EXCEL VBA 進階班

Lecture 8

SECTION 12. 物件導向介面

衍生性金融商品損益計算 (2)

物件導向介面

- 現在我們面臨到新的問題，報酬計算不只會有一般 Plain Vanilla 選擇權之方式，還可能會有數位選擇權（Digital Option）。
- 以下為數位選擇權簡介（目前市場上很少存在 Asset-or-nothing）。

物件導向介面

- Cash-or-nothing
- 買權：
  - 當到期時，資產價格 > 履約價格

    ➔ 買權持有者得到一筆契約上約定好的報酬
- 賣權：
  - 當到期時，資產價格 < 履約價格

    ➔ 賣權持有者得到一筆契約上約定好的報酬

物件導向介面

- Asset-or-nothing
- 買權：
  - 當到期時，資產價格 > 履約價格

    ➔ 買權持有者得到與該資產相同之金額
- 賣權：
  - 當到期時，資產價格 < 履約價格

    ➔ 賣權持有者得到與賣出該資產相同之金額

物件導向介面

- 我們想要一個計算 Cash-or-nothing 的歐式買權可以利用之前的 PlainVanilla 物件進行修改，首先我們須先新增一個新屬性，固定之報酬，將它命名為 fixedPayoff，由於其不能有負的，因此與 strike 相同 :

```
' Cash-or-nothing 固定報酬
Private fixedPayoff_ As Double

Public Property Let fixedPayoff(ByVal value As Double)

If value > 0 Then
    fixedPayoff_ = value
Else
    Err.Raise Number:=10002, Source:="fixedPayoff", Description:="固定報酬須大於零"
End If

End Property

Public Property Get fixedPayoff() As Double
fixedPayoff = fixedPayoff_
End Property
```

物件導向介面

- 除此之外更改報酬計算如下：

```
Public Function calculatePayoff(spot As TimeSeries) As TimeSeries
Dim nDays As Integer: nDays = UBound(spot.dates)
ReDim payoff(nDays) As Double
Dim i As Integer

For i = 1 To nDays
    If spot.dates(i) = expiryDate Then
        payoff(i) = IIf(putCallType_ * (spot.values(i) - strike_) > 0, buySellType_ * fixedPayoff_, 0)
        Exit For
    End If
Next i

calculatePayoff = fromArray(spot.dates, payoff, "payoff")
End Function
```

物件導向介面

- 建立代替建構子的函數：

```
Public Function newCashOrNothing(strike As Double, _
                                 expiryDate As Date, _
                                 putCallType As OptionType, _
                                 fixedPayoff As Double, _
                                 buySellType As BuyOrSell) As CashOrNothing

Dim newOption As New CashOrNothing
newOption.strike = strike
newOption.expiryDate = expiryDate
newOption.putCallType = putCallType
newOption.buySellType = buySellType
newOption.fixedPayoff = fixedPayoff
Set newCashOrNothing = newOption
End Function
```

物件導向介面

- 因此我們也可以利用 CashOrNothing 物件計算數位選擇權損益了：

```
Dim instrument As CashOrNothing
Set instrument = newCashOrNothing(100, DateValue("2013/12/14"), callOption, 50, sell)

Dim payoffSeries As TimeSeries
payoffSeries = instrument.calculatePayoff(spotPrice)
```

物件導向介面

- 但此時假如我持有的是投資組合，我會希望用一個函數來計算投資組合的加總損益。
- 注意 : 物件一樣可以擁有陣列 :

```vb
Dim portfolio(3) As PlainVanilla
Dim j As Integer
Dim payoffSeries As TimeSeries
For i = 1 To 3
    Set portfolio(i) = newPlainVanilla(100, dates(i), callOption, buy)
Next i

'計算投資組合損益加總
ReDim payoffSum(3) As Double
For i = 1 To 3
    payoffSeries = portfolio(i).calculatePayoff(spotPrice)

    For j = 1 To 3
        payoffSum(j) = payoffSum(j) + payoffSeries.values(j)
    Next j
Next i
payoffSeries.values = payoffSum
Range("A1:B4").value = toVariantArray(payoffSeries)
```

	A	B
1	Date	payoff
2	2013/12/13	2
3	2013/12/14	3.058321
4	2013/12/15	3.858457

物件導向介面

- 我們可以把上面的程式碼寫成一函數，方便重複利用。
- 撰寫一個 calculatePortfolioPayoff 函數，可以快速計算陣列內每個選擇權之損益並加總。
- 可以看到在 portfolio 部份我們假設是一維陣列，並沒有使用 Variant 變數進行 For Each，原因在於使用 Variant 時，該方法的引數不能是 user-defined type 或是自另類別。

```
Public Function calculatePortfolioPayoff(spotPrice As TimeSeries, portfolio() As PlainVanilla) As TimeSeries

Dim nDays As Integer: nDays = UBound(spotPrice.dates)
Dim nInstrument As Integer: nInstrument = UBound(portfolio)
Dim j As Integer
Dim i As Integer
ReDim payoffSum(nDays) As Double
Dim payoffSeries As TimeSeries

For i = 1 To nInstrument
    payoffSeries = portfolio(i).calculatePayoff(spotPrice)
    For j = 1 To nDays
        payoffSum(j) = payoffSum(j) + payoffSeries.values(j)
    Next j
Next i

calculatePortfolioPayoff.dates = spotPrice.dates
calculatePortfolioPayoff.values = payoffSum
calculatePortfolioPayoff.varName = "portfolio payoff"

End Function
```

物件導向介面

- 這時候有問題發生了 : 投資組合可能不只有 PlainVanilla，也可能會存在 CashOrNothing。
- 但我的陣列是使用 PlainVanilla，因此容不下 CashOrNothing。
- 在傳統的物件導向中，我們可以另 CashOrNothing 繼承於 PlainVanilla，因此 PlainVanilla 中容得下 CashOrNothing。
- 但 VBA 中並沒有繼承，該如何處理？

物件導向介面

- 一開始我們會想到使用 Variant 陣列或 Collection。
- 但是這樣必須放棄 TimeSeries 並且降低了程式的可讀性與提高出錯率。
- 有沒有其他的方法呢？

物件導向介面

- 回想一下不論是 CashOrNothing 或是 PlainVanilla，他們倆者皆為一個衍生性金融商品，其實他們是有很多共通性的。
- 最簡單的共通性是，給定現貨價格時間序列，衍生性金融商品皆可計算得出一個損益時間序列。
- 因此我們可以設立一個介面（Interface），名為 DerivativeSecurity。

物件導向介面

- 介面是用來描述不同類別的共同行為。
- CashOrNothing 與 PlainVanilla 在根本上是不同類別，但他們在計算報酬上有著相似的動作，都是數入一個時間序列，回傳一個時間序列。
- 任何類別只要滿足介面的定義，都可以算為其中一份子，介面本身並不需要了解有多少類別符合該規範。

物件導向介面

- 首先，新增一個類別稱為 DerivativeSecurity

物件導向介面

- 接著定義好 DerivativeSecurity 介面下的使用方法，calculatePayoff，只要定義好我們需要輸入一個時間序列，並回傳一個時間序列即可。

```
Option Explicit
Option Base 1
Public Function calculatePayoff(spot As TimeSeries) As TimeSeries: End Function
```

物件導向介面

- 接著在 PlainVanilla 與 CashOrNothing 類別中，我們需要定義其應用 DerivativeSecurity 介面，因此在開頭處加上 Implements DerivativeSecurity :

```
Option Explicit
Option Base 1
Implements DerivativeSecurity

Private strike_ As Double

' putCallType_ = 1 (call)
' putCallType_ = -1 (put)
Private putCallType_ As Integer
Private buySellType_ As Integer
Public expiryDate As Date
```

物件導向介面

- 接著我們需要讓 DerivativeSecurity 介面知道，該類別下之 calculatePayoff 是要讓其應用的，因此將名稱更改如下：

```vb
Public Function DerivativeSecurity_calculatePayoff(spot As TimeSeries) As TimeSeries
Dim nDays As Integer: nDays = UBound(spot.dates)
ReDim payoff(nDays) As Double
Dim i As Integer

For i = 1 To nDays
    If spot.dates(i) = expiryDate Then
        payoff(i) = IIf(putCallType_ * (spot.values(i) - strike_) > 0, buySellType_ * fixedPayoff_, 0)
        Exit For
    End If
Next i

DerivativeSecurity_calculatePayoff = fromArray(spot.dates, payoff, "payoff")
End Function
```

物件導向介面

- 至此，我們將陣列改為 DerivativeSecurity 之陣列即可在其中包含 PlainVanilla 與 CashOrNothing。
- 但到目前為止出現一個問題，我們會發現 DerivativeSecurity 只能使用 calculatePayoff 方法，如果我們要讀寫其他資訊則毫無辦法，為此，我們必須讓 DerivativeSecurity 擁有所有的選全該有的特性，例如履約價的讀寫：

物件導向介面

```
Option Explicit
Option Base 1
Public Function calculatePayoff(spot As TimeSeries) As TimeSeries: End Function
Public Property Let strike(ByVal value As Double): End Property
Public Property Get strike() As Double: End Property
```

物件導向介面

- 而 strike 之 Property 名稱也須跟著更改。

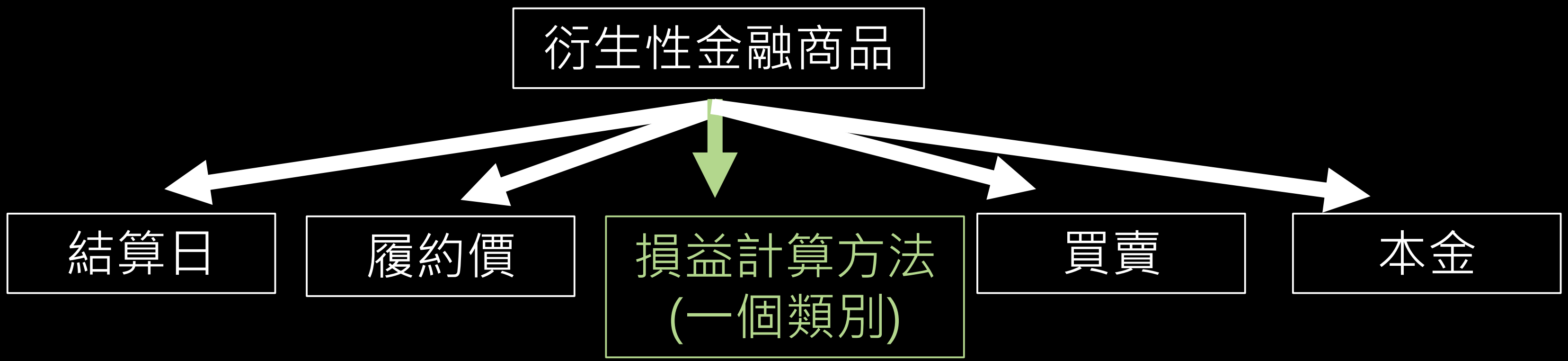
```
Public Property Let DerivativeSecurity_strike(ByVal value As Double)

If value > 0 Then
    strike_ = value
Else
    Err.Raise Number:=10001, Source:="strike", Description:="履約價須大於零"
End If

End Property

Public Property Get DerivativeSecurity_strike() As Double
strike = strike_
End Property
```

物件導向介面

- 介面牽一髮動全身，只要他一進行更改，所有應用該介面的類別必須全部更改至與其相符。
- CashOrNothing 與 PlainVanilla 本身就重複大量相同的特性，將相同的東西寫兩次已經完全喪失物件導向的優勢與精神。
- 有沒有辦法可以改善呢？

物件導向介面

- 我們可以再把衍生性金融商品細分，產品本身包含：



物件導向介面

- 因此我們可以把 Payoff 單獨分支出來處理。
- 首先，將 DerivativeSecurity 改為一個實際使用的類別，當中包含了各種方法 :
  - strike : 履約價，需大於零
  - expiryDate : 日期
  - buySellType : 買賣
  - amount : 本金或買賣單位數，需大於零

物件導向介面

- 想一下如何利用介面解決此問題呢？

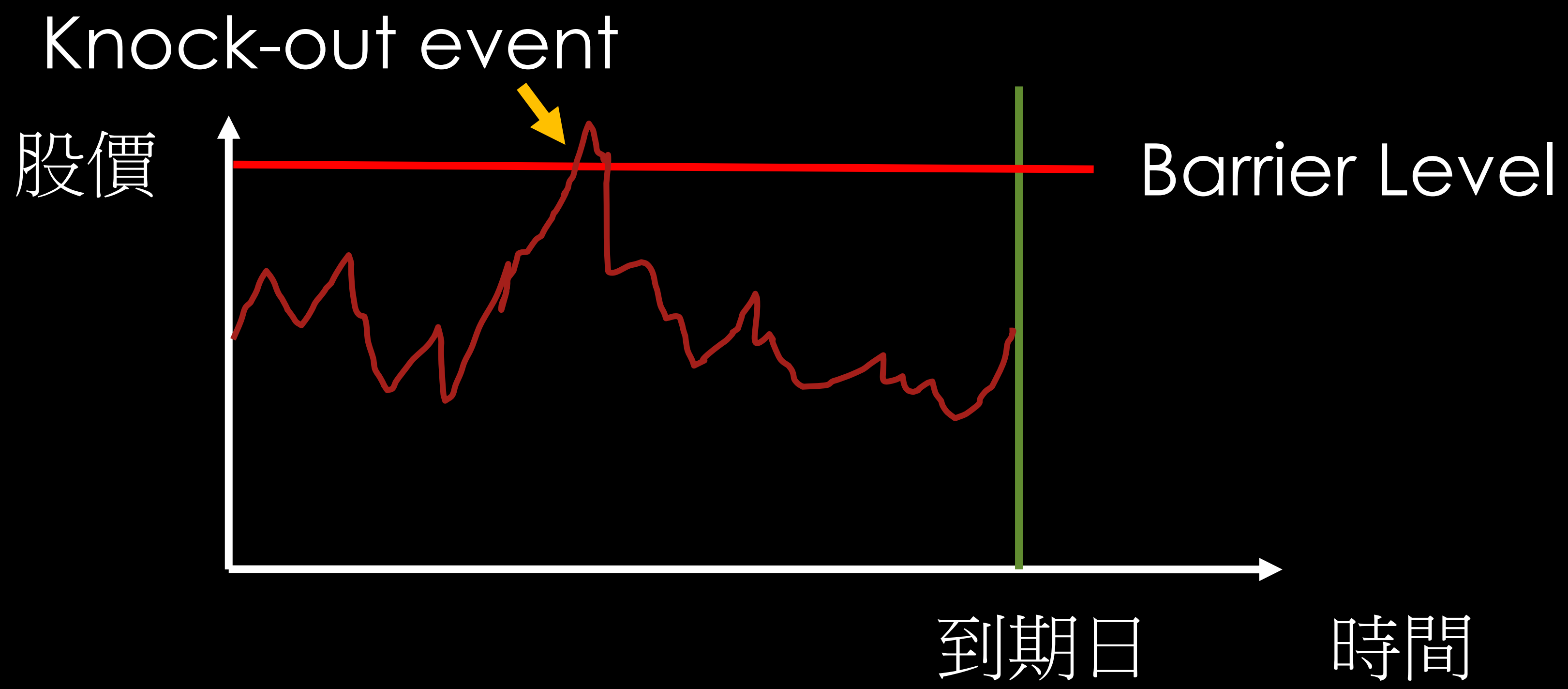
SECTION 13. 蒙地卡羅法簡介

蒙地卡羅法簡介：
為何需使用蒙地卡羅法

- 簡單的歐式買權與賣權有Black-Scholes可以快速作評價
- 但許多更加複雜的金融商品並沒有辦法找到類似的方程式
- 例如： 1. 美式選擇權

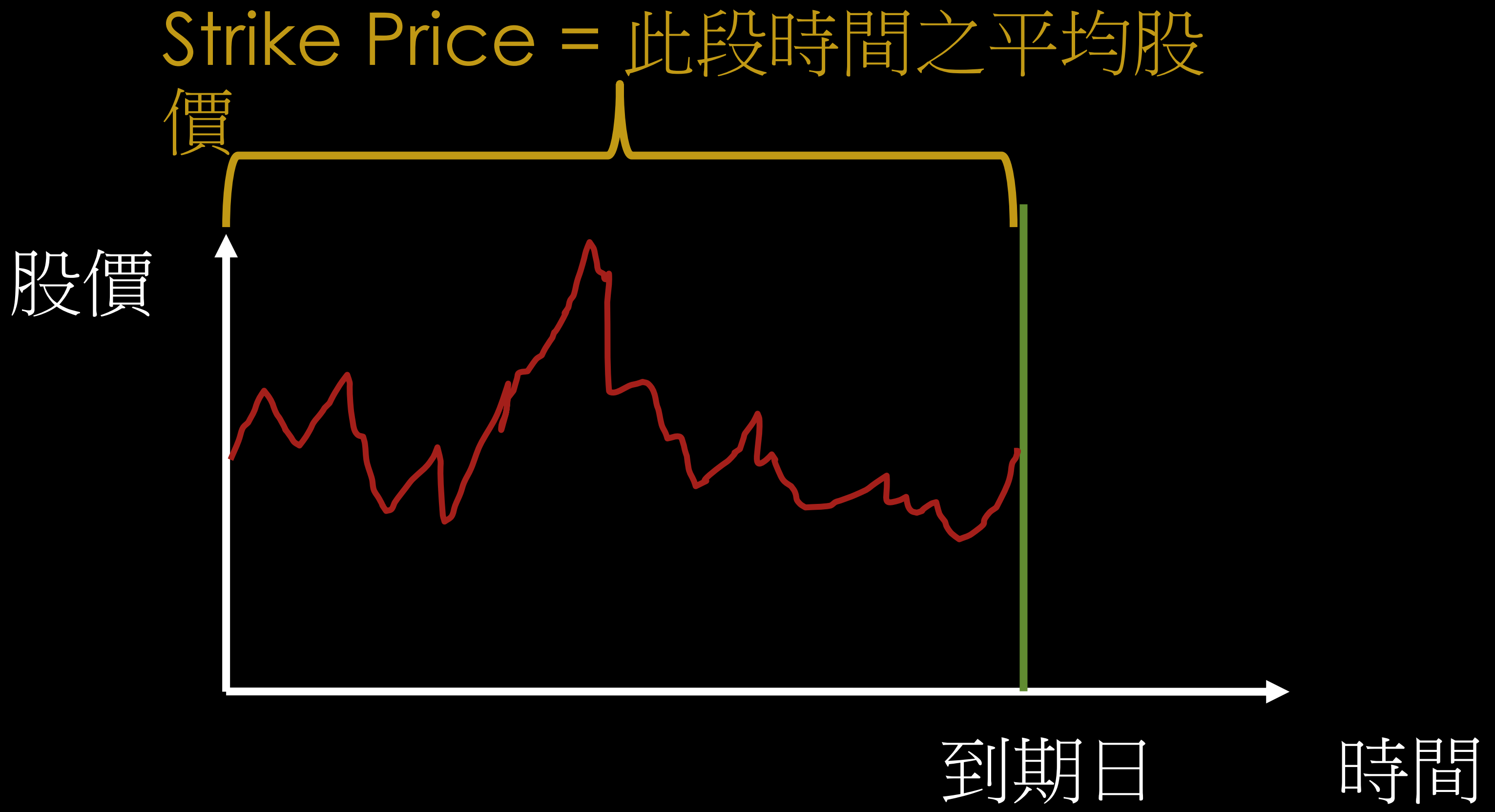
蒙地卡羅法簡介：為何需使用蒙地卡羅法

- 2. 障礙選擇權 (up-and-out):



蒙地卡羅法簡介：為何需使用蒙地卡羅法

- 3. 亞式選擇權



蒙地卡羅法簡介：
為何需使用蒙地卡羅法

- 4. Target Redemption Forward (TRF) :
- 投資人與銀行對賭匯率走勢，每月 (或每兩周)結算一次
- 賭對獲得 (本金 * 匯差)
- 賭錯損失 (本金*匯差*Ratio)
- 當損失或獲利超過一定上限則提早出場。

蒙地卡羅法簡介：
為何需使用蒙地卡羅法

- 例如. 一個賭日幣貶值之TRF
- USD/JPY之履約價格 = 110.00
- 12期，每期之間隔一個月
- 獲利出場條件 : 10點
- Notional Amount : USD 10,000
- Ratio : 2

蒙地卡羅法簡介：為何需使用蒙地卡羅法

- 到期時假如spot rate $<$ 110.00

  ➔ 契約持有人賣 JPY (2*10,000*110.00)，

  買 USD 2*10,000

- 到期時假如spot rate $\geq$ 110.00

  ➔契約持有人賣 JPY (10,000*110.00)，

  買 USD 10,000

蒙地卡羅法簡介：
為何需使用蒙地卡羅法

- 對於上述會與標的資產每日價格都有關之商品，要計算期望報酬必須知道整條未來的價格路徑。
- 只要模擬夠多條，各條路徑之報酬平均會趨近於理論上之期望報酬。

蒙地卡羅法簡介：步驟介紹

- Black-Scholes中，假設資產價格的變動率符合以下隨機過程：

- $\frac{dS(t)}{S(t)} = \mu\, dt + \sigma\, dW(t)$

- μ = 資產的平均年報酬率

- σ = 資產的年波動度，代表股票受隨機衝擊之影響之大小。

- 整理成log的樣式如下：

- $d \log S(t) = \left(\mu - \frac{1}{2}\sigma^2\right) dt + \sigma dW(t)$ ←資產瞬間報酬率

蒙地卡羅法簡介：步驟介紹

- 以下先說明如何模擬單一條股票路徑：
- 假設$\mu = 0.08$，$\sigma = 0.15$
- 一般常把時間推進設為一天一步，年化後的時間$\Delta t = \frac{1}{365}$ 或$\frac{1}{360}$。

蒙地卡羅法簡介：步驟介紹

- 標的資產起始價格為 $S(0) = 100$
- 經過一天之後變動為：

$$S\left(\frac{1}{365}\right) = S(0)e^{\left(\mu-\frac{\sigma^2}{2}\right)\times\frac{1}{365}+\sigma[W\left(\frac{1}{365}\right)-W(0)]}$$

$$= 100 \times e^{\left(0.08-\frac{(0.15)^2}{2}\right)\times\frac{1}{365}+(0.15)[W\left(\frac{1}{365}\right)-W(0)]}$$

蒙地卡羅法簡介：步驟介紹

- 同理，兩天後之變動為：

$$S\left(\frac{2}{365}\right)=S\left(\frac{1}{365}\right)e^{\left(\mu-\frac{\sigma^2}{2}\right)\times\frac{1}{365}+\sigma[W\left(\frac{2}{365}\right)-W(\frac{1}{365})]}=$$

$$100\times e^{\left(0.08-\frac{(0.15)^2}{2}\right)\times\frac{1}{365}+(0.15)[W\left(\frac{1}{365}\right)-W(0)]}\times e^{\left(0.08-\frac{(0.15)^2}{2}\right)\times\frac{1}{365}+(0.15)[W\left(\frac{2}{365}\right)-W(\frac{1}{365})]}$$

蒙地卡羅法簡介：步驟介紹

- 如需一年後價格模擬，則重複此動作365次，即

$$S\left(\frac{365}{365}\right)=S(0)e^{\left(\mu-\frac{\sigma^2}{2}\right)\left(\frac{365}{365}\right)+\sigma\sum_{i=1}^{365}[W\left(\frac{i}{365}\right)-W(\frac{i-1}{365})]}$$

蒙地卡羅法簡介：範例

- 現在假設要模擬三天後股價
- 步驟一：設定好現在時間點之股價，假設為100。

時間	t = 0，現在	$t = \frac{1}{365}$，一天	$t = \frac{2}{365}$，兩天	$t = \frac{3}{365}$，三天
資產價格	100			

蒙地卡羅法簡介 :範例

- 步驟二: 讓電腦在第一天隨機產生一個0~1之間的亂數，當作一機率。
  - 假設產出之亂數為 0.70

時間	t = 0，現在	$t=\frac{1}{365}$，一天	$t=\frac{2}{365}$，兩天	$t=\frac{3}{365}$，三天
資產價格	100	0.70		

- 步驟三: 帶入標準常態分配反函數，算此機率下累積機率函數之值。
  - $F^{-1}(0.70)=0.524401$

時間	t = 0，現在	$t=\frac{1}{365}$，一天	$t=\frac{2}{365}$，兩天	$t=\frac{3}{365}$，三天
資產價格	100	0.524401		

蒙地卡羅法簡介 :範例

- 步驟四:步驟三結果 × $\sqrt{一天的時間長度}$ = $[W(1/365) - W(0)]$之模擬
  - $W(1/365) - W(0) = 0.524401 \times \sqrt{1/365} = 0.027448$
  - 整個資產一天報酬率的模擬為：
  - $\exp[\left(0.08 - \frac{0.15^2}{2}\right) \times \frac{1}{365} + 0.15*0.027448\ ] = e^{0.006278} = 1.06298$

時間	t = 0，現在	$t = \frac{1}{365}$，一天	$t = \frac{2}{365}$，兩天	$t = \frac{3}{365}$，三天
資產價格	100	1.06298		

蒙地卡羅法簡介 :範例

- 步驟五:將一開始之價格乘上此報酬率，即為第一天價格之模擬
  - $100 \times 1.06298 = 106.298$

時間	t = 0，現在	$t = \frac{1}{365}$，一天	$t = \frac{2}{365}$，兩天	$t = \frac{3}{365}$，三天
資產價格	100	1.06298		

- 重複上述結果
- 步驟一與步驟二在R中有函數可以一次完成，
- R : rnorm()

蒙地卡羅法簡介 :模型設定

- 但在開始做模擬前，有以下問題要處理 :
  - 資產之報酬趨勢 $\mu = ?$
  - 資產之波動度 $\sigma = ?$
  - 如果要做更精準之評價，無風險利率r要如何設定?

蒙地卡羅法簡介 :風險中立評價法

- Black-Scholes所計算出之價格隱含無套利機會，亦即任何報酬與該選擇權完全相同之動態投資組合，期初的建構成本與選擇權之價格相符。
- 假設券商放空一單位選擇權，可以靠放空或持有標的資產，以及用無風險利率介到，建構出一個固定報酬的投資組合。

蒙地卡羅法簡介 :風險中立評價法

- 但以真實世界之機率測度$W(t)$做蒙地卡羅法所評價出來之價格是有套利機會的，因此根據Harrison與Krep在1979提出之方法，將機率測度由真實世界之機率測度轉為風險中立機率測度$\widetilde{W}(t)$後所評價出之價格也為無套利價格。

- $\widetilde{W}(t) = W(t) + \int_0^t \frac{\mu - r}{\sigma} dt$

- r = 無風險利率

蒙地卡羅法簡介 :風險中立評價法

- 經過測度轉換後之資產報酬率動態過程如下：
- $\frac{dS(t)}{S(t)} = r\,dt + \sigma\,d\widetilde{W}(t)$，$\widetilde{W}(t)$也一樣是一個Brownian Motion
- 整理成log的樣式如下：
- $d\log S(t) = \left(r - \frac{1}{2}\sigma^2\right)dt + \sigma\,d\widetilde{W}(t)$