# Exotic Option Pricing using Monte Carlo Simulation

HackerRank

Imagine you're a quantitative analyst on an Equity Volatility trading desk at a major investment bank. Your client, a large hedge fund, has requested price quotes for custom European basket options with knock-out features. The basket is created on three hypothetical assets: DummyTechCorp (DTC), DummyFinCorp(DFC) and DummyEnergyCorp(DEC).

You are now tasked with building this pricing model and providing prices for all the options. Your submission will be rated based on the aggregate accuracy of the submitted prices.

**Background:**

*Vanilla Options*

An option is a financial contract that gives the buyer the right, but not the obligation, to buy or sell an asset at a specified price (called the strike price) on or before a specified date (called the maturity). Options are widely used for hedging, speculation, and portfolio management. We concern ourselves only with 'European Options' in this challenge. European options can only be exercised on the specified maturity data and not before.

Generally, there are two types of European vanilla options.

Call Options:

- A call option gives the holder the right to buy the underlying stock at the strike price K on the maturity date.
- The payoff of a call option is given by max(S(t) – K,0) where S(t) is the price of underlying stock on maturity date.

Put Options:

- A put option gives the holder the right to sell the underlying asset at the strike price K.
- The payoff of a put option is given by max(K - S(t),0) where S(t) is the price of underlying stock on maturity date.

*Basket Options*

Basket options are comparable to the vanilla options but their payoff depends on the weighted average price of a collection of assets (the "basket"). They offer investors a way to gain exposure to a diversified portfolio with a single transaction. However, pricing these options accurately presents a significant challenge and require more sophisticated techniques to account for the correlations between the assets in the basket with complex real-world diffusions

*Knock-Out Call & Put Options*

Options can be made more investor friendly by adding a knock-out feature which gives the holder the right to buy(sell) the underlying asset at a specified strike price ( K ), but only if the price of the underlying asset does not breach a predefined barrier level ( B ) during the life of the option. If the barrier is breached, the option becomes void (or "knocks out"), and the holder receives no payoff

The payoff of the knock-out call option is:

- max(S(T) − K,0) when S(t) < B for all t ≤ T
- 0 if S(t) >= B at any time t ≤ T

The payoff of the knock-out put option is:

- max(K - S(T),0) when S(t) < B for all t ≤ T
- 0 if S(t) >= B at any time t ≤ T

**Model Assumptions:**

Your desk applies the following assumptions in pricing Exotic Equity Options

Each stock's price evolves over time in a way that is influenced by randomness, similar to how stock prices fluctuate in real life. This randomness is modeled using a Diffusion Process with a drift component and a random shock component which also accounts for volatility of the stock.

The spot price for each stock evolves according to the following stochastic differential equation ( SDE ) under the risk-neutral probability measure. The drift for all stocks is the same but each stock can have a different local volatility function under the risk-neutral probability measure:

```
dS(t)= r * S(t) * dt+ σ(S(t), t) * dW(t)
```

where, - r = Risk-free rate - S(t) = Stock price at time t - σ(S(t),t) = The local volatility function which depends on S(t) and t - dW(t) = Brownian motion increment for at time t

The stocks are correlated. This dependance is usually measured with their historical return correlation. The Brownian Motion shocks for all 3 stocks should account for this correlation provided.

Monte Carlo simulation based pricing models work by simulating a large number of possible price paths for the underlying assets based on their diffusion processes, evaluating the payoff on each path and then averaging the discounted payoffs of the option across all simulated paths.

σ(S(t),t) is calibrated to ensure that the Monte Carlo Pricing Pricing Model ( with stock correlations captured ) is able to match Vanilla European Options provided.

σ(S(t),t) is assumed to be piece-wise constant in time for each calibration option maturity date and piece-wise constant for each calibration option strike ‘K’ for any time t. The calibration options for this challenge use 5 strikes and 3 maturity dates for each stock. So, the local volatility function for each stock can be thought of as a 5 * 3 matrix.

Diffused Stock prices for all the assets in the basket are generated post calibration. These paths can be then used to obtain the Diffused Basket prices and evaluate the basked option prices.

**Market inputs:**

- All 3 stocks have a current market spot value of $100
- The risk-free rate to be used in 5%
- The pairwise correlations for 3 stocks is given

```
Stock1,Stock2,Correlation
DTC,DFC,0.75
DTC,DEC,0.5,
DFC,DEC,0.25
```

- To calibrate the Local Volatility model for each asset, you are given the prices of European call options (in $) for different strikes and maturities. Each row below corresponds to a different calibration option.

```
CalibIdx,Stock,Type,Strike ,Maturity,Price
1,DTC,Call,50,1y,52.44
2,DTC,Call,50,2y,54.77
3,DTC,Call,50,5y,61.23
4,DTC,Call,75,1y,28.97
5,DTC,Call,75,2y,33.04
6,DTC,Call,75,5y,43.47
7,DTC,Call,100,1y,10.45
8,DTC,Call,100,2y,16.13
9,DTC,Call,100,5y,29.14
10,DTC,Call,125,1y,2.32
11,DTC,Call,125,2y,6.54
12,DTC,Call,125,5y,18.82
13,DTC,Call,150,1y,0.36
14,DTC,Call,150,2y,2.34
15,DTC,Call,150,5y,11.89
16,DFC,Call,50,1y,52.45
17,DFC,Call,50,2y,54.9
18,DFC,Call,50,5y,61.87
19,DFC,Call,75,1y,29.11
20,DFC,Call,75,2y,33.34
21,DFC,Call,75,5y,43.99
22,DFC,Call,100,1y,10.45
23,DFC,Call,100,2y,16.13
24,DFC,Call,100,5y,29.14
25,DFC,Call,125,1y,2.8
26,DFC,Call,125,2y,7.39
27,DFC,Call,125,5y,20.15
28,DFC,Call,150,1y,1.26
29,DFC,Call,150,2y,4.94
30,DFC,Call,150,5y,17.46
31,DEC,Call,50,1y,52.44
32,DEC,Call,50,2y,54.8
33,DEC,Call,50,5y,61.42
34,DEC,Call,75,1y,29.08
35,DEC,Call,75,2y,33.28
36,DEC,Call,75,5y,43.88
37,DEC,Call,100,1y,10.45
38,DEC,Call,100,2y,16.13
39,DEC,Call,100,5y,29.14
40,DEC,Call,125,1y,1.96
41,DEC,Call,125,2y,5.87
42,DEC,Call,125,5y,17.74
43,DEC,Call,150,1y,0.16
44,DEC,Call,150,2y,1.49
45,DEC,Call,150,5y,9.7
```

**Evaluation Criteria:**

The client has provided the following basket options for you to price.Each row below corresponds to a different option you are supposed to price. Your responses will be rated based on aggregate accuracy across all options.

The Basket Underlier for all these options is the equal weighted portfolio of DTC, DFC and DEC stocks. Weights of each stock = 1/3 such that the Basket spot price = $100.

Knock out feature is up and out only.

```
Id,Asset,KnockOut,Maturity,Strike,Type
1,Basket,150,2y,50,Call
2,Basket,175,2y,50,Call
3,Basket,200,2y,50,Call
4,Basket,150,5y,50,Call
5,Basket,175,5y,50,Call
6,Basket,200,5y,50,Call
7,Basket,150,2y,100,Call
8,Basket,175,2y,100,Call
9,Basket,200,2y,100,Call
10,Basket,150,5y,100,Call
11,Basket,175,5y,100,Call
12,Basket,200,5y,100,Call
13,Basket,150,2y,125,Call
14,Basket,175,2y,125,Call
15,Basket,200,2y,125,Call
16,Basket,150,5y,125,Call
17,Basket,175,5y,125,Call
18,Basket,200,5y,125,Call
19,Basket,150,2y,75,Put
20,Basket,175,2y,75,Put
21,Basket,200,2y,75,Put
22,Basket,150,5y,75,Put
23,Basket,175,5y,75,Put
24,Basket,200,5y,75,Put
25,Basket,150,2y,100,Put
26,Basket,175,2y,100,Put
27,Basket,200,2y,100,Put
28,Basket,150,5y,100,Put
29,Basket,175,5y,100,Put
30,Basket,200,5y,100,Put
31,Basket,150,2y,125,Put
32,Basket,175,2y,125,Put
33,Basket,200,2y,125,Put
34,Basket,150,5y,125,Put
35,Basket,175,5y,125,Put
36,Basket,200,5y,125,Put
```

Key things you should consider to improve your submission accuracy are:

1. Ensure the Local Volatility parameterization is as mentioned above. Different Local Volatility parameterizations can lead to different answers and will impact your score.
2. Ensure pricing error on the calibration options is minimal. Pricing an exotic with poorly calibrated SDEs will lead to erroneous results.
3. Ensure Monte-Carlo simulation is converging by using sufficiently large Monte-Carlo Paths and fine discretization in time dimension

**Input Format**

Your can load the Basket Option details by reading the text provided above

```
import pandas as pd
import io

data = '''Id,Asset,KnockOut,Maturity,Strike,Type
DummyId1,Basket,150,2y,50,Call
DummyId2,Basket,175,2y,50,Put'''

data_input = io.StringIO(data)

data_df = pd.read_csv(data_input,sep = ',')
```

### Constraints

- Option Price should be non-negative

### Output Format

Your submission should print the result in the following format.

```
Id,Price
DummyId1,20
DummyId2,50
```

Even though you may use some cached data to optimize runtime, please provide an option in the code to regenerate the cached data for evaluation purposes.. Submissions which produce some result but incomplete code to generate the output will not be considered

### Sample Input 0

```
Id,Asset,KnockOut,Maturity,Strike,Type
1,Basket,150,2y,50,Call
2,Basket,175,2y,50,Put
```

### Sample Output 0

```
Id,Price
1,100
2,100
```

### Explanation 0

Check that Output Format is as expected