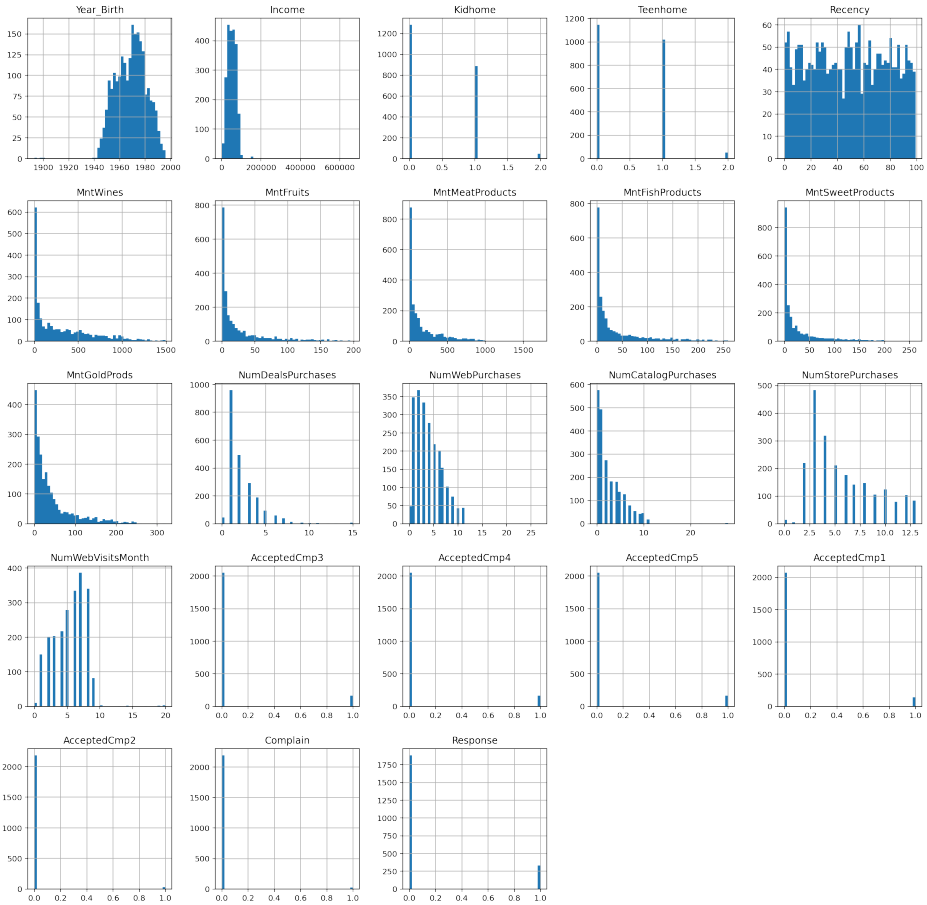
PTML: Projet Final

Moustapha Diop moustapha.diop@epita.fr
Martin POULARD martin.poulard@epita.fr
Mathieu Rivier mathieu.rivier@epita.fr
Théo Perinet theo.perinet@epita.fr
Marc Monteil marc.monteil@epita.fr

July 2022



Contents

1 Supervised learning

Les bibliothèques utilisées lors de cette partie sont les suivantes :

- Numpy : pour bénéficier des calculs vectorisés.
- Sklearn : contant tous les modèles de machine learning nécessaires à résolution de notre problème.
- Pandas : pour pouvoir manipuler avec efficacité des données tabulaires.
- Matplotlib et Seaborn : pour avoir des visualisations.

1.1 Choix du Dataset

Le but du dataset est de prédire la présence de maladies cardiaques chez un individu grâce à ses informations de santé. Le dataset utilisé pour les différents apprentissages est disponible à l'adresse suivante : `https://www.kaggle.com/datasets/fedesoriano/heart-failure-prediction`. Ce dataset comporte des critères numériques et catégoriques.

1.2 Exploration et nettoyage des donnés

Le dataset est composé de onze attributs dont cinq sont numériques. Ils décrivent dans l'ensemble l'état de santé du patient.

- Age: age of the patient [years].
- Sex: sex of the patient [M: Male, F: Female].
- ChestPainType: chest pain type [TA: Typical Angina, ATA: Atypical Angina, NAP: Non-Anginal Pain, ASY: Asymptomatic].
- RestingBP: resting blood pressure [mm Hg].
- Cholesterol: serum cholesterol [mm/dl].
- FastingBS: fasting blood sugar [1: if FastingBS $>$ 120 mg/dl, 0: otherwise]. : resting electrocardiogram results [Normal: Normal, ST: having ST-T wave abnormality (T wave inversions and/or ST elevation or depression of $>$ 0.05 mV), LVH: showing probable or definite left ventricular hypertrophy by Estes' criteria].
- MaxHR: maximum heart rate achieved [Numeric value between 60 and 202].
- ExerciseAngina: exercise-induced angina [Y: Yes, N: No].
- Oldpeak: oldpeak = ST [Numeric value measured in depression].
- ST_Slope: the slope of the peak exercise ST segment [Up: upsloping, Flat: flat, Down: downsloping].
- HeartDisease: output class [1: heart disease, 0: Normal].

Le dataset n'était pas entièrement complet. En effet, 172 mesures du taux de cholestérol sont manquantes. Nous avons tout d'abord pensé à utiliser la moyenne de cette mesure pour approcher les taux réels de cholestérol, cependant cette technique imputant trop sur la qualité des données, nous avons choisi de supprimer ces entrées du dataset. Nous avons également décidé après des expérimentations de supprimer des valeurs aberrantes dans les catégories 'Oldpeak' (>=4) et 'Cholesterol' (>=500), en effet, celle-ci sont non représentatives de la population et dégradent la qualité de l'apprentissage. Après ce traitement, l'échantillon d'entraînement présente les répartitions exprimées dans les histogrammes ci-dessous.

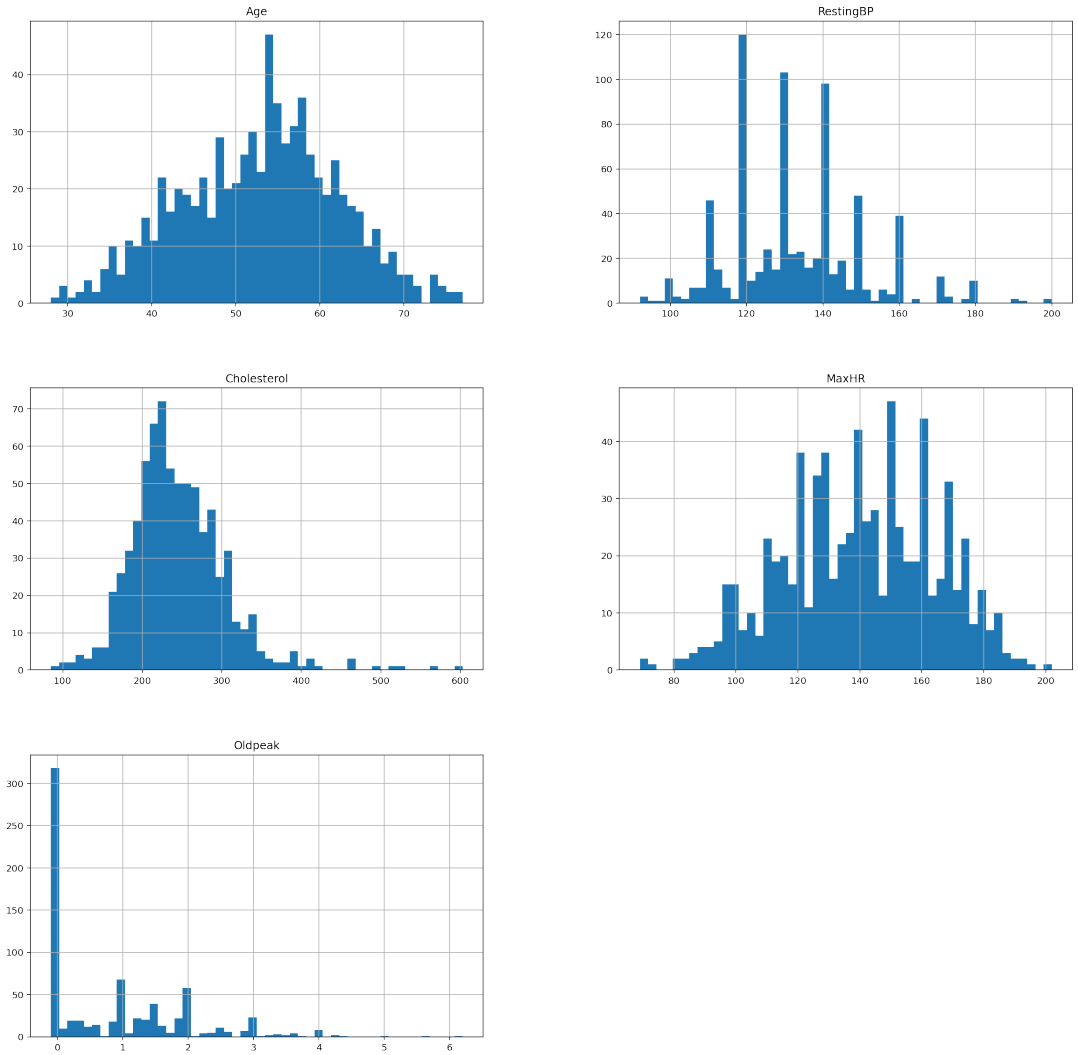


Figure 1: Histogramme des variables numériques

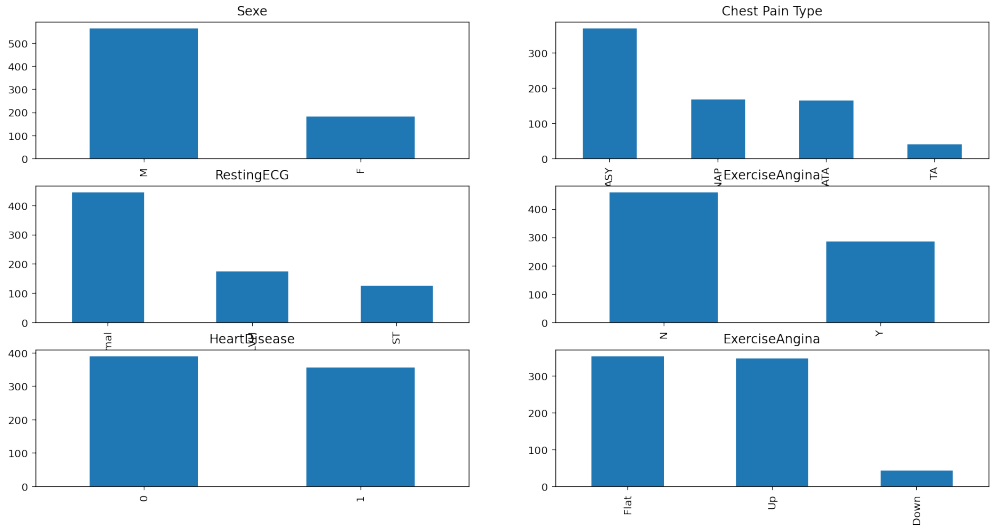


Figure 2: Histogramme des variables catégorielles

Les attributs ne présentent pas de corrélations notables, le minimum des coefficients de corrélation est de -0.38 et le maximum est 0.5.

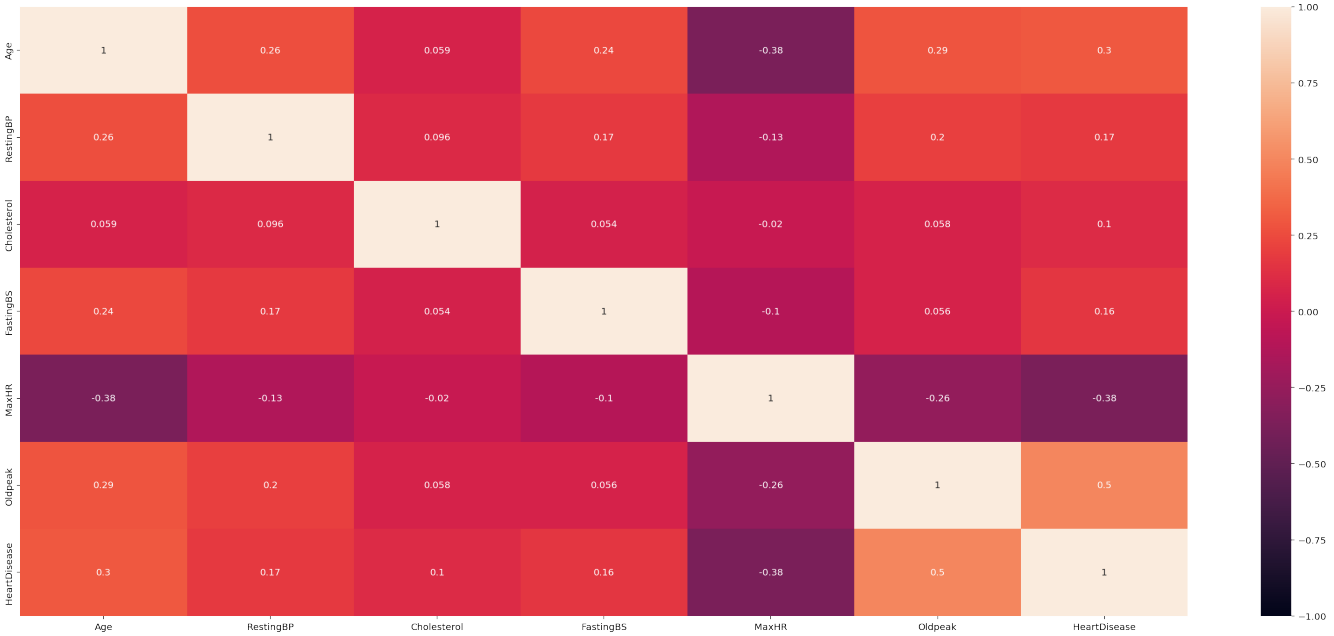


Figure 3: Matrice de corrélation

1.3 Prétraitement

Nous avons appliqué un "oneHotEncoder" sur les variables catégoriques, ainsi qu'une standardisation sur les variables numériques. Cela, pour pouvoir appliquer correctement les algorithmes d'apprentissage.

1.4 Modèles utilisés et réglage des hyperparamètres

Prédire la présence d'une maladie cardiaque est un problème de classification (malade/sain). Les techniques des K-Nearest Neighbors ainsi que celle de la régression logistique sont celles qui ont été utiliser en premier lieu pour construire des prédictions. Nous avons ensuite testé un modèle "plus exotique" : un multi-layer perceptron classifieur.

1.4.1 K-Nearest Neighbors

Afin de paramétrer au mieux cet algorithme, différentes valeurs du nombre de voisins ont été testés. Les valeurs les plus hautes sont 3 et 18.

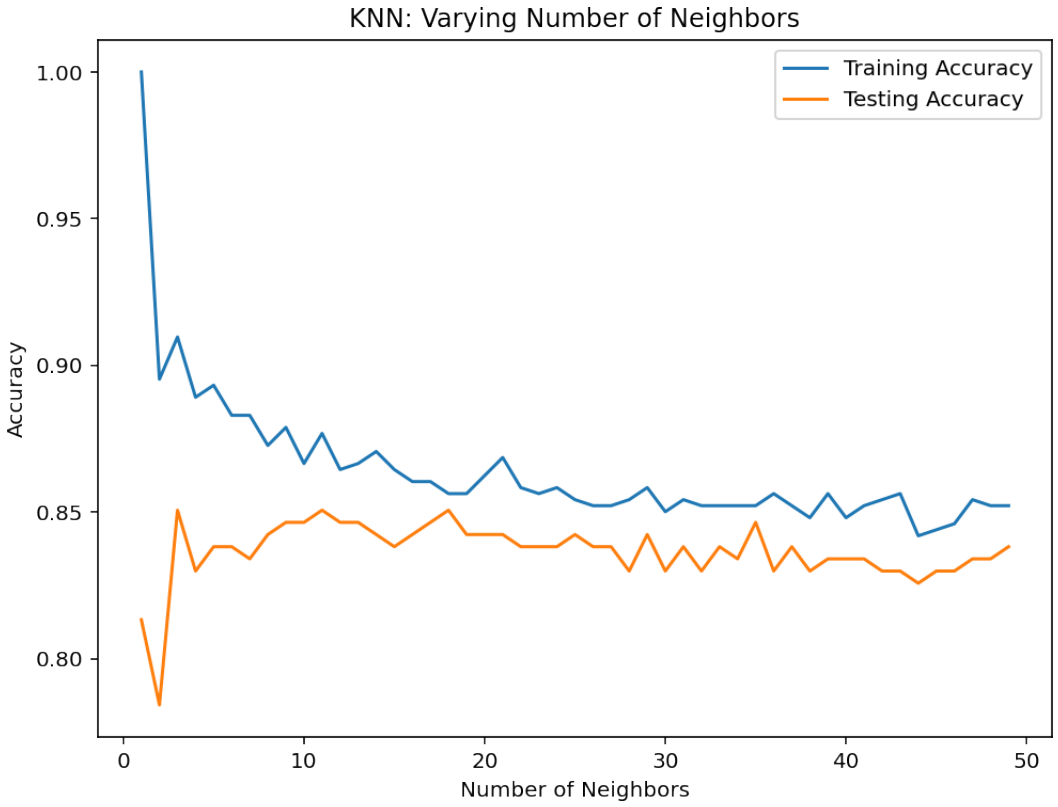


Figure 4: Valeurs d'accuracy en fonction du nombre de voisins

Avec les valeurs optimales de nombre de voisins, on obtient les scores suivants :

	precision	recall	f1-score	support
sain	0.83	0.91	0.87	128
malade	0.88	0.79	0.83	113
accuracy			0.85	241
macro avg	0.85	0.85	0.85	241
weighted avg	0.85	0.85	0.85	241

Table 1: métriques de KNN pour nb_neighbors=18

On peut cependant s'intéresser aux paramètres de l'ordre de la distance, ainsi qu'à la fonction poids utilisée dans les prédictions. Après plusieurs tentatives, nous sommes arrivés à obtenir de meilleurs résultats, avec un ordre de 1 pour le calcul de la distance et un poids inversement proportionnel à la distance.

	precision	recall	f1-score	support
sain	0.88	0.91	0.89	128
malade	0.89	0.86	0.87	113
accuracy			0.88	241
macro avg	0.88	0.88	0.88	241
weighted avg	0.88	0.85	0.88	241

Table 2: métriques de KNN pour nb$_n eighbors = 10, p = 1, weight =$ "$distance$"

1.4.2 régression logistique

Les résultats obtenus avec une régression logistique sont comparables à ceux obtenus avec KNN. Pour trouver les paramètres les mieux adaptés à notre problématique, nous avons utilisé un grid search qui isole les paramètres les plus performants en les testant tous.

	precision	recall	f1-score	support
sain	0.85	091	0.88	128
malade	0.89	0.82	0.86	113
accuracy			0.87	241
macro avg	0.87	0.87	0.87	241
weighted avg	0.87	0.87	0.87	241

Table 3: métriques de la régression logistique

On obtient la courbe ROC, celle-ci est utilisée pour mesurer la performance des classificateurs binaires.

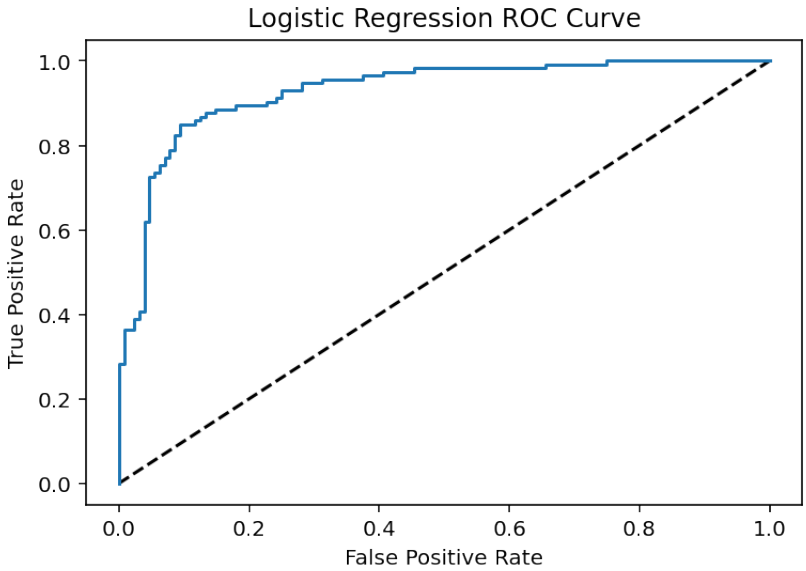


Figure 5: Courbe roc de la régression logistique

1.4.3 Multi-Layer Perceptron Classifier

Les MLP sont des algorithmes d'apprentissage automatique supervisés, ils apprennent (approximent) une fonction à partir du dataset. Contrairement à la régression logistique, les MLP comportent une ou plusieurs couches de neurones cachées, cela qui lui permet d'apprendre des modèles non linéaires. Cependant, les MLP présentent également des inconvénients, notamment les hyperparamètres. Ces derniers sont en effet difficiles à régler finement. Par ailleurs, si la fonction, représentant le modèle réel, n'est pas convexe, les descentes de gradient ne seront pas efficaces. Dans notre cas, après une simple exploration (non exhaustive et poussée) nous arrivons, malgré les défauts des MLP, nous arrivons à une test accuracy de 0.87.

	precision	recall	f1-score	support
sain	0.87	0.91	0.89	128
malade	0.90	0.83	0.86	113
accuracy			0.88	241
macro avg	0.88	0.87	0.87	241
weighted avg	0.88	0.88	0.88	241

Table 4: métriques de la régression logistique

2 Unpervised learning

Pour cette partie du projet, nous avons utilisé les bibliothèques suivantes :

- Numpy : pour bénéficier des calculs vectorisés
- Sklearn : contant tous les modèles de machine learning nécessaires à résolution de notre problème
- Pandas : pour pouvoir manipuler avec efficacité des données tabulaires
- Matplotlib et Seaborn : pour avoir des visualisations

2.1 Dataset choisi

Le jeu de donné utilisé est nommé "Customer Personality Analysis". Nous pouvons le télécharger depuis le site de Kaggle : `https://www.kaggle.com/datasets/imakash3011/customer-personality-analysis`. Ce dataset permet de mieux comprend la clientèle d'une entreprise et de s'adapter à celle-ci.

Il contient 2240 individus et 29 attributs découpés en 4 parties :

- People:
  - ID: Customer's unique identifier
  - Year_Birth: Customer's birth year
  - Education: Customer's education level
  - Marital_Status: Customer's marital status
  - Income: Customer's yearly household income
  - Kidhome: Number of children in customer's household
  - Teenhome: Number of teenagers in customer's household
  - Dt_Customer: Date of customer's enrollment with the company
  - Recency: Number of days since customer's last purchase
  - Complain: 1 if the customer complained in the last 2 years, 0 otherwise
- Products:
  - MntWines: Amount spent on wine in last 2 years
  - MntFruits: Amount spent on fruits in last 2 years
  - MntMeatProducts: Amount spent on meat in last 2 years
  - MntFishProducts: Amount spent on fish in last 2 years
  - MntSweetProducts: Amount spent on sweets in last 2 years
  - MntGoldProds: Amount spent on gold in last 2 years
- Promotion:

  - NumDealsPurchases: Number of purchases made with a discount
  - AcceptedCmp1: 1 if customer accepted the offer in the 1st campaign, 0 otherwise
  - AcceptedCmp2: 1 if customer accepted the offer in the 2nd campaign, 0 otherwise
  - AcceptedCmp3: 1 if customer accepted the offer in the 3rd campaign, 0 otherwise
  - AcceptedCmp4: 1 if customer accepted the offer in the 4th campaign, 0 otherwise
  - AcceptedCmp5: 1 if customer accepted the offer in the 5th campaign, 0 otherwise
  - Response: 1 if customer accepted the offer in the last campaign, 0 otherwise

- Place:
  - NumWebPurchases: Number of purchases made through the company's website
  - NumCatalogPurchases: Number of purchases made using a catalogue
  - NumStorePurchases: Number of purchases made directly in stores
  - NumWebVisitsMonth: Number of visits to company's website in the last month

2.2 Exploration

Lors de l'exploration des données, nous avons vu qu'il y avait 3 attributs catégoriels et 26 attributs numériques avec seulement 24 données manquantes que nous avons décidé de supprimer.

De plus, nous avons supprimé les attributs Z_CostContact, Z_Revenue, Dt_Customer et ID car ils ne sont pas utiles à récupérer pour décrire la clientèle.

Enfin, nous avons fait un histogramme pour chaque variable catégorielle et numérique:

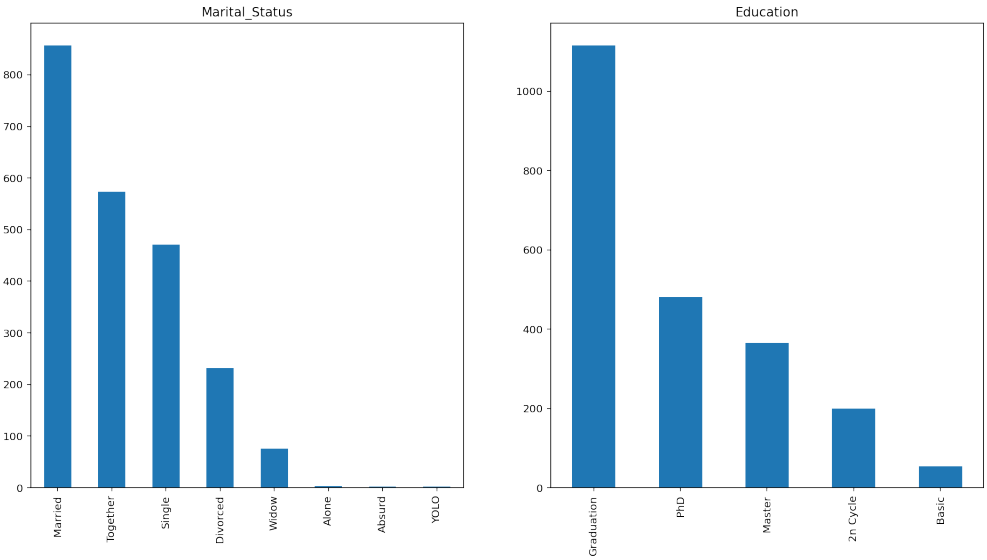


Figure 6: Histogramme des variables catégorielles

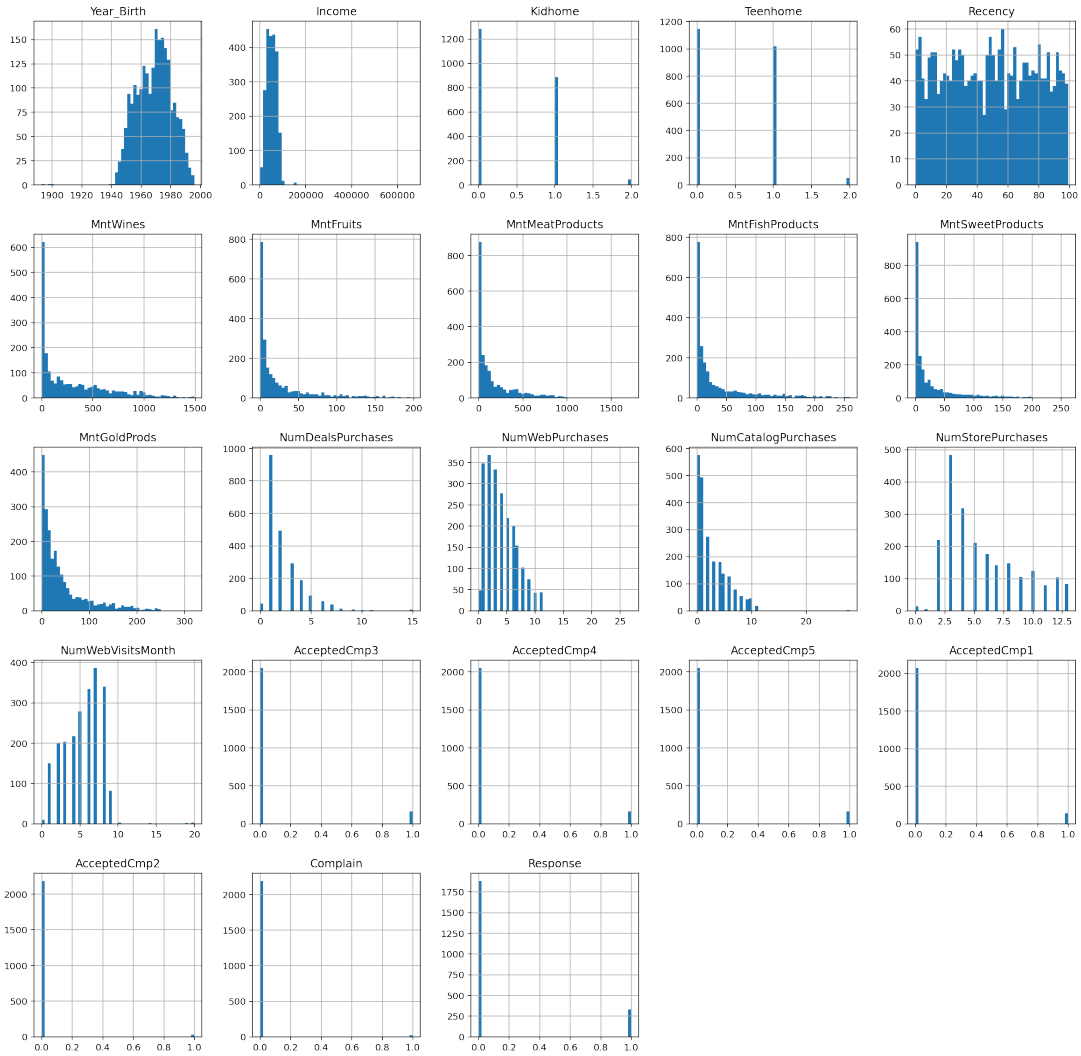


Figure 7: Histogramme des variables numériques

2.3 Features Engineering

Nous avons choisi de modifier et créer des colonnes afin d'obtenir plus amples informations sur les données. Cet apport pouvant être utilisé dans nos algorithmes par la suite afin d'avoir de meilleurs résultats.

Les modifications sont les suivantes :

- Création de la colonne Age en soustrayant la valeur dans Year_Birth à l'année 2021. 2021 étant l'année de la parution du jeu de donnée. La colonne Year_Birth est supprimée car devenue inutile.

- Modification de la colonne Education en gardant uniquement l'information si l'individu est diplômé ou non.

- Modification de la colonne Marital_Status en gardant uniquement l'information si l'individu est en couple ou célibataire.

- Création de la colonne Expenses contenant le montant de l'ensemble des dépenses (MntWines + MntFruits + MntMeatProducts + MntFishProducts + MntSweetProducts + MntGold-Prods).

- Création de la colonne Children contenant le nombre d'enfants (Kidhome + Teenhome).

2.4 Nettoyage des données

Nous avons remarqué la présence d'outliers dans les colonnes Age et Income.

En effet, certains clients ont plus de 100 ans ! Après analyse, nous avons décidé de supprimer les individus de plus de 85 ans.

De plus, certains clients ont des revenus très importants. Nous avons remarqué la présence d'un revenu suspect à 666 666 et d'un pic d'individus gagnant entre 150 000 et 160 000 pouvant être due à la présence d'un 1 en trop devant le revenu. Ainsi, nous avons choisi de supprimer les individus avec un revenu supérieur à 120 000.

Après le nettoyage des données, il reste finalement 2205 individus.

2.5 Corrélations entre attributs

Voici la matrice de corrélations entre attributs :

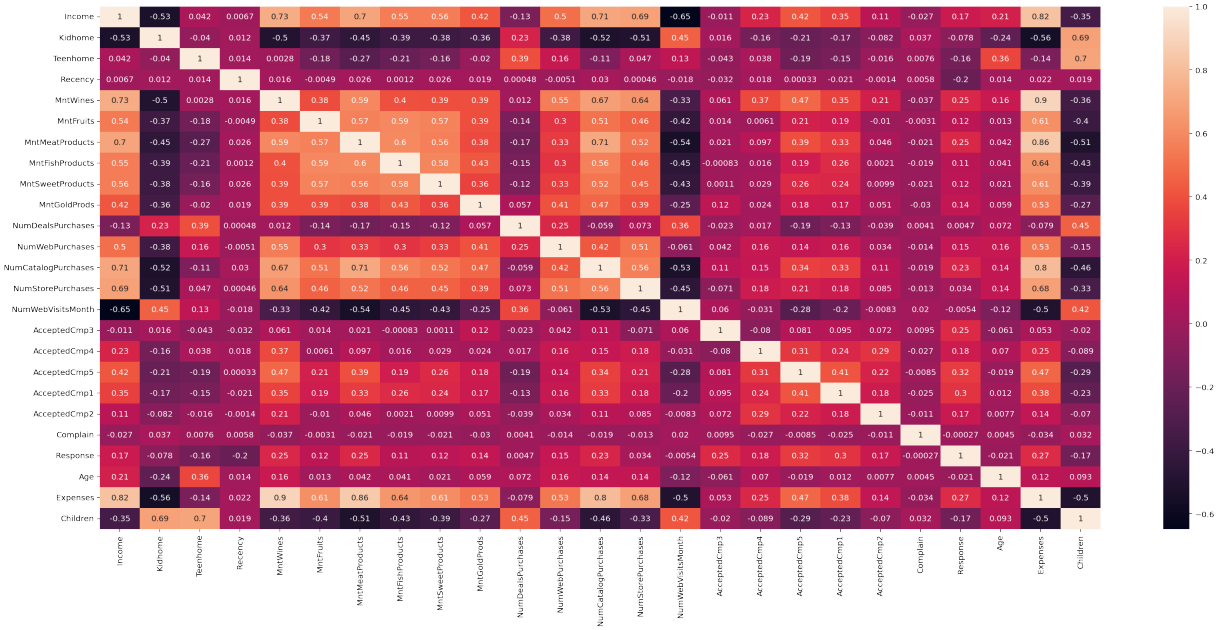


Figure 8: Matrice de corrélations entre attributs

Nous pouvons remarquer des corrélations évidentes dues au Features Engineering fait précédemment :

- Expenses avec MntWines, MntFruits, MntMeatProducts, MntFishProducts, MntSweetProducts et MntGoldProds.
- Children avec Kidhome et Teenhome.

Nous voyons également d'autres corrélations intéressantes/remarquables:

- L'achat de vin (MntWines) est corrélé au revenu (Income) et au montant des dépenses (Expenses) mais inversement corrélé au nombre d'enfants en bas âge (Kidhome). De plus l'achat de vin est, en comparaison aux autres types d'achats, corrélé aux campagnes de publicité (AcceptedCmpX), ce qui pourrait montrer une forte influence de ce type d'achat par la publicité.
- Le nombre d'enfants en bas âge (Kidhome) est inversement corrélé au montant des achats dans l'ensemble des catégories. Ce qui pourrait montrer la mauvaise rétention des familles avec enfants et ainsi indiquer que le magasin en question n'est pas pensée/adapter pour les familles avec des enfants.
- Le nombre d'achats via des réductions (NumDealsPurchases) est corrélé au nombre d'enfants (Children).
- Les dépenses (Expenses) sont fortement corrélées avec les revenus (Income).

2.6 Pré-traitement

Afin d'utiliser ces données dans nos algorithmes, les données numériques ont été normalisées et les données catégorielles transformées via un one hot encoder.

Par la suite, afin de diminuer la dimension des données, une pca a été effectuée sur les données.

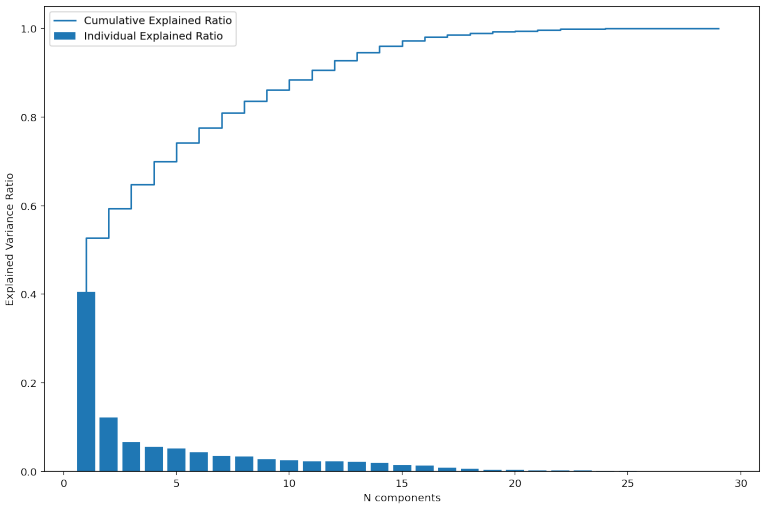


Figure 9: Pca explained variance

Nous remarquons que plus de 70% de la variance est expliquée en utilisant uniquement les 5 premières composantes de la PCA. Ainsi, nous allons utiliser uniquement ces 5 composantes par la suite.

2.7 Modèle utilisé et tuning

Nous avons décidé d'utiliser le modèle KMeans.

Pour choisir l'hyperparamètre indiquant le nombre de clusters à identifier par KMeans, nous avons regardé la valeur de l'inertie suivant ce nombre.

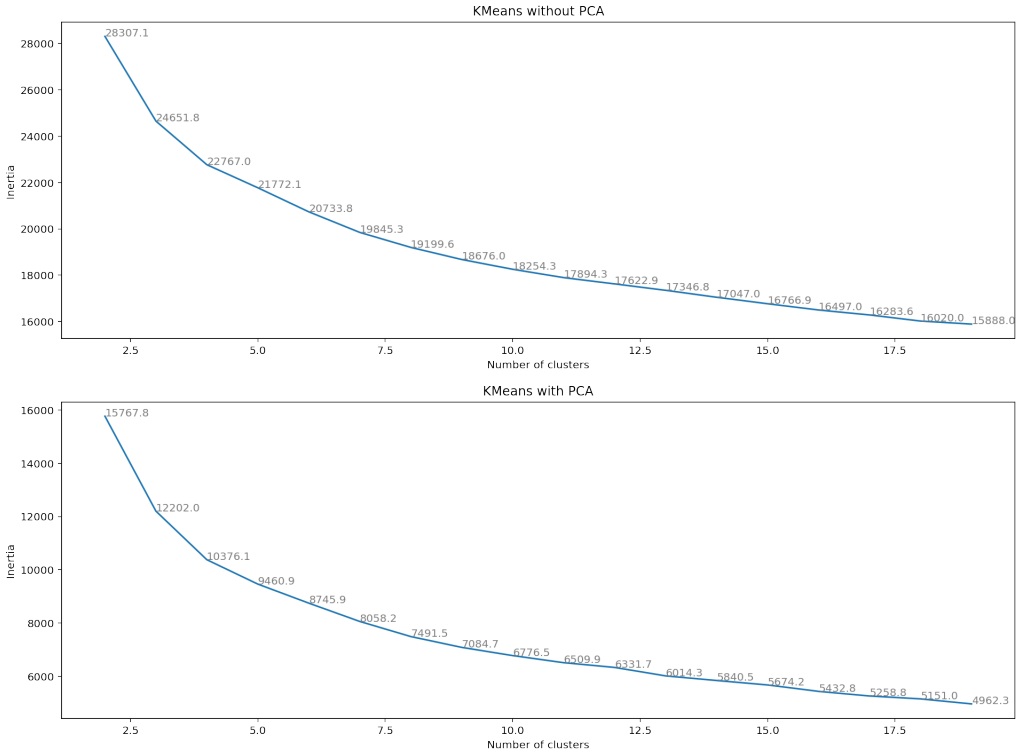


Figure 10: KMeans tuning the number of clusters

Nous pouvons voir que l'effet coude apparaît lorsque le nombre de clusters vaut 4. Ainsi, nous choisissons 4 comme valeur de l'hyperparamètre.

2.8 Résultats

Nous avons donc exécuté le KMeans avec l'hyperparamètre trouvé sur les données modifié par la pca (avec les 5 premières composantes).

Nous avons vérifié que la pca n'influençait pas de manière significative les résultats en regardant la corrélation entre la valeur des différents clusters trouvé par l'algorithme en utilisant ou non les données modifiés par la pca.

Le nombre d'individus dans chaque cluster est plutôt bien réparti :

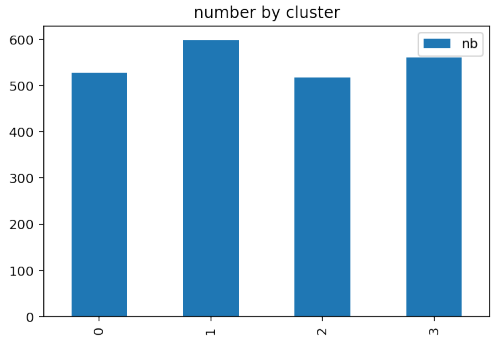


Figure 11: Number of individu by cluster

Voici la valeur moyenne des différents attributs dans chaque cluster :

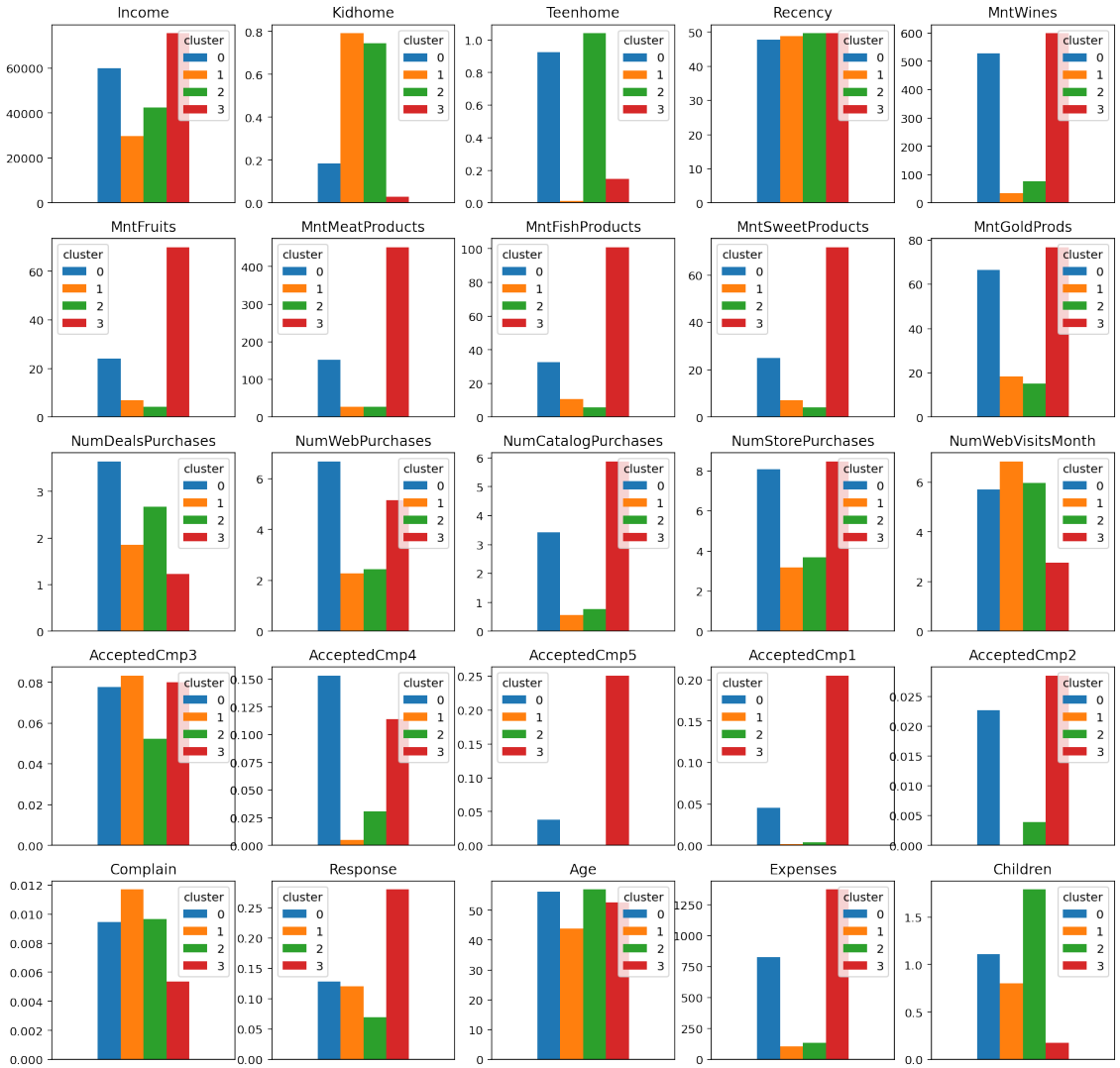


Figure 12: Mean value by attributs

Nous voyons que certains attributs sont peu pris en compte par KMeans (Recency, AcceptedCmp3 et Age) cependant certains attributs permettent de bien distinguer les différents clusters.

En regardant les différentes répartitions nous pouvons voir que le cluster :

- 0 représente les individus avec peu d'enfants en bas âge mais plusieurs adolescents. Il représente également les individus qui ont un fort revenu et qui consomment beaucoup. Ces individus sont moyennement sensibles à l'influence des publicités mais achètent beaucoup d'articles en promotions.

- 1 représente les individus avec des enfants en bas âge mais peu d'adolescents. Il représente également les individus qui ont un revenu bas et qui consomme peu. Ces individus ne sont pas sensibles à l'influence des publicités mais achètent les articles en promotions.

- 2 représente les individus avec des enfants en bas âge et plusieurs adolescents. Il représente également les individus qui ont un revenu moyen et qui consomme peu. Ces individus ne sont pas sensibles à l'influence des publicités mais achètent les articles en promotions.

- 3 représente les individus avec peu d'enfants en bas âge et peu d'adolescents. Il représente également les individus qui ont un fort revenu et qui consomment beaucoup. Ces individus sont sensibles à l'influence des publicités mais achètent peu d'articles en promotions.

Ainsi, nous remarquons que les clusters sont principalement déduits suivant le nombre d'enfants/adolescents, le revenu, le montant des dépenses et l'influence par les publicités.

En regardant la figure précédente, nous remarquons que les clusters 1 et 2 sont les clusters dépensant beaucoup moins que les autres mais ce sont eux qui achètent le plus (en proportion) de produit en promotion. Cela peut montrer une action déjà effective du magasin à attirer cette tranche de clientèle via les promotions.

Voici quelque couple d'attributs où nous voyons bien les différents groupes :

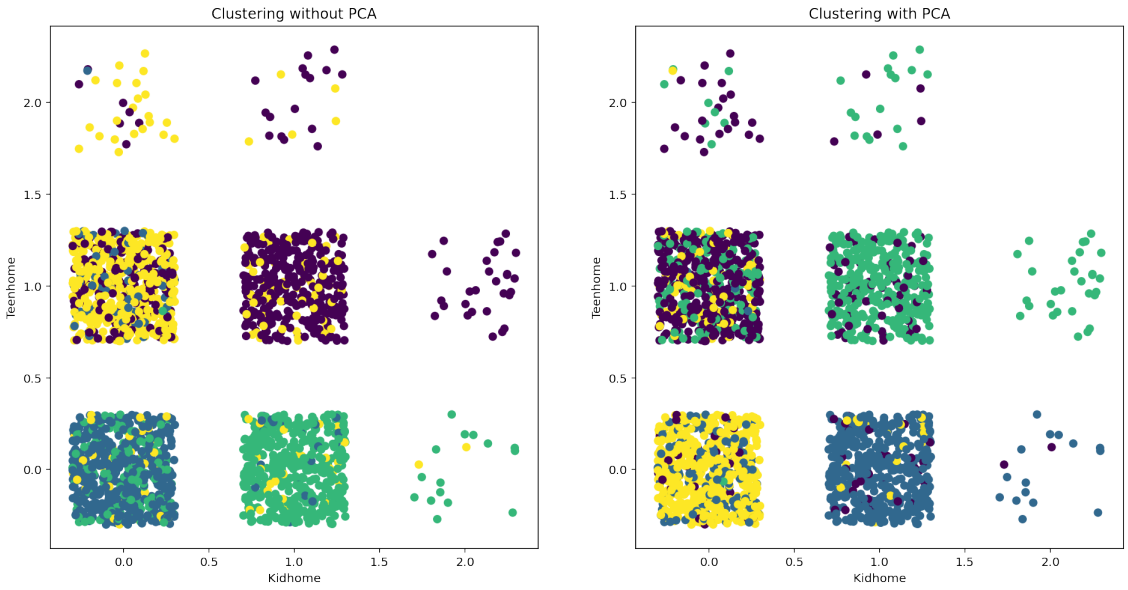


Figure 13: Clusters values depending on Teenhome and Kidhome attributs

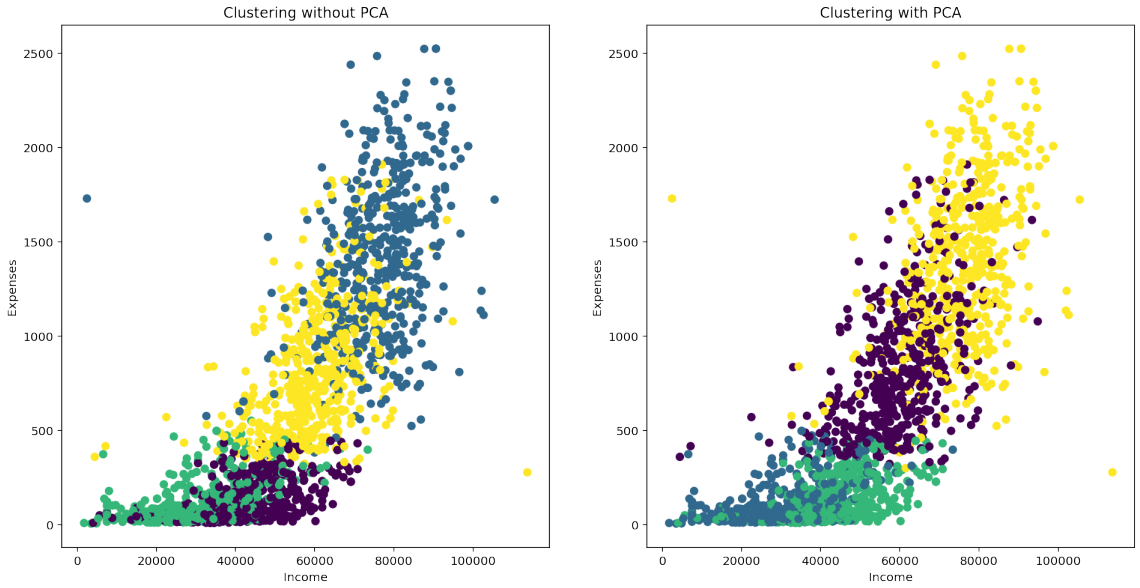


Figure 14: Clusters values depending on Expenses and Income attributs

2.9 Conclusion

Grâce à l'analyse des données, nous avons pu tirer beaucoup d'informations intéressantes sur les clients du magasin. Cela pouvant permettre d'adapter l'offre et la publicité du magasin afin de renforcer la clientèle habituelle (personnes sans enfants) ou attirer/fidéliser un nouveau segment de clients (familles avec des enfants).