DataMgmt - Laboratoire n°4
Data Preprocessing

16.12.2021

1. Introduction

Ce laboratoire a pour but d'explorer différents types de prétraitement des données afin d'analyser ensemble les données de 6 magasins vendant les mêmes produits. Chaque magasin possédant ses propres particularités, les données devront être donc nettoyées, transformées et intégrées afin de les combiner en un seul et même format. Ces données combinées pourront ensuite être analysées.

Nous allons utiliser :

- Jupyter Notebook pour organiser notre code et visualiser nos résultats.
- La librairie pandas pour le prétraitement et l'analyse des données.
- La librairie matplotlib pour créer nos graphiques.

1.1 Organisation

Ce laboratoire doit être réalisé par groupe de 2 étudiants au maximum.

1.2 Rendu

Rendre un zip contenant :

- Le code source de votre implémentation.
- Un mini-rapport répondant aux questions et contenant les graphiques demandés dans la partie *2.2 Analyse de données*.

1.3 Date de rendu

Voir sur moodle.

1.4 Environnement de travail

- Vous devez avoir Python 3.7 ou plus récent d'installé.
- Il peut vous être utile de créer un environnement virtuel.

- *requirements.txt* : contient la liste des dépendances python. Pour les installer :
  ```
  > pip install -r "requirements.txt"
  ```

  Il vous faudra également lancer un noyau ipython
  ```
  > ipython kernel install --user --name=lab4
  ```

  Puis lancer le notebook
  ```
  > jupyter notebook
  ```

1.5 Fichiers fournis

- Dans le dossier *data*, les données des 6 magasins que l'on va vouloir analyser.
- *index.ipynb*: fichier **à compléter** qui contiendra le code permettant de prétraiter et d'analyser nos données.

2. Travail demandé

2.1 Intégration des données

On souhaite tout d'abord combiner les différentes données des différents magasins. Chaque magasin possède des particularités qui lui sont propres et peut avoir des données incohérentes. Avant d'analyser ces données, il nous faut donc les nettoyer, les transformer, les réduire afin de pouvoir les combiner dans un format uniforme.

Les différentes particularités des données de chaque magasin sont listées ci-dessous :

2.1.1 Magasin n°1 : Fichier CSV

- Ce magasin possède des données propres. On va s'inspirer de la structure des données (noms et nombre de colonnes) de ce magasin pour joindre les données des autres magasins.

2.1.2 Magasin n°2 : Fichier Excel

- Certaines lignes contiennent des lignes vides (voir par exemple la première ligne). On vous demande donc de supprimer toutes ces lignes vides.

2.1.3 Magasin n°3 : Fichier JSON

- Certaines lignes contiennent des données incohérentes (elles contiennent les mêmes données que les entêtes, voir la première ligne comme exemple), on vous demande donc de supprimer ces lignes.
- Le format utilisé pour la date de commande (Order Date) est différent de celui utilisé dans les autres magasin (le format est par exemple "01/28/21 14:34"). On vous demande de convertir ces dates dans le même format que les autres données.
    - https://pandas.pydata.org/docs/reference/api/pandas.to_datetime.html?highlight=to_datetime#pandas.to_datetime

2.1.4 Magasin n°4 : Fichier XML

- Comme il s'agit d'un fichier XML, les noms des éléments suivent une convention et ne peuvent donc pas contenir d'espace. On vous demande donc de renommer les entêtes des colonnes afin de respecter la structure des données des autres magasins.
- Ce magasin possède une seule colonne (purchaseAddress) contenant toutes les données relatives à une adresse. Il faudra distribuer cette colonne en 3 (Street Address, City et State) pour respecter la structure des données des autres magasins.
- Ce magasin possède également une colonne processedBy qui indique quelle personne a traité la commande. Cette colonne ne nous intéresse pas et on vous demande donc de la supprimer du dataframe.

2.1.5 Magasin n°5 : Fichier Parquet

- Certains produits de ce magasin ont un prix unitaire incohérent de 0 (voir par exemple la première ligne). On vous demande de modifier toutes ces données incohérentes en les remplaçant par le prix du produit correspondant pour ce magasin. Tous les produits de ce magasin ont toujours le même prix, on va donc rechercher le prix de tous les produits de ce magasin afin de remplacer ces valeurs incohérentes. Par exemple, si on trouve une commande ayant commandé un USB-C Charging Cable avec un prix unitaire de 0, on vous demande de remplacer ce prix unitaire par 11.49 car ce magasin vend toujours ses USB-C Charging Cable à 11.49.

2.1.6 Magasin n°6 : Fichier SQLite

- Les entêtes des colonnes doivent également être renommés afin de respecter la structure des données des autres magasins.

- La colonne concernant la quantité commandée de chaque produit est manquante. On vous demande de recréer vous-même cette colonne en utilisant les données des 5 magasins précédent. Pour cela, pour chaque produit où cette quantité est manquante, on va prendre l'arrondi entier de la moyenne de la quantité commandée pour ce produit sur toutes les autres commandes des 5 autres magasins. Par exemple, pour le produit USB-C Charging Cable, on va chercher quelle est la moyenne de quantité commandée pour ce produit, on trouve par exemple une moyenne de 3.35. On va donc prendre la valeur 3 comme quantité commandée pour tous les produits USB-C Charging Cable de ce magasin n°6.

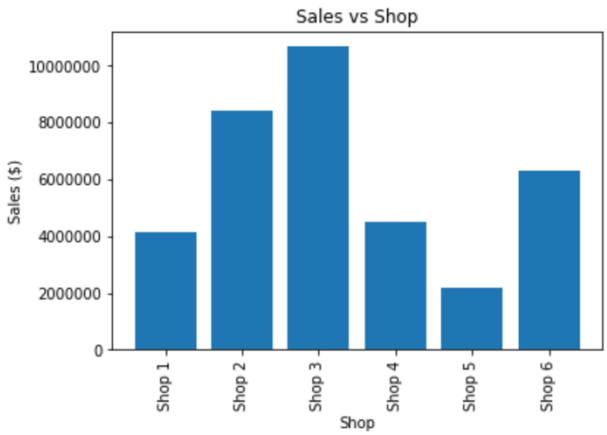
    - https://pandas.pydata.org/docs/reference/groupby.html

2.2 Analyse des données

Après avoir traité et intégré les données de ces 6 magasins, on va vouloir effectuer quelques analyses simples sur ces données, et on particulier à répondre aux questions suivantes :

2.2.1 Prix total des ventes par magasins

- On souhaite mettre en évidence le prix total des ventes des différents **magasins** les uns par rapport aux autres. On souhaite pour cela afficher un graphique ayant la forme suivante :

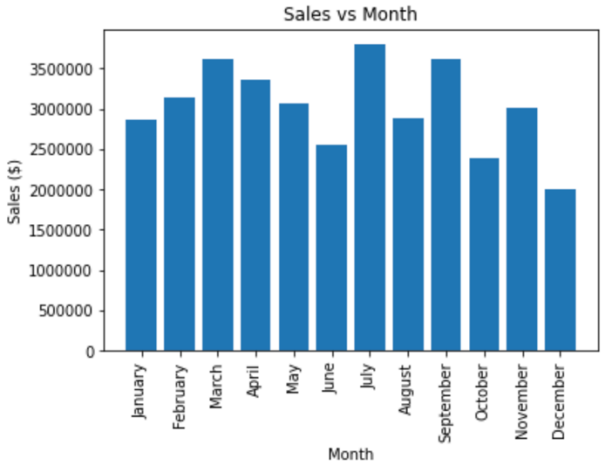


- On vous demande également de répondre aux questions suivantes :

- Quel magasin possède le prix total des ventes le plus élevé ? Quel est ce prix total ?

2.2.2 Prix total des ventes par mois

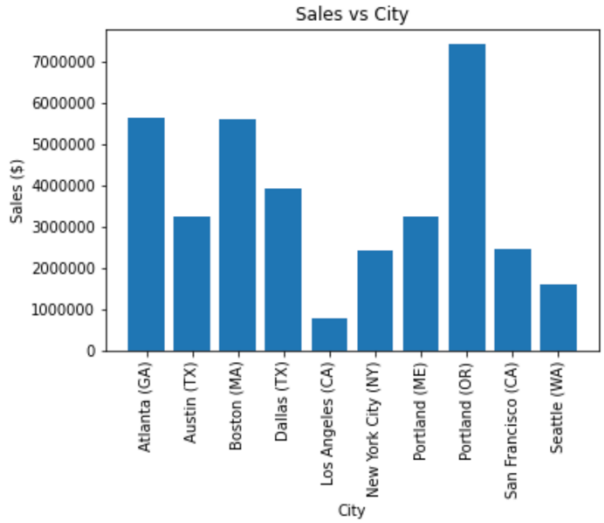
- On souhaite à présent comparer le prix total des ventes des différents **mois** les uns par rapport aux autres. On souhaite pour cela afficher un graphique ayant la forme suivante :



- On vous demande également de répondre aux questions suivantes :
  - Quel mois possède le prix total des ventes le plus élevé ? Quel est ce prix total ?

2.2.3 Prix total des ventes par villes

- On souhaite à présent comparer le prix total des ventes des différentes **villes** les unes par rapport aux autres. On souhaite pour cela afficher un graphique ayant la forme suivante :

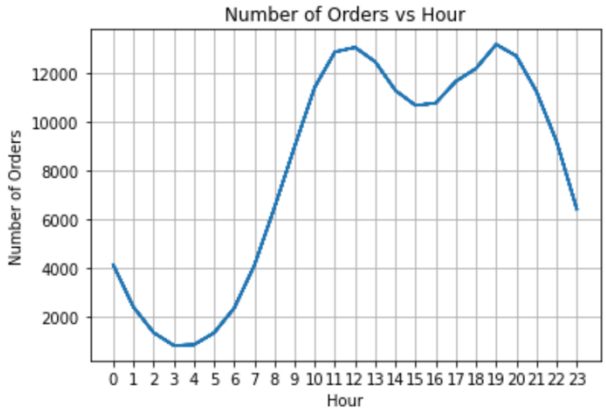


- Attention au fait que certaines villes possèdent le même nom, par exemple Portland dans le Maine (ME) et Portland dans l'Oregon (OR). Il vous faudra utiliser l'état où se situent ces villes pour les différencier.

- On vous demande également de répondre aux questions suivantes :
  - Quelle ville possède le prix total des ventes le plus élevé ? Quel est ce prix total ?

2.2.4 Nombre de commandes selon l'heure de la journée

- On souhaite à présent comparer le nombre de commandes effectuées selon l'heure de la journée. On souhaite pour cela afficher un graphique ayant la forme suivante :



- On vous demande également de répondre aux questions suivantes :
    - Quelle sont les 3 heures ayant le nombre de commandes effectuées le plus élevé ? Pour chacune de ces 3 heures, quel est le nombre de commandes effectuées ?

2.2.5 Produits souvent vendus ensemble

On souhaite pour finir déterminer les produits qui sont régulièrement achetés ensemble (lors d'une même commande). On vous demande de lister le top 10 des produits qui sont achetés conjointement et le nombre de fois où ils ont été vendus ensemble.

Exemple de réponse :

```
('27in FHD Monitor', 'Wired Headphones') : 2298
('27in FHD Monitor', 'USB-C Charging Cable') : 1338
('ThinkPad Laptop', 'Wired Headphones') : 1248
('ThinkPad Laptop', 'USB-C Charging Cable') : 775
('USB-C Charging Cable', 'Wired Headphones') : 663
('27in FHD Monitor', 'Bose SoundSport Headphones') : 309
('Bose SoundSport Headphones', 'Wired Headphones') : 181
('LG Dryer', 'Wired Headphones') : 179
('ThinkPad Laptop', 'Bose SoundSport Headphones') : 167
('USB-C Charging Cable', 'Bose SoundSport Headphones') : 127
```