



Département d'Informatique

Licence Professionnelle: Intelligence Artificielle et Science de
Données

Mémoire de Projet Apprentissage Automatique I

“ Prévision de L'attrition de la clientèle
dans le secteur de télécommunication ”

Membres du Groupe:

- TAIBI Sara
- BAAZIZI Hafida
- OUFKIRI Chaimaa
- BAKRIM Sana

Encadrée par:

- Mme. FATNA EL MENDILI

*Année Académique:
2023/2024*

Remerciements

Au nom d'Allah le tout miséricordieux, le très miséricordieux.

Nous sommes sincèrement reconnaissants envers le Tout-Puissant Allah pour ses grâces, bénédictions, force, et guidance, et surtout pour sa bienveillance depuis le début de notre vie académique jusqu'à présent.

Nos remerciements vont en premier lieu à Pr. EL MENDILI, notre enseignante du module "Apprentissage Automatique I", dont les cours éclairés ont été la pierre angulaire de ce projet. Sa passion pour son travail et ses explications détaillées ont été une source constante d'inspiration. Nous tenons à exprimer notre gratitude pour ses recommandations constructives et les conseils pertinents prodigués lors des étapes préliminaires de ce projet.

Nous saluons également l'ensemble de l'administration de l'École Supérieure de Technologie de Meknès pour avoir créé un environnement propice à l'apprentissage.

Nous exprimons hautement notre plus profonde gratitude à nos collègues et à nos nombreux amis qui ont enduré ce long processus avec nous, en offrant toujours une merveilleuse collaboration et des encouragements.

Enfin, nos remerciements vont à vous, cher lecteur, d'avoir accepté de juger ce modeste travail.

Dédicace

Nous souhaitons dédier ce travail modeste à :

Tout d'abord, à nos chers parents qui ont toujours été une source de motivation et de soutien dans nos études. Ce travail est le fruit de leur soutien constant et de leurs encouragements tout au long de notre parcours scolaire. Nous leur exprimons notre profonde gratitude à travers cette humble dédicace.

Nous prenons également plaisir à dédier ce projet à nos propres efforts, à notre dévouement et aux sacrifices que nous avons consentis pour mener à bien ce travail.

BAKRIM Sana

OUFKIRI Chaimaa

BAAZIZI Hafida

TAIBI Sara

Table des matières

Remerciements	1
Dédicace	2
Table des matières	3
Liste des figures	4
Liste de Tableaux	5
Résumé	6
Abstract	6
Introduction générale	7
Chapitre1: Description Générale de projet	9
1. Context	9
1.1. Secteur de Télécommunication :	9
1.2. Gestion de Relation Client	9
2. Problématique	10
2.1. Attrition de la clientèle:	10
2.2. Taux d'attrition	11
3. Solution Proposée	11
3.1. Prévision:	11
3.2. Machine Learning	12
a) Définition	12
b) Type d'apprentissage automatique	12
c) Les mesures de performance	13
Chapitre2 : Développement et Implémentation	15
1. Outils de développement	15
1.1. Langage de programmation	15
1.2. Environment de development	16
a) Jupyter	16
b) Streamlit	16
1.3. Librairies	17
2. Structure de projet	19
2.1. Structure de code	19
2.2. Diagramme d'architecture de projet	20
3. Mise en Oeuvre	21
3.1. Description de jeux de données	21
3.1.1. Compréhension de jeux de données	21
3.1.2. Visualisation de données	25
3.2. Analyse Exploratoire	28
3.3. Modélisation	43
Chapitre3 : Résultats et Interprétations	58
1. Comparaison de résultats	58
1.1. Comparaison des Modèles avec Tabulate :	58
1.2. Lazy Predict	59
1.3. Validation croisée :	60
2. Test de prédiction	61
Chapitre 4 : Réalisation de l'application	63
1. Accueil :	63
2. Dataset :	64
3. Visualisation des données :	64
4. Manipulation de Données :	66
5. Analyse de Répartition :	70
6. Prédiction :	71
Conclusion	75

Liste des figures

Figure 1 : Gestion de relation client	10
Figure 2 : Première section de structure de code	19
Figure 3 : Deuxième section de structure de code	20
Figure 4 : Troisième section de structure de code	20
Figure 5 : Diagramme d'architecture	21
Figure 6 : Affichage d'un aperçu de dataset	22
Figure 7 : Distinction entre données numériques et catégorielles	25
Figure 8 : Distribution de variable City	25
Figure 9 : Distribution de variable Zip Code	26
Figure 10 : Distribution de variable Latitude	26
Figure 11 : Visualisation des valeurs aberrants	27
Figure 12 : Matrice de visualisation des valeurs manquantes	28
Figure 13 : Copie du dataset	29
Figure 14 : Code de suppression des colonnes	29
Figure 15 : Colonnes restantes dans dataset	29
Figure 16 : Distribution de Total charges avant transformation	30
Figure 17 : Code de Skew avant transformation	30
Figure 18 : Code de Transformation de racine carrée	31
Figure 19 : Code de Skew après transformation	31
Figure 20 : Distribution de Total charges après transformation	31
Figure 21 : Affichage des valeurs uniques de "Contrat"	31
Figure 22 : Affichage de la colonne "Tenure in Months"	32
Figure 23 : Affichage de la colonne "Monthly Charge"	32
Figure 24 : code de remplacement des Nans en formule	32
Figure 25 : Variable Gender à deux modalités	33
Figure 26 : Code d'Encodage de Gender	33
Figure 27 : Autres variables à deux modalités que Gender	34
Figure 28 : Résultat de traitement des Nulles	34
Figure 29 : Matrice de corrélation	35
Figure 30 : Distribution de la Variable Gender	36
Figure 31 : Distribution de la Variable Age	36
Figure 32 : Box-Plot du Number of Dependents avant traitement des aberrantes	40
Figure 33 : Box-Plot du Number of Dependents après traitement des aberrantes	41
Figure 34 : Count-Plot de Y identifiant le déséquilibre existant	42
Figure 35 : Vérification d'équilibrage par shape	42
Figure 36 : Count-Plot de Y identifiant les classes après rééquilibrage	43
Figure 37 : Code de division de données	43
Figure 38 : Initiation de RandomForest	46
Figure 39 : Précision de RandomForest	47
Figure 40 : F1Score de RandomForest	47
Figure 41 : Recall de RandomForest	47
Figure 42 : Rapport de classification de RandomForest	48
Figure 43 : Matrice de confusion de RandomForest	48
Figure 44 : Courbe ROC de RandomForest	49
Figure 45 : Initiation de LogisticRegression	49
Figure 46 : Precision de LogisticRegression	50
Figure 47 : F1Score de LogisticRegression	50
Figure 48 : Recall de LogisticRegression	50
Figure 49 : Rapport de classification de LogisticRegression	50
Figure 50 : Matrice de confusion de LogisticRegression	51
Figure 51 : Courbe ROC de LogisticRegression	51

Figure 52 : Initiation du XGB	52
Figure 53 : Précision de XGB	52
Figure 54 : F1Score de XGB	52
Figure 55 : Recall de XGB	53
Figure 56 : Rapport de classification de XGB	53
Figure 57 : Matrice de confusion de XGB	54
Figure 58 : Courbe ROC de XGB	54
Figure 59 : Initiation du Gradient	55
Figure 60 : Précision de Gradient	55
Figure 61 : F1Score de Gradient	55
Figure 62 : Recall de Gradient	55
Figure 63 : Rapport de classification de Gradient	56
Figure 64 : Matrice de confusion de Gradient	56
Figure 65 : Courbe ROC de Gradient	57
Figure 66 : Table de comparaison	59
Figure 67 : Table de LazyPredict	60
Figure 68 : Résultat de test	62
Figure 69 : Interface variables catégorielles	65
Figure 70 : Interface valeurs manquantes	66
Figure 71 : Interface valeurs aberrantes	66
Figure 72 : Interface Suppression des Colonnes Inutiles	67
Figure 73 : Interface Total charges avant transformation	67
Figure 74 : Interface Total charges après transformation	68
Figure 75 : Interface Encodage des Variables Catégorielles	68
Figure 76 : Interface Correlation	69
Figure 77 : Interface Détection de Valeurs Aberrantes Avant Traitement	69
Figure 78 : Interface Détection de Valeurs Aberrantes Après Traitement	70
Figure 79 : Interface Déséquilibre des Données	70
Figure 80 : Interface Équilibrage des Données	71
Figure 81 : Interface de formulaire de prediction	72
Figure 82 : Interface de résultat (client joined)	73
Figure 83 : Interface de résultat (client stayed)	73
Figure 84 : Interface de résultat (client churned)	74

Liste de Tableaux

Table 1 : Description des colonnes de dataset	24
Table 2 : Table d'analyse statistiques	38

Résumé

Ce présent rapport est rédigé dans le cadre du mini-projet de clôture du module d'apprentissage automatique I en semestre 5 de la licence professionnelle en intelligence artificielle et sciences de données au sein de l'École Supérieure de Technologie de Meknès. Notre projet porte sur la prévision de l'attrition de la clientèle en utilisant un modèle de prédiction du désabonnement des clients qui revêt une importance particulière dans l'identification des clients susceptibles de se désabonner, permettant ainsi la mise en place de mesures appropriées pour les retenir.

Abstract

This report is written within the context of the machine learning I module closing project in semester 5 of the professional license degree in artificial intelligence and data science at the Higher School of Technology of Meknes. Our project focuses on predicting customer churn using a prediction model which is of particular importance in identifying customers likely to churn, thereby enabling the implementation of appropriate measures for retaining them.

Introduction générale

Aujourd'hui, l'Intelligence Artificielle (IA) et la Science des Données émergent en tant que forces motrices de l'innovation technologique, façonnant de manière significative notre compréhension et notre utilisation de l'informatique. Ces domaines jouent un rôle central dans la résolution de problèmes complexes, l'automatisation de processus, et la prise de décisions basées sur des données.

Le Machine Learning, branche de l'intelligence artificielle qui permet aux systèmes de s'améliorer automatiquement par l'expérience, offre des solutions innovantes à des problèmes complexes. Sa présence est devenue indispensable dans divers secteurs, propulsant des avancées significatives dans la prise de décisions, la prévision et l'automatisation de tâches.

Dans ce contexte évolutif, nous avons entrepris un projet ambitieux, sous la direction de notre professeur de Machine Learning. L'objectif est de mettre en œuvre des techniques avancées de ML pour créer des modèles de prédiction du churn des clients dans le secteur de télécommunication, afin d'identifier les clients les plus proches d'abandonner leur opérateur téléphonique actuel.

Ce rapport détaille toutes les phases, les étapes que nous avons suivies depuis le lancement du projet jusqu'à la livraison de notre travail.

Selon la synthèse présentée ci-dessus, ce mémoire s'articule autour de trois chapitres principaux :

- **Le premier chapitre** : sous le titre « **Description générale du projet** » ,

Aborde l'étude du contexte général du projet, la problématique et la solution proposée.

- **Le deuxième chapitre** : sous le titre « **Développement et Implémentation** » ,

Définit l'environnement de développement, les outils spécifiques utilisés, décrit la structure et la mise en œuvre de notre projet, ainsi que les différentes étapes qui le composent depuis la définition de la base de données utilisée et sa préparation, jusqu'à la modélisation.

- **Le troisième chapitre** : sous le titre « **Résultats et Interprétations** » ,

Où nous exposerons la comparaison des résultats obtenus avant de passer à la prédition.

- **Le quatrième chapitre** : "Réalisation de l'Application", présente notre application concrète développée avec Streamlit dans le cadre du projet axé sur la prédition de l'attrition des clients dans le secteur des télécommunications.

Chapitre1:

Description Générale de projet

1. Context

Notre projet peut s'appliquer à plusieurs secteurs et domaines qui fournissent des services aux clients. Nous intéressons à l'application au favor de secteur de Télécommunication pour un jeu de données d'une entreprise de télécommunications en Californie au deuxième trimestre de 2022.

1.1. Secteur de Télécommunication :

L'évolution des télécommunications a débuté au cours de la première moitié du XIXe siècle avec l'introduction du télégraphe électrique permettant l'envoi de messages à l'aide de lettres et de chiffres, arrivant au point où les télécommunications mobiles ont ensuite émergées offrant la possibilité aux individus de communiquer vocalement et rester connectés sans déplacement.

A l'heure actuelle, les télécommunications font partie intégrante d'un secteur industriel dynamique et crucial qui fournit des services de communication à travers des réseaux variés tels que les téléphones mobiles, les lignes terrestres, l'internet haut débit et d'autres technologies qui génèrent des millions multiples de chiffres chaque année dans le monde entier.

Aujourd'hui, on parle de la télécommunication étant un domaine qui ne cesse jamais d'avancer et d'évoluer avec l'émergence de la 5G, l'internet des objets (IoT) et d'autres innovations qui continuent de redéfinir la manière dont nous nous connectons et communiquons.

1.2. Gestion de Relation Client

Dans le secteur des télécommunications, la gestion de relation client(GRC) représente un défi; maintenir cette relation revêt une importance cruciale à cause de la forte concurrence au niveau de cette industrie et compte tenu que les clients ont un large éventail d'options parmi les différents fournisseurs de services.

La gestion de relation client est un processus lourde impliquant la mise en oeuvre de stratégies visant à établir, maintenir et améliorer les relations avec les clients, tout en répondant à leurs besoins et en optimisant leur expérience.

En résumé, l'attribution des mesures dans ce côté (voir la figure de schéma illustrant) par la variété des services offerts et la diversité des offres tarifaires et d'autres avantages, constitue un élément clé pour fidéliser la clientèle et rester compétitif.

Comme la figure montre Une bonne stratégie de gestion relation client implique de concentrer prioritairement sa stratégie sur le client, de passer d'une vision centrée sur le produit à une vision centrée sur le client.



Figure 1: Gestion de relation client

2. Problématique

Nous introduisons dans cette partie la problématique de l'attrition de la clientèle dans le secteur de télécommunication et son évaluation.

2.1. Attrition de la clientèle:

L'attrition de la clientèle, également connue sous le nom de « churn » se réfère à la perte de clients dans une entreprise. C'est un enjeu majeur dans le domaine de télécommunications, l'action par laquelle un client part est la résiliation en mettant fin à son engagement vis-à-vis d'un service. Cette procédure de désabonnement peut varier en termes de durée et de complexité

selon les circonstances. Pour les entreprises de télécommunications, ce phénomène est évalué à travers le taux d'attrition ou de résiliation que nous découvrons dans ce qui suit.

2.2. Taux d'attrition

Le taux de désabonnement ou d'attrition des clients constitue l'une des métriques essentielles à évaluer pour une entreprise en expansion. Bien que cette mesure ne soit pas toujours la plus réjouissante, elle offre à l'entreprise une évaluation franche de sa fidélisation client.

Sa formule de calcul :

$$\text{Taux d'attrition} = \frac{\text{Nombre Clients Perdus}}{\text{Nombre Clients Total}}$$

Dans notre contexte, nous prenons en considération par supposition les deux types de clients : les clients involontaires, que la société de télécommunication décide de supprimer pour des raisons telles que la fraude et le non-paiement, ainsi que les résiliations volontaires, qui surviennent de manière incontrôlée et intentionnelle de la part des clients.

Par conclusion, un taux de désabonnement élevé ou en constante augmentation peut avoir des conséquences négatives sur la rentabilité et la croissance de l'entreprise.

3. Solution Proposée

Le projet à la main propose la méthode de prévision à laquelle nombreuses entreprises se tournent pour anticiper la résiliation des clients.

3.1. Prévision:

La prévision est l'analyse anticipative d'une situation permettant, par déduction, calcul ou mesure scientifique, de connaître à l'avance son évolution. Elle constitue la science de la description de l'avenir.

Selon Fayol, considéré comme le père de la direction moderne, "la prévision est l'essence même de la gestion". Le succès d'une entreprise dépend largement de l'efficacité des prévisions et de la préparation aux événements futurs en contribuant à la prise de décision.

Prédire l'attrition des clients avant sans occurrence est donc essentiel pour le succès global d'une entreprise de télécommunication, il s'agit de tourner vers des analyses prédictives pour créer des modèles anticipant ce phénomène.

Le projet vise en premier lieu à comprendre les facteurs qui contribuent à l'attrition de la clientèle et utiliser ses facteurs pour la prévision à l'aide de l'apprentissage automatique.

3.2. Machine Learning

En résumé, notre projet offre une opportunité précieuse d'appliquer des méthodes de machine learning pour résoudre un défis spécifique au secteur des télécommunications en contribuant à la rétention et à la satisfaction de la clientèle.

Dans cette partie nous présentons les différents définitions du l'apprentissage automatique et nous citons quelques notions de base.

a) Définition

Le **Machine Learning** (apprentissage automatique) est une discipline de l'intelligence artificielle qui permet aux systèmes informatiques d'apprendre et de s'améliorer à partir de l'expérience, sans être explicitement programmés. Il repose sur l'utilisation d'algorithmes et de modèles statistiques pour permettre aux ordinateurs de réaliser des tâches spécifiques sans instruction directe.

b) Type d'apprentissage automatique

Nous distinguons principalement deux grands types d'apprentissage :

- **Apprentissage supervisé :**

L'approche d'apprentissage supervisé, au cœur de notre démarche, vise à élaborer des modèles capables de comprendre la relation entre les données d'apprentissage et les résultats escomptés. Ce processus repose sur une base d'apprentissage où chaque exemple est préalablement étiqueté avec sa classe respective, permettant au modèle d'assimiler les caractéristiques distinctives de chaque catégorie.

Cette approche implique la manipulation d'un ensemble de données structuré, où chaque observation est associée à un ensemble de variables représentées par un vecteur X. En plus de ces variables, chaque observation possède une valeur de sortie Y, dénommée "valeur supervisée", jouant un rôle crucial dans la classification. En optant pour l'apprentissage supervisé dans notre projet, nous capitalisons sur sa capacité à former des modèles performants en exploitant des données préalablement annotées, renforçant ainsi la précision de nos prédictions.

- **Apprentissage non-supervisé :**

Ce type d'Apprentissage est donc l'objectif de concevoir un modèle structurant l'information sans préconnaissance des comportements, catégories, ou classes des données d'apprentissage. Dans cette approche, l'objectif est de découvrir ces éléments. Par exemple, un chef de magasin de location cherchant à comprendre les préférences de ses clients pourrait regrouper ces derniers en fonction de leurs habitudes d'achat, sans préalablement connaître ces habitudes.

- En conclusion, cette diversité d'approches offre des solutions adaptées à différents contextes et besoins. Dans le cadre de notre projet, nous avons principalement adopté une approche de classification qui fait partie de l'apprentissage supervisé, mettant en œuvre des modèles pour catégoriser des données dans des classes prédéterminées.

c) Les mesures de performance

Pour notre projet spécification, nous avons recourues aux mesures de performance qui fournissent une évaluation détaillée de la capacité de nos modèles à générer des prédictions précises, en mettant en lumière des aspects tels que la précision, le rappel, le F1 Score, la matrice de confusion, le rapport de classification, et les courbes ROC.

- **Précision :**

La précision mesure le nombre de vrais positifs parmi toutes les prédictions positives du modèle. C'est une métrique importante pour évaluer la précision des prédictions positives.

- **Rappel :**

Le rappel, également appelé sensibilité, mesure le nombre de vrais positifs parmi toutes les instances réellement positives. Il donne une indication de la capacité du modèle à identifier correctement les cas positifs.

- **F1 Score :**

Le F1 Score est une métrique qui combine à la fois la précision et le rappel. Il est particulièrement utile lorsque les classes sont déséquilibrées, fournissant une mesure globale de la performance du modèle.

- **Matrice de Confusion :**

La matrice de confusion présente le nombre de vrais positifs, vrais négatifs, faux positifs et faux négatifs. Elle offre une vue détaillée des performances du modèle pour chaque classe.

- **Rapport de Classification :**

Le rapport de classification résume les performances du modèle en fournissant la précision, le rappel, le F1 Score et le support pour chaque classe. Il offre une compréhension détaillée des performances par classe.

- **Courbes ROC :**

Les courbes ROC (Receiver Operating Characteristic) mesurent la capacité du modèle à discriminer entre les classes en variant le seuil de décision. L'aire sous la courbe ROC (ROC AUC) quantifie la performance globale du modèle.

Chapitre2 :

Développement et Implémentation

1. Outils de développement

1.1.Langage de programmation

Python est un langage de programmation polyvalent, interprété et orienté objet, reconnu pour sa simplicité syntaxique et sa lisibilité accrue. Conçu pour favoriser la productivité des développeurs, Python offre une approche intuitive qui facilite l'écriture et la compréhension du code. Son origine remonte à la fin des années 1980, et depuis, il a gagné une popularité considérable grâce à sa communauté active et à son adoption généralisée dans divers domaines, notamment le développement web, l'automatisation, la science des données, et le machine learning.

Caractéristiques Clés de Python :



- **Simplicité**: Syntaxe épurée pour une programmation simplifiée, laissant place à la logique.
- **Lisibilité** : Priorité à la clarté du code, favorisant la collaboration et la maintenance.
- **Polyvalence** : Adapté à une variété de tâches, avec des bibliothèques spécialisées en web, données, et machine learning.
- **Communauté Active** : Vaste et collaborative, fournissant ressources et bibliothèques open source.
- **Écosystème Riche** : Dynamique avec des outils comme NumPy, scikit-learn, et TensorFlow pour étendre ses capacités.
- **Interprétré** : Flexibilité accrue, exécution sans compilation séparée.

En résumé, Python combine la simplicité, la lisibilité, la polyvalence, une communauté active, un écosystème riche, et une exécution interprétée, ce qui en fait un choix privilégié pour le développement de projets variés, y compris ceux impliquant des applications de machine learning.

1.2. Environment de development

a) Jupyter

Jupyter est une application web open source qui permet de créer et de partager des documents live incluant des codes, des équations, des visualisations et du texte narratif. Il prend en charge plusieurs langages de programmation, mais est particulièrement populaire dans le domaine du data science et du machine learning.



Ces Fonctionnalités :

- ***Environnement Interactif*** : Jupyter offre un environnement interactif où les utilisateurs peuvent exécuter du code en temps réel, facilitant l'exploration et l'analyse des données.
- ***Notebooks Multiples Langages*** : Il prend en charge divers langages de programmation, dont Python, R et Julia, permettant une flexibilité pour différents scénarios.
- ***Visualisation Dynamique*** : Les résultats, tels que les graphiques et les diagrammes, peuvent être visualisés de manière dynamique dans le même document.
- ***Partage Facile*** : Les notebooks Jupyter peuvent être partagés facilement, favorisant la collaboration et la communication dans le domaine du développement et de la recherche.
- ***Intégration avec Widgets*** : La possibilité d'intégrer des widgets interactifs dans les notebooks améliore l'expérience utilisateur et facilite la manipulation des données.
- ***Support pour l'Éducation*** : Souvent utilisé dans l'enseignement et la formation, Jupyter permet d'intégrer du texte explicatif, des équations mathématiques et du code dans un seul document, offrant une expérience d'apprentissage interactive.
- ***Extensions et Add-ons*** : Jupyter dispose d'une communauté active qui crée des extensions et des add-ons, élargissant ainsi ses fonctionnalités de base.

b) Streamlit

Streamlit est un framework open source qui permet aux développeurs de créer rapidement des applications web interactives à partir de scripts Python. Son principal objectif est de simplifier le processus de développement et de permettre aux utilisateurs de convertir facilement leurs analyses de données et leurs modèles machine learning en applications web.

Ces Caractéristiques et Fonctionnalités :



- **Simplicité d'Utilisation** : Streamlit propose une syntaxe simple et intuitive, accessible même aux développeurs novices en développement web.
- **Facile à déployer** : IL permet la conversion rapide de scripts Python en applications web avec quelques lignes de code.
- **Widgets Intégrés** : Offre des widgets intégrés (curseurs, boutons) pour améliorer l'interaction et l'expérience utilisateur.
- **Support pour le Machine Learning** : Spécialement conçu pour la présentation de modèles ML, Streamlit intègre des fonctionnalités pour afficher des résultats, des graphiques et des prédictions.
- **Mise à Jour en Temps Réel** : Les modifications dans le script Python se reflètent automatiquement dans l'application, facilitant un processus de développement itératif.
- **Personnalisation Graphique** : Bien que limité, Streamlit offre des options de personnalisation graphique pour ajuster l'apparence de l'application.
- **Documentation Automatique** : Les commentaires du code Python sont convertis automatiquement en documentation interactive, facilitant la compréhension.
- **Community Active** : Bénéficiant d'une communauté active, Streamlit évolue rapidement avec des partages de ressources et d'extensions.

1.3.Librairies

Dans le cadre de notre projet de machine learning, nous avons exploité un ensemble de bibliothèques Python puissantes pour diverses tâches, allant de la manipulation de données à la création de modèles et à l'évaluation des performances. Voici un aperçu des principales bibliothèques utilisées et certaines fonctions comprises :

- **Pandas**:

Pandas est une bibliothèque Python qui offre des structures de données flexibles, principalement les DataFrames, pour la manipulation et l'analyse de données tabulaires.

- **NumPy**:

NumPy est une bibliothèque fondamentale pour le calcul numérique en Python, introduisant des tableaux multidimensionnels (ndarrays) et des fonctions mathématiques.

- **Matplotlib :**

Matplotlib est une bibliothèque de visualisation de données en Python, permettant la création de graphiques statiques, de diagrammes et de visualisations interactives.

- **Ipywidgets :**

ipywidgets offre des widgets interactifs pour les notebooks Jupyter, facilitant la création d'interfaces utilisateur interactives.

- **Seaborn :**

Seaborn est une bibliothèque de visualisation de données basée sur Matplotlib, simplifiant la création de graphiques statistiques attrayants.

- **Missingno :**

Missingno est une bibliothèque pour la visualisation des données manquantes, permettant d'identifier rapidement les valeurs manquantes dans un ensemble de données.

- **SciPy.stats :**

SciPy.stats fait partie de la bibliothèque SciPy et offre des fonctions statistiques avancées, y compris des tests statistiques et des distributions de probabilité.

- **Imbalanced-Learn (imblearn) :**

Imbalanced-Learn est une bibliothèque dédiée à la gestion des problèmes de classes déséquilibrées en machine learning, offrant des méthodes pour équilibrer les jeux de données.

- **Scikit-Learn :**

Scikit-Learn est une bibliothèque de machine learning en Python, offrant des outils pour la préparation des données, la création de modèles, et l'évaluation des performances.

- **LazyPredict :**

LazyPredict est une bibliothèque facilitant la sélection rapide de modèles de machine learning sans nécessiter une configuration détaillée.

- **Tabulate :**

Tabulate est une bibliothèque pour formater des tableaux à partir de données, facilitant la présentation structurée des résultats.

- **Cross val score (Scikit-Learn) :**

La fonction `cross_val_score` de Scikit-Learn permet d'évaluer les performances d'un modèle via une validation croisée.

- **StandardScaler (Scikit-Learn) :**

StandardScaler est une méthode de la bibliothèque Scikit-Learn utilisée pour normaliser les données, en les mettant à l'échelle pour obtenir une moyenne nulle et une variance unitaire.

- **Zscore :**

La fonction zscore de la bibliothèque scipy.stats permet de calculer le z-score (score z) pour normaliser une distribution de données.

- **BorderlineSMOTE (imblearn) :**

BorderlineSMOTE est une technique d'oversampling de la bibliothèque Imbalanced-Learn utilisée pour équilibrer les classes dans un ensemble de données en synthétisant des exemples supplémentaires pour la classe minoritaire.

- **OneVsRestClassifier (Scikit-Learn) :**

OneVsRestClassifier de Scikit-Learn permet d'effectuer une classification multiclasse en utilisant plusieurs classificateurs binaires.

- **Label binarize (Scikit-Learn) :**

label_binarize de Scikit-Learn permet de binariser des étiquettes multiclasse pour les besoins de certaines évaluations de modèles.

2. Structure de projet

2.1. Structure de code

Au niveau de notre environnement de développement Jupyter, la structure de code dans le fichier .ipynb suit un processus typique pour la compréhension des données, l'analyse exploratoire des données (AED) et le choix d'algorithme d'apprentissage automatique. Voici une description section par section avec les étapes incluses:

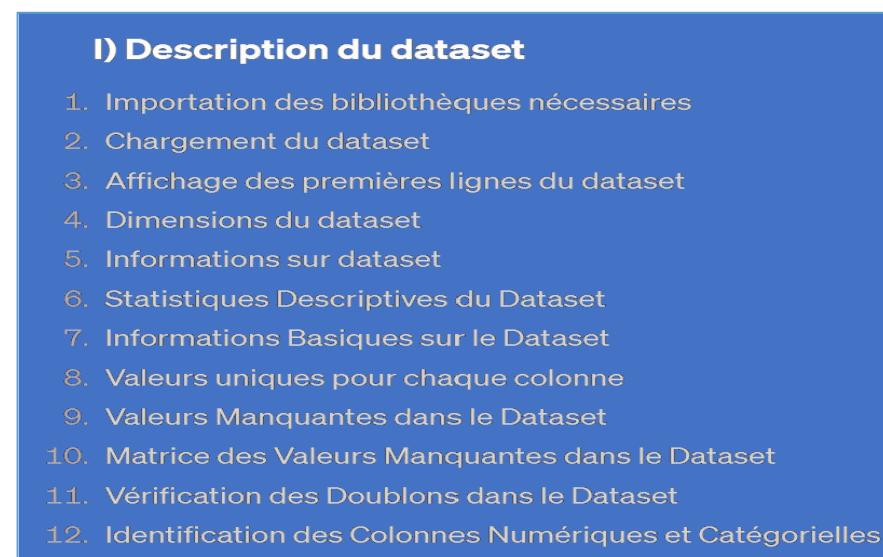


Figure 2: Première section de structure de code

II) Analyse exploratoire des données (AED)

1. Visualisation avec Bar Plot
2. Identification des Valeurs Aberrantes
3. Copie du Dataset
4. Suppression de Colonnes dans la Copie du Dataset
5. Vérification de Suppression de Colonnes dans la Copie du Dataset
6. Traitement de "Total Charges"
7. Encodage des Variables Binaires
8. Encodage des Variables Catégorielles
9. Remplissage des Valeurs Manquantes
10. Matrice de Corrélation avec la Variable Cible
11. Distribution des Variables
12. Analyse de la Répartition des Variables
13. Traitement des Valeurs Aberrantes
14. Traitement du Déséquilibre des Données

Figure 3: Deuxième section de structure de code

III) Choix de l'algorithme d'apprentissage automatique

1. Sélection des Algorithmes
2. Division du dataset en Ensembles d'Entraînement, de Validation et de Test
3. Entraînement et évaluation
4. Comparaison des performances
5. Évaluation sur l'ensemble de test

Figure 4: Troisième section de structure de code

2.2. Diagramme d'architecture de projet

Le diagramme d'architecture ci-dessous est basé sur les étapes de processus de la science des données, ces étapes s'inscrivent dans un flux cohérent, commençant par l'obtention des données, passant par le nettoyage et l'exploration, pour finalement aboutir à la modélisation et l'évaluation des modèles, en terminant par la présentation des résultats et leurs interprétations.

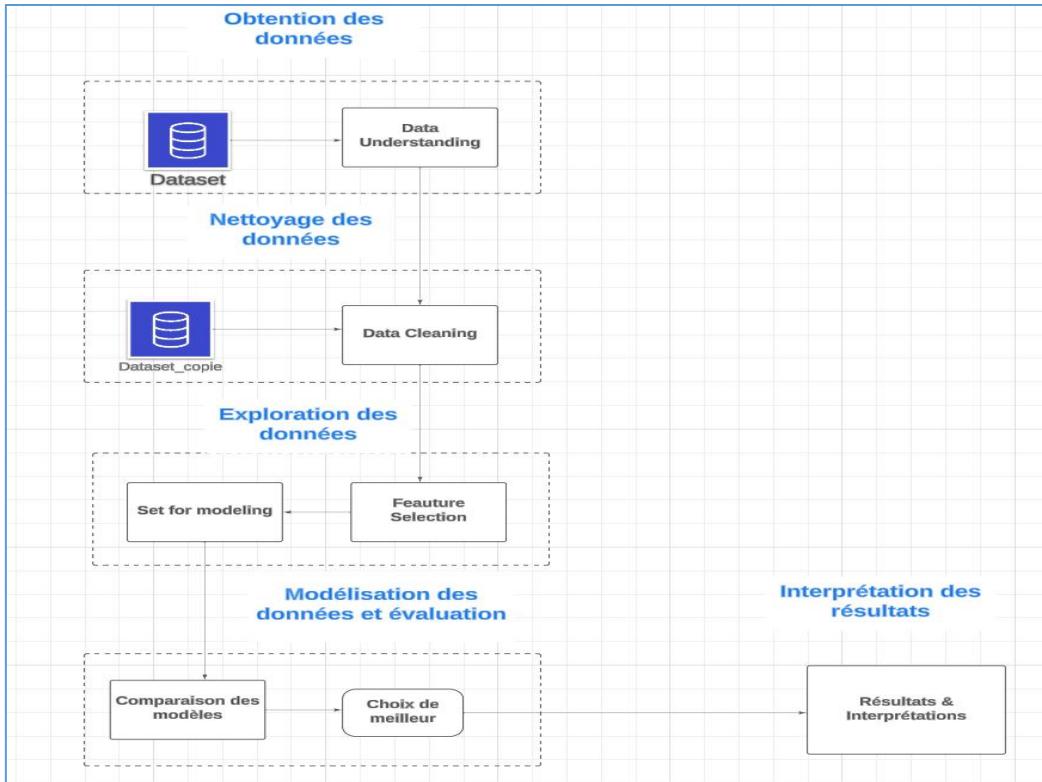


Figure 5: Diagramme d'architecture

Avec «Feature Selection», nous procédons à la sélection des caractéristiques pertinentes pour la modélisation. Le «Set for modeling» comprend les étapes d'équilibrage de données, de division des données en vue de planifier la prochaine étape, à savoir la création et l'évaluation des modèles. En bref, cela résulte donc la question suivante: quels sont nos objectifs avec ces données, et quelles actions nous devons faire avant d'entamer la modélisation.

3. Mise en Oeuvre

3.1. Description de jeux de données

3.1.1. Compréhension de jeux de données

L'ensemble de données fourni semble être une compilation d'informations relatives à la clientèle dans le secteur des télécommunications ou des services par abonnement. Ces données ont été structurées sous la forme d'un tableau. Chaque enregistrement représente un client et contient des détails sur leur démographie, leur emplacement géographique, leur ancienneté, leurs services d'abonnement, leur statut pour le trimestre (nouveau client, client existant, ou client ayant résilié), et bien plus encore.

Pour notre étude nous avons choisi une base de données sur le site « [kaggle.com](https://www.kaggle.com) » Notre ensemble de données contient 7043 lignes et 38 variables.

	Customer ID	Gender	Age	Married	Number of Dependents	City	Zip Code	Latitude	Longitude	Number of Referrals	...	Payment Method	Monthly Charge	Total Charges	Total Refunds	total Extra Data Charges	Dist Ct
0	0002-ORFBO	Female	37	Yes	0	Frazier Park	93225	34.827662	-118.999073	2	...	Credit Card	65.6	593.30	0.00	0	:
1	0003-MKNFE	Male	46	No	0	Glendale	91206	34.162515	-118.203869	0	...	Credit Card	-4.0	542.40	38.33	10	
2	0004-TLHLJ	Male	50	No	0	Costa Mesa	92627	33.645672	-117.922613	0	...	Bank Withdrawal	73.9	280.85	0.00	0	
3	0011-IGKFF	Male	78	Yes	0	Martinez	94553	38.014457	-122.115432	1	...	Bank Withdrawal	98.0	1237.85	0.00	0	:
4	0013-EXCHZ	Female	75	Yes	0	Camarillo	93010	34.227846	-119.079903	3	...	Credit Card	83.9	267.40	0.00	0	

5 rows × 38 columns

Figure 6: Affichage d'un aperçu de dataset

Le tableau ci-dessous explique le rôle de chaque colonne :

Caractéristiques	Description	Type
Customer ID	Un identifiant unique pour chaque client	Object
Gender	Genre (Male, Femelle)	object
Age	L'âge actuel du client.	Int64
Married	Indique si le client est marié : Oui, Non.	object
Number of Dependents	Indique le nombre de personnes à charge vivant avec le client (personnes à charge pouvant être des enfants, des parents, des grands-parents, etc.).	Int64
City	La ville de la résidence principale du client.	object
Zip Code	Le code postal de la résidence principale du client.	int64
Latitude	La latitude de la résidence principale du client.	float64
Longitude	La longitude de la résidence principale du client.	float64
Number of Referrals	Indique le nombre de fois où le client a recommandé un ami ou un membre de sa famille à cette entreprise à ce jour.	Int64
Tenure in Months	Indique le nombre total de mois pendant lesquels le client a été abonné à l'entreprise à la fin du trimestre spécifié ci-dessus.	Int64
Offer	Identifie la dernière offre marketing acceptée par le client : Aucune, Offre A, Offre B, Offre C, Offre D, Offre E.	object

Phone Service	Indique si le client souscrit au service téléphonique résidentiel de l'entreprise : Oui, Non.	Object
Avg Monthly Long Distance Charges	Indique les frais mensuels moyens de longue distance du client, calculés jusqu'à la fin du trimestre.	Float64
Multiple Lines	Indique si le client souscrit à plusieurs lignes téléphoniques auprès de l'entreprise : Oui, Non (si le client n'est pas abonné au service téléphonique résidentiel, cela sera Non).	Object
Internet Service	Indique si le client souscrit au service Internet de l'entreprise : Oui, Non.	Object
Internet Type	Indique le type de connexion Internet du client : DSL, Fibre Optique, Câble (si le client n'est pas abonné au service Internet, cela ne sera Aucun).	Object
Avg Monthly GB Download	Indique le volume moyen de téléchargement en gigaoctet du client, calculé jusqu'à la fin du trimestre spécifié	Float64
Online Security	Indique si le client souscrit à un service de sécurité en ligne supplémentaire fourni par l'entreprise : Oui, Non (si le client n'est pas abonné au service Internet, cela sera Non).	Object
Online Backup	Indique si le client souscrit à un service de sauvegarde en ligne supplémentaire fourni par l'entreprise : Oui, Non (si le client n'est pas abonné au service Internet, cela sera Non).	Object
Device Protection Plan	Indique si le client souscrit à un plan de protection des appareils supplémentaire pour son équipement Internet fourni par l'entreprise : Oui, Non (si le client n'est pas abonné au service Internet, cela sera Non).	Object
Premium Tech Support	Indique si le client souscrit à un plan de support technique supplémentaire de l'entreprise avec des temps d'attente réduits : Oui, Non (si le client n'est pas abonné au service Internet, cela sera Non).	Object
Streaming TV	Indique si le client utilise son service Internet pour diffuser des émissions de télévision auprès d'un fournisseur tiers sans frais supplémentaires : Oui, Non (si le client n'est pas abonné au service Internet, cela sera Non).	Object
Streaming Movies	Indique si le client utilise son service Internet pour diffuser des films auprès d'un fournisseur tiers sans frais supplémentaires : Oui, Non (si le client n'est pas abonné au service Internet, cela sera Non).	Object
Streaming Music	Indique si le client utilise son service Internet pour diffuser de la musique auprès d'un fournisseur tiers sans frais supplémentaires : Oui, Non (si le client n'est pas abonné au service Internet, cela sera Non).	Object

Unlimited Data	Indique si le client a payé des frais mensuels supplémentaires pour avoir des téléchargements/téléversements de données illimités : Oui, Non (si le client n'est pas abonné au service Internet, cela sera Non).	Object
Contract	Indique le type de contrat actuel du client : Mensuel, Un an, Deux ans.	Object
Paperless Billing	Indique si le client a choisi la facturation sans papier : Oui, Non.	Object
Payment Method	Indique comment le client paie sa facture : Retrait Bancaire, Carte de Crédit, Chèque Postal.	object
Monthly Charge	Indique le montant total mensuel actuel que le client doit payer pour l'ensemble de ses services fournis par l'entreprise.	Float64
Total Charges	Indique les frais totaux du client, calculés jusqu'à la fin du trimestre spécifié ci-dessus.	Float64
Total Refunds	Indique les remboursements totaux du client, calculés jusqu'à la fin du trimestre spécifié ci-dessus.	Float64
Total Extra Data Charges	Indique les frais totaux du client pour les téléchargements de données supplémentaires par rapport à ceux spécifiés dans son plan, à la fin du trimestre spécifié ci-dessus.	Int64
Total Long Distance Charges	Indique les frais totaux du client pour les appels longue distance par rapport à ceux spécifiés dans son plan, à la fin du trimestre spécifié ci-dessus.	Float64
Total Revenue	Indique le revenu total de l'entreprise provenant de ce client, calculé jusqu'à la fin du trimestre spécifié ci-dessus (Frais Totaux - Remboursements Totaux + Frais Supplémentaires pour Données + Frais pour Longue Distance)	Float64
Customer Status	Indique le statut du client à la fin du trimestre : Résilié, Resté, Rejoint. Une catégorie générale pour la raison de résiliation du client, qui est demandée lorsqu'il quitte l'entreprise : Attitude, Concurrent, Insatisfaction, Autre, Prix (directement lié à la Raison de Résiliation).	Object
Churn Category	Une catégorie générale pour la raison de résiliation du client, qui est demandée lorsqu'il quitte l'entreprise : Attitude, Concurrent, Insatisfaction, Autre, Prix (directement lié à la Raison de Résiliation).	Object
Churn Reason	La raison spécifique pour laquelle le client quitte l'entreprise, qui est demandée lorsqu'il quitte l'entreprise	Object

Table 1: Description des colonnes de dataset

3.1.2. Visualisation de données

La visualisation des données constitue un élément fondamental dans le domaine du machine learning. Elle joue un rôle essentiel en permettant une meilleure compréhension des caractéristiques des données, l'identification de tendances, la détection d'anomalies, et facilite la prise de décisions éclairées tout au long du processus d'apprentissage automatique.

Lorsqu'il s'agit de représenter graphiquement des données, la distinction entre données numériques et catégorielles est souvent cruciale.

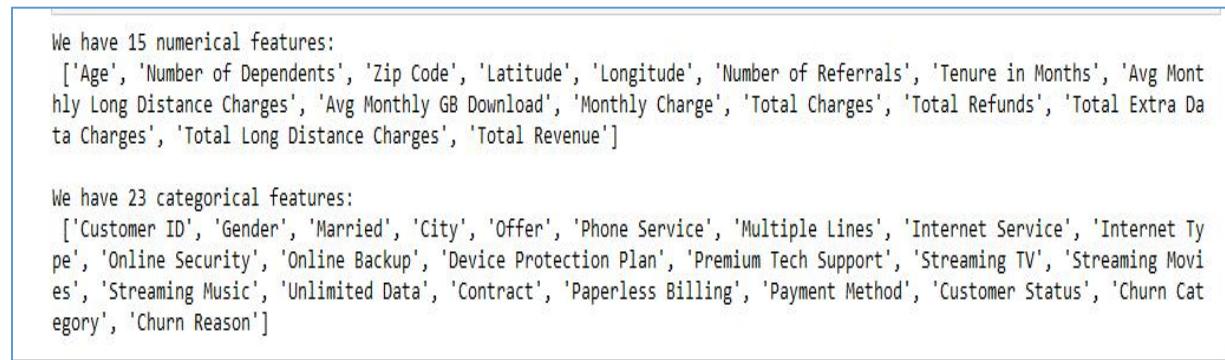


Figure 7: Distinction entre données numériques et catégorielles

À cet effet, l'utilisation d'un diagramme en barres, également connu sous le nom de bar plot, se révèle pertinente. Dans ce contexte, nous avons mis en place une fonction `bar_plot` dédiée, conçue pour créer des diagrammes en barres et ainsi visualiser la distribution des différentes colonnes du jeu de données. Cette approche permet d'appréhender rapidement les variations et les proportions au sein des différentes catégories, offrant ainsi une représentation visuelle claire et concise des informations contenues dans la dataset.

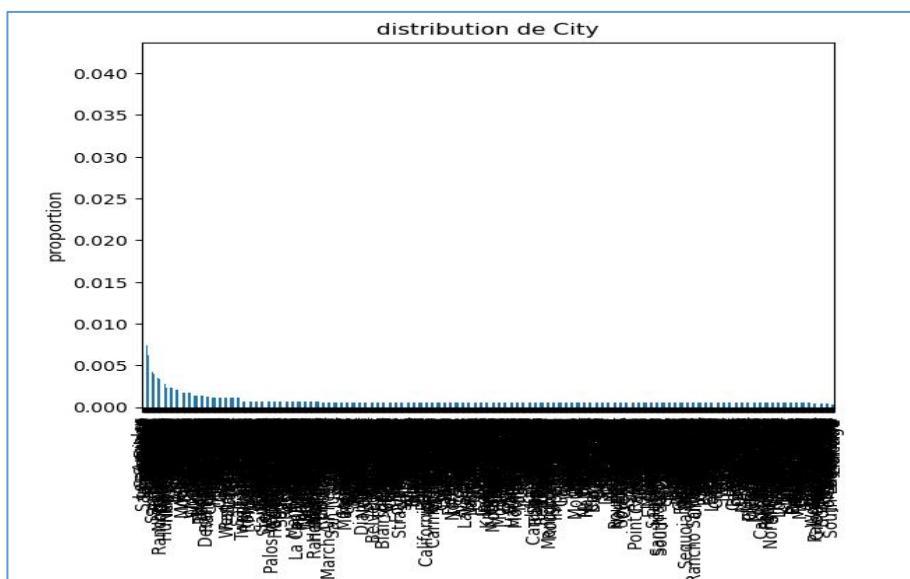


Figure 8: Distribution de variable City

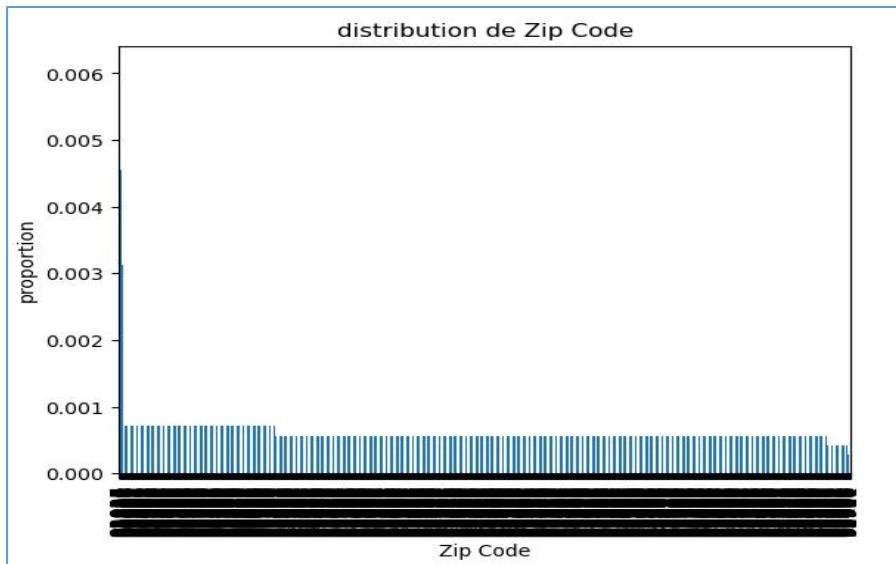


Figure 9: Distribution de variable Zip Code

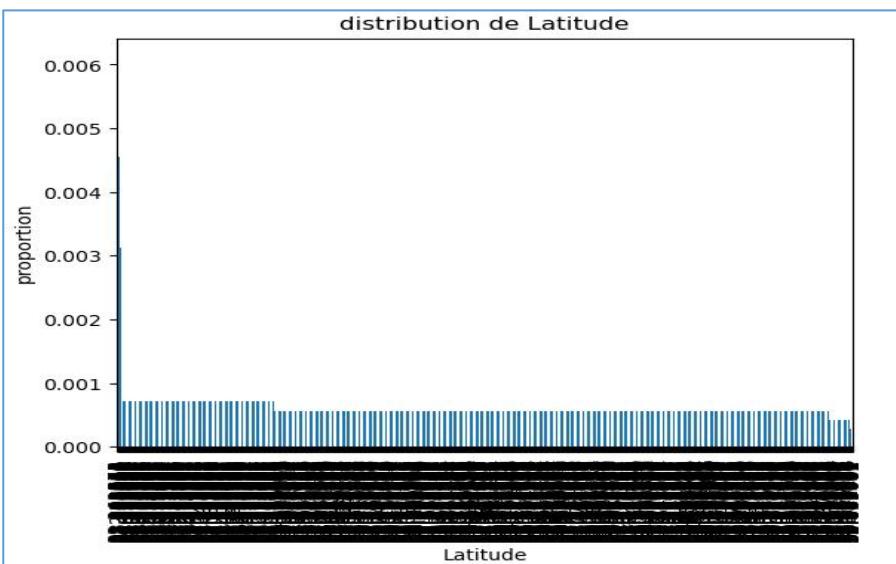


Figure 10: Distribution de variable Latitude

- Nous avons observé dans les 3 diagrammes que les variables comme **City**, **Zip Code**, et **Latitude** sont des informations géographiques qui peuvent ne pas avoir d'impact direct sur la décision de résiliation. Les clients peuvent quitter ou rester pour des raisons liées aux services, aux coûts, à la satisfaction, etc., qui ne sont pas directement liées à leur emplacement géographique.

Concernant les valeurs aberrants de la dataset avant nettoyage, Ce graphique(**Figure 11**) identifie les valeurs aberrantes dans les caractéristiques numériques en fonction du statut du client (actif ou résilié) à l'aide des box-plot.

- D'après cette figure nous avons plusieurs valeurs aberrantes dans les variables: '**Number of Dependents**', '**Avg Monthly GB Download**', '**Total Refunds**', '**Total Long Distance Charges**', '**Total Revenue**'.

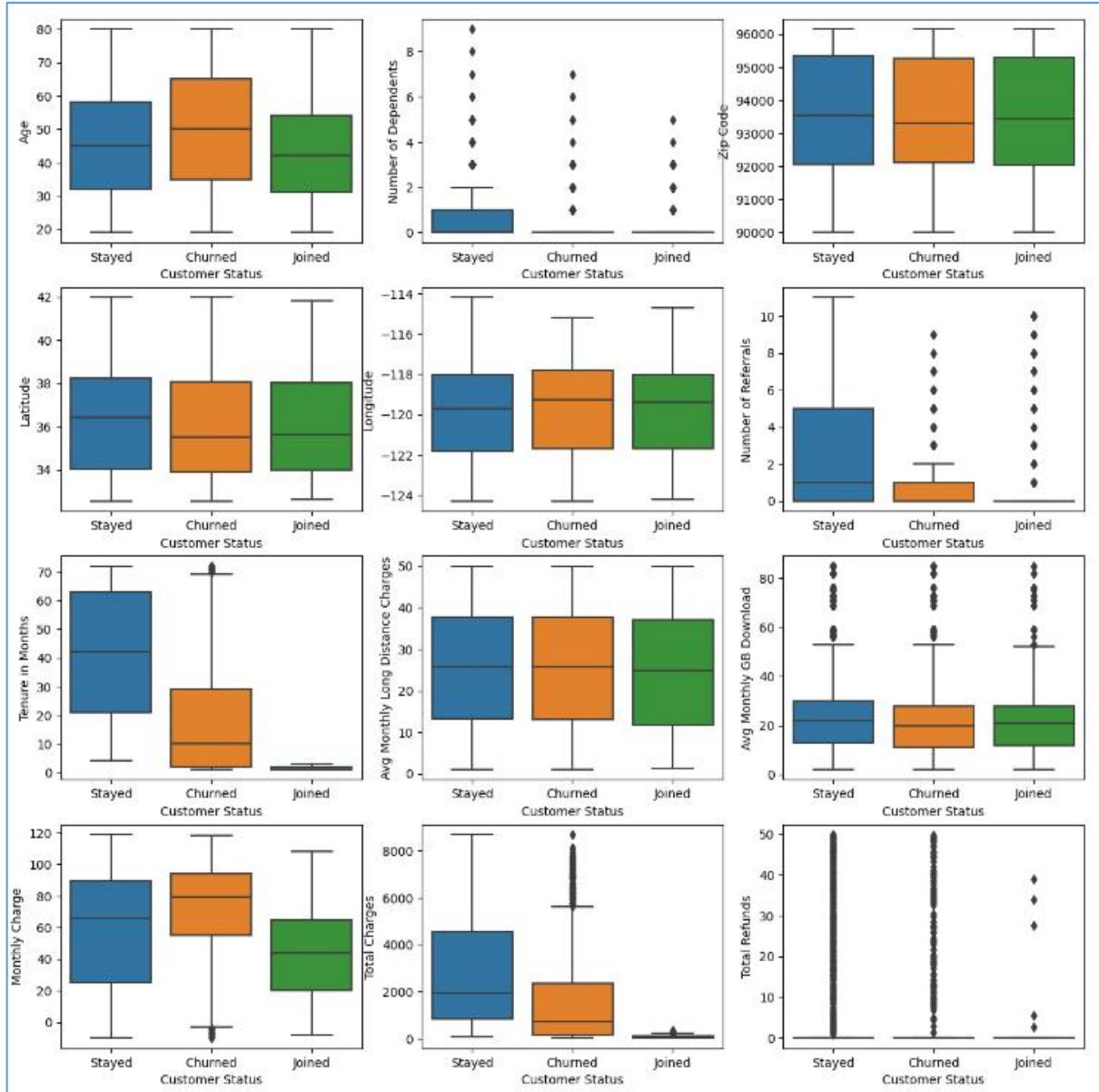


Figure 11: Visualisation des valeurs aberrants

Pour l'étape de la visualisation des valeurs manquantes, nous avons utilisé la bibliothèque **missingno** avec une matrice qui visualise la répartition des ces données manquantes dans la dataset. Les cases sont colorées en blanc là où les données sont présentes et en noir là où les données sont manquantes.

Bandes Horizontales Noires : Les bandes horizontales noires indiquent des lignes complètes sans aucune valeur manquante. Ces lignes sont généralement complètes pour toutes les colonnes.

Bandes Verticales Noires : Les bandes verticales noires indiquent des colonnes complètes sans aucune valeur manquante. Ces colonnes ne contiennent aucune valeur manquante dans l'ensemble du jeu de données.

Bandes Blanches : Les bandes blanches indiquent la présence de données manquantes. La largeur de la bande blanche représente le pourcentage de données manquantes pour une colonne donnée.

Axes : Les axes X et Y de la matrice représentent respectivement les colonnes et les lignes du DataFrame.

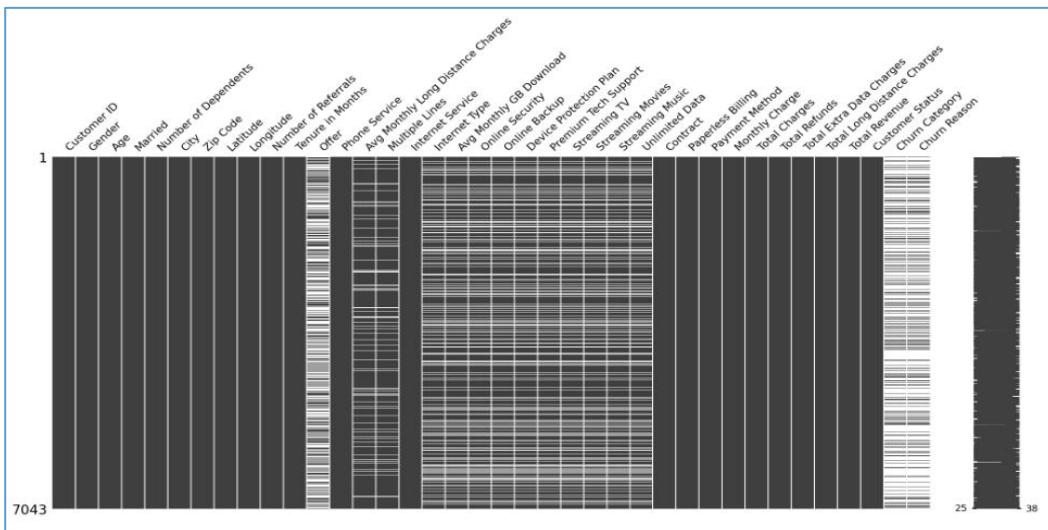


Figure 12: Matrice de visualisation des valeurs manquantes

- La matrice générée, révèle des zones de l'ensemble de données où des informations sont absentes. Les cases blanches indiquent les emplacements où des valeurs sont manquantes, tandis que les cases noires représentent les données disponibles.
- Dans l'ensemble, plusieurs colonnes présentent des lacunes dans les informations, notamment celles liées aux offres, aux services Internet, aux frais mensuels, aux fonctionnalités supplémentaires, aux raisons de résiliation, et autres.

3.2. Analyse Exploratoire

Pour entamer l'étape d'analyse exploratoire, une copie de dataset initial est crée sous le nom **dataset_copy** pour bien identifié l'état des données avant et après sans toucher le dataset d'origine.

```

1 dataset_copy=dataset.copy()
2 dataset_copy

```

	Customer ID	Gender	Age	Married	Number of Dependents	City	Zip Code	Latitude	Longitude	Number of Referrals	...	Payment Method	Monthly Charge	Total Charges	Total Refunds	Total Extra Data Charges
0	0002-ORFBO	Female	37	Yes	0	Frazier Park	93225	34.827662	-118.999073	2	...	Credit Card	65.60	593.30	0.00	0
1	0003-MKNFE	Male	46	No	0	Glendale	91208	34.162515	-118.203869	0	...	Credit Card	-4.00	542.40	38.33	10
2	0004-TLHLJ	Male	50	No	0	Costa Mesa	92627	33.646672	-117.922613	0	...	Bank Withdrawal	73.90	280.85	0.00	0
3	0011-IGKFF	Male	78	Yes	0	Martinez	94553	38.014457	-122.115432	1	...	Bank Withdrawal	98.00	1237.85	0.00	0
4	0013-EXCHZ	Female	75	Yes	0	Camarillo	93010	34.227846	-119.079903	3	...	Credit Card	83.90	267.40	0.00	0
...
7038	9987-LUTYD	Female	20	No	0	La Mesa	91941	32.759327	-116.997260	0	...	Credit Card	55.15	742.90	0.00	0
7039	9992-RRAMN	Male	40	Yes	0	Riverbank	95367	37.734971	-120.954271	1	...	Bank Withdrawal	85.10	1873.70	0.00	0
7040	9992-UJOEL	Male	22	No	0	Elk	95432	39.108252	-123.645121	0	...	Credit Card	50.30	92.75	0.00	0
7041	9993-LHIEB	Male	21	Yes	0	Solana Beach	92075	33.001813	-117.263628	5	...	Credit Card	67.85	4627.65	0.00	0
7042	9995-HOTOH	Male	36	Yes	0	Sierra City	96125	39.800599	-120.636358	1	...	Bank Withdrawal	59.00	3707.60	0.00	0

7043 rows x 38 columns

Figure 13 : Copie du dataset

Nous explorons ensemble les étapes suivies comme suit :

- **Suppression des valeurs inutiles :**

Suite à la phase de visualisation, nous avons identifié plusieurs variables qui ne semblent pas influencer la cible (statut du client).

Nous avons supprimé les colonnes inutiles en utilisant la fonction "drop".

Suppression de Colonnes dans la Copie du Dataset

```

]: dataset_copy.drop(['Customer ID','Total Refunds','Zip Code','Latitude','Longitude','Churn Category', 'Churn Reason','Offer
dataset_copy.drop(['City','Avg Monthly Long Distance Charges','Avg Monthly GB Download','Total Extra Data Charges', 'Total
<                                     >

```

Figure 14: Code de suppression des colonnes

- Ainsi, les colonnes restantes dans la dataset sont :

```

Index(['Gender', 'Age', 'Married', 'Number of Dependents', 'Tenure in Months',
       'Phone Service', 'Multiple Lines', 'Internet Service', 'Internet Type',
       'Online Security', 'Online Backup', 'Device Protection Plan',
       'Premium Tech Support', 'Streaming TV', 'Streaming Movies',
       'Streaming Music', 'Unlimited Data', 'Contract', 'Paperless Billing',
       'Payment Method', 'Monthly Charge', 'Total Charges', 'Customer Status'],
      dtype='object')

```

Figure 15: Colonnes restantes dans dataset

- **Traitement du "Total Charges" :**

La variable "Total Charges" dans notre dataset joue un rôle essentiel dans notre analyse, représentant les coûts totaux facturés aux clients. Cependant, lors de l'exploration initiale, nous avons identifié des aspects nécessitant une attention particulière. Cette partie détaille les étapes de traitement appliquées à cette variable pour garantir la fiabilité de nos résultats.

La visualisation ci-dessous, générée à partir d'un displot de la colonne 'Total Charges' dans notre dataset, offre un aperçu de la distribution des montants totaux facturés aux clients.

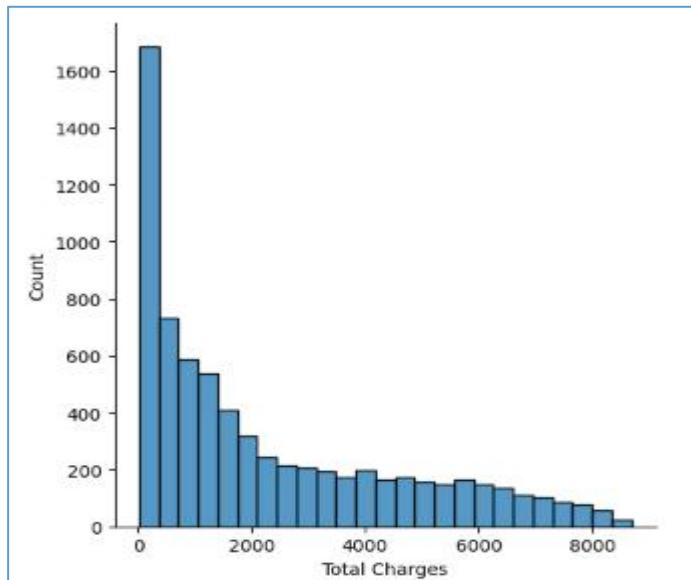


Figure 16: Distribution de Total charges avant transformation

L'observation du graphique révèle une asymétrie non symétrique par rapport au centre, indiquant une possible distribution non normale des charges totales. Cette étape évalue l'asymétrie de la distribution des montants totaux facturés dans notre ensemble de données, permettant ainsi de quantifier la forme et la tendance de cette distribution.

```
1 dataset_copy['Total Charges'].skew()  
0.9637910860571924
```

Figure 17: Code de Skew avant transformation

La distribution initiale présentait une asymétrie notable, comme indiqué par le coefficient de skewness initial de 0.96. Afin de rendre la distribution plus symétrique, nous avons appliqué une transformation racine carrée à la variable "Total Charges". Cette approche vise à atténuer les effets des valeurs extrêmes et à rendre la distribution plus proche de la normalité.

```
dataset_copy['Total Charges']=np.sqrt(dataset_copy['Total Charges'])
```

Figure 18: Code de Transformation de racine carrée

Après la transformation, nous avons effectué des tests de skewness pour évaluer l'asymétrie de la distribution.

```
1 dataset_copy['Total Charges'].skew()  
0.31143070707337783
```

Figure 19: Code de Skew après transformation

Le résultat de la skewness a diminué à 0.31, suggérant une réduction significative de l'asymétrie.

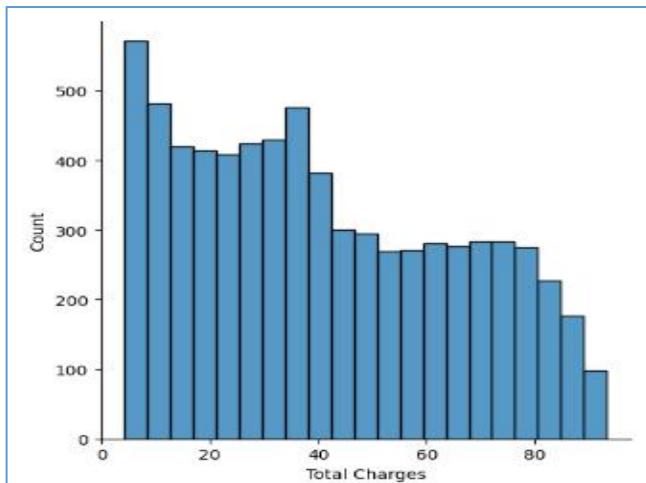


Figure 20: Distribution de Total charges après transformation

Après l'application de la transformation racine carrée, le graphe montre une distribution des charges totales qui semble avoir été ajustée, avec une tendance vers une symétrie par rapport au centre comparé à la distribution initiale.

La commande ci-dessous retourne les valeurs uniques présentes dans la colonne 'Contract', indiquant les différents types de contrats disponibles dans le dataset.

```
1 dataset_copy['Contract'].unique()  
array(['One Year', 'Month-to-Month', 'Two Year'], dtype=object)
```

Figure 21 : Affichage des valeurs uniques de “Contrat”

La commande suivante affiche la colonne 'Tenure in Months', présentant la durée de l'abonnement en mois pour chaque entrée du dataset.

1	dataset_copy['Tenure in Months']
0	9
1	9
2	4
3	13
4	3
	..
7038	13
7039	22
7040	2
7041	67
7042	63
	Name: Tenure in Months, Length: 7043, dtype: int64

Figure 22 : Affichage de la colonne “Tenure in Months”

La commande ci-dessous retourne la colonne 'Monthly Charge', indiquant les frais mensuels associés à chaque abonnement.

1	dataset_copy['Monthly Charge']
0	65.60
1	-4.00
2	73.90
3	98.00
4	83.90
	..
7038	55.15
7039	85.10
7040	50.30
7041	67.85
7042	59.00
	Name: Monthly Charge, Length: 7043, dtype: float64

Figure 23 : Affichage de la colonne “Monthly Charge”

Après avoir visualisé les colonnes 'Tenure in Months' et 'Monthly Charge', nous avons décidé de remplacer la variable 'Total Charges' en utilisant la formule :

$$\text{Total Charge} = \text{Tenure in Months} * \text{Monthly Charge}$$

Cette formule a été appliquée pour transformer la colonne catégorielle 'Total Charge' en une représentation numérique basée sur la durée de l'abonnement et les frais mensuels.

```

1 ind = dataset_copy[dataset_copy['Total Charges'].isnull()].index.tolist()
2 for i in ind:
3     if dataset_copy['Contract'].iloc[i] == 'Two year':
4         dataset_copy['Total Charges'].iloc[i] = dataset_copy['Tenure in Months'].iloc[i] *
5                                         dataset_copy['Monthly Charge'].iloc[i, ] * 24
6     elif dataset_copy['Contract'].iloc[i] == 'One year':
7         dataset_copy['Total Charges'].iloc[i] = dataset_copy['Tenure in Months'].iloc[i] *
8                                         dataset_copy['Monthly Charge'].iloc[i, ] * 12
9     else:
10        dataset_copy['Total Charges'].iloc[i] = dataset_copy['Tenure in Months'].iloc[i] *
11                                         dataset_copy['Monthly Charge'].iloc[i, ]

```

Figure 24 : code de remplacement des Nans en formule

En fonction du type de contrat pour une entrée donnée ('Two year', 'One year', ou autre), le code remplit la valeur manquante dans 'Total Charges' en utilisant une formule spécifique. La formule multiplie la 'Durée de l'Abonnement' par les 'Frais Mensuels', ajustée en fonction de la période du contrat (24 mois pour 'Two year', 12 mois pour 'One year', et la durée du contrat pour les autres).

L'objectif de ce code est de remplir de manière intelligente les valeurs manquantes dans la colonne 'Total Charges' en utilisant des informations contextuelles dérivées d'autres caractéristiques du contrat. Cela garantit une utilisation efficace des données disponibles pour créer des estimations réalistes des coûts totaux facturés.

- ***Encodage des variables numériques et catégorielles:***

En général, L'encodage est essentiel car de nombreux algorithmes d'apprentissage automatique ne peuvent pas traiter les variables catégorielles directement sous forme de chaînes de caractères. Ils nécessitent des données numériques. Ainsi, l'encodage transforme ces informations catégorielles en une forme que les modèles peuvent interpréter et utiliser efficacement.

Nous avons traité les variables qui ont deux modalités.

```
Gender :  
['Female' 'Male']
```

Figure 25: Variable Gender à deux modalités

Encodage de **Gender** est faite sous la manière suivante :

```
dataset_copy['Gender']=dataset_copy['Gender'].apply(lambda row:1 if row=="Female" else 0)
```

Figure 26: Code d'Encodage de Gender

Nous avons utilisé une fonction lambda pour chaque élément de la colonne 'Gender'. La fonction lambda est définie avec un argument, nommé ici 'row', qui correspond à la valeur de chaque entrée de la colonne 'Gender'.

- Si la valeur de row est égale à "Female", la fonction renvoie 1.
- Sinon, elle renvoie 0.

Suivant par le traitement des colonnes qui ont deux modalités, à l'exception de la colonne 'Gender':

```
[ 'Married',
  'Phone Service',
  'Multiple Lines',
  'Internet Service',
  'Online Security',
  'Online Backup',
  'Device Protection Plan',
  'Premium Tech Support',
  'Streaming TV',
  'Streaming Movies',
  'Streaming Music',
  'Unlimited Data',
  'Paperless Billing']
```

Figure 27: Autres variables à deux modalités que Gender

- Les modalités de ces colonnes sont "YES" et "NO". Nous avons remplacé "YES" par 1 et "NO" par 0.

Pour l'encodage des valeurs catégorielles, lorsqu'une colonne présente plus de deux modalités, nous lui attribuons des valeurs arbitraires en utilisant le codage catégorique (cat code). Cette méthode, tout en codant les catégories, remplace également les valeurs manquantes par -1.

Nous avons également remplacé les valeurs manquantes codées avec -1 par "na". Dans la section suivante dédiée à l'analyse des valeurs nulles, nous prévoyons de les traiter en les remplaçant par la moyenne.

Ensuite, nous avons remplacé les valeurs nulles de **Internet Type** par la moyenne. Après avoir effectué l'analyse et le nettoyage des données, la dataset est désormais clair. Voici le résultat affiché dans cette figure.

```
In [41]:  ► dataset_copy.isnull().sum()
Out[41]: Gender          0
Age            0
Married        0
Number of Dependents 0
Tenure in Months 0
Phone Service   0
Multiple Lines  0
Internet Service 0
Internet Type    0
Online Security  0
Online Backup    0
Device Protection Plan 0
Premium Tech Support 0
Streaming TV     0
Streaming Movies 0
Streaming Music   0
Unlimited Data   0
Contract         0
Paperless Billing 0
Payment Method   0
Monthly Charge   0
Total Charges    0
Customer Status  0
dtype: int64
```

Figure 28: Résultat de traitement des Nulles

Passons à l'analyse de corrélation dans l'étape suivante:

- **Analyse de corrélations :**

Le Heatmap ci-dessus visualise la matrice de corrélation entre toutes les variables de dataset. Les valeurs annotées indiquent le degré de corrélation, allant de -1 à 1. Une couleur plus foncée représente une corrélation plus forte. Ce graphique permet une identification visuelle rapide des relations entre les différentes variables.

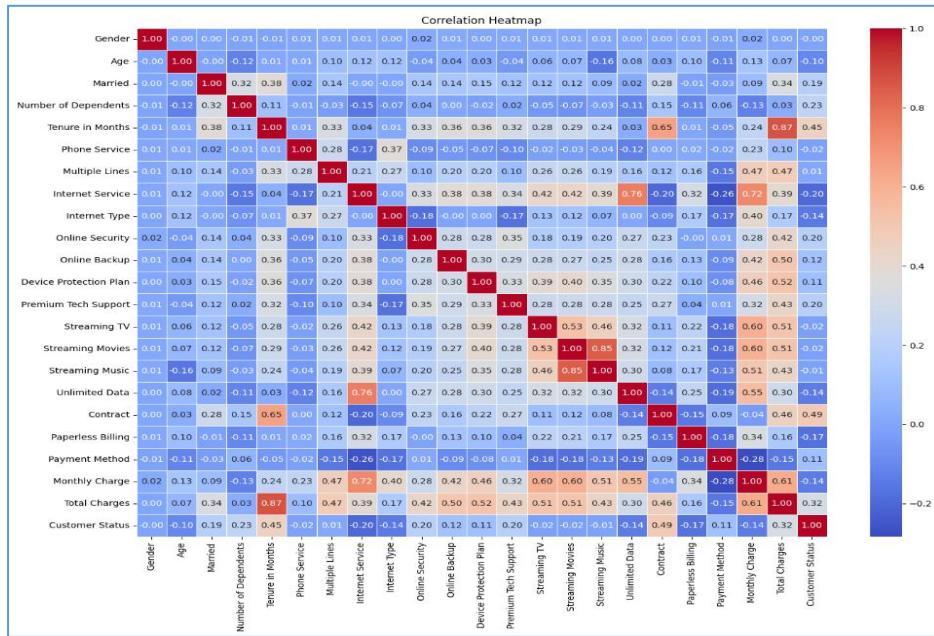


Figure 29: Matrice de corrélation

- **Analyse de répartition :**

Nous avons utilisé une approche interactive pour examiner la distribution de chaque variable dans le dataset. À l'aide de la bibliothèque ipywidgets, nous avons généré des **count plots** pour visualiser graphiquement la répartition. L'utilisateur peut sélectionner la variable à explorer, offrant une flexibilité dans l'analyse.

Exemples de Graphes :

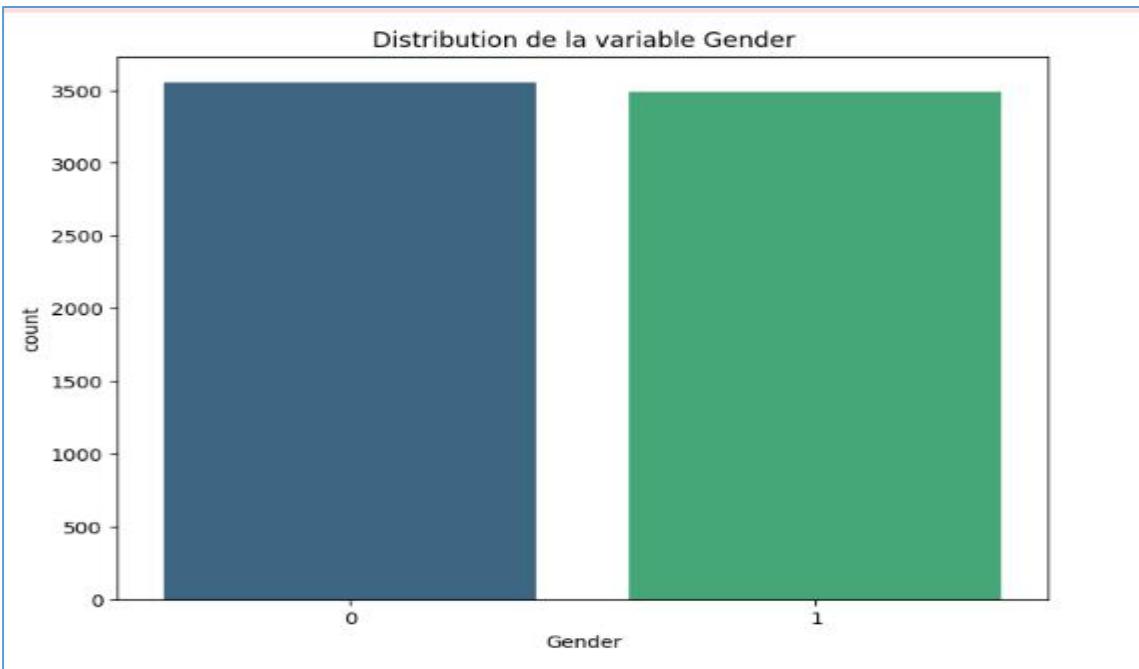


Figure 30: Distribution de la Variable Gender

Ce graphique illustre la répartition des genres dans la dataset. Les barres montrent la fréquence de chaque genre.

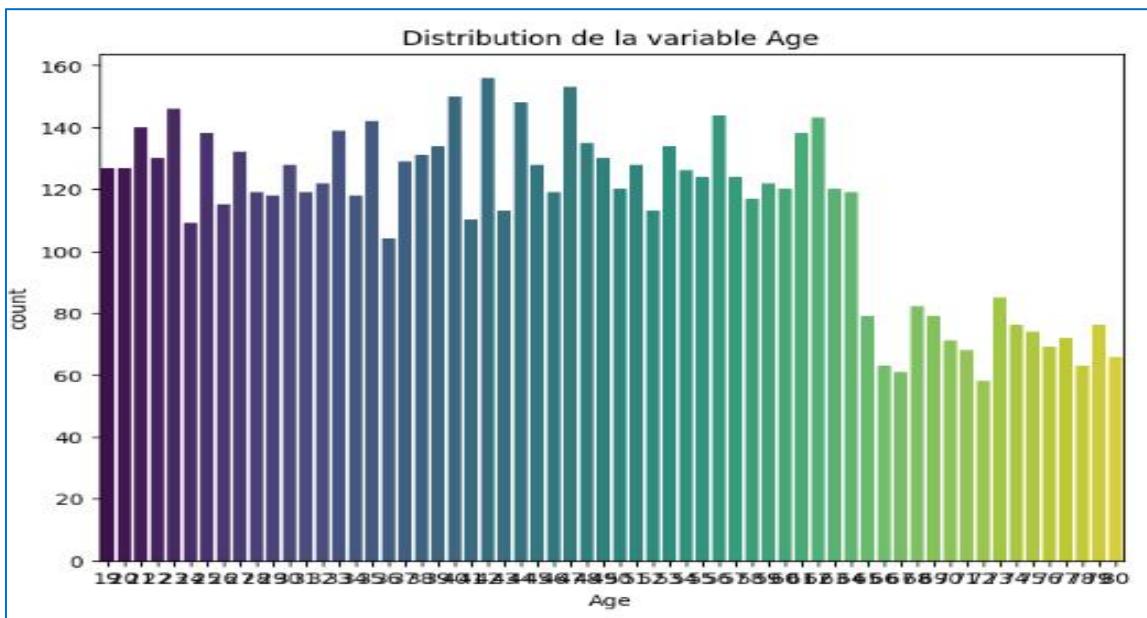


Figure 31: Distribution de la Variable Age

Ce dernier graphique présente la distribution des âges. Il peut aider à identifier les tendances d'âge dans la dataset.

Analyse Statistique :

Nous avons également effectué une analyse statistique des variables numériques, calculant la moyenne, la médiane et l'écart-type. Ces mesures fournissent des insights sur la tendance centrale, la dispersion et la forme de la distribution.

- **Moyenne :**

La moyenne, également connue sous le nom de moyenne arithmétique, représente la valeur obtenue en additionnant toutes les observations d'une variable numérique et en divisant le total par le nombre d'observations. Elle offre une mesure de la tendance centrale des données et est sensible aux valeurs extrêmes.

- **Médiane :**

La médiane est la valeur centrale d'un ensemble de données trié par ordre croissant. Si le nombre d'observations est pair, la médiane est la moyenne des deux valeurs centrales. Cette mesure de tendance centrale est robuste aux valeurs extrêmes et divise la distribution en deux parties égales.

- **Écart-type :**

L'écart-type quantifie la dispersion des valeurs par rapport à la moyenne. Il indique à quel point les valeurs d'un ensemble de données sont éloignées de la moyenne. Un écart-type plus élevé suggère une dispersion plus grande des données, tandis qu'un écart-type plus faible indique une dispersion moindre.

- **Tableau des Statistiques :**

	Variable	Moyenne	Médiane	Écart-type
0	Gender	0.495244	0.000000	0.500013
1	Age	46.509726	46.000000	16.750352
2	Married	0.483033	0.000000	0.499748
3	Number of Dependents	0.468692	0.000000	0.962802
4	Tenure in Months	32.386767	29.000000	24.542061
5	Phone Service	0.903166	1.000000	0.295752
6	Multiple Lines	0.421837	0.000000	0.493888
7	Internet Service	0.783331	1.000000	0.412004
8	Internet Type	1.399674	1.399674	0.650925
9	Online Security	0.286668	0.000000	0.452237
10	Online Backup	0.344881	0.000000	0.475363
11	Device Protection Plan	0.343888	0.000000	0.475038
12	Premium Tech Support	0.290217	0.000000	0.453895
13	Streaming TV	0.384353	0.000000	0.486477
14	Streaming Movies	0.387903	0.000000	0.487307
15	Streaming Music	0.353259	0.000000	0.478016
16	Unlimited Data	0.673719	1.000000	0.468885
17	Contract	0.754792	0.000000	0.848468
18	Paperless Billing	0.592219	1.000000	0.491457
19	Payment Method	0.499645	0.000000	0.599483
20	Monthly Charge	63.596131	70.050000	31.204743
21	Total Charges	40.967710	37.343674	24.538001
22	Customer Status	1.404799	2.000000	0.878514

Table 2: Table d'analyse statistiques

Interprétation des Résultats :

- ✓ **Gender** : La variable "Gender" présente une répartition équilibrée entre les genres, avec une moyenne de 0.495, indiquant une proportion relativement égale de chaque genre dans l'ensemble de données.
- ✓ **Age** : L'âge moyen est de 46 ans, avec une médiane proche, suggérant une distribution relativement symétrique des âges dans l'ensemble de données.
- ✓ **Married** : La variable "Married" montre une distribution équilibrée, avec une moyenne de 0.483, indiquant que la majorité des individus ne sont pas mariés.
- ✓ **Number of Dependents** : La moyenne de 0.469 indique qu'en moyenne, les individus ont moins d'une personne à charge, avec une distribution légèrement asymétrique.

- ✓ **Tenure in Months** : La durée moyenne de l'abonnement est de 32 mois, avec une médiane de 29 mois, suggérant une distribution légèrement étirée vers la droite.
- ✓ **Phone Service** : La plupart des individus ont un service téléphonique (moyenne = 0.903), indiquant une forte présence de ce service dans l'ensemble de données.
- ✓ **Multiple Lines** : La variable "Multiple Lines" montre une distribution bimodale, avec une majorité d'individus ayant une seule ligne et une proportion significative ayant plusieurs lignes.
- ✓ **Internet Service** : Environ 78% des individus ont un service Internet (moyenne = 0.783), montrant une forte présence de ce service.
- ✓ **Internet Type** : La moyenne de 1.40 suggère une concentration autour d'un type d'accès à Internet spécifique, nécessitant une exploration plus approfondie.
- ✓ **Online Security** : La variable "Online Security" présente une distribution asymétrique, avec une majorité d'individus n'ayant pas de sécurité en ligne.
- ✓ **Online Backup** : La distribution de la variable "Online Backup" est asymétrique, avec une majorité d'individus n'ayant pas de sauvegarde en ligne.
- ✓ **Device Protection Plan** : La majorité des individus n'ont pas de plan de protection pour leurs appareils, comme indiqué par la distribution asymétrique de "Device Protection Plan".
- ✓ **Premium Tech Support** : Une grande partie des individus n'a pas de support technique premium, comme indiqué par la distribution asymétrique de "Premium Tech Support".
- ✓ **Streaming TV** : La variable "Streaming TV" montre une distribution bimodale, indiquant deux groupes distincts d'individus en fonction de leur utilisation du service de streaming TV.
- ✓ **Streaming Movies** : La distribution de la variable "Streaming Movies" est similaire à celle de "Streaming TV", avec une séparation nette en deux groupes.
- ✓ **Streaming Music** : La majorité des individus n'ont pas de service de streaming musical, comme indiqué par la distribution asymétrique de "Streaming Music".
- ✓ **Unlimited Data** : Environ 67% des individus ont un accès illimité aux données, montrant une présence significative de cette fonctionnalité.
- ✓ **Contract** : La variable "Contract" présente une forte concentration autour de la valeur 0.75, indiquant une majorité d'individus avec des contrats.

- ✓ **Paperless Billing** : Environ 59% des individus utilisent des factures sans papier, montrant une adoption significative de cette méthode.
- ✓ **Payment Method** : La distribution de la variable "Payment Method" suggère une répartition relativement équilibrée des méthodes de paiement.
- ✓ **Monthly Charge** : Le coût moyen mensuel est d'environ 63.60, avec une dispersion importante, comme indiqué par l'écart-type élevé.
- ✓ **Total Charges** : Les "Total Charges" ont une moyenne de 40.97, mais avec une dispersion importante, suggérant une variabilité significative dans les montants totaux facturés.
- ✓ **Customer Status** : La variable "Customer Status" montre une distribution bimodale, indiquant deux groupes distincts de statuts clients.

● **Traitement des valeurs aberrantes :**

Après avoir détecté graphiquement les valeurs aberrantes à l'aide des boîtes à moustaches (box-plots) en tant que première étape de compréhension et de description du jeu de données, nous avons identifié la présence en cours de valeurs aberrantes exclues de leur contexte au sein d'une colonne. Cette colonne est intitulée "Number of Dependents", Le processus de son traitement se résume à l'utilisation de **la méthode de Capping**: également appelée troncature, consistant à définir une limite supérieure et/ou inférieure pour les données. Cette limite a été ajustée à 1 pour répondre à nos besoins, car elle offre de meilleurs résultats en termes de minimisation des valeurs extrêmes.

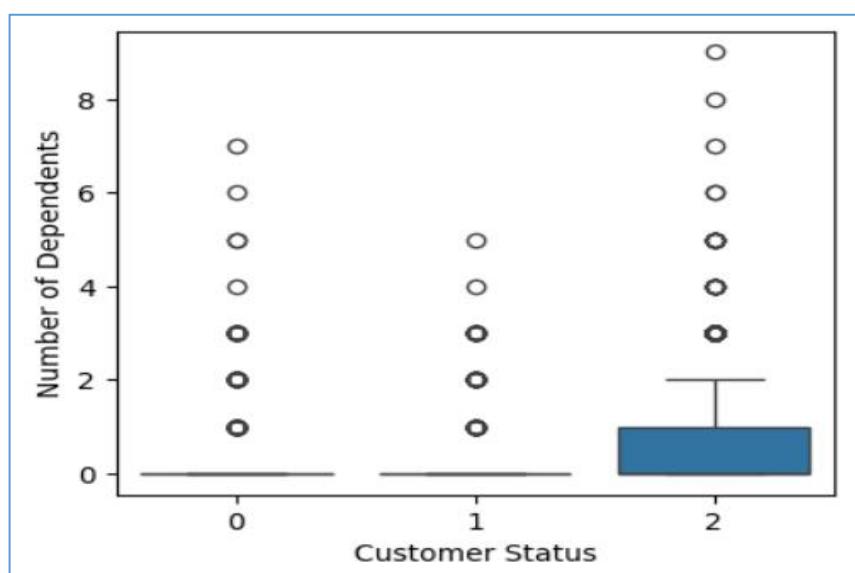


Figure 32: Box-Plot du Number of Dependents avant traitement des aberrantes

Ensuite, un test Z-score mesurant la déviation d'une valeur par rapport à la moyenne des valeurs de "Number of Dependents" a été effectué pour identifier les valeurs aberrantes qui seront remplacées par -seuil si elles sont inférieures à la limite et + seuil si elles sont supérieures à la limite. Cela revient à restreindre toutes les valeurs dans un intervalle [-seuil, +seuil]. Enfin, la colonne originale est remplacée par les valeurs nettoyées et limitées.

En résumé, l'identification des observations aberrantes s'effectue par la méthode des Z-scores, la détermination des seuils appropriés pour le capping se fait en testant plusieurs exemples, et le remplacement des valeurs aberrantes consiste à fixer les valeurs supérieures au seuil supérieur à ce seuil et les valeurs inférieures au seuil inférieur à ce seuil.

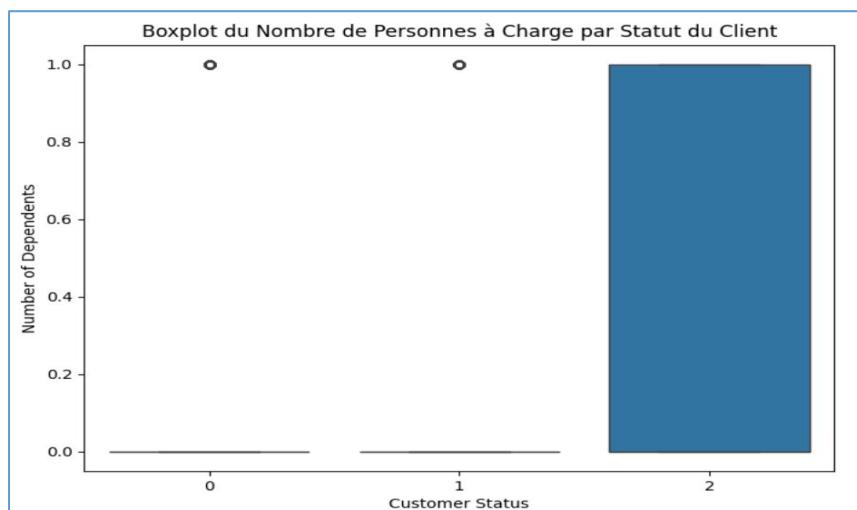


Figure 33: Box-Plot du Number of Dependents après traitement des aberrantes

- Après l'application de cette approche, le box-plot du "Number of Dependents" en fonction de notre variable cible (Customer Status) ne montre plus de valeurs extrêmes, comme le montrent les observations avant et après dans les figures illustrées auparavant.

- **Traitement du Déséquilibre des Données :**

Lors de l'entraînement de modèles d'apprentissage automatique, le déséquilibre des données pose des défis majeurs, car ces modèles ont naturellement tendance à être biaisés en faveur de la classe majoritaire, ce qui peut altérer les résultats et affecter la performance du modèle.

L'utilisation de graphiques tels que le Count-plot avant l'équilibrage s'avère essentielle pour identifier les classes majoritaires et minoritaires dans notre jeu de données, mettant ainsi en lumière le déséquilibre existant illustré dans la figure ci-dessous:

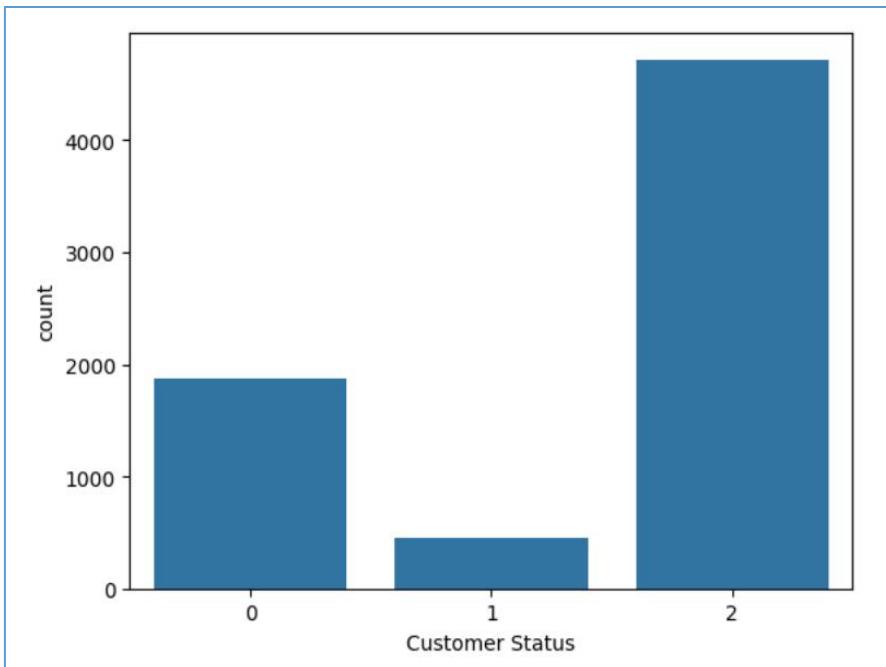


Figure 34: Count-Plot de Y identifiant le déséquilibre existant

Afin de résoudre ce problème, nous avons opté à l'utilisation de la méthode de Sur-échantillonnage, également appelée Oversampling, en particulier à l'aide de l'implémentation de BorderlineSMOTE (Synthetic Minority Over-sampling Technique) disponible dans la bibliothèque de machine learning imbalanced-learn.

Le processus de rééquilibrage des données nécessite auparavant la préparation ou formation des caractéristiques X, extraites de notre ensemble de données, en excluant les colonnes "Customer Status", "Gender", ainsi La colonne "Customer Status" est bien spécifiée en tant que cible Y.

Ensuite, nous faisons appel à BorderlineSMOTE qui aide à équilibrer les classes en générant des exemples synthétiques le long de la frontière entre les classes, là où la séparation entre eux est moins évidente.

Nous obtenons le résultat suivante en terme de forme de X et Y:

```
X shape: (14160, 21)
Y shape: (14160,)
```

Figure 35: Vérification d'équilibrage par shape

Et graphiquement L'équilibre est clairement identifiable et la distribution des classes est modifiée.

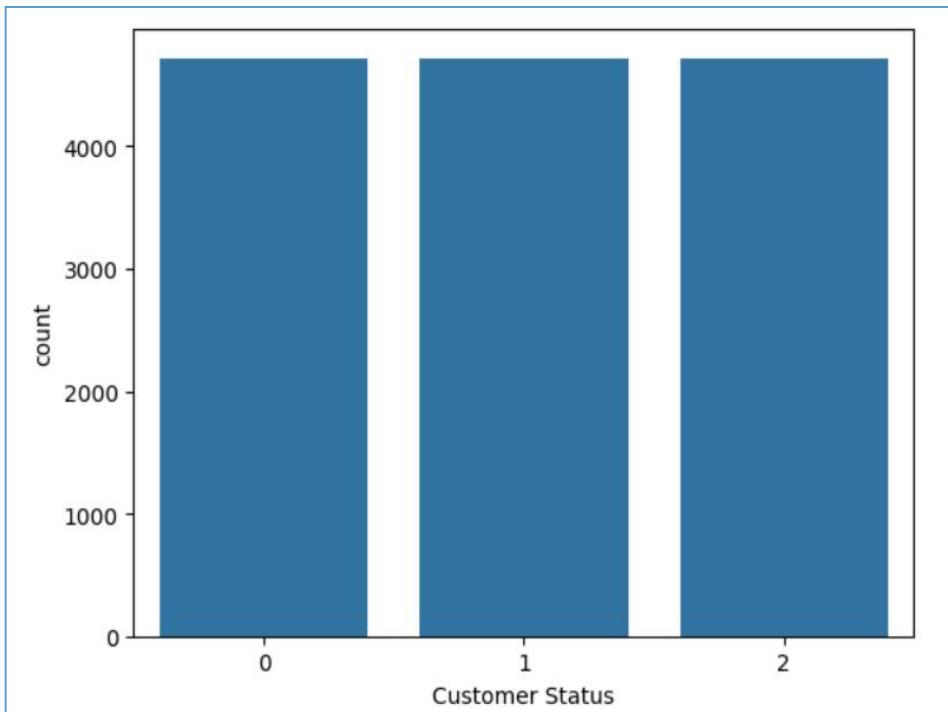


Figure 36: Count-Plot de Y identifiant les classes après rééquilibrage

3.3.Modélisation

- **Division des données :**

Le principe de division des données en machine learning est fondamental pour évaluer la performance d'un modèle de manière impartiale.

La division des données consiste à diviser le jeu de données initial en plusieurs ensembles distincts, généralement trois ensembles principaux : l'ensemble d'entraînement (training set), l'ensemble de validation (validation set), et l'ensemble de test (test set).

```
In [57]: M from sklearn.model_selection import train_test_split
X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size=0.4, random_state=1111,stratify=Y)
X_val, X_test, Y_val, Y_test = train_test_split(X_test, Y_test, test_size=0.5, random_state=1111, stratify=Y_test)
```

Les données sont divisées en trois ensembles distincts : l'ensemble d'entraînement, l'ensemble de validation et l'ensemble de test, avec des proportions respectives de 60%, 20% et 20%. La stratification est utilisée pour maintenir la répartition des classes dans chaque ensemble.

Figure 37: Code de division de données

Ensemble d'entraînement :

- **Objectif :** Ce jeu de données, représentant 60% de l'ensemble global, est principalement destiné à l'entraînement des modèles. Il fournit une base pour que les

algorithmes apprennent les relations complexes entre les différentes caractéristiques et le statut du client.

- **Processus de sélection :** Les données ont été sélectionnées de manière stratifiée pour garantir une représentation équilibrée des différentes classes ou catégories.

Ensemble de validation :

- **Objectif :** Cet ensemble, représentant 20% de l'ensemble global, est spécifiquement dédié à l'ajustement des hyperparamètres des modèles. Son utilisation permet d'optimiser les performances en évitant le sur-ajustement aux données d'entraînement.

Ensemble de test :

- **Objectif :** Les données de test, représentant les 20% restants, sont réservées pour évaluer la performance finale des modèles sur des données réellement inconnues, simulant ainsi les conditions du monde réel.

Choix de l'algorithme d'apprentissage automatique :

Lors de la sélection des algorithmes pour notre problème de prédiction d'attrition client, nous avons opté pour quatre approches distinctes qui ont démontré leur efficacité dans des contextes similaires :

Les Forêts Aléatoires (Random Forest) :

Caractéristiques :

- Algorithme d'ensemble basé sur des arbres de décision.
- Robuste aux overfitting grâce à la construction d'arbres multiples.
- Capable de gérer des ensembles de données complexes avec des interactions non linéaires.
- Performant pour la classification.

La Régression Logistique (Logistic Regression):

Caractéristiques :

- Algorithme linéaire simple et interprétable.
- Bien adapté aux problèmes de classification binaire ou multi-classe.
- Nécessite moins de ressources computationnelles.
- Efficace pour des ensembles de données de taille modérée.

Le gradient boosting :

Le gradient boosting : est une technique d'apprentissage automatique qui appartient à la famille des méthodes d'ensemble. L'objectif principal du gradient boosting est de construire un modèle prédictif fort en combinant plusieurs modèles plus faibles, généralement des arbres de décision, pour améliorer la précision et les performances prédictives.

Le processus de gradient boosting implique plusieurs étapes :

1. **Construction du premier modèle:** Un modèle initial (généralement simple) est construit pour expliquer les tendances globales dans les données.
2. **Calcul du résidu:** Les erreurs (résidus) entre les prédictions du modèle initial et les vraies valeurs sont calculées.
3. **Construction du modèle suivant:** Un nouveau modèle est construit pour prédire les résidus du modèle précédent. Cela se fait en ajustant le modèle pour minimiser les erreurs résiduelles.
4. **Mise à jour des prédictions:** Les prédictions du modèle initial sont mises à jour en ajoutant les prédictions du nouveau modèle ajusté.
5. **Répétition des étapes 2-4:** Les étapes 2 à 4 sont répétées jusqu'à ce qu'un certain critère d'arrêt soit atteint, comme le nombre d'itérations prédéfini.
6. **Combinaison des modèles:** Les modèles individuels sont pondérés et combinés pour former un modèle global.

L'idée clé derrière le "gradient" dans le gradient boosting est d'optimiser les paramètres du modèle de manière itérative en utilisant des méthodes d'optimisation basées sur le gradient, telles que la descente de gradient. Cela permet de minimiser la fonction de perte globale du modèle.

XGBoost (eXtreme Gradient Boosting) :

XGBoost est une implémentation populaire de l'algorithme de gradient boosting. C'est une bibliothèque open-source populaire implémentant l'algorithme de gradient boosting. Développée par Tianqi Chen, XGBoost est particulièrement connue pour sa performance élevée, sa rapidité et son efficacité dans les compétitions de science des données et de machine learning.

Caractéristiques :

- Ensemble d'arbres de décision : Utilisation d'un ensemble de modèles d'arbres de décision, souvent des arbres peu profonds.

- Optimisation basée sur le gradient : Minimisation itérative de la fonction de perte en utilisant des méthodes d'optimisation basées sur le gradient.
- Régularisation : Intégration de termes de régularisation (l1 et l2) pour éviter le surajustement.
- Gestion automatique des valeurs manquantes : Capacité à traiter automatiquement les données avec des valeurs manquantes.
- Polyvalence des tâches : Adapté à la classification, à la régression et au classement.
- Efficacité et parallélisme : Conçu pour être rapide, avec prise en charge du parallélisme pour accélérer l'entraînement.
- Régularisation et validation croisée intégrées : Techniques de régularisation telles que la réduction du nombre d'arbres et fonction de validation croisée intégrée pour estimer le nombre optimal d'itérations.

En résumé, **XGBoost** est apprécié pour sa puissance, sa rapidité, sa capacité à gérer divers types de données, et son efficacité dans la prévention du surajustement grâce à des techniques de régularisation.

● *Evaluation des algorithmes*

Random forest :

✓ **Principe de Random Forest :**

Random Forest est un algorithme d'ensemble basé sur l'idée de la combinaison de multiples arbres de décision. Chaque arbre est formé sur un sous-ensemble aléatoire des données, et le résultat final est obtenu par vote majoritaire.

✓ **Initiation du Modèle :**

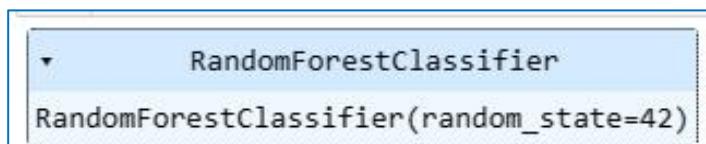


Figure 38: Initiation de RandomForest

✓ **Prédictions sur l'Ensemble de Validation :**

Les prédictions du modèle ont été générées pour l'ensemble de validation (X_{val}).



```
Précision : 0.8919491525423728
```

Figure 39: Précision de RandomForest

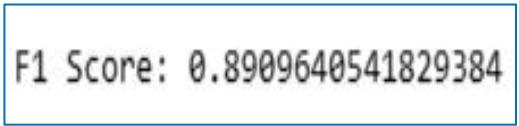
Le modèle Random Forest a atteint une précision de 89.19%, indiquant sa capacité à classer correctement environ 89.19% des exemples de l'ensemble de validation.

✓ **Calcul des Métriques :**

Les métriques suivantes ont été calculées et interprétées :

F1 Score :

Le F1 Score de 0.8909 suggère que le modèle maintient un équilibre entre précision et rappel.



```
F1 Score: 0.8909640541829384
```

Figure 40: F1Score de RandomForest

Rappel :

Le modèle affiche un rappel de 89.19%, indiquant sa capacité à identifier correctement environ 89.19% des cas positifs.



```
Recall : 0.8919491525423728
```

Figure 41: Recall de RandomForest

✓ **Rapport de Classification :**

Le modèle Random Forest démontre une solide performance sur l'ensemble de validation, avec une précision globale de 89%. Il parvient à bien distinguer les différentes classes, notamment avec une précision de 92% pour la Classe 1 et un rappel de 98%. Cette capacité à identifier correctement les exemples positifs contribue à un F1-Score global de 89%, indiquant un bon équilibre entre précision et rappel.

Classification Report:

	precision	recall	f1-score	support
0	0.85	0.82	0.83	944
1	0.92	0.98	0.95	944
2	0.90	0.88	0.89	944
accuracy			0.89	2832
macro avg	0.89	0.89	0.89	2832
weighted avg	0.89	0.89	0.89	2832

Figure 42: Rapport de classification de RandomForest

✓ Matrice de Confusion :

La matrice de confusion révèle les performances du modèle pour chaque classe.

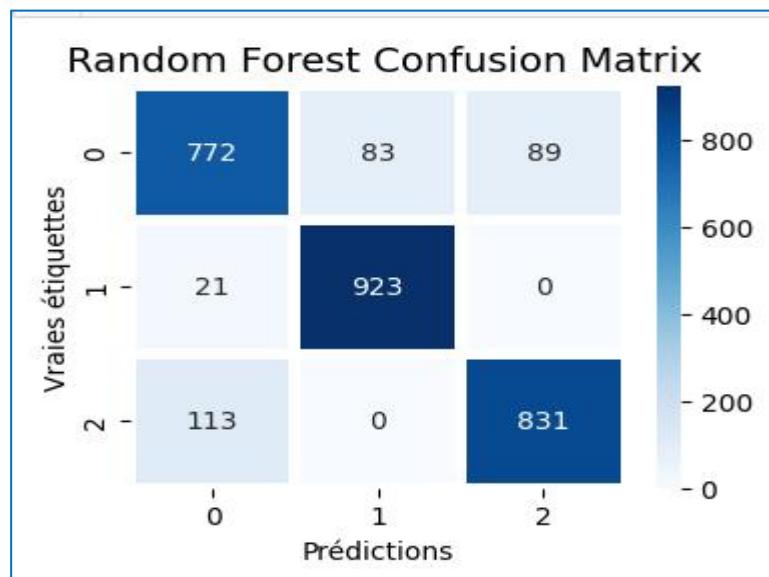


Figure 43: Matrice de confusion de RandomForest

✓ Courbes ROC :

Les courbes ROC ont été générées pour évaluer la capacité du modèle à discriminer entre les différentes classes.

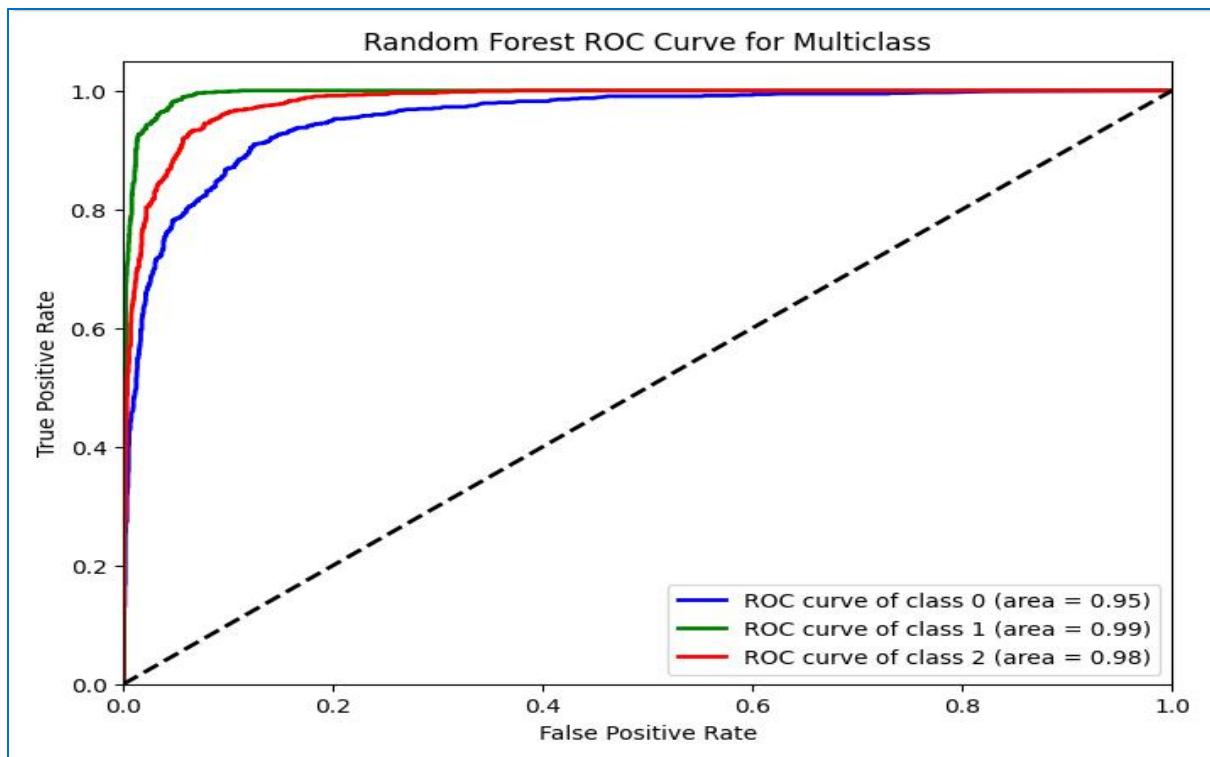


Figure 44: Courbe ROC de RandomForest

Les courbes ROC illustrent graphiquement la capacité du modèle à discriminer entre les classes.

La Régression Logistique :

✓ Principe de la Régression Logistique :

La régression logistique est une technique de modélisation utilisée pour prédire la probabilité d'un événement en fonction de plusieurs variables indépendantes. Contrairement à la régression linéaire, la régression logistique est utilisée pour les problèmes de classification binaire ou multiclasse.

✓ Initiation du Modèle :

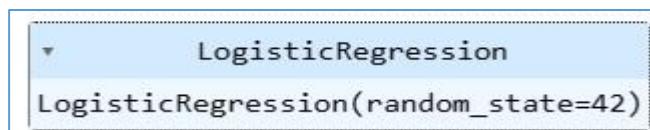


Figure 45: Initiation de LogisticRegression

✓ Prédictions sur l'Ensemble de Validation :

Les prédictions du modèle ont été générées pour l'ensemble de validation (X_{val}).

Précision : 0.794138418079096

Figure 46: Precision de LogisticRegression

Le modèle de régression logistique a atteint une précision de 79.41%, indiquant sa capacité à classer correctement environ 79.41% des exemples de l'ensemble de validation.

✓ **Calcul des Métriques :**

F1 Score :

Avec un F1 Score de 79.10, le modèle maintient un bon équilibre entre précision et rappel.

F1-Score : 0.7910877026833937

Figure 47: F1Score de LogisticRegression

Rappel :

Le modèle affiche un rappel de 79.41%, indiquant sa capacité à identifier correctement environ 79.41% des cas positifs.

Recall: 0.794138418079096

Figure 48: Recall de LogisticRegression

✓ **Rapport de Classification :**

La régression logistique montre une performance solide avec une précision globale de 79%. Elle distingue bien les différentes classes, avec une précision de 81% pour la Classe 1 et un rappel de 94%. Cela se traduit par un F1-Score global de 79%.

Classification Report:				
	precision	recall	f1-score	support
0	0.72	0.66	0.69	944
1	0.81	0.94	0.87	944
2	0.85	0.78	0.82	944
accuracy			0.79	2832
macro avg	0.79	0.79	0.79	2832
weighted avg	0.79	0.79	0.79	2832

Figure 49: Rapport de classification de LogisticRegression

✓ **Matrice de Confusion :**

La matrice de confusion détaille les performances du modèle pour chaque classe, permettant d'identifier les vrais positifs, les vrais négatifs, les faux positifs et les faux négatifs.

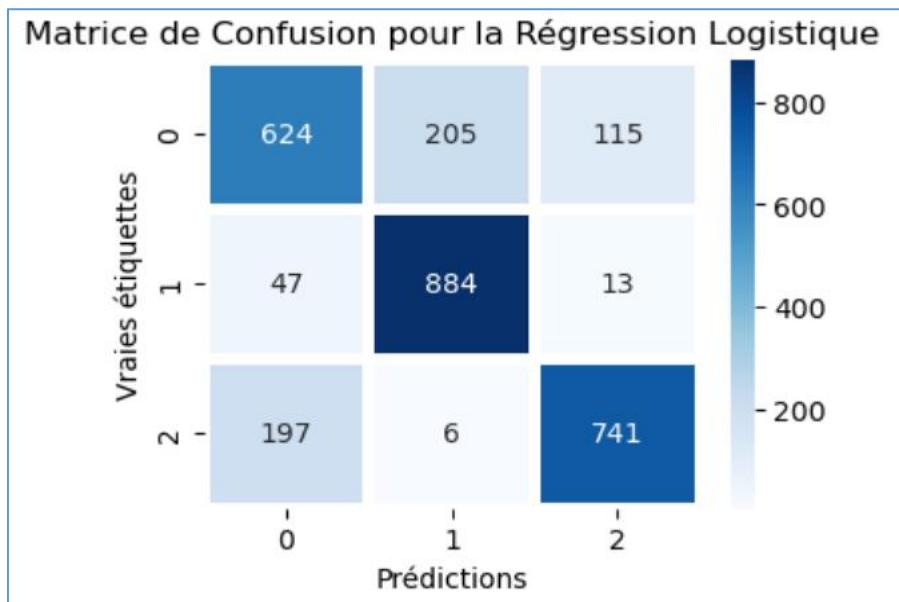


Figure 50: Matrice de confusion de LogisticRegression

✓ **Courbes ROC :**

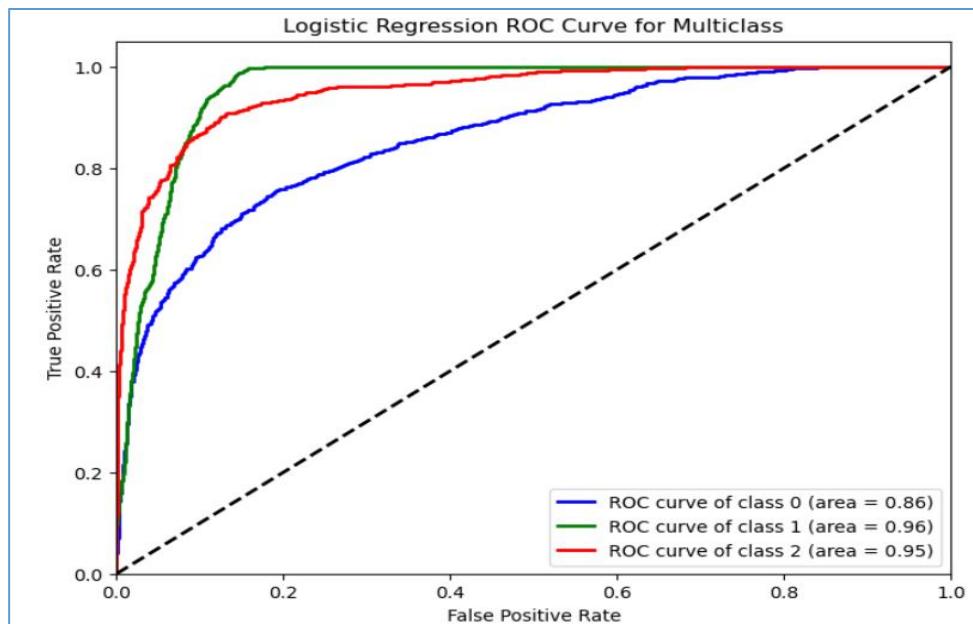


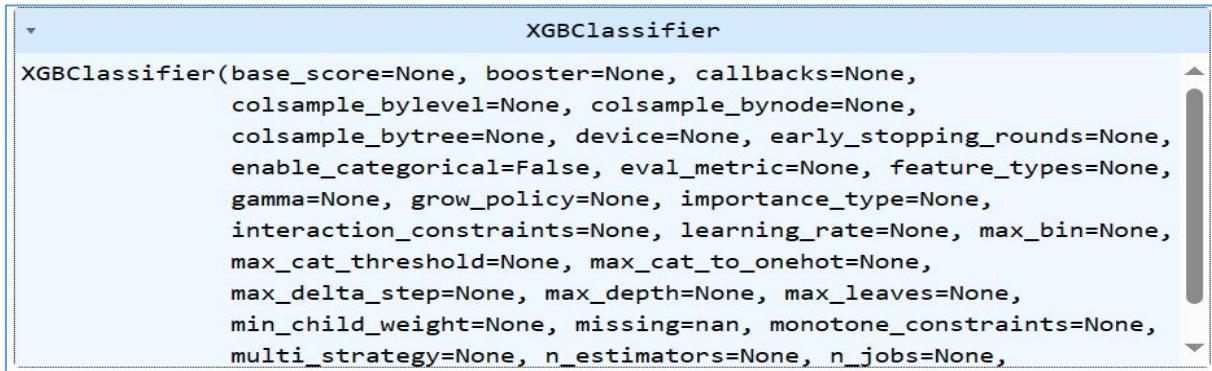
Figure 51: Courbe ROC de LogisticRegression

XGBoost :

✓ Principe de XGBoost :

XGBoost (Extreme Gradient Boosting) est une implémentation populaire de l'algorithme de boosting en machine learning, conçue pour être hautement performante, rapide et extensible. Il combine les avantages des arbres de décision et des techniques de boosting.

✓ Initiation du Modèle :

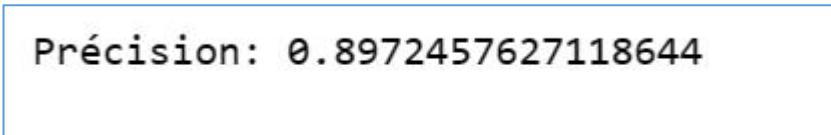


```
XGBClassifier(base_score=None, booster=None, callbacks=None,
              colsample_bylevel=None, colsample_bynode=None,
              colsample_bytree=None, device=None, early_stopping_rounds=None,
              enable_categorical=False, eval_metric=None, feature_types=None,
              gamma=None, grow_policy=None, importance_type=None,
              interaction_constraints=None, learning_rate=None, max_bin=None,
              max_cat_threshold=None, max_cat_to_onehot=None,
              max_delta_step=None, max_depth=None, max_leaves=None,
              min_child_weight=None, missing=nan, monotone_constraints=None,
              multi_strategy=None, n_estimators=None, n_jobs=None,
```

Figure 52: Initiation du XGB

✓ Prédictions sur l'Ensemble de Validation :

Les prédictions du modèle ont été générées pour l'ensemble de validation (X_val).



Précision: 0.8972457627118644

Figure 53: Précision de XGB

Le modèle XGBoost a atteint une précision de 89.72%, indiquant sa capacité à classer correctement environ 89.72% des exemples de l'ensemble de validation.

✓ Calcul des Métriques :

F1 Score :

Avec un F1 Score de 0.896, le modèle maintient un excellent équilibre entre précision et rappel.



F1-Score : 0.8961888555479403

Figure 54: F1Score de XGB

Rappel :

Le modèle affiche un rappel de 89.61%, montrant sa capacité à identifier correctement environ 89.61% des cas positifs.

```
Recall : 0.8972457627118644
```

Figure 55: Recall de XGB

✓ **Rapport de Classification :**

XGBoost présente une performance robuste avec une précision globale de 90%. Il distingue efficacement les différentes classes, avec une précision de 91% pour la Classe 1 et un rappel de 98%. Cela conduit à un F1-Score global de 90%.

Classification Report :				
	precision	recall	f1-score	support
0	0.87	0.82	0.84	944
1	0.91	0.98	0.94	944
2	0.91	0.90	0.90	944
accuracy			0.90	2832
macro avg	0.90	0.90	0.90	2832
weighted avg	0.90	0.90	0.90	2832

Figure 56: Rapport de classification de XGB

✓ **Matrice de Confusion :**

La matrice de confusion détaille les performances du modèle pour chaque classe, permettant de visualiser les vrais positifs, les vrais négatifs, les faux positifs et les faux négatifs.

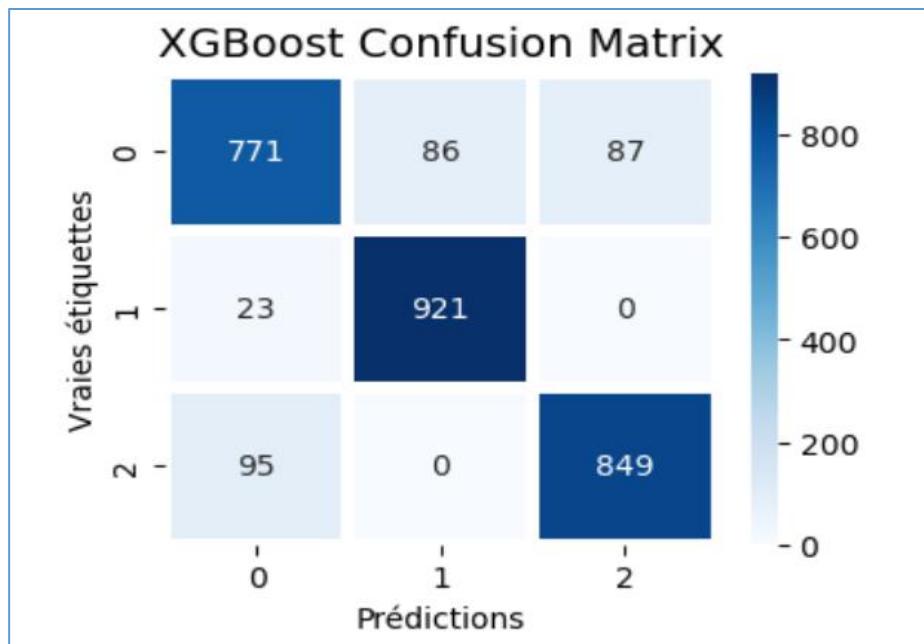


Figure 57: Matrice de confusion de XGB

✓ **Courbes ROC :**

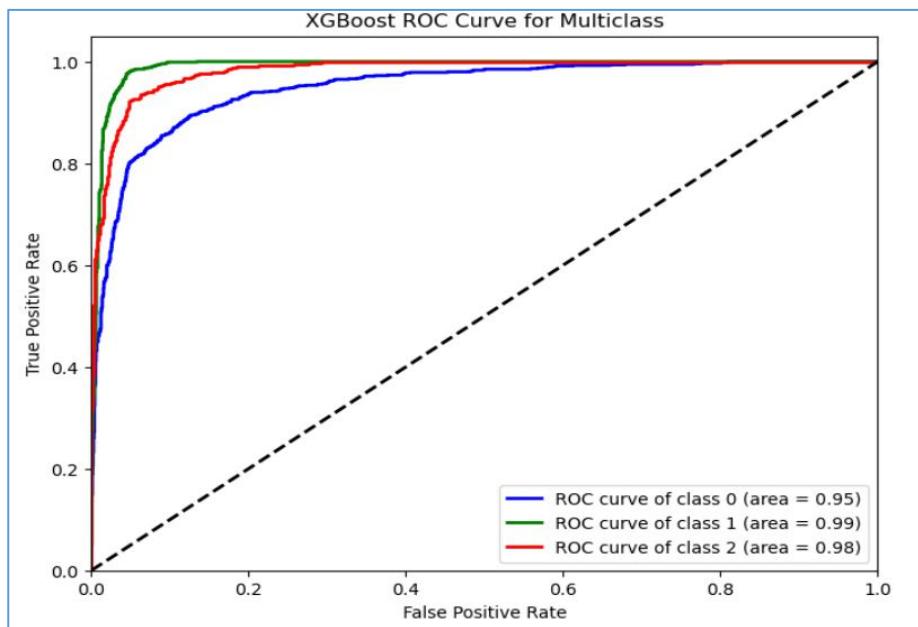


Figure 58: Courbe ROC de XGB

Gradient Boosting :

✓ **Principe de Gradient Boosting :**

Le Gradient Boosting est une technique d'ensemble qui construit un modèle prédictif en combinant les prédictions de plusieurs modèles simples (souvent des arbres de décision) de

manière séquentielle. À chaque étape, le modèle tente de corriger les erreurs des prédictions précédentes, en se concentrant sur les exemples où le modèle a le plus échoué.

✓ **Initialisation du Modèle :**

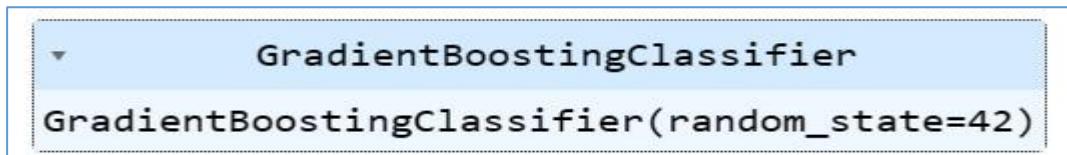


Figure 59: Initiation du Gradient

✓ **Prédictions sur l'Ensemble de Validation :**

Les prédictions du modèle ont été générées pour l'ensemble de validation (X_{val}).

```
Précision : 0.8513418079096046
```

Figure 60: Précision de Gradient

Le modèle basé sur le Gradient Boosting a atteint une précision de 85.13%, ce qui signifie qu'il peut classer correctement environ 85.13% des exemples de l'ensemble de validation.

✓ **Calcul des Métriques :**

F1 Score :

Avec un F1 Score de 0.848, le modèle parvient à maintenir un bon équilibre entre précision et rappel.

```
F1-Score : 0.848425317288205
```

Figure 61: F1Score de Gradient

Rappel :

Le modèle présente un rappel de 85.13%, indiquant sa capacité à identifier correctement environ 85.13% des cas positifs.

```
Recall: 0.8513418079096046
```

Figure 62: Recall de Gradient

✓ **Rapport de Classification :**

Le modèle Gradient Boosting démontre une performance solide avec une précision globale de 85%. Il est efficace pour distinguer les différentes classes, notamment avec une précision de 85% pour la Classe 1 et un rappel de 98%. Cela conduit à un F1-Score global de 85%.

Classification Report:				
	precision	recall	f1-score	support
0	0.81	0.72	0.76	944
1	0.85	0.98	0.91	944
2	0.89	0.85	0.87	944
accuracy			0.85	2832
macro avg	0.85	0.85	0.85	2832
weighted avg	0.85	0.85	0.85	2832

Figure 63: Rapport de classification de Gradient

✓ **Matrice de Confusion :**

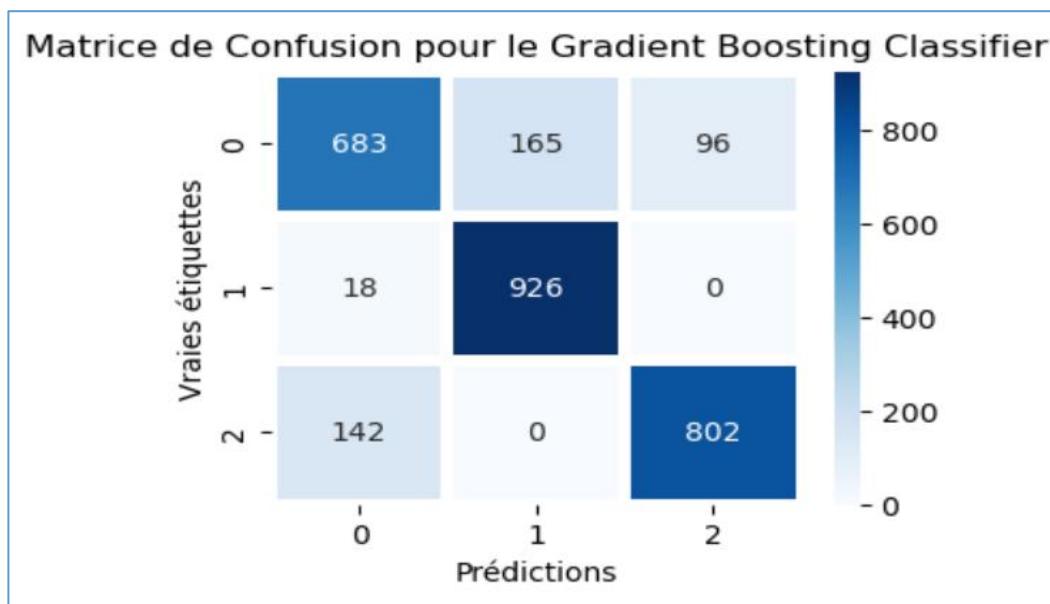


Figure 64: Matrice de confusion de Gradient

✓ Courbes ROC :

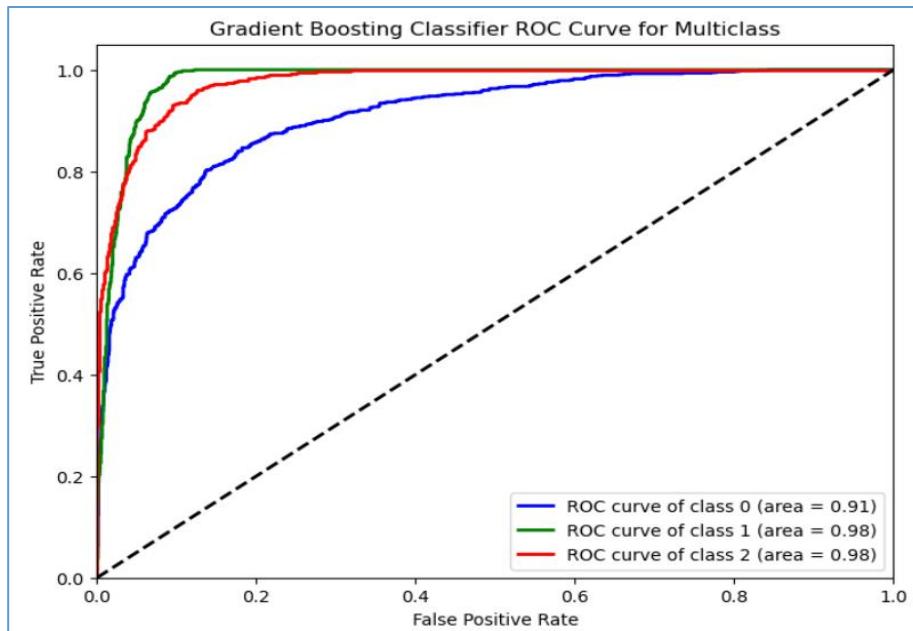


Figure 65: Courbe ROC de Gradient

Chapitre3:

Résultats et Interprétations

1. Comparaison de résultats

Cette section vise à établir une comparaison rigoureuse entre plusieurs modèles, tels que Random Forest, Régression Logistique, Gradient Boosting et XGBoost. Nous explorerons différentes approches, incluant l'utilisation de la bibliothèque Tabulate, Lazypredict, et la validation croisée, pour déterminer le modèle présentant la meilleure précision dans la classification de notre ensemble de validation.

1.1. Comparaison des Modèles avec Tabulate :

Nous avons évalué et comparé les performances de plusieurs modèles, dont Random Forest, Régression Logistique, Gradient Boosting et XGBoost. Chaque modèle a été entraîné sur l'ensemble d'entraînement et évalué sur l'ensemble de validation. Les résultats ont été comparés en termes de précision, et le meilleur modèle a été identifié.

Processus de cette Comparaison :

- ***Importation des Bibliothèques :***

La bibliothèque Tabulate a été importée pour la création de tableaux.

- ***Création du Tableau :***

Les précisions de chaque modèle ont été rassemblées dans un tableau pour une comparaison facile.

- ***Affichage du Tableau :***

Le tableau a été affiché, présentant les précisions de chaque modèle.

- ***Identification du Meilleur Modèle :***

Le modèle avec la meilleure précision a été identifié.

- ***Affichage du Résultat :***

Le tableau et le meilleur modèle avec sa précision ont été affichés.

Modèle	Précision
Random Forest	0.891949
Logistic Regression	0.801201
Gradient Boosting	0.856992
XGBoost	0.89654

Meilleur Modèle : XGBoost avec une Précision de : 0.8965395480225988

Figure 66: Table de comparaison

Le tableau de comparaison montre que XGBoost est le modèle avec la meilleure précision, atteignant 89.65%. Il surpasse ainsi les autres modèles, tels que Random Forest, Régression Logistique et Gradient Boosting. Cette précision élevée suggère que XGBoost est plus performant dans la classification des exemples de l'ensemble de validation.

1.2. Lazy Predict

La comparaison des performances de nos algorithmes est faisable facilement à l'aide de la bibliothèque LazyPredict qui simplifie le processus de sélection de meilleur algorithme en ajustant un grand nombre de modèles de machine learning disponibles dans scikit-learn et à la fois obtenir rapidement leurs performances en utilisant les données fournies par l'affichage des scores de précision, F1 score et le temps pris par chaque modèle ...

D'après la figure, on observe clairement que nos algorithmes choisies font partie des modèles les mieux classées.

Model	Accuracy	Balanced Accuracy	ROC AUC	F1 Score	\
XGBClassifier	0.89	0.89	None	0.89	
RandomForestClassifier	0.89	0.89	None	0.89	
LGBMClassifier	0.89	0.89	None	0.88	
ExtraTreesClassifier	0.88	0.88	None	0.88	
BaggingClassifier	0.88	0.88	None	0.88	
DecisionTreeClassifier	0.85	0.85	None	0.85	
SVC	0.83	0.83	None	0.83	
LabelPropagation	0.82	0.82	None	0.82	
LabelSpreading	0.82	0.82	None	0.82	
ExtraTreeClassifier	0.81	0.81	None	0.81	
QuadraticDiscriminantAnalysis	0.81	0.81	None	0.81	
LogisticRegression	0.81	0.81	None	0.81	
KNeighborsClassifier	0.81	0.81	None	0.81	
CalibratedClassifierCV	0.81	0.81	None	0.80	
LinearSVC	0.81	0.81	None	0.80	
SGDClassifier	0.80	0.80	None	0.79	
NuSVC	0.79	0.79	None	0.78	
GaussianNB	0.79	0.79	None	0.78	
AdaBoostClassifier	0.77	0.77	None	0.77	
RidgeClassifier	0.76	0.76	None	0.75	
RidgeClassifierCV	0.76	0.76	None	0.75	
LinearDiscriminantAnalysis	0.76	0.76	None	0.75	
BernoulliNB	0.72	0.72	None	0.71	
NearestCentroid	0.70	0.70	None	0.70	
Perceptron	0.64	0.64	None	0.64	
PassiveAggressiveClassifier	0.62	0.62	None	0.60	
DummyClassifier	0.33	0.33	None	0.17	

Figure 67: Table de LazyPredict

1.3. Validation croisée :

Nous avons ensuite utilisé la validation croisée pour évaluer la performance de nos modèles sur l'ensemble de validation. Voici comment cela a été réalisé, comme indiqué dans le code :

Méthode de Validation Croisée :

- ✓ **Bibliothèque** : Nous avons utilisé la fonction `cross_val_score` de la bibliothèque scikit-learn pour effectuer la validation croisée.

Classificateurs utilisés :

- ✓ Nous avons défini trois classificateurs : Random Forest, Logistic Regression, et XGBoost.

Scores et Moyenne :

- ✓ Pour chaque classificateur, nous avons appliqué la validation croisée avec 5 plis (5-fold cross-validation), obtenant ainsi une liste de scores d'exactitude pour chaque pli.
- ✓ Nous avons calculé la moyenne de ces scores pour évaluer la performance globale de chaque modèle sur l'ensemble de validation.

Les résultats ont été stockés dans un DataFrame pour une analyse plus approfondie.

Résultats de la Validation Croisée :

Nous avons évalué la performance de nos modèles à l'aide de la validation croisée sur l'ensemble de validation. Voici un récapitulatif des résultats obtenus :

1. Random Forest :

- **Scores individuels** : [[0.8166, 0.8801, 0.8677, ...]]
- **Moyenne des scores** : 0.8509
- **Observations** : Le Random Forest a démontré une performance globale élevée, avec des scores cohérents à travers les plis de la validation croisée.

2. Logistic Regression :

- **Scores individuels** : [[0.7901, 0.8166, 0.8023, ...]]
- **Moyenne des scores** : 0.7998
- **Observations** : La Régression Logistique a montré des performances solides, bien que légèrement inférieures au Random Forest.

3. XGBoost :

- **Scores individuels** : [[0.8342, 0.8871, 0.8675, ...]]
- **Moyenne des scores** : 0.8609
- **Observations** : XGBoost s'est avéré être le modèle le plus performant, avec la plus haute moyenne des scores de validation croisée.

Discussion des Résultats :

En analysant ces résultats, le modèle XGBoost émerge comme le modèle le plus performant, affichant la moyenne des scores la plus élevée lors de la validation croisée sur l'ensemble de validation. Cela suggère que XGBoost pourrait être le choix privilégié en termes de précision.

2. Test de prédiction

Après avoir évalué plusieurs modèles, la configuration des performances a déterminé que XGBoost est le plus performant pour anticiper si un client va rester ou résilier son contrat. Cette

fonctionnalité du test de prédiction interactive permet à l'utilisateur d'entrer manuellement les détails d'un client. Ensuite, en se basant sur ces informations, le modèle XGBoost préalablement formé réalise une prédiction. Les résultats de cette prédiction sont présentés, donnant une estimation comme "Rejoint", "Rester" ou "Résilié" en fonction des données fournies.

```
Entrez la valeur pour 'Married' (0 pour Non, 1 pour Oui): 1
Entrez la valeur pour 'Phone Service' (0 pour Non, 1 pour Oui): 1
Entrez la valeur pour 'Multiple Lines' (0 pour Non, 1 pour Oui): 1
Entrez la valeur pour 'Internet Service' (0 pour Non, 1 pour Oui): 1
Entrez la valeur pour 'Online Security' (0 pour Non, 1 pour Oui): 1
Entrez la valeur pour 'Online Backup' (0 pour Non, 1 pour Oui): 1
Entrez la valeur pour 'Device Protection Plan' (0 pour Non, 1 pour Oui): 1
Entrez la valeur pour 'Premium Tech Support' (0 pour Non, 1 pour Oui): 1
Entrez la valeur pour 'Streaming TV' (0 pour Non, 1 pour Oui): 1
Entrez la valeur pour 'Streaming Movies' (0 pour Non, 1 pour Oui): 1
Entrez la valeur pour 'Streaming Music' (0 pour Non, 1 pour Oui): 1
Entrez la valeur pour 'Unlimited Data' (0 pour Non, 1 pour Oui): 1
Entrez la valeur pour 'Paperless Billing' (0 pour Non, 1 pour Oui): 1
Entrez la valeur pour 'Total Charges': 11111
Entrez la valeur pour 'Number of Dependents': 11111
Entrez la valeur pour 'Contract': 11111
Entrez la valeur pour 'Age': 11111
Entrez la valeur pour 'Internet Type': 1111111
Entrez la valeur pour 'Payment Method': 12332232
Entrez la valeur pour 'Tenure in Months': 12211221
Entrez la valeur pour 'Monthly Charge': 21222121
/n
Predictions: [2]
Result: Stayed
```

Figure 68: Résultat de test

Chapitre 4:

Réalisation de l'application

Ce chapitre présente notre application, résultat concret de notre travail au sein de ce mini-projet. L'attrition client, définie comme la perte de clients ou d'abonnés, est évaluée par le pourcentage de clients perdus par rapport au total sur une période définie. Notre application, développée avec Streamlit, propose une solution proactive en utilisant un modèle prédictif basé sur un jeu de données spécifique.

L'objectif de notre application est d'identifier les clients susceptibles de se désabonner, permettant ainsi aux entreprises de prendre des mesures préventives pour les retenir. Il s'agit d'une démonstration concrète de notre travail sur le mini-projet dont le sujet était axé sur la prédiction de l'attrition des clients dans le secteur des télécommunications. Pour atteindre cet objectif, nous avons exploité les fonctionnalités de Streamlit, une bibliothèque de création d'applications web conviviale en Python.

Exploration de l'Application

Dans cette section, nous allons explorer les différentes fonctionnalités de notre application.

Le menu de notre application offre plusieurs options pour une exploration complète de l'analyse de l'attrition de la clientèle. Il se compose des sections suivantes :

1. Accueil :

C'est une introduction à l'application, accueillant l'utilisateur avec une présentation du projet et une illustration visuelle de l'objectif.



Figure 69 : Interface Accueil

2. Dataset :

Cette fonctionnalité affiche les cinq premières lignes du jeu de données, avec des informations supplémentaires telles que la forme du dataset, les détails des colonnes, et une description statistique.

	Customer ID	Gender	Age	Married	Number of Dependents	City	Zip Code	Latitude	Longitude
0	0002-ORFBO	Female	37	Yes		0 Frazier Park	93,225	34.8277	-118.9991
1	0003-MKNFE	Male	46	No		0 Glendale	91,206	34.1625	-118.2039
2	0004-TLHLJ	Male	50	No		0 Costa Mesa	92,627	33.6457	-117.9226
3	0011-IGKFF	Male	78	Yes		0 Martinez	94,553	38.0145	-122.1154
4	0013-EXCHZ	Female	75	Yes		0 Camarillo	93,010	34.2278	-119.0799

Figure 70 : Interface Dataset

3. Visualisation des données :

C'est une exploration graphique des variables numériques et catégorielles, ainsi que la visualisation des valeurs manquantes et aberrantes.

✓ **Distribution des Variables Numériques :**

Exploration graphique de la distribution des variables numériques.

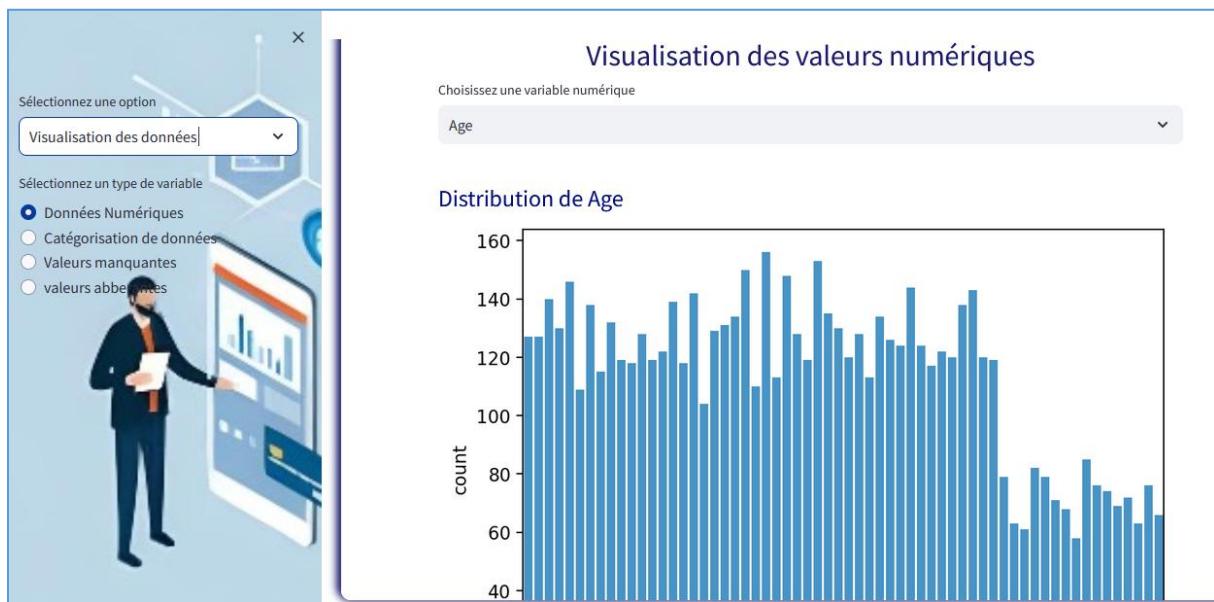


Figure 71 : Interface variables numériques

✓ **Distribution des Variables Catégorielles :**

Exploration graphique de la distribution des variables catégorielles.



Figure 69 : Interface variables catégorielles

✓ **Visualisation des Valeurs Manquantes :**

Graphique montrant la présence de valeurs manquantes dans le dataset.

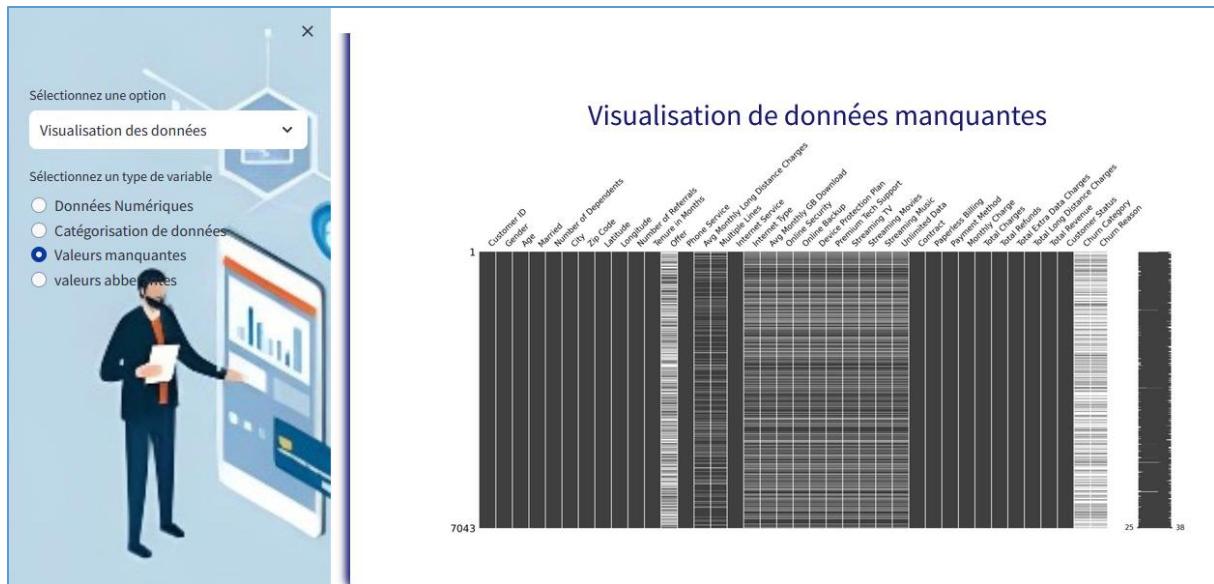


Figure 70 : Interface valeurs manquantes

✓ *Visualisation des Valeurs Aberrantes :*

Exploration graphique de la présence de valeurs aberrantes dans le dataset.

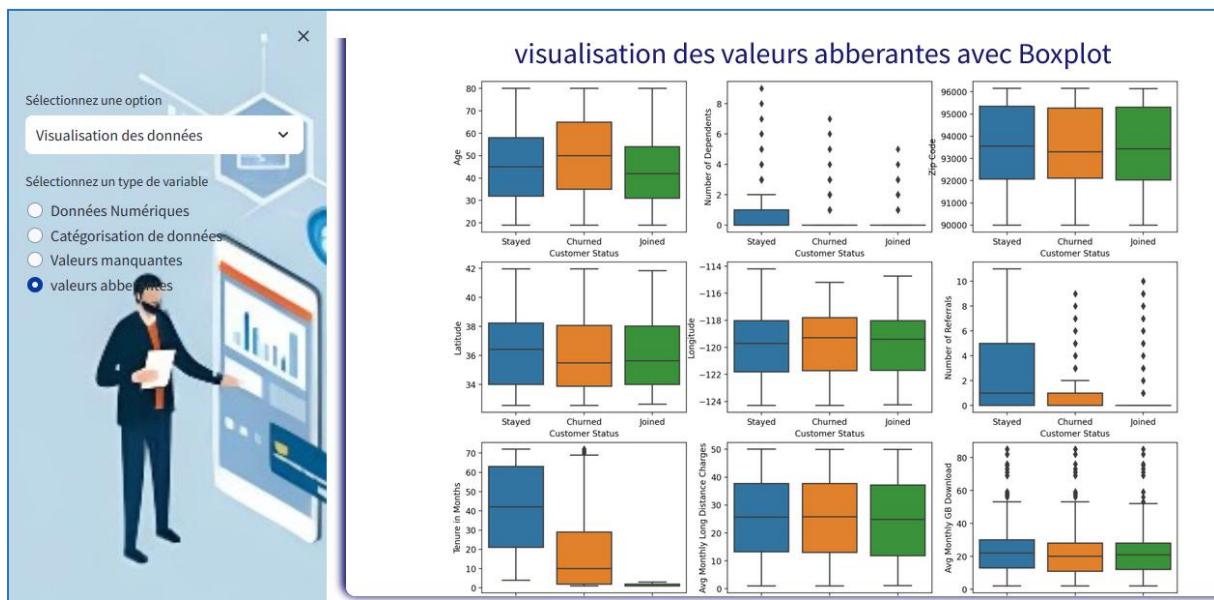


Figure 71 : Interface valeurs aberrantes

4. Manipulation de Données :

La section offre une gamme complète de fonctionnalités visant à préparer le jeu de données pour l'analyse et la modélisation. Voici une ventilation détaillée des composants de cette section :

✓ ***Supprimez les Colonnes Inutiles :***

Cette fonctionnalité permet à l'utilisateur de visualiser dataset après suppression des colonnes jugées non pertinentes pour l'analyse.

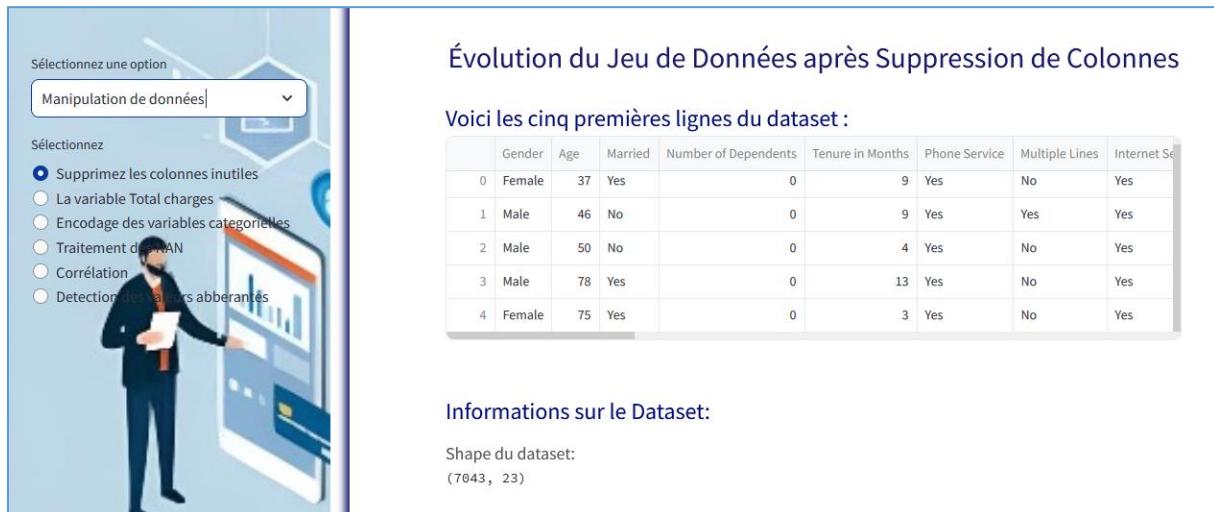


Figure 72 : Interface Suppression des Colonnes Inutiles

✓ ***La Variable "Total Charges"***

Cette section vise à explorer cette variable sous deux perspectives : avant et après transformation.

- Distribution de Total Charges avant Transformation :

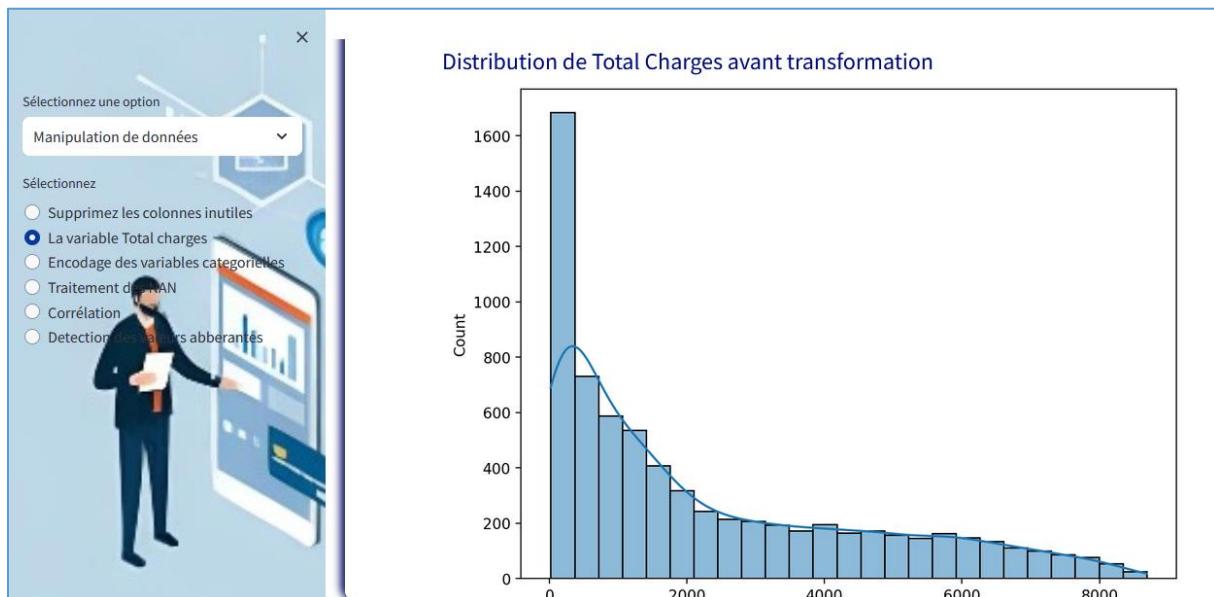


Figure 73 : Interface Total charges avant transformation

- Distribution de Total Charges après Transformation :

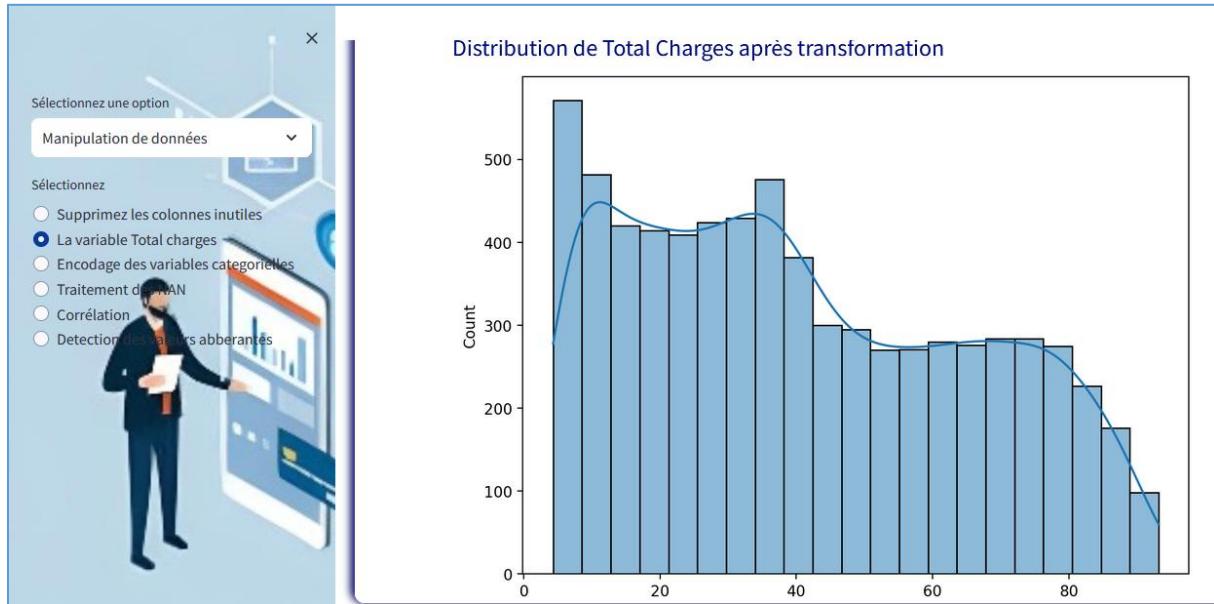


Figure 74 : Interface Total charges après transformation

✓ **Encodage des Variables Catégorielles :**

Pour garantir la compatibilité avec les algorithmes de Machine Learning, les variables catégorielles sont encodées de manière appropriée.



Figure 75 : Interface Encodage des Variables Catégorielles

✓ **Corrélation :**

Permet à l'utilisateur d'explorer visuellement les relations de corrélation entre différentes variables.

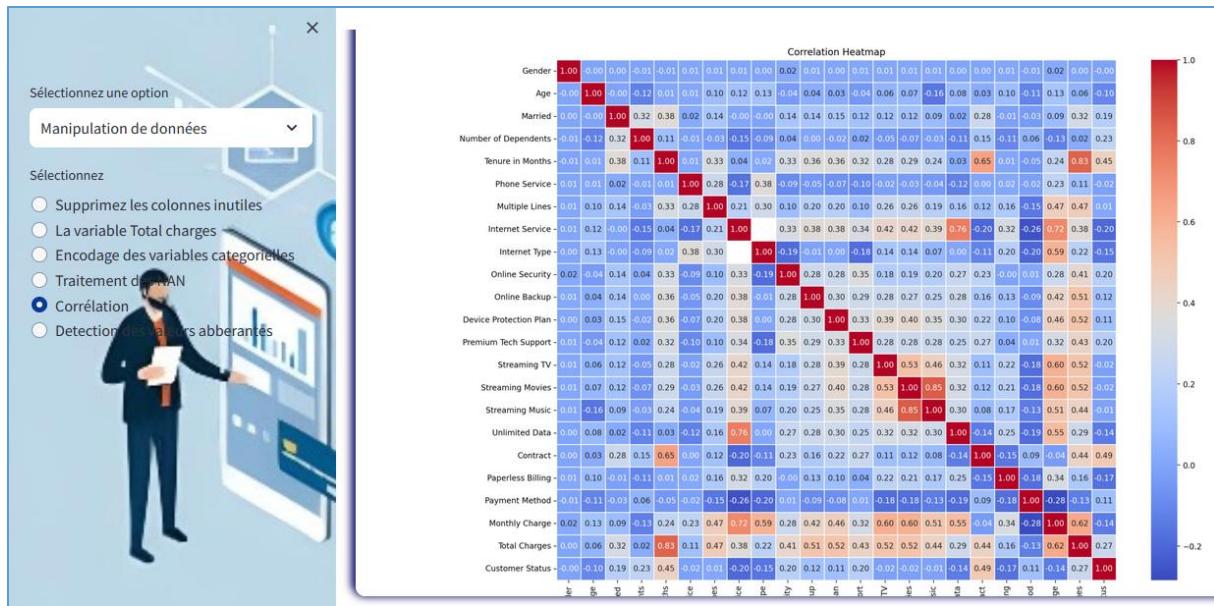


Figure 76 : Interface Correlation

✓ **Détection de Valeurs Aberrantes :**

Cette fonctionnalité utilise la visualisation avant et après traitement des valeurs aberrantes dans le jeu de données.

- Avant Traitement :

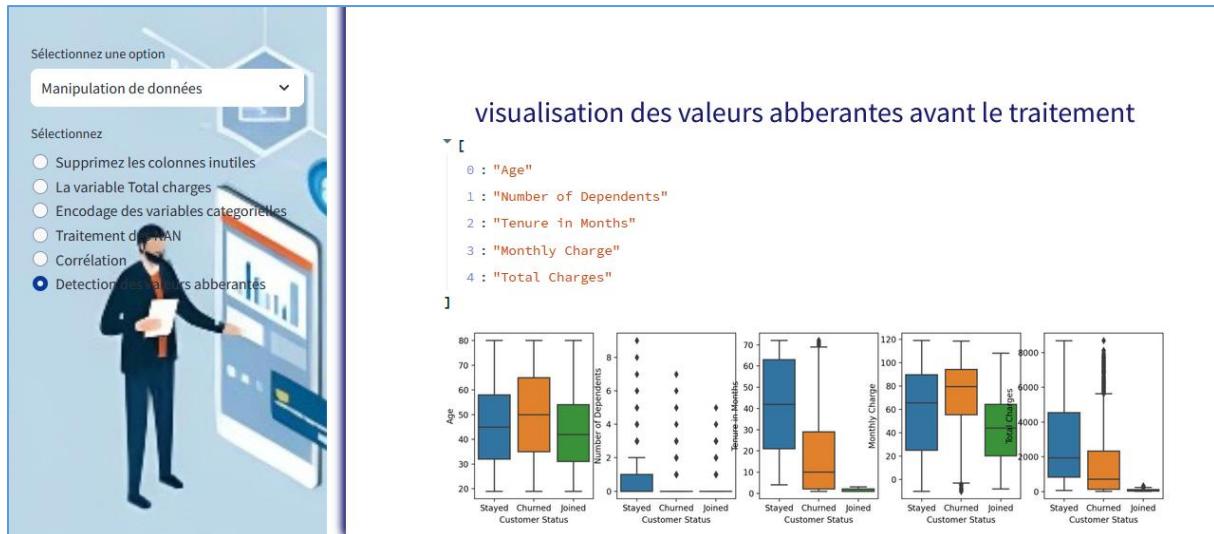


Figure 77 : Interface Détection de Valeurs Aberrantes Avant Traitement

- Après Traitement :

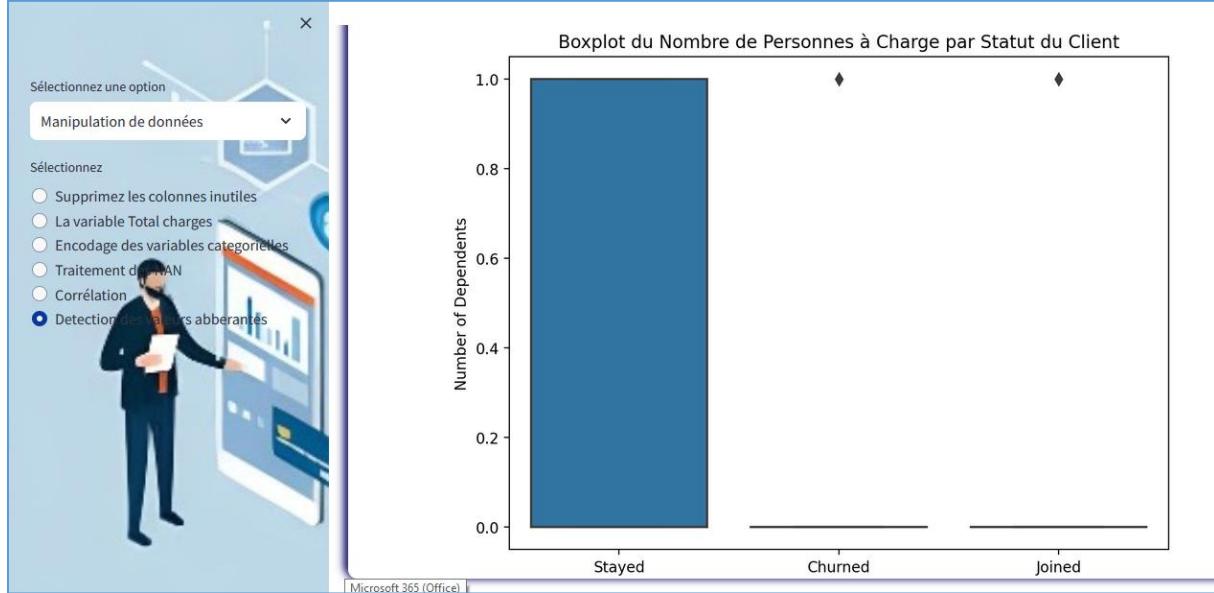


Figure 78 : Interface Détection de Valeurs Aberrantes Après Traitement

5. Analyse de Répartition :

La section est conçue pour traiter le déséquilibre des données, en particulier avant et après l'oversampling. Elle offre les options suivantes :

✓ *Déséquilibre des Données (Avant) :*

Cette fonctionnalité présente une visualisation des classes du jeu de données avant l'application de toute technique d'oversampling.

La figure ci-dessous représente cette fonctionnalité :

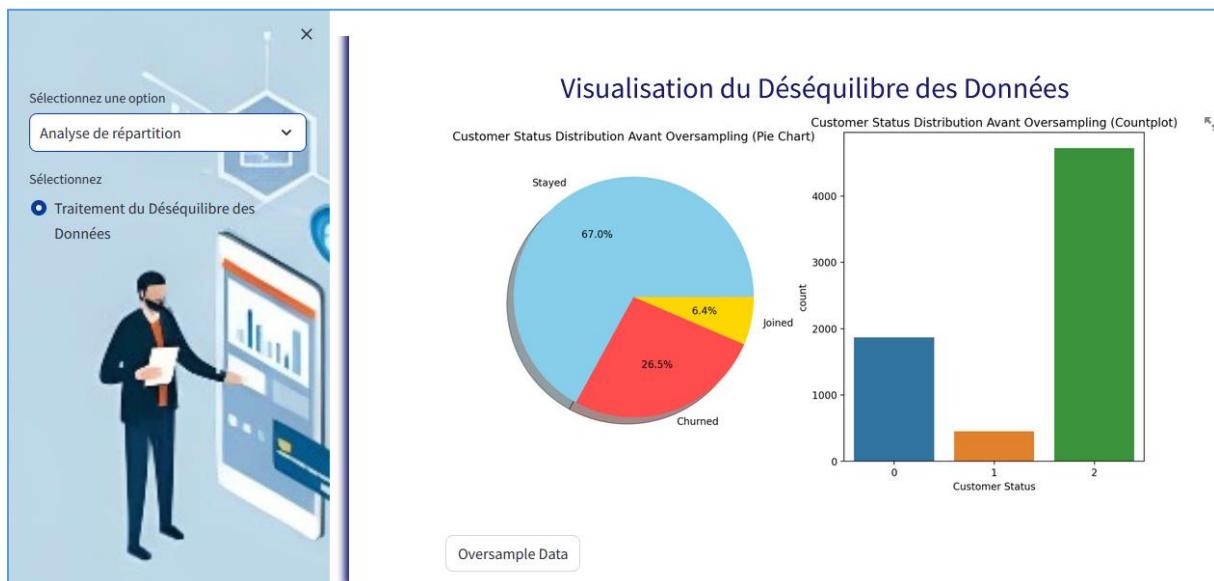


Figure 79 : Interface Déséquilibre des Données

✓ **Équilibrage des Données (Après) :**

L'oversampling est appliqué pour augmenter la représentation des classes minoritaires.

La figure ci-dessous montre lorsqu'on clique sur le button “**Oversampling data**” l'application de cette méthode:

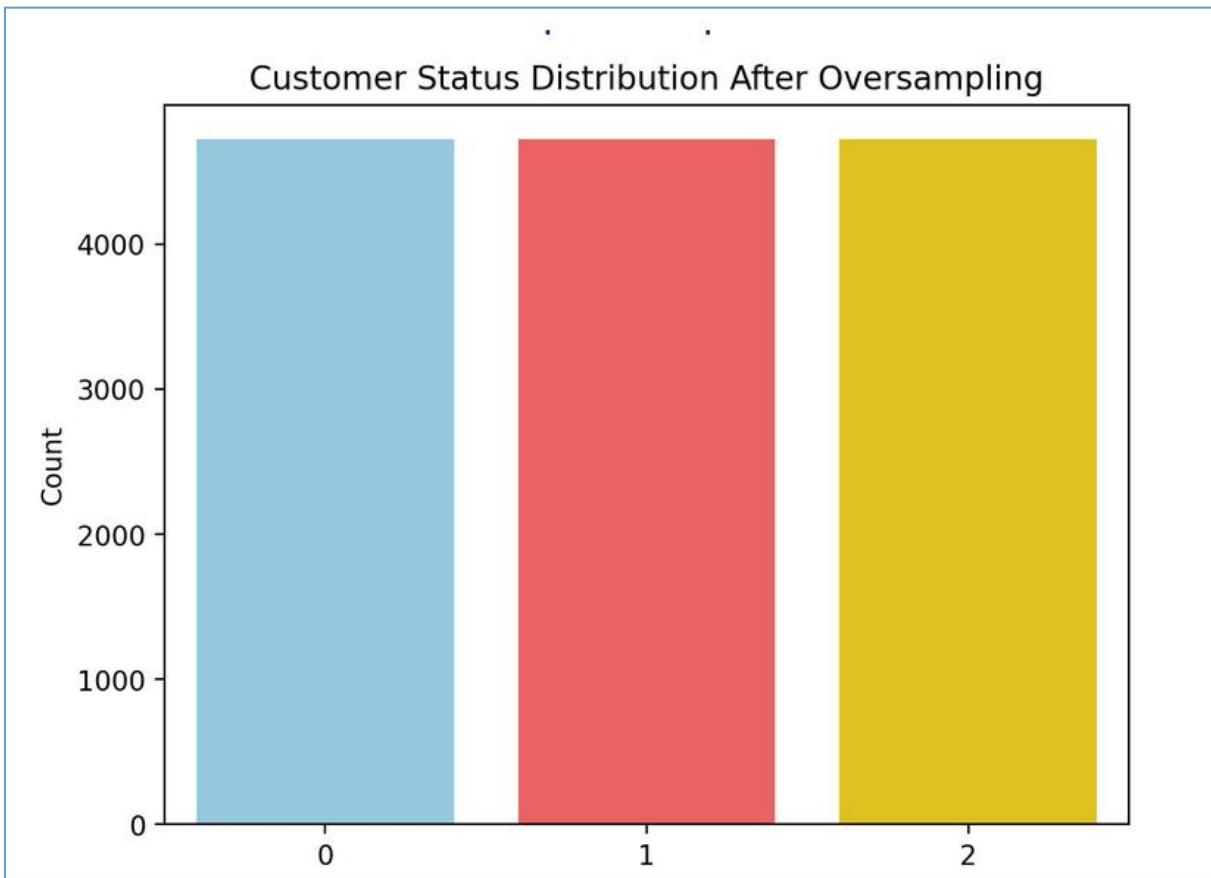


Figure 80 : Interface Équilibrage des Données

6. Prédiction :

La section "Prédiction" intègre une interface conviviale pour permettre à l'utilisateur de saisir les caractéristiques d'un client et d'obtenir la prédiction de son statut.

L'utilisateur remplit un formulaire avec les informations du client.

La figure ci-dessous illustre ce formulaire :



Sélectionnez une option

Prédiction

Test de prédition

Married	Phone Service	Multiple Lines
<input checked="" type="radio"/> Yes	<input checked="" type="radio"/> Yes	<input checked="" type="radio"/> Yes
<input type="radio"/> No	<input type="radio"/> No	<input type="radio"/> No
Internet Service	Online Security	Online Backup
<input checked="" type="radio"/> yes	<input checked="" type="radio"/> Yes	<input checked="" type="radio"/> Yes
<input type="radio"/> No	<input type="radio"/> No	<input type="radio"/> No
Device Protection Plan	Premium Tech Support	Unlimited Data
<input checked="" type="radio"/> Yes	<input checked="" type="radio"/> Yes	<input checked="" type="radio"/> Yes
<input type="radio"/> No	<input type="radio"/> No	<input type="radio"/> No
Streaming TV	Streaming Movies	Streaming Music
<input checked="" type="radio"/> Yes	<input checked="" type="radio"/> Yes	<input checked="" type="radio"/> Yes
<input type="radio"/> No	<input type="radio"/> No	<input type="radio"/> No

Internet Type	Contrat	Paperless Billing
<input checked="" type="radio"/> DSL	<input checked="" type="radio"/> Month-to-Month	<input checked="" type="radio"/> Yes
<input type="radio"/> Fiber Optic	<input type="radio"/> One Year	<input type="radio"/> No
<input type="radio"/> No	<input type="radio"/> Two Year	

Payment Method		
<input checked="" type="radio"/> Electronic Check		
<input type="radio"/> Mailed Check		
<input type="radio"/> Bank Transfer (Automatic)		
<input type="radio"/> Credit Card (Automatic)		

Âge

19

46

80

Number of Dependents

0

0

9

Number of Dependents	Tenure in Months	Monthly Charge
0	32	63.60
0	1	-10.00
9	72	118.75

Total Charges		
2280.38	2280.38	2280.38
18.80	18.80	18.80
8684.80	8684.80	8684.80

Figure 81 : Interface de formulaire de prediction

72

Les résultats possibles si on clique sur prédire sont les suivants :

✓ **Résultat de Prédiction (Client Joined) :**

Si le modèle prédit que le client rejoint, une vue du résultat est présentée comme suit :

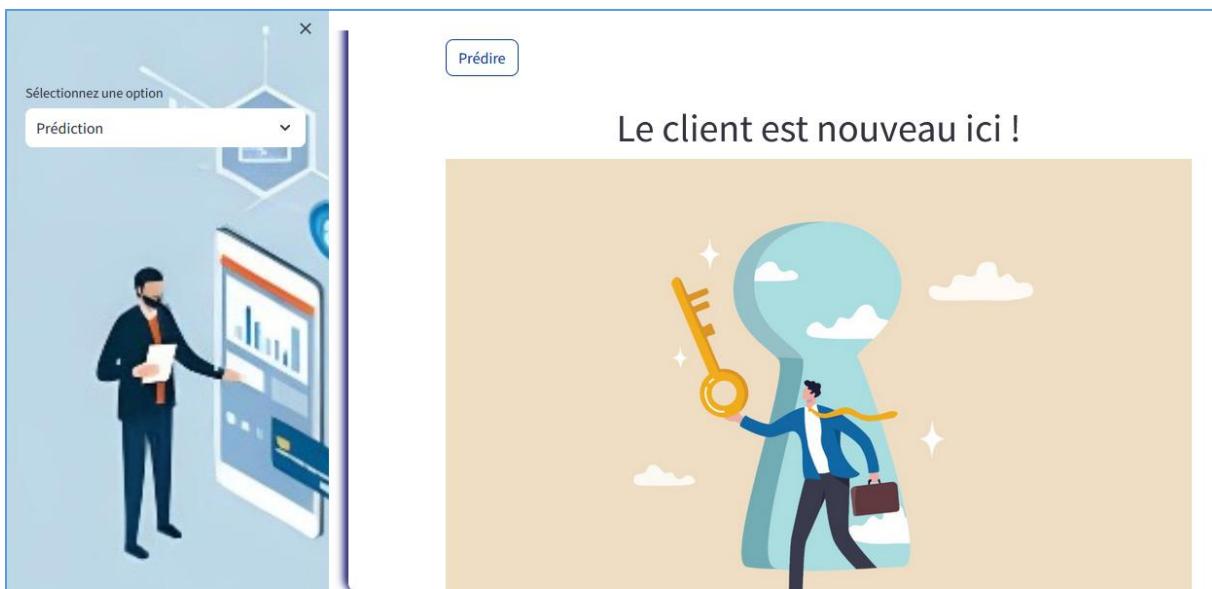


Figure 82 : Interface de résultat (client joined)

✓ **Résultat de Prédiction (Client Stayed) :**

De même, si le modèle prédit que le client reste, l'utilisateur voit le résultat de cette prédiction, comme montré ci-dessous :



Figure 83 : Interface de résultat (client stayed)

✓ **Résultat de Prédiction (Client Churned) :**

Lorsque le modèle prédit que le client va se désabonner, l'interface affiche une vue spécifique du résultat, comme illustré ci-dessous :



Figure 84 : Interface de résultat (client churned)

En résumé, le chapitre de réalisation de l'application a été une étape cruciale de notre projet, détaillant le développement, les fonctionnalités, et la concrétisation de notre solution de prédiction de l'attrition client. À travers des sections telles que l'accueil, la visualisation des données, la manipulation de données, l'analyse de répartition, et la prédiction, nous avons démontré la puissance de notre application construite avec Streamlit. Chaque élément a été soigneusement conçu pour offrir une expérience utilisateur complète, de l'exploration des données à la prise de décision basée sur les résultats de prédiction. Cette application représente le résultat tangible de notre travail acharné et de notre compréhension approfondie des concepts de machine learning appliqués à la problématique de l'attrition client dans le secteur des télécommunications.

Conclusion

Ce projet de clôture du module d'apprentissage automatique I représente une étape significative dans notre parcours académique en intelligence artificielle à l'École Supérieure de Technologie de Meknès. Notre engagement dans la prévision de l'attrition de la clientèle à travers un modèle de prédiction du désabonnement des clients a été guidé par la nécessité de relever les défis contemporains auxquels sont confrontées les entreprises, en particulier dans le secteur des télécommunications.

Au fil de ce travail, nous avons traversé les différentes phases du processus, de la compréhension de la problématique à la mise en œuvre concrète de solutions. L'utilisation de techniques avancées de Machine Learning pour anticiper le churn des clients a été une expérience enrichissante, nous offrant une perspective pratique sur l'application des connaissances acquises au cours de notre formation.

Les quatre chapitres principaux ont permis de structurer notre démarche, allant de la description générale du projet à la mise en place concrète du modèle, pour finalement aboutir à l'analyse approfondie des résultats obtenus et la réalisation de l'application. Ces résultats ont souligné l'efficacité du modèle dans l'identification des clients à risque, ouvrant ainsi la voie à des actions proactives visant à les retenir.

En conclusion, ce projet a été une opportunité d'appliquer nos connaissances théoriques dans un contexte pratique et concret. Il a renforcé notre compréhension des enjeux liés à la prédiction du churn des clients et a contribué à notre développement professionnel dans le domaine de l'intelligence artificielle. Ce travail représente non seulement une réalisation académique, mais aussi un jalon important dans notre cheminement vers une compréhension plus approfondie et une application plus efficace des techniques de Machine Learning dans des contextes réels.

