



**REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE**  
**Ministère de l'Enseignement Supérieur et de la Recherche Scientifique**  
**Université Mohamed Khider – BISKRA**

**Faculté des Sciences Exactes, des Sciences de la Nature et de la Vie**

**Département d'informatique**

N° d'ordre : SIOD03/M2/2022

## **Mémoire**

Présenté pour obtenir le diplôme de master académique en

# **Informatique**

Parcours : Systèmes d'information, Optimisation et Décision (SIOD)

---

## **Prédire l'arrêt cardiaque à l'aide de l'apprentissage profond**

---

**Par :**

**KHERCHOUCHE JAOUIDA**

Soutenu le 22/06/2022 devant le jury composé de :

Chighoub Fouzia

MCB

Président

MEADI Mohamed Nadjib

MCB

Rapporteur

Chami Djazia

MAA

Examineur

Année universitaire 2021-2022

# الجمهورية الجزائرية الديمقراطية الشعبية

وزارة التعليم العالي والبحث العلمي

جامعة محمد خيضر بسكرة

## تصريح شرفي

(خاص بالالتزام بقواعد النزاهة العلمية لإنجاز بحث)

أنا الممضي أسفله،

السيد (ة): خرشوش جويده الصفة: انثى طالب: ثانية ماستر تحسين نظم المعلومات واتخاذ القرار .

الحامل لبطاقة التعريف الوطنية رقم: 201473872 والصادرة بتاريخ: 30/04/2017

المسجل بكلية: العلوم الدقيقة و علوم الطبيعة و الحياة

قسم: الإعلام الآلي

والمكلف بإنجاز مذكرة تخرج في الماستر

عنوانها: Prédire l'arrêt cardiaque a l'aide de l'apprentissage profond

أصرح بشرفي أنني ألتزم بمراعاة المعايير العلمية والمنهجية ومعايير الأخلاقيات المهنية والنزاهة الأكاديمية المطلوبة في إنجاز البحث المذكور أعلاه.

التاريخ: 21/06/2022

توقيع المعني:





**REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE**  
**Ministère de l'Enseignement Supérieur et de la Recherche Scientifique**  
**Université Mohamed Khider – BISKRA**

**Faculté des Sciences Exactes, des Sciences de la Nature et de la Vie**

**Département d'informatique**

N° d'ordre : SIOD03/M2/2022

## **Mémoire**

Présenté pour obtenir le diplôme de master académique en

# **Informatique**

Parcours : Systèmes d'information, Optimisation et Décision (SIOD)

---

## **Prédire l'arrêt cardiaque à l'aide de l'apprentissage profond**

---

**Par :**

**KHERCHOUCHE JAOUIDA**

Soutenu le 22/06/2022 devant le jury composé de :

Chighoub Fouzia

MCB

Président

MEADI Mohamed Nadjib

MCB

Rapporteur

Chami Djazia

MAA

Examineur

Année universitaire 2021-2022

## *Dedication*

*Je tiens tout d'abord, à remercier ALLAH le tout puissant qui m'a donné la force, la volonté, et la patience d'accomplir ce modeste travail .*

*Je remercie le flambeau de ma vie et de mon soutien moral, ma chère mère. Et celui à qui je dois mon respect et mon amour est mon cher père qui m'a donné courage et soutien psychologique.*

*Ensuite, je tiens à exprimer mes vifs remerciements à monsieur le professeur Miadi Najib, mon encadreur de recherche pour ses efforts, ses conseils, et sa disponibilité.*

*Je dédie également ce message à tous les membres de ma famille et à tous mes amis et surtout Benzekri Mohamed, Kharashi Dina et Harez Oumaima qui m'ont soutenu tout au long du processus. J'apprécierai toujours tout ce qu'ils ont fait.*

# Abstract

Heart and vascular diseases are the main cause of the high number of deaths on the planet, so it has become very necessary to detect and classify heart diseases to reduce the risk of developing them. Predicting heart disease is the toughest challenge in healthcare. Deep learning and wearable healthcare technologies have helped us in this industry as it is constantly evolving to make the most of patient data. We can use this data to analyze and predict heart disease at low cost with good diagnostic accuracy. The objective of this project is to develop a heart disease prediction system using deep learning, our deep learning algorithm is used to predict a person's risk of developing heart disease based on characteristics (e.g., cholesterol, blood pressure, age, sex, etc.) that were extracted from the datasets.

We collected the data used from records provided by the kaggle website and from patient data which is the risk factors for heart disease and is similar to heart disease data on the machine learning website. 'UCI, we suggest to use 1D CNN in training and test stages to classify data and predict cardiac arrest, the results we got are good.

Keywords : deep learning, machine learning, one-dimensional convolutional neural network, data processing, training, testing.

# Résumé

Les maladies cardiaques et vasculaires sont la principale cause du nombre élevé de décès sur la planète, il est donc devenu très nécessaire de détecter et de classer les maladies cardiaques pour réduire le risque de les développer. Prédire les maladies cardiaques est le défi le plus difficile dans le secteur de la santé. L'apprentissage profond et les technologies de santé portables nous ont aidés dans ce secteur, car il évolue constamment pour tirer le meilleur parti des données des patients. Nous pouvons utiliser ces données pour les analyser et prédire les maladies cardiaques à faible coût avec une bonne précision diagnostique.

L'objectif de ce projet est de développer un système de prédiction des maladies cardiaques en utilisant l'apprentissage en profondeur, notre algorithme d'apprentissage en profondeur il est utilisé pour prédire le risque d'une personne de développer une maladie cardiaque en fonction de caractéristiques (par exemple, cholestérol, tension artérielle, âge, sexe, etc.) qui ont été extraites des ensembles de données. Nous avons collecté les données utilisées à partir des enregistrements fournis par le site Web kaggle et provenant des données des patients qui sont les facteurs de risque de maladie cardiaque et sont similaires aux données sur les maladies cardiaques sur le site Web d'apprentissage automatique de l'UCI, nous suggérons d'utiliser 1D CNN dans les étapes de formation et de test pour classer les données et prédire l'arrêt cardiaque, les résultats que nous avons obtenus sont bons.

Mots-clés : apprentissage profond, apprentissage automatique, Réseau neuronal convolutif unidimensionnel, traitement de données, formation, tests.

## تلخيص

تعد أمراض القلب والأوعية الدموية السبب الرئيسي لارتفاع عدد الوفيات على هذا الكوكب ، لذلك أصبح من الضروري للغاية الكشف عن أمراض القلب وتصنيفها لتقليل مخاطر الإصابة بها.

إن التنبؤ بأمراض القلب هو التحدي الأصعب في مجال الرعاية الصحية لقد ساعدنا التعلم العميق وتقنيات الرعاية الصحية القابلة للارتداء في هذا المجال حيث لأنها في تطور مستمر لتحقيق أقصى استفادة من بيانات المرضى ، ويمكننا استخدام هذه البيانات لتحليل والتنبؤ بأمراض القلب بتكلفة منخفضة وبخير دقة التشخيص.

الهدف من هذا المشروع هو تطوير نظام للتنبؤ بالأمراض القلبية باستخدام التعلم العميق خوارزمية التعلم العميق التي نتبعها تستعمل للتنبؤ باحتمال إصابة الانسان بأمراض القلب على أساس الميزات (مثل الكوليسترول ، ضغط الدم والعمر والجنس وغيرها ) التي تم استخراجها من مجموعة البيانات.

قمنا بجمع البيانات المستخدمة من السجلات التي يقدمها موقع kaggle على الويب و مصدرها هو بيانات المريض و التي تتمثل في عوامل الخطر لحالة القلب و هي مشابهة لبيانات امراض القلب الموجودة في موقع UCI للتعلم الآلي. نقترح استخدام 1D CNN في خطوات التدريب و الاختبار لتصنيف البيانات و التنبؤ بالسكتة القلبية النتائج التي حصلنا عليها جيدة.

الكلمات المفتاحية: التعلم العميق ، التعلم الآلي ، الشبكة العصبية الالتفافية أحادية البعد ، معالجة البيانات، التدريب ، الاختبار.

# Table des matières

<b>Introduction générale</b>	<b>3</b>
<b>1 Structure et fonction du cœur</b>	<b>5</b>
1.1 Introduction	5
1.2 Définition et rôle et du coeur	5
1.2.1 Ventricule droit	6
1.2.2 Ventricule gauche	6
1.2.3 Oreillette gauche	6
1.2.4 Oreillette droite	7
1.3 Circulation sanguine	7
1.4 Cycle cardiaque	7
1.5 Activité électrique	8
1.6 Rythme sinusal normal	9
1.7 Rythme sinusal irrégulier	10
1.8 Maladies du Coeur	10
1.8.1 Maladies vasculaires	10
1.8.2 Maladie coronarienne	10
1.8.3 Insuffisance cardiaque	11
1.8.4 L'angine de poitrine	11
1.8.5 L'infarctus du myocarde	11
1.8.6 Asthme cardiaque	11
1.8.7 L'arrêt cardiaque	11
1.9 Différence entre l'arrêt cardiaque et la crise cardiaque	11
1.10 Les causes de L'AC	12
1.11 Les symptômes de L'AC	12
1.12 Définition d'électrocardiogramme	13
1.13 Dissociation électromécanique	13
1.14 Réanimation cardiorespiratoire	13
1.15 Conclusion	14
<b>2 Apprentissage automatique et Apprentissage profond</b>	<b>15</b>
2.1 Introduction	15
2.2 Intelligence Artificielle	15
2.3 Data mining	16
2.4 Apprentissage automatique	17

2.4.1	Types d'apprentissage automatique . . . . .	18
2.5	Apprentissage profond . . . . .	19
2.6	Applications du l'apprentissage profond . . . . .	20
2.6.1	Différentes architectures de l'apprentissage profond . . . . .	20
2.7	Ouvrages connexes . . . . .	25
2.7.1	Forêt aléatoire . . . . .	25
2.7.2	Machine à vecteurs de support . . . . .	26
2.7.3	Bayes naïfs . . . . .	27
2.7.4	Travail de l'arbre de résolution . . . . .	27
2.8	Conclusion . . . . .	28
<b>3</b>	<b>Conception</b> . . . . .	<b>29</b>
3.1	Introduction . . . . .	29
3.2	Architecture globale du système . . . . .	29
3.3	Architecture détaillée . . . . .	30
3.3.1	Recueillir un ensemble de données sur les maladie cardiaques . . . . .	31
3.3.2	Prétraitement des données . . . . .	33
3.3.3	Apprentissage CNN . . . . .	35
3.3.4	Usage . . . . .	38
3.3.5	Fonction RELU . . . . .	39
3.3.6	Fonction SoftMax . . . . .	40
3.3.7	MaxPooling . . . . .	41
3.4	Conclusion . . . . .	41
<b>4</b>	<b>Implémentation</b> . . . . .	<b>42</b>
4.1	Introduction . . . . .	42
4.2	Préparation de l'environnement de mise en oeuvre . . . . .	42
4.3	Outils et langages de développement . . . . .	42
4.3.1	Python . . . . .	42
4.3.2	Anaconda Distribution . . . . .	43
4.3.3	Définition de tensorflow . . . . .	45
4.3.4	TensorFlow dans Anaconda . . . . .	45
4.3.5	Keras . . . . .	45
4.3.6	Modèle sous Keras . . . . .	46
4.3.7	Keras GPU vs CPU . . . . .	46
4.3.8	Matplotlib . . . . .	46
4.3.9	Pandas . . . . .	46
4.3.10	Sklearn . . . . .	47
4.4	Utilisation d'environnements virtuels dans Jupyter Notebook et Python . . . . .	47
4.4.1	Lancer jupyter . . . . .	47
4.4.2	Interface Jupyter . . . . .	48
4.5	Les bibliothèques . . . . .	50
4.6	Description des bibliothèques et modules utilisés . . . . .	50
4.6.1	Numpy . . . . .	50
4.6.2	Conv1D . . . . .	50
4.6.3	Flatten . . . . .	51
4.6.4	Dense . . . . .	51
4.6.5	Adam . . . . .	51

4.7	Structures de données . . . . .	51
4.7.1	Partie pf ensemble de données utilisé . . . . .	51
4.7.2	Pré-traitement des données . . . . .	51
4.7.3	Apprentissage des données . . . . .	53
4.7.4	Créer des formulaires . . . . .	54
4.7.5	Training . . . . .	54
4.7.6	Les paramètres d'apprentissage du réseau . . . . .	54
4.7.7	Expériences et résultats obtenus . . . . .	55
4.7.8	Test . . . . .	57
4.8	Résultats obtenus . . . . .	57
4.8.1	Présentation des performances obtenu . . . . .	57
4.8.2	Évaluation du modèle sur les données de test . . . . .	59
4.9	Conclusion . . . . .	59
	<b>Conclusion générale</b>	<b>61</b>

# Table des figures

1.1	Positionnement de coeur [19] . . . . .	6
1.2	Schéma générale de cœur[74] . . . . .	7
1.3	Phases électriques et mécaniques au cours d'un cycle cardiaque [42] . . . . .	8
1.4	Les événement électrique du cycle cardiaque [74] . . . . .	9
1.5	Rythme sinusal normal [23] . . . . .	9
1.6	Fibrillation ventriculaire [40] . . . . .	10
1.7	Tracé d'un RSP sur un ECG [40] . . . . .	13
1.8	Chaine de suivre [61] . . . . .	14
2.1	Différence entre l'apprentissage automatique et profond[7] . . . . .	16
2.2	Étapes du processus d'exploration de données [64] . . . . .	17
2.3	Processus d'apprentissage automatique [70] . . . . .	18
2.4	Différents types d'apprentissage automatique [27] . . . . .	19
2.5	Structure de l'apprentissage profond [12] . . . . .	20
2.6	Architecture de mémoire à long terme [66] . . . . .	21
2.7	Architecture de réseau de croyance profond [44] . . . . .	22
2.8	Architecture de autoencodeur [69] . . . . .	23
2.9	Architecture de codage automatique [81] . . . . .	23
2.10	Architecture de cryptage automatique variable [78] . . . . .	24
2.11	Architecture générale du réseau neuronal convolutif [35] . . . . .	25
2.12	Exemple d'algorithme de forêt aléatoire [41] . . . . .	26
2.13	Exemple d'algorithme de machine vectorielle de support [41] . . . . .	27
2.14	théorème de Bayes [71] . . . . .	27
3.1	Architecture générale du système . . . . .	30
3.2	Architecture détaillée du système . . . . .	31
3.3	Phase de prétraitement . . . . .	33
3.4	La Formule de gestion des valeurs manquantes [83] . . . . .	33
3.5	La Formule de gestion des valeurs erronées [83] . . . . .	34
3.6	La formule de Normalisation des données [83] . . . . .	34
3.7	Conception détaillée du réseau 1D-CNN . . . . .	37
3.8	Phase de modèle de l'apprentissage de 1D CNN . . . . .	38
3.9	Phase de test du modèle 1D CNN . . . . .	39
3.10	La formule de la fonction Relu [34] . . . . .	39
3.11	Un graph de l'application de la fonction d'activation Relu [36] . . . . .	40
3.12	La formule de la fonction Softmax [34] . . . . .	40

3.13	Un graph de l'application de la fonction d'activation Softmax [36]	41
3.14	Max Pooling [21]	41
4.1	logo python [63]	43
4.2	Installation de anaconda [1]	44
4.3	Fenêtre d'accueil d'ANACONDA NAVIGATOR [1]	44
4.4	Logo TensorFlow [63]	45
4.5	keras [62]	46
4.6	Logo jupyter [63]	47
4.7	Lancer jupyter [14]	48
4.8	Interface Jupyter[14]	48
4.9	Interface Jupyter [14]	49
4.10	Interface Jupyter [14]	49
4.11	cellules Jupyter [14]	50
4.12	Importer des bibliothèques et des modules	50
4.13	maladie	51
4.14	pas maladie	51
4.15	Partie du code	52
4.16	partie du code	53
4.17	Divisez les données	53
4.18	Modèle séquentiel	54
4.19	training Model 1D CNN	54
4.20	paramétrers du 1D CNN	55
4.21	modéliser l'entraînement avec des valeurs de précision et de perte	55
4.22	Notre modèle 1D CNN	56
4.23	résultat du Modèle forêt aléatoire	56
4.24	modèle fonctionne bien	57
4.25	La précision du modèle obtenue par 1D CNN	58
4.26	La perte du modèle obtenue par 1D CNN	58
4.27	Matrice de confusion	59

# Liste des tableaux

3.1	Ensemble de données sue les maladies cardiaques [71] . . . . .	32
4.1	Comparaison . . . . .	57

---

## Liste des abréviations

AC : Arrêt cardiaque.

VD : Ventricule droit.

VG : Ventricule gauche.

OG : Oreillette gauche.

OD : Oreillette droite.

SA : Noeud sino-auriculaire .

AV : Noeud auriculoventriculaire .

IA : Intelligence artificielle.

DL : Deep learning.

SA : Noeud sinusal.

IM : Infarctus du myocarde.

AV : Noeud auriculo-ventriculaire.

NS : Noeud sinoauriculaire.

BP : Back-propagation

FP : Forward-propagation

NN : Neural network.

FV :Fibrillation ventriculaire.

RSP : Rythme sans pouls.

KDD : Knowledge discovery of databases.

RNN : Recurrent neural network.

GRU : Gated recurrent unit.

VAE : Variational autoencoder.

CNN :Convolutional neural network.

---

MCV : Maladies cardiovasculaires .

LDL : Lipoprotéines de petite densité.

RCR : Réanimation cardiorespiratoire.

DEA : Défibrillateur externe automatisé.

ECG : Electrocardiogram.

ACR : Arrêt cardio-respiratoire .

SMUR : Service mobile d'urgence et de réanimation.

SAMU : Service d'aide médicale urgente .

LSTM :Long Short-Term Memory.

# Introduction générale

Nous soulignons les maladies cardiaques parce qu'elles sont l'une des principales causes de décès prématuré dans le monde. L'Organisation mondiale de la santé (OMS) a estimé que 17,9 millions de personnes sont mortes de maladies cardiaques dans le monde en 2016. Le cœur agit comme une double pompe auto-ajustable qui distribue le sang à tous les organes du corps et tout défaut conduit directement à la maladie cardiaque et à la mort. Les maladies cardiaques sont définies comme un groupe de conditions qui affectent le cœur en raison d'un mode de vie malsain, du tabagisme, de l'alcool et de l'ecstasy mangeant de la graisse, ce qui rend difficile de faire un diagnostic rapide. Certains facteurs clés nous aident à réduire le risque de maladie cardiaque car la prédiction des maladies cardiaques est très sensible et très dangereuse et l'utilisation de la technologie et de l'apprentissage profond est devenue nécessaire car elle nous aide à détecter les maladies cardiaques et les arrêts cardiaques à un stade précoce pour réduire la mortalité.

L'objectif de ce sujet est d'étudier le prototype d'un système intelligent de prédiction des maladies cardiaques utilisant le réseau neuronal de dérivation unidimensionnelle (1D CNN) pour aider les professionnels de la santé à trouver des traitements efficaces à faible coût.

## **Structure de travail du projet**

### **1. Le premier chapitre :**

Nous allons essayer de présenter le membre le plus important du corps humain, qui représente le cœur et ses éléments, composants, structure, comment il fonctionne et concepts liés aux vaisseaux sanguins, puis aborder les maladies qui peuvent affecter le cœur, dont la plus importante est l'arrêt cardiaque et ses causes et symptômes et comment intervenir pour les traiter.

### **2. Le deuxième chapitre :**

Fournir des concepts et des définitions de l'intelligence artificielle, de l'extraction de données, de l'apprentissage automatique, des types, de l'apprentissage profond, des types et des structures de chaque type .

### **3. Le troisième chapitre :**

Fournir l'environnement, les langages de programmation et les outils utilisés avant le programme utilisé, et la structure pour décrire le système de ce travail expliquer la base de données et concevoir le modèle du réseau neuronal contourner unidimensionnel et certains graphiques avec la mention d'un autre modèle et le comparer au réseau neuronal contourner unidimensionnel.

### **4. Le quatrième chapitre :**

Nous montrons l'environnement de travail, le langage de programmation et tous les outils utilisés dans ce projet, et nous fournissons quelques images de l'interface de l'application et de la base de données que nous avons utilisées avec une étude du réseau de neurones convolutionnels unidimensionnels et sa comparaison avec un autre modèle.

# Structure et fonction du cœur

## 1.1 Introduction

Le cœur humain est un organe qui pompe le sang dans tout le corps à travers les vaisseaux sanguins les circulations sanguines qui fournissent de l'oxygène aux tissus et éliminent le dioxyde de carbone et comme l'a dit le Dr Lawrence Phillips, cardiologue à Langone Medical " Si [le cœur] n'est pas en mesure de fournir du sang aux organes et aux tissus, ils mourront "[15]. Dans ce premier chapitre nous vous présenterons les définitions et les concepts du cœur et des vaisseaux sanguins qui nous montrent comment fonctionne le système cardiovasculaire. nous vous présenterons également les causes et les symptômes liés au cœur qui entraînent des maladies cardiaques qui provoquent des même des dommages psychologiques qui conduisent inévitablement à un arrêt cardiaque et à une mort subite.

## 1.2 Définition et rôle et du coeur

Le cœur est une pompe musculaire linéaire, valve, conique, taille du poing pesant 300 grammes chez les adultes [80], il y a 4 à 5 litres de sang en circulation répartis dans tout le corps, le cœur pompe environ 8000 litres de sang chaque jour pour répondre aux besoins du corps. Le cœur est situé au milieu de la poitrine entre les poumons et varie en hauteur de 14 à 16 cm de large et 6 cm de large. Le cœur se compose de deux couches de couche interne et de couche externe et la couche externe contient des tissus sérologiques fibreux contenant du tamur, des cellules épithéliales et du tissu conjonctif, et la couche interne se compose également d'endocarde, de cellules épithéliales et de tissu conjonctif comme la montre la figure 1.1 [39].

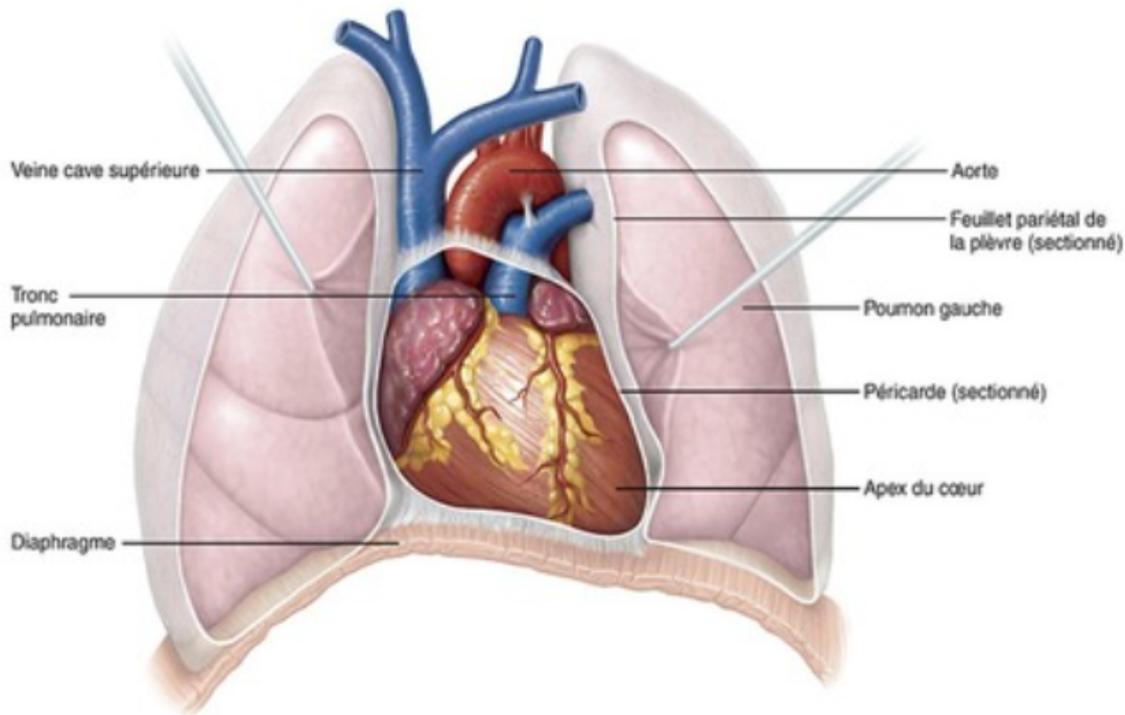


FIGURE 1.1 – Positionnement de coeur [19]

### 1.2.1 Ventricule droit

Le ventricule droit (VD) est l'une des deux grandes chambres inférieures du cœur d'une épaisseur d'environ 6 millimètres. Les parois sont constituées de trois couches musculaires disposées dans un vortex : la couche superficielle, la couche spirale profonde et la couche spirale gonflée déterminent un réseau complexe de bandes dans le ventricule droit [11] son rôle est de recevoir le sang de l'oreillette droite, qui à son tour l'envoie aux poumons par les artères [79].

### 1.2.2 Ventricule gauche

Le ventricule gauche (VG) reçoit du sang de l'atriculaire gauche qui pompez-le à travers l'artère aortique vers tous les tissus du corps, la surface du côté gauche du cœur est formée par le ventricule gauche et fait également partie de la surface inférieure et arrière. Le muscle cardiaque dans la paroi du ventricule gauche est plus épais. La barrière entre les ventricules apparaît de l'intérieur du ventricule gauche pour gonfler à Ventricule droit [79].

### 1.2.3 Oreillette gauche

L'oreillette gauche (OG) agit comme un canal et un réservoir pour le système circulatoire. Pendant la diastolique le flux des veines pulmonaires est dirigé vers le larynx et pendant la constriction ventriculaire, entrave l'écoulement jusqu'à ce que la valve mitrale s'ouvre [48]. Pour les parois molles de l'oreillette gauche sauf à l'intérieur de l'appendice de l'oreillette

gauche qui est rempli de vaisseaux sanguins, la paroi arrière est représentée par les veines pulmonaires supérieures et inférieures opposées [80].

### 1.2.4 Oreillette droite

L'oreillette droite (OD) est l'une des cavités supérieures du cœur qui reçoit tout le sang veineux pauvre en oxygène et riche en dioxyde de carbone apporté par la veine creuse, et un certain nombre de petits vaisseaux qui drainent les parois de la cavité elle-même. L'oreillette droite est légèrement plus grande que l'oreillette gauche, constituée des parois de l'oreillette droite, d'environ 3 millimètres d'épaisseur, à partir de deux couches de couche superficielle contenant l'oreillette et la couche interne, qui contient un certain nombre de petits faisceaux couvrant la cavité de l'oreille voir la figure 1.2 [10].

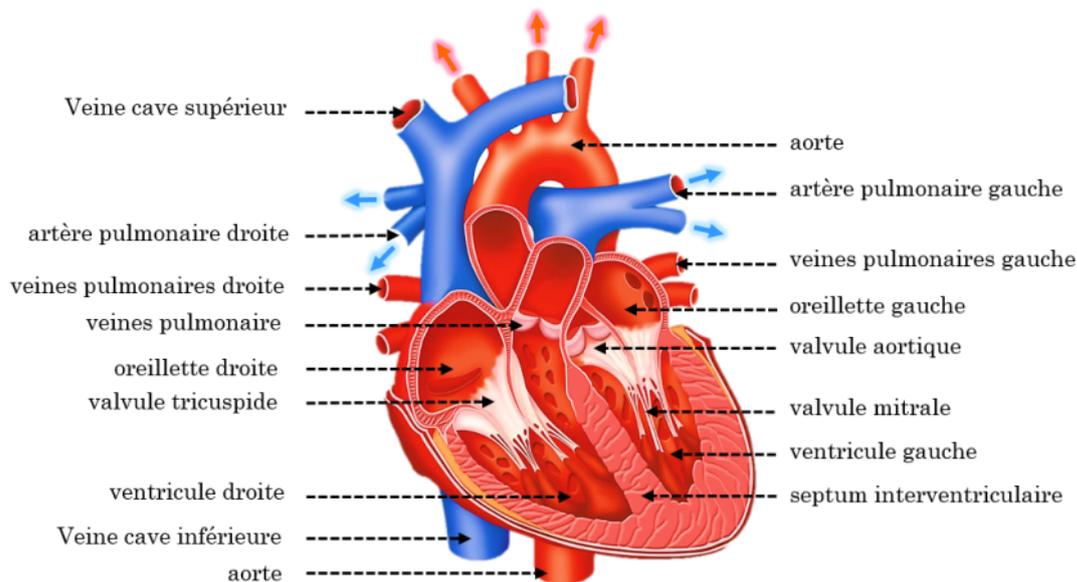


FIGURE 1.2 – Schéma générale de cœur[74]

## 1.3 Circulation sanguine

Le corps humain a deux voies de circulation : la circulation pulmonaire, qui est un cycle de basse pression, et la circulation systémique, qui est un cycle de haute pression. Les gros vaisseaux appelés artères transportent le sang du cœur vers les organes tandis que les veines renvoient le sang au cœur [42].

## 1.4 Cycle cardiaque

Le cœur est le siège du cycle cardiaque car il distribue le sang dans tout le corps et le cœur passe par deux étapes : le stade de contraction et le stade diastolique, les valves sont fermées afin que le sang puisse entrer dans l'atrium pendant la contraction de l'oreille

et les parois de l'écouteur sont contrôlées conduisant à l'ouverture des valves atriculaires ventriculaires puis à la pression artérielle dans l'atrior pour passer à travers les ventricules à travers les valves. Au fur et à mesure que la pression se poursuit, les valves semi-circulaires s'ouvrent et le sang atteint les artères qui le transportent vers les poumons. L'aorte transporte le sang vers toutes les parties du corps. Les valves semi-circulaires sont fermées avec du sang sortant des ventricules. Tout cela se produit pendant la diastolique et répète ensuite le même cycle lorsque les vaisseaux sont remplis de sang voir la figure 1.3[31].

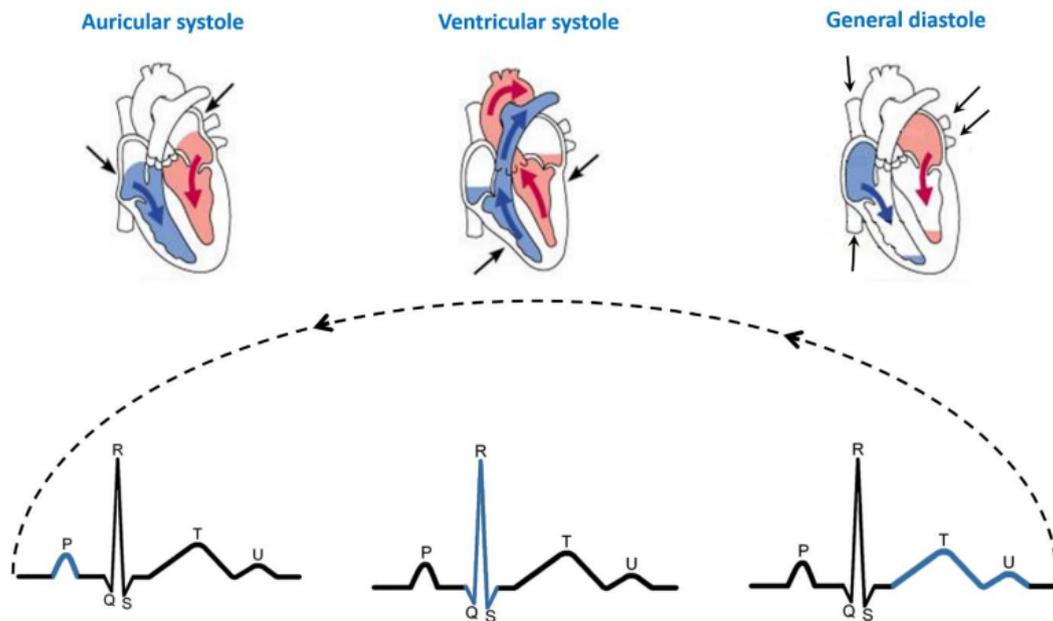


FIGURE 1.3 – Phases électriques et mécaniques au cours d'un cycle cardiaque [42]

## 1.5 Activité électrique

La contraction du muscle cardiaque est régulée par des impulsions électriques, générées par le noeud sinoauriculaire ou sinusales (NS). Ces impulsions provoquent l'excitation des oreillettes et des ventricules.

1. Le noeud SA est situé dans l'oreillette droite en entraînant leurs contractions. Cela permet le sang à pénétrer dans les ventricules.
2. Le noeud AV sépare les oreillettes et ventricules, et agit comme une barrière qui ralentit le signal électrique de pénétrer. Ce délai fourni à l'oreillette le temps de se contracter.
3. Réseau His-Purkinje : Cela est le cheminement des fibres du sang du cœur vers les poumons et le corps, ou l'augmentation de l'impulsion électrique dans les parois musculaires des ventricules.
4. Le graphique présente les événements de l'activité électrique d'une contraction normale du cœur, le noeud NS envoie une autre impulsion et le cycle est mis en évidence par un tracé ECG. généralement Ondes P amplifiée ou inversée indique souvent un

élargissement des oreillettes. L'infarctus du myocarde (IM) se produit lorsque le flux sanguin diminue ou s'arrête dans une partie du coeur voir la figure 1.4 [74].

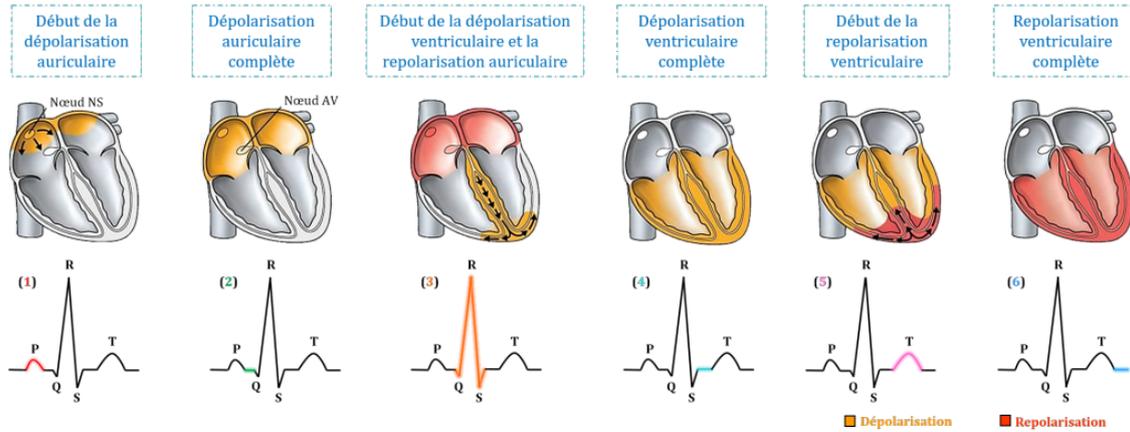


FIGURE 1.4 – Les événements électriques du cycle cardiaque [74]

## 1.6 Rythme sinusal normal

Le rythme normal du cœur correspond à l'activation physiologique de l'atrogie puis des ventricules dans le nœud sinusal. Le rythme des sinus a une fréquence cardiaque régulièrement normale comprise entre 60 et 80 / minute chez l'adulte au repos. L'ECG contient une série d'ondes P qui activent l'écouteur de haut en bas et de droite à gauche vers une onde P positive [26]. La fréquence cardiaque normale est contrôlée par entraînement spécifique dans l'oreille à droite se trouve le nœud sinusal et l'activité correcte du nœud sinusal est ajustée par les neurotransmetteurs émanant des systèmes de choline et d'adrénaline voir la figure 1.5 [58].

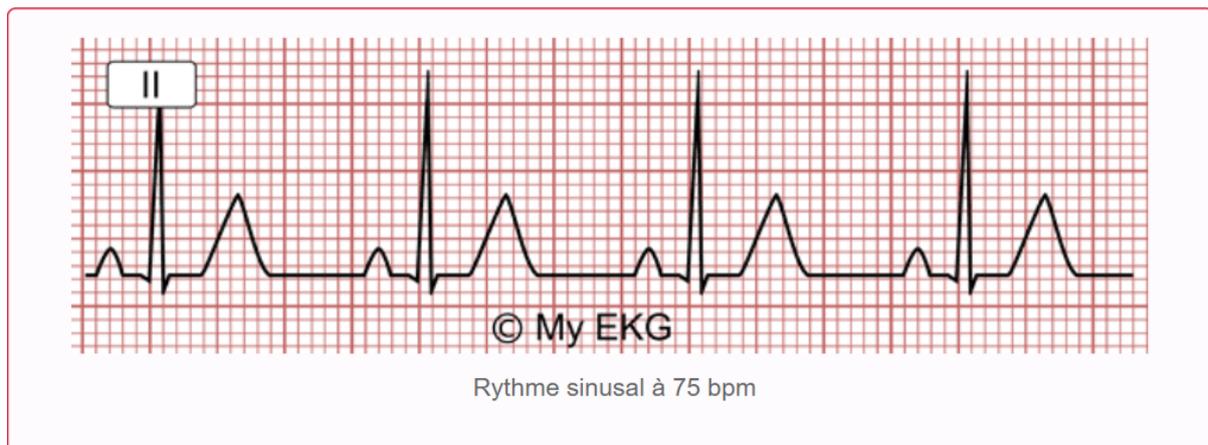


FIGURE 1.5 – Rythme sinusal normal [23]

## 1.7 Rythme sinusal irrégulier

L'arythmie est une anomalie ou un rythme irrégulier du cœur qui est causée par le corps qui ne pompe pas correctement le sang à la fréquence cardiaque normale du corps. Un patient souffrant d'arythmies ressent des symptômes tels que des battements de rythme cardiaque, des palpitations courtes, des évanouissements, des étourdissements, des douleurs thoraciques ou même l'absence totale de ces symptômes. Il existe trois types d'arythmies : la greffe cardiaque (lente) les arythmies ou les knock-ons [67] La tachycardie supraventriculaire, la fibrillation ventriculaire et le type le plus courant est la fibrillation ventriculaire car la fibrillation ventriculaire et la tachycardie peuvent être considérablement menacées par la vie humaine voir la figure 1.6 [4].

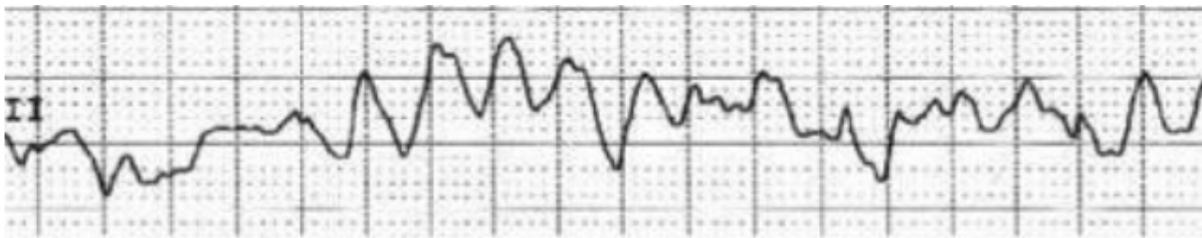


FIGURE 1.6 – Fibrillation ventriculaire [40]

## 1.8 Maladies du Coeur

Les maladies cardiaques entraînent un taux de mortalité important dans le monde entier il est devenu une menace pour la santé de nombreuses personnes [54], Les maladies cardiaques doivent être prédites car il s'agit d'une tâche complexe dans le domaine des sciences médicales nécessitant la nécessité de développer un système de détection des maladies cardiaques, [45] il y a beaucoup de maladies cardiovasculaires (MCV) qui affectent le cœur dont certaines que nous mentionnerons :

### 1.8.1 Maladies vasculaires

La cause de ces maladies est le durcissement des artères qui affecte d'autres vaisseaux ce qui entraîne une diminution du flux sanguin vers le cœur et l'affecte négativement [20].

### 1.8.2 Maladie coronarienne

Survient à la suite d'un blocage ou d'une sténose des artères coronaires [20] qui est l'accumulation de plaques dans les artères du cœur pendant une longue période en raison de taux élevés de cholestérol lipoprotéique (LDL) dans le plasma du patient ce qui entraîne une insuffisance cardiaque ou la mort [13].

### 1.8.3 Insuffisance cardiaque

Il existe deux principales causes d'insuffisance cardiaque à savoir l'hypertension artérielle et les crises cardiaques qui surviennent à la suite d'une insuffisance cardiaque. Le traitement immédiat de cette maladie a été diagnostiqué en prenant des médicaments qui aident le patient à vivre une vie normale et longue sans avoir besoin de rester à l'hôpital [20].

### 1.8.4 L'angine de poitrine

Se présente sous la forme d'un syndrome douloureux thoracique, Les symptômes de la douleur commencent après l'effort et surviennent à la suite d'un manque d'oxygène dans le cœur et donc d'un problème cardiaque [3].

### 1.8.5 L'infarctus du myocarde

Survient à la suite d'un manque de nutriments et d'oxygène au cœur en raison d'un blocage de l'artère coronaire et donc d'un défaut du muscle cardiaque et de son apport en oxygène [16].

### 1.8.6 Asthme cardiaque

Il survient à la suite d'un bronchospasme progressif où le patient ressent une respiration sifflante sévère lors de la respiration accompagnée d'essoufflement et de toux, et c'est l'une des maladies courantes et dangereuses de l'insuffisance cardiaque [72].

### 1.8.7 L'arrêt cardiaque

En 1962 l'arrêt cardiaque a été défini par Peter Safar, Henrik Paulsen, Leroy C Harris et Vladimir Negovsky lors d'un symposium international en Autriche comme « une condition dans laquelle le cœur est encore capable de travailler Mais il a cessé de battre, sans perdre tous ses emplois. » Au fur et à mesure que la technologie et les connaissances progressent cette définition a évolué et est devenue beaucoup de pertes à mesure que kt a été identifié par une interruption efficacité respiratoire et perte de conscience perte des principales impulsions de l'humérus de la cuisse et de l'artère carotide qui entraînent souvent la mort du patient [37].

## 1.9 Différence entre l'arrêt cardiaque et la crise cardiaque

L'arrêt cardiaque doit être distingué d'une crise cardiaque parce qu'une crise cardiaque est un manque de flux sanguin vers une partie du cœur et peut souvent conduire à un trouble électrique qui se termine par un arrêt cardiaque, mais l'arrêt cardiaque est une

perte soudaine de la fonction cardiaque, de la respiration et de la conscience conduit à un dysfonctionnement dans le travail de pompe cardiaque [25].

## 1.10 Les causes de L'AC

Les maladies cardiaques une condition à laquelle les médecins sont souvent confrontés à l'hôpital, y compris le service des urgences et l'unité de soins intensifs, où il y a de nombreux décès dus à la fibrillation ventriculaire et à la tachycardie ventriculaire [50] ce sont des maladies multifactorielles qui peuvent causer plusieurs facteurs [43].

- Maladie cardiaque génétique.
- Cardiopathie congénitale.
- Myocardite aiguë (myocardite) et surdosage.
- Hypoxie, qui se produit à la suite d'une forte baisse des niveaux d'oxygène .
- Hypertension artérielle [73].
- Alcool et tabagisme [49].
- Taux de cholestérol élevé.
- Antécédents familiaux de maladie cardiaque.
- Arythmie [72].
- Les rythmes cardiaques anormaux potentiellement mortels appelés fibrillation ventriculaire (FV) sont également parmi les causes les plus courantes où la FV se produit lorsque l'activité électrique du cœur devient volatile jusqu'à ce que le cœur cesse de pomper et de trembler [5].

## 1.11 Les symptômes de L'AC

Le cœur dispose d'un système électrique spécial qui pompe le sang dans tout le corps et fournit de l'oxygène et des nutriments essentiels aux cellules. Lorsqu'il y a un problème avec ce système il perturbe le pompage cardiaque et de nombreux symptômes surviennent à la suite de ce déséquilibre : [6]

- Perte de conscience.
- Absence de pouls.
- Manque de respiration.
- Effondrement soudain.
- Essoufflement.
- Stimulateur.
- Inconfort thoracique. [25].

## 1.12 Définition d'électrocardiogramme

L'électrocardiogramme (ECG) est une technique de maladie cardiaque bien établie pour analyser l'état cardiaque des patients, également connue sous le nom d'électrosynthèse de l'activité systolique du cœur, et cette activité est facilement enregistrée par des électrodes superficielles sur les membres ou la poitrine du patient. La fréquence cardiaque en termes de pouls par minute (bpm) peut être facilement calculée en calculant les pics R de l'onde ECG dans la minute suivant l'enregistrement. Les cardiologues peuvent maintenant l'utiliser pour diagnostiquer les arythmies et autres maladies en regardant la version imprimée des formes d'onde ECG [52].

## 1.13 Dissociation électromécanique

La Dissociation électromécanique ou activité électrique sans pouls en cas d'arrêt cardiaque, l'activité électrique du cœur se poursuit pendant un certain temps et le cœur reprend une activité électrique normale, mais son pouls ne permet pas une circulation efficace voir la figure 1.7[40].



FIGURE 1.7 – Tracé d'un RSP sur un ECG [40]

## 1.14 Réanimation cardiorespiratoire

Chaîne de survie Cette série a été introduite par Cummins en 1991 pour améliorer les chances de survie, une fois diagnostiquée l'objectif de la réanimation est de rétablir l'efficacité cardiaque le plus rapidement possible. Cela nécessite de passer un protocole bien écrit autour des quatre maillons de la « chaîne de survie » :

- RCR par témoins.
- défibrillation précoce réalisée par des secouristes ou des passants à l'aide d'un défibrillateur externe automatisé (DEA).
- Réanimation médicale par une équipe du Service médical d'urgence (SAMU) ou du Service de réanimation médicale d'urgence (SMUR).

Ces équipes médicales sont représentées par SAMU ou SMUR ces dernières années de grand développement afin de les rendre idéales car elles constituent une étape importante dans le système. La RCR précoce et la défibrillation subséquente peuvent être compensées par les liens suivants dans le système voir la figure 1.8.



FIGURE 1.8 – Chaîne de suivre [61]

Reconnaissance de L'arrêt cardiorespiratoire et alerte (ACR) est définie comme suit : Perte de conscience et absence de réponse du corps, essoufflement, la dernière chose est de ne pas enregistrer le pouls et le pouls périphérique disparaît lorsque la pression artérielle systolique est d'au moins 40 mmHg et que la palpation se produit dans un maximum de 10 secondes. Lorsque le système respiratoire cesse de fonctionner, trois étapes importantes doivent être prises :

- Dévisser ou ouvrir tout ce qui peut gêner la ventilation du patient (col de chemise, cravate).
- Assurez-vous qu'il n'y a pas de corps étrangers dans la bouche du patient lors de l'ouverture.
- Analyse de la détresse respiratoire pendant 10 secondes [61].

## 1.15 Conclusion

Dans ce chapitre, nous avons expliqué en détail l'anatomie du cœur et diverses maladies cardiaques qui peuvent affecter le cœur qui ont été étudiées grâce à l'enregistrement de Holter, et dans le deuxième chapitre, nous concentrerons sur les concepts scientifiques et technologiques les plus importants qui nous aident à étudier les données des patients cardiaques et à réduire l'incidence de ces maladies autant que possible.

# Apprentissage automatique et Apprentissage profond

## 2.1 Introduction

L'apprentissage profond est un sous-ensemble de l'apprentissage automatique qui a obtenu des résultats remarquables dans de nombreuses images médicales et tâches de classification [77] où l'apprentissage profond a commencé à affecter la recherche biologique et les applications biomédicales en raison de sa capacité à intégrer des ensembles de données complets capables de prédire les maladies cardiaques, le diabète et d'autres maladies [76]. Ce deuxième chapitre est organisé sous la forme suivante la première section définit les concepts d'intelligence artificielle et d'exploration de données et ses étapes, la section suivante contient la définition de l'apprentissage automatique et de l'apprentissage profond connaissant leurs types respectives. Quant à la dernière section, il s'agit de réseaux de neurones profonds.

## 2.2 Intelligence Artificielle

Il existe de nombreux concepts qui expriment l'intelligence artificielle dont deux que nous mentionnerons : Décrire intelligence artificielle comme l'apprentissage automatique est lié à l'intelligence artificielle est un domaine de l'informatique qui comprend la création de machines capables de travailler comme des humains et de voir le monde comme nous le faisons ,l'apprentissage automatique est également défini comme un état de l'intelligence artificielle basé sur le développement de logiciels à travers un ensemble de données complexes pour produire des prédictions intellectuelles sans intervention humaine [57]. Le terme «intelligence artificielle» est utilisé pour décrire les programmes informatiques qui imitent et imitent l'intelligence humaine tels que la résolution de problèmes et l'apprentissage. L'intelligence artificielle peut se référer à l'intelligence artificielle à usage général, où le système est autosuffisant et a une perception similaire de la perception humaine. L'apprentissage automatique est un sous-domaine de l'intelligence artificielle qui détecte les modèles de données sans instructions explicites. L'apprentissage automatique s'adapte à la plupart des techniques d'IA dans la préparation de la recherche médicale. Dans l'ap-

apprentissage automatique, la machine apprend à partir de données et effectue des tâches basées sur le modèle d'apprentissage voir la figure 2.1 [53].

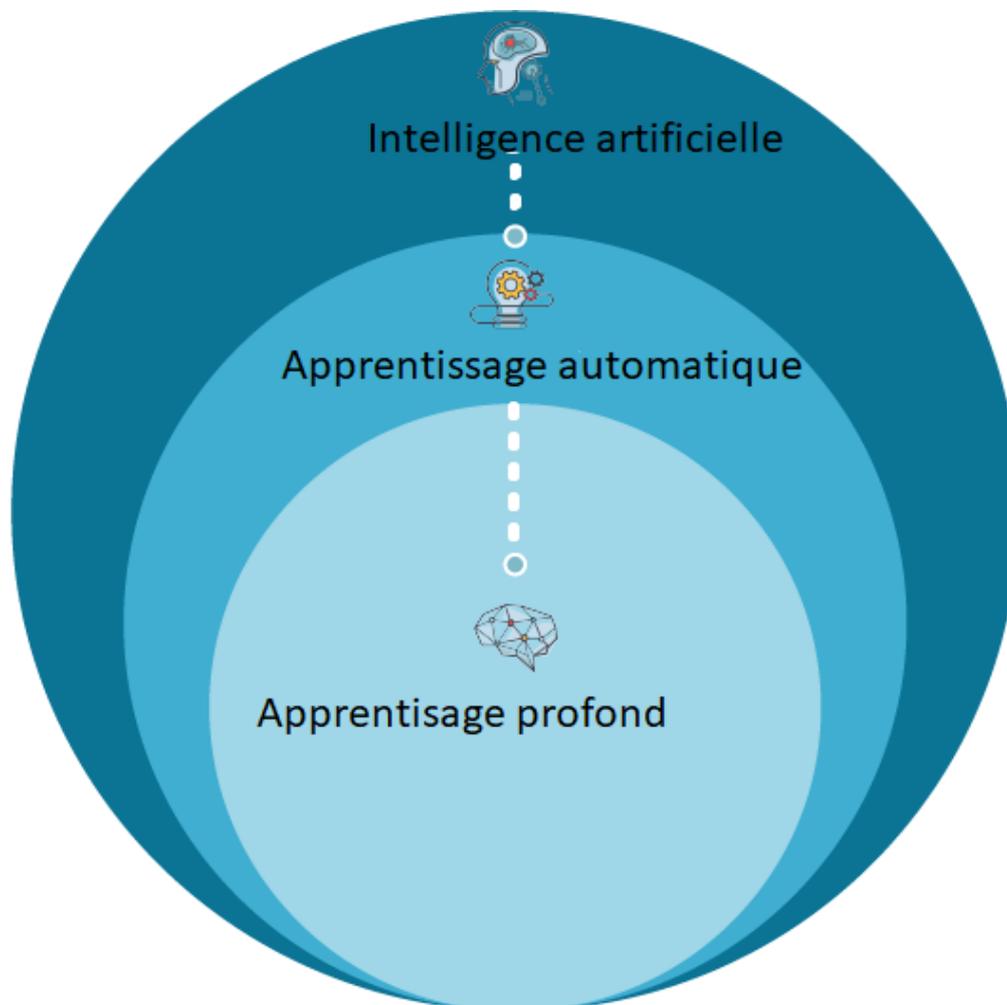


FIGURE 2.1 – Différence entre l'apprentissage automatique et profond[7]

## 2.3 Data mining

(Data mining) dans des domaines importants pour la recherche est utilisée dans plusieurs domaines différents tels que le financement, la recherche clinique l'éducation ,les soins de santé etc. La portée de l'extraction des données de santé a été examinée et étudiée par plusieurs chercheurs. Une approche analytique du caractère unique des données médicales dans les soins de santé est également fournie [65] .Data mining est un processus qui fournit un concept pour attirer l'attention des utilisateurs en raison de l'énorme quantité de données et de la nécessité de convertir de grandes quantités de données en informations utiles. Beaucoup de gens utilisent le terme « dispositif de découverte des connaissances » (KDD) à des fins d'exploration de données,Les connaissances sont extraites ou découvertes en sept étapes séquentielles utilisées pour extraire des données :

1. Effacement des données : nous supprimons les données qui contiennent du bruit et des données non pertinentes.

2. Intégration de données : Nous combinons de nombreuses données dans un magasin de données appelé données cibles.
3. Sélection des données : données extraites de la base de données telles qu'elles ont été traitées précédemment.
4. Conversion des données : À ce stade, les données sont intégrées dans Formats d'exploration standard appropriés en compilant et en résumant les processus.
5. Exploration de données : À ce stade, les méthodes s'appliquent pour récupérer le formulaire de données ou les règles.
6. Évaluation du modèle ici nous choisissons l'arbre avec précision.
7. Représentation des connaissances : dernière étape de la visualisation utilisée pour aider les utilisateurs à comprendre ou à connaître le résultat de l'extraction de données voir la figure 2.2 [70].

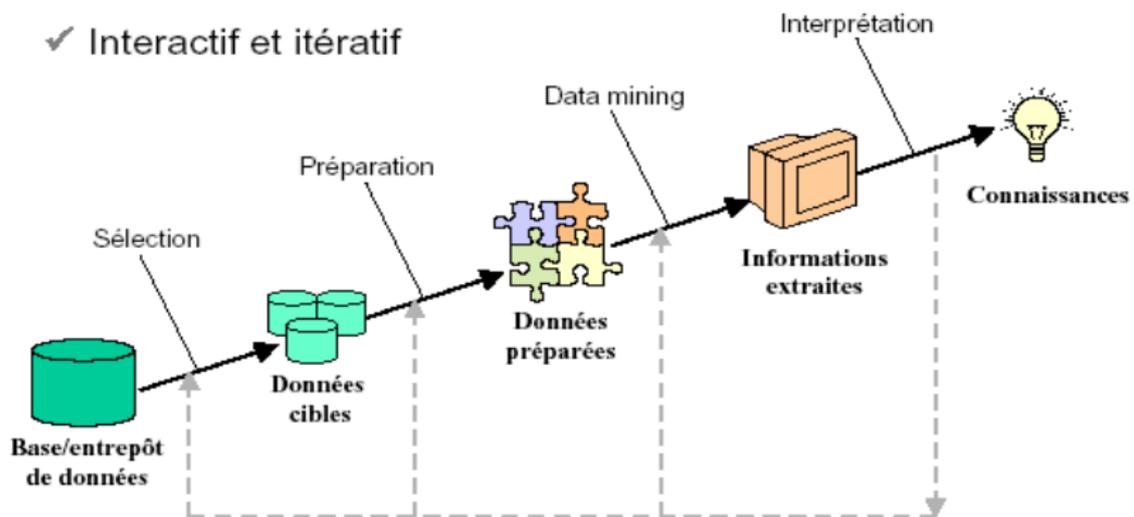


FIGURE 2.2 – Étapes du processus d'exploration de données [64]

## 2.4 Apprentissage automatique

Il existe plusieurs définitions de l'apprentissage automatique (data mining) y compris un algorithme d'apprentissage automatique qui utilise des données pour accomplir la tâche requise qui est en quelque sorte une « programmation cryptée » ce qui signifie qu'il modifie ou adapte automatiquement sa structure par répétition. Le processus de modification est appelé formation où des échantillons de données d'entrée sont fournis avec les résultats souhaités. L'algorithme peut contenir des paramètres numériques variables qui sont ajustés par une optimisation fréquente et peut également avoir un réseau de voies de calcul potentielles qu'il organise pour obtenir des résultats et les utiliser pour prédire les résultats [46].

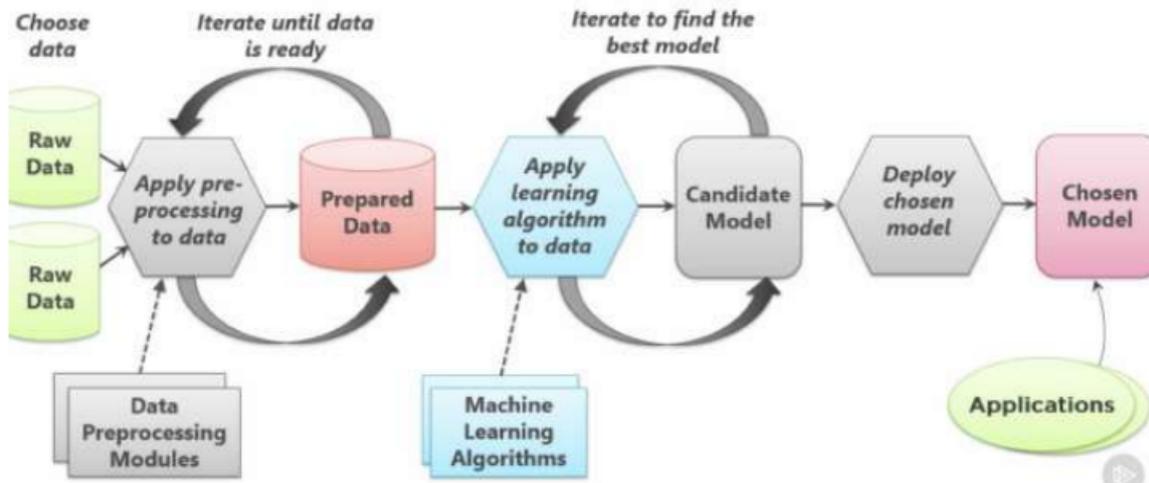


FIGURE 2.3 – Processus d'apprentissage automatique [70]

### 2.4.1 Types d'apprentissage automatique

« L'apprentissage automatique est le domaine d'étude qui donne aux ordinateurs la capacité d'apprendre sans être explicitement programmé » ( Arthur Samuel, 1959). Nous avons quatre types d'apprentissage automatique dont le plus important est :

1. Dans l'apprentissage supervisé il existe une classification dans chaque exemple de l'ensemble de données, qui peut être définie comme suit :  $(x_i, y_i)$   $x_i$  : Est le vecteur des paramètres de l'ensemble de données.  $y_i$  : Cette étiquette peut être un nombre, un vecteur ou une autre structure complexe. Le but de cet algorithme est de construire un modèle d'apprentissage prend en entrée un ensemble d'exemples et les sorties  $X$  sont déduites et comparées aux étiquettes  $Y$ . Les algorithmes les plus couramment utilisés dans l'apprentissage supervisé :
  - Réseaux de neurones.
  - Régression logistique.
  - K-voisins les plus proches.
  - Régression linéaire.
  - Prise en charge de la machine à vecteurs de travail.
  - Forêt aléatoire.
2. L'algorithme non supervisé prend le vecteur de fonction  $X$  comme entrée et produit un vecteur de sortie. Les algorithmes les plus couramment utilisés dans l'apprentissage non supervisé sont :
  - Algorithmes d'assemblage.
  - Algorithmes pour réduire les dimensions et la visualisation.
  - Algorithmes qui enseignent les règles de lien.
3. Apprentissage semi-supervisé : le but de cet algorithme est le même le rôle de données non classifiées améliore l'algorithme d'apprentissage pour produire un bon

modèle, Il existe deux ensembles de données classifiées et non classifiées qui sont confidentielles.

4. Apprentissage par renforcement : Cet algorithme surveille l'environnement en prenant l'état de l'environnement comme vecteur distinctif, Il est également permis d'effectuer des procédures dans chaque État. L'algorithme apprend par lui-même la meilleure politique pour maximiser la récompense comme la montre la figure 2.4 [38].

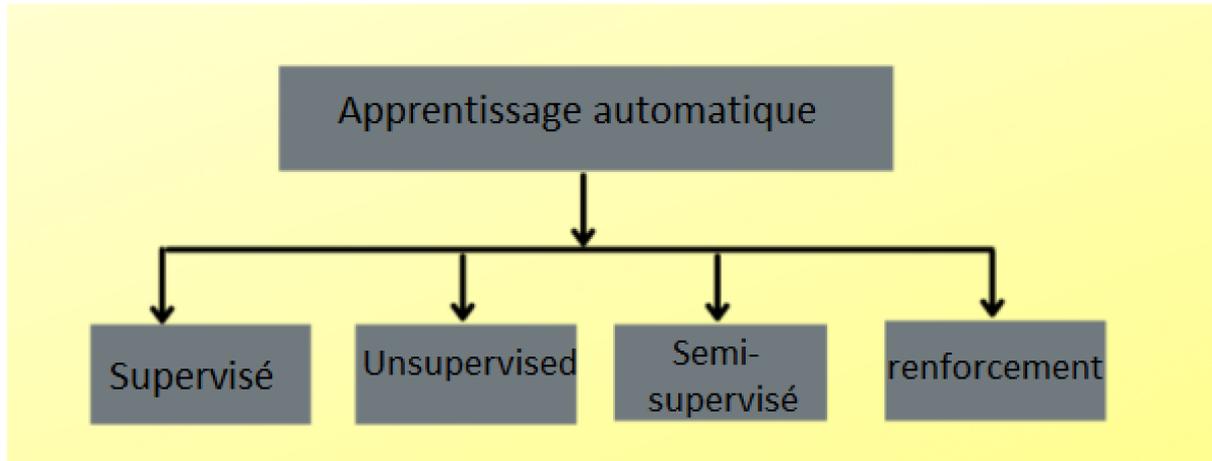


FIGURE 2.4 – Différents types d'apprentissage automatique [27]

## 2.5 Apprentissage profond

Apprentissage profond de l'intelligence artificielle et du type d'apprentissage automatique (ML) combine l'informatique, les statistiques et la théorie de la décision pour trouver des modèles dans des données complexes et souvent volumineuses. L'apprentissage profond est couramment utilisé dans les réseaux de neurones multicouches où il a déjà été démontré que l'apprentissage surpasse les experts et autres [56]. Je sais aussi que c'est une approche pour apprendre une fonctionnalité de haut niveau (caractéristiques) de densité de niveau de pixel brut qui est suffisamment utile pour distinguer les différents objets par un classifieur. L'apprentissage profond a démontré de grandes réalisations en matière de vision et d'apprentissage depuis que hinton et al. ont été proposés pour la première fois en tant que réseau de cryptage automatisé en profondeur en 1992 [81]. Les stratégies d'apprentissage automatique apparaissent dans des domaines aussi divers que la reconnaissance vocale, la classification des images et les jeux, et devraient. Ce DL peut également perturber la prise de décision clinique en intégrant des flux de données complexes, en tirant des conclusions intelligentes et éventuellement en personnalisant le traitement [56].

Stratégies d'apprentissage automatique dans des domaines aussi divers que la reconnaissance vocale, la classification d'images et les jeux, on s'attend à ce que ce DL peut également perturber la prise de décision clinique en intégrant des flux de données complexes, en tirant des conclusions intelligentes et en allouant éventuellement le traitement [56].

## 2.6 Applications du l'apprentissage profond

Les modèles NN sont utilisés dans le traitement numérique de l'image depuis de nombreuses années, mais ont été créés à l'origine par des machines Boltzmann et des dispositifs de cryptage automatique. Les technologies RNN montrent des performances améliorées dans de nombreux domaines tels que la reconnaissance de la langue de l'écriture manuscrite, la traduction, la modélisation et la modélisation de la parole vocale. Cette mémoire à court terme (LSTM) comprend une unité à parois fréquentes (GRU) et d'autres différences dans les réseaux neuronaux récurrents [60].

### 2.6.1 Différentes architectures de l'apprentissage profond

L'apprentissage profond est une forme avancée d'apprentissage automatique qui peut mener des processus complexes de données brutes sans aucune intervention humaine et peut le faire dans un immense espace de données comme le montre le laboratoire DeepMind de Microsoft en Californie, le premier laboratoire commercial d'apprentissage profond au monde .Il existe un certain nombre de structures et d'algorithmes différents utilisés dans l'apprentissage profond Au cours des dernières années, il y a eu six structures d'apprentissage profond voir la figure 2.5 [47]

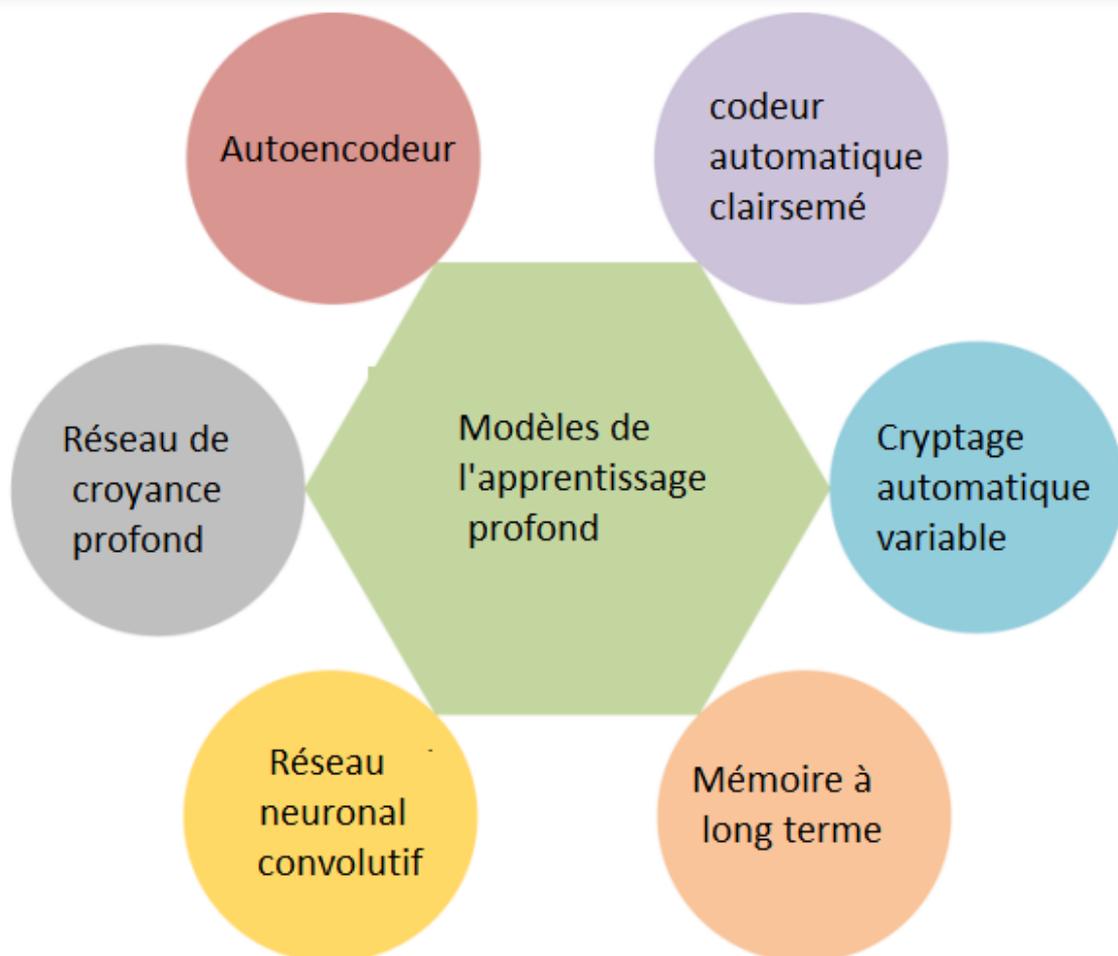


FIGURE 2.5 – Structure de l'apprentissage profond [12]

1. Mémoire à long terme (LSTM) : est un type de RNN utilisé pour résoudre le problème de l'oubli des dépendances à long terme dans les RNN chaque nœud du réseau contenant une cellule pour stocker ces dépendances, les différentes applications de LSTM sont :

- Prédire les chaînes temporelles
- Reconnaissance de l'écriture manuscrite
- Prédiction des stocks
- Prédiction de texte [30].

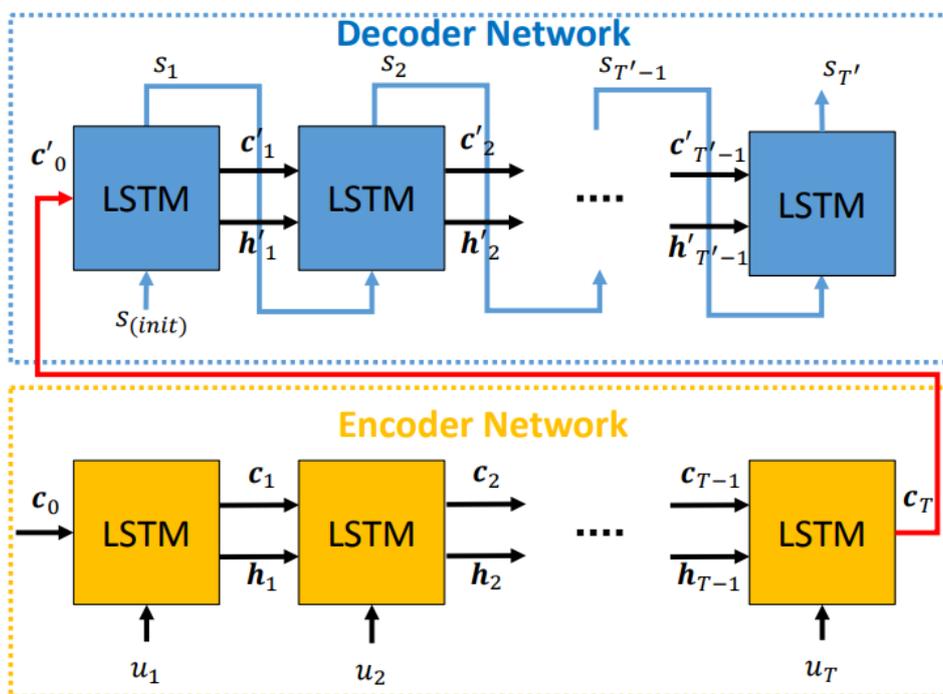


FIGURE 2.6 – Architecture de mémoire à long terme [66]

2. Réseau de croyance profond : les réseaux de croyance profond se composent de plusieurs couches de variables aléatoires et latentes. Il peut être considéré comme une forme spéciale du modèle obstétrique potentiel Bayésien. Par rapport aux réseaux de neurones artificiels, les DPN sont plus efficaces. L'amélioration des performances des DPN peut être largement attribuée au fait que les poids initiaux du réseau par rapport aux poids initialisés de manière aléatoire, ces poids sont plus proches de l'optimum global et peuvent donc apporter une meilleure performance [51]. L'objectif ultime est de créer une procédure de formation plus rapide et non supervisée basée sur la différence contradictoire de chaque sous-réseau comme la montre la figure 2.7 [9].

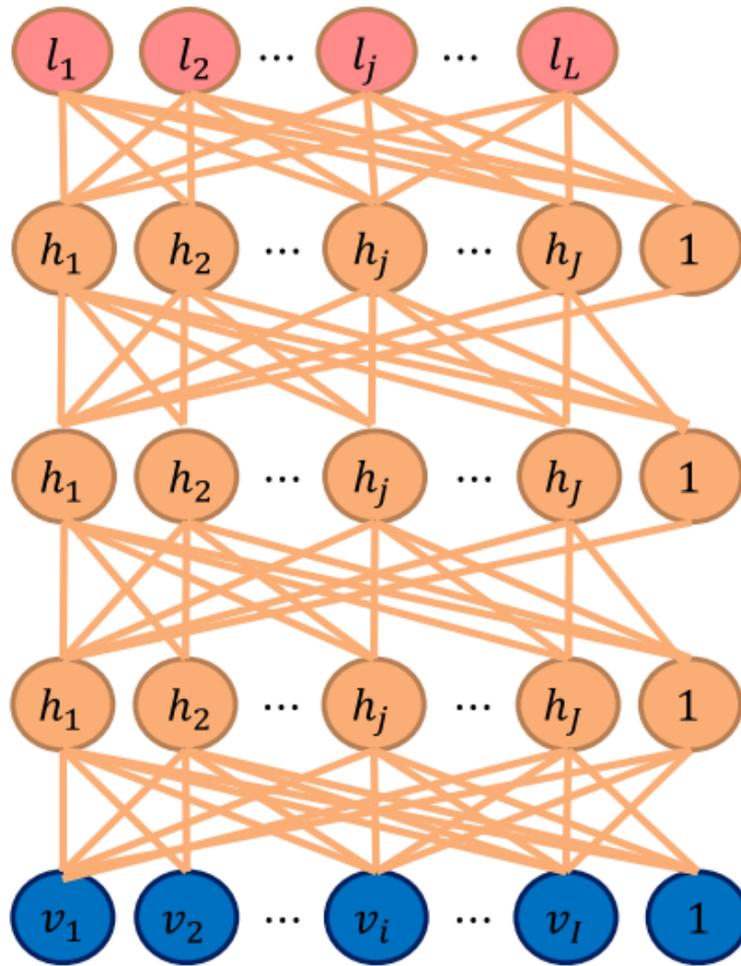


FIGURE 2.7 – Architecture de réseau de croyance profond [44]

3. Autoencodeur : les autoencodeurs sont des architectures DL pour l'apprentissage non supervisé. Ils ont été introduits pour la première fois en 1986 par Geoffrey Hinton. Ils sont principalement utilisés pour la compression de données et l'abstraction utile de données. Il existe des codeurs automatiques génératifs ou variationnels qui peuvent apprendre le modèle latent des données pour générer de nouvelles données. Autoencoder se compose d'un réseau d'encodeur et de décodeur. Pendant l'entraînement, des codeurs et des décodeurs sont utilisés, mais lorsqu'ils sont appliqués à de nouvelles données pour la réduction de la dimension, seul l'encodeur est utilisé. AE fonctionne dans des couches de neurones et est entraîné à l'aide de la rétropropagation comme la montre la figure 2.8[69].

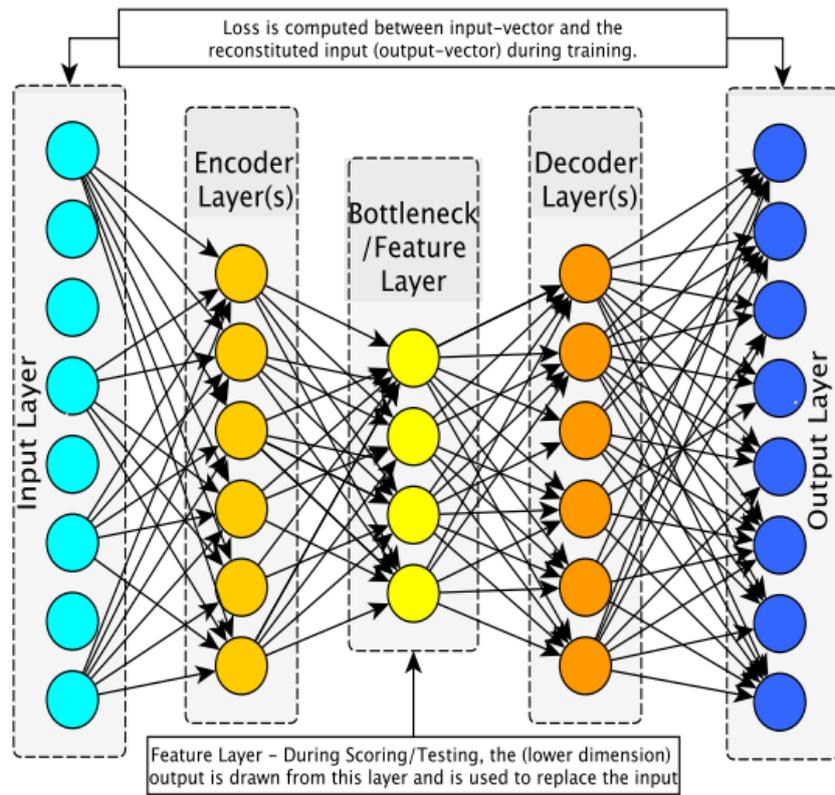


FIGURE 2.8 – Architecture de autoencodeur [69]

4. Codeur automatique clairsemé :est une partie importante des modèles d'apprentissage non utilisés pour développer l'apprentissage profond. Le codage automatique vise à compresser les représentations et à maintenir les informations de base pour reconstruire les données d'entrée alors qu'il est souvent utilisé pour apprendre des tâches ou pour réduire les dimensions comme la montre la figure 2.9[28].

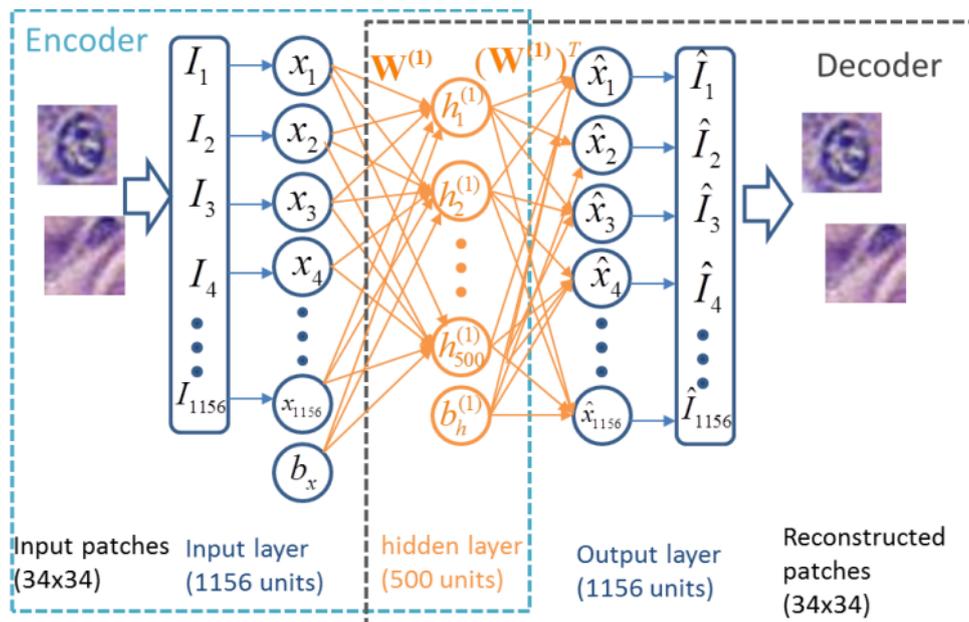


FIGURE 2.9 – Architecture de codage automatique [81]

5. Cryptage automatique variable :est un outil puissant, un moyen d'apprendre des représentations de données de haute dimension [59]. C'est aussi un modèle génératif de l'espace latent profond qui a eu beaucoup de succès dans de nombreuses applications telles que la génération d'images, les commentaires d'images, la conception de protéines et les modèles de langage. Les VAE apprennent à distribuer les données d'une manière qui peut générer de nouvelles données significatives comme la montre la figure 2.10 [78]

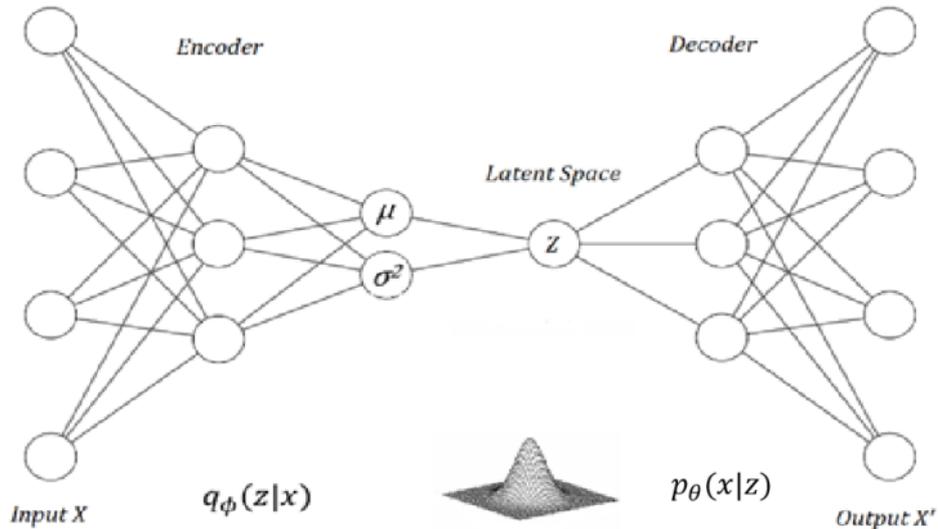


FIGURE 2.10 – Architecture de cryptage automatique variable [78]

6. Réseau neuronal convolutif (CNN) :le succès secret de CNN est de simuler le mécanisme des cellules cérébrales qui supprime les données en fonction de ses informations par exemple l'image peut être classée en fonction des valeurs de pixels de l'ingénierie de la forme et des bords etc. L'organe humain prend une décision appropriée en fonction des informations fournies par les neurones [47]. Le Neuro-Bypass Network (CNN) est un type de réseau neuronal synthétique profond également appelé CNN ou ConvNe qui est principalement appliqué dans l'analyse de données visuelles. Il est largement utilisé dans l'apprentissage en particulier dans les algorithmes d'apprentissage profond son travail est basé sur le processus de contournement mathématique. La structure de CNN se compose d'une série de couches y compris la couche de dérivation. La couche de normalisation, la couche d'assemblage, la couche entièrement connectée sont empilées ensemble pour former un modèle CNN complet.Cette forme est un exemple de streaming CNN LeNet pour traiter et catégoriser l'image d'entrée nombres l'entrée prend une image  $1 \times 28 \times 28$  de l'ensemble de données MNIST 1.Chez CNN il y a plus de couches entrecoupées de ces quatre couches de base :

- Couche entièrement connectée.
- Couche d'assemblage.
- Tournez l'emballage.

- Fonctions d'activation comme la montre la figure 2.11 [33].

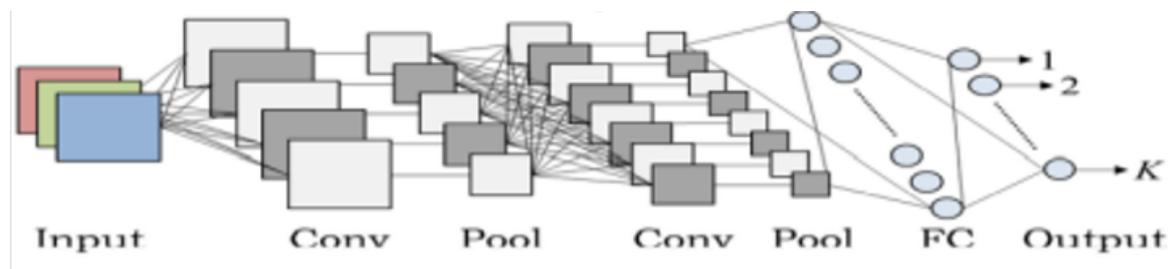


FIGURE 2.11 – Architecture générale du réseau neuronal convolutif [35]

## 2.7 Ouvrages connexes

Dans cette partie, nous discuterons de certaines des œuvres précédentes qui ont utilisé beaucoup de profondeur et des méthodes d'apprentissage automatique pour prédire les maladies cardiaques .

### 2.7.1 Forêt aléatoire

est un algorithme flexible utilisé dans la machine l'apprentissage, fréquemment utilisé en raison de sa simplicité et de la avantage de pouvoir utiliser en régression et classification. La forêt est l'ensemble de plusieurs arbre de décision en ajoutant un caractère aléatoire supplémentaire au modèle pendant la croissance des arbres et en prenant un sous-ensemble aléatoire de les caractéristiques sont prises en considération. Forêt aléatoire recherche la meilleure fonctionnalité parmi le sous-ensemble aléatoire résultant en une grande diversité de résultats, c'est-à-dire un meilleur modèle [41].

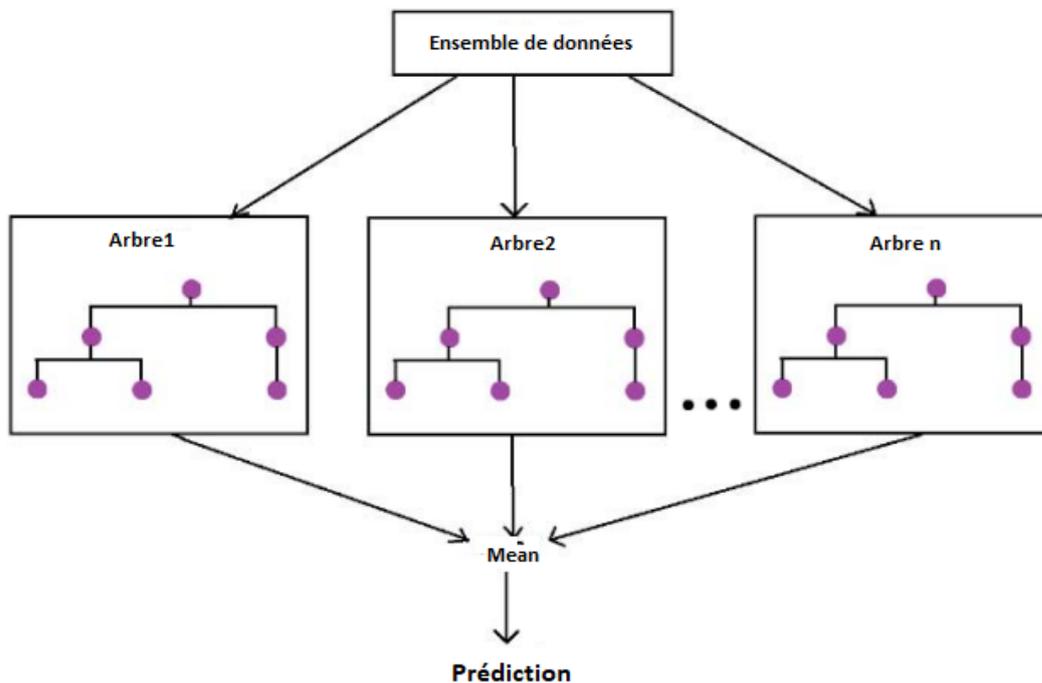


FIGURE 2.12 – Exemple d’algorithme de forêt aléatoire [41]

## 2.7.2 Machine à vecteurs de support

est un algorithme de classification. C’est un algorithme d’apprentissage supervisé. Il est préférable lorsque les données sont de grande dimension. L’une des principales raisons d’utiliser l’algorithme était à cause des noyaux. Pendant que implémentation de SVM, différentes fonctions du noyau peuvent être spécifié pour la fonction de décision. Les noyaux courants sont fourni, mais il est également possible de spécifier des noyaux personnalisés, ce qui le rend plus polyvalent. Dans SVM, les données sont mappées à un espace de dimension élevée qui catégorise les données même si les données sont linéairement inséparables. Un séparateur entre le catégories est trouvé après quoi l’hyperplan est dessiné en tenant compte du séparateur. Après cela, lorsque le prochain ensemble de l’entrée est alimentée l’algorithme peut prédire dans quelle classe les entrées mentiront. Les vecteurs de support sont des points d’observation alors que la SVM est un front qui différencie les deux classes [41].

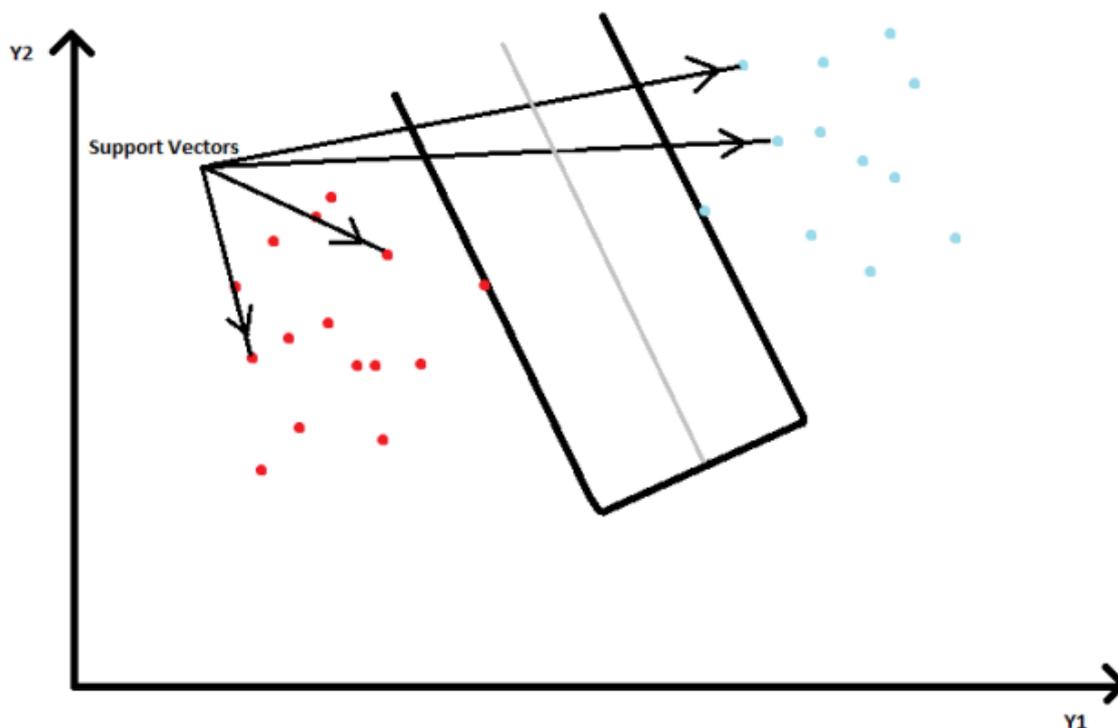


FIGURE 2.13 – Exemple d’algorithme de machine vectorielle de support [41]

### 2.7.3 Bayes naïfs

Basé sur le théorème de Bayes il examinera : la probabilité que les points de données soient classés voir la figure 2.12.

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

FIGURE 2.14 – théorème de Bayes [71]

### 2.7.4 Travail de l’arbre de résolution

Est un algorithme et est principalement utilisé pour résoudre problème de classification et problèmes de régression. dans arbre de décision, les exemples de faits seront divisés en deux parties ou plus groupes homogènes ou sous-ensembles avec différenciation dans entrée.

## 2.8 Conclusion

Dans ce chapitre nous avons expliqué l'intelligence artificielle et son rôle dans le domaine médical, après quoi nous avons essayé d'expliquer le concept principal de l'apprentissage automatique, de l'apprentissage profond et des algorithmes de chacun d'eux, où nous avons commencé par quelques définitions, prédictions, type d'apprentissage et nous avons donné quelques illustrations et équations.

## Conception

### 3.1 Introduction

Comme nous l'avons vu dans les chapitres précédents, ces années sont témoins d'un développement remarquable dans ce domaine combinant de grandes technologies de pointe qui sont DM, DL et technologies portables dans le secteur de la santé, car nous avons noté que plus de recherche ces dernières années dépendent des maladies cardiaques Basé sur les données d'une personne c'est-à-dire l'âge, le poids, le cholestérol, la consommation de glucose, l'activité physique, les antécédents de soins de santé et le mode de vie du patient. Dans ce chapitre, puisque notre choix est CNN comme étude de cas, nous présenterons notre vision conceptuelle et comment nous pouvons réaliser ce système c'est une proposition qui peut aider à prédire l'arrêt cardiaque précoce, qui dépend également des processus DL. Nous présenterons la conception globale du système 1D de CNN, puis passerons à sa conception détaillée, à ses éléments et à son travail, et fournirons quelques détails sur les ensembles de données que nous utilisons dans cette étude et sur la manière de les étudier.

### 3.2 Architecture globale du système

En général, nous pouvons représenter la structure du système pour la prédiction précoce de l'arrêt cardiaque et les étapes nécessaires comme suit : Nous utilisons un ensemble de données générales de kaggle liées aux maladies cardiaques, et nous pré-traitons les données. Nous pouvons décrire la phase de pré-traitement des données comme la phase d'extraction des données principalement décrit la conversion des données, nous devons donc rendre les données logiques. Ensuite divisez-le en deux parties l'une pour la formation et l'autre pour les tests afin d'être prêt à appliquer le modèle d'apprentissage profond proposé.

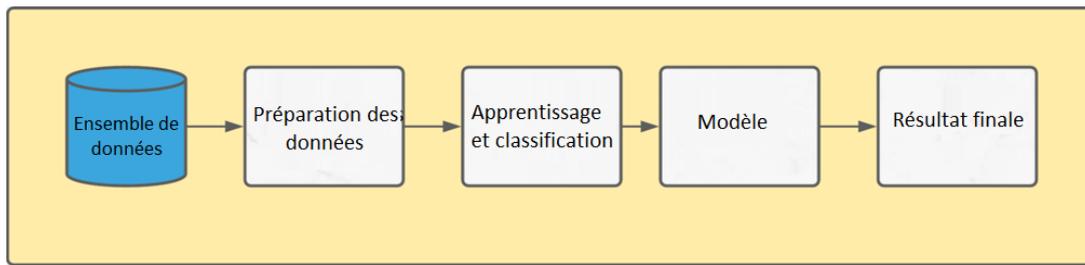


FIGURE 3.1 – Architecture générale du système

### 3.3 Architecture détaillée

Afin de construire un modèle d'apprentissage profond pour la classification avec 1D CNN cela peut être soigneusement classé, nous suggérons le plan dans la figure 3.2

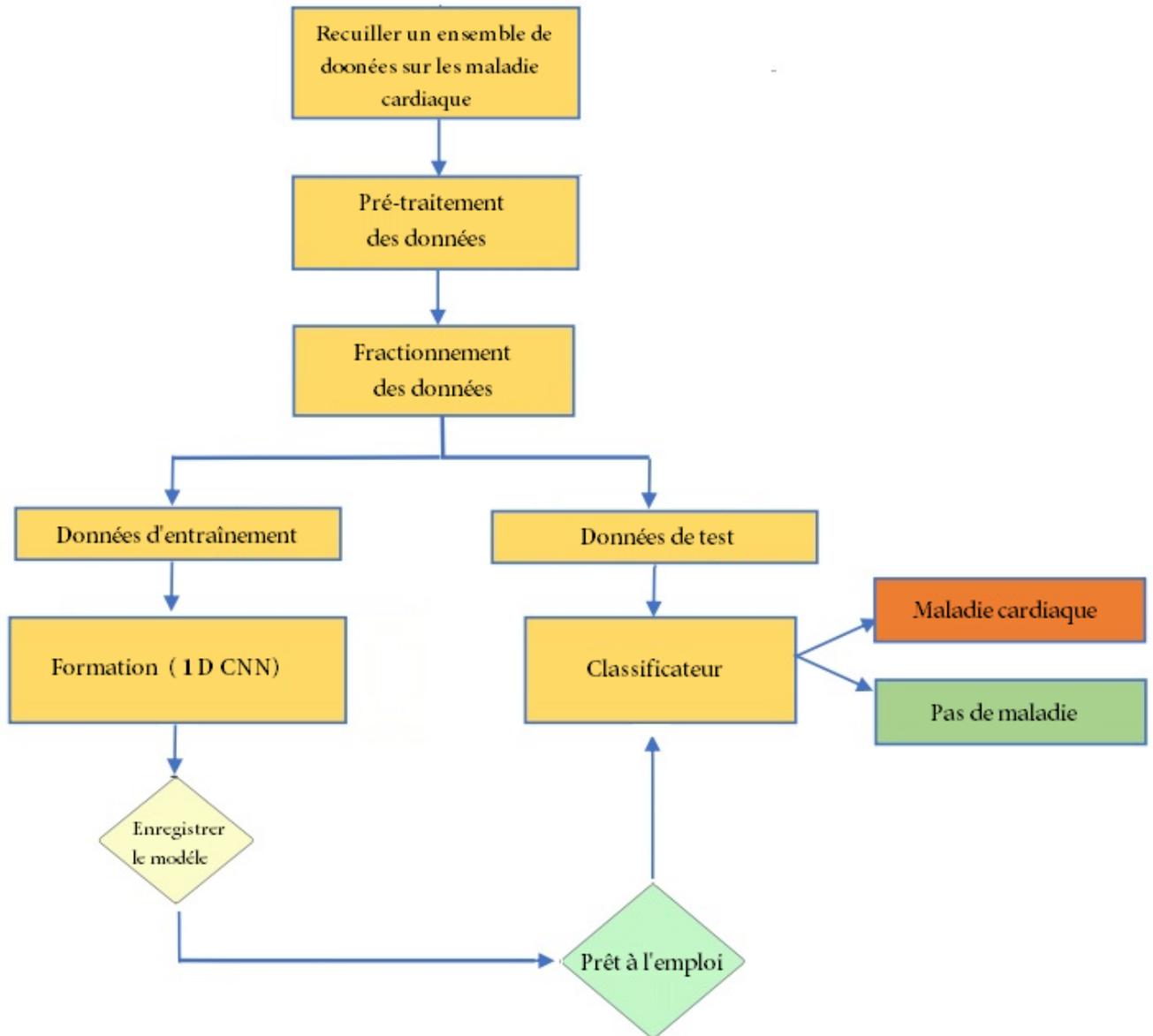


FIGURE 3.2 – Architecture détaillée du système

### 3.3.1 Recueillir un ensemble de données sur les maladie cardiaques

Nous avons utilisé une base de données publique de kaggle liée aux maladies cardiaques, cet ensemble de données se compose de quatre bases de données : Cleveland, Hongrie, Suisse et Long Beach V. Il contient 76 attributs y compris l'attribut attendu, mais tous les essais publiés indiquent l'utilisation d'un sous-ensemble de 14 d'entre eux. Le champ cible indique la maladie cardiaque du patient. Ce qui est un nombre valide de 0 = pas de maladie et 1 = maladie, Cet ensemble de données contient 303 lignes et 14 colonnes où chaque ligne correspond à un record.

1. (age).
2. (sex).

3. (cp).
4. (trestbps).
5. (chol).
6. (fbs).
7. (restecg).
8. (thalach).
9. (exang).
10. (oldpeak).
11. (slope).
12. (ca).
13. (thal).
14. (target) (l'attribut prédit) [14].

Fonctionnalités	Type	Description des données
âge	Numérique	Décrit l'âge de personne
sexe	Numérique	Décrit le sexe
cp	Numérique	Décrit les informations à propos de la douleur thoracique d'un personne
trestbps	Numérique	Décrit le Tension artérielle
chol	Numérique	Décrit le cholestérol Niveaux
fbs	Numérique	Fasting blood sugar levels
restecg	Numérique	Décrit électrocardiographique mesure
thalach	Numérique	Décrit la fréquencecardiaque maximale
exang	Numérique	Décrire l'information à propos de l'exercice
oldpeak	Numérique	Description numérique de l'exercice par rapport au repos
Slope	Numérique	Valeurs maximales de exercice
ca	Numérique	Informations sur les navires
thal	Numérique	taux de troubles sanguins
target	Numérique	Maladie cardiaque (0 = non, 1 = oui)

TABLE 3.1 – Ensemble de données sue les maladies cardiaques [71]

### 3.3.2 Prétraitement des données

Le prétraitement des données est une étape critique qui améliore la qualité des données afin de faciliter l'extraction d'informations significatives à partir de celui-ci. Il fait également référence au processus de nettoyer et organiser les données brutes afin de les rendre adaptées à la construction et à la formation Modèles d'apprentissage automatique. Les données prétraitées sont plus importantes que les algorithmes les plus sophistiqués, pour le point auquel les modèles d'apprentissage automatique entraînés sur de fausses données peuvent être préjudiciables à aboutissant aux résultats des « ordures ». Les données réelles que nous utilisons ne sont pas complètes, exactes, correctes ou cohérentes. Nous devons le faire résoudre ce problème car il est principalement causé par divers facteurs tels que la défaillance des compteurs intelligents, transfert peu fiable des données de mesure et maintenance non planifiée du système et des problèmes de stockage. Il y a différentes étapes nécessaires à ce stade pour y parvenir, Parmi eux :

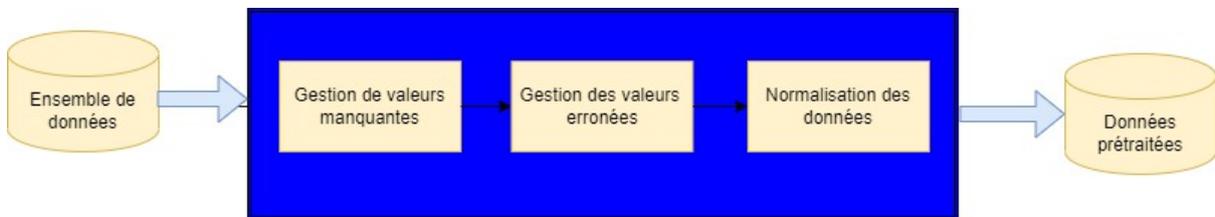


FIGURE 3.3 – Phase de prétraitement

#### Gestion des valeurs manquantes

Il est important de traiter les valeurs manquantes pour une gestion réussie des données. S'il manque les valeurs ne sont pas gérées de manière appropriée, il est possible qu'une inférence incorrecte au sujet de les données seront tirées à la suite d'un mauvais traitement. Le type de données perdues est une donnée numérique, qui apparaît sous la forme NaN. Nous profitons de l'hôpital un moyen de restaurer les valeurs perdues selon l'équation suivante : [83]

$$F(x_i) = \begin{cases} \frac{x_{i-1} + x_{i+1}}{2} & \text{if } x_i \in NaN, x_{i-1}, x_{i+1} \notin NaN \\ 0 & \text{if } x_i \in NaN, x_{i-1} \text{ or } x_{i+1} \in NaN \\ x_i & \text{if } x_i \notin NaN. \end{cases}$$

FIGURE 3.4 – La Formule de gestion des valeurs manquantes [83]

Où :

- $x_i$  : La valeur dans les données de consommation d'électricité sur une période.
- $x_{i+1}$  : la valeur suivante de  $x_i$ .

- $X_{i-1}$  : Valeur précédente de  $X_i$ .

### Gestion des valeurs erronées

La saisie de données inexactes est l'un des problèmes de saisie de données les plus courants. Un imprévu peut causer des problèmes immédiats ou à long terme. Cela pourrait également conduire à une inexactitude des enregistrements, les informations erronées et le désarroi. Avant d'apprendre nous devons corriger certaines valeurs incorrectes dans nos données. En conséquence nous utilisons l'équation suivante pour restaurer la valeur.

$$F(x_i) = \begin{cases} avg(x) + 2.std(x) & \text{if } x_i > avg(x) + 2.std(x) \\ x_i & \text{otherwise.} \end{cases}$$

FIGURE 3.5 – La Formule de gestion des valeurs erronées [83]

Où :

- $X$  : est un vecteur qui est composé de  $x_i$  jour par jour.
- $avg(x)$  : est la valeur moyenne de  $x$ .
- $std(x)$  : est l'écart-type de  $x$ .

### Normalisation des données

La normalisation des données est une technique permettant d'organiser les caractéristiques des données afin d'améliorer la cohérence des différents types d'entités dans un modèle de données. Il y a de nombreux objectifs à atteindre tout au long du processus de normalisation des données. La première étape consiste à éliminer toutes les données en double qui peuvent exister dans la collecte de données. Le deuxième objectif est d'organiser les données de manière rationnelle. Le réseau neuronal est sensible à diverses données. Pour éviter ce problème, nous devons d'abord normaliser le jeu de données. Nous avons utilisé la méthode de mise à l'échelle MAX-MIN, qui est basée sur l'équation suivante : [83]

$$F(x_i) = \frac{x_i - \min(x)}{\max(x) - \min(x)}$$

FIGURE 3.6 – La formule de Normalisation des données [83]

Où :

- $\min(x)$  : est la valeur minimale en  $x$ .
- $\max(x)$  : est la valeur maximale en  $x$ .

### 3.3.3 Apprentissage CNN

Les couches de convolution, les couches de regroupement et les couches entièrement connectées font partie du bâtiment composants de l'architecture CNN. Une architecture typique comprend un ou plusieurs des couches entièrement connectées suivies d'une pile de nombreuses couches de convolution et d'une couche de regroupement. La propagation directe sur un jeu de données d'apprentissage est l'étape lors de la transformation des données d'entrée en sortie à travers ces couches. Un algorithme de rétropropagation est une approche populaire pour l'entraînement des réseaux de neurones, et il repose fortement sur la fonction de perte et le gradient processus d'optimisation de la descente. La fonction de perte calcule les performances du formulaire dans Perles et poids spécifiques utilisant l'étalement frontal sur l'ensemble de données d'entraînement, apprenables les paramètres, tels que les grains et les poids, sont modifiés en fonction de la valeur de la perte en utilisant [82] Amélioration de la diffusion arrière et de la pente. Les méthodes d'encapsulation et d'assemblage peuvent être utilisées pour décrire les CNN 1D unidimensionnels. Certains paramètres doivent être donnés lors de la formation et de l'installation du formulaire. En voici quelques-uns en bref mentionnons :

**Taux d'apprentissage :** Le taux d'apprentissage est un paramètre super réglable qui contient valeur positive modeste, généralement comprise entre 0,0 et 1,0, utilisée dans l'entraînement neuronal réseau. Le taux d'apprentissage est un paramètre qui détermine la rapidité avec laquelle le modèle s'adapte à position. Il est servi comme une constante de promotion de poids pour mettre à jour de manière régulière et progressive .

**Optimiseurs dans CNN :** Les optimiseurs sont des techniques ou des approches qui ajustent les caractéristiques des réseaux de neurones, telles que les poids et le taux d'apprentissage, pour réduire les pertes. L'optimiseur dans un modèle d'apprentissage CNN peut améliorer les résultats du modèle et raccourcir le temps qu'il faut pour l'entraîner de quelques jours à des heures ou des minutes. Quelques-uns des nombreux optimiseurs disponibles sont : optimiseur de descente de gradient, descente de gradient stochastique, descente de gradient Mini batch, Optimiseur RMSprop et Optimiseur Adam.

**Fonction de perte :** Pour réduire l'erreur algorithmique, une fonction de perte est utilisée pour calculer cela erreur lorsqu'elle est utilisée pour déterminer les performances du modèle.

**Époque :** le nombre d'époques est égal au nombre de fois que l'ensemble de données complet est transmis en avant et en arrière par l'algorithme des réseaux de neurones. Chaque époque est terminé lorsque l'algorithme a affiché tous les échantillons du jeu de données

### Réseau neuronal convolutif unidimensionnel (1D CNN)

Des réseaux CNN appelés réseaux de neurones de contournement (1D CNN) ont récemment été développés et ces études ont conduit à il s'avère que pour certaines applications, les 1D CNN sont utiles et donc meilleurs que leurs homologues 2D dans le traitement avec des signaux 1D pour les raisons suivantes :

- FP et BP dans 1D cnn nécessitent des processus de baie simples qui signifient ceci la complexité de calcul des 1D CNN est beaucoup plus faible que celle des CNN 2D.
- Des études récentes montrent que les 1D CNN avec des architectures relativement peu profondes (c'est-à-dire un petit nombre de couches cachées et neurones) sont capables d'apprendre des tâches difficiles impliquant des signaux 1D que les 2D CNN exigent des architectures plus profondes à traiter et nous en concluons que les réseaux bruns peu profonds sont beaucoup plus faciles pour la formation et la mise en oeuvre.
- En raison de ses faibles exigences de calcul le 1D CNNs compact est bien adapté au temps réel et à faible coût en raison de ses faibles exigences de calcul surtout sur les appareils mobiles [55].

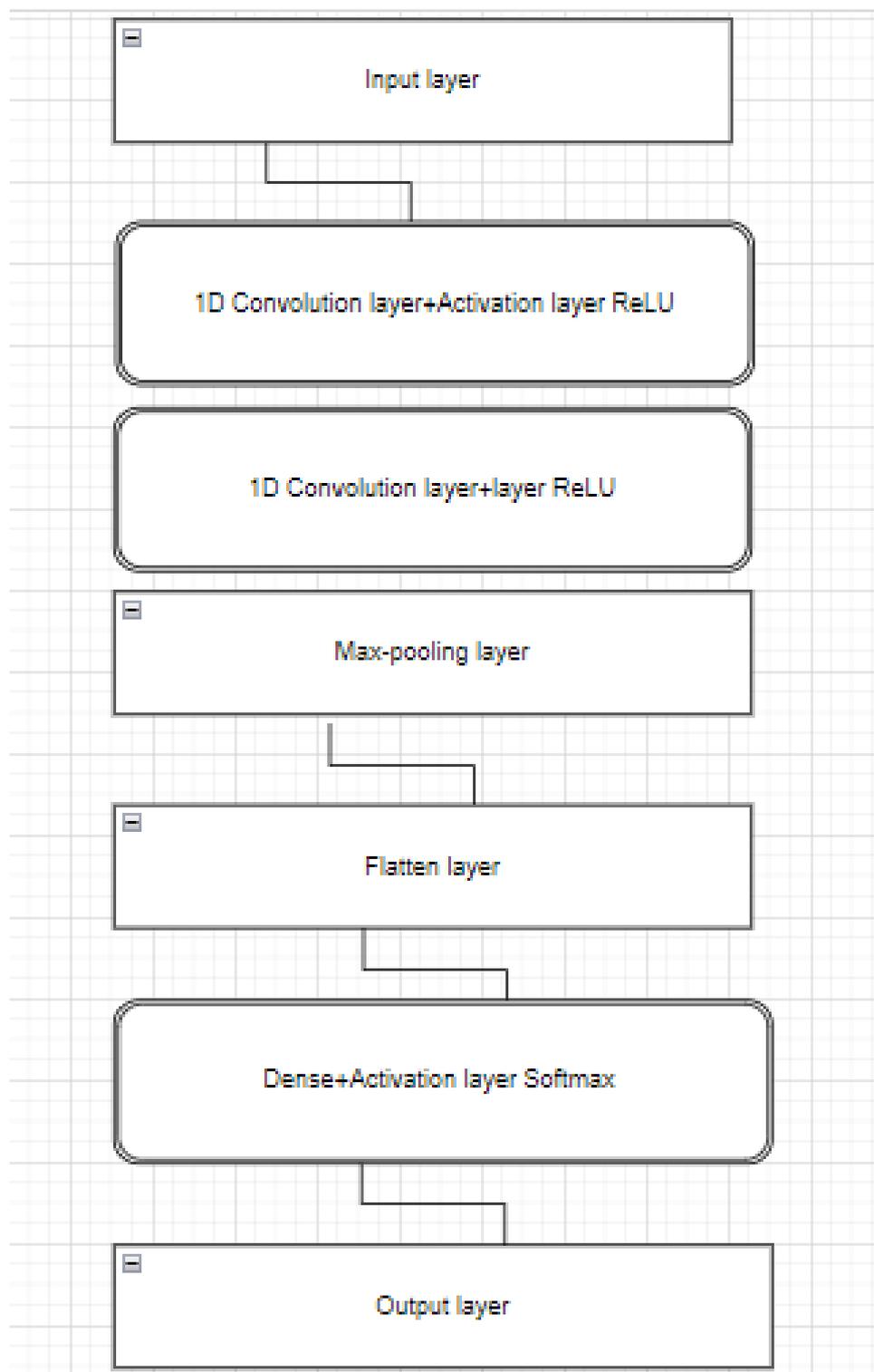


FIGURE 3.7 – Conception détaillée du réseau 1D-CNN

La structure proposée comprend :

- Envelopper la couche d'entrée.
- Une masse de contournement.

- 2 couches entièrement connectées.
- 1 couche de sortie.

Pour cette expérience et tout ce chapitre, un modèle CNN très simple a été considéré il se compose de sept couches. Les deux premières couches sont constituées de données d'entrée unidimensionnelles  $13 \times 1$  Il est juré par 64 filtres de taille 2 ,la troisième couche est une couche convolutive entièrement connectée avec 16 neurones.Nous appliquons Max-Pooling au deuxième calque de torsion simplement pour renvoyer la valeur maximale de toutes les valeurs. L'application de MaxPooling sur une couche de dérivation crée un bloc de débordement qui augmente la précision du formulaire.Après l'étape d'aplatissement qui est généralement utilisée comme lien entre la convolution et couches denses, nous retrouvons avec un long vecteur de données d'entrée que nous passons ensuite le réseau de neurones de contournement pour un traitement ultérieur.La sixième couche est également une couche entièrement connectée avec 3 neurones.La septième et dernière couche sera une couche de sortie avec deux classes possibles une selon le jeu de données.

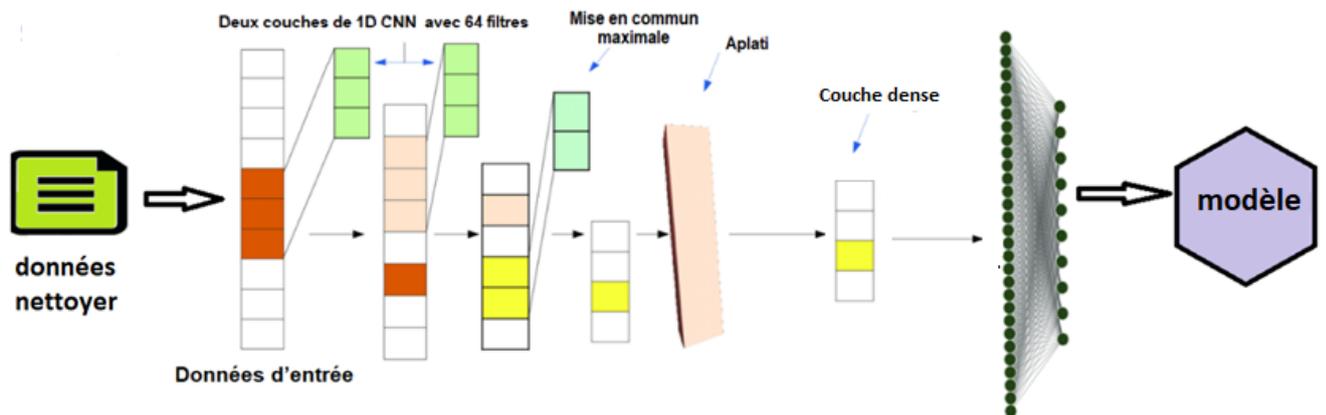


FIGURE 3.8 – Phase de modèle de l'apprentissage de 1D CNN

### 3.3.4 Usage

La phase d'utilisation prévoit la mise en œuvre effective du système, ce qui peut bénéficier au secteur de la santé. Nous allons passer par là lorsque nous aurons testé nos modèles et trouvé ils fonctionnent bien en obtenant de meilleurs résultat

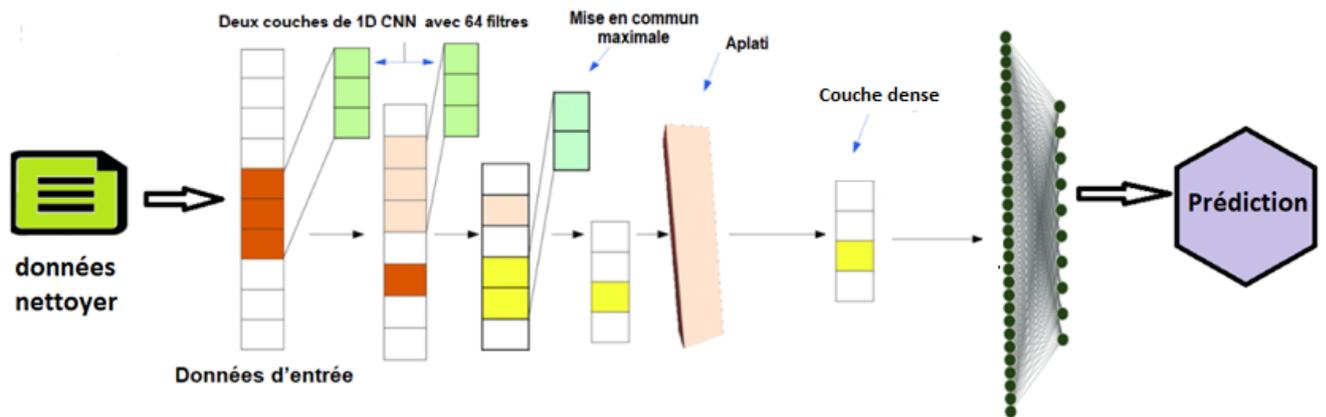


FIGURE 3.9 – Phase de test du modèle 1D CNN

Voici une description des processus clés qui contrôlent les couches de notre réseau 1D CNN que nous avons créé :

### 3.3.5 Fonction RELU

Les Fonctions d'activation : la fonction RELU (Rectified Linear Unit ) est la fonction d'activation la plus utilisée dans presque tous les réseaux neuronaux CNN. Elle active l'entrée juste lorsque l'entrée est plus élevée alors une certaine quantité ,la fonction Relu est définie par : [36].

$$f(x) = \max(0, x) = \begin{cases} x, & \text{si } x \geq 0 \\ 0, & \text{si } x < 0 \end{cases}$$

FIGURE 3.10 – La formule de la fonction Relu [34]

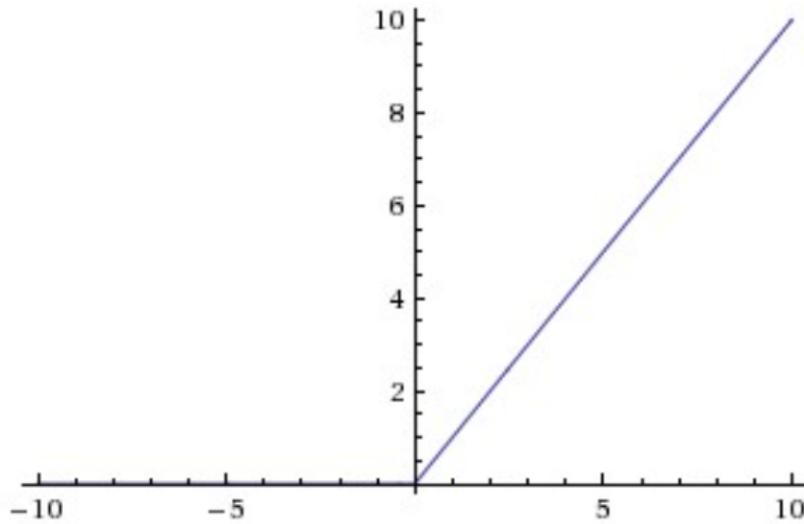


FIGURE 3.11 – Un graph de l’application de la fonction d’activation Relu [36]

### 3.3.6 Fonction SoftMax

Nous avons utilisé le SoftMax comme fonction d’activation dans le niveau de sortie parce que notre problème est un problème de classification multi-classes ou l’appartenance à la classe est requise sur plus de deux étiquettes de classe ,elle est définie par la formule suivante :[36].

$$f(x_i) = \frac{e^{x_i}}{\sum_j e^{x_j}}$$

FIGURE 3.12 – La formule de la fonction Softmax [34]

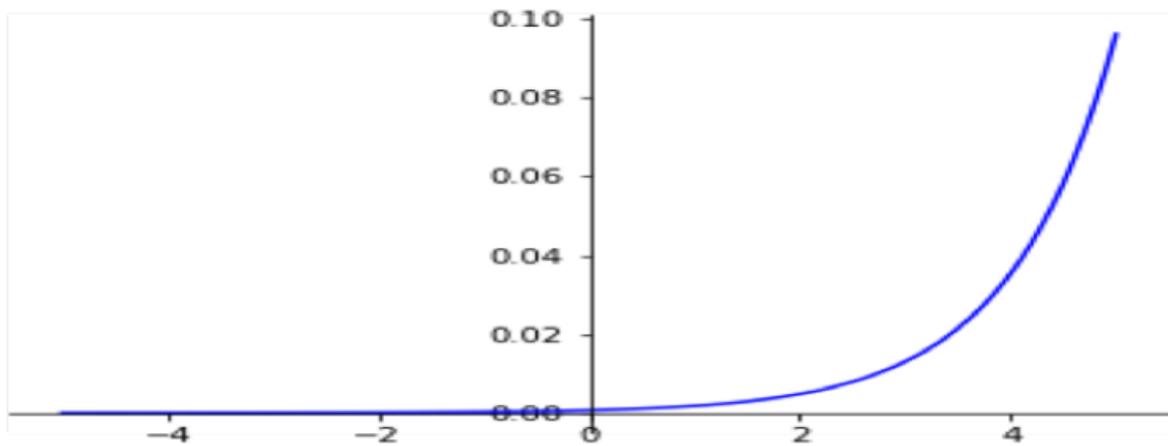


FIGURE 3.13 – Un graph de l’application de la fonction d’activation Softmax [36]

### 3.3.7 MaxPooling

Cette couche représente la deuxième couche du bloc de contournement, qui est appliquée pour aider à sur-apprendre en fournissant une forme abstraite de représentation. En outre il réduit le coût du compte en réduisant le nombre de paramètres à apprendre et en assurant fiabilité de la traduction de base de la représentation interne [36].

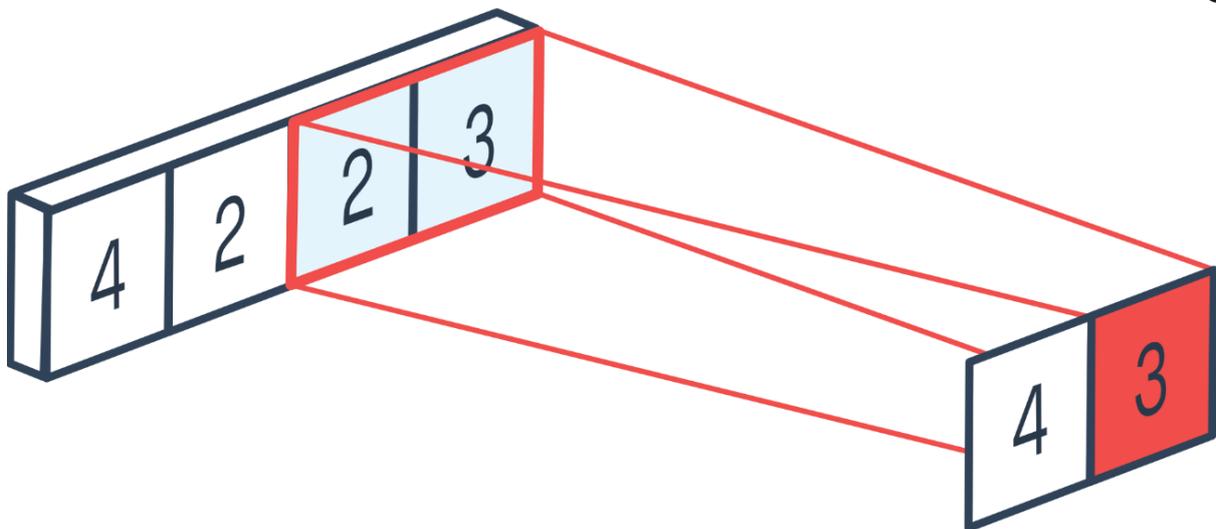


FIGURE 3.14 – Max Pooling [21]

## 3.4 Conclusion

Dans le chapitre trois, nous avons contribué à une nouvelle architecture CNN appelée 1D CNN . Nous détaillons les étapes et la structure proposée, et analysons, étudions et préparons la base de données pour appliquer la méthode proposée pour extraire des informations et obtenir des résultats..

# Chapitre **4**

## Implémentation

### 4.1 Introduction

Dans ce chapitre, nous présentons l'environnement de travail, le langage de programmation et tous les outils utilisés dans cette étude Jupyter Book, Python et Anaconda, et affichons quelques images de l'interface de l'application et de la base de données que nous avons utilisées en plus d'expliquer la méthode utilisée et les résultats obtenus et comparaison avec un autre modèle.

### 4.2 Préparation de l'environnement de mise en oeuvre

Lors de la préparation de l'environnement d'implémentation, nous avons installé la distribution Anaconda pour le système d'exploitation Windows 10, nous avons utilisé la version Anaconda 1.9.7. C'est l'installation : <https://www.anaconda.com/distribution/> [1].

### 4.3 Outils et langages de développement

Notre choix s'est orienté vers le langage de programmation Python et jupyter en utilisant le framework de haut niveau pour le Deep Learning Keras et Tensorflow, ainsi que des bibliothèques pour la visualisation mathématique et exploration de données (Matplotlib, Numpy, Pandas, sklearn).

#### 4.3.1 Python

Python est le langage parfait pour les scripts et le développement rapide dans de nombreux domaines. Il dispose de structures de données de haut niveau efficaces et d'une approche simple mais efficace de la programmation orientée objet. Une syntaxe de serpent élégante et une écriture dynamique ainsi que sa nature interprétée en font langage

idéal pour les scripts et le développement rapide d'applications dans de nombreux domaines sur la plupart des plates-formes. Python Translator et la bibliothèque standard spacieuse sont disponibles gratuitement sous la forme d'une double source pour toutes les principales plates-formes et contiennent de nombreux modules, programmes et outils gratuits Python [75].

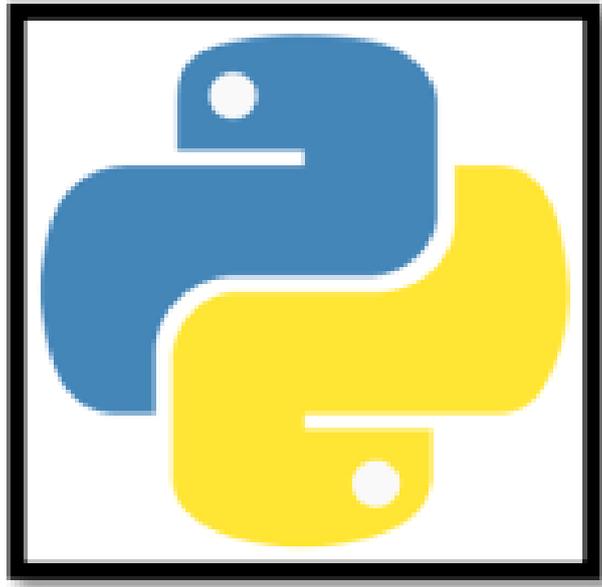


FIGURE 4.1 – logo python [63]

### 4.3.2 Anaconda Distribution

Anaconda est une distribution gratuite et open-source des langages de programmation python et R pour le calcul scientifique parmi ces applications : science des données, applications d'apprentissage automatique, traitement de données à grande échelle, analyse prédictive, etc. elle est facilitée la création des environnements sur python [2]. La distribution Anaconda open-source est le moyen le plus simple d'effectuer des données Python/R science et apprentissage automatique sur Linux, Windows et Mac OS X. Avec plus de 11 millions d'utilisateurs du monde entier, c'est la norme de l'industrie pour le développement, les tests et la formation sur une seule machine .

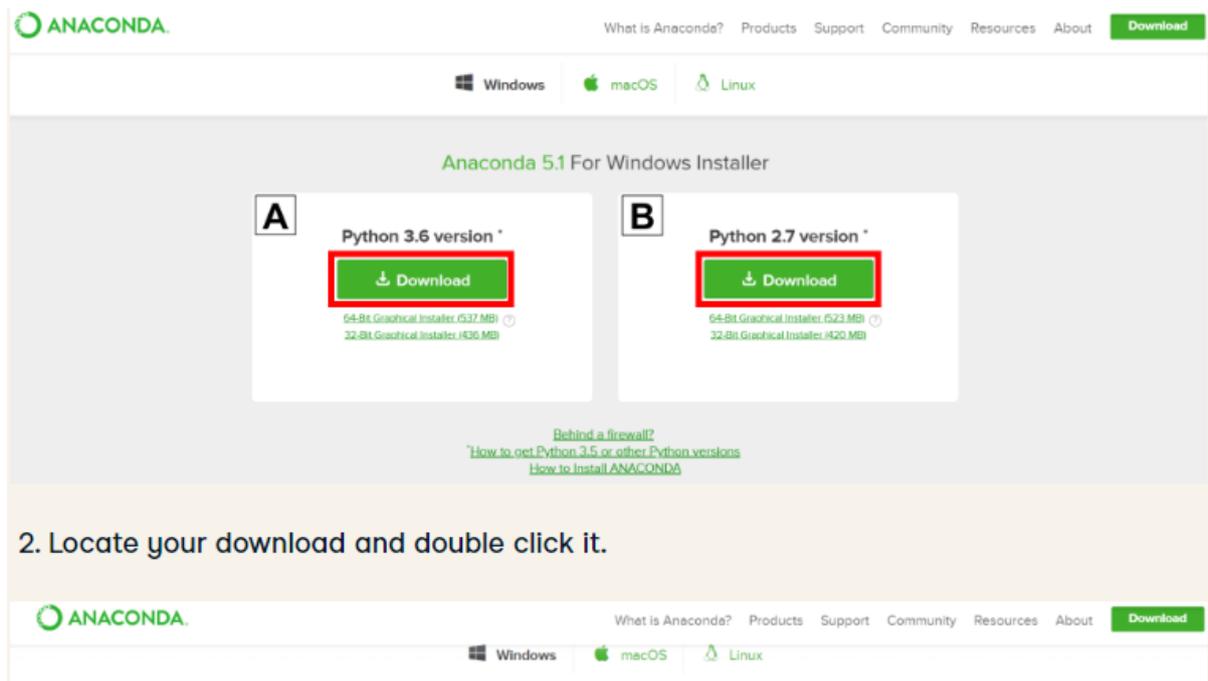


FIGURE 4.2 – Installation de anaconda [1]

1. Téléchargez rapidement plus de 1 500 packages de science des données Python/R.
2. Gérez les bibliothèques, les dépendances et les environnements avec Conda.
3. Développer et former des modèles d'apprentissage automatique et d'apprentissage profond avec scikit-learn, TensorFlow et Theano.
4. Analysez les données avec évolutivité et performances avec Dask, NumPy, pandas et Numba.
5. Visualisez les résultats avec Matplotlib, Bokeh, Datashader et Holoviews [1].

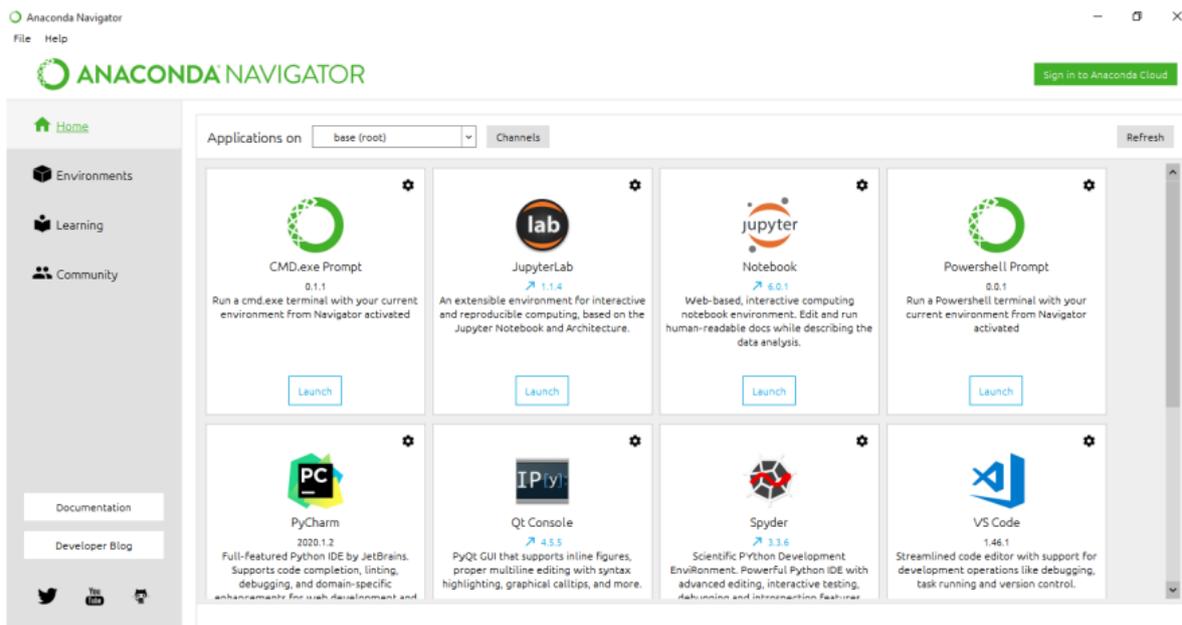


FIGURE 4.3 – Fenêtre d'accueil d'ANACONDA NAVIGATOR [1]

### 4.3.3 Définition de tensorflow

TensorFlow est un système d'apprentissage automatique qui fonctionne à grande échelle et dans des environnements hétérogènes. Il utilise des graphiques de flux de données pour représenter le calcul, l'état partagé et les opérations qui mutent cet état. TensorFlow prend en charge une variété d'applications en mettant l'accent sur l'entraînement et l'inférence sur les réseaux de neurones profonds[29].

### 4.3.4 TensorFlow dans Anaconda

Un certain nombre de méthodes peuvent être utilisées pour installer TensorFlow, telles que l'utilisation de pip pour installer les roues disponibles sur PyPI. L'installation de TensorFlow à l'aide de packages conda offre de nombreux avantages, notamment un système de gestion de packages complet, une prise en charge plus étendue de la plateforme, une expérience GPU plus simple et de meilleures performances du processeur. Ces packages sont disponibles via le référentiel Anaconda et leur installation est aussi simple que d'exécuter `conda Install TensorFlow` ou `conda Install tensorflow-gpu` à partir d'une interface de ligne de commande [63].



FIGURE 4.4 – Logo TensorFlow [63]

### 4.3.5 Keras

Keras est une bibliothèque open source écrite en Python pour l'apprentissage profond et est facile à utiliser, intuitive et rapide. Il peut être utilisé pour travailler à un niveau élevé de certains frameworks tels que TensorFlow ou Knowledge Toolkit de Microsoft.



FIGURE 4.5 – keras [62]

### 4.3.6 Modèle sous Keras

Keras est aussi appelé API de haut niveau, elle permet d'effectuer une classification suivant les étapes suivantes :

1. Importer les modules et dépendances.
2. Préparation des données en format acceptable par l'API.
3. Construire le modèle en utilisant `tf.keras` (Tensorflow-Keras) puis le compiler.
4. Entraîner le modèle avec un Dataset préalablement préparé.
5. Evaluation du modèle.
6. Sauvegarder le modèle prêt à être utilisé en production.

### 4.3.7 Keras GPU vs CPU

Keras a la capacité d'exécuter Tensorflow à un niveau élevé, pour les tâches d'apprentissage automatique (accessible uniquement sur la plate-forme cloud legoogle). Cependant, Keras a la possibilité de former un modèle de CPU ou de GPU. Le CPU dont le rôle est d'effectuer des calculs séquentiels, l'unité de traitement GRAPHICS est capable de faire fonctionner un grand nombre de comptes en parallèle [68].

### 4.3.8 Matplotlib

Matplotlib est une bibliothèque du langage de programmation Python destinée à tracer et visualiser des données sous formes de graphiques. Elle peut être combinée avec les bibliothèques python de calcul scientifique NumPy et SciPy. Matplotlib est distribuée librement et gratuitement sous une licence de style BSD. Sa version stable actuelle (la 2.0.1 en 2017) est compatible avec la version 3 de Python [32].

### 4.3.9 Pandas

Panda est un pack Python qui fournit des structures de données rapides, flexibles et expressives conçues pour rendre le travail avec des données classifiées facile et intuitif. *Vise*

à être une infrastructure d'analyse de haut niveau C'est une source de données pratiques en Python. Elle a un objectif plus large de devenir l'outil d'analyse de traitement de données open source le plus flexible disponible dans n'importe quelle langue [22].

### 4.3.10 Sklearn

Scikit-learn (Sklearn) est la bibliothèque la plus utile et la plus robuste pour l'apprentissage automatique en Python. Il fournit une sélection d'outils efficaces pour l'apprentissage automatique et la modélisation statistique, y compris la classification, la régression, le clustering et la réduction de la dimensionnalité via une interface de cohérence en Python. Cette bibliothèque, qui est en grande partie écrite en Python et construite sur NumPy, SciPy et Matplotlib [24].

## 4.4 Utilisation d'environnements virtuels dans Jupyter Notebook et Python

Jupyter Notebook est une application web open source qui permet Vous créez et partagez des documents qui contiennent un code direct et des équations. Les utilisations comprennent le nettoyage et la transformation des données, la simulation numérique, la modélisation statistique, les données Visualisation, apprentissage automatique, etc [63].



FIGURE 4.6 – Logo jupyter [63]

### 4.4.1 Lancer jupyter

Pour lancer un bloc-notes Jupyter ouvrez votre terminal et accédez au répertoire dans lequel vous souhaitez enregistrer votre bloc-notes. Tapez ensuite la commande `jupyter notebook` et le programmeinstanciera un serveur local à `localhost :8888` (ou un autre port spécifié).

```
[9] -> jupyter notebook
[I 18:31:51.264 NotebookApp] Serving notebooks from, local directory: /Users/janedoe
[I 18:31:51.264 NotebookApp] 0 active kernels
[I 18:31:51.264 NotebookApp] The Jupyter Notebook is running at:
[I 18:31:51.264 NotebookApp] http://localhost:8888/?token=f9b639294933fa68c64c78effb9b6519f9d8c45c903baa43
```

FIGURE 4.7 – Lancer jupyter [14]

Une fenêtre de navigateur devrait immédiatement apparaître avec l’interface Jupyter Notebook sinon vous pouvez utiliser l’adresse qu’elle vous donne. Les ordinateurs portables ont un jeton unique puisque le logiciel utilise des conteneurs Docker prédéfinis pour mettre les ordinateurs portables sur leur propre chemin unique [14].

## 4.4.2 Interface Jupyter

Nous sommes maintenant dans l’interface Jupyter Notebook et nous pouvons voir tous les fichiers dans le répertoire actuel. Tous les blocs-notes Jupyter peuvent être identifiés par le code du bloc-notes à côté de son nom. Si vous avez déjà un bloc-notes Jupyter dans le répertoire actuel que vous souhaitez afficher, recherchez-le dans la liste des fichiers et cliquez dessus pour l’ouvrir .



FIGURE 4.8 – Interface Jupyter[14]

Pour créer un nouveau bloc-notes, accédez à nouveau et sélectionnez Python 3. S’il y a d’autres blocs-notes Jupyter sur le système que vous souhaitez utiliser, nous pouvons cliquer sur Télécharger et accéder à ce fichier particulier.

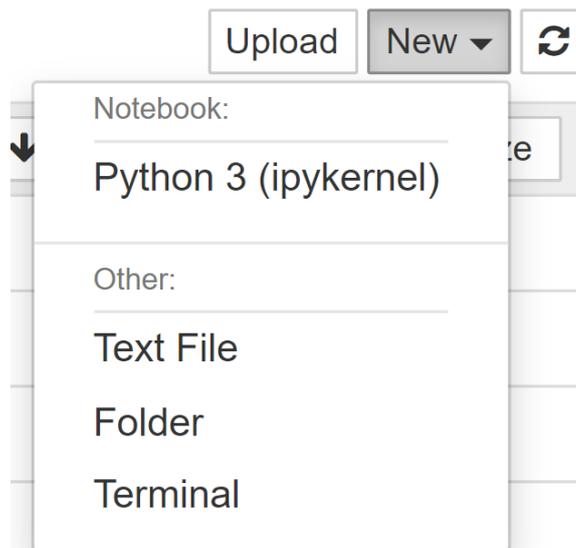


FIGURE 4.9 – Interface Jupyter [14]

Les blocs-notes en cours d'exécution auront une icône verte, tandis que ceux qui ne sont pas en cours d'exécution seront gris. Pour trouver tous les blocs-notes en cours d'exécution, cliquez sur l'onglet exécution pour voir une liste. A l'intérieur du bloc-notes lorsque vous ouvrez un nouveau bloc-notes Jupyter vous remarquerez qu'il contient une cellule. Les cellules sont la façon d'organiser les blocs notes et sont les zones où le code est écrit. Pour exécuter une partie du code, cliquez sur la cellule pour la sélectionner .

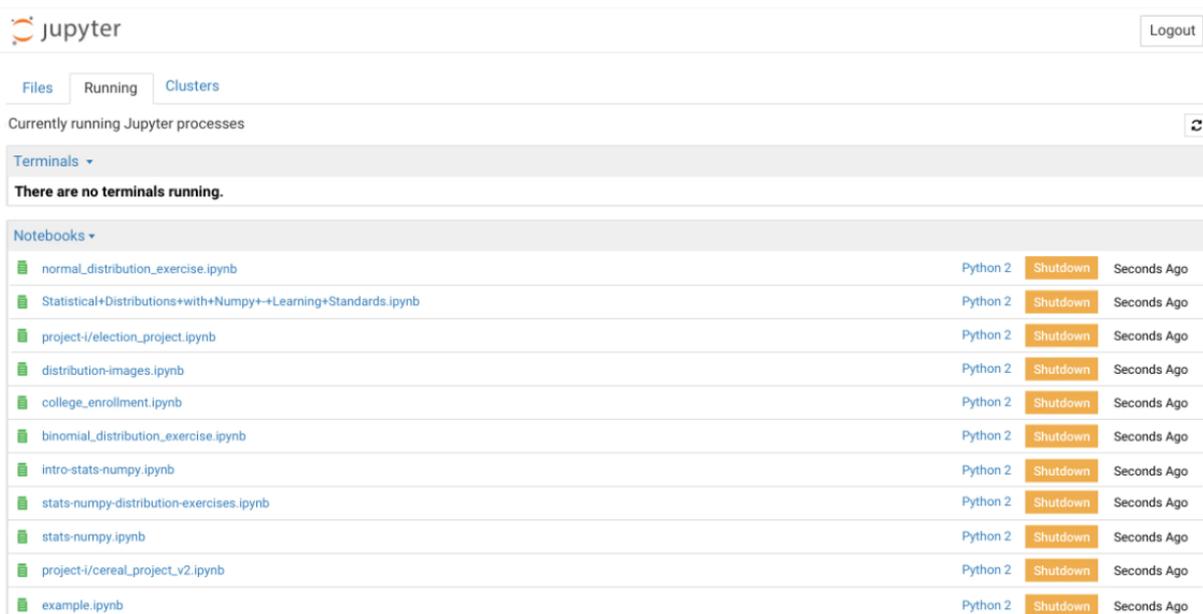


FIGURE 4.10 – Interface Jupyter [14]

A l'intérieur du bloc-notes lorsque vous ouvrez un nouveau bloc-notes Jupyter, vous remarquerez qu'il contient une cellule. Les cellules sont la façon d'organiser les blocsnotes et sont les zones où le code est écrit. Pour exécuter une partie du code cliquez sur la cellule pour la sélectionner [14].

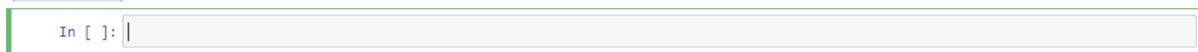


FIGURE 4.11 – cellules Jupyter [14]

## 4.5 Les bibliothèques

Importer des bibliothèques et des modules pour construire des réseaux de neurones profonds avec Keras et tensorflow, nous importons d'abord différentes bibliothèques et unités comme indiqué.

```
import numpy as np
from numpy import mean
from numpy import std
from numpy import dstack
from pandas import read_csv
from matplotlib import pyplot as plt
from tensorflow import keras
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense
from tensorflow.keras.layers import Flatten
from tensorflow.keras.layers import Dropout
from tensorflow.keras.layers import Conv1D
from tensorflow.keras.layers import MaxPooling1D, BatchNormalization
from tensorflow.keras.utils import to_categorical
from sklearn.metrics import confusion_matrix, ConfusionMatrixDisplay
import joblib
from sklearn.model_selection import train_test_split
import pandas as pd
```

FIGURE 4.12 – Importer des bibliothèques et des modules

## 4.6 Description des bibliothèques et modules utilisés

### 4.6.1 Numpy

NumPy est une bibliothèque Python utilisée pour travailler avec des tableaux. Il a également des fonctions pour travailler dans le domaine de l'algèbre linéaire, de la transformée de Fourier et des matrices, numPy a été créé en 2005 par Travis Oliphant. C'est un projet open source et vous pouvez l'utiliser librement [32].

### 4.6.2 Conv1D

Ce calque crée un noyau d'enveloppement qui est contourné avec le calque inséré à une distance unique (ou temporelle) pour produire un sutter de sorties, Si use bias vrai un vecteur de biais est créé et ajouté aux sorties. Enfin si l'activation n'est rien appliquez la également aux sorties. Lorsque vous utilisez ce calque comme premier calque dans un formulaire, fournissez un argument input shape (un ensemble de nombres corrects ou rien) [8].

### 4.6.3 Flatten

La classe d'aplatissement Keras est très importante et traite des entrées multidimensionnelles telles que les ensembles de données. La fonction `Keras.layers.flatten` aplatit les blocs d'entrée multidimensionnels en une seule dimension de sorte que vous pouvez structurer votre couche d'entrée, créer un modèle de réseau neuronal, puis transmettre efficacement ces données à chaque neurone du modèle [18].

### 4.6.4 Dense

Les réseaux denses sont des réseaux de contournement densément connectés qui sont très similaires à ResNet avec quelques différences fondamentales. ResNet utilise une méthode ajoutée qui signifie qu'ils prennent une sortie précédente comme entrée pour une couche future, et dans DenseNet prend toutes les sorties précédentes comme entrées pour une couche future [17].

### 4.6.5 Adam

Optimiseur qui implémente l'algorithme Adam est une méthode de descente de gradient stochastique basée sur l'estimation adaptative des moments du premier et du second ordre [32]

## 4.7 Structures de données

### 4.7.1 Partie pf ensemble de données utilisé

1	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	target
2	63	1	3	145	233	1	0	150	0	2.3	0	0	1	1
3	37	1	2	130	250	0	1	187	0	3.5	0	0	2	1
4	41	0	1	130	204	0	0	172	0	1.4	2	0	2	1
5	56	1	1	120	236	0	1	178	0	0.8	2	0	2	1
6	57	0	0	120	354	0	1	163	1	0.6	2	0	2	1
7	57	1	0	140	192	0	1	148	0	0.4	1	0	1	1

FIGURE 4.13 – maladie

67	1	0	160	286	0	0	108	1	1.5	1	3	2	0
67	1	0	120	229	0	0	129	1	2.6	1	2	3	0
62	0	0	140	268	0	0	160	0	3.6	0	2	2	0
63	1	0	130	254	0	0	147	0	1.4	1	1	3	0
53	1	0	140	203	1	0	155	1	3.1	0	0	3	0
56	1	2	130	256	1	0	142	1	0.6	1	1	1	0
48	1	1	110	229	0	1	168	0	1	0	0	3	0

FIGURE 4.14 – pas maladie

### 4.7.2 Pré-traitement des données

Au niveau principal l'ensemble de données est d'abord nettoyé et traité à l'aide de techniques de traitement. Nous téléchargeons les données sous forme d'ensemble de données

df et à l'aide de la fonction df, nous affichons les cinq premières lignes de l'ensemble de données avec toutes les colonnes. Il y a 14 colonnes, mais certains noms des colonnes ont une ambiguïté incompréhensible par exemple :cp est une douleur thoracique valant 1 à 4 et une pression artérielle de trestbps et est une variable constante afin de lever cette ambiguïté que nous renommons colonnes pour cp nous l'appelons comme douleur thoracique et trestbps comme pression artérielle et de la même manière nous complétons le reste des colonnes ,en utilisant les df types nous trouvons le type de données pour chaque colonne ,nous pouvons voir que la plupart d'entre elles sont à l'exception de la dépression de soulignement sc mais toutes les colonnes sont de type de données numérique,ensuite nous supprimons la colonne cible à l'aide de la fonction drop.

```
#Load and read the csv data
df = pd.read_csv("heart.csv")
```

```
# print first 5 rows of the data
df.head()
```

	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	target
0	63	1	3	145	233	1	0	150	0	2.3	0	0	1	1
1	37	1	2	130	250	0	1	187	0	3.5	0	0	2	1
2	41	0	1	130	204	0	0	172	0	1.4	2	0	2	1
3	56	1	1	120	236	0	1	178	0	0.8	2	0	2	1
4	57	0	0	120	354	0	1	163	1	0.6	2	0	2	1

```
df.columns = ['age', 'sex', 'chest_pain_type', 'resting_blood_pressure', 'cholesterol',
              'fasting_blood_sugar', 'rest_ecg', 'max_heart_rate_achieved',
              'exercise_induced_angina', 'st_depression', 'st_slope', 'num_major_vessels',
              'thalassemia', 'target']
```

FIGURE 4.15 – Partie du code

```
df.dtypes
age                int64
sex                int64
chest_pain_type    int64
resting_blood_pressure  int64
cholesterol        int64
fasting_blood_sugar  int64
rest_ecg           int64
max_heart_rate_achieved  int64
exercise_induced_angina  int64
st_depression      float64
st_slope           int64
num_major_vessels  int64
thalassemia        int64
target             int64
dtype: object
```

## Splitting the features and target

```
x, y = df.drop(columns=['target']) ,df.target
print(x.shape)
```

```
(303, 13)
```

```
x = np.expand_dims(x,-1)
print(x.shape)
```

```
(303, 13, 1)
```

```
print(np.unique(y))
print(np.unique(y).sum())
```

```
[0 1]
1
```

FIGURE 4.16 – partie du code

maintenant les données prêtes à créer un modèle 1D CNN.

### 4.7.3 Apprentissage des données

Après le prétraitement des données que nous utilisons indiqué dans la sous-section précédente, nous devrions diviser notre ensemble de données. Le fractionnement des données est un élément crucial de la science des données. La scission fait référence au processus de division des données en deux parties ou plus. Notre ensemble de données est divisé en deux parties, la première partie de nos données est utilisée pour entraîner le modèle avec 80% des données, et la deuxième partie est utilisée pour évaluer ou tester les données avec 20%.

## Splitting the data into Training data & Test data

```
xtrain, xtest, ytrain, ytest=train_test_split(x, y, test_size=0.2)
```

FIGURE 4.17 – Divisez les données

maintenant les données prêtes à créer un modèle 1D CNN.

#### 4.7.4 Créer des formulaires

Dans cette section, nous utiliserons 1D CNN pour expliquer comment nous avons utilisé deep Apprenez à prédire L'arrêt cardiaque ,Tout d'abord nous allons utiliser le modèle d'API série, qui est un moyen construire des modèles d'apprentissage profond qui nous permettent de les construire couche par couche dans l'ordre route.

```
model = Sequential()
```

FIGURE 4.18 – Modèle séquentiel

#### 4.7.5 Training

Pour entraîner le modèle 1D CNN, nous avons utilisé keras et tensorflow, une bibliothèque en langage serpent pour le classer comme le montre la figure ci-dessous.

```
model = Sequential()
#convolution
model.add(Conv1D(64, 2, activation="relu", input_shape=(13,1)))
# Max pooling
model.add(Conv1D(64, 2, activation="relu"))
model.add(Dense(16, activation="relu"))
model.add(MaxPooling1D())
#Flattening
model.add(Flatten())
#Full connection
model.add(Dense(2, activation = 'softmax'))
model.compile(loss = 'sparse_categorical_crossentropy',
              optimizer = "adam",
              metrics = ['accuracy'],
              )
model.summary()
```

FIGURE 4.19 – training Model 1D CNN

#### 4.7.6 Les paramètres d'apprentissage du réseau

Lors de la conception et de la configuration de notre modèle de deep learning nous avons essayé différentes configurations des réseaux des gens et de leur utilisation Nous proposons des paramètres de structure pour 1D CNN :

Model: "sequential\_1"

Layer (type)	Output Shape	Param #
conv1d_2 (Conv1D)	(None, 12, 64)	192
conv1d_3 (Conv1D)	(None, 11, 64)	8256
dense_2 (Dense)	(None, 11, 16)	1040
max_pooling1d_1 (MaxPooling1D)	(None, 5, 16)	0
flatten_1 (Flatten)	(None, 80)	0
dense_3 (Dense)	(None, 3)	243

=====  
 Total params: 9,731  
 Trainable params: 9,731  
 Non-trainable params: 0  
 =====

FIGURE 4.20 – paramètres du 1D CNN

Après cela nous passons au résumé des paramètres de lecteur pour ce formulaire bien qu'il n'ait pas atteint 2000 epochs, ce modèle prend un temps moyen par rapport aux autres types de réseaux et donne de très bons résultats.

```

history =model.fit(xtrain, ytrain, batch_size=32,epochs=2000,verbose=1,callbacks=[early_stopping] )
Epoch 1991/2000
8/8 [=====] - 0s 6ms/step - loss: 0.1046 - accuracy: 0.9545
Epoch 1992/2000
8/8 [=====] - 0s 5ms/step - loss: 0.1085 - accuracy: 0.9545
Epoch 1993/2000
8/8 [=====] - 0s 5ms/step - loss: 0.1133 - accuracy: 0.9587
Epoch 1994/2000
8/8 [=====] - 0s 6ms/step - loss: 0.1135 - accuracy: 0.9587
Epoch 1995/2000
8/8 [=====] - 0s 6ms/step - loss: 0.1070 - accuracy: 0.9711
Epoch 1996/2000
8/8 [=====] - 0s 6ms/step - loss: 0.1221 - accuracy: 0.9669
Epoch 1997/2000
8/8 [=====] - 0s 6ms/step - loss: 0.1339 - accuracy: 0.9504
Epoch 1998/2000
8/8 [=====] - 0s 6ms/step - loss: 0.1036 - accuracy: 0.9587
Epoch 1999/2000
8/8 [=====] - 0s 6ms/step - loss: 0.0966 - accuracy: 0.9587
Epoch 2000/2000
8/8 [=====] - 0s 5ms/step - loss: 0.1040 - accuracy: 0.9628
  
```

FIGURE 4.21 – modéliser l'entraînement avec des valeurs de précision et de perte

#### 4.7.7 Expériences et résultats obtenus

Pour obtenir un meilleur modèle, nous avons mené des expériences et appliqué l'ensemble de données de plusieurs manières y compris 1D CNN et random forest et nous avons

obtenu des résultats sommaires.

— **Réseau neuronal convolutif unidimensionnel (1D CNN)**

Pour arriver à un bon modèle, nous avons testé notre base de données sur CNN et obtenu les résultats ci-dessous nous sommes devenus efficaces résultats.

```
acc = model.evaluate(xtrain, ytrain)
print("Loss:", acc[0], " Accuracy:", acc[1])
```

8/8 [=====] - 0s 2ms/step - loss: 0.0866 - accuracy: 0.9752  
 Loss: 0.08661951124668121 Accuracy: 0.9752066135406494

FIGURE 4.22 – Notre modèle 1D CNN

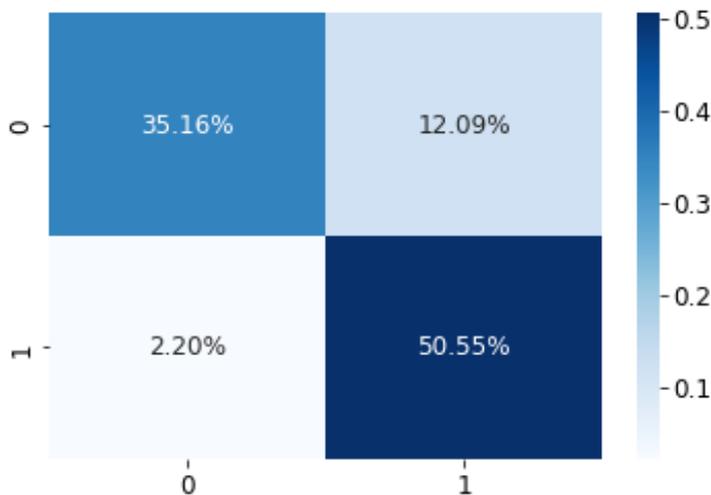
— **Forêt aléatoire** Les résultats du modèle forêt aléatoire n’ont pas donné d’excellents résultats par rapport à 1D CNN.

```
model = RandomForestClassifier( max_depth=5, n_estimators=110)
model.fit(X_train, y_train)
```

```
RandomForestClassifier(max_depth=5, n_estimators=110)
```

```
y_predict = model.predict(X_test)
```

```
cm = confusion_matrix(y_test, y_predict)
sns.heatmap(cm/np.sum(cm),annot=True,fmt=".2%", cmap='Blues')
plt.show()
ac = accuracy_score(y_test,y_predict)
re = recall_score(y_test,y_predict)
print('Accuracy is: ',ac*100)
print('recall is: ',re*100)
```



```
Accuracy is: 85.71428571428571
recall is: 95.83333333333334
```

FIGURE 4.23 – résultat du Modèle forêt aléatoire

— Résultat de la précision de compression

— **Compression accuracy result**

1D CNN	forêt aléatoire
97%	85%

TABLE 4.1 – Comparaison

Donc le meilleur modèle est CNN parce qu’il est plus grand qu’eux en ratio, il est approprié et efficace pour la détection a propos de l’arrêt cardiaque et des maladies cardiaques.

### 4.7.8 Test

```
cmd_obj.ax_.set(
title='Sklearn Confusion Matrix with labels!!',
xlabel='Predicted label',
ylabel='True label')

[Text(0.5, 1.0, 'Sklearn Confusion Matrix with labels!!'),
Text(0.5, 17.200000000000003, 'Predicted label'),
Text(17.199999999999999, 0.5, 'True label')]
```

FIGURE 4.24 – modèle fonctionne bien

## 4.8 Résultats obtenus

### 4.8.1 Présentation des performances obtenu

Pour visualiser les performances de notre 1D CNN de deep learning au fil du temps pendant training nous avons créée :

- Un graphe de "accuracy" sur l'ensemble de données train « acc » sur les époques d'entraînement (Figure 4.25).
- un graphe de "loss" sur l'ensemble de données train « loss » au cours des époques d'entraînement(Figure 4.26).

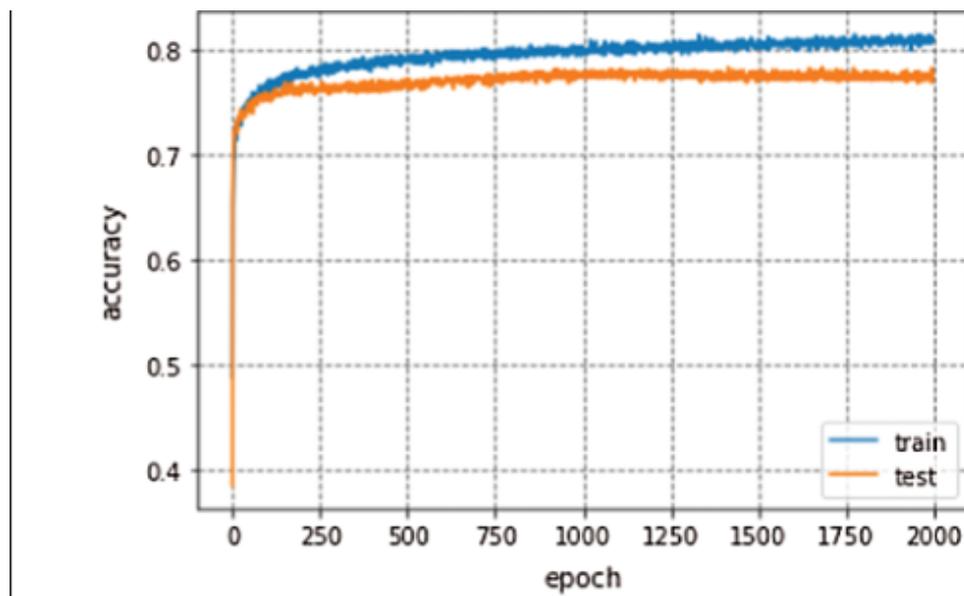


FIGURE 4.25 – La précision du modèle obtenue par 1D CNN

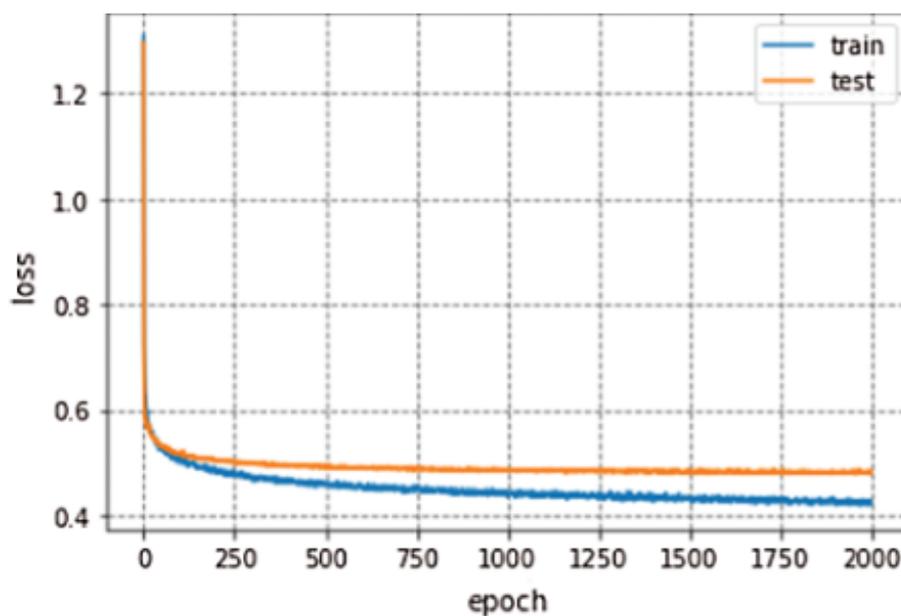


FIGURE 4.26 – La perte du modèle obtenue par 1D CNN

- D'après la Figure 4.25 La précision de l'apprentissage et de test augmente avec le Nombre d'époque ceci reflète qu'à chaque époque le modèle apprend plus d'informations.
- De même l'erreur d'apprentissage et de la validation diminue avec le nombre d'époque. Notre modèle a obtenu sa meilleure valeur de validation « accuracy, val-accuracy » dans l'époque numéro 2000 à cette époque les valeurs sont : une précision d'entraînement de 81.34% et l'erreur 5.1.

### 4.8.2 Évaluation du modèle sur les données de test

L'évaluation du test des données de test vise à estimer la précision de sa circulaire. Matrice de confusion est une technique de mesure du rendement utilisée dans l'évaluation de la performance du modèle construite dans la phase d'apprentissage sur un plateau des tests en comparant les catégories réelles et projetées. Fournit des informations plus détaillées évaluations correctes et incorrectes pour chaque catégorie. Confusion matricielle pour prédire avec précision les données de l'ensemble de test est comme le montre la figure suivante.

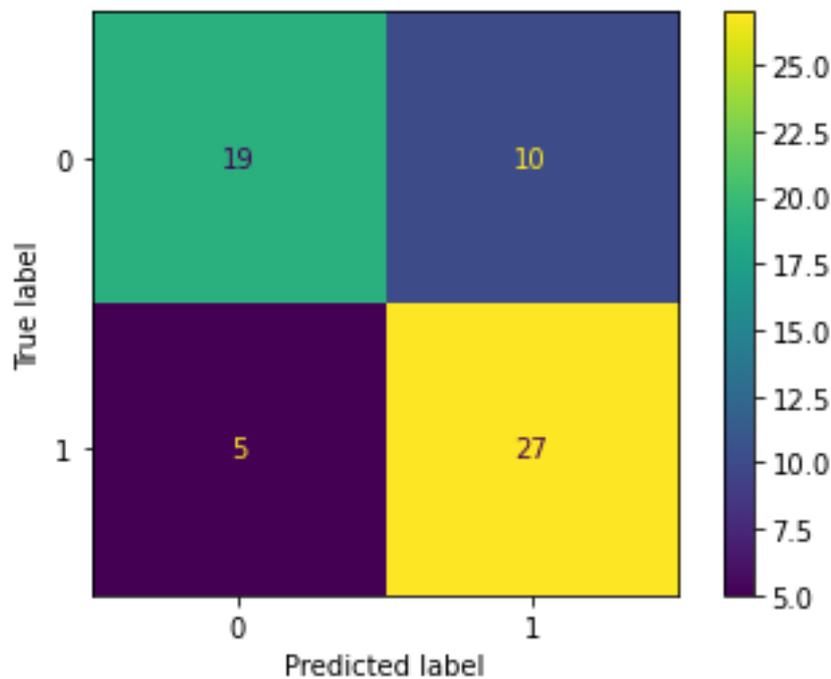


FIGURE 4.27 – Matrice de confusion

(Étiquette 0 = «Pas d'arrêt cardiaque» - Étiquette 1 = «Arrêt cardiaque») Chaque ligne de matrice de confusion représente des instances de classe réelles et chaque colonne représente les deux-points en tant qu'instances de classe valides. Les éléments horizontaux représentent la catégorie attendue. Notre système MP-CNN peut idéalement dire : nous avons 5 maladies classées comme aucune maladie, 19 ne sont pas des maladies classées comme aucune maladie, 27 il y a des maladies, elles ont été classées comme maladie, et 10 ne sont pas classées comme maladie. Notez que le modèle fait beaucoup d'erreurs. Dans les résultats globaux c'est presque bon.

## 4.9 Conclusion

Dans le chapitre quatre, nous avons contribué à une nouvelle structure CNN appelée 1D CNN que nous imaginons détaillant les étapes nécessaires pour afficher les outils et environnements utilisés lors du développement de ce projet, la création et la préparation de la base de données, la méthode proposée pour extraire les informations et les algo-

rithmes utilisés ainsi que les résultats obtenus montrant des gains de performances après comparaison.

# Conclusion générale

Les maladies cardiaques représentent l'une des maladies les plus dangereuses et récolte la vie de millions de des gens partout dans le monde. En conséquence, la prédiction des maladies cardiaques c'est un aspect important de l'analyse des données cliniques. Dans ce travail nous sommes concentrés sur l'application de techniques d'apprentissage profond qui ont la capacité de modéliser une classification de haute précision pour prédire les maladies cardiaques, ce qui peut aider en sauvant la vie des gens et en fournissant un traitement immédiat à éviter problèmes physiques et psychologiques.

Prédire l'arrêt cardiaque basée sur des techniques d'apprentissage profond il a été utilisé récemment et a montré son efficacité sur les prévisions traditionnelles. Nous avons mis en oeuvre la technologie de l'apprentissage profond 1D CNN Sur l'ensemble de données de cardiologie de kaggle pour prédire les maladies cardiaques et les arrêts cardiaques . Cette technique aide au diagnostic précoce des maladies cœur à un coût bon marché et avec une grande précision, la structure du Réseau neuronal convolutif unidimensionnel 1D CNN est l'un des meilleurs algorithmes de notation car elle donne les meilleurs résultats par rapport à d'autres algorithmes tels que forêt aléatoire.

# Bibliographie

- [1] ANACONDA DISTRIBUTION. URL : <https://www.anaconda.com/products/distribution>.
- [2] Anaconda (distribution Python). URL : [https://fr.wikipedia.org/wiki/Anaconda\\_\(distribution\\_Python\)](https://fr.wikipedia.org/wiki/Anaconda_(distribution_Python)).
- [3] Angine de poitrine. URL : <https://www.sante-sur-le-net.com/maladies/cardiologie/angine-de-poitrine/>.
- [4] Arrythmia. URL : <https://www.omicsonline.org/finland/arrythmia-peer-reviewed-pdf-ppt-articles/>.
- [5] Cardiac arrest. URL : <https://www.bhf.org.uk/information-support/conditions/cardiac-arrest>.
- [6] Cardiac Arrest. URL : [https://croi.ie/health/heart-conditions/cardiac-arrest/?gclid=Cj0KCQjw4uaUBhC8ARIsANUuDjXNuyrcdVLQGibhGEBXVwa-pkEiZoygK7n\\_xoaAtZFEALw\\_wcB](https://croi.ie/health/heart-conditions/cardiac-arrest/?gclid=Cj0KCQjw4uaUBhC8ARIsANUuDjXNuyrcdVLQGibhGEBXVwa-pkEiZoygK7n_xoaAtZFEALw_wcB).
- [7] Comprendre le machine learning et le deep learning. URL : <https://www.bial-r.com/2019/05/22/comprendre-le-machine-learning-et-le-deep-learning/>.
- [8] Conv1D layer. URL : [https://keras.io/api/layers/convolution\\_layers/convolution1d/](https://keras.io/api/layers/convolution_layers/convolution1d/).
- [9] Deep Belief Network. URL : <https://deeppai.org/machine-learning-glossary-and-terms/deep-belief-network>.
- [10] Définition du terme Oreillette droite :. URL : <https://www.doctissimo.fr/sante/dictionnaire-medical/oreillette-droite>.
- [11] Définition du terme Ventricule droit . URL : <https://www.doctissimo.fr/sante/dictionnaire-medical/ventricule-droit#:~:text=Le%20ventricule%20droit%20est%20l,et%20la%20couche%20bulbeuse%20spirale>.
- [12] Figure 2 - uploaded by Ibrahim A. Hameed. URL : [https://www.researchgate.net/figure/Six-different-types-of-deep-learning-models-applied-for-anomaly-detect-fig1\\_348231118](https://www.researchgate.net/figure/Six-different-types-of-deep-learning-models-applied-for-anomaly-detect-fig1_348231118).
- [13] Heart Disease in Canada. URL : <https://www.canada.ca/en/public-health/services/publications/diseases-conditions/heart-disease-canada.html>.
- [14] How to Use Jupyter Notebooks. URL : <https://www.codecademy.com/article/how-to-use-jupyter-notebooks>.

- [15] Human heart : Anatomy, function facts. URL : <https://www.livescience.com/34655-human-heart.html>.
- [16] Infarctus du myocarde. URL : <https://www.sante-sur-le-net.com/maladies/cardiologie/infarctus-du-myocarde/>.
- [17] Introduction to DenseNets (Dense CNN). URL : <https://www.analyticsvidhya.com/blog/2022/03/introduction-to-densenets-dense-cnn/>.
- [18] Keras Flatten with a DNN example in Python. URL : <https://neuralnetlab.com/keras-flatten-dnn-example/>.
- [19] Le cœur. URL : <https://quizlet.com/107725280/le-coeur-flash-cards/>.
- [20] Les types de maladies du cœur. URL : <https://www.coeuretavc.ca/maladies-du-coeur/qu-est-ce-que-les-maladies-du-coeur/les-types-de-maladies-du-coeur>.
- [21] Max pooling block 1D. URL : <https://peltarion.com/knowledge-center/documentation/modeling-view/build-an-ai-model/blocks/max-pooling-block-1d>.
- [22] pandas : powerful Python data analysis toolkit. URL : <https://pypi.org/project/pandas/>.
- [23] Rythme sinusal normal : comment le reconnaître sur un électrocardiogramme ? URL : <https://fr.my-ekg.com/conseils-ecg/rythme-sinusal-diagnostic.html>.
- [24] Scikit Learn Tutorial. URL : [https://www.tutorialspoint.com/scikit\\_learn/index.htm](https://www.tutorialspoint.com/scikit_learn/index.htm).
- [25] Sudden cardiac arrest. URL : <https://www.mayoclinic.org/diseases-conditions/sudden-cardiac-arrest/symptoms-causes/syc-20350634?fbclid=IwAR0QH3KaJ8aqWVF9Y-oQEDE0vMFNmsQgO6bk-9cRCT0HmxJYpKw7dZfxy1Q>.
- [26] Troubles du rythme - Généralités. URL : [https://www.fmc-tourcoing.org/PROGRAMME\\_FMC/RYTHME/troubles\\_du\\_rythme\(1\).htm](https://www.fmc-tourcoing.org/PROGRAMME_FMC/RYTHME/troubles_du_rythme(1).htm).
- [27] Types of Machine Learning Algorithms. URL : <https://pythonsansar.com/types-machine-learning-algorithms/>.
- [28] What happens in Sparse Autoencoder. URL : <https://medium.com/@syoya/what-happens-in-sparse-autencoder-b9a5a69da5c6>.
- [29] Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, et al. {TensorFlow} : A system for {Large-Scale} machine learning. pages 265–283, 2016.
- [30] Dilmi Abdelbaqi. Failure prediction in cloud environment using deep learning. 2020.
- [31] KABOT ABDELGHANI. Heart disease analysis based on data mining, deep learning and wearable technologies. 2019.
- [32] MAMEN ABDELKARIM. Développement d’une architecture cnn pour la classification des images radiologiques d’infections pulmonaires.
- [33] Karim Aderghal. Classification of multimodal mri images using deep learning : Application to the diagnosis of alzheimer’s disease. 2021.
- [34] MOHAMED AMINE AID. La détection des webshells pour assurer la sécurité des serveurs web. 2021.

- [35] Saleh Albelwi and Ausif Mahmood. A framework for designing the architectures of deep convolutional neural networks. page 242, 2017.
- [36] AHRIZ AYMEN. Développement d'une architecture cnn pour la prédiction du taux de survie des malades glioblastome gbm.
- [37] Valentine Baert. Evaluation des stratégies médicamenteuses dans la prise en charge des arrêts cardiaques extrahospitaliers. 2020.
- [38] Youssouf Benyettou. Drug design and discovery using artificial intelligence. 2019.
- [39] Charles-Henri Cappelaere. *Estimation du risque de mort subite par arrêt cardiaque a l'aide de méthodes d'apprentissage artificiel*. 2014.
- [40] Laurent Castra. Typologie des arrêts cardiaques au regard des inégalités sociales et territoriales de santé en ile-de-france : application au registre national des arrêts cardiaques (réac). 2018.
- [41] Utsav Chauhan, Vikas Kumar, Vipul Chauhan, Sumit Tiwary, and Amit Kumar. Cardiac arrest prediction using machine learning algorithms. pages 886–890, 2019.
- [42] Sophie Collin. Preoperative planning and simulation for artificial heart implantation surgery. 2018.
- [43] Jean-Marc Daigle. Les maladies du coeur et les maladies vasculaires cerebrales. 2007.
- [44] Li Deng. Three classes of deep learning architectures and their applications : a tutorial survey. page 58, 2012.
- [45] Ankita Dewan and Meghna Sharma. *Prediction of heart disease using a hybrid technique in data mining classification*. 2015.
- [46] Issam El Naqa and Martin J Murphy. Machine and deep learning in oncology, medical physics and radiology. 2022.
- [47] DJADJA et DJIAB and SARRA et DJIHAD. Convolutional neural networks-based model architecture for signal processing applications. 2020.
- [48] Anna Fyrenius, Lars Wigström, Tino Ebbers, Matts Karlsson, Jan Engvall, and Ann F Bolger. *Three dimensional flow in the human left atrium*. 2001.
- [49] Morten Grønabæk, Ulrik Becker, Ditte Johansen, Adam Gottschau, Peter Schnohr, Hans Ole Hein, Gorm Jensen, and Thorkild IA Sørensen. *Type of alcohol consumed and mortality from all causes, coronary heart disease, and cancer*. 2000.
- [50] Caleb Hernandez, Klaus Shuler, Hashibul Hannan, Chionesu Sonyika, Antonios Likourezos, and John Marshall. *CAUSE : cardiac arrest ultra-sound exam—a better approach to managing patients in primary non-arrhythmogenic cardiac arrest*. 2008.
- [51] Soufiane Hourri. Étude des techniques de deep learning en reconnaissance automatique du locuteur. 2021.
- [52] Ali Isin and Selen Ozdalili. *Cardiac arrhythmia detection using deep learning*. 2017.
- [53] Nobuyuki Kagiya, Sirish Shrestha, Peter D Farjo, and Partho P Sengupta. *Artificial intelligence : practical primer for clinical research in cardiovascular disease*. 2019.
- [54] M Kavitha, G Gnaneswar, R Dinesh, Y Rohith Sai, and R Sai Suraj. *Heart disease prediction using hybrid machine learning model*. 2021.
- [55] Serkan Kiranyaz, Onur Avcı, Osama Abdeljaber, Turker Ince, Moncef Gabbouj, and Daniel J Inman. 1d convolutional neural networks and applications : A survey. page 107398, 2021.

- [56] Chayakrit Krittanawong, Kipp W Johnson, Robert S Rosenson, Zhen Wang, Mehmet Aydar, Usman Baber, James K Min, WH Wilson Tang, Jonathan L Halperin, and Sanjiv M Narayan. *Deep learning for cardiovascular medicine : a practical primer*. 2019.
- [57] Louise IT Lee, Senthoran Kanthasamy, Radha S Ayyalaraju, and Rakesh Ganatra. *The current state of artificial intelligence in medical imaging and nuclear medicine*. 2019.
- [58] Guy Lerebours. Le rythme sinusal-mécanisme et fonction. *médecine/sciences*, pages 657–662, 2007.
- [59] Ruizhe Li, Xiao Li, Chenghua Lin, Matthew Collinson, and Rui Mao. A stable variational autoencoder for text modelling. 2019.
- [60] Awais Mehmood, Munwar Iqbal, Zahid Mehmood, Aun Irtaza, Marriam Nawaz, Tahira Nazir, and Momina Masood. *Prediction of heart disease using deep convolutional neural networks*. 2021.
- [61] Sarah Ménétré. Analyse de signaux d’arrêts cardiaques en cas d’intervention d’urgence avec défibrillateur automatisé : optimisation des temps de pause péri-choc et prédiction d’efficacité de défibrillation. 2011.
- [62] Tayeb Mhamed and Saoudi Yasmina. Etude comparative de solution de deep learning de classification d’images médicales. 2019.
- [63] Zakarya MIMOUNE, Abdelwhab OUAHAB, et al. Développement d’une architecture basée sur l’apprentissage profond (deep learning) pour la détection d’intrusion dans les réseaux. 2019.
- [64] Mohamed NEMICHE. Master masi.
- [65] Subhash Chandra Pandey. *Data mining techniques for medical data : a review*. 2016.
- [66] Seong Hyeon Park, ByeongDo Kim, Chang Mook Kang, Chung Choo Chung, and Jun Won Choi. Sequence-to-sequence prediction of vehicle trajectory via lstm encoder-decoder architecture. pages 1672–1678, 2018.
- [67] Bukola Salami. A deep learning (neural network) approach for ecg heartbeat classification : Arrhythmia detection. 2020.
- [68] Senani Samy. Réseaux de neurones convolutionnels pour la détection précoce de la rétinopathie diabétique. 2019.
- [69] Mohit Sewak, Sanjay K Sahay, and Hemant Rathore. An overview of deep learning architecture of deep neural networks and autoencoders. pages 182–188, 2020.
- [70] Aarti Sharma, Rahul Sharma, Vivek Kr Sharma, and Vishal Shrivatava. *Application of data mining—a survey paper*. 2014.
- [71] K Sowjanya and G Krishna Mohan. Predicting heart disease using machine learning classification algorithms and along with tpot (automl). *International Journal of Scientific and Technology Research*, pages 3202–3210, 2020.
- [72] Edit Tanai and Stefan Frantz. Pathophysiology of heart failure. *Comprehensive physiology*, pages 187–214, 2011.
- [73] SA Thorne. *Pregnancy in heart disease*. 2004.
- [74] Youssef Trardi. *Analyse multivariée pour le diagnostic de l’arythmie cardiaque*. 2020.
- [75] Guido Van Rossum and Fred L Drake Jr. *Python tutorial*. 1995.

- [76] Michael Wainberg, Daniele Merico, Andrew Delong, and Brendan J Frey. *Deep learning in biomedicine*. 2018.
- [77] Simon LF Walsh, Lucio Calandriello, Mario Silva, and Nicola Sverzellati. *Deep learning for classifying fibrotic lung disease on high-resolution computed tomography : a case-cohort study*. Elsevier, 2018.
- [78] Ruoqi Wei, Cesar Garcia, Ahmed El-Sayed, Viyaleta Peterson, and Ausif Mahmood. Variations in variational autoencoders-a comparative evaluation. pages 153651–153670, 2020.
- [79] Anthony J Weinhaus and Kenneth P Roberts. *Anatomy of the human heart*. 2005.
- [80] Robert H Whitaker. Anatomy of the heart. *Medicine*, pages 333–335, 2010.
- [81] Jun Xu, Lei Xiang, Renlong Hang, and Jianzhong Wu. Stacked sparse autoencoder (ssae) based framework for nuclei patch classification on breast cancer histopathology. pages 999–1002, 2014.
- [82] Rikiya Yamashita, Mizuho Nishio, Richard Kinh Gian Do, and Kaori Togashi. Convolutional neural networks : an overview and application in radiology. *Insights into imaging*, pages 611–629, 2018.
- [83] Zibin Zheng, Yatao Yang, Xiangdong Niu, Hong-Ning Dai, and Yuren Zhou. Wide and deep convolutional neural networks for electricity-theft detection to secure smart grids. *IEEE Transactions on Industrial Informatics*, pages 1606–1615, 2017.