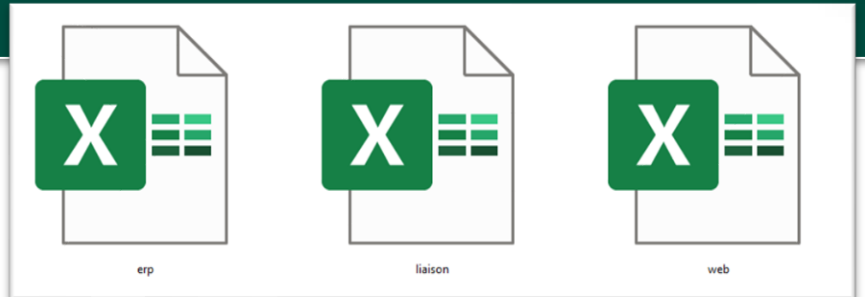
Optimisez la gestion des données d'une boutique avec Python

Melchiori Manuel
Data analyst
Juillet 2024

Analyses Exploratoires des Données

- ***3 Fichiers excel***




```
1 #Afficher les dimensions du dataset
2 print("Le tableau comporte {} observation(s) ou article(s)".format(df_erp.shape[0]))
```

Erp: 6 colonnes 825 lignes
Liaison: 2 colonnes 825 lignes
Web: 29 colonnes 1513 lignes

Noms des dataframes

```
1 #Consulter le nombre de colonnes
2 print("Le tableau comporte {} colonne(s)".format(df_erp.shape[1]))
```

Traitements réalisés

Contrôle des Données Manquantes (NaN)

•Les valeurs manquantes, souvent représentées par NaN (Not a Number), peuvent fausser notre analyse si elles ne sont pas correctement traitées.

Contrôle des Doublons

•Les doublons peuvent fausser nos résultats en donnant un poids disproportionné à certaines entrées.

Gestion des Valeurs Aberrantes

•Les valeurs aberrantes sont des valeurs qui sont nettement différentes des autres valeurs observées. A garder dans le contexte De vente de vins

Contrôle des Valeurs Incohérentes

•Nous avons vérifié la cohérence des données, par exemple en s'assurant qu'il n'y a pas de prix négatifs.
•Les valeurs incohérentes peuvent indiquer des erreurs dans la collecte ou l'enregistrement des données.

Traitement des Données Inexploitables

Les données inexploitables peuvent être des données qui sont mal formatées, incomplètes ou qui ne correspondent pas à nos besoins d'analyse.

Remarques éventuelles, pièges ou difficultés rencontrées

		product_id	onsale_web	price	stock_quantity	stock_status	purchase_price
0	0	3847	1	24.2	16	instock	12.88
1	1	3849	1	34.3	10	instock	17.54
2	2	3850	1	20.8	0	outofstock	10.64
3	3	4032	1	14.1	26	instock	6.92
4	4	4039	1	46.0	3	outofstock	23.77

Des incohérences deviennent évidentes dès la première lecture.

Remarques éventuelles, pièges ou difficultés rencontrées

Nombre d'articles avec un prix non renseigné POUR LE FICHIER ERP : 0

Prix minimum : -20.0

Prix maximum : 225.0

Prix inférieurs à 0 :

		product_id	onsale_web	price	stock_quantity	stock_status	purchase_price
0	151	4233	0	-20.0	0	outofstock	10.33
1	469	5017	0	-8.0	0	outofstock	4.34
2	739	6594	0	-9.1	19	instock	4.61

Des prix négatifs ?

Remarques éventuelles, pièges ou difficultés rencontrées

		sku	total_sales	post_date	product_type	post_name	post_modified
0	8	NaN	NaN	NaT	NaN	NaN	NaT
1	20	NaN	NaN	NaT	NaN	NaN	NaT
2	30	NaN	NaN	NaT	NaN	NaN	NaT
3	37	NaN	NaN	NaT	NaN	NaN	NaT
4	41	NaN	NaN	NaT	NaN	NaN	NaT
5	...	...	...	...	...	...	...
6	1384	NaN	NaN	NaT	NaN	NaN	NaT
7	1429	NaN	NaN	NaT	NaN	NaN	NaT
8	1432	NaN	NaN	NaT	NaN	NaN	NaT
9	1445	NaN	NaN	NaT	NaN	NaN	NaT
10	1457	NaN	NaN	NaT	NaN	NaN	NaT

Des données non exploitables

Jointure de liaison et erp

Pour le fichier liaison
La clé unique était visible

```
Nombre de valeurs présentes dans chaque colonne :

id_web        734
product_id    825
dtype: int64
```

```
Nombre de valeurs uniques dans la colonne 'product_id' :

825
```

Une jointure pour df_merge avec contrôle du nombre de lignes était adapté

```
df_merge = pd.merge(df_erp2, df_liaison, on='product_id', how='outer')
```

```
825 rows × 7 columns

Doublons dans df_merge:

0
```

Problématiques

Difficultés principales:
Le nettoyage de df_web

```python
#Visualisation des valeurs de la colonne sku, il s'agit d'un Id du produit, donc un entier
print("Visualisation des valeurs de la colonne 'sku' :")
display(df_web2['sku'])

# Trouver les valeurs qui ne sont pas des entiers
non_int_values = df_web2['sku'][~df_web2['sku'].apply(lambda x: str(x).isdigit())]
print("\nValeurs qui ne sont pas des entiers :")
display(non_int_values)

# Afficher le nombre de valeurs non entières
print("\nNombre de valeurs qui ne respectent pas la règle de codification (doivent être des entiers) :")
print('Il y a', non_int_values.value_counts().sum(), 'valeurs incorrectes')

# Recherche de valeurs nulles
null_values = df_web2['sku'].isnull()
print("\nValeurs nulles dans la colonne 'sku' :")
display(df_web2[null_values])

# Recherche de valeurs en double
duplicated_values = df_web2['sku'].duplicated()
print("\nValeurs en double dans la colonne 'sku' :")
display(df_web2[duplicated_values])
```

Jointure de df_merge et df_web2

Inversement pour le fichier Web

Id_web = sku

Sku du fichier web était également
Remplis de doublons, valeurs nulles, nan,,,

```
Nombre de valeurs présentes dans chaque colonne :

id_web         734
product_id     825
dtype: int64
```

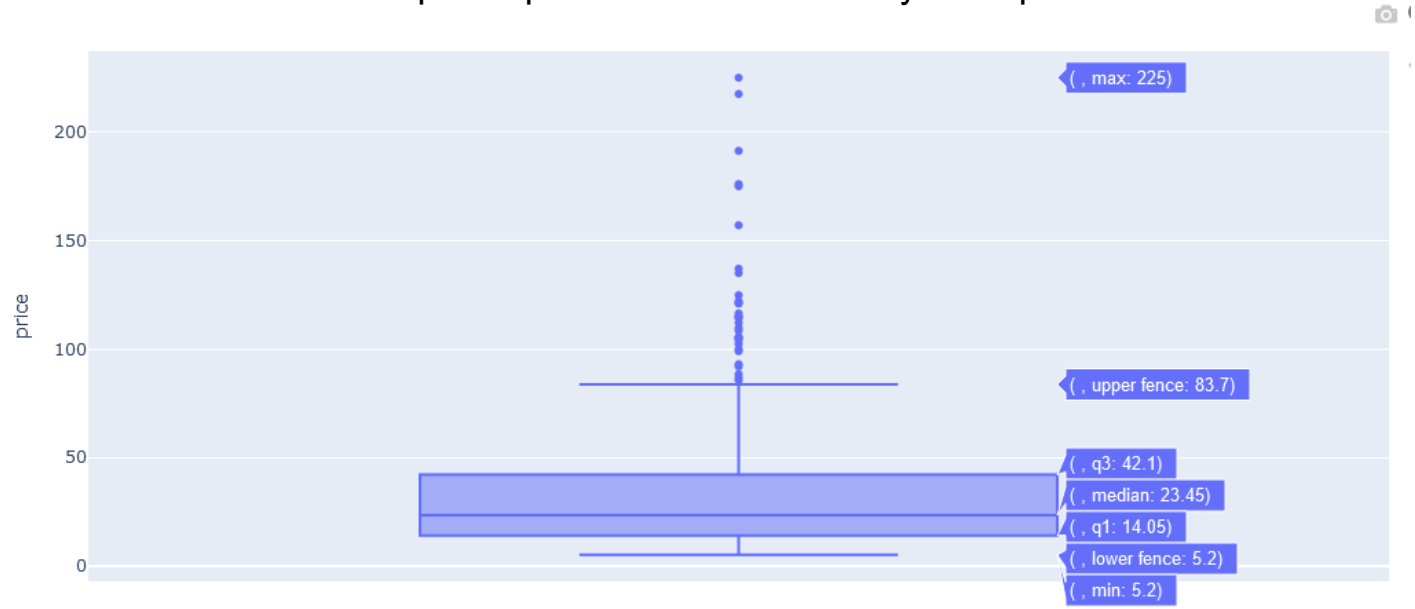
```
Articles sans correspondances dans la colonne 'id_web' :

91
```

```
DataFrame après suppression des doublons :
 714 rows × 6 columns
```

Une jointure interne de id_web sur Sku du df_web2 nettoyé, on arrive à un df de 714 lignes

Analyses univariées du prix

Boite a moustache très simplifiée pour la lecture et l'analyse du prix



Limites éventuelles de l'analyse

En analysant de façon globale, on ne fait pas la distinction entre les types de produits

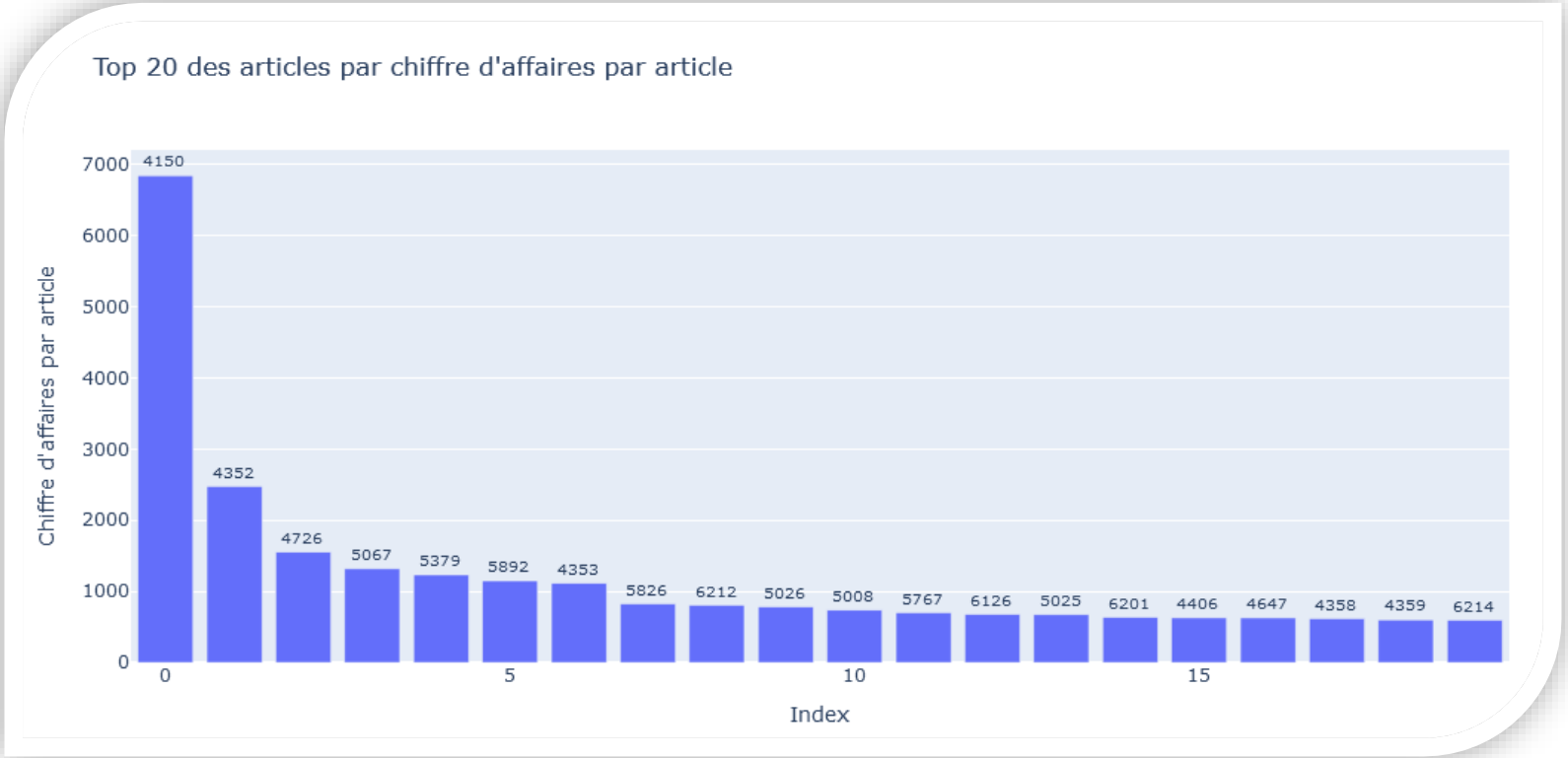
Les vins, champagne et cognac n'ont pas forcément les mêmes « zones » de prix
La propreté des jeux de données est également primordiale

Par exemple, parmi les vins les plus chers, nous avons :

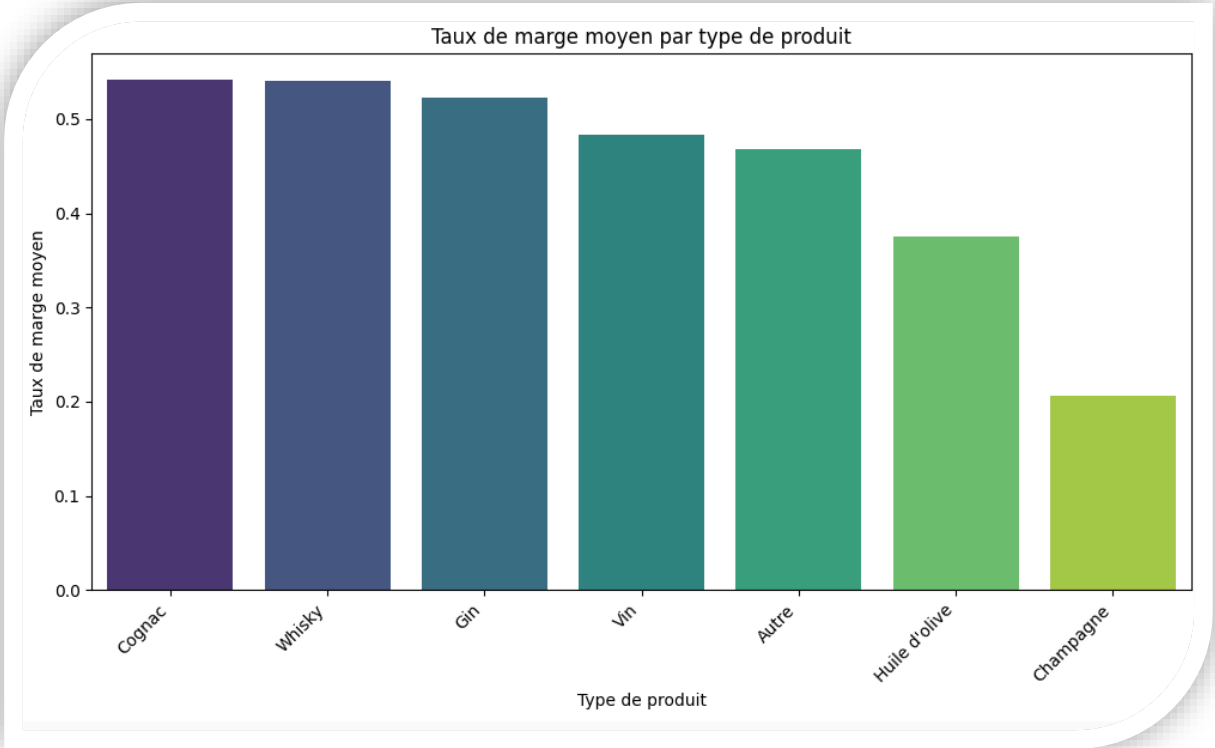
- **Champagne Egly-Ouriet Grand Cru Millésime 2008**
- **David Duband Charmes-Chambertin Grand Cru 2014**

Les produits de prestiges peuvent également secouer les données

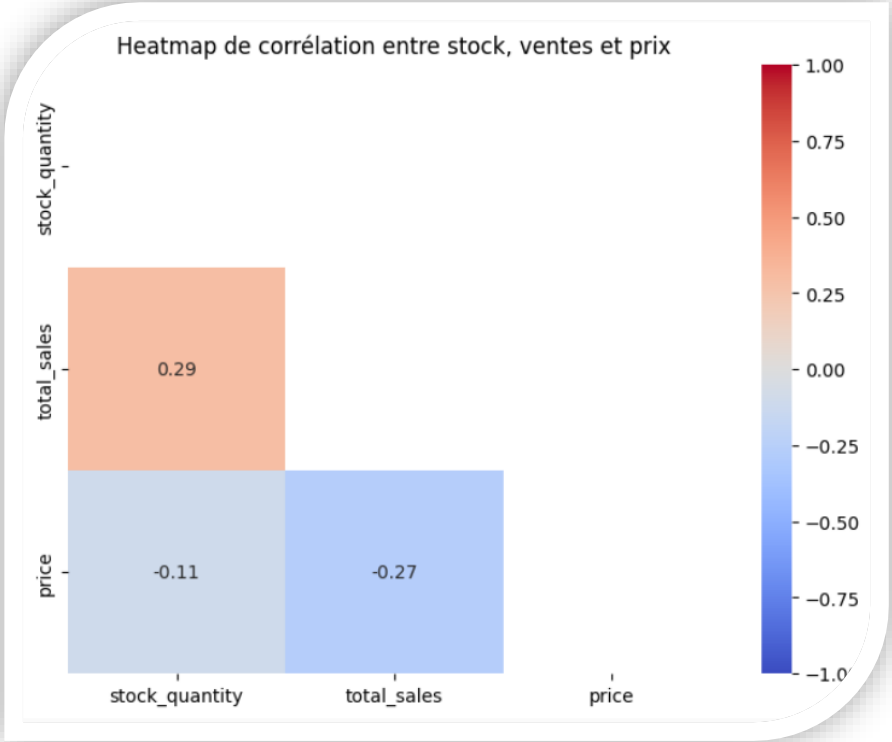
Analyses complémentaires
CA, quantités, stocks, taux de marge et correlations



Analyses complémentaires
CA, quantités, stocks, taux de marge et correlations



Analyses complémentaires
Corrélation via Heatmap



Actions pour la suite

-Nettoyage Continu des Données pour exploitation plus simplifiée, ou mettre en place des garde-fous, voir automatisation éventuelle

-Revue éventuelle des prix, on constate qu'une augmentation des prix menaient a moins de ventes,

-Mise en place d'une gestion de stock plus équilibrée

Conclusion

Projet a été assez exigeant en raison de la quantité importante de nettoyage de données à effectuer.

Les défis majeurs résidaient dans la correction de la syntaxe, qui peut être améliorée grâce à l'intelligence artificielle, et la répétition de certaines actions, notamment le copier-coller de mêmes types d'analyses sur différents fichiers.

Malgré ces défis, l'étude a pris forme et as été très enrichissante,