



UNIVERSITE D'ANTANANARIVO

ECOLE SUPERIEURE POLYTECHNIQUE



DEPARTEMENT TELECOMMUNICATION

MEMOIRE DE FIN D'ETUDES

En vue de l'obtention

Du **DIPLOME DE MASTER**

Mention : télécommunication

Parcours : Système de Traitement d'Informations

Par : MAHANAMANA Andriamiharisoa

**« PREDICTION D'ATTRITION ET RETENTION CLIENTS EN TELECOMMUNICATION
BASEE SUR LE MACHINE LEARNING »**

Soutenu le 13 avril 2018 devant la commission d'Examen composée de :

Président : M. RAKOTOMALALA Mamy Alain

Examineurs :

M. RATSIMBAZAFY Andriamanga

M. RANDRIARIJAONA Lucien Elino

M. ANDRIAMANALINA Ando Nirina

Directeur de mémoire : Mme ANDRIANTSILAVO Haja Samiarivonjy

REMERCIEMENTS

Tout d'abord, la Gloire est offerte à Dieu, Notre Seigneur Tout Puissant, qui nous a donné vie, santé et force pour surmonter toutes les difficultés, qui nous a prêté l'assistance nécessaire pour mener à bien ce présent travail.

Nos remerciements vont également à :

- Monsieur ANDRIANAHARISON Yvon, Professeur Titulaire, responsable du domaine des sciences de l'ingénieur
- Monsieur RAKOTOMALALA Mamy Alain, Maître de Conférences, et Chef de la mention Télécommunication, qui nous a fait l'honneur de présider le Jury de ce mémoire.
- Madame ANDRIANTSILAVO Haja Samiarivonjy, Assistant, Directeur de ce mémoire pour ses conseils et aides très précieux.

Je suis très reconnaissant envers les membres du Jury composés de :

- Monsieur RATSIMBAZAFY Andriamanga, Maître de Conférences
- Monsieur RANDRIARIJAONA Lucien Elino, Assistant
- Monsieur ANDRIAMANALINA Ando Nirina, Maître de Conférences

J'adresse un grand Merci à tous les Membres de ma Famille, à tous les Enseignants Chercheurs et Personnels administratifs de l'Ecole Supérieure Polytechnique d'Antananarivo.

TABLE DES MATIERES

REMERCIEMENTS	i
TABLE DES MATIERES	ii
NOTATIONS ET ABREVIATIONS.....	x
INTRODUCTION GENERALE.....	1
CHAPITRE 1 INITIATION EN SCIENCE DES DONNEES.....	2
1.1 Introduction.....	2
1.2 Définition	2
1.3 Domaine d'utilisation science des données	2
1.3.1 Recommandation des produits.....	2
1.3.2 Conduite autonome.....	3
1.4 Les données d'entrée.....	4
1.4.1 Données Structurées	4
1.4.2 Données semi-structurées	4
1.4.3 Données non Structurées	5
1.5 Les principaux types des données.....	5
1.5.1 Les données quantitatives	5
1.5.2 Les données qualitatives	5
1.6 Processus d'extraction de connaissance.....	6
1.6.1 Compréhension du problème métier.....	6
1.6.2 Récupération des données	7
1.6.3 Préparation des données	7
1.6.4 Modélisation des données ou construction de modèles.....	7
1.6.5 Evaluations	7
1.6.6 Déploiement.....	7

1.7	Data warehouse ou entrepôt de données	8
1.7.1	Présentation	8
1.7.2	Caractéristiques	9
1.7.3	Fonctions Data warehouse.....	9
1.7.4	Les classes des données	9
1.7.4.1	Les données agrégées	9
1.7.4.2	Les données détaillées	10
1.7.4.3	Les métadonnées.....	10
1.7.4.4	Les données historiées	10
1.7.5	Avantages	10
1.8	Data mining ou fouille des données	10
1.8.1	Initiation.....	10
1.8.2	Définition.....	11
1.8.3	Fonctions data mining	11
1.8.3.1	Classification	11
1.8.3.2	Clustering ou segmentation	11
1.8.3.3	Règles d'association	11
1.8.4	Utilisations.....	11
1.9	Text mining.....	12
1.9.1	Définition.....	12
1.9.2	Applications.....	12
1.9.3	Etapes de l'algorithme de text mining.....	12
1.10	Web mining.....	13
1.10.1	Définition	13
1.10.2	Utilisation de web mining	13

1.10.3	Applications	13
1.10.4	Avantages.....	13
1.11	Le Big data ou la science des données ?	14
1.12	Conclusion	14
CHAPITRE 2 LE MACHINE LEARNING.....		15
2.1	Introduction.....	15
2.2	Définitions.....	15
2.3	Concept général de l'apprentissage automatique.....	15
2.4	Les bases de la technologie de Machine Learning.....	16
2.4.1	Illustration.....	16
2.4.2	Données d'entrée	16
2.4.3	Learning.....	17
2.4.3.1	Formation et sélection d'un modèle prédictif.....	17
2.4.3.2	Evaluation du modèle	17
2.4.4	Données de sortie.....	17
2.5	Types d'apprentissage automatique	18
2.5.1	Apprentissage supervisé	18
2.5.2	Apprentissage non supervisé	20
2.5.3	Apprentissage par renforcement	21
2.6	Présentation de quelques algorithmes d'apprentissage automatique	22
2.6.1	K-plus proche voisin.....	22
2.6.2	L'arbre de décision	23
2.6.2.1	Structure d'un arbre	23
2.6.2.2	Principe général	24
2.6.2.3	Construction de l'arbre	24

2.6.2.4	Avantages	26
2.6.3	Random forest.....	26
2.6.4	Naïve Bayes	27
2.6.4.1	Introduction	27
2.6.4.2	Fondement	27
2.6.4.3	Principe.....	27
2.6.4.4	Notion en probabilité	27
2.6.4.5	Avantages et inconvénients	28
2.6.5	Régression logistique.....	28
2.6.5.1	Définition.....	28
2.6.5.2	Utilisations.....	28
2.6.5.3	Principe de fonctionnement	29
2.6.6	Support Vector Machine ou SVM.....	29
2.6.6.1	Définition.....	29
2.6.6.2	Principe.....	30
2.6.7	Le boosting	30
2.6.7.1	AdaBoost	30
2.6.7.2	Le gradient boosting	31
2.6.8	Algorithmes de clustering.....	31
2.6.8.1	Illustration.....	31
2.6.8.2	Types de clustering	31
2.6.8.3	K-Means clustering.....	32
2.6.8.4	Clustering hiérarchique.....	32
2.6.9	Deep learning.....	33
2.6.9.1	Aperçu	33

2.6.9.2	Initiation au Deep Learning	33
2.6.9.3	Fonctionnement	34
2.7	Evaluations modèle métriques	35
2.7.1	Matrice de confusion	35
2.7.2	Courbe ROC ou (Received Operating Characteristic)	36
2.7.3	Validation croisée	37
2.8	Avantages de la Machine Learning.....	38
2.9	Les relations de Machine Learning avec d'autres domaine.....	38
2.10	Conclusion	39
CHAPITRE 3 LES OUTILS DE MACHINE LEARNING		40
3.1	Introduction.....	40
3.2	Tâche d'un outil d'apprentissage automatique	40
3.2.1	Trouver un pattern dans les données	40
3.2.2	Développement d'un Data product.....	40
3.2.3	Visualisation des données.....	40
3.2.3.1	Histogramme	41
3.2.3.2	Bar plot	41
3.2.3.3	Box plot	41
3.2.3.4	Scatterplot Matrices.....	42
3.3	Machine learning avec MATLAB	43
3.3.1	Statistique et machine learning.....	43
3.3.1.1	Méthode de classification	43
3.3.1.2	Méthode de régression.....	43
3.3.1.3	Méthode clustering	44
3.3.2	Réseau de neurone	44

3.4	Machine learning avec R	44
3.4.1	Introduction	44
3.4.2	Les paquets ou extensions	45
3.4.2.1	Utiliser un paquet.....	45
3.4.2.2	Installer un paquet	45
3.4.3	Avantages de l'utilisation de R.....	45
3.5	Machine learning avec Python.....	46
3.5.1	Installation de paquet en python.....	46
3.5.2	Numpy	46
3.5.3	Pandas.....	47
3.5.3.1	Définition.....	47
3.5.3.2	Series	47
3.5.3.3	DataFrame	48
3.5.3.4	Illustration.....	48
3.5.4	Matplotlib et Seaborn	49
3.5.5	Scikit-Learn	50
3.5.6	Keras.....	50
3.6	Machine learning avec Java	50
3.6.1	WEKA	50
3.6.2	Java Machine Learning.....	51
3.6.3	Apache Mahout.....	51
3.6.4	Comparaison librairie Java	52
3.7	Machine learning avec Spark.....	52
3.7.1	Clusters Spark.....	53
3.7.2	DataFrame	53

3.8	Microsoft Azure Machine Learning.....	53
3.8.1	Les composants d'une expérience	54
3.8.2	Les étapes de la création d'une expérience	55
3.9	Conclusion	55
CHAPITRE 4 PRESENTATION DE L'OUTIL D'ATTRITION ET DE LA RETENTION ..		56
4.1	Introduction.....	56
4.2	Le churn	56
4.3	La rétention client	56
4.4	A propos de la plateforme.....	56
4.4.1	Objectifs.....	56
4.4.2	Les outils de développement	57
4.5	Présentations des interfaces	57
4.5.1	Section accueil.....	57
4.5.2	Page principale	58
4.6	Présentations du jeu de données	59
4.7	Feature engineering.....	59
4.7.1	Normalisation dataset	59
4.7.2	Validation du dataset	61
4.8	L'analyse exploratoire	62
4.8.1	Division jeux des données	67
4.9	Les algorithmes de classification utilisés.....	67
4.10	Evaluation et amélioration du modèle	67
4.10.1	K Nearest Neighbors ou KNN	67
4.10.1.1	Evaluation initiale.....	67
4.10.1.2	Amélioration.....	69

4.10.1.3	Comparaison	70
4.10.2	Régression logistique	70
4.10.2.1	Evaluation initiale.....	70
4.10.2.2	Amélioration.....	71
4.10.3	Les arbres de décision	72
4.10.3.1	Evaluation initiale.....	72
4.10.3.2	Améliorations	72
4.10.4	Les forêts aléatoires	74
4.10.4.1	Evaluation initiale.....	74
4.10.4.2	Amélioration.....	75
4.10.5	Réseau de neurones.....	76
4.10.5.1	Keras	76
4.10.5.2	Définition d'un modèle.....	77
4.10.5.3	Evaluations	78
4.11	Stratégie pour la rétention clients	79
4.11.1	Segmentation client avec le K-mean.....	80
4.11.1.1	Proposition de nouvelle offre	81
4.12	Conclusion	82
CONCLUSION GENERALE		83
ANNEXE1 EXTRAIT DU CODE SOURCE.....		84
ANNEXE 2 SOURCES DES DONNEES PUBLIC		85
BIBLIOGRAPHIE		87
FICHE DE RENSEIGNEMENTS		89
RESUME		
ABSTRACT		

NOTATIONS ET ABREVIATIONS

1. Majuscules latines

H_b	Entropie
X	variable aléatoire discrète
S	Représente l'échantillon
$P(A)$	Probabilité de l'évènement A
$P(A B)$	Probabilité pour A d'avoir B

2. Abréviations

2D	2 Dimensions
3D	3 Dimensions
AdaBoost	Adaptative Boosting
API	Application Programming Interface
CART	Classification And Regression Tree
CMD	Command
CRM	Customer Relationship Management
CSS	Cascading Style Sheets
CSV	Comma-Separated Values
ED	Entreposage des Données
FN	Faux Négatifs
FP	Faux Positifs
GNU	GNU'Not Unix
GPL	Genera Public Licence
HTML	HyperText Markup Language
IA	Intelligence Artificielle

IoT	Internet of Things
IQR	Interquartile Range
Java-ML	Java Machine Learning
JSON	JavaScript Object Notation
JVM	Java Virtual Machine
KDD	Knowledge Discovery in Database
KNN	K Nearest Neighbors
MacOs	Macintosh
MATLAB	Matrix Laboratory
MC	Matrice de Confusion
Mlib	Machine Learning Library
OCR	Optical Character Recognition
OLAP	On Line Analytical Processing
PCA	Principal Component Analysis
PDF	Portable Document Format
PNG	Portable Network Graphics
RDD	Resilient Distributed Dataset
RL	Régression Logistique
RNA	Réseau de Neurone Artificiel
ROC	Receiver Operating Characteristic
SQL	Structured Query Language
SVM	Support Vector Machines
VN	Vrais Négatifs
VP	Vrais Positifs
WEKA	Waikato Environment for Knowledge Analysis
XML	eXtensible Markup Language

INTRODUCTION GENERALE

Nous vivons au milieu d'un déluge de données. Selon des estimations récentes, 2,5 quintillions (10^{18}) octets de données sont générés quotidiennement. C'est tellement de données que plus de 90% des informations que nous stockons aujourd'hui ont été générées au cours de la dernière décennie seulement. Malheureusement, la plupart de ces informations ne peuvent pas être utilisées par les humains. Soit les données sont hors de portée des méthodes analytiques standard, soit tout simplement trop vastes pour que nos esprits limités puissent même les comprendre.

Grâce à l'apprentissage automatique ou Machine Learning, nos ordinateurs peuvent traiter, apprendre et à tirer des conclusions exploitables à notre place.

L'apprentissage automatique consiste à extraire des connaissances à partir de données. Son application est devenue omniprésente dans la vie de tous les jours au cours des dernières années.

Des recommandations automatiques sur les films à regarder ou sur les produits à acheter, des reconnaissances d'image, lorsqu'on consulte des sites Web complexes comme Facebook, Amazon. Il est très probable que chaque partie du site contienne plusieurs modèles d'apprentissage automatique.

L'objectif principal de ce mémoire est la «prédiction d'attrition et rétention clients en télécommunication basée sur le machine learning ».

Afin de mieux exposer le thème, ce livre sera divisé en quatre (4) parties. Dans un premier temps, on parlera de la généralité concernant la science des données ou Data Science.

Ensuite, on abordera les principes et les algorithmes d'apprentissage automatique jusqu'aux différentes méthodes d'évaluations d'un modèle.

Puis, on continuera avec la présentation des outils et les plateformes d'apprentissage automatique.

Finalement, la dernière partie traitera l'analyse exploratoire et prédictive de nos jeux des données avec l'utilisation de bibliothèque Python comme scikit learn et Keras.

CHAPITRE 1

INITIATION EN SCIENCE DES DONNEES

1.1 Introduction

Nous vivons dans un monde submergé par les données. Les sites web suivent à la trace chaque clic de chaque utilisateur. Tous les jours, votre smartphone enregistre votre localisation et votre vitesse à la seconde près. C'est ainsi qu'actuellement, la science de données ou data science est un sujet brûlant et en pleine croissance. Elle se situe sur l'intersection de plusieurs domaines, notamment l'informatique, les mathématiques et l'expertise fonctionnelle.

1.2 Définition

La science de données est un domaine qui joue un rôle clé dans la valorisation des données. Il est défini comme étant le processus d'extraction de connaissance ou de savoir utile généralement à partir de données massives. [1]

1.3 Domaine d'utilisation science des données

Le data science est utilisé pour aider les décideurs dans de nombreux secteurs tels que la science, l'ingénierie, l'économie, la politique, la finance et l'éducation. [2]

- Informatique : reconnaissance de forme, visualisation, entreposage de données, base de données, intelligence artificielle
- Mathématiques : modélisation mathématique
- Statistiques : modélisation statistique et stochastique, probabilité

1.3.1 Recommandation des produits

Amazon utilise la Data Science pour proposer des articles pertinents pour ses clients. Ceci en se basant sur leurs historiques de navigation, d'achat sur le site ainsi que les données des autres clients.



Figure 1.01 : *Système de recommandation d'Amazon*

1.3.2 Conduite autonome

La conduite autonome fait rêver plus d'une personne. Ces dernières années, de multiples prototypes fonctionnels ont vu le jour. Certains constructeurs, notamment Tesla Motors commercialisent d'ores et déjà des voitures électriques avec une conduite autonome.



Figure 1.02 : *Une voiture autonome du constructeur Tesla*

Grâce à des techniques de Deep Learning (un sous domaine du Machine Learning), Tesla a su mettre en œuvre des voitures fiables. L'adoption à échelle mondiale n'est qu'une question de temps. Par ailleurs, grâce à la fiabilité qu'offrent les voitures autonomes, des horizons meilleurs sont à espérer pour la sécurité routière.

1.4 Les données d'entrée

Les données d'entrée d'une application de science de données peuvent se présenter sous plusieurs catégories, comme par exemples, les données structurées, semi-structurées et enfin les données non structurées. Dorénavant, on va détailler ces catégories de données pour mieux les appréhender.

La figure 1.03 résume les données d'entrée et leurs exemples types.

Structurées	Semi-structurées	Non-structurées
Base de donnée	XML/JSON Email Pages Web	Audio Video Image

Figure 1.03: *Les données d'entrée en science de données*

1.4.1 Données Structurées

Les données structurées sont des données qui dépendent d'un modèle de données et résident dans un champ fixe dans un enregistrement. En tant que telles, il est souvent plus facile de stocker les données structurées dans des tableaux, des bases de données ou des fichiers Excel.

Le langage SQL ou Structured Query Language est le moyen privilégié pour gérer et scruter les données qui résident dans les bases de données. On peut également rencontrer des données structurées qui pourraient donner du mal à les stocker dans une base de données relationnelle traditionnelle telles que les données hiérarchiques comme par exemple un arbre généalogique.

1.4.2 Données semi-structurées

Les données semi-structurées n'ont pas le même niveau d'organisation et de prévisibilité des données structurées. Les données ne résident pas dans des champs fixes ou des enregistrements, mais contiennent des éléments qui peuvent séparer les données en diverses hiérarchies.

Des exemples de données semi-structurées sont :

- JSON
- XML
- Fichier .CSV

1.4.3 *Données non Structurées*

Les données non structurées sont des données qui n'ont pas de modèle de données fixe et qui ne sont pas arrangées de manière prédéfinie. Ce sont des données brutes, c'est-à-dire, sans prétraitement, et qui ne peuvent pas être stockés dans une table.

Les données non structurées peuvent être produite par toutes sortes d'acteurs et existe sous toutes sortes de forme telles que : email, message texte, transcription audio, vidéo de surveillance. [3]

1.5 Les principaux types des données

On distingue généralement deux (2) types des données :

- Les données quantitatives
- Les données qualitatives.

1.5.1 *Les données quantitatives*

Les données quantitatives sont des valeurs qui décrivent une quantité mesurable, sous la forme de nombres sur lesquels on peut faire des calculs (moyenne, médiane,...) et des comparaisons (égalité/différence, infériorité/supériorité,...). On fait parfois la différence entre :

- Les données quantitatives continues, qui peuvent prendre n'importe quelle valeur dans un ensemble de valeurs comme par exemple la température.
- Les données quantitatives discrètes, qui ne peuvent prendre qu'un nombre limité de valeurs dans un ensemble de valeurs : le nombre d'enfant par famille, le nombre de pièces d'un logement.

1.5.2 *Les données qualitatives*

Ils décrivent quant à elles des qualités ou des caractéristiques. Elles répondent à des questions de la forme « quel type ? » ou « quelle catégorie ? ». Ces valeurs ne sont plus des nombres, mais un ensemble de modalités. On ne peut pas faire de calcul sur ces valeurs, même dans l'éventualité où elles prendraient l'apparence d'une série numérique.

Elles peuvent toutefois être comparées entre elles et éventuellement triées. On distingue :

- Les données qualitatives nominales ou catégorielles, dont les modalités ne peuvent être ordonnées. Par exemple : la couleur des yeux (bleu, vert, marron), le sexe (homme, femme),
- Les données qualitatives ordinales, dont les modalités sont ordonnées selon un ordre logique. Par exemple : les tailles de vêtements (S, M, L), le degré d'accord à un test d'opinion (fortement d'accord, d'accord, pas d'accord, fortement pas d'accord).

1.6 Processus d'extraction de connaissance

Le processus de data science se décompose en six (6) étapes allant de la compréhension du problème métier au déploiement et la mise en production. Nous allons les présenter brièvement dans ce paragraphe. [4][5]

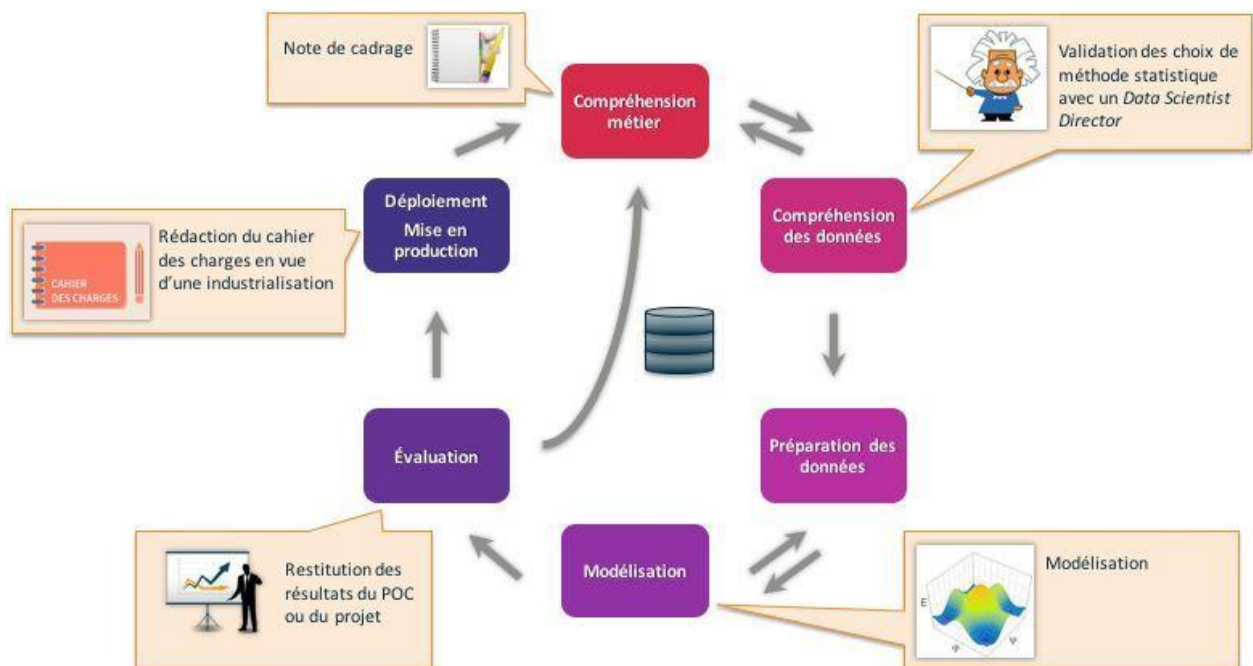


Figure 1.04 : *processus d'extraction de connaissance*

1.6.1 Compréhension du problème métier

La première étape consiste à bien comprendre les éléments métiers et problématiques que la Data Science vise à résoudre ou à améliorer.

1.6.2 Récupération des données

La deuxième étape consiste à collecter les données. Dans cette étape, on s'assure qu'on peut utiliser les données dans notre programme, ce qui signifie vérifier l'existence, la qualité et l'accès aux données. Les données peuvent également être fournies par des sociétés tierces et prennent de nombreuses formes allant des feuilles de calcul Excel à différents types de bases de données.

1.6.3 Préparation des données

Dans cette phase, on améliore la qualité de données et on les prépare pour une utilisation dans les étapes suivantes.

Cette phase comprend trois sous-phases :

- le nettoyage des données : supprime les fausses valeurs d'une source de données et les incohérences entre les sources de données.
- l'intégration des données : enrichit les sources de données en combinant les informations provenant de sources multiples.
- la transformation des données : garantit que les données sont dans un format à utiliser dans nos modèles.

1.6.4 Modélisation des données ou construction de modèles

C'est la phase de Data Science proprement dit. La modélisation comprend le choix, le paramétrage et le test de différents algorithmes ainsi que leur enchaînement, qui constitue un modèle. Ce processus est d'abord descriptif pour générer de la connaissance, en expliquant pourquoi les choses se sont passées. Il devient ensuite prédictif en expliquant ce qu'il va se passer, puis prescriptif en permettant d'optimiser une situation future.

1.6.5 Evaluations

L'évaluation vise à vérifier les modèles ou les connaissances obtenues afin de s'assurer qu'ils répondent aux objectifs formulés au début du processus. Elle contribue aussi à la décision de déploiement du modèle ou, si besoin est, à son amélioration. A ce stade, on teste notamment la robustesse et la précision des modèles obtenus.

1.6.6 Déploiement

Il s'agit de l'étape finale du processus. Elle consiste en une mise en production pour les utilisateurs finaux des modèles obtenus. Son objectif c'est de mettre la connaissance obtenue par

la modélisation dans une forme adaptée et l'intégrer au processus de prise de décision. Le déploiement peut ainsi aller, selon les objectifs, de la simple génération d'un rapport décrivant les connaissances obtenues jusqu'à la mise en place d'une application, permettant l'utilisation du modèle obtenu, pour la prédiction de valeurs inconnues d'un élément d'intérêt.

1.7 Data warehouse ou entrepôt de données

1.7.1 Présentation

Le concept d'entrepôt de données a été formalisé pour la première fois en 1990 par Bill Inmon. Il s'agissait de constituer une base de données orientée sujet, intégrée et contenant des informations historiques, non volatiles et exclusivement destinées aux processus d'aide à la décision.

Pour faire face aux nouveaux enjeux, l'entreprise doit collecter, traiter, analyser les informations de son environnement pour anticiper. Mais cette information produite par l'entreprise est surabondante, non organisée et éparpillée dans de multiples systèmes opérationnels hétérogènes et peut provenir de toutes les places de marchés (mondialisation des échanges).

Il devient fondamental de rassembler et d'homogénéiser les données afin de permettre l'analyse des indicateurs pertinents pour faciliter la prise de décisions.

L'objet de l'entrepôt de données est de définir et d'intégrer une architecture qui serve de fondation aux applications décisionnelles. [6][7]

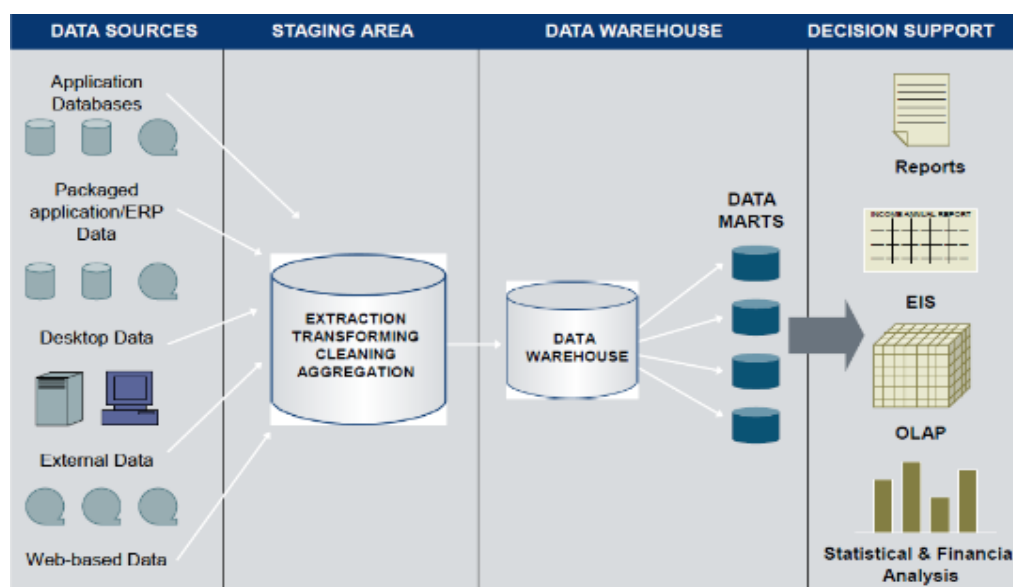


Figure 1.05 : *concept d'un entrepôt de données*

Data Mart ou Magasin de données : c'est un sous ensemble de l'entrepôt de données qui contient les données pour un secteur particulier de l'Entreprise.

OLAP ou On-Line Analytical Processing : il permet une analyse multidimensionnelle sur des bases de données volumineuses afin de mettre en évidence une analyse particulière de données (Cubes OLAP)

1.7.2 *Caractéristiques*

Un entreposage de données détient quelques caractéristiques particulières :

- Orientée sujet : un Entrepôt de données ou ED rassemble et organise des données associées aux différentes structures fonctionnelles de l'entreprise, pertinentes pour un sujet ou thème et nécessaire aux besoins d'analyse.
- Historiées : les données d'un ED représentent l'activité d'une entreprise durant une certaine période généralement sur plusieurs années permettant d'analyser les variations d'une donnée dans le temps.
- Intégrées : Les données, qui proviennent de diverses sources hétérogènes, sont consolidées et intégrées dans l'entrepôt.
- Non volatiles : une fois insérées dans l'entrepôt, les données ne sont jamais modifier ou effacées, elles sont conservés pour des analyse futures.

1.7.3 *Fonctions Data warehouse*

Il y a trois (3) fonctions essentielles dans le data Warehouse :

- Collecter des données de bases existantes et les charger,
- Gérer des données dans l'entrepôt,
- Analyser les données en vue de la prise de décision.

1.7.4 *Les classes des données*

Un entrepôt de données peut se structurer en quatre classes de données organisées selon un axe historique et un axe de synthèse.

1.7.4.1 Les données agrégées

Les données agrégées correspondent à des éléments d'analyse représentant les besoins des utilisateurs. Elles constituent déjà un résultat d'analyse et une synthèse de l'information contenue dans le système décisionnel, et doivent être facilement accessibles et compréhensibles.

1.7.4.2 Les données détaillées

Les données détaillées reflètent les événements les plus récents. Les intégrations régulières des données issues des systèmes de production vont habituellement être réalisées à ce niveau.

1.7.4.3 Les métadonnées

Les métadonnées constituent l'ensemble des données qui décrivent des règles ou processus attachés à d'autres données. Ces dernières constituent la finalité du système d'information.

1.7.4.4 Les données historiées

Chaque nouvelle insertion de données provenant du système de production ne détruit pas les anciennes valeurs, mais crée une nouvelle occurrence de la donnée.

1.7.5 *Avantages*

L'entreposage des données possède de multiples avantages comme par exemples :

- Permet de mener des analyses poussées sur différents sujets
- Fournit une vue consolidée des données en Entreprise
- Procure de l'information de qualité plus rapide
- Libère les ressources dédiées au traitement des transactions des tâches dédiées au traitement des transactions des tâches d'analyse
- Simplifie l'accès aux données

1.8 **Data mining ou fouille des données**

1.8.1 *Initiation*

Dans les mines d'or, ce n'est pas en faisant exploser les roches et en creusant des trous titanesques que l'on dénicher les plus grands trésors. Non, c'est en tamisant finement la matière, un processus complexe, où des masses de matériaux sans valeur sont éliminés pour ne garder que les parties précieuses. Le data mining fonctionne exactement de cette façon. Il s'agit de trouver les informations importantes et utiles dans les masses de données.

Avec l'énorme quantité de données stockées dans les fichiers, bases de données ou autres, il est de plus en plus important, voire même nécessaire, de développer de puissants moyens d'analyse, d'interprétation ainsi que d'extraction de connaissances efficace. [8][9]

1.8.2 Définition

La fouille de donnée est l'extraction non triviale (pas évident) d'informations implicites, précédemment inconnues et potentiellement utiles à partir d'ensemble de données de grande taille.

1.8.3 Fonctions data mining

La fouille des données permet de faire beaucoup de traitements. On va voir quelques-unes dans ce paragraphe.

1.8.3.1 Classification

Elle permet de prédire si une instance de données est membre d'un groupe ou d'une classe prédéfinie.

1.8.3.2 Clustering ou segmentation

Partitionnement logique de la base de données en clusters. Un cluster c'est un groupe d'instances ayant les mêmes caractéristiques.

1.8.3.3 Règles d'association

La règle d'association est utilisée lorsqu'on veut avoir la corrélation ou relation entre attribut. Il fait partie de la méthode non supervisée.

Exemple : achat riz + vin implique achat poisson

1.8.4 Utilisations

Le data mining s'applique dans de nombreux domaines dont nous allons voir certains d'entre eux :

- Retail ou commerce de détail : avec des analyses des achats effectués avec une carte de fidélité pour ensuite baser les résultats des stratégies pour les réductions et la stimulation des achats.
- Pouvoirs publics : avec de nouvelles façons de communiquer avec les citoyens pour détecter les infractions, comme le blanchiment d'argent.
- Finance : avec une meilleure détection des activités frauduleuse et de meilleures décisions de crédit sur la base des données historiques des clients.
- Production : en configurant des paramètres pour l'optimisation des environnements de production, pouvant être copiés d'une usine à l'autre.

1.9 Text mining

1.9.1 Définition

Text Mining est la découverte par ordinateur de nouvelles informations inconnues, en extrayant automatiquement des informations à partir de différentes ressources écrites. Un élément clé est la liaison entre les informations extraites ensembles pour former de nouveaux faits ou de nouvelles hypothèses à explorer plus loin par des moyens d'expérimentation plus conventionnels. [10]

1.9.2 Applications

Les applications typiques de l'exploration de texte peuvent inclure l'analyse des réponses aux sondages ouverts. Par exemple, on peut découvrir un certain nombre de mots ou de termes couramment utilisés par les répondants pour décrire les avantages et les inconvénients d'un produit ou d'un service (sous enquête), suggérant des idées fausses ou des confusions concernant les éléments de l'étude.

Une autre application consiste à aider à la classification automatique des textes. Par exemple, il est possible de filtrer automatiquement la plupart des courriers indésirables sur la base de certains termes ou mots qui ne sont pas susceptibles d'apparaître dans des messages légitimes. De cette manière, ces messages peuvent être automatiquement supprimés. De tels systèmes automatiques de classification des messages électroniques peuvent également être utiles dans des applications où les messages doivent être acheminés (automatiquement) au département ou à l'agence le plus approprié. Dans le même temps, les messages électroniques sont filtrés pour des messages inappropriés ou obscènes, qui sont automatiquement renvoyés à l'expéditeur avec une demande de suppression des mots ou du contenu incriminés. [11]

1.9.3 Etapes de l'algorithme de text mining

Il existe trois étapes pour un algorithme de text mining :

- **Train** : crée un dictionnaire d'attribut où l'attribut représente des mots d'articles liés à un sujet particulier. Choisissez uniquement les mots qui se produisent un nombre minimum de fois.
- **Filtre** : cette étape consiste à supprimer les mots communs connus qui peut être inutiles dans les articles de différenciation.
- **Classification** : Vérifiez chaque document à classer pour la présence et la fréquence des attributs choisis.

1.10 Web mining

1.10.1 Définition

Le web mining, parfois francisé en fouille du web, consiste en l'exploration de données techniques (volumineuses) à très grande échelle pour découvrir des modèles de l'Internet et de tout site web. Selon les objectifs d'analyse, l'exploration du Web peut être divisée en trois types, qui sont l'utilisation de recherches du Web, l'extraction de contenus Web et de structure d'exploration Web. [12]

1.10.2 Utilisation de web mining

L'exploration par le Web est le processus qui permet de savoir ce que les utilisateurs recherchent sur Internet. Certains utilisateurs peuvent ne regarder que des données textuelles alors que d'autres pourraient vouloir obtenir des données multimédia. L'exploration de l'utilisation du Web permet également de trouver le modèle de recherche pour un groupe particulier de personnes appartenant à une région particulière.

1.10.3 Applications

Il est utilisé dans l'extraction des informations utiles de l'analyse dans le journal du serveur Web contenant les détails des pages Web visités et des transactions.

L'analyseur de journal de serveur Web peut inclure des logiciels tels que NetTracker, AwStats pour voir à quelle fréquence le site Web est visité, quel type de produit est le meilleur et les pires vendeurs dans un site Web de commerce électronique par exemple. La possibilité de suivre le comportement de navigation des utilisateurs Web jusqu'à des clics de souris individuels permet de personnaliser les services pour les clients individuels à grande échelle. Cette personnalisation de masse des services aide non seulement les clients à satisfaire leurs besoins, mais aussi à fidéliser la clientèle. Grâce à une approche plus personnalisée et centrée sur le client, le contenu et la structure d'un site Web peuvent être évalués et adaptés aux préférences du client et les bonnes offres peuvent être faites au bon client.

1.10.4 Avantages

Le Web mining offre plusieurs avantages à l'Entreprise qui l'utilise. Par exemples, les entreprises peuvent établir une meilleure relation client en leur proposant exactement ce dont ils ont besoin. Les entreprises peuvent, consécutivement, trouver, attirer et fidéliser les clients, en économisant

ainsi sur les coûts de production en utilisant la connaissance acquise des besoins réels des clients. Ils peuvent accroître leurs rentabilités en ciblant les différents utilisateurs basés sur les profils créés. Le Web mining peuvent même trouver les clients qui fait défaut à un concurrent de l'entreprise et va essayer de garder leurs clients en fournissant des offres promotionnelles plus spécifique, réduisant ainsi le risque d'attrition des clients.

1.11 Le Big data ou la science des données ?

Le terme « big data » ou données massives en français est devenu extrêmement commun dans le vocabulaire quotidien des Entreprises. Le « big data » désigne l'ensemble des techniques permettant l'exploitation et l'utilisation de très gros volumes de données.

La science des données mélange la modélisation mathématique, statistique ainsi qu'informatique et s'applique donc aux données en général, pas spécifiquement au big data.

Par contre, des nombreux modèles complexes nécessitent une grande quantité de données afin de révéler leur potentiel.

1.12 Conclusion

Dans ce chapitre, on a vu la science des données et, plus particulièrement, les différentes techniques et processus d'extraction des connaissances. La science des données possède une large variété de domaine d'applications telles que la statistique, l'informatique ainsi que les mathématiques. En outre, elle participe à l'amélioration des processus de prise de décision en les rationalisant sur la base d'éléments qualitatifs. Dans le prochain chapitre, on va parler de Machine Learning ou apprentissage automatique qui présente un des moyens pour résoudre un problème en science des données.

CHAPITRE 2

LE MACHINE LEARNING

2.1 Introduction

L'expression « Machine Learning », ou en français, apprentissage automatique connaît actuellement un essor très important, tant dans le monde des entreprises qu'auprès du grand public. L'apprentissage automatique fait partie des champs d'étude de l'intelligence artificielle et se corrélait avec d'autre domaine comme les données massives ou big data. Dans ce chapitre, on va explorer le monde de l'apprentissage automatique et voir comment un ordinateur apprend à partir des données qui leurs sont disponibles.

2.2 Définitions

L'apprentissage automatique est une application de l'intelligence artificielle (IA) consistant à faire réagir les ordinateurs, d'apprendre et d'améliorer automatiquement l'expérience sans être explicitement programmée. Il se concentre sur le développement de programmes informatiques capables d'accéder aux données et de les utiliser pour apprendre par eux-mêmes. [14]

On dit qu'un programme informatique apprend de l'expérience E par rapport à une tâche T et à une mesure de performance P , si sa performance sur T , mesurée par P , s'améliore avec l'expérience E .

2.3 Concept général de l'apprentissage automatique

Le principal objectif de l'apprentissage automatique est de développer des modèles (algorithmes) d'apprentissage pour acquérir des connaissances à partir des données afin de faire des prédictions. Plutôt que de demander un humain de façonner manuellement les règles et de construire des modèles à partir de l'analyse de grandes quantités de données, elle offre une alternative plus efficace pour extraire les connaissances dans les données et améliorer les performances des modèles prédictifs. [15]

2.4 Les bases de la technologie de Machine Learning

2.4.1 Illustration

Les bases de la Machine Learning ou ML implique des données d'entrée, la phase d'apprentissage ou « learning », et des données de sortie comme illustré dans la figure 2.01.

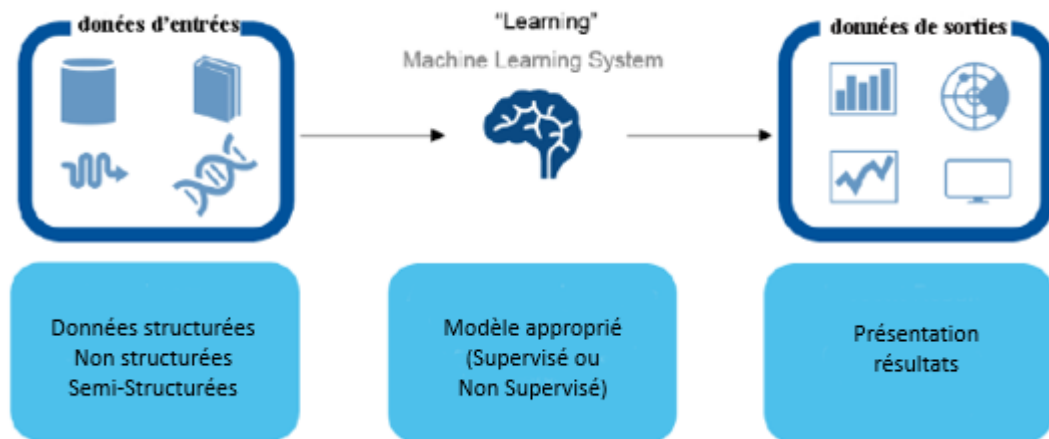


Figure 2.01 : *processus d'extraction de connaissance*

2.4.2 Données d'entrée

Une grande variété de données peut être utilisée comme entrée pour les besoins de l'apprentissage automatique. Par contre, la plupart du temps, ce sont des données brutes. Ainsi, le prétraitement des données est l'une des étapes les plus cruciales de toute application d'apprentissage automatique.

Ces données peuvent provenir de diverses sources, telles que dans le CRM (Customer Relationship Management) d'une entreprise, des bases de données mainframe ou des périphériques IoT (Internet Of Things) et les différents réseaux sociaux comme facebook, twitter, etc. De très gros volumes de données sont souvent transmis aux Machine Learning, parce que plus de données donnent souvent plus d'informations donc plus de précision sur l'analyse. Ceci est intensifié par l'ère des données massives ou big data, où les sources et les volumes d'informations explosent.

2.4.3 *Learning*

Cette phase de l'apprentissage automatique consiste aux processus d'apprentissage lui-même et il se subdivise en quelques étapes très importantes :

- Formation et sélection d'un modèle prédictif
- Evaluation du modèle

2.4.3.1 Formation et sélection d'un modèle prédictif

Le choix des modèles prédictifs est basé sur plusieurs critères tels que la nature des données à prédire, leur quantité et ce que l'on souhaite montrer à son travers. Une bonne connaissance des algorithmes d'apprentissage automatique permet de faire un pré-tri, mais le choix final des Data Scientists ne se fera qu'après tests et calculs (évaluation) de performance pour définir quel modèle se prête le mieux à une situation donnée.

2.4.3.2 Evaluation du modèle

Après avoir sélectionné un modèle qui a été inséré dans l'ensemble de données d'apprentissage, on peut utiliser l'ensemble de données de test pour estimer la qualité de son exécution sur ces données afin d'estimer l'erreur de généralisation. Si on est satisfait de la performance de notre modèle, on peut maintenant l'utiliser pour prédire des nouvelles données futures.

2.4.4 *Données de sortie*

Une machine learning peut être utilisée pour fournir des résultats prédictifs (c'est-à-dire fournir des prévisions) ou prescriptifs (c'est-à-dire suggérant des actions recommandées). Les résultats peuvent également fournir des résultats qui classent l'information ou mettent en évidence les zones à explorer. Ces données de sortie peuvent être stockées pour une nouvelle analyse, livrées sous forme de rapports.

2.5 Types d'apprentissage automatique

Dans cette section, nous examinerons les trois types d'apprentissage automatique : [16][17]

- l'apprentissage supervisé,
- l'apprentissage non supervisé,
- l'apprentissage par renforcement.

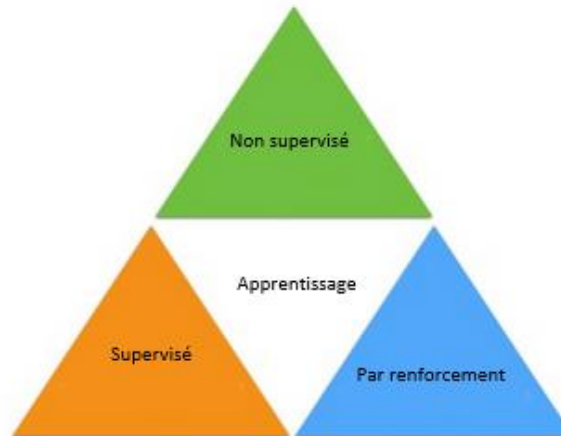


Figure 2.02 : *les types d'apprentissage automatique*

2.5.1 Apprentissage supervisé

Le principal objectif de l'apprentissage supervisé est d'apprendre un modèle à partir de données d'apprentissage étiquetées. Les algorithmes d'apprentissage automatique supervisés peuvent appliquer ce qui a été appris dans le passé à de nouvelles données en utilisant des exemples étiquetés ou label pour prédire des événements futurs. A partir de l'analyse d'un ensemble de données d'apprentissage connu, il produit une fonction déduite pour effectuer des prédictions sur les valeurs de sortie.

Ici, le terme supervisé se réfère à un ensemble d'échantillons où les signaux de sortie désirés (étiquettes ou labels) sont déjà connus.

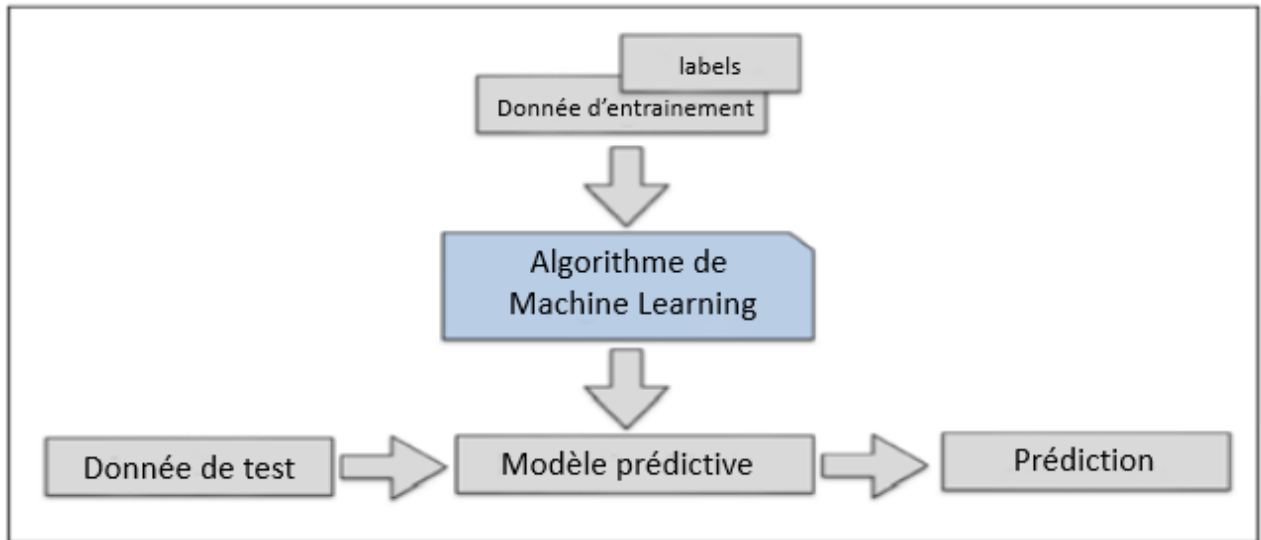


Figure 2.03 : *apprentissage supervisé*

On l'utilise souvent dans :

- La détection de courriers indésirables ou spam
- La reconnaissance de caractères manuscrits OCR.
- La reconnaissance vocale.

Considérant l'exemple du filtrage du spam par e-mail, nous pouvons former un modèle en utilisant un algorithme d'apprentissage automatique supervisé sur un corpus d'e-mails étiquetés, e-mail correctement marqués comme spam ou non-spam, pour prédire si un nouvel e-mail appartient à l'une ou l'autre des deux catégories.

Une tâche d'apprentissage supervisé avec des étiquettes de classes discrètes, est appelée tâche de classification. Une autre sous-catégorie de l'apprentissage supervisé est la régression, où le signal de résultat est une valeur continue.

Continue	Discret
Régression	Classification ou catégorisation

Tableau 2.01 : *les problèmes en apprentissage supervisé*

2.5.2 Apprentissage non supervisé

L'apprentissage automatique non supervisé est utilisé lorsque l'information utilisée pour l'apprentissage n'est ni classée ni étiquetée. L'algorithme doit découvrir par lui-même la structure plus ou moins cachée des données. L'apprentissage non supervisé est très souvent synonyme de clustering ou segmentation.

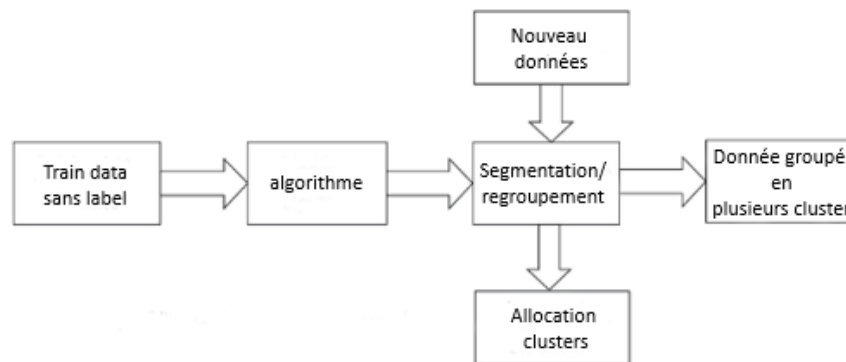


Figure 2.04 : *apprentissage non supervisé*

Continue	Discret
Réduction de dimension	clustering

Tableau 2.02 : *les problèmes en apprentissage non supervisé*

Applications de l'apprentissage non supervisé :

- Segmentation ou encore regroupement : partitionner les données en sous-groupes ou clusters, de manière non supervisée. Intuitivement, ces sous-groupes regroupent entre eux les observations similaires.
- réductions de dimension : réduire les nombres de dimension ou variables importants.
- Règle d'association : analyser les relations entre les variables ou détecter des associations.

2.5.3 Apprentissage par renforcement

Les algorithmes d'apprentissage par renforcement sont une méthode d'apprentissage qui interagit avec son environnement en produisant des actions et en découvrant des erreurs ou des récompenses. La recherche par essais/erreurs/récompenses différées est les caractéristiques les plus appropriées de l'apprentissage par renforcement. Cette méthode permet aux machines de déterminer automatiquement le comportement idéal dans un contexte spécifique afin de maximiser ses performances.

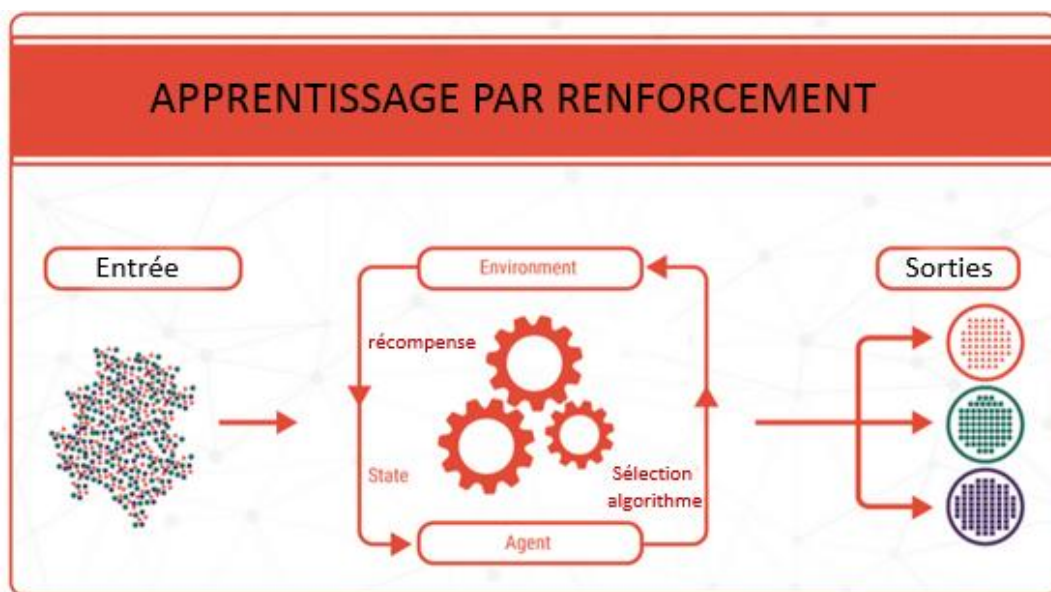


Figure 2.05: *apprentissage par renforcement*

Cette technique est très utilisée dans plusieurs domaines comme :

- Les Banques (trading, détection de fraudes, ...)
- Les jeux-vidéos
- Le pilotage automatisé de machines (voitures, hélicoptères,...)
- La robotique

2.6 Présentation de quelques algorithmes d'apprentissage automatique

Dans cette partie, on va voir quelques algorithmes d'apprentissage automatique les plus utilisés selon les différentes situations et problèmes.

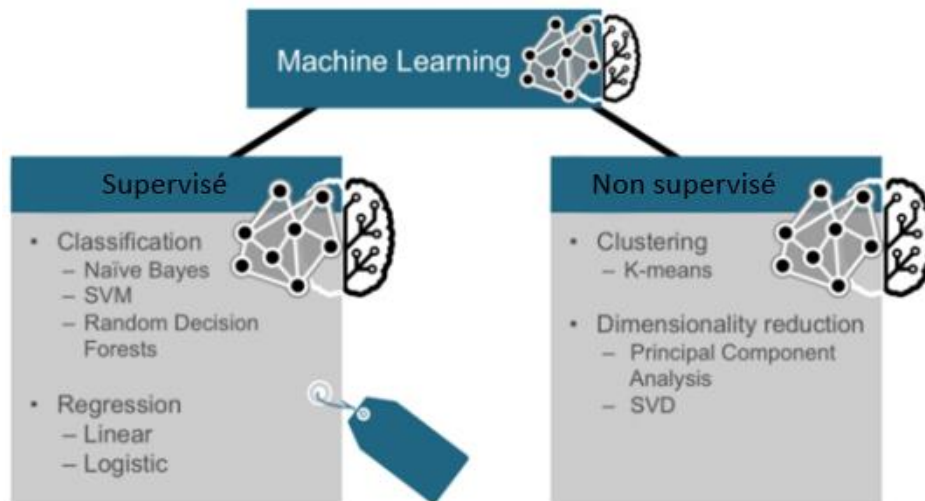


Figure 2.06 : *algorithmes d'apprentissage automatique*

2.6.1 *K-plus proche voisin*

Le k-plus proche voisin ou KNN est un exemple typique d'un algorithme d'apprentissage paresseux. Il est appelé paresseux pas à cause de sa simplicité apparente, mais parce qu'il n'apprend pas une fonction discriminante à partir des données d'apprentissage mais mémorise l'ensemble de données d'apprentissage à la place.

L'algorithme KNN est assez simple et peut être résumé par les étapes suivantes :

- Choisissez le nombre de k et une métrique de distance.
- Trouvez les k voisins les plus proches de l'échantillon que nous voulons classer.
- Attribuez l'étiquette du cours à la majorité des voix.

La figure 2.07 illustre comment un nouveau point (?) de données est affecté à la classe de label triangle basé sur le vote à la majorité parmi ses cinq plus proches voisins.

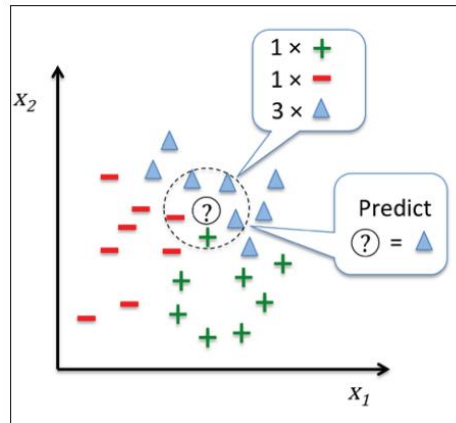


Figure 2.07 : *illustration algorithme de KNN*

Basé sur la métrique de distance choisie, l'algorithme KNN trouve les k échantillons dans l'ensemble de données d'apprentissage qui est le plus proche (le plus semblable) au point que nous voulons classer.

L'étiquette de classe du nouveau point de données est ensuite déterminée par un vote majoritaire parmi ses k plus proches voisins.

Le principal avantage d'une telle approche basée sur la mémoire est que le classificateur s'adapte immédiatement à la collecte de nouvelles données d'entraînement.

2.6.2 L'arbre de décision

2.6.2.1 Structure d'un arbre

- Nœud interne (nœud de décision) : étiquetés par des tests applicables à toute description d'une instance. Généralement, un nœud interne est égal à un test sur un unique attribut.
- Arcs issus d'un nœud interne : réponse possible au test du nœud.
- Feuille de l'arbre : étiquetées par une classe.
- chaque nœud interne ou feuille est repéré par sa position (liste des numéros des arcs qui permettent d'y accéder en partant de la racine).

2.6.2.2 Principe général

Les principaux algorithmes par arbre de décision sont CART et C4.5. Les arbres de décision sont un modèle d'apprentissage classique et naturel. Elles sont étroitement liées à la notion fondamentale de «diviser pour mieux régner». Il s'agit d'une méthode itérative, dite de partitionnement récursif des données. Effectivement, la méthode construit des classes d'individus, les plus homogènes possibles, en posant une succession de questions binaires de type booléenne (oui/non) sur les attributs de chaque individu. [18][19]

2.6.2.3 Construction de l'arbre

a. Mesure de la pureté des feuilles

Lorsque l'on se situe à un nœud donné de l'arbre, l'objectif est de créer deux feuilles qui soient plus homogènes que le nœud qui les précède. C'est pourquoi, Il faut disposer d'un moyen de mesurer cette homogénéité, ou pureté. Grâce à cela, à chaque nœud, le split est construit de manière à maximiser le gain d'information apporté par une question donnée sur la connaissance de la variable réponse.

On peut rencontrer plusieurs méthodes pour savoir l'attribut qui a le meilleur discriminant dont nous allons voir certaines d'entre eux dans ce paragraphe:

- Entropie probabiliste : L'entropie est souvent décrite comme une mesure du désordre ou d'incertitude par exemple pour un symbole, il prend la valeur nulle lorsque qu'il n'y a pas d'incertitude.

On note :

$$H_b(X) = - \sum_{i=1}^n p_i \log_b(p_i) \quad (2.01)$$

Avec :

X : variable aléatoire discrète prenant n valeur $x_1 \dots x_n$ de probabilité d'obtention respectives $p_1 \dots p_n$

b : base du logarithme (très souvent b=2)

- Indice de Gini : L'indice ou coefficient de Gini est une mesure, comprise entre 0 et 1, de la dispersion d'une distribution. Il est très souvent utilisé en économie ou en sociologie afin

de mesurer les inégalités sociales au sein d'un pays. Dans ce contexte, plus le coefficient est proche de 1 et plus la société est inégalitaire.

On note :

$$Gini(s) = \sum_{i=1}^k \frac{|S_i|}{|S|} \left(1 - \frac{|S_i|}{|S|}\right) = \sum_{i \neq j} \frac{|S_i| |S_j|}{|S|^2} \quad (2.02)$$

Avec :

S représente l'échantillon

$|S_i|$ Cardinal de S_i

b. Algorithme de construction

Pour la construction d'un arbre de décision, trois opérateurs majeurs sont nécessaires. D'abord, décider si un nœud est terminal. Ensuite, si un nœud n'est pas un terminal on lui associe un test. Enfin, s'il est un terminal alors on lui associe une classe.

Algorithme d'apprentissage générique
entrée : échantillon S
début
 Initialiser l'arbre courant à l'arbre vide ; la racine est le nœud courant
répéter
 Décider si le nœud courant est terminal
 Si le nœud est terminal **alors**
 Lui affecter une classe
 sinon
 Sélectionner un test et créer autant de nouveaux nœuds fils
 qu'il y a de réponses possibles au test
 FinSi
 Passer au nœud suivant non exploré s'il en existe
Jusqu'à obtenir un arbre de décision
fin

Figure 2.08 : L'algorithme de l'apprentissage générique

2.6.2.4 Avantages

La popularité de l'arbre de décision se justifie par les raisons suivantes :

- Il propose une décision aisément interprétable par rapport aux autres méthodes.
- De nouvelles options peuvent être ajoutées aux arbres existants et ils permettent de sélectionner l'option la plus appropriée parmi plusieurs.
- Avec l'arbre de décision, on a l'assurance en termes de vitesse lors d'une classification.
- Il offre la facilité d'association à d'autres outils de prise de décision.

2.6.3 *Random forest*

L'algorithme de Random forest ou forêt aléatoire est comme un algorithme d'amorçage avec le modèle CART de l'arbre de décision. La forêt aléatoire tente de construire plusieurs modèles CART avec différents échantillons et différentes variables initiales. La construction de chaque arbre en classification se base sur deux critères :

- Critère de Gini
- Critère entropie

Les arbres créés sont ensuite assemblés. La prédiction pour de nouvelles données est une moyenne (si régression) ou un vote (si classification). [18][20]

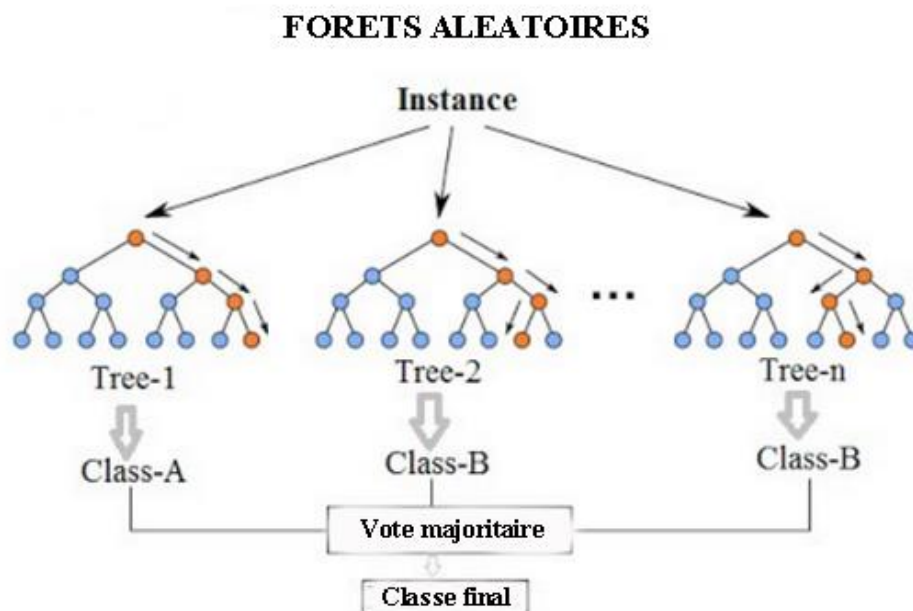


Figure 2.09: *Algorithme de forêt aléatoire simplifié*

2.6.4 Naïve Bayes

2.6.4.1 Introduction

Les classificateurs bayésiens sont des classificateurs statistiques. Ils peuvent prédire les probabilités d'appartenance à une classe, telles que la probabilité qu'un échantillon donné appartienne à une classe particulière. Le classificateur bayésien est basé sur le théorème de Bayes.

Les classificateurs bayésiens naïfs supposent que l'effet d'une valeur d'attribut sur une classe donnée est indépendant des valeurs des autres attributs. Cette hypothèse est appelée indépendance conditionnelle de classe. Il est fait pour simplifier le calcul impliqué et, dans ce sens, est considéré comme naïf.

2.6.4.2 Fondement

L'outil fondamental utilisé par la méthode bayésienne pour traiter l'information est la notion de probabilité. Etrangement, l'étude des probabilités est assez nouvelle en regard de son omniprésence dans les sciences et techniques modernes. Jusqu'au milieu du XVII^{ème} siècle, le terme probable, qualifiait une assertion démontrable, justifiable et n'impliquait pas de notion d'indéterminisme ou d'incertitude.

2.6.4.3 Principe

Dans le classificateur Naïve Bayes, on doit inférer des quantités gouvernées par des probabilités : on veut se servir de ces probabilités pour guider l'inférence. A chaque hypothèse on associe à une probabilité conditionnelle et la règle de Bayes. Cet algorithme vise alors à prédire le futur à partir du passé en supposant l'indépendance des attributs.

Bayésiennes : on estime la probabilité d'occurrence d'un événement sachant qu'une hypothèse préliminaire est vérifiée (connaissance). [18] [21]

2.6.4.4 Notion en probabilité

Dans ce paragraphe, on va voir quelques notions en probabilité nécessaire pour l'algorithme de naïve Bayes.

- La probabilité d'un événement A est noté $P(A)$
- Elle est comprise entre 0 et 1
- La probabilité d'un événement certain est 1 et celle d'impossible vaut 0.
- $P(A|B)$: Probabilité que l'événement A surviennent si l'événement B survient

- Théorème de Bayes :

$$P(A|B) = \frac{P(A \cap B)}{P(B)} \quad (2.03)$$

$$P(A \cap B) = P(A|B).P(B) = P(B|A).P(A) \quad (2.04)$$

2.6.4.5 Avantages et inconvénients

Avantages	Inconvénients
<ul style="list-style-type: none"> - L'algorithme de Naïve Bayes est très rapide pour la classification: parce que les calculs de probabilités ne sont pas très coûteux. - Il offre la possibilité de faire une classification même avec un petit jeu de données. 	<ul style="list-style-type: none"> - Il suppose l'indépendance des variables : c'est une hypothèse forte et qui est violée dans la majorité des cas réels. - Avec cet algorithme, il n'y a pas de sélection ou mise en évidence des variables pertinentes.

Tableau 2.03 : Avantages et inconvénients Naïve Bayes

2.6.5 Régression logistique

2.6.5.1 Définition

L'analyse de régression logistique étudie l'association entre une variable dépendante catégorielle et un ensemble de variables indépendantes (explicatives). Elle est utilisée lorsque la variable dépendante n'a que deux valeurs, telles que 0 et 1 ou Oui et Non. La régression logistique multinomiale est généralement réservée au cas où la variable dépendante a trois valeurs uniques ou plus, telles que Marié, Célibataire, Divorcé ou Veuf. Bien que le type de données utilisé pour la variable dépendante soit différent de celui de la régression multiple, l'utilisation pratique de la procédure est similaire. [18][22]

2.6.5.2 Utilisations

On parle de régression lorsqu'on cherche à prédire un attribut continu, c'est-à-dire le nombre de valeurs que l'attribut à prédire peut prendre est infini. Le type de régression le plus connu et le plus utilisé est la régression linéaire. D'un autre côté, lorsque l'attribut à prédire est discret, on parle de classification. Contrairement à ce que l'on pourrait penser, la régression logistique est une méthode de classification, au détail près qu'elle procède à une régression sur des attributs discrets. L'avantage est qu'en plus de renvoyer l'attribut prédit, l'algorithme renvoie un indicateur de confiance relatif à la prédiction.

2.6.5.3 Principe de fonctionnement

Supposons que l'on souhaite effectuer une classification binaire (valeur à prédire 0 et 1) à l'aide d'une régression logistique. Pour un échantillon donné, plutôt que de renvoyer une classe 0 ou 1, la régression logistique renvoie un résultat comprise entre 0 et 1 décrivant la probabilité d'appartenance de l'échantillon à la classe 1 ou à la classe 0. Afin d'approximer le résultat, on utilise une famille de fonctions particulières dont les valeurs sont toujours comprises entre 0 et 1 appelée la famille des fonctions logistiques.

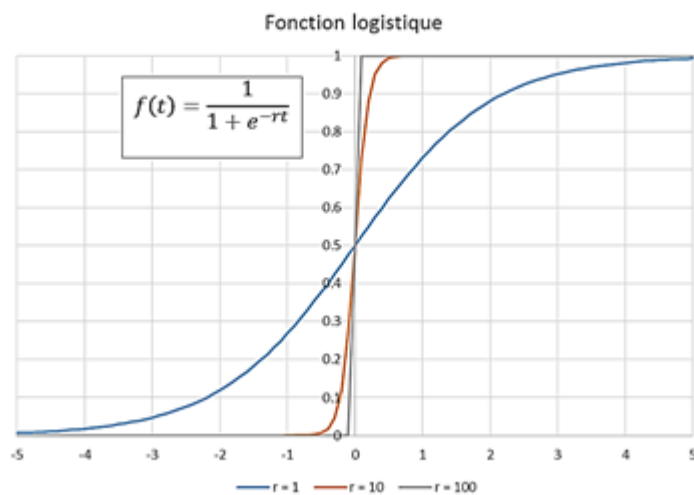


Figure 2.10 : *exemple d'un fonction logistique*

Effectuer l'entraînement consiste à trouver le paramètre r de la fonction logistique. En général, r est calculé par essais itératifs de sorte à maximiser la vraisemblance avec une méthode du gradient.

2.6.6 Support Vector Machine ou SVM

2.6.6.1 Définition

SVM est une famille d'algorithmes d'apprentissage automatique utilisés pour des problèmes mathématiques et d'ingénierie, notamment la reconnaissance de chiffres manuscrits, la reconnaissance d'objets, l'identification de locuteur, la détection de visages dans des images et la détection de cibles. C'est une méthode de classification binaire par apprentissage supervisé, introduit en 1995 par Vapnik et démocratisés à partir de 2000. [18][23]

2.6.6.2 Principe

L'algorithme se base principalement sur 3 astuces pour obtenir de très bonnes performances tant en qualité de prédiction qu'en complexité de calcul.

D'abord, on cherche l'hyperplan comme solution d'un problème d'optimisations sous-contraintes. La fonction à optimiser intègre un terme de qualité de prédiction et un terme de complexité du modèle. Ensuite, le passage à la recherche de la surface séparatrice non linéaire est introduit en utilisant un noyau kernel qui code une transformation non linéaire des données. Enfin, numériquement, toutes les équations s'obtiennent en fonction de certains produits scalaires utilisant le noyau et certains points de base de données (ce sont les supports vectors). [24]

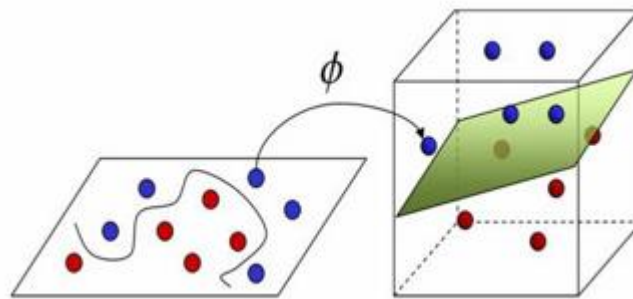


Figure 2.11 : *principe SVM*

2.6.7 Le boosting

On parle de boosting comme étant une méthode pour convertir des règles de prédiction peu performantes en une règle de prédiction très performante. En effet, plus précisément, étant donné un algorithme d'apprentissage faible qui peut toujours retourner une hypothèse de taux d'erreur $\leq 1/2 - \alpha$. Un algorithme de boosting peut construire de manière prouvée une règle de décision ou hypothèse de taux d'erreur $\leq \epsilon$

2.6.7.1 AdaBoost

Alors que le random forest construit plusieurs arbres en parallèle, le boosting construit lui aussi k arbres (ou d'autres algorithmes basiques), mais il le fait en série. L'arbre $k + 1$ aura accès à son prédécesseur, ou plus précisément à l'erreur de son prédécesseur. En conséquence, il sera pour ainsi dire un spécialiste de l'erreur passée et concentrera son effort sur la correction de ces erreurs désormais dévoilées. Pour un problème de classification, la prédiction n'est plus un vote à la majorité, mais une somme pondérée de chacun des algorithmes faibles.

2.6.7.2 Le gradient boosting

Le gradient boosting reprend les principes d'adaboost, mais les généralise à plusieurs fonctions de coût, quand adaboost n'en utilise qu'une seule. Cette généralisation est rendue possible par l'utilisation de la descente de gradient dans la construction itérative des algorithmes faibles.

2.6.8 *Algorithmes de clustering*

2.6.8.1 Illustration

Le clustering consiste à diviser la population ou les points de données en un certain nombre de groupes de sorte que les points de données dans les mêmes groupes soient plus semblables aux autres points de données du même groupe que ceux des autres groupes. En termes simples, le but est de séparer les groupes ayant des traits similaires et de les affecter en groupes.

Comprenons ceci avec un exemple. Supposons qu'on soit à la tête d'un magasin de location et que qu'on souhaite comprendre les préférences de vos clients pour développer votre activité. Est-il possible d'examiner les détails de chaque client et de concevoir une stratégie commerciale unique pour chacun d'entre eux ? Définitivement pas. Mais, ce qu'on peut faire c'est de regrouper tous nos clients en, disons, 10 groupes en fonction de leurs habitudes d'achat et d'utiliser une stratégie distincte pour les clients dans chacun de ces 10 groupes. Et c'est ce que nous appelons le clustering. [18][25]

2.6.8.2 Types de clustering

De manière générale, le clustering peut être divisé en deux (2) sous-groupes :

- **Hard clustering** : Dans le hard clustering, chaque point de données appartient complètement ou non à un cluster. Dans notre exemple plus haut, chaque client doit être placé dans l'un des dix (10) groupes.
- **Soft clustering** : chaque individu appartient à un cluster avec un certain degré de probabilité. En se basant toujours sur notre exemple plus haut, chaque client se voit attribuer une probabilité d'être dans l'un des 10 groupes de client du magasin.

2.6.8.3 K-Means clustering

k-means clustering est un algorithme d'apprentissage non supervisé, simple et populaire qui a son origine dans le traitement du signal. L'objectif de l'algorithme est de partitionner des exemples d'un ensemble de données en k clusters qui est fixé au préalable. Chaque exemple est un vecteur numérique qui permet de calculer la distance entre vecteurs en tant que distance euclidienne. Les clusters sont positionnées en tant que points et toutes les observations sont associées au cluster le plus proche, ensuite, calculés, puis ajustés. Le processus recommence en utilisant les nouveaux ajustements jusqu'à ce les centroïdes ne bouge plus de sa position.

L'exemple simple sur la figure 2.12 illustre la partition des données en $k = 2$ clusters.

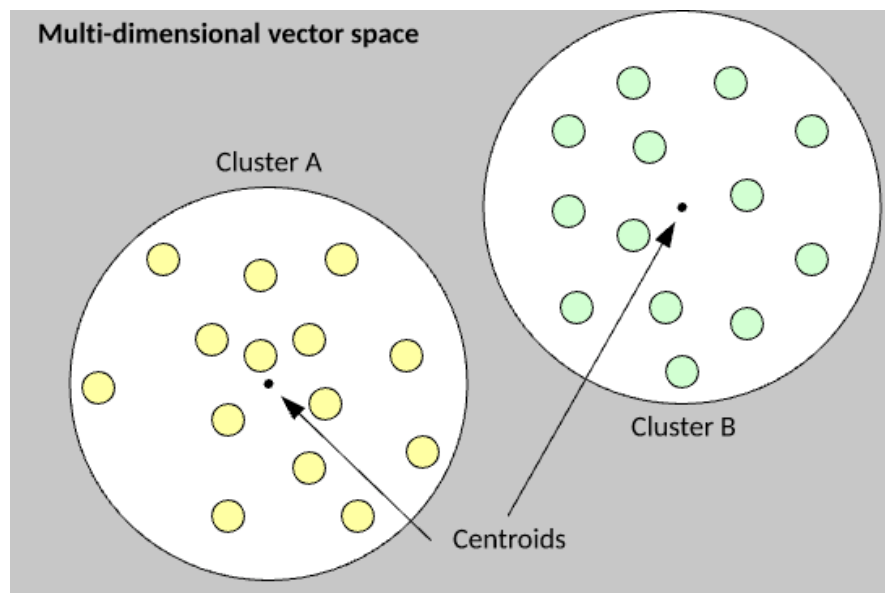


Figure 2.12 : Exemple de *k-means clustering*

2.6.8.4 Clustering hiérarchique

Le clustering hiérarchique permet de partitionner un jeu de données de manière hiérarchique. Il cherche à créer des clusters homogènes bien séparés par récurrence.

- Dans le cas du clustering *agglomératif* (ou *bottom-up*), on commence par considérer que chaque point est un cluster à lui tout seul. Ensuite, on trouve les deux clusters les plus proches, et on les réuni en un seul cluster. On répète cette étape jusqu'à ce que tous les points appartiennent à un seul cluster, constitué de l'agglomération de tous les clusters initiaux.

- L'approche inverse, le clustering *divisif* (ou *top-down*), consiste à initialiser avec un unique cluster contenant tous les points, puis à itérativement séparer chaque cluster en plusieurs, jusqu'à ce que chaque point appartienne à son propre cluster.

2.6.9 Deep learning

2.6.9.1 Aperçu

Le Deep learning ou l'apprentissage profond permet aux modèles de calcul composés de plusieurs couches de traitement, d'apprendre des représentations de données avec plusieurs niveaux d'abstraction. Les réseaux de neurones convolutifs ont apporté des avancées décisives dans le traitement des images, de la vidéo, de la parole et de l'audio, alors que les réseaux de neurones récurrents ont éclairé des données séquentielles telles que le texte et la parole, sur lesquels la communauté de chercheurs en intelligence artificielle s'est longtemps cassée la tête.

2.6.9.2 Initiation au Deep Learning

Concrètement, apprendre les coefficients du réseau neuronal revient à faire une extraction de caractéristiques de manière automatique sur les entrées. Et comme chaque couche du réseau neuronal est une représentation de plus haut niveau des entrées, superposer les couches permet d'avoir une représentation d'un niveau d'abstraction encore plus haut des entrées.

Une décision peut être prise en ajoutant une couche à un seul neurone, ce qui a pour effet de combiner l'effet de chaque composante de la dernière représentation des entrées. [26]

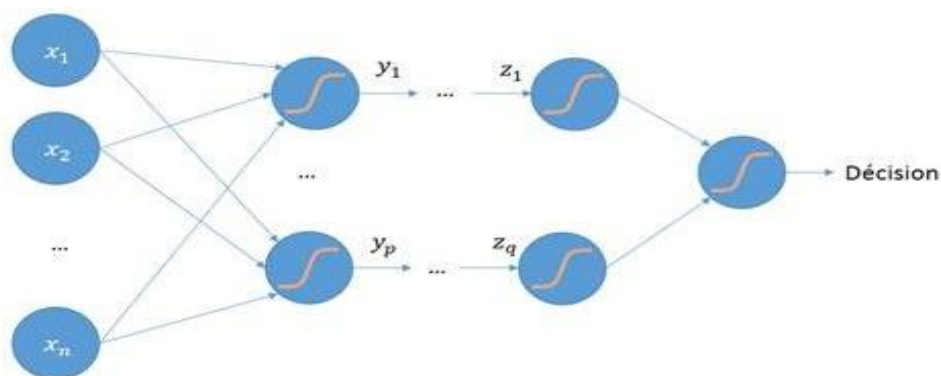


Figure 2.13 : *couche deep learning*

Les couches intermédiaires entre les entrées et le résultat sont appelées couches cachées. Augmenter le nombre de couches à nombre de neurones constants permet donc de représenter des fonctions plus complexes en utilisant moins de neurones.

2.6.9.3 Fonctionnement

Au sein du cerveau humain, chaque neurone reçoit environ 100 000 signaux électriques des autres neurones. Chaque neurone en activité peut produire un effet excitant ou inhibiteur sur ceux auxquels il est connecté. Au sein d'un réseau artificiel, le principe est similaire. Les signaux voyagent entre les neurones. Toutefois, au lieu d'un signal électrique, le réseau de neurones assigne un certain poids à différents neurones. Un neurone qui reçoit plus de charge exercera plus d'effet sur les neurones adjacents. La couche finale de neurones émet une réponse à ces signaux.

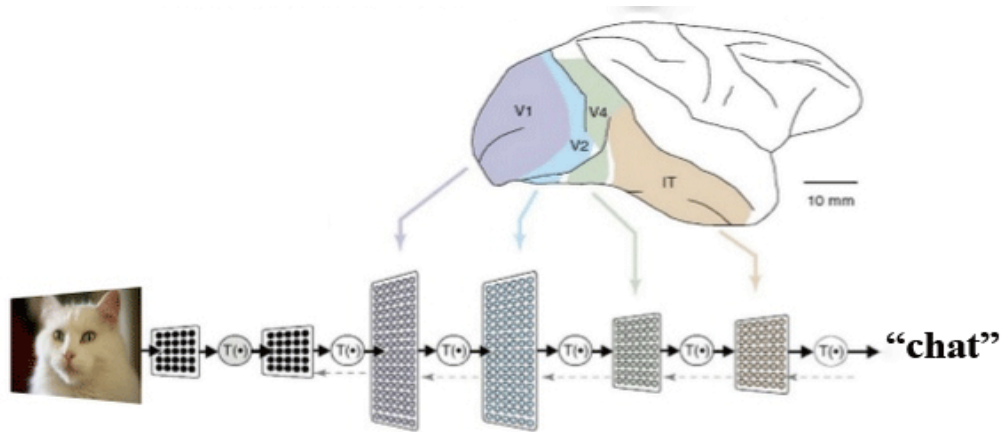


Figure 2.14 : *illustration deep learning*

Pour mieux comprendre le fonctionnement du Deep Learning, prenons un exemple concret de reconnaissance d'images (photos qui comportent au moins un chat). Pour pouvoir identifier les chats sur les photos, l'algorithme doit être en mesure de distinguer les différents types de chats, et de reconnaître un chat de manière précise quel que soit l'angle sous lequel il est photographié.

Afin d'y parvenir, le réseau de neurones doit être entraîné. Pour ce faire, il est nécessaire de compiler un ensemble d'images d'entraînement pour pratiquer le Deep Learning. Cet ensemble va regrouper des milliers de photos de chats différents, mélangés avec des images d'objets qui ne sont pas des chats. Ensuite, il faut les convertir en données et transférées sur le réseau. Les neurones artificiels assignent ensuite un poids aux différents éléments. La couche finale de neurones va alors rassembler les différentes informations pour déduire s'il s'agit ou non d'un chat. Le réseau de neurones va ensuite comparer cette réponse aux bonnes réponses indiquées par des personnes. Si les réponses correspondent, le réseau garde cette réussite en mémoire et s'en servira plus tard pour reconnaître les chats. Dans le cas contraire, le réseau prend note de son erreur et ajuste le poids placé sur les différents neurones pour corriger son erreur. Le processus est répété

des milliers de fois jusqu'à ce que le réseau soit capable de reconnaître un chat sur une photo dans toutes les circonstances. Cette technique d'apprentissage est dit supervisée.

Une autre technique d'apprentissage est celle de l'apprentissage non supervisée. Cette technique repose sur des données qui ne sont pas étiquetées. Les réseaux de neurones doivent reconnaître des patterns au sein des ensembles de données pour apprendre par eux-mêmes quels éléments d'une photo peuvent être pertinents. [27]

2.7 Evaluations modèle métriques

Bien que des études empiriques aient montré qu'il est difficile de décider quelle métrique utiliser pour différents problèmes, chacun d'entre eux à des caractéristiques spécifiques qui mesurent divers aspects des algorithmes à évaluer. [16][28]

2.7.1 Matrice de confusion

Une matrice de confusion sert à évaluer la qualité d'une classification. Le nombre de prédictions correctes et incorrectes est résumé avec les valeurs de comptage et ventilé par chaque classe. C'est la clé de la matrice de confusion. Elle est obtenue en comparant les données classées avec des données de référence qui doivent être différentes de celles ayant servi à réaliser la classification. Pour mesurer les performances de ce classificateur, il est d'usage de distinguer quatre (4) types d'éléments classés pour la classe voulue :

- Vrai Positif ou VP : éléments positif correctement prédit
- Vrai Négatif ou VN : éléments négatif correctement prédit
- Faux Positif ou FP : éléments négatif mal prédit
- Faux Négatif ou FN : éléments positif mal prédit

	prediction : TRUE	prediction : FALSE
réalité : TRUE	true positive	false negative
réalité : FALSE	false positive	true negative

Figure 2.15 :présentation de la matrice de confusion

2.7.2 Courbe ROC ou (Received Operating Characteristic)

Dans le cas d'un classificateur binaire, il est possible de visualiser ses performances sur ce que l'on appelle une courbe ROC. La courbe ROC est une représentation du taux de vrais positifs en fonction du taux de faux positifs. Son intérêt est de s'affranchir de la taille des données de test dans le cas où les données sont déséquilibrées. Cette représentation met en avant un nouvel indicateur qui est l'aire sous la courbe. Plus elle se rapproche de 1, plus le classificateur est performant.

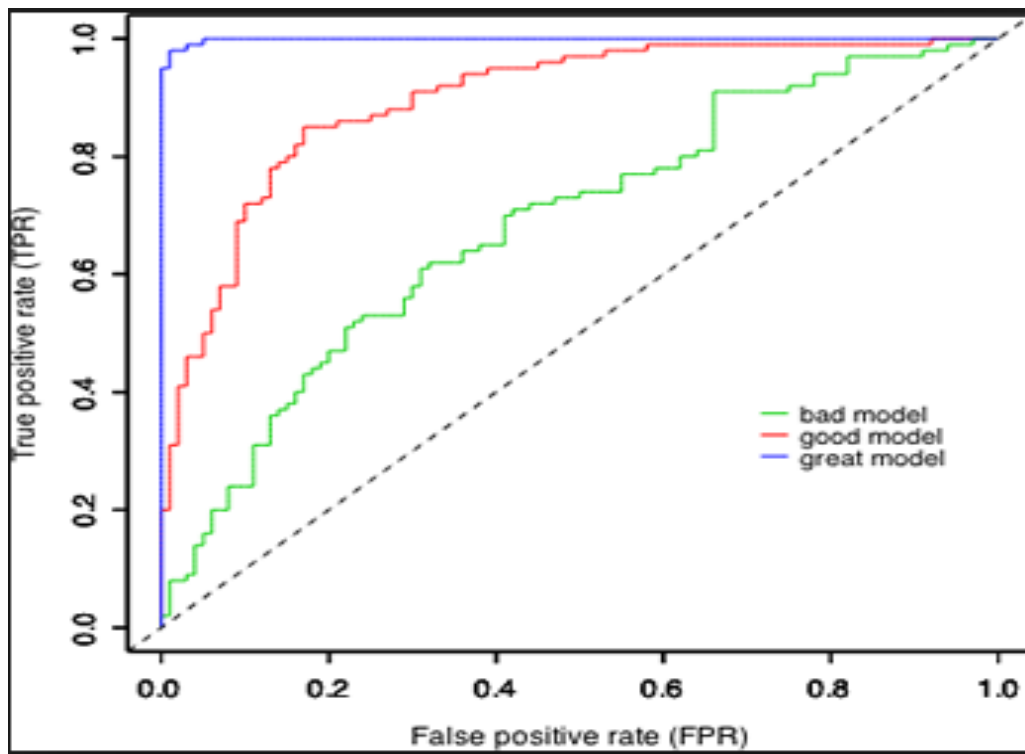


Figure 2.16 : présentation de la courbe ROC

2.7.3 Validation croisée

La validation croisée est un outil standard dans l'analyse et constitue une fonctionnalité importante pour nous aider à développer et affiner les modèles de machine learning.

Son principe de fonctionnement :

- On découpe le jeu de données en k parties ou folds à peu près égales. Tour à tour, chacune des k parties est utilisée comme jeu de test. Le reste (autrement dit, l'union des $k-1$ autres parties) est utilisé pour l'entraînement.
- A la fin, chaque observation a servi une fois dans un jeu de test, $(k-1)$ fois dans un jeu d'entraînement. On a donc une prédiction par observation de notre jeu initial, et aucune de ces prédictions n'a été faite avec un jeu d'entraînement qui contienne cette observation. On n'a donc pas violé le principe de ne pas valider le jeu d'entraînement.



Figure 2.17 :présentation de la validation croisée

2.8 Avantages de la Machine Learning

L'apprentissage automatique est un algorithme développé pour noter les changements dans les données et évoluer dans sa conception pour s'adapter aux nouvelles découvertes. Appliquée à l'analyse prédictive, cette fonctionnalité a un impact étendu sur les activités de développement des Entreprises.

L'apprentissage automatique a également un impact significatif sur le secteur financier. Parmi les avantages habituels de l'apprentissage automatique au sein de la Finance figurent la gestion de portfolio, le trading algorithmique, la souscription de prêts et plus important encore la détection de fraudes. Il facilite l'évaluation continue des données pour détecter et analyser les anomalies et les nuances. Cela aide à améliorer la précision des modèles et des règles financiers.

La segmentation de la clientèle et la prédiction sont les principaux défis rencontrés par les spécialistes du marketing aujourd'hui. Les unités de vente et de marketing disposeront d'énormes quantités de données pertinentes provenant de différents canaux, tels que, des visiteurs de sites Web et des campagnes par e-mail. Cependant, des prédictions précises pour des incitations et des offres de marketing individuelles peuvent être facilement réalisées avec la Machine Learning.

2.9 Les relations de Machine Learning avec d'autres domaine

En tant que domaine interdisciplinaire, l'apprentissage automatique partage des points communs avec les domaines mathématiques, statistique, théorie de l'information. C'est naturellement un sous-domaine de l'informatique, car notre objectif est de programmer les machines afin qu'elles apprennent. Dans un autre sens, l'apprentissage automatique peut être considéré comme une branche de l'intelligence artificielle, car après tout, la capacité de transformer l'expérience en est la pierre angulaire de l'intelligence humaine. Cependant, il convient de noter que, contrairement à l'Intelligence Artificiel traditionnelle, l'apprentissage automatique n'essaie pas de construire une imitation automatisée du comportement, mais plutôt d'utiliser les forces et les capacités spéciales des ordinateurs pour compléter l'intelligence humaine.

2.10 Conclusion

De nos jours, les techniques d'apprentissage automatique sont largement utilisées pour résoudre des problèmes réels, en stockant, en manipulant, en extrayant et en récupérant des données provenant des différentes sources. Des techniques d'apprentissage automatique supervisées ont été largement adoptées, mais ces techniques s'avèrent très coûteuses lorsque les systèmes sont mis en œuvre sur une large gamme de données. Ceci est dû au fait qu'une quantité importante d'effort et de coût est impliquée, pour l'obtention des données massive étiquetés.

CHAPITRE 3

LES OUTILS DE MACHINE LEARNING

3.1 Introduction

Au cours des dernières années, l'analyse des données devient de plus en plus primordiale pour les grandes Entreprises. De ce fait, de nombreuses technologies en analyse de données massives et d'apprentissage automatique sont apparues. Ce chapitre passe en revue différentes bibliothèques et plateformes d'apprentissage automatique. L'objectif est d'avoir un aperçu sur certains outils les plus utilisés par les data scientists pour résoudre les problèmes d'apprentissage automatique.

3.2 Tâche d'un outil d'apprentissage automatique

Il y a certains rôles ou tâches que doit remplir un outil d'apprentissage automatique. On va voir dans ce paragraphe quelques points essentiels.

3.2.1 *Trouver un pattern dans les données*

Cet aspect se concentre sur l'étude des amas de données qu'une entreprise a collectées auparavant dans le cadre de son activité. Le but étant de comprendre ces données, et les étudier en détail pour en tirer des informations utiles comme des tendances, des relations cachées entre les données et bien d'autre. L'objectif étant d'aider les décideurs à prendre des décisions stratégiques plus aguerries en se basant sur données factuelles.

3.2.2 *Développement d'un Data product*

Un Data Product est un logiciel qui se base sur des données comme entrée, et génère un résultat. Le résultat généré est calculé en se basant sur un modèle prédictif que le *Data Scientist* aura construit auparavant.

3.2.3 *Visualisation des données*

La visualisation des données est un moyen de représenter ces dernières de façon graphique et visuelle. Bien que les statistiques et l'analyse exploratoire des données soient utiles, elles ne sont, cependant, pas suffisantes. Visualiser les données permet de mieux les comprendre. En effet, le cerveau humain a une plus grande facilité à comprendre des concepts par des images. Exploiter cette faculté naturelle permettra une compréhension plus accrue des données. Il est important de connaître les techniques de base de visualisation des données. Notamment, les histogrammes, les scatter plot, les box plot. L'idée n'est pas de savoir les dessiner, mais de plutôt de savoir quand il

faut les utiliser, quelles sont leurs limites et surtout comment les interpréter. L'important est qu'à l'issue d'une étape de Data Visualisation, on ait des pistes et des hypothèses sur notre jeu de données qu'on vient de visualiser. [16][29]

3.2.3.1 Histogramme

Un histogramme est un graphique permettant de représenter la répartition des valeurs d'une variable continue. Chaque colonne de l'histogramme représente un intervalle de valeurs. La hauteur des colonnes indique le nombre d'instances dans cet intervalle. L'examen de l'histogramme permet de se faire une idée claire sur la distribution des valeurs de la caractéristique analysée.

3.2.3.2 Bar plot

Les Bar Plots sont utilisés pour visualiser des données qualitatives. Chaque Barre d'un bar plot représente une catégorie ou modalité et la hauteur de la barre indique la taille du groupe faisant partie de cette catégorie.

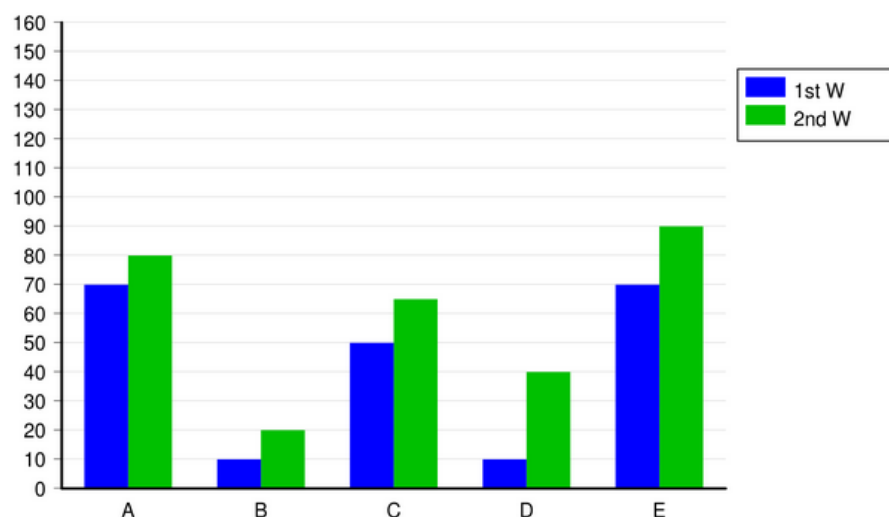


Figure 3.01 : *exemple de bar plot*

3.2.3.3 Box plot

Les Box plots ou boîtes à moustaches permettent de visualiser plusieurs paramètres statistiques d'une feature ou attribut. Notamment la médiane, l'écart interquartile ($IQR = \text{Interquartile Range}$) et la valeur maximale et minimale de la distribution. La représentation des informations des Box Plots est plus compacte que celle d'un histogramme, Toutefois, le degré de détails des

informations délivrées par un Box Plot est moindre. Nous pouvons avoir une idée de la tendance centrale des valeurs de chaque boîte en observant la position de la médiane. Si la médiane n'est pas au centre, on peut juger de la symétrie de la distribution (aplatissement et asymétrie). Par ailleurs, en se basant sur la longueur de la boîte, il est possible d'estimer la variabilité des valeurs pour chaque sous-groupe.

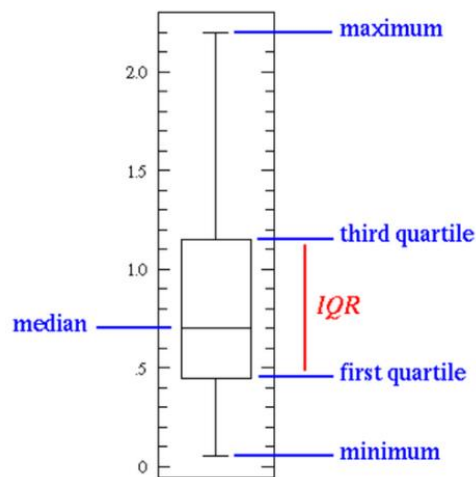


Figure 3.02 : *illustration box plot*

3.2.3.4 Scatterplot Matrices

Les Scatter Plot Matrices (Diagrammes de dispersion) permettent de visualiser la corrélation entre deux variables continues. On met le premier attribut sur l'axe des abscisses (X) et la deuxième sur les ordonnées (Y). La dispersion des points indique la relation entre les deux attributs.

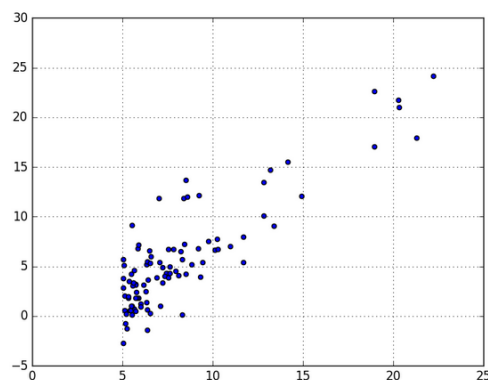


Figure 3.03 : *exemple scatterplot*

3.3 Machine learning avec MATLAB

On va commencer par l'apprentissage automatique avec le logiciel Matlab. Les produits MathWorks fournissent des algorithmes de haute qualité pour l'analyse des données ainsi que des outils graphiques pour visualiser les données. Les outils de visualisation sont un élément essentiel pour tout système d'apprentissage automatique. Ils peuvent être utilisés pour l'acquisition de données, par exemple, pour la reconnaissance d'images ou dans le cadre de systèmes de contrôle autonome de véhicules, ou pour le diagnostic et le débogage pendant le développement. Tous les paquets peuvent s'intégrer les uns aux autres et avec d'autres fonctions MATLAB pour produire des systèmes puissants pour l'apprentissage automatique. [30]

Les boîtes à outils les plus utilisées en matlab pour l'apprentissage automatique sont nombreuses, ceux qu'on va en parler sont les suivants :

- Statistique et machine learning
- Réseau de neurone

3.3.1 *Statistique et machine learning*

La boîte à outils Statistiques et Machine Learning fournit des méthodes d'analyse de données pour collecter des tendances et des modèles à partir de données massives. Ces méthodes ne nécessitent pas de modèle pour analyser les données. Ses fonctions peuvent être divisées en outils de classification, de régression et de classification.

3.3.1.1 Méthode de classification

Les méthodes de classification sont utilisées pour placer les données dans différentes catégories. Par exemple, des données, sous la forme d'une image, pourraient être utilisées pour classer une image d'un organe comme ayant une tumeur. La classification est utilisée pour la reconnaissance de l'écriture manuscrite, la notation de crédit et l'identification faciale. Les méthodes de classification comprennent les machines à vecteurs de support (SVM), les arbres de décision et les réseaux de neurones.

3.3.1.2 Méthode de régression

Les méthodes de régression nous permettent de créer des modèles à partir des données actuelles pour prédire les données futures. Les modèles peuvent ensuite être mis à jour lorsque de nouvelles données deviennent disponibles. Si les données ne sont utilisées qu'une seule fois pour créer le

modèle, il s'agit d'une méthode par lots. Une méthode de régression qui intègre les données à mesure qu'elles deviennent disponibles est une méthode récursive.

3.3.1.3 Méthode clustering

Le clustering trouve des regroupements naturels dans les données. La reconnaissance d'objet est une des applications de méthodes de clustering. Par exemple, si on veut trouver une voiture dans une image, on recherche des données associées à la partie d'une image qui est une voiture.

Alors que les voitures sont de différentes formes et tailles, ils ont de nombreuses caractéristiques en commun.

3.3.2 Réseau de neurone

Le MATLAB Neural Network Toolbox est une boîte à outils de réseau neuronal complète qui s'intègre parfaitement avec MATLAB. Il fournit des fonctions pour créer, former et simuler des réseaux de neurones. La boîte à outils comprend des réseaux de neurones convolutionnels et des réseaux d'apprentissage profond. Les réseaux de neurones peuvent être très intensifs en calcul en raison du grand nombre de nœuds et de poids associés, en particulier pendant l'entraînement. Neural Network Toolbox permet de distribuer des calculs sur des processeurs multi cœurs et des unités de traitement graphique si on a le « Parallel Computing Toolbox », un autre module MATLAB. On peut étendre encore plus loin à un cluster d'ordinateurs en réseau à l'aide de MATLAB Distributed Computing Server. Comme pour tous les produits MATLAB, Neural Network Toolbox offre de nombreuses fonctions graphiques et de visualisation qui facilitent la compréhension de vos résultats. La boîte à outils Neural Network est capable de gérer de grands ensembles de données. Cela peut être des giga-octets ou des téraoctets de données.

3.4 Machine learning avec R

3.4.1 Introduction

R, est un langage statistique puissant qui peut être utilisé pour manipuler et analyser des données. De plus, R fournit de nombreux modules d'apprentissage automatique et des fonctions de visualisation, qui permettent aux utilisateurs d'analyser les données très simplement. Plus important encore, R est open source et gratuit. [31]

3.4.2 Les paquets ou extensions

R possède des nombreuses extensions aussi appelés bibliothèques ou librairies en français et que l'on peut installer sous forme de paquet. L'opération peut se faire assez facilement dans R, mais il est possible qu'il y a quelque complication parfois par exemple au niveau de dépendance entre les paquets.

3.4.2.1 Utiliser un paquet

Pour pouvoir utiliser un paquet en R il faut faire la commande suivant :

```
library("nom_du_paquet")
```

Figure 3.04 : *importation paquet en R*

Par exemples : `library(tm)`, `library(XML)`

3.4.2.2 Installer un paquet

La plupart des paquets de R sont disponible à travers R. Lors de l'installation, R demande de choisir parmi un dépositaire de code.

```
install.packages("tm.plugin.webmining", dependencies = TRUE)  
install.packages("koRpus", dependencies = TRUE)
```

Figure 3.05 : *installation des paquets dans R*

3.4.3 Avantages de l'utilisation de R

L'utilisation de R simplifie grandement l'apprentissage automatique. Tout ce qu'on doit savoir c'est comment chaque algorithme peut résoudre le problème, et on peut simplement utiliser un paquet écrit pour générer rapidement des modèles de prédiction sur les données avec quelques lignes de commande. Par exemple, on peut effectuer Naïve Bayes pour le filtrage du courrier indésirable, effectuer des clusters k-means pour la segmentation client, utiliser la régression linéaire pour prévoir les prix des logements ou implémenter un modèle Markov caché pour prédire le marché boursier.

3.5 Machine learning avec Python

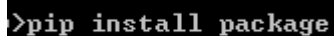
Maintenant que nous avons vu l'apprentissage automatique avec R, on va continuer sur notre lancée avec cet aperçu sur l'apprentissage en Python. Python est l'un des langages de programmation les plus populaires pour la science des données et bénéficie donc d'un grand nombre de bibliothèques complémentaires utiles développées par sa grande communauté. Dans ce paragraphe, nous récapitulons brièvement les principaux paquets les plus utilisés en machine learning de python. [32][33]

3.5.1 Installation de paquet en python

Pour installer Python efficacement avec toutes les principales librairies nécessaires, le mieux est de s'orienter vers Anaconda, qui est une distribution gratuite de Python incluant près de 200 packages et contenant des installeurs à la fois pour Python 2.x et 3.x.

Python est disponible pour les trois principaux systèmes d'exploitation : Microsoft Windows, Linux et Mac OS X. Il peut aussi être téléchargé et ensuite installé à partir de leur site officiel gratuitement (www.python.org).

Après avoir installé Python avec succès, nous pouvons exécuter pip qui est un installeur de python à partir du terminal de ligne de commande pour installer des paquets Python supplémentaires comme on peut voir sur la figure 3.06 :



```
>pip install package
```

Figure 3.06 : *installation d'un paquet sur python*

3.5.2 Numpy

La bibliothèque NumPy est un module complémentaire open source permettant d'effectuer des calculs mathématiques et numériques avec Python. Elle introduit une manipulation facilitée de grands tableaux ainsi que les matrices de données numériques.

Ses principales fonctionnalités sont les suivantes :

- Création et manipulation extrêmement performante des tableaux à n dimensions permettant des opérations arithmétiques vectorisées ;
- Opérations mathématiques rapides sur un tableau de données sans nécessité d'écrire des boucles ;
- Algèbre linéaire ;

- Possibilité d'intégration de code écrit en C, C++ et Fortran.

3.5.3 *Pandas*

3.5.3.1 Définition

Pandas est une bibliothèque Python open source pour l'analyse de données. Cela permet à Python de travailler avec des données de type tableur pour le chargement, la manipulation, l'alignement, la fusion de données.

Pour donner à Python ces fonctionnalités améliorées, Pandas introduit deux nouveaux types de données dans Python:

- Series
- DataFrame.

3.5.3.2 Series

La série est l'objet de la bibliothèque pandas conçue pour représenter des structures de données unidimensionnelles, de manière similaire à un tableau mais avec des fonctionnalités supplémentaires.

Sa structure interne est simple et est composée de deux tableaux associés les uns aux autres. Le tableau principal a pour but de contenir les données (données de tout type NumPy) auxquelles chaque élément est associé à une étiquette, contenue dans l'autre tableau, appelée Index.

Series	
index	value
0	12
1	-4
2	7
3	9

Figure 3.07 : *structure d'un objet Series*

Si on ne spécifie aucun index lors de la définition de la série, par défaut, Pandas attribue des valeurs numériques auto-incrémentées, à partir de 0 en tant que label. Dans ce cas, les labels correspondent à la position dans le tableau des éléments de l'objet Series.

3.5.3.3 DataFrame

Le DataFrame est une structure de données tabulaire très similaire à la feuille de calcul (les plus connues sont les feuilles de calcul Excel). Cette structure de données est conçue pour étendre le cas de la série à plusieurs dimensions.

En fait, le DataFrame consiste en une collection ordonnée de colonnes chacune pouvant contenir une valeur de type différent (numérique, chaîne, booléen, ...).

DataFrame			
index	columns		
	color	object	price
0	blue	ball	1.2
1	green	pen	1.0
2	yellow	pencil	0.6
3	red	paper	0.9
4	white	mug	1.7

Figure 3.08 : *structure d'un objet DataFrame*

3.5.3.4 Illustration

La richesse des fonctionnalités de la librairie pandas est une des raisons, si ce n'est la principale, d'utiliser Python pour extraire, préparer, éventuellement analyser, des données.

L'instruction suivante permet son importation :

```
>import Panda as pd
```

Figure 3.09 : *importation Pandas avec alias pd*

Pandas possède les fonctionnalités suivantes :

- Des structures de données avec des axes labélisés ;
- Manipulation de séries temporelles ;
- Gestion efficace des données manquantes ;
- Opérations relationnelles semblables au SQL.

3.5.4 Matplotlib et Seaborn

Nous savons tous que les images sont une forme de communication puissante. On l'utilise souvent pour mieux comprendre une situation ou pour condenser des informations dans une représentation graphique.

Matplotlib est un package Python pour le traçage 2D et 3D qui génère des graphiques de qualité. Il prend en charge le traçage interactif et non interactif et peut enregistrer des images dans plusieurs formats de sortie (PNG, PDF et bien d'autres). C'est une librairie hautement personnalisable, flexible et facile à utiliser. Il permet de faire des variétés de tracées comme de l'histogramme, des lignes simples, des dispersions ou scatter et des diagrammes. De l'autre côté, Seaborn est une bibliothèque de visualisation de python. Il est construit au-dessus de matplotlib et a comme avantages de donner de plus beau graphe et aussi d'être directement compatible avec la librairie pandas.

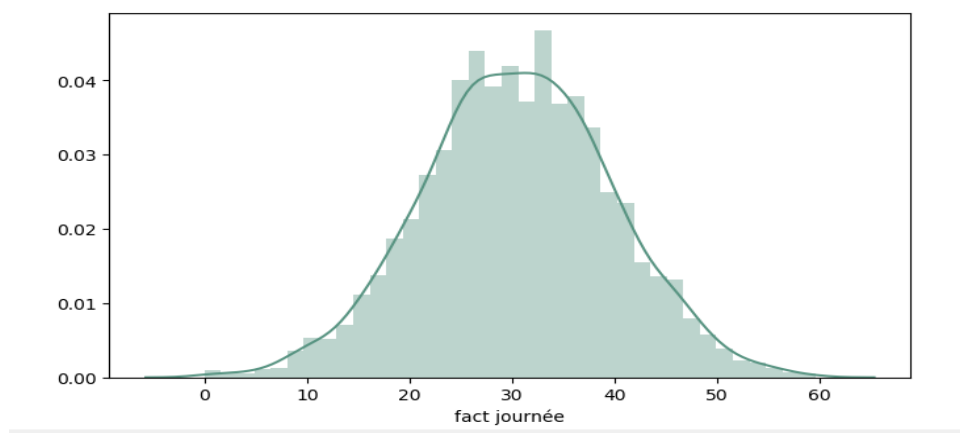


Figure 3.10 :*distribution de la facturation journée*

La double nature de Matplotlib lui permet d'être utilisé dans des scripts interactifs et non interactifs. Il peut être utilisé dans des scripts sans affichage graphique, intégrés dans des applications graphiques ou sur des pages Web.

3.5.5 *Scikit-Learn*

Depuis sa sortie en 2007, scikit-learn est devenu l'une des bibliothèques d'apprentissage automatique open source les plus populaires pour Python. scikit-learn fournit des algorithmes pour les tâches d'apprentissage automatique, y compris la classification, la régression, la réduction de dimension et la segmentation. Il fournit également des modules pour extraire des fonctionnalités, traiter des données et évaluer des modèles.

Scikit-learn inclut également une variété d'ensembles de données ou dataset, permettant aux développeurs de se concentrer sur des algorithmes plutôt que sur l'obtention et le nettoyage de données.

3.5.6 *Keras*

Keras est une bibliothèque Python minimaliste pour l'apprentissage en profondeur qui peut fonctionner sur Theano ou TensorFlow. Cette librairie a été développée pour rendre la mise en œuvre des modèles d'apprentissage en profondeur ou deep learning aussi rapide et facile que possible pour la recherche et le développement.

3.6 Machine learning avec Java

De nombreuses bibliothèques Java sont disponibles pour l'apprentissage automatique. Cela rend très pratique la possibilité de les interfacer dans des applications Java existantes et de tirer parti de puissantes capacités d'apprentissage automatique. [34]

3.6.1 *WEKA*

Weka, qui est l'abréviation de « Waikato Environment for Knowledge Analysis » est une bibliothèque d'apprentissage automatique développée à l'Université de Waikato, en Nouvelle-Zélande. Il est probablement la bibliothèque Java la plus connue. C'est une bibliothèque polyvalente capable de résoudre une grande variété de tâches d'apprentissage automatique, telles que la classification, la régression et la segmentation ou clustering. Il comporte une interface utilisateur graphique riche, une interface de ligne de commande et une API Java. [35]



Figure 3.11 : *logo officiel de WEKA*

3.6.2 *Java Machine Learning*

Le Java Machine Learning ou Java-ML, est une collection d'algorithmes d'apprentissage automatique avec une interface commune pour les algorithmes du même type. Il ne dispose que de fonctionnalités API Java, par conséquent, il est principalement destiné aux ingénieurs et aux programmeurs. Java-ML contient des algorithmes pour le prétraitement des données, la sélection des caractéristiques, la classification et le regroupement. En outre, il dispose de plusieurs ponts Weka pour accéder à Weka algorithmes directement à travers l'API Java-ML.[34]

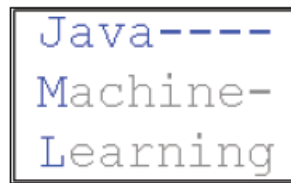


Figure 3.12 :*logo officiel de Java Machine Learning*

3.6.3 *Apache Mahout*

Le projet Apache Mahout vise à construire une bibliothèque d'apprentissage automatique évolutive. Il est construit sur des architectures distribuées évolutives, telles que Hadoop, en utilisant le paradigme MapReduce, qui est une approche pour le traitement et la génération de grands ensembles de données avec un algorithme distribué parallèle utilisant un groupe de serveurs.

Mahout intègre une interface de console et une API Java à des algorithmes évolutifs pour le clustering, la classification et le filtrage collaboratif. Il est aussi capable de résoudre plusieurs problèmes d'Entreprise tel que la recommandation, le regroupement et la classification. [36]



Figure 3.13 :*logo d'apache Mahout*

3.6.4 Comparaison librairie Java

Le tableau 3.01 nous résume et compare toutes les librairies d'apprentissage automatique en Java qu'on a parlées plus haut.

Nom librairie	domaines	License	architecture	algorithmes
Weka	Usage général	GNU GPL	Une seule machine	Arbre de décision, forêt aléatoire, réseau de neurone, naïve Bayes, clustering Hiérarchique
Java ML	Usage général	GNU GPL	Une seule machine	K-means Clustering, chaine de Markov, bagging, forêt aléatoire, arbre de décision
Apache Mahout	Classification, Système de recommandation et Clustering	Apache 2.0	Distribué, seule machine	naïve Bayes, forêt aléatoire, régression logique, K-means Clustering

Tableau 3.01 : comparaison bibliothèque Java

3.7 Machine learning avec Spark

Apache Spark, ou simplement Spark, est une plate-forme pour les constructions de traitement de données à grande échelle sur Hadoop, mais, contrairement à Mahout, elle n'est pas liée au paradigme MapReduce. Au lieu de cela, il utilise des caches en mémoire pour extraire un ensemble de données de travail, le traiter et répéter la requête.



Figure 3.14 : emblème de spark

Apache Spark permet à l'utilisateur de lire, transformer et agréger des données, ainsi que de former et de déployer des modèles statistiques sophistiqués avec facilité. Les API Spark sont accessibles en Java, Scala, Python, R et SQL. Il peut être utilisé pour créer des applications ou les regrouper en bibliothèques à déployer sur un cluster ou effectuer des analyses rapides interactives via des ordinateurs portables comme par exemple sur Jupyter, Spark-Notebook, Databricks.

Apache Spark expose une foule de bibliothèques familières aux data scientist, qui ont déjà travaillé avec Pandas de Python ou les dataframes de R. En outre, Apache Spark propose plusieurs algorithmes, modèles statistiques et frameworks déjà implémentés et ajustés: MLlib et ML pour

l'apprentissage automatique, GraphX et GraphFrames pour le traitement graphique c'est-à-dire la visualisation et Spark Streaming (DStreams et Structured). [37]

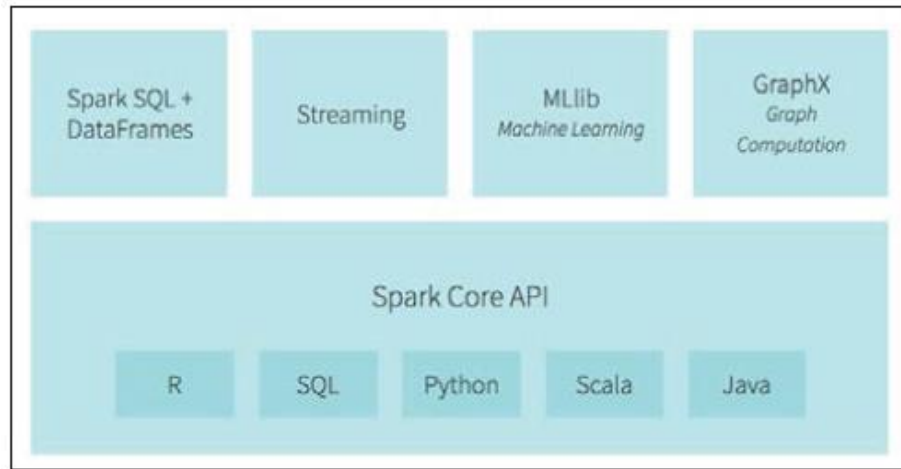


Figure 3.15 :*architecture spark*

3.7.1 *Clusters Spark*

Un cluster Spark est composé de deux types de processus : un programme pilote et plusieurs exécuteurs. En mode local, tous ces processus sont exécutés dans la même JVM ou Java Virtual Machine si le langage utilisé est Java. Dans un cluster, ces processus sont généralement exécutés sur des nœuds distincts.

3.7.2 *DataFrame*

Les DataFrames, comme les RDD, sont des collections immuables de données réparties entre les nœuds d'un cluster. Cependant, contrairement aux RDD, dans DataFrames, les données sont organisées en colonnes nommées. Le DataFrame de spark a le même concept que celle qu'on a déjà vue pour Pandas de Python.

3.8 Microsoft Azure Machine Learning

Azure Machine Learning permet aux data scientists et aux développeurs de transformer les données en informations à l'aide d'analyses prédictives. En facilitant l'utilisation des modèles prédictifs dans les solutions de bout en bout par les développeurs, Azure Machine Learning permet d'obtenir facilement des informations exploitables. A l'aide de Machine Learning Studio,

les data scientists et les développeurs peuvent rapidement créer, tester et développer des modèles prédictifs à l'aide d'algorithmes d'apprentissage machine de pointe. [38]



Figure 3.16 :*Machine Learning Microsoft Azure*

3.8.1 Les composants d'une expérience

Une expérience est faite des composants clés nécessaires pour construire, tester et évaluer un modèle prédictif. Dans Azure Machine Learning, une expérience contient deux composants : dataset ou jeux de données et module.

Un dataset contient des données qui ont été téléchargées dans Machine Learning Studio. Il est utilisé lors de la création d'un modèle prédictif. Machine Learning Studio fournit également plusieurs exemples de jeux de données pour faciliter le démarrage de la création d'une première expérience. Lorsque l'on explore le Machine Learning Studio, on peut télécharger des jeux de données supplémentaires.

Un module est un algorithme qu'on utilise lors de la construction d'un modèle prédictif. Machine Learning Studio fournit un large ensemble de modules pour prendre en charge le flux de travail de bout en bout de la science des données, à partir de la lecture de données provenant de différentes sources de données, du prétraitement des données, de la construction, et enfin de la validation d'un modèle prédictif.

Quelques éléments du module :

- Statistiques élémentaires : Calcule des statistiques élémentaires telles que la moyenne, l'écart type d'un jeu de données donné.
- Modèle d'entraînement : Ce module forme un algorithme de classification ou de régression sélectionné avec un jeu de données d'entraînement donné.
- Modèle de validation croisée : Ce module est utilisé pour effectuer une validation croisée afin d'éviter un ajustement excessif. Par défaut, ce module utilise la validation croisée 10 fois.

3.8.2 *Les étapes de la création d'une expérience*

D'abord les trois (3) premières étapes de la création d'une expérience fait partie de la phase de création d'un modèle qui sont :

- Etape 1 : charger les données
- Etape 2 : Prétraitement des données
- Etape 3 : Définir les attributs

Ensuite, dans l'étape 4, c'est la phase de la formation d'un modèle. Donc, à ce niveau, on choisit et applique un ou plusieurs algorithmes d'apprentissage automatique.

Enfin, pour la dernière étape, on test le modèle en faisant une prédiction avec de nouvelles données.

3.9 Conclusion

Dans ce chapitre, on a vu les différents outils, quelque fois plateforme tout entier, pour accomplir un projet d'apprentissage automatique. L'outil doit proposer la possibilité de visualisation, d'analyse de données ainsi que de la mise en place rapide des algorithmes d'apprentissage pour la prédiction. En outre, les langages dit langages de scripts (python, matlab...) sont utilisés par les data scientists pour les analyses initiaux du jeu de données, sans encore se soucier de l'architecture final que pourra avoir la machine. Et, d'autre comme Spark, Azuren ou encore TensorFlow sont beaucoup plus prisés par les Entreprises pour la mise en production. Finalement, c'est toutefois possible d'exploiter plusieurs outils lors d'un projet d'apprentissage automatique comme par exemple cohabiter spark et python avec l'API pyspark ou encore utiliser R dans Java avec l'utilisation de R-java.

CHAPITRE 4

PRESENTATION DE L'OUTIL D'ATTRITION ET DE LA RETENTION

4.1 Introduction

Le désabonnement des clients affecte toutes les entreprises, le mot Churn ou attrition client fait face au risque qu'un client passe d'une entreprise à une autre. La prédiction et l'analyse des churn peuvent aider une entreprise à développer une stratégie durable pour les programmes de rétention des clients. Attirer des milliers de nouveaux clients ne vaut rien si un nombre égal s'en va. En prenant conscience du pourcentage de désabonnés, nous pouvons proposer des analyses, des causes du désabonnement et un exemple de programme de rétention des clients.

Dans cet ouvrage, on va se consacrer sur la présentation de notre outil de prédiction de désabonnement pour une Entreprise de télécommunication.

4.2 Le churn

Les personnes financières rapportent le churn comme les anciens clients qui ne sont plus abonnés pour une raison quelconque. Le personnel du marketing considère l'attrition client comme étant les personnes qui choisissent de s'abonner avec une autre entreprise.

Churn prediction ou prédiction de désabonnement des clients désigne la prédiction de perte des consommateurs en faisant appel à des techniques d'analyse statistique, science des données ou des apprentissages automatiques. [39]

4.3 La rétention client

La rétention client définit la capacité de l'entreprise à garder ses clients, à faire en sorte qu'ils continuent d'utiliser ses services c'est-à-dire à les retenir. Comme pour le churn, la rétention client est aussi un indicateur : le taux de rétention client exprime la proportion des clients qui reste fidèle à une entreprise donnée pendant un moment donné. [40]

4.4 A propos de la plateforme

4.4.1 Objectifs

Le premier objectif de la plateforme est de prédire le taux de désabonnement à partir de l'analyse exploratoire, qui consiste à faire des analyses statistiques et par des techniques de visualisation des données, ensuite, on va renforcer l'étude à l'aide du Machine Learning. Enfin, par une

segmentation de nos jeux de données, on va pouvoir obtenir une offre et facturation plus cohérentes.

4.4.2 Les outils de développement

Pour la réalisation de notre data production, on a combiné quelques outils que nous avons déjà vu dans le chapitre précédemment.

- Langage : python
- Interface : Flask qui est un microframework intégré à python ainsi que toutes les technologies web tels que javascript, html, css
- Traitement de données et mathématiques : Pandas, NumPy
- Visualisations : matplotlib et seaborn
- Bibliothèque d'apprentissage automatique : scikit-learn et Keras (avec l'utilisation de backend de TensorFlow)

4.5 Présentations des interfaces

4.5.1 Section accueil

L'interface d'accueil permet de s'authentifier pour entrer dans la plateforme, de plus on peut voir le résumé du projet ainsi que la présentation du mémoire elle-même en cliquant sur les deux icônes présentes dans la figure 4.01 :

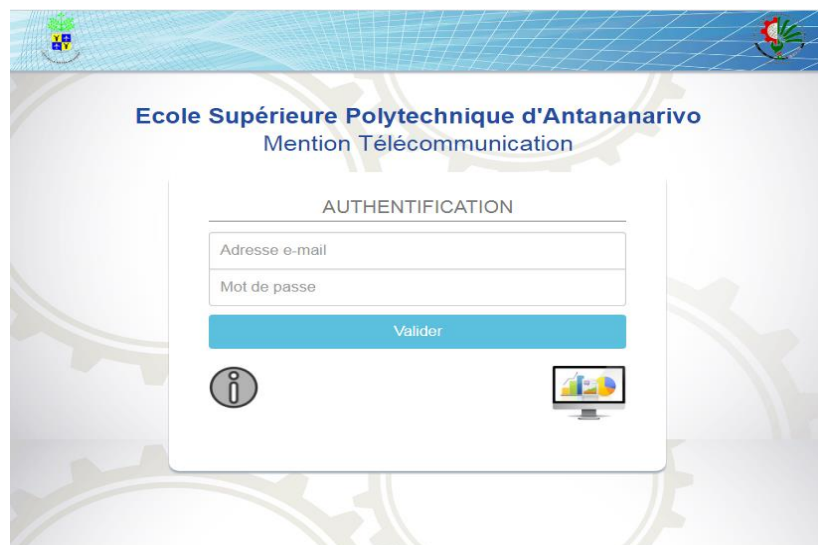


Figure 4.01 :page d'accueil

4.5.2 Page principale

Après l'authentification, on est dirigé dans la page principale. On aperçoit trois icônes qui nous dirigent respectivement vers la partie Présentation data, features engineering et data visualisation. Une barre de navigation qui représente les différents menus tels que :

- Dataset : qui a le même rôle que les trois icônes cités plus haut ;
- Prédiction : pour regarder la prédiction faite par les différents algorithmes existant dans la plateforme ;
- Rétention clients : cette partie de la plateforme se consacre sur la stratégie de la rétention clients.

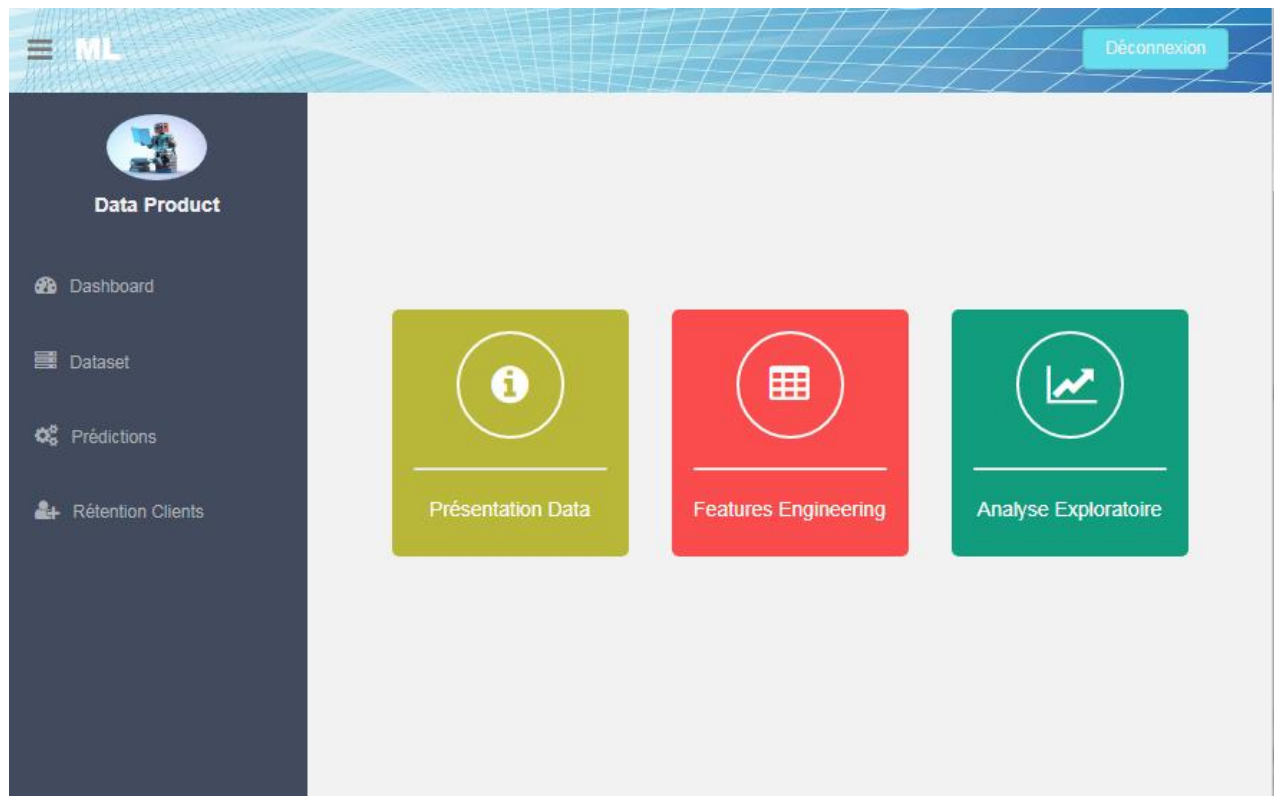


Figure 4.02 :page principale

4.6 Présentations du jeu de données

Le jeu de données que nous avons choisi d'utiliser est un jeu de données public concernant les appels et services d'une entreprise de télécommunication durant un concours organisé par Kaggle.

Noms des colonnes	Types	Descriptions
State	chaîne de caractère	L'Etat du client
Durée compte	numérique	Durée du client avec l'opérateur
Région	chaîne de caractère	La région du client
N°tel	discret	Le numéro de téléphone de l'abonné
Plan int/ Messagerie vocal	booléen	1 si l'abonné l'utilise Si non 0
Nbr vmail	numérique	Nombre des messages vocaux
Tot min journée/soirée/nuit/int	numérique	Nombre total de minutes d'appel pendant la journée/soirée/nuit/ et international
Tot appel journée/soirée/nuit/int	entier	Nombre d'appel total durant la journée/soirée/nuit et international
Fact journée/soirée/nuit/int	numérique	Coût d'appel pendant la journée/soirée/nuit/international
Appel service client	entier	Nombre d'appel effectué par chaque client au service clientèle
churn	booléen	C'est le label qui définit si un utilisateur va rester ou non

Tableau 4.01 :présentation du dataset

4.7 Feature engineering

La features engineering est une étape très importante pour avoir une meilleure modèle d'apprentissage automatique. Elle consiste à nettoyer et à rendre exploitable les données. [32]

4.7.1 Normalisation dataset

Notre jeu des données ne représente pas des valeurs manquantes, donc on va passer directement à l'étape de normalisation. Ici, on va transformer les valeurs comme true/false, yes/no en une valeur booléenne c'est-à-dire 0 et 1 sinon, ils ne seront pas interprétables durant l'apprentissage et à

changer les noms de colonnes. L'opération se fait en cliquant sur l'icône normalisée présente dans la figure 4.03.

NORMALISATION																					
Normalisée		Dataset Initial																			
	state	account length	area code	phone number	international plan	voice mail plan	number vmail messages	total day minutes	total day calls	total day charge	total eve minutes	total eve calls	total eve charge	total night minutes	total night calls	total night charge	total intl minutes	total intl calls	total intl charge	customer service calls	churn
0	KS	128	415	382-4657	no	yes	25	265.1	110	45.07	197.4	99	16.78	244.7	91	11.01	10.0	3	2.70	1	False
1	OH	107	415	371-7191	no	yes	26	161.6	123	27.47	195.5	103	16.62	254.4	103	11.45	13.7	3	3.70	1	False
2	NJ	137	415	358-1921	no	no	0	243.4	114	41.38	121.2	110	10.30	162.6	104	7.32	12.2	5	3.29	0	False
3	OH	84	408	375-9999	yes	no	0	299.4	71	50.90	61.9	88	5.26	196.9	89	8.86	6.6	7	1.78	2	False
4	OK	75	415	330-6626	yes	no	0	166.7	113	28.34	148.3	122	12.61	186.9	121	8.41	10.1	3	2.73	3	False

Figure 4.03 :*jeu des données initial*

Après avoir cliqué sur l'icône normalisée du figure 4.03, on voit bien que les valeurs dans les colonnes suivantes : « plan int, messagerie vocale et churn » ont bien été changées ; et que toutes les colonnes ont obtenu un nouveau nom.

	State	Durée compte	Région	N° tel	plan int	messagerie vocal	nbr vmail	tot min journée	tot appel journée	fact journée	tot min soirée	tot appel soirée	fact soirée	tot min nuit	tot appel nuit	fact nuit	tot min int	tot appel int	fact int	appel service clients	churn
0	KS	128	415	382-4657	0	1	25	265.1	110	45.07	197.4	99	16.78	244.7	91	11.01	10.0	3	2.70	1	0
1	OH	107	415	371-7191	0	1	26	161.6	123	27.47	195.5	103	16.62	254.4	103	11.45	13.7	3	3.70	1	0
2	NJ	137	415	358-1921	0	0	0	243.4	114	41.38	121.2	110	10.30	162.6	104	7.32	12.2	5	3.29	0	0
3	OH	84	408	375-9999	1	0	0	299.4	71	50.90	61.9	88	5.26	196.9	89	8.86	6.6	7	1.78	2	0
4	OK	75	415	330-6626	1	0	0	166.7	113	28.34	148.3	122	12.61	186.9	121	8.41	10.1	3	2.73	3	0

Figure 4.04 :*jeu des données normalisées*

4.7.2 Validation du dataset

Maintenant que notre dataset est normalisé, on va passer à l'étape de la validation. D'abord, le calcul des corrélations entre les attributs est primordial pour avoir une idée sur les attributs qui pourront être importants ou non. La figure 4.05 montre la matrice de corrélation pour tous les attributs.

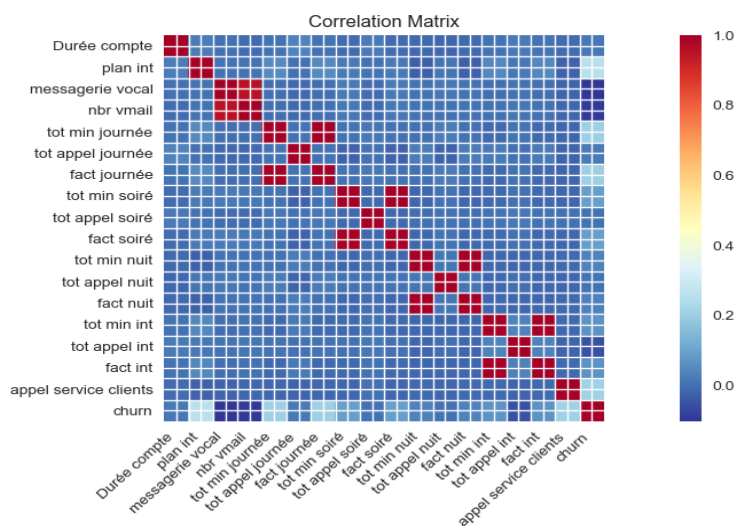


Figure 4.05 : la matrice de corrélation de notre jeu des données

Les cases en rouge évoquent une forte corrélation (peut être liaison) entre deux (2) attributs et les bleues foncées suppose qu'il n'y en a pas.

Mais ce qui nous intéresse le plus pour notre modèle, c'est la corrélation des attributs par rapport au churn. Pour être plus explicite, on va les représenter sous forme d'un graphe comme le montre la figure 4.06.

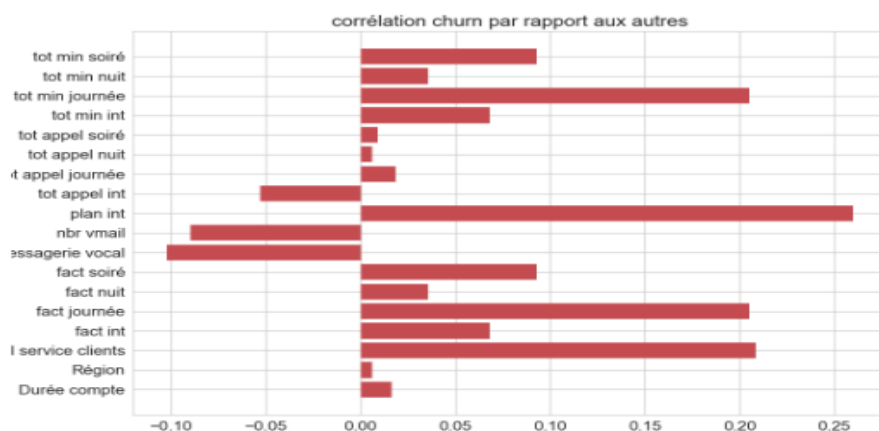


Figure 4.06 : les corrélation des attributs par rapport à churn

Quatre (4) attributs dépassent un taux d'interdépendance de 0,20 tels que le total minute journée, le plan international, la facturation journée et l'appel service client. Toutefois, une grande corrélation n'implique pas une causalité. Mais à partir de ce résultat nous pouvons déjà effacer quelques attributs (state, région, n° tel, tot appel journée/soirée/nuite) qui ne présentent aucune corrélation ou presque négligeable (moins de 0.05) par rapport au label churn.

	Durée compte	plan int	messaging vocal	nbr vmail	tot min journée	fact journée	tot min soiré	fact soiré	tot min nuit	fact nuit	tot min int	tot appel int	fact int	appel service clients	churn
15	161	0	0	0	332.9	56.59	317.8	27.01	160.6	7.23	5.4	9	1.46	4	1
16	85	0	1	27	196.4	33.39	280.9	23.88	89.3	4.02	13.8	4	3.73	1	0
17	93	0	0	0	190.7	32.42	218.2	18.55	129.6	5.83	8.1	3	2.19	3	0
18	76	0	1	33	189.7	32.25	212.8	18.09	165.7	7.46	10.0	5	2.70	1	0
19	73	0	0	0	224.4	38.15	159.5	13.56	192.8	8.68	13.0	2	3.51	1	0

Figure 4.07 :*dataset validé*

4.8 L'analyse exploratoire

On sait maintenant les corrélations entre les caractéristiques ou features, mais pour mieux comprendre la donnée, il faut passer par l'analyse exploratoire. Explorer les données est essentiellement la recherche des éventuelles relations, connexion entre les attributs à l'aide des graphes ou des techniques statistiques. [29][41]

Au préalable, seule la distribution de « total minute journée » montre une différence de modalité selon l'état de churn (0 ou 1). Cela pourrait suggérer que cet attribut pourrait avoir une importance pour la prédiction de désabonnement.

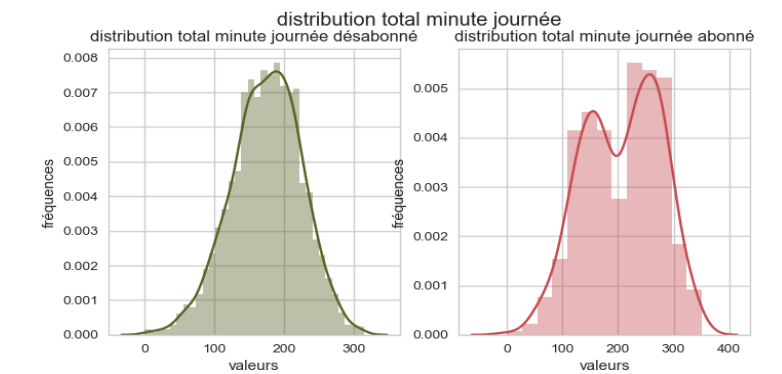


Figure 4.08 :*distribution « total minute journée »*

Ensuite, on a observé qu'il y a une forte corrélation entre le « total des minutes journée/ soirée » et le label « churn ».

Pour vérifier leurs impacts sur le désabonnement, on va tracer leurs dispersions en prenant compte la valeur du label churn.

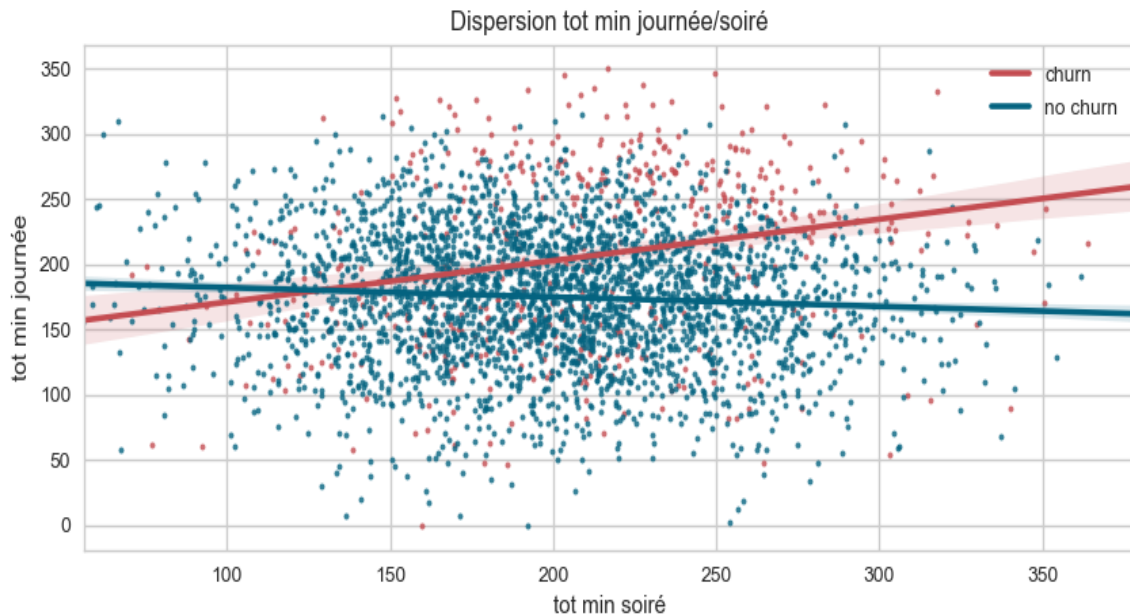


Figure 4.09 : *la dispersion entre total des minutes journée et soirée*

Quelque chose qu'on peut noter sur la figure 4.09 c'est qu'on a une régression linéaire entre la minute de journée et le churn. Plus le nombre de minute dans la journée augmente, plus le nombre de désabonnement augmente également (concentration des points rouges au-dessus de la ligne rouge). Et aussi, une corrélation presque négligeable pour les clients qui vont rester.

En outre, deux autres caractéristiques nous intriguent : la première : le « nombre de messagerie vocale » et la seconde : le « plan international » à cause de leurs fortes corrélations et aussi de leur type booléen.

Notre objectif est d'essayer de trouver des tendances dans notre jeu de données par rapport à l'utilisation ou non de ces deux services.

Après avoir fait les analyses nécessaires nous avons obtenu deux résultats bien distincts.

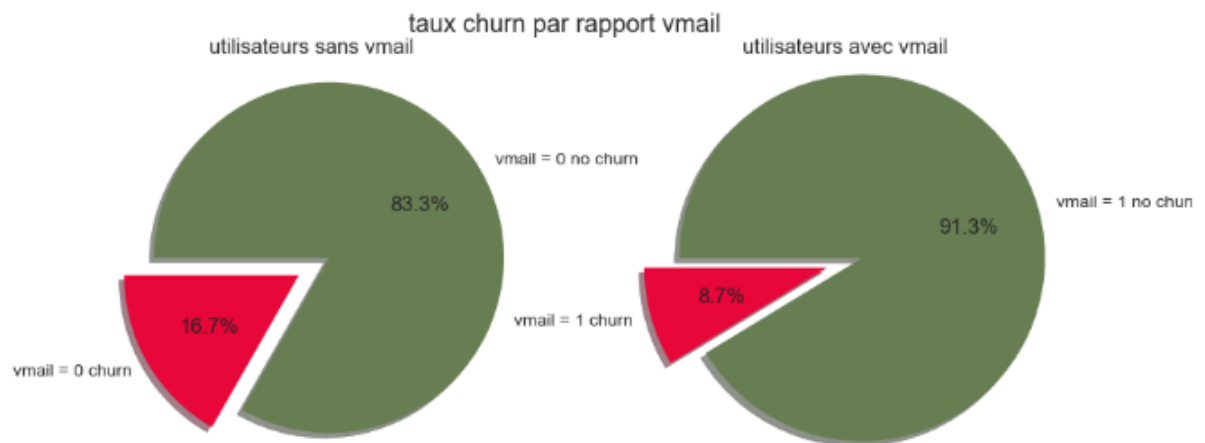


Figure 4.10 : *la tendance des désabonnements par rapport à la messagerie vocale*

En interprétant ces deux graphes, on peut tirer que les utilisateurs qui utilisent la messagerie vocale continuent généralement leur abonnement, ceux qui décident de partir représentent moins de 10%, plus précisément 8,7%. Par contre, pour les clients qui ne l'utilisent pas, on constate que 16,7% décident de quitter l'entreprise soit le double du premier cas.

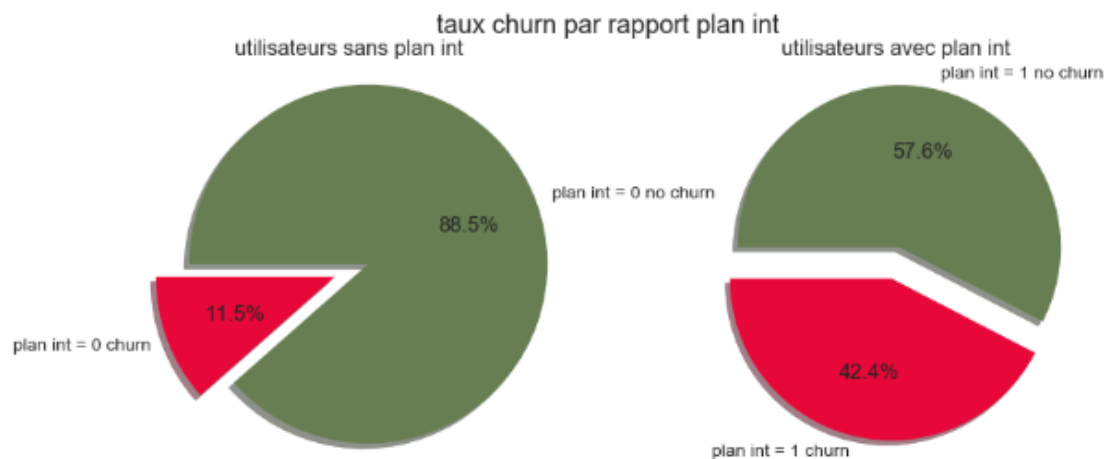


Figure 4.11 : *la tendance des désabonnements par rapport au plan international*

En appliquant les mêmes principes que pour la messagerie vocale, on observe une tendance dissimilaire. L'utilisation du plan international provoque le désabonnement des clients (42,4% contre 11,5%).

On va vérifier la tendance vu dans figure 4.11 à l'aide d'une boîte à moustache.

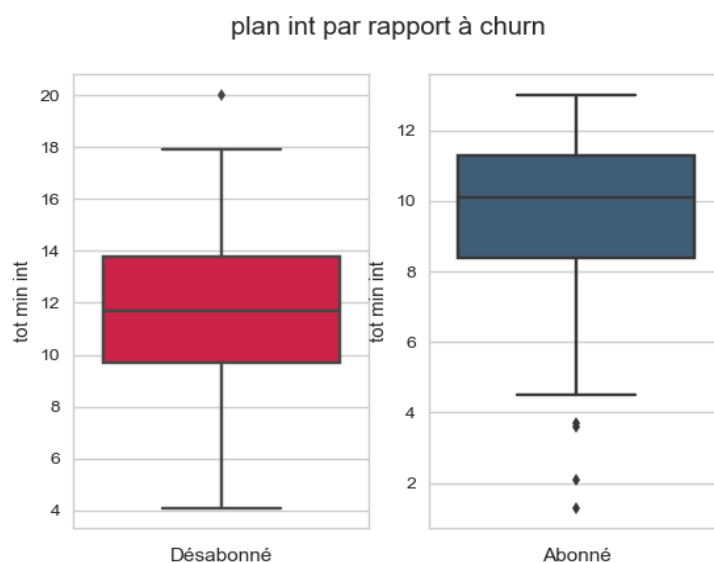


Figure 4.12 :boite à moustache plan international par rapport à churn

Pour les désabonnés on a une concentration de minute entre 10 et 14 et pour les abonnés la concentration est moins dense, de 8 à un peu plus de 11. Ce graphe confirme ce que nous avons dit à propos du plan international, c'est-à-dire que plus les utilisateurs utilisent le plan international plus les risques de désabonnement sont grands.

	Désabonné	Abonné
Total minute international minimum	4	Un peu supérieur à 4
Total minute international maximum	18	12
médiane	12	10
Valeurs aberrantes	20	En dessous de 4

Tableau 4.02 :interprétation de la boite à moustache

Avant de terminer notre analyse, il nous reste encore à explorer un dernier attribut possédant une forte corrélation qui est le nombre d'appel aux services clients. Traçons d'abord le pourcentage d'abonnement/désabonnement par rapport aux nombres d'appels aux services clients.

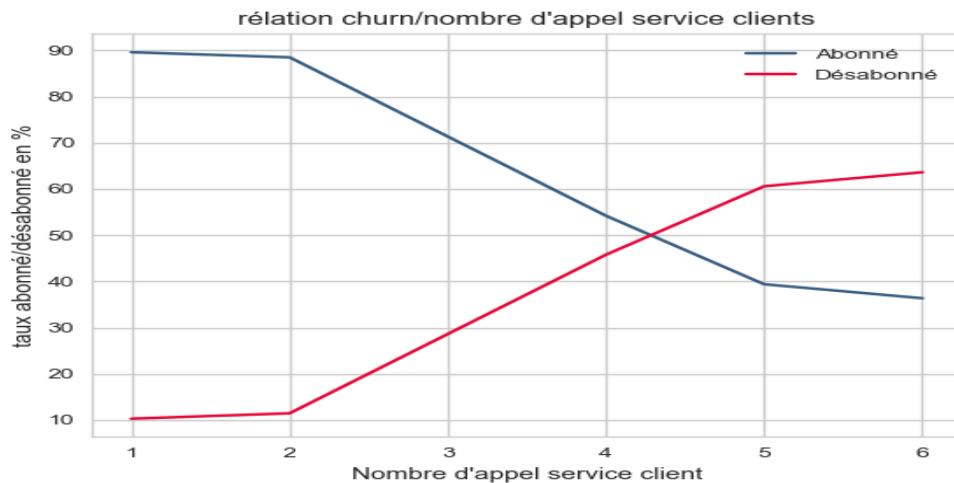


Figure 4.13 :relations entre l'attribut churn et les nombres d'appels au service client

Le taux de désabonnement augmente avec le nombre d'appels au service client. Le taux des désabonnés devient plus nombreux à partir de cinq (5) appels.

Pour mieux comprendre la tendance, on va utiliser la présentation avec la boîte à moustache.

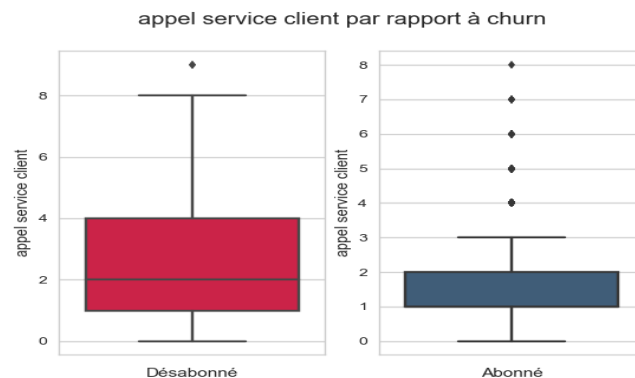


Figure 4.14 :présentation sous boîte à moustache des appels au service client et le churn

Généralement, les utilisateurs qui ont effectué plus de trois (3) appels sont les plus susceptibles de se désabonner. La figure 4.14 montre qu'il y a une certaine ressemblance entre les utilisateurs parmi les deux catégories de zéro jusqu'à trois (3) appels. Mais au-delà de 3 appels, la tendance de désabonnement devient de plus en plus précise, parce qu'à partir de quatre (4) appels, il ne reste plus que des valeurs aberrantes pour le cas abonné.

4.8.1 Division jeux des données

Dorénavant, on a constaté quelques tendances dans notre jeu de données, il est temps de la diviser en jeu d'entraînement ou « train » pour l'entraînement de notre modèle et en jeu de test ou « test » pour l'évaluation. Pour diviser les jeux de données il faut s'assurer que tous les cas sont présents dans l'un et l'autre (train/test), avoir suffisamment de données test (par exemple : 70 train et 30 test), éviter d'utiliser la même donnée pour le train et le test.

4.9 Les algorithmes de classification utilisés

Les algorithmes qu'on va présenter dans cet ouvrage représentent ceux qui ont eu la meilleure performance pour résoudre la prédiction de désabonnement et aussi du point de vue temps d'exécution, à part pour la régression logistique.

- KNN
- Régression Logistique
- Arbre de décision
- Forêt aléatoire
- Réseau de neurone

4.10 Evaluation et amélioration du modèle

Après avoir présenté les algorithmes qu'on a utilisés, nous allons maintenant voir plus profondément l'évaluation de nos modèles. En ce qui concerne l'amélioration de nos modèles, on a choisi différentes techniques pour chacun des algorithmes.

4.10.1 *K Nearest Neighbors* ou KNN

4.10.1.1 Evaluation initiale

Pour notre modèle initiale, on a déjà appliqué un prétraitement à notre dataset qui est la features sélection. Par conséquent, toutes les caractéristiques ne sont plus utilisées, mais seulement celles qui sont jugées plus importantes par notre modèle.

Les deux figures 4.15 et 4.16 présentent les résultats du modèle initial :

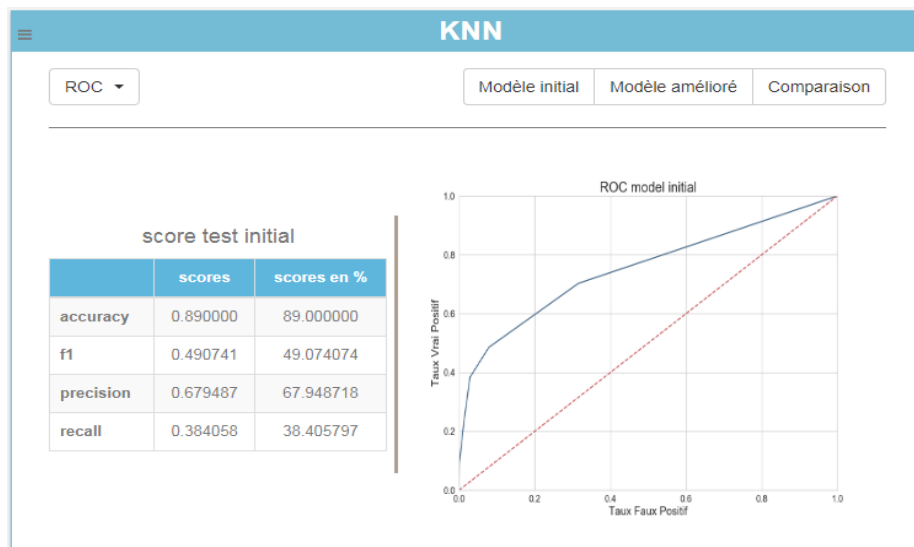


Figure 4.15 :évaluation du modèle avec KNN par ROC

Notre courbe ROC s'écarte trop rapidement de l'axe de y, cela se traduit par une faible précision et ne monte pas suffisamment (manque de rappel).

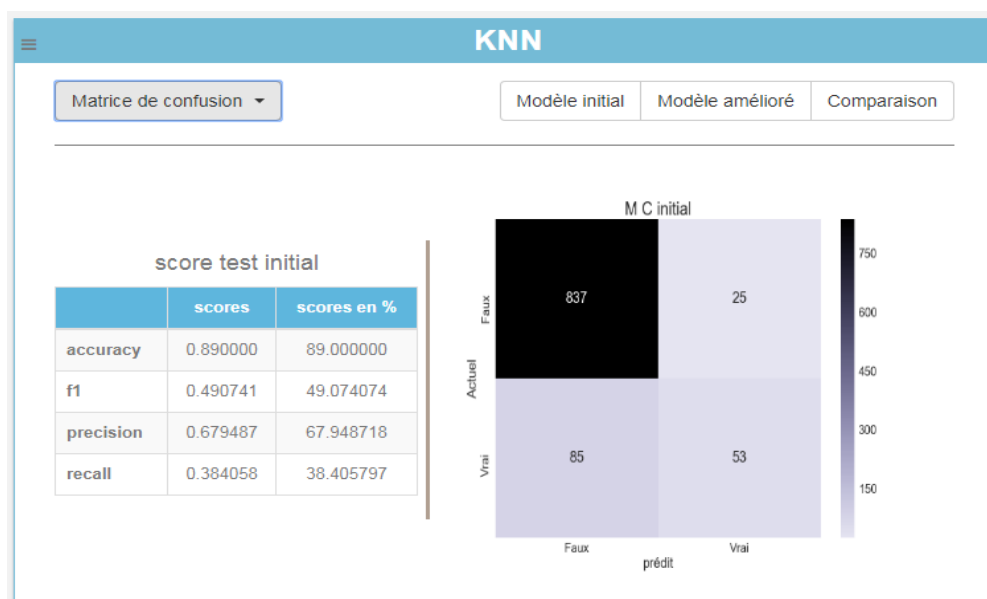


Figure 4.16 :évaluation du modèle avec KNN par Matrice de Confusion

Le modèle avec l'algorithme de KNN a réussi à trouver 53 Vrai Positif. Ce résultat signifie que les 53 utilisateurs vont réellement se désabonner (Bien classé). Et 85 utilisateurs classés en Faux Négatif (mal classé). Ils sont prédit dans la classe des utilisateurs qui vont rester alors qu'en réalité ils vont se désabonner (leurs churn = 0).

4.10.1.2 Amélioration

On a réussi à améliorer l'algorithme KNN en standardisant d'abord les données d'entrée et en cherchant les meilleurs paramètres parmi les combinaisons possibles en utilisant « Grid Search CV » présent dans la bibliothèque scikit learn.

Après réévaluation, on a un résultat nettement supérieur que précédemment, comme illustre les figures 4.17.

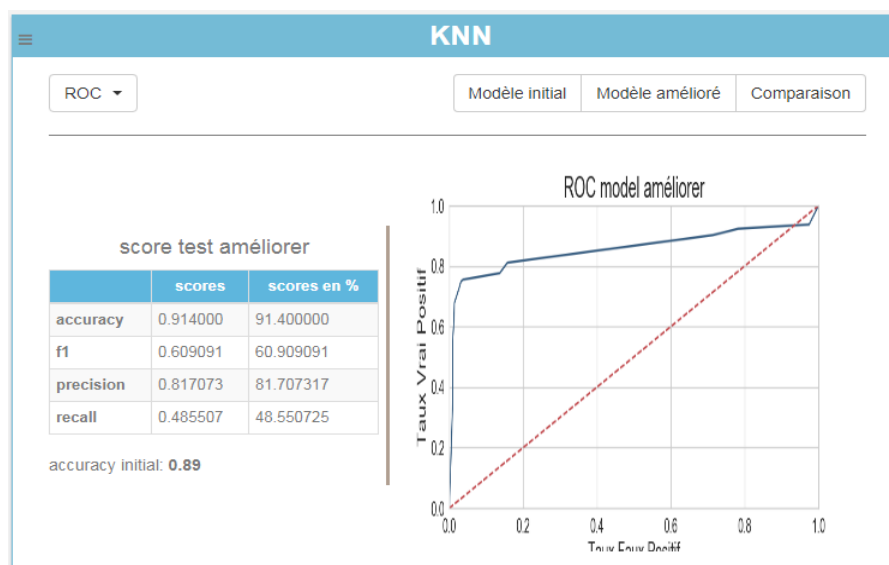


Figure 4.17 :évaluation du modèle avec KNN amélioré par la courbe ROC

La courbe ROC se rapproche plus de 1 (axe y) par rapport au modèle initial, cela se traduit par l'augmentation du rappel et ne s'écarte plus hâtivement (augmentation de la précision).

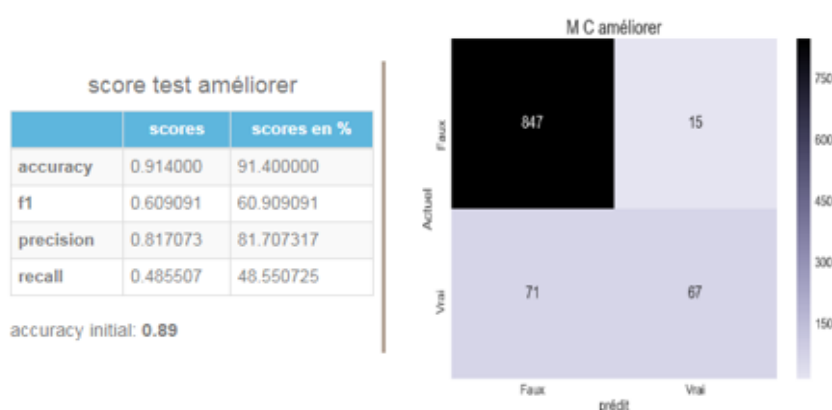


Figure 4.18 :évaluation du modèle avec KNN amélioré par Matrice de Confusion

Notre modèle avec KNN amélioré (figure 4.18) a bien classé 67 utilisateurs qui vont se désabonner contre 53 auparavant. On remarque aussi l'augmentation de 2,4% de notre score d'accuracy ou score d'exactitude ce qui n'est pas négligeable.

4.10.1.3 Comparaison

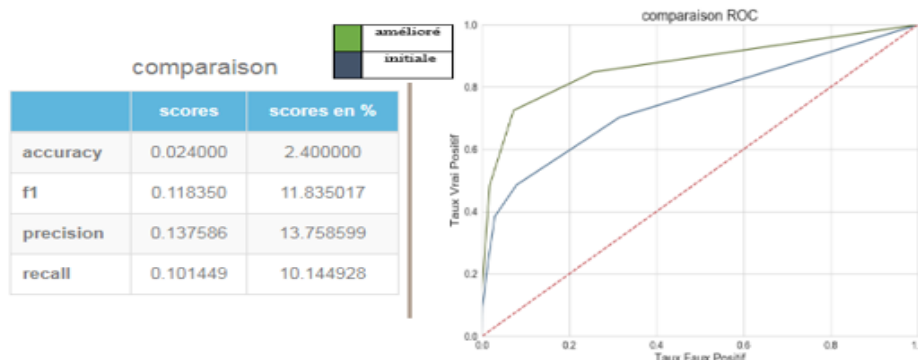


Figure 4.19 : comparaison entre modèle initial et améliorée

Notre modèle a gagné en performance surtout en précision mais aussi au niveau du rappel.

4.10.2 Régression logistique

4.10.2.1 Evaluation initiale

Le calcul de l'exactitude de notre modèle en régression logistique ou RL a donné le score de 0,860 Soit plus de 85%, par contre ce modèle n'est pas performant du tout après l'avoir évalué avec des différentes techniques comme illustré dans la figure 4.20 :

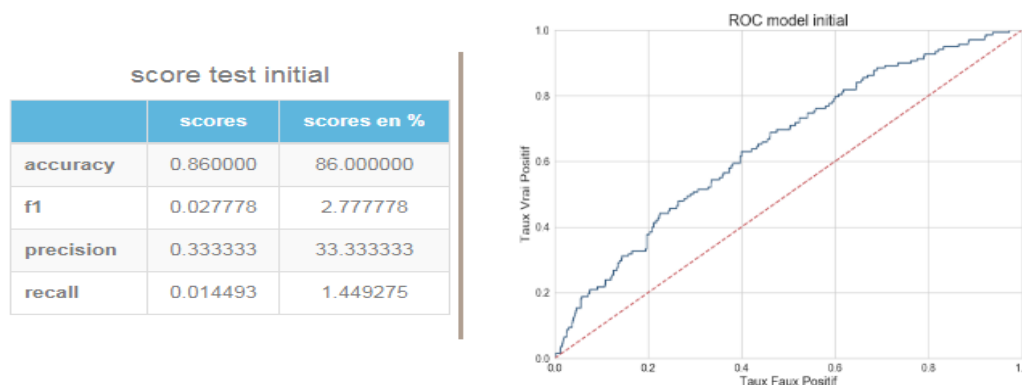


Figure 4.20 : évaluation du modèle avec RL initial par la courbe ROC

La trajectoire presque parallèle à la ligne rouge de la courbe ROC symbolise un mauvais modèle, surtout au niveau du rappel, il n'a pas la capacité de prédire le VP (f1 presque négligeable).

score test initial		
	scores	scores en %
accuracy	0.860000	86.000000
f1	0.027778	2.777778
precision	0.333333	33.333333
recall	0.014493	1.449275

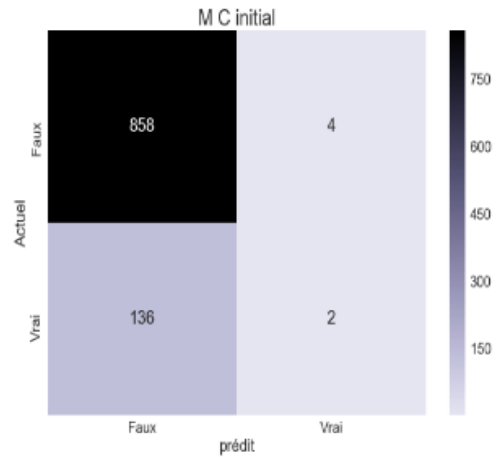


Figure 4.21 :évaluation du modèle RL initial par la M C

La matrice de confusion est en accord avec notre courbe ROC avec la prédiction que de 2 utilisateurs qui vont réellement se désabonner.

4.10.2.2 Amélioration

Afin d'améliorer le résultat initial, on a procédé à la standardisation des données d'entrée en lui appliquant le PCA (Principal Component Analysis) pour la réduction de dimension et la standardisation des données d'entrée.

score test améliorer		
	scores	scores en %
accuracy	0.881000	88.100000
f1	0.278788	27.878788
precision	0.851852	85.185185
recall	0.166667	16.666667

accuracy initial: **0.86**

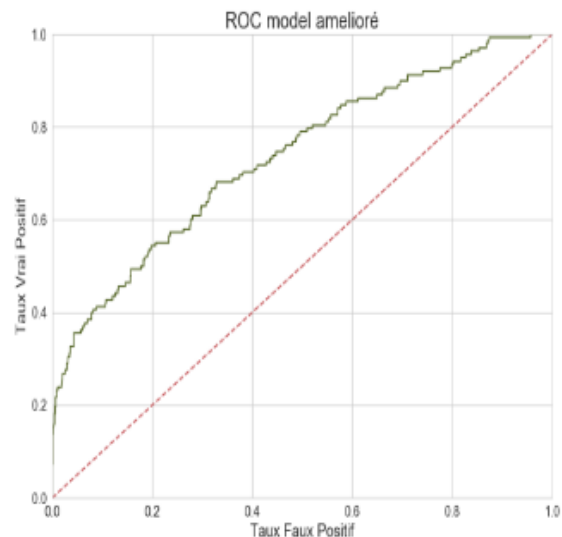


Figure 4.22 :résultat de l'amélioration du modèle avec RL

L'augmentation est de 2,1% pour le nouveau modèle, on a réussi à mieux classer les utilisateurs selon leurs classes respectives mais le résultat n'est pas encore satisfaisant. Le modèle n'est pas vraiment adapté pour notre prévision même si son score d'exactitude semble bon.

4.10.3 Les arbres de décision

4.10.3.1 Evaluation initiale

Les arbres de décision sont une méthode d'apprentissage supervisé non paramétrique. L'objectif est de créer un modèle qui prédit la valeur d'une variable cible, dans notre cas c'est la valeur de churn en apprenant des règles de décision déduites des caractéristiques des données.

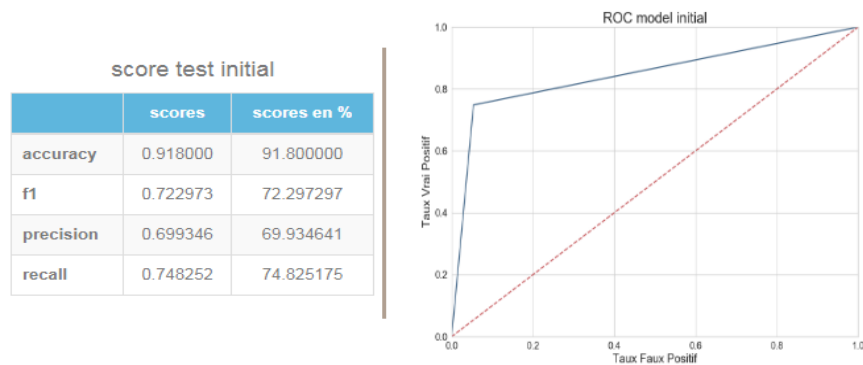


Figure 4.23 :évaluation du modèle initiale de l' arbre des décisions par la courbe ROC

Notre score d'exactitude ou accuracy est de 0.918, d'après la courbe ROC, le modèle possède une meilleure capacité en termes de précision et de rappel.

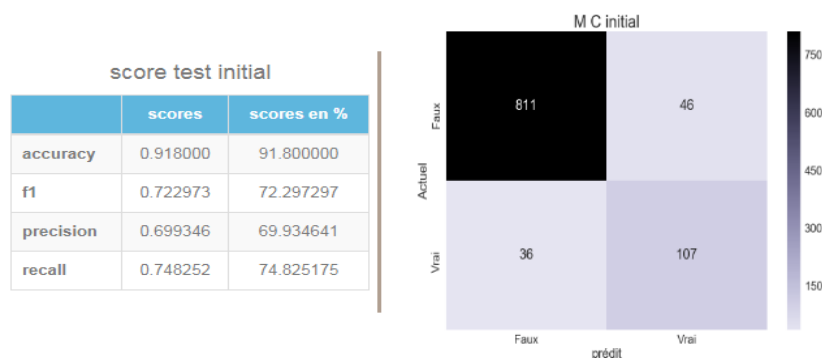


Figure 4.24 :évaluation du modèle initiale de l' arbre des décisions par la MC

En interprétant la matrice, on voit que notre modèle est capable de bien classer 107 utilisateurs VP, c'est-à-dire qu'ils vont vraiment partir.

4.10.3.2 Améliorations

En raison de son aspect instable, et sa forte propension à l'overfitting ou sur-apprentissage, globalement, on n'utilise plus les arbres de décision. On va plutôt l'utiliser en tant que classifieurs faibles à la base de méthodes ensemblistes qui est AdaBoost. Pour augmenter encore plus la

performance du modèle combiné sans recours au risque d'une éventuel overfitting, avant l'entraînement on va appliquer une technique de prétraitement qui est la sélection des caractéristiques. [16]

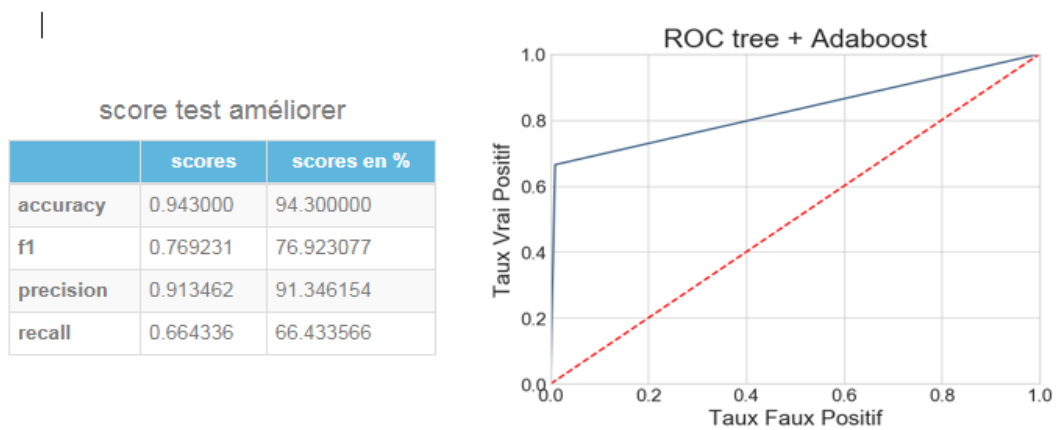


Figure 4.25 : *résultat après l'amélioration de l' arbre des décisions*

On remarque une forte augmentation en précision et une légère baisse au niveau du rappel (figure4.25), mais en général notre modèle s'est bien amélioré en terme de capacité, maintenant il est capable de mieux classer les utilisateurs selon leurs classes respectives.

Pour mieux comprendre cela, regardons l'évaluation par la matrice de confusion, on voit bien que le taux de Vrai Positif a diminué (de 103 à 98, on a moins prédit des utilisateurs qui vont quitter l'entreprise), en contrepartie notre FN a augmenté (811 contre 852, cela se traduit par l'augmentation de la précision).

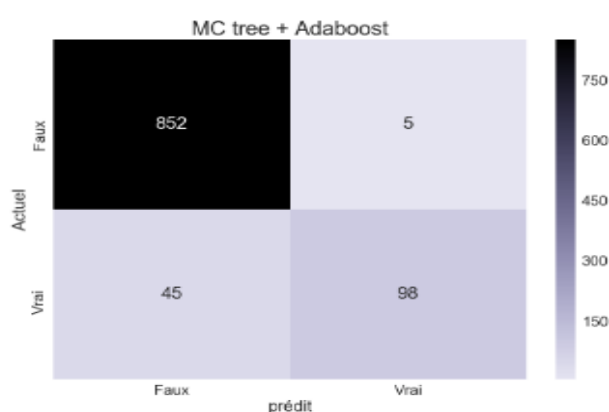


Figure 4.26 : *évaluation du modèle amélioré par la matrice de confusion*

4.10.4 Les forêts aléatoires

4.10.4.1 Evaluation initiale

Les forêts aléatoires sont une amélioration naturelle des arbres de décision. Au lieu de créer un seul arbre, l'algorithme va créer plusieurs arbres aléatoirement. En raison de la sélection des caractéristiques automatiques qu'on a mises en place, le modèle initial est assez instable et le score d'exactitude change $\pm 0,01$.

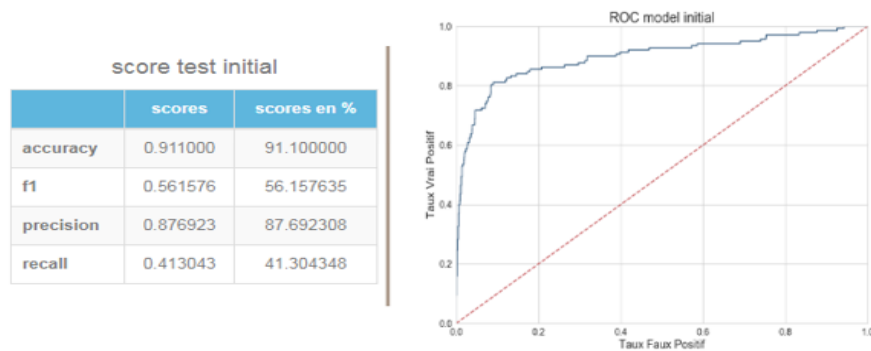


Figure 4.27 :évaluation du modèle initiale avec les forêts aléatoires par la courbe ROC

On a déjà une meilleure précision, mais un rappel un peu moins de 0.5, c'est ce dernier qu'on va essayer d'améliorer pour notre modèle.

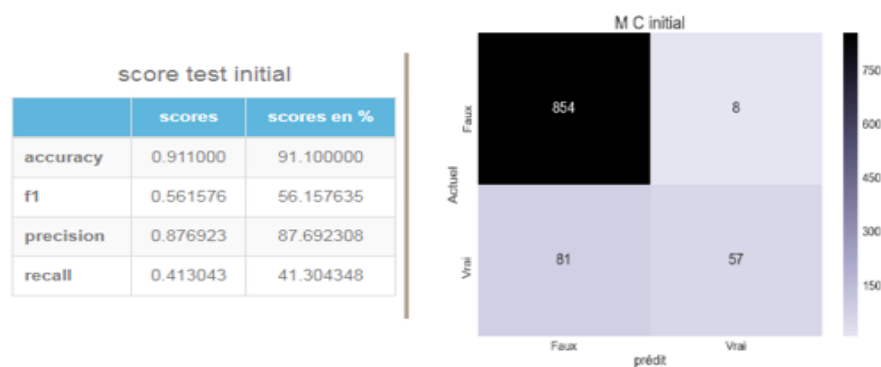


Figure 4.28 :évaluation du modèle initial par la matrice de confusion

La matrice de confusion affirme ce que nous avons dit plus haut, le faible rappel de notre modèle conduit à une faible prédiction de Vrai Positif avec seulement un taux de 57 utilisateurs.

4.10.4.2 Amélioration

Pour améliorer le rappel de notre modèle initial, on va procéder à la recherche de la meilleure combinaison possible des hyperparamètres. Suite à plusieurs séries de tests et des recherches (méthode de grille) de la meilleure combinaison, on a remarquablement amélioré notre modèle. Le tableau 4.03 résume les résultats de recherches des paramètres concernés :

Paramètres	Description	Valeurs
n_estimators	Le nombre d'arbres dans la forêt : [entier]	20
Criterion	Critère séparation feuille d'un arbre : [Gini ou Entropie]	Gini
Max_depth	c'est la profondeur maximale de chaque arbre. Paramètre important qui dépend du niveau d'interaction entre les variables.	10
Max_features	C'est le nombre maximum de variable qu'on tire aléatoirement pour chaque arbre : [entier]	8

Tableau 4.03 :la meilleure combinaison pour nos forêts aléatoires

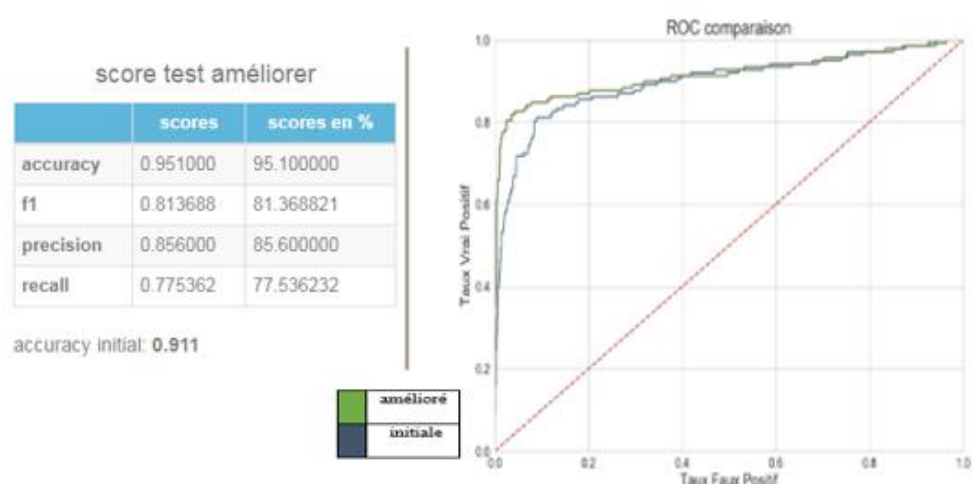


Figure 4.29 :évaluation et comparaison par la courbe ROC

L'évaluation de nos modèles par la courbe ROC montre une certaine augmentation de la valeur du rappel qui passe de 0,413 à 0,775.

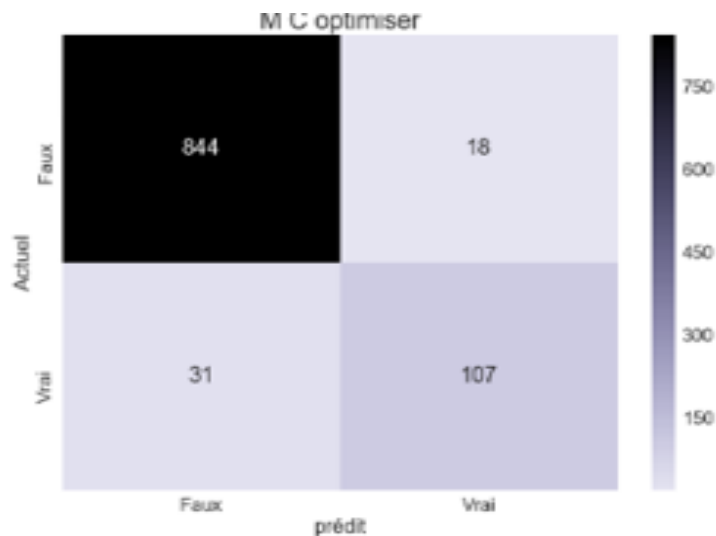


Figure 4.30 : la MC du modèle avec les forêts aléatoire amélioré

Notre modèle fait de moins en moins d’erreur de classification et une nette augmentation de Vrai Positif (de 57 à 107).

4.10.5 Réseau de neurones

Les réseaux neuronaux artificiels ou RNA constituent une approche populaire pour résoudre des problèmes complexes, tel que le problème de la prédiction du taux de désabonnement.

Plusieurs structures ont été testées durant la construction de la topologie de notre réseau de neurone.

4.10.5.1 Keras

Keras est une bibliothèque python puissant pour développer et évaluer des modèles d’apprentissage profond. Sa flexibilité le rend très puissant parce que keras peut utiliser le backend Theano et Tensorflow pour effectuer les différents calculs numériques.

4.10.5.2 Définition d'un modèle

On va créer un modèle Séquentiel et ajouter des couches une à la fois jusqu'à ce que nous soyons satisfaits de notre topologie.

```
32 def constr_classif():
33
34     classif = Sequential()
35
36     classif.add(Dense(activation="softmax", input_dim=9, units=12, kernel_initializer="normal"))
37
38     # ajout seconde couche cacher
39     classif.add(Dense(activation="softmax", units=9, kernel_initializer="uniform"))
40
41
42     # ajout couche de sortie
43     classif.add(Dense(activation="sigmoid", units=1, kernel_initializer="uniform"))
44
45     classif.compile(optimizer='adam', loss='binary_crossentropy', metrics=['accuracy'])
46     return classif
```

Figure 4.31 : proposition d'une topologie

- La première chose que fait notre fonction « constr_classif() » est de créer une instance de « Sequential model » nommée classif (ligne 34). Un modèle séquentiel en Keras est un pipeline linéaire (une pile) de couches de réseaux neuronaux
- Ensuite, à la ligne 36, on définit une seule couche avec 12 neurones artificiels qui attend 9 variables d'entrée et on a soumis un poids spécifique à chaque neurone qui est de « normal » en appelant la méthode « add » de notre classif.
- L'ajout des autres couches se fait par le même principe mais avec des valeurs différentes.
- Enfin, après la définition de toutes les couches, on va pouvoir compiler notre modèle par la méthode « compile » de keras.

En utilisant la fonction « summary() », nous pouvons voir le résumé de notre topologie réseau, représenté par la figure 4.32:

Layer (type)	Output Shape	Param #
dense_1 (Dense)	(None, 12)	120
dense_2 (Dense)	(None, 9)	117
dense_3 (Dense)	(None, 1)	10

Figure 4.32 : résumé de notre topologie réseau

4.10.5.3 Evaluations

Nous avons formé notre réseau de neurones, maintenant on va l'évaluer avec notre jeu de données de test.

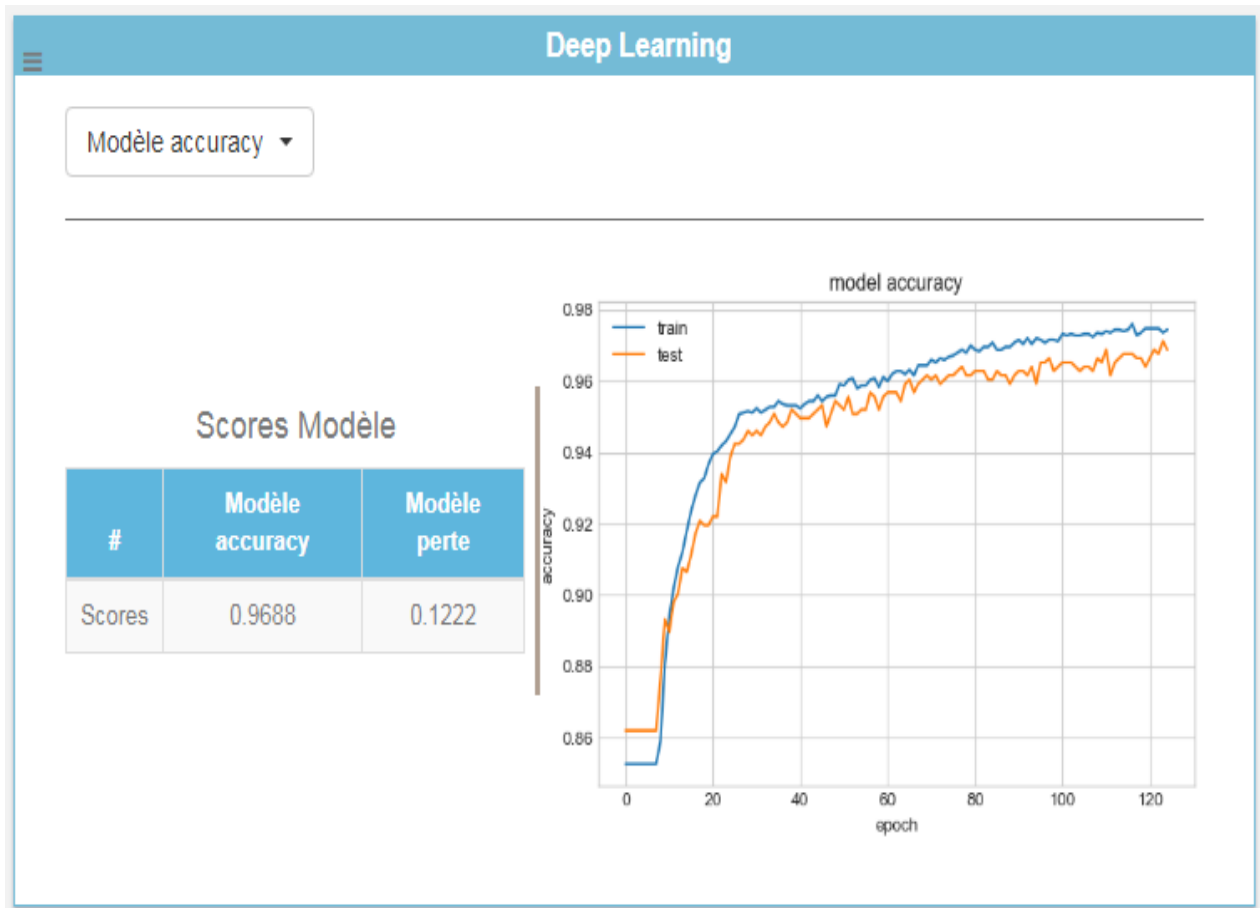


Figure 4.33 : *évaluation du score de la modèle accuracy train/test*

Dans la figure 4.33, on remarque deux scores : d'abord, le score du modèle accuracy et le score du modèle perte qui sont de 0.9652 et 0.1252 respectivement.

Le terme accuracy ici, a le même sens que l'accuracy que nous avons vu durant l'évaluation de nos différents algorithmes et le score du modèle perte est la somme des erreurs faites par notre modèle (figure 4.34).

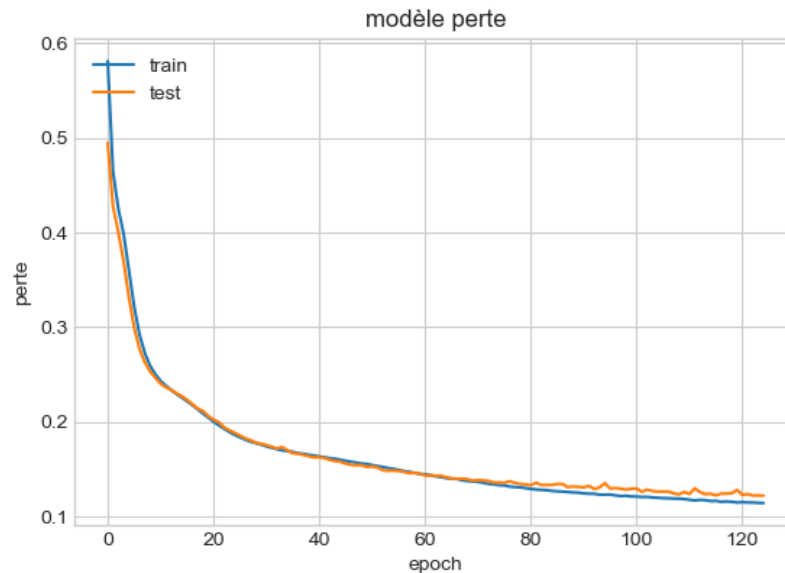


Figure 4.34 : *évaluation du score de modèle perte train/test*

En conséquence, pour avoir un bon modèle, il faut que l'accuracy soit le plus haut possible et inversement le score du modèle perte, le plus bas possible.

Il est utile d'observer comment la précision augmente sur les ensembles d'entraînement et de test lorsque le nombre d'époques (epochs) augmente (les courbes à droite de notre figure 4.33).

Comme on peut le voir, ces deux courbes touchent environ 125 époques, d'où, il n'est pas nécessaire de s'entraîner plus loin après ce point sinon, on risque d'avoir de problème de sur-apprentissage.

4.11 Stratégie pour la rétention clients

La rétention clients est très importante pour une entreprise donnée. Dans cet ouvrage, on va présenter un exemple de stratégie en effectuant du clustering sur notre jeu des données. C'est-à-dire, segmenter les clients en 3 groupes pour trouver une meilleure cohésion entre l'offre et la facturation. L'algorithme qu'on a utilisé est le K-mean.

4.11.1 Segmentation client avec le K-mean

Avant la segmentation de nos clients, présentons d'abord leurs dispersions au niveau de nombres total min d'appel durant la soirée et la journée.

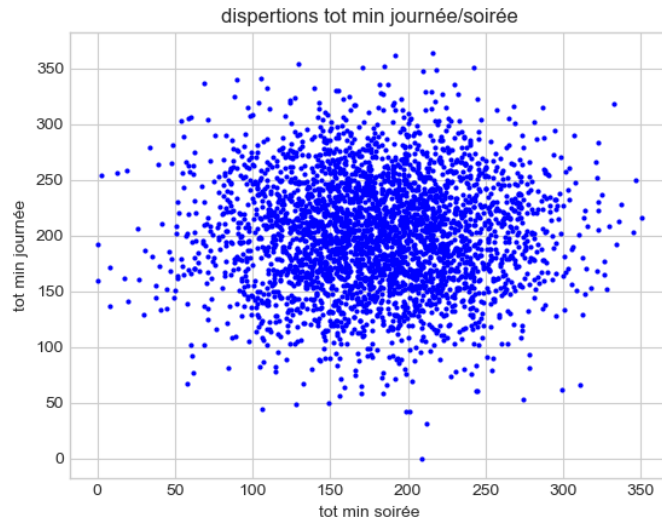


Figure 4.35 : *dispersion clients*

Notre objectif principal est de diviser les clients en trois (3) sous-groupe ou clusters. On regroupe les clients qui ont des comportements similaires et auxquels on pourra adresser une offre plus ciblée.

Chaque classe doit avoir une valeur d'écart type la plus minimale possible et que l'effectif de chaque clusters doit être plus ou moins égale.

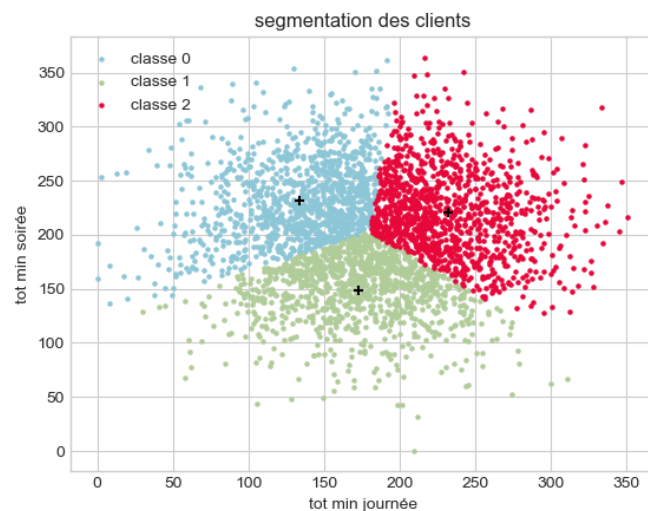


Figure 4.36 : *clients segmentés en trois clusters*

Les clients sont maintenant segmentés en trois clusters par rapport au total minute journée et soirée et aussi on peut voir que chaque centre de cluster se retrouve plus ou moins au milieu de chacune des classes.

classes	Total minute journée	Total minute soirée
0	133.478	232.200
1	172.396	148.884
2	231.729	221.057

Tableau 4.04 :les centres des clusters

4.11.1.1 Proposition de nouvelle offre

En se basant sur les différentes classes de nos clients, proposons maintenant une nouvelle offre plus cohérente pour chacun des utilisateurs.

Le besoin moyen des clients avant la segmentation est présenté par le tableau 4.05 :

tot min journée	fact. journée	tot min soirée	fact soirée	tot min nuit	fact nuit	tot min int	fact int
179.775	30.562	200.980	17.083	200.872	9.040	10.237	2.764

Tableau 4.05 :besoin des clients initialement

classes	Tot min journée	Fact journée	tot min soirée	fact soirée	tot min nuit	fact nuit	tot min int	fact int
Classe 0	231.894	39.422	221.563	18.833	200.020	9.000	10.217	2.760
Classe 1	172.852	29.385	149.156	12.678	201.418	9.063	10.217	2.760
Classe 2	133.390	22.677	232.064	19.725	201.194	9.053	10.278	2.775

Tableau 4.06 :nouvelle offre

On remarque que par rapport à l'offre et la facturation de notre jeu de données initial et celle après segmentation, seules les colonnes concernant la journée et la soirée ont subi un changement notable.

- Pour les utilisateurs dans la classe 0, l'ancien forfait ne les suffise pas, ils ont besoin de plus d'appel (journée, soirée).

- Pour la classe 1 leur appel durant la journée n'a pas beaucoup changé, mais l'appel durant la soirée a largement diminué de 200 à 149.
- Enfin pour la classe 2, ils n'ont pas l'habitude d'effectuer des appels durant la journée c'est pour cela qu'on voit une large diminution des minutes d'appel dans la journée, mais ils ont tendance à appeler durant la soirée.

En ce qui concerne la facturation, nous l'avons fixé par rapport à la proportion de l'augmentation ou de la diminution de la durée d'appel seulement. Ce qui n'est pas suffisant, parce que pour une meilleure stratégie de rétention client, il faut proposer des prix attractifs par rapport au besoin réel du client. D'où pour améliorer notre facturation, des études marketing complètes devront être faites basées sur les données que nous avons fournies.

4.12 Conclusion

L'objectif de ce chapitre était d'analyser les données et de réaliser la prédiction de désabonnement avec les différents algorithmes de machine learning. L'exactitude ne suffit pas pour évaluer un modèle, il faut au moins calculer sa précision et son rappel. Les forêts aléatoires et le réseau de neurone ont été la plus performante en termes de score (95,1% et 96,5% respectivement) mais avec un temps d'exécution plus long. Il faut faire attention lors du paramétrage des algorithmes au risque de faire face au sous apprentissage ou au sur apprentissage. La segmentation est un moyen pour avoir une meilleure cohésion entre les utilisateurs du même groupe, ce qui est nécessaire pour mieux les cibler.

CONCLUSION GENERALE

D'abord, nous avons vu les théories sur les méthodes d'extraction de la connaissance dans le domaine de la science de données et aussi les grands problèmes de machine learning qui sont la régression et la classification.

De plus, on a présenté quelques outils nécessaires pour commencer l'apprentissage automatique.

Ensuite, l'analyse exploratoire de notre jeu de données à l'aide des outils statistiques et des techniques de visualisation nous a fait découvrir les tendances et les comportements des utilisateurs. L'outil de prédiction des taux de désabonnement a été développé en python avec l'utilisation des différentes librairies telles que pandas, NumPy, scikit-learn et enfin keras.

Initialement, toutes les méthodes ont été évaluées sans utilisation de différentes techniques d'amélioration telles que la standardisation des données d'entrée, la recherche de la meilleure combinaison possible des paramètres, la sélection automatique des caractéristiques, ou encore l'utilisation de boosting.

Les résultats comparatifs ont montré une certaine progression des performances pour tous les algorithmes utilisés après leurs améliorations. La segmentation des clients en trois classes par l'algorithme de K-means nous a permis de mieux les cibler afin de créer des nouvelles offres plus ordonnées.

Dans les travaux futurs, il est possible d'effectuer de l'analyse de survie des abonnés ou de systèmes de recommandation pour compléter et améliorer cet ouvrage, ainsi, d'utiliser des jeux de données plus larges et plus détaillés de l'industrie en télécommunication afin de maximiser la signification statistique de nos résultats.

ANNEXE1 EXTRAIT DU CODE SOURCE

```
12  #création fonction pour le ROC
13  def courbe_roc(y_test,y_pred,titre,color):
14      sns.set_style('whitegrid')
15      plt.title(titre, fontsize=22)
16      fpr, tpr, threshold = roc_curve(y_test,y_pred)
17      plt.xticks(fontsize=15)
18      plt.yticks(fontsize=15)
19      plt.plot(fpr,tpr,color)
20      plt.plot([0, 1], [0, 1], 'r--')
21      plt.xlim([0,1])
22      plt.ylim([0,1])
23      plt.ylabel("Taux Vrai Positif", fontsize=18)
24      plt.xlabel("Taux Faux Positif", fontsize=18)

...

26  #création fonction matrice de confusion
27  def matrice_confusion(y_test,y_pred,titre):
28      cnf_metrix = (metrics.confusion_matrix(y_test,y_pred))
29      cmap = sns.cubehelix_palette(50, hue=0.5, rot=0, light=0.9, dark=0, as_cmap=True)
30      sns.set(font_scale=1.8)
31      sns.heatmap(cnf_metrix, cmap=cmap, xticklabels=['Faux', 'Vrai'], yticklabels=['Faux', 'Vrai'],annot=True, fmt="d")
32      plt.title(titre, fontsize=25)
33      plt.xticks(fontsize=18)
34      plt.yticks(fontsize=18)
35      plt.xlabel("prédit", fontsize=20)
36      plt.ylabel("Actuel", fontsize=20)

...

44  #fonction scoring
45  def scoring_fonct(y_test,y_pred):
46      dictionnaire = {}
47      dictionnaire['accuracy'] = metrics.accuracy_score(y_test,y_pred)
48      dictionnaire['recall'] = metrics.recall_score(y_test,y_pred)
49      dictionnaire['precision'] = metrics.precision_score(y_test,y_pred)
50      dictionnaire['f1'] = metrics.f1_score(y_test,y_pred)
51      dictionnaire = pd.Series(dictionnaire)
52      dictionnaire = pd.DataFrame(dictionnaire)
53      dictionnaire.columns = ['scores']
54      dictionnaire['scores en %'] = dictionnaire['scores']*100
55      return dictionnaire
```

ANNEXE 2 SOURCES DES DONNEES PUBLIC

A2.1 Données politiques et gouvernementales

Data.gov

<http://data.gov>

C'est la ressource pour la plupart des données liées au gouvernement.

Socrata

<http://www.socrata.com/resources/>

Socrata est un bon endroit pour explorer les données liées au gouvernement. En outre, il fournit un outil visualisation pour explorer les données.

Bureau du recensement américain

<http://www.census.gov/data.html>

Ce site fournit des informations sur les citoyens américains couvrant les données sur la population, les données géographiques et l'éducation.

A2.2 Données de santé

Healthdata.gov

<https://www.healthdata.gov/>

Ce site fournit des données médicales sur l'épidémiologie et les statistiques démographiques.

Centre d'information sur la santé et les soins sociaux du NHS

<http://www.hscic.gov.uk/home>

Les ensembles de données sur la santé du National Health Service du Royaume-Uni.

A2.3 Données social et grand public

Graphique Facebook

<https://developers.facebook.com/docs/graph-api>

Facebook fournit cette API qui vous permet d'interroger l'énorme quantité d'informations que les utilisateurs sont partager avec le monde.

Kaggle

<https://www.kaggle.com/>

Kaggle propose beaucoup des jeux de donnée grand public qui sont disponible de tous thèmes et propose souvent des concours.

Topsy

<http://topsy.com/>

Topsy fournit une base de données interrogeable de tweets publics remontant à 2006 ainsi que plusieurs outils pour analyser les conversations.

Tendances Google

<http://www.google.com/trends/explore>

Statistiques sur le volume de recherche (en proportion de la recherche totale) pour un terme donné, depuis 2004.

A2.4 Données financières

Google Finance

<https://www.google.com/finance>

Quarante années de données boursières, mises à jour en temps réel.

BIBLIOGRAPHIE

- [1] T. O'Reilly, « *What is Data Science ?* », O'Reilly Media, 10 Avril 2011
- [2] Y. Benzaki, « *Data scientist : du rêve à la réalité* », 2017
- [3] E. Wieringa, « *Unstructured data* », 2016
- [4] J. Crowell, « *The philosophy and process of data science* », Novembre 2016
- [5] D. Cielen, B. Meysman, « *Introducing Data Science* », 2016
- [6] J. Desons, « *Entrepôt de données* », Décembre 2017
- [7] R. Kimball, M. Ross, « *The data warehouse : Guide de conduite de projet* », 23 Février 2005
- [8] F. Nielsen, « *Data Mining with Python* », 29 Novembre 2017
- [9] S. Kumar, « *introduction to data mining* », 2004.
- [10] R. Rakotomalala, « *Introduction au Text Mining : Principes et applications* », 2017
- [11] S. Weiss, N. Indurkha, T. Zhang, « *Text Mining-predictive methods* », Springer, 2005
- [12] S. Stendahl, A. Andersson, G. Strömberg, « *Web Mining* », 2015
- [13] F. Pennerath, « *Data Science* », 2016
- [14] A. de Goursac, « *Le Machine Learning : Envol Vers le prédictif* », MYRIAD 2016.
- [15] S. Olson, S. Raschka, « *Python Machine Learning* », Septembre 2015
- [16] P. Lemberger, M. Batty, M. Morel, J. Raffaëlli, « *Manuel du data scientist* », 2017
- [17] F. Fessant, « *Apprentissage non supervisé* », 28 Septembre 2006
- [18] T. Ayodele, « *Types of Machine Learning Algorithms* », 2017
- [19] F. Santos, « *Arbres de décision* », 27 Mars 2015
- [20] E. Debreuve, T. Morpheme, « *An introduction to random forests* », University Nice Sophia Antipolis, 2016
- [21] R. Rakotomalala, « *Le classifieur bayésien naïf : Modèle d'indépendance Conditionnelle* », 2017

- [22] S. Sayad, « *Logistic Regression* », University of Toronto, 2010
- [23] M. Hasan, F. Boris, « *SVM: Machines à Vecteurs de Support ou Séparateurs à Vastes Marges* », Versailles St Quentin, 2006
- [24] S. Tollari, « *Apprentissage automatique et reduction du nombre de dimensions* », Novembre 2015
- [25] S. Vialle, « *Big Data : informatique pour les données et calculs massifs* », Juin 2017
- [26] C. Gagné, « *Apprentissage et reconnaissance – GIF – 4101/ GIF 7005* », université laval, 30 Novembre 2016
- [27] Y. LeCun, « *Le deep learning, une révolution en intelligence artificielle* », Février 2016
- [28] A.Cornuéjols, « *Evaluation de l'apprentissage* », 2017
- [29] L. Lebart, M. Piron, A. Morineau, « *Visualisation et inférence en fouilles de données* », 2017
- [30] M.Paluszczek, S. Thomas, « *MATLAB Machine Learning* », 2017
- [31] P. Lafayer, « *Le logiciel R : Maitriser les langages, Effectuer des analyses statistique* », 2018
- [32] S. Raschka, « *Python Machine Learning* », 2016
- [33] C.Müller, S.Guido, « *Le Machine Learning avec Python* », 2017
- [34] U. Kamath, K. Shopella, « *Mastering Java Machine Learning* », 2017
- [35] F. Boudin, « *Machine Learning avec WEKA* », 2012
- [36] J. Withanawasam, « *Apache Mahout Essentials* », PACKT, 2015
- [37] N. Pentreath, « *Apprentissage automatique avec Spark* », 2015
- [38] Z. Tejada, « *Mastering Azure Analytics* », O'Reilly Media, 2016
- [39] R. Mattison, « *The telco churn management* », XiT Press, 2005
- [40] A. Delers, « *Gestion de la relation client* », 2016
- [41] F. Nelli, « *Python Data Analytics* », APRESS, 2015

FICHE DE RENSEIGNEMENTS

Nom : MAHANAMANA

Prénom : Andriamiharisoa

Tél : +261 34 48 060 00

E-mail : andriamiharisoa10@gmail.com



Adresse : Lot IAH 44 Bis Avaratsena Itaosy

Titre de mémoire :

**«PREDICTION D'ATTRITION ET RETENTION CLIENTS EN TELECOMMUNICATION
BASEE SUR LE MACHINE LEARNING »**

Mots clés :

Extraction de connaissance, analyse exploratoire, Apprentissage automatique, Classification, segmentation

Nombre de pages : 89

Nombre de tableaux : 10

Nombre de figures : 74

Directeur de mémoire : Mme. ANDRIANTSILAVO Haja Samiarivonjy

Tél : +261 33 14 223 23/+261 34 06 796 96

E-mail :xahajas@yahoo.fr

RESUME

Nous présentons une étude comparative sur quelques algorithmes d'apprentissage automatique appliquées au problème de la prédiction de désabonnement et de la rétention clients dans l'industrie de télécommunication. Le travail consiste à appliquer des prétraitements et de l'analyse exploratoire sur le jeu des données avant l'évaluation et l'amélioration des modèles. Nos meilleurs classificateurs sont l'algorithme des forêts aléatoires et le réseau de neurone avec une exactitude plus de 95%. Pour la segmentation du client nous avons utilisé l'algorithme de K-means afin de proposer des offres plus ordonnées à nos utilisateurs respectifs.

Mots clés :

Extraction de connaissance, analyse exploratoire, Apprentissage automatique, Classification, segmentation

ABSTRACT

We present a comparative study on some machine learning algorithms applied to the problem of churn prediction and customer retention in the telecommunication industry. The work consists of applying preprocessing and exploratory data analysis, before the evaluation and improvement of the models. Our best classifiers are the random forest algorithm and the neuron network with an accuracy of more than 95%. For customer segmentation we used the K-means algorithm to provide more orderly offers to our respective users.

Keywords :

Knowledge Extraction, Exploratory Analysis, Machine Learning, Classification, Clustering