

Worcester Polytechnic Institute  
Department of Mathematical Sciences  
Professor: Stephan Sturm  
Teaching Assistant: Qingyun Ren

Spring 2017

## MA 573

# Computational Methods of Mathematical Finance

## Assignment 3

due on Thursday, February 9, in class

1. Assume a financial markets where stock and money market account follow the dynamics

$$\begin{aligned} dS_t &= \mu S_t dt + \sigma S_t dW_t, & S_0 &= 100; \\ dB_t &= r B_t dt, & B_0 &= 1, \end{aligned}$$

with  $\mu = 0.03$ ,  $\sigma = 0.25$  and  $r = 0.01$ . You want to calculate the price of a European call option with maturity  $T = 1$  and strike  $K = 120$  using Monte-Carlo integration.

- (a) Determine a priori analytically how many samples you will need that you get the correct price up to the dime (i.e., the price is in an interval of  $\pm \$0.05$  around the true price) that is correct in 95% of the cases.
- (b) Perform the Monte Carlo calculation 100 times with the number of samples given by (a). Count how often the price is correct up to the dime and how often not.
- (c) How many samples do you will have to use to get a price that is correct up to the cent in 95% of the cases?

2. Assume that the financial loss of a firm is given by the random variable  $X = 100.000 * Y - 220.000$  where  $Y$  is a Weibull distributed random variable with shape parameter  $k = 0.6$  and scale parameter  $\lambda = 1$ . Note that the Weibull distribution is given via the density

$$f_Y(y) = \begin{cases} \frac{k}{\lambda} \left(\frac{x}{\lambda}\right)^{k-1} e^{-\left(\frac{x}{\lambda}\right)^k} & \text{if } x > 0; \\ 0 & \text{if } x \leq 0. \end{cases}$$

- (a) Calculate the 95% value-at risk of the loss  $X$  analytically.
  - (b) Calculate the value-at-risk of the loss  $Y$  numerically, using the a sample size of 100,000.
3. Assume that  $(X^1, X^2)^T$  is a bivariate normal distribution with mean vector  $(\mu_1, \mu_2)^T = (2, 3)^T$  such that  $X^1$  has variance  $\sigma_1^2 = 3.2$ ,  $X^2$  has variance  $\sigma_2^2 = 0.7$ , and correlation  $\rho = \rho_{12} = 0.6$ .
- (a) Calculate the covariance matrix  $\Sigma$  of  $(X^1, X^2)^T$ .
  - (b) Calculate the Cholesky factorization of  $\Sigma$  by hand.
  - (c) Calculate the Cholesky factorization of  $\Sigma$  by writing a Python algorithm based on the ideas discussed in class. Compare it with the result from (b) as well as the built-in numerical method in numpy.
  - (d) Produce 1000 samples of the random vector  $(X^1, X^2)^T$  (using the Cholesky factorization from (b), (c) and the built-in random number generator for independent standard normals). Plot the samples (in a two-dimensional scatterplot). Discuss the plot by relating the given parameters to properties of the plot.
4. Assume that a company holds four assets  $A = (A^1, A^2, A^3, A^4)^T$  such that the losses of the assets are multivariate normal distributed with mean vector  $(\mu_1, \mu_2, \mu_3, \mu_4)^T = (-1000, -700, 300, -200)$  and covariance matrix

$$\Sigma = \begin{pmatrix} 144 & 72 & 120 & 300 \\ 72 & 100 & 180 & 230 \\ 120 & 180 & 389 & 880 \\ 300 & 230 & 880 & 4469 \end{pmatrix}.$$

The asset weights are given by  $w = (w_1, w_2, w_3, w_4) = (0.4, 0.2, 0.3, 0.1)$ , thus the whole portfolio will face the loss  $L = wA$ . Calculate the 99% value at-risk of the loss  $L$  using a simulation with 10,000 samples.

5. (a) Simulate Brownian motion by adding iteratively up scaled standard normal variables with scaling factor  $\Delta_t := t_k - t_{k-1} = \frac{T}{N}$  where  $T$  is the time horizon plotted and  $N$  the number of increments.
  - (b) Plot 10 independently simulated paths of Brownian motion in one picture, using  $T = 1$  and  $N = 10, 100, 1000, 10,000$  (to be clear: for every value of  $N$  10 paths, for every  $N$  a different plot).
  - (c) Now generate the Brownian paths using the method via Cholesky factorization. Compare the run-time of both methods and comment on it.
6. Assume a financial markets where stock and money market account follow the dynamics

$$\begin{aligned} dS_t &= \mu S_t dt + \sigma S_t dW_t, & S_0 &= 100; \\ dB_t &= r B_t dt, & B_0 &= 1, \end{aligned}$$

with  $\mu = 0.17$ ,  $\sigma = 0.25$  and  $r = 0.02$ . We want to price an up-and-out barrier put option with maturity  $T = 2$ , strike  $K = 105$  and  $B = 120$ .

- (a) Produce 10,000 sample paths of geometric Brownian motion of the stock price under the risk neutral measure by producing the underlying Brownian motion stepwise from increments with stepsize  $\Delta_t := t_k - t_{k-1} = \frac{1}{5000}$ .
- (b) Use the sample paths to calculate by Monte-Carlo integration the price of the Barrier options.
- (c) In how many cases the actual payoff of the option is zero due to the fact that the barrier was hit before maturity? How des this number change if we use instead barrier values  $B = 105$  and  $B = 180$ ?
- (d) Based on your investigation in (c), is there any method how the calculation can be sped up?

*Note:* All programming problems should be either in Python 2.7 or Python 3.5. Matlab and R are accepted, but no support for these languages is provided. Please comment the programs *extensively* and send them in a .zip file with title **Lastname\_HW3.zip** and sujet line "MA 573 HW3 **Lastname**" to Qingyun Ren [qren@wpi.edu](mailto:qren@wpi.edu) before the due date of the homework (replacing the bold words by your actual last name). Please provide printouts of programs amd plots that one can comment on them.