# 1 Geometric brownian motion with mean-reversion

## 1.1 Simulation

The corresponding SDE for geometric brownian motion with mean-reversion for spot prices is the following:

$$dS = \alpha(\mu - \ln S)Sdt + \sigma S dz.$$

For simplicity it can rewritten for the natural logarithm of the spot price by using Ito's lemma, $x = \ln S$, as follows:

$$F(t, x) = \ln x$$

$$F_t' = 0,\ F_x' = \frac{1}{x},\ F_{xx}'' = -\frac{1}{x^2}$$

$$\begin{aligned} dx = dF(t, S) &= \alpha(\mu - \ln S)S\frac{1}{S}dt - \frac{1}{2}\sigma^2\frac{S^2}{S^2}dt + \frac{S}{S}\sigma dz \\ &= [\alpha(\mu - x) - \frac{1}{2}\sigma^2]dt + \sigma dz. \end{aligned}$$

It can be discretised as

$$\Delta x_i = [\alpha(\mu - x_i) - \frac{1}{2}\sigma^2]\Delta t + \sigma\sqrt{\Delta t}\varepsilon_i. \tag{1}$$

In my R code it is implemented as following:

```
gbm_mr = function(x_0, alpha, mu, sigma, len, n){
  x = rep(0, n) #x will contain values of stochastic process
  # x[t] - value of stochastic process at time (t-1)*len/n
  x[1] = x_0 #initializing process at time zero
  dt = len/n #difference in time
  for (i in (2:n)){
    dx = (alpha*(mu - x[i-1]) - 1/2*sigma^2)*dt + sigma*sqrt(dt)*rnorm(1, 0, 1)
    x[i] = x[i-1] + dx
  }
  return = x
}
```

where *len* is the length of the time interval and $n$ is the number of observations.

## 1.2 Calibration

The algorithm for finding coefficients for (1) is based on linear regression. As a result from the Stochastic Analysis there exist exact solution for SDE, we can write it in discretized form:

$$x_{i+1} = x_i e^{-\alpha\Delta t} + \mu(1 - e^{-\alpha\Delta t}) + \sigma\sqrt{\frac{1 - e^{-2\alpha\Delta t}}{2\alpha}}\varepsilon_i$$

$$ax_i + b + \varepsilon.$$

Coefficients $a$ and $b$ can be found by fitting linear regression in R. So the relation between the linear fit and initial parameters:

$$\alpha = -\frac{\ln a}{\Delta t}$$

$$\mu = \frac{b}{1-a}$$

$$\sigma = sd(\varepsilon)\sqrt{\frac{-2\ln a}{\Delta t(1-a^2}}$$

In the program testing calibration of all the models is done for data simulated from functions, because it is easier to see whether parameters fit original ones.

# 2 Geometric Brownian motion with mean-reversion and stochastic volatility

## 2.1 Simulation

One of the variants of improvement of the previous model is the possibility of volatility to be non-constant. In the R program non-deterministic approach is implemented, i.e. volatility also follows SDE.

$$\begin{cases} dx = [\alpha(\mu - x) - \frac{1}{2}\sigma^2]dt + \sigma dz \\ dV = a(\bar{V} - V)dt + \sqrt{V}dw \\ V = \sigma^2 \end{cases}$$

It can be written in discrete from:

$$\begin{cases} \Delta x_i = [\alpha(\mu - x_i) - \frac{1}{2}\sigma^2]\Delta t + \sigma\sqrt{\Delta t}\varepsilon_{1,i} \\ \Delta V_i = a(\bar{V} - V_i)\Delta t + \sqrt{V_i\Delta t}(\rho\varepsilon_{1,i} + \sqrt{1-\rho^2}\epsilon_{2,i} \\ V = \sigma^2 \end{cases},$$

where $\varepsilon_1$ and $\varepsilon_2$ are independent standard random variables, $rho$ is the correlation coefficient to obtain random shock to the variance. The simulating is done by the following function:

```
gbm_sv = function(x_0, mu, alpha,len, n, a, avg_v, ksi, rho){
  dt = len/n
  x = rep(0, n)
  v = rep(0, n)
  x[1] = x_0
  v[0] = avg_v
  eps_1 = rnorm(n, 0, 1)
```

```
  eps_2 = rnorm(n, 0, 1)
  for (i in (2:n)){
    dv = a * (avg_v - v[i-1])*dt + ksi*sqrt(v[i-1]*dt)*(rho*eps_1[i]
        + sqrt(1 - rho^2)*eps_2[i])
    v[i] = v[i-1] + dv
    dx = (alpha*(mu - x[i-1]) - 1/2*v[i])*dt + sqrt(v[i])*sqrt(dt)*rnorm(1, 0, 1)
    x[i] = x[i-1] + dx
  }
  return = x
}
```

### 2.2 Calibration

Calibration of parameters $\mu$ and $\alpha$ is nearly the same as for model with constant volatility. To estimate the dynamics of volatility CARCH(1,1) is used. We need to modify the function of generating geometric brownian motion with stochastic volatility. In this approach we get volatility as a vector. New function for simulating:

```
gbm_sv_garch = function(x_0, alpha, mu, v, len, n){
  x = rep(0, n)
  x[1] = x_0
  for (i in (2:n)){
    dx = (alpha*(mu - x[i-1]) - 1/2*v[i])*dt + v[i]*sqrt(dt)*rnorm(1, 0, 1)
    x[i] = x[i-1] + dx
  }
  return = x
}
```

## 3 Geometric brownian motion with jumps

Another way to improve model of geometric brownian motion is to allow for spot prices to have jumps. It is useful, when very large price "strikes" exist in data and Brownian motion generate these large price changes is virtually zero. Dynamics of the spot prices with jumps can be modelled by the following SDE:

$$dS = \mu S dt + \sigma S dz + \kappa S dq.$$

It can be discretised as follows:

$$\Delta x_i = (r - \phi\kappa - \frac{1}{2}\sigma^2)\Delta t + \sigma\sqrt{\Delta t}\varepsilon_{1i} + (\kappa + \gamma\varepsilon_{2i})(u_i > \phi\Delta t).$$

### 3.1 Calibration

The calibration for jump coefficients is done by the following iterative procedure. For the purpose of finding jumps the standard deviation of changes of prices is found. If this change of price is greater that three times standard deviation, we call it jump and delete. Then do the same procedure again until the process stabilize. Then coefficients are found from the following relationship:

$$\phi = \text{number of jumps / time period of data}$$

$$\gamma = \text{sd(jump returns)}$$

As the mean jump size is usually very difficult to estimate robustly and should therefore usually be set to zero (Clewlow and Strickland, 2000).

Then we need to filter data from jumps and apply technique for estimating parameters of usual geometric brownian motion:

$$\Delta x_i = (r - \frac{1}{2}\sigma^2)\Delta t + \sigma\sqrt{\Delta t}\varepsilon_{1i} = a + \sigma\sqrt{\Delta t}\varepsilon_{1i}.$$

$r$ and $\sigma$ can be estimated as the coefficients from linear regression.

# 4 Geometric brownian motion with jumps and seasonality

This approach combine method of simulating jumps and non-constant volatility. This model is nearly the same as geometric Brownian motion with jumps but with the possibility of volatility to be a vector.

The calibration of non-constant volatility is done by applying GARCH(1,1) model to the filtered from jumps spot prices.