VaR Simulation for American Options

- Write a macro that estimates VaR for American calls, puts, and straddles by
  - Simulating multiple paths for a geometric brownian motion asset price process
  - Estimating the value of the option at every step along each path
  - Estimating the VaR of the option for each horizon ending at every step

American Options VaR.xlsm - Microsoft Excel non-commercial use

	A	B	C	D	E
1	Stock Price (S0)	50		**Estimated Option Value**	
2	Risk Free Rate (rf)	4%			
3	Dividend Yield (q)	0%			
4	Volatility (sigma)	40%		**Horizon**	**VaR**
5	Time to Maturity (T)	1		(in number of steps)	
6	Number of Steps (m)	252			
7	Number of Paths (n)	1000			
8					
9	Strike Price (K)	52			
10	Option Type	Call			
11					
12	VaR Confidence Level	40%			

Run

American Options VaR.xlsm - Microsoft Excel non-commercial use

E1 f_x 8.25276504249614

	A	B	C	D	E
1	Stock Price (S0)	50		**Estimated Option Value**	8.25277
2	Risk Free Rate (rf)	4%			
3	Dividend Yield (q)	0%			
4	Volatility (sigma)	40%		**Horizon**	**VaR**
5	Time to Maturity (T)	1		(in number of steps)	
6	Number of Steps (m)	252		1	5.10924
7	Number of Paths (n)	1000		2	5.03408
8				3	5.03357
9	Strike Price (K)	52		4	5.03306
10	Option Type	Call		5	5.03255
11				6	5.10674
12	VaR Confidence Level	40%		7	5.10624
13				8	5.10574
14				9	5.10524
15				10	5.10474
16				11	5.10424
17				12	5.10374
18				13	5.10324
19				14	5.10274
20				15	5.10224
21				16	5.10174
22				17	5.10124
23				18	5.10074
24				19	5.10024
25				20	5.09974
26				21	5.09924
27				22	5.17517

Run

- At step j + 1 of a path, price S changes as follows:

$$\boldsymbol{S_{j+1} = S_j + (r - q)S_j\Delta t + \sigma S_j\sqrt{\Delta t}Z_j}$$

  - r is the risk-free rate
  - q is the asset dividend yield
  - Δt is the time between 2 steps
  - σ is the annual asset volatility
  - Z_j is a random value from a standard normal distribution
  - (Z_j = WorksheetFunction.NormInv(Rnd(), 0, 1)))

Asset Price Dynamics

- For each path, compute the option value $\boldsymbol{V_m}$ at terminal step m: $\mathbf{max}(\boldsymbol{K} - \boldsymbol{S_m}, \mathbf{0})$
- Work back from step m to step m-1 to step m-2 to…to step 1 to get $\boldsymbol{V_1}$ for each path
- At each step i-1, the option value, $\boldsymbol{V_{i-1}}$, is the highest of the exercise value, $\mathbf{max}(\boldsymbol{K} - \boldsymbol{S_{i-1}}, \mathbf{0})$ and the continuation value, $\boldsymbol{E}[\boldsymbol{e^{-r\Delta t}V_i} \mid \boldsymbol{S_{i-1}}]$

Put Valuation

- Set $\boldsymbol{V_0}$ to $\boldsymbol{e^{-r\Delta t}V_1}$ for each path
- Average $\boldsymbol{V_0}$ over all paths to estimate the option value

Put Valuation

- At intermediate step i-1, estimate the regression coefficients in $\boldsymbol{e^{-r\Delta t} V_i = a + b\, S_{i-1} + c\, S_{i-1}{}^2}$ by using the observations from in-the-money paths $(K - S_{i-1} > 0)$

- For each of the n paths, plug $\boldsymbol{S_{i-1}}$ into the regression formula to come up with the continuation value, $\boldsymbol{E[e^{-r\Delta t} V_i \mid S_{i-1}]}$

Continuation Value

- If the continuation value $\boldsymbol{E}[\boldsymbol{e}^{-\boldsymbol{r}\Delta\boldsymbol{t}}\boldsymbol{V}_{\boldsymbol{i}} \mid \boldsymbol{S}_{\boldsymbol{i}-\mathbf{1}}]$ is greater than the exercise value $\mathbf{max}(\boldsymbol{K} - \boldsymbol{S}_{\boldsymbol{i}-\mathbf{1}}, \mathbf{0})$, set the option value $\boldsymbol{V}_{\boldsymbol{i}-\mathbf{1}}$ to $\boldsymbol{e}^{-\boldsymbol{r}\Delta\boldsymbol{t}}\boldsymbol{V}_{\boldsymbol{i}}$ in order to reduce standard errors

Continuation Value

- For horizon i, the absolute VaR at the (1-α) confidence level is -1 times (the α-th percentile of the distribution of $\boldsymbol{V_i}$ across the n paths, minus the initial average value $\boldsymbol{V_0}$)

VaR Estimation

Data Structure

```
Option Explicit


Type BS_PathType
    S0 As Double         'initial asset price
    q As Double          'dividend yield
    sigma As Double      'volatility
    rf As Double         'risk free rate
    t As Double          'period length
    dt As Double         'time step
    m As Long            'no of steps
    n As Long            'no of paths
End Type
```

```
Dim A As BS_PathType

A.S0 = Cells(1, 2)
```

OLS Regression

```
Public Function OLS(x As Variant, y As Variant) As Variant
    Dim Beta As Variant
    Dim i As Integer

    With Application
        Beta = .MMult(.MInverse(.MMult(.Transpose(x), x)), .MMult(.Transpose(x), y))
    End With

    ReDim res(0 To UBound(Beta) - 1) As Double
    For i = 0 To UBound(res)
        res(i) = Beta(i + 1, 1)
    Next i
    OLS = res
End Function
```