



الجمهورية الجزائرية الديمقراطية الشعبية
République Algérienne Démocratique Et Populaire
وزارة التعليم العالي والبحث العلمي
Ministère De L'enseignement Supérieur Et De La Recherche Scientifique



Université Constantine 1 Frères Mentouri
Faculté des Sciences de la Nature et de la Vie

جامعة قسنطينة 1 الإخوة منتوري
كلية علوم الطبيعة والحياة

Département : Biologie Appliquée

قسم : التطبيقية البيولوجيا

Mémoire présenté en vue de l'obtention du Diplôme de Master

Domaine : Sciences de la Nature et de la Vie

Filière : Biotechnologies

Spécialité : Bio-informatique

N° d'ordre :

N° de série :

Intitulé :

Une plateforme médecin/patient pour la prise en charge intelligente des patients atteints de maladies rares

Présenté par : GASSI Mostefa Habib Errahmane

Le : 10/06/2024

REDJEM Nassim

Jury d'évaluation :

Président : BELLIL Ines (Pr)

Encadrant : BOUCHEHAM Anouar (MCA)

Co-encadrant : CHEHILI Hamza (MCA)

Examineur : DAAS Mohamed Skander (MCA)

Année universitaire
2023 – 2024

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

Remerciement

*Nous remercions **ALLAH** tout-puissant de nous avoir aidés à réaliser ce mémoire et de nous avoir donné la volonté, la force et le courage de terminer ce travail.*

Nous souhaitons exprimer notre profonde gratitude à toutes les personnes qui ont contribué à la réalisation de ce mémoire.

*Tout d'abord, nous tenons à remercier nos directeurs de mémoire, Messieurs **BOUCHEHAM Anouar** et **CHEHILI Hamza**, pour leur encadrement, leurs précieux conseils, et leur disponibilité tout au long de ce projet. Leur patience et leur expertise ont été essentielles à l'avancement de notre travail.*

*Nous remercions également les membres du jury, Pr. **BELLIL Ines** et Dr. **DAAS Mohamed Skander**, pour leur temps, leurs remarques constructives et leurs suggestions pertinentes qui ont permis d'améliorer la qualité de ce mémoire.*

Nous n'oublions pas nos familles, en particulier nos parents, pour leur amour inconditionnel, leur soutien constant et leurs encouragements sans faille. Leur présence a été une source de motivation inestimable.

Enfin, nous tenons à remercier toute l'équipe pédagogique de l'université de Constantine 1, en particulier les professeurs de bio-informatique qui nous ont aidés tout au long de nos études.

Dédicace

Je dédie ce travail à :

À ma mère, pour son amour, ses encouragements et ses sacrifices.

*À mon père **Abderrezak**, pour son soutien, son affection et la confiance qu'il m'a accordé.*

À mes trois chères sœurs

À la mémoire de mon grand-père.

*À mon binôme **Habib**, pour sa patience, de sa compassion et du fait qu'il a toujours été là pour moi tout au long des années que nous avons passées ensemble.*

À ma famille, pour leur compréhension, leur patience et leur présence réconfortante tout au long de ce parcours.

À mes amis, pour leur soutien moral, leur amitié précieuse et les moments de partage qui ont rendu cette aventure plus agréable.

REDJEM Nassim

Dédicace

Je dédie ce mémoire à :

D'abord et avant tout, à mes chers parents. Leur amour, leur éducation, leurs sacrifices et leurs encouragements ont été pour moi une source inestimable d'inspiration et de motivation. Votre soutien indéfectible m'a donné la force d'atteindre ce stade de ma vie et d'être la personne que je suis aujourd'hui.

*Je dédie également ce travail à mes chers frères, **Seif Eddine**, **Abed El Djilil**, **Ayoub**, et à ma chère sœur **Rayene**. Vous m'avez toujours entouré de votre amour et de vos encouragements. Vous êtes la base sur laquelle j'ai construit tous mes succès, et je vous en serai éternellement reconnaissant.*

*À mon binôme **Nassim**, qui est bien plus qu'un partenaire de projet, tu es comme un frère pour moi. Ta présence constante, ta patience et ton amitié au fil des ans ont été inestimables. Ta persévérance a considérablement enrichi notre travail et m'a permis de donner le meilleur de moi-même. Je te remercie pour tout.*

À mes amis, pour leur soutien moral, leur précieuse amitié et les moments de joie partagés. Votre présence a rendu ce voyage inoubliable. Vos encouragements et votre soutien inconditionnel m'ont été d'une grande aide et je vous en suis très reconnaissante.

GASSI Mostefa Habib Errahmane

RÉSUMÉ

Les maladies rares présentent des défis uniques en matière de diagnostic, de traitement et de gestion des patients et sont souvent mal comprises. Ce travail de master présente la conception et la réalisation d'une plateforme innovante destinée au diagnostic des maladies rares en commençant par le diabète néonatal permanent mellitus (PNDM). Cette initiative vise à combler les lacunes existantes dans le diagnostic rapide et précis de ces maladies, également à collecter des informations sur ces maladies pour enrichir les connaissances des médecins et de la communauté, tout en facilitant la communication entre eux. La plateforme se base sur l'utilisation de modèles de l'apprentissage profond, une technique avancée de l'intelligence artificielle permettant d'analyser et d'interpréter des informations médicales complexes. Ces modèles ont été entraînés sur des ensembles de données spécifiques aux maladies rares, avec une première application ciblée sur le PNDM. Le choix du PNDM comme point de départ découle de l'importance de cette maladie et des défis qu'elle pose en termes de diagnostic précoce et de traitement. Le modèle a une performance globale solide avec une valeur élevée d'exactitude de 89%.

Mots clés : Apprentissage Profond, Plateforme intelligente, Diagnostic, Maladies rares, Intelligence Artificiel, PNDM.

ABSTRACT

Rare diseases present unique challenges in diagnosis, treatment and patient management, and are often poorly understood. This master's thesis presents the design and implementation of an innovative platform for the diagnosis of rare diseases, starting with neonatal permanent diabetes mellitus (PNDM). This initiative aims to fill existing gaps in the rapid and accurate diagnosis of these diseases, as well as to collect information on these diseases to enrich the knowledge of doctors and the community, while facilitating communication between them. The platform is based on the use of deep learning models, an advanced artificial intelligence technique for analyzing and interpreting complex medical information. These models have been trained on datasets specific to rare diseases, with an initial application targeting the PNDM. The choice of PNDM as a starting point stems from the importance of this disease and the challenges it poses in terms of early diagnosis and treatment. The model has a strong overall performance, with a high accuracy value of 89%.

Keywords: Deep Learning, Intelligent Platform, Diagnostics, Rare Diseases, Artificial Intelligence, PNDM.

ملخص

تمثل الأمراض النادرة تحديات فريدة من نوعها من حيث التشخيص والعلاج وإدارة المرضى، وغالبًا ما تكون غير مفهومة جيدًا. تقدم رسالة الماجستير هذه تصميم وتنفيذ منصة مبتكرة لتشخيص الأمراض النادرة، بدءاً من داء السكري الدائم لحديثي الولادة (PNDM). والهدف من هذه المبادرة هو سد الثغرات الموجودة في التشخيص السريع والدقيق لهذه الأمراض، وكذلك جمع المعلومات عن هذه الأمراض لإثراء معرفة الأطباء والمجتمع، مع تسهيل التواصل بينهم. تعتمد المنصة على استخدام نماذج التعلم العميق، وهي تقنية ذكاء اصطناعي متقدمة لتحليل وتفسير المعلومات الطبية المعقدة. وقد تم تدريب هذه النماذج على مجموعات بيانات خاصة بالأمراض النادرة، مع تطبيق أولي يستهدف PNDM. ينبع اختيار مرض PNDM كنقطة انطلاق من أهمية هذا المرض والتحديات التي يطرحها من حيث التشخيص والعلاج المبكر. يتمتع النموذج بأداء عام قوي مع دقة عالية تبلغ 89%.

الكلمات المفتاحية: التعلم العميق، منصة ذكية، التشخيص، الأمراض النادرة، الذكاء الاصطناعي، PNDM.

Liste des Figures

Figure 1 : Les maladies rares en chiffres.	6
Figure 2 : Les types des MRs par localisation anatomique	8
Figure 3 : Impact psychologique des maladies rares	14
Figure 4 : Types d'apprentissage automatique.	18
Figure 5 : La relation entre les trois branches l'intelligence artificielle(IA), (ML) et (DL).	21
Figure 6 : Neurone informatique superposé à un schéma de neurone biologique.	22
Figure 7 : Architecture de Réseaux de Neurones Artificiels.	23
Figure 8 : Fonction ReLU	24
Figure 9 : Fonction Sigmoidé	24
Figure 10 : Fonction Tanh	25
Figure 11 : Fonction Softmax	25
Figure 12 : La position de l'anomalie dans le chromosome 6	32
Figure 13 : Mécanisme d'action des sulfamides hypoglycémiant	35
Figure 14 : Code python pour l'importation des bibliothèques nécessaires.	48
Figure 15 : Code python pour Collecte et nettoyage des données.	49
Figure 16 : Code python pour supprimer les lignes en double.	50
Figure 17 : Code python pour Standardisation des variables catégorielles en valeurs numériques.	51
Figure 18 : Code python pour remplacer les valeurs booléennes par des valeurs numériques.	51
Figure 19 : Code python pour supprimée les colonnes (Mutation).	52
Figure 20 : Code python pour la séparation des données.	53
Figure 21 : Les valeurs de la colonne 'PNDM' (l'étiquette).	53
Figure 22 : Visualisons la distribution des classes.	54
Figure 23 : Equilibrer les données.	55
Figure 24 : K-fold validation croisée	56
Figure 25 : Initialisation de K-Fold cross validation et Standardisation des valeurs.	57
Figure 26 : Conception du modèle.	58
Figure 27 : Calcul des métriques.	61
Figure 28 : Les valeurs de chaque métrique pour le pli (fold).	61

Figure 29 : Calcule des moyennes de chaque métrique.....	62
Figure 30 : Tracé la matrice de confusion.	63
Figure 31 : Tracé la Courbe ROC.	63
Figure 32 : Tracé la Courbe de précision-rappel.	64
Figure 33 : Sauvegarde du modèle.	65
Figure 34 : Structure HTML.	66
Figure 35 : Le fichier settings.py	67
Figure 36 : Le fichier urls.py	67
Figure 37 : Le fichier views.py	68
Figure 38 : L'authentification des utilisateurs.	69
Figure 39 : Intégration du modèle de prédiction dans le fichier views.	69
Figure 40 : Intégration du modèle de prédiction dans le fichier views.py.	70
Figure 41 : Intégration du modèle de prédiction dans le fichier views.py.	70
Figure 42 : Le fichier models.py	71
Figure 43 : Relation entre les tables de fichier models.py de la base de données	72
Figure 44 : Relation entre les tables de l'authentification de la base de données	73
Figure 45 : Les valeurs moyennes de chaque métrique.....	75
Figure 46 : La matrice de confusion.....	76
Figure 47 : La courbe ROC.....	77
Figure 48 : La courbe de précision-rappel.....	78
Figure 49 : Page de création de compte.....	81
Figure 50 : Page de connexion à des comptes déjà existants.....	81
Figure 51 : Page d'accueil de la plateforme RareHealth	82
Figure 52 : Formulaire de prédiction.....	83
Figure 53 : Page de résultat de prédiction.....	83

Liste des Tableaux

Tableau 1 : La différence entre (ML) et (DL)	21
Tableau 2 : Statistiques sur les données utilisées.....	39
Tableau 3 : Les caractéristiques de l'ordinateur utilisé	42
Tableau 4 : Les bibliothèques python utilisées.	46
Tableau 5 : Les valeurs de chaque métrique pour les plis.	74
Tableau 6 : Comparaison du notre travail avec un autre.	79

Liste des Abréviation

MRs : Maladies Rares

ADN : Acide Désoxyribonucléique

DL : Deep Learning

ML : Machine Learning

ANN : Artificial Neural Network

MLP : Multi-Layer Perceptron

RNN : Recurrent Neural Network

CNN : Convolutional Neural Network

GAN : Generative Adversarial Network

PNDM : Le Diabète Néonatal Mellitus Permanent

NGS : Next-generation sequencing

ATP : Adenosine triphosphate

CSV : comma-separated values

SGM : Standard Generalized Markup Language

HTML: Hypertext Mark-up Language

CSS: Cascading Style Sheets

JS: JavaScript

CPU : Central Processing Unit

GPU : Graphics Processing Unit

BSD : Berkeley Software Distribution

SMOTE : Synthetic Minority Over-Sampling Technique

ROC : Receiver Operating Characteristic

TPR : True Positive Rate

FPR : False Positive Rate

URL : Uniform Resource Locator

VN : Vrais négatifs / VP : Vrais positifs

FP : Faux positifs / FN : Faux négatifs

AUC : Area Under the Curve

Table des matières

Introduction Générale	1
Chapitre I : Les Maladies Rares	4
1- Introduction.....	5
2- Classification des maladies rares	6
2.1 Classification par Origine	6
2.1.1 Maladies Génétiques	6
2.1.2 Maladies Infectieuses.....	7
2.1.3 Maladies Métaboliques.....	7
2.1.4 Maladies Auto-immunes	7
2.1.5 Maladies Neoplasiques (Cancers Rares)	7
2.2 Classification par Système d'Organes Affecté.....	8
2.3 Classification par Impact	9
2.3.1 Maladies Bénignes	9
2.3.2 Maladies Chroniques.....	9
2.3.3 Maladies Invalidantes.....	9
2.3.4 Maladies Mortelles.....	9
3- Caractéristiques des maladies rares.....	10
3.1 La diversité	10
3.2 La rareté.....	10
3.3 La gravité.....	10
3.4 La complexité.....	11
4- Défis Diagnostiques et Cliniques.....	11
4.1 Défis Diagnostiques	11
4.1.1 Retard de Diagnostic	11
4.1.2 Manque de Connaissances.....	11
4.1.3 Tests Diagnostiques Limités	11
4.2 Défis Cliniques.....	12
4.2.1 Manque de Traitements Spécifiques	12
4.2.2 Accès aux Soins Spécialisés	12
4.2.3 Coût des Traitements.....	13
4.2.4 Recherche Insuffisante	13
5- Impact des maladies rares.....	13
5.1 La vie des patients	13
5.2 La famille des patients.....	15

5.3	La société.....	15
Chapitre II : Apprentissage automatique.....		16
1-	Introduction.....	17
2-	Importance de l'apprentissage automatique.....	17
3-	Types d'apprentissage automatique.....	18
3.1	Apprentissage supervisé.....	18
3.1.1	Régression Linéaire.....	19
3.1.2	K-nearest neighbors (KNN).....	19
3.1.3	Arbres de Décision.....	19
3.1.4	Support Vector Machines (SVM).....	20
3.2	Apprentissage non supervisé.....	20
3.3	Apprentissage par renforcement.....	20
4-	Apprentissage Profond (deep learning).....	21
4.1	Réseaux de Neurones Artificiels (ANN).....	22
4.2	Architecture de Réseaux de Neurones Artificiels.....	22
4.3	Les différentes fonctions d'activation.....	23
	Les différents types de fonctions d'activation utilisées dans les réseaux de neurones artificiels sont variés et chacun présente des caractéristiques spécifiques. Voici quelques types courants de fonctions d'activation :.....	24
4.4	Types de réseaux de neurones artificiels.....	26
4.4.1	Réseaux de neurones multicouches perceptrons (MLP).....	26
4.4.2	Réseaux de neurones récurrents (RNN).....	26
4.4.3	Réseaux de neurones à convolution (CNN).....	26
4.4.4	Réseaux antagonistes (GAN).....	27
Chapitre III : Le Diabète Néonatal Mellitus Permanent (PNDM).....		29
1-	Introduction.....	29
2-	Etudes Expérimentales.....	29
2.1	Étude sur les Mutations Génétiques du Diabète Néonatal Permanent.....	29
2.2	Diagnostic du Diabète.....	30
3-	Étiologie et Génétique du PNDM.....	31
3.1	Mutations dans les gènes codant le canal potassique dépendants de l'ATP.....	31
3.2	Mutations du gène de l'insuline (INS).....	31
3.3	Mutations dans les gènes codant la glucokinase.....	32
3.4	Anomalies d'une région du chromosome 6.....	32

4-	Diagnostic du Diabète Néonatal Permanent	33
4.1	Observation des symptômes cliniques	33
4.2	Tests génétiques	33
4.3	Antécédents familiaux	34
5-	Approches Thérapeutiques du PNDM	34
5.1	Traitement par sulfamide hypoglycémiant	34
5.2	Traitement par l'insuline	35
6-	Stratégies pour améliorer les Soins et le traitement	35
6.1	Diagnostic Précoce et Précis.....	35
6.2	Prise en charge médicale	36
6.3	Recherche et développement	36
6.4	Soutien et éducation	36
	Chapitre IV : Étude Computationnelle	37
1-	Introduction.....	38
2-	Matériel.....	38
2.1	Données cliniques	38
2.2	Configuration de la machine.....	42
2.3	Technologies utilisées pour la création du modèle et le développement de la plateforme.....	42
2.3.1	Environnements de travail.....	42
2.3.2	Langages de programmation.....	43
2.3.3	Base de données	44
2.3.4	Frameworks	44
2.3.5	Bibliothèques	45
3-	Modèle d'apprentissage profond pour la prédiction de PNDM	48
3.1	L'importation des bibliothèques Python nécessaire.....	48
3.2	Prétraitement des données	48
3.3	Construction du modèle d'apprentissage profond	55
3.4	Les mesures d'évaluation du modèle	58
3.5	Sauvegarde du modèle.....	64
4-	Développement de la Plateforme	65
4.1	Création de l'environnement virtuel et installation de Django	65
4.2	Création du projet et des applications	65

4.3	Intégration de HTML, CSS et JavaScript.....	66
4.4	Configuration et routage.....	66
4.5	Gestion des utilisateurs et authentification.....	68
4.6	L'intégration du modèle intelligent dans la plateforme	69
4.7	Conception et gestion de la base de données.....	71
4.8	Hébergement et déploiement	73
5-	Résultats et discussion.....	73
5.1	Résultats d'apprentissage	74
5.2	Plateforme.....	80
5.2.1	Création de compte	80
5.2.2	Connexion.....	81
5.2.3	Page d'accueil.....	82
5.2.4	Interface de diagnostic.....	82
	Conclusion et Perspectives	85
	Plan Financier	87
1-	Les couts et charges.....	87
1.1	Matériels requis.....	87
1.1.1	Achat du mobilier.....	87
1.1.2	Financement pour lancer notre projet (fond de roulement).....	87
1.1.3	Cout globale du projet	88
2-	Calcul prévisionnel du coût de revient du produit	88
2.1	Tarifs des ventes.....	89
2.2	Calcule prévisionnel du chiffre d'affaires de la 1' ère année	89
	BMC.....	87
	Références.....	95

Introduction
Générale

Les maladies rares touchant un faible pourcentage de la population sont souvent graves, chroniques et invalidantes. Les personnes souffrant d'une maladie rare sont également plus vulnérables d'un point de vue psychologique, social, économique et culturel. Le diagnostic des patients atteints de maladies rares revêt une importance fondamentale au sein de la communauté scientifique et médicale, car elle permet d'améliorer la compréhension des maladies et de leurs mécanismes sous-jacents, de développer de nouveaux traitements et thérapies et d'améliorer la prise en charge des patients et leur qualité de vie. Au fil du temps, diverses méthodes ont été utilisées, allant des techniques biochimiques et moléculaires aux approches d'imagerie médicale et de génétique.

Cependant, en raison de leur rareté, ces maladies présentent des défis uniques en matière de diagnostic, de traitement et de gestion des patients. Elles sont souvent mal comprises, ce qui peut entraîner des délais de diagnostic prolongés. L'accès aux médecins spécialistes et aux traitements spécialisés peut être limité. Les patients atteints de maladies rares et leurs médecins se trouvent souvent confrontés à un manque de ressources et de soutien adéquats. C'est pour ça qu'il est essentiel d'explorer de nouvelles approches basées sur l'intelligence artificielle pour améliorer la prise en charge des maladies rares et d'obtenir des résultats plus objectifs et fiables.

Notre solution novatrice consiste en une plateforme médecin/patient pour la prise en charge intelligente des patients atteints de maladies rares. En combinant l'intelligence artificielle et l'analyse de données avec l'expertise médicale, Notre plateforme vise à relever les défis actuels de la santé. Notre objectif est de faciliter le diagnostic des patients souffrant de maladies rares par des médecins spécialisés, les non-spécialistes et les structures de santé publique/privée. Nous visons également à identifier et à collecter des informations sur ces maladies pour enrichir les connaissances des médecins, des patients et de la communauté, tout en facilitant la communication entre eux.

En utilisant l'apprentissage profond, notre plateforme offre un diagnostic et une analyse approfondie des données de santé des patients en temps réel d'une manière efficace, facile et rapide.

En favorisant une communication en temps réel et les collaborations entre les professionnels de la santé pour le partage des connaissances et les échanges d'informations sécurisés, notre plateforme facilite et renforce les liens entre les médecins spécialisés, les non-spécialistes et les patients, en créant un réseau collaboratif dédié à l'amélioration du diagnostic, du traitement et de la gestion des patients atteints de maladies rares.

Ce mémoire de master fournit une présentation détaillée de la réalisation de notre initiative, en mettant des généralités sur les maladies rares, l'apprentissage profond et la méthode utilisée, une étude approfondie sur le diabète néonatal permanent, l'étude expérimentale et le modèle économique.

Chapitre I : Les Maladies Rares

1- Introduction

Les maladies rares sont des maladies qui affectent une petite proportion de la population. Selon l'Union européenne, une maladie est considérée comme rare quand elle touche moins d'une personne sur 2,000. Un grand nombre d'entre elles sont également des maladies dites « orphelines », c'est-à-dire qu'elles sont des conditions rares pour lesquelles les options de traitement sont limitées ou inexistantes [1].

Il existe 7000 à 8000 maladies rares identifiées. On estime qu'entre 3,5 et 5,9 % de la population mondiale vit au quotidien avec une maladie rare, soit entre 263 et 446 millions de personnes. 80% des maladies sont génétiques et 75% affectent les enfants, la figure 1 donne des chiffres sur les maladies rares. Ces maladies sont souvent graves, chroniques et invalidantes et seraient responsables de plus de 30% de mortalité infantile [2].

Les personnes souffrant d'une maladie rare sont également plus vulnérables d'un point de vue psychologique, social, économique et culturel. En outre, le manque de connaissances scientifiques et médicales signifie que de nombreux patients ne reçoivent pas de diagnostic : leur maladie reste inconnue. Il s'agit de personnes qui éprouvent le plus de difficultés à bénéficier d'un soutien approprié.

Les médicaments orphelins sont des produits médicamenteux destinés au diagnostic, à la prévention ou au traitement des maladies rares. Ces médicaments sont qualifiés "d'orphelins" car, dans des conditions normales de marché, il n'est pas rentable pour l'industrie pharmaceutique de développer et de commercialiser des produits destinés à un nombre restreint de patients souffrant de pathologies rares. Les médicaments développés pour ce marché non rentable ne seraient pas viables financièrement pour le fabricant détenteur du brevet. Pour les sociétés pharmaceutiques, le coût de mise sur le marché d'un médicament orphelin ne serait pas couvert par les ventes attendues du produit. C'est pourquoi les gouvernements et les organisations de patients atteints de maladies rares ont souligné la nécessité d'incitations économiques pour encourager les sociétés pharmaceutiques à développer et commercialiser des médicaments destinés aux patients atteints de maladies rares, "orphelins" de traitements [3].

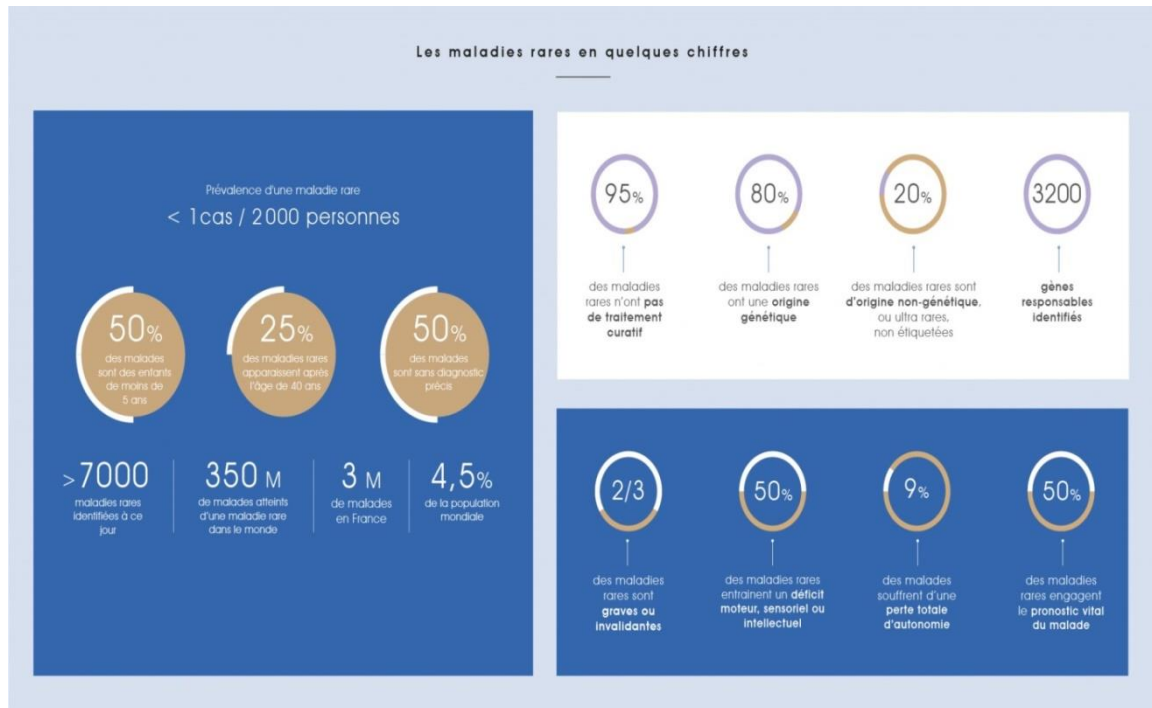


Figure 1 : Les maladies rares en chiffres. [4]

2- Classification des maladies rares

2.1 Classification par Origine

2.1.1 Maladies Génétiques

Les maladies génétiques sont causées par des anomalies de l'ADN. Ces anomalies peuvent prendre de nombreuses formes, comme des mutations dans un seul gène, des anomalies dans plusieurs gènes, des modifications de la structure chromosomique ou encore des mutations qui affectent la régulation d'un gène mais n'altèrent pas sa séquence. Ces maladies peuvent être héréditaires, transmises d'un parent à l'enfant, ou être nouvelles sans antécédents familiaux [5].

Des exemples de troubles génétiques comprennent la mucoviscidose, la dystrophie musculaire de Duchenne et la trisomie 21 (syndrome de Down). Ces conditions peuvent affecter n'importe quel système du corps et sont responsables d'environ 80 % des maladies rares découvertes.

2.1.2 Maladies Infectieuses

Les maladies infectieuses rares sont déclenchées par divers agents pathogènes tels que les virus, les bactéries, les champignons ou les parasites. Ces affections peuvent être peu communes en raison de leur répartition géographique restreinte ou de la spécificité de leur vecteur de transmission [6].

Par exemple, la fièvre de Lassa est principalement présente en Afrique de l'Ouest et est causée par un virus transmis par le rongeur *Mastomys natalensis* [7].

2.1.3 Maladies Métaboliques

Les maladies métaboliques, également appelées erreurs innées du métabolisme, surviennent lorsque le corps rencontre des perturbations dans ses processus biochimiques essentiels pour convertir les nutriments en énergie et en matériaux cellulaires nécessaires. Ces troubles peuvent être causés par des défauts au niveau des enzymes, qui sont des protéines essentielles pour catalyser les réactions chimiques dans le corps. Lorsque ces enzymes sont déficientes, le corps peut avoir des difficultés à décomposer correctement les lipides, les glucides ou les protéines [8].

2.1.4 Maladies Auto-immunes

Les maladies auto-immunes surviennent lorsque le système immunitaire, qui normalement protège l'organisme contre les agents pathogènes, commence à attaquer par erreur les tissus sains du corps. Ce dysfonctionnement peut se manifester par la production d'anticorps ou de cellules immunitaires qui ciblent et endommagent les cellules, les tissus ou les organes du corps. Cette réaction auto-immune peut entraîner une inflammation, des lésions et un dysfonctionnement des organes touchés, provoquant ainsi les symptômes caractéristiques des maladies auto-immunes. Ces affections peuvent affecter divers organes et systèmes du corps, et leur gravité varie en fonction de la maladie spécifique et de la manière dont le système immunitaire réagit [9].

2.1.5 Maladies Néoplasiques (Cancers Rares)

Les maladies néoplasiques, plus couramment connues sous le nom de cancers, représentent un spectre étendu de conditions médicales caractérisées par une croissance cellulaire anormale et incontrôlée. Lorsque ces cancers sont moins

fréquents, ils sont regroupés sous la catégorie des "cancers rares". Ces types de cancers se distinguent souvent des formes plus courantes par leur localisation spécifique, le type de cellules impliquées, ou leur comportement biologique unique [10].

2.2 Classification par Système d'Organes Affecté

Le classement systématique des maladies rares peut être effectué en fonction des systèmes organiques qu'elles touchent.

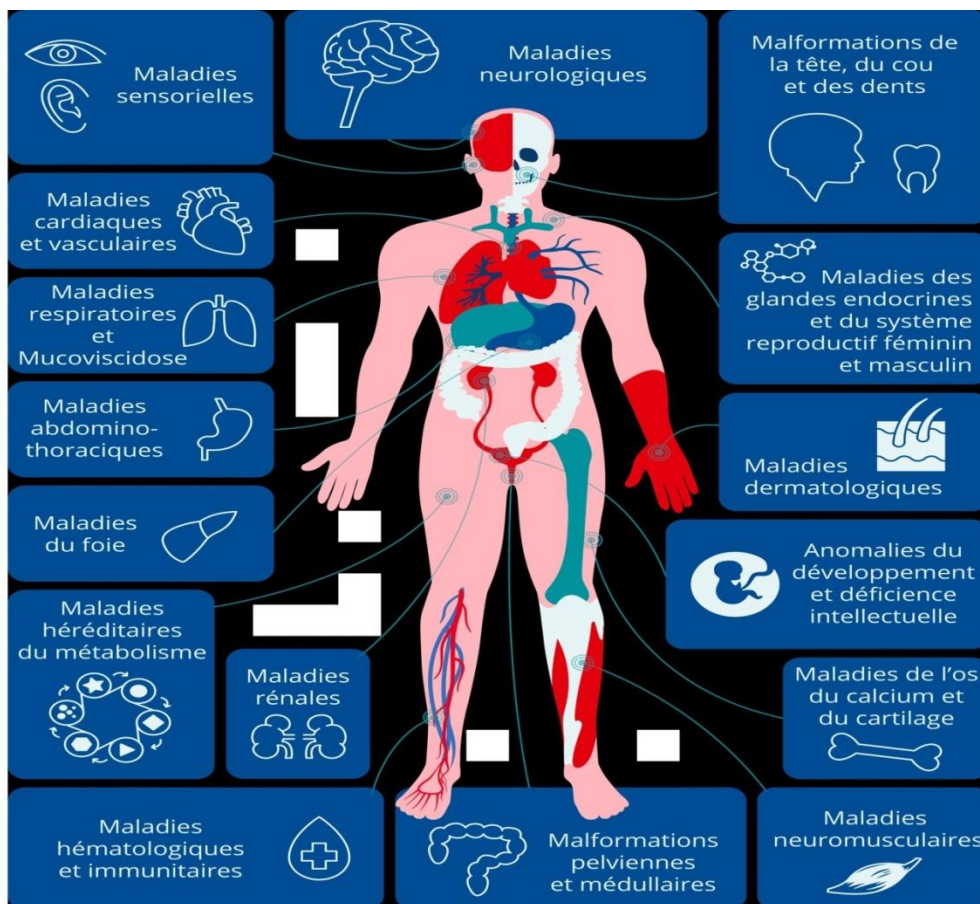


Figure 2 : Les types des MRs par localisation anatomique [11].

- Maladies Neurologiques :

Maladies affectant le système nerveux central et périphérique.

- Maladies Cardiovasculaires :

Maladies affectant le cœur et les vaisseaux sanguins.

- Maladies des Organes Sensoriels :

Maladies affectant les yeux, les oreilles, le nez et la peau.

- Maladies Gastro-intestinales :

Maladies affectant l'estomac, l'intestin, ou d'autres organes digestifs.

- Maladies Hématologiques :

Maladies du sang et des organes producteurs de sang.

- Maladies Endocriniennes :

Maladies affectant les glandes endocrines, qui produisent des hormones.

- Maladies Musculosquelettiques :

Maladies affectant les muscles, les os, ou les articulations.

2.3 Classification par Impact

2.3.1 Maladies Bénignes

Sont généralement caractérisées par le fait qu'elles n'ont pas de conséquences graves et ne mettent pas la vie en danger.

2.3.2 Maladies Chroniques

Les maladies chroniques sont des affections qui persistent tout au long de la vie d'une personne et qui nécessitent généralement des soins médicaux continus.

2.3.3 Maladies Invalidantes

Les maladies invalidantes sont des affections médicales qui limitent de manière significative l'activité et l'autonomie des personnes qui en sont atteintes. Contrairement à une simple déficience, ces maladies entravent grandement la capacité des individus à fonctionner de manière indépendante dans leur vie quotidienne

2.3.4 Maladies Mortelles

Les maladies mortelles sont des affections médicales qui peuvent entraîner la mort si elles ne sont pas traitées efficacement.

Il est important de noter qu'une maladie rare peut appartenir à plusieurs catégories à la fois.

3- Caractéristiques des maladies rares

3.1 La diversité

Il existe une grande diversité de maladies rares, qui présentent un large éventail de symptômes et de troubles, présentant des variations non seulement d'une maladie à l'autre, mais également au sein d'une même maladie. La nature complexe et diversifiée des maladies rares présente des obstacles en termes de diagnostic, de traitement et de contrôle, ce qui souligne la nécessité d'une sensibilisation accrue et d'efforts de recherche visant à traiter ces affections multidimensionnelles et souvent invalidantes [12].

Les maladies rares peuvent présenter une variabilité selon les régions géographiques en raison de l'hétérogénéité génétique au sein des populations ou en raison de facteurs environnementaux ou socioculturels [13].

3.2 La rareté

La population limitée de patients atteints de maladies rares peu communes pose des défis en matière de collecte de données et d'exécution de la recherche clinique, ce qui entraîne une augmentation des coûts. Ces maladies présentent souvent des obstacles particuliers en raison de leur rareté, ce qui entraîne des périodes prolongées d'erreur de diagnostic ou d'absence de diagnostic.

Les connaissances concernant la prévalence et les données épidémiologiques de la plupart des maladies sont insuffisantes [13].

3.3 La gravité

Les maladies rares se caractérisent souvent par leur gravité, leur caractère chronique et leur évolution progressive, affectant généralement de multiples systèmes et organes, exerçant ainsi une influence considérable sur la qualité de vie des personnes atteintes de ces maladies [13].

3.4 La complexité

Les maladies rares se caractérisent par leur complexité, qui englobe de faibles taux d'incidence, des antécédents génétiques, des affections graves et une progression rapide.

Les symptômes de nombreuses maladies peuvent se manifester à la naissance ou apparaître pendant l'enfance, alors que d'autres ne se manifestent qu'à l'âge adulte [13].

4- Défis Diagnostiques et Cliniques

4.1 Défis Diagnostiques

4.1.1 Retard de Diagnostic

Le diagnostic tardif est un problème important dans les maladies rares, des études montrant des délais prolongés pour obtenir un diagnostic correct. Les recherches indiquent que plus de la moitié des personnes atteintes de maladies rares connaissent des retards de diagnostic, le délai moyen d'obtention d'un diagnostic allant de 2 à 6,18 ans. Les facteurs contribuant aux retards comprennent le fait d'être une femme, l'apparition des symptômes entre 30 et 44 ans et des types spécifiques des MRs comme les troubles mentaux et comportementaux ou les maladies du système nerveux [14].

4.1.2 Manque de Connaissances

Le manque de connaissances et de sensibilisation aux MRs parmi les professionnels de santé et les étudiants en médecine est un problème important mis en évidence dans les contextes fournis. Des études menées dans divers pays, révèlent une compréhension insuffisante des MRs, entraînant un diagnostic retardé et des soins inadéquats. Ce manque de sensibilisation retarde la prise en charge opportune des MRs, ce qui a un impact négatif sur la qualité de vie des personnes touchées et de leurs familles [15].

4.1.3 Tests Diagnostiques Limités

Le manque de tests diagnostiques pour les maladies rares pose un défi important pour un diagnostic rapide et précis. L'établissement de critères

diagnostiques précis basés sur des données étendues et la création de bases de données publiques accessibles pour les maladies rares peuvent améliorer la précision du diagnostic.

Les variants pathogènes dans les gènes ne sont identifiés que dans 20 % à 40 % des MRs, principalement dans les régions codantes des protéines, soulignant le besoin critique d'amélioration des outils diagnostiques pour les maladies rares.

Un meilleur diagnostic et des thérapies ciblées sont essentiels pour prendre en charge efficacement les MRs, étant donné leur nature chronique et invalidante, leurs coûts de santé élevés et le nombre limité d'options de traitement [16].

4.2 Défis Cliniques

4.2.1 Manque de Traitements Spécifiques

Malgré les avancées majeures dans la recherche sur les maladies rares et les incitations réglementaires, la plupart des maladies rares manquent encore de traitements spécifiques. Seulement environ 4 % des maladies rares disposent de thérapies approuvées, 95 % d'entre elles ne bénéficiant pas de thérapies approuvées en raison de petites populations de patients, laissant de nombreux patients sans options de traitement adéquates. Des défis tels que la faible prévalence des maladies, la connaissance limitée de la physiopathologie des maladies et les coûts élevés de développement entravent la création de traitements pour les maladies rares [17].

4.2.2 Accès aux Soins Spécialisés

Trouver des spécialistes qui connaissent et comprennent une maladie rare particulière peut être difficile, surtout dans les régions éloignées ou moins développées. L'accès à des soins spécialisés pour les maladies rares est crucial pour les patients et leurs familles, comme le soulignent diverses études. Le Plan d'action pour les maladies rares en Angleterre vise à améliorer l'accès aux soins spécialisés et aux traitements pour les personnes atteintes de maladies rares. Les études insistent sur l'importance de la collaboration intersectorielle et des interfaces fluides entre les médecins généralistes et les centres d'expertise afin de fournir des soins appropriés aux patients atteints de maladies rares [18].

4.2.3 Coût des Traitements

Les coûts de traitement de maladies rares telles que la drépanocytose et l'hémoglobinurie paroxystique nocturne peuvent être élevés. Certains médicaments coûtant jusqu'à 400 000 dollars par an et par patient dans les pays à revenu élevé [19].

4.2.4 Recherche Insuffisante

La recherche sur les maladies rares est souvent restreinte en raison de la rareté individuelle de chaque affection. Ce qui explique en partie la connaissance limitée qui persiste souvent. Cela se reflète dans la description précise de leurs symptômes, de leurs marqueurs diagnostiques, ainsi que dans la compréhension des mécanismes sous-jacents de ces maladies [20].

5- Impact des maladies rares

5.1 La vie des patients

Les maladies rares ont un impact significatif sur les patients, entraînant des odyssées diagnostiques, des charges financières et des défis psychologiques, comptant largement sur les soignants pour la gestion des soins physiques, le soutien émotionnel et la coordination des services. Les patients atteints de maladies rares sont souvent confrontés à une errance diagnostique prolongée, ce qui retarde la mise en place de traitements adaptés et peut entraîner des complications supplémentaires. En plus des défis médicaux, les patients font face à des charges financières importantes, notamment en raison des coûts élevés des traitements spécifiques et des soins nécessaires [21].

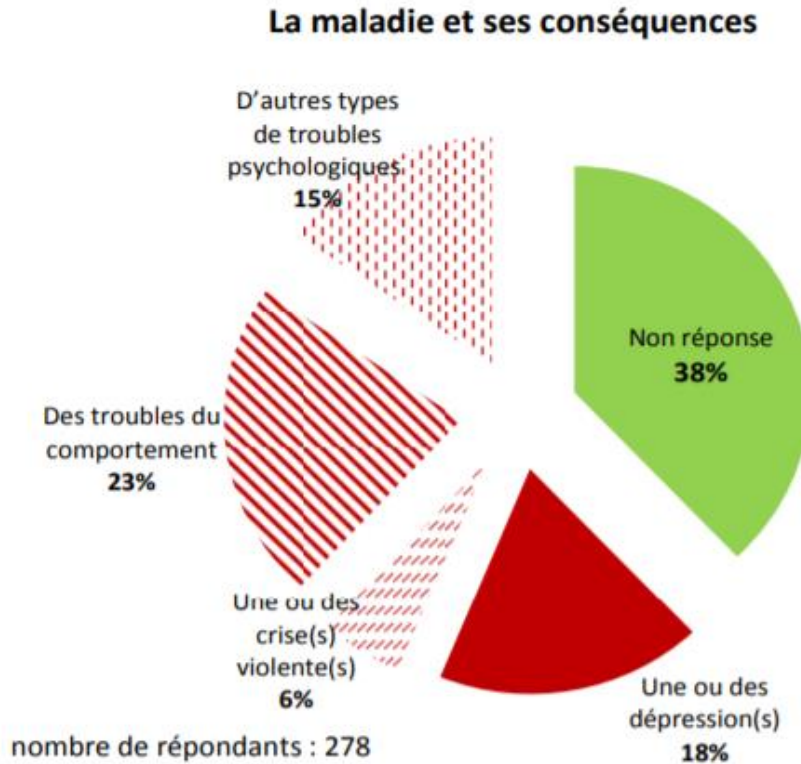


Figure 3 : Impact psychologique des maladies rares [22]

L'observatoire des maladies rares, une étude permettant d'évaluer l'impact de ces pathologies sur la vie des patients, a mis en évidence certains phénomènes concernant leur poids psychologique :

- Plus d'un tiers des patients se voient proposer un suivi psychologique.
- Les patients jeunes, âgés de 2 à 17 ans, se voient plus fréquemment proposer un suivi psychologique que les patients adultes.

Ces troubles psychologiques peuvent se manifester de diverses façons, notamment par de la dépression, des accès de violence ou des troubles du comportement.

Selon une étude du Rare Barometer, **37 %** des répondants déclarent être éprouvent fréquemment des sentiments de tristesse et de désespoir, et peuvent se retrouver isolés socialement. Le manque de compréhension de leur maladie, le soutien insuffisant, le stress lié à l'attente d'un diagnostic et la gravité de leur pathologie contribuent à l'apparition de troubles psychologiques. De plus, les altérations

physiques causées par la maladie peuvent également impacter la santé mentale des patients [22].

5.2 La famille des patients

Les familles des patients atteints de maladies rares font face à des défis importants, notamment en termes de besoins non comblés, de délais prolongés pour obtenir un diagnostic, de contraintes financières et d'impacts émotionnels significatifs. Cette pression peut entraîner des tensions au sein de la famille, comme en témoignent les résultats de l'étude menée par l'observatoire des malades, où une grande majorité des participants ont signalé des tensions dans leur relation de couple, allant parfois jusqu'à des conflits graves pouvant mener à une séparation ou un divorce [22].

5.3 La société

Les maladies rares ont un impact significatif sur la société, et diverses études mettent en évidence leurs implications économiques, sociales et sur la qualité de vie. Le fardeau économique des maladies rares est considérable, avec les coûts associés aux soins médicaux, aux pertes de productivité et aux soins informels, mettant à rude épreuve les systèmes de santé et les familles.

L'étude européenne du Rare Barometer révèle que 21% des 1095 participants ont été absents du travail pendant 90 jours ou plus en raison de rendez-vous médicaux liés à une maladie rare. Cela représente un trimestre d'absence professionnelle, soit un impact considérable sur la vie professionnelle des personnes atteintes et la société.

L'absence de traitements efficaces pour de nombreuses maladies rares exacerbe les défis auxquels sont confrontés les patients, soulignant la nécessité d'améliorer la recherche, les politiques et les systèmes de soutien pour résoudre les problèmes complexes liés aux maladies rares dans la société [22].

Chapitre II :
Apprentissage
automatique

1- Introduction

L'apprentissage automatique, un sous-ensemble de l'intelligence artificielle, permet aux machines d'apprendre sans programmation explicite en améliorant les performances grâce à des expériences passées ou à des exemples de formation [23].

Il vise à programmer des machines pour qu'ils apprennent et résolvent automatiquement des problèmes complexes dépassant les capacités humaines, tels que la conduite, la reconnaissance vocale et la compréhension des images [24].

Les algorithmes d'apprentissage automatique s'adaptent et s'améliorent au fur et à mesure qu'ils apprennent à partir des données, en faisant des prédictions et en prenant des décisions sur la base des données dans diverses applications, du filtrage du courrier indésirable à la traduction linguistique [25].

En utilisant des techniques informatiques pour apprendre directement à partir des données, l'apprentissage automatique englobe des approches d'apprentissage supervisées, non supervisées et par renforcement, améliorant ainsi son adaptabilité à l'évolution des environnements et des tâches. Cet aperçu met en lumière l'importance de l'apprentissage automatique dans la technologie moderne et ses nombreuses applications dans notre vie quotidienne.

2- Importance de l'apprentissage automatique

L'apprentissage automatique joue un rôle crucial dans divers secteurs, offrant des avantages tels qu'une meilleure prise de décision, une meilleure expérience client et des avantages concurrentiels [26].

Il permet aux organisations d'extraire des informations précieuses à partir de grandes quantités de données, ce qui permet de réaliser des prévisions plus précises et d'accélérer le traitement des informations [26].

Les algorithmes d'apprentissage automatique peuvent apprendre à partir de données sans être programmés explicitement, ce qui permet de développer des modèles capables d'évaluer des ensembles de données complexes rapidement et automatiquement [27].

De plus, l'apprentissage automatique contribue à une prise en charge clinique individualisée en fournissant des prévisions sur l'état de santé futur du patient, bien que l'interopérabilité de ces algorithmes puisse être un défi [28].

Dans l'ensemble, l'intérêt croissant pour l'apprentissage automatique est dû à des facteurs tels que la réduction de la puissance de calcul, la disponibilité et le stockage des données. Ce qui en fait un outil important permettant aux entreprises d'acquérir un avantage concurrentiel et de prendre des décisions éclairées.

3- Types d'apprentissage automatique

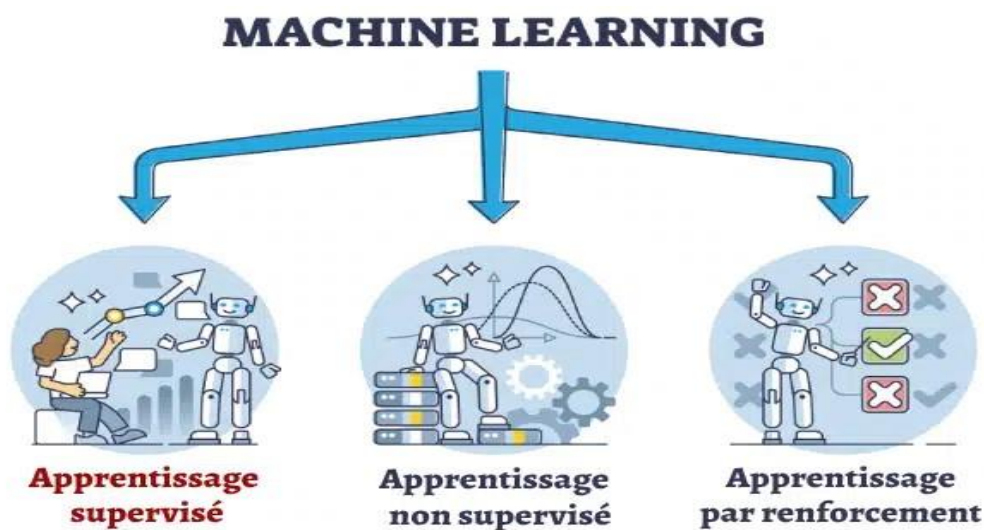


Figure 4 : Types d'apprentissage automatique. [29]

3.1 Apprentissage supervisé

L'apprentissage supervisé est un aspect fondamental de l'apprentissage automatique dans lequel les modèles sont entraînés à l'aide d'ensembles de données étiquetés pour prédire les résultats avec précision. Cette formation consiste à fournir des paires d'entrées-sorties au modèle, ce qui lui permet d'apprendre la relation entre les points de données.

Le processus comprend des phases d'entraînement avec des paires d'entrées-sorties connues et des tests avec des données inédites pour évaluer les performances du modèle [30].

De plus, l'apprentissage supervisé s'étend aux défis de l'exploration de données, tels que la prédiction des valeurs de sortie à partir de vecteurs d'entrée, même lorsque les valeurs de sortie sont agrégées sur plusieurs entrées [31].

Dans l'apprentissage supervisé, les deux principaux types de tâches sont la classification et la régression. La classification attribue des étiquettes aux points de données en fonction de caractéristiques, et la régression prédit des valeurs continues.

- Il existe de nombreux algorithmes d'apprentissage supervisé différents, chacun ayant ses propres avantages et inconvénients. Parmi les algorithmes d'apprentissage supervisé les plus courants, on trouve :

3.1.1 Régression Linéaire

La régression linéaire par apprentissage supervisé consiste à cartographier des paires d'entrées-sorties pour établir une relation. Il s'agit d'un type de modèle de régression utilisé dans l'apprentissage automatique, où la variable de sortie est numérique. La régression linéaire peut être effectuée pour une ou plusieurs variables, en utilisant des techniques telles que la descente de gradient et la régression polynomiale [32].

3.1.2 K-nearest neighbors (KNN)

K-Nearest Nearest Neighbors (KNN) est un algorithme de classification largement utilisé dans l'apprentissage automatique, dans laquelle une instance est classée en fonction de la classe majoritaire de ses plus proches voisins dans l'espace des entités [33].

3.1.3 Arbres de Décision

Les arbres de décision sont une technique d'apprentissage automatique utilisée pour la classification et la régression. Ils sont essentiels dans les scénarios de prise de décision complexes comportant de l'incertitude, des décisions interdépendantes et des périodes de temps multiples, fournissant des solutions optimales. Il s'agit de structures hiérarchiques dont les nœuds représentent les tests d'attributs et les résultats, ce qui facilite les prévisions [34].

3.1.4 Support Vector Machines (SVM)

Les machines à vecteurs de support (SVM) sont un puissant modèle d'apprentissage supervisé utilisé pour la reconnaissance des formes, applicable à la fois aux tâches de classification et de régression. Les SVM créent un hyperplan dans un espace de saisie transformé pour séparer les classes tout en maximisant la marge jusqu'aux exemples les plus proches, dérivé d'un problème d'optimisation de la programmation quadratique [35].

3.2 Apprentissage non supervisé

L'apprentissage non supervisé est une approche d'apprentissage automatique dans laquelle les modèles identifient des modèles dans les données sans variables cibles explicites pour la prédiction. Contrairement à l'apprentissage supervisé, les algorithmes d'apprentissage non supervisé fonctionnent uniquement avec les données d'entrée pour découvrir des corrélations et des informations cachées dans les ensembles de données. Ces algorithmes visent à découvrir les structures inhérentes aux données elles-mêmes, en s'appuyant sur les riches caractéristiques intrinsèques des données [36].

Les méthodes d'apprentissage non supervisé, telles que les techniques de clustering telles que les K-means et le clustering hiérarchique, et les méthodes de réduction de dimensionnalité telles que l'analyse en composantes principales (PCA), jouent un rôle crucial dans la découverte de connaissances en extrayant des informations significatives à partir d'ensembles de données non étiquetés . Ce paradigme d'apprentissage se concentre sur la capture de la structure statistique des modèles d'entrée sans avoir besoin de résultats cibles ou d'évaluations explicites [37].

3.3 Apprentissage par renforcement

L'apprentissage par renforcement est une technique d'apprentissage automatique dans laquelle un agent apprend à prendre des décisions en interagissant avec un environnement et en recevant un feedback sous forme de récompense. Cette approche, inspirée de l'apprentissage par essais et erreurs chez les humains et les animaux, permet à l'agent d'apprendre de manière autonome une politique optimale grâce à une interaction continue avec un environnement dynamique. L'apprentissage par renforcement a trouvé des applications dans divers domaines tels que le jeu vidéo,

contrôle de robots et même le dépassement des performances humaines dans des tâches complexes [38].

4- Apprentissage Profond (deep learning)

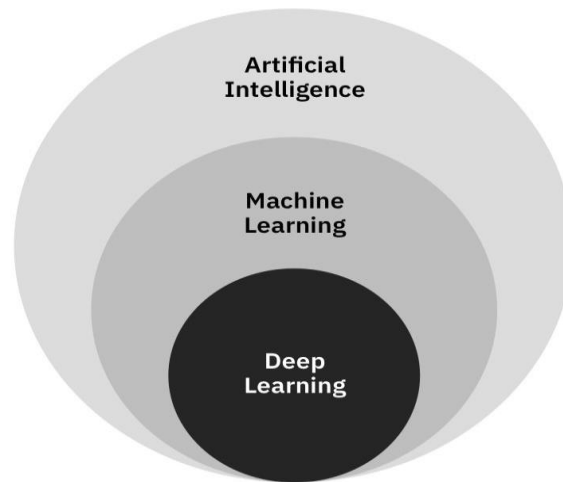


Figure 5 : La relation entre les trois branches l'intelligence artificielle(IA), (ML) et (DL) [39].

L'apprentissage profond est un sous-ensemble de l'apprentissage automatique (Machine Learning) qui utilise des réseaux de neurones profonds pour modéliser des relations complexes entre les données d'entrée et les prédictions de sortie. Ces réseaux, inspirés du cerveau humain, sont constitués de nœuds interconnectés qui traitent les informations.

Tableau 1 : La différence entre (ML) et (DL) [40].

Facteurs	Apprentissage profond	Apprentissage automatique
Données d'exigence	Nécessite des données volumineuses	Peut être réglé de différentes manières
Précision	Fournir une grande précision	Donne moins de précision
Types de données en sortie	Valeurs numériques, Image, Texte, Audio, Voix ...	Valeurs numériques
Temps d'entraînement	Prend du temps à s'entraîner	Prend moins de temps à s'entraîner
Dépendance matérielle	Nécessite un GPU pour s'entraîner correctement	Entraîner sur GPU
Réglage des hyperparamètres	Peut être réglé de différentes manières	Capacité de réglage limitée

4.1 Réseaux de Neurones Artificiels (ANN)

Les réseaux de neurones artificiels (ANN) sont des systèmes informatiques inspirés des réseaux de neurones biologiques, composés de neurones interconnectés qui traitent les informations. Les neurones des ANN sont organisés en couches, y compris les couches d'entrée, les couches cachées et les couches de sortie, avec des connexions (liens) transmettant des informations entre eux [41].

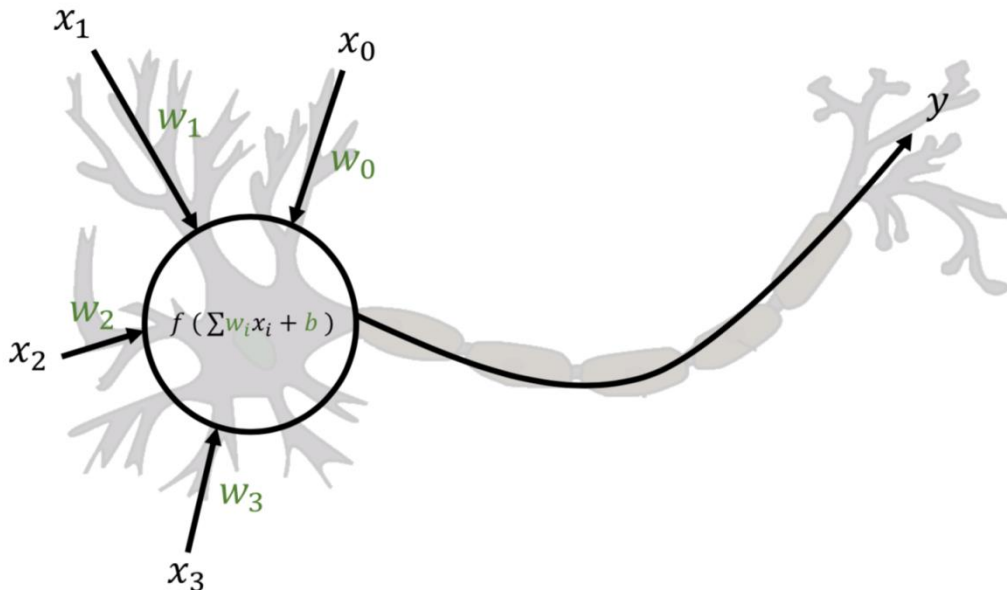


Figure 6 : Neurone informatique superposé à un schéma de neurone biologique. [42]

4.2 Architecture de Réseaux de Neurones Artificiels

Couches : Les neurones artificiels sont organisés en couches. Chaque couche reçoit des entrées de la couche précédente, les traite et produit des sorties qui sont envoyées à la couche suivante.

- Les neurones d'entrée reçoivent des données d'entrée, les traitent et les transmettent aux neurones dans la couche suivante.
- Les neurones cachés prennent les données de sortie des précédents neurones en entrée, calculent de nouvelles données de sortie et les transmettent aux couches suivantes.

- Les neurones de la dernière couche cachée transmettent les données de sortie en entrée des neurones de la couche de sortie afin de produire les données de sortie finale.

Poids : Les poids sont les paramètres des réseaux de neurones artificiels. Ils déterminent la force de la connexion entre les neurones et influencent la façon dont le réseau traite les entrées.

Biais : Les biais sont des constantes ajoutées aux entrées des neurones artificiels. Ils permettent de décaler les sorties des neurones.

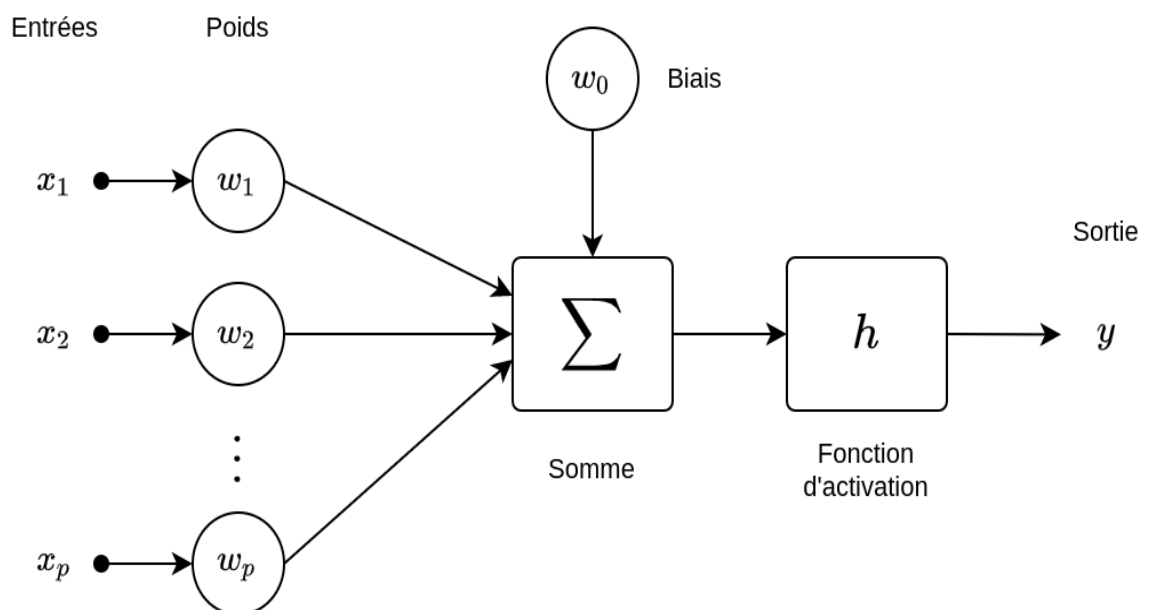


Figure 7 : Architecture de Réseaux de Neurones Artificiels. [43]

Fonction d'activation : est une fonction mathématique appliquée à un signal en sortie d'un neurone artificiel. Elle reproduit le potentiel d'activation biologique du cerveau humain, décidant du passage ou non de l'information en fonction du seuil de stimulation. En pratique, elle détermine si une réponse du neurone est activée. Ainsi, un neurone applique la fonction suivante :

$$X = \sum(\text{entree} \times \text{poids}) + \text{biais}$$

4.3 Les différentes fonctions d'activation

Les différents types de fonctions d'activation utilisées dans les réseaux de neurones artificiels sont variés et chacun présente des caractéristiques spécifiques. Voici quelques types courants de fonctions d'activation :

Linear (Linéaire) : Utilisée en couche de sortie pour des tâches de régression, cette fonction maintient les sorties identiques aux entrées. Son intervalle de sortie est $(-\infty; +\infty)$.

Unité linéaire rectifiée (ReLU) : est la fonction d'activation la plus couramment utilisée en Deep Learning. Elle donne x si x est supérieur à 0, 0 sinon. Autrement dit, c'est le maximum entre x et 0. Son intervalle de sortie $(0; +\infty)$.

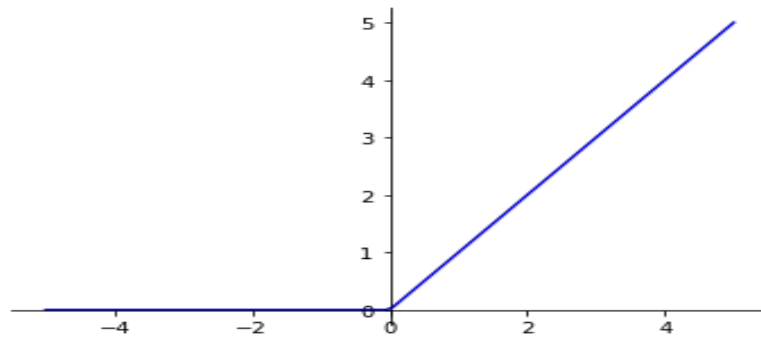


Figure 8 : Fonction ReLU [44]

Sigmoïde : La fonction Sigmoïde est la fonction d'activation employée dans la dernière couche d'un réseau de neurones conçu pour réaliser une tâche de classification binaire. En raison de problèmes de saturation et de gradients nuls, il est moins efficace que d'autres pour les couches cachées. Son intervalle de sortie : $(0 ; 1)$.

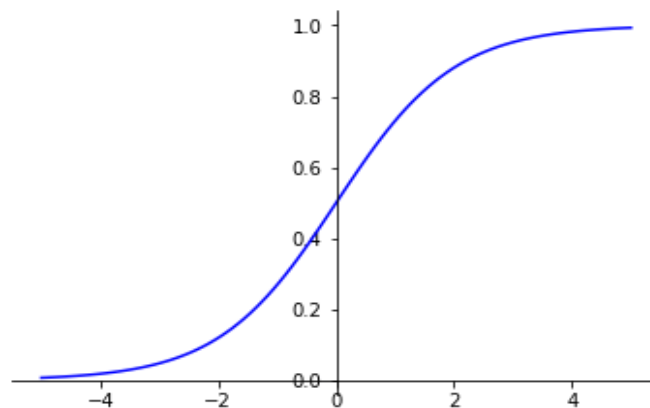


Figure 9 : Fonction Sigmoïde [44]

Tanh : La fonction tanh sert à normaliser les valeurs d'entrée. Elle peut aussi remplacer la fonction Sigmoidé dans la couche finale d'un modèle de classification binaire. Intervalle de sortie : (-1,1).

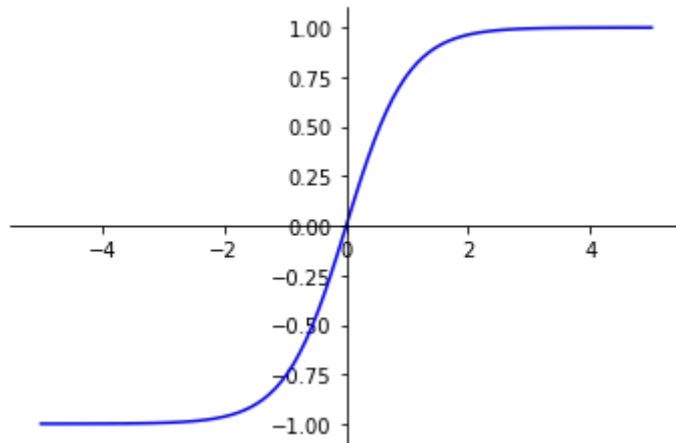


Figure 10 : Fonction Tanh [44]

Softmax : La fonction Softmax est utilisée comme fonction d'activation dans la dernière couche d'un réseau de neurones conçu pour réaliser une classification multi-classes. Intervalle de sortie $(-\infty; +\infty)$.

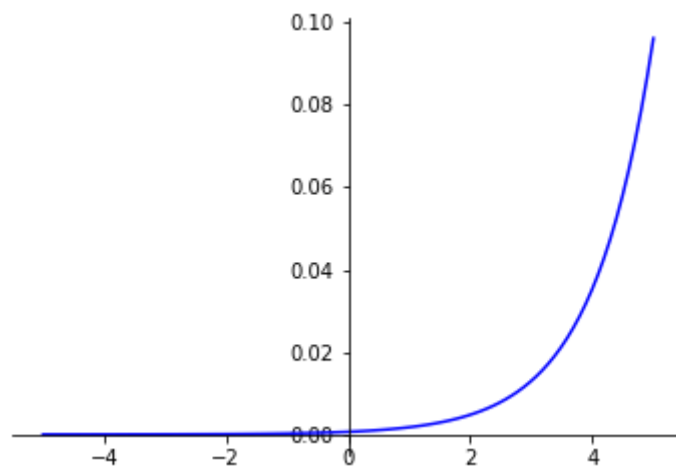


Figure 11 : Fonction Softmax [44]

4.4 Types de réseaux de neurones artificiels

4.4.1 Réseaux de neurones multicouches perceptrons (MLP)

Les MLP sont les réseaux de neurones les plus simples et les plus couramment utilisés. Ils sont composés d'une couche d'entrée, d'une ou plusieurs couches cachées qui jouent un rôle crucial dans les tâches de classification et de régression et d'une couche de sortie. Chaque neurone utilise une fonction d'activation non linéaire, souvent une sigmoïde ou une ReLU, à une somme pondérée de ses entrées par des poids synaptiques. Les neurones d'une couche donnée reçoivent des connexions de tous les neurones de la couche antérieure, sans toutefois être connectés entre eux au sein de la même couche [45].

4.4.2 Réseaux de neurones récurrents (RNN)

Les réseaux de neurones récurrents (RNN) sont une catégorie spéciale de réseaux de neurones artificiels conçus pour traiter des données séquentielles, comme les séries temporelles ou les textes. Leur particularité réside dans leur capacité à mémoriser des informations passées pour influencer les prédictions futures. Les RNN peuvent conserver les informations traitées par leurs nœuds, ce qui permet de nourrir progressivement le modèle avec les résultats accumulés. L'information a la capacité de circuler en boucle, retournant à une couche antérieure pour contribuer à la formation d'une mémoire interne au système [46].

4.4.3 Réseaux de neurones à convolution (CNN)

Les réseaux de neurones à convolution (CNN) sont des architectures spécialisées dans le traitement de données visuelles, sonores ou de parole, grâce à l'utilisation de couches de convolution, de pooling et de couches entièrement connectées. Ils sont particulièrement efficaces pour détecter des motifs et caractéristiques dans les données, ce qui les rend idéaux pour des applications de classification d'images et de vision par ordinateur [47].

4.4.4 Réseaux antagonistes (GAN)

Les réseaux antagonistes génératifs (GAN) sont une classe d'algorithmes en apprentissage non supervisé dans le domaine de l'intelligence artificielle. Ils comprennent deux composantes en interaction compétitive : un générateur et un discriminateur. Le générateur a pour but de créer de nouvelles données, tandis que le discriminateur évalue la qualité de ces données pour déterminer si elles sont réelles ou artificielles. Les deux réseaux s'améliorent continuellement à travers un processus d'entraînement itératif, opérant dans un scénario de jeu à somme nulle où les succès de l'un correspondent aux échecs de l'autre [48].

Chapitre III :
Le Diabète Néonatal
Mellitus Permanent
(PNDM)

1- Introduction

Le diabète néonatal est une maladie génétique rare touchant 1 naissance vivante sur 90 000, caractérisée par une hyperglycémie sévère due à des taux d'insuline faibles ou absents. Il peut se manifester par un diabète néonatal transitoire ou permanent, avec des anomalies génétiques souvent liées à des problèmes structuraux du pancréas ou à un dysfonctionnement des cellules bêta, le diabète néonatal transitoire dont le traitement par l'insuline est souvent arrêté au cours de la première année de vie et le diabète néonatal permanent nécessitant alors un traitement à vie [50].

Cette pathologie se manifeste généralement dans les premières semaines de vie, souvent dans un contexte de retard de croissance intra-utérin. Après une période initiale, la maladie peut entrer en rémission durant quelques mois, avant de réapparaître de manière définitive. La maladie a été signalée dans tous les groupes ethniques et touche aussi bien les garçons que les filles [51].

Le Diabète Néonatal Mellitus Permanent (PNDM) une maladie génétique endocrinienne chronique caractérisée par une hyperglycémie lorsque l'organisme est incapable, soit de produire suffisamment d'insuline, se manifestant pendant la période néonatale, avec une période de rémission pendant la petite enfance suivie d'une récurrence à un stade ultérieur de la vie chez la plupart des patients [51].

Le Diabète Néonatal Permanent (PNDM) représente une forme rare de diabète qui se manifeste dans les premiers mois de vie et qui exige une gestion médicale continue. Ce chapitre vise à explorer les aspects génétiques, cliniques et thérapeutiques du PNDM, en mettant l'accent sur les mutations génétiques spécifiques qui entraînent cette maladie et les options de traitement disponibles selon la mutation identifiée.

2- Etudes Expérimentales

2.1 Étude sur les Mutations Génétiques du Diabète Néonatal Permanent

L'étude de Sara Al-Khawaga et al. [52] sur les caractéristiques cliniques et génétiques du diabète néonatal permanent (DNP) au Qatar a inclus des patients diagnostiqués dans les six premiers mois de vie avec une hyperglycémie persistante nécessitant une

intervention médicale. Les patients ont été identifiés à partir des dossiers médicaux d'hôpitaux et de cliniques spécialisées en endocrinologie pédiatrique. Les données cliniques ont été collectées à partir des dossiers médicaux et comprenaient l'âge au diagnostic, les symptômes cliniques (tels que déshydratation, perte de poids, polyurie), les antécédents familiaux de diabète et les résultats des tests de glycémie. Un examen physique détaillé a également été réalisé pour chaque patient. Pour l'analyse génétique, des échantillons de sang ont été prélevés pour l'extraction de l'ADN. Les échantillons ont été analysés par séquençage Sanger et séquençage de nouvelle génération (NGS) pour identifier les mutations dans les gènes KCNJ11, ABCC8, INS, et GCK.

Les données de séquençage ont été analysées à l'aide de logiciels bio-informatiques pour identifier et confirmer les mutations. Les mutations identifiées par NGS ont été confirmées par séquençage Sanger, et dans certains cas, une étude familiale a été réalisée pour déterminer si les mutations étaient héritées ou de novo. Les résultats ont révélé plusieurs mutations dans les gènes KCNJ11, ABCC8, et INS parmi les patients qataris, avec une réponse positive aux traitements par sulfonyles pour les mutations KCNJ11 et ABCC8, suggérant une alternative aux injections d'insuline. Cette méthodologie combinant la collecte de données cliniques, le séquençage génétique avancé et la validation rigoureuse des résultats a permis une caractérisation détaillée des aspects cliniques et génétiques du DNP au Qatar, fournissant une base solide pour la compréhension et la gestion personnalisée de cette maladie rare.

2.2 Diagnostic du Diabète

Dans l'étude intitulée "Utilisation de l'apprentissage automatique pour détecter précocement le diabète mellitus" réalisée par Luís et al. (2023), un modèle prédictif a été créé afin de pouvoir repérer rapidement les premiers signes du diabète en utilisant des méthodes d'apprentissage automatique. Les chercheurs ont recueilli des données cliniques telles que l'âge, l'indice de masse corporelle (IMC), la glycémie à jeun, la pression artérielle et les antécédents familiaux de diabète. Ensuite, ils ont appliqué différentes techniques d'apprentissage automatique comme les réseaux de neurones. En capturant les relations complexes entre les variables, cela a été démontré que les MLP offraient une précision supérieure dans la prédiction du diabète. Les résultats ont démontré que les MLP avaient la capacité de prédire le diabète avec efficacité, ce qui

permettait une détection précoce et des interventions anticipées. Selon l'étude, l'intégration de ces modèles dans les systèmes de soins de santé pourrait permettre une meilleure prise en charge des patients à risque en offrant des interventions précoces et personnalisées.

3- Étiologie et Génétique du PNDM

3.1 Mutations dans les gènes codant le canal potassique dépendants de l'ATP

Le canal pancréatique, qui est activé par l'ATP, joue un rôle essentiel dans la régulation de la sécrétion d'insuline en réponse à des taux élevés de glucose. On l'appelle canal potassique en raison de la présence d'un pore dans la membrane plasmique qui facilite le passage des ions potassium (K⁺). Kir6.2 est le nom donné aux quatre sous-unités internes qui constituent ce pore. Chacune ayant un site de fixation pour l'ATP, qui favorisent le flux d'ions potassium et donc assurent la conductance potassique. Le canal est aussi constitué de SUR1, qui sont quatre sous-unités externes régulatrices connues pour leur rôle dans la régulation de son activité [54].

Le gène KCNJ11 est responsable de la codification de la sous-unité Kir6.2 du canal potassique, tandis que le gène ABCC8 dirige la synthèse de la sous-unité régulatrice SUR1. Ces gènes sont respectivement situés sur les chromosomes 11 et 8. Les mutations des gènes KCNJ11 et ABCC8 sont fréquemment associées au développement du diabète néonatal permanent [55].

Les mutations dans l'un ou l'autre de ces gènes entraînent une ouverture inappropriée et persistante des canaux potassiques, même en présence d'hyperglycémie. Cette ouverture continue empêche la membrane cellulaire de se dépolariser correctement, ce qui inhibe la capacité de la cellule bêta à émettre un signal nécessaire pour déclencher la libération d'insuline [55].

3.2 Mutations du gène de l'insuline (INS)

La deuxième cause la plus courante de diabète néonatal permanent est liée à des anomalies du gène code pour l'insuline (INS). Les mutations de ce gène provoquent souvent un mauvais repliement de l'insuline, entraînant l'accumulation de

protéines mal formées dans divers compartiments intracellulaires. À long terme, cela peut conduire à la mort des cellules. Généralement, le diagnostic de cette condition est posé avant l'âge de 6 mois, bien que certains cas ne soient diagnostiqués qu'autour de 12 mois ou même plus tard chez les jeunes enfants [56].

3.3 Mutations dans les gènes codant la glucokinase

Le gène GCK code pour la glucokinase, une enzyme essentielle dans le processus de détection du glucose par les cellules bêta du pancréas. Des mutations dans ce gène peuvent être à l'origine du diabète néonatal permanent (PNDM), généralement diagnostiqué au cours des six premiers mois de vie. Les mutations hétérozygotes inactivantes du gène GCK sont associées à un diabète de type 2 précoce, également connu sous le nom de MODY 2 (Maturity-Onset Diabetes of the Young, type 2). Par ailleurs, les mutations homozygotes inactivantes du gène GCK peuvent causer un PNDM [57].

3.4 Anomalies d'une région du chromosome 6

Une des causes du diabète néonatal est liée à la surexpression de gènes situés sur le bras long du chromosome 6, précisément dans la région 6q24. Les individus présentant une anomalie dans cette zone développent généralement un diabète dès les premiers jours ou semaines de vie. Ce diabète peut entrer en rémission au cours de la première année, mais une hyperglycémie réapparaît souvent à l'adolescence et persiste à l'âge adulte dans la plupart des cas [58].

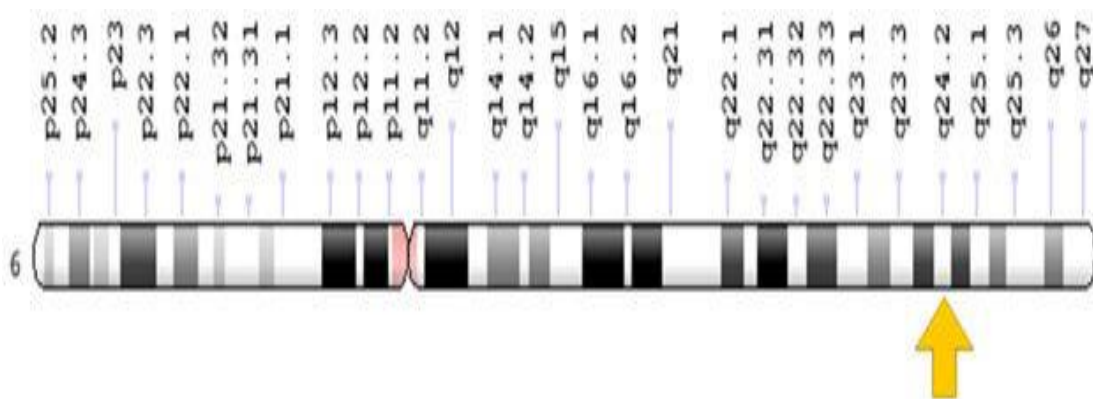


Figure 12 : La position de l'anomalie dans le chromosome 6 [59]

4- Diagnostic du Diabète Néonatal Permanent

4.1 Observation des symptômes cliniques

Les symptômes du diabète néonatal permanent (DNP) apparaissent généralement dans les premiers jours ou les premières semaines de vie. Ils peuvent inclure :

- **Hyperglycémie** : augmentation anormale de la glycémie dans le sang.
- **Polydipsie** : Soif excessive et besoin accru de liquides, difficile à observer directement chez les nourrissons mais pouvant se manifester par une recherche fréquente de nourriture ou un allaitement plus fréquent.
- **Polyurie** : Le bébé urine beaucoup plus que d'habitude, ce qui peut entraîner une déshydratation.
- **Perte de poids**: Le bébé ne prend pas de poids ou perd du poids, même s'il mange suffisamment.
- **Les anomalies congénitales** : les plus fréquentes sont une macroglossie et une hernie ombilicale.
- **Faiblesse et léthargie**: Le bébé semble fatiguer et manque d'énergie.
- **Irritabilité**: Le bébé peut être plus irritable que d'habitude.
- **Peau sèche et muqueuses sèches**: La peau du bébé peut être sèche et craquelée, et sa bouche peut être sèche.

4.2 Tests génétiques

Le diagnostic du diabète néonatal permanent (PNDM) repose sur des tests génétiques visant à identifier des mutations dans des gènes spécifiques tels que GCK, KCNJ11, INS, et ABCC8. Ces analyses sont essentielles, surtout chez les individus issus de familles avec des antécédents de diabète, afin de détecter les mutations génétiques causant le PNDM. Les mutations des gènes KCNJ11 et ABCC8, en particulier, sont liées à une dysfonction des cellules bêta du pancréas et à une production insuffisante d'insuline, symptômes typiques du PNDM. Ainsi, les tests génétiques sont déterminants pour un diagnostic précis du PNDM, permettant une

meilleure orientation du traitement et de la prise en charge des patients affectés par cette maladie rare mais significative [60].

4.3 Antécédents familiaux

L'analyse génétique des patients souffrant de diabète néonatal permanent (PNDM) a mis en évidence diverses mutations, incluant des mutations de novo dans des gènes. De plus, cette étude a mis en lumière l'importance des antécédents familiaux pour le diagnostic et la compréhension de maladies génétiques telles que le PNDM, en particulier dans les populations où la consanguinité est fréquente. Ainsi, une évaluation approfondie des antécédents familiaux est essentielle dans les cas de PNDM pour identifier les causes génétiques potentielles et les schémas héréditaires, ce qui est déterminant pour un diagnostic précis et une prise en charge adéquate de la maladie [61].

5- Approches Thérapeutiques du PNDM

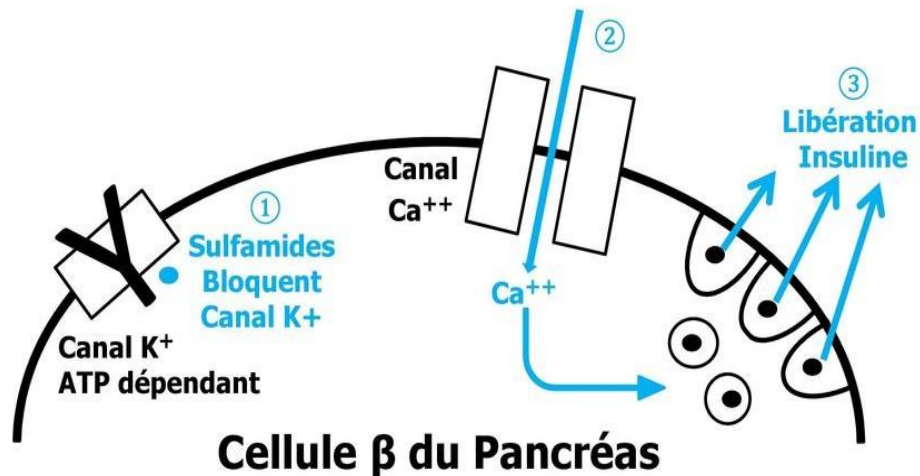
Le traitement du diabète néonatal permanent (DNP) dépend de la cause sous-jacente de la maladie. L'objectif principal du traitement est de normaliser la glycémie et de prévenir les complications.

5.1 Traitement par sulfamide hypoglycémiant

Les approches thérapeutiques du diabète néonatal permanent (PNDM) incluent les sulfonylurées comme le glibenclamide et le répaglinide.

Les sulfamides hypoglycémiants agissent en bloquant les canaux potassiques sensibles à l'ATP des cellules bêta-pancréatiques, induisant ainsi une dépolarisation membranaire qui ouvre les canaux calciques voltage-dépendants. Cette ouverture entraîne une entrée de calcium, déclenchant la libération d'insuline des cellules bêta, en évitant les dommages neurologiques et l'apoptose des cellules bêta lors d'une administration à long terme.

L'initiation précoce d'un traitement à la sulfonylurée est recommandée pour un meilleur contrôle métabolique à long terme dans les cas de PNDM causés par les mutations : KCNJ11 et ABCC8 qui sont les principaux responsables du maladie [62].



Les Sulfamides Hypoglycémiant bloquent les canaux potassiques ATP-dépendants des cellules β pancréatiques

Figure 13 : Mécanisme d'action des sulfamides hypoglycémiant [63]

5.2 Traitement par l'insuline

La plupart des enfants atteints du diabète néonatal permanent ont besoin d'une insulinothérapie substitutive pour contrôler leur glycémie. Cette insuline peut être administrée par injection sous-cutanée ou par pompe à insuline et ne peut pas être prise par voie orale car elle serait détruite par la digestion [64].

L'administration d'insuline rapidement après le diagnostic est cruciale pour prévenir la décompensation métabolique et l'acidocétose diabétique chez les patients [65].

6- Stratégies pour améliorer les Soins et le traitement

6.1 Diagnostic Précoce et Précis

- Afin de protéger la santé des bébés, il est essentiel de mettre en place des programmes de dépistage et diagnostic pour identifier rapidement les nouveau-nés à risque ou présentant déjà des symptômes de diabète néonatal. Ces programmes permettront de détecter précocement cette maladie et d'agir vite pour prendre soin de ces tout-petits.

- Utilisation des tests génétiques qui sont un outil précieux pour identifier les mutations spécifiques associées au diabète néonatal. Grâce à ces analyses, les médecins pourront poser un diagnostic rapide et fiable, ce qui est crucial pour prendre en charge au mieux ces bébés malades.

6.2 Prise en charge médicale

- Utiliser des dispositifs de surveillance continue de la glycémie sera d'une grande aide pour surveiller de près la glycémie et ajuster les traitements en temps réel.
- Élaborer des directives pour le traitement clinique fondé sur des données scientifiques probantes du diabète rare chez les nouveau-nés.

6.3 Recherche et développement

- Établir des bases de données pour collecter des données sur les patients atteints de diabète néonatal rare peut aider à améliorer la compréhension de cette maladie et à développer de meilleures stratégies de traitement.
- Soutenir la recherche sur les causes génétiques et les mécanismes du diabète néonatal pour développer de nouveaux traitements plus efficaces.

6.4 Soutien et éducation

- Développer des programmes éducatifs pour informer les patients et les familles sur la maladie, ses options de traitement et sa prise en charge.
- Fournir un soutien psychologique aux parents pour les aider à faire face au stress et aux défis liés à la gestion du diabète néonatal.

Chapitre IV :
Étude
Computationnelle

1- Introduction

Dans ce chapitre, nous examinons en détail la prédiction de la maladie rare (PNDM) en utilisant les méthodes d'apprentissage profond. Dans un premier temps, nous exposons en détail le matériel utilisé dans notre étude, incluant les données cliniques recueillies. Par la suite, nous présentons de manière approfondie la méthode utilisée, en mettant en évidence l'emploi d'un réseau de neurones feedforward, plus précisément un perceptron multicouche MLP (Multi-Layer *Perceptron*) basé sur le modèle séquentiel de Keras. Ensuite, nous discutons des résultats obtenus à travers notre modèle et de leur importance dans la prédiction de la maladie PNDM.

2- Matériel

2.1 Données cliniques

Notre travail repose sur l'utilisation de données cliniques telles que des évaluations médicales et des examens liés à la maladie PNDM :

Les données utilisées ont été obtenues à partir de la plateforme Web Kaggle. Il s'agit d'un ensemble des évaluations médicales du PNDM simulées qui sont basées sur des données médicales réelles et des recherches novatrices. Ces données complètes incluent six caractéristiques qui pourraient contribuer à prédire le PNDM : Âge, taux d'HbA1c, historique familial, poids de naissance, taux d'insuline, retard de développement, données génétiques (mutation). La colonne de la mutation génétique a été supprimée afin de faciliter le diagnostic et le patient n'a pas à faire le séquençage de l'ADN. Les données sont regroupées dans un seul fichier CSV qui regroupe les données de 100 000 individus.

Les données sont disponibles à travers le lien suivant:

<https://www.kaggle.com/datasets/slmsshk/pndm-prediction-dataset/data>

Le tableau ci- dessous récapitule les diverses statistiques sur les données utilisées.

Tableau 2 : Statistiques sur les données utilisées.

Caractéristiques	Min	Max	Moy	Individus
Age (années)	1	11	6.0005	100.000
HbA1c (DCCT unit)	2.8306 %	11.3152 %	6.9967 %	100.000
Niveau d'insuline (iU/ml)	3.2420	13.3044	5.0030	100.000
Poids de naissance (Kg)	0.4271	4.9104	2.8015	100.000
Historique familial	Variables catégorielles (Oui et Non)			Oui = 9976 Non = 90024
Retard du développement				Oui = 14899 Non = 85101
PNDM	0	1	0.0482	Malade = 4822 Non-malade = 95178

- Description des caractéristiques et leur importance à prédire le PNDM :
 - Age: L'âge joue un rôle crucial dans le diagnostic du PNDM et des différentes formes de diabète : Les conséquences de l'âge au diagnostic du (PNDM) et de tous les types de diabète sont significatives pour les résultats et la prise en charge des patients. Il revêt aussi une importance cruciale car il a un impact sur la gravité et la progression de la maladie. Les conclusions indiquent qu'un diagnostic plus précoce du diabète est lié à des résultats moins favorables, tels qu'un risque plus élevé de problèmes cardiovasculaires et de décès [66]. Cette relation perdure même en prenant en considération la durée du diabète [66]. On ne comprend pas

encore pleinement les mécanismes qui sous-tendent cette relation, mais ils peuvent inclure des éléments tels qu'une durée plus longue du diabète, des formes plus agressives de diabète et un risque accru de dysfonction vasculaire, un mauvais contrôle de la glycémie et une résistance à l'insuline [66][67]. En outre, les individus plus jeunes diagnostiqués avec le diabète peuvent nécessiter un traitement et une surveillance plus stricts en raison du risque accru de complications [68]. De plus, l'âge au moment du diagnostic peut avoir un impact sur la sélection des stratégies de traitement et l'importance accordée aux mesures préventives telles que le dépistage et la prévention des complications macro vasculaires [68].

- **HbA1c:** L'importance de la connaissance de l'HbA1c dans le diagnostic du PNDM réside dans sa capacité à fournir une mesure à long terme du contrôle glycémique [69]. HbA1c est un outil de diagnostic précieux dans le PNDM, car il reflète la concentration plasmatique moyenne de glucose sur une période de semaines à mois, ce qui est particulièrement pertinent chez les nouveau-nés où les niveaux de glucose peuvent fluctuer rapidement [69]. L'HbA1c peut aider à diagnostiquer la PNDM à la naissance ou peu après, ce qui permet d'initier rapidement la thérapie par insuline et de réduire le risque d'hyperglycémie sévère et de complications associées [70]. Les mesures régulières de l'HbA1c permettent aux fournisseurs de soins de santé de surveiller l'efficacité de la thérapie par insuline et d'ajuster les régimes de traitement en conséquence, assurant un contrôle glycémique optimal et réduisant au minimum le risque de complications à long terme [70]. En outre, il peut aider à différencier le PNDM d'autres formes de diabète, comme le diabète sucré néonatal transitoire (TNDM), qui se caractérise par une carence temporaire de sécrétion d'insuline [70].
- **Niveau d'insuline:** Les valeurs d'insuline sont essentielles pour le diagnostic et la prise en charge du diabète, y compris le PNDM. Les taux d'insuline peuvent permettre de détecter la présence d'une hypoglycémie, qui est une caractéristique des patients atteints de PNDM. En général, les niveaux d'insuline sont très bas ou inobservables dans le PNDM en raison de la totale absence de sécrétion d'insuline [71]. Les professionnels de santé peuvent bénéficier d'une surveillance régulière des niveaux d'insuline afin d'évaluer l'efficacité de la thérapie par insuline et

d'adapter les régimes de traitement en conséquence afin d'assurer un contrôle glycémique optimal [71].

- Poids de naissance et retard du développement : Dans la PNDM, la relation entre le poids à la naissance et le retard de développement a des conséquences significatives pour le diagnostic et la prise en charge. De nombreuses recherches ont démontré que les patients souffrant de PNDM ont un poids de naissance inférieur à celui des patients souffrant de diabète sucré néonatal transitoire (TNDM) [72][73]. Cela suggère que la gravité de la maladie est liée au degré de retard de croissance intra-utérine, qui se reflète dans le poids de naissance. Des poids de naissance plus faibles sont associés à une insuffisance plus sévère de l'insuline et à un risque plus élevé de retard du développement pendant la période critique du développement du cerveau, y compris des retards cognitifs et moteurs [72][73].
- Histoire familiale : Le diagnostic du diabète sucré néonatal permanent (PNDM) est influencé par l'histoire familiale. En cas de PNDM, l'histoire familiale peut être utile pour obtenir des informations précieuses sur la base génétique de la maladie. Selon les études, il a été démontré que des mutations dans des gènes tels que KCNJ11, INS, ABCC8 et GCK peuvent entraîner la PNDM [74] [75]. De plus, les antécédents familiaux sont parfois trompeurs, comme on le constate dans les cas où l'historique familial initial n'a pas confirmé le diagnostic de diabète monogénique jusqu'à ce que l'analyse moléculaire découvre des mutations de novo [76]. En outre, une recherche sur les patients diabétiques de type 1 a mis en évidence que l'existence d'une histoire familiale positive du diabète du type 2 peut influencer le phénotype et le génotype du diabète de type 1, ce qui entraîne des variations d'âge au moment du diagnostic et des symptômes métaboliques [77]. La connaissance de l'histoire familiale est donc cruciale pour le diagnostic et la prise en charge de la PNDM et d'autres formes de diabète monogénique.

2.2 Configuration de la machine

Le tableau ci-dessous présente les caractéristiques de l'ordinateur utilisé

Tableau 3 : Les caractéristiques de l'ordinateur utilisé

Ordinateur	Caractéristiques
Processeur	Intel(R) Celeron(R) CPU N3060 @ 1.60GHz 1.60GHz
Mémoire installée RAM	4.00 GB
Stockage	500 Go
Système d'exploitation	Windows 10 professionnel
Type de système	64-bit Operating System, x64-based processor

2.3 Technologies utilisées pour la création du modèle et le développement de la plateforme

2.3.1 Environnements de travail

- Visual Studio Code 1.89.0 : Visual Studio Code (VS Code) est un éditeur de code source et un environnement de développement intégré (IDE) de Microsoft. Il est open source et cross-Platform, c'est-à-dire qu'il fonctionne sur Windows, Linux et Mac. Il a été conçu pour les développeurs Web, mais il prend en charge de nombreux autres langages de programmation tels que C++, C#, Python, Java, etc. Il offre de nombreuses fonctionnalités comme la coloration syntaxique, l'auto-complétion, la mise en évidence des erreurs, la navigation de code, le débogage, la gestion de versions, l'intégration avec Git, et beaucoup d'autres [78].
- Google Colab: Google Colaboratory, communément appelé Google Colab, a été développé par Google Research en 2017. Colab est un service hébergé de notebooks Jupyter qui ne nécessite aucune configuration et qui permet d'accéder

gratuitement à des ressources informatiques, y compris des CPU, des GPU et des TPU. Colab est particulièrement adapté à l'apprentissage automatique et l'apprentissage profond [79].

2.3.2 Langages de programmation

- Python 3.10 : Python a été développé par la Python Software Foundation en 1990 et est un langage de programmation open source. Il s'agit d'un langage de programmation robuste et accessible à l'apprentissage. Son équipement comprend des structures de données de qualité supérieure et offre une approche facile mais efficace de la programmation orientée objet. Python est un langage parfait pour l'écriture de scripts et le développement rapide d'applications dans de nombreux domaines et sur la plupart des plateformes, car sa syntaxe est élégante, son typage dynamique et son interprétation pratique [80].
- HTML 5 : Hypertext Mark-up Language, est un langage informatique généraliste. Il repose sur le langage SGM (Standard Generalized Markup Language) [81]. HTML ne s'agit pas d'un langage de programmation spécifique, mais plutôt d'un langage qui permet de structurer du contenu [82].
- CSS 3 : Cascading Style Sheets est un langage informatique qui décrit la présentation des documents HTML et XML. Il offre la possibilité de définir les normes de conception et de style pour les différents éléments d'une page Web, tels que la structure, la couleur, la police, les marges, le rembourrage, les bordures, les animations, et bien d'autres encore. [83] [84].
- JS: JavaScript est un langage de programmation informatique. C'est une forme de code qui permet de créer un contenu plus dynamique, plus animé sur une page Web. Dans le développement web, le JavaScript permet aux pages web de disposer d'une meilleure réactivité et interactivité [85].

2.3.3 Base de données

- PostgreSQL: Est un système de gestion de base de données relationnelle orienté objet puissant et open source qui peut gérer en toute sécurité les données les plus complexes. La conformité et l'extensibilité SQL sont les priorités de PostgreSQL [86].

2.3.4 Frameworks

- Django 5.0.1 : Est un Framework Web Python open source de haut niveau conçu pour un développement rapide, offrant une approche de conception cohérente et réaliste [87]. Il est connu pour sa simplicité, sa stabilité, son évolutivité et sa flexibilité, ce qui en fait un choix populaire parmi les développeurs pour la création d'applications Web dynamiques [88]. Les caractéristiques clés de Django comprennent un design robuste, un développement logiciel rapide, une excellente documentation et une courbe d'apprentissage relativement facile [89]. En outre, Django est utilisé pour créer des plates-formes efficaces et conviviales, intégrant des principes d'ingénierie logicielle et diverses technologies comme Python, HTML, CSS et PostgreSQL [90].

Voici une explication succincte des fichiers clés dans une application Django :

- Le fichier `settings.py` : Ce fichier regroupe les paramètres de configuration essentiels de l'application Django, notamment la configuration de la base de données, les clés secrètes, les fichiers statiques et les fichiers médias [91].
- Le fichier `urls.py` : Ce fichier est dédié à la gestion des URLs (Uniform Resource Locators) de l'application Django. Il établit les correspondances entre les URLs et les vues associées, ces dernières étant responsables du traitement des requêtes HTTP et de la génération des réponses [91].
- Les fichiers de modèles (`models.py`) : Ces fichiers définissent les modèles de données propres à l'application. Ils spécifient la structure des tables dans la base de données ainsi que les relations entre celles-ci [91].

- Les fichiers de vues (views.py) : Ces fichiers contiennent les logiques de traitement des requêtes et la préparation des données à afficher. Les vues sont généralement chargées de renvoyer des réponses HTTP ou de rediriger vers d'autres vues [91].
- Les fichiers de templates (.html) : Ces fichiers définissent la structure et l'apparence des pages Web via des modèles HTML. Ils peuvent intégrer des données dynamiques fournies par les vues [91].
- Les fichiers de migrations : Situés dans le répertoire "migrations", ces fichiers sont générés automatiquement par Django lorsqu'il détecte des modifications dans les modèles de données. Ils facilitent la mise à jour transparente de la structure de la base de données [91].

Ces fichiers constituent une composante cruciale de l'architecture d'un projet Django, mais il est possible qu'il y ait d'autres fichiers et répertoires selon les besoins spécifiques de l'application. En outre, Django propose une pléthore de fonctionnalités supplémentaires telles que l'administration automatique, la gestion des formulaires, l'authentification des utilisateurs, etc. ce qui en fait un choix prisé pour le développement Web [91].

- Bootstrap: Est un Framework front-end qui rationalise la création de sites Web réactifs et adaptés aux mobiles [92]. Il offre un ensemble d'outils complet comprenant des outils HTML, CSS et JavaScript, ce qui le rend populaire pour son adaptabilité, le soutien du navigateur et les composants réutilisables [93]. Le système de grille de Bootstrap simplifie la configuration et le style de la mise en page pour divers éléments HTML, améliorant la polyvalence et la compatibilité avec les plug-ins JavaScript [94].

2.3.5 Bibliothèques

La liste des bibliothèques python utilisées dans notre travail est présentée dans le tableau ci-dessous, avec leur description et leur version correspondante.

Tableau 4 : Les bibliothèques python utilisées.

Bibliothèque	Description
Numpy 1.26.4	Numpy est le package fondamental pour l'informatique scientifique en Python. C'est une bibliothèque Python qui fournit un objet d'array multidimensionnel, divers objets dérivés (tels que les array masqués et les matrices) et une gamme de routines pour des opérations rapides sur les arches, y compris mathématiques, logique, manipulation de la forme, tri, sélection, I/O, transformations de Fourier discrètes, algèbre linéaire de base, opération statistique de base et simulation aléatoire et bien plus encore [95].
Pandas 2.2.2	Pandas est une bibliothèque open source sous licence BSD qui fournit des structures de données hautes performances et faciles à utiliser et des outils d'analyse de données pour le langage de programmation Python [96].
Matplotlib 3.8.4	Matplotlib est une bibliothèque complète pour la création de visualisations statiques, animées et interactives en Python. Matplotlib produit des chiffres de qualité de publication dans une variété de formats de copie et d'environnements interactifs à travers les plates-formes. Matplotlib peut être utilisé dans des scripts Python, des shells Python/IPython, les serveurs d'applications Web et divers outils d'interface graphique utilisateur [97].
TensorFlow-CPU 2.10	TensorFlow est une bibliothèque de logiciels open source pour les calculs numériques de haute performance. Son architecture flexible permet un déploiement facile de l'informatique sur une variété de plateformes (CPU, GPU, TPU) et des ordinateurs de bureau à des groupes de serveurs, aux appareils mobiles et aux périphériques de pointe. Développé à l'origine par des chercheurs et des ingénieurs de l'équipe de Google Brain au sein de la société d'IA de Google, il apporte un soutien solide pour l'apprentissage automatique, l'apprentissage profond et le noyau de calcul numérique flexible est utilisé dans de nombreux autres domaines scientifiques [98].

Keras 3.3.3	<p>Keras est une bibliothèque open source de calcul de réseaux neuronaux, principalement développée en Python. Elle propose une interface de haut niveau extrêmement modulaire et extensible, permettant ainsi la création et la formation de modèles sans avoir besoin d'une expertise approfondie [99]. Il est composé d'un frontend comprenant des fonctionnalités d'interface unifiées et d'une base qui interfère avec des frameworks d'apprentissage profond tels que TensorFlow, Theano et CNTK [100].</p>
Scikit-learn 1.4.2	<p>C'est une bibliothèque Python qui permet d'accéder à des implémentations efficaces de nombreux algorithmes communs. Il offre également une API propre et cohérente. Par conséquent, l'un des grands avantages de Scikit-Learn est qu'une fois que vous avez compris l'usage et la syntaxe de base de Scikit-Learn pour un type de modèle, la transition vers un nouveau modèle ou un nouvel algorithme est très simple. La bibliothèque permet non seulement de modéliser, mais peut également gérer les étapes de prétraitement [101].</p>
Imbalanced-learn 0.12.2	<p>Imbalanced Learn est une bibliothèque Python open source. Il propose une variété complète de méthodes de rééchantillonnage, de sous-échantillonnage, d'excédent et de combinaison hybride afin d'équilibrer les classes et d'améliorer les performances des modèles. Les techniques avancées pour traiter les classes minoritaires et majoritaires incluent le SMOTE (Synthetic Minority Over-Sampling Technique), le Random Under-Sampling et d'autres exemples [102].</p>
Joblib 1.4.2	<p>Joblib est une bibliothèque Python pour exécuter en parallèle des tâches informatiques intensives. Il offre un ensemble de fonctionnalités permettant d'effectuer des opérations simultanées sur de vastes ensembles de données et de cacher les résultats des fonctions calculées coûteuses. Joblib est très pratique pour les modèles d'apprentissage automatique, car il vous offre la possibilité d'enregistrer l'état de votre calcul et de reprendre votre travail ultérieurement ou sur une autre machine [103].</p>
Shutil 0.1.0	<p>Le module Shutil propose plusieurs fonctionnalités avancées pour manipuler les fichiers et les collections de fichiers. Plus précisément, des fonctionnalités sont proposées permettant la copie et la suppression de fichiers [104].</p>

3- Modèle d'apprentissage profond pour la prédiction de PNDM

Dans cette section, nous exposons les diverses approches employées afin d'atteindre l'objectif de notre approche.

3.1 L'importation des bibliothèques Python nécessaire

Commençons par importer les bibliothèques Python requises en utilisant les commandes indiquées dans la figure 14.

```
[ ] # necessary libraries
import matplotlib.pyplot as plt
import tensorflow as tf
import pandas as pd
import numpy as np
import joblib
import shutil

[ ] from sklearn.metrics import accuracy_score, f1_score, precision_score, recall_score, roc_auc_score
from sklearn.metrics import precision_recall_curve
from sklearn.preprocessing import StandardScaler
from tensorflow.keras.models import save_model
from sklearn.metrics import confusion_matrix
from sklearn.model_selection import KFold
from imblearn.over_sampling import SMOTE
from sklearn.metrics import roc_curve
from sklearn.utils import shuffle
from tensorflow import keras
```

Figure 14 : Code python pour l'importation des bibliothèques nécessaires.

3.2 Prétraitement des données

Différentes fonctions sont employées pour filtrer et nettoyer les données, ce qui les rend plus adaptées. Les données utilisées ont été obtenues à partir de la plateforme Web Kaggle sous la forme d'un fichier CSV que nous avons téléchargé et placé dans un répertoire GitHub. Voici les étapes principales :

- Collecte et nettoyage des données: Dans cette partie, nous commençons par charger les données à partir de l'URL du répertoire GitHub. Nous utilisons la fonction `read_csv()` de la bibliothèque Pandas pour charger les données depuis le fichier CSV situé à l'URL spécifiée. Les données sont stockées dans la variable `PNDM_dataset`. Ensuite, nous affichons les cinq premières lignes à l'aide de la méthode `head()`. Cela nous donne un aperçu initial des données pour nous assurer qu'elles ont été chargées correctement (figure 15).

```
[ ] # Load data
file_url = 'https://raw.githubusercontent.com/G-habib/pfe_models/main/PNDM.csv'
PNDM_dataset=pd.read_csv(file_url)

[ ] PNDM_dataset.head()
```

	Age	HbA1c	Genetic Info	Family History	Birth Weight	Developmental Delay	Insulin Level	PNDM
0	3	4.840927	Mutation	Yes	3.128268	No	5.585608	0
1	3	5.694742	Mutation	No	2.059342	No	3.141359	1
2	7	6.843595	No mutation	No	2.718667	Yes	4.639313	0
3	2	6.480186	No mutation	No	3.087017	No	6.217178	0
4	4	7.052861	Mutation	No	3.481472	No	3.368892	0

Figure 15 : Code python pour Collecte et nettoyage des données.

Nous utilisons la méthode `drop_duplicates ()` pour supprimer les lignes en double dans le fichier des données, en conservant uniquement la première occurrence de chaque ligne dupliquée. Cela garantit que chaque entrée est unique. Enfin, nous affichons les données dans son intégralité pour vérifier que les données ont été correctement chargées et nettoyées (figure 16).

```
[ ] PNDM_dataset.drop_duplicates(keep='first', inplace=True)
PNDM_dataset
```

	Age	HbA1c	Genetic Info	Family History	Birth Weight	Developmental Delay	Insulin Level	PNDM
0	3	4.840927	Mutation	Yes	3.128268	No	5.585608	0
1	3	5.694742	Mutation	No	2.059342	No	3.141359	1
2	7	6.843595	No mutation	No	2.718667	Yes	4.639313	0
3	2	6.480186	No mutation	No	3.087017	No	6.217178	0
4	4	7.052861	Mutation	No	3.481472	No	3.368892	0
...
99995	9	7.810662	No mutation	Yes	2.031233	No	3.059450	0
99996	11	7.163781	Mutation	Yes	3.003822	No	2.891719	0
99997	10	5.521820	No mutation	No	3.798452	Yes	6.015360	0
99998	10	7.539210	No mutation	No	3.117501	No	4.128981	0
99999	8	6.976317	No mutation	No	2.882062	No	3.179945	0

100000 rows x 8 columns

Figure 16 : Code python pour supprimer les lignes en double.

- Standardisation des variables catégorielles en valeurs numériques : Dans cette section, nous identifions les caractéristiques catégorielles qui doivent être converties en valeurs numériques. Les caractéristiques sélectionnées sont "Genetic Info" (informations génétiques), "Family History" (antécédents familiaux) et "Developmental Delay" (retards de développement). Nous utilisons la méthode `get_dummies()` de Pandas pour créer des variables factices (dummy variables) à partir des caractéristiques catégorielles. Cela convertit chaque valeur unique dans les caractéristiques catégorielles en une nouvelle colonne avec des valeurs booléennes (True/False). Ensuite, nous fusionnons les nouvelles colonnes créées avec les données originales à l'aide de la méthode `concat()`. Cela nous donne des données étendues avec les variables factices ajoutées. Nous supprimons ensuite les caractéristiques catégorielles d'origine des données, car elles ont été remplacées par les variables factices (Figure 17).

```
[ ] # standardization
# normalisation of Genetic Info, Family history, Developmental delay : into numerical values
cat_feature = ['Genetic Info', 'Family History', 'Developmental Delay']
dummy_PNDM_dataset = pd.get_dummies(PNDM_dataset[cat_feature])
PNDM_dataset = pd.concat([PNDM_dataset, dummy_PNDM_dataset],axis=1)
PNDM_dataset = PNDM_dataset.drop(cat_feature, axis=1)
PNDM_dataset.shape
```

(100000, 11)

```
[ ] PNDM_dataset.head()
```

	Age	HbA1c	Birth Weight	Insulin Level	PNDM	Genetic Info_Mutation	Genetic Info_No mutation	Family History_No	Family History_Yes	Developmental Delay_No	Developmental Delay_Yes
0	3	4.840927	3.128268	5.585608	0	True	False	False	True	True	False
1	3	5.694742	2.059342	3.141359	1	True	False	True	False	True	False
2	7	6.843595	2.718667	4.639313	0	False	True	True	False	False	True
3	2	6.480186	3.087017	6.217178	0	False	True	True	False	True	False
4	4	7.052861	3.481472	3.368892	0	True	False	True	False	True	False

Figure 17 : Code python pour Standardisation des variables catégorielles en valeurs numériques.

Enfin, nous remplaçons les valeurs booléennes (True/False) dans les colonnes nouvellement créées par des valeurs numériques (1 pour True et 0 pour False). Cela assure que toutes les valeurs sont numériques. Nous vérifions la forme (shape) des données après ces modifications pour nous assurer que tout s'est déroulé comme prévu (Figure 18).

```
[ ] PNDM_dataset[['Genetic Info_Mutation', 'Genetic Info_No mutation', 'Family History_No', 'Family History_Yes',
'Developmental Delay_No', 'Developmental Delay_Yes']] = PNDM_dataset[['Genetic Info_Mutation',
'Genetic Info_No mutation', 'Family History_No', 'Family History_Yes',
'Developmental Delay_No', 'Developmental Delay_Yes']].replace({True: 1, False: 0})
```

```
[ ] PNDM_dataset.to_csv('PNDM_2.csv', index=False)
PNDM_dataset_2=pd.read_csv('PNDM_2.csv')
PNDM_dataset_2.head()
```

	Age	HbA1c	Birth Weight	Insulin Level	PNDM	Genetic Info_Mutation	Genetic Info_No mutation	Family History_No	Family History_Yes	Developmental Delay_No	Developmental Delay_Yes
0	3	4.840927	3.128268	5.585608	0	1	0	0	1	1	0
1	3	5.694742	2.059342	3.141359	1	1	0	1	0	1	0
2	7	6.843595	2.718667	4.639313	0	0	1	1	0	0	1
3	2	6.480186	3.087017	6.217178	0	0	1	1	0	1	0
4	4	7.052861	3.481472	3.368892	0	1	0	1	0	1	0

Figure 18 : Code python pour remplacer les valeurs booléennes par des valeurs numériques.

Séparation des données en caractéristiques et en colonne de sortie : Dans cette partie, nous supprimons les colonnes "Genetic Info_Mutation" et "Genetic Info_No mutation". Ces colonnes ont été supprimées pour simplifier le processus de diagnostic et le patient ne fait pas un séquençage de l'ADN. Ensuite, nous enregistrons les données modifiées dans un nouveau fichier CSV appelé "PNDM_3.csv". Cela nous permet de conserver une copie des données avec les colonnes supprimées pour une utilisation ultérieure, si nécessaire. Nous chargeons ensuite le nouveau fichier CSV "PNDM_3.csv" dans un nouveau DataFrame appelé "PNDM_dataset_3". Nous affichons ensuite les cinq premières lignes du DataFrame pour vérifier que les colonnes ont été correctement supprimées (Figure 19).

```
[ ] PNDM_dataset_2.drop(columns=['Genetic Info_Mutation', 'Genetic Info_No mutation'], inplace=True)
PNDM_dataset_2.to_csv('PNDM_3.csv', index=False)
PNDM_dataset_3=pd.read_csv('PNDM_3.csv')
PNDM_dataset_3.head()
```

	Age	HbA1c	Birth Weight	Insulin Level	PNDM	Family History_No	Family History_Yes	Developmental Delay_No	Developmental Delay_Yes
0	3	4.840927	3.128268	5.585608	0	0	1	1	0
1	3	5.694742	2.059342	3.141359	1	1	0	1	0
2	7	6.843595	2.718667	4.639313	0	1	0	0	1
3	2	6.480186	3.087017	6.217178	0	1	0	1	0
4	4	7.052861	3.481472	3.368892	0	1	0	1	0

Figure 19 : Code python pour supprimée les colonnes (Mutation).

Enfin, nous séparons les données (caractéristiques) de l'étiquette (label). Nous stockons les caractéristiques dans la variable X en supprimant la colonne 'PNDM' du DataFrame (Figure 20). La variable Y contient les étiquettes, c'est-à-dire les valeurs de la colonne 'PNDM' (Figure 21).

```
[ ] # separating the data and label
X = PNDM_dataset_3.drop(columns = 'PNDM', axis=1)
X = PNDM_dataset_3.drop(columns = 'PNDM', axis=1)
Y = PNDM_dataset_3['PNDM']
```

	Age	HbA1c	Birth Weight	Insulin Level	Family History_No	Family History_Yes	Developmental Delay_No	Developmental Delay_Yes
0	3	4.840927	3.128268	5.585608	0	1	1	0
1	3	5.694742	2.059342	3.141359	1	0	1	0
2	7	6.843595	2.718667	4.639313	1	0	0	1
3	2	6.480186	3.087017	6.217178	1	0	1	0
4	4	7.052861	3.481472	3.368892	1	0	1	0
...
99995	9	7.810662	2.031233	3.059450	0	1	1	0
99996	11	7.163781	3.003822	2.891719	0	1	1	0
99997	10	5.521820	3.798452	6.015360	1	0	0	1

Figure 20 : Code python pour la séparation des données.

```
[ ] Y
```

0	0
1	1
2	0
3	0
4	0
..	
99995	0
99996	0
99997	0
99998	0
99999	0

Name: PNDM, Length: 100000, dtype: int64

Figure 21 : Les valeurs de la colonne 'PNDM' (l'étiquette).

- Équilibrage des données : Tout d'abord, nous visualisons la distribution des classes de la variable cible ('PNDM') à l'aide d'un diagramme à barres (Figure 22).

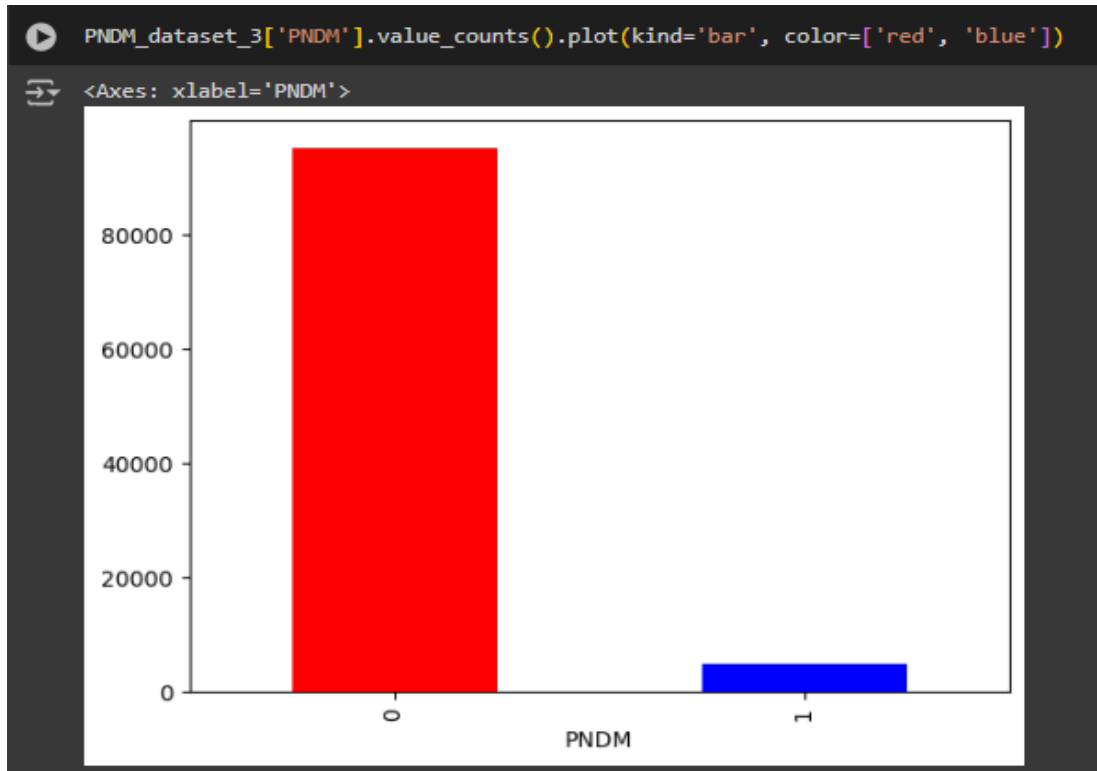


Figure 22 : Visualisons la distribution des classes.

Et nous avons remarqué que l'ensemble des données n'est pas équilibré, le nombre de négatifs (0 = 95178) est beaucoup plus grand que le positif (1 = 4822), donc l'équilibrage du groupe de données doit être fait.

Ensuite, nous utilisons la technique de sur échantillonnage SMOTE (Synthetic Minority Over-sampling Technique) pour équilibrer les données. Nous créons une instance de la classe SMOTE avec un paramètre de `random_state` fixé à 101 pour assurer la reproductibilité des résultats. Ensuite, nous appliquons la méthode `fit_resample()` de SMOTE sur les caractéristiques (X) et les étiquettes (Y) pour obtenir des données équilibrées. Enfin, nous visualisons à nouveau la distribution des classes de la variable cible ('PNDM') dans les données équilibrées à l'aide d'un diagramme à barres. Cette ligne de code compte le nombre d'occurrences de chaque classe ('0' et '1') dans les étiquettes équilibrées (Y_balanced) et trace ensuite un

graphique de barres pour illustrer cette distribution. Les classes sont colorées en rouge ('0') et bleu ('1') (Figure 23).

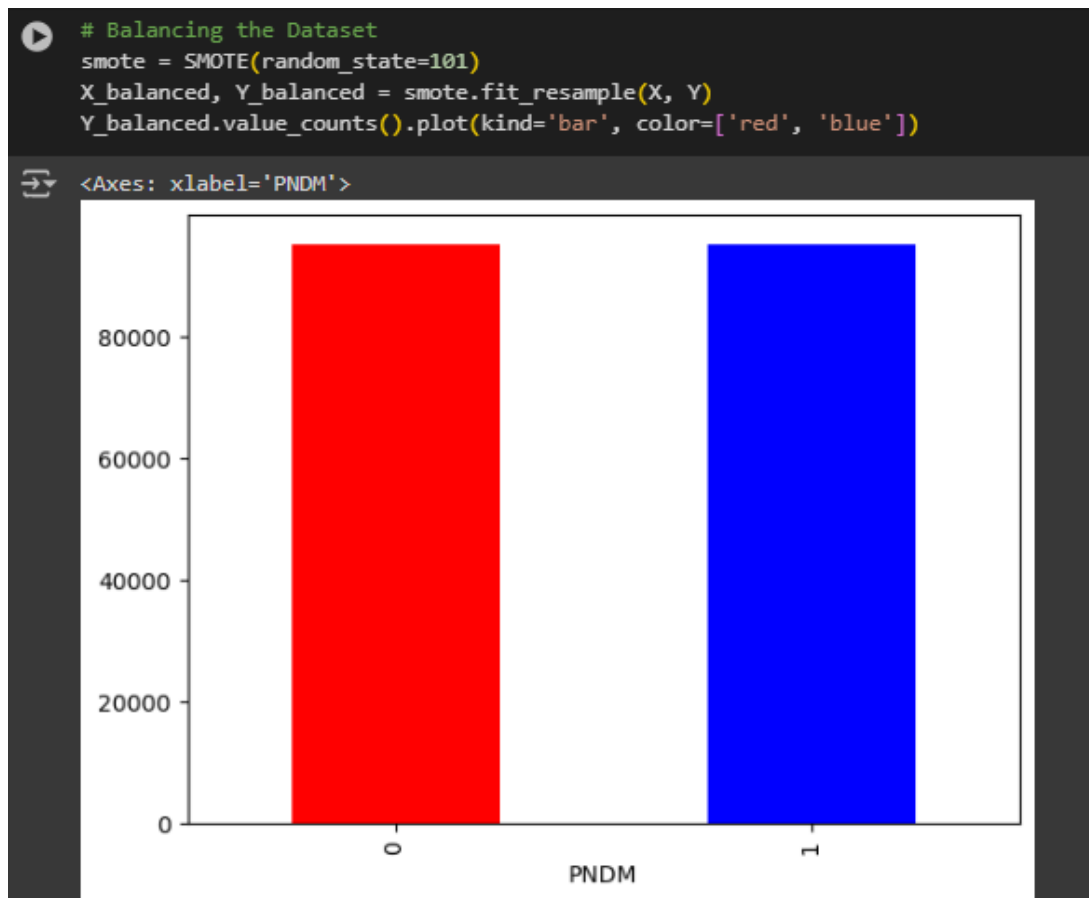


Figure 23 : Equilibrer les données.

3.3 Construction du modèle d'apprentissage profond

Le processus de construction du modèle d'apprentissage profond pour la prédiction de la maladie PNDM commence par l'initialisation avec la validation croisée K-fold (K-Fold cross validation) pour une évaluation robuste. Ensuite, les valeurs des caractéristiques sont standardisées pour garantir une comparaison équitable entre les caractéristiques. Enfin, le modèle est construit en configurant les couches du réseau neuronal, en compilant le modèle avec une fonction de perte appropriée et en entraînant le modèle sur les données d'entraînement. Voici les étapes en détails:

- Initialisation de K-Fold cross validation : Nous initialisons la validation croisée K-fold avec $k=10$ pour diviser nos données équilibrées en 10 plis (folds). Cela permettra d'évaluer la performance du modèle de manière robuste. Nous créons des listes pour stocker les scores d'évaluation et les métriques que nous allons calculer pour chaque fold lors de la validation croisée.

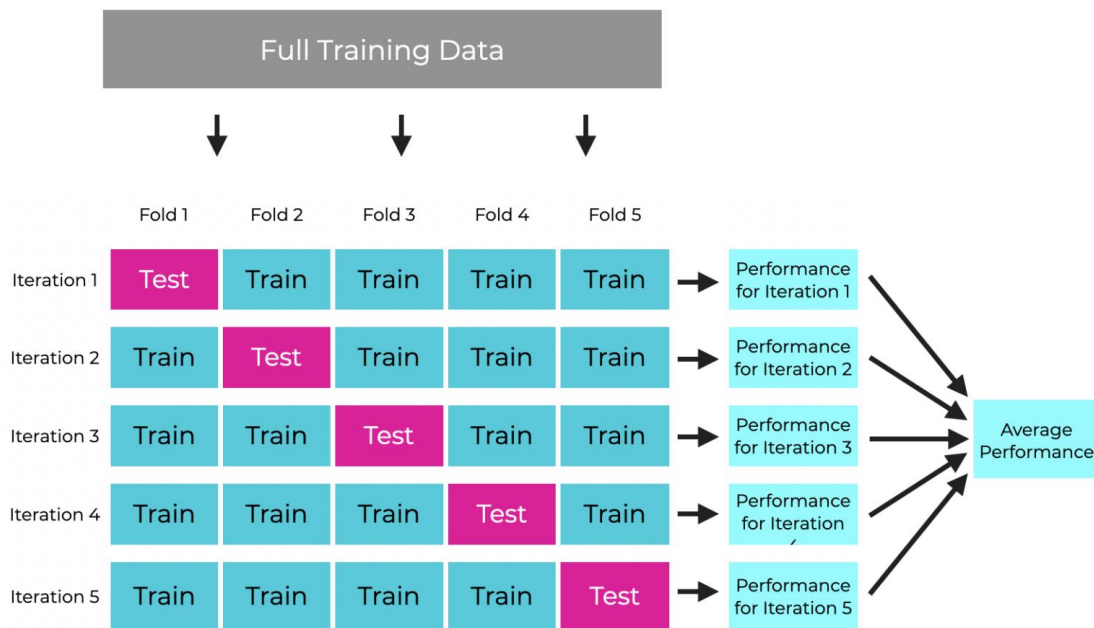


Figure 24 : K-fold validation croisée [105].

- Standardisation des valeurs des caractéristiques : Nous standardisons les caractéristiques en utilisant un StandardScaler, qui soustrait la moyenne et divise par l'écart type de chaque caractéristique. Cela permet de mettre toutes les caractéristiques à la même échelle, ce qui est important pour de nombreux algorithmes d'apprentissage automatique, notamment les réseaux de neurones. Nous appliquons d'abord la méthode `fit_transform()` sur les données d'entraînement pour calculer la moyenne et l'écart type de chaque caractéristique et standardiser les données. Ensuite, nous utilisons la méthode `transform()` sur les données de test pour les standardiser en utilisant les mêmes paramètres calculés à partir des données d'entraînement. Nous fixons la graine aléatoire pour TensorFlow à 3 en utilisant la fonction `set_seed()`. Cela garantit que les valeurs aléatoires générées par le réseau neuronal sont cohérentes à chaque exécution du

code. Fixer la graine aléatoire est important pour assurer la reproductibilité des résultats lors de l'entraînement du modèle (Figure 25).

```
[ ] # Initialize KFold cross-validation with k=10
kf = KFold(n_splits=10, shuffle=True, random_state=42)

# List to store the evaluation scores and metrics
accuracies = []
f1_scores = []
precisions = []
recalls = []
specificities = []
auc_rocs = []

[ ] # Iterate through each fold
for fold, (train_index, test_index) in enumerate(kf.split(X_balanced, Y_balanced)):
    X_train, X_test = X_balanced.iloc[train_index], X_balanced.iloc[test_index] # Split the data into training and testing sets
    Y_train, Y_test = Y_balanced.iloc[train_index], Y_balanced.iloc[test_index]

    # standardization of X data
    scaler = StandardScaler()
    X_train_std = scaler.fit_transform(X_train)
    X_test_std = scaler.transform(X_test)

    tf.random.set_seed(3) # generate random values of parameters every time the neural network runs
```

Figure 25 : Initialisation de K-Fold cross validation et Standardisation des valeurs.

- Conception du modèle
- Configuration des couches du réseau neuronal : Dans cette partie, nous configurons les couches du réseau neuronal. Nous utilisons une architecture avec plusieurs couches denses, chaque couche étant suivie d'une fonction d'activation 'relu' et d'une couche Dropout pour régulariser le modèle et éviter le surapprentissage. La dernière couche utilise une fonction d'activation 'sigmoid', car nous avons deux classes de sortie.
- Compilation du réseau neuronal : Nous compilons le modèle en spécifiant l'optimiseur ('adam'), la fonction de perte ('sparse_categorical_crossentropy') et les métriques que nous souhaitons suivre lors de l'entraînement ('accuracy').
- Entraînement du modèle : Enfin, nous entraînons le modèle sur les données d'entraînement normalisées pendant 40 époques (itérations d'entraînement). Nous

utilisons les données de validation par défaut fournies par Keras pour évaluer les performances du modèle à chaque époque. Après la construction du modèle, des prédictions sont effectuées en utilisant les données de test. Le modèle est utilisé pour prédire les probabilités des classes cibles à partir des caractéristiques de test, puis les prédictions finales sont obtenues en choisissant la classe avec la probabilité la plus élevée. Cela permet d'évaluer les performances du modèle sur des données invisibles et d'obtenir des informations sur sa capacité à généraliser à de nouvelles observations (Figure 26).

```
[ ]
# setting up the layers of Neural Network
model = keras.Sequential([
    keras.layers.Flatten(input_shape=(8,)),
    keras.layers.Dense(64, activation='relu'),
    keras.layers.Dropout(0.5),
    keras.layers.BatchNormalization(),
    keras.layers.Dense(32, activation='relu'),
    keras.layers.Dropout(0.5),
    keras.layers.BatchNormalization(),
    keras.layers.Dense(16, activation='relu'),
    keras.layers.Dropout(0.5),
    keras.layers.BatchNormalization(),
    keras.layers.Dense(2, activation='sigmoid')
])

# compiling the Neural Network
model.compile(optimizer='adam',
              loss='sparse_categorical_crossentropy',
              metrics=['accuracy'])

# Training the model
model.fit(X_train_std, Y_train, epochs=40, verbose=0)

# Predictions
Y_pred_proba = model.predict(X_test_std)
Y_pred = np.argmax(Y_pred_proba, axis=1)
```

Figure 26 : Conception du modèle.

3.4 Les mesures d'évaluation du modèle

Pour évaluer la performance du modèle, plusieurs métriques sont calculées pour chaque pli (fold) de la validation croisée. Ces métriques comprennent l'exactitude, le score F1, la précision, le rappel, la spécificité et l'AUC-ROC. Ensuite, les métriques

moyennes sont calculées à partir des résultats de chaque pli (fold) pour fournir une évaluation globale de la performance du modèle. En plus des métriques, la matrice de confusion ainsi que la courbe ROC et de précision-rappel sont tracées pour visualiser les performances du modèle et aider à interpréter ses prédictions.

- Les métriques d'évaluation

- Exactitude (Accuracy) : La proportion des résultats véritables (les résultats véritablement positifs et les résultats négatifs) parmi le nombre total de cas examinés. Il mesure la fréquence avec laquelle le classificateur fait la bonne prédiction [106].

$$\text{Exactitude} = \frac{VP + VN}{VP + VN + FP + FN}$$

Où (VP) est le nombre de vrais positifs, (FN) est le nombre de faux négatifs, (FP) est le nombre de faux positifs et (VN) est le nombre de vrais négatifs.

- Précision (Precision) : Le rapport entre les véritables positifs et la somme des vrais positifs et des faux positifs. Il indique la qualité des prédictions positives faites par le classificateur [106].

$$\text{Précision} = \frac{VP}{VP + FP}$$

- Rappel (Recall) : également connu sous le nom de sensibilité ou taux vrai positif, le rapport entre les vrais positifs et la somme des vrais positifs et des faux négatifs. Il mesure la capacité du classificateur à trouver tous les échantillons positifs [106].

$$\text{Rappel} = \frac{VP}{VP + FN}$$

- F1 Score : La moyenne harmonieuse de la précision et du rappel. Il fournit un équilibre entre la précision et le rappel dans les cas où l'un peut être plus important que l'autre [106].

$$F1\ score = 2 \times \frac{Rappel \times Précision}{Rappel + Précision}$$

- Spécificité (Specificity) : également connue sous le nom de True Negative Rate, le rapport entre les véritables négatifs et la somme des vrais négatifs et des faux positifs. Il mesure la capacité du classificateur à identifier correctement les échantillons négatifs [107].

$$Spécificité = \frac{VN}{VN + FP}$$

- AUC-ROC (Area Under the Receiver Operating Characteristic Curve) : zone sous la courbe des caractéristiques opérationnelles du récepteur. Il s'agit d'une mesure du rendement des problèmes de classification à différents seuils. ROC est une courbe de probabilité, et AUC représente le degré ou la mesure de séparabilité, indiquant à quel point le modèle est capable de distinguer entre les classes [107].

- Calcul des métriques

Nous ajoutons les valeurs calculées de chaque métrique à des listes pour les stocker et les utiliser ultérieurement (Figure 27).

```
# Evaluation metrics
accuracy = accuracy_score(Y_test, Y_pred)
f1 = f1_score(Y_test, Y_pred)
precision = precision_score(Y_test, Y_pred)
recall = recall_score(Y_test, Y_pred)
tn, fp, fn, tp = confusion_matrix(Y_test, Y_pred).ravel()
specificity = tn / (tn + fp)
auc_roc = roc_auc_score(Y_test, Y_pred_proba[:,1])

# Append metrics to lists
accuracies.append(accuracy)
f1_scores.append(f1)
precisions.append(precision)
recalls.append(recall)
specificities.append(specificity)
auc_rocs.append(auc_roc)

# Print metrics for the current fold
print(f"Fold {fold + 1} - Accuracy: {accuracy}, F1 Score: {f1}, Precision: {precision}, Recall: {recall}, Specificity: {specificity}, AUC-ROC: {auc_roc}")
```

Figure 27 : Calcul des métriques.

Ensuite, nous imprimons les valeurs de chaque métrique pour le pli (fold) actuel de la validation croisée, afin de surveiller les performances du modèle à chaque itération (Figure 28).

```
595/595 [=====] - 1s 2ms/step
Fold 1 - Accuracy: 0.897726413111998, F1 Score: 0.9064333108952783, Precision: 0.8301188903566711, Recall: 0.9981997246637722, Specificity: 0.7989158761596998, AUC-ROC: 0.901153689
595/595 [=====] - 1s 2ms/step
Fold 2 - Accuracy: 0.8961966799747846, F1 Score: 0.906554431098008, Precision: 0.8295110341843358, Recall: 0.9993744135126681, Specificity: 0.79142403388036, AUC-ROC: 0.8998077593951
595/595 [=====] - 1s 2ms/step
Fold 3 - Accuracy: 0.897825173355747, F1 Score: 0.9076053394138047, Precision: 0.8318530128874957, Recall: 0.998536636354134, Specificity: 0.7960713908543668, AUC-ROC: 0.90150345971
595/595 [=====] - 1s 2ms/step
Fold 4 - Accuracy: 0.8988758142466905, F1 Score: 0.9085119528539517, Precision: 0.8332316275825996, Recall: 0.9987460815047022, Specificity: 0.797908303401648, AUC-ROC: 0.9071457966
595/595 [=====] - 1s 2ms/step
Fold 5 - Accuracy: 0.8978897247320866, F1 Score: 0.906092708882604, Precision: 0.8304920913884007, Recall: 0.9968357768167915, Specificity: 0.798116169544741, AUC-ROC: 0.90182443158
595/595 [=====] - 1s 2ms/step
Fold 6 - Accuracy: 0.8952511031729354, F1 Score: 0.904447000191681, Precision: 0.8262125722290317, Recall: 0.9990472157526995, Specificity: 0.7930135557872784, AUC-ROC: 0.9014178116
595/595 [=====] - 1s 2ms/step
Fold 7 - Accuracy: 0.901076963488311, F1 Score: 0.9105930392668914, Precision: 0.8358612273361228, Recall: 1.0, Specificity: 0.8006563624814736, AUC-ROC: 0.9077643663571299
595/595 [=====] - 1s 2ms/step
Fold 8 - Accuracy: 0.8997635933806146, F1 Score: 0.9083133109082172, Precision: 0.8329073763990482, Recall: 0.9987319032019444, Specificity: 0.8019222732971166, AUC-ROC: 0.909592218
595/595 [=====] - 1s 2ms/step
Fold 9 - Accuracy: 0.8975571315996048, F1 Score: 0.9070366132723111, Precision: 0.8311200419360475, Recall: 0.9982161594963274, Specificity: 0.7966333508679643, AUC-ROC: 0.894148498
595/595 [=====] - 1s 2ms/step
Fold 10 - Accuracy: 0.8982926188599948, F1 Score: 0.9075011944577162, Precision: 0.8307382785164451, Recall: 0.999894714676774, Specificity: 0.7971060081786725, AUC-ROC: 0.907115051

Training time: 1h
```

Figure 28 : Les valeurs de chaque métrique pour le pli (fold).

Enfin, nous calculons les valeurs moyennes de chaque métrique sur tous les plis (folds) de la validation croisée pour obtenir une évaluation globale de la performance du modèle. Nous imprimons les valeurs moyennes de chaque métrique pour évaluer la performance globale du modèle (Figure 29).

```
[ ] # Calculate average metrics
    avg_accuracy = np.mean(accuracies)
    avg_f1 = np.mean(f1_scores)
    avg_precision = np.mean(precisions)
    avg_recall = np.mean(recalls)
    avg_specificity = np.mean(specificities)
    avg_auc_roc = np.mean(auc_rocs)

# Print average metrics
print("Average Accuracy:", avg_accuracy)
print("Average F1 Score:", avg_f1)
print("Average Precision:", avg_precision)
print("Average Recall:", avg_recall)
print("Average Specificity:", avg_specificity)
print("Average AUC-ROC Score:", avg_auc_roc)
```

Figure 29 : Calcule des moyennes de chaque métrique.

- Tracé la matrice et les courbes
- Matrice de confusion : La matrice de confusion est tracée pour visualiser les performances du modèle. Elle montre le nombre de prédictions correctes et incorrectes pour chaque classe. Les prédictions correctes sont représentées sur la diagonale de la matrice. Les valeurs hors diagonale indiquent les prédictions incorrectes (Figure 30).

```

▶ # Plot Confusion Matrix
plt.figure()
conf_matrix = confusion_matrix(Y_test, Y_pred)
plt.imshow(conf_matrix, interpolation='nearest', cmap=plt.cm.Blues)
for i in range(conf_matrix.shape[0]):
    for j in range(conf_matrix.shape[1]):
        plt.text(j, i, format(conf_matrix[i, j], 'd'),
                 horizontalalignment="center",
                 color="white" if conf_matrix[i, j] > conf_matrix.max() / 2 else "black")
plt.title("Confusion Matrix")
plt.colorbar()
plt.xticks([0, 1], ['Predicted Negative', 'Predicted Positive'])
plt.yticks([0, 1], ['Actual Negative', 'Actual Positive'])
plt.xlabel('Predicted labels')
plt.ylabel('True labels')
plt.show()

```

Figure 30 : Tracé la matrice de confusion.

- Courbe ROC : La courbe ROC (Receiver Operating Characteristic) est tracée pour évaluer la capacité du modèle à discriminer entre les classes positives et négatives. Elle représente le taux de vrais positifs (TPR) en fonction du taux de faux positifs (FPR) pour différents seuils de classification (Figure 31).

```

▶ # Plot ROC Curve
plt.figure()
fpr, tpr, thresholds = roc_curve(Y_test, Y_pred_proba[:, 1])
plt.plot(fpr, tpr, label='ROC Curve (AUC = %0.2f)' % avg_auc_roc)
plt.plot([0, 1], [0, 1], 'k--')
plt.xlim([0.0, 1.0])
plt.ylim([0.0, 1.05])
plt.xlabel('False Positive Rate')
plt.ylabel('True Positive Rate')
plt.title('Receiver Operating Characteristic')
plt.legend(loc="lower right")
plt.show()

```

Figure 31 : Tracé la Courbe ROC.

- Courbe de précision-rappel : La courbe de précision-rappel est tracée pour évaluer la précision du modèle à différentes valeurs de rappel. Elle représente la précision en fonction du rappel pour différents seuils de classification. L'AUC de cette courbe est également calculée pour fournir une mesure globale de la performance du modèle dans la prédiction des classes positives (Figure 32).

```
# Plot Precision-Recall Curve
plt.figure()
precision, recall, _ = precision_recall_curve(Y_test, Y_pred_proba[:, 1])
plt.plot(recall, precision, label='Precision-Recall curve (AUC = %0.2f)' % avg_auc_roc)
plt.xlabel('Recall')
plt.ylabel('Precision')
plt.ylim([0.0, 1.05])
plt.xlim([0.0, 1.0])
plt.title('Precision-Recall Curve')
plt.legend(loc="lower left")
plt.show()
```

Figure 32 : Tracé la Courbe de précision-rappel.

3.5 Sauvegarde du modèle

Dans cette dernière étape, nous enregistrons le modèle et le scaler de standardisation pour une utilisation future. Nous enregistrons le scaler de standardisation utilisé pour mettre à l'échelle les caractéristiques dans un fichier. Cela nous permet de conserver les mêmes paramètres de standardisation lors de la préparation des données pour de nouvelles prédictions. Ensuite, nous enregistrons le modèle entraîné dans un fichier au format HDF5. Cela permet de sauvegarder l'architecture du modèle ainsi que ses poids et ses configurations d'entraînement. Finalement, nous déplaçons le fichier contenant le modèle sauvegardé vers un répertoire Drive. Cela garantit que le modèle est stocké dans un emplacement sécurisé et facilement accessible pour une utilisation ultérieure (Figure 33).


```
[ ] # saving the scaler of standardization
scaler_save_path = '/content/drive/MyDrive/PFE ML DL Models/scaler_v3.pkl'
joblib.dump(scaler, scaler_save_path)

[ ] # saving the model
model.save('PNDM_disease_DL_prediction_v3.h5')
shutil.move('PNDM_disease_DL_prediction_v3.h5', '/content/drive/MyDrive/')
```

Figure 33 : Sauvegarde du modèle.

4- Développement de la Plateforme

4.1 Création de l'environnement virtuel et installation de Django

Au début du développement de la plateforme, nous avons créé un environnement virtuel nommé (env) par la commande « py -m venv env » pour assurer la gestion des dépendances dans un environnement propre et isolé. Cela jouait un rôle essentiel dans la préservation de la cohérence à travers les différentes étapes du développement et dans l'évitement de conflits avec d'autres projets. Puis, on a activé l'environnement par la commande « env\Scripts\activate.bat ». Ensuite, Django a été installé dans cet environnement par la commande « py -m pip install Django ».

4.2 Création du projet et des applications

Dans cette partie, nous avons créé la structure du projet comprenant un répertoire principal nommé (proj) et plusieurs applications Django, chacune pour un objectif distinct. Premièrement, l'application (base) qui gère les pages d'accueil et d'autres pages de services généraux. Ensuite, l'application (PNDMapp) pour gérer le modèle de prédiction et l'affichage des résultats. Puis, l'application (register) qui gère les processus d'enregistrement et d'authentification des utilisateurs, pour un contrôle d'accès sécurisé aux fonctionnalités de la plateforme.

4.3 Intégration de HTML, CSS et JavaScript

Notre plateforme contient plusieurs pages HTML pour différentes fonctionnalités et interactions (Figure 34). Et nous avons lié ces pages à des feuilles de style externes CSS pour garantir une interface utilisateur cohérente et facile à utiliser. Aussi, des fichiers JavaScript ont été intégrés pour améliorer l'interactivité, la validation des formulaires, les requêtes et améliorant ainsi l'expérience utilisateur globale.

```
{% load static %}
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no" />
    <meta name="description" content="" />
    <meta name="author" content="" />
    <title> Rare Diseases Platform </title>
    <!-- Favicon-->
    <link rel="icon" type="image/x-icon" href="{% static 'images/favicon.ico' %}" />
    <!-- Font Awesome icons (free version)-->
    <script src="{% static 'js/fontawsome.js' %}" crossorigin="anonymous"></script>
    <!-- Google fonts-->
    <link href="https://fonts.googleapis.com/css?family=Montserrat:400,700" rel="stylesheet"
    <link href="https://fonts.googleapis.com/css?family=Roboto+Slab:400,100,300,700" rel="sty
    <!-- Core theme CSS (includes Bootstrap)-->
    <link href="{% static 'css/styles.css' %}" rel="stylesheet" />
  </head>
  <body id="page-top">
    <!-- Navigation-->
    <nav class="navbar navbar-expand-lg navbar-dark fixed-top" id="mainNav">
      <div class="container">
        <a class="navbar-brand" href="/">RH</a>
```

Figure 34 : Structure HTML.

4.4 Configuration et routage

Dans cette partie, nous commençons la configuration de la plateforme. Tout d'abord, la configuration était gérée via le fichier settings.py, où divers paramètres tels que les applications installées, les configurations de base de données et la gestion des fichiers statiques étaient définis (Figure 35).

```

from pathlib import Path

# Build paths inside the project like this: BASE_DIR / 'subdir'.
BASE_DIR = Path(__file__).resolve().parent.parent

SECRET_KEY = 'django-insecure-)bmuav!#wiq(y54x5n7$j!j$vmg@+h_krlr_+mu2_bi-ok#p_=*'

# SECURITY WARNING: don't run with debug turned on in production!
DEBUG = True
ALLOWED_HOSTS = []

# Application definition

INSTALLED_APPS = [
    'django.contrib.admin',
    'django.contrib.auth',
    'django.contrib.contenttypes',
    'django.contrib.sessions',
    'django.contrib.messages',
    'django.contrib.staticfiles',
    'base.apps.BaseConfig',
    'register.apps.RegisterConfig',
    'PNDMapp.apps.PndmappConfig',
]

```

Figure 35 : Le fichier settings.py

Ensuite, le routage des pages web était géré à travers les fichiers urls.py de projet et les applications (Figure 36), pour définir les URL qui mappent les vues correspondantes des fichiers views.py, qui contiennent la logique pour rendre les pages, traiter les entrées des utilisateurs et interagir avec la base de données, assurant un flux de données entre l'interface utilisateur et le backend (Figure 37).

```

from django.contrib import admin
from django.urls import path, include
from register import views as v
from django.contrib.auth import views as auth_views

urlpatterns = [
    path('', include('PNDMapp.urls')),
    path('admin/', admin.site.urls),
    path('', include('base.urls')),
    path('register/', v.register, name="register"),
    path('', include('django.contrib.auth.urls')),
    path('', include('register.urls')),
    path('login/', auth_views.LoginView.as_view(), name='login'),
]

```

Figure 36 : Le fichier urls.py

```
import os
os.environ["CUDA_VISIBLE_DEVICES"] = "-1"
✓ from django.http import HttpResponse
  from django.shortcuts import render, redirect
  from django.contrib.auth import authenticate, login, logout
  from .models import *
  from django.contrib import messages
  from django.contrib.auth.decorators import login_required
  # Create your views here.

  @login_required(login_url='login')
  ✓ def home(request):
    |     return render(request, "index.html", {})

    @login_required(login_url='login')
  ✓ def diseases_view(request):
    |     return render(request, 'diseases.html')

    @login_required(login_url='login')
  ✓ def treatment_view(request):
    |     return render(request, 'treatment.html')
```

Figure 37 : Le fichier views.py

4.5 Gestion des utilisateurs et authentification

Notre plateforme supportait plusieurs utilisateurs avec différents rôles et permissions. Et pour cela, nous avons géré l'authentification des utilisateurs via l'application register, assurant des processus de connexion et d'enregistrement sécurisés. D'abord, les patients avaient accès à des fonctionnalités spécifiques telles que réaliser leurs propres diagnostics, voir la liste de médecins, prendre des rendez-vous, et interagir avec d'autres patients et médecins. Ensuite, les médecins peuvent également effectuer des diagnostics pour leurs patients, gérer des rendez-vous et répondre aux questions des patients. Mais les patients n'ont pas accès aux pages des médecins, et vice versa. Cette différenciation des rôles et des autorisations des utilisateurs était cruciale pour assurer la sécurité et l'intégrité de la plate-forme (Figure 38).

```

def register(response):
    if response.user.is_authenticated:
        return redirect("/")
    else:
        if response.method == "POST":
            form = RegisterForm(response.POST)
            if form.is_valid():
                form.save()
                return redirect("/")
            else:
                form = RegisterForm()
                return render(response, "register/register.html", {"form":form})
def loginpage(request):
    if request.user.is_authenticated:
        return redirect("/")
    else:
        if request.method == 'POST':
            username = request.POST.get('username')
            password = request.POST.get('password')
            user = authenticate(request, username=username, password=password)
            if user is not None:
                login(request, user)
                return redirect('/')
            else:
                messages.info(request, 'Username OR password is incorrect')

```

Figure 38 : L'authentification des utilisateurs.

4.6 L'intégration du modèle intelligent dans la plateforme

Cette partie du code permet d'intégrer le modèle intelligent dans la plateforme.

```

import os
os.environ["CUDA_VISIBLE_DEVICES"] = "-1"
from django.shortcuts import render
from django.contrib.auth.decorators import login_required
from tensorflow import keras
from keras.models import load_model
import joblib
model_path = r'C:\Users\HP 2018\Desktop\New folder (2)\hb\proj\savedModels\PNDM_disease_DL_prediction_v2.h5'
model = load_model(model_path)
# Create your views here.
@login_required(login_url='login')
def predictor(request):
    return render(request, 'index.html')
@login_required(login_url='login')

def formInfo(request):
    age = request.POST.get('age')
    HbA1c = request.POST.get('HbA1c')
    Birth_Weight = request.POST.get('Birth_Weight')
    Insulin_level = request.POST.get('Insulin_level')
    Family_History = request.POST.get('Family_History')
    Developmental_Delay = request.POST.get('Developmental_Delay')

```

Figure 39 : Intégration du modèle de prédiction dans le fichier views.

```

# Convert Yes/No to 0/1
if Family_History == 'Yes':
    Family_History_No = '0'
    Family_History_Yes = '1'
else:
    Family_History_No = '1'
    Family_History_Yes = '0'

if Developmental_Delay == 'Yes':
    Developmental_Delay_No = '0'
    Developmental_Delay_Yes = '1'
else:
    Developmental_Delay_No = '1'
    Developmental_Delay_Yes = '0'

# the prediction
input_data = [[float(age), float(HbA1c), float(Birth_Weight), float(Insulin_level),
               int(Family_History_No), int(Family_History_Yes), int(Developmental_Delay_No),
               int(Developmental_Delay_Yes)]]
print(input_data)
# change the input_data to a numpy array
import numpy as np
input_data_np = np.array([input_data])
print(input_data_np)

```

Figure 40 : Intégration du modèle de prédiction dans le fichier views.py.

```

# Load the scaler
scaler_path = r'C:\Users\HP 2018\Desktop\New folder (2)\hb\proj\savedModels\scaler.pkl'
scaler = joblib.load(scaler_path)
scaler.feature_names_in_ = None
input_data_std = scaler.transform(input_data_np)
print(input_data_std)

y_predct = model.predict(input_data_std)
print(y_predct)

y_pred_label = [np.argmax(y_predct)]
print(y_pred_label)

if (y_pred_label[0] == 0):
    y_pred = 'do not have PNDM'
else:
    y_pred = 'have PNDM'

return render(request, 'result.html', {'result' : y_pred})

```

Figure 41 : Intégration du modèle de prédiction dans le fichier views.py.

4.7 Conception et gestion de la base de données

Dans cette partie, une base de données complète avait été conçue pour stocker et gérer les données de manière efficace à travers le fichier `models.py` (Figure 42). Les principales tables incluaient : Premièrement, la table "Patient" stockait les informations et les détails personnels sur les patients. Ensuite, la table "Doctor" contient des informations sur les médecins inscrits sur la plateforme. Puis, la table "Prediction" stockait les résultats du modèle de prédiction. Enfin, la table "Appointment" traite les détails des rendez-vous entre patients et médecins. Cette approche structurée assurait la cohérence et l'intégrité des données à travers la plateforme.

```
from django.db import models
from django.contrib.auth.models import User
# Create your models here.
class Patient(models.Model):
    GENDER = (
        ('Male', 'Male'),
        ('Female', 'Female'),
    )
    first_name = models.CharField(max_length=50, null=True)
    last_name = models.CharField(max_length=50, null=True)
    gender = models.CharField(max_length=200, null=True, choices=GENDER)
    phone = models.CharField(max_length=20, null=True)
    email = models.CharField(max_length=200, null=True)
    country = models.CharField(max_length=50, null=True)
    city = models.CharField(max_length=50, null=True)
    address = models.CharField(max_length=200, null=True)
    date_created = models.DateTimeField(auto_now_add=True, null=True)
    def __str__(self):
        return self.first_name
class Doctor(models.Model):
    SPECIALTY = (
        ('General doctor', 'General doctor'),
        ('Medical Geneticist', 'Medical Geneticist'),
        ('Rheumatologist', 'Rheumatologist'),
        ('Neurologist', 'Neurologist'),
```

Figure 42 : Le fichier `models.py`

En utilisant PostgreSQL, nous avons créé le schéma de base de données : c'est la structure et l'organisation des informations liées à la collecte et à l'analyse des données des patients (Figures 43 et 44). Il joue un rôle essentiel dans la gestion, le

stockage et la manipulation des données afin de garantir une utilisation facile, efficace et sécurisée.

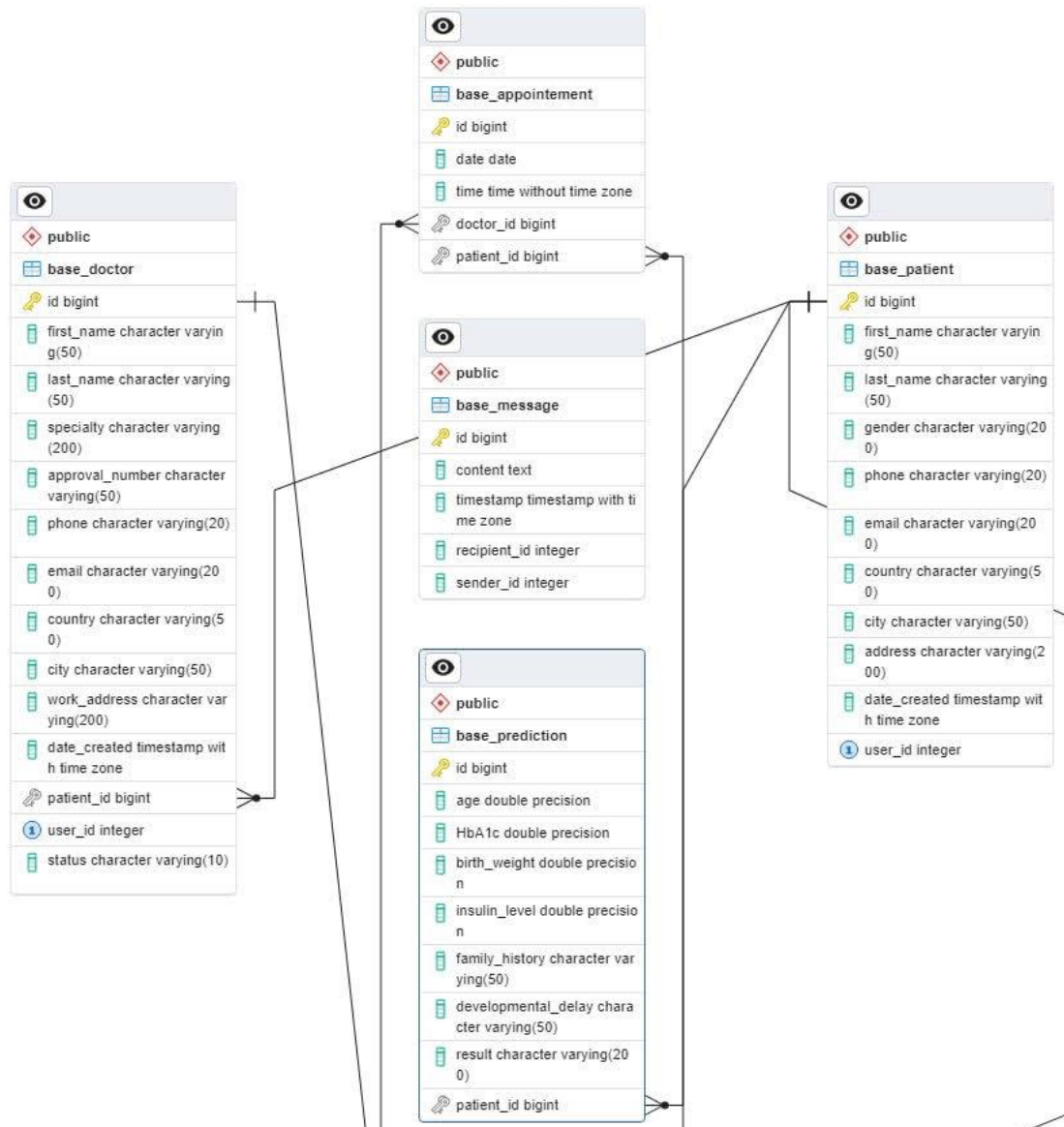


Figure 43 : Relation entre les tables de fichier models.py de la base de données

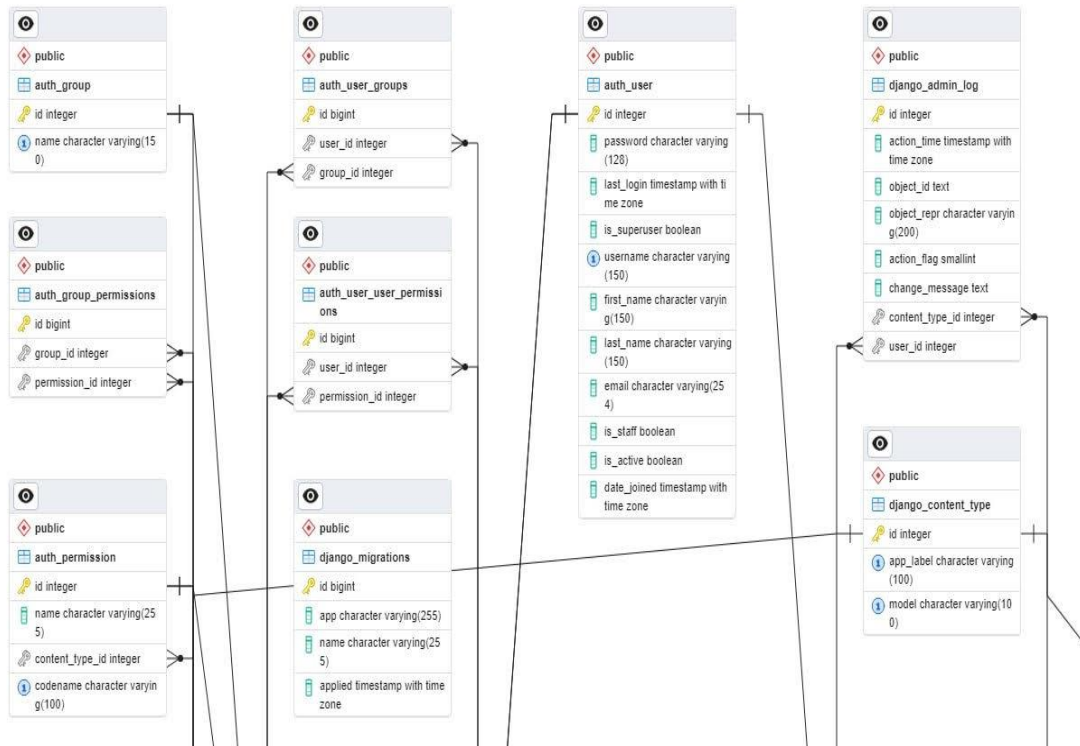


Figure 44 : Relation entre les tables de l'authentification de la base de données

4.8 Hébergement et déploiement

L'étape finale impliquait de placer la plateforme sur un prestataire de services fiable afin de garantir son accessibilité et ses performances. Et pour cela, nous avons opté pour la plateforme cloud Render. Dans le déploiement, il a été nécessaire de configurer l'environnement serveur et de mettre en place la base de données. La plateforme est disponible à travers le lien suivant :

<https://rarehealth-platform.onrender.com>

5- Résultats et discussion

Les résultats de ce travail sont divisés en deux parties : la première partie consiste à la création du modèle DL pour la prédiction de PNDM en utilisant Multi-Layer Perceptron.

La seconde partie consiste à la création de la plateforme en utilisant Django Framework.

5.1 Résultats d'apprentissage

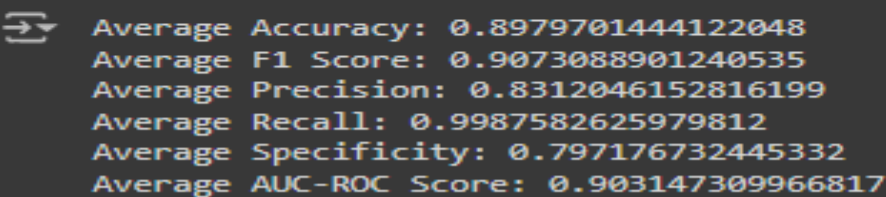
À la fin de l'apprentissage, une évaluation du modèle a été réalisée après l'application de diverses techniques d'équilibrage, des ajustements des couches et des époques d'apprentissage, ainsi que des modifications des fonctions d'activation telles que ReLU et Sigmoid, le tout dans le but d'optimiser les performances du modèle. Voici les résultats obtenus après 10 plis de validation croisée (K-Fold cross validation) :

Tableau 5 : Les valeurs de chaque métrique pour les plis.

Plis	Exactitude	Score F1	Précision	Rappel	Spécificité	AUC-ROC
1	0.8977	0.9064	0.8301	0.9981	0.7989	0.9011
2	0.8961	0.9065	0.8295	0.9993	0.7914	0.8998
3	0.8978	0.9076	0.8318	0.9985	0.7960	0.9015
4	0.8988	0.9085	0.8332	0.9987	0.7979	0.9071
5	0.8970	0.9060	0.8304	0.9968	0.7981	0.9018
6	0.8952	0.9044	0.8262	0.9990	0.7930	0.9014
7	0.9010	0.9105	0.8358	1	0.8006	0.9077
8	0.8997	0.9083	0.8329	0.9987	0.8019	0.9095
9	0.8975	0.9010	0.8311	0.9982	0.7966	0.8941
10	0.8982	0.9075	0.8307	0.9998	0.7971	0.9071

Les valeurs moyennes de chaque métrique sur tous les plis (folds) de la validation croisée pour obtenir une évaluation globale de la performance du modèle (Figure 45) :

```
# Print average metrics
print("Average Accuracy:", avg_accuracy)
print("Average F1 Score:", avg_f1)
print("Average Precision:", avg_precision)
print("Average Recall:", avg_recall)
print("Average Specificity:", avg_specificity)
print("Average AUC-ROC Score:", avg_auc_roc)
```



```
⇒ Average Accuracy: 0.8979701444122048
Average F1 Score: 0.9073088901240535
Average Precision: 0.8312046152816199
Average Recall: 0.9987582625979812
Average Specificity: 0.797176732445332
Average AUC-ROC Score: 0.903147309966817
```

Figure 45 : Les valeurs moyennes de chaque métrique.

- L'exactitude moyenne de 89,79 % indique que le modèle a correctement classé environ 89,79 % des échantillons.
- Le score F1 moyen de 90,73 % montre que le modèle maintient un excellent équilibre entre la précision et le rappel.
- Avec une précision moyenne de 83,12 %, le modèle démontre une bonne qualité dans les prédictions positives.
- Le rappel moyen de 99,87 % reflète l'efficacité du modèle à identifier les cas positifs.
- La spécificité moyenne de 79,71 % révèle une capacité limitée du modèle à reconnaître correctement les échantillons négatifs.
- L'AUC-ROC moyen de 90,31 % indique que le modèle possède une très bonne capacité de discrimination entre les classes.

En somme, les résultats montrent que le modèle a une performance globale solide avec des valeurs élevées pour l'exactitude, le score F1, le rappel et l'AUC-ROC, bien que la précision et la spécificité soient légèrement plus faibles.

Par la suite, nous avons visualisé la matrice de confusion (Figure 46), la courbe ROC (Figure 47) et la courbe de précision-rappel (Figure 48).

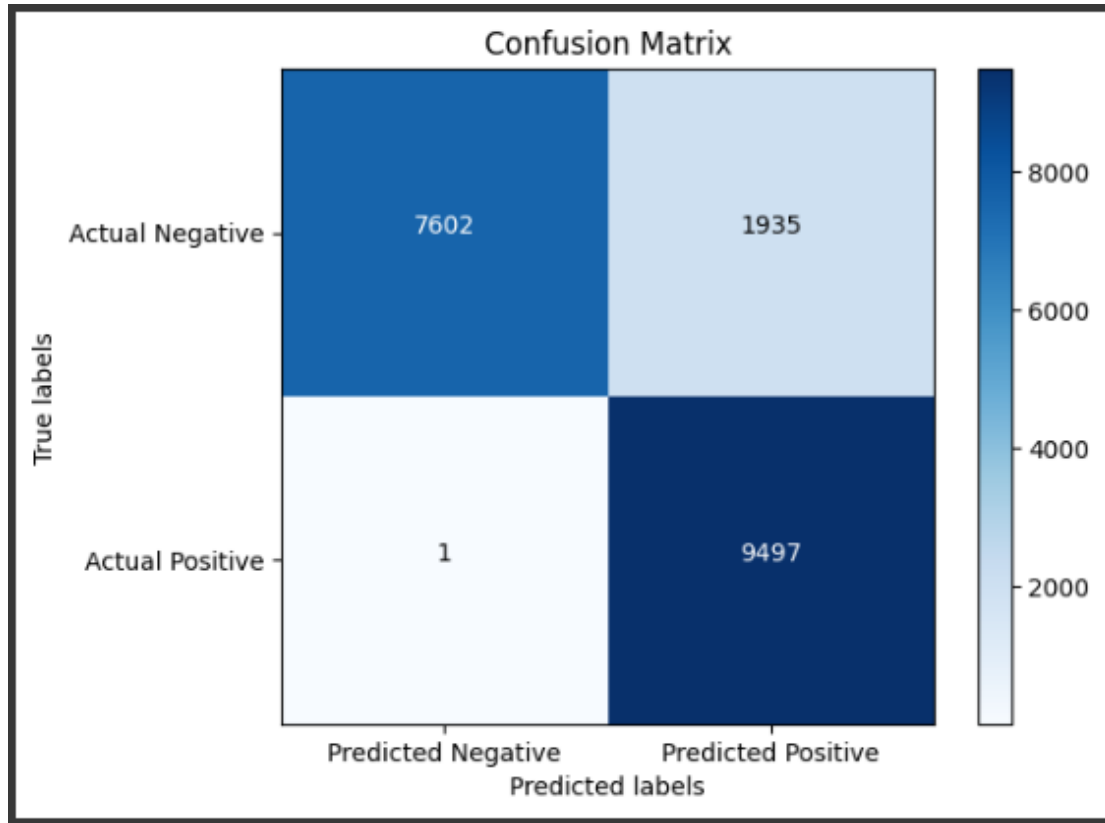


Figure 46 : La matrice de confusion.

La matrice de confusion est un outil essentiel pour évaluer les performances d'un modèle de classification. Elle se compose de quatre éléments principaux :

- Vrais négatifs (VN) : 7602 prédictions correctes où le modèle a prédit la classe négative et c'était effectivement le cas.
- Faux positifs (FP) : 1935 prédictions incorrectes où le modèle a prédit la classe positive alors que c'était en réalité négatif.
- Faux négatifs (FN) : 1 prédiction incorrecte où le modèle a prédit la classe négative alors que c'était positive.
- Vrais positifs (VP) : 9497 prédictions correctes où le modèle a prédit la classe positive et c'était effectivement le cas.

Avec un seul FN, notre modèle montre une excellente sensibilité ou rappel, ce qui est crucial pour des maladies rares comme le PNDM où manquer un cas positif peut être très préjudiciable. Cependant, le nombre relativement élevé de FP suggère que le modèle pourrait être amélioré en termes de spécificité.

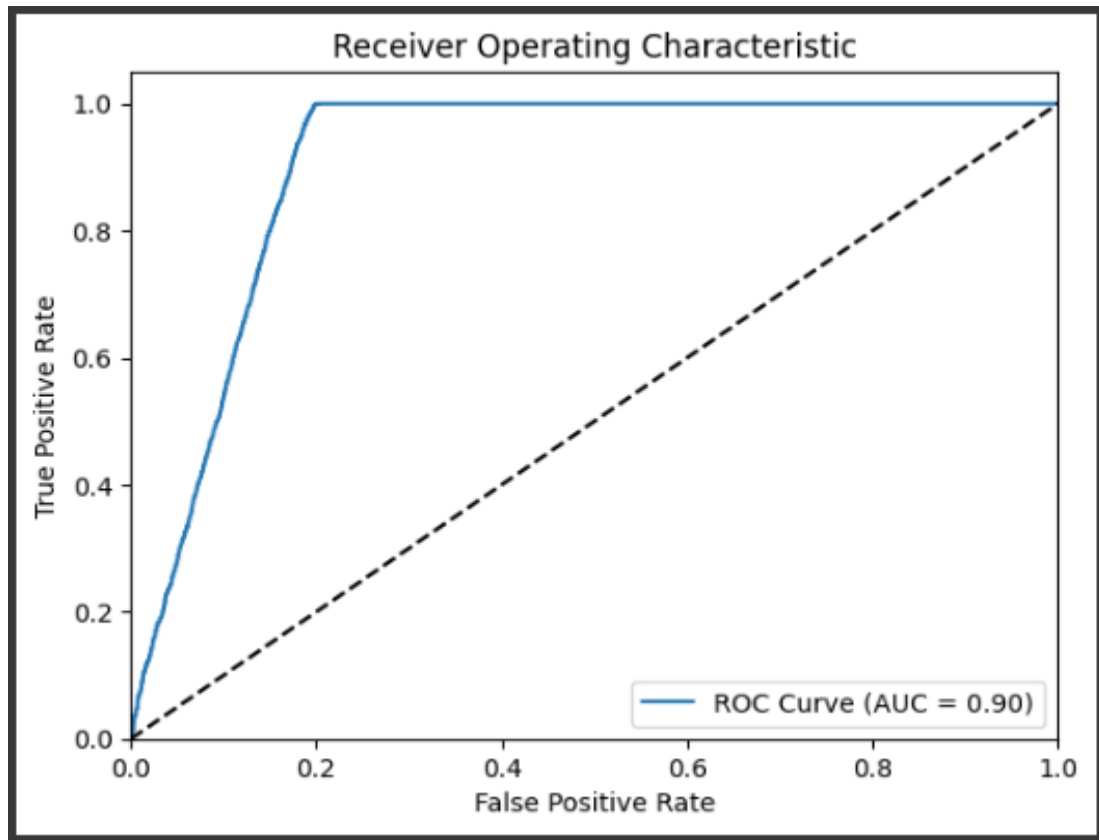


Figure 47 : La courbe ROC.

La courbe ROC est un graphique important pour évaluer les performances d'un modèle de classification binaire. Elle illustre la relation entre le taux de vrais positifs (TPR) et le taux de faux positifs (FPR) à différents seuils de décision :

- Taux de Vrais Positifs (TPR) : il représente la sensibilité du modèle, c'est-à-dire sa capacité à identifier correctement les cas positifs.
- Taux de Faux Positifs (FPR) : il indique la proportion de cas négatifs incorrectement identifiés comme positifs par le modèle.

La courbe ROC de notre modèle semble bien se comporter, avec une AUC (Area Under the Curve) de 0.90, ce qui est assez élevé. Une AUC de 1 représente un modèle parfait, tandis qu'une AUC de 0.5 indique une performance équivalente à une classification aléatoire. Donc notre modèle a une bonne capacité à distinguer entre les classes positives et négatives.

La ligne diagonale en pointillés représente la performance d'un classificateur aléatoire. La courbe ROC est bien au-dessus de cette ligne, ce qui indique que le modèle a une performance significativement meilleure que le hasard.

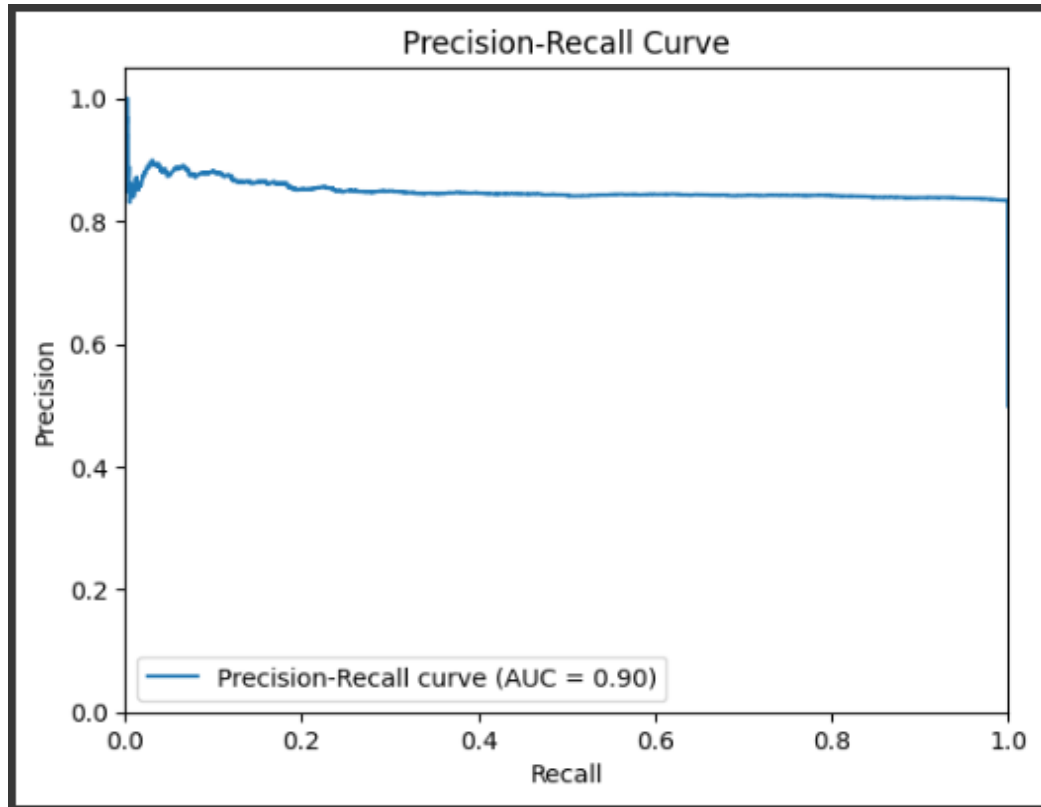


Figure 48 : La courbe de précision-rappel

La courbe précision-rappel est un outil utile pour évaluer les performances d'un modèle de classification, en particulier lorsque les classes sont déséquilibrées. Elle montre la relation entre la précision (la proportion de vrais positifs parmi les prédictions positives) et le rappel (la proportion de vrais positifs parmi les cas réels positifs).

Dans la courbe la précision commence très élevée, ce qui indique que le modèle a une forte probabilité de prédiction correcte lorsqu'il prédit une classe positive. Et elle maintient une précision élevée sur tous les niveaux de rappel, ce qui suggère que le modèle est capable de retrouver une grande partie des cas positifs sans sacrifier la précision.

Aussi, la courbe a une AUC (Area Under the Curve) de 0.90, ce qui est excellent. Donc notre modèle semble bien équilibré en termes de précision et de rappel.

En résumé, notre modèle semble performant pour la prédiction de la maladie PNDM, avec une bonne sensibilité (rappel) et une forte probabilité de prédiction correcte (précision). Cependant, il est toujours bon de comparer ces résultats avec d'autres modèles ou des études. En raison de la rareté des études utilisant des méthodes d'intelligence artificielle pour prédire la PNDM, notre comparaison a juxtaposé les résultats à des recherches axées sur le diabète mellitus, compte tenu de la relation inhérente entre la PNDM et le diabète mellitus. En comparant notre travail à l'étude de Henry Semakula et al. « A predictive model for early detection of diabetes mellitus using machine learning » [108], nous pouvons prendre en compte différents critères tels que les caractéristiques des données, la méthode utilisée et l'exactitude. Les résultats de la comparaison sont les suivants :

Tableau 6 : Comparaison du notre travail avec un autre.

	Les caractéristiques	Méthode	L'exactitude
Notre travail	Age	DL : MLP (Multi-Layer Perceptron)	89,79 %
	HbA1c		
	Niveau d'insuline		
	Poids de naissance		
	Historique familial		
	Retard du développement		
Le modèle de prédiction de	Age	ML : Random Forest Classifier Naïve Bayes Classifier	85.4 % 80.2 %
	Gender		
	Systolic blood pressure		
	Residence		
	Diastolic blood pressure		
	Body Mass Index.		
	Family Member with		

diabète mellitus	Diabetes,	Support Vector Machine Classifier	80.7 %
	Alcohol intake		
	Smoker		
	Hypertension		
	Obesity	AdaBoost Classifier	84.2 %
	Achieves WHO recommended minimum physical activity level		

5.2 Plateforme

Dans cette partie, nous présentons notre plateforme RareHealth, c'est une plateforme innovante conçue pour faciliter le diagnostic de diabète néonatal permanent (PNDM), et pour améliorer la communication entre les professionnels de la santé et les patients. La plateforme offre une variété de services incluant la création de comptes, le diagnostic de PNDM, et des espaces dédiés pour les médecins et les patients. Voici des captures d'écran des services du RareHealth :

5.2.1 Création de compte

Cette page permet aux nouveaux utilisateurs de s'inscrire sur la plateforme (Figure 49).

Create your Account

Username:

Required. 150 characters or fewer. Letters, digits and @/./+/-/_ only.

Email:

Password:

- Your password can't be too similar to your other personal information.
- Your password must contain at least 8 characters.
- Your password can't be a commonly used password.
- Your password can't be entirely numeric.

Password confirmation:

Enter the same password as before, for verification.

SIGN UP

Already have an account? [LOGIN](#)

Figure 49 : Page de création de compte.

5.2.2 Connexion

Ce page permet aux utilisateurs de se connecter à leur compte (Figure 50).

Welcome

Username:

Password:

Don't have an account? Please [SIGN UP](#)

LOG IN

Figure 50 : Page de connexion à des comptes déjà existants.

5.2.3 Page d'accueil

Ce page est la page d'accueil de notre plateforme RareHealth, où les utilisateurs trouvent des services selon leurs catégories (Figure 51).

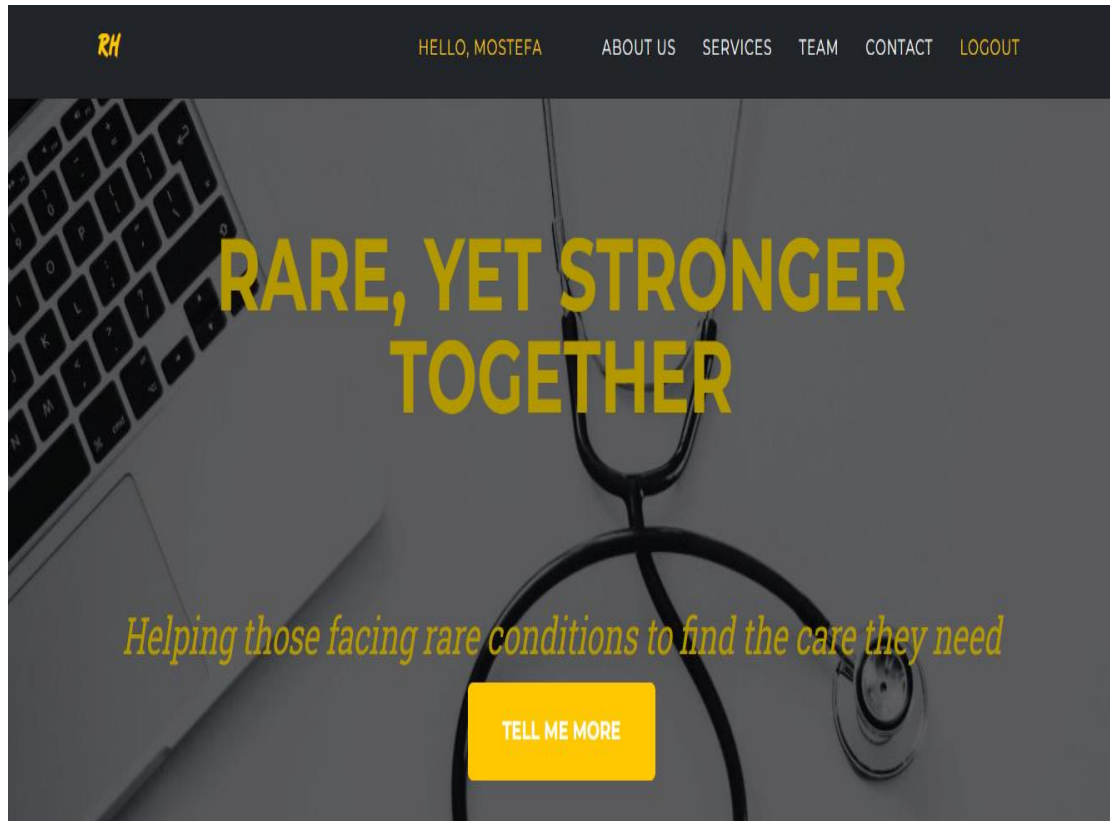


Figure 51 : Page d'accueil de la plateforme RareHealth

5.2.4 Interface de diagnostic

Ce page permet aux utilisateurs de soumettre des données cliniques dans un formulaire (Figure 52). Pour un diagnostic du PNDM par un modèle de classification intelligente qui fournit des résultats avec une précision de 89 % (Figure 53).

Age :

HbA1c :

Birth Weight :

Insulin level :

Family History :

Developmental Delay :

Diagnose

Figure 52 : Formulaire de prédiction.



Based on the symptoms you've provided :

The patient you're diagnosing have PNDM



Figure 53 : Page de résultat de prédiction.

Conclusion et Perspectives

Le travail a commencé par la collecte de données cliniques essentielles pour diagnostiquer la maladie rare PNDM (Diabète Néonatal Permanent) à l'aide de l'apprentissage profond par un modèle de classification MLP. Cette analyse nous a permis de développer un modèle intelligent et performant que nous avons ensuite intégré dans une plateforme conviviale. Cette plateforme facilite le diagnostic des patients souffrant de la maladie PNDM, que ce soit par des médecins spécialisés, des non-spécialistes, des structures de santé publique/privée ou par le patient lui-même, de manière efficace, facile et rapide. Elle permet également de collecter des informations sur ces maladies pour enrichir les connaissances des médecins et de la communauté, tout en facilitant la communication entre eux.

Les résultats obtenus avec notre modèle de classification sont très prometteurs, affichant une précision élevée. Ces résultats démontrent l'efficacité de notre approche et la robustesse de notre modèle dans le diagnostic des maladies rares. Notre projet revêt une grande importance dans le domaine de la bio-informatique et des soins de santé, en offrant des outils innovants pour le diagnostic et la gestion des maladies rares.

Ces avancées pourraient avoir des implications significatives dans les domaines de la bio-informatique et de la santé, permettant ainsi une meilleure compréhension de la diversité des maladies rares et de leur évolution. En poursuivant nos recherches et en intégrant de nouvelles technologies, nous espérons rendre notre outil encore plus performant et accessible, ouvrant ainsi la voie à des innovations continues dans le diagnostic médical. D'autre part, nous prévoyons d'étendre les fonctionnalités de la plateforme pour inclure la détection et le diagnostic d'autres maladies rares. Cela nécessiterait l'incorporation de nouveaux modèles intelligents adaptés à chaque pathologie. En outre, nous souhaitons enrichir notre base de connaissances en intégrant des articles de recherche, des études de cas et des retours d'expérience des utilisateurs, ce qui pourrait transformer la plateforme en une véritable ressource collaborative pour les professionnels de santé. Enfin, une application pratique de notre projet consisterait à l'utiliser dans des contextes réels, comme les hôpitaux et les cliniques, pour évaluer son efficacité et son impact sur le terrain.

Plan Financier

1- Les couts et charges

1.1 Matériels requis

1.1.1 Achat du mobilier

Mobilier	Prix	Quantité	Total
Bureau	20.000 DA	3	60.000 DA
Chaises	5.000 DA	7	35.000 DA
PC portable	100.000 DA	2	200.000 DA
Imprimante	30.000 DA	1	30.000 DA
Modem	9.000 DA	1	9.000 DA
Pc bureau puissant	130.000 DA	1	130.000 DA
Caméra de surveillance	35.000 DA	3	105.000 DA
Total			569.000 DA

1.1.2 Financement pour lancer notre projet (fond de roulement)

- Matière première: 00 DA
 - Matières diverses (Électricité, internet, l'eau) : 60.000 DA
- **Fonds de roulement (période 6 mois):** matière première + matière divers = 60.000 DA
- **Pour location du locale:**
- Loyer : 600.000 DA

➤ **Les services annexes à prendre en compte lors de l'ouverture du projet :**

- Assurance: 20.000 DA

1.1.3 Cout globale du projet

Achat du mobilier + Font de roulement + Assurance = 569.000 DA + 60.000DA + 20.000 DA = 649.000 DA

2- Calcul prévisionnel du coût de revient du produit

- Matière divers (Électricité, internet, l'eau) : 60.000 DA
- Matière divers (12 mois) : 00 DA *12 = 00 DA
- Amortissement = Coût totale des équipements / 5 (l'âge de tous les équipements est 5 ans) = 569.000 DA / 5 = 113.800 DA
- Loyer : 600.000 DA (**12 mois**)
- Salaire

Employés	Salaire
1 Développeur Full-stack	80.000 DA
1 Expert en sécurité informatique	60.000 DA
1 Responsable du support client	35.000 DA
Totale	175.000 DA

- Salaire (12 mois) : 175.000 DA * 12 = 2.100.000 DA
- Totale des charges fixes = salaire des employés + amortissement + loyer + assurance = 2.100.000 DA + 113.800 DA + 600.000 DA + 20.000 DA = 2.833.800 DA
- Totale des charges variables = matières diverses = 720.000 DA
- Charge totale = Totale des charges fixes +Totale des charges variables = 2.833.800 DA + 720.000 DA = 3.553.800 DA

2.1 Tarifs des ventes

- Horaire de travail : 5/7 et 8 h/24 pendant 11 mois
- Nous comptons intégrer 30 médecins par mois, donc 330 médecins par an
- Le coût d'abonnement = charge totale / 330 = 3.553.800 DA/ 330 = 10.769 DA
- Le bénéfice est de 30%, calculé comme suit : $(10.769 \text{ DA} * 30) / 100 = \mathbf{3.230 \text{ DA}}$

Donc : le prix de vente est = 10.769 DA + **3.230 DA = 13.999 DA**

Désignations	Prix d'abonnement
Abonnement	13.999 DA

2.2 Calcule prévisionnel du chiffre d'affaires de la 1^{ère} année

- Vente mensuelle : $(13.999 \text{ DA} * 30) = 419.970 \text{ DA}$
- Chiffre d'affaires annuel est de : $419.970 \text{ DA} * 11 \text{ mois} = 4.619.670 \text{ DA}$
- Le bénéfice attendu de l'entreprise pour la première année = Chiffre d'affaires annuel - Charge totale = $4.619.670 \text{ DA} - 3.553800 \text{ DA} = \mathbf{1.065.870 \text{ DA}}$

BMC

Model d'affaire économique (BMC) :

Partenaires clés	Activités clés	Proposition de valeur	Relation avec les clients	Segments de clientèle
<ul style="list-style-type: none"> • Professionnels de la santé : les Médecins spécialistes et les médecins non spécialistes. • Établissements de santé publics et privés. • Fournisseurs de technologie : Collaborer avec des entreprises technologiques pour améliorer les capacités et la sécurité de la plateforme. • Fournisseurs de services d'hébergement pour assurer la disponibilité de la plateforme en ligne. • Associations et groupes des 	<ul style="list-style-type: none"> • La plateforme est constamment développée et améliorée afin de satisfaire les besoins des utilisateurs et les progrès technologiques. • Mettre en place des technologies d'intelligence artificielle de pointe pour analyser les données de santé des patients et produire des modèles de prédiction. • Collecte et organisation des données des patients à partir de diverses sources pour garantir la qualité et l'intégrité des données. • Server les utilisateurs de la plateforme médecin_patient d'une assistance 	<ul style="list-style-type: none"> • Amélioration de la précision des diagnostics: Utilisation d'algorithmes d'apprentissage profond alimentés par l'intelligence artificielle pour analyser les données des patients et faciliter des diagnostics plus rapides et plus précis. • La communication et la collaboration : Permettre la communication en temps réel et le partage des connaissances entre les professionnels 	<ul style="list-style-type: none"> • Formation continue et ressources : Proposer des webinaires, des tutoriels et d'autres supports pour améliorer la compréhension des maladies rares et l'utilisation de la plateforme. • Mise à disposition de canaux d'assistance dédiés pour les professionnels de la santé et les patients. • Intégrer le retour d'information des utilisateurs. 	<ul style="list-style-type: none"> • Patients atteints de maladies rares. • Professionnels de la santé: Médecins spécialisés et non spécialisés. • Établissements de santé publics et privés.

<p>patients atteints de maladies rares :</p> <p>Partenariat avec des groupes de défense des patients pour recueillir des informations et améliorer l'engagement des patients.</p> <ul style="list-style-type: none"> • Sponsors. 	<p>technique et d'une résolution rapide des problèmes.</p> <ul style="list-style-type: none"> • Travailler en collaboration avec les fournisseurs de technologie afin d'intégrer les dernières avancées technologiques dans la plateforme. • Création de formations destinées aux professionnels de la santé pour les familiariser avec l'utilisation de la plateforme. 	<p>de santé et les patients pour optimiser les soins.</p> <ul style="list-style-type: none"> • Fournir une base de données riche en informations fiables et en ressources de soutien. 		
	<p>Ressources clés</p>		<p>Canaux de distribution</p>	
	<ul style="list-style-type: none"> • Ressources humaines : Full-stack développeur, Experts en sécurité informatique, Equipe de support client. • Infrastructure : Serveurs et bases de données, outils de maintenance. • Logiciels. • Données des patients. 		<ul style="list-style-type: none"> • Plateforme en ligne accessible aux professionnels de la santé et aux patients. • Partenariats avec des organismes de santé : Collaboration avec des hôpitaux, des cliniques et des instituts de 	

			recherche afin d'intégrer la plateforme dans le secteur de la santé.	
Structure des couts		Sources de revenus		
<ul style="list-style-type: none"> • Frais généraux et administratifs. • Partenariats et licences. • Développement et maintenance de la plateforme. • Frais de serveur et Sécurité. • Support client et formation. • Coûts de marketing et de promotion. 		<ul style="list-style-type: none"> • Frais d'abonnement pour les professionnels et les organisations de santé pour l'accès aux fonctionnalités de la plateforme. • Partenariats avec les organisations de santé et les instituts de recherche. • Vente de données de patients anonymisées à des instituts de recherche à des fins de recherche médicale. • Services des formations personnalisées et des analyses de données. 		

Références

Référence :

[1] Clejeune. (2023, 1 mars). Maladie orpheline : les réponses à vos questions. Date de consultation : 14-03-2024, Fai2r. URL : <https://www.fai2r.org/les-types-de-pathologies/maladie-orpheline/>

[2] Des solutions thérapeutiques pour des maladies rares - LFB. (2021, 1 février). Date de consultation: 14-03-2024, LFB Corp. URL : <https://www.groupe-lfb.com/responsabilites/les-maladies-rares/>

[3] Mariela, Yaneva, –, Deliverska. (2011). Rare diseases and genetic discrimination. doi: 10.5272/JIMAB.2011171.116

[4] Calvi, T. (s. d.). Sanofi et MIS s'associent pour diagnostiquer les maladies rares grâce à l'Intelligence Artificielle. ActuaIA. Date de consultation : 16-03-2024, URL <https://www.actuia.com/actualite/sanofi-et-mis-sassocient-pour-diagnostiquer-les-maladies-rares-grace-a-lintelligence-artificielle/>

[5] Catherine, Dekeuwer. (2015). Defining Genetic Disease. doi: 10.1007/978-94-017-8887-8_7

[6] Ashima, Natta. (2016). Infectious Disease: An Overview of its Causes, Symptoms and Treatment. Research and reviews: journal of medical and health sciences.

[7] Daniel G Bausch, Pierre E Rollin, La fièvre de Lassa, Annales de l'Institut Pasteur / Actualités, Volume 8, Issue 3, 1997, Pages 223-231, ISSN 0924-4204, [https://doi.org/10.1016/S0924-4204\(97\)87656-1](https://doi.org/10.1016/S0924-4204(97)87656-1).

[8] Maladies métaboliques - la prise en charge à l'Hôpital des enfants | HUG - Hôpitaux Universitaires de Genève. (s. d.). Date de consultation : 17-03-2024. URL <https://www.hug.ch/enfants-ados/specialites-medicales-chirurgicales/maladies-metaboliques>.

[9] Fernandez, J., & Msd, M. (2022, 11 octobre). Maladies auto-immunes. Manuels MSD Pour le Grand Public. Date de consultation : 17-03-2024. URL <https://www.msmanuals.com/fr/accueil/troubles-immunitaires/r%C3%A9actions-allergiques-et-autres-troubles-d%E2%80%99hypersensibilit%C3%A9/maladies-auto-immunes>

[10] Lafaurie, L. (2021, 7 avril). Néoplasie : définition, symptômes ; mes, cause, traitements. Date de consultation: 17-03-2024. URL <https://sante.journaldesfemmes.fr/fiches-maladies/2707329-neoplasie-definition-symptomes-cause-traitements-pulmonnaire-mammaire-colique/>

[11] Accueil - Filières de santé maladies rares. (2024, 15 février). Filières de Santé Maladies Rares. Date de consultation: 17-03-2024. URL <https://www.filieresmaladiesrares.fr/>

- [12] Mariela, Yaneva, –, Deliverska. (2011). Rare diseases and genetic discrimination. doi: 10.5272/JIMAB.2011171.116
- [13] Causes et caractéristiques. (s. d.). sciensano.be. Date accessed : 19-03-2024. URL <https://www.sciensano.be/fr/sujets-sante/maladies-rares/causes-et-caracteristiques>
- [14] Juan, Benito-Lozano., Blanca, López-Villalba., Greta, Arias-Merino., Manuel, Posada, de, la, Paz., Verónica, Alonso-Ferreira. (2022). Diagnostic delay in rare diseases: data from the Spanish rare diseases patient registry. *Orphanet Journal of Rare Diseases*, doi: 10.1186/s13023-022-02530-3
- [15] "Health Awareness Campaigns and Their Role in Promoting Awareness of Rare Diseases." *Journal of Design Sciences and Applied Arts (Print)*, undefined (2023). doi: 10.21608/jdsaa.2023.173961.1236
- [16] Caijun, Zhu., Wei, Hong., L., Ding., Xue, Zhang. (2022). Towards key scientific questions in the diagnosis and treatment of rare diseases: Summary from the 297th Meeting of the Shuangqing Forum. *Zoological Research*, doi: 10.24272/j.issn.2095-8137.2022.068
- [17] dzierzyslawappxg819. (2022). Overcoming challenges in research and development of rare eye diseases. *Indian journal of ophthalmology*, doi: 10.4103/ijo.ijo_1083_22
- [18] Noa, Rosenberg., Sibren, van, den, Berg., Nina, N., Stolwijk., Bart, Jacobs., Hendrika, C., Post., Anna, M.G., Pasmooij., Saco, J., de, Visser., Carla, E., M., Hollak. (2023). Access to medicines for rare diseases: A European regulatory roadmap for academia. *Frontiers in Pharmacology*, 14 doi: 10.3389/fphar.2023.1142351
- [19] Lucio, Luzzatto., Julie, Makani. (2022). Treating Rare Diseases in Africa: The Drugs Exist but the Need Is Unmet. *Frontiers in Pharmacology*, doi: 10.3389/fphar.2021.770640
- [20] Qiaoqiao, Han., Hengtao, Fu., Xiaoyue, Chu., Ruixin, Wen., Mia, Zhang., Tao, You., Peng, Fu., Jian, Qin., Tao, Cui. (2022). Research advances in treatment methods and drug development for rare diseases. *Frontiers in Pharmacology*, doi: 10.3389/fphar.2022.971541
- [21] Maladies rares : plus d'équité pour les patients. (2024, février 29). Sanofi. Date de consultation : 25-03-2024. URL <https://www.sanofi.fr/fr/media/sanofi-today/votre-sante/maladies-rares-plus-d-equite-pour-les-patients>
- [22] Arnaud, C. (2021, 28 février). Journée mondiale des maladies rares : isolement et troubles psychologiques, Date de consultation : 25-03-2024 URL <https://www.carenity.com/infos-maladie/magazine/actualites/journee-mondiale-des-maladies-rares-isolement-et-troubles-psychologiques-parlonsen-1644>

- [23] Robert, J. (2024, 14 mars). Machine Learning : Définition, fonctionnement, utilisations. Formation Data Science | DataScientest.com. Date de consultation : 26-03-2024. URL <https://datascientest.com/machine-learning-tout-savoir>
- [24] Zouinar, M. (2020). Evolutions of Artificial Intelligence : What issues for human activities and Human-Machine relationships at work ? *Activités*, 17 1. Date de consultation : 26-03-2024. URL <https://doi.org/10.4000/activites.4941>
- [25] Algorithmes d'apprentissage automatique bases sur NIF exploration des techniques avancees (2024). Date de consultation : 27-03-2024. URL <https://fastercapital.com/fr/contenu/Algorithmes-d-apprentissage-automatique-bases-sur-NIF---exploration-des-techniques-avancees.html>
- [26] Wenling, Li. (2023). The impact of computing and machine learning on complex problem-solving. *Engineering reports*, doi: 10.1002/eng2.12702
- [27] Sheng, Chieh, Lu., Christine, Leon, Swisher., Caroline, Chung., David, A., Jaffray., Chris, Sidey-Gibbons. (2023). On the importance of interpretable machine learning predictions to inform clinical decision making in oncology. *Frontiers in Oncology*, doi: 10.3389/fonc.2023.1129380
- [28] (2023). A survey on medical and diseases prediction using machine learning. *International journal of engineering technology and management sciences*, doi: 10.46647/ijetms.2023.v07i02.069
- [29] Woo, P., & Woo, P. (2024, 5 février). Apprentissage supervisé : essentiel à l'&rsquo ; apprentissage automatique. Date de consultation : 30-03-2024. URL LeDigitaliseur. <https://ledigitaliseur.fr/ia/apprentissage-supervise/>
- [30] Pierre, Geurts., Alexandre, Irrthum., Louis, Wehenkel. (2009). Supervised learning with decision tree-based methods in computational and systems biology. *Molecular BioSystems*, doi: 10.1039/B907946G
- [31] (2023). A Supervised Learning Regression method for the analysis of oral sensitivity of healthy subjects and patients with chemosensory loss. doi: 10.21203/rs.3.rs-2912168/v1
- [32] David, R., Musicant., Janara, Christensen., J.F., Olson. (2007). Supervised Learning by Training on Aggregate Outputs. doi: 10.1109/ICDM.2007.50
- [33] Mohamed, A., Mahfouz. (2023). Incorporating density in k-nearest neighbors regression. *International Journal of Advanced Research in Computer Science*, doi: 10.26483/ijarcs.v14i3.6989
- [34] Nicolae, Marginean., Janetta, Sirbu., Dan, Racovițan. (2010). Decision trees - a perspective of electronic decisional support. doi: 10.29302/OECONOMICA.2010.12.2.15

- [35] Hui, Xue., Qiang, Yang., Songcan, Chen. (2009). SVM: Support Vector Machines. doi: 10.1201/9781420089653-10
- [36] Kyle, Hsu., Sergey, Levine., Chelsea, Finn. (2018). Unsupervised Learning via Meta-Learning. arXiv: Learning <https://doi.org/10.48550/arXiv.1810.02334>
- [37] Marina, Forell. (2023). Unsupervised Learning. doi: 10.1007/978-3-031-28674-2_14
- [38] Hanae, Moussaoui., Nabil, El, Akkad., Mohamed, Benslimane. (2023). Reinforcement Learning: A review. International Journal of Computing and Digital Systems, doi: 10.12785/ijcds/1301118
- [39] Avani, P. (2021, 14 décembre). Biggest Confusion : AI vs ML vs Deep Learning - Popat Avani - Medium. Medium. Date de consultation : 04-04-2024. URL <https://popatavani666.medium.com/biggest-confusion-ai-vs-ml-vs-deep-learning-8aa11343fd12>
- [40] Notions de base sur l'apprentissage profond. (s. d.). Date de consultation : 04-04-2024. URL https://rstudio-pubs-static.s3.amazonaws.com/952034_2603404c75594d7db90577c1e8cdb5d4.html#r%C3%A9seaux-de-neurones-%C3%A0-convolution-cnn
- [41] Muhammet, Unal., Ayça, Ak., Vedat, Topuz., Hasan, Erdal. (2013). Artificial Neural Networks. doi: 10.1007/978-3-642-32900-5_2
- [42] Introduction à l'apprentissage profond (deep learning) de l'intelligence artificielle —. (s. d.). Date de consultation : 06-04-2024. URL <https://culturesciencesphysique.ens-lyon.fr/ressource/IA-apprentissage-Rousseau.xml>
- [43] Jumelle, M. (s. d.). Réseaux de neurones : tout comprendre. Formation Tech et Data en ligne | Blent.ai. 06-04-2024. URL <https://blent.ai/blog/a/reseaux-de-neurones-tout-comprendre>
- [44] Keldenich, T. (2023, 8 décembre). Fonction d'activation : Comment elle fonctionne ? - Meilleur Guide. Inside Machine Learning. Date de consultation : 07-04-2024. URL <https://inside-machinelearning.com/fonction-dactivation-comment-ca-marche-une-explication-simple/>
- [45] Introduction aux perceptrons multi-couches (réseaux de neurones à propagation avant) — Notes de cours IFT6266 Hiver 2012. (s. d.). Date de consultation : 07-04-2024. URL https://www.iro.umontreal.ca/~pift6266/H12/html/mlp_fr.html
- [46] Belaidi, N. (s. d.). Réseaux récurrents (RNN) : on vous explique tout. Formation Tech et Data en ligne | Blent.ai. Date de consultation : 07-04-2024. URL <https://blent.ai/blog/a/rnn-on-vous-explique-tout>

- [47] Qu'est-ce qu'un réseau de neurones convolutifs ? | IBM. (s. d.). Date de consultation : 08-04-2024. URL <https://www.ibm.com/fr-fr/topics/convolutional-neural-networks>
- [48] Kassel, R. (2023, 9 novembre). Réseaux antagonistes génératifs ou GAN : Qu'est-ce que c'est ? Formation Data Science | DataScientest.com. Date de consultation : 08-04-2024. URL <https://datascientest.com/generative-adversarial-network-tout-savoir>
- [49] Jacques, Beltrand., Jacques, Beltrand., Kanetee, Busiah., Laurence, Vaivre-Douret., Laurence, Vaivre-Douret., Anne, Laure, Fauret., Marianne, Berdugo., Marianne, Berdugo., Hélène, Cavé., Michel, Polak., Michel, Polak. (2020). Neonatal Diabetes Mellitus.. *Frontiers in Pediatrics*, doi: 10.3389/FPED.2020.540718
- [50] Orphanet : Diabète néonatal transitoire. (s. d.). 08-04-2024. URL <https://www.orpha.net/fr/disease/detail/99886#:~:text=Forme%20g%C3%A9n%C3%A9tiquement%20h%C3%A9r%C3%A9rog%C3%A8ne%20de%20diab%C3%A8te,chez%20la%20plupart%20des%20patients>
- [52] Sara Al-Khawaga, Mohammed Idris, Saras Saraswathi, Basma Haris, Reem Hasnah, Amira Saeed, Hakeem Almabrazi, Najeeb Syed, Jithesh Puthen V., Ahmed El Awwa, Amal Khalifa, Fawziya Al-Khalaf, Goran Petrovski, Essam M. Abdelalim, Khalid Hussain. (2019). The clinical and genetic characteristics of permanent neonatal diabetes (PNDM) in the state of Qatar. *Molecular Genetics & Genomic Medicine*, doi: 10.1002/MGG3.753.
- [53] Luís, Edmundo, Teixeira, de, Arruda, Furtado. (2023). A predictive model for early detection of diabetes mellitus using machine learning. doi: 10.58694/20.500.12479/1592
- [54] An, N. K. (s. d.). Sulfamides hypoglycémiantes. *Pharmacomedicale*. Date de consultation : 16-04-2024. URL <https://pharmacomedicale.org/medicaments/par-specialites/item/sulfamides-hypoglycemiantes>
- [55] Orphanet : Diabète néonatal. (s. d.). 16-04-2024. URL <https://www.orpha.net/fr/disease/detail/224>
- [56] Roopa, Kanakatti, Shankar., Catherine, Pihoker., Lawrence, M., Dolan., Debra, Standiford., Angela, Badaru., Dana, Dabelea., Beatriz, L., Rodriguez., Mary, Helen, Black., Giuseppina, Imperatore., Andrew, T., Hattersley., Sian, Ellard., Lisa, K., Gilliam. (2012). Permanent Neonatal Diabetes Mellitus: Prevalence and Genetic Diagnosis in the SEARCH for Diabetes in Youth Study. *Pediatric Diabetes*, doi: 10.1111/PEDI.12003
- [57] Karolina, Antosik., Piotr, Gnys., Elisa, De, Franco., Maciej, Borowiec., Malgorzata, Mysliwiec., Sian, Ellard., Wojciech, Młynarski. (2016). Single patient in GCK-MODY family successfully re-diagnosed into GCK-PNDM through targeted

next-generation sequencing technology. *Acta Diabetologica*, doi: 10.1007/S00592-015-0786-0

[58] Mgozlan. (2024, 2 mars). Diabète néonatal : quand la maladie se manifeste dans les six premiers mois de la vie. *Le Diabète Dans Tous Ses États*. <https://www.le-diabete-dans-tous-ses-etats.precidiab.org/autres-types-de-diabete/quand-le-diabete-se-manifeste-dans-les-six-premiers-mois-de-la-vie/#:~:text=Les%20mutations%20sur%20les%20g%C3%A8nes,aux%20mutations%20du%20canal%20potassique>

[59] Figure 1 : Schematic view of chromosome 6 where the PLAGL1 gene is. . . (s. d.). ResearchGate. https://www.researchgate.net/figure/Schematic-view-of-chromosome-6-where-the-PLAGL1-gene-is-located-in-the-long-arm-of-this_fig1_360261444

[60] Adriana, Mangué, Esquiaveto-Aun., Maricilda, Palandi, de, Mello., Maria, Fernanda, Vanti, Macedo, Paulino., Walter, José, Minicucci., Gil, Guerra-Júnior., Sofia, Helena, Valente, de, Lemos-Marini. (2015). A new compound heterozygosis for inactivating mutations in the glucokinase gene as cause of permanent neonatal diabetes mellitus (PNDM) in double-first cousins. *Diabetology & Metabolic Syndrome*, doi: 10.1186/S13098-015-0101-9

[61] Monica, T., Marin., Michael, L., Coffey., Joni, Beck., Paul, S., Dasari., Rebecca, Allen., Sowmya, Krishnan. (2016). A Novel Approach to the Management of Neonatal Diabetes Using Sensor-Augmented Insulin Pump Therapy With Threshold Suspend Technology at Diagnosis. *Diabetes Spectrum*, doi: 10.2337/DIASPECT.29.3.176

[62] Conor, McClenaghan., Novella, Rapini., Domenico, Umberto, De, Rose., Jian, Gao., Jacob, Roeglin., Carla, Bizzarri., Riccardo, Schiaffini., Eloisa, Tiberi., Mafalda, Mucciolo., Annalisa, Deodati., Alessandro, Perri., Giovanni, Vento., Fabrizio, Barbetti., Colin, G., Nichols., Stefano, Cianfarani. (2022). Sulfonylurea-Insensitive Permanent Neonatal Diabetes Caused by a Severe Gain-of-Function Tyr330His Substitution in Kir6.2. *Hormone Research in Paediatrics*, doi: 10.1159/000521858

[63] *Actualités en Pharmacologie Clinique - PDF Téléchargement Gratuit*. (s. d.). Date de consultation : 22-04-2024. URL <https://docplayer.fr/9104109-Actualites-2014-2015-en-pharmacologie-clinique.html>

[64] National Center for Biotechnology Information. (2017 Dec 16). Neonatal Diabetes Mellitus: An Update on Diagnosis and Management Date de consultation : 22-04-2024. URL <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5928785/>

[65] Shara, S., Azad., Esmá, R., Isenovic., Subhashini, Yaturu., Shaker, A., Mousa. (2013). Insulin Therapy for Diabetes. doi: 10.5772/56379

[66] National Center for Biotechnology Information. (2022 Sep 30). Associations of Age at Diagnosis and Duration of Diabetes With Morbidity and Mortality Among Older Adults. Date de consultation : 23-04-2024. URL <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9526092/>

[67] The Lancet Diabetes and Endocrinology. September 11, 2023. Life expectancy associated with different ages at diagnosis of type 2 diabetes in high-income countries: 23 million person-years of observation. Date de consultation : 23-04-2024. URL <https://www.thelancet.com/journals/landia/article/PIIS2213-8587%2823%2900223-1/fulltext#%20>

[68] Nanayakkara, N., Curtis, A. J., Heritier, S., Gadowski, A. M., Pavkov, M. E., Kenealy, T., Owens, D. R., Thomas, R. L., Song, S., Wong, J., Chan, J. C., Luk, A. O., Penno, G., Ji, L., Mohan, V., Amutha, A., Romero-Aroca, P., Gasevic, D., Magliano, D. J., . . . Zoungas, S. (2020). Impact of age at type 2 diabetes mellitus diagnosis on mortality and vascular complications : systematic review and meta-analyses. *Diabetologia*, 64(2), 275-287. <https://doi.org/10.1007/s00125-020-05319-w>

[69] World Health Organization. (2011). Glycated haemoglobin (HbA1c) for the diagnosis of diabetes. Use Of Glycated Haemoglobin (HbA1c) In The Diagnosis Of Diabetes Mellitus - NCBI Bookshelf. Date de consultation : 25-04-2024. URL <https://www.ncbi.nlm.nih.gov/books/NBK304271/>

[70] Seery, C. (2023, 29 octobre). What is HbA1c ? - Definition, Units, Conversion, Testing & Control. *Diabetes*. <https://www.diabetes.co.uk/what-is-hba1c.html>

[71] National Center for Biotechnology Information. (2017 Dec 16). Neonatal Diabetes Mellitus: An Update on Diagnosis and Management. Date de consultation : 25-04-2024. URL <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5928785/>

[72] Aguilar-Bryan, L., & Bryan, J. (2008). Neonatal diabetes mellitus. *Endocrine Reviews*, 29(3), 265-291. Date de consultation : 26-04-2024. URL <https://doi.org/10.1210/er.2007-0029>

[73] National Center for Biotechnology Information. (2019 Sep 23). Postnatal growth of infants with neonatal diabetes: Insulin pump (CSII) versus Multiple Daily Injection (MDI) therapy. Date de consultation : 28-04-2024. URL <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7233682/>

[74] Adriana, Mangué, Esquiaveto-Aun., Maricilda, Palandi, de, Mello., Maria, Fernanda, Vanti, Macedo, Paulino., Walter, José, Minicucci., Gil, Guerra-Júnior., Sofia, Helena, Valente, de, Lemos-Marini. (2015). A new compound heterozygosis for inactivating mutations in the glucokinase gene as cause of permanent neonatal

diabetes mellitus (PNM) in double-first cousins. *Diabetology & Metabolic Syndrome*, doi: 10.1186/S13098-015-0101-9

[75] Karolina, Antosik., Piotr, Gnys., Elisa, De, Franco., Maciej, Borowiec., Malgorzata, Mysliwiec., Sian, Ellard., Wojciech, Młynarski. (2016). Single patient in GCK-MODY family successfully re-diagnosed into GCK-PNMD through targeted next-generation sequencing technology. *Acta Diabetologica*, doi: 10.1007/S00592-015-0786-0

[76] Cristina, Colom., Josep, Oriola., Silvia, Martinez., Francisco, Blanco-Vaca., Francisco, Blanco-Vaca., Roser, Casamitjana., Rosa, Corcoy., Rosa, Corcoy. (2013). Family history in the diagnosis of monogenic diabetes "leads and misleads".. *Hormones (Greece)*, doi: 10.1007/BF03401313

[77] Anna, Parkkola., Maaret, Turtinen., Taina, Härkönen., Jorma, Ilonen., Mikael, Knip. (2021). Family history of type 2 diabetes and characteristics of children with newly diagnosed type 1 diabetes. *Diabetologia*, doi: 10.1007/S00125-020-05342-X

[78] BILITY. (s. d.). Définition Visual Studio Code - Bility - Agence de développement web sur-mesure. Date de consultation : 03-05-2024. URL <https://bility.fr/definition-visual-studio-code/>

[79] Google Colab. (s. d.). Date de consultation : 04-05-2024. URL <https://research.google.com/colaboratory/faq.html?hl=fr>

[80] Le tutoriel Python — Documentation Python 3.10.13. (s. d.). Date de consultation : 04-05-2024. URL <https://docs.python.org/fr/3.10/tutorial/>

[81] Rédaction, L. (s. d.). HTML : qu'est-ce que c'est ? Futura. Date de consultation : 05-05-2024. URL <https://www.futura-sciences.com/tech/definitions/internet-html-480/>

[82] Archambeau, T. (s. d.). HTML - Définition. Date de consultation : 05-05-2024. URL <https://glossaire.infowebmaster.fr/html/>

[83] Ionos, L. É. (2021, 12 juillet). Ceci est un titre. IONOS Digital Guide. <https://www.ionos.fr/digitalguide/sites-internet/web-design/quest-ce-que-le-css/>

[84] Rédaction, L. (s. d.-a). CSS : qu'est-ce que c'est ? Futura. Date de consultation : 05-05-2024. URL <https://www.futura-sciences.com/tech/definitions/internet-css-4050/>

[85] Guersan, G. (2022, 6 octobre). Qu'est-ce que le JavaScript ? - Définition développement web. Guillaume Guersan. Date de consultation : 07-05-

2024. URL <https://guersanguillaume.com/ressources/glossaire-marketing-digital/definition-javascript/>
- [86] Qu'est-ce PostgreSQL. (s. d.). Oracle France. Date de consultation : 07-05-2024. URL <https://www.oracle.com/fr/database/definition-postgresql/>
- [87] Sufyan, bin, Uzayr. (2022). Mastering Django. doi: 10.1201/9781003310495
- [88] Alok, Kumar, Srivastava. (2022). Django , The Python Web Framework. Indian Scientific Journal Of Research In Engineering And Management, doi: 10.55041/ijsrem13183
- [89] Tim, Ewing., Jillian, Redfern., Amanda, Alexander., Richard, Medina., Emma, Birath. (2021). Django as a Mission Planning Tool Interface for the CYGNSS Mission. doi: 10.1109/AERO50100.2021.9438383
- [90] BurchCarl. (2010). Django, a web framework using Python. Journal of Computing Sciences in Colleges, doi: 10.5555/1747137.1747166
- [91] Introduction to Django. (s. d.). Date de consultation : 10-05-2024. URL https://www.w3schools.com/django/django_intro.php
- [92] Sufyan, bin, Uzayr. (2022). Bootstrap. doi: 10.1201/9781003309383
- [93] Sufyan, bin, Uzayr. (2022). Mastering Bootstrap. doi: 10.1201/9781003310501
- [94] Diego, Javier, Bastidas-Logroño., Oscar, Omar, Espíndola-Lara., Alex, Darío, Palma-Rivera. (2020). Implementación del bootstrap como una metodología ágil en la web. doi: 10.35381/R.K.V5I9.648
- [95] NumPy documentation — NumPy v1.26 Manual. (s. d.). Date de consultation : 12-05-2024. URL <https://numpy.org/doc/stable/>
- [96] pandas documentation — pandas 2.2.2 documentation. (s. d.). Date de consultation : 12-05-2024. URL <https://pandas.pydata.org/docs/>
- [97] matplotlib. (2024, 15 mai). PyPI. Date de consultation : 14-05-2024. URL <https://pypi.org/project/matplotlib/#description>
- [98] tensorflow-cpu. (2024, 9 mars). PyPI. Date de consultation : 14-05-2024. URL <https://pypi.org/project/tensorflow-cpu/>
- [99] Liangqu, Long., Xiang-Ying, Zeng. (2022). Keras Advanced API. doi: 10.1007/978-1-4842-7915-1_8

- [100] Max, Pumperla., Daniel, Cahall. (2022). Elephas: Distributed Deep Learning with Keras & Spark. The Journal of Open Source Software, doi: 10.21105/joss.04073
- [101] Melanie. (2023, 30 octobre). Scikit Learn : Discover the Python library dedicated to Machine Learning. Data Science Courses | DataScientest. <https://datascientest.com/en/scikit-learn-discover-the-python-library-dedicated-to-machine-learning>
- [102] Melanie. (2024b, mars 26). Imbalanced Learn : the Python library for rebuilding ML datasets. Data Science Courses | DataScientest. Date de consultation : 20-05-2024. URL <https://datascientest.com/en/imbalanced-learn-the-python-library-for-rebuilding-ml-datasets#:~:text=Designed%20to%20address%20problems%20related,classes%20and%20improve%20model%20performance.>
- [103] Sharma, P. (2023, 27 février). How to Save and Load Machine Learning Models in Python Using Joblib Library ? Analytics Vidhya. <https://www.analyticsvidhya.com/blog/2023/02/how-to-save-and-load-machine-learning-models-in-python-using-joblib-library/>
- [104] shutil — High-level file operations. (s. d.). Python Documentation. Date de consultation : 22-05-2024. URL <https://docs.python.org/3/library/shutil.html>
- [105] Ebner, J. (2023, 27 décembre). Cross validation, explained - sharp sight. Sharp Sight. Date de consultation : 22-05-2024. URL <https://www.sharpsightlabs.com/blog/cross-validation-explained/>
- [106] Gupta, R. (2023, 6 avril). Accuracy, Precision, Recall, F-1 Score, Confusion Matrix, and AUC-ROC. Medium. <https://medium.com/@riteshgupta.ai/accuracy-precision-recall-f-1-score-confusion-matrix-and-auc-roc-1471e9269b7d>
- [107] Tremblay, C. (2023, 4 février). Sensitivity, Specificity et ROC curve. Kobia. Date de consultation : 23-05-2024. URL <https://kobia.fr/classification-metrics-sensitivity-specificity-roc/>
- [108] Henry Semakula, Luís, Edmundo, Teixeira, de, Arruda, Furtado. (2023). A predictive model for early detection of diabetes mellitus using machine learning. doi: 10.58694/20.500.12479/1592

Année universitaire : 2023-2024

**Présenté par : REDJEM Nassim
GASSI Mostefa Habib Errahmane**

Thème

Une plateforme médecin/patient pour la prise en charge intelligente des patients atteints de maladies rares

Mémoire pour l'obtention du diplôme de Master en Bio-informatique

Les maladies rares présentent des défis uniques en matière de diagnostic, de traitement et de gestion des patients et sont souvent mal comprises. Ce travail de master présente la conception et la réalisation d'une plateforme innovante destinée au diagnostic des maladies rares en commençant par le diabète néonatal permanent mellitus (PNDM). Cette initiative vise à combler les lacunes existantes dans le diagnostic rapide et précis de ces maladies, également à collecter des informations sur ces maladies pour enrichir les connaissances des médecins et de la communauté, tout en facilitant la communication entre eux. La plateforme se base sur l'utilisation de modèles de l'apprentissage profond, une technique avancée de l'intelligence artificielle permettant d'analyser et d'interpréter des informations médicales complexes. Ces modèles ont été entraînés sur des ensembles de données spécifiques aux maladies rares, avec une première application ciblée sur le PNDM. Le choix du PNDM comme point de départ découle de l'importance de cette maladie et des défis qu'elle pose en termes de diagnostic précoce et de traitement. Le modèle a une performance globale solide avec une valeur élevée d'exactitude de 89%.

Mots-clefs : Apprentissage Profond, Plateforme intelligente, Diagnostic, Maladies rares, Intelligence Artificiel, PNDM.

Président du jury : BELLIL Ines (Pr)

Encadrant : BOUCHEHAM Anouar (MCA)

Co-encadrant : CHEHILI Hamza (MCA)

Examineur : DAAS Mohamed Skander (MCA)