in Econometrics*, John Wiley & Sons. New York.

Zhang, Y. and Li, R. (1996). Fixed-interval smoothing algorithm based on singular value decomposition, *Proceedings of the 1996 IEEE international conference on control applications*, pp. 916–921.