Segmentez des clients d'un site e-commerce

par Ana Bernal
Mentor: Samir Tanfous

Programme

1 Rappel **mission**

2 Nettoyage + feature engineering + **exploration**

3 **Modélisation** (essais + **profil clients** clusters)

4 Simulation **délai** de maintenance

5 Conclusion

1. Rappel mission

Rappel mission

Bonjour,

Pour des raisons de confidentialité, nous ne pouvons pas vous fournir beaucoup de données à ce stade. Ensuite, en raison de ressources limitées, nous avons dû vous fournir l'ensemble des données, alors que seule une partie va vous intéresser. Nos dashboards internes nous indiquent en effet que seuls 3 % des clients du fichier de données partagé avec vous ont réalisé plusieurs commandes.

Nous sommes confiants sur le fait que les données à disposition suffiront pour réaliser un premier clustering. Cela a déjà été fait par d'autres prestataires par le passé, avec encore moins de données.

La segmentation proposée doit être exploitable et facile d'utilisation par notre équipe Marketing. Elle doit au minimum pouvoir différencier les bons et moins bons clients en termes de commandes et de satisfaction. Nous attendons bien sûr une segmentation sur l'ensemble des clients.

Dans un deuxième temps, une fois le modèle de segmentation choisi, nous souhaiterions que vous nous fassiez une recommandation de fréquence à laquelle la segmentation doit être mise à jour pour rester pertinente, afin de pouvoir effectuer un devis de contrat de maintenance.
Pour information, le code fourni doit respecter la convention PEP8, pour être utilisable par Olist.

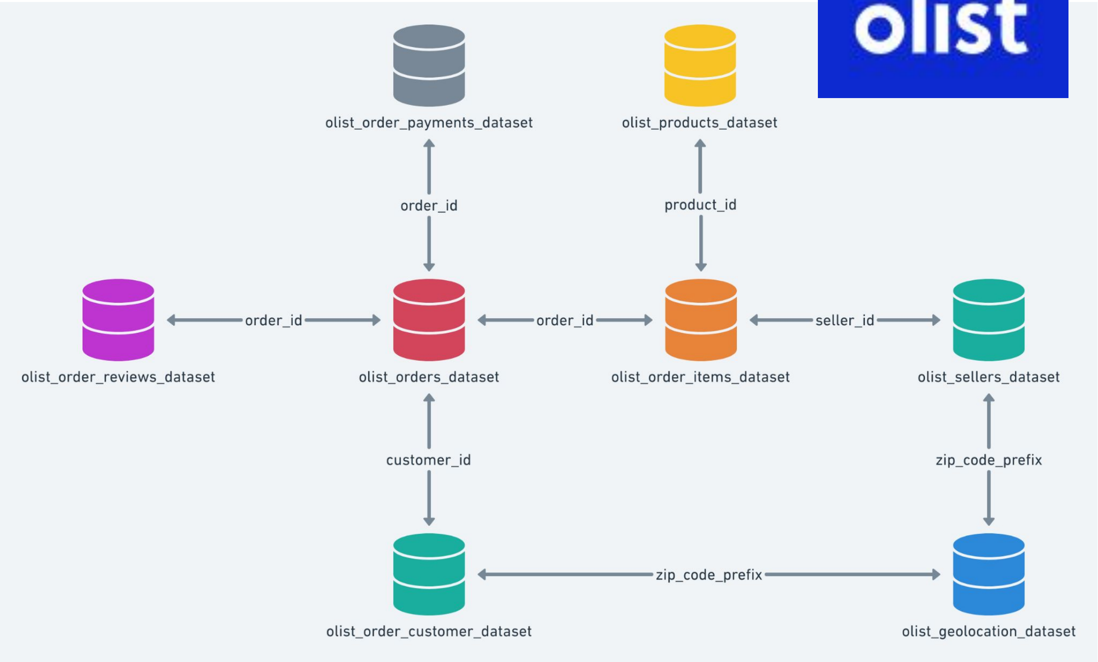
Bien à vous,
Juan, de l'équipe Marketing

Objectifs

1. Segmentation clients
2. Délai de maintenance

2. Nettoyage + feat. eng. + exploration

Nettoyage + feature engineering + exploration



- 8 dataframes
- But: 1 seul dataframe

Nettoyage + feature engineering + exploration

Aperçu des dataframes

	numb_rows	numb_cols	isna_cols_list	isna_cols_perc
name				
geolocation	1000163	5	[]	[]
order-reviews	99224	7	[review_comment_title, review_comment_message]	[88.34, 58.7]
translation-category	71	2	[]	[]
customers	99441	5	[]	[]
products	32951	9	[product_category_name, product_name_lenght, p...	[1.85, 1.85, 1.85, 1.85, 0.01, 0.01, 0.01, 0.01]
sellers	3095	4	[]	[]
order-items	112650	7	[]	[]
order-payments	103886	5	[]	[]
orders	99441	8	[order_approved_at, order_delivered_carrier_da...	[0.16, 1.79, 2.98]

Nettoyage + feature engineering + exploration

Aperçu des dataframes

	numb_rows	numb_cols	isna_cols_list	isna_cols_perc
name				
geolocation	1000163	5	[]	[]
order-reviews	99224	7	[review_comment_title, review_comment_message]	[88.34, 58.7]
translation-category	71	2	[]	[]
customers	99441	5	[]	[]
products	32951	9	[product_category_name, product_name_lenght, p...	[1.85, 1.85, 1.85, 1.85, 0.01, 0.01, 0.01, 0.01]
sellers	3095	4	[]	[]
order-items	112650	7	[]	[]
order-payments	103886	5	[]	[]
orders	99441	8	[order_approved_at, order_delivered_carrier_da...	[0.16, 1.79, 2.98]

dataframes de base

Nettoyage + feature engineering + exploration

Aperçu des dataframes

	numb_rows	numb_cols	isna_cols_list	isna_cols_perc
name				
geolocation	1000163	5	[]	[]
order-reviews	99224	7	[review_comment_title, review_comment_message]	[88.34, 58.7]
translation-category	71	2	[]	[]
customers	99441	5	[]	[]
products	32951	9	[product_category_name, product_name_lenght, p...	[1.85, 1.85, 1.85, 1.85, 0.01, 0.01, 0.01, 0.01]
sellers	3095	4	[]	[]
order-items	112650	7	[]	[]
order-payments	103886	5	[]	[]
orders	99441	8	[order_approved_at, order_delivered_carrier_da...	[0.16, 1.79, 2.98]

dataframes de base

clé principale:
`customer_unique_id`

Nettoyage + feature engineering + exploration

Aperçu des dataframes

	numb_rows	numb_cols	isna_cols_list	isna_cols_perc
name				
geolocation	1000163	5	[]	[]
order-reviews	99224	7	[review_comment_title, review_comment_message]	[88.34, 58.7]
translation-category	71	2	[]	[]
customers	99441	5	[]	[]
products	32951	9	[product_category_name, product_name_lenght, p...	[1.85, 1.85, 1.85, 1.85, 0.01, 0.01, 0.01, 0.01]
sellers	3095	4	[]	[]
order-items	112650	7	[]	[]
order-payments	103886	5	[]	[]
orders	99441	8	[order_approved_at, order_delivered_carrier_da...	[0.16, 1.79, 2.98]

plusieurs items par achat ➡ des choix à faire

Nettoyage + feature engineering + exploration

Aperçu des dataframes

name	numb_rows	numb_cols	isna_cols_list	isna_cols_perc
geolocation	1000163	5	[]	[]
order-reviews	99224	7	[review_comment_title, review_comment_message]	[88.34, 58.7]
translation-category	71	2	[]	[]
customers	99441	5	[]	[]
products	32951	9	[product_category_name, product_name_lenght, p...	[1.85, 1.85, 1.85, 1.85, 0.01, 0.01, 0.01, 0.01]
sellers	3095	4	[]	[]
order-items	112650	7	[]	[]
order-payments	103886	5	[]	[]
orders	99441	8	[order_approved_at, order_delivered_carrier_da...	[0.16, 1.79, 2.98]

base de données très complète
(sauf les reviews)

Nettoyage + feature engineering + exploration

Quelques stats générales avant merge

Nettoyage + feature engineering + exploration

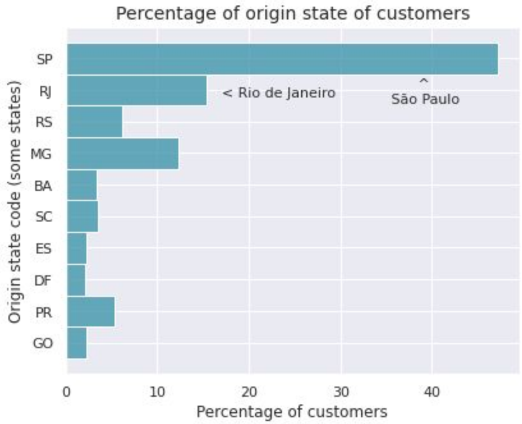
Quelques stats générales avant merge

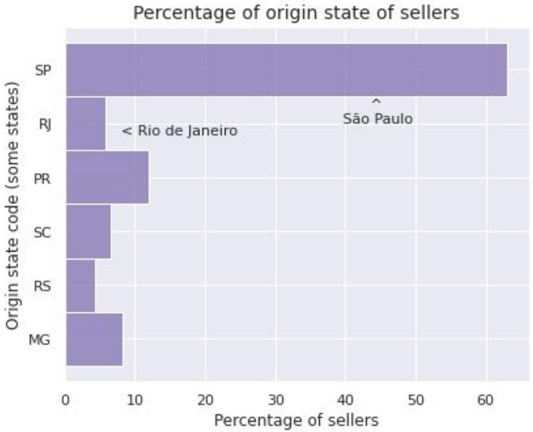
Localisations géographiques (clients + vendeurs)

Nettoyage + feature engineering + exploration

Quelques stats générales avant merge

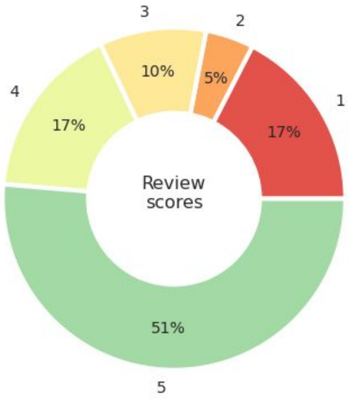
Localisations géographiques (clients + vendeurs)

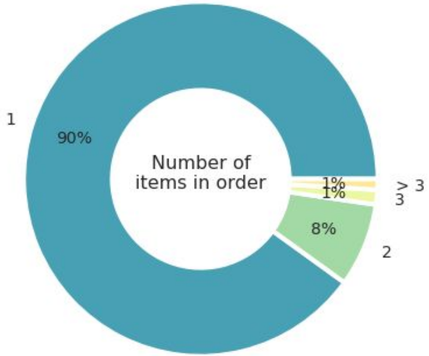


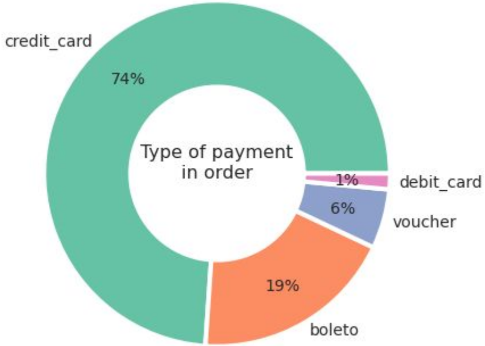


Nettoyage + feature engineering + exploration

Quelques stats générales avant merge

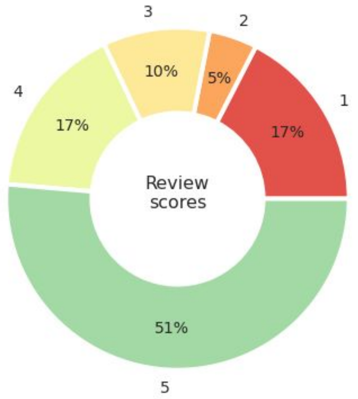


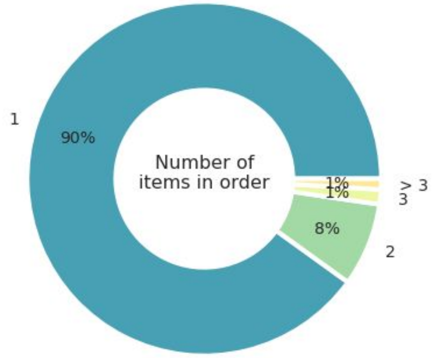


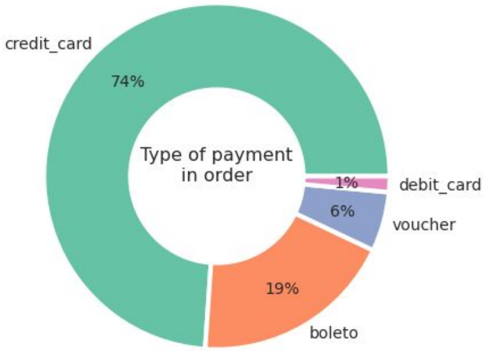


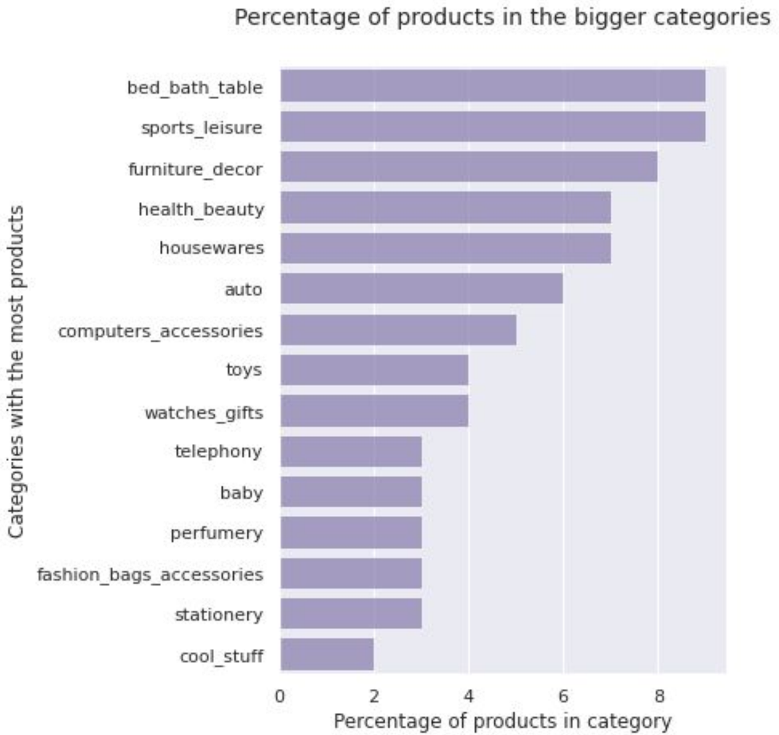
Nettoyage + feature engineering + exploration

Quelques stats générales avant merge



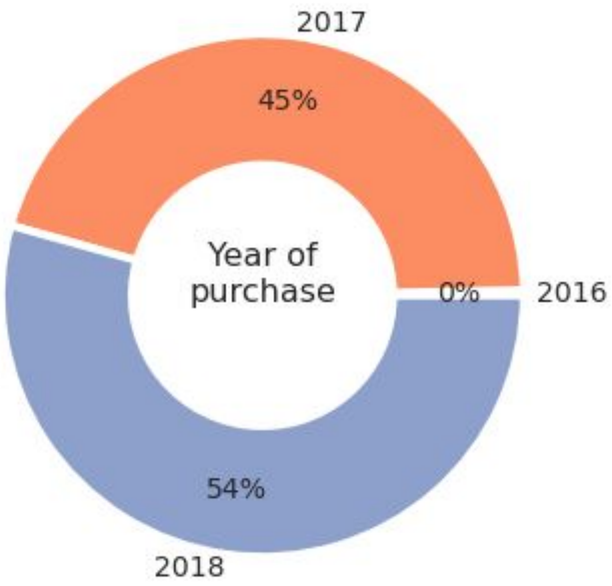


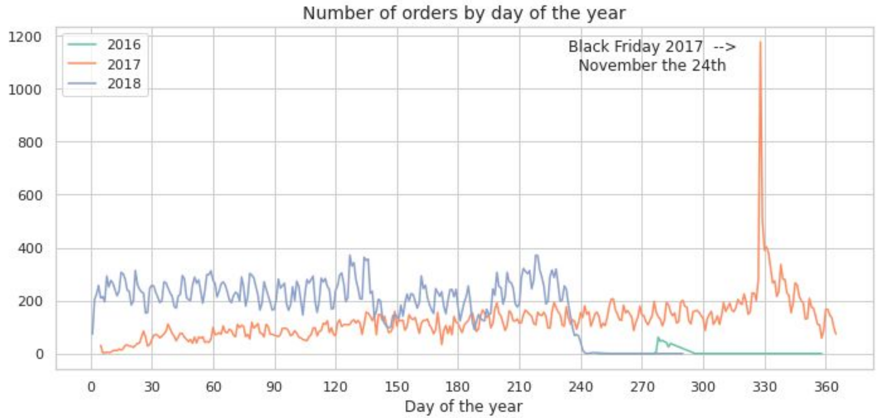




Nettoyage + feature engineering + exploration

Quelques stats générales avant merge





Nettoyage + feature engineering + exploration

Quelques commentaires sur merge: choix

Nettoyage + feature engineering + exploration

Quelques commentaires sur merge: choix

```
dataframe: (9/9)
  orders
columns :
   order_id , customer_id ,
   order_status , order_purchase_timestamp ,
   order_approved_at , order_delivered_carrier_date ,
   order_delivered_customer_date , order_estimated_delivery_date ,
   order_estimated_delivery_date
```

Nettoyage + feature engineering + exploration

Quelques commentaires sur merge: choix

```
dataframe: (9/9)
 orders
columns :
   order_id , customer_id ,
   order_status , order_purchase_timestamp ,
   order_approved_at , order_delivered_carrier_date ,
   order_delivered_customer_date , order_estimated_delivery_date ,
   order_estimated_delivery_date
```

```
dataframe: (4/9)
 customers
columns :
   customer_id , customer_unique_id ,
   customer_zip_code_prefix , customer_city ,
   customer_state
```

Nettoyage + feature engineering + exploration 2

Quelques commentaires sur merge: choix

dataframe: (9/9)
orders
columns :
order_id , customer_id ,
order_status , order_purchase_timestamp ,
order_approved_at , order_delivered_carrier_date ,
order_delivered_customer_date , order_estimated_delivery_date ,
order_estimated_delivery_date

dataframe: (4/9)
customers
columns :
customer_id , customer_unique_id ,
customer_zip_code_prefix , customer_city ,
customer_state

dataframe: (2/9)
order-reviews
columns :
review_id , order_id ,
review_score , review_comment_title ,
review_comment_message , review_creation_date ,
review_answer_timestamp

stratégie:
moyenne

Nettoyage + feature engineering + exploration

2

Quelques commentaires sur merge: choix

```
dataframe: (9/9)
  orders
columns :
  order_id , customer_id ,
  order_status , order_purchase_timestamp ,
  order_approved_at , order_delivered_carrier_date ,
  order_delivered_customer_date , order_estimated_delivery_date ,
  order_estimated_delivery_date
```

```
dataframe: (4/9)
  customers
columns :
  customer_id , customer_unique_id ,
  customer_zip_code_prefix , customer_city ,
  customer_state
```

```
dataframe: (2/9)
  order-reviews
columns :
  review_id , order_id ,
  review_score , review_comment_title ,
  review_comment_message , review_creation_date ,
  review_answer_timestamp
```

stratégie:
moyenne

```
dataframe: (7/9)
  order-items
columns :
  order_id , order_item_id ,
  product_id , seller_id ,
  shipping_limit_date , price ,
  freight_value
```

stratégie:

- # total items
- item le + cher

Nettoyage + feature engineering + exploration

Quelques commentaires sur merge: choix

dataframe: (9/9)
orders
columns :
order_id , customer_id ,
order_status , order_purchase_timestamp ,
order_approved_at , order_delivered_carrier_date ,
order_delivered_customer_date , order_estimated_delivery_date ,
order_estimated_delivery_date

dataframe: (4/9)
customers
columns :
customer_id , customer_unique_id ,
customer_zip_code_prefix , customer_city ,
customer_state

dataframe: (2/9)
order-reviews
columns :
review_id , order_id ,
review_score , review_comment_title ,
review_comment_message , review_creation_date ,
review_answer_timestamp

stratégie:
moyenne

dataframe: (7/9)
order-items
columns :
order_id , order_item_id ,
product_id , seller_id ,
shipping_limit_date , price ,
freight_value

stratégie:
- # total items
- item le + cher

dataframe: (8/9)
order-payments
columns :
order_id , payment_sequential ,
payment_type , payment_installments ,
payment_value

stratégie:
- $ total commande
- mode paiement + cher

Nettoyage + feature engineering + exploration 2

Quelques commentaires sur merge: choix

dataframe: (9/9)
orders
columns :
order_id , customer_id ,
order_status , order_purchase_timestamp ,
order_approved_at , order_delivered_carrier_date ,
order_delivered_customer_date , order_estimated_delivery_date ,
order_estimated_delivery_date

dataframe: (4/9)
customers
columns :
customer_id , customer_unique_id ,
customer_zip_code_prefix , customer_city ,
customer_state

dataframe: (2/9)
order-reviews
columns :
review_id , order_id ,
review_score , review_comment_title ,
review_comment_message , review_creation_date ,
review_answer_timestamp

stratégie:
moyenne

dataframe: (7/9)
order-items
columns :
order_id , order_item_id ,
product_id , seller_id ,
shipping_limit_date , price ,
freight_value

stratégie:
- # total items
- item le + cher

dataframe: (8/9)
order-payments
columns :
order_id , payment_sequential ,
payment_type , payment_installments ,
payment_value

stratégie:
- $ total commande
- mode paiement + cher

dataframe: (5/9)
products
columns :
product_id , product_category_name ,
product_name_lenght , product_description_
product_photos_qty , product_weight_g ,
product_length_cm , product_height_cm ,
product_width_cm

Nettoyage + feature engineering + exploration

Quelques commentaires sur merge: choix

dataframe: (9/9)
orders
columns :
order_id , customer_id ,
order_status , order_purchase_timestamp ,
order_approved_at , order_delivered_carrier_date ,
order_delivered_customer_date , order_estimated_delivery_date ,
order_estimated_delivery_date

dataframe: (4/9)
customers
columns :
customer_id , customer_unique_id ,
customer_zip_code_prefix , customer_city ,
customer_state

dataframe: (2/9)
order-reviews
columns :
review_id , order_id ,
review_score , review_comment_title ,
review_comment_message , review_creation_date ,
review_answer_timestamp

stratégie:
moyenne

dataframe: (7/9)
order-items
columns :
order_id , order_item_id ,
product_id , seller_id ,
shipping_limit_date , price ,
freight_value

stratégie:
- # total items
- item le + cher

dataframe: (8/9)
order-payments
columns :
order_id , payment_sequential ,
payment_type , payment_installments ,
payment_value

stratégie:
- $ total commande
- mode paiement + cher

dataframe: (5/9)
products
columns :
product_id , product_category_name ,
product_name_lenght , product_description_
product_photos_qty , product_weight_g ,
product_length_cm , product_height_cm ,
product_width_cm

dataframe: (1/9)
geolocation
columns :
geolocation_zip_code_prefix , geolocation_lat
geolocation_lng , geolocation_city ,
geolocation_state

stratégie:
médiane pour lat/long

Nettoyage + feature engineering + exploration

Quelques commentaires sur merge: choix

dataframe: (9/9)
orders
columns :
order_id , customer_id ,
order_status , order_purchase_timestamp ,
order_approved_at , order_delivered_carrier_date ,
order_delivered_customer_date , order_estimated_delivery_date ,
order_estimated_delivery_date

dataframe: (4/9)
customers
columns :
customer_id , customer_unique_id ,
customer_zip_code_prefix , customer_city ,
customer_state

dataframe: (2/9)
order-reviews
columns :
review_id , order_id ,
review_score , review_comment_title ,
review_comment_message , review_creation_date ,
review_answer_timestamp

stratégie:
moyenne

dataframe: (7/9)
order-items
columns :
order_id , order_item_id ,
product_id , seller_id ,
shipping_limit_date , price ,
freight_value

stratégie:
- # total items
- item le + cher

dataframe: (8/9)
order-payments
columns :
order_id , payment_sequential ,
payment_type , payment_installments ,
payment_value

stratégie:
- $ total commande
- mode paiement + cher

dataframe: (5/9)
products
columns :
product_id , product_category_name ,
product_name_lenght , product_description_
product_photos_qty , product_weight_g ,
product_length_cm , product_height_cm ,
product_width_cm

dataframe: (1/9)
geolocation
columns :
geolocation_zip_code_prefix , geolocation_lat
geolocation_lng , geolocation_city ,
geolocation_state

stratégie:
médiane pour lat/long

Création RFM

Nettoyage + feature engineering + exploration

2

Quelques commentaires sur merge: choix

```
dataframe: (9/9)
 orders
columns :
  order_id , customer_id ,
  order_status , order_purchase_timestamp ,
  order_approved_at , order_delivered_carrier_date ,
  order_delivered_customer_date , order_estimated_delivery_date ,
  order_estimated_delivery_date
```

```
dataframe: (4/9)
 customers
columns :
  customer_id , customer_unique_id ,
  customer_zip_code_prefix , customer_city ,
  customer_state
```

```
dataframe: (2/9)
 order-reviews
columns :
  review_id , order_id ,
  review_score , review_comment_title ,
  review_comment_message , review_creation_date ,
  review_answer_timestamp
```

stratégie:
moyenne

```
dataframe: (7/9)
 order-items
columns :
  order_id , order_item_id ,
  product_id , seller_id ,
  shipping_limit_date , price ,
  freight_value
```

stratégie:

- # total items
- item le + cher

```
dataframe: (8/9)
 order-payments
columns :
  order_id , payment_sequential ,
  payment_type , payment_installments ,
  payment_value
```

stratégie:

- $ total commande
- mode paiement + cher

```
dataframe: (5/9)
 products
columns :
  product_id , product_category_name ,
  product_name_lenght , product_description_
  product_photos_qty , product_weight_g ,
  product_length_cm , product_height_cm ,
  product_width_cm
```

```
dataframe: (1/9)
 geolocation
columns :
  geolocation_zip_code_prefix , geolocation_lat
  geolocation_lng , geolocation_city ,
  geolocation_state
```

stratégie:
médiane pour lat/long

Création RFM

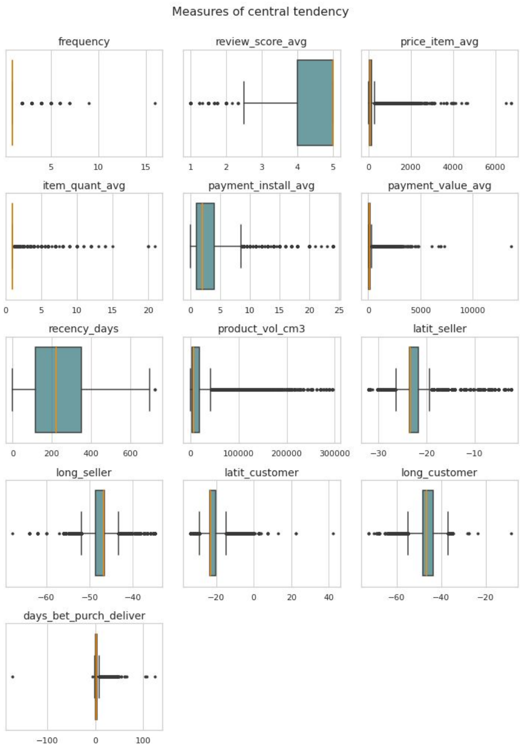
group by:
`customer_unique_id`

stratégie:

- moyenne pour $
- commande + chère

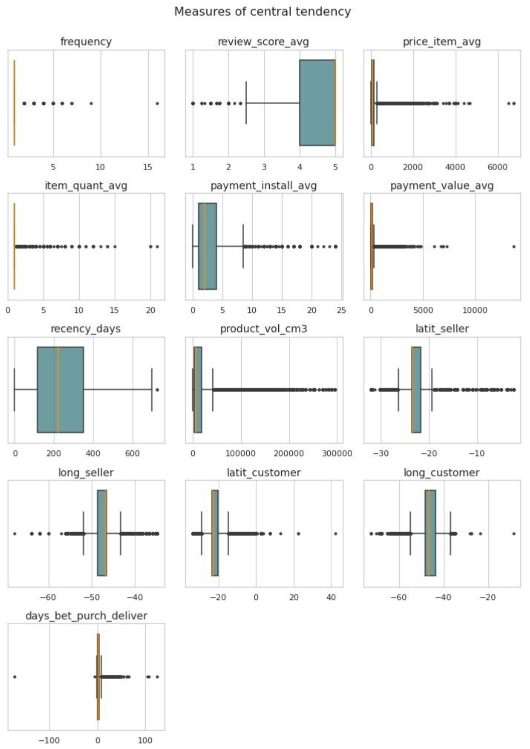
Nettoyage + feature engineering + exploration 2

Nettoyage et création variables

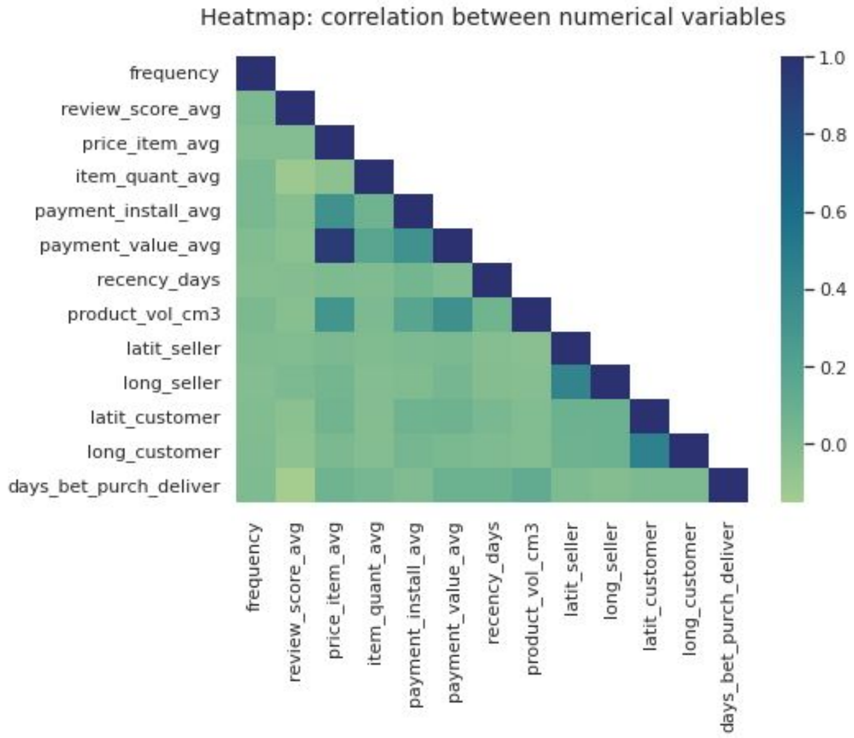


Nettoyage + feature engineering + exploration

Nettoyage et création variables



Corrélations



Nettoyage + feature engineering + exploration

Taille définitive du dataframe:

95 245 lignes

26 colonnes

3. Modélisation

Modélisation

KMeans

Modélisation

KMeans

DBSCAN

Clustering Hiérarchique

Modélisation

3 essais : 3 différents ensembles de features

KMeans

Modélisation

3 essais : 3 différents ensembles de features

KMeans

RFM:

- Recence
- Fréquence
- Montant

Modélisation

3

3 essais : 3 différents ensembles de features

KMeans

RFM:

- Recence
- Fréquence
- Montant

RFM + review score

Modélisation

3 essais : 3 différents ensembles de features

KMeans

RFM:

- Recence
- Fréquence
- Montant

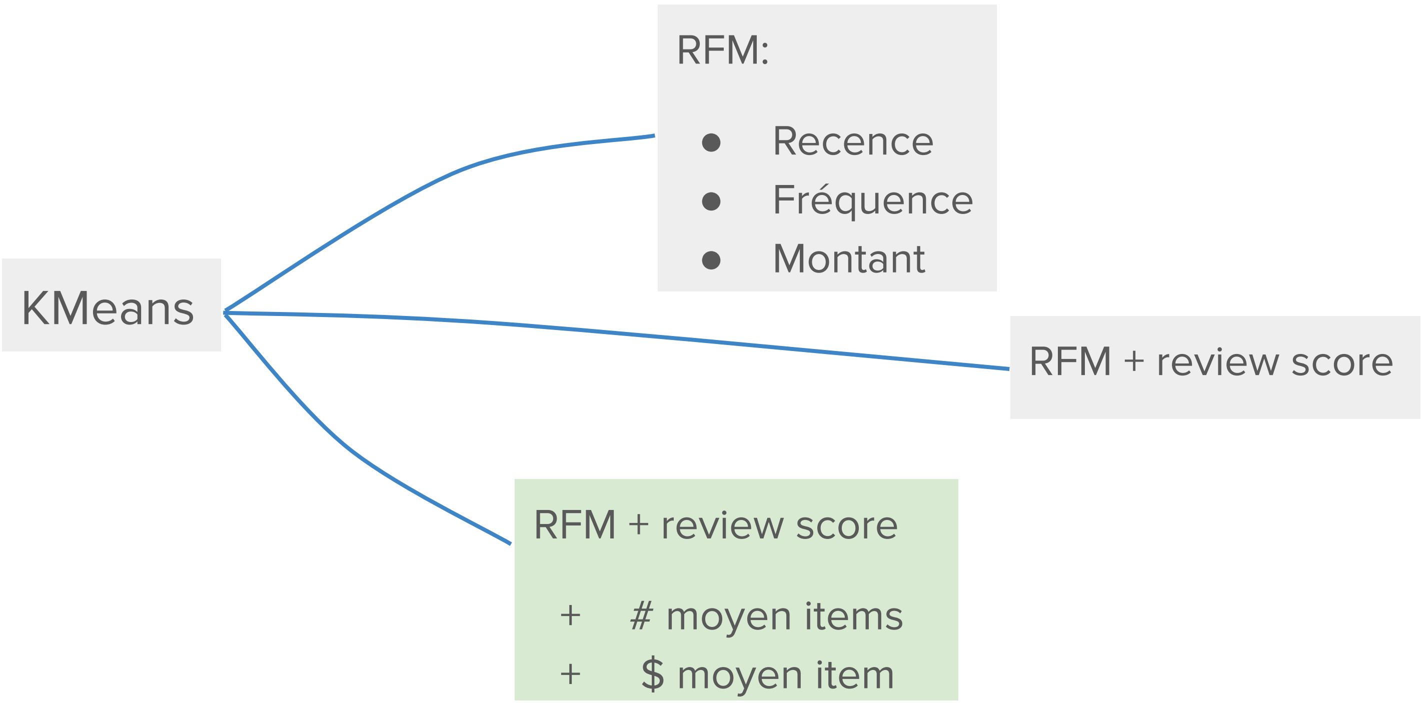
RFM + review score

RFM + review score

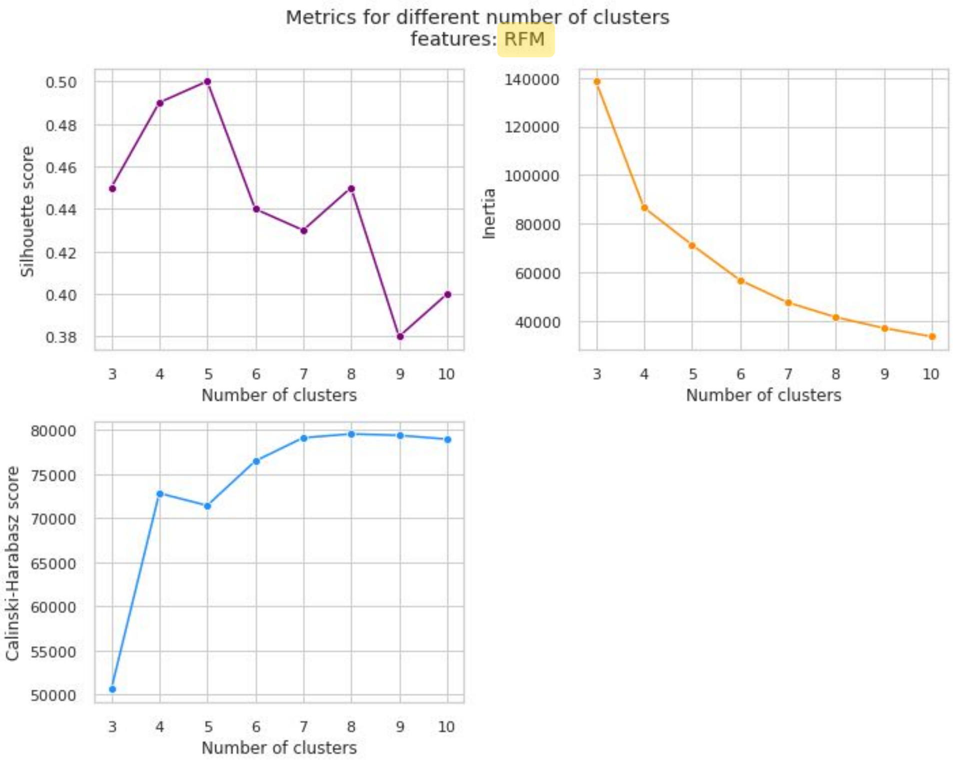
+ # moyen items
+ $ moyen item

Modélisation

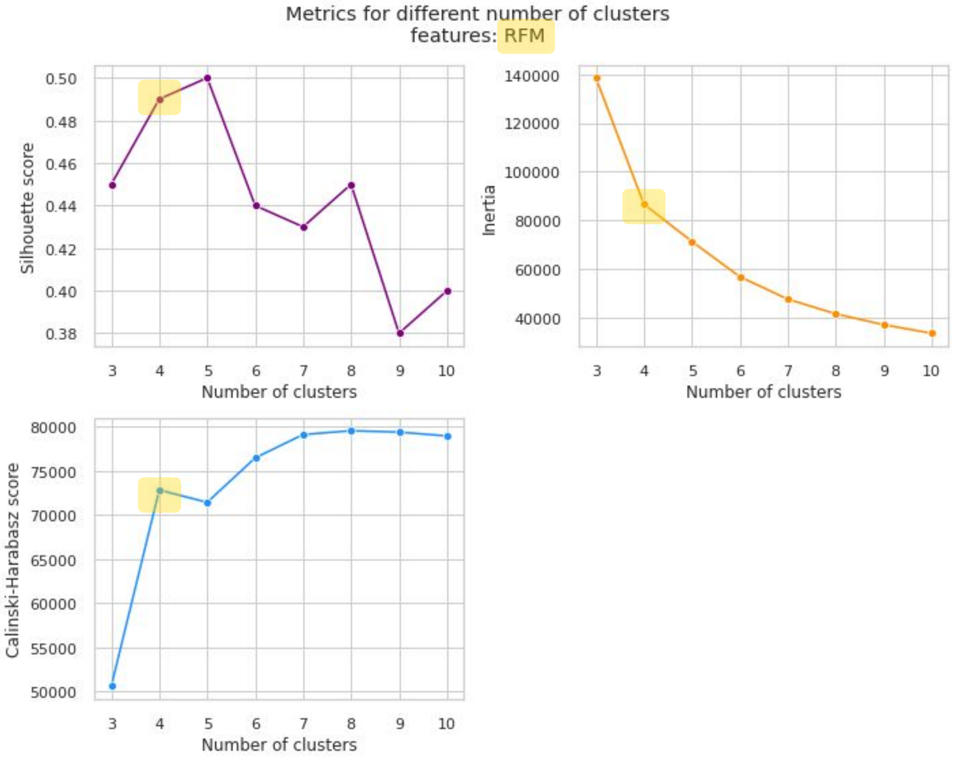
3 essais : 3 différents ensembles de features



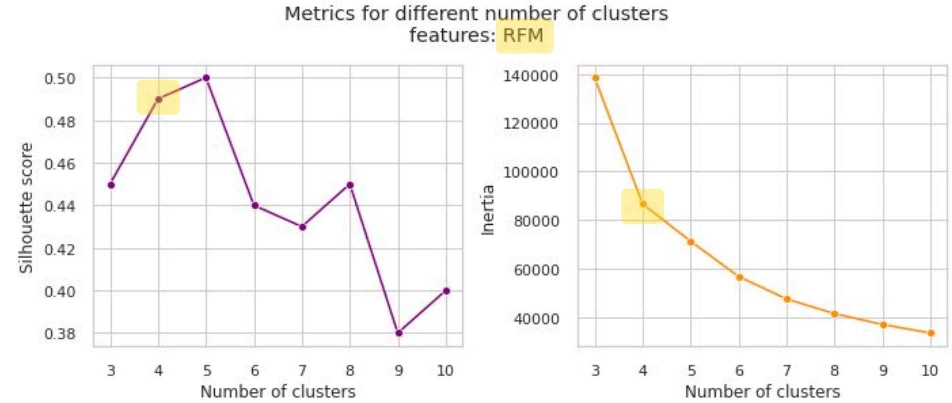
Modélisation KMeans. Features: RFM

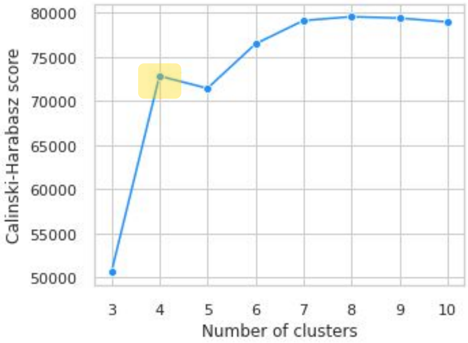


Modélisation KMeans. Features: RFM



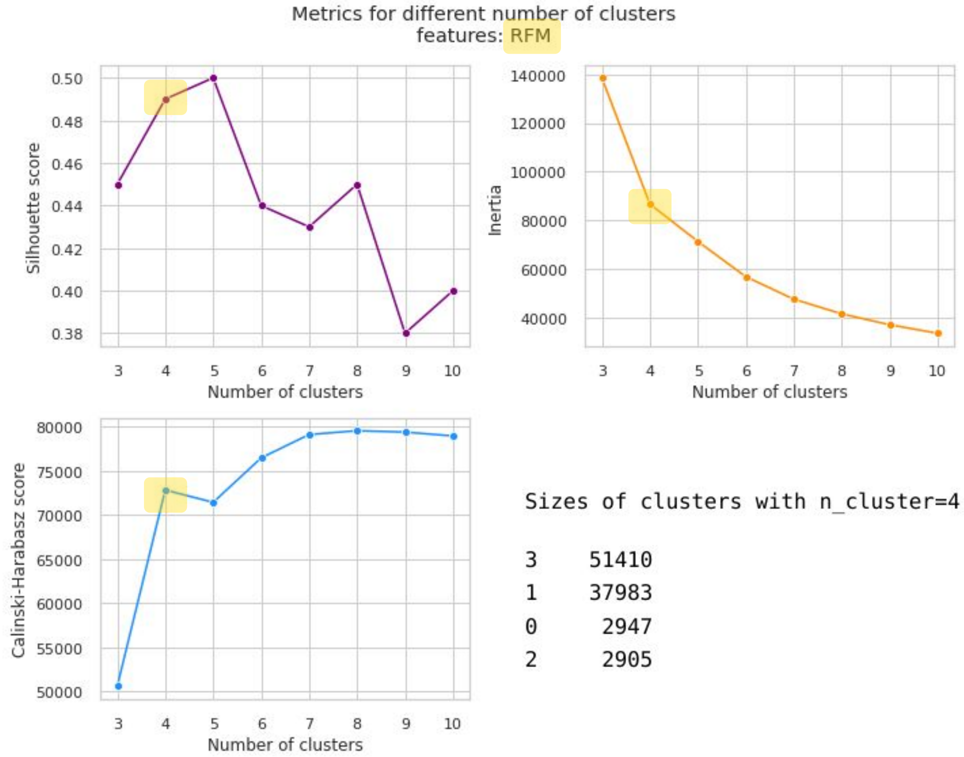
Modélisation KMeans. Features: RFM

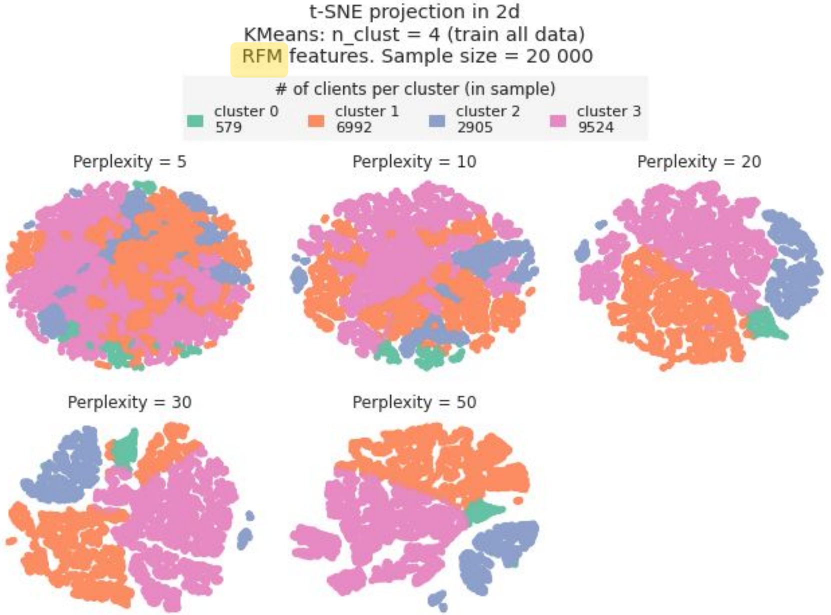





```
Sizes of clusters with n_cluster=4:

3    51410
1    37983
0     2947
2     2905
```

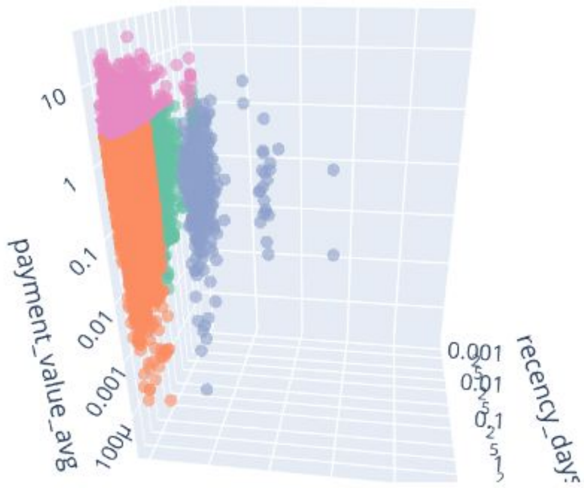
Modélisation KMeans. Features: RFM



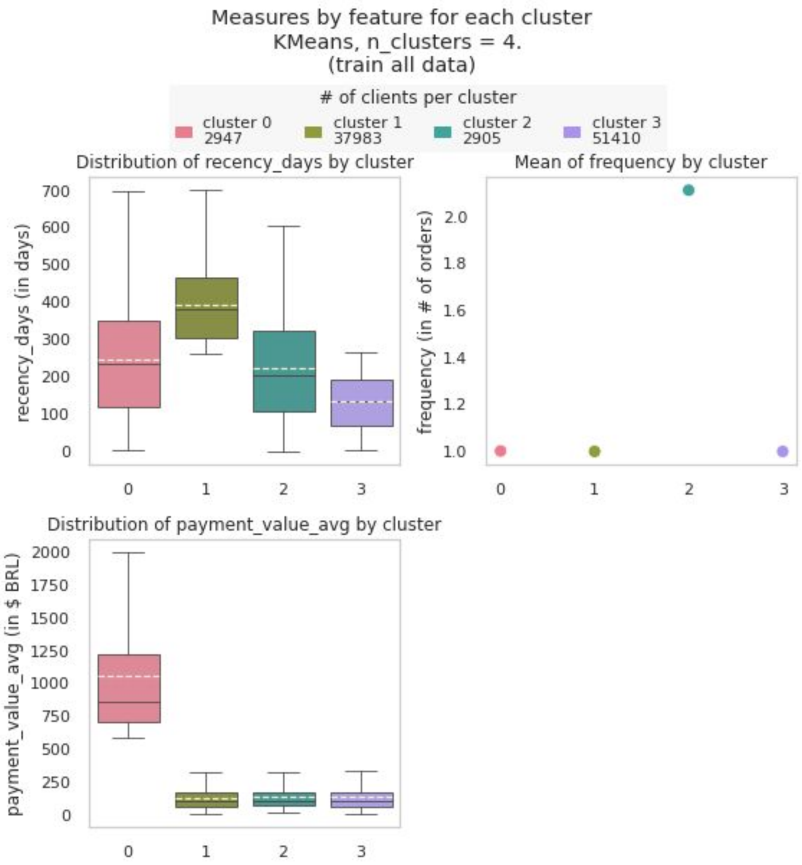



```
Sizes of clusters with n_cluster=4:

3    51410
1    37983
0     2947
2     2905
```

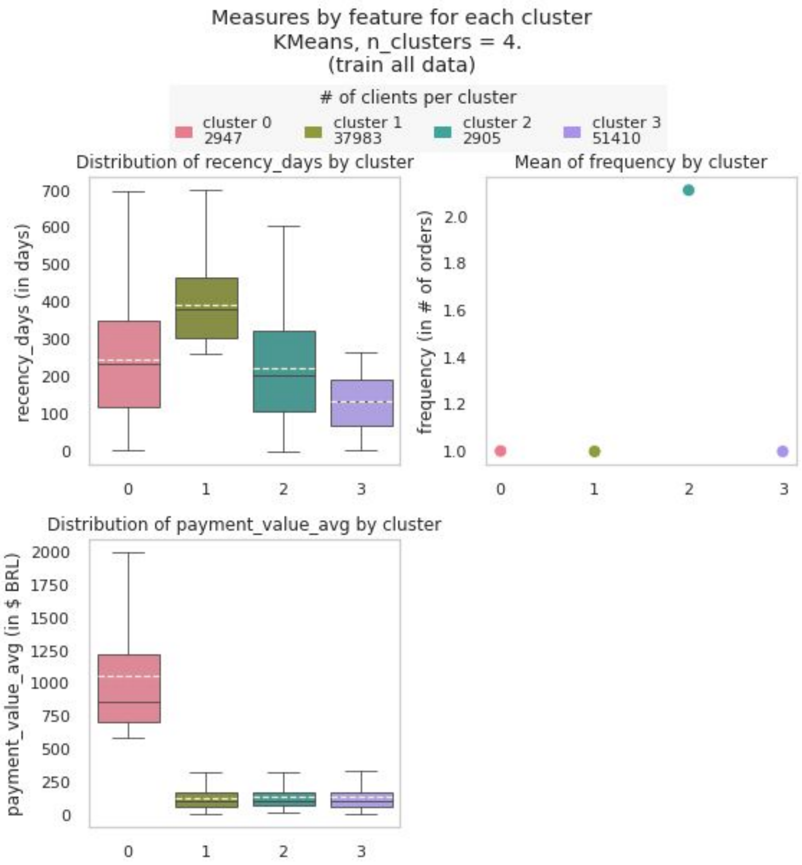
KMeans: n_clust = 4 (train all data)
features: RFM



Modélisation KMeans. Features: RFM



Modélisation KMeans. Features: RFM



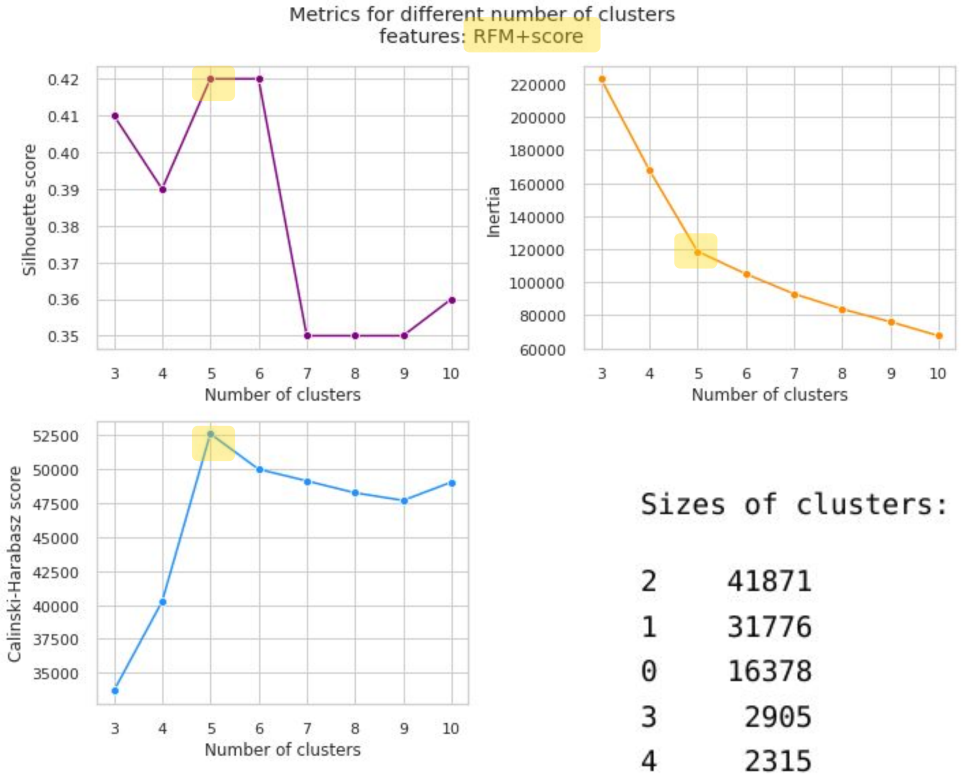
Une 1ère segmentation

Etiquette cluster	**Profil client**
3	Acheté récemment, client moyen
1	Non-régulier, acheté il y a **longtemps**
0	**hauts dépenses**
2	Clients **fréquents**

Modélisation

KMeans. Features: RFM + score

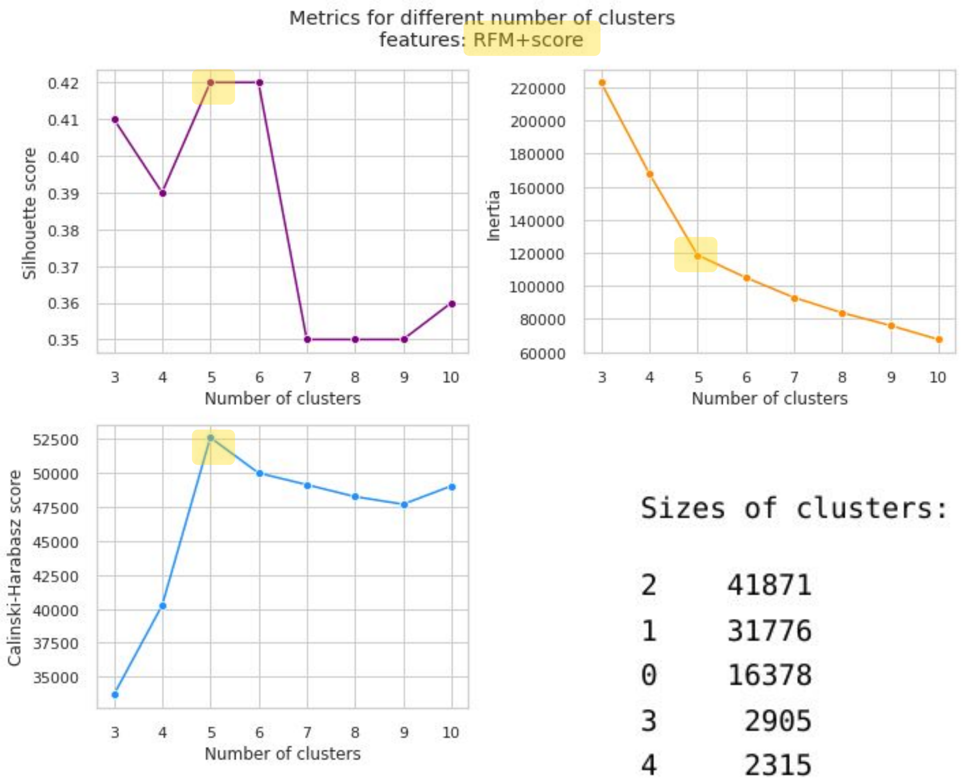
3



Sizes of clusters:

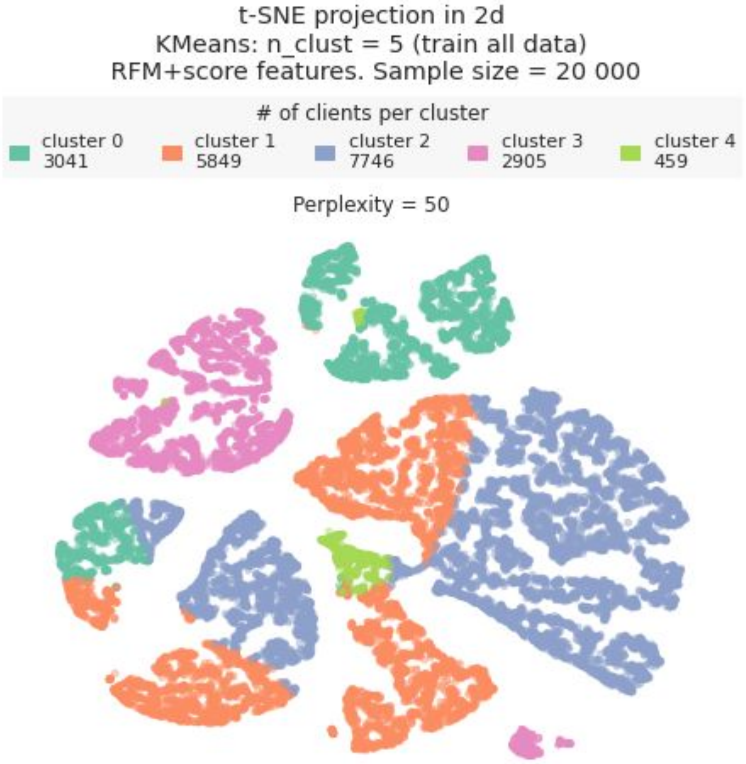
2	41871
1	31776
0	16378
3	2905
4	2315

Modélisation KMeans. Features: RFM + score



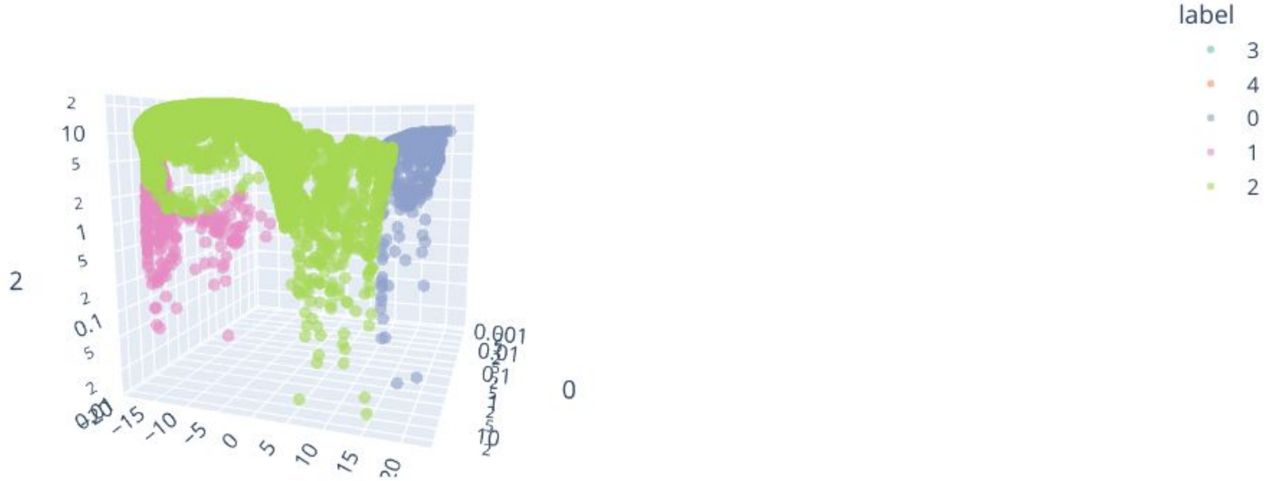
Sizes of clusters:

2	41871
1	31776
0	16378
3	2905
4	2315

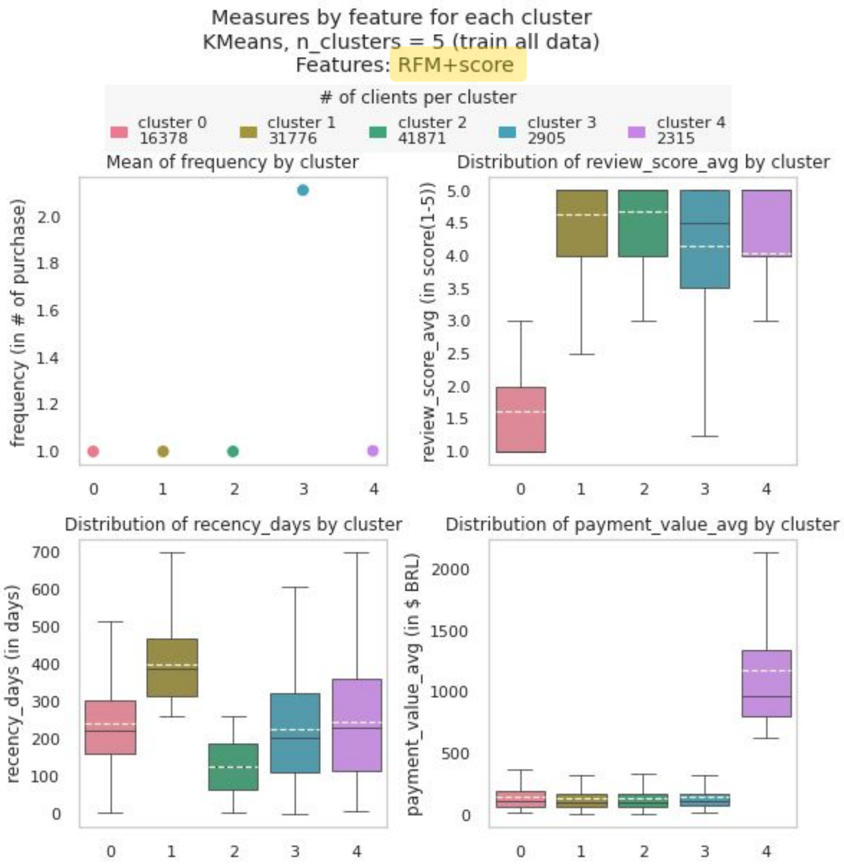


KMeans. Features: RFM + score

t-SNE projection 3D
KMeans n_clust = 5 (train all data)
features: RFM+score

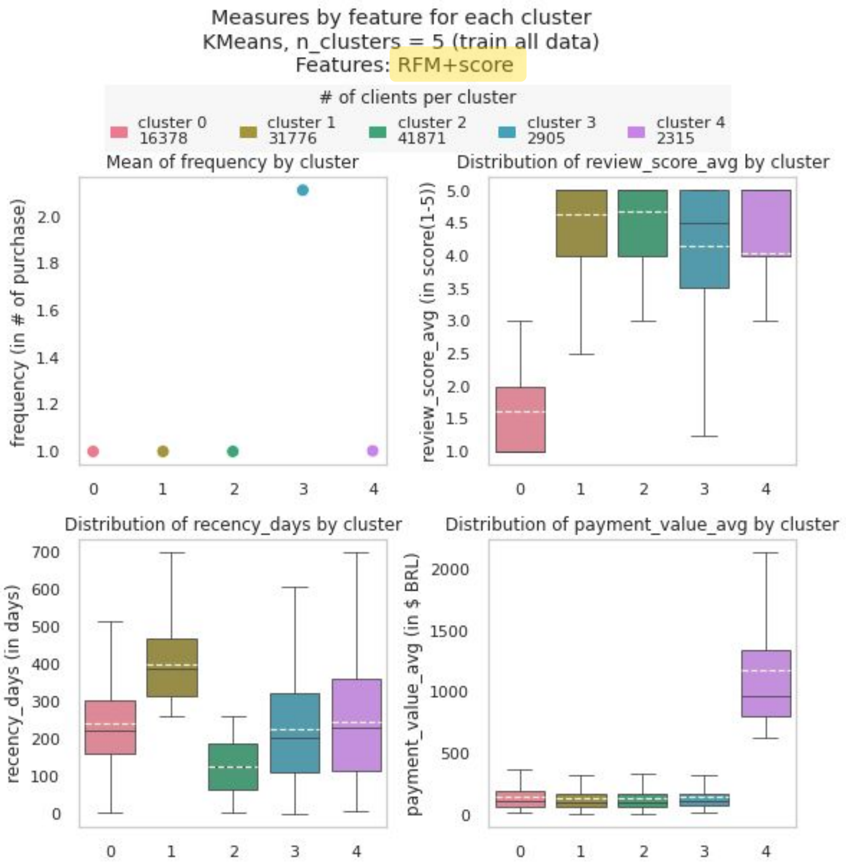


Modélisation KMeans. Features: RFM + score



Modélisation

KMeans. Features: RFM + score

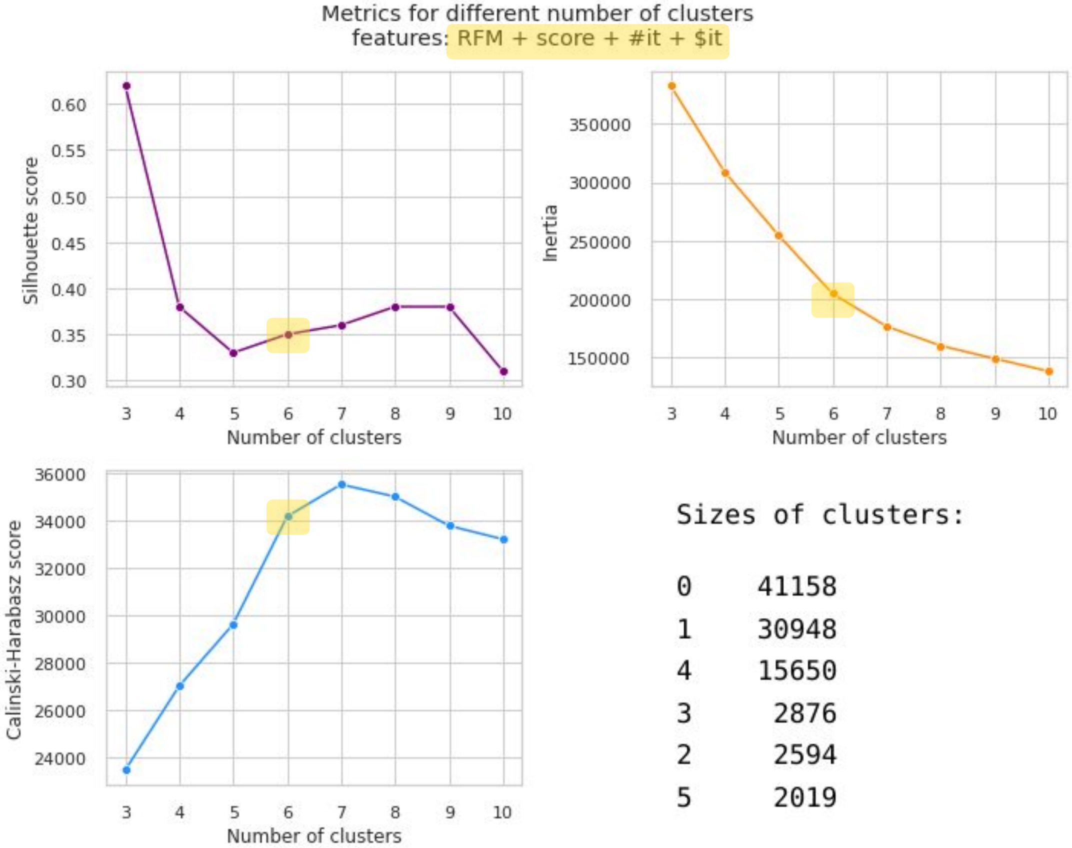


Une 2ème segmentation

Etiquette cluster	Profil client
2	Acheté récemment, **client moyen**
1	Non-régulier, acheté il y a **longtemps, content.**
4	**hautes dépenses**
3	Clients **fréquents**
0	**Pas satisfait**

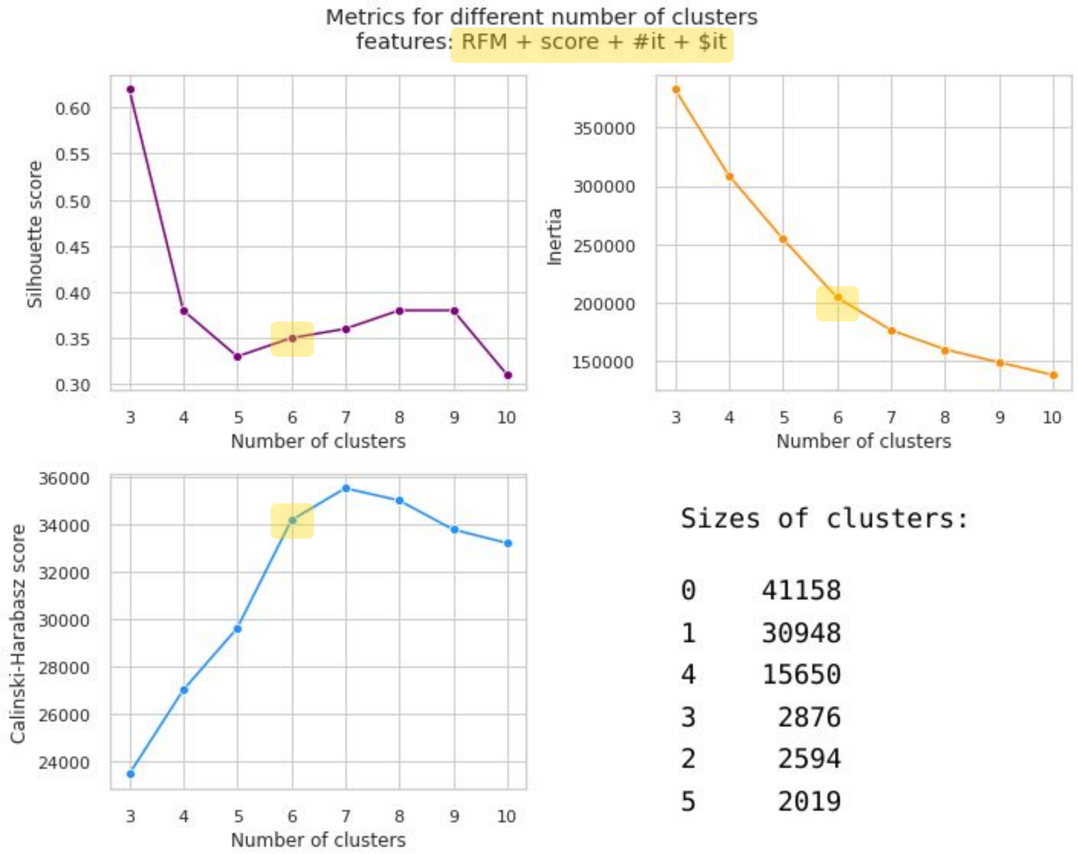
Modélisation

KMeans. Features: RFM + score + #item + $item



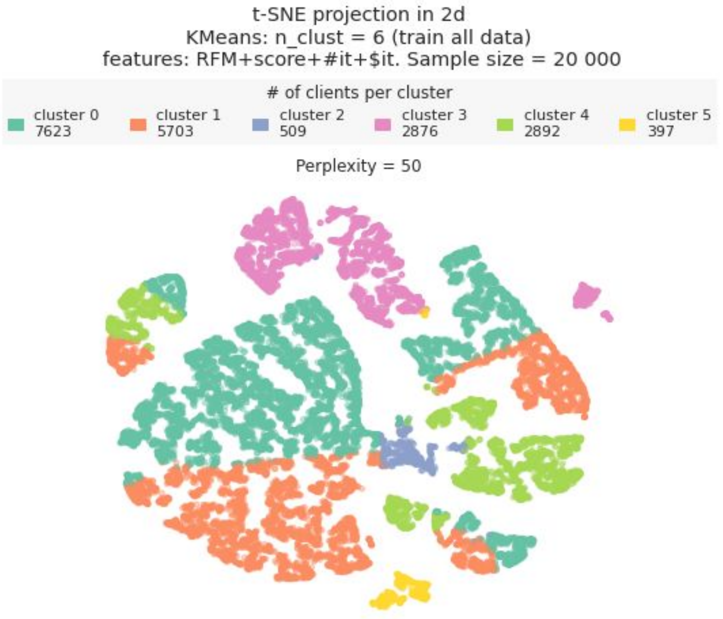

```
Sizes of clusters:

0    41158
1    30948
4    15650
3     2876
2     2594
5     2019
```

Modélisation KMeans. Features: RFM + score + #item + $item



Sizes of clusters:

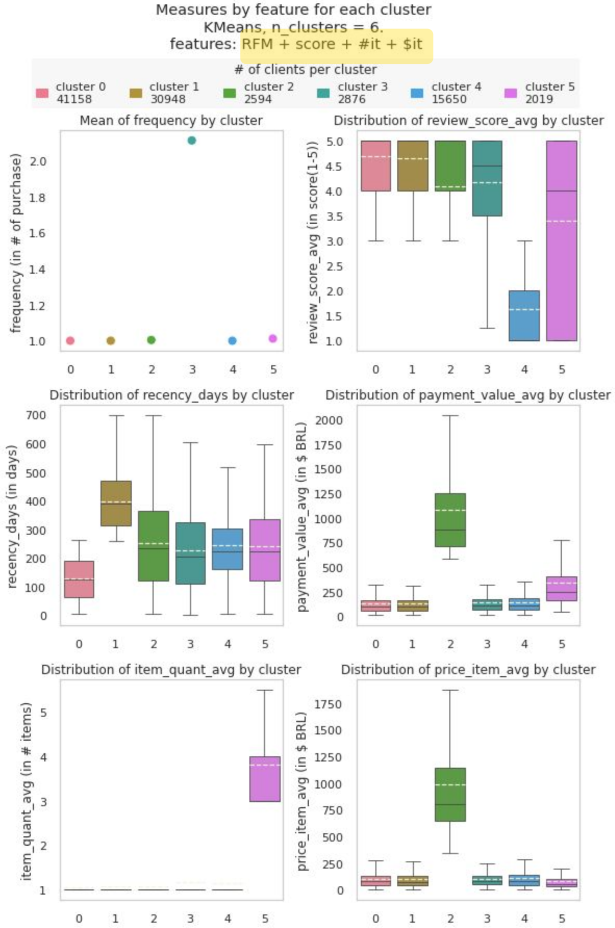
0	41158
1	30948
4	15650
3	2876
2	2594
5	2019



Modélisation KMeans. Features: RFM + score + #item + $item

t-SNE projection 3D
KMeans n_clust = 6 (train all data)
features: RFM+score+#it+$it.

Modélisation KMeans. Features: RFM + score + #item + $item

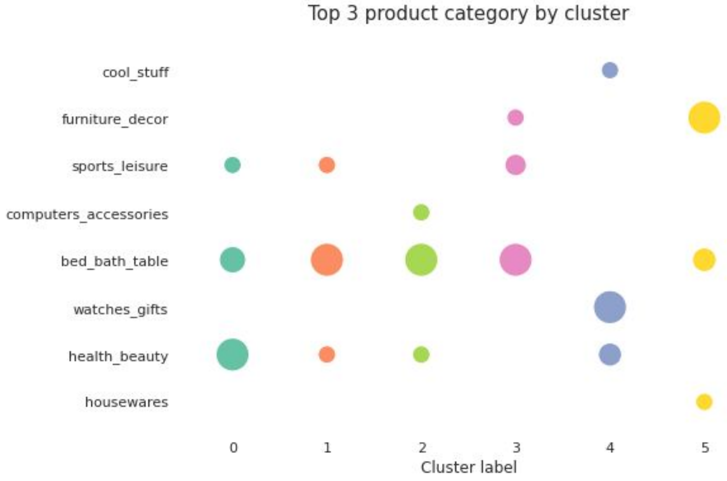


Segmentation client

Etiquette cluster	Profil client
0	Acheté récemment, **client moyen**
1	Non-régulier, acheté il y a **longtemps, content.**
2	💸 **Hautes dépenses**
3	Clients **fréquents**
5	🔢 **Grande quantité** d'items
4	🙁 **Pas satisfait**

Modélisation

KMeans. Features: RFM + score + #item + $item



Segmentation client

Etiquette cluster	**Profil client**
0	Acheté récemment, **client moyen**
1	Non-régulier, acheté il y a **longtemps, content.**
2	💸 **Hautes dépenses**
3	Clients **fréquents**
5	🔢 **Grande quantité** d'items
4	😕 **Pas satisfait**

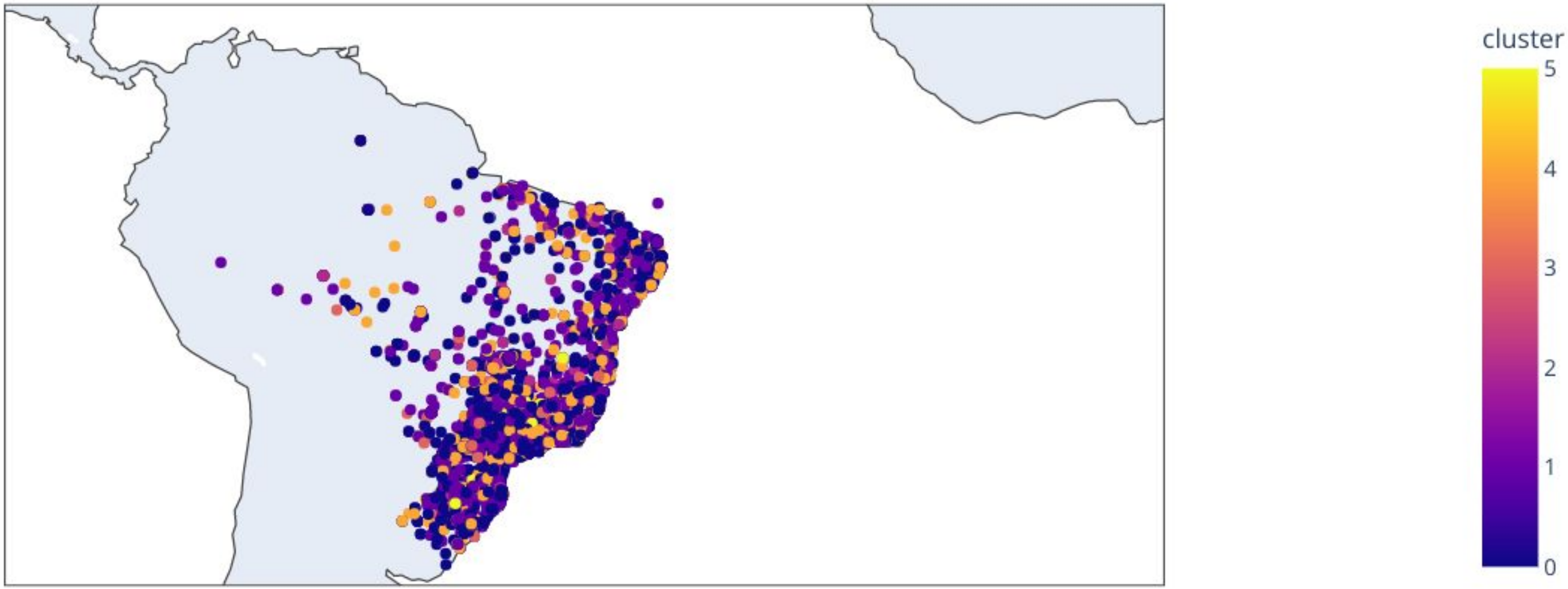
Modélisation

KMeans. Features: RFM + score + #item + $item

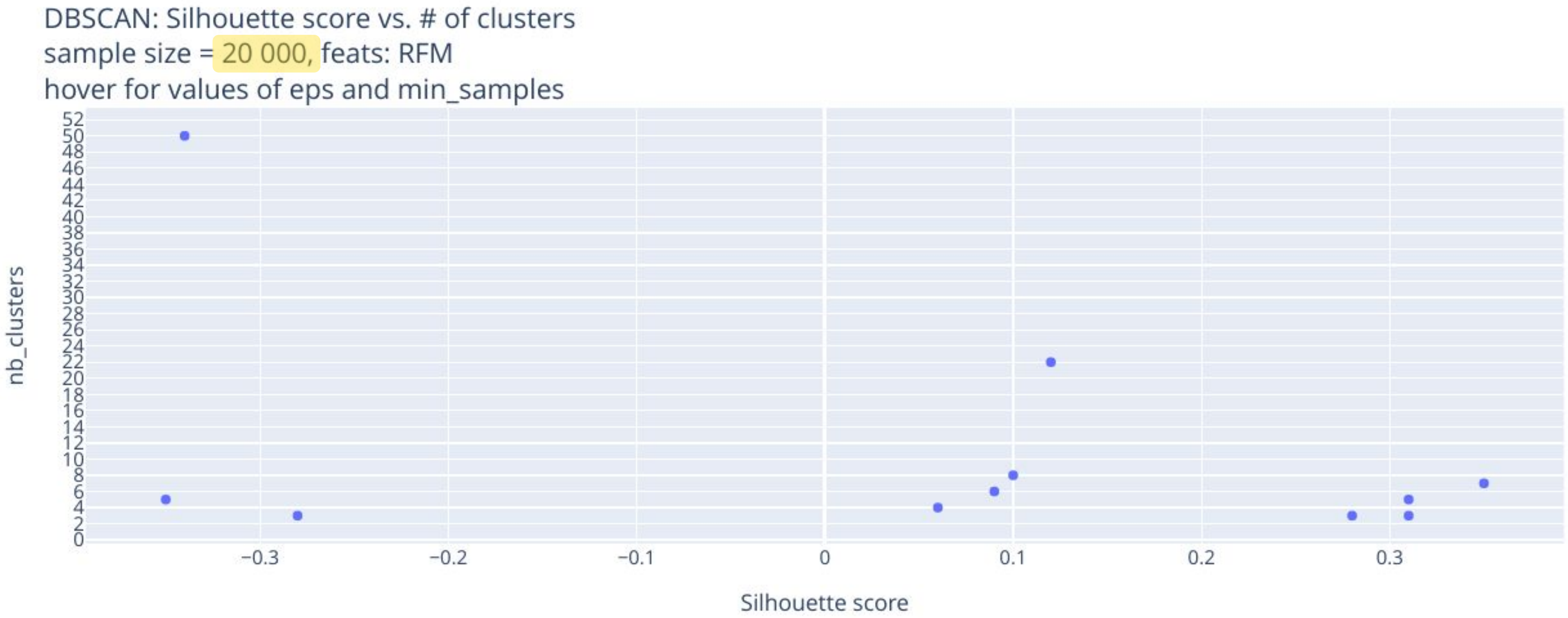
Segmentation client

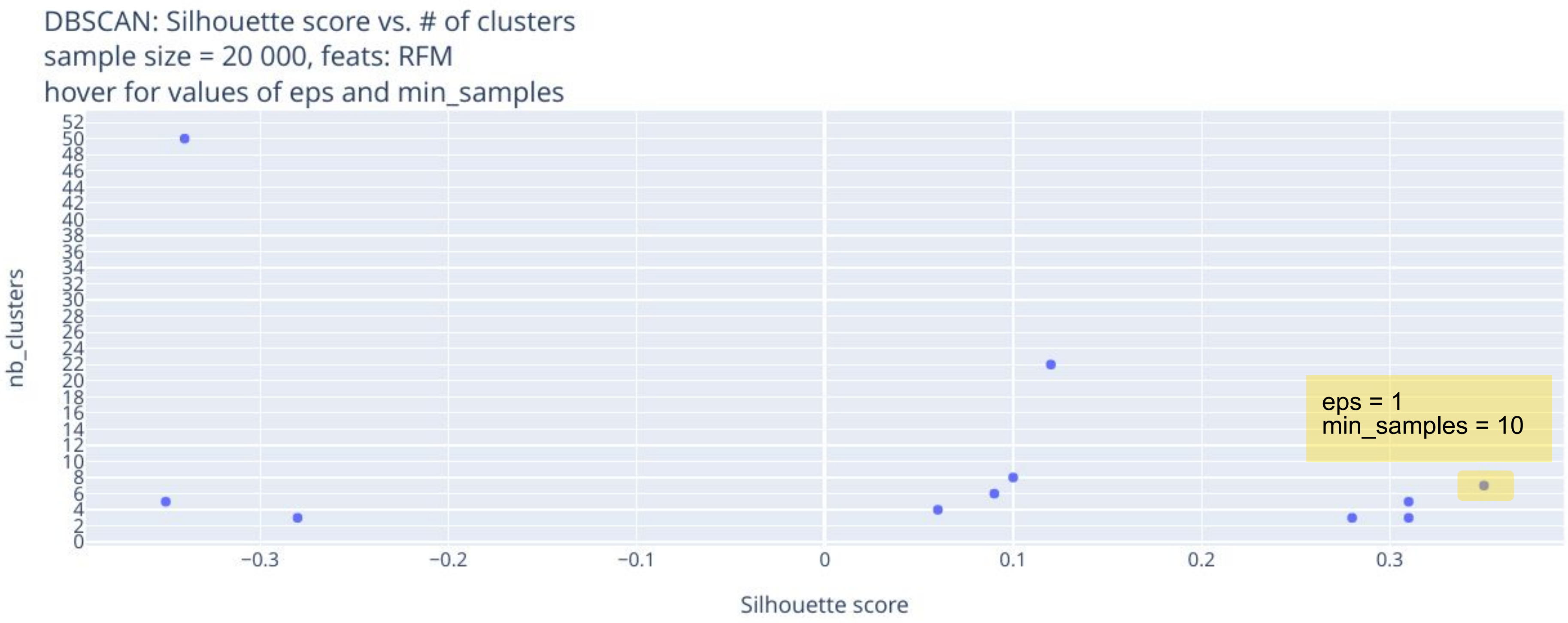
Etiquette cluster	Profil client
0	Acheté récemment, **client moyen**
1	Non-régulier, acheté il y a **longtemps, content.**
2	💸 **Hautes dépenses**
3	Clients **fréquents**
5	🔢 **Grande quantité** d'items
4	🙁 **Pas satisfait**

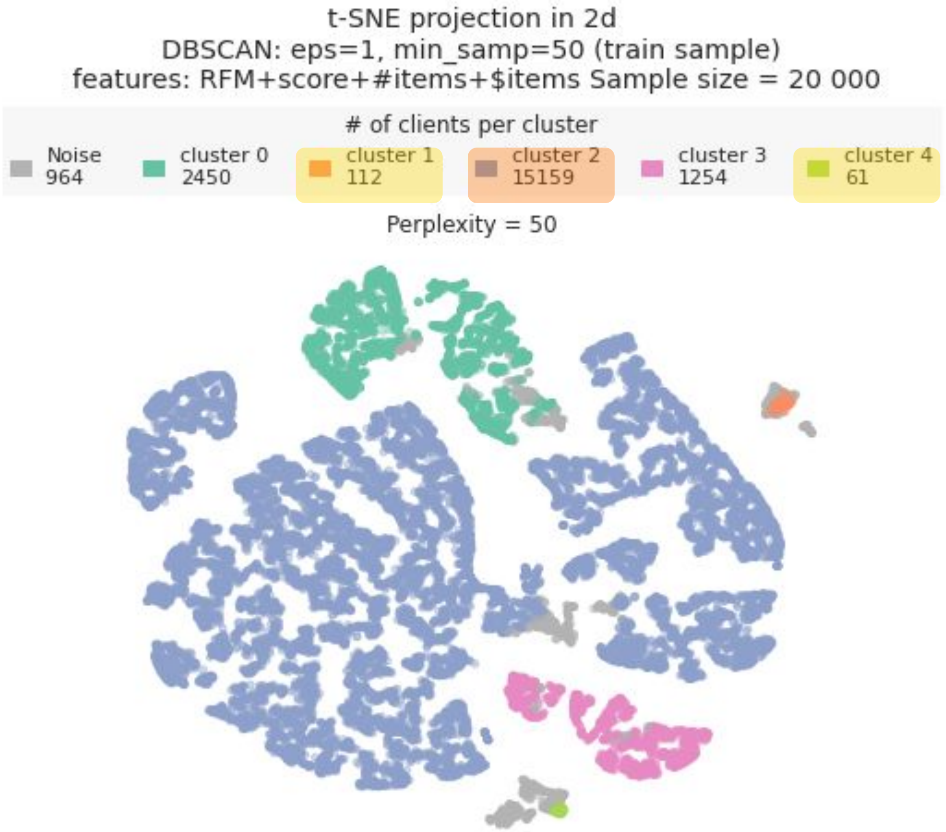
Répartition géographique



DBSCAN et Clustering hiérarchique


DBSCAN: Silhouette score vs. # of clusters
sample size = 20 000, feats: RFM
hover for values of eps and min_samples
nb_clusters
52
50
48
46
44
42
40
38
36
34
32
30
28
26
24
22
20
18
16
14
12
10
8
6
4
2
0
−0.3
−0.2
−0.1
0
0.1
0.2
0.3
Silhouette score


DBSCAN: Silhouette score vs. # of clusters
sample size = 20 000, feats: RFM
hover for values of eps and min_samples
nb_clusters
52
50
48
46
44
42
40
38
36
34
32
30
28
26
24
22
20
18
16
14
12
10
8
6
4
2
0
−0.3
−0.2
−0.1
0
0.1
0.2
0.3
Silhouette score
eps = 1
min_samples = 10



t-SNE projection in 2d
DBSCAN: eps=1, min_samp=50 (train sample)
features: RFM+score+#items+$items Sample size = 20 000
# of clients per cluster
Noise 964
cluster 0 2450
cluster 1 112
cluster 2 15159
cluster 3 1254
cluster 4 61
Perplexity = 50

Obstacles d'exécution

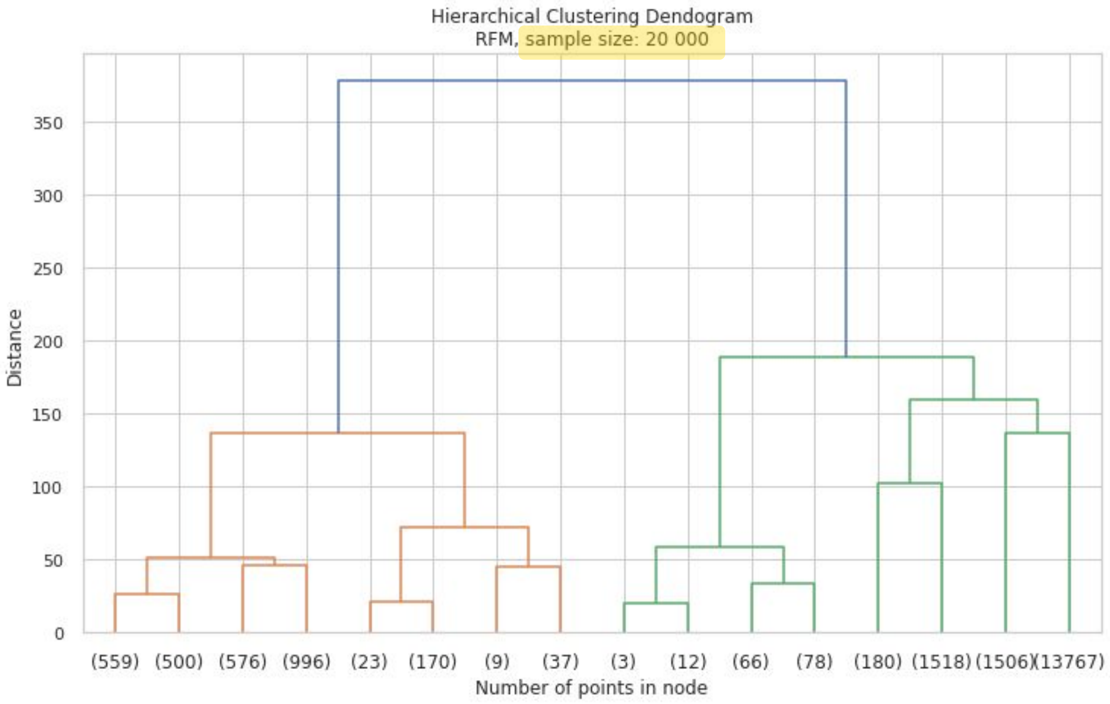
```python
# Kernel crashed when continuing this observations
cls = DBSCAN(
    eps= 0.5,
    min_samples= 100
).fit(X_norm)
```

[ ] Python

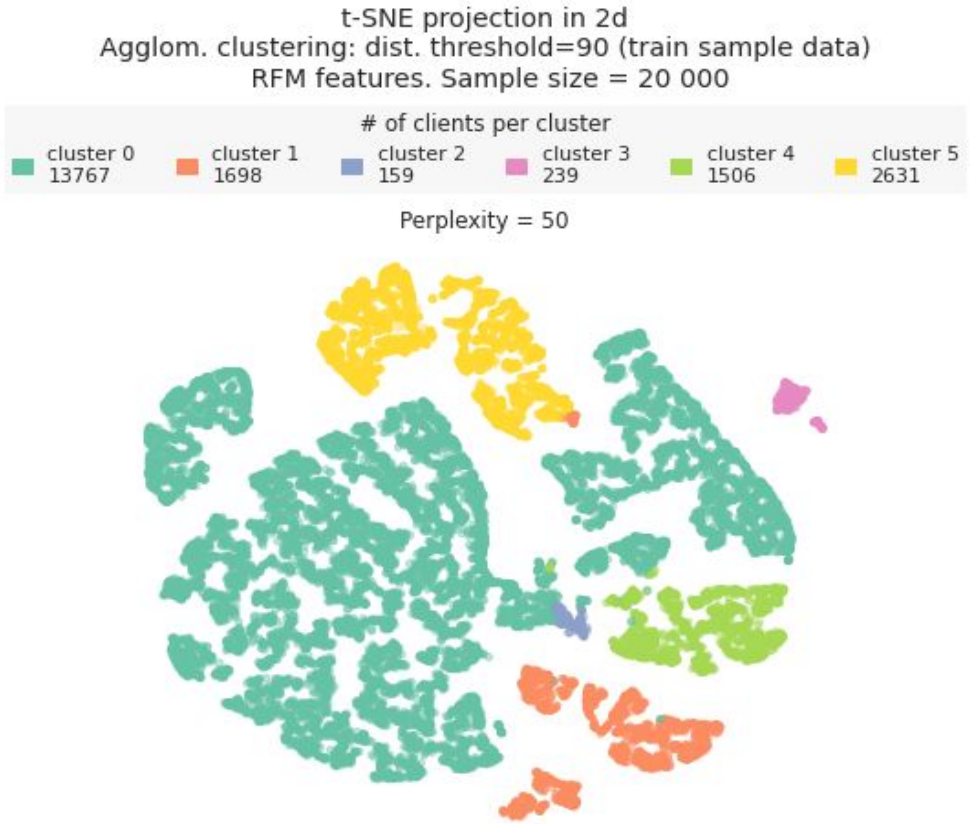
```
Canceled future for execute_request message before replies were done
```

```
The Kernel crashed while executing code in the the current cell or a previous cell. Please review the code in the cell(s) to identify a possible cause of the failure. Click here for more info. View Jupyter log for further details.
```

Clustering hiérarchique



Clustering hiérarchique



4. Simulation maintenance

Simulation maintenance

Stratégie pour évaluer stabilité

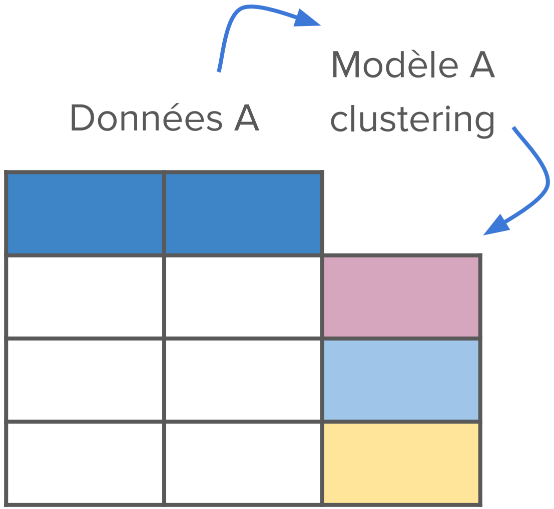
Simulation maintenance

Stratégie pour évaluer stabilité

Données A

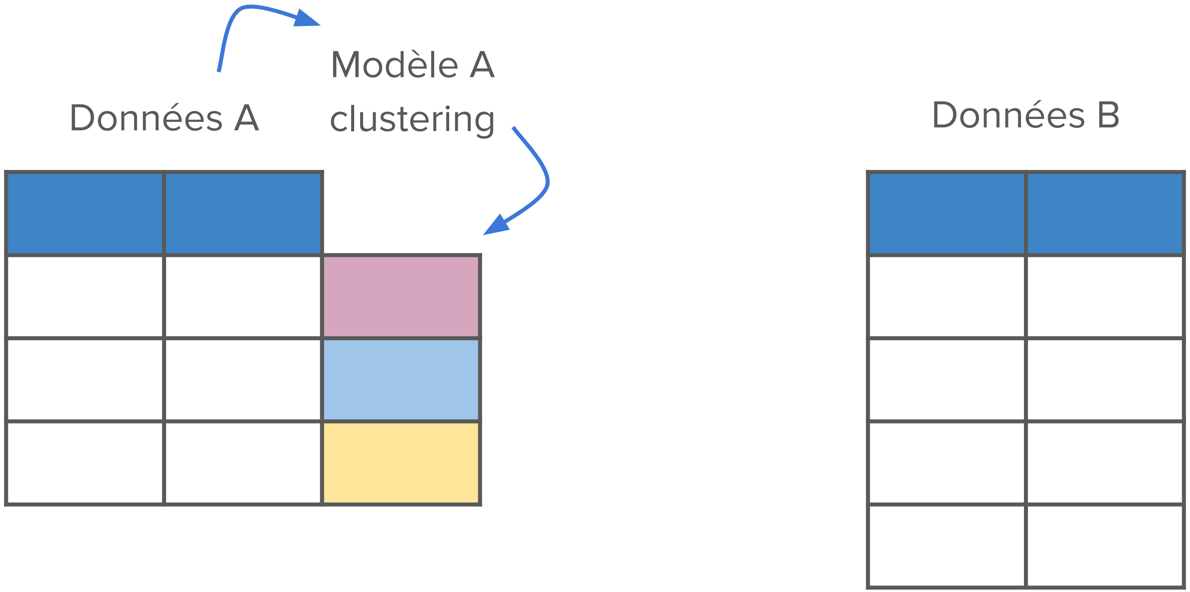
Simulation maintenance

Stratégie pour évaluer stabilité



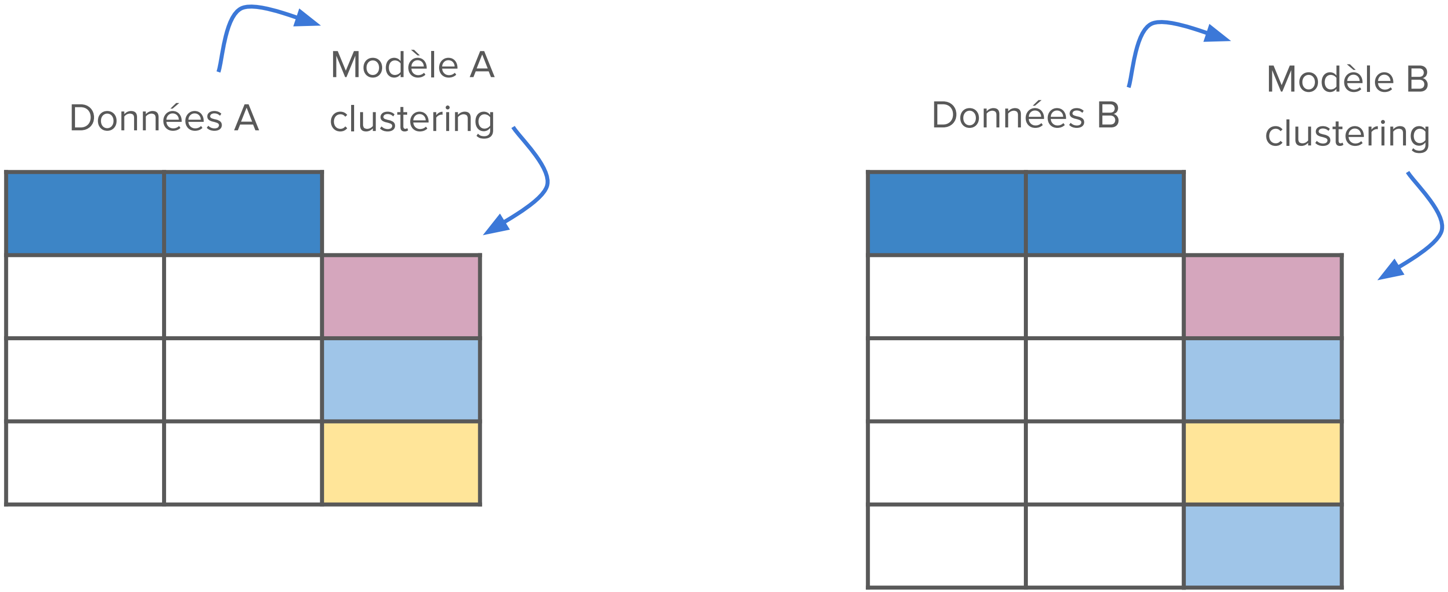
Simulation maintenance

Stratégie pour évaluer stabilité



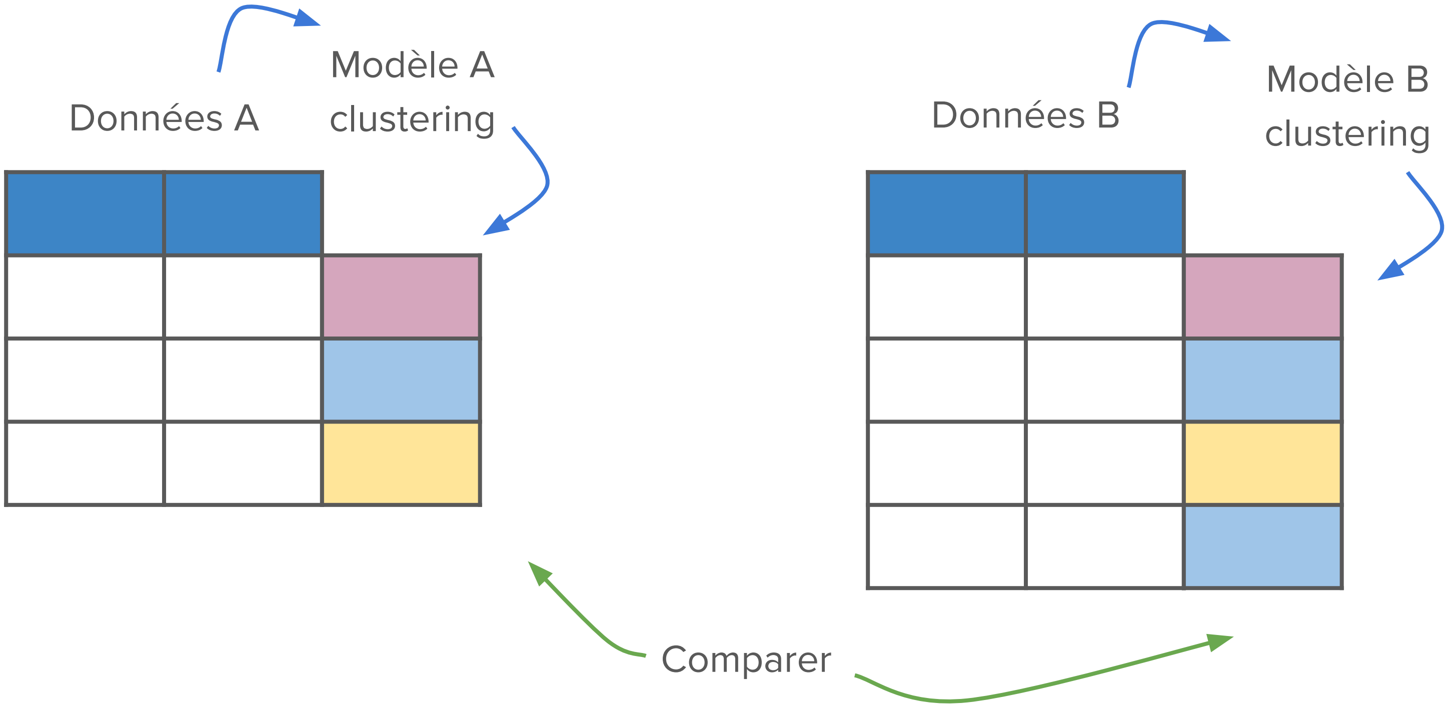
Simulation maintenance

Stratégie pour évaluer stabilité



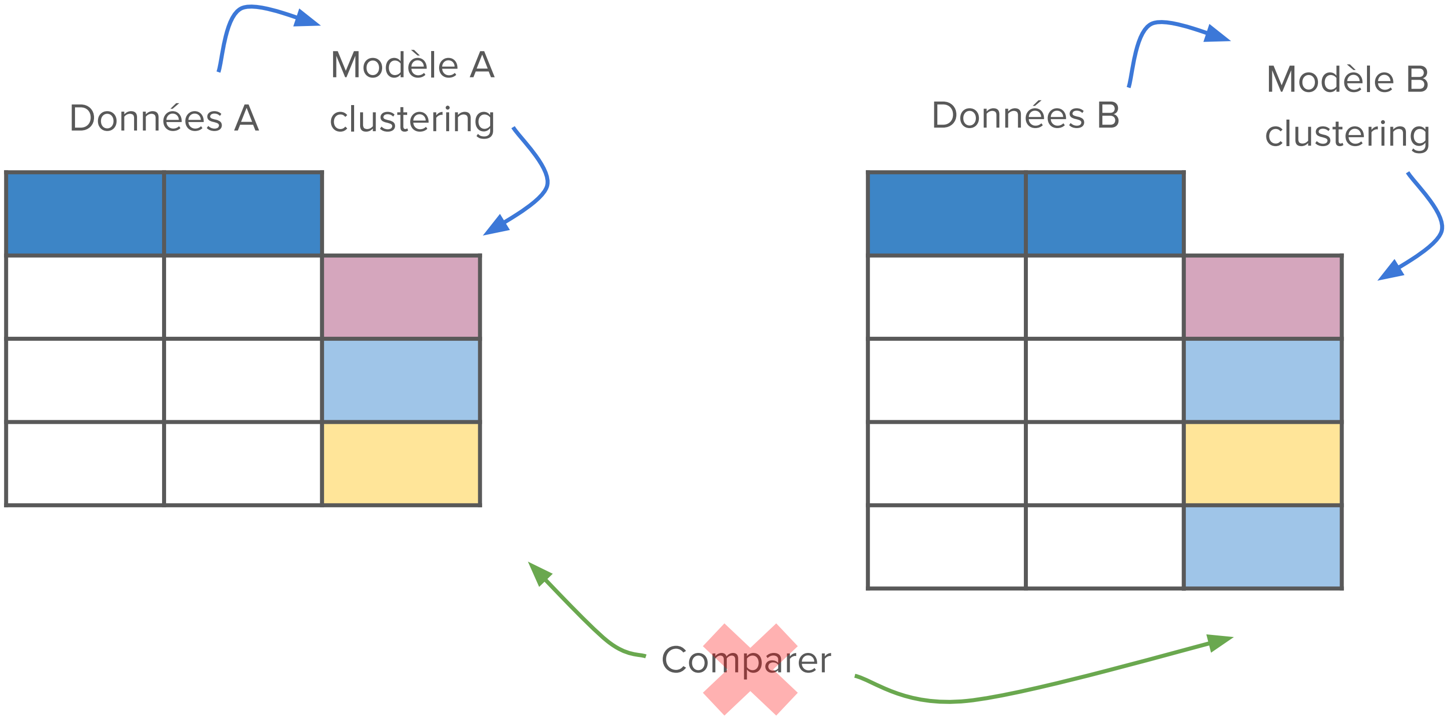
Simulation maintenance

Stratégie pour évaluer stabilité



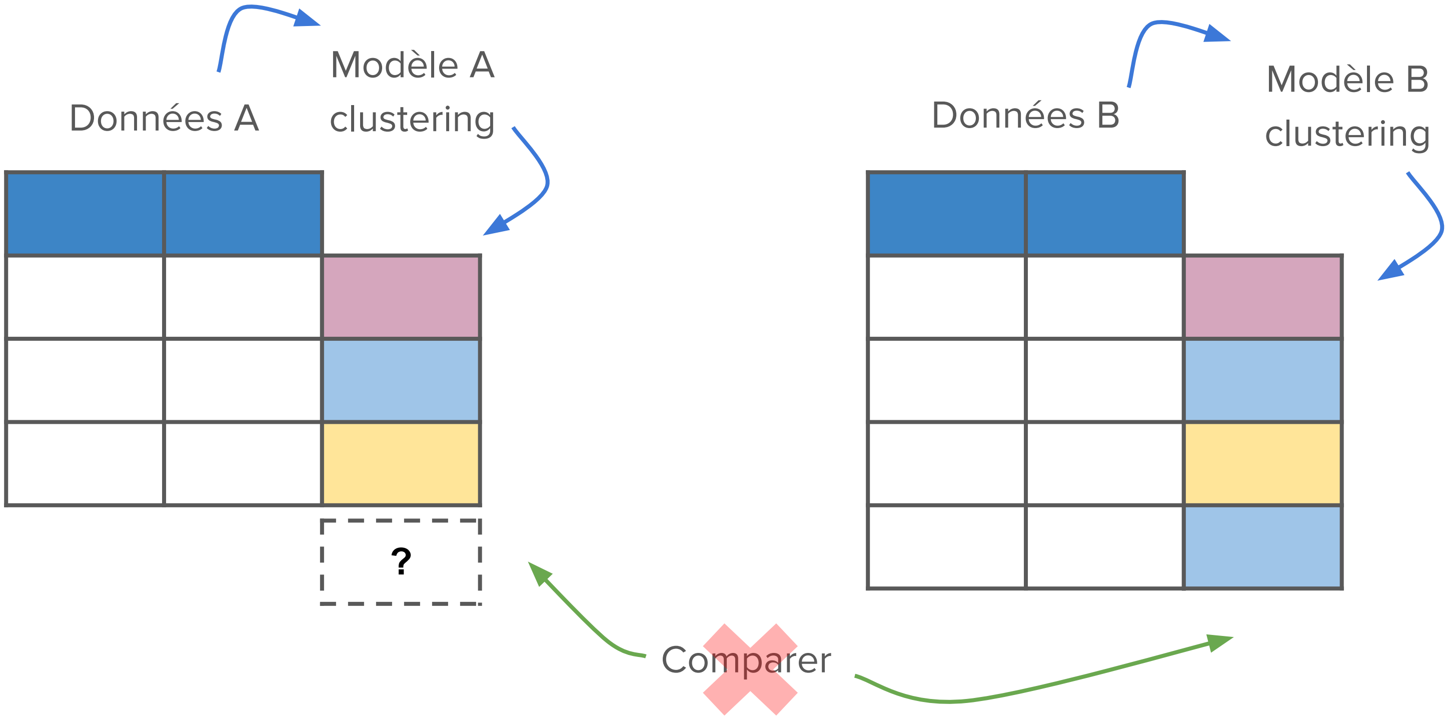
Simulation maintenance

Stratégie pour évaluer stabilité



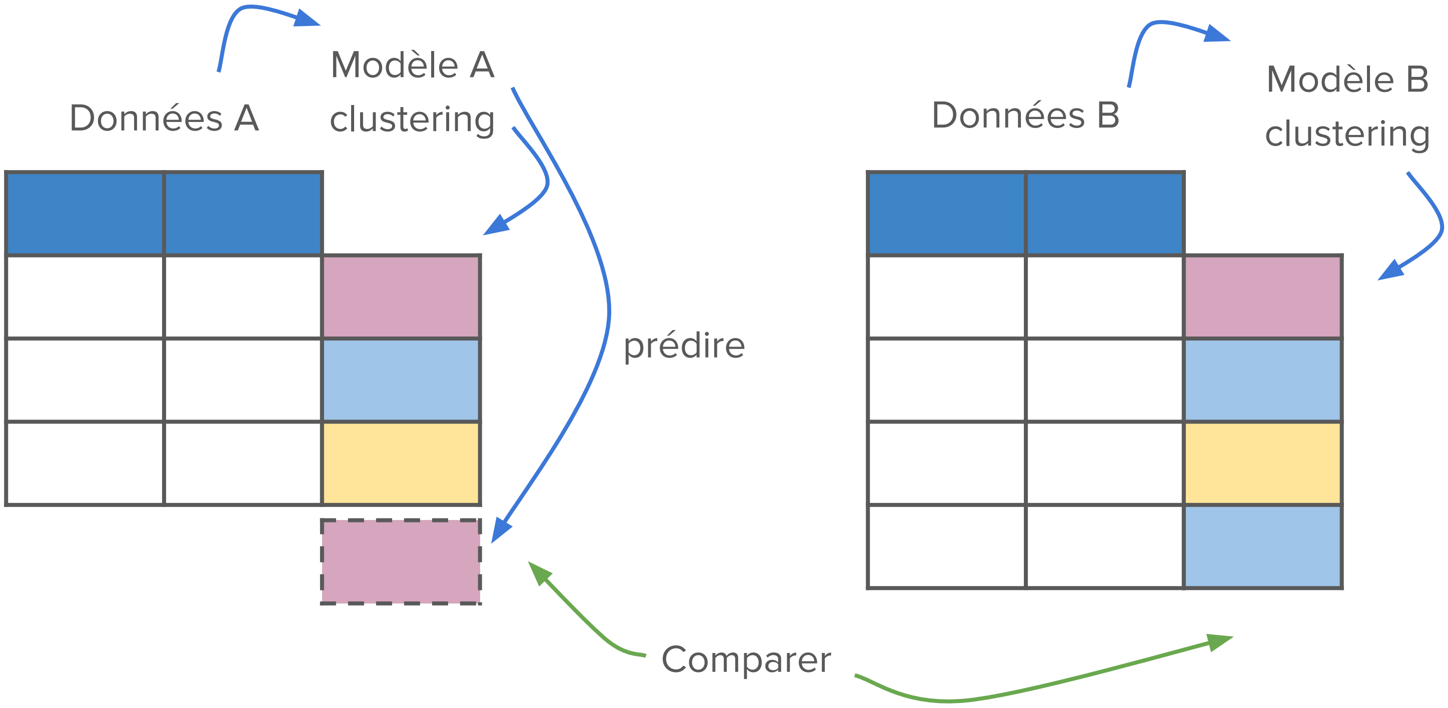
Simulation maintenance

Stratégie pour évaluer stabilité



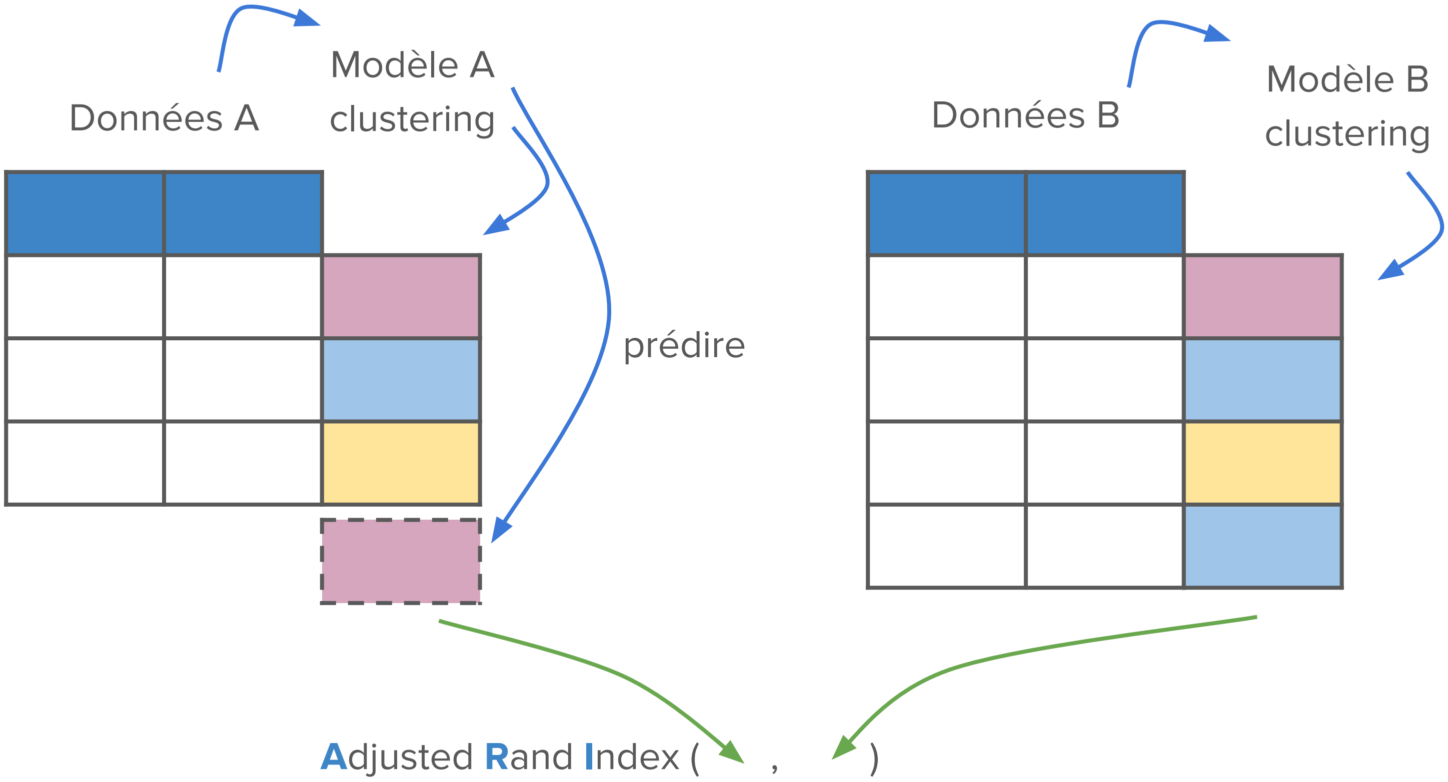
Simulation maintenance

Stratégie pour évaluer stabilité



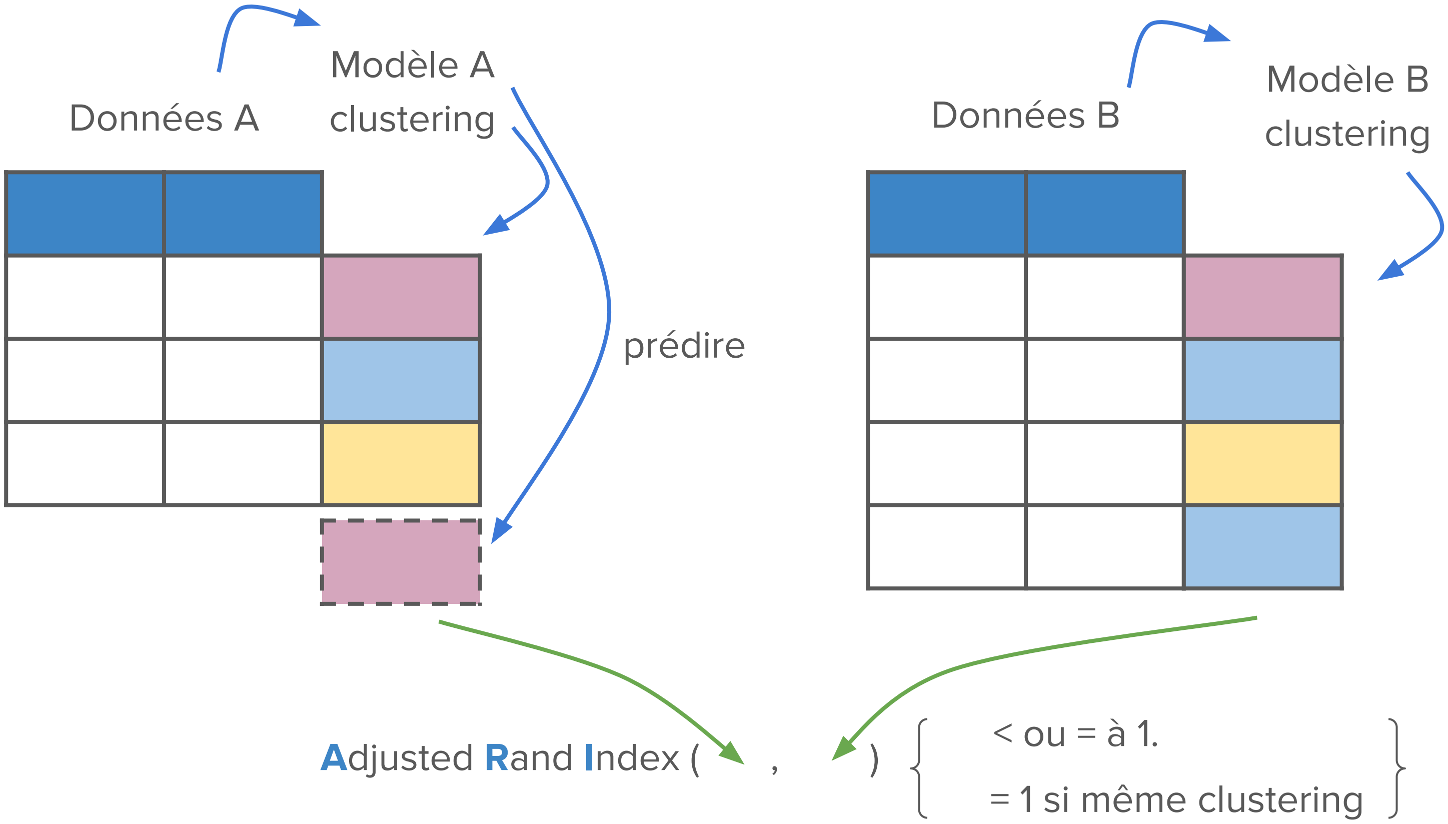
Simulation maintenance

Stratégie pour évaluer stabilité



Simulation maintenance

Stratégie pour évaluer stabilité



Simulation maintenance

Stratégie pour évaluer stabilité

Simulation maintenance

Stratégie pour évaluer stabilité

date fixée **start**

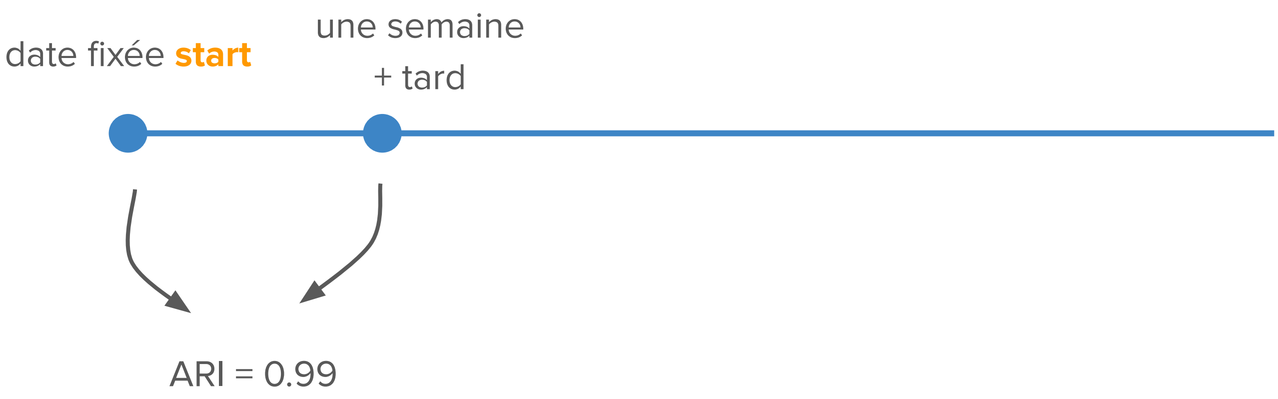
Simulation maintenance

4

Stratégie pour évaluer stabilité

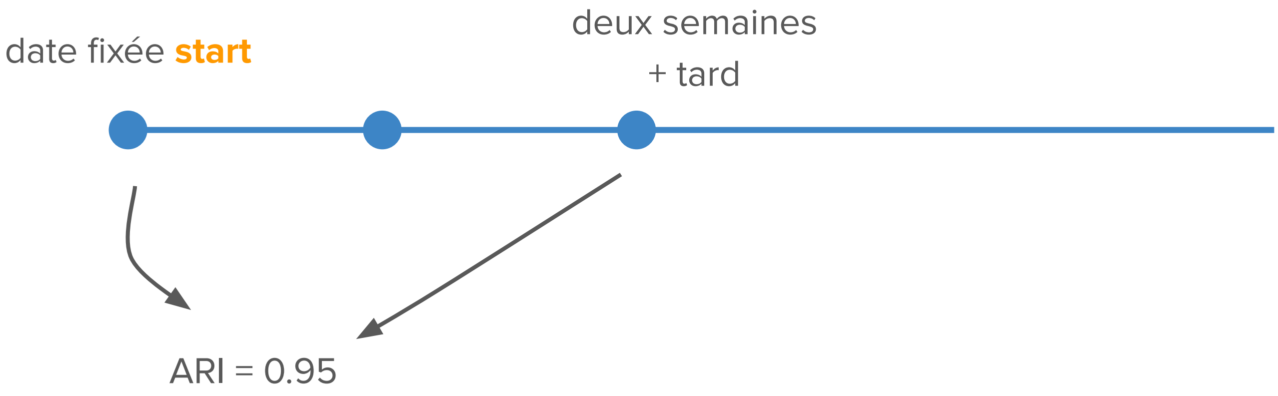
Simulation maintenance

Stratégie pour évaluer stabilité



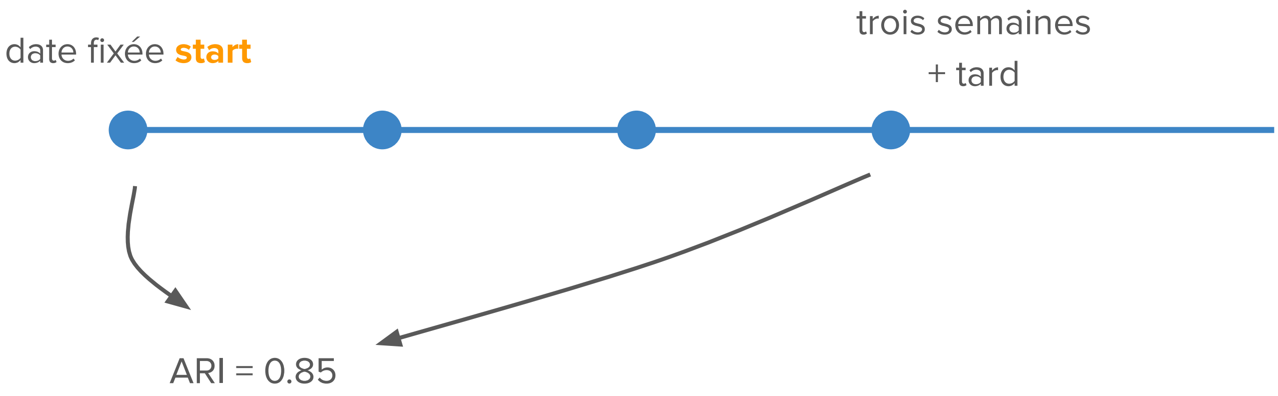
Simulation maintenance

Stratégie pour évaluer stabilité



Simulation maintenance

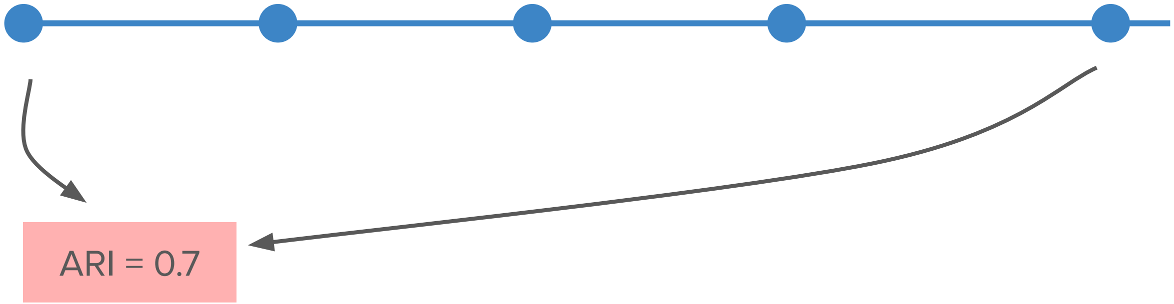
Stratégie pour évaluer stabilité



Simulation maintenance

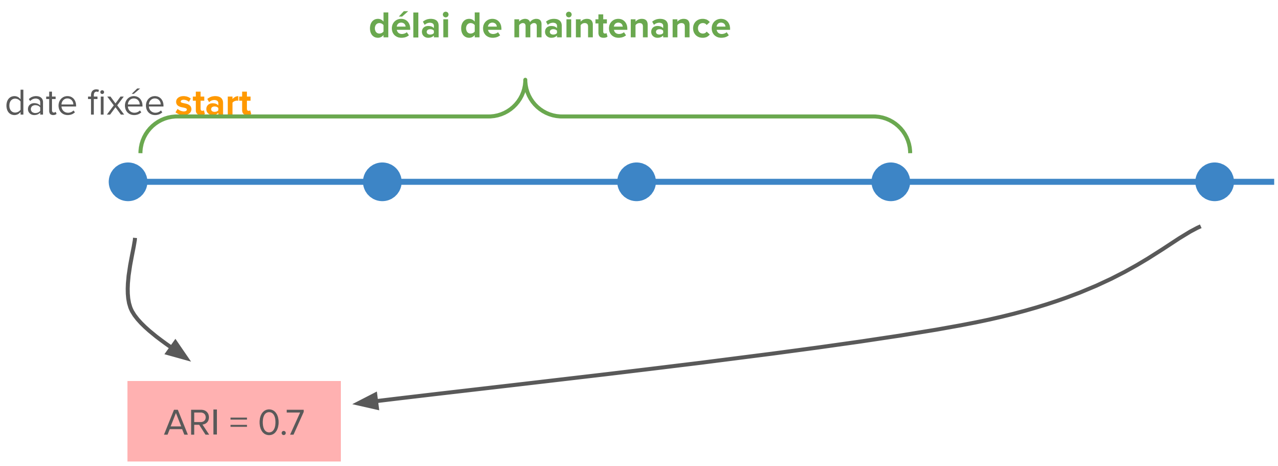
Stratégie pour évaluer stabilité

date fixée **start**



Simulation maintenance

Stratégie pour évaluer stabilité



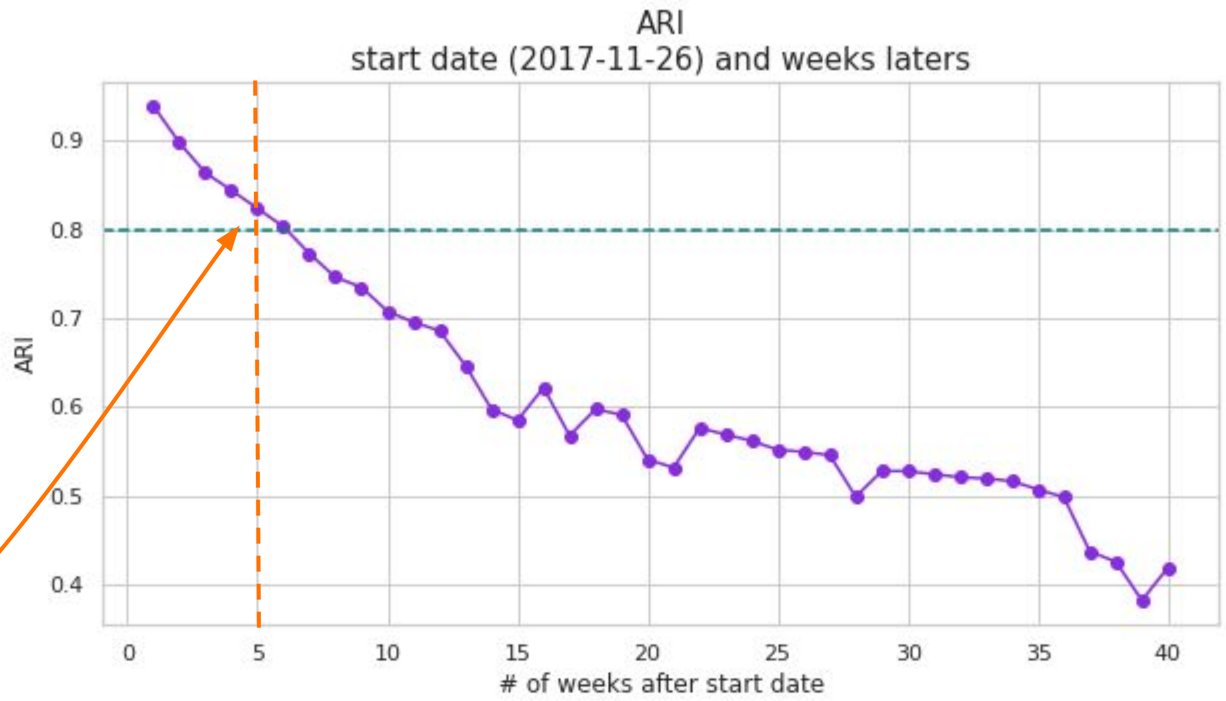
Simulation maintenance

date de départ:
2017-11-26

Simulation maintenance

Délai conseillé de maintenance:

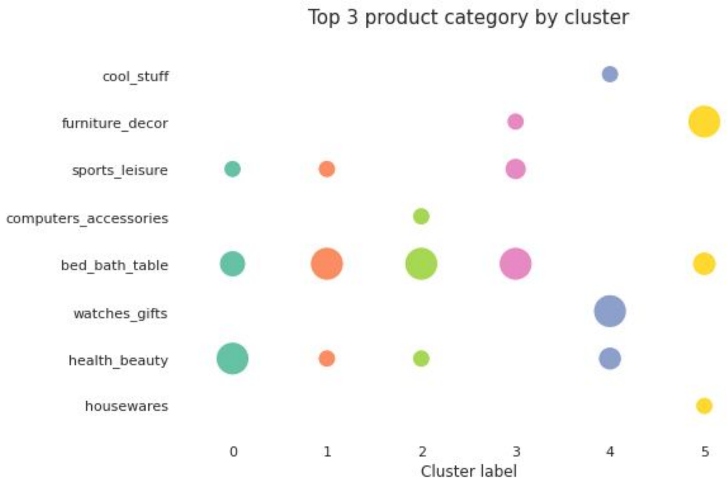
5 semaines



Conclusions et aller + loin

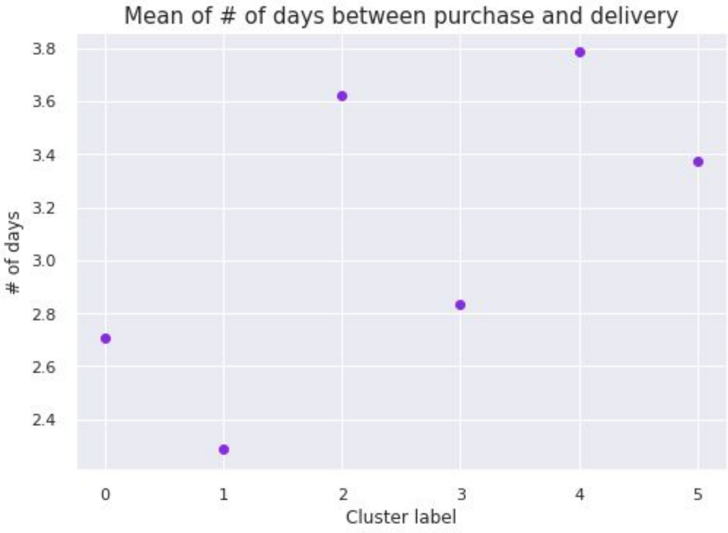
- Segmentation exploitable

Conclusions et aller + loin



Etiquette cluster	Profil client
0	Acheté récemment, **client moyen**
1	Non-régulier, acheté il y a **longtemps, content.**
2	💸 **Hautes dépenses**
3	Clients **fréquents**
5	🔢 **Grande quantité** d'items
4	🙁 **Pas satisfait**

Conclusions et aller + loin



Etiquette cluster	Profil client
0	Acheté récemment, **client moyen**
1	Non-régulier, acheté il y a **longtemps, content.**
2	💸 **Hautes dépenses**
3	Clients **fréquents**
5	🔢 **Grande quantité** d'items
4	🙁 **Pas satisfait**

Conclusions et aller + loin

Segmentation clients Olist

- Segmentation exploitable

Conclusions et aller + loin

Segmentation clients Olist

- Segmentation exploitable
- Meilleur segmentation si accès à plus de données

Conclusions et aller + loin

Segmentation clients Olist

Aller + loin

- Segmentation exploitable ➔ **creuser pour améliorer profil**
- Meilleur segmentation si accès à plus de données

Conclusions et aller + loin

Segmentation clients Olist

Aller + loin

- Segmentation exploitable ➜ **creuser pour améliorer profil**
- Meilleur segmentation si accès à plus de données

Projet OpenClassrooms

- Beaucoup de progrès python, pandas et visualisation

Conclusions et aller + loin

Segmentation clients Olist

Aller + loin

- Segmentation exploitable ➔ **creuser pour améliorer profil**
- Meilleur segmentation si accès à plus de données

Projet OpenClassrooms

- Beaucoup de progrès python, pandas et visualisation
- Organisation + systématique.