



République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Université de Bordj Bou Arreridj Mohammed El Bachir El Ibrahimi
Faculté des Mathématiques et d'Informatique
Département d'Informatique

Mémoire

En vue de l'obtention du diplôme de Master
Domaine : Mathématiques et Informatique
Filière : Informatique
Option : Ingénierie de l'Informatique Décisionnelle (IID)

Intitulé

**Un système de prédiction du diabète par les
réseaux de neurone artificiels**

Présenté par :

- Ibtissam TABET

Proposé et dirigé par :

-Dr. Manel SAAD SAOUD

Devant le jury composé de :

Année Universitaire 2020/2021

Dédicace

Au nom de Dieu Clément et Miséricordieux

Dieu soit loué, qui nous a éclairés sur le chemin de la connaissance, nous a aidés à accomplir ce devoir et nous a accordé le succès dans l'accomplissement de cette œuvre

*Je dédie ce modeste travail de fin de formation à mes très chers parents Ahmed et Fahima Bouchibane qui m'ont été un grand soutien moral dans les moments difficiles durant la formation
Ainsi qu'à ma chère sœur Afaf et mes frères (Billel et Adel) et leurs femmes et leurs enfants et à toute ma famille*

A mon mari Jugurtha Amarouche qui m'a soutenu de finir ce travail et à toute sa famille

A mon encadrante Dr. Saad Saoud Manel d'avoir dirigé ce travail.

A tous mes amis

Sans oublier tous mes enseignants pour leur contribution à ma formation

A tous qui ont contribué de près ou de loin à la réalisation de ce travail, qu'ils trouvent ici la traduction de ma gratitude et ma reconnaissance.

Ibtissam TABET

Remerciements

C'est avec un immense plaisir que nous tenons à remercier très sincèrement toutes les personnes qui nous ont aidé et qui ont ainsi contribué à la réalisation de ce mémoire.

Nous tenons à remercier notre encadrante Dr. Saad saoud manel d'avoir dirigé ce travail.

Nos vifs 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.

Nous souhaitons exprimer notre gratitude envers nos familles et tous nos amis pour leur soutien et encouragements tout au long de ce travail.

Enfin, nous voudrions également remercier toutes les personnes qui ont participé de près ou de loin à la réalisation de ce travail.

ملخص

يعتبر مرض السكري اليوم من أكثر الأمراض المزمنة شيوعاً، وبسبب بعض المضاعفات، فهو يعد من أكثر الأمراض فتناً في العالم و يعد الاكتشاف المبكر لمرض السكري مهماً جداً لعلاج الفوري لأنه يمكن أن يوقف تطور المرض.

يمكن أن تساعد الطريقة المقترحة في التنبؤ بظهور مرض السكري في المستقبل من خلال تحويل المهمة إلى مشكلة تصنيف، تم تصميم نموذجنا بشكل أساسي باستخدام الطبقات المخفية لشبكة عصبية عميقة.

في بحثنا قمنا بتركيب عدد من المعلمات واستخدمنا وظيفة فقدان الانتروبيا الثنائية، والتي حصلت على نموذج تنبؤ للشبكة العصبية العميقة بدقة عالية.

أظهرت النتائج التجريبية كفاءة نموذج التعلم العميق للتنبؤ بالسكري المقترحة أفضل دقة تدريب على مجموعات بيانات مرض السكري من هنود بيما.

تظهر النتائج التجريبية تحسينات نموذجنا المقترح مقارنة بأساليب الدولة الحديثة.

الكلمات المفتاحية: مرض السكري، التنبؤ، شبكة عصبية، الانتروبيا الثنائية، الدقة، التعلم العميق.

Abstract

Today, diabetes is one of the most common chronic diseases, and due to some complications, it is one of the most deadly diseases in the world. Early detection of diabetes is very important for its prompt treatment because it can stop the progression of the disease.

The proposed method can help predict the onset of diabetes in the future by turning the task into a classification problem, our model is mainly designed using the hidden layers of a deep neural network.

In our research, we fitted a number of parameters and used the binary entropy loss function, which obtained a deep neural network prediction model with high accuracy.

The experimental results showed the efficiency of the proposed diabetes prediction deep learning model with the best training accuracy on diabetes datasets from Pima Indians.

The experimental results show the improvements of our proposed model compared to the modern state methods.

Keywords: diabetes, prediction, neural network, binary entropy, accuracy, deep learning.

Résumé

Aujourd'hui, le diabète est l'une des maladies chroniques les plus courantes et, en raison de certaines complications, c'est l'une des maladies les plus mortelles au monde. La détection précoce du diabète est très importante pour son traitement rapide car elle peut arrêter la progression de la maladie.

La méthode proposée peut aider à prédire l'apparition du diabète dans le futur en transformant la tâche en problème de classification, notre modèle est principalement conçu en utilisant les couches cachées d'un réseau de neurones profonds.

Dans notre recherche, nous avons ajusté un certain nombre de paramètres et utilisé la fonction de perte d'entropie binaire, qui a obtenu un modèle de prédiction de réseau neuronal profond avec une grande précision.

Les résultats expérimentaux ont montré l'efficacité du modèle d'apprentissage en profondeur de prédiction du diabète proposé avec la meilleure précision d'entraînement sur les ensembles de données sur le diabète des Indiens Pima.

Les résultats expérimentaux montrent les améliorations de notre modèle proposé par rapport aux méthodes d'état modernes.

Mots-clés : diabète, prédiction, réseau de neurones, entropie binaire, précision, apprentissage profond.

Table des matières

Liste des figures

Liste des tableaux

Introduction générale

| | |
|----------------------------------|---|
| 1. Contexte | 1 |
| 2. Problématique..... | 2 |
| 3. Objectif et contribution..... | 2 |
| 4. Plan de mémoire | 2 |

Chapitre 01 : la prédiction du diabète

| | |
|--|----|
| 1. Introduction..... | 4 |
| 2. Définition du diabète..... | 4 |
| 3. Les différents types de diabète..... | 4 |
| 4. Le diagnostic..... | 5 |
| 5. Les symptômes courants du diabète..... | 5 |
| 6. Les complications du diabète..... | 6 |
| 7. Définition de la prédiction..... | 7 |
| 8. La prédiction du diabète..... | 7 |
| 8.1 Définition..... | 7 |
| 8.2 L'objectif..... | 7 |
| 9. Les séries chronologiques (temporelles) | 7 |
| 9.1 Définition..... | 7 |
| 9.2 Caractéristique d'une série chronologique..... | 8 |
| 10. Les techniques de prédiction..... | 9 |
| 10.1 Les méthodes qualitatives (subjectives) | 9 |
| 10.2 Les méthodes quantitatives (objectifs) | 10 |
| 11. Conclusion..... | 11 |

Chapitre II : Les réseaux de neurones artificiels

| | |
|---|----|
| 1. Introduction..... | 13 |
| 2. L'objectif..... | 13 |
| 3. Neurones formels (artificiels)..... | 13 |
| 4. Les réseaux de neurones artificiels..... | 14 |

| | |
|--|----|
| 4.1 Définition..... | 14 |
| 4.2 Architectures des réseaux de neurones..... | 15 |
| 4.3 Fonction d'activation..... | 17 |
| 4.4 4.4 L'apprentissage..... | 19 |
| 4.4.1 Les types d'apprentissage..... | 19 |
| 4.4.2 Les règle d'apprentissage..... | 20 |
| 5. Le perceptron multicouche..... | 21 |
| 5.1 Architecture..... | 21 |
| 5.2 Rétro-propagation..... | 23 |
| 6. Procédure de développement des réseaux de neurones..... | 24 |
| 7. conclusion..... | 26 |

Chapitre III : implémentation et résultats

| | |
|--|----|
| 1. Introduction..... | 28 |
| 2. Environnement et outils de travail..... | 28 |
| 2.1 Langage de programmation..... | 28 |
| 2.2 Bibliothèques Python..... | 28 |
| 2.3 Editeur de code..... | 30 |
| 3. Analyse exploratoire de données..... | 30 |
| 3.1. Distribution de données..... | 30 |
| 3.2. Séparation ou division de la base de données..... | 32 |
| 3.3 Apprentissage des réseaux de neurones..... | 32 |
| 3.4 Test & Validation..... | 34 |
| 4. Implémentation et résultats obtenus..... | 34 |
| 4.1 Description des différentes structures..... | 34 |
| 4.2 Description des différentes Procédures..... | 34 |
| 4.3 Résultats et discussion..... | 35 |
| 4.4 Prédiction du diabète..... | 36 |
| 5. conclusion..... | 36 |

Conclusion générale.....38

Bibliographie

Liste des figures

| | |
|---|----|
| Figure 1.5 : Symptômes de diabète..... | 6 |
| Figure 1.9.1 : Représentation graphique d'une série chronologique..... | 8 |
| Figure 1.10 : Les techniques de prédiction. | 9 |
| Figure 2.3 : Représentation graphique d'un neurone. | 14 |
| Figure 2.4.2.1 : réseau à une couche. | 15 |
| Figure 2.4.2.2 : un réseau multicouche. | 16 |
| Figure 2.4.2.3 : Exemple d'un réseau récursif..... | 16 |
| Figure 2.4.2.4 : Structure du réseau de neurones 'Lartice'..... | 17 |
| Figure 2.4.3.1 : Fonction d'activation..... | 18 |
| Figure 2.4.3.2 : Fonction d'activation binaire à seuil..... | 18 |
| Figure 2.4.3.3 : Fonction d'activation à rampe avec saturation..... | 18 |
| Figure 2.4.3.4 : Fonction d'activation sigmoïde..... | 19 |
| Figure 2.5.1.1 : Tangente hyperbolique..... | 22 |
| Figure 2.5.1.2 : Schéma d'un PMC..... | 22 |
| Figure 2.5.2.1 : Principe de la rétro-propagation..... | 24 |
| Figure 3.2.2 : Bibliothèques Python..... | 28 |
| Figure 3.2.3 : Editeur de code Sublime Text..... | 30 |
| Figure 3.3.1 : l'ensemble de données "Pima Indian Diabetes"..... | 31 |
| Figure 3.4.3 : Visualisation des performances du modèle entraîné..... | 35 |
| Figure 3.4.4 : Exemple de détection d'une fausse information..... | 36 |

Liste des tableaux

| | |
|---|----|
| Tableau 3.3.1: Description des caractéristiques du l'ensemble de données "Pima Indian Diabetes"..... | 31 |
| Tableau 3.4.2 : Principales procédures de notre application..... | 34 |

Introduction générale

1. Contexte :

La population diabétique mondiale ne cesse d'augmenter, en 1998 était de 150 millions, ce chiffre doublera en 2025.

Cette épidémie est liée à plusieurs facteurs dont le vieillissement de la population, les régimes hypercaloriques, l'obésité et les changements de mode de vie dominés par la sédentarité.

Il existe une extrême hétérogénéité de la prévalence du diabète d'un pays à l'autre, L'Algérie est en pleine transition épidémiologique et le diabète pose un vrai problème de santé publique par le biais des complications chroniques dominées par les complications cardiovasculaires, le pied diabétique, l'insuffisance rénale chronique et la rétinopathie.

Selon une enquête de l'institut national de santé publique le diabète occupe la quatrième place dans les maladies chroniques non transmissibles.

Actuellement la sensibilisation de la population par l'identification des facteurs de risque qui peuvent être à l'origine du diabète, fait l'objectif des différents acteurs dans le domaine de la santé publique.

Cependant, la reconnaissance et l'identification de ces facteurs repose généralement sur des études faites sur une grande population.

Ces études sont regroupées sous forme des bases de données informatisées dans les hôpitaux et les instituts médicaux.

Chaque fois la taille de ses bases de données médicales est grande, l'analyse visuelle et l'exploitation de ses données devient très complexe pour les experts humains. Pour cette raison des techniques dites intelligentes d'extraction et d'analyse de données ont été utilisées.

L'utilisation de systèmes de classification pour le diagnostic médical est en développement progressif. Il n'y a aucun doute que l'évaluation des données du patient et les décisions des experts sont les facteurs les plus importants dans le diagnostic. Mais, les systèmes experts et les différentes techniques d'intelligence artificielle ont prouvé dans les dernières années leurs efficacités d'aider les experts dans le domaine médical.

Problématique :

Le diabète est connu depuis l'antiquité comme un trouble avec ruine de miel. La déficience de la prise en charge provoque des complications particulièrement en limitant l'activité de la personne malade et conduit à la mort. Il est un problème majeur de santé publique à l'échelle mondiale. Son évolution est silencieuse et insidieuse jusqu'à l'apparition de complications lourdes de conséquences en terme de morbidité et de mortalité.

Devant cette situation inquiétante, il est devenu indispensable d'étudier et comprendre ce phénomène pour en trouver des solutions techniques performantes et efficaces.

2. Objectif et contribution :

Dans ce projet, notre objectif est d'explorer le domaine de diabète, en particulier la tâche de prédiction du diabète. Nous avons également exploité les réseaux de neurone artificiels pour développer un système de prédiction du diabète.

3. Plan de mémoire :

La suite de ce mémoire est organisée en trois chapitres :

Le premier chapitre consistera en la présentation de la prédiction du diabète, Nous parlerons d'abord du diabète ses différents types, le diagnostic, ses symptômes courants, ses complications, Ensuite, Nous allons faire une simple explication de la prédiction et ses techniques.

Le deuxième portera sur les réseaux de neurone artificiels, nous concentrerons sur l'apprentissage supervisée et nous expliquerons l'architectures des réseaux de neurones, les types et le règles d'apprentissage.

Enfin, dans le quatrième et dernier chapitre, nous présenterons les résultats obtenus par notre modèle après avoir mené une analyse de données. Puis, nous dresserons une conclusion générale du projet.

Chapitre I

La pr diction du diab te

1. Introduction :

Le diabète est parmi les maladies les plus répandues à travers le monde.

Il représente un groupe hétérogène de maladies métaboliques et constitue un véritable problème de santé publique dans le monde. Il touche environ 422 millions de personnes, soit environ 4% de la population mondiale et on s'attend à une augmentation de 5,4% d'ici 2025.

Il est responsable de 9% de la mortalité totale, tuant 4 millions de malades chaque année.

Cette maladie pousse les humains à rechercher toutes sortes de moyens pour préserver leur santé et à savoir ce qui pourrait leur arriver dans le future en se basant sur plusieurs facteurs tels que l'analyse et l'étude des données passées.

2. Définition du diabète :

« Le diabète est une maladie chronique qui survient lorsque le pancréas ne produit pas assez d'insuline ou lorsque l'organisme n'est pas capable d'utiliser efficacement l'insuline qu'il produit. Il en résulte une concentration accrue de glucose dans le sang (hyperglycémie) » [1].

3. Les différents types de diabète :

- **Le diabète de type 1 :**

Il est également appelé diabète insulino-dépendant ou diabète juvénile. Le corps ne produit pas d'insuline ou en quantités insuffisante. Cette maladie auto-immune peut toucher des personnes de tout âge mais elle est généralement diagnostiquée chez les enfants ou les jeunes adultes. Le diabète de type 1 représente 5 à 15 % des cas de diabète. Son incidence augmente à un taux relativement stable. [3]

Les diabétiques de type 1 doivent s'injecter quotidiennement de l'insuline.

- **Le diabète de type 2 :**

Le diabète de type 2 est dû à une résistance à l'insuline entraînant une carence insulino-dépendante à terme. Les personnes diabétiques de type 2 peuvent présenter les mêmes symptômes que les diabétiques de type 1, mais il se peut aussi que les symptômes ne soient pas

Chapitre I : La prédiction du diabète

identifiés ; la maladie peut rester non-diagnostiquée pendant des années après son apparition. Par conséquent, on estime que près de la moitié des diabétiques de type 2 s'ignorent. [3]

Il se déclare généralement chez les personnes de plus de 40, il représente 85 à 95 % des cas de diabète dans les pays développés.

C'est une maladie évolutive, Le traitement évolue donc tout au long de la vie du patient. Au départ, une bonne hygiène alimentaire peut suffire, puis il faut parfois passer aux comprimés et enfin aux injections d'insuline. [3]

- **Le diabète gestationnel :**

« Le diabète gestationnel apparaît pendant la grossesse chez la femme n'ayant jamais eu de diabète auparavant. Il présente un risque pour la mère et l'enfant car il augmente pour les deux la probabilité de développer un diabète de type 2 par la suite, Il est le plus souvent détecté lors d'un dépistage prénatal et non pas parce les symptômes ». [3]

4. Le diagnostic :

« Le diagnostic de diabète se fait sur la mesure de la glycémie au laboratoire d'analyses médicales, lors d'une prise de sang. » [8]

Il s'effectue le plus souvent lorsque les premières complications surviennent. Un délai moyen de 7 ans environ s'écoule entre le moment où une glycémie est anormalement élevée sans avoir été repérée et le diagnostic. Le diagnostic est simplement fait par une prise de sang montrant une hyperglycémie à plus de 1,20g/l constatée à au moins 2 reprises. Le dosage sanguin de l'hémoglobine glyquée est devenu le meilleur indice de surveillance du diabète et des traitements antidiabétiques. L'hémoglobine glyquée ou HbA1C évalue la glycémie moyenne des 3 à 4 mois précédant le bilan. [8]

5. Les symptômes courants du diabète :

L'apparition du diabète accompagne des symptômes suivants :

- Troubles de la vision.
- Perte de poids.
- Somnolence.

Chapitre I : La prédiction du diabète

- Envie fréquente d'uriner.
- Soif intense.
- Langue sèche.
- Douleurs abdominales.
- Malaise et nausées.
- La fatigue.
- La faiblesse.



Figure 1.5 : Symptômes de diabète.

6. Les complications du diabète :

Les complications du diabète sont nombreuses et peuvent être sévères. La majorité des complications liées au diabète peuvent être évitées, réduites ou retardées si le diabète est dépisté et traité précocement et correctement. Les principales complications du diabète sont :

- La rétinopathie diabétique (responsable à terme d'une cécité)
- Les complications cardio-vasculaires,
- La néphropathie diabétique aboutissant à l'insuffisance rénale, la neuropathie diabétique, les infections, les ulcères de pied et de jambe. [8]

7. Définition de la prédiction :

La prédiction est une technique qui utilise des données historiques comme entrées pour faire des estimations éclairées qui sont prédictives pour déterminer la direction des tendances futures. [9]

Le principal objectif de prévision est de réduire l'incertitude liée à la non connaissance du futur.

8. La prédiction du diabète :

8.1 Définition :

La prédiction du diabète consiste à prévoir le risque de développer un diabète au future, en utilisant les données passées qu'on a à notre disposition.

8.2 L'objectif:

On utilise la prédiction du diabète pour Fournir un score de risque de diabète clinique simple et identifier les caractéristiques qui prédisent le diabète plus tard à l'aide de variables disponibles dans le contexte clinique ainsi que de variables biologiques et de polymorphismes.

9. Les séries chronologiques (temporelles) :

9.1 Définition : « une série chronologique ou une chronique est une suite d'observations ordonnées dans le temps prises à des intervalles réguliers » [6].

Les données d'une série chronologique sont notées X_t avec $t=1, 2, \dots, T$ La valeur d'un modèle de prédiction pour la période t est notée par Y_t :

- Pour $t \leq T$: Y_t représente la simulation de l'historique.
- Pour $t > T$: Y_t représente les valeurs de prévision.

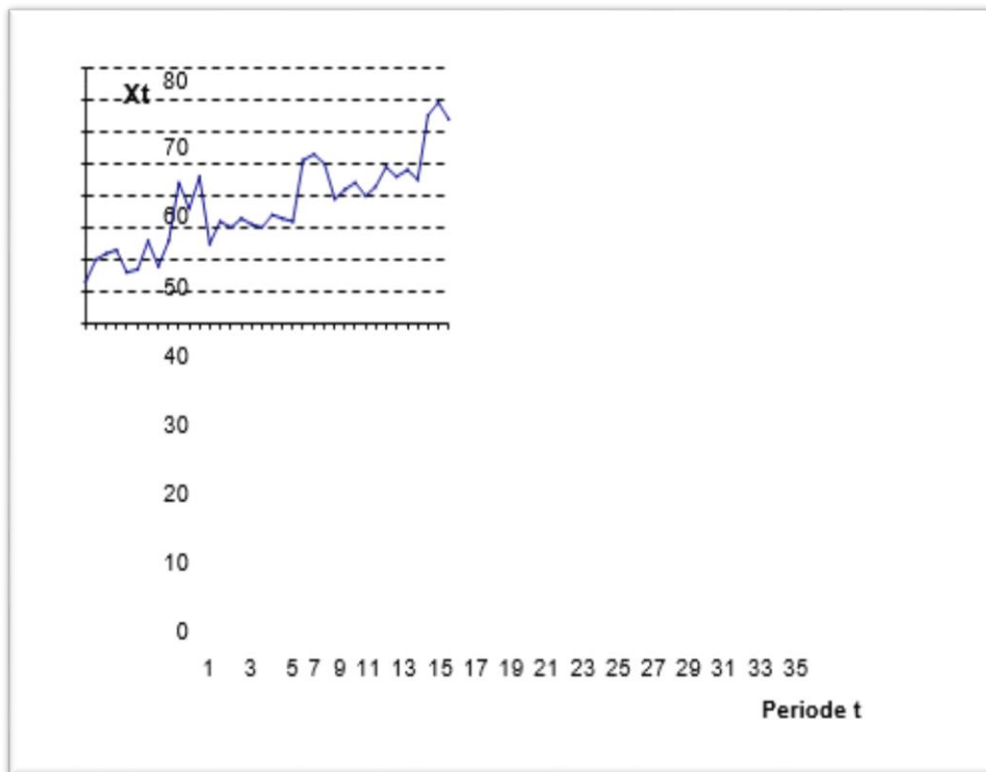


Figure 1.9.1 : Représentation graphique d'une série chronologique.

9.2 Caractéristique d'une série chronologique :

On isole habituellement trois composants dans les séries chronologiques :

- **La tendance** : caractéristique d'un phénomène à démontrer un patron stable dans un sens déterminé dans le temps. Le patron peut être linéaire (modélisé par une droite) ou non linéaire (logarithmique, exponentiel...). [6]
- **La saisonnalité** : caractéristique d'un phénomène qui se répète à intervalle fixé, par exemple à tous les hivers, à tous les mois, etc... [6]
- **Variation ponctuelle** : ce sont des variations dues à des circonstances exceptionnelles. (Comme conditions climatiques, exceptionnelles, grèves...). Ces phénomènes doivent être corrigés pour garantir la qualité du modèle de prévision. [6]

10. Les techniques de prédiction :

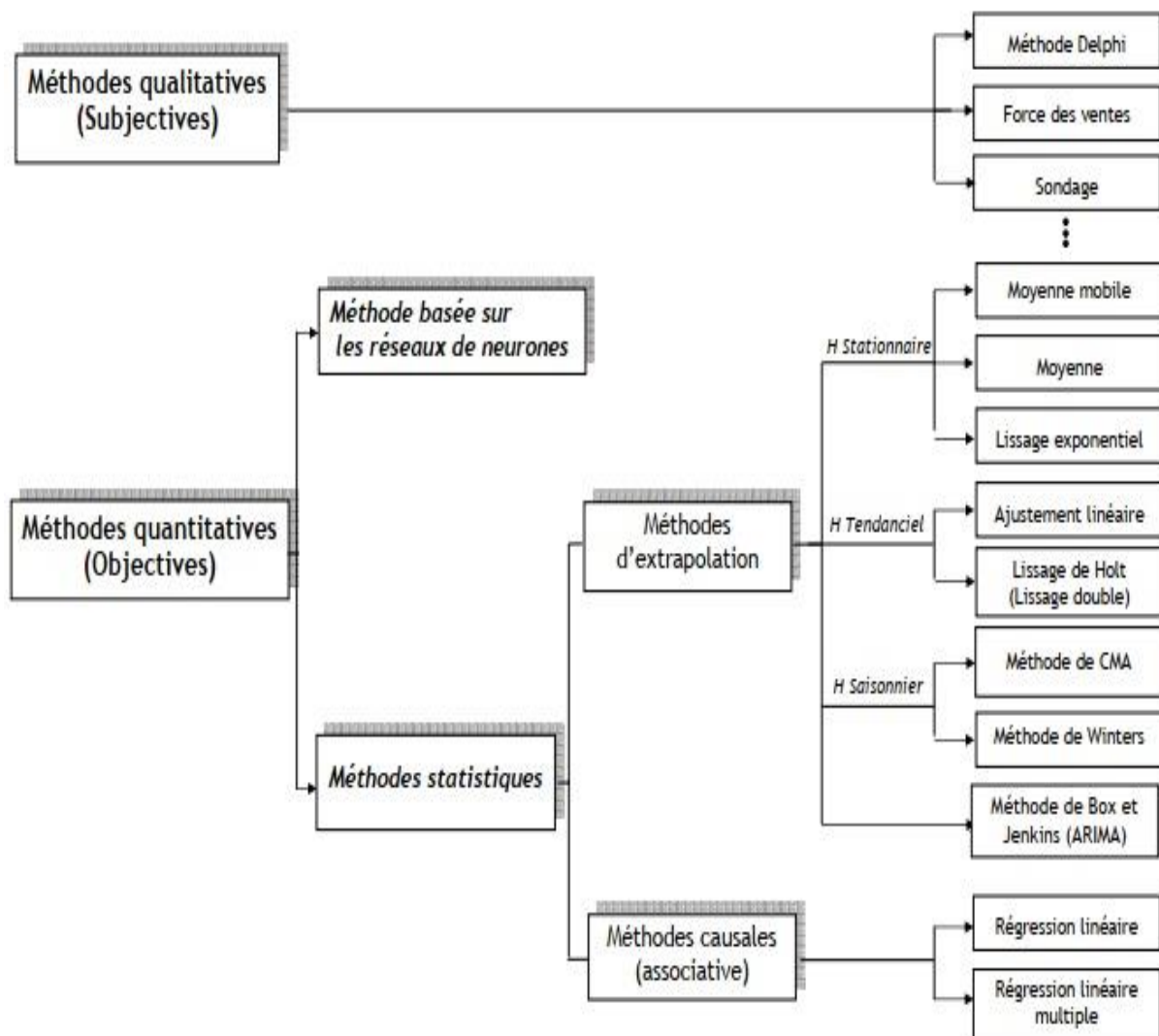


Figure 1.10 : Les techniques de prédiction.

10.1 Les méthodes qualitatives (subjectives) :

Elles sont basées sur le jugement humain et l'expérience professionnelle des responsables de domaine, les méthodes les plus utilisées sont :

- **Méthode Delphi** : Cette méthode est couramment utilisée pour prévoir les tendances sur la base des informations fournies par un panel d'experts. Cette série d'étapes est basée sur la méthode Delphi, qui fait référence à l'Oracle de Delphi. Il suppose que les

réponses d'un groupe sont plus utiles et impartiales que les réponses fournies par une seule personne. [6]

- **Force de ventes.**
- **Sondage.**
- **Etude de marché.**

Ces techniques exploitent des données non chiffrables et difficiles à décrire numériquement.

10.2 Les méthodes quantitatives (objectifs) :

« Les méthodes quantitatives se basent sur l'étude de l'historique (les données de passé), elles font l'hypothèse que le comportement des données du passé va se poursuivre dans le futur ». [6]

On distingue deux catégories dans les méthodes quantitatives:

10.2.1 Les méthodes d'extrapolation:

« Ces techniques recherchent à déterminer l'avenir de la variable à prévoir à partir de l'analyse des données du passé concernant cette même variable » [6]

On peut classer ces méthodes en fonction de l'historique à modéliser :

- Historique stationnaire.
- Historique avec tendance.
- Historique avec saisonnalité.

10.2.2 Les méthodes causales (ou associative) :

« Sur la base de donnée passée, elle cherche à établir une relation entre la variable à prévoir (variable expliquée) et une ou plusieurs variables (variables explicatives) ». [6]

10.2.3 La méthode basée sur les réseaux de neurones artificiels :

« Les réseaux de neurones ont prouvé d'être parmi les meilleures méthodes qui détectent les relations cachées entre les variables, une fois le réseau analyse ces données (l'apprentissage du réseau), il devient capable de générer des prévisions à partir les dépendances trouvées entre la variable à prévoir et les autres variables ». [6]

Les entreprises du domaine de la technologie utilisent des méthodes d'intelligence artificielle (IA) pour prévoir un domaine de croissance spécifique. Cette méthode de prévision

Chapitre I : La prédiction du diabète

fournit des résultats extrêmement précis à l'aide d'algorithmes mathématiques. La science derrière l'intelligence artificielle prédit de nombreux résultats pour les utilisateurs et aide à générer les suggestions qui apparaissent sur certains sites. [6]

11. Conclusion :

Dans ce chapitre on a parlé tout d'abord sur diabète : sa définition et ses différents types ainsi le diagnostic, les symptômes courants et les complications.

Dans une deuxième partie sur la prédiction, dont la prédiction du diabète : sa définition et leur objectif, nous avons cité les séries chronologiques et les techniques de prédiction qui comprennent les réseaux de neurones artificiels que nous avons choisis comme meilleure solution pour la prédiction du diabète.

Chapitre II

Les réseaux de neurones artificiels

1. Introduction :

La conception structurelle du cerveau humain est la principale inspiration derrière la condensation du modèle. Les réseaux de neurones artificiels sont utilisés de manière productive dans diverses disciplines telles que Sciences de l'environnement, étude de l'esprit humain, étude des ordinateurs ... etc. Il est également utilisé dans de nombreux domaines tels que la comptabilité, la gestion, la prise de décision, la fabrication... etc. Il est devenu un modèle populaire et récemment utilisé pour identifier les maladies et prédire la survie des patients.

Dans le but de prédire le diabète, nous présentons dans ce chapitre une explication générale sur les réseaux de neurone artificiels et la conception de prédiction du diabète basé sur l'apprentissage automatique supervisé.

2. L'objectif :

Le diabète est une maladie chronique caractérisée par une glycémie élevée. Cela peut causer de nombreuses maladies compliquées comme un accident vasculaire cérébral, une insuffisance rénale, une crise cardiaque, etc. L'objectif principal de cette étude est de développer un système basé sur des réseaux de neurones artificiels pour prédire les patients diabétiques.

3. Neurones formels (artificiels) :

« Le neurone artificiel est un processeur élémentaire. Il reçoit un nombre variable d'entrées en provenance de neurones appartenant à un niveau situé en amont. A chacune des entrées est associé un poids w représentatif de la force de la connexion. Chaque processeur élémentaire est doté d'une sortie unique, qui se ramifie ensuite pour alimenter un nombre variable de neurones appartenant à un niveau situé en aval. A chaque connexion est associé un poids. » [2]

La modélisation consiste à mettre en œuvre un système de réseaux neuronaux sous un aspect non pas biologique mais artificiel, cela suppose que d'après le principe biologique on aura une correspondance pour chaque élément composant le neurone biologique, donc une modélisation pour chacun d'entre eux. [2]

Un neurone se compose généralement d'une entrée formée des variables sur lesquelles opèrent ce neurone et une sortie représentant la valeur de la fonction réalisée

Chapitre II : Les réseaux de neurones artificiels

(fonction d'activation). La sortie du neurone est une fonction non linéaire d'une combinaison des entrées x_i (signaux d'entrées) pondérées par les paramètres w_i (poids synaptiques). [2]

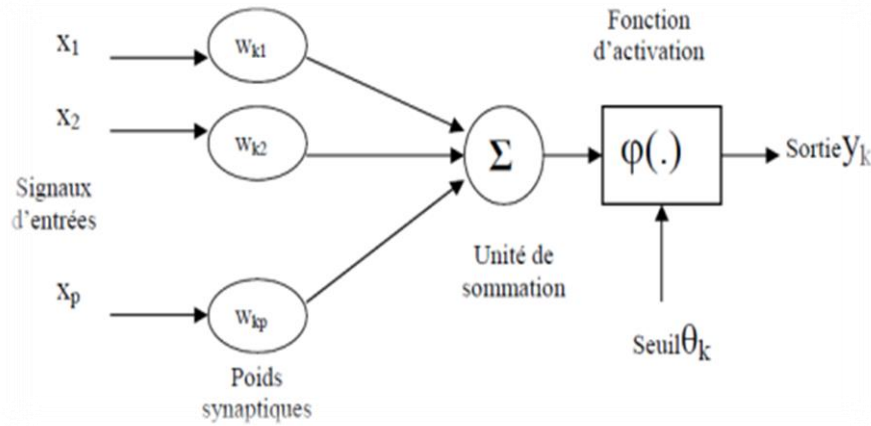


Figure 2.3 : Représentation graphique d'un neurone.

Mathématiquement le neurone est une fonction algébrique non linéaire, paramétrée, à valeurs bornées. La sortie du neurone est donnée par l'expression suivante :

$$y_k = \phi \left(\sum_{j=1}^p w_{kj} \cdot x_j - \theta_k \right)$$

où : $x_1, x_2, x_3, \dots, x_p$: entrées,

$w_{k1}, w_{k2}, w_{k3}, \dots, w_{kp}$: Poids synaptiques du neurone k ,

θ_k : Le Seuil,

$\phi(\cdot)$: La fonction d'activation,

y_k : est la sortie d'activation.

4. Les réseaux de neurones artificiels :

4.1 Définition :

Un réseau neuronal artificiel (ANN) est l'élément d'un système informatique conçu pour simuler la façon dont le cerveau humain analyse et traite les informations. C'est le fondement de l'intelligence artificielle (IA) et résout des problèmes qui s'avèreraient impossibles ou difficiles selon les normes humaines ou statistiques. Les ANN ont des capacités d'auto-

Chapitre II : Les réseaux de neurones artificiels

apprentissage qui leur permettent de produire de meilleurs résultats à mesure que davantage de données deviennent disponibles.

4.2 Architectures des réseaux de neurones :

- Réseau à une couche :

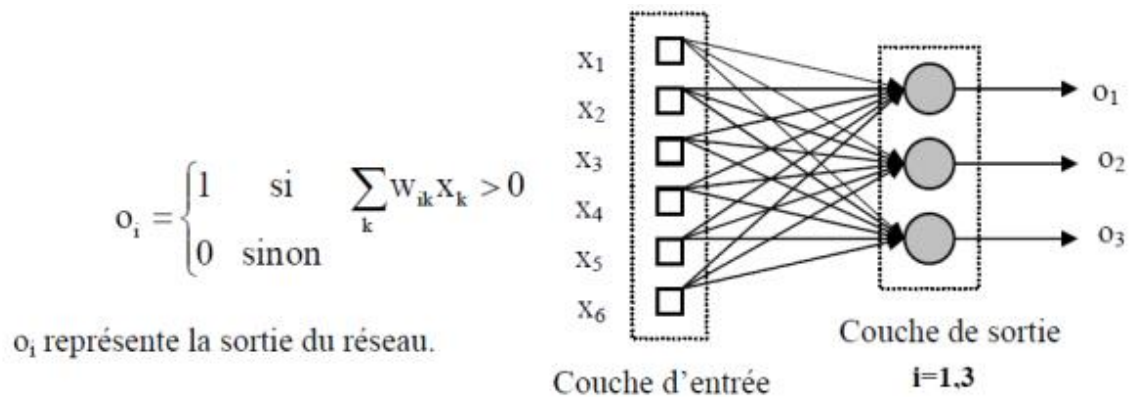


Figure 2.4.2.1 : réseau à une couche.

- Les réseaux multicouches :

Il existe :

- 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.

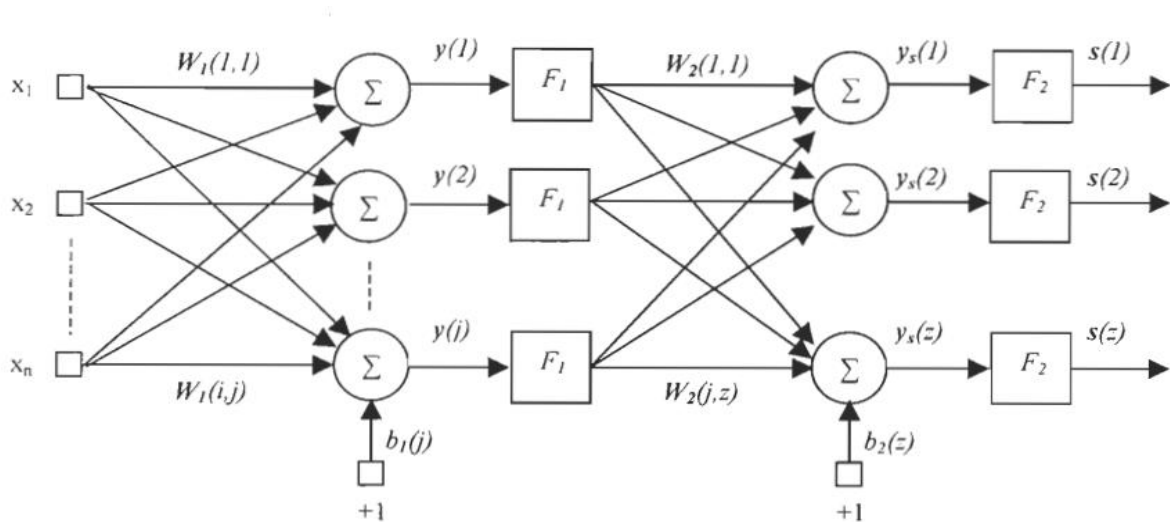


Figure 2.4.2.2 : un réseau multicouche.

- Les réseaux récurrentifs

Les réseaux récurrentifs se distinguent des autres réseaux par la connexion des sorties de neurones avec leurs entrées. La sortie d'un neurone peut être connectée avec l'entrée du même neurone ou avec celles des autres neurones. L'importance de ces réseaux est qu'ils permettent d'apprendre la dynamique de systèmes, c'est-à-dire qu'ils peuvent imiter le comportement temporel en insérant des délais dans les boucles, reliant l'entrée à la sortie du réseau ou dans des couches internes.

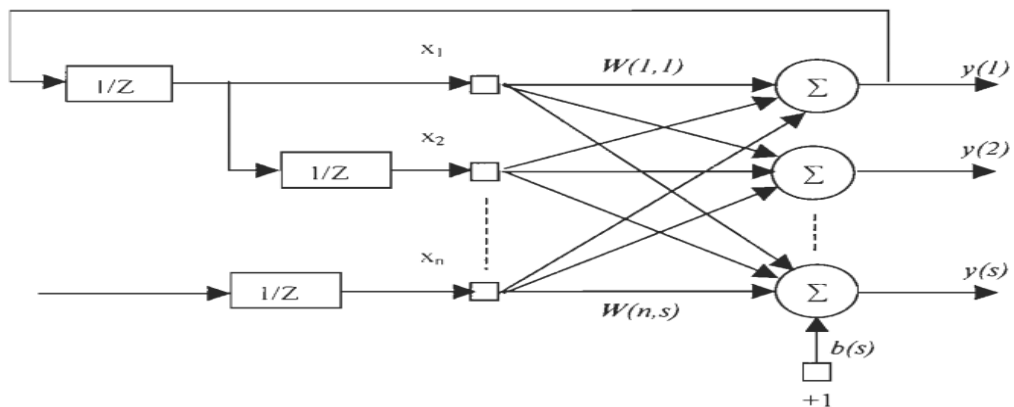


Figure 2.4.2.3 : Exemple d'un réseau récurrentif.

- **Structure 'Lartice' :**

C'est un réseau dont les neurones sont organisés en lignes et en colonnes, où toutes les entrées de neurones sont reliées aux nœuds d'entrées. Ces réseaux peuvent être d'une, de deux ou de plusieurs dimensions.

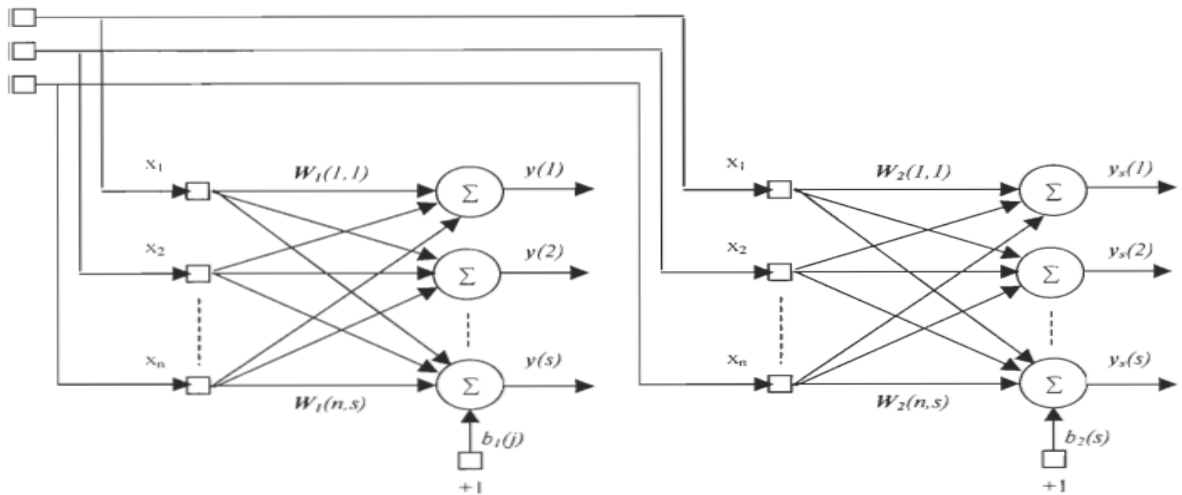


Figure 2.4.2.4 : Structure du réseau de neurones 'Lartice'.

4.3 Fonction d'activation

La fonction d'activation (ou fonction de seuillage, ou encore fonction de transfert) sert à introduire une non-linéarité dans le fonctionnement du neurone. [10]

Les fonctions de seuillage présentent généralement trois intervalles :

En dessous du seuil : le neurone est non-actif (souvent dans ce cas, sa sortie vaut 0 ou -1).

Aux alentours du seuil : une phase de transition.

Au-dessus du seuil : le neurone est actif (souvent dans ce cas, sa sortie vaut 1).

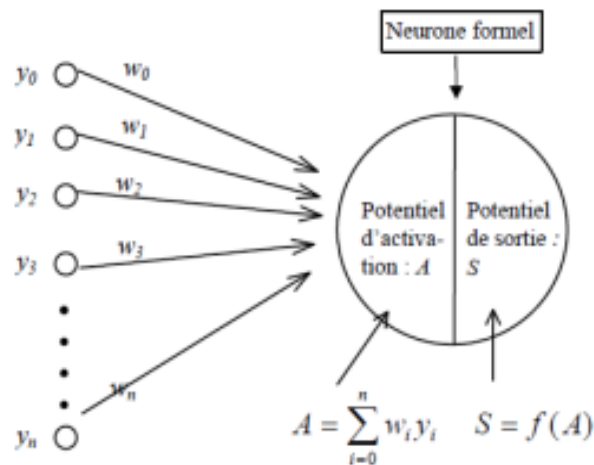


Figure 2.4.3.1 : Fonction d'activation.

On peut identifier trois principaux types de fonctions les plus connues:

- **Binaire à seuil :**



Figure 2.4.3.2 : Fonction d'activation binaire à seuil.

- **Rampe avec saturation :** Cette fonction représente un compromis entre la fonction linéaire et la fonction seuil: entre ses deux bornes, elle confère au neurone une combinaison linéaire de l'entrée. À la limite, la fonction linéaire est équivalente à la fonction seuil.

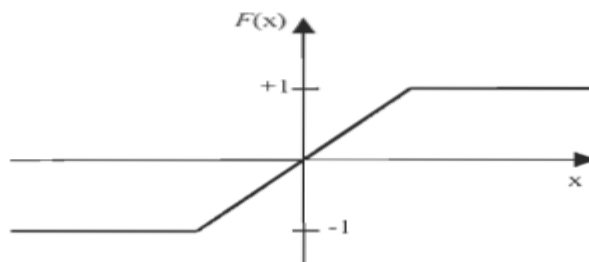


Figure 2.4.3.3 : Fonction d'activation à rampe avec saturation.

- **Fonction sigmoïde:** La fonction sigmoïde est une fonction continue qui maintient la sortie dans l'intervalle $[0,1]$. Son avantage principal est l'existence de sa dérivée en tout point. Elle est employée en général dans le perceptron multicouche.

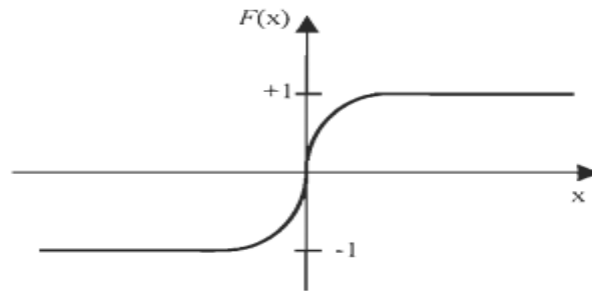


Figure 2.4.3.4 : Fonction d'activation sigmoïde.

4.4 L'apprentissage :

Une caractéristique des réseaux de neurones est leur capacité à apprendre (par exemple à reconnaître une lettre, un son...). 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.

4.4.1 Les types d'apprentissage :

On distingue usuellement au moins trois types d'apprentissage machine : l'apprentissage par renforcement, l'apprentissage supervisé et l'apprentissage non supervisé.

L'apprentissage par renforcement suppose que, lors de ses pérégrinations, un agent (entité qui agit de façon autonome) reçoit des récompenses ou des punitions en fonction des actions qu'il exécute. Il s'agit alors d'établir automatiquement, à partir des retours d'expérience, des stratégies d'action des agents qui maximisent l'espérance de récompenses. Ces techniques développées depuis la fin des années 1950 ont fait leurs preuves à la fois dans le domaine des jeux et dans celui de la robotique. [11]

L'apprentissage supervisé suppose que l'on donne des exemples étiquetés, comme des images de lettres manuscrites avec le nom de la lettre correspondante (étiquettes a, b, Z...). L'apprentissage consiste alors à construire une fonction capable de déterminer la lettre de

Chapitre II : Les réseaux de neurones artificiels

l'alphabet à laquelle se rapporte chaque image. Cette forme d'apprentissage a fait des progrès considérables ces dernières années. [11]

Enfin, le dernier type d'apprentissage repose sur un ensemble d'exemples non étiquetés que l'on cherchera à structurer en rassemblant les exemples apparemment les plus proches et en les distinguant de ceux qui en paraissent éloignés. Il s'agit donc, pour la machine, de pouvoir organiser des connaissances et acquérir des notions nouvelles. Ainsi, pour un ensemble d'instruments de musique, on peut chercher à ce que la machine construise automatiquement des catégories en distinguant, par exemple, les instruments à cordes, à vent, à percussion, parmi les cordes, les cordes pincées, frappées, frottées, etc. [11]

L'apprentissage supervisé recourt à des techniques variées fondées sur la logique ou la statistique et s'inspirant de modèles psychologiques, physiologiques ou éthologiques. Parmi celles-ci, des techniques anciennes reposant sur un modèle très approximatif du tissu cérébral.

4.4.2 Les règles d'apprentissage :

- **Règle de Hebb :**

C'est la méthode d'apprentissage la plus ancienne (1949), elle est inspirée de la biologie.

L'algorithme d'apprentissage de Hebb modifie de façon itérative (petit à petit) les poids pour adapter la réponse obtenue à la réponse désirée. Il s'agit en fait de modifier les poids lorsqu'il y a erreur seulement. [2]

1. Initialisation des poids et du seuil s à des valeurs (petites) choisies au hasard.

2. Présentation d'une entrée $E = (e_1, \dots, e_n)$ de la base d'apprentissage.

3. Calcul de la sortie obtenue x pour cette entrée :

$$a = \sum (w_i \cdot e_i) - s$$
 (la valeur de seuil est introduite ici dans le calcul de la somme pondérée).

$$x = \text{signe}(a) \text{ (si } a > 0 \text{ alors } x = +1 \text{ sinon } x = -1 \text{)}.$$

4. Si la sortie x est différente de la sortie désirée d_l pour cet exemple d'entrée E_l alors :

Modification des poids (μ est une constante positive qui spécifie le pas de modification des poids) :

Chapitre II : Les réseaux de neurones artificiels

$w_{ij}(t+1) = w_{ij}(t) + \mu \cdot (x_i \cdot x_j)$ (x_i, x_j : sorties des neurones i et j , μ : facteur d'apprentissage).

5. Tant que tous les exemples de la base d'apprentissage ne sont pas traités correctement : modification des poids et retour à l'étape 2.

- **Rétro-propagation du gradient de l'erreur :**

Cet algorithme est utilisé dans les réseaux de type feedforward, ce sont des réseaux de neurones à couches, ayant une couche d'entrée, une couche de sortie, et au moins une couche cachée. Il n'y a pas de récursivité dans les connexions, et pas de connexions entre neurones de la même couche. Le principe de la rétro propagation consiste à présenter au réseau un vecteur d'entrées, de procéder au calcul de la sortie par propagation à travers les couches, de la couche d'entrées vers la couche de sortie en passant par les couches cachées. Cette sortie obtenue est comparée à la sortie désirée, une erreur est alors obtenue. A partir de cette erreur, est calculé le gradient de l'erreur qui est à son tour propagé de la couche de sortie vers la couche d'entrée, d'où le terme de rétro-propagation. Cela permet la modification des poids du réseau et donc l'apprentissage. [2]

5. 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. [15]

5.1 Architecture :

Un neurone de perceptron réalise un produit scalaire entre son vecteur d'entrées x et un vecteur de paramètres w appelé poids, y ajoute un biais b , et utilise une fonction d'activation f pour déterminer sa sortie. [15]

$$y = f(x \cdot w + b).$$

Les fonctions d'activation doivent être de préférence strictement croissantes et bornées. Les fonctions classiquement utilisées sont la fonction linéaire, la tangente hyperbolique ($f 1$) et la fonction sigmoïde standard ($f 2$) :

$$f_1(x) = \tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$$
$$f_2(x) = \frac{\tanh(x) + 1}{2}$$

La différence entre ces deux dernières fonctions est le domaine des valeurs prises, qui est de]-1 ; 1[pour la tangente hyperbolique et de]0 ; 1[pour la sigmoïde standard.

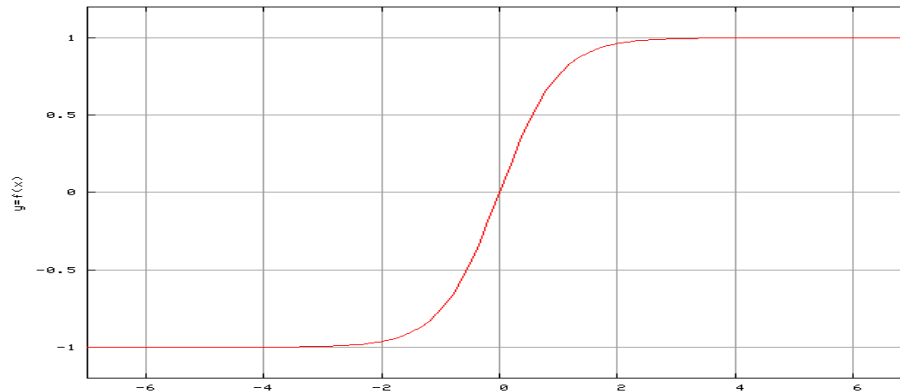


Figure 2.5.1.1 : Tangente hyperbolique.

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. [15]

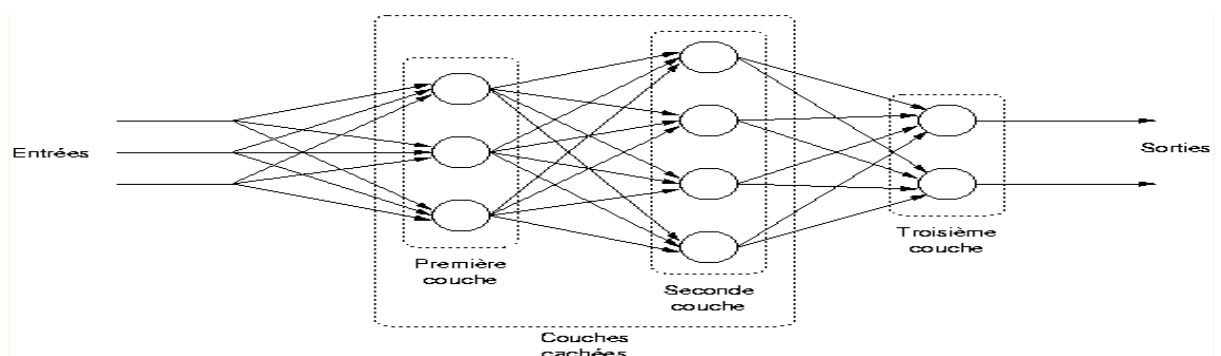


Figure 2.5.1.2 : Schéma d'un PMC.

Nous allons considérer de plus que tous les neurones d'une couche ont la même fonction d'activation, mais qu'elle peut différer d'une couche à l'autre. La fonction d'activation des neurones de la couche est notée f_l et on définit une extension vectorielle de f_l par :

$$\mathbf{f}_1(\mathbf{a}_1) = (f_1(a_{11}), f_1(a_{12}), \dots, f_1(a_{1n_1}))$$

On peut alors écrire la relation suivante pour exprimer la sortie d'une couche en fonction de son entrée :

$$\mathbf{a}_1 = \mathbf{f}_1(\mathbf{a}_{1-1} \cdot \mathbf{W}_1)$$

5.2 Rétro-propagation :

L'apprentissage d'un perceptron se fait avec une descente de gradient, algorithme décrit en annexe, section 1. Dans le cas d'un perceptron à une couche l'expression de l'évolution des poids est assez simple. En effet l'erreur du réseau est de la forme :

$$e_{i,k} = f_1(\mathbf{w}_{1,i} \cdot \mathbf{x}_k) - y_{i,k}$$

Où i est le numéro de la sortie et k celui de l'exemple de la base d'apprentissage. La performance du réseau, une erreur quadratique moyenne, est :

$$\epsilon = \frac{1}{N_a} \sum_{k=1}^{N_a} \sum_{i=1}^{N_s} e_{i,k}^2$$

- Évolution des poids durant une étape de l'apprentissage d'un perceptron à une couche :

$$\delta \mathbf{w}_{1,i} = -2\eta \sum_{k=1}^{N_a} e_{k,i} f_1'(\mathbf{w}_{1,i} \cdot \mathbf{x}_k) \mathbf{x}_k$$

- Performance du perceptron multicouches :

$$\epsilon = \frac{1}{N_a} \sum_{k=1}^{N_a} \|\mathbf{a}_{N_s,k} - \mathbf{y}_k\|^2$$

- Évolution des poids durant une étape de l'apprentissage d'un perceptron multicouches :

$$\delta \mathbf{w}_{1,i} = -2\eta \sum_{k=1}^{N_a} e_{l,k,i} f_l'(\mathbf{w}_{1,i} \cdot \mathbf{a}_{l-1,k}) \mathbf{a}_{l-1,k}$$

- Erreur par couche dans un perceptron multicouche :

$$\begin{cases} e_{N_c, k} = a_{N_c, k} - y_k \\ e_{l, k, i} = \sum_{j=1}^{n_{l+1}} e_{l+1, k, j} w_{l+1, j, i} f'_{l+1}(\mathbf{w}_{l+1, j} \cdot \mathbf{a}_{l, k}) \quad l = 1 \dots N_c - 1, \quad i = 1 \dots n_l \end{cases}$$

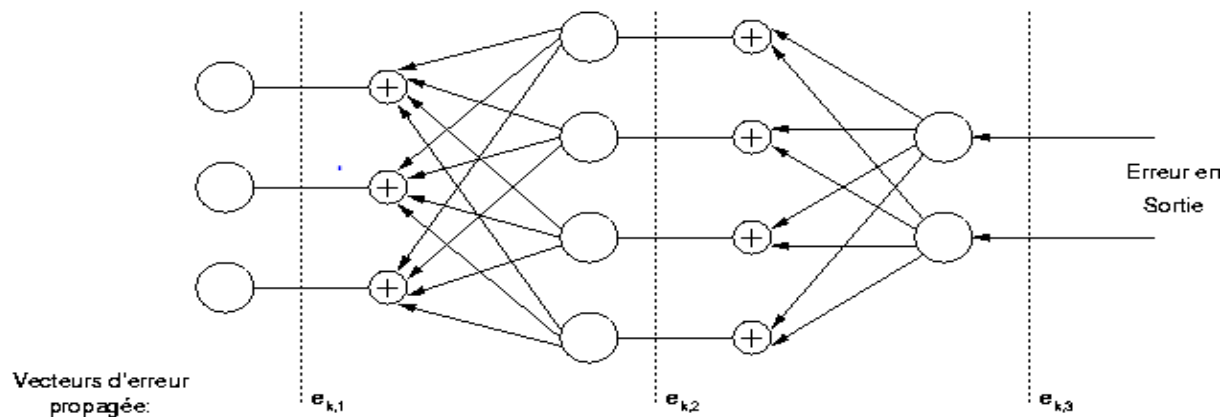


Figure 2.5.2.1 : Principe de la rétro-propagation.

6. Procédure de développement des réseaux de neurones :

Le cycle classique de développement peut être séparé en huit étapes :

1. Collecte des données :

L'objectif de cette étape est de recueillir des données, à la fois pour développer le réseau de neurones et pour le tester. Dans le cas d'applications sur des données réelles, l'objectif est de rassembler un nombre de données suffisant pour constituer une base représentative des données susceptibles d'intervenir en phase d'utilisation du système neuronal.

2. Analyse des données :

Il est souvent préférable d'effectuer une analyse des données de manière à déterminer les caractéristiques discriminantes pour détecter ou différencier ces données. Ces caractéristiques constituent l'entrée du réseau de neurones.

Cette détermination des caractéristiques a des conséquences à la fois sur la taille du réseau (et donc le temps de simulation), sur les performances du système (pouvoir de séparation, taux de détection), et sur le temps de développement (temps d'apprentissage).

3. Séparation ou division de la base de données :

Afin de développer une application à base de réseaux de neurones, il est nécessaire de disposer de deux bases de données : une base pour effectuer l'apprentissage et une autre pour tester le réseau obtenu et déterminer ses performances. Afin de contrôler la phase d'apprentissage, il est souvent préférable de posséder une troisième base de données appelée « base de validation croisée ».

Il n'y a pas de règle pour déterminer ce partage de manière quantitative. Il résulte souvent d'un compromis tenant compte du nombre de données dont on dispose et du temps imparti pour effectuer l'apprentissage.

4. Constriction des réseaux avec des structures adaptées :

Il existe un grand nombre de types de réseaux de neurones, avec pour chacun des avantages et des inconvénients. Le choix des réseaux peut dépendre :

- De la tâche à effectuer (classification, association, prévision...),
- De la nature des données (dans notre cas, des données présentant des variations au cours du temps).

5. Normalisation des 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.

6. Apprentissage des réseaux de neurones :

Tous les modèles de réseaux de neurones requièrent un apprentissage. Plusieurs types d'apprentissages peuvent être adaptés à un même type de réseau de neurones. Les critères de choix sont souvent la rapidité de convergence ou les performances de généralisation.

Le critère d'arrêt de l'apprentissage est souvent calculé à partir d'une fonction de coût, caractérisant l'écart entre les valeurs de sortie obtenues et les valeurs de références.

7. Test et validation :

Une fois le réseau de neurones entraîné (après apprentissage), il est nécessaire de le tester sur une base de données différente de celles utilisées pour l'apprentissage. Ce test permet à la fois d'apprécier les performances du système neuronal et de détecter le type de données qui pose problème. Si les performances ne sont pas satisfaisantes, il faudra soit modifier l'architecture du réseau, soit modifier la base d'apprentissage.

8. Dé-normalisation des données :

C'est la dernière phase dans le cycle de développement des réseaux de neurones artificiels dont laquelle les données prennent ses valeurs réelles.

7. Conclusion :

Dans ce chapitre, nous avons d'abord parlé de l'objectif des réseaux de neurones artificiels pour la prédiction du diabète, puis nous avons d'expliquer les réseaux de neurones artificiels: définition et architecture ainsi que la fonction d'activation et d'apprentissage.

Le type de RNA choisi dans ce chapitre est l'apprentissage profonds supervisée, bien qu'il est le plus approprié pour notre prédiction du diabète.

Chapitre III

Implémentation et résultats

1.Introduction :

Dans le but de développer un système de prédiction du diabète on a utilisé un ensemble de données sur le diabète des aborigènes de l'inde.

Le type de RNA qu'on a utilisé l'apprentissage supervisé bien qu'il est le plus adapté dans ce domaine.

L'application créée dans ce travail est développée en utilisant plusieurs outils.

2. Environnement et outils de travail :

2.1 Langage de programmation :

Nous avons choisi Python version 3.7.4 pour implémenter notre système de prédiction du diabète par les réseaux de neurone artificiels. Est un langage de programmation interprété, multi-paradigme et multiplateformes. Il favorise la programmation impérative structurée, fonctionnelle et orientée objet. Il est doté d'un typage dynamique fort, d'une gestion automatique de la mémoire par ramasse-miettes et d'un système de gestion d'exceptions. [15]

Le langage Python est placé sous une licence libre proche de la licence BSD7 et fonctionne sur la plupart des plates-formes informatiques, des smartphones aux ordinateurs centraux8, de Windows à Unix avec notamment GNU/Linux en passant par macOS, ou encore Android, iOS, et peut aussi être traduit en Java. Il est conçu pour optimiser la productivité des programmeurs en offrant des outils de haut niveau et une syntaxe simple à utiliser. [16]

2.2 Bibliothèques Python :



Figure 3.2.2 : Bibliothèques Python.

Les bibliothèques et package utilisé :

- **Keras** : Keras est une bibliothèque open source écrite en Python (sous licence MIT) basée principalement sur les travaux du développeur de Google François Chollet dans le cadre du projet ONEIROS (Open-ended Neuro-Electronic Intelligent Robot Operating System). Une première version du logiciel multiplateforme a été publiée le 28 mars 2015. Le but de cette bibliothèque est de permettre la constitution rapide de réseaux neuronaux. Dans ce cadre, Keras ne fonctionne pas comme un framework propre mais comme une interface de programmation applicative (API) pour l'accès et la programmation de différents frameworks d'apprentissage automatique. TensorFlow font notamment partie des frameworks pris en charge par Keras. [17]
- **TensorFlow** : est une bibliothèque open source de Machine Learning, créée par Google, permettant de développer et d'exécuter des applications d'apprentissage automatique et d'apprentissage en profondeur. [18]
- **Scikit-learn** : est une bibliothèque libre Python destinée à l'apprentissage automatique. Elle est développée par de nombreux contributeurs² notamment dans le monde académique par des instituts français d'enseignement supérieur et de recherche comme Inria³. Elle propose dans son framework de nombreuses bibliothèques d'algorithmes à implémenter clé en main, à disposition des data scientists. Elle comprend notamment des fonctions pour estimer des forêts aléatoires, des régressions logistiques, des algorithmes de classification, et les machines à vecteurs de support. Elle est conçue pour s'harmoniser avec d'autres bibliothèques libres Python, notamment NumPy et SciPy. [19]
- **NumPy** : est une extension du langage de programmation Python, destinée à manipuler des matrices ou tableaux multidimensionnels ainsi que des fonctions mathématiques opérant sur ces tableaux. Plus précisément, cette bibliothèque logicielle libre et open source fournit de multiples fonctions permettant notamment de créer directement un tableau depuis un fichier ou au contraire de sauvegarder un tableau dans un fichier, et manipuler des vecteurs, matrices et polynômes. [20]
- **Le package h5py** : est une interface Pythonic au format de données binaires HDF5.

Il vous permet de stocker d'énormes quantités de données numériques et de manipuler facilement ces données à partir de NumPy. Par exemple, vous pouvez découper en ensembles de données de plusieurs téraoctets stockés sur disque, comme s'il s'agissait de véritables tableaux NumPy. Des milliers d'ensembles de données peuvent être stockés dans un seul fichier, classés et étiquetés comme vous le souhaitez.

2.3 Editeur de code :

Pour éditer le code de ce système, nous avons utilisé Sublime Text 3.0 qui est un éditeur de texte générique codé en Python et C++, disponible sur Windows, Mac et Linux. Depuis la version 2.0, sortie le 26 juin 2012, l'éditeur prend en charge 44 langages de programmation majeurs, tandis que des plugins sont souvent disponibles pour les langages plus rares. [21]



Figure 3.2.3 : Editeur de code Sublime Text.

3. Analyse exploratoire de données :

3.1. Distribution de données :

Nous avons nommé notre ensemble de données « prima-indians-diabetes ».

L'ensemble de données sur le diabète des aborigènes Pima est fourni avec l'aimable autorisation de l'Institut national du diabète et des maladies digestives et rénales et de Vincent Sigillito du Laboratoire de physique appliquée de l'Université Johns Hopkins qui était le donateur original de l'ensemble de données. Les données réelles elles-mêmes sont obtenues par l'auteur de cette recherche sur le site Web de l'UCI (Université de Californie, Irvine). Ces données ont été utilisées dans le passé par les chercheurs pour enquêter sur d'éventuels signes vitaux pouvant être utilisés pour indiquer la présence de diabète chez les patients selon les normes de l'Organisation mondiale de la santé (OMS). Il y a un total de 768 instances d'entraînement incluses dans cet ensemble de données. [13]

Chapitre III : implémentation et résultats

| Caractéristiques | Type de variable | Intervalle |
|---|------------------|--------------------------------------|
| Grossesses : (nombre de fois enceinte). | Entier | 0–17 |
| Glucose : (Concentration de glucose plasmatique à 2 heures dans un test de tolérance au glucose par voie orale). | Réel | 0–199 |
| Pression artérielle : (Pression sanguine diastolique). | Réel | 0–122 |
| Épaisseur de la peau : (Épaisseur du pli cutané du triceps (mm)). | Réel | 0–99 |
| Insuline : (Insuline sérique de 2 heures (mu U/ml)). | Réel | 0–846 |
| IMC : (Indice de masse corporelle). | Réel | 0–67.1 |
| Fonction Pedigree Diabète. | Réel | 0.078–2.42 |
| Age. | Entier | 21–81 |
| Résultat. | Binaire | Testé positif pour le diabète = 1 |

Tableau 3.3.1: Description des caractéristiques du l'ensemble de données "Pima Indian Diabetes".

Chaque instance d'entraînement a huit caractéristiques et une variable de classe qui fournit l'étiquette de cette instance d'entraînement. La variable de classe prend la valeur binaire 0 ou 1, 0 indiquant une personne en bonne santé et 1 indiquant un patient diabétique.

| | | | | | | | | | |
|----|----|-----|----|----|-----|------|-------|----|---|
| 1 | 6 | 148 | 72 | 35 | 0 | 33.6 | 0.627 | 50 | 1 |
| 2 | 1 | 85 | 66 | 29 | 0 | 26.6 | 0.351 | 31 | 0 |
| 3 | 8 | 183 | 64 | 0 | 0 | 23.3 | 0.672 | 32 | 1 |
| 4 | 1 | 89 | 66 | 23 | 94 | 28.1 | 0.167 | 21 | 0 |
| 5 | 0 | 137 | 40 | 35 | 168 | 43.1 | 2.288 | 33 | 1 |
| 6 | 5 | 116 | 74 | 0 | 0 | 25.6 | 0.201 | 30 | 0 |
| 7 | 3 | 78 | 50 | 32 | 88 | 31.0 | 0.248 | 26 | 1 |
| 8 | 10 | 115 | 0 | 0 | 0 | 35.3 | 0.134 | 29 | 0 |
| 9 | 2 | 197 | 70 | 45 | 543 | 30.5 | 0.158 | 53 | 1 |
| 10 | 8 | 125 | 96 | 0 | 0 | 0.0 | 0.232 | 54 | 1 |
| 11 | 4 | 110 | 92 | 0 | 0 | 37.6 | 0.191 | 30 | 0 |
| 12 | 10 | 168 | 74 | 0 | 0 | 38.0 | 0.537 | 34 | 1 |
| 13 | 10 | 139 | 80 | 0 | 0 | 27.1 | 1.441 | 57 | 0 |

Figure 3.3.1 : l'ensemble de données "Pima Indian Diabetes".

3.2. Séparation ou division de la base de données :

L'ensemble de données est divisé en entrée (X) et sortie (Y). X sont des entrées et Y est la sortie, puis x et y sont divisés en groupes. La partie test représentera 30% de l'ensemble de données total et le réseau neuronal utilisera les 70% restants pour s'entraîner.

3.3 Apprentissage des réseaux de neurones :

Dans notre cas, l'apprentissage est supervisé c'est-à-dire que les réseaux s'adaptent par comparaison entre les résultats qu'ils ont calculés en fonction des entrées fournies, et la réponse attendue en sortie. Ainsi, les réseaux vont subir des modifications jusqu'à ce qu'ils trouvent la bonne sortie, c'est à-dire celle attendue, correspondant à une entrée donnée.

Pour ce type d'apprentissage, nous avons utilisé une fonction d'activation de type sigmoïde pour la couche de sortie et une fonction d'activation de type relu pour la couche d'entrée ainsi que pour la couche cachée.

Le modèle réseau neuronal est créé de manière séquentielle. Des couches denses (entièrement connectées) sont ajoutées une par une en spécifiant la fonction d'activation Sigmoïde.

Sigmoïde est utilisé à la place de relu car relu donne 1 ou 0, et la sortie doit être une probabilité et sigmoïde donnera un flottant entre 0 et 1, ce qui sera la probabilité prédite d'un diagnostic de diabète positif.

Un réseau de neurones qui utilise l'entropie binaire calcule la perte (fonction de perte) et les rapports de gradient d'Adam (optimisés) pour améliorer la précision métrique.

- La fonction d'activation relu a été utilisée pour tous les nœuds à l'exception de ceux de la couche de sortie, qui donne 1 ou 0.
- La couche de sortie utilise l'activation sigmoïde pour donner une valeur comprise entre 0 et 1, ce qui serait la chance prévue de diagnostic.
- **Fonction de perte ou de coût:**

On a utilisé La fonction de perte d'entropie croisée binaire car la sortie contient 1 pour un diagnostic positif ou 0 pour un diagnostic négatif. L'entropie croisée est utilisée pour évaluer la différence entre deux probabilités. Habituellement, la distribution « vraie » (celle que le réseau de neurones essaie de faire correspondre) est exprimée en termes de 1 ou 0 pour vrai ou faux dans les problèmes de classification.

Chapitre III : implémentation et résultats

- **Optimiseur :**

Adam a été utilisé comme optimiseur car c'est un moyen efficace d'appliquer la descente de gradient. La descente de gradient est un algorithme d'optimisation utilisé pour trouver les valeurs des paramètres d'une fonction qui minimise une fonction de perte, dans ce cas l'entropie croisée binaire discutée ci-dessus.

- **Procédure de descente de gradient :**

Gradient Descente commence par les valeurs initiales du paramètre de la fonction. Cela peut être 0 ou une valeur aléatoire (généralement un petit nombre) entre -1 et 1. Utilisons des coefficients comme paramètre pour cet exemple.

Coefficient = 0,0.

- La perte du coefficient est évaluée en le branchant dans la fonction et en calculant la perte. Dans ce cas, nous avons utilisé l'entropie croisée binaire qui calcule la différence entre prédite et réelle. Prédite provient de la couche de sortie des réseaux neuronaux et provient en fait de l'ensemble de données.

Coût = f (coefficient) ou coût = évaluer (f (coefficient))

- Le dérivé de la perte est ensuite calculé. La dérivée fait référence à la pente de la fonction en un point donné. La raison pour laquelle la pente est importante est de trouver la direction (signe +/-) pour déplacer les valeurs de coefficient afin d'obtenir une perte inférieure à l'itération suivante. Les itérations sont également appelées époques.

Delta = dérivé (coût)

- Dans notre cas, la dérivée et le signe sont négatifs, par conséquent les valeurs de coefficient peuvent être mises à jour en conséquence pour augmenter la précision. Un paramètre de taux d'apprentissage (alpha) doit être spécifié pour contrôler le degré de modification des coefficients à chaque itération. Le taux d'apprentissage est important pour que nous puissions déterminer la précision, progressivement à chaque itération sans changer de poids de manière drastique. Pour ce projet, le taux d'apprentissage est de 0,001.

Coefficient = coefficient - (alpha * delta)

Ce processus est répété jusqu'à ce que la perte de coefficients (coût) soit de 0,0 ou suffisamment proche de zéro pour être suffisamment bonne.

3.4 Test & Validation :

Dans cette étape, et comme son nom l'indique, on a utilisé une base de test et de validation indépendamment de la base d'apprentissage (30% de la base initiale) afin de tester objectivement la performance des réseaux choisis. Le vecteur de poids optimal qu'on a obtenu au sein de la phase d'apprentissage sera chargé comme vecteur de poids aux patterns de la base de validation afin de tester la performance de notre système. Les résultats obtenus seront présentés dans ce qui suit.

4. Implémentation et résultats obtenus :

4.1 Description des différentes structures :

Nos réseaux de neurones s'implémentent avec leurs différentes couches qui sont constituées elles-mêmes de plusieurs neurones. On a utilisé une couche d'entrée et trois couches cachées ainsi qu'une couche de sortie.

4.2 Description des différentes Procédures :

L'ensemble des différentes procédures et la description de leurs principales méthodes qu'on a utilisées sont cités dans le tableau 3.4.2 :

| Procédures | Description de ses différentes méthodes |
|-----------------------|--|
| Pré-traitement | Chargement de données, construction des bases d'apprentissage ou de validation. |
| Apprentissage | Initialisation, sauvegarde du vecteur des poids optimaux. |
| Validation | Chargement du vecteur des poids optimal et propagation des données. |
| Affichage | Affichage des résultats d'apprentissage ou de validation sur l'ensemble des bases. |

Tableau 3.4.2 : Principales procédures de notre application.

4.3 Résultats et discussion :

Nous avons appelé la fonction pour adapter le réseau neuronal aux données (entraînement du réseau) sur x_{train} et y_{train} l'entrée et la sortie de la partie d'apprentissage des données. La validation de la précision sera effectuée à l'aide de x_{test} et y_{test} la partie de test, ceci est fait afin que le modèle puisse être validé sur des combinaisons de données qu'il n'a jamais vues auparavant.

Epoch est une itération du processus de formation, chaque époque se compose de deux parties. Le premier est le passage avant, les données de test sont transmises à travers le modèle, puis pour chaque prédiction, une entropie croisée binaire est effectuée pour calculer l'erreur. La deuxième partie consiste à optimiser le modèle à l'aide d'Adam par descente de gradient pour ajuster les poids en fonction des résultats de l'entropie croisée binaire. Enfin, les données de test sont exécutées dans le modèle pour obtenir un score de précision. Il est défini de répéter cela 1000 fois pour affiner les poids du modèle.

Le modèle est ensuite évalué sur toutes les données de test pour vérifier sa précision finale. Nous avons constaté que ce réseau neuronal a une précision de 85%.

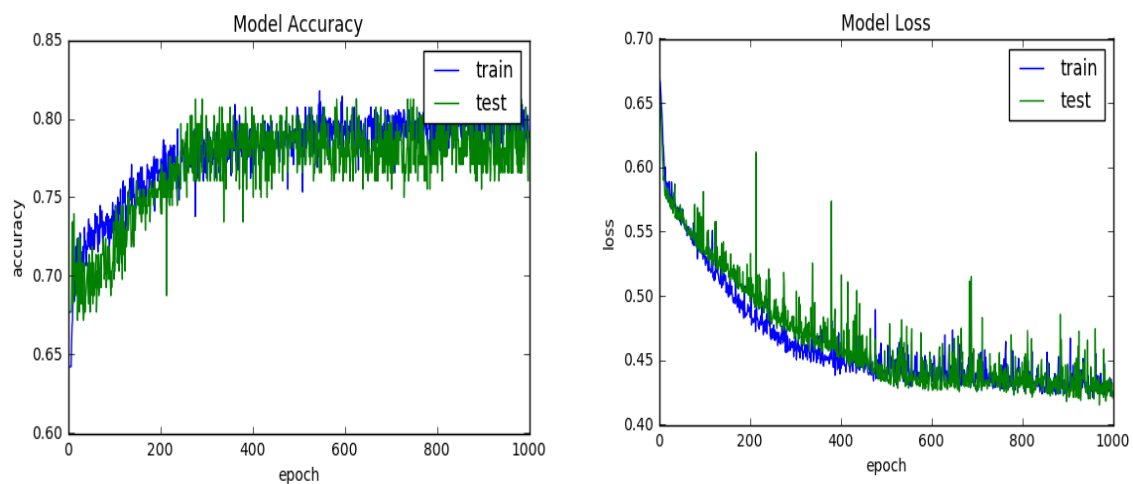


Figure 3.4.3 : Visualisation des performances du modèle entraîné.

Cette figure décrit les résultats du travail expérimental terminé. Quatre tracés sont tracés pour les éléments suivants : la précision d'apprentissage, la perte d'apprentissage, la précision de validation et la perte de validation. Tous décrivent les précisions obtenues lors des expériences. Les principaux objectifs étaient d'obtenir une grande précision et la plus faible

Chapitre III : implémentation et résultats

perte de données. A partir de ces figures, on peut voir que les résultats expérimentaux de ce travail ont atteint un très bon rapport entre la précision et la perte.

4.4 Prédiction du diabète :

La dernière étape dans notre projet est la prédiction du diabète.

Après avoir enregistré le modèle afin qu'il puisse être rechargé plus tard dans un autre fichier de prédiction, nous pouvons maintenant l'utiliser pour prévoir au lieu de l'entraîner avant chaque prédiction.

L'utilisateur doit saisir des données telles que les suivantes : Grossesses, Glucose, Pression artérielle, Épaisseur de la peau, Insuline, IMC, Fonction Pedigree Diabète, Age pour faire la prédiction.

```
Please Enter the Following Metrics one at a time
Enter Metric 1: 1
Enter Metric 2: 2
Enter Metric 3: 3
Enter Metric 4: 4
Enter Metric 5: 5
Enter Metric 6: 6
Enter Metric 7: 7
Enter Metric 8: 8
[[0.99720687]]
Would you like to enter more data:(y/n )n
```

Figure 3.4.4 : Exemple de détection d'une fausse information.

5. conclusion :

Notre système développé en langage de programmation Python version 3.7.4 offre à l'utilisateur le choix pour faire apprendre et tester les structures choisies pour la prédiction du diabète.

Dans ce chapitre, nous avons présenté les différentes étapes que nous avons prises pour parvenir au développement et au bon fonctionnement de notre système de prédiction du diabète.

Conclusion générale

Conclusion générale

La prédiction et les réseaux de neurones artificiels ont été les deux axes principaux sur lesquelles est basée notre recherche.

Le domaine de la médecine a été choisi dans ce projet pour effectuer la prédiction de diabète.

Cette application a été implémentée en Python version 3.7.4.

Dans ce contexte, notre travail a visé plus d'un objectif. En premier lieu, il nous a permis d'explorer le domaine médical et savoir si le patient est diabétique ou non. En outre, nous avons eu l'opportunité de mettre en pratique toutes nos connaissances dans le domaine de réseaux de neurone artificiels et l'apprentissage profonds supervisé sur un thème sanitaire aussi important.

Dans ce contexte, notre travail a visé plus d'un objectif. En premier lieu, il nous a permis d'explorer le domaine médical et Savoir si le patient est diabétique ou non. En outre, nous avons eu l'opportunité de mettre en pratique toutes nos connaissances dans le domaine de réseaux de neurone artificiels et l'apprentissage profonds supervisé sur un thème sanitaire aussi important.

Ce système permet à l'utilisateur de savoir s'il est diabétique ou non en saisissant ses propres données.

Bibliographie

Bibliographie

- [1] <https://www.caledonia.nc/actualite/le-diabete-cette-maladie-silencieuse>.
- [3] www.novonordisk.com.
- [2] Mémoire : un système de prévision la charge électrique.
- [6] <http://mohamed.beida.free.fr/Docs/Prevision.pdf>.
- [7] Heike Labud, nicole Jenni et noémie Marcoz Validé par Dresses Daniela sofrà et Patricia Halfon. Date de la fiche: mai 2015. Date de la prochaine révision: mai 2017.
- [8] <https://sante.journaldesfemmes.fr/fiches-maladies/2499922-diabete-type-1-2-definition-causes-symptomes-traitement-regime-complications/>.
- [9] <https://www.investopedia.com/terms/f/forecasting.asp#:~:text=What%20Is%20Forecasting%3F,an%20upcoming%20period%20of%20time>.
- [10] https://fr.wikipedia.org/wiki/R%C3%A9seau_de_neurones_artificiels#R%C3%A9seaux_%C3%A0_apprentissages_supervis%C3%A9s.
- [11] <https://www.universalis.fr/encyclopedie/apprentissage-profond-deep-learning/1-differents-types-d-apprentissage-machine/#:~:text=On%20distingue%20usuellement%20au%20moins,et%20l'apprentissage%20non%20supervis%C3%A9>.
- [12] <https://www.becoz.org/these/memoirehtml/ch06s04.html>.
- [13] <https://www.kaggle.com/uciml/pima-indians-diabetes-database>.
- [15] [https://fr.wikipedia.org/wiki/Python_\(langage\)](https://fr.wikipedia.org/wiki/Python_(langage)).
- [16] https://m.facebook.com/Nkfondation/photos/a.2307858662815735/2787456238189306/?type=3&refsrc=deprecated&_rdr.
- [17] <https://www.ionos.fr/digitalguide/web-marketing/search-engine-marketing/quest-ce-que-keras/>.
- [18] <https://www.lebigdata.fr/tensorflow-definition-tout-savoir>.
- [19] <https://fr.wikipedia.org/wiki/Scikit-learn>.
- [20] <https://fr.wikipedia.org/wiki/NumPy#:~:text=NumPy%20est%20une%20biblioth%C3%A8que%20pour,math%C3%A9matiques%20op%C3%A9rant%20sur%20ces%20tableaux.&text=NumPy%20est%20la%20base%20de,Python%20autour%20du%20calcul%20scientifique>.
- [21] https://fr.wikipedia.org/wiki/Sublime_Text.