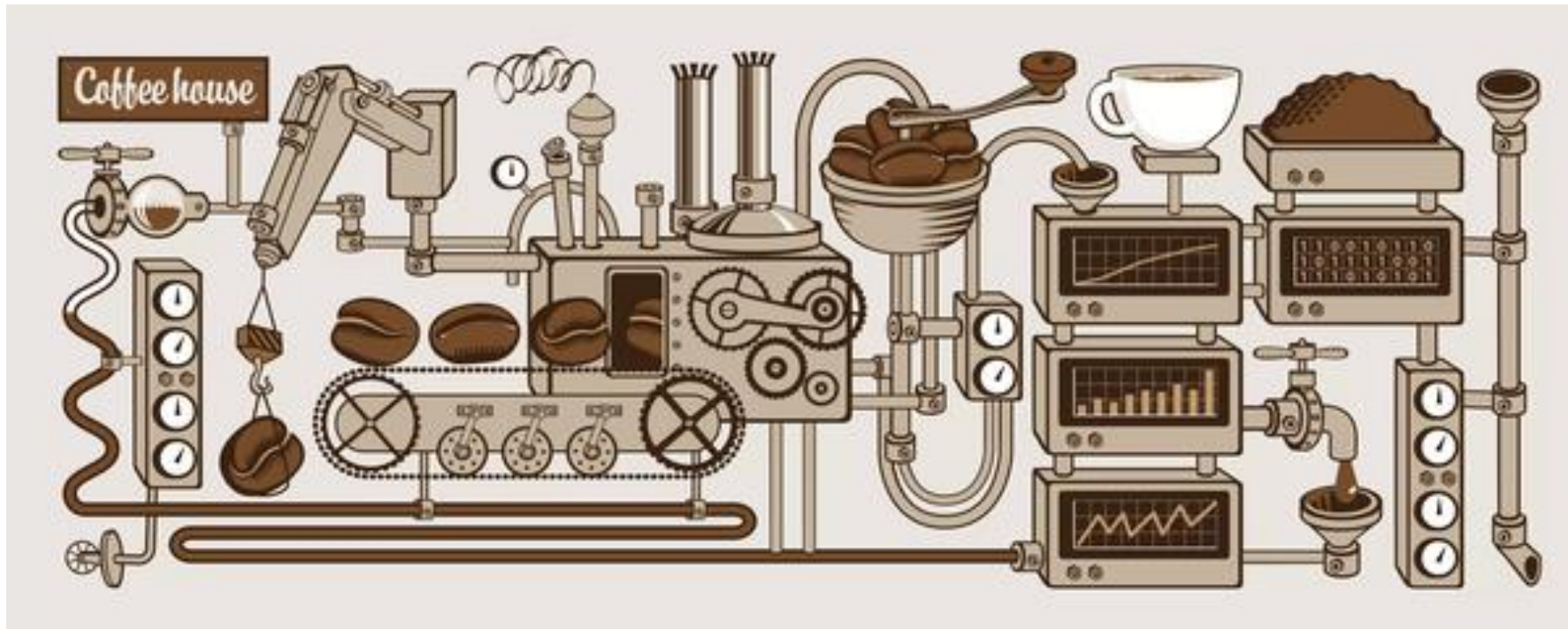


Projet 4 : Analysez les ventes de votre entreprise



Mise en contexte

- Ventes
- Listes clients
- Produits



3 CSV

- Partie nettoyage
- Partie analyse de données
- Partie corrélations



3 Parties

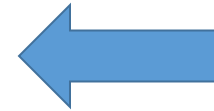
Nettoyage des données

Focus sur le DataFrame
customers



```
print(customers.client_id.describe(), "\n")
print(customers.sex.describe(), "\n")
print(customers.birth.describe(), "\n")

#aucun doublon parmi les clients
#Il n'y a que des F et des M parmi les sexes
#Toutes les années sont comprises entre 1929 et 2004
```



Constatations grâce à
describe

```
count      8623
unique      8623
top         c_5235
freq         1
Name: client_id, dtype: object
```

```
count      8623
unique         2
top          f
freq       4491
Name: sex, dtype: object
```

```
count      8623.000000
mean       1978.280877
std         16.919535
min        1929.000000
25%        1966.000000
50%        1979.000000
75%        1992.000000
max        2004.000000
Name: birth, dtype: float64
```

Nettoyage des données

```
customers["age"]=2023-customers.birth  
print(customers.age.describe(),"\n")
```



Création colonne
"age"

#Il n'y a aucun client mineur ou âgé de 100 ans ou plus

```
count      8623.000000  
mean        44.719123  
std         16.919535  
min         19.000000  
25%         31.000000  
50%         44.000000  
75%         57.000000  
max         94.000000  
Name: age, dtype: float64
```

Nettoyage des données

Focus sur le
DataFrame products



```
[5]: print(products.id_prod.describe(), "\n")
      print(products.price.describe(), "\n")
      print(products.categ.describe(), "\n")

      #Il n'y a pas de doublons parmi les id_prod
      #Il y a au moins un prix négatif, c'est une anomalie
```

```
count      3287
unique      3287
top         0_667
freq         1
Name: id_prod, dtype: object
```

```
count      3287.000000
mean        21.856641
std         29.847908
min         -1.000000
25%          6.990000
50%         13.060000
75%         22.990000
max         300.000000
Name: price, dtype: float64
```

```
count      3287.000000
mean         0.370246
std          0.615387
min          0.000000
25%          0.000000
50%          0.000000
75%          1.000000
max          2.000000
Name: categ, dtype: float64
```

Nettoyage des données

```
print(products.price[products.price<0])
print("\nIl y a un produit à prix négatif\n")
products.price=products.price[products.price>0]
products=products.dropna()
print(products.price[products.price<0])
```

```
731    -1.0
Name: price, dtype: float64
```

Il y a un produit à prix négatif

```
Series([], Name: price, dtype: float64)
```

```
print(products.categ[(products.categ==0) | (products.categ==1) | (products.categ==2)].count().sum())
print(products.categ.count().sum())
#Il y a 3286 résultats, tous les produits sont bien de catégorie 0, 1 ou 2
```

```
3286
3286
```

Nettoyage des données

```
df_merge=pd.merge(transactions,products,how="left")
df_produit_inconnu=df_merge[df_merge.price.isna()==True]
print(df_produit_inconnu.head())
print(df_produit_inconnu.groupby(df_produit_inconnu.id_prod).count())

print("\nLe produit 0_2245 a été acheté 103 fois alors qu'il n'est pas répertorié dans la liste des produits")
print("\n200 achats test ont été effectués sur le produit T_0 qui n'est pas répertorié dans la liste des produits")

products=products[products.id_prod!="0_2245"]
products=products.dropna()
products=products.reset_index()
```

	id_prod	date	session_id	client_id	price	\
1431	T_0	test_2021-03-01 02:30:02.237420	s_0	ct_1	NaN	
2365	T_0	test_2021-03-01 02:30:02.237446	s_0	ct_1	NaN	
2895	T_0	test_2021-03-01 02:30:02.237414	s_0	ct_1	NaN	
5955	T_0	test_2021-03-01 02:30:02.237441	s_0	ct_0	NaN	
6235	0_2245	2021-06-17 03:03:12.668129	s_49705	c_1533	NaN	

	categ
1431	NaN
2365	NaN
2895	NaN
5955	NaN
6235	NaN

	date	session_id	client_id	price	categ
id_prod					
0_2245	103	103	103	0	0
T_0	200	200	200	0	0

Le produit 0_2245 a été acheté 103 fois alors qu'il n'est pas répertorié dans la liste des produits

200 achats test ont été effectués sur le produit T_0 qui n'est pas répertorié dans la liste des produits

Nettoyage des données

Focus sur le
DataFrame
transactions

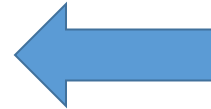


```
print(transactions.id_prod.describe(), "\n")
print(transactions.date.describe(), "\n")
print(transactions.session_id.describe(), "\n")
print(transactions.client_id.describe(), "\n")
```

*#On remarque une date marquée d'un "test" au début comme vu précédemment
#On remarque qu'un client a acheté 12855 produits*

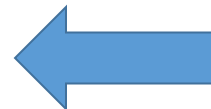
```
count      337016
unique      3266
top         1_369
freq        1081
Name: id_prod, dtype: object
```

```
count      337016
unique      336855
top    test_2021-03-01 02:30:02.237413
freq         13
Name: date, dtype: object
```



```
count      337016
unique    169195
top         s_0
freq        200
Name: session_id, dtype: object
```

```
count      337016
unique      8602
top         c_1609
freq       12855
Name: client_id, dtype: object
```




Nettoyage des données

```
print(transactions.date[transactions.date<"2021-01-01 00:00:00.000000"])\nprint(transactions.date[transactions.date>"2023-01-01 00:00:00.000000"])\nprint(transactions.date[transactions.date>"A"])\nprint("200 transactions étaient un test")
```




```
transactions.date=transactions.date[transactions.date>"2021-01-01 00:00:00.000000"]\ntransactions.date=transactions.date[transactions.date<"2023-01-01 00:00:00.000000"]\ntransactions=transactions.dropna()
```

Nettoyage des données




```
transactions_2=transactions.client_id
transactions_2=transactions_2.groupby(transactions_2).count()
transactions_2=transactions_2.sort_values()
print(transactions_2.tail()) #nombre d'achats effectués par client
#On repère que les clients c_4958, c_3454, c_6714 et c_1609 ont tous fait plus de 2500 achats en moins de 2 ans. Il faudra les
# retirer dans certaines analyses
```

```
client_id
c_7959      195
c_4958     2562
c_3454     3275
c_6714     4473
c_1609    12855
Name: client_id, dtype: int64
```



```
df_merge2=pd.merge(transactions,customers,how="left")
df_client_inconnu=df_merge2[df_merge2.sex.isna()==True]
print(df_client_inconnu)
#Tous les clients présent dans les transactions sont répertoriés dans le fichier client
```

```
Empty DataFrame
Columns: [id_prod, date, session_id, client_id, sex, birth, age]
Index: []
```

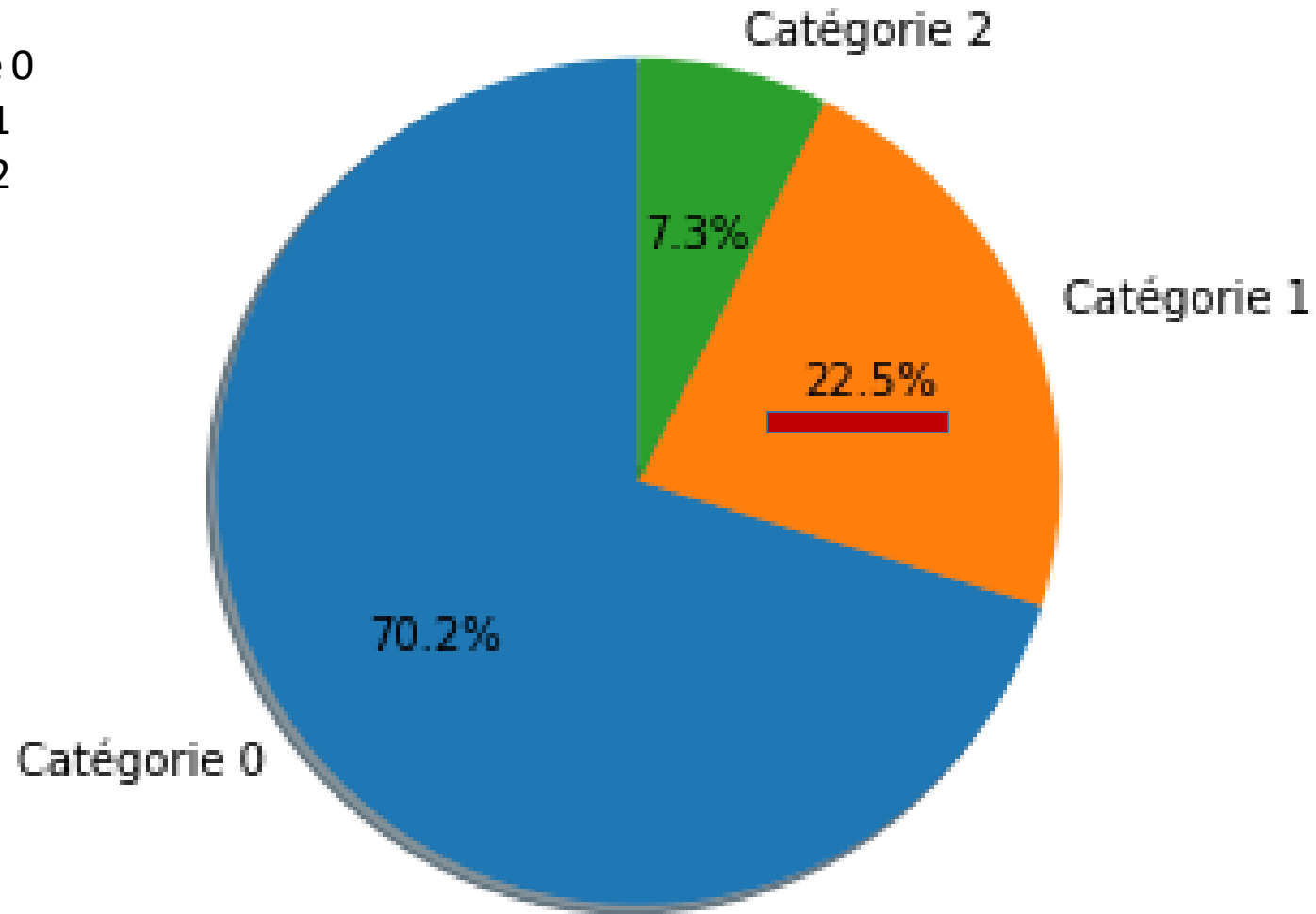


```
print(transactions.session_id.groupby(transactions.session_id>"s").count())
print(transactions.session_id.groupby(transactions.session_id<"u").count())
#Pas d'irrégularité dans les session_id
```

```
session_id
True      336816
Name: session_id, dtype: int64
session_id
True      336816
Name: session id. dtvne: int64
```

Analyse de données

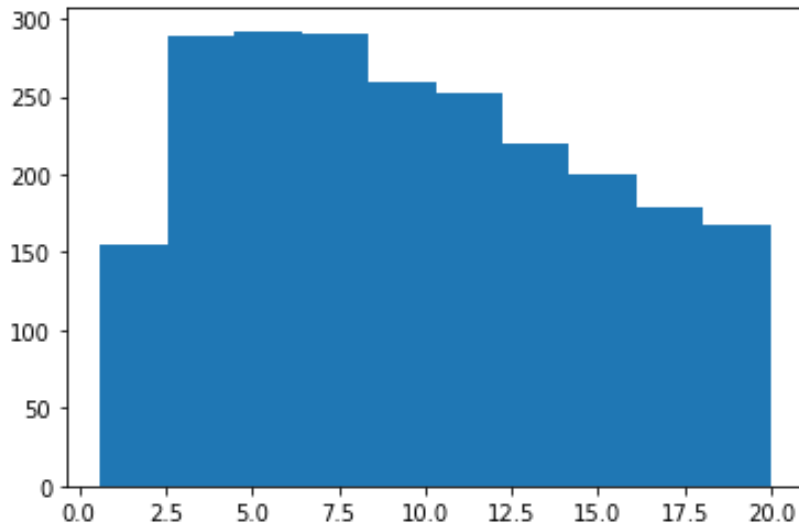
2308 produits de catégorie 0
739 produits de catégorie 1
239 produits de catégorie 2



Analyse de données



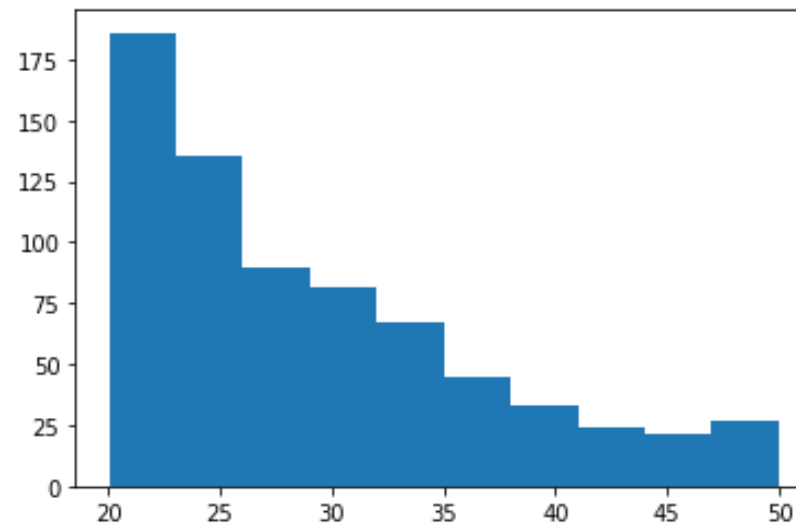
Produits en-dessous de 20 Euros



Moyenne des prix : 9.82
Médiane : 9.22
Mode : 4.99



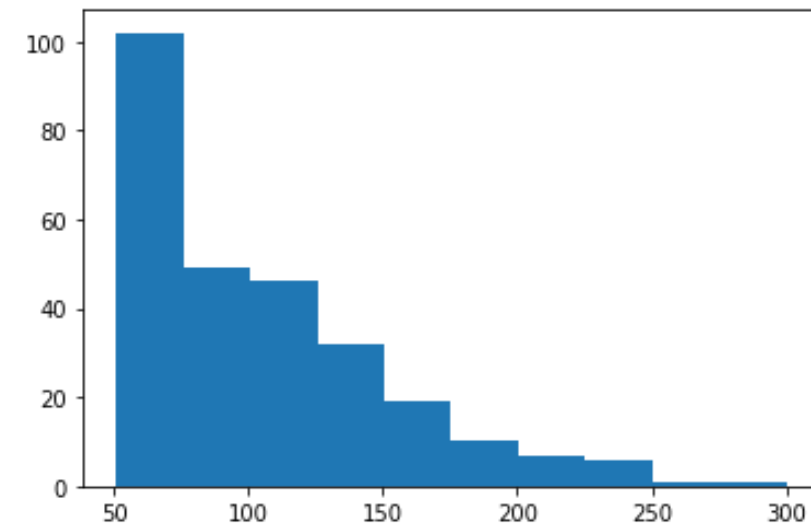
Produits entre 20 et 50 Euros



Moyenne des prix : 29.42
Médiane : 26.99
Mode : 22.99



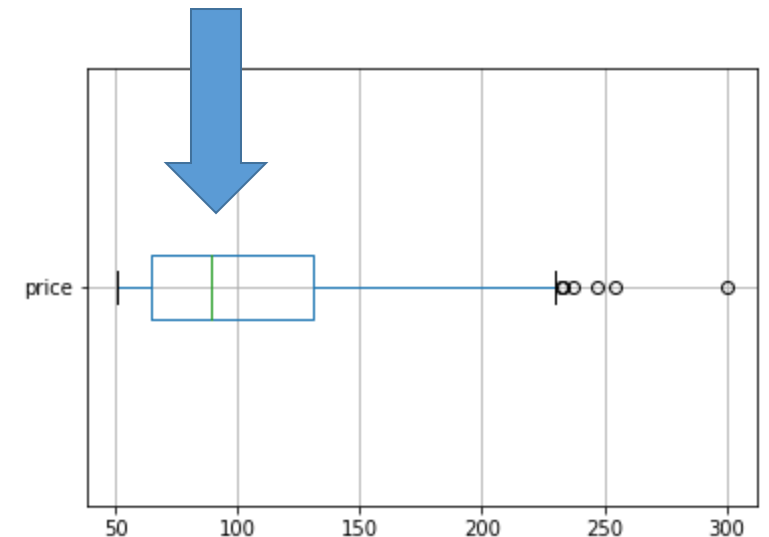
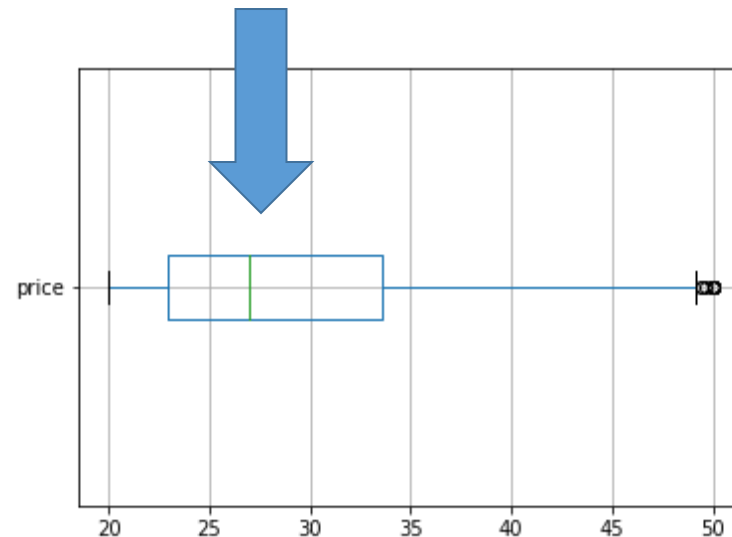
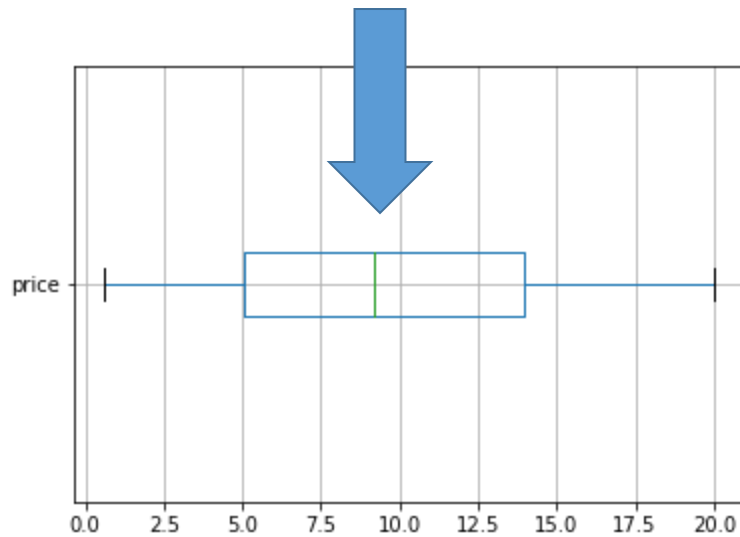
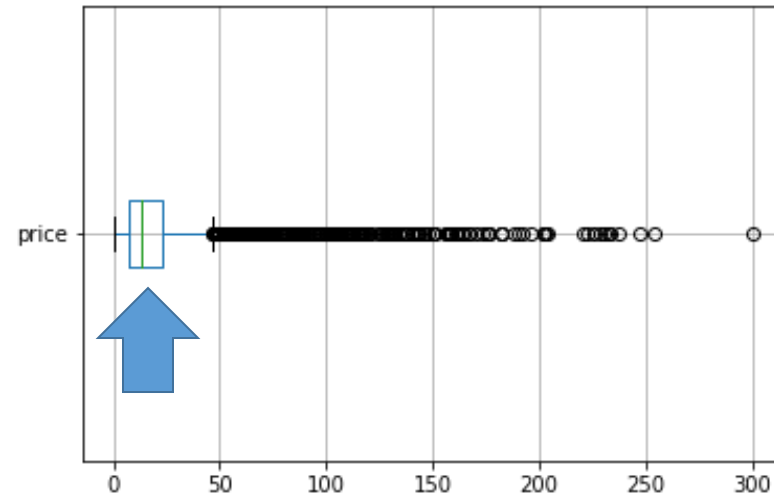
Produits au-dessus de 50 Euros



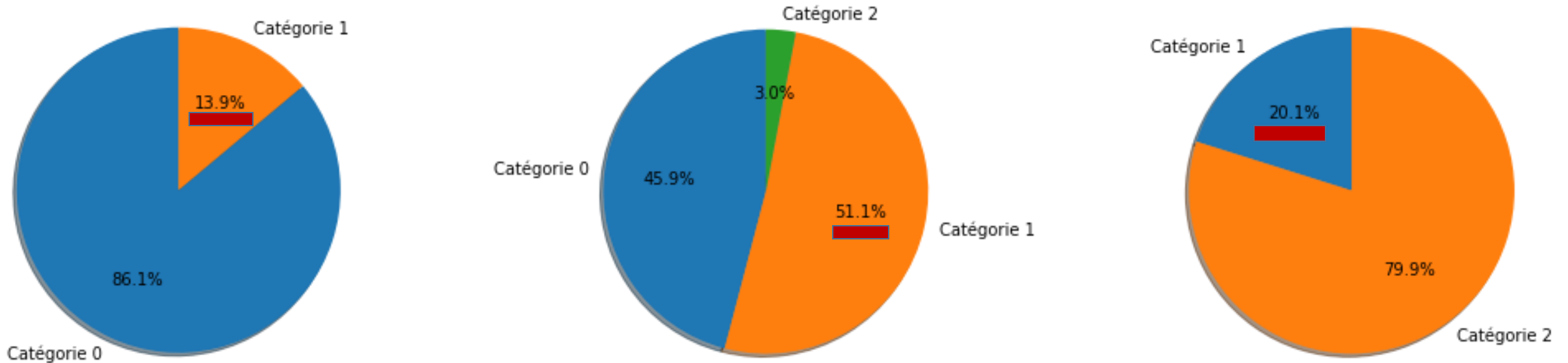
Moyenne des prix : 103.75
Médiane : 89.54
Mode : 50.99

Une majorité de produits en dessous de 20 Euros

Analyse de données



Analyse de données



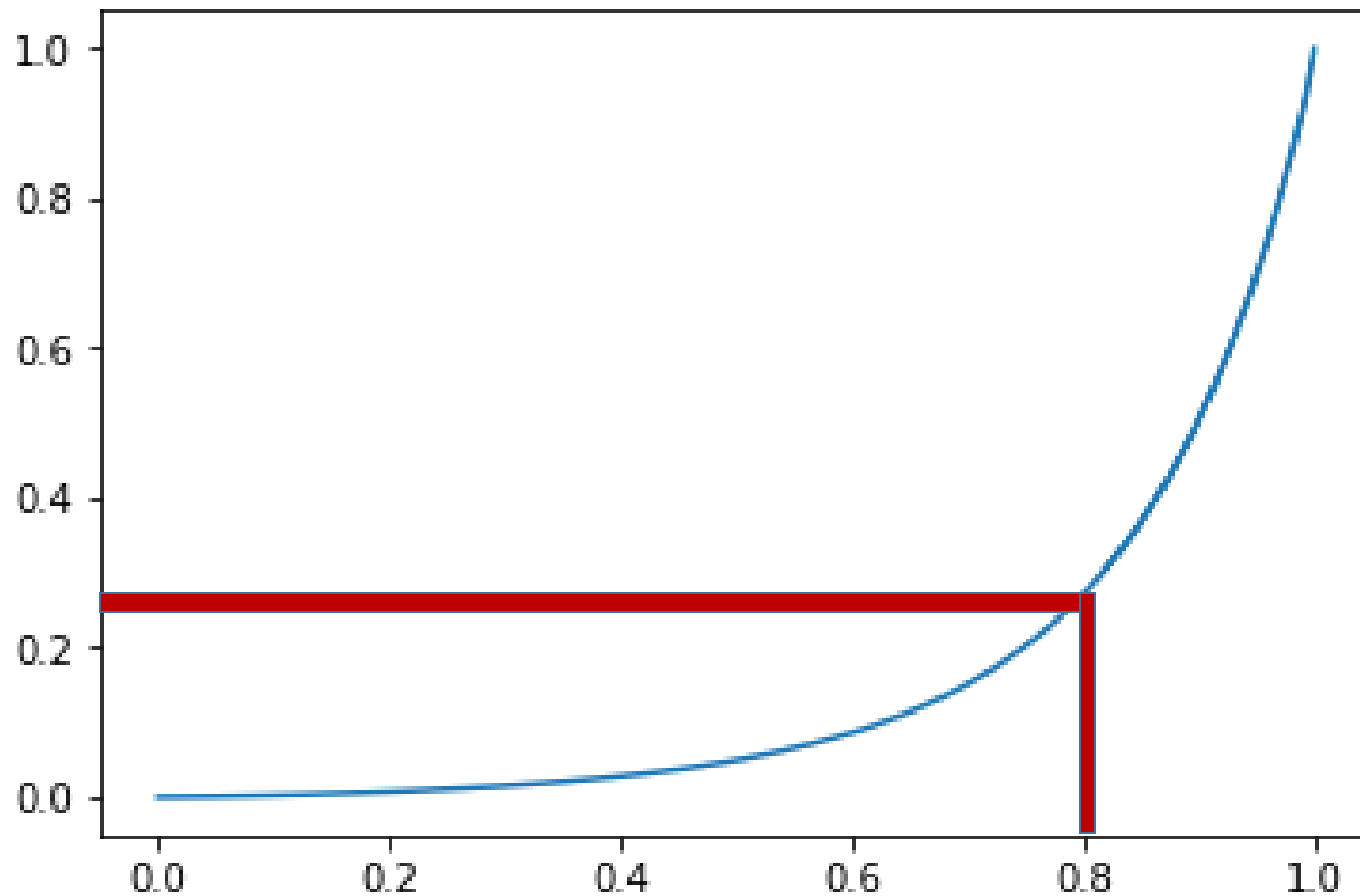
2308 produits de catégorie 0 dont 1982 à moins de 20Euros et 326 entre 20 et 50 Euros

739 produits de catégorie 1 dont 321 à moins de 20Euros, 363 entre 20 et 50 Euros et 55 au delà de 50 Euros

239 produits de catégorie 2 dont 21 entre 20 et 50 Euros et 218 au delà de 50 Euros

Analyse de données

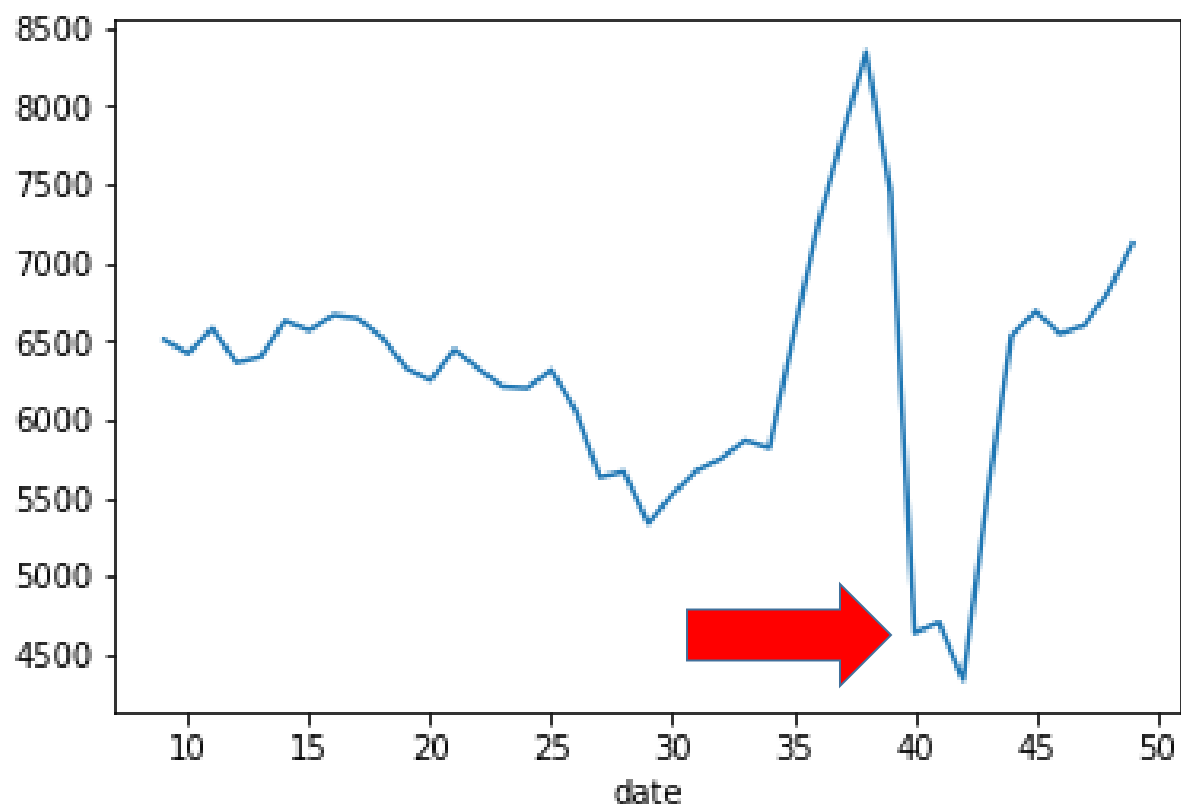
Indice de Gini :
0.6902366694430301



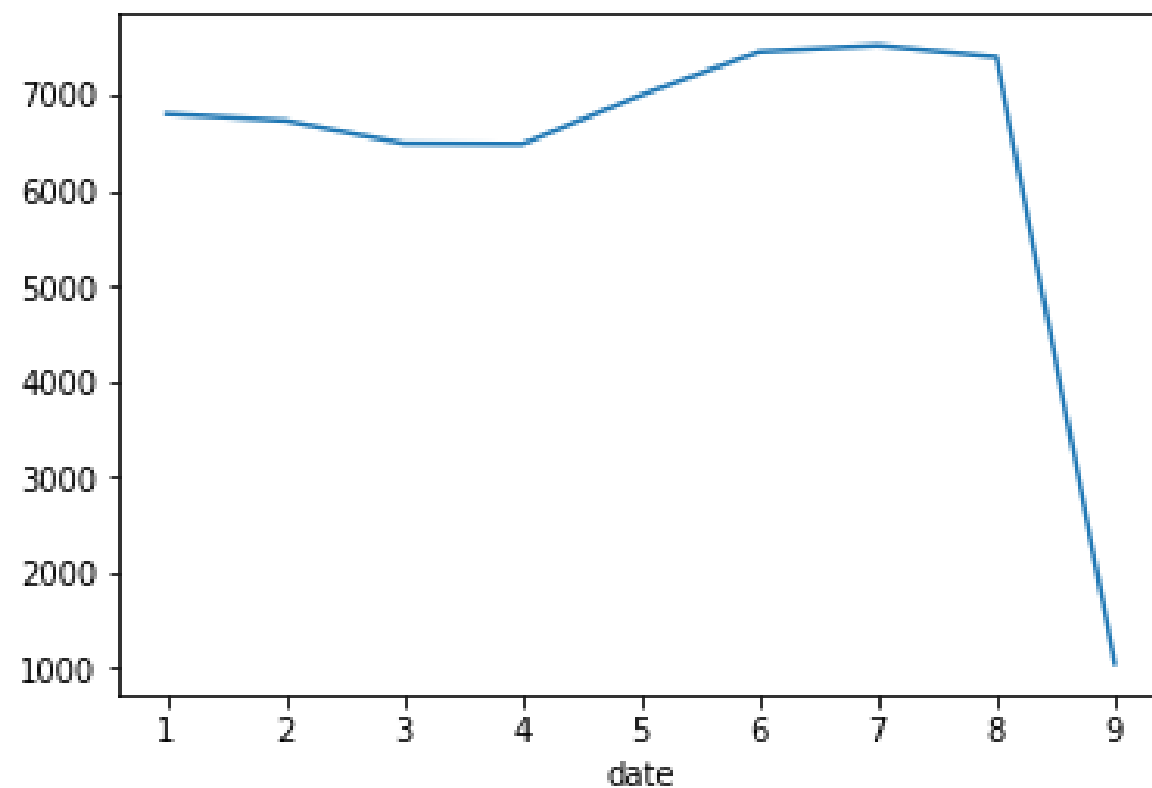
25% des produits représentent
80% des produits vendus

Analyse de données

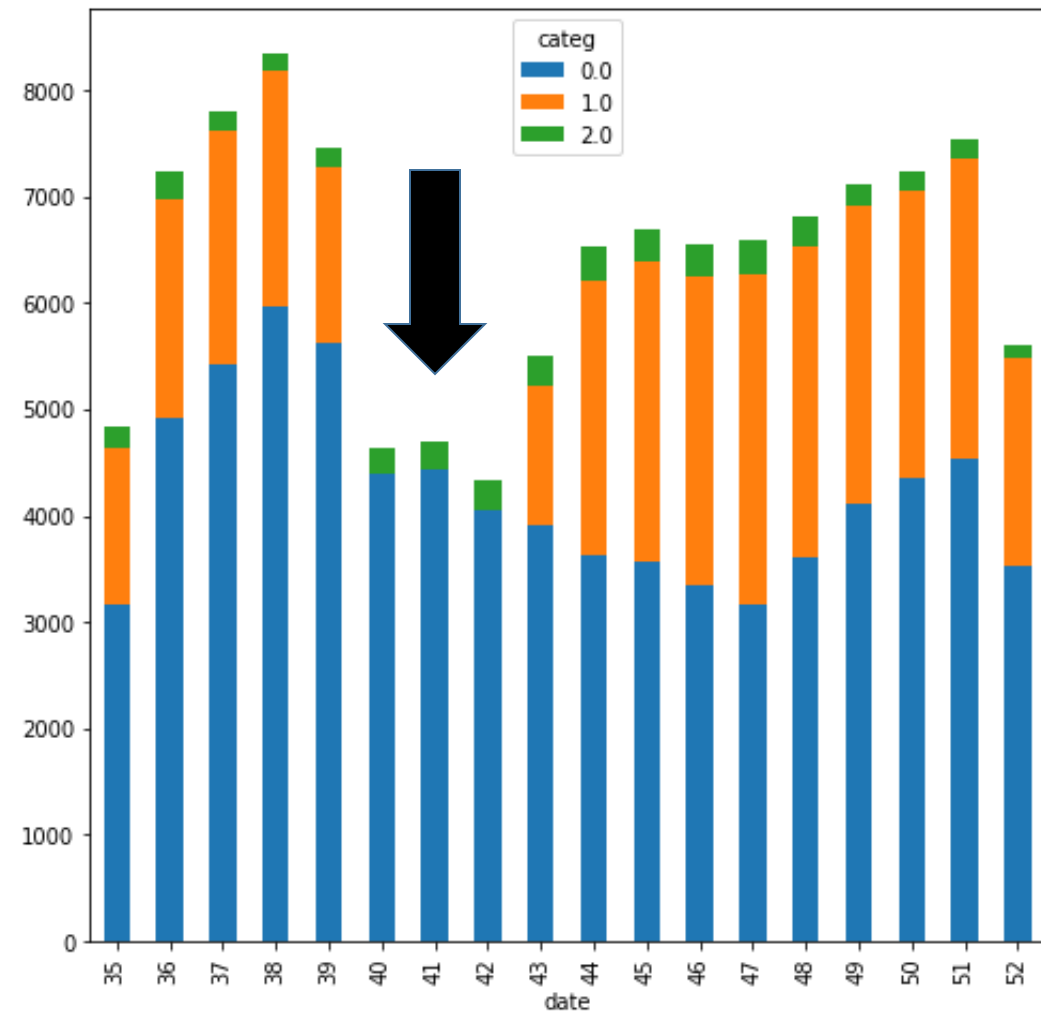
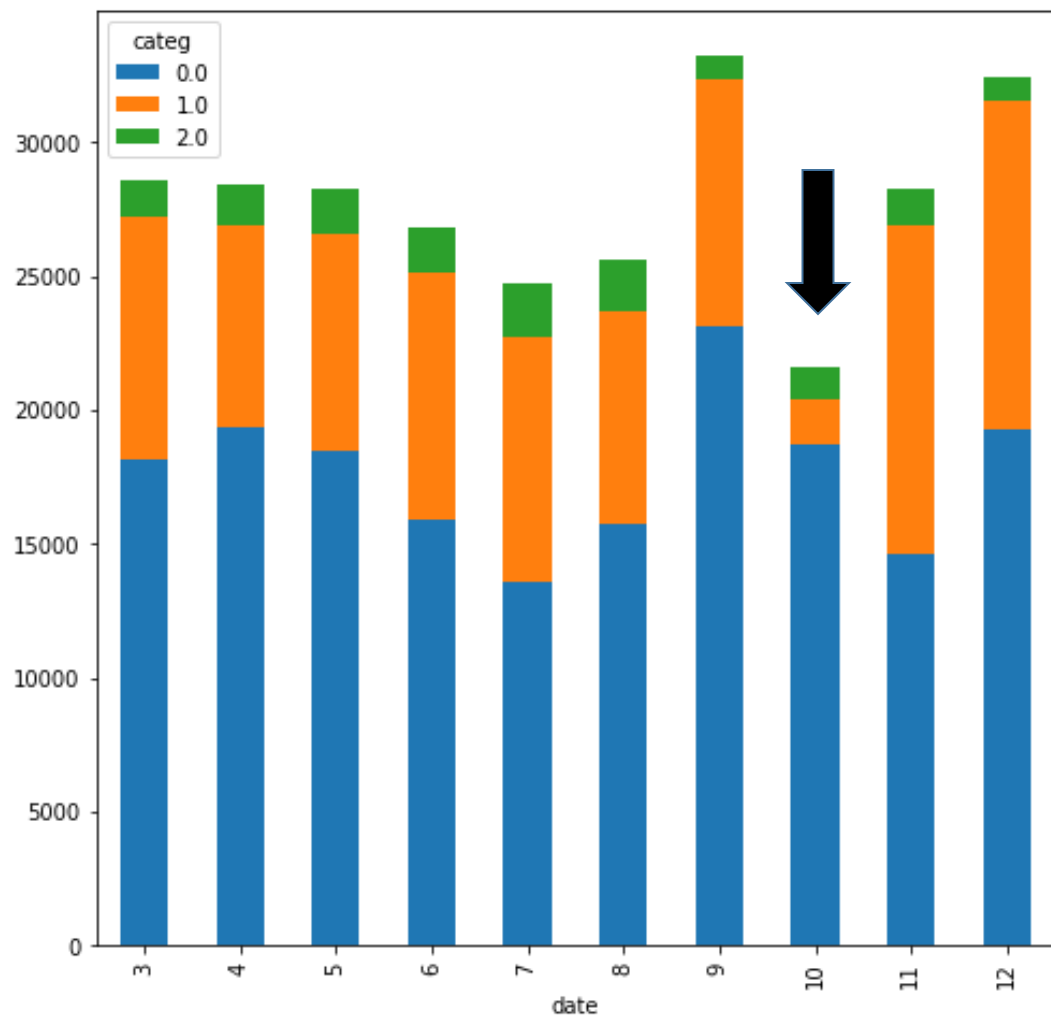
CA 2022



CA 2023

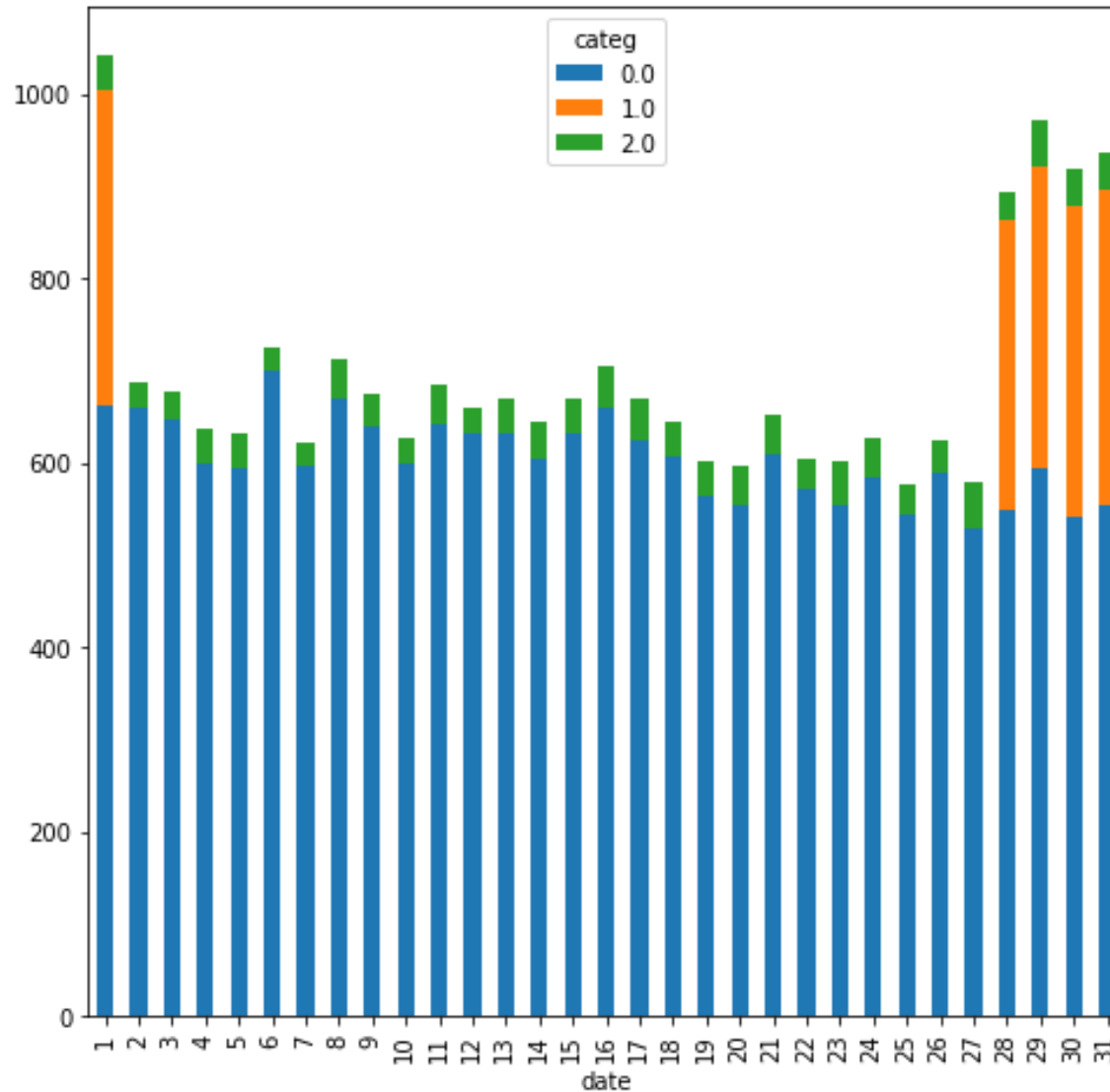


Analyse de données



Analyse de données

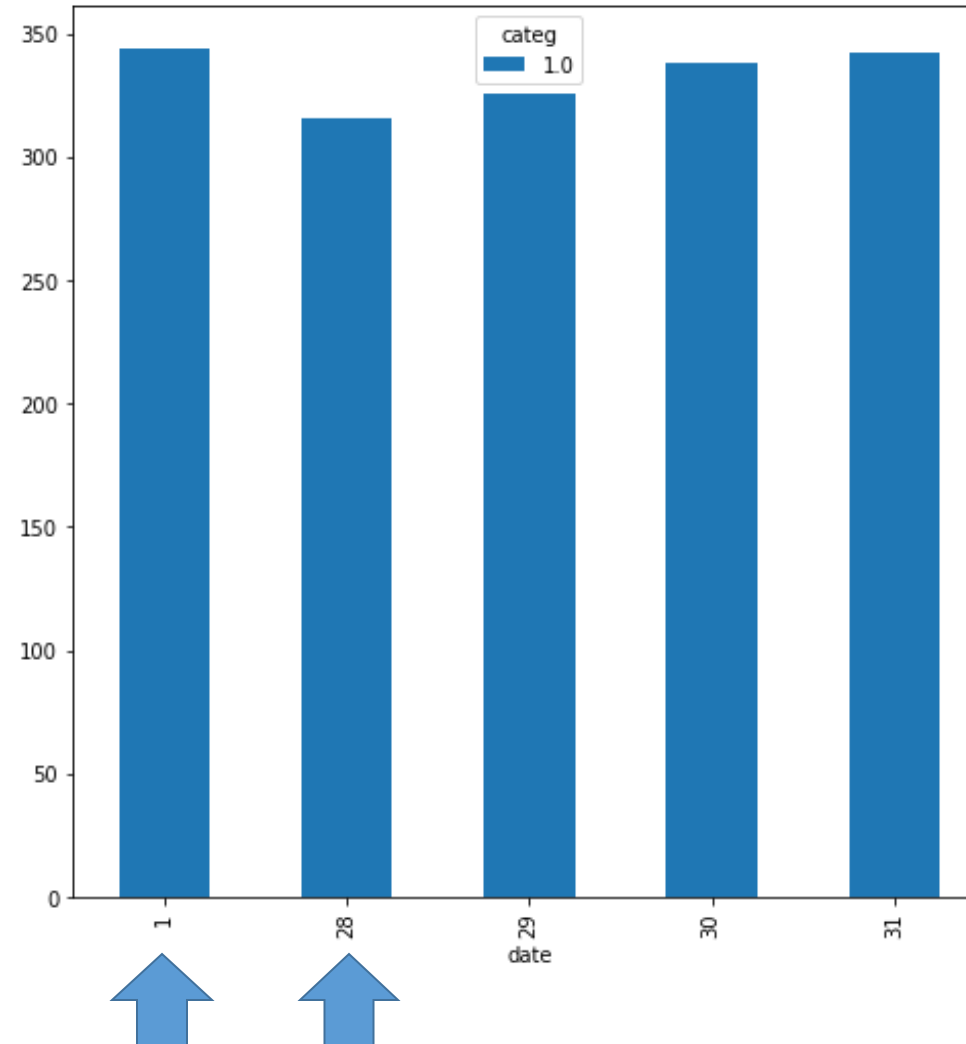
Focus sur les jours du mois
d'octobre



Aucune vente de produits
de catégorie 1 du 2 au 27
octobre

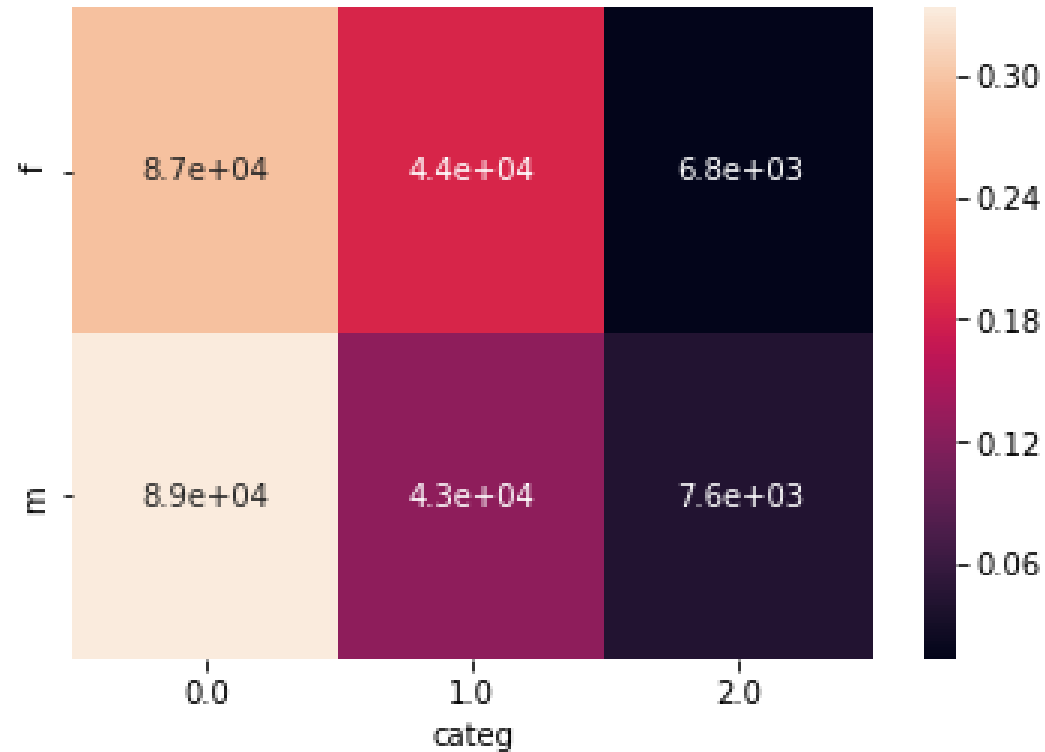
Analyse de données

Nombre de vente par jour
de produits de catégorie 1
au mois d'octobre



Problème de récolte de
données ?

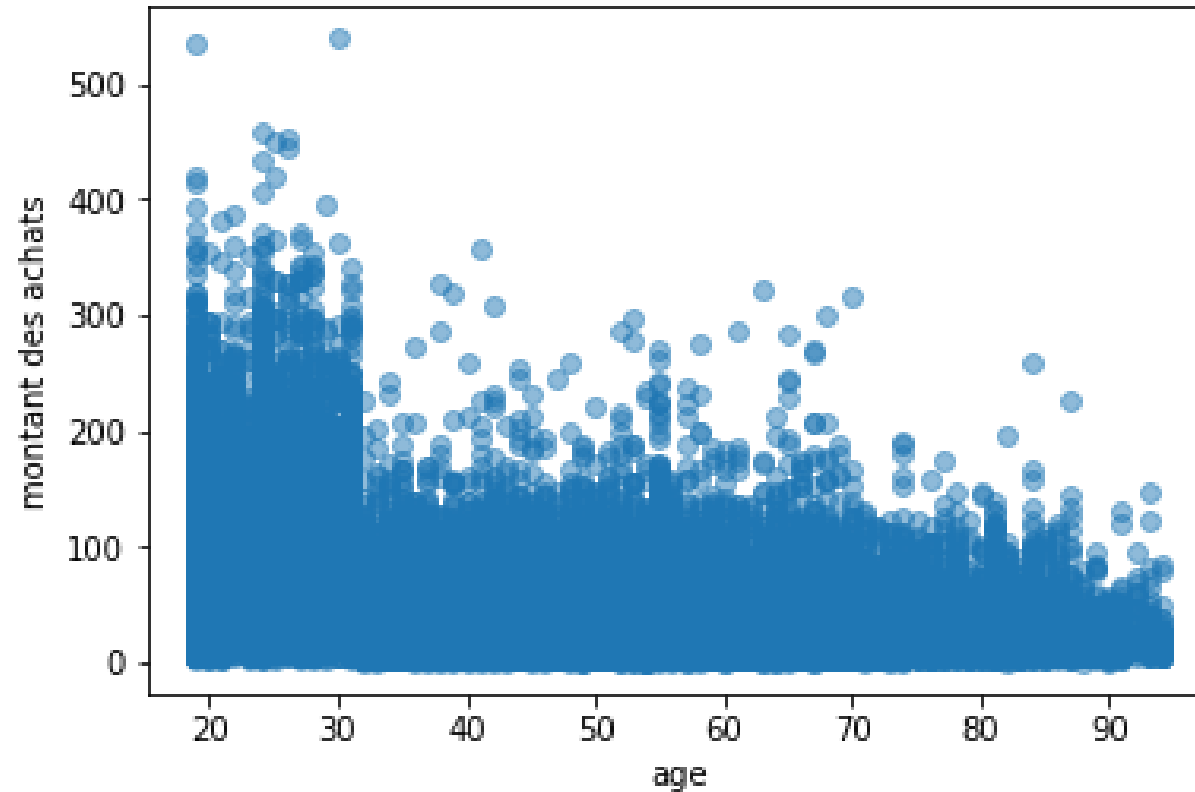
Corrélations : sexe & catégorie



Hypothèse H0 : On suppose les variables sexe et catégorie d'être indépendantes

P value = 0.0% < 5% on rejette donc l'hypothèse. Les deux variables sont corrélées

Corrélations : âge & montant total d'achat



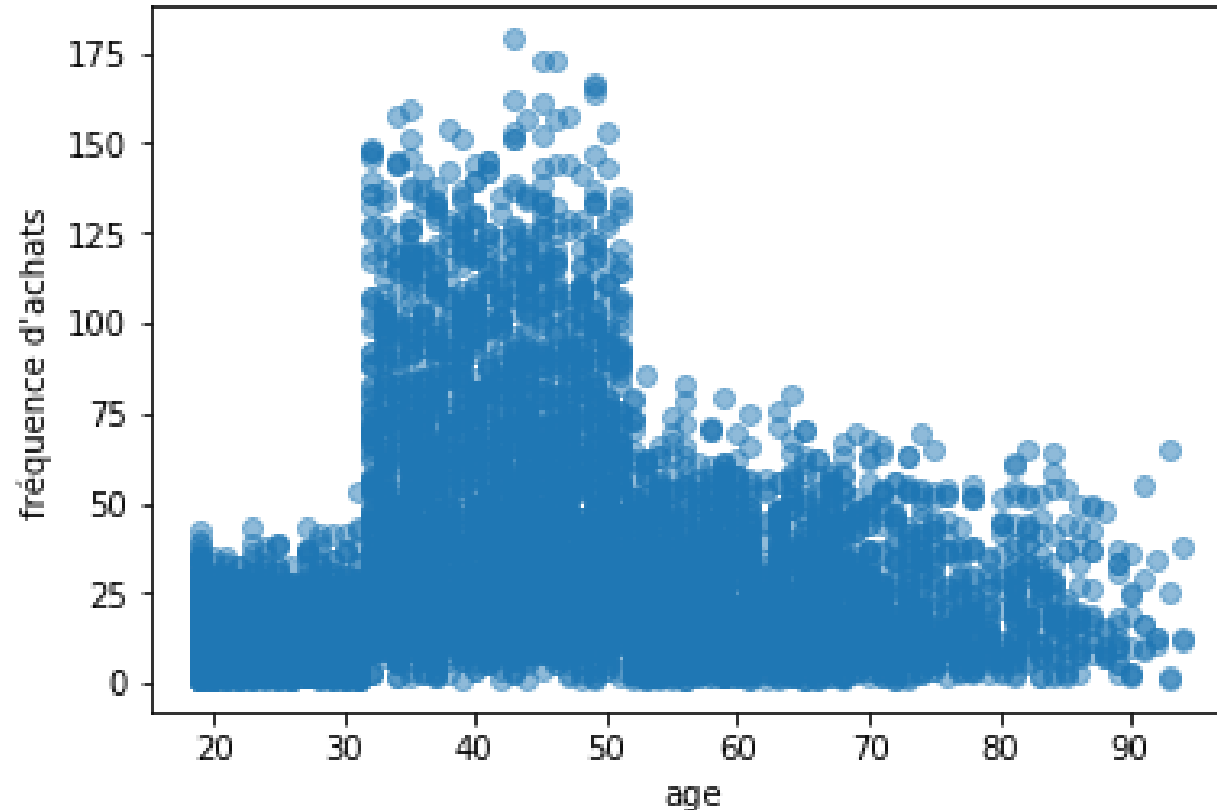
Hypothèse H_0 : On suppose les variables âge et montant total d'achat d'être indépendantes

Pearsonr : 0.3388886032649182

Spearmanr : 0.34620377687930026

P value = 0.0% < 5% on rejette donc l'hypothèse. Les deux variables sont corrélées

Corrélations : âge & fréquence d'achat



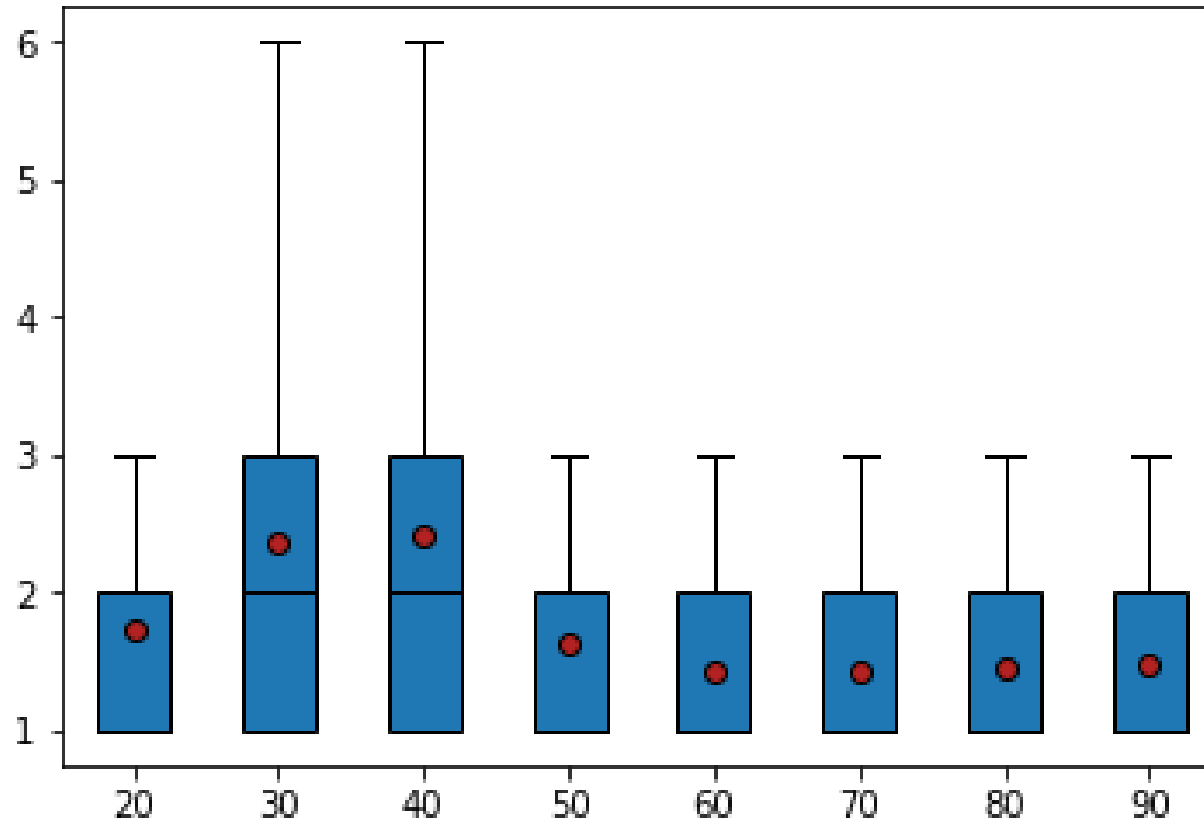
Hypothèse H_0 : On suppose les variables âge et fréquence d'achat d'être indépendantes

Pearsonr : -0.02469561183885214

Spearmanr : -0.12621799613693305

pvalue : 8.614354986383228e-32 % < 5% on rejette donc l'hypothèse. Les deux variables sont corrélées

Corrélations : âge & taille du panier moyen



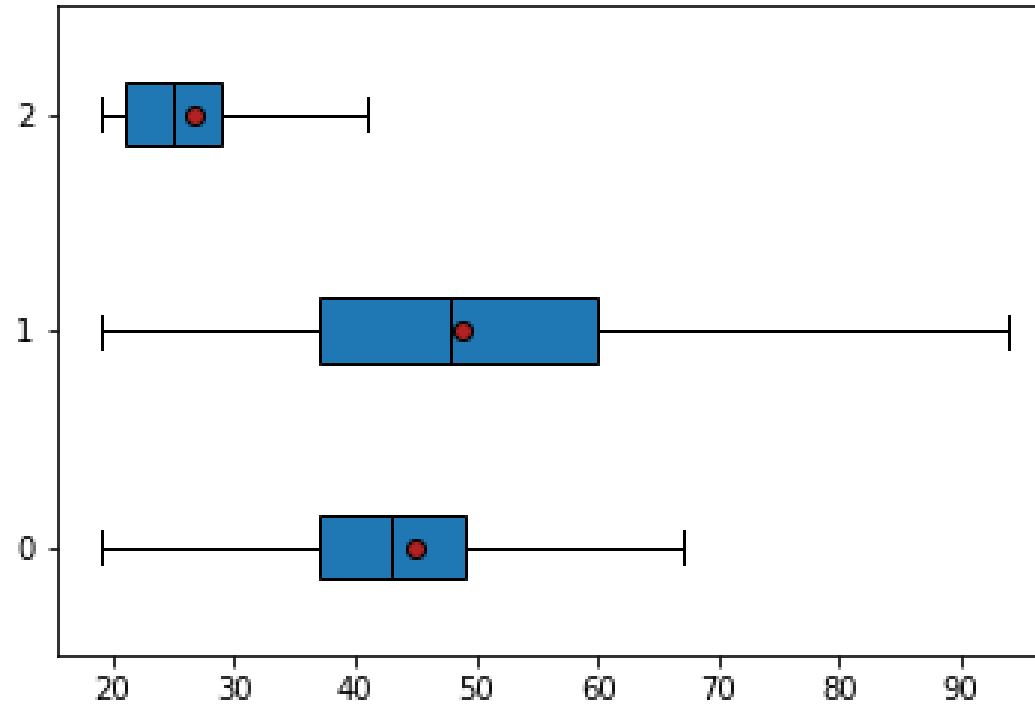
Hypothèse H_0 : On suppose les variables âge et taille du panier moyen d'être indépendantes

Pearsonr : 0.20033407869284361

Spearmanr : 0.2441759897471839

pvalue : 0.0 % < 5% on rejette donc l'hypothèse. Les deux variables sont corrélées

Corrélations : âge & catégorie



Hypothèse H0 : On suppose les variables âge et catégorie d'être indépendantes

P value = 0.0% < 5% on rejette donc l'hypothèse. Les deux variables sont corrélées

Merci pour votre attention