Mini projet Spark

Consommation de carburants en France
Mag 3 - 2023-2024

Objectifs

Livrable : un notebook Python (Google Colab autorisé) implémentant les fonctionnalités requises comme spécifié ci-après

Objectif : charger et traiter les données sur le prix des carburants avec Spark :

- ❑ Collecter des données
- ❑ Préparer les données
- ❑ Visualiser les variations des prix des carburants
- ❑ Modéliser l'évolution des prix

Collecte de données

Téléchargez les données sur les prix des carburants de 2019 à 2023 (2 années de suite minimum) depuis le dépôt Github suivant : https://github.com/rvm-courses/GasPrices). Ce sont des fichiers préparés à partir du site Web d'origine.

N'hésitez pas à télécharger des années précédentes supplémentaires si vous souhaitez tester les performances sur des volumes plus élevés, mais deux années sinon doivent suffire.

Téléchargez également le fichier Stations-service et le fichier Services (versions 2023).

Préparation des données – étape 1

Avec PySpark :

- Lire et fusionner tous les fichiers de carburants
- Date fractionnée en année, mois, semaine de l'année
- Rendre les données disponibles sous forme de tableau afin de pouvoir utiliser Spark SQL

Grâce à des statistiques de base, déterminez quels types de carburants présentent un certain intérêt pour la suite du projet. Deux d'entre eux présentent peu d'intérêt et pourront être filtrés pour la suite du projet.

Préparation des données – étape 2

Calculer l'indice des prix pour chaque station par semaine :

- Calculez une nouvelle variable appelée « Indice des prix » pour chaque type de carburant vendu dans une station tel que :

$$\boldsymbol{Price\ Index} = \mathbf{100} \times \left(\frac{\boldsymbol{Day\ Price\ in\ station} - \boldsymbol{Average\ Day\ Price\ in\ France}}{\boldsymbol{Average\ Day\ Price\ in\ France}} + \mathbf{1} \right)$$

Calculer l'index de la semaine :

- Calculez une nouvelle variable appelée « Index de la semaine » pour chaque enregistrement en comptant le nombre de semaines depuis la première semaine du fichier.
- Exemple : si vous avez chargé les années 2019 à 2021, la première semaine de 2019 doit être numérotée 1, la dernière semaine de 2021 doit être numérotée 156 (3 x 52)

Visualisation de données

En utilisant matplotlib/seaborn ou plotly :

- Représenter l'évolution hebdomadaire du prix moyen des carburant sur la France tel que :
  - Le prix moyen de chaque type de carburant est représenté par une ligne
  - Les coordonnées en X sont l'index précédemment calculé de la semaine
  - La coordonnée Y est le prix moyen du type de carburant en France sur la semaine.

Astuce : pensez à utiliser Seaborn FacetGrid pour un graphique à plusieurs lignes.

Visualisation des données – Question bonus

Cette question donne droit à des points bonus (+10 %, maximum à 100 %)

- Représenter une carte avec le prix moyen pour chaque type de carburant de France, par exemple au niveau départemental
- Pour cela, vous pouvez calculer le prix moyen agrégé par département pour un carburant que vous choisissez, pour une période donnée (une année) et visualiser le résultat en reprenant le code de l'exemple ci-dessous.

Références : le notebook suivant présente un exemple d'utilisation de Folium pour dessiner une carte au niveau d'un département.

https://github.com/HerveMignot/geopython/blob/main/notebooks/Cartographie_avec_geopandas_%26_folium.ipynb

Modélisation – Prévision du prix le lendemain

- Construire un modèle basé sur Spark ML pour prévoir le prix du lendemain pour un type de carburant donné dans une station donnée
- N'envisagez pas d'utiliser des modèles de séries chronologiques (tels que AR/ARMA/ARIMA) mais appuyez-vous sur les techniques existantes de Spark ML / MLLib telles que LinearRegression , RandomForestRegressor
- Fournissez des mesures de précision pertinentes et un graphique de dispersion pertinent entre le réel et la prévision
- **Astuces** : pensez à utiliser des fonctionnalités de décalage (telles que le prix de la veille, et de la veille, etc.) pour construire le modèle.
  Voir référence : https://arxiv.org/abs/2101.02118
- **Remarque importante** : nous n'attendons pas trop de réglage fin du modèle, mais plutôt un pipeline fonctionnel (c'est-à-dire que la précision finale du modèle ne sera pas utilisée pour évaluer votre travail)

Rendu du projet

Le projet sera rendu sous forme d'un notebook :

- soit un fichier ipynb
- soit un lien vers le notebook dans Google Colab.

Votre code sera revu et évalué avec les commentaires dans le notebook. Assurez-vous de documenter à minima votre code, c'est important pour l'évaluation ! L'ajout de tests pertinents serait un plus.

Aucun livrable supplémentaire n'est attendu.

Des questions ?

herve.mignot@equancy.com