Filtering in state space models

The purpose of this computer exercise is to study methods for modelling non-linear state-space models, the Kalman filter and the Bootstrap particle filter. You should solve all the regular problems to pass the computer exercise, and you need to solve the extra problems if you want to get 1 bonus credits.

It is preferred to use Matlab, but you are allowed to use any programming language or package of your choice. Please note that you are required to document your code extra carefully if you choose not to use Matlab,

1 Preparations for the exercise

Read chapters 8.1, 12 and 14 in [2], hand-outs from [3] and this instruction. You are also expected to prepare the computer exercise by writing the Matlab functions needed for the exercise.

Before the computer exercise some of the questions below will be posed. All of the posed question must be answered correctly in order to pass the computer exercise.

2 Catalogue of questions

1. Assume affine drift and diffusion terms. Derive the bond price to $p(t,T) = e^{\mathbf{A}(t,T)+\mathbf{B}(t,T)r(t)}$, where $\mathbf{A}(t,T)$ and $\mathbf{B}(t,T)$ are given by Equations (11.98-11.99) in [2]. Find the system of equations that gives $\mathbf{A}(t,T)$ and $\mathbf{B}(t,T)$.

2. Calculate $\mathbf{A}(t,T)$ and $\mathbf{B}(t,T)$ for the Merton model, $dr(t) = \theta dt + \sigma dW(t)$.

3. Calculate the transition density for the Cox-Ingersoll-Ross model by using an Euler approximation using the theory in chapter 12 in [2]. Can you derive the Milstein approximation?

4. Write down the Bootstrap particle filter algorithm for the stochastic volatility model in Matlab or pseudo code.

3 Filtering

This text does not attempt to be a complete exposition of the filtering problem. For a more thorough text, please refer to [2] and [3]. In this computer exercise you will study two different filtering methods. First you will use the Kalman filter which is the closed form solution to the the linear, Gaussian filtering problem. Next you will use the more general Monte Carlo filter. Assume we are given a system of differential equations on the form:

$$\begin{aligned} \mathrm{d}X_t &= \mu(t, X_t)\mathrm{d}t + \sigma(t, X_t)\mathrm{d}W_t \quad (1) \\ Y_t &= f(t, X_t) + e_t \quad (2) \end{aligned}$$

where $\mu(t, x)$, $\sigma(t, x)$ and $f(t, x)$ are some deterministic functions and e some measurement noise. We usually refer to Equation (1) as the *state equation* and to Equation (2) as the *measurement equation*. Sometimes the filtering problem is stated in discrete form using transition kernels where p denotes a general probability density:

$$\begin{aligned} X_t &\sim p(X_t|X_{t-1}) \quad (3) \\ Y_t &\sim p(Y_t|X_t) \quad (4) \end{aligned}$$

We assume that the process X_t is Markovian and that Y_t are conditionally independent given X_t for all t. The filtering problem concerns finding the conditional distribution of the hidden state given the observations, i.e. $p(x_{0:t}|y_{1:t})$. Let's consider the simultaneous density $p(x_{0:t}, y_{1:t})$. Applying Bayes' theorem twice, with different conditioning, yields :

$$p(x_{0:t}|y_{1:t})p(y_{1:t}) = p(y_{1:t}|x_{0:t})p(x_{0:t}). \quad (5)$$

Re-arrange the expression:

$$p(x_{0:t}|y_{1:t}) = \frac{p(y_{1:t}|x_{0:t})p(x_{0:t})}{p(y_{1:t})}. \quad (6)$$

Using the RHS in Equation (5) once more yields: $p(y_{1:t}, x_{0:t}) = p(y_{1:t}|x_{0:t})p(x_{0:t})$. Integrating out $x_{0:t}$ gives $\int p(y_{1:t}, x_{0:t})dx_{0:t} = \int p(y_{1:t}|x_{0:t})p(x_{0:t})dx_{0:t} = p(y_{1:t})$, i.e the marginal density for $y_{1:t}$. Plugging this in to Equation (6) yields:

$$p(x_{0:t}|y_{1:t}) = \frac{p(y_{1:t}|x_{0:t})p(x_{0:t})}{\int p(y_{1:t}|x_{0:t})p(x_{0:t})dx_{0:t}}. \quad (7)$$

This is the *smoothing* or *joint filter density*, the distribution of the hidden state given observations of the observable variable. For practical purposes it is desirable to have this expression in a recursive form. Note that:

$$p(x_{0:t}|y_{1:t-1}, y_t) = \frac{p(x_{0:t}, y_t|y_{1:t-1})}{p(y_t|y_{1:t-1})}. \quad (8)$$

Now we consider $p(x_{0:t}, y_t|y_{1:t-1}) = p(y_t|x_{0:t}, y_{1:t-1})p(x_{0:t}|y_{1:t-1})$. Now consider the first factor on the RHS. If we use the assumption that the observations are conditionally independent this can be written: $p(y_t|x_{0:t}, y_{1:t-1}) = p(y_t|x_{0:t}) = p(y_t|x_t)$. The second factor on the RHS can be written:

$$p(x_{0:t-1}, x_t|y_{1:t-1}) = p(x_t|x_{0:t-1}, y_{1:t-1})p(x_{0:t-1}|y_{1:t-1}) = p(x_t|x_{t-1})p(x_{0:t-1}|y_{1:t-1}).$$

where we use the Markov property in the final equality. Putting this together we get:

$$p(x_{0:t}|y_{1:t}) = \frac{p(x_{0:t}, y_t|y_{1:t-1})}{p(y_t|y_{1:t-1})} = \frac{p(y_t|x_t)p(x_t|x_{t-1})p(x_{0:t-1}|y_{1:t-1})}{p(y_t|y_{1:t-1})}. \tag{9}$$

This is the recursive formulation of Equation (7). We now consider the marginal densities of the recursive filtering equations. First, we want to find $p(x_t|y_{1:t-1})$. This is the predictive density for the current state, given all previous observations. Consider the conditional density $p(x_t|x_{t-1}) = p(x_t|x_{t-1}, y_{1:t-1})$ (since $x_t|x_{t-1}$ is indep. of y_{t-1}) which is equal to $p(x_t, x_{t-1}|y_{1:t-1})/p(x_{t-1}|y_{1:t-1})$. If we rearrange the terms and integrate with respect to x_{t-1} we get:

$$\int p(x_t|x_{t-1})p(x_{t-1}|y_{1:t-1})dx_{t-1} = \int p(x_t, x_{t-1}|y_{1:t-1})dx_{t-1} = p(x_t|y_{1:t-1}). \tag{10}$$

Also we need the conditional density for the current state given the all previous and the current observation, i.e. $p(x_t|y_{1:t})$. Using Bayes' theorem we get:

$$p(x_t|y_t, y_{1:t-1}) = \frac{p(y_t, x_t|y_{1:t-1})}{p(y_t|y_{1:t-1})} = \frac{p(y_t|x_t)p(x_t|y_{1:t-1})}{p(y_t|y_{1:t-1})}. \tag{11}$$

Finally, if we recognize the identity $p(y_t|y_{1:t-1}) = \int p(y_t|x_t)p(x_t|y_{1:t-1})dx_t$ we get:

$$p(x_t|y_{1:t}) = \frac{p(y_t|x_t)p(x_t|y_{1:t-1})}{\int p(y_t|x_t)p(x_t|y_{1:t-1})dx_t}. \tag{12}$$

The reason for writing the filtering equations on this form is that we know all of the probabilities from our state space model. Here we refer to Equation (10) as the *predictive density* and to Equation (12) as the *filtering* or *updating density*. Solving these equations is not as easy as it might seem. It turns out that closed form solutions are only possible for a few special cases resulting in the Kalman Filter and the finite state filter. If we consider the system given above as an example in Eqs. (1) and (2) and want to calculate the predictive and filtering densities, the first thing to do is to write it on the form in Eqs. (3) and (4).

In the case of an SDE, we usually discretize the dynamics or use the exact transition density if it is available (e.g. the CIR has non-central-χ^2 transition density). Say we use an Euler discretization in our example and assume the measurement noise to be Gaussian with mean 0 and variance ξ. We then get the following system:

$$p(x_{t+1}|x_t) \sim N(x_t + \mu(t, x_t)\Delta t, \sigma(t, X_t)\sigma(t, X_t)^T\Delta t) \tag{13}$$

$$p(y_t|x_t) \sim N(f(t, x_t), \xi) \tag{14}$$

Looking at the filtering equations above, we see that if the measurement function $f(t, x)$ is linear, the equations can be solved. We then get the Kalman filter. In case of non-linear functions, we will have to resort to numerical methods or simulation to solve the equations.

3.1 Kalman filter

In the case that $\mu(t, x)$, $\sigma(t, x)$ and $f(t, x)$ in Equation (1) and Equation (2) are linear and e are Gaussian i.i.d. we can use the Kalman filter to solve the filtering problem. A general linear system can be written:

$$Y_t = CX_t + D + v_t \tag{15}$$
$$X_{t+1} = AX_t + B + w_{t+1} \tag{16}$$

where w_t has covariance matrix Q and v_t has covariance matrix R. We will know derive the Kalman filter from these equations and from eqs. (10) and (12). From this we know that $p(x_t|y_{1:t}) \propto p(y_t|x_t)p(x_t|y_{1:t-1})$. Assume that the filtering density $p(x_{t-1}|y_{1:t-1})$ at time $t-1$ is known and that it is Gaussian with mean $\tilde{x}$ and covariance matrix $\tilde{P}$. We can then write the density $p(x_t|y_{1:t-1})$, using Equation (16) as a Gaussian with mean $A\tilde{x}$ and covariance $A\tilde{P}A^T + Q =: \Sigma$, since linear transformations of Gaussians and sums of Gaussians are again Gaussian. Next consider $p(y_t|x_t)$. This will also be a Gaussian, with mean Cx_t and covariance matrix R. Plugging in the densities gives:

$$p(x_t|y_{1:t}) \propto p(y_t|x_t)p(x_t|y_{1:t-1}) \tag{17}$$
$$= \phi(y_t, Cx_t, R) \cdot \phi(x_t, A\tilde{x}, \Sigma) \tag{18}$$

where $\phi(\cdot, \mu, \sigma^2)$ is the Gaussian density. Writing out the expressions and rearranging terms (exercise), the predictive density turns out to be Gaussian with mean $x_{t-1} + K(y_t - CA\tilde{x})$ and covariance $(I - KC)\tilde{P}$, which happens to be the Kalman filter equations ($K = \Sigma C^T(C\Sigma C^T + R)^{-1}$).

Sometimes one can find a transformation that allows the noise the be approximated fairly well by a Gaussian distribution. In this case the Kalman filter might still be used with good results. Also, sometimes the functions $\mu(t, x)$, $\sigma(t, x)$ and $f(t, x)$ can be approximated locally with linear functions. We can then construct a filter using this for variance updating. This is referred to as the Extended Kalman filter. See [2] for the Kalman filter and Extended Kalman filter algorithms.

3.2 Particle filter

If the measurement noise is non-Gaussian, or if $\mu(t, x)$, $\sigma(t, x)$ and $f(t, x)$ are non-linear one might have to solve the filtering equations by simulation. This is called the Monte Carlo filter, or Particle Filter. See the hand-outs [3] for a description of the basic Bootstrap filter.

4 Computer Exercises

4.1 Affine short rate model

Short interest models are difficult to handle statistically since we can not observe the short rate directly, only the price of bonds. Given a model for the short rate $dr(t) = \mu(t, r(t))dt + \sigma(t, r(t))dW(t)$ we can calculate bond prices according to the pricing formula (Equation (11.66) in [2]

$$p(t,T) = \mathbf{E}^{\mathbb{Q}}[e^{-\int_t^T r(u)du}|\mathcal{F}_t].$$

It is also possible to solve a PDE (according to Feynman-Kac representation theorem) to calculate bond prices. It has the following form:

$$\begin{aligned} F_t + \mu F_r + \frac{1}{2}\sigma^2 F_{rr} - rF &= 0, \\ F(T,r) &= 1. \end{aligned}$$

Note that this equation holds for models for which the short rate is specified for some martingale measure. For linear, or more general affine, models the PDE is reduced to a system of ODE's. Affine structure means that the drift and squared diffusion term are affine:

$$\begin{aligned} \mu(t, r(t)) &= \alpha(t)r + \beta(t), \\ \sigma(t, r(t)) &= \sqrt{\gamma(t)r + \delta(t)}. \end{aligned}$$

This framework can be extended in several ways, allowing for a rich class of models to be used. We will use a model in which we treat the short rate as a hidden state and use bond prices, or rather yields, for calibration The yield can be written as: $p(t,T) = e^{-Y(t,T)(T-t)}$. We can then cast the problem as a state-space model and use filtering methods to find the short rate:

$$\begin{aligned} \mathrm{d}r(t) &= (\mathbf{A}(t,T)(t)r(t) + \beta(t))\mathrm{d}t + \sqrt{\gamma(t)r(t) + \delta(t)}\mathrm{d}W(t), \\ Y(t_k) &= -\frac{1}{T-t_k}\Big(\log \mathbf{A}(t_k,T) - \mathbf{B}(t_k,T)r(t_k)\Big) + \varepsilon_k. \end{aligned}$$

here ε_k is measurement noise, typically multivariate Gaussian.

We will use the Cox-Ingersoll-Ross (CIR) model, specified under the $\mathbb{Q}$-measure as

$$\mathrm{d}r(t) = a(b - r(t))\mathrm{d}t + \sigma\sqrt{r(t)}\mathrm{d}W(t).$$

for the short rate. We can calculate the bond price under the CIR model to:

$$F^T(t,r) = \mathbf{A}(T-t)e^{-\mathbf{B}(T-t)r(t)},$$

where

$$\begin{aligned}\mathbf{B}(x) &= \frac{2(e^{\gamma x}-1)}{(\gamma+a)(e^{\gamma x}-1)+2\gamma},\\ \mathbf{A}(x) &= \left[\frac{2\gamma e^{(a+\gamma)(x/2)}}{(\gamma+a)(e^{\gamma x}-1)+2\gamma}\right]^{2ab/\sigma^2},\end{aligned}$$

and

$$\gamma = \sqrt{a^2+2\sigma^2}.$$

Note: In [1] the term A_0 is given in log form. Use the form given here for your calculations. For the CIR-model there are closed form expressions for the bond price and for European calls on zero-coupon bond.

We will use both simulated and real data to to estimate the interest rate. The real data is US T-bills from 1983 to 1998. All data is sampled monthly, i.e. ($\Delta t = 1/12$) and the bonds have maturities 3 months, 6 months, 1 year, 2 years, 3 years and 5 years.

This gives model on state space form that can be written as a state equation:

$$\mathrm{d}r(t) = a(b-r(t))\mathrm{d}t + \sigma\sqrt{r(t)}\mathrm{d}W(t),$$

and six measurement equations

$$\begin{aligned}Y(t_k)^{3m} &= -\frac{1}{1/4}\Big(\log\mathbf{A}(t_k,1/4)-\mathbf{B}(t_k,1/4)r(t_k)\Big)+\varepsilon_k^{3m},\\ Y(t_k)^{6m} &= -\frac{1}{1/2}\Big(\log\mathbf{A}(t_k,1/2)-\mathbf{B}(t_k,1/2)r(t_k)\Big)+\varepsilon_k^{6m},\\ Y(t_k)^{1y} &= -\frac{1}{1}\Big(\log\mathbf{A}(t_k,1)-\mathbf{B}(t_k,1)r(t_k)\Big)+\varepsilon_k^{1y},\\ Y(t_k)^{2y} &= -\frac{1}{2}\Big(\log\mathbf{A}(t_k,2)-\mathbf{B}(t_k,2)r(t_k)\Big)+\varepsilon_k^{2y},\\ Y(t_k)^{3y} &= -\frac{1}{3}\Big(\log\mathbf{A}(t_k,3)-\mathbf{B}(t_k,3)r(t_k)\Big)+\varepsilon_k^{3y},\\ Y(t_k)^{5y} &= -\frac{1}{5}\Big(\log\mathbf{A}(t_k,5)-\mathbf{B}(t_k,5)r(t_k)\Big)+\varepsilon_k^{5y}.\end{aligned}$$

Assignment: Use the Kalman filter to estimate the short rate in the simulated model.

```
>> load CIRstruct2
```

Yields data is given by `yhat`, parameters are given in the `cir` struct.

Assignment: (Extra) Use the Kalman filter to estimate the short rate *and* parameters for the real data from January 2009 to October 2022. This can be done by maximizing the (log-)likelihood for the observations, or by augmenting

the latent state vector with the parameters and apply a filter to this non-linear system.

```
>> load FredData
```

4.2 Stochastic Volatility

We will study the following stochastic volatility model:

$$\log \sigma_t^2 = \kappa_0 + \kappa_1 \log \sigma_{t-1}^2 + \nu_{t-1} \quad (19)$$

$$z_t = \sigma_t \epsilon_t, \quad (20)$$

where ν is i.i.d $N(0, \tau^2)$ and ϵ is standard normal. We will study this model using the Kalman filter and the Monte Carlo filter. Load the data:

```
>> load StochVol
```

The parameters used in the simulation was $\kappa_0 = -2$, $\kappa_1 = 0.7$ and $\tau = 1$.

As discussed before, sometimes we can do simplifying approximations of a model. In this case, we can find a simpler formulation such that we can approximate the model as a linear Gaussian model.

Assignment: Use the Kalman filter to estimate the volatility process.

Hint: We can transform the measurement equation to:

$$\log z_t^2 = \log \sigma_t^2 + \log \epsilon_t^2$$

where the noise $\log \epsilon_t^2$ has density:

$$\frac{1}{\sqrt{2\pi}} \exp\left(\frac{\epsilon}{2} - \frac{\exp(\epsilon)}{2}\right)$$

This density can be approximated with a Gaussian distribution with mean $\eta = -(\ln(2) + \gamma) \approx -1.2704$ and variance $\pi^2/2$.

Assignment: (Extra) Use the Particle filter to estimate the volatility process. See the hand-outs for a description of the particle filter. Can you modify the algorithm to filter the volatility process when ϵ_t has t-distribution?

5 Feedback

Comments and ideas relating to the computer exercise are always welcome. Send them to Magnus wiktorsson, `magnus.wiktorsson@matstat.lu.se`

6 MATLAB–routines

`fminsearch` Simplex based optimisation routine.

`fminunc` Numerical minimization of a multidimensional function. The routine is based on quasi-Newton (BFGS) and it can return the minising parameter value, the minimal function value and the Hessian. The cryptic name comes from *function minimization unconditional*.

`MLmax` Customized BHHH (Quasi-Newton type) optimization algorithm for maximum likelihood estimation. Maximises the likelihood function by using the score function's quadratic variations to estimate Fisher's Information matrix. Needs the loglikelihood returned as a columnvector. `>> [xout,logL,CovM]=MLmax(@lnL,x0,indata)`

References

[1] Björk, T. (2020), *Arbitrage Theory in Continuous Time*, 4th ed., Oxford University Press, New York.

[2] Lindström, E., Madsen, H., Nielsen, J. N., (2015) *Statistics for Finance*, Chapman and Hall/CRC, ISBN 9781482228991.

[3] Lopes, H. F. and Tsay, Ruey S. "Particle Filters and Bayesian Inference in Financial Econometrics." ; *Journal of Forecasting*, 2011, 30(1), pp. 168-209.