

Segmentation clients d'un site e-commerce





Sommaire

PARTIE 1 – PROBLÉMATIQUE MÉTIER

PARTIE 2 – LES DONNÉES

PARTIE 3 – EXPLORATION

PARTIE 4 – MODÉLISATION

PARTIE 5 – FRÉQUENCE DE MAINTENANCE

CONCLUSION

PARTIE 1 - PROBLÉMATIQUE MÉTIER

PROBLÉMATIQUE MÉTIER

- Proposer une segmentation des clients :
 - Facilement interprétable : Clusters et profils client associés clairs
 - Facilement exploitable : Communication ciblée pertinente
 - Tout en s'adaptant aux données :
 - Peu de données
 - Peu de clients récurrents
 - Déterminer une fréquence de maintenance pertinente :
 - Segmentation toujours performante
- vs
- Limitation des coûts

PARTIE 2 - LES DONNÉES

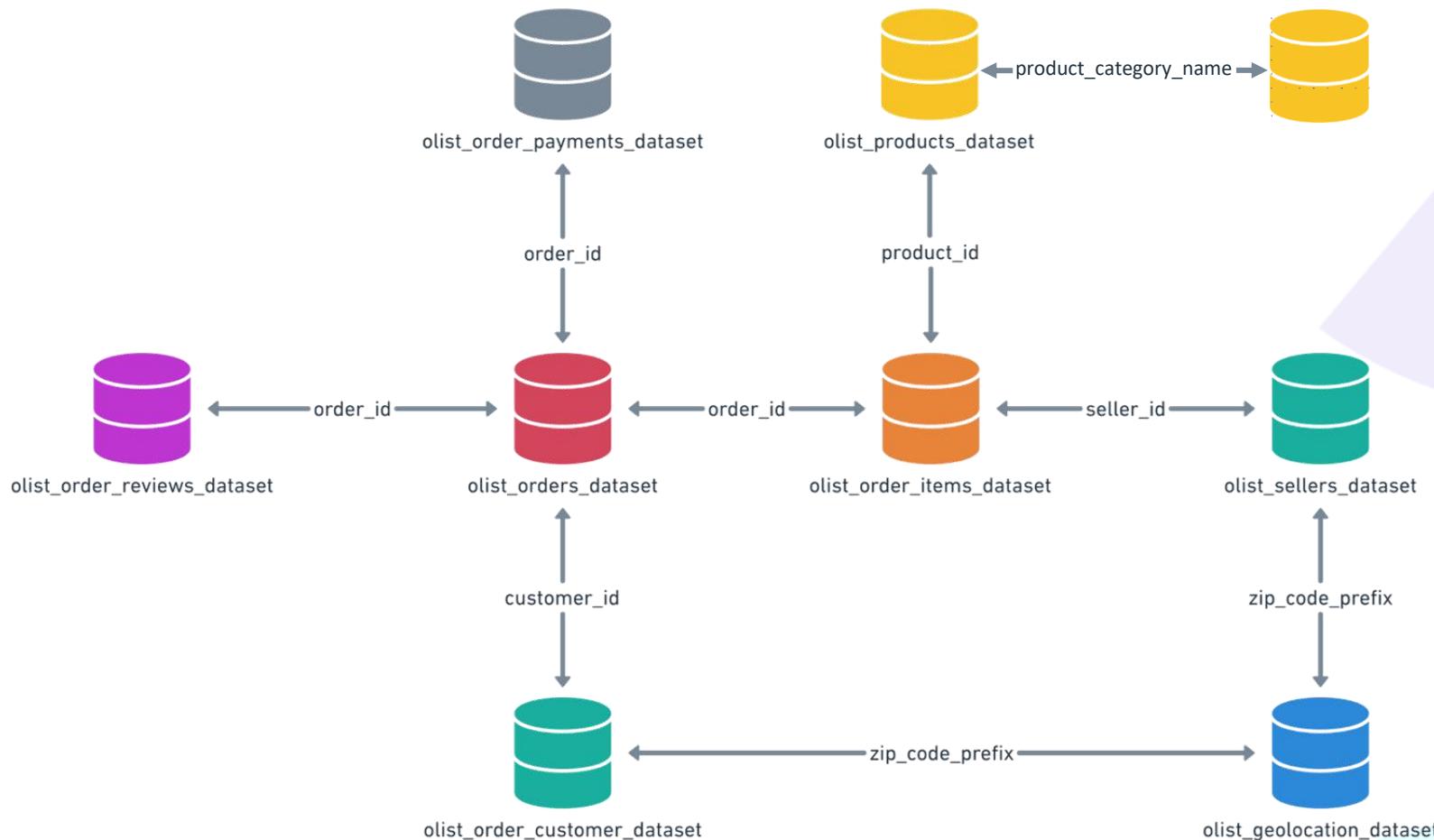
DESCRIPTION DU JEU DE DONNÉES

- Plusieurs jeux de données :

orders.csv	Majoritairement informations temporelles sur les commandes
customers.csv	Clients et leur localisation
geolocation.csv	Codes postaux et leurs coordonnées GPS
items.csv	Articles achetés dans chaque commande et leurs prix
sellers.csv	Vendeurs et leur localisation
products.csv	Produits vendus par la plateforme
translations.csv	Traduction brésilien / anglais de la catégorie des produits
payments.csv	Informations sur les paiements (type, paiement en plusieurs fois, etc.)
reviews.csv	Commentaires et notes clients associés à chaque commande

DESCRIPTION DU JEU DE DONNÉES

- Des jeux de données liés par des **identifiants** :



PREMIER CLEANING

- Gestion de la mémoire

	rawImports	lowMemory	difference
customers	31060010	23537862	-24 %
geolocs	153182803	23905202	-84 %
orders	61833750	26801247	-56 %
items	41342714	19168754	-53 %
payments	18679981	12522692	-32 %
reviews	44823214	31777025	-29 %
products	7124794	6007145	-15 %
sellers	691007	429576	-37 %
translations	10604	14954	41 %

- Problèmes formatage
`xxx_city` et `xxx_state`
(caractères spéciaux, accents)
- Suppression de certaines colonnes (commentaires, etc.)
- `product_category_name` - catégories manquantes

OBTENIR UN DATASET UNIQUE

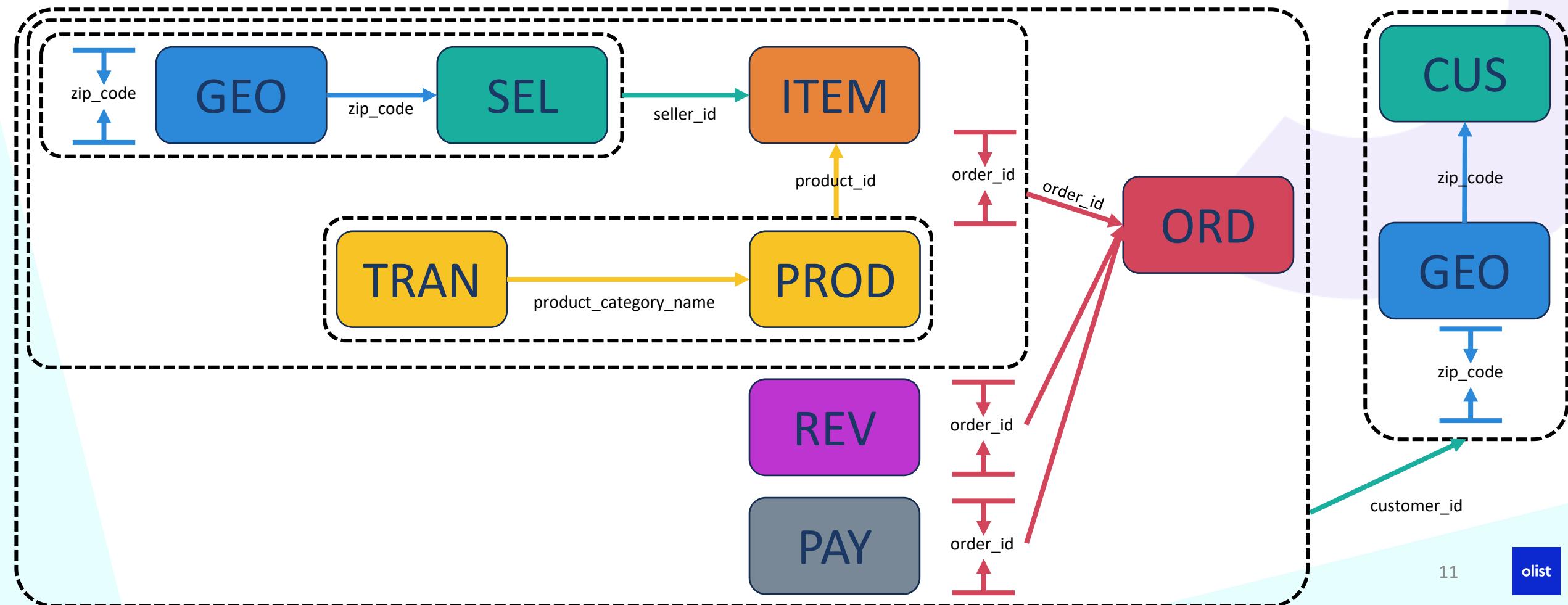
- Doublons sur les ids

	Customers	Geolocs	Orders	Items	Payments	Reviews	Products	Sellers
customer_id	0.0		0.0					
customer_unique_id	3345.0							
customer_zip_code_prefix	84447.0							
geolocation_zip_code_prefix		981148.0						
order_id			0.0	13984.0	4446.0	551.0		
order_item_id				112629.0				
product_id				79699.0			0.0	
seller_id				109555.0				0.0
review_id						814.0		
seller_zip_code_prefix								849.0

- Correspondances

- Jointures généreront des valeurs manquantes
- ex : REV → ORD : 1% de NaN supplémentaire
- ex : ITEMS → ORD : idem

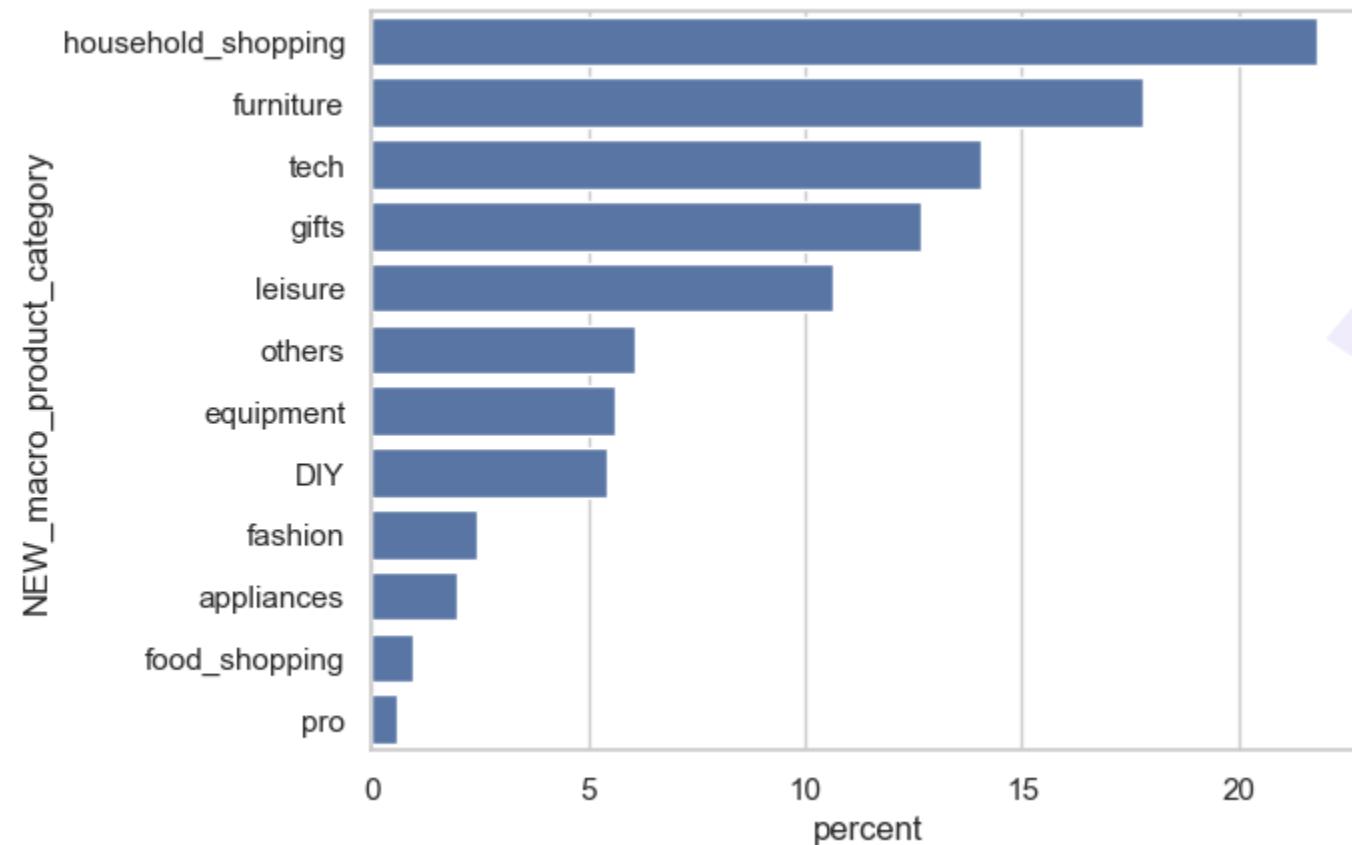
OBTENIR UN DATASET UNIQUE



PARTIE 3 - EXPLORATION

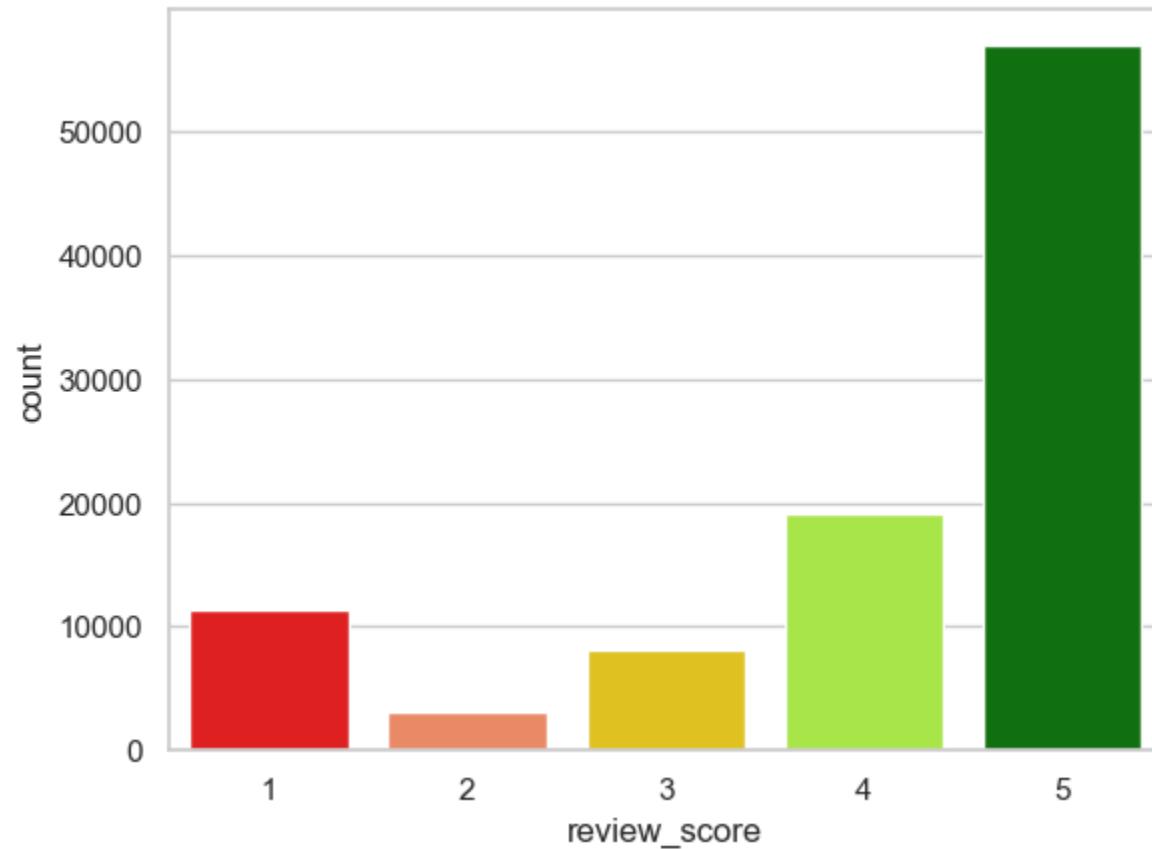
CATÉGORIES DE PRODUIT

- 74 catégories → 12 macro-catégories



NOTES CLIENTS

- Des retours clients positifs :



COLONNES TEMPORELLES

- Quel ordre ?

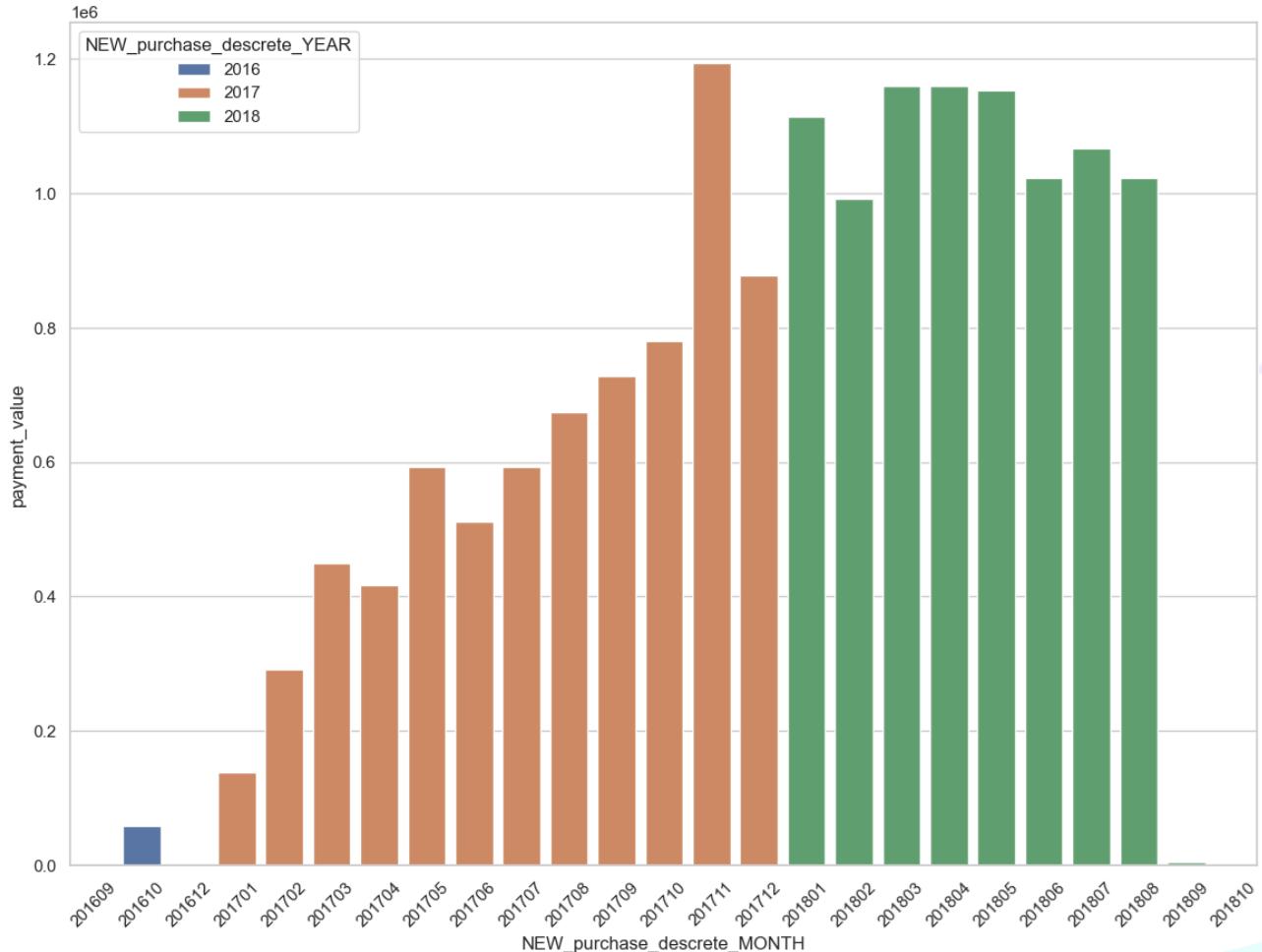
- `order_purchase_timestamp` : l'achat
- `order_approved_at` : l'approbation de la commande
- `order_delivered_carrier_date` : la prise en charge du transporteur
- `shipping_limit_date` : la limite d'envoi (pour le vendeur)
- `order_delivered_customer_date` : la réception de la commande chez le client
- `review_creation_date` : la création du commentaire
- `review_answer_timestamp` : la réponse au commentaire
- `order_estimated_delivery_date` : l'estimation de la réception de la commande

- Des données sur quelle période ?

- Du 4 septembre 2016 au 9 avril 2020
- 3 ans et 7 mois

COLONNES TEMPORELLES

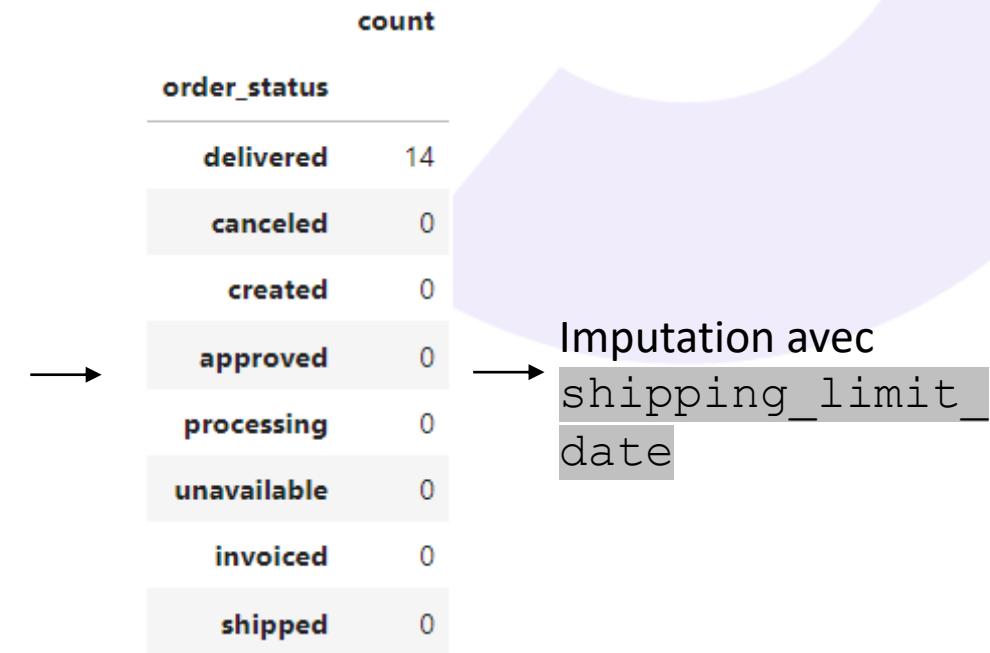
- Discréteriser le mois d'achat et l'année d'achat



COLONNES TEMPORELLES

- Imputations :
 - Grâce à la colonne `order_status`
 - Explorations de différents cas :

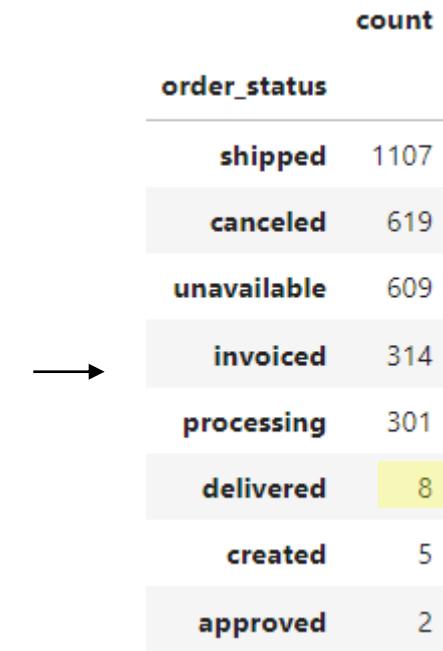
	<code>order_approved_at</code>	<code>order_delivered_carrier_date</code>	<code>order_delivered_customer_date</code>
0	Not NaN	Not NaN	Not NaN
1	Not NaN	Not NaN	NaN
2	Not NaN	NaN	Not NaN
3	NaN	Not NaN	Not NaN
4	Not NaN	NaN	NaN
5	NaN	Not NaN	NaN
6	NaN	NaN	Not NaN



COLONNES TEMPORELLES

- Imputations :
 - Grâce à la colonne `order_status`
 - Explorations de différents cas :

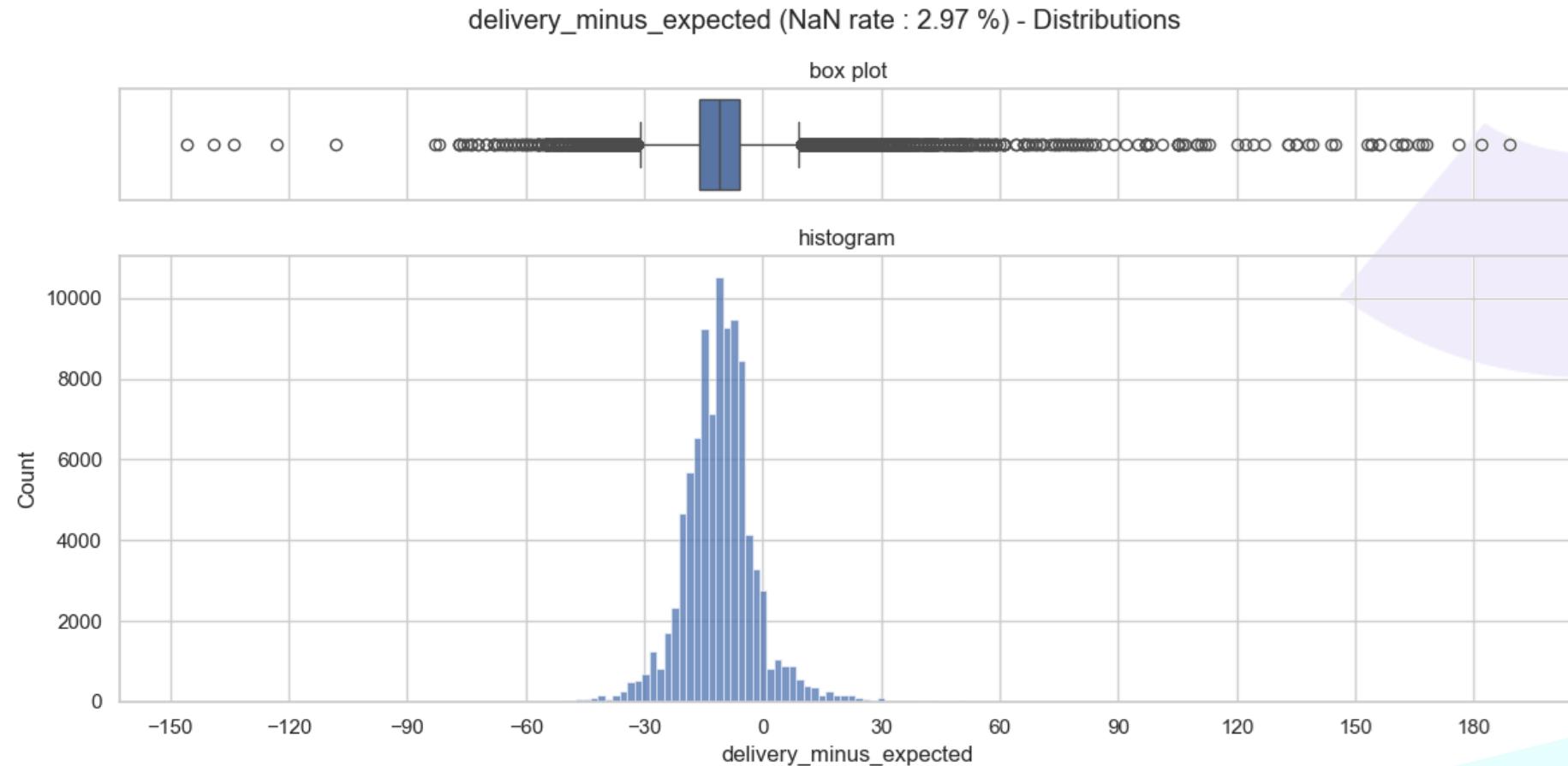
	<code>order_approved_at</code>	<code>order_delivered_carrier_date</code>	<code>order_delivered_customer_date</code>
0	Not NaN	Not NaN	Not NaN
1	Not NaN	Not NaN	NaN
2	Not NaN	NaN	Not NaN
3	NaN	Not NaN	Not NaN
4	Not NaN	NaN	NaN
5	NaN	Not NaN	NaN
6	NaN	NaN	Not NaN



→ Imputation avec
→ durée médiane de
livraison

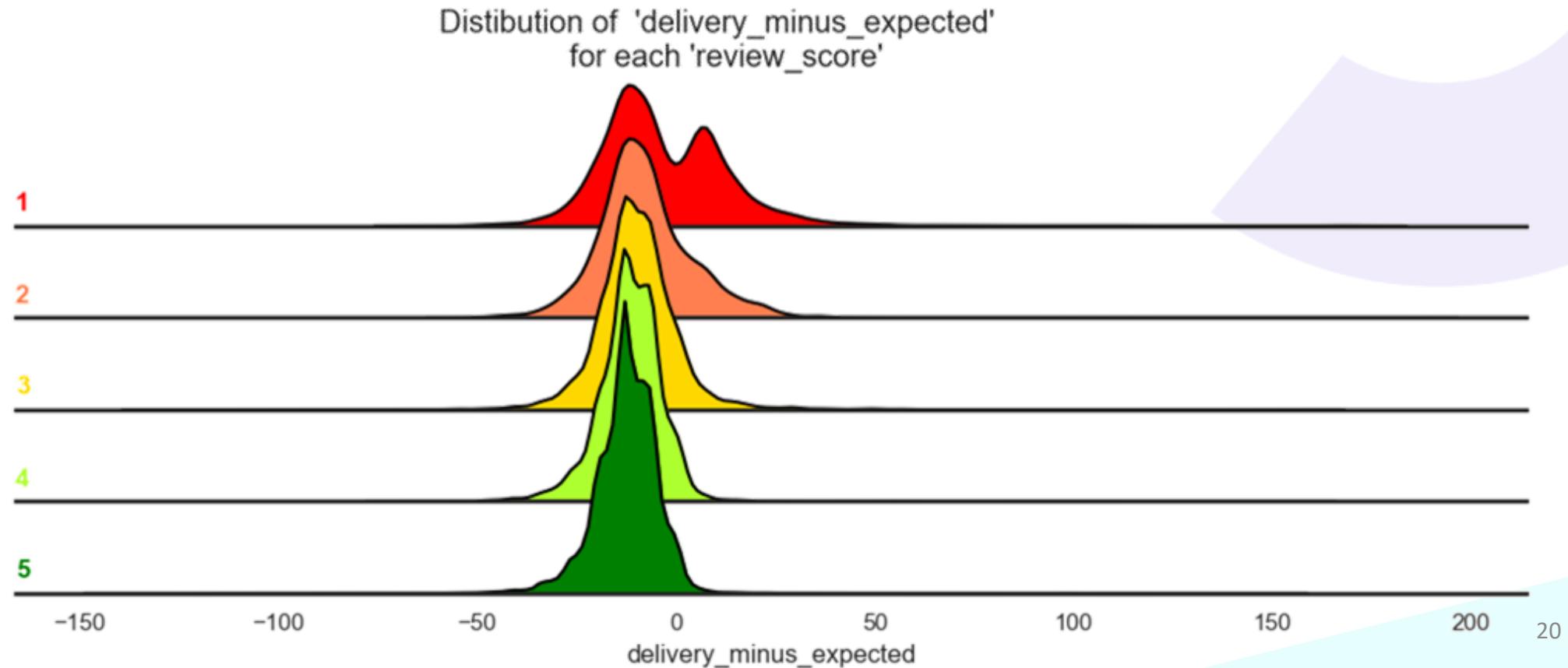
COLONNES TEMPORELLES

- Feature engineering : Des retards de livraison ?



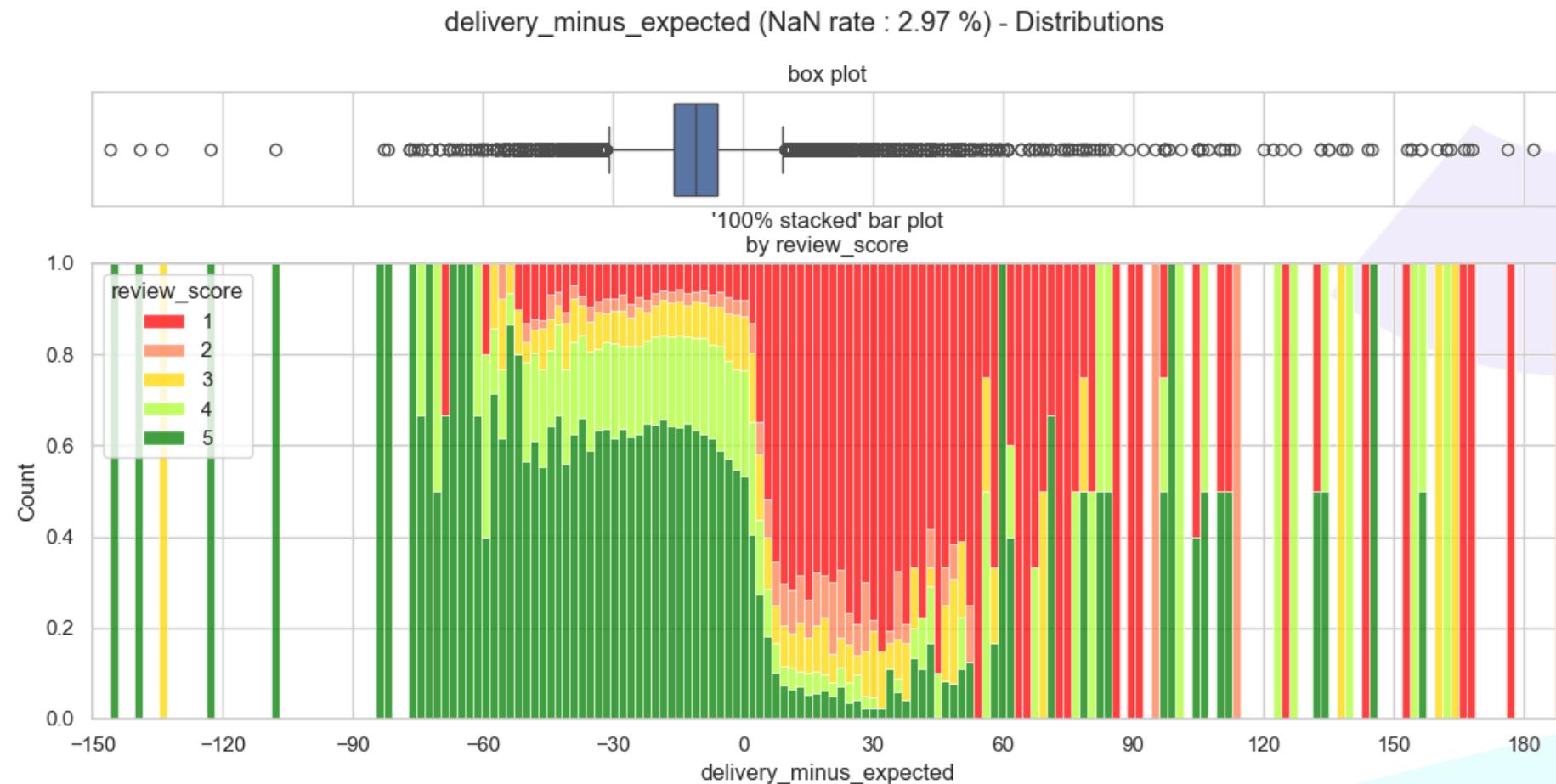
COLONNES TEMPORELLES

- DELAY : Lien avec `review_score`



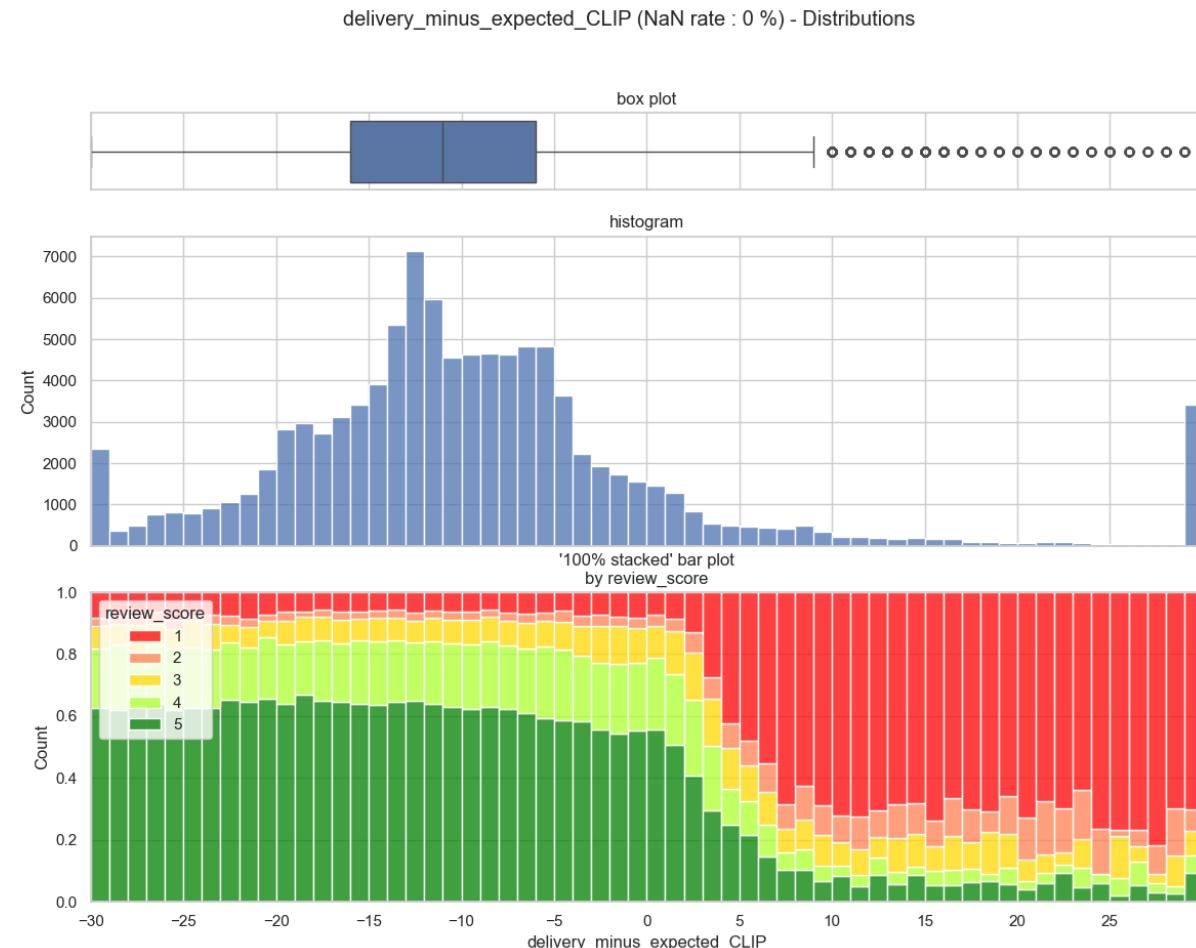
COLONNES TEMPORELLES

- **DELAY** : Lien avec review_score



COLONNES TEMPORELLES

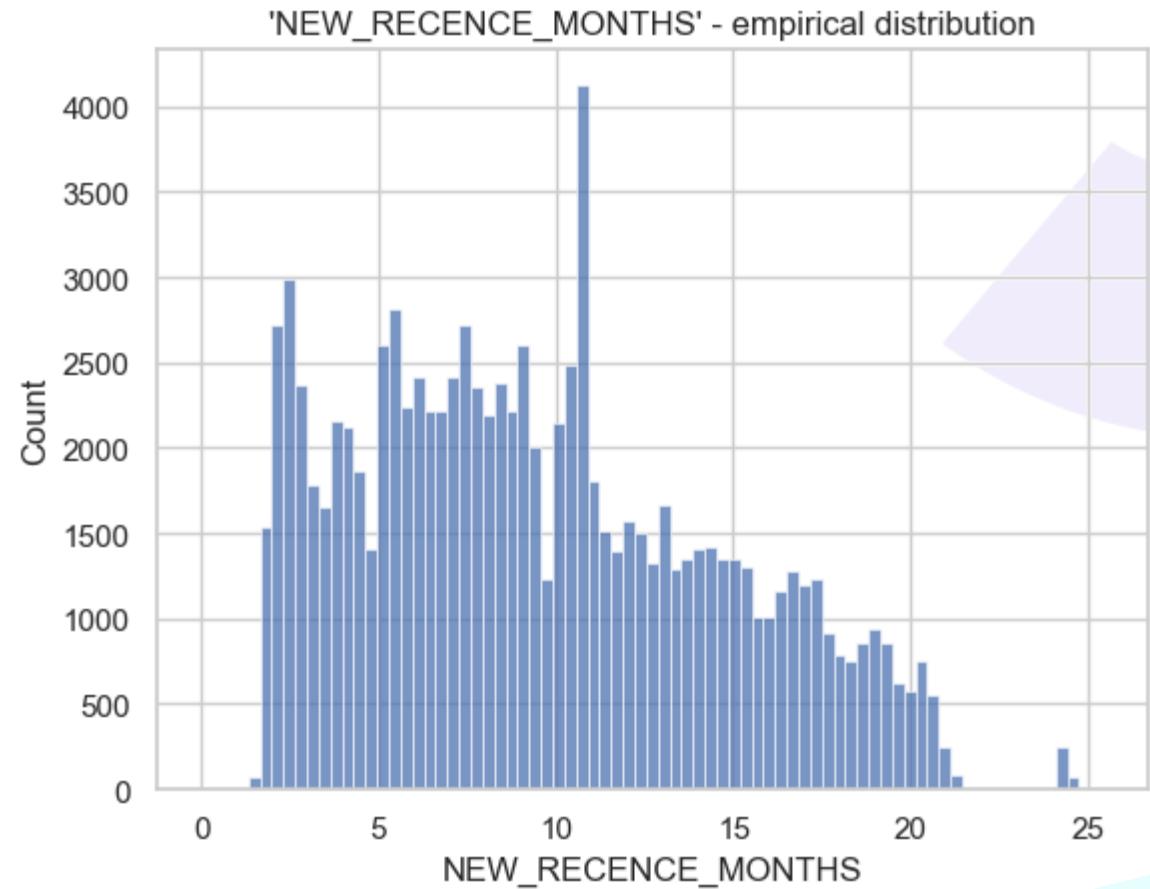
- **DELAY** : bornage [- 1 m ; 1 m] + imputations à « 1 mois »



COLONNES TEMPORELLES

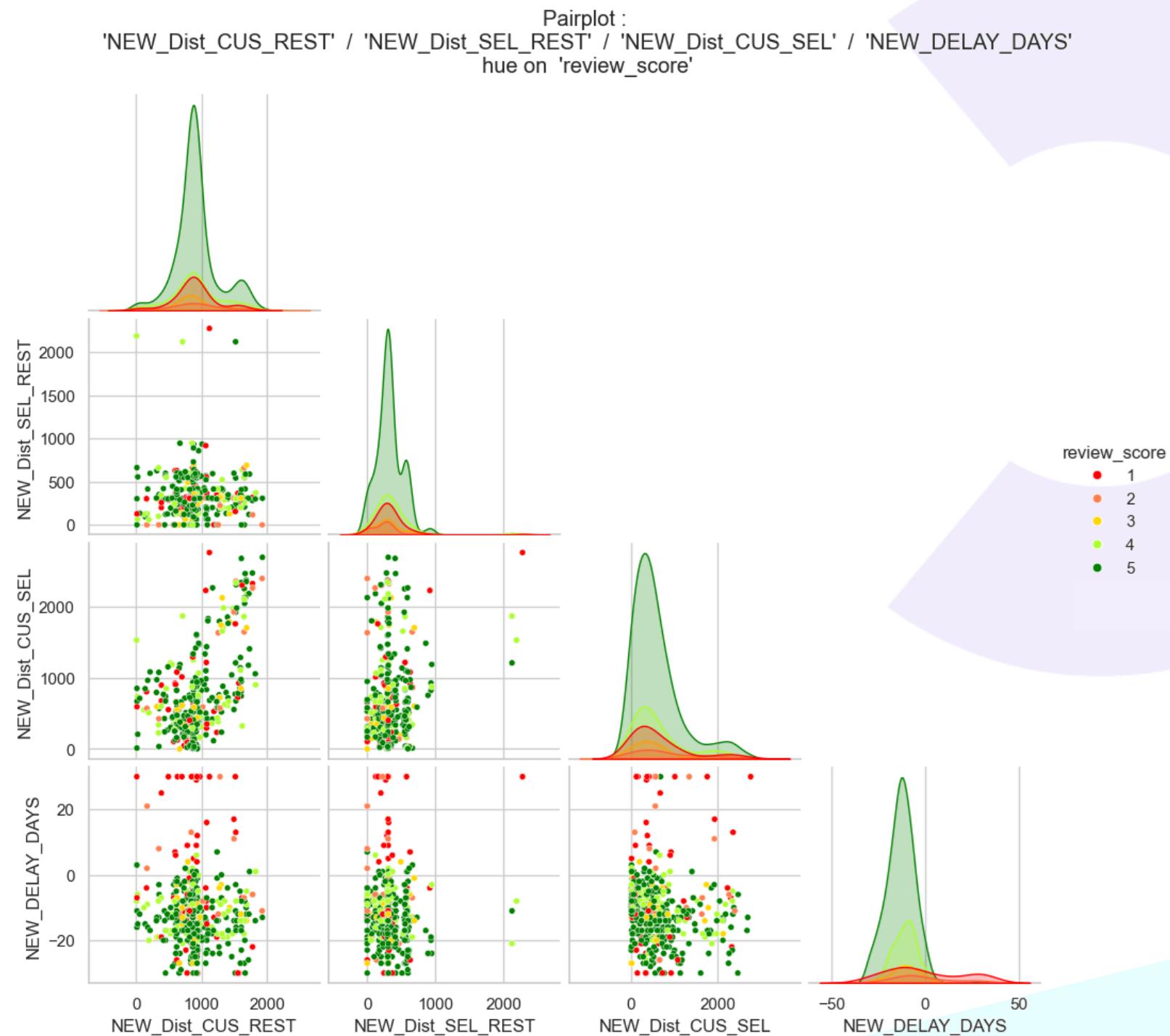
- Feature engineering : Récense

	NEW_RECENCE_MONTHS
%NaN	0.0%
count	99441.0
mean	9.544724409448818
std	5.052215407288729
min	0.0
25%	5.46
50%	8.94
75%	13.15
max	25.41
dtype	float64



DISTANCES

- Grâce aux coord. GPS, calcul de distances :



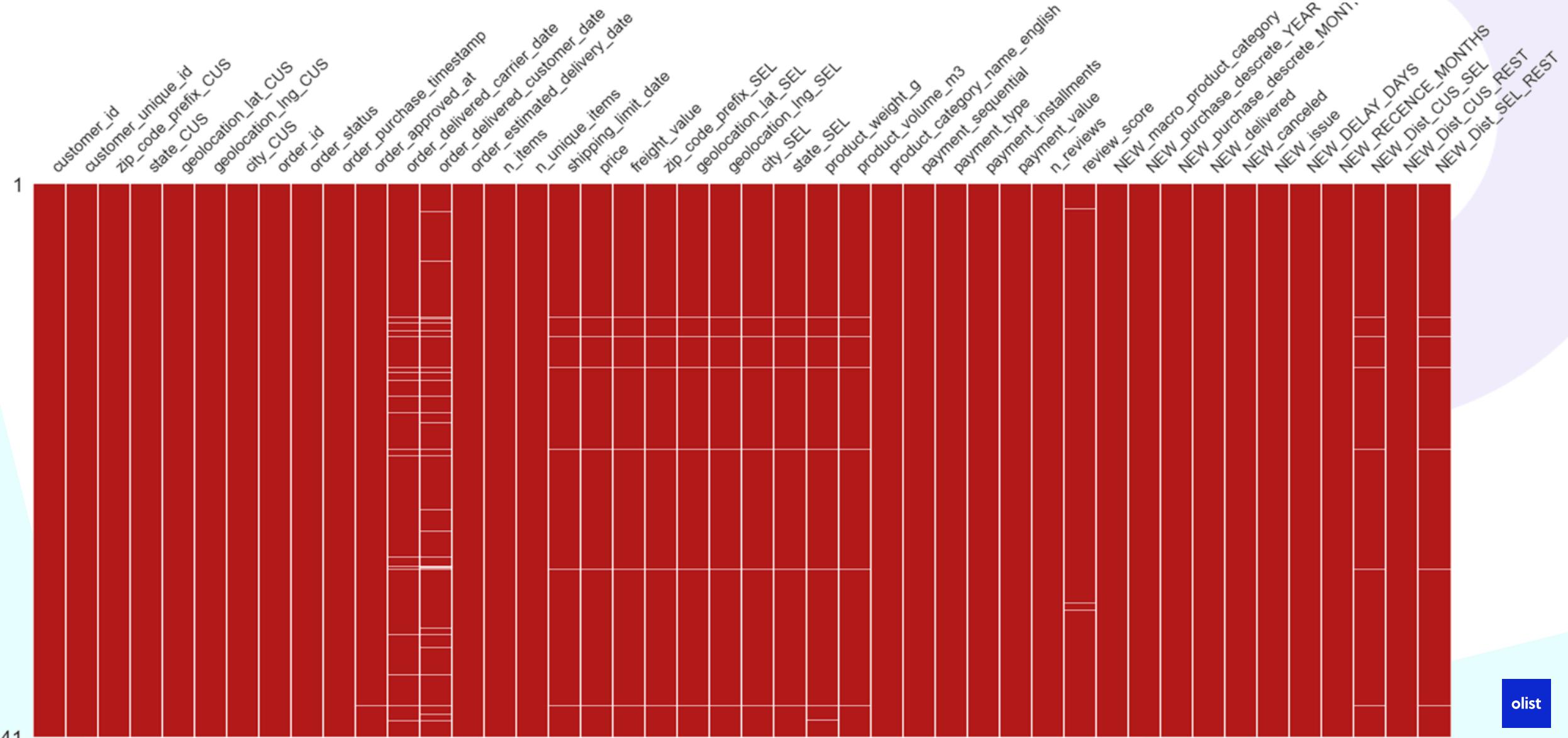
CORRELATIONS

Kendall

Quelques liens :

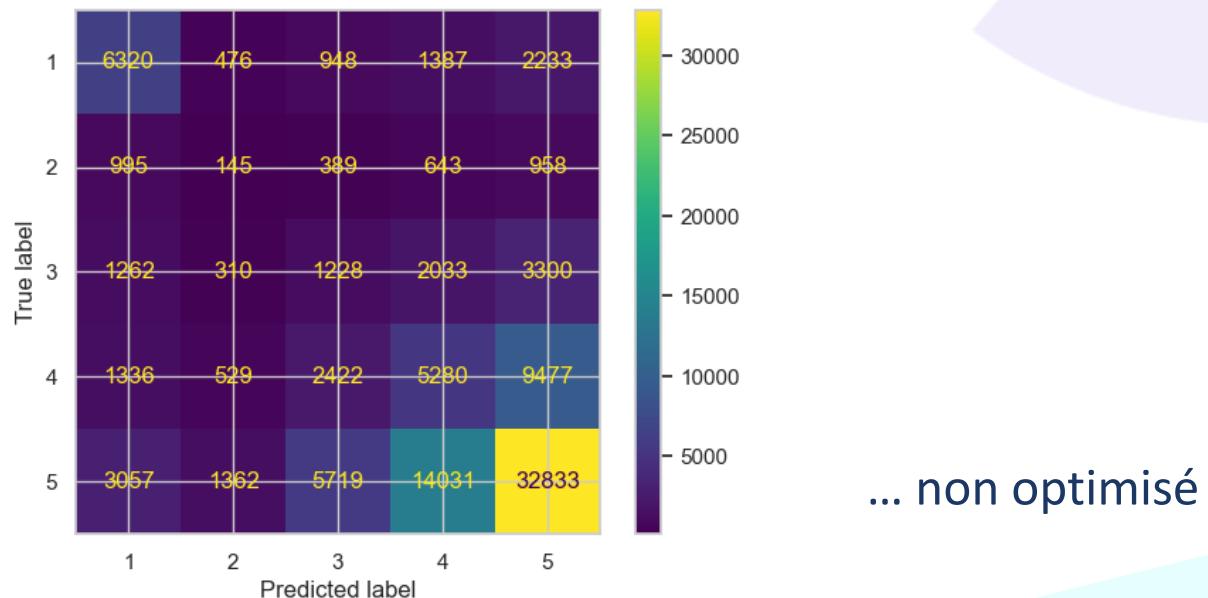
- Colonnes temporelles ou affiliées
 - Poids / volume et frais de livraison
 - Distance client/vendeur et frais de livraison
 - Etc.

VALEURS MANQUANTES

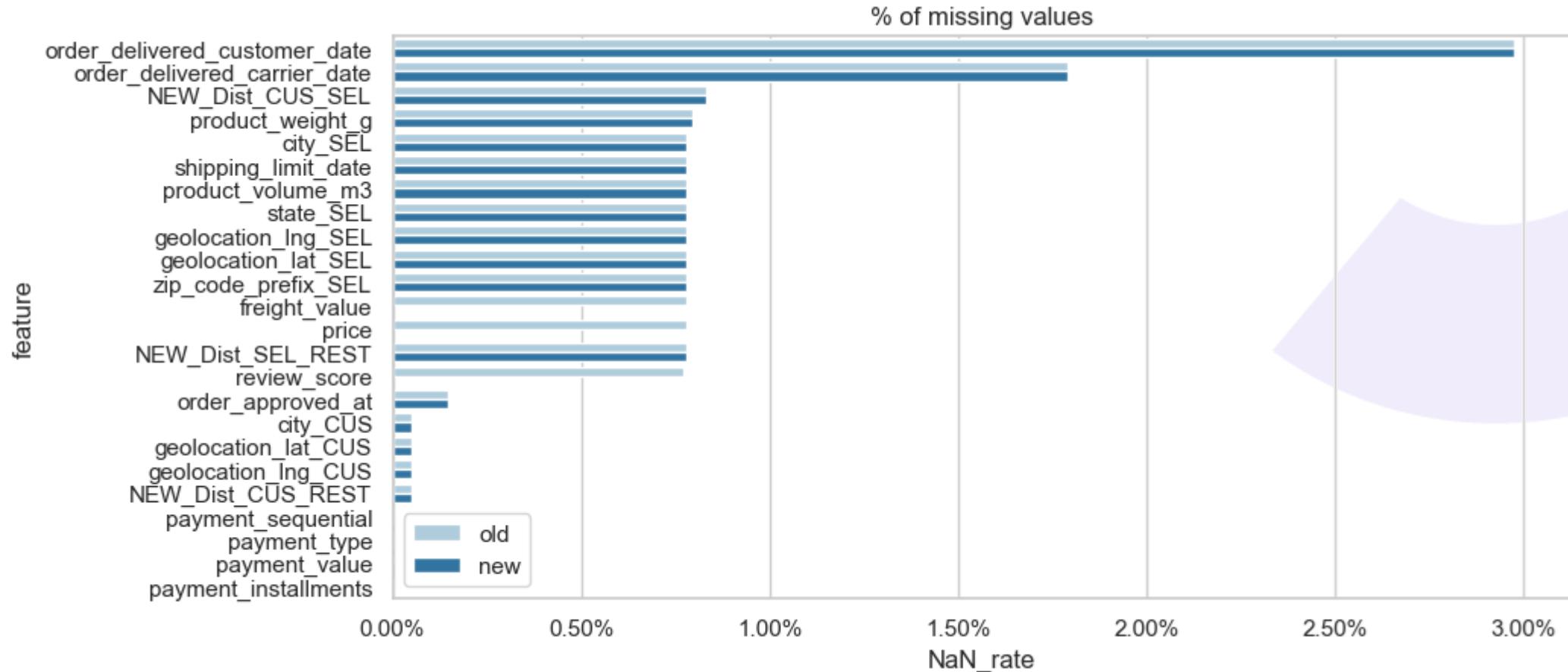


VALEURS MANQUANTES - IMPUTATIONS

- `freight_value`, `price` et `payment_value` en utilisant :
$$(prix + livraison) * Narticles = paiement.$$
- `review_score` en utilisant un modèle de machine learning :



VALEURS MANQUANTES - BILAN



PARTIE 4 - MODÉLISATION

QUELLES FEATURES ?

Sélectionner des features intéressantes pour le service marketing

Tout en vérifiant que ces features ressortent en termes de variance

→ ACP

Dans un premier temps :

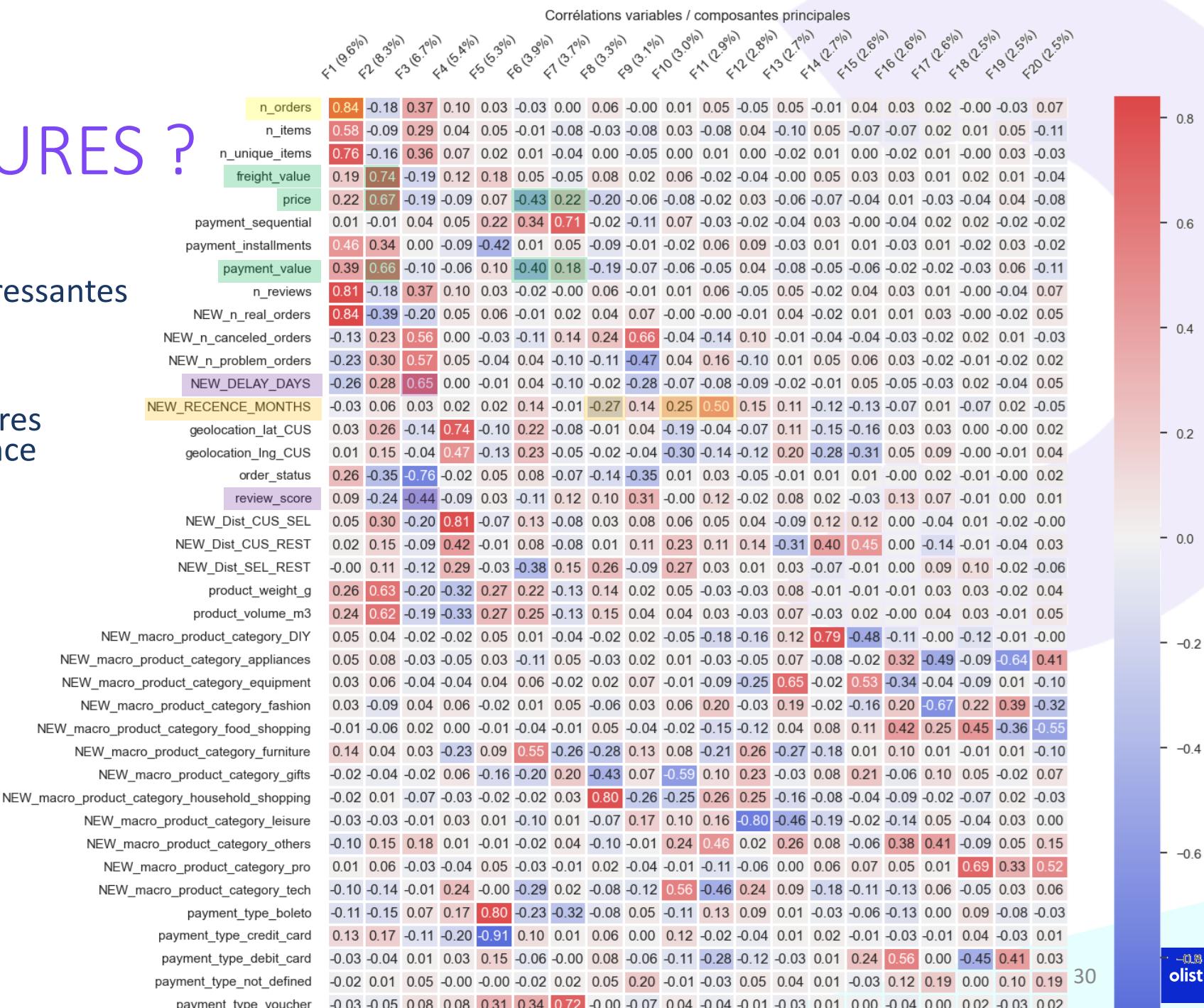
RECENSE

FREQUENCY

MONETARY

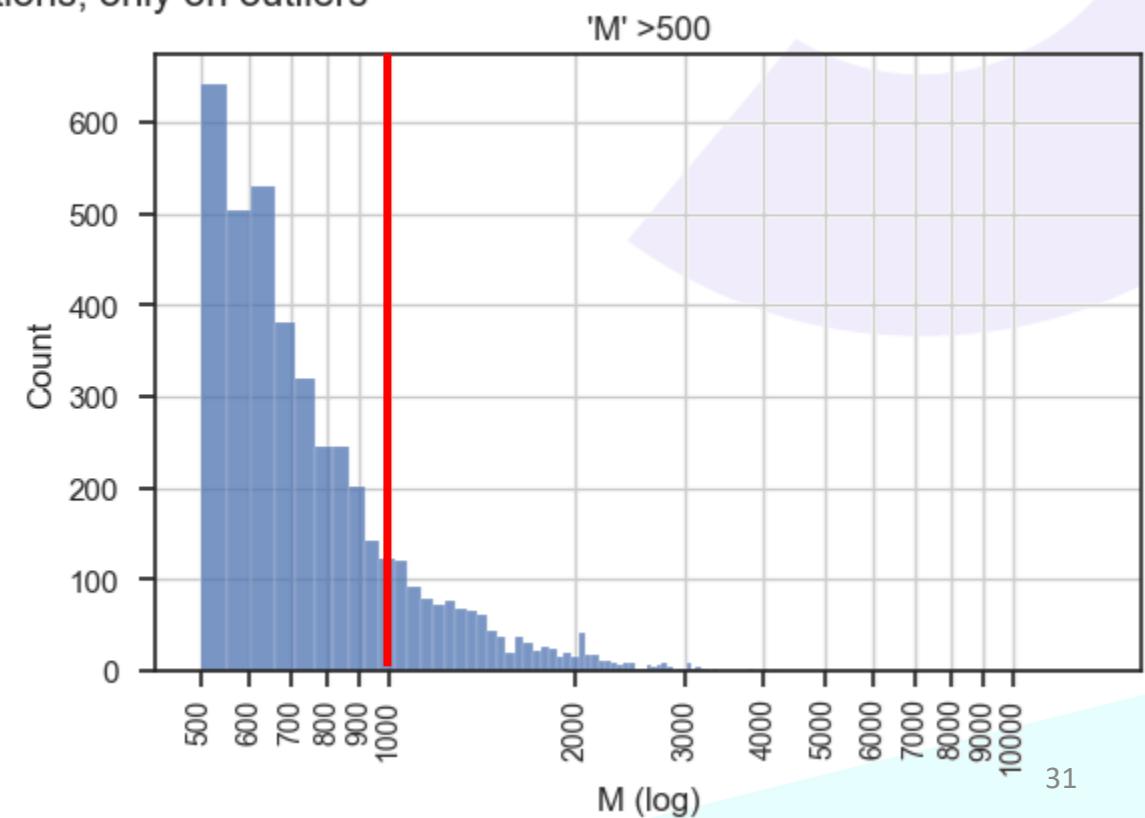
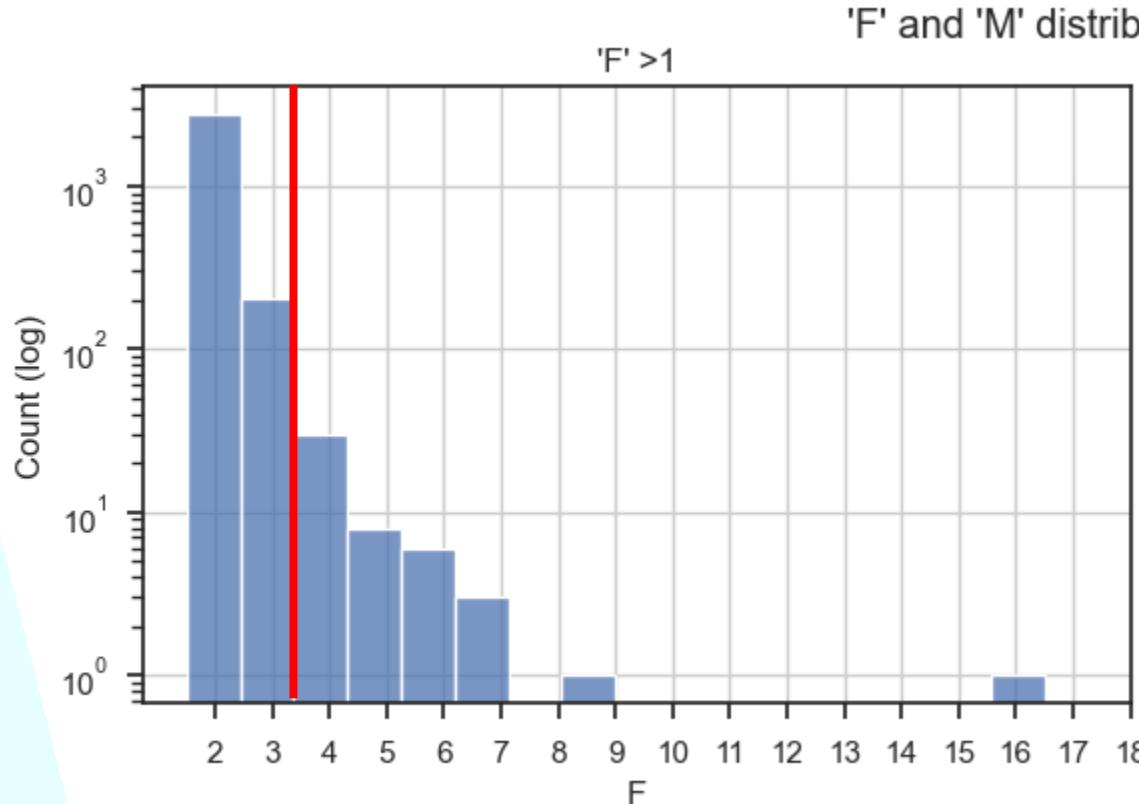
Peut être intéressant :

DELAY, REVIEW



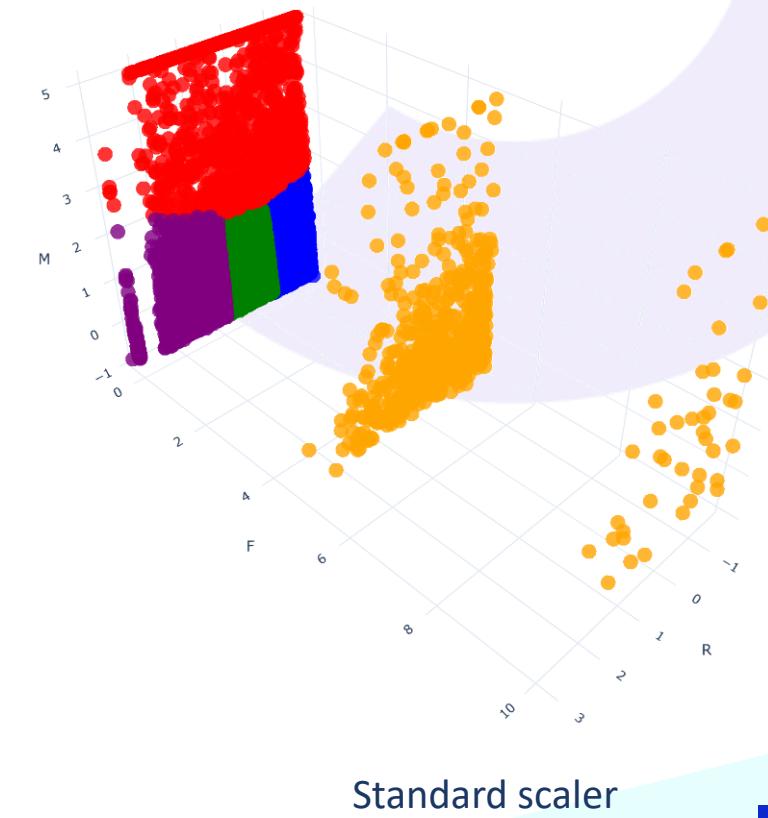
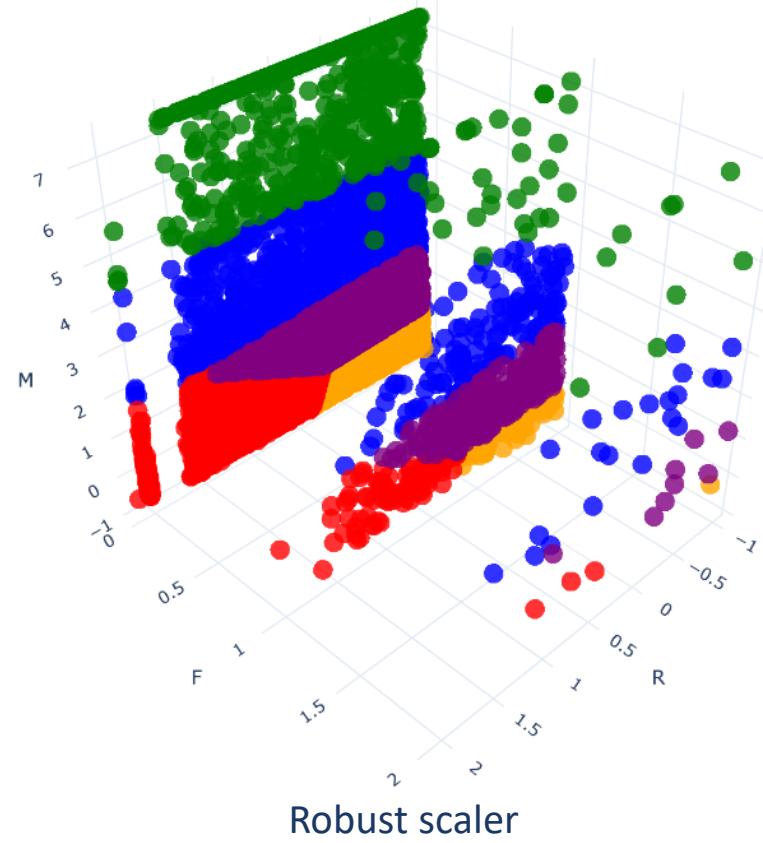
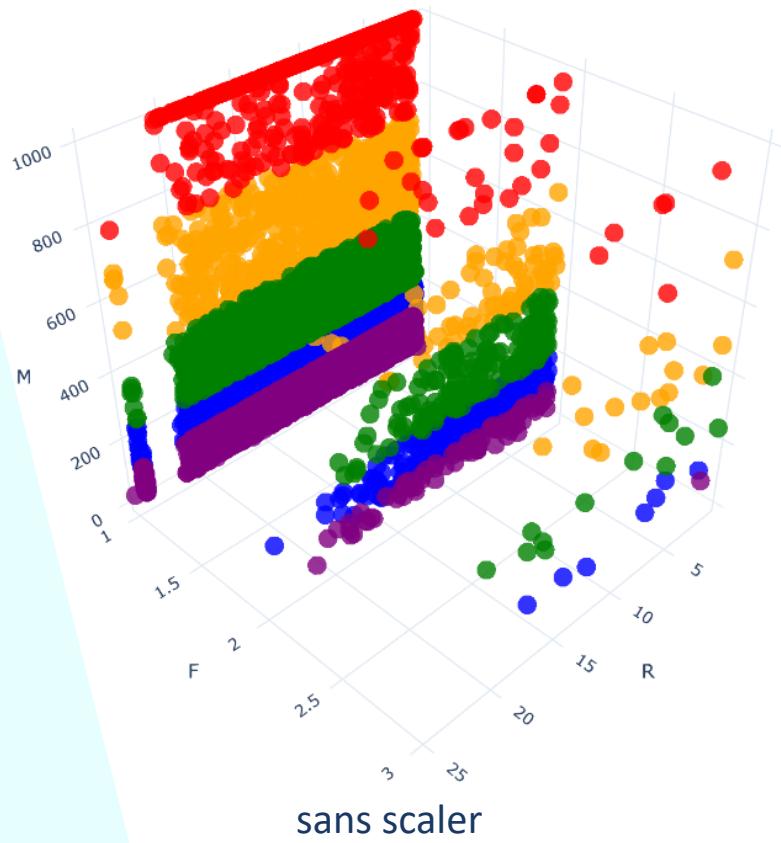
RFM – gestion des outliers

- Très peu de clients avec plus de 3 commandes ou dépensant plus de 1000 réals :



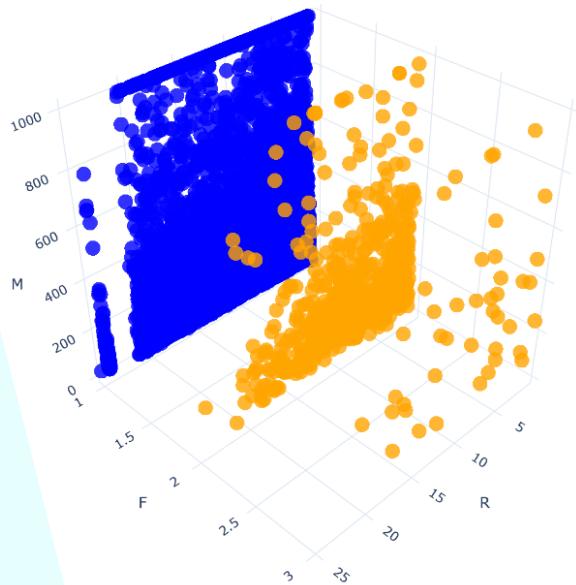
RFM + KMeans

- Tests de différents scalers :

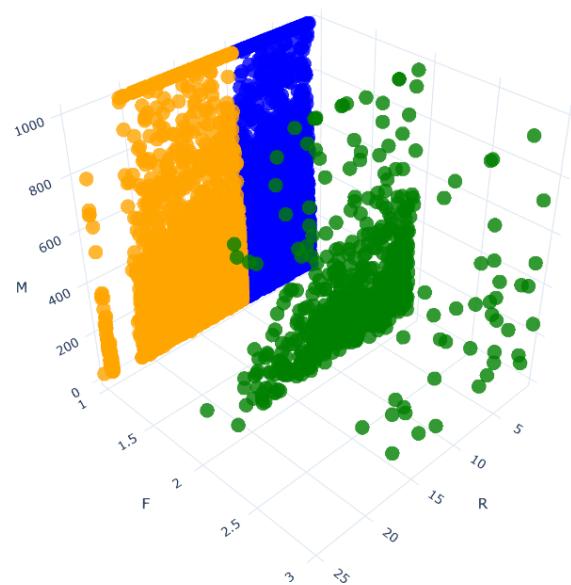


RFM + KMeans

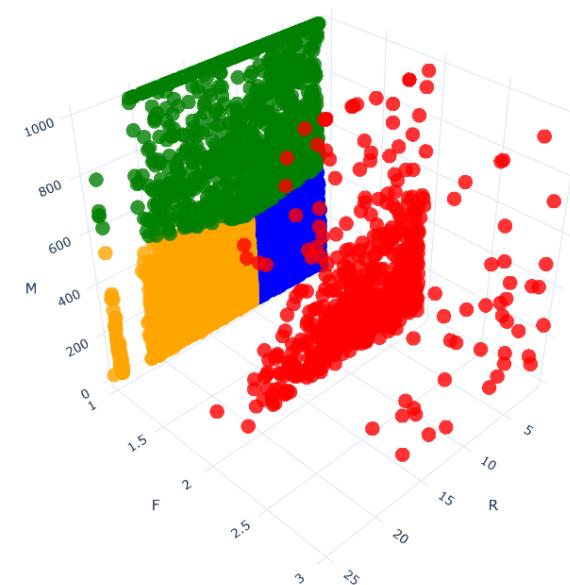
- Tests de différents n_clusters :



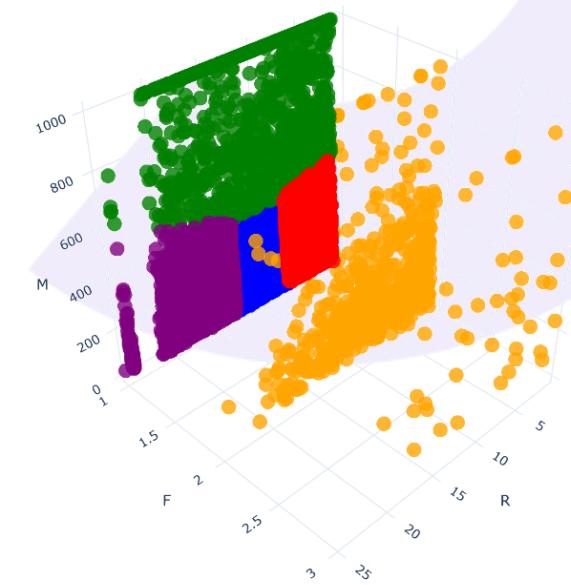
K = 2



K = 3



K = 4

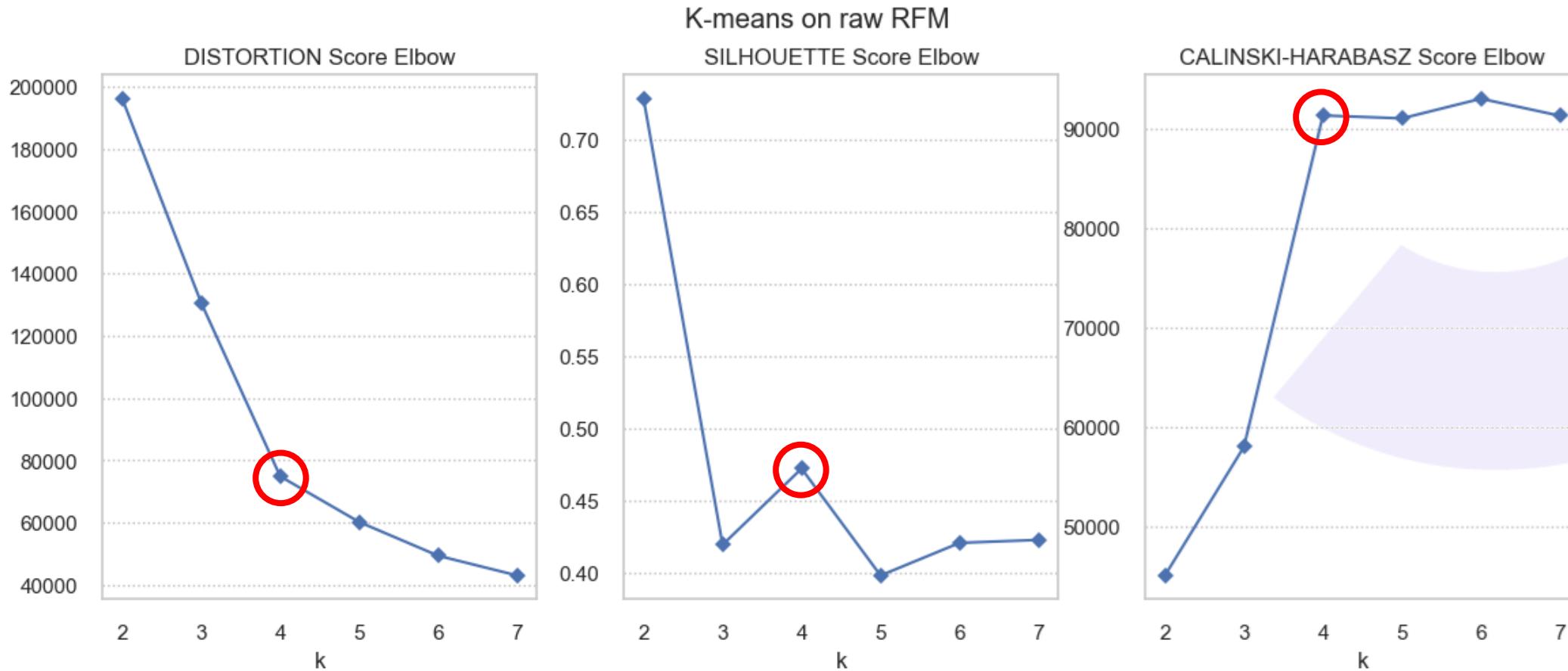


K = 5

RFM + Kmeans – Méthode du coude

- Aide à la décision pour déterminer `n_clusters`
- Basé sur une métrique d'évaluation
- Faire évoluer K jusqu'à ce que la métrique n'évolue plus significativement dans le bon sens
- 3 métriques :
 - Distorsion
 - Coefficient moyen de Silhouette
 - Indice de Calinski - Harabasz

RFM + Kmeans – Méthode du coude



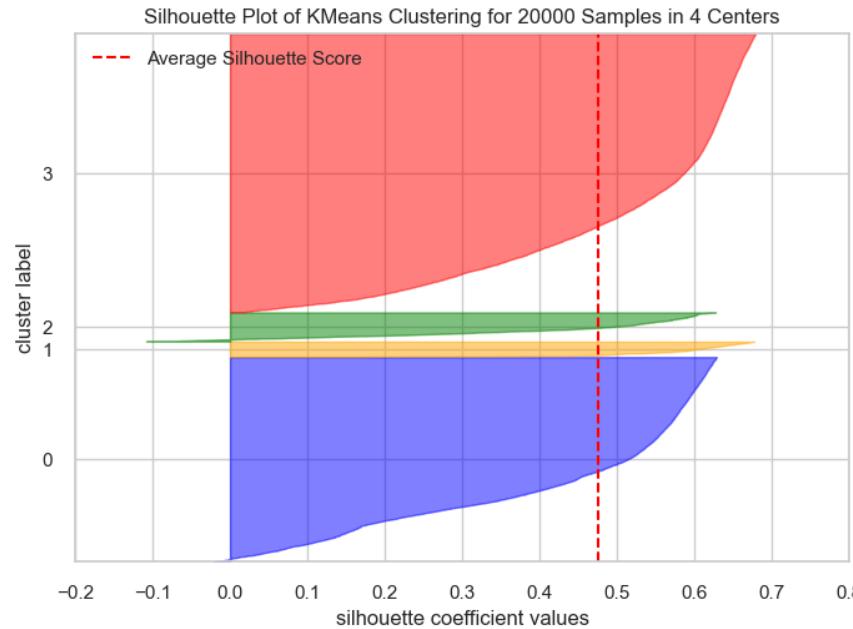
→ $k = 4$

RFM + Kmeans – Évaluer notre clustering

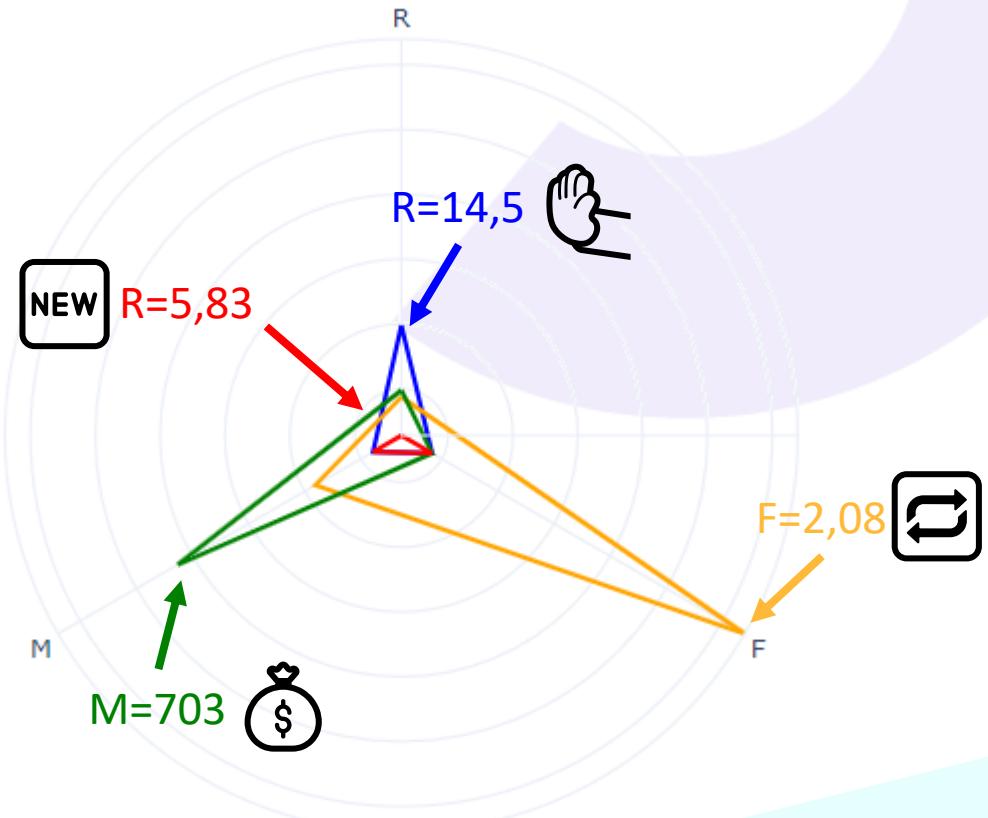
- Métriques

	Silhouette	Calinski-H	Davies-Bouldin
KMeans on RFM	0.47	91287.94	0.68

- Silhouette plot



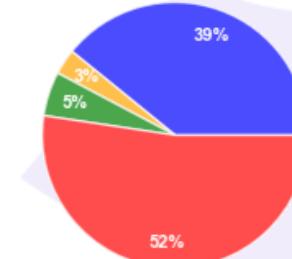
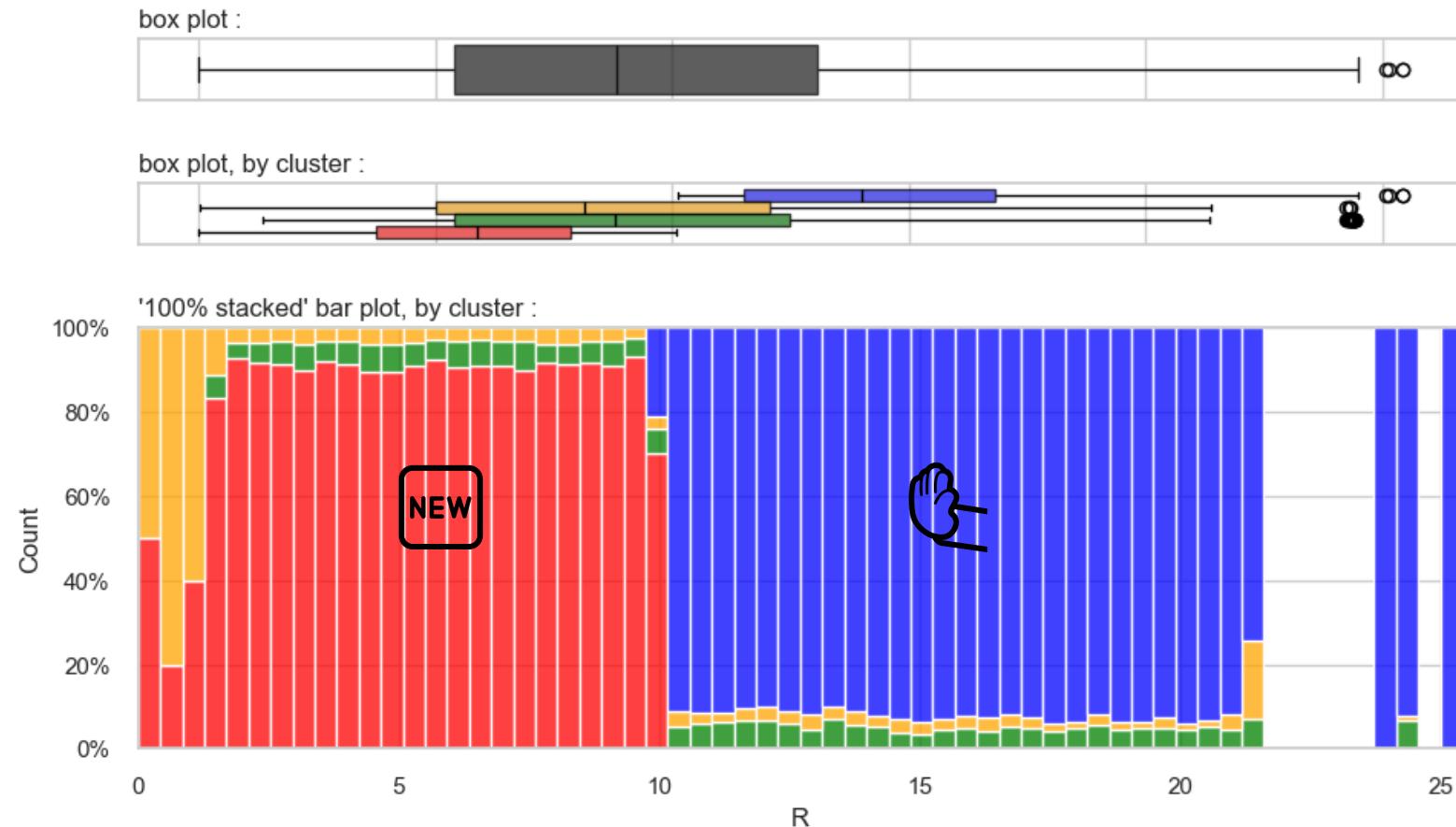
- Radar plot



RFM + Kmeans – Évaluer notre clustering

- Graphique en barres empilées

Kmeans on RFM - feature 'R' distribution, by cluster



RFM + Kmeans – Évaluer notre clustering

- Graphique en barres empilées

Kmeans on RFM - feature 'F' distribution, by cluster

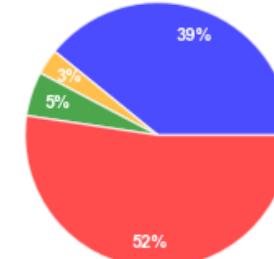
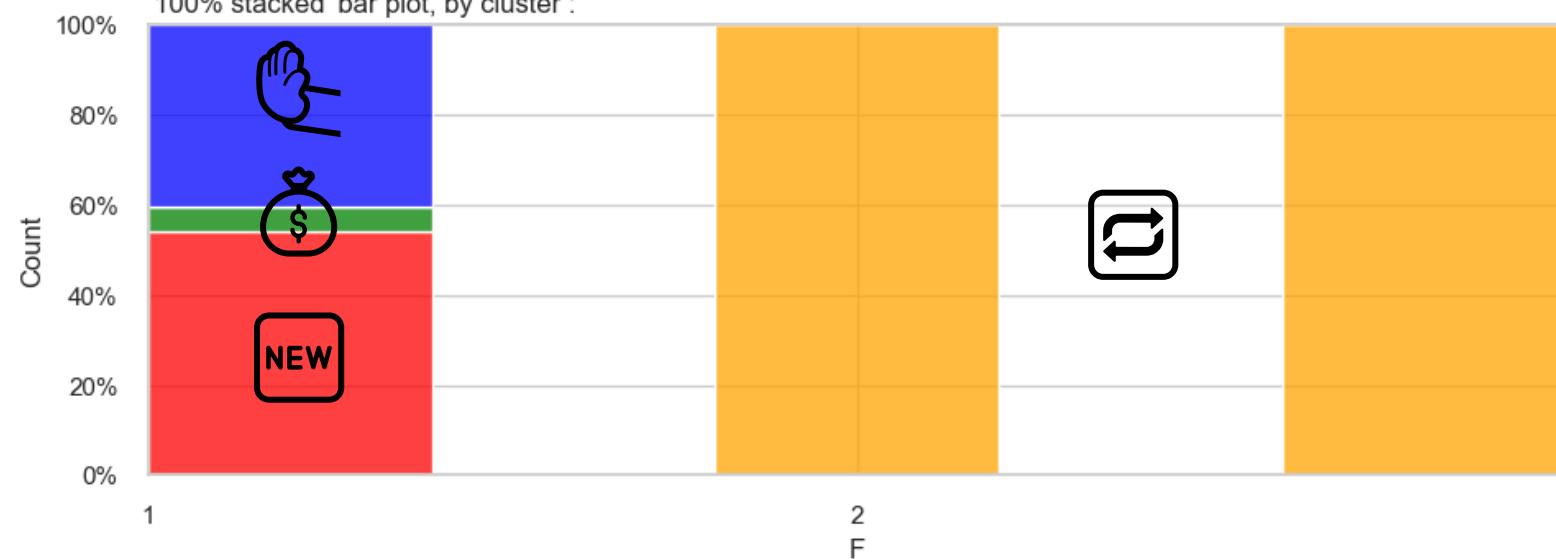
box plot :



box plot, by cluster :



'100% stacked' bar plot, by cluster :

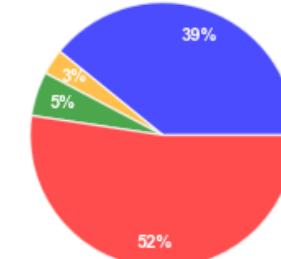
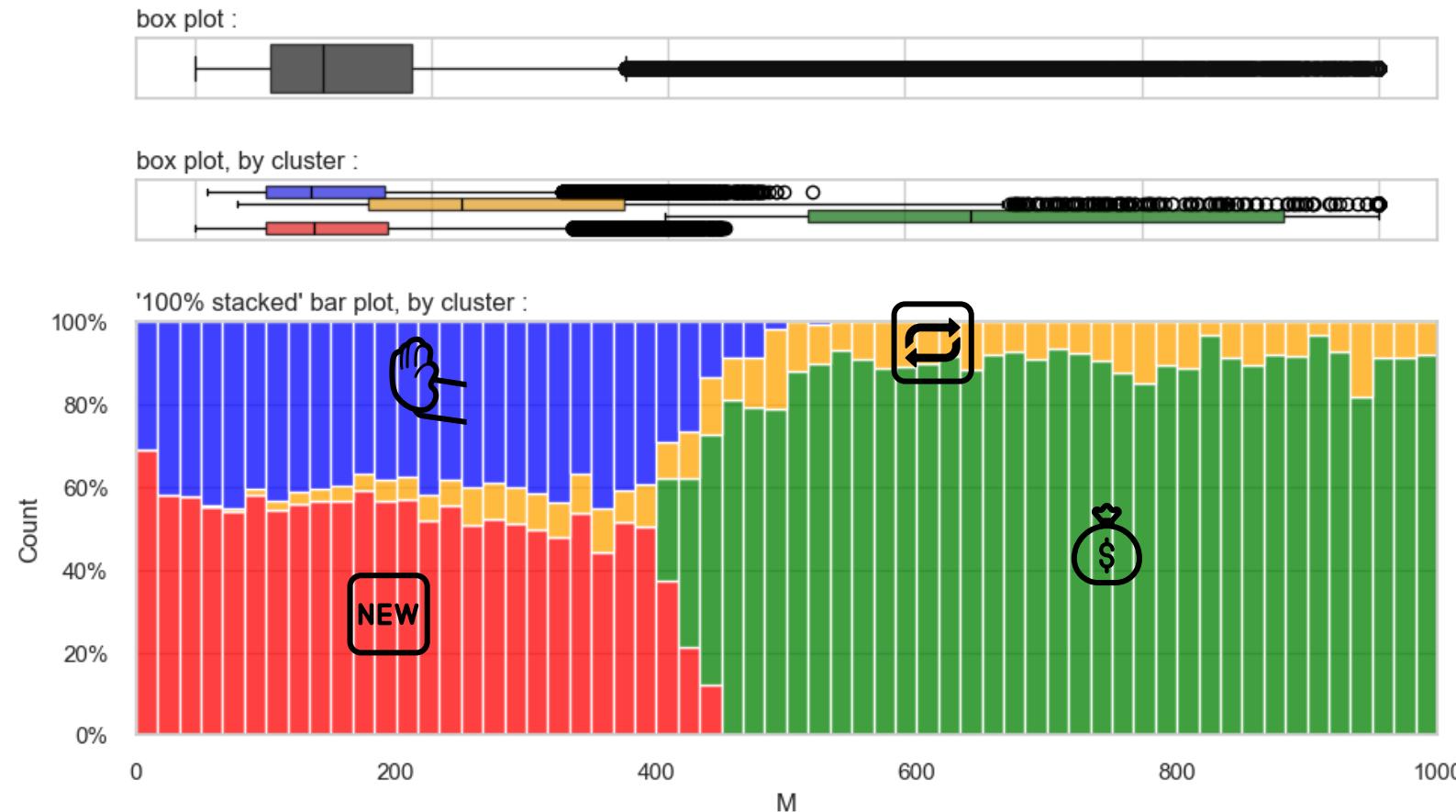


cluster
0
1
2
3

RFM + Kmeans – Évaluer notre clustering

- Graphique en barres empilées

Kmeans on RFM - feature 'M' distribution, by cluster



RFM + Kmeans – Nos clusters

- « dépensiers » 💰
 - Intéressants, il faut gagner leur confiance
- « récurrents » 🔍
 - Les plus intéressants, peuvent devenir des fidèles
- « ex-clients » 🤲
 - Pas vraiment de potentiel, ne pas investir sur eux
- « nouveaux » 🚀
 - Intéressants, agir rapidement pour déclencher une 2nd commande

RFM + Kmeans – Quelles actions ?

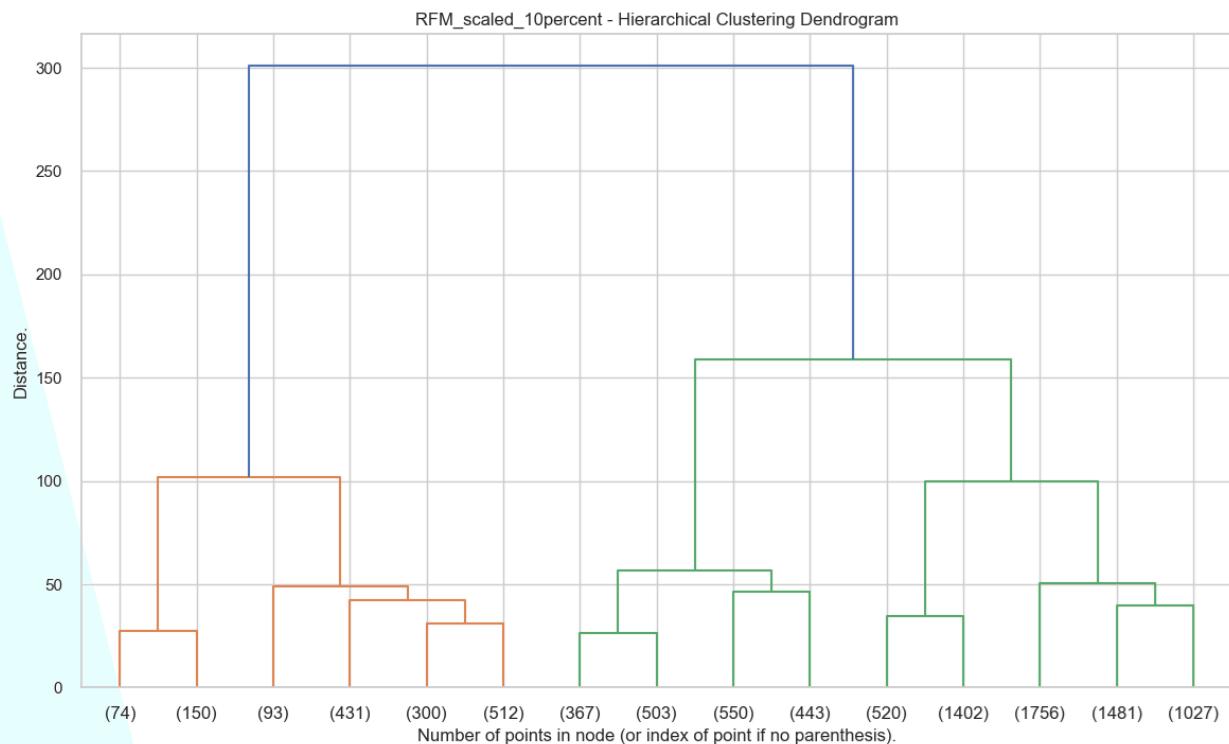
- « dépensiers » 💰
 - (gros) bon d'achat ? (gros) cadeau de bienvenue ? ventes privées articles onéreux ?
- « récurrents » 🔄
 - Recherche catégorie préférée ? Ventes privées ? Dispositif fidélité ?
- « ex-clients » 🖐
 - Pas d'action
- « nouveaux » 🌟
 - Bon d'achat ? Cadeau de bienvenue ?

RFM + Clustering hiérarchique

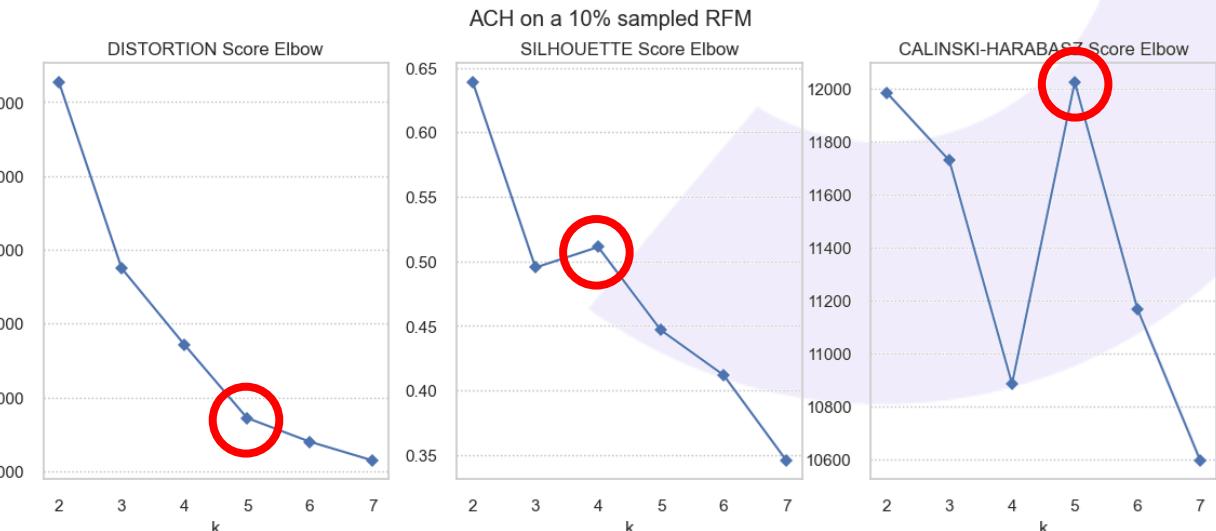
- Difficultés techniques :
 - Matériel informatique à disposition pas assez pourvu en mémoire
 - Temps de calcul trop longs
- Astuces pour tester :
 - Réduire le nombre de points – seulement 10% des données
 - Créer des « points » pertinents grâce à KMeans

RFM + Clustering hiérarchique – n_clusters ?

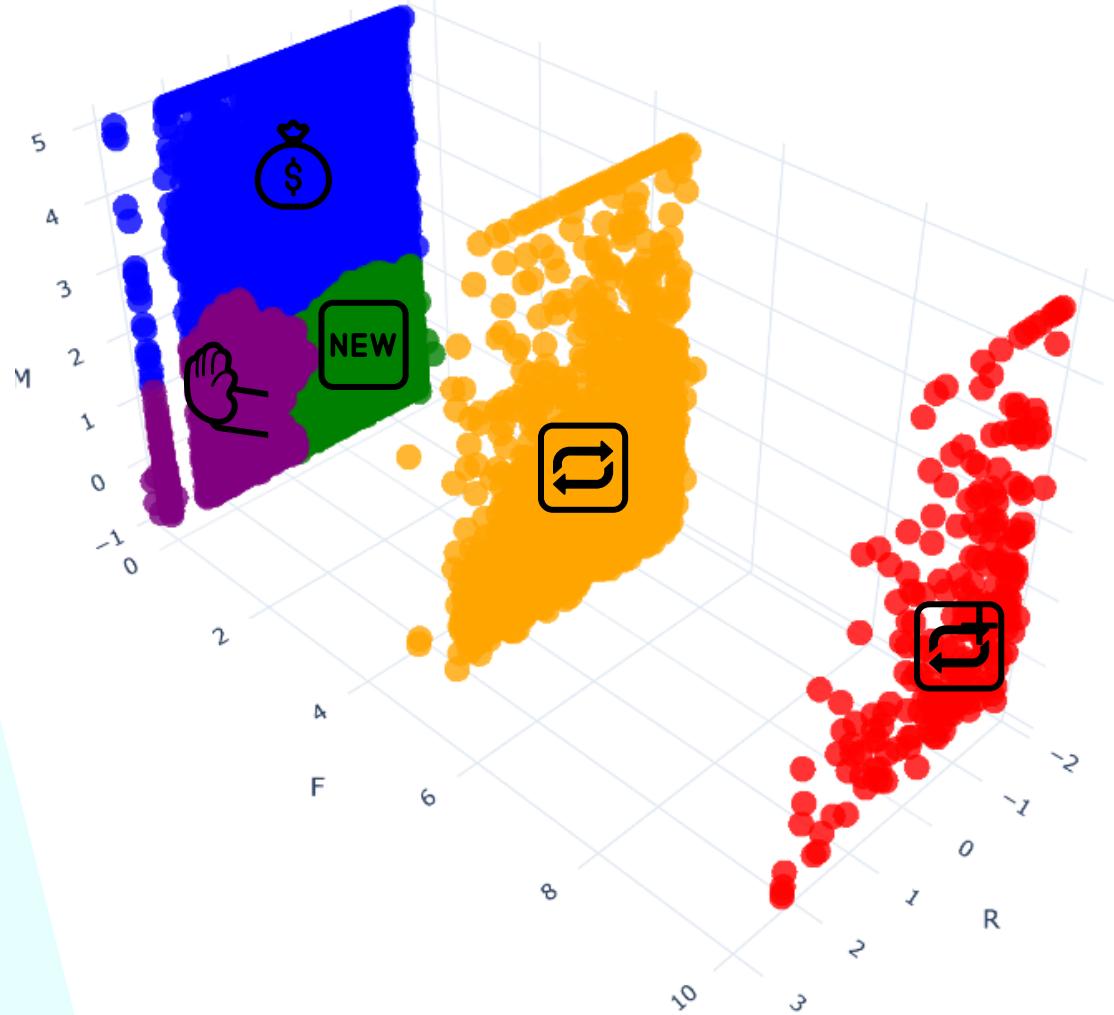
- Dendrogramme :



- myElbowVisualizer :



RFM + Clustering hiérarchique – 3D & métriques



- Se rapproche du clustering KMeans
- Cependant :
 - Limite de « nouveaux » / « ex » semble différente
 - « récurrents » séparés en $F = 2$ et $F \geq 3$
- Métriques :

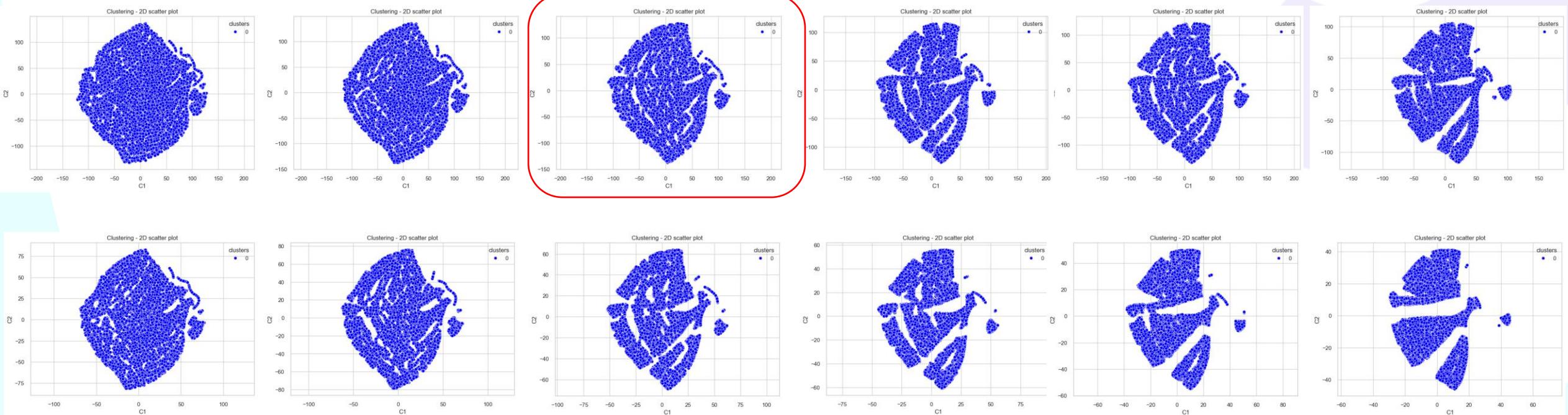
	Silhouette	Calinski-H	Davies-Bouldin
KMeans on RFM	0.47	91287.94	0.68
ACH on RFM 10 percent	0.45	12023.80	0.71

RFM + tSNE – comprendre le manifold ?

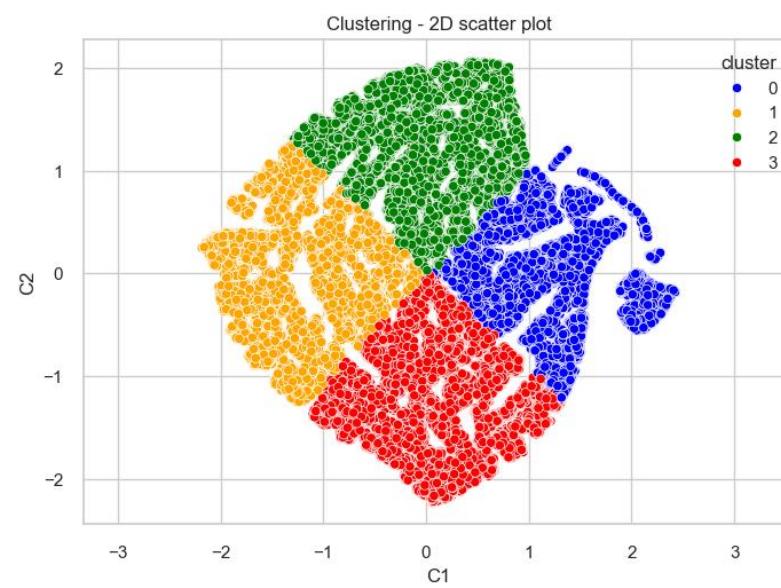
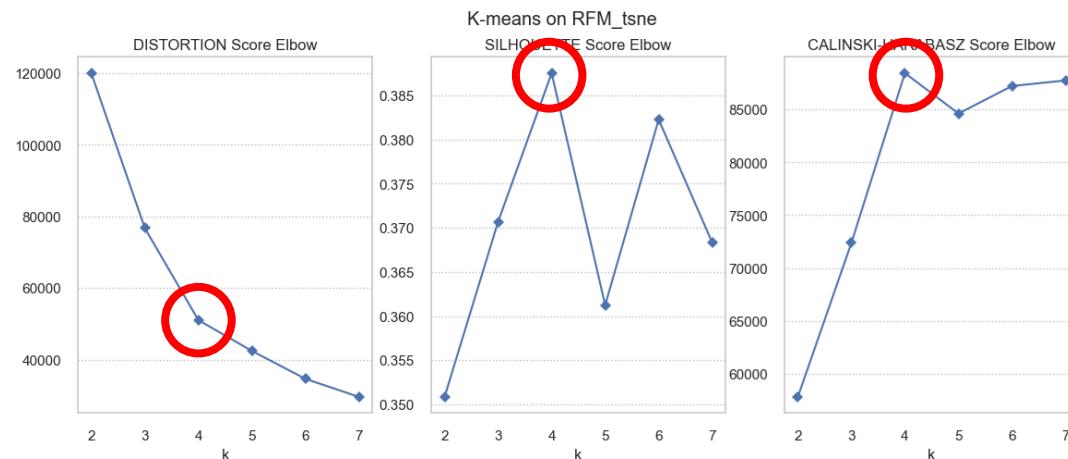
- Test d'une librairie différente : OpenTSNE
 - Explorez des valeurs plus hautes de perplexity
 - Un nouveau paramètre : exaggeration
 - Accroître la densité
 - Utile pour un nombre de points importants

RFM + tSNE – comprendre le manifold ?

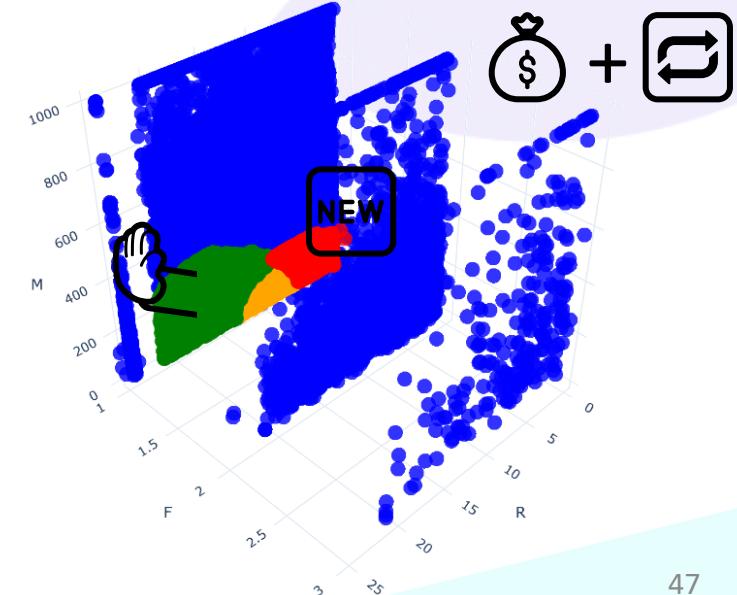
- perplexity augmente →
- exaggeration augmente ↓



RFM + tSNE + KMeans



	Silhouette	Calinski-H	Davies-Bouldin
KMeans on RFM	0.47	91287.94	0.68
ACH on RFM 10 percent	0.45	12023.80	0.71
KMeans on RFM_tsne - metrics on RFM_tsne	0.39	88459.09	0.81
KMeans on RFM_tsne - metrics on RFM	0.26	22686.63	1.26

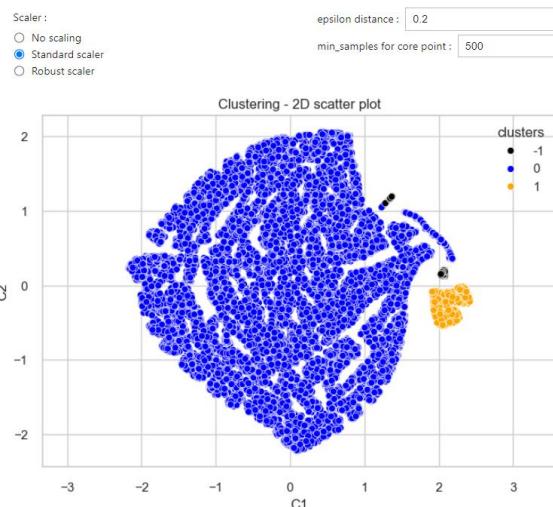
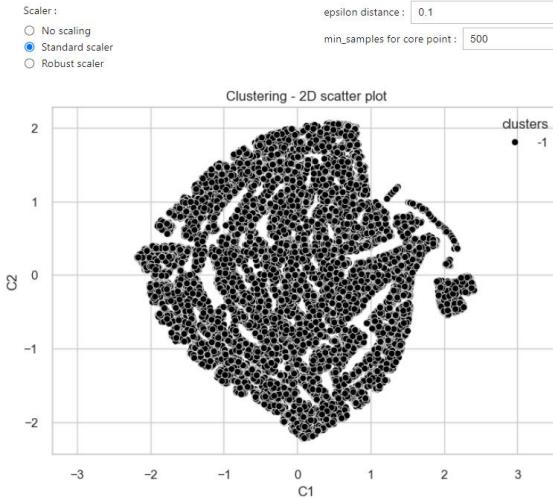


RFM + tSNE + DBSCAN

- Algorithme à densité
- Tests pour différentes valeurs de `epsilon` et `nmin`
- Les résultats n'ont pas été très concluants également
- Métriques :

	Silhouette	Calinski-H	Davies-Bouldin
KMeans on RFM	0.47	91287.94	0.68
ACH on RFM 10 percent	0.45	12023.80	0.71
KMeans on RFM_tsne - metrics on RFM_tsne	0.39	88459.09	0.81
KMeans on RFM_tsne - metrics on RFM	0.26	22686.63	1.26
DBSCAN on RFM_tsne - metrics on RFM_tsne	-0.17	2511.08	0.69
DBSCAN on RFM_tsne - metrics on RFM	-0.20	16312.35	0.86

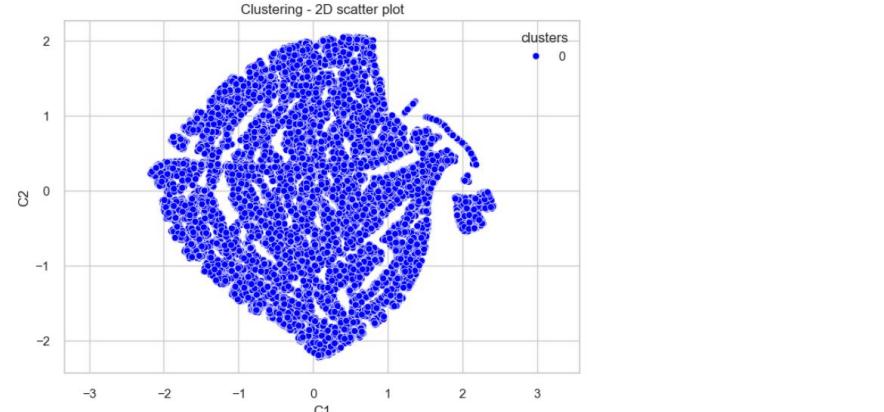
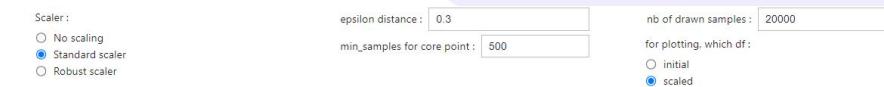
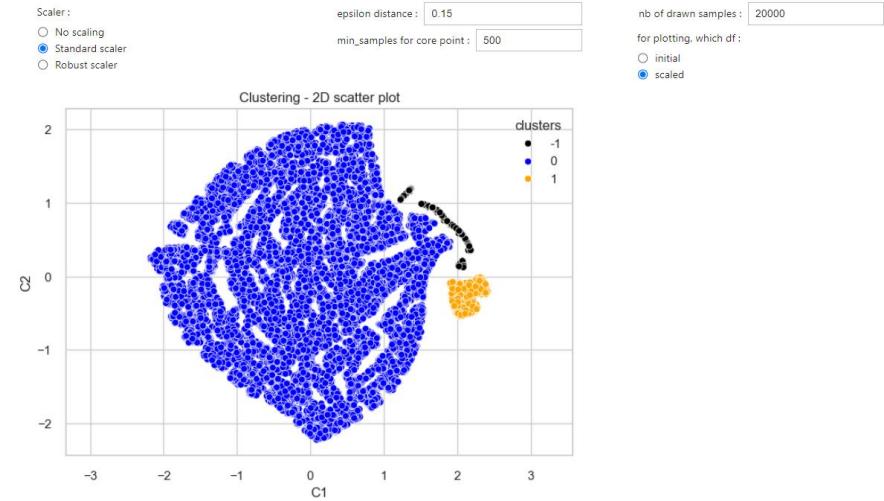
RFM + tSNE + DBSCAN – faire varier epsilon



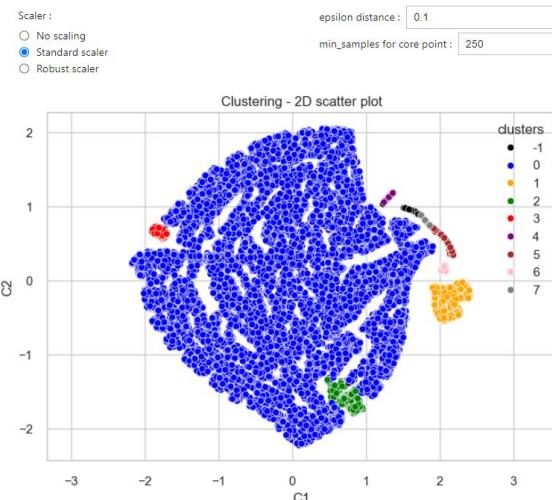
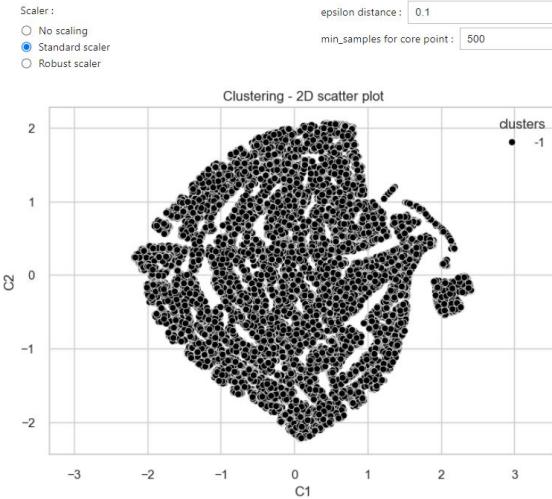
nb of drawn samples : 20000

for plotting, which df :

- initial
- scaled



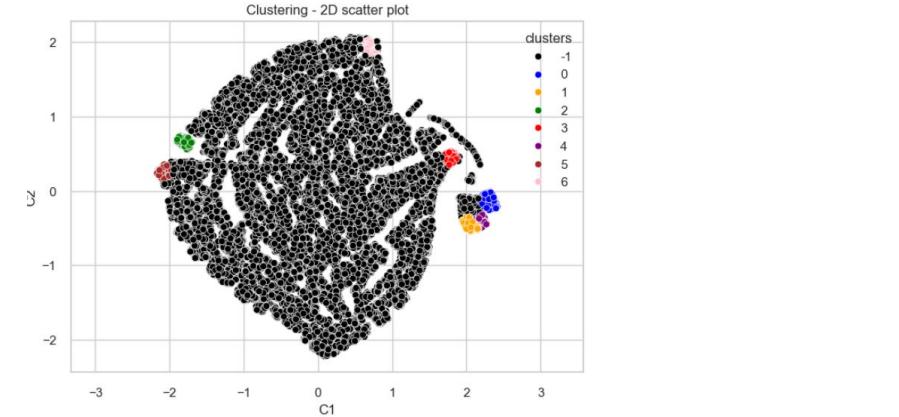
RFM + tSNE + DBSCAN – faire varier nmin



nb of drawn samples : 20000
for plotting, which df : initial
 scaled

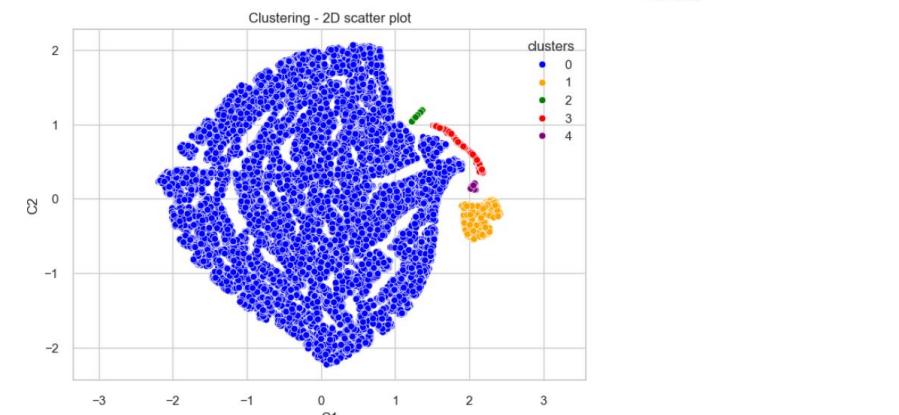
Scaler : No scaling Standard scaler Robust scaler

epsilon distance : 0.1
min_samples for core point : 450
nb of drawn samples : 20000
for plotting, which df : initial
 scaled

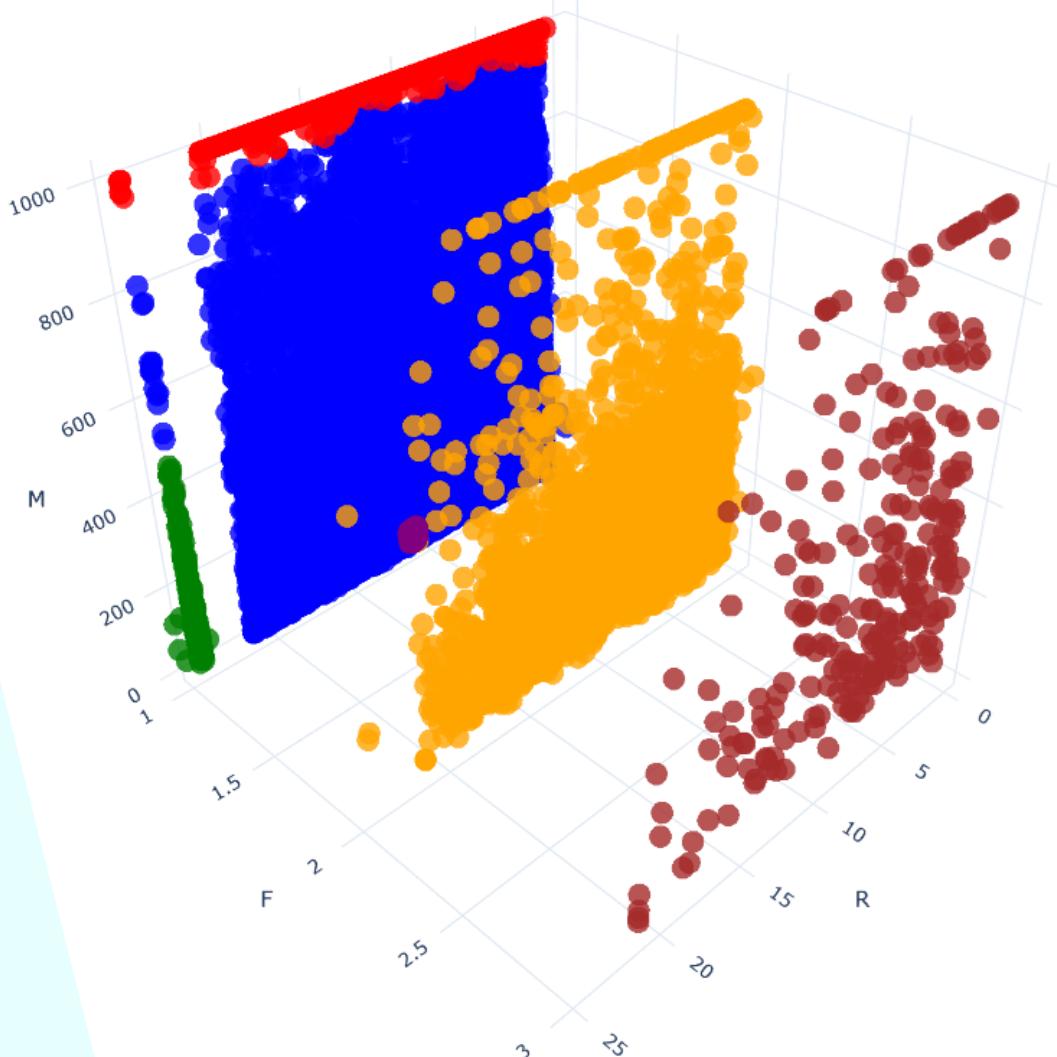


Scaler : No scaling Standard scaler Robust scaler

epsilon distance : 0.1
min_samples for core point : 150
nb of drawn samples : 20000
for plotting, which df : initial
 scaled



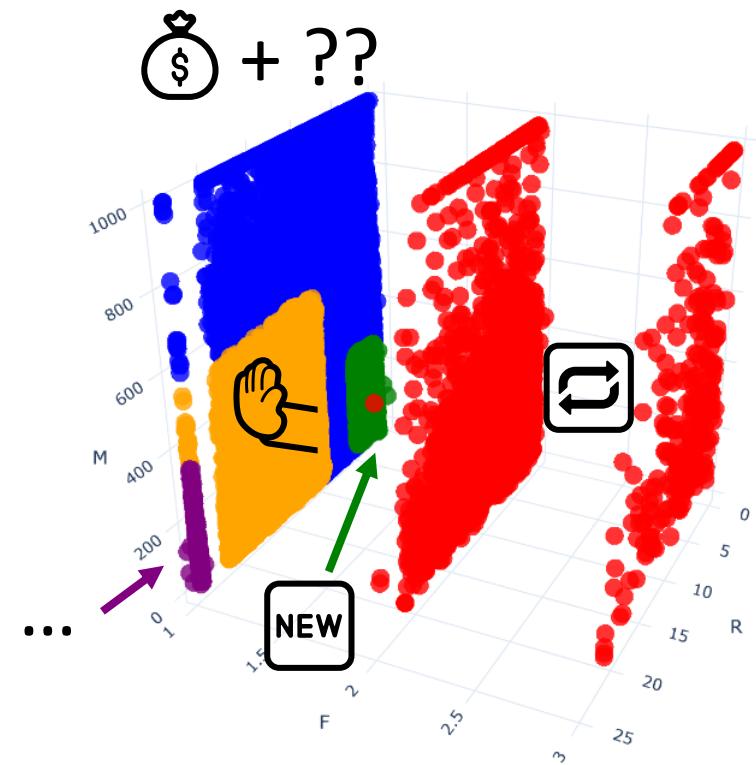
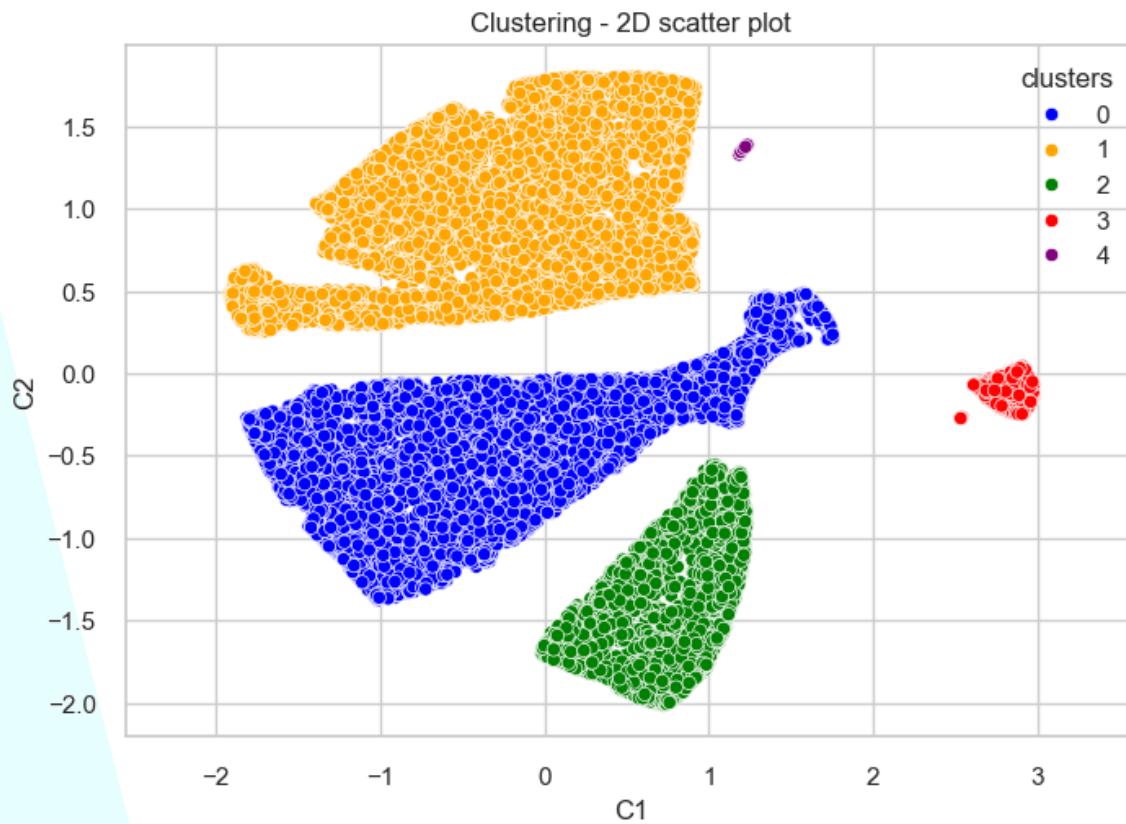
RFM + tSNE + DBSCAN – faire varier nmin



- Presque tous les clients dans un même cluster ...
- Tous ces clients avec $F = 1$ sont en effet séparés, mais ...
 - Juste le gap autour de « R » = 22
 - Clients très dépensiers car zone plus dense (mais densité créée lors de la gestion des outliers...)
 - Autour de « R » = 10, sans doute dû au pic de commandes du novembre 2017 ...

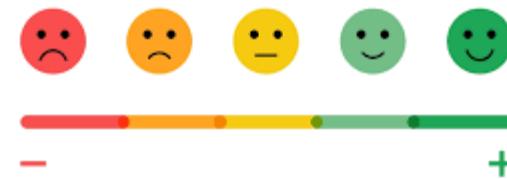
RFM + tSNE 2 + DBSCAN

- Utilisation couplée de l'exaggeration et de DBSCAN !



RFMDR : RFM + DELAY + review_score

- Nous avions vu que ces features, très intéressantes d'un point de vue marketing, ressortaient en termes de variance.
- Le niveau de satisfaction et les éventuels retards pourraient apporter du sens à notre segmentation, et un ciblage plus fin.

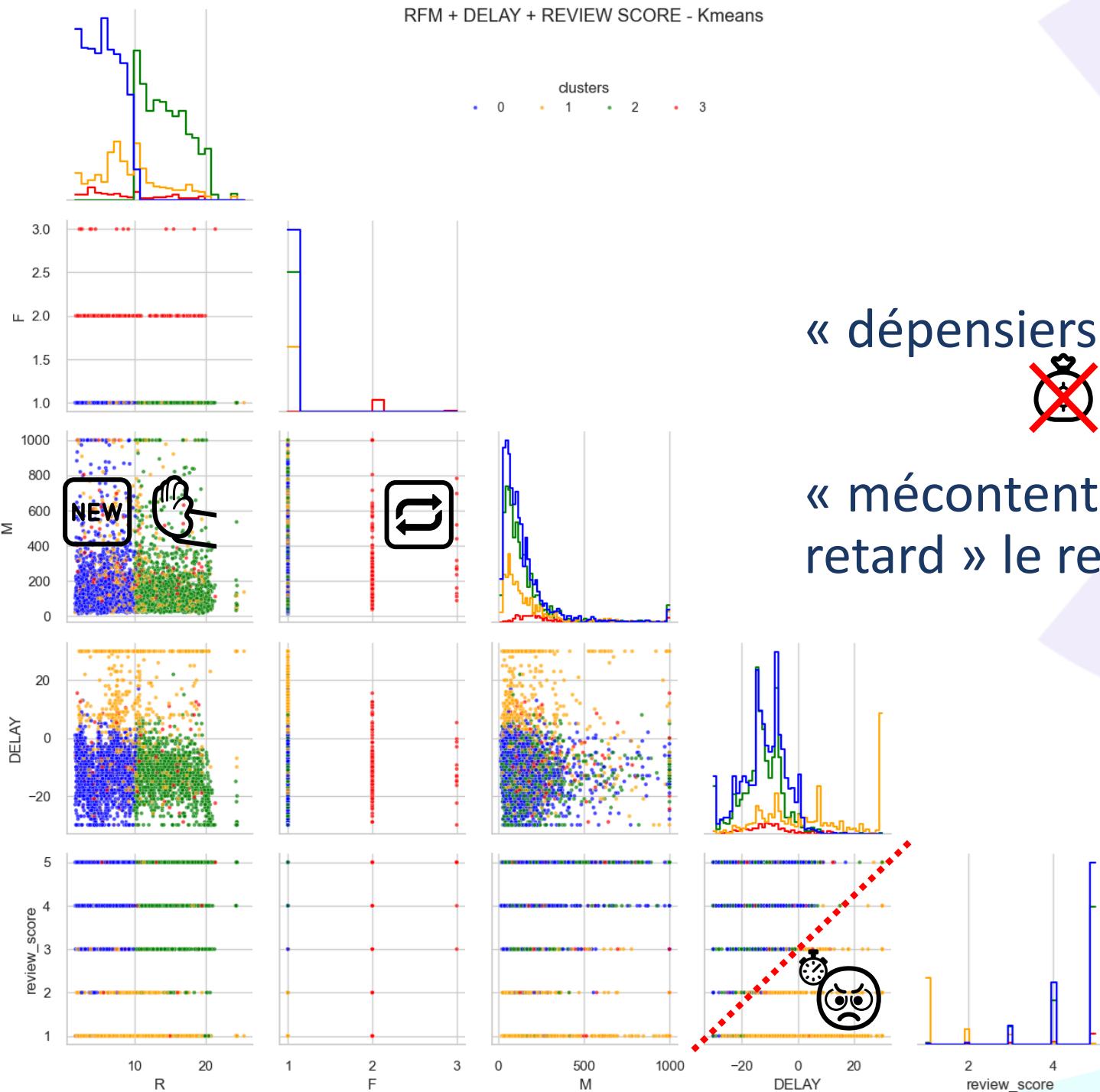


- Challenge pour la compréhension des clusters : 5D vs Représentation graphique

RFMDR + KMeans

- Tests de différents `n_clusters` :

- K = 4 :



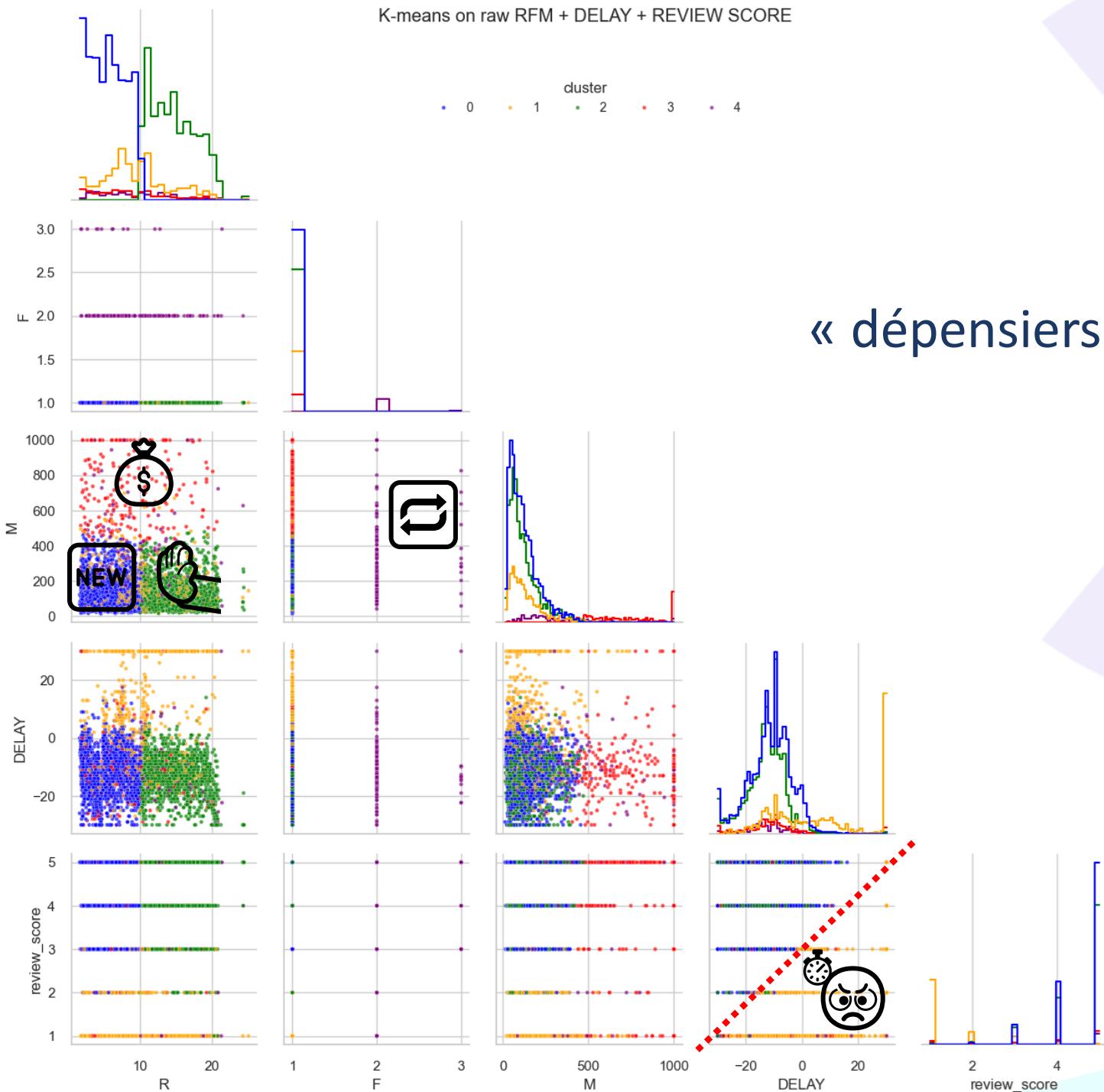
« dépensiers » a disparu



!!!

« mécontents et/ou avec retard » le remplace

- K = 5 :

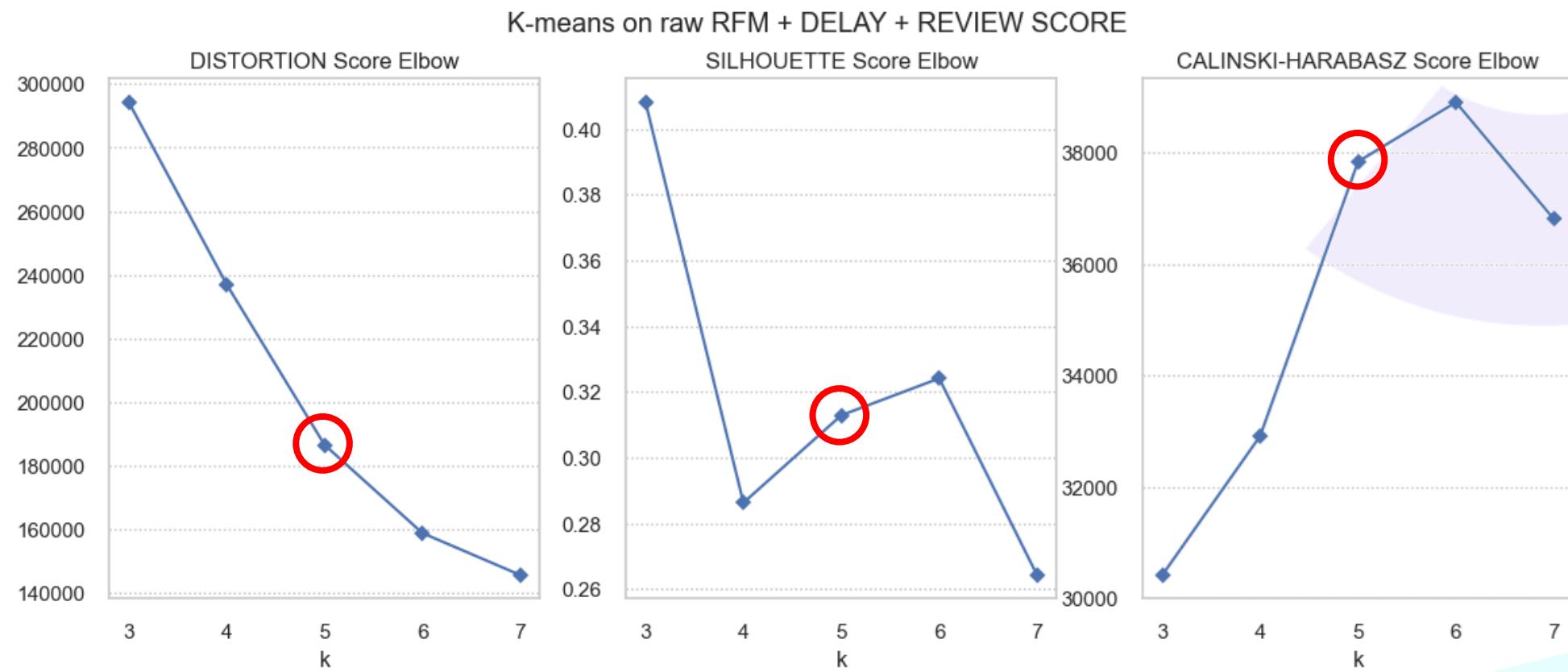


- K = 6 :



RFMDR + Kmeans – méthode du coude

- Des cassures moins nettes :

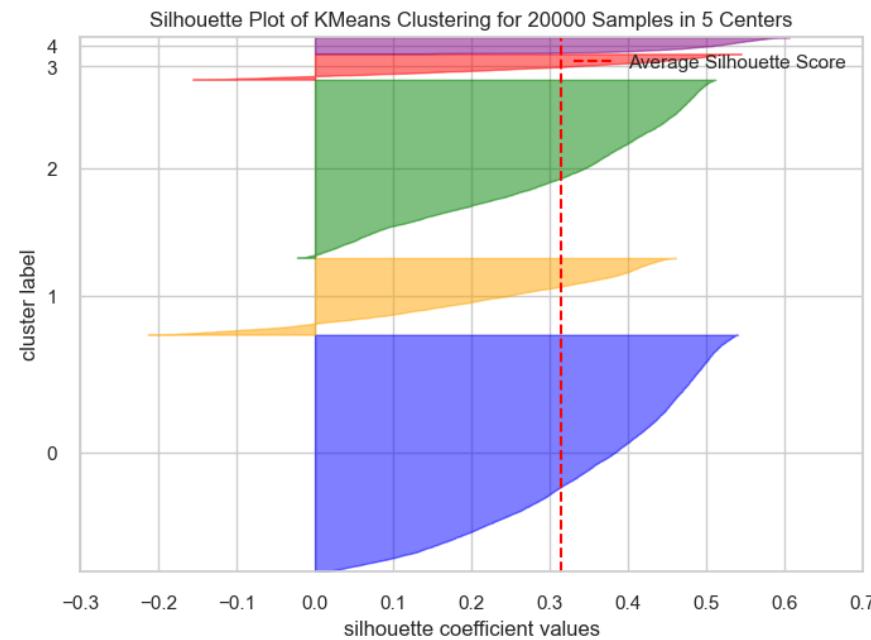


RFMDR + Kmeans – Évaluer notre clustering

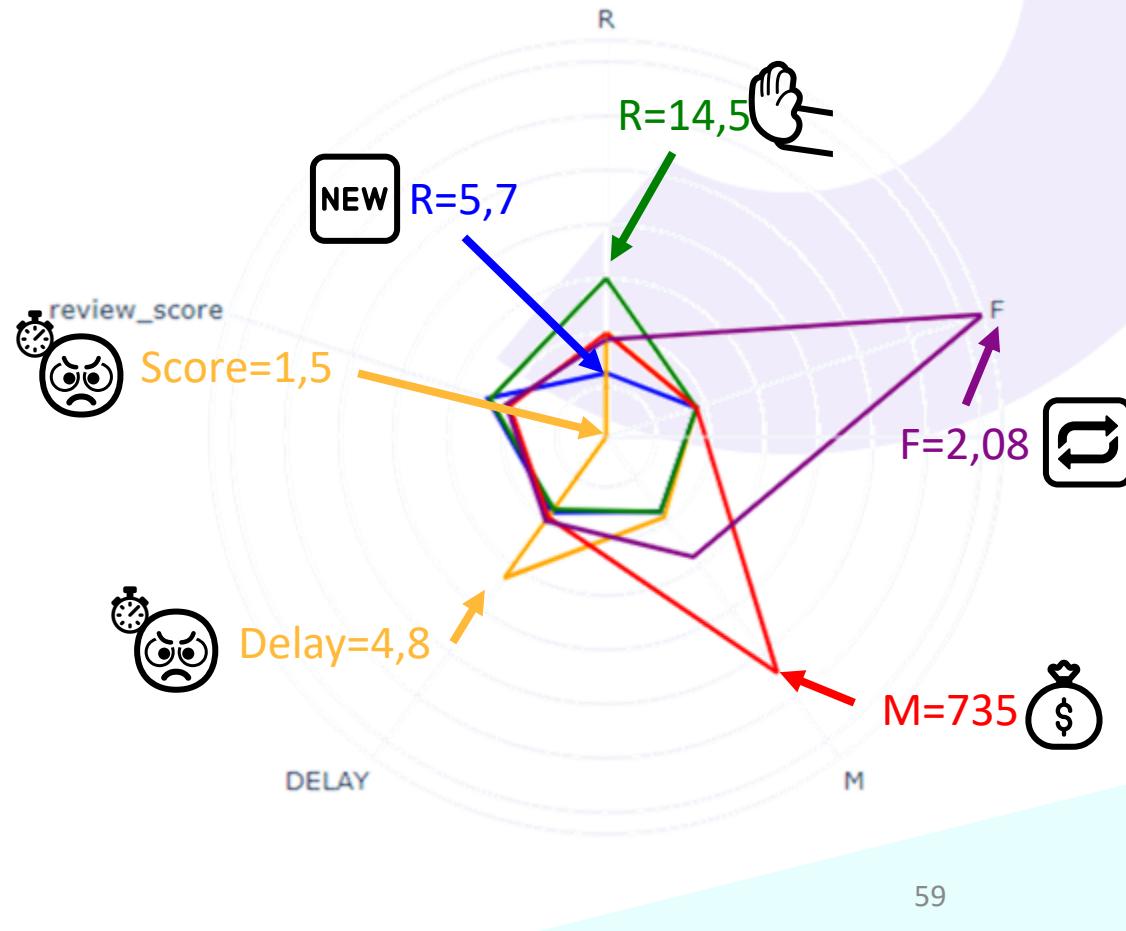
- Métriques

	Silhouette	Calinski-H	Davies-Bouldin
KMeans on RFMDR	0.32	38907.9	0.96

- Silhouette plot



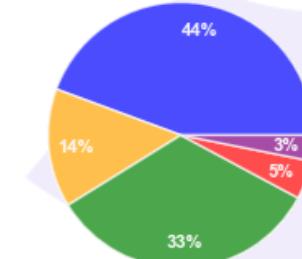
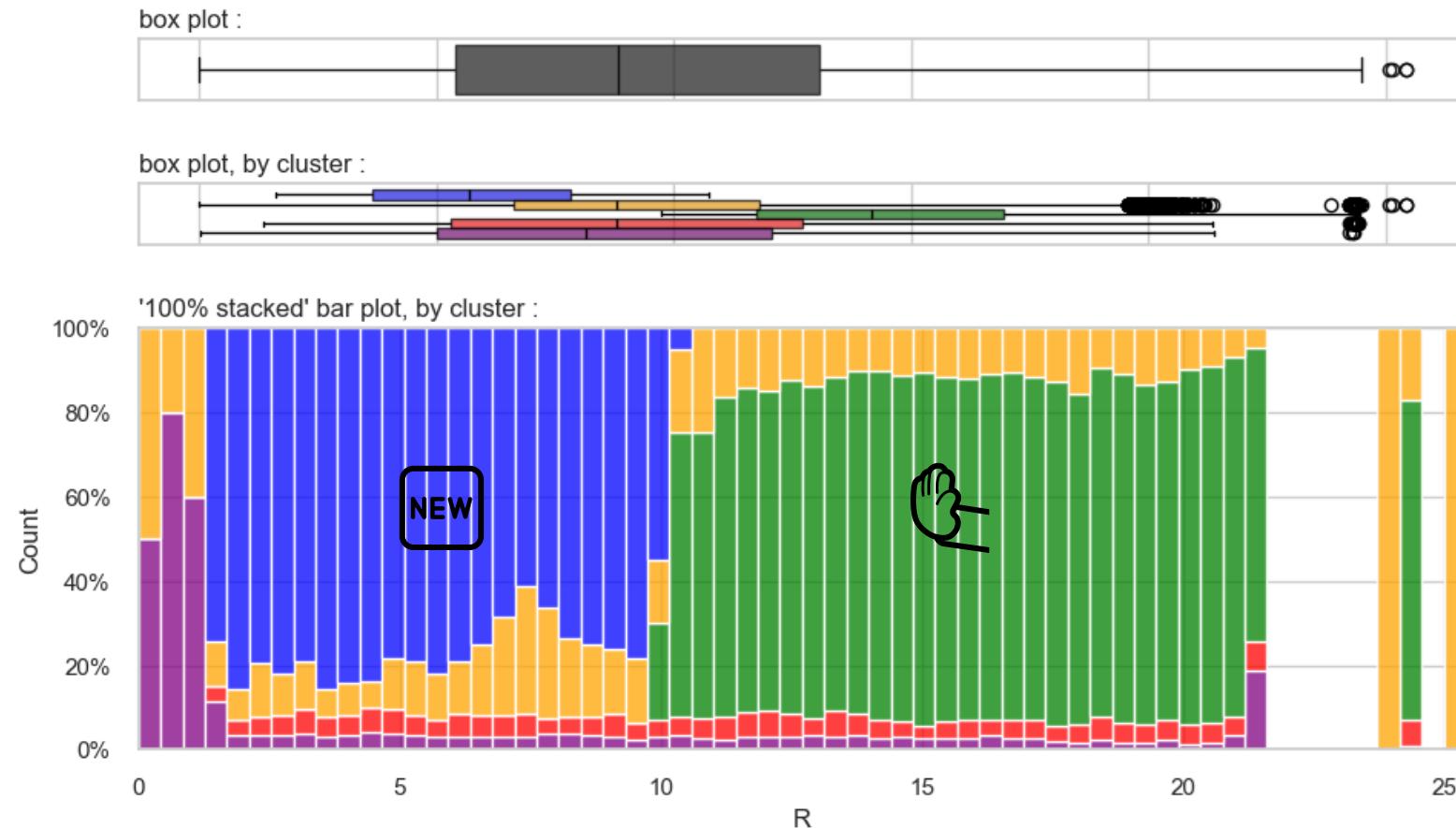
- Radar plot



RFMDR + Kmeans – Évaluer notre clustering

- Graphique en barres empilées

Kmeans on RFM + DELAY + REVIEW SCORE - feature 'R' distribution, by cluster

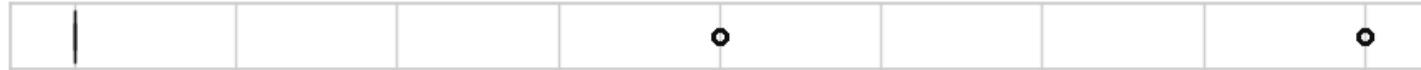


RFM + Kmeans – Évaluer notre clustering

- Graphique en barres empilées

Kmeans on RFM + DELAY + REVIEW SCORE - feature 'F' distribution, by cluster

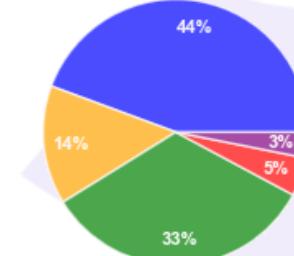
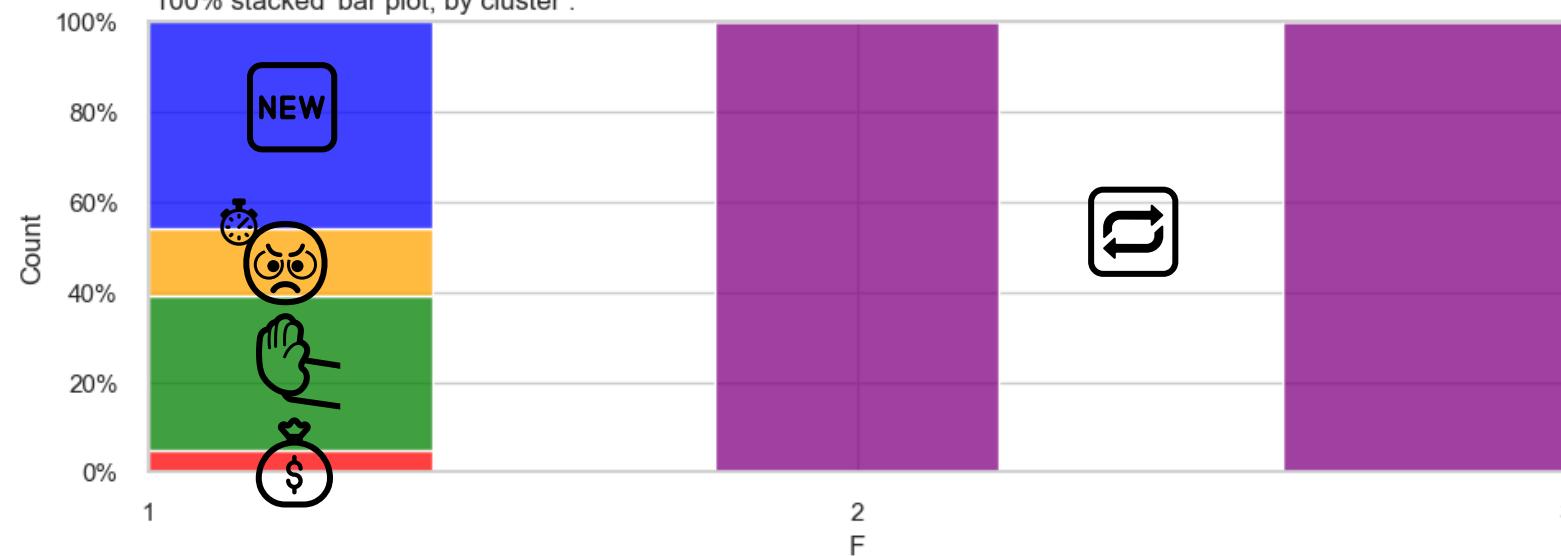
box plot :



box plot, by cluster :



'100% stacked' bar plot, by cluster :

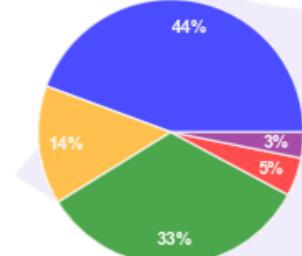
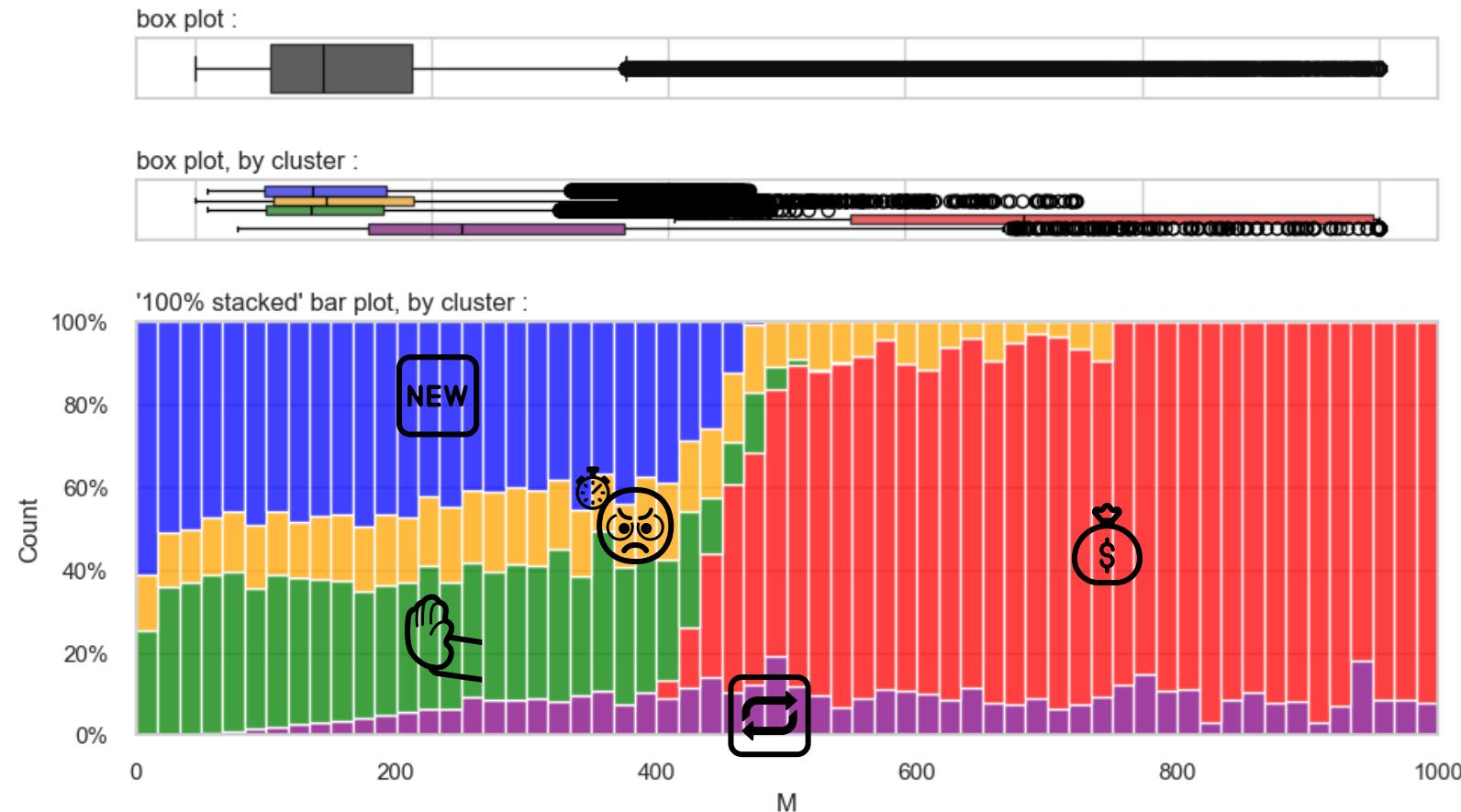


cluster
0
1
2
3
4

RFM + Kmeans – Évaluer notre clustering

- Graphique en barres empilées

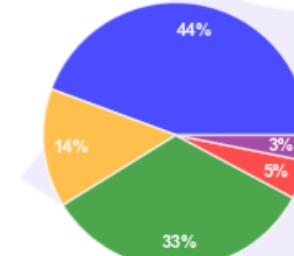
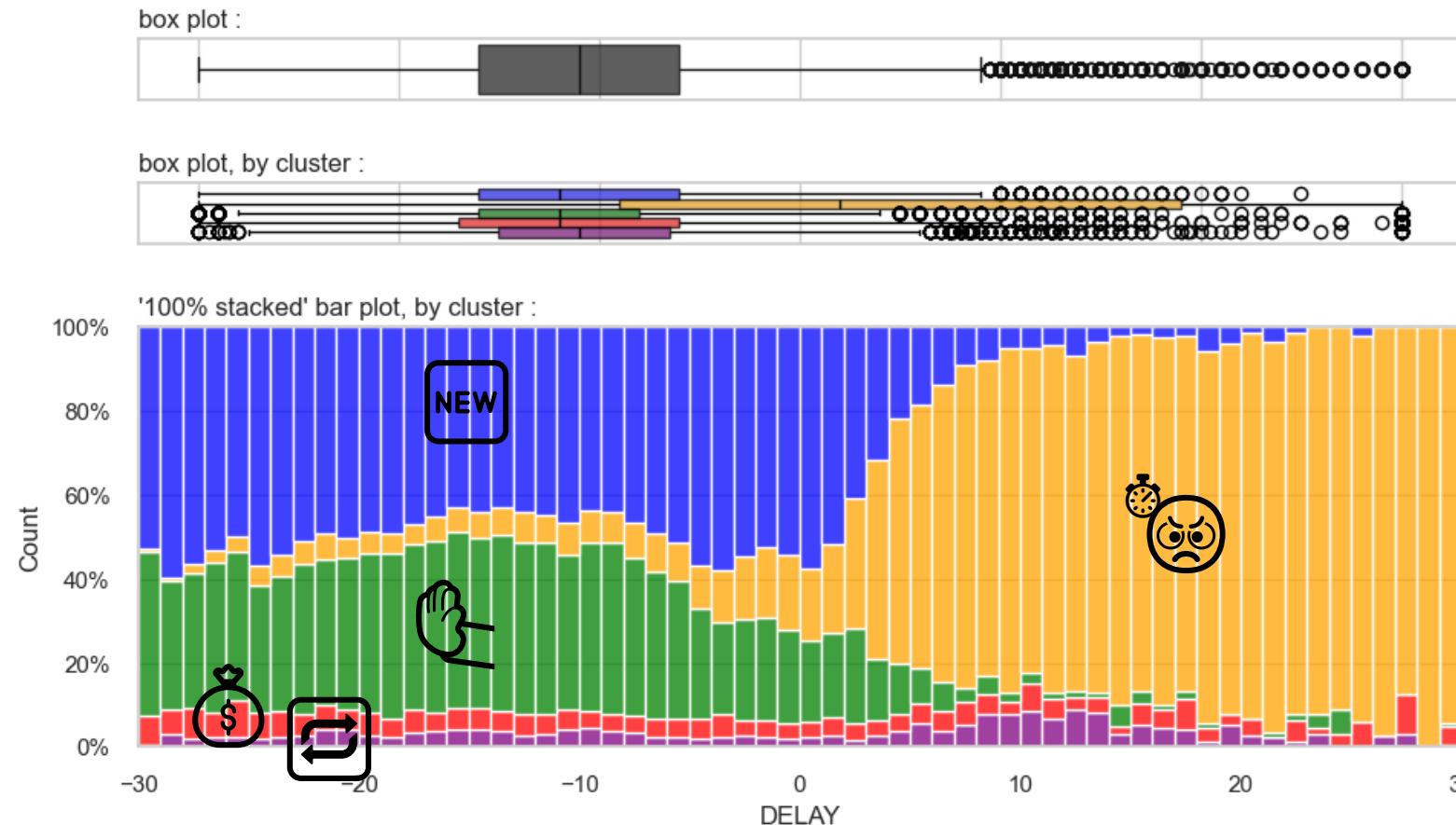
Kmeans on RFM + DELAY + REVIEW SCORE - feature 'M' distribution, by cluster



RFM + Kmeans – Évaluer notre clustering

- Graphique en barres empilées

Kmeans on RFM + DELAY + REVIEW SCORE - feature 'DELAY' distribution, by cluster



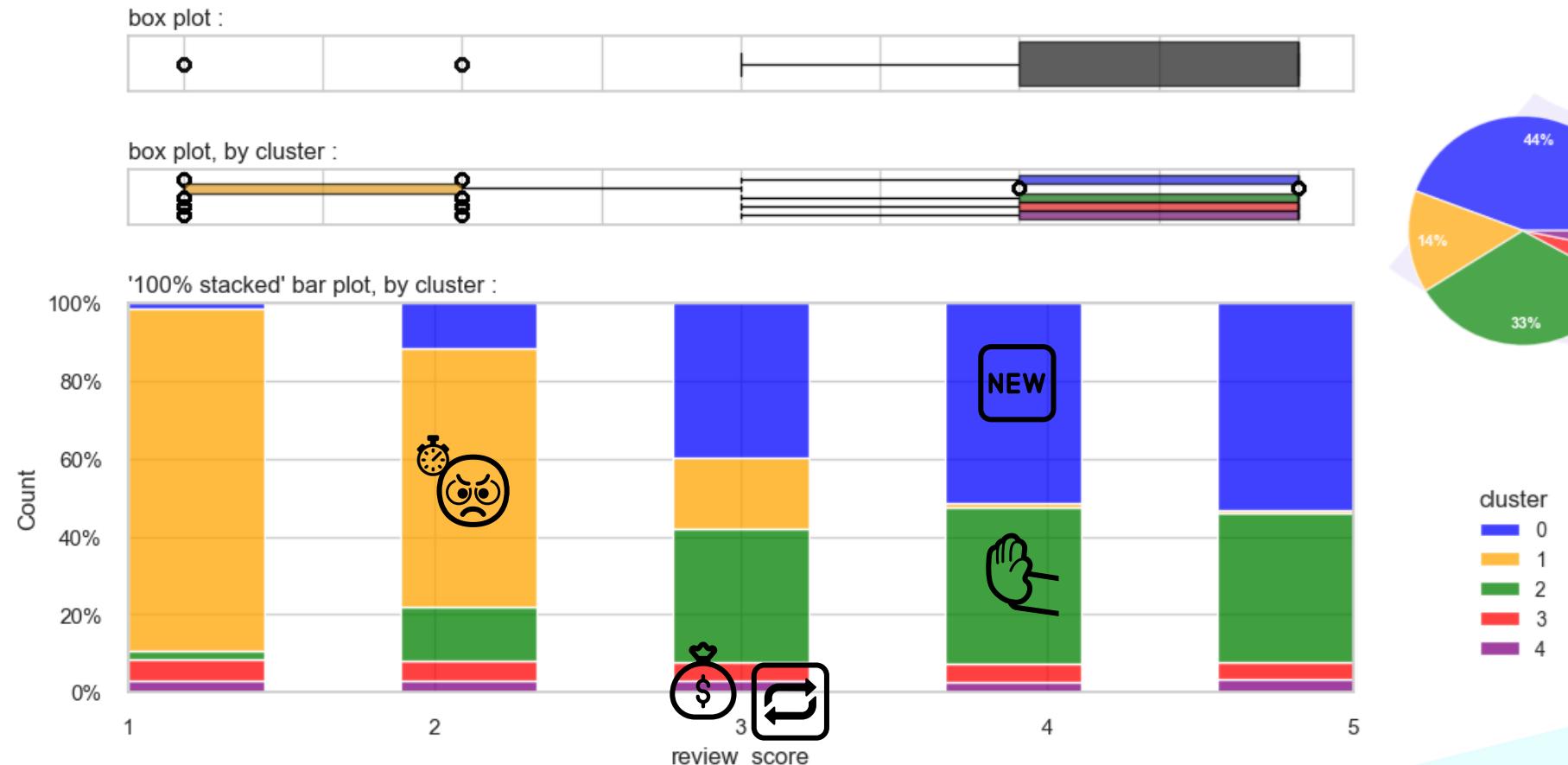
cluster

- 0
- 1
- 2
- 3
- 4

RFM + Kmeans – Évaluer notre clustering

- Graphique en barres empilées

Kmeans on RFM + DELAY + REVIEW SCORE - feature 'review_score' distribution, by cluster



RFMDR + Kmeans – Nos clusters

- « dépensiers » 💰
 - Intéressants, il faut gagner leur confiance
- « récurrents » 🔍
 - Les plus intéressants, peuvent devenir des fidèles
- « ex-clients » 🖐
 - Pas vraiment de potentiel, ne pas investir sur eux
- « nouveaux » 🚀
 - Intéressants, agir rapidement pour déclencher une 2nd commande
- « mécontents et/ou avec retard » 🕒 ☹
 - Potentiel faible, se faire pardonner

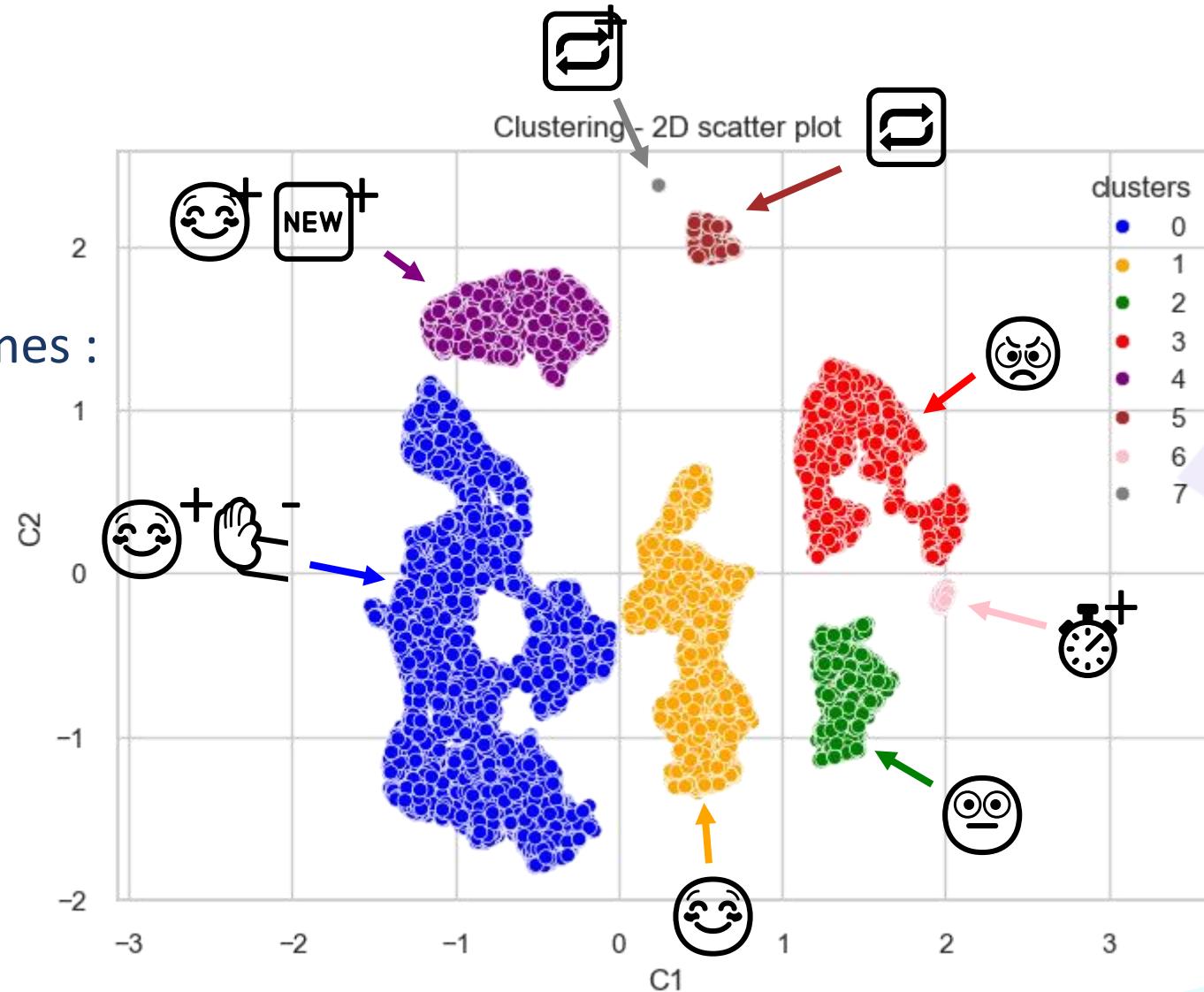
RFMDR + Kmeans – Quelles actions ?

- « dépensiers » 
 - (gros) bon d'achat ? (gros) cadeau de bienvenue ? ventes privées articles onéreux ?
- « récurrents » 
 - Recherche catégorie préférée ? Ventes privées ? Dispositif fidélité ?
- « ex-clients » 
 - Pas d'action
- « nouveaux » 
 - Bon d'achat ? Cadeau de bienvenue ?
- « mécontents ou avec retard » 
 - Comprendre. Enquête ?
 - Proposer un remboursement des frais de livraison ?
 - Offrir un bon d'achat ?

RFMDR + tSNE + DBSCAN

- Nouvelle tentative infructueuse...

- Les différentes formes :

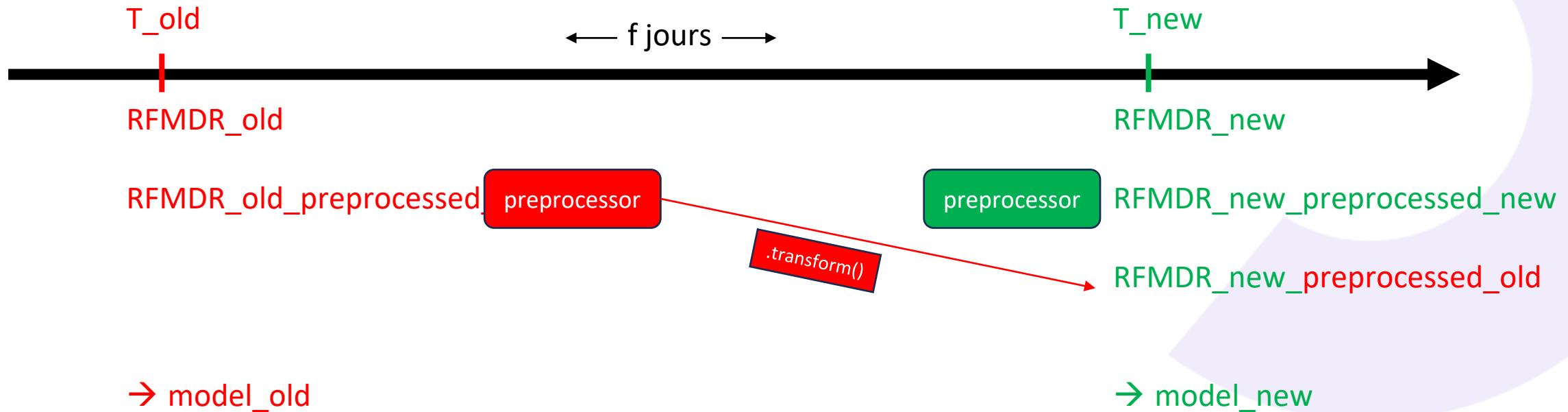


PARTIE 5 - FRÉQUENCE DE MAINTENANCE

Enjeu

- Maintenir notre segmentation      pertinente
au cours du temps
- Comment s'adapter aux nouveaux clients ?
Aux nouvelles commandes ?
À l'évolution naturelle de la RÉCENSE ?
Etc.
- SOLUTION : mettre à jour le modèle
- PROBLÉMATIQUE : à quelle fréquence conduire ses mises à jour ?

Méthode pour trouver « f »



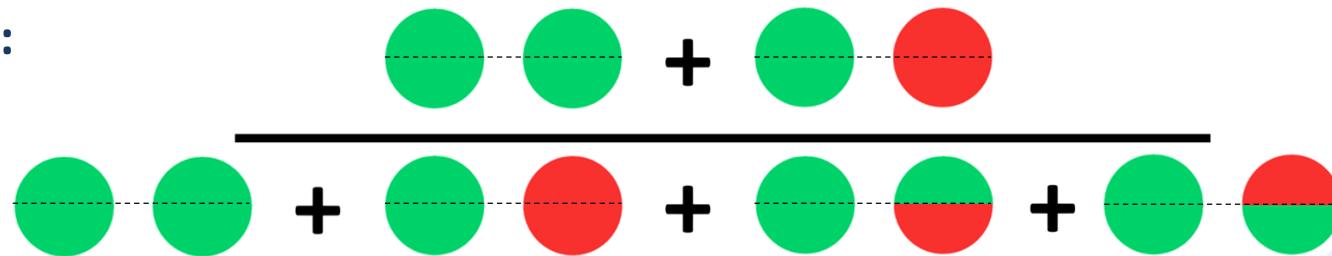
`model_new.predict(RFMDR_new_preprocessed_new)`

VS

`model_old.predict(RFMDR_new_preprocessed_old)`

Comment comparer ?

- Version « **ajustée** » de l'**indice de Rand**
- Indice de Rand : mesure de **similarité** entre deux segmentations d'un même jeu de données
 - **Proportion :**



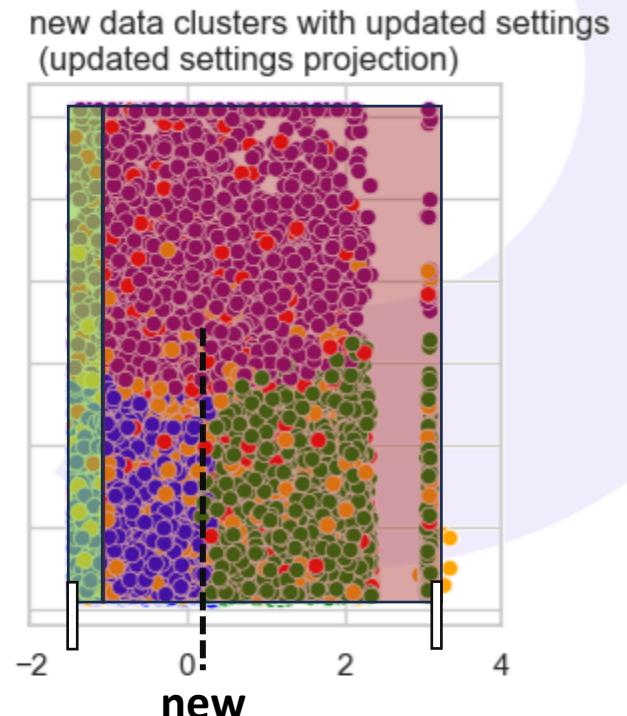
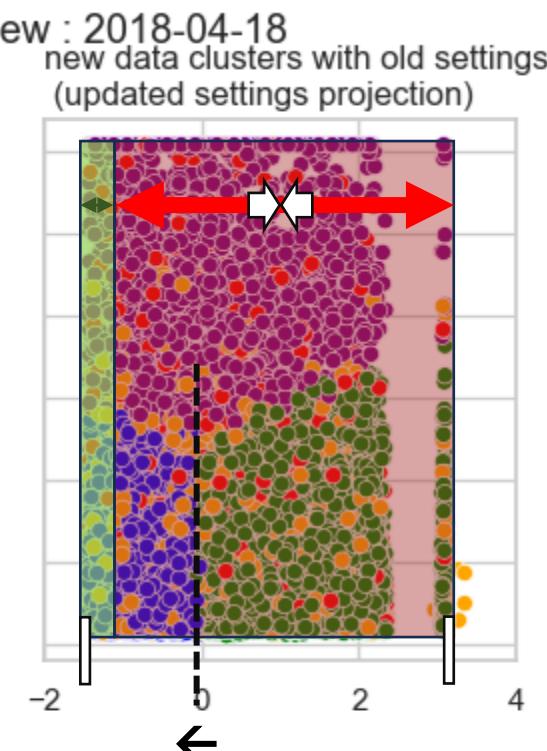
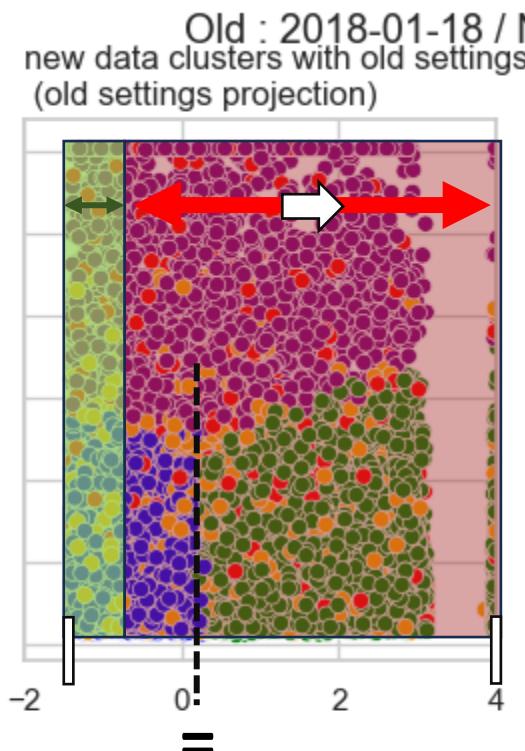
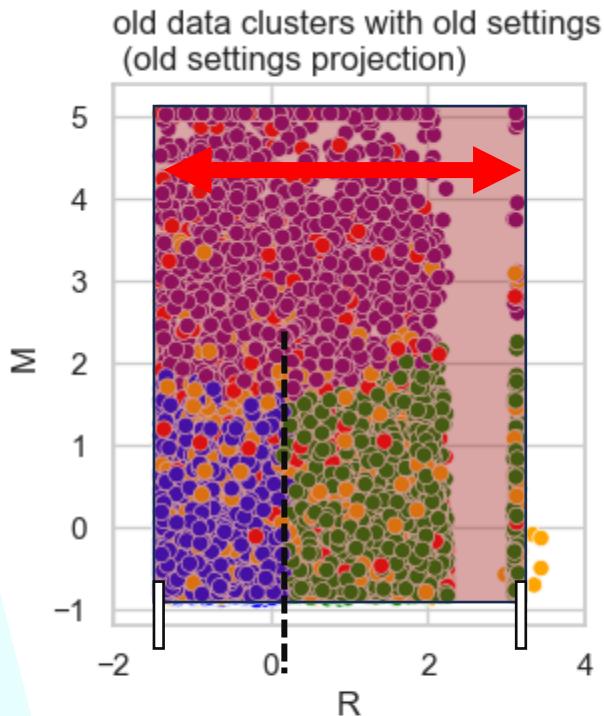
- Pourquoi ajuster le RI ?
 - Pb 1 : pour un clustering aléatoire, l'espérance n'est pas constante (pas 0,5 par exemple)
 - Pb 2 : facile de « tricher »

$$\text{ARI} : \frac{\text{RI} - E(\text{RI})}{\text{Max(RI)} - E(\text{RI})}$$

- $E(\text{ARI}) = 0$ pour des segmentations aléatoires
- Peut être < 0

Objectif :
ARI = 0,9

Test avec $f = 3$ mois

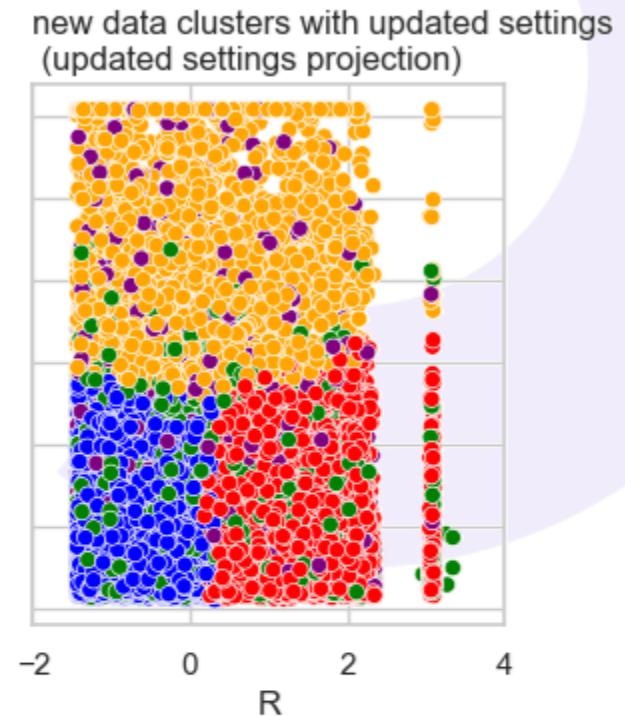
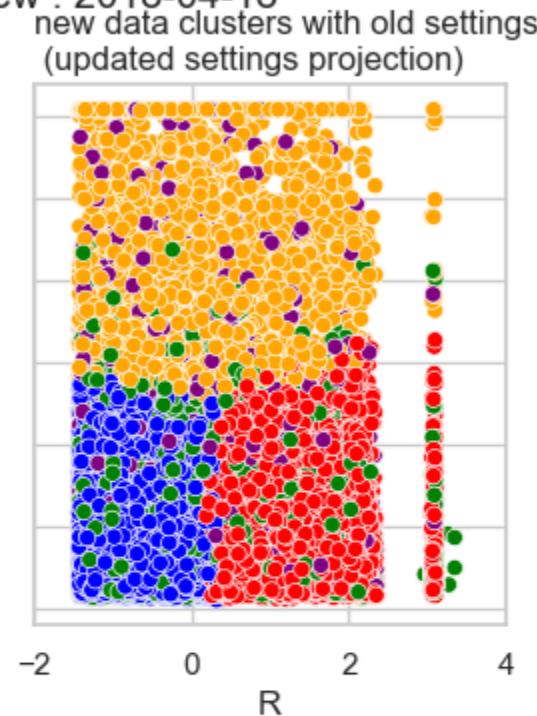
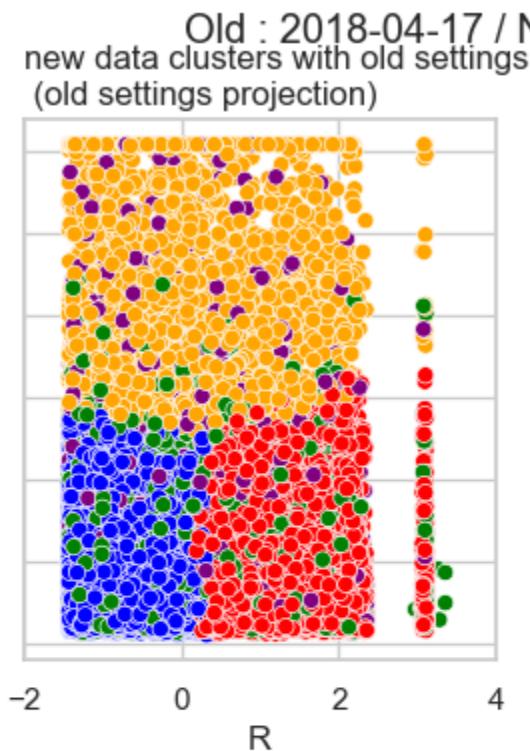
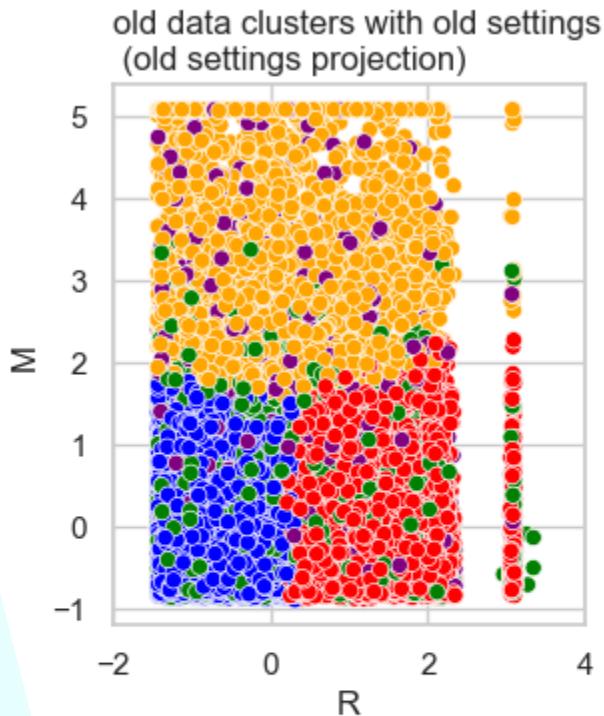


cluster_old_old

- 0
- 1
- 2
- 3
- 4

old VS updated : ARI = 0.77

Test avec $f = 1$ jour



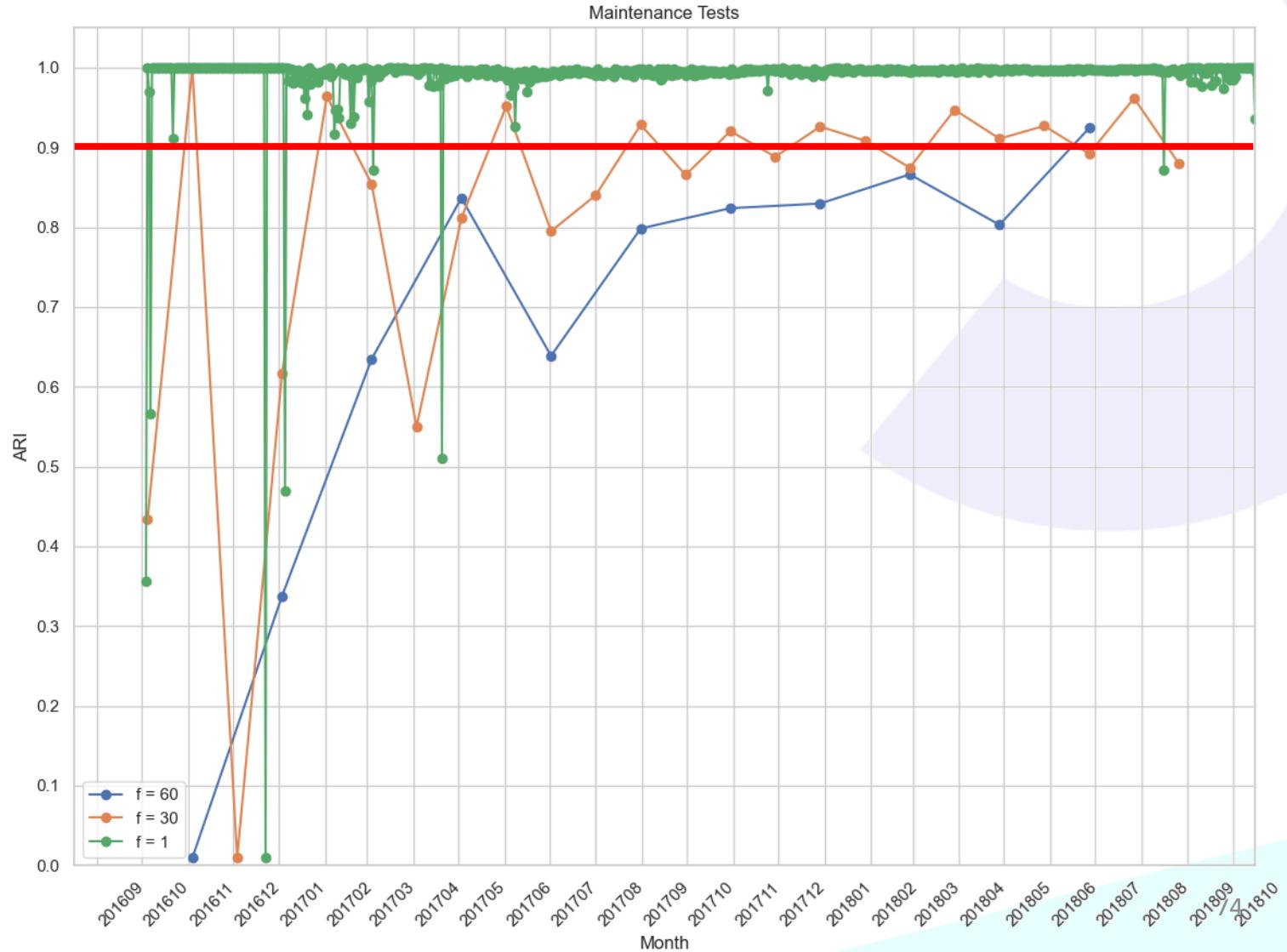
cluster_old_old

- 0
- 1
- 2
- 3
- 4

old VS updated : ARI = 1.0

Maintenances successives – premier test

- $f = 2$ mois, 1 mois , 1 jour
- seuil = 0,9
- période d'étude : entière

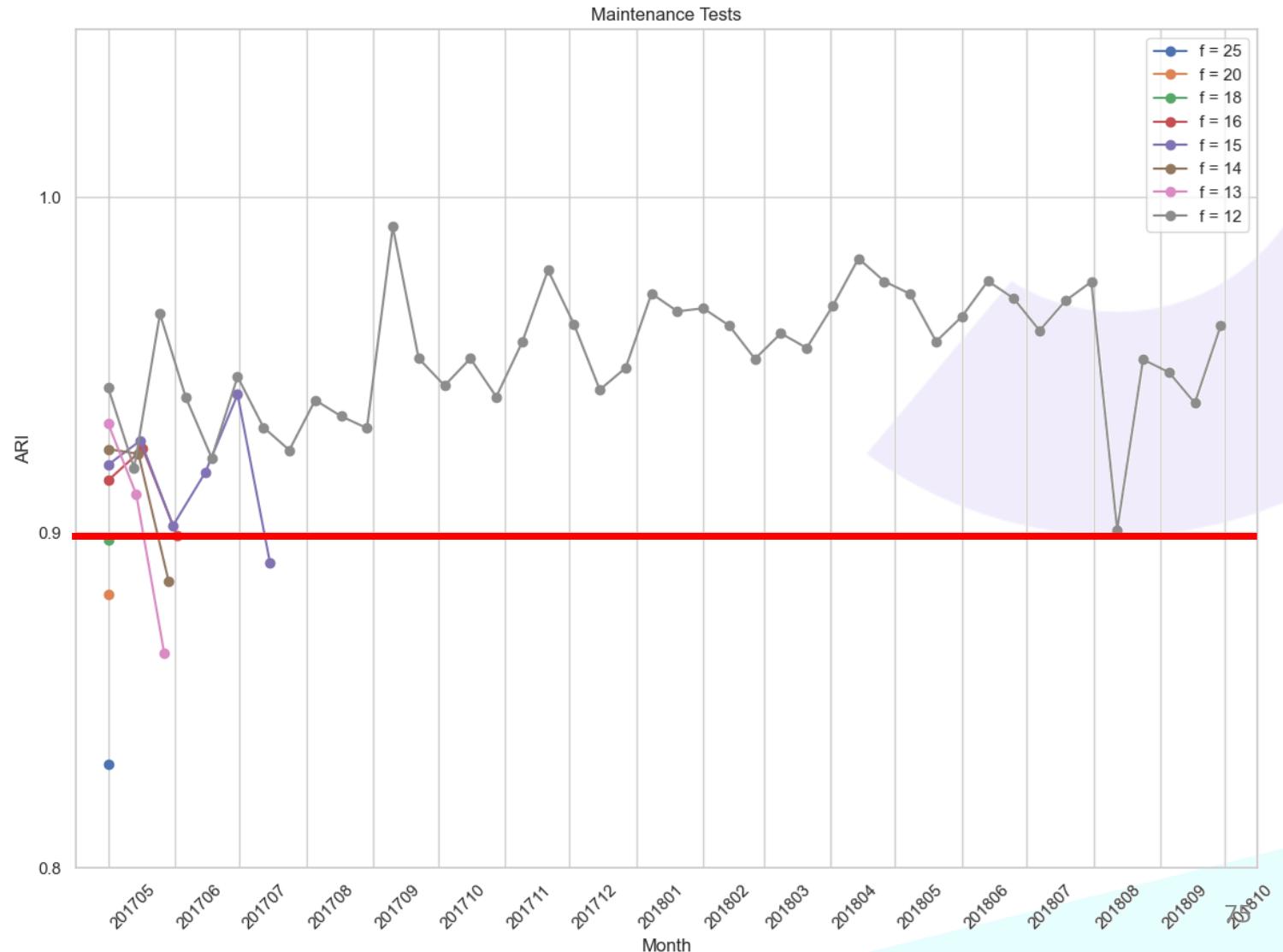


Maintenances successives

- $f = 25, 20, 18, 16, 15, 14, 13, 12, 11, 10, 9, 8$
- seuil = 0,9
- période d'étude : mai 2017 → fin
- 1 échec → fréquence suivante
- fréquence gagnante → arrêt



$f = 12$ jours



CONCLUSION

La modélisation

- Différents modèles testés, tout d'abord dans le **cadre classique RFM**
- puis en rajoutant l'**écart date livraison estimée/réelle** et la **note client**, permettant d'améliorer notre segmentation
- Tentative de clustering sur l'espace de redescription issu d'un **t-SNE...**
- ... mais les clusters n'étaient **pas les plus cohérents**
- Tentative de clustering **DBSCAN** avec des **paramètres intéressants de t-SNE...**
- ... mais résultat mitigé : clusters **soit trop précis, soit inexploitables**
- de plus t-SNE stochastique → très **difficile pour la maintenance**

Nos clusters

- « nouveaux » 
- « récurrents » 
- « dépensiers » 
- « ex-clients » 
- « clients mécontents ou ayant subi un retard » 

Fréquence de mise à jour proposée

- Nous avons simulé la mise en place d'une maintenance avec pour objectif des niveaux de similarité avant / après mise à jour de 0,9
- Nous proposons à Olist de maintenir cette segmentation sur une base de 12 jours

merci