Optimisez la gestion du stock d’une boutique en nettoyant ses données

Soumare Djibril
Business Intelligence Analyst

07/07/2023

Analyses Exploratoires des Données

- *Analyse exploratoire de chaque variable du fichier erp.xlsx*

```
df_erp.info()

<class 'pandas.core.frame.DataFrame'>
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)
```

Variable	maximum	minimum
price	225	5,2
stock_quantity	578	0
onsale_web	717 online	108 offline

- *Analyse exploratoire de chaque variable du fichier caracteristiques.xlsx*

```
df_caracteristiques.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 611 entries, 0 to 610
Data columns (total 13 columns):
 #   Column                  Non-Null Count  Dtype
---  ------                  --------------  -----
 0   post_name               611 non-null    object
 1   poids                   611 non-null    object
 2   Région                  586 non-null    object
 3   Domaine                 577 non-null    object
 4   Appellation             559 non-null    object
 5   Couleur                 566 non-null    object
 6   Cépage                  571 non-null    object
 7   Millésime               541 non-null    float64
 8   Garde                   569 non-null    object
 9   Contenance              611 non-null    object
 10  Degré d'alcool          586 non-null    object
 11  Température dégustation 574 non-null    object
 12  Alliance mets           574 non-null    object
```

- *Analyse exploratoire de chaque variable du fichier we.xlsx*

```
df_liaison.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 825 entries, 0 to 824
Data columns (total 2 columns):
 #   Column      Non-Null Count  Dtype
---  ------      --------------  -----
 0   product_id  825 non-null    int64
 1   id_web      734 non-null    object
dtypes: int64(1), object(1)
```

Analyses Exploratoires des Données

- *Analyse exploratoire de chaque variable du fichier web.xlsx*

valeurs respectant pas la règle de codification dans (sku)

```
0       bon-cadeau-25-euros
797                 13127-1
1209    bon-cadeau-25-euros
1511                13127-1
Name: sku, dtype: object
```

Identification de valeur en double dans l'indentifiant web (uk)

```
df_web.loc[df_web['sku'].duplicated(keep=False),:]
```

1513 rows × 28 columns

Identification de colonnes à conserver

```
df_web.columns[df_web.count()!=0]

Index(['sku', 'virtual', 'downloadable', 'rating_count', 'average_rating',
       'total_sales', 'tax_status', 'post_author', 'post_date',
       'post_date_gmt', 'post_title', 'post_excerpt', 'post_status',
       'comment_status', 'ping_status', 'post_name', 'post_modified',
       'post_modified_gmt', 'post_parent', 'guid', 'menu_order', 'post_type',
       'post_mime_type', 'comment_count'],
```

Identification de colonnes à supprimer

```
df_web.columns[df_web.count()==0]

Index(['tax_class', 'post_content', 'post_password', 'post_content_filtered']
```

```
df_web.info()

<class 'pandas.core.frame.DataFrame'>
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   datetime64[ns]
 10  post_date_gmt         1430 non-null   datetime64[ns]
 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   datetime64[ns]
 20  post_modified_gmt     1430 non-null   datetime64[ns]
 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: datetime64[ns](4), float64(10), int64(3), object(11)
```

Fusion ou consolidations des données

Cette opération consiste à regrouper l'ensemble de datasets en une seule dont DataFrame finale est nommée 'df_merge'.

Jonction du fichier df_erp et df_liaison

df_erp et df_liaison sont reliées par une variable commune, **product_id**,

```
df_erp_liaison=pd.merge(df_erp,df_liaison,on='product_id')
df_erp_liaison.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 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   id_web          734 non-null    object
```

Jonction entre df_erp_liaison et df_web

La jointure entre ces deux df est réalisée à travers leur clé primaire (id_web et sku)dont une correspondance des valeurs existent dans les deux.

```
df_erp_liaison_web=pd.merge(df_erp_liaison, df_web, on="id_web", how="left")
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 825 entries, 0 to 824
Data columns (total 28 columns):
```

Jonction entre df_erp_liaison_web et df_caracteristiques

Le regroupement du df_caracteristiques avec le fichier df_merge est effectuée par une variable identique (post_name).

```
df_erp_liaison_web_caracteristiques=pd.merge(df_erp_liaison_web, df_caracteristiques, on='post_name', how='left')
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 825 entries, 0 to 824
Data columns (total 40 columns):
```

Analyses univariées du prix

Utilisation de la fonction describe de Pandas pour l'étude des mesures de dispersions

```
df_erp_liaison_web_caracteristiques['price'].describe()
```

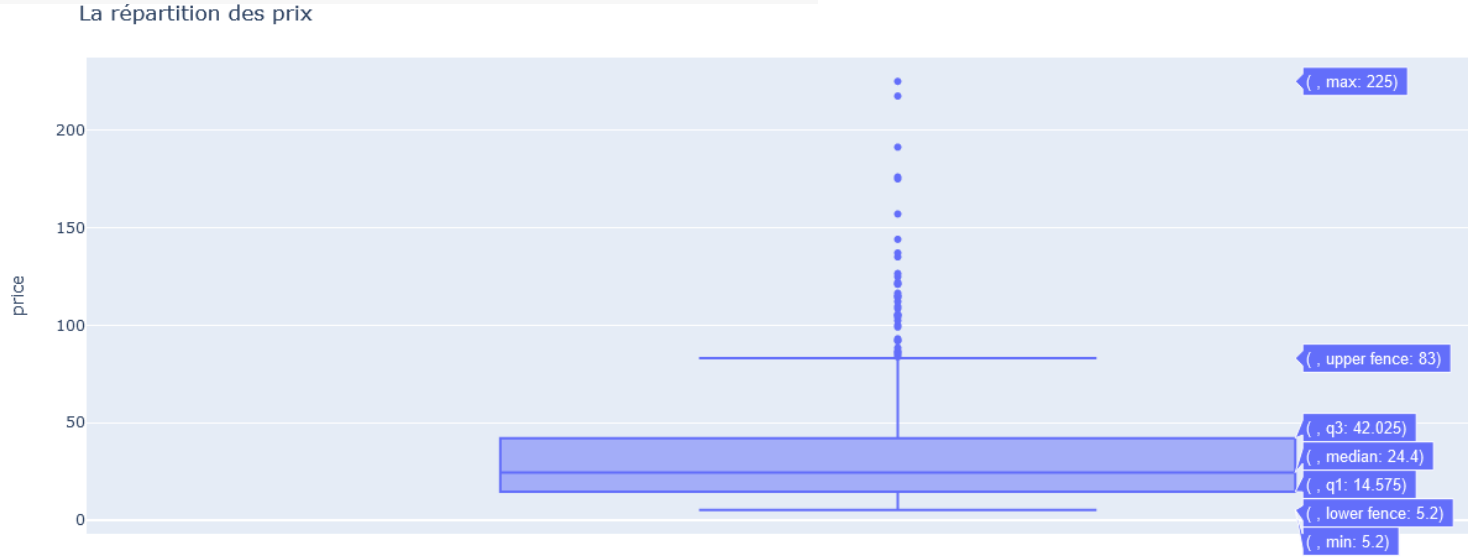
```
count    825.000000
mean      32.415636
std       26.795849
min        5.200000
25%       14.600000
50%       24.400000
75%       42.000000
max      225.000000
```

Au regard de ce résultat, on constate que le prix moyen de 50% des articles est inferieur au à la moyenne de prix de l'ensemble des produits et pour 75% des articles ont un prix moyen supérieur à la moyenne. Ce la signifie que les prix de produits sont très hétérogènes.

Analyses univariées du prix

Utilisation d'une boite à moustache de la répartition des prix avec plotly express

```
import plotly.express as px
figure=px.box(df_erp_liaison_web_caracteristiques, y='price')
figure.update_layout(title='La répartition des prix')
figure.show()
```

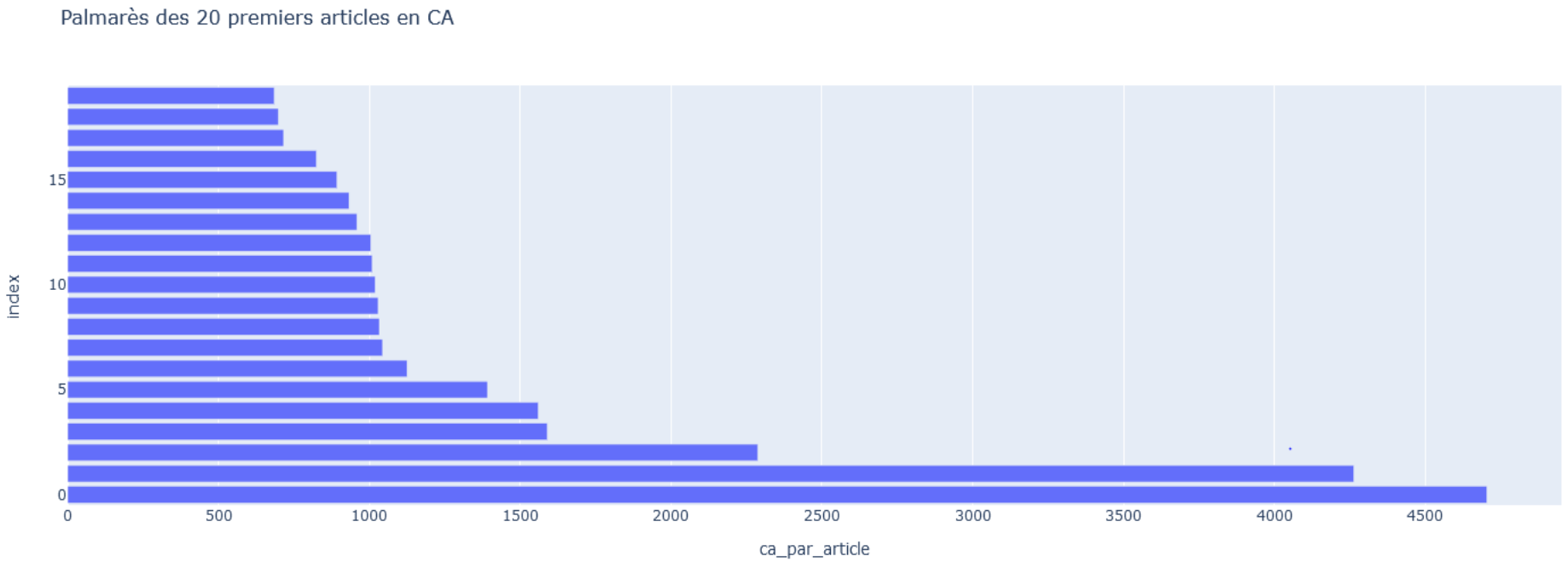


Observation de cette figure permet de ressortir que les prix qui peuvent être considérés comme des valeurs atypiques.
Les produits ayant le prix supérieur à 83 sont au-delà du seuil de 75% de produits qui ont un prix moyen de 42.

Analyses univariées du CA

Les 20 premier articles un total CA de 78318,6

```
top_20_articles=df_erp_liaison_web_caracteristiques.head(20)
#Graphique en barre des 20 premiers articles avec plotly express
fig = px.bar(top_20_articles, x='ca_par_article', title='Palmarès des 20 premiers articles en CA')
fig.show()
```



Point sur les compétences apprises

- *Dans le cadre de ce projet , je puis decouvrir plusieurs connaissances sur :*
- *connaitre nombre des valeurs nulls*
- *Identifier et supprimer les valeurs en double*
- *Selectionner et supprimmer les lignes et les colonne*
- *Créer les nouvelles variables*
- *Faire la jointure entre les jeux de données.*

- *Quelques difficultés ont été rencontrées lors de la réalisation des différentes étapes de la mission,*
- *Identification de type d'encodage pour charger le fichier CSV*
- *Créer de nouveaux champs et faire directement les calculs*
- *Créer des graphiques avec méthode plotly express*