Segmentez des clients d'un site e-commerce

BASE DE DONNÉES ANONYMISÉE COMPORTANT DES INFORMATIONS SUR LES COMMANDES CLIENTS DE OLIST:
HTTPS://WWW.KAGGLE.COM/DATASETS/OLISTBR/BRAZILIAN-ECOMMERCE

Plan

Analyse Exploratoire

Feature Engineering

Choix du mode de visualisation des clusters

Clustering avec RFM

Clustering avec RFMS

Simulation délais de maintenance

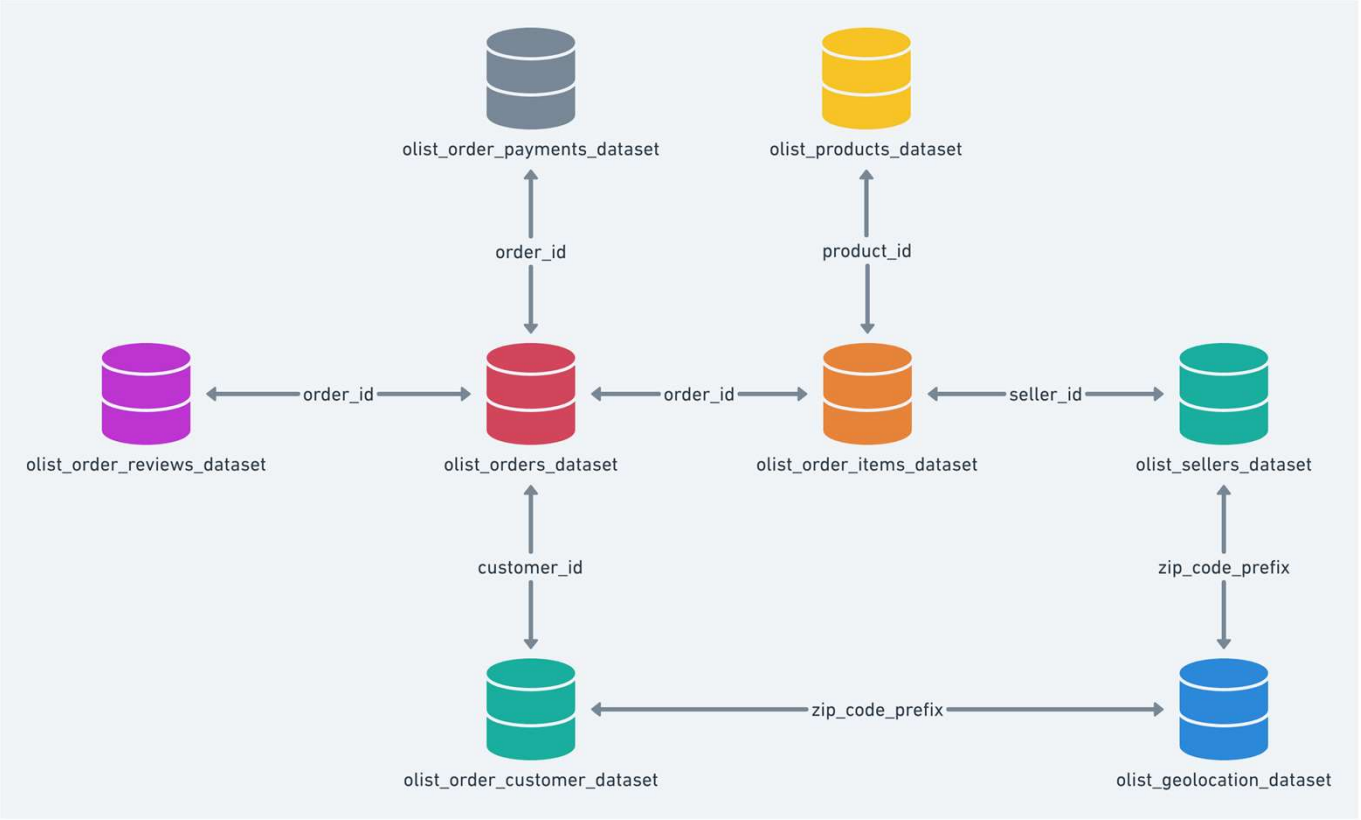
Synthèse et Conclusion

Analyse exploratoire

Présentation du jeu de données

8 datasets :

- **commandes,**
- **objets commandés**
- **produits**
- **vendeurs**
- **paiements**
- **Clients**
- **Reviews**
- **géolocalisation**



Analyse exploratoire

Présentation du jeu de données

customers

```
customer_id
customer_unique_id
customer_zip_code_prefix
customer_city
customer_state
```

orders

```
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_items

```
order_id
order_item_id
product_id
seller_id
shipping_limit_date
price
freight_value
```

Merged Datasets*	**clé**	**DataFrame**
orders - customers	customer_id	oc
oc – order_items	order_id	Order_customer_orderitems_merge

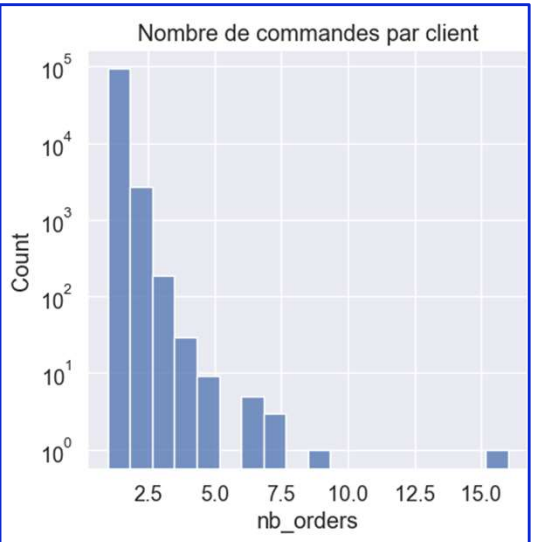
rfm0_df

94721 lignes

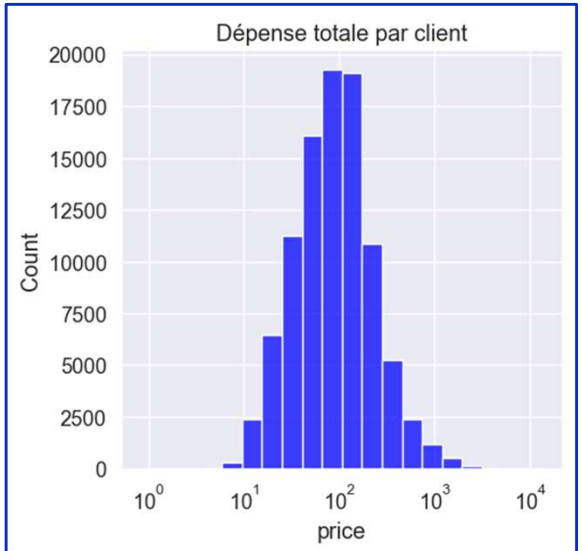
**inner merge*

Analyse exploratoire

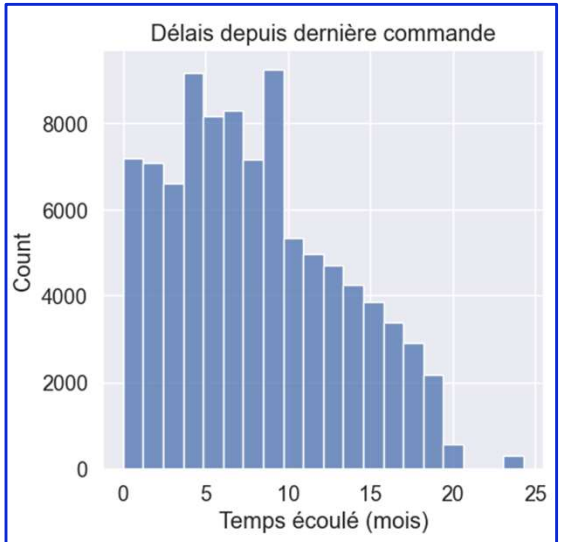
Présentation du jeu de données



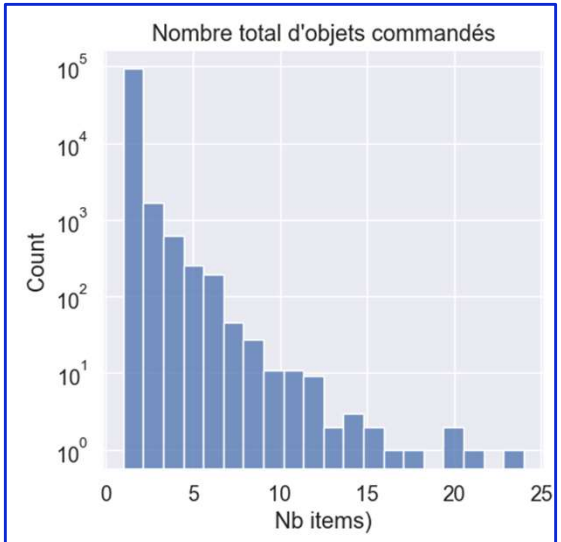
nb commande max : 16



Mediane ~ 100 real
Moyenne ~200 Real



Jusqu'à 2 ans
Majorité <1 an



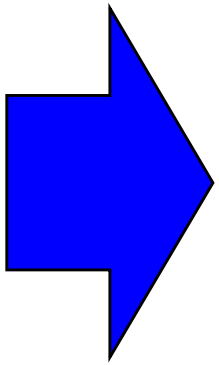
Nb total max : 24

Feature engineering

RFM* et/ou RFMS

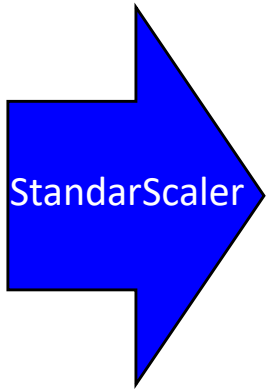
3 (ou 4) Features:
- nb_orders
- price
- **order_purchase_timestamp**

- review_score**



3 (ou 4) Features:
- nb_orders
- price
- **Elapsed_time**

- review_score



3 (ou 4) Features:
- **nb_orders_scaled**
- **price_scaled**
- **Elapsed_time_scaled**

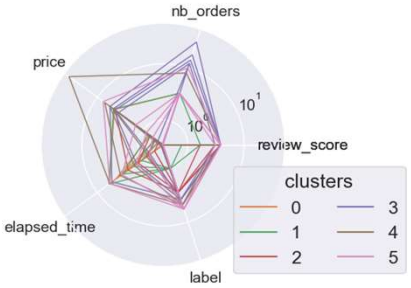
- **review_score_scaled**

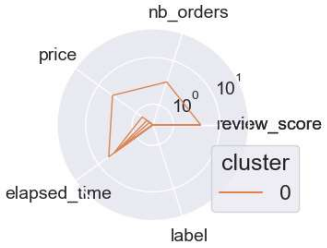
* Récence : elapsed_time, Fréquence : Nb_orders, Valeur : Price
**review score est celui de la dernière commande

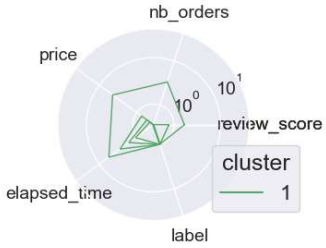
Choix du mode de visualisation des clusters

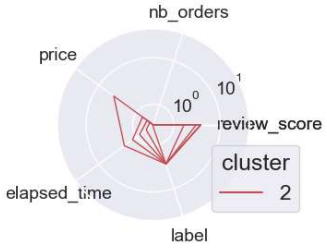
Exemple : Modélisation k-means (k=6)

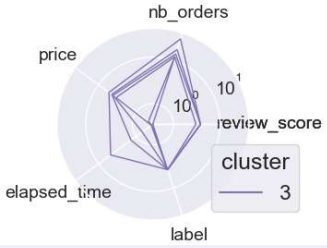
Mauvaise distinction entre les différents clusters | comparaison difficile

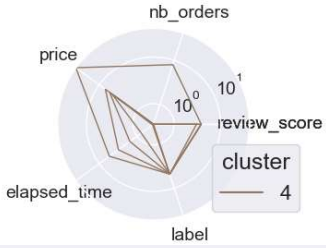


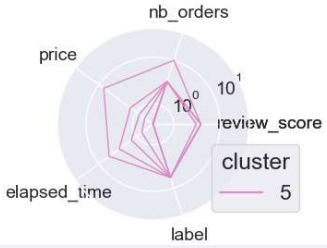


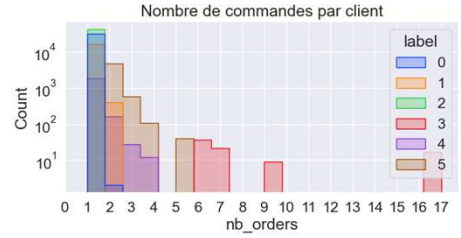


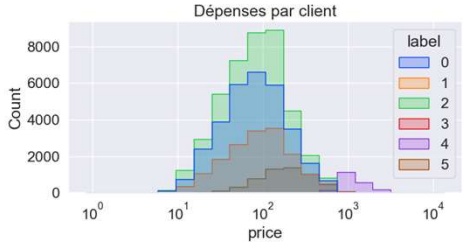


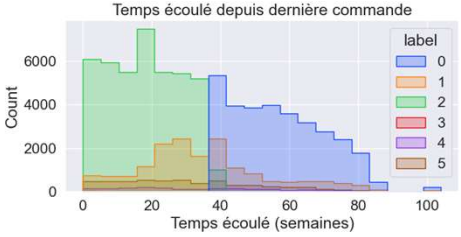


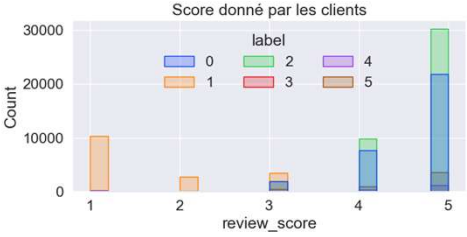










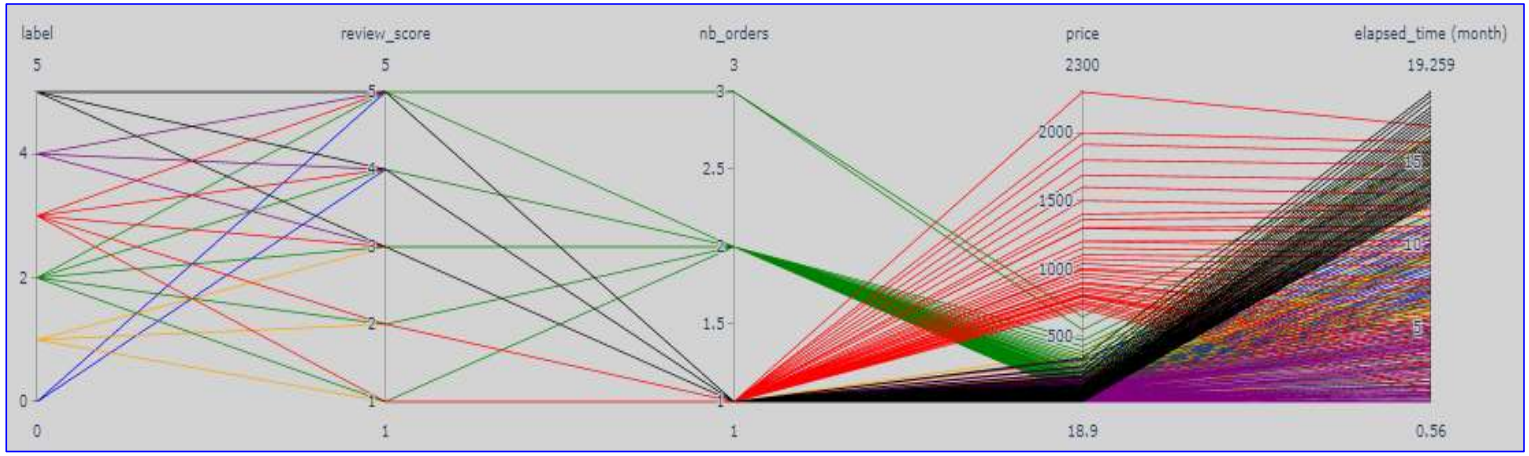


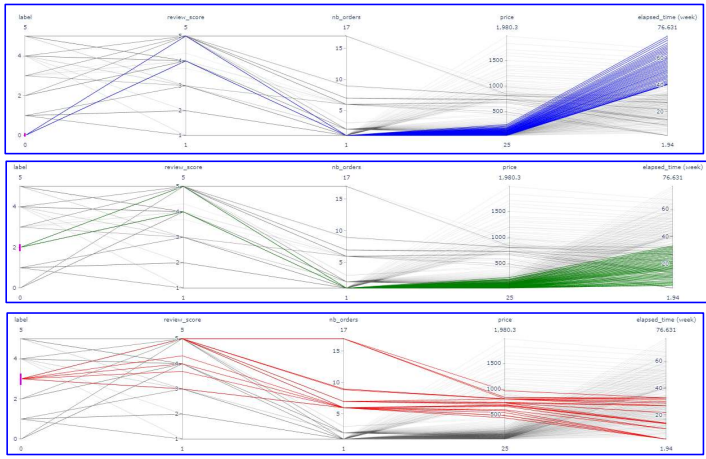
Mauvaise distinction entre les différents clusters

Choix du mode de visualisation des clusters

Exemple : Modélisation k-means (k=6)

Type de graphe : Parallel coordinates **Package** : Plotly





- ✓ **Visualisation des clusters en un coup d'œil,**
- ✓ **Distinction des valeurs par features**
- ✓ **Distinction des clusters par filtre**

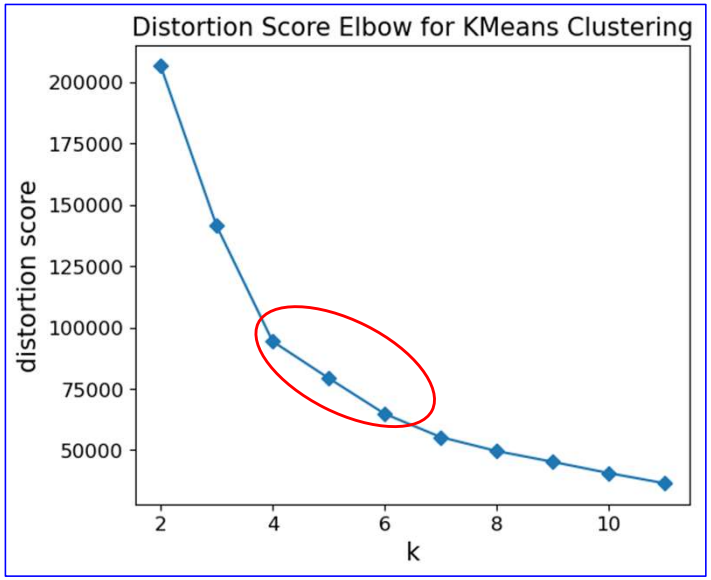
Filtrage : 5% outliers 'trop haut'
5% outliers 'trop bas'

EVALUATION DES MODELES DE CLUSTERING

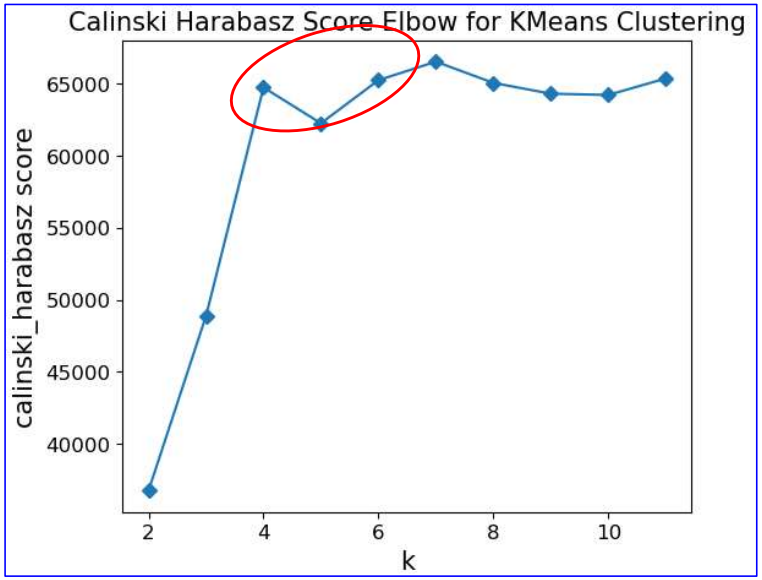
RFM* features

Modélisation k-means

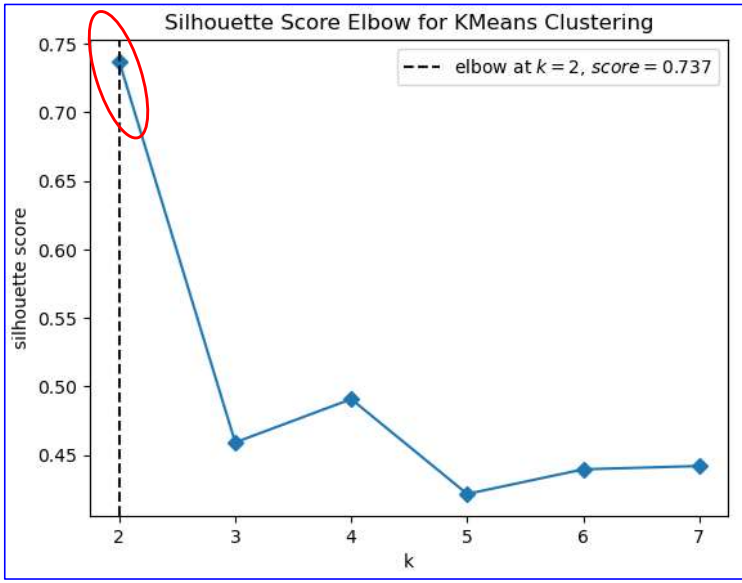
*Récence : elapsed_time, Fréquence : Nb_orders, Valeur : Price



distance intra



distance inter / distance intra



distance inter – distance intra

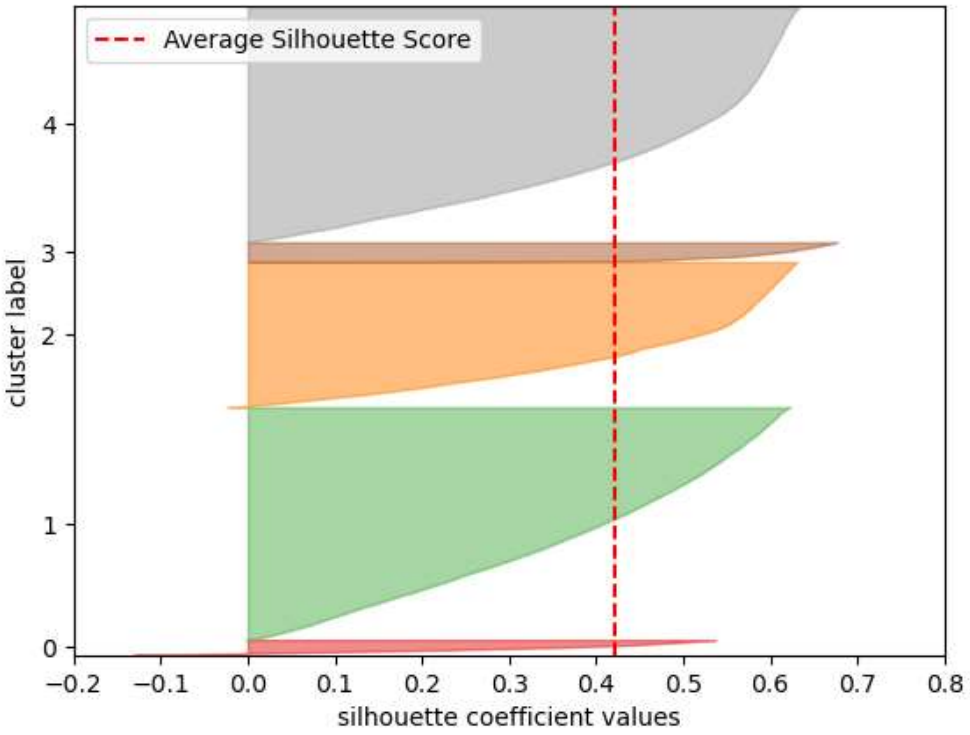
k optimum entre 4 et 6

RFM features

Modélisation k-means (k=5)

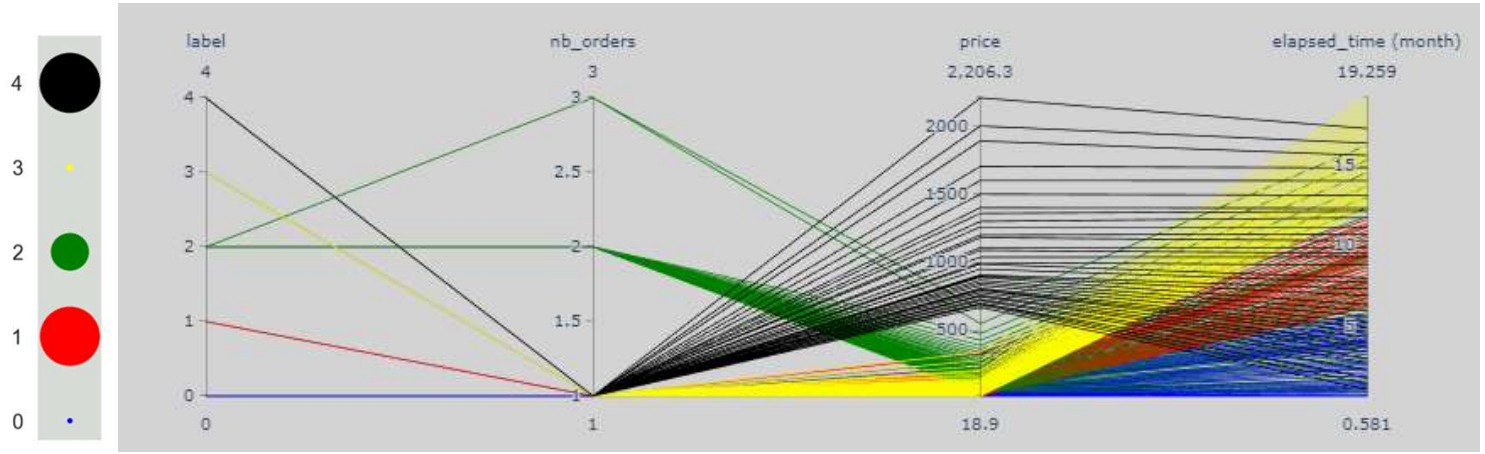
Clusters	Quantité
4	34589
3	2743
2	21236
1	34077
0	2075

Silhouette plot KMeans in 5 centers



RFM features

Modélisation k-means (k=5)



Cluster 0	Récent
Montant Cmd	≤250 R
Nb commandes	1
Dernière cmd	$< 6mois$

Cluster 1	Moins récent
Montant Cmd	≤250 R
Nb commandes	1
Dernière cmd	$6mois - 12\ mois$

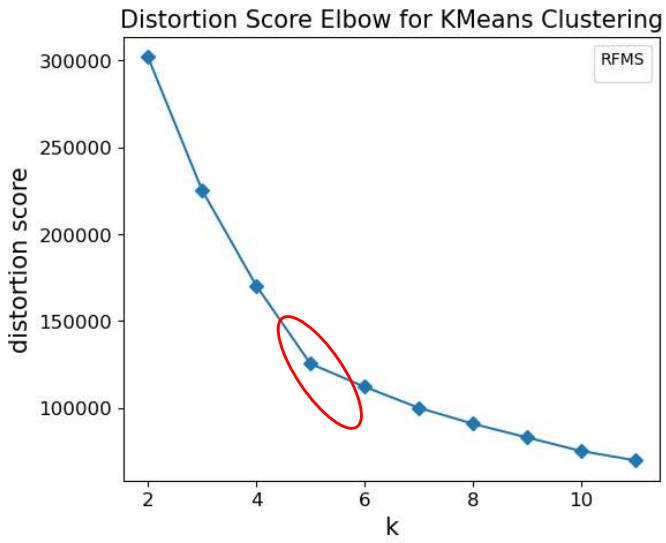
Cluster 3	Anciens
Montant Cmd	≤ 300 R
Nb commandes	1
Dernière cmd	$> 12\ mois$

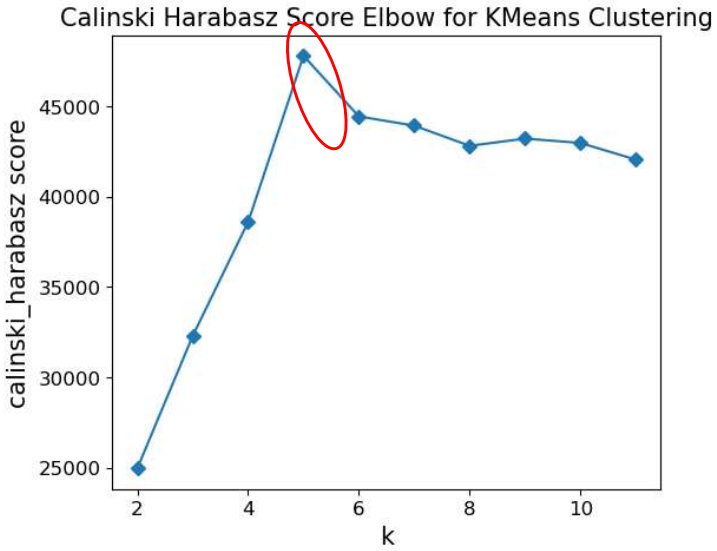
Cluster 4	Gros Budget
Montant Cmd	> 700 R
Nb commandes	1
Dernière cmd	$< 16\ mois$

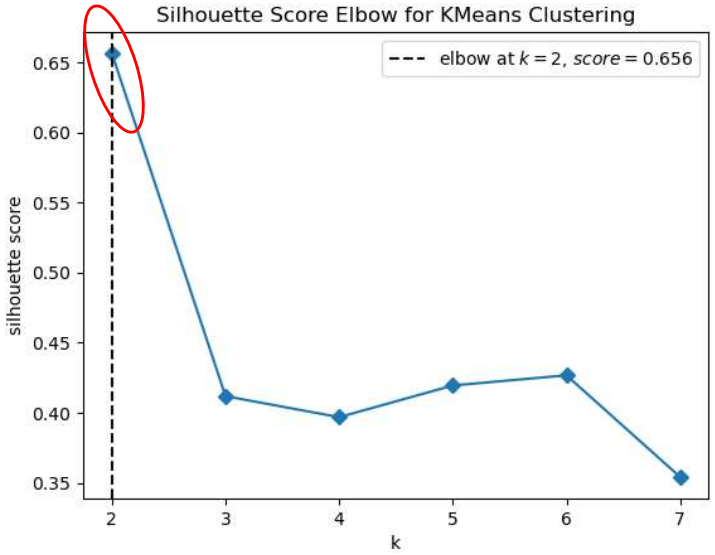
Cluster 2	Fidèle
Montant Cmd	≤ 700 R
Nb commandes	> 2
Dernière cmd	$< 16\ mois$

RFMS features

Modélisation k-means





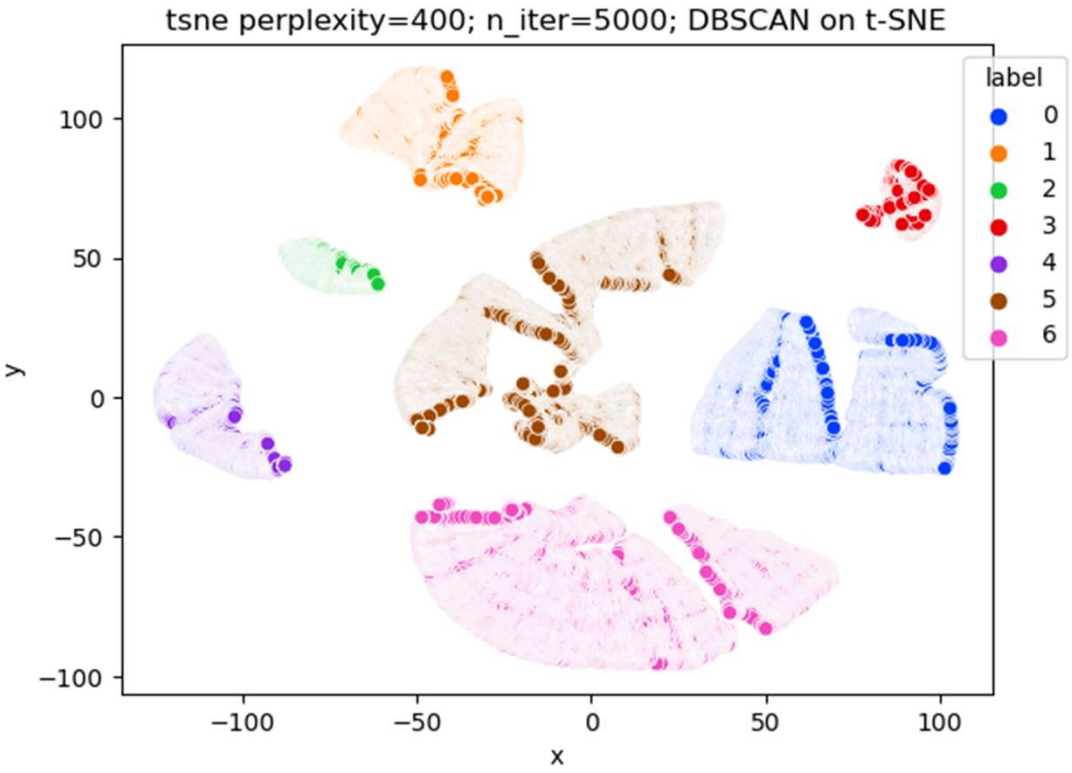


RFMS features

Visualisation et clustering

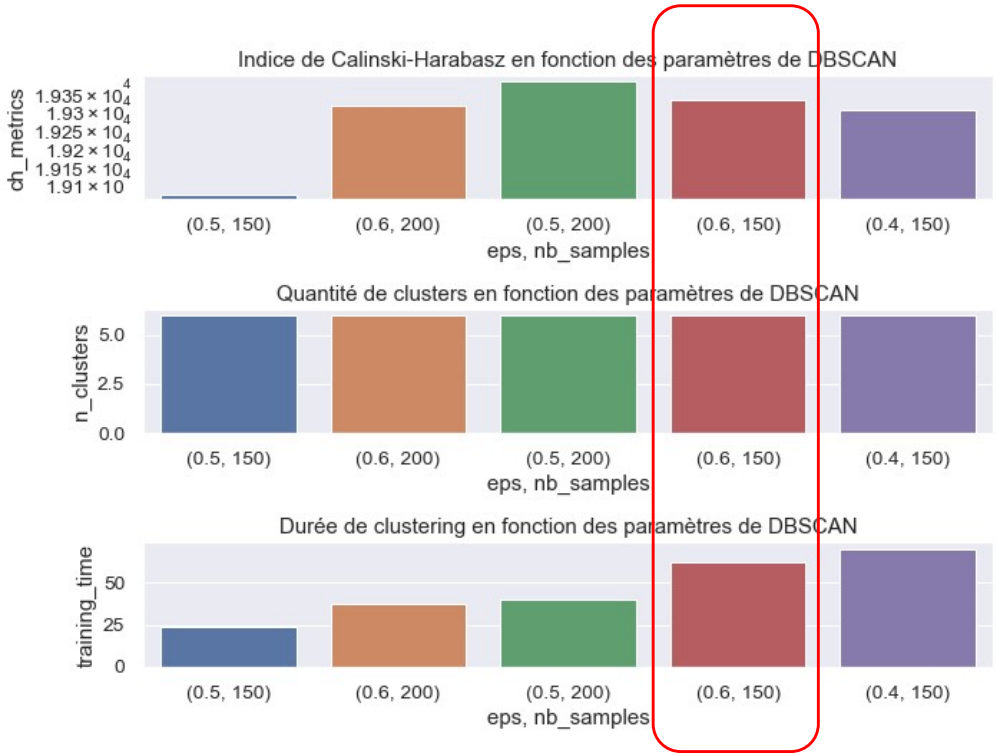
DBSCAN sur t-SNE

- **Modélisation avec DBSCAN rapide et « facile »**
- **t-SNE besoin important en ressources de calcul (> #heures)**
- **Segmentation peu pertinente vs. Marketting**



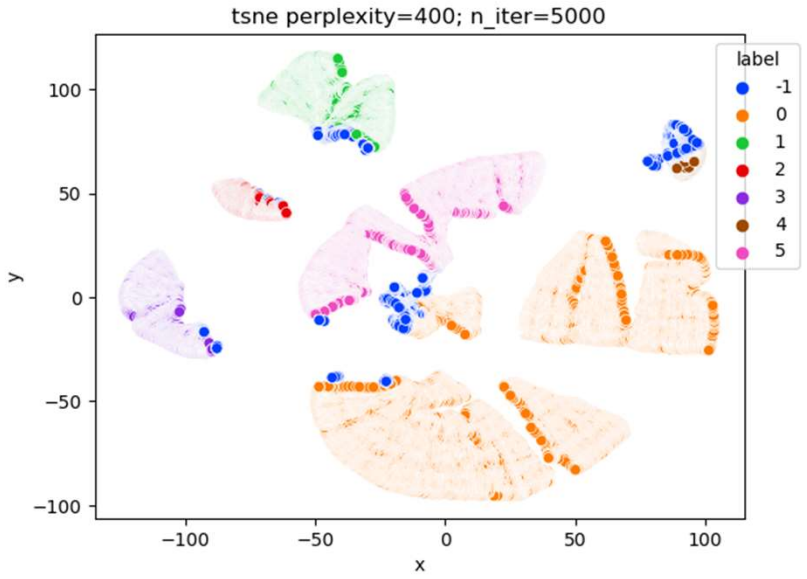
RFMS features

Modélisation DBSCAN: choix des hyperparamètres



DBSCAN [0.5, 200] vs. DBSCAN[AUTRES]

ARI ~ 0.99

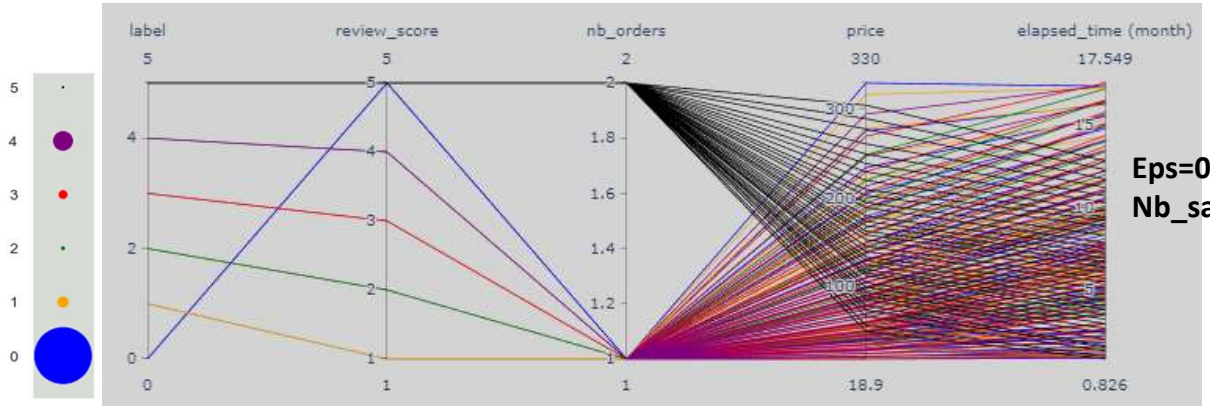


(Eps:0.6, Nb=150) Segmentation la plus pertinente pour Marketting

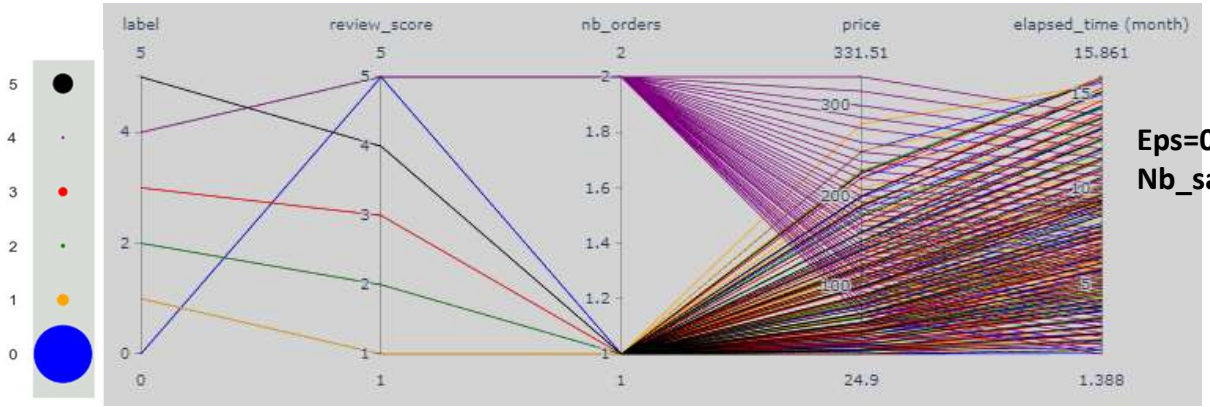
RFMS features

Modélisation DBSCAN

- **DBSCAN tant à créer des clusters autour des valeurs des variables categorielles si le nb cluster ≅ nb categories**
- **Temps d'entrainement assez long pour nos données (#minutes, gourmand en mémoire).**



Eps=0.5
Nb_samples_min=200

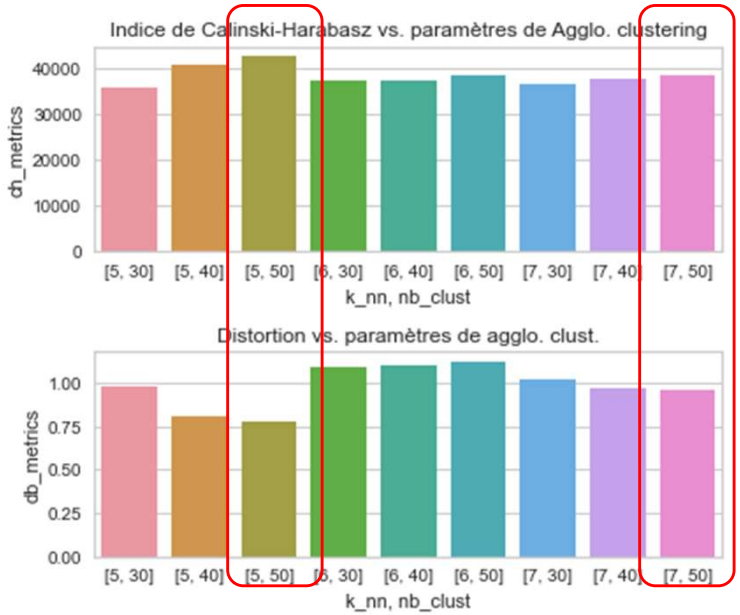


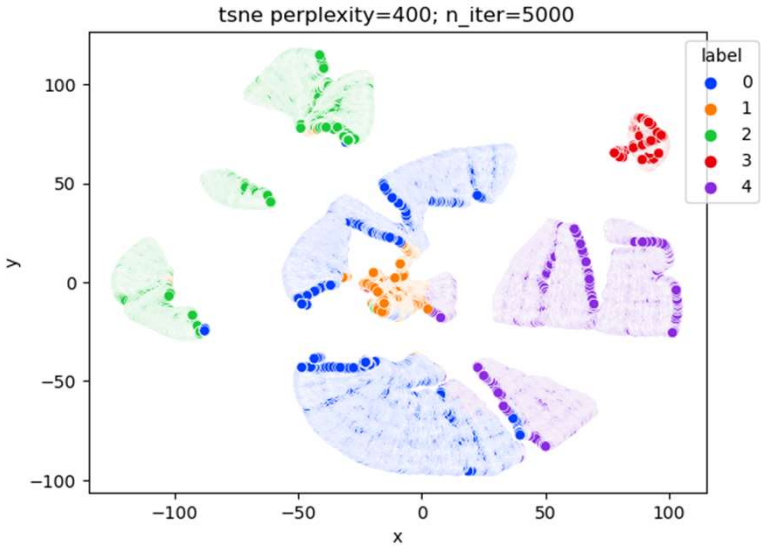
Eps=0.6
Nb_samples_min=150

RFMS features

Modélisation Agglomerative clustering: choix des hyperparamètres

kNN (« ward ») en pré-traitement des données : gain en temps et en ressource





RFMS features

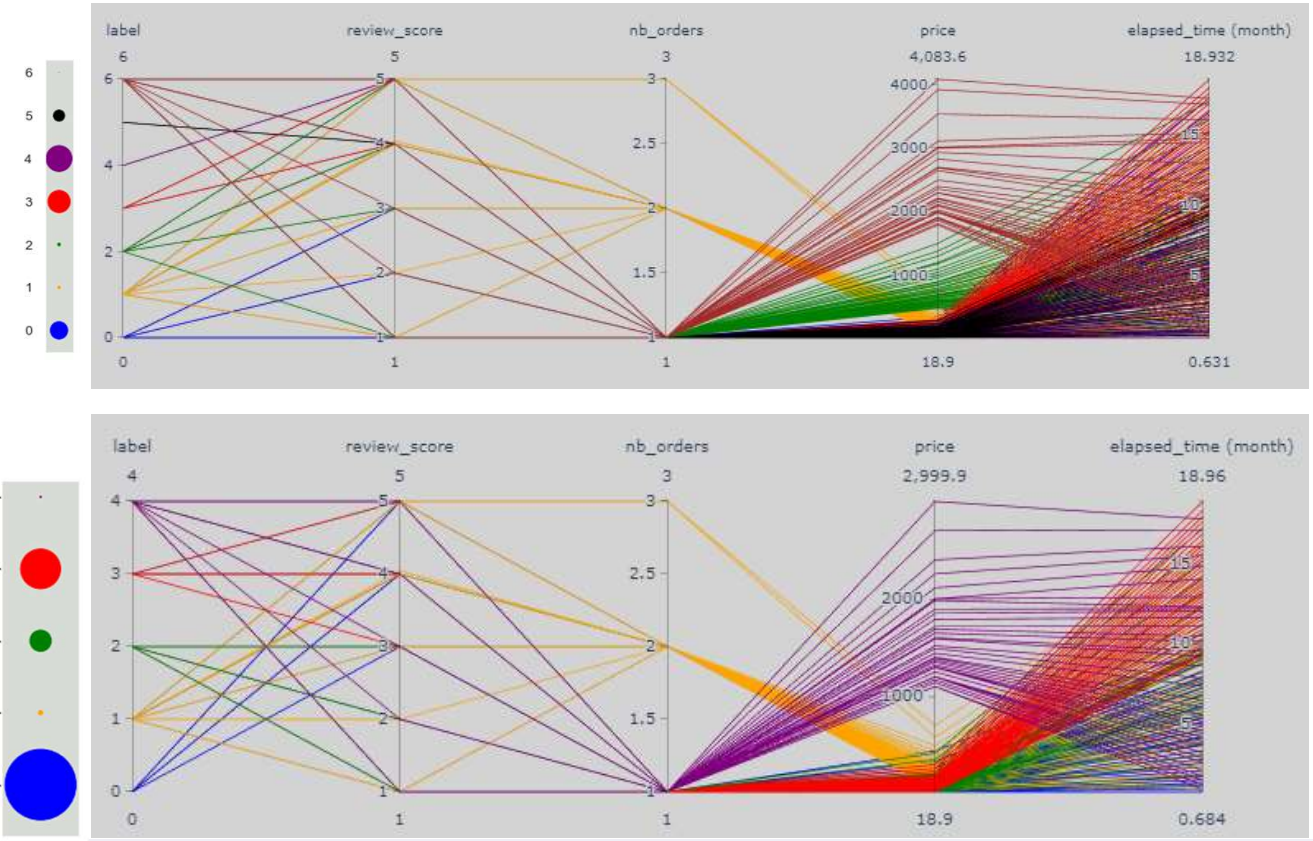
Modélisation Agglomerative clustering: knns=30

nb_clusters=7 :

- **Cluster 6 trop petit : 229**
- **Cluster 4 et 5 peuvent être regroupé en 1 cluster**
- **Finalement nb_clusters=5**

nb clusters=5

Cluster	Quan.	type
4	804	Gros Budget
3	27463	Plutôt satisfait ancien
2	14639	mécontent
1	2900	fidèles
0	48915	Plutôt satisfait récent



RFMS features

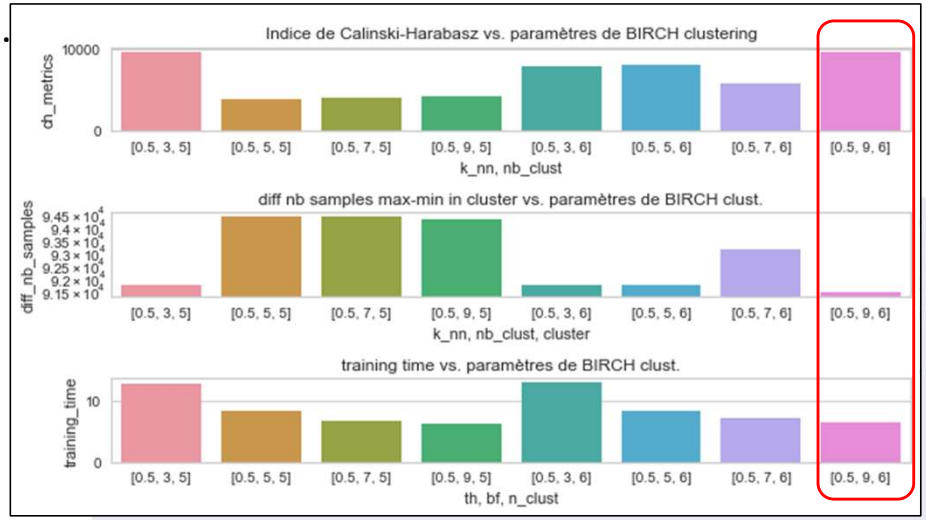
Modélisation Birch

Birch clustering très efficace en temps de calcul.
2 étapes:
- Construction du clustering Feature Tree
- Agglomérative clustering

Mais
↗Threshold : ↘ le nombre de sub-cluster.
créé un grand cluster + clust. minuscules :
→ donne trop d'importances aux outliers

↘ Threshold : agglomerative clustering pas efficace → grande ressource mémoire nécessaire – long temps de calcul

Pour rendre cette algo efficace il est nécessaire d'appliquer un filtrage entre Birch et Agglo clustering (slide suivante).



Birch clustering (n_clust=6) optimum pour th=0.5, bf=9

nb samples
390
6
46
2853
1
91425

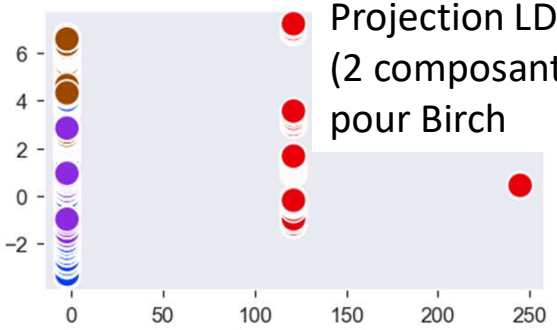
Projection LDA (2 composantes) pour Birch

RFMS features

Modélisation Birch:

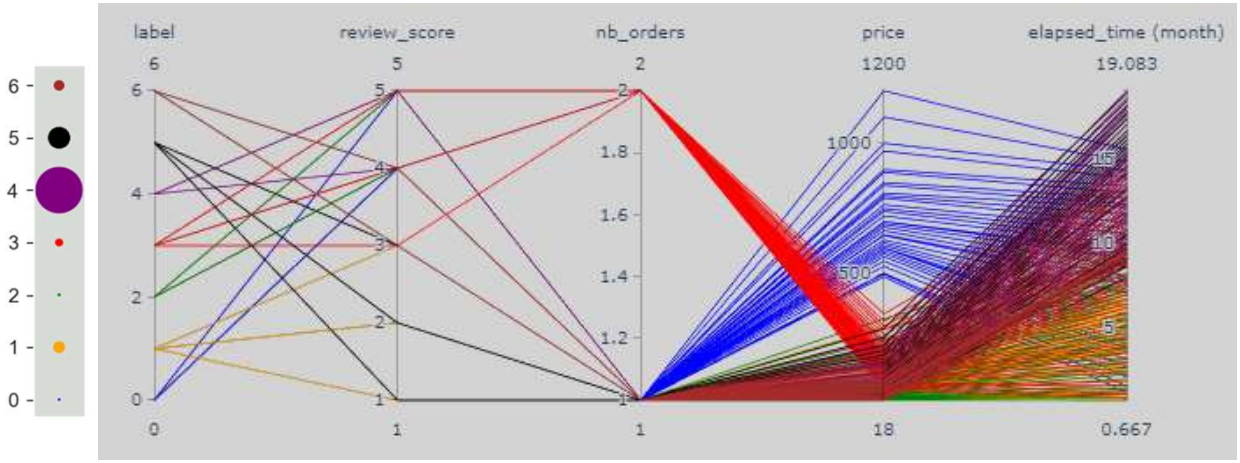
Après filtrage segmentation plus intéressante, mais:

- Filtrage de 3496 clients
- un peu fastidieux



Projection LDA (2 composantes) pour Birch

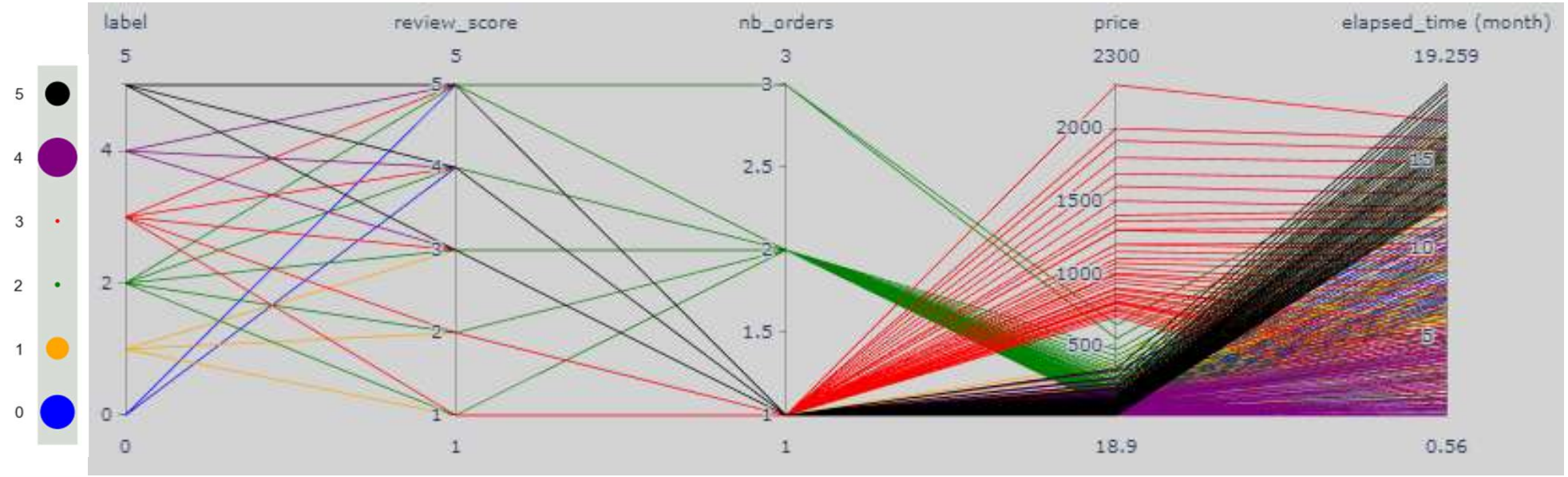
Cluster	Quan.	type
6	8893	Moyennement satisfait
5	19698	Plutôt satisfait anciens
4	43317	Plutôt satisfait récents
3	6371	Mécontents anciens
2	1689	fidèles
1	9935	Mécontents récents
0	1322	Gros Budget



RFMS features

Modélisation k-means (k=6, random state=7)

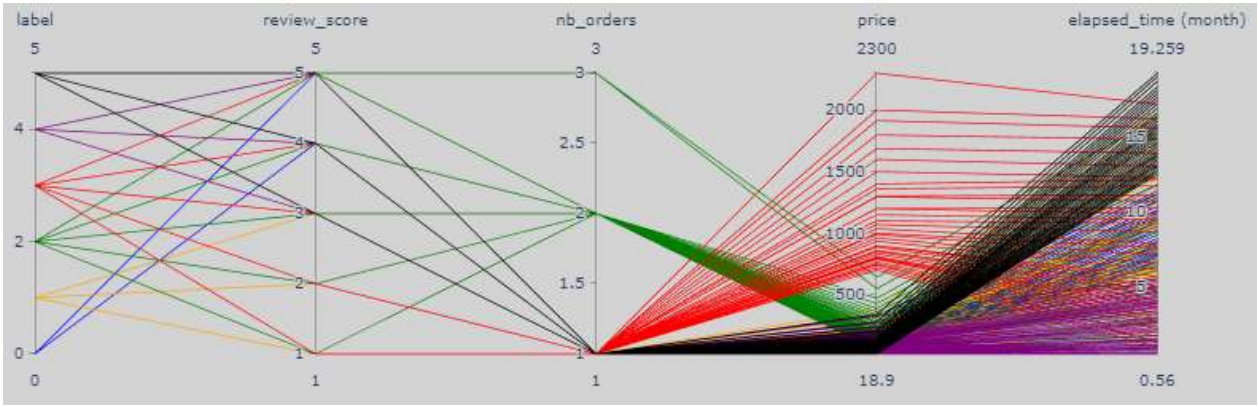
18013
29654
1955
2871
16585
25643



RFMS features

Modélisation k-means (k=6, random state=7)

18013	5
29654	4
1955	3
2871	2
16585	1
25643	0



Cluster 1	Mécontent
Review_score	≤ 3
Montant Cmd	< 400
Nb commandes	1
Dernière cmd	$17\ mois$

Cluster 3	Gros Budget
Review_score	-
Montant Cmd	>700 R
Nb commandes	1
Dernière cmd	$17\ mois$

Cluster 0	Satisfait 6 mois
Review_score	≥ 4
Montant Cmd	≤ 300
Nb commandes	1
Dernière cmd	$6\ mois\ -12\ mois$

Cluster 4	Plutôt satisfait récent
Review_score	> 3
Montant Cmd	< 300 Real
Nb commandes	1
Dernière cmd	$6\ mois$

Cluster 2	fidèle
Review_score	–
Montant Cmd	≤ 700
Nb commandes	> 2
Dernière cmd	$16\ mois$

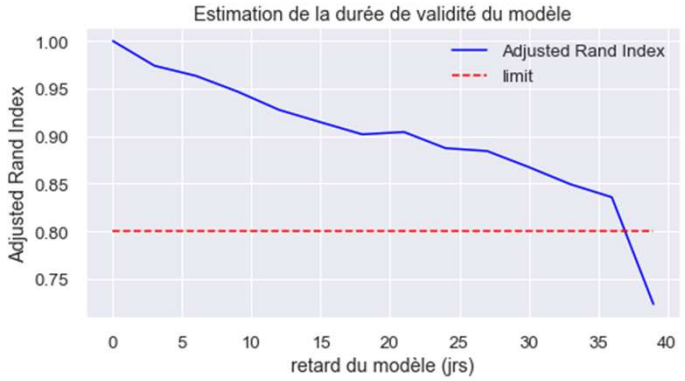
temps

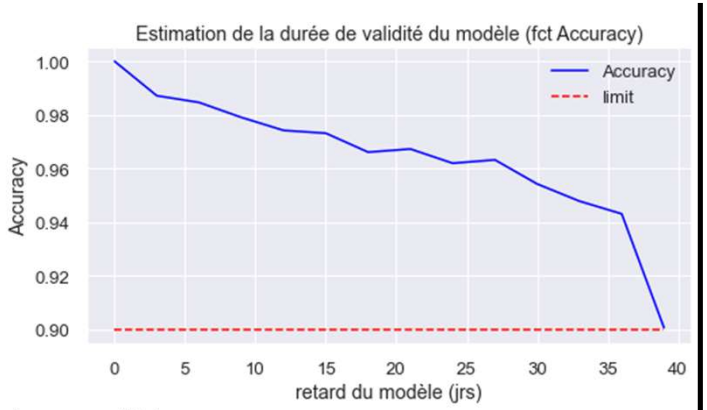
Cluster 5	Plutôt satisfait old
Review_score	≥ 3
Montant Cmd	≤ 300
Nb commandes	1
Dernière cmd	$> 12\ mois$

Simulation

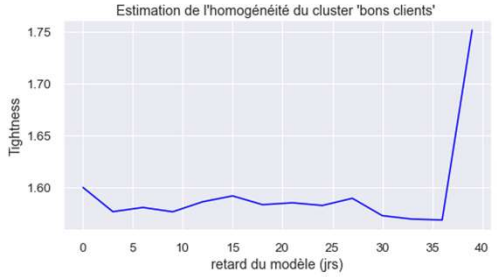
Définition du délai de maintenance

Modèle Kmeans k=6

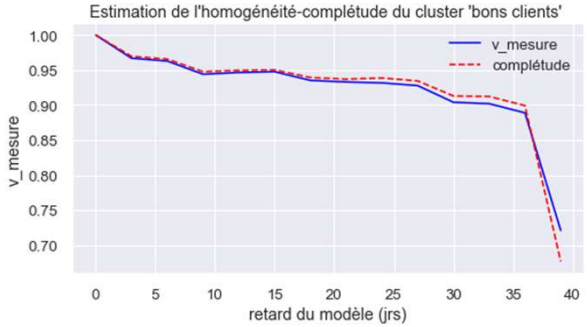




La durée de validité du modèle 7 semaines



Le cluster « bons clients » se dégradent rapidement



Synthèse et Conclusion

- Différents algorithmes ont été testés:
  - Kmeans, DBSCAN, Agglomerative Clustering, BIRCH
- Différents outils de visualisation ont été testés
  - web graph, parallel coordinates,
  - t-SNE pour la réduction de dimension et la visualisation des clusters
- La pré-sélection des modèles a été réalisés par Calinski-Harabasz index.
- Le modèle Kmeans (k=6) a été choisi:
  - Segmentation des clients pertinente pour le Marketting
- Une mise à jour toutes les 7 semaines est nécessaire pour conserver un clustering pertinent.