

République Algérienne Démocratique et Populaire
Ministère de l'enseignement Supérieur et de la Recherche Scientifique
Université de Mohamed El Bachir El Ibrahimi de Bordj Bou Arréridj
Faculté des Mathématiques et d'Informatique
Département d'informatique



MEMOIRE

Présenté en vue de l'obtention du diplôme

Master en informatique

Spécialité : Ajoutez votre spécialité ici

THEME

Le thème de ce mémoire ICI

Présenté par :

NOM ET PRENOM DE L'ETUDIANT 1

NOM ET PRENOM DE L'ETUDIANT 2

Soutenu publiquement le : jj/mm/aaaa

Devant le jury composé de :

Président :

Examineur :

Encadreur :

2022/2023

Dédicace

Écrivez votre dédicace ici
Une courte dédicace est préférable

Remerciement

Gardez-le aussi court et direct que possible.

Résumé

Écrivez votre résumé en français ici. Présentation en quelques lignes du contenu du rapport. Visez 1 paragraphe, mais ne pas dépasser 2 paragraphes. Présentez l'objectif du projet, les résultats obtenus et leur importance.

Abstract

Write your abstract in English here.

ملخص

اكتب ملخصك باللغة العربية هنا

Table des matières

Liste des abréviations	x
Liste des figures	xi
Liste des tableaux	xiii
Liste des Algorithmes	xiv
1 Introduction Générale	1
1.1 Introduction	1
1.2 Diagnostic médical	1
1.2.1 Historique diagnostic	2
1.2.2 Définitions diagnostic	2
1.2.3 Les étapes de diagnostic	3
1.3 Symptômes médicaux	3
1.3.1 Définitions les symptômes	4
1.3.2 Exemples de symptômes	4
1.4 Pronostic médical	4
1.4.1 Définition Pronostic médical	5
1.4.2 Les type Pronostic médical	5
1.5 La relation entre diagnostic, les symptômes et Pronostic médical	5
1.6 Conclusion	5
2 Apprentissage Automatique	6
2.1 Introduction	6
2.2 L'apprentissage Automatique	6

2.3	Types d'Apprentissage Automatique	7
2.3.1	Apprentissage Supervisé	7
2.3.2	Apprentissage Non-Supervisé	8
2.4	Définition de classification	8
2.5	Les types de classification	8
2.5.1	La classification binaire (Binary classification)	9
2.5.2	La classification multi-classe	9
2.5.3	La classification multi-label	10
2.6	Techniques de Classification	10
2.6.1	Les k plus proches voisins (KNN)	10
2.6.2	k-means	11
2.6.3	Régression Logistique	12
2.6.4	Machine à Vecteurs de Supports (S V M)	13
2.6.5	Apprentissage profondeur	13
2.7	Comparaison entre techniques de classification	15
2.8	Conclusion	16
3	Deep Learning	17
3.1	Introduction	17
3.2	L'apprentissage profond	17
3.2.1	Définition l'apprentissage profond	17
3.2.2	Domaines d'application de l'apprentissage profonde	18
3.3	Principes de fonctionnement	19
3.3.1	Les types des couches dans d'apprentissage profond	19
3.4	Les poids	20
3.5	La fonction d'activation	20
3.5.1	La fonction Relu	20
3.5.2	La fonction d'un softmax	21
3.5.3	La fonction sigmoïde	22
3.6	Les différents types de model deep Learning	23
3.6.1	Le réseau de neurone convolutif (CNN)	23
3.6.2	Réseau de neurones récurrents (RNN)	24
3.6.3	Les réseaux LSTM	25

3.6.4	Réseaux de neurones artificiel (ANN)	26
3.7	Conclusion	27
4	Méthodologie	28
4.1	Introduction	28
4.2	Notre Objectif	28
4.3	Prétraitement	30
4.4	Représentation des Données	32
4.4.1	La Représentation en Vecteur de Caractéristiques Binaires	32
4.5	Méthodologie de Travail	32
4.6	Evaluation des Modèles de Classification	34
4.6.1	Rappel	34
4.6.2	Précision	34
4.6.3	Exactitude	34
4.6.4	Score f1	35
4.6.5	Temps d'entraînement	35
4.7	Optimisation des modèles de classification	35
4.7.1	Adam (Estimation de moment adaptatif)	35
4.7.2	SGD (Descente de gradient stochastique)	36
4.8	Pertes des modèles de classification	37
4.8.1	Entropie croisée catégorielle éparées	37
4.9	Conclusion	39
A	Titre de l'annexe ici	40
1.1		40
1.1.1		40
B	Titre de l'annexe ici	41
2.1		41
2.1.1		41

(Cette liste est optionnelle, voici un exemple)

Table des figures

2.1	Les différentes méthodes d'apprentissage automatique.	7
2.2	Exemple de problème de classification binaire.	9
2.3	Exemple de problème de classification binaire , multi-classe et multi-label . . .	10
2.4	Pour $k = 3$ la classe majoritaire du point central est la classe B, mais si on change la valeur du voisinage $k = 6$ la classe majoritaire devient la classe A . . .	11
2.5	L'algorithme k-means regroupe les données en k cluster, ici $k = 3$. Les centres de gravité sont représentés par de petit cercle.	11
2.6	Modèle la régression logistique.	12
2.7	Un diagramme d'un hyperplan avec des vecteurs de support dans un espace vectoriel à deux dimensions.	13
2.8	Architecture Apprentissage profondeur.	14
3.1	La relation entre l'intelligence artificielle, le Machine Learning et le deep learning.	18
3.2	les couches d'apprentissage profond.	19
3.3	Fonction d'activation.	20
3.4	La fonction Relu.	21
3.5	La fonction d'un softmax.	22
3.6	La fonction sigmoïde.	22
3.7	Structure générale d'un réseau CNN.	23
3.8	Architecture de RNN.	24
3.9	Le module répétitif dans un LSTM.	25
3.10	Architecture de réseau de neurones artificiels.	26
4.1	Tux, le pingouin	29

4.2	Tux, le pingouin	33
-----	----------------------------	----

Liste des tableaux

4.1	Table caption	34
-----	-------------------------	----

List of Algorithms

Chapitre 1

Introduction Générale

Chaque rapport doit commencer par une introduction générale dans laquelle le contexte du projet est clairement expliqué. Cette introduction devrait également inclure l'objectif du projet et le plan du reste du rapport. Cette introduction ne devrait pas dépasser 2 pages. Soyez concis et clair, et écrivez uniquement ce qui est nécessaire à écrire.

1.1 Introduction

L'évaluation médicale est un processus fondamental dans la pratique clinique, visant à évaluer l'état de santé d'un patient et à déterminer les meilleures stratégies de prise en charge. Dans ce contexte, l'Évaluation Médicale Compréhensive (EMC) émerge comme une approche intégrative, cherchant à appréhender la globalité des besoins du patient au-delà de la simple symptomatologie.

1.2 Diagnostic médical

Le diagnostic médical, fruit de la science et de l'expertise, il éclaire le chemin vers la compréhension des symptômes, guidant ainsi le parcours vers la guérison .

1.2.1 Historique diagnostic

A ses origines, la médecine était essentiellement magico-religieuse, le diagnostic appartenait aux devins, aux oracles et aux prêtres. Avec Hippocrate, la conception de la médecine devint plus rationnelle et le diagnostic (encore volontiers confondu avec le pronostic) fondé sur un examen clinique minutieux. Au XVII^e et au XVIII^e siècle, la démonstration de la circulation du sang par Harvey introduisit la médecine dans le monde de la mécanique ; Morgagni fonda l'anatomopathologie et la méthode anatomoclinique prit naissance. Son essor se fit dans le cadre du développement de la sémiologie, lui-même lié à la découverte de la percussion à la fin du XVIII^e siècle (Auenbrugger) et à celle de l'auscultation (Laennec) au début du XIX^e siècle. Au cours de ce même siècle elle atteignit sa plénitude avec l'apparition de l'anatomopathologie microscopique (Broca). A la méthode anatomoclinique vint s'adjoindre avec Claude Bernard la méthode physio clinique introduisant la physiopathologie et la biologie. Le XIX^e siècle vit les débuts de la bactériologie, de l'immunologie et de l'imagerie médicale. Le XX^e siècle fut marqué par le développement de l'endoscopie et des prélèvements biopsiques rendus nécessaires aussi par l'essor de la chirurgie, l'artériographie précédant les méthodes d'explorations non traumatisantes : explorations isotopiques, tomодensitométrie et résonance magnétique nucléaire, l'usage de l'échographie et de l'effet doppler.

La biochimie, la génétique permirent progressivement de substituer à l'étalon or de la confrontation anatomoclinique des critères plus subtils, le passage de l'échelon microscopique au stade de l'échelon moléculaire modifiant progressivement mais radicalement nos conceptions du diagnostic en médecine. Ainsi la médecine des deux derniers siècles a vu se confirmer la rationalité de la démarche médicale puis son développement scientifique, des lésions tissulaires on est passé à la biologie moléculaire.

1.2.2 Définitions diagnostic

Le diagnostic est le processus d'évaluation d'un état de fonctionnement donné. Si cet état est comparé avec un état de référence, il s'agit d'évaluation de dérive de fonctionnement.

1.2.3 Les étapes de diagnostic

- **Collecte d'informations** : Recueillir des données pertinentes liées au problème ou à la situation, que ce soit des symptômes, des données techniques, des antécédents, etc.
- **Identification du problème** : Analyser les informations collectées pour déterminer la nature du problème. Cela peut impliquer la comparaison des données avec des normes ou des critères établis.
- **Élaboration d'hypothèses** : Formuler des hypothèses sur les causes possibles du problème en se basant sur les informations disponibles.
- **Tests et investigations** : Mettre en place des tests ou des investigations pour valider ou invalider les hypothèses formulées. Cela peut inclure des examens médicaux, des tests techniques, des simulations, etc.
- **Analyse des résultats** : Examiner les résultats des tests et des investigations afin de confirmer la cause du problème ou de revoir les hypothèses si nécessaire.
- **Établissement du diagnostic** : Formuler un diagnostic final en identifiant la cause principale du problème ou de la situation, en tenant compte de toutes les informations et des résultats obtenus.
- **Proposition de solutions** : Proposer des solutions ou des recommandations pour résoudre le problème diagnostiqué. Cela peut impliquer des traitements médicaux, des ajustements techniques, des interventions psychologiques, etc.
- **Suivi et évaluation** : Mettre en place un suivi pour évaluer l'efficacité des solutions proposées et ajuster si nécessaire. Cela peut également inclure des mesures préventives pour éviter la récurrence du problème .

1.3 Symptômes médicaux

Les symptômes médicaux sont souvent les signaux précurseurs d'un état de santé sous-jacent, nécessitant une évaluation médicale approfondie pour établir un diagnostic précis et élaborer un plan de traitement adapté.

1.3.1 Définitions les symptômes

Les symptômes se réfèrent à des signes ou manifestations d'une maladie, d'un trouble ou d'une condition médicale. Ce sont des indicateurs observables ou ressentis par le patient, ainsi que détectés par les professionnels de la santé. Les symptômes médicaux peuvent varier en fonction de la maladie ou du trouble et Il est important de noter que les symptômes ne sont pas la maladie elle-même, mais plutôt des signaux qui indiquent la présence possible d'un problème de santé. Ils peuvent être utilisés pour aider les professionnels de la santé à poser un diagnostic et à recommander un plan de traitement approprié.

1.3.2 Exemples de symptômes

- La fatigue est définie comme une sensation d'épuisement survenant durant ou après une activité habituelle. Une cause organique ou psychiatrique est retrouvée dans la majorité des cas.
- Le vomissement est le rejet du contenu de l'estomac par la bouche. Il correspond à un réflexe mécanique de défense de l'organisme destiné à vider l'estomac. Il est possible de vomir des aliments, de la bile ou beaucoup plus rarement, du sang.
- La fièvre est une température corporelle anormalement élevée, qui dépasse 38°C.
- La toux est l'expiration brusque et sonore de l'air contenu dans les poumons provoquée par une irritation des voies respiratoires.
- La douleur thoracique désigne toute douleur ou toute sensation anormale et pénible localisée dans la zone du thorax.
- La perte d'appétit est la perte de l'envie de manger, peut aussi être appelée anorexie.
- Les céphalées sont un problème très courant. Il existe plusieurs types de céphalées, les céphalées de tension étant les plus fréquentes. Bien qu'en général bénignes, les céphalées peuvent être le symptôme d'une maladie grave.

1.4 Pronostic médical

Le pronostic médical, étroitement lié aux symptômes médicaux présentés, offre une perspective éclairante sur l'évolution potentielle de la condition de santé, permettant aux professionnels de la santé de prendre des décisions informées pour un plan de traitement optimal.

1.4.1 Définition Pronostic médical

Le pronostic médical est une évaluation anticipée de l'évolution probable d'une maladie chez un patient, basée sur des données cliniques et des connaissances médicales, afin de prédire les perspectives de guérison, de rémission, de stabilisation ou de progression de la condition.

1.4.2 Les type Pronostic médical

- **Pronostic vital** : Il concerne la probabilité de survie d'un patient, souvent exprimée en termes de taux de survie à un certain nombre d'années.
- **Pronostic qualité de vie** : Il prend en compte l'impact de la maladie ou du traitement sur la qualité de vie globale du patient.
- **Pronostic génétique** : Il se base sur des facteurs génétiques et évalue la probabilité de développement de maladies héréditaires.
- **Pronostic fonctionnel** : Il évalue la capacité du patient à maintenir une fonction normale ou à retrouver une fonction normale après un traitement.

1.5 La relation entre diagnostic, les symptômes et Pronostic médical

La relation entre le diagnostic, les symptômes et le pronostic médical est cruciale, les symptômes fournissant des indices précieux pour orienter le processus diagnostique, tandis que le diagnostic éclaire le pronostic en permettant une évaluation anticipée de l'évolution probable de la maladie, guider ainsi les choix de traitement et les décisions médicales.

1.6 Conclusion

Cette étude souligne l'interdépendance cruciale entre le diagnostic, les symptômes et le pronostic médical dans le contexte de la médecine moderne. Une compréhension approfondie de ces éléments est essentielle pour assurer des soins de qualité et des résultats positifs pour les patients. Le chapitre suivant présente l'apprentissage automatique afin de pouvoir analyser les données médicales.

Chapitre 2

Apprentissage Automatique

2.1 Introduction

L'apprentissage automatique, en anglais est l'appelé "machine Learning" . Représente une discipline passionnante au cœur de l'intelligence artificielle. Ce domaine cherche à développer des modèles informatiques capables d'apprendre à partir de l'expérience et d'améliorer leurs performances au fil du temps, sans être explicitement programmés.

2.2 L'apprentissage Automatique

L'apprentissage automatique est un type d'intelligence artificielle (IA), c'est une science qui permet aux ordinateurs d'apprendre sans être explicitement programmés « Arthur Samuel, 1959 ». Plus précisément, l'apprentissage automatique fait référence au développement, l'analyse et l'implémentation de méthodes qui permettent à une machine (au sens large) d'évoluer et de remplir des tâches associées à une intelligence artificielle grâce à un processus d'apprentissage. Cet apprentissage permet d'avoir un système qui s'optimise en fonction de l'environnement, les expériences et les résultats observés. Dans le domaine médicale, l'apprentissage automatique a été conçu pour réaliser l'analyse de données médicales, surtout lorsque l'évolution numérique a fourni des moyens (capteurs) peu coûteux permettant de recueillir et de stocker des informations importantes liées aux patients et maladies. Par exemple, les algorithmes d'apprentissage sont utiles au médecin lors du diagnostic des patients, afin d'améliorer la vitesse, la précision et la fiabilité de son diagnostic.

2.3 Types d'Apprentissage Automatique

Il existe fondamentalement quatre types d'apprentissage automatique : Supervisé, semi-supervisé et non-supervisé et Apprentissage par renforcement. Dans notre étude, nous utilisons l'apprentissage supervisé pour construire des modèles pour la prédiction des maladies. Dans la suite de cette section, nous allons présenter les deux types d'apprentissage les plus utilisés qui sont l'apprentissage supervisé et apprentissage non supervisé.

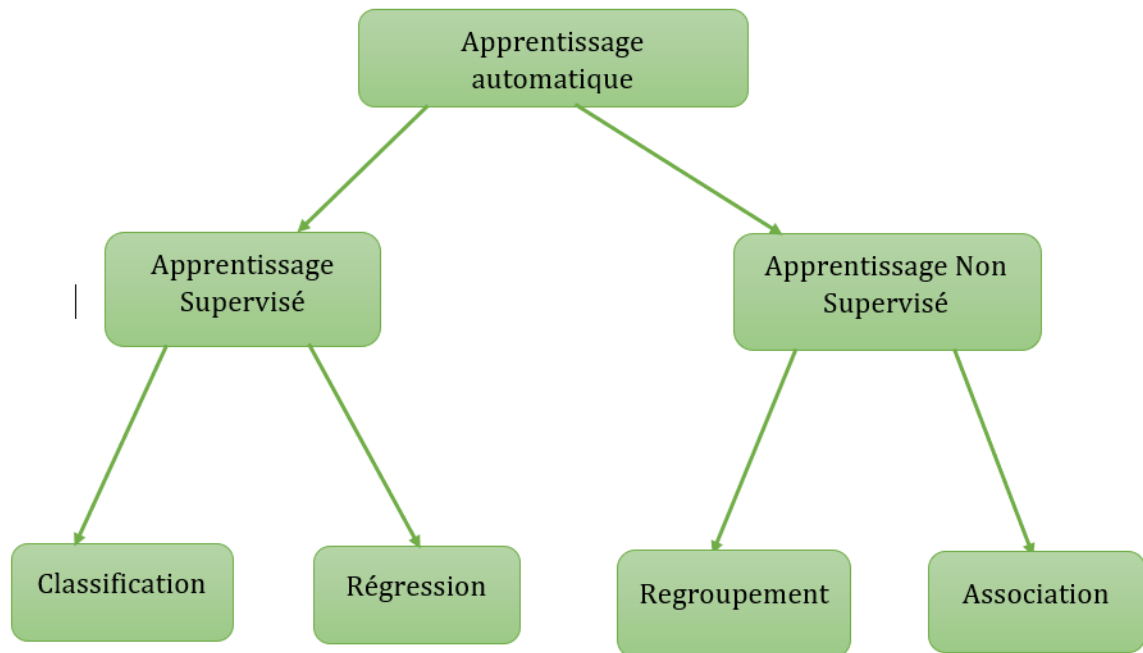


FIGURE 2.1 – Les différentes méthodes d'apprentissage automatique.

2.3.1 Apprentissage Supervisé

Dans l'apprentissage supervisé, l'ordinateur est fourni avec des exemples d'entrées qui sont étiquetés avec les sorties souhaitées. Le but de cette méthode est que l'algorithme puisse « apprendre » en comparant sa sortie réelle avec les sorties « apprises » pour trouver des erreurs et modifier le modèle en conséquence. L'objectif des algorithmes d'apprentissage supervisé est d'apprendre une fonction qui mappe les vecteurs de caractéristiques (entrées) aux étiquettes (sortie), sur la base d'exemples de paires entrée-sortie.

Comme il est illustré dans la Figure 1 l'apprentissage supervisé peut être utilisé pour deux types de tâches principales : la classification et la régression. Les algorithmes de classification cherchent à prédire la classe ou la catégorie à laquelle appartient une donnée d'entrée tandis

que les algorithmes de régression servent à prédire une valeur numérique continue à partir de variables d'entrée. Dans ce travail, nous nous intéressons aux méthodes de classification.

2.3.2 Apprentissage Non-Supervisé

Dans l'apprentissage non supervisé, les données sont non étiquetées, de sorte que l'algorithme d'apprentissage trouve tout seul des points communs parmi ses données d'entrée. Les données non étiquetées étant plus abondantes que les données étiquetées, les méthodes d'apprentissage automatique qui facilitent l'apprentissage non supervisé sont particulièrement utiles. L'objectif de l'apprentissage non supervisé peut être aussi simple que de découvrir des modèles cachés dans un ensemble de données. Les plus fréquents problèmes connus dans ce type sont :

- Le clustering qui consiste à regrouper un ensemble d'éléments hétérogènes sous forme de sous-groupes homogènes.
- La réduction de dimension qui consiste à prendre des données dans un espace de grande dimension, et à les remplacer par des données dans un espace de plus petite dimension sans perdre la variance .

2.4 Définition de classification

Un modèle de classification est un modèle d'apprentissage automatique qui attribue des étiquettes ou des catégories à des instances d'entrée en les associant à un ensemble fini de classes prédéfinies. L'objectif est de créer une fonction qui peut généraliser à de nouvelles données et assigner correctement des classes à des observations inconnues en fonction des modèles appris à partir des données d'entraînement. Les classes peuvent être binaires (deux classes, comme oui/non) ou multinomiales (plus de deux classes), et le modèle cherche à discriminer entre ces classes en fonction des caractéristiques des données d'entrée.

2.5 Les types de classification

Dans cette partie, nous présentons les types de classification :

2.5.1 La classification binaire (Binary classification)

La classification binaire (ou la classification binomiale) est une transformation de données qui vise à répartir les membres d'un ensemble dans deux groupes disjoints selon que l'élément possède ou non une propriété / fonctionnalité donnée, mais peut également être interprété comme vrai et faux, 1 et 0, ou toute autre combinaison de deux valeurs. Par exemple : tests médicaux visant à déterminer si un patient est atteint d'une certaine maladie ou non.

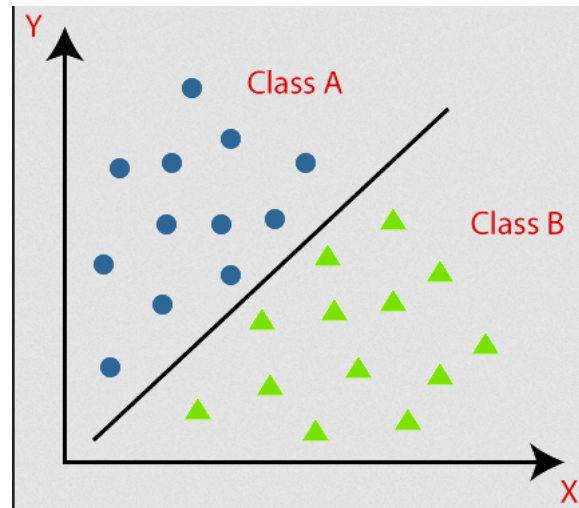


FIGURE 2.2 – Exemple de problème de classification binaire.

- L'image montre qu'il y a deux classes : classe A et classe B.

2.5.2 La classification multi-classe

La classification multi-classe désigne une tâche de classification comportant plus de deux classes, par exemple, La classification des visages, classification des espèces végétales...Ect. Un jeu de données multi-classe n'a qu'une seule classe en sortie, comme dans la classification binaire.

2.5.3 La classification multi-lable

Jusqu'à présent, chaque instance était toujours affectée à une seule classe. Dans certains cas, le classificateur peut produire plusieurs classes pour chaque instance. Par exemple, un classificateur pour reconnaître des types de maladies : que faire s'il reconnaît plusieurs symptômes en même temps ? Bien entendu, il doit noter chaque symptôme associé à une maladie spécifique. Supposons qu'un classificateur ait été formé pour reconnaître trois symptômes : « fatigue, nausée et température élevée ». Ainsi, lorsqu'il apparaît « fatigué et une forte température ». Il devrait produire $[1, 0, 1]$ (c'est-à-dire : fatigué oui, nausée non, fièvre oui.) Un système de classification qui produit plusieurs binaires est appelé système de classification multi-étiquettes.

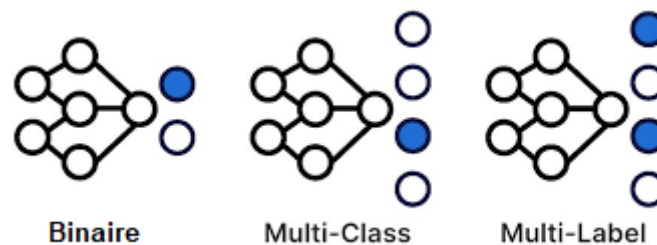


FIGURE 2.3 – Exemple de problème de classification binaire , multi-classe et multi-lable .

2.6 Techniques de Classification

Dans cette partie, nous présentons les algorithmes de classification :

2.6.1 Les k plus proches voisins (KNN)

L'algorithme des k plus proches voisins en anglais est l'appelé « K-Nearest Neighbors (KNN) » est un algorithme de classification supervisé. Chaque observation de l'ensemble d'apprentissage est représentée par un point dans un espace à n dimensions ou n est le nombre de variables prédictives. Pour prédire la classe d'une observation, on cherche les k points les plus proches de cet exemple. La classe de la variable cible, est celle qui est la plus représentée parmi les k plus proches voisins. Il existe des variantes de l'algorithme ou on pondère les k observations en fonction de leur distance à l'exemple dont on veut classer les observations les plus éloignées de notre exemple seront considérées comme moins importantes.

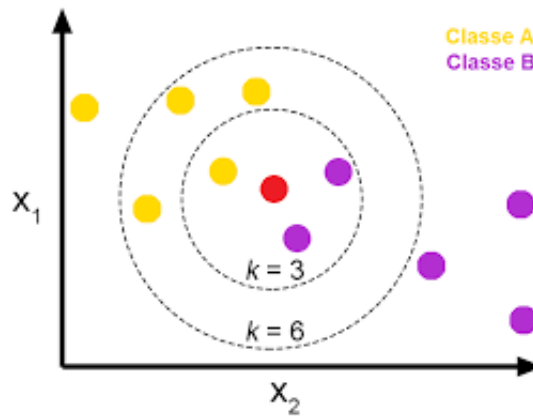


FIGURE 2.4 – Pour $k = 3$ la classe majoritaire du point central est la classe B, mais si on change la valeur du voisinage $k = 6$ la classe majoritaire devient la classe A .

2.6.2 k-means

L'algorithme des k-moyennes en anglais est l'appelé « k-means » est un algorithme non supervisé. Chaque observation est représentée par un point dans un espace à n dimensions ou n est le nombre de variables descriptives. À partir d'un ensemble d'apprentissage de M observations $[X^1, \dots, X^M]$ cet algorithme va repartir ces observations en k clusters de manière à ce que la distance euclidienne qui sépare les points au centre de gravité du groupe auquel ils sont affectés soit minimale. Les étapes de l'algorithme sont :

- Choisir k points qui représentent la position moyenne des clusters.
- répéter jusqu'à stabilisation des points centraux :
 - affecter chacun des M points au plus proche des k points centraux.
 - mettre à jour les points centraux en calculant les centres de gravité des k cluster.

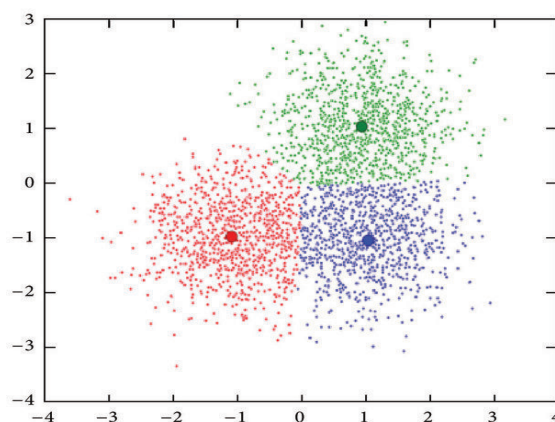


FIGURE 2.5 – L'algorithme k-means regroupe les données en k cluster, ici $k = 3$. Les centres de gravité sont représentés par de petit cercle.

2.6.3 Régression Logistique

L'analyse de régression est souvent utilisée pour faire des prédictions, comprendre les variables indépendantes par rapport à la variable dépendante et étudier la forme de leur relation. Dans des circonstances limitées, l'analyse de régression peut être utilisée pour déduire la relation causale entre la variable indépendante et la variable dépendante. La régression est un algorithme robuste lorsqu'il s'agit de classer des ensembles de problèmes, et a une fonction logistique (fonction sigmoïde) au cœur de celui-ci. Dans cet algorithme, les valeurs d'entrée sont combinées en fonction de coefficients ou de poids pour donner les valeurs de sortie/prédites.

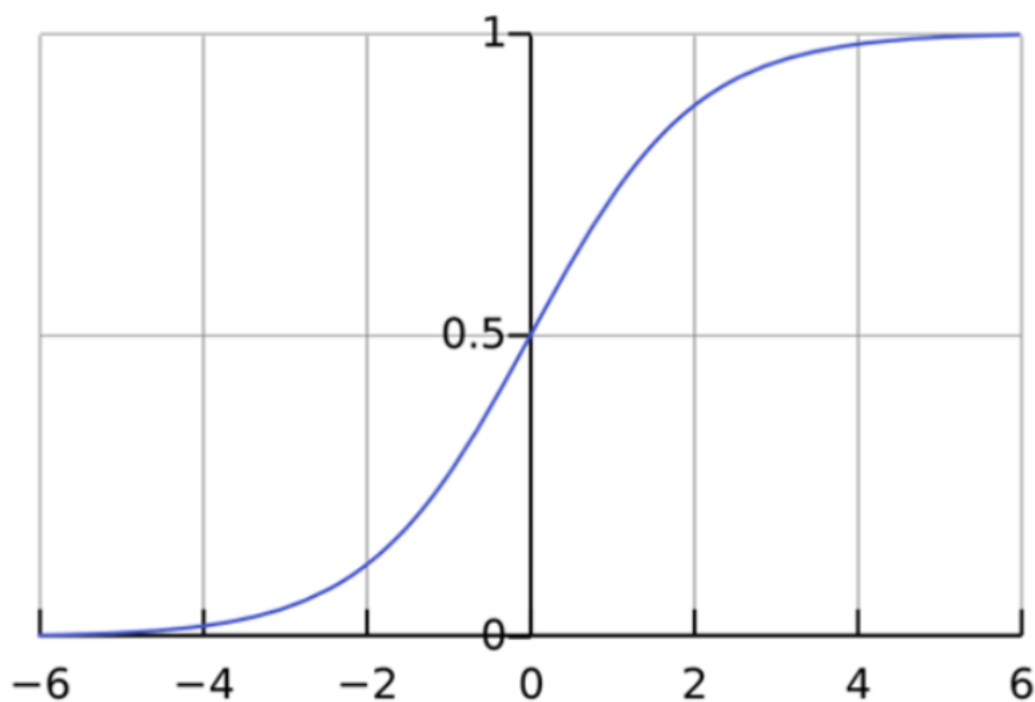


FIGURE 2.6 – Modèle la régression logistique.

2.6.4 Machine à Vecteurs de Supports (S V M)

Machine à Vecteurs de Supports en anglais cela l'appelé « Support Vector Machine » est l'un des algorithmes d'apprentissage supervisé les plus populaires, utilisé pour les problèmes de classification et de régression. Cependant, il est principalement utilisé pour les problèmes de classification dans l'apprentissage automatique. Le but de l'algorithme SVM est de créer la meilleure ligne ou limite de décision qui peut séparer l'espace à n dimensions en classes afin que nous puissions facilement mettre le nouveau point de données dans la bonne classe à l'avenir.

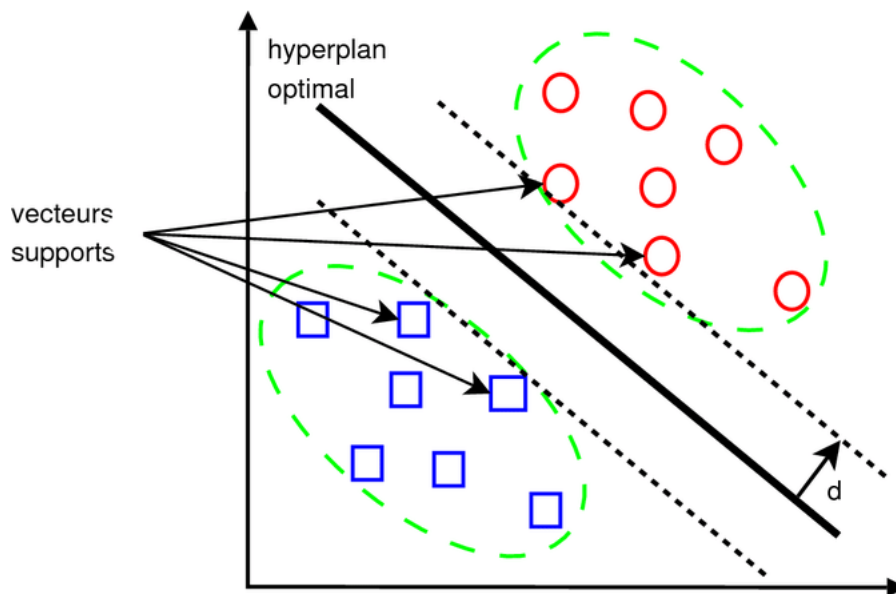


FIGURE 2.7 – Un diagramme d'un hyperplan avec des vecteurs de support dans un espace vectoriel à deux dimensions.

- La figure 2.7 montre un exemple simple d'un hyperplan avec des vecteurs de support. Cette technique peut être utilisée pour séparer deux classes de points dans un espace vectoriel, ce qui est utile pour des tâches d'apprentissage automatique telles que la classification et la régression. L'objectif d'un hyperplan avec des vecteurs de support est de trouver la meilleure séparation possible entre deux classes de points dans un espace vectoriel. Cela peut être utilisé pour des tâches de classification, telles que la reconnaissance d'image ou le traitement du langage naturel.

2.6.5 Apprentissage profondeur

Apprentissage profond en anglais est l'appelé « Deep Learning » est un type d'intelligence artificielle dérivé du machine Learning (apprentissage automatique) où la machine est capable

d'apprendre par elle-même, contrairement à la programmation où elle se contente d'exécuter à la lettre des règles prédéterminées.

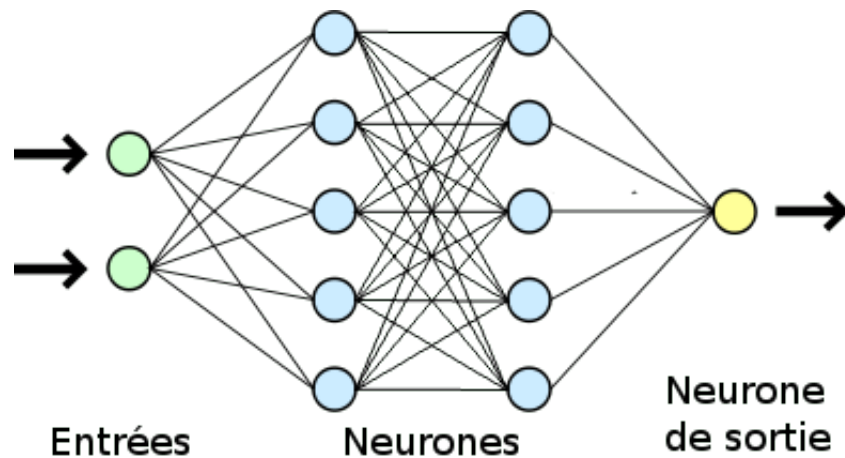


FIGURE 2.8 – Architecture Apprentissage profondeur.

La figure 2.8 fournie montre une architecture d'apprentissage profond composée de plusieurs couches empilées les unes sur les autres. Chaque couche est composée d'un certain nombre de neurones artificiels qui sont interconnectés. Les neurones sont responsables de traiter les informations et de les transmettre à la couche suivante.

- **Couches d'entrée** : Elles reçoivent les données brutes, comme des images ou du texte.
- **Couches cachées** : Elles traitent les données et extraient des caractéristiques de plus en plus complexes.
- **Couche de sortie** : Elle produit la sortie finale, comme une classification ou une prédiction.

2.7 Comparaison entre techniques de classification

Techniques de Classification	Avantages	Inconvénients
KNN	- Simple à concevoir	<ul style="list-style-type: none"> - Sensible aux bruits. - Pour un nombre de variable prédictives très grands, le calcul de la distance devient très coûteux.
k-means	-implémentable pour des grands volumes de données	<ul style="list-style-type: none"> - Le choix du paramètre k n'est pas découvert mais choisi par l'utilisateur . - La solution dépend des k centre de gravité choisi lors de l'initialisation
Régression Logistique	- Ses résultats sont faciles à interpréter.	<ul style="list-style-type: none"> - La phase d'apprentissage peut être longue car l'optimisation des coefficients est complexe. - Sa linéarité empêche la prise en compte des interactions entre les variables.
S V M	<ul style="list-style-type: none"> - Il permet de traiter des problèmes de classification non linéaire complexe. - Les SVM constituent une alternative aux réseaux de neurones car plus faciles à entraîner. 	<ul style="list-style-type: none"> - Les SVM peuvent être lents à entraîner sur de grands ensembles de données. - Cela peut affecter la précision du modèle et le rendre moins fiable.
Apprentissage profondeur	<ul style="list-style-type: none"> - Adaptabilité à Divers Domaines. - Performances Exceptionnelles dans des Tâches Complexes. 	<ul style="list-style-type: none"> - Interprétabilité Limitée. - Besoin d'Expertise Technique Élevée.

2.8 Conclusion

Dans ce chapitre, nous avons présenté les algorithmes d'apprentissage supervisé. Après avoir présenté les deux principaux types d'apprentissage automatique, une description détaillée de chaque méthode de classification a été fournie, en expliquant le principe de fonctionnement de chaque méthode ainsi que ses avantages et inconvénients. Le chapitre suivant présente les méthodes d'apprentissage profondeur qui ont été appliquées dans le domaine de la santé.

Chapitre 3

Deep Learning

3.1 Introduction

L'apprentissage profond ou le deep Learning est un nouveau domaine de recherche du Machine Learning(ML), qui a été introduit dans le but de rapprocher le ML de son objectif principal : l'intelligence artificielle. Il concerne les algorithmes inspirés par la structure et le fonctionnement du cerveau. Ils peuvent apprendre plusieurs niveaux de représentation dans le but de modéliser des relations complexes entre les données.

3.2 L'apprentissage profond

L'apprentissage profond est une technologie d'intelligence artificielle révolutionnaire qui permet aux machines d'apprendre à partir de données massives, ouvrant ainsi la voie à des avancées majeures. Malgré ses succès, il présente des défis, mais continue de repousser les frontières de l'intelligence artificielle.

3.2.1 Définition l'apprentissage profond

L'apprentissage profond en anglais est appelée «Deep Learning » une technique d'apprentissage automatique révolutionnaire qui a permis de réaliser des progrès importants dans de nombreux domaines tels que la vision par ordinateur, la reconnaissance de la parole et la traduction automatique. Grâce à sa capacité à apprendre par elle-même et à résoudre des problèmes

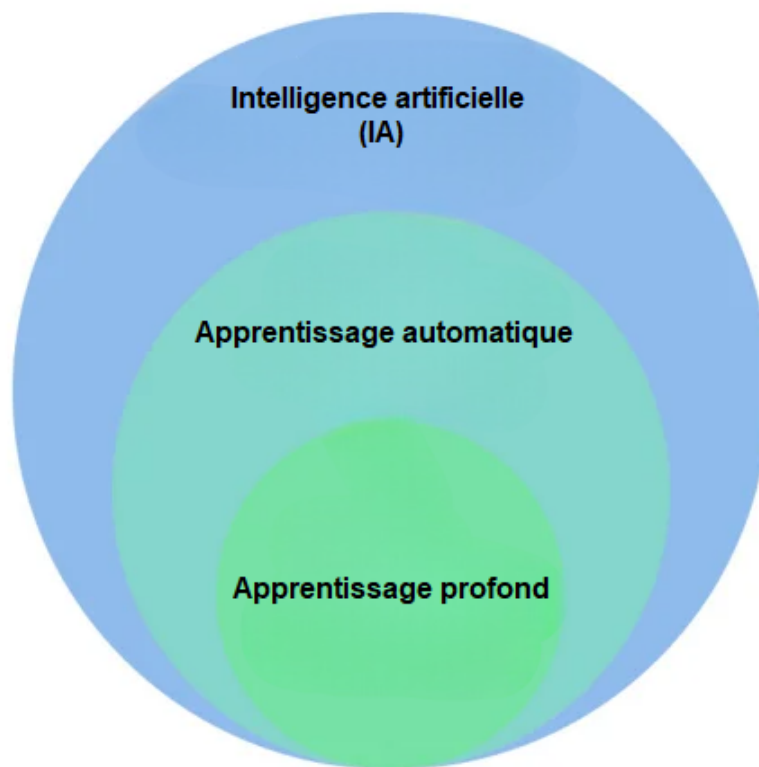


FIGURE 3.1 – La relation entre l'intelligence artificielle, le Machine Learning et le deep learning.

complexes, l'apprentissage profond est un outil puissant qui continuera à jouer un rôle majeur dans l'évolution de l'intelligence artificielle.

3.2.2 Domaines d'application de l'apprentissage profonde

Le deep Learning utilisé dans plusieurs domaines différents tels que :

- **Traitement du langage naturel (NLP)** : L'apprentissage profond est employé dans la compréhension du langage naturel, la traduction automatique, la génération de texte, la classification de texte, et l'analyse des sentiments.
- **Santé** : L'apprentissage profond est appliqué dans le diagnostic médical, la segmentation d'images médicales, la prédiction de maladies, et la découverte de médicaments.
- **Vision par ordinateur** : L'apprentissage profond est largement utilisé pour la reconnaissance d'images, la segmentation d'images, la détection d'objets, la classification d'images, et la génération d'images.
- **Robotique** : L'apprentissage profond est employé dans la perception sensorielle des

robots, la planification de mouvements, et l'interaction homme-robot.

3.3 Principes de fonctionnement

D'apprentissage en profondeur sont formés sur la base des structures de données complexes qu'ils rencontrent. Ils construisent des modèles informatiques composés de plusieurs couches (couche d'entrée, couche cachée, couche de sortie) de traitement pour créer plusieurs niveaux d'abstraction pour représenter les données.

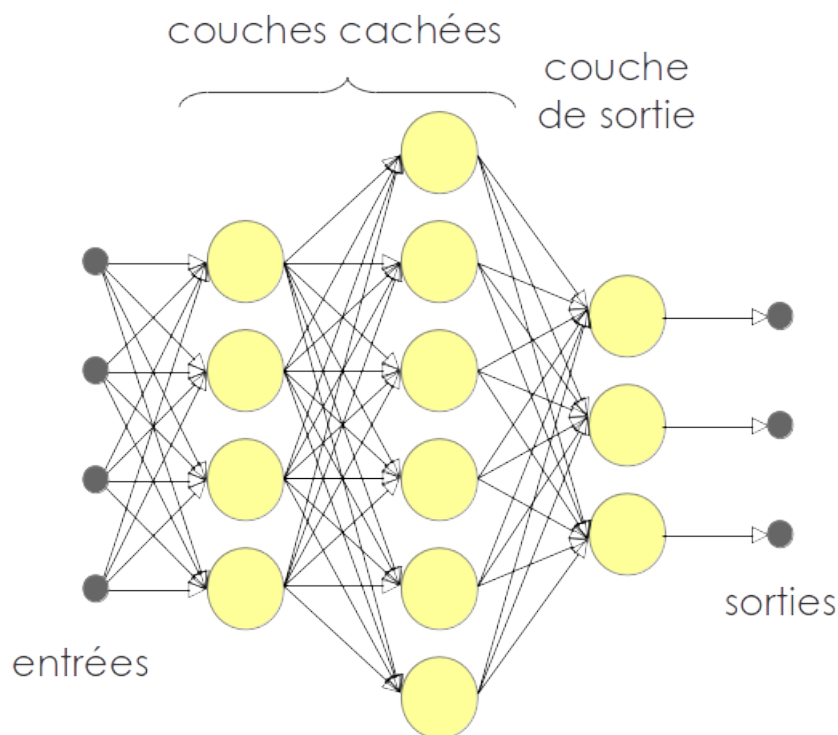


FIGURE 3.2 – les couches d'apprentissage profond.

3.3.1 Les types des couches dans d'apprentissage profond

- **Couche d'entrée** : composée d'un ensemble de neurones qui favoriseront la propagation des informations dans le réseau neuronal. Elle comprend un nombre de neurones généralement égal au nombre de caractéristiques constituant l'enregistrement en entrée (p. ex. une image, une transaction, etc.).
- **Couches intermédiaires** : servant à traiter l'information propagée dans le réseau de neurones pour capturer les caractéristiques de son apprentissage.

- **Couche de sortie** : formée d'un ensemble de neurones représentant les différentes classes de résultat. Dans une problématique de classification, les classes sont les différentes possibilités que le résultat peut offrir.

3.4 Les poids

Les poids des flèches sont utilisés pour accorder de l'importance à certaines fonctionnalités par rapport à d'autres, afin d'obtenir les résultats souhaités. La somme pondérée de tous les poids des flèches est calculée pour chaque neurone d'une couche cachée, et chacun de ces neurones exécute une fonction d'activation qui lui est propre.

3.5 La fonction d'activation

Une fonction d'activation, qui associe à chaque valeur agrégée une unique valeur de sortie dépendant du seuil.

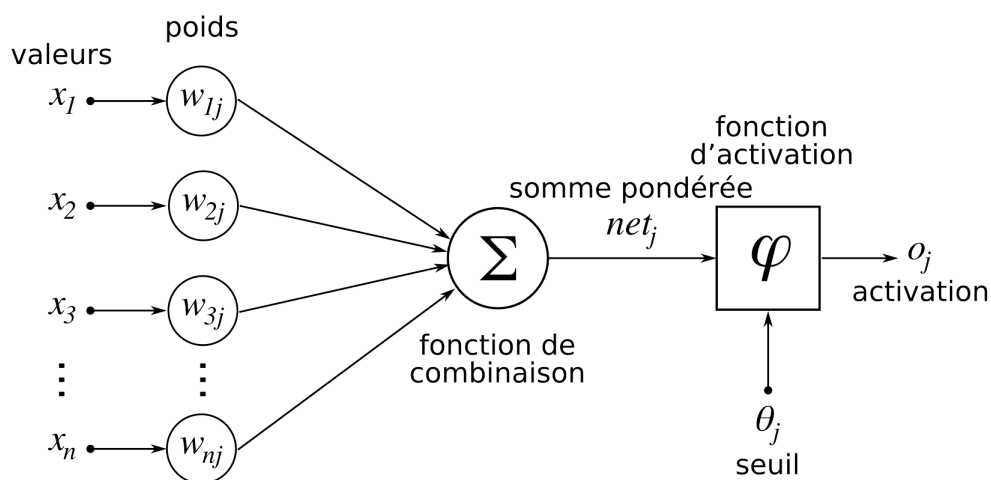


FIGURE 3.3 – Fonction d'activation.

3.5.1 La fonction Relu

La fonction Unité Linéaire Rectifiée en anglais, appelée Rectified Linear Unit (ReLU), est la fonction d'activation la plus couramment utilisée en Deep Learning. Elle est définie comme suit :

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

Cette fonction renvoie x si x est supérieur à 0, et 0 sinon. Autrement dit, elle calcule le maximum entre x et 0.

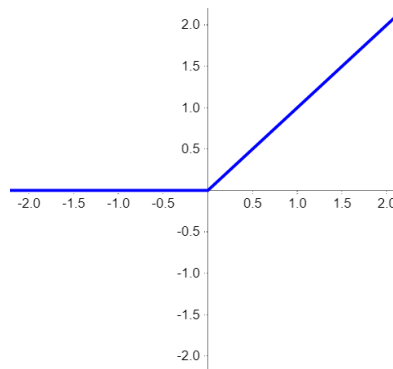


FIGURE 3.4 – La fonction Relu.

Cette fonction permet d'appliquer un filtre en sortie de couche. Elle laisse passer les valeurs positives dans les couches suivantes et bloque les valeurs négatives. Ce filtre permet alors au modèle de se concentrer uniquement sur certaines caractéristiques des données, les autres étant éliminées.

3.5.2 La fonction d'un softmax

La fonction Softmax est la fonction d'activation utilisée en dernière couche d'un réseau de neurones construit pour effectuer une tâche de classification multi-classes. Pour chaque sortie, Softmax donne un résultat entre 0 et 1. De plus, si l'on additionne ces sorties entre elles, le résultat donne 1.

La fonction Softmax est définie mathématiquement comme suit :

$$\text{Softmax}(x) = \frac{e^x}{\sum_i e^{x_i}}$$

La fonction Softmax, grâce à sa caractéristique de produire des résultats qui, additionnés, donnent 1, respecte les lois de probabilité. Elle est donc le noyau dur d'un réseau de neurones construit pour effectuer une tâche de classification multi-classes.

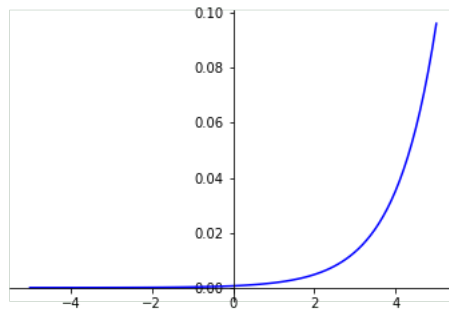


FIGURE 3.5 – La fonction d’un softmax.

3.5.3 La fonction sigmoïde

La fonction sigmoïde est la fonction d’activation utilisée en dernière couche d’un réseau de neurones construit pour effectuer une tâche de classification binaire. Elle donne une valeur entre 0 et 1.

La fonction sigmoïde est définie mathématiquement comme suit :

$$\text{Sigmoid}(x) = \frac{1}{1 + e^{-x}}$$

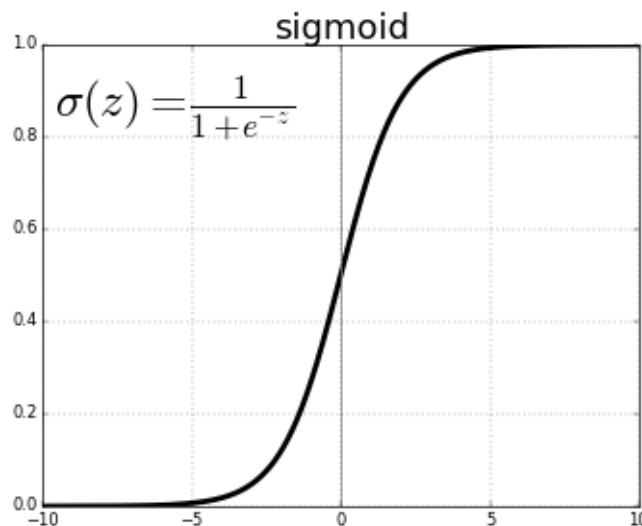


FIGURE 3.6 – La fonction sigmoïde.

Cette valeur peut être interprétée comme une probabilité. Dans une classification binaire, la fonction d’activation sigmoïde permet alors d’obtenir, pour une donnée, la probabilité d’ap-

partenir à une classe. Dans cet exemple, plus le résultat de la sigmoïde est proche de 1, plus le modèle considère que la critique est positive. Inversement, plus le résultat de la sigmoïde est proche de 0, plus le modèle considère que la critique est négative. La fonction d'activation sigmoïde permet donc d'obtenir un résultat ambivalent, donnant une indication sur deux classes à la fois.

3.6 Les différents types de model deep Learning

On va présenter dans cette section les modèles de deep Learning utilisés dans notre proposition à savoir les CNN, RNN, LSTM et ANN.

3.6.1 Le réseau de neurone convolutif (CNN)

Le nom 'Réseau de neurones à convolution' indique que le réseau emploie une opération mathématique appelée la convolution. Les réseaux de convolution sont un type spécialisé de réseaux neuronaux qui utilisent la convolution à la place de la multiplication matricielle générale dans au moins une de leurs couches. Les CNN sont l'un des meilleurs algorithmes d'apprentissage pour faire l'opération de convolution qui aide à l'extraction de fonctionnalités utiles à partir de points de données corrélés localement. La sortie des noyaux convolutifs est ensuite affectée à l'unité de traitement non linéaire (fonction d'activation), qui non seulement aide à apprendre les abstractions, mais intègre également la non-linéarité dans l'espace des fonctionnalités.

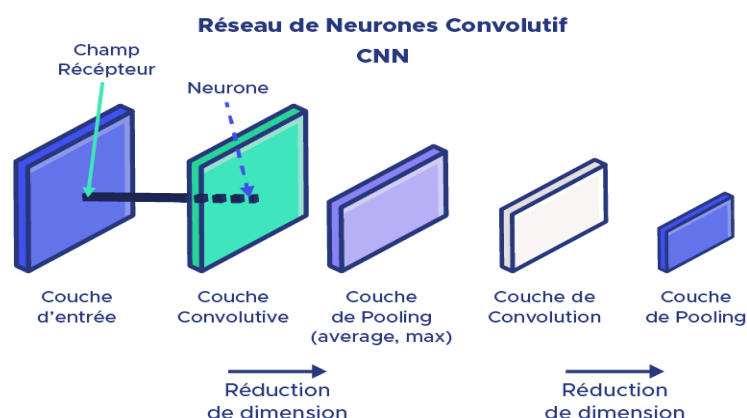


FIGURE 3.7 – Structure générale d'un réseau CNN.

3.6.2 Réseau de neurones récurrents (RNN)

Un réseau de neurones récurrent (RNN, Recurrent Neural Network) est un type de réseau de neurones artificiels principalement utilisé dans la reconnaissance vocale et le traitement automatique du langage naturel et la traduction automatique. Les RNN sont conçus de manière à reconnaître les caractéristiques séquentielles et les modèles d'utilisation des données requis pour prédire les scénarios faisant intervenir le contexte dans la prédiction d'un résultat .

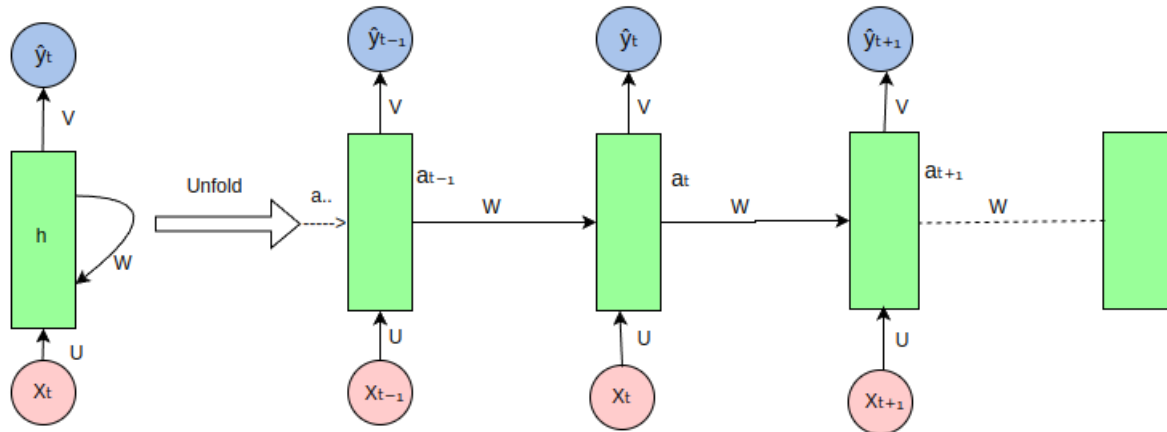


FIGURE 3.8 – Architecture de RNN.

3.6.3 Les réseaux LSTM

Les réseaux mémoire à long terme (LSTM : Long Short-Term Memory, en anglais) sont des dérivés de RNN. Ils peuvent apprendre et mémoriser des dépendances sur une longue durée. Les LSTM conservent ainsi les informations mémorisées sur le long terme. Ils sont particulièrement utiles pour prédire des séries chronologiques, car ils se rappellent des entrées précédentes. Outre ce cas d'utilisation, les LSTM sont également utilisés pour composer des notes de musique et reconnaître des voix.

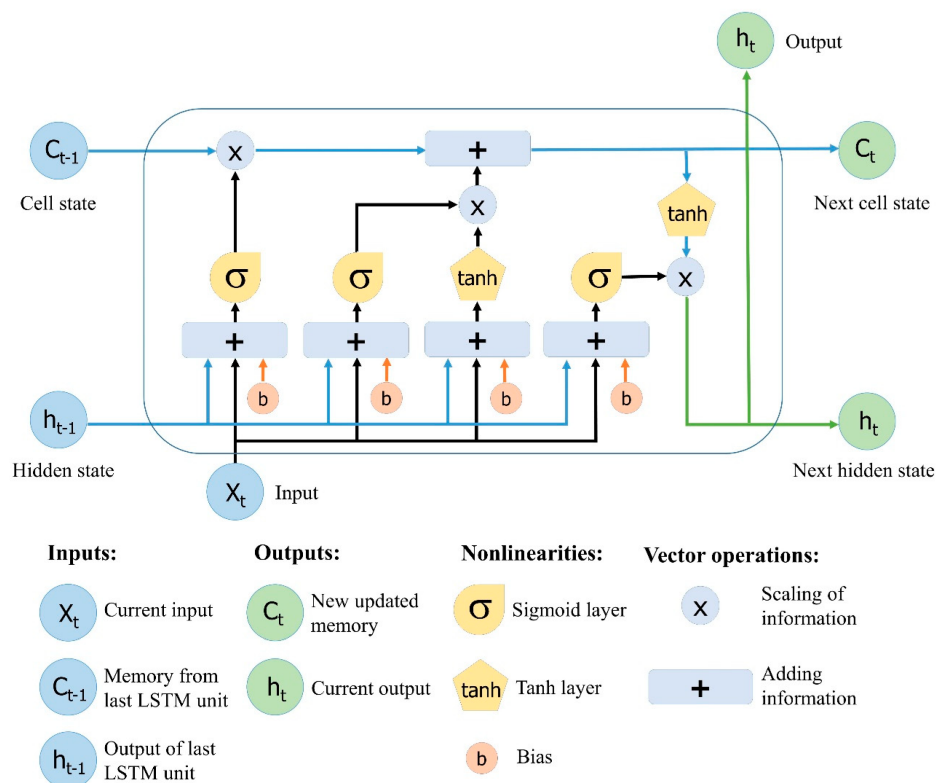


FIGURE 3.9 – Le module répétitif dans un LSTM.

3.6.4 Réseaux de neurones artificiel (ANN)

C'est une structure constituée de suite successive de couches de nœuds et qui permet de définir une fonction de transformation non linéaire des vecteurs d'entrées (composés dans le cas de classification des mots pondérés de leur poids) en vecteur de catégories. La disposition des neurones dans le réseau ainsi que le nombre de couches utilisées ont une influence sur le résultat de classification. Comparés aux autres méthodes de classification par apprentissage supervisé, les réseaux de neurones artificiels sont habituellement utilisés pour des tâches de classification. Par analogie avec la biologie, ces unités sont appelées neurones formels. Un neurone formel est caractérisé par :

- Le type des entrées et des sorties.
- Une fonction d'entrée.
- Une fonction de sortie.

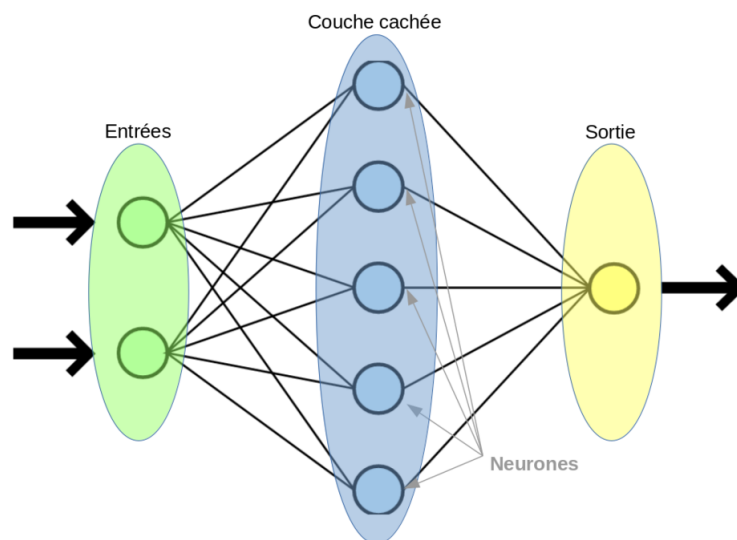


FIGURE 3.10 – Architecture de réseau de neurones artificiels.

3.7 Conclusion

Dans ce chapitre nous avons présenté l'apprentissage en profondeur (DL) et ses différents types de réseaux de neurones artificiels : les réseaux de neurones artificiels (ANN), les réseaux de neurones convolutifs (CNN) et les réseaux de neurones récurrents (RNN), y compris les réseaux à mémoire longue durée (LSTM).

Le chapitre suivant se concentre sur l'expérimentation de la construction de modèles de réseaux de neurones artificiels avec compression de données utilisant l'algorithme des composantes principales (PCA). Nous appliquerons également différents types d'algorithmes d'optimisation.

Chapitre 4

Méthodologie

4.1 Introduction

Au cours des chapitres précédents, nous avons exploré les différents types d’algorithmes et d’apprentissage automatique et d’apprentissage profond, ainsi que la notion de classification. En se basant sur ces concepts, ce chapitre se concentre sur l’expérimentation de la construction de modèles de réseaux de neurones artificiels dans deux scénarios : avec compression de données en utilisant l’algorithme des composants principales (ACP) et sans compression de données. Nous appliquerons également différents types d’algorithmes d’optimisation. Les résultats des tests et les évaluations seront enregistrés afin de déterminer l’efficacité des techniques utilisées et de comparer les modèles construits.

4.2 Notre Objectif

Cette étude vise à mettre en évidence l’importance de l’utilisation des données basé sur un étude statistique préalable dans la construction des modèles de prédiction supervisée. Nous menons une analyse comparative pour évaluer l’impact de la compression des données sur les algorithmes de classification profonde, ainsi que l’étendue de l’impact des algorithmes d’optimisation. Pour ce faire, nous avons utilisé un ensemble de données sur lesquels nous avons réalisé ces tests.

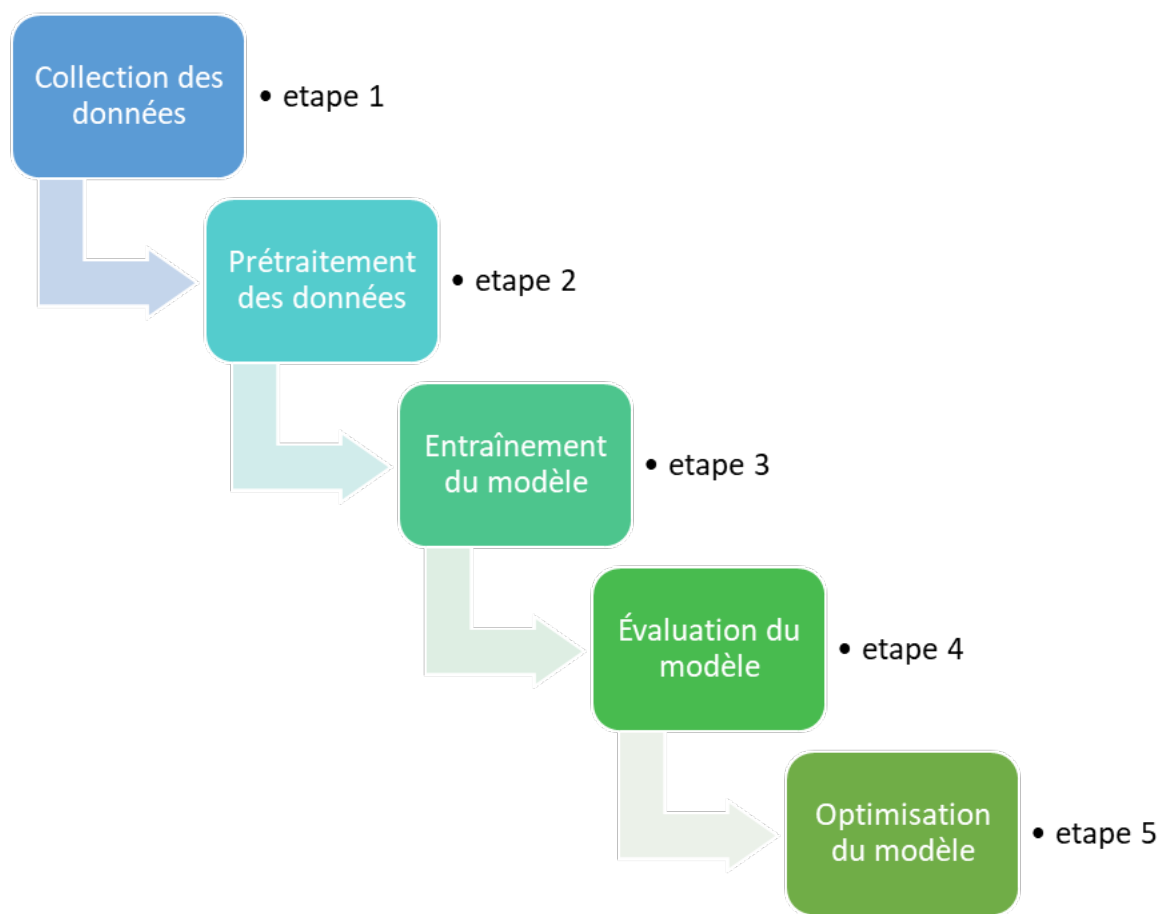


FIGURE 4.1 – Tux, le pingouin

4.3 Prétraitement

Le prétraitement des données, également connu sous le nom de nettoyage des données, représente une phase cruciale dans l'analyse de données. Cette étape englobe la transformation des données brutes en données exploitables en éliminant les erreurs, les valeurs manquantes, les doublons, et en les standardisant dans un format approprié. Les étapes suivantes illustrent le processus de prétraitement des données que nous avons effectué : /

1. **Collection des Données** : Pour cette étude, nous avons exploré et choisi un jeu de données approprié sur la plateforme **Kaggle**, comprenant environ 5 000 enregistrements en tenant compte de sa pertinence par rapport aux objectifs spécifiques de notre cas d'étude. Après avoir récupéré le jeu de données, nous avons effectué des ajustements pour l'adapter aux exigences et aux paramètres définis dans le cadre de notre étude.
2. **Importation des Données** : Pour l'importation de données dans l'environnement **Python**, il est courant de recourir à la bibliothèque **Pandas**, renommée pour sa capacité à gérer divers formats de fichiers de données. Plus précisément, la fonction **read_csv()** de pandas est souvent utilisée pour importer des données à partir de fichiers au format **CSV**. Cette fonctionnalité permet de charger les données contenues dans un fichier CSV dans une structure de données appelée **DataFrame**, offrant ainsi une manipulation et une analyse aisées des données dans Python.
3. **Répartition des Données Entraînement et Test** : Dans notre étude, nous avons effectué une étape cruciale de division des données de l'ensemble d'entraînement et de test. Nous avons attribué 80% des données à l'ensemble d'entraînement, tandis que les 20% restants ont été réservés à l'ensemble de test. Cette répartition nous a permis de disposer d'un ensemble suffisamment large pour entraîner nos modèles tout en conservant un ensemble distinct pour évaluer leur performance. Cela garantit une évaluation fiable de la capacité de généralisation de nos modèles sur des données qu'elles n'ont pas encore vues.
4. **Sélectionner les Colonnes Nécessaires** : Pour cette étude, nous avons sélectionné un ensemble de données considérable en fonction de sa pertinence pour notre cas d'étude portant sur la prédiction des maladies basée sur les symptômes. Après avoir récupéré le jeu de données, nous avons utilisé la fonction **drop()** pour éliminer les paramètres (colonnes) non significatifs pour notre modèle, dans le but de sélectionner les symptômes

les plus significatifs pour la prédiction des maladies. Ces symptômes ont été choisis avec soin en tenant compte de leur corrélation avec les maladies cibles et de leur importance dans la précision des diagnostics prédictifs. Cette étape a été réalisée en collaboration avec des experts médicaux et des médecins afin d'assurer la pertinence clinique et la validité des symptômes sélectionnés.

5. **Suppression des Documents Contenant des Valeurs Nulles** : La tâche appliquée initialement pour vérifier la présence de valeurs nulles dans un DataFrame Python. Ensuite, nous avons utilisé la méthode **isna().sum()** de la bibliothèque Pandas, permettant de calculer le nombre de valeurs nulles par colonne ou par ligne dans le DataFrame. Après avoir identifié les colonnes ou les lignes contenant des valeurs nulles, nous avons procédé à la suppression de ces entrées à l'aide de la méthode **dropna()** de Pandas. Cette opération a permis d'éliminer les lignes ou les colonnes contenant des valeurs nulles, assurant ainsi la qualité des données utilisées dans l'analyse subséquente.
6. **Compression des Colonnes (la Méthode de ACP)** : Dans ce projet, nous avons utilisé la méthode PCA (Analyse en Composantes Principales) de la bibliothèque **scikit-learn**, avec l'argument **n_components=2**. Cette étape de réduction de dimensionnalité avec **PCA_ncomponents=2** a pour l'objectif de comprimer les données vers un espace à deux dimensions. En fixant le nombre de composantes principales à 2, nous avons cherché à réduire la dimensionnalité du jeu de données tout en préservant au mieux sa structure et ses informations essentielles. Cette approche permet une représentation plus concise des données, ce qui facilite la visualisation, l'analyse et l'interprétation des résultats.
7. **Encodage des Caractéristiques**
 - **Encodage multi-étiquette** : nous avons réalisé un encodage multi-étiquette sur les variables relatives aux symptômes dans le jeu de données. Cela signifie que chaque symptôme présent dans les données a été transformé en une série de variables binaires, où chaque variable représente la présence ou l'absence d'un symptôme spécifique. Cette approche permet de traiter les symptômes comme vecteur des caractéristiques.
 - **Encodage de Label** : En ce qui concerne la variable cible, qui représente le pronostic de la maladie, nous avons utilisé un encodage de label en utilisant la fonction **LabelEncoder()** de la bibliothèque **scikit-learn**. Cela signifie que les différentes classes de pronostic de maladie ont été encodées avec des labels numériques

uniques. Cette méthode est couramment utilisée pour traiter les variables cibles dans les problèmes de classification, où chaque classe est représentée par un nombre entier unique, facilitant ainsi l'entraînement des modèles d'apprentissage en profondeur. Cette étape appelé la normalisation de label .

4.4 Représentation des Données

4.4.1 La Représentation en Vecteur de Caractéristiques Binaires

La représentation de vecteur de caractéristiques binaires est utilisée pour représenter les symptômes des patients. Chaque symptôme est représenté par une variable binaire, où 1 indique la présence du symptôme et 0 indique son absence. Par exemple, si nous avons un ensemble de symptômes comprenant "fièvre", "toux", "maux de tête", "fatigue", et qu'un patient présente seulement la fièvre et la toux, sa représentation en vecteur de caractéristiques binaires serait [1, 1, 0, 0]. Cette représentation permet de traiter les symptômes comme des variables catégorielles dans nos modèles d'apprentissage profond, facilitant ainsi l'analyse et la prédiction des diagnostics de maladies en fonction des symptômes présentés par les patients.

4.5 Méthodologie de Travail

Dans le cadre de notre étude, nous avons entrepris une évaluation comparative des performances de modèles d'apprentissage profond, notamment des réseaux de neurones artificiels (ANN), avec et sans l'utilisation de l'Analyse en Composantes Principales (ACP), sur un ensemble de données composé de 4962 enregistrements. Notre objectif principal était de déterminer le modèle offrant les meilleures performances en termes de précision, de rappel, de score F1 et d'exactitude. Dans la phase de prétraitement des données, différentes techniques ont été appliquées. Pour coder les variables liées aux symptômes, nous avons opté pour la représentation en vecteur de caractéristiques binaires, tandis que les variables cibles ont été encodées en utilisant l'encodage de label. De plus, notre processus de prétraitement comprenait une phase de sélection des caractéristiques significatives et l'élimination des valeurs nulles, les données étant nettoyées selon cette méthode avant d'être utilisées pour l'entraînement et le test des modèles. Les résultats obtenus ont ensuite été analysés et comparés afin de déterminer le modèle présentant les performances les plus prometteuses pour la tâche de classification considérée.

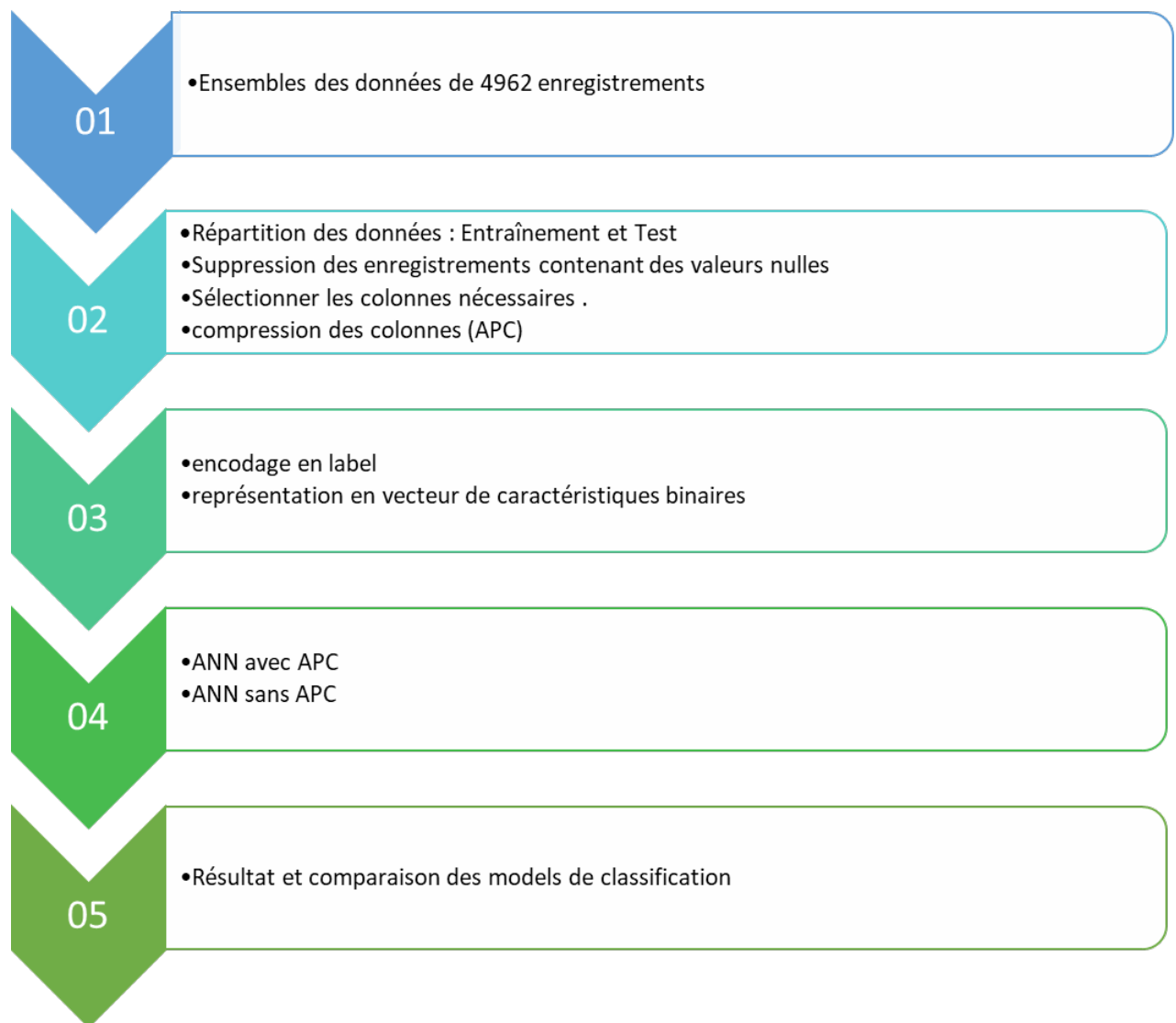


FIGURE 4.2 – Tux, le pingouin

4.6 Evaluation des Modèles de Classification

Dans notre étude, nous avons sélectionné trois mesures de performance essentielles pour évaluer les modèles : la précision, le rappel et l'exactitude (Accuracy en anglais). Ces mesures ont été choisies pour fournir une évaluation holistique des performances des modèles dans la classification des données médicales. En plus de ces métriques, nous avons également pris en compte le score F1

		prédite		Total
		Positive	Negative	
réelle	Positive	VP	FP	$VP + FP$
	Negative	FN	VN	$FN + VN$
Total		$VP + FN$	$FP + VN$	

TABLE 4.1 – Table caption

4.6.1 Rappel

Le rappel **R** est une métrique qui mesure à quelle fréquence un modèle identifie correctement les instances positives (vrais positifs) parmi tous les échantillons positifs réels dans l'ensemble de données.

$$R = \frac{VP}{VP + FN}$$

4.6.2 Précision

La précision **P** mesure la proportion d'échantillons correctement identifiés comme positifs parmi tous les échantillons identifiés comme positifs par le modèle.

$$P = \frac{VP}{VP + FP}$$

4.6.3 Exactitude

L'exactitude **E** est une mesure qui évalue la proportion totale d'échantillons correctement classés parmi tous les échantillons dans l'ensemble de données. C'est-à-dire qu'elle mesure la capacité globale du modèle à prédire correctement les classes de tous les échantillons, qu'ils

soient positifs ou négatifs.

$$E = \frac{VP + VN}{VP + VN + FP + FN}$$

4.6.4 Score f1

Le score **F1** est une métrique de performance qui combine à la fois la précision et le rappel d'un modèle en une seule valeur. Il offre une mesure globale de la capacité du modèle à classer correctement les échantillons positifs tout en minimisant à la fois les faux positifs et les faux négatifs.

$$F1 = \frac{2 * Prcision * Rappel}{Prcision + Rappel} = \frac{2 * VP}{2 * VP + FP + FN}$$

4.6.5 Temps d'entraînement

Le temps d'entraînement **T** du modèle fait référence à la durée nécessaire pour entraîner un modèle d'apprentissage sur un ensemble de données donné .

$$T = t_{\text{fin}} - t_{\text{début}}$$

4.7 Optimisation des models de classification

4.7.1 Adam (Estimation de moment adaptatif)

Ce sont les étapes principales de l'algorithme Adam pour mettre à jour les poids (3.4) pendant le processus d'entraînement. L'algorithme Adam combine la prise en compte des moments 1 et 2 ainsi que le taux d'apprentissage pour mettre à jour les poids, ce qui lui permet d'améliorer la stabilité et la vitesse du processus d'entrainement dans de nombreux cas. Pour accéder aux poids et aux biais de notre modèle, on peut utiliser les fonctions **model.weights** et **model.bias.numpy()**.

1. **Initialisation** :Les poids du modèle sont initialisés de manière aléatoire.
2. **Calcul des Gradients** :Les gradients des poids du modèle par rapport à la fonction de coût sont calculés à l'aide de la technique de la rétropropagation du gradient.
3. **Mise à jour des moments 1 et 2** :Les moments 1 et 2 sont mis à jour en utilisant les

formules suivantes :

$$m_t = \beta_1 \cdot m_{t-1} + (1 - \beta_1) \cdot \text{gradient}_t$$

$$V_t = \beta_2 \cdot V_{t-1} + (1 - \beta_2) \cdot \text{gradient}_t^2$$

(t est le temps actuel et β_1 et β_2 sont des paramètres qui contrôlent l'importance des moments.)

4. **Correction des biais 1 et 2** : Les biais 1 et 2 sont corrigés pour compenser les biais initiaux :

$$m_{t_{\text{corr}}} = m_t (1 - \beta_1^t)$$

$$V_{t_{\text{corr}}} = V_t (1 - \beta_2^t)$$

5. **Mise à jour des poids** : Les poids sont mis à jour en utilisant la formule suivante :

$$W_{t+1} = W_t - (lr \cdot (\sqrt{v_{t_{\text{corr}}} + \epsilon}) \cdot m_{t_{\text{corr}}})$$

(lr est le taux d'apprentissage et ϵ est une petite valeur pour éviter la division par zéro.)

6. **Répétition** : Les étapes 2 à 5 sont répétées jusqu'à ce qu'une condition d'arrêt spécifiée soit remplie.

4.7.2 SGD (Descente de gradient stochastique)

Ce sont les principales étapes de l'algorithme SGD pour mettre à jour les poids pendant le processus d'entraînement. Les poids sont mis à jour en fonction du gradient calculé et du taux d'apprentissage. Les étapes sont appliquées à chaque échantillon ou lot de données dans le cas de l'entraînement par mini-lots (mini-batch training) ou à chaque donnée dans le cas de l'entraînement par lots complets (batch training). Le processus est répété jusqu'à ce que les conditions spécifiées, telles que le nombre de cycles ou l'atteinte de la précision souhaitée, soient remplies

1. **Initialisation** : Les poids du modèle sont initialisés de manière aléatoire.
2. **Calcul des gradients** : Les gradients des poids du modèle par rapport à la fonction de coût sont calculés en utilisant la technique de la rétro propagation du gradient.

3. **Mise à Jour des poids** : Les poids sont mis à jour en utilisant la formule suivante :

$$W_{t+1} = W_t - lr \cdot \text{gradient}(t)$$

(où W_t représente les poids actuels, W_{t+1} est le poids mis à jour, et lr est le taux d'apprentissage.)

4. **Répétition** : Les étapes 2 et 3 (calcul des gradient + mise à jour de poids) sont répétées pour chaque donnée d'entraînement ou pour un nombre spécifié de cycles (epochs) jusqu'à ce qu'un critère d'arrêt spécifié soit atteint

4.8 Pertes des modèle de classification

Dans les modèles de classification, l'optimiseur utilise les informations de la fonction de perte pour ajuster les poids du modèle. Son objectif est de réduire l'erreur de prédiction au fil du temps. En somme, la fonction de perte fournit une mesure de performance que l'optimiseur utilise pour guider le processus d'optimisation vers la solution optimale.

4.8.1 Entropie croisée catégorielle éparses

La fonction de perte "entropie croisée catégorielle éparses", est couramment utilisée dans les problèmes de classification où les étiquettes sont fournies sous forme d'entiers plutôt que sous forme de vecteurs multi-étiquette. Elle mesure la différence entre les prédictions du modèle et les étiquettes réelles associées aux données. La formule mathématique de la fonction de perte entropie croisée catégorielle éparses est la suivante :

$$\text{SCE}(y, t) = - \sum_i t_i \cdot \log(y_i)$$

- SCE représente l'entropie croisée catégorielle éparses.
- y est la sortie (prédiction) produite par le modèle.
- t est la valeur réelle cible (étiquette), où t_i est soit 0 soit 1.
- \log représente le logarithme naturel.
- La somme est effectuée sur toutes les classes i .

Cette fonction de perte mesure la dissimilarité entre la distribution de probabilité prédite par le modèle (y) et la distribution de probabilité réelle (t) pour chaque exemple dans l'ensemble de

données. Elle est souvent utilisée dans les tâches de classification avec plusieurs classes.

4.9 Conclusion

Dans ce chapitre, nous avons clarifié la stratégie générale de notre travail en exposant clairement nos objectifs. Nous avons décrit la méthodologie adoptée pour mener une étude comparative entre deux modèles de classification des symptômes, combinés à différentes méthodes d'optimisation été adopté. Nous avons expliqué que la mise en œuvre de cette comparaison sur un corpus de données réelles sera détaillée dans le prochain chapitre, où nous présenterons également les résultats obtenus, ainsi que leur analyse approfondie et leur discussion.

Annexe A

Titre de l'annexe ici

1.1

1.1.1

Annexe B

Titre de l'annexe ici

2.1

2.1.1