Dynamic Models with R
Book Summary

State space models serve as a good entry point in to modern Bayesian inference for time series. The authors in the preface mention that

> State space models provide a very flexible yet fairly simple tool for analyzing dynamic phenomena and evolving systems, and have significantly contributed to extend the classical domains of application of statistical time series analysis to non-stationary, irregular processes, to systems evolving in continuous-time, to multivariate, continuous and discrete data.

Indeed any time series model that you would have built till date can be transformed in to a state space model. If you an `R` programmer, you would have noticed that even the most basic time series model is converted in to a state space model. For example the help on `arima` gives the following description :

> The exact likelihood is computed via a state-space representation of the ARIMA process, and the innovations and their variance found by a Kalman filter. The initialization of the differenced ARMA process uses stationarity and is based on Gardner et al. (1980). For a differenced process the non-stationary components are given a diffuse prior (controlled by kappa). Observations which are still controlled by the diffuse prior (determined by having a Kalman gain of at least 1e4) are excluded from the likelihood calculations. (This gives comparable results to arima0 in the absence of missing values, when the observations excluded are precisely those dropped by the differencing.)

So, not having a decent understanding of state space models might make it difficult to understand what exactly is happening under the hood. There are very good books that are solely dedicated to Bayesian forecasting, Kalman Filter and State space models. Where does this book fit in ? Firstly, one of the authors of the book has contributed to a fantastic `R` package called `dlm` and they try to show code and visuals right through the book. This is something that has been possible because of the "Reproducible Research" packages available in `R`. This means that as you learn about various principles of DLM, you can actually look at how they can be put in practice. However I feel this book cannot be your first book to understand Bayesian time series analysis as the entire Bayesian inference is covered in the first chapter. This is hardly enough if you are just beginning. If you are a newbie, I think spending some time understand Bayes fundas, working out the math from a book like "Bayesian Forecasting and Dynamic Models" would be a good beginning. Once you slog through the math from Mike West and Jeff Harrison's book, this book will be very appealing for couple of reasons

- There is a great amount of notational similarity between "Bayesian Forecasting and Dynamic Models' and "Dynamic Models with R".
- Both cover DLM from a Bayesian perspective

Also some understanding of Gibbs sampling, Metropolis algorithm and general fundas about Montecarlo are needed to understand the final two chapters of the book this book. The reason for this I guess is that interest in State space models has gone up in the last two decades mainly due to the availability of efficient computational tools. Basic steps in computations of DLM such as *forward filtering backward sampling* algorithms and sequential MCMC have been possible only in the recent decade or so. Hence a strong understanding of atleast the basic MCMC algos might be needed to go over this book. Ok, I think the above rambling needs to have a disclaimer. If you are mathematically savvy and can understand stuff at the first go(unlike me, who takes innumerable readings to understand), you can probably read the book with out spending much time on the other literature on Bayesian inference. Let me attempt to summarize each of the five chapter briefly.

Introduction : Basic Notions about Bayesian Inference

The chapter starts by giving a historical time line for the developments in Dynamic linear models. They were mainly developed in engineering in 1960s and were adopted in the statistical time series literature in 1970-1980s. Bayesian framework has several advantages both methodical and computational. Dynamic linear models are based on the idea of describing the output of a dynamic system, as a function of nonobservable state processes affected by random errors. This way of modeling the temporal dependence in the data, by conditioning on the latent variables, is simple and extremely powerful, and it is quite natural in the Bayesian approach. Another advantage of DLM is that the computations are done recursively, the conditional distributions of interest can be updated, incorporating the new data, without requiring the storage of all the past history. This is advantageous in the case of in situations where one needs to put in a place a model that does real time inferencing. The few paragraphs at the beginning make it abundantly clear that the book approaches state space via Bayes principles. There are books that deal with DLM from a frequentist point of view such as "Time Series Analysis by State space methods" by Durbin and Koopman. Personally I found the Bayesian approach quite appealing. The first chapter of the book gives a crash course on Bayesian inference. Starting from the very basic, i.e. Bayes rule, it touches upon all the aspects of a typical Bayesian analysis and thus answering the following questions :

- How to model the Markovian nature of the data ?
- How to specify a conjugate prior given the likelihood function ?
- What is the importance of loss function and how does it affect the analysis ?
- What are the different types of priors that one can choose ?
- How does one employ Bayesian inference in a linear regression model ?
- What are the most common MCMC method used for Bayesian analysis ?

Dynamic Linear Models

The second chapter introduces the general setting of state space models and dynamic linear models. It starts off by citing the important literature in the field of State space models. One of the biggest advantages of State space models is that they can be used to model non stationary data. In the traditional time series literature, one usually removes nonstationarity from the time series and then fits a relevant model. In one sense, this transformation of the original series removes out all the interpretability of the model. State space models makes the modeler consciously identify the components of the stochastic series. The intention is to capture the latent variable that is driving the system. So, you start with an observation equation, a system equation and the distribution of error terms. The rest of calculations flow from Bayesian inference. I have learnt from West and Harrison's book, what I think is one of the best notation to represent any DLM.
Let's say we are dealing with the following DLM :

$$\begin{aligned}
\text{Observation equation :} \quad & Y_t = F_t\mu_t + \nu_t, \quad \nu_t \sim N[0, V_t] \\
\text{System equation :} \quad & \mu_t = \lambda\mu_{t-1} + \omega_t, \quad \nu_t \sim N[0, W_t] \\
\text{Initial Information :} \quad & \mu_0 \sim N[m_0, C_0]
\end{aligned}$$

The above DLM can be concisely represented by a quadruple $\{F_t, \lambda, V_t, W_t\}$. In fact once you have this representation in mind, you can map quadruples to specific classes of DLM.

The chapter does a wonderful job of explaining terms such as prediction and filtering via a simple example and a few visuals. In the process it highlights the importance of signal to noise ratio in making the correct inferences from the model. Most of the principles of DLM can be understood by looking at the system as a Directed Acyclic Graph. Once you start viewing it as a DAG , the distributions for one step ahead prediction, filtering, smoothing and forecasting become very easy to comprehend. The chapter does not overwhelm the reader with all the math all at once. The authors pick up a dataset, display the respective raw time series and overlay it with different types of series as they go along. Filtered estimate is displayed and then the math relating to it is discussed. Same is the case with smoothing and forecast estimates. Thus a reader kind of knows what the math will accomplish. In this code-explain-code-explain cycle, the authors gently introduce the various aspects of `dlm` package. There is some stuff online about this package but I don't think it is as comprehensively dealt at any other place like this book. The authors explain the following aspects about the package in great detail:

- How to define a DLM ?
- How to define a time varying coefficient DLM ?
- How to get one step ahead prediction from the model ?

- How to use the `dlmFilter` function ?
- How to use the `dlmSmooth` function ?
- How to use the `dlmForecast` function ?
- How to extract the one step ahead prediction variance and filtered state variance ?

There is enough `R` code for a reader to get a decent idea about using the package. One particularly interesting aspect mentioned in this chapter is about the way State space models handle missing observations. Real life data are awash with missing observations. One is pleasantly surprised how missing data for observations can easily be incorporated in to the State space framework. The advantage of using this package is that the authors have already coded the functionality of taking care of the missing observations in filtering and smoothing functions. The chapter also deals with some basic diagnostics that one can perform to check for any model misspecification. If you read Koopman's book, there is a heavy emphasis on a different version of DLM, the innovation DLM. The chapter gives the rationale on why one would go with this form of DLM. Towards the end of the chapter, Controllability and observability of time-invariant DLMs are dealt. I think one can skip this section in the first read. It was little overwhelming for me in the first read and hence got back to this section after understanding the importance of controllability and observability from West and Harrison's book. I constantly keep referring to West and Harrison's book mainly because the notation is extremely clean and most of the derivations are done at a leisurely pace that any reader can follow. Personally I think Observability and Controllability treatment in this book needs to be either improved or completely removed from the book.

Model Specification

This chapter starts with a section on classical time series models. It starts with simple exponential smoothing and Holt Winters trend corrected smoothing. The EWMA methods are at best exploratory in nature and a big limitation with these methods is that they are not based on probabilistic or statistical model. Hence it is impossible to assess the uncertainty of the forecasts. The basic setup for HoltWinters and the way the model parameters are dealt in any standard text. For more easy reading, one can go through the book, "Data Smart" by John Foreman where he shows the workings of the model via Excel. ARIMA models are mentioned too and relevant `R` packages such as `des1` are mentioned for the interested tinkerer. EWMA or ARIMA models can be cast in to an DLM framework. The fundamental philosophy behind ARIMA and DLM is different. ARIMA models provide a black-box approach to data analysis, offering the possibility of forecasting future observations, but with a limited interpretability of the fitted model. On the other hand, the DLM framework encourages the analyst to think in terms of easily interpretable, albeit unobservable, processes - such as trend and seasonal components - that drive the time series.

Once the classic time series inference mechanisms are covered, the chapter introduces univariate DLMs that can serve as an umbrella for the classic time series models. Any ARIMA, EWMA can be cast in to a DLM. Local level model and Linear Growth model are explored via sample datasets and all relevant aspects of DLM model analysis except parameter inference is shown via code and visuals. These include

- Defining a Local level model
- Obtaining Filtered estimates
- Obtaining Smoothing estimates
- Obtaining Filtered state variances
- Obtaining Smoothed state variances
- Diagnostic checking of DLM
- Connection with the HoltWinters estimate for large t
- Comparing two models from a frequentist perspective

Seasonal factor models are also introduced where relevant matrices are built to use along with the `dlm` package functions. The good think about the package is that you just have to specify the appropriate seasonality and the representation is take care by the package implementation. One can also use Fourier frequencies as basis to fit the seasonal component of the DLM. An extensive discussion about the usage of fourier frequencies in modeling seasonal component is provided in this chapter. 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 auto-correlation 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 performed efficiently by applying the general recursion used to compute the likelihood of a DLM. There are many DLM representations for an ARMA model and the chapter presents one of the most widely used one. Clear explanations are given to equip the reader to cast and ARMA model to DLM representation.

The chapter contains an interesting example where an observation series is assumed to have two components, local linear trend model and ARMA model. These two components are specified and MLE is used to estimate the parameters. However in the first version of the example, the parameters that are fit do not have any restrictions on ARMA stationarity. Hence the authors show a second version of the exampler where they impose ARMA stationarity and estimate the parameters. The section on univariate models concludes with a regression model. Using CAPM as the context, Regression DLM is applied to a stock to analyze the time varying β_t, the sensitivity of the asset excess returns to the changes in the market. The last section of the chapter deals with multivariate time series and ways in which DLM framework can be leveraged for better analysis and interpretation.

Parameter Estimation

In all the previous chapters of the book, the parameters of DLM were conveniently provided and hence filtering, smoothing and forecasting were carried out. This chapter starts with a practical premise that parameters of DLM are often unknown. There are two ways to go about it. First is that you can look at the observational data and then use MLE to estimate the parameters. Use those estimates to do filtering, smoothing and forecasting. The second way is from a Bayesian stand point where the parameters are unknown random variables and hence the posterior distribution of interest is the joint conditional distribution of the state vectors and the unknown parameters. In order to proceed with the second method, one must understand MCMC and sequential MCMC.

For estimating the parameters via MLE, the likelihood function for the observation is written in terms of parameters and `optim` function is used to minimize log likelihood. The `dlm` package has a function `dlmMLE` that takes in the observation series, a model object to give out MLE estimates. One needs to be careful about the kind of parameterization that needs to be done for MLE convergence. Most often the parameters estimated are logarithms of observational and state variance. Hence to compute the standard errors from the output, one needs to use the delta method to get the estimates on the correct scale. This is a nifty point. I think it is very instructive to code up your own MLE function and see how your implementation and performance of the function compares with the function in the package.

The common practice of plugging the MLE in to the filtering and smoothing recursions suffers from the difficulties in taking properly into account the uncertainty about MLE. The Bayesian approach offers a more consistent formulation of the problem. The unknown parameters are regarded as a random vector and joint distributions of the state vector along with the parameters is computed at each point in time. However one cannot use the plain vanilla MCMC as every new observation demands us running a new MCMC chain. Hence one needs to resort to sequential MCMC and particle filter algorithms to make the whole estimation procedure computationally feasible.

Just to get the intuition right, conjugate priors are used for the unknown parameters and since closed form solutions are available for filtered and smoothed estimates, all sounds good. However in practice, one needs to resort to Simulation based Bayesian inference. The chapter highlights *Forward Filtering Backward Sampling* algorithm that serves as a building block for MCMC methods. FFBS essentially involves running a Kalman filter and then sampling from the filtered estimates, thus providing a random sample of state variables, given a set of observations.

For a completely specified DLM, one not containing unknown parameters, draws from the posterior distribution of the states can be obtained using FFBS algorithm. In the more realistic situation of a DLM containing an unknown parameter vector with a prior distribution in the observation, system or variance matrices, one needs to MCMC sampling. Almost all the Markov chain sampler for posterior analysis of a DLM falls in to three categories

- Gibbs samplers which include states as latent variables
- Margin samplers - Generates draws from the posterior of the parameter
- Hybrid samplers - Generates draws from the joint posterior of the states and the parameter

The General strategy for simulating a sample from the joint posterior of state vector and parameters is : Simulate a draw of state vectors given the observations using FFBS. Subsequently, simulate a value of parameter given the state data and observational data. The latter step is obviously problem specific. If the parameter vector is multivariate, one can use Gibbs sampling to run through the various components to generate a sample from the multivariate distribution. One can also use a hybrid sampler, i.e separate out the parameters in to two sets and then use Metropolis and FFBS. The pseudo code for both the above cases are given in the chapter. There are also functions in the package to carry out some of the steps that are generic for any problem. Few examples are also given so that reader gets an idea about the way the two steps can be put in place, i.e. Run FFBS first and then simulate the parameter values given the state vector simulated sample and observations. The chapter ends with several examples that illuminate the way MCMC is done for DLMs.

Sequential MCMC

The last chapter is only 22 pages long but is one of the challenging sections in the book. Several new types of modern MCMC algos are discussed in the such as particle filter, auxiliary particle filter and sequential Montecarlo with unknown parameters. The authors give a tip of iceberg view of this whole new area of online inference. However the way the material is presented in the section, it will motivate any curious reader to look up on other references mentioned in the chapter.

Takeaway

`dlm` package is one of the best resources out there in the open source community that can be used for DLM inference. The fact that one of the authors is also the contributors to the package has made this book apt for practitioners. However the book is best understood after having a working knowledge of Bayesian inference. By understanding and thinking in State space framework, a modeler gets many more options to model univariate or multivariate time series data. This book does an amazing job in explaining the nuts and bolts of State space models in Bayesian setting.