Mathematical Modeling and Computation in Finance

With Exercises and Python and MATLAB Computer Codes

C.W. Oosterlee & L.A. Grzelak

Solutions to exercises from Chapter 4

`https://QuantFinanceBook.com`

Ex. 4.1. *a.* The code can be found under the Python icon.

b. The following code under the Python icon provides us with the solution.

Based on the 3D scatter plot diagram of the implied volatility versus the strike price, one can clearly observe the volatility smile. The volatility smile becomes more pronounced when the time to maturity is close to 0.

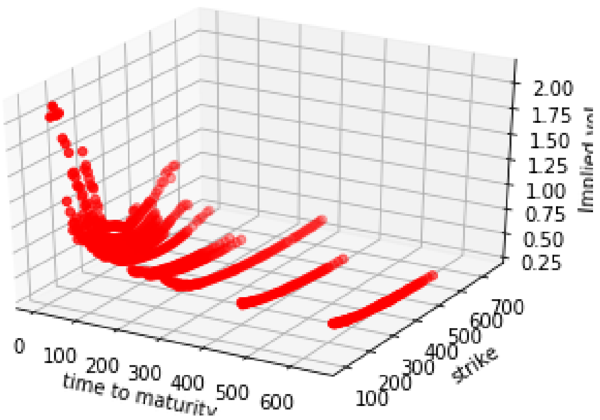

Figure 1: Results for Exercise 4.1b

Ex. 4.3. *a.* Remember that,

$$
\begin{aligned}
d_{1,2} &= \frac{\log(S/K) + (r \pm \sigma^2/2)(T-t)}{\sigma\sqrt{T-t}} \\
&= \left(\frac{X_F}{t_*} \pm \frac{t_*}{2}\right).
\end{aligned}
$$

b. The condition $X_F \leq 0$ implies $S\mathrm{e}^{r(T-t)} \leq K$. This can be interpreted as the definition of forward moneyness. It provides information about the forward value of the underlying, as compared to the strike price.

c. We obtain,

$$p = \frac{V_p(t, S(t))}{S(t)} = \frac{K\mathrm{e}^{-r(T-t)}}{S(t)} F_{\mathcal{N}(0,1)}(-d_2) - F_{\mathcal{N}(0,1)}(-d_1)$$
$$= \mathrm{e}^{-X_F} F_{\mathcal{N}(0,1)}(-d_2) - F_{\mathcal{N}(0,1)}(-d_1).$$

d. We find,

$$\frac{\partial p}{\partial t_*} = \mathrm{e}^{-X_F} f_{\mathcal{N}(0,1)}(-d_2) \left(\frac{X_F}{t_*^2} + \frac{1}{2} \right) - f_{\mathcal{N}(0,1)}(-d_1) \left(\frac{X_F}{t_*^2} - \frac{1}{2} \right).$$

Note that $d_2 = d_1 + t_*$ and $f_{\mathcal{N}(0,1)}(-d_1) = f_{\mathcal{N}(0,1)}(-d_2)\mathrm{e}^{-X_F}$. Substituting this, gives us,

$$\frac{\partial p}{\partial t_*} = f_{\mathcal{N}(0,1)}(-d_1).$$

This derivative has a probabilistic interpretation, as it is equal to the probability density function of the standard normal distribution evaluated at $-d_1$. It is the sensitivity of a scaled put with respect to a scaled time.

Ex. 4.5. *a.* We have,

$$\frac{\partial^2 c}{\partial y \partial w} = -d_1 \frac{\partial d_1}{\partial y} \frac{\partial c}{\partial w}$$
$$= \frac{\partial c}{\partial w} \left(\frac{1}{2} - \frac{y}{w} \right).$$

b. We start with,

$$\frac{\partial c}{\partial y} = c - S_0 F_{\mathcal{N}(0,1)}(d_1),$$
$$\frac{\partial^2 c}{\partial y^2} = \frac{\partial c}{\partial y} + \frac{S_0}{\sqrt{w}} f_{\mathcal{N}(0,1)}(d_1)$$
$$= \frac{\partial c}{\partial y} + 2\frac{\partial c}{\partial w}.$$

Ex. 4.7. The following computer code provides us with the requested answers.

As output, we obtain,

```
a. value1 = 0.13688917293214575,
b. value2 = 0.024216392555889792,
c. value3 = 0.747197522225466,
d. value4 = 2.6582099178678.
```

For payoffs 2 and 3 we are able to determine prices analytically,

`value2 = 0.023877663895613327 and value3 = 0.7465691813081832.`

Ex. 4.9. We have the following code, providing us with the desired plots.

The butterfly arbitrage appears only in the nearest neighbor interpolation. Although the linear and cubic spline interpolation routines appear to be free of butterfly arbitrage, the linear interpolation does not give rise to a differentiable probability distribution. It is clear that smoothening the interpolation by preserving the slope will reduce arbitrage. See Figure 2.

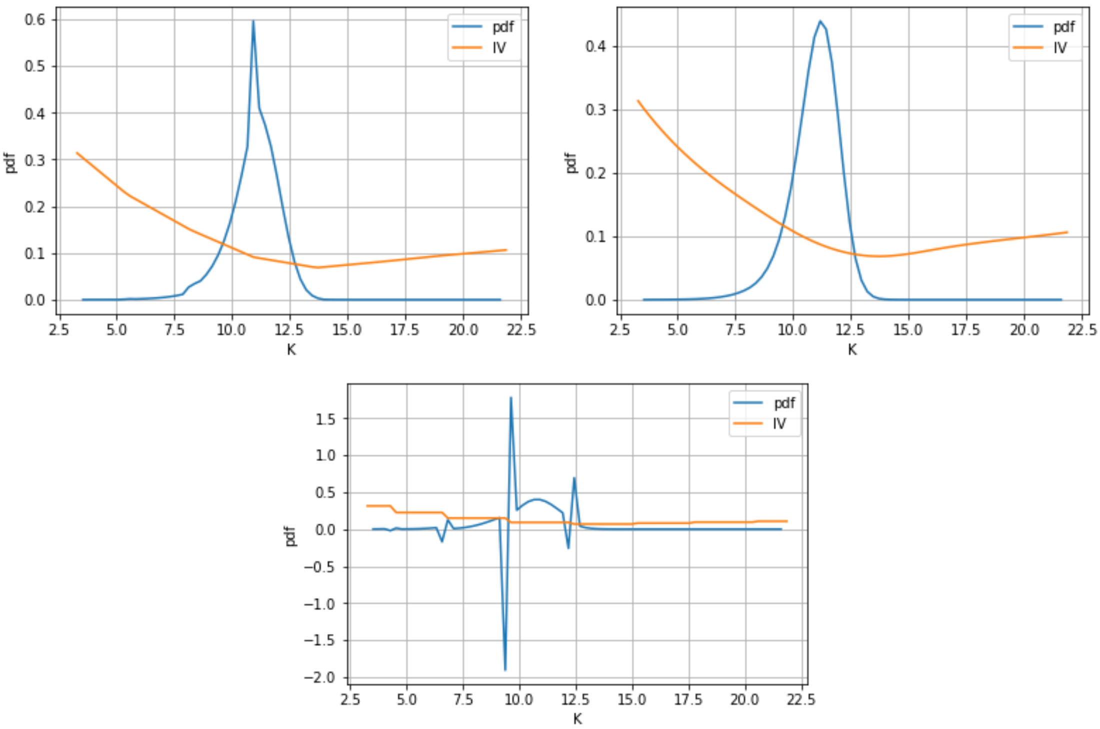

Figure 2: Results for Exercise 4.9 for different interpolation routines. Upper left figure: linear. Upper right figure: cubic spline. Lower figure: nearest neighbor interpolation.

Ex. 4.11. *a.* We start with the first expression,

$$\mathbb{E}^{\mathbb{Q}}[\log(S_1(T))^{S_2(T)}|\mathcal{F}_0] \quad = \quad \mathbb{E}^{\mathbb{Q}}[\log(S_1(T))|\mathcal{F}_0]\mathbb{E}^{\mathbb{Q}}[S_2(T)|\mathcal{F}_0],$$

which is based on the independence of the assets S_1 and S_2. The Feynman-Kac theorem relates the expected value of a payoff to the value of an option at time t. Therefore,

$$\mathbb{E}^{\mathbb{Q}}[H(S,T)|\mathcal{F}_0] = \mathrm{e}^{r(T-t_0)}V(t_0,S).$$

We can determine the value at the right-hand side by using the Breeden-Litzenberger model, for which the value is determined as,

$$\begin{aligned} V(t_0,S_0;K,T) \quad &= \quad \mathrm{e}^{-r(T-t_0)}H(S_F,T) + \int_0^{S_F} V_p(t_0,S_0;y,T)\frac{\partial^2 H(T,S)}{\partial y^2}dy \\ &\quad + \int_{S_F}^{\infty} V_c(t_0,S_0;y,T)\frac{\partial^2 H(T,S)}{\partial y^2}dy. \end{aligned}$$

To be able the compute the first expectation, consider the payoff function $H(T,S(T)) = S(T)$. Both integrals vanish due to the vanishing second derivative of the payoff function. So,

$$V(t_0,S_0;K,T) = \mathrm{e}^{-r(T-t_0)}H(S_F,T) = S_0,$$

since $S_F = S(t_0)\mathrm{e}^{r(T-t_0)}$. And,

$$\mathbb{E}^{\mathbb{Q}}[S_2(T)|\mathcal{F}_0] = S_2(t_0)\mathrm{e}^{r(T-t_0)}.$$

For the other expectation, the payoff becomes $H(T,S(T)) = \log(S(T))$. Hence,

$$\begin{aligned} \mathbb{E}^{\mathbb{Q}}[\log(S_1(T))|\mathcal{F}_0] =& r(T-t_0)\log(S_1(t_0)) - \mathrm{e}^{r(T-t_0)}\int_0^{S_F} V_p(t_0,S_0;y,T)\frac{1}{y^2}dy \\ &- \mathrm{e}^{r(T-t_0)}\int_{S_F}^{\infty} V_c(t_0,S_0;y,T)\frac{1}{y^2}dy. \end{aligned}$$

b. We rewrite the expression as the expected value over a logarithm, as follows,

$$\mathbb{E}^{\mathbb{Q}}[\log \prod_{i=1}^{N} \frac{S_1(t_{i+1})}{S_1(t_i)} \frac{S_2(t_{i+1})}{S_2(t_i)} | F(t_0)] = \mathbb{E}^{\mathbb{Q}}[\log(S_1(t_{t_N}))|F(t_0)] - \mathbb{E}^{\mathbb{Q}}[\log(S_1(t_{t_1}))|F(t_0)]$$
$$+\mathbb{E}^{\mathbb{Q}}[\log(S_2(t_{t_N}))|F(t_0)] - \mathbb{E}^{\mathbb{Q}}[\log(S_2(t_{t_1}))|F(t_0)].$$

Each of these terms can be calculated via the expression, which was derived in part *a*).