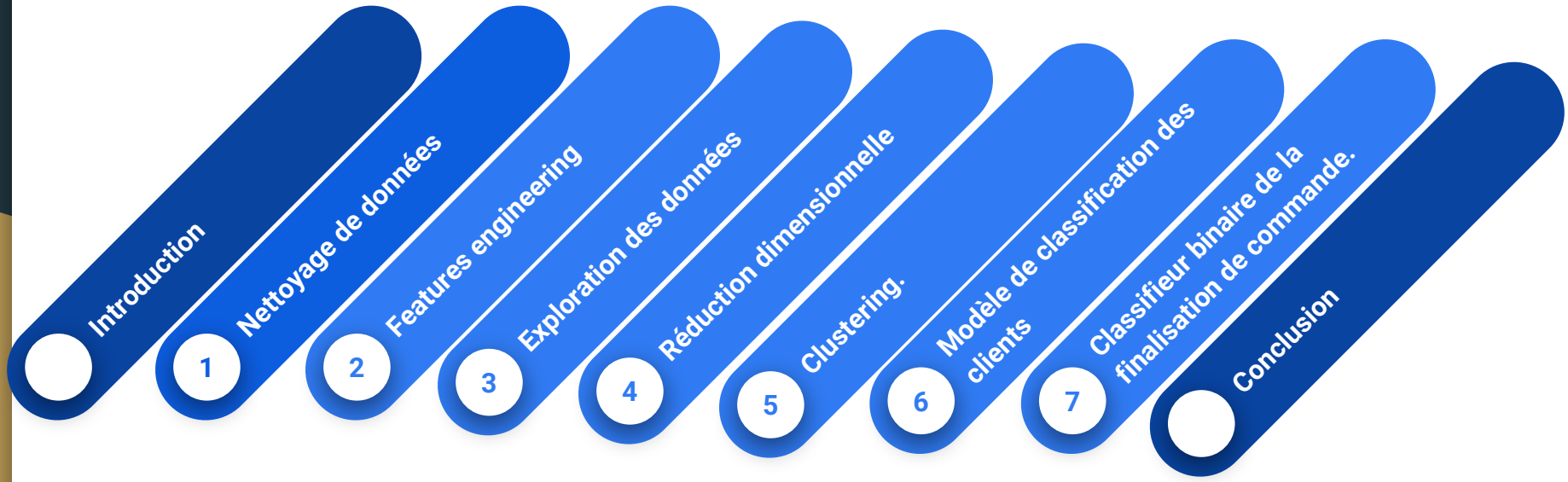
Projet 5: Segmenter les comportements de clients

Réalisé par:Taher Haggui

Plan:


Introduction
1
Nettoyage de données
2
Features engineering
3
Exploration des données
4
Réduction dimensionnelle
5
Clustering.
6
Modèle de classification des clients
7
Classifieur binaire de la finalisation de commande.
Conclusion

Problématique:

Créer une application python, permettant de classer les clients en fonction de leur comportements temporelles d’achat.

Comment ?:

1.Recenser une base de données de milliers de transactions.

2.Nettoyage et exploration des données .

3.Tester plusieurs modèles.

4.Développement d’une application prenant en entrée une séquence temporelle client.

Résultats attendus:

Définir des catégories clients dignes d'intérêt.

Une application python permettant de classer le client des sa première opération d’achat.

Nettoyage de données: Valeurs manquantes

état initiale des valeurs manquantes

	InvoiceNo	StockCode	Description	Quantity	InvoiceDate	UnitPrice	CustomerID	Country
missing values number	0	0	1454	0	0	0	135080	0

Valeurs manquantes: Variable "Description"

Méthode d'imputation: **Chercher la description du produit par la valeur de sa référence s'il existe**

Créer une fonction permettant d'imputer les valeurs manquantes de la variable description par ses correctes valeurs.

Imputer 1342 valeurs manquantes. Il reste que 122 valeurs manquantes dans cette variable.

Valeurs manquantes: Variable “CustomerID”

Approche 1: **Chercher la valeur correcte de chaque valeurs manquante dans les lignes de même facture.**

Zéro valeurs manquantes ont été remplacés.

Approche 2: **Créer des nouvelles identifiants uniques pour ces valeurs manquantes.**

Créer une fonction permettant d’incrémenter des identifiants clients pour les valeurs manquantes à partir de la valeur maximale utilisée dans le jeu de données.

Nous avons remplacé toutes les valeurs manquantes de cette variable.

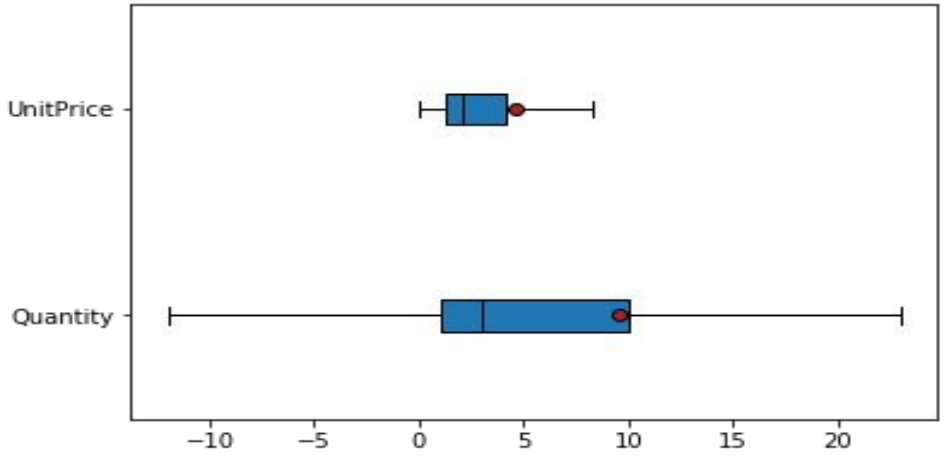
Valeurs manquantes:

Nouvelle état des valeurs manquantes.

	InvoiceNo	StockCode	Description	Quantity	InvoiceDate	UnitPrice	CustomerID	Country
missing values number	0	0	112	0	0	0	0	0

Nous avons supprimé la variable “Description”. C’est une variable inutile pour la segmentation des clients.

Valeurs aberrantes :



Nous constatons des valeurs négatives pour la variable Quantity.

Supprimer les transactions correspondaient à une remise.

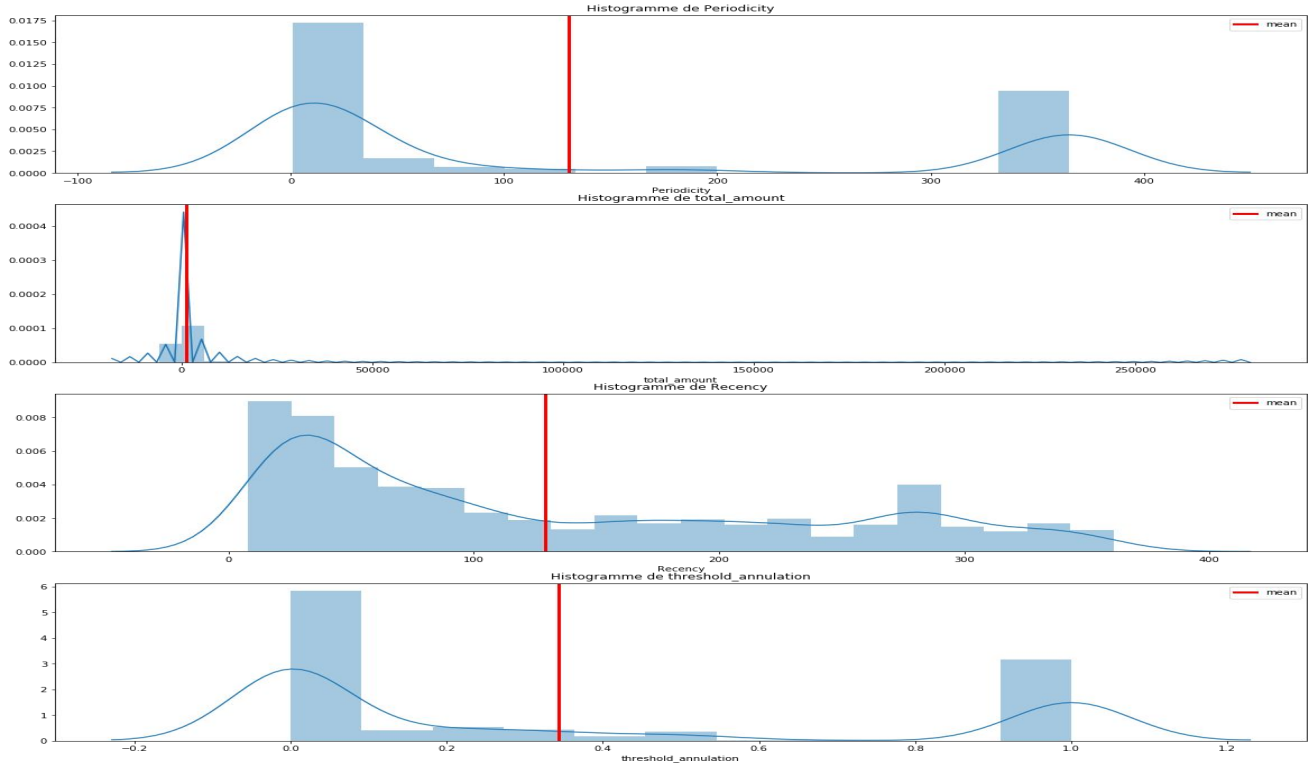
Features engineering:

- **Periodicity : correspond à la périodicité d'achat de chaque client.**
- **purchase_amount : correspond au montant de chaque transaction.**
- **total_amount : correspond au montant total annuel des achats de chaque client.**
- **Recency: correspond au temps écoulé depuis la dernière transaction d'achat de chaque client .**
- **Purchasing: une variable binaire permettant d'indiquer si la transaction a été réalisée ou annulée .**
- **threshold_annulation : correspond au pourcentage des commandes annulées de chaque client.**

Colonnes inutiles:

- InvoiceNo
- StockCode
- Quantity
- InvoiceDate
- UnitPrice
- Country

Exploration des données : Analyse univariée

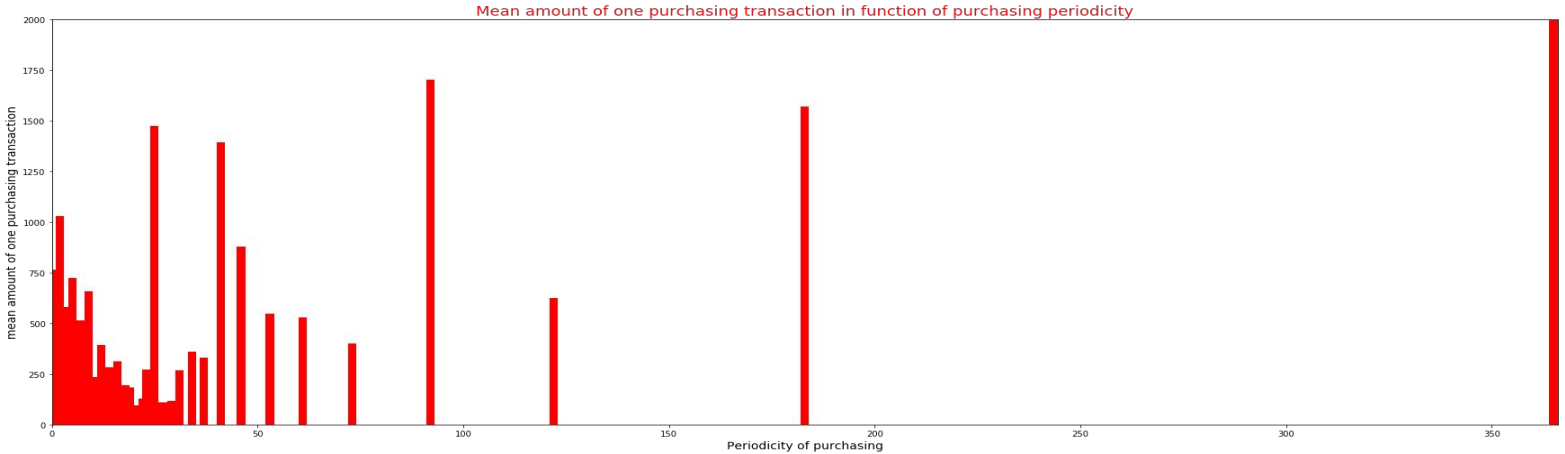


Exploration des données : Analyse univariée

	Periodicity	total_amount	Recency	threshold_annulation
mean	130.291	1206.81	129.092	0.343219
mode	0 365 dtype: int64	0 0.0 dtype: float64	0 23 dtype: int64	0 0.0 dtype: float64
Q1	6	0	36	0
Q3	365	1089.08	218.75	1
median	20	298.405	88	0

- ➔ **La période moyenne d'achat s'élève à 130 jours.**
- ➔ **50% des clients ont une périodicité inférieure à 20 jours.**
- ➔ **le montant moyen d'achat annuel du client s'élève à 1206 euros.**

Exploration des données : Analyse multivariée

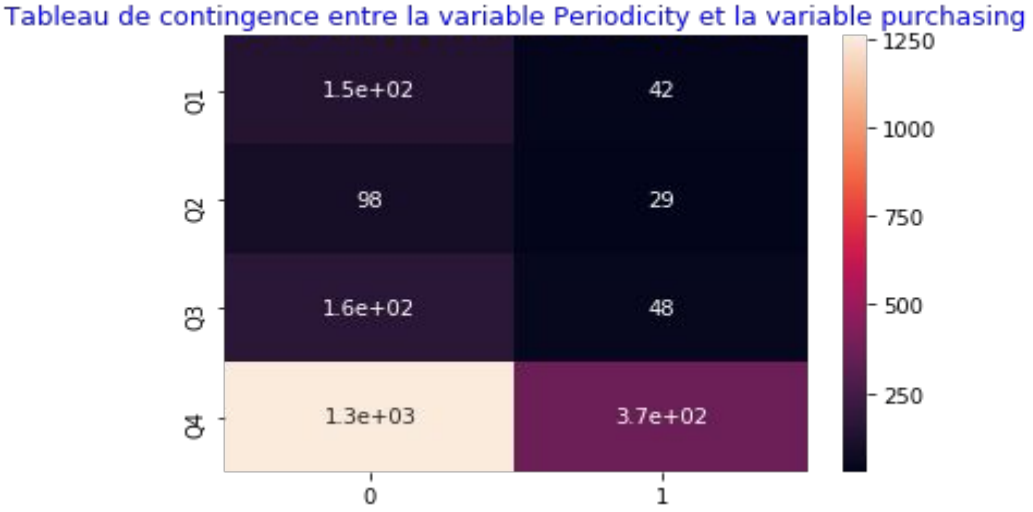


Nous pouvons constater d'après le graphe ci-dessus que les clients qui achètent moins souvent, ont des paniers moyen élevé.

Exploration des données : Analyse multivariée

- ➔ y-a-t-il une corrélation entre la périodicité d'achat du client et la finalisation de la commande.
- ➔ y-a-t-il une corrélation entre le montant de la transaction et la finalisation de la commande.
- ➔ y-a-t-il une corrélation entre le temps depuis la derniére achat et la finalisation de la commande.
- ➔ y-a-t-il une corrélation entre l'historique d'annulation de commande du client et la finalisation de la commande.

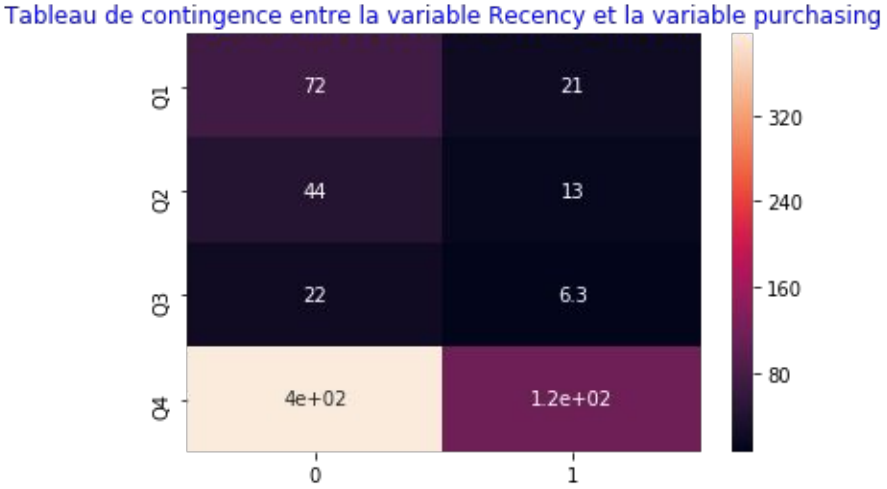
Exploration des données : Analyse multivariée



Coefficient de corrélation=2152

➔ La finalisation de la commande est fortement corrélés à la périodicité d'achat.

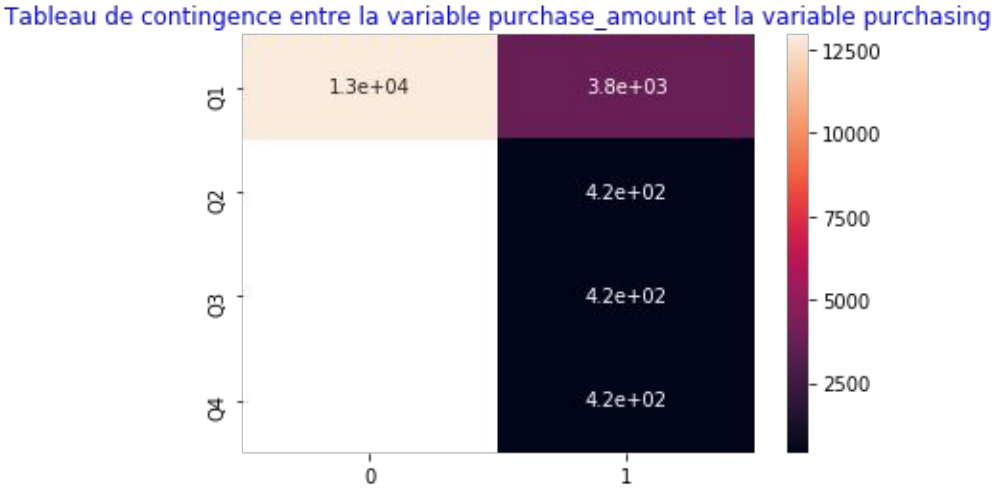
Exploration des données : Analyse multivariée



Coefficient de corrélation=670

➔ La finalisation de la commande est très corrélés au temps depuis la dernière transaction.

Exploration des données : Analyse multivariée

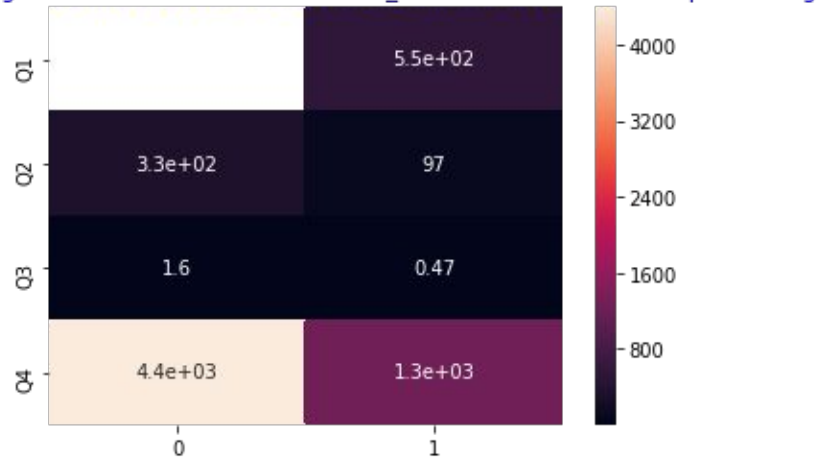


Coefficient de corrélation=17941

➔ La finalisation de la commande est très fortement corrélés au montant de la transaction.

Exploration des données : Analyse multivariée

Tableau de contingence entre la variable threshold_annulation et la variable purchasing

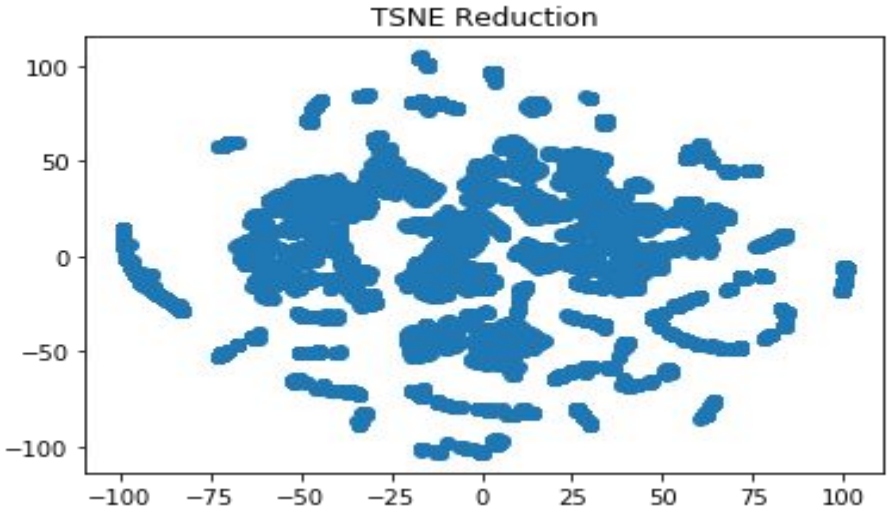


Coefficient de corrélation=50158

➔ La finalisation de la commande est très fortement corrélés à l'historique d'annulation du client.

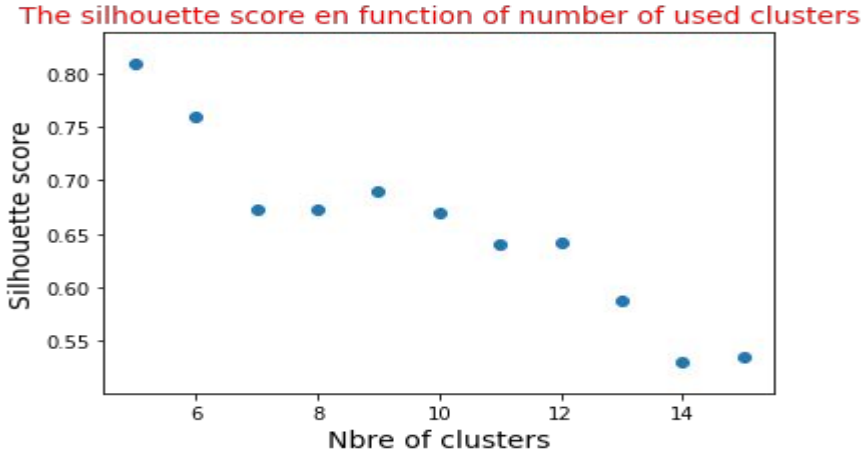
Réduction dimensionnelle:

Choix de méthode : La technique TSNE, permet de garder la structure locale.



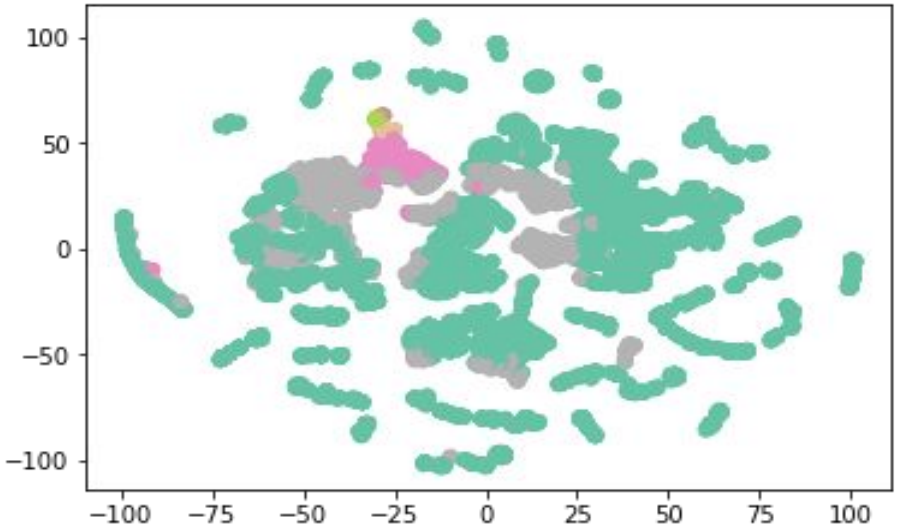
Clustering:

Méthode: KMEAN
Critère d'optimisation: le coefficient de silhouette.



Choix retenue= 9 clusters

Clustering:



Modèle de classification des clients:

Pour définir le meilleur modèle de classification des clients, nous avons testé plusieurs modèles à savoir :

- Bagging
- Random forest
- Adaboost
- SVM
- KNeigboors

Modèle de classification des clients:

Pour tester ce méthodes:

- Séparer les données en données d'entraînement et données test.
- Déterminer les meilleurs hyperparamètres de chaque méthode, grâce à la méthode de validation croisée.

Modèle de classification des clients: méthode retenue

	Accuracy
Bagging	0.998351
Random Forest	0.997113
Adaboost	0.933196
SVC	0.851546
KNeighbors	0.995052

Nous avons opté pour le modèle de forêt aléatoire. Il a l'avantage de ne pas overfitter les données d'entraînement.

Classifieur binaire de la finalisation de commande:

Pour définir le meilleur modèle de classification , permettant de prédire si le client va passer à l'achat. Nous avons testé les modèles suivants :

- Bagging
- Random forest
- Adaboost
- SVM
- KNeigboors
- Logistic

Classifieur binaire de la finalisation de commande: Méthode retenue

	Accuracy
Bagging	0.999175
RandomForest	0.999175
Adaboost	0.988041
SVC	0.915876
KNeighbors	0.992165
Logistic	0.934845

Nous avons opté pour le modèle de forêt aléatoire. Il a l'avantage de ne pas overfitter les données d'entraînement.

Application Python

Nous avons créé une application python, permettant de classer le client dès sa première opération d'achat. Elle se base sur:

- Une séquence temporelle client (introduit sous forme d'un fichier excel ou csv).
- La méthode de forêt aléatoire , avec les meilleurs hyperparamètres :

Nbre d'arbres de décision :	40
Nbre de variables	< 3

Conclusion:

- ★ Tout au long de ce travail, nous avons pu définir un modèle permettant de classer les clients en fonction de leur comportement temporelles d'achat dans 9 catégories dignes d'intérêt.

- ★ Nous avons pu définir un modèle permettant de prédire si le client va passer à l'achat.

- ★ Nous avons développé une application python, permettant de classer le client en fonction de leur historiques d'achat.