# Projet 4 : Soutenance

Segmentez des clients d’un site e-commerce

Gaëtan PELLETIER

# Sommaire

- Problématique, interprétation et pistes de recherche envisagées
- Nettoyage des données, feature engineering et exploration/analyse
- Modélisations effectuées
- Choix du modèle final - Stabilité
- Synthèse

# Projet 4 : Soutenance

Problématiques,
interprétation
et pistes de recherche envisagées

# Problématiques

**D’après les données fournies par l’entreprise Olist, les problématiques sont :**

- Quels sont les différents types d’utilisateurs de la plateforme ?
- Quelle est la fréquence à laquelle la segmentation doit être mise à jour ?

# Interprétation

- **Quels sont les différents types d'utilisateurs ?**
  - Choix de features caractérisant le comportement d'un utilisateur
  - Cible à prédire : aucune
  - Segmentation de clients interprétable pour Olist
- **Quelle est la fréquence à laquelle la segmentation doit être mise à jour ?**
  - Analyse de la stabilité du modèle, au cours du temps.

# Pistes de recherche envisagées

- **Nettoyage des données**
- **Analyse des features :**
  - Distribution des features.
  - Indépendance des features entre elles ?
- **Transformation des données**
- **Présentation d'une segmentation RFM**
- **Mise en place d'un algorithme non supervisé (K-Means)**
- **Analyse de la stabilité du modèle**

# Nettoyage des données, feature engineering et exploration/analyse

# Nettoyage des données

- **Mémoire Ram :**
  - Les données utilisent 37,7MB de mémoire RAM (sans géolocalisation)

- **On ne garde que les commandes « delivered »**

- **On ne garde que les variables quantitatives**

- **Suppression des « NaN »**

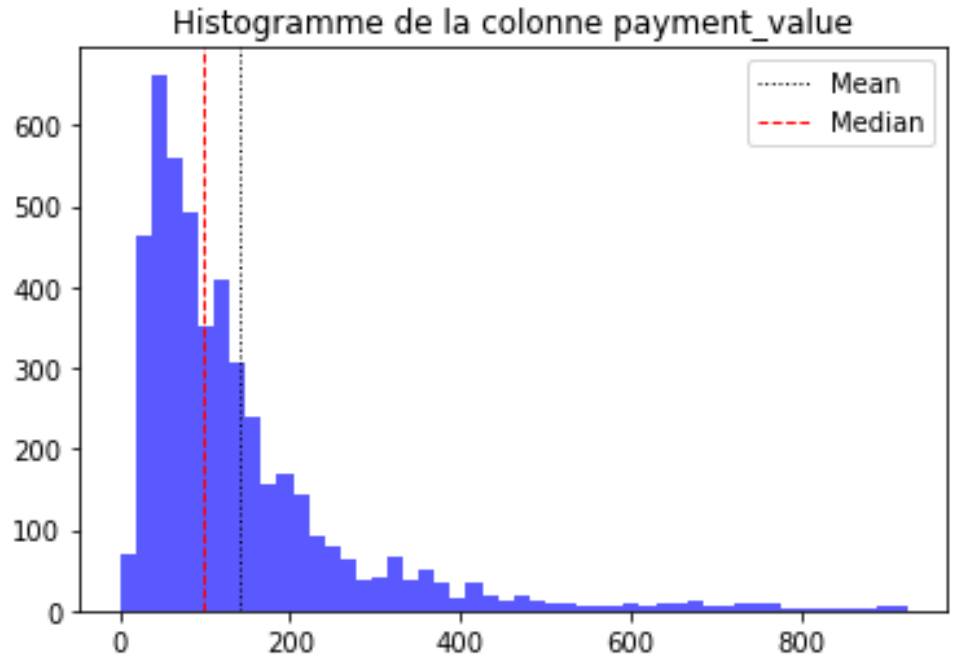
# Feature engineering

- **Création de features :**
  - Récence
  - Fréquence de visite sur la plateforme olist
  - Montant dépensé
  - Temps de livraison
- **Utilisation d'un logarithme pour obtenir une distribution normale (segmentation RFM) :**

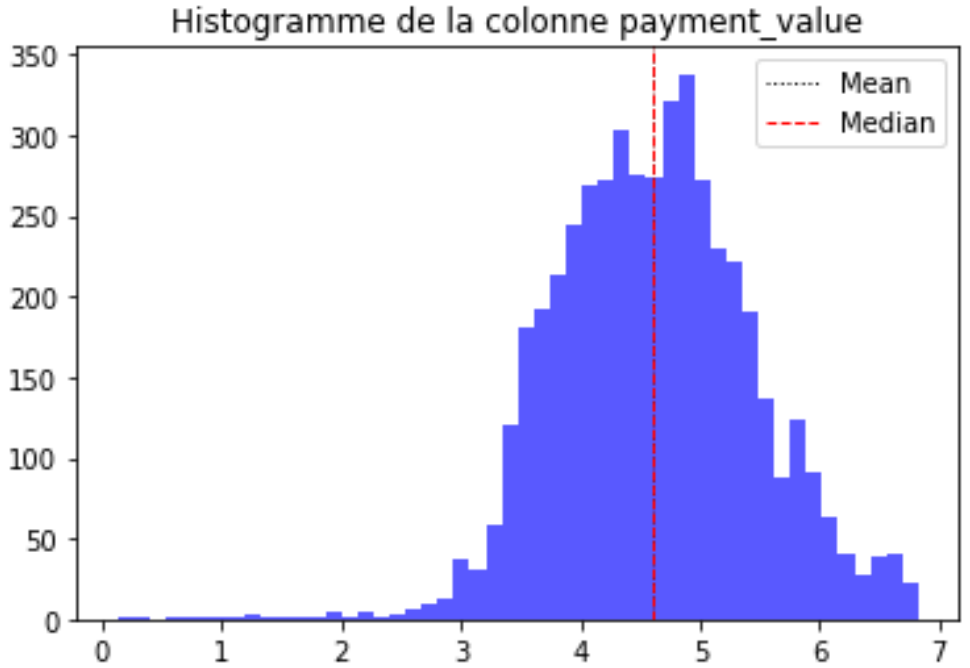
  → transformation $x = \log(x + 1)$
- **Création de scores pour la segmentation RFM**
- **Utilisation de QuantileTransformer (pour le modèle K-Means)**

# Exploration / Analyse

## Analyse univariée de payment_value :




→ log :




| | | |
|---|---|---|
| mean: | 141.024 | skewness: 2.530 |
| median: | 100.000 | kurtosis: 7.890 |
| var: | 18465.645 | |
| ect: | 135.888 | |

| | | |
|---|---|---|
| mean: | 4.613 | skewness: -0.150 |
| median: | 4.615 | kurtosis: 1.040 |
| var: | 0.695 | |
| ect: | 0.834 | |

# Annexes

## Analyse bivariée de delivery_time :

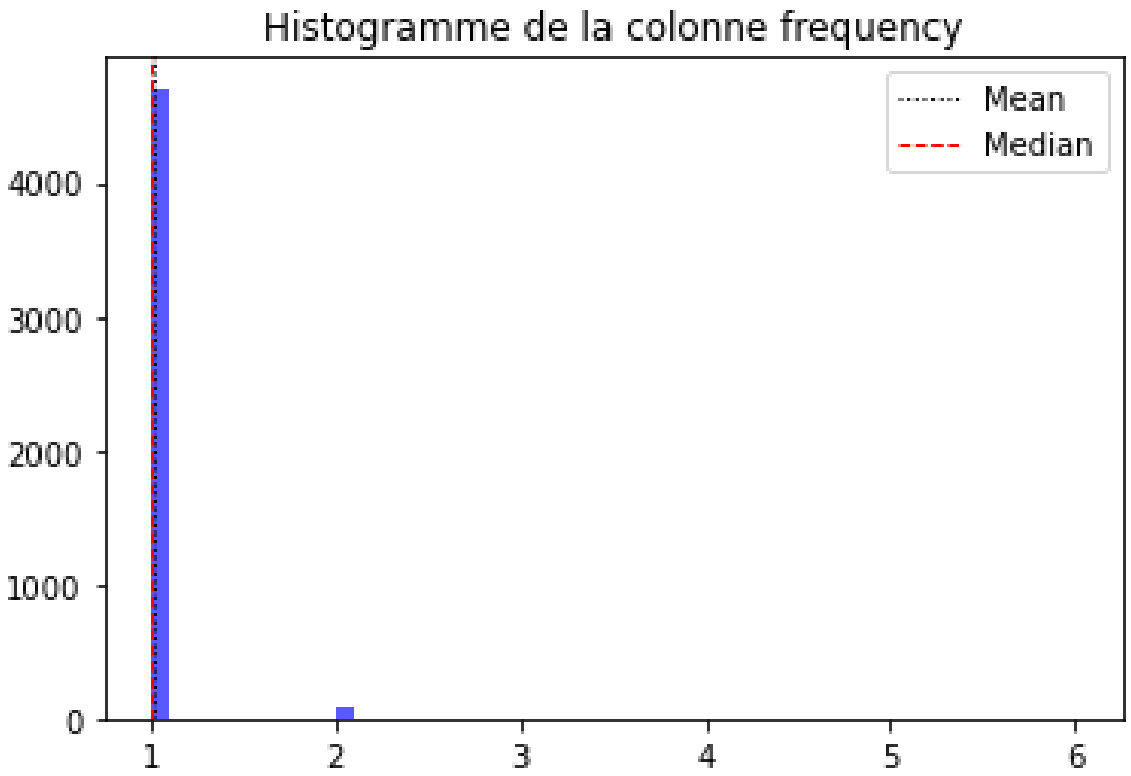
→ on vérifie que les features ne sont pas trop fortement corrélées entre elles

| | delivery_time corr with: | corr | p-value |
|---|---|---|---|
| 4 | review_score | -0.192996 | 2.111405e-41 |
| 0 | payment_value | 0.043166 | 2.812985e-03 |
| 2 | recency | -0.028765 | 4.655999e-02 |
| 1 | frequency | -0.020084 | 1.646817e-01 |

→ on effectue la même vérification pour les autre features

# Annexes

## Analyse univariée de frequency :




| | | | |
|---|---|---|---|
| mean: | 1.021 | skewness: | 12.080 |
| median: | 1.000 | kurtosis: | 219.520 |
| var: | 0.030 | | |
| ect: | 0.172 | | |

# Qualité des données

- ACP avec 1 composante

```
Explained variance: 74.6 %
```

```
(Mean(statistic=4.748823630811029e-16, minmax=(-0.08500604604740075, 0.08500604604740165)),
 Variance(statistic=12.787900842396299, minmax=(12.358003813290345, 13.217797871502253)),
 Std_dev(statistic=3.576017455549721, minmax=(3.5159091039477475, 3.636125807151694)))
```

- Interprétations :
  - Les ordres de grandeur de la moyenne, de l'écart-type et de la variance sont similaires
  - Chaque individu est proche de l'individu moyen

    → ils ont sensiblement le même comportement d'achat
  - Cela va négativement impacter les segmentations

    → les clusters risquent de ne pas être de très bonne qualité
  - Les observations se basent sur une composante d'ACP représentant 75 % de la variance du dataset

# Résumé choix des features

## Segmentation RFM :

- Récence
- Fréquence
- Montant dépensé (passage au log)

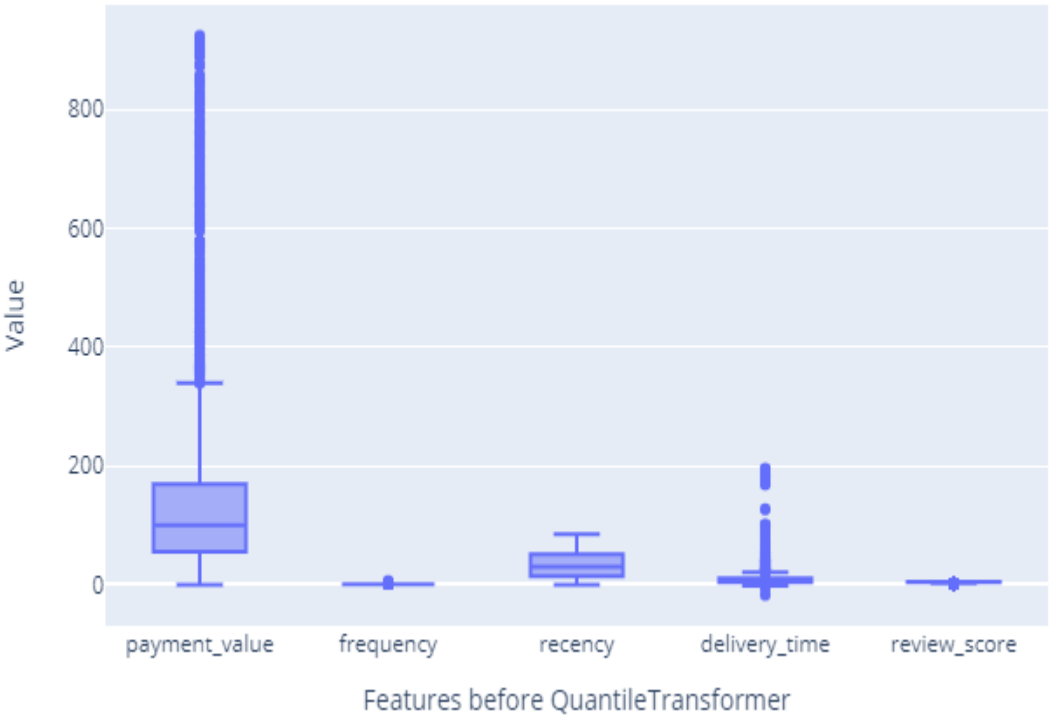
# Résumé choix des features

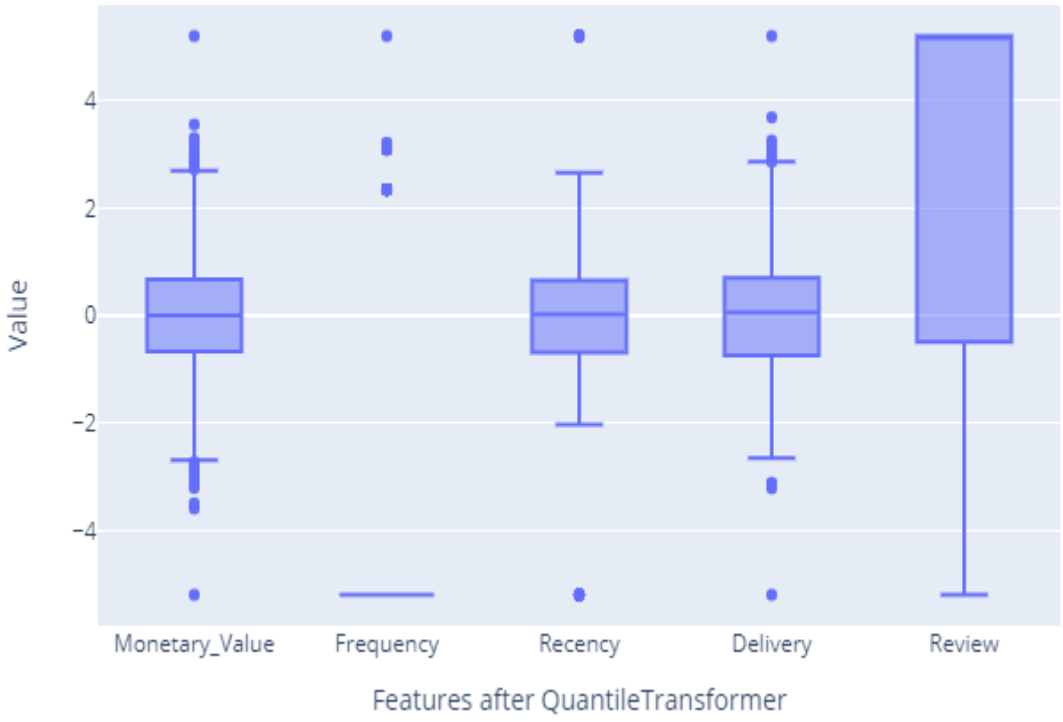
## Modèle non supervisé (K-Means) :

- Récence
- Fréquence
- Montant dépensé
- Temps de livraison
- Review score

# Impact du scaler

## Transformation avec QuantileTransformer




```
Thanks to this scaler, the range of the different features is the same.
All the features will have the same weight in the clustering model.
```

# Modélisations effectuées

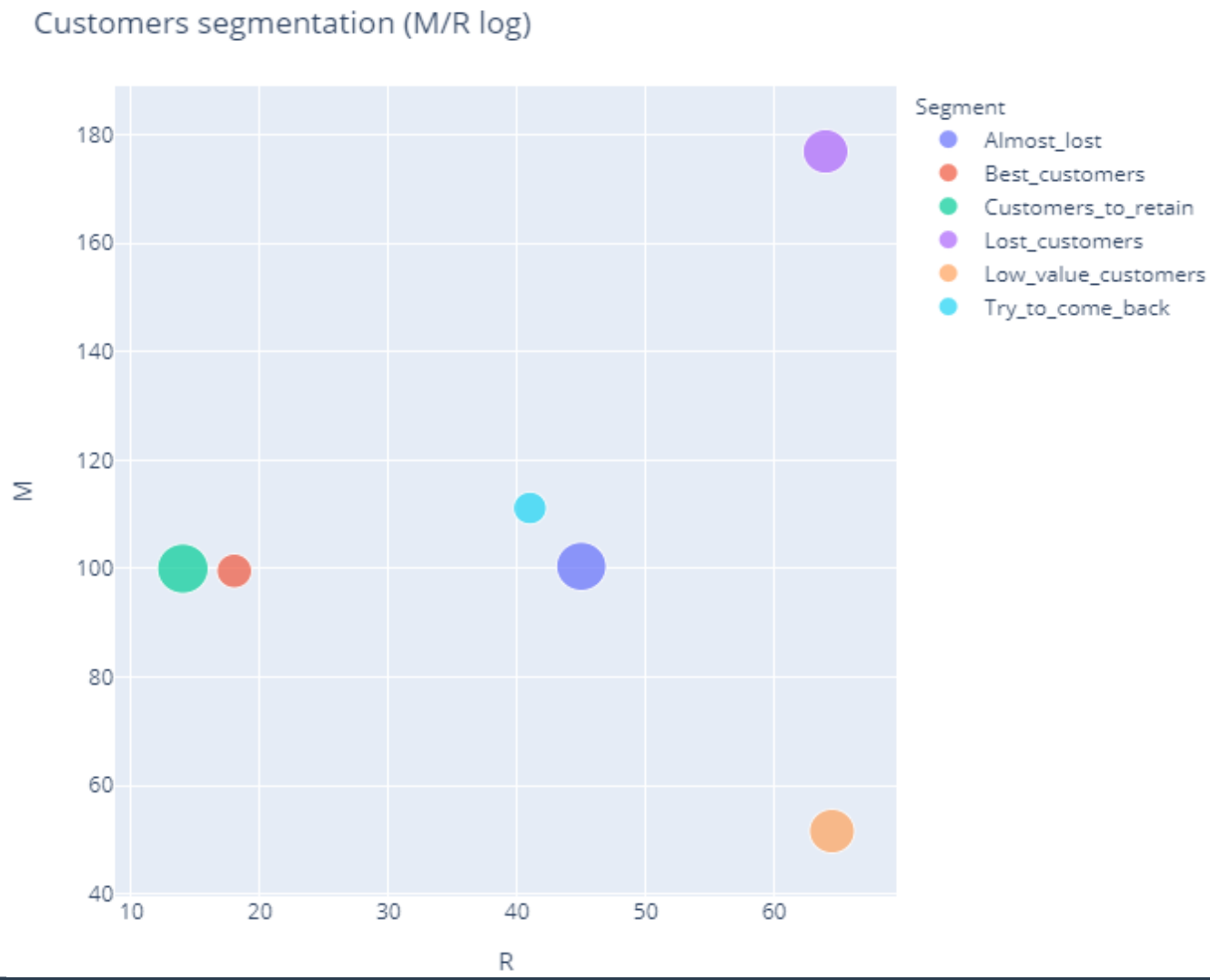
# Segmentation RFM

# Modélisations effectuées - RFM

- **Création d'un dataset de 3 mois**
- **Montant dépensé → passage au log**
- **Attribution d'un score (de 1 à 3) :**
  - R et M : binning en 3 intervalles d'amplitude égale (distribution normale)
  - F : Choix arbitraire (très faible variation des valeurs)
- **Segmentation des clients selon leurs scores**

# Modélisations effectuées - RFM

## Répartition des clients



# Modélisations effectuées - RFM

- **Limitations de la segmentation RFM :**
  - Clients évalués sur seulement 3 features
  - Les choix des scores sont arbitraires

    (e.g. qu'est-ce qu'une bonne ou mauvaise récence ?)

- **Pour éviter les choix arbitraires et enrichir la segmentation, nous allons utiliser un algorithme non supervisé : K-Means**

# Algorithme non supervisé
# K-Means

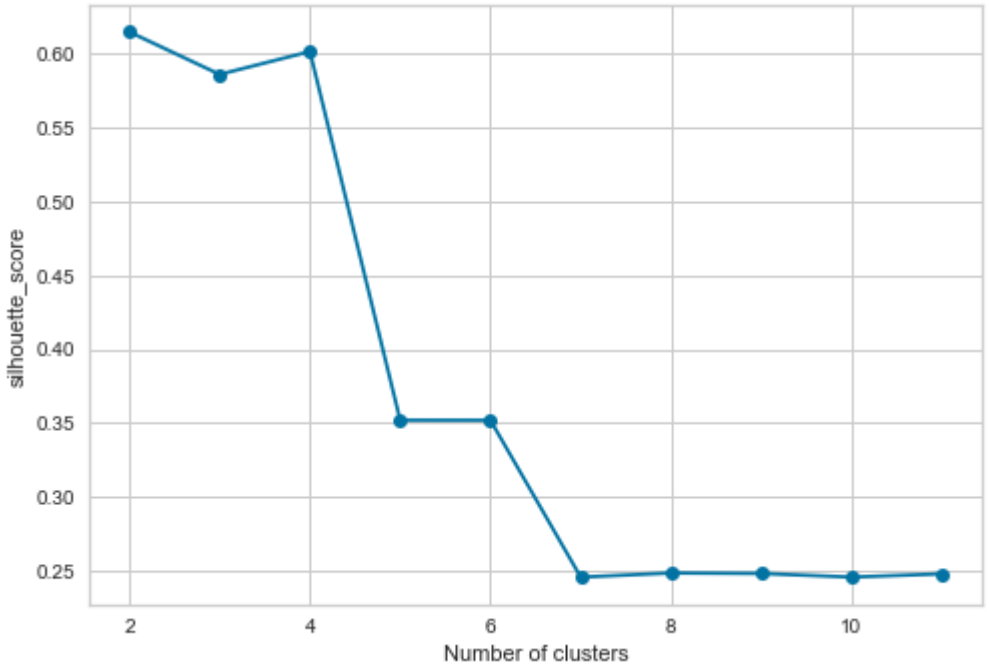
# K-Means

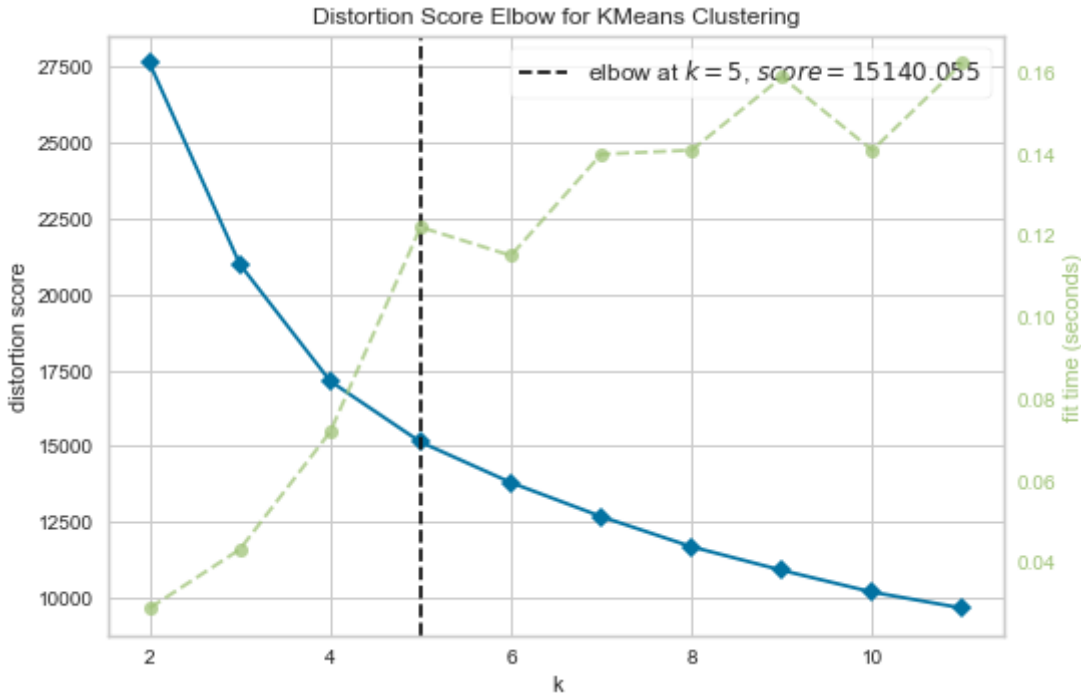
## Étapes effectuées avec le modèle K-Means :

- Détermination du nombre de clusters
- Visualisation des clusters créés
- Qualité des clusters
- Détermination des caractéristiques des clusters

# K-Means

## Détermination du nombre de clusters

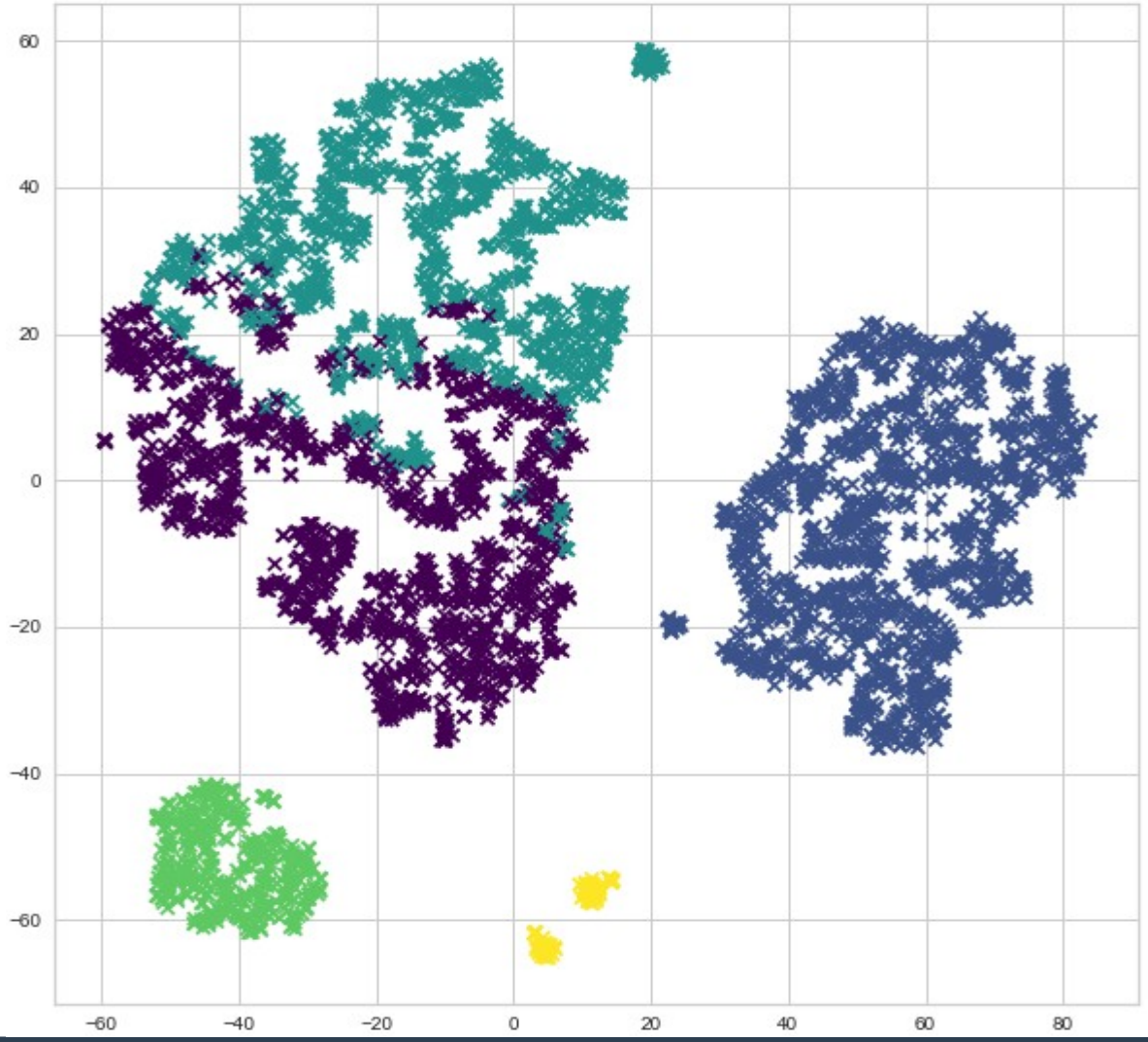



```
The clustering model will use 5 clusters
```

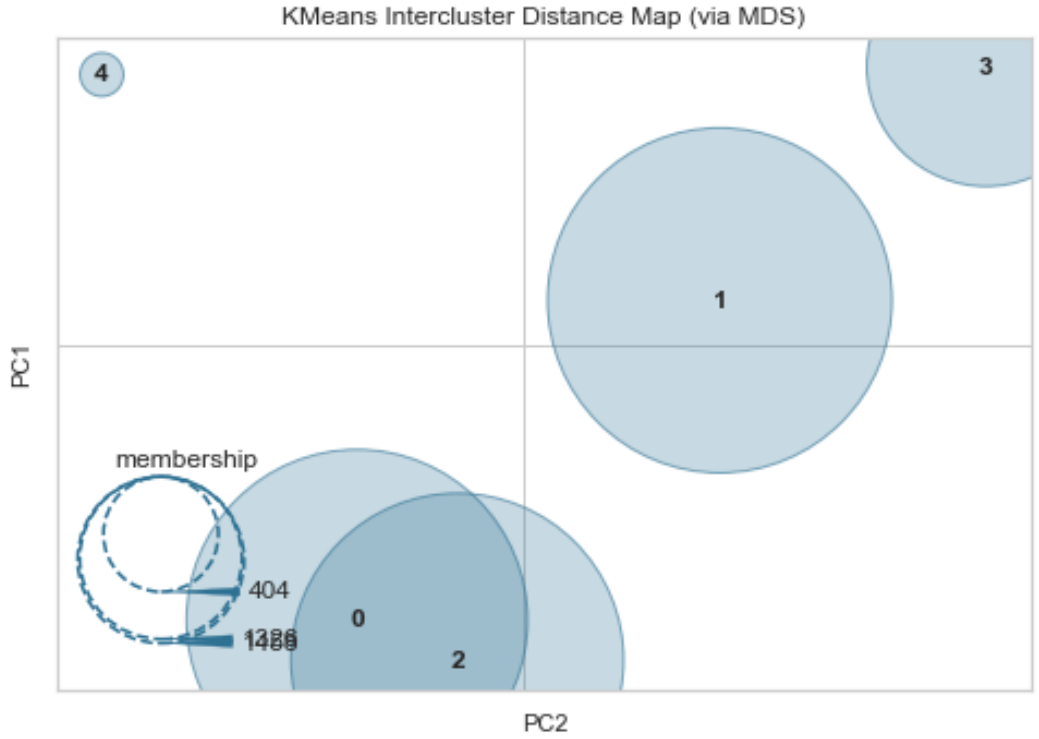
# K-Means

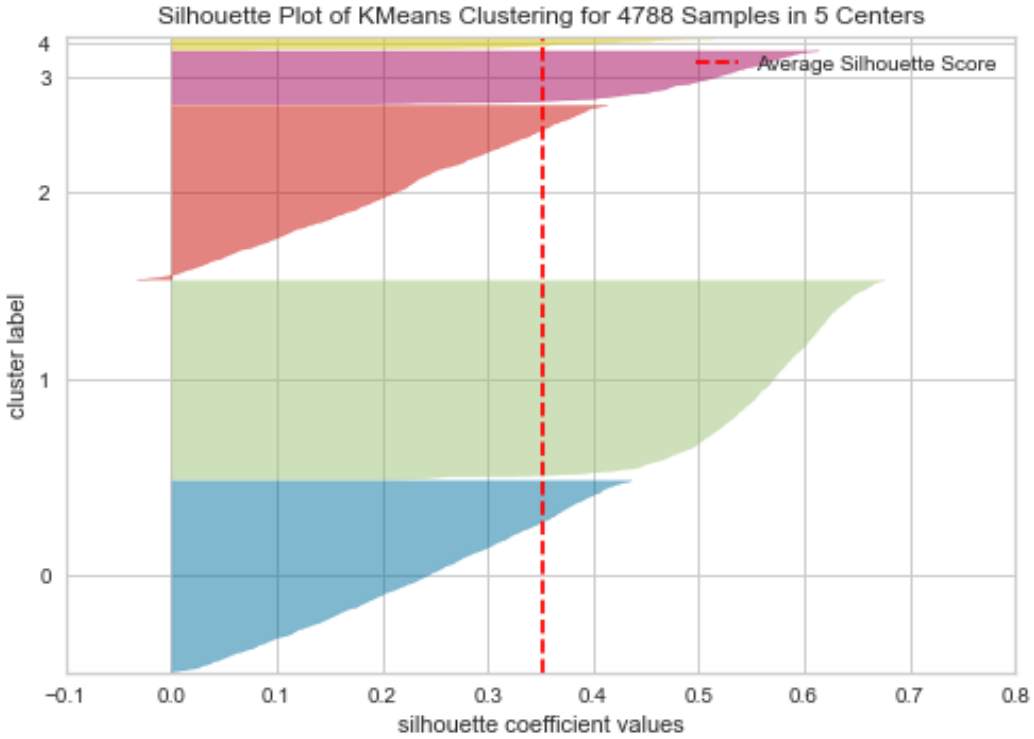
## Visualisation des clusters créés : t-sne



# K-Means

## Qualité des clusters



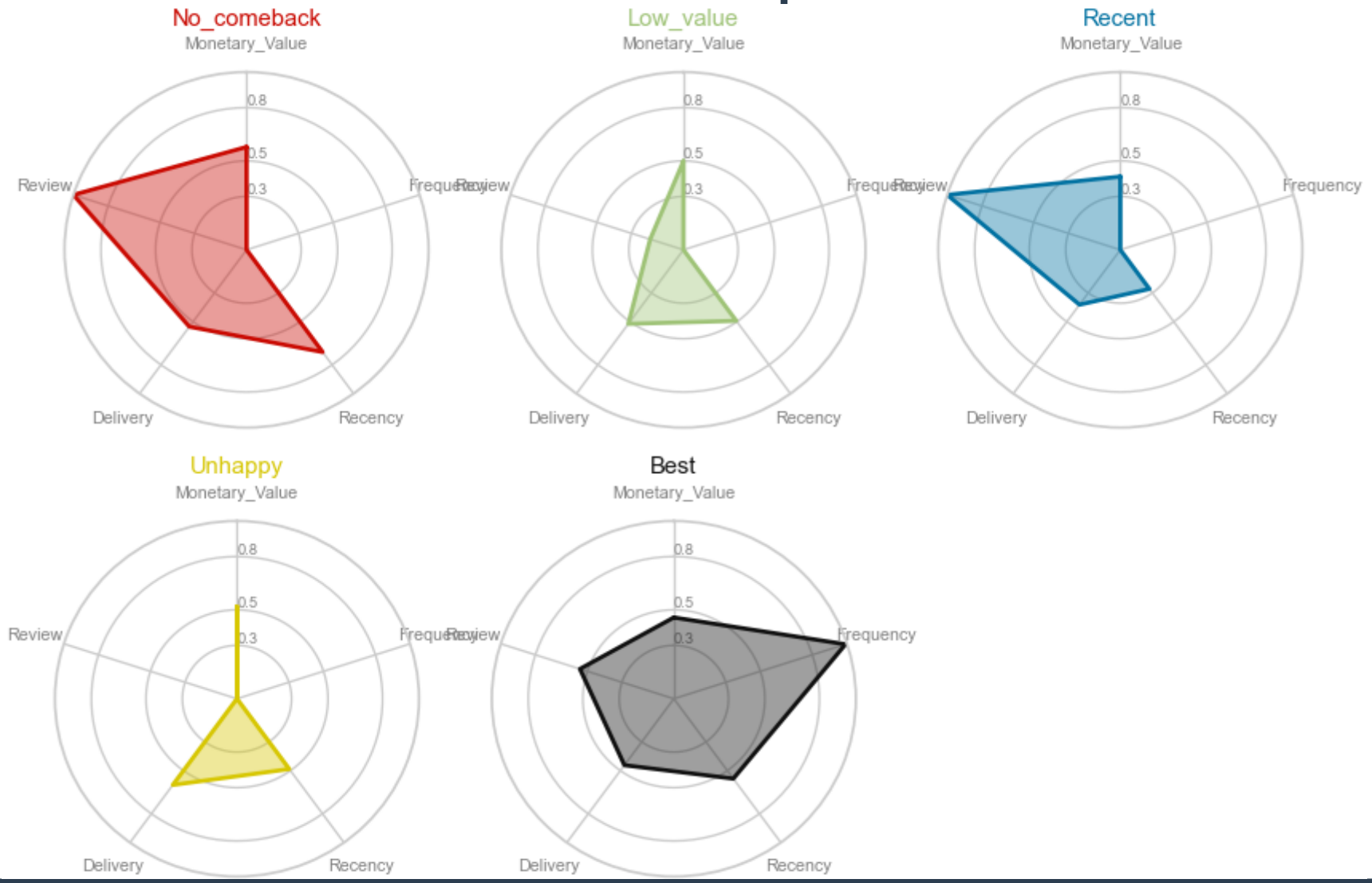




We can see 3 clusters distant from each other.
But 2 clusters overlap.

As expected, because of the dataset,
the quality of the clusters is not very good.

# K-Means

## Détermination des caractéristiques des clusters

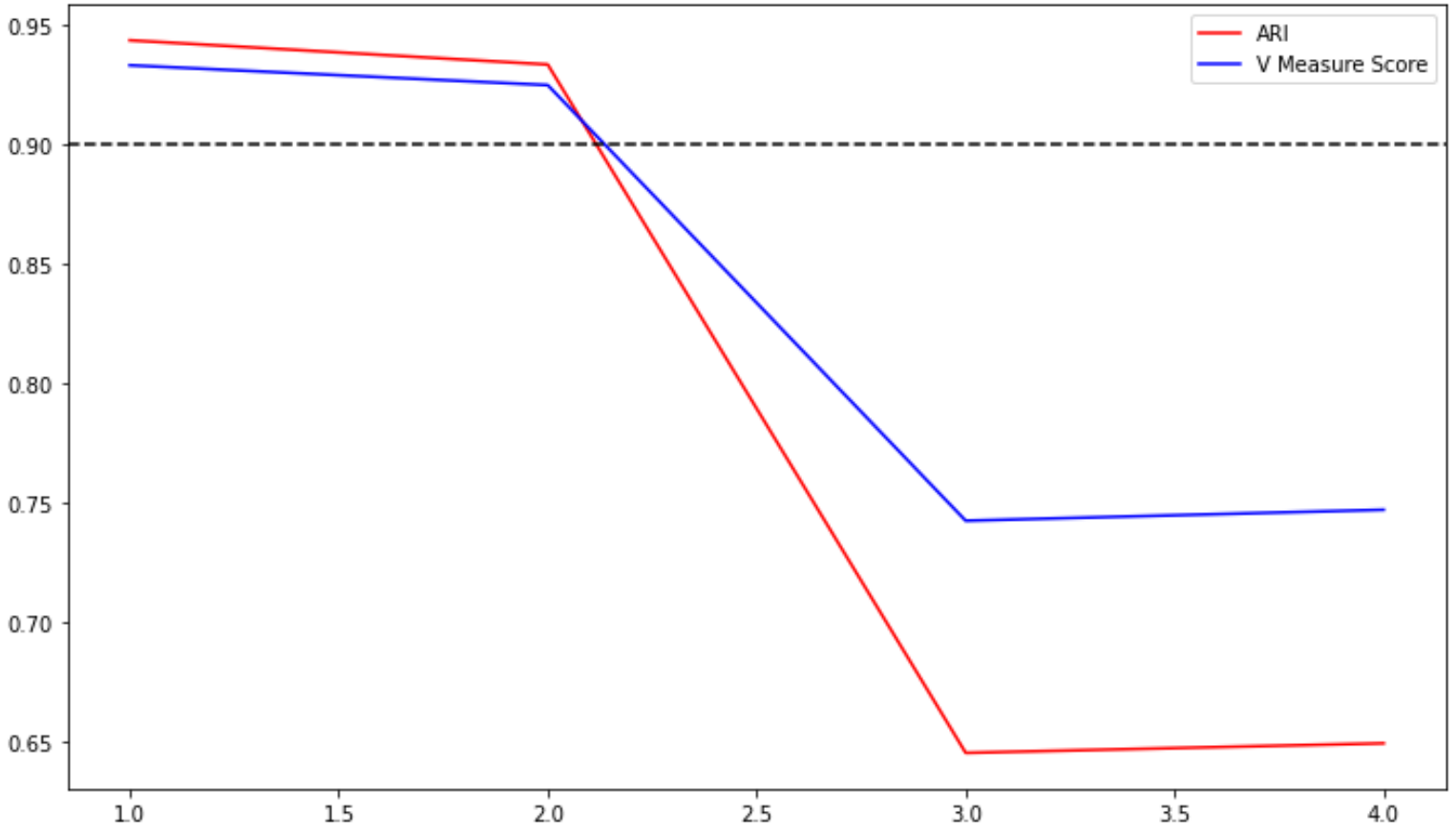


# Choix du modèle final
# Stabilité

# K-Means - Stabilité

## ARI et V Measure

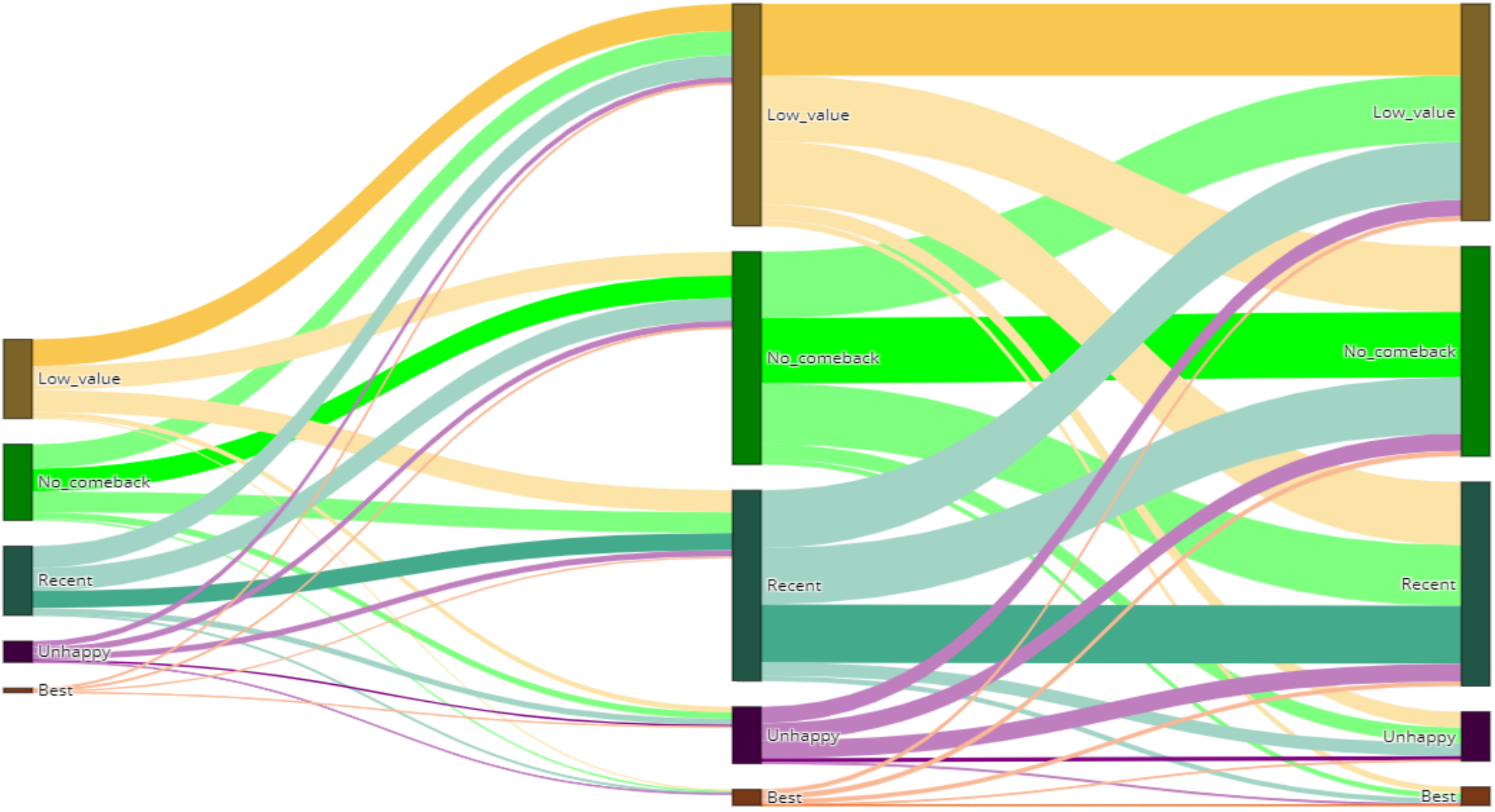
The x axis represent the number of trimesters we add to the dataset.




After 2 additions (9 months in total), the randscore is still above 0.90.
After 3 additions (12 months in total), the randscore is under 0.90.
We will have to fit the model once again after 9 months.

# K-Means - Stabilité

## Diagramme de Sankey



# Synthèse

# Synthèse

- **Nettoyage/Analyse puis Transformation des features :**
  - Création de features (e.g. fréquence, ...)
  - log(x + 1) et scores pour RFM
  - QuantileTransformer pour K-Means
- **Segmentation RFM**
- **Modèle non supervisé : K-Means**
- **Avantages de notre modèle par rapport à RFM :**
  - Nombre de features non limité

    → meilleure souplesse pour comprendre le comportement des clients
  - Pas d'attribution arbitraire de scores (données brutes utilisées)
  - Évaluation possible de la qualité des clusters
  - Prédiction des comportements des clients

    → anticiper des pertes de clients,

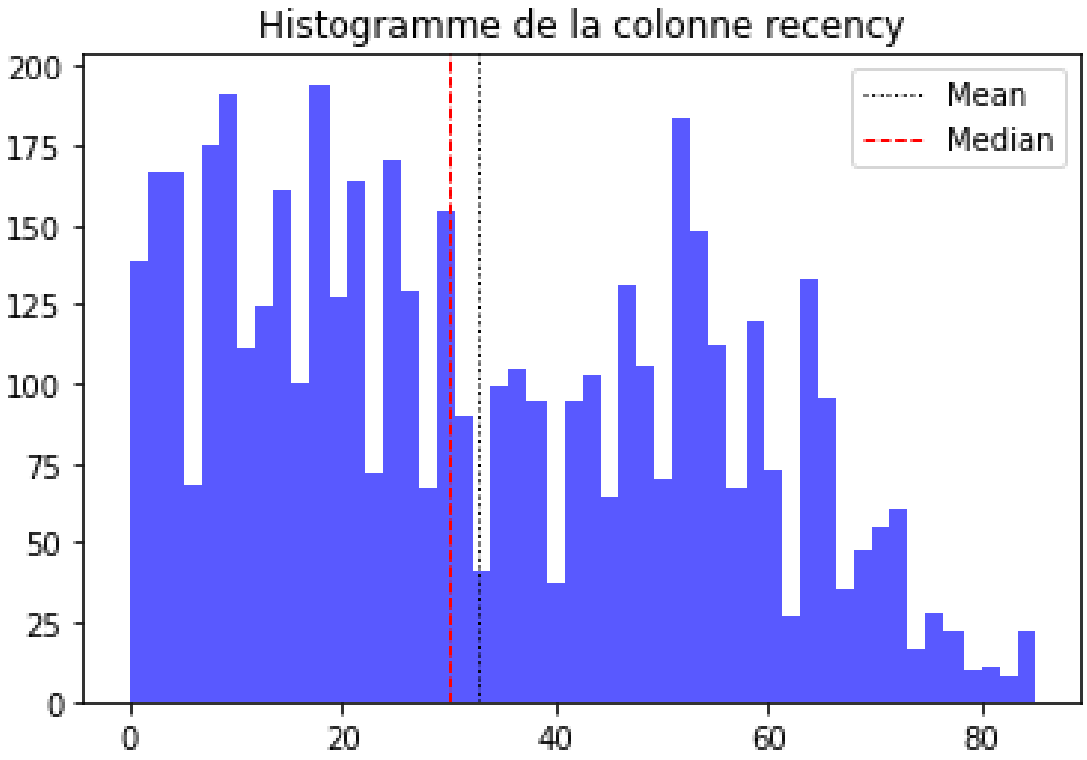
    → adapter sa stratégie marketing plusieurs mois à l'avance

# Merci de votre attention

# Annexes

# Annexes

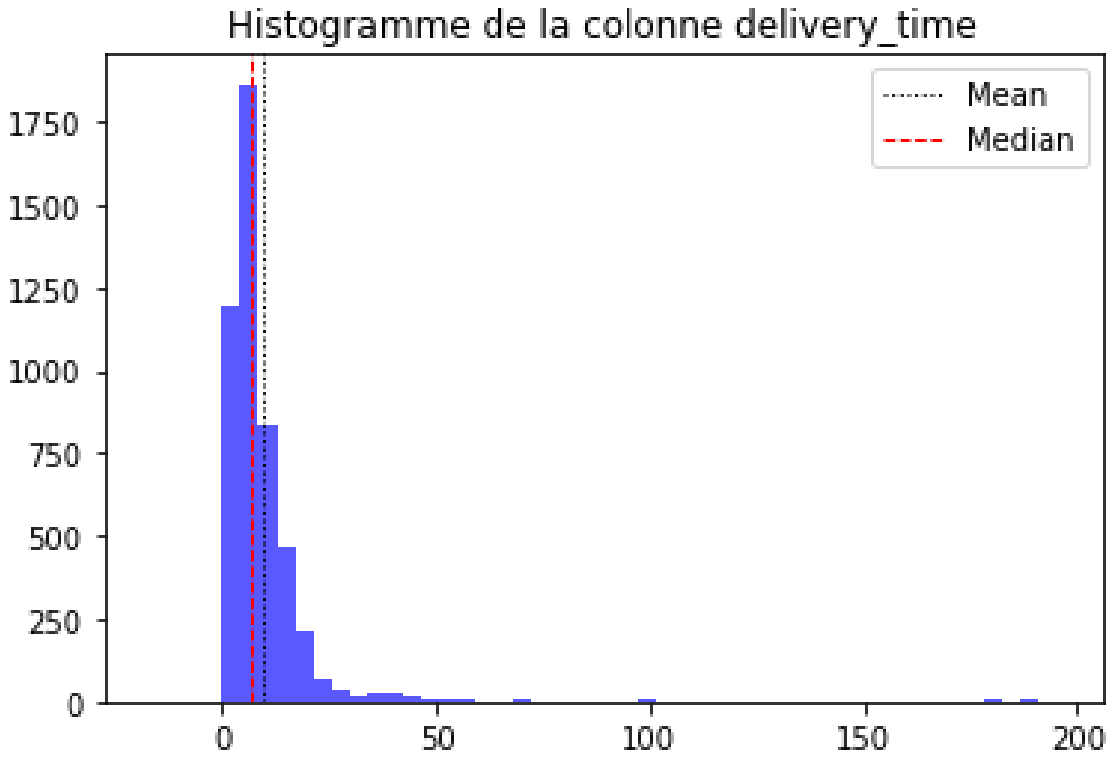
## Analyse univariée de recency :




| | | | |
|---|---|---|---|
| mean: | 32.894 | skewness: | 0.290 |
| median: | 30.000 | kurtosis: | -1.050 |
| var: | 467.605 | | |
| ect: | 21.624 | | |

# Annexes

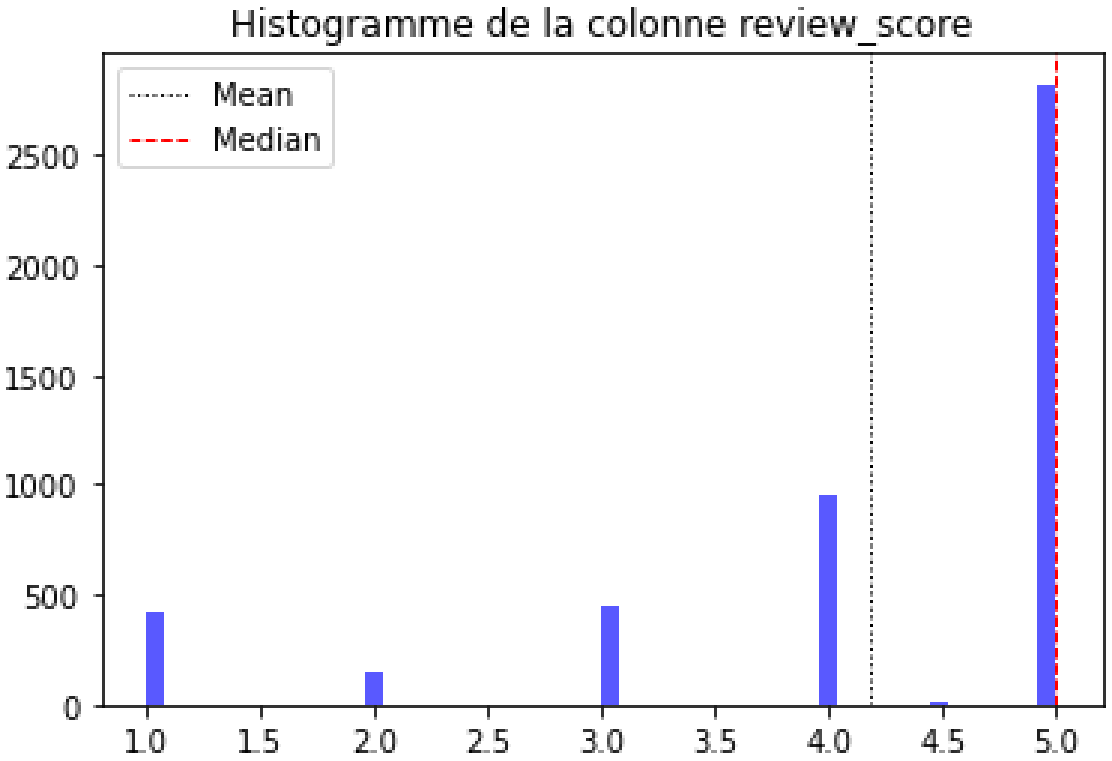
## Analyse univariée de delivery_time:




```
mean:    9.200    skewness: 9.390
median:  7.000    kurtosis: 122.290
var:     146.611
ect:     12.108
```

# Annexes

## Analyse univariée de review_score:




```
mean:     4.178   skewness: -1.510
median:   5.000   kurtosis: 1.140
var:      1.549
ect:      1.244
```

# Exploration / Analyse

## Analyse bivariée de payment_value :

→ on vérifie que les features ne sont pas trop fortement corrélées entre elles

| | payment_value corr with: | corr | p-value |
|---|---|---|---|
| 3 | delivery_time | 0.043166 | 0.002813 |
| 4 | review_score | -0.029205 | 0.043303 |
| 1 | frequency | -0.028028 | 0.052466 |
| 2 | recency | 0.013633 | 0.345599 |

# Annexes

## Analyse bivariée de recency :

→ on vérifie que les features ne sont pas trop fortement corrélées entre elles

| | recency corr with: | corr | p-value |
|---|---|---|---|
| 1 | frequency | 0.050972 | 0.000418 |
| 3 | delivery_time | -0.028765 | 0.046560 |
| 4 | review_score | 0.014284 | 0.323070 |
| 0 | payment_value | 0.013633 | 0.345599 |

# Annexes

## Analyse bivariée de frequency :

→ on vérifie que les features ne sont pas trop fortement corrélées entre elles

| | frequency corr with: | corr | p-value |
|---|---|---|---|
| 2 | recency | 0.050972 | 0.000418 |
| 0 | payment_value | -0.028028 | 0.052466 |
| 3 | delivery_time | -0.020084 | 0.164682 |
| 4 | review_score | -0.010005 | 0.488843 |

# Annexes

## Analyse bivariée de review_score :

→ on vérifie que les features ne sont pas trop fortement corrélées entre elles

| | review_score corr with: | corr | p-value |
|---|---|---|---|
| 3 | delivery_time | -0.192996 | 2.111405e-41 |
| 0 | payment_value | -0.029205 | 4.330305e-02 |
| 2 | recency | 0.014284 | 3.230701e-01 |
| 1 | frequency | -0.010005 | 4.888426e-01 |

# Résumé nettoyage

- Dataset pour segmentation RFM :

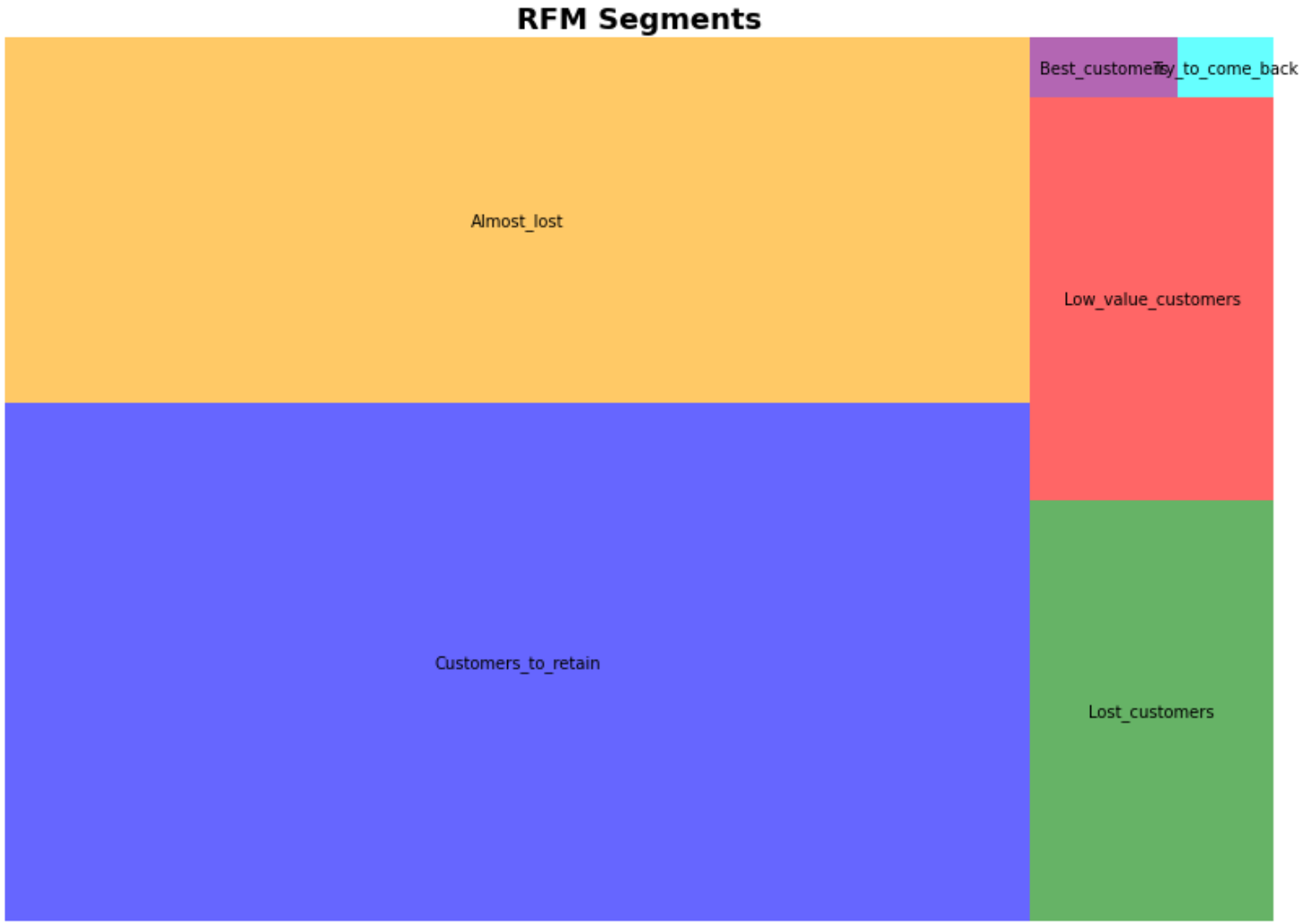
```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 4788 entries, 0 to 4848
Data columns (total 3 columns):
 #   Column         Non-Null Count  Dtype
---  ------         --------------  -----
 0   payment_value  4788 non-null   float64
 1   frequency      4788 non-null   int64
 2   recency        4788 non-null   int64
dtypes: float64(1), int64(2)
memory usage: 149.6 KB
```

- Dataset pour modèle non supervisé :

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4788 entries, 0 to 4787
Data columns (total 5 columns):
 #   Column          Non-Null Count  Dtype
---  ------          --------------  -----
 0   Monetary_Value  4788 non-null   float64
 1   Frequency       4788 non-null   float64
 2   Recency         4788 non-null   float64
 3   Delivery        4788 non-null   float64
 4   Review          4788 non-null   float64
dtypes: float64(5)
memory usage: 187.2 KB
```

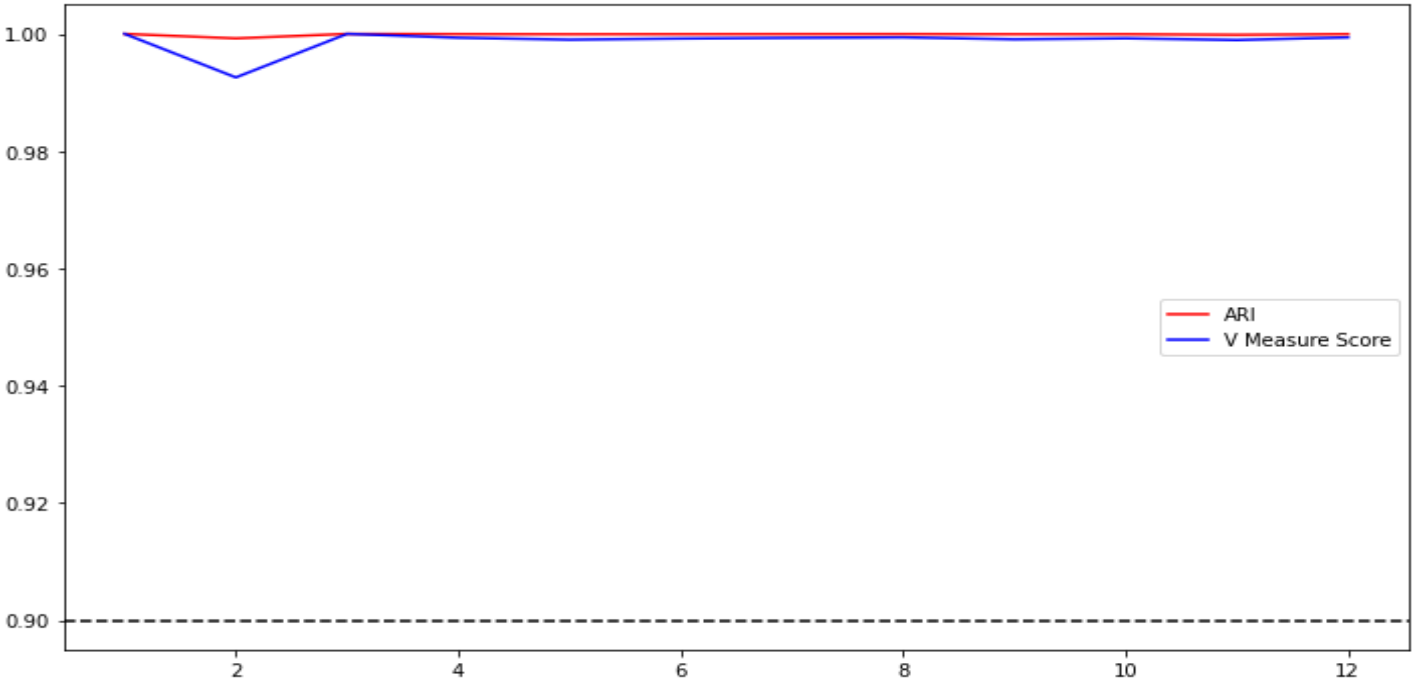
# Modélisations effectuées - RFM

## Taille des segments



# Annexes

## Kmeans tous les mois :

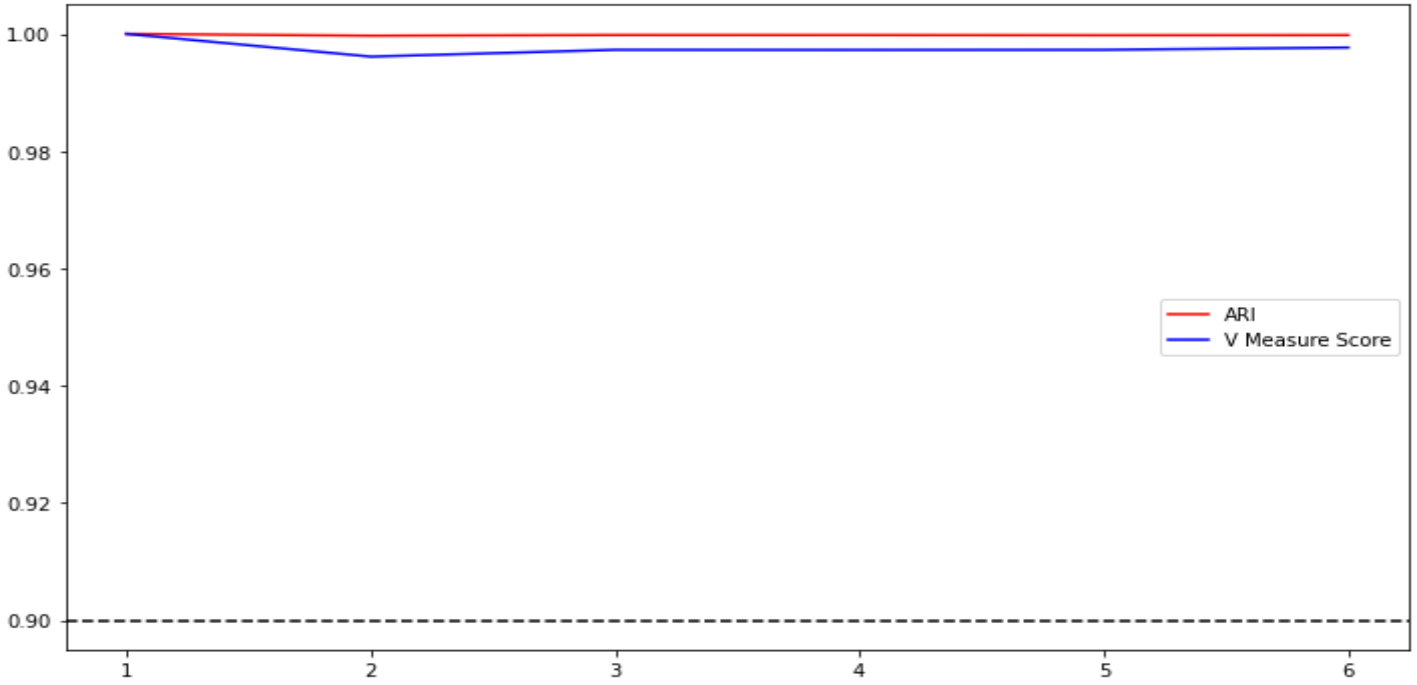



| | group | Monetary_Value | Frequency | Recency | Delivery | Review |
|---|---|---|---|---|---|---|
| 0 | 0 | 0.497603 | 0.000000 | 0.000000 | 0.552465 | 0.725182 |
| 1 | 1 | 0.502806 | 0.000000 | 0.535227 | 0.464171 | 1.000000 |
| 2 | 2 | 0.499450 | 0.000000 | 0.531342 | 0.531963 | 0.245777 |
| 3 | 3 | 0.380616 | 0.984239 | 0.606890 | 0.420489 | 0.818684 |
| 4 | 4 | 0.524472 | 0.028829 | 0.543192 | 0.589379 | 0.000000 |

The dataset is only split with frequency and review features.
The average of the others are too similar.

# Annexes

## Kmeans tous les 2 mois :

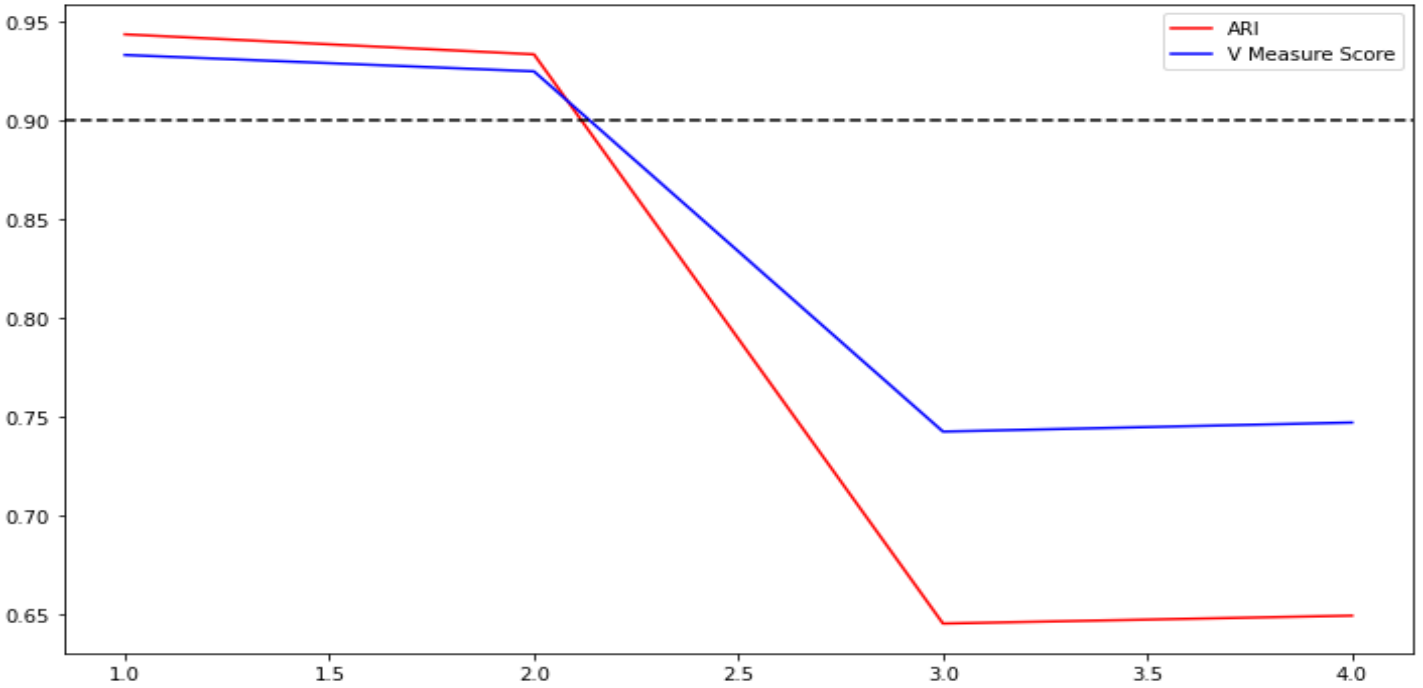



| | group | Monetary_Value | Frequency | Recency | Delivery | Review |
|---|---|---|---|---|---|---|
| 0 | 0 | 0.513561 | 0.000000 | 0.503364 | 0.568226 | 0.000000 |
| 1 | 1 | 0.501391 | 0.000000 | 0.502114 | 0.469659 | 1.000000 |
| 2 | 2 | 0.497012 | 0.000000 | 0.477953 | 0.512701 | 0.231233 |
| 3 | 3 | 0.447644 | 0.990663 | 0.587667 | 0.421393 | 0.640079 |

The dataset is only split with frequency and review features.
The average of the others are too similar.

# Annexes

## Kmeans tous les 3 mois :

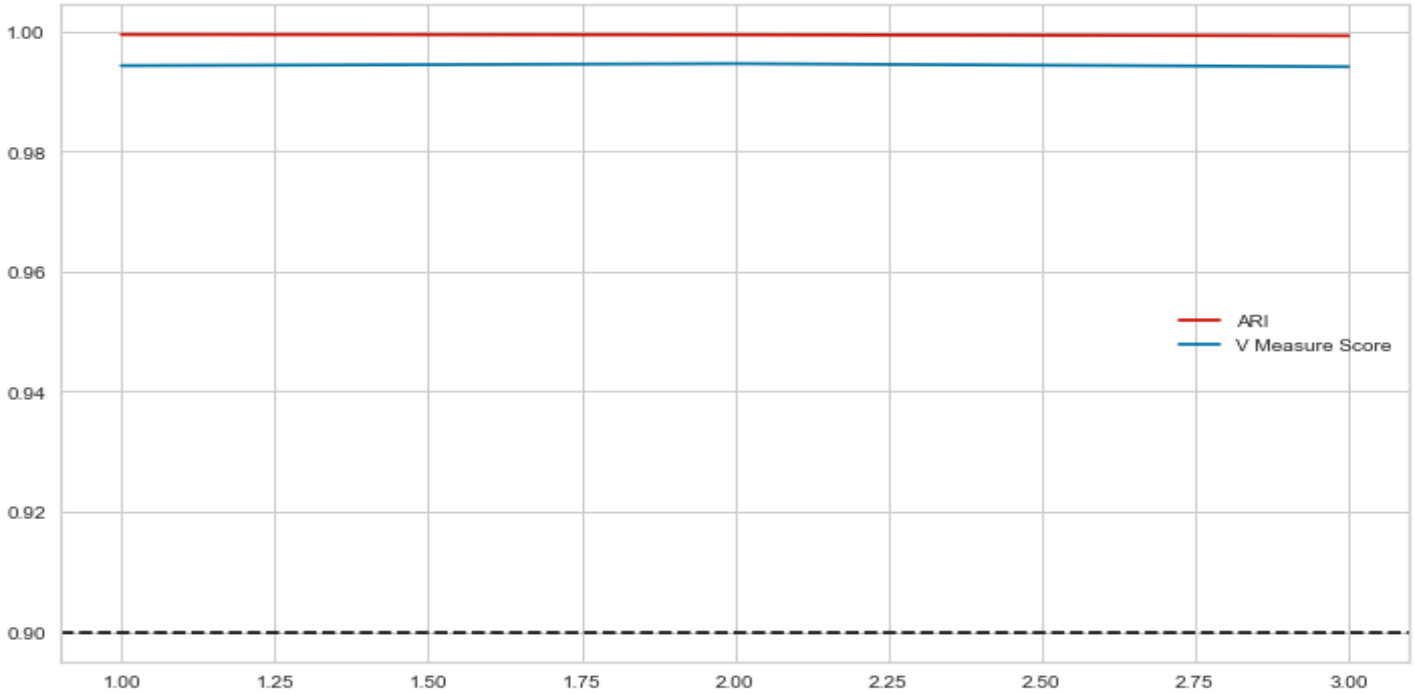



| | group | Monetary_Value | Frequency | Recency | Delivery | Review |
|---|---|---|---|---|---|---|
| 0 | 0 | 0.578581 | 0.000000 | 0.710791 | 0.534507 | 1.000000 |
| 1 | 1 | 0.498874 | 0.000000 | 0.493724 | 0.514953 | 0.192047 |
| 2 | 2 | 0.411967 | 0.000000 | 0.272093 | 0.381391 | 1.000000 |
| 3 | 3 | 0.518713 | 0.000000 | 0.489073 | 0.599369 | 0.000000 |
| 4 | 4 | 0.457174 | 0.991394 | 0.555596 | 0.461900 | 0.542325 |

The 5 groups seem relatively different according to all the features.

# Annexes

## Kmeans tous les 4 mois :

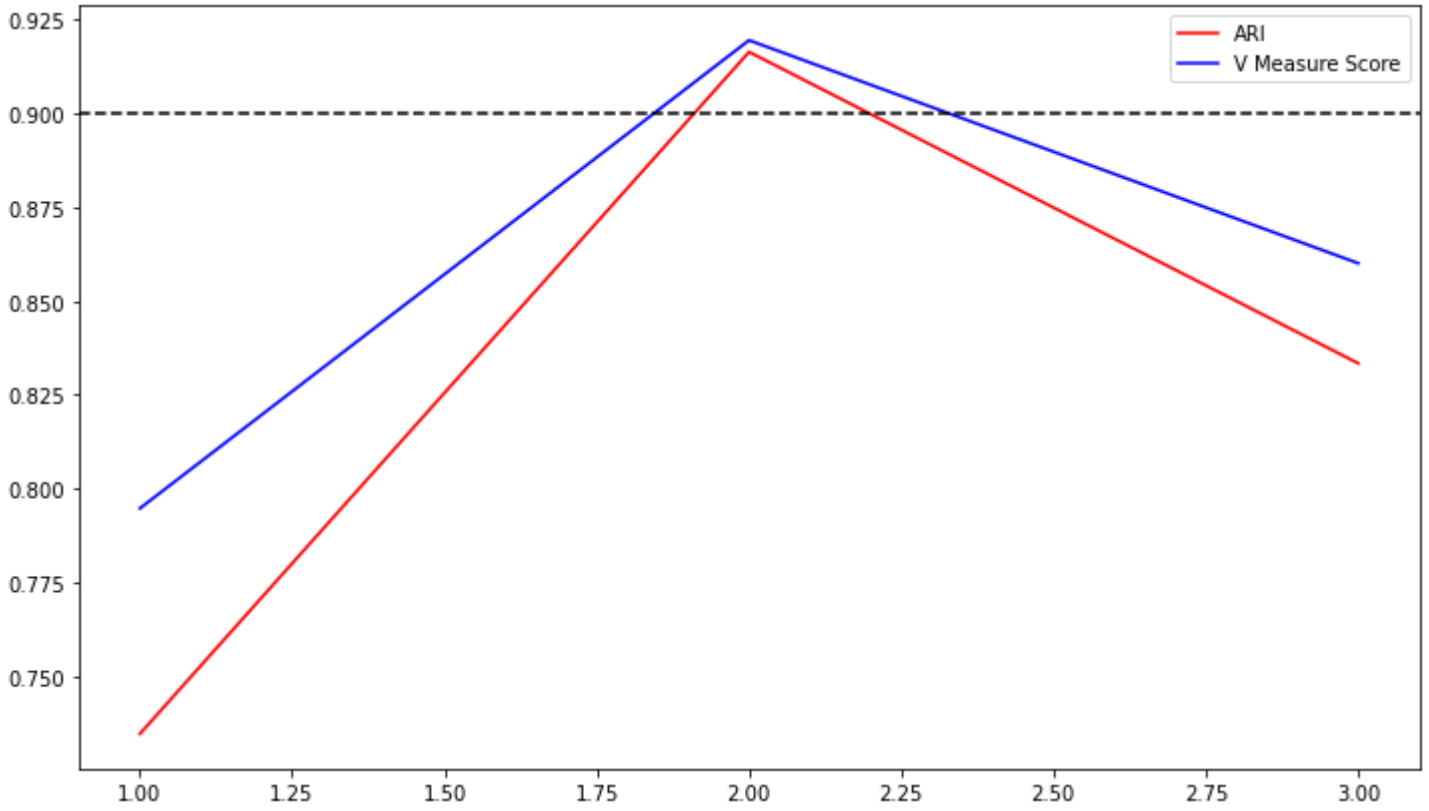



| | group | Monetary_Value | Frequency | Recency | Delivery | Review |
|---|---|---|---|---|---|---|
| 0 | 0 | 0.499659 | 0.000000 | 0.505882 | 0.459455 | 1.000000 |
| 1 | 1 | 0.495862 | 0.000000 | 0.490119 | 0.521923 | 0.252373 |
| 2 | 2 | 0.528131 | 0.000000 | 0.485867 | 0.614667 | 0.000000 |
| 3 | 3 | 0.449736 | 0.991304 | 0.531338 | 0.446655 | 0.611774 |

For all the groups, the monetary and recency features do not seem very different.

# Annexes

## Kmeans tous les 6 mois :




```
For the first prediction, the rand score is under 0.75.
We will not use 6 months in order to train our algorithm.
```

# K-Means - Stabilité

## Diagramme de Sankey