

**Interest Rate Modelling and Derivative Pricing, WS 2018/19**

**Exercise 3**

**1. Black-Formula for log-normal model**

Consider the lognormal model

$$dS(t) = \sigma \cdot S(t) \cdot dW(t)$$

with constant volatility  $\sigma$ .

(a) Show that the forward price of a call option is given by

$$\mathbb{E} \left[ [S(T) - K]^+ \mid \mathcal{F}_t \right] = \text{Black} \left( S(t), K, \sigma\sqrt{T-t}, 1 \right)$$

with

$$\text{Black}(F, K, \nu, \phi) = \phi \cdot [F \cdot \Phi(\phi \cdot d_1) - K \cdot \Phi(\phi \cdot d_2)], \quad d_{1,2} = \frac{\ln(F/K)}{\nu} \pm \frac{\nu}{2}$$

and  $\Phi$  being the cumulated standard normal distribution function.

(b) Use put-call parity to derive the put option price

$$\mathbb{E} \left[ [K - S(T) - K]^+ \mid \mathcal{F}_t \right] = \text{Black} \left( S(t), K, \sigma\sqrt{T-t}, -1 \right)$$

**2. Shifted Log-normal model**

Consider the shifted log-normal model

$$dS(t) = \frac{\sigma}{\lambda} \cdot [S(t) + \lambda] \cdot dW(t)$$

with shift  $\lambda$  and re-scaled shifted log-normal volatility  $\sigma/\lambda$ .

(a) Show that the shifted log-normal model converges to the normal model with normal volatility  $\sigma$  in the sense that

$$\lim_{\lambda \rightarrow \infty} \mathbb{E} \left[ [S(T) - K]^+ \mid \mathcal{F}_t \right] = \text{Bachelier} \left( S(t), K, \sigma\sqrt{T-t}, 1 \right).$$

(b) Confirm your results numerically with Python and illustrate the result.

**3. SABR model volatility approximation**

Consider the SABR model with implied normal volatility approximation

$$\sigma_N(S(t), K, T) = \nu \cdot \frac{S(t) - K}{\hat{\chi}(\zeta)} \cdot [1 + I^1(S(t), K) \cdot T]$$

with

$$S_{av} = \frac{S(t) + K}{2}, \quad \zeta = \frac{\nu}{\alpha} \cdot \int_K^{S(t)} \frac{dx}{C(x)}, \quad \hat{\chi}(\zeta) = \ln \left( \frac{\sqrt{1 - 2\rho\zeta + \zeta^2} - \rho + \zeta}{1 - \rho} \right),$$

$$I^1(K) = \frac{2\gamma_2 - \gamma_1^2}{24} \alpha^2 C(S_{av})^2 + \frac{\rho\nu\alpha\gamma_1}{4} C(S_{av}) + \frac{2 - 3\rho^2}{24} \nu^2,$$

$$\gamma_1 = \frac{C'(S_{av})}{C(S_{av})}, \quad \gamma_2 = \frac{C''(S_{av})}{C(S_{av})}.$$

More concrete, we get with  $C(S) = S^\beta$  and  $\beta \in (0, 1)$

$$\zeta = \frac{\nu}{\alpha} \cdot \frac{S(t)^{1-\beta} - K^{1-\beta}}{1-\beta}, \quad \gamma_1 = \frac{\beta}{S_{av}}, \quad \gamma_2 = \frac{\beta(\beta-1)}{S_{av}^2}$$

- (a) Based on the formula for  $\sigma_N$ , derive the volatility approximation for the normal SABR model with  $\beta = 0$  via

$$\lim_{\beta \rightarrow 0} \sigma_N(S(t), K, T).$$

- (b) Based on the formula for  $\sigma_N$ , derive the volatility approximation for the CEV model with  $\beta > 0$  and  $\nu = 0$  via

$$\lim_{\nu \rightarrow 0} \sigma_N(S(t), K, T).$$

- (c) Show that, for  $\beta \rightarrow 0$  and  $\nu \rightarrow 0$  above approximation converges to the normal volatility, i.e.

$$\lim_{\beta, \nu \rightarrow 0} \sigma_N(S(t), K, T) = \alpha.$$

- (d) Confirm your results numerically, e.g. via Python and illustrate the results.

#### 4. Shifted SABR model

Consider the SABR model normal volatility approximation for general local volatility functions  $C(S)$  in Exercise 3.

- (a) Derive the approximation for the shifted SABR model with local volatility function

$$C(S) = (S + \lambda)^\beta$$

- (b) Recall the model-independent pricing result for shifted processes  $\tilde{S}(t) = S(t) - \lambda$  and Vanilla option pricing formula  $V(S(t), K)$ ,

$$\tilde{V}(\tilde{S}(t), K) = \mathbb{E} \left[ (\tilde{S}(T) - K)^+ \mid \mathcal{F}_t \right] = V(\tilde{S}(t) + \lambda, K + \lambda)$$

Is the approximation from (a) consistent to this model-independent result? Does your conclusion change if you replace the arithmetic average  $S_{av} = \frac{S(t)+K}{2}$  by the geometric average  $S_{av} = \sqrt{S(t) \cdot K}$ ?

- (c) Modify the Python code in `SabrModel.py` to allow for a shift parameter  $\lambda$ . Analyse the approximation accuracy of the normal volatility formula from (a) compared to the model-implied normal volatilities obtained via Monte-Carlo simulation. Does approximation accuracy change if we incorporate a shift in the formula?