



## DS P5 – Segmentez les comportements de clients



# Sommaire

1. Contexte
2. Exploration
3. Algorithme
4. Modélisation
5. Conclusion

# Contexte - Objectif

## Les données

Dataset: OnlineRetail.xlsx (transaction d'achat d'un site de vente en ligne britannique, 8 variables, 541909 lignes)

| # | Column      |
|---|-------------|
| 0 | InvoiceNo   |
| 1 | StockCode   |
| 2 | Description |
| 3 | Quantity    |
| 4 | InvoiceDate |
| 5 | UnitPrice   |
| 6 | CustomerID  |
| 7 | Country     |

## Modèle de classification des clients

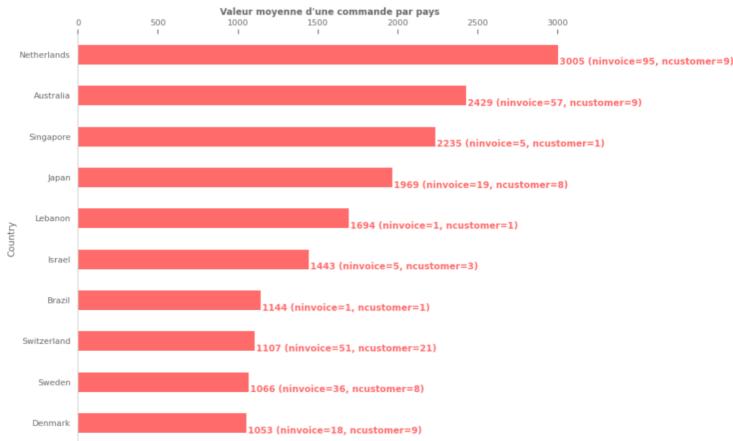
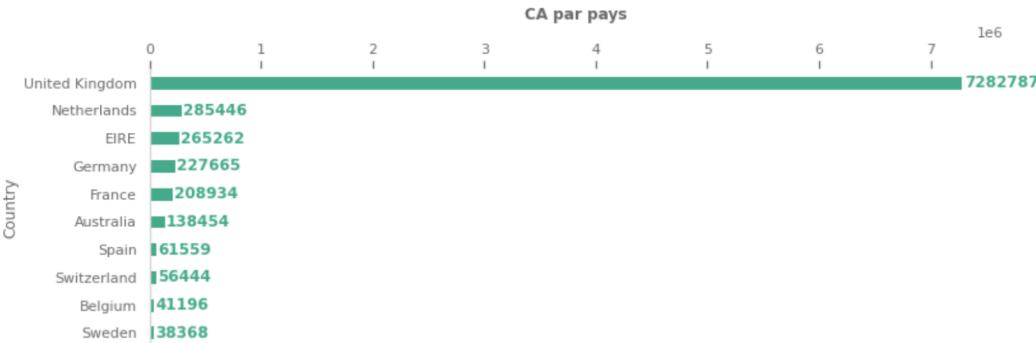
Comprendre les différents types de clients pour détecter les plus susceptibles de passer à l'achat.

# Contexte - Problématique

- Quelles variables utiliser pour déterminer les catégories intéressantes?
- Quel modèle choisir pour classer les clients dans ces catégories?
- Comment améliorer les performances du modèle?

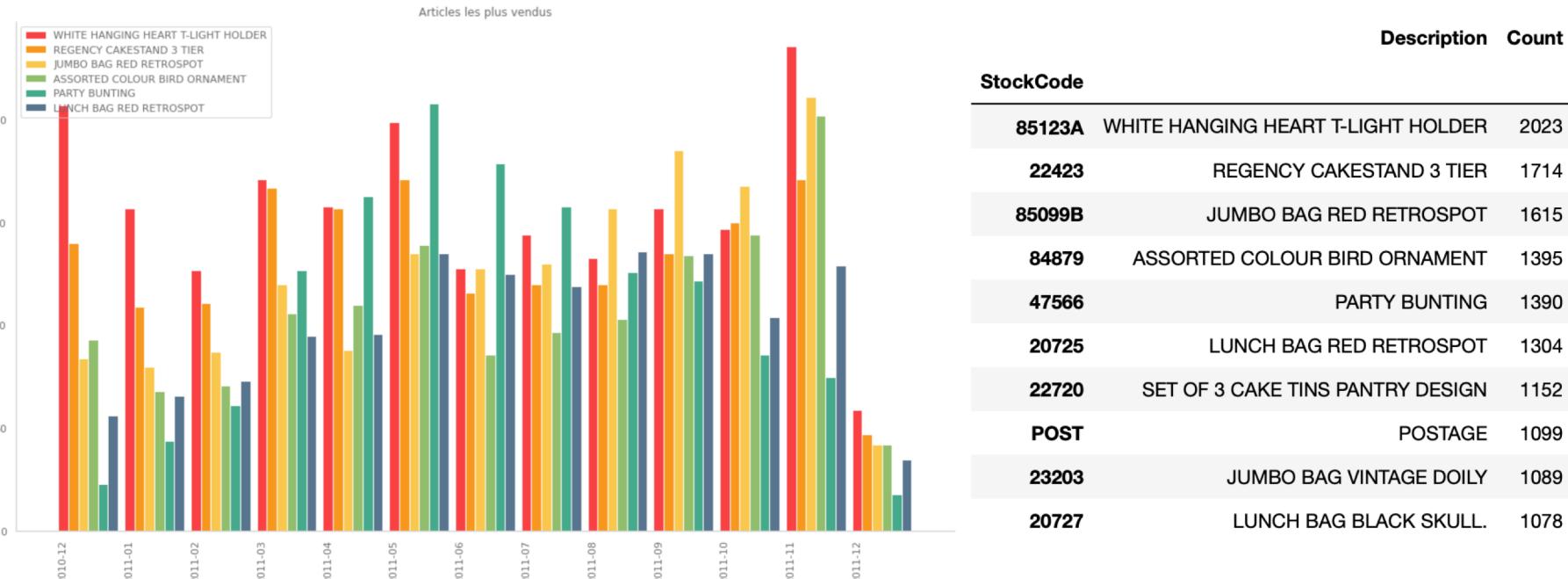


# Exploration - Vente



- CA croissant sur l'année
- CA total le plus élevé <> CA moyen le plus élevé par pays

# Exploration - Produit



- Le classement des ventes mensuelles diffère du classement des ventes annuelles

# Exploration - Client

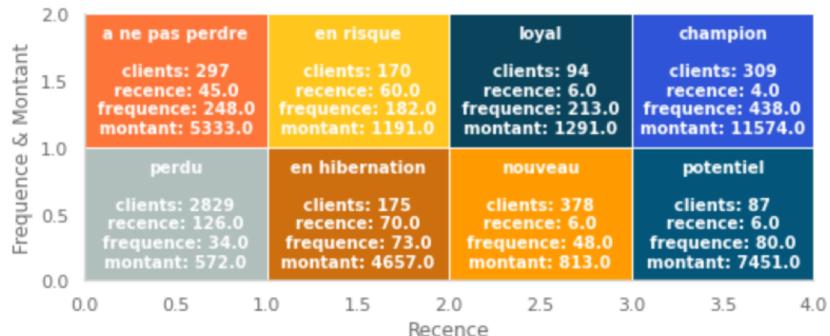
## Segmentation RFM

- Récence : date du dernier achat ou dernier contact client
- Fréquence : fréquence des achats sur une période de référence donnée
- Montant : somme des achats cumulés sur cette période

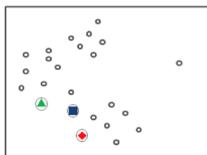
| RFMScore | CustomerID | Recence | Frequence | Montant | RScore  | FScore | MScore |
|----------|------------|---------|-----------|---------|---------|--------|--------|
| 7        | 222        | 309     | 4.0       | 438.0   | 11574.0 | 2.0    | 2.0    |
| 6        | 221        | 94      | 6.0       | 213.0   | 1291.0  | 2.0    | 2.0    |
| 5        | 212        | 87      | 6.0       | 80.0    | 7451.0  | 2.0    | 1.0    |
| 4        | 211        | 378     | 6.0       | 48.0    | 813.0   | 2.0    | 1.0    |
| 3        | 122        | 297     | 45.0      | 248.0   | 5333.0  | 1.0    | 2.0    |
| 2        | 121        | 170     | 60.0      | 182.0   | 1191.0  | 1.0    | 2.0    |
| 1        | 112        | 175     | 70.0      | 73.0    | 4657.0  | 1.0    | 1.0    |
| 0        | 111        | 2829    | 126.0     | 34.0    | 572.0   | 1.0    | 1.0    |

## Score RFM selon le principe de Pareto

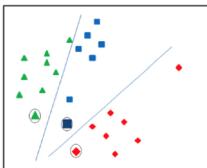
- 2 : Le quantile des premier 20% a le score le plus élevé
- 1: Le quantile des dernier 80% a le score le plus faible



# Algorithme – Kmeans

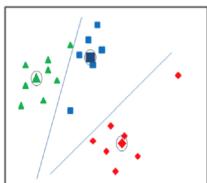


1. Définir les centroïdes aléatoirement en leur associant une étiquette à chacun



2. Associer chaque point au centroïde le plus proche et l'étiqueter correspondante

$$d = \sqrt{\sum_i (x - x_i^2) + (y - y_i)^2}$$

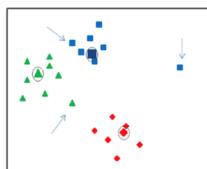


3. Recalculer le centre de chaque cluster et mettre à jour son centroïde

$$CP(x_1, x_2, \dots, x_k) = \left( \frac{\sum_{i=1}^k x1st_i}{k}, \frac{\sum_{i=1}^k x2nd_i}{k}, \dots, \frac{\sum_{i=1}^k xnth_i}{k} \right)$$

Ex.: centroïde de 3 points, (2,4), (5,2) and (8,9)

$$CP = \left( \frac{2+5+8}{3}, \frac{4+2+9}{3} \right) = (5,5)$$



4. Répéter l'assignation et calculer les nouveaux centroïdes jusqu'à ce que les nouveaux centroïdes ne bougent plus par rapport aux précédents

# Algorithme – Score Silhouette

$$s(\mathbf{o}) = \frac{b(\mathbf{o}) - a(\mathbf{o})}{\max\{a(\mathbf{o}), b(\mathbf{o})\}}$$

- $a(\mathbf{o})$ : moyenne des distances de chaque points aux autres points d'un même cluster
- $b(\mathbf{o})$ : distance moyenne d'un point d'un cluster aux autres points du cluster le plus proche

Ce score peut varier entre -1 et +1.

- +1 signifie que l'observation est située bien à l'intérieur de son propre cluster
- 0 signifie que l'observation se situe près d'une frontière,
- -1 signifie qu'une observation a été attribuée au mauvais cluster

# Modélisation supervisée – Bayes Naïf

| Class | TP | FP  | TN  | FN  | TPR | FPR      | Accuracy |          |
|-------|----|-----|-----|-----|-----|----------|----------|----------|
| 5     | 5  | 9   | 2   | 660 | 20  | 0.310345 | 0.003021 | 0.968162 |
| 6     | 6  | 11  | 7   | 658 | 20  | 0.354839 | 0.010526 | 0.961207 |
| 7     | 7  | 14  | 2   | 655 | 26  | 0.350000 | 0.003044 | 0.959828 |
| 2     | 2  | 21  | 17  | 648 | 95  | 0.181034 | 0.025564 | 0.856594 |
| 1     | 1  | 20  | 11  | 649 | 103 | 0.162602 | 0.016667 | 0.854406 |
| 4     | 4  | 31  | 80  | 638 | 40  | 0.436620 | 0.111421 | 0.847909 |
| 3     | 3  | 11  | 25  | 658 | 105 | 0.094828 | 0.036603 | 0.837297 |
| 0     | 0  | 552 | 271 | 117 | 6   | 0.989247 | 0.698454 | 0.707188 |

- Pré-requis:
  - suppression des valeurs atypiques par méthode des quartiles
  - Encodage des score RFM
- Target: score RFM vu en exploration
- Exactitude: 0.62
- R2: 0.32
- L'exactitude par classe est meilleure pour les classes 5 (212), 6 (221) et 7 (222)
- Le modèle donne globalement 62% de mesures correctes
- Faible R2: le modèle explique le score seulement à 32%

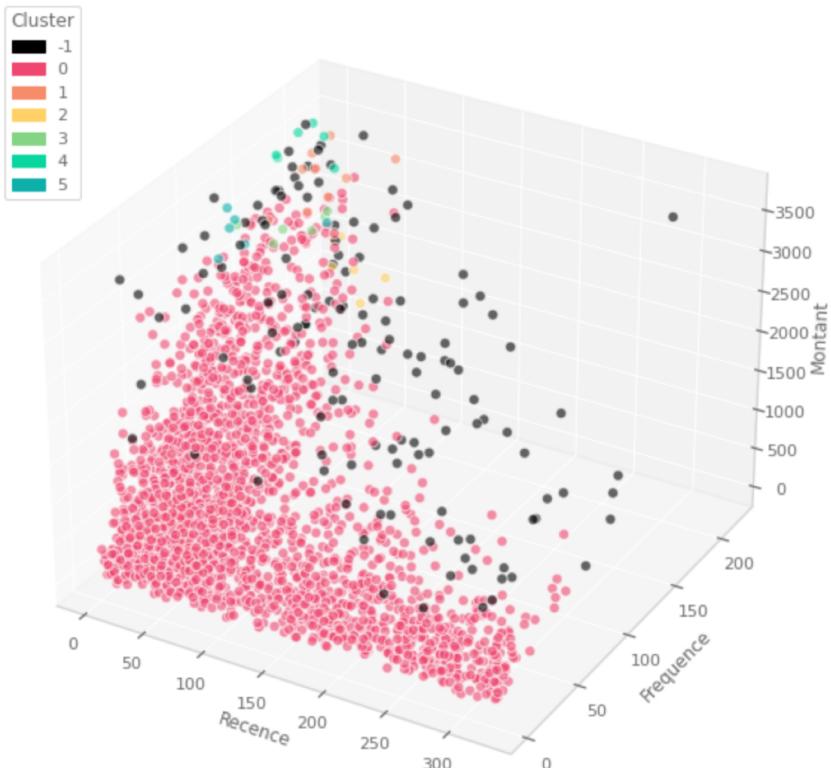
# Modélisation supervisée – KNN

| Class | TP | FP  | TN | FN  | TPR | FPR      | Accuracy |          |
|-------|----|-----|----|-----|-----|----------|----------|----------|
| 5     | 5  | 2   | 9  | 946 | 3   | 0.400000 | 0.009424 | 0.987500 |
| 7     | 7  | 6   | 10 | 942 | 7   | 0.461538 | 0.010504 | 0.982383 |
| 6     | 6  | 3   | 15 | 945 | 4   | 0.428571 | 0.015625 | 0.980352 |
| 3     | 3  | 23  | 13 | 925 | 10  | 0.696970 | 0.013859 | 0.976313 |
| 1     | 1  | 24  | 7  | 924 | 17  | 0.585366 | 0.007519 | 0.975309 |
| 2     | 2  | 19  | 19 | 929 | 8   | 0.703704 | 0.020042 | 0.972308 |
| 4     | 4  | 65  | 46 | 883 | 19  | 0.773810 | 0.049516 | 0.935834 |
| 0     | 0  | 806 | 17 | 142 | 68  | 0.922197 | 0.106918 | 0.917715 |

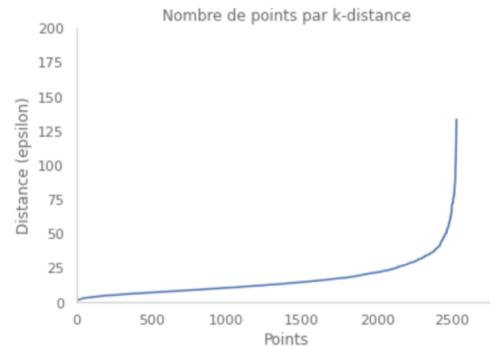
- Pré-requis:
  - suppression des valeurs atypiques par méthode des quartiles
  - Encodage des score RFM
- Target: score RFM vu en exploration
- Exactitude: 0.87
- R2: 0.39
- L'exactitude par classe est > 90% pour chacune des classes
- Le modèle donne globalement 87% de mesures correctes
- Faible R2: le modèle explique le score seulement à 39%

# Modélisation non supervisée -DBScan

Client RFM - Clustering DBSCAN



- **Paramètres**
  - Epsilon: 57
  - Min samples:  $2 * 3$
- **Score de silhouette: 0.3**

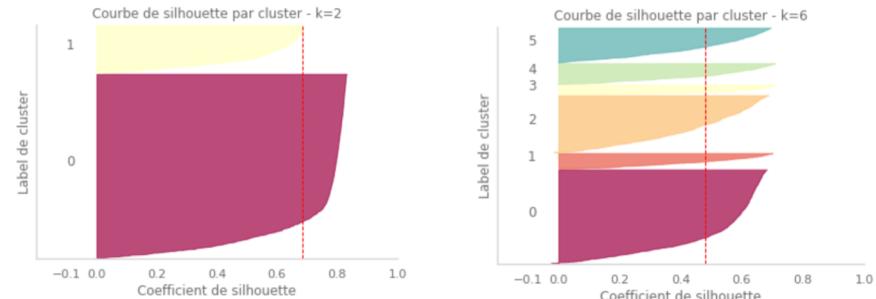


- **Score de silhouette faible: la distance entre les clusters n'est pas significative**

# Modélisation non supervisée – KMeans



- Détermination du nombre maximisé de cluster selon le score Silhouette



- Validation croisées sur 3 périodes

```
date max période[0]:2011-05-17 10:08:00 Score Silhouette[0]:0.5554376071972892
date max période[1]:2011-09-29 15:58:00 Score Silhouette[1]:0.5283042689635439
date max période[2]:2011-12-09 12:50:00 Score Silhouette[2]:0.5492622956571671
```

- Score de silhouette  $\sim 0.5$  : la distance entre les clusters est nette sur l'axe des montants mais pas sur les autres axes

# Conclusion

- Les +
  - Le clustering Kmeans complète la vision de la segmentation manuelle
  - Le clustering non supervisé permet de lever la limitation sur la taille du jeu de données
  - Classification manuelle par segments RFM, classique mais permet d'agir
- Les -
  - La segmentation RFM n'a pas de matching avec le clustering K-Means
- Pour la suite
  - Ajouter des variables au modèle
  - Tester d'autres modèles de classification supervisées : SVM multiclass