



RESTER LIVRES

1 novembre, 2021

ANALYSE

DES

VENTES

SOMMAIRE

Analyse des ventes | 01/11/2021

MISSION 1

PREPROCESSING

- Exploration des données
- nettoyage des données
- Traitement des valeurs manquantes
- Ajouts de valeurs supplémentaires

MISSION 2

ANALYSE

- Segmentation des clients
- Clients Professionnel
- Clients Particulier

MISSION 3

TEST STATISTIQUE

- Corrélation
- CHI-2
- ANOVA
- Conclusion



RESTER LIVRES

MISSION 1

PREPROCESSING

01



RESTER LIVRES

EXPLORATION DES DONNÉES



3 jeux de donnée nous on était fournis

Clients

- 8623 individus
- homme et femme de 17 à 92 ans
- aucune valeur manquante
- aucune anomalie détectée
- aucun doublons

produits

- 3287 individus
- 3 catégories de produit
- des prix qui s'étale de -1 a 300 €
- aucune valeur manquante
- aucun doublons

transaction

- 337016 individus
- 74 valeurs aberrantes
- aucune valeur manquante
- 126 doublons

jointure Externe

Jointure :

jointure entre client + produit (on conserve que les clés de transactions) les clients qui n'ont pas fait d'achats et les produits invendus sont écartés

la clé primaire reste celle de transaction : 'date' + 'client_id',

```
Entrée [29]: df = transaction.merge(client, how='left', on='client_id').merge(produit, how='left', on='id_prod')
```

```
Entrée [30]: df
```

Out[30]:

			id_prod	date	session_id	client_id	sex	birth	price	categ
0	0_1483	2021-04-10 18:37:28.723910	s_18746	c_4450	f	1977	4.99	0.00		
1	2_226	2022-02-03 01:55:53.276402	s_159142	c_277	f	2000	65.75	2.00		
2	1_374	2021-09-23 15:13:46.938559	s_94290	c_4270	f	1979	10.71	1.00		
3	0_2186	2021-10-17 03:27:18.783634	s_105936	c_4597	m	1963	4.20	0.00		
4	0_1351	2021-07-17 20:34:25.800563	s_63642	c_1242	f	1980	8.99	0.00		
...
336885	1_671	2021-05-28 12:35:46.214839	s_40720	c_3454	m	1969	31.99	1.00		
336886	0_759	2021-06-19 00:19:23.917703	s_50568	c_6268	m	1991	22.99	0.00		
336887	0_1256	2021-03-16 17:31:59.442007	s_7219	c_4137	f	1968	11.03	0.00		
336888	2_227	2021-10-30 16:50:15.997750	s_112349	c_5	f	1994	50.99	2.00		
336889	0_1417	2021-06-26 14:38:19.732946	s_54117	c_6714	f	1968	17.99	0.00		

336890 rows × 8 columns



IDENTIFICATION DES VALEURS MANQUANTES

La jointure met en évidence certain point qui était absent des 3 autres dataframe

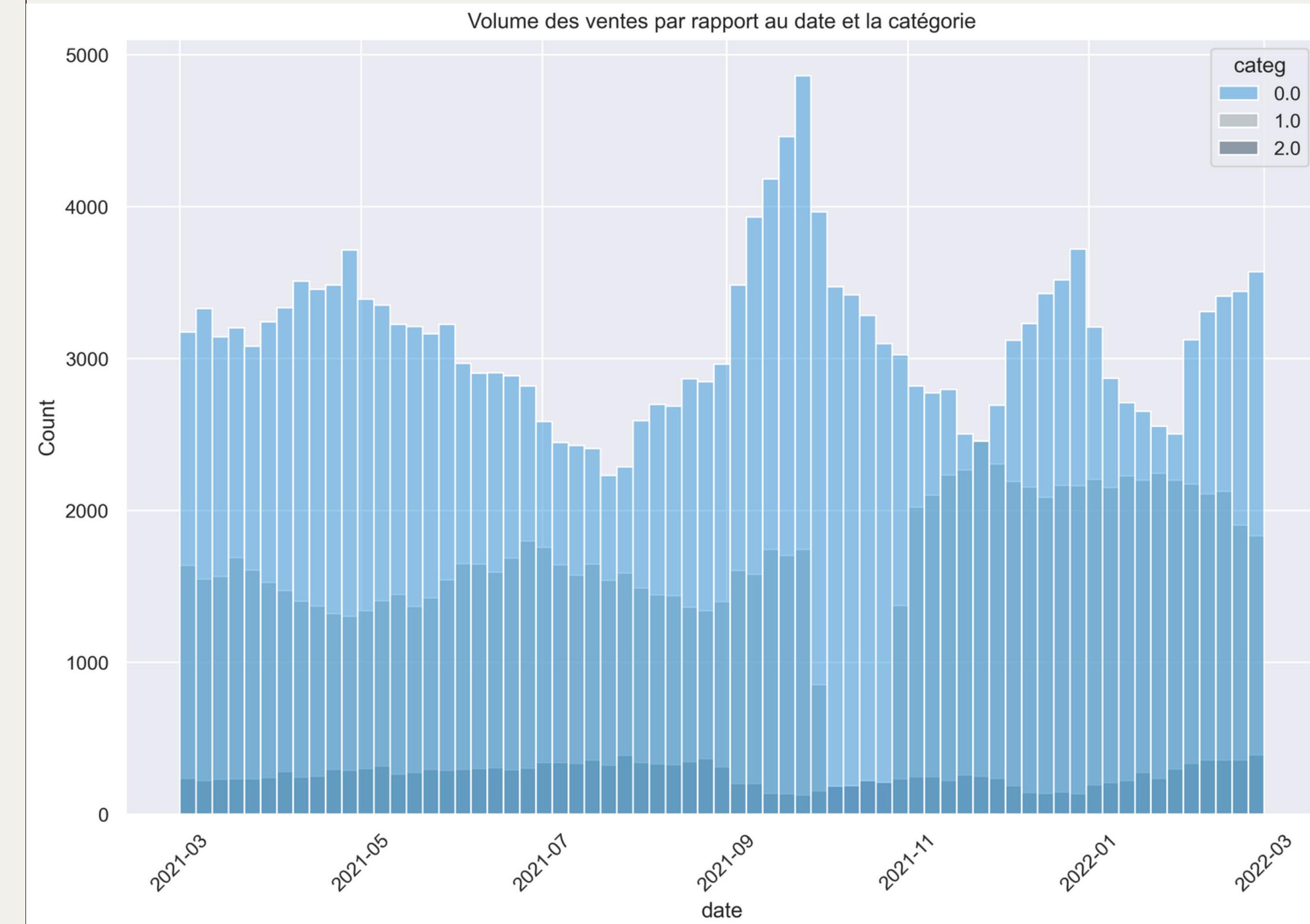
Différence entre les clés étrangère :

- 21 clients inactifs
- 22 livre invendue
- 1 livre vendue non référencé

96 valeurs manquantes qui correspondent au produit 0_2245

Date manquante :
la catégorie 1 est introuvable au mois d'octobre

DATE MANQUANTE



absence de la catégorie 1 au mois d'octobre



VALEURS MANQUANTES

Les valeur manquante

```
Entrée [42]: p_0_2245 = df[df['id_prod']=='0_2245']  
p_0_2245
```

Out[42]:

	index	id_prod	date	session_id	client_id	sex	birth	price	categ	
	6231	6235	0_2245	2021-06-17 03:03:12.668129	s_49705	c_1533	m	1972	NaN	NaN
	10797	10802	0_2245	2021-06-16 05:53:01.627491	s_49323	c_7954	m	1973	NaN	NaN
	14045	14051	0_2245	2021-11-24 17:35:59.911427	s_124474	c_5120	f	1975	NaN	NaN
	17480	17486	0_2245	2022-02-28 18:08:49.875709	s_172304	c_4964	f	1982	NaN	NaN
	21071	21078	0_2245	2021-03-01 00:09:29.301897	s_3	c_580	m	1988	NaN	NaN
	
	322523	322597	0_2245	2021-04-06 19:59:19.462288	s_16936	c_4167	f	1979	NaN	NaN
	329226	329300	0_2245	2021-03-30 23:29:02.347672	s_13738	c_7790	f	1983	NaN	NaN
	330297	330371	0_2245	2021-12-03 14:14:40.444177	s_128815	c_6189	f	1984	NaN	NaN
	335331	335405	0_2245	2021-04-27 18:58:47.703374	s_26624	c_1595	f	1973	NaN	NaN
	336020	336094	0_2245	2021-05-01 03:35:03.146305	s_28235	c_5714	f	1972	NaN	NaN

96 rows × 9 columns

Les Valeurs manquantes du produit 0_2245 :
prix / catégorie



RESTER LIVRES

NETTOYAGE DES DONNÉES

SUPPRESSION DES VALEURS DE TEST



Entrée [35]: # les données de test

```
lt = df[
    (df['date'].str.contains('test_')) &
    (df['price']<=0) &
    (df['id_prod']=='T_0') &
    (df['session_id']=='s_0')]
lt
```

Out[35]:

	id_prod	date	session_id	client_id	sex	birth	price	categ
1431	T_0	test_2021-03-01 02:30:02.237420	s_0	ct_1	m	2001	-1.00	0.00
2365	T_0	test_2021-03-01 02:30:02.237446	s_0	ct_1	m	2001	-1.00	0.00
2895	T_0	test_2021-03-01 02:30:02.237414	s_0	ct_1	m	2001	-1.00	0.00
5955	T_0	test_2021-03-01 02:30:02.237441	s_0	ct_0	f	2001	-1.00	0.00
7283	T_0	test_2021-03-01 02:30:02.237434	s_0	ct_1	m	2001	-1.00	0.00
...
264229	T_0	test_2021-03-01 02:30:02.237416	s_0	ct_1	m	2001	-1.00	0.00
288815	T_0	test_2021-03-01 02:30:02.237415	s_0	ct_1	m	2001	-1.00	0.00
293003	T_0	test_2021-03-01 02:30:02.237421	s_0	ct_0	f	2001	-1.00	0.00
298292	T_0	test_2021-03-01 02:30:02.237423	s_0	ct_1	m	2001	-1.00	0.00
317233	T_0	test_2021-03-01 02:30:02.237448	s_0	ct_0	f	2001	-1.00	0.00

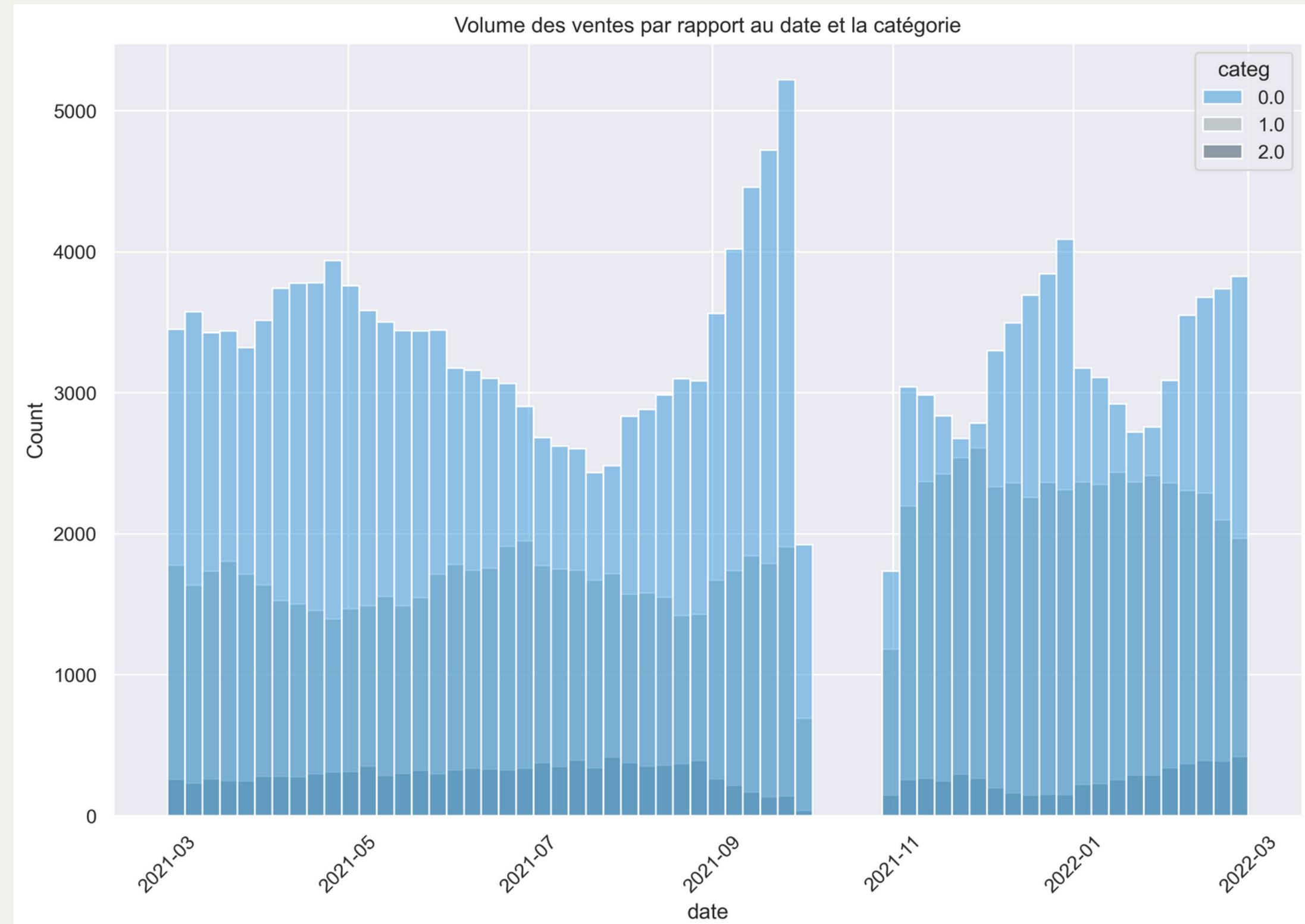
74 rows × 8 columns

Entrée [36]: lbf = len(df) # Nombre de lignes dans df avant suppression des lignes tests

```
df = df.drop(lt.index).reset_index() # Suppression des lignes tests
print(lbf - len(df), 'lignes tests supprimées') # Nombre de lignes dans data après nettoyage
```

74 lignes tests supprimées

SUPPRESSION DU MOIS D'OCTOBRE



6 % du dataset a été supprimé



RESTER LIVRES

TRAITEMENT DES DONNÉES

IMPUTATION PAR UNE VALEUR FIXE



```
: df_test = df.dropna() # sample des commandes sans le produit 0_2245

# Conversion en string des 2 premiers caractères de la valeur de 'id_prod'
df_test['id_prod'] = df_test['id_prod'].str[:2]

print('Préfixes de \'id_prod\' :') # Pour chaque catégorie, le préfixe unique de 'id_prod'
for i in df_test['categ'].unique():
    print('- catégorie', i, ':',
          df_test[df.dropna()['categ']==i]['id_prod'].unique())

Préfixes de 'id_prod' :
- catégorie 0.0 : ['0_']
- catégorie 2.0 : ['2_']
- catégorie 1.0 : ['1_']
```

On peut également déterminer une valeur fixe pour les 2 variables. Pour la catégorie, on peut se fier aux préfixes des identifiants de produits : 0_, 1_ et 2_. Ces préfixes correspondent invariablement à la catégorie du produit concerné. On choisit donc 0 comme catégorie du produit 0_2245.

```
: # remplacement des valeurs manquantes de la variable 'categ' par 0
df['categ'].replace(np.nan, 0, inplace=True)
produit_2245 = df[df['id_prod']=='0_2245']
produit_2245.sample(3)
```

IMPUTATION PAR LA MÉDIANE



```
Entrée [49]: print('Catégorie 0 :',
  '\n- prix moyen :', round(df[df['categ']==0]['price'].mean(), 2), # prix moyen
  '\n- prix médian :', df[df['categ']==0]['price'].median(), # prix médian
  '\n- mode :', df[df['categ']==0]['price'].mode().values[0]) # prix le plus fréquent
```

```
Catégorie 0 :
- prix moyen : 10.65
- prix médian : 9.99
- mode : 4.99
```

on décide de faire une imputation par la médiane pour traité ces valeur manquante

```
Entrée [50]: df['price'].replace(np.nan, 9.99, inplace=True)
```

AJOUTS DE VALEURS SUPPLÉMENTAIRE



```
df['mois'] = pd.DatetimeIndex(df['date']).month

df['date_fixe'] = df['date'].dt.date # Variable temporaire

df = df.merge(
    df.groupby('client_id').count()['date'].reset_index().rename(columns={'date': 'total_ventes'}),
    how='left', on='client_id')

df['ventes_mensuelles'] = round(df['total_ventes'] / 11)

df = df.merge(
    df.pivot_table(
        index=['client_id', 'date_fixe'],
        values='price',
        aggfunc='count').reset_index().pivot_table(
        index='client_id').reset_index().rename(
        columns={'price': 'taille_panier_moyen'}),
    on='client_id', how='left')

df = df.merge(
    df.pivot_table(
        index=['client_id', 'date_fixe'],
        values='price').reset_index().pivot_table(
        index='client_id').reset_index().rename(
        columns={'price': 'panier_moyen'}),
    on='client_id', how='left').drop('date_fixe', axis=1)

df = df.merge(
    df.pivot_table(
        index='client_id', values='price',
        aggfunc='sum').reset_index().rename(
        columns={'price': 'total_achats'}),
    on='client_id', how='left')
```

AJOUTS DE VALEURS SUPPLÉMENTAIRE



```
#age des client
year = datetime.now().year
df['age'] = year - df['birth']

#classification par tranche d'âge
"""
le découpage peut se faire par tranche d'âge :
0-3 ans, 4-8 ans, 9-12 ans, 13-17 ans, 18-24 ans, 25-34 ans, 35-44 ans, 45-54 ans, 55-64 ans, 65-74 ans,
75-84 ans et les 85 ans et plus.
ou bien plus simplement faire des tranche d'age par décennie:
18-30,
30-40,
40-50,
50-60,
70-80,
80+
"""

df['tranche_age'] = '18-30'
df['tranche_age'].loc[df[df['age']>=30].index] = '30-40'
df['tranche_age'].loc[df[df['age']>=40].index] = '40-50'
df['tranche_age'].loc[df[df['age']>=50].index] = '50-60'
df['tranche_age'].loc[df[df['age']>=60].index] = '60-70'
df['tranche_age'].loc[df[df['age']>=70].index] = '70-80'
df['tranche_age'].loc[df[df['age']>=80].index] = '80et+'
```

Dataframe Final

id_prod	date	session_id	client_id	sex	birth	price	categ	mois	total_ventes	ventes_mensuelles	taille_panier_moyen	panier_moyen	total_achats	age	tranche_age
2_175	2022-01-18 14:04:25.515681	s_151665	c_533	m	2004	60.99	2.00	1	7	1.00	1.75	50.79	358.14	17	18-30
0_1407	2021-09-02 15:56:25.398271	s_84097	c_3430	f	1973	13.99	0.00	9	133	12.00	2.96	14.10	1,832.42	48	40-50
1_282	2021-03-05 17:46:30.067204	s_2183	c_6113	f	1993	23.20	1.00	3	12	1.00	3.00	40.51	380.68	28	18-30
0_1380	2022-01-08 19:42:29.924784	s_146967	c_8289	f	1973	7.45	0.00	1	20	2.00	2.22	10.42	210.26	48	40-50
0_1345	2021-05-02 19:09:58.406191	s_28977	c_107	f	1984	17.99	0.00	5	69	6.00	2.56	12.58	886.84	37	30-40
1_628	2021-12-05 23:52:43.080276	s_130018	c_4114	f	1986	41.37	1.00	12	69	6.00	2.16	12.93	878.86	35	30-40
0_1532	2021-08-13 14:58:38.539051	s_75231	c_7436	f	1973	17.14	0.00	8	9	1.00	3.00	12.17	93.46	48	40-50
1_44	2022-02-19 13:56:45.396081	s_167623	c_1077	f	1985	14.49	1.00	2	100	9.00	3.03	13.83	1,322.76	36	30-40
0_1453	2021-05-02 03:27:38.970363	s_28680	c_4445	m	1991	7.99	0.00	5	38	3.00	4.75	13.72	528.21	30	30-40
1_250	2021-07-06 03:21:46.789928	s_58467	c_4429	f	1979	20.76	1.00	7	85	8.00	3.27	16.05	1,210.89	42	40-50

ANALYSE

02

SEGMENTATION DES CLIENTS

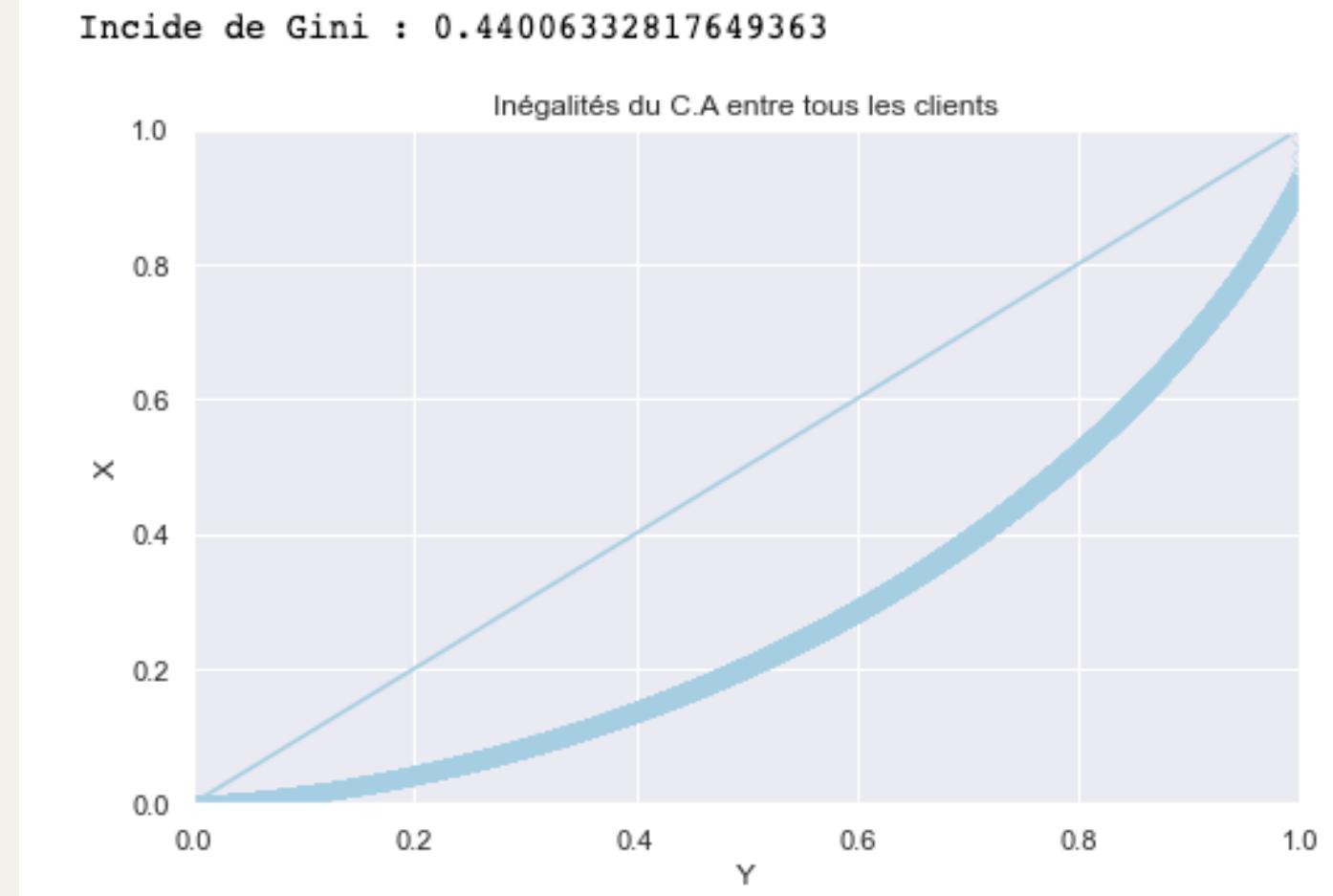


```
: ca = df.pivot_table(  
    index='client_id', values=[  
        'total_achats','ventes_mensuelles','taille_panier_moyen','total_ventes','panier_moyen'  
    ].sort_values(by='total_achats', ascending=False).reset_index()  
  
ca.head(10)
```

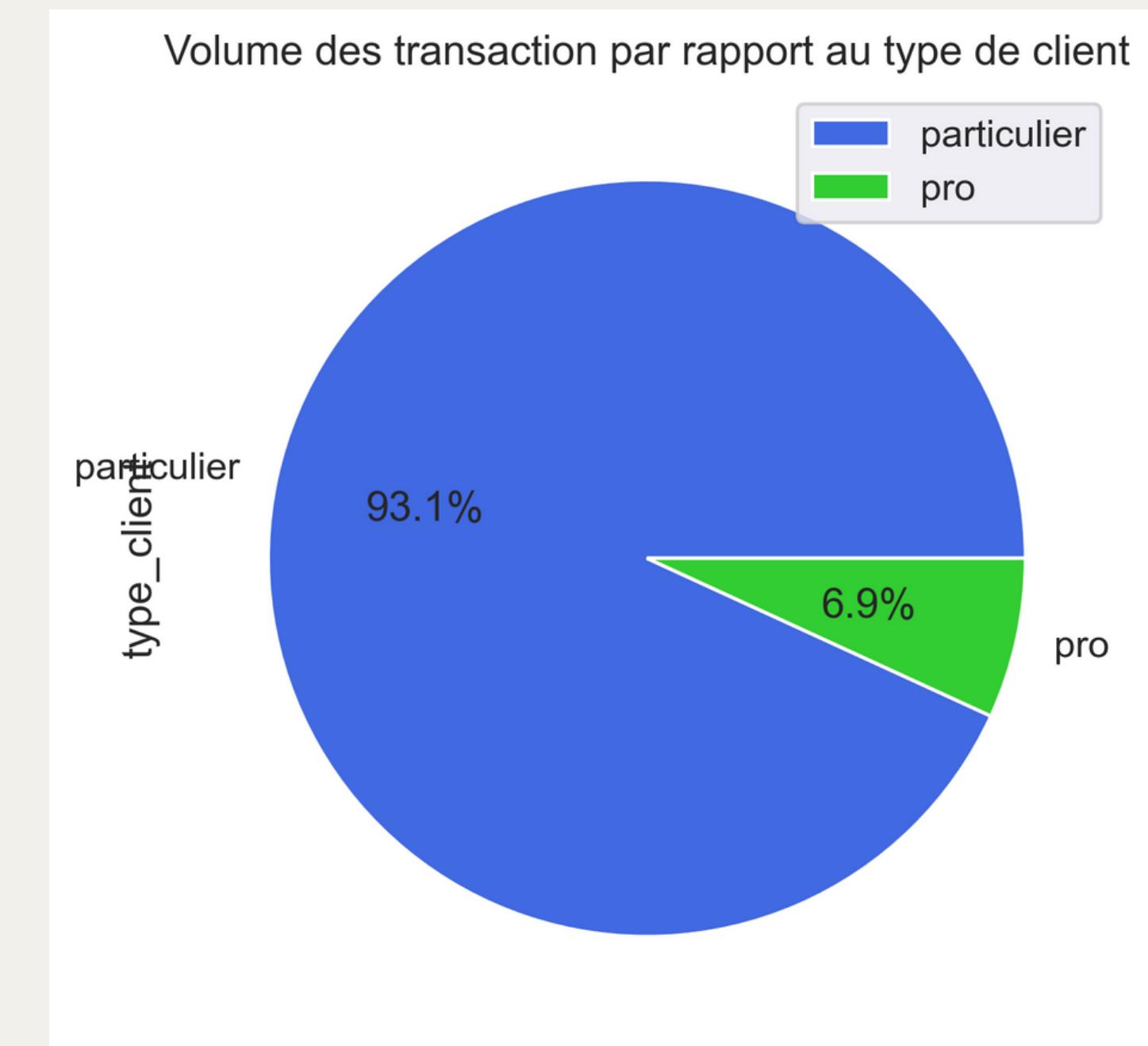
	client_id	panier_moyen	taille_panier_moyen	total_achats	total_ventes	ventes_mensuelles
0	c_1609	12.72	35.41	151,018.91	11861	1,078.00
1	c_4958	55.31	7.42	137,456.83	2463	224.00
2	c_6714	16.66	12.98	69,493.36	4193	381.00
3	c_3454	16.62	9.39	52,845.11	3145	286.00
4	c_8026	13.47	2.88	2,434.49	184	17.00
5	c_7421	13.58	2.83	2,406.17	178	16.00
6	c_7319	13.57	2.75	2,366.20	168	15.00
7	c_3263	13.82	3.05	2,346.34	177	16.00
8	c_8392	13.76	2.76	2,332.08	171	16.00
9	c_2899	53.69	1.62	2,313.54	47	4.00

Clients professionnels

Clients particuliers



CLIENT PROFESSIONNEL

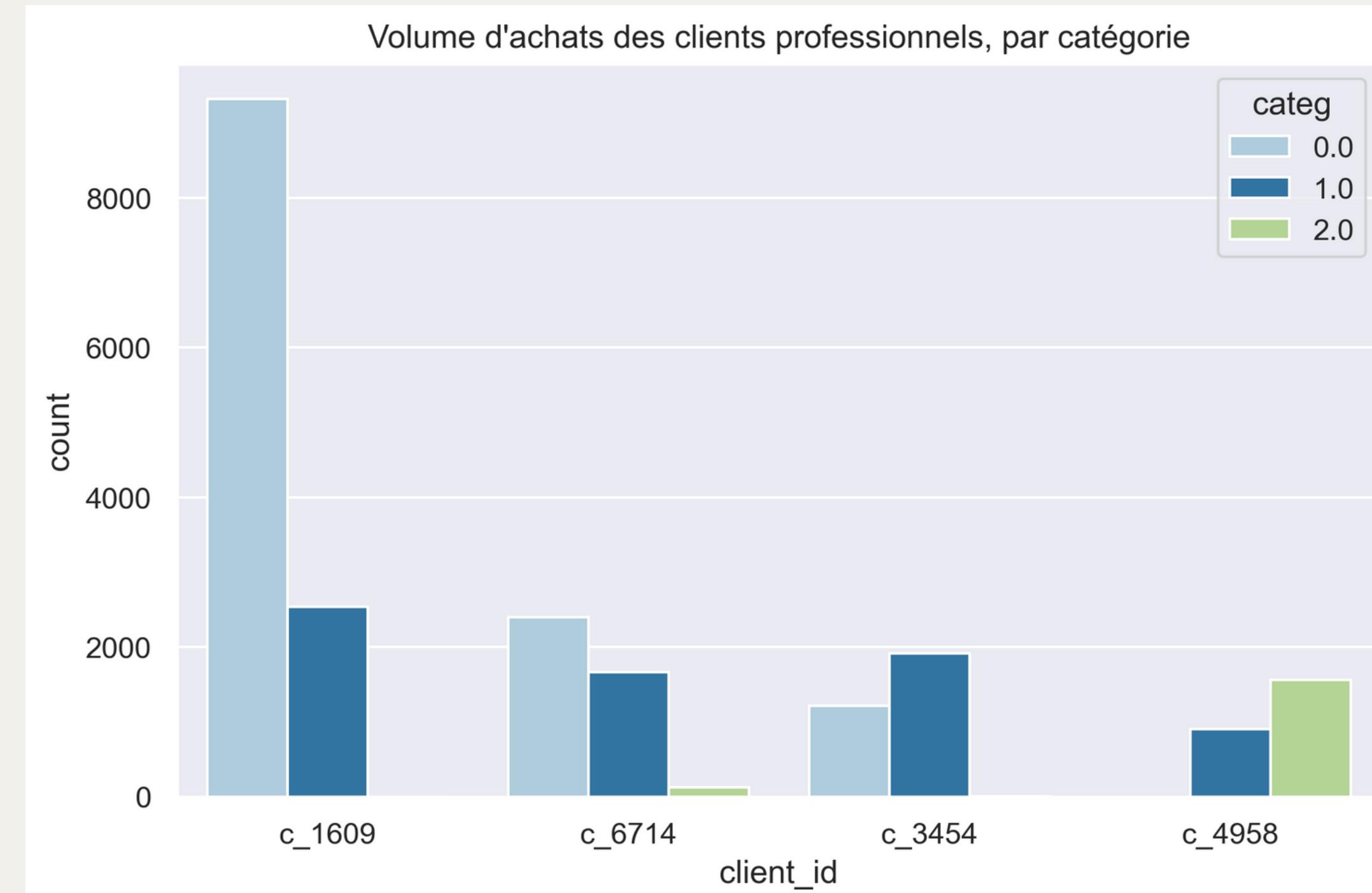


CLIENT PROFESSIONNEL

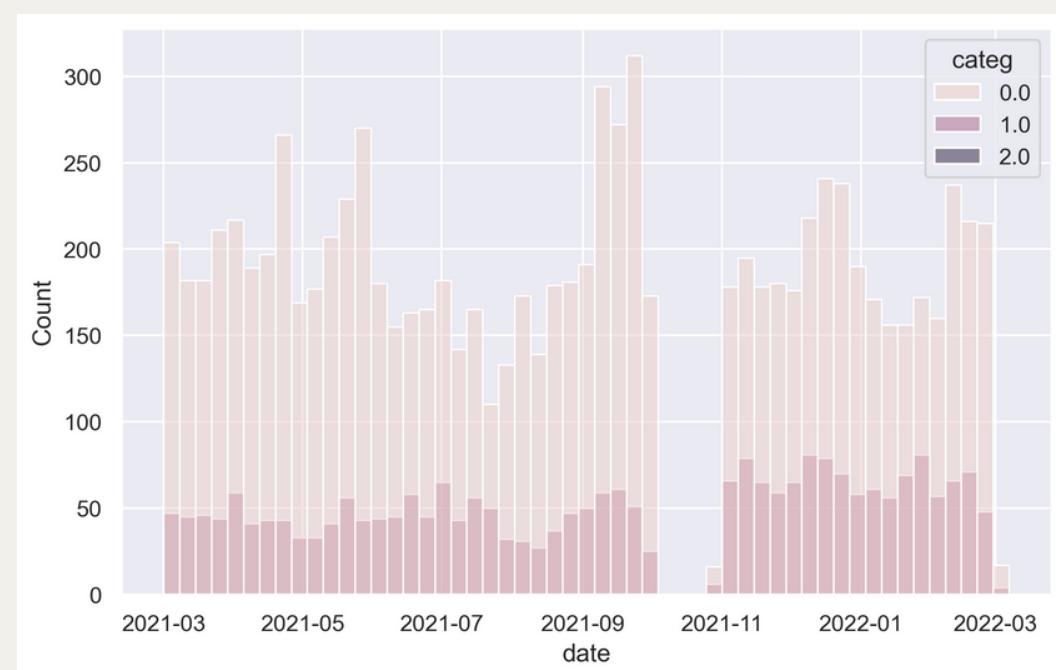


7,48%

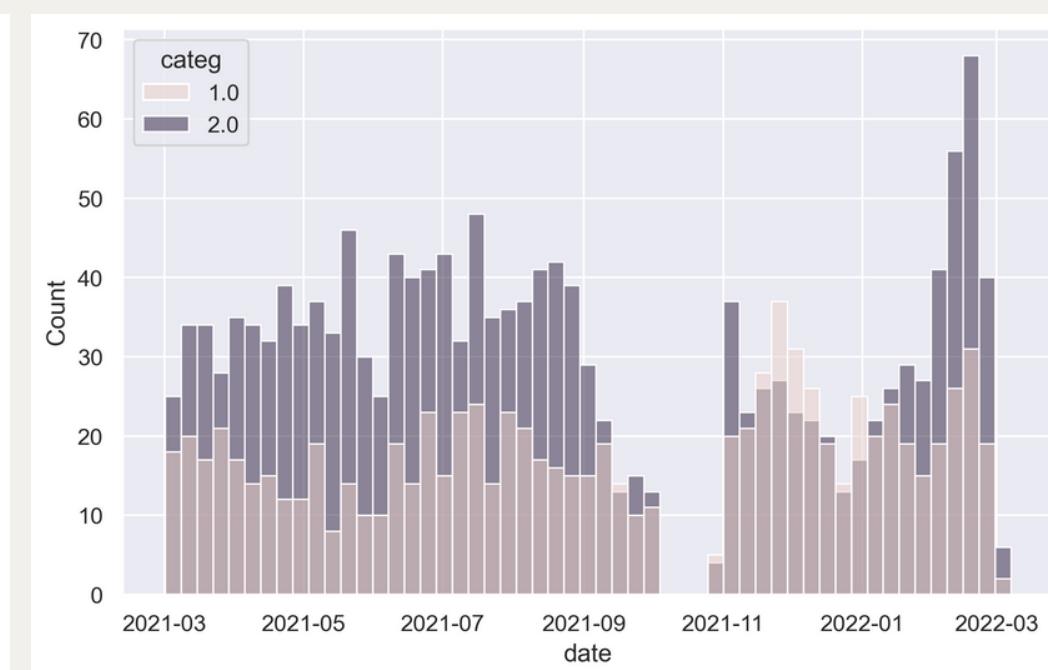
Le pourcentage du chiffre d'affaires que représentent les professionnelles



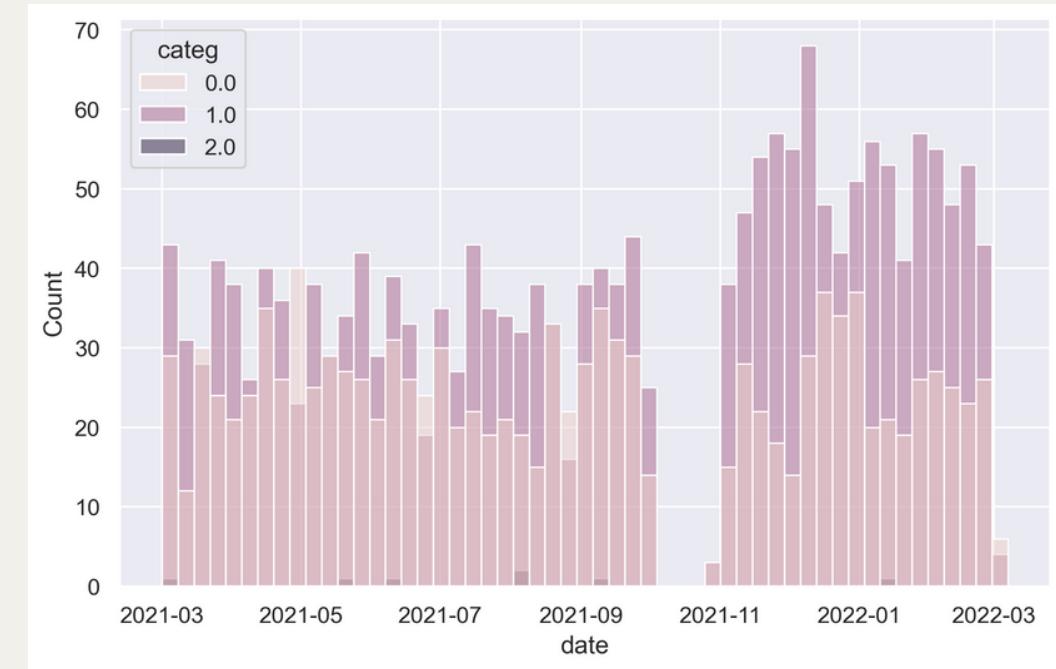
CLIENT PROFESSIONNEL



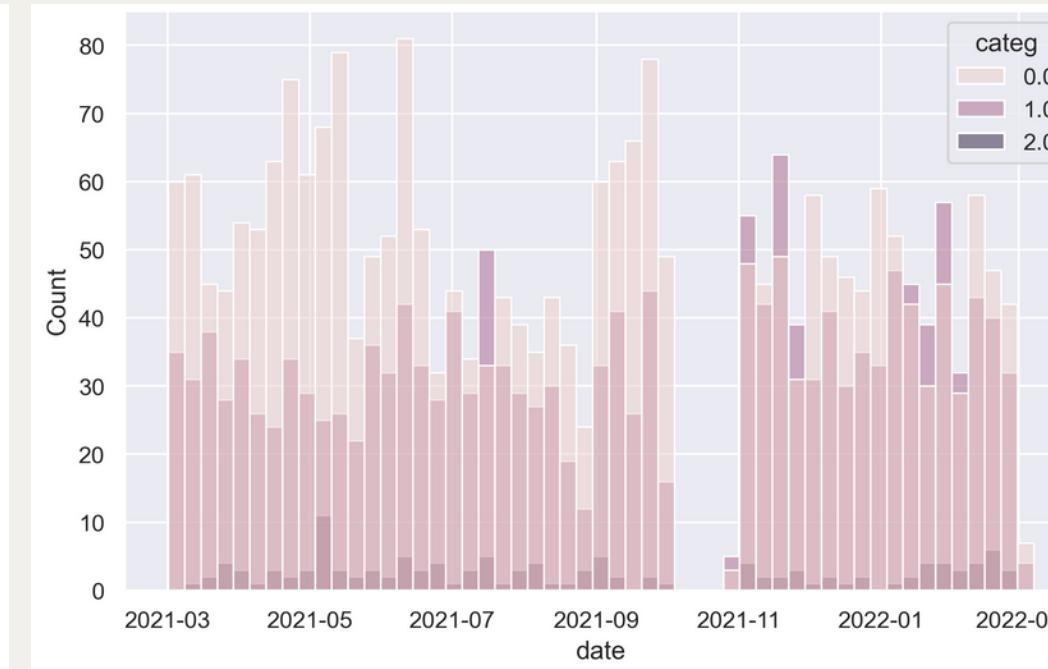
CLIENT C_1609



CLIENT C_6714



CLIENT C_3454



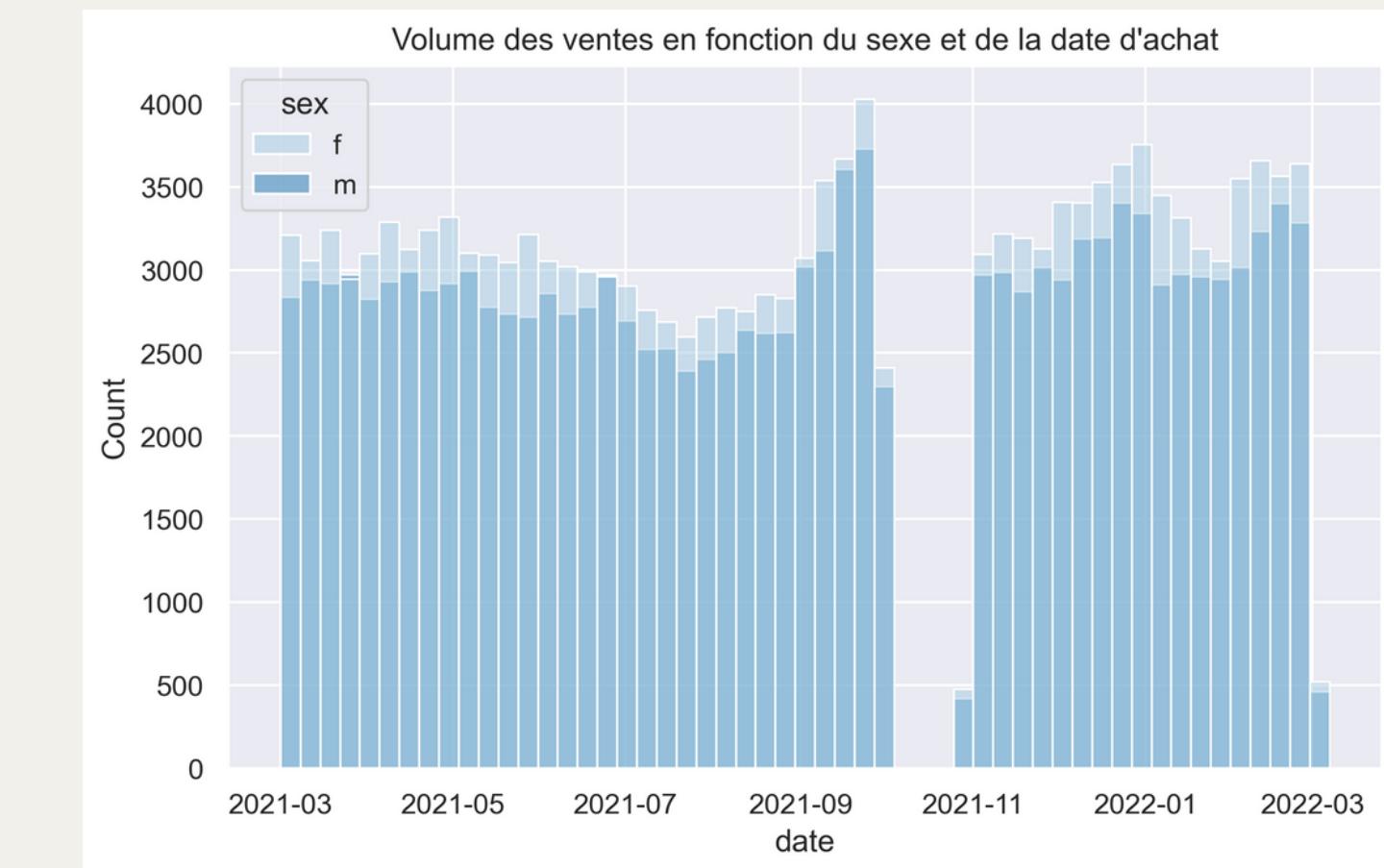
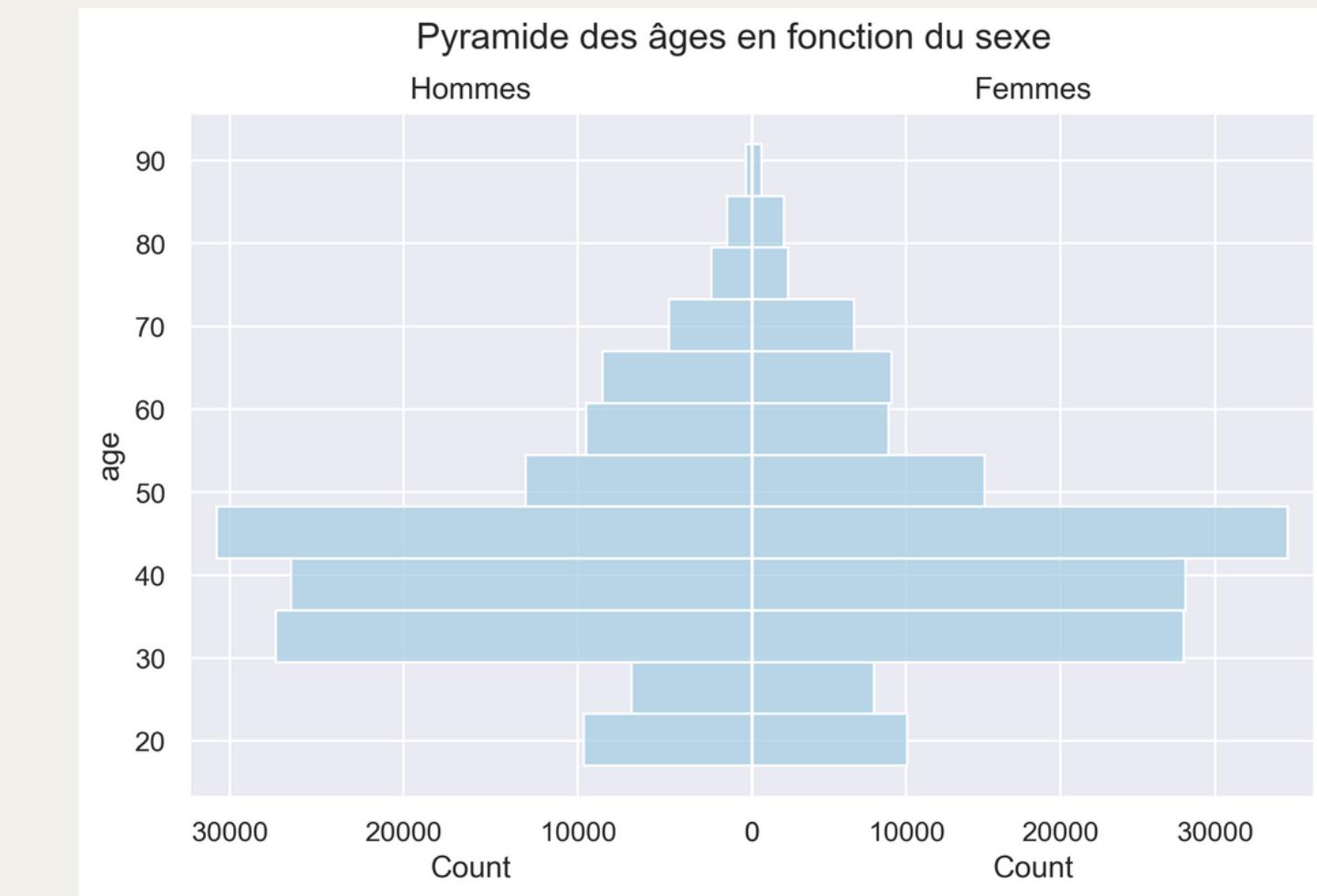
CLIENT C_4958

Clients professionnels

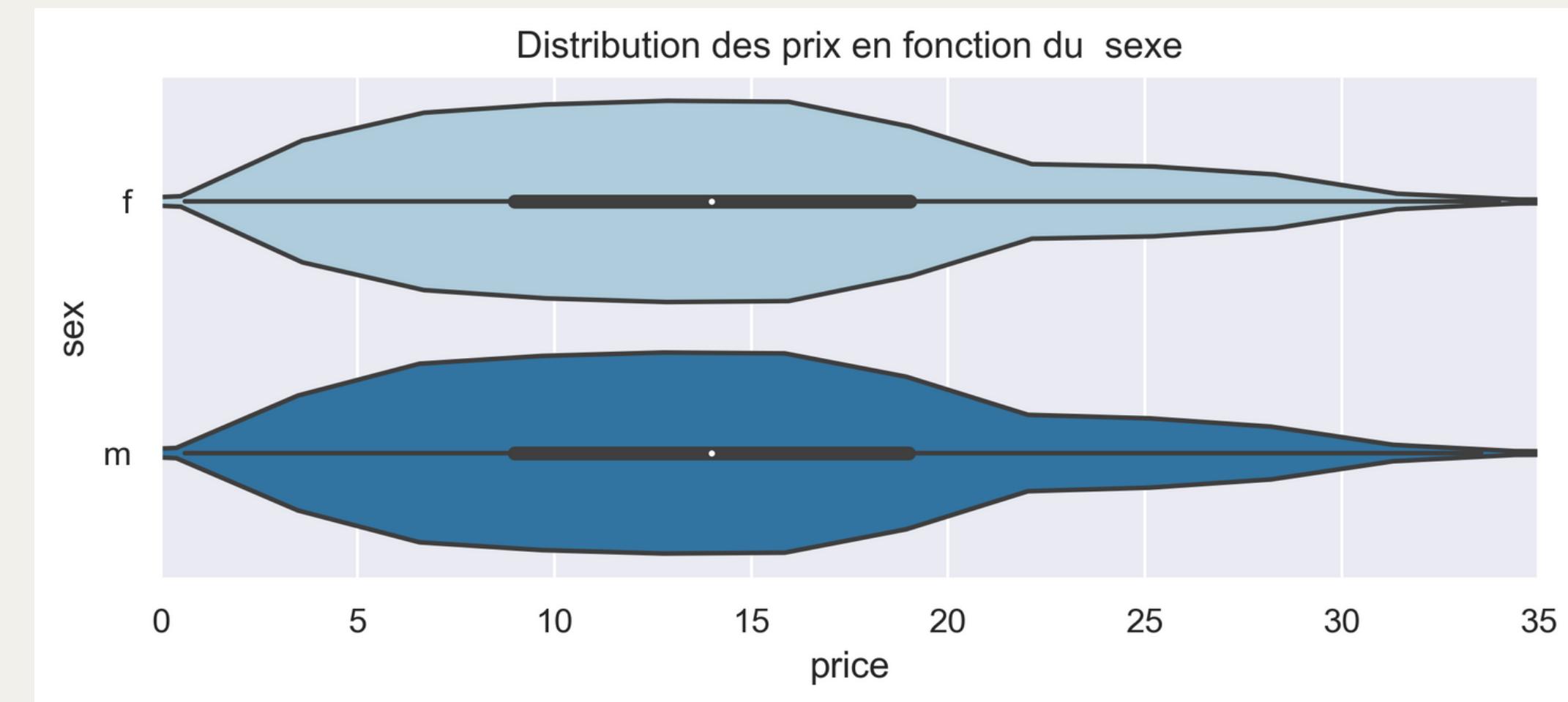


	1	2	3	4
identifiant client	1609	4958	3454	6714
\$ chiffre d'affaires annuel	151018€	137456€	69493€	52845€
👤 Catégorie avec le plus d'achat	0	0	1	2
✓ date avec le plus d'achat	septembre	septembre	décembre	février

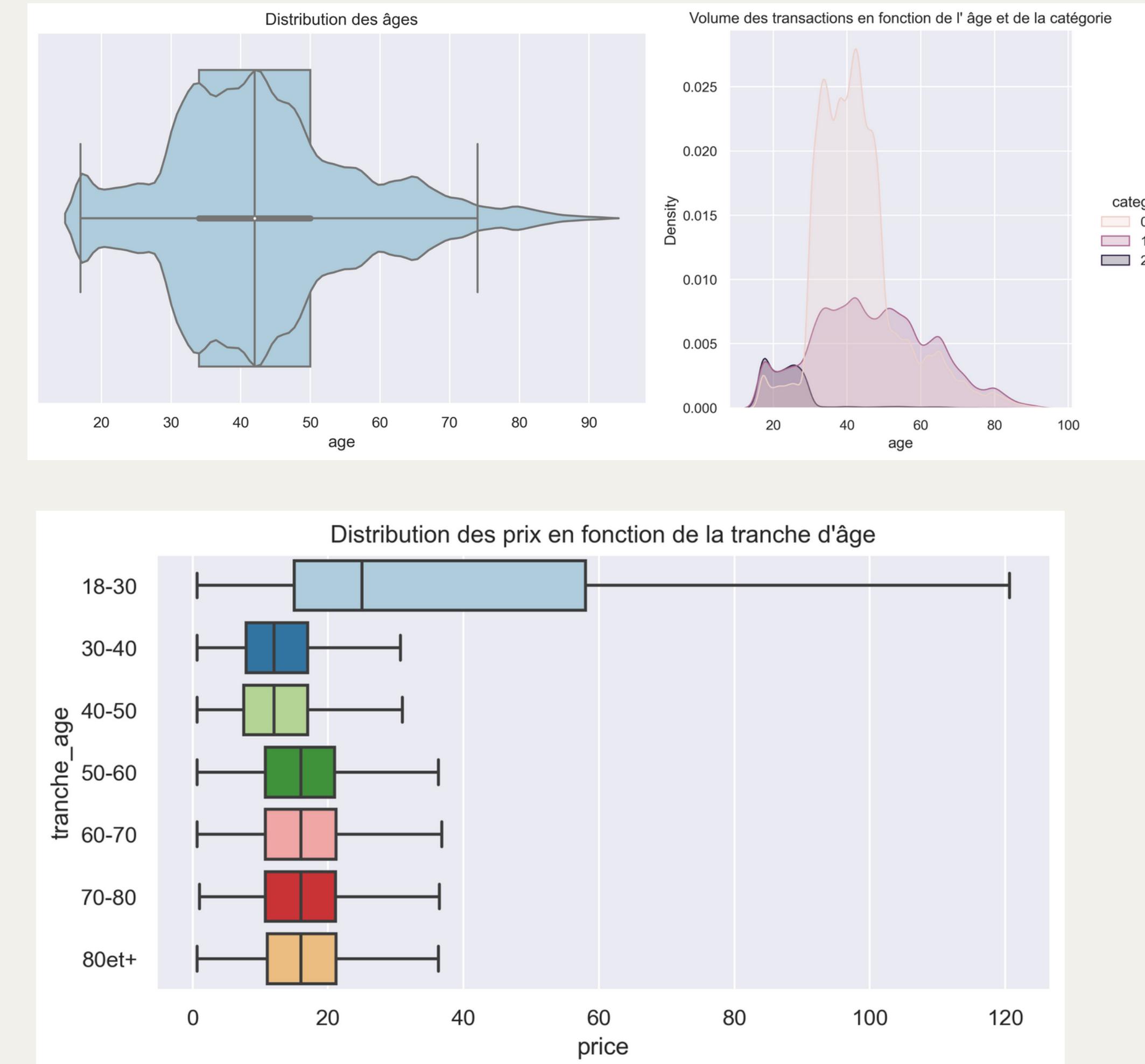
CLIENT PARTICULIER



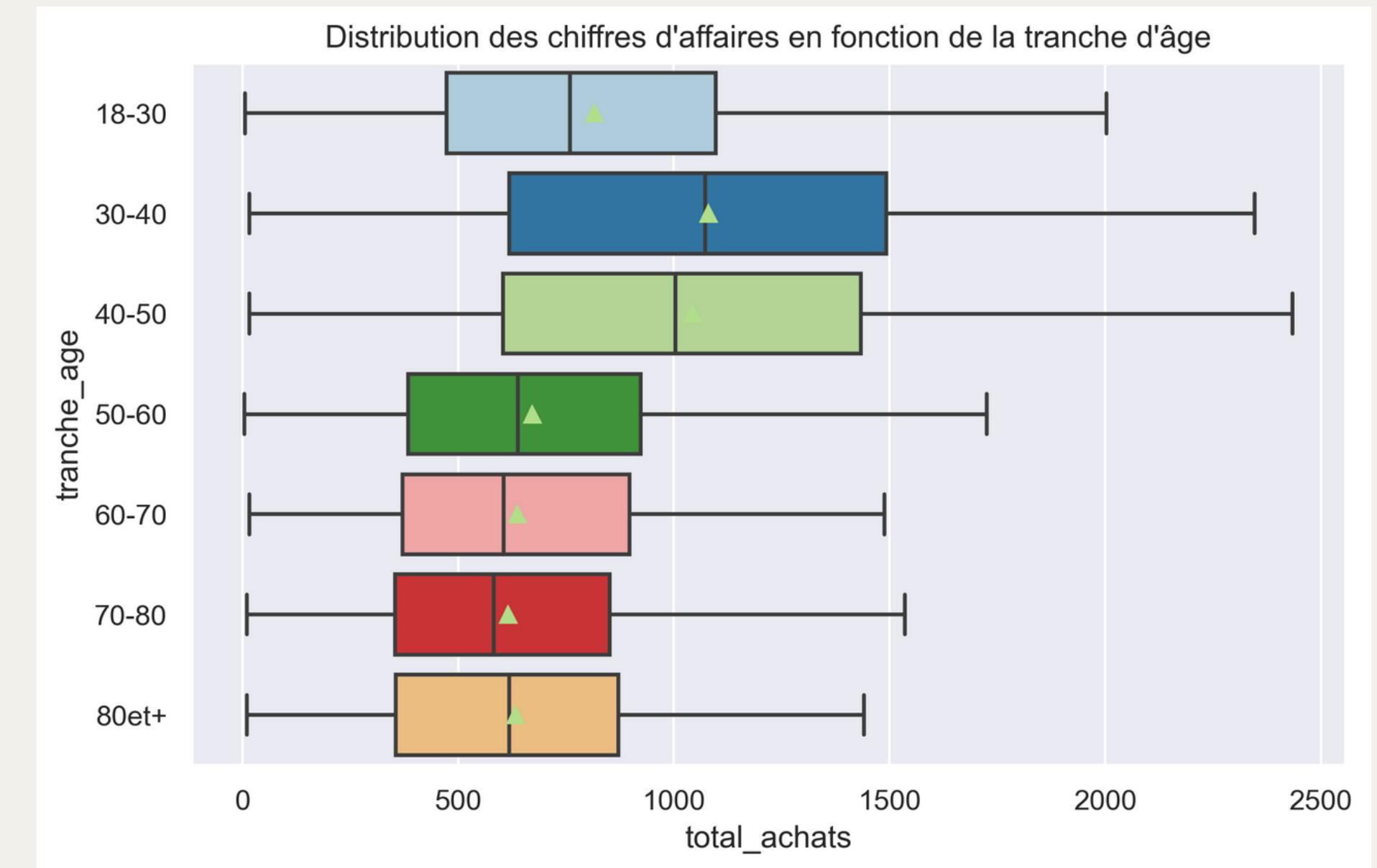
CLIENT PARTICULIER



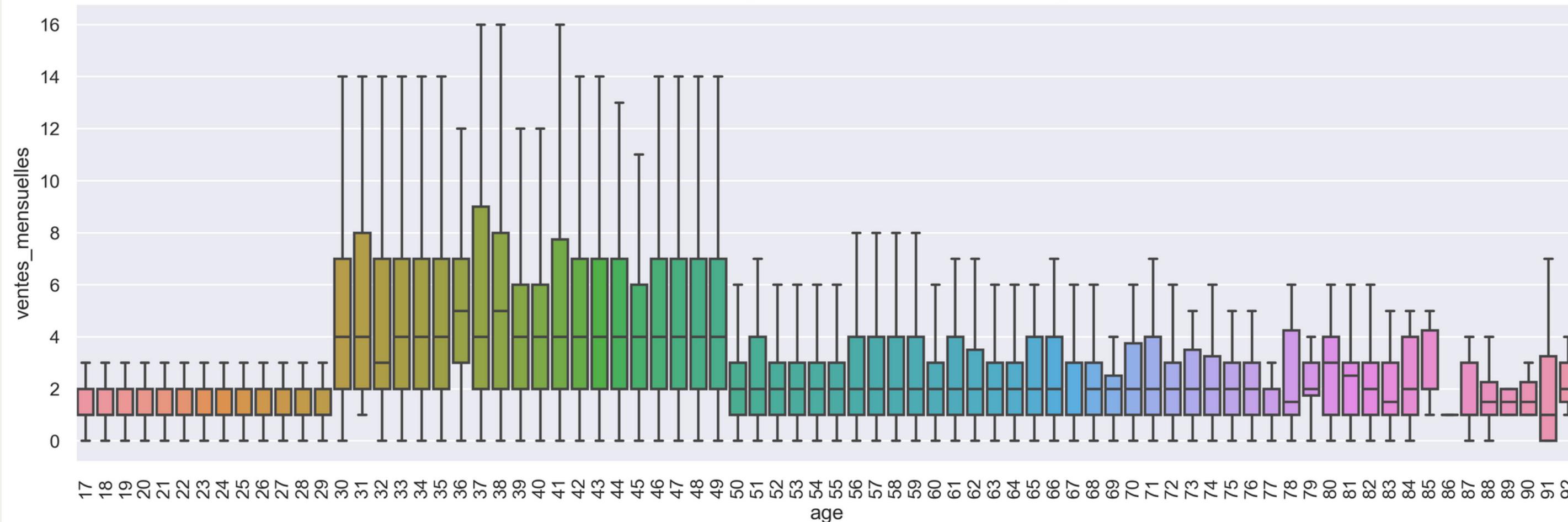
CLIENT PARTICULIER



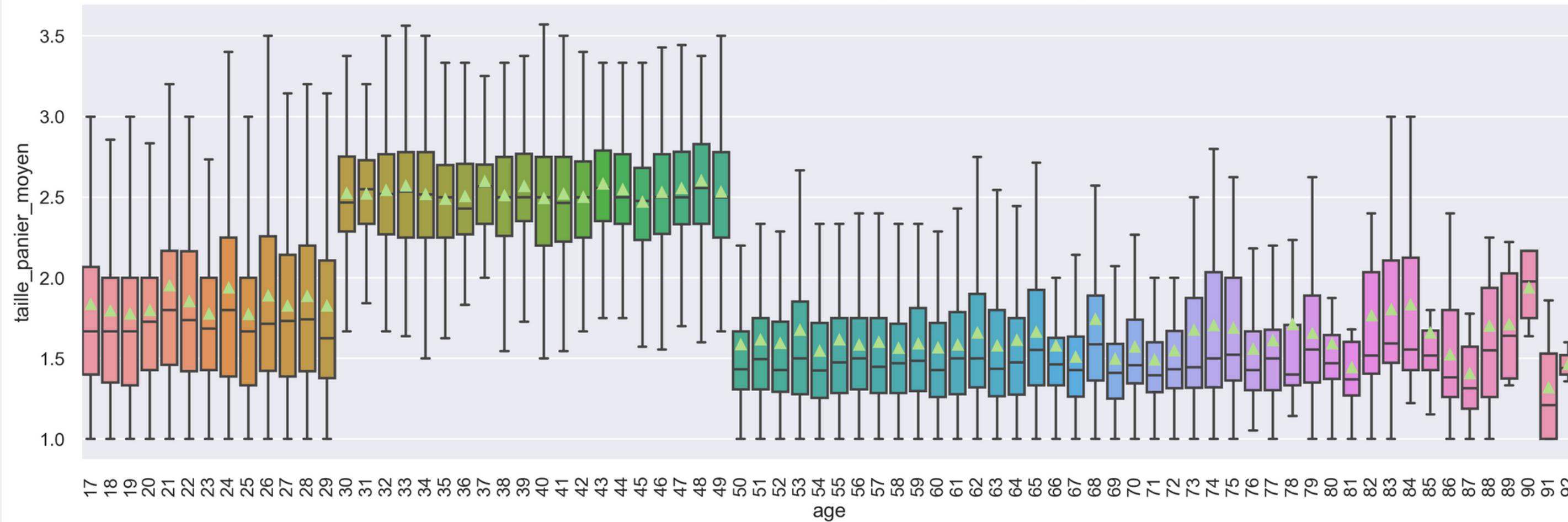
CLIENT PARTICULIER



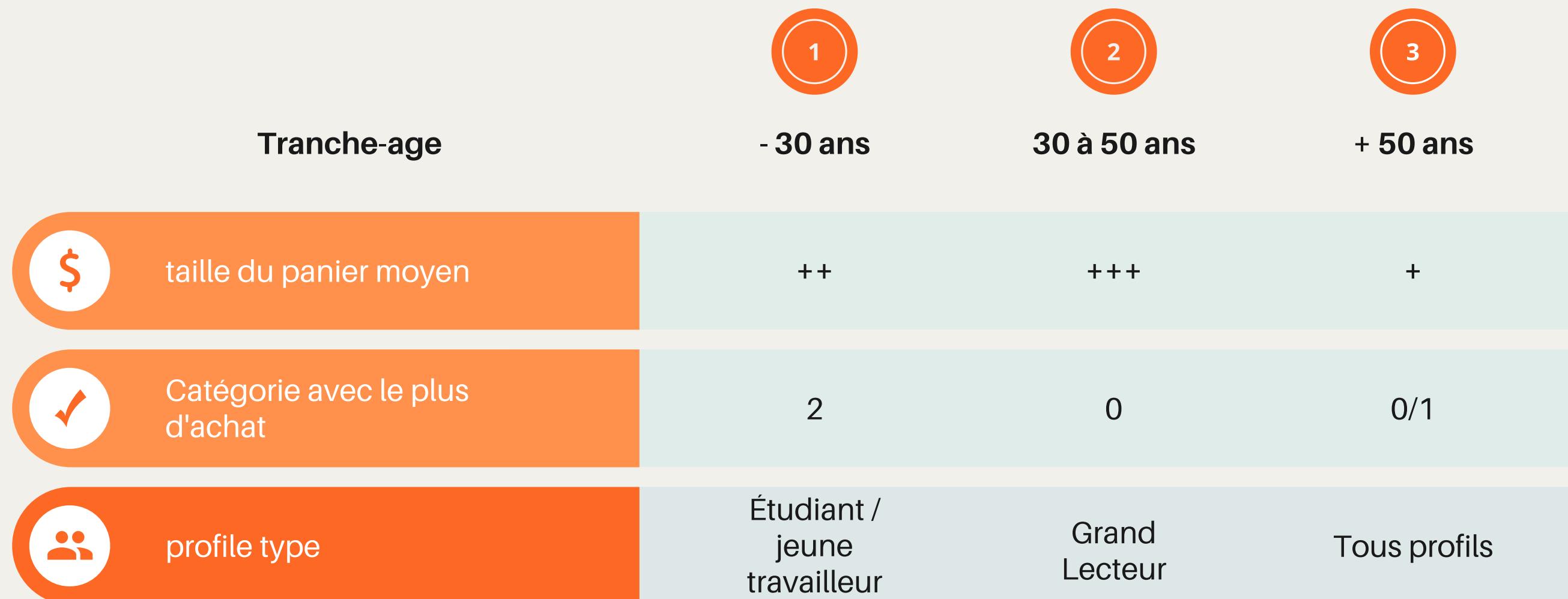
Distribution des fréquences d'achat, en fonction de l'âge



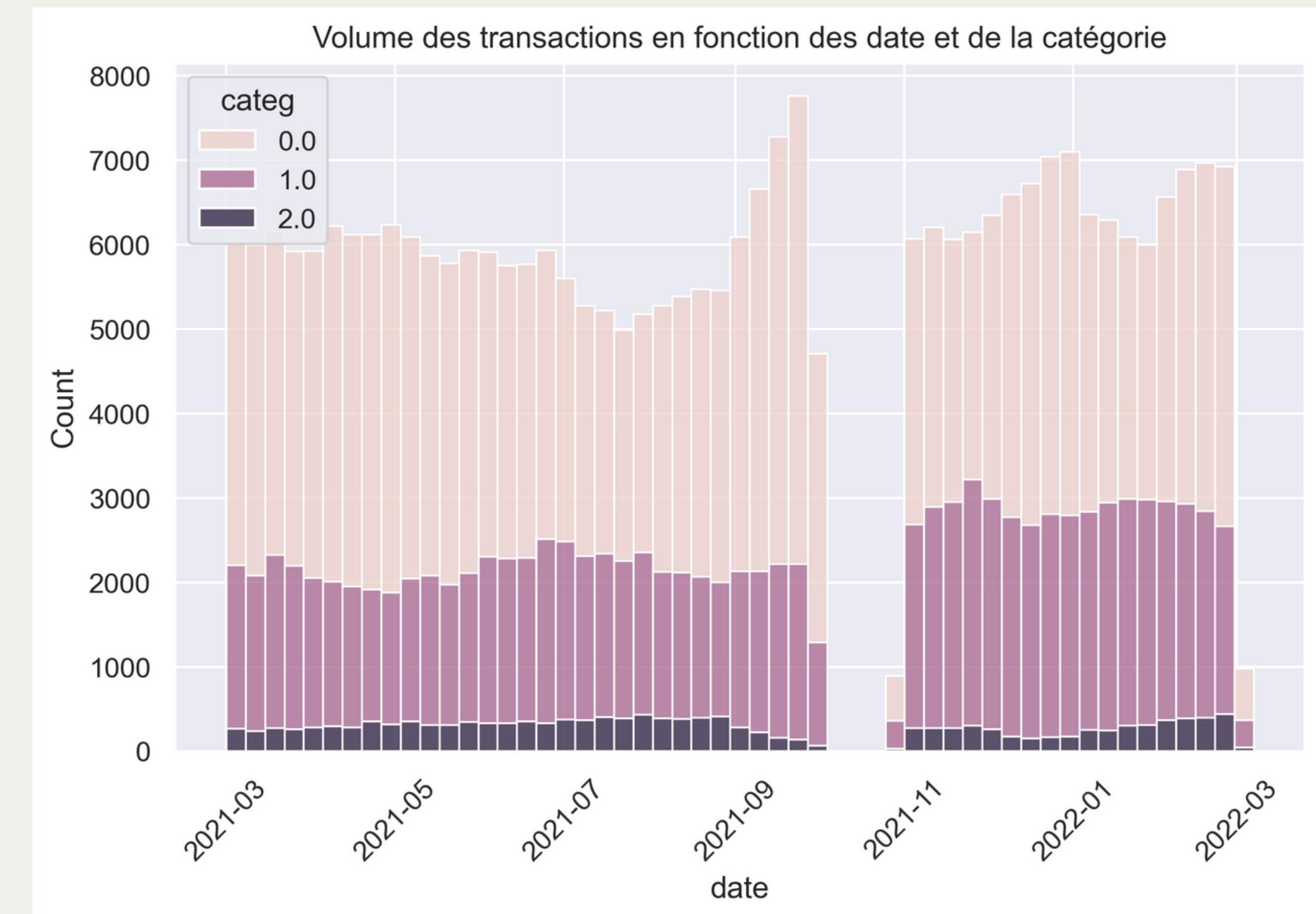
Distribution de la taille du panier moyen, en fonction de l'âge



Client particulier



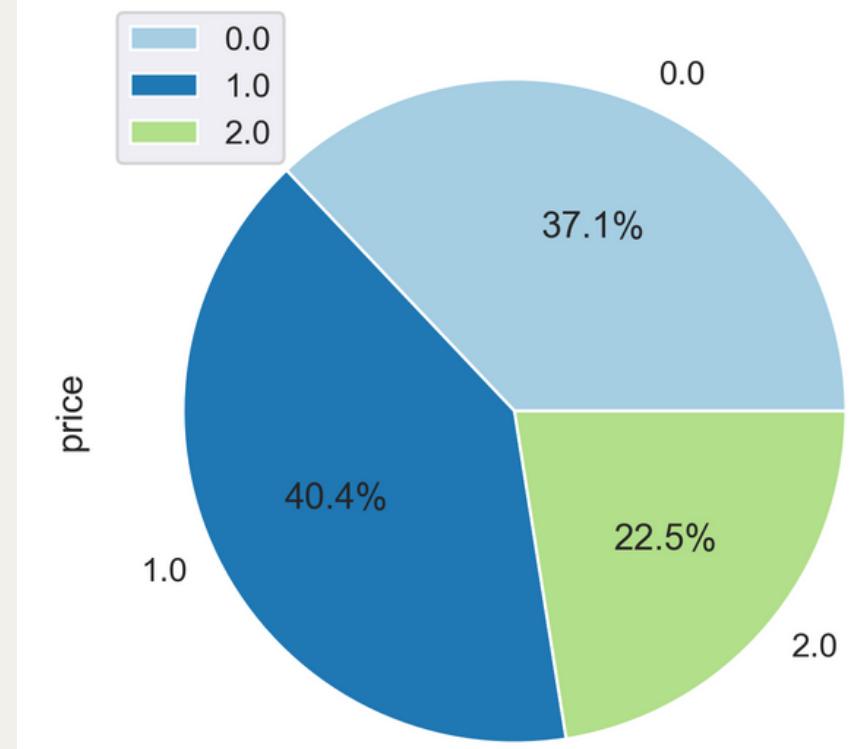
CLIENT PARTICULIER



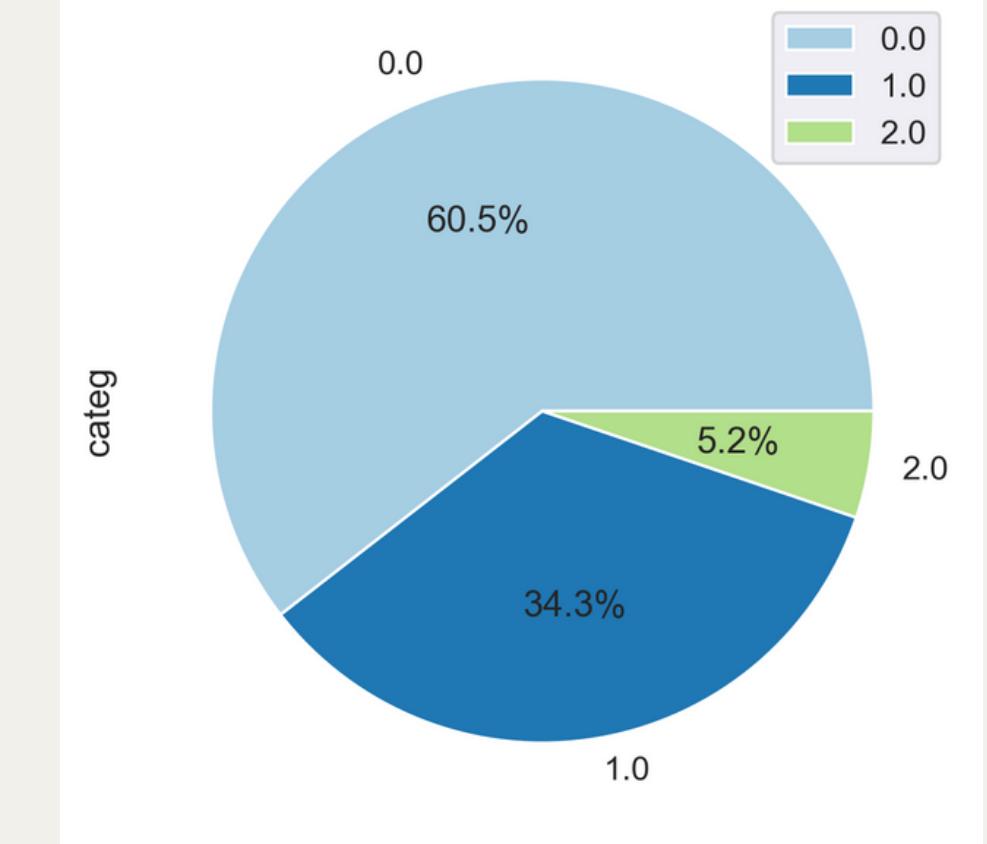
CLIENT PARTICULIER



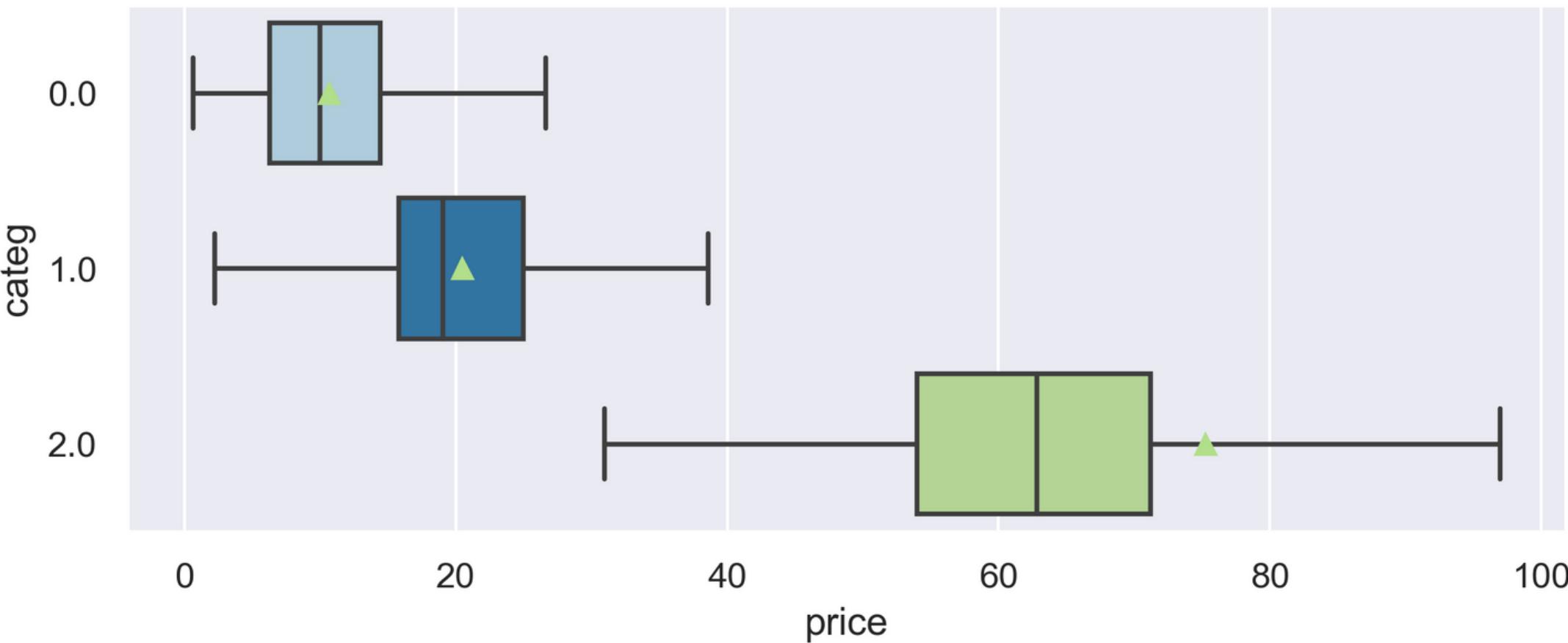
Répartition du chiffres d'affaires en fonction de la catégorie



Volume des transactions en fonction de la catégorie



Distribution des prix en fonction de la catégorie



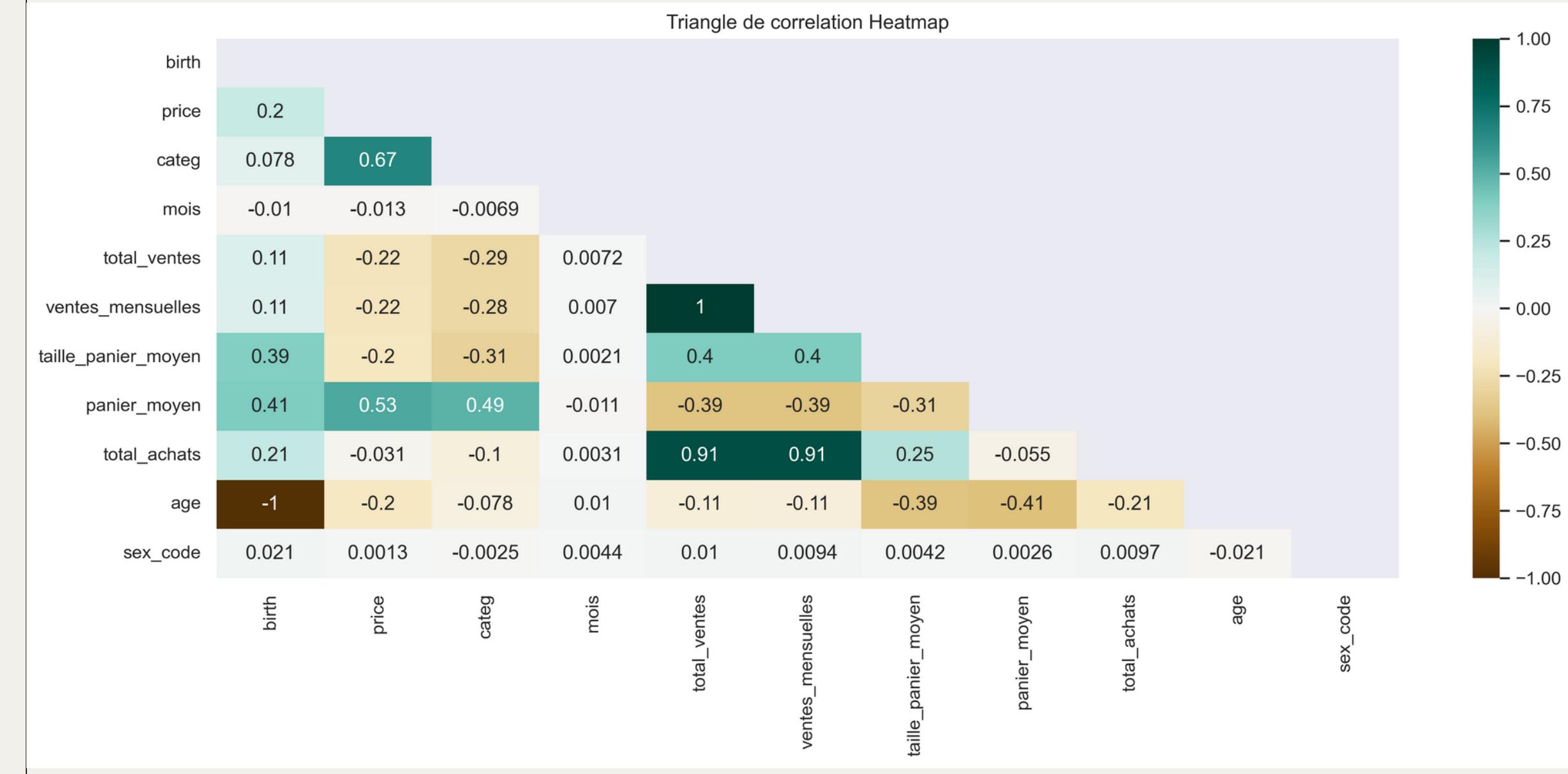
CLIENT PARTICULIER



TEST STATISTIQUE

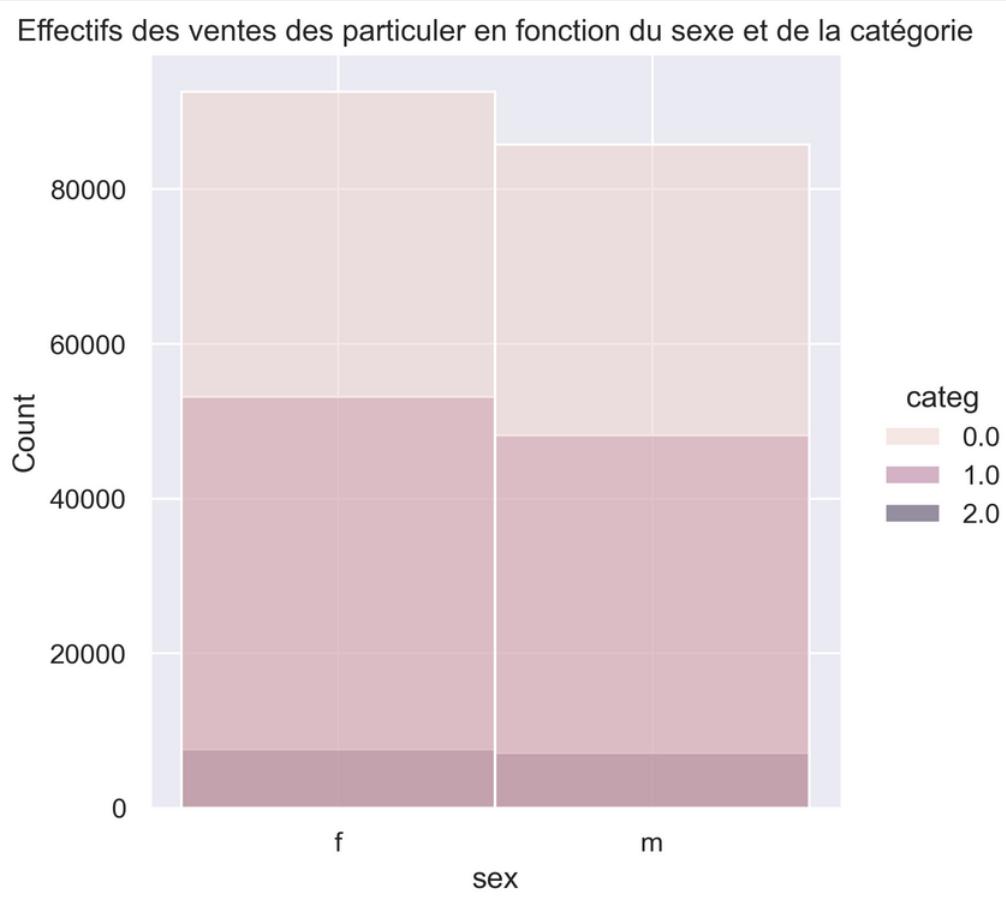
3

TEST DE CORRÉLATION



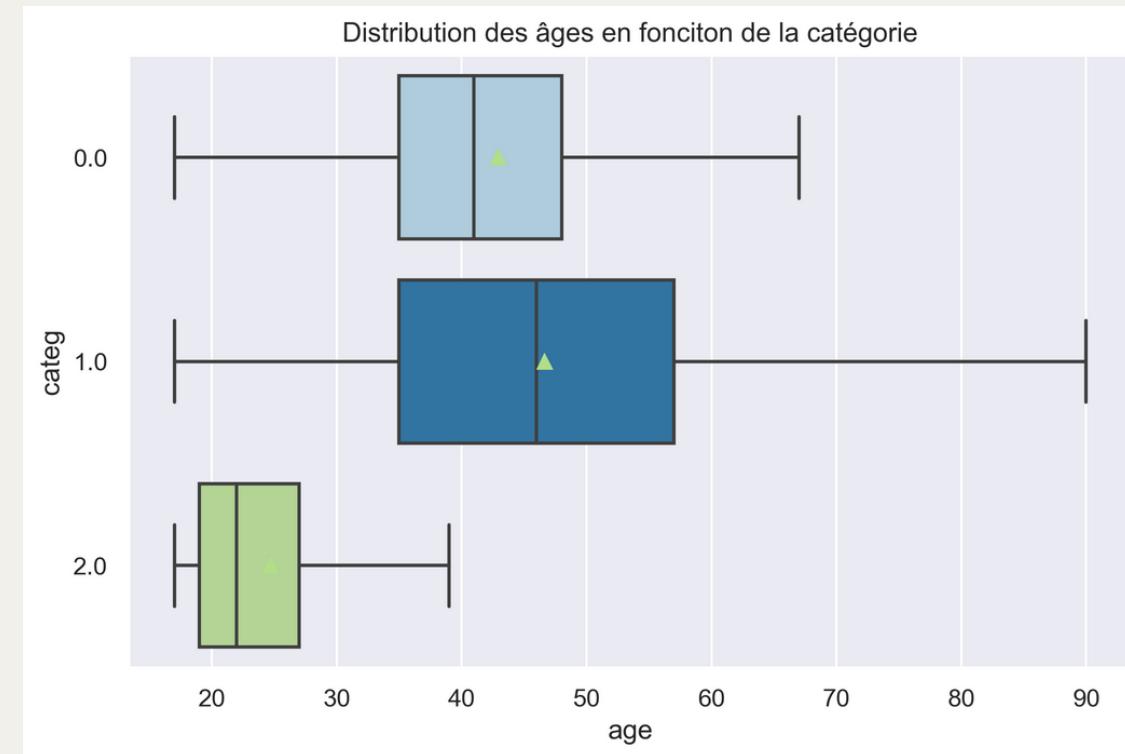


HYPOTHÈSE NULLE / ALTERNATIVES



H0 : le sexe n'a pas d'impact sur la catégorie

H1 : le sexe a un impact sur la catégorie



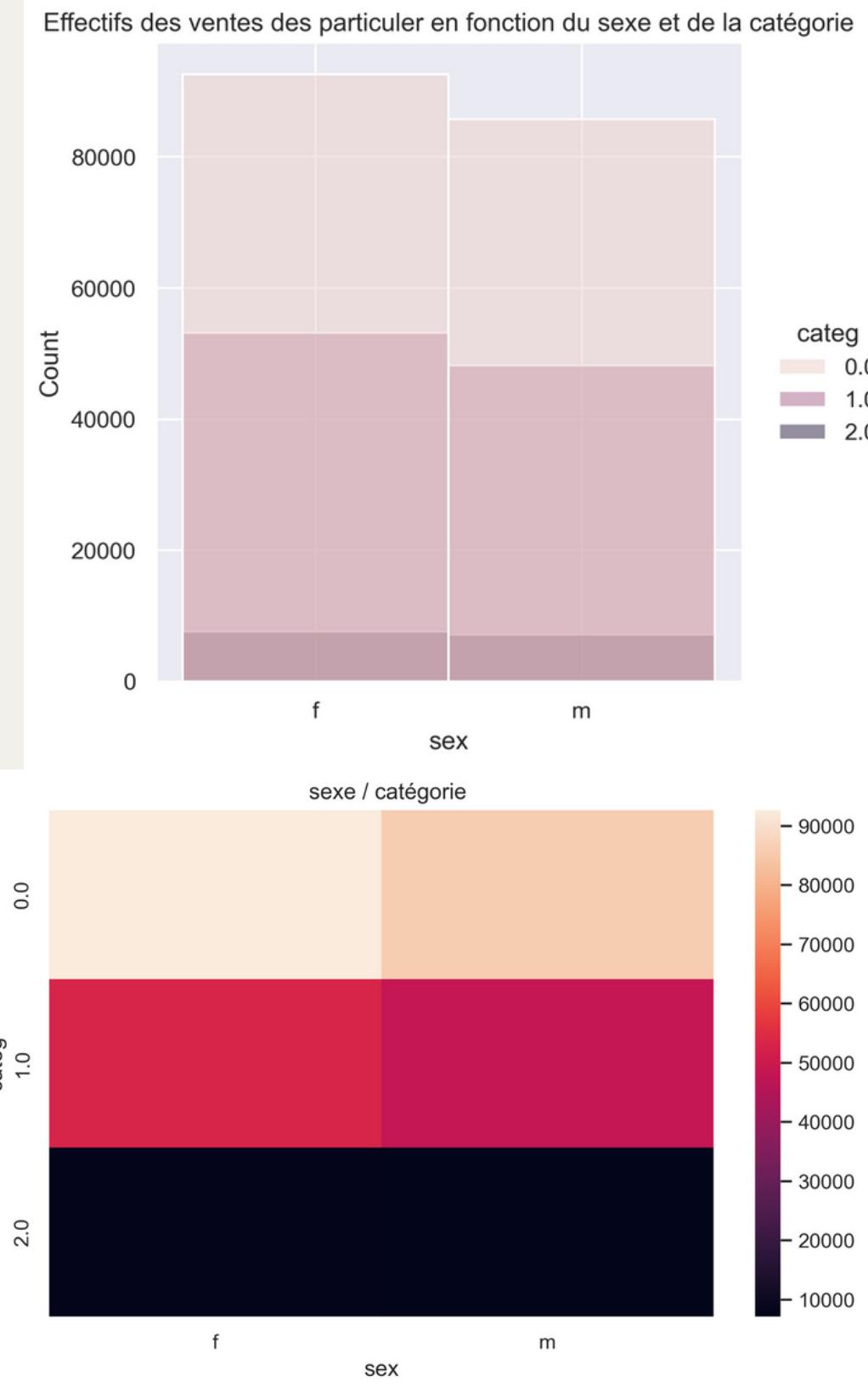
H0 : l'âge n'influe pas sur la catégorie

H1 : l'âge influe sur la catégorie

CHI-2



Variable tester :
sexe et catégorie



Test d'indépendance entre 2 variables catégoriques

2 degrés de liberté :
sex : -1
categ : -1

Conditions :

- pas besoin de normalité
- au moins 1 valeur dans chaque cellule de la table de contingence
- au moins 80% des valeur égales ou supérieure à 5

Résultats
p-value : 0.2
 H_0 acceptée



ANOVA

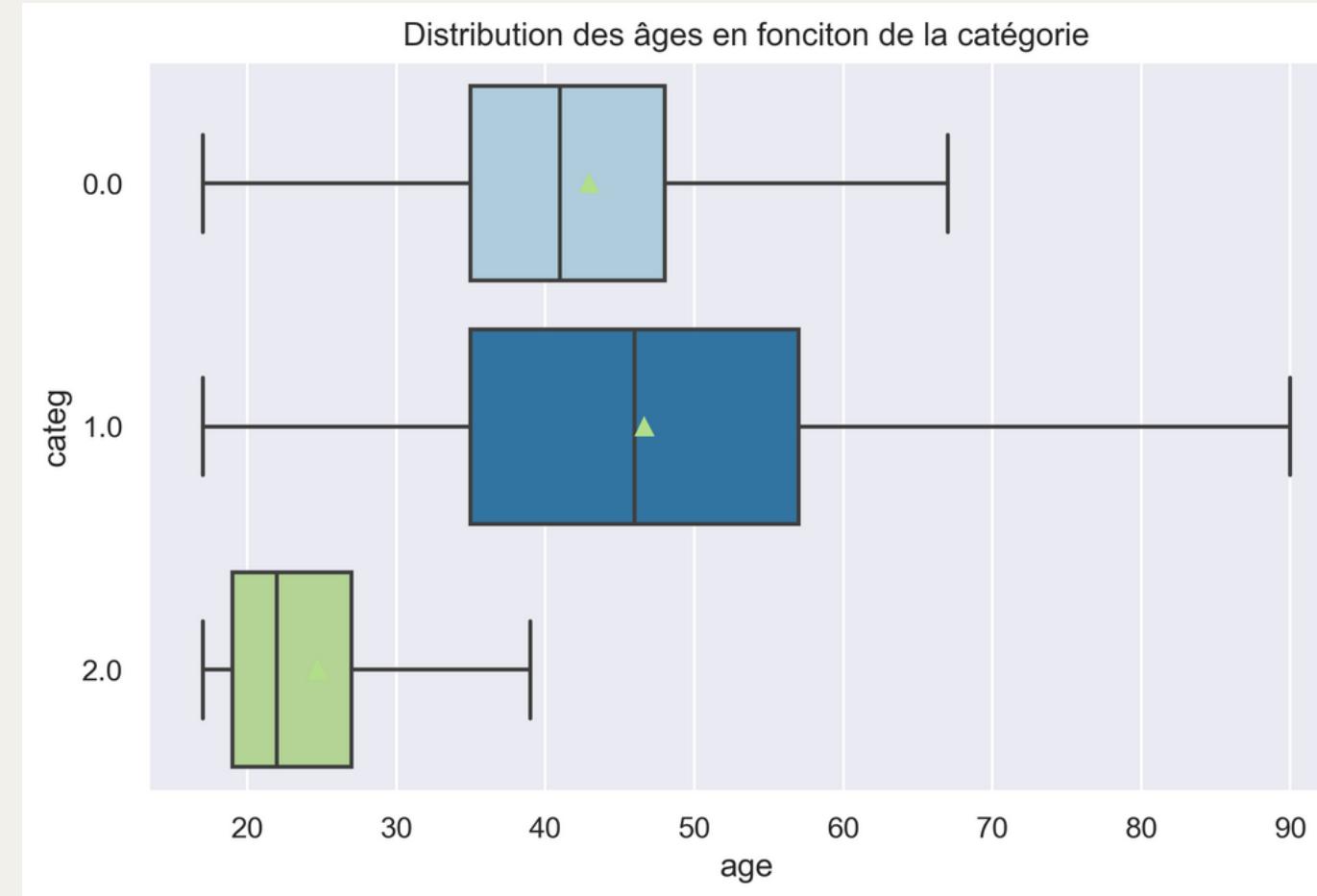
Variable tester :

Age et catégorie

Teste d'anova

p-value : 0.0

H_0 rejettee
probable corrélation



Analyse de la variance entre group:

- variable qualitative: categ
- variable quantitative: age

hypothèse nulle :

les moyennes des groupes sont égales
probablement pas de corrélation des variables

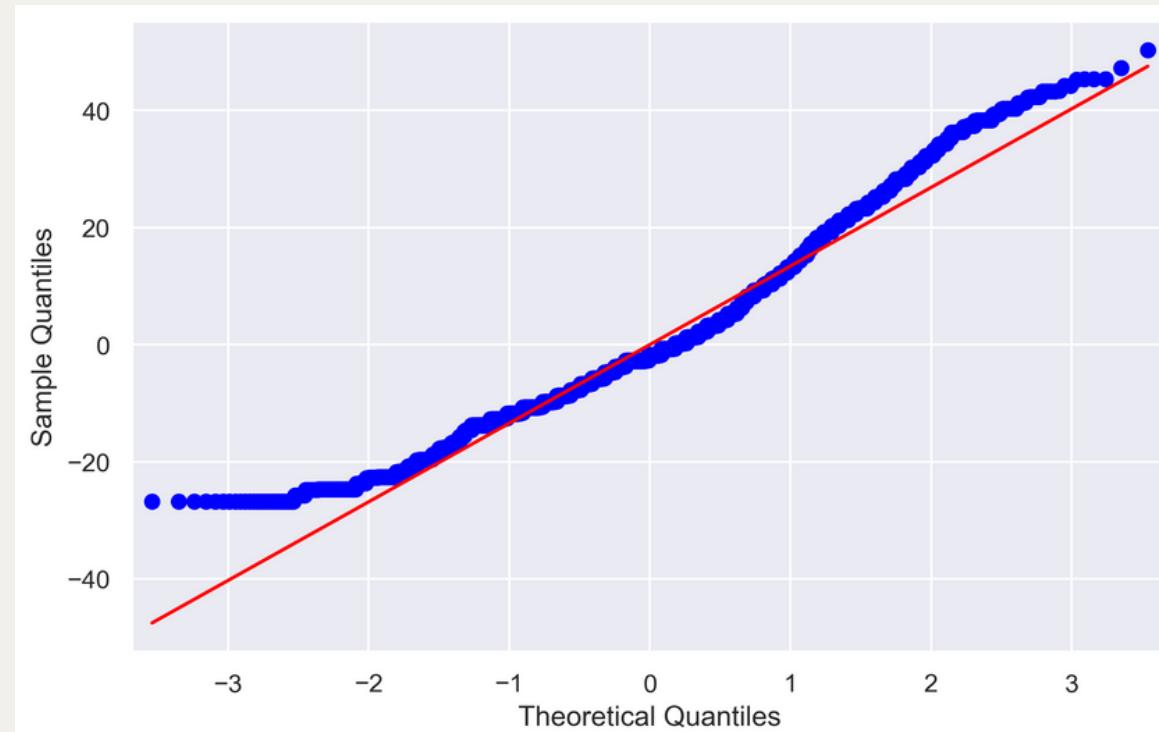
Hypothèse alternative :

les moyennes des groupes sont différentes
probablement de corrélation des variables

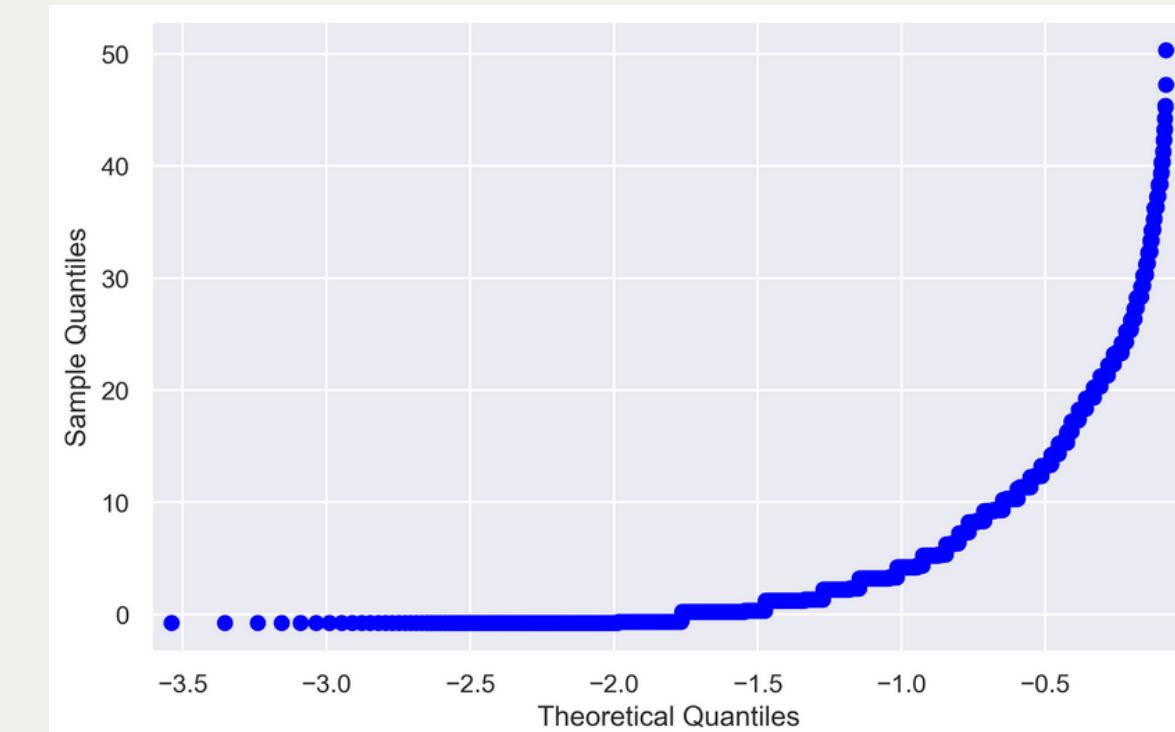
ANOVA CONDITIONS



Test de la normalité de la distribution Test de shapiro



SANS BOXCOX
P-VALUE: 0.0
REJET DE L'HYPOTHÈSE 0
LA DISTRIBUTION N'EST PROBABLEMENT PAS NORMALE



APRÈS BOXCOX
P-VALUE: 1.0
L'HYPOTHÈSE 0 EST ACCEPTÉ
LA DISTRIBUTION EST PROBABLEMENT NORMALE



ANOVA CONDITIONS

Homogénéité des variances

Test de Levene

p-value: 0.0

h1: les variances ne
sont pas égales

Test alternatif : Welch ANOVA

p-value: 0.0

h1: les moyenne sont
différentes

CONCLUSION

- **3 catégories de livre ordonnées par prix**
- **3 groupes de clients :**
 - **Étudiants de moins de 30 ans**
 - **Les grands lecteurs de 30 à 50 ans**
 - **Tous les autres profils**
- **Les corrélations :**
 - **Le sexe n'est pas corrélé à la catégorie**
 - **l'age est corrélé au chiffre d'affaires**



RÉPONSE À VOS QUESTION

