LAMPIRAN 1. KODE PROGRAM FORM WEIBUL

Option Explicit
Private WithEvents MyWeibull As Weib
Private Const CHART_DATA_DIV As Integer = 50

Private Sub cmdCost_Click()
    If Not IsNumeric(txtc1.Text) Then Exit Sub
    If Not IsNumeric(txtc2.Text) Then Exit Sub

    Dim c1 As Double, c2 As Double, dt As Double, d_T As Double
    Dim x As Integer
    x = MsgBox("Apakah Anda yakin dengan biaya yang Anda masukkan", vbYesNo + vbInformation, "Biaya")
    If x = vbYes Then
        c1 = CDbl(txtc1.Text)
        c2 = CDbl(txtc2.Text)
        dt = 0.01
        d_T = 0.1

        cmdCost.MousePointer = vbHourglass
        lblMinCost.Caption = "Wait..."
        lblPeriod.Caption = "Wait..."
        DoEvents
        MyWeibull.Cost c1, c2, dt, d_T
        cmdCost.MousePointer = vbArrow
        txtty2.Visible = True
        txttx2.Visible = True
    End If
End Sub

Private Sub cmdWeib_Click()
    Dim DataArr() As Double
    Dim DataCnt As Integer
    Dim x As Integer
    x = MsgBox("Apakah Anda yakin dengan data yang anda masukan", vbYesNo + vbInformation, "Perhatian")
    If x = vbYes Then
        txtty1.Visible = True
        txttx1.Visible = True
        cmdWeib.MousePointer = vbHourglass
        lblParam(0).Caption = "Wait..."
        lblParam(1).Caption = "Wait..."
        lblParam(2).Caption = "Wait..."
        DoEvents
        Dim idx As Integer
        DataCnt = lstEntry.ListCount
        If DataCnt = 0 Then Exit Sub
        ReDim Preserve DataArr(DataCnt - 1)
        For idx = 0 To DataCnt - 1 Step 1

        Next idx
        cmdWeib.MousePointer = vbArrow
    End If
End Sub
DataArr(idx) = lstEntry.List(idx)
Next idx
MyWeibull.Crunch DataArr()
cmdWeibull.MousePointer = vbArrow
End If
End Sub

Private Sub Form_Load()
    Set MyWeibull = New Weib
    InitChart chartWeibull(0)
    InitChart chartWeibull(1)
End Sub

Private Sub InitChart(ByRef chart As MSChart)
    With chart
        .Plot.UniformAxis = False
        .Plot.Axis(VtChAxisIdx).ValueScale.Auto = False
        .Plot.Axis(VtChAxisIdx).ValueScale.MajorDivision = 10
        .Plot.Axis(VtChAxisIdx).Labels(1).Format = "0.0"
        .Plot.Axis(VtChAxisIdxY).ValueScale.Auto = False
        .Plot.Axis(VtChAxisIdxY).ValueScale.Minimum = 0
        .Plot.Axis(VtChAxisIdxY).ValueScale.MajorDivision = 10
        .Plot.Axis(VtChAxisIdxY).Labels(1).Format = "0.00"
        .RowCount = CHART_DATA DIV 1
    End With
End Sub

Private Sub Form_QueryUnload(Cancel As Integer, UnloadMode As Integer)
    If MsgBox("Are you sure want to exit?", vbYesNo + vbQuestion, "PERINGATAN") = vbNo Then
        Cancel = 1
    Else
        Unload Me
    End If
End Sub

Private Sub Form_Unload(Cancel As Integer)
    Set MyWeibull = Nothing
End Sub

Private Sub lstEntry_KeyDown(KeyCode As Integer, Shift As Integer)
    If KeyCode <> 46 Then Exit Sub
    lstEntry.RemoveItem lstEntry.ListIndex
End Sub

Private Sub MyWeibull_CostReady(cnt As Integer, T() As Double; F_T() As Double, MaxF_t As Double, MinF_t As Double, BestPeriod As Double)
    Dim i As Integer
    With MyWeibull(1)
        .RowCount = cnt
        .Plot.Axis(VtChAxisIdxY).ValueScale.Maximum = MaxF_t
    End With
End Sub
.Plot.Axis(VtChAxisIdY).ValueScale.Minimum = MinF_t

.Plot.Axis(VtChAxisIdX).ValueScale.Maximum = T(cnt - 1)
.Plot.Axis(VtChAxisIdX).ValueScale.Minimum = T(0)

For i = 1 To cnt Step 1
  .Row = 1
  .Column = 1
  .Data = T(i - 1)
  .Column = 2
  .Data = F_T(i - 1)
  Next i
End With

lblMinCost.Caption = "Min Cost: " & Format(MinF_t, "0.00")
lblPeriod.Caption = "Best Period: " & Format(BestPeriod, "0.00")
End Sub

Private Sub MyWeibull_WeibullReady(n As Integer, sortarr() As Double, a As Double, b As Double, c As Double)
  Dim x As Double, i As Integer
  Dim max As Double, min As Double, div As Double
  Dim y As Double, max_y As Double

  lblParam(0).Caption = "a: " & Format(a, "0.00")
lblParam(1).Caption = "b: " & Format(b, "0.00")
lblParam(2).Caption = "c: " & Format(c, "0.00")

  max = sortarr(n - 1)
  min = c
  div = (max - min) / CHART_DATA_DIV
  max_y = 3.402823E+38

  With chartWeibull(0)
    .Plot.Axis(VtChAxisIdX).ValueScale.Maximum = max
    .Plot.Axis(VtChAxisIdX).ValueScale.Minimum = min
  For i = 1 To .RowCount Step 1
    x = min + (i - 1) * div
    .Row = i
    .Column = 1
    .Data = x
    .Column = 2
    y = MyWeibull.GetWeib(x)
    If y > max_y Then max_y = y
    .Data = y
  Next i
  .Plot.Axis(VtChAxisIdY).ValueScale.Maximum = max_y
  End With
  End Sub

Private Sub txtEntry_GotFocus()
  txtEntrySelStart = 0
  txtEntrySelLength = Len(txtEntry.Text)
End Sub
Private Sub txtEntry_KeyPress(KeyAscii As Integer)
    Dim Entry As String
    If KeyAscii <> 13 Then Exit Sub
    Entry = txtEntry.Text
    If IsNumeric(Entry) Then
        lstEntry.AddItem txtEntry.Text
    Else
        MsgBox "Harus Bilangan!"
    End If
    txtEntry_GotFocus
End Sub
LAMPIRAN 2. KODE PROGRAM CLASS WEIB

Option Explicit

Event Weibull1Ready(ByVal n As Integer, ByVal sortarr() As Double, ByVal a As Double, ByVal b As Double, ByVal c As Double)
Event CostReady(ByVal cnt As Integer, ByVal T() As Double, ByVal F_T() As Double, ByVal MaxF_t As Double, ByVal MinF_t As Double, ByVal BestPeriod As Double)
Private best_a As Double
Private best_b As Double
Private best_c As Double
Private sortarr() As Double

Public Sub Crunch(ByVal DataArr() As Double)
'---Sort:---
Sort DataArr(), sortarr()

'---Rank Probability:---
Dim p() As Double, w() As Double
RankProb sortarr(), p(), w()

'---Seek for best parameters:---
Dim resol As Double
resol = 0.01
BestParam sortarr(), w(), resol
End Sub

Public Function GetWeib(ByVal x As Double) As Double
Dim temp As Double
If x <= best_c Then
    temp = 0#
Else
    temp = (best_a / best_b)
    temp = temp * (((x - best_c) / best_b) ^ (best_a - 1))
    temp = temp * Exp(-1 * (((x - best_c) / best_b) ^ best_a))
End If
GetWeib = temp
End Function

Public Sub Cost(ByVal cl As Double, ByVal c2 As Double, ByVal dt As Double, ByVal d_T As Double)
Dim i As Integer
Dim n As Integer, cnt As Integer
Dim T() As Double, F_T() As Double
Dim MaxF_t As Double, MinF_t As Double
Dim BestPeriod As Double
MaxF_t = -1.79769313486231E+308
MinF_t = 1.79769313486231E+308
n = UBound(sortarr()) + 1
cnt = (sortarr(n - 1) - 0) / d_T
ReDim Preserve T(cnt - 1)
ReDim Preserve F_T(cnt - 1)
For i = 1 To cnt Step 1
    T(i - 1) = 0 + (i * d_T)
    F_T(i - 1) = GetCost(c1, c2, dt, T(i - 1))
    If F_T(i - 1) > MaxF_t Then MaxF_t = F_T(i - 1)
    If F_T(i - 1) < MinF_t Then
        MinF_t = F_T(i - 1)
        BestPeriod = T(i - 1)
    End If
Next i
RaiseEvent CostReady(cnt, T(), F_T(), MaxF_t, MinF_t, BestPeriod)
End Sub

Private Sub Sort(_
    ByRef InArr() As Double, _
    ByRef OutArr() As Double)

    Dim InSize As Integer, InIdx As Integer
    Dim InVal As Double
    Dim OutSize As Integer, OutIdx As Integer
    Dim SftIdx As Integer

    InSize = UBound(InArr())
    ReDim Preserve OutArr(0)
    OutArr(0) = InArr(0)
    OutSize = 0
    For InIdx = 1 To InSize Step 1
        InVal = InArr(InIdx)
        OutSize = OutSize + 1
        ReDim Preserve OutArr(OutSize)
        OutArr(OutSize) = InVal
        For OutIdx = 0 To OutSize - 1 Step 1
            If InVal < OutArr(OutIdx) Then
                For SftIdx = OutSize To OutIdx + 1 Step -1
                    OutArr(SftIdx) = OutArr(SftIdx - 1)
                Next SftIdx
                OutArr(OutIdx) = InVal
            Exit For
        End If
    Next OutIdx
Next InIdx
End Sub

Private Sub RankProb(_
    ByRef sortarr() As Double, _
    ByRef p() As Double, _
ByRef w() As Double

Dim n As Integer, i As Integer, prob As Double

n = UBound(sortarr()) + 1
ReDim Preserve p(n - 1)
ReDim Preserve w(n - 1)
For i = 1 To n Step 1
    prob = (i - 0.5) / n
    p(i - 1) = prob
    w(i - 1) = Log(1 / (1 - prob))
Next i
End Sub

Private Sub BestParam(
    ByRef sortarr() As Double, _
    ByRef w() As Double, _
    ByVal resol As Double)

Dim a As Double, b As Double, c As Double
Dim StepUp As Integer
Dim Step As Integer
Dim old_sse As Double
Dim new_sse As Double
Dim d As Double, e As Double
Dim f As Double, g As Double
Dim n As Integer
Dim est_x() As Double
Dim best_sse As Double

n = UBound(sortarr()) + 1
a = 1
Step = 0
old_sse = 1.79769313486231E+308
best_sse = 1.79769313486231E+308
ReDim Preserve est_x(n - 1)
Do Until StepUp > 5
    a = a + (Step * resol)
    d = get_d(n, sortarr(), w(), a)
    e = get_e(n, sortarr())
    f = get_f(n, w(), a)
    g = get_g(n, w(), a)
    b = ((n * d) - (e * f)) / ((n * g) - (f * f))
    c = ((c * g) - (d * f)) / ((n * g) - (f * f))
GetEstimatedX n, a, b, c, w(), est_x()
new_sse = Getsse(n, est_x(), sortarr())
If new_sse < best_sse Then
    best_sse = new_sse
    best_a = a
End Sub
best_b = b
best_c = c
End If
If new_sse > old_sse Then
    StepUp = StepUp + 1
ElseIf new_sse < old_sse Then
    StepUp = 0
End If
old_sse = new_sse
Step = Step + 1
Loop
RaiseEvent WeibullReady(n, sortarr(), best_a, best_b, best_c)
End Sub

Private Function get_d( _
    ByVal n As Integer, _
    ByRef sortarr() As Double, _
    ByRef w() As Double, _
    ByVal a As Double) As Double

    Dim i As Integer
    Dim x As Double
    Dim temp As Double
    For i = 1 To n Step 1
        x = sortarr(i - 1)
        temp = temp + (x * (w(i - 1) ^ (1 / a)))
    Next i
    get_d = temp
End Function

Private Function get_e( _
    ByVal n As Integer, _
    ByRef sortarr() As Double) As Double

    Dim i As Integer
    Dim temp As Double
    For i = 1 To n Step 1
        temp = temp + sortarr(i - 1)
    Next i
    get_e = temp
End Function

Private Function get_f( _
    ByVal n As Integer, _
    ByRef w() As Double, _
    ByVal a As Double) As Double

    Dim i As Integer
    Dim temp As Double
    For i = 1 To n Step 1
        temp = temp + (w(i - 1) ^ (1 / a))
    Next i
get_f = temp
End Function

Private Function get_g(_
    ByVal n As Integer, _
    ByRef w() As Double, _
    ByVal a As Double) As Double
    Dim i As Integer
    Dim temp As Double
    For i = 1 To n Step 1
        temp = temp + (w(i - 1) ^ (2 / a))
    Next i
    get_g = temp
End Function

Private Sub GetEstimatedX(_
    ByVal n As Integer, _
    ByVal a As Double, _
    ByVal b As Double, _
    ByVal c As Double, _
    ByRef w() As Double, _
    ByRef est_x() As Double)
    Dim i As Integer
    For i = 0 To n - 1 Step 1
        est_x(i) = c + b * (w(i) ^ (1 / a))
    Next i
End Sub

Private Function GetSsse(_
    ByVal n As Integer, _
    ByVal est_x() As Double, _
    ByVal x() As Double) As Double
    Dim i As Integer, temp As Double
    For i = 0 To n - 1 Step 1
        temp = temp + ((est_x(i) - x(i)) ^ 2)
    Next i
    GetSsse = temp
End Function

Private Function GetCost(_
    ByVal c1 As Double, _
    ByVal c2 As Double, _
    ByVal dt As Double, _
    ByVal T As Double) As Double
    Dim integral As Double
    integral = GetIntegral(dt, T)
Dim FT As Double
FT = GetCummWeib(T)
GetCost = c1 + (c2 * FT)
GetCost = GetCost / (integral + (T * (1 - FT)))
End Function

Private Function GetIntegral(ByVal dt As Double, ByVal T As Double) As Double
Dim t_ As Double
Dim y1 As Double, y2 As Double, temp As Double
Dim Cumm As Double
For t_ = 0 To T Step dt
ten = 0#
y1 = GetWeib(t_)
y2 = GetWeib(t_ + dt)
If y1 < y2 Then
    temp = y1 * dt
Else
    temp = y2 * dt
End If
ten = temp + (0.5 * Abs(y2 - y1) * dt)
end = end + temp
Cumm = Cumm + temp
Next t_
GetIntegral = Cumm
End Function

Private Function GetCummWeib(ByVal T As Double) As Double
Dim temp As Double
If T <= best_c Then
    temp = 0#
Else
    temp = Exp(-1 * (((T - best_c) / best_b) ^ best_a))
end = end + temp
End If
GetCummWeib = temp
End Function