

Clustering en finance

1. Installer la librairie mpl_finance (pip install)
2. Importer les librairies suivantes :

pandas, numpy et matplotlib.pyplot
candlestick_ohlc de mpl_finance
KMeans depuis sklearn.cluster
normalize depuis sklearn.preprocessing
datetime, timedelta puis datetime
3. Importer les données du fichier yahoofinance-GOOG-20040819-20180120 dans un dataframe actions_df
4. Isoler les données des trente derniers jours enregistrés puis vérifier le type associé aux données de la colonne « Date »
5. Créer les colonnes suivantes
 - a. moyenne : $(\text{Close} + \text{High} + \text{Low})/3$
 - b. variation : $\text{Close} - \text{Open}$
 - c. variation_relative = $\text{variation}/\text{moyenne}$
 - d. ampli_relative = $(\text{High} - \text{Low})/\text{moyenne}$
 - e. variation_jour = variation de la moyenne d'un jour à l'autre
 - f. variation_jour_relative = $\text{variation_jour}/\text{moyenne}$
6. Représenter la variation de l'action moyenne sur 30 jours
7. Utiliser candlestick_ohlc pour représenter un graphique de variation du cours en passant en paramètre les colonnes "Date", "Open", "Close", "High", "Low", colorup = 'g', colordown='r'
8. Normalisez (norm : l2, axis=1)
9. Regrouper les opérations 5 et 8 dans une fonction

Les données historiques sont disponibles sur yahoo finance en utilisant la requête suivante :

https://query1.finance.yahoo.com/v7/finance/download/*****?period1=*****&period2=*****&interval=1d&events=history&crumb=osPYmrnpclly

Les astérisques correspondent respectivement :

- Au symbole de l'entreprise exemple GOOG, AAPL ...
- Début de la période d'enregistrement
- Fin de la période d'enregistrement

Les dates sont exprimées en Timestamp

10. Créer une fonction download prenant en argument, un symbole et retournant les données des 30 derniers jours à partir de la date d'exécution.(utiliser la librairie requests et la méthode post puis récupérer l'attribut text du résultat)
11. Utiliser le dictionnaire de symboles fournis avec les données pour récupérer les données des entreprises listées et concaténer les résultats
12. Appliquez un clustering Kmeans à 8 clusters

13. Evaluer la performance de ce clustering
14. Rechercher éventuellement le meilleur nombre de clusters.

Attendu question 7.

