Kazandji Olivier / Huang-Dubois Nicolas

* Objectif du TP

L’objectif de ce TP est de programmer en VBA un algorithme de trading qui va décider des différents instants d’ouverture, de clôture ainsi que du prix d’achat/vente d’un produit ou de sa quantité.

* Les bases de notre programme

Puisque le CAC40 est actuellement en pleine croissance et que notre algorithme est uniquement autorisé d’acheter puis de vendre. Nous avons choisi d’utiliser les données du CAC40 sur les 4 derniers mois.

Ces informations ont été extraites de Yahoo Finance.

Nous avons décidé de travailler sur 4 mois afin de faciliter la lecture des données de notre graphique.

* Les algorithmes du programme

Importation des données

Comme dit précédemment, les données utilisées sont extraites de Yahoo Finance. Notre macro « impoter\_cac » affiche sur notre classeur les données suivantes :

* La date du jour
* Le prix à l’ouverture de la bourse
* Le prix le plus haut
* Le prix le plus bas
* Le prix à la clôture

Les données sont classées par ordre de date chronologique, de la plus ancienne à la plus récente. Ceci facilitera le traitement de notre graphique.

Explication du code :

On efface déjà toutes les éventuelles valeurs présentes et graphiques dans la feuille de travail

|  |
| --- |
| Sheets("CAC40").Cells.Clear  Sheets("CAC40").Select  ActiveSheet.DrawingObjects.Delete |

On a ensuite importé les données depuis Yahoo finance en s’aidant de la fonction « Enregistrer une macro »

|  |
| --- |
| With Sheets("CAC40").QueryTables.Add(Connection:="URL;http://fr.finance.yahoo.com/q/hp?s=%5EFCHI", Destination:=Sheets("CAC40").Range("$A$100"))  End With |

Après l’importation des données, nous avions eu besoin de modifier l’emplacement des cellules pour classer par ordre chronologique, de la plus ancienne à la plus récente.

Processus de trading

Tout d’abord les opérations de trading se font par le biais d’un UserForm. Celui-ci demande premièrement de rentrer le montant voulu dans notre porte-monnaie. Notre UserForm est composé de 6 autres boutons. 2 boutons relatifs aux opérations de trading :

* Début de trading
* Fin de trading

Et 4 boutons traitant des actions dites « utilitaires » :

* Afficher document
* Graphique
* Imprimer
* Envoyer

Notre algorithme de trading se base sur les valeurs à l’ouverture de la bourse. Il marche de la manière suivante, il prend la valeur la plus ancienne, appelons là « t ».

Pour réaliser une vente : nous devons obligatoirement acheter la valeur en « t » car la vente à découvert n’est pas autorisée dans cette simulation. Si la valeur en «t » est supérieur à la valeur en « t+1 » nous la vendons, mais si la valeur en « t+1 » est supérieur à la valeur en « t » nous ne faisons rien et nous nous referons à la prochaine valeur soit « t+2 », et ainsi de suite.

Explication du code :

La première action de l’algorithme est de comparer la valeur du jour n°2 avec celle du premier jour.

Nous avons fait en sorte de mettre les conditions de trading dans l’achat. Donc on vérifie que le portefeuille peut acheter et qu’on a une tendance haussière.

|  |
| --- |
| If Range("B" & i).Value < Portefeuille And Range("B" & i - 1).Value < Range("B" & i).Value Then  Range("I" & i).Value = Range("B" & i).Value  Portefeuille = Portefeuille + Range("B" & i).Value  ValeurAction = Range("B" & i).Value  End If |

Concernant l’aspect graphique, nous affichons uniquement les réels points critiques de l’algorithme, c’est-à-dire un achat ou une vente unique. Si l’algorithme veut acheter et vendre, nous ne l’affichons pas car cela donne des informations inutiles au graphique et à l’utilisateur.

Fin de trading

La macro « FinTrading » ferme notre UserForm.

Explication du code :

Nous avons fait en sorte d’utiliser End. Les seules solutions que nous avons trouvé pour arrêter la macro est l’utilisation des touches Ctrl + Pause ou Echap (risquant de provoquer un crash de Excel). Le vrai problème est que nous n’avons pas accès à la souris lorsqu’une macro VBA est en lancement.

Graphique

Cette macro affiche un graphique décrivant les actions effectuées par notre algorithme. Nous avons en abscisse la date et en ordonnée le prix de l’action du CAC40 à l’ouverture. Les actions de ventes et d’achats sont également visibles sur le graphique. Nous avons ainsi un historique précis sur les 4 mois.

Explication du code :

Nous avons eu beaucoup de difficulté pour tracer un graphique convenable représentant nos données. Nous avons ainsi utilisé l’enregistreur de macro pour cette partie du code.

Suppression des différents graphiques dans la page

|  |
| --- |
| ActiveSheet.DrawingObjects.Delete |

Voici la partie du code la plus intéressante que nous pourrions réutiliser afin de réaliser d’autres graphiques en VBA.

|  |
| --- |
| ActiveSheet.Shapes.AddChart2(332, xlLineMarkers).Select  ActiveChart.SetSourceData Source:=Range("CAC40!$A$1:$B$67,CAC40!$I$1:$J$67" \_  )  ActiveChart.ChartTitle.Select  ActiveChart.ChartTitle.Text = \_  "Graphique représentant les actions du processus de trading" |

Imprimer

La macro « imprimer » va imprimer les valeurs extraites de Yahoo Finance, les étapes de trading et le graphique. L’impression se lance avec l’imprimante par défaut.

|  |
| --- |
| ' Imprime la page actuelle CAC40 en utilisant l'imprimante par défault  Sheets("CAC40").Select  ActiveSheet.PrintOut |

Envoyer

La macro « envoyer »

|  |
| --- |
| ' Attention Outlook est obligatoire pour envoyer les mail  ActiveWorkbook.SendMail "kazandji@yahoo.fr", "Tp VBA Olivier Kazandji Nicolas Huang-Dubois ESILV S7", True |

Document

Nous avons écrit les informations importantes dans un fichier document.txt. Voici les principales étapes pour créer un fichier txt, placé sur le fichier local du classeur Excel.

Création du fichier :

|  |
| --- |
| Open ThisWorkbook.Path & "\document.txt" For Output As #1 |

Ecriture dans le fichier :

|  |
| --- |
| For i = 2 To 67 Step (1)  If Range("I" & i).Value <> 0 Then  Print #1, Range("A" & i).Value & " Achat d'une action CAC40 d'une valeur de: " & Range("I" & i).Value  ElseIf Range("J" & i).Value <> 0 Then  Print #1, Range("A" & i).Value & " Vente d'une action CAC40 d'une valeur de: " & Range("J" & i).Value  End If  Next  ' Fermeture du fichier  Close #1 |

Ouverture du fichier :

|  |
| --- |
| Set sh = CreateObject("WScript.Shell")  sh.Run (ThisWorkbook.Path & "\document.txt") |

On ajoute aussi une gestion d’erreur au cas où le chemin de l’utilisateur contient des espaces ou caractères spéciaux.

|  |
| --- |
| On Error GoTo erreur  ‘ Code  erreur:  MsgBox "Attention erreur: Le chemin de votre dossier comprend surement des charactères spéciaux. Impossible de l'ouvrir.", vbExclamation, "Erreur !" |