
#Question No: 02

```
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
from math import erf
```

```
def remove(X):
    y = [x for x in X if not np.isnan(x)]
    return y
```

```
def read(filename):
    df = pd.read_csv(filename)
    df.set_index('Date',inplace=True)
    data = df.to_dict()
    for key, vals in data.items():
        data[key] = remove(list(vals.values()))
    return data
```

```
def historical_volatility(prices, duration):
    required_price = prices[-duration:]
```

```
    R = []
    for i in range(1, len(required_price)):
        ri = (required_price[i] - required_price[i-1])/required_price[i-1]
        R.append(ri)
```

```
    var = np.var(R)
    sigma_d = np.sqrt(var)
    sigma_a = np.sqrt(252)*sigma_d
    return sigma_a
```

```
def d(x, tau, sig, K, r):
    d_plus = (1/(sig*np.sqrt(tau)))*(np.log(x/K) + tau*(r + (sig*sig)/2))
    d_minus = (1/(sig*np.sqrt(tau)))*(np.log(x/K) + tau*(r - (sig*sig)/2))

    return d_plus, d_minus
```

```
def N(x):
    return 0.5*(1 + erf(x/np.sqrt(2)))
```

```
def C(t, x, T, sig, K, r):
    if x == 0:
        return 0
    if t == T:
        return max(x - K, 0)

    tau = T-t
    d_plus, d_minus = d(x, tau, sig, K, r)

    price = x*N(d_plus) - K*np.exp(-r*tau)*N(d_minus)

    return price
```

```
return price
```

```
def P(t, x, T, sig, K, r):  
    call_price = C(t,x,T,sig,K,r)  
    put_price = call_price + K*np.exp(-r*(T-t)) - x  
  
    return put_price
```

```
bse_data = read('bsedata1.csv')  
nse_data = read('nsedata1.csv')
```

```
def function():
```

```
    r = 0.05  
    T = 0.5  
    t = 0  
    n_days = 20
```

```
A = np.arange(0.5, 1.51, 0.1)
```

```
for company, prices in bse_data.items():  
    S_0 = prices[-1]  
    K = S_0  
    sig = historical_volatility(prices, n_days)  
    call_price = C(t, S_0, T, sig, K, r)  
    put_price = P(t, S_0, T, sig, K, r)  
    print(company, "(BSE) (K = S_0)")  
    print("The Call Price = %.6f"%call_price)  
    print("The Put Price = %.6f"%put_price)  
    print("***** ")
```

```
data = {"K":[], "Call Price":[], "Put Price":[]}
```

```
for a in A:
```

```
    K = a*S_0  
    call_price = C(t, S_0, T, sig, K, r)  
    put_price = P(t, S_0, T, sig, K, r)  
    data["K"].append(K)  
    data["Call Price"].append(call_price)  
    data["Put Price"].append(put_price)
```

```
df = pd.DataFrame(data)
```

```
df.to_csv(f"function_{company}.csv", header=True, index=False)
```

```
for company, prices in nse_data.items():  
    S_0 = prices[-1]  
    K = S_0  
    sig = historical_volatility(prices, n_days)  
    call_price = C(t, S_0, T, sig, K, r)  
    put_price = P(t, S_0, T, sig, K, r)  
    print(company, "(NSE) (K = S_0)")  
    print("The Call Price = %.6f"%call_price)  
    print("The Put Price = %.6f"%put_price)  
    print("*****")
```

```
data = {"K":[], "Call Price":[], "Put Price":[]}
```

```
for a in A:
```

```
    K = a*S_0  
    call price = C(t. S 0. T. sig. K. r)
```

```

        put_price = P(t, S_0, T, sig, K, r)
        data["K"].append(K)
        data["Call Price"].append(call_price)
        data["Put Price"].append(put_price)

df = pd.DataFrame(data)
df.to_csv(f"function_{company}.csv", header=True, index=False)

def main():

    function()

if __name__ == '__main__':
    main()

print("*****")

The Put Price = 13.957949
*****
Havells (BSE) (K = S_0)
The Call Price = 246.893889
The Put Price = 229.800941
*****
Reliance (BSE) (K = S_0)
The Call Price = 449.376493
The Put Price = 421.697668
*****
BombayDyeing (BSE) (K = S_0)
The Call Price = 92.469301
The Put Price = 89.659569
*****
Berger Paints (BSE) (K = S_0)
The Call Price = 101.176026
The Put Price = 93.012249
*****
Nokia (BSE) (K = S_0)
The Call Price = 1.972952
The Put Price = 1.829256
*****
ICICI (NSE) (K = S_0)
The Call Price = 126.922704
The Put Price = 118.030569
*****
TataMotors (NSE) (K = S_0)
The Call Price = 63.409978
The Put Price = 59.146000
*****
Jp Morgan (NSE) (K = S_0)
The Call Price = 4.240006
The Put Price = 3.939281
*****
Voltas (NSE) (K = S_0)
The Call Price = 189.623536
The Put Price = 175.957573
*****
BajajFinsv (NSE) (K = S_0)
The Call Price = 2554.851342
The Put Price = 2394.934871

```

```
-----
*****
Havells (NSE) (K = S_0)
The Call Price = 247.518125
The Put Price = 230.444929
*****
Reliance (NSE) (K = S_0)
The Call Price = 447.433846
The Put Price = 420.010036
*****
Asian Paints (NSE) (K = S_0)
The Call Price = 353.633592
The Put Price = 319.732865
*****
Berger Paints (NSE) (K = S_0)
The Call Price = 102.928326
The Put Price = 94.773191
*****
MuthootFin (NSE) (K = S_0)
The Call Price = 129.765112
```