Projet: Traitement de données — Analyse et Prédiction de Marchés Financiers

🎯 Objectif global

Vous êtes analyste de marché (seul ou en binôme).

Votre mission est d'explorer, de nettoyer et d'analyser les données historiques du CAC40 afin de concevoir un modèle de prédiction réaliste (statistique ou machine learning) capable d'anticiper le prochain mouvement de marché.
L'objectif n'est pas seulement de prédire, mais de **comprendre et justifier** les comportements observés à travers une analyse complète, personnelle et critique. Vous disposez des fichiers ci-dessous:

cac40.csv

close_cac40.csv

STO.parquet

Ces jeux de données contiennent de nombreuses **erreurs, incohérences et anomalies** (valeurs aberrantes, manquantes, doublons, formats erronés, etc.). Votre mission est d'en extraire une compréhension pertinente des comportements de marché, et éventuellement de tenter une modélisation prédictive.

🔍 1. Exploration et compréhension

1. Analyse exploratoire des données (Exploratory Data Analysis - EDA)

Commencez par une **analyse libre et approfondie** de vos données CAC40. Cette phase vous permet de **comprendre la structure, la qualité et les comportements du marché** avant toute modélisation.

Étapes suggérées

1. **Importation et inspection**
   - Charger vos fichiers CSV ou Parquet.
   - Vérifier la **qualité des données** : types de colonnes, etc …
   - Optimiser la mémoire lors de l'importation (types appropriés, float32…).

- Dans le **rapport**, indiquez et justifiez clairement :
  - Comment vous avez optimiser les performances lors de l'import?
  - Ce que vous pensez de premier abord sur la qualité de la donnée ?

2. **Nettoyage et mise en forme**
   - Vérifier la cohérence temporelle (dates continues, aucun jour manquant).
   - Gérer les valeurs manquantes : suppression ou imputation justifiée.
   - Détecter et corriger les doublons.
   - Transformer les colonnes si nécessaire (Date → datetime, prix → float).
   - Dans le **rapport**, indiquez et justifiez clairement :
     - Quelles valeurs ont été supprimées ou transformées ?
     - Pourquoi ces choix ont été faits ?
     - L'impact de votre nettoyage sur la mémoire (avant/après).
3. **Analyse visuelle de base via graphiques (Matplotlib)**
   - Évolution temporelle du CAC40 (Close) sur toute la période.
   - Corrélation entre variables clés : prix, volume, volatilité.
   - Histogrammes pour visualiser la distribution des rendements.

🕵️ 2. Détection d'anomalies

Après avoir exploré et nettoyé vos données, vous devez effectuer une **analyse des anomalies** afin d'identifier et d'écarter les points atypiques susceptibles de fausser la modélisation prédictive.

L'objectif est de **détecter les comportements anormaux** dans l'évolution du CAC40 — par exemple des variations de prix, de volume ou de volatilité inhabituelles.

🔍 Tâche attendue

Proposez et implémentez **votre propre méthode de détection d'anomalies**, en justifiant vos choix.
Vous pouvez, par exemple, vous appuyer sur :

- **Approches statistiques simples** : seuils basés sur l'écart-type, le *z-score*, les quartiles ou la médiane absolue …

- **Approches temporelles** : comparaison à une **moyenne mobile**, détection de **ruptures de tendance**, ou observation de pics soudains dans la volatilité.

💡 Visualisations suggérées

- **Courbe du cours** avec les anomalies **mises en évidence** (par exemple, points rouges ou annotations sur la ligne du CAC40). Possibilité d'effectuer des zooms sur ces périodes.
- **Histogramme** des variations journalières pour visualiser la distribution et les valeurs extrêmes.
- Comparaison entre l'indice du CAC40 et URO STOXX qui suit en partie le CAC40, certaines anomalies peuvent ne pas être réellement des anomalies mais expliques par un évènement.

Dans le **rapport**, indiquez et justifiez clairement :

- La ou les méthodes pour détecter les anomalies?
- Pourquoi ces choix ont été faits ?
- L'explications si possible des ces anomalies (événements médiatiques etc ..)

📈 3. Phase d'analyse statistique

Cette étape vise à approfondir la compréhension du comportement du marché à travers une **analyse quantitative** des données historiques.

L'objectif est d'identifier les **structures temporelles** et les **caractéristiques statistiques** du CAC40 et d'aller vers une prédiction. Afin d'avoir une idée du comportement du CAC40 des indicateurs ont été créés.

🎯 Objectifs

- Calculer les indicateurs du marché qui sont:
  - Calculer la m**oyennes mobiles multiples** : MA courte (10jours) et la MA longue (20 jours) : *df['MA_10'] = df['Close'].rolling(10).mean()*
  - Calculer le Momentum: Momentum long (20 jours) qui correspond différence entre le prix actuel et le prix N jours avant. df['Close'] - df['Close'].shift(20)
- Sauvegarder les données traitées au format Parquet.
- A l'aide de ces indicateurs, appliquer la règle de statistique suivante:

On prédit que le cours va augmenter si la moyenne mobile sur 10 jours est supérieure à la moyenne mobile sur 20 jours **et** si le prix a augmenté par rapport à il y a quelques jours. Sinon, on prédit que le cours va baisser.

- Calculer la précision du modèle en comparant les prédictions aux valeurs réelles, et exprimer le résultat sous forme de moyenne ou de pourcentage corrects (accuracy).

💡 Visualisations suggérées

- Diagramme qui représente le cours réel (close) et affiche les points où le modèle prédit une hausse ou une baisse, afin de visualiser facilement les prédictions par rapport aux valeurs réelles. »

🧠 4. Prédiction avec du Machine Learning

Maintenant qu'on a :

- des **données propres**,
- des **variables dérivées** (rendement, volatilité, moyennes mobiles),
- et une **bonne compréhension** du comportement du CAC40.

On va donc construire une **prédiction simple et réaliste. Pour cela, vous pouvez utiliser skitik-learn et le model** RandomForestClassifier qui a pour paramètre `n_estimators=100, random_state=42`. Ce qui donne :

model = RandomForestClassifier(n_estimators=100, random_state=42)

Le model doit prendre en entrée - Features:

`'MA_5','MA_20','EMA_5','Momentum_5','ROC_5','Volatility_5','High_Low_Range','Volume_SMA_5','Volume_SMA_10'.` Les variables `'MA_5','MA_20','Momentum_5'` ont déjà été calculées plus haut.

EMA correspond a la moyenne mobile exponentiel est peut être calculé de cette façon:

```
df['EMA_5'] = df['Close'].ewm(span=5, adjust=False).mean()
```

ROC correspond au taux de variations du prix de fermeture:

```
df['ROC_5'] = df['Close'].pct_change(5)
```

La volatilité(expliquée plus bas en détail):

```
df['Volatility_5'] = df['Close'].pct_change().rolling(5).std()
```

Le High_low_Range correspond a l'écart entre le prix le plus haut et le plus bas:

```
df['High_Low_Range'] = df['High'] - df['Low']
```

Et le volume SMA correspond a la Moyenne mobile du volume

```
df['Volume_SMA_5'] = df['Volume'].rolling(5).mean()
```

Le But de la prédiction est de prédire le cours du CAC40 du jour suivant

- 1 si le marché monte,
- 0 s'il baisse.

💡 Visualisations suggérées

- Diagramme qui représente le cours réel (close) et affiche les points où le modèle prédit une hausse ou une baisse, afin de visualiser facilement les prédictions par rapport aux valeurs réelles. »

5. Livrables attendus

1. 🧠 **Notebook Jupyter ou script Python** complet et commenté
   - Code structuré (nettoyage → analyse → visualisation → prédiction)
   - Commentaires clairs
2. 🗒️ **Rapport synthétique** :
   - Approche choisie et la Méthodologie
   - Principaux résultats / graphiques
   - Interprétation et limites

🎓 6. Évaluation (guideline)

Critère	Pondération	Détail
Qualité du nettoyage / préparation	25 %	Gestion des erreurs, justification des choix
Analyse et détection d'anomalies	25 %	Pertinence, rigueur, originalité

Visualisations / interprétations	25 %	Clarté, créativité, storytelling
Modélisation prédictive (option)	5 %	Pertinence, validation, explication
Rapport et clarté globale et participation	20 %	Structure, lisibilité, cohérence

Petites informations:

La volatilité, c'est quoi ?

En simple : La volatilité mesure à quel point le prix d'un actif varie dans le temps.

C'est une **mesure du risque ou de l'incertitude** :

- Si un actif bouge beaucoup (hausse et baisse fortes), il est **très volatil**.
- S'il bouge peu (prix stable), il est **faiblement volatil**.

Exemple concret

Imagine que le CAC40 fasse ceci sur 5 jours :

Jour	Cours de clôture	Variation (%)
Lundi	7000	—
Mardi	7020	+0.3%
Mercredi	7030	+0.1%
Jeudi	6990	-0.6%
Vendredi	7025	+0.5%

Ces variations sont faibles → **faible volatilité**.

Mais on a a :

Jour	Cours de clôture	Variation (%)
Lundi	7000	—
Mardi	7200	+2.8%
Mercredi	6800	-5.5%
Jeudi	7050	+3.7%
Vendredi	6900	-2.1%

Ici les variations sont fortes → **forte volatilité**.

La **volatilité** est souvent mesurée comme l'**écart-type** des rendements d'un actif.

Le rendement journalier correspond à la variation relative du prix de clôture d'un jour sur l'autre :

$$R_t = \frac{C_t - C_{t-1}}{C_{t-1}}$$

ou

Ct = prix de clôture à la date

Ct−1 = prix de clôture à la date précédent

Rt = rendement journalier

Résultat : un pourcentage qui peut être positif (gain) ou négatif (perte).

```
# Exemple simple
df = pd.DataFrame({'Close': [100, 102, 101, 105]})
# Calcul du pourcentage de variation
df['Return'] = df['Close'].pct_change()
print(df)
```

La méthode pct_change() calcule le pourcentage de variation entre une ligne et la précédente dans une série ou un DataFrame.