



Parcours OpenClassRooms

Data Scientist

P5 Segmentation client pour un site de e-commerce

Développé sur un Notebook Jupyter Colaboratory



Pictures used for educational purpose only

Sommaire

I. Problématique et pistes de recherches

II. Nettoyage, feature engineering et analyse exploratoire

III. Pistes de modélisations

IV. Optimisation du modèle final

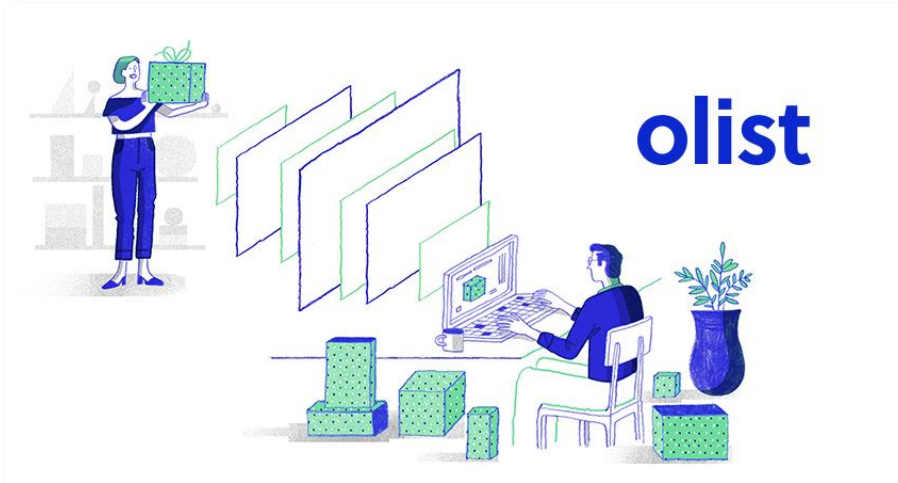
V. Bilan et perspectives

VI. Perspectives d'améliorations

I. Problématique et pistes de recherches

1. Olist : son activité, ses besoins

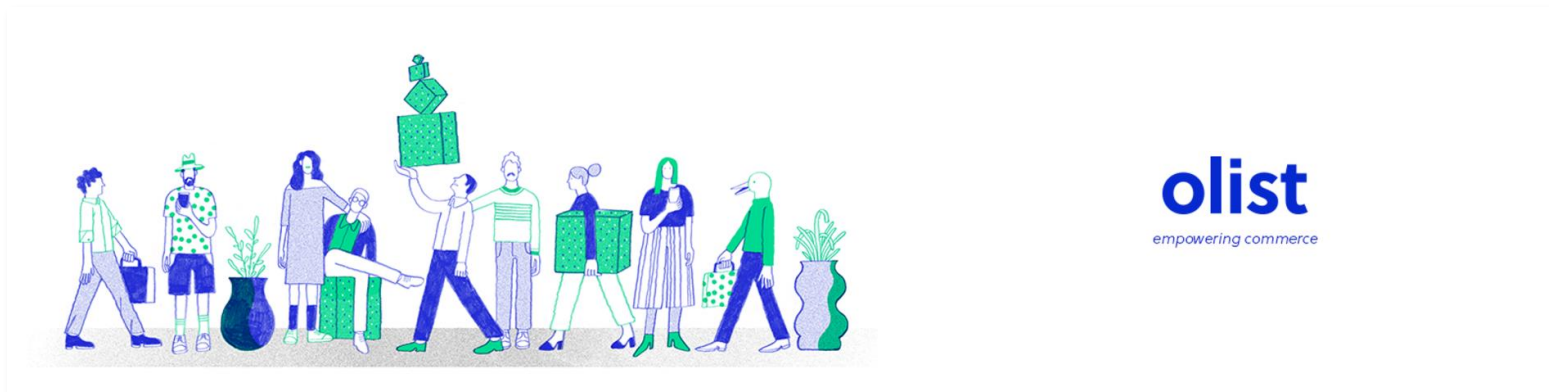
Création en 2015



Vente par correspondance



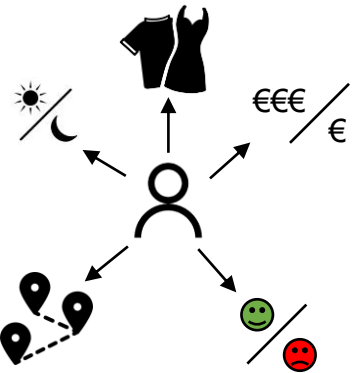
Activités principales au Brésil



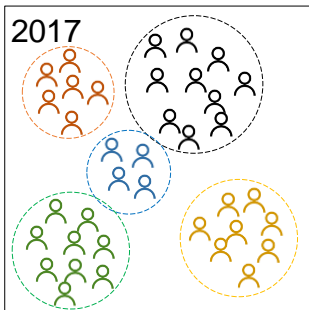
→ Besoin d'identifier les différents types de clients, afin d'optimiser les campagnes de marketing

I. Problématique & pistes de recherches

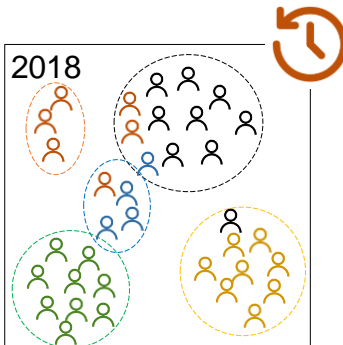
2. Le cahier des charges



- 1) Comprendre les différents types de clients
 - a) grâce à leurs comportements
 - b) et grâce à leurs données personnelles.



- 2) Utiliser des méthodes non supervisées afin de regrouper les clients de profils similaires.



- 3) Proposer un contrat de maintenance, basé sur une analyse de la stabilité des groupes au cours du temps.

I. Problématique & pistes de recherches

3. Le livrable recherché

La segmentation



Client achetant le week-end, budgets importants, ...



Client achetant des produits de décoration, passant commande en matinée ...



Client mécontent, subissant des retards de livraison, ...



Client éloigné géographiquement, faibles budgets, ...

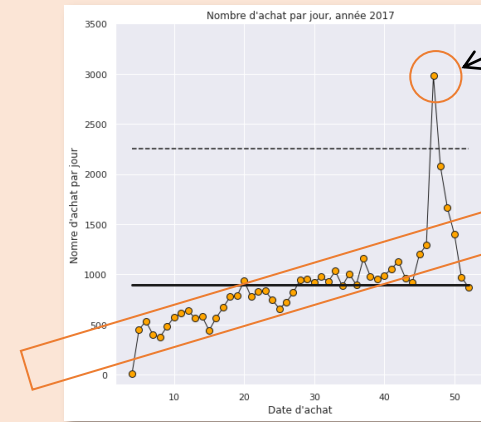


...

→ Finalité : orienter les campagnes marketing d'Olist selon les profils clients identifiés

La fréquence de mise à jour

Tenir compte des **événements ponctuels** (vacances, fêtes nationales brésiliennes, ...)



Tenir compte des **tendances générales** (accès à internet grandissant, croissance d'Olist, ...)

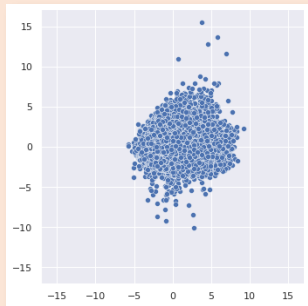
→ Finalité : garantir la fiabilité des segmentations sans pour autant de mises à jour excessives.

I. Problématique & pistes de recherches

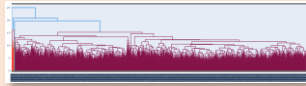
4. Les algorithmes et les métriques à disposition

Algorithmes de représentation

PCA



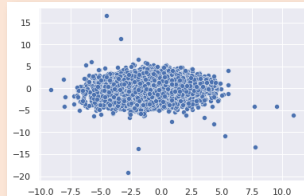
Dendrogramme



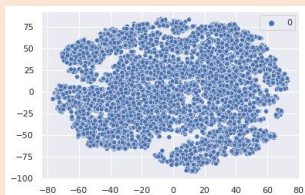
LLE



MDS



t-SNE

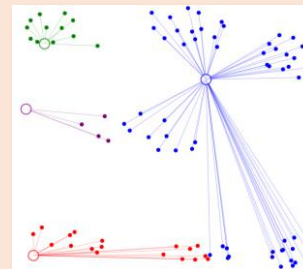


Métriques d'évaluation

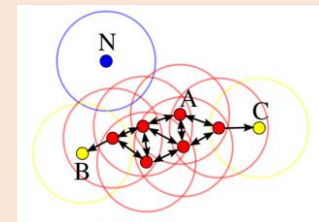
- Inertie (k-means)
- Coefficient de Silhouette
- Coefficient de Calinski-Harabasz
- Coefficient de Davies-Bouldin
- Proportion de données parasites (DBSCAN)

Algorithmes de segmentation

k-means / k-prototypes

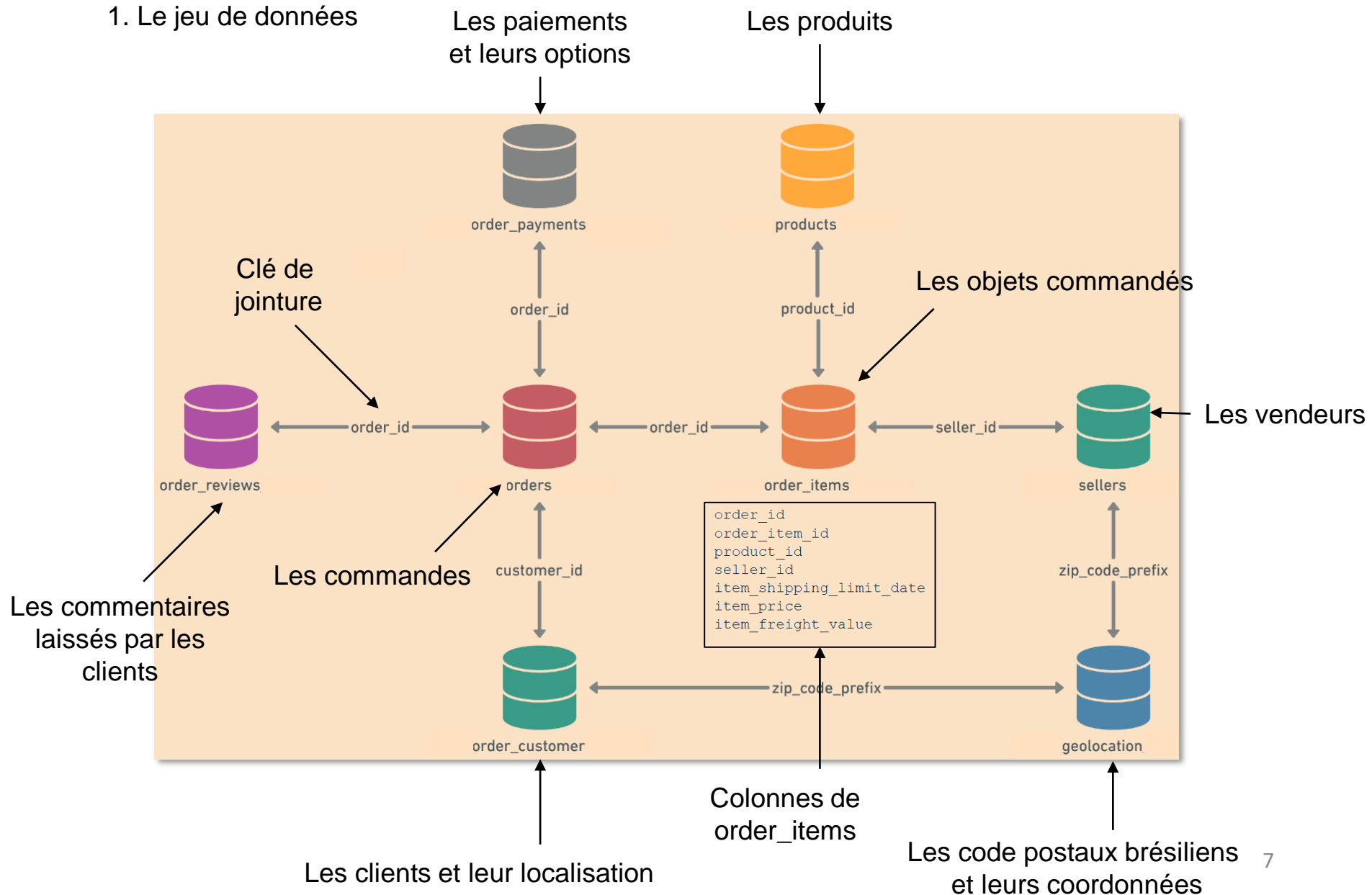


DBSCAN



II. Nettoyage, feature engineering et analyse exploratoire

1. Le jeu de données



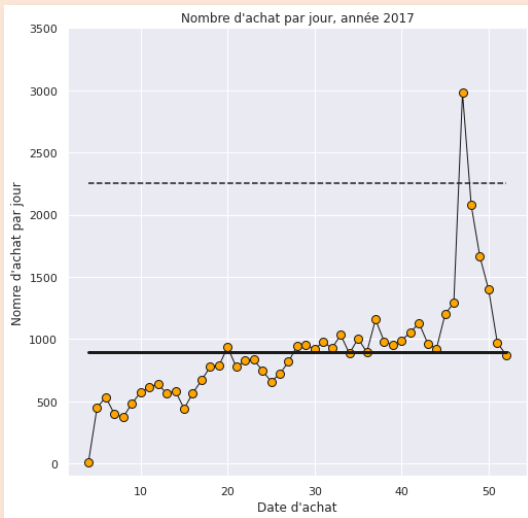
II. Nettoyage, feature engineering et analyse exploratoire

2. Le comportement utilisateur

L'analyse « RFM »

Recency

How recently did the customer purchase?



→ La date d'achat est présente dans le jeu de données final

Frequency

How often do they purchase?



→ La majorité des clients n'a passé qu'une seule commande. La fréquence d'achat ne concerne donc que les clients fidèles, minoritaires : elle n'est pas retenue pour la modélisation.

Monetary value

How much do they spend?

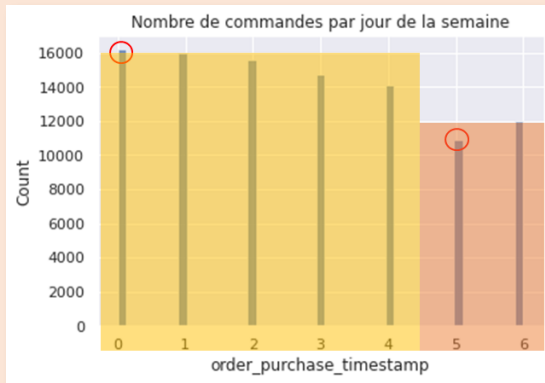


→ Les montants des paiements sont présents dans le jeu de données final.

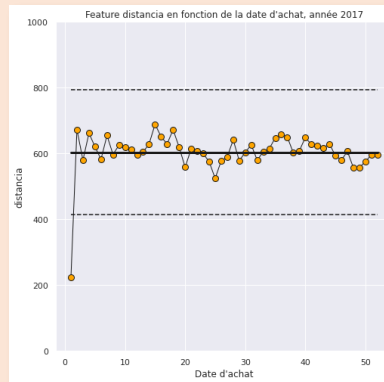
II. Nettoyage, feature engineering et analyse exploratoire

3. Les caractéristiques créées

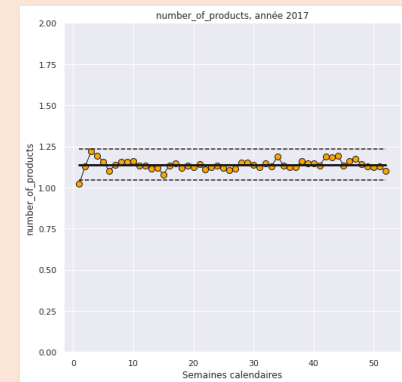
Jours de la semaine



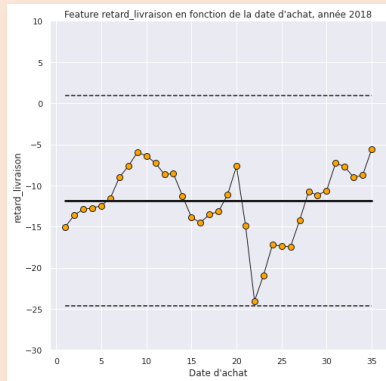
Distance



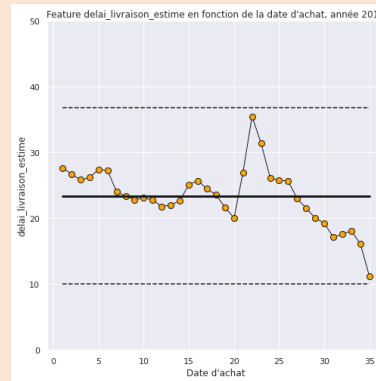
Nombre de produits par commande



Retard de livraison



Délai de livraison estimé



Caractéristiques écartées

Date du commentaire,

Date d'expédition,

Dimensions du produit,

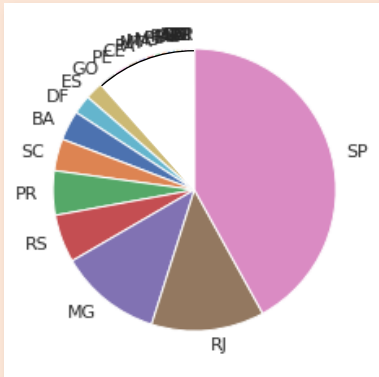
...

→ Certaines caractéristiques présentent des variations et / ou déséquilibres importants.

II. Nettoyage, feature engineering et analyse exploratoire

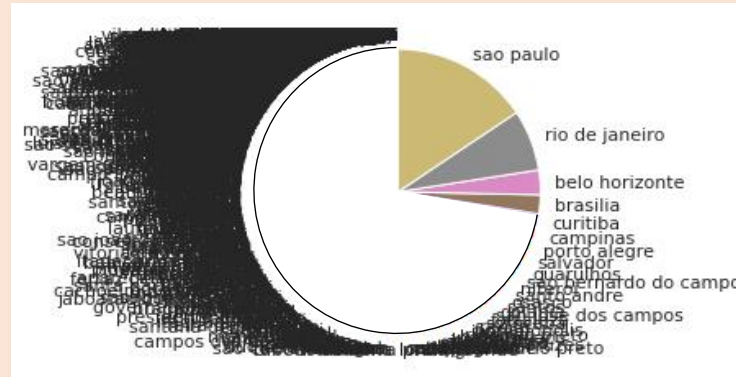
4. Les caractéristiques encodées

Etats : > 2% du total



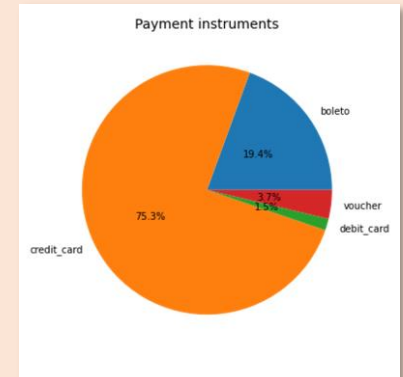
+ 9 caractéristiques

Villes : > 2% du total



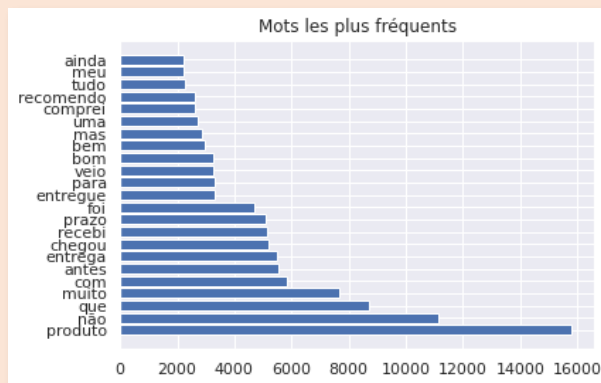
+ 3 caractéristiques

Moyens de paiement : tous



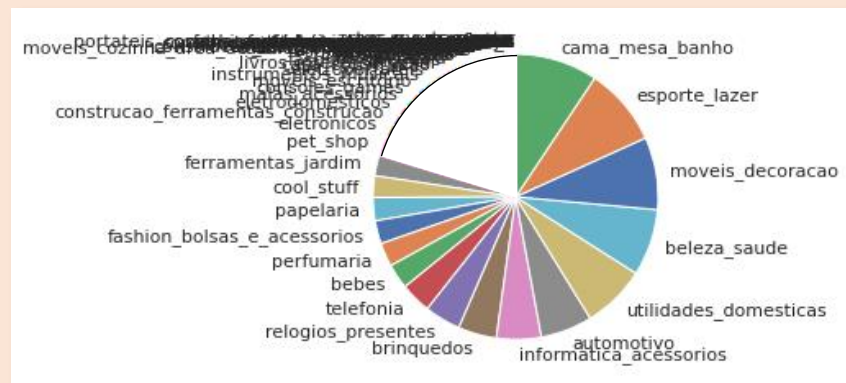
+ 3 caractéristiques

Commentaires : au cas par cas



+ 11 caractéristiques

Catégories de produit : > 2% du total

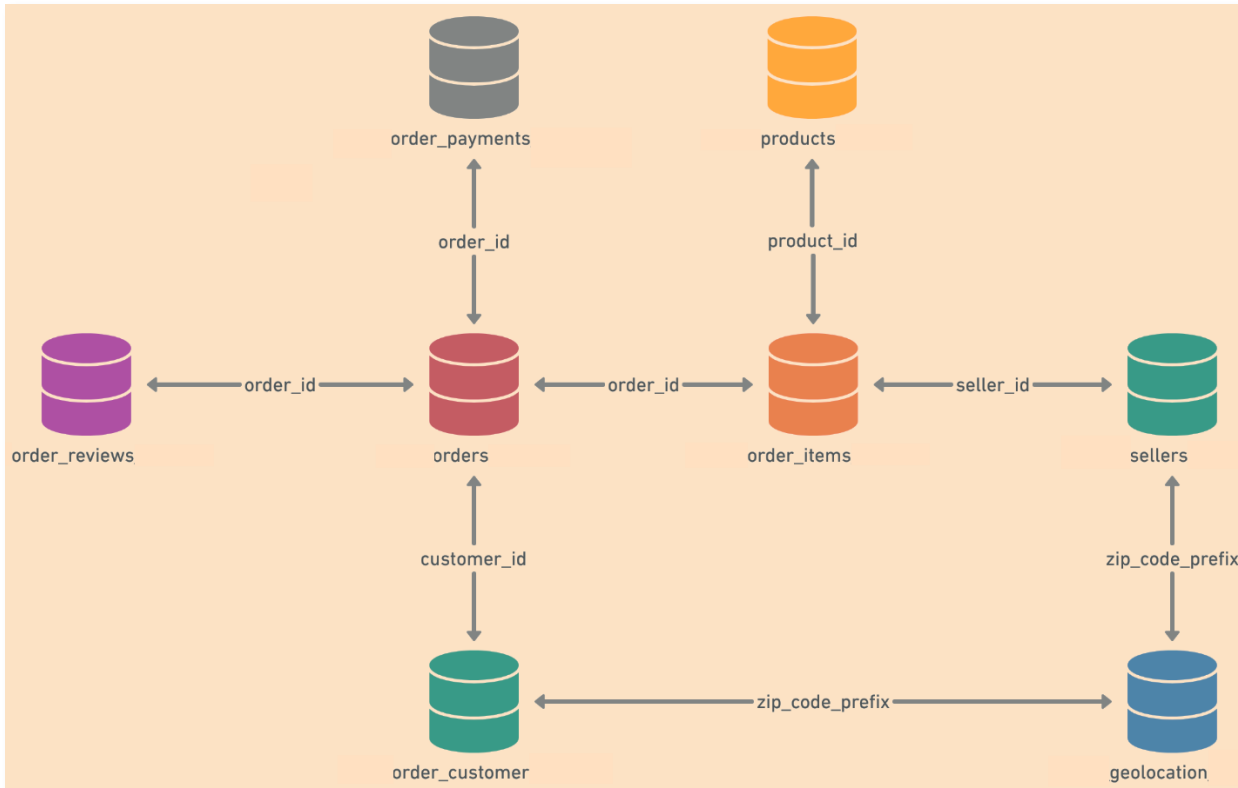


+ 15 caractéristiques

→ Bilan de l'encodage : 41 caractéristiques créées

II. Nettoyage, feature engineering et analyse exploratoire

5. La méthode de fusion



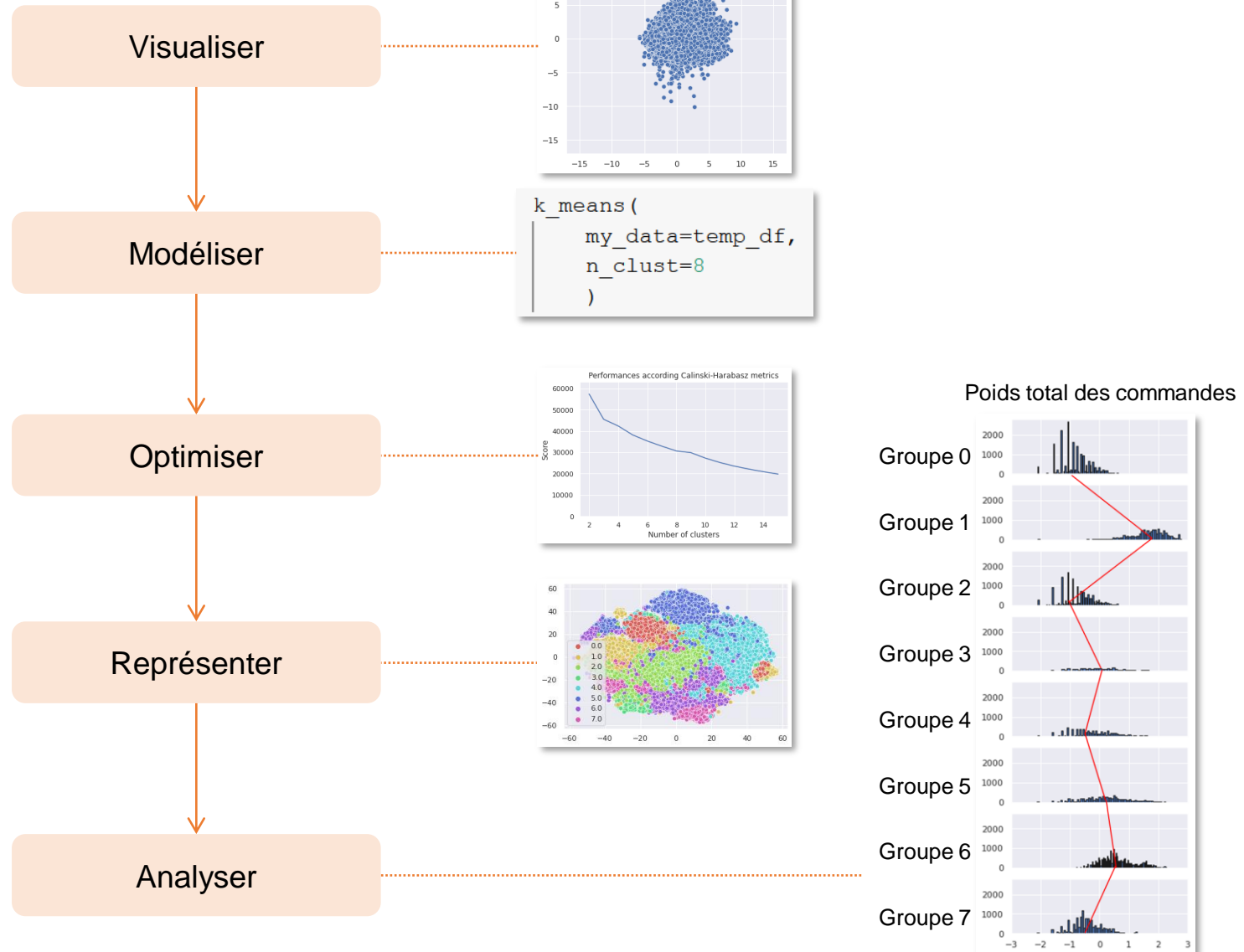
Méthode

1. Une seule ligne par commande
2. Pour chaque commande on garde en mémoire :
 - a. les moyens de paiement;
 - b. les catégories de produits;
 - c. les mots des commentaires.
3. Pour d'autres caractéristiques en revanche, on somme ou on moyenne :
 - a. le volume;
 - b. le poids;
 - c. la quantité de photos.

	 order_id	item_price	number_of_products	product_weight_g	product_volume_cm3	item_freight_value
0	e481f51cbdc54678b7cc49136f2d6af7	29.99	1	500.0	1976.0	8.72
1	128e10d95713541c87cd1a2e48201934	29.99	1	500.0	1976.0	7.78
2	0e7e841ddf8f2de2bad69267ecfbcf	29.99	1	500.0	1976.0	7.78
3	bfc39df4f36c3693ff3b63fcbca9e90a	29.99	1	500.0	1976.0	14.10
4	8736140c61ea584cb4250074756d8f3b	75.90	1	238.0	3000.0	7.79

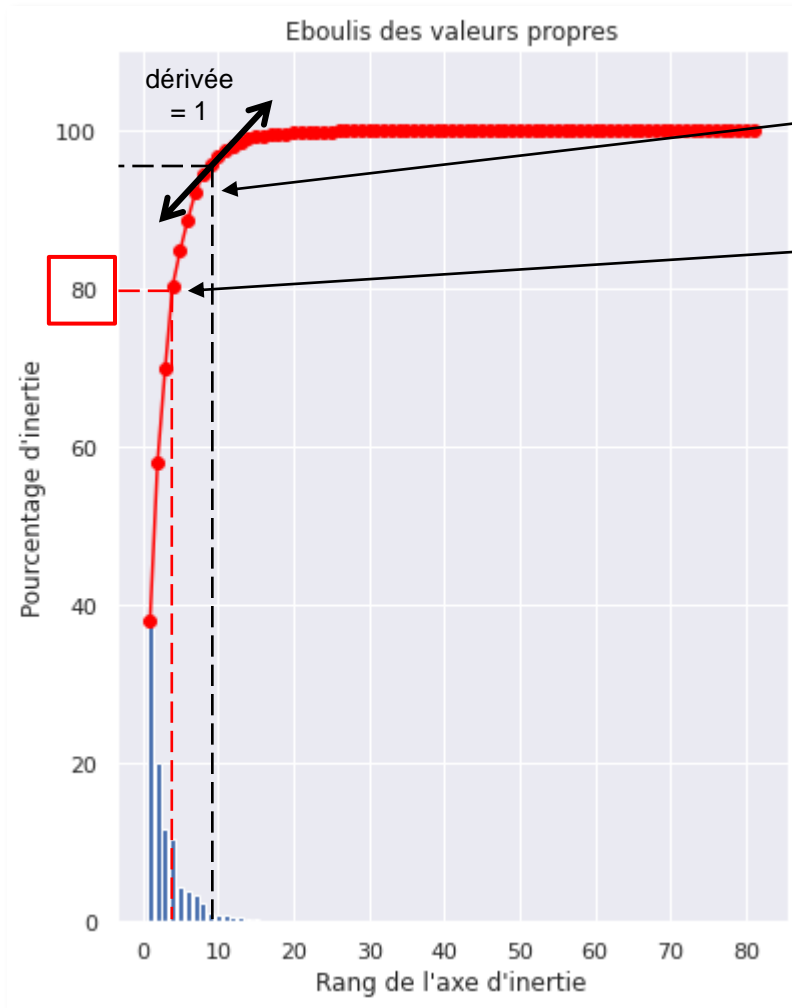
III. Pistes de modélisations

1. La démarche



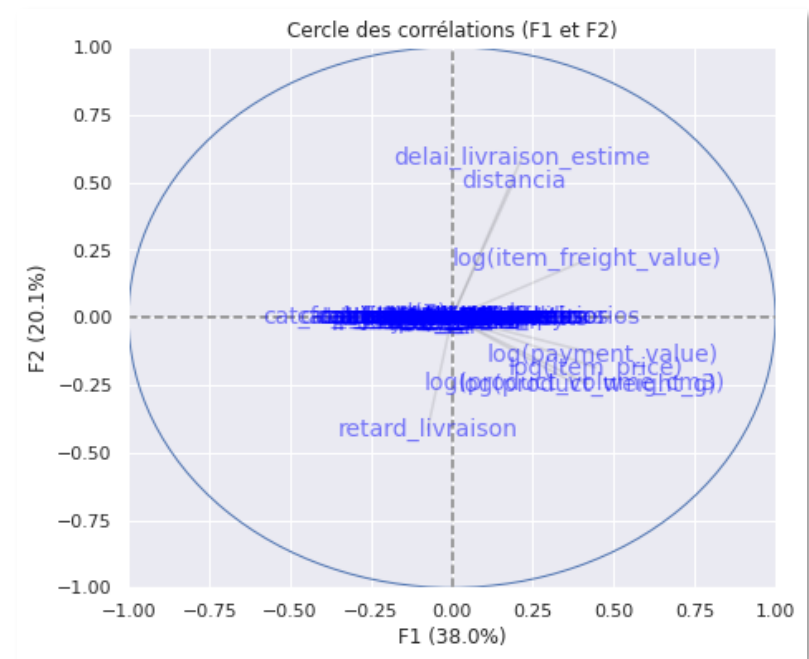
III. Pistes de modélisations

2. Analyse des composantes principales



Rythme d'apprentissage : ≈ 10 caractéristiques suffisent

Niveau d'apprentissage : ≈ 4 caractéristiques suffisent



→ On crée un nouveau jeu de données avec les 10 caractéristiques les plus importantes.

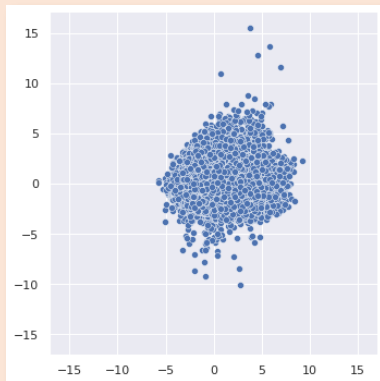
Les modélisations seront effectuées sur ce dernier, et parallèlement sur le jeu de donnée complet. 13

III. Pistes de modélisations

3. Les visualisations

Projection dans le plan principal

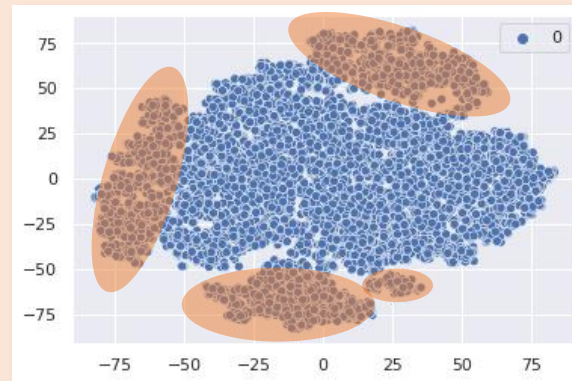
PCA



Interprétation : pas de segmentation discernable

Jeu de données complet

t-SNE

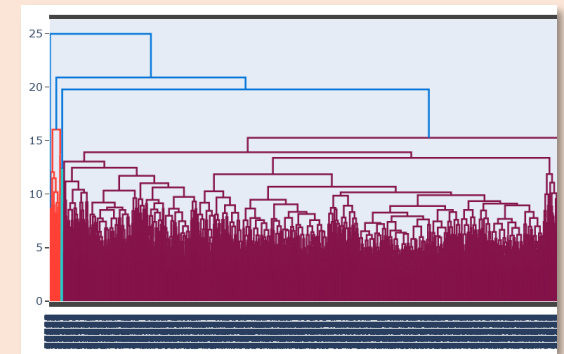


Interprétation : 5 groupes, peut-être plus

10% du jeu de données

Segmentation ascendante

Dendrogramme



Interprétation : 4 groupes, peut-être plus

10% du jeu de données

→ Rechercher entre 5 et 10 groupes sur le jeu de données complet

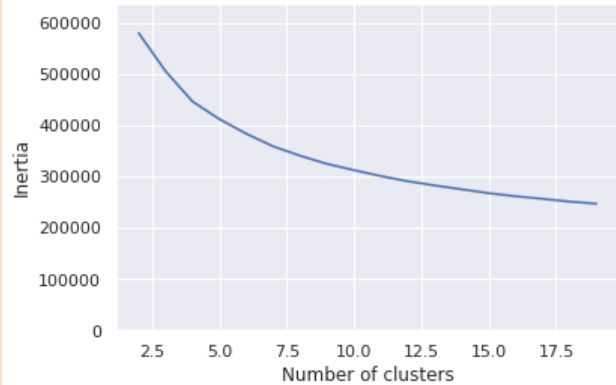
III. Pistes de modélisations

4. k-means

Les métriques

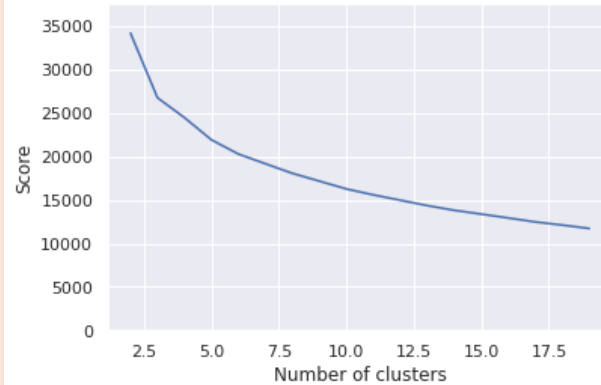
Inertie

Inertia according number of clusters



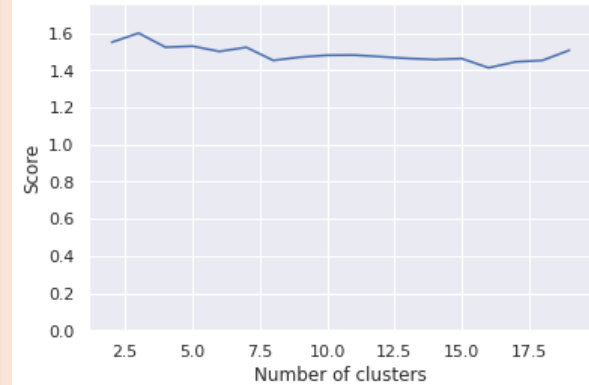
Calinski-Harabasz

Performances according Calinski-Harabasz metrics



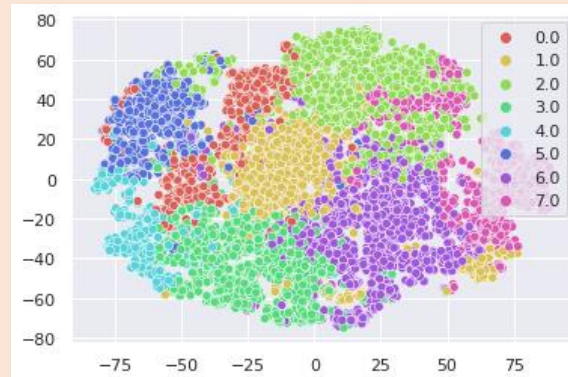
Davies-Bouldin

Performances according Davies-Bouldin metrics



(coefficient de Silhouette
très long à calculer)

Visualisation t-SNE sur 10% des données



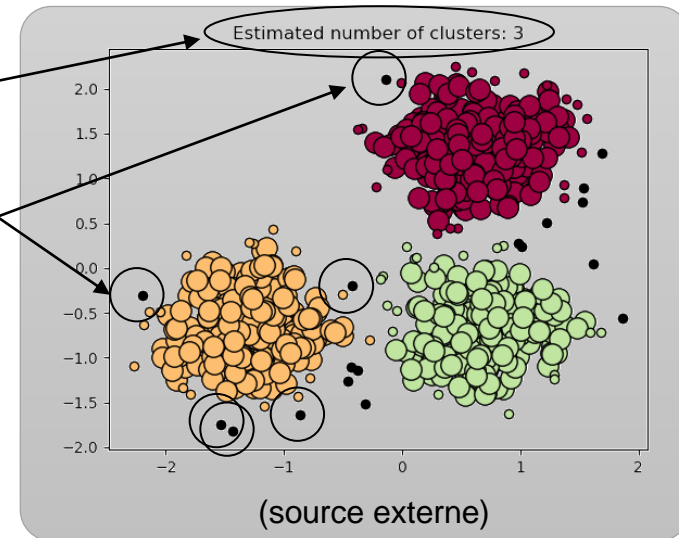
III. Pistes de modélisations

5. DBSCAN

L'algorithme DBSCAN fournit naturellement :

Le nombre de groupes identifiés

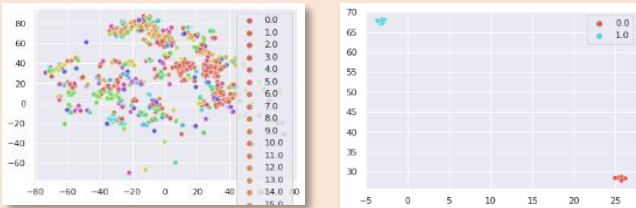
Le nombre de points de données parasites (« noise points »)



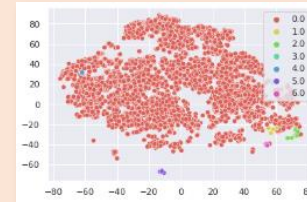
Inconvénients

DBSCAN est très long, et n'arrive pas à trouver un juste milieu :

Trop (360) ou trop peu (2) de groupes



Groupes de tailles déséquilibrées



Trop de points parasites (≈ 50.000)



Métriques peu convaincantes

Calinski-Harabasz coefficient: 814.382
Davies-Bouldin coefficient: 1.913

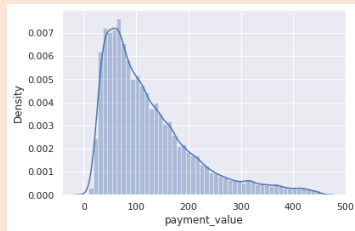
III. Pistes de modélisations

6. k-prototype et alternative

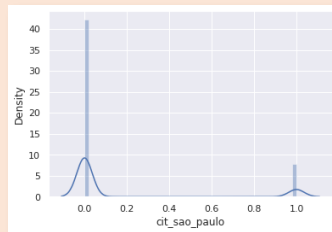
Problématique

k-means calcule des distances entre points. Mais dans le cas de nombreuses données encodées en 0 ou 1, le calcul de distance perd de sa pertinence.

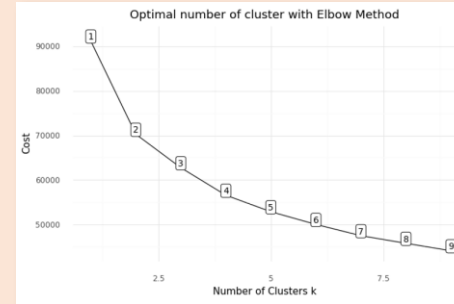
Répartition quantitative



Répartition binaire (données encodées)



Solution : l'algorithme k-prototype

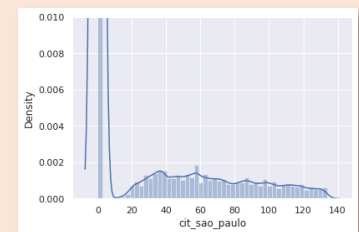


Inconvénient : très long, près de 25 minutes pour une seule modélisation.

Alternative: transformer les données encodées, pour leur donner une répartition similaire aux données quantitatives.

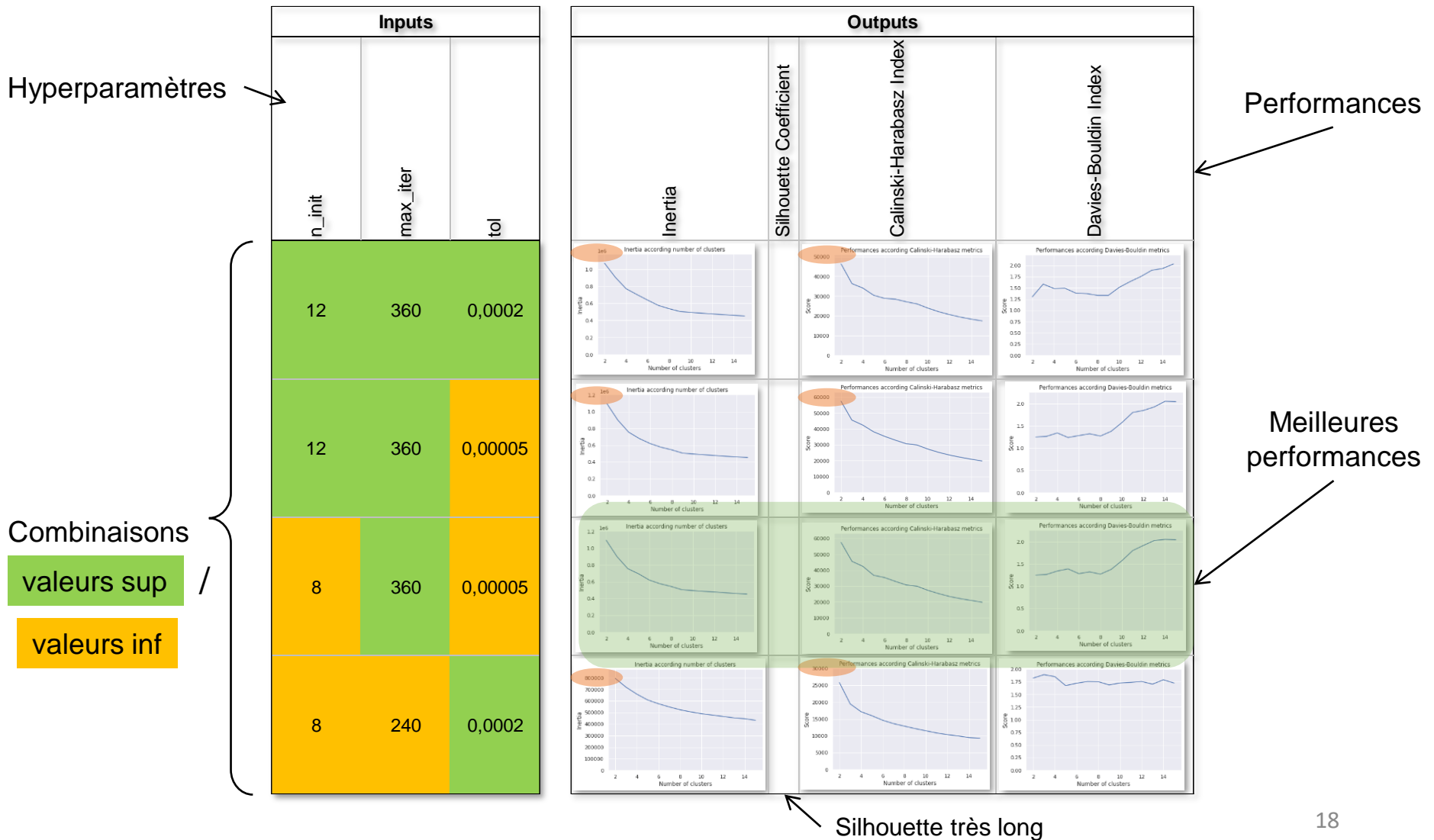
payment_value	sta_SP	sta_SP
38.71	1.0	38.71
37.77	1.0	37.77
37.77	1.0	37.77
44.09	0.0	0.00
83.69	1.0	83.69

Répartition modifiée



IV. Optimisation du modèle final

1. Le plan d'expérience



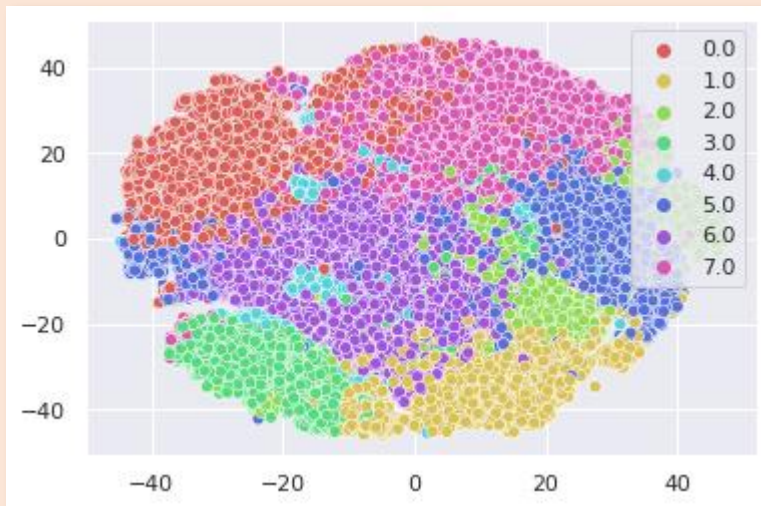
IV. Optimisation du modèle final

2. La visualisation des groupes

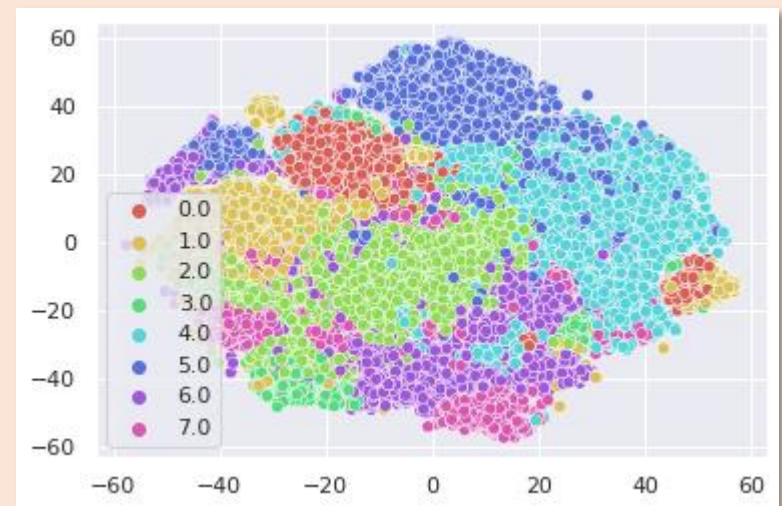
Projection dans le plan t-SNE

Coloration propre à chaque groupe identifié par k-means

Avec le PCA (10 caractéristiques)



Sans le PCA (71 caractéristiques)



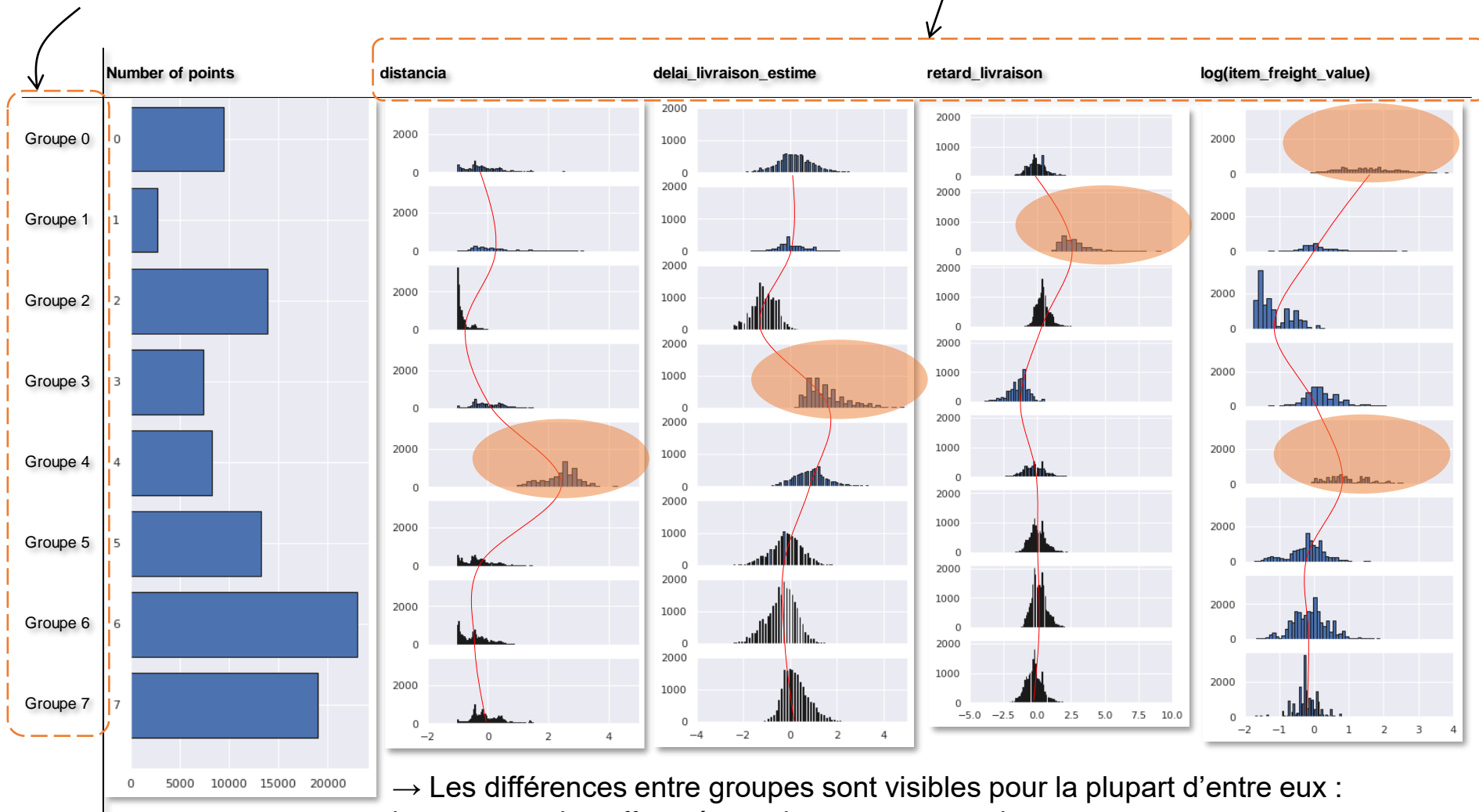
→ Cohérence entre l'algorithme de **visualisation** t-SNE et l'algorithme de **segmentation** k-means pour ce jeu de données

IV. Optimisation du modèle final

3. Vérification manuelle des différences entre groupes

Groupes identifiés par k-means

Quelques caractéristiques

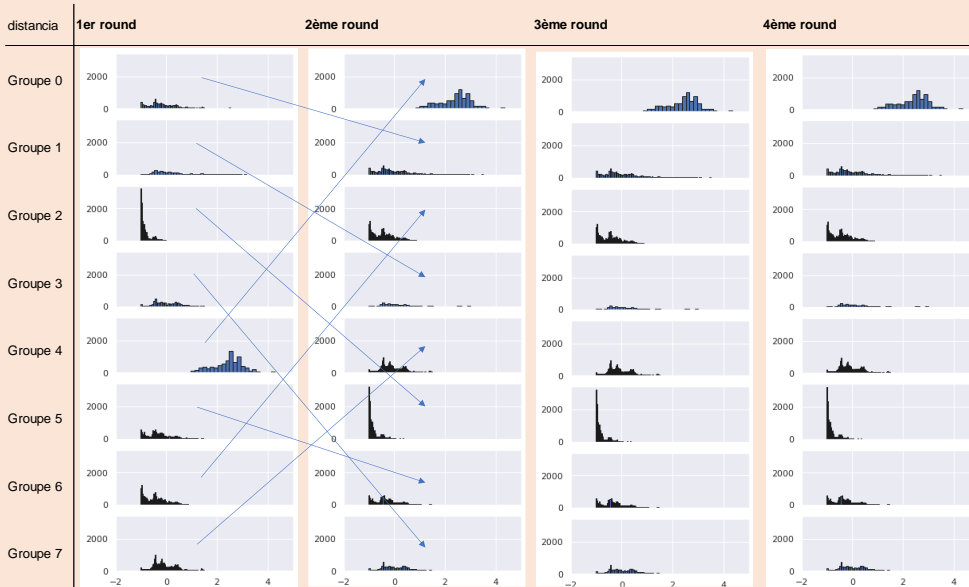


→ Les différences entre groupes sont visibles pour la plupart d'entre eux : la segmentation effectuée par k-means est pertinente.

IV. Optimisation du modèle final

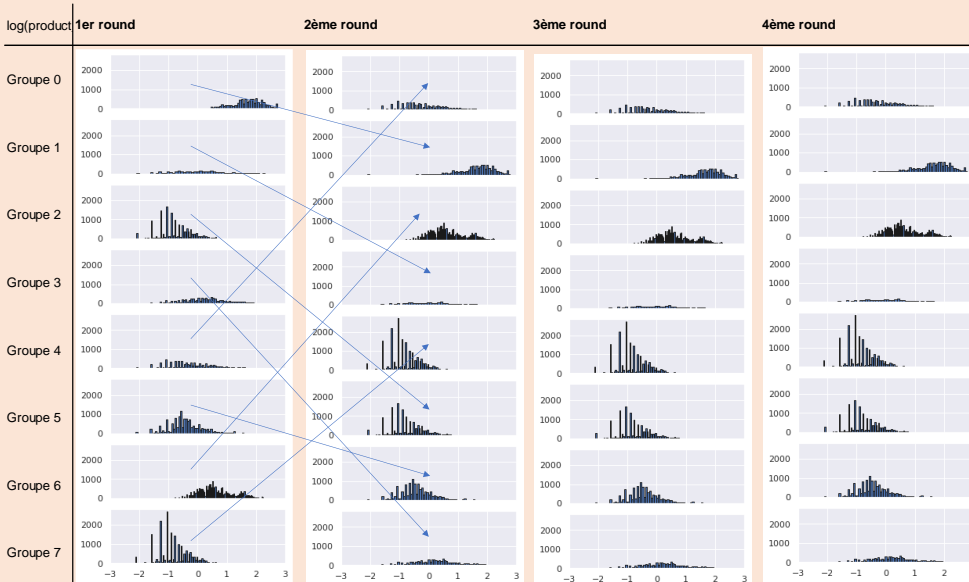
4. La stabilité de k-means

Distance



Segmentation stable


Poids



Segmentation stable

IV. Optimisation du modèle final

5. Le livrable



	Groupe 0	Groupe 1	Groupe 2	Groupe 3	Groupe 4	Groupe 5	Groupe 6	Groupe 7
Distance					Elevé			
Délai de livraison estimé				Très élevé	Elevé			
Retard de livraison		Très faible		Très élevé				
Frais de port	Très élevé		Faible		Elevé			
Prix par objet	Très élevé		Faible			Elevé		Faible
Montant de la commande	Très élevé		Faible			Elevé		Faible
Volume	Très élevé		Faible			Faible	Elevé	Faible
Poids	Très élevé		Très faible			Faible	Elevé	Faible
...

Livraisons très en avance

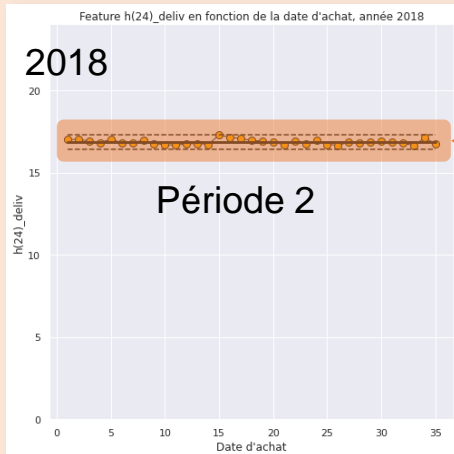
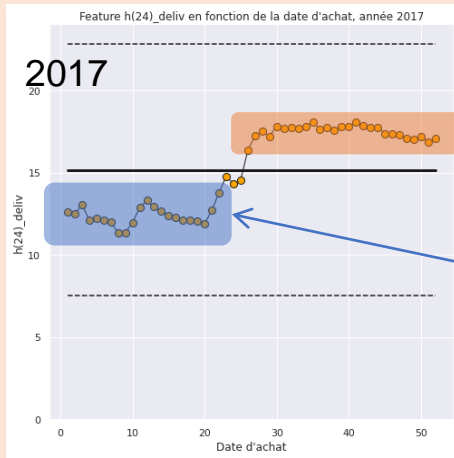
Livraisons très en retard

IV. Optimisation du modèle final

6. Le choix de la fréquence de mise à jour

Evolution de l'heure de livraison sur l'ensemble du jeu de données

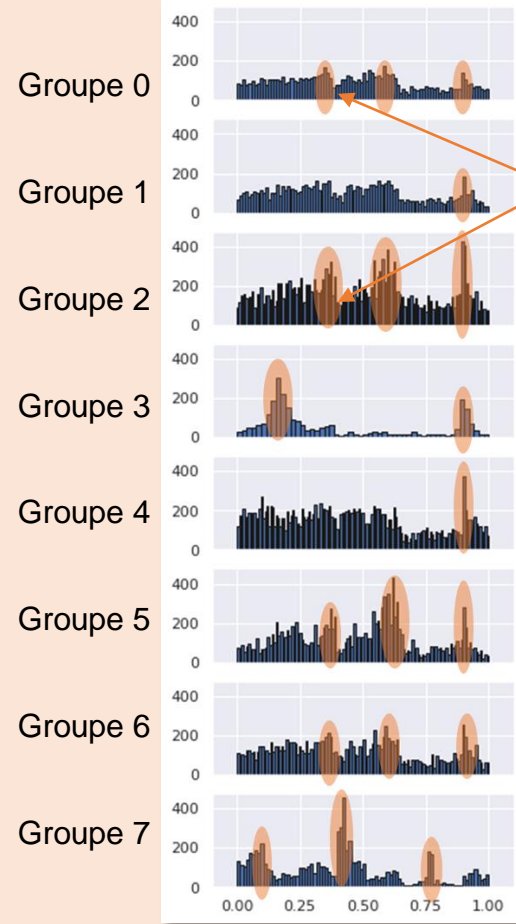
(→ ne nécessite pas de segmentation)



→ tendances
ponctuelles
→ relancer
l'algorithme selon les
variations brusques
des caractéristiques

Répartition des dates d'achat dans les différents groupes


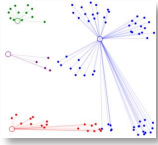
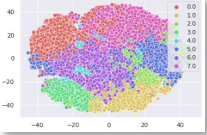


(→ nécessite une segmentation)



Tendance trimestrielle

→ tendances
générales
→ relancer
l'algorithme selon les
périodes identifiées

V. Bilan et perspectives

Sujet	Commentaire
 Le jeu de données	Le jeu de donnée est exploitable pour la problématique: le jeu final contient plus de 98.000 points de données, pour 100.000 originellement.
 Les algorithmes	Les différents algorithmes donnent des résultats similaires et concordent globalement sur la segmentation, à l'exception de DBSCAN (impropre pour ce jeu de donnée ?).
 La segmentation	Les groupes trouvés par k-means présentent chacun des particularités . La segmentation est stable au cours des relances successives.
 La fréquence de mise à jour	Mettre à jour l'algorithme tous les trimestres . Attention cependant à surveiller les variations affectant les caractéristiques .
 La vérification de l'efficacité	Recommandation : n'utiliser l'algorithme que sur la moitié des groupes, et évaluer l'impact des campagnes marketing sur les deux populations.

VI. Perspectives d'amélioration

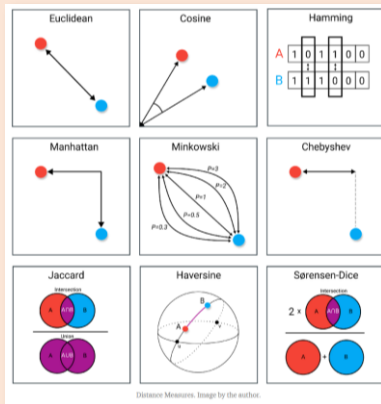
Etudier les clients fidèles



Analyser textuellement les commentaires



Utiliser d'autres types de distances



Utiliser UMAP



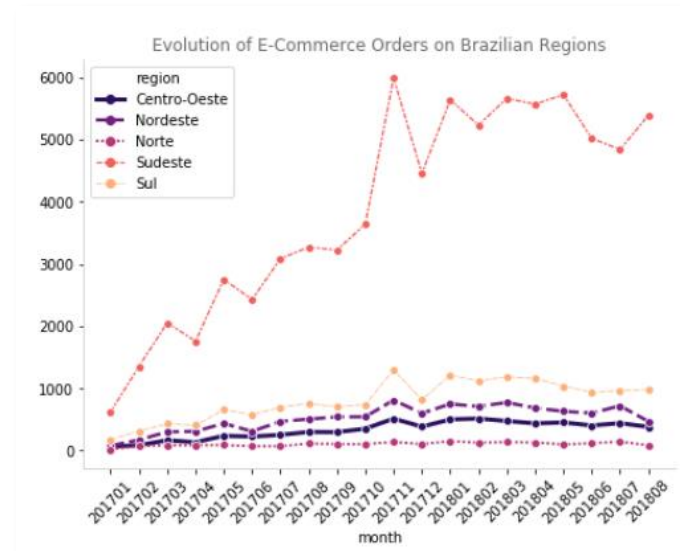
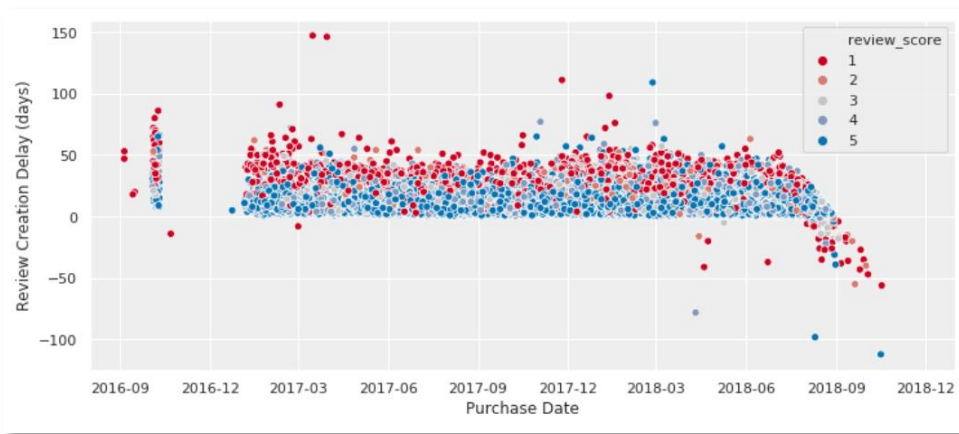
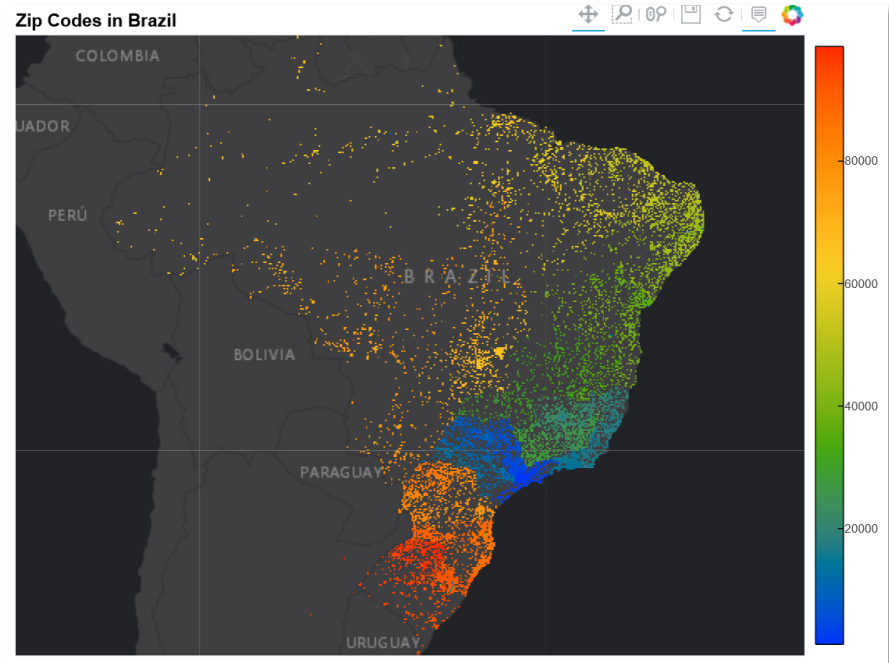
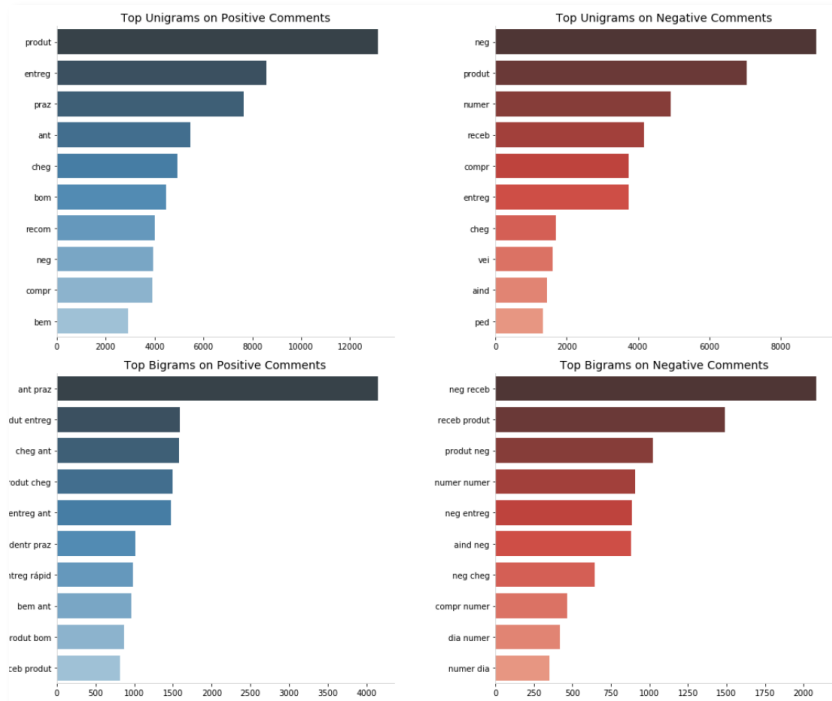
Utiliser les identifiants clients uniques



Fin de la présentation



Merci pour votre attention !



Nice! Another good view is to use the folium plugin [HeatMapWithTime](#) to see the evolution of e-commerce orders among time.

For [limitations](#) purpose (i.e. jupyter and Chrome limitations for total number of points shown at HeatMapWithTime, we will show the evolution of orders from January 2018 to July 2018)

Also, it's possible that the plugin HeatMapWithTime doesn't work properly from a [issue](#) fixed on version 0.11 (it's seems that the version of the kernel is 0.10). It is the case for you, just updating the version of folium library would fix it.

