



MINISTÈRE DE L'ENSEIGNEMENT SUPÉRIEUR  
DE LA RECHERCHE SCIENTIFIQUE ET DE L'INNOVATION  
**UNIVERSITÉ SULTAN MOULAY SLIMANE**  
**ECOLE NATIONALE DES SCIENCES APPLIQUÉS**  
**DE KHOURIBGA**



# Mémoire de fin d'études

pour l'obtention du

**Diplôme d'Ingénieur d'État**

**Filière : Informatique et Ingénierie des Données**



**Présenté par :**

**Adham AROUBITE**

## Développement d'un système de génération automatique de données d'entraînement pour les modèles de prédiction de churn

**Sous la supervision :**

**M. Hamza KHALFI**

**M. Mohamed MEJDOUBI**

**Membres du jury :**

<b>M. Abdelghani GHAZDALI</b>	<b>Président</b>
<b>M. Mohammed JERRADI</b>	<b>Examinateur</b>
<b>Mme Hind HAFSI</b>	<b>Examinatrice</b>
<b>M. Hamza KHALFI</b>	<b>Encadrant Interne</b>
<b>M. Mohamed MEJDOUBI</b>	<b>Encadrant Externe</b>

**Année Académique : 2023**



# Sommaire

Dédicace	ii
Remerciements	iii
Résumé	iv
Abstract	v
Table des matières	vi
Table des figures	viii
Liste des tableaux	x
Liste des sigles et acronymes	xi
Introduction	1
1 Présentation de l'entreprise d'accueil	4
2 Cadre du stage	10
3 Paramètres métier	16
4 Implémentation de la solution	27
5 EDA et Entraînement du modèle de prédiction de churn	42
Conclusion	57
Bibliographie	58

# Dédicace

“

*À ma chère mère, Ce travail est une preuve de l'amour sans limite que tu m'as donné. Chaque page reflète ta résistance et les sacrifices que tu as faits pour moi. Ta persévérance a guidé mon chemin.*

*À mon cher père, Cette réalisation est pour toi qui as toujours cru en mes capacités, même avant que je les réalise moi-même. Ton soutien et ta foi en moi ont façonné le courage et la détermination qui m'ont amené ici.*

*À mon frère, Tu es mon modèle, la référence à laquelle je me réfère constamment. Tes conseils ont été une source précieuse de sagesse et d'orientation. Tu es mon bras droit, un soutien indispensable à mes côtés. Cette réussite est autant la tienne que la mienne.*

*À ma soeur, Ta confiance en moi a toujours guidé mon chemin. Tu n'as jamais laissé la place au doute, toujours à mes côtés pour m'encourager, me soutenir et croire en moi. Je suis fier de toi et de la personne que tu es devenue. Cette reconnaissance est également un hommage à ta force et à ta confiance en moi.*

*À mes chers professeurs, Votre enseignement de qualité a été le fondement de mes connaissances. Vous avez alimenté mon désir d'apprendre et m'avez inspiré à chaque étape de ma carrière universitaire. Ma réussite est aussi le résultat de votre dévouement.*

*Avec tout mon respect et ma gratitude.*

”

**- Adham Aroubite**

# Remerciements

Je tiens à remercier chaleureusement Monsieur Hamza Khalfi. Son encadrement, ses remarques judicieuses, son encouragement constant, ses conseils et sa motivation ont été d'une importance capitale tout au long de ce travail. Sa contribution a été déterminante pour la réalisation de ce projet et il m'aurait été impossible d'atteindre ce niveau sans lui.

Une gratitude particulière est due à mon encadrant chez inwi, Monsieur Mohamed Mejdoubi. Son aide, son soutien et les opportunités qu'il m'a offertes ont été essentiels pour moi. J'ai beaucoup appris de lui et il reste une constante source d'inspiration.

Je tiens à exprimer ma reconnaissance à Mohamed Amrouane et Hajar Zekroum, les deux data scientists de mon équipe. Leur accueil chaleureux, leur échange d'informations précieuses et leur soutien tout au long de ce parcours ont été inestimables.

Ma gratitude s'étend également à l'ensemble du département Performance et Connaissance Client de inwi, en particulier l'équipe Data Science, pour leur collaboration, leur soutien et leur esprit d'équipe qui ont grandement contribué à l'élaboration de ce travail.

Je remercie également l'équipe professorale de l'ENSA de Khouribga pour leur formation rigoureuse et de qualité qui m'a préparé pour ce défi. Leur expertise, leur dévouement et leur passion pour l'enseignement ont été une source d'inspiration tout au long de mon parcours académique.

Enfin, un grand merci aux membres du jury qui ont accepté de consacrer leur temps précieux à l'évaluation de ce travail

Ce travail est le fruit de la collaboration et de l'effort de chacun d'entre vous. Je souhaite exprimer ma gratitude la plus profonde pour vos contributions qui ont rendu possible l'achèvement de ce projet.

# Résumé

Ce rapport présente une solution innovante à une problématique majeure dans le domaine du machine learning et du big data. Sa mise en œuvre a eu un impact significatif sur l'équipe data science de INWI.

Traditionnellement, les données utilisées pour l'entraînement des modèles sont souvent stockées sous forme de logs ou d'événements, rendant leur utilisation directe pour l'entraînement des modèles difficile. De plus, l'écriture manuelle des agrégations pour transformer ces logs en données formatées pour l'entraînement est un processus coûteux, laborieux, et souvent limité par la complexité et la diversité des données.

Pour répondre à ces défis, nous avons développé un système automatisé de génération de données d'entraînement, y compris les labels. Ce système, à partir d'un simple fichier de configuration, transforme les données de logs en un format prêt pour l'entraînement des modèles de machine learning, en générant des agrégations de manière automatisée. Cette automatisation réduit considérablement le temps, les coûts et les efforts liés à la préparation des données, libérant ainsi du temps pour se concentrer sur l'exploration et la compréhension des modèles, la découverte de nouvelles tendances cachées et l'expérimentation avec différents paramètres métiers sans la nécessité de tout réécrire.

L'efficacité de notre système a été démontrée dans un cas d'application de prédiction de churn. Les modèles entraînés avec les données générées par notre système ont démontré une performance significativement améliorée par rapport à ceux entraînés avec des données préparées manuellement. De plus, le système sert d'excellent outil de prototypage, permettant d'évaluer rapidement la faisabilité et le potentiel d'un projet avant de s'y engager pleinement.

En conclusion, notre système offre une solution prometteuse pour optimiser la préparation des données d'entraînement, améliorer la performance des modèles de machine learning et faciliter le prototypage rapide.

---

**Mots clés :** Apprentissage automatique, Prédiction de churn, Automatisation des agrégations, Données d'entraînement

---

# Abstract

This report presents an innovative solution to a major challenge in the field of machine learning and big data. Its implementation had a significant impact on the data science team of INWI.

Traditionally, the data used for training models is often stored as logs or events, making their direct use for training difficult. Additionally, manually writing aggregations to transform these logs into training formatted data is a costly, labor intensive process, often limited by the complexity and diversity of the data.

In response to these challenges, we have developed an automated training data generation system, including labels. This system, based on a simple configuration file, transforms log data into a format ready for training machine learning models, generating aggregations in an automated manner. This automation significantly reduces the time, cost, and effort associated with data preparation, thereby freeing up time to focus on exploring and understanding the models, discovering new hidden data patterns, and experimenting with different business parameters without the need to rewrite everything.

The efficacy of our system was demonstrated in a churn prediction use case. Models trained with data generated by our system demonstrated significantly improved performance compared to those trained with manually prepared data. Furthermore, the system serves as an excellent prototyping tool, allowing for a quick assessment of a project's feasibility and potential before fully committing to it.

In conclusion, our system offers a promising solution for optimizing training data preparation, improving machine learning model performance, and facilitating rapid prototyping across various application domains.

---

**Keywords :** Machine Learning, Churn Prediction, Aggregation Automation, Training data

---

# Table des matières

Dédicace	ii
Remerciements	iii
Résumé	iv
Abstract	v
Table des matières	vi
Table des figures	viii
Liste des tableaux	x
Liste des sigles et acronymes	xi
Introduction	1
<b>1 Présentation de l'entreprise d'accueil</b>	<b>4</b>
Introduction	4
1.1 Présentation de l'organisme d'accueil	4
1.2 Historique de l'entreprise	5
1.3 Infrastructure bigdata chez inwi	5
1.3.1 Présentation du département connaissance client et performances	5
1.3.2 Flux des données : d'une antenne BTS vers une valeur business	6
<b>2 Cadre du stage</b>	<b>10</b>
Introduction	10
2.1 Problématique	10
2.2 Solution	14
2.3 Planification et calendrier des activités	15
<b>3 Paramètres métier</b>	<b>16</b>
Introduction	16
3.1 churn	16
3.1.1 définition	16
3.1.2 Acquisition des nouveaux clients vs rétention des anciens clients	17
3.2 Postpayé vs prépayé	18
3.3 Cutoff_date	20
3.4 Base actif	20



3.4.1	Définition . . . . .	20
3.4.2	Data leakage . . . . .	22
3.5	Labels . . . . .	22
3.5.1	Définition . . . . .	22
3.5.2	Comment labelliser un client . . . . .	23
3.6	Historique des données . . . . .	25
<b>4</b>	<b>Implémentation de la solution</b>	<b>27</b>
	Introduction . . . . .	27
4.1	Données utilisées . . . . .	27
4.2	Outils . . . . .	27
4.3	Fichier config . . . . .	29
4.4	Génération automatique de la base actif et des labels . . . . .	29
4.4.1	Introduction . . . . .	29
4.4.2	Paramètres du fichier de configuration . . . . .	30
4.4.3	Code . . . . .	31
4.4.4	Résultats . . . . .	33
4.5	Génération des agrégations . . . . .	35
4.5.1	définition . . . . .	35
4.5.2	Concept . . . . .	35
4.5.3	Paramètres du fichier de configuration . . . . .	36
4.5.4	Explication de chaque agrégation et de sa fonction . . . . .	38
4.5.5	Résultats . . . . .	39
4.6	Interface graphique . . . . .	39
<b>5</b>	<b>EDA et Entraînement du modèle de prédiction de churn</b>	<b>42</b>
	Introduction . . . . .	42
5.1	EDA . . . . .	42
5.2	Entraînement du modèle de prédiction de churn . . . . .	48
5.2.1	Système de génération des données d'entraînement . . . . .	48
5.2.2	Prétraitement des données . . . . .	50
5.2.3	Équilibrage des classes . . . . .	50
5.2.4	Sélection des features . . . . .	52
5.2.5	Entraînement du modèle . . . . .	54
5.2.6	Métriques de performance . . . . .	54
	<b>Conclusion</b>	<b>57</b>
	<b>Bibliographie</b>	<b>58</b>

# Table des figures

1.1	logo de l'entreprise inwi . . . . .	4
1.2	Image d'une antenne BTS . . . . .	7
2.1	Diagramme de Gantt . . . . .	15
3.1	Illustration de seau qui fuit représentant le churn des clients . . . . .	17
3.2	Diagramme Cutoff date . . . . .	20
3.3	Diagramme paramètre d'activité . . . . .	21
3.4	Diagramme paramètre label . . . . .	24
3.5	Diagramme paramètre historique . . . . .	26
4.1	logo du langage de programmation Python . . . . .	28
4.2	logo de Jupyter Notebook . . . . .	28
4.3	logo de MySQL . . . . .	29
4.4	Première partie du fichier de configuration . . . . .	30
4.5	Diagramme des étapes de la génération de la base active . . . . .	32
4.6	Exemple du fichier sql pour la génération de la base active généré automatiquement par le système selon les paramètres du fichier de configuration . . . . .	32
4.7	Diagramme des étapes de la génération des labels . . . . .	33
4.8	Exemple du fichier sql pour la génération des labels généré automatiquement par le système selon les paramètres du fichier de configurations . . . . .	33
4.9	Deux tables générées automatiquement : base_active et base_active_labels . . . . .	33
4.10	Première page de la table base_active . . . . .	34
4.11	Première page de la table base_active_labels . . . . .	34
4.12	Graphe de distribution des labels généré automatiquement . . . . .	35
4.13	Exemple d'un table temporaire du fichier sql finale . . . . .	36
4.14	Deuxième partie du fichier de configuration . . . . .	37
4.15	Table train_data . . . . .	39
4.16	Partie de la table train_data . . . . .	39
4.17	Choix de la date cutoff par interface graphique . . . . .	40
4.18	Remplissage des paramètres de labellisation par interface graphique et bouton de visualisation de la distribution des classes . . . . .	40
4.19	Remplissage des paramètres additionnels par interface graphique . . . . .	41
4.20	Choix des tables et des colonnes à prendre en considération lors de la génération des aggregations par interface graphique et bouton de génération données d'entrainements . . . . .	41
5.1	Histogramme des montants de recharge (Version anonyme) . . . . .	44
5.2	Grappe du nombre de recharges par canal (Version anonyme) . . . . .	44

5.3	Graphe de du nombre de recharges par canal (canal 4, 5, 6 et 7) (Version anonyme) . .	45
5.4	Graphe de la somme des montants recharges par canal (Version anonyme) . . . . .	45
5.5	Graphe du nombre de recharges par jour (Version anonyme) . . . . .	46
5.6	Graphe du nombre de recharges par type de recharge (Version anonyme) . . . . .	46
5.7	Graphe de la somme des montants recharge par type de recharge (Version anonyme) .	47
5.8	Graphe du nombre de recharges par jour de la semaine (Version anonyme) . . . . .	47
5.9	Nombre de recharges par jour de la semaine en fonction des types de recharge (Version anonyme) . . . . .	48
5.10	Le fichier de configuration choisi pour le modèle de prédiction de churn . . . . .	49
5.11	Intervalles de temps choisis . . . . .	50
5.12	Répartition des classes . . . . .	51
5.13	dispersion des points de données sur un plan 2d avant suréchantillonnage . . . . .	51
5.14	dispersion des points de données sur un plan 2d après suréchantillonnage . . . . .	51
5.15	dispersion des points de données sur un plan 3d avant suréchantillonnage . . . . .	52
5.16	dispersion des points de données sur un plan 3d après suréchantillonnage . . . . .	52
5.17	Graphe de distribution des labels après suréchantillonnage . . . . .	52
5.18	Graphe des scores des top 50 caractéristiques . . . . .	53
5.19	Graphe de comparaison des deux courbes ROC des deux modèles . . . . .	55

# Liste des tableaux

3.1	Logs de recharges de 3 MDNs . . . . .	25
3.2	MDNs labellisés après calcul (0 : non churn, 1 : churn) . . . . .	25
5.1	Rapport de classification du 1er modèle en utilisant le système de génération des données d'entraînement . . . . .	55
5.2	Rapport de classification du 2ème modèle sans utiliser le système de génération des données d'entraînement . . . . .	55

# Liste des sigles et acronymes

MDN	Mobile Directory Number
BTS	Base Transceiver Station
JSON	JavaScript Object Notation
KNN	k-nearest neighbors
SVM	Support Vector Machines
XGBoost	eXtreme Gradient Boosting
SMOTE	Synthetic Minority Over-sampling Technique
ACP	Analyse en Composantes Principales

# Introduction Générale

La **data science** est une discipline qui vise à extraire des connaissances à partir de données, en utilisant des techniques statistiques, informatiques et mathématiques. C'est un domaine émergent qui permet aux entreprises de collecter, traiter et analyser des données massives pour améliorer leur prise de décision et leur performance.

Les **télécommunications** jouent un rôle clé dans notre société en connectant les individus et les organisations à travers le monde. À mesure que la demande en services de télécommunication augmente, les entreprises de ce secteur sont confrontées à une concurrence accrue et à une pression constante pour innover et offrir des services de meilleure qualité à des prix compétitifs. Dans ce contexte, la data science est devenue un outil essentiel pour les entreprises de télécommunications qui cherchent à améliorer leur compréhension des comportements des clients, optimiser leurs réseaux et développer des produits et services plus adaptés aux besoins des consommateurs. Ces entreprises ont été particulièrement affectées par l'adoption de la data science en raison de la quantité considérable de données qu'elles collectent auprès de leurs clients. Cependant, ces données ne sont pas exploitables sans un traitement adéquat. C'est là que la data science entre en jeu en fournissant des outils et des techniques pour analyser les données, extraire des informations utiles et les utiliser pour prendre des décisions commerciales éclairées.

L'un des défis majeurs auxquels font face les entreprises de télécommunications est le **churn**, c'est-à-dire la perte de clients au profit de concurrents. Le taux de churn a un impact significatif sur la rentabilité d'une entreprise, car il est souvent plus coûteux d'acquérir de nouveaux clients que de retenir les clients existants. Cette problématique est d'autant plus cruciale dans le secteur des télécommunications, où la concurrence est féroce et les clients peuvent facilement passer d'un opérateur à un autre en quête de meilleures offres et de services plus performants. La data science permet aux entreprises de télécommunications d'identifier les facteurs qui influencent le churn et de développer des stratégies pour minimiser ce phénomène. En utilisant des techniques d'apprentissage automatique et de modélisation prédictive, les data scientists peuvent analyser les données historiques des clients pour détecter les tendances et les schémas qui mènent au churn. Parmi ces facteurs, on retrouve des éléments tels que la satisfaction des clients, la qualité du service, les offres promotionnelles des concurrents, les changements dans les habitudes de consommation et les évolutions technologiques. Les entreprises de télécommunications disposent de vastes ensembles de données qui peuvent être utilisés pour développer des modèles prédictifs du churn tels que les appels, les SMS, l'utilisation d'Internet, les plaintes des clients, les données de facturation, les interactions sur les réseaux sociaux, les recharges et les offres promotionnelles. En analysant ces données, les data scientists peuvent détecter les signaux précurseurs du churn et identifier les clients à risque. Une fois que les clients à

risque sont identifiés, les entreprises de télécommunication peuvent prendre des mesures proactives pour les retenir. Ces mesures peuvent inclure des offres personnalisées, des améliorations de la qualité de service, une communication ciblée ou la mise en place de programmes de fidélité. L'objectif est d'adresser les causes profondes du churn et de créer une expérience client plus engageante et satisfaisante, qui incite les clients à rester fidèles à l'entreprise. De plus, la data science permet également aux entreprises de télécommunications de segmenter leur clientèle en fonction des profils de risque de churn. Cette segmentation peut être utilisée pour ajuster les stratégies de marketing et de service à la clientèle, en ciblant les offres et les messages en fonction des besoins et des préférences spécifiques de chaque segment de clients.

Le **feature engineering**, qui est le processus de transformation et de sélection des variables explicatives utilisées pour construire un modèle prédictif, est une étape cruciale dans le développement de modèles de prédiction du churn. Les entreprises de télécommunications génèrent et collectent d'énormes quantités de données, allant des informations démographiques et comportementales des clients aux données sur l'utilisation des services et la performance du réseau. Le défi pour les data scientists est de sélectionner les features pertinentes et de les transformer de manière à maximiser la performance du modèle prédictif. Le feature engineering peut inclure des techniques telles que la sélection de variables, la normalisation, la catégorisation, la création de nouvelles variables à partir de combinaisons d'autres variables et l'analyse de l'importance relative de chaque feature pour la prédiction. L'objectif est d'extraire le maximum d'informations à partir des données disponibles et de les structurer de manière à faciliter l'apprentissage automatique et la modélisation prédictive. Le choix des features et leur préparation ont un impact direct sur la qualité du modèle. Cependant, il est important de noter que la plupart des projets de data science peuvent prendre beaucoup de temps sans garantie de résultats concluants. Cela est dû en partie à la nécessité d'une expertise approfondie dans le domaine concerné pour comprendre et interpréter les données de manière significative. De plus, une grande partie du temps est souvent consacrée à la génération manuelle de données à partir d'agrégations d'autres données, ce qui peut entraîner un gaspillage considérable de ressources et d'efforts.

L'importance d'un système de génération automatique de données d'entraînement pour les modèles ne saurait être sous-estimée dans ce contexte. Un tel système permettrait aux data scientists de se concentrer davantage sur la compréhension et l'interprétation des résultats du modèle, plutôt que de passer un temps précieux à générer et préparer manuellement les données d'entraînement. Cela pourrait non seulement accélérer le processus de développement du modèle, mais aussi améliorer la qualité et la fiabilité des prédictions en permettant aux data scientists de consacrer davantage de temps à l'analyse et à l'optimisation du modèle. En outre, la mise en place d'un système de génération automatique de données d'entraînement pourrait également faciliter la collaboration entre les data scientists et les experts du domaine, en leur permettant de partager leurs connaissances et leur expertise de manière plus efficace. Cela pourrait conduire à une meilleure compréhension des facteurs qui influencent le churn et à l'identification de nouvelles variables pertinentes à inclure dans le modèle prédictif. Un autre avantage d'un tel système serait la possibilité d'explorer rapidement différentes combinaisons de features et de modèles pour déterminer la meilleure approche pour résoudre un problème donné. Cela pourrait permettre aux entreprises de télécommunications d'optimiser leurs modèles de prédiction du churn plus rapidement et plus efficacement, et de prendre des décisions plus éclairées sur les stratégies à mettre en place pour retenir leurs clients et améliorer leur rentabilité.

En conclusion, l'application de la data science dans le secteur des télécommunications offre de

nombreuses opportunités pour améliorer la compréhension des comportements des clients, optimiser les réseaux, développer des produits et services plus adaptés aux besoins des consommateurs, et aborder des défis majeurs tels que le churn. Dans ce contexte, le feature engineering, en particulier la génération des données, est un élément clé pour développer des modèles prédictifs performants, en sélectionnant et en transformant les variables explicatives pertinentes.

Le projet de fin d'étude et le rapport qui en découle se concentreront sur le développement d'un système automatique de génération de données d'entraînement, qui permettra d'expérimenter avec différents paramètres métiers et de tester diverses combinaisons de features pour optimiser les modèles de prédiction du churn. L'objectif ultime est de fournir à l'entreprise de télécommunications **INWI** un outil efficace et automatisé pour accélérer le processus de développement des modèles prédictifs, améliorer la qualité et la fiabilité des prédictions, et faciliter la prise de décisions éclairées en matière de stratégies de rétention des clients et d'amélioration de la rentabilité afin de renforcer sa compétitivité sur le marché.



# Présentation de l'entreprise d'accueil

## Introduction

Dans ce chapitre, nous allons nous pencher sur l'organisme qui a accueilli ce projet, à savoir INWI, un des leaders du secteur des télécommunications au Maroc. Nous allons donner un aperçu des différentes équipes qui travaillent sur les données chez INWI, en mettant en évidence leur structure, leurs rôles et comment elles collaborent pour atteindre les objectifs de l'entreprise.

### 1.1 Présentation de l'organisme d'accueil

Inwi, anciennement **Wana**, est un acteur majeur du secteur des télécommunications au Maroc. Inwi a su s'imposer comme le troisième opérateur téléphonique du pays, après Maroc Telecom et Orange Maroc (anciennement Meditel). Avec un réseau couvrant plus de **92%** du territoire national et plus de **50 000 agences**, Inwi offre ses services aussi bien aux particuliers qu'aux entreprises.

Devenue filiale d'**Al Mada** (ex-SNI) et du consortium koweïtien **Al Ajial-Zaïn**, Wana Corporate opère sur les segments de la **téléphonie fixe et mobile**, de l'**internet** ainsi que sur celui du **cloud** à travers sa marque « Inwi ». La société répond aux besoins variés de sa clientèle grâce à une gamme complète de produits et services de communication.

Inwi se distingue également par son slogan "**M3akoum koul youm**" (avec vous, tous les jours), reflétant son engagement à accompagner ses clients au quotidien et à leur fournir des solutions innovantes adaptées à leurs besoins. Aujourd'hui, l'opérateur emploie près de **1 200** collaborateurs et compte plus de **12 millions** de clients mobiles.



FIGURE 1.1 : logo de l'entreprise inwi

## 1.2 Historique de l'entreprise

Wanadoo Maroc a été créée en 1999 par **Karim Zaz** à **Casablanca**. En 2004, après le retrait de **France Télécom**, Wanadoo Maroc a vendu sa filiale marocaine "**Connect**" à des investisseurs locaux, **Attijariwafa bank** et la **Caisse de dépôt nationale** par l'intermédiaire de sa filiale, **Fipar holding**. En 2005, l'**ONA** est devenu l'actionnaire de référence de Maroc Connect. En 2006, Maroc Connect a remporté la troisième licence **3G** au Maroc, et en 2007, elle est devenue **Wana**, le troisième opérateur de téléphonie et d'internet 3G.

En 2009, Wana a remporté la troisième licence GSM au Maroc avec sa marque Inwi. En février de la même année, le consortium Koweïtien **Zain/Alajial** a pris le contrôle de 31% du capital de Inwi. Un an après son lancement, Inwi a déclaré **cinq millions** de clients en février 2010, et 13,5% de parts de marché en décembre 2010. En termes de parts de marché, Inwi détient, fin **juin 2013**, 26,40% du marché de la téléphonie mobile et 56,50% du marché de la téléphonie fixe. Sa part du marché de l'internet 3G au Maroc atteint 13,82%. En mars 2015, Wana a obtenu sa licence 4G pour la marque Inwi, et en septembre 2016, elle a été le premier opérateur à offrir la technologie **VoLTE** (Voice Over LTE) à ses abonnés 4G. En 2016, Inwi a également reçu le **Marocain Digital Awards** pour son initiative sociale "**Dir Iddik**".

Le 4 avril 2019, **Nadia Fassi-Fehri**, PDG du groupe, a annoncé le lancement de **Win**, un nouvel opérateur télécoms entièrement digital en partenariat avec notamment **Salesforce** et **Vlocity**. Le 30 septembre 2020, **Azzedine El Mountassir Billah** a été nommé président directeur général de transition pour Wana Corporate, avec pour mission de consacrer Wana Corporate comme un acteur significatif au service de la numérisation de ses clients et de lancer l'internationalisation de la société en fixant un cap de développement ambitieux en Afrique. En 2020, Inwi a conclu un contrat avec la **FRMF** pour devenir le nouveau sponsor du championnat national, qui s'appelle désormais "**Botola Pro Inwi**". Grâce à ses initiatives sociales et technologiques, Inwi continue de renforcer sa position sur le marché des télécommunications au Maroc.

## 1.3 Infrastructure bigdata chez inwi

### 1.3.1 Présentation du département connaissance client et performances

Le département **Connaissance client et Performances**, au cœur de la stratégie d'Inwi, constitue un pilier majeur pour garantir la satisfaction des clients et assurer la croissance de l'entreprise. Composé de plusieurs équipes aux compétences complémentaires, ce département s'emploie à analyser en profondeur les données relatives aux clients et au marché afin de répondre efficacement aux enjeux auxquels l'opérateur doit faire face. En capitalisant sur leur expertise et leur synergie, ces équipes sont en mesure d'orienter les décisions stratégiques et de soutenir l'évolution constante des offres d'Inwi pour répondre aux attentes des clients et aux exigences du marché. Le rôle crucial du département Connaissance client et Performances réside dans sa capacité à identifier les besoins et les attentes des clients, à anticiper les tendances du marché et à déceler les opportunités de croissance et d'innovation. Ce faisant, il permet à Inwi d'ajuster et d'optimiser en continu ses services et produits, en s'appuyant sur des analyses rigoureuses et des informations pertinentes. La collaboration étroite

entre les différentes équipes du département est un facteur clé de cette réussite, permettant de croiser les compétences et les perspectives pour élaborer des solutions adaptées aux enjeux spécifiques de l'opérateur.

- L'équipe **Business Intelligence et MOA** est chargée de la collecte, de l'analyse et de la présentation des données internes et externes pour faciliter la prise de décisions stratégiques. Cette équipe travaille en étroite collaboration avec les autres départements pour identifier les indicateurs clés de performance et mettre en place des tableaux de bord permettant de suivre l'évolution de ces indicateurs. Elle est également responsable de la coordination des projets transversaux, assurant ainsi la bonne exécution des initiatives stratégiques de l'entreprise.
- L'équipe **Géomarketing** est spécialisée dans l'analyse spatiale des données, permettant ainsi de comprendre et d'exploiter la dimension géographique des informations clients et du marché. Elle intervient dans la définition des zones de couverture, l'implantation des points de vente, la segmentation des clients en fonction de critères géographiques et l'optimisation des actions marketing ciblées.
- L'équipe **Market Insights** se concentre sur l'étude du marché et des tendances qui l'influencent. En analysant les informations provenant de sources variées, cette équipe fournit des insights précieux sur les attentes des clients, la concurrence et l'évolution du marché. Ces analyses permettent d'anticiper les opportunités et les défis à venir et d'adapter en conséquence les stratégies d'Inwi.
- L'équipe **Analytics** travaille sur l'analyse des données pour aider l'entreprise à prendre des décisions basées sur des informations quantitatives et qualitatives. En plus de cela, elle est également en charge du reporting pour communiquer les résultats de ses analyses et permettre aux différentes parties prenantes de suivre la performance de l'entreprise. L'équipe Analytics est donc responsable de la conception, de la mise en œuvre et du suivi des modèles économétriques permettant d'évaluer l'impact des actions marketing, ainsi que de la présentation de ces résultats de manière claire et accessible à tous.
- L'équipe **Data Science**, qui est l'équipe dans laquelle je travaille, est dédiée à l'exploration et à l'extraction d'informations pertinentes à partir de vastes ensembles de données. En utilisant des techniques avancées d'apprentissage automatique et d'intelligence artificielle, cette équipe développe des modèles prédictifs et des algorithmes pour résoudre des problèmes complexes, tels que la prédiction du churn ou la segmentation comportementale des clients. Grâce à ces approches innovantes, l'équipe Data Mining et Data Science contribue à renforcer la position d'Inwi en tant qu'acteur majeur sur le marché des télécommunications.

### 1.3.2 Flux des données : d'une antenne BTS vers une valeur business

Le flux de données chez Inwi commence au niveau des antennes **BTS** (Base Transceiver Station) et aboutit à la création de valeur business pour l'entreprise.



FIGURE 1.2 : Image d'une antenne BTS

Les antennes BTS sont des équipements essentiels dans le réseau de télécommunications, faisant le lien entre les téléphones mobiles des clients et le réseau de l'opérateur. Elles jouent un rôle central dans la communication sans fil, en convertissant les signaux radioélectriques en signaux numériques et vice versa, permettant ainsi le transfert de données et la communication vocale. Chaque fois qu'un client effectue une activité, telle qu'une **recharge de crédit**, la **réception ou l'émission d'un appel**, **l'envoi d'un SMS**, ou **l'utilisation de données mobiles**, un processus de sélection de la BTS la plus appropriée est enclenché. Ce processus prend en compte plusieurs facteurs, tels que la distance entre le client et les BTS environnantes, la qualité du signal, la capacité de la station et les interférences potentielles. Une fois la BTS la plus proche et offrant la meilleure qualité de signal sélectionnée, celle-ci détecte et traite l'activité du client en temps réel. Les métadonnées générées par l'activité du client sont ensuite capturées et enregistrées. Ces données sont essentielles pour la gestion du réseau, la facturation, la recharge, la résolution des problèmes techniques et l'amélioration de l'expérience client. En outre, elles constituent une source précieuse d'informations pour les équipes de data science, qui peuvent les utiliser pour développer des modèles prédictifs, analyser les tendances et les comportements des clients, et générer des insights stratégiques pour l'entreprise.

Une fois l'activité détectée par la BTS, les données générées sont transmises à une "**landing zone**", où elles sont collectées et stockées temporairement. La landing zone est une zone de stockage intermédiaire qui a pour fonction principale de consolider les données en provenance de différentes sources, notamment des multiples antennes BTS et autres éléments du réseau de télécommunications, avant de les traiter et de les préparer pour les analyses ultérieures. Cette étape est essentielle pour assurer la qualité, la cohérence et l'intégrité des données utilisées dans les processus d'analyse et de prise de décision. Les données brutes provenant des BTS sont souvent stockées sous forme de fichiers binaires, qui contiennent des informations détaillées sur les activités des clients et les performances du réseau. Ces fichiers binaires sont ensuite acheminés vers des data centers, où ils sont temporairement stockés dans un système de fichiers distribués, qui est l'Hadoop Distributed File System (HDFS). Celui-ci est un système de stockage conçu pour gérer de grandes quantités de données, offrant une redondance et une tolérance aux pannes pour garantir la disponibilité et la durabilité des données.

Ensuite, les données sont transférées vers une "**federated zone**", un espace de stockage qui facilite l'intégration et la fusion de diverses sources de données pertinentes. Dans cette zone, les données collectées et préparées sont combinées avec d'autres informations provenant de sources internes et

externes à l'entreprise, telles que les bases de données clients, les systèmes de facturation, les CRM, les données de géolocalisation ou les données de marché. L'objectif principal de cette étape est d'enrichir les données initiales en les complétant et en les associant à d'autres informations contextuelles, afin de les rendre plus exploitables et d'améliorer leur potentiel d'analyse. La "federated zone" permet également de transformer et d'adapter les données en fonction des besoins spécifiques de l'entreprise et des équipes qui les exploiteront. Les données peuvent être reformatées, agrégées, filtrées ou dérivées pour générer de nouvelles variables ou métriques, qui faciliteront l'analyse et la modélisation ultérieures. Cette zone permet à différentes équipes au sein de l'entreprise d'accéder et d'analyser les mêmes données, sans avoir à les déplacer ou à les dupliquer. Des outils tels que le **Hive Metastore** et **Apache Thrift** sont utilisés pour gérer et interagir avec les données dans la federated zone. Hive Metastore est un composant essentiel du framework Apache Hive, qui permet de stocker et de gérer les métadonnées des tables de données, telles que les schémas, les localisations des fichiers et les statistiques de partitionnement. Il sert de référentiel centralisé pour les métadonnées et permet aux équipes de data science et aux applications d'accéder rapidement et facilement aux informations sur les données stockées dans la federated zone. Le Hive Metastore est accessible via une interface basée sur le protocole Apache Thrift, qui est un protocole de communication inter-langages léger et performant. Apache Thrift, quant à lui, est un framework de **Remote Procedure Call** (RPC) qui permet la communication entre des applications écrites dans différents langages de programmation. Dans le contexte de la federated zone, Apache Thrift facilite l'accès aux données stockées dans le Hive Metastore, permettant aux équipes et aux applications d'interagir avec les métadonnées et de réaliser des opérations telles que la création, la modification ou la suppression de tables, ainsi que la gestion des schémas de données.

Après avoir été traitées et préparées dans la "federated zone", les données sont acheminées vers une "**analytics zone**". Cette dernière constitue l'environnement final où les données sont mises à disposition des différentes équipes du département connaissance client pour l'analyse, la modélisation et l'extraction d'insights précieux. Cette zone est conçue pour faciliter l'accès aux données et leur exploitation, en offrant des outils et des ressources adaptés aux besoins des différentes équipes. Dans l'analytics zone, les données sont organisées et structurées de manière à optimiser leur accessibilité et leur utilisation. Elles sont généralement en ensembles de données spécifiques à certains use cases ou domaines d'application, tels que la prédiction du churn, la segmentation des clients, l'analyse des comportements d'utilisation, ou encore l'optimisation des campagnes marketing. Ces bases de données sont maintenues et actualisées en continu, afin de garantir la disponibilité des informations les plus récentes et pertinentes pour les analyses.

Les équipes de data science chez Inwi exploitent une variété d'outils pour interroger, analyser et exploiter les données stockées dans l'infrastructure Big Data de l'entreprise. Chacun de ces outils présente des fonctionnalités spécifiques pour aider les data scientists à explorer, filtrer, agréger et manipuler les données, afin de mieux comprendre les tendances, les modèles et les relations entre les différentes variables.

Ces outils comportent :

- **SAS Analytics** : SAS Analytics is a robust and versatile software suite that provides advanced features for data analysis, data mining, forecasting, and statistical modeling. Data scientists at Inwi use SAS Analytics to explore and analyze data, test hypotheses, and develop complex predictive models. The SAS platform offers a user-friendly interface and visualization tools to facilitate the communication of analysis results.

- **Apache Hue** : Apache Hue is an open-source web interface that allows data scientists to interact with data stored in Hadoop systems. Hue provides an intuitive SQL editor that enables users to execute SQL queries and visualize the results in the form of interactive tables and charts. Data science teams at Inwi use Hue to explore data, filter relevant records, and aggregate information to gain useful insights.
- **Apache Impala** : Apache Impala est un moteur de requêtes SQL open source qui permet aux data scientists d'interroger et d'analyser des données stockées dans les systèmes Hadoop et Apache Hive. Conçu pour offrir des performances élevées et une faible latence, Impala permet aux équipes de data science chez Inwi de réaliser des analyses interactives sur de grandes quantités de données. Impala prend en charge les requêtes SQL complexes et facilite l'exploration et la manipulation des données pour découvrir des informations précieuses.
- **Apache Hive** : Apache Hive est un système de gestion de données open source conçu pour fonctionner avec les données stockées dans les systèmes Hadoop. Hive permet aux data scientists d'exécuter des requêtes SQL sur des données structurées et semi-structurées, en utilisant un langage de requête appelé HiveQL. Les équipes de data science chez Inwi utilisent Apache Hive pour stocker, organiser et interroger les données, en tirant parti de la scalabilité et de la flexibilité offertes par l'écosystème Hadoop.

Pour des analyses plus approfondies et l'entraînement de modèles de machine learning, les équipes de data science chez Inwi utilisent également JupyterLab et PySpark, deux technologies complémentaires qui offrent des fonctionnalités avancées pour explorer et exploiter les données.

- **JupyterLab** : JupyterLab est un environnement de développement interactif et évolutif pour les notebooks Jupyter. Il permet aux data scientists d'écrire, d'exécuter et de partager du code, des visualisations et des explications textuelles dans un format structuré et facile à comprendre. Les notebooks Jupyter supportent plusieurs langages de programmation, tels que Python, R et Scala, ce qui permet aux équipes de data science chez Inwi de choisir le langage le plus adapté à leur projet. De plus, JupyterLab offre une intégration avec de nombreuses bibliothèques et outils de data science, tels que NumPy, pandas, scikit-learn et TensorFlow, pour faciliter l'analyse des données, la visualisation et la modélisation statistique.
- **PySpark** : PySpark est la bibliothèque Python pour Apache Spark, un moteur de traitement de données distribué qui est spécialement conçu pour traiter de grandes quantités de données et pour faciliter l'implémentation d'algorithmes de machine learning. PySpark fournit une API Python conviviale qui permet aux data scientists de tirer parti des fonctionnalités de traitement parallèle et distribué offertes par Spark, tout en travaillant dans un langage de programmation familier. Les équipes de data science chez Inwi utilisent PySpark pour effectuer des transformations de données, des agrégations et des jointures sur de grands ensembles de données, ainsi que pour entraîner et évaluer des modèles de machine learning à l'aide d'algorithmes avancés et optimisés pour des environnements distribués.



# Cadre du stage

## Introduction

Dans ce chapitre, nous allons aborder le cadre spécifique du stage qui a donné lieu à ce rapport. Nous commencerons par présenter la problématique à laquelle nous avons été confrontés. Ensuite, nous détaillerons la solution que nous avons proposée pour répondre à ce défi.

## 2.1 Problématique

L'essor de la data science et de l'apprentissage automatique a profondément transformé les entreprises et leur façon de travailler. Les projets de data science permettent d'extraire des informations précieuses à partir de grandes quantités de données, d'optimiser les processus métier et de prendre des décisions éclairées. Cependant, ces projets peuvent également s'avérer complexes et chronophages, ce qui limite le nombre de projets de data science réalisés chaque année et réduit leur impact potentiel sur l'organisation. L'un des principaux défis auxquels sont confrontées les équipes de data science dans la réalisation de projets réside dans l'écriture manuelle des agrégations pour générer des données d'entraînement. Cette tâche cruciale, qui consiste à combiner et synthétiser les données pour faciliter l'analyse et la modélisation, s'avère souvent extrêmement complexe et exigeante en termes de temps et de compétences. Parmi les principaux défis, on trouve :

1. **Le passage d'un format log vers un format entité** : Les données temporelles des clients, souvent sous forme de logs, sont des enregistrements chronologiques d'événements ou d'actions effectuées par les clients au fil du temps. Ces données sont particulièrement précieuses pour les projets de machine learning, car elles permettent de capturer les dynamiques temporelles et les modèles de comportement des clients. Cependant, la préparation de ces données pour les modèles de machine learning présente plusieurs défis, notamment le passage d'un format de log à un format entité. Dans un format de log, les événements sont enregistrés de manière séquentielle, avec des informations détaillées sur chaque action effectuée par un client à un moment donné. Ces données peuvent inclure des informations telles que l'heure et la date de l'événement, l'identifiant du client (MDN), le type d'événement (recharge, appel, données mobiles...), ainsi que des attributs spécifiques à l'événement. Bien que ce format soit utile pour

stocker et analyser les données de manière chronologique, il n'est pas idéal pour l'entraînement des modèles de machine learning, qui nécessitent généralement des données sous forme d'entités. Le format entité, en revanche, organise les données en unités distinctes, où chaque entité représente un "blueprint" d'un client. Ces entités contiennent des informations agrégées sur le comportement et les caractéristiques d'un client sur une période donnée, permettant aux modèles de machine learning d'apprendre et de faire des prédictions sur la base de ces informations. Pour passer d'un format de log à un format entité, les équipes de data science doivent transformer et agréger les données de manière à ce qu'elles soient structurées autour de chaque client individuel. Cette transformation peut être coûteuse et complexe à réaliser manuellement, car elle implique généralement plusieurs étapes, telles que le regroupement des événements par client, l'agrégation des événements sur différentes périodes ou fenêtres temporelles, et la création de nouvelles variables ou de nouvelles caractéristiques à partir des données agrégées. Ces étapes peuvent être difficiles à mettre en œuvre et à maintenir, en particulier lorsque les données sont volumineuses, diverses et dynamiques.

2. **Le long temps de l'écriture des agrégations** : L'écriture manuelle des agrégations peut prendre jusqu'à **5 à 6 mois** pour certains projets de data science complexes, ce qui représente une part importante du temps total consacré au projet. Cette situation peut avoir un impact considérable sur la progression des projets et la productivité globale des équipes de data science, car elles doivent consacrer une part importante de leur temps et de leurs ressources à cette étape critique du processus. De plus, l'écriture manuelle des agrégations peut également entraîner des erreurs et des incohérences dans les données d'entraînement, ce qui peut nuire à la performance des modèles de machine learning et, par conséquent, à la qualité des résultats obtenus. Des recherches ont révélé que près de **80%** des projets de data science sont retardés en raison de problèmes liés à la qualité des données, dont beaucoup sont directement liés à l'écriture manuelle des agrégations.
3. **Risque et coût des prestataires externes** : Dans de nombreux cas, les entreprises sont contraintes de faire appel à des prestataires externes pour combler les lacunes en matière de compétences et de ressources. Le recours à des consultants ou des entreprises spécialisées en data science peut offrir une solution temporaire pour résoudre les problèmes liés à l'écriture manuelle des agrégations et répondre aux besoins spécifiques des projets. Cependant, cette approche présente également plusieurs inconvénients et risques potentiels. Tout d'abord, le recours à des prestataires externes peut engendrer des coûts supplémentaires significatifs pour l'entreprise. Les frais de consultation et de gestion de projet associés à l'embauche d'experts externes peuvent être élevés, en particulier pour les projets de longue durée ou les projets impliquant des compétences très spécialisées. De plus, les coûts indirects, tels que la coordination entre les équipes internes et externes, la communication et la gestion des délais, peuvent également peser sur le budget global du projet. Ensuite, les prestataires externes peuvent ne pas être aussi familiarisés avec les objectifs et les besoins spécifiques de l'entreprise, ce qui peut entraîner des erreurs ou des incohérences dans les agrégations de données. Les prestataires peuvent ne pas avoir une compréhension approfondie des processus métier, des indicateurs clés de performance (KPI) et des exigences réglementaires de l'entreprise, ce qui peut entraîner des problèmes de qualité dans les données d'entraînement et, par conséquent, affecter la performance des modèles de machine learning. Par ailleurs, la dépendance à l'égard de prestataires externes peut poser des problèmes de confidentialité et de sécurité des données. Le partage de données sensibles



avec des tiers peut exposer l'entreprise à des risques en termes de fuites d'informations, de violations de la réglementation sur la protection des données, ou d'atteintes à la propriété intellectuelle. Pour minimiser ces risques, les entreprises doivent mettre en place des protocoles de sécurité rigoureux, tels que des accords de non-divulgence (NDA), des politiques d'accès aux données et des procédures de contrôle et de suivi des prestataires externes. Enfin, le recours à des prestataires externes peut également créer une dépendance à long terme, ce qui peut rendre l'entreprise vulnérable en cas de changement de prestataire ou de rupture de la relation contractuelle. Pour éviter cette situation, les entreprises devraient envisager d'investir dans le développement de compétences internes, la formation et la montée en compétences de leurs employés, ainsi que l'adoption d'outils et de technologies d'automatisation pour faciliter l'écriture des agrégations et réduire la dépendance à l'égard des prestataires externes.

4. **Le risque de s'engager dans un projet sans valeur** : La génération manuelle des données d'entraînement pour les modèles de machine learning soulève un défi majeur pour les équipes de data science en termes d'évaluation du potentiel des données et de la faisabilité des projets de data science. La préparation manuelle de ces données implique un investissement considérable en temps et en ressources humaines, ce qui limite la capacité des équipes à explorer et à expérimenter différents scénarios avant de s'engager dans un projet spécifique. De ce fait, elles sont souvent contraintes de se lancer dans un projet sans avoir la certitude que les données disponibles seront suffisamment informatives et pertinentes pour générer des modèles performants et fournir des insights business valables. Cette incertitude peut entraîner un niveau de risque élevé pour les organisations qui investissent dans des projets de data science. Les entreprises peuvent se retrouver à allouer d'importantes ressources, telles que le temps, l'argent et les compétences de leur personnel, sans avoir l'assurance que le projet aboutira à des résultats satisfaisants. Dans ce contexte, les projets de data science peuvent s'avérer être des impasses extrêmement coûteuses et interminables, sans offrir de réelles perspectives de succès. En conséquence, les entreprises pourraient perdre du temps, de l'argent et des opportunités de croissance et d'innovation en s'engageant dans des projets de data science qui ne sont pas viables ou prometteurs. L'incapacité d'évaluer le potentiel des données et la faisabilité des projets de data science lors de la préparation manuelle des données d'entraînement peut ainsi avoir des conséquences négatives sur la performance globale de l'entreprise et sur la réalisation de ses objectifs stratégiques.
5. **Biais humain lors du choix des agrégations** : l'intervention des experts métier est souvent nécessaire. Ceux-ci possèdent une connaissance approfondie du domaine et des données associées. En s'appuyant sur leur expérience et leur compréhension du contexte, ces experts sont capables d'identifier les tables et les colonnes pertinentes à inclure dans les jeux de données d'entraînement. Cependant, cette approche présente certains inconvénients, car elle peut limiter la diversité et l'étendue des combinaisons de variables explorées lors de la préparation des données d'entraînement. Lorsque les données d'entraînement sont basées principalement sur l'expertise de l'expert métier, il est possible que certaines combinaisons potentiellement importantes de variables ne soient pas prises en compte. Cela peut être dû à des biais inhérents à l'expertise humaine, à des lacunes dans la connaissance ou à une incapacité à envisager toutes les combinaisons possibles. En conséquence, les données d'entraînement peuvent ne pas refléter pleinement le potentiel des données disponibles et peuvent conduire à des modèles de machine learning moins performants et moins généralisables. De plus, cette approche peut éga-

lement entraîner une sous-utilisation des techniques statistiques et des méthodes d'exploration de données automatisées, qui sont capables d'identifier des relations complexes et des interactions entre les variables de manière plus systématique et rigoureuse. En se reposant principalement sur l'expertise de l'expert métier, on risque de passer à côté de ces précieuses informations et de ne pas exploiter pleinement les données disponibles pour l'entraînement des modèles.

6. **Peu de temps est consacré à la compréhension du modèle** : Les équipes de data science consacrent généralement une part importante de leurs ressources et de leur temps à cette phase laborieuse, ce qui peut entraîner une réduction du temps disponible pour se concentrer sur d'autres aspects essentiels du projet, tels que l'analyse des résultats des modèles, l'interprétation des données et l'extraction des insights business. Lorsque les équipes de data science passent trop de temps à préparer manuellement les données d'entraînement, elles peuvent ne pas disposer d'un laps de temps suffisant pour effectuer une analyse approfondie des résultats des modèles de machine learning. Cela peut limiter leur capacité à comprendre les relations complexes et les interactions entre les variables, à identifier les mécanismes sous-jacents qui expliquent les performances du modèle et à déterminer les meilleures stratégies pour optimiser et améliorer les modèles. Par conséquent, les équipes peuvent être contraintes de prendre des décisions basées sur des informations incomplètes ou une compréhension limitée des modèles, ce qui peut nuire à leur efficacité globale. En outre, un manque de temps pour l'interprétation des résultats peut également entraver la capacité des équipes de data science à tirer des insights business pertinents et exploitables à partir des modèles. Les insights business sont cruciaux pour permettre aux entreprises de prendre des décisions éclairées et d'orienter leurs actions et leurs investissements de manière plus efficace. Si les équipes de data science ne parviennent pas à fournir ces insights en temps voulu, cela peut avoir un impact négatif sur la performance globale de l'entreprise et sur la réalisation de ses objectifs stratégiques. De plus, le temps considérable consacré à la génération des résultats des données d'entraînement peut également résulter en une attention réduite portée à la compréhension du modèle lui-même. Les équipes de data science pourraient ne pas être en mesure d'analyser les comportements du modèle, d'évaluer l'importance des différentes variables, ou de détecter et d'expliquer d'éventuels biais dans les résultats du modèle. Cette situation peut entraîner des erreurs de modélisation et des inefficacités qui pourraient autrement être évitées si suffisamment de temps était alloué à l'étude du modèle.
7. **Difficulté d'expérimentation** : Un autre défi majeur pour les équipes de data science concerne l'expérimentation avec différents paramètres métiers. En effet, l'adaptation des données d'entraînement pour tester différentes combinaisons de paramètres peut être un processus long et laborieux, nécessitant souvent de réécrire des scripts d'agrégation et de prétraitement des données à chaque fois. Cette contrainte limite la capacité des équipes à explorer et à évaluer l'impact de diverses combinaisons de paramètres sur les résultats des modèles, ce qui peut entraîner une sous-optimisation des modèles et une moins bonne performance globale. Lorsque les équipes de data science n'ont pas la possibilité de tester facilement différentes combinaisons de paramètres métiers, elles peuvent passer à côté de configurations potentiellement plus performantes et efficaces. Cela peut également les empêcher de détecter des problèmes et des inefficacités dans les modèles qui pourraient être résolus en ajustant les paramètres métiers appropriés. De plus, la génération manuelle des données d'entraînement limite la possibilité d'itérer rapidement et d'améliorer les modèles, ce qui est essentiel pour garantir des perfor-

mances optimales et pour s'adapter aux changements de conditions et de contexte. L'incapacité de mener des expériences efficaces avec différents paramètres métiers peut également avoir un impact sur la compréhension globale des modèles de machine learning par les équipes de data science. Sans la possibilité de tester différentes combinaisons de paramètres, il est plus difficile de comprendre comment chaque paramètre influence les résultats du modèle et d'acquérir des connaissances précieuses sur les mécanismes sous-jacents des modèles.

## 2.2 Solution

Face à la problématique évoquée précédemment concernant la préparation manuelle des données d'entraînement, une solution prometteuse consisterait à développer un système intelligent de génération automatique de données d'entraînement à partir d'un fichier de configuration contenant divers paramètres. Ce système vise à résoudre les problèmes liés à la préparation manuelle des données et à améliorer l'efficacité et la productivité des équipes de data science.

Grâce aux paramètres fournis par les utilisateurs, le système intelligent est capable de générer automatiquement un fichier SQL qui permet la création de la base active, des labels et des différentes agrégations, en tenant compte des informations spécifiées dans le fichier de configuration. Ce système simplifie le processus de préparation des données d'entraînement pour les modèles de machine learning en automatisant les étapes clés. Un aspect important de cette solution est qu'elle permet d'effectuer automatiquement des opérations de "group by" par client lors de la génération des agrégations. Cela facilite le passage d'un format de données basé sur des logs, où les événements sont enregistrés de manière séquentielle, à un format entité, où chaque entité représente un "blueprint" d'un client. En conséquence, les données sont plus faciles à analyser et à utiliser pour l'entraînement des modèles de machine learning. Le système intelligent génère également toutes sortes de combinaisons statistiques entre les colonnes en fonction de leur type, optimisant ainsi l'utilisation des informations disponibles. De plus, étant donné qu'il s'agit de données temporelles, le système prend en compte les variations temporelles et les fenêtres de temps de différentes tailles lors de la génération des agrégations. Cette approche permet d'obtenir des données d'entraînement plus complètes et représentatives, en tenant compte de la dynamique temporelle et de l'évolution des comportements des clients.

Le système intelligent intègre également un mécanisme permettant d'effectuer automatiquement des opérations de "left join" avec la base active lors de la génération des agrégations. Cette approche garantit que seuls les clients actifs sont pris en compte dans le processus, en fonction des différentes définitions d'une base active fournies par l'utilisateur dans le fichier de configuration. En filtrant les données pour ne conserver que les clients actifs, le système permet d'obtenir des agrégations plus pertinentes et plus représentatives du comportement des clients d'intérêt. Cette démarche est essentielle pour garantir que les modèles de machine learning développés à partir de ces données d'entraînement sont bien adaptés aux besoins spécifiques de l'entreprise et qu'ils fournissent des résultats significatifs et exploitables.

Ainsi, le système de génération automatique de données d'entraînement offre une solution complète et flexible pour la préparation des données destinées aux modèles de machine learning. En automatisant les étapes clés du processus, notamment la des différents paramètres métier, sélection des tables et des colonnes, la génération de la base active, la labellisation des données, l'agrégation

des données, la prise en compte des variations temporelles et des fenêtres de temps, le système permet aux équipes de data science de se concentrer sur l'analyse des résultats et l'extraction d'insights business pertinents, tout en optimisant l'utilisation des ressources et en réduisant les risques associés aux projets de data science.

## 2.3 Planification et calendrier des activités

Le diagramme de Gantt suivant représente graphiquement le calendrier des différentes activités et tâches au cours de cette expérience professionnelle.

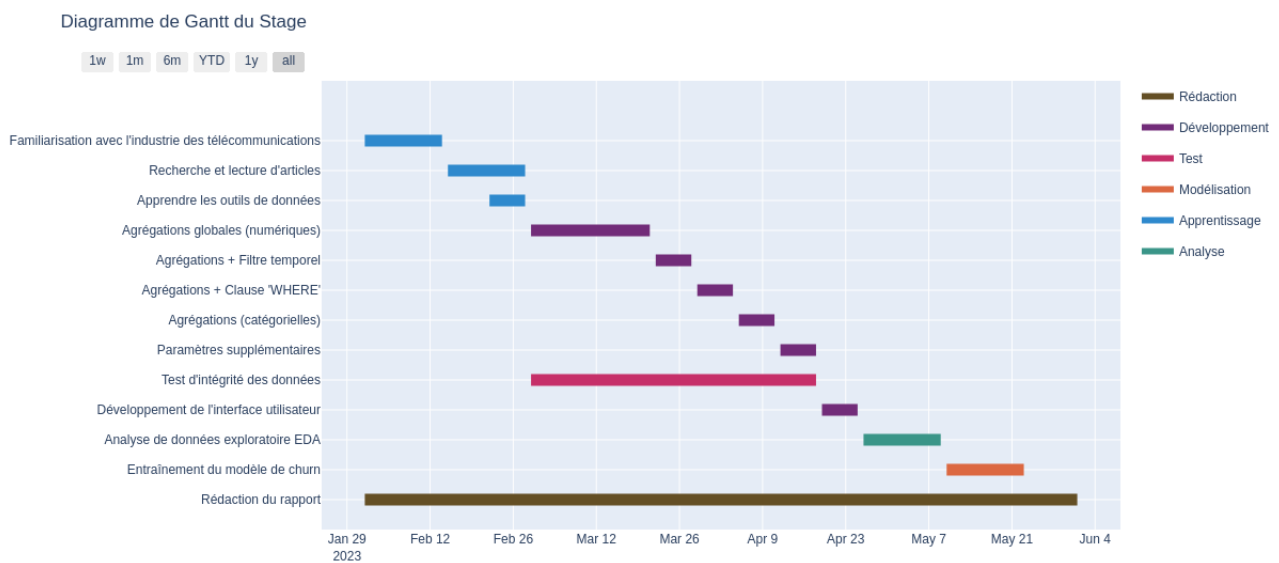


FIGURE 2.1 : Diagramme de Gantt

# Paramètres métier

## Introduction

Dans le domaine de la data science, il est crucial de bien comprendre le métier sous tous ses aspects pour être en mesure de mener des analyses pertinentes, d'entraîner des modèles performants et d'interpréter correctement leurs résultats. La compréhension approfondie du métier permet aux équipes de data science de cerner les enjeux business et d'aligner leurs efforts sur les objectifs stratégiques des entreprises. Ainsi, l'importance de bien maîtriser les différents paramètres métier spécifiques au domaine de la télécommunication ne saurait être sous-estimée, en particulier ceux concernant la problématique du churn, un enjeu majeur pour les opérateurs de télécommunication.

Dans les paragraphes suivants, nous aborderons en détail les différents paramètres métier essentiels à considérer lors de la préparation des données d'entraînement pour les modèles de machine learning dans le contexte de la télécommunication.

## 3.1 churn

### 3.1.1 définition

Le churn [1], également connu sous le nom d'attrition ou de taux d'attrition, est un concept clé dans diverses industries qui mesure le taux de clients qui cessent d'utiliser les services ou les produits d'une entreprise sur une période donnée. Comprendre et gérer le churn est essentiel pour la réussite d'une entreprise, car il reflète sa capacité à fidéliser sa clientèle et à maintenir une base de clients stable. Un taux de churn élevé peut indiquer des problèmes tels que l'insatisfaction des clients, une concurrence accrue, des offres moins attractives ou d'autres facteurs qui peuvent impacter négativement la rentabilité et la croissance de l'entreprise. Dans le secteur des télécommunications, le churn est d'une importance particulière en raison de la concurrence intense et des investissements importants requis pour acquérir et retenir les clients. Le churn dans les télécommunications peut être classé en différentes catégories, telles que le churn volontaire et le churn involontaire. Le churn volontaire se produit lorsque les clients décident activement de mettre fin à leur relation avec l'opérateur, par exemple en raison d'une insatisfaction vis-à-vis de la qualité du service, de prix plus élevés par rapport à la concurrence, ou de l'attrait d'offres promotionnelles proposées par d'autres opérateurs. Le

churn involontaire, en revanche, survient lorsque les clients cessent d'utiliser les services de l'opérateur pour des raisons indépendantes de leur volonté, comme le changement d'un emploi utilisant une flotte ayant un certain opérateur, un déménagement à l'étranger, ou des problèmes techniques rendant le service inutilisable.



FIGURE 3.1 : Illustration de seau qui fuit représentant le churn des clients

Le calcul du taux de churn dans les télécommunications peut varier en fonction des critères et des méthodes utilisées. Généralement, il est calculé en divisant le nombre de clients ayant quitté l'opérateur pendant une période donnée par le nombre total de clients au début de cette période. Il peut être exprimé en pourcentage ou en valeur absolue, et il peut être analysé sur différentes périodes (mensuelle, trimestrielle, annuelle) pour mieux comprendre les tendances et les variations saisonnières. Analyser et gérer efficacement le churn dans les télécommunications est crucial pour assurer la rentabilité et la croissance à long terme de l'entreprise, car la rétention des clients existants est souvent moins coûteuse que l'acquisition de nouveaux clients.

$$\text{taux du churn} = \left( \frac{\text{nombre de clients ayant quitté l'opérateur pendant une période donnée}}{\text{nombre total de clients au début de cette période}} \right) \times 100$$

### 3.1.2 Acquisition des nouveaux clients vs rétention des anciens clients

La rétention des clients existants présente de nombreux avantages par rapport à l'acquisition de nouveaux clients, tant en termes de coûts qu'en termes d'impact global sur l'entreprise. Comme mentionné précédemment, la rétention des clients existants est souvent considérée comme moins coûteuse que l'acquisition de nouveaux clients en raison des dépenses de marketing et des promotions nécessaires pour attirer de nouveaux clients. Cependant, il existe d'autres avantages importants liés à la rétention des clients qui renforcent davantage l'importance de combattre le churn.

Tout d'abord, les clients fidèles peuvent fournir des revenus stables et prévisibles à l'entreprise. La fidélisation des clients existants permet de maintenir une base de revenus solide et de réduire la volatilité des revenus causée par la perte de clients et l'incertitude liée à l'acquisition de nouveaux clients. Les entreprises ayant une forte rétention des clients sont également mieux positionnées pour résister aux fluctuations du marché et aux changements dans l'environnement concurrentiel. Ensuite, la rétention des clients permet d'obtenir de précieux retours d'information sur les produits et les services de l'entreprise. Les clients fidèles et satisfaits sont souvent plus enclins à partager leurs opinions et leurs suggestions, ce qui peut aider l'entreprise à identifier les domaines d'amélioration et à inno-



ver pour mieux répondre aux besoins de sa clientèle. Cette rétroaction peut être un atout précieux pour l'amélioration continue des produits et services, ainsi que pour le développement de nouvelles offres pour répondre aux demandes changeantes du marché. De plus, la rétention des clients peut avoir un effet positif sur la réputation de l'entreprise et sa marque. Les clients satisfaits sont plus susceptibles de parler positivement de l'entreprise à leurs amis, leur famille et leurs collègues, ce qui peut renforcer la réputation de l'entreprise et augmenter sa visibilité sur le marché. Cet effet de bouche-à-oreille peut être un puissant outil de marketing, car les recommandations personnelles sont souvent considérées comme plus fiables et convaincantes que les publicités ou les promotions traditionnelles. Enfin, la rétention des clients peut également contribuer à améliorer l'efficacité opérationnelle de l'entreprise. Les clients fidèles sont généralement plus familiers avec les processus et les systèmes de l'entreprise, ce qui peut réduire le temps et les ressources consacrés à l'assistance et à la formation des nouveaux clients. De plus, les entreprises peuvent tirer parti des connaissances et de l'expérience des clients existants pour identifier les inefficacités et les opportunités d'amélioration dans leurs opérations.

L'acquisition de nouveaux clients présente plusieurs inconvénients qui peuvent nuire à la performance globale d'une entreprise, en particulier lorsqu'elle est comparée aux avantages de la rétention des clients existants. Parmi les désavantages liés à l'acquisition de nouveaux clients, on trouve les coûts élevés, puisque les dépenses de marketing, les promotions et les remises offertes pour attirer de nouveaux clients peuvent être onéreuses. Les entreprises doivent investir des ressources importantes pour identifier, cibler et convertir de nouveaux clients, ce qui peut affecter leur rentabilité et leur efficacité opérationnelle. De plus, l'acquisition de nouveaux clients nécessite généralement beaucoup de temps et d'efforts pour convaincre les prospects de changer de fournisseur ou de s'engager dans un nouvel achat. Il peut falloir surmonter des obstacles tels que la fidélité à la marque, l'inertie du client ou les préoccupations concernant les coûts et la qualité du service, ce qui peut entraîner un processus d'acquisition long et complexe. Les nouveaux clients peuvent également présenter un risque d'attrition précoce, surtout s'ils ont été attirés par des offres promotionnelles ou des remises temporaires. Une fois ces avantages expirés, les clients peuvent chercher d'autres options ou retourner chez leur précédent fournisseur, ce qui peut entraîner un taux de churn élevé parmi les nouveaux clients. La rentabilité incertaine est un autre désavantage, car il n'est pas garanti que les nouveaux clients deviendront rentables pour l'entreprise. Certains peuvent ne pas générer suffisamment de revenus pour couvrir les coûts d'acquisition, tandis que d'autres peuvent nécessiter un soutien et une assistance supplémentaires, ce qui augmente les coûts opérationnels. Enfin, en se concentrant trop sur l'acquisition de nouveaux clients, une entreprise peut compromettre la qualité de son offre et la satisfaction de sa clientèle existante. Les ressources consacrées à l'acquisition de nouveaux clients peuvent réduire les investissements dans l'amélioration des produits et services existants, ce qui peut entraîner une détérioration de la qualité et de la réputation de la marque.

## 3.2 Postpayé vs prépayé

Dans l'industrie des télécommunications, les termes "prépayé" et "postpayé" font référence à deux types de plans tarifaires couramment offerts par les opérateurs de téléphonie mobile.

Le plan **prépayé**, comme son nom l'indique, implique que les clients paient à l'avance pour les services qu'ils prévoient d'utiliser (une recharge grattable, transfert de la recharge par dealer, envoi

de la recharge par application bancaire. . .). Cela signifie qu'ils achètent un certain nombre de minutes d'appel, de messages textes ou de données internet, qui sont ensuite déduits de leur solde à mesure qu'ils utilisent le service. Une fois que ce solde est épuisé, le service est généralement interrompu jusqu'à ce que le client recharge son compte. Ce type de plan offre une grande flexibilité et un contrôle total sur les dépenses, car il n'y a pas de factures mensuelles fixes et les clients ne peuvent pas dépenser plus que le montant qu'ils ont prépayé. Les plans prépayés sont souvent préférés par ceux qui ont des besoins d'utilisation variables, ceux qui souhaitent éviter les contrats à long terme, ou ceux qui ont des préoccupations concernant le crédit ou le budget.

D'autre part, le plan **postpayé** fonctionne sur la base d'un abonnement mensuel fixe où les clients sont facturés après avoir utilisé les services. Les clients souscrivent à un plan spécifique qui comprend une certaine allocation de minutes d'appel, de messages textes et de données internet pour une redevance mensuelle fixe. Si les clients dépassent leur allocation, ils sont généralement facturés à un taux spécifié pour leur utilisation supplémentaire. Les plans postpayés sont souvent assortis d'un engagement contractuel à long terme, généralement de 6 à 12 mois. Ces plans sont généralement préférés par les utilisateurs fréquents de services de téléphonie mobile qui préfèrent la commodité d'un accès ininterrompu et illimité aux services et la possibilité de payer après utilisation. Ce plan ne concerne pas seulement les forfaits et abonnements mobiles, mais aussi l'ADSL, la fibre optique et la box i-dar de inwi

La labellisation des clients en termes de churn varie considérablement entre les plans postpayés et prépayés. Pour les clients postpayés, la détermination du churn est relativement simple. Comme ces clients ont un contrat de forfait avec l'opérateur, leur statut de churn peut être clairement identifié lorsqu'ils résilient ce contrat. Cette résiliation est généralement bien documentée, permettant à l'opérateur de connaître la date exacte du churn et de suivre précisément l'évolution du taux de churn. Cependant, pour les clients prépayés, la définition du churn est beaucoup plus complexe et abstraite. Ces clients n'ont pas de contrat formel avec l'opérateur, mais effectuent plutôt des recharges à leur gré. Par conséquent, il n'y a pas de moment clair et défini où un client prépayé peut être considéré comme ayant churné. Un client peut cesser de recharger son compte pendant une période, puis revenir et recharger à nouveau. Par conséquent, la détermination du moment exact du churn d'un client prépayé nécessite l'établissement d'une définition claire basée sur des critères spécifiques. La définition de ces critères sera discutée et élaborée plus en détail dans les paragraphes suivants. En ce qui concerne le système intelligent de génération automatique des données d'entraînement pour les modèles de prédiction du churn, celui-ci se concentre principalement sur les clients prépayés. La complexité inhérente à la définition du churn pour les clients prépayés rend ce système particulièrement précieux. Il permet aux data scientists de tester différentes définitions de churn sans avoir à réécrire les scripts d'agrégation de données à chaque fois. En d'autres termes, ce système intelligent offre une flexibilité et une adaptabilité importantes dans l'exploration des meilleures façons de définir et de comprendre le churn pour les clients prépayés. Cela permet une expérimentation rapide et efficace, ce qui peut conduire à des définitions plus précises et plus utiles de churn, et en fin de compte aboutir à des modèles de prédiction de churn plus robustes.



### 3.3 Cutoff\_date

La '**cutoff date**', ou date limite, est une notion clé en matière d'analyse prédictive et de modélisation des données. Elle correspond à un point précis dans le temps qui sert de frontière entre le passé et le futur lors de la préparation des données d'entraînement pour un modèle de prédiction. Cette date est déterminante car elle permet de segmenter les données en deux ensembles : les données historiques, qui sont utilisées pour entraîner le modèle, et les données futures, que le modèle cherche à prédire. En d'autres termes, toutes les données postérieures à cette date ne sont pas incluses dans l'entraînement du modèle, mais sont utilisées pour la labélisation. Ainsi, en se basant sur les informations disponibles jusqu'à la '**cutoff date**', le modèle doit être capable de prédire quels clients vont churner dans le futur. Cela permet de tester la capacité du modèle à généraliser à partir des données d'entraînement pour faire des prédictions précises sur des données non vues auparavant.

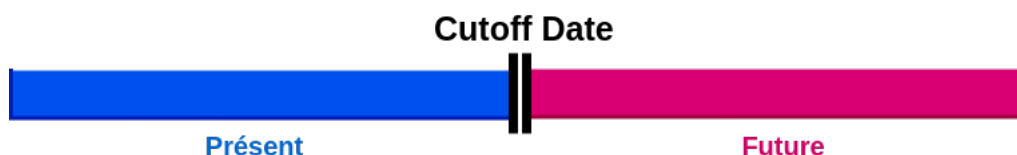


FIGURE 3.2 : Diagramme Cutoff date

Le paramètre de '**cutoff date**' constitue en effet le fondement de toutes les étapes suivantes dans le processus de prédiction du churn. Il s'agit d'un repère temporel crucial qui influe directement sur la définition des autres paramètres. C'est en quelque sorte le pivot autour duquel s'articulent toutes ces définitions. Dans les sections suivantes, nous examinerons plus en détail comment ce paramètre clé sert de point d'ancrage permettant le calcul des différents intervalles temporels en s'alliant avec d'autres paramètres métier pour faciliter la mise en place du processus de prédiction du churn. Nous verrons comment il intervient dans la définition de la base active des clients, la labellisation des clients en tant que churneurs ou non-churneurs, ainsi que la construction de l'historique des données d'entraînement. Ces éléments constituent ensemble le cœur de l'approche prédictive du churn, et leur bonne compréhension et mise en œuvre sont essentielles pour le succès de cette démarche.

### 3.4 Base actif

#### 3.4.1 Définition

La "**base active**" fait référence à l'ensemble des clients qui ont été actifs pendant une certaine période de temps. Cette notion est couramment utilisée dans divers secteurs, et notamment dans l'industrie des télécommunications. Dans ce contexte, un client actif est généralement défini comme un client qui a utilisé les services de l'opérateur (émission ou réception des appels, utilisation des données internet, recharge du compte ...) pendant une période de temps définie.

Deux paramètres principaux sont pris en compte pour définir la base active dans le contexte de la prédiction du churn : le **paramètre d'activité** et le **paramètre de tenure** ou d'ancienneté.

Le paramètre d'activité est un paramètre clé utilisé pour définir la "base active" d'une entreprise.

C'est une date spécifique, généralement exprimée en nombre de semaines, qui sert de point de référence pour définir un intervalle de temps sous la forme [cutoff date - paramètre d'activité, cutoff date]. Par exemple, si la cutoff date est fixée au **1er février 2023** et que le paramètre d'activité est fixé à **4 semaines** avant, alors l'intervalle de temps d'activité serait [**1er janvier 2023, 1 février 2023**]. L'activité peut être définie de différentes manières en fonction des spécificités de l'entreprise et de son secteur d'activité. Dans l'industrie des télécommunications, l'activité du client peut être quantifiée de deux manières différentes, à savoir, par la "**somme**" ou par le "**count**" :

- **Somme** : Pour l'exemple des recharges, on effectue la somme des montants de recharge pour chaque client pendant l'intervalle d'activité [paramètre d'activité, paramètre d'activité + cutoff date]. Les clients dont le montant total des recharges dépasse un certain seuil sont alors considérés comme actifs.
- **Compte** : Dans le même contexte des recharges, on compte le nombre de recharges effectuées par chaque client. Si le nombre de recharges effectuées par un client dépasse un certain seuil pendant l'intervalle spécifié, ce client est alors considéré comme actif.

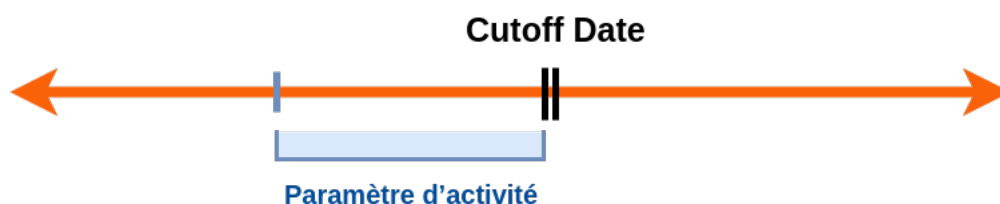


FIGURE 3.3 : Diagramme paramètre d'activité

Le paramètre de "**tenure**", ou ancienneté, est un autre critère important utilisé pour définir la base active d'une entreprise. Il s'agit d'une mesure du temps pendant lequel un client a été actif avec l'entreprise, généralement exprimée en semaines. Il est utilisé pour ajouter une condition d'ancienneté aux critères d'activité d'un client. Par exemple, si le paramètre de "**tenure**" est fixé à **12 semaines**, cela signifie que pour être considéré comme actif, un client doit non seulement avoir réalisé une activité pendant l'intervalle de temps défini par le paramètre d'activité et la date de coupure, mais aussi avoir été client de l'entreprise pendant au moins 12 semaines. Cela garantit que seuls les clients ayant un historique de données complet sont inclus dans la base active. L'importance de ce paramètre réside dans le fait qu'il permet d'éliminer les nouveaux clients qui n'ont pas encore accumulé un historique de données suffisant pour permettre une prédiction précise de leur comportement futur. En effet, si un client est trop récent, il peut y avoir des lacunes dans les données disponibles pour ce client, ce qui pourrait entraîner des prédictions moins précises ou même trompeuses. Le calcul de l'ancienneté d'un client dans le contexte de la télécommunication est une opération cruciale qui doit prendre en compte certaines spécificités du secteur. Traditionnellement, l'ancienneté pourrait être calculée en se basant simplement sur la date d'activation de la carte SIM du client. Cependant, ce calcul peut être trompeur en raison de pratiques courantes dans cette industrie. Par exemple, de nombreuses cartes SIM sont vendues par des revendeurs externes. Parfois, ces revendeurs peuvent utiliser les cartes SIM avant de les vendre aux clients. Ils peuvent le faire pour diverses raisons, souvent pour profiter du crédit préchargé sur la carte SIM. En conséquence, la date d'activation de la carte SIM n'est pas nécessairement la date à laquelle le client a commencé à utiliser les services de l'opérateur de

télécommunications. Pour contourner ce problème, les opérateurs de télécommunications peuvent utiliser la **date CPR1** pour calculer l'ancienneté d'un client. La date CPR1 représente généralement la date de la première recharge effectuée par le client. On peut raisonnablement supposer que cette date est plus proche du moment où le client a réellement commencé à utiliser les services de l'opérateur. L'ancienneté est alors calculée en soustrayant la date CPR1 de la date de coupure. Cela donne une mesure plus précise de la durée pendant laquelle le client a été actif avec l'opérateur.

La mise en place d'une base active permet de garantir que seuls les clients ayant un historique de données complet et récent sont inclus dans l'entraînement du modèle. Cela permet d'éviter le phénomène de **"data leakage"**, qui pourrait fausser les performances du modèle. Nous reviendrons plus en détail sur ce phénomène dans la partie suivante.

### 3.4.2 Data leakage

Le "data leakage" est une problématique majeure en apprentissage automatique qui peut compromettre l'intégrité et la fiabilité d'un modèle de prédiction. Lorsqu'on parle de fuite de données, on se réfère à une situation où des informations normalement inaccessibles ou inconnues au moment de la prédiction sont utilisées pour construire le modèle. Cela peut se produire si, par exemple, des données futures ou des données de l'ensemble de test, qui sont censées rester inconnues pendant l'entraînement, sont intégrées dans l'ensemble d'entraînement. La conséquence de ce phénomène est que le modèle peut présenter une performance artificiellement élevée lors de l'entraînement, car il a accès à des informations qu'il ne devrait pas avoir. Cela peut donner une fausse impression de précision et d'efficacité du modèle. Toutefois, lorsque ce modèle est utilisé pour faire des prédictions sur de nouvelles données, sa performance peut chuter de manière significative. En effet, dans une situation réelle, le modèle ne disposera pas de ces informations anticipées et son pouvoir prédictif sera alors bien inférieur à ce qui a pu être observé lors de l'entraînement.

La mise en place d'une base active aide énormément à éviter toute sorte de data leakage puisque tous les clients considérés lors de la préparation des données d'entraînement sont actifs dans l'intervalle [paramètre d'activité, paramètre d'activité + cutoff date]. L'avantage majeur de cette approche est qu'elle exclut automatiquement tous les clients qui ont déjà churné avant la date de coupure. Cela signifie que toutes les informations utilisées pour former le modèle de prédiction sont strictement limitées à des données provenant de clients actifs avant la date de coupure. En d'autres termes, le modèle n'est pas exposé à des informations post-churn pendant sa phase d'entraînement, ce qui pourrait autrement conduire à une sur-optimisation ("overfitting") du modèle sur ces données et à une performance dégradée lors de la généralisation à de nouveaux clients. Ainsi, aucune donnée provenant du futur dont le modèle n'est pas censé savoir n'est prise en considération lors de l'entraînement.

## 3.5 Labels

### 3.5.1 Définition

Les labels, aussi appelés variables cibles ou dépendantes, jouent un rôle fondamental dans l'apprentissage supervisé, qui est une catégorie majeure de l'apprentissage automatique. L'objectif de cette approche est de développer des modèles capables de prédire une sortie spécifique, basée sur une ou

plusieurs entrées. Ces sorties prévues sont justement ce que nous appelons les "labels". Par exemple, dans le cas d'un algorithme de classification, qui est un sous-ensemble de l'apprentissage supervisé, le label pourrait indiquer la catégorie à laquelle appartient une observation particulière. En d'autres termes, il s'agit de la "vérité terrain" que nous cherchons à prédire. Ces labels sont connus et disponibles lors de l'entraînement, mais le but final est que le modèle puisse faire des prédictions précises lorsque ces labels ne sont pas disponibles, c'est-à-dire, sur de nouvelles données non vues auparavant.

Dans le cadre de la prédiction du churn, nous avons affaire à une tâche de classification binaire, ce qui signifie que notre objectif est de prédire l'une des deux classes possibles pour chaque client : churn qui est généralement représenté par un 1, en non churn généralement représenté par un 0. Ces labels sont utilisés pour entraîner le modèle de prédiction du churn. Le modèle cherche à trouver des motifs, des tendances ou des caractéristiques dans les données d'entrée qui sont associées à chaque label. C'est en identifiant ces associations que le modèle est capable de prédire si un client particulier est susceptible de quitter l'opérateur dans le futur. Généralement dans le contexte de la prédiction du churn, on observe souvent un déséquilibre dans la distribution des classes en faveur de la classe non churn, étant donné que naturellement, les clients ont tendance à garder leur opérateur plutôt que de le changer, ce qui fait de la classe non churn la classe majoritaire. Cette situation peut rendre la tâche de prédiction plus difficile car le modèle peut être biaisé en faveur de la classe majoritaire. Il est donc crucial lors de l'entraînement du modèle d'employer des techniques appropriées pour gérer ce déséquilibre des classes tels que :

- **Le suréchantillonnage (Oversampling)** : Cette technique consiste à augmenter la taille de la classe minoritaire en ajoutant des copies de ses instances jusqu'à ce qu'elle atteigne une taille comparable à celle de la classe majoritaire. Cela permet au modèle d'avoir plus d'exemples de la classe minoritaire pour apprendre, ce qui peut améliorer sa performance.
- **Le sous-échantillonnage (Undersampling)** : Au contraire du suréchantillonnage, le sous échantillonnage réduit la taille de la classe majoritaire en supprimant certaines de ses instances. Cependant, cette méthode peut entraîner la perte d'informations importantes si elle n'est pas utilisée avec précaution.
- **La création de poids de classes (Class Weights)** : Cette technique consiste à attribuer des poids plus importants à la classe minoritaire pendant l'entraînement, de sorte que le modèle prête plus d'attention à ces instances.
- **L'utilisation de métriques d'évaluation adaptées** : Plutôt que de se concentrer uniquement sur l'exactitude, qui peut être trompeuse en présence d'un déséquilibre des classes, il peut être plus utile de se concentrer sur d'autres métriques comme la précision, le rappel, le score F1, ou l'aire sous la courbe ROC (AUC-ROC).

### 3.5.2 Comment labelliser un client

La labellisation des clients pour le churn est une tâche délicate qui repose sur un paramètre clé appelé "**paramètre label**". Ce paramètre est exprimé en semaines et est utilisé pour définir une période de temps future [**Cutoff date, Cutoff date + paramètre label**]. Les données contenues dans cet intervalle de temps ne sont pas utilisées pendant l'entraînement du modèle, mais plutôt uniquement utilisées pour la labellisation des clients puisqu'elles proviennent du futur d'un point de vue modélisation.

L'idée est de vérifier le comportement des clients pendant cet intervalle pour les étiqueter soit comme "churn" ou "non churn".

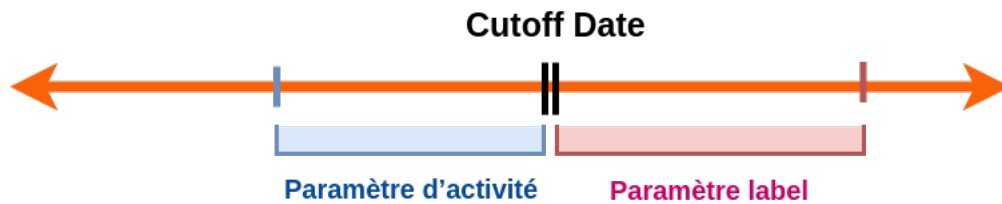


FIGURE 3.4 : Diagramme paramètre label

Le calcul des labels peut se faire en fonction de plusieurs critères tels que les appels émis ou reçus, la consommation de données, ou les recharges effectuées par les clients. Puisque les données des clients sont des données temporelles sous formes de logs, nous disposons de plusieurs options pour agréger ces informations afin de synthétiser et résumer les actions de chaque client pendant l'intervalle de labellisation. Les deux approches courantes qui sont utilisées sont la "**somme**" et le "**count**".

La "**somme**" consiste à additionner certaines valeurs associées aux activités d'un client dans une période donnée. Par exemple, si nous considérons le critère des recharges, nous pouvons additionner le montant total des recharges effectuées par un client pendant la période de labellisation. Cela nous donne une idée de l'intensité de l'activité du client pendant cette période, et par suite une affectation des labels orientée revenus.

Le "**count**" quant à lui se concentre sur le nombre d'activités réalisées, indépendamment de leur valeur. Dans le cas des recharges, cela pourrait signifier compter le nombre total de recharges effectuées par un client, sans se soucier du montant de chaque recharge. Cela nous donne une idée de la fréquence des activités du client.

Une fois l'agrégation effectuée, nous déterminons un seuil qui servira à attribuer les labels. Si le résultat de l'agrégation (somme ou count) pour un client est supérieur à ce seuil, le client est considéré comme non churn et reçoit le label 0. Cela signifie que le client a maintenu un certain niveau d'activité pendant la période de labellisation. En revanche, si le résultat de l'agrégation est inférieur ou égal au seuil, le client est considéré comme ayant churné et reçoit le label 1. Cela indique que le client a réduit ou arrêté son activité en dessous d'un certain niveau, ce qui est interprété comme un signe de désengagement ou de churn.

Prenons un exemple concret pour illustrer ce processus de labellisation. Supposons que nous fixons notre cutoff date au **1er janvier 2023**. Ensuite, nous définissons un paramètre de label de **12 semaines**. C'est la période future après la date de coupure pendant laquelle nous observerons l'activité des clients pour déterminer s'ils ont churné ou non. Ceci nous donne l'intervalle suivant **[1 janvier 2023, 1 avril 2023]**. En utilisant la "**somme**" comme type d'agrégation et les recharges comme critère, nous additionnons tous les montants de **recharges** effectuées par chaque client pendant cette période de 12 semaines. Nous fixons ensuite un seuil à **10**. Ce seuil est la ligne de démarcation qui nous permet de distinguer les clients qui ont churné de ceux qui n'ont pas churné. Ainsi, les clients dont la somme des montants de recharges pendant cet intervalle de temps est supérieure strictement à 10 sont considérés comme étant toujours engagés avec l'opérateur, et par suite sont labellisés comme non churn, ce qui est représenté par le label 0. D'autre part, les clients dont la somme des montants de recharges est inférieure ou égale à 10 sont considérés comme ayant réduit leur engagement avec

l'opérateur à un niveau jugé insuffisant, ils sont donc labellisés comme churn, ce qui est représenté par le label 1.

On applique ceci sur le tableau suivant :

MDN	MONTANT_RECHARGE	DATE_RECHARGE
MDN01	50	19 janvier 2023
MDN01	50	04 février 2023
MDN01	100	22 mars 2023
<del>MDN02</del>	<del>100</del>	<del>10 septembre 2022</del>
<del>MDN02</del>	<del>20</del>	<del>16 novembre 2022</del>
MDN02	5	22 mars 2023
<del>MDN03</del>	<del>100</del>	<del>06 décembre 2022</del>
MDN03	50	24 janvier 2023
MDN03	100	03 mars 2023

TABLE 3.1 : Logs de recharges de 3 MDNs

Les lignes barées ne sont pas prises en considération puisque leurs dates de recharge n'appartiennent pas à l'intervalle de temps [1 janvier 2023, 1 avril 2023]

$$\text{MDN01 : } \sum \text{MONTANT\_RECHARGE} = 50 + 50 + 100 = 200 > 10 \quad \Rightarrow \text{non churn}$$

$$\text{MDN02 : } \sum \text{MONTANT\_RECHARGE} = 5 < 10 \quad \Rightarrow \text{churn}$$

$$\text{MDN03 : } \sum \text{MONTANT\_RECHARGE} = 50 + 100 = 150 > 10 \quad \Rightarrow \text{non churn}$$

Poursuite on aura les labels suivant :

MDN	LABELS
MDN01	0
MDN02	1
MDN03	0

TABLE 3.2 : MDNs labellisés après calcul (0 : non churn, 1 : churn)

### 3.6 Historique des données

Lorsqu'il s'agit de construire un modèle prédictif à partir de données historiques, il est souvent essentiel d'équilibrer la quantité d'information utilisée avec la pertinence et l'actualité de ces données. En effet, toutes les données disponibles ne sont pas forcément utiles ou pertinentes pour la prédiction que nous essayons de faire. C'est pourquoi il est parfois préférable de n'utiliser qu'une partie de ces données, sélectionnées en fonction de leur pertinence et de leur actualité.

Pour ce faire, on introduit un paramètre appelé "**paramètre historique**". Ce paramètre, généralement exprimé en semaines, définit la fenêtre temporelle des données à utiliser pour l'entraînement du modèle. En pratique, le paramètre historique crée un intervalle de temps qui s'étend de la cutoff date moins le paramètre historique jusqu'à la cutoff date [**cutoff date - paramètre historique, cutoff**

**date]**. Il est essentiel de mentionner que cet intervalle ne comprend que les données des clients qui font partie de la base active, c'est-à-dire des clients qui ont été actifs pendant l'intervalle défini par le paramètre d'activité. Ainsi, au lieu d'utiliser toutes les données historiques disponibles pour chaque client de la base active, nous n'utilisons que les données qui se situent dans l'intervalle de temps défini par le paramètre historique. Cette approche permet d'ajuster la quantité d'information que le modèle prend en compte lors de l'apprentissage, en se concentrant sur les informations les plus pertinentes et les plus récentes.

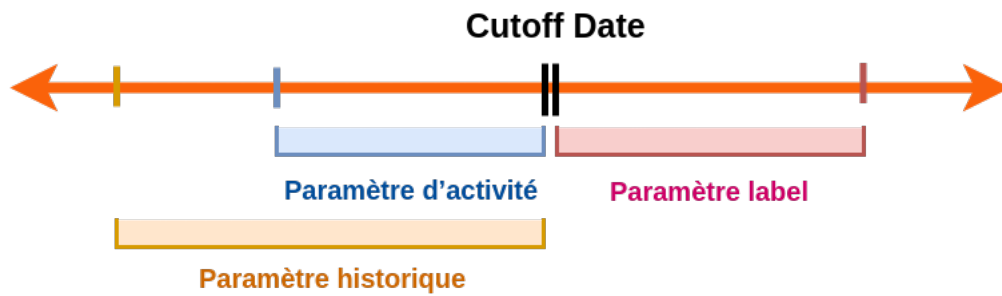


FIGURE 3.5 : Diagramme paramètre historique

La mise en œuvre de cette stratégie permet d'éviter le risque d'introduire du bruit ou des informations potentiellement trompeuses dans le modèle. Par exemple, si un client a significativement changé son comportement au fil du temps, les données les plus anciennes pourraient ne plus être représentatives de son comportement actuel et pourraient donc induire le modèle en erreur. En se concentrant sur une période plus récente, nous nous assurons que le modèle est formé sur les données qui sont les plus susceptibles de refléter l'état actuel des choses.



# Implémentation de la solution

## Introduction

Dans le présent chapitre, nous allons nous pencher sur les détails techniques de l'implémentation de notre système de génération de données d'entraînement. Nous dévoilerons les coulisses du fonctionnement du système ainsi que les défis que nous avons rencontrés et surmontés.

### 4.1 Données utilisées

Lors du développement du système de génération automatique des agrégats pour les modèles de prédiction du churn, j'ai dû faire face à un certain nombre de contraintes liées à la confidentialité des données. Étant donné que les données de l'entreprise sont sensibles, je n'ai pas eu le droit d'accéder directement aux bases de données de l'opérateur. Au lieu de cela, j'ai reçu deux fichiers csv contenant les données nécessaires à mon travail.

Le premier fichier contient des données extraites de la table des **recharges** (On verra en détails le contenu de cette table dans le chapitre 5 de l'analyse exploratoire des données et l'entraînement du modèle). Le deuxième fichier csv contient une table statique contenant les MDN et leurs dates CPR1.

Afin de préserver l'anonymat des clients, j'ai procédé à un mappage des MDNs pour les transformer en une version anonyme. Par exemple, le MDN **0606060606** est devenu **MDN01**.

Enfin, j'ai importé l'ensemble de ces données dans un serveur local **PHPMyAdmin MySQL** pour simuler une base de données. Cela m'a permis de travailler de manière sécurisée et confidentielle, tout en disposant de toutes les données nécessaires pour développer le système de génération automatique des agrégats pour les modèles de prédiction du churn.

### 4.2 Outils

Le premier outil que j'ai utilisé pour ce projet était **Python**, plus spécifiquement la version **3.7.16**. Ce langage de programmation, réputé pour sa manipulation et son analyse de données, est favorisé en raison de sa syntaxe claire et intuitive qui le rend accessible aux débutants tout en étant puissant



pour les développeurs expérimentés. Il offre une gamme étendue de packages, notamment Pandas, NumPy et scikit-learn, qui sont essentiels pour le travail de la data science. Ces outils fournissent des capacités allant de la manipulation de données à l'apprentissage automatique.

Pour le développement du projet, j'ai créé un environnement virtuel Conda pour isoler l'espace de travail du projet et ses dépendances, en évitant ainsi les conflits de versions de packages et en garantissant que les applications utilisent la bonne version des packages installés. Ainsi, cela a permis une gestion plus précise des packages Python et une meilleure reproductibilité sur les machines de mes collègues.

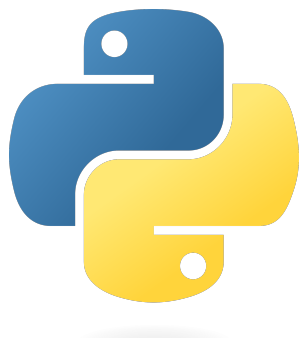


FIGURE 4.1 : logo du langage de programmation Python

Au cours du processus de développement, j'ai effectué de nombreux tests dans des **notebooks Jupyter**. Les notebooks Jupyter sont des outils interactifs qui permettent d'écrire et d'exécuter du code Python, de le documenter et de visualiser les résultats, tout cela dans un seul et même environnement. Ils sont très utilisés dans le domaine de la data science pour le prototypage rapide, l'expérimentation de modèles et l'exploration de données.



FIGURE 4.2 : logo de Jupyter Notebook

Par ailleurs, **MySQL** a joué un rôle crucial dans mon travail. Celui-ci est un système de gestion de bases de données relationnelles open source largement utilisé pour le stockage de données. Dans mon cas, j'ai utilisé MySQL pour stocker les données des recharges et les informations des clients de l'opérateur simulées localement avec un serveur PHPMyAdmin, ainsi que pour tester le bon fonctionnement du système de génération des données d'entraînements. L'utilisation d'une base de données MySQL a servi d'un premier pas avant de migrer vers un environnement Big Data en HDFS avec Impala comme moteur de requêtes SQL.



FIGURE 4.3 : logo de MySQL

## 4.3 Fichier config

L'un des principaux composants de ce projet est le **fichier de configuration**. Il s'agit d'un fichier de format **JSON** (JavaScript Object Notation) qui est un format largement utilisé pour stocker et échanger des données. Ce fichier contient l'ensemble des paramètres métiers, ainsi que les différentes tables et leurs colonnes que l'utilisateur souhaite prendre en compte pour la génération des données d'entraînement. Le fichier de configuration offre un moyen pratique et flexible pour l'utilisateur de définir les spécificités.

Une fois le fichier de configuration rempli, l'utilisateur peut lancer le script Python via le terminal en passant le fichier de configuration en argument. La commande serait donc de la forme suivante : `"python auto-inwi.py config.json"`. Le script lira alors les paramètres du fichier de configuration et générera les données d'entraînement en conséquence.

Nous explorerons plus en détail les différentes sections du fichier de configuration dans les sections suivantes.

## 4.4 Génération automatique de la base actif et des labels

### 4.4.1 Introduction

Le fichier de configuration est organisé en plusieurs sections, chacune correspondant à un aspect spécifique de l'agrégation des données. La première partie du fichier de configuration est consacrée à la définition de la **cutoff date** et aux paramètres concernant la **base active**, notamment la génération de celle-ci et la création des labels. Dans la section suivante, nous allons explorer en détail les différents paramètres que l'utilisateur peut définir dans cette première partie du fichier de configuration, et comment ceux-ci influencent la génération des données d'entraînement.

```
"cutoff_date": "2022-06-15 00:00:00",
"base_active": {
  "generate": 1,
  "active": {
    "static_table": "static_MDN",
    "static_key": "MDN",
    "CPR1_column": "date_CPR1",
    "tenure": "12",
    "active_table": "sample_rech",
    "active_key": "MDN",
    "active_column": "MONTANT_RECHARGE",
    "active_date_column": "ID_DATE",
    "agg": "count",
    "threshold": "1",
    "time_window": "10"
  },
  "labels": {
    "table": "sample_rech",
    "column": "MONTANT_RECHARGE",
    "agg": "count",
    "threshold": "0",
    "time_window": "8"
  }
}
```

FIGURE 4.4 : Première partie du fichier de configuration

#### 4.4.2 Paramètres du fichier de configuration

Le fichier de configuration contient plusieurs paramètres essentiels pour le processus d'agrégation des données et la création de la base active. Pour commencer, le paramètre **"cutoff date"** représente la date de coupure que nous avons déjà discutée et est présentée sous forme de timestamp.

Dans la section **"base\_active"**, plusieurs paramètres clés sont définis :

- **"generate"** : Ce paramètre peut prendre deux valeurs. Si "generate" est à 0, l'utilisateur n'a pas besoin de générer la base active et les labels, et peut donc utiliser une base active préexistante qu'il souhaite utiliser. Dans ce cas, la section "base\_active" du fichier de configuration n'est pas prise en compte. Si "generate" est à 1, le script générera la base active et ses labels.

Pour la partie de génération de la base active :

- **"static\_table"** : Il s'agit généralement d'une table qui contient des informations statiques sur les MDNs, telles que le numéro de téléphone et la date de la CPR1.
- **"static\_key"** : Il s'agit de la clé primaire de la table statique, généralement le MDN du client dans le contexte prepaid.
- **"CPR1\_column"** : Ce paramètre indique la colonne dans la table statique qui contient la date de la CPR1. Si l'utilisateur ne spécifie pas ce paramètre, le critère de l'ancienneté ne sera pas pris en compte lors de la génération de la base active.
- **"tenure"** : Ce paramètre indique le nombre de semaines que doit avoir un client pour être considéré comme ancien.

- **"active\_table"** : Il s'agit de la table à prendre en compte lors de la définition de l'activité des clients.
- **"active\_key"** : Il s'agit de la clé primaire de la table d'activité, généralement le MDN.
- **"active\_column"** : Il s'agit de la colonne de la table d'activité sur laquelle on veut calculer l'activité des clients.
- **"active\_date\_column"** : Il s'agit de la colonne date de la table d'activité.
- **"agg"** : Ce paramètre indique le type d'agrégation à utiliser pour générer la base active ("sum" ou "count").
- **"threshold"** : Il s'agit du seuil que doit dépasser un client pour être considéré comme actif.
- **"time\_window"** : Il s'agit du paramètre d'activité, qui est un nombre de semaines qui indique la période que l'on veut prendre comme intervalle d'activité [cutoff date - paramètre d'activité , cutoff date].

Pour la partie de génération des labels, on a les paramètres suivants :

- **"table"** : Il s'agit de la table à prendre en considération pour la génération des labels. Elle peut être la même que la table d'activité ou une autre table.
- **"column"** : Il s'agit de la colonne de cette table sur laquelle on souhaite calculer les labels.
- **"agg"** : Ce paramètre indique le type d'agrégation à utiliser pour le calcul des labels ("sum" ou "count").
- **"threshold"** : Il s'agit du seuil que doit dépasser un client actif pour obtenir un label 0 (non churn) ou 1 (churn)
- **"time\_window"** : Il s'agit du paramètre de label, qui est un nombre de semaines qui définit l'intervalle de labellisation [cutoff date, cutoff date + paramètre de label].

#### 4.4.3 Code

Une fois que les paramètres sont correctement définis dans le fichier de configuration JSON, le système récupère ces informations pour les utiliser en tant qu'entrées dans des fonctions spécifiques, notamment "def base\_active" et "def label". Ces fonctions génèrent des requêtes SQL dynamiques en concaténant les paramètres spécifiés dans le fichier de configuration. Les requêtes SQL résultantes sont sauvegardées dans des fichiers .sql, qui sont ensuite exécutés automatiquement par le système. Cela conduit à la création des tables requises, y compris la base active et la base active labellisée. Toutes ces fonctions sont écrites dans leurs propres fichiers et regroupées sous forme d'une bibliothèque, qui est ensuite importée dans le fichier principal pour être utilisée lors de l'exécution du script.

Le premier fichier sql généré concerne la génération de la base active. Tout d'abord, on identifie les clients qui ont un total (somme ou nombre) de la colonne définie pour l'activité (par exemple, le montant des recharges) supérieur au seuil défini. Cela est effectué sur un intervalle de temps défini par [cutoff date - paramètre d'activité , cutoff date]. Ce calcul est effectué sur la table spécifiée pour l'activité (par exemple, la table des recharges).

En parallèle, on identifie les clients dont la durée d'ancienneté (calculée en soustrayant la date CPR1 à la date de coupure) est strictement supérieure à un seuil prédéfini. Ce calcul est effectué sur la table statique contenant la date CPR1.

Ensuite, ces deux ensembles de clients sont combinés en utilisant une opération "inner join", de sorte que seuls les clients qui remplissent les deux conditions sont conservés. Le résultat de cette opération est la base active

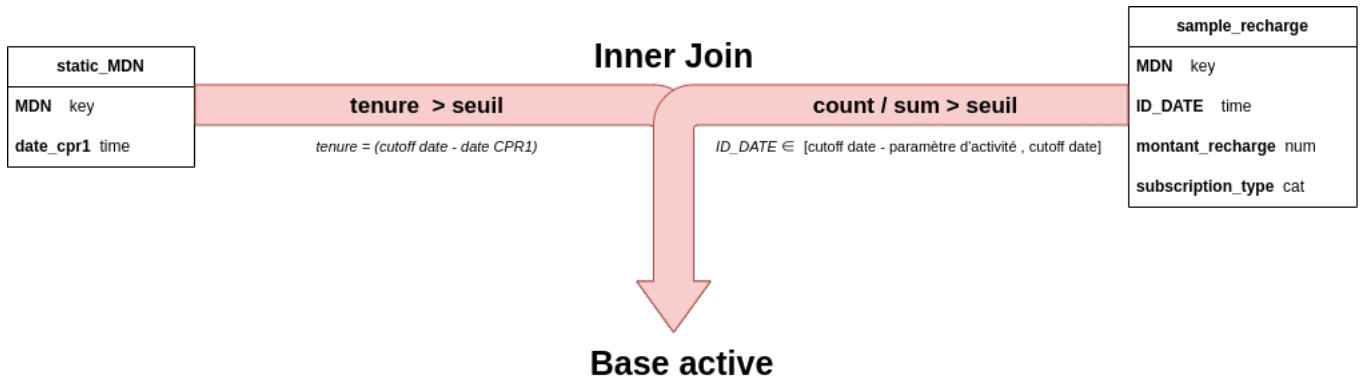


FIGURE 4.5 : Diagramme des étapes de la génération de la base active

```
DROP TABLE if EXISTS base_active;
CREATE TABLE base_active as
SELECT sample_rech.MDN
FROM sample_rech
JOIN static_MDN ON sample_rech.MDN = static_MDN.MDN
WHERE sample_rech.ID_DATE BETWEEN DATE_SUB('2022-06-15 00:00:00', INTERVAL 10 WEEK) AND '2022-06-15 00:00:00'
AND static_MDN.date_CPR1 < DATE_SUB('2022-06-15 00:00:00', INTERVAL 12 WEEK)
GROUP BY sample_rech.MDN
HAVING COUNT(*) > 1;
```

FIGURE 4.6 : Exemple du fichier sql pour la génération de la base active généré automatiquement par le système selon les paramètres du fichier de configuration

La génération des labels est effectuée par une autre requête SQL qui s'appuie sur deux tables : la table que l'on a choisie pour calculer les labels (par exemple la table des recharges) et la table de la base active qui a été générée par la requête précédente.

Dans un premier temps, on calcule le total (somme ou nombre) de la colonne choisie pour le calcul des labels (par exemple le montant des recharges) pour chaque client (group by MDN). Si ce total est strictement supérieur au seuil défini, le client se voit attribuer un label 0. Dans le cas contraire, le client se voit attribuer un label 1.

Ensuite, un "left join" est effectué entre cette nouvelle table de labels et la table de la base active. Cela permet de ne conserver que les labels correspondant aux clients qui figurent dans la base active. Le résultat de cette opération est une base active labellisée.

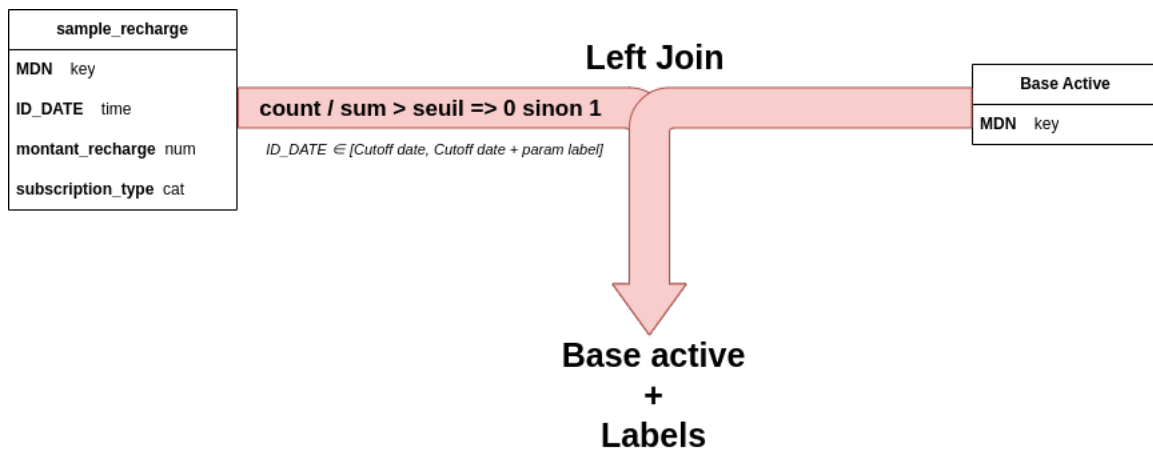


FIGURE 4.7 : Diagramme des étapes de la génération des labels

```

DROP TABLE IF EXISTS base_active_labels;
CREATE TABLE base_active_labels AS
SELECT base_active.MDN,
CASE WHEN EXISTS (
  SELECT 1
  FROM (
    SELECT MDN
    FROM sample_rech
    WHERE ID_DATE >= '2022-06-15 00:00:00'
    AND ID_DATE < DATE_ADD('2022-06-15 00:00:00', INTERVAL 8 WEEK)
    GROUP BY MDN
    HAVING COUNT(*) > 0
  ) s
  WHERE s.MDN = base_active.MDN
) THEN 0 ELSE 1 END AS label
FROM base_active;

```

FIGURE 4.8 : Exemple du fichier sql pour la génération des labels généré automatiquement par le système selon les paramètres du fichier de configurations

#### 4.4.4 Résultats

Les deux dernières requêtes s'exécutent d'une manière automatique l'une après l'autre et créent les deux tables suivantes :

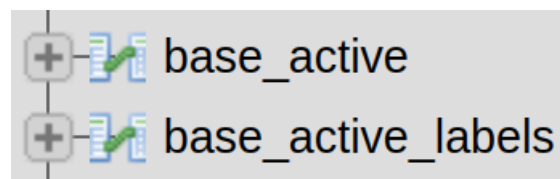


FIGURE 4.9 : Deux tables générées automatiquement : base\_active et base\_active\_labels

MDN
MDN01
MDN03
MDN04
MDN09
MDN13
MDN14
MDN20
MDN38
MDN39
MDN44
MDN62
MDN68
MDN71
MDN74
MDN84
MDN86
MDN91
MDN94

FIGURE 4.10 : Première page de la table base\_active

MDN	label
MDN01	1
MDN03	0
MDN04	1
MDN09	1
MDN13	0
MDN14	0
MDN20	0
MDN38	0
MDN39	0
MDN44	0
MDN62	0
MDN68	0
MDN71	0
MDN74	0
MDN84	0
MDN86	0
MDN91	0
MDN94	1

FIGURE 4.11 : Première page de la table base\_active\_labels

Après avoir généré la table de la base active labélisée, un **graphique de distribution des labels** est automatiquement créé pour visualiser la répartition des deux labels avant de commencer à générer les agrégats. Comme mentionné précédemment, les labels ne sont généralement pas équilibrés et sont en faveur du label non-churn, c'est-à-dire le label 0. Si la distribution des labels montre un équilibre différent, c'est un indicateur que quelque chose pourrait ne pas fonctionner correctement. Dans ce cas, il est recommandé de revoir les paramètres entrés dans le fichier de configuration pour s'assurer qu'ils reflètent bien l'activité attendue des clients et qu'ils ne sont pas à l'origine d'une distribution des labels erronée.

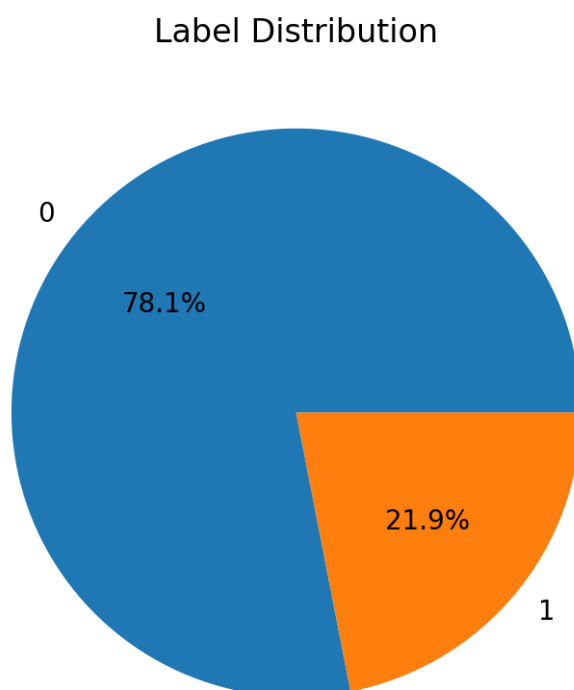


FIGURE 4.12 : Graphe de distribution des labels généré automatiquement

## 4.5 Génération des agrégations

### 4.5.1 définition

Dans le contexte de l'apprentissage automatique, surtout lorsqu'on travaille avec des données sous forme de logs, l'agrégation des données est une étape essentielle. Les logs sont des enregistrements détaillés de chaque action ou événement survenu sur une période donnée. Cependant, pour le machine learning, il est souvent plus utile de résumer ces informations en caractéristiques plus générales ou agrégées, car les modèles d'apprentissage automatique s'appuient sur des données sous formes d'entités pour tirer des conclusions et faire des prédictions.

Une agrégation signifie rassembler les données en fonction de certaines clés (par exemple MDN) et effectuer un calcul récapitulatif, tel que la somme, la moyenne, le maximum, le minimum ou le compte sur ces groupes de données.

Ceci permet de générer donc de nouvelles colonnes qui résument les données existantes, permettant ainsi de découvrir de nouveaux insights et fournissant des informations plus riches au modèle d'apprentissage automatique, en révélant de ce fait des tendances dans les données qui étaient autrefois invisibles.

### 4.5.2 Concept

La génération des agrégations est un processus qui comprend plusieurs étapes. Tout d'abord, il est essentiel de reconnaître que la nature de l'agrégation dépend fortement du type de la colonne concernée. Le processus général commence par le calcul de l'agrégation de la colonne en question sur une



période donnée délimitée par l'intervalle [cutoff date - paramètre historique,cutoff date]. Cet intervalle assure que l'historique des données reste limité, comme expliqué précédemment.

Après cette opération, les données sont regroupées par la clé primaire (souvent MDN) de la table concernée, transformant les logs en entités distinctes. Chaque agrégation est ensuite créée sous forme de table temporaire. Enfin, une table finale contenant les données d'entraînement est créée en effectuant des 'left join' de ces tables temporaires à la table de la base active labellisée. Cela permet de ne conserver que les agrégations relatives aux clients actifs.

Le script principal parcourt chaque colonne indiquée dans le fichier de configuration et effectue des agrégations en fonction du type de la colonne. Un ensemble de fonctions, écrites dans leur propre bibliothèque, prend en entrée les paramètres du fichier de configuration, réalise des concaténations et génère le dernier fichier SQL qui produit la table contenant les données d'entraînement.

Pour minimiser le coût des 'left joins', le nombre de tables temporaires a été réduit en combinant toutes les agrégations qui peuvent être sélectionnées simultanément dans leur propre table temporaire. Ce processus implique le retour des fonctions de génération d'agrégation non pas sous forme de chaînes de caractères, mais sous forme de paires clé-valeur (où la clé est le SELECT et la valeur est la clause WHERE). Ces paires sont ensuite combinées dans un dictionnaire, qui est transformé de manière à combiner toutes les clés ayant la même valeur dans leur propre table temporaire. Cela réduit de manière significative le nombre de jointures à gauche.

En fin de compte, ce processus aboutit à la création d'un fichier SQL contenant un ensemble de tables temporaires représentant les différentes agrégations, ainsi que des 'left join' avec la base active labellisée. L'exécution de cette requête SQL génère alors la table finale contenant les données d'entraînement pour le modèle.

```
CREATE TEMPORARY TABLE temp14 AS
SELECT
MDN AS MDN14,

COUNT(num_total) AS 'count_num_total_canal:Recharge GAB_last_8_weeks',
MIN(num_total) AS 'min_num_total_canal:Recharge GAB_last_8_weeks',
MAX(num_total) AS 'max_num_total_canal:Recharge GAB_last_8_weeks',
AVG(num_total) AS 'avg_num_total_canal:Recharge GAB_last_8_weeks',
SUM(num_total) AS 'sum_num_total_canal:Recharge GAB_last_8_weeks',
COUNT(MONTANT_RECHARGE) AS 'count_MONTANT_RECHARGE_canal:Recharge GAB_last_8_weeks',
MIN(MONTANT_RECHARGE) AS 'min_MONTANT_RECHARGE_canal:Recharge GAB_last_8_weeks',
MAX(MONTANT_RECHARGE) AS 'max_MONTANT_RECHARGE_canal:Recharge GAB_last_8_weeks',
AVG(MONTANT_RECHARGE) AS 'avg_MONTANT_RECHARGE_canal:Recharge GAB_last_8_weeks',
SUM(MONTANT_RECHARGE) AS 'sum_MONTANT_RECHARGE_canal:Recharge GAB_last_8_weeks'
FROM sample_rech
WHERE canal="Recharge GAB" AND id_date BETWEEN '2022-08-31 00:00:00' - INTERVAL 8 WEEK AND '2022-08-31 00:00:00'
AND
id_date BETWEEN '2022-08-31 00:00:00' - INTERVAL 12 WEEK AND '2022-08-31 00:00:00'
GROUP BY MDN;
```

FIGURE 4.13 : Exemple d'un table temporaire du fichier sql finale

De plus, chaque agrégation est accompagnée d'un alias automatique pour aider à organiser de manière claire et efficace les colonnes finales des données d'entraînement. Cette pratique permet de rendre le sens et la fonction de chaque agrégation facilement compréhensibles en ajoutant un niveau de lisibilité et de transparence aux données.

Par exemple, on peut aisément comprendre que la colonne **max\_MONTANT\_RECHARGE\_canal:Recharge GAB\_last\_8\_weeks** veut dire qu'il s'agit du maximum du montant des recharges lorsqu'il s'agit du canal recharge par GAB dans les dernières 8 semaines.

### 4.5.3 Paramètres du fichier de configuration

La deuxième partie du fichier de configuration est divisée en deux sous-sections : 'params' et 'tables'.

```
"params": {
  "categorical_columns_threshold": 5,
  "param_history": "24",
  "time_windows": [
    1,
    2,
    3,
    4
  ],
  "main_table": "base_active_labels"
},
"tables": {
  "base_active_labels": {
    "MDN": "key"
  },
  "sample_rech": {
    "MDN": "key",
    "ID_DATE": "time",
    "MONTANT_RECHARGE": "num",
    "NB_EVENT": "num",
    "TYPE_RECH": "cat",
    "SUBSCRIPTION_TYPE": "cat"
  }
}
```

FIGURE 4.14 : Deuxième partie du fichier de configuration

'**params**' contient des paramètres supplémentaires qui peuvent affecter la manière dont les données sont traitées :

- '**categorical\_columns\_threshold**' : Ce paramètre est utilisé lors du traitement des colonnes numériques avec une clause 'where' pour les valeurs distinctes des colonnes catégorielles. Les valeurs uniques dans une colonne catégorielle qui apparaissent moins souvent que le seuil spécifié sont regroupées dans une catégorie séparée appelée 'other'. Cela facilite la gestion des valeurs moins fréquentes et évite la surcharge des modèles avec un trop grand nombre de catégories.
- '**param\_history**' : C'est le paramètre d'historique, qui indique la période de temps pour laquelle les données historiques sont considérées lors de la création des agrégations.
- '**time\_windows**' : Ce sont les fenêtres temporelles en semaines pour lesquelles les agrégations sont calculées. Il s'agit d'une liste de nombres, chacun représentant une fenêtre temporelle différente.
- '**main\_table**' : Il s'agit de la table contenant la base active labellisée.

La section '**tables**' contient des informations sur les tables dont les agrégations doivent être générées. Chaque table contient les colonnes à prendre en considération, ainsi que leur type :

- '**key**' : Clé principale de la table.
- '**num**' : Colonne numérique.
- '**cat**' : Colonne catégorielle.
- '**time**' : Colonne timestamp qui indique le moment où chaque enregistrement a été créé.

#### 4.5.4 Explication de chaque agrégation et de sa fonction

La génération des agrégations dépend de la nature de la colonne : **numérique** ou **catégorielle**.

Pour les colonnes **numériques**, trois types d'agrégations sont utilisés :

1. **Les agrégations globales** : Cela inclut le minimum (min), le maximum (max), la moyenne (avg), la somme (sum) et le nombre (count) des valeurs dans la colonne. Chacune de ces agrégations donne un aperçu différent des données :
  - **Min** et **Max** permettent d'identifier l'étendue des valeurs.
  - **Avg** fournit la valeur moyenne, offrant une mesure centrale des données.
  - **Sum** donne la valeur totale, qui peut être utile pour comprendre l'ampleur globale des données.
  - **Count** fournit le nombre total d'observations, ce qui peut aider à comprendre la taille de l'ensemble de données.
2. **Agrégation globale + filtre temporel** : Il s'agit des mêmes agrégations globales, mais appliquées à des fenêtres de temps spécifiques, comme spécifié dans le fichier de configuration (par exemple, le minimum d'une colonne numérique au cours des deux dernières semaines). Ces agrégations sont utiles car elles permettent de comprendre comment les valeurs ont changé au fil du temps, ce qui peut aider à identifier des tendances ou des modèles saisonniers.
3. **Agrégations globales avec une clause WHERE pour chaque valeur distincte des colonnes catégorielles** : Ces agrégations permettent de comprendre les valeurs numériques en relation avec les catégories. Par exemple, on peut calculer la somme ou la moyenne d'une colonne numérique pour chaque catégorie d'une colonne catégorielle. Ceci est utile pour identifier les différences entre les catégories. Pour limiter le nombre de colonnes générées, les valeurs distinctes dont la moyenne d'occurrence est inférieure au seuil défini dans le fichier de configuration sont regroupées dans leur propre catégorie appelée 'other'. Ce regroupement permet de gérer efficacement les catégories moins fréquentes et de réduire la dimensionnalité des données.

Pour les colonnes **catégorielles**, les types d'agrégations utilisés sont les suivants :

1. **Count distinct** : Cette agrégation est utile pour comprendre la variété des valeurs dans une colonne. En comptant le nombre distinct de catégories pour chaque clé primaire, on peut avoir une idée du degré de variabilité ou de diversité de la colonne.
2. **Le mode** : Cette agrégation permet d'identifier la catégorie la plus fréquente et dominante pour chaque clé primaire.
3. **Normalized Count** : Il s'agit d'une méthode plus avancée d'agrégation catégorielle. Ici, le nombre de records pour chaque valeur unique est divisé par le nombre total de records pour chaque clé primaire. Ensuite, on pivote les colonnes de manière à avoir le compte normalisé de chaque valeur distincte comme sa propre colonne. Cette méthode permet de capturer à la fois l'importance relative de chaque catégorie (par la normalisation) et la présence ou l'absence de chaque catégorie (par le pivotement). Cette agrégation peut aider à révéler des modèles subtils dans les données catégorielles qui pourraient être manqués par d'autres méthodes d'agrégation.

### 4.5.5 Résultats

Le produit final de ce processus est un fichier .sql qui s'exécute automatiquement, générant la table 'train\_data'. Cette table contient les diverses agrégations calculées et les étiquettes associées à chaque MDN. Nous obtenons à la fin un ensemble de données organisé, comprenant les entités d'entrée **X** et les cibles **y**, prêt à être utilisé pour l'entraînement de notre modèle.

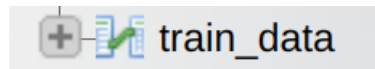


FIGURE 4.15 : Table train\_data

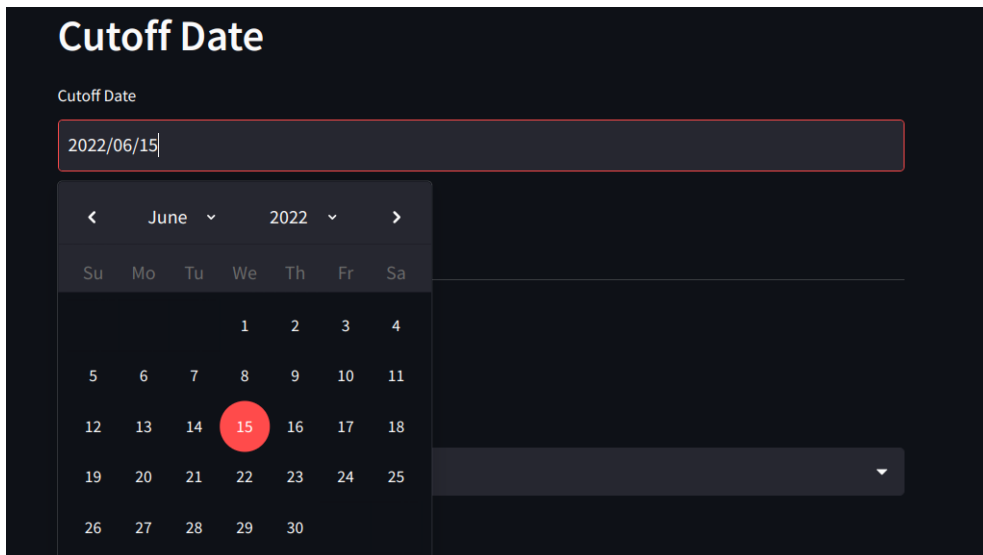
MDN	label	min_MONTANT_RECHARGE	max_MONTANT_RECHARGE	avg_MONTANT_RECHARGE	sum_MONTANT_RECHARGE	count_MONTANT_RECHARGE
MDN01	1	5	200	78.33333333333333	470	6
MDN03	0	5	200	93	465	5
MDN04	0	5	80	38.333333333333336	230	6
MDN06	0	15	150	68.33333333333333	205	3
MDN13	0	10	200	102	510	5
MDN14	0	40	80	60	120	2
MDN18	1	25	40	35	105	3
MDN30	0	10	100	55	220	4
MDN31	0	10	20	15	30	2
MDN38	0	20	100	60	120	2
MDN39	0	40	200	93.33333333333333	280	3
MDN44	0	10	25	16.666666666666668	50	3
MDN53	0	5	100	55	275	5

FIGURE 4.16 : Partie de la table train\_data

Il est crucial de souligner que le nombre de colonnes générées par ce processus peut être très élevé, ce qui peut conduire à un phénomène appelé '**la malédiction de la dimensionnalité**'. Ce terme fait référence aux divers problèmes rencontrés lors de la manipulation ou de l'analyse de données de très haute dimension, notamment les problèmes liés à l'efficacité, la pertinence des données, et l'overfitting du modèle. Ainsi, avant de commencer l'entraînement du modèle, il est souvent recommandé d'appliquer une technique de réduction de la dimensionnalité afin de diminuer le nombre de variables tout en conservant le maximum d'information utile.

## 4.6 Interface graphique

Pour faciliter encore plus le processus de configuration, on a développé une interface graphique conviviale grâce à **Streamlit**, une bibliothèque Python open source qui permet aux développeurs et aux data scientists de créer rapidement des applications web pour leurs projets de machine learning et de data science. Cette interface utilisateur sert à remplir le fichier de configuration, ce qui rend le processus beaucoup plus intuitif et moins sujet à des erreurs de syntaxe ou de formatage.



The screenshot shows a web interface titled "Cutoff Date". Below the title, there is a text input field containing "2022/06/15". Below this field is a calendar widget for June 2022. The calendar has a header with navigation arrows, the month "June", and the year "2022". The days of the week are listed as "Su", "Mo", "Tu", "We", "Th", "Fr", and "Sa". The dates are arranged in a grid. The date "15" is highlighted with a red circle. To the right of the calendar is a dark sidebar area.

FIGURE 4.17 : Choix de la date cutoff par interface graphique



The screenshot shows a web interface titled "Labels". It contains several configuration sections:

- Labels Table:** A text input field containing "sample\_rech".
- Labels Column:** A text input field containing "MONTANT\_RECHARGE".
- Labels Aggregation (count or sum):** A dropdown menu with "count" selected. Below the dropdown are two buttons: "count" and "sum".
- Labels Time Window (Weeks):** A text input field containing "8", with minus and plus buttons on the right.
- Distribution Churn Label:** A button at the bottom.

FIGURE 4.18 : Remplissage des paramètres de labellisation par interface graphique et bouton de visualisation de la distribution des classes

The 'Params' interface is a dark-themed form with the following sections:

- Categorical Columns Threshold:** A numeric input field containing the value '10' with minus and plus icons for adjustment.
- Param History (Weeks):** A numeric input field containing the value '2'.
- Time Windows (Weeks):** A row of five red buttons labeled '1 x', '2 x', '3 x', '4 x', and '8 x', followed by a clear icon and a dropdown arrow.
- Main Table (base active):** A text input field containing 'base\_active\_labels'. A tooltip message 'Nom par défaut de la table base active' is visible next to the field.

FIGURE 4.19 : Remplissage des paramètres additionnels par interface graphique

The 'Table 2' interface is a dark-themed form with the following sections:

- Table 2 Name:** A text input field containing 'sample\_rech'.
- Number of columns for Table 2:** A numeric input field containing the value '2' with minus and plus icons.
- Column 1 Name for Table 2:** A text input field containing 'MDN'.
- Column 1 Type for Table 2:** A dropdown menu with 'key' selected.
- Column 2 Name for Table 2:** A text input field containing 'MONTANT\_RECHAGE'.
- Column 2 Type for Table 2:** A dropdown menu with 'num' selected.
- Generate training data:** A button located at the bottom of the form.

FIGURE 4.20 : Choix des tables et des colonnes à prendre en considération lors de la génération des aggregations par interface graphique et bouton de génération données d’entrainements

# EDA et Entraînement du modèle de prédiction de churn

## Introduction

Afin de mettre en œuvre et de démontrer l'efficacité de notre système de génération de données d'entraînement, nous allons comparer les performances de deux modèles. Le premier modèle sera entraîné sur un jeu de données dont les agrégations ont été calculées manuellement. Le deuxième modèle en revanche utilisera des données d'entraînement générées par notre système automatisé de génération d'agrégations. En comparant les performances de ces deux modèles, nous serons en mesure d'évaluer l'efficacité de notre système.

## 5.1 EDA

Avant de procéder à l'entraînement du modèle, il est essentiel de réaliser une analyse exploratoire des données, souvent appelée **EDA** (Exploratory Data Analysis). C'est une étape cruciale du processus d'analyse de données qui permet de comprendre la nature et la structure des données avec lesquelles nous travaillons. Cela peut comprendre l'identification des tendances, des anomalies, des modèles et des relations entre les variables.

Dans notre cas, nous allons effectuer une EDA sur un jeu de données provenant de la table des **recharges**. Cet ensemble de données comprend **212700** enregistrements de recharges effectuées par **7272 MDN** uniques sur une période de **6 mois**, du **1er mai 2022** au **31 octobre 2022**. C'est l'ensemble de données "granulaires" non agrégé que nous analyserons pour cette EDA.

les colonnes sont de la tables des recharges sont les suivants :

- **MDN** : la clé primaire des clients.
- **ID\_DATE** : la date exacte de la recharge sous forme de timestamp.
- **CHANNEL** : une colonne catégorielle qui contient sept valeurs distinctes représentant les différentes méthodes de recharge (canal) employées par le client. Ces méthodes sont :

- **Transfert dealer** : Cette méthode implique qu'un revendeur (dealer) transfère du crédit sur le compte du client. Les dealers sont généralement des agents commerciaux ou des boutiques spécialisées dans la vente de crédit téléphonique.
  - **Recharge client** : Cela se produit lorsque le client recharge lui-même son compte, habituellement en grattant une carte de recharge achetée dans un magasin et en entrant le code dans son téléphone.
  - **Recharge GAB** : Cette méthode est utilisée lorsque le client recharge son compte via un Guichet Automatique de Banque (GAB). C'est une option particulièrement populaire parmi les clients qui utilisent des cartes bancaires.
  - **Recharge en ligne** : Ici, le client utilise une plateforme en ligne ou une application mobile pour recharger son compte. C'est souvent une option pratique qui permet aux clients de recharger leur compte où qu'ils soient et à tout moment.
  - **INWI MONEY** : Cette méthode est spécifique aux clients utilisant INWI MONEY, une solution de paiement numérique offerte par INWI. Les clients peuvent utiliser INWI MONEY pour recharger leur compte directement à partir de leur téléphone.
  - **MyInwi** : Il s'agit d'une recharge effectuée via l'application MyInwi. Les clients peuvent utiliser cette application pour gérer leur compte et effectuer des recharges.
  - **Recharge Wafacash** : Enfin, cette méthode implique que le client recharge son compte via une agence Wafacash, une entreprise qui offre des services de transfert d'argent et de paiement de factures.
- **NUM\_TOTAL** : une colonne numérique qui indique le nombre de recharges effectuées par jour.
  - **AMNT\_TOTAL\_TTC** : une colonne numérique qui représente le montant de la recharge.
  - **RCH\_TYPE** : une colonne catégorielle qui indique le type d'étoile (promotion) utilisé lors de la recharge. Elle contient dix valeurs distinctes : **1, 2, 3, 4, 5, 6, 7, 8, 9** et **-1**, ce dernier signifiant que le client n'a utilisé aucune étoile.

En raison de la sensibilité des données, toutes les analyses et graphiques produits lors de cette étude seront anonymes. Les valeurs distinctes des colonnes catégorielles seront remplacées par des termes génériques tels que "**cat1, cat2, cat3 ...**". De plus, les axes X et Y des graphiques n'auront pas de légendes afin de garantir l'anonymat des chiffres. Cette approche garantit le respect, la confidentialité et la sécurité des données.

Le premier graphe s'agit d'un histogramme illustrant le nombre de recharges effectuées par montant de recharge. On voit clairement que le montant de recharge le plus effectuée est le **montant 'x1 dh'** suivie du **montant 'x5 dh'**, et puis le **montant 'x15 dh'**



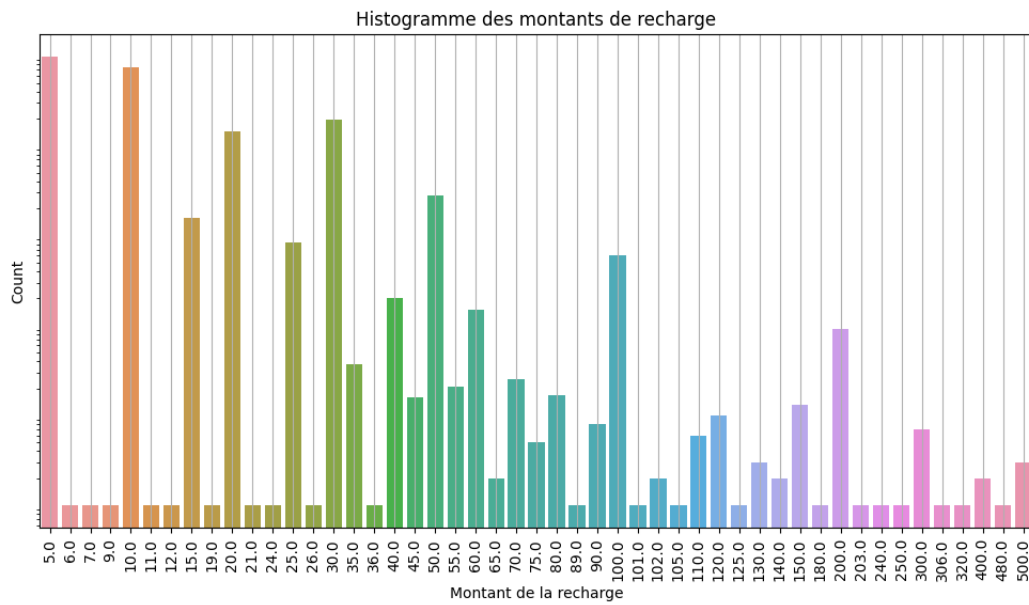


FIGURE 5.1 : Histogramme des montants de recharge (Version anonyme)

Le deuxième graphique montre le nombre de recharges effectuées selon le type du canal utilisé. Les trois canaux 1, 2 et 3 sont les plus populaires, avec le **canal 1** étant de loin le plus utilisé.

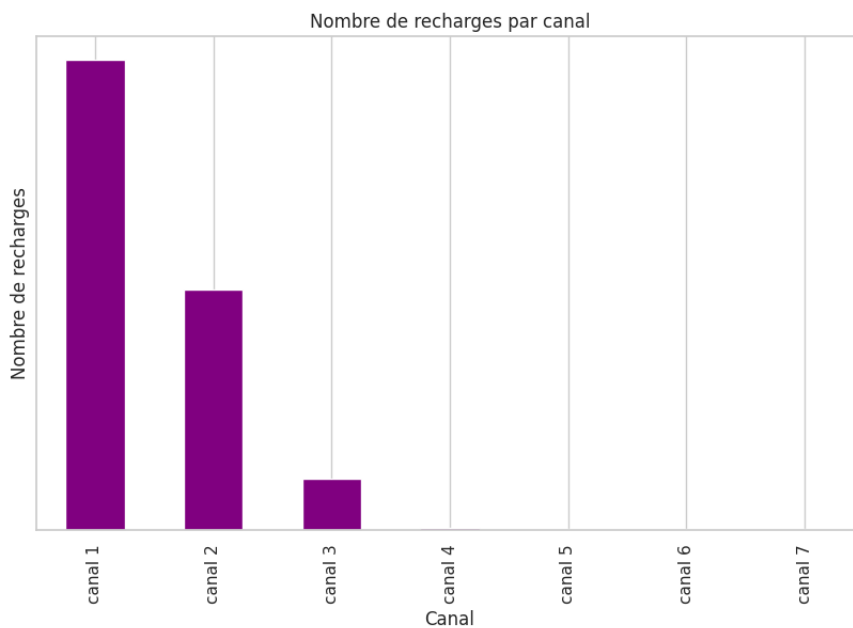


FIGURE 5.2 : Graphe du nombre de recharges par canal (Version anonyme)

On effectue un autre graphe que pour les canaux 4, 5, 6 et 7 avec leur propre échelle pour une meilleur visualisation

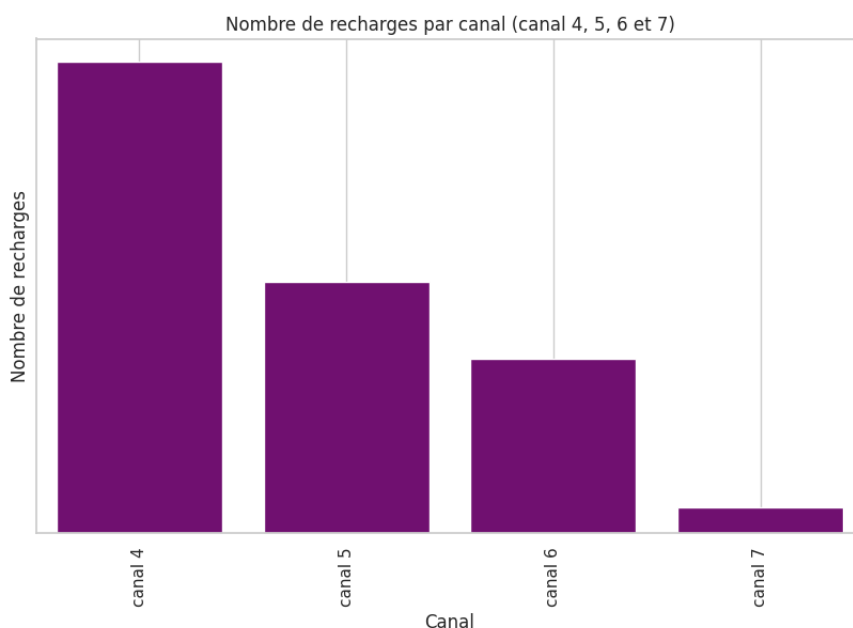


FIGURE 5.3 : Graphe de du nombre de recharges par canal (canal 4, 5, 6 et 7) (Version anonyme)

Dans le même contexte des canaux, un autre graphique intéressant est celui de la somme totale des montants rechargés par canal. Alors que le **canal 2** représente seulement la moitié du **canal 1** en termes de nombre de recharges, le montant total rechargé via ces deux canaux est similaire. Cela suggère que les utilisateurs du **canal 2** chargent généralement des montants plus importants.

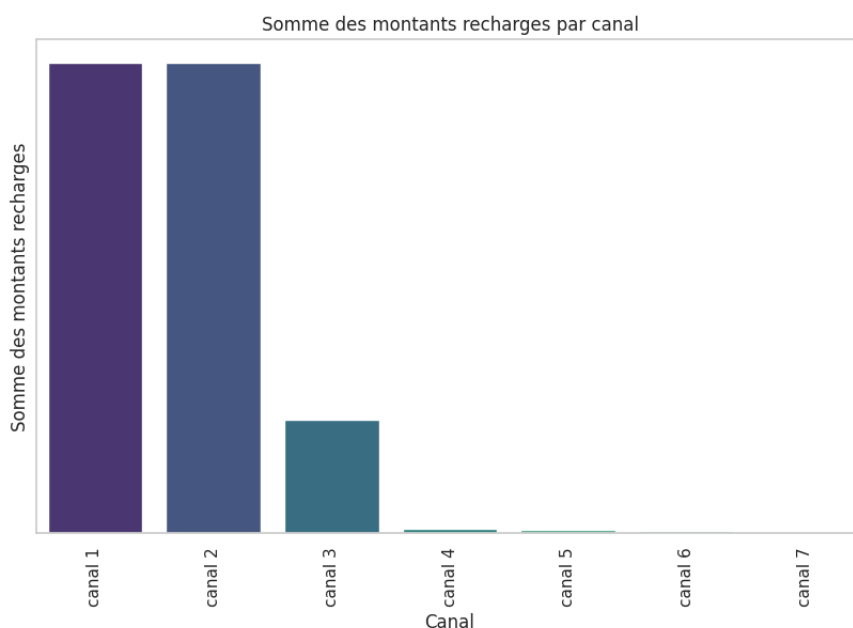


FIGURE 5.4 : Graphe de la somme des montants recharges par canal (Version anonyme)

Concernant les dates, le graphe suivant montre le nombre de recharges effectuées par jours du **1er mai 2022** au **31 octobre 2022**. Cela peut aider à identifier les jours où le plus grand nombre de recharges est effectué, ce qui peut être précieux pour le ciblage des promotions et des offres.

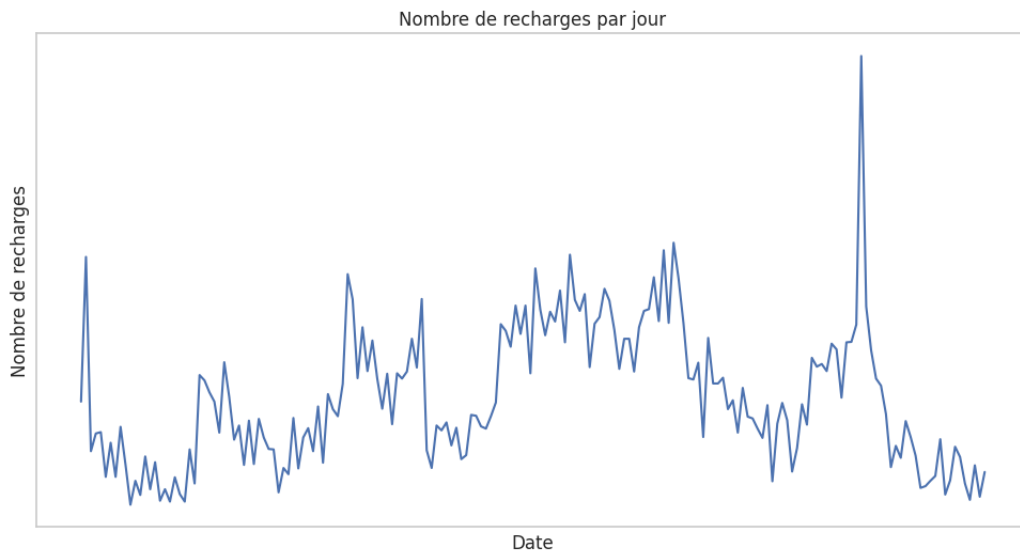


FIGURE 5.5 : Graphe du nombre de recharges par jour (Version anonyme)

Maintenant, on visualise le nombre de recharges effectué pour chaque type de recharges (étoile utilisée après la recharge). Le **type 1, 2 et 3** sont les plus élevés en terme de nombre de recharges.

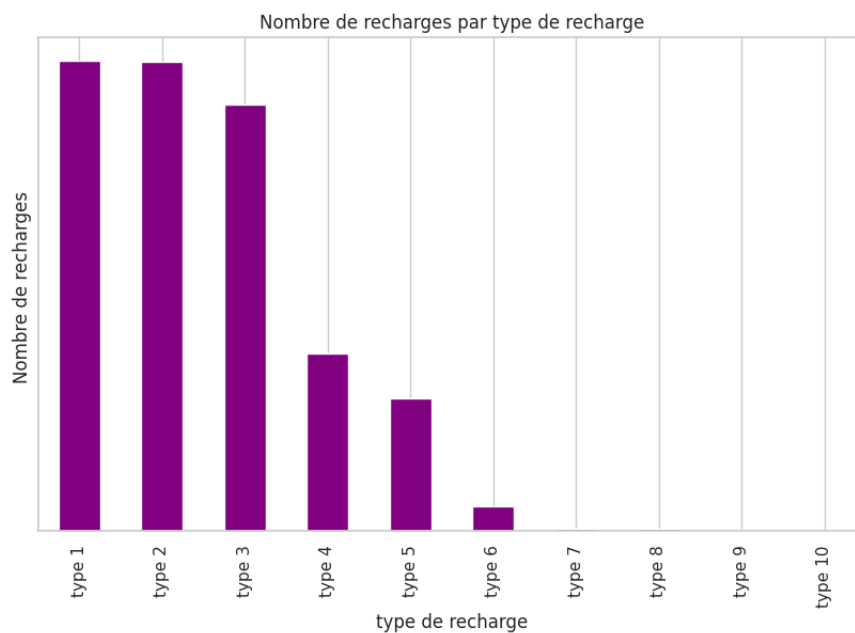


FIGURE 5.6 : Graphe du nombre de recharges par type de recharge (Version anonyme)

Cependant, en termes de montant total rechargé par type de recharge, même si le **type 4** ne représente qu'un tiers du **type 2**, ils sont très proches en termes de montant total de recharges. Le **type 4** dépasse même le **type 1** et **3** en termes de chiffre d'affaires, ce qui montre le montant énorme rechargé en utilisant le **type 4**. Le nombre faibles des recharges du **type 4** est dû au fait que cette promotion n'est pas toujours valable tout au long de l'année, mais plutôt dans des périodes spécifiques.

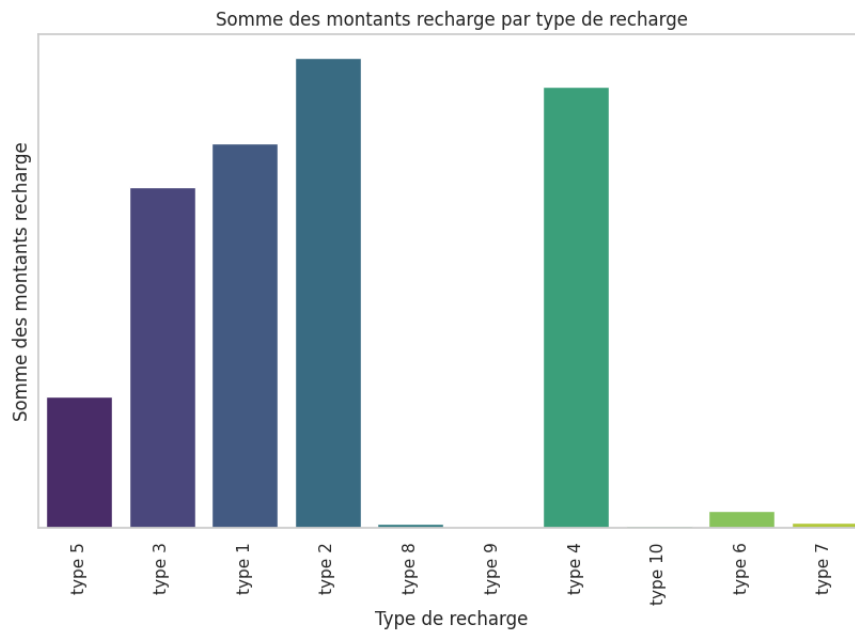


FIGURE 5.7 : Graphe de la somme des montants recharge par type de recharge (Version anonyme)

Le graphe suivant montre que le nombre de recharges par jour de semaine est équilibré. Il n’y a pas de jour particulier où le nombre de recharges est élevé par rapport à un autre jour.

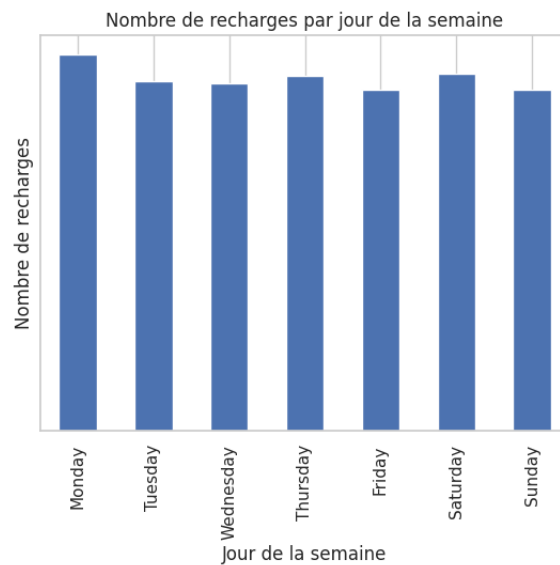


FIGURE 5.8 : Graphe du nombre de recharges par jour de la semaine (Version anonyme)

Même en fonction de type de recharge, la répartition du nombre de recharge par jour reste la même au long de la semaine

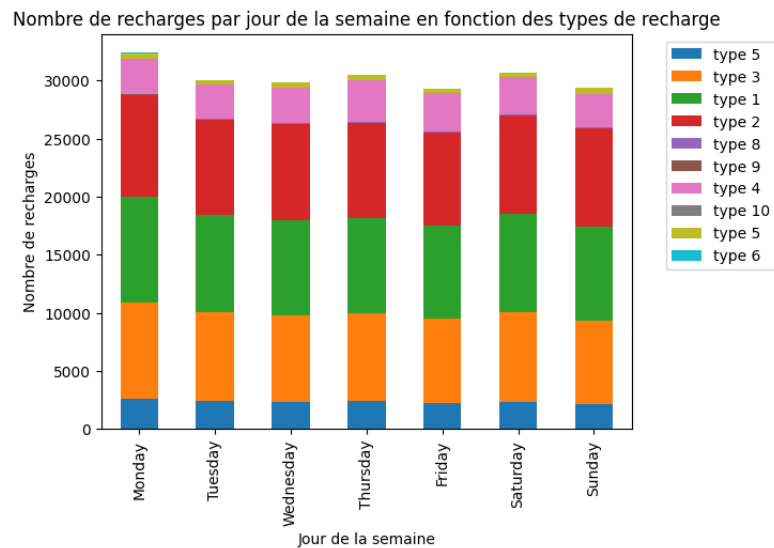


FIGURE 5.9 : Nombre de recharges par jour de la semaine en fonction des types de recharge (Version anonyme)

## 5.2 Entraînement du modèle de prédiction de churn

### 5.2.1 Système de génération des données d'entraînement

Maintenant que nous avons terminé notre analyse exploratoire des données, nous commençons l'étape d'entraînement du modèle. Pour ce faire, nous utilisons notre système de génération de données d'entraînement basé sur la table précédente des recharges que nous avons utilisée lors de l'EDA.

Pour ceci, on utilise les paramètres suivants dans le fichier de configuration :

```
{
  "cutoff_date": "2022-08-31 00:00:00",
  "base_active": {
    "generate": 1,
    "active": {
      "static_table": "parc_prep",
      "static_key": "MDN",
      "CPR1_column": "date_cpr1",
      "tenure": "4",
      "active_table": "sample_rech",
      "active_key": "MDN",
      "active_column": "amnt_total_ttc",
      "active_date_column": "id_date",
      "agg": "count",
      "threshold": "1",
      "time_window": "4"
    },
    "labels": {
      "table": "sample_rech",
      "column": "amnt_total_ttc",
      "agg": "sum",
      "threshold": "10",
      "time_window": "8"
    }
  },
  "params": {
    "categorical_columns_threshold": 5,
    "param_history": "12",
    "time_windows": [
      2,
      4,
      6,
      8
    ],
    "main_table": "base_active_labels"
  },
  "tables": {
    "base_active_labels": {
      "MDN": "key",
      "id_date": "time",
      "num_total": "num",
      "amnt_total_ttc": "num",
      "channel": "cat",
      "rch_type": "cat"
    },
    "sample_rech": {
      "MDN": "key",
      "id_date": "time",
      "num_total": "num",
      "amnt_total_ttc": "num",
      "channel": "cat",
      "rch_type": "cat"
    }
  }
}
```

FIGURE 5.10 : Le fichier de configuration choisi pour le modèle de prédiction de churn

Puisque les données des recharges s'étendent du **1er mai 2022** au **31 octobre 2022**, on a choisi une cutoff date qui est le **31 août 2022**.

Concernant l'activité, nous avons choisi un critère d'ancienneté de **4 semaines**, réduisant ainsi le nombre de clients de **7272** à **6545**. La deuxième condition d'activité que nous avons prise en compte est un nombre de recharges supérieur à **1** dans les **4 dernières semaines** à partir de la cutoff date. Cela a réduit le nombre de clients de **6545** à **3917**.

Pour les labels, nous avons considéré tout client ayant effectué une **somme de recharges** supérieure à **10 dh** dans les **10 semaines** après la cutoff date comme étant non-churn (label 0), et les personnes n'ayant pas satisfait à cette condition comme étant churn (label 1).

Nous avons également fixé un seuil de regroupement des valeurs uniques des colonnes catégorielles en termes de moyenne d'occurrence de **5%**, et un historique de **12 semaines** avec des fenêtres temporelles de **2,4,6** et **8 semaines**.

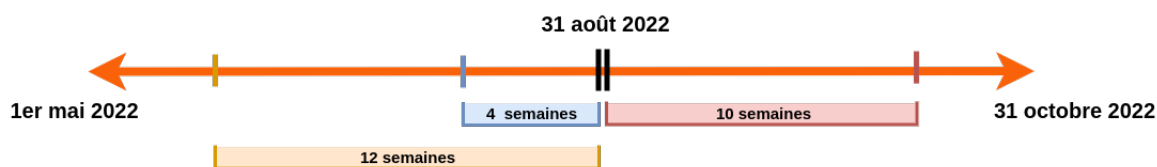


FIGURE 5.11 : Intervalles de temps choisis

En ce qui concerne les tables, nous avons utilisé les colonnes de la table précédente, en spécifiant leurs types.

L'exécution du script a donné comme résultat un dataset de **582 colonnes**

### 5.2.2 Prétraitement des données

Afin de préparer nos données pour la modélisation, nous avons effectué un '**one-hot encoding**' sur deux de nos colonnes catégorielles : '**mode\_rch\_type**' et '**mode\_channel**'. Celui-ci est une technique utilisée pour convertir les valeurs catégorielles en une forme que les modèles de machine learning peuvent mieux comprendre. Essentiellement, pour chaque valeur unique dans la colonne, une nouvelle colonne est créée. Pour chaque enregistrement, ces nouvelles colonnes auront une valeur de 1 si l'enregistrement avait cette valeur dans la colonne d'origine, et 0 sinon.

Pour les valeurs nulles dans nos données, nous avons choisi de les remplacer par la valeur **-9999**. Cette approche peut sembler inhabituelle, mais elle est en fait très efficace pour plusieurs raisons :

- **Indication de l'absence de données** : En utilisant une valeur largement négative et hors de la plage typique de notre jeu de données, comme -9999, nous créons implicitement un indicateur de l'absence de données. Cela peut aider le modèle à apprendre un schéma basé sur ces valeurs manquantes, plutôt que de simplement les voir comme 0 ou la valeur moyenne.
- **Préservation de la pureté des nœuds** : Les algorithmes basés sur des arbres, comme les random forests, effectuent des divisions en fonction de mesures de pureté. Si nous remplaçons les valeurs manquantes par un grand nombre négatif, ces valeurs seront probablement séparées lors des premières divisions des arbres ce qui maintient la pureté des nœuds des variables d'origine.
- **Distinction entre les valeurs manquantes et zéro** : Si nous remplaçons les valeurs manquantes par des zéros, il pourrait être difficile pour le modèle de distinguer entre les cas où la valeur est réellement zéro et où elle est manquante. Cela pourrait poser problème si ces deux situations ont des implications différentes.

### 5.2.3 Équilibrage des classes

La distribution des classes est de **81%** pour le **label 0** (non churn), et **19%** pour le **label 1** (churn), montrant un déséquilibre en faveur du label 0.

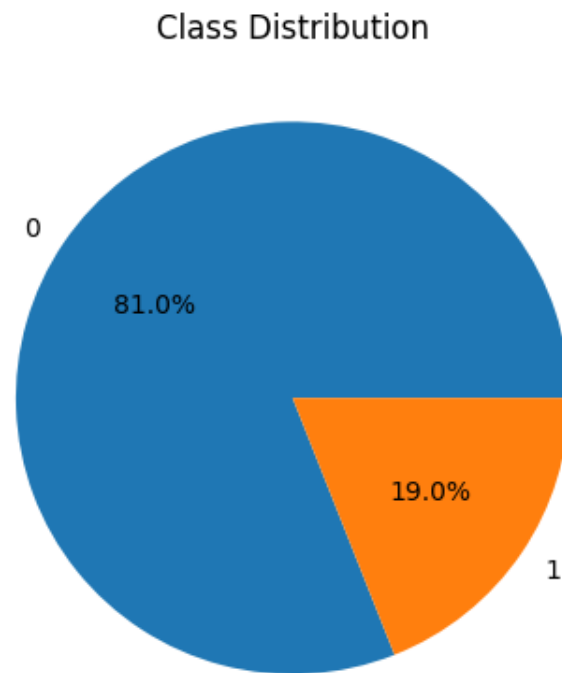


FIGURE 5.12 : Répartition des classes

Pour surmonter ce problème, nous avons utilisé une technique d'équilibrage des données appelée **Synthetic Minority Over-sampling Technique (SMOTE)** [5]. La méthode SMOTE fonctionne en créant des entités synthétiques de la classe minoritaire, qui est la classe 1 dans notre cas. Cela se fait en sélectionnant au hasard un point de la classe minoritaire et en calculant les  $k$ -voisins les plus proches pour ce point. Les points synthétiques sont alors ajoutés entre le point choisi et ses voisins.

Les deux graphiques ci-dessous illustrent les données avant et après équilibrage, en utilisant une **analyse en composantes principales (ACP)** [3] avec 2 et 3 composantes respectivement pour une meilleure visualisation.

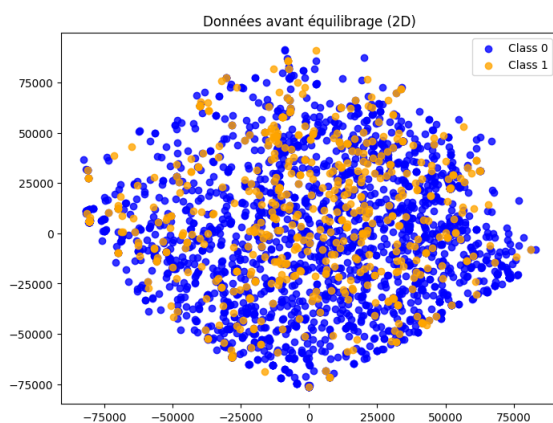


FIGURE 5.13 : dispersion des points de données sur un plan 2d avant suréchantillonnage

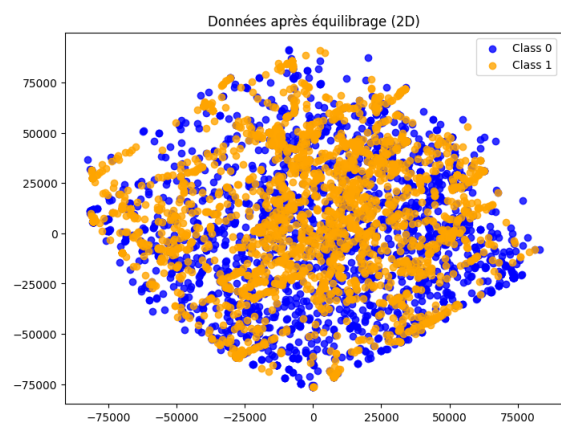


FIGURE 5.14 : dispersion des points de données sur un plan 2d après suréchantillonnage



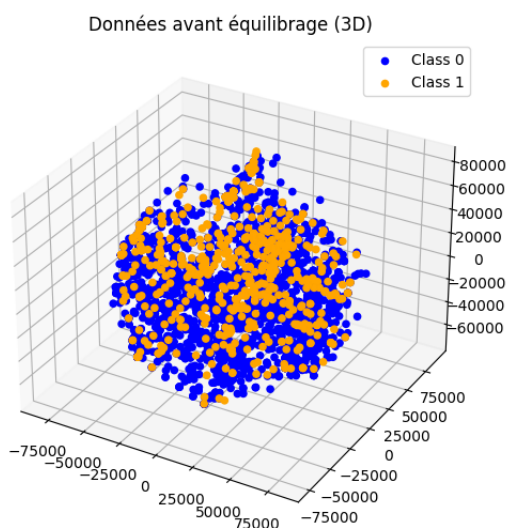


FIGURE 5.15 : dispersion des points de données sur un plan 3d avant suréchantillonnage

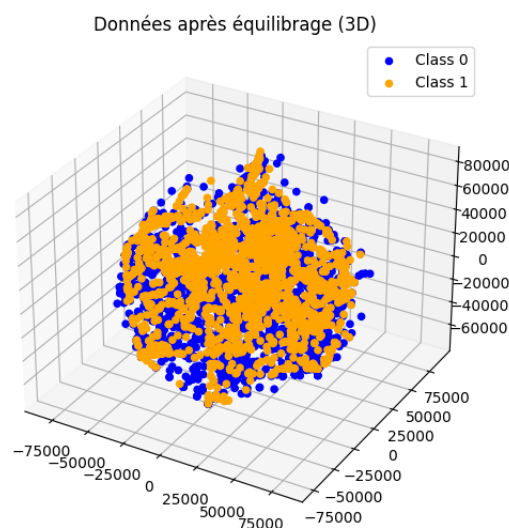


FIGURE 5.16 : dispersion des points de données sur un plan 3d après suréchantillonnage

La distribution des classes après suréchantillonnage de la classe 1 devient comme suit

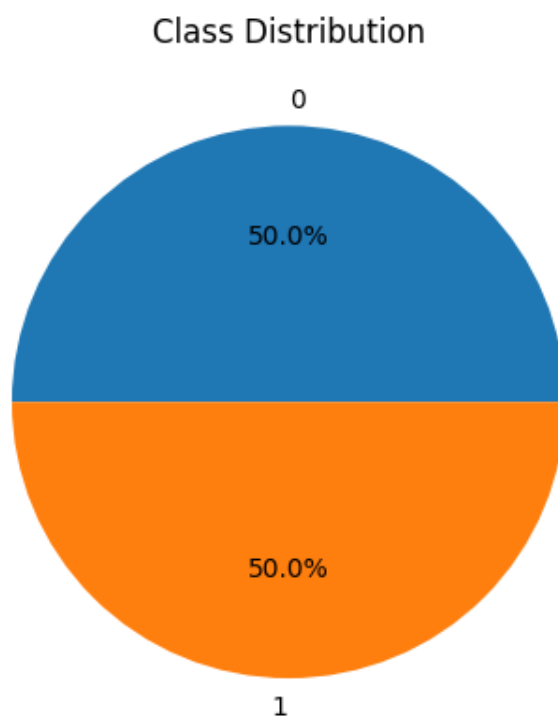


FIGURE 5.17 : Graphe de distribution des labels après suréchantillonnage

### 5.2.4 Sélection des features

En raison de la grande taille de notre jeu de données (**582 caractéristiques**), nous avons choisi d'utiliser la **RFE [2]** pour sélectionner les **50 meilleures caractéristiques**. Le **Recursive Feature Elimination (RFE)** est une technique utilisée pour sélectionner les fonctionnalités les plus importantes dans un jeu de données.

L'idée de base derrière la RFE est assez simple, elle consiste à entraîner d'abord le modèle sur l'ensemble initial de caractéristiques et à obtenir l'importance de chacune de ces caractéristiques. Cette

importance peut être déterminée par les coefficients du modèle comme dans le cas d'une régression linéaire, ou par les caractéristiques d'importance comme dans le cas des arbres de décision. Ensuite, la caractéristique la moins importante est supprimée de l'ensemble de caractéristiques. Le modèle est alors réentraîné sur cet ensemble de caractéristiques réduit, et le processus se répète. Le processus se poursuit jusqu'à ce qu'un nombre prédéfini de caractéristiques soit atteint, dans notre cas 50.

Dans notre modèle, nous avons utilisé le **RandomForestClassifier** comme algorithme pour le RFE. Celui-ci fournit une mesure d'importance des caractéristiques qui peut être utilisée pour éliminer les caractéristiques les moins importantes.

Les top 50 caractéristiques choisis sont les suivant

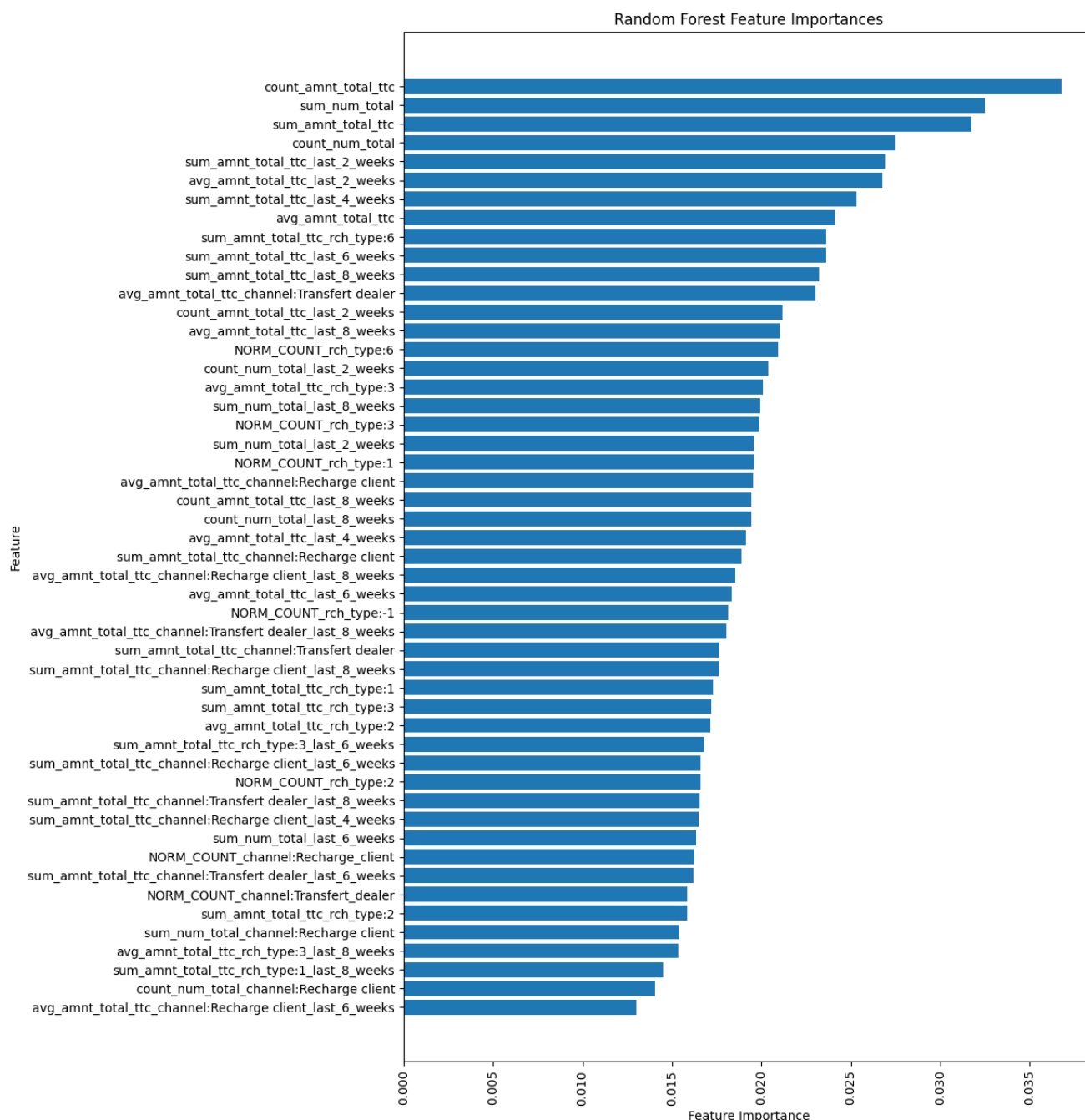


FIGURE 5.18 : Graphe des scores des top 50 caractéristiques

### 5.2.5 Entraînement du modèle

Concernant l'entraînement de notre modèle, nous avons entrepris une approche en deux étapes en utilisant la technique de **Grid Search**. Le premier Grid Search avait pour but de sélectionner l'algorithme de machine learning le plus approprié pour notre problème. Pour ce faire, nous avons entraîné des modèles utilisant quatre algorithmes différents : **KNN** (k-nearest neighbors), **SVM** (Support Vector Machines), **XGBoost** (eXtreme Gradient Boosting) et **Random Forest**. Après avoir comparé les métriques de performance de chaque modèle, nous avons décidé d'opter pour le Random Forest car il offrait les meilleurs résultats en termes de précision et de robustesse.

La deuxième étape du Grid Search visait à optimiser les hyperparamètres du modèle Random Forest. C'est une étape cruciale car le réglage fin des hyperparamètres peut améliorer considérablement la performance du modèle. Le Grid Search a systématiquement évalué différentes combinaisons d'hyperparamètres pour déterminer celles qui produisaient le meilleur modèle. Les résultats du Grid Search ont indiqué que la meilleure configuration d'hyperparamètres pour notre modèle Random Forest était la suivante :

- **"profondeur maximale"** (max\_depth) : Il s'agit de la profondeur maximale des arbres dans la forêt. Le résultat qu'on a obtenu '**None**' signifie que les arbres seront développés jusqu'à ce que toutes les feuilles soient pures, c'est-à-dire jusqu'à ce que chaque feuille contient des échantillons d'une seule classe ou jusqu'à ce qu'elles contiennent moins d'échantillons que le minimum défini par "min\_samples\_split".
- **"min\_samples\_leaf"** : Ce paramètre définit le nombre minimum d'échantillons requis pour être au niveau d'une feuille (c'est-à-dire à l'extrémité) de l'arbre. Le résultat qu'on a obtenu '**1**' signifie qu'il peut y avoir des feuilles avec un seul échantillon de données.
- **"min\_samples\_split"** : Il s'agit du nombre minimum d'échantillons nécessaires pour diviser un nœud interne. Le résultat qu'on a obtenu '**2**' signifie qu'un nœud doit contenir au moins deux échantillons de données pour être divisé en sous-nœuds.
- **"n\_estimators"** : Il s'agit du nombre d'arbres dans la forêt. Le résultat qu'on a obtenu '**300**' signifie que la forêt finale de notre modèle consistait en 300 arbres de décision différents.

### 5.2.6 Métriques de performance

Une fois nos données d'entraînement préparées, l'étape suivante consiste à tester les performances de notre modèle sur un ensemble de données inédit que le modèle n'avait jamais vues auparavant. Pour ce faire, nous avons divisé au début notre ensemble de données en un ratio de **80%** pour l'entraînement et **20%** pour le test. Nous avons conservé la distribution originale des classes pour cet ensemble de données de test afin de refléter le plus fidèlement possible la réalité du monde réel.

Par ailleurs, afin d'évaluer l'efficacité de notre système de génération de données d'entraînement, nous avons également entraîné un second modèle sans utiliser ce système. Comme les agrégations pour ce deuxième modèle ont été écrites manuellement, leur nombre et leur qualité étaient nécessairement limités en raison des contraintes de temps. Pour une comparaison équitable, nous avons veillé à ce que le temps consacré à la rédaction de ces agrégations soit équivalent à celui nécessaire pour générer les agrégations à l'aide de notre système. Ainsi, en comparant les performances de ces deux modèles, nous avons pu évaluer l'efficacité et l'utilité de notre système de génération de données d'entraînement.

Class	Precision	Recall	F1-Score
0	0.87	0.94	0.90
1	0.76	0.75	0.79
Accuracy	0.84		
Macro Avg	0.72 / 0.75 / 0.77		
Weighted Avg	0.82 / 0.84 / 0.82		
Support	784		

TABLE 5.1 : Rapport de classification du 1er modèle en utilisant le système de génération des données d'entraînement

Class	Precision	Recall	F1-Score
0	0.84	0.63	0.72
1	0.22	0.45	0.29
Accuracy	0.52		
Macro Avg	0.53 / 0.54 / 0.50		
Weighted Avg	0.72 / 0.60 / 0.64		
Support	784		

TABLE 5.2 : Rapport de classification du 2ème modèle sans utiliser le système de génération des données d'entraînement

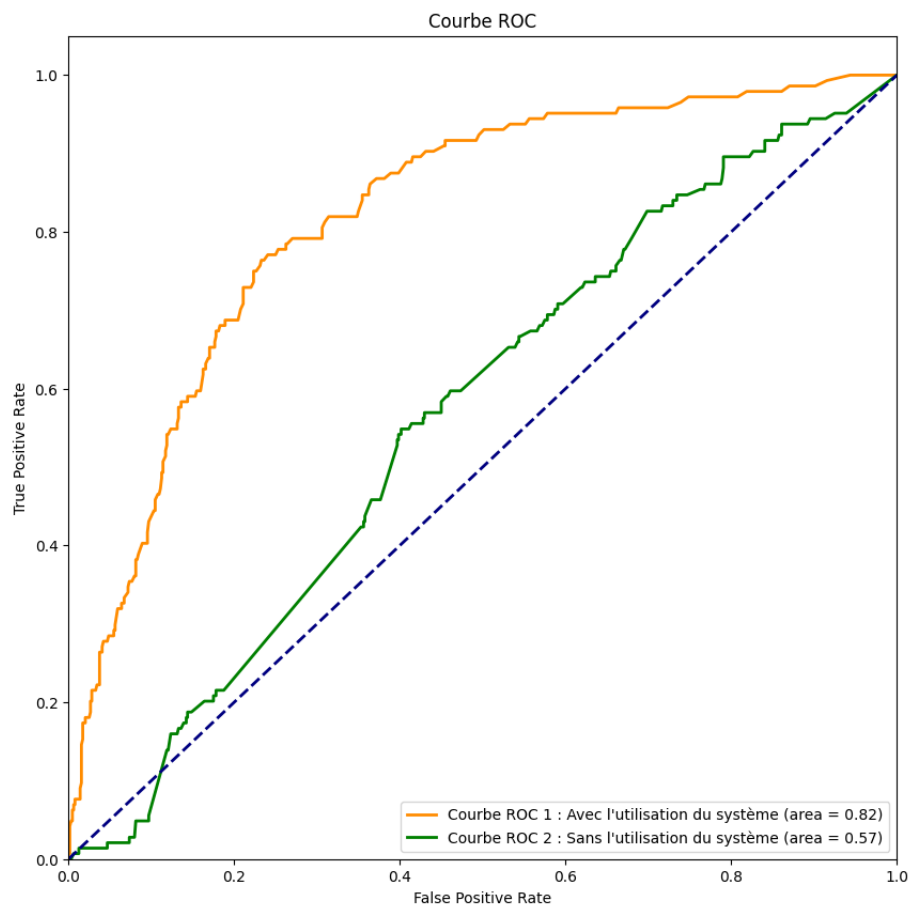


FIGURE 5.19 : Graphe de comparaison des deux courbes ROC des deux modèles

En examinant et en comparant les métriques de performance de nos deux modèles, on constate que le premier modèle qui a été formé en utilisant notre système de génération de données d'entraînement a nettement surpassé le deuxième modèle qui a été formé sans utiliser ce système.

Pour le premier modèle, la courbe ROC [4] est de **82%**, démontrant une bonne performance de notre modèle. La précision 'accuracy' est de **84%**, indiquant que notre modèle prédit correctement les labels dans environ 84% des cas. La précision, le rappel et le score F1 pour chaque classe démontrent également une performance relativement bonne et bien équilibrée pour les deux classes.

D'autre part, le deuxième modèle affiche une courbe ROC de seulement **57%**, indiquant une performance médiocre. Sa précision n'est que de **52%**, soit beaucoup moins que celle du premier modèle. La précision, le rappel et le score F1 sont également nettement inférieurs à ceux du premier modèle.

En guise de conclusion, l'utilisation de notre système de génération de données d'entraînement a réellement amélioré la performance de notre modèle de prédiction. Cela démontre l'efficacité et l'utilité de notre système pour préparer des données d'entraînement, et renforce l'argument en faveur de l'automatisation de ce processus.

# Conclusion Générale

En conclusion, le système de génération de données d'entraînement que nous avons développé et présenté dans ce rapport s'est révélé être un véritable succès. Il a su répondre à un ensemble de problématiques complexes, en facilitant le passage de données au format log vers un format entité mieux adapté à l'entraînement des modèles de machine learning. De plus, il a grandement réduit le temps d'écriture manuelle des agrégations, minimisant ainsi les coûts et l'effort humain.

Au-delà de ces améliorations opérationnelles, notre système a aussi permis de limiter le biais humain dans le choix des agrégations à créer. En automatisant ce processus, nous avons pu garantir une meilleure représentativité des données et favoriser la découverte de nouvelles structures significatives.

Surtout, il est important de souligner l'impact positif de ce système sur la performance de nos modèles de prédiction. En particulier, notre modèle de prédiction de churn a vu ses performances considérablement améliorées, ce qui démontre l'efficacité et l'utilité de notre système pour la préparation de données d'entraînement.

Ces résultats prometteurs nous encouragent à poursuivre le développement de ce système, en l'adaptant à d'autres scénarios d'usage et en intégrant de nouvelles fonctionnalités pour le rendre encore plus puissant et polyvalent. Nous sommes convaincus que l'automatisation du processus de préparation des données d'entraînement représente une avancée significative pour l'application de l'apprentissage automatique à des problématiques concrètes et complexes.

C'est avec une anticipation vivifiante que nous attendons de voir comment cette technologie évoluera pour continuer à améliorer le rendement de l'équipe data science à INWI.

# Bibliographie

- [1] J. Ahn. A survey on churn analysis. 2020.
- [2] D. Brzezinski. Recursive feature elimination. 2020.
- [3] A. Charpentier, S. Mussard, and T. Ouraga. Principal component analysis : A generalized gini approach. 2019.
- [4] T. Gneiting and P. Vogel. Receiver operating characteristic (roc) curves. 2018.
- [5] M. Mukherjee and M. Khushi. Smote-enc : A novel smote-based method to generate synthetic data for nominal and continuous features. 2021.

---