

DATA SCIENTIST

PROJET 5 :
Segmentez des clients
d'un site e-commerce



Nom	customers	Geolocalisation	Items	Payment	Reviews	Orders	Products	Sellers	Prod_categorie
Lignes	99441	1000163	112650	112650	100000	99441	32951	3095	71
Colonnes	5	5	7	7	7	8	9	4	2

9 dataframes

Order_id est la clé de 3

Sommaire

Partie 1 : nettoyage et analyse

I, création du jeu de données à partir de la base client

- a. Jonction des jeux de données
- b. Nettoyage des données
- c. Sélection d'une période cohérente

II, Transformation des données avec client en clé

III, Analyses

- a. Première analyse
- b. Détection d'outliers avec Isolation Forest
- c. Sélection des features pertinentes

Partie 2 : modélisation

I, Feature engineering

- II, PCA
- a. Kmeans
- b. DBSCAN

III, TSNE

- a. Kmeans
- b. DBSCAN

IV, Tests de stabilité

- a. Stabilité à l'initialisation
- b. Stabilité temporelle

V, Analyse des groupes

PARTIE I

112,650 items commandés au total sous 98666 commandes.

On commence par fusionner ces items avec la table orders afin de récupérer les informations concernant les clients et les modalités de livraisons.
(jonction gauche sur order_id).

Ensuite jonction du dataframe nouvellement créé avec les jeux de données :

- product (jointure gauche avec product_id)
- reviews (jointure gauche avec order_id)
- Customers (jointure gauche avec customer_id)
- Payment (jointure gauche avec order_id)

Sélection des colonnes

```
1 df.isna().sum()
```

order_id	0
order_item_id	0
product_id	0
seller_id	0
price	0
freight_value	0
customer_id	0
order status	0
order purchase timestamp	0
order delivered customer date	2454
product category name	1603
review_id	0
review score	0
review_comment_title	99045
review_comment_message	64244
review_creation_date	0
customer unique id	0
customer_zip_code_prefix	0
customer_city	0
customer state	0
payment type	3
dtype: int64	

```
1 data.order_status.value_counts()
```

delivered	110197
shipped	1185
canceled	542
invoiced	359
processing	357
unavailable	7
approved	3

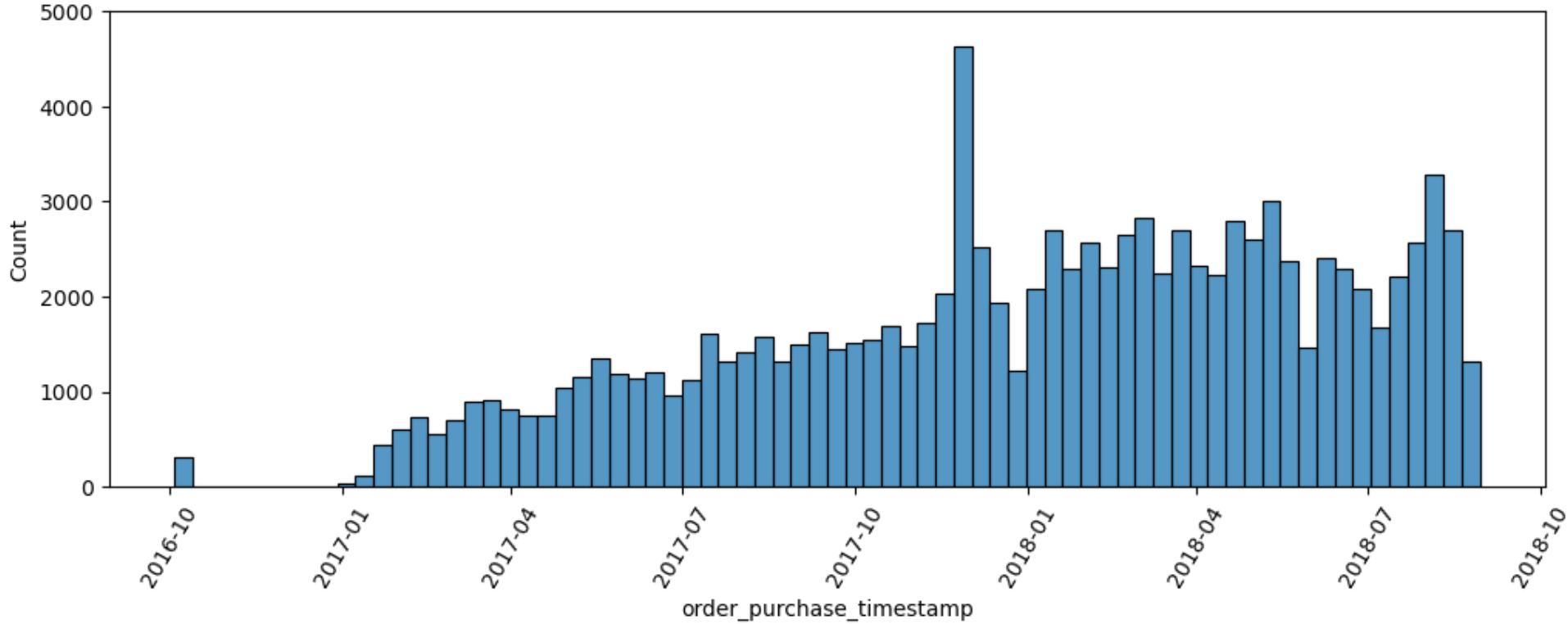
Name: order_status, dtype: int64

Conservation des seules commandes délivrées

Effacement des lignes ayant des données manquantes

```
1 data.isna().sum()
```

price	0
order_purchase_timestamp	0
order_delivered_customer_date	8
product_category_name	1537
review_score	0
customer_unique_id	0
customer_state	0
payment_type	0
dtype: int64	



Les données représentant l'année 2016 sont marginales.

Dans un souci de cohérence, nous allons travailler sur l'année 2017 et vérifierons notre travail avec les données de l'année 2018.

Méthode des pivot_table avec customer_unique_id comme index afin d'avoir des informations propres à chaque clients.

Création de nouvelles colonnes :

- temporelles
 - date_first_purchase
 - date_last_purchase
 - delivery_time (en jours)
- monétaires
 - min_purchase_amount
 - max_purchase_amount
 - total_purchase_amount
- avis
 - count_review_score
 - mean_review_score
- catégories
 - item_most_purchase

1	df.isna().sum()
	customer_unique_id 0
	date_first_purchase 0
	date_last_purchase 0
	delivery_time 2
	min_purchase_amount 0
	max_purchase_amount 0
	total_purchase_amount 0
	count_review_score 0
	mean_review_score 0
	total_purchase_count 0
	item_most_purchase 0
	customer_state 0
	payment_type 0
	dtype: int64

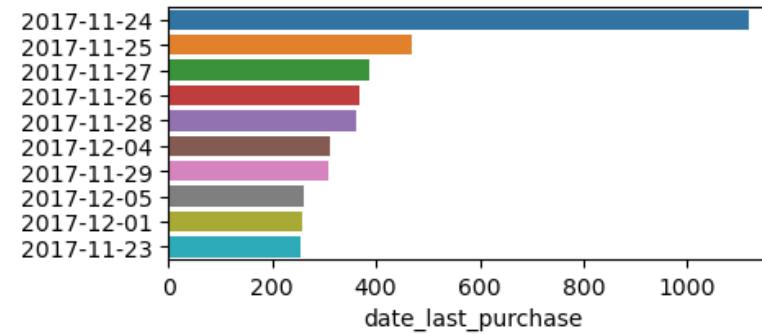
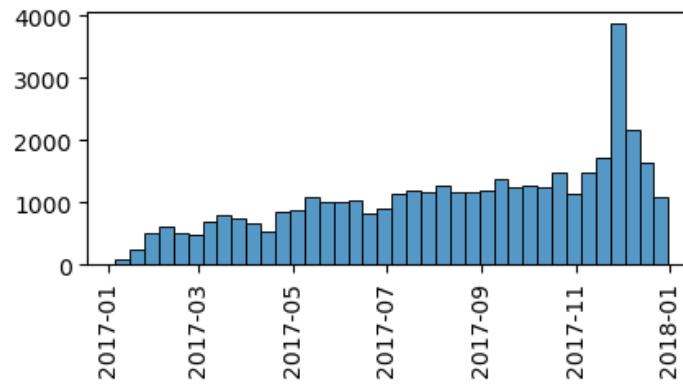
Remplacement des valeurs nulles de delivery_time par la moyenne.

A, PREMIÈRES ANALYSES

III, ANALYSES

PARTIE 1

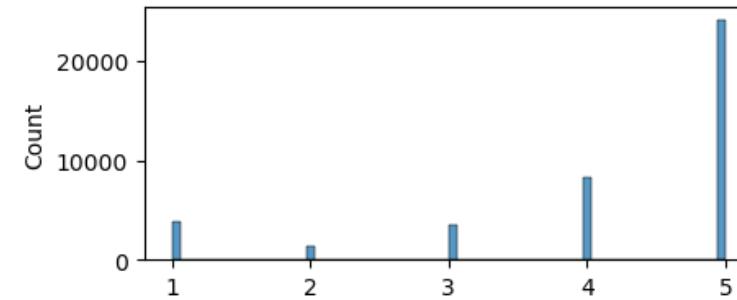
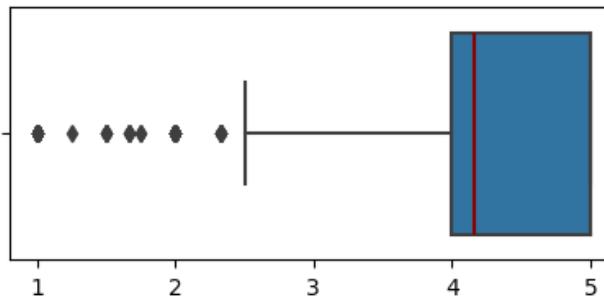
Analyse univariée de date_last_purchase



400
articles vendus
par **jour** pour les
bonne journées

24 novembre
meilleur vente
de l'année
Black Friday
x2,5

Analyse univariée de mean_review_score



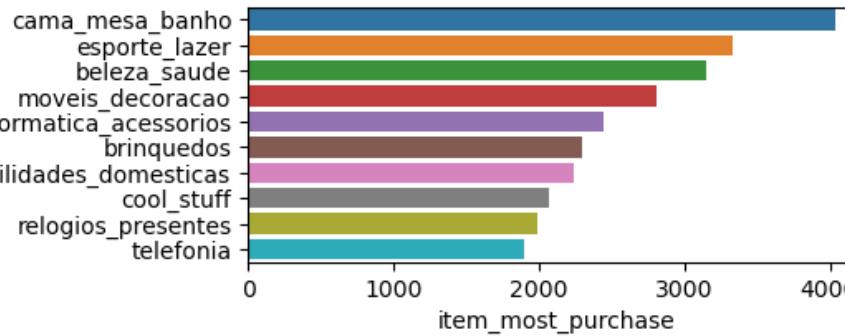
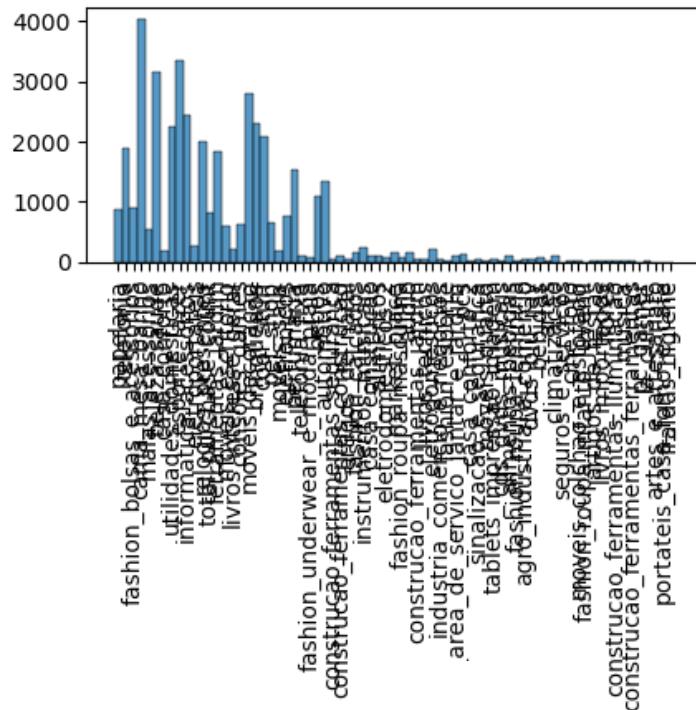
4+ note
moyenne des
utilisateurs

A, PREMIÈRES ANALYSES

III, ANALYSES

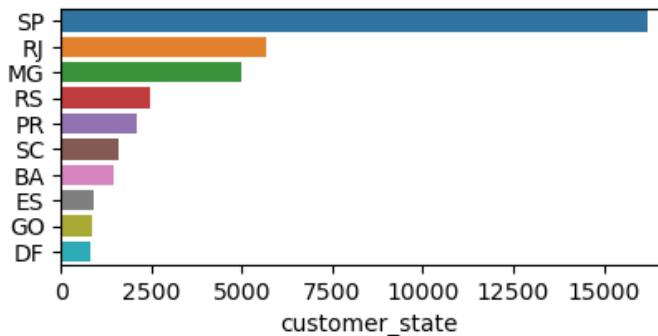
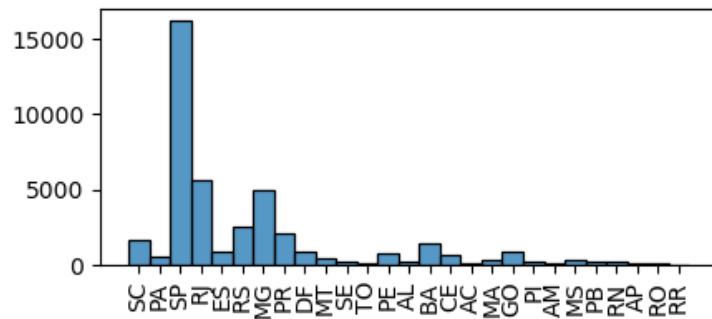
PARTIE 1

Analyse univariée de item_most_purchase



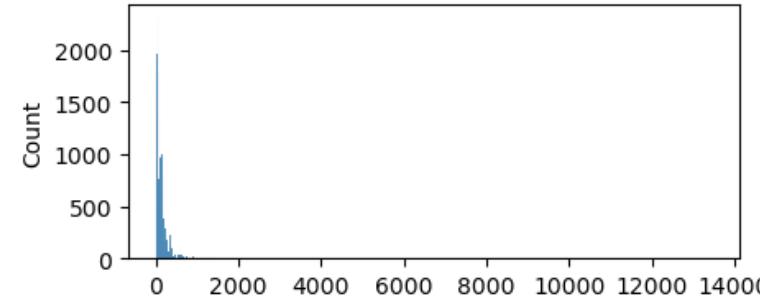
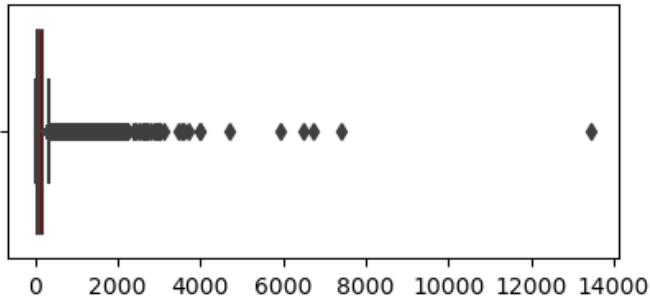
1^{er} catégorie :
linge de maison
10%

Analyse univariée de customer_state

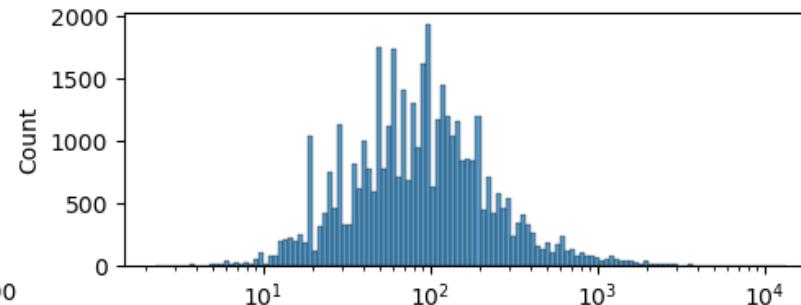
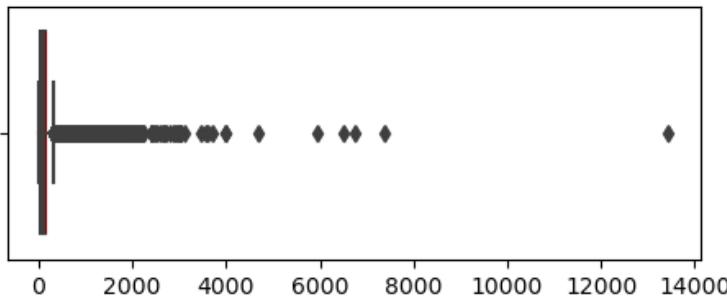


64% des ventes sont faites par l'Etat de Sao Paulo

Analyse univariée de total_purchase_amount



Analyse univariée de total_purchase_amount



141,57 \$

montant
dépensé en
moyenne par
client sur l'année

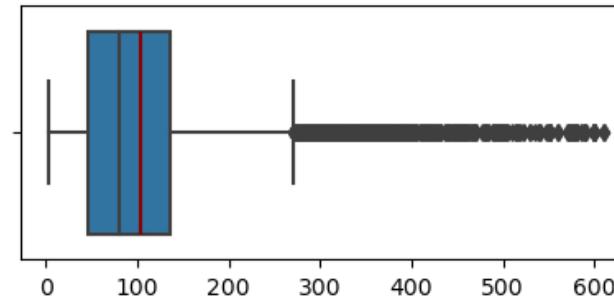
222,60 \$

déviation standard
observée.
Données disparates

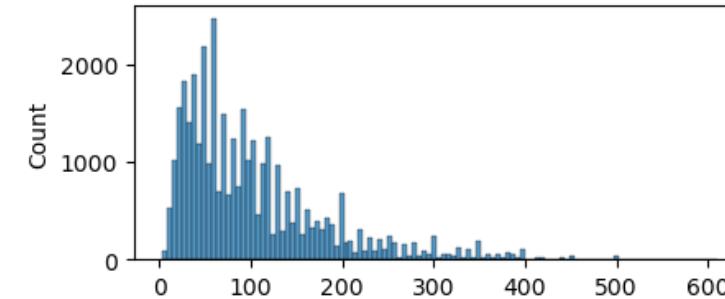
```
1 df.total_purchase_amount.describe()
```

	count	mean	std	min	25%	50%	75%	max
	41354.000000	141.579976	222.606338	2.290000	48.000000	89.000000	155.000000	13440.000000
Name:	total_purchase_amount	, dtype:	float64					

```
1 df_if2.iforest.value_counts()  
1 37218  
-1 4136  
Name: iforest, dtype: int64
```



```
1 df_if2.total_purchase_amount.describe()  
count    37218.000000  
mean     103.359559  
std      84.441551  
min      2.290000  
25%     44.990000  
50%     79.900000  
75%     135.000000  
max     610.000000  
Name: total_purchase_amount, dtype: float64
```



Isolation Forest propose d'écartier 10% de données considérées comme extrêmes.

Sans surprise, les clients ayant dépensés les plus fortes sommes d'argent + de 610\$ annuel sont écartés).

Ils représentent 2% de la clientèle et sont trop éloignés des autres clients pour établir un model pertinent.

Leur étude devrait se faire séparément.

Si on applique une segmentation **RFM stricte**, on gardera les colonnes :

- date last purchase (recency)
- total purchase count (frequency)
- total purchase amount (monetary)

Cependant, pour ne pas perdre trop d'informations, nous allons garder d'autres features tels que :

- delivery time (durée de la livraison)
- mean review score
- item most purchase
- customer state
- payment type

On finit donc avec un jeu de données de 8 colonnes et 37218 lignes.

PARTIE II

1 df.head(2)

		date_last_purchase	delivery_time	total_purchase_amount	mean_review_score	total_purchase_count	item_most_purchase	customer_state	payment_type
d									
4		2017-03-10 21:05:03	25.0	69.00	3.0	1	papelaria	SC	credit_card
8		2017-10-12 20:29:41	20.0	25.99	4.0	1	telefonia	PA	credit_card

Label encoder
transformation des chaînes de caractère en nombre

Standard Scaler
normalisation par $z = (x - u) / s$

1 dfx.head(2)

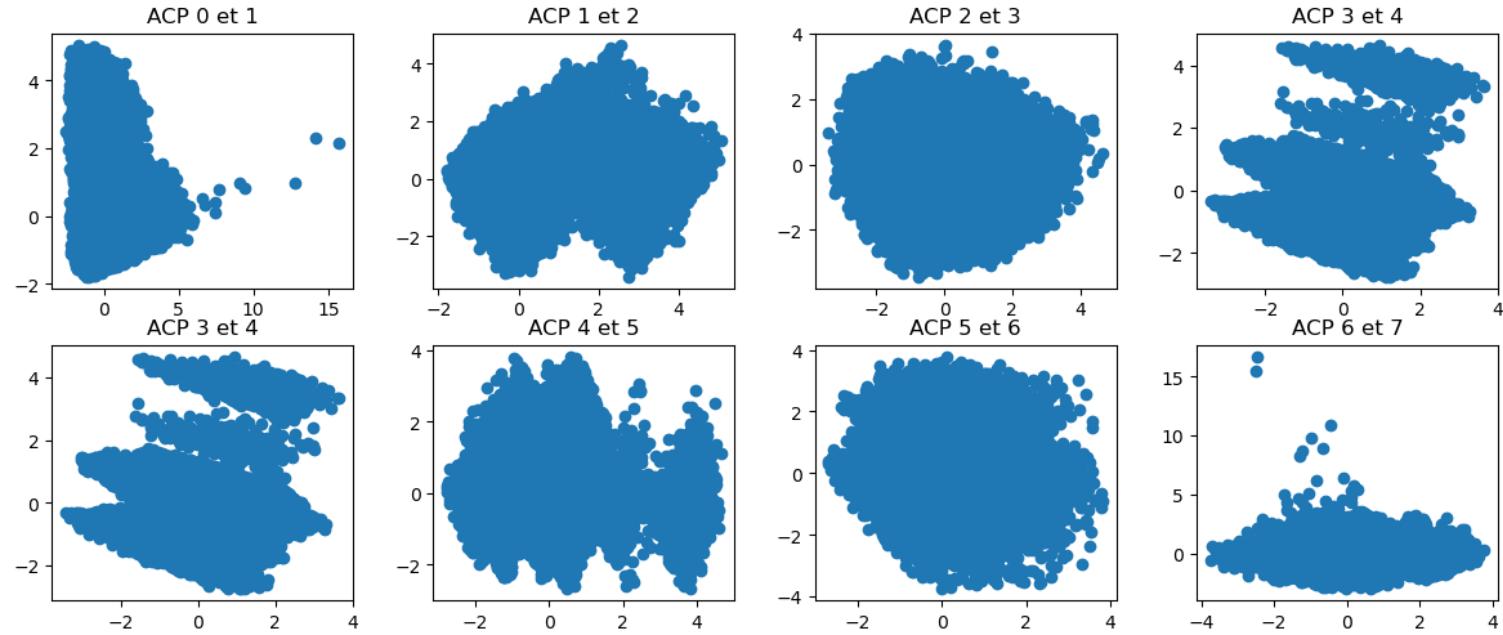
		date_last_purchase	delivery_time	total_purchase_amount	mean_review_score	total_purchase_count	item_most_purchase	customer_state	payment_type
d									
4		-1.673282	1.750170	-0.406909	-1.085904	-0.291456	1.039466	0.627001	0.259844
8		0.604093	1.084675	-0.916262	-0.224568	-0.291456	1.457271	-0.806548	0.259844

Afin de pouvoir effectuer une partition pertinente de nos données, nous allons opérer une réduction de dimension afin d'avoir une représentation plus intelligible des datas tout en essayant de garder un maximum d'informations.

Nous allons essayer deux méthodes :

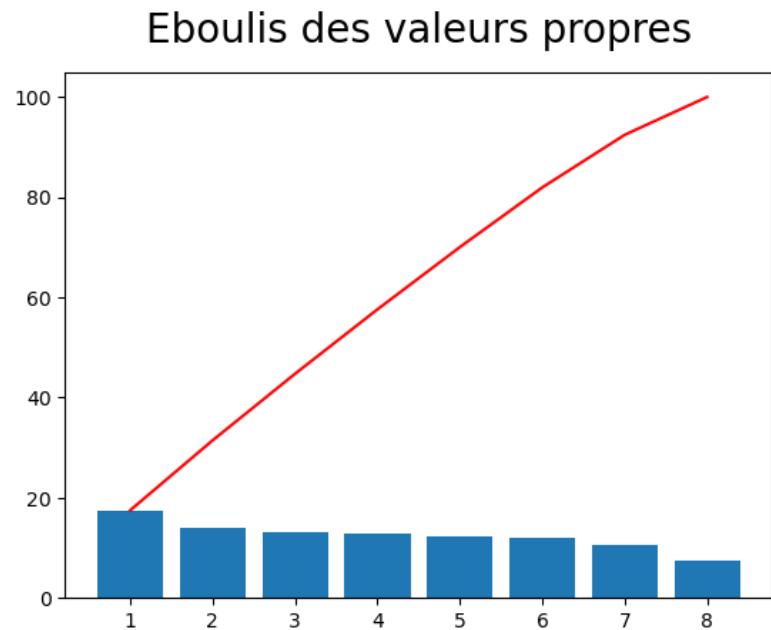
- une linéaire : l'analyse principale des composantes PCA
- une non linéaire : T-distributed Stochastic Neighbor Embedding ou T-SNE.

L'ANALYSE DES COMPOSANTES PRINCIPALES



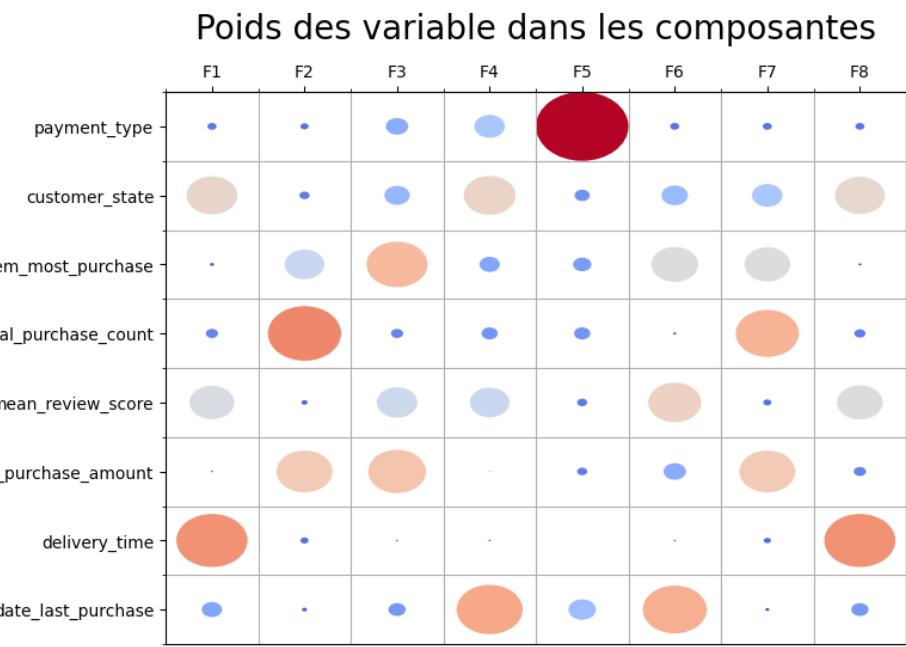
8 dimensions
nécessaires pour
expliquer 100 % de
la variance

Pas de 'coude'
dans l'éboulis des
valeurs propres.
Pas de grandes
différences
d'importance entre
les composantes.

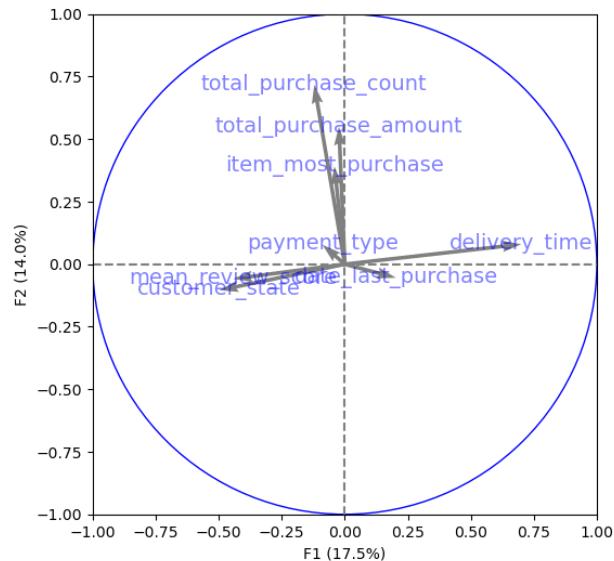


Il n'y a **pas de corrélation claire** entre une composante et une colonne (a part payment type et F5).

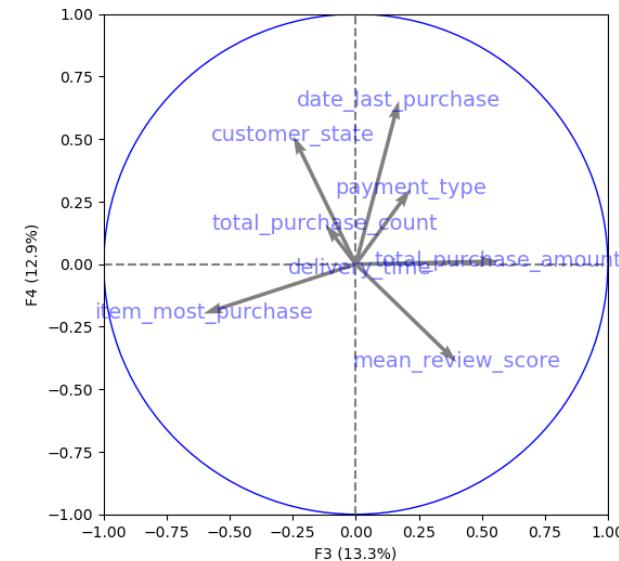
F1 explique 17,5% de la variance, F8 7,6%



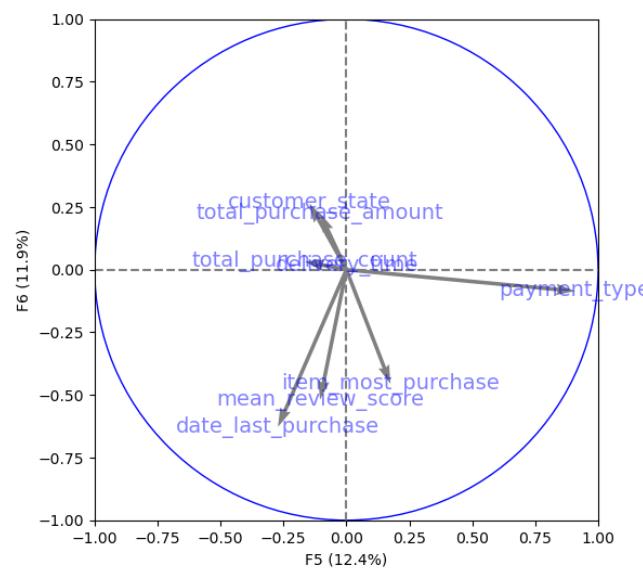
Cercle des corrélations (F1 et F2)



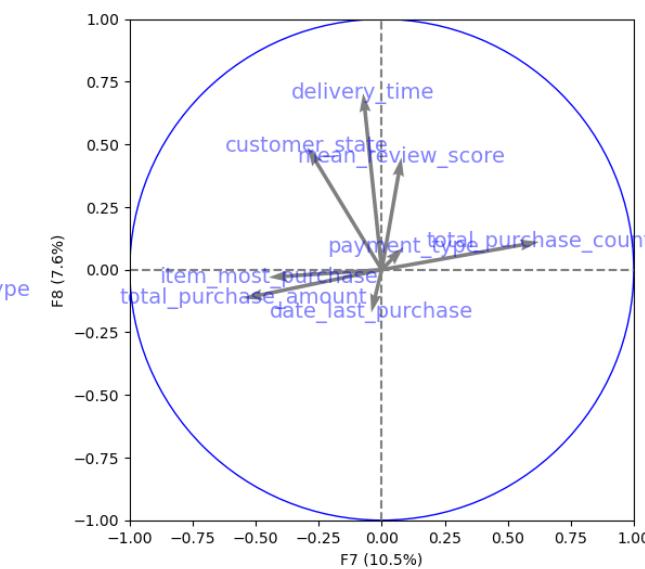
Cercle des corrélations (F3 et F4)



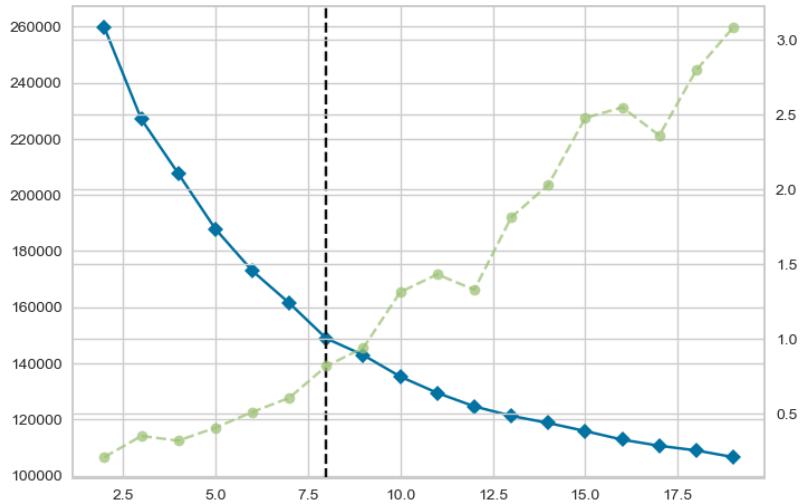
Cercle des corrélations (F5 et F6)



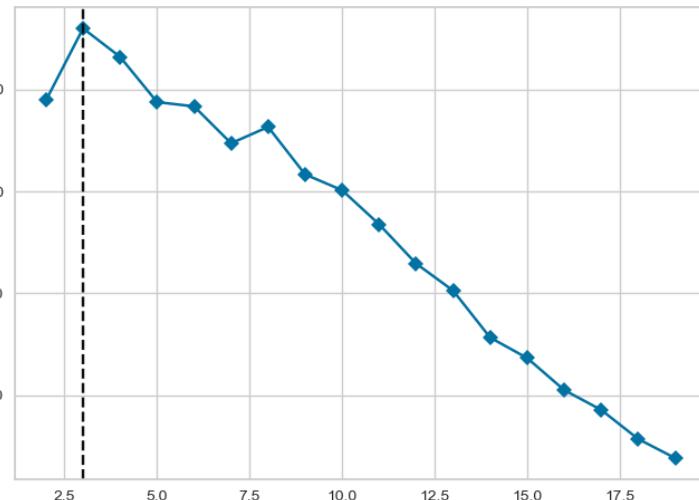
Cercle des corrélations (F7 et F8)



Détermination du nombre de groupe par distorsion et dispersion

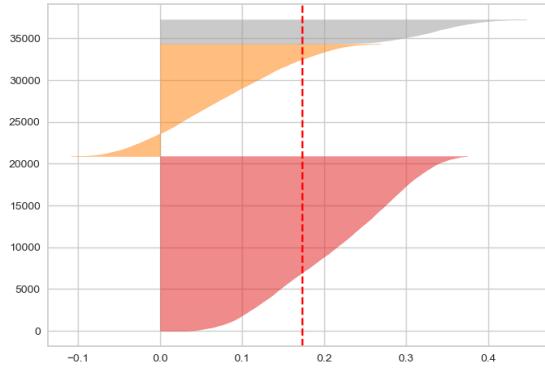
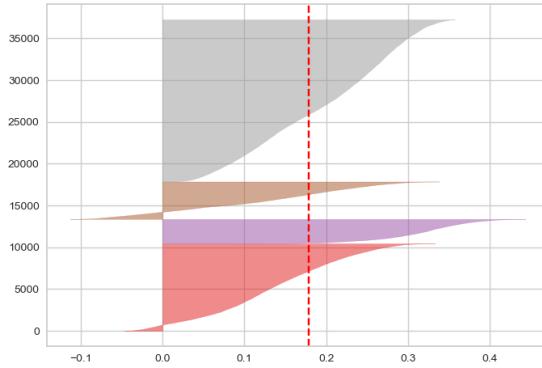
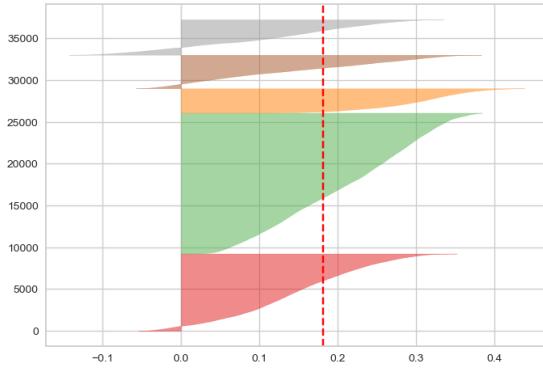
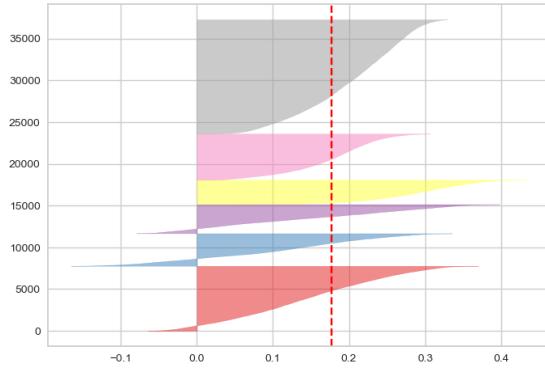
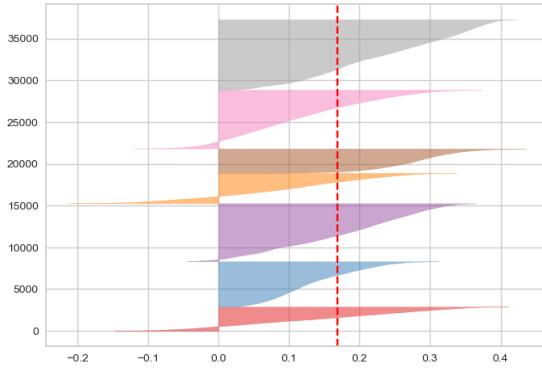
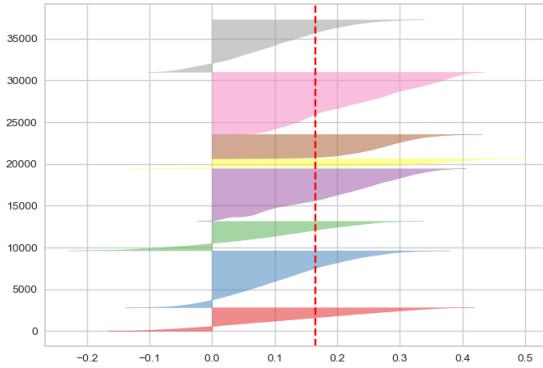


Le calcul de la distorsion (SSR entre les points et les centres) nous propose 8 groupes
Score total 148876



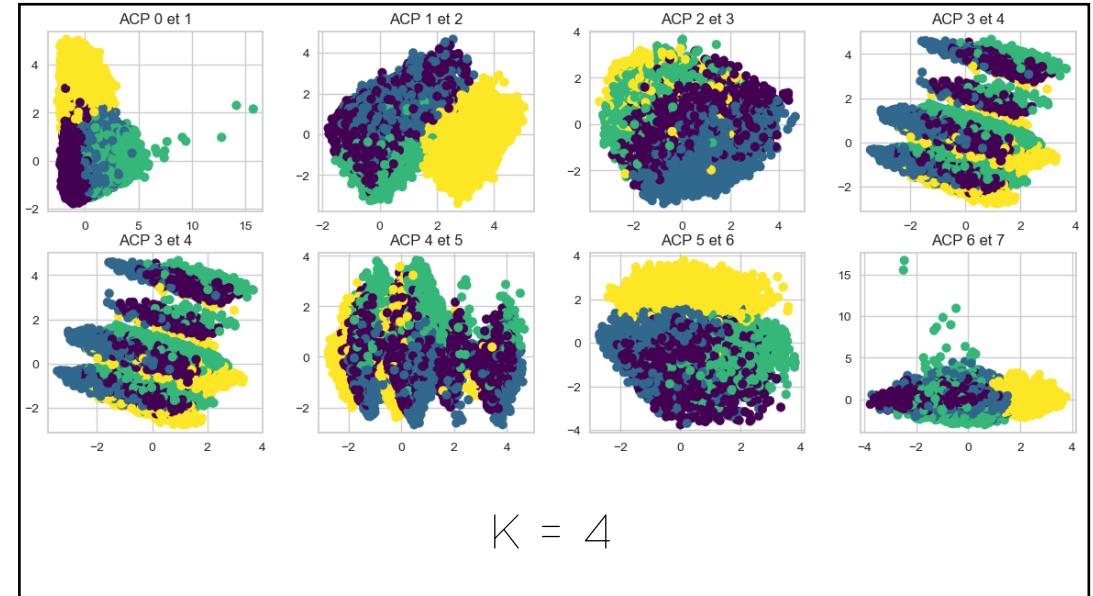
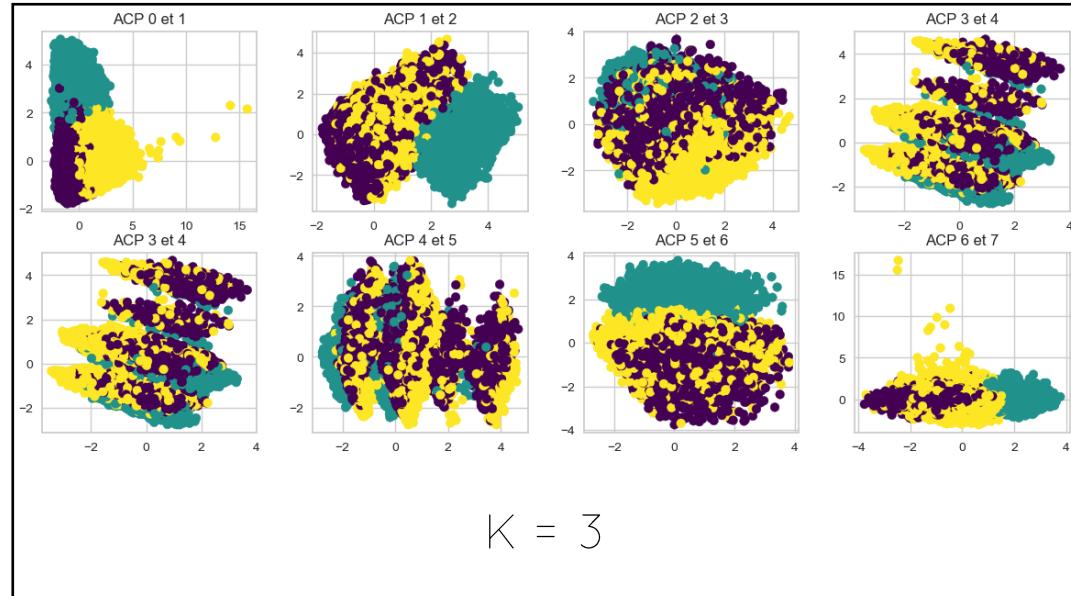
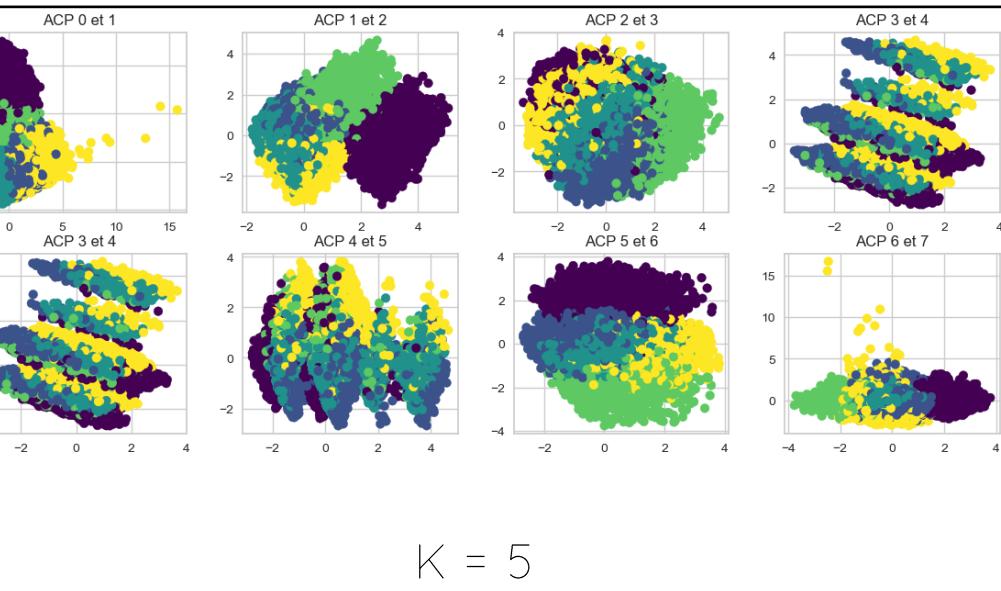
Le calcul par la méthode Calinski-harabasz, se basant sur le ratio de dispersion intra et extra groupes, nous propose une solution à 3 groupes.

Détermination du nombre de groupe par le coefficient silhouette

 $K = 3$  $K = 4$  $K = 5$  $K = 6$  $K = 7$  $K = 8$

< 0,2

Le calcul des coefficients silhouette pour différentes valeurs de k ne nous donne pas de partitionnement optimale.

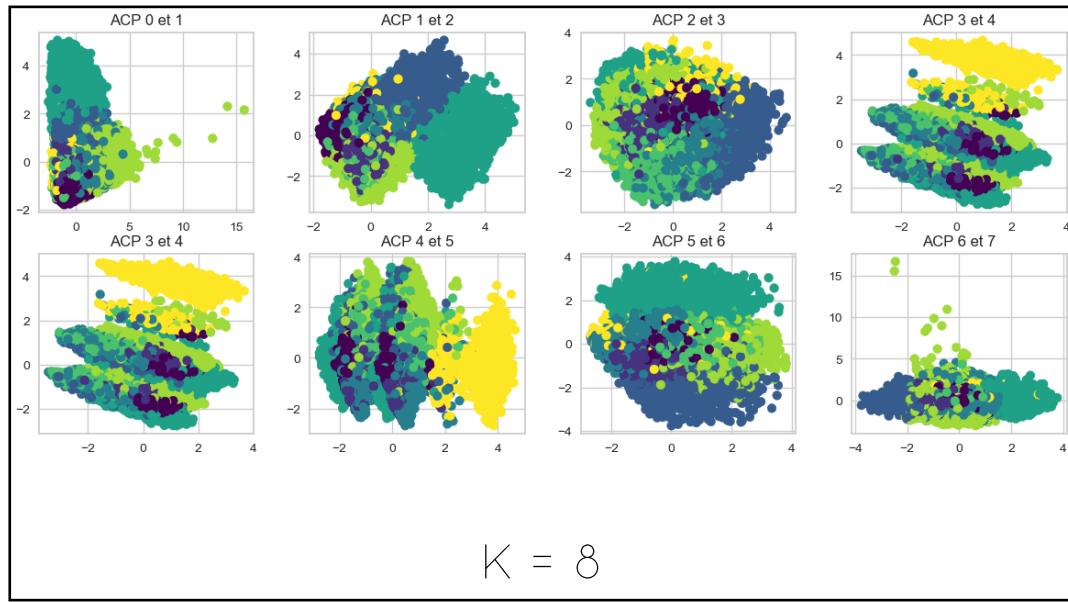


Dès la troisième composante (44% d'explication de la variance), les groupes apparaissent mélangés.

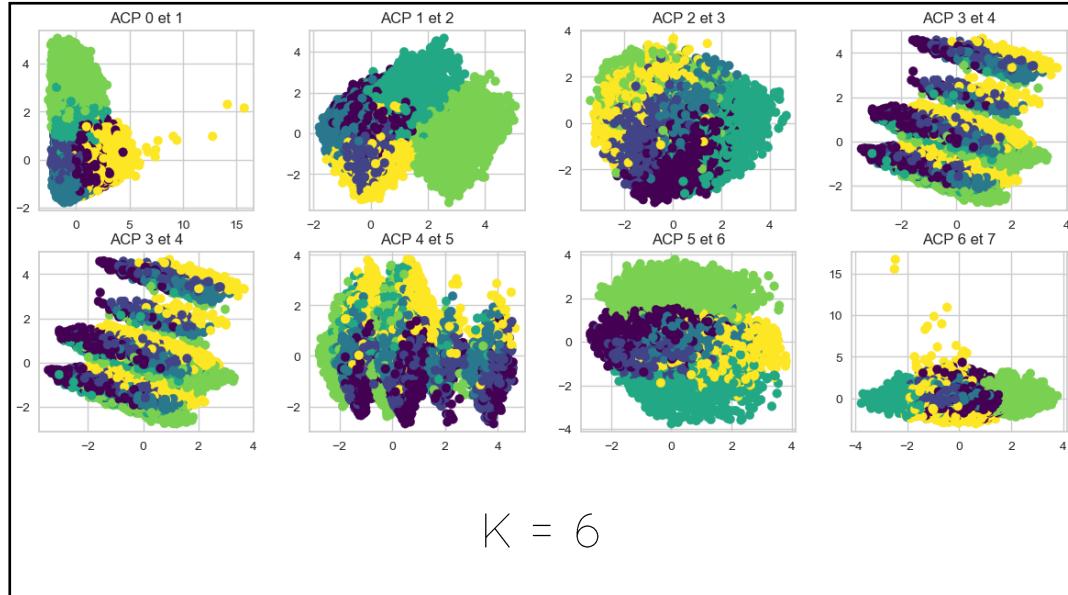
A, KMEANS

II, PRINCIPAL COMPOSANT ANALYSIS

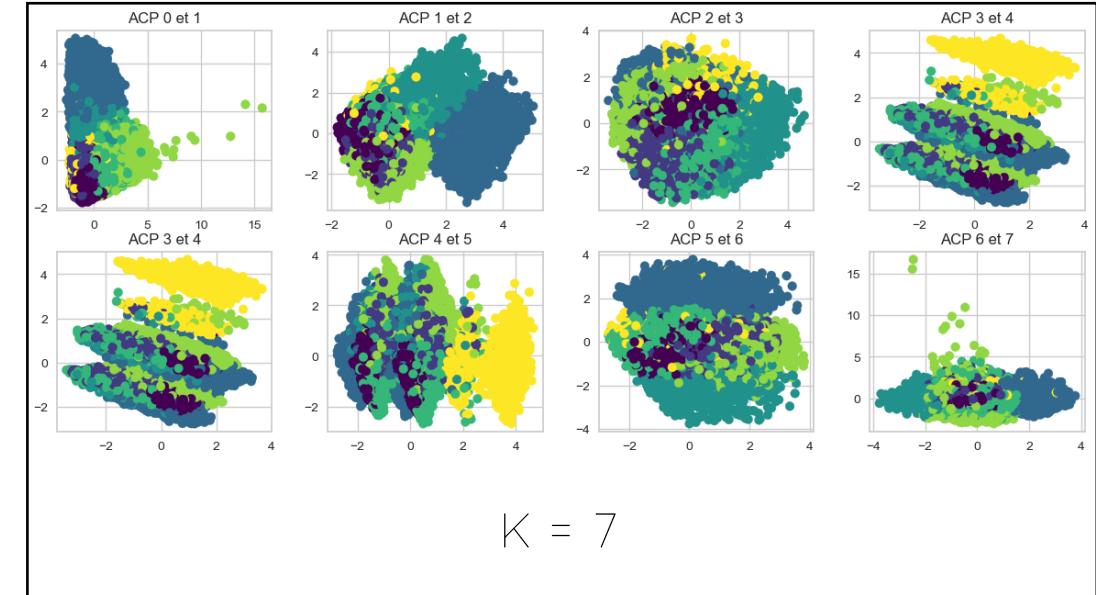
PARTIE 2



$K = 8$



$K = 6$



$K = 7$

Il semble que la combinaison PCA et KMeans ne soit pas la meilleur pour déterminer des partitions pertinentes des données.

B, DBSCAN

II, PRINCIPAL COMPOSANT ANALYSIS

PARTIE 2

	first	second	group	max_g	min_g
0	0.5	10.0	86.0	18798.0	4.0
1	0.5	50.0	6.0	31327.0	154.0
2	0.5	100.0	4.0	34332.0	652.0
3	1.0	10.0	14.0	24972.0	8.0
4	1.0	50.0	7.0	21872.0	52.0
5	1.0	100.0	4.0	19965.0	144.0
6	2.0	10.0	3.0	34291.0	14.0
7	2.0	50.0	3.0	34264.0	61.0
8	2.0	100.0	3.0	34225.0	114.0

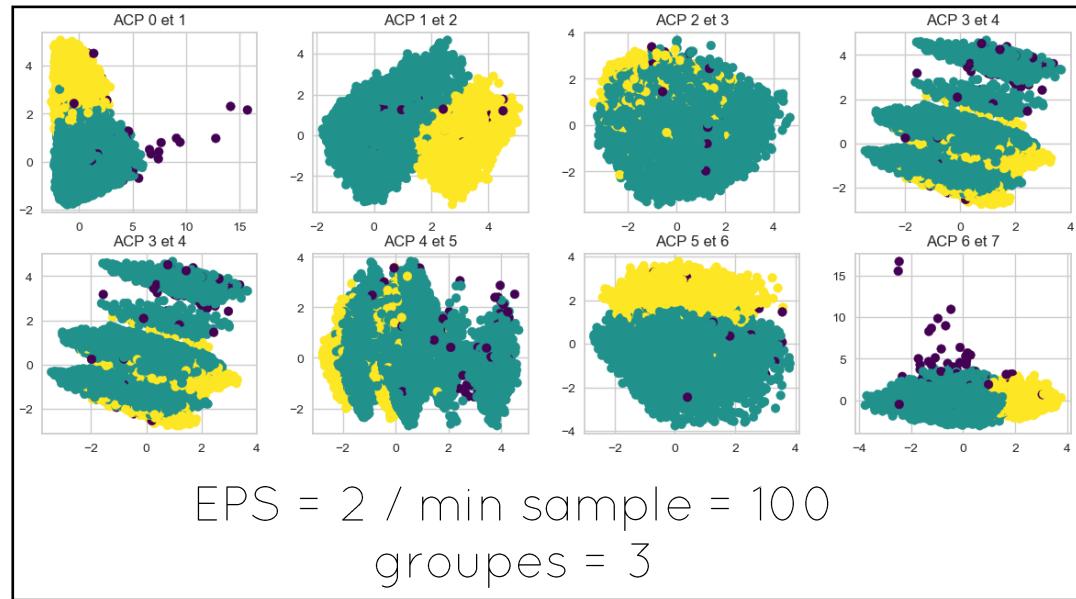
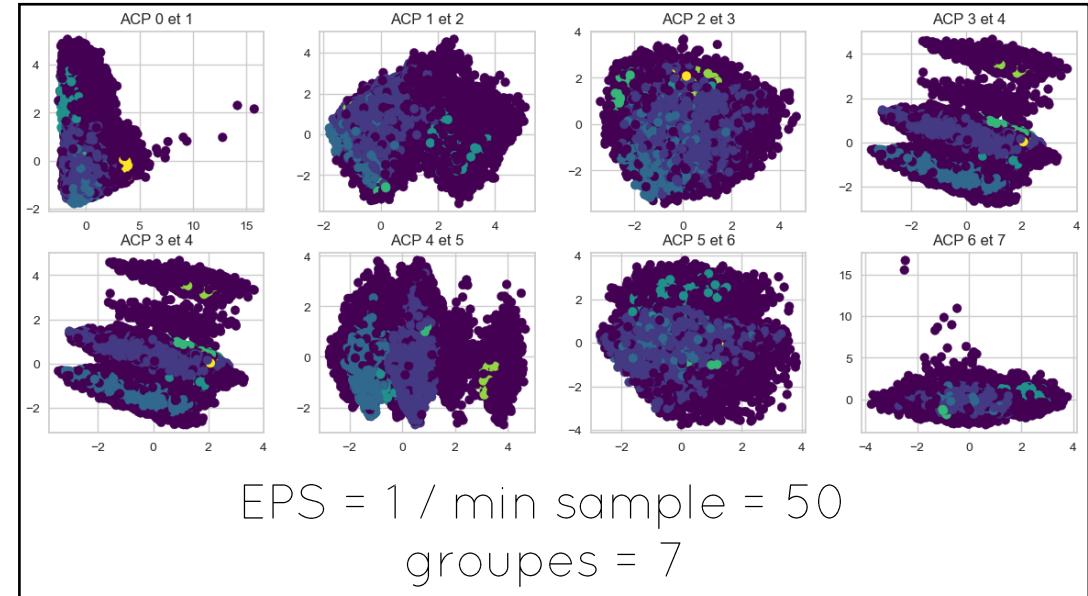
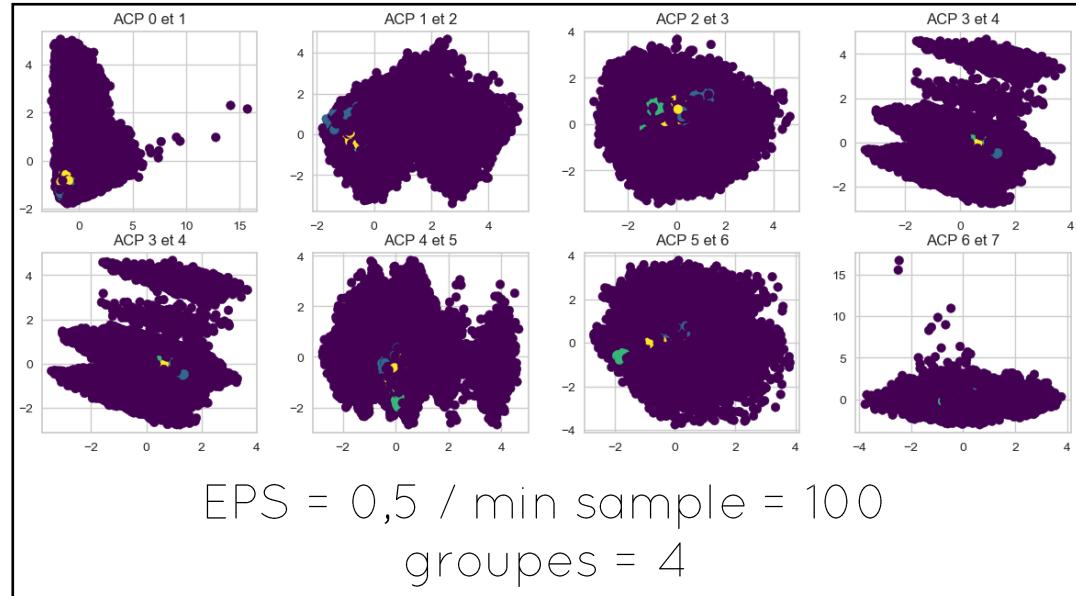
Contrairement au Kmean, le DBSCAN trouve lui-même le nombre de groupe.

Ce en fonction des paramètres epsilon et min_sample (distance maximum entre deux points et nombre de points minimum pour constituer un groupe)

B, DBSCAN

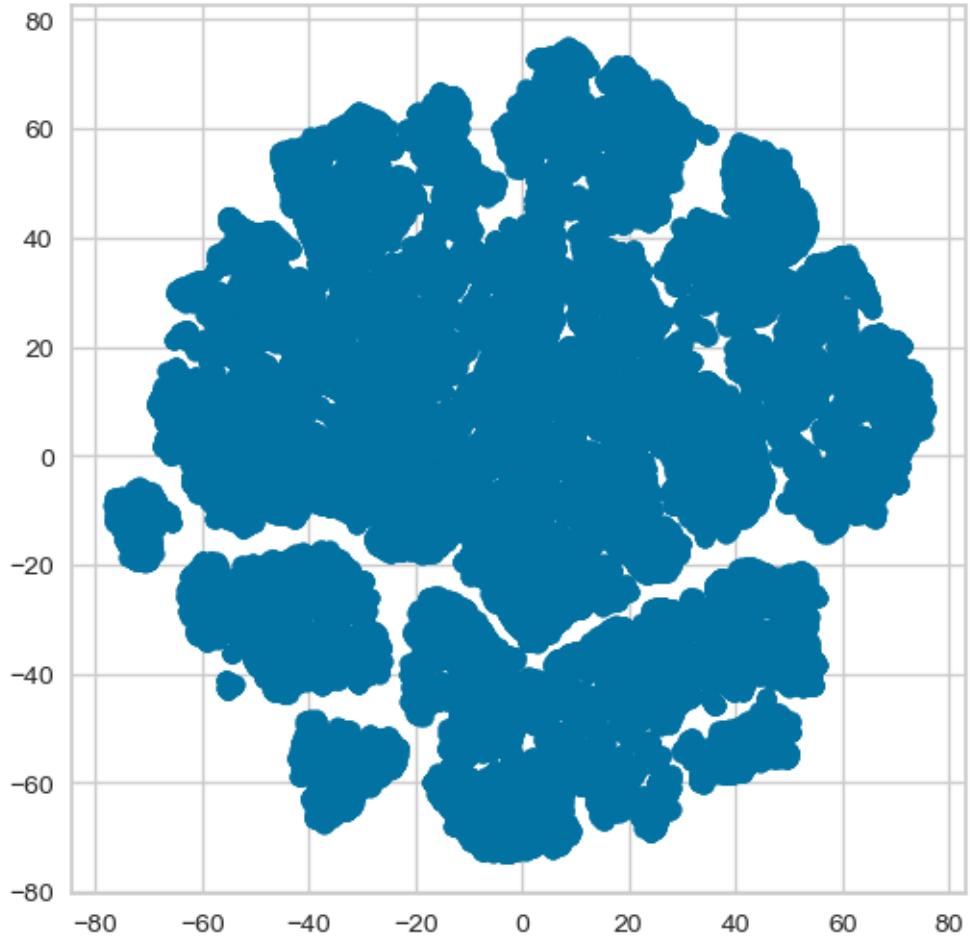
II, PRINCIPAL COMPOSANT ANALYSIS

PARTIE 2



Là encore, les groupes sont trop hétérogènes et mélangés.
La combinaison ACP et DBSCAN n'apparaît pas comme optimale non plus pour nos données.

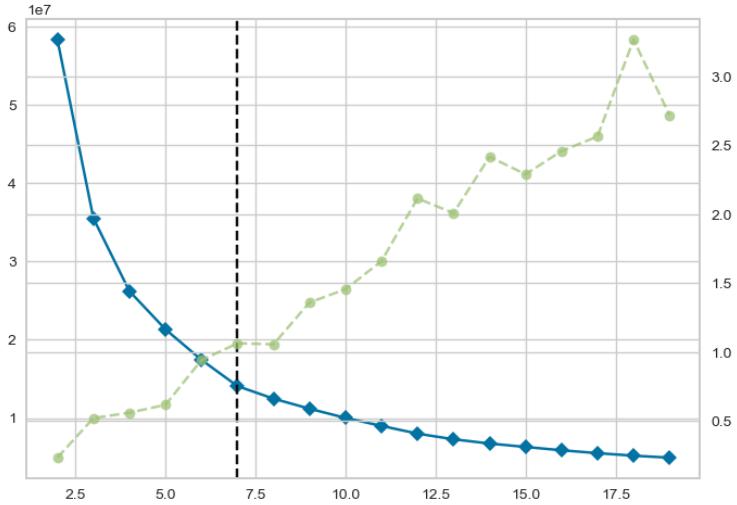
T-S Σ E



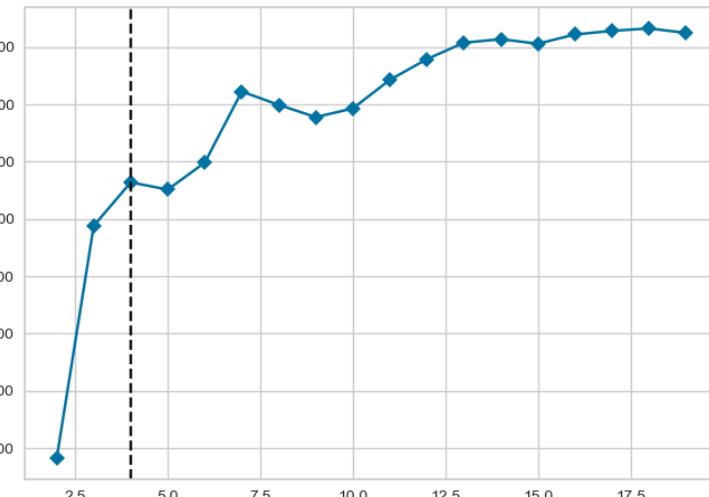
Comme l'ACP, le T-SNE est un outil de visualisation des données.

Il réduit le nombre de dimension tout en gardant les similarités entre points via un calcul de densité reposant sur une distribution T.

Détermination du nombre de groupe par distorsion et dispersion

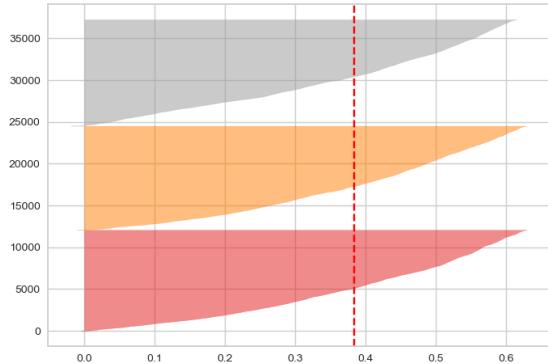
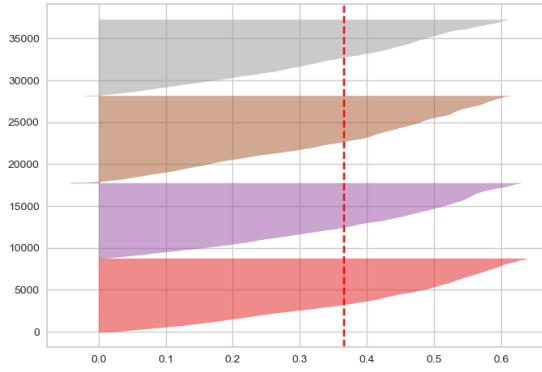
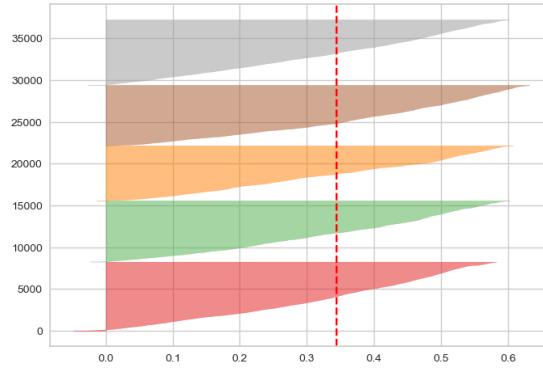
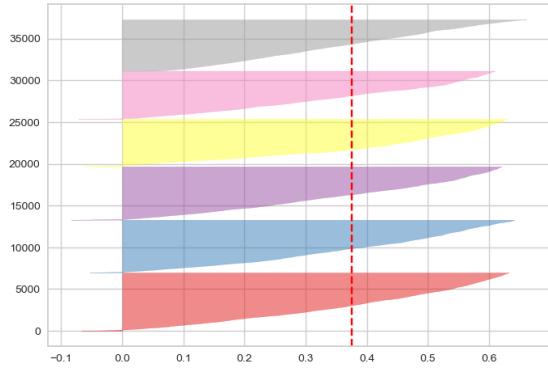
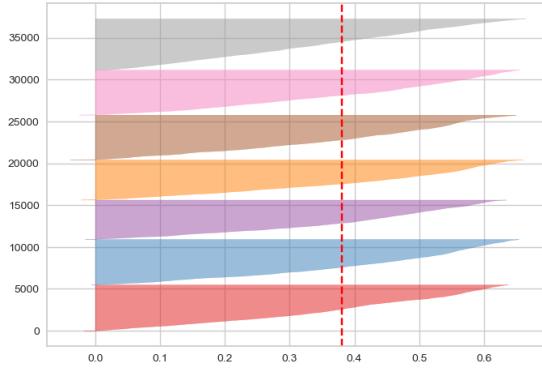
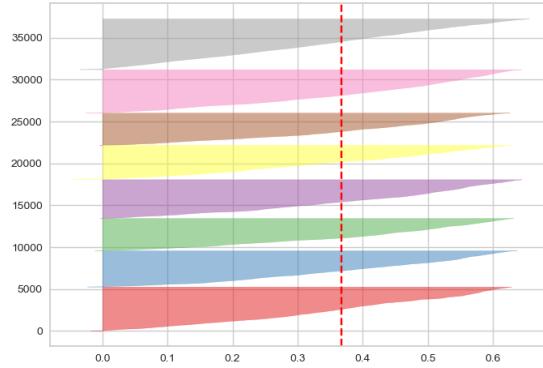


Le calcul de la **distorsion** (SSR entre les points et les centres) nous propose **7 groupes**.

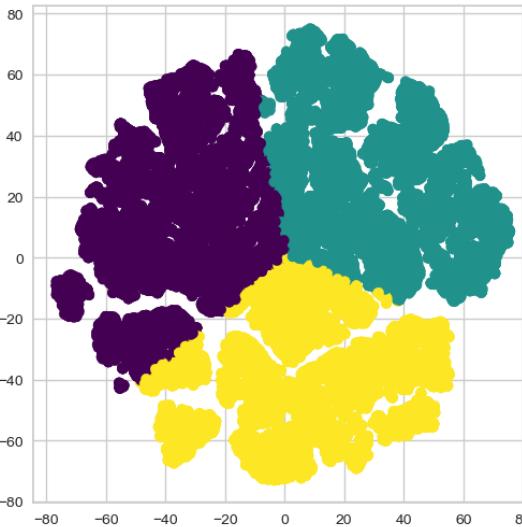


Le calcul par la méthode **Calinski-harabasz**, se basant sur le ratio de dispersion intra et extra groupes, nous propose une solution à **4 groupes**.

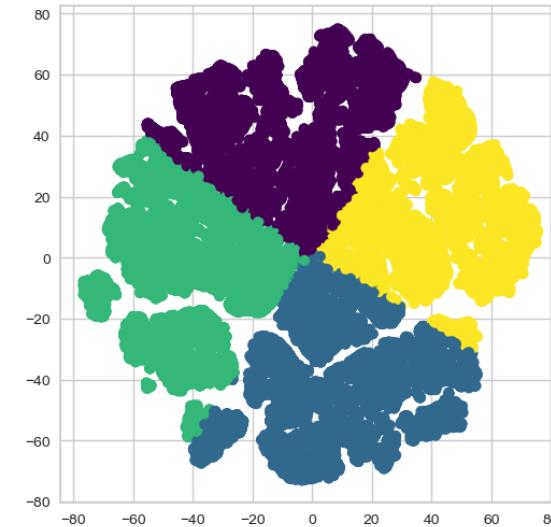
Détermination du nombre de groupe par le coefficient silhouette

 $K = 3$  $K = 4$  $K = 5$  $K = 6$  $K = 7$  $K = 8$ **$K = 7$**

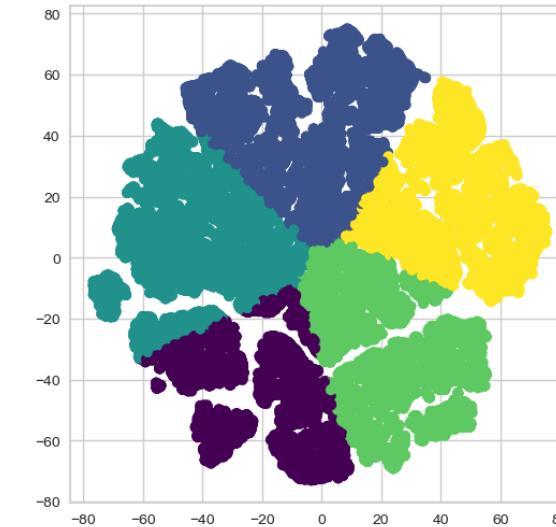
pour toutes les valeurs de K , les groupes apparaissent plus équilibrés qu'avec la PCA.



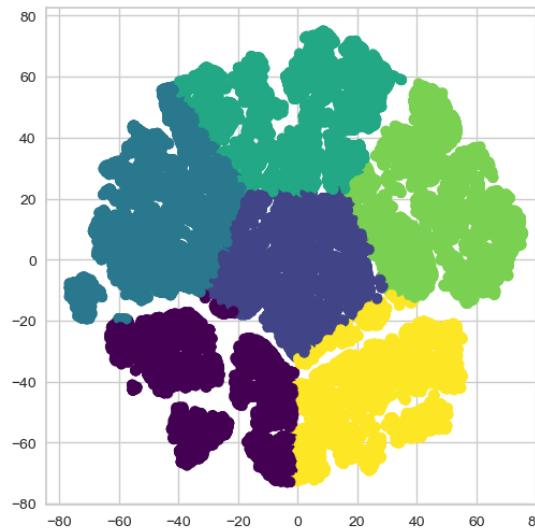
$K = 3$



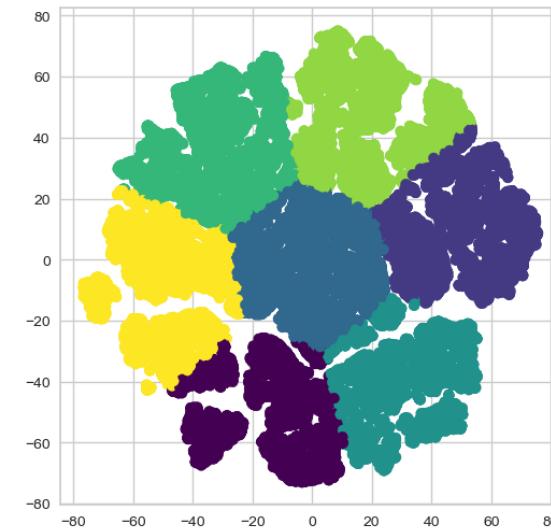
$K = 4$



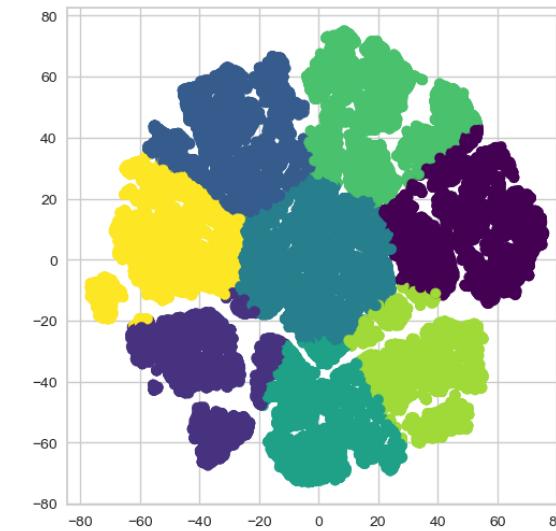
$K = 5$



$K = 6$

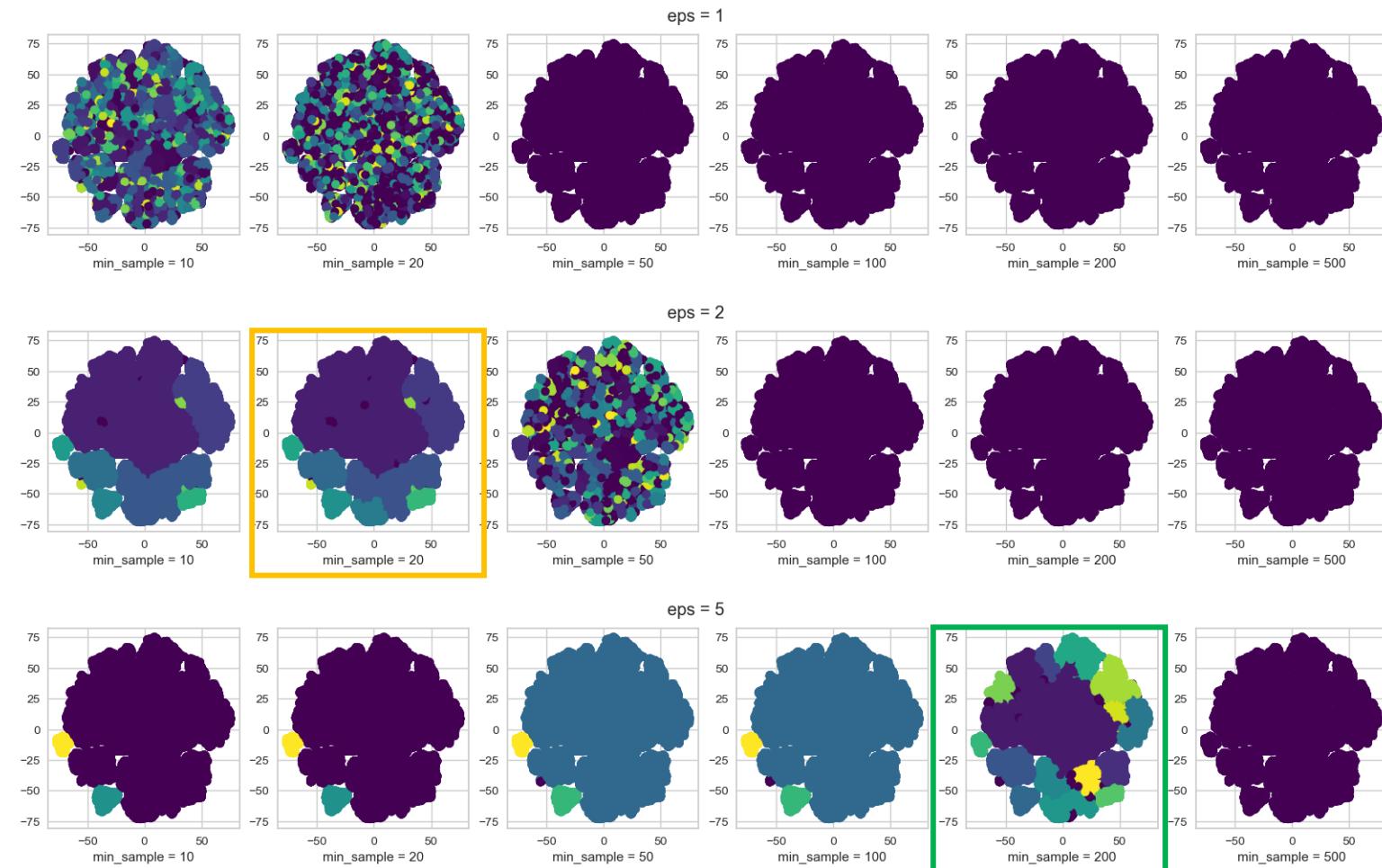


$K = 7$



$K = 8$

Graphiquement tous les groupes sont cohérents et équilibrés.
La combinaison T-SNE et Kmean apparaît comme très pertinente pour partitionner nos données.



	first	second	group	max_g	min_g
0	1	10	423	1582	4
1	1	20	607	15178	13
2	1	50	1	37218	37218
3	1	100	1	37218	37218
4	1	200	1	37218	37218
5	1	500	1	37218	37218
6	2	10	12	21303	20
7	2	20	13	21244	20
8	2	50	193	8961	22
9	2	100	1	37218	37218
10	2	200	1	37218	37218
11	2	500	1	37218	37218
12	5	10	3	35771	477
13	5	20	3	35771	477
14	5	50	4	35740	31
15	5	100	4	35736	35
16	5	200	16	17234	477
17	5	500	1	37218	37218

Le DBSCAN n'arrive pas à partitionner les données aussi bien que le Kmeans.
Les groupes sont soit trop nombreux, soit déséquilibrés.

La combinaison du T-SNE et du Kmean est la solution qui permet la meilleur partition des données.

Toutefois, pour confirmer que cette partition soit utilisable, il faut s'assurer qu'elle soit stable, tant à l'initiation que dans le temps.

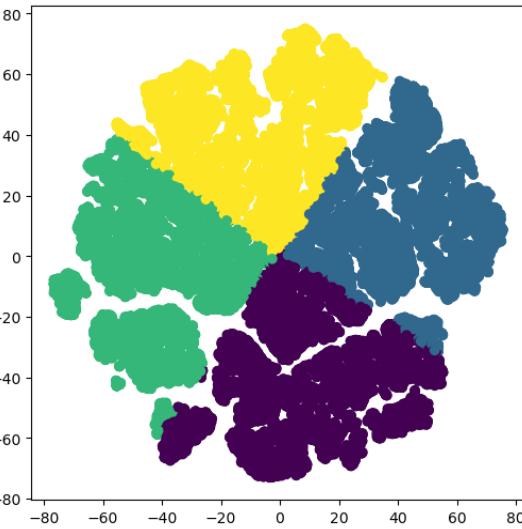
Les différentes méthodes de détermination de groupes pour le Kmean nous ont orientés vers les valeurs 4 et 7.

Cependant, nous allons choisir 6 groupes, seule valeur amenant une stabilité suffisante.

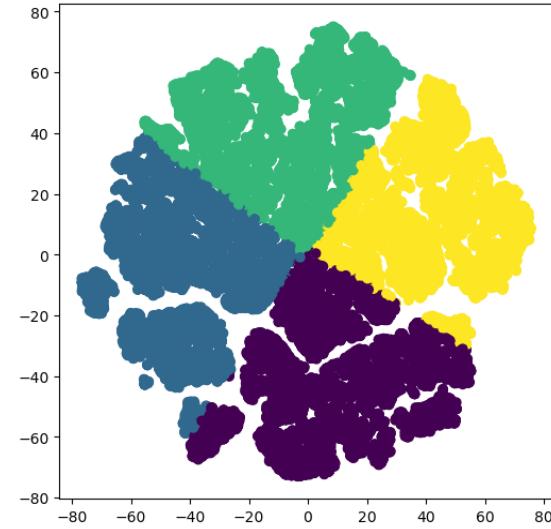
A, TEST DE STABILITÉ POUR K=4

IV, TEST DE STABILITÉ

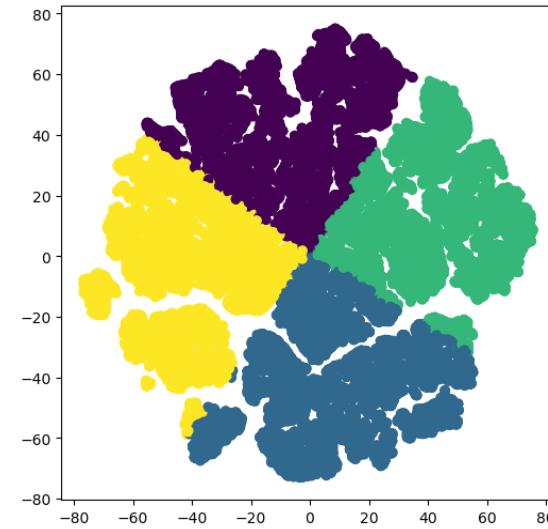
PARTIE 2



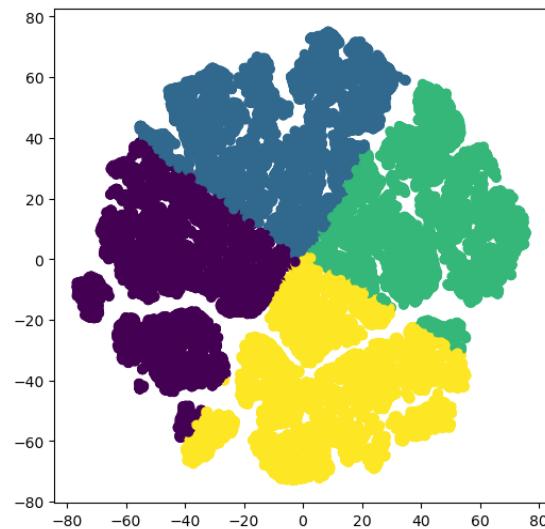
itération 1



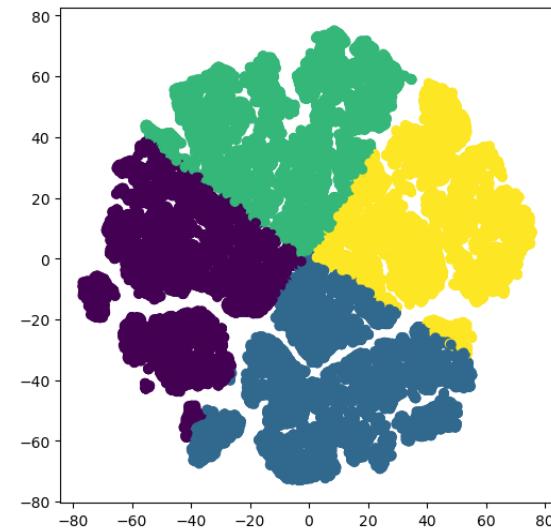
itération 2



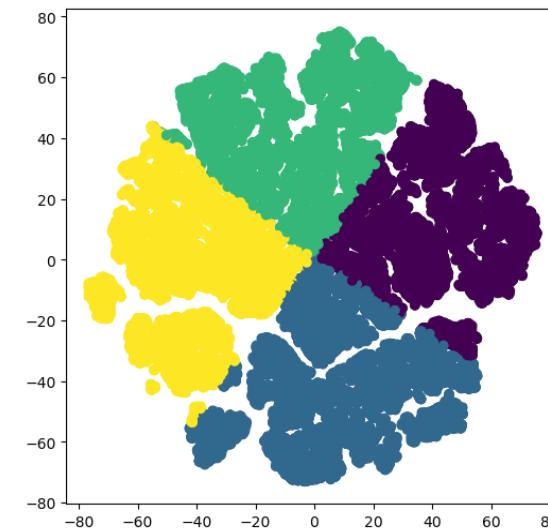
itération 3



itération 4



itération 5



itération 6

Initiation
RANDOM en
boucle

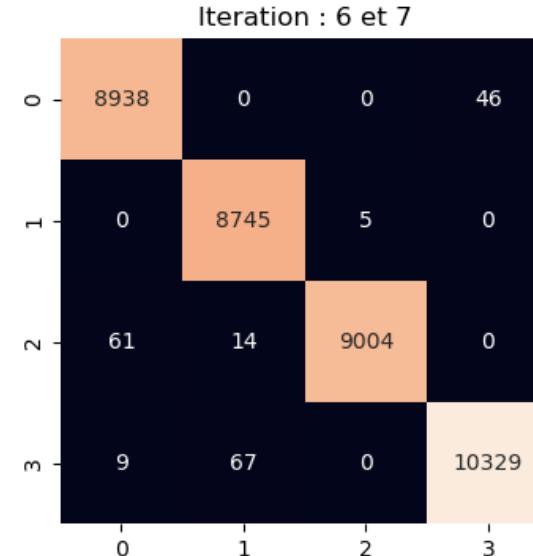
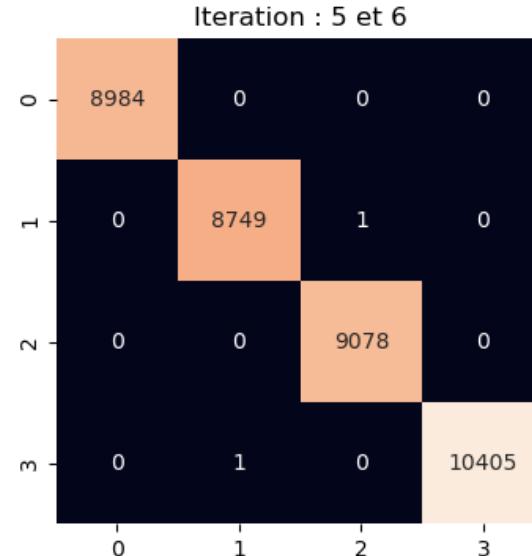
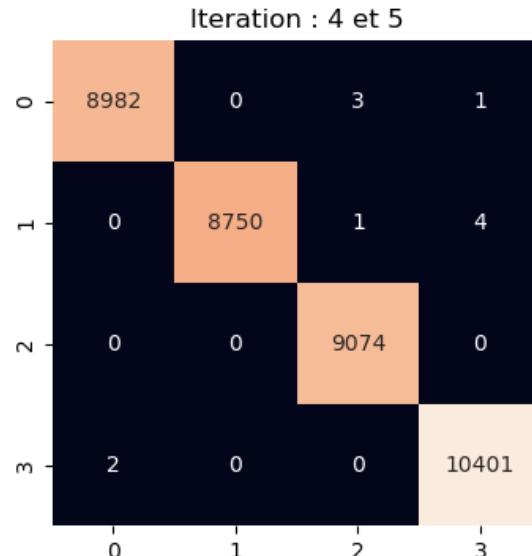
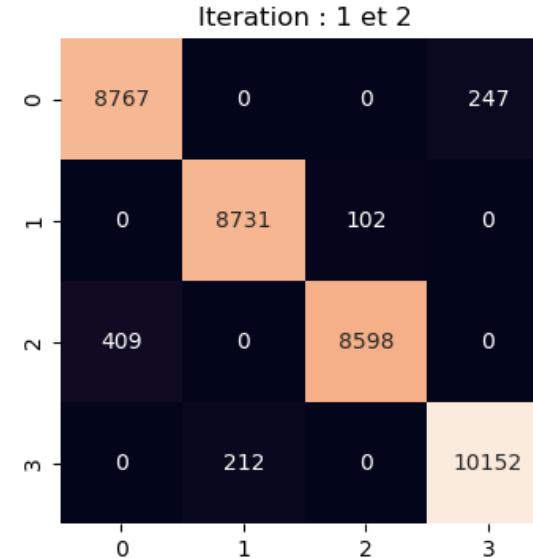
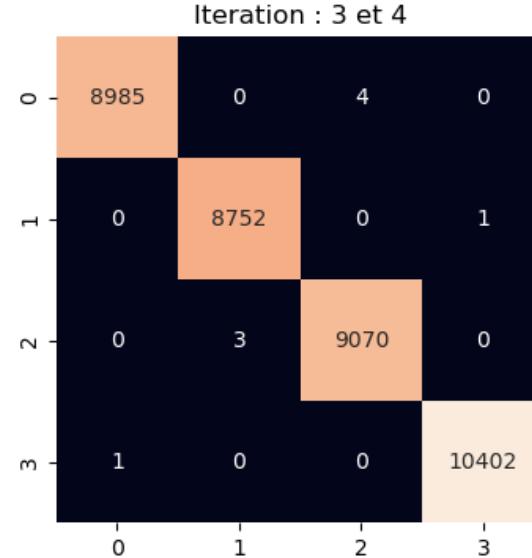
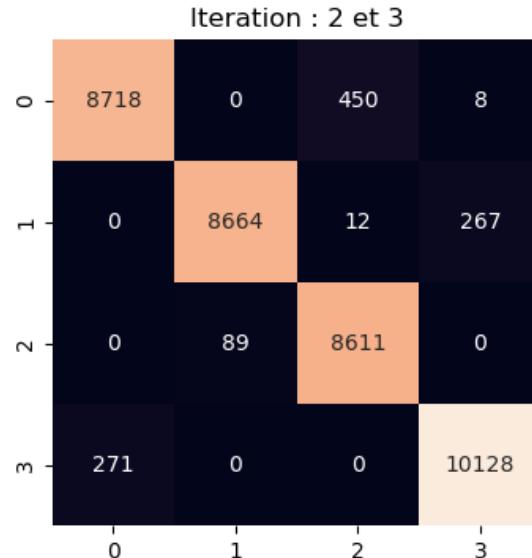
Pas de
différence
notable
graphiquement.

A, TEST DE STABILITÉ POUR K=4

IV, TEST DE STABILITÉ

PARTIE 2

Test numérique de stabilité à l'initiative avec K=4



D'une initiation à l'autre, les groupes se recoupent.

Les matrices sont quasiment diagonales.

A, TEST DE STABILITÉ POUR K=4

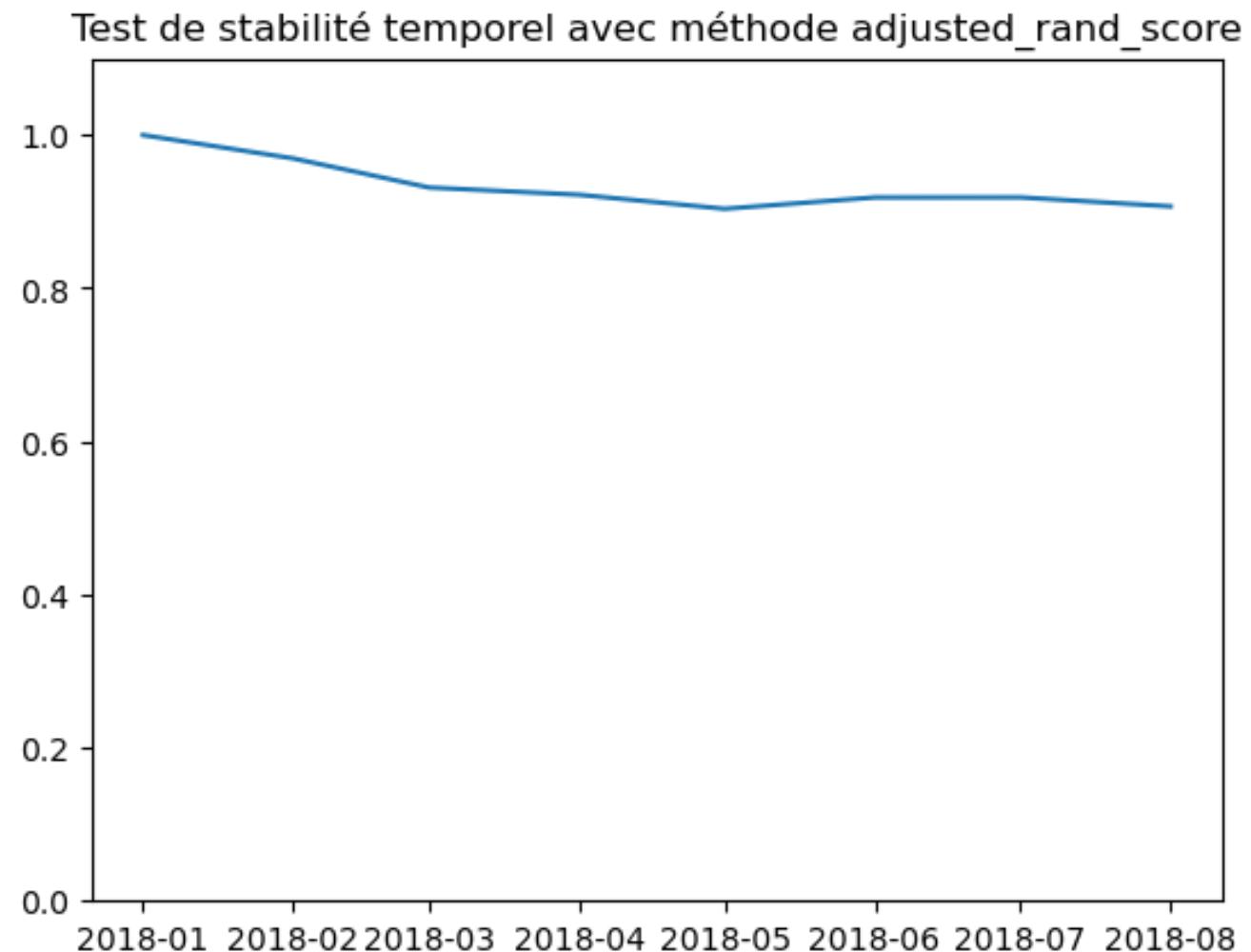
IV, TEST DE STABILITÉ

PARTIE 2

On test la stabilité temporelle avec la méthode adjusted random score (**ARI**) en comparant les données prévues et réelles.

L'abscisse indique la date de fin de la période, la date de début est invariablement le 1^{er} janvier 2017.

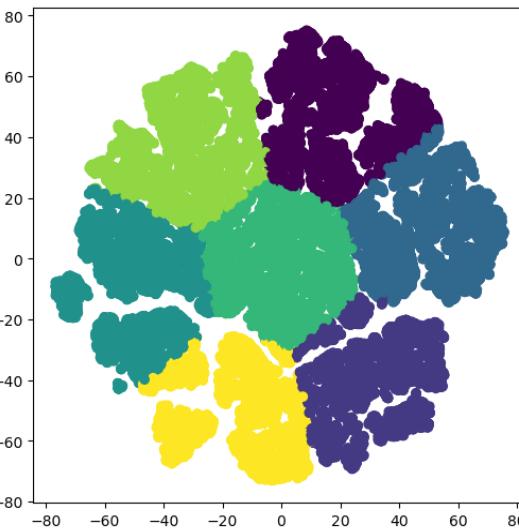
Les groupes **stables** dans le temps, le recul que nous avons ne fait **pas apparaître de cassure**.



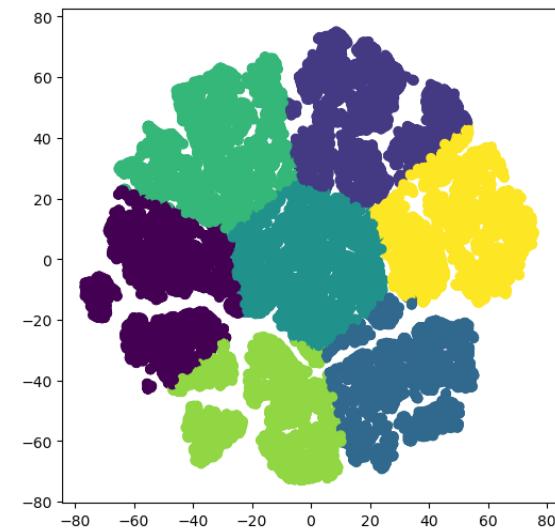
B, TEST DE STABILITÉ POUR K=7

IV, TEST DE STABILITÉ

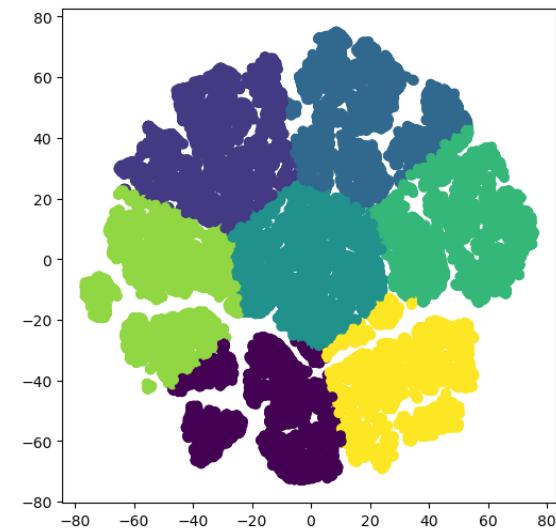
PARTIE 2



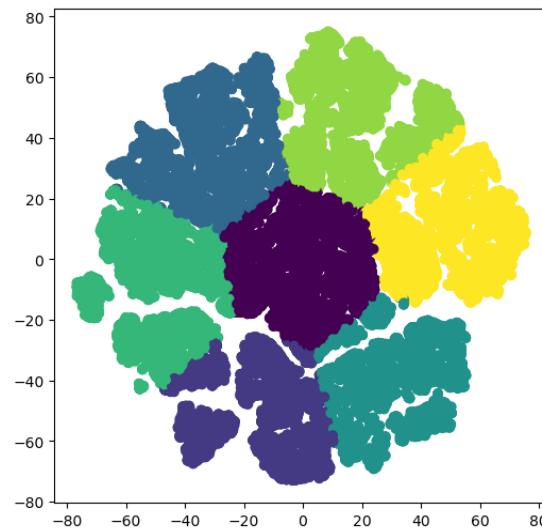
itération 1



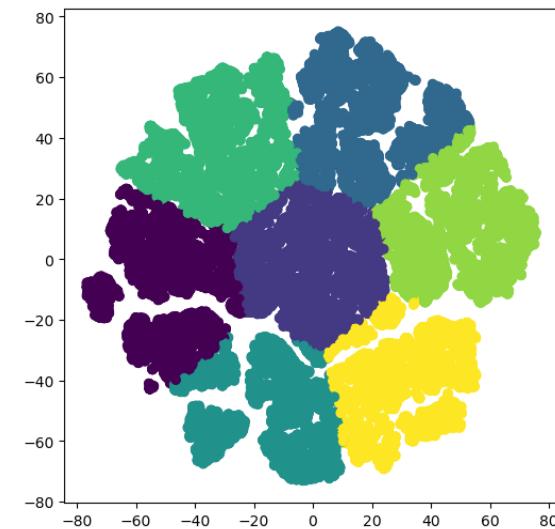
itération 2



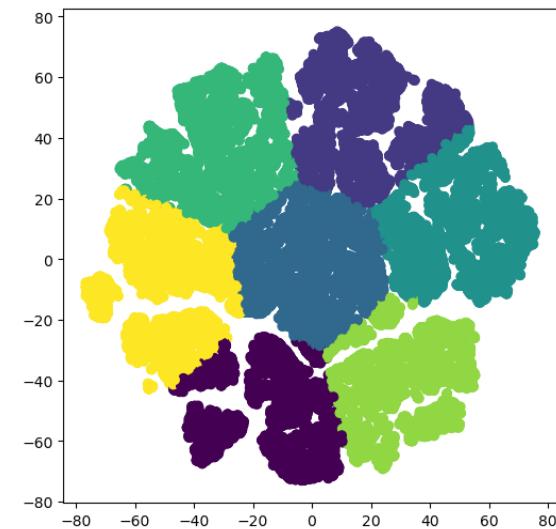
itération 3



itération 4



itération 5



itération 6

Initiation
RANDOM en
boucle

Pas de
différence
notable
graphiquement.

B, TEST DE STABILITÉ POUR K=7

IV, TEST DE STABILITÉ

PARTIE 2

Test numérique de stabilité à l'initiative avec K=7

Iteration : 2 et 3							
0	1	2	3	4	5	6	7
0	-5360	0	7	0	0	0	1
1	0	5416	0	0	0	0	0
2	0	0	5331	0	11	1	0
3	0	2	0	4780	0	4	0
4	0	0	0	24	4665	2	0
5	5	0	1	0	0	6092	5
6	0	18	0	0	0	0	5493

Iteration : 3 et 4							
0	1	2	3	4	5	6	7
0	-5360	0	0	0	0	5	0
1	0	5415	0	2	0	0	19
2	6	0	5332	0	0	1	0
3	0	0	0	4780	24	0	0
4	0	0	9	0	4667	0	0
5	0	0	1	4	2	6092	0
6	1	0	0	0	0	5	5493

Iteration : 1 et 2							
0	1	2	3	4	5	6	7
0	-5367	0	3	0	0	0	0
1	0	5414	0	0	0	0	6
2	0	0	5339	0	5	0	0
3	0	2	0	4776	0	0	0
4	0	0	0	10	4686	2	0
5	0	0	1	0	0	6101	0
6	1	0	0	0	0	0	5505

Iteration : 4 et 5							
0	1	2	3	4	5	6	7
0	-5367	0	0	0	0	0	0
1	0	5413	0	1	0	1	0
2	2	0	5340	0	0	0	0
3	0	0	0	4782	4	0	0
4	0	0	6	0	4687	0	0
5	0	0	0	0	2	6101	0
6	0	0	0	0	0	5	5507

Iteration : 5 et 6							
0	1	2	3	4	5	6	7
0	-5361	0	8	0	0	0	0
1	0	5392	0	0	0	2	19
2	0	0	5337	0	9	0	0
3	0	7	0	4767	0	9	0
4	0	0	0	6	4683	4	0
5	8	0	14	0	0	6080	5
6	0	0	0	0	0	0	5499

Iteration : 6 et 7							
0	1	2	3	4	5	6	7
0	-5361	0	0	0	0	8	8
1	0	5392	0	7	0	0	0
2	9	0	5336	0	0	14	0
3	0	0	0	4767	6	0	0
4	0	0	9	0	4683	0	0
5	0	2	0	9	4	6080	0
6	0	19	0	0	0	3	5501

D'une initiation à l'autre, les groupes se recoupent.

Les matrices sont quasiment diagonales.

B, TEST DE STABILITÉ POUR K=7

IV, TEST DE STABILITÉ

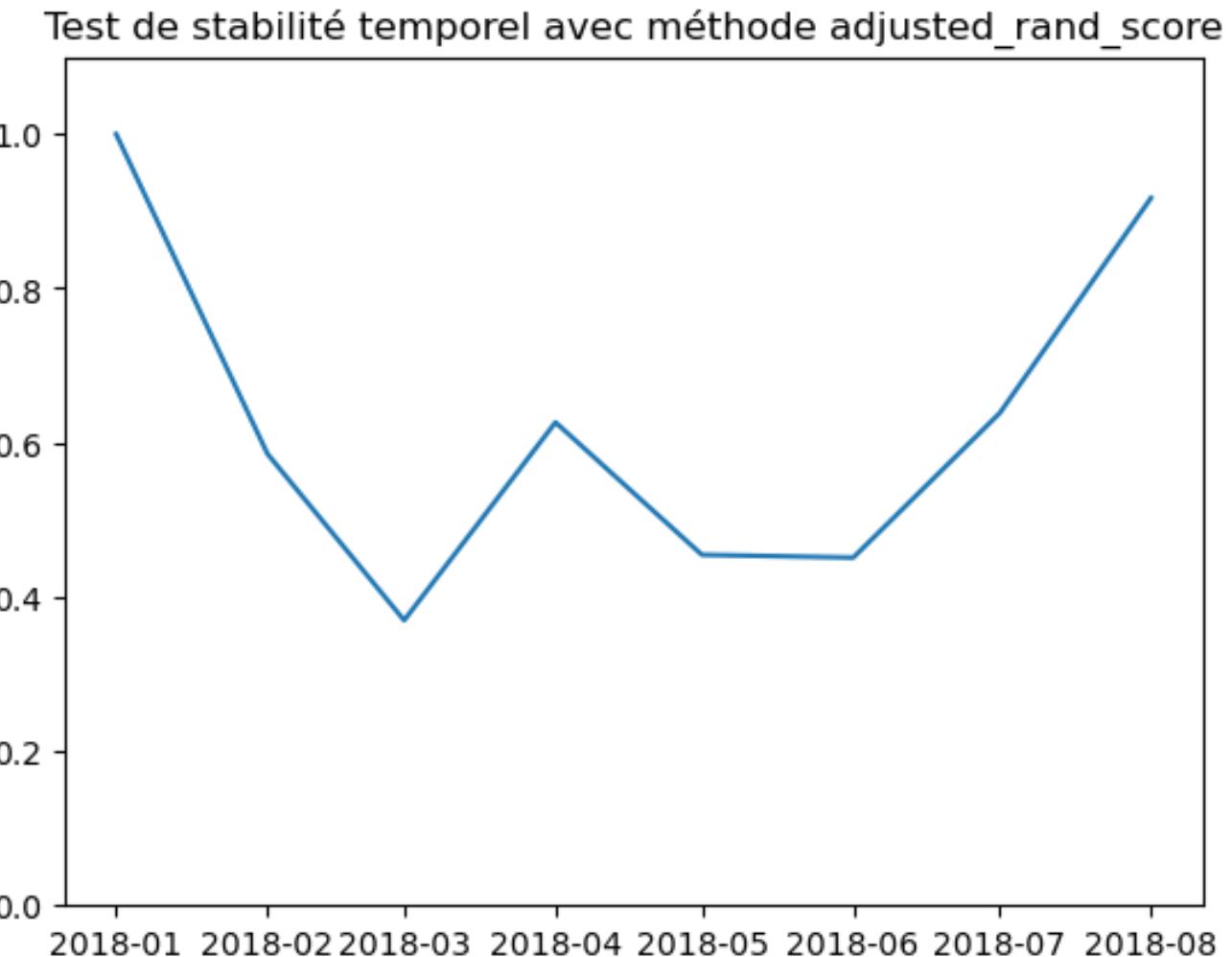
PARTIE 2

On test la stabilité temporelle avec la méthode adjusted random score (**ARI**) en comparant les données prévues et réelles.

L'abscisse indique la date de fin de la période, la date de début est invariablement le 1^{er} janvier 2017.

Dans ce cas de figure, le coefficient ARI chute dès le 2^{ème} mois t atteint son minimum au 3^{ème}.

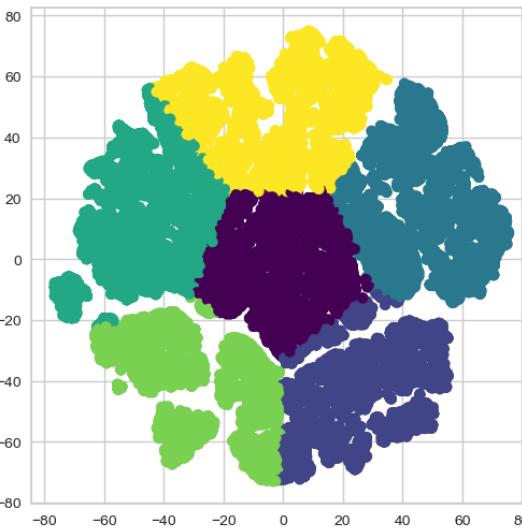
Cette solution n'est pas stable temporellement.



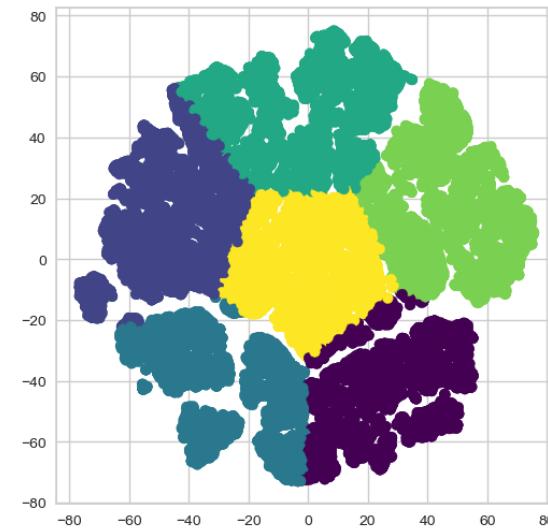
C, TEST DE STABILITÉ POUR $K=6$

IV, TEST DE STABILITÉ

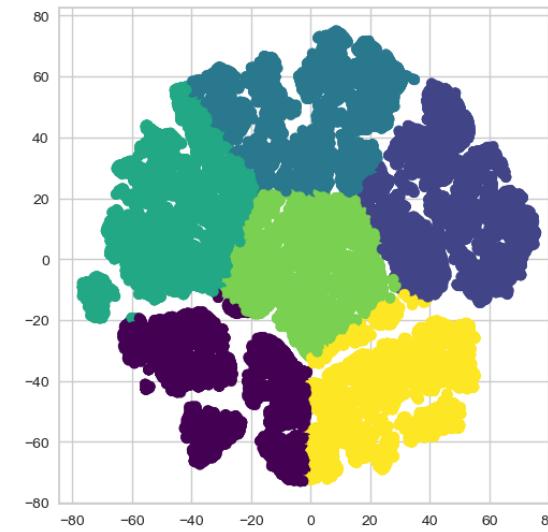
PARTIE 2



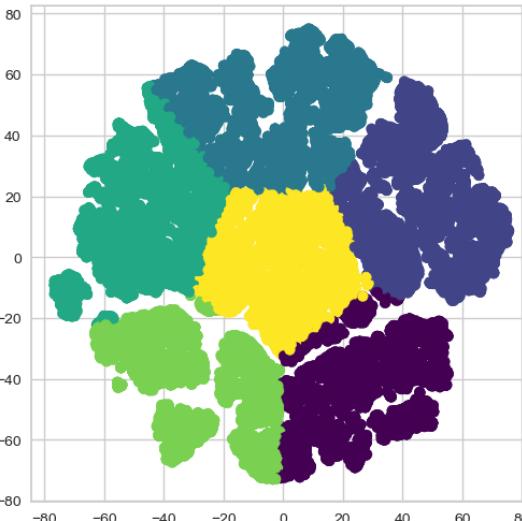
itération 1



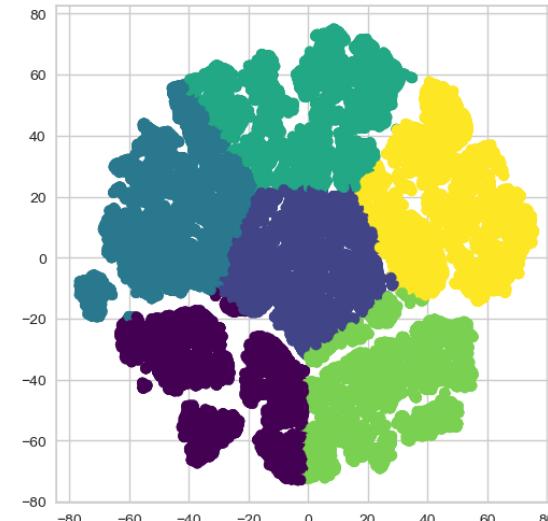
itération 2



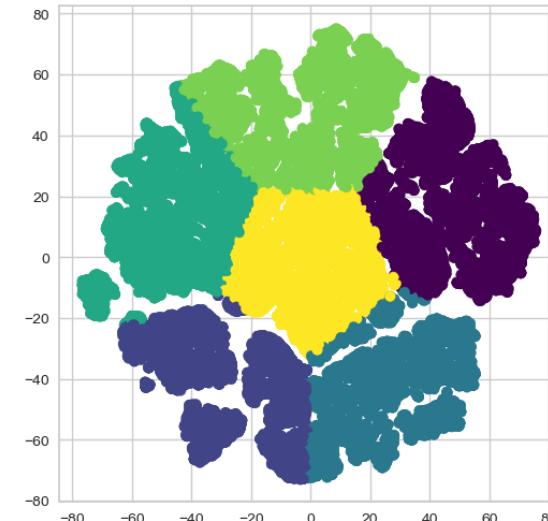
itération 3



itération 4



itération 5



itération 6

Initiation
RANDOM en
boucle

Pas de
différence
notable
graphiquement.

C, TEST DE STABILITÉ POUR K=6

IV, TEST DE STABILITÉ

PARTIE 2

Test numérique de stabilité à l'initiative

Iteration : 1 et 2						
0	1	2	3	4	5	6
0	6197	0	197	0	0	0
1	0	6937	0	18	0	21
2	0	17	6267	0	0	8
3	0	0	0	5621	82	18
4	43	0	0	0	5650	1
5	18	0	2	1	0	6120

Iteration : 2 et 3						
0	1	2	3	4	5	6
0	6205	0	0	0	53	0
1	0	6861	17	0	0	76
2	167	0	6299	0	0	0
3	0	31	0	5464	0	145
4	0	0	0	228	5504	0
5	77	0	7	0	75	6009

Iteration : 3 et 4						
0	1	2	3	4	5	6
0	6374	0	0	0	0	75
1	0	6872	0	20	0	0
2	14	0	6297	0	0	12
3	0	0	0	5565	127	0
4	6	0	0	0	5556	70
5	0	76	2	152	0	6000

Iteration : 4 et 5						
0	1	2	3	4	5	6
0	6366	0	1	0	0	27
1	0	6934	0	14	0	0
2	8	0	6291	0	0	0
3	0	0	0	5633	97	7
4	7	0	0	0	5658	18
5	0	33	11	28	0	6085

Iteration : 5 et 6						
0	1	2	3	4	5	6
0	6101	0	280	0	0	0
1	0	6957	0	0	0	10
2	0	21	6276	0	0	6
3	0	0	0	5640	9	26
4	53	0	0	0	5702	0
5	15	0	7	0	0	6115

Iteration : 6 et 7						
0	1	2	3	4	5	6
0	6102	0	0	0	65	2
1	0	6930	21	0	0	27
2	265	0	6288	0	0	10
3	0	25	0	5593	0	22
4	0	0	0	148	5563	0
5	25	0	0	16	18	6098

D'une initiation à l'autre, les groupes se recoupent.

Les matrices sont quasiment diagonales.

C, TEST DE STABILITÉ POUR K=6

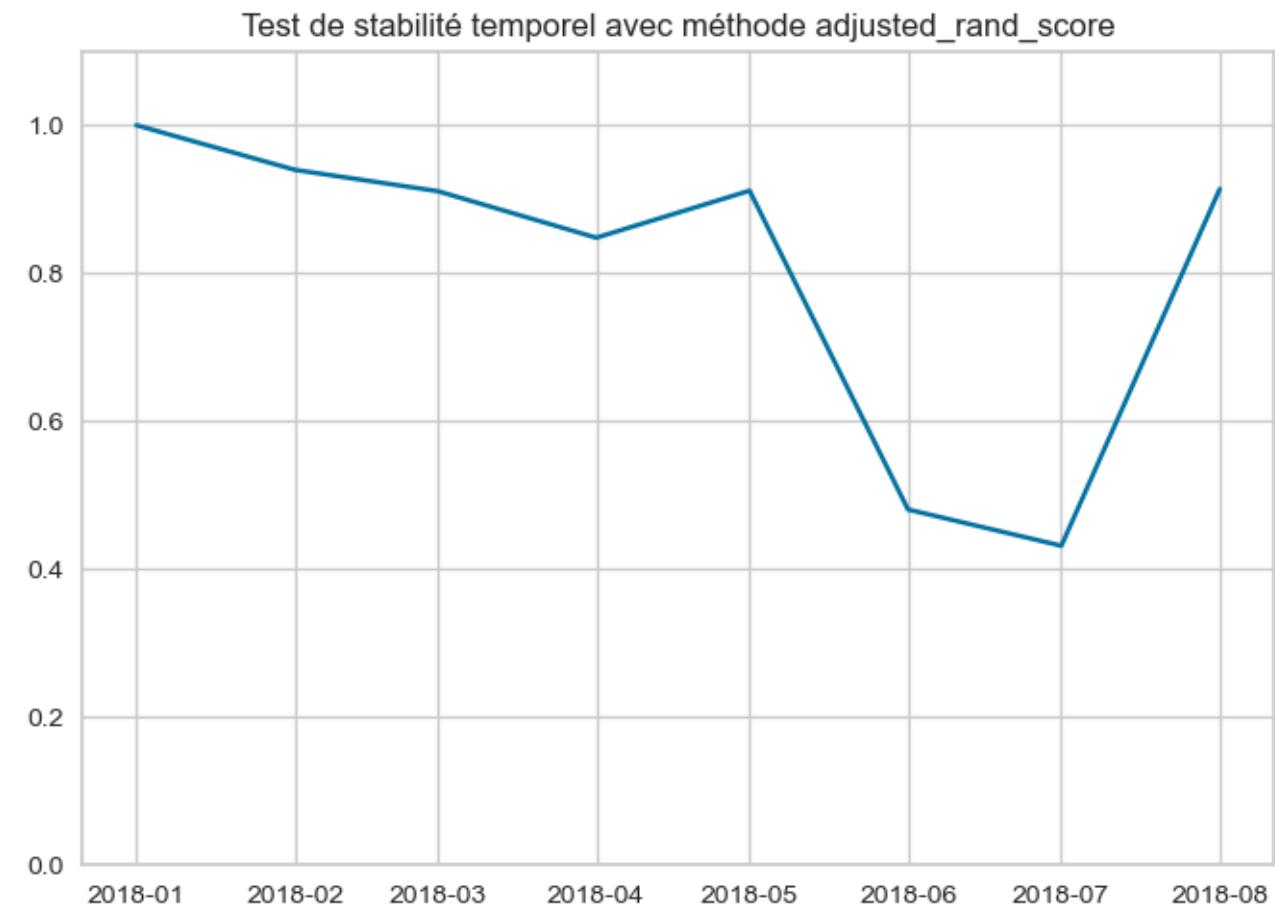
IV, TEST DE STABILITÉ

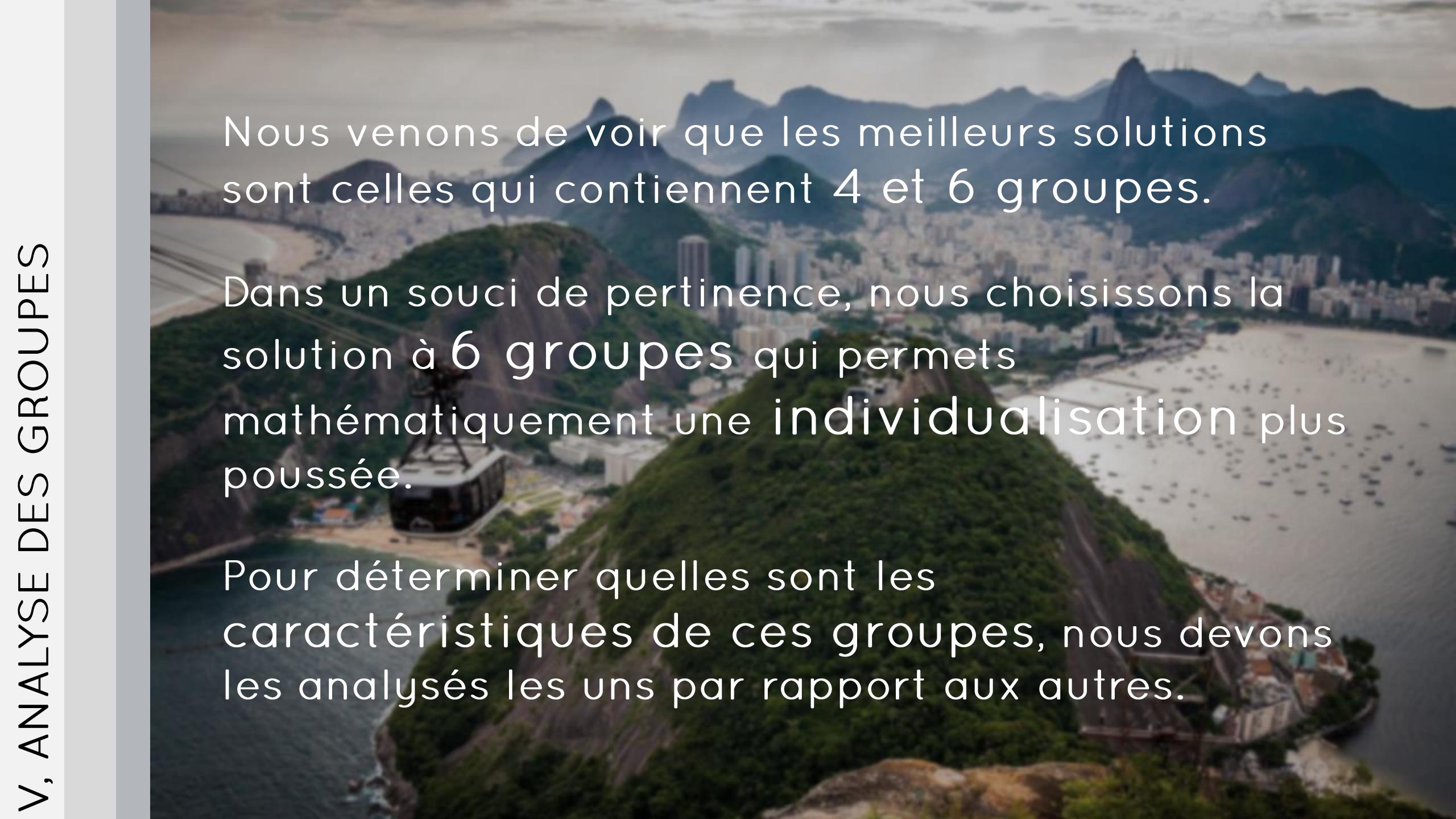
PARTIE 2

On test la stabilité temporelle avec la méthode adjusted random score (**ARI**) en comparant les données prévues et réelles.

L'abscisse indique la date de fin de la période, la date de début est invariablement le 1^{er} janvier 2017.

On note une **cassure** claire du ARI entre **mai et juin**.
On préconise donc une mise à jour de la segmentation tous les **5 mois**.

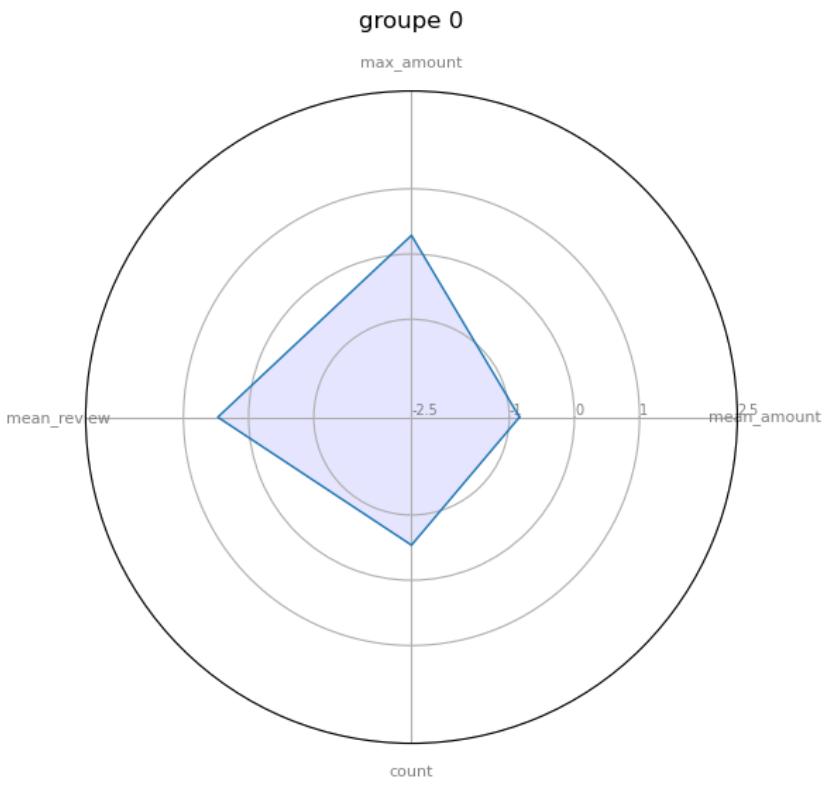


A scenic aerial photograph of Rio de Janeiro, Brazil. The image captures the city's unique topography where dense urban areas are built on the slopes of several large, green mountains. In the foreground, a cable car station is visible on a hillside. A winding road or path leads up the mountain. The city extends along a coastline where a large body of water meets the shore. The sky is overcast with soft, diffused light.

Nous venons de voir que les meilleures solutions sont celles qui contiennent 4 et 6 groupes.

Dans un souci de pertinence, nous choisissons la solution à 6 groupes qui permets mathématiquement une individualisation plus poussée.

Pour déterminer quelles sont les caractéristiques de ces groupes, nous devons les analysés les uns par rapport aux autres.

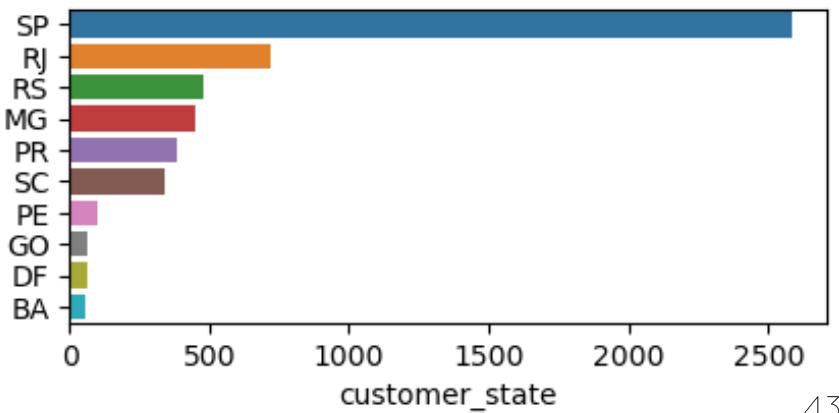
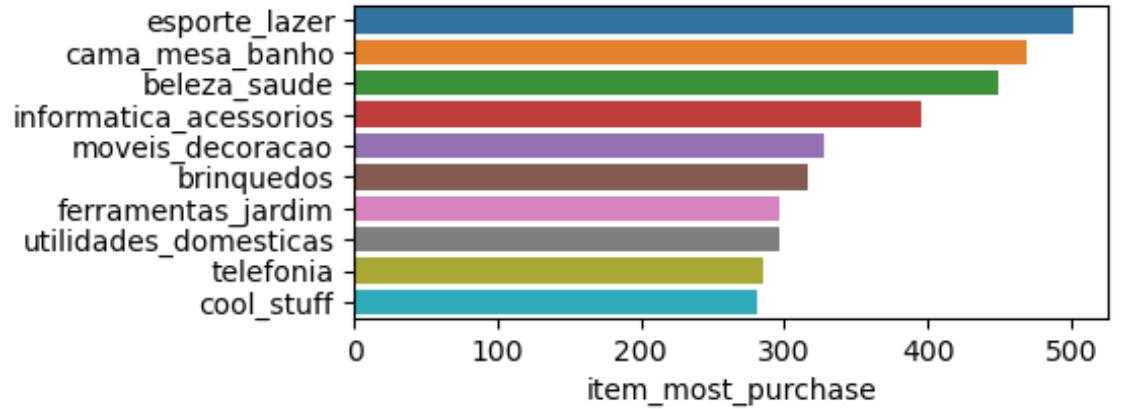
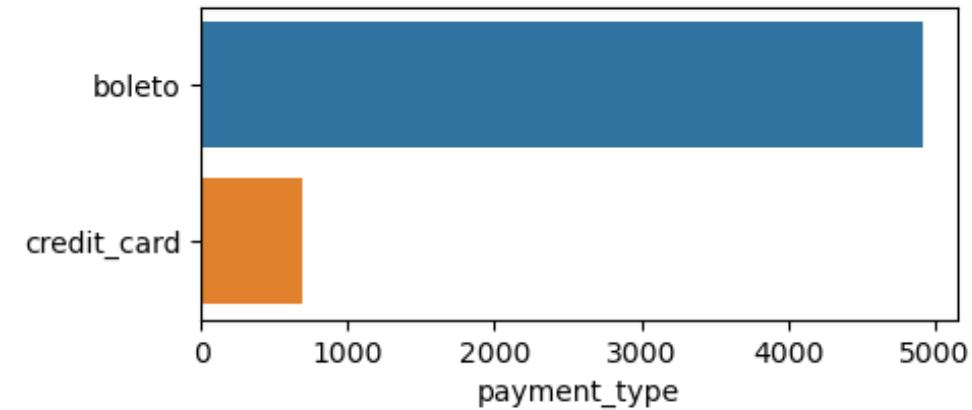


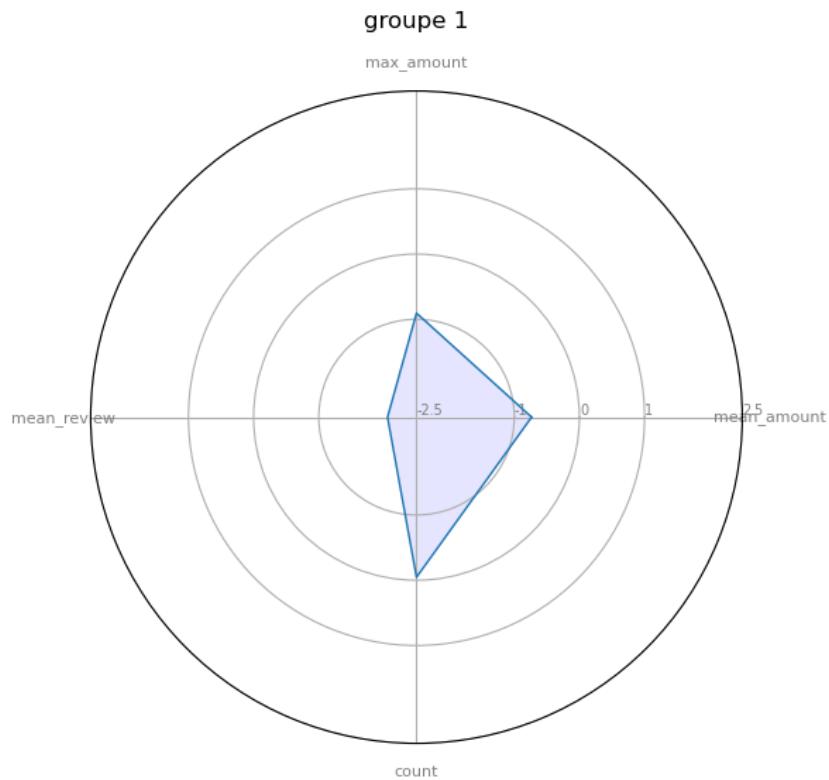
Volatiles

Un achat, paiement en cash principalement.
Petits montants.

Client relativement satisfaits.
Majoritairement de Sao Paulo.

Marketing promotionnel, vente flash.
Clients impulsifs.





Petits budget déçus

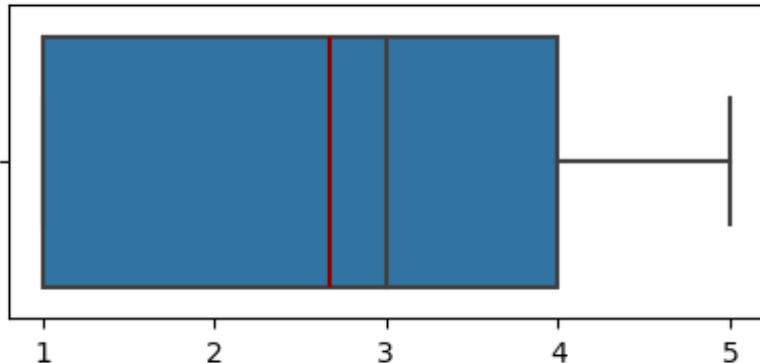
Petites transactions, linge de maison et produits ménagers.

Mauvaise notation.

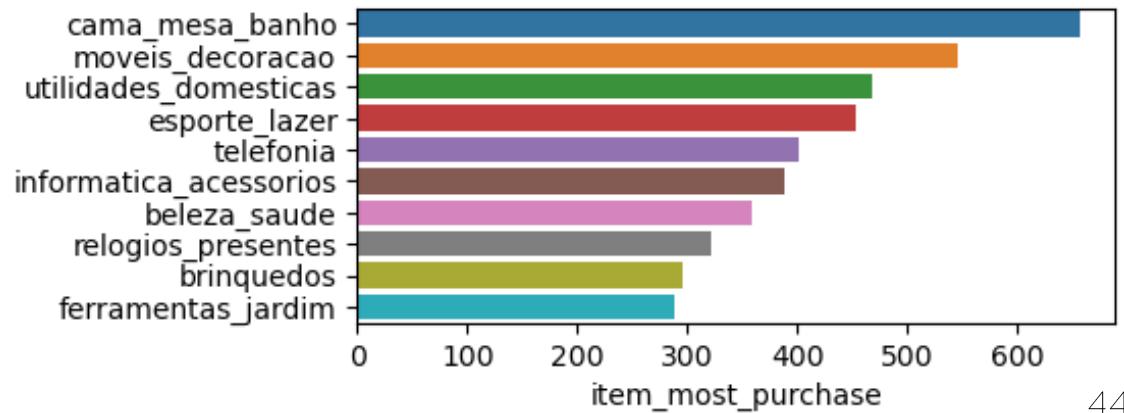
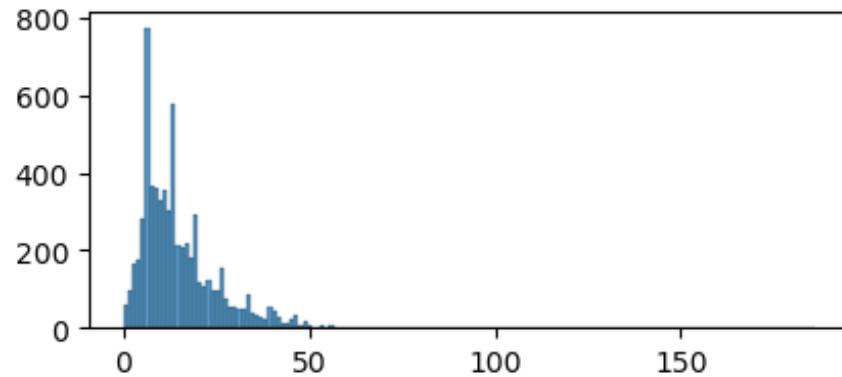
Voir quels produits centralisent les critiques.

Peu à faire pour les clients.

Notes moyennes



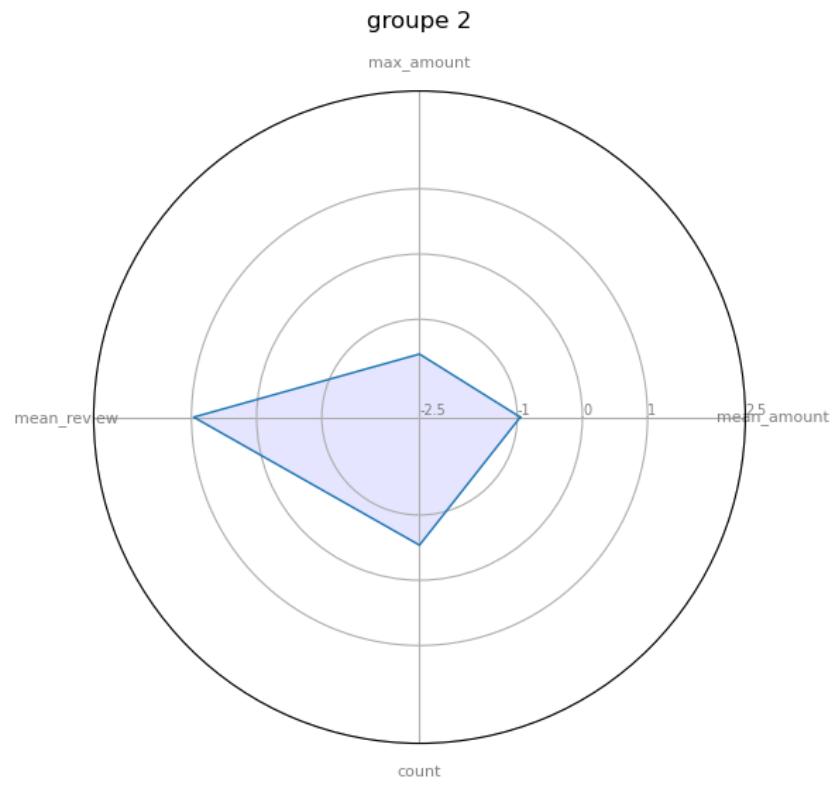
Délai de livraison



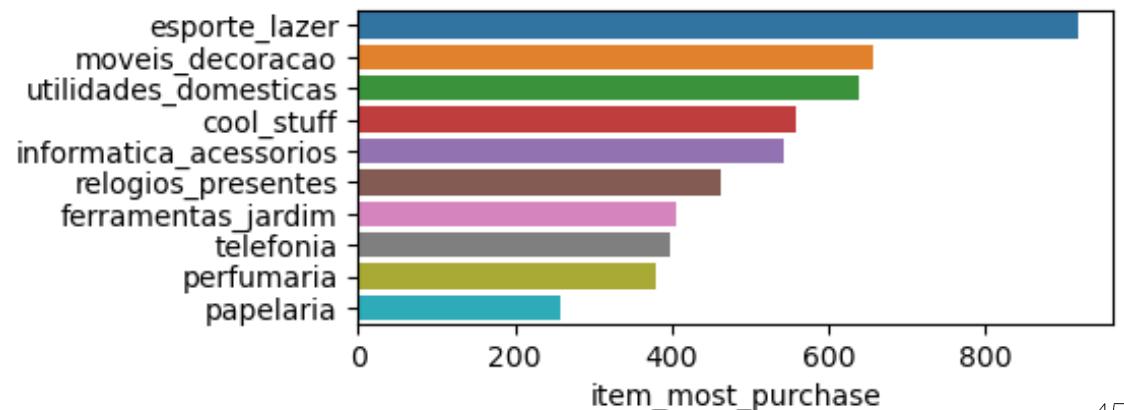
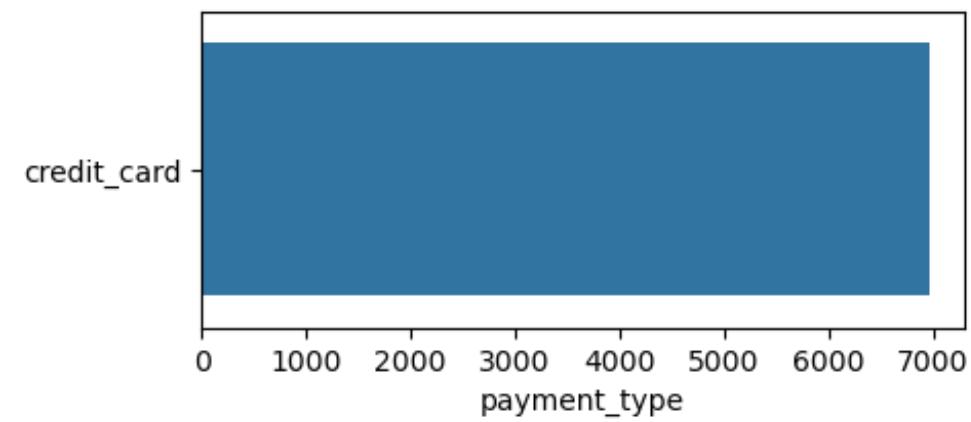
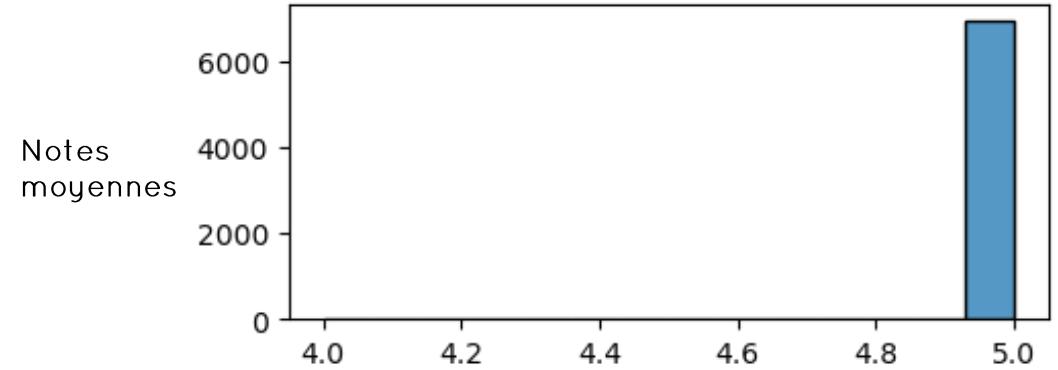
GROUPE 2

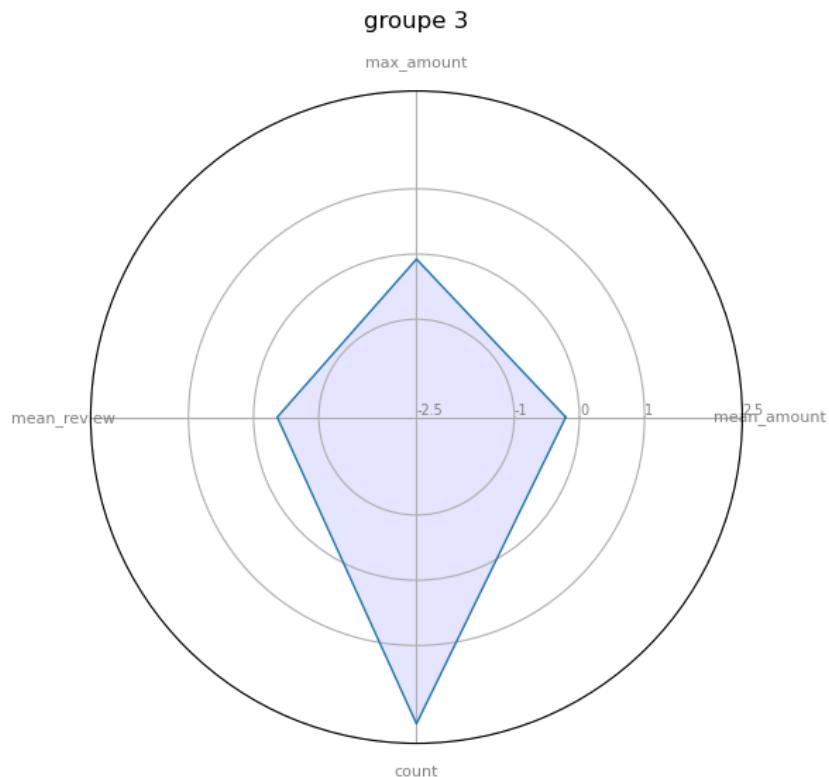
V. ANALYSE DES GROUPES

PARTIE 2



Petits budget très satisfaits.
Petites transactions, sport et loisir, petite décoration.
Excellente notation.
Paiements en carte de crédit exclusivement.
Envoi régulier de promotions sur les produits des catégories ciblées.





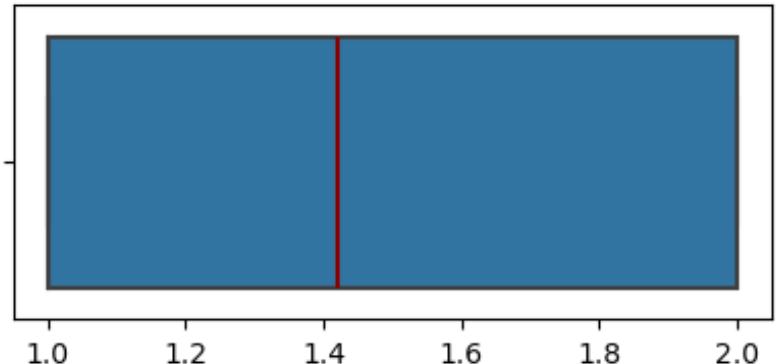
Client potentiellement réguliers

Dépenses moyennes, notation correcte.

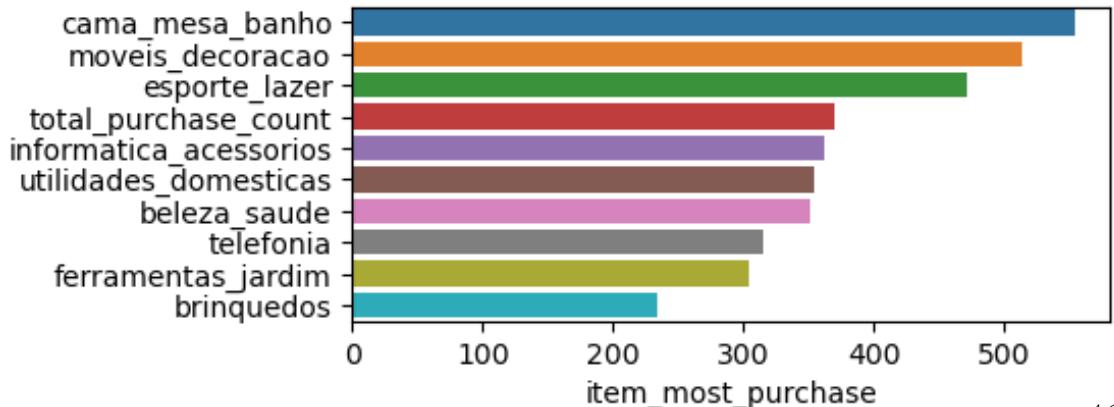
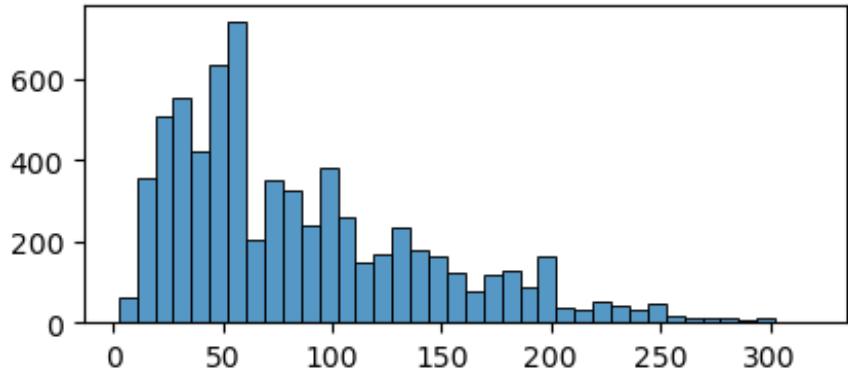
Ont, où peuvent potentiellement, réaliser plusieurs achats.

Client à cibler. Entretenir un lien via newsletter et promotions.

Nombre d'achats.



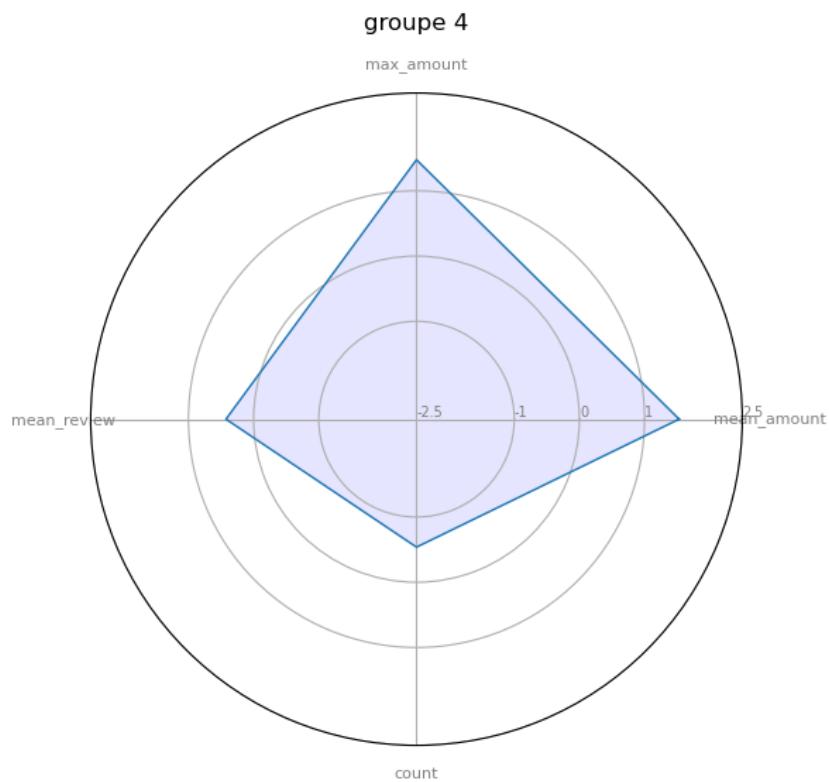
Montants achats



GROUPE 4

V. ANALYSE DES GROUPES

PARTIE 2



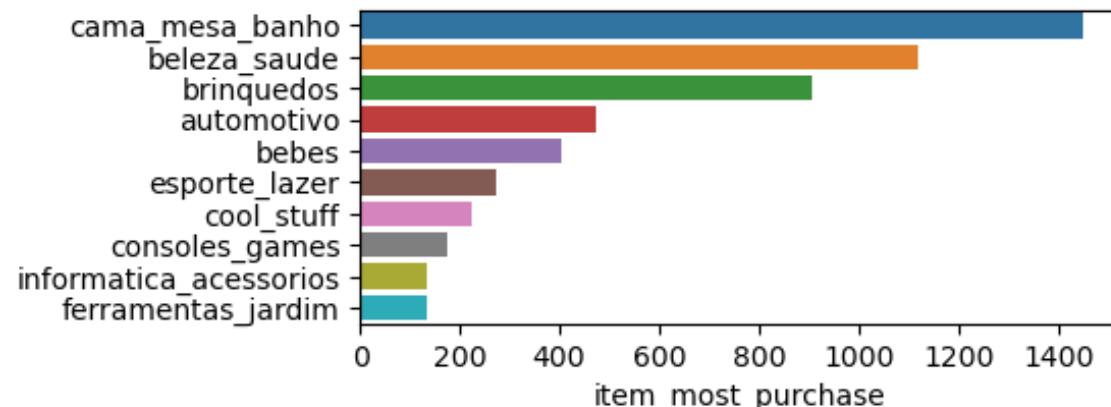
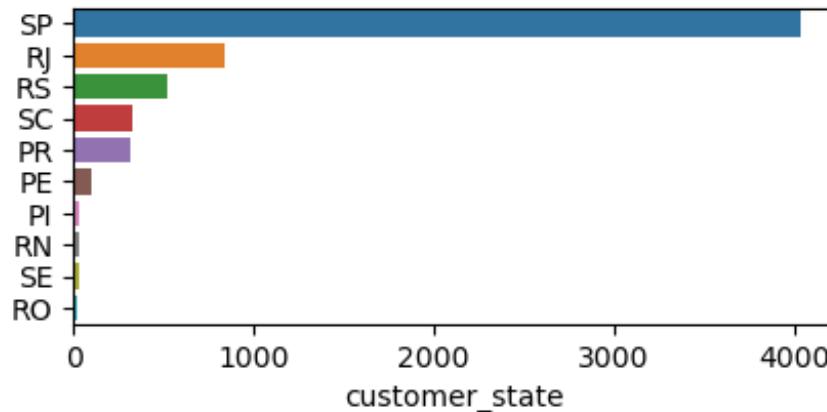
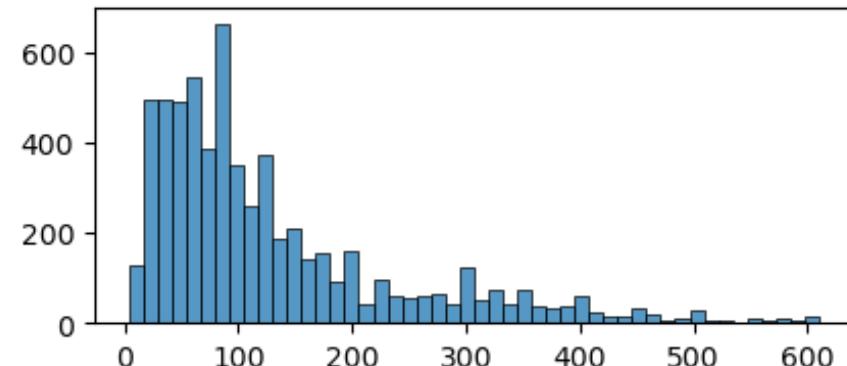
Clientèle aisée cosmopolite.

Dépense moyenne de 132\$ (+30% par rapport à moyenne de tous les groupes).

Majoritairement de Sao Paulo.

Produits de beauté et jouets.

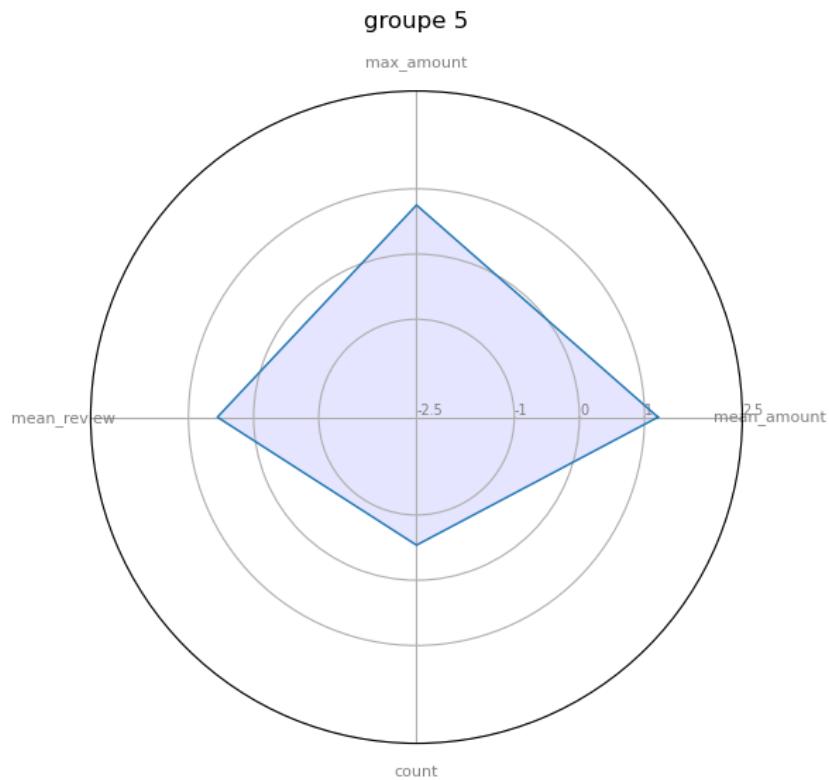
Montant achats



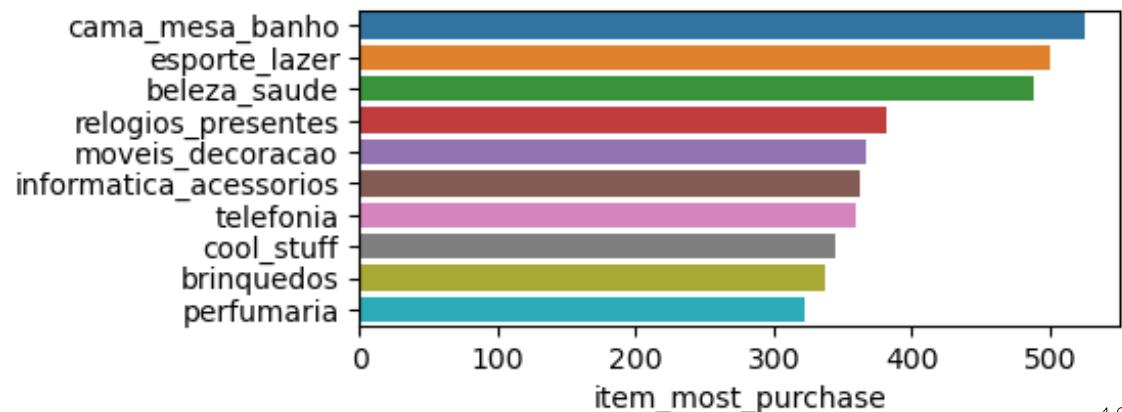
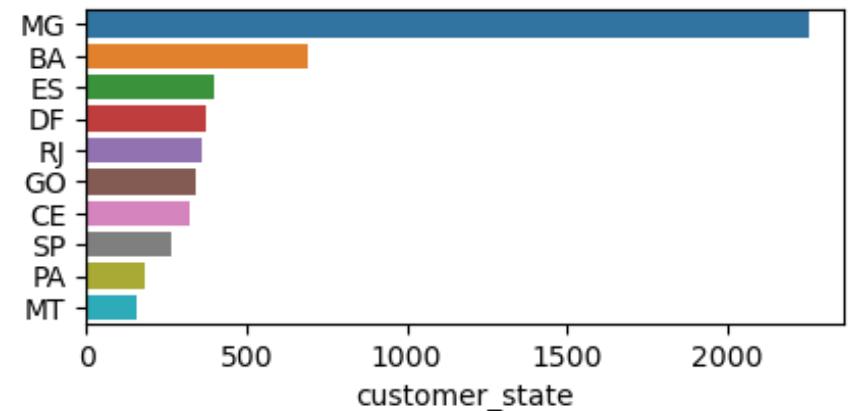
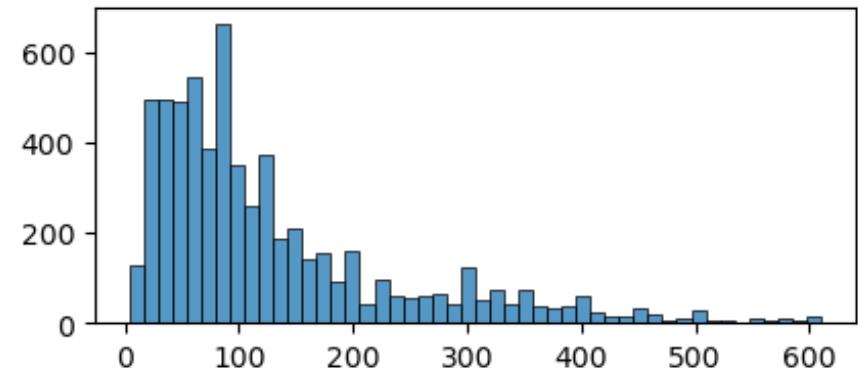
GROUPE 5

V. ANALYSE DES GROUPES

PARTIE 2



Clientèle aisée conservatrice.
Dépense moyenne de 126\$.
Majoritairement dans le Minas Gerais
Sports, loisirs et objets religieux.



Conclusion

Notre segmentation en 6 groupes a permis d'identifier des profils différents.

Cependant, le faible recul que nous avons (1 an et demi) rend les frontières entre ces groupes perméables.

Avec du temps, la singularité de chaque groupes devrait augmenter ce qui permettra de faire des recommandations plus précises.

