République Algérienne Démocratique et Populaire

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université A. MIRA BEJAIA

جامعة بجاية
Tasdawit n Bgayet
Université de Béjaïa

Faculté des Sciences Exactes
Département Informatique

Mémoire de fin d'études

En vue de l'obtention du diplôme de MASTER professionnel en Informatique.

Parcours : Génie logiciel.

Intitulé du projet :

Prédiction de la perte de clients dans une entreprise commerciale : Candia

---

Réalisé par :

BENZENATI Kenza

CHIKHI Katia

Soutenu le 06/07/2022 devant le jury composé de :

Président	Dr Z.FARAH	Maître de Conf.A	U. A/Mira Béjaia.
Rapporteur	Dr A.SEBAA	Maître de Conf.A	ESTIN, Béjaia.
Examinateur	Dr F.KACIMI	Maître de Conf.B	ESTIN, Béjaia.

Année académique 2021-2022

Résumé

Les entreprises en situation de concurrence doivent être en mesure de prévoir les clients ayant une forte propension à la résiliation en utilisant leurs données de manière plus efficace.

Les modèles de prédiction de l'attrition des clients sont très prometteurs en tant qu'outils puissants pour améliorer la fidélisation des clients.

Notre projet consiste à prédire les clients les plus susceptibles de quitter l'entreprise Tchin-Lait (Candia) en analysant d'abord ses données en utilisant l'outil Microsoft Power Bi et en appliquant les modèles prédictifs les plus populaires , à savoir les réseaux de neurones, machine à vecteurs de support (SVM) et boosting de gradient XGBoost. Les résultats obtenus sur l'ensemble de test ont été évalués à l'aide de la matrice de confusion et la précision. Il a été constaté que le meilleur classificateur était XGBoost avec une précision de 0,92

Mots-clés : - Fidélisation des clients, prédiction de désabonnement des clients, apprentissage automatique, apprentissage profond, apprentissage supervisé, classification binaire.

Abstract

Competitive companies need to be able to predict customers with a high propensity to churn by using their data more effectively.

Customer attrition prediction models hold great promise as powerful tools for improving customer retention.

Our project consists of predicting customers who are most likely to leave the Tchin-Lait (Candia) company by first analyzing its data using Microsoft Power Bi tool, and applying the most popular predictive models, namely neural networks, support vector machine (SVM) and XGBoost gradient boosting. Obtained results on the test set were evaluated using confusion matrix and accuracy. It was found that the best classifier was XGBoost with an accuracy of 0.92

Keywords : - Customers retention, customer churn prediction, machine learning, deep learning, supervised learning, binary classification.

Dédicaces

A mes chers parents,
Qui n'ont ménagé aucun effort, qui ont toujours su m'écouter, me comprendre, partager mes hauts et mes bas et me tirer toujours vers le haut.

A mes frères et soeurs,
Dyhia, Idir et Amir, vous êtes la lumière de ma vie, merci pour tout ce que vous faites pour moi au quotidien, merci pour le soutien que vous m'apportez, c'est grâce à vous que j'ai eu le courage de continuer.

A mon binome Katia CHIKHI,
ta présence et ton écoute ont été le meilleur des soutiens, c'était merveilleux de travailler à tes côtés et d'avoir partager cette experience avec toi.

Kenza BENZENATI

Dédicaces

A mon frère Lounis,
J'espère que, du monde qui est sien maintenant, il apprécie cet humble geste comme preuve de reconnaissance de la part d'une sœur qui a toujours prié pour le salut de son âme. Que Dieu le tout puissant lui accorde sa miséricorde et l'accueille dans son vaste paradis.

Aux meilleurs parents qu'une fille puisse avoir,
ceux qui ont travaillé chaque jour pour me fournir tout ce dont j'aurais besoin et qui n'ont jamais cessé de m'encourager pour que je puisse atteindre mes objectifs et affronter les aléas de la vie.

A mon frère Faouzi,
C'est une chance d'avoir un grand frère tellement amusant, intelligent et toujours à l'écoute comme toi.

A ma soeur Nesrine,
En plus d'être la plus gentille des sœurs, tu es également une véritable amie ,merci d'être toujours là pour moi.

A mon binôme Kenza BENZENATI ,
pour son soutien moral et sa compréhension tout au long de ce projet .

Katia CHIKHI

Remerciements

En premier lieu, nos remerciements vont à notre encadrant, Monsieur Abderrazak SEBAA qui a été d'une aide précieuse, ses remarques pertinentes et son sens du détail nous ont beaucoup aidé.

Nos remerciements vont également aux membres du jury pour avoir accepté d'examiner notre travail et de l'enrichir par leurs propositions.

Nous tenons aussi à remercier tous nos amis pour leur présence et leur soutien moral surtout notre cher ami Hichem AITOUAKLI.

Kenza BENZENATI et Katia CHIKHI

Table des matières

Table des figures

Liste des tableaux

INTRODUCTION GÉNÉRALE

Le phénomène de la perte de clients est répandu dans presque tous les secteurs, il est important de savoir que leurs attentes changent à travers le temps et peuvent se tourner vers la concurrence, d'après un rapport d'Harris Interactive, 89% des consommateurs changent d'entreprise après avoir eu une mauvaise expérience client.[1] Notre projet se focalise sur la perte de la clientèle dans le secteur commercial. Ce secteur introduit généralement des méthodes plus commerciales que informatiques lorsqu'il s'agit de comprendre le comportement de leurs clients, mais dans notre cas on a favorisé les technologies de pointe comme le machine learning et le deep learning pour prédire le désabonnement.

Plusieurs recherches ont prouvé que le machine learning est massivement utile pour l'analyse de données. Il permet de développer, de tester et d'appliquer des algorithmes d'analyse prédictive sur différents types de données afin d'accélérer l'analyse et de la rendre plus précise.

A travers notre mémoire, nous allons nous intéresser de près au phénomène de la perte de client en analysant minutieusement les données qu'on a à notre disposition avec les outils nécessaires, pour comprendre et développer des solutions à ce problème.
Tout d'abord, dans le premier chapitre, nous allons présenter l'entreprise d'accueil, puis évoquer la problématique de notre projet tout en expliquant ce que c'est le désabonnement, la prédiction du désabonnement et ses facteurs. Ensuite, dans le deuxième chapitre, nous allons nous intéresser au Machine learning, ses types d'apprentissages, ses algorithmes et les réseaux de neurones.
Dans le troisième chapitre nous allons voir l'environnement machine, les frameworks et bibliothèques utilisées. Le quatrième chapitre est consacré à la préparation de notre jeu de données, effectivement dans cette partie nous allons procéder au traitement et à la visualisation de nos données, ainsi nous pouvons passer au cinquième et dernier chapitre dans lequel nous avons développé nos modèles de prédiction et observer les résultats obtenus.

CHAPITRE 1

PRÉSENTATION DE L'ENTREPRISE ET OBJECTIF DU TRAVAIL

1.1 Introduction

Dans un monde où la concurrence sur le marché ne cesse de croître, la fidélisation des clients est l'une des questions les plus importantes pour les entreprises, car celles qui ne le font pas verront leurs concurrents, dotés de meilleurs systèmes de fidélisation, les dépasser rapidement, mais les données traditionnelles de satisfaction et les tentatives de sauvetage des clients qui partent ne s'attaquent pas aux causes profondes qui poussent les clients à partir donc les décideurs ont besoin d'avoir la clarté sur les abonnés afin de savoir sur quels facteurs agir pour les fidéliser.

1.2 Présentation de l'entreprise

Depuis 1952, Tchin-Tchin est à l'origine une entreprise familiale, spécialisée dans les boissons gazeuses, de ce fait, elle a capitalisé une longue expérience dans le conditionnement des produits sous forme liquide. L'entrée de grandes multinationales sur le marché des boissons gazeuses et la croissance exponentielle de la limonade locale l'obligent à revoir sa stratégie ; ainsi, l'idée de passer au lait UHT donne naissance à Tchin-Lait qui est une société privée de droit Algérien, constitué juridiquement en SARL (Société A Responsabilité Limités) et fondée par M. Fawzi BERKATI en 1999, située à l'entrée de la ville de Bejaia.
Tchin Lait produit et commercialise le lait longue conservation UHT (Ultra Haute Température) sous le label Candia, depuis mai 2001. Le choix du procédé UHT (lait traité à Ultra Haute Température, permettant une conservation longue durée hors chaîne de froid) résulte du fait que le lait existant en Algérie est un lait frais pasteurisé, il requiert la continuité et la non rupture de la chaîne de froid, depuis son conditionnement jusqu'à sa consommation finale, en passant par son stockage et son transport.
N'étant pas laitier de tradition, Tchin-Lait a opté pour un partenariat avec CANDIA, leader européen du lait. Ce contrat de franchise n'est qu'un partenariat entre l'entreprise Tchin-Lait et CANDIA, où chacune des parties trouve son intérêt : CANDIA peut, grâce aux contrats de franchise, étendre le marché et la notoriété de ses produits à l'échelle internationale ; Tchin-Lait, quant à elle, peut bénéficier du savoir-faire CANDIA pour produire des produits de bonne qualité qui, de plus, sont déjà bien connus du marché.
La gamme de produits Tchin-Lait est constituée de Lait longue conservation, Laits chocolatés , Laits et jus, Poudre Instantanée et Boissons aux fruits.

Elle est dotée d'un capital social de 1.000.000.000 DZD et comporte plus de 580 Salariés.
la figure suivante montre l'organigramme de l'entreprise Tchin-lait :

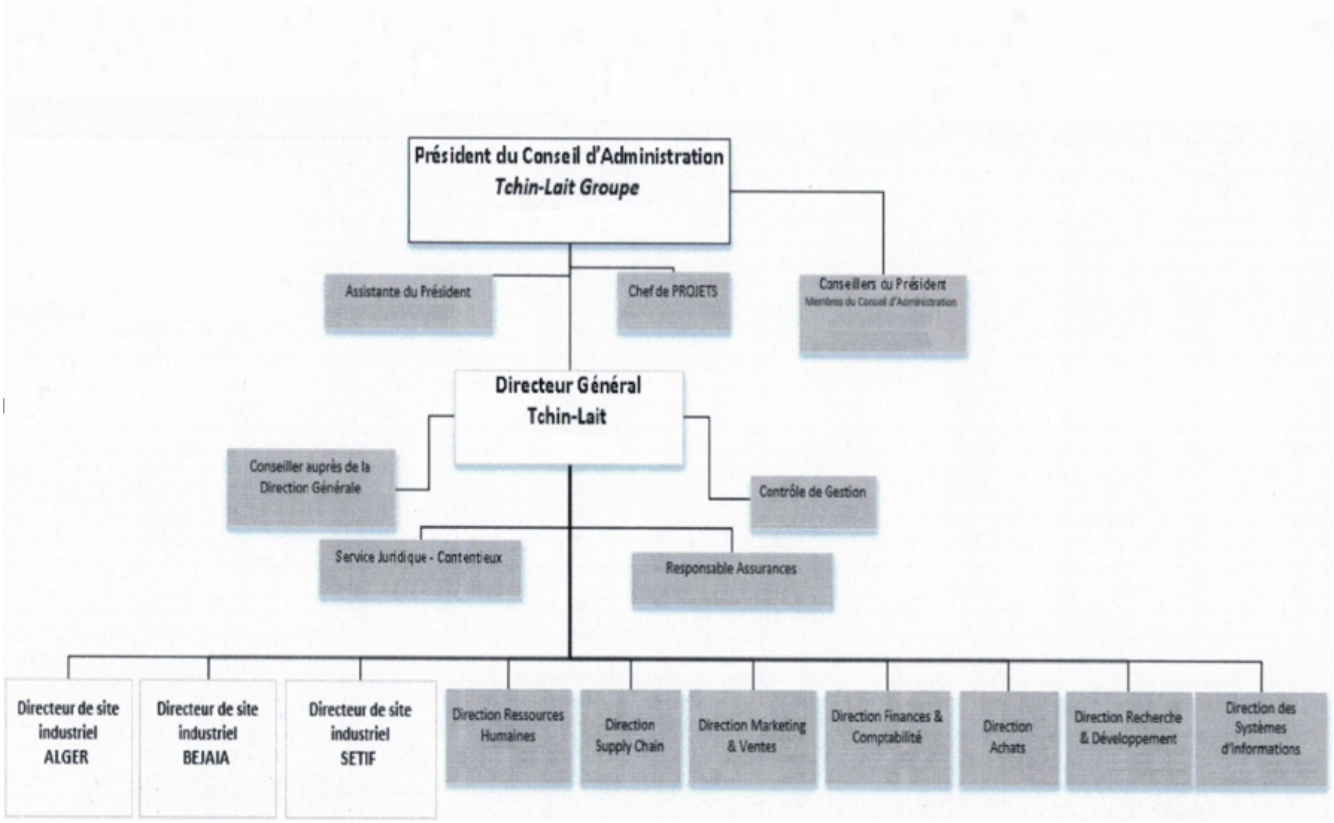


Figure 1.1 – Organigramme de l'entreprise Tchin-lait

Tchin-Lait dispose de 51 clients distributeurs, Ils sont répartis dans les quatres coins de l'Algérie, la région Centre dispose de 12 distributeurs, 4 sont basés à Alger. La figure suivante montre le réseau de distribution des produits de l'entreprise :

Figure 1.2 – Le réseau de distribution des produits de l'entreprise :

1.3 Formulation du problème

1.3.1 Définition de la fidélisation des clients

La fidélisation de la clientèle désigne les activités et les mesures prises par les entreprises et les fournisseurs de services pour inciter les clients existants à continuer d'acheter des produits ou des services auprès d'eux .Une petite amélioration de la fidélisation des clients peut conduire à une augmentation significative des bénéfices.[35]

1.3.2 Définition du désabonnement

Le désabonnement des clients est un terme utilisé pour désigner la situation où le client quitte un produit ou un service d'une entreprise.
En outre, le client abandonne le contrat pour certaines raisons comme les changements dans la situation qui rend impossible pour lui de continuer à demander le service, par exemple des problèmes financiers ou le changement de la localisation géographique, etc[35]
Il existe deux types de désabonnement :

- **Le désabonnement interne** Le client passe à une offre différente mais commercialisée par la même entreprise.
- **Le désabonnement externe** désigne le passage du client d'un fournisseur de services à un autre (concurrent).
  Notre étude concerne le désabonnement externe.

Ce phénomène est généralement mesuré par le taux d'attrition qui constitue un indicateur important pour les entreprises. Ce taux représente le pourcentage de clients perdus sur une période donnée par rapport au nombre total de clients au début de cette période.[24]

1.3.3 Définition de la prédiction de désabonnement des clients

La prédiction de désabonnement consiste à déterminer quels sont les clients les plus susceptibles de résilier un abonnement, c'est-à-dire de quitter une entreprise, en fonction de plusieurs facteurs afin de se concentrer davantage sur eux et savoir sur quels facteurs agir pour les fidéliser.[23]

1.3.4 Facteurs indiquant que les clients vont se désabonner

Le phénomène de l'attrition peut être expliqué par plusieurs facteurs poussant les clients à se désengager. Parmis ces facteurs on a :

- **Le coût :** Les clients recherchent des marchés où les prix des produits sont moins élevés.[32]
- **La satisfaction des clients :** C'est l'attitude globale du client envers un produit ou un service après l'avoir utilisé. Cette satisfaction aide les clients à rester dans l'entreprise et prévient la perte de clients.[32]
- **La qualité du produit :** L'ensemble des propriétés et caractéristiques d'un service ou d'un produit qui lui confèrent l'aptitude à satisfaire des besoins exprimés ou implicites. Elle dépend du nombre d'attentes des clients auxquelles l'organisation répond.
- **Les réclamations :** Nombre de fois où un client dépose une réclamation concernant des problèmes de livraison ou qualité de produit.[37]
- **La durée de l'abonnement :** Nombre de mois pendant lesquels le client est resté dans l'entreprise.
  Il existe d'autres facteurs qui influencent l'attrition de la clientèle comme les réductions, la nature d'activité des clients et d'autres, chaque entreprise a donc le choix de choisir les indicateurs les plus performants et les plus adéquats avec son activité.

1.3.5 Description du problème

La perte des clients constitue un vrai problème pour les entreprises évoluant dans les différents secteurs d'activité, surtout en situation de concurrence car un client perdu est un client gagné pour un concurrent. Dans le cas de notre entreprise, il est nécessaire de fidéliser ses clients surtout pour les années à venir car le marché est en cours d'évolution.
C'est pourquoi des modèles de prédiction de désabonnement à la fois précis et compréhensibles sont nécessaires, afin d'identifier respectivement les clients fragiles qui sont sur le point de se

désabonner et leurs raisons de le faire.
Il s'agit d'un problème de classification binaire où les clients qui sont partis sont séparés de ceux qui sont restés. Afin de résoudre ce problème, l'apprentissage automatique et l'apprentissage profond se sont révélés être des techniques très efficaces pour prévoir des informations sur des données précédentes.

1.4 Conclusion

Ce chapitre nous a servi à présenter la problématique traitée, on a d'abord fait une présentation de l'entreprise d'accueil, ensuite on a défini la fidélisation des clients, le désabonnement, la prédiction du désabonnement ainsi que les facteurs indiquant que les clients vont se désengager, enfin nous avons formulé le problème et les raisons qui nous ont amenées à réaliser ce projet.

CHAPITRE 2

TECHNIQUES DU MACHINE LEARNING ET DU DEEP LEARNING POUR LA PRÉDICTION

2.1 Introduction

Le Machine Learning est un domaine d'étude de l'Intelligence Artificielle, au cours des deux dernières décennies, il est devenu un outil courant dans presque toutes les tâches qui nécessitent l'extraction d'informations à partir de grands ensembles de données. En 1959, Arthur Samuel développe le premier programme de jeu de Dames doté d'une intelligence artificielle. Ce programme avait appris à jouer aux Dames tout seul, sans recevoir la moindre instruction de son développeur, et depuis ce domaine ne cesse d'évoluer et d'imprégner presque tous les domaines. Nous sommes entourés par une technologie basée sur l'apprentissage automatique, les moteurs de recherche apprennent à nous fournir les meilleurs résultats, les appareils photo numériques apprennent à détecter les visages et les voitures sont équipées de systèmes de prévention des accidents conçus à l'aide d'algorithmes d'apprentissage automatique. Cette technologie est également largement utilisée dans des applications scientifiques telles que la bioinformatique, la médecine et l'astronomie.
Depuis l'apparition de l'apprentissage profond la puissance de traitement de données augmente de façon exponentielle, grâce aux réseaux neuronaux artificiels ce domaine a excellé dans les nouvelles frontières que le machine learning ne pouvait dépasser.

2.2 Machine learning

2.2.1 Définition

Le Machine Learning ou apprentissage automatique est un domaine de l'Intelligence Artificielle qui permet à l'ordinateur d'apprendre à partir de l'étude des données, en effet c'est un programme qui analyse les données et apprend à prédire les résultats.

D'après Arthur Lee Samuel le pionnier américain dans le domaine des jeux vidéo et de l'intelligence artificielle : *Le Machine Learning est la science de donner à une machine la capacité d'apprendre, sans la programmer de façon explicite.* [2]

2.2.2 Types d'apprentissage

Le fonctionnement du machine Learning dépend des données dont on dispose, si les données sont étiquetées on aura affaire à un problème d'apprentissage supervisé sinon on sera obligé d'utiliser un algorithme d'apprentissage non supervisé.

2.2.2.1 Apprentissage supervisé

Il consiste à entraîner un modèle en utilisant un ensemble de données étiquetées, cet ensemble a des paramètres d'entrées et de sorties correctes, ce qui lui permet d'obtenir des résultats d'étiquetage précis lorsqu'on lui présente des données jamais vues auparavant (non étiquetées), par exemple La machine peut apprendre à reconnaître une photo d'un animal après qu'on lui ait montré des millions de photos de cet animal. Ou bien, elle peut apprendre à traduire le français en japonais après avoir vu des millions d'exemples de traduction français-japonais.[36]
L'apprentissage supervisé fonctionne en 4 étapes :

- **Préparation de données :** Cette étape consiste à maitre en place un Dataset (x,y) qui contient les exemples (les données), que la machine doit étudier. Il inclut 2 types de variables :
  - Une variable objectif (target) "y".
  - Une ou plusieurs variables caractéristiques (features) "x".

  Il devient possible de prédire de nouvelles valeurs "y" à partir de valeurs de x en développant un modèle.
- **Développement d'un modèle aux paramètres aléatoires :** Il est construit à partir de données, comme un modèle statistique. C'est lui qui effectue les prédictions $f(x)$, sachant que les paramètres sont choisis au hasard.
- **Développement d'une Fonction Coût :** La Fonction Coût évalue la performance du modèle en calculant les erreurs entre les prédictions du modèle $f(x)$ et les valeurs "y" attendues dans le Dataset.
- **Développement d'un algorithme d'apprentissage :** L'étape cruciale du travail, cet algorithme a pour but de trouver le modèle qui minimise la Fonction Coût.

L'apprentissage supervisé peut être séparé en deux types de problèmes lors de l'extraction de données : **la classification** et **la régression** :[2]

2.2.2.1.1 La classification

La classification est une sous-catégorie de l'apprentissage supervisé dont l'objectif est de prédire les étiquettes de classe catégorielle de nouvelles instances sur la base d'observations passées. Ces étiquettes de classe sont discrètes, sa tâche principale est de trouver la fonction du mapping de l'entrée (x) avec la sortie discrète (y).
Il existe quatre types de tâches de classification :

- **Classification binaire :** La classification binaire est considérée comme la tâche la plus simple dans l'étude des algorithmes d'apprentissage automatique.Il s'agit des tâches de classification qui ont deux étiquettes de classe. En général, elles impliquent une classe qui correspond à l'état normal et une autre classe qui correspond à l'état anormal.
  Par exemple, dans le diagnostic médical, un classificateur binaire pour une maladie spécifique pourrait prendre les symptômes d'un patient et prédire si le patient est en bonne santé ou s'il est atteint d'une maladie. Les résultats possibles du diagnostic sont positifs ou négatifs.
- **Classification multi-classes :** Dans la classification multi-classes, plus de deux catégories sont prédéfinies. Il est possible de la décomposer en classifications binaires. Elle fait l'hypothèse que chaque échantillon est affecté à une et une seule étiquette. Contrairement à la classification binaire, la classification multi-classes ne comporte pas la notion de résultats normaux et anormaux. Au lieu de cela, les exemples sont classés comme appartenant à une classe parmi une série de classes connues.
- **Classification multi-étiquettes :** Contrairement aux deux modèles de classification précédents, dans la classification multi-étiquettes, chacune des instances de données est associée à un vecteur de sorties, au lieu d'une seule valeur. La longueur de ce vecteur est

fixée en fonction du nombre d'étiquettes différentes dans l'ensemble de données. Chaque élément du vecteur sera une valeur binaire, indiquant si l'étiquette correspondante est pertinente ou non pour l'échantillon.

— **Classification multidimensionnelle :** Dans la classification multidimensionnelle, un vecteur de sortie est également associé à chaque instance de données. Plutôt qu'une seule valeur.Cependant, chaque élément de ce vecteur peut prendre n'importe quelle valeur parmi un ensemble prédéfini, sans être limité à être binaire. La classification multidimensionnelle est utilisée pour catégoriser des images, de la musique, des textes et des ressources analogues, mais en prédisant pour chaque étiquette une valeur dans un ensemble plus large que la classification multi- étiquettes.

2.2.2.1.2 La régression

La régression est la tâche à laquelle les algorithmes d'apprentissage supervisé sont applicables, c'est un processus qui permet de trouver la relation entre les variables dépendantes et indépendantes grâce à des méthodes mathématiques qui permettent aux data scientistes de prédire un résultat continu (y) en fonction de la valeur d'une ou plusieurs variables prédictives (x). Il existe deux types de régression : la régression univariée, où une seule valeur de sortie est estimée et la régression multivariée où plus d'une valeur de sortie est effectuée. Le modèles de régression les plus courants c'est la régression linéaire.[33]

— **Régression linéaire :** La régression linéaire est un outil statistique d'apprentissage largement utilisé permettant de modéliser la relation entre certaines variables "explicatives" et certains résultats réels. Ce modèle cible les valeurs prédites basées sur des variables indépendantes, Il n'utilise aucune fonction d'activation et ne nécessite pas de valeur seuil.[26]

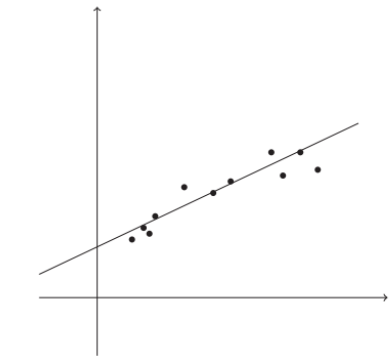


Figure 2.1 – Graphe de regression linéaire.

2.2.2.2 Apprentissage non supervisé

Dans l'apprentissage non supervisé, les données ne sont pas étiquetées. L'objectif est que l'algorithme explore lui-même la structure de données d'entrée afin d'en extraire des informations significatives sans être guidé. Les modèles d'apprentissage non supervisé sont utilisés pour deux tâches principales : le regroupement et la réduction de la dimensionnalité. Il est moins précis que les modèles d'apprentissage supervisé.[25]

2.2.2.2.1 Regroupement (Clustering)

Le clustering est une technique d'analyse exploratoire des données qui permet d'organiser les objets en sous-groupes significatifs (clusters), chaque groupe contient des objets ayant un certain degré de similarité. C'est pourquoi il est aussi parfois appelé classification non

supervisée.
Les algorithmes de clustering les plus courants sont le K-Means, propagation par affinité et modèle de mélange gaussien.[28]

2.2.2.2.2 Réduction de la dimensionnalité

C'est une approche couramment utilisée dans l'apprentissage non supervisé, elle fait référence aux techniques qui réduisent le nombre de variables d'entrée et les comprimer dans un sous-espace de dimension plus petite tout en conservant la plupart des informations pertinentes. Elle est souvent utilisée pour faciliter la visualisation des données. Il existe deux manières d'appliquer la technique de réduction de dimension à savoir : les méthodes de sélection de caractéristiques et les méthodes d'auto-encodage.[27]

2.2.3 Algorithmes d'apprentissage automatique

2.2.3.1 Machine à vecteurs de support SVM

Machine à vecteurs de support est un algorithme d'apprentissage automatique supervisé, préféré pour la classification mais il est aussi très utile pour la régression. L'objectif de SVM est de trouver un hyperplan qui crée une frontière entre les types de données c'est-à-dire de créer la meilleure ligne de décision capable de séparer l'espace à n dimensions en classes afin que nous puissions facilement placer le nouveau point de données dans la bonne catégorie à l'avenir, et fonctionne très bien avec une quantité limitée de données à analyser.

2.2.3.2 eXtreme Gradient Boosting XGBoost

XGBoost (eXtreme Gradient Boosting) est un algorithme d'apprentissage supervisé qui utilise des arbres de décision peu profonds construits séquentiellement et applique le principe d'amplification des apprenants faibles en utilisant l'architecture de descente de gradient pour fournir des résultats précis et une méthode de formation hautement évolutive qui évite le surajustement.Cet algorithme offre une meilleure combinaison de performances de prédiction et de temps de traitement par rapport à d'autres algorithmes.

2.3 Deep learning

2.3.1 Historique

Les premiers réseaux de neurones ont été inventées en 1943 par deux mathématiciens et neuroscientifiques du nom de Warren McCulloch et Walter Pitts, dans leurs article scientifique intitulé " a logical calculus of the ideas immanent in nervous activity", ils ont pu expliquer comment ils ont réussi à programmer des neurones artificiels en s'inspirant du fonctionnement des neurones biologiques, leur modèles n'étaient conçus que pour traiter des entrées logiques.[2]
En 1957, un psychologue américain qui porte le nom de Frank Rosenblatt trouva comment améliorer ce modèle en proposant le premier algorithme de l'apprentissage de l'histoire du deep learning basé sur le perceptron qui s'appuie sur la théorie de Hubb, mais son modèle était linéaire. [2]
C'est quelques années après que l'un des pères du deep learning Geoffrey Hinton développa le perceptron multicouches, le premier véritable réseau de neurones artificiels développé en 4 étapes qui sont : forward propagation, le calcul de la fonction coût, rétropropagation et corriger chaque paramètre du modèle grâce à la descente de gradient.

Le modèle perceptron multicouches continue d'évoluer avec l'apparition de fonctions d'activation car elles offrent de meilleures performances. Dans les années 1990, le chercheur Yann LeCun invente les premiers réseaux de neurones qui permettent de reconnaître et de traiter les images en introduisant des filtres mathématiques appelés convolution et pulling.
Actuellement, les réseaux de neurones sont largement utilisés dans divers domaines de l'IA surtout grâce à l'apparition de l'apprentissage profond, et cette technologie puissante ne cesse d'évoluer.

2.3.2 Définition

L'apprentissage profond est un type d'intelligence artificielle dérivé de l'apprentissage automatique. Il utilise des algorithmes complexes et des réseaux de neurones pour former un modèle. Contrairement au machine learning, l'apprentissage profond peut fonctionner avec des données non structurées et il est capable de se former de façon autonome, Il améliore également de lui-même ses prévisions et ses prises de décision, sans qu'aucune intervention humaine ne soit requise.
Le terme "profond" est approprié, il fait référence au grand nombre de couches potentiellement utilisées.

2.3.3 Les réseaux de neurones

L'apprentissage profond est inspiré par la structure du cerveau humain, en termes de deep learning cette structure s'appelle un réseau de neurones artificiels. Elle est disposée en plusieurs couches interconnectées entre elles. La première couche correspond aux neurones d'entrée et la dernière transmet les résultats de sortie. Entre les deux se trouvent plusieurs couches intermédiaires par lesquelles l'information est traitée.[3]

2.3.4 Le perceptron

C'est un modèle mathématique d'un neurone biologique, qui fait la classification binaire en séparant linéairement deux classes de données, c'est l'unité de base du réseau de neurone artificiel. [30]

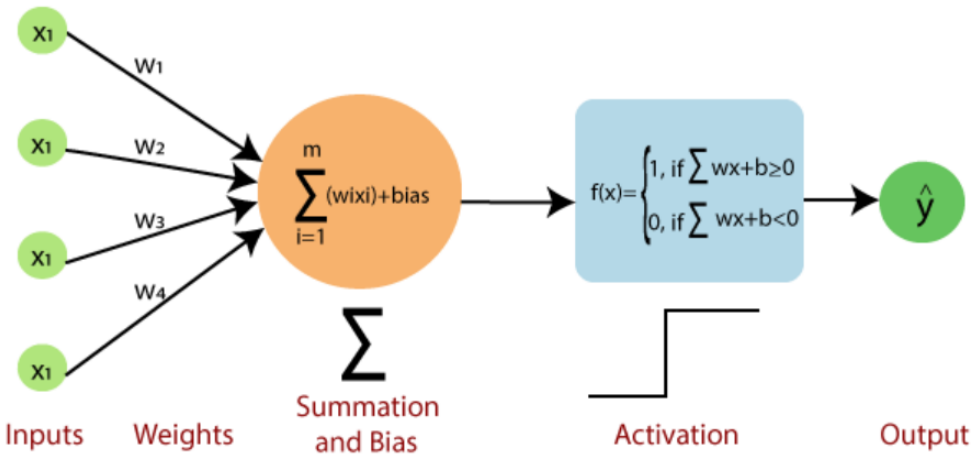


Figure 2.2 – Le perceptron

Un perceptron fonctionne en prenant des entrées. Il les multiplie ensuite avec les poids respectifs. Ces produits sont ensuite additionnés avec le biais. La fonction d'activation

prend la somme pondérée et le biais comme entrées et renvoie une sortie finale. L'apprentissage se produit en modifiant les poids reliant les neurones.

2.3.4.1 Types de perceptrons

— **Le perceptron simple :** Il ne dispose que de deux couches ; la couche en entrée et la couche en sortie. Ce modèle peut apprendre uniquement des fonctions linéaires séparables ce qui fait qu'il reste basique et limité dans ses applications.
— **Le perceptron multicouches :** Le perceptron multicouches permet à un réseau neuronal d'effectuer des mappings non-linéaires et arbitraires, offrant une puissance de calcul supérieure et complexe. Il se compose d'au moins trois couches : une couche d'entrée, une couche cachée et une couche de sortie.

2.3.4.2 Les fonctions d'activation

Elle définit la sortie d'un neurone en fonction d'un ensemble d'entrées, cela signifie qu'il décidera si l'entrée du neurone est importante ou non dans le processus de prédiction .Son but principal est d'introduire la non-linéarité dans le réseau.
Dans le deep learning, un réseau de neurones sans fonction d'activation n'est qu'un modèle de régression linéaire.
Il existe une variété de fonctions d'activation non linéaires qui peuvent être utilisées mais les plus populaires sont :
— **La fonction logistique (Sigmoïde) :** La fonction Sigmoïde donne une valeur entre 0 et 1, utile comme fonction d'activation lorsque l'on s'intéresse aux probabilités plutôt qu'aux valeurs précises, elle est donc très utilisée pour les classifications binaires. Elle est moins efficace pour une utilisation pour les couches cachées. [4]
Le résultat est entre 0 et 1.

$$\boxed{Sigmoid(x) = \frac{1}{(1+exp(-x))}}$$

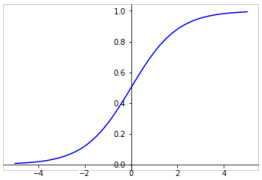


Figure 2.3 – Graphe de la fonction sigmoide
[4]

— **La fonction ReLU (Rectified Linear Unit) :** Elle permet un entraînement plus rapide comparé aux fonctions sigmoid et tanh, c'est la fonction d'activation la plus simple et la plus utilisée surtout pour les perceptron multicouches. [4]
Elle retourne x si x est supérieur à 0, 0 sinon.

$$\boxed{ReLU(x) = max(x, 0)}$$

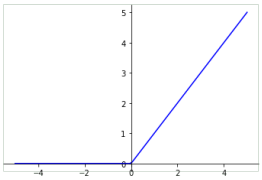


Figure 2.4 – Graphe de la fonction ReLU
[4]

— **la fonction TanH :** La fonction tanh est la fonction de la tangente hyperbolique. Cette fonction est comme la fonction Sigmoïde, utilisée dans la classification binaire. Il s'agit en fait d'une version mathématiquement décalée de la fonction sigmoïde, Tanh fonctionne mieux que celle ci dans la plupart des cas.[4]
Le résultat est entre -1 et 1

$$tanh(x) = \frac{(exp(x)\check{}exp(-x))}{(exp(x)+exp(-x))}$$

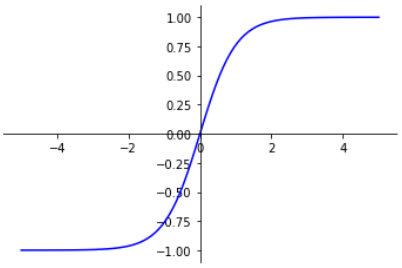


Figure 2.5 – Graphe de la fonction Tanh
[4]

2.3.4.3 La fonction de perte

Presque tous les algorithmes d'apprentissage des réseaux neuronaux sont formulés en utilisant une fonction de perte. Elle mesure la différence entre la réponse correcte et la réponse prédite par l'algorithme. Si les prévisions s'écartent trop des résultats réels, la fonction de perte produira un très grand nombre. S'ils sont assez bons, elle produira un nombre inférieur.[34]

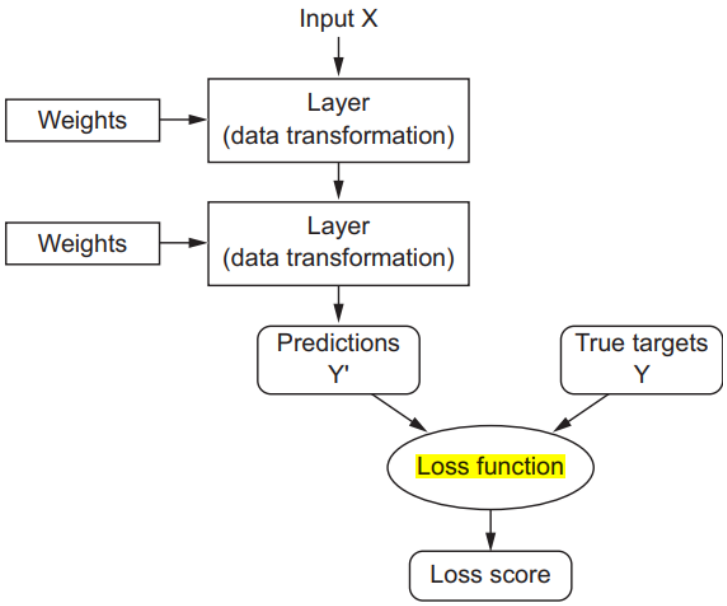


Figure 2.6 – Fonction de perte
[31]

L'entropie croisée et l'erreur quadratique moyenne sont les deux principales fonctions de perte à utiliser lors de la formation de modèles de réseaux de neurones.[5]

— **L'entropie croisée :** Elle peut être appliquée à la fois aux problèmes de classification binaire et multi-classes.
Sa formule mathématique est :

$$\boxed{H_p(q) = \frac{-1}{n} \sum_{i=1}^{n} Y_i.log(p(Y_i)) + (1 - Y_i).log(1 - p(Y_i))}$$

— **l'erreur quadratique moyenne :** c'est l'un des meilleurs choix de fonctions de perte pour la régression.

$$\boxed{MSE = \frac{1}{n} \sum(\text{Y-}\hat{\text{Y}})^2}$$

2.3.4.4 Descente de gradient

Est un algorithme d'optimisation utilisé lors de la formation d'un modèle d'apprentissage automatique .Il permet de trouver les poids qui minimisent autant que possible notre fonction de perte Ce processus nécessite un certain nombre d'itérations jusqu'à ce que la sortie de la fonction de perte soit dans la plage souhaitée. [29]

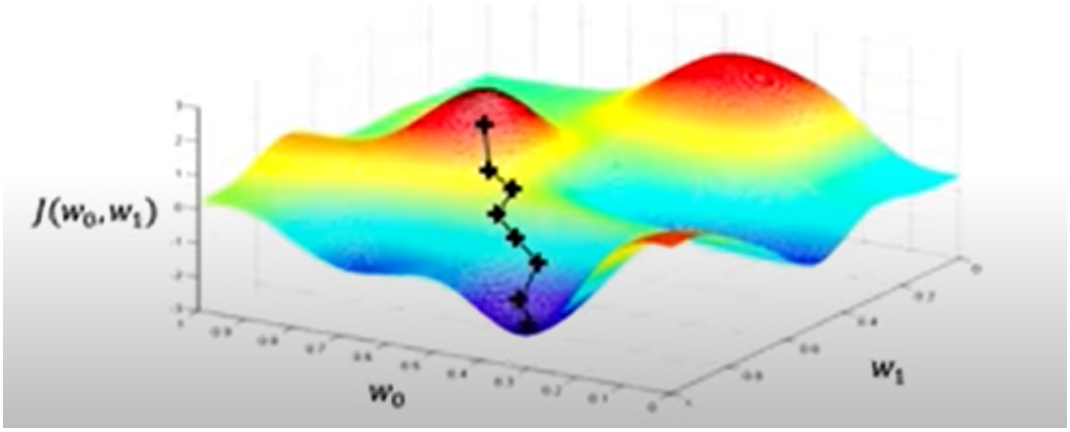


Figure 2.7 – Descente de gradient
[5]

Son idée principale est de descendre jusqu'à ce qu'un coût minimum local ou global soit atteint. À chaque itération, nous faisons un pas dans la direction opposée du gradient où la taille de l'étape est déterminée par la valeur du taux d'apprentissage, ainsi que la pente du gradient.

Algorithm 1 l'algorithme de descente de grqdient

1. initialiser les poids de façon aléatoire
repeat
   2. calculer le gradient $\frac{\delta \mathbf{J}(\mathbf{W})}{\delta \mathbf{W}}$
   3. mettre à jour les poids $\mathbf{W} \leftarrow \mathbf{W} - \eta \frac{\delta \mathbf{J}(\mathbf{W})}{\delta \mathbf{W}}$
until Convergence
4. retourner les poids.

2.3.4.4.1 Rétropropagation du gradient

La méthode d'apprentissage dans son ensemble est généralement connue sous le nom de rétropropagation. C'est la façon dont les ajustements des poids requis par l'algorithme d'optimisation sont propagés dans le réseau neuronal de la dernière couche vers la première. Elle fait référence à la méthode de calcul du gradient.[6]

2.3.4.5 Taux d'apprentissage

Learning rate en anglais, souvent noté α ou parfois η, c'est l'hyperparamètre le plus important lors de la configuration du réseau neuronal, il indique la vitesse à laquelle les coefficients évoluent. Cette quantité peut être fixe ou variable. L'une des techniques les plus populaires s'appelle Adam, qui signifie « Adaptive Moment Estimation » qui a un taux d'apprentissage qui s'adapte au fil du temps, et contrôle le degré de modification du modèle en réponse à l'erreur estimée chaque fois que les poids du modèle sont mis à jour. [7]

2.3.5 problème de surajustement(Overfitting)

Il signifie qu'un réseau neuronal fournira d'excellentes performances de prédiction sur les données d'entraînement sur lesquelles il est construit, mais sera peu performant sur des instances de test non vues.

De nombreuses stratégies connues sous le nom de régularisation sont explicitement conçues pour résoudre ce problème .

2.3.5.1 La regularisation

La régularisation est une technique qui ajoute des informations à un modèle afin d'éviter l'apparition d'un surajustement. Ce processus consiste à minimiser une fonction de coût pour pénaliser les modèles complexes et surajustés. Il est considéré comme un type de régression qui minimise les estimations des coefficients à zéro afin de réduire la taille d'un modèle ceci implique la suppression des poids supplémentaires.
En décourageant l'utilisation de modèles très complexes et flexibles, le risque d'overfitting (surajustement) est réduit.

— **Le décrochage (Dropout) :**
Est une stratégie de régularisation puissante et peu coûteuse en termes de calcul.Elle modifie le réseau pour l'empêcher de se sur-ajuster, en supprimant au hasard des nœuds ainsi que toutes leurs connexions entrantes et sortantes pendant la formation. elle peut également être considéré comme une technique d'ensemble dans l'apprentissage.

— **L'arrêt précoce :**
C'est l'une des formes de régularisation les plus utilisées en apprentissage profond. Sa popularité est due à son efficacité et à sa simplicité. elle s'applique lorsque nous entraînons un modèle avec une méthode itérative, telle que Gradient Descent ,elle consiste à garder une partie de l'ensemble d'apprentissage comme ensemble de données de validation et à arrêter l'entraînement au moment où les performances sur cet ensemble commencent à se dégrader.

2.4 Conclusion

Dans ce deuxième chapitre, nous avons présenté de manière globale l'apprentissage automatique et ses différents points sous jacent, des définitions et algorithmes de ce domaine , comme nous avons pu parler de l'apprentissage profond et des reseaux de neurones qui sont un sujet de recherche depuis plusieurs années.
Nous concluons que l'apprentissage profond a excellé dans les nouvelles frontières où l'apprentissage automatique prenait du retard.

CHAPITRE 3

OUTILS ET FRAMEWORKS

3.1 Introduction

La création d'un modèle prédictif performant nécessite le choix des technologies adéquates. Dans ce chapitre nous allons définir les différents outils et bibliothèques utilisés pour la visualisation et l'implémentation de nos modèles.

3.2 Environnement machine

- **HP EliteBook 840 G5 :**
  - **Système d'exploitation :** Windows 10 Professionnel 64 bits
  - **Processeur :** Intel(R) Core(TM) i5-7300U CPU @ 2.60GHz.
  - **Mémoire :** 8 GB.
  - **Carte mère :** HP 83B2 KBC version 23.53.00
- **DELL Latitude 7280 :**
  - **Système d'exploitation :** Windows 10 Professionnel 64 bits
  - **Processeur :** Intel(R) Core(TM) i5-6300U CPU @ 2.40GHz 2.50GHz.
  - **Mémoire :** 8 GB.
  - **Carte mère :** : 0KK5D1 version A00.

3.3 Langage de programmation

3.3.1 Python

Python est un langage de programmation interprété de haut niveau, créé par Guido van Rossum en 1991. Il est largement utilisé par la communauté des développeurs en raison de la productivité accrue qu'il offre . Il prend en charge plusieurs paradigmes de programmation, y compris la programmation structurée, orientée objet et fonctionnelle. Python est utilisé dans tous les domaines, de l'apprentissage automatique à la création de sites Web, en passant par le test de logiciels. Il est multiplateforme et ses programmes peuvent s'exécuter sous Windows, Linux et macOS.

Figure 3.1 – Python [8]

3.4 Outils d'implémentation et de visualisation

3.4.1 Anaconda

Anaconda est une distribution libre et open source des deux langages de programmation Python et R , elle contient plus de 8 000 packages open source de science des données et d'apprentissage automatique tels que numpy, pandas, scipy, sklearn, tensorflow, pytorch, matplotlib, construits et compilés par Anaconda pour tous les principaux systèmes d'exploitation et architectures.

Figure 3.2 – Anaconda [9]

— **Le navigateur anaconda :** est une interface utilisateur graphique (GUI) de bureau incluse dans la distribution Anaconda qui nous permet de lancer des applications comme Jupiter Lab, VSCode, RStudio et de gérer facilement les packages.

3.4.1.1 Jupyter notebook

Jupyter Notebook est une application Web open source qui offre aux utilisateurs la création et le partage des documents comprenant du code, des équations et d'autres ressources multimédias. Il est utilisé pour toutes sortes de tâches de science des données telles que l'analyse exploratoire, le nettoyage, la transformation, la visualisation et la modélisation statistique des données. L'un de ses avantages majeurs est qu'il peut être converti en un certain nombre de formats de sortie standard tel le HTML, Powerpoint, LaTeX, PDF, ReStructuredText, Markdown et Python via l'interface Web. Cette flexibilité permet aux data scientists de partager facilement leur travail avec d'autres.

Figure 3.3 – Jupyter [10]

3.4.1.2 Spyder IDE

Spyder est un environnement de développement intégré (IDE) gratuit inclus avec Anaconda, écrit en Python, pour Python. Il comprend des fonctionnalités d'édition, de test interactif, de débogage et d'introspection. C'est un outil de développement complet avec l'exploration de données, l'exécution interactive, l'inspection approfondie et les capacités de visualisation d'un package scientifique.

Figure 3.4 – Spyder IDE [11]

3.4.2 Microsoft Power Bi

Selon Gartner, Power BI est le principal outil de l'informatique décisionnelle. Plus de 97% des entreprises l'utilisent . C'est un ensemble de services logiciels, d'applications et de connecteurs qui fonctionnent ensemble pour explorer les données pour obtenir des informations cohérentes, visuellement immersives et interactives, ces données peuvent être une feuille de calcul Excel ou une collection d'entrepôts de données hybrides basés sur le cloud et sur site. Il a deux composants : bureau et service. Dans notre projet nous avons utilisé le composant bureau c'est à dire Power Bi Desktop.

Figure 3.5 – Power BI [12]

— **Microsoft Power BI Desktop :** C'est une application gratuite à installer sur l'ordinateur local utilisée pour l'analyse de données et de création de rapports. Elle comporte trois vues :
  — **Rapport :** C'est dans cette vue qu'on crée des rapports et des visuels.
  — **Données :** dans cette vue, On trouve les données utilisées dans le modèle associé à notre rapport.
  — **Modèle :** dans cette vue, on gère les relations entre les tables de notre modèle de données.

3.4.3 Microsoft Excel

Excel est un logiciel créé par Microsoft qui permet de stocker et d'analyser tout type de données ainsi que l'insertion d'images, de formes, de graphiques, de tableaux croisés dynamiques dans les feuilles Excel pour comprendre les données. Excel est utilisé dans presque toutes les industries comme le secteur financier, bancaire, l'analyse de données et bien d'autres.

Figure 3.6 – Excel [13]

3.5 Bibliothèques

3.5.1 NumPy

Signifie Python numérique. C'est une bibliothèque Python open source à usage général qui fournit des outils pour gérer les tableaux à n dimensions. NumPy offre à la fois la flexibilité de Python et la vitesse d'un code C compilé bien optimisé. Sa syntaxe facile à utiliser le rend hautement accessible et productif pour les programmeurs de tous horizons.

Figure 3.7 – NumPy [14]

3.5.2 Pandas

Le nom "Pandas" fait référence à la fois à "Panel Data" et à "Python Data Analysis" et a été créé par Wes McKinney en 2008. Elle fait partie des bibliothèques logicielles de base pour la science de données en python. Elle offre des structures de données puissantes, expressives et flexibles qui facilitent la manipulation et l'analyse des données.

Figure 3.8 – Pandas [15]

3.5.3 Seaborn

Seaborn est une bibliothèque de la visualisation de données en langage Python basée sur matplotlib. Cet outil permet de créer des graphiques statistiques attrayants et informatifs grace aux différents styles et palettes de couleur par défaut et s'intègre avec les structures Pandas, il permet d'explorer et de comprendre rapidement les données et offre différents types de visualisation. L'un des avantages de Seaborn est l'intégration renforcée avec Pandas, il est mieux intégré que Matplotlib pour travailler avec les data frames de Pandas.

Figure 3.9 – Seaborn [16]

3.5.4 Matplotlib

Matplotlib est une bibliothèque open source ,utilisée pour créer des visualisations statiques, et interactives en Python.Elle a été créée par John D. Hunter en 2002, principalement écrite en python avec quelques segments écrits en C, Objective-C et Javascript. Elle offre des tracés de qualité, des figures interactives qui peuvent zoomer, faire un panoramique, mettre à jour, en plus de l'utilisation d'un large éventail de packages tiers construits sur Matplotlib. Cette bibliothèque offre une flexibilité accrue, c'est la meilleure option lorsque les performances sont parfois supérieures.

Figure 3.10 – Matplotlib [17]

3.5.5 Tensorflow

Tensorflow est une bibliothèque open source développée par google et dédiée à l'apprentissage automatique. Elle offre un écosystème complet et flexible d'outils, de bibliothèques et de ressources communautaires permettant aux développeurs de créer et de déployer facilement des applications avec le langage python. Son principal avantage est l'abstraction, il permet la création des graphiques de flux de données, des structures qui décrivent la manière dont les données se déplacent dans un graphique ou une série de nœuds de traitement.

Figure 3.11 – Tensorflow [18]

3.5.6 Keras

Keras est une bibliothèque de logiciels open source écrite en python, s'exécutant sur la plate-forme d'apprentissage automatique TensorFlow. Elle a été développée par Google pour la mise en œuvre de réseaux de neurones. La raison principale de l'utiliser découle de sa convivialité. Elle est Composée d'une bibliothèque de composants d'apprentissage automatique couramment utilisés, notamment des fonctions d'activation et des optimiseurs, l'API Keras offre également une prise en charge des réseaux de neurones récurrents et convolutifs.

Figure 3.12 – Keras [19]

3.5.7 Scikit-learn

Scikit-learn est une bibliothèque open source écrite en Python, facilement interopérable avec les bibliothèques NumPy, pandas et matplotlib.Elle fournit un ensemble d'outils efficaces pour l'apprentissage automatique, notamment la classification, la régression, le regroupement et la réduction de la dimensionnalité.

Figure 3.13 – Scikit-learn [20]

3.5.8 XGBoost

XGBoost est une bibliothèque distribuée optimisée d'amplification de gradient conçue pour être très efficace , flexible et portable. Elle fonctionne sous Linux, Windows,macOS et implémente des algorithmes d'apprentissage automatique dans le cadre du Gradient Boosting. XGBoost fournit un boost d'arborescence parallèle qui résout de nombreux problèmes de science des données de manière rapide et précise.

Figure 3.14 – XGBoost [21]

3.5.9 Streamlit

Streamlit est une bibliothèque Python open source qui facilite la création, le partage et le déploiement de puissantes applications Web personnalisées pour l'apprentissage automatique et la science des données. Cet outil est compatible avec les principales bibliothèques Python telles que scikit-learn, Keras, PyTorch, SymPy(latex), NumPy, pandas, Matplotlib et d'autres. Avec Streamlit il n'est plus nécessaire d'écrire un backend, de définir des routes, de gérer des requêtes HTTP, d'écrire du HTML, du CSS, du JavaScript, tout est en Python pur. Aucune expérience frontale n'est requise.

Figure 3.15 – Streamlit [22]

3.6 Conclusion

Dans ce chapitre nous avons pu voir les outils et bibliothèques utilisés soit pour la visualisation ou pour l'implémentation de notre projet.

CHAPITRE 4

MÉTHODOLOGIE ET PRÉPARATION DE DONNÉES

4.1 Introduction

Ce chapitre est consacré à la partie préparation et visualisation de données.
Le but de cette étape cruciale est de s'assurer que les données utilisées dans la construction des modèles d'apprentissage automatique produisent des résultats fiables, Cela implique de nombreuses tâches discrètes telles que la collecte, le prétraitement initial, le nettoyage, l'identification des incohérences pour ensuite construire et valider des hypothèses.

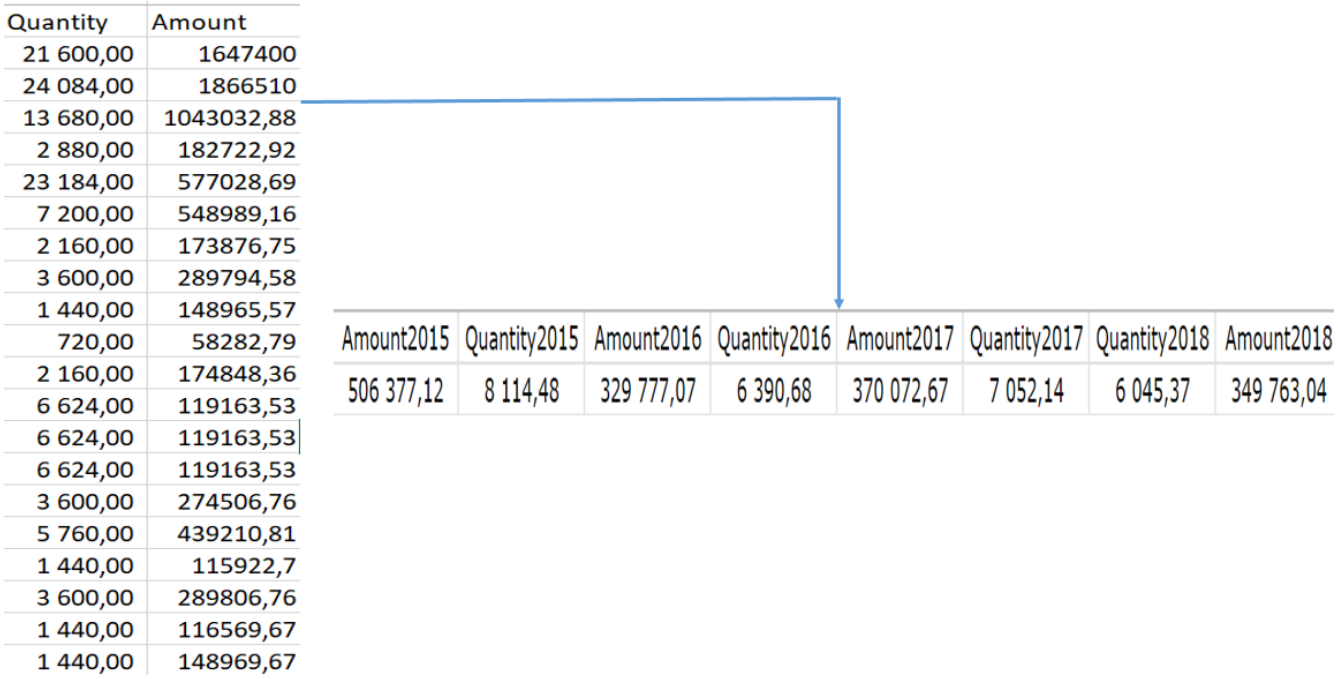
4.2 Traitement de l'ensemble de données

Pour notre étude, l'entreprise nous a fourni 2 bases de données se présentant comme suit :

- Une table Excel contenant les informations concernant les clients de 2015 à 2020.
- Une autre table Excel contenant la liste des achats effectués par les clients de 2015 à 2020.

Afin de bien visualiser nos données, nous avons pensé à créer une seule table "Customers" qu'on a utilisé comme data frame pour notre étude, qui contient les informations pertinentes des deux tables précédentes dans laquelle chaque ligne représente un client unique.
La table des ventes contient plusieurs lignes qui représentent le même client, donc pour remédier à ce problème :

- Nous avons utilisé les tableaux dynamiques croisés et les filtres avancés d'Excel pour calculer les moyennes des variables numériques pour chaque année.

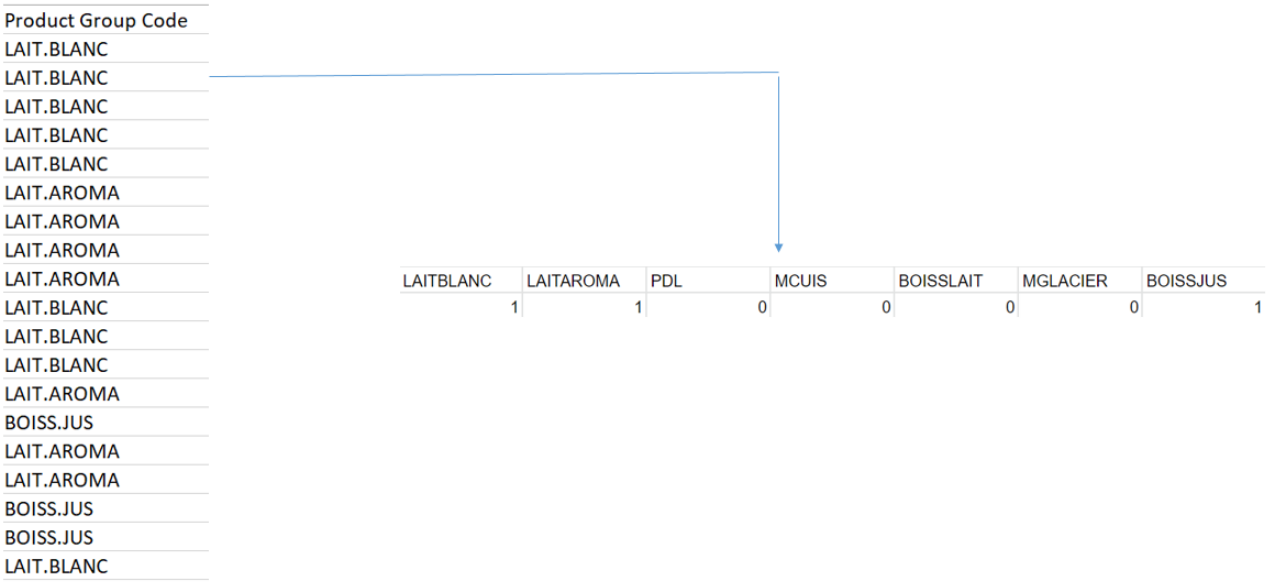


Quantity	Amount
21 600,00	1647400
24 084,00	1866510
13 680,00	1043032,88
2 880,00	182722,92
23 184,00	577028,69
7 200,00	548989,16
2 160,00	173876,75
3 600,00	289794,58
1 440,00	148965,57
720,00	58282,79
2 160,00	174848,36
6 624,00	119163,53
6 624,00	119163,53
6 624,00	119163,53
3 600,00	274506,76
5 760,00	439210,81
1 440,00	115922,7
3 600,00	289806,76
1 440,00	116569,67
1 440,00	148969,67

Amount2015	Quantity2015	Amount2016	Quantity2016	Amount2017	Quantity2017	Quantity2018	Amount2018
506 377,12	8 114,48	329 777,07	6 390,68	370 072,67	7 052,14	6 045,37	349 763,04

Figure 4.1 – Transformation des deux variables Amount et Quantity.

— Pour les colonnes ayant des valeurs catégorielles on a créé des variables qui portent le nom des valeurs et on a complété les cellules par des 0 ou 1. Par exemple : " Product Group Code" on l'a transformé en 7 colonnes, chaque colonne porte le nom d'un type de produit,si le client l'a déjà acheté on met sa valeur à 1 sinon 0.



Product Group Code
LAIT.BLANC
LAIT.BLANC
LAIT.BLANC
LAIT.BLANC
LAIT.BLANC
LAIT.AROMA
LAIT.AROMA
LAIT.AROMA
LAIT.AROMA
LAIT.BLANC
LAIT.BLANC
LAIT.BLANC
LAIT.AROMA
BOISS.JUS
LAIT.AROMA
LAIT.AROMA
BOISS.JUS
BOISS.JUS
LAIT.BLANC

LAITBLANC	LAITAROMA	PDL	MCUIS	BOISSLAIT	MGLACIER	BOISSJUS
1	1	0	0	0	0	1

Figure 4.2 – Transformation des variables catégorielles.

- Afin d'améliorer notre modèle de prédiction, nous avons ajouté d'autres variables à partir d'informations de colonnes déjà existantes comme " Kilométrage" qu'on a calculé à partir de "Région", et "tenure" qu'on a calculé à partir des dates de transaction.
- Pour pouvoir créer un modèle prédictif, nous devons d'abord créer la variable cible , pour cela

nous avons ajouté une colonne nommée “Churn” qui a la valeur “yes” si le client a quitté l’entreprise et “no” sinon (selon un manager de l’entreprise si un client n’effectue aucune transaction en décembre, il est considéré comme étant un churner).

— Notre ensemble de données contient 189 lignes et 40 variables, ou chaque ligne représente un client. Le tableau ci-dessous explique le rôle de chaque variable.

Liste des variables		
Nom de la variable	Description	Type
Customer	Il représente l'identifiant du client	Numérique
Nom	il représente le nom du client	chaine de caractère
Région	la position géographique du client	chaine de caractère
Code postal	le code postal de la wilaya du client	Numérique
Kilometrage	la distance entre l'usine de béjaïa et le client	Numérique
NTéléphone	le numéro de téléphone du client	Numérique
E-mail	l'email du client	chaine de caractère
GroupeComptaClient	groupe comptabilité client	chaine de caractère
GroupeComptaMarché	groupe comptabilité marché si il est algérien il prend la valeur Local, sinon Etranger.	chaine de caractère
GroupeComptaMarchéTVA	Groupe comptabilité du marché TVA il prend la valeur ASSUJETTI ou EXONERE	chaine de caractère
CodeConditionsPaiement	dans notre cas, le paiement est comptant c'est-à-dire un paiement réalisé en une seule fois.	chaine de caractère
CodeDevise	DZD si le client est local, EUR sinon	chaine de caractère
CodeActivité	Code d'activité du client	Numérique
NatureActivity	La nature de l'activité du client	chaine de caractère
CodeConditionLivraison	on a 3 codes : virement bancaire (VIR), postal(VIRP) et par chèque(CHEQ)	chaine de caractère
Churn	"yes" si le client a quitté, "no" sinon	chaine de caractère
VAT0	Taxe sur valeur ajoutée= 0	Numérique
VAT17	Taxe sur valeur ajoutée= 17	Numérique
VAT19	Taxe sur valeur ajoutée= 19	Numérique

LineDiscountMoyP	Moyenne de remise pour chaque client	Numérique
LAITBLANC	“1” si le client a acheté un produit de type lait blanc ,“0” sinon	Numérique
LAITAROMA	“1” si le client a acheté un produit de type lait aromatisé ,“0” sinon	Numérique
PDL	“1” si le client a acheté un produit de type lait en poudre ,“0” sinon	Numérique
MCUIS.	“1” si le client a acheté un produit de type Maître cuisinier ,“0” sinon	Numérique
BOISSLAIT	“1” si le client a acheté un produit de type boisson lait,“0” sinon	Numérique
MGLACIER	“1” si le client a acheté un produit de type Maître glacier ,“0” sinon	Numérique
BOISSJUS	“1” si le client a acheté un produit de type boisson jus ,“0” sinon	Numérique
Quantity2015	la moyenne de la quantité achetée pour chaque client en 2015	Numérique
Amount2015	le montant d’achats de 2015 pour chaque client	Numérique
Quantity2016	la moyenne de la quantité achetée pour chaque client en 2016	Numérique
Amount2016	le montant d’achats de 2016 pour chaque client	Numérique
Quantity2017	la moyenne de la quantité achetée pour chaque client en 2017	Numérique
Amount2017	le montant d’achats de 2017 pour chaque client	Numérique
Quantity2018	la moyenne de la quantité achetée pour chaque client en 2018	Numérique
Amount2018	le montant d’achats de 2018 pour chaque client	Numérique
Quantity2019	la moyenne de la quantité achetée pour chaque client en 2019	Numérique
Amount2019	le montant d’achats de 2019 pour chaque client	Numérique

Quantity2020	la moyenne de la quantité achetée pour chaque client en 2020	Numérique
Amount2020	le montant d'achats de 2020 pour chaque client	Numérique
tenure	le nombre de mois que le client est resté dans l'entreprise	Numérique

Table 4.1 – Liste des variables

— **Nettoyage de données** : Les variables non pertinentes affectent la précision de notre modèle, pour cela nous les avons supprimé à l'aide de la bibliothèque pandas.
Exemple : Nous avons supprimé la variable "Code postal" car on peut facilement extraire cette information à partir de la variable Région.

— **Traitement des valeurs manquantes :** Les valeurs nulles affectent les performances de notre modèle donc il faut les remplacer par d'autres valeurs. Pour vérifier leur existence, nous avons utilisé la méthode pandas **isnull().sum()** qui retourne le nombre de valeurs manquantes de chaque variable du dataset.

```
ıtrée [6]: # checking null value
           df1.isnull().sum()

Out[6]: Customer                0
        Nom                     0
        Région                  0
        Code postal             0
        Kilometrage             0
        NTéléphone              5
        NTélécopie            189
        E-mail                 86
        Contact               187
        GroupeComptaClient      0
        GroupeComptaMarché      0
        GroupeComptaMarchéTVA   0
        CodeConditionsPaiement  0
        CodeDevise              0
        CodeActivité            3
        NatureActivity          4
        CodeConditionLivraison 46
        Churn                   0
        VAT0                    0
        VAT17                   0
        VAT19                   0
        LineDiscountMoyP        0
        LAITBLANC               0
        LAITAROMA               0
        PDL                     0
         MCUIS.                 0
         BOISSLAIT              0
         MGLACIER               0
         BOISSJUS               0
         Amount2015           138
         Quantity2015         138
         Amount2016           113
         Quantity2016         113
         Amount2017           123
         Quantity2017         123
         Quantity2018          96
         Amount2018            96
         Quantity2019          75
         Amount2019            75
         Quantity2020          82
         Amount2020            82
         tenure                 0
```

Figure 4.3 – Nombre de valeurs manquantes dans notre dataset

Pour gérer ces valeurs manquantes, nous avons remplacé les valeurs numériques par des "0" et les valeurs catégorielles par " ?".

— **Traitement des valeurs catégorielles** : Les valeurs catégorielles ne seront pas acceptées par le modèle de prédiction donc nous les avons repéré en utilisant la méthode dtypes, ensuite on les a converti en valeurs numériques :
  — Nous avons remplacé “yes” par 1 et “no” par 0 dans la variable churn .
  — Nous avons remplacé “Local” par 1 et “Etranger” par 0 dans les variables GroupeComptaClient et GroupeComptaMarché.
  — Nous avons remplacé “ASSUJETTI” par 1 et “EXONERE” par 0 dans la variable GroupeComptaMarchéTVA.
  — Nous avons utilisé **l’encodage one-hot** qui est un processus simple de représentation d’une colonne de catégories sous forme de matrice binaire élargie étiquetée pour les variables NatureActivity et CodeConditionLivraison.
  — Nous avons importé **LabelEncoder** pour encoder les valeurs catégorielles restantes en valeurs numériques.
— **Mise à l’échelle des données** : Nos données ont un ordre de grandeur différent , Cette différence d’échelle peut affecter la performance de notre modèle, Pour palier à ce problème, nous avons utilisé **MinMaxScaler** qui traduit chaque caractéristique individuellement de sorte qu’elle se trouve dans la plage donnée sur l’ensemble d’apprentissage, par exemple entre zéro et un.

4.3 Visualisation

A l’aide de l’outil power BI nous avons pu étudier le comportement des clients et construire puis valider des hypothèses.

— Dans la carte géographique de l’Algérie ci-dessous, les clients sont regroupés en fonction de la région et la variable “Churn”.
  Nous remarquons que le phénomène de la perte des clients ne dépend pas de la région.

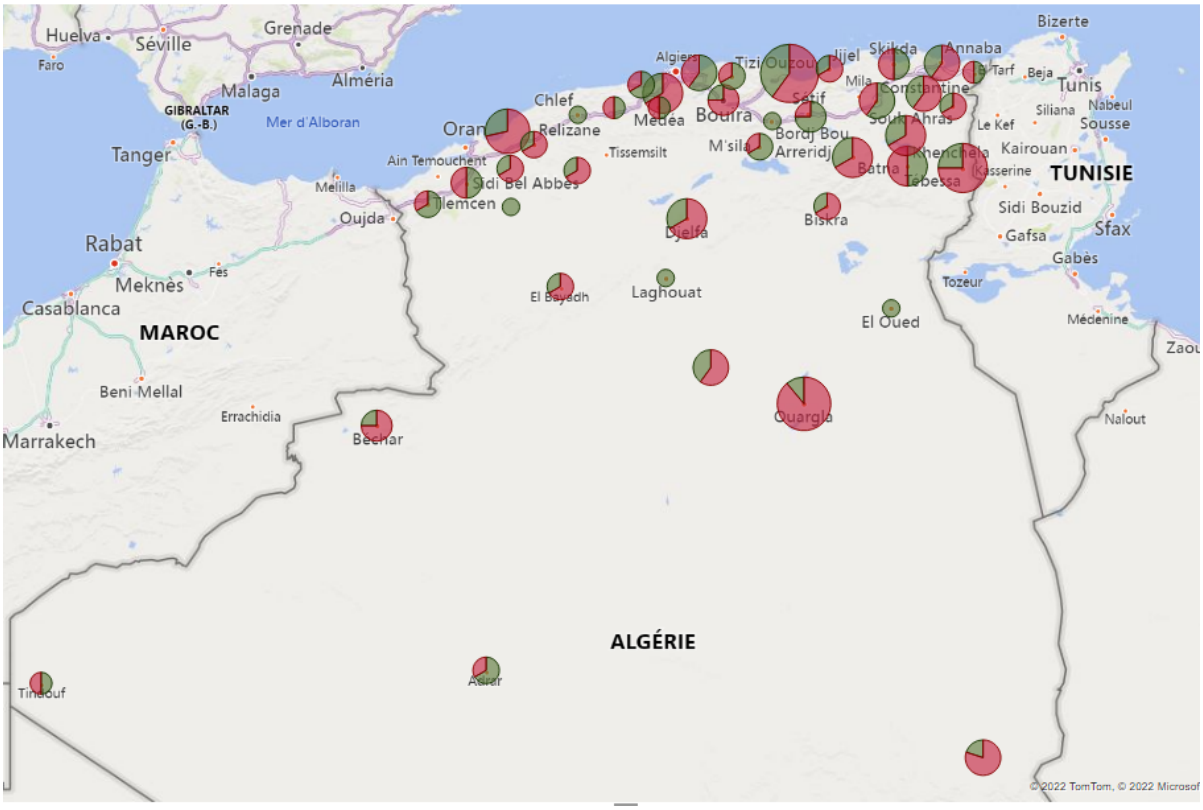


Figure 4.4 – Carte géographique qui regroupe les clients en fonction de la région et la variable Churn.

— Les courbes ci-dessous représentent le comportement de deux clients depuis 2015 à 2020 par rapport aux quantités et montants, nous avons noté que :
  — **Le comportement d'un churner :** Quantity et Amount diminuent jusqu'à ce que le client quitte l'entreprise.
  — **Le comportement d'un non churner :** La courbe de Amount et Quantity est en croissance.
  Nous avons déduit que quand la quantité achetée diminue, le client est susceptible de quitter l'entreprise.

Figure 4.5 – Courbe qui compare le comportement d'un client qui a quitté et un client qui est resté par rapport a Quantity et Amount.

— Nous avons utilisé l'outil influenceur clé de Power BI pour comprendre et voir l'impact des produits achetés et la TVA sur l'attrition des clients.
  — VAT influence le désabonnement des clients, car comme le montre la figure ci-dessus,
    — Lorsque VAT17= 1 la probabilité que Churn= "yes" augmente de 2.21 fois
    — Lorsque VAT19= 0 la probabilité que Churn= "yes" augmente de 2.29 fois.

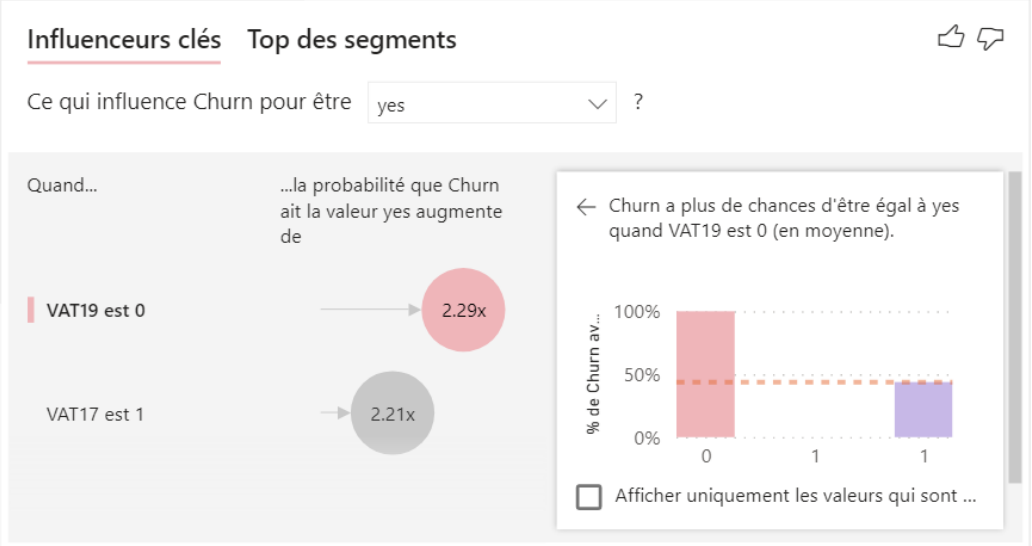


Figure 4.6 – Churn par rapport à la variable VAT

— l'attrition de la clientèle dépend du type de produits achetés car comme le montre la figure ci-dessous :
  — Lorsque le client n'achète pas Maître Cuisinier ("MCUIS.= 0") la probabilité que Churn= "yes" augmente de 2.95 fois.
  — Lorsque le client n'achète pas Maître Glacier ("MGLACIER= 0") la probabilité que Churn= "yes" augmente de 2.22 fois.
  — Lorsque le client n'achète pas le lait aromatisé ("LAITAROMA = 0") la probabilité que Churn= "yes" augmente de 1.59 fois.

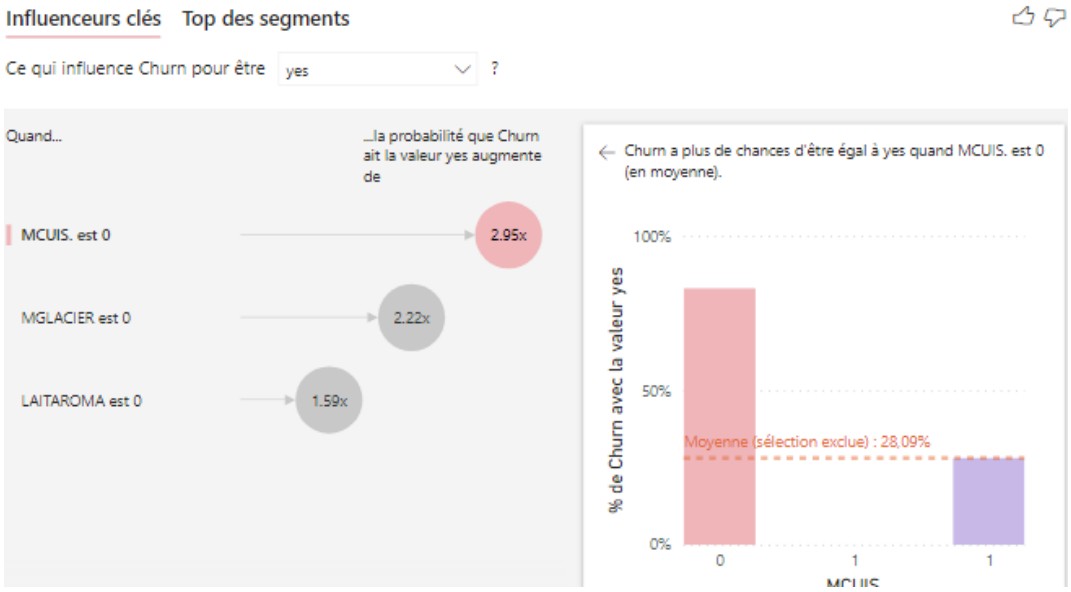


Figure 4.7 – L'attrition par rapport aux produits achetés.

— La figure ci- dessous représente la différence entre le nombre des clients qui ont quitté l'entreprise (churn = yes) et ceux qui sont restés (Churn ="No") par rapport à leur nature d'activité. Nous remarquons que les clients ayant comme nature d'activité :
  — Société de catering et de services hôteliers.
  — Catering.
  — Hébergement et restauration.

— Distribution agroalimentaire.
Ont tous quitté l'entreprise.

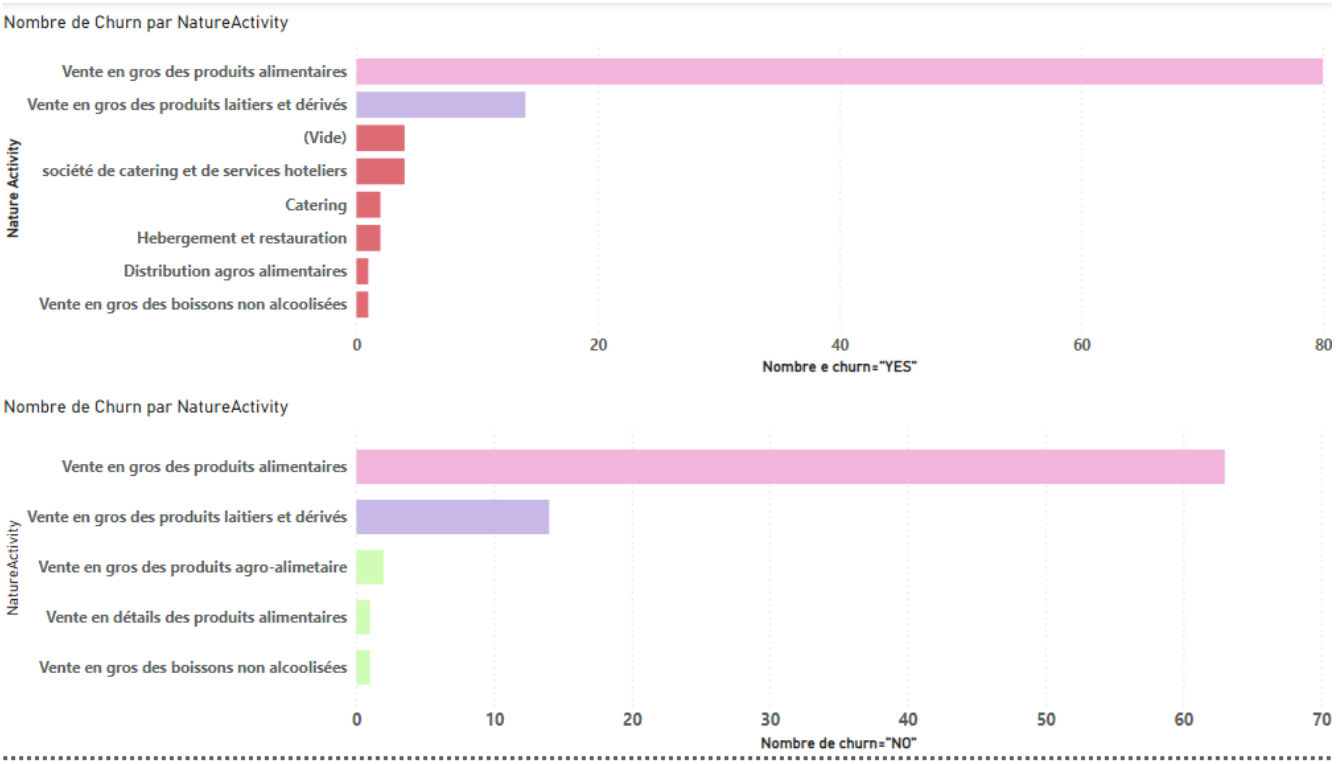


Figure 4.8 – Attrition par rapport à la nature d'activité des clients

— A travers le graphe ci-dessous , nous pouvons tirer comme information que les nouveaux clients sont plus susceptibles de quitter l'entreprise.

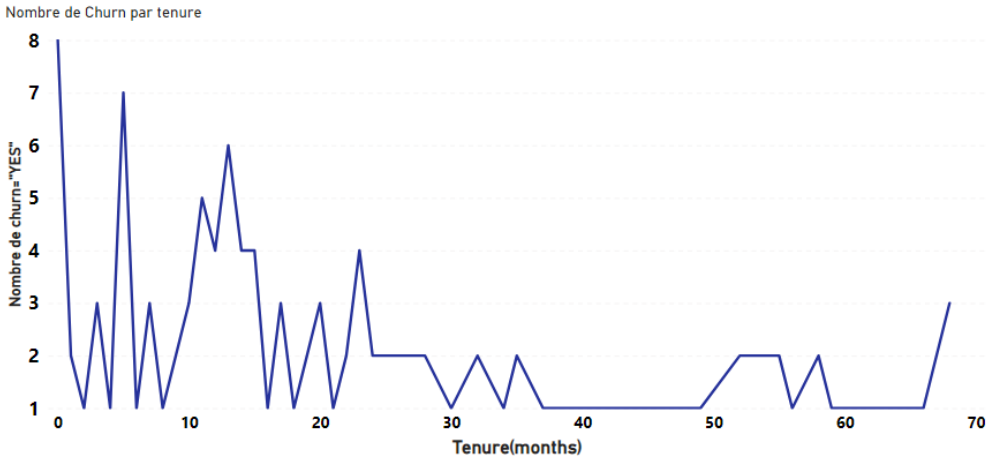


Figure 4.9 – Attrition par rapport à la variable tenure

— A travers le graphe ci-dessous nous remarquons que le nombre le plus élevé de clients qui ont quitté l'entreprise est atteint lorsque la moyenne de la remise est égale à zéro.

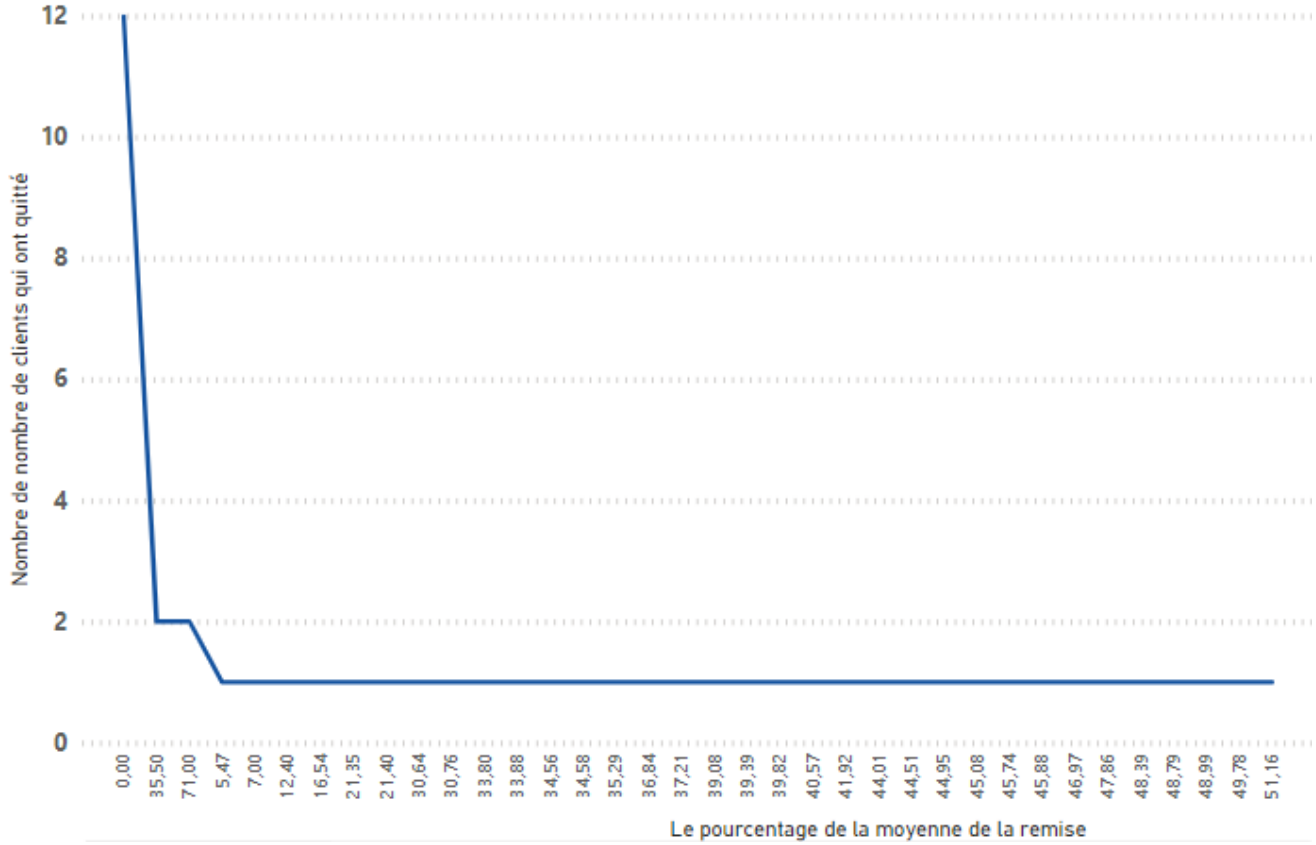


Figure 4.10 – Attrition par rapport à la moyenne de la remise

— Notre ensemble de données contient 189 lignes où chaque ligne représente un client et 40 variables. Le tableau ci-dessous explique le rôle de chaque variable.

4.4 Fractionnement de l'ensemble de données

— **Variables indépendantes et dépendantes :** L'ensemble de données est séparé en valeurs x et y. y serait la colonne "Churn" tandis que x serait la liste restante des variables indépendantes dans l'ensemble de données.
Les variables dépendantes c'est à dire celles qu'on va utiliser pour la prédiction de l'attrition sont : "NatureActivity" ,"CodeConditionLivraison", "VAT17", "VAT19", "LAITAROMA", "PDL","MCUIS" , "BOISSLAIT", "MGLACIER","BOISSJUS", "tenure", "Kilometrage", "Amount2015", "Quantity2015", "Amount2016", "Quantity2016", "Amount2017", "Quantity2017", "Quantity2018", "Amount2018", "Quantity2019", "Amount2019", "Quantity2020", "Amount2020", "LineDiscountMoyP".
— **Données d'apprentissage et de test :** l'ensemble de données principal est divisé avec les proportions suivantes : 80 % pour les données d'apprentissage et 20 % pour les données de test.

4.5 Les algorithmes à utiliser

Le choix d'un algorithme dépend fortement de la tâche à résoudre, notre cas est un problème de classification binaire, pour cela nous avons utilisé ces algorithmes :
— Les réseaux de neurones.
— Machine à vecteurs de support (SVM).
— Boosting de gradient XGBoost.

4.6 Conclusion

Dans ce chapitre, nous avons commencé par la préparation et la compréhension de notre jeu de données, ensuite nous avons procédé à la manipulation et la visualisation de nos variables. Enfin nous avons séparé nos variables en colonnes dépendantes et indépendantes pour passer à la création de notre réseau de neurones.

CHAPITRE 5

IMPLÉMENTATION ET ÉVALUATION

5.1 Introduction

Cette partie est consacrée à l'implémentation et l'évaluation de notre projet. Dans ce chapitre nous allons voir les algorithmes d'apprentissage automatique utilisés ainsi que les tests réalisés, pour ensuite finir avec les résultats obtenus par nos modèles et une comparaison entre ces résultats.

5.2 Les algorithmes d'apprentissage utilisés

5.2.1 Modèle réseau de neurones

5.2.1.1 Définition du modèle

- Nous avons dévisé l'ensemble de données principales avec les proportions suivantes : 80 % pour les données d'apprentissage et 20 % pour les données de test.
- Nous avons utilisé le modèle séquentiel car il est plus intuitif pour les architectures simples.
- Nous avons défini quatre couches :
  - Une couche d'entrée qui contient les 35 variables d'entrée.
  - Deux couches intermédiaires qui contiennent respectivement 18 et 3 neurones cachés.
  - Une couche de sortie qui contient le résultat de la prédiction (0 ou 1).
- Nous avons utilisé deux fonctions d'activation :
  - **ReLU** : Car elle permet un entraînement plus rapide comparé aux autres fonctions.
  - **Sigmoïde** : Car notre sortie est égale à 0 ou 1.

5.2.1.2 Compilation du modèle

La compilation est la dernière étape de la création d'un modèle neuronal artificiel. Elle définit la fonction de perte, l'optimiseur et les métriques que nous devons passer en paramètres.

- **la fonction de perte** : Nous avons utilisé **l'entropie croisée** car elle est la plus adaptée pour les problèmes de classification.
- **l'optimiseur** : Nous avons utilisé **Adam** qui est utilisé pour calculer les taux d'apprentissage adaptatifs pour chaque paramètre.
- **Les métriques** : Nous avons définit **accuracy (la précision)** qui est le rapport entre le nombre de prédictions correctes et le nombre total de prédictions.
- Afin de réduire le risque de **surajustement**, Nous avons utilisé **l'arrêt précoce** qui est une forme de régularisation simple et efficace.

5.2.1.3 Résultats obtenus

— La figure suivante montre le résultat de la fonction de précision “accuracy” pour les données d’entrainement et de test, nous remarquons qu’elle est en croissance ce qui signifie que notre modèle est performant.

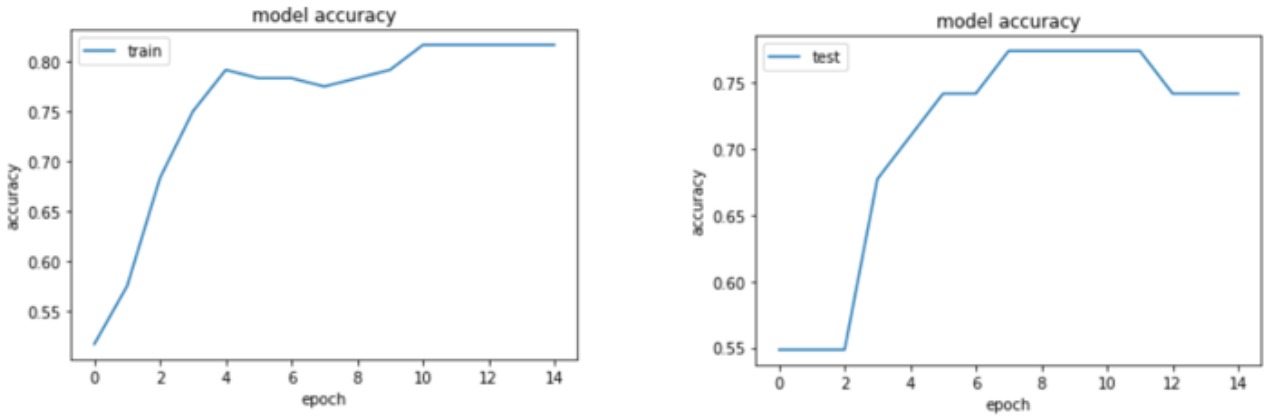


Figure 5.1 – Graphe de la fonction accuracy pour les données d’entrainemet et de test.

— Les figures suivante montre le résultat de la fonction de perte pour les données d’entrainement et de test, sachant que si les prévisions s’écartent trop des résultats réels, la fonction de perte produira un très grand nombre, s’ils sont assez bons, elle produira un nombre inférieur, et c’est ce qui s’est produit dans notre cas :

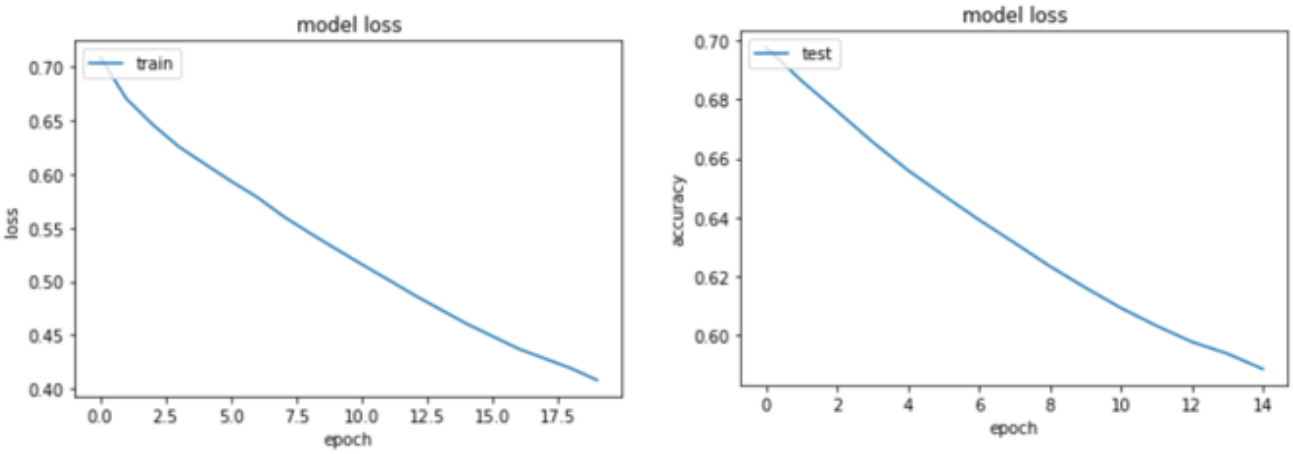


Figure 5.2 – Graphe de la fonction Loss pour les données d’entrainement et de test.

— Afin de mesurer la performance d’un modèle d’apprentissage automatique on utilise généralement la matrice de confusion qui est basée sur le nombre d’enregistrements de test correctement et incorrectement prédit par le modèle, la figure suivante montre le résultat de la matrice de confusion de notre modèle :
On a trouvé quatre catégories de résultat :
— **Vrai positif** : élément de la classe « 1 » correctement prédit(16).
— **Vrai négatif** : élément de la classe « 0 » correctement prédit.(16).
— **Faux positif** : élément de la classe « 1 » mal prédit(3).
— **Faux négatif** : élément de la classe « 0 » mal prédit(3).
Alors on trouve que l’erreur est entre (3,3).

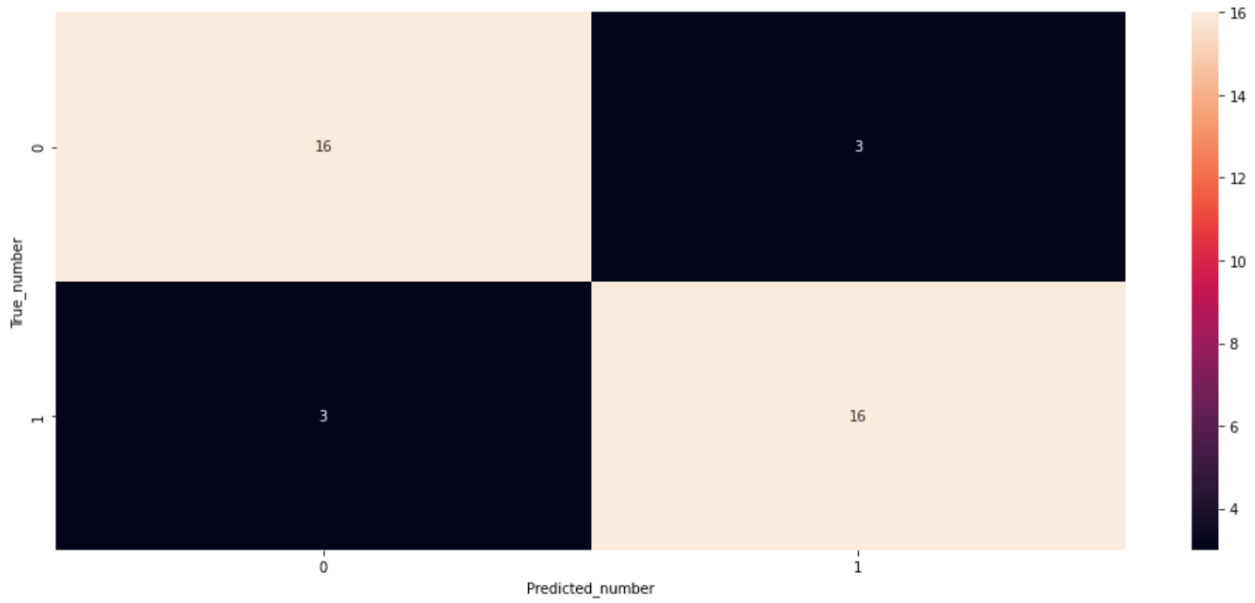


Figure 5.3 – Matrice de confusion de notre modèle de réseau de neurones.

— Pour juger la qualité de nos prédictions, nous avons calculé les métriques de notre matrice de confusion comme suit :

$$précision(accuracy) = \frac{VP + VN}{VP + VN + FP + FN}.$$

$$sensibilité(sensitivity) = \frac{VP}{VP + FN}.$$

$$spécificité(specificity) = \frac{VN}{VN + FP}.$$

Les résultats sont dans le tableau suivant :

Les Métriques	précision	sensibilité	spécificité
Réseau de neurones	0.84	0.84	0.84

Table 5.1 – Résultats de l'évaluation du modèle réseau de neurones

— Selon ce modèle, nous pouvons prédire les churners 84% du temps, et c'est un bon résultat pour prévenir l'attrition des clients.

5.2.2 Modèle SVM

— La mise en œuvre est simple et directe, nous avons utilisé le package svm de Scikit Learn, initialiser la machine à vecteurs de support, puis donner "Linear" comme paramètre pour notre noyau.car la formation d'un SVM avec un noyau linéaire est plus rapide qu'avec n'importe quel autre noyau.

5.2.2.1 Résultats obtenus

— La figure ci-dessous représente la matrice de confusion obtenue pour SVM, on a trouvé quatre catégories de résultat :
  — **Vrai positif** : élément de la classe « 1 » correctement prédit (15).
  — **Vrai négatif** : élément de la classe « 0 » correctement prédit.(16).

— **Faux positif** : élément de la classe « 1 » mal prédit.(4).
— **Faux négatif** : élément de la classe « 0 » mal prédit (3).
Alors on trouve que l'erreur est entre (3,4).

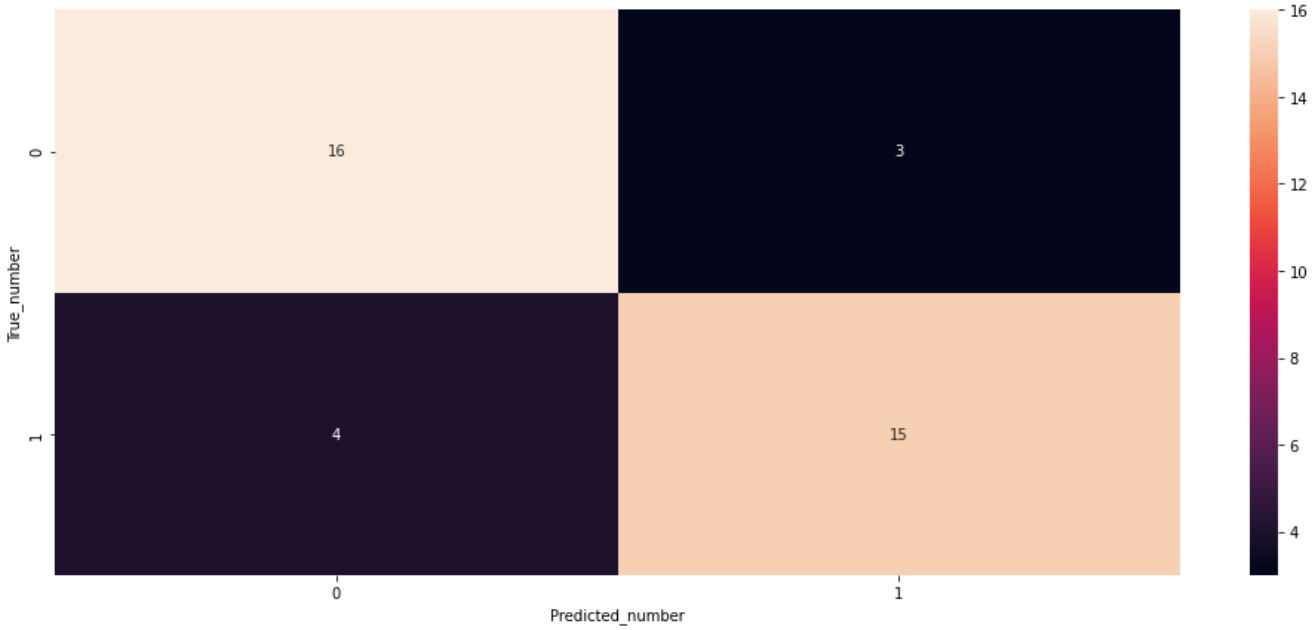


Figure 5.4 – Matrice de confusion de notre modèle SVM.

— Nous avons calculé les métriques de notre table de confusion comme nous avons fait avec le modèle réseau de neurones. Les résultats sont dans le tableau suivant :

Les Métriques	précision	sensibilité	spécificité
SVM	0.81	0.83	0.80

Table 5.2 – Résultats de l'évaluation du modèle SVM

— Selon ce modèle, nous pouvons prédire les churners 81% du temps, et d'après ces métriques le modèle réseau de neurones donne de meilleures résutats que SVM.

5.2.3 Modèle XGBoost

— Pour la mise en œuvre nous avons créé XGBClassifier puis nous l'avons adapté à notre ensemble de données d'entraînement.

5.2.3.1 Résultats obtenus

— La figure ci-dessous représente la matrice de confusion obtenue pour Xgboost, On a trouvé quatre catégorie de résultat :
— **Vrai positif** : élément de la classe « 1 » correctement prédit(18).
— **Vrai négatif** : élément de la classe « 0 » correctement prédit(17).
— **Faux positif** :élément de la classe « 1 » mal prédit(1).
— **Faux négatif** : élément de la classe « 0 » mal prédit(2).
Alors on trouve que l'erreur est entre (1,2).

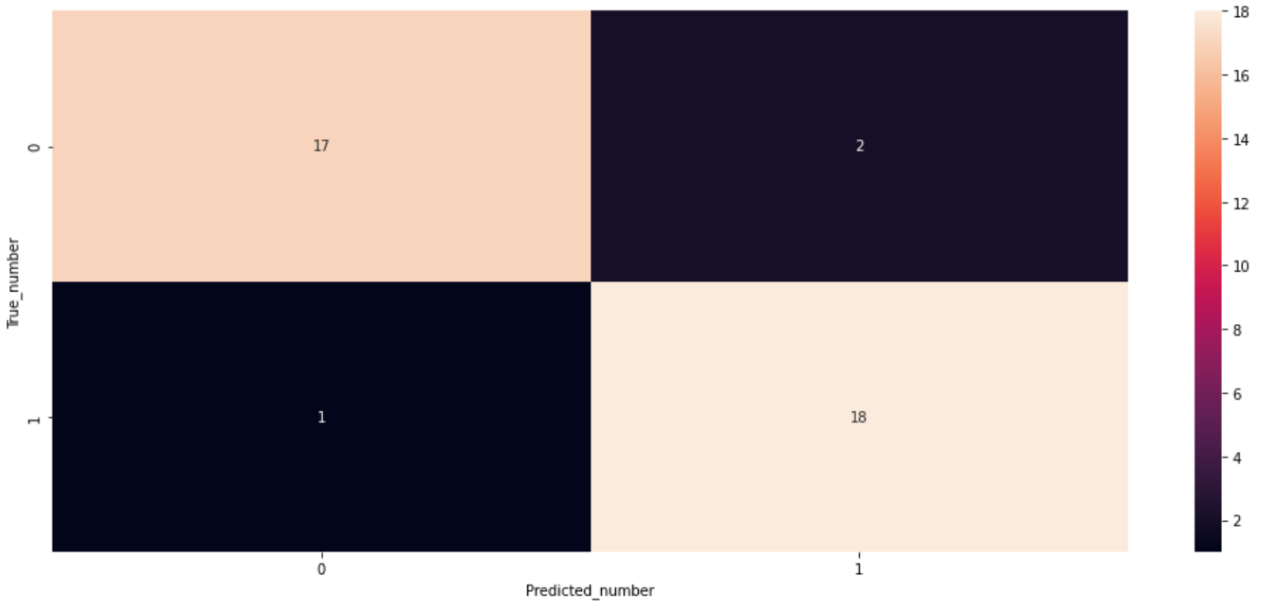


Figure 5.5 – Matrice de confusion de notre modèle xgboost

Les résultats sont dans le tableau suivant :

Les Métriques	précision	sensibilité	spécificité
XGBoost	0.92	0.9	0.94

Table 5.3 – Résultats de l'évaluation du modèle XGBoost

— Selon ce modèle, nous pouvons prédire les churners 90% du temps donc le pourcentage d'erreur est extrement réduis, et c'est le meilleur résultat que nous avons obtenu des trois modèles.

— La figure ci-dessous montre le résultat de la comparaison entre les valeurs originales et prédites pour chaque clients :
Si original_churn = predicted_churn alors :
notre modèle a pu prédire un résultat correct pour ce client, sinon il n'a pas réussi à le prédire. Sur 30 clients, il n'a pas réussi à prédire 3.

Out[35]:

	orignal_churn	predicted_churn
144	1	1
10	0	0
5	1	0
59	1	1
103	0	0
19	1	1
83	1	1
47	1	1
2	1	1
102	0	0
152	1	1
35	1	1
117	0	0
138	0	0
76	0	1
56	1	1
20	1	1
110	0	0
72	0	0
183	1	1
39	0	1
52	1	1
69	1	1
146	1	1
130	0	0
1	1	1
105	0	0
70	1	1
149	1	1
142	0	0

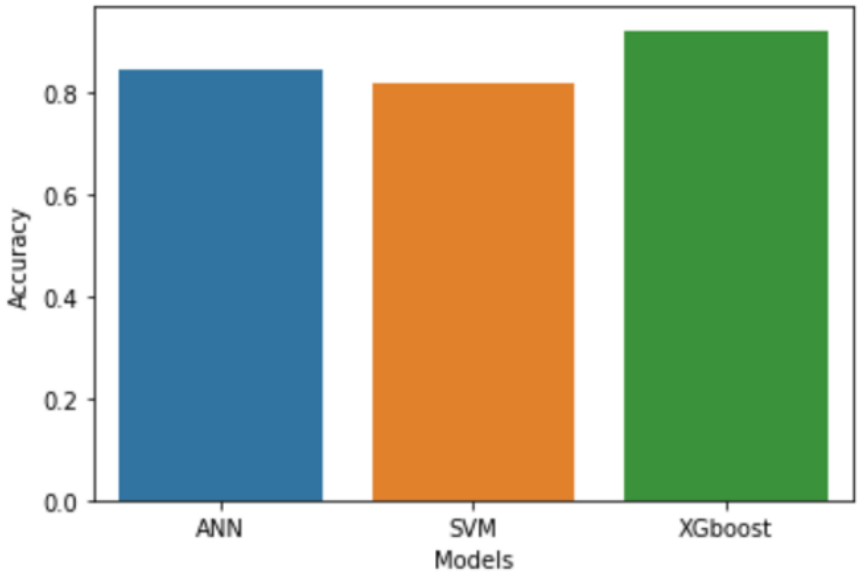
Figure 5.6 – La table de prédiction

5.2.4 Comparaison des résultats des algorithmes

Dans cette partie nous allons voir les résultats obtenus de la comparaison de nos modèles, la figure ci-dessous illustre ces résultats :

— **XGboost** :Il nous a fourni de meilleurs résultats avec une précision= 0,92.
— **ANN** : le modèle réseau de neurones vient en deuxième position avec une précision= 0,84.
— **SVM** :le Modèle machine à vecteur de support vient en dernière position avec une précision= 0,82.

Donc on va utiliser XGboost pour prédire la perte de clients.



	Models	Accuracy
0	ANN	0.842105
1	SVM	0.815789
2	XGboost	0.921053

Figure 5.7 – Résultats de la comparaison entre nos modèles

5.3 Interface de prédiction

Afin que nous puissions mieux visualiser le résultat de notre prédiction, nous avons conçu une application pour tester les résultats de notre étude. Elle est constituée d'une interface qui contient 10 boîtes de sélection, un widgets de curseur, 14 champs de saisie et un bouton de prédiction.
L'utilisateur introduit les informations du client et clique sur le bouton.

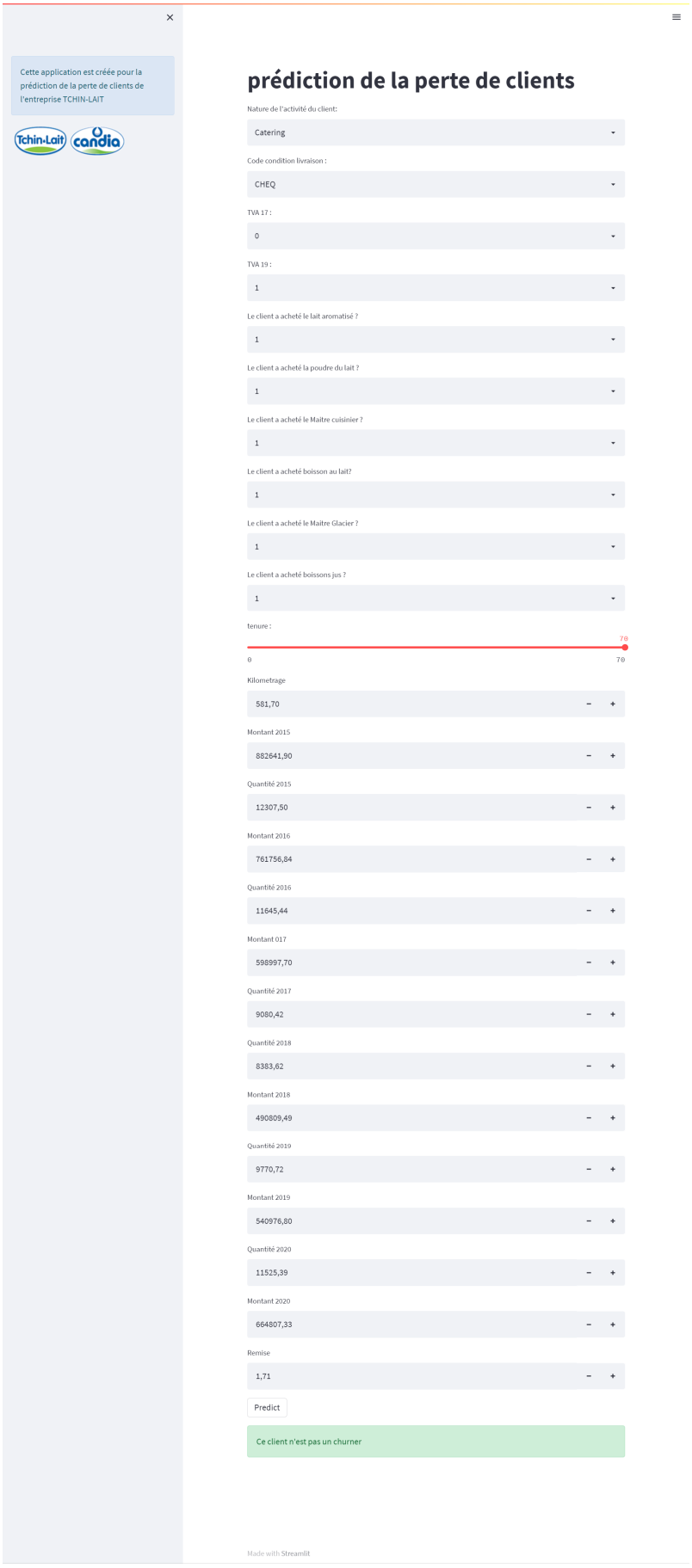


Figure 5.8 – L'interface de notre application

Le résultat est affiché comme suit :

— Si le client est susceptible de quitter l'entreprise le message affiché est : " Ce client est un churner".

— Sinon le message affiché est : "Ce client n'est pas un churner".

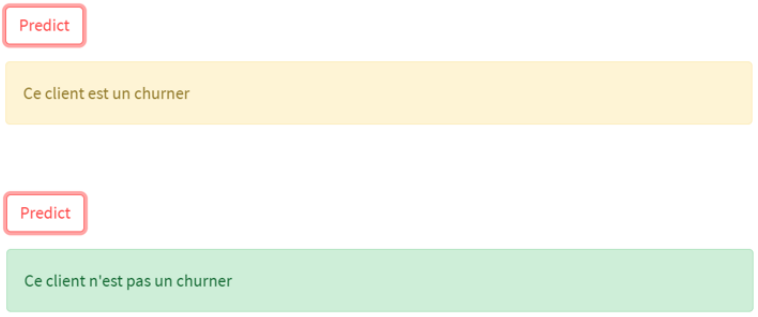


Figure 5.9 – Résultat de prédiction

5.4 Conclusion

Dans ce dernier chapitre, nous avons construit nos trois modèles de prédiction qui sont le modèle réseau de neurones, SVM et XGBoost. Puis, discuter les résultats obtenus et finalement montrer ces résultat à travers notre interface de prédiction.

CONCLUSION GÉNÉRALE

Ce projet de fin d'étude a comme but de prédire l'attrition de clients dans l'entreprise Tchinlait Candia. Pour pouvoir réaliser ce travail, nous avons intégré l'entreprise et collecté toutes les informations utiles concernant ses clients. Ensuite nous avons effectué plusieurs étapes de traitement sur nos données pour construire et valider des hypothèses afin d'arriver à l'étape de création de notre modèle de prédiction basé sur des algorithmes de machine learning et de deep learning.

Travailler sur ce projet nous a permis de découvrir un nouveau domaine passionnant hors de ce que nous avons étudié durant notre cursus en master, effectivement la science de données est basée sur des technologies très puissantes qui ont révolutionné plusieurs secteurs depuis les 10 dernières années et nous en sommes ravies car nous avons maintenant une idée précise de ce que nous voulons faire à l'avenir dans nos carrières professionnelles.

Nous sommes certains que grâce à notre étude et à ce modèle de prédiction, l'entreprise candia pourra comprendre et réduire l'attrition de ses clients, surtout pour les années à venir quand la concurrence fera part du marché.

Nous pensons aussi que ce manuscrit sera bénéfique et pourra servir de références pour d'autres personnes souhaitant s'engager et travailler dans un projet similaire.

Perspectives :
Notre Projet pourrait être amélioré selon les perspectives suivantes :

- Estimer le temps d'apparition d'une résiliation potentielle pour chaque client afin de prendre des mesures de fidélisation en temps voulu, en utilisant les séries temporelles.
- Trouver d'autres facteurs qui peuvent affecter l'attrition de la clientèle afin d'améliorer notre modèle de prédiction.
- Améliorer notre application de prédiction en ajoutant une partie visualisation, l'utilisateur pourra ainsi visualiser les données sans utiliser un autre outil.

BIBLIOGRAPHIE

[1] https://harris-interactive.fr/solutions/.Consulté le 23/05/2022.

[2] https://machinelearnia.com/machine-learning-introduction/Consulté le 07/03/2022.

[3] https://www.geeksforgeeks.org/introduction-deep-learning/?ref=gcseConsulté le 22/03/2022.

[4] https://inside-machinelearning.com/fonction-dactivation-comment-ca-marche-une-explication-simple/Consulté le 13/03/2022.

[5] https://medium.com/hackernoon/gradient-descent-aynk-7cbe95a778da. Consulté le 22/03/2022.

[6] https://www.analyticsvidhya.com/blog/2021/06/complete-guide-to-prevent-overfitting-in-neural-networks-part-2/Consulté le 22/03/2022.

[7] https://stanford.edu/~shervine/l/fr/teaching/cs-229/pense-bete-apprentissage-profondConsulté le 22/03/2022.

[8] https://www.python.org/Consulté le 11/05/2022.

[9] https://www.anaconda.com/Consulté le 11/05/2022.

[10] https://jupyter.org/Consulté le 11/05/2022.

[11] https://www.spyder-ide.org/.Consulté le 13/05/2022.

[12] https://docs.microsoft.com/en-us/power-bi/fundamentals/power-bi-overviewConsulté le 15/05/2022.

[13] https://corporatefinanceinstitute.com/resources/excel/study/excel-definition-overview/.Consulté le 13/05/2022.

[14] https://numpy.org/.

[15] https://pandas.pydata.org/.Consulté le 15/05/2022.

[16] https://seaborn.pydata.org/.Consulté le 15/05/2022.

[17] https://matplotlib.org/Consulté le 20/05/2022.

[18] https://www.tensorflow.org/learn.Consulté le 20/05/2022.

[19] https://keras.io/about/.Consulté le 20/05/2022.

[20] https://scikit-learn.org/.Consulté le 01/06/2022.

[21] https://xgboost.readthedocs.io/en/stable/.Consulté le 13/06/2022.

[22] https://streamlit.io/.Consulté le 23/05/2022.

[23] Assef Jafar Abdelrahim Kasem Ahmad and Kadan Aljoumaa. Customer churn prediction in telecom using machine learning in big data platformr. (2) :24.

[24] Lanseur Akila and Ait sidhoum Houria. Les déterminants du churn client dans le secteur des télécommunications : étude des trois opérateurs de la téléphonie mobile en algérie. (677) :690.

[25] Chris Albon. python machine learning cookbook. O'Reilly Media, United states of America, 2018.

[26] Shai Shalev-Shwartz andShai Ben-David. understanding machine learning from theory to algorithms. Cambridge University Press, 32 Avenue of the Americas, New York, NY 10013-2473, USA, 2014.

[27] Blaine Bateman, Ashish Ranjan Jha, Benjami, Johnston, and Ishita Mathur. The Supervised Learning Workshop. Packt, Birmingham B3 2PB, UK, 2020.

[28] Michael W. Berry and Azlinah Mohamed. Unsupervised and Semi-Supervised Learning for data science. Springer, M. Emre Celebi, Computer Science Department, Conway, Arkansas, USA, 2014.

[29] Charu C.aggarwal. Neural networks and deep learning. springer, New York, 2018.

[30] Eugene Charniak. Introduction to deep learning. The MIT press, London, England, 2018.

[31] François Chollet. Deep learning with python. Manning, United States of America, 2018.

[32] Roya Hejazinia and Mahdi Kazemi. Prioritizing factors influencing customer churn. (229) :236.

[33] Yoshua Bengio lan Goodfellow and Aaron Courville. DEEP LEARNING. The MIT press, United states of America, 2016.

[34] John Paul Mueller and Luca Massaron. Deep learning for dummies. John Wiley Sons, New jersey, 2019.

[35] N.Kamalraj and A.Malathi. A survey on churn prediction techniques in communication sector. (1) :42.

[36] Sebastian Raschka and vahid mirjalili. python machine learning. Packt, Mumbai, 2017.

[37] Shuhua Xu Xiaohang Zhang, Ji Zhu and Yan Wan . Predicting customer churn through interpersonal influence. (99) :104.