# 亞式選擇權訂價實作：American-style Asian Single-barrier Up-and-out Calls Pricing based on the CRR binomial tree

**R04922092** 鄭允中

這次的亞式選擇權定價實作，要在各種數學式的代換中對於細節的準確地掌握，著實花了我不少時間，不過藉由這次實作讓我對選擇權的訂價有更細部的認識 : )

感謝耕竹和聖軒不厭其煩的指導！

圖文不符，寫完這堆數字蟲要療癒一下XD

## 一、問題描述：American-style Asian single-barrier up-and-out calls Pricing

Write a program to price American-style Asian single-barrier up-and-out calls based on the CRR binomial tree.

The payoff of this call at expiration date is max(average - X, 0) if the running average never touches or penetrates the barrier and 0 if otherwise.

Note also that the call may be exercised at any time before the expiration date due to its American-style characteristic.

Inputs:

- S (stock price at time 0),
- X (strike price),
- H (barrier, which is higher than S),
- t (maturity in years),
- s (%) (annual volatility),
- r (%) (continuously compounded annual interest rate),
- n (number of periods), and
- k (number of states per node).

For example, when

*S = 100, X = 80, H = 130, t = 1 (years), s = 30%, r = 10%, n = 100, and k = 300,*

the price is about 25.6562.

## 二、執行方式

只有一個.cpp檔，使用

```
g++ asian_pricing.cpp -o asian_pricing.o
```

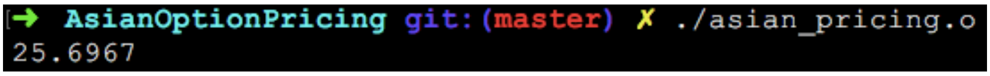
進行編譯，並直接執行

測資會讀取 test.txt ， 預設測資如下

<u>test.txt</u>

```
100
80
130
1
0.3
0.1
100
300
```

即會得到 American-style Asian single-barrier up-and-out calls Price



```
[→ AsianOptionPricing git:(master) ✗ ./asian_pricing.o
25.6967
```

三、設計與實作

### 1. Binomial Tree 與 Black-Scholes Model

下面是 Binomial Tree 與 Black-Scholes Model 參數間的對應關係

- Black-Scholes formula needs 5 parameters : S, X, σ, τ, and r.
- Binomial tree algorithms take 6 inputs : S, X, u, d, r^, and n.

The connections are :

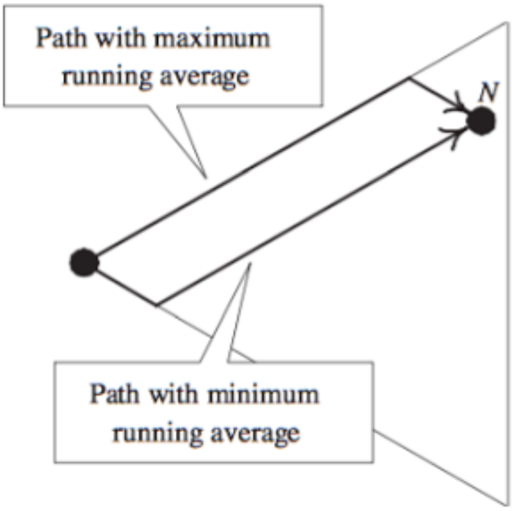
$$\begin{aligned} u &= e^{\sigma\sqrt{\tau/n}}, \\ d &= e^{-\sigma\sqrt{\tau/n}}, \\ \hat{r} &= r\tau/n. \end{aligned}$$

and pseudo probability is :

$$p \equiv (e^{r\tau/n} - d)/(u - d).$$

### 2. Create Running Average Tree with Interpolation

為了求亞式選擇權裡的標的物平均價格，在 binomal tree 中的每個節點存 k 個 state (測資為300)，每一個 state 用該節點可能產生的最大值和最小值去做內插。


Path with maximum running average
N
Path with minimum running average

$$A_m(j,i) \equiv \left(\frac{k-m}{k}\right) A_{\min}(j,i) + \left(\frac{m}{k}\right) A_{\max}(j,i)$$

$$m = 0, 1, \ldots, k.$$

而最大值(Amax)、最小值(Amin) 為下面兩個價格路徑和公式除以 j+1個離散時間。

Amax :

$$S_0(1 + \overbrace{u + u^2 + \cdots + u^{j-i} + u^{j-i}d + \cdots + u^{j-i}d^i}^{j})$$
$$= S_0 \frac{1-u^{j-i+1}}{1-u} + S_0 u^{j-i} d \frac{1-d^i}{1-d}.$$

Amin :

$$S_0(1 + \overbrace{d + d^2 + \cdots + d^i + d^i u + \cdots + d^i u^{j-i}}^{j})$$
$$= S_0 \frac{1-d^{i+1}}{1-d} + S_0 d^i u \frac{1-u^{j-i}}{1-u}.$$

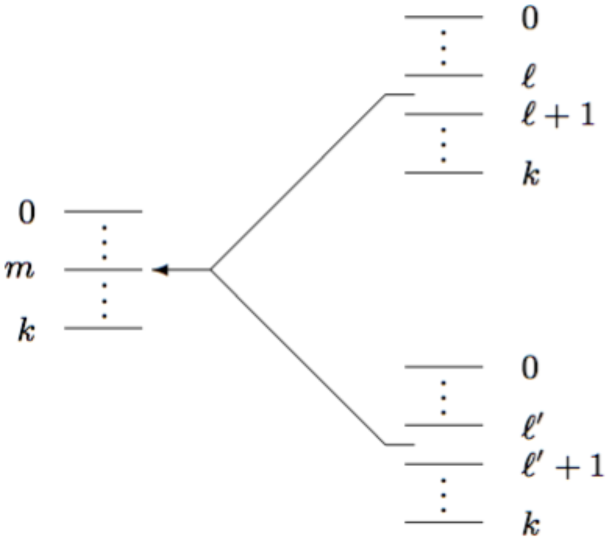
實作上，開一個三維陣列去儲存這些內插的資訊：

```
double ***asianMTree = Create3DArray(node_num, node_num, k+1);

for(int j=0; j<node_num; j++){
    for(int i=0; i<=j; i++){
        for(int m=0; m<=k; m++){
            asianMTree[j][i][m] = InterpoStates(j, i, m);
        }
    }
}
```

### 3. Pure Asian Call Pricing

再生成一棵 Price Tree，用 Backward induction 從葉子回推到根節點，每一個state都要做，共四個步驟求 Cu、Cd (用剛剛的Running Average Tree的資訊來內插)，再由 Cu、Cd 求出該點的價格 :



下面以Cu為例，Cd同理：

a. 求Au (Running Average)

算往上走一格後，整個路徑上的標的物價格平均

$$A_{\mathrm{u}} \equiv \frac{(j+1)\,a + S_0 u^{j+1-i} d^i}{j+2}.$$

[感覺] 加權平均：像是全班30人平均身高是160 加一個身高170的人 全班身高平均變多少的感覺

b. 求足碼 l

足碼 l 由下列公式求出：

$$\ell = \left\lfloor \frac{A_{\mathrm{u}} - A_{\min}(j,i)}{[A_{\max}(j,i) - A_{\min}(j,i)]/k} \right\rfloor$$

[小心] 對於 l 的例外處理

要找到內插的足碼 l 必須符合下面的條件

HTML generated using hilite.me 否則直接給定邊界值

```
if(Au > AvgMax(j+1,i) || (AvgMax(j+1,i)-Au)< 0.0001 ){
    Cu = CTree[j+1][i][k];
}
else if(Au < AvgMin(j+1,i)){
    Cu = CTree[j+1][i][0];
}
else{
 //Interpolation
}
```

c. 求對應的內插比例 x 後，求Cu

$$A_{\mathrm{u}} = xA_{\ell}(j+1,i) + (1-x)\,A_{\ell+1}(j+1,i), \quad 0 < x \leq 1.$$

$$C_{\mathrm{u}} \equiv xC_{\ell}(j+1,i) + (1-x)\,C_{\ell+1}(j+1,i).$$

[注意] 在最後一個時間點(leave node)的Cu不用內插，也沒辦法內插，由最單純的 call price 給出

$$\begin{aligned} C_{\mathrm{u}} &= \max(A_{\mathrm{u}} - X, 0), \\ C_{\mathrm{d}} &= \max(A_{\mathrm{d}} - X, 0). \end{aligned}$$

最後，在求出Cu、Cd後代入風險中立的機率折現後即為該state的價格

$$[\,pC_{\mathrm{u}} + (1-p)\,C_{\mathrm{d}}\,]\,e^{-r\Delta t}.$$

重複此步驟 (所有state都要算) 直到Price Tree樹根，樹根的價格即為亞式選擇權價格，在該節點所有 state 的值會相同！

## 4. American-style and Single-barrier Issue

如果該點標的物價格碰到barrier (H) 本選擇權直接無效；並因為是美式選擇權，隨時可以履約，因此判斷如下：

```
if(asianMTree[j][i][m] > H){
    CTree[j][i][m] = 0;
}
else if( americanC > asianC ){
    CTree[j][i][m] = americanC;
}
else{
    CTree[j][i][m] = asianC;
}
```

CTree[0][0][0] 即為 American-style Asian single-barrier up-and-out calls Price.

## References

Y.-D. Lyuu - Principles of Financial Computing
http://www.csie.ntu.edu.tw/~lyuu/finance1.html

wiki - Binomial options pricing model
https://en.wikipedia.org/wiki/Binomial_options_pricing_model