Analyse des données de la Boutique Bottleneck

David GAMARD

Formation OpenClassrooms Data Analyst

Missions

- ► Nettoyage des données
- ► Rapprochement des données de l'ERP et du CMS
- ► Calcul du chiffre d'affaire par produits
- ► Calcul du chiffre d'affaire total
- ► Analyse des prix des produits

bottleneck
fine wine spirit

Ressources

- ► Données venant de l'ERP
- ► Données venant du CMS
- ► Table de liaison

bottleneck
fine wine spirit

Outils

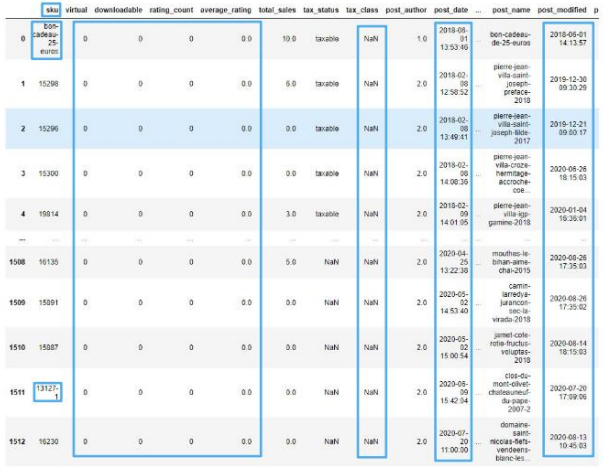
- Langage de programmation : Python

- Logiciel : Jupyter

bottleneck
fine wine spirit

Nettoyage des données
Première vu des données WEB

```
RangeIndex: 1513 entries, 0 to 1512
Data columns (total 28 columns):
 #   Column                Non-Null Count  Dtype
---  ------                --------------  -----
 0   sku                   1428 non-null   object
 1   virtual               1513 non-null   int64
 2   downloadable          1513 non-null   int64
 3   rating_count          1513 non-null   int64
 4   average_rating        1430 non-null   float64
 5   total_sales           1430 non-null   float64
 6   tax_status            716 non-null    object
 7   tax_class             0 non-null      float64
 8   post_author           1430 non-null   float64
 9   post_date             1430 non-null   object
 10  post_date_gmt         1430 non-null   object
 11  post_content          0 non-null      float64
 12  post_title            1430 non-null   object
 13  post_excerpt          716 non-null    object
 14  post_status           1430 non-null   object
 15  comment_status        1430 non-null   object
 16  ping_status           1430 non-null   object
 17  post_password         0 non-null      float64
 18  post_name             1430 non-null   object
 19  post_modified         1430 non-null   object
 20  post_modified_gmt     1430 non-null   object
 21  post_content_filtered 0 non-null      float64
 22  post_parent           1430 non-null   float64
 23  guid                  1430 non-null   object
 24  menu_order            1430 non-null   float64
 25  post_type             1430 non-null   object
 26  post_mime_type        714 non-null    object
 27  comment_count         1430 non-null   float64
dtypes: float64(10), int64(3), object(15)
```



	sku	virtual	downloadable	rating_count	average_rating	total_sales	tax_status	tax_class	post_author	post_date	...	post_name	post_modified	p
0	bon-cadeau-25-euros	0	0	0	0.0	10.0	taxable	NaN	1.0	2018-06-01 13:53:49	...	bon-cadeau-de-25-euros	2018-06-01 14:13:57	
1	15298	0	0	0	0.0	5.0	taxable	NaN	2.0	2018-02-08 12:50:52	...	pierre-jean-villa-saint-joseph-preface-2018	2019-12-30 09:30:29	
2	15296	0	0	0	0.0	0.0	taxable	NaN	2.0	2018-02-08 13:49:41	...	pierre-jean-villa-saint-joseph-tilde-2017	2019-12-21 09:00:17	
3	15300	0	0	0	0.0	0.0	taxable	NaN	2.0	2018-02-08 14:00:36	...	pierre-jean-villa-croze-hermitage-accroche-coe...	2020-08-26 18:15:03	
4	19814	0	0	0	0.0	3.0	taxable	NaN	2.0	2018-02-09 14:01:05	...	pierre-jean-villa-igp-gamine-2018	2020-01-04 19:39:01	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
1508	16135	0	0	0	0.0	5.0	NaN	NaN	2.0	2020-04-25 13:22:38	...	mouthes-le-bihan-ame-chai-2015	2020-08-26 17:35:03	
1509	15891	0	0	0	0.0	0.0	NaN	NaN	2.0	2020-05-02 14:53:40	...	camin-larredya-jurancon-sec-la-virada-2018	2020-08-26 17:35:02	
1510	15887	0	0	0	0.0	0.0	NaN	NaN	2.0	2020-05-02 15:00:54	...	jamet-cote-rotie-fructus-voluptas-2019	2020-08-14 18:15:03	
1511	13127-1	0	0	0	0.0	0.0	NaN	NaN	2.0	2020-06-09 15:42:04	...	clos-du-mont-olivet-chateauneuf-du-pape-2007-2	2020-07-20 17:59:06	
1512	16230	0	0	0	0.0	0.0	NaN	NaN	2.0	2020-07-20 11:00:00	...	domaine-saint-nicolas-fiefs-vendeens-blanc-les...	2020-08-13 10:45:03	

Constats :

- Erreurs lexical (bon-cadeau-25-euros et 13127-1)
- Beaucoup de valeurs nulles (NaN) et de zéro
- Des dates au format datetime

bottleneck
fine wine spirit

Nettoyage des données
Suppressions des colonnes inutiles

```
le data set de web contien 1513 lignes
le data set de web contien 29 colonnes
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1513 entries, 0 to 1512
Data columns (total 29 columns):
 #   Column                Non-Null Count  Dtype
---  ------                --------------  -----
 0   sku                   1428 non-null   object
 1   virtual               1513 non-null   int64
 2   downloadable          1513 non-null   int64
 3   rating_count          1513 non-null   int64
 4   average_rating        1430 non-null   float64
 5   total_sales           1430 non-null   float64
 6   tax_status            716 non-null    object
 7   tax_class             0 non-null      float64
 8   post_author           1430 non-null   float64
 9   post_date             1430 non-null   datetime64[ns]
 10  post_date_gmt         1430 non-null   datetime64[ns]
 11  post_content          0 non-null      float64
 12  product_type          1429 non-null   object
 13  post_title            1430 non-null   object
 14  post_excerpt          716 non-null    object
 15  post_status           1430 non-null   object
 16  comment_status        1430 non-null   object
 17  ping_status           1430 non-null   object
 18  post_password         0 non-null      float64
 19  post_name             1430 non-null   object
 20  post_modified         1430 non-null   datetime64[ns]
 21  post_modified_gmt     1430 non-null   datetime64[ns]
 22  post_content_filtered  0 non-null     float64
 23  post_parent           1430 non-null   float64
 24  guid                  1430 non-null   object
 25  menu_order            1430 non-null   float64
 26  post_type             1430 non-null   object
 27  post_mime_type        714 non-null    object
 28  comment_count         1430 non-null   float64
dtypes: datetime64[ns](4), float64(10), int64(3), object(12)
memory usage: 342.9+ KB
```

Nous decidons alors de garder les uniquement les valeurs qui nous servirons , a savoir les colonnes liées a la vente , et aux quantités

```
#Selon vous, quelles sont les colonnes à conserver ?


df_web = df_web.drop([
    'average_rating', 'post_parent', 'menu_order', 'comment_count',
    'tax_status', 'post_author', 'post_date', 'post_date_gmt',
    'post_status', 'comment_status', 'ping_status', 'post_name',
    'post_modified', 'post_modified_gmt', 'guid',
    'post_mime_type','tax_class','rating_count','downloadable','post_content','post_password',' post_content_filtered','post_content_filtered
], axis='columns', errors='ignore')

df_web.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1513 entries, 0 to 1512
Data columns (total 6 columns):
 #   Column        Non-Null Count  Dtype
---  ------        --------------  -----
 0   sku           1428 non-null   object
 1   total_sales   1430 non-null   float64
 2   product_type  1429 non-null   object
 3   post_title    1430 non-null   object
 4   post_excerpt  716 non-null    object
 5   post_type     1430 non-null   object
dtypes: float64(1), object(5)
memory usage: 71.1+ KB
```

- Beaucoup de colonnes comporte plusieurs lignes vide (83 lignes)
- 4 Colonnes comportant que des valeurs nulles (NaN)
- 3 Colonnes comportant beaucoup de valeurs nulle (NaN)

bottleneck
fine wine spirit

Nettoyage des données
Suppression des valeurs anormales sur la clé

```
# affichage Anomalie 1 :
df_web[df_web["sku"] == 'bon-cadeau-25-euros']

# choix apporté je conserve cette apellation bon-cadeau-25-euros et mais je supprime l'un des deux dounlons pour garder l'authentique "prod
df_web = df_web.drop(index=842)
df_web[df_web["sku"] == 'bon-cadeau-25-euros']
```

	sku	total_sales	product_type	post_title	post_excerpt	post_type
1387	bon-cadeau-25-euros	7.0	NaN	Bon cadeau de 25€	Parlons ...	product

```
# affichage Anomalie 2 :
df_web[df_web["sku"] == "13127-1"]

# choix apporté  : je supprime l'un des deux doublons pour garder l'authentique "product"
df_web = df_web.drop(index=1117)
```

```
df_web[df_web["sku"] == "13127"]
```

sku	total_sales	product_type	post_title	post_excerpt	post_type

- Recherche des ‘sku’ ne respectant pas la règle de codification

bottleneck
fine wine spirit

Nettoyage des données
Séléction des données « product » et non «attachement »

Quelques variables, dont "post_type", différent. Cette dernière prend soit la valeur "product", soit la valeur "attachment". Nous n'allons conserver que les lignes "product". Malgré tout, par mesure de sécurité, vérifions si "post_type" ne prend pas d'autre(s) valeur(s).

```
df_web['post_type'].unique()
```

```
array(['attachment', 'product', nan], dtype=object)
```

La vérification est concluante, on peut supprimer les lignes.

```
df_web = df_web.loc[df_web['post_type'] != 'attachment']
```

Voila maintenant a quoi ressemble le dataframe.

```
df_web
```

	sku	total_sales	product_type	post_title	post_excerpt	post_type
2	14692	5.0	Vin	Château Fonréaud Bordeaux Blanc Le Cygne 2016	Grâce à la complémentarité des 3 cépages ...	product
4	15328	2.0	Vin	Agnès Levet Côte Rôtie Maestria 2017	<span style="float: none; background-color: tr...	product
6	16515	10.0	Vin	Château Turcaud Bordeaux Rouge Cuvée Majeure 2018	\n<div id="container-wrapper...	product
8	NaN	NaN	NaN	NaN	NaN	NaN
11	16585	15.0	Vin	Xavier Frissant Touraine Sauvignon 2019	Un joli sauvignon frais et minéral, avec d'int...	product
...	...	...	...	...	...	...
1503	13074	4.0	Vin	Château de Vaudieu Châteauneuf-du-Pape L'Avenu...	"L'Avenue" est issue d'une parcelle de vieux g...	product
1505	16322	0.0	Vin	Moulin de Gassac IGP Pays d'Hérault Guilhem Ro...	Belle complexité aromatique alliant fruits rou...	product
1507	12365	10.0	Vin	Parés Baltà Penedès Electio 2013	Une cuvée produite avec une très vieille vigne...	product
1508	16326	5.0	Vin	Camin Larredya Jurançon Moelleux Au Capcéu 2018	Sur le millésime 2017, Au Capceu du domaine Ca...	product
1509	15662	15.0	Vin	Chermette Domaine du Vissoux Beaujolais Griott...	C'est le Beaujolais typique : fruité, frais, g...	product

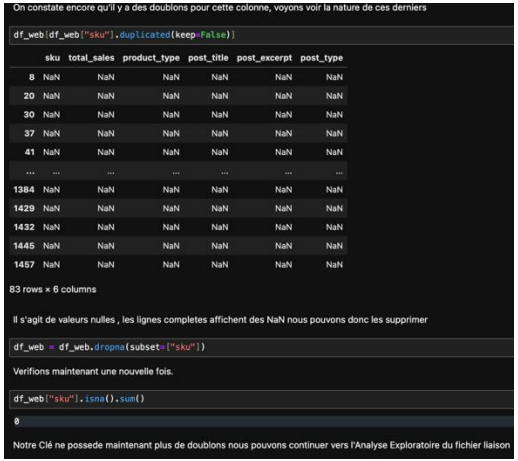
799 rows × 6 columns

- Tout les produits ont des doublons
- Chaque produit est attacher à une pièce joint (image,etc…)

- Reconstruction du DataFrame avec les données intéressante pour notre étude

bottleneck
fine wine spirit

Nettoyage des données
Traitement des valeurs nulles (NaN)



On constate encore qu'il y a des doublons pour cette colonne, voyons voir la nature de ces derniers

```
df_web[df_web["sku"].duplicated(keep=False)]
```

	sku	total_sales	product_type	post_title	post_excerpt	post_type
8	NaN	NaN	NaN	NaN	NaN	NaN
20	NaN	NaN	NaN	NaN	NaN	NaN
30	NaN	NaN	NaN	NaN	NaN	NaN
37	NaN	NaN	NaN	NaN	NaN	NaN
41	NaN	NaN	NaN	NaN	NaN	NaN
...	...	...	...	...	...	...
1384	NaN	NaN	NaN	NaN	NaN	NaN
1429	NaN	NaN	NaN	NaN	NaN	NaN
1432	NaN	NaN	NaN	NaN	NaN	NaN
1445	NaN	NaN	NaN	NaN	NaN	NaN
1457	NaN	NaN	NaN	NaN	NaN	NaN

83 rows × 6 columns

Il s'agit de valeurs nulles , les lignes completes affichent des NaN nous pouvons donc les supprimer

```
df_web = df_web.dropna(subset=["sku"])
```

Verifions maintenant une nouvelle fois.

```
df_web["sku"].isna().sum()
```

```
0
```

Notre Clé ne possede maintenant plus de doublons nous pouvons continuer vers l'Analyse Exploratoire du fichier liaison

- Remplacement de zéro par des valeur (NaN)
- Suppression des lignes contenant que des valeurs nulles (NaN)
- Passage de 1513 lignes à 799 lignes dans le DataFrame

bottleneck
fine wine spirit

Nettoyage des données
Première vu des données ERP

```
RangeIndex: 825 entries, 0 to 824
Data columns (total 5 columns):
 #   Column          Non-Null Count  Dtype
---  ------          --------------  -----
 0   product_id      825 non-null    int64
 1   onsale_web      825 non-null    int64
 2   price           825 non-null    float64
 3   stock_quantity  825 non-null    int64
 4   stock_status    825 non-null    object
dtypes: float64(1), int64(3), object(1)
```

	product_id	onsale_web	price	stock_quantity	stock_status
0	3847	1	24.2	0	outofstock
1	3849	1	34.3	0	outofstock
2	3850	1	20.8	0	outofstock
3	4032	1	14.1	0	outofstock
4	4039	1	46.0	0	outofstock
...	...	...	...	...	...
820	7203	0	45.0	30	instock
821	7204	0	45.0	9	instock
822	7247	1	54.8	23	instock
823	7329	0	26.5	14	instock
824	7338	1	16.3	45	instock

825 rows × 5 columns

Constats :

- Aucune valeur nulles (NaN)
- Pas d'erreur lexical

bottleneck
fine wine spirit

Nettoyage des données
Suppression des valeurs nulles sur la clé

```
print(dataerp.isnull().sum())
```

```
product_id        0
onsale_web        0
price             0
stock_quantity    0
stock_status      0
dtype: int64
```

- Aucune valeur nulles (NaN)

```
dataerp.loc[dataerp['product_id'].duplicated(keep=False),:]
```

product_id	onsale_web	price	stock_quantity	stock_status

- Aucun doublon

bottleneck
fine wine spirit

Nettoyage des données
Première vu des données de liaison

```python
#Consulter le nombre de colonnes
print(f"le data set de liaison contien {df_liaison.shape[1]} colonnes")
#La nature des données dans chacune des colonnes
print(f" la nature des données dans chacune des colonnes est : ")
print (df_liaison.dtypes,"\n")
#Le nombre de valeurs présentes dans chacune des colonnes
print("Nombre de valeurs non nulles dans chaque colonne :")
print(df_liaison.count(), "\n")
```

```
le data set de liaison contien 2 colonnes
 la nature des données dans chacune des colonnes est :
id_web        object
product_id     int64
dtype: object

Nombre de valeurs non nulles dans chaque colonne :
id_web        734
product_id    825
dtype: int64
```

- ► Aucune valeur nulles (NaN) dans la colonne ‘product_id’
- ► Des valeurs nulles (NaN) dans la colonne ‘id_web’

bottleneck
fine wine spirit

Nettoyage des données
Correction du nom de la colonne érroné

```
 #   Column                Non-Null Count  Dtype
---  ------                --------------  -----
 0   id_web                1428 non-null   object
```

```
df_liaison = df_liaison.rename(columns = {'id_web':'sku'})
df_liaison
```

```
 #   Column                Non-Null Count  Dtype
---  ------                --------------  -----
 0   sku                   1428 non-null   object
```


bottleneck
fine wine spirit

Nettoyage des données
Vérification et traitement des valeurs nulles (NaN) et des doublons

```
#Les valeurs de la colonne "id_web" OU "product_id" sont-elles toutes uniques?
df_liaison.isna().sum()
```

```
sku           91
product_id     0
dtype: int64
```

Il manque 91 valeurs pour "sku". Les produits concernés ne pourront pas être rapprochés de leur(s) vente(s), il est donc inutile de les conserver.

```
df_liaison = df_liaison.dropna().reset_index(drop=True)
```

Pour finir, nous nous assurons que les codes "product_id" et "sku" ne contiennent pas de doublons.

```
# Test des doublons sur le code produit et le SKU
test_doublons = df_liaison.duplicated(subset=['product_id', 'sku']).sum()

print(f'Il y a {test_doublons} doublons dans les codes produits et les SKU.')

del test_doublons
```

```
Il y a 0 doublons dans les codes produits et les SKU.
```

- Suppression des 91 valeurs nulles (NaN)
  - Aucun doublon

bottleneck
fine wine spirit

Fusion de données
Premier merge

- Premier merge entre les données de liaison et les données ERP

```
#Fusion des fichiers df_erp et df_liaison
df_merge = pd.merge((df_erp), (df_liaison), on = ['product_id'], how = 'outer' ,indicator=True)
```

	product_id	onsale_web	price	stock_quantity	stock_status	purchase_price	stock_quantity_corrigé	sku	_merge
19	4055	0	86.1	0	outofstock	37.88	0	NaN	left_only
49	4090	0	73.0	0	outofstock	33.79	0	NaN	left_only
50	4092	0	47.0	0	outofstock	25.25	0	NaN	left_only
119	4195	0	14.1	0	outofstock	7.36	0	NaN	left_only
131	4209	0	73.5	0	outofstock	33.01	0	NaN	left_only
...	...	...	...	...	...	...	...	...	...
814	7196	0	31.0	55	instock	31.20	55	NaN	left_only
815	7200	0	31.0	6	instock	15.54	6	NaN	left_only
816	7201	0	31.0	18	instock	16.02	18	NaN	left_only
817	7203	0	45.0	30	instock	23.48	30	NaN	left_only
818	7204	0	45.0	9	instock	24.18	9	NaN	left_only

bottleneck
fine wine spirit

Fusion de données
Deuxième merge

- Deuxième merge entre le premier merge et les données WEB

```
#Fusionnez les datasets df_merge et df_web
df_final = pd.merge((df_merge), (df_web), on = ['sku'], how = 'outer', indicator=True)
df_final
```

eb	price	stock_quantity	stock_status	purchase_price	stock_quantity_corrigé	sku	total_sales	product_type	post_title	post_excerpt	post_type	_merge
.0	8.6	26.0	instock	4.22	26.0	38	10.0	Vin	Emile Boeckel Crémant Brut Blanc de Blancs	Ce Crémant est vif et délicat, gourmand et cro...	product	both
.0	41.0	11.0	instock	20.12	11.0	41	6.0	Vin	Marcel Windholtz Eau de Vie de Marc de Gewurzt...	Les eaux de vie naissent d'une subtile alchimi...	product	both
.0	39.0	123.0	instock	24.86	123.0	304	8.0	Champagne	Champagne Gosset Grande Réserve	Le nez, ouvert et expressif, évoque les fruits...	product	both
.0	59.9	13.0	instock	27.18	13.0	523	0.0	Cognac	Cognac Normandin Mercier VFC	Issus des meilleurs crus de Grande et de Petit...	product	both
.0	22.5	76.0	instock	13.78	76.0	531	8.0	Champagne	Champagne Petit Lebrun & Fils Blanc de Bla...	Cuvée bien équilibrée à la fois vive et souple...	product	both
...	...	...	...	...	...	...	...	...	...	...	...	...
.0	31.0	55.0	instock	31.20	55.0	NaN	NaN	NaN	NaN	NaN	NaN	left_only
.0	31.0	6.0	instock	15.54	6.0	NaN	NaN	NaN	NaN	NaN	NaN	left_only
.0	31.0	18.0	instock	16.02	18.0	NaN	NaN	NaN	NaN	NaN	NaN	left_only
.0	45.0	30.0	instock	23.48	30.0	NaN	NaN	NaN	NaN	NaN	NaN	left_only
.0	45.0	9.0	instock	24.18	9.0	NaN	NaN	NaN	NaN	NaN	NaN	left_only

bottleneck
fine wine spirit

Analyse des données
Calcul du CA par produit et du CA total

```
##############################
# Calculer le CA du site web #
##############################

#Créez une colonne calculant le CA par article

df_final["ca_par_article"] = df_final["price"] * df_final["total_sales"]
df_final


#Calculez la somme de la colonne "ca_par_article"


ca_total = df_final["ca_par_article"].sum()
ca_col_total = round (ca_total , 2)

print (f" le chiffre d'affaire total est de {ca_total} euros ")


#Ce résultat correspond au chiffre d'affaire du site web
df_final
```

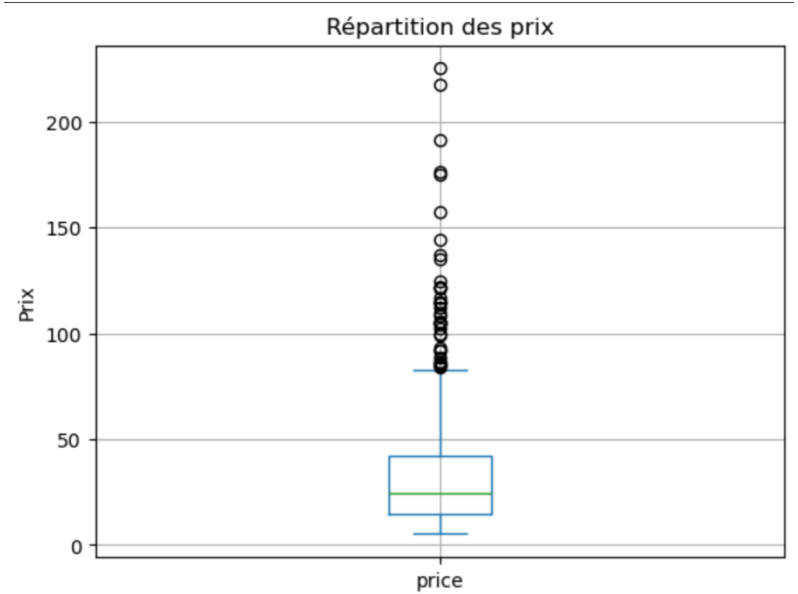
```
le chiffre d'affaire total est de 143680.1 euros
```

_quantity	stock_status	purchase_price	stock_quantity_corrigé	sku	total_sales	product_type	post_title	post_excerpt	post_type	_merge	ca_par_article
26.0	instock	4.22	26.0	38	10.0	Vin	Emile Boeckel Crémant Brut Blanc de Blancs	Ce Crémant est vif et délicat, gourmand et cro...	product	both	86.0
11.0	instock	20.12	11.0	41	6.0	Vin	Marcel Windholtz Eau de Vie de Marc de Gewurzt...	Les eaux de vie naissent d'une subtile alchimi...	product	both	246.0
123.0	instock	24.86	123.0	304	8.0	Champagne	Champagne Gosset Grande Réserve	Le nez, ouvert et expressif, évoque les fruits...	product	both	312.0
13.0	instock	27.18	13.0	523	0.0	Cognac	Cognac Normandin Mercier VFC	Issus des meilleurs crus de Grande et de Petit...	product	both	0.0
							Champagne Petit	Cuvée bien			

Chiffre d'affaire total
143680.1 euros €

bottleneck
fine wine spirit

Analyse des données
Analyse des prix des produits

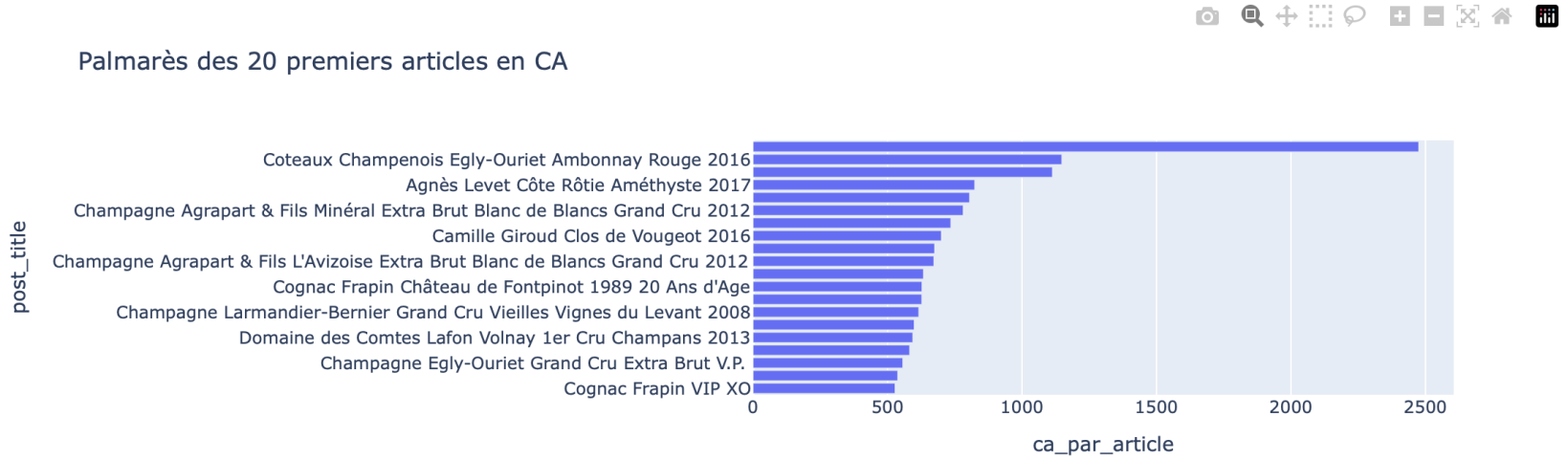


- Moyenne : 32.35
- Mediane : 24.4

- Q1 : 14.6
- Q3 : 42.0
- IQR : 27.4
- Min = 5.2
- Min = 84.0

- 50% des prix sont comprit entre 14€ et 42€
- 75% des prix sont comprit entre 5,2€ et 42€
- Les Outlier sont supérieur à 84€

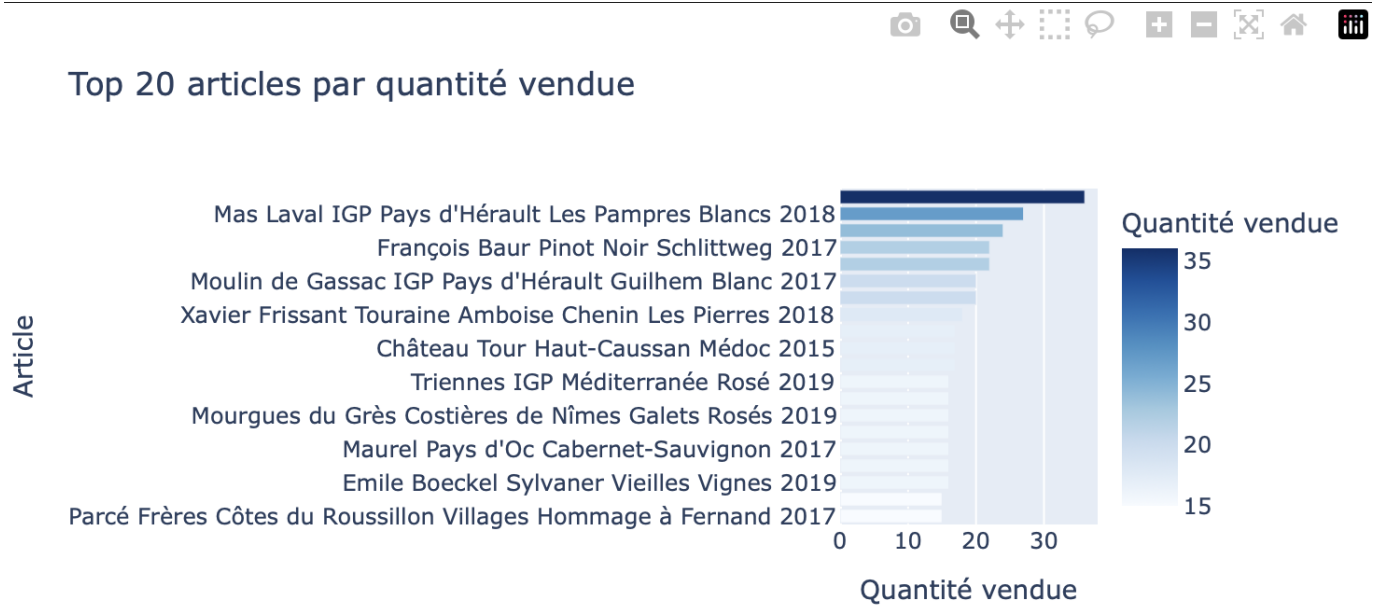
Analyse des données
Analyse des prix des produits



Nombre d'articles représentant 80% du CA : 434 Ces articles représentent 52.67% du catalogue total.

bottleneck
fine wine spirit

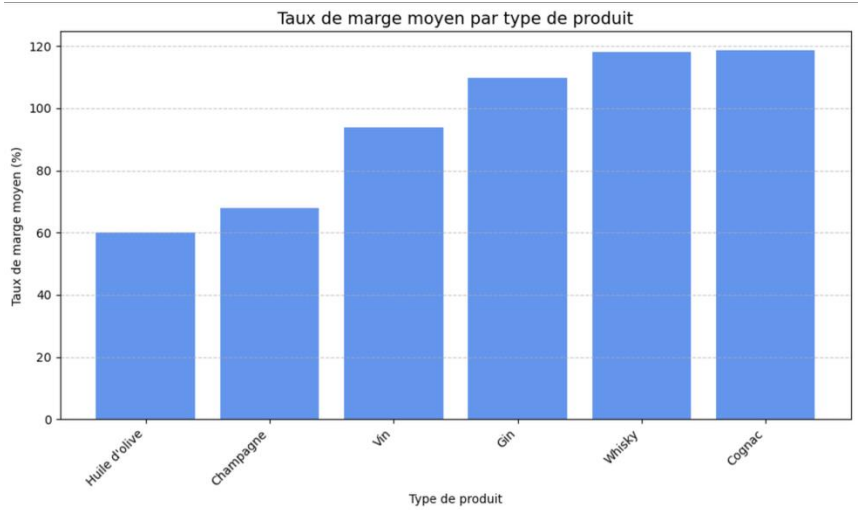
Analyse des données
Analyse des Quantités vendues des produits



Analyse des données
Taux de marge moyen par type de produit

1. ✅ Les spiritueux (Cognac, Whisky, Gin) sont les produits les plus rentables. Il peut être judicieux d'en renforcer la promotion ou les stocks.

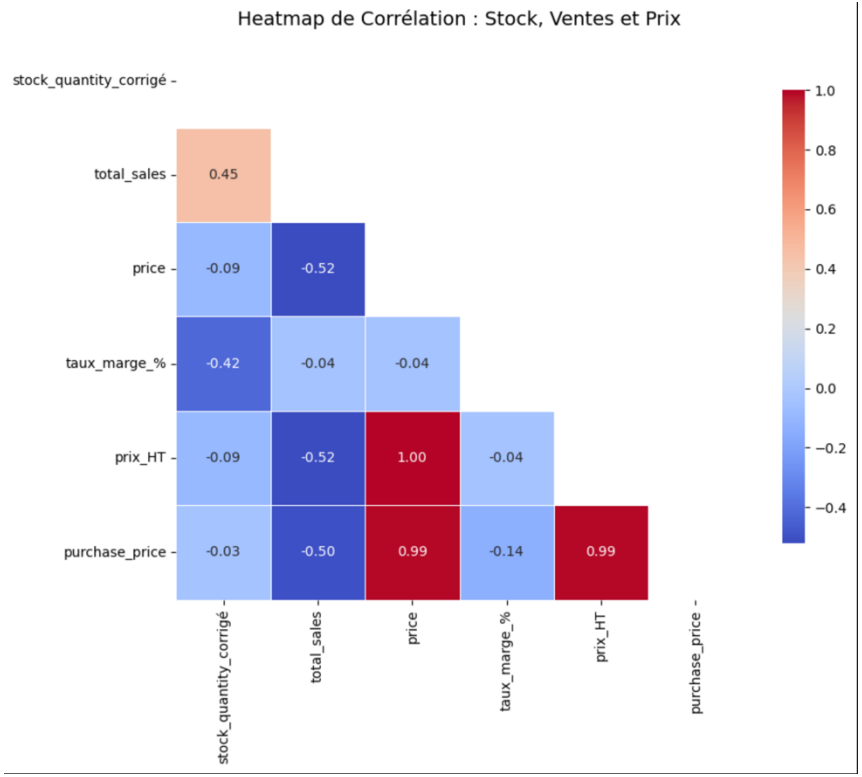
2. ⚠️ Huile d'olive et Champagne ont une marge relativement faible. Cela peut indiquer :
   - Des coûts d'achat élevés.
   - Une fixation des prix de vente trop basse.
   - Une faible valeur ajoutée ou concurrence accrue.



3. 🔄 Le vin est intermédiaire : rentable mais peut être optimisé

bottleneck
fine wine spirit

Analyse des données
Heatmap de corélation



bottleneck
fine wine spirit

Synthèse sur le nettoyage

- Fichier WEB :
- ► Beaucoup de données manquantes
- ► Des colonnes non-utilisé
- ► Des doublon pour chaque produits
- ► Des erreurs lexical à revoir

- Fichier ERP :
- ► Faire attention au cohésion entre la quantité et le statut des stocks

- Fichier Liaison :
- ► Quelques valeurs nulles

bottleneck
fine wine spirit

Synthèse sur l’analyse

- ► Chiffre d’affaire de 143680.1€
- ► 50% des prix sont compris entre 14€ et 42€
- ► Les Outlier sont des valeurs Extrême
- ► 51,46% des articles du catalogue produisent 80% du chiffre d’affaires de la boutique
- ►Les produits du type Gin , Whisky et Cognac sont les plus rentables

bottleneck
fine wine spirit