Projet   
Machine Learning

Telco

short line**4 DataScience 2**

Réalisé par:

**BELAIDI Nada**

**CHARAABI Khaled**

**LAKHAL Abir**

**MAGHREBI Racha**

**KHECHANA Fares**

**Table de matière**

[**Compréhension du problème métier**](#_wj249xa64hfo) **2**

[1-problématique:](#_61q0w4ufj2dz) 2

[2-Objectif:](#_gy5mfzex9e0w) 2

[**Compréhension des données**](#_rorudpeomw2q) **2**

[**III-Préparation des données**](#_uqtd0acr3ecl) **4**

[1-nettoyage des données](#_txxew7mgdlxz) 4

[2-Feature Selection:](#_2d7h9mp62ocv) 6

[A- Corrélation:](#_3j1oy6j504b9) 6

[B-manipulation des données :fonction get\_dummies:](#_zbmgref1wx6c) 6

[C-SFS:](#_66sx60ofxr6j) 7

[C-1-Feature selection SFS (AUC)](#_p3tyj1v6wqpu) 8

[C-2-Feature selection SFS (Accuracy)](#_xns9x65d9z8k) 8

[D-SBS:](#_5qhw5q2jgcpn) 8

[D-1-Feature selection SBS (AUC)](#_zh7xs6opq8xn) 9

[D-2-Feature selection SBS (Accuracy)](#_ot1kv02vl33x) 9

[E-SFFS:](#_lnizk6z4v6r2) 9

[E-1-Feature selection SFFS(Auc)](#_94ft8zsdoly2) 9

[E-2-Feature selection SFFS(Accuracy)](#_yvcgugepvoho) 10

[F-SBFS:](#_7ipt1pbk6l4m) 10

[F-1-Feature selection SBFS(Auc)](#_n28t5a87lg78) 10

[F-2-Feature selection SBFS(Accuracy)](#_v5wzj7ajh8hf) 10

[3- transformation des données:](#_g85jz2jnh4ns) 11

[4-Réduction des dimensionnalité:](#_8qx02210927n) 11

[**III-Modelisation**](#_deiyhf4w4gjr) **13**

[1-K Nearest Neighbors](#_73ylibnktg0f) 13

[Les résultats du modèle KNN :](#_h8cfyky1hypl) 14

[2-Arbre de décision :CART](#_q7jg1yat8tpe) 14

[Les résultats du modèle CART :](#_ahf8jln5s2mx) 17

[3-Random Forest:](#_qc3qa7qa3ff2) 18

[Les résultats du modèle Random Forest:](#_bopfvbg54rx6) 19

[4- Machine à vecteurs de support SVM:](#_1z6ljt1xw3x7) 20

[Les résultats du modèle SVM:](#_flmj9x4rppf6) 21

[5- Naive Bayes:](#_vi4hhxtbmn6s) 21

[Les résultats du modèle Naive Bayes:](#_h9sjxkywcesp) 22

[6-Régression logistique:](#_fvqpzxpc5bop) 23

[Les résultats du modèle Régression logistique :](#_weco954hkq3m) 23

[**VI-Evaluation:**](#_pesa48d8fktl) **24**

# Compréhension du problème métier

## 1-problématique:

Il va sans dire que l’un des problèmes majeurs auquel et souvent confronter n’importe quelle entreprise est celui de satisfaire sa clientèle pour pouvoir la maintenir. pour se faire, elle doit prévoir une stratégie commerciale efficace et rentable

Une telle stratégie doit tenir compte des prévisions des taux de désabonnement et parer à une telle éventualité, du coup la facilité de changer d’opérateur est l’un des défis majeurs auxquels elle doit faire face. Pour fidéliser les clients existants, les organisations doivent améliorer le service-client, hausser la qualité des produits et être en mesure de savoir à l’avance quels clients susceptibles de se désabonner. En prévoyant le taux de désabonnement des clients, les entreprises peuvent immédiatement prendre des mesures pour les fidéliser.

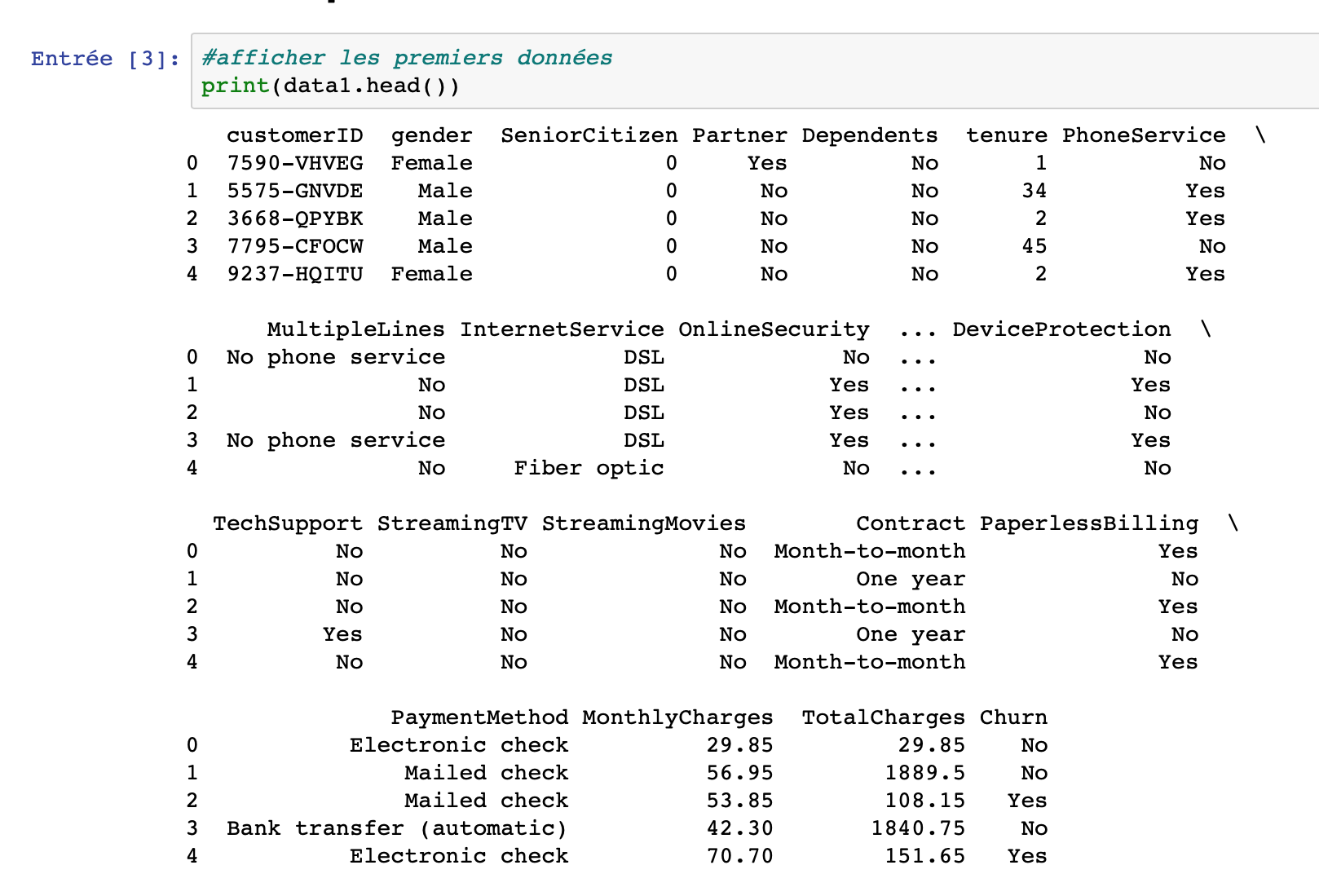
La prédiction peut être effectuée en analysant les données des clients à l'aide de techniques d'exploration.

## 2-Objectif:

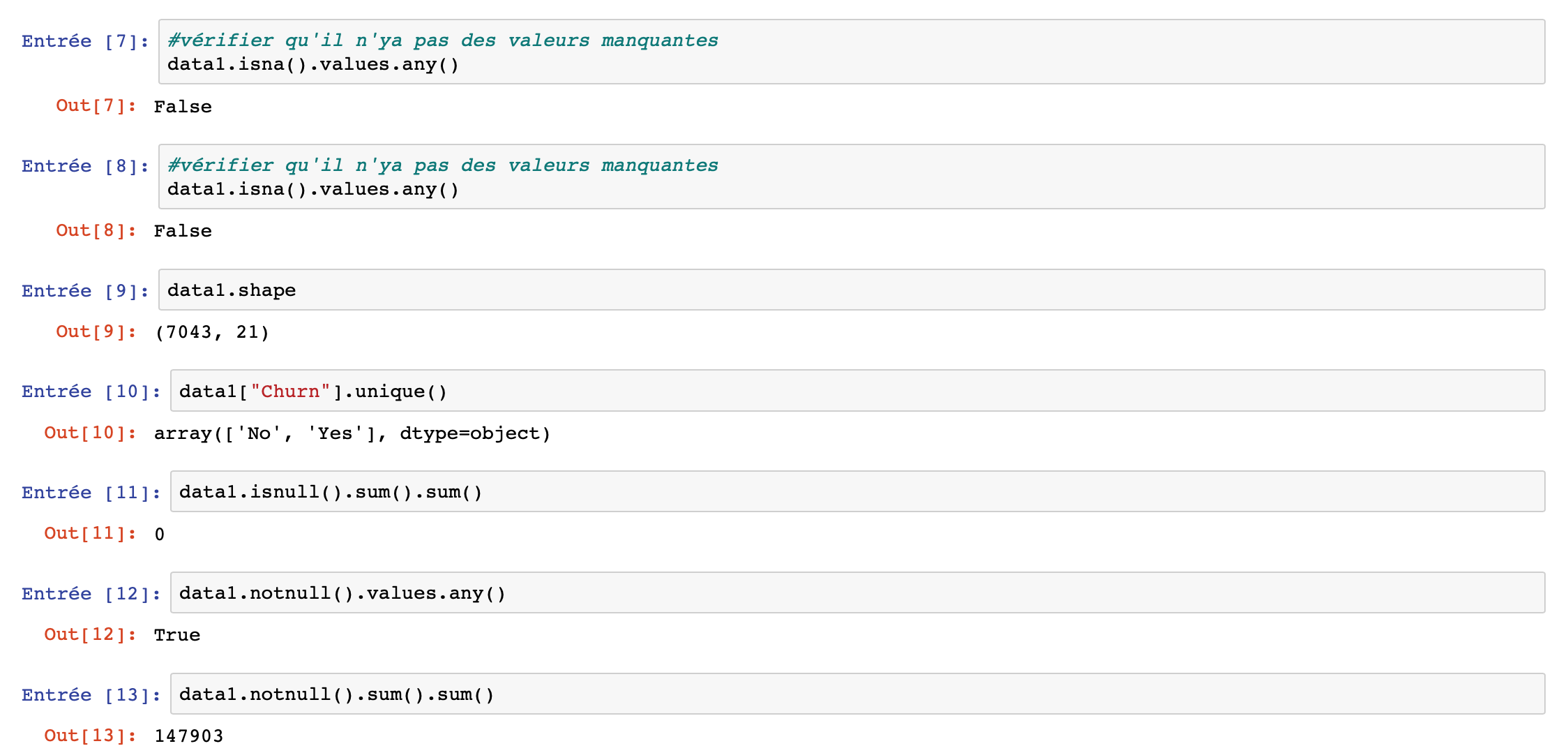
Au cours de notre projet, notre objectif est de tester et trouver l’algorithme d’apprentissage adéquat pour résoudre un problème de machine learning qui consiste à vérifier si le client peut se désabonner de son opérateur Telco ou non.

# Compréhension des données

L'ensemble de données que nous étudions dans ce projet est le taux de désabonnement des clients Telco. Cet ensemble de données est publié en ligne avec un accès public à des fins d'apprentissage.

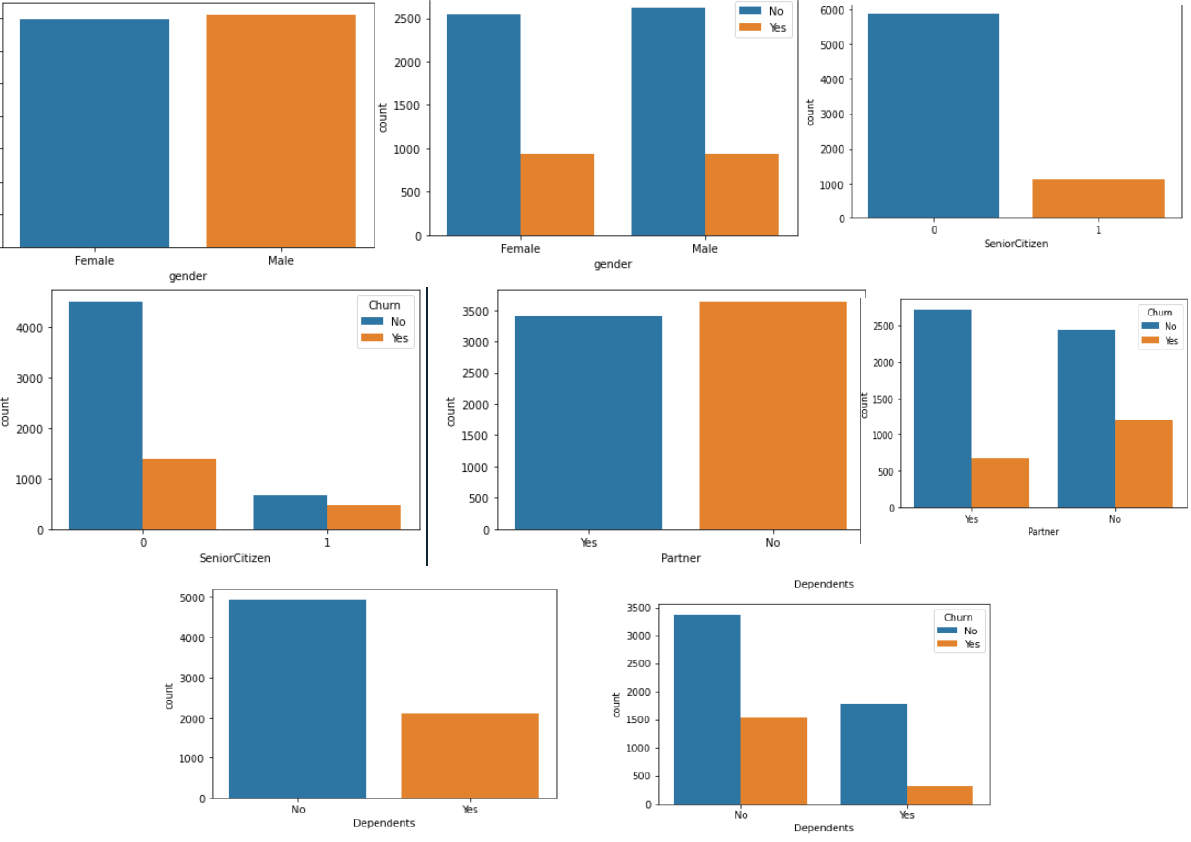


On peut visualiser ici les 5 premières lignes de notre DataFrame.



Notre dataset contient 21 attributs avec 7043 lignes.

Ici ,on a testé l’existence des valeurs Nan .Pas de valeurs manquantes ou bien nulles.



Ces représentations graphiques montrent les relations entre le “Churn” et le reste des fonctionnalités .

Répartition des fonctionnalités avec des valeurs continues:

- Les nouveaux clients dont l'abonnement dure moins de 10 mois ont une tendance plus élevée à l'attrition.

- Total des charges ≈ charges mensuelles \* mois d'occupation. Puisque nous avons observé que les churners ont des mois d'ancienneté bas mais des charges mensuelles élevées, leurs charges totales sont égales au total des charges pour les non-churners puisqu'ils ont des mois d'ancienneté élevés mais des charges mensuelles faibles (proportions inversées) résultant de l'absence d'argument final.

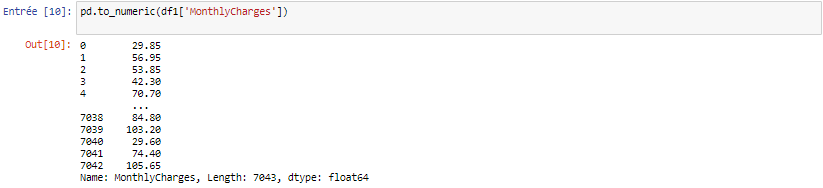
# III-Préparation des données

## 1-nettoyage des données



Pas de valeurs manquantes ou bien nulles, on a pas besoin d’effectuer l’imputation.







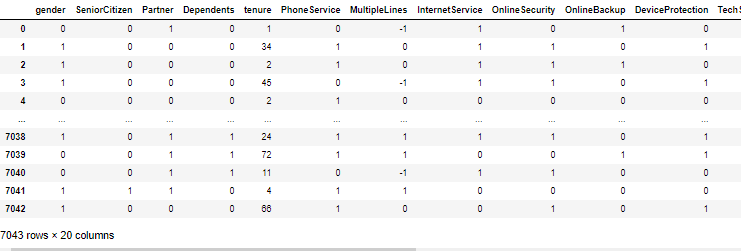
Ici, on a changé les colonnes non numériques en d'autres à valeurs permettant d’effectuer l’ACP.

La valeur 489 ème de la colonne ‘TotalCharges” est un espace qui ne peut pas être transformé en une valeur numérique, on la remplace par 0.

******

******

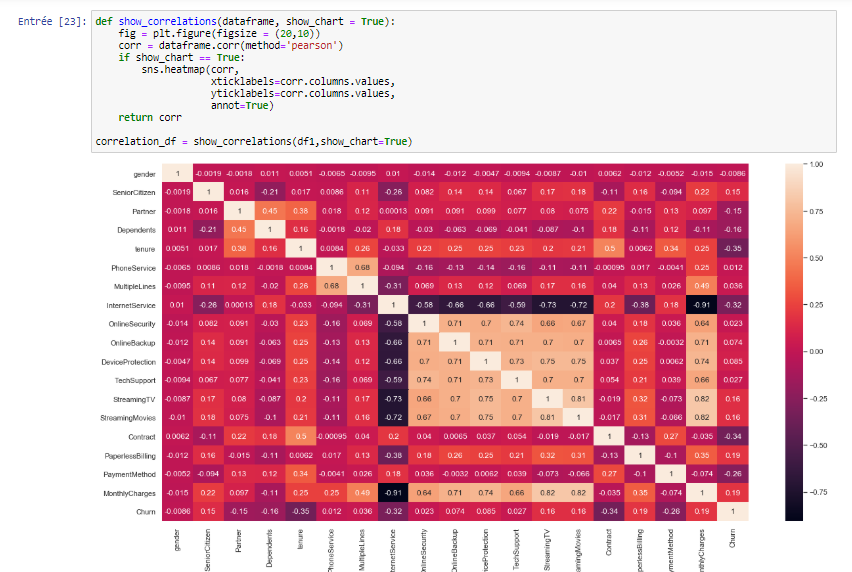
On se retrouve avec cette dataframe:



## 2-Feature Selection:

### **A**- Corrélation:

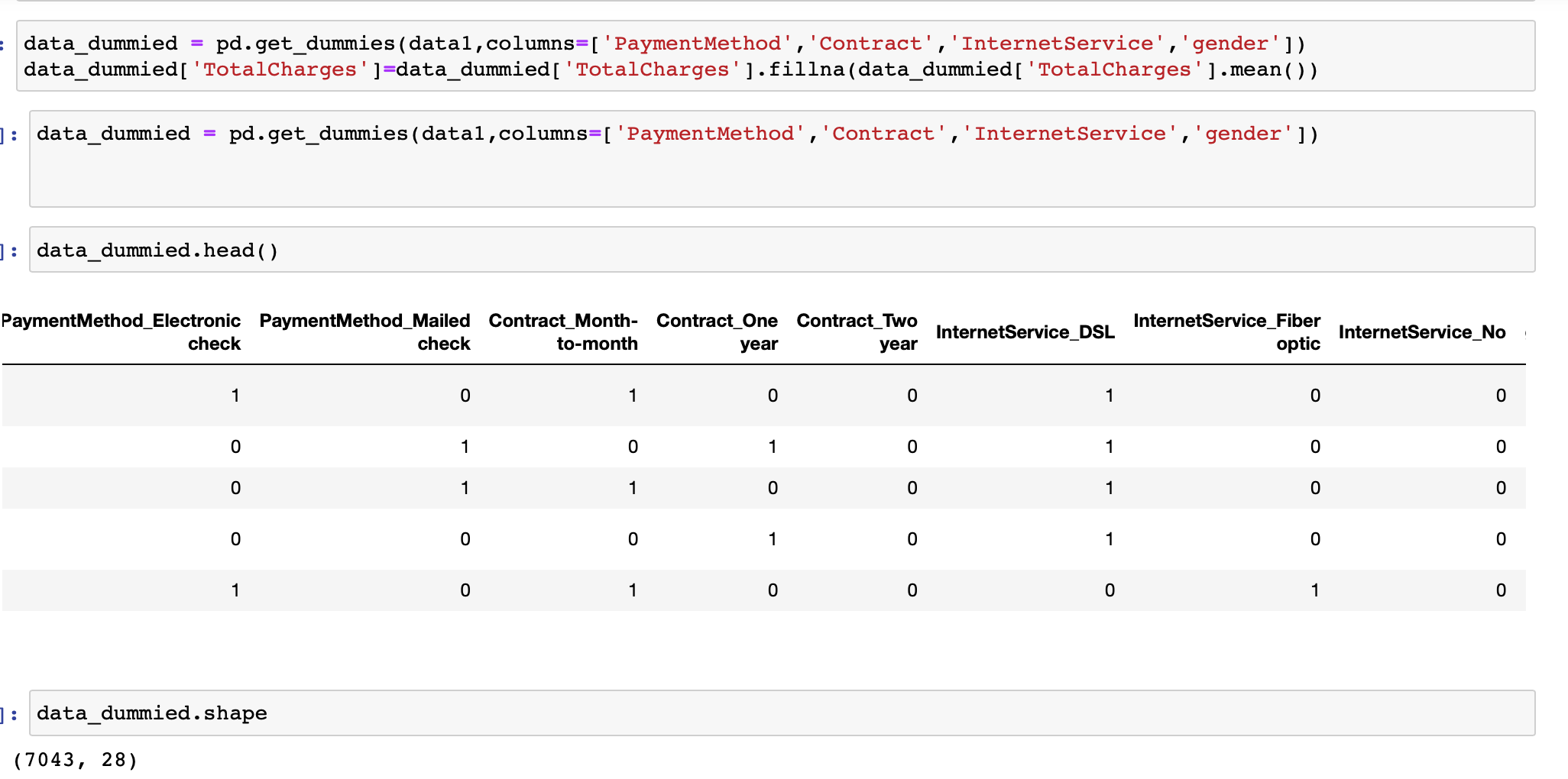
On a opté pour une méthode de sélection basée sur la corrélation comme on nous l'a demandé dans le document du projet et on a affiché notre heat map pour bien voir nos résultats.



On remarque que la corrélation maximale qu’on a obtenue est 0.19.Donc, on a choisi d'éliminer les corrélations négatives.

### **B**-manipulation des données :fonction get\_dummies:

La fonction get\_dummies () est utilisée pour convertir des variables catégorielles en variables factices / indicateurs



Ici, on a opté à manipuler les données avec la fonction get\_dummies où nous a donné un résultat de division de notre dataset sur 28 attributs

### **C**-SFS:

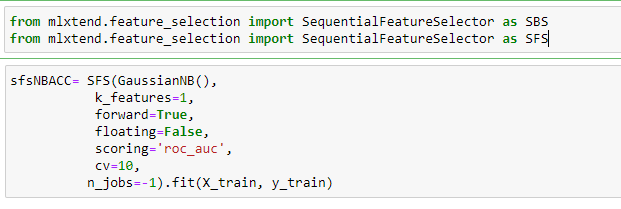
Tout d'abord, la meilleure caractéristique unique est sélectionnée (c'est-à-dire en utilisant une fonction de critère).

Ensuite, des paires d'entités sont formées en utilisant l'une des caractéristiques restantes et cette meilleure caractéristique, et la meilleure paire est sélectionnée.

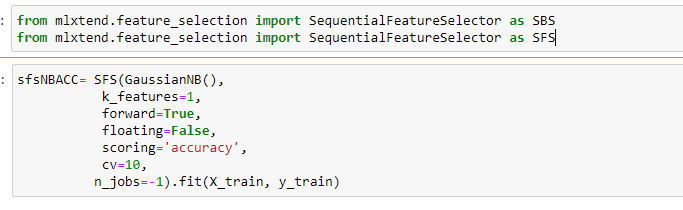
Ensuite, des triplets de caractéristiques sont formés en utilisant l'une des caractéristiques restantes et ces deux meilleures caractéristiques, et le meilleur triplet est sélectionné.

Cette procédure se poursuit jusqu'à ce qu'un nombre prédéfini d'entités soit sélectionné.

#### C-1-Feature selection SFS (AUC)



#### C-2-Feature selection SFS (Accuracy)



### **D**-SBS:

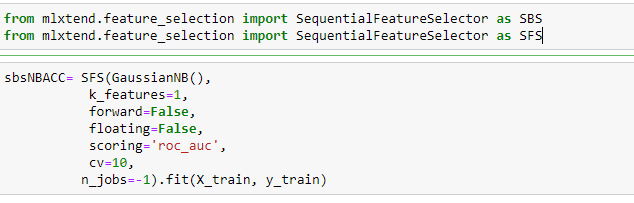
Premièrement, la fonction de critère est calculée pour toutes les n entités.

Ensuite, chaque caractéristique est supprimée une à la fois, la fonction de critère est calculée pour tous les sous-ensembles avec n-1 caractéristiques, et la pire caractéristique est rejetée.

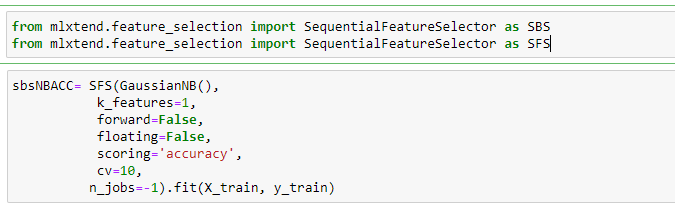
Ensuite, chaque caractéristique parmi les n-1 restants est supprimée une par une, et la pire caractéristique est rejetée pour former un sous-ensemble avec n-2 caractéristiques.

Cette procédure se poursuit jusqu'à ce qu'il reste un nombre prédéfini d'entités.

#### D-1-Feature selection SBS (AUC)



#### D-2-Feature selection SBS (Accuracy)



### **E**-SFFS:

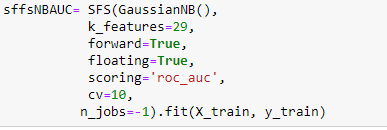
-La sélection avant flottante séquentielle (SFFS) commence à partir de

l'ensemble vide.

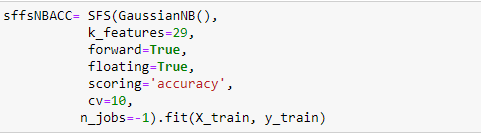
-Après chaque pas en avant, SFFS effectue des pas en arrière

tant que la fonction objectif augmente.

#### E-1-Feature selection SFFS(Auc)



#### E-2-Feature selection SFFS(Accuracy)



### **F**-SBFS:

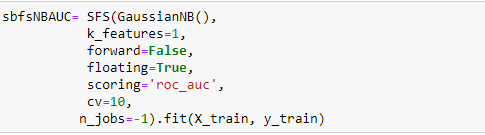
-La sélection arrière flottante séquentielle (SFBS) démarre

de l'ensemble complet.

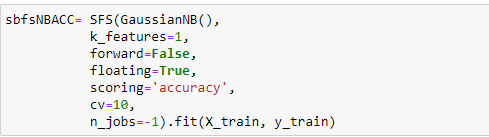
-Après chaque pas en arrière, SFBS effectue des pas en avant

tant que la fonction objectif augmente.

#### F-1-Feature selection SBFS(Auc)



#### F-2-Feature selection SBFS(Accuracy)

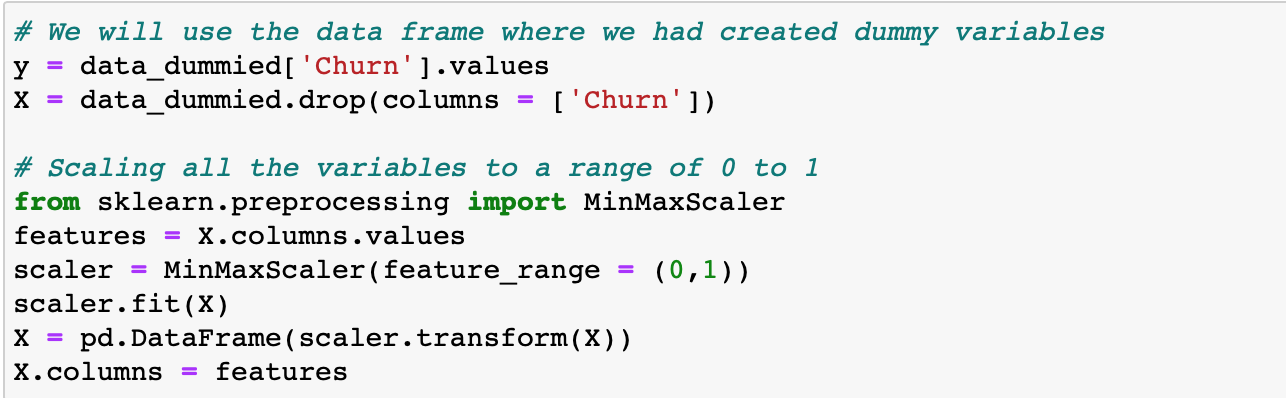




## 3- transformation des données:

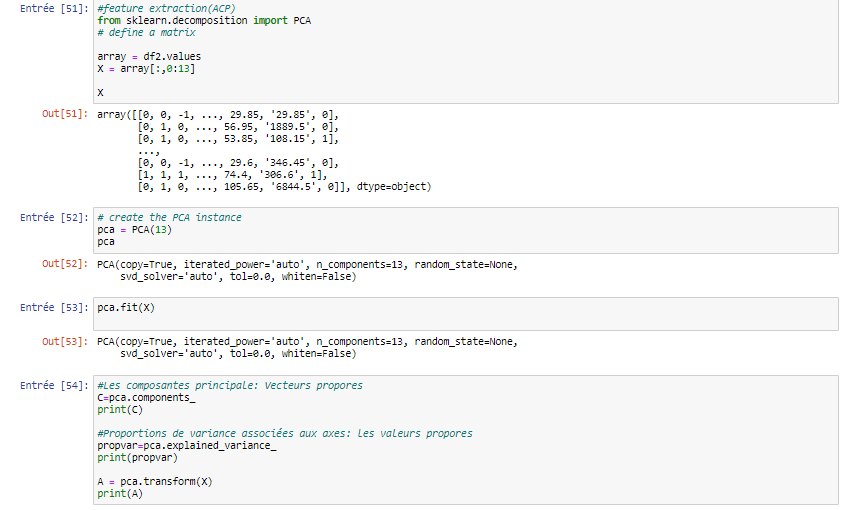
#### Normalisation

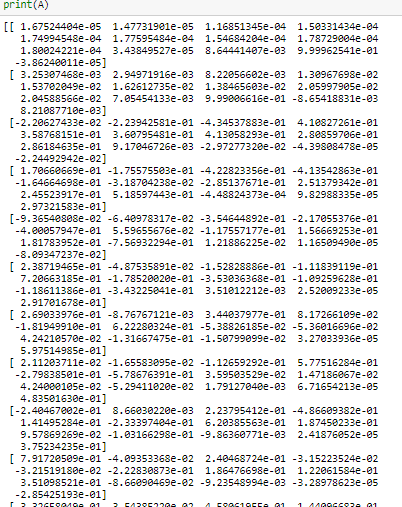
- avec MinMaxScaler:



## 4-Réduction des dimensionnalité:

On passe maintenant à l'application de l’ACP:



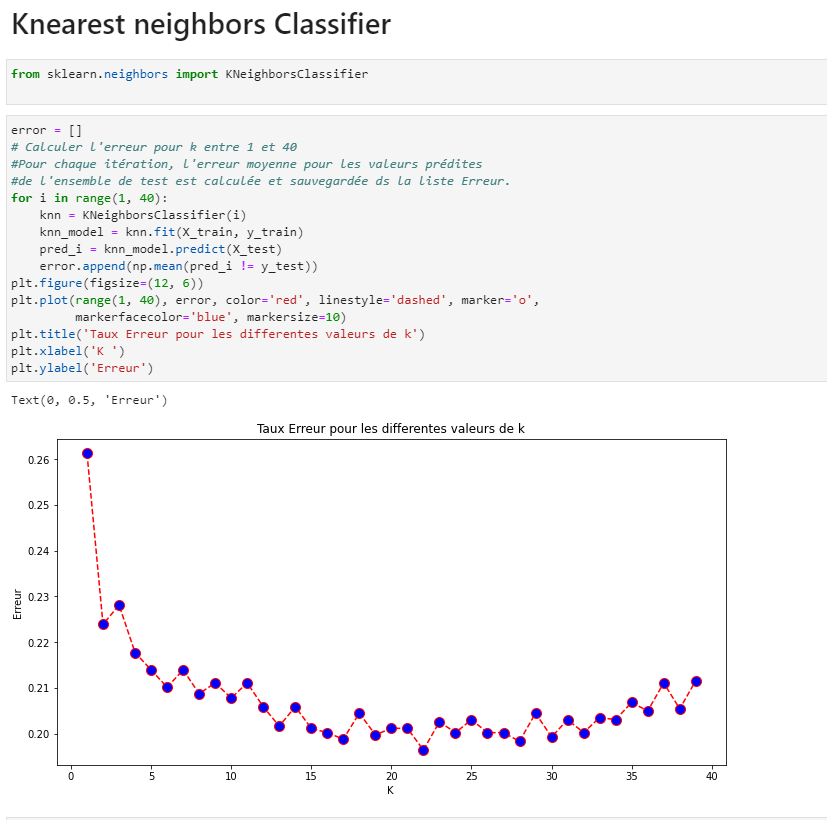
Et voici une partie de ce qu’on a obtenu:

# III-Modelisation

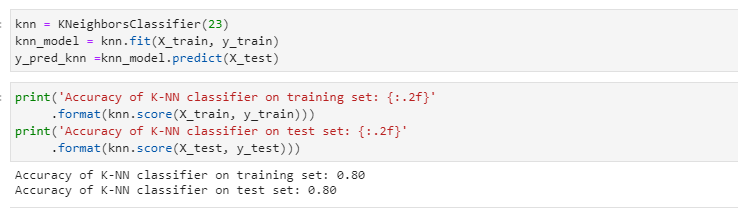
## 1-K Nearest Neighbors

En appliquant le KNN avec une plage allant de 1 vers 40 On a obtenu comme suit le graph donnant le taux d’erreur pour les différentes valeurs de K.

On a choisi la valeur de K la plus petite, c.à.d ayant le taux d’erreur le plus petit d'où on a choisit **K=23.**



On passant maintenant notre train et test data frames on a obtenu comme suit des résultats de **80% accuracy** pour le **training set** et **80% accuracy** pour le t**est set**



### Les résultats du modèle KNN :

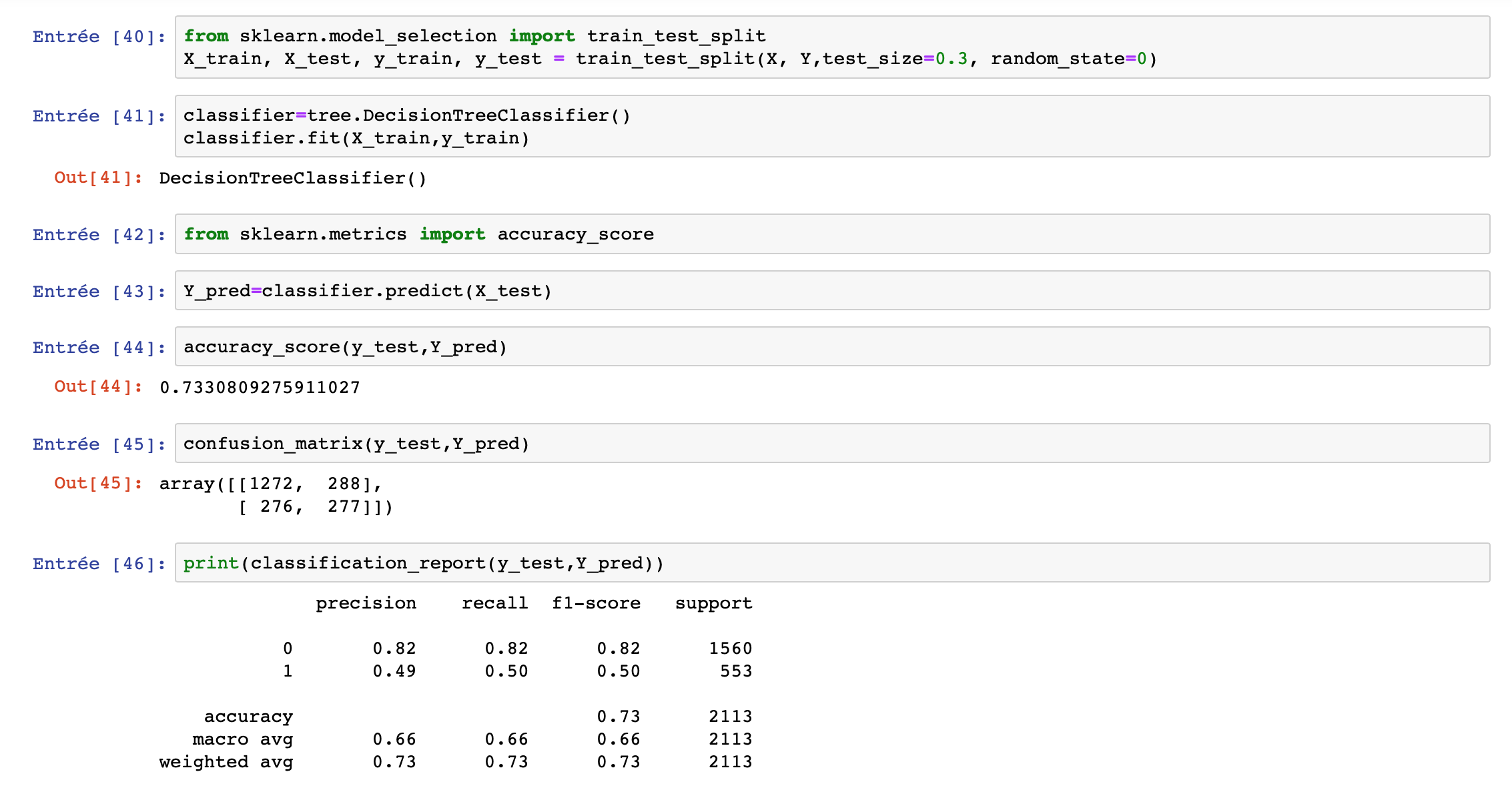
|  |  |  |  |
| --- | --- | --- | --- |
| Feature selection | training | testing | score |
| correlation | 0.80 | 0.79 | 0.79 |
| get\_dummies | 0.79 | 0.79 | 0.78 |
| SFS (AUC) | 0.80 | 0.81 | 0.8054898248935163 |
| SFS (Accuracy) | 0.80 | 0.81 | 0.8054898248935163 |
| SBS (AUC) | 0.79 | 0.81 | 0.8078561287269286 |
| SBS (Accuracy) | 0.79 | 0.81 | 0.8054898248935163 |
| SFFS (AUC) | 0.80 | 0.81 | 0.8054898248935163 |
| SFFS (Accuracy) | 0.80 | 0.81 | 0.8054898248935163 |
| SBFS (AUC) | 0.80 | 0.81 | 0.8054898248935163 |
| SBFS (Accuracy) | 0.80 | 0.81 | 0.8054898248935163 |

### 

## 2-Arbre de décision :CART

On a divisé l’ensemble des observations X et l’ensemble de classe Y, chacun en deux sous-ensemble :

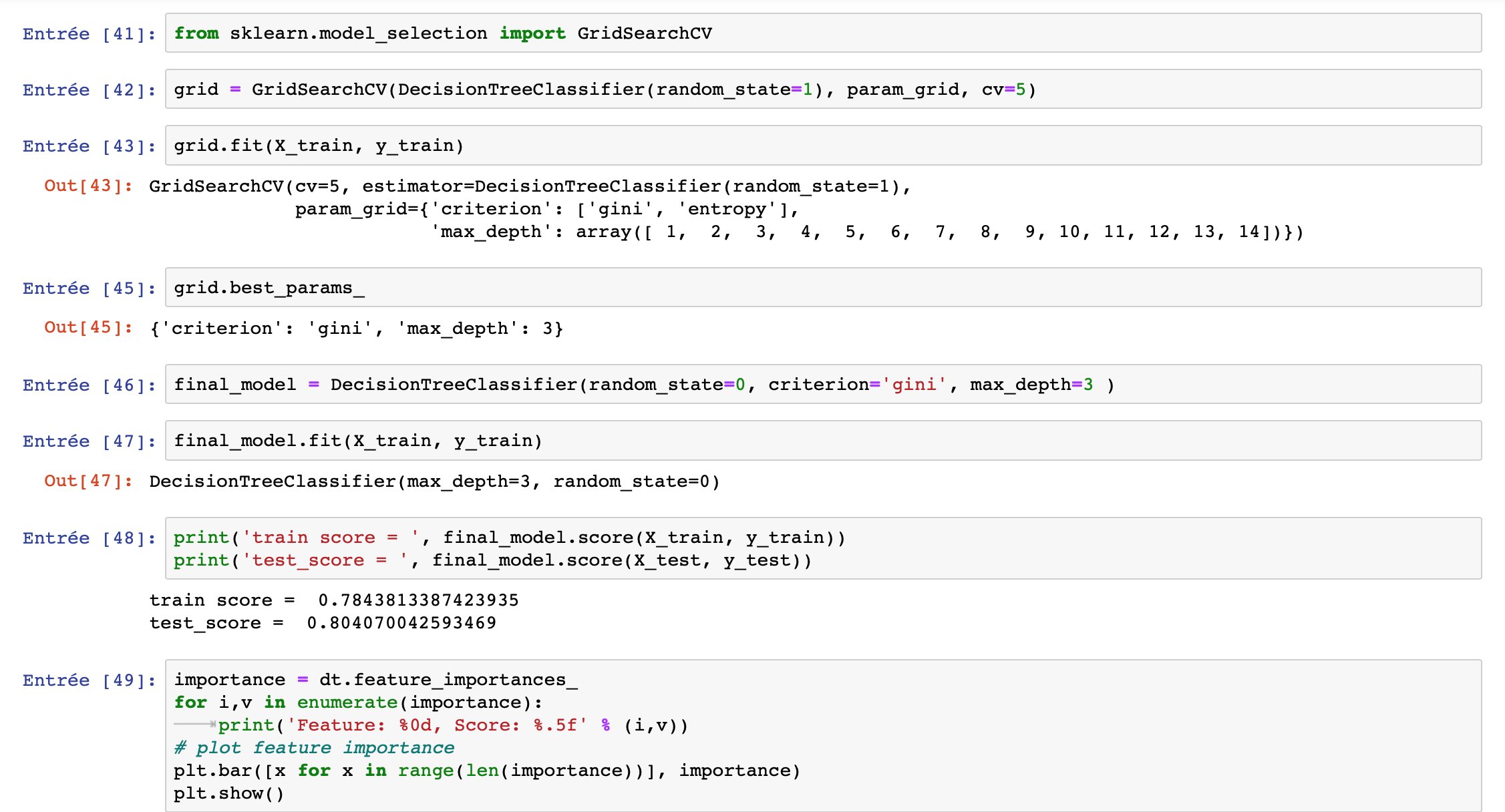
* un sous-ensemble d'apprentissage : 70% de l'ensemble initial
* un sous-ensemble de test : 30% de l'ensemble initial

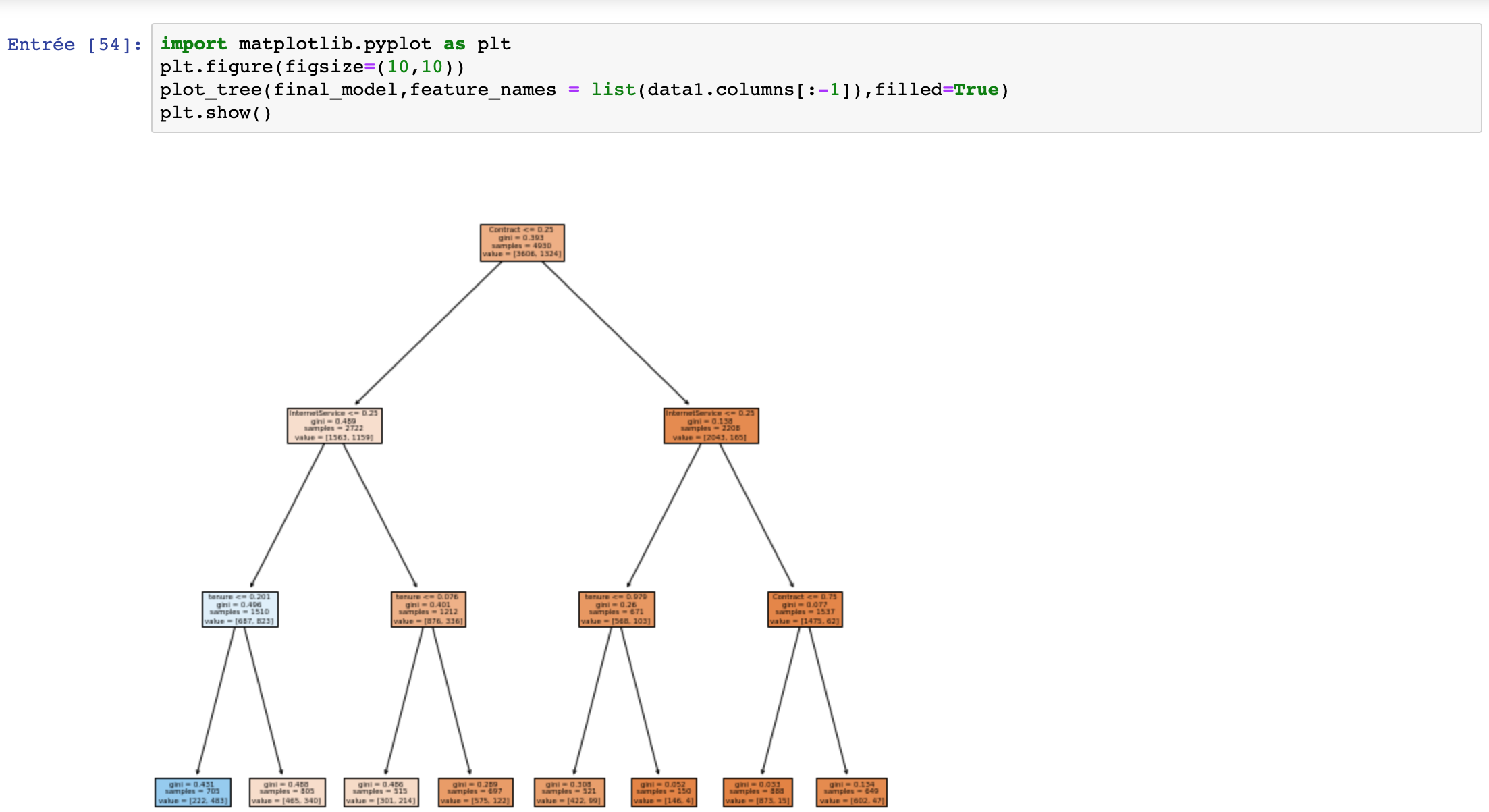


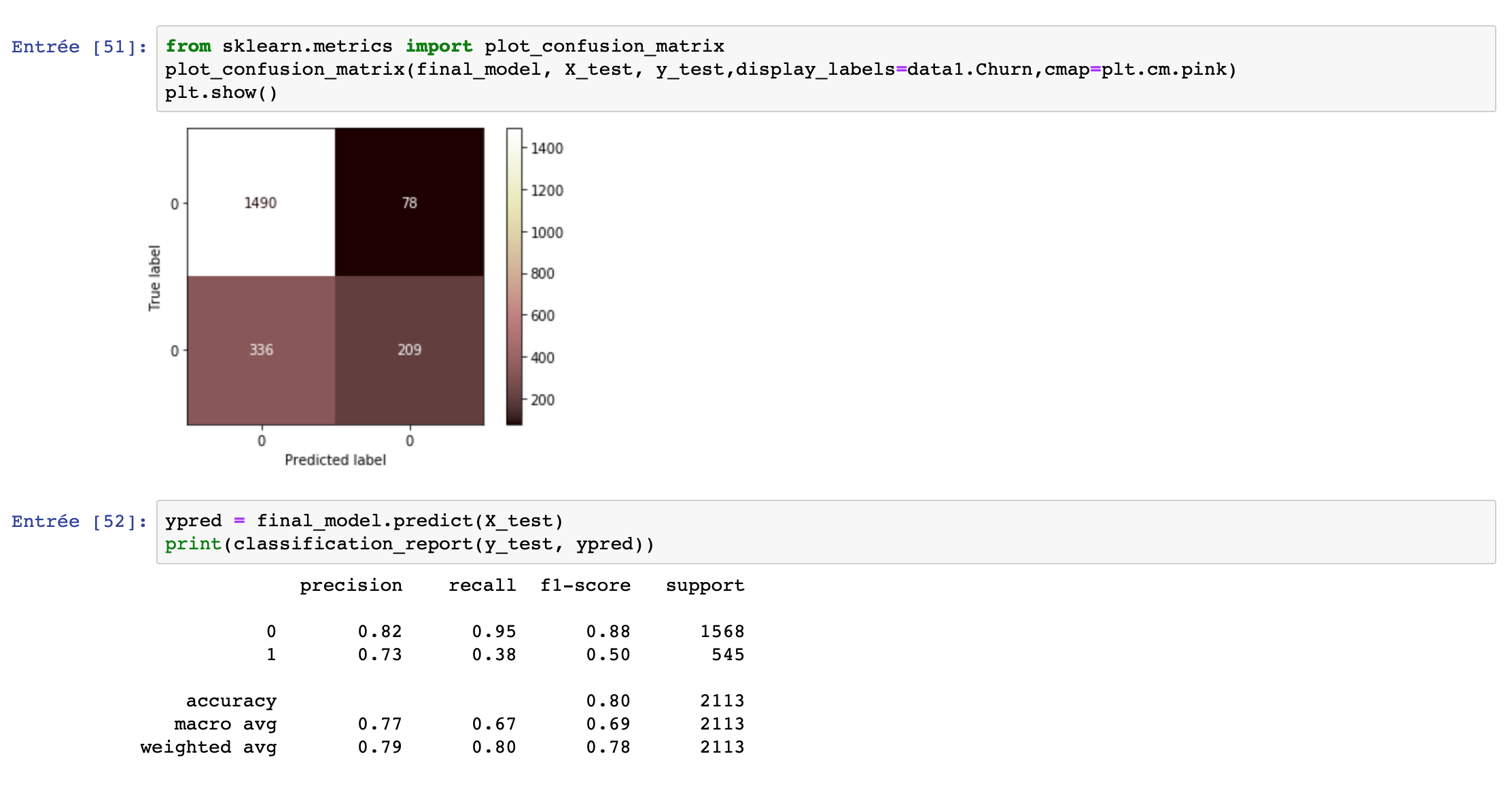
l'application de l'arbre de décision a donnée une performance plus ou moins bonne 0.73 , concernant les précisions on remarque une très bonne prédiction pour les personnes qui ne vont se désabonner (0.82) , par contre il y a une baisse de précision pour les personnes qui vont se désabonner 0.49 (à améliorer )

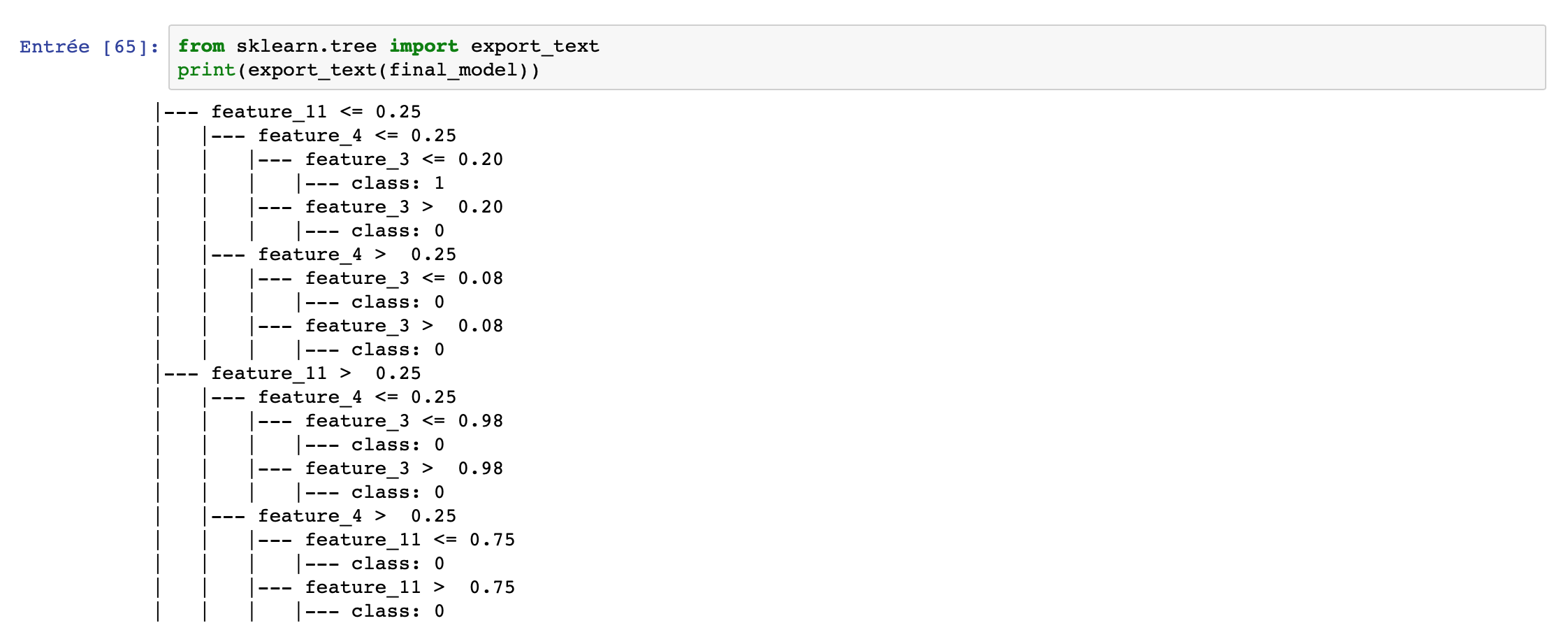
L'arbre est trop long et difficile à interpréter. Pour faire face à ce problème on va essayer de régler les hyperparamètre de notre modèle

#### 









Après application de Search Grid CV qui nous a aidé à déterminer les meilleures valeurs des hyperparamètres réglés et après évaluation de la performance de final\_model sur le sous-ensemble de données approprié on a trouvé 0.82 pour les personnes qui ne vont se désabonner et 0.73 pour les personnes qui vont se désabonner avec une performance de 0.80

|  |  |
| --- | --- |
| train\_score | test\_score |
| 78% | 80% |

### 

### Les résultats du modèle CART :

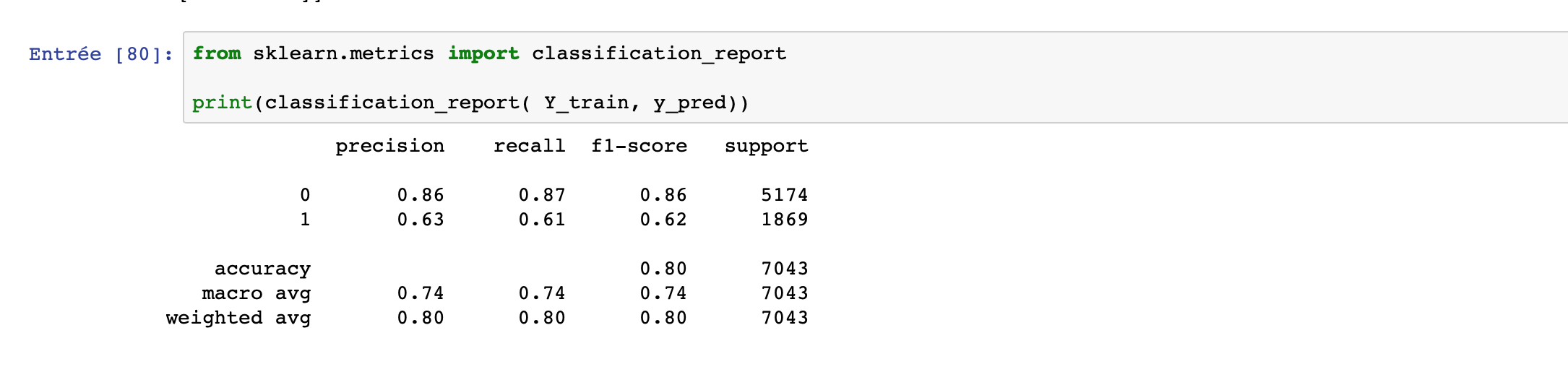
### 

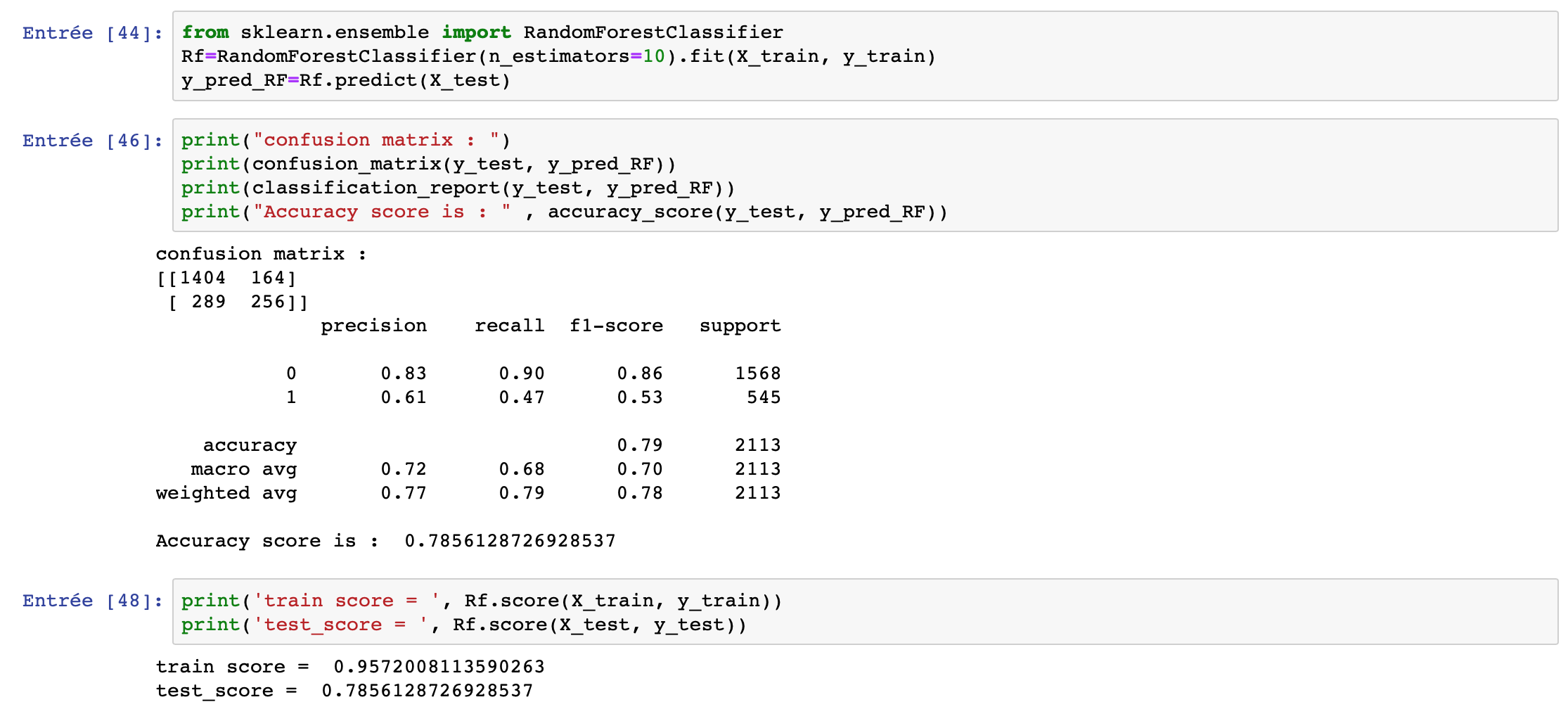
|  |  |  |  |
| --- | --- | --- | --- |
| Feature selection | training | testing | score |
| correlation | 0.79 | 0.77 | 0.77 |
| dummies | 0.78 | 0.79 | 0.80 |
| SFS (AUC) | 0.7845841784989858 | 0.8035967818267865 | 0.8035967818267865 |
| SFS (Accuracy) | 0.7845841784989858 | 0.8035967818267865 | 0.8035967818267865 |
| SBS (AUC) | 0.7845841784989858 | 0.8035967818267865 | 0.8035967818267865 |
| SBS (Accuracy) | 0.7845841784989858 | 0.8035967818267865 | 0.8035967818267865 |
| SFFS (AUC) | 0.7845841784989858 | 0.8035967818267865 | 0.8035967818267865 |
| SFFS (Accuracy) | 0.7845841784989858 | 0.8035967818267865 | 0.8035967818267865 |
| SBFS (AUC) | 0.7845841784989858 | 0.8035967818267865 | 0.8035967818267865 |
| SBFS (Accuracy) | 0.7845841784989858 | 0.8035967818267865 | 0.8035967818267865 |

### 

## 3-Random Forest:







Après traitement de l’algorithme Random Forest, il nous donné une performance de 0.78 , concernant les précisions on remarque une très bonne prédiction pour les personnes qui ne vont se désabonner (0.83) , pour les personnes qui vont se désabonner 0.61

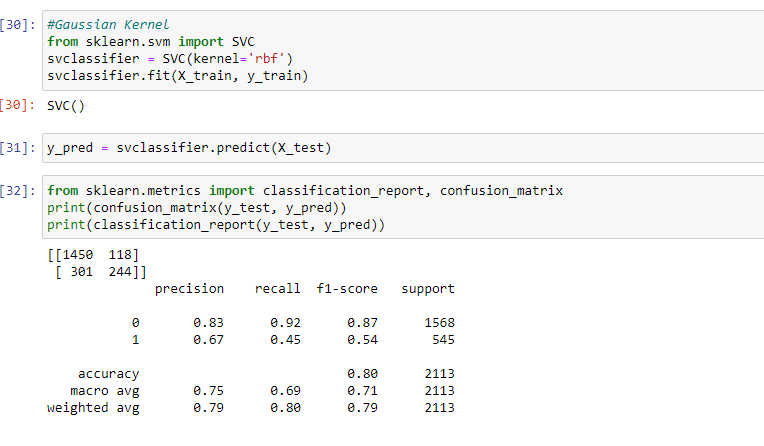
### 

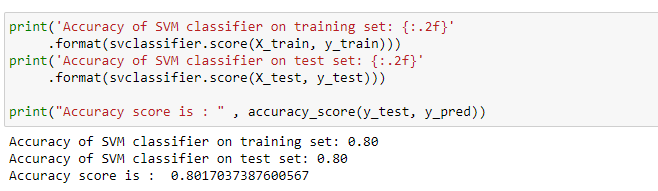
### Les résultats du modèle Random Forest:

|  |  |  |  |
| --- | --- | --- | --- |
| Feature selection | training | testing | score |
| correlation | 0.95 | 0.77 | 0.78 |
| dummies | 0.80 | 0.67 | 0.80 |
| SFS (AUC) | 0.9494929006085193 | 0.7723615712257453 | 0.7723615712257453 |
| SFS (Accuracy) | 0.9466531440162271 | 0.7742546142924751 | 0.7665314401622718 |
| SBS (AUC) | 0.949290060851927 | 0.7723615712257453 | 0.7723615712257453 |
| SBS (Accuracy) | 0.9501014198782961 | 0.780407004259347 | 0.780407004259347 |
| SFFS (AUC) | 0.9486815415821501 | 0.7756743965925225 | 0.7756743965925225 |
| SFFS (Accuracy) | 0.945841784989858 | 0.7714150496923805 | 0.7714150496923805 |
| SBFS (AUC) | 0.9466531440162271 | 0.7756743965925225 | 0.7756743965925225 |
| SBFS (Accuracy) | 0.9498985801217038 | 0.77520113582584 | 0.77520113582584 |

### 

## 4- Machine à vecteurs de support SVM:





l’algorithme Random SVM, il nous donné une performance de 0.80 , concernant les précisions on remarque une très bonne prédiction pour les personnes qui ne vont se désabonner (0.84) , pour les personnes qui vont se désabonner 0.65

### Les résultats du modèle SVM:

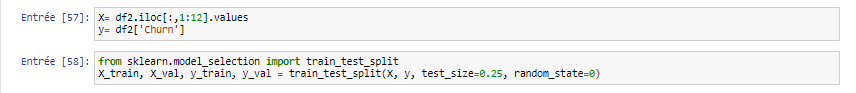
|  |  |  |  |
| --- | --- | --- | --- |
| Feature selection | training | testing | score |
| correlation | 0.80 | 0.80 | 0.80 |
| dummies | 0.81 | 0.81 | 0.80 |
| SFS (AUC) | 0.80 | 0.80 | 0.8017037387600567 |
| SFS (Accuracy) | 0.80 | 0.80 | 0.8017037387600567 |
| SBS (AUC) | 0.78 | 0.80 | 0.8035967818267865 |
| SBS (Accuracy) | 0.78 | 0.80 | 0.8035967818267865 |
| SFFS (AUC) | 0.80 | 0.80 | 0.8054898248935163 |
| SFFS (Accuracy) | 0.80 | 0.80 | 0.8017037387600567 |
| SBFS (AUC) | 0.80 | 0.70 | 0.8017037387600567 |
| SBFS (Accuracy) | 0.80 | 0.80 | 0.8017037387600567 |

## 5- Naive Bayes:

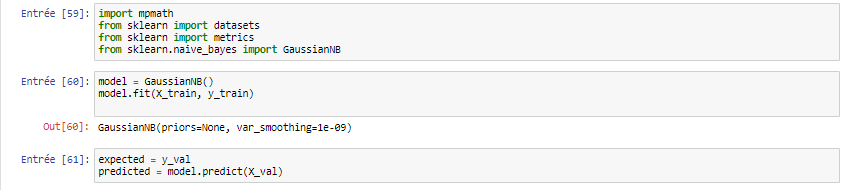
On a commencé par diviser notre dataset en deux parties:

-Partie entrainement (75%) .

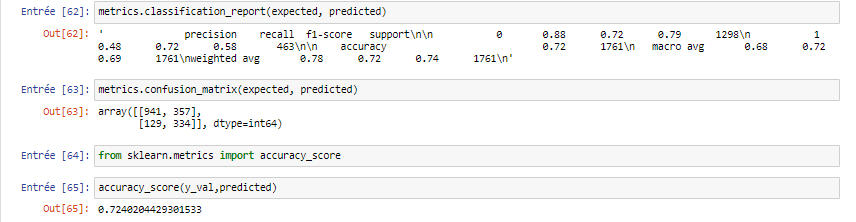
-Partie test (25%) .



Puis on a appliqué notre modèle sur ces deux derniers.



Notre modèle nous a donné les résultats suivants:

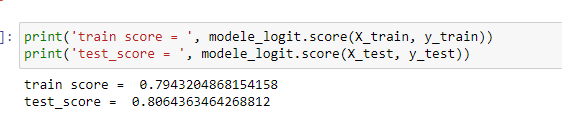


### Les résultats du modèle Naive Bayes:

|  |  |  |  |
| --- | --- | --- | --- |
| Feature selection | training | testing | score |
| correlation | 0.90 | 0.49 | 0.72 |
| dummies | 0.80 | 0.50 | 0.73 |
| SFS (AUC) | 0.8045841784989858 | 0.7835967818267865 | 0.7425461429247515 |
| SFS (Accuracy) | 0.748052738336714 | 0.7125461429247515 | 0.7425461429247515 |
| SBS (AUC) | 0.748052738336714 | 0.7125461429247515 | 0.7425461429247515 |
| SBS (Accuracy) | 0.748052738336714 | 0.7125461429247515 | 0.7425461429247515 |
| SFFS (AUC) | 0.90 | 0.50 | 0.7425461429247515 |
| SFFS (Accuracy) | 0.90 | 0.50 | 0.7425461429247515 |
| SBFS (AUC) | 0.90 | 0.50 | 0.7425461429247515 |
| SBFS (Accuracy) | 0.90 | 0.50 | 0.7425461429247515 |

### 

## 6-Régression logistique:



### Les résultats du modèle Régression logistique :

|  |  |  |  |
| --- | --- | --- | --- |
| Feature selection | training | testing | score |
| correlation | 0.60 | 0.62 | 0.80 |
| dummies | 0.7949290060851927 | 0.8064363464268812 | 0.8102224325603408 |
| SFS (AUC) | 0.7943204868154158 | 0.8064363464268812 | 0.8069096071935636 |
| SFS (Accuracy) | 0.7943204868154158 | 0.8064363464268812 | 0.8069096071935636 |
| SBS (AUC) | 0.7945233265720081 | 0.8078561287269286 | 0.8078561287269286 |
| SBS (Accuracy) | 0.7945233265720081 | 0.8078561287269286 | 0.8078561287269286 |
| SFFS (AUC) | 0.7943204868154158 | 0.8064363464268812 | 0.8064363464268812 |
| SFFS (Accuracy) | 0.7949290060851927 | 0.7949290060851927 | 0.8069096071935636 |
| SBFS (AUC) | 0.7949290060851927 | 0.8064363464268812 | 0.8017037387600567 |
| SBFS (Accuracy) | 0.7949290060851927 | 0.8064363464268812 | 0.8064363464268812 |

# 

# VI-Evaluation:

Cette section analyse les résultats obtenus à partir des expériences réalisées dans le cadre de la recherche. Ce chapitre évalue la puissance prédictive des modèles d'apprentissage automatique supervisé. Chaque modèle a été comparé par son score F1. Une comparaison a également été réalisée avec les résultats obtenus avec un jeu de données déséquilibré et des données échantillonnées.

En appliquant plusieurs algorithmes diffèrent après avoir passé par les étapes de la méthodologie CRISP et après avoir les tester avec plusieurs méthodes de feature selection, celles déjà proposées dans les deux articles donnés et d’autres méthodes pour améliorer ces résultats .

on a trouvé la valeur la plus performante est le modèle SBS et SBFS appliquée sur l’algorithme Classifier **K Nearest Neighbors ( KNN)** avec **81% pour le Test set est aussi pour le train set.**.

