

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Université M'hamed Bougara de Boumerdès



Faculté des Sciences
Département des Mathématiques

*Optimisation d'un modèle de classification des churner
DNN à l'aide de l'Algorithme Génétique.*

Mémoire présenté,
Par

M^r. Mohamed Aimen Bouzouad & M^r. Mohamed Sayhi

Pour l'obtention du diplôme de Master en Mathématiques Financières
Soutenu publiquement le mois de novembre 2020 , devant le jury
composé de :

Présidente	M ^{me}	Larabi Ghenima	M.A.A	U.M.B.B.
Examinatrice	M ^{me}	Drici Wassila	M.C.B	U.M.B.B.
Promotrice	M ^{me}	Gatt Fella	M.A.A	U.M.B.B.

Année Universitaire 2019 – 2020

Table des matières

Table des figures	4
Liste des tableaux	6
1 Data Mining	12
1.1 Définition	13
1.2 Principe et spécificité	14
1.2.1 Stratégie du Data Mining	14
1.2.2 Méthodologie	14
1.3 Taxinomie des méthodes	15
1.3.1 Selon les objectifs	15
1.3.2 Méthodes d'apprentissage	16
1.4 Data mining et statistiques	17
1.5 Méthodes du Data Mining à l'étude du churn	17
1.5.1 Étude sur le churn des clients utilisant la méthode d'analyse de survie	18
1.5.2 Étude sur le churn utilisant la méthode d'arbre de dé- cision	18
1.5.3 Étude de churn utilisant la régression logistique	19
1.5.4 Étude sur le churn utilisant la méthode de Support Vector Machine (SVM)	20
1.5.5 Étude de Churn utilisant les réseaux de neurones	21

TABLE DES MATIÈRES

2	Réseaux de neurones	23
2.1	Neurone formel	24
2.2	Modélisation d'un neurone formel	26
2.2.1	Fonction d'activation	27
2.3	Défférents modèles de réseaux de neurones	28
2.3.1	Modèle de Hopfield	28
2.3.2	Modèle de Kohonen	28
2.3.3	Le modèle de Perceptron	29
2.3.4	Le modèle Adaline	30
2.4	Types de réseaux de neurones	30
2.4.1	Réseaux de neurones non bouclés	30
2.4.2	Les réseaux de neurones bouclés (récurrents)	34
2.5	Apprentissage des Réseaux de neurones artificiels	34
2.6	Étapes de la conception d'un réseau de neurone :	35
3	Algorithme Génétique	36
3.1	Les heuristiques	37
3.2	Les métaheuristiques	37
3.2.1	Classification des méthodes approchée	38
3.2.2	Applications	38
3.3	Historique des algorithmes génétiques	39
3.4	Définitions	39
3.5	Fonctionnement	40
3.6	Les opérateurs	40
3.6.1	L'opérateur de sélection	40
3.6.2	L'opérateur de croisement	41
3.6.3	L'opérateur de mutation	42

TABLE DES MATIÈRES

4	Application	44
4.1	Langage R	45
4.1.1	Pourquoi utiliser R	46
4.2	Définition de PACKAGE	46
4.3	Description des données	48
4.4	Validation croisée (Cross validation)	51
4.5	Classification déséquilibrée (Imbalanced Classification)	54
4.5.1	Les méthodes pour traiter les ensembles de données déséquilibrés	55
4.6	Implémentation du Réseau de Neurone	58
4.7	Implémentation Algorithme Génétique	59
4.8	Interprétation des résultats	59
	Bibliographie	64

Table des figures

1.1	Les différentes Sciences et Techniques utilisées dans le DATA MINING	13
1.2	Architecture du Data Mining	14
2.1	Analogie entre le neurone biologique et artificiel	24
2.2	Modèle d'un neurone artificiel	25
2.3	Modèle de Kohonen	29
2.4	Modèle de Perceptron	30
2.5	Un réseau de neurones mono-couche	31
2.6	Un perceptron multi-couches contenant trois couches	32
2.7	Un réseau de neurones à connexion complète	33
2.8	Un réseau de neurones à connexions locales	33
2.9	Un réseau de neurone bouclé	34
3.1	Exemple de croisement	41
3.2	Exemple de mutation	42
4.1	Une partie du data	49
4.2	Une partie du data	49
4.3	Over-fitting et Under-fitting	52
4.4	TEST et TRAIN data	53
4.5	K fold cross validation	54

TABLE DES FIGURES

4.6	Les deux classes churner et non churner	55
4.7	Les deux classes churner et non churner dans TRAIN DATA	57
4.8	Les deux classes churner et non churner de TRAIN DATA après l'utilisation de la fonction ovun sample	58
4.9	L'entraînement du modèle MLP	60

Liste des tableaux

2.1	Analogie entre entre neurone biologique et neurone artificiel	24
4.1	Description des données	50
4.2	Codification de la variable CHURN	50
4.3	Codification de la variable STATE	51
4.4	les résultats de l' AG Pour le paramètre Units	61
4.5	les résultats de l' AG Pour le paramètre Drop out	61

Remerciements

Nous tenons tout d'abord à remercier ALLAH le tout puissant et miséricordieux de nous avoir donné le courage et la patience d'accomplir ce modeste travail.

Nous remercions tout particulièrement notre promotrice madame F. Gatt pour avoir acceptée de nous encadrer, et la confiance qu'elle nous a accordé, sa disponibilité, ses précieux conseils et le soutien qu'elle nous a accordé.

Nous remercions monsieur H.Bounoua, pour l'intérêt qu'il a porté à ce travail en acceptant de co-diriger cette étude, pour sa disponibilité, ses orientations et ses remarques fructueuses.

Nos remerciements s'adressent également à l'ensemble des enseignants du Département des Mathématiques et spécialement spécialité Mathématiques Financières.

Un grand merci à nos familles surtout nos parents, qui nous ont aidé à suivre nos études dans les meilleures conditions et qui nous ont toujours soutenues et encouragé sans limite.

Nous tenons également à remercier les membres du jury : Mme Larabi Ghenima et Mme Drici Wassila d'avoir accordé de leurs temps pour expertiser notre travail, nous espérons qu'ils en soient satisfaits.

Enfin, nous remercierons tous ceux qui ont participé de loin ou de près à la réalisation de ce travail.

Dédicaces

J'adresse tout d'abord mes dédicaces les plus sincères, à mes défunts grand-parents.

À mes chers parents, pour tous leurs sacrifices, leur amour, leur tendresse, leur soutien et leurs prières tout au long de mes études.

À mon frère Akram et mes amis Bahae et Yasser pour leurs appuie et leurs encouragement, merci.

À ma fiancée Sara pour ses encouragements et son soutien moral.

À ma grand-mère pour m'avoir toujours encouragé dans mes études.

À toute ma famille pour leur soutien tout au long de mon parcours universitaire.

À mon ami et mon binôme Mohamed, qui était toujours avec moi je te remercie infiniment, je te souhaite tout la réussite.

À tous mes amis avec qui j'ai passé d'agréables moments.

À mes professeurs et promoteurs compétents Mme. Gatt et Mr. Bounoua, qui étaient vraiment patients avec nous et qui nous ont beaucoup appris dans notre parcours universitaire.

À mes professeurs compétents Mme. Mouhoub et Mr. Bezoui

À toute ma promotion et mes enseignants.

À tous les gens que j'aime et dont je n'ai pas cité les noms.

Merci à tous.

MOHAMED AIMEN

Dédicaces

J'adresse tout d'abord mes dédicaces les plus sincères, à mes défunts grand-parents.

À mes chers parents, pour tous leurs sacrifices, leur amour, leur tendresse, leur soutien et leurs prières tout au long de mes études.

À mon frère Abdelazize et mon ami Rachid Mammeri pour leurs appuie et leurs encouragement, merci.

À toute ma famille pour leur soutien tout au long de mon parcours universitaire.

À mon ami et mon binôme Mohamed Aimen, qui était toujours avec moi je te remercie infiniment, je te souhaite tout la réussite du monde.

À tous mes amis avec qui j'ai passé d'agréables moments.

À mes professeurs et promoteurs compétents Mme. Gatt et Mr. Bounoua, qui étaient vraiment patients avec nous et qui nous ont beaucoup appris dans notre parcours universitaire.

À toute ma promotion et mes enseignants.

À tous les gens que j'aime et dont je n'ai pas cité les noms.

Merci à tous.

MOHAMED

INTRODUCTION GÉNÉRALE

Les compagnies de téléphone, les fournisseurs de services internet, les sociétés de télévision à péage, les compagnies d'assurance et les services de surveillance des alarmes utilisent souvent l'analyse et le taux d'attrition de la clientèle comme l'un de leurs principaux paramètres commerciaux, car le coût de la fidélisation d'un client existant est bien inférieur à celui de l'acquisition d'un nouveau client. Les entreprises de ces secteurs ont souvent des branches de service à la clientèle qui tentent de reconquérir les clients défectueux, car les clients de longue date récupérés peuvent avoir une valeur beaucoup plus importante pour une entreprise que les clients nouvellement recrutés.

Les entreprises font généralement une distinction entre les départs volontaires et les départs involontaires. La résiliation volontaire est due à la décision du client de changer d'entreprise ou de prestataire de services, tandis que la résiliation involontaire est due à des circonstances telles que le transfert du client dans un établissement de soins de longue durée, son décès ou son éloignement. Dans la plupart des applications, les raisons involontaires de désabonnement sont exclues des modèles analytiques. Les analystes ont tendance à se concentrer sur le **churn** volontaire, car il se produit généralement en raison de facteurs de la relation entreprise-client que les entreprises contrôlent, comme la façon dont les interactions de facturation sont traitées ou la façon dont l'aide après-vente est fournie.

Dans le cadre de leurs politiques de fidélisation clients, les entreprises sont amenées à prendre des décisions ciblées en terme de **marketing** et de **pricing**, et avoir une certaine précision concernant les clients ciblés par leurs campagnes.

Pour cela, le recours à des modèles de prévision et de profiling (classification) s'avère nécessaire. Cette phase de tri des clients ciblés revient à construire et de paramétrer des modèles de prévision de sorte à affiner le ciblage des clients potentiellement intéressés par les offres proposées.

Dans le contexte économique actuel et la concurrence rude, à laquelle fait face le secteur de la télécommunication, les entreprises ont entamé une stratégie de fidélisation de leurs clientèle qui est la plus rentable économiquement parlant, et cela consiste à distinguer les clients fidèles des clients qui, potentiellement peuvent changer d'opérateur et sur quels facteurs s'appuyer afin de les fidéliser. Ce problème est connu sous l'appellation du "**Churn**".

Donc, pour cela nous proposons **un Réseau de Neurone** en implimant l'**Algorithme génétique** pour la prédiction des clients cherner .

Ce mémoire est structuré en quatre chapitre :

- Le premier chapitre est introductif, dans lequel nous rappelons quelques notion de base du **Data Mining** ,et on définit les différents méthodes utilisées pour l'étude sur le **churn** des clients.
- Dans le deuxième chapitre, on introduit les définitions et les différents types de réseau de neurone .
- Le troisième chapitre portera sur les concepts théoriques de l'algorithme génétiques.
- Le quatrième et dernier chapitre fera objet de la présentation des données, et l'application et les résultats acquis.

1

Data Mining

Le développement des outils informatiques et de calcul induit à reformuler des possibilités pour le stockage, le traitement et à l'analyse d'ensembles de données très volumineux. Cette évolution, ainsi que la popularité de nouvelles techniques algorithmiques et outils graphiques, conduisent au développement et à la commercialisation des solutions logicielles intégrant un ensemble de méthodes à base d'heuristiques, d'apprentissages et de statistiques connues sous la terminologie de Data Mining [17].

1.1 Définition

Le Data Mining traduit généralement en fouille de données est un ensemble de techniques d'exploration de données permettant d'extraire d'une base de données des connaissances sous la forme de modèles de description afin de décrire le comportement actuel et/ou de prédire le comportement futur du système représenté par ces données[1].

Le Data Mining se situe à la croisée des bases de données, l'intelligence Artificielle, statistique et l'analyse de données ... : figure[1.1]

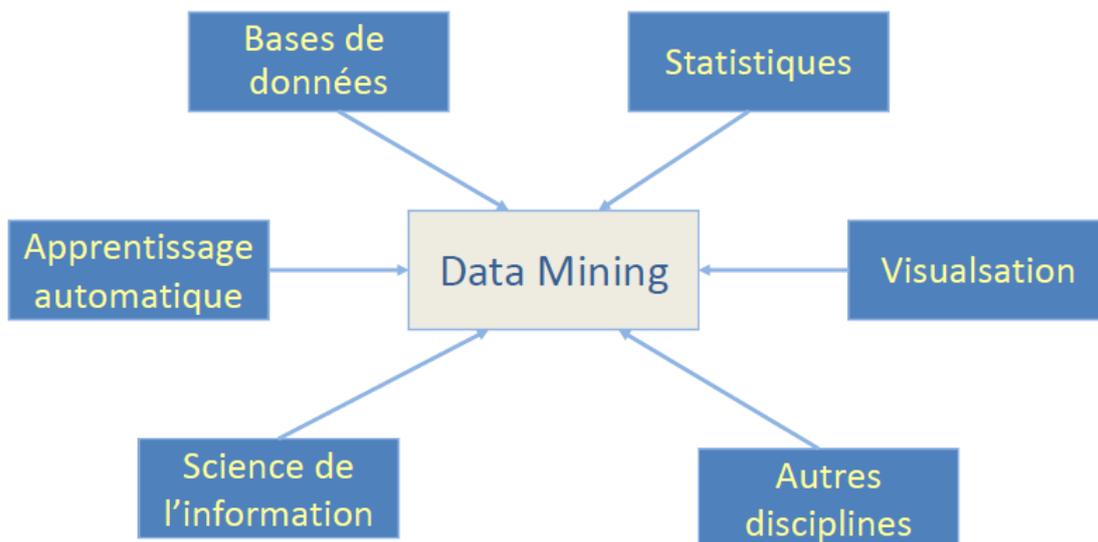


FIGURE 1.1: Les différentes Sciences et Techniques utilisées dans le DATA MINING

Le Data Mining se propose donc d'extraire des connaissances à partir de grands volumes de données qui peuvent être stockées de manière diverse, dans des bases de données, dans un (ou plusieurs) entrepôt de données (data Warehouse), mais qui peuvent aussi être récupérées de sources riches plus ou moins structurées comme internet (Web), ou encore en temps réel (retrait d'argent dans un distributeur de billets...).

1.2 Principe et spécificité

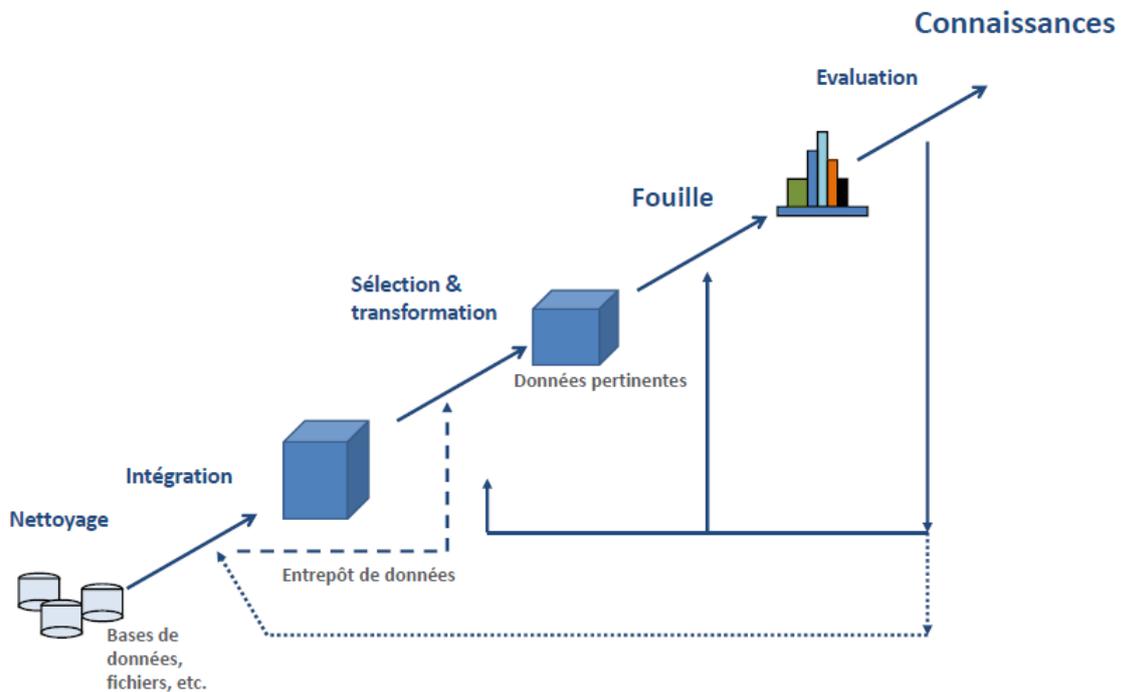


FIGURE 1.2: Architecture du Data Mining

1.2 Principe et spécificité

1.2.1 Stratégie du Data Mining

Les statisticiens utilisent en générale des données expérimentales pour étudier un phénomène, en fixant certaines hypothèses. Par contre, le Data Mining utilise les données pour formuler des hypothèses valables.

Le Data Mining tente alors de réaliser un arbitrage entre validité scientifique, interprétabilité des résultats et facilité d'utilisation, dans un environnement professionnel où le temps d'étude joue un rôle majeur et où les analystes ne sont pas toujours des statisticiens[15].

1.2.2 Méthodologie

Le Data Mining est un processus d'extraction de connaissances d'un large volume de données comportant les étapes suivantes :

1.3 Taxinomie des méthodes

1. Formaliser ou identifier un problème que l'organisation cherche à résoudre en termes de données en cernant les objectifs.
2. Accéder aux données appropriées.
3. Préparer les données en vue des traitements et utilisations futures (collecter, nettoyer, enrichir, coder et normaliser les données).
4. Modéliser les données en leur appliquant des algorithmes d'analyse.
5. Évaluer et valider les connaissances ainsi extraites des analyses.
6. Déployer les analyses dans l'entreprise pour une utilisation effective.

Ce processus est cyclique, le Data Mining participe à une meilleure compréhension de l'activité de l'organisation, et à une rationalisation avancée du stockage de l'information et des données[1].

1.3 Taxinomie des méthodes

Afin de construire des modèles à partir des données, le Data Mining se propose d'utiliser un ensemble d'algorithmes issus de disciplines scientifiques diverses (statistiques, intelligence artificielle, base de données...). Donc, le but est de trouver des schémas « intéressants » (des patterns ou motifs de conception) selon des critères fixés au départ, et extraire de ces données un maximum de connaissances utiles à l'entreprise. Il existe deux typologies de méthodes à savoir :

1.3.1 Selon les objectifs

Selon des objectifs voulus, nous avons trois types de méthodes :

1. **Classification** : Consiste à examiner les caractéristiques d'un objet et lui attribuer une classe (attribuer ou non un prêt à un client, établir un diagnostic...).

Donc, la classification consiste à apprendre à classer des objets dans des classes prédéfinies à partir d'exemples déjà classés. L'application de la classification construit un modèle à partir d'un ensemble d'apprentissage, où les éléments sont groupés dans des classes prédéfinies, et

1.3 Taxinomie des méthodes

ensuite utilise ce modèle pour classifier automatiquement de nouveaux objets dans l'une des classes prédéfinies.

2. **Régression** : Consiste à prédire la valeur future d'un attribut en fonction d'autres attributs (prédire la "qualité" d'un client en fonction de son revenu, ...). Le but de la prédiction est d'attribuer à des objets une valeur pour une variable continue.
3. **Segmentation(Clustering)** : Consiste à former des groupes homogènes à l'intérieur d'une population. Le clustering non supervisée consiste à chercher des groupes (ou classes) d'objets en se basant sur les caractéristiques de ces variables[15].

1.3.2 Méthodes d'apprentissage

Les méthodes non-supervisées

L'objectif de la méthode non supervisée est de trouver des relations entre les variables explicatives et permettant d'augmenter les connaissances du domaine étudié. Elles permettent de travailler sur un ensemble de données dans lequel aucune des données ou des variables à disposition n'a d'importance particulière par rapport aux autres, c'est-à-dire un ensemble de données dans lequel aucune variable n'est considérée individuellement comme la cible de l'objectif de l'analyse. Parmi les techniques disponibles, nous avons :

- Techniques à base de réseau de neurones.
- Techniques utilisées classiquement dans le monde des statistiques : Clustering ascendante hiérarchique, k-means et les nuées dynamiques (Recherche des plus proches voisins), les classifications mixtes, les classifications relationnelles...
- Les techniques dites de recherche d'associations.

Les méthodes supervisées

Dans le cadre de l'approche supervisée, les données sont constituées d'un ensemble de caractéristiques, appelées variables exogènes ou variable explicative, décrivant chaque individu. Chaque individu possède une caractéristique

particulière appelée variable endogène variable à expliquer ou variable cible. L'objectif de la fouille de données supervisée est de trouver des relations entre les variables exogènes permettant d'expliquer et/ou de prévoir le comportement de la variable endogène. La fouille de données supervisée se décompose en deux étapes : apprentissage d'un modèle sur une partie des données et validation du modèle sur l'autre partie des données. Les données sont donc divisées en deux ensembles : l'ensemble d'apprentissage et l'ensemble de test. Parmi les techniques disponibles, nous avons :

- Techniques à base d'arbres de décision.
- Techniques statistiques de régressions linéaires et non linéaires.
- Techniques à base de réseau de neurones : Perceptron mono ou multicouches avec ou sans rétro-propagation des erreurs.
- Techniques à base d'algorithme génétique.
- Techniques à base d'inférence bayésienne (Réseau bayésien).

1.4 Data mining et statistiques

Les méthodes d'exploration de données sont certainement plus complexes que celles des statistiques descriptives élémentaires. Ils sont basés sur des outils d'intelligence artificielle (réseaux de neurones), la théorie de l'information (arbres de décision), la théorie de l'apprentissage automatique ,et surtout les statistiques inférentielles et l'analyse de données conventionnelles comprenant l'analyse factorielle, le clustering et le discriminant... .

1.5 Méthodes du Data Mining à l'étude du churn

Le churn du client comme concept nouvellement développé a été largement utilisé dans les télécommunications. La rétention des clients est une question de plus en plus pressante dans l'environnement concurrentiel actuel. Parmi les méthodes utilisées pour l'études du churn :

1.5.1 Étude sur le churn des clients utilisant la méthode d'analyse de survie

Définition analyse de survie

L'analyse de survie est une technique statistique qui permet de faire l'estimation du temps écoulé jusqu'à la survenue d'un événement précis. Par exemple, elle permet de déterminer la durée de survie de patients ayant eu un infarctus cardiaque ou la durée de chômage d'une personne avant l'obtention d'un nouvel emploi.

Application de l'analyse de survie à l'étude du churn

La technique proposée par [Guozheng Zhang et al] comporte une phase de clustering[7].

Cette phase consiste à faire une segmentation (clustering) pour séparer les clients en groupes hétérogènes puis la deuxième phase l'analyse de survie. Pour la segmentation, les clients sont regroupés selon leurs caractères de survie puis l'analyse de survie donne la fonction de survie/risque qui sert à identifier la tendance de churn et tester la validité du regroupement pour obtenir la segmentation correcte des clients.

Cette approche s'est réalisée en plusieurs étapes. Tout d'abord la mission a été définie en sélectionnant et en confirmant les attributs des clients utiles à l'étude du churn. Les données sur le comportement général des clients sont préparées et extraites de l'entrepôt de données (datawarehouse). La segmentation est ensuite réalisée à l'aide de technique de Data Mining comme la méthode de K-means en regroupant les clients ayant des possibilités de survie similaires. Un numéro est assigné à chaque groupe de clients après la segmentation.

1.5.2 Étude sur le churn utilisant la méthode d'arbre de décision

Définition de la méthode de l'arbre de décision

L'arbre de décision est une technique statistique classé parmi les apprentissages supervisés. Les arbres de décision fonctionnent en partitionnant récur-

sivement l'ensemble d'entraînement afin d'obtenir des sous-ensembles aussi purs que possible pour une classe cible donnée.

Application de l'arbre de décision à l'étude du churn

La plupart des méthodes de segmentation des clients sont basées sur le revenu moyen par utilisateur ou Average Revenue Per User (ARPU) en anglais et ne prend pas en considération les revenus futurs ou le coût de l'entretien du client.

Cette approche proposée par [Han et al] propose un nouveau moyen de segmentation. Ce modèle est utilisé pour faire une prédiction sur le cycle de vie d'un client en observant seulement ses informations démographiques. Cinq modèles de décision est appliqués sur chaque client : valeur actuel, valeur historique, prédiction de valeur à long terme, crédit et fidélité[8].

1.5.3 Étude de churn utilisant la régression logistique

Définition de la méthode de régression logistique

La régression logistique est une technique de data mining qui permet de faire la classification. Elle est l'un des modèles d'analyse multi variée. Elle permet de mesurer l'association entre la survenue d'un évènement (variable expliquée qualitative) et les facteurs susceptibles de l'influencer (variables explicatives). Elle s'applique dans divers domaine par exemple :

- En médecine, elle permet par exemple de trouver les facteurs qui caractérisent un groupe de sujets malades par rapport à des sujets sains.
- Dans le domaine des assurances, elle permet de cibler une fraction de la clientèle qui sera sensible à une police d'assurance sur tel ou tel risque particulier.
- Dans le domaine bancaire, pour détecter les groupes à risque lors de la souscription d'un crédit.
- En économétrie, pour expliquer une variable discrète. Par exemple, les intentions de vote aux élections.

Application de la régression logistique à l'étude du churn

Une étude de [Oghojafor et al, 2012] utilise la régression logistique pour examiner les effets des facteurs socio-économiques sur l'attrition (churn) des clients. Cela est faite en analysant les facteurs qui influence le client de changer d'opérateur fournisseur de service. La variable cible catégorielle est donc l'intention de quitter son actuel opérateur. L'objectif est donc d'évaluer les impacts des facteurs démographiques et socio-économiques sur la volonté de churn. Les facteurs sont l'âge, le sexe, le type de travail, le revenu, le statut marital[10].

Deux modèles ont été construits lors de cette approche. Le premier est construit à partir des attributs communs à la prédiction de churn tels que les frais d'appel, le type de service et le nombre de connexion mobile. Le second modèle utilise les attributs du premier modèle mais s'ajoutant des facteurs démographiques et socio-économiques. Cette approche montre que la plupart des variables indépendantes sont très importantes pour la prédiction. Les taux d'appels élevés et les mauvaises installations de services se distinguent parmi les déterminants identifiés.

1.5.4 Étude sur le churn utilisant la méthode de Support Vector Machine (SVM)

Définition de la méthode SVM

Une machine à vecteurs de support (Support Vector Machine ou SVM en anglais) est un classificateur discriminant formellement défini par un hyperplan de séparation.

d'autres termes, les données de formation étiquetées données (apprentissage supervisé), l'algorithme produit un hyperplan optimal qui classe de nouveaux exemples.

Application de la méthode de SVM à l'étude du Churn

Une des approches étudiées [Coussement et Poel, 2008] consistait à appliquer SVM à un contexte de churn d'abonnement aux journaux. L'objectif était de

construire un modèle de churn précis en utilisant cette technique. La performance de prévision du churn du client a été comparée à celle de la régression logistique et aux prévisions aléatoires. Les auteurs ont choisi comme mesure principale d'évaluer leurs modèles de zone sous courbe (Area Under Curve ou AUC en anglais)[3].

1.5.5 Étude de Churn utilisant les réseaux de neurones

Définition de la méthode des réseaux de neurones

Le réseau de neurones artificiels est un modèle de calcul utilisé dans l'apprentissage par machine (Machine Learning), l'informatique et d'autres disciplines de recherche tel que le data mining. Elle est classée parmi les apprentissages supervisés. Un réseau neuronal s'inspire du fonctionnement des neurones biologiques et prend corps dans un ordinateur sous forme d'un algorithme. Le réseau neuronal peut se modifier lui-même en fonction des résultats de ses actions, ce qui permet l'apprentissage et la résolution de problèmes sans algorithme, donc sans programmation classique.

Algorithme de réseau de neurones

- Assigner un poids aléatoire pour chaque arête pour débiter l'algorithme.
- En utilisant les entrées et les liens, trouver le taux d'activation des noeuds cachés.
- En utilisant le taux d'activation de noeuds cachés et les liens vers les sorties, trouver le taux d'activation des noeuds de sorties.

Application de la méthode des réseaux de neurones à l'étude du churn

Les techniques hybrides de data mining combinant deux techniques ou plus ont gagné de la visibilité. Dans un grand nombre de domaines de problèmes, ces techniques s'avèrent offrir de meilleures performances que les techniques simples.

Une approche de [Tsai et Lu, 2009] considère deux modèles hybrides[14].

1.5 Méthodes du Data Mining à l'étude du churn

Il combine deux techniques de réseaux neurones différentes pour la prédiction de churn : réseaux de neurones artificiels à rétro-propagation (Back-propagation Artificial Neural Network ANN) et des cartes auto-organisées (Self-Organizing Map (SOM)). Le premier modèle hybride est la combinaison de deux ANN, l'autre est une combinaison d'un ANN et d'un SOM. La première technique des deux modèles effectue une tâche de réduction des données en filtrant les données non représentatives pour l'apprentissage. Ensuite, les sorties sont utilisées pour créer le modèle de prédiction basé sur la seconde technique. A fin d'évaluer la performance de ces modèles, trois ensembles de données tests différents ont été développés.

Une autre approche de [Bott, 2014] étudie l'application de réseaux neurones Perceptron multicouches (Multi Layer Perceptron (MLP)) avec la rétro-propagation pour identifier les facteurs qui ont une plus grande influence sur le churn des clients. Deux méthodes ont été examinées et comparées ; La méthode typique de modification par erreur et la méthode basée sur les poids ANN. Dans les deux méthodes, trois attributs ont été identifiés comme importants pour la prédiction de Churn par les deux modèles. Ces attributs sont les frais mensuels totaux, le total des appels sortants internationaux et le service 3G.

2

Réseaux de neurones

De façon formelle, un Réseau de Neurone (RN) est une fonction mathématique à laquelle sont associés des variables, un résultat et des paramètres ajustables (poids). À partir d'un ensemble de données (mesures, résultats de calcul, indices économiques ou financiers, etc.)

On peut ajuster un RN en choisissant convenablement ses paramètres. La propriété fondamentale des Réseaux de Neurones est l'émergence d'un comportement globale intelligent à partir de comportements de base élémentaires en plus de leurs capacités d'apprentissage, fait d'eux une représentation mathématique très avantageuse pour la modélisation statique et dynamique des processus[4].

L'idée principale des réseaux de neurones est la suivante : On se donne une unité simple, un neurone, qui est capable de réaliser quelques calculs élémentaires. On relie ensuite entre elles un nombre important de ces unités et on essaye de déterminer la puissance de calcul du réseau ainsi obtenu. Il est important de noter que ces neurones manipulent des données numériques et non pas symboliques.

2.1 Neurone formel

Un neurone est une unité de calcul élémentaire dont le modèle est issu de certains principes de fonctionnement du neurone biologique. Sa première modélisation date des années quarante. Elle a été présentée par **Warren McCulloch** et **Walter Pitts**. S'inspirant de leurs travaux sur les neurones biologiques[5].

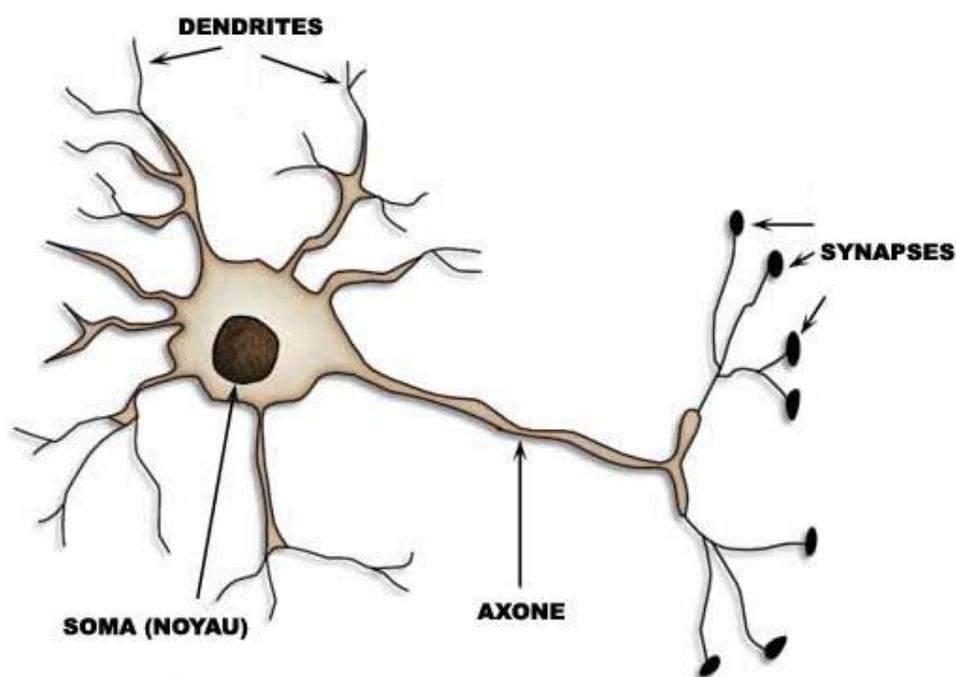


FIGURE 2.1: Analogie entre le neurone biologique et artificiel

Par analogie entre le neurone biologique et artificiel nous constatons ce qui suite :

neurone artificiel	neurone biologique
Poids de connexions	Synapses
Signale de sortie	Axones
Signale d'entrée	Dendrites
Fonction d'activation	Noyau

TABLE 2.1: Analogie entre neurone biologique et neurone artificiel

2.1 Neurone formel

Le neurone formel est caractérisé par :

- Les X_i représentent les vecteurs d'entrées, elles proviennent soit des sorties d'autres neurones, ou bien des données initiales .
- Les W_{ij} sont les poids synaptiques du neurone j . Ils correspondent à l'efficacité synaptique dans les neurones biologiques ($W_{ij} > 0$: synapse excitatrice; $W_{ji} < 0$: synapse inhibitrice). Ces poids pondèrent les entrées et peuvent être modifiés par apprentissage.
- Biais : entrée prend souvent les valeurs -1 ou $+1$ qui permet d'ajouter de la flexibilité au réseau en permettant de varier le seuil de déclenchement du neurone par l'ajustement des poids et du biais lors de l'apprentissage.
- Noyau : intègre toutes les entrées et le biais et calcul la sortie du neurone selon une fonction d'activation qui est souvent non linéaire pour donner une plus grande flexibilité d'apprentissage.

D'après la figure suivante :

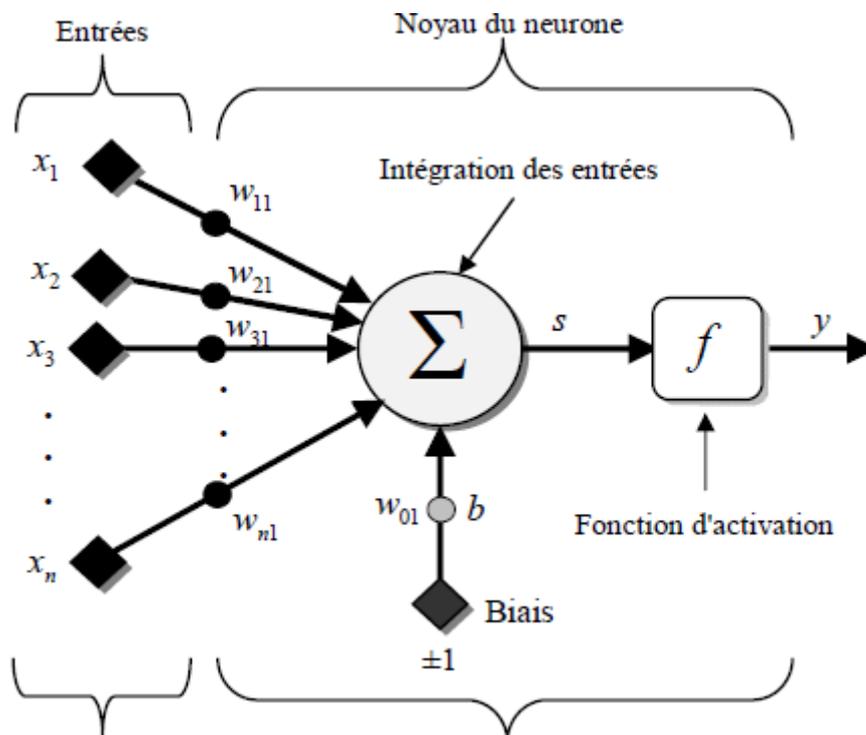


FIGURE 2.2: Modèle d'un neurone artificiel

Un neurone réalise simplement une fonction non linéaire, paramétrée de ses variables d'entrée. L'intérêt des neurones réside dans les propriétés qui ré-

2.2 Modélisation d'un neurone formel

sultent de leurs associations aux réseaux, c'est-à-dire de la composition des fonctions non linéaires réalisées par chacun des neurones. On distingue deux types de réseaux de neurones : les réseaux non bouclés (réalise une ou plusieurs fonctions de ses entrées, par composition des fonctions réalisées par chacun des neurones) et les réseaux bouclés (le graphe des connexions peut contenir des cycles).

2.2 Modélisation d'un neurone formel

La modélisation consiste à mettre en oeuvre un système de réseau de neurones sous un aspect non pas biologique mais artificiel. Cela suppose que d'après le principe biologique on aura une correspondance pour chaque élément composant le neurone biologique, donc une modélisation pour chacun d'entre eux. Le résumé de cette modélisation est donnée par le tableau ([2.1] page 21), qui nous a permis de voir clairement la transition entre le neurone biologique et le neurone formel. Le modèle mathématique d'un neurone artificiel est illustré à la figure [2.2]. Un neurone est essentiellement constitué d'un intégrateur qui effectue la somme pondérée de ses entrées. Le résultat S de cette somme est ensuite transformé par une fonction de transfert f qui produit la sortie y du neurone. Les n entrées du neurone correspondent au vecteur $x = [x_1, x_2, x_3, \dots, x_n]^T$, alors que $W = [W_1, W_2, W_3, \dots, W_n]^T$ représente le vecteur des poids de neurone.[5]

La sortie S est donnée par l'équation suivante :

$$\begin{aligned} S &= \sum_{i=1}^n W_{i1}x_i \pm b \\ &= W_{11}x_1 + W_{21}x_2 + W_{31}x_3 + \dots + W_{n1}x_n \pm b \end{aligned} \quad (2.1)$$

Avec b le biais est un neurone dans lequel la fonction d'activation est en permanence égale à 1. On peut aussi écrire sous la forme matricielle :

$$S = W^T x \pm b \quad (2.2)$$

2.2 Modélisation d'un neurone formel

Cette sortie correspond à une somme pondérée des poids et des entrées plus ce qu'on nomme le biais b du neurone. Le résultat S de la somme pondérée s'appelle le niveau d'activation du neurone. Le biais b s'appelle aussi le seuil d'activation du neurone. Lorsque le niveau d'activation atteint ou dépasse le seuil b , alors l'argument de f devient positif (ou nul). Sinon, il est négatif.[19]

2.2.1 Fonction d'activation

Une fonction d'activation est une fonction mathématique utilisé sur un signal. Elle va reproduire le potentiel d'activation que l'on retrouve dans le domaine de la biologie du cerveau humain. Elle va permettre le passage d'information ou non de l'information si le seuil de stimulation est atteint. Concrètement, elle va avoir pour rôle de décider si on active ou non une réponse du neurone. Les fonctions d'activations les plus utilisées sont les fonctions « seuil » (hard limit), «linéaire» et «sigmoïde».

Type de fonctions d'activations :

Les différents types de neurones se distinguent par la nature de leur fonction d'activation. Les principaux types sont :

- La fonction linéaire (la fonction identité) : Utilisé en couche de sortie pour une utilisation en cas de régression. On peut la caractériser de nulle, puisque les unités de sortie seront identiques à leur niveau d'entré. Intervalle de sortie $] - \infty, +\infty[$.
- La fonction seuil $f(x) = \mathbb{I}_{[0, +\infty[}(x)$.
- la fonction sigmoïde(logistic) $f(x) = \frac{1}{1+e^{-x}}$:la plus populaire depuis des décennies. Mais aujourd'hui, elle devient beaucoup moins efficace par rapport à d'autre lors de l'utilisation de les couches cachées. Elle perd de l'information due à une saturation que cela soit pour la phase de feed forward (La technique du « feedforward » consiste à formuler des demandes ou à proposer des options ou des solutions tournées vers l'avenir, contrairement à la technique du feedback, qui fait un retour sur une situation passée) ou de backpropagation (Dans l'apprentissage machine, la rétropropagation est un algorithme largement utilisé dans

2.3 Différents modèles de réseaux de neurones

la formation de réseaux neuronaux à action anticipée pour l'apprentissage supervisé.), en donnant des effets non linéaires au réseau due à un paramètre unique. Elle a aussi des soucis de gradient 0 avec des entrées étant très large, même si le soucis est minimalisé avec les systèmes utilisant des batch par lots (mini batch) (en informatique, un traitement par lots (batch processing) est un enchaînement automatique d'une suite de commandes). cette fonction est utilisée en couche de sortie pour la classification binaire, avec un intervalle de sortie $[0, 1]$

- *ReLU* (rectified linear unit) $f(x) = \max(0, x)$: Ce sont les fonctions les plus populaires de nos jours. Elles permettent un entraînement plus rapide comparé aux fonctions sigmoïde, étant plus légères. Attention au phénomène de "Dying ReLU", auquel on préférera les variations de ReLU. Très utilisé pour les CNN, et les réseaux de multi-perceptron. Intervalle de sortie $[0, +\infty[$.

2.3 Différents modèles de réseaux de neurones

2.3.1 Modèle de Hopfield

Le modèle de Hopfield fut présenté en 1982. Ce modèle très simple est basé sur le principe des mémoires associatives (Les mémoires associatives servent à accélérer la recherche d'une donnée dans un ensemble de données.). C'est d'ailleurs la raison pour laquelle ce type de réseau est dit associatif (par analogie avec le pointeur qui permet de récupérer le contenu d'une case mémoire).

Le modèle de Hopfield utilise l'architecture des réseaux entièrement connectés et récurrents (dont les connexions sont non orientées et ou chaque neurone n'agit pas sur lui-même). Les sorties sont en fonction des entrées et du dernier état pris par le réseau.

2.3.2 Modèle de Kohonen

Ce modèle a été présenté par T. Kohonen en 1982 en se basant sur des constatations biologiques. Il a pour objectif de présenter des données complexes et

2.3 Différents modèles de réseaux de neurones

appartenant généralement à un espace discret de grandes dimensions dont la topologie est limitée à une ou deux dimensions.

Les cartes de Kohonen sont réalisées à partir d'un réseau à deux couches, une en entrée et une en sortie. Notons que les neurones de la couche d'entrée sont entièrement connectés à la couche de sortie.

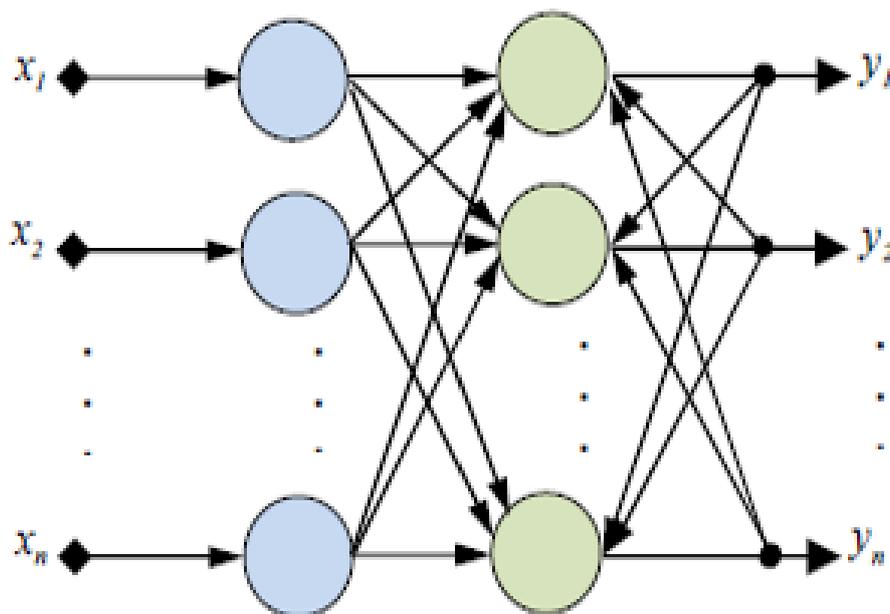


FIGURE 2.3: Modèle de Kohonen

2.3.3 Le modèle de Perceptron

Le mécanisme perceptron fut inventé par le psychologue F. Rosenblatt à la fin des années 50.

Il représentait sa tentative d'illustrer certaines propriétés fondamentales des systèmes intelligents en générale.

Le réseau dans ce modèle est formé de trois couches : Une couche d'entrée, fournissant des données à une couche intermédiaire, chargée des calculs, cela en fournissant la somme des impulsions qui lui viennent des cellules auxquelles elle est connectée, et elle répond généralement suivant une loi définie avec un seuil, elle-même connectée à la couche de sortie (couche de décision). Cette dernière couche renvoie des signaux à la couche intermédiaire, jusqu'à ce que leurs connexions se stabilisent (figure [2.4]).

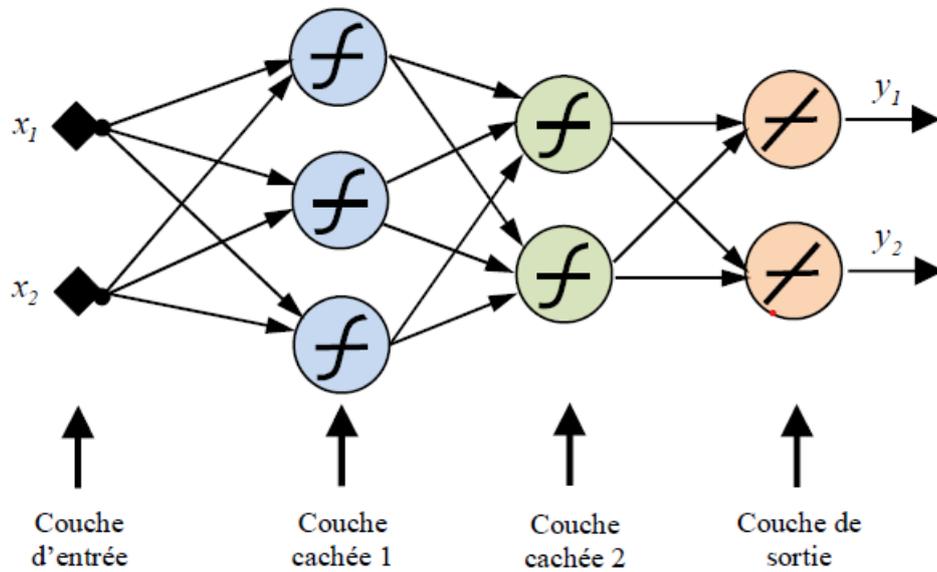


FIGURE 2.4: Modèle de Perceptron

2.3.4 Le modèle Adaline

L'adaline de Widrow et Hoff est un réseau à trois couches : une d'entrée, une couche cachée et une couche de sortie. Ce modèle est similaire au modèle de perceptron, seule la fonction de transfert change, mais reste toujours linéaire. Les modèles des neurones utilisés dans le perceptron et l'adaline sont des modèles linéaires.

2.4 Types de réseaux de neurones

2.4.1 Réseaux de neurones non bouclés

Un réseau de neurones non bouclé réalise une ou plusieurs fonctions algébriques de ses entrées, par composition des fonctions réalisées par chacun de ses neurones.

Un réseau de neurones non bouclé est représenté graphiquement par un ensemble de neurones "connectés" entre eux, l'information circulant des entrées vers les sorties sans "retour en arrière".

On distingue quatre types :

2.4 Types de réseaux de neurones

1 Réseaux de neurones mono-couches :

La structure d'un réseau mono-couche est telle que des neurones organisés en entrée soient entièrement connectés à d'autres neurones organisés en sortie par une couche modifiable de poids (figure [2.5]).

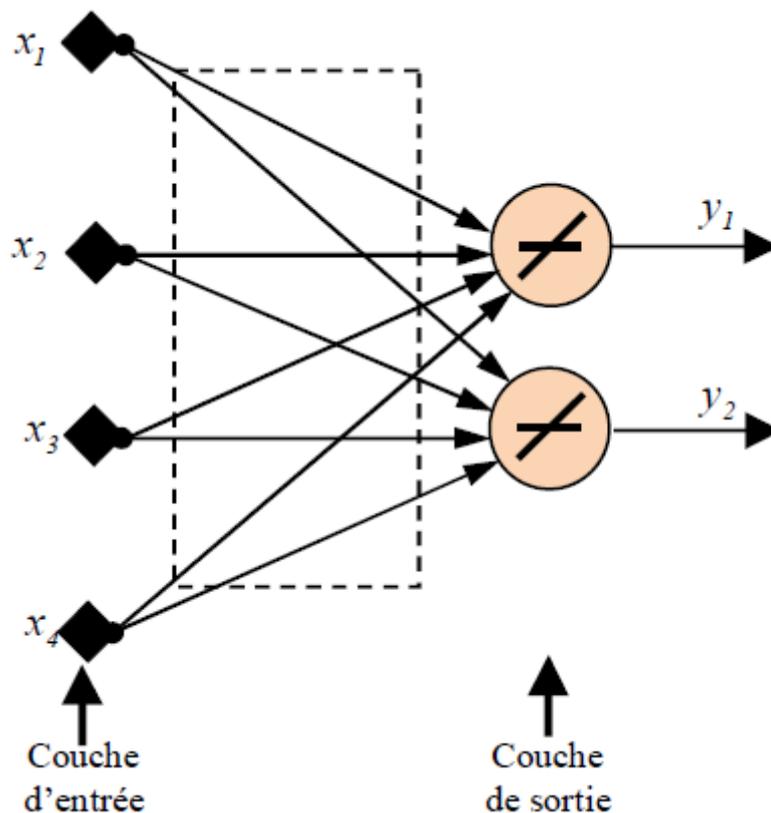


FIGURE 2.5: Un réseau de neurones mono-couche

2 Réseaux de neurones multi-couches :

- Couche entrée : contient l'ensemble des neurones d'entrées, cette couche est une couche passive, ses neurones n'effectuent aucun traitement.
- Couches cachées : les couches intermédiaires n'ayant aucun contact avec l'extérieur.
- Couche de sorties : contient l'ensemble des neurones de sorties.

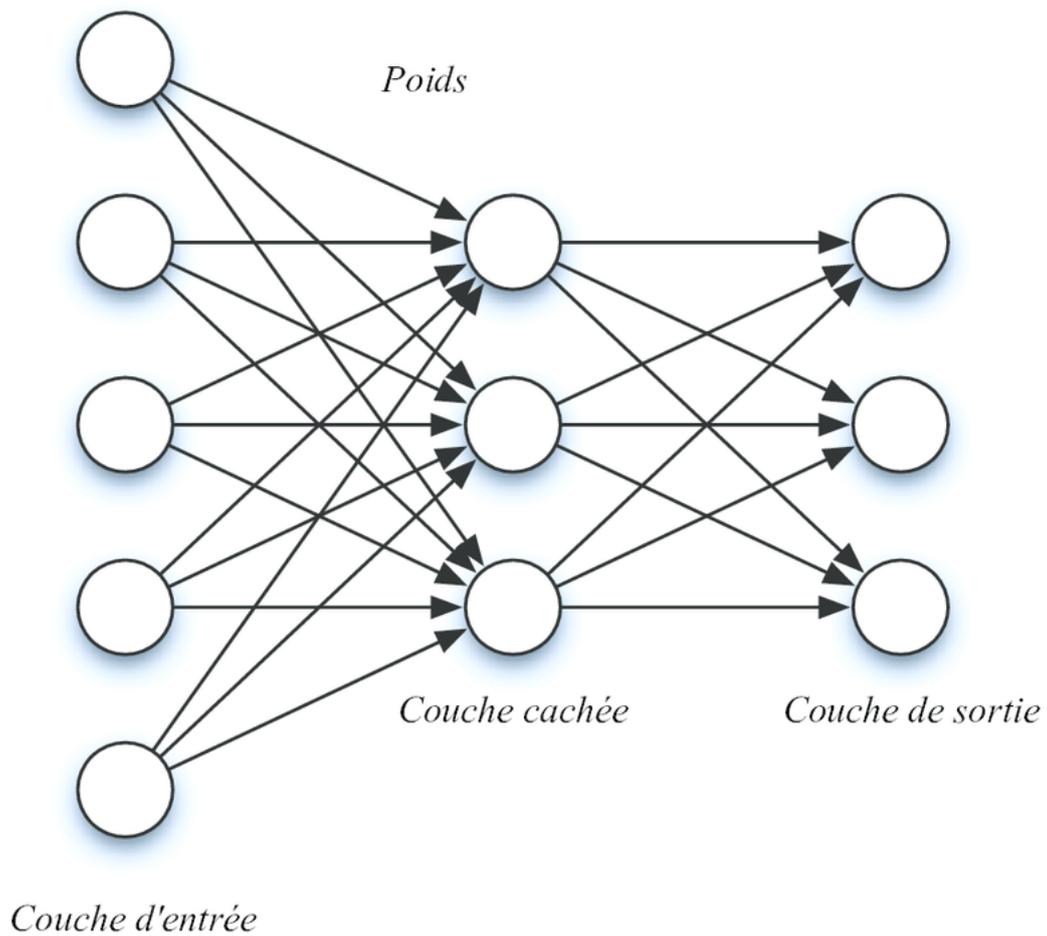


FIGURE 2.6: Un perceptron multi-couches contenant trois couches

3 Réseaux de neurones à connexion complète :

C'est la structure d'interconnexion la plus générale. Chaque neurone est connecté à tous les neurones du réseau (et à lui-même) (figure [2.7]).

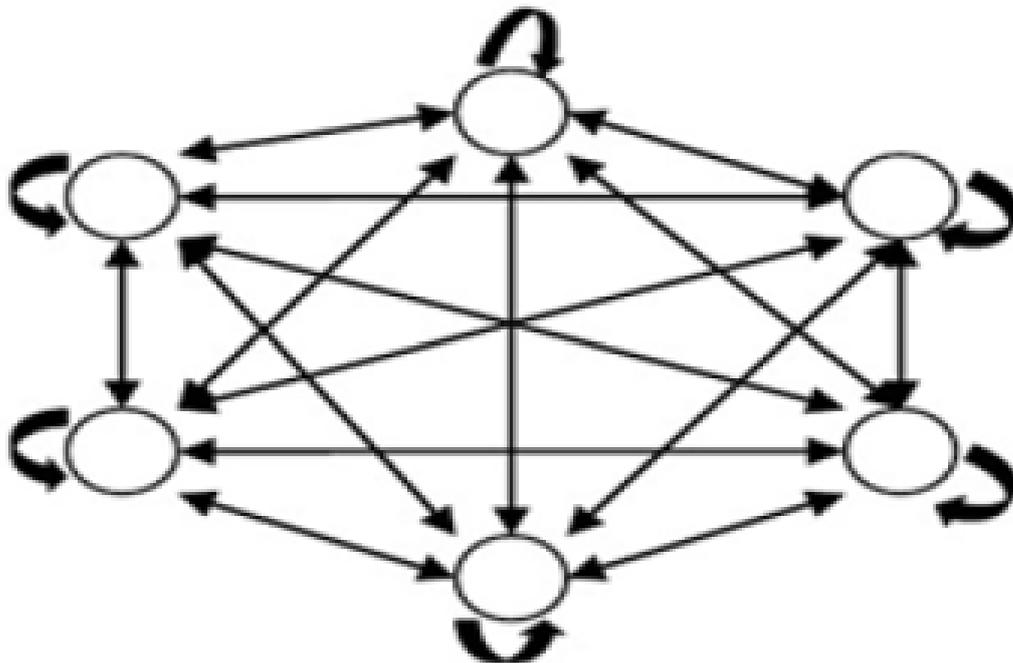


FIGURE 2.7: Un réseau de neurones à connexion complète

4 Réseaux de neurones à connexions locales :

Il s'agit d'une structure multicouche, mais Chaque neurone entretient des relations avec un nombre réduit et localisé de neurones de la couche de sortie. Les connexions sont donc moins nombreuses que dans le cas d'un réseau multicouche classique (figure [2.8]).

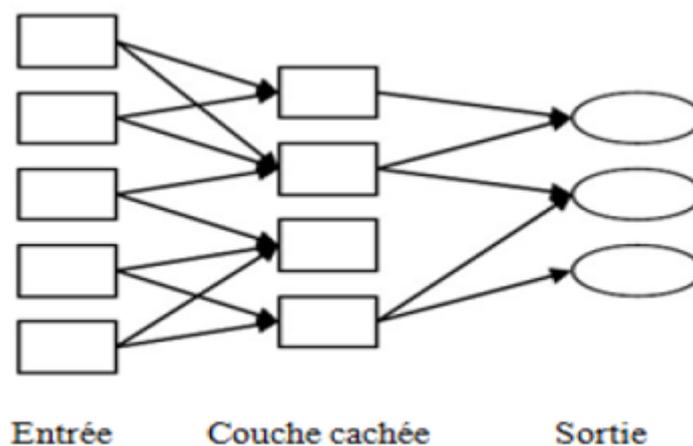


FIGURE 2.8: Un réseau de neurones à connexions locales

2.4.2 Les réseaux de neurones bouclés (récurrents)

Un réseau de neurone bouclé à temps discret réalise une ou plusieurs équations aux différences non linéaire, par composition des fonctions réalisées par chacun des neurones et des retards associés à chacune des connexions. Ces réseaux caractérisent par la présence d'au moins une boucle de rétroaction au niveau des neurones ou entre les couches, et la prise en compte de l'aspect temporel du phénomène (figure[2.9]).

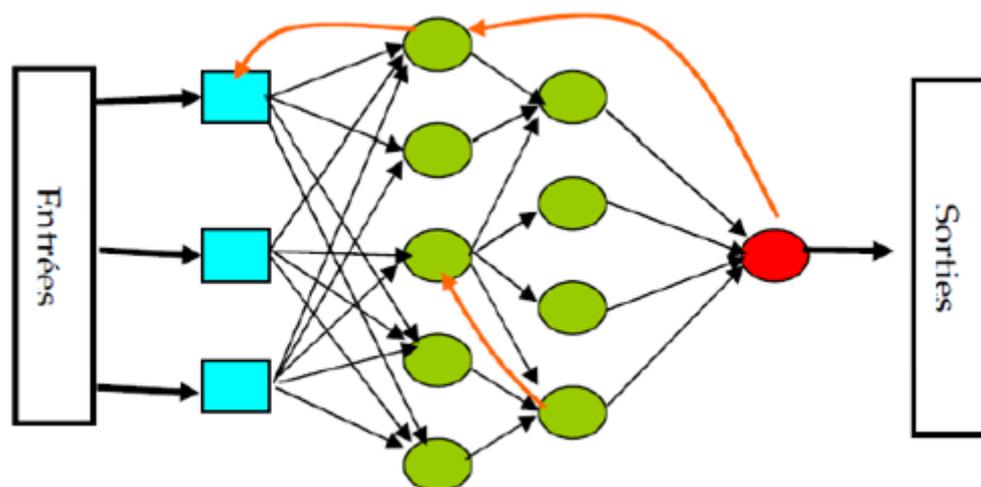


FIGURE 2.9: Un réseau de neurone bouclé

2.5 Apprentissage des Réseaux de neurones artificiels

L'apprentissage des RNA, est une phase qui permet de déterminer ou de modifier les paramètres du réseau. Plusieurs algorithmes d'apprentissage ont été développés depuis la première règle d'apprentissage de Hebb en 1949.

Les RNA se divisent en deux principales classes, les réseaux à apprentissage **supervisés** et les réseaux à apprentissage **non supervisés**.

- Pour les réseaux à apprentissage supervisés (Perceptron, Adaline), on présente au réseau des entrées, et au même temps les sorties que l'on désirerait pour cette entrée. Le réseau doit alors se reconfigurer, c'est-à-dire calculer ses poids afin que la sortie qu'il donne corresponde bien à la sortie désirée.

2.6 Étapes de la conception d'un réseau de neurone :

- Pour les réseaux à apprentissage non supervisé (Hopfield, Kohonen), on présente une entrée au réseau et on le laisse évoluer librement jusqu'à ce qu'il se stabilise.

2.6 Étapes de la conception d'un réseau de neurone :

Pour construire un réseau de neurones, la première chose à faire est de bien choisir ses échantillons de données d'apprentissage, de tests et validation. Ce n'est qu'en suite que le choix du type de réseau interviendra. Afin de clarifier un peu les idées, voici chronologiquement les quatre grandes étapes qui doivent guider la création d'un réseau de neurones :

- **Choix et préparation des échantillons** : Le processus d'élaboration d'un réseau de neurones commence toujours par le choix et la préparation des échantillons de données.
- **Élaboration de la structure du réseau** : La structure du réseau dépend étroitement du type des échantillons.
- **Apprentissage** : L'apprentissage consiste tout d'abord à calculer les pondérations optimales des différentes liaisons, en utilisant un échantillon.
- **Validation et Tests** : Alors que les tests concernent la vérification des performances d'un réseau de neurones hors échantillon et sa capacité de généralisation, la validation est parfois utilisée lors de l'apprentissage. Une fois le réseau calculé, il faut toujours procéder à des tests afin de vérifier que notre réseau réagit correctement.

3

Algorithme Génétique

Les Algorithmes Génétiques **AGs** ont été initialement développés par John Holland (1975).

Leurs champ d'application sont très vastes. Outre l'économie (minimisation du risque des portefeuilles), ils sont utilisés pour l'optimisation de fonctions, en finance, en théorie du contrôle optimal (recherche opérationnelle), ou encore en théorie des jeux répétés et différentiels (en l'occurrence dans les jeux évolutionnaires et le dilemme du prisonnier) et la recherche d'information (Google) ainsi que la recherche des plus courts chemins en théorie des graphes (routages Internet ou GPS). La raison de ce grand nombre d'applications est claire : simplicité et efficacité. Bien sûr, d'autres techniques d'exploration stochastiques existent, la méthode de Monte-Carlo peut-être considéré comme un concept similaire.

3.1 Les heuristiques

Les heuristiques sont des méthodes spécifiques à un problème particulier. Elles nécessitent des connaissances du domaine du problème traité. En fait, se sont des règles empiriques qui se basent sur l'expérience et les résultats acquis afin d'améliorer les recherches futures. Plusieurs définitions des heuristiques ont été proposées par plusieurs chercheurs, parmi lesquelles :

- Une méthode heuristique (ou simplement une heuristique) est une méthode qui aide à découvrir la solution d'un problème en faisant des conjectures plausibles mais faillible de ce qui est la meilleure chose à faire[6].
- Une heuristique est une règle d'estimation, une stratégie, une méthode ou astuce utilisée pour améliorer l'efficacité d'un système qui tente de découvrir les solutions des problèmes complexes[9].
- Les heuristiques sont des ensembles de règles empiriques ou des stratégies qui fonctionnent, en effet, comme des règles d'estimation[12].

3.2 Les métaheuristiques

Méta, du grec « au-delà », heuristique du grec qui signifie « trouver ». Une métaheuristique est un algorithme d'optimisation visant à résoudre des problèmes d'optimisation difficile (la majorité d'entre eux sont dans les domaines de la recherche opérationnelle, de l'ingénierie ou de l'intelligence artificielle) pour lesquels on ne connaît pas de méthode classique plus efficace.

Les métaheuristiques sont généralement des algorithmes stochastiques itératifs, qui progressent vers un optimum global, c'est-à-dire l'extremum global d'une fonction, par échantillonnage d'une fonction objectif [18].

La définition de **S.Martello** et **I.H.Osman** « Une métaheuristique est un processus maître itératif qui guide et modifie les opérations des heuristiques subordonnées pour produire efficacement des solutions de haute qualité. Il peut manipuler une solution unique complète ou incomplète ou une collection de solutions à chaque itération. Les heuristiques subordonnées peuvent être des procédures de haut ou bas niveau, ou une simple recherche locale,

ou simplement une méthode de construction.»[13].

3.2.1 Classification des méthodes approchée

En générale , être regroupées sous trois catégories :

- Métaheuristiques basées sur la recherche locale.
- Métaheuristiques basées sur la population.
- Métaheuristiques hybrides.

Métaheuristiques basées sur la recherche locale :

La recherche locale, appelée aussi méthode de descente, désigne une classe de méthodes appliquant des médications élémentaires à une solution en vue d'obtenir des solutions localisées dans son voisinage à l'intérieur de l'espace de solutions. Le voisinage est exploré jusqu'à ce qu'une condition d'arrêt soit satisfaite. Ce mécanisme de recherche a été intégré dans plusieurs métaheuristiques dont, parmi les plus connues :

- Le recuit simulé (simulated annealing).
- La recherche tabou.

Métaheuristiques basées sur la population

En tout temps on dispose d'une "base" de plusieurs points, appelée population. L'exemple le plus connu est l'algorithme génétique [18].

3.2.2 Applications

Les métaheuristiques sont souvent employées pour leur facilité de programmation et de manipulation. Elles sont en effet facilement adaptables à tout type de problème d'optimisation. Toutefois, elles sont le plus judicieusement employées sur des problèmes d'optimisation difficile, où des méthodes d'optimisation plus classiques (méthodes déterministes, notamment) montrent leurs limites.

3.3 Historique des algorithmes génétiques

Le premier à rechercher le contenu d'algorithme génétique est le professeur de psychologie John Henry Holland (avec ses collègues), en 1960. La première chose que ces chercheurs ont introduit la prise en compte de l'opérateur d'enjambement (cross-over) en complément des mutations. Et c'est cet opérateur qui permet le plus souvent de se rapprocher de l'optimum d'une fonction en combinant les gènes contenus dans les différents individus de la population. La première publication du résultat de la recherche a été publiée en 1975 dans le magazine « Adaptation in Natural and Artificial System ».

3.4 Définitions

1. Un "algorithme génétique" est défini par un : individu, chromosome, séquence et une solution potentielle au problème donné.
2. Une "population" est un ensemble de chromosomes ou de points de l'espace de recherche.
3. "L'environnement" est assimilé à l'espace de recherche.
4. La fonction que nous cherchons à optimiser est appelée "fonction de fitness".

Les algorithmes génétiques appartiennent à la famille des algorithmes évolutionnistes.

Leur but est d'obtenir une solution approchée à un problème d'optimisation, lorsqu'il n'existe pas de méthode exacte (ou que la solution est inconnue) pour le résoudre en un temps raisonnable.

Les algorithmes génétiques utilisent la notion de sélection naturelle et l'appliquent à une population de solutions potentielles au problème donné. Ils sont des algorithmes d'optimisation stochastiques itérés fondés sur les mécanismes de la sélection naturelle et de la génétique[16].

3.5 Fonctionnement

Leur fonctionnement est relativement simple :

1. Nous partons avec une population de solutions potentielles (chromosomes) initiales arbitrairement choisies. Nous évaluons leur performance (fitness) relative.
2. Sur la base de ces performances, nous créons une nouvelle population de solutions potentielles en utilisant des opérateurs évolutionnaires simples : la sélection, le croisement et la mutation.
3. Nous recommençons ce cycle jusqu'à ce que nous trouvions une solution satisfaisante.

Algorithm 1 L'algorithme génétique

```
1: Début
2:   Initialiser les paramètres nécessaires ;
3:   Initialiser une population de N individus ;
4:   Évaluer les N individus ;
5:   Tant que la condition d'arrêt n'est pas satisfaite faire :
6:     Utiliser l'opérateur de sélection pour sélectionner K individus ;
7:     Appliquer l'opérateur de croisement sur les K individus avec la probabilité  $P_c$  ;
8:     Appliquer l'opérateur de mutation sur les individus avec la probabilité  $P_m$  ;
9:     Utiliser l'opérateur d'évaluation pour évaluer les enfants obtenus ;
10:    Utiliser l'opérateur de sélection pour remplacer quelques individus parents ;
11:  Fin Tant que
12: Fin
```

3.6 Les opérateurs

3.6.1 L'opérateur de sélection

Cet opérateur est peut-être le plus important puisqu'il permet aux individus d'une population de survivre, de se reproduire ou de mourir. En règle générale, la probabilité de survie d'un individu sera directement liée à son efficacité relative au sein de la population.

La méthode la plus utilisée et connue est la roue de loterie biaisée (roulette

3.6 Les opérateurs

wheel) de Goldberg (1989).

Selon cette méthode, chaque chromosome sera dupliqué dans une nouvelle population proportionnellement à sa valeur d'adaptation.

Nous effectuons, en quelque sorte, autant de tirages avec remise qu'il y a d'éléments dans la population. Ainsi, dans le cas d'un codage binaire, la fitness d'un chromosome particulier étant $f(d(A))$ la probabilité avec laquelle il sera réintroduit dans la nouvelle population de taille N est :

$$\frac{f(d(A))}{\sum_{j=1}^N f(d(A))} \quad (3.1)$$

Les individus ayant une grande fitness ont donc plus de chance d'être sélectionnés. Nous parlons alors de "sélection proportionnelle".

3.6.2 L'opérateur de croisement

L'opérateur de croisement permet la création de nouveaux individus selon un processus fort et simple. Il permet donc l'échange d'information entre les chromosomes (individus). Tout d'abord, deux individus, qui forment alors un couple, sont tirés au sein de la nouvelle population issue de la reproduction. Puis un (potentiellement plusieurs) site de croisement est tiré aléatoirement (chiffre entre -1 et 1). Enfin, selon une probabilité P_i que le croisement s'effectue, les segments finaux (dans le cas d'un seul site de croisement) des deux parents sont alors échangés autour de ce site :

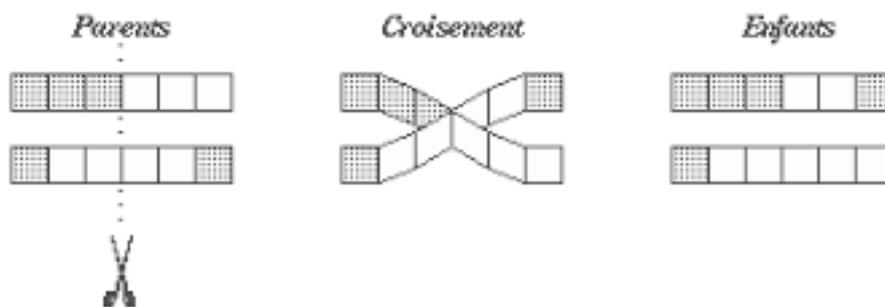


FIGURE 3.1: Exemple de croisement

Cet opérateur permet la création de deux nouveaux individus. Toutefois, un individu sélectionné lors de la reproduction ne subit pas nécessairement l'action d'un croisement. Ce dernier ne s'effectue qu'avec une certaine probabilité. Plus cette probabilité est élevée et plus la population subira de changement. [20]

3.6.3 L'opérateur de mutation

Le rôle de cet opérateur est de modifier aléatoirement, avec une certaine probabilité, la valeur d'un composant de l'individu. Dans le cas du codage binaire, chaque bit $a_i \in \{0, 1\}$ est remplacé selon une probabilité P_m par son inverse $a'_i = 1 - a_i$. Comme le montre la figure ci dessus. Tout comme plusieurs lieux de croisement peuvent être possibles, nous pouvons très bien admettre qu'une même chaîne puisse subir plusieurs mutations.

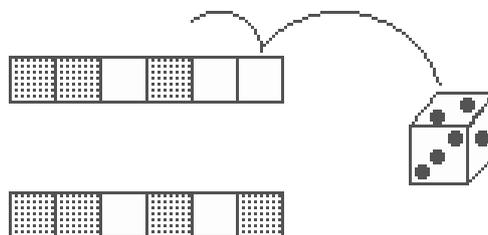


FIGURE 3.2: Exemple de mutation

La mutation est traditionnellement considérée comme un opérateur marginal bien qu'elle confère en quelque sorte aux algorithmes génétiques .

Cet opérateur est d'une grande importance. Il a de fait un double rôle : celui d'effectuer une recherche locale et/ou de sortir d'une trappe (recherche éloignée). [20]

Les opérateurs de l'algorithme génétique sont guidés par un certain nombre de

3.6 Les opérateurs

paramètres fixés à l'avance. La valeur de ces paramètres influence la réussite ou non d'un algorithme génétique. Ces paramètres sont les suivants :

- La taille de la population N , et la longueur du codage de chaque individu (dans le cas de codage binaire). Si N est trop grand, le temps de calcul de l'algorithme peut s'avérer très important, et si N est trop petit, il peut converger trop rapidement vers un mauvais chromosome.
- La probabilité de croisement P_c . Elle dépend de la forme de la fonction de fitness. Son choix est en général heuristique (tout comme pour P_m). Plus elle est élevée, plus la population subit évidemment de changements importants. Les valeurs généralement admises sont comprises entre 0.5 et 0.9.
- La probabilité de mutation P_m . Ce taux est généralement faible puisqu'un taux élevé risque de conduire à une solution sous-optimale.

4

Application

L'objet de ce dernier chapitre est de mettre en pratique la méthode abordée dans le chapitre précédent avec les données « Telecom customer attrition ». Dans le but d'atteindre notre objectif, nous devons passer par deux étapes bien distinctes : préparation des données et analyse descriptive.

La deuxième étape consiste en l'implémentation de l'algorithme génétique et d'effectuer plusieurs évaluations sur ces derniers. Pour terminer, la troisième partie consiste à sélectionner les meilleurs hyperparamètres du modèle en fonction des données à disposition afin de garantir la meilleure performance possible .

4.1 Langage R

R est un langage de programmation et un logiciel libre destiné aux statistiques et à la science des données soutenu par la R Foundation for Statistical Computing. Il fait partie de la liste des paquets GNU (On appelle paquet GNU un paquet logiciel maintenu par le projet GNU et soutenu par la Free Software Foundation) et est écrit en C (langage). R est une implémentation du langage de programmation S (est un langage de programmation de très haut niveau et un environnement d'analyse des données et des graphiques conçu dans les années 1975-1976 par John Chambers) avec l'ajout de la portée lexicale, inspirée du Scheme ((prononciation : /skim/) est un langage de programmation dérivé du langage fonctionnel Lisp, créé dans les années 1970 au Massachusetts Institute of Technology (MIT) par Gerald Jay Sussman et Guy L. Steele), et du ramasse-miettes (informatique). Le projet R naît en 1993 comme un projet de recherche de Ross Ihaka et Robert Gentleman à l'université d'Auckland (Nouvelle-Zélande).

Depuis 1997, une vingtaine de développeurs forment l'équipe de développement de R (R Development Core team).

Les membres de cette équipe ont les droits d'écriture sur le code source. Le 23 avril 1997 débute le Comprehensive R Archive Network (CRAN) puis le 5 décembre 1997, R est intégré au Projet GNU.

La version R 1.0.0, première version officielle du langage R, est publiée le 29 février 2000.

En 2003, l'équipe de développement crée la R Foundation for Statistical Computing pour soutenir le projet R et devenir un point de contact de référence pour ceux qui veulent prendre contact avec la communauté R. À ce moment, le langage compte plus de 200 bibliothèques développées par la communauté scientifique qui utilise R.

La version 2.0 est publiée le 4 octobre 2004 et la version 3.0 le 3 avril 2013. En 2015, plusieurs acteurs économiques importants comme IBM, Microsoft ou encore la société R-Studio créent le R Consortium pour soutenir la communauté R et financer des projets autour de ce langage.

4.1.1 Pourquoi utiliser R

- Il permet l'utilisation des méthodes statistiques classiques à l'aide de fonctions prédéfinies.
- Il permet de créer ses propres programmes dans un langage de Programmation assez simple d'utilisation.
- Il permet d'utiliser des techniques statistiques innovantes et récentes à l'aide de package développés par les chercheurs et mis à disposition sur le site du CRAN.
- Open Source, évolutif, la référence scientifique, en augmentation constante.

4.2 Définition de PACKAGE

Les packages sont des collections de fonctions R. Le répertoire où les packages sont stockés est appelée library. R est livré avec un ensemble standard de packages. D'autres sont disponibles pour le téléchargement et l'installation.

- Le **TIDYVERSE** est une collection de paquets R conçus pour la science des données. Tous les paquets partagent une philosophie de conception, une grammaire et des structures de données sous-jacentes ¹.
- Le paquet **JANITOR** est un paquet R qui a des fonctions simples pour l'examen et le nettoyage des données sales. Il a été conçu pour les utilisateurs débutants et intermédiaires de R et est optimisé pour la convivialité ².
- Le paquet **DATA TABLE** offre une solution rapide et efficace en termes de mémoire : lecteur et rédacteur de fichiers, des agrégations, des mises à jour, des fonctions équivalentes, range et intervalle, ajuster les tableaux , dans une syntaxe courte et flexible, pour un développement plus rapide ³.
- Le paquet **DPLYR** est Un outil rapide et cohérent pour travailler avec des objets semblables à des cadres de données, en mémoire et hors mémoire ⁴.

1. <https://www.tidyverse.org>

2. <https://cran.r-project.org/web/packages/janitor/index.html>

3. <https://www.rdocumentation.org/packages/data.table/versions/1.13.0>

4. <https://www.rdocumentation.org/packages/dplyr/versions/0.7.8>

4.2 Définition de PACKAGE

- Le paquet **ROSE** fournit des fonctions permettant de traiter les problèmes de classification binaire en présence de classes déséquilibrées. Des échantillons synthétiques équilibrés sont générés selon ROSE (Menardi et Torelli, 2013). Des fonctions qui mettent en œuvre des remèdes plus traditionnels au déséquilibre des classes sont également fournies, ainsi que différentes mesures pour évaluer la précision d'un apprenant. Celles-ci sont estimées par des méthodes d'attente, de bootstrap ou de validation croisée⁵.
- **reticulate** Interface avec les modules, les classes et les fonctions "Python". Lors d'un appel en "Python", les types de données R sont automatiquement convertis en types "Python" équivalents. Lorsque des valeurs sont renvoyées de "Python" à R, elles sont reconverties en types R. Compatible avec toutes les versions de "Python" supérieur ou égale à la version 2.7⁶.
- Interface avec **Keras**⁷, une "API" de réseaux neuronaux de haut niveau. Keras a été développé pour permettre une expérimentation rapide, prend en charge les réseaux basés sur la convolution et les réseaux récurrents (ainsi que les combinaisons des deux), et fonctionne de manière transparente sur les appareils à processeur central et à processeur graphique.
- Interface avec **TensorFlow**⁸, une bibliothèque de logiciels libres pour le calcul numérique à l'aide de graphiques de flux de données. Les nœuds du graphique représentent les opérations mathématiques, tandis que les bords du graphique représentent les tableaux de données multidimensionnelles (tenseurs) qui communiquent entre eux. L'architecture flexible vous permet de déployer le calcul sur un ou plusieurs "CPU" ou "GPU" dans un ordinateur de bureau, un serveur ou un appareil mobile avec une seule "API". TensorFlow a été développé à l'origine par des chercheurs et des ingénieurs travaillant au sein de l'équipe "Cerveau" de l'organisation de recherche "Machine Intelligence" de Google,

5. <https://rdrr.io/cran/ROSE/man/ROSE-package.html>

6. <https://rstudio.github.io/reticulate/>

7. <https://keras.io>

8. <https://www.tensorflow.org/>

4.3 Description des données

dans le but de mener des recherches sur l'apprentissage machine et les réseaux neuronaux profonds, mais le système est suffisamment général pour être également applicable dans une grande variété d'autres domaines.

- Le paquet **GA** boîte à outils flexible et polyvalente mettant en œuvre des algorithmes génétiques (AG) pour l'optimisation stochastique. Des représentations binaires, à valeur réelle et de permutation sont disponibles pour optimiser une fonction de fitness, c'est-à-dire une fonction fournie par les utilisateurs en fonction de leur fonction objective. Plusieurs opérateurs génétiques sont disponibles et peuvent être combinés pour explorer les meilleurs réglages pour la tâche en cours. En outre, les utilisateurs peuvent définir de nouveaux opérateurs génétiques et évaluer facilement leurs performances. La recherche locale utilisant des algorithmes d'optimisation générale peut être appliquée de manière stochastique pour exploiter des régions intéressantes. Les AG peuvent être exécutés de manière séquentielle ou parallèle, en utilisant une parallélisation maître-esclave explicite ou une approche par îlots à gros grains⁹.

4.3 Description des données

Les données sont la consommation quotidienne de la facture téléphonique dans les ÉTATS UNIS.

Le DATA se compose de 3333 lignes (clients) , et 21 colonnes (Dans les colonnes, nous trouvons plusieurs informations Nombre d'appels ...etc),chaque ligne représente une observation,c'est-à-dire la consommation d'un client enregistrée sur toute la journée.Chaque ligne est indexée par un indicateur du client (PHONE) .Le tableau ci-dessous présente un descriptif de chacune des variables initiales (tableau 4.1).

9. <https://www.rdocumentation.org/packages/GA/versions/3.2>

4.3 Description des données

	state	account_length	area_code	phone	intl_plan	v_mail_plan	v_mail_message	day_mins	day_calls	day_charge
1	17	128	415	382-4657	0	1	25	265.1	110	45.07
2	36	107	415	371-7191	0	1	26	161.6	123	27.47
3	32	137	415	358-1921	0	0	0	243.4	114	41.38
4	36	84	408	375-9999	1	0	0	299.4	71	50.90
5	37	75	415	330-6626	1	0	0	166.7	113	28.34
6	2	118	510	391-8027	1	0	0	223.4	98	37.98
7	20	121	510	355-9993	0	1	24	218.2	88	37.09
8	25	147	415	329-9001	1	0	0	157.0	79	26.69
9	19	117	408	335-4719	0	0	0	184.5	97	31.37
10	50	141	415	330-8173	1	1	37	258.6	84	43.96
11	16	65	415	329-6603	0	0	0	129.1	137	21.95
12	40	74	415	344-9403	0	0	0	187.7	127	31.91
13	13	168	408	363-1107	0	0	0	128.8	96	21.90
14	27	95	510	394-8006	0	0	0	156.6	88	26.62
15	13	62	415	366-9238	0	0	0	120.7	70	20.52
16	35	161	415	351-7269	0	0	0	332.9	67	56.59
17	14	85	408	350-8884	0	1	27	196.4	139	33.39
18	47	93	510	386-2923	0	0	0	190.7	114	32.42
19	46	76	510	356-2992	0	1	33	189.7	66	32.25
20	44	73	415	373-2782	0	0	0	224.4	90	38.15
21	10	147	415	396-5800	0	0	0	155.1	117	26.37
22	6	77	408	393-7984	0	0	0	62.4	89	10.61
23	4	130	415	358-1958	0	0	0	183.0	112	31.11
24	41	111	415	350-2565	0	0	0	110.4	103	18.77

FIGURE 4.1: Une partie du data

eve_mins	eve_calls	eve_charge	night_mins	night_calls	night_charge	intl_mins	intl_calls	intl_charge	cust_serv_calls	churn
197.4	99	16.78	244.7	91	11.01	10.0	3	2.70	1	FALSE
195.5	103	16.62	254.4	103	11.45	13.7	3	3.70	1	FALSE
121.2	110	10.30	162.6	104	7.32	12.2	5	3.29	0	FALSE
61.9	88	5.26	196.9	89	8.86	6.6	7	1.78	2	FALSE
148.3	122	12.61	186.9	121	8.41	10.1	3	2.73	3	FALSE
220.6	101	18.75	203.9	118	9.18	6.3	6	1.70	0	FALSE
348.5	108	29.62	212.6	118	9.57	7.5	7	2.03	3	FALSE
103.1	94	8.76	211.8	96	9.53	7.1	6	1.92	0	FALSE
351.6	80	29.89	215.8	90	9.71	8.7	4	2.35	1	FALSE
222.0	111	18.87	326.4	97	14.69	11.2	5	3.02	0	FALSE
228.5	83	19.42	208.8	111	9.40	12.7	6	3.43	4	TRUE
163.4	148	13.89	196.0	94	8.82	9.1	5	2.46	0	FALSE
104.9	71	8.92	141.1	128	6.35	11.2	2	3.02	1	FALSE
247.6	75	21.05	192.3	115	8.65	12.3	5	3.32	3	FALSE
307.2	76	26.11	203.0	99	9.14	13.1	6	3.54	4	FALSE
317.8	97	27.01	160.6	128	7.23	5.4	9	1.46	4	TRUE
280.9	90	23.88	89.3	75	4.02	13.8	4	3.73	1	FALSE
218.2	111	18.55	129.6	121	5.83	8.1	3	2.19	3	FALSE
212.8	65	18.09	165.7	108	7.46	10.0	5	2.70	1	FALSE
159.5	88	13.56	192.8	74	8.68	13.0	2	3.51	1	FALSE
239.7	93	20.37	208.8	133	9.40	10.6	4	2.86	0	FALSE
169.9	121	14.44	209.6	64	9.43	5.7	6	1.54	5	TRUE
72.9	99	6.20	181.8	78	8.18	9.5	19	2.57	0	FALSE
137.3	102	11.67	189.6	105	8.53	7.7	6	2.08	2	FALSE

FIGURE 4.2: Une partie du data

4.3 Description des données

state	La localisation du client dans le pays
account length	Le Nombre de mois avec le fournisseur de télécommunications actuel
area code	Code régional
phone	Le numéro de téléphone de chaque client
intl plan	International
v mail plan	International vidéo mail
v mail message	International vidéo message
day mins	Le nombre de minutes dans le matin
day calls	Le nombre d'appels dans le matin
day charge	Le taux de chargement dans le matin
Eve mins	Le nombre de minutes dans le soir
eve calls	Le nombre d'appels dans le soir
eve charge	Le taux de chargement dans le soir
night mins	Le nombre de minutes dans la nuit
Night calls	Le nombre d'appels dans la nuit
night charge	Le taux de chargement dans la nuit
intl mins	Le nombre de minutes international
intl calls	Le nombre d'appels international
intl charge	Le taux de charge international
cust serv calls	Communications entre le client et le service téléphonique
churn	Variable binaire 1 si le client est un churner 0 sinon.

TABLE 4.1: Description des données

Tout d'abord, il faut changer les variables qui se façonnent des caractères (la variable STATE , et la variable CHURN), c'est pourquoi nous avons fait la Codification suivante :

Pour la variable CHURN :

FALSE	0
TRUE	1

TABLE 4.2: Codification de la variable CHURN

Pour la variable STATE :

4.4 Validation croisée (Cross validation)

Code	State	Codifi	Code	State	Codifi	Code	State	Codifi
AK	ALASKA	1	KY	KENTUCKY	18	NY	NEW YORK	35
AL	ALABAMA	2	LA	LOUISIANA	19	OH	OHIO	36
AR	ARKANSAS	3	MA	MASSACHUSETTS	20	OK	OKLAHOMA	37
AZ	ARIZONA	4	MD	MARYLAND	21	OR	OREGON	38
CA	CALIFORNIA	5	ME	MAINE	22	PA	PENNSYLVANIA	39
CO	COLORADO	6	MI	MICHIGAN	23	RI	RHODE ISLAND	40
CT	CONNECTICUT	7	MN	MINNESOTA	24	SC	SOUTH CAROLINA	41
DC	D OF COLOMBIA	8	MO	MISSOURI	25	SD	SOUTH DAKOTA	42
DE	DELAWARE	9	MS	MISSISSIPPI	26	TN	TENNESSE	43
FL	FLORIDA	10	MT	MONTANA	27	TX	TEXAS	44
GA	GEORGIA	11	NC	NORTH CAROLINA	28	UT	UTAH	45
HI	HAWAII	12	ND	NORTH DAKOTA	29	VA	VIRGINIA	46
IA	IOWA	13	NE	NEBRASKA	30	VT	VERMONT	47
ID	IDAHO	14	NH	NEW HAMPSHIRE	31	WA	WASHINGTON	48
IL	ILLINOIS	15	NJ	NEW JERSEY	32	WI	WISCONSIN	49
IN	INDIANA	16	NM	NEW MEXICO	33	WV	WEST VIRGINIA	50
KS	KANSAS	17	NV	NEVADA	34	WY	WYOMING	51

TABLE 4.3: Codification de la variable STATE

4.4 Validation croisée (Cross validation)

Une fois que nous avons fini de former notre modèle, nous ne pouvons pas supposer qu'il fonctionnera bien sur des données qu'il n'a pas encore vues. En d'autres termes, nous ne pouvons pas être sûrs que le modèle aura la précision et la variance souhaitées dans l'environnement de production. Nous avons besoin d'une sorte d'assurance de l'exactitude des prévisions que notre modèle produit. Pour cela, nous devons valider notre modèle. Ce processus consistant à décider si les résultats numériques quantifiant les relations hypothétiques entre les variables, sont acceptables en tant que descriptions des données, est connu sous le nom de validation. Pour évaluer les performances de tout modèle d'apprentissage machine, nous devons le tester sur des données invisibles. En se basant sur les performances du modèle sur des données invisibles, nous pouvons dire que notre modèle est :

- **Over-fitting** : Le sur-ajustement se produit quand le modèle se rapproche tellement de la fonction qu'il prête trop attention au bruit. Le modèle apprend la relation entre les entités et les étiquettes en tellement de détails et capte le bruit. Cela peut causer des fluctuations aléatoires dans la fonction.
- **Under-fitting** : Le sous-ajustement est le contraire du sur-ajustement.

4.4 Validation croisée (Cross validation)

C'est lorsque le modèle ne se rapproche pas assez de la fonction et est donc incapable de saisir la tendance sous-jacente des données. Il ignore une grande partie des entités et conséquemment produit des étiquettes incorrectes.

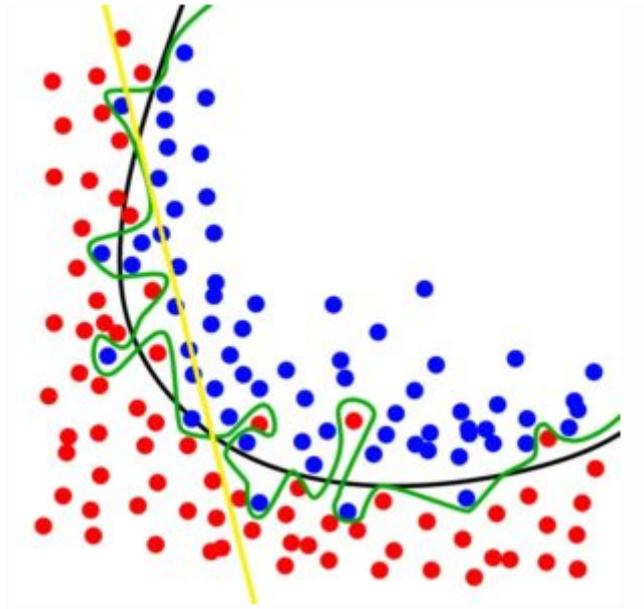


FIGURE 4.3: Over-fitting et Under-fitting

Le diagramme illustre le sur-ajustement (la ligne jaune). Alors que la ligne noire représente le modèle généralisé, la ligne verte représente un modèle sur-ajusté. Il tient compte de chaque point et amène le modèle à s'adapter au bruit, qu'on ne veut pas.

La validation croisée en anglais Cross Validation (CV) est l'une des techniques utilisées pour tester l'efficacité des modèles d'apprentissage machine, c'est aussi une procédure de ré-échantillonnage utilisée pour évaluer un modèle si nous disposons de données limitées.

Pour réaliser un CV, nous devons conserver un échantillon des données qui n'est pas utilisé pour former le modèle, puis utiliser cet échantillon pour tester/valider.

Les quelques techniques courantes utilisées pour le Cross validation :

- **Train-Test** Approche fractionnée : Cette approche consiste à diviser au hasard les données complètes en ensembles de formation et de test. Et à effectuer ensuite la formation modèle sur l'ensemble de formation et utiliser l'ensemble de test à des fins de validation, idéalement en

4.4 Validation croisée (Cross validation)

divisant les données en (70%/30% ou bien 80%/20%)



FIGURE 4.4: TEST et TRAIN data

- **K-Folds** Cross Validation : est une méthode populaire et facile à comprendre, elle donne généralement lieu à un modèle moins biaisé par rapport aux autres méthodes. Parce qu'il garantit que chaque observation de l'ensemble de données original a la chance d'apparaître dans l'ensemble de formation et de test. C'est l'une des meilleures approches si nous disposons d'un nombre limité de données d'entrée. Cette méthode suit les étapes suivantes :

4.5 Classification déséquilibrée (Imbalanced Classification)

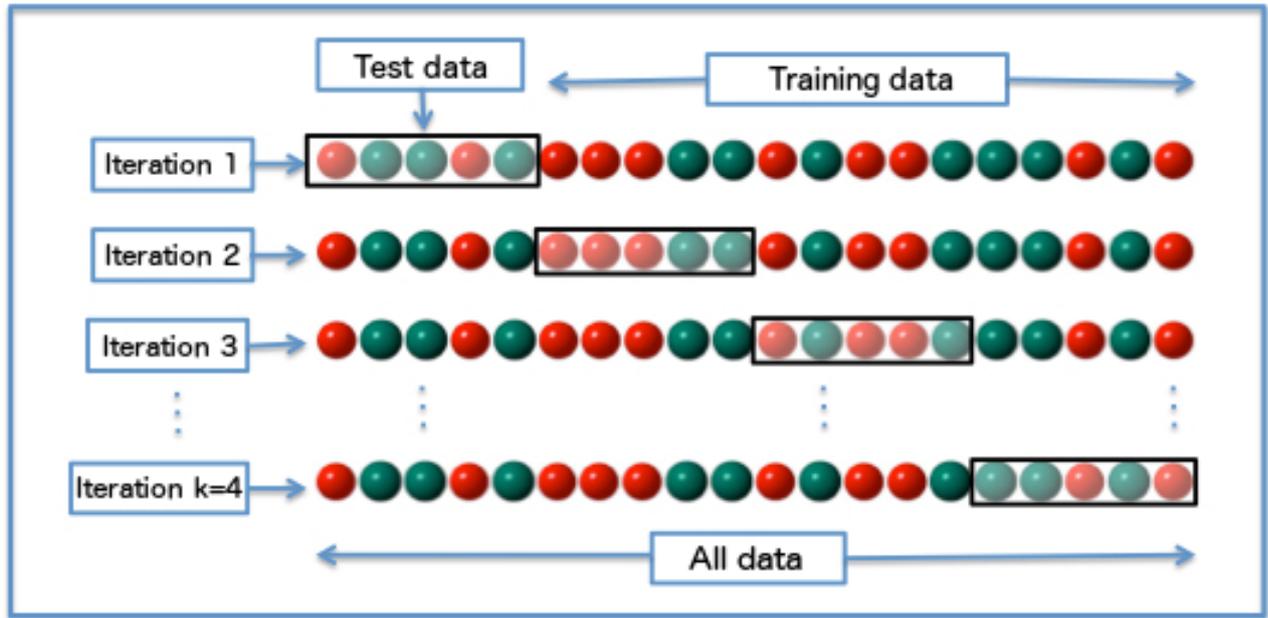


FIGURE 4.5: K fold cross validation

Dans notre cas , nous utilisons la méthode **Train-Test approche** , pour cela on utilise la fonction **sample** , avec la sélection des proportions suivantes (70% pour le **Train** et 30 % pour le **Test**).

4.5 Classification déséquilibrée (Imbalanced Classification)

Certains problèmes de classification affichent un déséquilibre au niveau des classes. C'est à- dire que l'une des classes est minoritaire par rapport à la population globale. En réalisant un diagramme sur notre tableau de données, nous nous sommes rendu compte que 2850 (85.50855%) clients appartiennent à la classe non churner et les 483 (14.49145%) clients restants à la classe churner.(figure4.6)

4.5 Classification déséquilibrée (Imbalanced Classification)

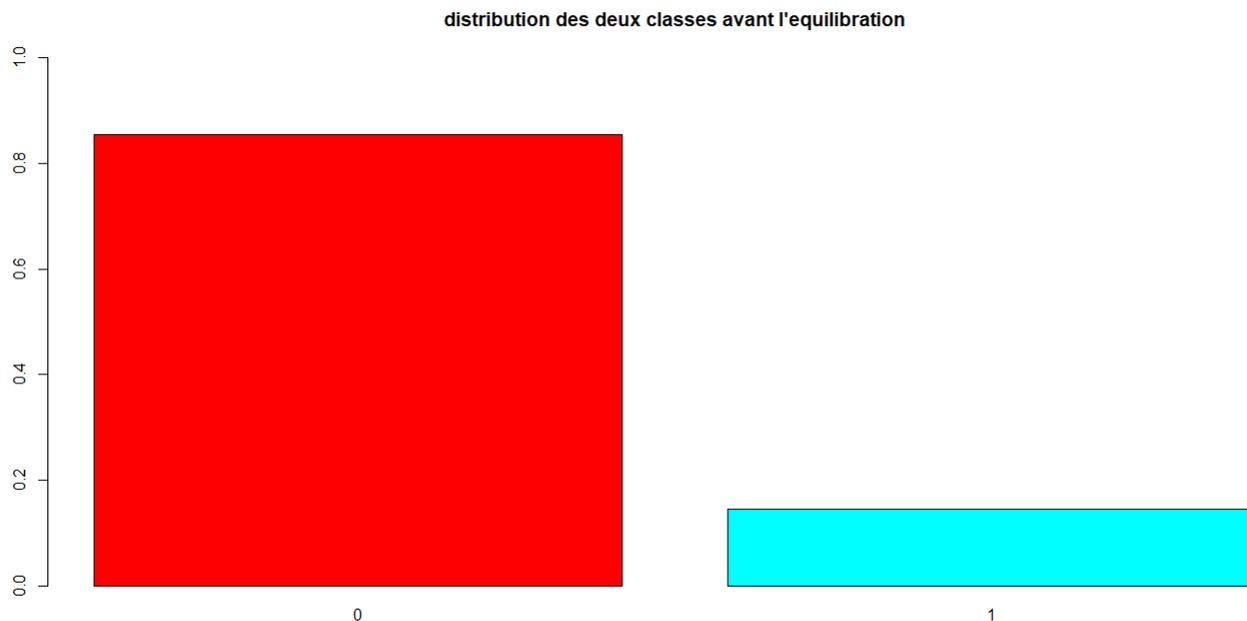


FIGURE 4.6: Les deux classes cherner et non cherner

La classification déséquilibrée est un problème d'apprentissage supervisé dans lequel une classe est largement supérieure à l'autre. Ce problème est plus fréquemment rencontré dans les problèmes de classification binaire que dans les problèmes de classification à plusieurs niveaux.

Le terme **déséquilibré** fait référence à la disparité rencontrée dans la variable dépendante.

Par conséquent, un problème de classification déséquilibré est un problème dans lequel la variable dépendante a une proportion déséquilibrée de classes. En d'autres termes, un ensemble de données qui présente une distribution inégale entre ses classes est considéré comme déséquilibré.

4.5.1 Les méthodes pour traiter les ensembles de données déséquilibrés

Ces méthodes sont largement connues sous le nom de "méthodes d'échantillonnage". En général, ces méthodes visent à modifier une donnée déséquilibrée en une distribution équilibrée en utilisant un mécanisme quelconque.

4.5 Classification déséquilibrée (Imbalanced Classification)

La modification se fait en changeant la taille de l'ensemble de données d'origine et en fournissant la même proportion d'équilibre.

Les méthodes utilisées pour traiter les ensembles de données déséquilibrés :

- **Sous-échantillonnage** (Undersampling) : Cette méthode fonctionne avec la classe majoritaire. Elle réduit le nombre d'observations de la classe majoritaire afin d'équilibrer l'ensemble des données. Cette méthode est à utiliser de préférence lorsque l'ensemble de données est énorme et la réduction du nombre d'échantillons d'entraînement permet d'améliorer le temps d'exécution et les problèmes de stockage.
- **Suréchantillonnage** (Oversampling) : Cette méthode fonctionne avec la classe minoritaire. Elle reproduit les observations de la classe minoritaire pour équilibrer les données. Elle est également connue sous le nom de suréchantillonnage. Tout comme le sous-échantillonnage, cette méthode peut également être divisée en deux types : Suréchantillonnage aléatoire et suréchantillonnage informatif.
- **Génération de données synthétiques** (Synthetic Data Generation) : En termes simples, au lieu de reproduire et d'ajouter les observations de la classe minoritaire, elle surmonte les déséquilibres en générant des données artificielles. C'est aussi une sorte de technique de suréchantillonnage.

En règle générale, lorsque nous sommes confrontés à ce type problème, on se retrouve avec un modèle qui va très bien prédire la classe majoritaire. Or que, dans notre cas, c'est la classe minoritaire qui nous intéresse. Pour remédier à cela, nous avons utilisé une technique qui consiste à rééquilibrer les données destinées à l'apprentissage (**Génération de données synthétiques**). En d'autres termes, nous avons diminué les données de la classe majoritaire, tout en rajoutant des données synthétisées à l'aide de la classe minoritaire.

On utilise le paquet **ROSE** pour la résolution de ce problème, on applique la **classification déséquilibrée** sur les données de **TRAIN DATA**.

On remarque que le **TRAIN DATA** se compose de 70% des données initiales, et dans ces données on trouve 2022 valeurs dans la classe non churner (% 85.87), et 335 valeurs dans la classe churner (% 14.23)(figure 4.7).

4.5 Classification déséquilibrée (Imbalanced Classification)



FIGURE 4.7: Les deux classes churner et non churner dans **TRAIN DATA**

La fonction **ovun sample** crée des échantillons éventuellement équilibrés par des exemples minoritaires de **sur-échantillonnage** aléatoire, des exemples majoritaires de **sous-échantillonnage** ou une combinaison de **sur-sous-échantillonnage** .

Dans notre cas , nous fixons le nombre d'échantillon (**2000**).et à partir de la nouvelle distribution est la suivante : (figure 4.8)

4.6 Implémentation du Réseau de Neurone

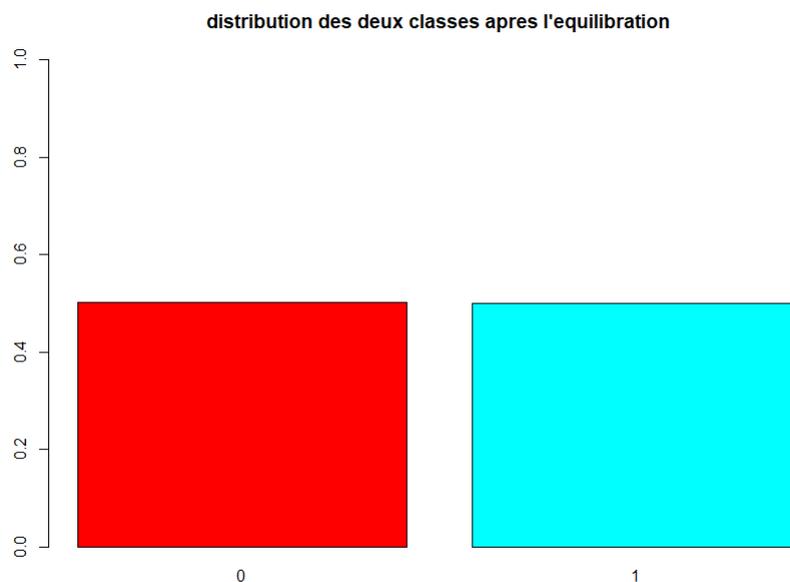


FIGURE 4.8: Les deux classes churner et non churner de **TRAIN DATA** après l'utilisation de la fonction **ovun sample**

on remarque que la nouvelle classe churner a 49.9% et la classe non churner a 50.1% , donc les deux classes sont équilibrées.

4.6 Implémentation du Réseau de Neurone

Pour commencer à construire un modèle, nous devons d'abord initialiser un modèle séquentiel à l'aide de la fonction **keras-model-sequential()** de l'**API KERAS**. Ensuite, nous sommes prêts à commencer la modélisation. Quant aux fonctions d'activation que nous utiliserions, il est préférable d'utiliser ici l'une des plus courantes afin de nous familiariser avec le Keras et les réseaux de neurones, à savoir la fonction d'activation **ReLU**. Cette fonction d'activation du redresseur est utilisée dans une couche cachée, ce qui est généralement une bonne pratique.

En outre, nous voyons également que la fonction d'activation softmax est utilisée dans la couche de sortie. nous faisons cela parce que nous voulons nous assurer que les valeurs de sortie sont comprises entre 0 et 1 et peuvent être utilisées comme des probabilités prédites.

4.7 Implémentation Algorithme Génétique

Dans **R**, on utilise le paquet **GA** dans tout ce qui concerne les algorithmes génétiques.

Dans notre cas on utilise la fonction **ga** pour la résolution du modèle pour obtenir le meilleur résultat pour les deux variables **units** , **Drop out** du modèle **MLP** .

La fonction **ga** prend en paramètres obligatoires :

- La fonction fitness et dans notre cas c'est le modèle du réseau de neurone.
- La métrique à maximiser.
- La population .
- Le nombre d'itérations : pour limiter le nombre d'itération afin de réduire le temps d'exécution qui peut être relativement long.
- Les bornes des hyperparamètres.
- Et pour les trois opérateurs (sélection,croisement,mutation),la fonction les prend automatiquement ou bien par défaut .

4.8 Interprétation des résultats

Après l'entraînement du modèle **MLP** (réseau de neurone)[figure(4.9)],et l'utilisation de l'algorithme génétique pour avoir les paramètres de ce dernier (avec les données de train data).

4.8 Interprétation des résultats

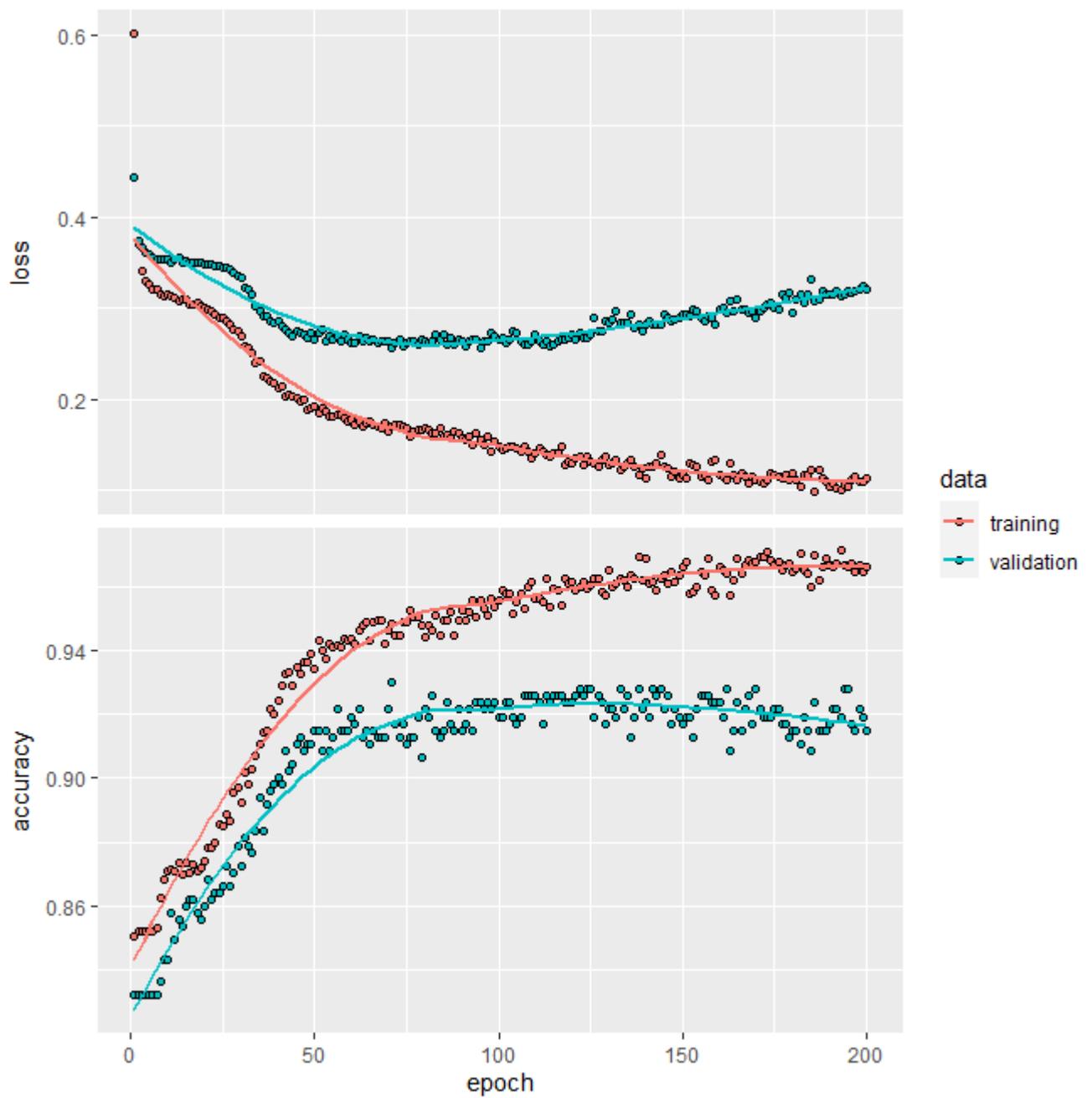


FIGURE 4.9: L'entraînement du modèle MLP

4.8 Interprétation des résultats

Type	Permutation
Taille de la population	10
Nombre de génération	25
Élitisme	1
Probabilité de croisement	0.8
Probabilité de mutation	0.1
Itération	25
Valeur de fitness	0.954023
Solution	X=[29 ; 26 ; 24 ; ... ; 30 ; 22]

TABLE 4.4: les résultats de l'AG Pour le paramètre **Units**

Type	Real-valued
Taille de la population	10
Nombre de génération	25
Élitisme	1
Probabilité de croisement	0.8
Probabilité de mutation	0.1
Itération	25
Valeur de fitness	0.9592476
Solution	X=[0.03301862 ; 0.07883593]

TABLE 4.5: les résultats de l'AG Pour le paramètre **Drop out**

On utilise les données *test_data* pour avoir les résultats finales de notre étude (prédiction du churn).

Après avoir effectuer l'algorithme nous avons obtenu le score suivant :

La valeur de la précision des prévisions (accuracy) est 91.9003%

Loss	Accuracy
0.2186976	0.919

4.8 Interprétation des résultats

On constate que la modélisation arrive à 797 non churn à true , et 88 churn à true .

		Predicted	
		0	1
Actual	0	797	53
	1	25	88

Il y avait 797 clients réellement négatives sont des non churn qu'ont été correctement classés (True negatives).

Et 88 clients réellement positifs churn ont été correctement classés (True positives).

Cependant, l'algorithme a mal classifié 25 clients qui sont non churn et il dit qui sont des churn (False positives).

Et il a mal classifié 53 clients qui sont des chun et il dit qui sont non churn (False negatives).

Et on a :

$$Acc = \frac{797+88}{797+88+25+53} = 0.919003$$

Et un taux d'erreur :

$$1 - Acc = 0.081$$

Et un taux de **Recall (Sensibilité)** (ou bien **Sensitivity** en anglais, Il s'agit d'une mesure des observations réelles qui sont étiquetées (prédites) correctement :

$$Recall = 0.969586$$

Conclusion

La prédiction de l'attrition ou la tâche de reconnaître les clients susceptibles d'interrompre l'utilisation du service est une question lucrative et essentielle du secteur des télécommunications. La perte de clientèle est souvent un problème critique pour le secteur des télécommunications car les clients ne tardent pas à partir s'ils ne prévoient pas ce qu'ils vont voir. Les clients ont surtout besoin d'un bon rapport qualité-prix, d'un coût compétitif et d'une meilleure qualité de service. La prévision en temps utile des clients fidèles qui ont l'intention de quitter l'entreprise peut aider à les identifier et à les soumettre à une action proactive afin de les conserver.

Bibliographie

- [1] ALAIN BACCINI et PHILIPPE BESSE *Data mining I Exploration Statistique* Version septembre 2005.
- [2] Bastide.S *Contribution and use of Bayesian inference methods in the field of clinical diagnostic studies* Theses, Université Montpellier (2016) .
- [3] Coussement K. and Poel D. V. d. *Churn prediction in subscription services : An application of support vector machines while comparing two parameter selection techniques.* Expert Systems with Applications, 2008.
- [4] ERIC DAVALO et PATRICK NAIM *DES RESEAUX DE NEURONES* .
- [5] Eric Davalo, Patrick Naïm *Neural Networks (1991, Macmillan Education UK)* [Macmillan Computer Science Series]
- [6] E. A. Feigenbaum and J. Feldman. (Edirors). *Computers and thought.* McGraw-Hill Inc. pp.192. New York, 1963.
- [7] Guozheng Zhang¹ and Yun Chen² *An intergrated data mining and survival analysis model for customer segmentation.*
- [8] Han S.H., Lu S. X., and Leung S. C. H *Segmentation of telecom customers basedon customer value by decision tree model.* Expert Systems with Applications, 2012.
- [9] J. R. Slagle. McGraw-Hill. *Artificial intelligence : The heuristic programming approach.* pp. 3. New York, 1971.

BIBLIOGRAPHIE

- [10] Oghojafor B. E. A., Mesike G. C., Omoera C. I., and Bakare R. D. *Modelling telecom customer attrition using logistic regression. African Journal of Marketing Management, 2012.*
- [11] Robert, C. P. (2007). *The Bayesian choice : from decision-theoretic foundations to computational implementation.* Springer texts in statistics. Springer, 2nd ed edition.
- [12] R. L. Solso. *Cognitive psychology.* Harcourt Brace Jovanovich, Inc., New York. pp. 436, 1979.
- [13] S. Voss, S. Martello, I. H. Osman, and C. Roucairol, *Meta-Heuristics – Advances and Trends in Local Search Paradigms for Optimization.* Kluwer Academic Publishers, The Netherlands, 1999.
- [14] Tsai C. F and Lu Y. H. *Customer churn prediction by hybrid neural network* Expert Systems with Applications, 2009.
- [15] (Wiley Series in Computational Statistics) *Data Mining and Statistics for Decision Making*-Wiley (2011)
- [16] . (1995), *A distributed parallel genetic algorithm for solving optimal growth models*, Computational Economics 8(3), 159–180.
- [17] *Apprentissage Statistique et Data mining* Version Octobre 2006.
- [18] *A Tutorial with Applications to Production and Logistics* Metaheuristic Search Concepts Berlin Heidelberg (2010)
- [19] *Modélisation, classification et commande par réseaux de neurones : principes fondamentaux, méthodologie de conception et illustrations industrielles* Paris.1995.
- [20] *Présentation des algorithmes génétiques et de leurs applications en économie* 7 septembre 2001.