

Formation Data Scientist

**Soutenance Projet 5 :
Segmentez des clients d'un site e-commerce**

-- Mélanie WARY --

The logo for 'olist' is displayed in white lowercase letters on a blue rectangular background.

PROBLÉMATIQUE

Olist, une solution de vente sur les marketplaces en ligne, souhaite fournir à ses équipes d'e-commerce une **segmentation des clients** qu'elles pourront utiliser au quotidien pour leurs campagnes de communication

MISSIONS

- (1) fournir à l'équipe marketing une **description actionnable de la segmentation**
- (2) fournir une **proposition de contrat de maintenance**

INTERPRÉTATION / PISTES DE RECHERCHE :

- **Clustering non supervisé**
 - ↳ Nombre de clusters ET interprétation de chaque cluster facilement exploitables
- **Analyse de la stabilité des segments au cours du temps**
 - ↳ pour évaluer la fréquence à laquelle la segmentation doit être mise à jour
- Données à disposition peu fournies = travail conséquent de **feature engineering**

DONNÉES À DISPOSITION

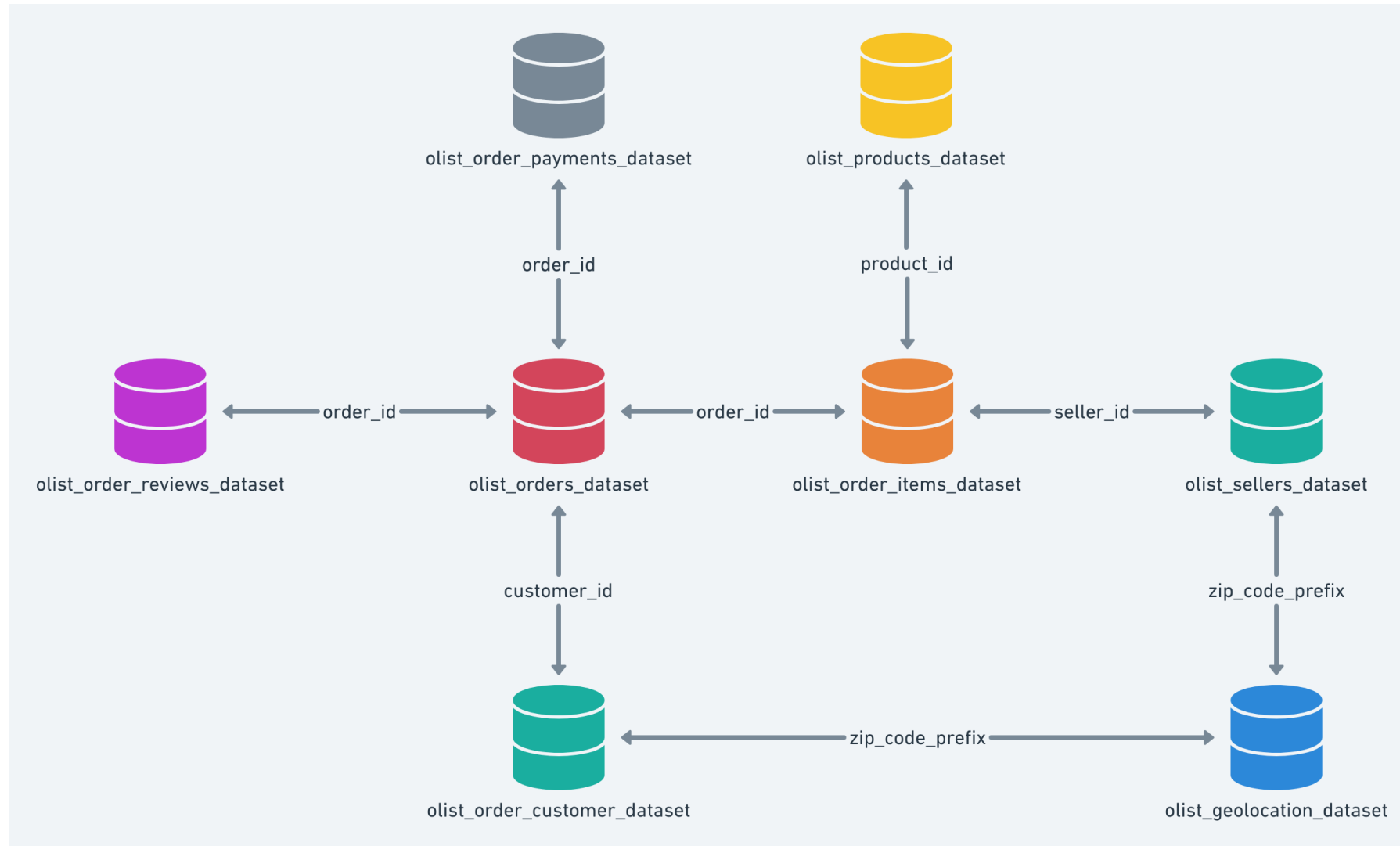
9 fichiers de données :

- ☐ olist_customers_dataset.csv
- ☐ olist_geolocation_dataset.csv
- ☐ olist_order_items_dataset.csv
- ☐ olist_order_payments_dataset.csv
- ☐ olist_order_reviews_dataset.csv
- ☐ olist_orders_dataset.csv
- ☐ olist_products_dataset.csv
- ☐ olist_sellers_dataset.csv
- ☐ product_category_name_translation.csv

96096 clients -- 99441 commandes -- 32951 produits -- 3095 vendeurs

1. Merge des fichiers de données

... de sorte à ne perdre aucune données



2. Cleaning

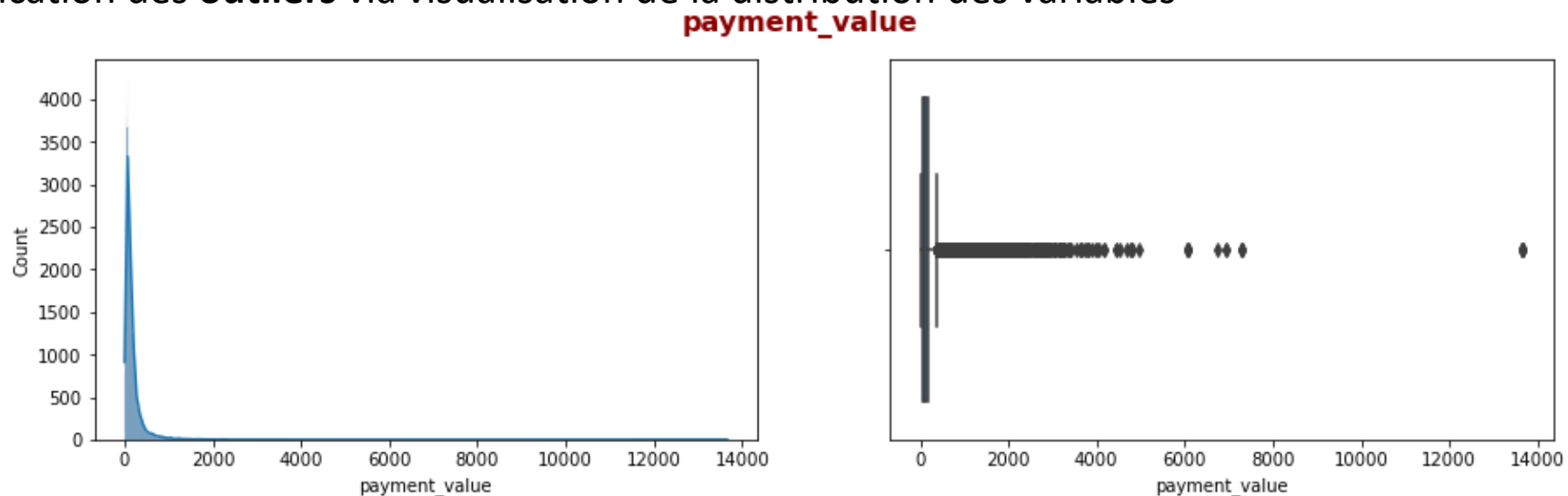
➤ Comblement des NaN et NaT

- Par la classe/valeur la plus fréquente ou moyenne (e.g. `payment_type`)
- Par calcul (e.g. `payment_value = price + freight_value` si `payment_sequential == 1`)
- Par combinaison calcul & valeur moyenne (e.g. `order_approved_at = order_purchase_timestamp + t` où `t` = temps moyen entre `order_purchase_timestamp` et `order_approved_at`)
- Par 'unknown' pour variables catégorielles (e.g. `seller_state`, `product_category`)

➤ Traitement des valeurs aberrantes (e.g. `payment_sequential == 2` seulement)

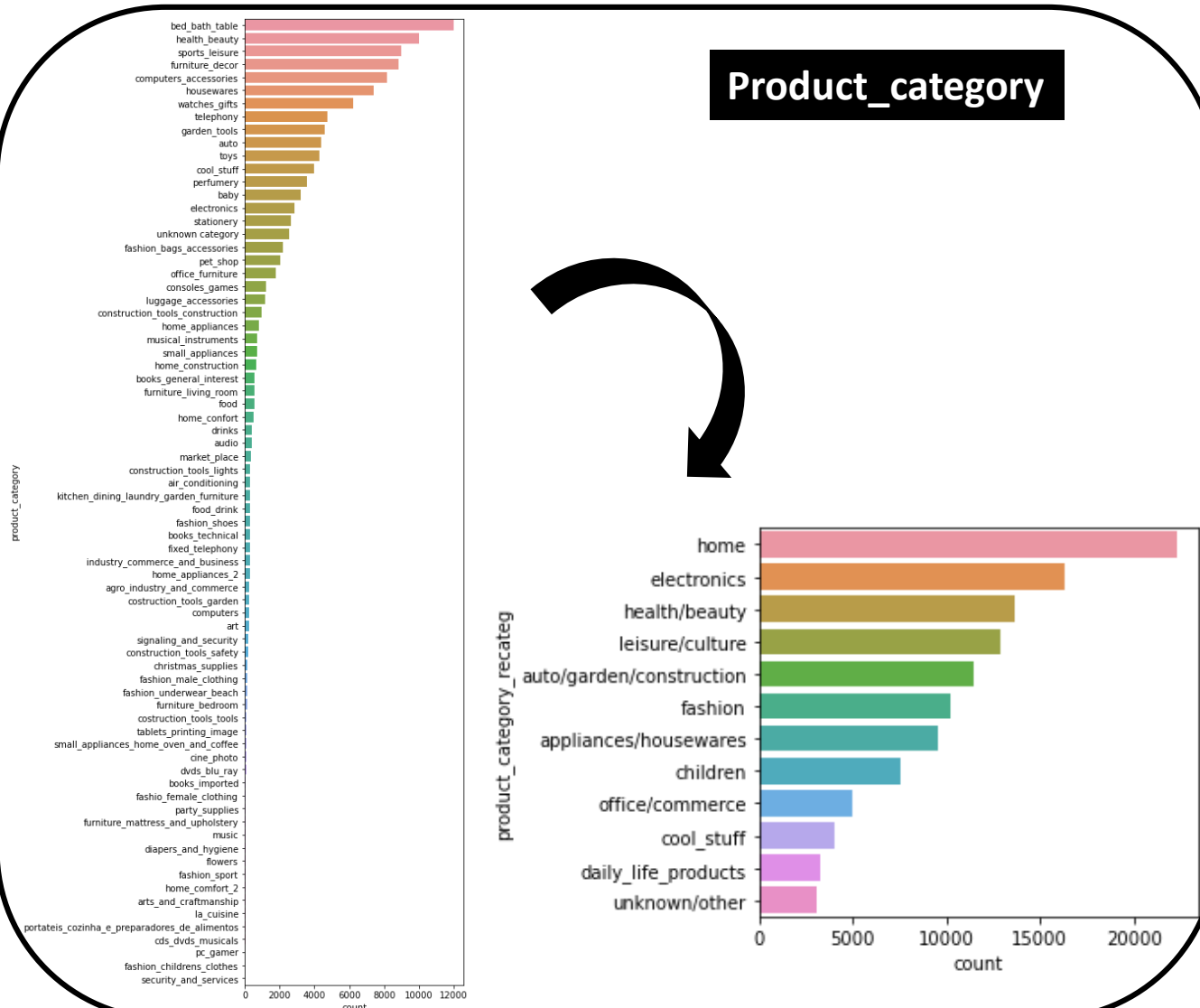
➤ Suppression colonnes non nécessaires

➤ Vérification des **outliers** via visualisation de la distribution des variables



3. Feature engineering

➤ Recatégorisation homogénéisée et simplifiée des variables catégorielles



3. Feature engineering

➤ **Recatégorisation homogénéisée et simplifiée** des variables catégorielles

➤ **Création d'un dataframe** où chaque ligne correspond à un client et regroupe les valeurs moyennes/modales/min/max des variables existantes + de **variables nouvellement créées**:

- | | | |
|--|--|--|
| <input type="checkbox"/> customer_region | <input type="checkbox"/> Recency (days) | <input type="checkbox"/> mean_nb_items |
| <input type="checkbox"/> mean_payment_sequential | <input type="checkbox"/> Seniority(days) | <input type="checkbox"/> mean_improvement- purchase_timedelta(mins) |
| <input type="checkbox"/> most_freq_payment_type | <input type="checkbox"/> Frequency(days) | <input type="checkbox"/> min_improvement- purchase_timedelta(mins) |
| <input type="checkbox"/> mean_payment_installments | <input type="checkbox"/> last_order_amount | <input type="checkbox"/> mean_estimated_delivery_tim elapse(days) |
| <input type="checkbox"/> most_freq_product_categ | <input type="checkbox"/> mean_order_amount | <input type="checkbox"/> Local_consumer_grade |
| <input type="checkbox"/> nb_product_categ | <input type="checkbox"/> max_order_amount | <input type="checkbox"/> use_of_voucher |
| <input type="checkbox"/> mean_review_score | <input type="checkbox"/> min_item_price | |
| <input type="checkbox"/> last_review_score | <input type="checkbox"/> max_item_price | |
| <input type="checkbox"/> nb_orders | <input type="checkbox"/> mean_item_price | |

3. Feature engineering

➤ **Recatégorisation homogénéisée et simplifiée** des variables catégorielles

➤ **Création d'un dataframe** où chaque ligne correspond à un client et regroupe les valeurs moyennes/modales/min/max des variables existantes + de **variables nouvellement créées**:

| | | |
|-----------------------------|---------------------|---|
| ■ customer_region | ■ Recency (days) | ■ mean_nb_items |
| ■ mean_payment_sequential | ■ Seniority(days) | ■ mean_improvement- purchase_timedelta(mins) |
| ■ most_freq_payment_type | ■ Frequency(days) | ■ min_improvement- purchase_timedelta(mins) |
| ■ mean_payment_installments | ■ last_order_amount | ■ mean_estimated_delivery_tim elapse(days) |
| ■ most_freq_product_categ | ■ mean_order_amount | ■ Local_consumer_grade |
| ■ nb_product_categ | ■ max_order_amount | ■ use_of_voucher |
| ■ mean_review_score | ■ min_item_price | |
| ■ last_review_score | ■ max_item_price | |
| ■ nb_orders | ■ mean_item_price | |

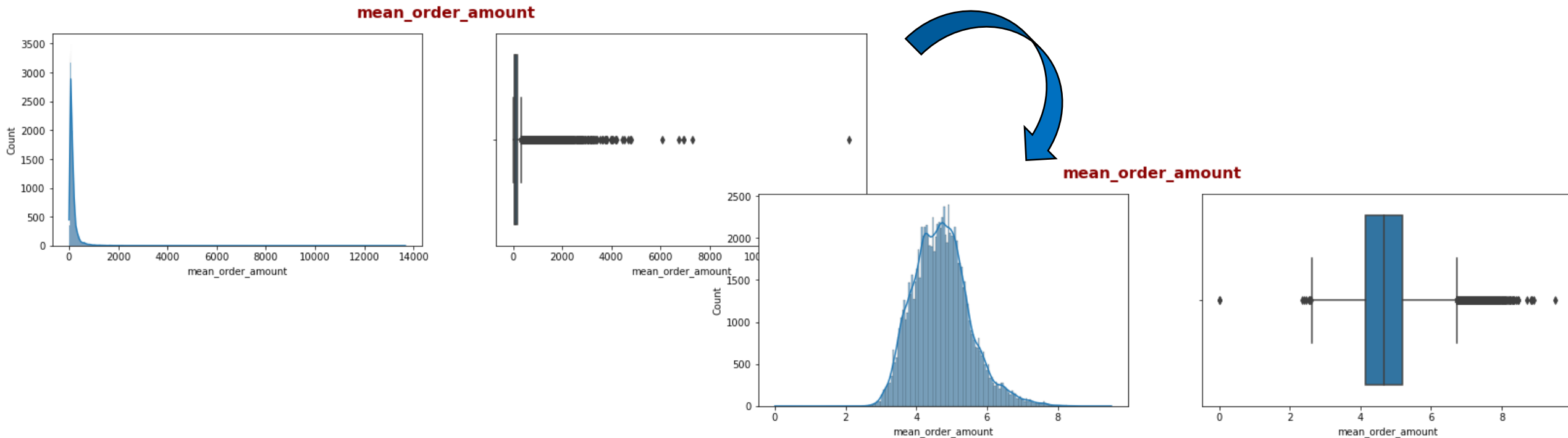
Localisation – **habitudes de paiement** – **dynamique de commande** – **intérêts de consommation**
– **moyens financiers** – **satisfaction sur olist**

3. Feature engineering

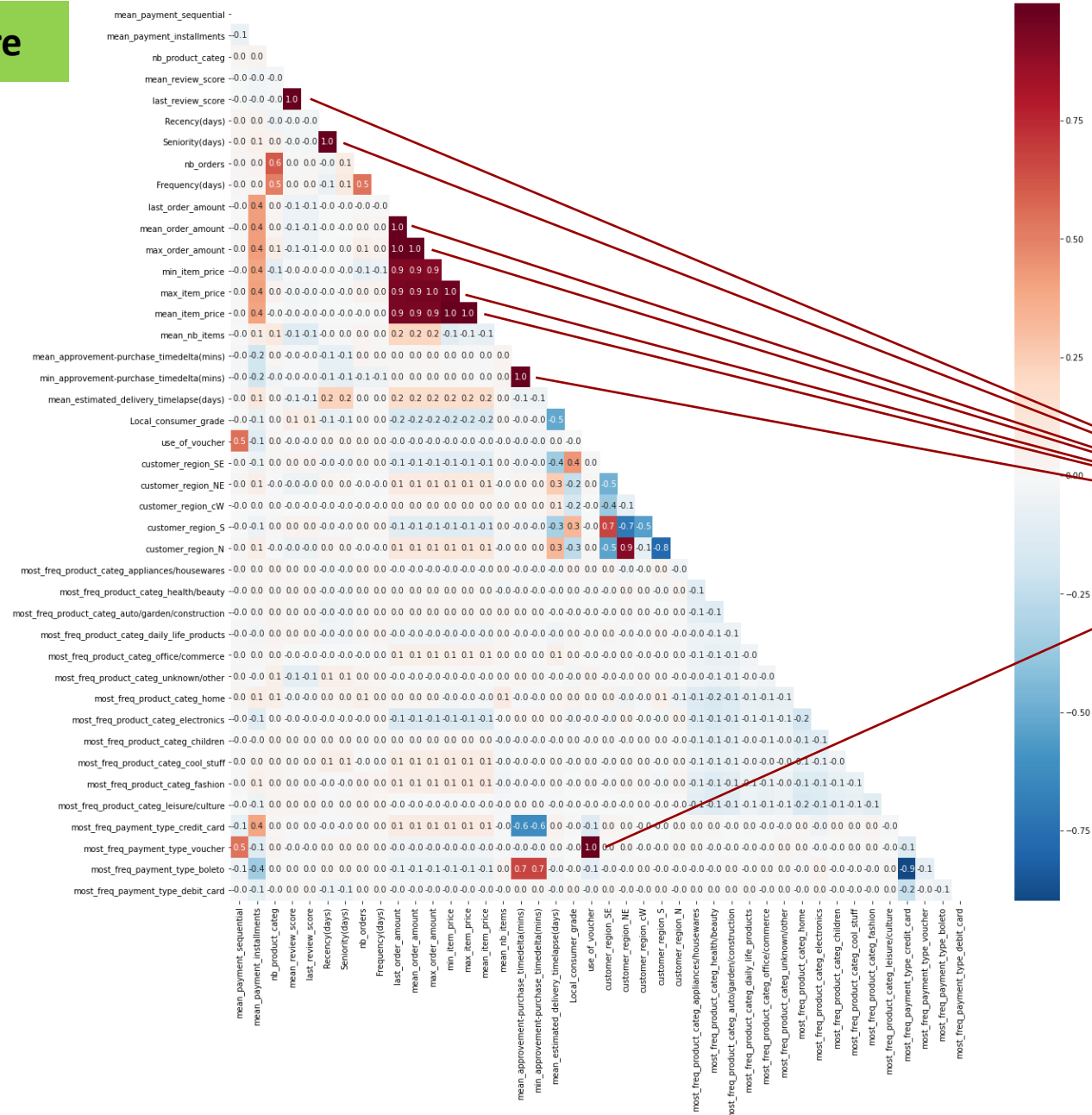
- **Recatégorisation homogénéisée et simplifiée** des variables catégorielles
- **Création d'un dataframe** où chaque ligne correspond à un client et regroupe les valeurs moyennes/modales/min/max des variables existantes + de **variables nouvellement créées**:
 - **customer_region**
 - ❑ mean_payment_sequential
 - **most_freq_payment_type**
 - ❑ mean_payment_installments
 - **most_freq_product_categ**
 - ❑ nb_product_categ
 - ❑ mean_review_score
 - ❑ last_review_score
 - ❑ nb_orders
 - ❑ Recency (days)
 - ❑ Seniority(days)
 - ❑ Frequency(days)
 - ❑ last_order_amount
 - ❑ mean_order_amount
 - ❑ max_order_amount
 - ❑ min_item_price
 - ❑ max_item_price
 - ❑ mean_item_price
 - ❑ mean_nb_items
 - ❑ mean_improvement_purchase_timedelta(mins)
 - ❑ min_improvement_purchase_timedelta(mins)
 - ❑ mean_estimated_delivery_time_elapse(days)
 - ❑ Local_consumer_grade
 - ❑ use_of_voucher
- **Encodage des variables catégorielles** via sorte de One Hot Encoding modifié pour prendre en compte la présence de valeurs de type liste (et non de type simple string) dans ces colonnes

3. Feature engineering

- **Recatégorisation homogénéisée et simplifiée** des variables catégorielles
- **Création d'un dataframe** où chaque ligne correspond à un client et regroupe les valeurs moyennes/modales/min/max des variables existantes + de **variables nouvellement créées**:
- **Encodage des variables catégorielles**
- **Transformation logarithmique de variables numériques** si nécessaire



4. Analyse exploratoire



Suppression
variables
redondantes
(coeff correl = 1)

Suppression variables redondantes

- | | | |
|---|---|---|
| ■ customer_region | <input checked="" type="checkbox"/> Recency (days) | <input checked="" type="checkbox"/> mean_nb_items |
| <input checked="" type="checkbox"/> mean_payment_sequential | <input checked="" type="checkbox"/> Seniority(days) | <input checked="" type="checkbox"/> mean_improvement- purchase_timedelta(mins) |
| ■ most_freq_payment_type | <input checked="" type="checkbox"/> Frequency(days) | <input checked="" type="checkbox"/> min_improvement- purchase_timedelta(mins) |
| <input checked="" type="checkbox"/> mean_payment_installments | <input checked="" type="checkbox"/> last_order_amount | <input checked="" type="checkbox"/> mean_estimated_delivery_tim elapsed(days) |
| ■ most_freq_product_categ | <input checked="" type="checkbox"/> mean_order_amount | <input checked="" type="checkbox"/> Local_consumer_grade |
| <input checked="" type="checkbox"/> nb_product_categ | <input checked="" type="checkbox"/> max_order_amount | <input checked="" type="checkbox"/> use_of_voucher |
| <input checked="" type="checkbox"/> mean_review_score | <input checked="" type="checkbox"/> min_item_price | |
| <input checked="" type="checkbox"/> last_review_score | <input checked="" type="checkbox"/> max_item_price | |
| <input checked="" type="checkbox"/> nb_orders | <input checked="" type="checkbox"/> mean_item_price | |

12 variables numériques

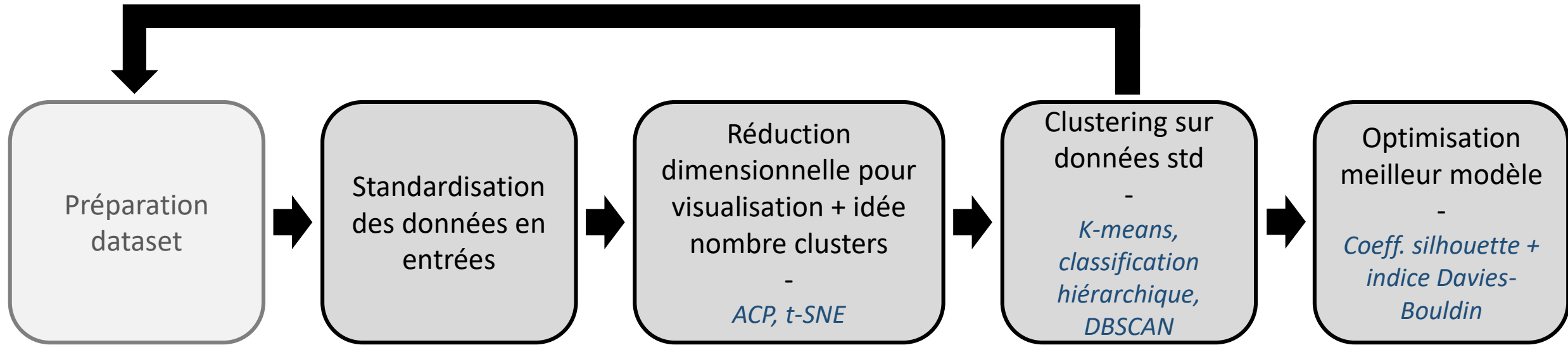
+

21 variables issues encodage
des 3 variables catégorielles

=

33 variables
pour clustering

Clustering : méthodologie



1. Clustering hiérarchique

NON ADAPTÉ

3.3. Clustering hiérarchique

```
Entrée [357]: hierarch_clusterer = AgglomerativeClustering(n_clusters=19)
             hierarch_cluster_labels = hierarch_clusterer.fit_predict(data_std)
             np.unique(hierarch_cluster_labels)
             # MEMORY ERROR: non adapté aux échantillons avec de nbx individus
```

MemoryError

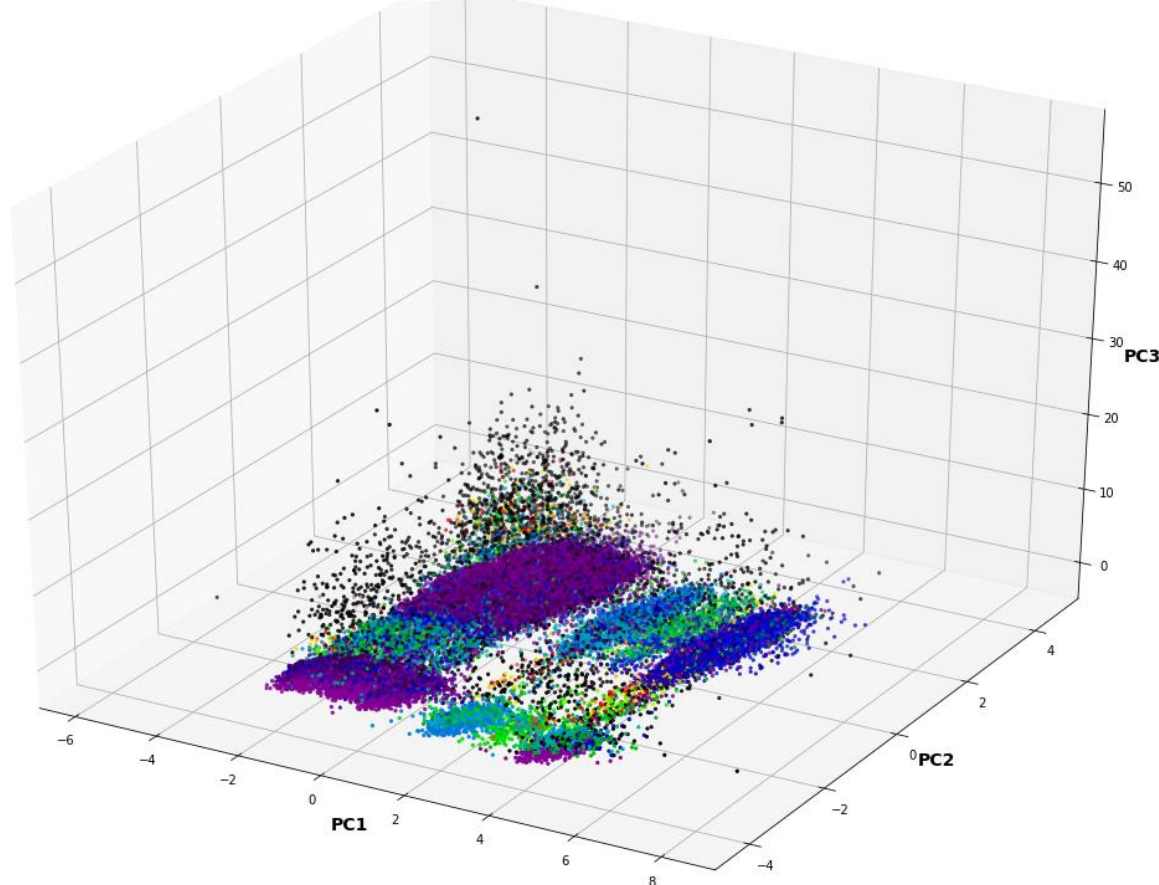
Traceback (most recent call last)

➔ Non adapté aux échantillons avec de nombreux individus, or notre dataset comprend ca. 100 000 individus

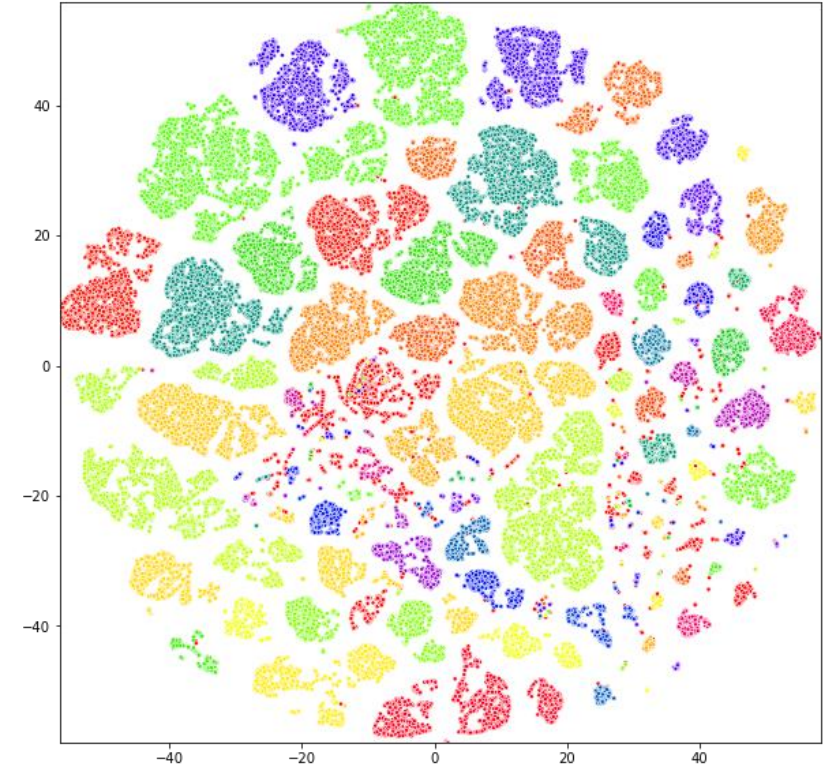
2. DBSCAN

Nombre de clusters élevé...

Représentation des 226 clusters (DBSCAN eps=3, min_samples=5) sur les axes de la PCA



Représentation des 226 clusters (DBSCAN eps=3, min_samples=5) via t-SNE à 2 composantes

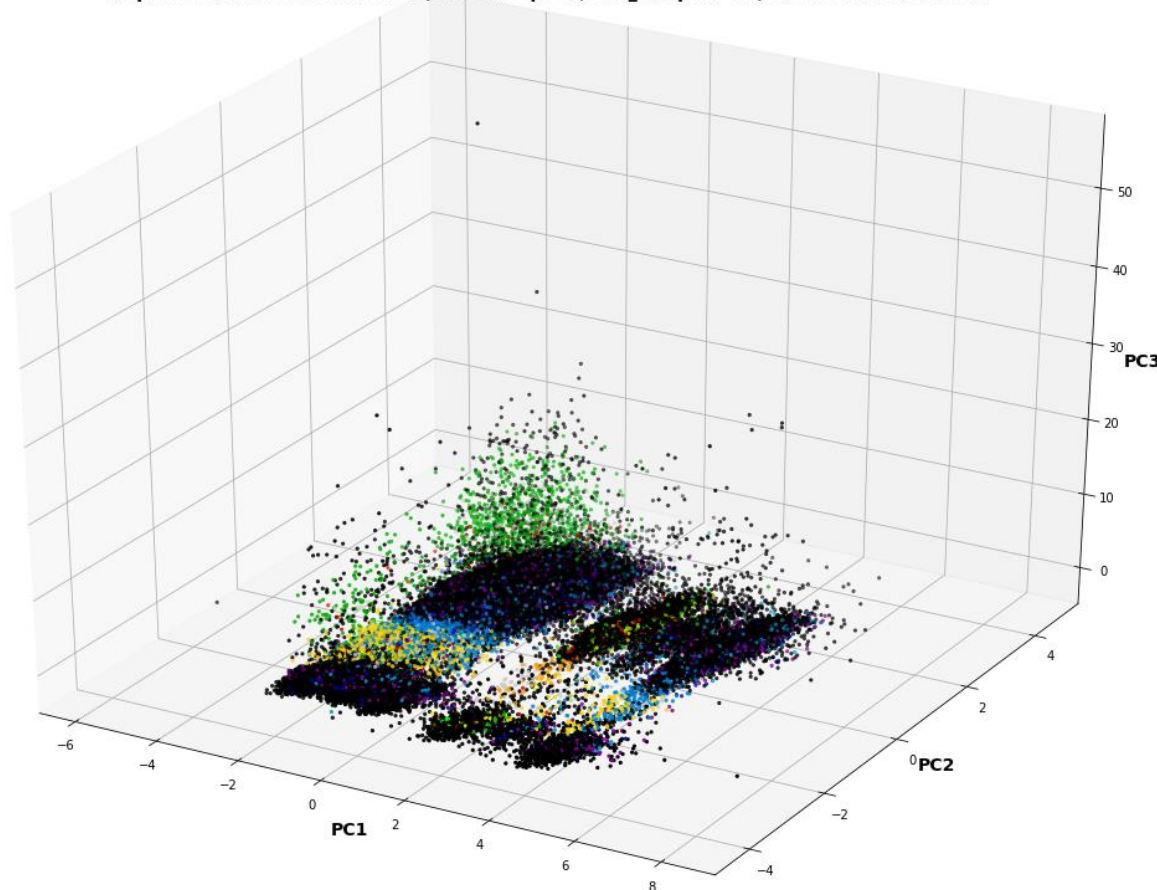


| | | | | | | | | | | |
|------|------|------|------|-------|-------|-------|-------|-------|-------|-------|
| • -1 | • 20 | • 41 | • 62 | • 83 | • 104 | • 125 | • 146 | • 166 | • 186 | • 206 |
| • 0 | • 21 | • 42 | • 63 | • 84 | • 105 | • 126 | • 147 | • 167 | • 187 | • 207 |
| • 1 | • 22 | • 43 | • 64 | • 85 | • 106 | • 127 | • 148 | • 168 | • 188 | • 208 |
| • 2 | • 23 | • 44 | • 65 | • 86 | • 107 | • 128 | • 149 | • 169 | • 189 | • 209 |
| • 3 | • 24 | • 45 | • 66 | • 87 | • 108 | • 129 | • 150 | • 170 | • 190 | • 210 |
| • 4 | • 25 | • 46 | • 67 | • 88 | • 109 | • 130 | • 151 | • 171 | • 191 | • 211 |
| • 5 | • 26 | • 47 | • 68 | • 89 | • 110 | • 131 | • 152 | • 172 | • 192 | • 212 |
| • 6 | • 27 | • 48 | • 69 | • 90 | • 111 | • 132 | • 153 | • 173 | • 193 | • 213 |
| • 7 | • 28 | • 49 | • 70 | • 91 | • 112 | • 133 | • 154 | • 174 | • 194 | • 214 |
| • 8 | • 29 | • 50 | • 71 | • 92 | • 113 | • 134 | • 155 | • 175 | • 195 | • 215 |
| • 9 | • 30 | • 51 | • 72 | • 93 | • 114 | • 135 | • 156 | • 176 | • 196 | • 216 |
| • 10 | • 31 | • 52 | • 73 | • 94 | • 115 | • 136 | • 157 | • 177 | • 197 | • 217 |
| • 11 | • 32 | • 53 | • 74 | • 95 | • 116 | • 137 | • 158 | • 178 | • 198 | • 218 |
| • 12 | • 33 | • 54 | • 75 | • 96 | • 117 | • 138 | • 159 | • 179 | • 199 | • 219 |
| • 13 | • 34 | • 55 | • 76 | • 97 | • 118 | • 139 | • 160 | • 180 | • 200 | • 220 |
| • 14 | • 35 | • 56 | • 77 | • 98 | • 119 | • 140 | • 161 | • 181 | • 201 | • 221 |
| • 15 | • 36 | • 57 | • 78 | • 99 | • 120 | • 141 | • 162 | • 182 | • 202 | • 222 |
| • 16 | • 37 | • 58 | • 79 | • 100 | • 121 | • 142 | • 163 | • 183 | • 203 | • 223 |
| • 17 | • 38 | • 59 | • 80 | • 101 | • 122 | • 143 | • 164 | • 184 | • 204 | • 224 |
| • 18 | • 39 | • 60 | • 81 | • 102 | • 123 | • 144 | • 165 | • 185 | • 205 | • 225 |
| • 19 | • 40 | • 61 | • 82 | • 103 | • 124 | • 145 | | | | |

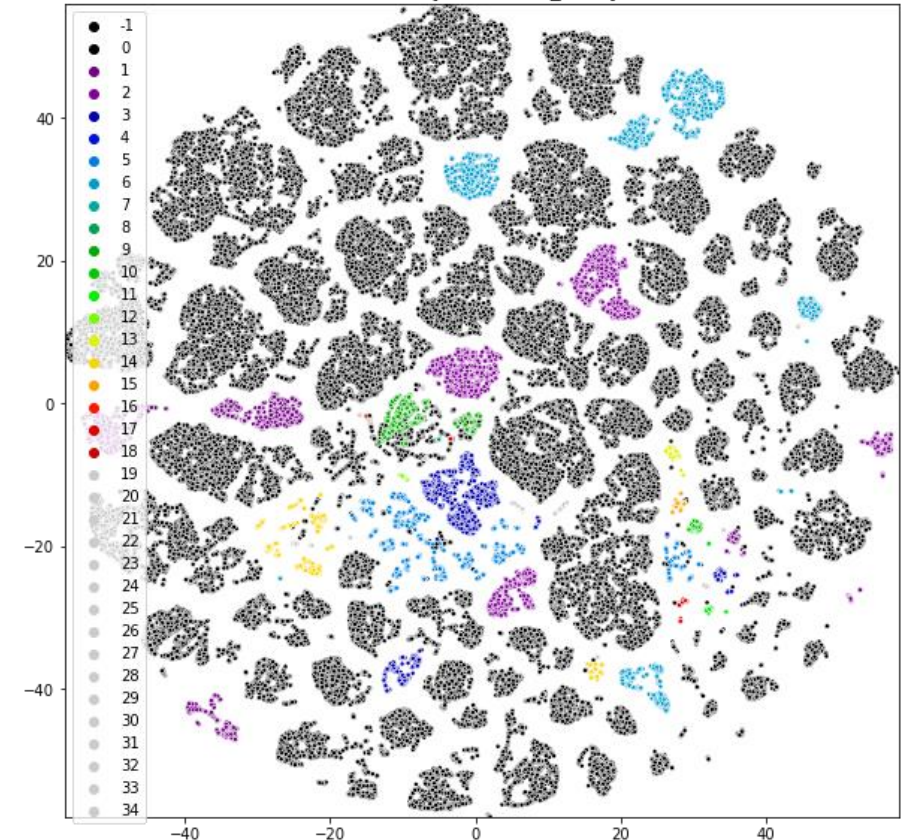
2. DBSCAN

Nombre de clusters élevé... même après optimisation hyperparamètres (35 au mieux)...

Représentation des 35 clusters (DBSCAN eps=5, min_samples=10) sur les axes de la PCA



Représentation des 35 clusters (DBSCAN eps=5, min_samples=10) via t-SNE à 2 composantes



The logo for 'olist' is displayed in white text on a blue rectangular background.

PROBLÉMATIQUE

Olist, une solution de vente sur les marketplaces en ligne, souhaite fournir à ses équipes d'e-commerce une **segmentation des clients** qu'elles pourront utiliser au quotidien pour leurs campagnes de communication

MISSIONS

- (1) fournir à l'équipe marketing une **description actionnable de la segmentation**
- (2) Fournir une **proposition de contrat de maintenance**

INTERPRÉTATION / PISTES DE RECHERCHE :

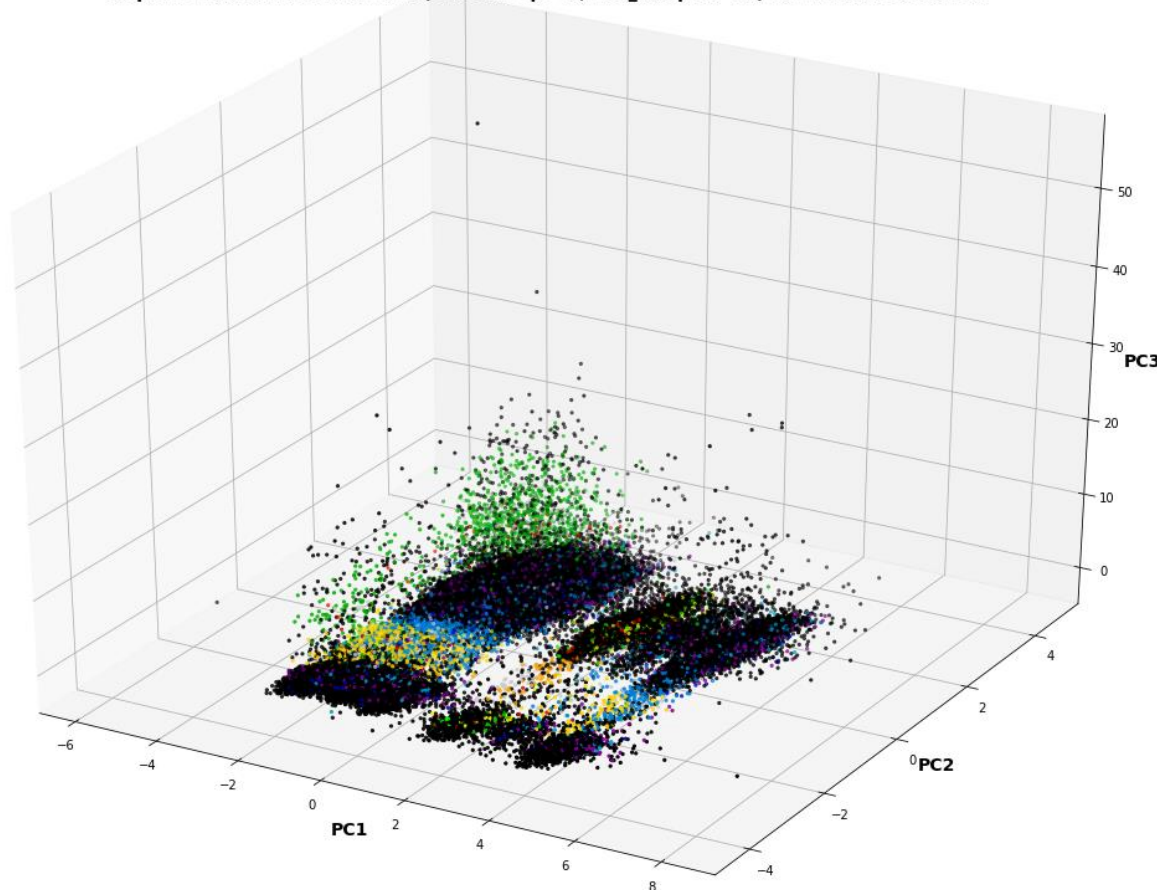
- **Clustering non supervisé**
 - ↳ Nombre de clusters ET interprétation de chaque cluster facilement exploitables
- **Analyse de la stabilité des segments au cours du temps**
 - ↳ pour évaluer la fréquence à laquelle la segmentation doit être mise à jour
- Données à disposition peu fournies = travail conséquent de **feature engineering**

2. DBSCAN

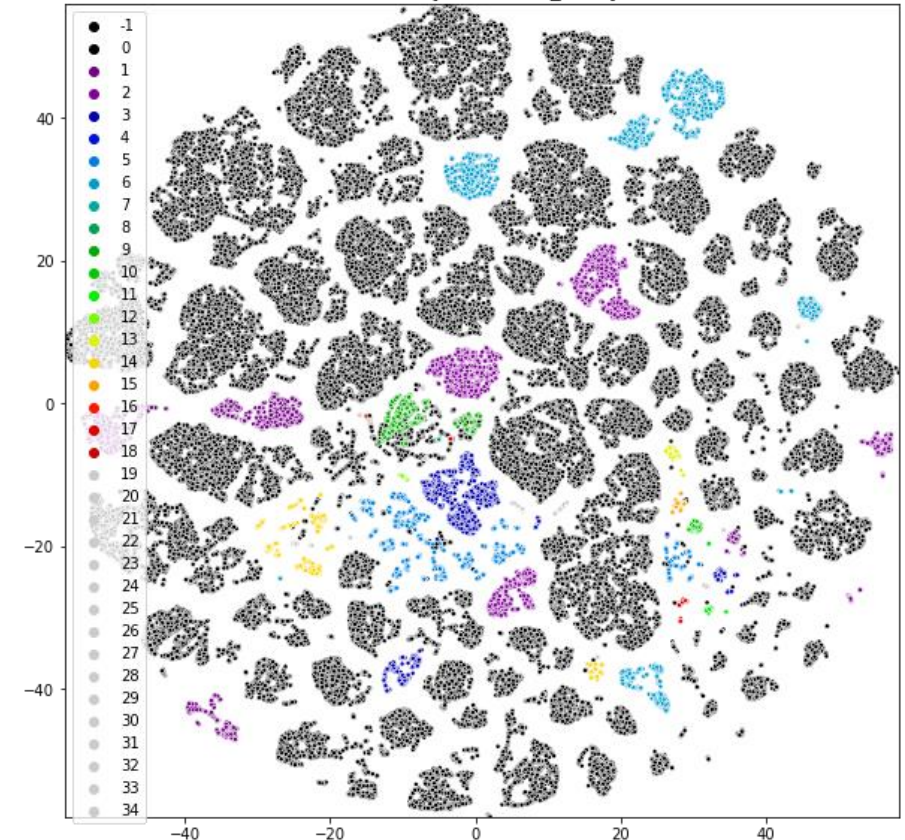
Nombre de clusters élevé... même après optimisation hyperparamètres (35 au mieux)...

NON ADAPTÉ

Représentation des 35 clusters (DBSCAN eps=5, min_samples=10) sur les axes de la PCA



Représentation des 35 clusters (DBSCAN eps=5, min_samples=10) via t-SNE à 2 composantes



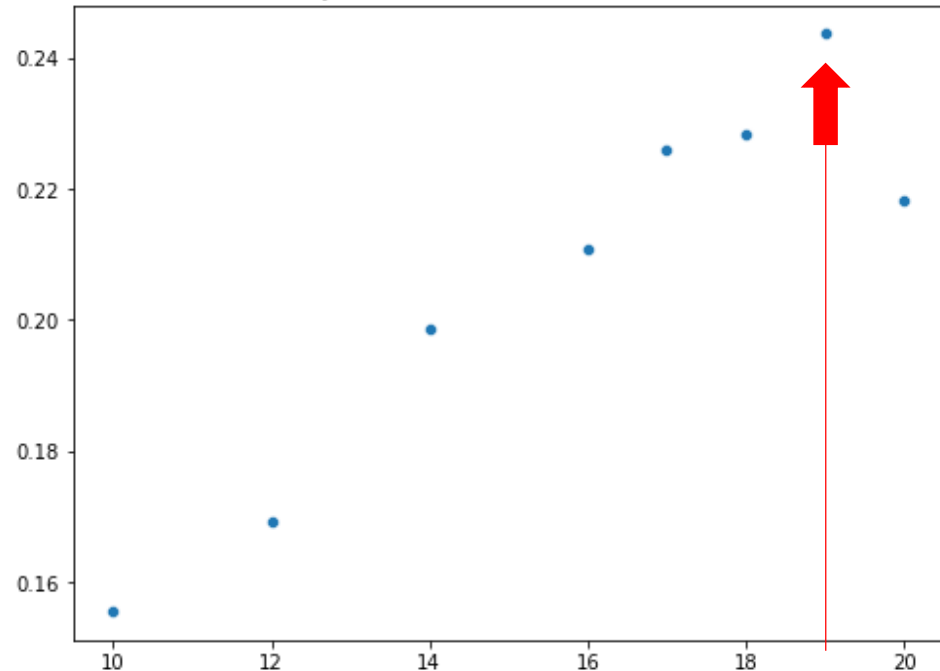
3. k-means

Optimisation nombre k de clusters

Via le coefficient de silhouette

- ↪ qui varie entre -1 (pire classification)
et +1 (meilleure classification)

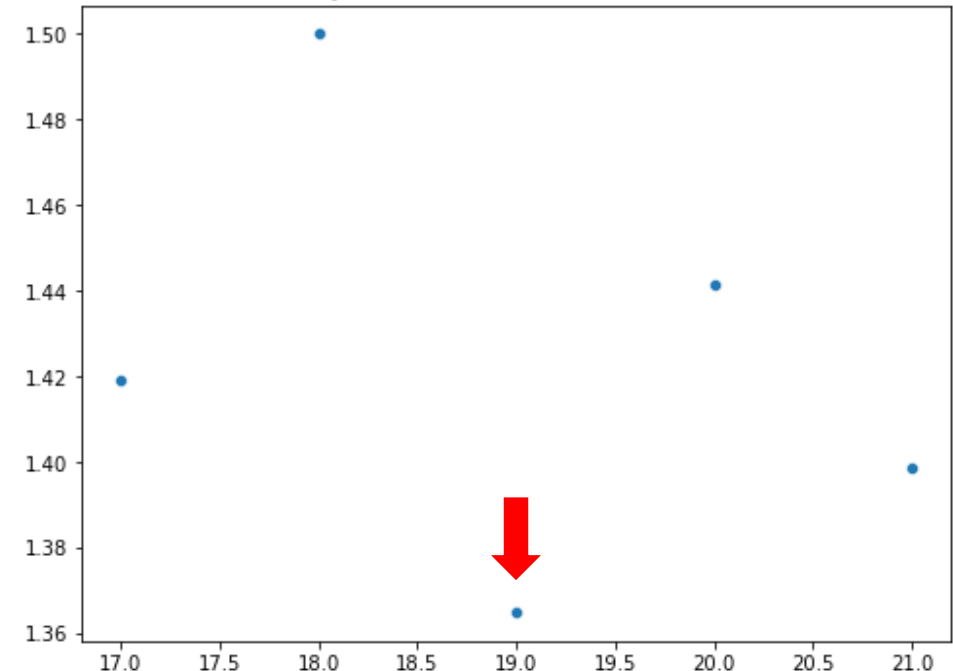
Coefficient de silhouette moyen en fonction du nombre k de clusters (méthode kmeans)



Via l'indice de Davies-Bouldin

- ↪ qui varie entre 0 (meilleure classification)
et + l'infini (pire classification)

Indice de Davies-Bouldin moyen en fonction du nombre k de clusters (méthode kmeans)



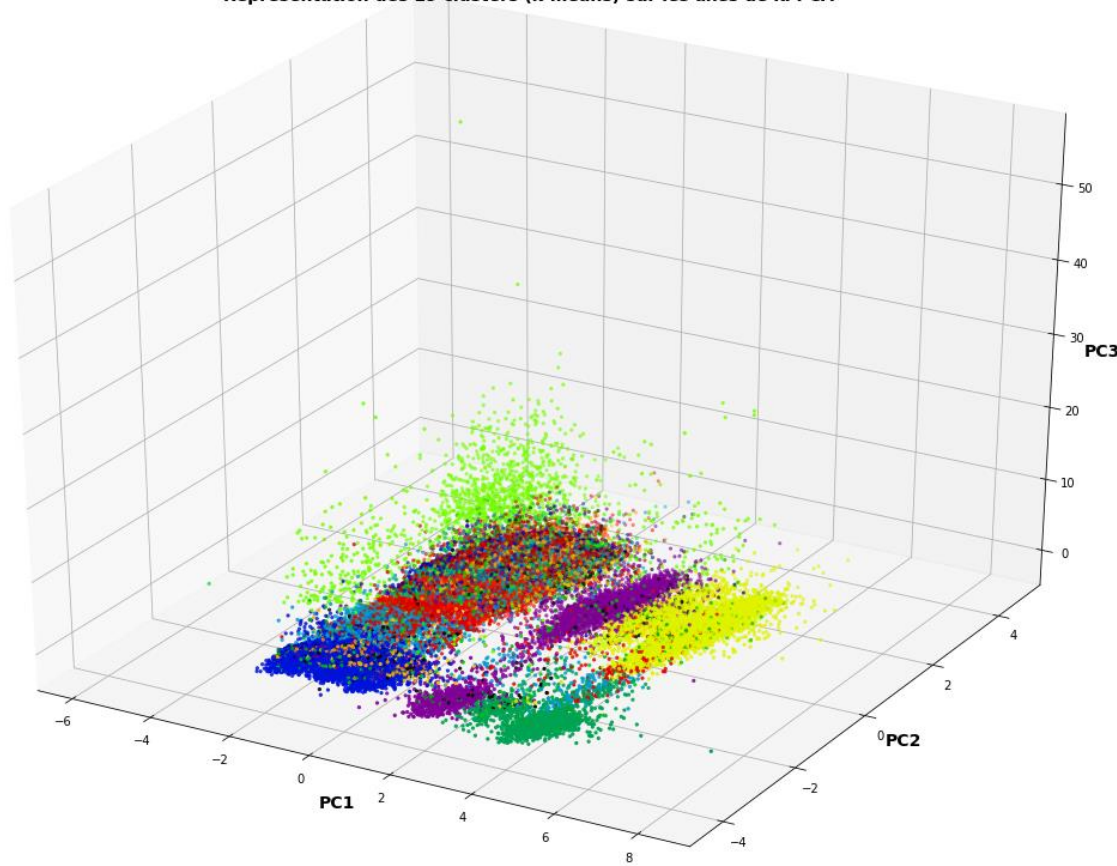
NOMBRE OPTIMAL DE CLUSTERS : k = 19

3. k-means

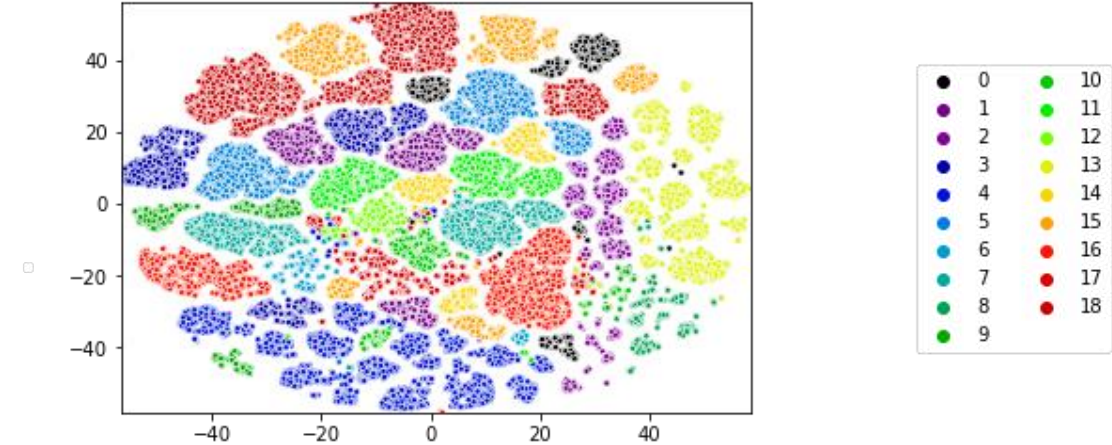
k = 19

Visualisation

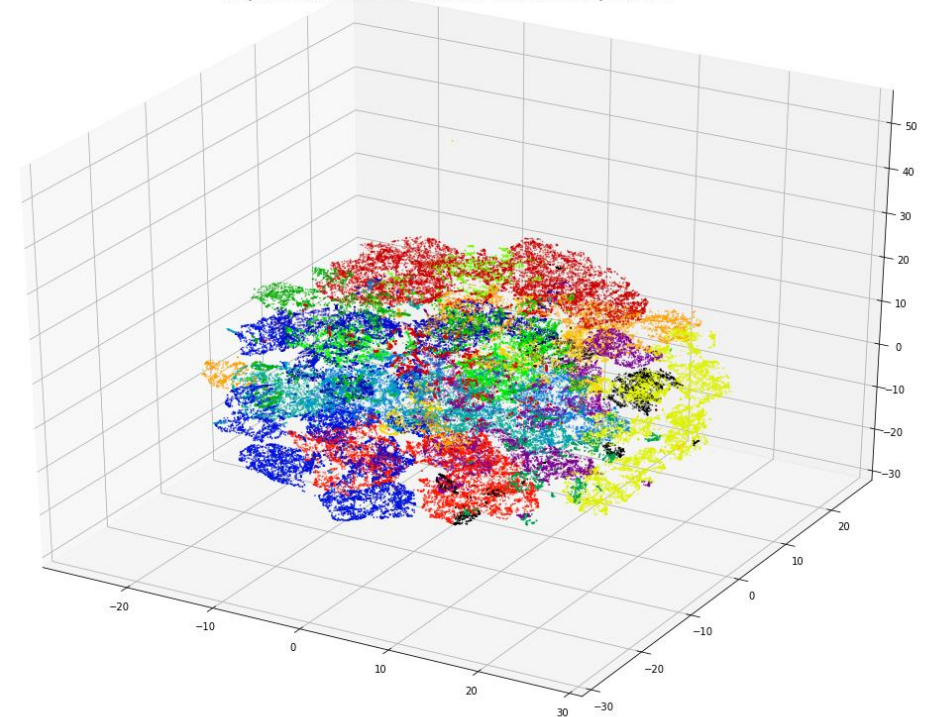
Représentation des 19 clusters (k-means) sur les axes de la PCA



Représentation des 19 clusters (k-means) via t-SNE à 2 composantes



Représentation des individus via t-SNE à 3 composantes



3. k-means

k = 19

Interprétation

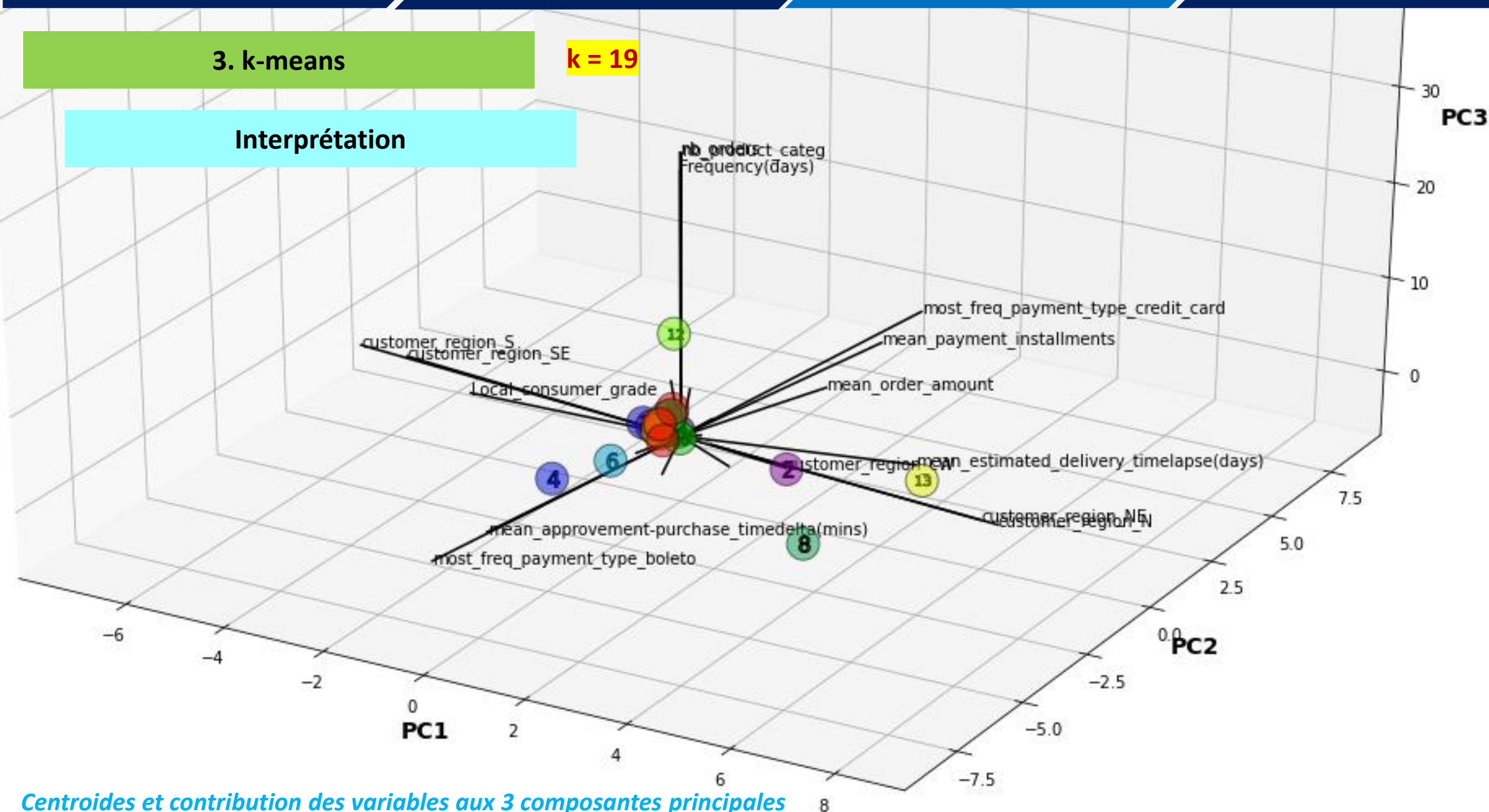
Via statistiques des clusters sur chaque variable

| | size | mean_payment_sequential | mean_payment_installments | nb_product_cat | last_review_score | Recency(days) | nb_orders | Frequency(days) |
|---------|-------|-------------------------|---------------------------|----------------|-------------------|---------------|-----------|-----------------|
| cluster | | | | | | | | |
| 0 | 3014 | 1.000664 | 3.312929 | 1.016257 | 4.180159 | 342.603209 | 1.003318 | 0.043646 |
| 1 | 5158 | 1.001616 | 2.887683 | 1.009694 | 4.139007 | 305.042514 | 1.007173 | 0.077304 |
| 2 | 4808 | 1.001733 | 2.994720 | 1.003744 | 4.059484 | 284.750407 | 1.016223 | 0.149367 |
| 3 | 6058 | 1.001568 | 3.057472 | 1.008584 | 4.140971 | 273.730546 | 1.014361 | 0.057971 |
| 4 | 10066 | 1.000000 | 1.000055 | 1.003775 | 4.087622 | 288.365441 | 1.013213 | 0.116322 |
| 5 | 6992 | 1.001371 | 3.574306 | 1.002860 | 4.232122 | 275.761916 | 1.014588 | 0.173476 |
| 6 | 1493 | 1.001340 | 1.005861 | 1.014066 | 4.149364 | 217.279088 | 1.029471 | 0.707552 |
| 7 | 6371 | 1.001962 | 2.892508 | 1.003924 | 4.162612 | 287.621711 | 1.017109 | 0.272366 |
| 8 | 1994 | 1.000000 | 1.000502 | 1.005517 | 3.969910 | 311.582049 | 1.016048 | 0.236991 |
| 9 | 2482 | 1.013027 | 2.398033 | 1.010475 | 4.241338 | 258.387026 | 1.015310 | 0.211777 |
| 10 | 2417 | 1.013377 | 2.690250 | 1.038891 | 3.292925 | 351.585325 | 1.024824 | 0.047974 |
| 11 | 5450 | 1.001468 | 3.473478 | 1.011009 | 4.124037 | 266.492127 | 1.008257 | 0.099360 |
| 12 | 1628 | 1.028809 | 3.394366 | 1.942875 | 4.154791 | 233.600406 | 2.189803 | 66.819927 |
| 13 | 7633 | 1.004389 | 4.211763 | 1.003013 | 3.866370 | 298.780291 | 1.014149 | 0.123047 |
| 14 | 3115 | 1.001124 | 2.971161 | 1.009310 | 4.047512 | 282.005188 | 1.006421 | 0.036659 |
| 15 | 6365 | 1.002304 | 3.373648 | 1.003456 | 4.144698 | 281.480476 | 1.013511 | 0.158441 |
| 16 | 7500 | 1.001667 | 2.483156 | 1.000267 | 4.059733 | 283.571649 | 1.013200 | 0.129058 |
| 17 | 3155 | 1.598961 | 1.342673 | 1.009509 | 4.051664 | 307.711640 | 1.021870 | 0.313917 |
| 18 | 10397 | 1.001720 | 3.989178 | 1.002116 | 4.004713 | 300.313487 | 1.024142 | 0.183611 |

3. k-means

 $k = 19$

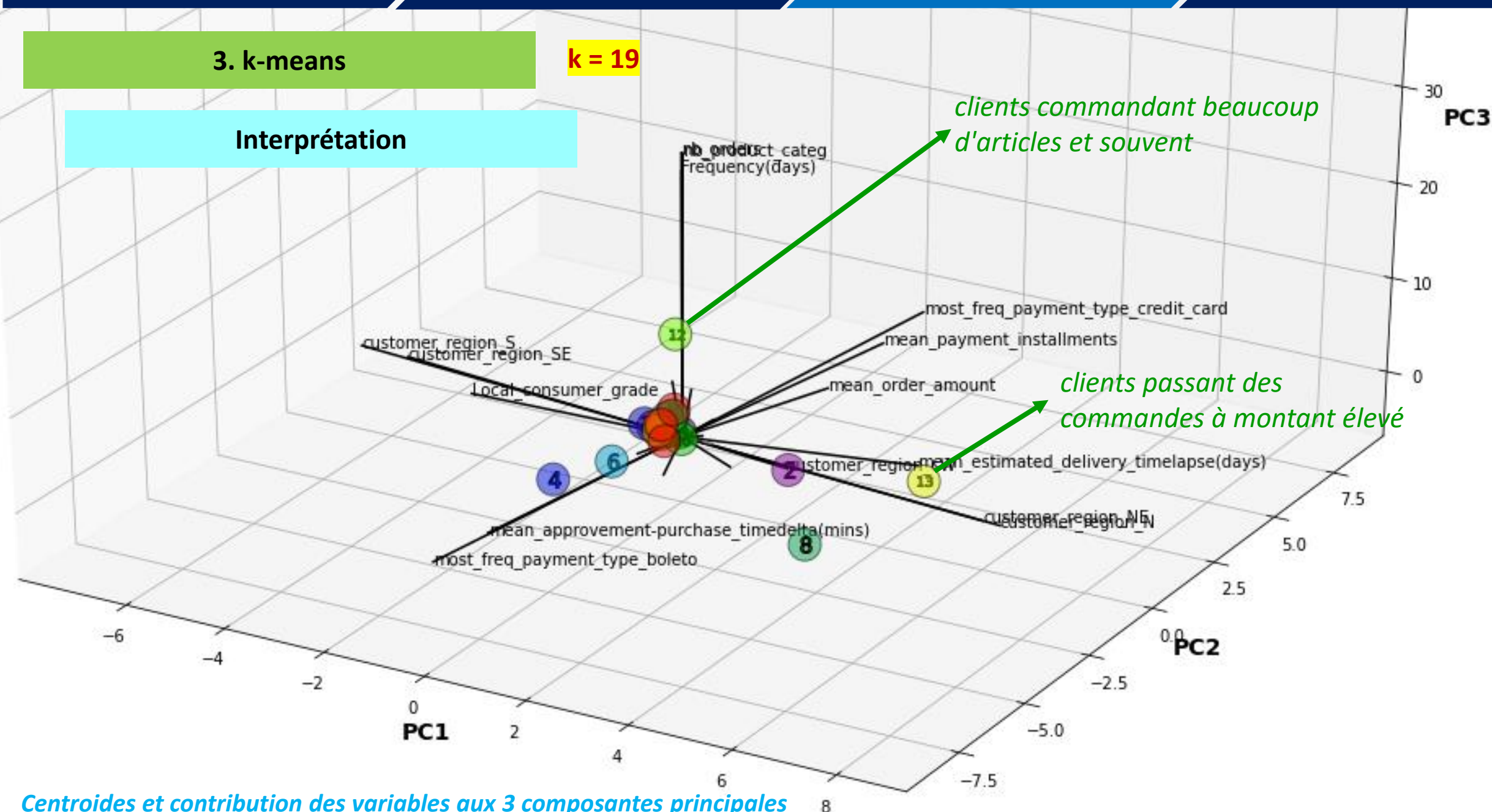
Interprétation



3. k-means

 $k = 19$

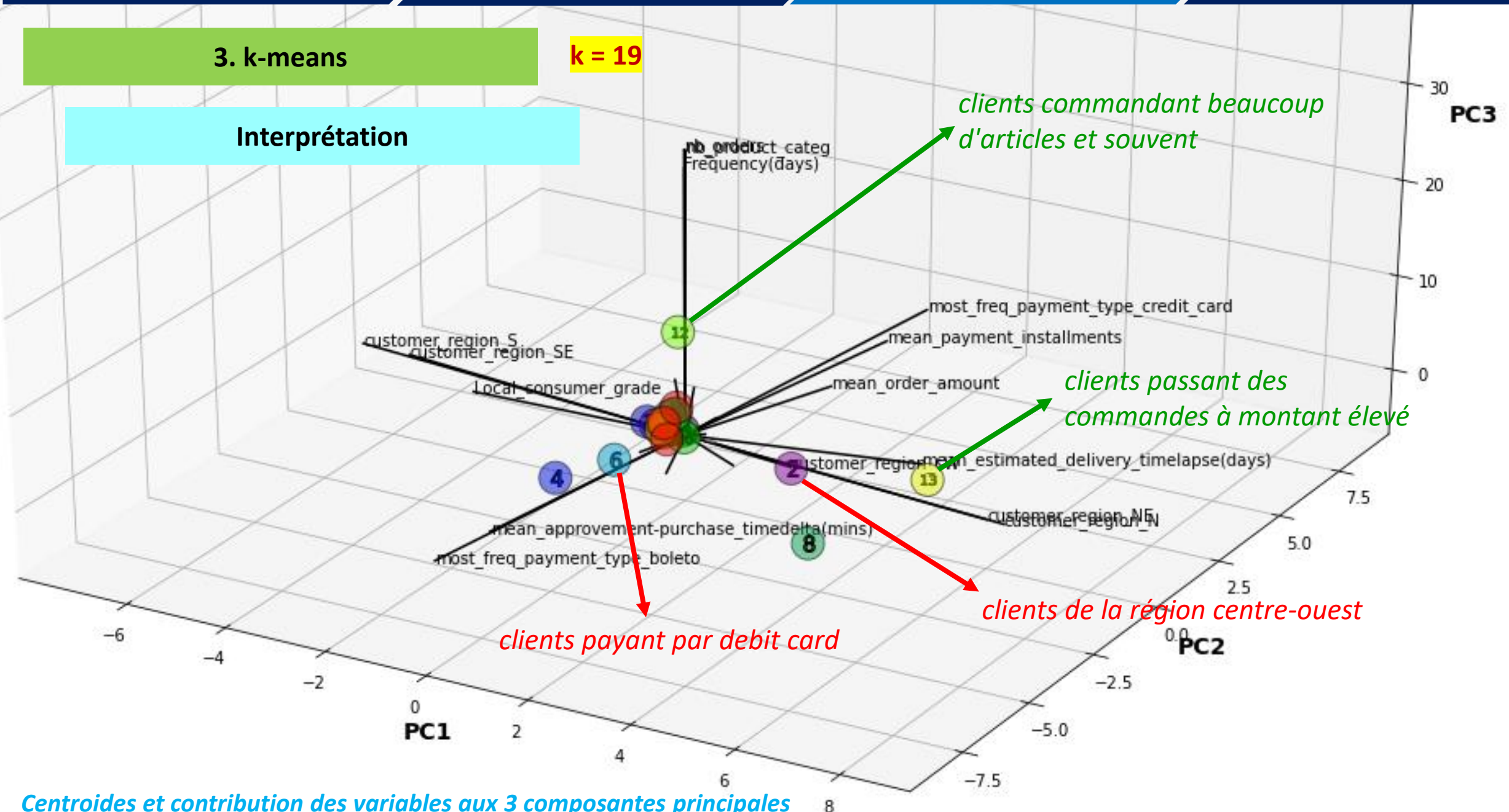
Interprétation



3. k-means

 $k = 19$

Interprétation



3. k-means

 $k = 19$

Interprétation

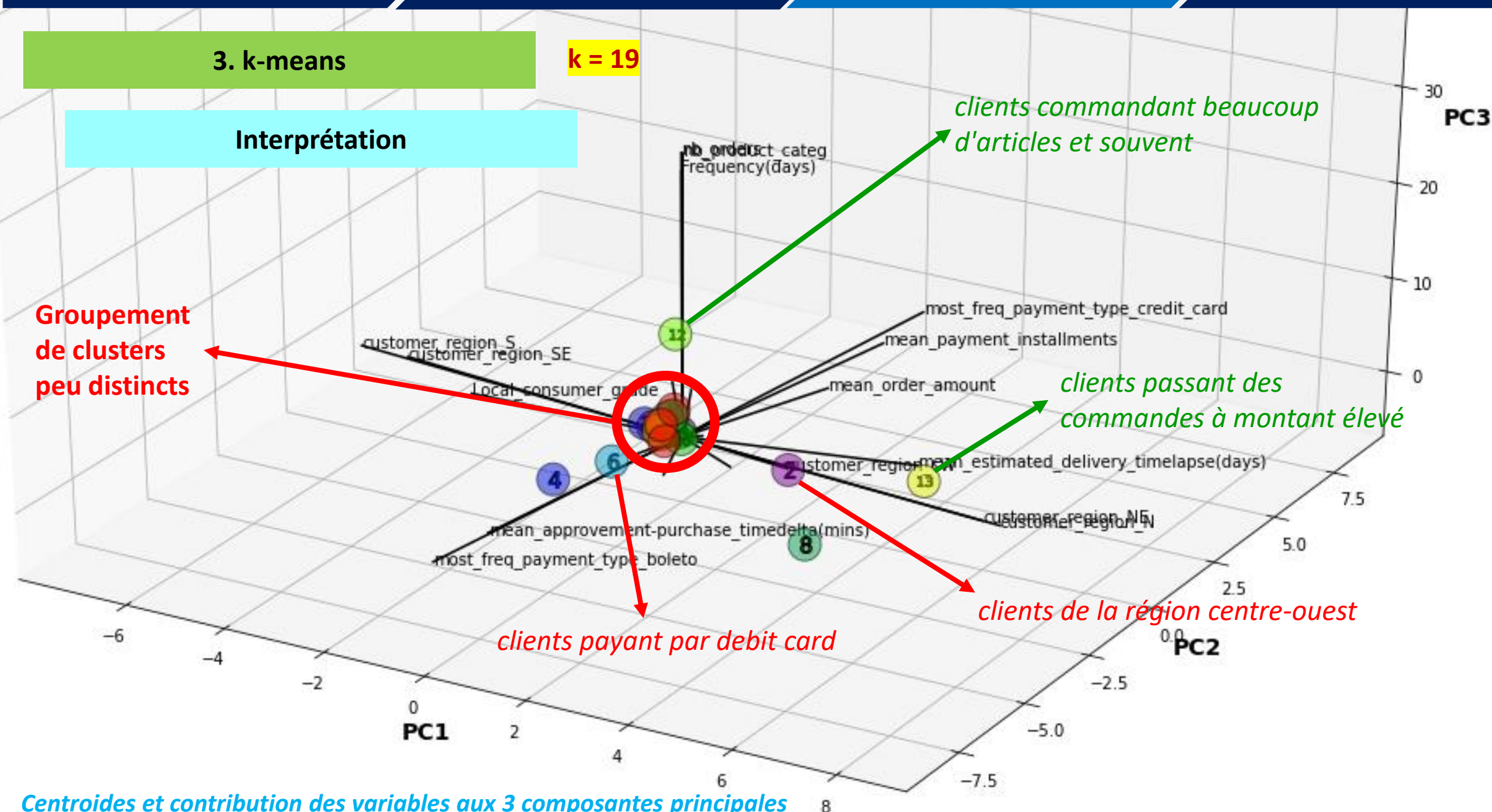
Groupement
de clusters
peu distincts

clients commandant beaucoup
d'articles et souvent

clients passant des
commandes à montant élevé

clients payant par debit card

clients de la région centre-ouest



3. k-means

k = 19

Nb élevé de clusters

Interprétation

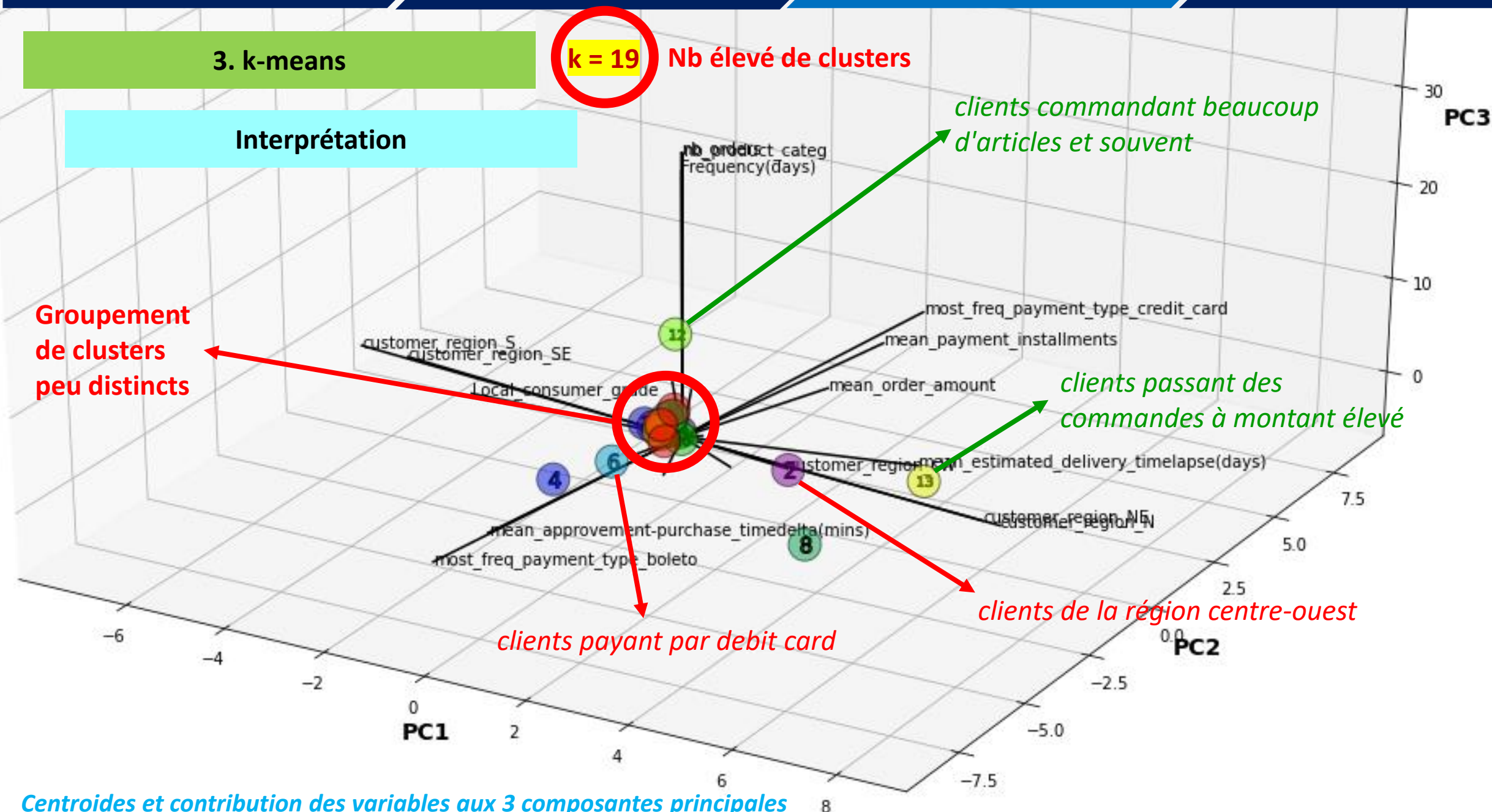
Groupement
de clusters
peu distincts

clients commandant beaucoup
d'articles et souvent

clients passant des
commandes à montant élevé

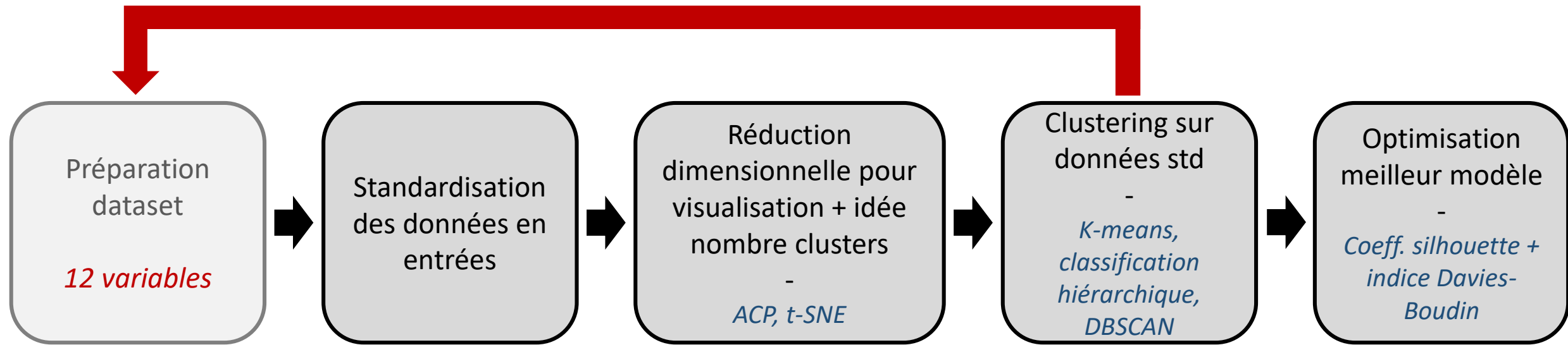
clients payant par debit card

clients de la région centre-ouest



Clustering : méthodologie

Suppression des colonnes encodées: région, catégorie la plus fréquente, mode de paiement le plus fréquent

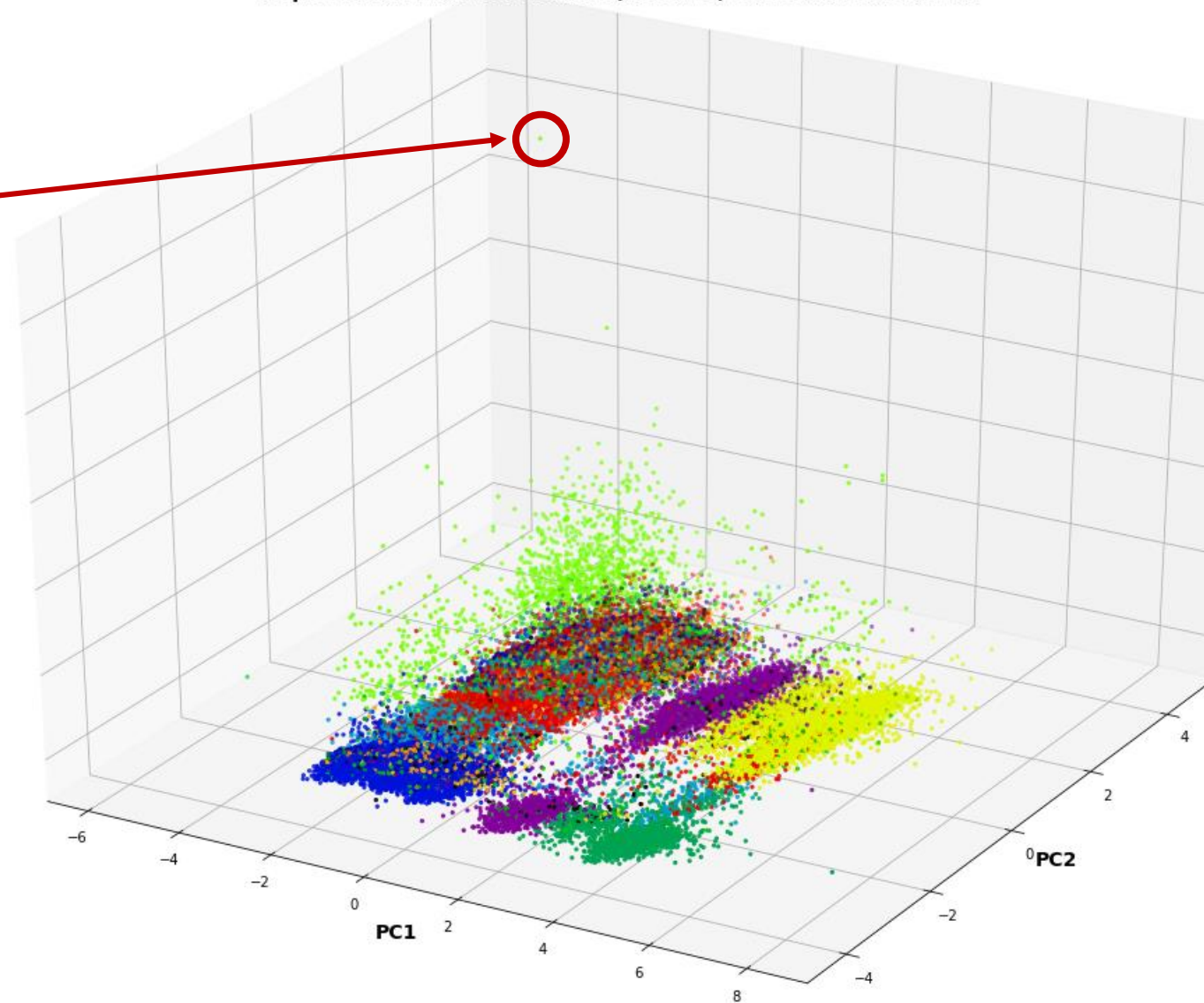


CI

Représentation des 19 clusters (k-means) sur les axes de la PCA

Suppression des colonnes encodées: région, catég

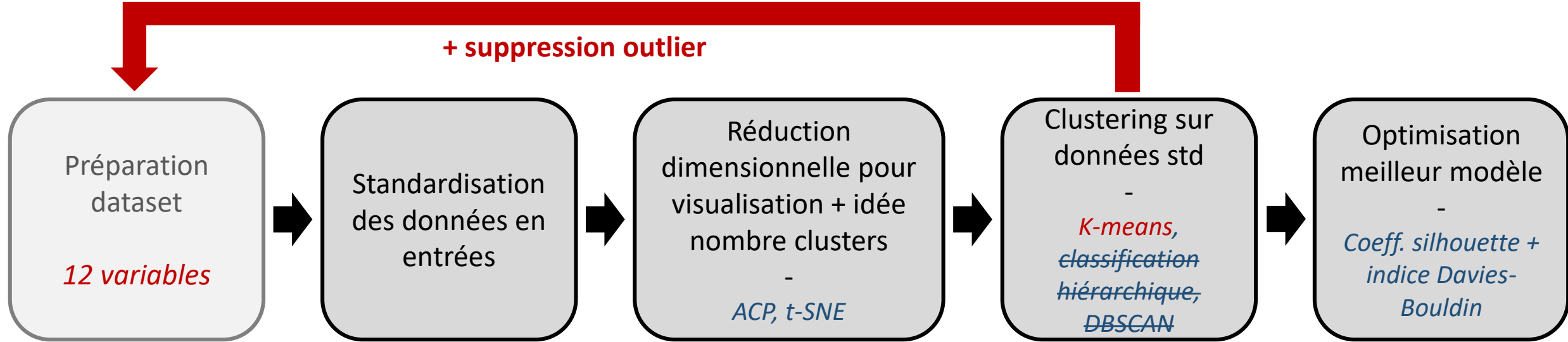
+ suppression outlier

Préparation
dataset*12 variables*Standardisation
des données en
entrées

Clustering : méthodologie

Suppression des colonnes encodées: région, catégorie la plus fréquente, mode de paiement le plus fréquent

+ suppression outlier

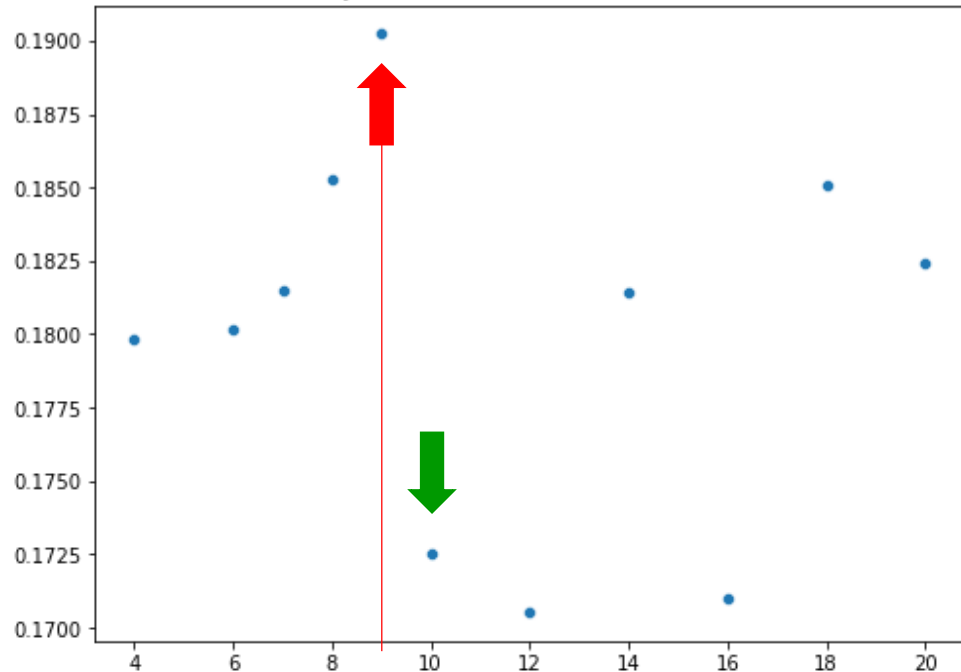


Optimisation nombre k de clusters

Via le coefficient de silhouette

- ↪ qui varie entre -1 (pire classification)
et +1 (meilleure classification)

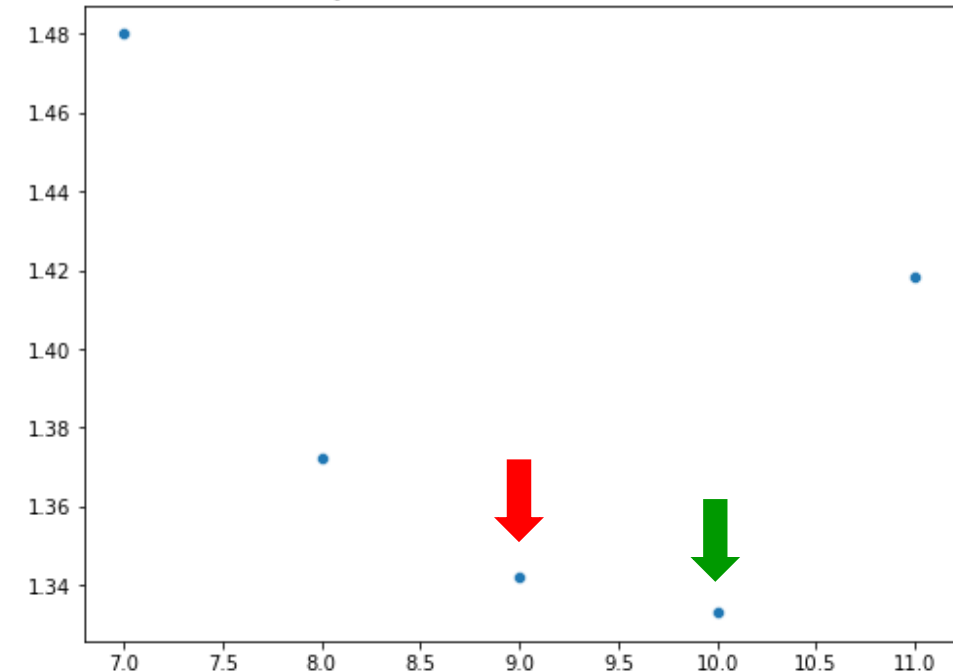
Coefficient de silhouette moyen en fonction du nombre k de clusters (méthode kmeans)



Via l'indice de Davies-Bouldin

- ↪ qui varie entre 0 (meilleure classification)
et + l'infini (pire classification)

Indice de Davies-Bouldin moyen en fonction du nombre k de clusters (méthode kmeans)



NOMBRE OPTIMAL DE CLUSTERS : k = 9

Visualisation/interprétation

K-means optimisé, $k = 9$

1- clients satisfaits, achetant compulsivement et localement (pressés de recevoir leurs produits?)

0- clients satisfaits, mais dépensant peu, achetant localement et/ou pressés de recevoir leurs produits

7- clients satisfaits, achetant localement mais non compulsivement / prenant le temps de réfléchir avant de valider leur commande

4- clients récents attirés par vouchers

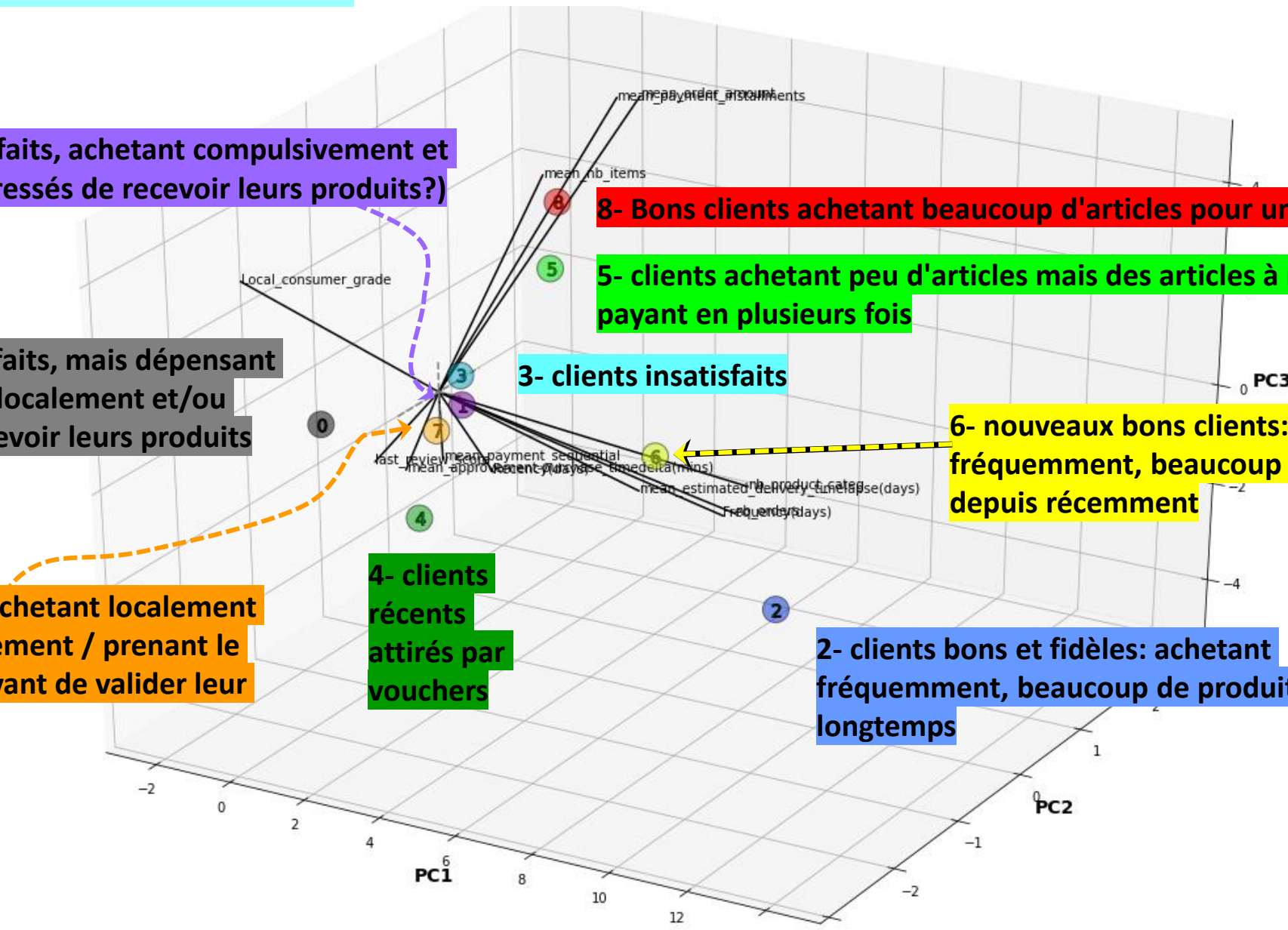
3- clients insatisfaits

2- clients bons et fidèles: achetant fréquemment, beaucoup de produits, depuis longtemps

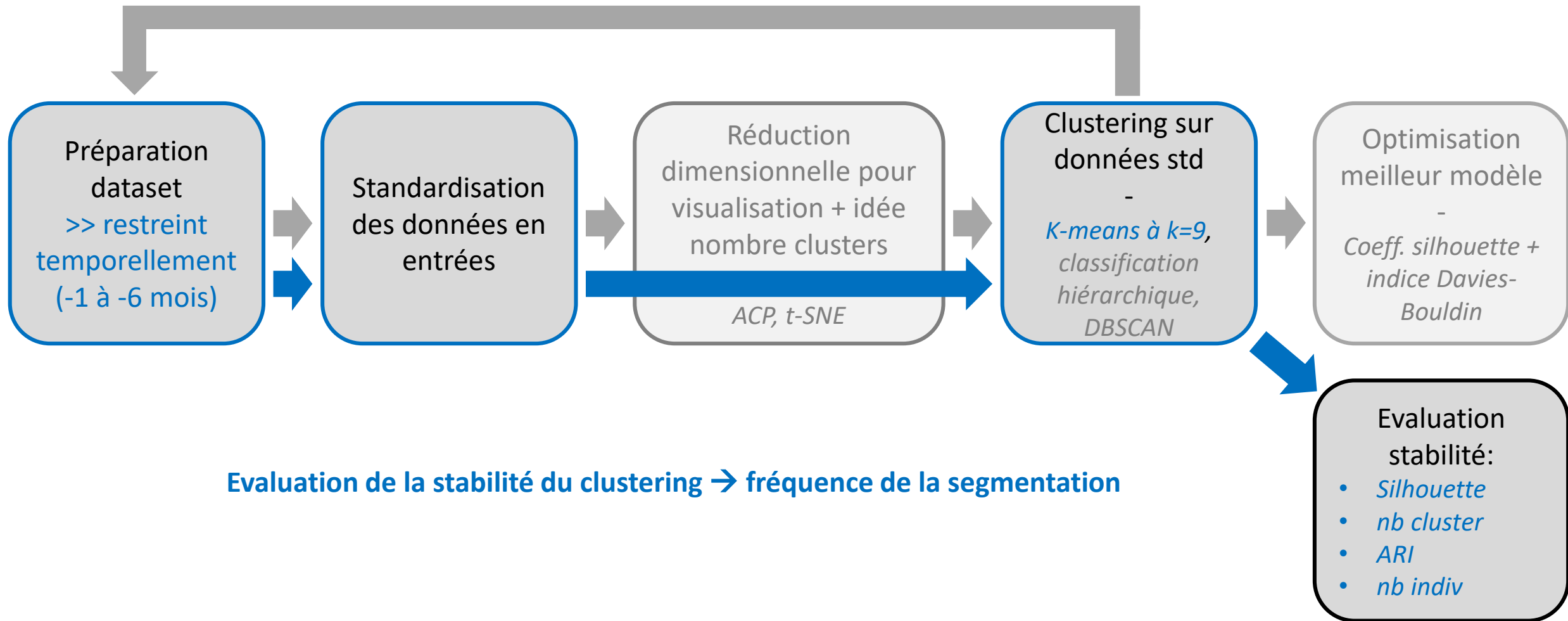
8- Bons clients achetant beaucoup d'articles pour un montant total élevé

5- clients achetant peu d'articles mais des articles à montants élevés et payant en plusieurs fois

6- nouveaux bons clients: achetant fréquemment, beaucoup de produits, mais depuis récemment

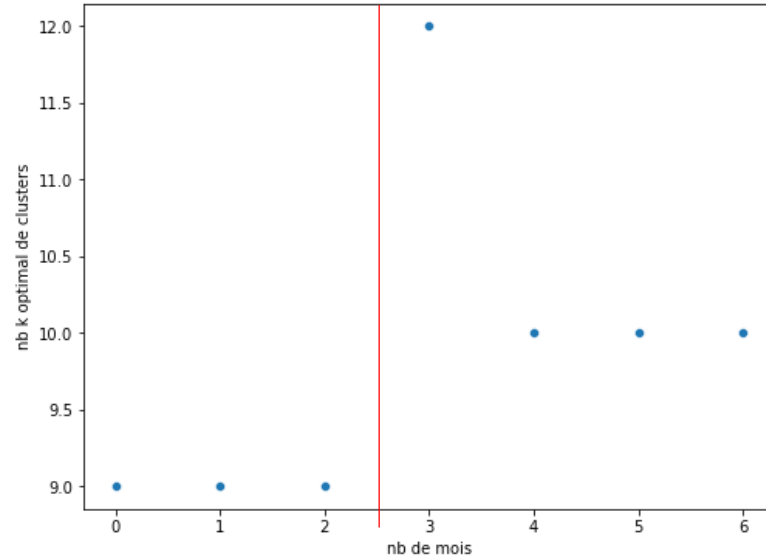


Clustering : méthodologie



Evaluation de la stabilité du clustering → fréquence de la segmentation

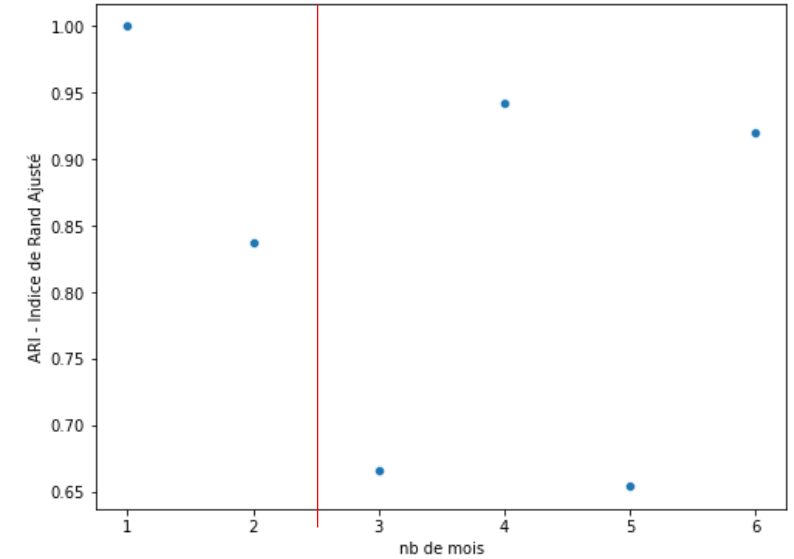
Stabilité du clustering : nb k optimal de clusters en fonction du nombre de mois les plus récents enlevés au dataset avant de réaliser le clustering kmeans à k clusters



❖ **nb optimal clusters:** stable sur 2 mois, change à partir de 3 mois

❖ **ARI:** ca. 84% des paires de points correctement groupées à 2 mois, mais ca. 67% à 3 mois (+ nouveaux clients = non classés)

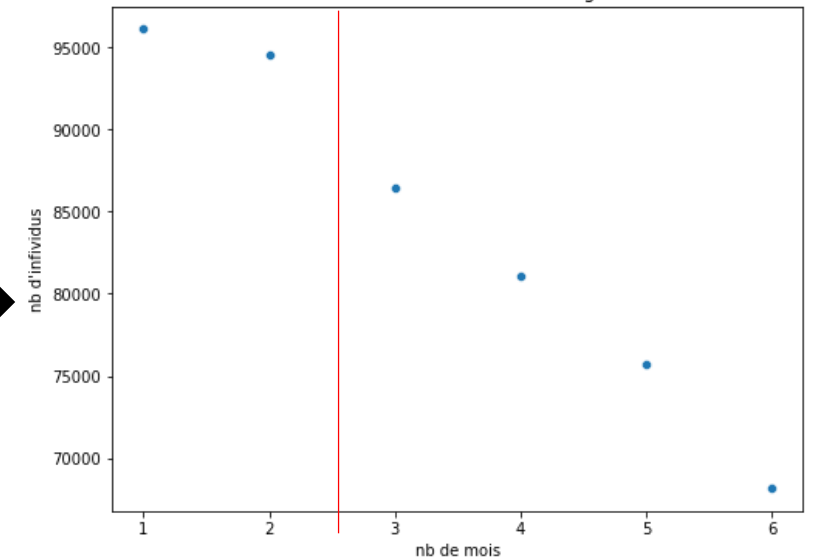
Stabilité du clustering : ARI en fonction du nombre de mois les plus récents enlevés au dataset avant de réaliser un clustering kmeans à 9 clusters



❖ **coeff de silhouette:** décroissance + marquée à 2 et 3 mois, mais non exceptionnelle (ca. 0.004)

❖ **nb clients:** 98% des clients segmentés à 2 mois, 90% à 3 mois (mais dépend dynamique création de nouveaux clients)

Stabilité du clustering : nb d'individus en fonction du nombre de mois les plus récents enlevés au dataset avant de réaliser un clustering kmeans à 9 clusters



→ Fréquence conseillée = 3 mois

Conclusion :

- Avantages:

MISSIONS

- ✓ fournir à l'équipe marketing une **description actionnable de la segmentation**
- ✓ fournir une **proposition de contrat de maintenance**

- Limites:

- Fréquence peu variée (= difficulté d'appliquer une segmentation RFM classique)

Perspectives :

- Affinage du ciblage:
 - Filtre par catégorie de produit la plus fréquente
 - Filtre par région pour clients consommateurs locaux
 - Filtre par type de paiement le plus fréquent (vouchers)

INTERPRÉTATION :

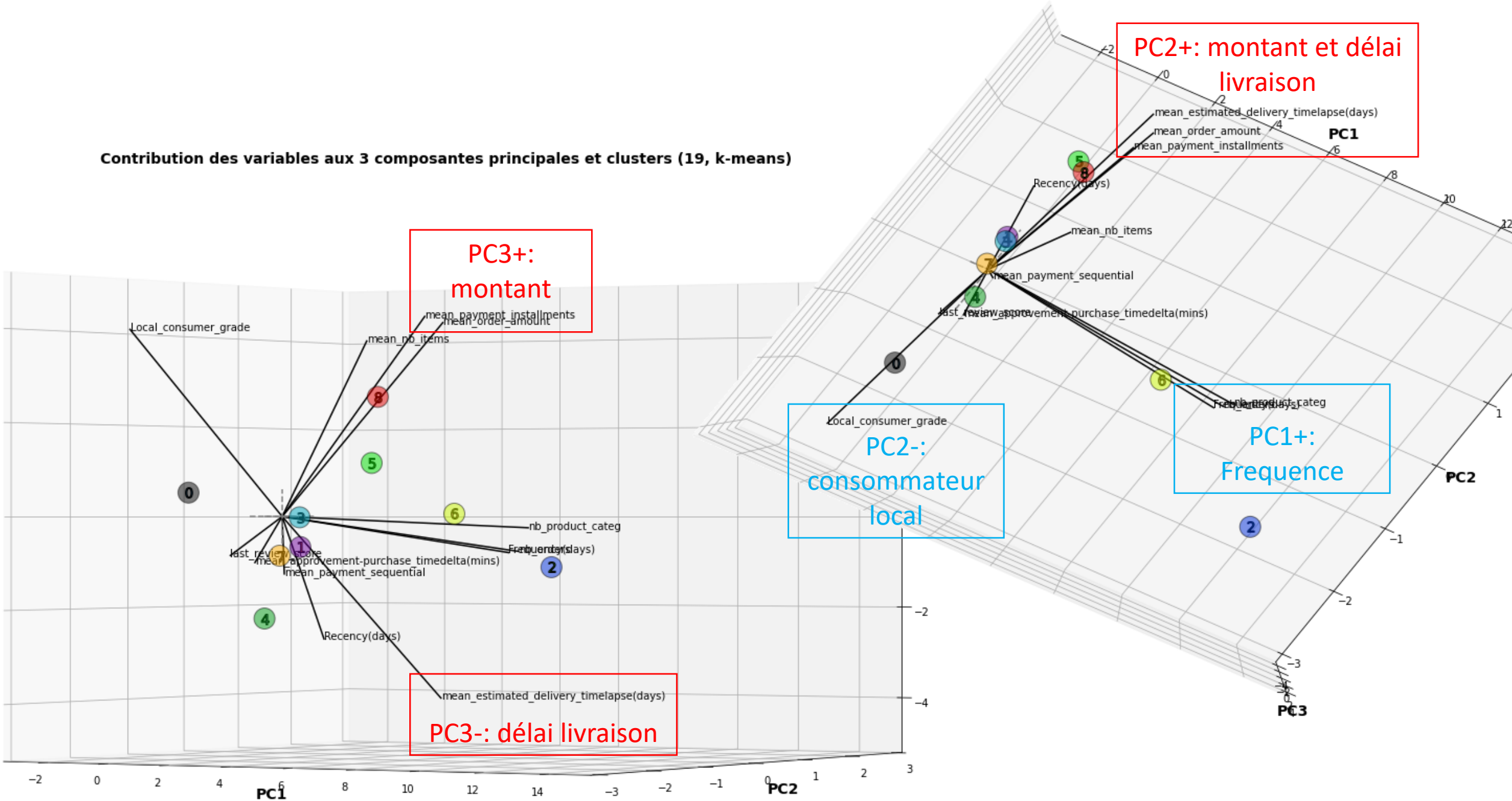
➤ **Clustering non supervisé**

- ↳ ✓ Nombre de clusters (=9) ET
- ✓ interprétation de chaque cluster facilement exploitables

➤ **Analyse de la stabilité des segments au cours du temps**

- ↳ pour ✓ évaluer la fréquence à laquelle la segmentation doit être mise à jour (= 3 mois)

Contribution des variables aux 3 composantes principales et clusters (19, k-means)



Visualisation/interprétation

K-means optimisé, $k = 9$

1- clients satisfaits, achetant compulsivement et localement car pressés de recevoir leurs produits

produits : last_review_score max, mean_approvement_purchase_timedelta(mins) min, mean_estimated_delivery_timelapse(days) max, Local_consumer_grade min

0- clients satisfaits, mais dépensant peu, achetant localement et/ou pressés de recevoir leurs produits

produits : last_review_score close to max, mean_order_amount min, mean_estimated_delivery_timelapse(days) min, Local_consumer_grade max

7- clients satisfaits, achetant localement car pressés de recevoir leurs produits mais non compulsivement / prenant le temps de réfléchir avant de valider leur commande

produits : last_review_score close to max, mean_approvement_purchase_timedelta(mins) max, mean_estimated_delivery_timelapse(days) close to max, Local_consumer_grade close to min

4- clients récents attirés par vouchers

produits : mean_payment_sequential max, Recency(days) max

3- clients insatisfaits

produits : last_review_score min

5- clients achetant peu d'articles mais des articles à montants élevés et payant en plusieurs fois

produits : mean_payment_installments max, mean_order_amount max

8- clients achetant beaucoup d'articles pour un montant total élevé

produits : mean_order_amount close to max, mean_nb_items max

6- nouveaux bons clients: achetant fréquemment, beaucoup de produits, mais depuis récemment

produits : nb_orders high, nb_product_categ max, Frequency(days) high, mean_nb_items relatively high

2- clients bons et fidèles: achetant fréquemment, beaucoup de produits, depuis longtemps

produits : nb_product_categ close to max, Recency(days) min, nb_orders max, Frequency(days) max

