République Algérienne Démocratique et Populaire
Ministère de l'enseignement supérieur et de la recherche scientifique

Université de 8 Mai 1945 – Guelma -
Faculté des Mathématiques, d'Informatique et des Sciences de la matière

Memoire de Fin d'étude Master

Filière : Informatique

Option :

Systèmes informatique

Thème

Prédiabète : Un Système de Détection et prédiction de diabète

Encadré Par :
Dr Boughareb Djalila

Présenté par :
*S*ahli Souha

Juin 2022

Remerciements

C’est avec un immense plaisir que je tiens à remercier très sincèrement toutes les personnes qui nous ont aidé et qui ont ainsi contribué à la réalisation de ce mémoire. Je tiens à remercier mon encadrante Dr. Boghareb djalila d’avoir dirigé ce travail. Mes remerciements vont également aux membres du jury pour l’intérêt qu’ils ont porté à notre projet de fin d’études en acceptant d’examiner ce travail et de l’enrichir par leurs propositions. J’exprime ma gratitude envers ma famille et tous mes amis pour leur soutien et encouragements tout au long de ce travail. Enfin, je voudrai également remercier toutes les personnes qui ont participé de près ou de loin à la réalisation de ce travail.

Résumé

Aujourd'hui, le diabète est l'une des maladies chroniques les plus courantes qui peut causer certaines complications qui peuvent causer parfois la mort. Donc un besoin urgent d'un outil de pronostic pouvant aider les médecins à détecter la maladie à un stade précoce et à recommander les changements de mode de vie nécessaires pour arrêter la progression de cette maladie. L'apprentissage profond est un besoin urgent d'aujourd'hui pour éliminer l'effort humain et proposer une automatisation plus élevée avec moins d'erreurs. Dans ce projet, un système de détection et de prédiction de diabète est développé en se basant sur une approche de deep learning (ANN+GAN). Les expérimentations menéessur la collection de données Pima ont données des résultats de prédiction encourageants en comparaison avec deux autres approches d'apprentissage automatique que nous avons implémentées à savoir : SVM et KNN.

Mots-clé : Indian pima dataset, intelligence artificial , deep learning, machine learnning, diabetes

ABSTRACT

Today, diabetes is one of the most common chronic diseases that can cause certain complications that can sometimes cause death. So, there is an urgent need for a prognostic tool that can help doctors detect the disease at an early stage and recommend the necessary lifestyle changes to stop the progression of this disease. Deep learning is an urgent need today to eliminate human effort and offer higher automation with fewer errors. In this project, a diabetes detection and prediction system is developed based on a deep learning approach (ANN+GAN). Experiments conducted on the Pima data collection have given encouraging prediction results in comparison with two other machine learning approaches that we have implemented, namely: SVM and KNN.

Keywords: Indian pima dataset, artificial intelligence, deep learning, machine learning,diabetes

.

Table des matières

Liste des tableaux

Table des figures

Introduction Générale

Avec l'augmentation considérable du nombre de malades chroniques en Algérie, et en particulier les malades diabétiques, et l'insuffisance des infrastructures et des moyens d'hospitalisation, les services de santé privés sont de plus en plus coûteux que jamais et la situation s'aggrave d'année en année. En effet le diabète n'est pas une maladie mortelle cependant elle peut avoir des séquelles néfastes sur le fonctionnement de certains organes comme les reins, les yeux et les organes périphériques. Comme on le dit « il vaut mieux prévenir que guérir », il est recommandé de prévenir la maladie pour l'éviter ou la prendre en charge à un stade précoce. Dans ce projet nous proposons de mettre en œuvre une application réelle et utile permettant à partir d'un ensemble de données médicales de détecter si la personne est atteinte du diabète ou de prédire si elle est en phase de pré diabète. Le système proposé emploi une technique de deep Learning (DL) pour l'apprentissage et la prise de décision. Nous avons également implémenté d'autres algorithmes (SVM et KNN) pour tester et comparer les résultats de notre approche avec ces techniques déjà utilisées dans la littérature. L'apprentissage profond ouvre de nouvelles perspectives dans le domaine de la santé. Avec des flux de données considérables, aujourd'hui il est possible d'appliquer des algorithmes qui donnent des réponses automatiques et plus précises aux problèmes médicaux.

Organisation de mémoire

Le mémoire est organisé en trois chapitres : Le premier chapitre comprend une présentation du deep Learning et ses différentes techniques appliquées en domaine médicale. Quelques travaux connexes sont également présentés en fin du chapitre. Le deuxième chapitre est consacré à l'étude conceptuelle de notre système. Le troisième chapitre présente les outils utilisés dans l'implémentation de l'approche, l'interface du système et finalement, la présentation des tests et résultats.

Objectif et motivation

L'objectif de ce projet est de proposer une nouvelle approche de prédiction du diabète qui donne des résultats utiles et efficaces. Ceci va aider à prédire si un patient donné va être un futur diabétique. Si c'est le cas le patient va essayer de prendre les précautions nécessaires (suivre un régime alimentaire adéquat, pratiquer du sport périodiquement, etc.) pour prévenir cette maladie chronique.

Chapitre 1

Application du deep Learning en domaine médical

1.1 Introduction

Le Deep Learning a révolutionné le domaine de l'intelligence artificielle. Basée sur des réseaux neuronaux artificiels, c'est une technologie qui tente d'imiter les secrets du fonctionnement neuronal du cerveau humain dans l'espoir qu'un jour il puisse le reproduire virtuellement et le transposer sur des machines qui deviendraient aussi intelligentes et autonomes que les humains. Cette technologie a permis aux scientifiques de faire des progrès considérables dans la reconnaissance et la classification des données. [Deluzarche, 2021]

Dans ce chapitre on va parler sur le fonctionnement et les techniques de deep learning dans le domaine médical et en particulier l'utilisation de deep learning pour la prédiction du diabète.

1.2 Définition du deep Learning :

L'apprentissage profond (Deep Learning) est un ensemble de techniques d'apprentissage automatique qui a permis des avancées importantes en intelligence artificielle dans les dernières années. Dans l'apprentissage automatique, un programme analyse un ensemble de données afin de tirer des règles qui permettront de tirer des conclusions sur des nouvelles données. L'apprentissage profond est basé sur ce qui a été appelé, par analogie, des « Réseaux de neurones artificiels », composés de milliers d'unités (les « neurones ») qui effectuent chacune de petites opérations simples. Les résultats d'une première couche de « neurones » servent d'entrée aux calculs d'une deuxième couche et ainsi de suite [Ghediri *et al.*, 2021].

L'apprentissage en profondeur permet aux modèles informatiques avec plusieurs couches de traitement d'apprendre plusieurs degrés d'abstraction pour les représentations de données. Ces techniques ont considérablement amélioré l'état de l'art en matière de reconnaissance vocale, de reconnaissance visuelle d'objets, de détection d'objets et de divers autres domaines tels que le développement de médicaments et la génomique [Bird *et al.*, 2009].

1.3 Domaine d'application du Deep Learning :

Les domaines d'application des techniques de DL sont divers. En effet, ces techniques se développent dans le domaine de l'informatique appliquées aux NTIC, à la robotique, à la reconnaissance ou comparaison de formes, la sécurité, la santé, la pédagogie assistée par l'informatique, et plus généralement à l'intelligence artificielle.

L'apprentissage profond permet à un ordinateur déformable d'analyser les émotions révélées par un visage photographié ou filmé, ou analyser les mouvements et position des doigts d'une main, ce qui peut être utile pour traduire le langage des signes, améliorer le positionnement automatique d'une caméra, etc.

Elles sont utilisées pour certaines formes d'aide au diagnostic médical (ex : reconnaissance automatique d'un cancer en imagerie médicale), ou de prospective ou de prédiction (ex : prédiction des propriétés d'un sol filmé par un robot ou la prédiction des maladies).

1.4 Exemples d'Application de Deep Learning :

L'apprentissage profond est utilisé dans différents secteurs, de la conduite automatisée aux Dispositifs médicaux. Grâce au Deep Learning nous pouvons maintenant [w1] :

- Faire une colorisation des images en noir et blanc.
- Ajouter des sons à des films silencieux.
- Faire de la traduction automatique.
- Faire de la classification des objets en photographies.
- Générer d'écriture automatique.
- Génération de légende d'image.
- Jeu automatique.

1.5 Les types d'apprentissage profonds :

Dans l'apprentissage profonds, les modèles prédictifs utilisent divers algorithmes fondamentaux pour déduire des relations mathématiques à partir des données de formation. Il existe trois types de méthodes d'apprentissage :

1.5.1 Apprentissage non supervisé :

Dans l'apprentissage non supervisé le modèle est alimenté par des données de formation non classifiés (c'est-'a-dire uniquement les entrées). Ensuite, le modèle classe les points de données de test dans différentes classes en trouvant des points communs entre elles.

1.5.2 Apprentissage semi-supervisé :

Comme son nom l'indique, l'apprentissage semi-supervisée hérite des donnés de de l'apprentissage supervisé et de l'apprentissage non supervisé. Un ensemble de données semi-supervisées contient principalement des points de données de formation non classifiées ainsi que de petites quantités de données classifiés.

1.6 Les techniques du Deep Learning :

1.6.1 Les réseaux neuronaux convolutifs (CNN) :

Les premiers réseaux neuronaux ont été développés par Yann lecun en 1988, il s'agit d'un réseau exploité par le traitement d'image et à la détection d'objets. CNN est un type de réseau de neurones spécialisés en traitement de données ayant une topologie pareille à une grille. Qui se sont classer très efficaces dans des divers domaines comme la reconnaissance et la classification d'images et vidéos. Il est utilisé pour identifier les visages, les objets, panneaux de circulation et auto-conduite des voitures. Récemment. Dans le ML, un réseau convolutif est un type de réseau de neurones feed-forward, il a été inspiré par des processus biologiques. Il existe cinq (5) principales opérations illustrées dans le CNN à savoir[w2] :

- La couche convolution.
- La couche Rectified Linear Unit.
- La couche pooling.
- La couche entièrement connectée.
- La couche de perte.

1.6.2 Les réseaux neuronaux récurrents (RNN) :

Les couches de réseau dans le RNN forme des cycles d'érigés, en d'autres termes, elle utilise la sortie d'une couche comme une nouvelle entrée dans une nouvelle couche. Les réseaux neuronaux récurent sont habituellement crées pour sous-titrer des images, traduire automatiquement ou pour traiter le langage naturel comme ils aident à expliquer les informations temporelles ou séquentielles. Les RNN est utilisé dans la détection automatique des apnées du sommeil sur l'ECG nocturne [Prakash *et al.*, 2021] et dans le traitement automatique de la parole [Gelly, 2017].

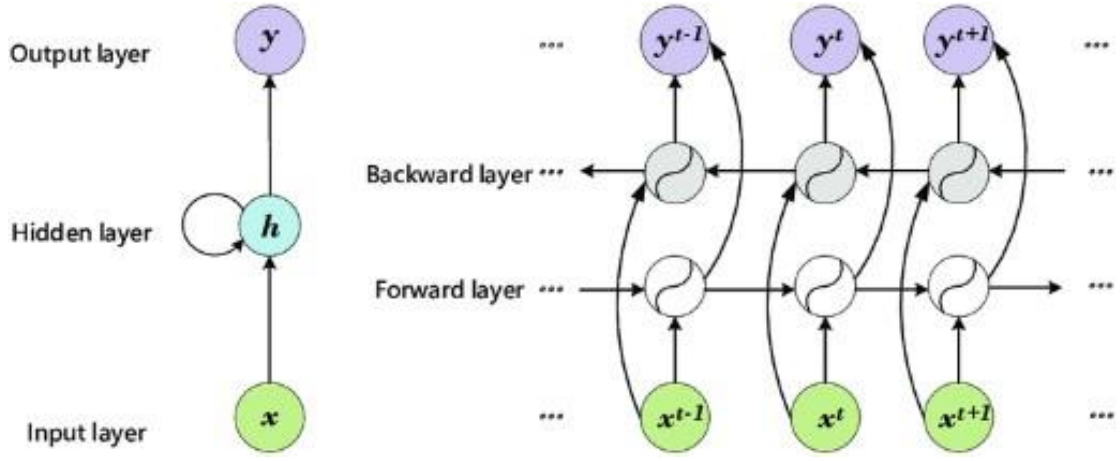

Figure 1.1 – l'architecture de réseau récurent

1.6.3 Les réseaux neuronaux à long terme et court terme (LSTM) :

les LSTM sont des types de RNN aptes à apprendre et mémoriser des dépendances à long terme le RNN mémorise les sorties de ces réseaux dans le but de les utiliser comme nouvelle entrées. Il y a aussi des explications comme au RNN comme les réseaux de mémoire à long terme et à court terme qui sont utilisés dans la composition musicale, la reconnaissance ou bien dans le développement de nouveaux médicaments [W3].

1.6.4 Radial basis function network (RBFN) :

les réseaux à fonction de base radiale dont des types de réseaux neuronaux à propagation avant qui exploitent des fonctions de base radiales comme activation. Les RBFN contiennent une couche d'entrées, de sortis cachée. Ils sont utilisés aussi à la prédiction des séries chronologiques, la classification et à la régression. Les RBFN sont utiliser pour résoudre le problème de la stabilité des Bras robotique aérien (ARA), les ARA permettent aux drones aériens d'interagir et d'influencer les objets dans divers environnements. [w4]

1.6.5 Les réseaux antagonistes génératifs (GAN) :

C'est un algorithme de deep Learning En utilisant les GAN on peut créer de nouvelles instances de données qui ressemblent aux données sue lesquelles ils ont été formés. Les GAN se composent d'un générateur et d'un discriminateur [3]. Les réseaux antagonistes génératifs sont utilisés synthèse de mouvement respiratoire par réseau antagoniste génératif doublement conditionnel en imagerie tomodensitométrique 4D . Ils sont utilisé aussi pour la reconstruction super-résolution et la segmentation en IRM (imagerie par résonance magnétique) [Delannoy *et al.*, 2020].

1.6.6 K plus proches voisins (KNN) :

La méthode des k plus proches voisins (k-nearest neighbor, KNN) est une méthode supervisée. Elle a été utilisée dans l'estimation statistique et la reconnaissance des modèles comme une technique non paramétrique, cela signifie qu'elle ne fait aucune hypothèse sur la distribution des données. L'algorithme KNN est parmi les algorithmes plus simples d'apprentissage automatique. Il est un type d'apprentissage basé sur l'apprentissage paresseux (lazy Learning). En d'autres termes, il n'y a pas de phase d'entraînement explicite ou très minime. Cela signifie que la phase d'entraînement est assez rapide. [] La méthode KNN suppose que les données se trouvent dans un espace de caractéristiques. Cela signifie que les points de données sont dans un espace métrique. Les données peuvent être des sca- laires ou même des vecteurs multidimensionnels [] Cette méthode utilise principalement deux paramètres : une fonction de similarité pour comparer les individus dans l'espace de caractéristiques et le nombre k qui décide combien de voisins influencent la classification. [] Pour tester la similarité entre deux vecteurs, la distance est utilisée. Elle permet de mesurer le degré de différence entre deux vecteurs. Il existe plusieurs types de distance parmi lesquels on trouve :

Distance euclidienne : x,y sont des vecteur

La distance de Minkowsky :

x,y sont des vecteurs

p : parametre

$$d(x,y)=\sqrt{\sum_{i=1}^{n}(x_i-y_i)^2}$$

$$d(x,y)=\left(\sum_{i=1}^{n}|x_i-y_i|^p\right)^{1/p}$$

La distance de Manhattan :
x,y sont des vecteurs

$$d(x,y)=\sum_{i=1}^{n}|x_i-y_i|$$

1.6.7 Les réseaux de neurones artificiels (ANN) :

Les ANN sont des systèmes adaptatifs artificiels qui s'inspirent des processus de fonctionnement du cerveau humain. Ce sont des systèmes capables de modifier leur structure interne par rapport à un objectif de fonction. Ils sont particulièrement adaptés à la résolution de problèmes de type non linéaire, étant capables de reconstruire les règles floues qui régissent la solution optimale de ces problèmes [w5]. Les éléments de base de l'ANN sont les nœuds, également appelés éléments de traitement (PE), et les connexions. Chaque nœud a sa propre entrée, à partir de laquelle il reçoit des communications d'autres nœuds et/ou de l'environnement et sa propre sortie, à partir de laquelle il communique avec d'autres nœuds ou avec l'environnement. Enfin, chaque nœud a une fonction f par laquelle il transforme sa propre entrée globale en sortie (Fig. 1) [w5].

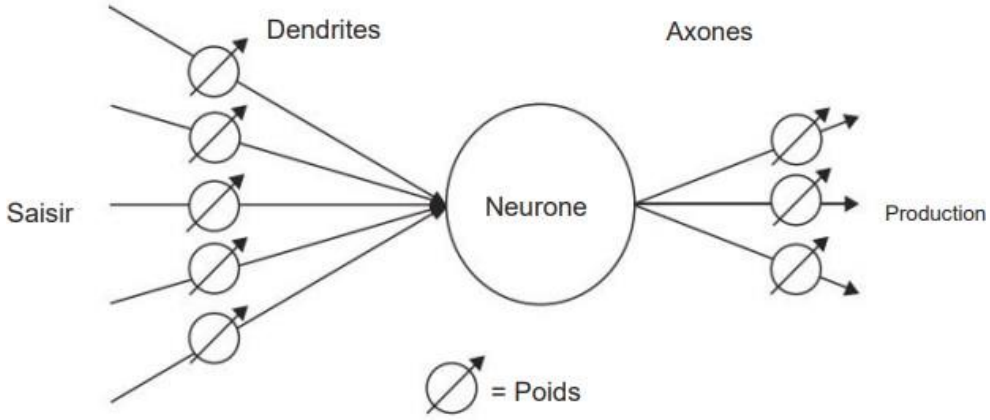

Figure 1.2 – Schéma d'un élément de traitement unique (PE) contenant un neurone, des Dendrites pondérées et des axones pour traiter les données d'entrée et calculer une sortie

Les applications potentielles de la méthodologie ANN dans les sciences pharmaceutiques vont de l'interprétation des données analytiques à la conception des médicaments et des formes de dosage, en passant par la biopharmacie et la pharmacie clinique. [Agatonovic-Kustrin & Beresford, 2000]

1.7 Comment ça marche l'algorithme du deep Learning :

Après avoir vu plusieurs exemples, un réseau de neurone artificiel peut être utilisé dans la reconnaissance d'images, pour identifier un animal par exemple. Chaque couche du réseau identifie une caractéristique de l'animal : la silhouette, la tête, les deux oreilles, les pattes. . .etc. les réseaux de neurones artificiels sont également très utilisés dans le traitement automatique du langage ou Natural langage processing (NLP) [w3].

1.8 Les réseaux à une couche :

Le réseau a une couche dispose deux couches : couche d'entrée et couche de sortie.figure(1.3)

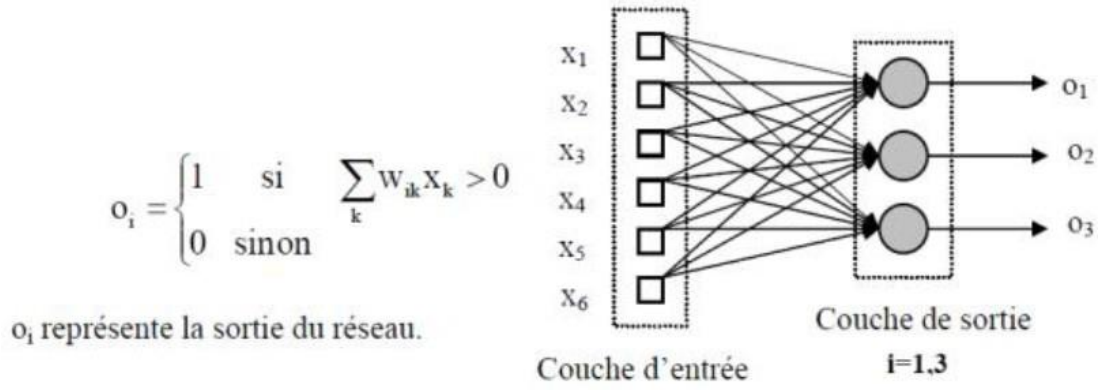

Figure 1.3 – réseau a une couche

1.9 Les réseaux multicouches :

Le perceptron multicouche (multi layer perceptron MLP) est un classificateur de type réseau neuronal formel organisé en plusieurs couches (Figure 1.4) au sein des quelles une information circule de la couche d'entrée vers la couche de sortie uniquement. Chaque couche est constituée d'un nombre variable de neurones, les neurones de la couche de sortie correspondant toujours aux sorties du système

- **Couche d'entrée :** l'ensemble des neurones d'entrée.
- **Couche de sortie :** l'ensemble des neurones de sortie.
- **Couches cachées :** l'ensemble des couches intermédiaires, elles n'ont aucun contact avec l'extérieur.

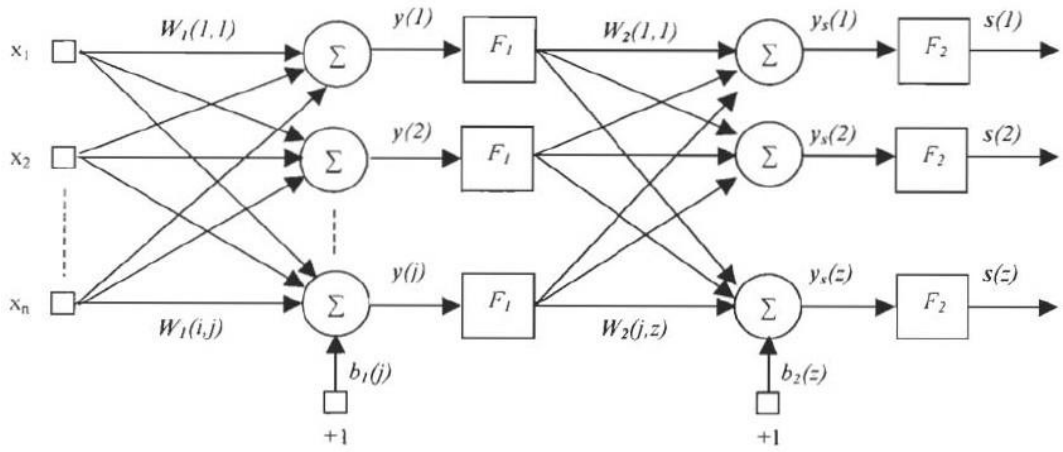

Figure 1.4 – réseau multi couches

1.10 Le Deep Learning en détection et prédiction des maladies :

Aujourd'hui la médecine moderne ne peut s'exercer sans l'utilisation d'images, quelques soient dans la dermatologie, la radiologie, cardiologie, urologie, gastroentérologiques. . .etc.

La médecine de précision est une nouvelle approche de la recherche clinique et des soins aux patients qui met l'accent sur la compréhension et le traitement des maladies en intégrant les données multimodales ou multiomiques d'une personne pour prendre des décisions adaptées au patient. Avec les ensembles de données vastes et complexes générés à l'aide d'approches diagnostiques de médecine de précision, de nouvelles techniques de traitement et de compréhension de ces données complexes étaient nécessaires. Dans le même temps, l'informatique a progressé rapidement pour développer des techniques qui permettent le stockage, le traitement et l'analyse de ces ensembles de données complexes, un exploit que les statistiques traditionnelles et les premières technologies informatiques ne pouvaient pas accomplir.

L'apprentissage automatique, une branche de l'intelligence artificielle, est une méthodologie informatique qui vise à identifier des tendances complexes dans les données qui peuvent être utilisées pour faire des prédictions ou des classifications sur de nouvelles données invisibles ou pour l'analyse de données exploratoires avancées.

L'analyse par apprentissage automatique des données multimodales de la médecine de précision permet d'analyser de vastes ensembles de données et, en fin de compte, de mieux comprendre la santé et les maladies humaines. Cet examen porte sur l'utilisation de l'apprentissage automatique pour les « mégadonnées » de la médecine de précision, dans le contexte de la génétique, de la génomique et au-delà [Iyer *et al.*, 2015].

Le deep Learning est utilisé dans la prédiction des maladies par exemples :

— **Prédiction de COVID-19 :**

Le COVID-19 s'est révélé être une maladie virale infectieuse et mortelle, et sa propagation rapide et massive est devenue l'un des plus grands défis du monde.

Les chercheurs ont fourni un examen complet du rôle de l'apprentissage profond et l'apprentissage automatique dans la recherche de techniques de prédiction pour le

COVID-19. Un modèle mathématique a été formulé pour analyser et détecter sa menace potentielle. Le modèle proposé est un algorithme de détection intelligent basé sur le cloud utilisant une machine à vecteurs de support (CSDC-SVM) avec des tests de validation croisés. Les résultats expérimentaux ont atteint une précision de 98,4

- **Le projet SCRUM-Japan Genesis :** vise à établir un algorithme, appelé séquençage virtuel (VSQ), en utilisant la technologie d'apprentissage profond (DL) et les diagnostics pathologiques pour la prédiction des anomalies du génome du cancer [24].
- **Prédiction du développement de la maladie d'Alzheimer :** pour des patients atteints d'une déficience cognitive légère. [Valenchon, 2019]
- **Les maladies cardio-vasculaires (MCV) :** désignent, pour la plupart, des affections comprenant des veines limitées ou obstruées qui peuvent provoquer une crise cardiaque, une angine de poitrine ou un accident vasculaire cérébral. Le classificateur d'apprentissage automatique prédit l'affection en fonction de l'état de l'effet secondaire subi par le patient. [Kumar *et al.*, 2020]
- **Détection et prédiction prématuré des maladies cardiaques :** à l'aide de l'optimisation de KNN qui à donner un résultat de 95.71 % [28]

1.11 Prédiction du diabète

Le diabète est parmi les maladies les plus rependues à travers le monde. Il est considéré comme une maladie qui se propage comme une épidémie dans le Monde entier, elle peut atteindre toutes les générations (enfants, jeunes, les personnes Agées). Cette maladie peut entrainer des effets très graves en termes de défaillance d'organes et peut entrainer la mort aussi. Le diabète a deux types [Deluzarche, 2021] :

- **Diabète de Type 1 :** le diabète de type-1 du a une absence d'insuline par le pancréas. Les Diabétiques de type-1 doivent s'injecter quotidiennement de l'insuline. [w6]
- **Diabète de type-2 :** Le diabète de type 2 est dû à une résistance à l'insuline entraînant Une carence insulinique à terme. Les techniques d'exploration de données ont supplanté les méthodologies existantes en offrant une meilleure prédiction, une meilleure exactitude et une meilleure précision. De plus, l'apprentissage automatique est une technologie de l'intelligence artificielle qui apprend les relations entre les nœuds sans formation préalable.

Dans cette section, nous passerons en revue certaines études antérieures pour prouver le Concept d'utilisation des méthodes d'exploration de données dans le modèle de prédiction de la conduite, principalement pour le diabète [Naz & Ahuja, 2020a].

L'objectif principal de notre travail est de proposer un outil de développement pour la prédiction et la détection précoce du diabète avec une meilleure précision. Il existe une grande quantité de données disponibles sur Internet ou dans des sources externes, on va présenter leurs méthodes et les résultats obtenu dans le tableau ci dessue : (tableau 1)

Méthodes	Résultats(%)	Auteurs
DNN-BFGS	77.09	Abdullah caliskan et al statlog [Caliskan *et al.*, 2018]
KNN	67.7	H. kahramanli et al statlog [Kahramanli & Allahverdi, 2008a]
FAR	75.7	H. kahramanli et al statlog [Kahramanli & Allahverdi, 2008a]
PLP	81.9	Aliza Ahmad [Kahramanli & Allahverdi, 2008b]
DNN	82	Aliza ahmad [Kahramanli & Allahverdi, 2008b]
KNN	76.6	Abdullah caliskan et al statlog [Caliskan *et al.*, 2018]
K-means and DT	90.03	Chen W [Chen *et al.*, 2017]
DL,ANN,SVM and DT(Highest accuracy achieved using DT)	98.07	Naz et Sachin Ahuja [El_Jerjawi & Abu-Naser, 2018]
NB	79.56	Iyer A, Jeyalatha S, SumbalyR [Iyer *et al.*, 2015]
Neural Network with Genetic Algorithm	87.46	Mohammad S, Dadgar H, Kaardaan M. [Deluzarche, 2021]
LDA - MWSVM	89.74	Çalişir D, Doğantekin
PCA, K-Means Algorithme	72	Patil RN [Naz & Ahuja, 2020b]

TABLE 1.1 – résultats de diverses méthodes des travaux connexes.

LDA – MWSVM : Linear Discriminant Analysis-Morlet Wavelet Support Vector.
PCA : Principal Component Analysis
BFGS : Broyden Fletcher Goldfard Shanno
DT : Decision tree
PLP : Pperceptual linear Predective
FAR : Forest Accuracy Report

Plusieurs recherches ont été faites sur la classification des données sur le diabète à l'aide de l'ensemble de données « Pima Indian diabete » on a présenté quelques travaux et leurs résultats dans le tableau 01 : Utilisation d'une stratégie de formation simple pour le classificateur DNN avec le populaire algorithme d'optimisation L-BFGS qui a obtenu la précision de classification de 77,09% [Caliskan *et al.*, 2018]. La précision de la classification obtenue en utilisant seulement le KNN a donnée 76.6% [Kahramanli & Allahverdi, 2008a]. Et en utilisant K-means et DT qui a été proposée par Chen avec une précision de 90,03% [Chen *et al.*, 2017]. Sujarani et al Ont proposé un réseau neuronal de régression générale et ont atteint une précision de classification de 80,21% [Deluzarche, 2021]. Naz Sachin Ahuja ont fait un modèle avec les algorithme ANN, SVM, DL et DT et ils ont obtenu une précision de 98% [El_Jerjawi & Abu-Naser, 2018]et elle est classé parmi les meilleures recherches.

1.12 Conclusion

Dans ce chapitre on a parlé tout d'abord du deep Learning : sa définition et ses différentes techniques ainsi que ces domaines d'application. Nous avons également cité

les techniques de prédiction qui comprennent les réseaux de neurones artificiels que nous avons choisis comme meilleure solution pour la prédiction du diabète.

Chapitre 2

Conception

2.1 Introduction

Dans ce chapitre nous présentons les étapes de conception de notre système de prédiction de diabète. Une modélisation UML est proposée suivie de la méthodologie de travail partant de la collecte de données, puis leur prétraitement jusqu'à l'application de l'approche de deep learning proposée.

2.2 Modelisation UML :

2.2.1 Définition du Langage UML

Le langage UML (Unified Modeling Language, ou langage de modélisation unifié), est un langage visuel constitué d'un ensemble de schémas, appelés diagrammes, qui donnent chacun une vision différente du projet à traiter. Dans le but de capturer l'aspect dynamique du système, nous avons créé le diagramme de séquence présentée par la figure (2.1).

Diagramme de séquences :

Description : pour que l'utilisateur obtiendra le résultat de la prédiction il doit remplir le formulaire avec les informations nécessaires (taux d'insuline, la glycémie, âge, épaisseur de la peau, etc.) puis le modèle prédit si l'utilisateur a le diabète ou non. La figure 2.1 représente le diagramme de séquences.

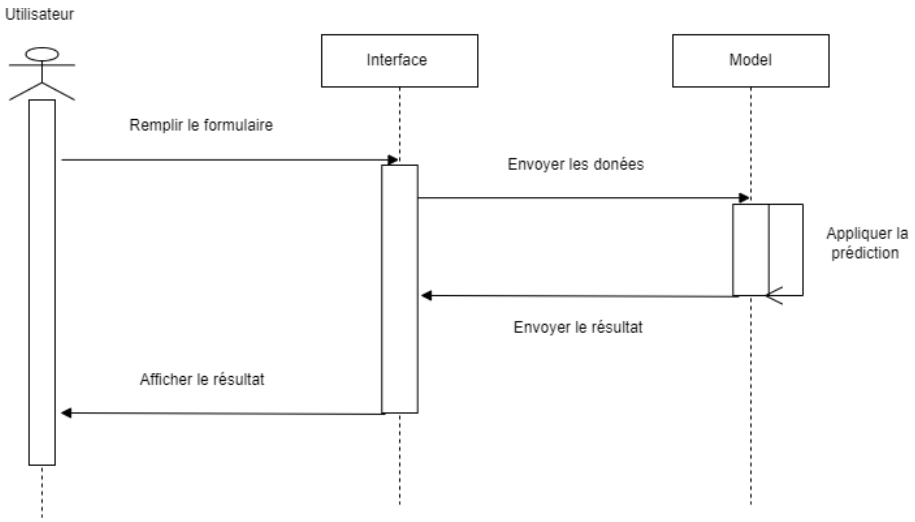

Figure 2.1 – diagramme de séquence.

2.3 Méthodologie :

2.3.1 Enssemble des données (data set) :

La principale motivation de l'utilisation de l'ensemble de données PIMA est la suivante : la plupart des gens dans le monde suivent le même style de vie, avec une plus grande dépendance à l'égard des aliments transformés et un déclin de l'activité physique. PID est une étude de cohorte à long terme depuis 1965 par le NIDDK en raison du risque maximal de diabète.

L'ensemble de données contient certains paramètres de diagnostic et des mesures grâce auxquels le patient peut être identifié avec presque tout type de maladie chronique ou de diabète avant le temps. Le PID est composé d'un total de 768 instances, dont 268 échantillons ont été identifiés comme diabétiques et 500 comme non-diabétiques. Les 8 attributs les plus influents qui ont contribué à la prédiction du diabète sont les suivants : plusieurs grossesses de la patiente, IMC, taux d'insuline, âge, tension artérielle, épaisseur de la peau, glycémie, etc.

le tableau ci-dessus présente les caractéristiques des attribues utiliser dans PIMA [Naz & Ahuja, 2020c] .

Le dataset Pima peut être téléchargeable depuis depuis ce lien : https ://www.kaggle.com/uciml/pima-indians-diabetes-database/data.

2.3.2 Prétraitement des données :

L'objectif de cette étape est de collecter une quantité de données suffisante pour constituer une base représentative de données susceptibles d'intervenir dans la phase d'appren-

caractéristiques	**intervalle**
Grossesses : (nombre de fois enceinte).	1-17
Glucose : (Concentration de glucose plasmatique à 2 heures dans un test de tolérance au glucose par voie orale).	0-199
Pression artérielle : (Pression sanguine diastolique).	0-122
Épaisseur de la peau : (Épaisseur du pli cutané du triceps (mm)).	0-99
Insuline : (Insuline sérique de 2 heures (mu U/ml)).	846
IMC : (Indice de masse corporelle).	0-67.1
Fonction Pedigree Diabète.	0.078–2.42
Age	21-81
résultat	0/1

TABLe 2.1 – Description des caractéristiques de l'ensemble de données "Pima Indian Diabètes".

tissage. En effet, il est souvent souhaitable d'analyser les données de manière à déterminer des caractéristiques d'identification pour détecter ou différencier les données. Ces caractéristiques constituent l'entrée du réseau neuronal.

2.3.3 Normalisation de données :

De manière générale, les bases de données doivent subir un prétraitement afin d'être adaptées aux entrées et sorties du réseau de neurones. Un prétraitement courant consiste à effectuer une normalisation appropriée, qui tienne compte de l'amplitude des valeurs acceptées par le réseau.

2.3.4 L'apprentissage :

La caractéristique principale des réseaux de neurones est leur capacité à apprendre (par exemple à reconnaître une lettre, un son, etc.). Mais cette connaissance n'est pas acquise dès le départ. La plupart des réseaux de neurones apprennent par l'exemple en suivant un algorithme d'apprentissage, pour développer une application à base des ANN, il est nécessaire de diviser les données en trois ensembles : deux pour l'apprentissage et une autre pour le test.

2.3.5 Architecture de l'ANN :

Alors, après avoir téléchargé Pima, nous avons passé au choix de l'architecture du réseau de neurones artificiels. Alors, nous avons essayé plusieurs architectures. Puis, nous avons choisi d'utiliser un modèle séquentiel comprenant 3 couches. Les deux premières couches sont activées par la fonction 'relu'. La première couche contient 1000 neurones et 500 dans la seconde. Tandis que, la couche de sortie est activée par la fonction 'sigmoïde' car nous avons une sortie 0 ou 1. L'architecture du réseau est présentée par la figure (2.2).

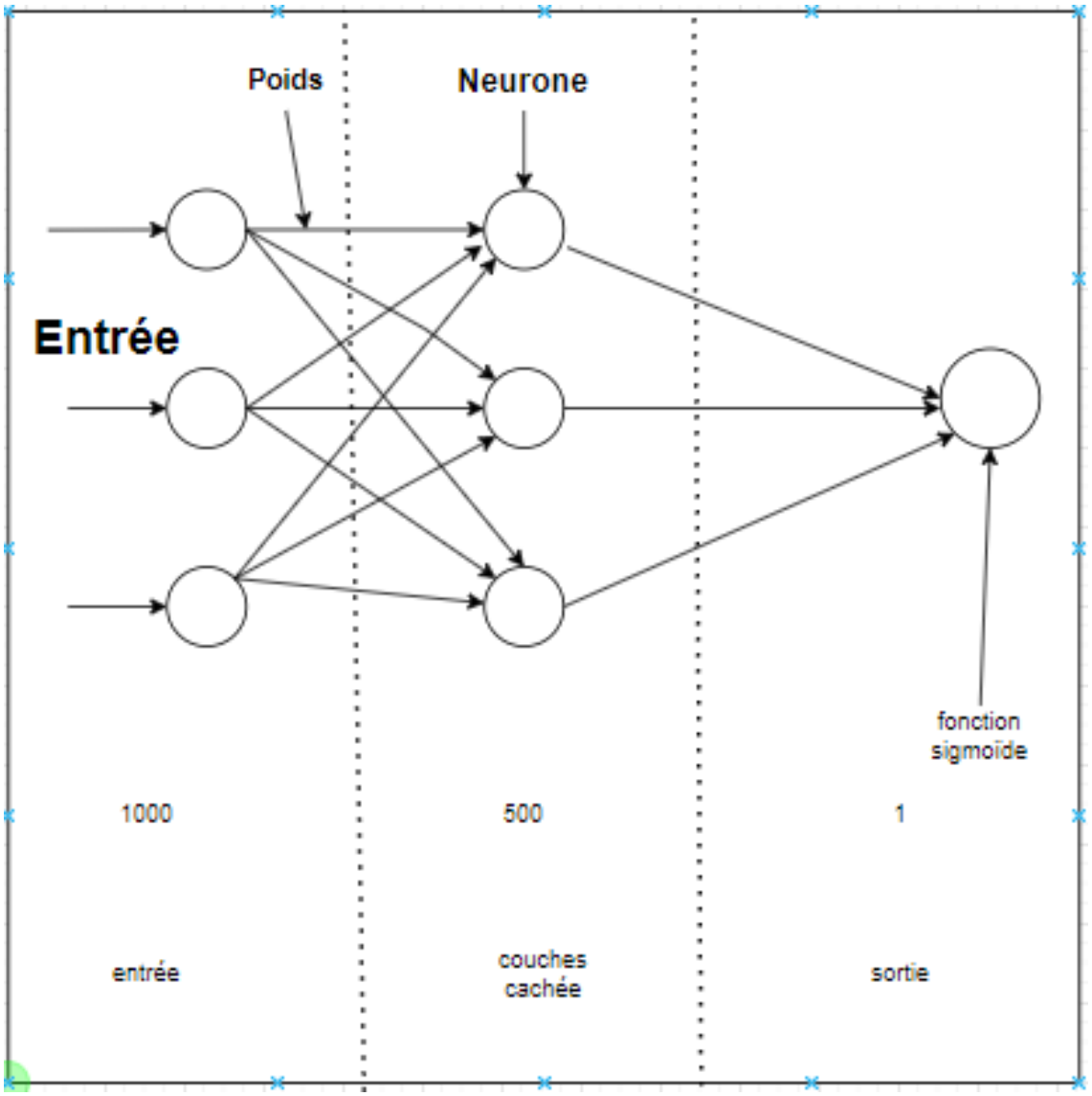

Figure 2.2 – l'architecture de l'ANN dans l'apprentissage

a) Les fonctions d'activation :

Nous allons utiliser les fonctions d'activation pour déterminer la sortie du réseau neuronal, comme oui ou non. Elle fait correspondre les valeurs résultantes à une valeur comprise entre 0 et 1 ou -1 et 1, etc.

b) Fonction d'activation ReLU (Rectified Linear Unit) :

La fonction ReLU est la fonction d'activation la plus utilisée dans le monde à l'heure actuelle, puisqu'elle est utilisée dans presque tous les réseaux neuronaux convolutifs ou l'apprentissage profond, voir la figure (2.3) [w7].

c) La fonction sigmoïde :

La principale raison pour laquelle nous utilisons la fonction sigmoïde est qu'elle existe entre 0 et 1. Par conséquent, elle est particulièrement utilisée pour les modèles où nous

devons prédire la probabilité en tant que sortie. Comme la probabilité de quoi que ce soit n'existe que dans la plage de 0 à 1, la fonction sigmoïde est le bon choix figure (2.3) [w8].

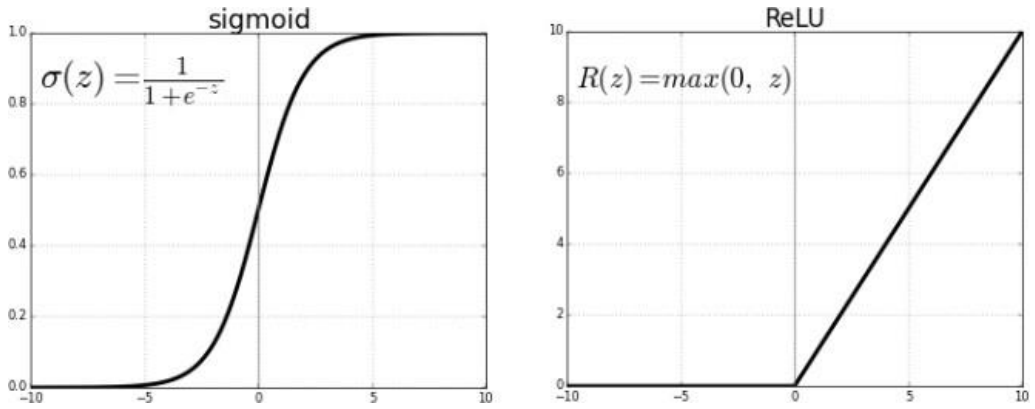

Figure 2.3 – ReLU v/s Logistic Sigmoid.

d) Le perceptron multicouche :

Le perceptron multicouche (PMC) est la deuxième grande famille de réseaux de neurones. Après avoir décrit l'architecture de ces réseaux on va aborder leur apprentissage, et le concept de rétro propagation de l'erreur. Un neurone perceptron effectue un produit scalaire entre son vecteur d'entrée x et un vecteur de paramètres w appelé poids, lui ajoute un biais b et utilise une fonction d'activation f pour déterminer sa sortie (équation 1).

1 : y = f (x . w + b)

Le perceptron est organisé en plusieurs couches. La première couche est reliée aux entrées, puis ensuite chaque couche est reliée à la couche précédente. C'est la dernière couche qui produit les sorties du PMC. Les sorties des autres couches ne sont pas visibles à l'extérieur du réseau, et elles sont appelées pour cette raison couches cachées.

2.4 Les réseaux antagonistes génératifs (GAN)

En raison de la taille modeste de notre ensemble de données (environ 600 lignes), nous avons utilisé GAN (Generative Adversarial Networks) pour générer d'autres données et augmenter la base.

Goodfellow et ses collègues ont proposé les GAN comme architecture d'apprentissage en profondeur en 2014 [w9].

Les GAN sont constitués de deux parties : un générateur et un discriminateur. Ils peuvent créer des données synthétiques à partir de zéro. Le générateur crée de fausses données à partir d'une entrée de bruit aléatoire ; le discriminateur détermine si les échantillons sont réels ou faux (produits par le générateur). Les performances du discriminateur sont utilisées pour mettre à jour et optimiser le générateur et le discriminateur.

Après avoir défini les modèles générateur et discriminateur, nous définirons le modèle Gan. C'est aussi un modèle séquentiel qui combine un discriminateur et un générateur.

a) Modèle d'un générateur

Le modèle aura trois couches, la fonction 'relu' activant deux d'entre elles. Puis, la couche de sortie sera déclenchée par la fonction 'linéaire', et sa dimension sera la même que la dimension de l'ensemble de données (9 colonnes).

b) Modèle de descripteur

Le discriminateur est également un modèle séquentiel simple à trois couches denses. La fonction 'relu' active les deux premières couches, tandis que la fonction 'sigmoïde' active la couche de sortie puisqu'elle déterminera si les échantillons d'entrée sont réels (True) ou faux (False).

2.5 Imputation des données

Les valeurs manquantes ou invalide ont un impact certain sur la qualité des résultats. L'imputation des données est le processus de remplacement des données manquantes, invalidequi ont échoué aux vérifications avec des valeurs substituées.

Dans le but d'imputer des valeurs nulles dans dataset Pima, nous avons utilisé le module le python fancyimput. il s'agit d'une bibliothèque de techniques d'imputation pour les données manquantes. Fancyimpute impute les valeurs manquantes à l'aide d'une méthode d'apprentissage automatique. Fancyimpute impute les valeurs manquantes dans toutes les colonnes. En utilisant Fancyimpute, les données manquantes peuvent être imputées de deux manières :

1. KNN (K-Nearest Neighborest) un algorithme d'apprentissage automatique supervisé utilisé pour résoudre les problèmes de classification et de régression.
2. MICE (Multiple Imputation by Chained Equation) : est une méthode robuste et informative de traitement des données manquantes dans les ensembles de données.

Dans notre modèle on a choisi les KNN car ils sont plus simples à utiliseret ils ont donné des meilleurs résultats.

2.5.1 Test et validation

Une fois le réseau de neurones entraîné il faut le tester sur une base de données différente de celle utilisée pour l'apprentissage. Pour évaluer les performances de l'ANN. Si les performances ne sont pas satisfaisantes, l'architecture du réseau doit être modifiée ou la bibliothèque d'apprentissage doit être modifiée.

2.6 Conclusion :

Dans ce chapitre on a présenté les différentes étapes d'analyse et la conception de notre modèle.

Dans le prochain chapitre on va parler de l'implémentation qui présente la partie pratique de notre travaille.

Chapitre 3

Implémentation

3.1 Introduction

L'objectif de ce chapitre est de présenter les étapes de l'implémentation de notre application de détection et prédiction de diabète.

D'abord, on va commencer par la présentation de notre application. Les ressources utilisées dans la création de l'application, et les interfaces graphiques. Ensuite, le traitement des données collectées, et les résultats obtenus. Ce chapitre est composé de trois parties à savoir : la présentation outils de développement, le traitement des données collectées, et les résultats obtenus.

3.2 Présentation des outils de développement

1. **Python:**

   Python est un langage de programmation simple mais puissant avec d'excellentes fonctionnalités pour le traitement des données linguistiques. Python peut être téléchargé gratuitement sur :

   http://www.python.org/.

   Nous avons choisi Python parce qu'il a une courbe d'apprentissage superficielle, sa syntaxe et sa sémantique sont transparentes, et il a une bonne fonctionnalité de gestion des chaînes. En tant que langage interprété, Python facilite l'exploration interactive. En tant que langage orienté objet, Python permet d'encapsuler et de réutiliser facilement les données et la méthode. En tant que langage dynamique, Python permet d'ajouter des attributs à des objets à la volée et de taper dynamiquement une variable, ce qui facilite le développement rapide. Python est livré avec une vaste bibliothèque standard, y compris des composants pour la programmation graphique, le traitement numérique et la connectivité.

   Python est très utilisé dans l'industrie, la recherche scientifique et l'éducation dans le monde entier. Python est souvent loué pour la façon dont il facilite la productivité, la qualité et la maintenabilité des logiciels [w10].

2. **Google Colab:**

   Est un produit de Google Research. Il permet à n'importe qui d'écrire et d'exécuter le code Python de son choix par le biais du navigateur. C'est un environnement particulièrement adapté à la machine learning, à l'analyse de données et à l'éducation. En termes plus techniques, colab est un service hébergé de notebooks Jupyter qui ne nécessite aucune configuration et permet d'accéder gratuitement à des ressources informatiques, dont des GPU. [w11].

3. **Pandas :**

   Pandas est une bibliothèque écrite pour le langage de programmation Python elle nous permet de manipuler et analysé les données. Elle propose en particulier des structures de données et des opérations de manipulation de tableaux numériques et de séries temporelles [w12].

4. **Scikit-learn :**

   Scikit-learn (anciennement scikits. Learn) et également connu sous le nomde sklearn) est une bibliothèque d'apprentissage automatique pour le langage de programmation Python.

   Elle comporte des divers algorithmes de classification, de régression et de clustering, notamment les machines vectorielles de support, les forêts aléatoires, l'amplification de gradient, kmeans et DBSCAN, et est conçu pour interagir avec les bibliothèques numériques et scientifiques Python NumPy et SciPy [w13].

5. **Keras :**

   Keras est une bibliothèque logicielle open source qui fournit une interface Python pour les réseaux de neurones artificiels et d'apprentissage automatique. Keras agit comme une interface pour la bibliothèque TensorFlow [w14].

   Keras est l'une des bibliothèques Python les plus puissantes et les plus faciles à utiliser pour développer et évaluer des modèles d'apprentissage en profondeur. Il enveloppe les bibliothèques de calcul numérique efficaces Theano et TensorFlow.

   L'avantage de ceci est principalement que vous pouvez commencer avec les réseaux de neurones de manière simple et amusante [w15] .

3.3 Résultats des étapes d'apprentissage et du test

À ce stade, on a divisé notre ensemble de données en deux sous-ensembles, où 80 % constitue l'ensemble d'apprentissage à 20 % l'ensemble de test. La figure (3.1) et la figure (3.2) présentent la division de notre dataset en données d'entrainement 602 lignes et données de test 191 lignes.

	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome
0	6	148	72	35	0	33.6	0.627	50	1
1	1	85	66	29	0	26.6	0.351	31	0
2	8	183	64	0	0	23.3	0.672	32	1
3	1	89	66	23	94	28.1	0.167	21	0
4	0	137	40	35	168	43.1	2.288	33	1
...	...	...	...	...	...	...	...	...	...
744	13	153	88	37	140	40.6	1.174	39	0
754	8	154	78	32	0	32.4	0.443	45	1
758	1	106	76	0	0	37.5	0.197	26	0
760	2	88	58	26	16	28.4	0.766	22	0
761	9	170	74	31	0	44.0	0.403	43	1

602 rows × 9 columns

Figure 3.1 : Données d'entrainement 602 lignes.

	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome
0	6	108	44	20	130	24.0	0.813	35	0
1	2	118	80	0	0	42.9	0.693	21	1
2	10	133	68	0	0	27.0	0.245	36	0
3	2	197	70	99	0	34.7	0.575	62	1
4	0	151	90	46	0	42.1	0.371	21	1
...	...	...	...	...	...	...	...	...	...
187	10	101	76	48	180	32.9	0.171	63	0
188	2	122	70	27	0	36.8	0.340	27	0
189	5	121	72	23	112	26.2	0.245	30	0
190	1	126	60	0	0	30.1	0.349	47	1
191	1	93	70	31	0	30.4	0.315	23	0

Figure 3.2 : Données de test 191 lignes

3.4 Génération de nouvelles données à partir de l'ensemble d'apprentissage :

Dans le but d'augmenter la taille du dataset on a pensé à créer un bruit aléatoire dans l'espace latent et le remodeler pour qu'il corresponde aux dimensions de l'entrée du modèle de générateur avec la fonction de génération de points latents suivant :

```
def generate_latent_points(latent_dim, n_samples):
    x_input = randn(latent_dim * n_samples)
    x_input = x_input.reshape(n_samples, latent_dim)
    return x_input
```

Figure 3.3 : Generation des points latents.

Pour générer de nouvelles données, nous définissons la fonction de génération de faux échantillons suivants :

```
def generate_fake_samples(generator, latent_dim, n_samples):
    x_input = generate_latent_points(latent_dim, n_samples)
    X = generator.predict(x_input)
    y = np.zeros((n_samples, 1))

    return X, y
```

Figure 3.4 : Génération des faux data.

Les points latents générés seront l'entrée du générateur (bruit aléatoire). Le générateur générera un tableau numpy à partir du bruit aléatoire d'entrée. L'étiquette sera 0 car il s'agit de fausses données. Nous avons crée une nouvelle fonction pour générer des échantillons réels, qui sélectionnera des échantillons de l'ensemble de données réel au hasard. Pour l'échantillon de données réelles, l'étiquette est 1.

```
def generate_real_samples(n):
    X = data.sample(n)
    y = np.ones((n, 1))
    return X, y
```

Figure 3.5 : Génération des données réelles.

Le model de générateur :

Le modèle aura trois couches, la fonction 'relu' activant deux d'entre elles. La couche de sortie sera déclenchée par la fonction 'linéaire'.

La couche d'entrée contient 15 neurone, et la deuxième les couches cachées contiennent 30 neurones, et la couche de sortie 9 norones.

Le modèle discriminateur :

Le discriminateur est également un modèle séquentiel simple à trois couches denses. La fonction 'relu' active les deux premières couches, tandis que la fonction 'sigmoïde' active la couche de sortie puisqu'elle déterminera si les échantillons d'entrée sont réels (True) ou faux (False).

```
Model: "sequential_2"
_________________________________________________________________
 Layer (type)                 Output Shape              Param #
=================================================================
 dense_6 (Dense)              (None, 15)                165

 dense_7 (Dense)              (None, 30)                480

 dense_8 (Dense)              (None, 9)                 279

=================================================================
Total params: 924
Trainable params: 924
Non-trainable params: 0
_________________________________________________________________
```

Figure 3.6 : Modèle générateur.

```
Model: "sequential_3"
_________________________________________________________________
 Layer (type)                 Output Shape              Param #
=================================================================
 dense_9 (Dense)              (None, 25)                250

 dense_10 (Dense)             (None, 50)                1300

 dense_11 (Dense)             (None, 1)                 51

_________________________________________________________________
```

Figure 3.7 : Modèle discriminateur.

Après avoir défini les modèles générateur et discriminateur, nous définirons le modèle GAN. C'est aussi un modèle séquentiel qui combine un discriminateur et un générateur.

Imputation:

Nous avons généré un dataset avec 1602 lignes (602 real data + 1000 generated data) donc nous avons imputer les données avec la bibliothèque fancyImpute en utilisant le code présenté dans la figure (3.8).

```
from fancyimpute import KNN
knn_imputer = KNN()
# imputing missing valuer avec knn imputer
df = knn_imputer.fit_transform(df)
```

Figure 3.8 : fancy impute.

Ensuite, nous avons entrainer le dataset avec le model suivant avec juste 50 itérations.

```
def nn():
    model = Sequential()
    model.add(Dense(1000, input_dim=8, activation='relu'))
    model.add(Dense(500, activation='relu'))
    model.add(Dense(1, activation='sigmoid')) # 1 output neuron
```

Figure 3.9 : Entrainement de modele.

Le résultat est présenté dans la matrice de confusion suivante : La matrice de confusion est une matrice qui mesure la qualité d'un systeme de classification.elle compare les données réelles pour une variable cible a celle prédites par un modele.

Accuracy:0.828
Classification Report

	precision	recall	f1-score	support
0	0.87	0.85	0.86	122
1	0.75	0.79	0.77	70
accuracy			0.83	192
macro avg	0.81	0.82	0.82	192
weighted avg	0.83	0.83	0.83	192

Figure 3.10 : Entrainement de modele.

3.4.1 Prédiction/ détection d'un nouveau cas de diabète

Dans le but de prédire si un nouveau patient est un futur diabétique, nous avons créé une interface qui permet d'introduire les valeurs des facteurs vitaux suivants à du patient à savoir : le poids, l'épaisseur de la peau, l'âge, taux d'insuline.etc. Puis le système retourne le résultat de prédiction. L'interface du système est présentée dans la figure (3.11).

3.4.2 Résultats des expérimentations

A cette étape, nous avons effectué une comparaison entre les résultats de l'approche du deep Learning que nous avons proposé et 2 autres algorithmes d'apprentissage automatique qui sont : SVM et KNN. Les métriques d'évaluation utilisées étaient : l'accuracy, la précision, le rappel et le F1 score. Les formules de calcul de la précision, le rappel et le F1 score sont présentées ci-dessous respectivement dans les équations (1, 2,3 et 4).

- **1 :** Précision=TP/(TP+FP)
- **2 :** Rappel =TP/(TP+FN)

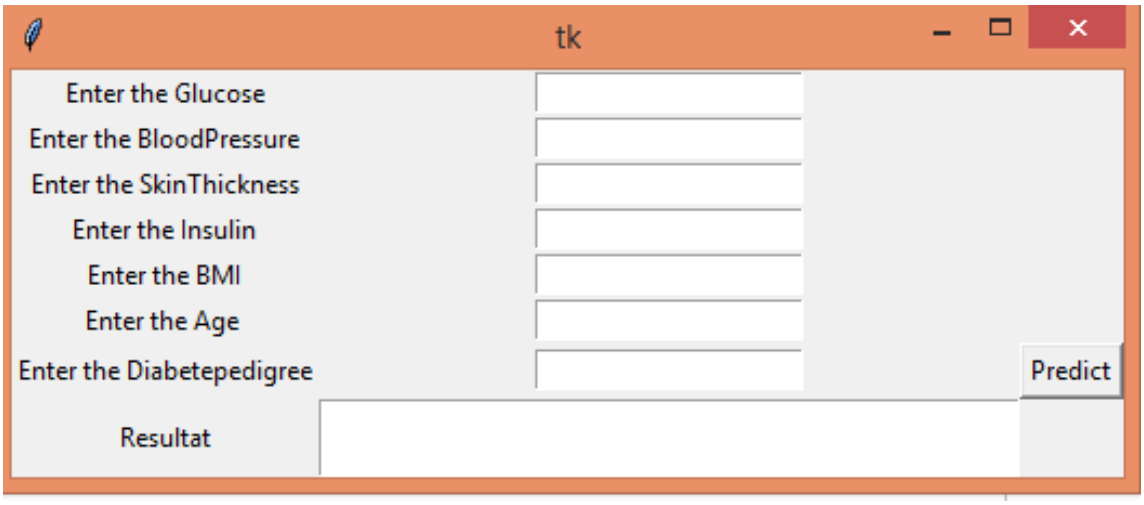

Figure 3.11 : Interface.

- **3 :** F1score=2 (précision.rappel)/(précision+rappel)=TP/(TP+1/2(FP+FN))
- **4 :** accuracy=(TP+TN) /(TP+TN+FP+FN)

  telle que :
- TP=nombre de vrai positif
- FN=nombre des faux négatifs
- TN= nombre de vrais négatifs
- FP= nombre de faux positif

En entrainant notre modèle avec SVM on a obtenu : un accuracy de 77 %, et avec les différents paramètres de KNN on a obtenu les résultats suivants :

- KNN 3 : un accuracy de 65%
- KNN 5 un accuracy de 66 %
- KNN 7 UN accuracy de 68 %
- KNN 10 un accuracy de 76 %

Le tableau et le graphe ci-dessus (figure 3.12 et tableau 3.1) présentent les résultats obtenus. Nous constatons que notre approche a généré les meilleurs résultats en comparaison avec ceux obtenus en appliquant les SVM et le KNN. Aussi, les résultats obtenus par le SVM sont meilleurs que ceux obtenus par le KNN.

	accuracy	recall	precision	F1 score
Notre approche	83%	82%	81%	82%
SVM	77%	72%	75%	73%
KNN 3	65%	65%	64%	64%
KNN 5	66%	67%	64%	64%
KNN 7	68%	67%	67%	67%
KNN 10	76%	76%	75%	76%

Table 3.1 : Resultats des experimentations.

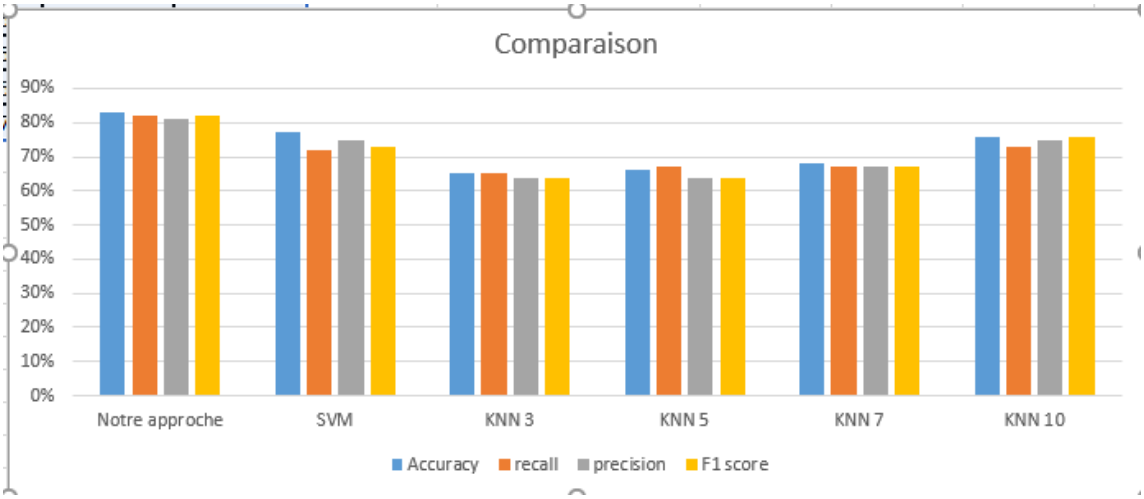

Figure 3.12 : Resultats des experimentations.

3.5 Conclusion

Dans ce chapitre, nous avons présenté les différentes étapes que nous avons menées pour parvenir au développement et au bon fonctionnement de notre système de prédiction du diabète. Nous voulons mentionner que notre approche de deep Learning a généré les meilleurs résultats (accuracy de 83%) en comparaison avec ceux obtenus en appliquant les SVM (77%) et le KNN (76%).

Conclusion générale

Une grande partie de la population humaine souffre du diabète. Si cette maladie n'est pas traitée, elle représentera un risque énorme pour le monde. Pour cela nous avons pensé à développer un système robuste sous la forme d'une application dans le but d'aider les spécialistes de la santé dans la détection précoce du diabète. Par conséquent, dans notre recherche proposée, nous avons mis en pratique divers classifieurs sur la base de diabète indienne PIMA. Les expérimentations nous a permis de prouver que l'exploration de données et l'algorithme d'apprentissage automatique peuvent réduire les facteurs de risque et améliorer les résultats de prédiction.

Bibliographie

[Agatonovic-Kustrin & Beresford, 2000] Agatonovic-Kustrin, S, & Beresford, Rosemary. 2000. Basic concepts of artificial neural network (ANN) modeling and its application in pharmaceutical research. *Journal of pharmaceutical and biomedical analysis*, **22**(5), 717–727.

[Bird *et al.*, 2009] Bird, Steven, Klein, Ewan, & Loper, Edward. 2009. *Natural Language Processing with Python*. Culemborg, Netherlands: Van Duuren Media.

[Caliskan *et al.*, 2018] Caliskan, Abdullah, Yuksel, Mehmet Emin, Badem, Hasan, & Basturk, Alper. 2018. Performance improvement of deep neural network classifiers by a simple training strategy. *Engineering Applications of Artificial Intelligence*, **67**, 14–23.

[Chen *et al.*, 2017] Chen, Wenqian, Chen, Shuyu, Zhang, Hancui, & Wu, Tianshu. 2017. A hybrid prediction model for type 2 diabetes using K-means and decision tree. *Pages 386–390 of: 2017 8th IEEE international conference on software engineering and service science (ICSESS)*. IEEE.

[Delannoy *et al.*, 2020] Delannoy, Quentin, Pham, Chi-Hieu, Cazorla, Clément, Tor-Díez, Carlos, Dollé, Guillaume, Meunier, Hélène, Bednarek, Nathalie, Fablet, Ronan, Passat, Nicolas, & Rousseau, François. 2020. Réseaux antagonistes génératifs pour la reconstruction super-résolution et la segmentation en IRM. *In: Extraction et Gestion des Connaissances-Atelier Apprentissage Profond: Théorie et Applications (APTA@ EGC)*.

[Deluzarche, 2021] Deluzarche, Céline. 2021 (05). *Deep Learning : qu'est-ce que c'est ?*

[El_Jerjawi & Abu-Naser, 2018] El_Jerjawi, Nesreen Samer, & Abu-Naser, Samy S. 2018. Diabetes prediction using artificial neural network. *International Journal of Advanced Science and Technology*, **121**.

[Gelly, 2017] Gelly, Gregory. 2017. *Reseaux de neurones recurrents pour le traitement automatique de la parole*. Ph.D. thesis, Université Paris Saclay (COmUE).

[Ghediri *et al.*, 2021] Ghediri, Rebahi, KhawLa, Semri, Kenza, Belhouchette, & Imane. 2021. La Reconnaissance des émotions de base par Les réseaux de neurones: application de deep Learning.

[Iyer *et al.*, 2015] Iyer, Aiswarya, Jeyalatha, S, & Sumbaly, Ronak. 2015. Diagnosis of diabetes using classification mining techniques. *arXiv preprint arXiv:1502.03774*.

[Kahramanli & Allahverdi, 2008a] Kahramanli, Humar, & Allahverdi, Novruz. 2008a. Design of a hybrid system for the diabetes and heart diseases. *Expert systems with applications*, **35**(1-2), 82–89.

[Kahramanli & Allahverdi, 2008b] Kahramanli, Humar, & Allahverdi, Novruz. 2008b. Design of a hybrid system for the diabetes and heart diseases. *Expert systems with applications*, **35**(1-2), 82–89.

[Kumar *et al.*, 2020] Kumar, N Komal, Sindhu, G Sarika, Prashanthi, D Krishna, & Sulthana, A Shaeen. 2020. Analysis and prediction of cardio vascular disease using machine learning classifiers. *Pages 15–21 of: 2020 6th International Conference on Advanced Computing and Communication Systems (ICACCS)*. IEEE.

[Naz & Ahuja, 2020a] Naz, Huma, & Ahuja, Sachin. 2020a. Deep learning approach for diabetes prediction using PIMA Indian dataset. *Journal of Diabetes & Metabolic Disorders*, **19**(1), 391–403.

[Naz & Ahuja, 2020b] Naz, Huma, & Ahuja, Sachin. 2020b. Deep learning approach for diabetes prediction using PIMA Indian dataset. *Journal of Diabetes & Metabolic Disorders*, **19**(1), 391–403.

[Naz & Ahuja, 2020c] Naz, Huma, & Ahuja, Sachin. 2020c. Deep learning approach for diabetes prediction using PIMA Indian dataset. *Journal of Diabetes & Metabolic Disorders*, **19**(1), 391–403.

[Prakash *et al.*, 2021] Prakash, Prem, Sebban, Marc, Habrard, Amaury, Barthelemy, Jean-Claude, Roche, Frédéric, & Pichot, Vincent. 2021. Détection automatique des apnées du sommeil sur l'ECG nocturne par un apprentissage profond en réseau de neurones récurrents (RNN). *Médecine du Sommeil*, **18**(1), 43–44.

[Valenchon, 2019] Valenchon, Juliette. 2019. *Graph-based machine learning algorithms for predicting disease outcomes*. McGill University (Canada).

Webographie

[W1] , magentaNadia, A. I. (2019-2020). UneapprocheIApourlareconnaissancedesexpressionsfac Algérie/Bouira. /, Dernier accès au site : 06/03/2022

[W2] , magentamémoirefind\T1\textquoterightetudeLaReconnaissanceDesémotionsdebaseParLe ApplicationdeDeepLearningPrésenterpar:RebahiGhediriImaneSemriKhawLa/, Dernier accès au site : 06/03/2022

[W3] , magentahttps://intelligence-artificielle.com/top-algortithme-deep-learning/amp//, Dernier accès au site : 06/03/2022

[W4] , magentahttps://doi.org/10.3390/electronics10121501/, Dernier accès au site : 06/03/2022

[W5] , magentaIntroductiontoartificialneuralnetworksenzogrossiEuropeanJournalofGastroe Hepatology/, Dernier accès au site : 06/03/2022

[W6] , magentahttps://www.novonordisk.com//, Dernier accès au site : 06/03/2022

[W7] , magentahttps://www.inside-machinelearning.com/fonction-dactivation-comment-ca-/, Dernier accès au site : 06/03/2022

[W8] , magentahttp://math.univ-lyon1.fr/~jberard/cours-www.pdf/, Dernier accès au site : 06/03/2022

[w9] , magentaHttps://arxiv.org/abs/1406.2661/, Dernier accès au site : 06/03/2022

[w10] magentahtarch./, Dernier accès au site : 06/06/2022

[w11] , magentahttps://research.google.com/colaboratory/faq.html?hl=fr/, Dernier accès au site : 06/06/2022

[w12] , magentahttps://fr.wikipedia.org/wiki/Pandas/, Dernier accès au site : 06/06/2022

[w13] , magentahttps://en.wikipedia.org/wiki/Scikit-learn/, Dernier accès au site : 06/06/2022

[w14] , magentaN.KetkarandE.Santana, Deeplearningwithpython. Springer, 2017, vol. 1. /, Dernier accès au site : 06/06/2022

[W15] , magentaF.Chollet, \T1\textquotedblleftBuildingautoencodersinkeras, \T1\textquotedblrightTheKerasBlog, vol. 14, 2016. /, Dernier accès au site : 06/06/2022