Black_Scholes_Stock_Calls

September 29, 2021

1 Black Scholes Stock Calls Inputs

[1]:
```
import numpy as np
import scipy.stats as ss
import matplotlib.pyplot as plt
import yfinance as yf
```

[2]:
```
dfo = yf.Ticker("AAPL")
```

[3]:
```
dfo.options
```

[3]:
```
('2020-10-15',
 '2020-04-23',
 '2021-09-16',
 '2020-06-18',
 '2020-07-16',
 '2021-06-17',
 '2020-04-30',
 '2020-05-07',
 '2022-06-16',
 '2020-05-14',
 '2020-04-16',
 '2020-09-17',
 '2020-05-28',
 '2021-01-14',
 '2020-05-21',
 '2022-01-20',
 '2020-12-17')
```

[4]:
```
dfo_exp = dfo.option_chain('2020-05-28')
```

[5]:
```
dfo_exp.calls
```

[5]:

	contractSymbol	lastTradeDate	strike	lastPrice	bid	ask	\
0	AAPL200529C00230000	2020-04-09 19:39:18	230.0	43.00	41.00	44.80	
1	AAPL200529C00242500	2020-04-09 14:05:04	242.5	33.28	30.60	34.80	
2	AAPL200529C00252500	2020-04-09 19:45:27	252.5	25.49	23.50	27.20	

3	AAPL200529C00255000	2020-04-09 15:46:43	255.0	23.62	21.80	24.75	
4	AAPL200529C00257500	2020-04-09 15:47:48	257.5	21.80	20.30	22.95	
5	AAPL200529C00265000	2020-04-09 18:22:32	265.0	16.62	16.45	17.65	
6	AAPL200529C00267500	2020-04-09 19:06:43	267.5	15.24	14.95	16.20	
7	AAPL200529C00270000	2020-04-09 19:04:16	270.0	13.80	13.55	14.75	
8	AAPL200529C00272500	2020-04-09 19:41:31	272.5	13.10	12.30	13.40	
9	AAPL200529C00275000	2020-04-09 19:51:26	275.0	11.45	10.95	12.10	
10	AAPL200529C00277500	2020-04-09 18:40:19	277.5	10.50	9.80	10.95	
11	AAPL200529C00280000	2020-04-09 19:59:54	280.0	9.05	8.65	9.40	
12	AAPL200529C00282500	2020-04-09 15:05:57	282.5	7.75	7.70	8.75	
13	AAPL200529C00287500	2020-04-09 18:48:52	287.5	6.03	5.75	6.85	
14	AAPL200529C00290000	2020-04-09 19:59:54	290.0	5.37	4.95	5.90	
15	AAPL200529C00295000	2020-04-09 19:49:46	295.0	3.90	3.55	4.50	
16	AAPL200529C00300000	2020-04-09 17:54:32	300.0	2.72	2.47	3.35	
17	AAPL200529C00305000	2020-04-09 19:42:12	305.0	2.03	1.64	2.39	
18	AAPL200529C00310000	2020-04-09 18:42:02	310.0	1.28	1.05	1.64	
19	AAPL200529C00315000	2020-04-09 16:19:40	315.0	0.95	0.74	0.99	
20	AAPL200529C00320000	2020-04-09 16:25:36	320.0	0.68	0.45	0.67	
21	AAPL200529C00325000	2020-04-09 19:44:51	325.0	0.42	0.29	0.48	
22	AAPL200529C00330000	2020-04-09 19:42:12	330.0	0.30	0.20	0.36	
23	AAPL200529C00335000	2020-04-09 16:31:55	335.0	0.21	0.21	0.30	

	change	percentChange	volume	openInterest	impliedVolatility	inTheMoney	\
0	43.00	Infinity	25	NaN	0.569096	True	
1	33.28	Infinity	4	NaN	0.528081	True	
2	25.49	Infinity	3	NaN	0.490178	True	
3	23.62	Infinity	5	NaN	0.462957	True	
4	21.80	Infinity	10	NaN	0.452947	True	
5	16.62	Infinity	2	NaN	0.418219	True	
6	15.24	Infinity	63	NaN	0.412207	True	
7	13.80	Infinity	9	NaN	0.404669	False	
8	13.10	Infinity	56	NaN	0.398077	False	
9	11.45	Infinity	28	NaN	0.391119	False	
10	10.50	Infinity	5	NaN	0.386420	False	
11	9.05	Infinity	14	NaN	0.369330	False	
12	7.75	Infinity	2	NaN	0.374396	False	
13	6.03	Infinity	1	NaN	0.363105	False	
14	5.37	Infinity	2	NaN	0.354438	False	
15	3.90	Infinity	4	NaN	0.345954	False	
16	2.72	Infinity	31	NaN	0.337714	False	
17	2.03	Infinity	7	NaN	0.328010	False	
18	1.28	Infinity	6	NaN	0.318244	False	
19	0.95	Infinity	16	NaN	0.302375	False	
20	0.68	Infinity	1	NaN	0.298347	False	
21	0.42	Infinity	10	NaN	0.298835	False	
22	0.30	Infinity	1	NaN	0.302253	False	
23	0.21	Infinity	1	NaN	0.310554	False	

```
   contractSize currency
0       REGULAR      USD
1       REGULAR      USD
2       REGULAR      USD
3       REGULAR      USD
4       REGULAR      USD
5       REGULAR      USD
6       REGULAR      USD
7       REGULAR      USD
8       REGULAR      USD
9       REGULAR      USD
10      REGULAR      USD
11      REGULAR      USD
12      REGULAR      USD
13      REGULAR      USD
14      REGULAR      USD
15      REGULAR      USD
16      REGULAR      USD
17      REGULAR      USD
18      REGULAR      USD
19      REGULAR      USD
20      REGULAR      USD
21      REGULAR      USD
22      REGULAR      USD
23      REGULAR      USD
```

[6]:
```
symbol = 'AAPL'
start = '2019-12-01'
end = '2020-04-02'
```

[7]:
```
df = yf.download(symbol,start,end)
```

```
[**********************100%**********************]  1 of 1 completed
```

[8]:
```
df.head()
```

[8]:
```
              Adj Close       Close        High         Low        Open  \
Date
2019-12-02  263.534546  264.160004  268.250000  263.450012  267.269989
2019-12-03  258.835724  259.450012  259.529999  256.290009  258.309998
2019-12-04  261.120270  261.739990  263.309998  260.679993  261.070007
2019-12-05  264.951172  265.579987  265.890015  262.730011  263.790009
2019-12-06  270.069031  270.709991  271.000000  267.299988  267.480011

               Volume
Date
```

```
2019-12-02  23621800
2019-12-03  28607600
2019-12-04  16795400
2019-12-05  18606100
2019-12-06  26518900
```

[9]:
```
df.tail()
```

[9]:
```
             Adj Close       Close        High         Low        Open  \
Date
2020-03-26  258.440002  258.440002  258.679993  246.360001  246.520004
2020-03-27  247.740005  247.740005  255.869995  247.050003  252.750000
2020-03-30  254.809998  254.809998  255.520004  249.399994  250.740005
2020-03-31  254.289993  254.289993  262.489990  252.000000  255.600006
2020-04-01  240.910004  240.910004  248.720001  239.130005  246.500000

              Volume
Date
2020-03-26  63021800
2020-03-27  51054200
2020-03-30  41994100
2020-03-31  49250500
2020-04-01  44054600
```

[10]:
```
returns = df['Adj Close'].pct_change().dropna()
```

[11]:
```
from datetime import datetime
from dateutil import relativedelta

d1 = datetime.strptime(start, "%Y-%m-%d")
d2 = datetime.strptime('2020-05-28', "%Y-%m-%d")
delta = relativedelta.relativedelta(d2,d1)
print('How many years of investing?')
print('%s years' % delta.years)
```

```
How many years of investing?
0 years
```

[12]:
```
maturity_days = (df.index[-1] - df.index[0]).days
print('%s days' % maturity_days)
```

```
121 days
```

[13]:
```
S0 = df['Adj Close'][-1]
K = dfo_exp.calls['strike'][1]
r = 0.1
sigma = returns.std()
```

```
T = maturity_days/252
```

[14]:
```
print("S0\tCurrent Stock Price:", S0)
print("K\tStrike Price:", K)
print("r\tContinuously compounded risk-free rate:", r)
print("sigma\tVolatility of the stock price per year:", sigma)
print("T\tTime to maturity in trading years:", T)
```

```
S0      Current Stock Price: 240.91000366210938
K       Strike Price: 242.5
r       Continuously compounded risk-free rate: 0.1
sigma   Volatility of the stock price per year: 0.0369388726875486
T       Time to maturity in trading years: 0.4801587301587302
```

[15]:
```
def d1(S0, K, r, sigma, T):
    d1 = (np.log(S0/K) + (r + sigma**2 / 2) * T)/(sigma * np.sqrt(T))
    return d1
```

[16]:
```
def d2(S0, K, r, sigma, T):
    d2 = (np.log(S0 / K) + (r - sigma**2 / 2) * T) / (sigma * np.sqrt(T))
    return d2
```

[17]:
```
def BlackScholesCall(S0, K, r, sigma, T):
    BSC = S0 * ss.norm.cdf(d1(S0, K, r, sigma, T)) - K * np.exp(-r * T) * ss.
 ↪norm.cdf(d2(S0, K, r, sigma, T))
    return BSC
```

[18]:
```
def BlackScholesPut(S0, K, r, sigma, T):
    BSP = K * np.exp(-r * T) * ss.norm.cdf(-d2(S0, K, r, sigma, T)) - S0 * ss.
 ↪norm.cdf(-d1(S0, K, r, sigma, T))
    return BSP
```

[19]:
```
Call_BS = BlackScholesCall(S0, K, r, sigma, T)
Call_BS
```

[19]: 9.912965733098048