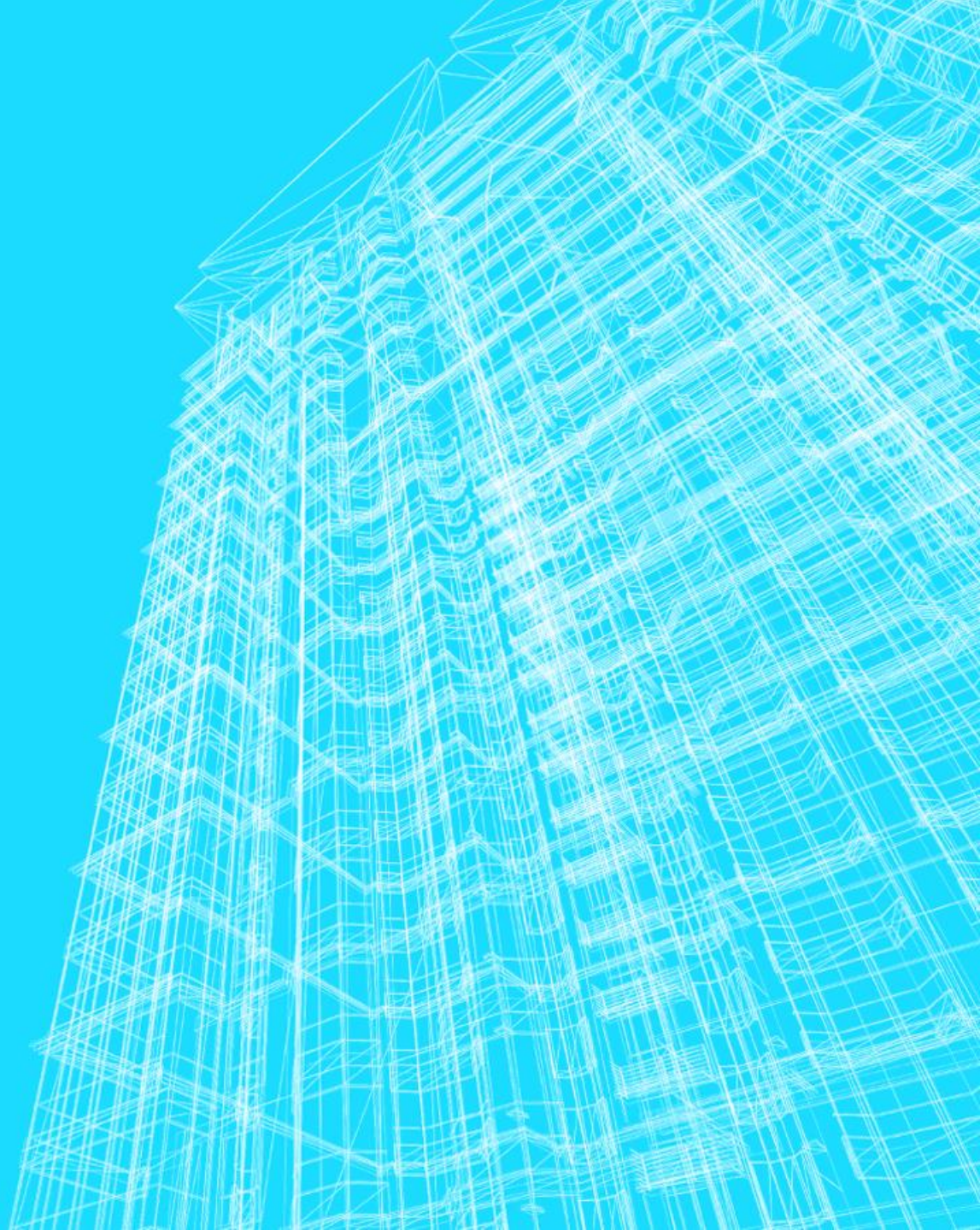
RESTER LIVRES

Cheikhou FOFANA



ANALYSE DES VENTES DE « RESTER LIVRES »

- Présentation des fichiers
- Préparation des données
- Analyse des ventes
- Analyse des corrélations
- Conclusion

I- PRÉSENTATION DES FICHIERS

- 3 fichiers à disposition

Products

```
products.head()
```

	id_prod	price	categ
0	0_1421	19.99	0
1	0_1368	5.13	0
2	0_731	17.99	0
3	1_587	4.99	1
4	0_1507	3.99	0

Transactions

```
transact.head()
```

	id_prod	date	session_id	client_id
0	0_1483	2021-04-10 18:37:28.723910	s_18746	c_4450
1	2_226	2022-02-03 01:55:53.276402	s_159142	c_277
2	1_374	2021-09-23 15:13:46.938559	s_94290	c_4270
3	0_2186	2021-10-17 03:27:18.783634	s_105936	c_4597
4	0_1351	2021-07-17 20:34:25.800563	s_63642	c_1242

Customers

```
customers.head()
```

	client_id	sex	birth
0	c_4410	f	1967
1	c_7839	f	1975
2	c_1699	f	1984
3	c_5961	f	1962
4	c_5320	m	1943

CUSTOMERS

```
customers.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 8623 entries, 0 to 8622
Data columns (total 3 columns):
 #   Column      Non-Null Count  Dtype 
---  ------      --------------  ----- 
 0   client_id   8623 non-null   object
 1   sex         8623 non-null   object
 2   birth       8623 non-null   int64 
dtypes: int64(1), object(2)
memory usage: 202.2+ KB
```

```
#Proportion d'hommes dans le fichier
customers[customers['sex'] == 'm'].shape
```

```
(4132, 3)
```

```
#Proportion de femmes
customers[customers['sex'] == 'f'].shape
```

```
(4491, 3)
```

On a 8623 clients répartis en 4132 d'hommes et 4491 femmes

```
#Checker l'existence de doublons
print('Il n\'y a pas de doublons dans le fichier:',
      customers.size == customers.drop_duplicates('client_id').size)
```

```
Il n'y a pas de doublons dans le fichier: True
```

```
#Suppression des clients de tests
customers = customers.loc[(customers['client_id'] != 'ct_0')
                          & (customers['client_id'] != 'ct_1')]
```

TRANSACTIONS

```
transact.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 336816 entries, 0 to 337015
Data columns (total 4 columns):
 #   Column      Non-Null Count   Dtype 
---  ------      --------------   ----- 
 0   id_prod     336816 non-null  object
 1   date        336816 non-null  object
 2   session_id  336816 non-null  object
 3   client_id   336816 non-null  object
dtypes: object(4)
memory usage: 12.8+ MB
```

```
#Supprimons les transactions de tests
transact = transact[(transact['client_id'] != 'ct_0')
                    & (transact['client_id'] != 'ct_1')]
```

```
#Checker l'existence de doublons
print('Il n\'y a pas de doublons dans le fichier:',
      transact.size == transact.drop_duplicates('date').size)
```

```
Il n'y a pas de doublons dans le fichier: True
```

PRODUCTS

```
products.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 3286 entries, 0 to 3286
Data columns (total 3 columns):
 #   Column   Non-Null Count  Dtype
---  ------   --------------  -----
 0   id_prod  3286 non-null   object
 1   price    3286 non-null   float64
 2   categ    3286 non-null   int64
dtypes: float64(1), int64(1), object(1)
memory usage: 102.7+ KB
```

```
#Checker l'existence de doublons
print('Il n\'y a pas de doublons dans le fichier:',
      products.size == products.drop_duplicates('id_prod').size)
```

```
Il n'y a pas de doublons dans le fichier: True
```

```
#Suppression du produit de test
products = products[products['id_prod'] != 'T_0']
```

II- PRÉPARATION DES DONNÉES

```
# Jointure entre transact et products
df = pd.merge(transact, products, on = ['id_prod'], how='outer')
```

Identifier les produits non vendus et les supprimer

```
#Identifier les produits qui n'ont pas été vendus
df.loc[(df['date'].isnull()) | (df['client_id'].isnull()), 'id_prod'].unique()
```

```
array(['0_1016', '0_1780', '0_1062', '0_1119', '0_1014', '1_0', '0_1318',
       '0_1800', '0_1645', '0_322', '0_1620', '0_1025', '2_87', '1_394',
       '2_72', '0_310', '0_1624', '0_525', '2_86', '0_299', '0_510',
       '0_2308'], dtype=object)
```

categ	0	1	2
price	17.0	2.0	3.0

22 articles n'ont pas été vendus dont 17 de catégorie 0, 2 de la catégorie 1 et 3 de la catégorie 2.

Il y a 103 transactions concernant le produit d'id ***0_2245*** dont on ignore sa catégorie et son prix.

```
df[(pd.isnull(df['price']) == True) |
   (df['price'] == 0)].groupby(['id_prod']).agg('count')
```

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

```
#Supprimons les 22 produits
#non vendu du dataset
df.dropna(inplace=True)
df.isnull().sum()
```

```
id_prod       0
date          0
session_id    0
client_id     0
price         0
categ         0
```

On décide d'imputer les colonnes *categ* et *price* par respectivement 0 et la moyenne de categ 0.

```
# Imputons les colonnes categ et price par respectivement 0
#et la moyenne de la categ 0 pour le produit 0_2245
df['categ'] = df[['categ']].fillna(value=0)
df['price'] = df[['price']].fillna(value=10.65)
```

Jointure entre notre dataset nettoyé et customers

```
#Jointure entre notre dataset et customers
df = pd.merge(df, customers, on = ['client_id'], how='outer')
```

```
# Identifier les visiteurs
df.loc[df['id_prod'].isnull(), 'client_id'].unique()
```

```
array(['c_8253', 'c_3789', 'c_4406', 'c_2706', 'c_3443', 'c_4447',
       'c_3017', 'c_4086', 'c_6930', 'c_4358', 'c_8381', 'c_1223',
       'c_6862', 'c_5245', 'c_5223', 'c_6735', 'c_862', 'c_7584', 'c_90',
       'c_587', 'c_3526'], dtype=object)
```

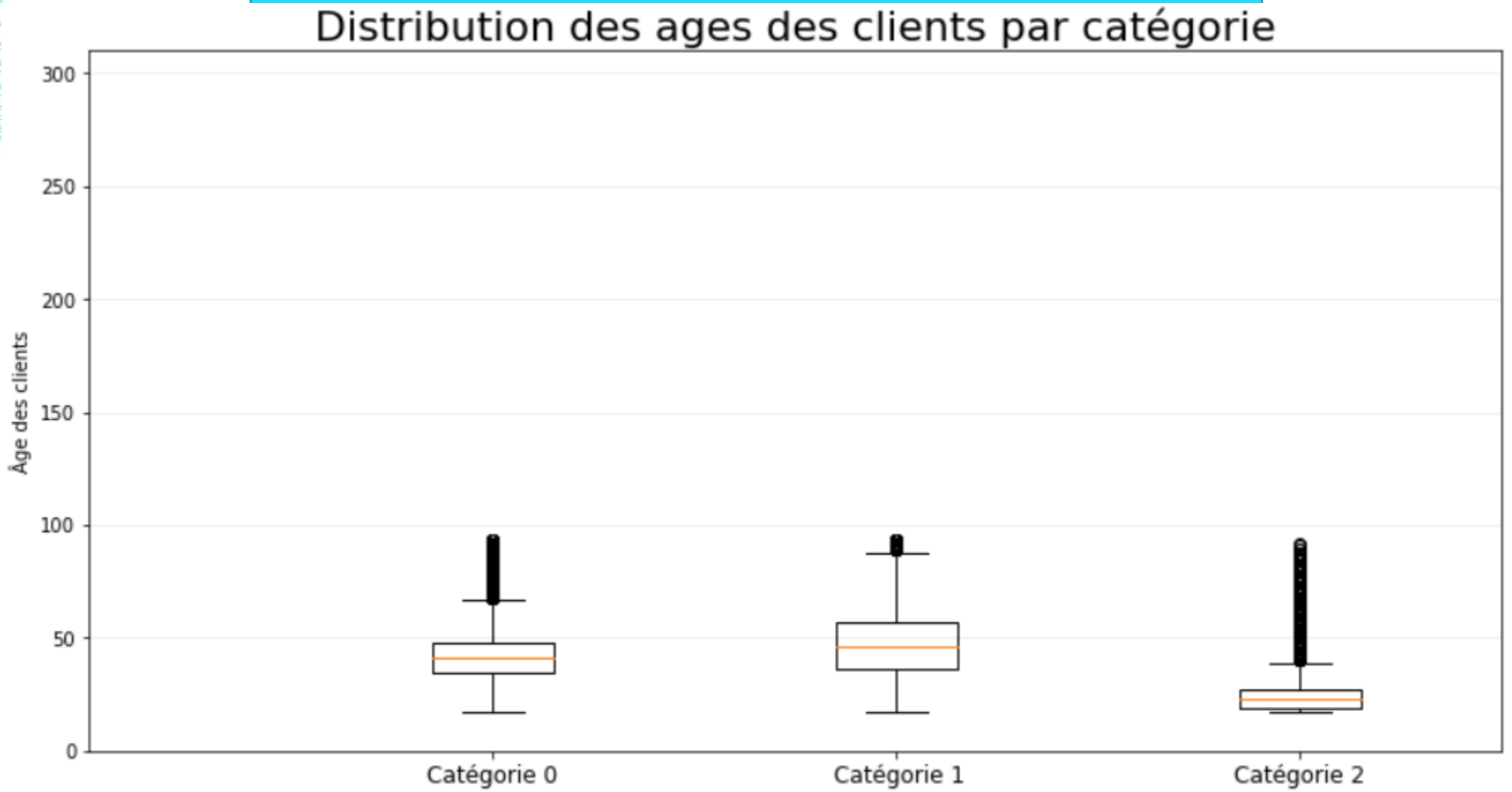
On dénombre 21 visiteurs sur le site qui n’ont pas acheté.
On les supprime du dataset aussi.

Dataset final avant extraction des valeurs aberrantes **336816** lignes et 13 colonnes

```
#Le dataset final
df.head()
```

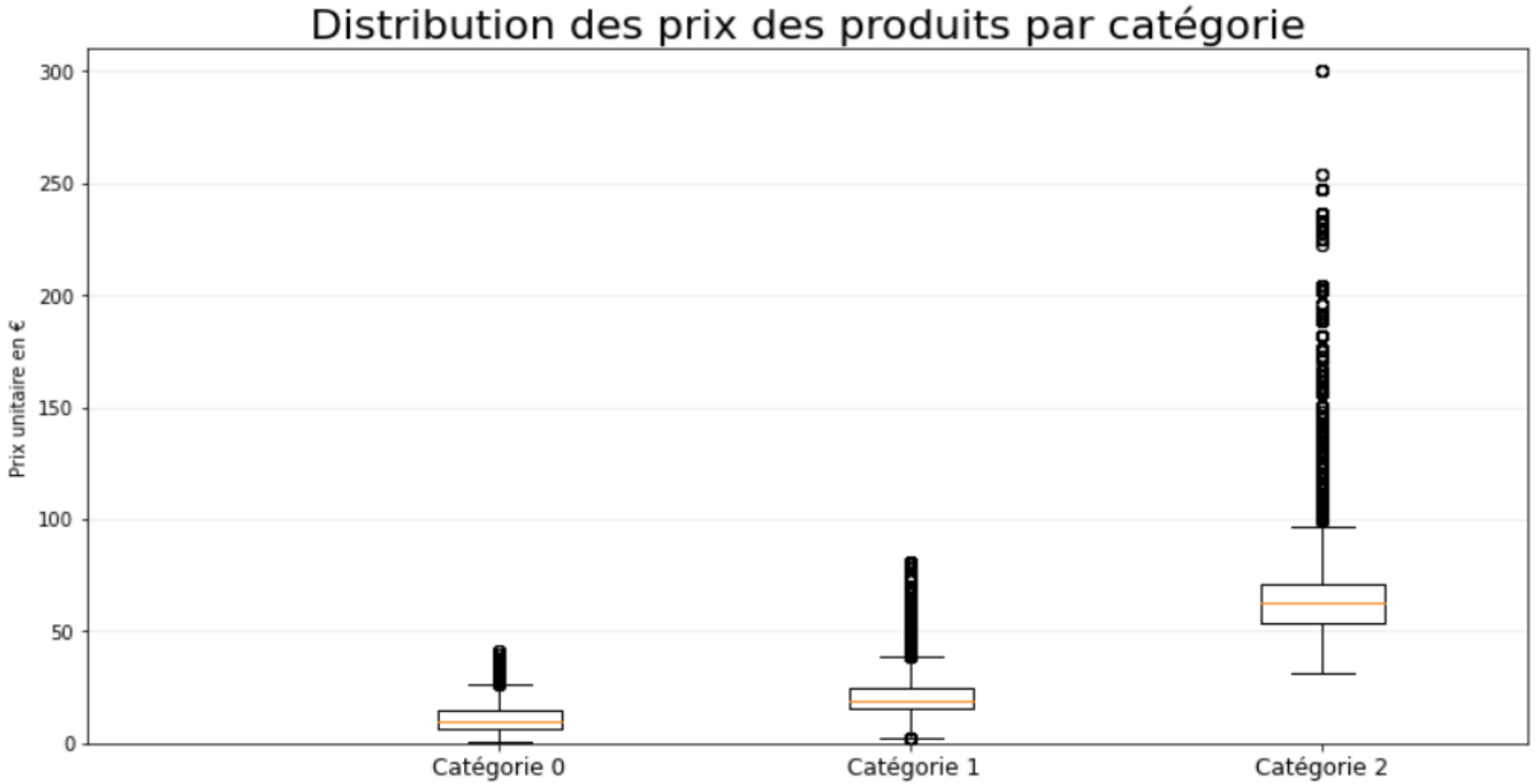
	client_id	session_id	sex	age	date	id_prod	categ	price	year	month	day	n_mois	heure
0	c_4450	s_18746	0	44	2021-04-10 18:37:28.723910	0_1483	0	4.99	2021	4	10	4	18
1	c_4450	s_97382	0	44	2021-09-29 11:14:59.793823	0_1085	0	3.99	2021	9	29	9	11
2	c_4450	s_81509	0	44	2021-08-27 19:50:46.796939	0_1453	0	7.99	2021	8	27	8	19
3	c_4450	s_81509	0	44	2021-08-27 20:07:25.878440	0_1405	0	4.99	2021	8	27	8	20
4	c_4450	s_141302	0	44	2021-12-28 11:45:04.072281	0_1392	0	6.30	2021	12	28	12	11

Valeurs aberrantes sur la colonne *age*




```
df = df.loc[((df['categ'] == 0) & (df['age'] < 66)) |
            ((df['categ'] == 1) & (df['age'] < 88)) |
            ((df['categ'] == 2) & (df['age'] < 37))]
```

Valeurs aberrantes sur la colonne *price*



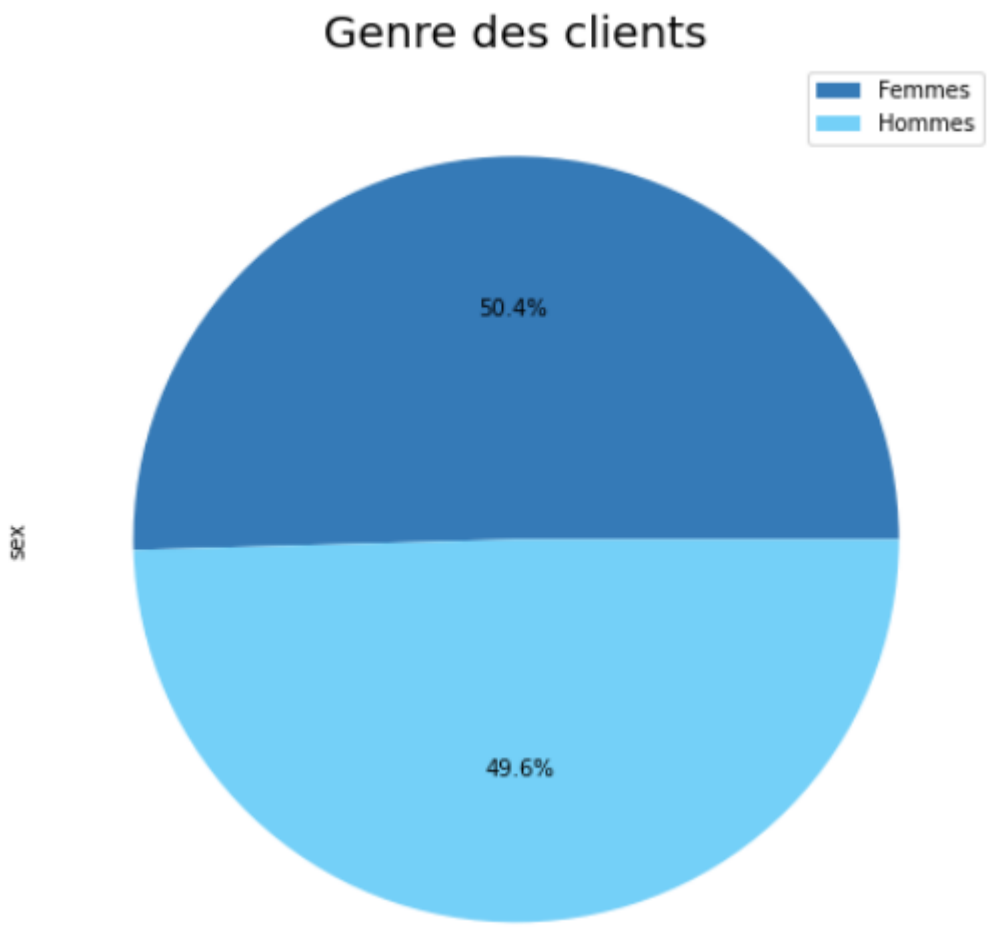

```
df = df.loc[((df['categ'] == 0) & (df['price'] < 26.5)) |
            ((df['categ'] == 1)  & (df['price'] < 37)) |
            ((df['categ'] == 2) & (df['price'] < 93.5))]
```

Dataset final avec **318436** lignes et 14 colonnes

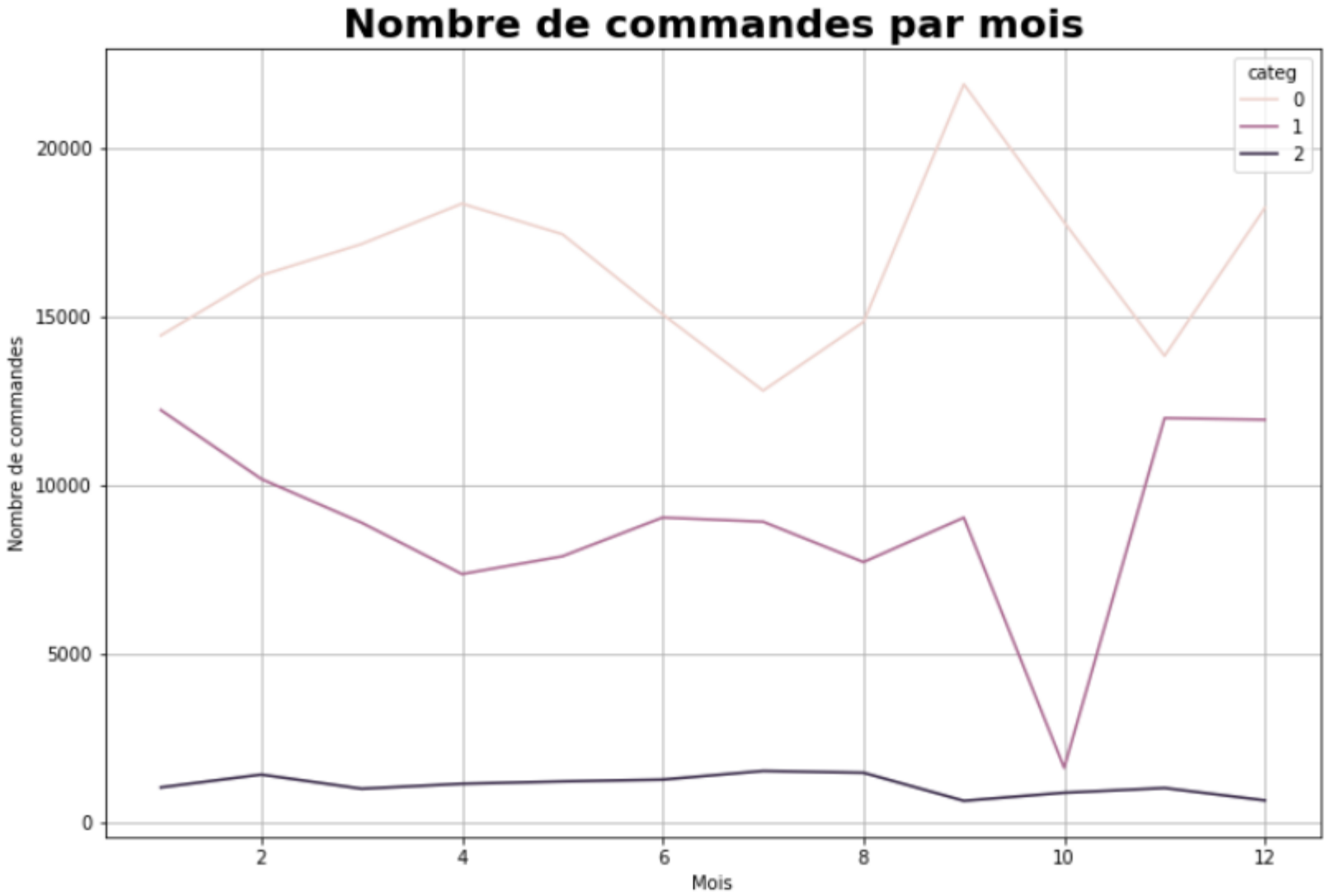
```
df.head()
```

	client_id	session_id	sex	age	date	id_prod	categ	price	year	month	day	n_mois	heure	partie_jour
0	c_4450	s_18746	0	44	2021-04-10 18:37:28.723910	0_1483	0	4.99	2021	4	10	4	18	Afternoon
1	c_4450	s_97382	0	44	2021-09-29 11:14:59.793823	0_1085	0	3.99	2021	9	29	9	11	Morning
2	c_4450	s_81509	0	44	2021-08-27 19:50:46.796939	0_1453	0	7.99	2021	8	27	8	19	Afternoon
3	c_4450	s_81509	0	44	2021-08-27 20:07:25.878440	0_1405	0	4.99	2021	8	27	8	20	Afternoon
4	c_4450	s_141302	0	44	2021-12-28 11:45:04.072281	0_1392	0	6.30	2021	12	28	12	11	Morning

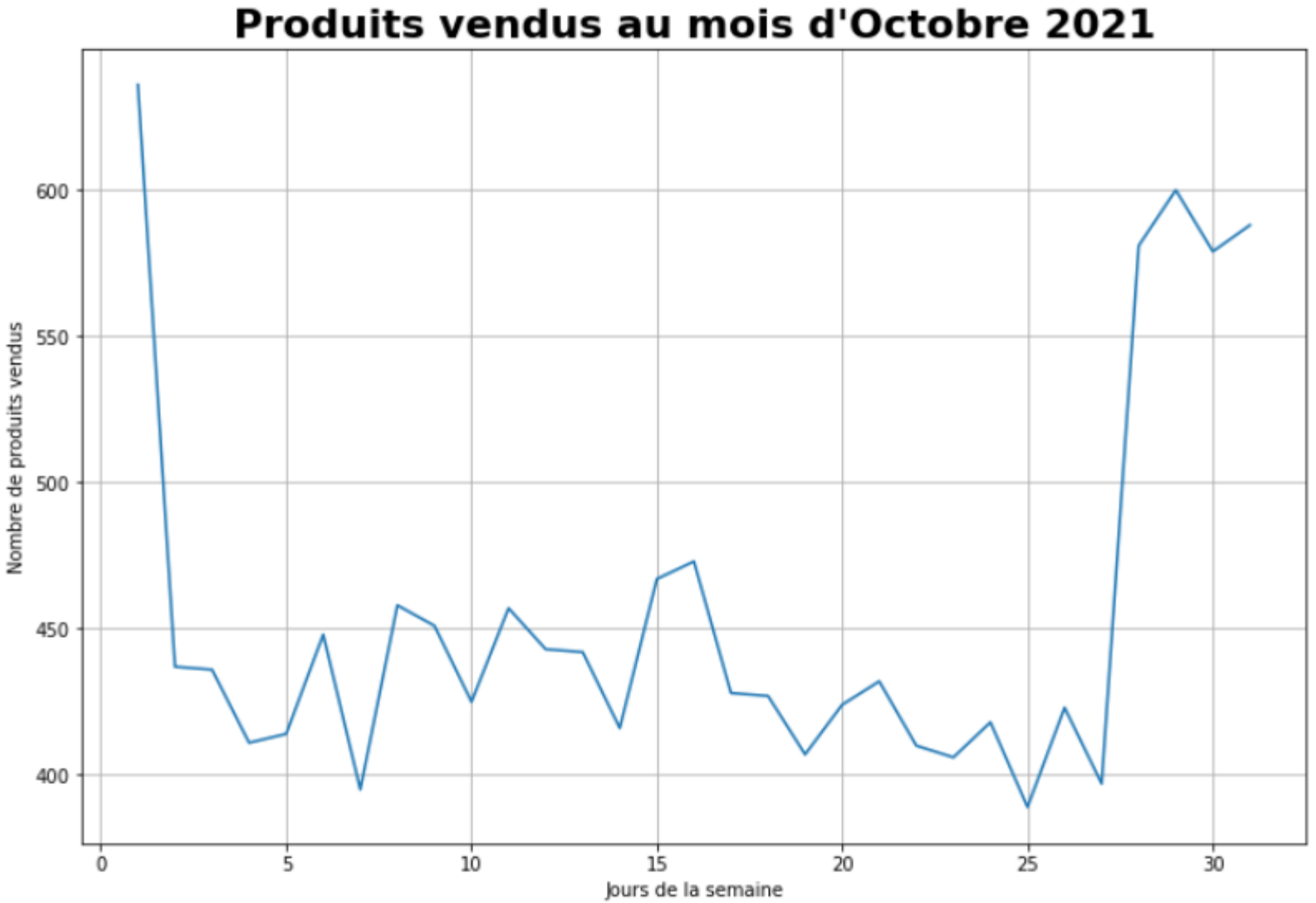
III- ANALYSE DES VENTES



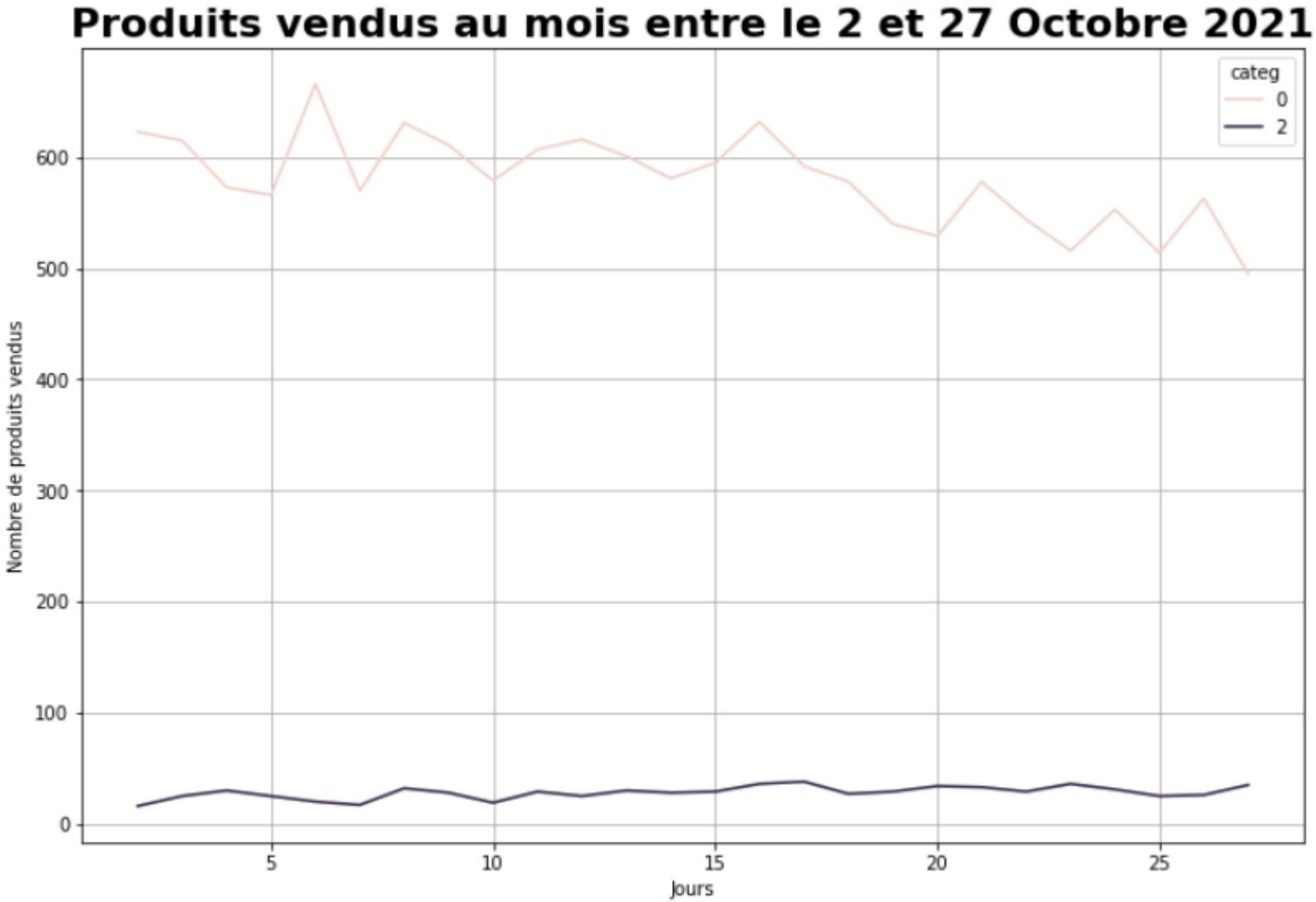
On a la même proportion d’hommes et de femmes.



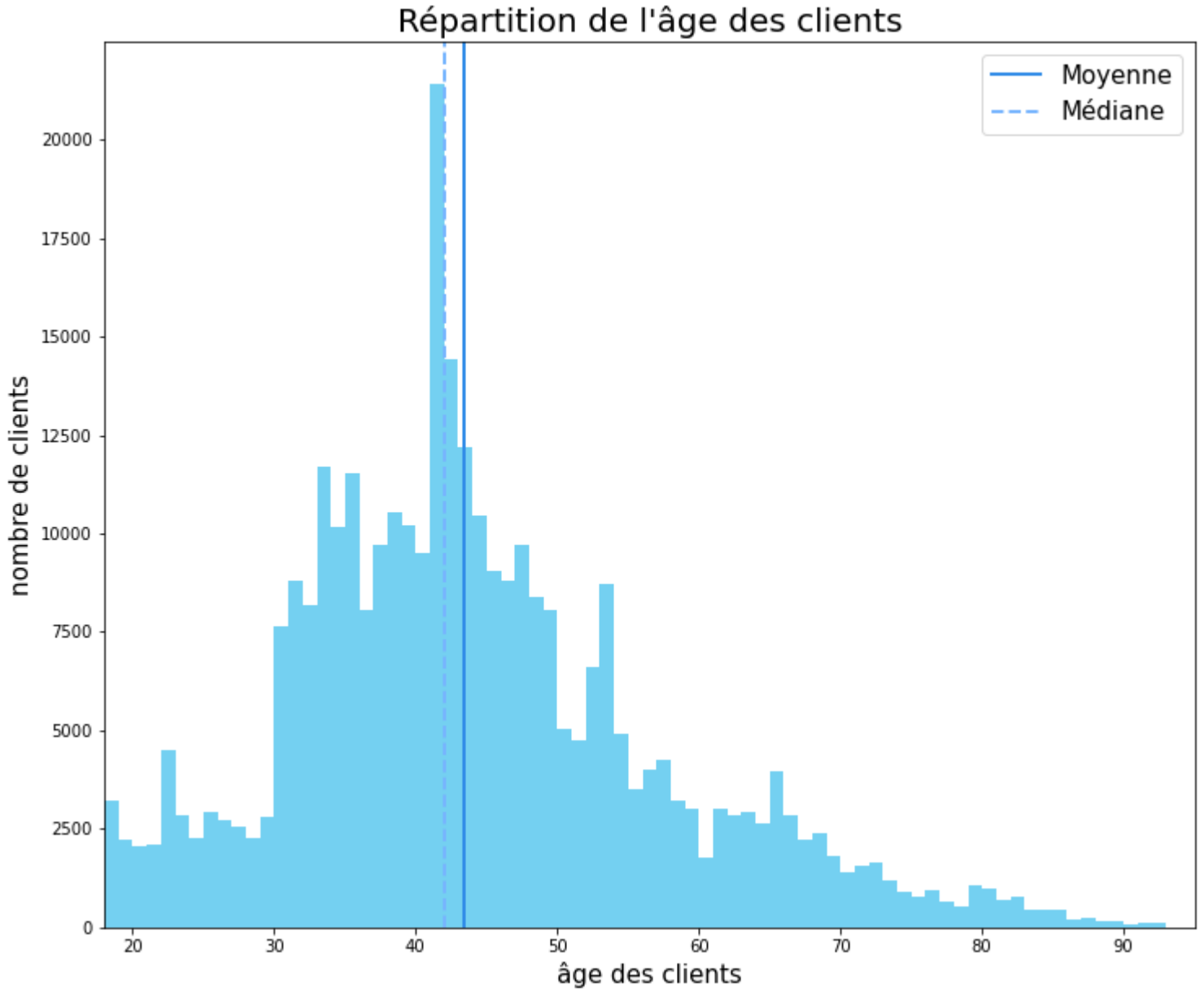
On observe une suite chute drastique au mois d'Octobre où les commandes des produits de la catégorie 1, c'est dû certainement une anomalie.
Au même moment, on observe une relative stabilité des ventes des autres catégories de produits avec la catégorie 0 qui fait un saut spectaculaire au mois de Septembre.



L’anomalie débute le 2 Octobre et se poursuit jusqu’au 27 Octobre.
Pas moins de 992 articles n’ont pas connus la moindre vente au cours du mois.

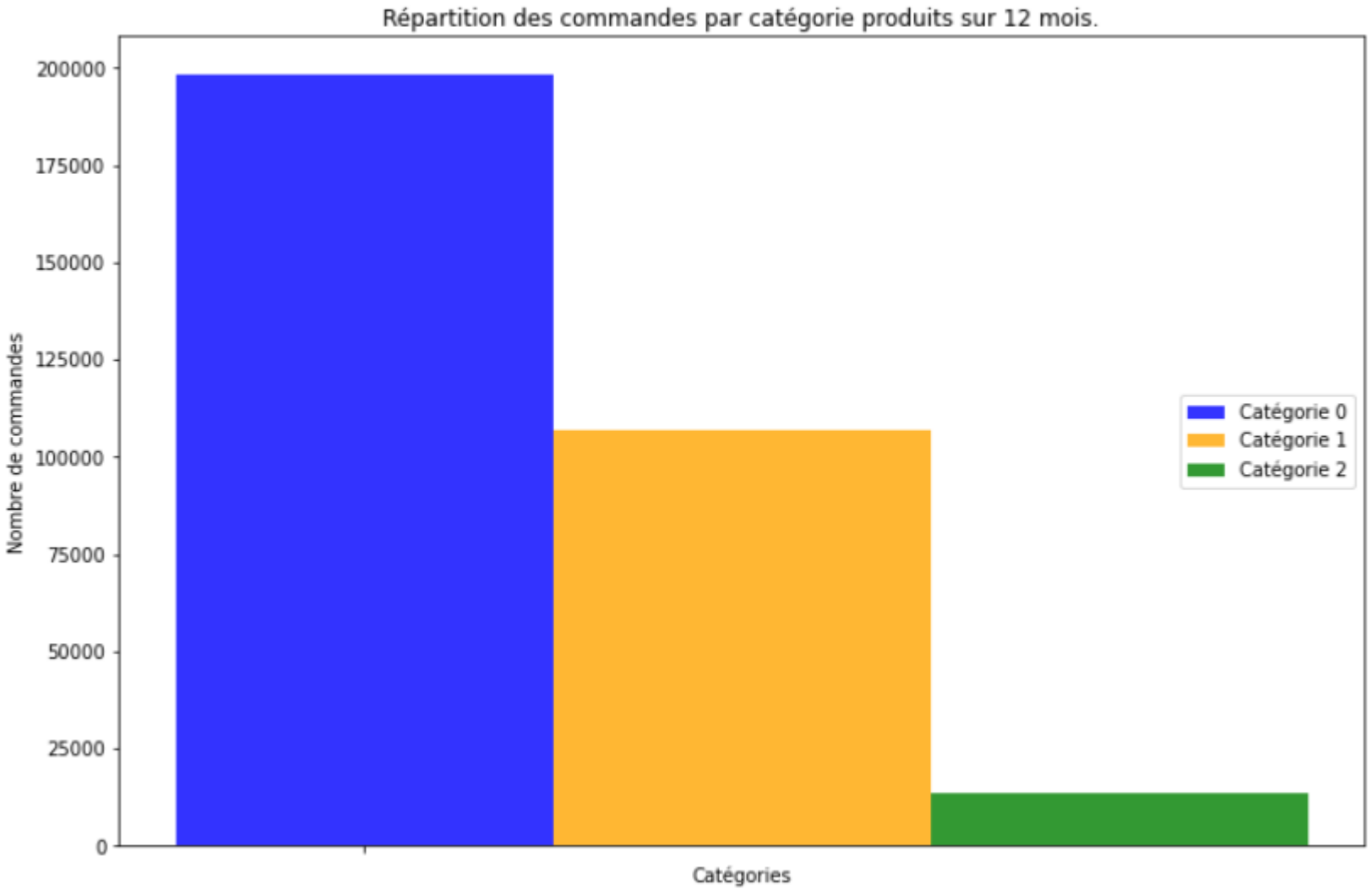


On constate l'absence totale des produits de la catégorie 1 dans les ventes entre le 2 et 27 Octobre.
Par conséquent, on décide de ne pas tenir en compte du mois d'Octobre pour la suite de l'analyse.



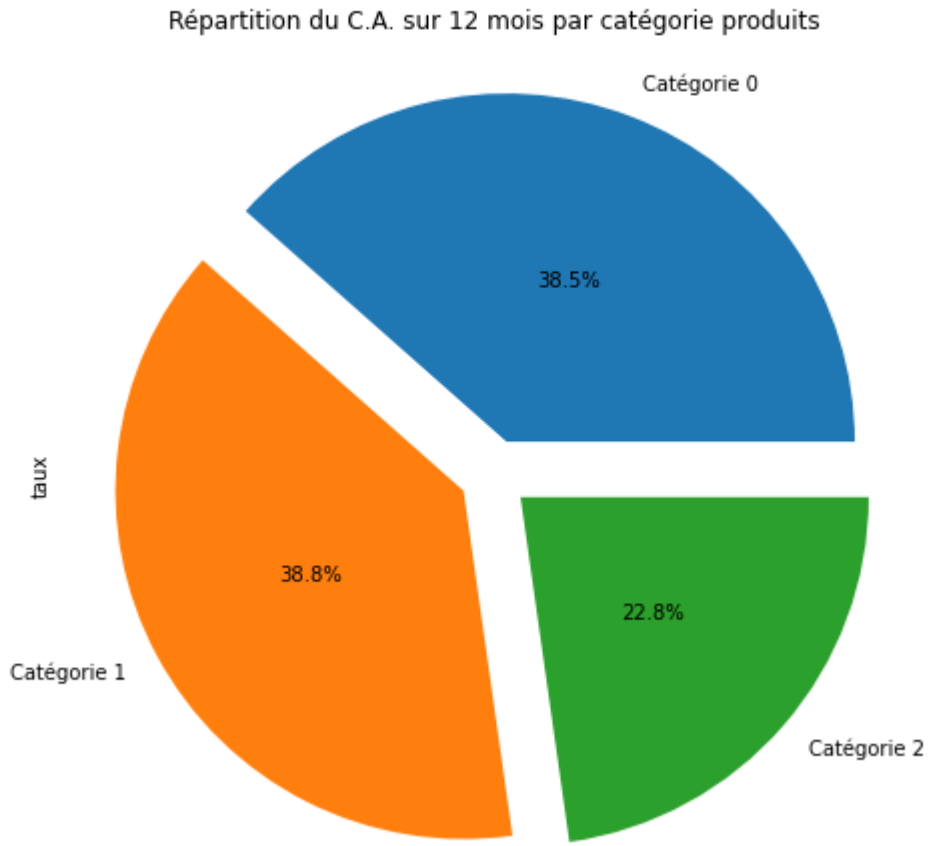
Les clients les plus actifs ont entre 32 et 55 ans.
Les femmes passent 2 fois plus de commandes que les hommes.
L'âge le plus représentatif est 41 ans.

Le plus grand nombre de commandes pour la catégorie 0

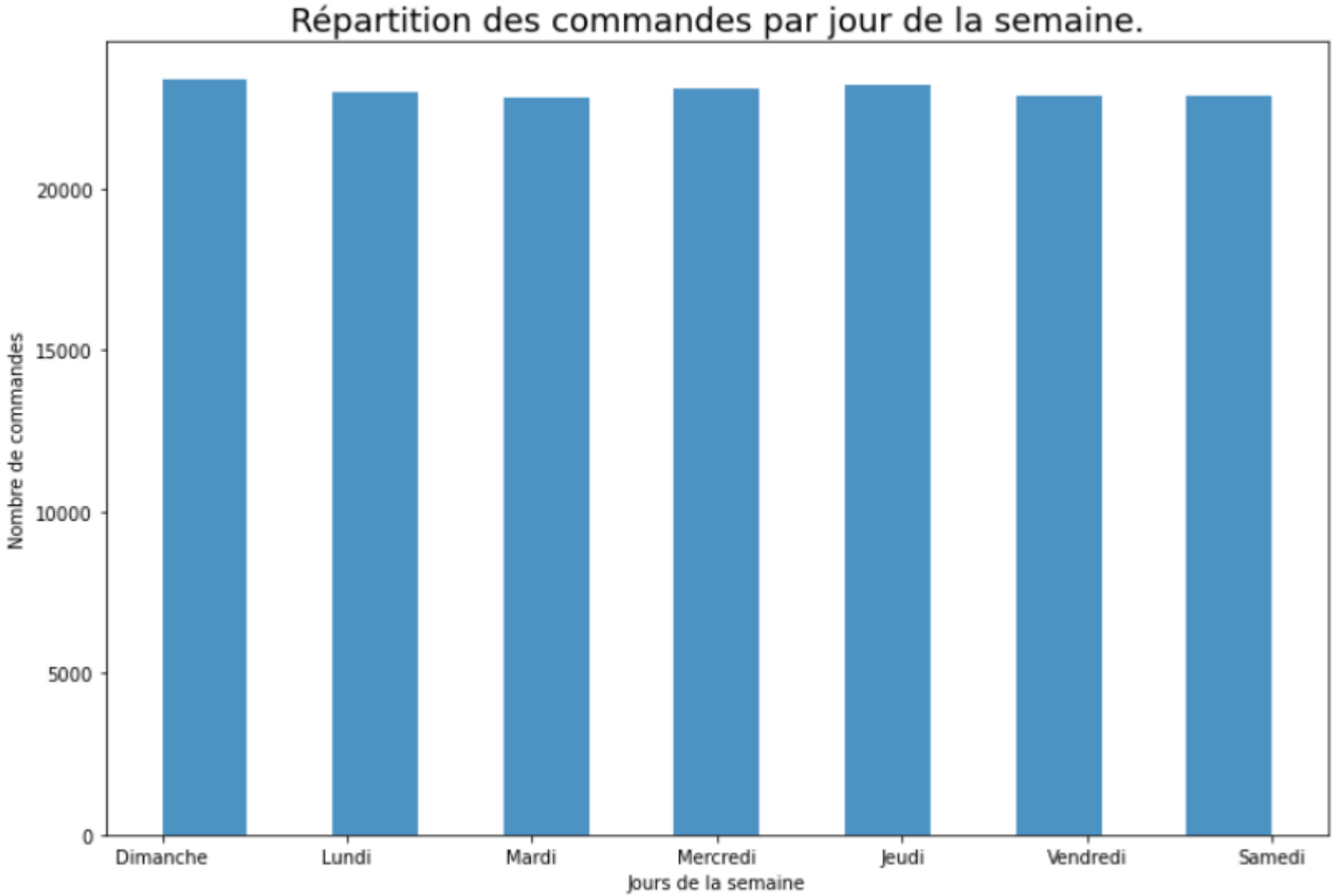


La catégorie 0 présente près de 2 fois plus de commandes que la catégorie 1 et plus de 8 fois la catégorie 2.

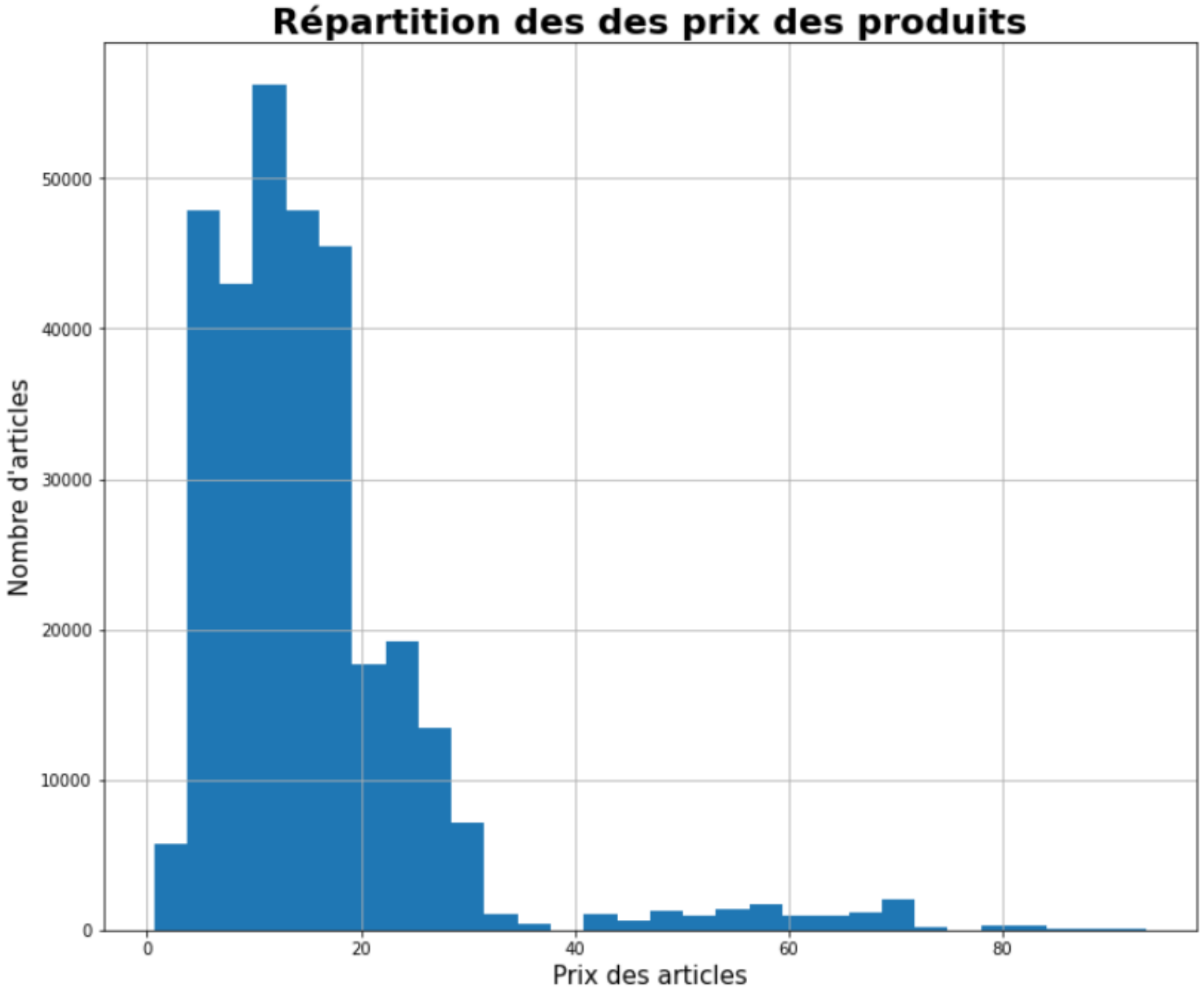
La catégorie 1 génère le plus gros chiffre d'affaires



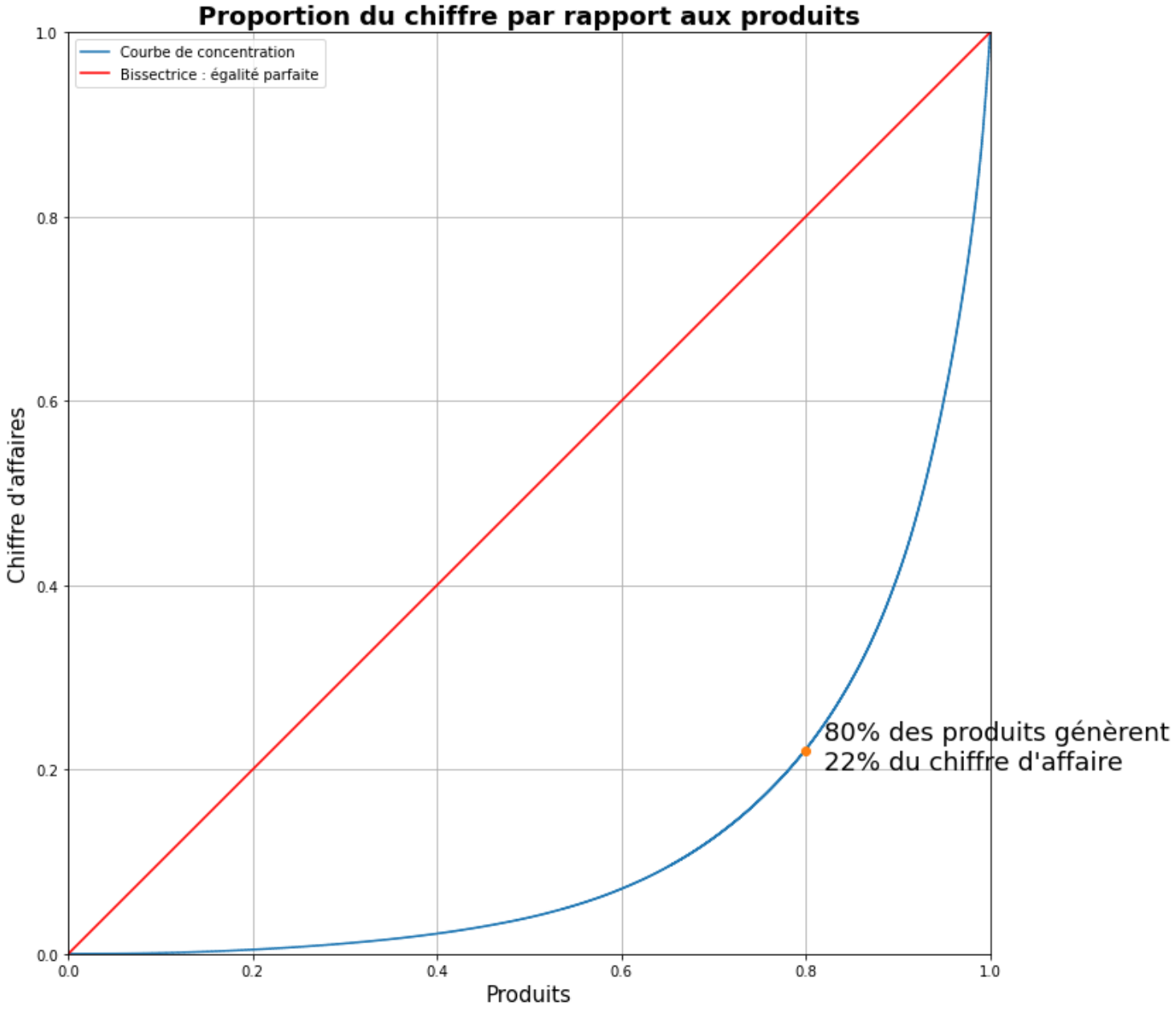
La catégorie 1 génère le plus gros chiffre d’affaire légèrement devant la catégorie 0.



Le nombre de commandes par jour de la semaine est relativement constant avec une légère hausse et les Jeudis et Dimanches.



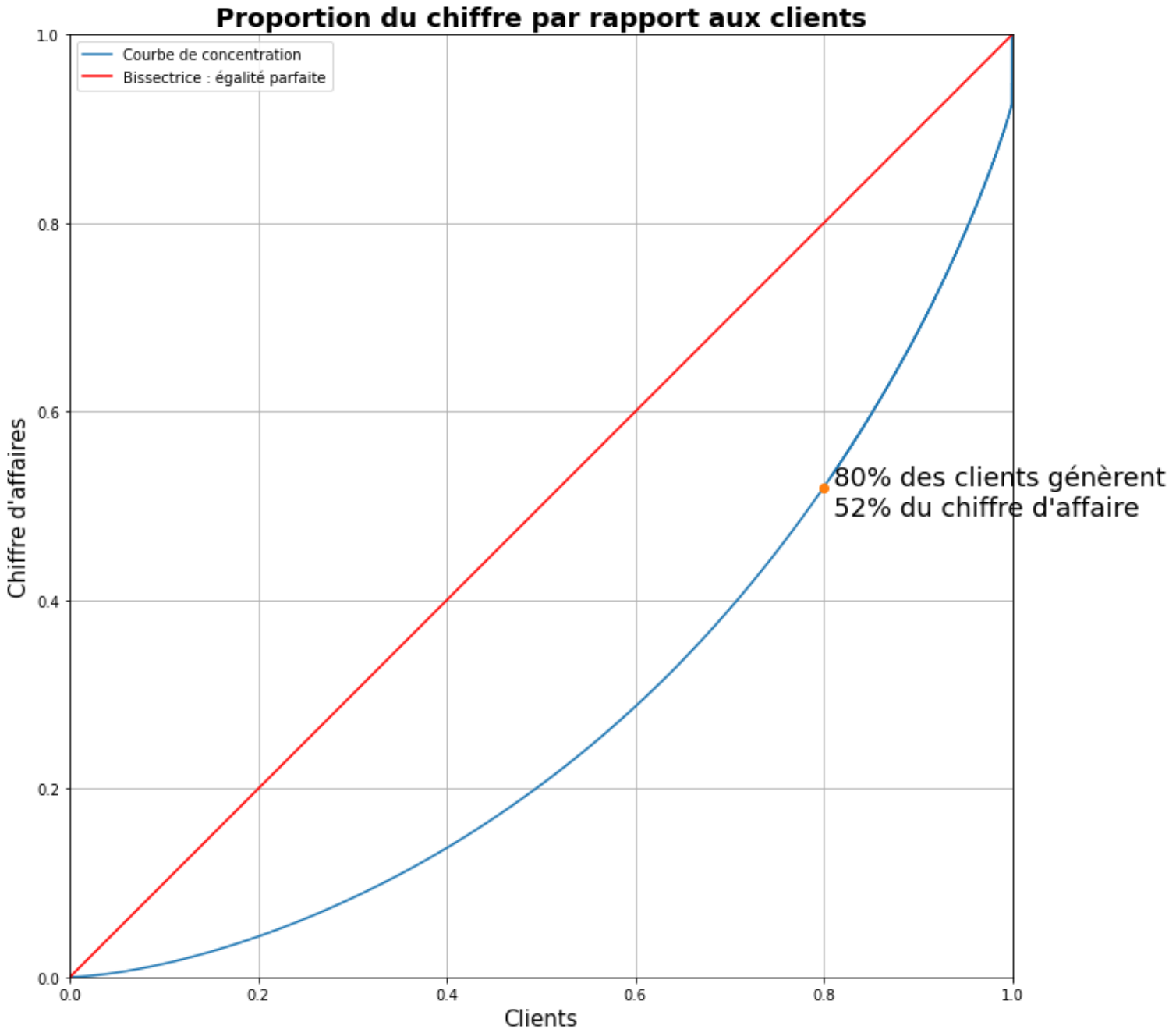
Les prix varient majoritairement entre 2€ et 25€.



L'indice de Gini est: 0.74

La courbe de Lorenz montre une concentration du chiffre d'affaire. En effet, 20% des produits génèrent 78% du chiffre d'affaire.

Gini = 0,74

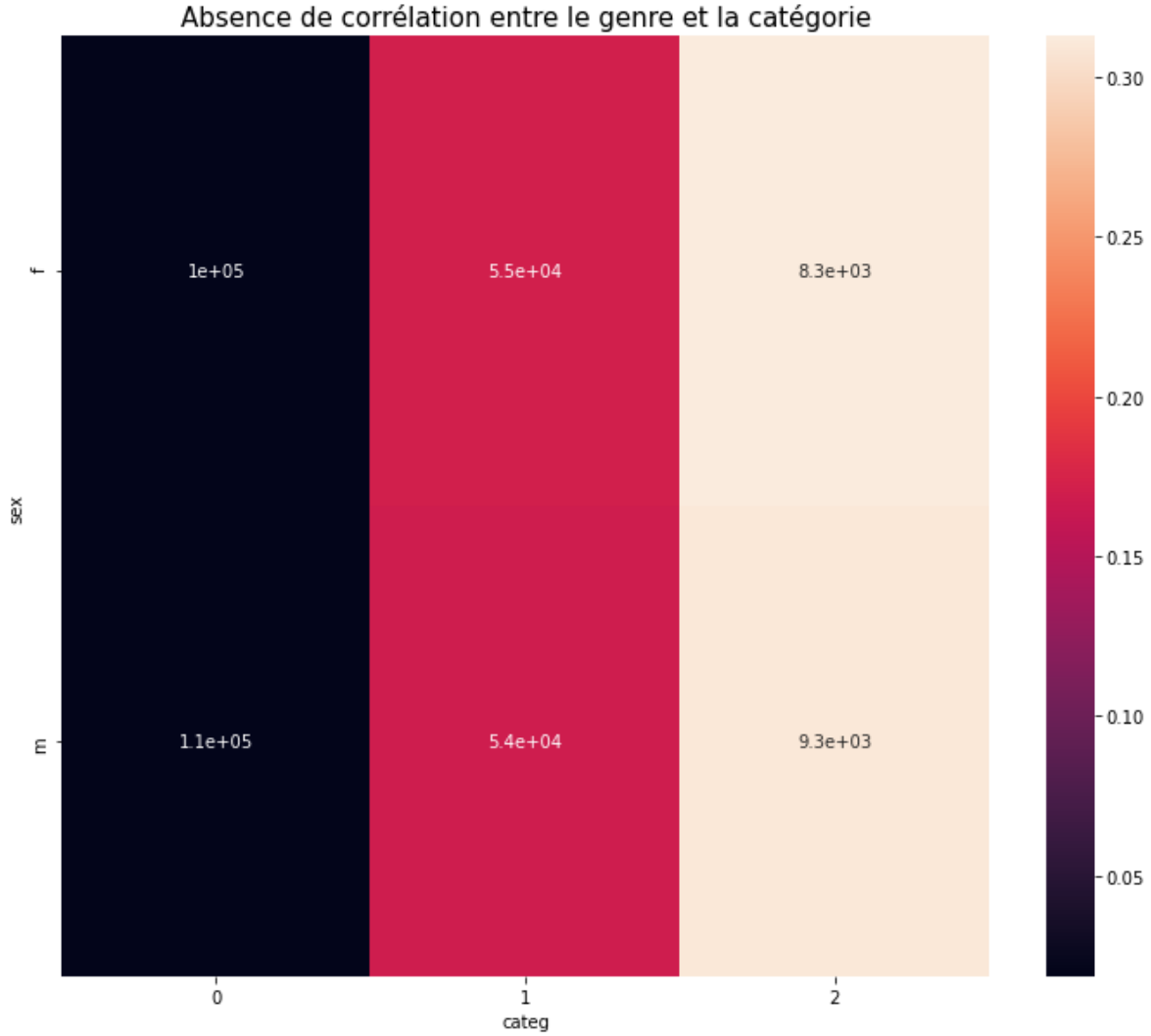


L'indice de Gini est: 0.44

La courbe de Lorenz montre une concentration du chiffre d'affaire.
En effet, 20% des clients génèrent 48% du chiffre d'affaire.

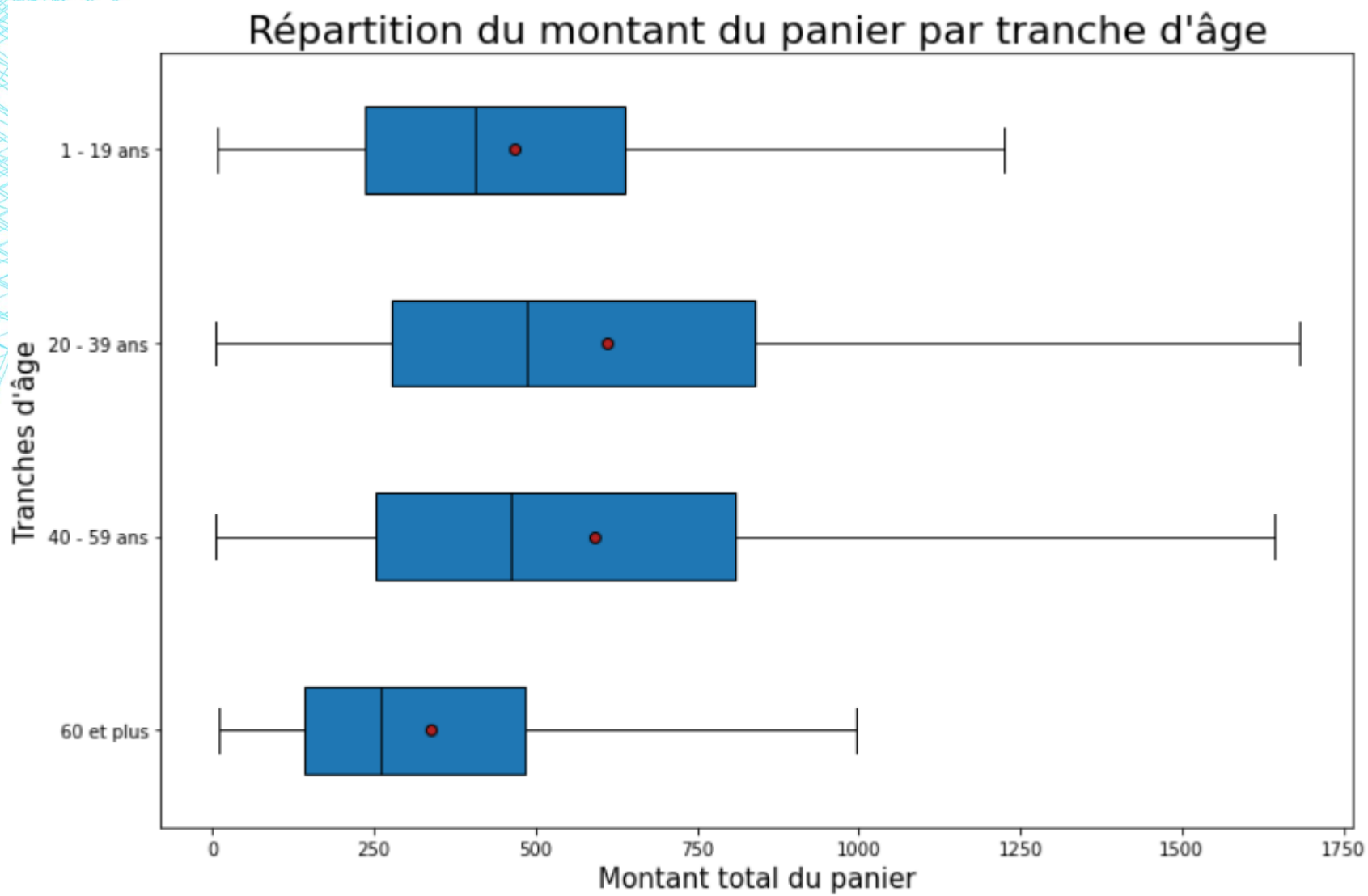
Gini = 0,44

IV- ANALYSE DES CORRÉLATIONS



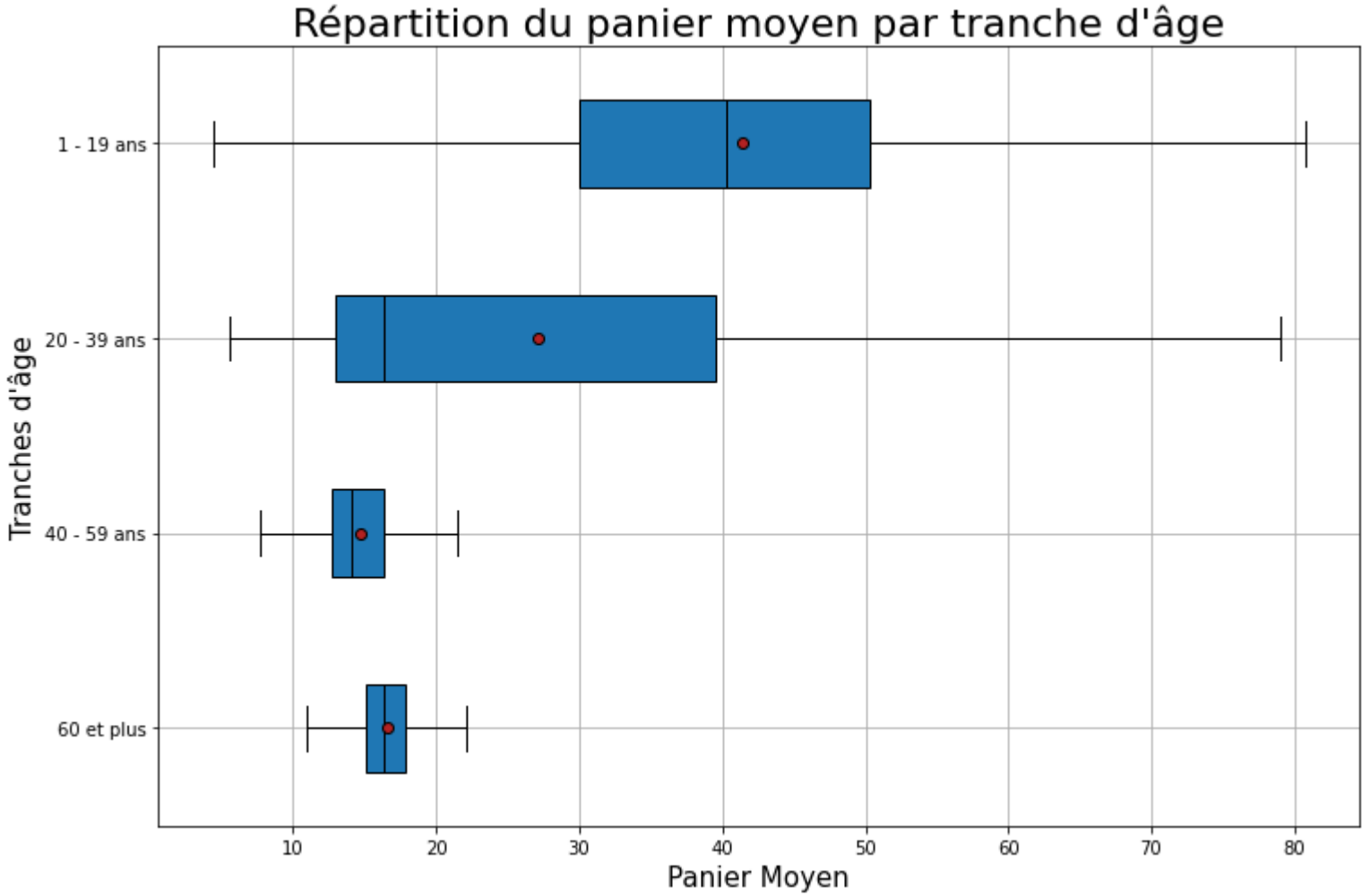
Globalement, il n'y a pas de corrélation entre le genre et la catégorie, néanmoins on observe une certaine corrélation entre le sexe et la catégorie 2.

Les variables sex et categ sont indépendantes donc il n'y a pas de corrélation entre elles car xi_n = 81.73.



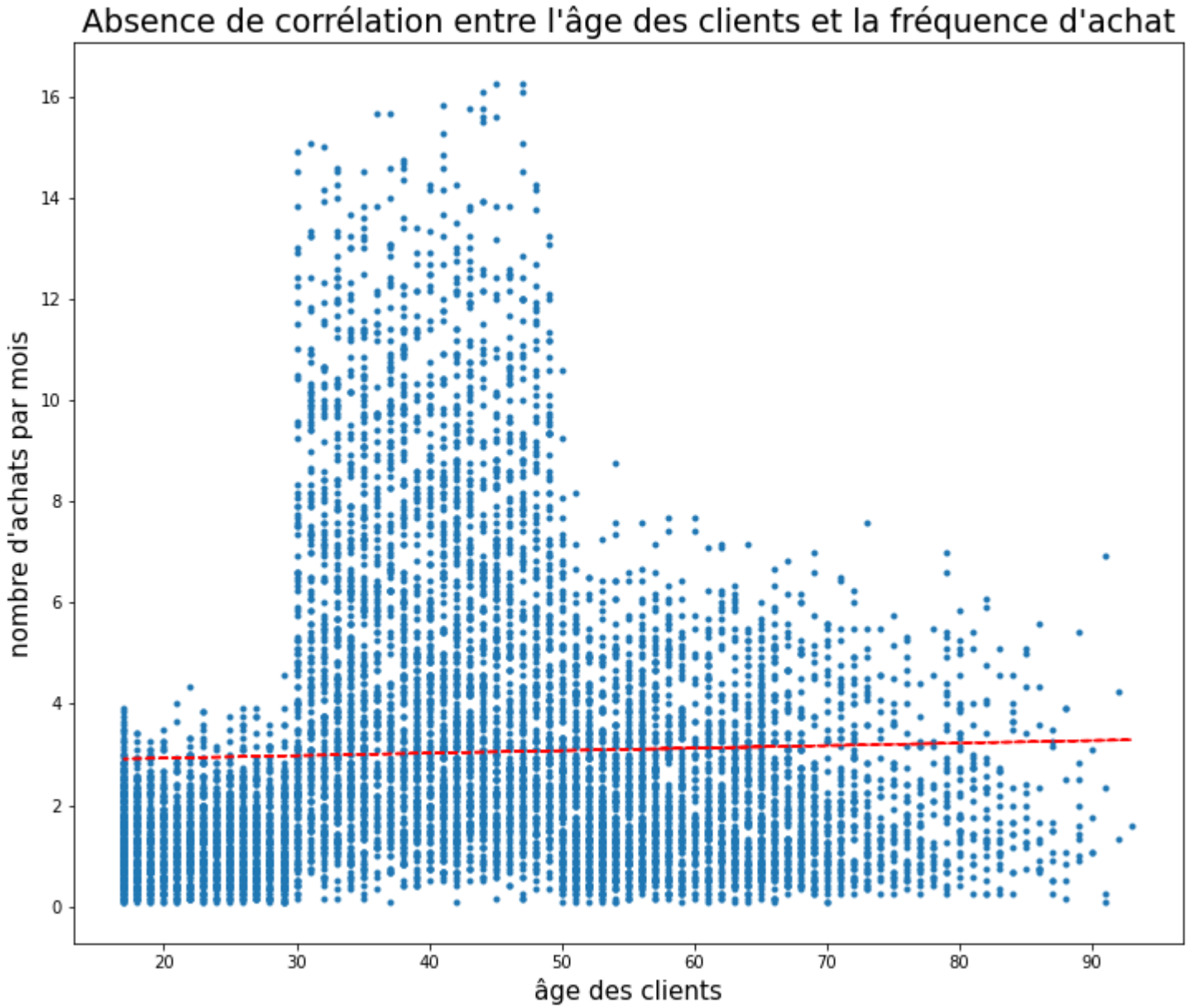
Le montant total du panier ne varie pas en fonction de l'âge, donc pas de corrélation.

$\eta 2 = 0,04$



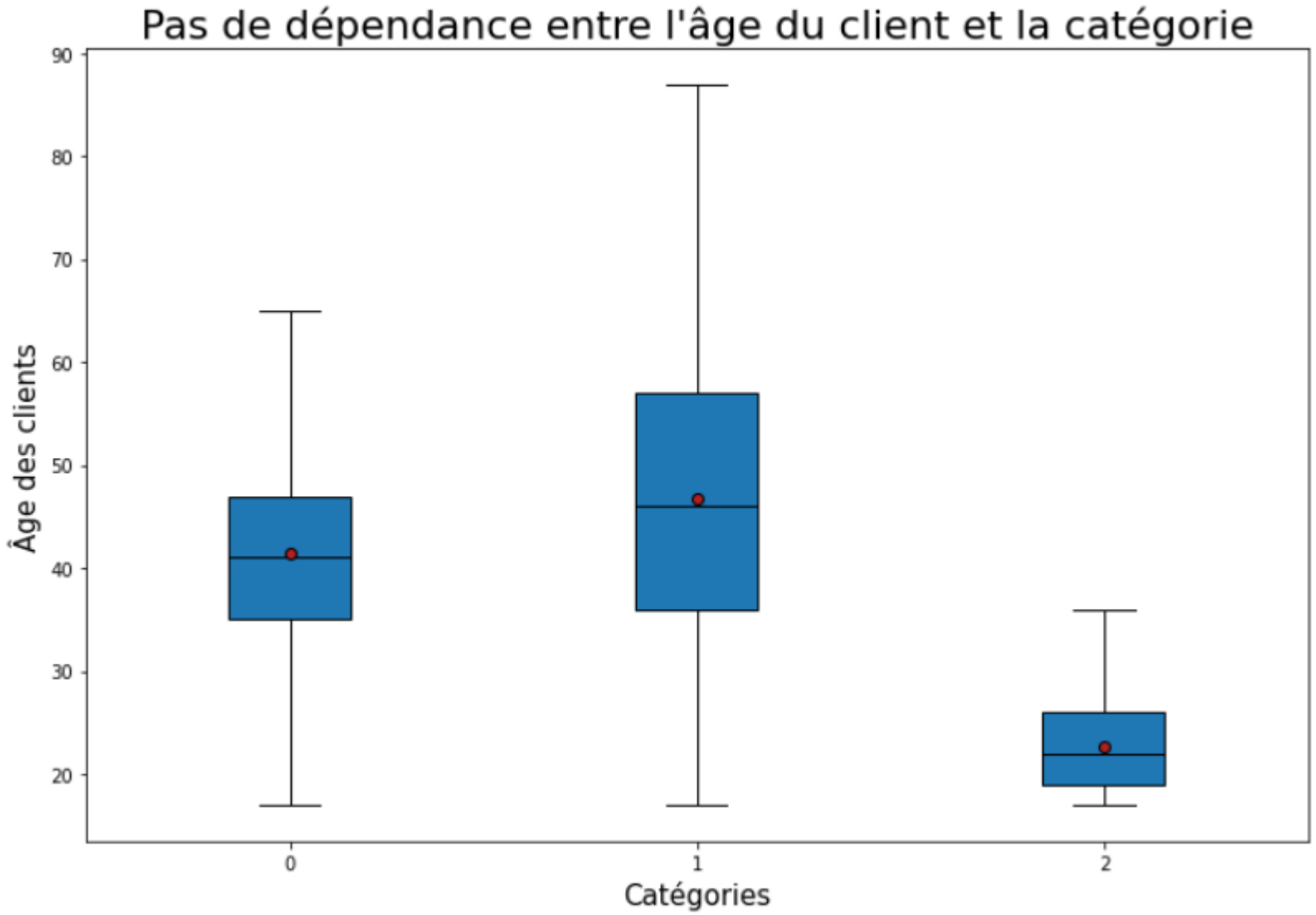
On observe une forte corrélation entre l'âge et le panier moyen. En effet, plus l'âge évolue plus le panier moyen diminue.

$\eta 2 = 0,3$



Absence totale de corrélation entre l'âge des clients t la fréquence d'achats.

$r^2 = 0,03$



Il n’ y a une certaine corrélation entre l’âge des clients et la catégorie.
On constate que la catégorie 1 intéresse tous les âges alors que la catégorie 2 n’intéresse que les jeunes de moins de 40 ans.

$\eta 2 = 0,12$

V- CONCLUSION

- Les 32 – 55 ans dépensent relativement plus, plus fréquemment et le panier moyen contient plus d'articles que pour les autres tranches d'âge. Mais ils sont davantage intéressés par les catégories 1 et 0, qui ont des prix moins élevés que la catégorie 2.

- Ce sont les 18-30 ans qui achètent davantage des produits de la catégorie 2, qui sont les produits les plus chers.

Ce sont les produits de la catégorie 1 qui intéressent le plus de clients différents, il s'agit d'une "catégorie tout public".

Nous ne pouvons pas déterminer de différence majeur entre chacun les sexes.