bsm_geometric_asian_option

January 30, 2019

Black and Scholes formula

Recall that BS model assumes the distribution of stock as lognormal. In particular, it writes

$$\ln \frac{S(T)}{S(0)} \sim \mathcal{N}((r - \frac{1}{2}\sigma^2)T, \sigma^2 T)$$

with respect to risk neutral measure. In the above, the parameters stand for

- $S(0)$: The initial stock price
- $S(T)$: The stock price at T
- r: interest rate
- σ: volatility

The call and put price with maturity T and K will be known as C_0 and P_0 given as below:

$$C_0 = \mathbb{E}[e^{-rT}(S(T) - K)^+] = S_0 e^{-\delta T}\Phi(d_1) - Ke^{-rT}\Phi(d_2),$$

and

$$P_0 = \mathbb{E}[e^{-rT}(S(T) - K)^-] = Ke^{-rT}\Phi(-d_2) - S_0 e^{-\delta T}\Phi(-d_1),$$

where d_i are given as

$$d_1 = \frac{(r + \frac{1}{2}\sigma^2)T - \ln \frac{K}{S_0}}{\sigma\sqrt{T}},$$

and

$$d_2 = \frac{(r - \frac{1}{2}\sigma^2)T - \ln \frac{K}{S_0}}{\sigma\sqrt{T}},$$

Ex.

Find BS call and put price for the given parameters below

```
In [1]: #An example is given here
        s0 = 100.0
        K = 110.0
        r=0.0475
        sigma = 0.20
        maturity = 1.
```

```
In [2]: import numpy as np
        from contract_v01 import VanillaOption
        from sde_1d_v01 import Sde_1d, Gbm_1d
```

```python
In [3]: #initiate option and sde instances
        option1 = VanillaOption(otype = 1, strike = 110., maturity= 1., market_price=15.)
        option2 = VanillaOption(otype = -1, strike = 110., maturity= 1., market_price=15.)
        gbm1 = Gbm_1d(init_state=100., drift_ratio=.0475, vol_ratio=.2)
        maturity = option1.maturity
        print('>>>>>>>>>> Exact call value is ' + str(gbm1.bsm_price(option1)))
        print('>>>>>>>>>> Exact put value is ' + str(gbm1.bsm_price(option2)))
```

```
>>>>>>>>>> Exact call value is 5.943273183452838
>>>>>>>>>> Exact put value is 10.84042522804176
```

Application to Geometric asian option price

Geometric asian call option with maturity T and strick K has its pay off as

$$C(T) = (A(T) - K)^+,$$

where $A(T)$ is geometric average of the stock price at times $0 \le t_1 < t_2, \ldots, < t_n = T$, i.e.

$$A(T) = (S(t_1)S(t_2)\ldots S(t_n))^{1/n}.$$

The call price can be thus written by

$$C_0 = \mathbb{E}[e^{-rT}(A(T) - K)^+].$$

We set $t_0 = 0$. Under the above BS model, one can show that the distribution of $A(T)$ is again a lognormal in the form of

$$\ln \frac{A(T)}{S(0)} \sim \mathcal{N}(\hat{\mu}T, \hat{\sigma}^2 T),$$

where

$$\mu = r - \frac{1}{2}\sigma^2$$

$$\hat{\mu}T = \frac{\mu}{n}\sum_{i=1}^{n} t_i,$$

$$\hat{\sigma}^2 T = \frac{\sigma^2}{n^2}\sum_{j=0}^{n-1}(n-j)^2(t_{j+1} - t_j).$$

Therefore, we can rewrite the price formula as, with $Z \sim \mathcal{N}(0,1)$

$$C_0 = \mathbb{E}[e^{-rT}(S(0)e^{\hat{\mu}T + \hat{\sigma}\sqrt{T}Z} - K)^+] = e^{(\hat{r}-r)T} BSM(S(0), \hat{r}, \hat{\sigma}, K, T, call)$$

with

$$\hat{r} = \hat{\mu} + \frac{1}{2}\hat{\sigma}^2.$$

Put formula is similar.

$$P_0 = \mathbb{E}[e^{-rT}(S(0)e^{\hat{\mu}T + \hat{\sigma}\sqrt{T}Z} - K)^-] = e^{(\hat{r}-r)T} BSM(S(0), \hat{r}, \hat{\sigma}, K, T, put)$$

ex

If the time stepsize is uniform m partitions on $[0, T]$ for the geometric asian option, the average points include actually $n = m + 1$ points:

$$t_i = \frac{T}{m}(i-1), \ \forall i = 1, 2, \ldots m+1,$$

Note that $t_1 = 0$. Then one can determine $\hat{r}$ and $\hat{\mu}$ as follows:

$$\hat{\mu} = \frac{\mu}{2}$$

$$\hat{\sigma}^2 = \frac{\sigma^2(2m+1)}{6(m+1)}$$

$$\hat{r} = \hat{\mu} + \frac{1}{2}\hat{\sigma}^2.$$

In [5]:
```
'''==============
Test BSM geometric asian option price
==============='''
gbm1 = Gbm_1d(init_state=100., drift_ratio=0.0475, vol_ratio=.2)
gao1 = gbm1.bsm_geometric_asian_price(
                            otype = 1,
                            strike = 110.,
                            maturity = 1,
                            num_step = 4
                            )
print('>>>>>> geometric call option value is ' + str(gao1))
```

```
>>>>>> geometric call option value is 1.607164727431734
```