

Sujet

L'objectif est de calculer les **Volume Weighted Average Price** horaires du Bitcoin, pour chacun des échanges à disposition.

INPUT:

Vous aurez à disposition, sous forme de fichiers .csv, la liste des **trades de bitcoin** ayant eu lieu sur différents **exchanges** de Cryptomonnaies. Vous devrez importer les données issues des différents fichiers.

MANIPULATION DE DONNEES:

Vous devrez ensuite modifier la structure des données pour répondre à vos besoins, probablement effectuer des **jointures** depuis les différents fichiers.

CALCUL:

Enfin, une fois les données préparées, vous pourrez passer à l'étape de calcul des **VWAP pour chaque tranche d'une heure**, pour chaque **exchange**, puis au **global** (tous les échanges réunis). Une fois le calcul effectué, vous pourrez exporter vos résultats sous la forme d'un nouveau fichier .csv.

Page Wikipédia du **Volume Weighted Average Price** :

https://en.wikipedia.org/wiki/Volume-weighted_average_price

Formule :

$$P_{WVAP} = \frac{\sum_j P_j Q_j}{\sum_j Q_j}$$

Avec :

- P_{VWAP} Le Prix Moyen Pondéré par le Volume.
- P_j Le prix du trade j
- Q_j Le volume du trade j
- j représente chaque trade ayant eu lieu au sein d'un intervalle d'une heure.

Aller plus loin

Autres mesures à calculer :

- Volume Weighted Median Price
 - Page Wikipédia de la médiane pondérée : https://en.wikipedia.org/wiki/Weighted_median
 - Créez une fonction qui vous permet de choisir le mode de calcul des cas particuliers (lower, upper, ...) via un paramètre
- Ecart-type
- OHLC

Calculez les mêmes métriques à des **fréquences différentes** :

- Intervalles de **5 minutes**
- Intervalles de **30 minutes**

Vous devrez probablement gérer des valeurs manquantes. Expliquez vos choix !

Rendu

Le rendu du projet Python prendra la forme d'un Jupyter Notebook.

Ce Jupyter Notebook devra être composé de **trois parties** correspondant aux grandes étapes énoncées plus haut. Vous pouvez délimiter ces parties en les dénotant sous forme de titre de premier niveau. Les trois parties sont les suivantes :

- INPUT
- MANIPULATION DES DONNEES
- CALCUL

Chacune des parties contiendra le code correspondant à l'étape, et l'une des dernières cellules devra afficher le résultat sous la forme d'un DataFrame Pandas (ou bien un DataFrame par mesure calculée).

Résultat attendu

Comme précisé plus tôt, il sera important de stocker le **résultat final** sous la forme d'un (ou plusieurs) **DataFrame Pandas**.

Ce DataFrame devra donc, logiquement :

- Être **indexé par des dates**
- Avoir **une ligne par heure** de l'intervalle de temps des données fournies
- Contenir **une colonne par exchange** contenant le VWAP horaire sur cet exchange
- Contenir **une colonne « globale »** contenant le VWAP portant sur tous les trades, peu importe l'échange

Quelques derniers points :

- Le Jupyter Notebook devra être **totalelement exécutable**, sans générer d'erreur.
- Placez des commentaires au sein de vos cellules de code, ou bien décrivez votre code dans des cellules de texte. Le but est d'avoir une « documentation » de votre travail, afin que n'importe qui puisse comprendre rapidement votre code.
- Vous devrez déposer le Jupyter Notebook sur l'espace approprié de votre ENT : <https://cours.univ-paris1.fr/>