


Anexo 1– Códigos em VBA utilizados

Cálculo do preço justo de uma opção pelo modelo de Black e Scholes Generalizado

Function GBlackScholes(CallPutFlag As String, S As Double, X As Double, T As Double, r As Double, b As Double, v As Double) As Double
    Dim d1 As Double, d2 As Double
    d1 = (Math.Log(S / X) + (b + v ^ 2 / 2) * T) / (v * Math.Sqrt(T))
    d2 = d1 - v * Math.Sqrt(T)
    If CallPutFlag = "c" Then
    ElseIf CallPutFlag = "p" Then
    End If
End Function

Cálculo da volatilidade implícita pelo modelo de Black e Scholes Generalizado

Function GImpliedVolatilityBisection(CallPutFlag As String, S As Double, X As Double, T As Double, r As Double, b As Double, cm As Double) As Variant
    Dim vLow As Double, vHigh As Double, vi As Double
    Dim cLow As Double, cHigh As Double, epsilon As Double
    Dim counter As Integer
    vLow = 0.005
    vHigh = 4
    epsilon = 0.00000001

cLow = GBlackScholes(CallPutFlag, S, X, T, r, b, vLow)
cHigh = GBlackScholes(CallPutFlag, S, X, T, r, b, vHigh)
counter = 0
vi = vLow + (cm - cLow) * (vHigh - vLow) / (cHigh - cLow)
While Abs(cm - GBlackScholes(CallPutFlag, S, X, T, r, b, vi)) > epsilon
    counter = counter + 1
    If counter = 100 Then
        GImpliedVolatilityBisection = "NA"
        Exit Function
    End If
    If GBlackScholes(CallPutFlag, S, X, T, r, b, vi) < cm Then
        vLow = vi
    Else
        vHigh = vi
    End If
    cLow = GBlackScholes(CallPutFlag, S, X, T, r, b, vLow)
cHigh = GBlackScholes(CallPutFlag, S, X, T, r, b, vHigh)
    vi = vLow + (cm - cLow) * (vHigh - vLow) / (cHigh - cLow)
Wend
GImpliedVolatilityBisection = vi
End Function

Function SkewKurtCorradoSu(CallPutFlag As String, S As Double, X As Double, T As Double, r As Double, b As Double, v As Double, Skew As Double, Kurt As Double) As Double

    Dim Q3 As Double, Q4 As Double
    Dim d1 As Double, d2 As Double
    Dim CallValue As Double
    d1 = (Math.Log(S / X) + (b + v ^ 2 / 2) * T) / (v * Math.Sqr(T))
d2 = d1 - v * Math.Sqr(T)
\[
Q_4 = \frac{1}{24} S v \sqrt{T} \left( (d_1^2 - 1 - 3v \sqrt{T}d_2) \left( \frac{1}{2\pi} \exp\left(-\frac{d_1^2}{2}\right) \right) + v^3 T^{1.5} \right.
\]
\[
\left. \text{WorksheetFunction.NormSDist}(d_1) \right)
\]
\[
Q_3 = \frac{1}{6} S v \sqrt{T} \left( (2v \sqrt{T} - d_1) \left( \frac{1}{2\pi} \exp\left(-\frac{d_1^2}{2}\right) \right) + v^2 T \text{WorksheetFunction.NormSDist}(d_1) \right)
\]
\[
\text{CallValue} = \text{GBlackScholes}("c", S, X, T, r, b, v) + \text{Skew} * Q_3 + (\text{Kurt} - 3) * Q_4
\]
\[
\text{If CallPutFlag = "c" Then}
\]
\[
\text{SkewKurtCorradoSu} = \text{CallValue}
\]
\[
\text{Else}
\]
\[
\text{SkewKurtCorradoSu} = \text{CallValue} - S \text{Math.Exp}((b - r) * T) + X * \text{Math.Exp}(-r * T)
\]
\[
\text{End If}
\]
\[
\text{End Function}
\]

**Cálculo da volatilidade implícita pelo modelo de Corrado e Su**

Function CSImpliedVolatilityBisection(CallPutFlag As String, S As Double, X As Double, T As Double, r As Double, b As Double, Skew As Double, Kurt As Double, cm As Double) As Variant

Dim vLow As Double, vHigh As Double, vi As Double
Dim cLow As Double, cHigh As Double, epsilon As Double
Dim counter As Integer

vLow = 0.005
vHigh = 4
epsilon = 0.001
cLow = SkewKurtCorradoSu(CallPutFlag, S, X, T, r, b, vLow, Skew, Kurt)
cHigh = SkewKurtCorradoSu(CallPutFlag, S, X, T, r, b, vHigh, Skew, Kurt)
counter = 0

\[
\text{vi} = \text{vLow} + (\text{cm} - \text{cLow}) * (\text{vHigh} - \text{vLow}) / (\text{cHigh} - \text{cLow})
\]

While Abs(cm - SkewKurtCorradoSu(CallPutFlag, S, X, T, r, b, vi, Skew, Kurt)) > epsilon
\[
\text{counter} = \text{counter} + 1
\]
If counter = 1000 Then
    CSImpliedVolatilityBisection = "NA"
    Exit Function
End If

If SkewKurtCorradoSu(CallPutFlag, S, X, T, r, b, vi, Skew, Kurt) < cm
Then
    vLow = vi
Else
    vHigh = vi
End If

cLow = SkewKurtCorradoSu(CallPutFlag, S, X, T, r, b, vLow, Skew, Kurt)
cHigh = SkewKurtCorradoSu(CallPutFlag, S, X, T, r, b, vHigh, Skew, Kurt)

vi = vLow + (cm - cLow) * (vHigh - vLow) / (cHigh - cLow)

Wend

CSImpliedVolatilityBisection = vi

End Function