DEKKAR Ahmed EL RAIS David IMMOUN Massinissa

Rapport de Projet

Détection de l’Attrition Client

Dans le cadre du master informatique 1ère année de l'Université de Paris, il nous a été donné pour objectif d'explorer et d'analyser des données relatives à l'attrition clients dans une entreprise d'opérateur mobile/internet et de proposer un modèle de prédiction performant.

Ce projet est accompagné de deux jeux de données, le premier (**DATA1_churn_analysis**) correspond à des clients ayant souscrit un abonnement mobile et le second (**DATA2_churn_analysis**) correspond à des clients ayant souscrit un abonnement internet.

De ce fait chaque partie sera divisée en deux sous parties chacune traitant un jeu de données.

Concernant les critères de conception, le langage de programmation utilisé pour réaliser ce projet est Python 3.0.

Les différentes bibliothèques utilisées sont :

- **Scikit-Learn :** permet l'apprentissage supervisée.
- **Pandas :** permet la manipulation et l'analyse des données.
- **Seaborn :** permet la tracer et visualiser les données sous forme graphique.
- **Imblearn :** fournit un certain nombre de technique de rééchantillonnage couramment utilisée dans les ensembles de données montrant un fort déséquilibre entre les classes.

Ce rapport retrace les travaux et les choix effectués durant ce projet.

Sommaire

I. Statistique Descriptive

1. Client ayant souscrit à un abonnement mobile

A. Description des données

Chacun des 2000 individus qui composent le jeu de données est constitué de 14 attributs:

- **Network_Age:** Variable Quantitative, correspond au temps passé depuis que le client à souscrit aux services de l'opérateur.
- **Aggregate_Total_Rev:** Variable Quantitative, correspond aux revenus que l'opérateur gagne à travers l'abonnement du client.
- **Aggregate_SMS_Rev:** Variable Quantitative, correspond aux revenus gagnés à travers le service de SMS.
- **Aggregate_Data_Rev:** Variable Quantitative, correspond aux revenus gagnés à travers le système de Données Mobile.
- **Aggregate_Data_Vol:** Variable Quantitative, correspond à la quantité de Données Mobile utilisée.
- **Aggregate_Calls:** Variable Quantitative, correspond au nombre d'appel émis.
- **Aggregate_Onnet_Rev:** Variable Quantitative, correspond aux revenus réalisés par l'opérateur lorsque le client entre en contact avec des clients du même opérateur.

- **Aggregate_Offnet_Rev:** Variable Quantitative, correspond aux revenus réalisés par l'opérateur lorsque le client entre en contact avec des client d'autre opérateur.

- **Aggregate_Complaint_Count:** Variable Quantitative, correspond au nombre de réclamation effectuée par le client.

- **Aug_User_Type:** Variable Qualitative, correspond au type de donnée mobile utilisé pendant le mois d'Août.

- **Sep_User_Type:** Variable Qualitative, correspond au type de donnée mobile utilisé pendant le mois de Septembre.

- **Aug_Fav_A:** Variable Qualitative, correspond à l'opérateur favori du client au mois d'Août.

- **Sep_Fav_A:** Variable Qualitative, correspond à l'opérateur favori du client au mois de Septembre.

- **Class:** Variable Qualitative, indique si la personne a quitté l'opérateur.

On remarque que le jeu de donnée contient plusieurs valeurs manquantes:
- **245** valeurs manquantes pour **Aug_User_Type**
- **206** valeurs manquantes pour **Sep_User_Type**
- **1** valeurs manquantes pour **Aug_Fav_A**
- **1** valeurs manquantes pour **Sep_Fav_A**

Pour résoudre ce problème, il existe une multitude de solutions.

Une solution classique serait de supprimer les individus qui ont un attribut manquant **(Complete Case Analysis)**, cependant le point faible de cette technique est qu'on écarte beaucoup d'observation (279 soit environ 14% du jeu de donnée), ce qui peut rendre instable les résultats (manque de performance du modele).

Une autre solution serait de faire une régression linéaire pour retrouver les valeurs manquantes (**Imputation par Régression).**

On peut aussi remplacer les valeurs quantitatives manquantes par les moyennes et les valeurs qualitatives par les modes, cependant ceci peut modifier le jeu de données, la moyenne étant très sensible aux valeurs aberrantes.

Une dernière méthode serait l'utilisation de l'aléatoire **(Hot Deck),** ce qui peut avoir pour inconvénient de modifier le jeu de données.

La solution optimal serait d'utiliser plusieurs techniques et de comparer les différents résultat.

Parmi les solutions proposées certaines ne peuvent être utilisée, en effet, par la présence de plusieurs valeurs aberrantes (cf. **partie B. Répartition des données**) on ne peut pas remplacer les valeurs manquantes par la moyenne/mode.

Pour la suite de ce projet, nous allons utiliser la méthode **Complete Case Analysis,** ce choix ce justifie par le fait qu'il serait préférable de supprimer 14% de notre jeu de donnée et donc de perdre en performance mais d'être certain d'avoir un jeu de donnée non altéré par des valeurs aberrante.

On passe donc d'un jeu de données de 2000 individus à un jeu de données de 1721 individus.

Il n'y a pas d'attributs constants.

B. Répartition des données

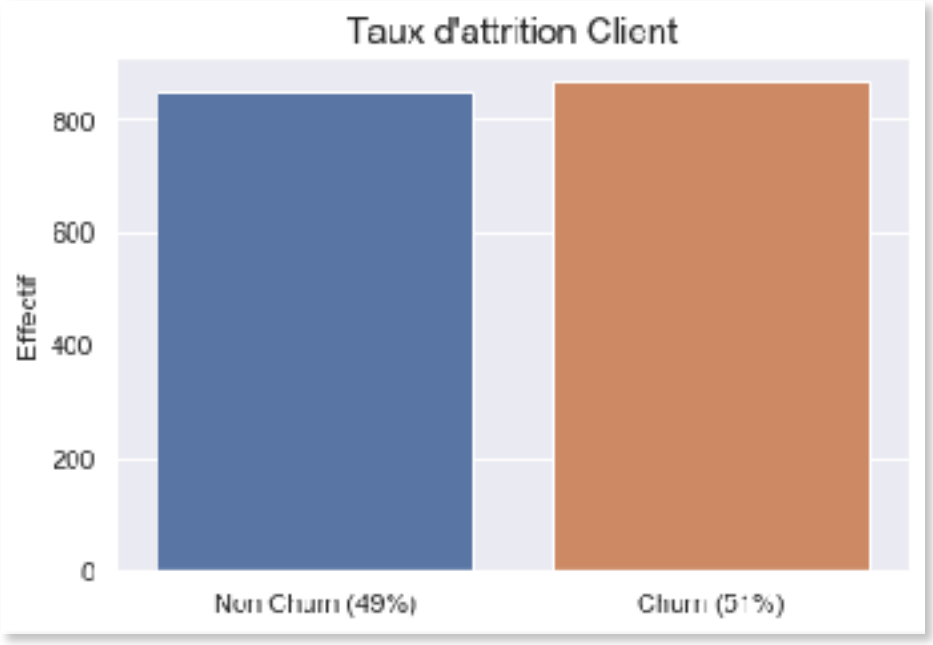


Figure 1.a Répartitions des individus en fonction de l'attrition client.

On remarque que les individus sont équitablement distribués (Churn: 868, Non Churn: 853), ce qui nous permettra de concevoir un bon système de classification car les échantillons sont bien représentés.

	Network Age	Total Rev	SMS Rev	Data Rev	Data Vol	Calls	ONNET Rev	OFFNET Rev	Complaint Count
Moyenne	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Ecart-Type	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Minimum	-1.20	-1.12	-0.79	-0.55	-0.44	-0.78	-0.61	-0.64	-0.53
Quartile 1	-0.86	-0.80	-0.67	-0.53	-0.44	-0.59	-0.50	-0.57	-0.53
Mediane	-0.20	-0.29	-0.32	-0.39	-0.40	-0.44	-0.42	-0.40	-0.53
Quartile 3	0.81	0.52	0.20	0.09	-0.11	0.33	0.14	0.09	0.27
Maximum	3.05	5.47	5.24	10.07	5.73	4.39	5.43	5.44	5.11

Tableau 1.a Description des variables (critères de dispersion et de position) version normaliser.

En étudiant les différents attributs, on s'aperçoit qu'ils n'ont pas la même moyenne ni le même écart-type, pour établir un système de classification performant il faut normaliser (centrer et réduire) les données.

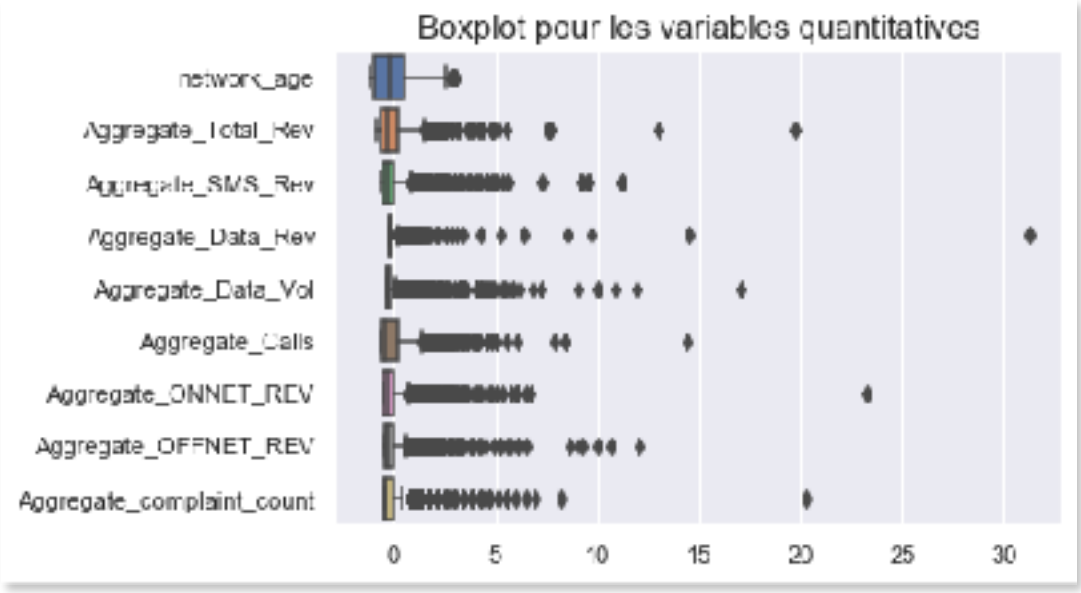


Figure 2.a Boxplot pour les différentes variables quantitatives

Par lecture du graphe, on peut supposer que le dataset contient plusieurs valeurs aberrantes (par exemple pour l'attribut **Aggregate_Data_Rev**, la valeur supérieur à 30 peut être considérée comme une valeur aberrante).

Cependant nous n'avons pas d'avis d'un expert dans le domaine et nos connaissances sont limitées, nous ne pouvons pas affirmer avec exactitude qu'il s'agit bien de valeur aberrante.

Par mesure de précaution nous allons utiliser la **Z-Value** pour traiter les valeurs aberrantes, à un intervalle de confiance de 95%, nous devons supprimer les valeurs qui ont une Z-Value inférieur/supérieur à 1.96.

Prendre une Z-Value égale à 1.96 ne permet pas d'avoir un modele performant au vu du nombre de valeur extrême supérieur, nous allons donc prendre une Z-Value égale à 3.0, cette valeur est choisis car par tâtonnement elle permet d'avoir les meilleures performances.

Après suppression des valeurs aberrantes, on passe de 1721 individus à 1546. Le nombre d'individus qui se sont désabonner (Churn) est égale a 805 soit 52% de la population et la proportion de non désabonnement (Non Churn) est égale à 741 soit 48% de la population.

Les individus restent équitablement distribués.

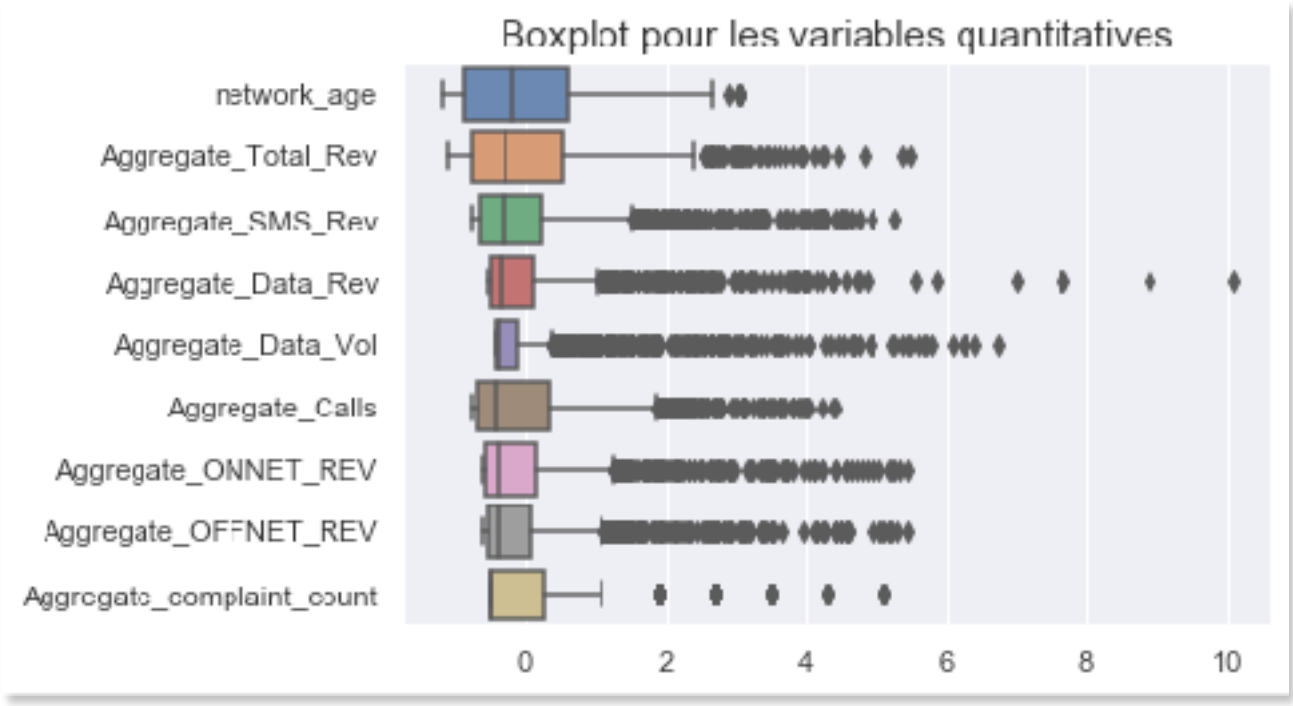


Figure 3.a Boxplot pour différentes variables quantitatives, après application de la Z-Value

Malgré la suppression des valeurs aberrantes, le jeu de données présente encore beaucoup de valeurs extrêmes supérieures, et peut être des valeurs aberrantes **(cas de Aggregate_complaint_count),** il faudra en tenir compte lors de la classification car elles pourraient influencer les résultats.

En étudiant les caractéristiques de tendance centrale et de dispersion, on s'aperçoit que toutes les variables sont asymétriques et présentent beaucoup de valeurs extrêmes supérieures largement supérieures à la règle **q3+1.5*IIQ** (potentiellement des valeurs aberrantes).

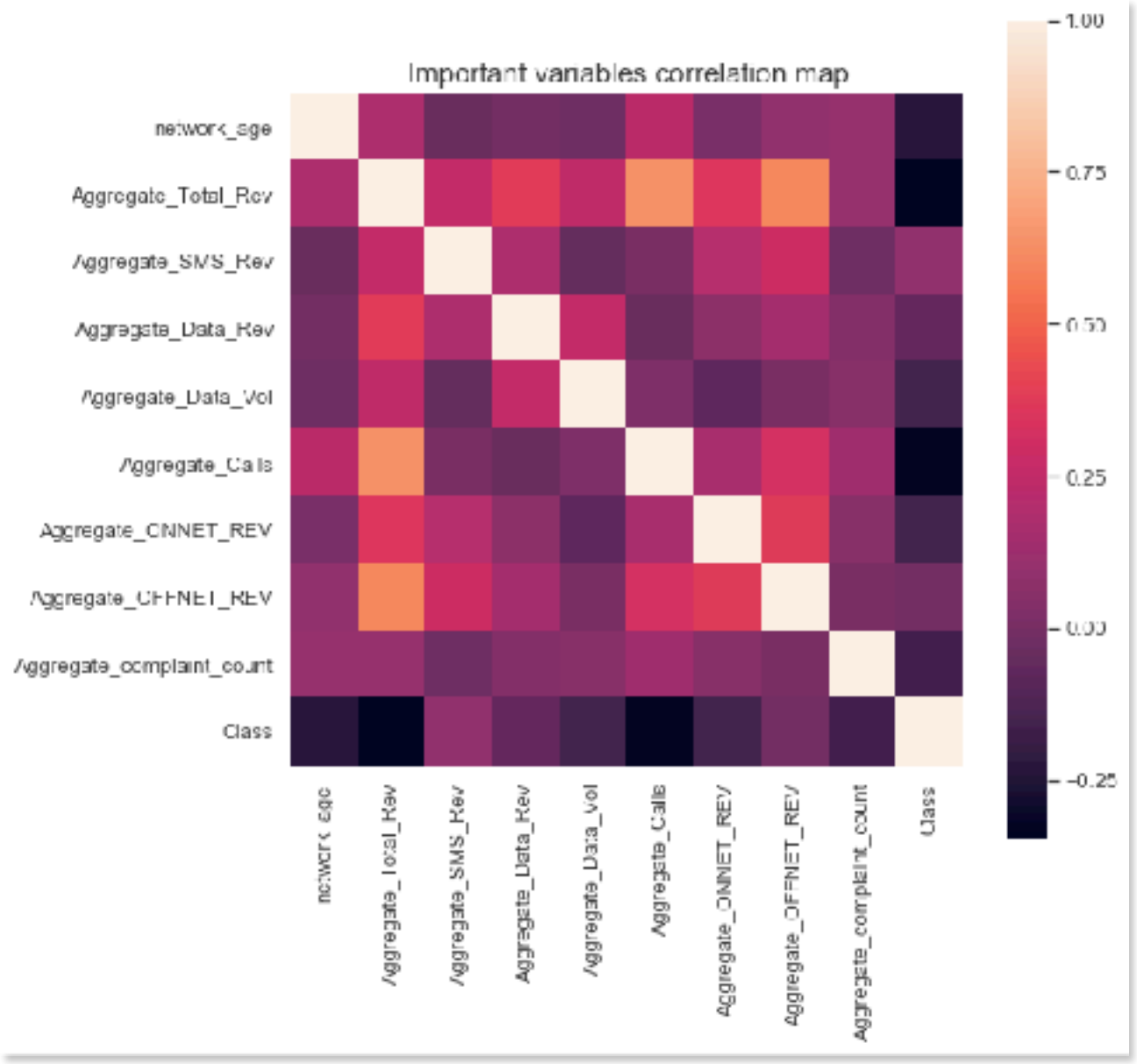


Figure 4.a Matrice de corrélation des attributs.

Les variables **Total Rev** et **Calls** (corrélation : 0.64) sont bien corrélées, donc on suppose que les revenus s'appuie principalement (par comparaison avec **Data** et **SMS**) sur les appels, donc plus la valeurs de **Calls** sera élevée et plus la valeur de **Total Rev** le sera aussi.

Les variables **Total Rev** et **Data Rev** (corrélation : 0.38) sont légèrement corrélées on peut en déduire que la consommation de données mobile augmente légèrement le revenus, ce qui suppose que **Data Rev** peut influencer faiblement la décision de désabonnement (Churn).

Les variables **Total Rev** et **ONNET Rev** (corrélation : 0.36) et les variables **Total Rev** et **OFFNET Rev** (corrélation : 0.60) sont assez corrélées : donc les (appels, SMS) vers des opérateurs différents coutent plus chère que des (appels, SMS) entre des individus qui appartiennent au même opérateur.

Les opérateurs les moins populaires sont donc plus susceptibles de rencontrer de l'attrition de client.

Les variables **OFFNET Rev** et **ONNET Rev** (corrélation : 0.37), sont légèrement corrélées, ce qui est logique au cours de l'abonnement de l'individus des revenus OFFNET et ONNET sont effectués.

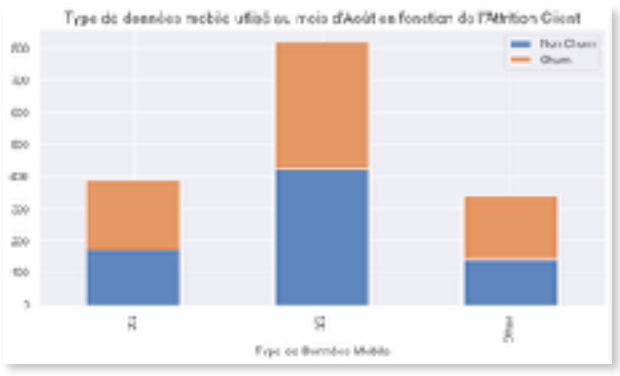
Les variable **OFFNET Rev** et **Calls** (corrélation : 0.32), sont légèrement corrélées, ce qui suppose que les appels sont principalement (par opposition à la corrélation entre **ONNET Rev** et **Calls,** corrélation : 0.17) fait vers des opérateurs différents ce qui a pour effet d'augmenter la valeur de **Total Rev.**

Hormis c'est variables, il ne semble pas y avoir de corrélation entre les autres variables.

C. Corrélation des données avec Churn

1.Variable Qualitative

Afin d'étudier la corrélation des données avec Churn, étudiant la proportion de Churn et de non Churn pour chaque variable catégoriel.



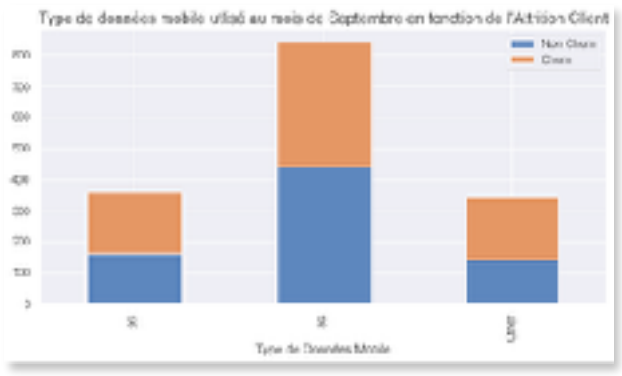


Figure 5.a Proportion de Churn en fonction du type de données mobile utilisées au mois d'Août (à gauche) et au mois de Septembre (à droite).

En ce qui concernent les différentes données mobile utilisées pendant le mois d'Août et le mois de Septembre, on remarque que les graphes suivent une même tendance, donc on suppose que le mois n'influe pas sur la prise de décision de désabonnement.

En ce qui concerne les différents types de données, pour les utilisateurs qui utilisent un type de données mobile hors la 3G, l'attrition client est plus marqués donc on suppose que sa peut influer sur le désabonnement.

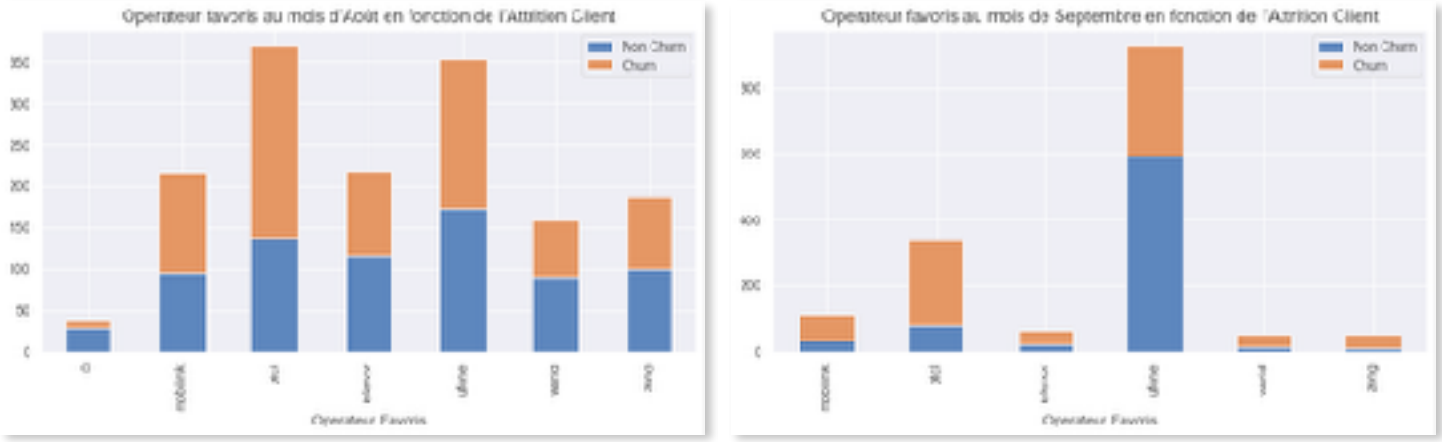


Figure 6.a Proportion de Churn en fonction de l'opérateur favoris au mois de Août (à gauche) et au mois de Septembre (à droite).

Dans le graphe ci-dessus, on voit que les individus qui préfèrent les opérateurs (**hors ufone**) ont tendances à se désabonner.
Par contre, les gens qui préfère cet opérateur (**ufone**) ont tendances à rester.

2.Variable Quantitative

Dans un second temps, étudiant la proportion de Churn et de non Churn pour chaque variable quantitative.

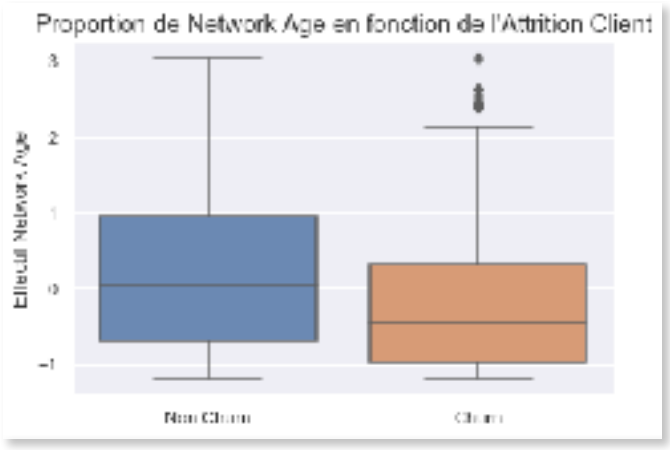


Figure 7.a Durée de l'abonnement en fonction de l'attrition client.

On remarque que les individus les plus anciens sont les moins susceptible à l'attrition.

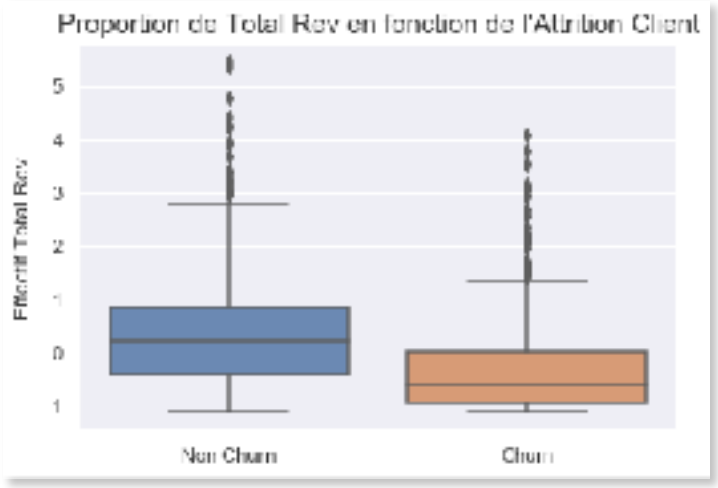


Figure 8.a Revenue Total en fonction de l'attrition client.

On constate que les individus non Churn paye plus chère que les individus Churn, par contre, il y a plus de valeur extreme pour les churn il paye plus que la médiane, donc ils sont plus susceptible de se désabonner.

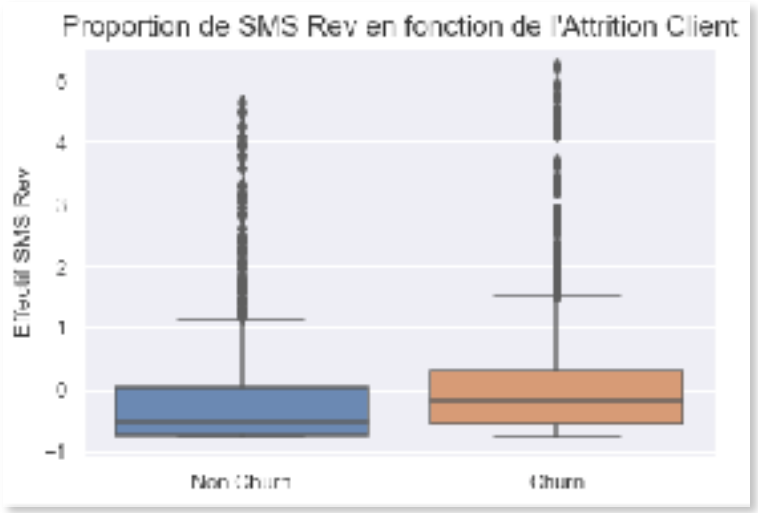
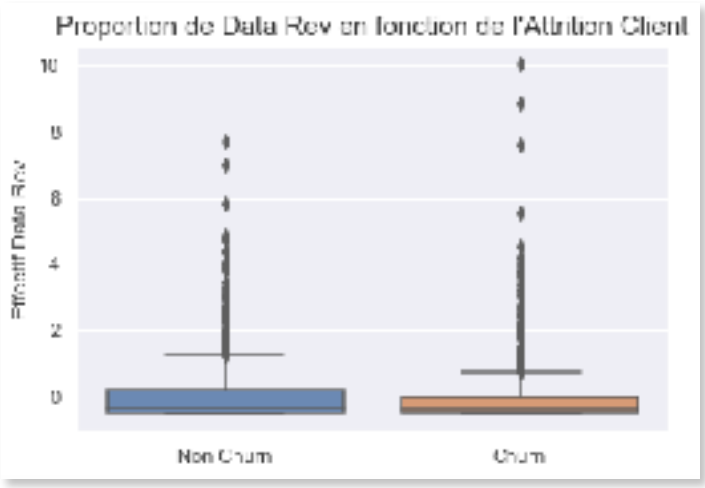


Figure 9.a Revenue des SMS en fonction de l'attrition client.

En ce qui concerne les SMS, on remarque que plus un individu envoie un SMS plus il est susceptible à l'attrition client.



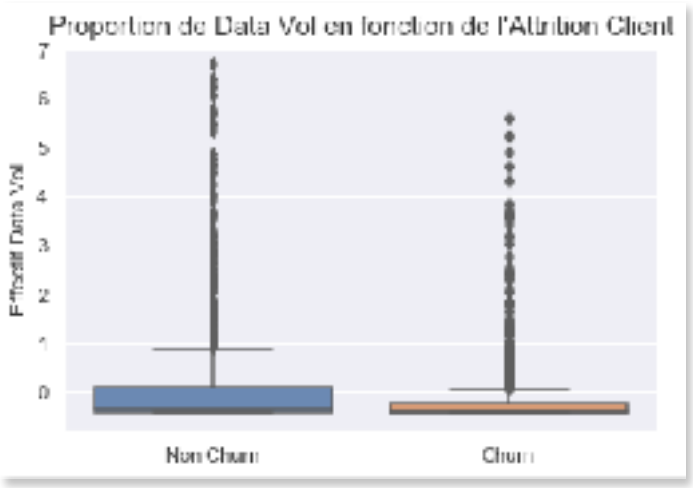


Figure 10.a Information sur les données mobile en fonction de l'attrition client (à gauche les revenus liée au données mobiles) et à droite (les consommations de données mobiles).

On remarque à travers les revenus des données mobile et la quantité des données mobiles que un individu qui Churn utilise moins les données mobiles qu'un individu qui ne Churn pas.

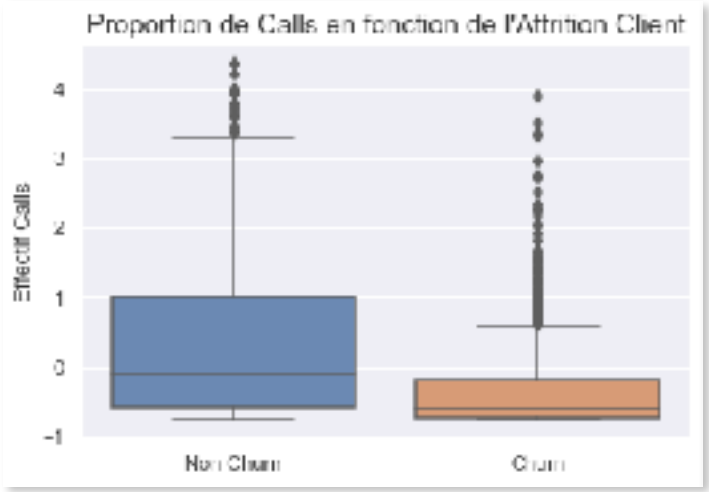
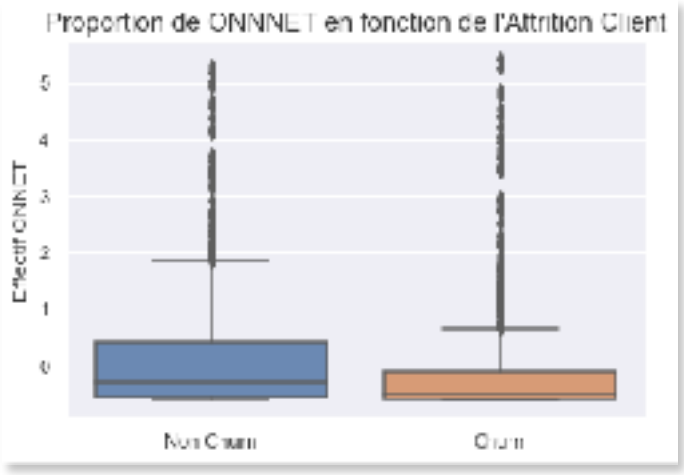


Figure 11.a Proportion d'appel émis en fonction de l'attrition client.

Les individus non Churn effectue plus d'appel que les Churn.



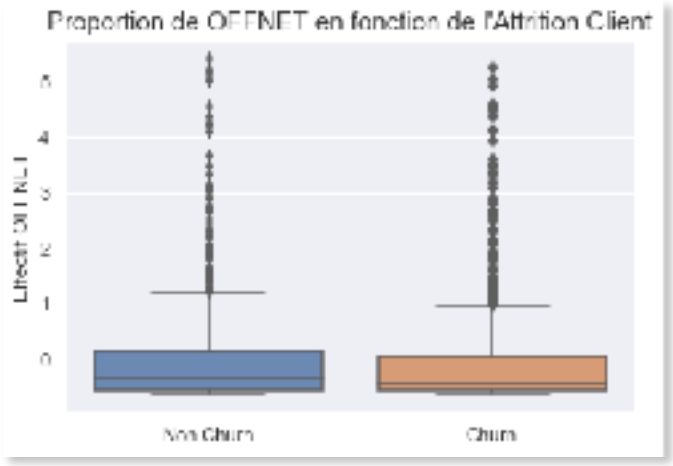


Figure 12.a Different type de consommation en fonction de l'attrition client.

On constate que si un individu effectue des opérations ONNET, il est plus probable de rester avec le même opérateur car si il change d'opérateur il effectuera plus d'opération OFFNET ce qui lui augmentera sa facture.

Pour le deuxième graphe, on remarque que les non churn effectue moins de OFFNET que de ONNET et pour les churn l'effectif dans OFFNET est supérieur à ONNET, donc favorise plus l'attrition client.

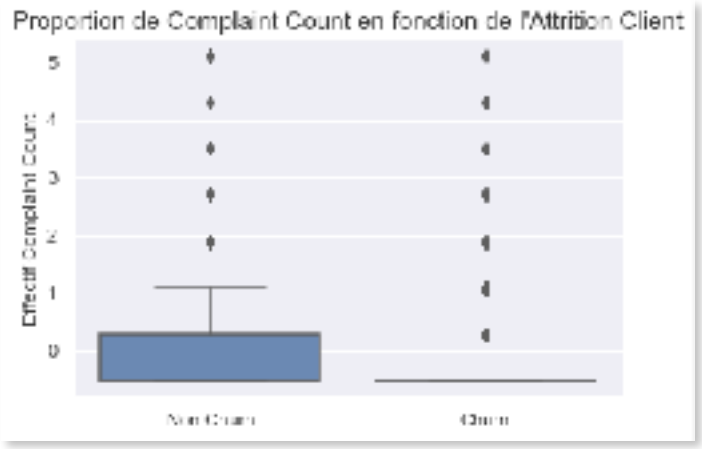


Figure 13.a Nombre de plainte effectue en fonction de l'attrition client.

On remarque que le nombre de plainte dans non churn est plus important donc on en déduit que plus l'individu fait des plaintes donc plus il est intéressé par l'opérateur donc il est plus susceptible de ne pas se désabonner.

Donc à travers l'étude des différents tableau et graphique, on peut conclure que plus un individu profite de son abonnement à travers les différents services fourni et moins il sera susceptible à l'attrition client.

De plus les variables qui semblent fortement liées à l'attrition client sont **: Total Rev, Network Age, Aggregate Calls, Aggregate_ONNET_REV et Complaint Count.**

Pour la suite du projet nous devons convertir les variables qualitative en variable numérique car les modèles de Machines Learning requièrent que toutes les entrées doivent être numériques.

Nous allons remplacer chaque label d'une variables qualitatives par un nombre entier **(LabelEncoder),** nous préférons utiliser cette

technique plutôt que la méthode des **dummy variables** car il y a plusieurs label distinct.

2. Client ayant souscrit à un abonnement internet

Chacun des 3333 individus qui composent le jeu de données est constitué de 21 attributs:

- **State:** Variable Qualitative, correspond à l'état dans lequel vie l'individu.

- **Account Length:** Variable Quantitative, correspond au nombre de jours pendant lesquels l'utilisateur possède le compte.

- **Area Code:** Variable Quantitative, correspond à l'indicateur téléphonique de la zone géographique.

- **Phone Number:** Variable Qualitative, correspond au numéro de téléphone de l'individu.

- **International Plan:** Variable Qualitative, correspond à la présence d'un abonnement à l'international.

- **Voice Mail Plan:** Variable Qualitative, correspond à la présence d'une boite vocale.

- **Number Vmail Messages:** Variable Quantitative, correspond au nombre de message dans la boite vocale que l'individu a envoyé.

- **Total Day Minutes:** Variable Quantitative, correspond au nombre de minute (cumulée) d'appel pendant la journée.

- **Total Day Calls:** Variable Quantitative, correspond au nombre d'appel pendant la journée.

- **Total Day Charge:** Variable Quantitative, correspond au montant total que l'utilisateur a facturé à la société de télécommunications pour les appels pendant la journée.

- **Total Eve Minutes:** Variable Quantitative, correspond au correspond au nombre de minute (cumulée) d'appel durant l'après midi.

- **Total Eve Calls:** Variable Quantitative, correspond au correspond au nombre d'appel durant l'après midi.

- **Total Eve Charge :** Variable Quantitative, correspond au montant total que l'utilisateur a facturé à la société de télécommunication pour les appels pendant l'après midi.

- **Total Night Minutes:** Variable Quantitative, correspond au nombre de minute (cumulée) d'appel pendant la nuit.

- **Total Night Calls:** Variable Quantitative, correspond au nombre d'appel pendant la nuit.

- **Total Night Charge:** Variable Quantitative, correspond au montant total que l'utilisateur a facturé à la société de télécommunication pour les appels pendant la nuit.

- **Total Intl Minutes:** Variable Quantitative, correspond au nombre de minute d'appel vers l'international.

- **Total Intl Calls:** Variable Quantitative, correspond au nombre d'appel à l'international.

- **Total Intl Charge:** Variable Quantitative, correspond au montant total que l'utilisateur a facturé à la société de télécommunication pour les appels internationaux.

- **Customer Service Calls:** Variable Quantitative, correspond au nombre d'appel au service client.

- **Churn:** Variable Qualitative, indique si le client à quitter l'opérateur.

Le jeu de données ne comporte aucune valeur manquante.

Cependant, il existe un attribut constant (chaque individus possède son numéro de téléphone) **Phone Number**, nous pouvons donc la supprimer du jeu de données.

B. Répartition des données

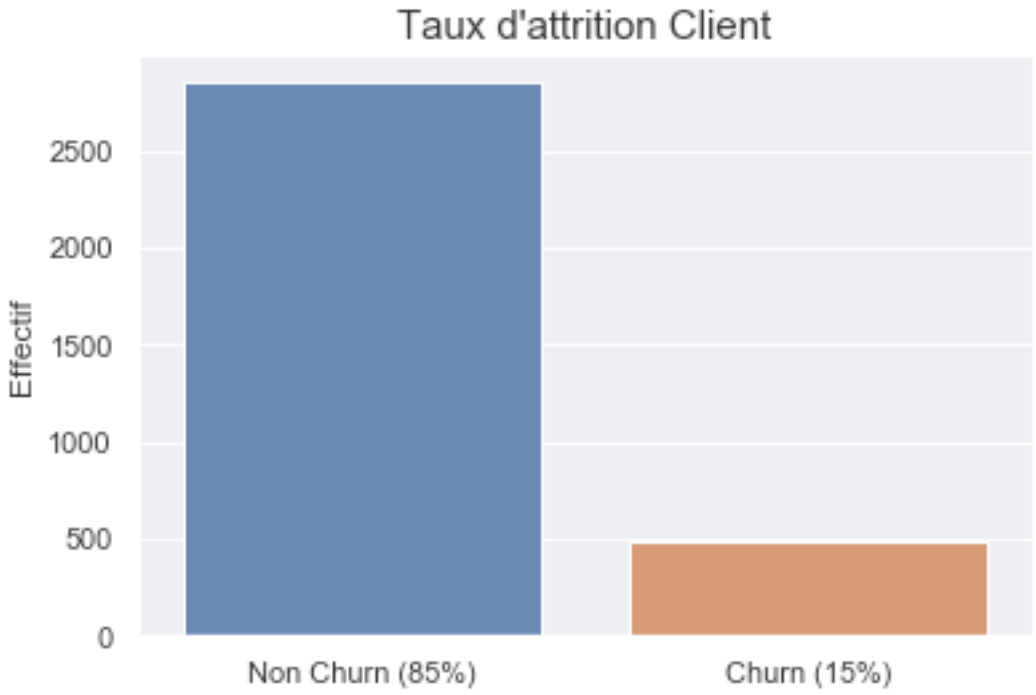


Figure 1.b Répartitions des individus en fonction de la variable Churn.

On remarque que les individus ne sont pas équitablement distribués (Churn: 483, Non Churn: 2850), on a environ 6 fois plus de chance de tomber sur un individu désabonné que sur un individu non désabonné.

Pour ne pas avoir un modèle avec un biais pour le classe Non Churn, nous devons rétablir la répartition des individus.

Dans la suite de ce projet, nous étudieront la technique **SMOTE (Synthetic Minority Over-sampling Technique)**, comme solution à ce problème.

De la même manière que le dataset précédent, nous allons étudier la répartition des données à travers une normalisation et suppression des valeurs aberrantes à un taux de confiance de 95%, soit une Z-Value égale à 1.91.

Après suppression des valeurs aberrantes, on passe de 3333 individus à 1805. Le nombre d'individus qui se sont désabonner (Non Churn) est égale a 1601 soit 88% de la population et la proportion de non désabonnement (Churn) est égale à 204 soit 12% de la population.

	Account Length	Area Code	Number Vmail Messages	Day Minutes	Day Calls	Day Charge	Eve Minutes	Eve Calls	Eve Charge	Night Minutes	Night Calls	Night Charge	Intl Minutes	Intl Calls	Intl Charge	Customer Service Calls
Moyenne	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Ecart-Type	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Minimum	-2.17	-0.70	-0.59	-2.25	-2.17	-2.25	-2.21	-2.20	-2.21	-2.23	-2.22	-2.23	-2.31	-1.64	-2.31	-1.28
Quartile 1	-0.76	-0.70	-0.59	-0.73	-0.72	-0.73	-0.75	-0.73	-0.75	-0.74	-0.73	-0.74	-0.67	-0.69	-0.66	-0.40
Mediane	0.01	-0.53	-0.59	-0.01	-0.03	-0.01	0.04	0.04	0.04	0.01	0.05	0.01	0.02	-0.11	0.02	-0.40
Quartile 3	0.71	1.38	-0.59	0.72	0.73	0.72	0.73	0.74	0.73	0.74	0.76	0.74	0.75	0.40	0.76	0.48
Maximum	2.25	1.69	2.62	2.21	2.18	2.21	2.23	2.21	2.25	2.20	2.19	2.20	2.22	2.45	2.23	2.25

Tableau 1.b. Description des variables (critères de dispersion et de position).

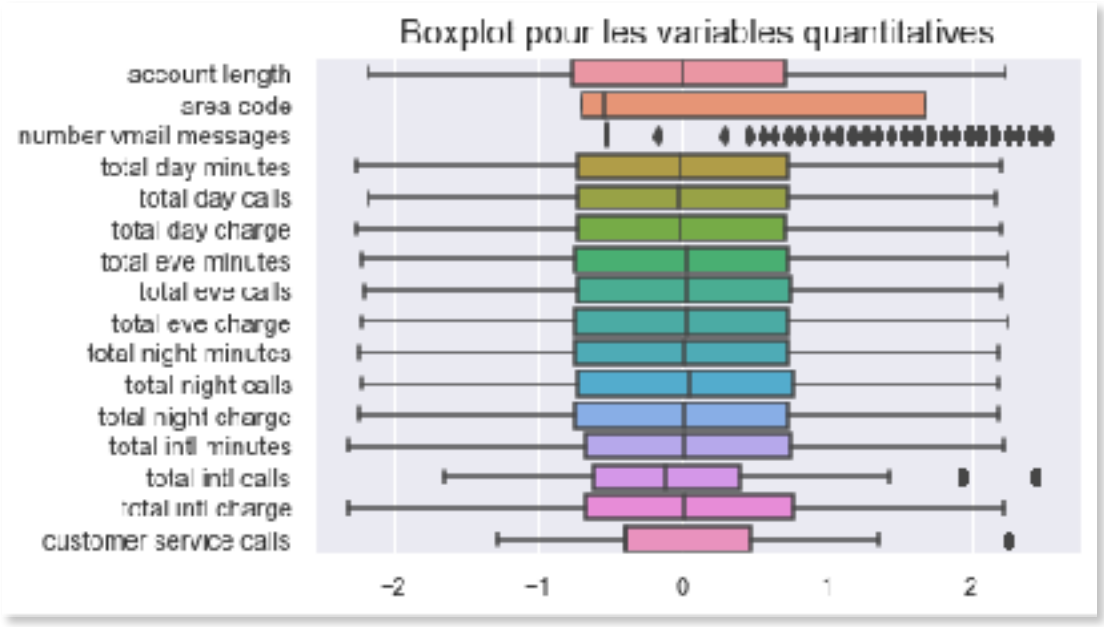


Figure 2.b. Boxplot pour les différentes variables quantitatives.

En étudiant les caractéristiques de dispersion et de tendances centrales, on s'aperçoit que hormis la variable **Area Code,** le graphe ne présente quasiment pas de valeur extrême supérieur et quasiment tous les boxplot sont symétrique ce qui nous permettra d'obtenir de bon résultat pour notre modele.

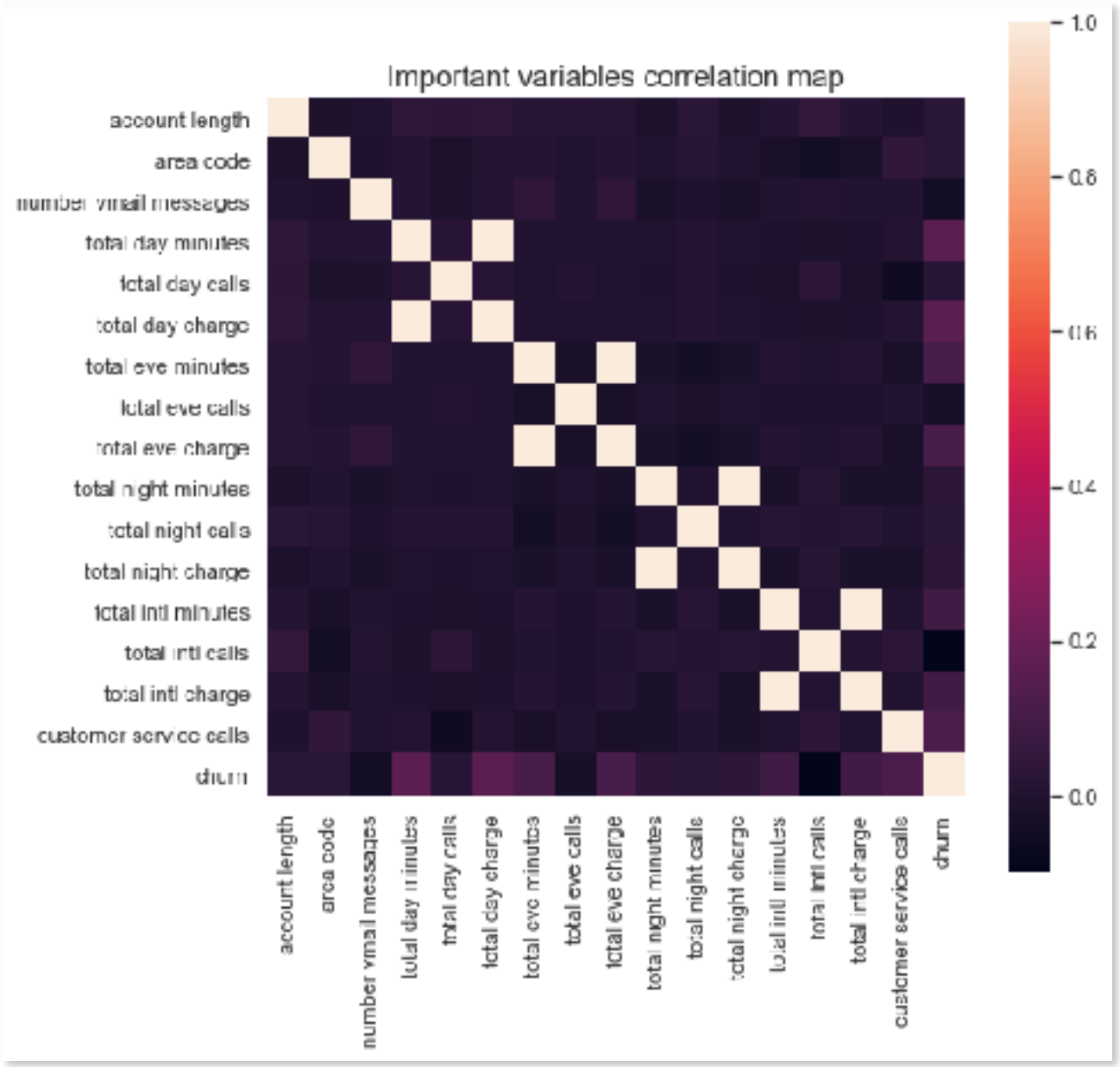


Figure 3.b. Matrice de corrélation des attributs.

Les variables **Total Day Charge** et **Total Day Minutes** (corrélation: 1) sont très bien corrélées, ce qui est logique, lorsqu'un individu effectue un appel pendant la journée, il sera facturée le montant de sa communication.

Nous pouvons émettre la même conclusion sur les variables : **Total Night Charge** et **Total Night Minutes** (corrélation: 1) et **Total Intl Charge** et **Total Intl Minutes** (corrélation: 1).

Les variables **Total Day Charge** et **Total Eve Minutes** (corrélation: 1) sont très bien corrélées, donc on suppose que lorsqu'un individu effectue un appel pendant l'après midi, ceci fait augmenter sa facture de la journée.

C. Corrélation des données avec Churn

1.Variable Qualitative

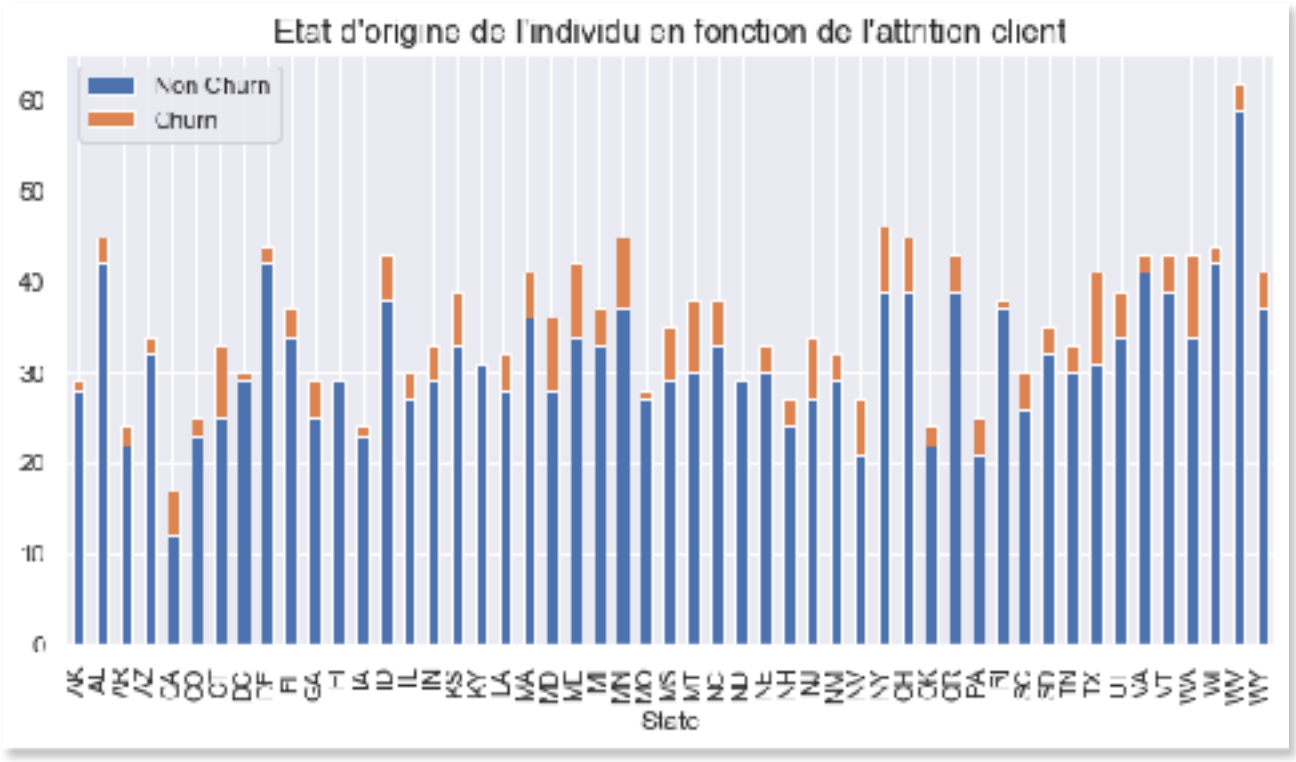


Figure 4.b. Etats d'origine en fonction de l'attrition client.

Nous remarquons que certains Etats présente moins d'attrition client pas exemple : RI, WI, MO et certains ont une proportion plus élevée comme WA, MD et TX.

Donc on suppose que l'état pourrait être utile pour prédire l'attrition client.

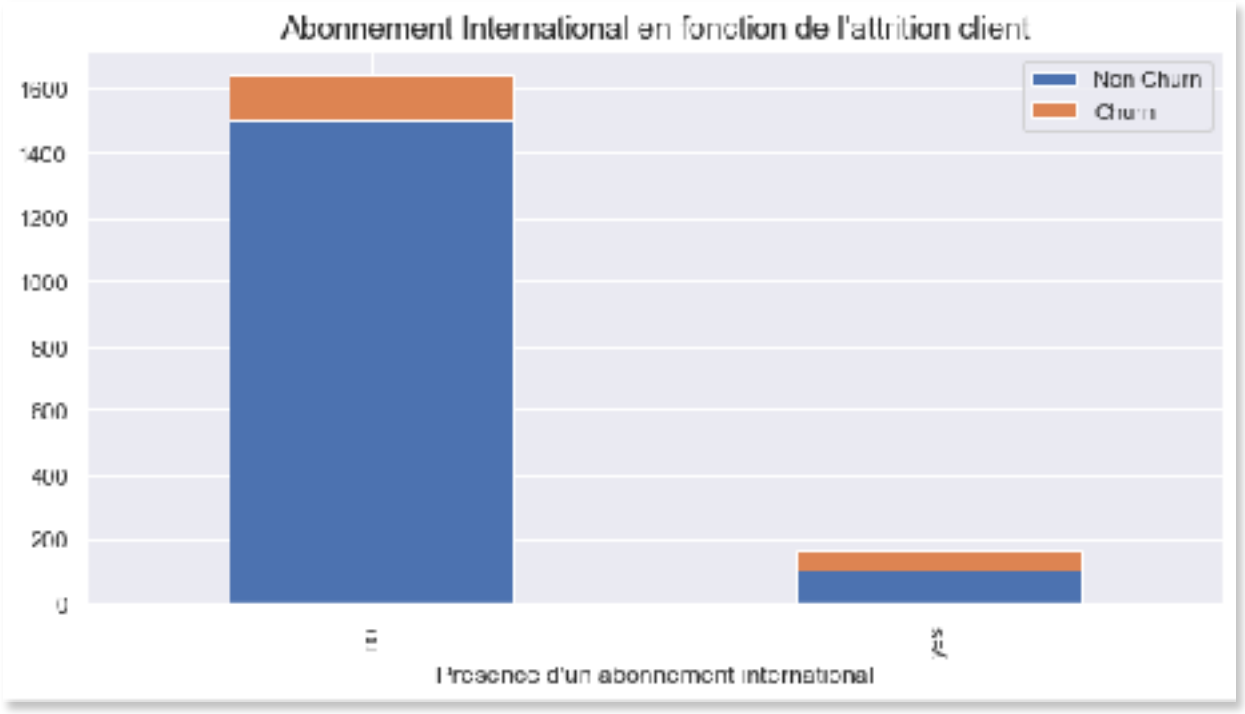


Figure 5.b. Présence d'un abonnement international en fonction de l'attrition client.

On remarque que les individus ne possédant pas un abonnement à l'international sont moins susceptible de se désabonner tandis que les individus présentant un abonnement à l'international sont divisé environ à 50%.

Donc l'abonnement à l'international nous sera utile pour prédire l'attrition client.

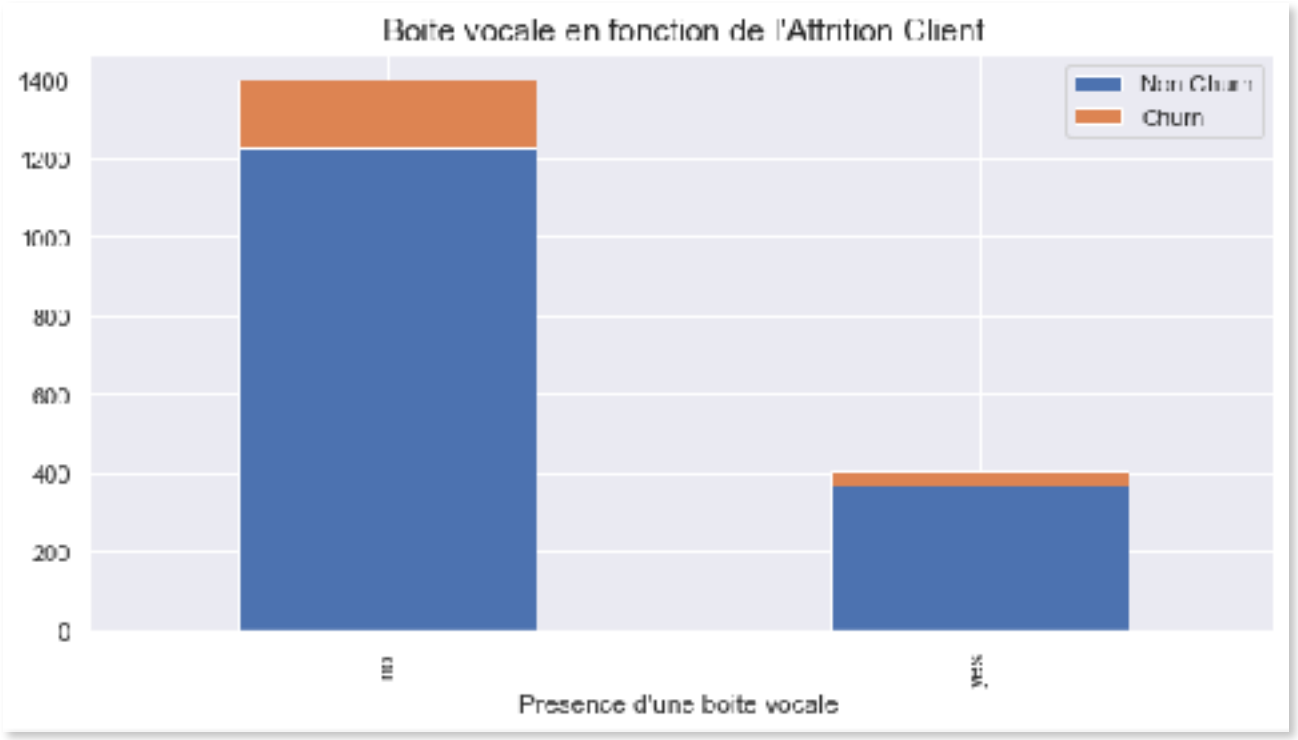


Figure 6.b. Présence d'une boite vocale en fonction de l'attrition client.

Le boxplot de la présence d'une boite vocale, suit la même tendance que le boxplot précédent, on suppose donc que la présence de la boite vocale peut influencer l'attrition client.

2.Variable Quantitative

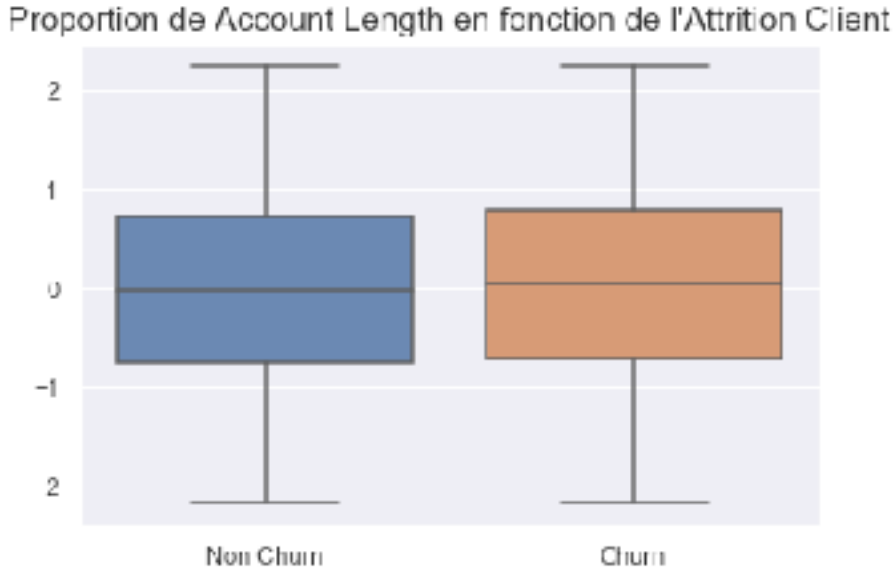


Figure 7.b. Chiffre d'affaire de l'opérateur en fonction de l'attrition client.
On remarque que les boxplot sont quasi-similaire donc Account Length n'est pas un critère de classification.

De même, les variables **Area Code, Number Vmail Messages, Total Day Calls, Total Eve Calls, Total Night Minutes,** T**otal Night Calls** et **Total Night Charge** ont quasiment le même boxplot on suppose qu'il ne sont pas des critères de classification.

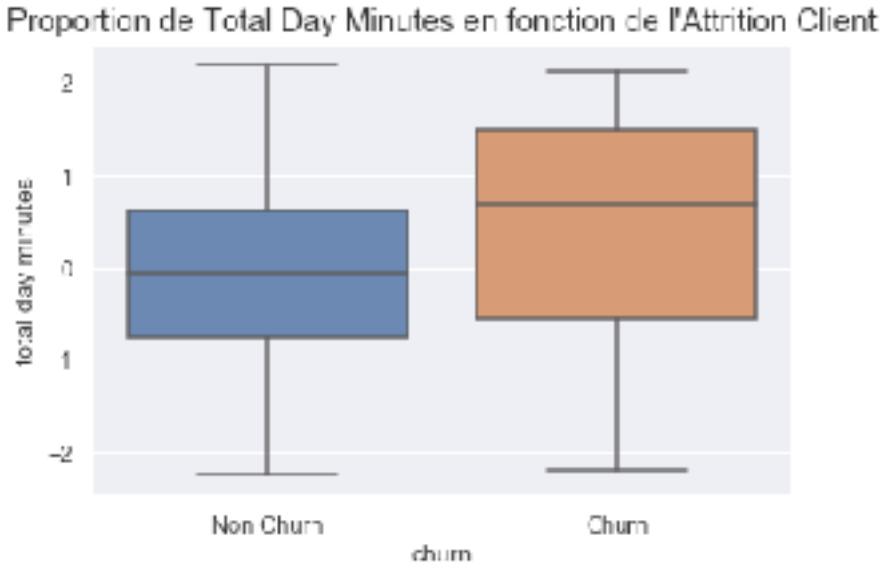


Figure 8.b. Nombre de minutes d'appel par jour en fonction de l'attrition client.
On remarque que les individus désabonné passe plus de temps au téléphone pendant la journée.

On obtient le même comportement avec les différents boxplot: **Total Day Charge, Total Eve Minutes, Total Eve Charge, Total Intl Minutes** et **Total Intl Charge.**

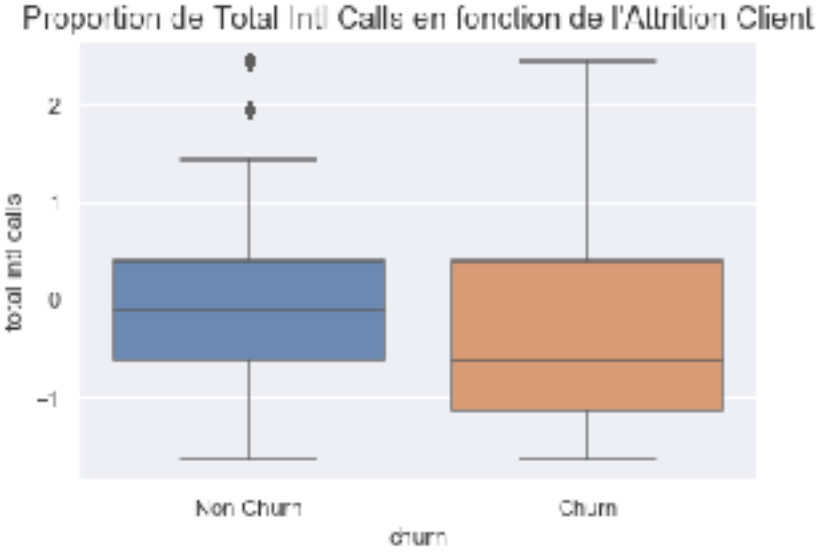


Figure 9.b. Nombre d'appel à l'international en fonction de l'attrition client.
On remarque que plus un individu fait des appel à l'international plus il est susceptible de rester fidèle à l'opérateur.

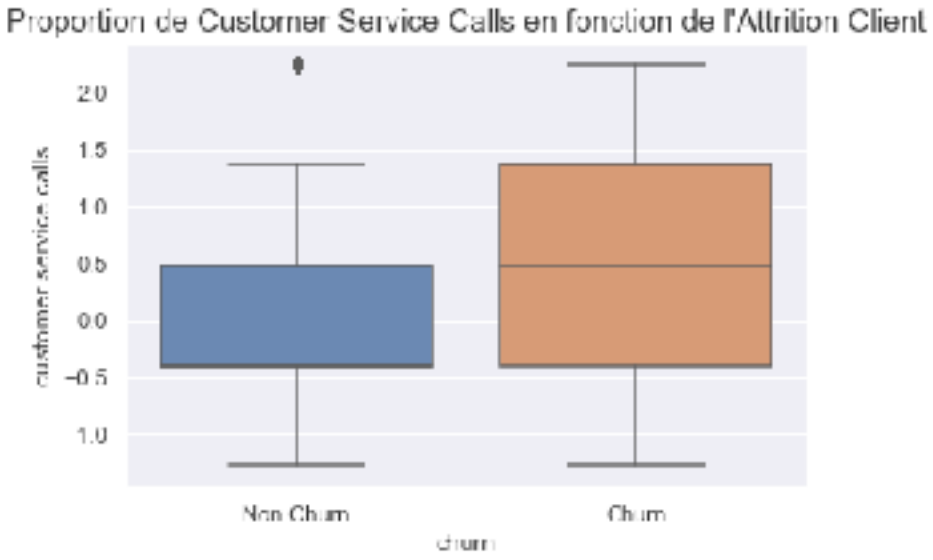


Figure 10.b. Nombre d'appel au service client en fonction de l'attrition client.
On remarque que plus un individu fait des appels au client plus il est susceptible de se désabonné.

Donc à travers l'étude des différents tableau et graphique, on peut conclure qu'un individus susceptible de se désabonner est caractérisé par une durée d'appel longue (pendant la journée), les appels se font la plupart du temps au niveau national avec un nombre important d'appel au service client.

Donc les variables qui semble fortement liées à l'attrition client sont : Total Day Minutes, Total Day Charge, Total Eve Minutes, Total Eve Charge, Total Intl Minutes et Total Intl Charge, Total Intl Calls et Customer Service Calls.

Pour les variables qualitatives : **International plan** et **voice mail plan** nous allons remplacer leur valeurs par des 0 et des 1 (cas des variables binaires).

Et pour la variable **state**, nous allons utiliser la même technique que précédemment c'est-à-dire remplacer chaque label d'une variables qualitatives par un nombre entier.

II.Régression Logistique

La régression logistique est une technique prédictive.

Elle vise à construire un modèle permettant de prédire/expliquer les valeurs prises par une variable cible qualitative à partir d'un ensemble de variables explicatives quantitatives ou qualitatives.

1.Client ayant souscrit à un abonnement mobile

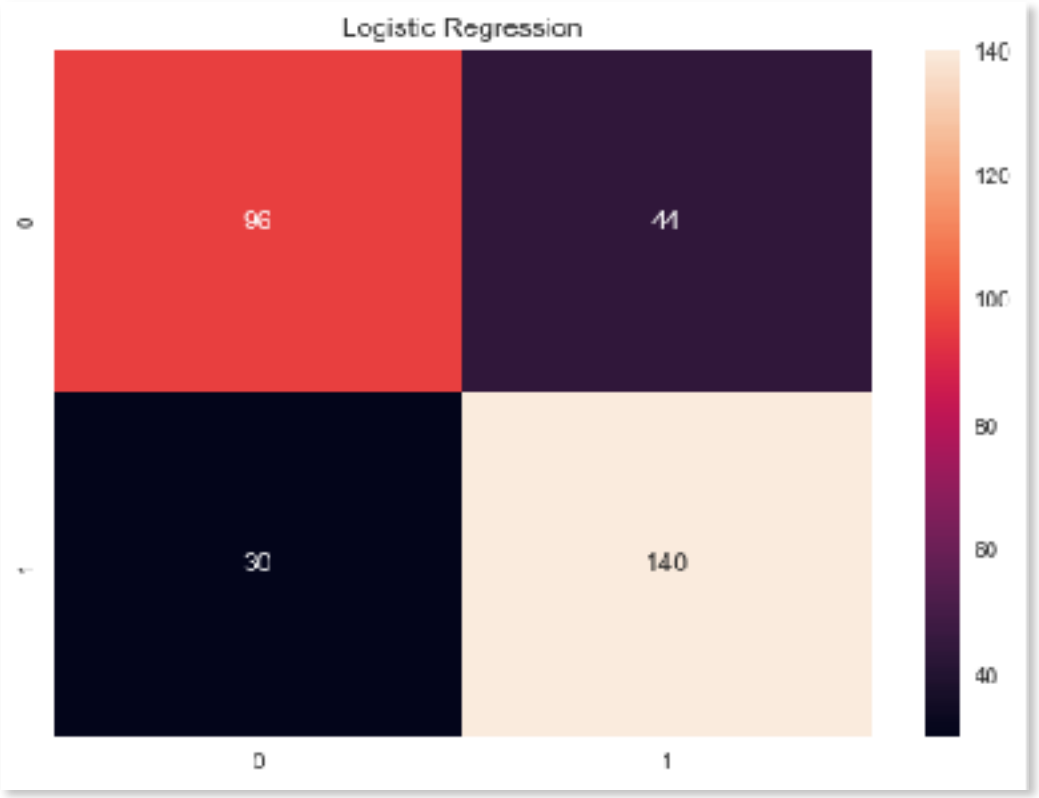


Figure 14.a Matrice de confusion.

Par lecture de la matrice de confusion, on s'aperçoit que notre modele est bon mais pas excellent plusieurs individus sont mal classé (74).

Pour pouvoir analyser de manière précise les performances du modele nous devons étudier le rapport de classification.

	precision	recall	f1-score	support
Non Churn	0.76	0.69	0.72	140
Churn	0.76	0.82	0.79	170
accuracy			0.76	310
macro avg	0.76	0.75	0.76	310
weighted avg	0.76	0.76	0.76	310

Tableau 2.a Rapport de la classification en utilisant le Régression Logistique.

Le rapport de classification justifie ce que nous disions précédemment, le modèle est bon, le modèle arrive a effectuer un bon classement environ 3 fois sur 4 (**76%**).

De plus le modele reconnait autant les individus Churn et non Churn, ceci est à une distribution équitable des individus dans l'attribut churn et la suppression des valeurs aberrantes.

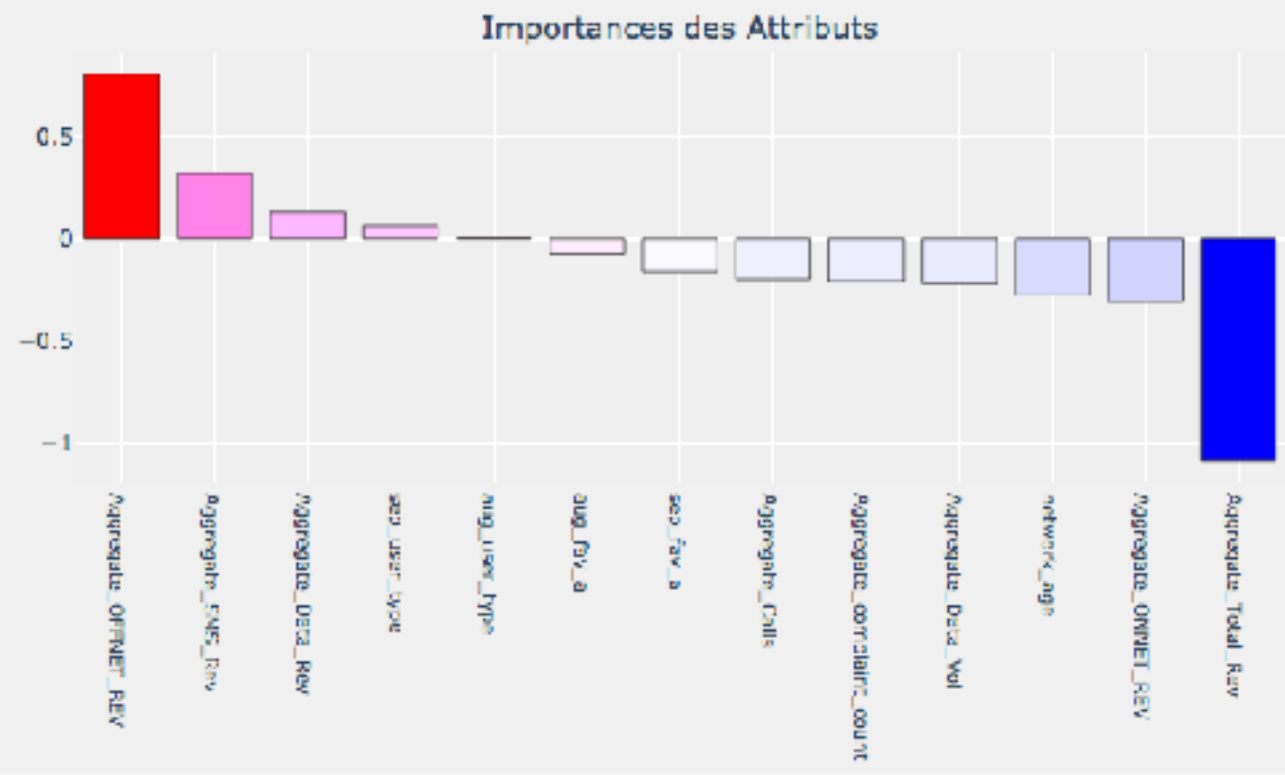


Figure 15.a Classement des attributs les plus importants

Les attributs les plus important sont :

- **Aggregate_OFFNET_Rev**
- **Aggregate_SMS_Rev**
- **Aggregate_Total_Rev**
- **Aggregate_ONNET_Rev**

2. Client ayant souscrit à un abonnement internet

Comme nous l'avons vu dans le chapitre précédent, nous devons équilibrer le dataset.

Il faut savoir qu'il existe plusieurs stratégies d'échantillonnage qui permettent de gérer un jeu de données déséquilibrée.

La première étant la suppression aléatoire des individus de la classe majoritaire **(sous échantillonnage)** avec un risque de supprimer des individus représentatifs pour l'échantillonnage, on peut cependant éviter cet inconvénient, on sélectionnant les individus de la classe majoritaire qui sont les moins sensibles au bruit et qui ne sont pas loin de leur voisinage **(Lien de Tomek).**

La seconde étant d'augmenter les individus de la classe minoritaire **(sur échantillonnage)** avec un risque de sur-apprentissage, on peut aussi éviter cet inconvénient on utilisant la méthode **SMOTE.**

La méthode **SMOTE** consiste à augmenter le rappel pour la classe minoritaire en générant des individus synthétiques.

Pour ne pas avoir un modèle avec un biais pour le classe Non Churn, nous étudierons comme solution la technique **SMOTE (Synthetic Minority Over-sampling Technique).**

Nous allons d'abord commenter les performances du modele sans utilisation de la technique SMOTE.

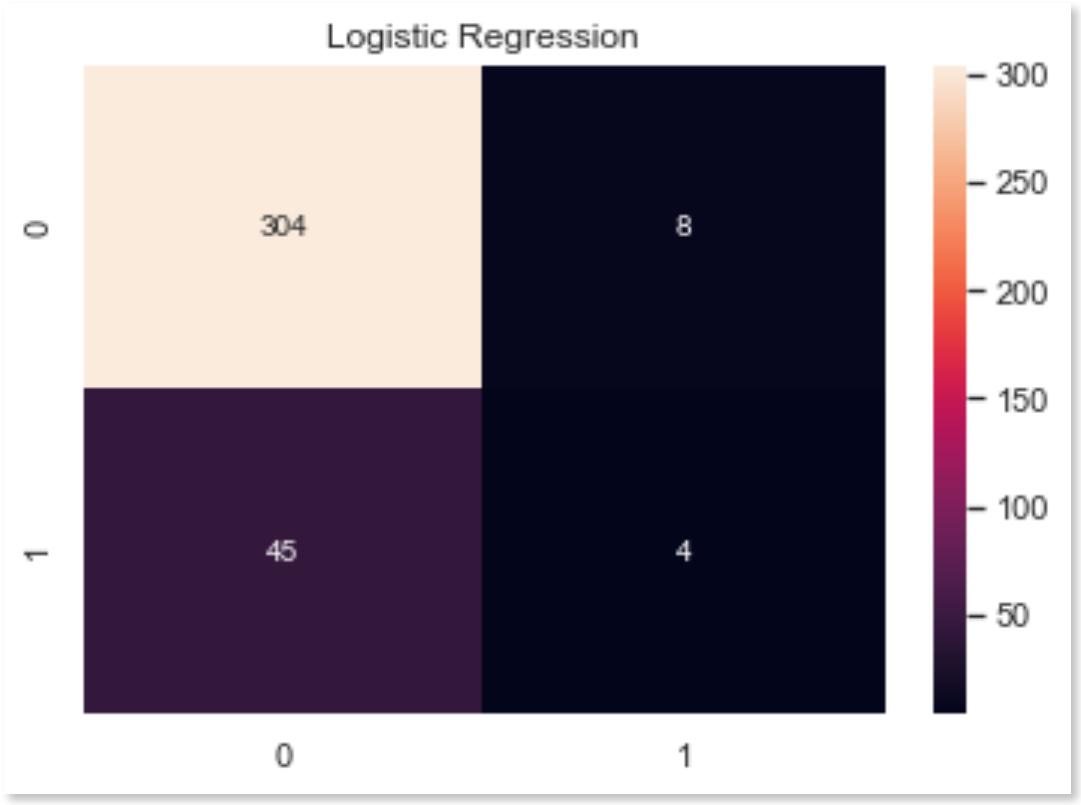


Figure 11.b Matrice de confusion (sans utilisation de la méthode SMOTE)

On observant la matrice de confusion, on se rend compte que seulement 53 individus n'ont pas bien était classe.

Cependant lors de l'apprentissage, notre modèle avait un biais pour la classe Non Churn.

Pour avoir une opinion globale, nous devons étudier les performances du système avec et sans utilisation de la technique SMOTE.

	precision	recall	f1-score	support
Non Churn	0.87	0.97	0.92	312
Churn	0.33	0.08	0.13	49
accuracy			0.85	361
macro avg	0.60	0.53	0.53	361
weighted avg	0.80	0.85	0.81	361

Figure 2.b Performance du modele (sans utilisation de la méthode SMOTE).

	precision	recall	f1-score	support
Non Churn	0.91	0.77	0.83	312
Churn	0.26	0.51	0.34	49
accuracy			0.73	361
macro avg	0.58	0.64	0.59	361
weighted avg	0.82	0.73	0.77	361

Figure 3.b Performance du modele (avec utilisation de la méthode SMOTE).

On en conclut que nous perdons en qualité de reconnaissance, mais nous avons un meilleur système de reconnaissance pour la classe Churn en utilisant la méthode SMOTE.

Cependant, le modele reconnait mieux les individus non churn que les individus churn.

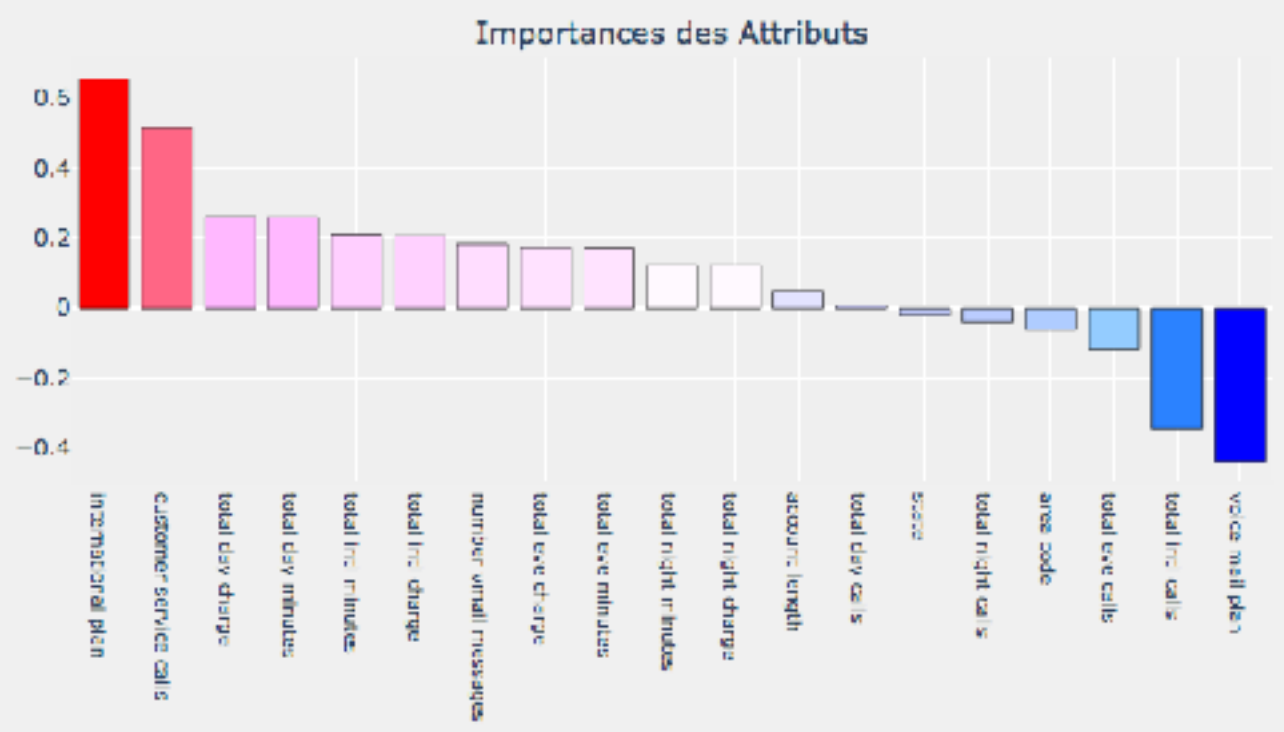


Figure 12.b Classement des attributs les plus importants

Les attributs les plus important sont :

- **International Plan**
- **Customer Service Calls**
- **Voice Mail Plan**
- **Total Intl Calls**

III.Arbre de Décision

Les arbres de décision (AD) sont une catégorie d'arbres utilisée dans l'exploration de données et en informatique décisionnelle. Ils emploient une représentation hiérarchique de la structure des données sous forme des séquences de décisions (tests) en vue de la prédiction d'un résultat ou d'une classe. Chaque individu (ou observation), qui doit être attribué(e) à une classe, est décrit(e) par un ensemble de variables qui sont testées dans les nœuds de l'arbre. Les tests s'effectuent dans les nœuds internes et les décisions sont prise dans les nœuds feuille.

Algorithme de classification: CART

Pour l'étude du modèle Churn, nous allons utiliser l'algorithme de CART afin de générer l'arbre de décision pour nos deux modèles de données.

L'algorithme de construction de l'arbre de CART se repose sur 3 méthodes principales :

- Construction de l'arbre maximal
- Élagage
- Sélection final

Cette méthode produit un arbre de décision binaire, par la division successive des données en 2 sous-dataset au départ elle n'applique pas une règle d'arrêt.

C'est juste après qu'on applique un élagage pour obtenir la meilleur sélection.

1. Client ayant souscrit à un abonnement mobile

Dans un premier temps, étudions les résultats du rapport de classifications.

DecisionTreeClassifier(criterion = 'gini')

Score du test : **0.67**

Air sous la courbe : **0.68**

Figure 16.a Arbre de décision complet

	precision	recall	f1-score	support
Non Churn	0.72	0.43	0.54	140
Churn	0.65	0.86	0.74	170
accuracy			0.67	310
macro avg	0.69	0.65	0.64	310
weighted avg	0.68	0.67	0.65	310

Tableau 3.a Performance du modèle

Dans cette première approche, on a un arbre de décision complet c'est-à-dire de profondeur maximale.

Ce modèle nous fourni un taux d'erreur de 33%, ce qui nous donne un arbre qui n'est pas bon en terme de prédiction.

On remarque que dans notre modèle il n'y a pas de feuilles pures(ce qui nous donnes un taux de confiance de 100%).

Les premiers nœuds représentent les attributs les plus corrélés avec la target(attribut Class) comme nous le montre la matrice de corrélation (vu dans les chapitre précédent).

Comme racine nous avons l'attribut '**Aggregate_Total_Rev**', c'est sur cette attribut que notre algorithme s'appuie pour donner la meilleure division.

Pour ce premier arbre on va pas l'analyser en détail, car le taux de réussite n'est pas assez bon. En revanche on va procéder à l'élagage de ce dernier pour obtenir de meilleurs résultats.

Dans un second temps, interprétons les différents résultats.

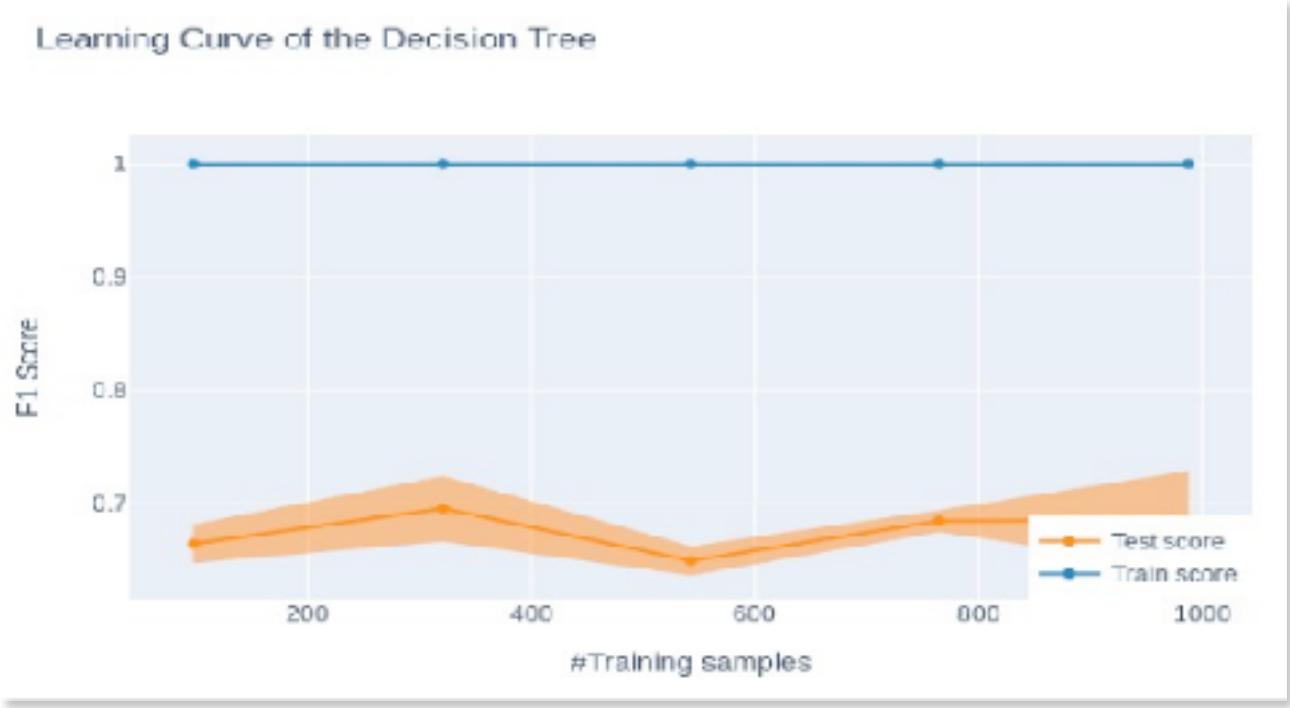


Figure 17.a Courbe d'apprentissage des abonnements mobiles

L'arbre de décision n'est pas ajusté, car le score d'entraînement est supérieur au score de test d'environ 0,3.

	predicted no churn	predicted churn
true no churn	0.61	0.39
true churn	0.34	0.67

Tableau 4.a Matrice de Confusion

La matrice de confusion nous indique qu'il y a un biais qui tend vers la prédiction des désabonnement cela en faveur des clients 'non churn'.

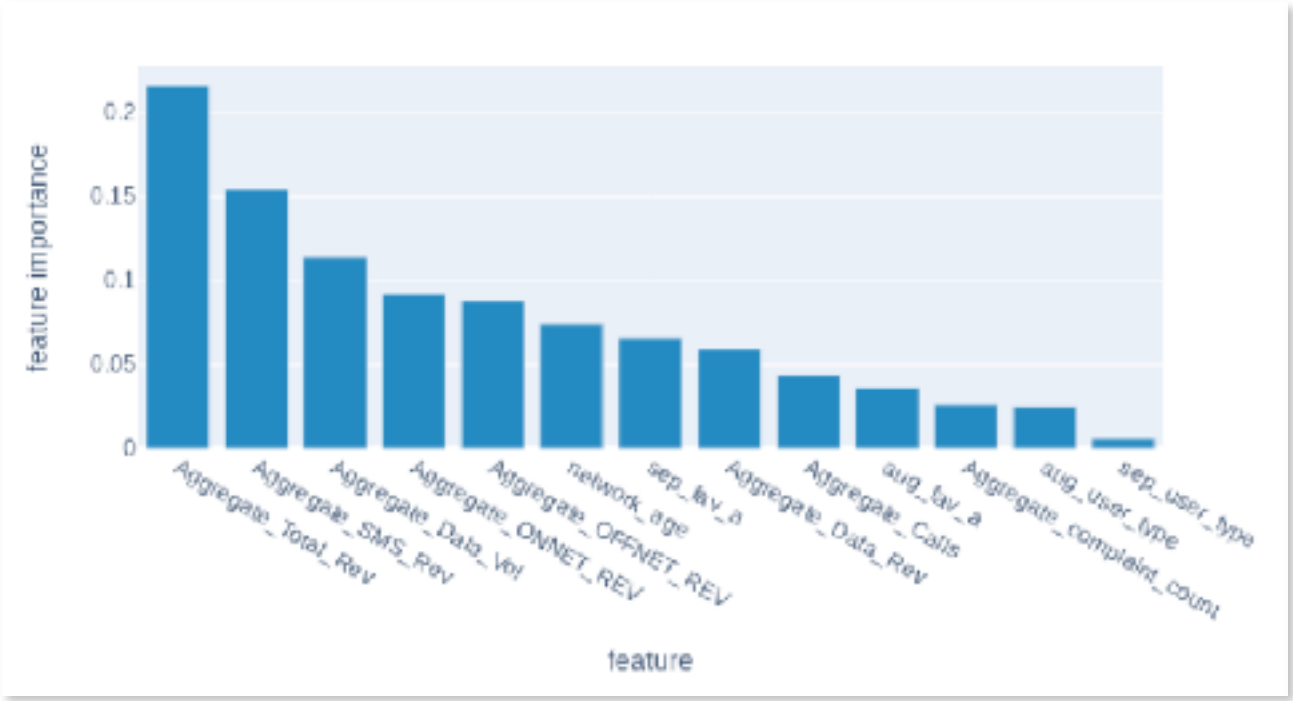


Figure 18.a Classement des attributs les plus important pour l'arbre de décision.

On peut observer que les 3 premiers attributs sont plus importants que les autres, mais on ne peut pas tirer une conclusion car on a vu que ce premiers modèle n'est pas bon.

Pour conclure, avec un arbre de décision complet notre modèle nous fourni un score de 0.67, qui est un score mauvais pour la classifications des abonnées on a vu aussi que le modèle n'est pas du tout ajusté.

Elagage avec l'arbre de décision

Dans cette partie on va utiliser l'arbre obtenu précédemment, on va procéder à l'élagage.

Pour cela, on va donner à notre fonction **DecisionTreeClassifier(criterion= "gini")** 2 paramètres en plus:

min_weight_fraction_leaf: La fraction pondérée minimale de la somme totale des poids (de tous les échantillons d'entrée) devant se trouver à une feuille d'un nœud.

max_depth: La profondeur maximale de l'arbre.

Nous allons utiliser la fonction DTC sur une **max_depth** de 0 à 12 (nombre d'attributs) et **min_weight_fraction_leafentre** [0.01, 0.02, …,0.48,0.5].

On obtiens ainsi les résultats suivants.

Score du test: **0.72**
Air sous la courbe: **0.68**

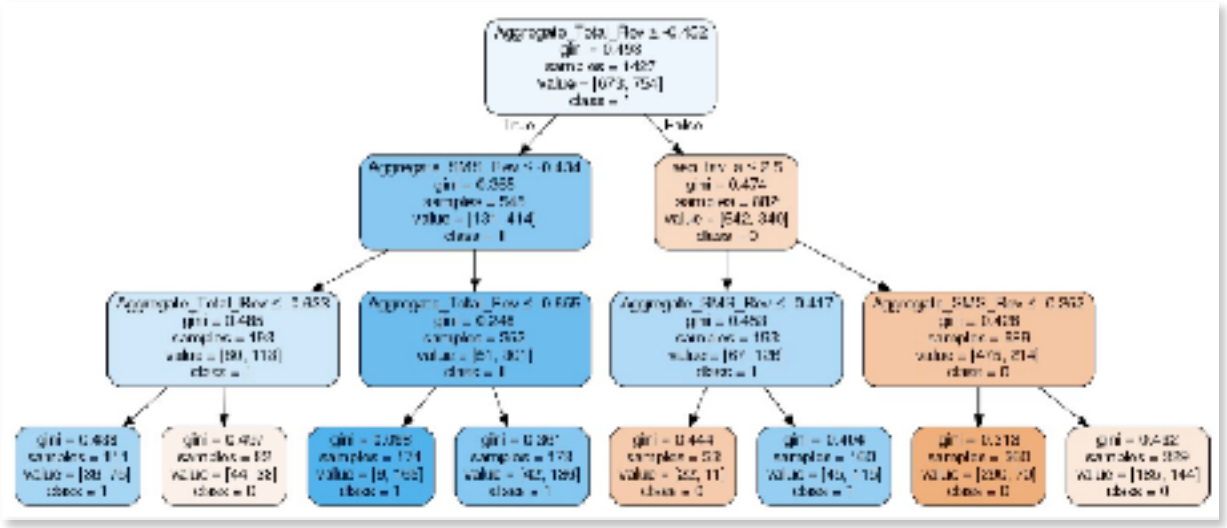


Figure 19.a Arbre de décision avec élagage sur les abonnements téléphoniques

Passons à l'interprétations des résultats:

	predicted no churn	predicted churn
true no churn	0.75	0.25
true churn	0.30	0.70

Figure 5.a Résultat de la matrice de confusion des abonnements mobiles.

Les résultats de la matrice de confusion nous montre un léger biais qui tends vers la prédiction des désabonnements, ce qui donne une faible prédiction des clients qui fidèles.

Figure 20.a courbe d'apprentissage des abonnements mobiles avec élagage.

La courbe d'apprentissage sur les abonnements mobiles, nous indique qu'on a une croissance cela selon le nombre d'individus étudiés (la longueur de l'échantillon) avec un sommet de 765 individus.

Par l'analyse de ces 2 courbes, on peut déduire qu'il y a un intervalle entre (543;765) les deux augmentent et nous fournisse le meilleur score.

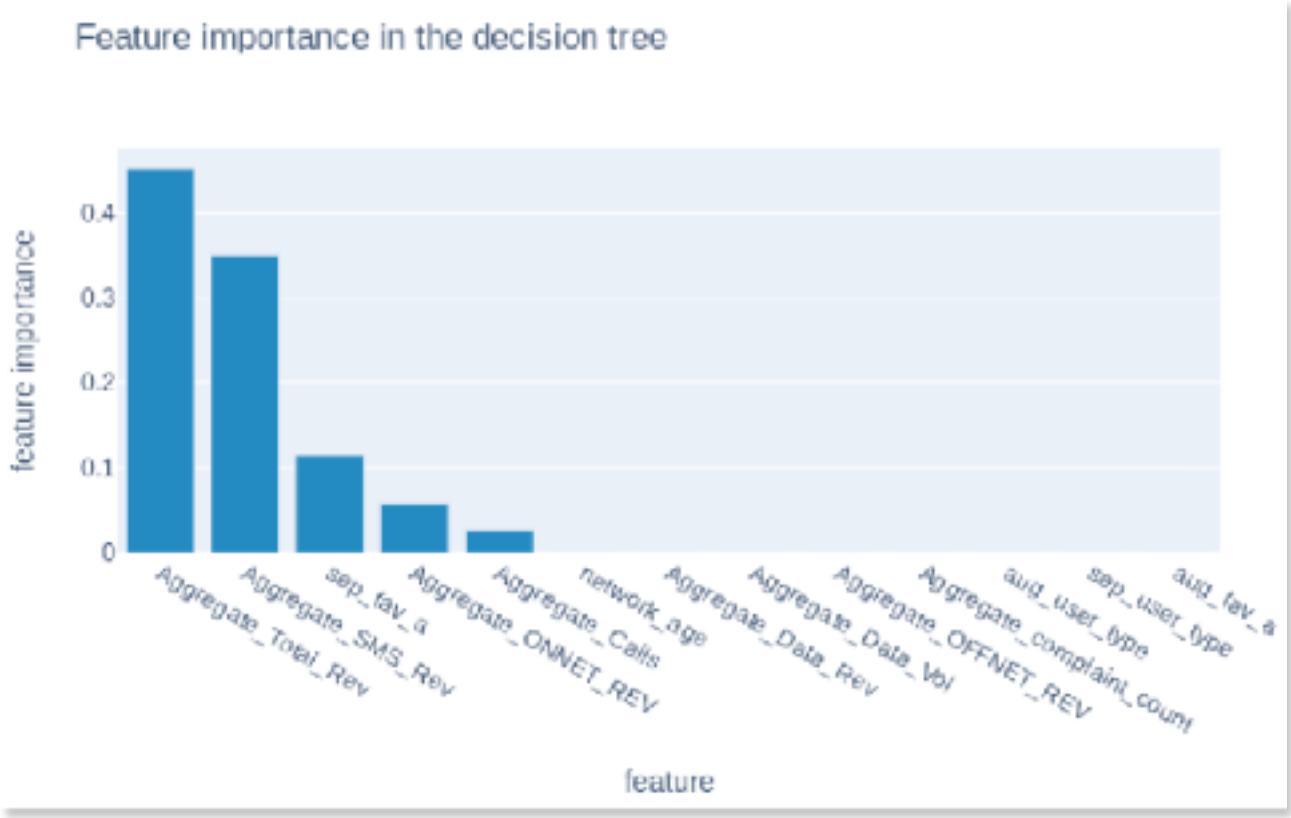


Figure 21.a Attributs les plus importants pour l'arbre de décision avec élagage.

L'histogramme ci-dessous nous montre les attributs les plus importants sur lesquels l'arbre de décision s'appuie.

Nous allons nous restreindre:

- Aggregate_Total_Rev
- Aggregate_SMS_Rev
- sep_fav_a
- Aggregate_ONNET_REV
- Aggregate_Calls

En conclusion, après élagage de l'arbre on a amélioré le score de 0,05 unité et on a réduit le taux d'overfitting mais le modèle n'est pas assez bon car il nous fournit un score de 0,72 qui n'est pas assez bon pour la classification.

Se restreindre sur les attributs les plus importants.

Grace aux résultats obtenus avec l'histogramme des attributs importants obtenu précédemment, on construit l'arbre de décision en utilisant ces attributs (features).

Les résultats obtenus sont:

Score du test: **0.78**
Air sous la courbe: **0.77**

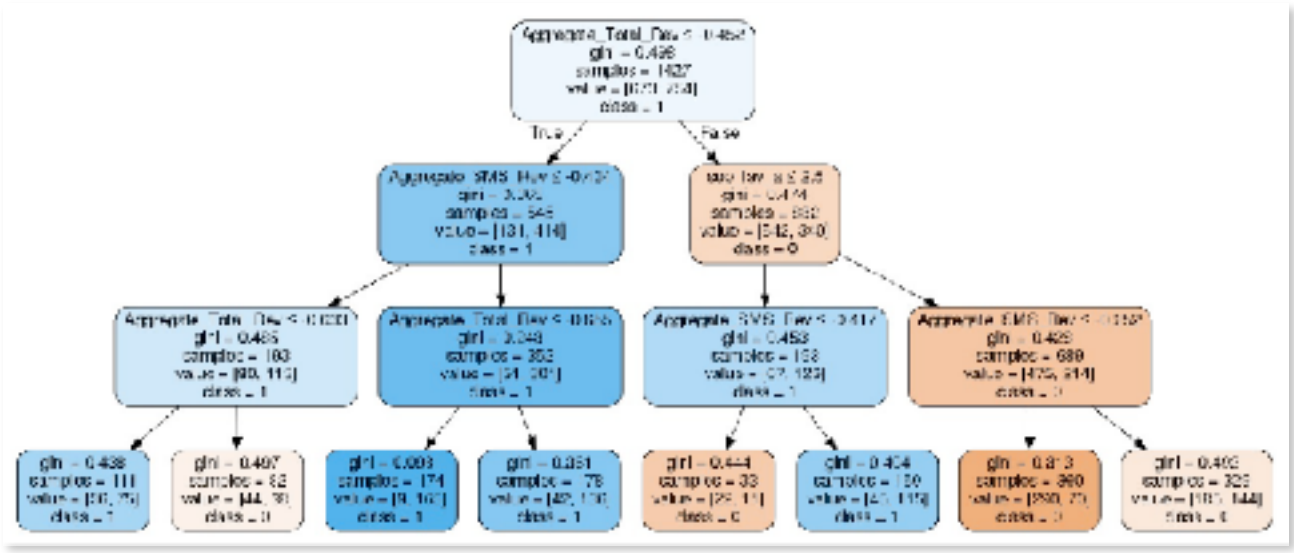


Figure 22.a Arbre de décision pour les abonnements internet avec les attributs restreints.

	precision	recall	f1-score
Non Churn	0.76	0.83	0.79
Churn	0.80	0.73	0.76
accuracy			0.78
macro avg	0.78	0.78	0.78
weighted avg	0.78	0.78	0.78

Tableau 6.a. Performance du modèle

Interprétons les différents résultats:

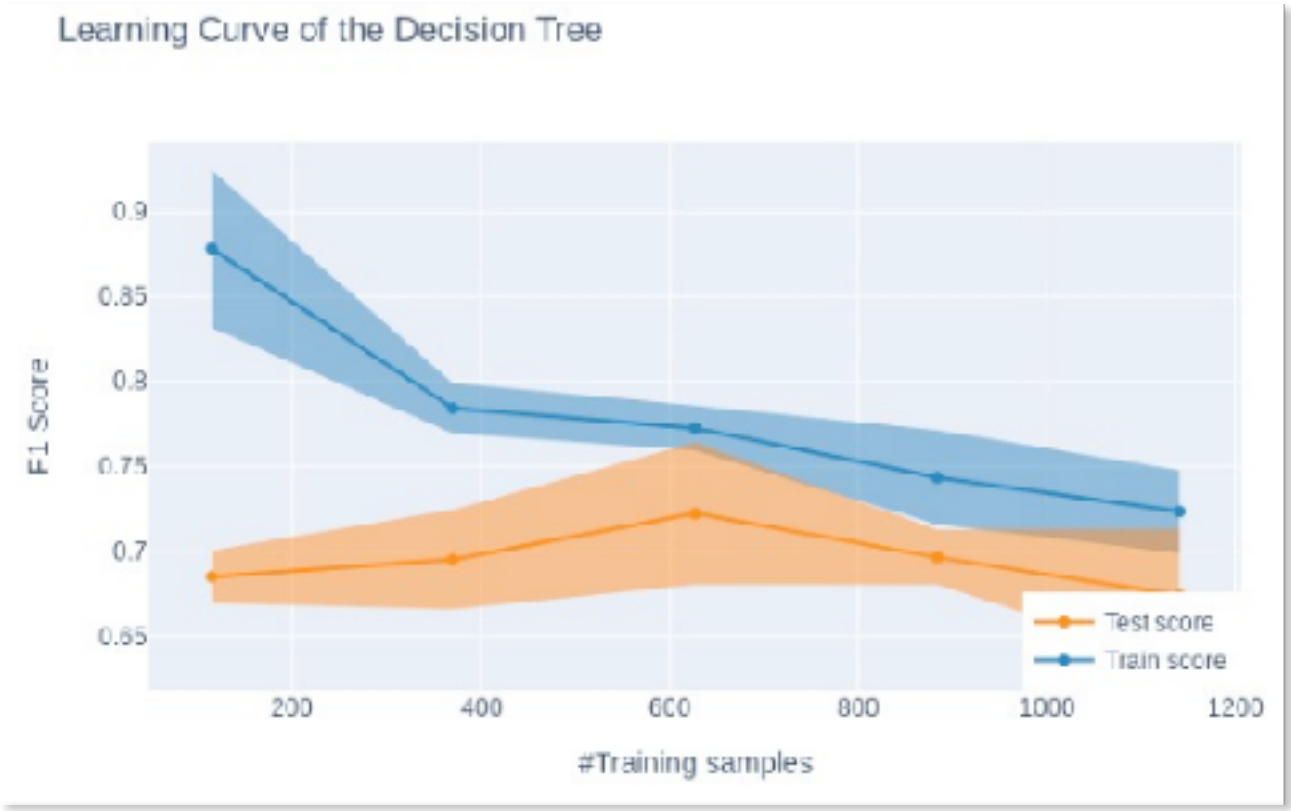


Figure 23.a Courbe d'apprentissage des abonnements mobiles avec les meilleurs attributs

À partir de 672 l'overffiting entre le score d'entraînement et de test se réduit et il est linéairement ajusté, on remarque que la croissance du score d'entraînement jusqu'à un certains nombre d'individus(672), puis sa décroissance ce qui nous indique que le modèle sur apprend.

	predicted no churn	predicted churn
true no churn	0.80	0.20
true churn	0.31	0.69

Tableau 7.a Matrice de confusion

La matrice de confusion nous indique qu'il y a un biais qui tends vers la prédiction des clients «no churn» ce qui impact sur la prédiction des clients «no churn».

Pour conclure, en utilisant les meilleurs attributs on a obtenu un score de 0.78, on a réduit le taux d'overfitting entre le score test et celui de l'entraînement qui sont linéairement ajusté à partir de 672 individus.

De ce qui est de l'exactitude des prédiction le modèle nous ne fournit pas un résultats exactes, car il a tendance à prédire les désabonnements que ce qui impact sur les résultat des clients qui se désabonnes.

2. Client ayant souscrit à un abonnement internet

Pour obtenir l'arbre de décision on utilise la fonction:
DecisionTreeClassifier(criterion="gini")

Score du test **: 0.94**
Air sous la courbe **: 0.82**

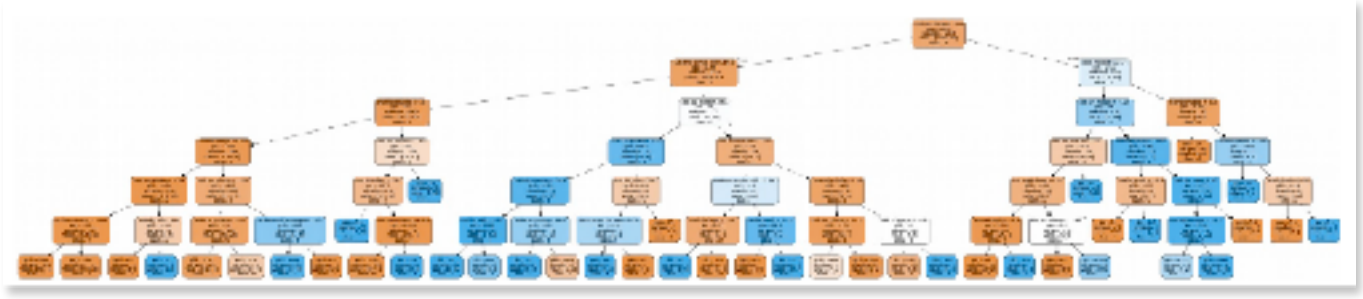


Figure 13.b Arbre de décision des abonnements internet

	precision	recall	f1-score
Non Churn	0.95	0.99	0.97
Churn	0.90	0.66	0.76
accuracy			0.94
macro avg	0.92	0.82	0.86
weighted avg	0.94	0.94	0.94

Tableau 2.b Performance du modèle

	predicted no churn	predicted churn
true no churn	0.96	0.04
true churn	0.16	0.84

Tableau 3.b Matrice de confusion

La matrice de confusion nous indique qu'il y a un biais qui tend vers la classe «non churn», elle nous indique que notre modèle est bon mais il y a plusieurs individus qui sont mal classés 12%, et l'air sous la courbe qui nous indique le taux des vrais positifs est de 0.82 est un assez bon résultats.

Figure 14.b Courbe d'apprentissage abonnement Internet

Plus la taille de l'échantillon est grand plus le score de test augmente et le score d'entraînement diminue, ce qui réduit loverfitting.

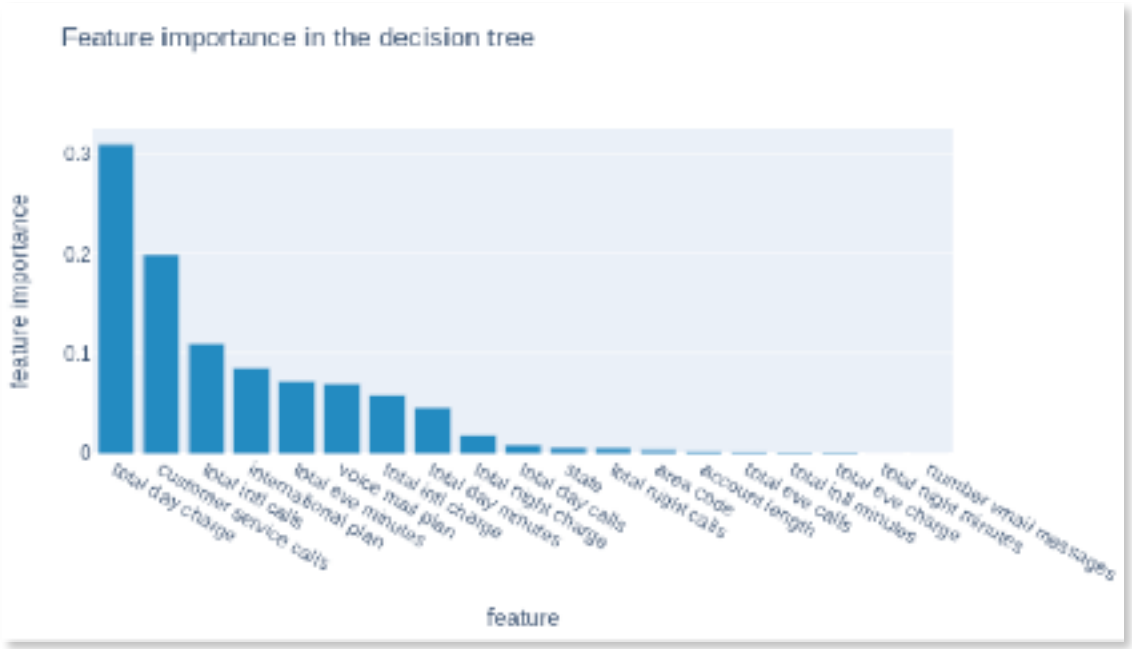


Figure 15.b Attribut les plus importants pour les abonnements internet

On conclut donc que, l'arbre de décision nous donne un bon modèle avec un score de 0.94, avec une bonne classifications.

Lorsqu'on se restreint aux attributs les plus importants on obtient pratiquement les mêmes résultat c'est-à-dire 0.94.

IV.Random Forest & Boosting

1.Random Forest

L'algorithme des «forêts aléatoires» (ou Random Forest parfois aussi traduit par forêt d'arbres décisionnels) est un algorithme de classification qui réduit la variance des prévisions d'un arbre de décision seul, améliorant ainsi leurs performances. Pour cela, il combine de nombreux arbres de décisions dans une approche de type bagging.

Le mot Bagging est une contraction de Bootstrap Aggregation. Le bagging est une technique utilisée pour améliorer la classification notamment celle des arbres de décision, considérés comme des « classifieurs faibles », c'est-à-dire à peine plus efficaces qu'une classification aléatoire.

A. Client ayant souscrit à un abonnement mobile

On a utilisé un classificateur de forêt aléatoire avec 100 arbres et une profondeur maximale d'arbres de 20.

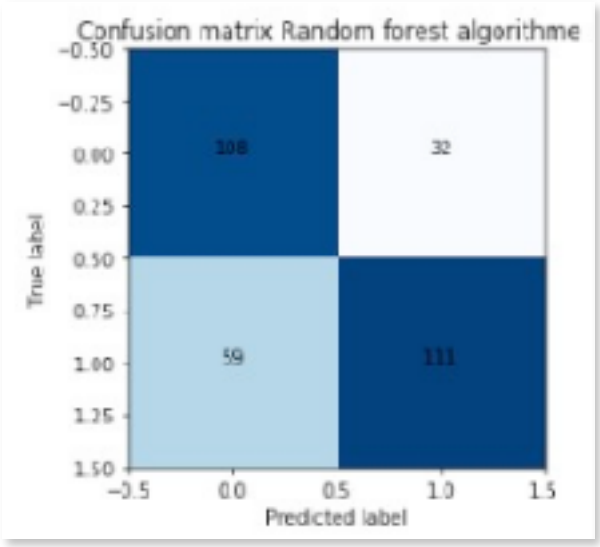


Figure 24.a Matrice de confusion

Par lecture de la matrice de confusion, on s' aperçoit que notre modele est bon mais pas excellent plusieurs individus sont mal classé (91).

Comme précédemment analysons les performances du modele à partir du rapport de classification.

	precision	recall	f1-score	support
Non Churn	0.66	0.74	0.70	140
Churn	0.76	0.69	0.73	170
accuracy			0.71	310
macro avg	0.71	0.71	0.71	310
weight.avg	0.72	0.71	0.71	310

Tableau 8.a. Performance du modele.

Les mesures de performance sont assez bonnes pour prédire l'attrition client. Mais pas excellent, en effet nous avons un recul moyen égale a 0.65, une précision moyenne à 0.65 et une accuracy égale à 0.71.

Donc on ne peut pas conclure que le modèle est un bon classifieur.

Figure 25.a Tree of Random Forest Algorithme.

On remarque qu'à partir de ce graphe, chaque sample est bien classé.

B. Client ayant souscrit à un abonnement internet

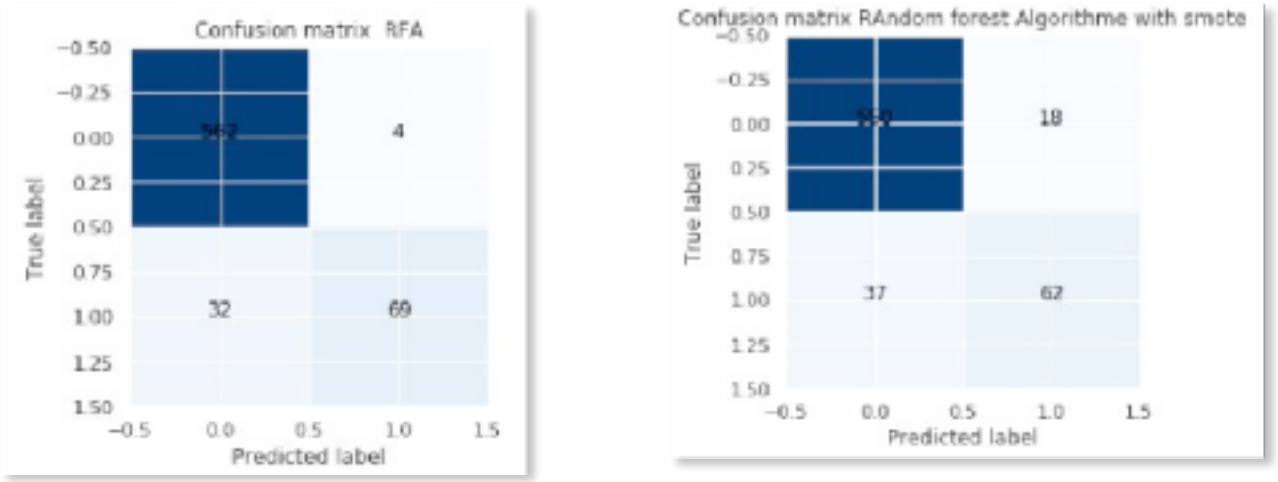


Figure 16.b Matrice de confusion sans la méthode SMOTE (à gauche) et avec la méthode SMOTE (à droite).

On a obtenus une accuracy environ égale 0.95 sans la méthode SMOTE et en utilisant la méthode SMOTE on a une accuracy égal à 0.92.

On constate que l'accuracy est un peu plus élevée dans le dataset déséquilibré que dans le dataset équilibre, nous retrouvons le même comportement dans les forets de décision.

Le jeu de données déséquilibre est plus claire et facile à traiter que dans le dataset équilibré et cette différence est du au fait que notre modele est biaisé, on apprend plus sur la classe Non Churn et on arrive donc mieux à les classifier.

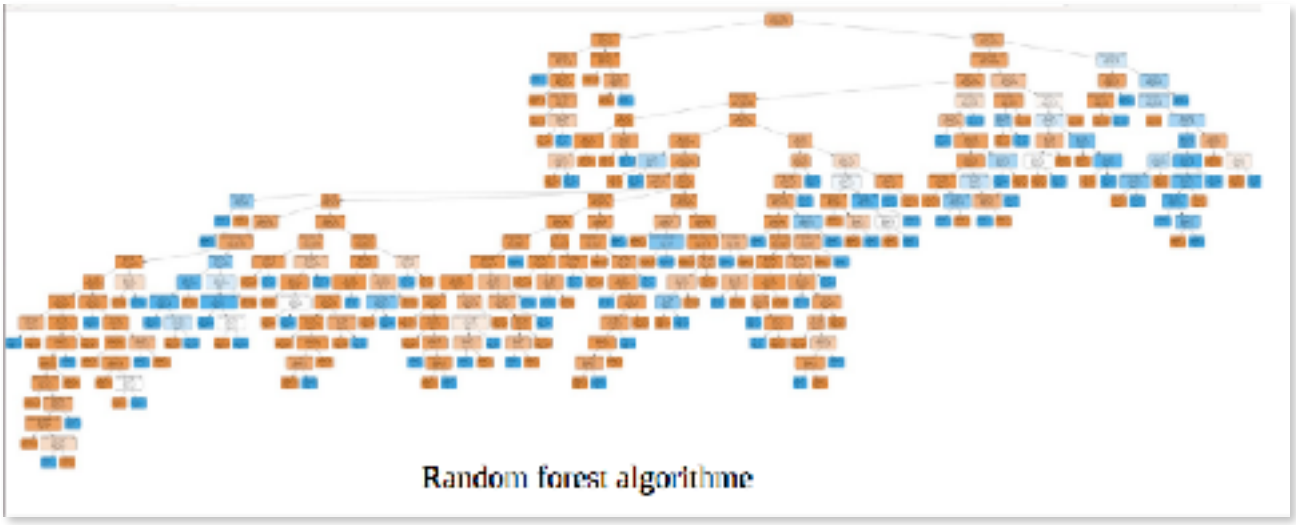


Figure 17.b Tree of Random Forest Algorithme sans la methode SMOTE.

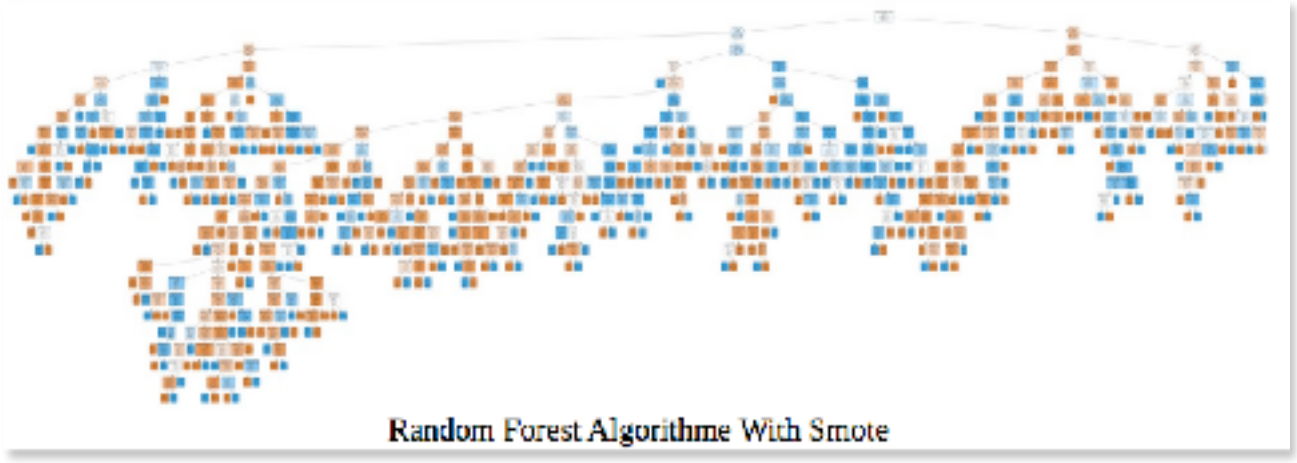


Figure 18.b Tree of Random Forest Algorithme avec la methods SMOTE.

Concernant l'étude de modèle on remarque qu'avec une accuracy de 95% et 92% on peut prédire avec précision l'attrition client. Cela ne permet de dire que ce modèle est bon pour la classification des données.

2. Boosting

En apprentissage automatique, le boosting est un méta-algorithme d'ensemble pour principalement réduire le biais, ainsi que la variance de l'apprentissage supervisé, et une famille d'algorithmes d'apprentissage automatique qui convertissent les apprenants faibles en apprenants forts.

A. Client ayant souscrit à un abonnement mobile

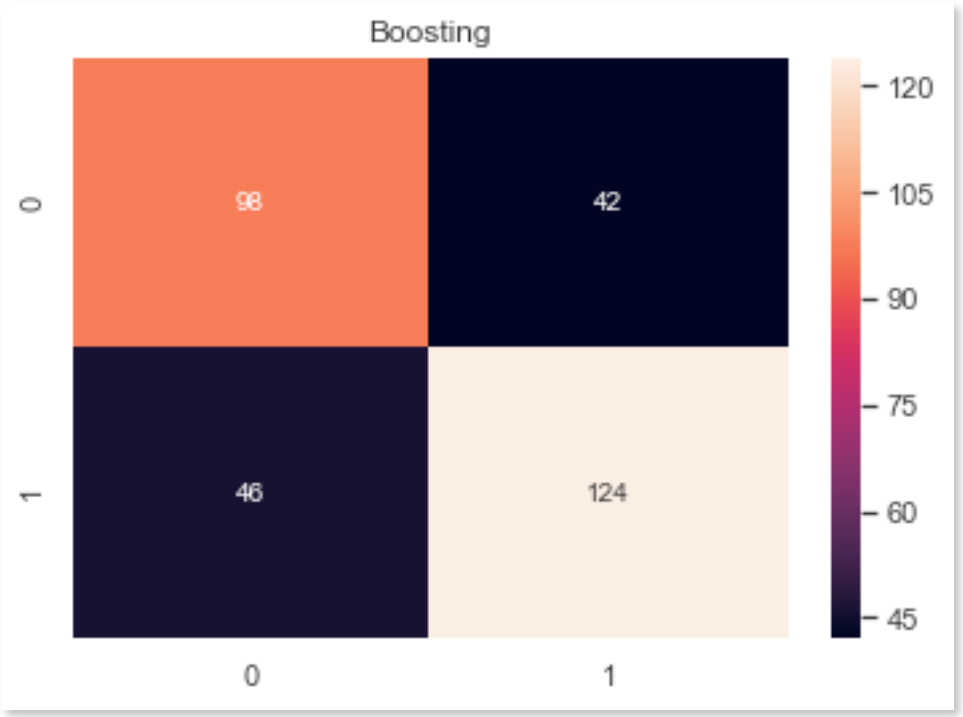


Figure 26.a Matrice de confusion

A partir de la matrice de confusion, on s 'aperçoit que notre modele est assez bon plusieurs individus sont mal classé (88).

	precision	recall	f1-score	support
Non Churn	0.68	0.70	0.69	140
Churn	0.75	0.73	0.74	170
accuracy			0.72	310
macro avg	0.71	0.71	0.71	310
weighted avg	0.72	0.72	0.72	310

Tableau 9.a Performance du modele.

On s'aperçoit que le modèle est assez bon on a une accuracy égale à 0.72.

Donc, on ne peut pas prédire l'attrition client avec precision.

B. Client ayant souscrit à un abonnement internet

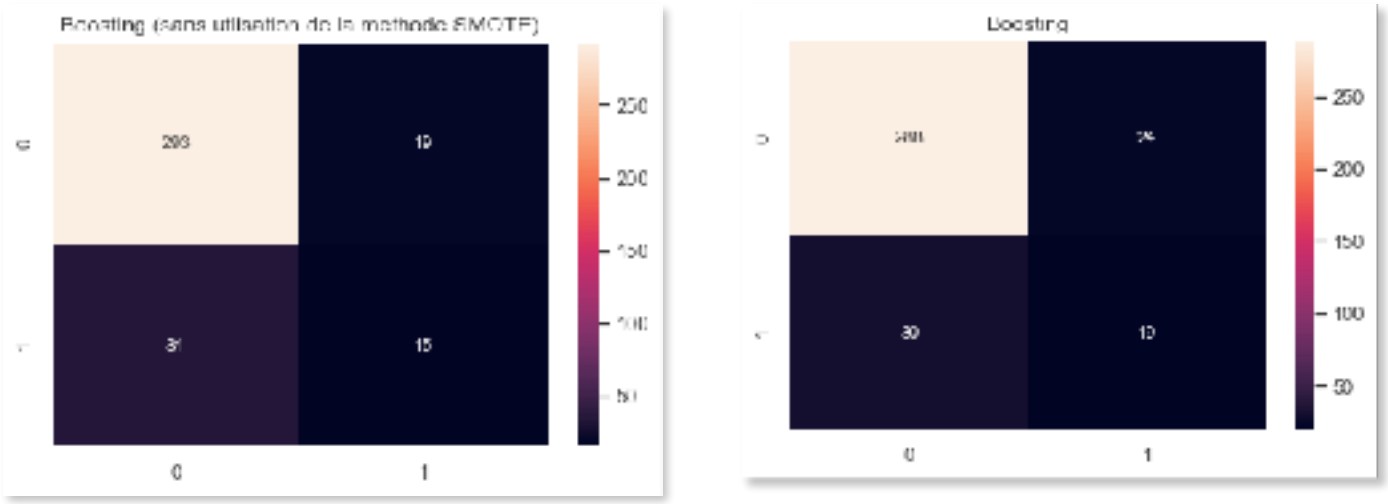


Figure 19.b Matrice de confusion sans la méthode SMOTE (à gauche) et avec la méthode SMOTE (à droite).

On obtient une accuracy pour le modele sans SMOTE 0.87 et avec la méthode SMOTE 0.85, donc on a une bonne accuracy général donc on peut prédire avec précision l'attrition client.

V. Support Vector Machine

Les « supports vectors machines » appelés aussi « maximum margin classifier» sont des techniques d'apprentissage supervisé basées sur la théorie de l'apprentissage statistique ou automatique, destinées a résoudre des problèmes de discrimination.

SVM est utilisé pour résoudre des problèmes de discrimination ,c'est a dire décider a quelle classe appartient un échantillon (classification).

Nous sommes dans un problème de discrimination à deux classes (discrimination binaire) présence ou non de l'attrition client.

A. Client ayant souscrit à un abonnement mobile

1. SVM linear Kernel

La façon la plus simple d'utiliser un SVC est d'utiliser un Kernel linéaire, ce qui signifie que la frontière de décision est une ligne droite (ou un hyperplan dans des dimensions supérieures « vectors support »).

Les kernels linéaires sont rarement utilisés dans la pratique car ils sont plus simple et avec jeux donnés plus grands c'est peu probable de tomber sur un classifieur linéaire qui va représenter bien nos jeux donnés.

Interprétation des résultats obtenus :

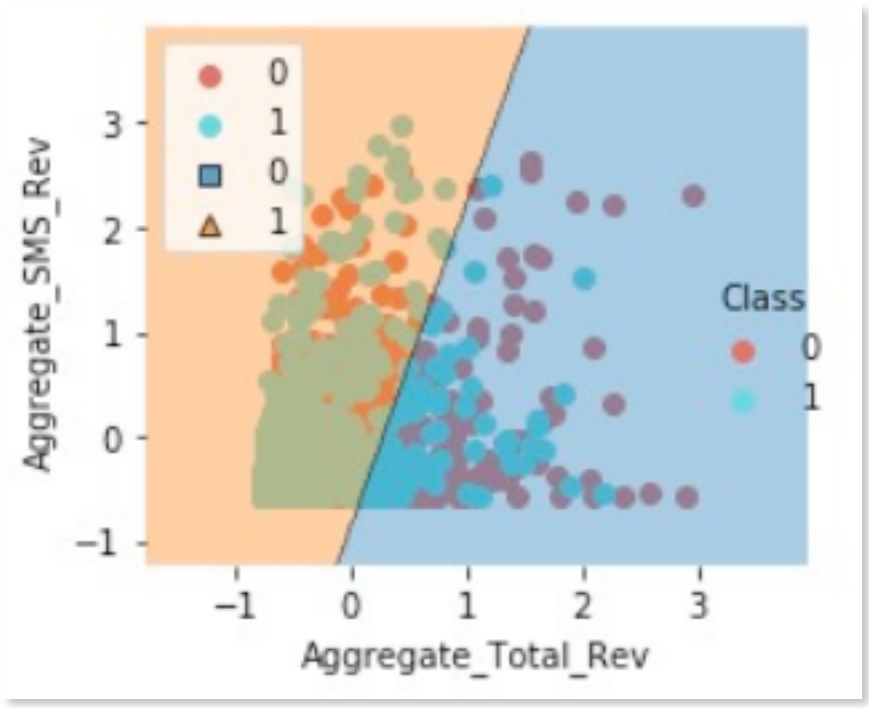


Figure 27.a Région de décision

On remarque que la droite ne sépare pas les données d'une façons explicite, donc on suppose que ce n'est un bon classifieur pour notre jeu de données.

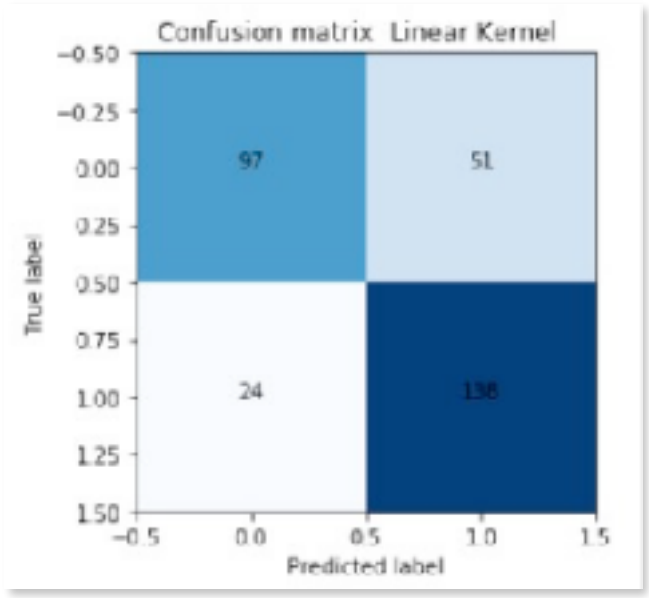


Figure 28.a Matrice de confusion

D’après la matrice de confusion on obtient:

Accuracy **: 0.7580645161290323**
Precision **: 0.8518518518518519**
Recall **: 0.8518518518518519**

On a obtenu une accurracy de 75%, ce qui n’est pas assez satisfaisant pour dire que ce modèle est bon par ailleurs si on analyse le graphe on regarde bien que les données ne sont pas bien séparées et on peut pas prédire si le client est un churn/non churn avec precision.

Pour réaliser un bon modèle qui exprime notre jeux de données on ne va pas se limiter de SMV linéaire mais on va utiliser different méthode de kernel parce que notre jeu de données n’est pas séparable d’une façon linéaire.

Kernel méthode ne permet pas de présenter les données sous un autre espace sans se soucier de connaître la façon de la transformation de données dans cet espace et parmi les méthodes de kernel on a **:** RBF, poly, Sigmoid,… .

2. RBF Kernel (Radial Basis function)

Les paramètres SVC pour le kernel RBF sont **:**

gamma: Est un paramètre du noyau RBF. Lorsque le gamma est faible, la «courbe» de la frontière de décision est très faible et donc la région de décision est très large. Lorsque le gamma est élevé, la «courbe» de la frontière de décision est élevée.

C: Est un paramètre de l'apprenant SVC et est la pénalité pour une mauvaise classification d'un point de données. Lorsque C est petit, le classificateur est d'accord avec les points de données mal classés (biais élevé, faible variance). Lorsque C est grand, le classificateur est fortement pénalisé pour les données mal classées et, par conséquent, se penche en arrière pour éviter tout point de données mal classifié (biais faible, variance élevée).

```
ClassfierRBF =SVC( kernel=‘rbf',random_state=0,gamma=0.002,C=1000)
```

Evaluation du modele :

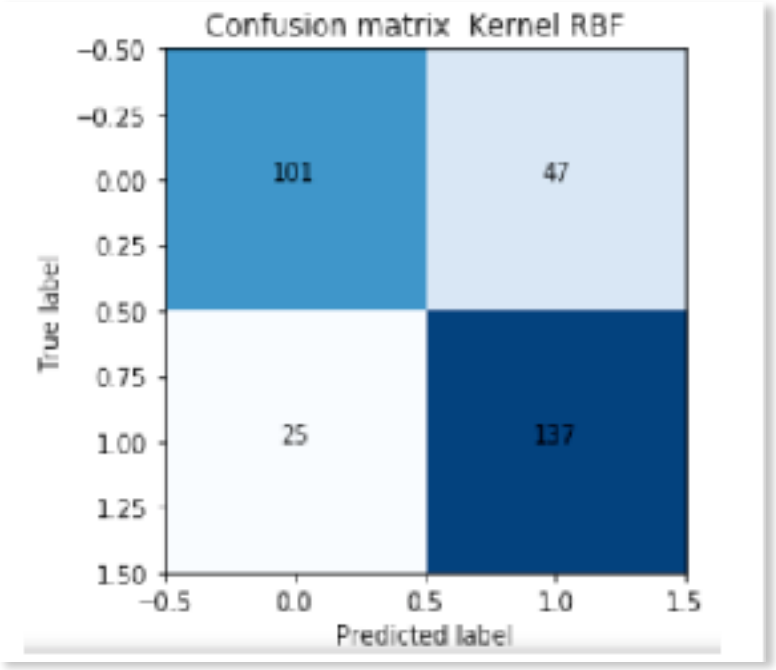


Figure 29.a Matrice de confusion (Kernel RBF)

D'après la matrice de confusion on obtient:

Accuracy : **0.7677419354838709**
Precision : **0.845679012345679**
Recall : **0.845679012345679**

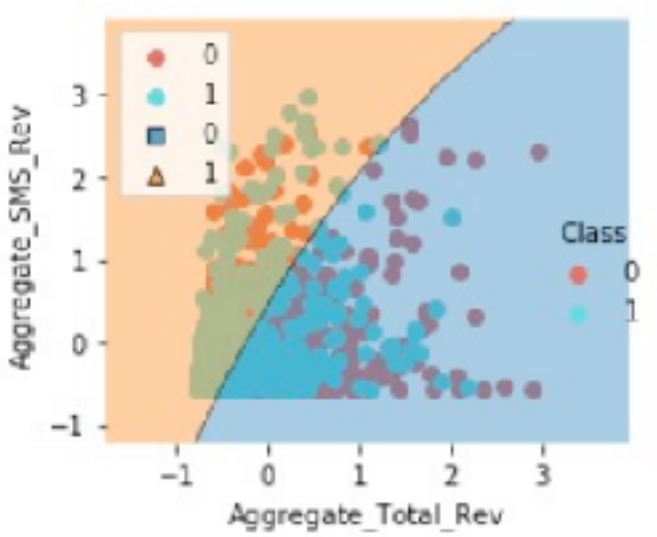


Figure 30.a Region de decision (Kernel RBF)

On a obtenu une accurracy de 77%, ce qui n'est pas assez satisfaisant pour dire que ce modèle est bon par ailleurs si on analyse le graphe on regarde bien que les données ne sont pas bien séparées quoique les données sont séparées par une courbe et on peut pas prédire si le client est un churn/non churn avec precision.

3. Poly Kernel (Polynomiale)

Les paramètres SVC pour le kernel Polynomiale sont :

ClassfierPoly= SVC(kernel='poly',degree=2)

Les paramètres de SVC: **dégrée= dégrée de polynomiale**

Interprétation des résultats :

Accuracy **: 0.7548387096774194**
Precision **: 0.8209876543209876**
Recall **: 0.8209876543209876**

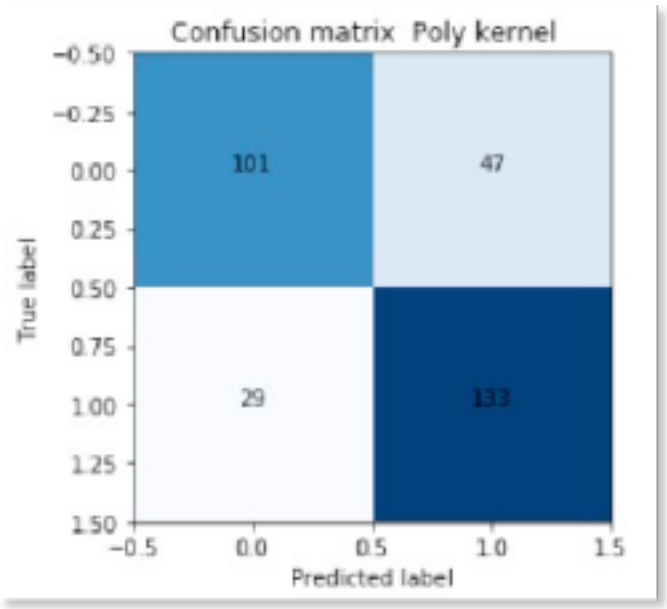


Figure 31.a Matrice de confusion (Kernel Polynomial)

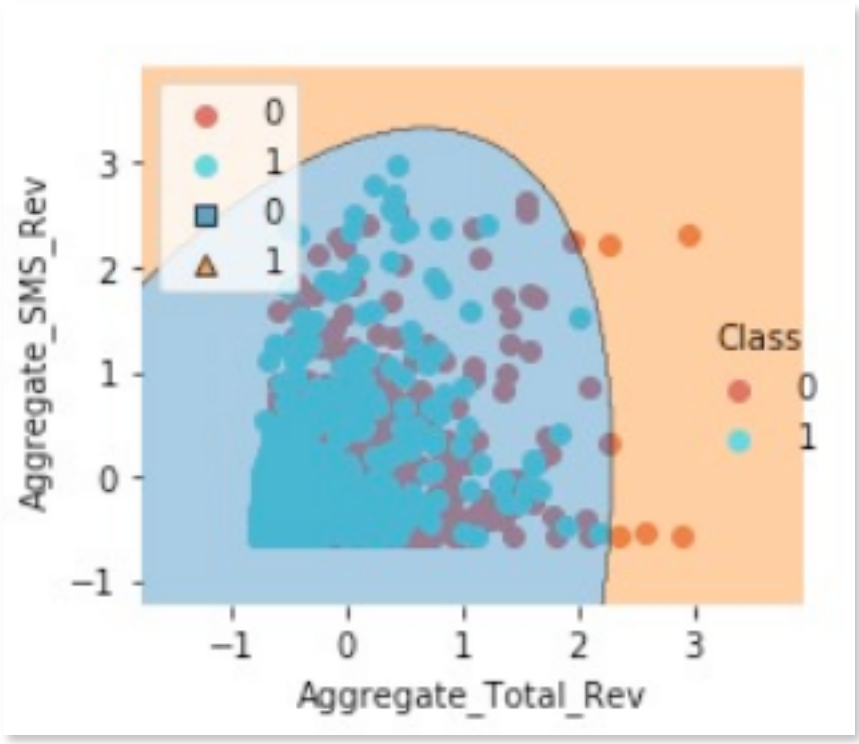


Figure 32.a Region de decision (Kernel Polynomial)

On a obtenu une accurracy de 75 % qui n'est pas assez satisfaisant pour dire que ce modèle est bon par ailleurs si on analyse le graphe on regarde bien que les données ne sont pas bien séparées et on ne peut pas prédire si le client est un churn/non churn avec precision.

4. Sigmoid Kernel (Sigmoid)

Interprétation des résultats :

Accuracy : 0.7225806451612903
Precision : 0.8024691358024691
Recall : 0.8024691358024691

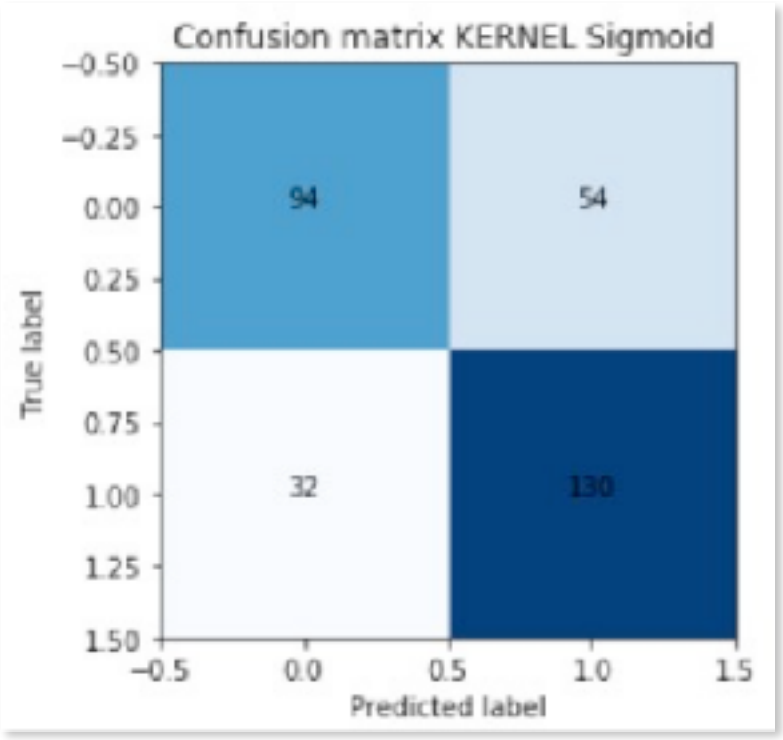


Figure 33.a Matrice de confusion (Kernel Sigmoid)

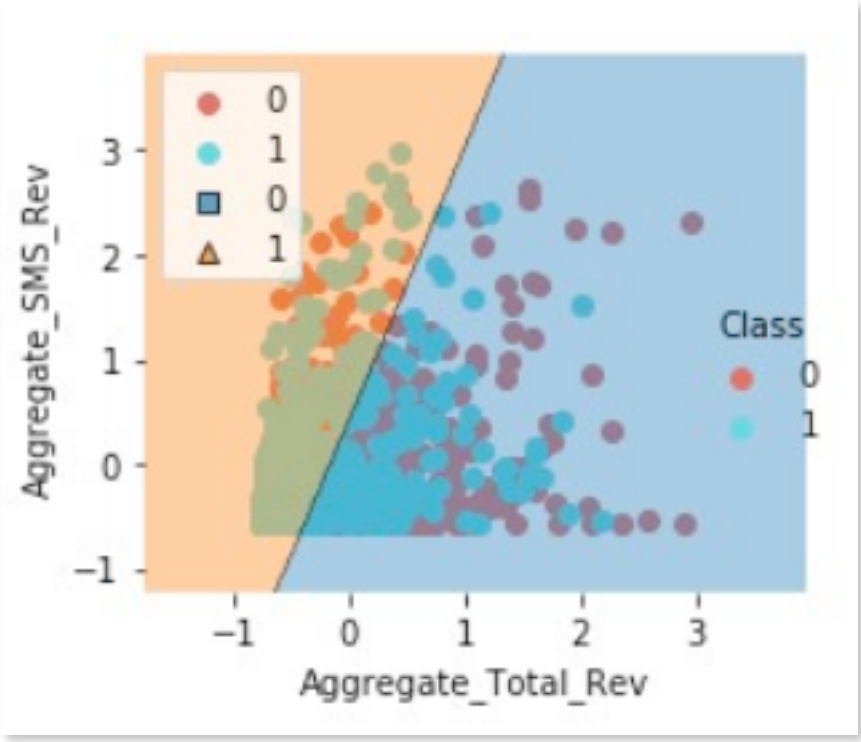


Figure 34.a Region de decision (Kernel Sigmoid)

On a obtenu une accurracy de 72% qui n'est pas assez satisfaisant pour dire que ce modèle est bon par ailleurs si on analyse le graphe, on constate que les données ne sont pas bien séparées au niveau de sigmoid et on peut pas prédire si le client est un churn/non churn avec precision.

Conclusion Global sur SVM :

On constate que les résultats obtenus de notre réalisation de SVM ne sont pas vraiment satisfaisants, car les données ne sont pas bien séparées par apport à leurs classes et le taux de l'accuracy est entre [72%, 76%] ce qui reflète une mauvaise séparation des données.

La meilleure méthode de kernel à partir des résultats est la méthode RBF avec une accuracy à 77%.

B. Client ayant souscrit à un abonnement internet

Remarque : pour les méthodes Linear kernel, RBF, Sigmoid, Poly nous suivront les mêmes étapes que précédemment.

Donc nous allons seulement interpréter les résultats obtenus et conclure entre SVM avec Smote et sans Smote, comparer les modèles SVM et choisir le meilleur qui représente bien la séparation des données.

1. SVM linear Kernel

Dataset Déséquilibré		Dataset Equilibré	
Accuracy:	0.8680659670164917	Accuracy:	0.6543859649122807
Precision:	0.0	Precision:	0.0
Recall:	0.0	Recall:	0.7885304659498208

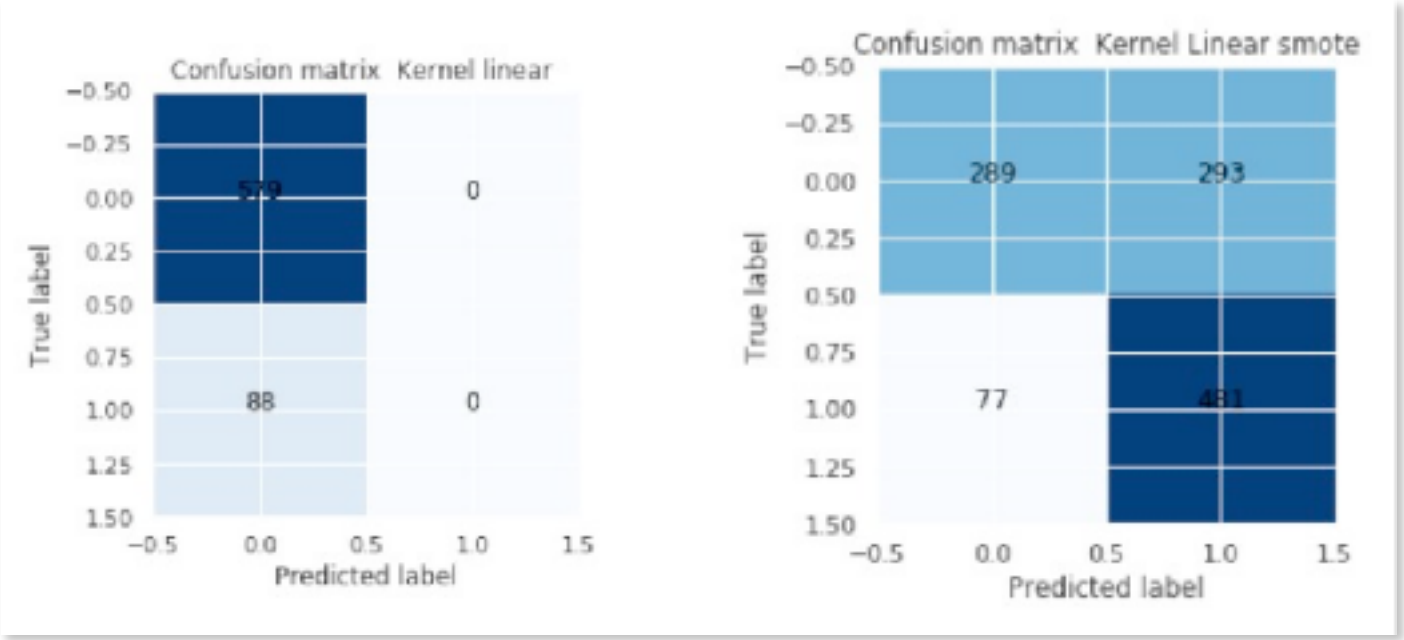


Figure 20.b Matrice de confusion sans la méthode SMOTE (à gauche) et avec la méthode SMOTE (à droite).

On remarque que dans le dataset déséquilibré l'accuracy est à 86% par contre dans la dataset équilibrée l'accuracy et à 65%, cette différence est causée par le principe de la méthode SMOTE vue précédemment.

Concernant l'évaluation des modèles linéaire, on conclut que pour un pourcentage de 86 et 65 n'est pas suffisant pour dire que c'est un bon modèle qui sépare les données donc on peut pas prédire si le client est un churn/non churn avec precision.

2. Kernel RBF

Dataset Déséquilibré

Accuracy: 0.904047976011994
Precision: 0.5227272
Recall: 0.5227272

Dataset Equilibré

Accuracy: 0.875438596491228
Precision: 0.8655913978494624
Recall: 0.8655913978494624

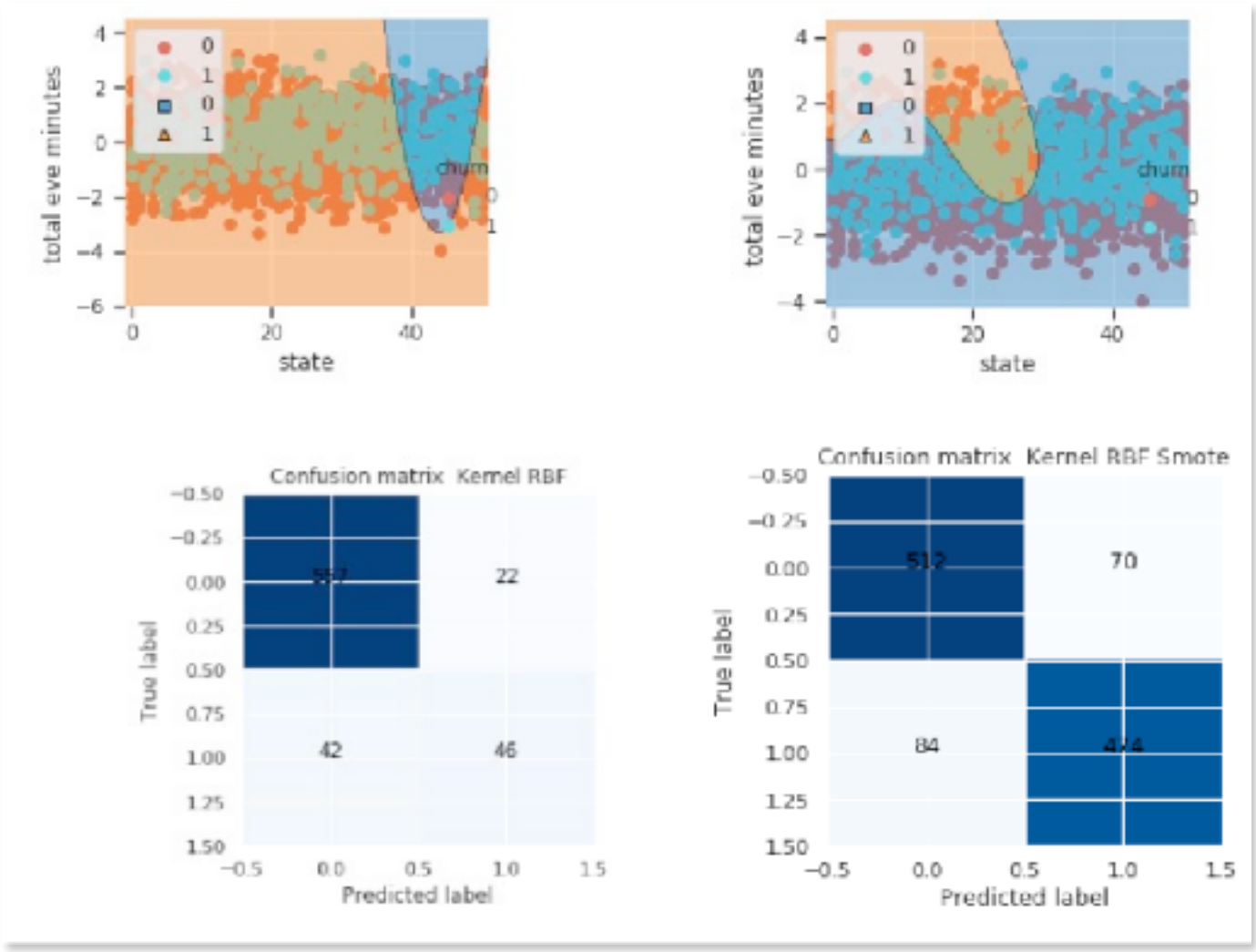


Figure 21.b Région de décisions sans la méthode SMOTE (en haut à gauche) et avec la méthode SMOTE(en haut à droite) et la matrice de confusion sans la méthode SMOTE (en bas à gauche) et avec la méthode SMOTE (en bas à droite).

On remarque que dans le dataset déséquilibré l'accuracy est à 90% par contre dans la dataset équilibrée l'accuracy et à 87%, cette différence est causée par le principe de la méthode SMOTE vue précédemment.

Concernant l'évaluation des modèles non linéaire RBF on conclut que pour un pourcentage de 90% on peut juger que le modèle classifie bien les données et que c'est un bon modèle donc on peut prédire si le client est un churn/non churn avec precision.

3. Kernel Polynomial

Dataset Déséquilibré

Accuracy: 0.8830584707646177
Precision: 0.11363636363
Recall: 0.11363636363

Dataset Equilibré

Accuracy: 0.8131578947368421
Precision: 0.7562724014336918
Recall: 0.7562724014336918

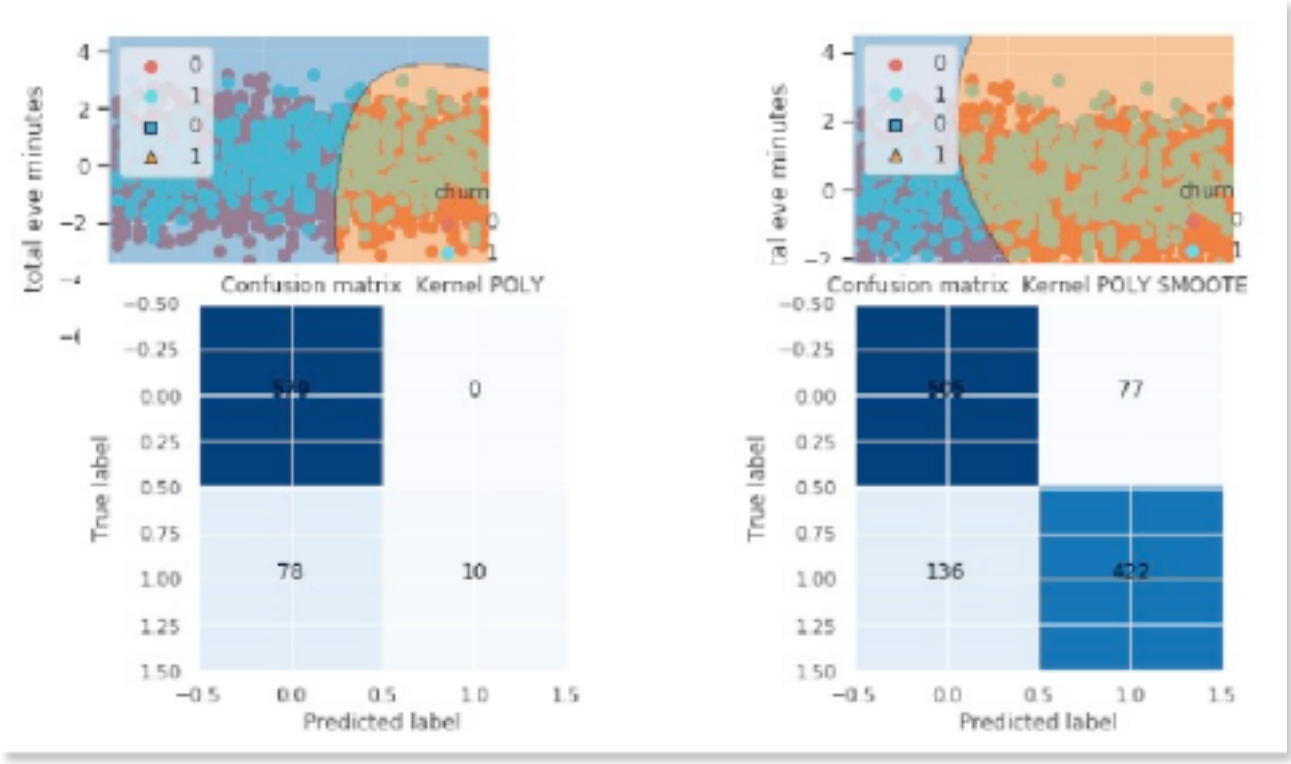


Figure 22.b Région de décisions sans la méthode SMOTE (en haut à gauche) et avec la méthode SMOTE(en haut à droite) et la matrice de confusion sans la méthode SMOTE (en bas à gauche) et avec la méthode SMOTE (en bas à droite).

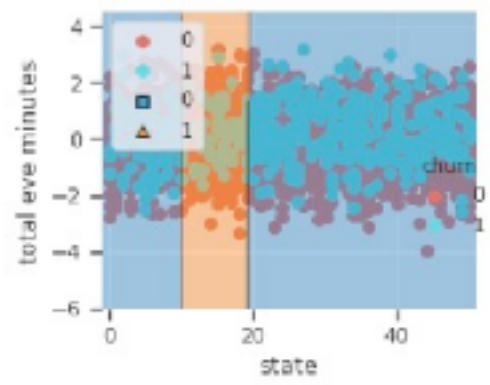
On remarque que dans le dataset déséquilibré l'accuracy est à 81% par contre dans la dataset équilibrée l'accuracy et à 88%, cette différence est causée par le principe de la méthode SMOTE vue précédemment.

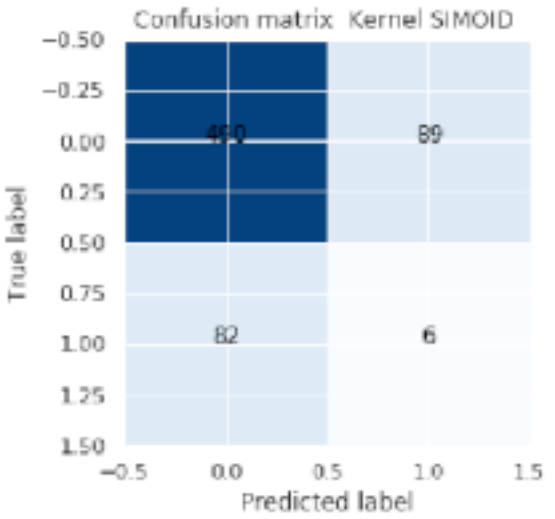
Concernant l'évaluation des modèles non linéaire Polynomiale on conclut que pour un pourcentage de 86 et 81 n'est pas suffisant pour dire que c'est un bon modèle qui sépare les données donc on peut pas prédire si le client est un churn/non churn avec precision.

4. Kernel Sigmoid

Dataset Déséquilibré

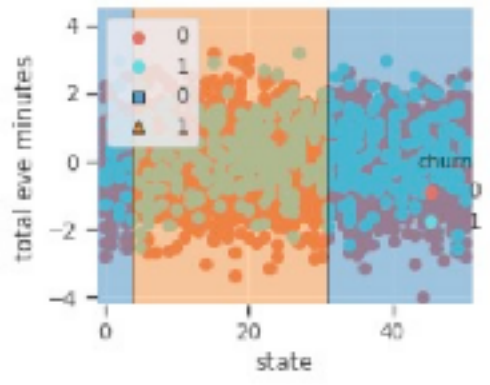
Accuracy: 0.7436281859070465
Precision: 0.0681818
Recall: 0.0681818





Dataset Equilibré

Accuracy: 0.5157894736842106
Precision: 0.525089605734767
Recall: 0.525089605734767



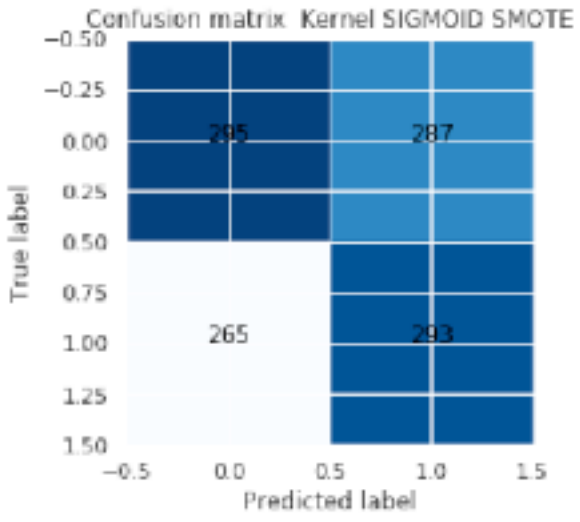


Figure 23.b Région de décisions sans la méthode SMOTE (en haut à gauche) et avec la méthode SMOTE(en haut à droite) et la matrice de confusion sans la méthode SMOTE (en bas à gauche) et avec la méthode SMOTE (en bas à droite).

On remarque que dans le dataset déséquilibré l'accuracy est à 75 % par contre dans la dataset équilibrée l'accuracy et à 51%, cette différence est causée par le principe de la méthode SMOTE vue précédemment.

Concernant l'évaluation des modèles linéaire, on conclut que pour un pourcentage de 75 et 51 n'est pas suffisant pour dire que c'est un bon modèle qui sépare les données donc on ne peut pas prédire si le client est un churn/non churn avec precision.

Conclusion Global sur SVM :

On constate que les résultats obtenus de notre réalisation de SVM sont probablement bon car on a un modele dont on a obtenu plus de 90%.

La meilleure méthode de kernel en se basons sur les résultats c'est la méthode RBF dont on a une accuracy à 90 %.

VI.Conclusion et Perspective

L'objectif principal de cette étude est de déterminer le meilleur modèle pour résoudre ce problème.

Pour déterminer le meilleur modèle, nous devons comparer tous les modèles en ce qui concerne le score.

Modèle	Score de test
Régression logistique	0.76
Arbre de décision	0.78
Random forest	0.71
Boosting	0.72
SVM	0.77

Tableau 10.a Score de précision pour le premier dataset.

Modèle	Score de test
Régression logistique	0.85
Arbre de décision	0.94
Random forest	0.95
Boosting	0.85
SVM	0.90

Tableau 5.b Score de précision pour le second dataset.

En observant les différents résultats, on s'aperçoit que l'arbre de décision et la méthode SVM avec kernel RBF nous donne les meilleurs résultats pour le premier dataset.

En ce qui concernent le second dataset l'arbre de décision et Random Forest nous permet d'avoir les meilleurs prédictions.

Les différentes préconisation que nous ferons pour une entreprise pour la prédiction de churn sont multiples, nous allons en énumérer certaines.

L'entreprise pourrait proposer des prix compétitif (stratégie de tarification).

Par exemple analyser les habitudes du client au cours de plusieurs mois et lui proposer l'abonnement le plus adaptée afin de le fidéliser et de baisser sa facture ou bien lui proposer un rabais sur les différents services les plus utiliser.

Les client n'hésite pas à se désabonner ou de changer d'opérateur si ils ne trouvent pas ce qu'ils recherchent, l'entreprise doit donc diversifier ses offres.

L'attrition client étant directement lié à la satisfaction client, récompenser les client pour leur fidélité à travers un système de points permettant d'obtenir des réductions.

Si le client appel le service clientèle, l'opérateur pourrait lui envoyait un email personnalisée pour le notifie que les problèmes qu'il subi ont bien était pris en compte et qu'une réduction ou bien une récompense sera effectue sur sa prochaine facture.

Enfin des stratégies doivent être mise en place pour réduire le nombre d'appel au service client.

Parmi les différentes perspective possible pour ce projet, nous pouvons citer différentes technique de Machine Learning comme la méthode des **K Plus Proche Voisins (KNN)** pour sa simplicité ou bien l'utilisation d'un réseau de neurones pour avoir les meilleurs résultats (les réseaux de neurones excellent pour la prédiction).