NESSAH Hakim

KHERMOUCHE Chafi

NASER Khalid

Data Visualisation

REPORT: CHURN and MARKETING CAMPAIGN

Data Preprocessing:

Dans un premier temps, il était nécessaire de rendre notre base de données exploitable pour la mise en place de nos modèles par la suite. Nous avons pris la décision de tout d'abord filtrer nos bases de données séparément pour ensuite pouvoir les fusionner et les soumettre à nos modèles de Machine Learning.

Dans le processus de préparation des données, nous avons suivi plusieurs étapes clefs : La transformation de certaines variables de date, le traitement des valeurs manquantes et des valeurs extrêmes. Voici les opérations faites selon la base de données :

Base de données principale (data1)

Conversion de la variable « si2014 » en facteur et des variables « ultimo_ing.x », « abb13 », « abb14 » en objet de date.

Création de la variable « abb14_renewed ».

Elle prend la valeur 1 si nous disposons d'une date de renouvellement en 2014 (« abb14 » n'est pas manquant) et 0 sinon. Cette nouvelle variable est une sorte d'indicateur de renouvellement et elle simplifie l'interprétation en se concentrant sur la question de savoir si un renouvellement a eu lieu en 2014, plutôt que de savoir quand il a eu lieu. Ceci peut être particulièrement utile si la question principale d'intérêt est de savoir si les clients renouvellent ou non, plutôt que d'analyser le moment de leur renouvellement.

Traitements des dates manquantes

Pour ceux qui n'ont jamais visité le musée (ne figure pas dans in13) et donc n'ont pas de date de dernière visite, on remplace la date de dernière visite (variable « ultimo_ing.x ») par la date de début d'abonnement (variable « abb13 »).

En revanche, ceux qui ont déjà visité le musée (figure dans an13) mais dont la date de dernière visite est manquante, on peut leur attribuer une date trouvée dans la base de données des visites (an13).

Base de données des visites (in13)

Nous n'avons effectué aucune modification sur ces données qui nous semblaient complètes (pas de valeurs manquantes) et fiables pour la suite.

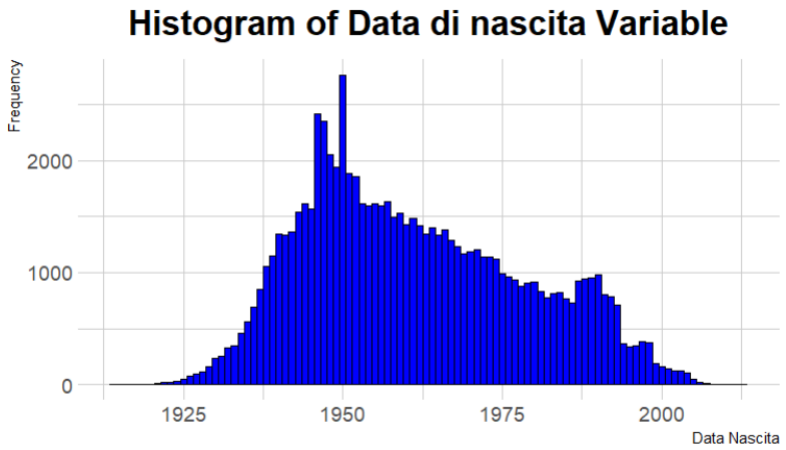
Base de données des informations clients (an13)

Suppression de la variable « professionne » qui ne contenait que des valeurs manquantes.

Autrement dit, nous ne disposons d'aucune information sur le statut professionnel des individus donc nous avons fait comme si cette variable n'existait pas.

Nous avons attribué la valeur "Unknown" aux données manquantes de la variable « sesso » (genre de l'individu).

Nous avons dû traiter des valeurs aberrantes présentes dans certaines variables de cette base de données, comme la variable 'data di nascita', qui contenait des valeurs non sensées telles que 902, 903, 2013, 2027, etc. La distribution de ces valeurs est représentée dans l'histogramme ci-dessous.



Après avoir cela, nous avons pu enfin fusionner nos bases de données en une seule base que l'on a nommé « churn » et encoder l'ensemble de nos variables binaires. Cet encodage explique pourquoi nous retrouvons, dans notre code, des variables telles que « sesso_F », « tipo_pag_ACQUISTO ON-LINE » etc….

Modélisation (Machine Learning) :

Cette partie est consacrée à l'implémentation de nos modèles dans le but de prédire le taux de renouvellement (variable cible = « si2014 ») dans un but purement marketing.

Dans notre recherche des meilleurs résultats possible, nous avons mis en place différents tests statistiques (que nous n'allons pas détailler ici) sur les variables afin d'obtenir les features (X) les plus pertinentes pour nos prédictions.

Puisque nous sommes dans un problème de classification (prédiction d'une variable qualitative, ici si2014), nous avons choisi les modèles suivants : **La régression logistique, les arbres de décisions, le Random Forest et enfin le XG Boost.**

Avant le tuning des paramètres

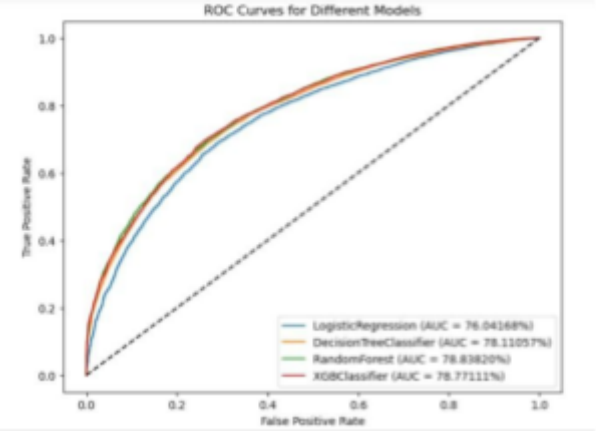
	Model	Training Accuracy %	Testing Accuracy %
0	Logistic Regression	73.625842	73.558772
1	Decision Tree Classifier	93.573746	67.400799
2	Random Forest Classifier	93.573746	69.990017
3	XGBoost Classifier	77.448840	74.962566

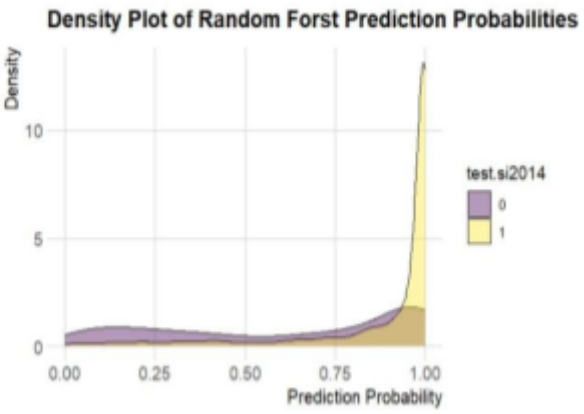
Après le tuning des paramètres

Model	Training Accuracy %	Testing Accuracy %
Tuned Logistic Regression	73.64%	73.53%
Tuned Decision Tree Classifier	75.77%	74.69%
Tuned Random Forest Classifier	77.20%	75.24%
Tuned XGBoost Classifier	75.65%	75.32%

Globalement, nous observons une amélioration significative des performances de l'échantillon de test (le seul qui nous intéresse) à l'image du modèle de Random Forest qui passe de 69% à 75% de performance. En d'autres mots, nos modèles prédisent de manière correcte 3 fois sur 4, c'est-à-dire qu'il va prédire un non-renouvellement pour une personne qui le souhaite effectivement 3 fois sur 4.

Pour ce qui est de la courbe ROC et de la distribution des probabilités, nous avons obtenu les résultats suivants :





La courbe ROC représente la relation entre le taux de vrais positifs (sensibilité du modèle) et le taux de faux positifs (1 - Spécificité) pour différentes valeurs de seuil de classification. Plus précisément, l'axe des Y représente la capacité du modèle à correctement prédire les personnes qui renouvellent leur abonnement (si2014 = 1) et l'axe des X représente la proportion des individus qui ne renouvelle pas leur abonnement, mais qui ont été classés de ceux qui souscrivent à un renouvellement. Un modèle idéal aurait une courbe ROC se rapprochant du coin supérieur gauche avec un angle droit.

L'AUC (l'aire sous la courbe) mesure la performance globale du modèle. Une AUC-ROC proche de 1 suggère une bonne capacité prédictive, tandis qu'une valeur de 0.5 indique une performance aléatoire.

Dans notre cas, le modèle le plus performant est sans surprise le XG Boost avec une aire égale à 0,78. Dans ce cas, une AUC de 0.78 suggère que le modèle peut distinguer raisonnablement bien entre les exemples positifs et négatifs, mais il y a encore de la place pour l'amélioration.

Enfin, on observe une distribution de probabilités assez centrée autour de 1 pour le modèle Random Forest. Cela est à la fois une bonne chose puisque le modèle va correctement classifier les "non churners" avec une forte probabilité, mais aussi une mauvaise chose, car le modèle va difficilement prédire le départ de ceux qui vont effectivement suspendre leur abonnement.

Courbe de profit

Avant de montrer la courbe de profit, nous avons pensé qu'il était nécessaire de tout d'abord expliqué comment nous avons compris la question : La campagne de marketing à la possibilité de contacter l'ensemble des individus présents dans la base de données. Cette prise de contact à pour principal objectif de réduire le taux de non-renouvellement au sein de sa clientèle. Cependant, les appels constituent un cout égal à 20 centimes par individu.

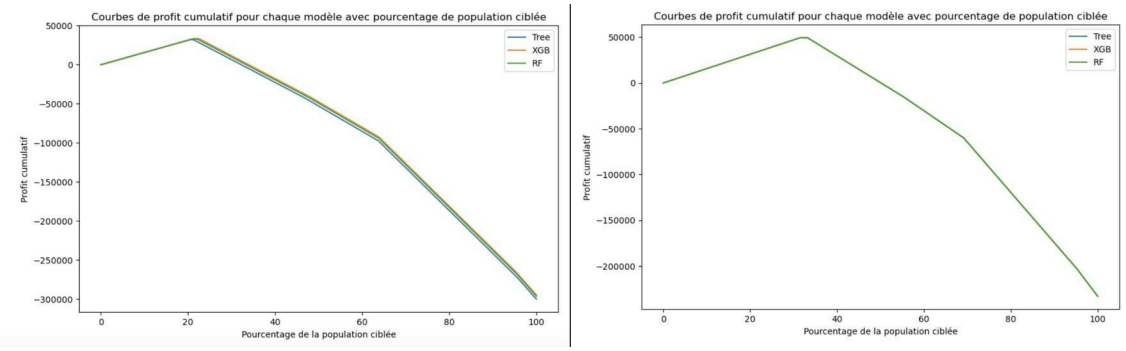
Sachant cela, il existe deux possibilités :

- La campagne à appeler un churner, ce qui rend l'appel pertinent et financièrement avantageux. **Le profit est alors égal à (10 – cost).**
- La campagne à appeler un non-churner, ce qui veut dire que nous appelons un individu qui comptait déjà renouveler son abonnement. Cela constitue uniquement un cout pour la campagne. **Le profit est de (10 – importo – cost).**

Voici la formule de profit globale que nous avons choisi ainsi que la courbe associée :

```
Profit = (10 - cost)*(1 - si2014) + (10 - importo
- cost)*si2014.
Avec si2014 = 0 où 1
```

Clients churners : si2014=0 et Client non-churners : si2014=1



Le graphique de gauche, lui, est construit selon la répartitions prédites (par les modèles de machine Learning) de la variables cible « si2014 », ce qui explique que les courbes diffèrent légèrement entre elles, dû à leurs performances légèrement différentes. En effet, nous observons 24533 churners (si2014=0) sur les 80140 individus, ce qui représente à peu près 30% de la population.

Cela tombe bien, on retrouve un profit maximal à ce niveau de la population qui tend à se rapprocher du profit maximale estimé par la courbe théorique, qui est d'environ 50 000.

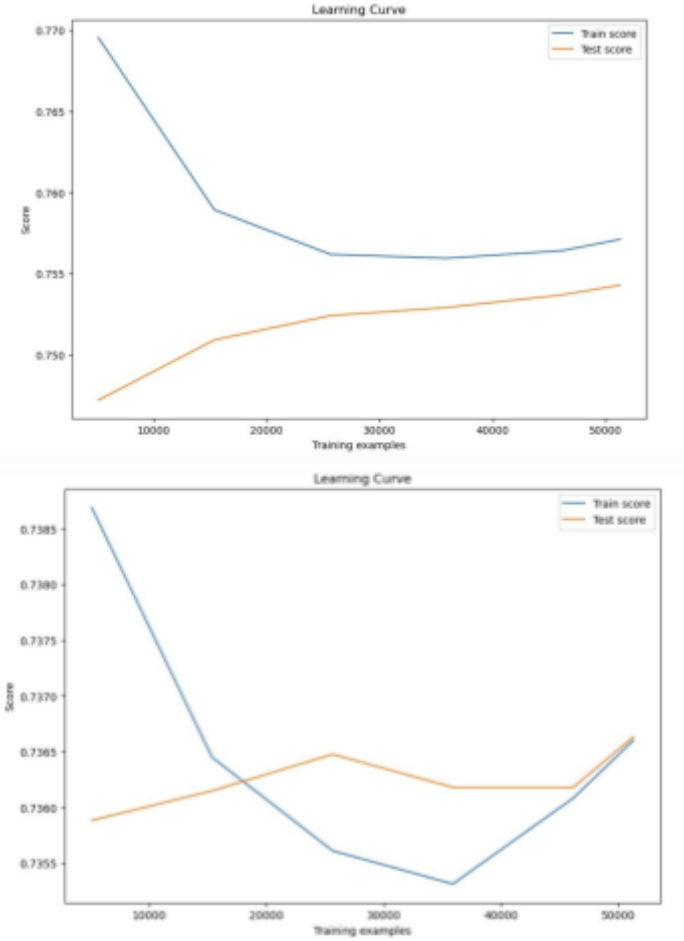
Le graphique de droite (courbe théorique) représente les courbes de profits selon la répartitions réelles (telle qu'elle est dans notre base de données) de la variables cible « si2014 ». On observe que le point maximal (50 000) est atteint en un niveau de population légèrement plus élevé, 35% environ. En d'autres mots, 35% de la population ont été classée comme des churners et nous rapporte des profits positifs.
Les courbes pour chacun des modèles se superposent car le graphique est construit indépendamment des prédictions et dépend seulement d'un ensemble réel de valeurs tirées de la variable expliquée.

Pour finir, on observe que nos modèles ont été un véritable outil d'aide à la décision dans le cadre de la campagne d'appel puisque les résultats s'apparentent assez bien à la réalité car la courbe de profit obtenue avec nos modèles de prédictions correspond à la courbe théorique.

Nous pourrions éventuellement améliorer ces modèles et les rendre plus performant en développant la phase pré-traitement, simplement en récoltant plus de données auprès de nos clients ce qui nous permettra d'ajouter plus de variables, lors de la phase de data engineering.

Annexe :

Courbe d'apprentissage pour les modèles : XGBoost Classifier et Logistic Regression



Graphique représentant l'importance des fonctionnalités dans notre modèle principal le XGBoost Classifier :