Assignment for Module 3

April 2023

Instructions: This is a Computational Finance mini project, comprising THREE (3) questions, to be completed using Python. C++ is also allowed, but Excel/VBA is not permitted. As this is the half way point of the CQF, this assessment is designed for delegates to show independence and maturity in interpretation of slightly open ended problems. It will test

- finding and understanding the relevant lectures, Python labs and tutorials in module 3; as well as the Python labs,
- ability to experiment and demonstrate initiative in mathematical and numerical methods,
- willingness to work outside narrow instruction that are typical of maths based tests/exams.

Queries will need to be sent through the Support button on the portal

Tasks

1. **Outline:** Compare the **Euler-Maruyama scheme**, **Milstein scheme** and **closed form solution** for a Geometric Brownian motion. As an initial example you may use the following set of sample data

$$\begin{aligned} \text{Today's stock price } S_0 &= 100 \\ \text{Time } T &= 1 \text{ year} \\ \text{volatility } \sigma &= 20\% \\ \text{constant risk-free interest rate } r &= 5\% \end{aligned}$$

You may find it useful to use seed() in the random number generation. Comment on your results. Hence use the expected value of the discounted payoff under the risk-neutral density $\mathbb{Q}$

$$V(S,t) = e^{-r(T-t)}\mathbb{E}^{\mathbb{Q}}\left[\textbf{Payoff}\right],$$

for the appropriate form of payoff, to consider **Asian** call and put options. As a guide, include fixed and floating strike; discrete and continuous sampling; arithmetic and geometric averaging. Consider using the above data with,

$$\begin{aligned} \text{Strike } E &= 100 \\ \text{Time to expiry } (T-t) &= 1 \text{ year} \end{aligned}$$

Then vary the data to see the affect on the option price. **Note: There is no additional credit for calculating the greeks. [40 Marks]**

2. **Outline:** The model problem is

$$\frac{d^2y}{dx^2} + P(x)\frac{dy}{dx} + Q(x)\,y = f(x), \tag{1}$$

subject to the boundary conditions $y(a) = \alpha$, and $y(b) = \beta$.
Suppose $a = x_0 < x_1 < x_2 < \cdots < x_{n-1} < x_n = b$ represents a regular partition of the interval $[a, b]$. This means $x_i = a + ih$, where $i = 0, 1, 2, ..., n$ and $\delta x = \dfrac{b-a}{n}$. The points

$$x_1 = a + \delta x,\ x_2 = a + 2\delta x, \ldots, x_{n-1} = a + (n-1)\,\delta x$$

are called interior mesh points of the interval $[a, b]$. We can approximate the derivative terms using a Taylor series expansion. Let

$$y_i = y(x_i),\ P_i = P(x_i),\ Q_i = Q(x_i),\ f_i = f(x_i)$$

Using numerical approximations for each derivative term in (1) show that this can be expressed as

$$\left(1 + \frac{\delta x}{2}P_i\right)y_{i+1} + \left(-2 + \delta x^2 Q_i\right)y_i + \left(1 - \frac{\delta x}{2}P_i\right)y_{i-1} = \delta x^2 f_i;\quad i = 1, ..., n-1. \tag{2}$$

We know the boundary conditions are

$$\begin{aligned} y_0 &= y(x_0) = y(a) = \alpha && (3)\\ y_n &= y(x_n) = y(b) = \beta. && (4) \end{aligned}$$

Show that (2), (3) and (4) can be expressed as a matrix inversion problem $A\mathbf{x} = \mathbf{b}$, and give the forms of A, $\mathbf{x}$ and $\mathbf{b}$ for an arbitrary value n.
Use the method you have developed above, and $n = 10$, to solve the following differential equation problem

$$\frac{d^2y}{dx^2} + 3\frac{dy}{dx} + 2y = 4x^2,\ y(1) = 1,\ y(2) = 6.$$

The coefficients in (2) **must** be generated inside your computer program using appropriate control structures. Use any Python linear systems function to solve the resulting linear system and output your results in two columns (x_i, y_i). Repeat the exercise for $n = 50, 100$. What do you notice? Plot a graph for varying n. **[30 Marks]**

3. This question is on *Monte Carlo* integration. Calculate the following

i. $\displaystyle\int_1^3 x^2 dx$

ii. $\displaystyle\int_0^\infty \exp(-x^2)\,dx$

iii. $\displaystyle\frac{1}{\sqrt{2\pi}}\int_{-\infty}^\infty x^4 \exp(-x^2/2)\,dx$

Compare your computed solutions to the exact values and show graphically how the error in each case behaves as the random numbers increase. **[30 Marks]**

Your completed assignment should centre on a report to include, for each question:

- Brief outline of the problem and numerical procedure used
- Results - appropriate tables and comparisons.
- Any interesting observations and problems encountered.
- Conclusion and references

For a Python Jupyter Notebook, a detailed notebook will become the complete report (write-up, code, results).

Score key

60-65 Pass

66-70 Good

71-79 Very Good

80-89 Excellent

90-95 Outstanding

96+ Exceptional

Note: An assessment of this form differs from mathematical exercises that can attract full marks. The key above is provided for this reason.