VBA Project

BONNEAU Nathan
CHARBONNIER Thibault

Summary

Subject

CHF/USD vs Gold/USD

Correlation, regression and statistics

Scraping Data CHF/USD

Preamble :   We will import the CHF/USD price data over the last 6 months from Yahoo Finance.

- We adapted the code seen during a CM to import data from yahoo finance:

- The code creates an XMLHTTP object and sends a request with the URL of the page. The query response is retrieved by the innerHTML object. (corresponds to the currency table)

- The first "for" loop extracts text from the cells in the first row of the table and places them in the corresponding cells in the first row of the active worksheet.

- Next, the code uses two loops to iterate through the remaining rows of the table (starting from the second row) and the cells in each row

```
url = "https://fr.finance.yahoo.com/quote/CHFUSD%3DX/history?p=CHFUSD%3DX" ''yahoo finance url which corresponds
  ''make a request
Set xmlHttp = CreateObject("MSXML2.XMLHTTP")
xmlHttp.Open "GET", url, False
xmlHttp.send
Set html = CreateObject("htmlfile")
html.body.innerHTML = xmlHttp.responseText ''receive the result of the request
Set table = html.getElementsByTagName("table")(0)
Set rows = table.getElementsByTagName("tr")
For i = 0 To rows(0).Cells.Length - 1
    Cells(1, i + 1) = rows(0).Cells(i).innerText ''extract text from HTML table cells
Next i
For i = 1 To rows.Length - 2
  For j = 0 To rows(i).Cells.Length - 1
    cellValue = rows(i).Cells(j).innerText
    If IsNumeric(cellValue) Then
      RealValue = CDbl(Replace(cellValue, ".", ",")) ''replaces the . in , so that excel understands
      Cells(i + 1, j + 1) = RealValue
      Cells(i + 1, j + 1).NumberFormat = "0.0000" ''rigth format
    Else
      Cells(i + 1, j + 1) = cellValue
    End If
  Next j
Next i
```

Scraping Data CHF/USD from Yahoo Finance

Date	Ouverture	Élevé	Faible	Clôture*	Cours de c	Volume
05 janv. 2024	1,1765	1,1826	1,1670	1,1765	1,1765	-
04 janv. 2024	1,1780	1,1796	1,1725	1,1780	1,1780	-
03 janv. 2024	1,1763	1,1783	1,1690	1,1763	1,1763	-
02 janv. 2024	1,1878	1,1881	1,1762	1,1878	1,1878	-
01 janv. 2024	1,1894	1,1894	1,1878	1,1894	1,1894	-
29 déc. 2023	1,1857	1,1963	1,1855	1,1857	1,1857	-
28 déc. 2023	1,1872	1,2005	1,1869	1,1872	1,1872	-
27 déc. 2023	1,1715	1,1888	1,1700	1,1715	1,1715	-
26 déc. 2023	1,1686	1,1709	1,1656	1,1686	1,1686	-
25 déc. 2023	1,1695	1,1702	1,1654	1,1695	1,1695	-
22 déc. 2023	1,1679	1,1743	1,1664	1,1679	1,1679	-
21 déc. 2023	1,1595	1,1678	1,1585	1,1595	1,1595	-
20 déc. 2023	1,1624	1,1627	1,1561	1,1624	1,1624	-
19 déc. 2023	1,1533	1,1637	1,1527	1,1533	1,1533	-
18 déc. 2023	1,1486	1,1538	1,1481	1,1486	1,1486	-
15 déc. 2023	1,1532	1,1562	1,1497	1,1532	1,1532	-
14 déc. 2023	1,1481	1,1589	1,1456	1,1481	1,1481	-
13 déc. 2023	1,1427	1,1431	1,1393	1,1427	1,1427	-
12 déc. 2023	1,1388	1,1457	1,1384	1,1388	1,1388	-
11 déc. 2023	1,1369	1,1383	1,1345	1,1369	1,1369	-
08 déc. 2023	1,1423	1,1440	1,1349	1,1423	1,1423	-

Plot CHF/USD

Preamble :   Now that we have extracted the data, it is interesting to display it graphically in an Excel window.

```
Set dateRange = Range("A2:A100")
Set priceRange = Range("F2:F100") ''define as a range

priceData = priceRange.Value

dates = dateRange.Value

Charts.Add
ActiveChart.ChartType = xlXYScatterSmoothNoMarkers ''creating a new
With ActiveChart.SeriesCollection.NewSeries
    .XValues = dates
    .Values = priceData
    .Name = "CHF/USD"
End With

ActiveChart.HasLegend = False
ActiveChart.Axes(xlCategory).HasTitle = True
ActiveChart.Axes(xlCategory).AxisTitle.Characters.Text = "Dates"
ActiveChart.Axes(xlValue).HasTitle = True
ActiveChart.Axes(xlValue).AxisTitle.Characters.Text = "Prix"
ActiveChart.Axes(xlCategory).CategoryType = xlTimeScale
ActiveChart.Axes(xlCategory).TickLabels.NumberFormat = "dd mmmm"
```

- We store date and price ranges. We use Range to specify cell ranges. We create a chart using the Add method of the Charts object. We use smooth dispersion (xlXYScatterSmoothNoMarkers).We define the abscissa (.XValues) and ordinate (.Values) and the name of the graph (.Name). With .Axes we define the different properties of our graph.

➔ We can now observe our graph on a new excel sheet
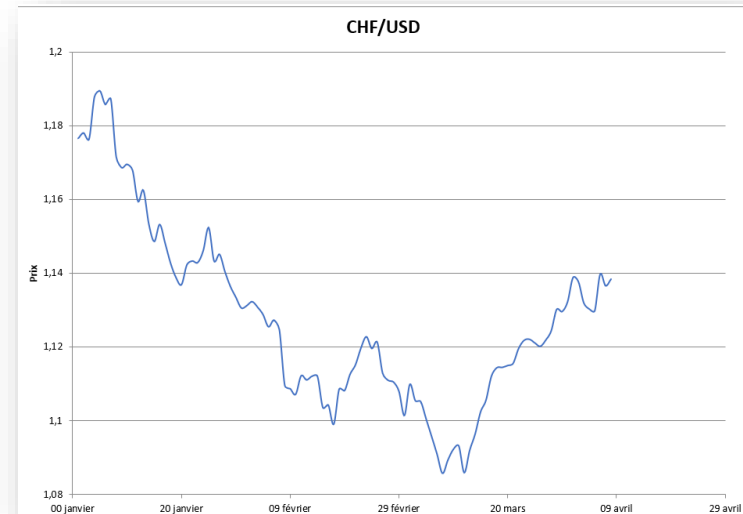
Statistics calculations

Preamble : Now to have a better overview of our data, we can calculate different statistics on VBA and check our calculations with Excel functions. (max, min, mean, median, standard deviation)

- Maximum & Minimum :

To calculate the maximum, simply initiate the value of max with our first data and thanks to a loop we replace the value of max when we find a larger value. Same logic for the minimum.

```
'''calculating the maximum of our data
maxVal = Range("F2").Value ''init the first value for the max
For Each i In dataRange
    If i.Value > maxVal Then ''if we have a value greater than the max, it becomes max
        maxVal = i.Value
    End If
Next i
```

Minimum : 1,0858
Maximum : 1,1894

```
''calculating the mean of our data
j = 0
For Each i In dataRange
    sum = sum + i.Value ''sum of the data and then divide by the number of data
    j = j + 1
Next i
mean = sum / j
```

Mean : 1,127929

- Mean :

To get the average, simply add all our data (sum) and divide it by the total number of values (j),

6

Statistics calculations

- Median :

Once dataValues are defined (data range). We use two loops (k and t)
to sort the values in ascending order using a bubble sort method. The dataValues are swapped as necessary to put them in the correct order.
Then we just need to take the value in the middle if it's odd. Otherwise if it is even we take the average of the two middle values.

```
dataValues = Range("F2:F101").Value
For k = 1 To UBound(dataValues, 1) ''bubble sorting carried out in a previous TD
    For t = k + 1 To UBound(dataValues, 1)
        If dataValues(t, 1) < dataValues(k, 1) Then
            tempVal = dataValues(k, 1)
            dataValues(k, 1) = dataValues(t, 1)
            dataValues(t, 1) = tempVal
        End If
    Next t
Next k

If UBound(dataValues, 1) Mod 2 = 0 Then ''Case where the length of our data is even
    medianVal = (dataValues(UBound(dataValues, 1) \ 2, 1) + dataValues(UBound(dataValues, 1) \ 2 + 1, 1)) / 2
Else ''case where the length is odd
    medianVal = dataValues((UBound(dataValues, 1) + 1) \ 2, 1)
End If
```

median : 1,12455

- Standard deviation :

We store in sum: the squared difference of the datarange values minus the mean.
The standard deviation is nothing more than the root of sum divided by the length of the datarange.

```
sum = 0
For Each i In dataRange
    sum = sum + (i.Value - mean) ^ 2 '' sum of values minus mean squared
Next i
stdDev = Sqr(sum / dataRange.rows.Count) '' divide by the length and take the square root
```

Standard Deviation: 2,45266479364792E-02

Checking statistics

- It is important to check if our calculations carried out on VBA are correct and for this we can use the Excel functions already implemented. For example: MAX, MIN, STDEV, MEDIAN, AVERAGE.

Checking statistics using the Excel function				
Maximum	Minimum	Standard deviation	Median	Mean
1,1894	1,0858	0,024650209	1,1246	1,1279

Scraping Data
GOLD USD

Preamble :   Now we will scrape GOLD/USD data from Yahoo Finance.

- We are using the CHF/USD code by changing the link to Gold/USD

- However, after importing we notice lines containing hyphens. We must remove them.

Scraping GOLD USD

Date	Ouverture	Élevé	Faible	Clôture*	Cours de c	Vo
05 janv. 2024	2044,5000	2048,1000	2042,4000	2042,4000	2042,4000	
04 janv. 2024	2041,6000	2044,5000	2038,0000	2042,3000	2042,3000	
03 janv. 2024	2034,2000	2044,0000	2034,2000	2034,2000	2034,2000	
02 janv. 2024	2063,5000	2073,7000	2057,1000	2064,4000	2064,4000	
01 janv. 2024	-	-	-	-	-	
29 déc. 2023	2068,0000	2068,9000	2062,1000	2062,4000	2062,4000	
28 déc. 2023	2081,6000	2087,3000	2066,5000	2073,9000	2073,9000	
27 déc. 2023	2067,3000	2081,9000	2064,8000	2081,9000	2081,9000	
26 déc. 2023	2060,0000	2060,8000	2054,2000	2058,2000	2058,2000	
25 déc. 2023	-	-	-	-	-	

- Our DeletingLines() code takes GOLDUSD as the active page and sees that it is the index of the last data in column F (last). Then we loop through the dataRange and with the If condition, we delete each time there is a hyphen in column F, we delete the row.

```
Sub DeletingLines()
    Dim sheet As Worksheet
    Dim last As Long
    Dim i As Variant
    Dim dataRange As Range

    Set sheet = ThisWorkbook.Sheets("GOLDUSD") ''I define my acti
    last = sheet.Cells(sheet.rows.Count, "F").End(xlUp).Row ''I r
    Set dataRange = sheet.Range("F2:F" & last) ''the datarange co

    For Each i In dataRange
        If i.Value = "-" Then
            i.EntireRow.Delete ''if a box of F contains - I delet
        End If
    Next i
End Sub
```

Plot and statistics
GOLD USD

Preamble : We will reuse the previous functions to display the curve and study the statistics related to GOLD USD.

- For the plot, simply change the incoming data and the title.

- For statistics it's the same principle!

- We also use Excel functions to confirm the correct functioning of our VBA code.
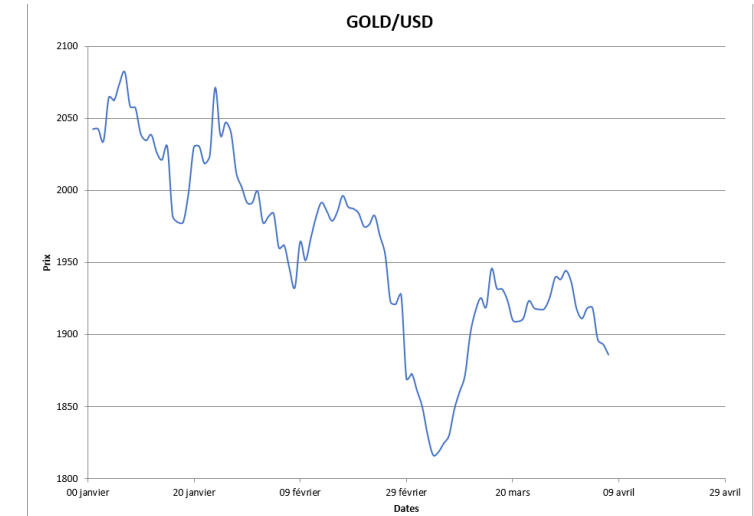
Minimum : 1816,6
Maximum : 2081,9
Mean : 1959,71734693878
median : 1963
Standard Deviation: 64,1631705434337

Checking statistics using the Excel function				
Maximum	Minimum	dard devia	Median	Mean
2081,9000	1816,6000	66,55645	1978,4000	1966,5500

Preamble :   The objective is to study the links between the CHFUSD and the GOLDUSD for this we must ensure that the dates correspond.

We determine the last line of each CHFUSD and GOLDUSD date range with the End(XlUp) property.
The For Each loop iterates through each date cell in the chfusdDates range.

Inside the loop, the If Not condition checks if the date being processed in "CHFUSD" also exists in the date range of "GOLDUSD". If the date is found, the code continues to copy the corresponding data.
If the copyRangeCHF and copyRangeGOLD ranges are not empty, the corresponding values are copied to the appropriate columns of the COMPARISON sheet

We give the correct names to the columns.

```
last = Comparison.Cells(rows.Count, 1).End(xlUp).Row + 1 ''last row used in the "COMPARISON" page

For Each dateCell In chfusdDates
    If Not goldusdDates.Find(dateCell.Value, LookIn:=xlValues, LookAt:=xlWhole) Is Nothing Then ''search
        Set copyRangeCHF = chfusd.Range("A:F").Find(dateCell.Value, LookIn:=xlValues, LookAt:=xlWhole)
        Set copyRangeGOLD = goldusd.Range("A:F").Find(dateCell.Value, LookIn:=xlValues, LookAt:=xlWhole)

        If Not copyRangeCHF Is Nothing And Not copyRangeGOLD Is Nothing Then '' if this date doesn't exi
            Comparison.Cells(last, 1).Value = copyRangeCHF.Value
            Comparison.Cells(last, 2).Value = copyRangeCHF.Offset(0, 5).Value
            Comparison.Cells(last, 3).Value = copyRangeGOLD.Value
            Comparison.Cells(last, 4).Value = copyRangeGOLD.Offset(0, 5).Value ''it copies the correspon
            last = last + 1
        End If
    End If
Next dateCell

''print the names of columns
Range("A1").Value() = "Date"
Range("D1").Value() = "goldusd"
Range("C1").Value() = "Date"
Range("B1").Value() = "chfusd"
```

Preamble :   The objective is to carry out a regression to find a*x +b with x the price of CHFUSD.

- A little code to put the right titles and prepare the layout.

- We are going to use the Solver function directly from the VBA code for this we are obliged to write the formulas in Excel so that the solver function can be executed and understand the filling logic

- We must therefore fill with random values a and b and calculate in Excel our model, its error and its final precision (sum of errors).

```
Sub PreparingForRegression()
    Dim Comparison As Worksheet
    ''writes the text to prepare the regression
    Set Comparison = ThisWorkbook.Sheets("COMPARISON")
    Range("E1").Value() = "Model"
    Range("F1").Value() = "Error^2"
    Range("G1").Value() = "Sum"
    Range("H1").Value() = "a"
    Range("I1").Value() = "b"

End Sub
```

a	b

- We therefore created the GenerateRandomNumbers VBA code to avoid the user having to enter values by hand. It uses the Rnd() function which will create random numbers in boxes H2 (a) and I2 (b).

```
Dim sheet As Worksheet
Set sheet = ActiveSheet

sheet.Range("H2").Value = (Rnd() * 1000) - 10
sheet.Range("I2").Value = (Rnd() * 1000) - 10 ''generate a random number in cell h2 and i2
```

- The excel formulas are for the model: a* chfusd+ b.

Model
=$H$2*$B2+$I$2

- To calculate the error^2: simply the difference between our estimate and the true value of GoldUsd.

Error^2
=(D2-E2)^2

- And the precision of the model which is the sum of the errors^2.

Sum
=SUM(F2:F97)

=> When we use the chart display by going directly to the "Insert" tab we find the following linear regression:

CHFUSD VS GOLDUSD
y = 2151,4x - 464,78
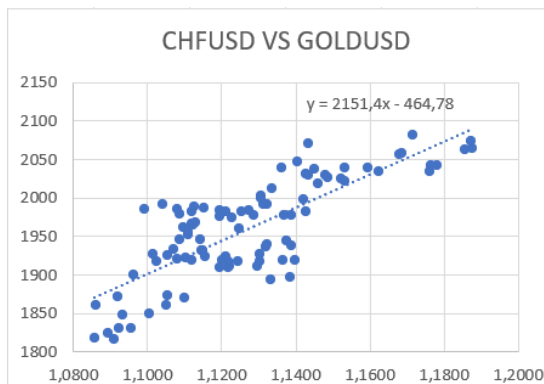
The objective being to find these values by using the solver via VBA.

Linear Regression

- We use a code seen in CM, we start the solver with the Reset function and set the precision and the non-negativity condition.

- Then all that remains is to say the coefficients which vary (a and b which correspond to coeff1stdorder) and the value to be minimized (sum which corresponds to error1storder).

```
Sub GraphTendlineSOlver()

    '' SEEN DURING A CM
    '1st Order Polynomial Model
    SolverReset
    SolverOptions Precision:=0.00001 ''accuracy
    SolverOptions AssumeNonNeg:=False ''assume that non negative

    SolverOk SetCell:="error1stOrder", MaxMinVal:=2, ValueOf:=0, ByChange:="coeff1stdorder", _
    Engine:=1, EngineDesc:="GRG Nonlinear" ''setCell is the cell that we want to minimize with va
    ''define

    SolverSolve UserFinish:=True
    SolverFinish KeepFinal:=2, ReportArray:=0
End Sub
```

- And the result is:

a	b
2151,36	-464,675

- We find the same values as on the chart!

Compute Correlation

▪ We go through our data without taking the first line into account (and we take the opportunity to count the total number of data with n). We calculate sumofB, sumofD, sumofBD (sum of the products Bi * Di), the sum of B squared and the sum of D squared. Then we have all the elements to calculate the correlation and use a msgbox to print the result.

```
Set shet = ThisWorkbook.Sheets("COMPARISON") ''define my active page
last = shet.Cells(shet.rows.Count, "B").End(xlUp).Row ''the index of my last data

For i = 2 To last

    If Not IsEmpty(shet.Cells(i, "B")) And Not IsEmpty(shet.Cells(i, "D")) Then ''if the box is not empty
        n = n + 1 ''number our dataset
        sumofB = sumofB + shet.Cells(i, "B").Value ''sum of B (colonne B)
        sumofD = sumofD + shet.Cells(i, "D").Value ''sum of D (colonne D)
        sumofBD = sumofBD + (shet.Cells(i, "B").Value * shet.Cells(i, "D").Value) 'sum of D*B
        sumofBsquared = sumofBsquared + (shet.Cells(i, "B").Value ^ 2) ''sum of B^2
        sumofDsquared = sumofDsquared + (shet.Cells(i, "D").Value ^ 2) ''sum of D^2
    End If
Next i


correlation = (n * sumofBD - sumofB * sumofD) / ((Sqr(n * sumofBsquared - sumofB ^ 2)) * (Sqr(n * sumofDsquared - sumofD ^ 2)))
MsgBox "The correlation between CHFUSD and GOLDUSD is : " & correlation ''pirnt it
```

The correlation between CHFUSD and GOLDUSD is : 0,793684646838532

▪ And with the CORREL function directly on Excel we obtain:

Correlation
0,7936846

Sources and acknowledgments

Most formulas and concepts come from :

- ESILV Course – Excel + VBA -2023, by *Hamid Belmekki*