Parameter estimation in SDE:s

This computer exercise concerns some estimation of parameters in stochastic differential equations. It contains two main assignments with several sub-assignments. Many of these are labelled "Extra". You should solve all the regular problems (you need to solve at least two extra problems if you want to get 1 bonus credits from the computer exercise.)

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

1 Preparations for the exercise

Read chapters 8.1, 12 and 13 in [1] and this instruction. Then you should prepare the computer exercise by writing down 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

You should be able to answer the following questions before the computer exercise.

1. Discretize and find suitable moment restrictions for the CKLS-model, see Equation (2).
2. Write down the approximate likelihood function for the CIR model when discretized using the Euler scheme.
3. (If you do exercise 4.3) Calculate first, second and third order moments for the Cox-Ingersoll-Ross process using Dynkin's formula (see equation 4).

3 Computer Exercises

3.1 Parameter estimation in stochastic differential equations

You will meet two different processes in this part of the computer exercise: the Cox-Ingersoll-Ross (CIR) process and the CKLS-process. the CIR process is often used to model interest rates and is defined as:

$$dr_t = \kappa(\theta - r_t)dt + \sigma\sqrt{r_t}dW_t. \tag{1}$$

This can be generalized to the CKLS model given by:

$$dr_t = \kappa(\theta - r_t)dt + \sigma r_t^\gamma dW_t. \tag{2}$$

These models can also be used to model the volatility in stochastic variance models, since they always remain positive. Load the data by writing:

```
>> load cirdata.mat
>> load cklsdata.mat
```

By discretizing the stochastic differential equation one can use for instance GMM for parameter estimation. The discretization is usually done with an Euler scheme but of course higher order methods works even better. The advantage of using the discretized process is that we can estimate complex models without much problem.

Assignment: Estimate the parameters and their covariance matrices in the models above by using approximate Quasi Maximum Likelihood (by discretizing the model and using the likelihood generated by the approximate model).

Assignment: Estimate the parameters and their covariance matrices in the models above by using GMM and moment conditions of your choice. Compare the estimates with the Quasi Likelihood estimates derive above. **Note**: You *CANNOT* use MLmax to compute confidence intervals.

3.2 Exact likelihood for the CIR model

The CIR process has a known transition density . If we let $\Delta_k = t_k - t_{k-1}$, $c = 2\kappa/(\sigma^2(1 - \exp(-\kappa\Delta_k)))$, and Y = 2cX, then $Y_{t_k}|Y_{t_{k-1}}$ is distributed as a noncentral chi-squared with $2\kappa\theta/\sigma^2$ degrees of freedom and noncentrality parameter $Y_{t_{k-1}}\exp(\kappa\Delta_k)$. Using this we can write up the density as

$$p_{X_{t_k}|X_{t_{k-1}}}(x_{t_k}|x_{t_{k-1}}) = ce^{-u-v}\left(\frac{v}{u}\right)^{q/2} I_q(2\sqrt{uv}) \tag{3}$$

where $u = cx_{t_{k-1}} \exp(-\kappa\Delta_k)$, $v = cx_{t_k}$, $q = 2\kappa\theta/\sigma^2 - 1$ and $I_q(z)$ is the modified Bessel function of the first kind of order q (`besseli(q,z)` in Matlab).

Assignment: Estimate the parameters and their covariance matrix for the Cox-Ingersoll-Ross model by using the exact likelihood. Compare the result with the previous estimates where you used GMM, Simulated ML, [2] and/or Exact moments.

4 Improved approximate estimators (Extra)

All problems in this section are extra assignments. Solve at least two for bonus credits

4.1 Transformation of data for the CIR-model and the Shoji & Ozaki approximated likelihood (Extra)

Using the transformation $y_t = 2\sqrt{r_t}$ we obtain a state independent diffusion term. The Itō formula gives

$$\begin{aligned}
dy_t &= \left[\frac{1}{\sqrt{r_t}}\kappa(\theta - r_t) + \frac{-1}{2r_t^{3/2}}\frac{1}{2}\sigma^2 r_t\right] dt + \frac{1}{\sqrt{r_t}}\sqrt{r_t}\sigma dW_t \\
&= \left[\frac{2}{y_t}\kappa(\theta - \frac{y_t^2}{4}) + \frac{-4}{y_t^3}\sigma^2\frac{y_t^2}{8}\right] dt + \sigma dW_t \\
&= \left[\frac{2}{y_t}\kappa\theta - \kappa\frac{y_t}{2} + \frac{-2}{y_t}\frac{\sigma^2}{4}\right] dt + \sigma dW_t \\
&= \left[\frac{2}{y_t}\left(\kappa\theta - \frac{\sigma^2}{4}\right) - \kappa\frac{y_t}{2}\right] dt + \sigma dW_t \\
&= \mu(y_t)dt + \sigma dW_t
\end{aligned}$$

We can now use the [2] method to approximate the likelihood with a Gaussian likelihood.

$$\sum_{k=2}^{N} \log(p_{y_{t_k}|y_{t_{k-1}}}(y_{t_k})) \approx \sum_{k=2}^{N} \left(\frac{-(y_{t_k} - m_k)^2}{2v_k} - \frac{1}{2}\log(2\pi v_k) \right)$$

$$\begin{aligned}
m_k &= y_{t_{k-1}} + \frac{a_k}{b_k} K_k + \frac{\sigma^2 c_k}{2b_k^2}(K_k - b_k \Delta_k) \\
v_k &= \frac{\sigma^2}{2b_k}(\exp(2b_k\Delta_k) - 1) \\
\Delta_k &= t_k - t_{k-1} \\
K_k &= \exp(b_k \Delta_k) - 1 \\
a_k &= \mu(y_{t_{k-1}}) = \frac{2}{y_{t_{k-1}}}\left(\kappa\theta - \frac{\sigma^2}{4}\right) - \kappa \frac{y_{t_{k-1}}}{2} \\
b_k &= \mu'(y_{t_{k-1}}) = \frac{-2}{y_{t_{k-1}}^2}\left(\kappa\theta - \frac{\sigma^2}{4}\right) - \frac{\kappa}{2} \\
c_k &= \mu''(y_{t_{k-1}}) = \frac{4}{y_{t_{k-1}}^3}\left(\kappa\theta - \frac{\sigma^2}{4}\right)
\end{aligned}$$

Assignment: Estimate the parameters and their covariance matrices in the CIR model by using the [2] approximated likelihood.

4.2 Simulated Maximum Likelihood (Extra)

Sometimes it is possible to use Maximum Likelihood on a simulated likelihood in stochastic differential equations. Assume that we have observations y_n, $n = 1, \dots N$ from some model. Also assume that the sampling interval is Δ. We now want to find the likelihood

$$L(\theta) = \prod_{i=1}^{N} p(y_i | y_{i-1}, \theta).$$

If this is not available in closed form, one might approximate it using simulation. First we discretize the dynamics of the stochastic differential equation using a scheme that gives a simple transition probability, preferably Euler. Then, divide the interval Δ into M subintervals of length $\delta = \Delta/M$. The idea is the to simulate K trajectories on a grid of size δ up to subinterval $M-1$ starting in y_n. Do this for every time n, resulting in K samples at every n. Under the Euler discretization, the transition density from $M-1$ to M is Gaussian with mean $\mu_{n,k}$ and standard deviation $\sigma_{n,k}$, both given by the model in question. Then we approximate the likelihood by the following:

$$\begin{aligned} L(\theta) &= \prod_{n=1}^{N} p(y_{n+1}|y_n,\theta) \\ &\approx \prod_{n=1}^{N} \left(\frac{1}{K} \sum_{k=1}^{K} \phi(y_{n+1}, \mu_{n,k|\theta}, \sigma_{n,k|\theta}) \right) \end{aligned}$$

where $\phi(y,\mu,\sigma)$ is the density of the Normal distribution with mean μ and standard deviation σ. The log likelihood is given by

$$l(\theta) = \log(L(\theta)) = \sum_{n=1}^{N} \log(p(y_{n+1}|y_n),\theta) \approx \sum_{n=1}^{N} \left(\log \left(\frac{1}{K} \sum_{k=1}^{K} \phi(y_{n+1}, \mu_{n,k|\theta}, \sigma_{n,k|\theta}) \right) \right)$$

We then use numerical optimization techniques to maximize the loglikelihood. In this exercise we keep the number of intermediate steps fairly small, say $M = 2$ or 3. If M is too big, the variance of the sample will become large and make the estimation difficult. The remedy for this is to use an importance sampler.

Assignment: Estimate the parameters and their covariance matrices in the models above by using Simulated Maximum Likelihood.

Assignment: Do the same thing for the transformed data using the transformed CIR model.

Hint: It is usually a good idea to use the same sequence of random numbers each time the likelihood is evaluated. This is referred to as Common Random Numbers and is a way to avoid the Monte Carlo error in the minimization. Practically this is done by drawing a large number of random numbers, in this case $N \cdot (M-1) \cdot K$ and use these as input to the likelihood.

4.3 Exact Moments (GMM/EF) (Extra)

Sometimes the bias that discretization brings can be avoided by calculating the moments exactly. The moments can be calculated using the so called Dynkin's formula, which is nothing more than an application of the Itō formula. Dynkin's formula is given by:

$$E\left[f(X_t)|\, X_{t-1}\right] = f(x_0) + E\left[\int_0^t \mathcal{A}f(X_s)ds|X_{t-1}\right] \tag{4}$$

where the operator $\mathcal{A}$ (standard Itō's formula) is defined as,

$$\mathcal{A}f = \sum_{i=1}^{n} \mu_i \frac{\partial f}{\partial x_i} + \frac{1}{2} \sum_{i,j=1}^{n} \left(\sigma\sigma^T\right)_{i,j} \frac{\partial^2 f}{\partial x_i \partial x_j}.$$

and where the process $\{X_t\}$ is the solution to $dX(t) = \mu(t, X(t))dt + \sigma(t, X(t))dW(t)$. For our purposes $\{X_t\}$ is a one-dimensional process which makes Dynkin's formula a lot easier ($n = 1$).

Hint: To show how Dynkins formula can be used we will study an example with $f(x) = x^2$. Suppose that we want to calculate the moments of the CIR process.

$$\begin{aligned}
\mathcal{A}x^2 &= \kappa(\theta - x)\frac{\partial x^2}{\partial x} + \frac{1}{2}\left((\sigma^2 x)\frac{\partial^2 x^2}{\partial x^2}\right) \\
&= \kappa(\theta - x) \cdot 2x + \frac{1}{2}\sigma^2 x \cdot 2 \\
&= -2\kappa x^2 + (2\kappa\theta + \sigma^2)x
\end{aligned}$$

By plugging this into the formula we get:

$$E^X\left[X_t^2\right] = x_0^2 + E^X\left[\int_0^t (-2\kappa X_s^2 + (2\kappa\theta + \sigma^2)X_s)ds\right]$$

Finally we assume that the expectation and the time integral can be interchanged, followed by taking the derivatives with respect to t on both sides.

$$\frac{\partial E^X\left[X_t^2\right]}{\partial t} = -2\kappa E^X\left[X_t^2\right] + (2\kappa\theta + \sigma^2)E^X\left[X_t\right].$$

Extra: Estimate the parameters and their covariance matrix for the Cox-Ingersoll-Ross process by using exact moment, that is using $f(x) = \{x, x^2, x^3\}$.

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 minimising parameter value, the minimal function value and the Hessian. The cryptic name comes from *function minimization unconditional*.

`MLmax` Customized Quasi-Newton based optimization algorithm for *maximum likelihood estimation*. Maximises the likelihood function by using the score function's quadratic variations to estimate Fisher Information matrix. Needs the log-likelihood returned as a vector.

```
>> [xout,logL,CovM]=MLmax(@lnL,x0,indata)
```

References

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

[2] Shoji, I. and Ozaki, T. (1998). Estimation for nonlinear stochastic differential equations by a local linearization method, *Stochastic Analysis and Applications*, **16**, 733–752.