Jacobs University
Fall 2022

November 16, 2022

Stochastic Methods Lab

Assignment Sheet 9

Due on November 23, 2022

Note: The work is to be submitted via `git`, as discussed in class. The coding language is Python. Please make sure that your code actually runs and produces the requested output. Please make your code readable for the instructor and TA, and include comments wherever necessary. Please submit .py source code, not jupyter notebooks. Theoretical questions may be submitted as a scan of handwritten notes or typed up (e.g., using LaTeX). The submission deadline is midnight of the stated due date.

Problem 1 [10 points]

Let us consider the exponential Ornstein-Uhlenbeck stochastic process

$$dX_t = \mu(1 - c \ln X_t)X_t dt + \sigma X_t dW_t, \tag{1}$$

with $\mu, \sigma > 0$ and $0 \leq c \leq 1$.

(a) Generate an ensemble of paths of the process (1) on the interval $[0, 1]$, and plot the empirical mean and standard deviation together with 10 sample paths. Produce at least two plots with reasonable parameters such that the influence of the parameter c as compared to geometric Brownian motion becomes visible. Briefly describe what happens when c is near zero or c is near 1.

(b) Now assume the model (1) for the stock price (instead of geometric Brownian motion as we did in class). Use Ito's lemma and "Merton's trick" to derive a deterministic equation for the option price (similar to the Black-Scholes equation we derived in class). The result might seem surprising; briefly explain why it isn't.

Problem 2 [10 points]

Rewrite your binomial tree function `binomial_tree(payoff,n,r,sigma,S,K,T)` from one of the previous assignment sheets so that it stores the option value at each node of the tree. Then visualize the tree using `imshow` for some reasonable parameters. Here, you have to think about an appropriate color map, how to mask the missing values (hint: use Numpy's masked arrays), and how to best map the computed values to pixel coordinates.