

---

République Algérienne Démocratique et Populaire  
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique  
Université 20 Août 1955- Skikda  
Faculté des Sciences  
Département d'informatique



## **Mémoire**

en vue de l'obtention du Diplôme de Master en Informatique

**Option : Systèmes d'information Avancés et Applications**

## **Thème**

**Apprentissage profond pour la prédiction des maladies  
cardiovasculaires à partir des signaux ECG.**

### **Réalisé par :**

- DENHADJI Nesrine
- GRINE Hanna

### **Devant le jury:**

- DR BOUGUEROUA SALAH   PRESIDENT
- DR BOULENMOUR Imene   EXAMINATRICE
- DR BOUGAMOUZA Fateh   ENCADREUR

Année Universitaire 2022-2023

---

# Remerciement

*On remercie dieu le tout puissant de nous avoir donné la santé et la volonté d'entamer et terminer ce mémoire.*

*Nous tenons à exprimer nos remerciements les plus vifs à notre encadreur, Dr. BOUGAMOUZA Fateh, qui a su nous guider et nous aider dans ce travail avec beaucoup de tact et de gentillesse, et qui nous a permis de découvrir un domaine très intéressant. Qu'il trouve ici notre estime et notre profond respect.*

*Nous tenons également à remercier toutes les personnes qui ont participé, à titre professionnel ou personnel, à la réalisation de ce travail.*

*Nos remerciements vont également à tous ceux qui ont accepté avec bienveillance de participer au jury de ce mémoire.*

---

# Dédicaces

*Avec tout respect et amour je dédie ce modeste travail :*

*À mon chère père*

*Bien que tu ne sois plus physiquement présent, je sens ta force et ton soutien qui continuent de m'entourer. Tu resteras à jamais gravé dans mon cœur et dans mes souvenirs.*

*À ma chère mère*

*Pour exprimer ma gratitude envers votre amour inconditionnel, votre soutien indéfectible, vos encouragements constants et vos innombrables sacrifices.*

*À mes chères sœurs et chers frères*

*Ilhem ,Zahra,Adel et Rostom*

*Pour leurs encouragements , amour , conseils et leur soutien tout au long de mes études. Que Dieu les protège et leur accorde succès et bonheur.*

*À mes chères, Neveux et nièces*

*Rima ,Anas ,Sabrine, Assil , Amir*

*Vous êtes les petites personnes qui la remplissent ma vie de joie et de bonheur.*

*À mon binôme, Hanna*

*Pour sa entant et sa sympathie*

*À mes chères amis, Halima & Doha*

*Pour leurs aides et supports dans les moment difficiles*

*À mes collègues, À mes chères cousines ainsi qu'à toutes les personnes de ma famille.*

*Enfin, je dédie ce travail à moi-même, en reconnaissant mes efforts, ma persévérance et ma détermination à franchir les obstacles pour atteindre mes objectifs académiques.*

***Nesrine***

---

# Dédicaces

*Je tiens à dédier humblement ce travail :*

À mes très chers parents-qui ont toujours été présents à mes côtés. Leur soutien constant, leurs encouragements et leurs sacrifices pour m'offrir une éducation ont été d'une valeur inestimable. Je suis profondément reconnaissante envers eux pour leur amour inconditionnel et leur dévouement sans faille.

À ma précieuse sœur,Nadjeh Je me souviens de tous ces moments où elle a été là pour m'écouter avec attention.Elle a toujours su trouver les mots justes pour me prodiguer des conseils précieux et m'encourager.

À mes chers frères, mes belles-sœurs et mon beau-frère, pour leurs encouragements constants.

À mes adorables neveux et nièces,qui savent toujours comment procurer la joie et le bonheur pour toute la famille.

À ma meilleure amie Ines, qui a été généreuse avec son amour et son encouragement malgré la distance qui nous sépare.

À mon binôme ,Nesrine

À tous ceux qui m'aiment et que j'aime.

***Hanna***

# ملخص

في هذه الدراسة، قمنا بتطوير نظام مكون من مرحلتين لتصنيف إشارات القلب، مع التركيز على كشف الاضطرابات النظمية. تتمحور المرحلة الأولى حول استخراج سمات الإشارات، وذلك باستخدام نمذجة الذات الراجعة. تساعد هذه المرحلة في تحويل الإشارات إلى معلومات ذات دلالة. في المرحلة الثانية، نستخدم شبكات عصبونية، بما في ذلك خوارزمية التحويل التكاملي المنزلق (CNN) لتنفيذ تصنيف الإشارات.

وإجراء اختبارات متغيرة في حجم "MIT-BIH Arrhythmia" نقوم بتقييم أداء نظامنا باستخدام قاعدة بيانات نوافذ الوقت وعدد معاملات الانحدار الذاتي. تتيح لنا هذه الطريقة الاستفادة الكاملة من المعلومات الموجودة في إشارات القلب والحصول على تصنيف دقيق. في هذه الدراسة، استخدم خوارزمية التحويل التكاملي المنزلق. سمح لنا بتحقيق دقة بنسبة 98.6%. هذا التقني في التعلم الآلي أثبتت قدرتها على التعلم من النماذج المعقدة وتصنيف إشارات القلب بدقة وفعالية.

الكلمات الرئيسية: اضطرابات النظمية، التعلم العميق، شبكات عصبونية، تصنيف الإشارات، قاعدة بيانات "MIT-BIH Arrhythmia"، ECG، CNN.

---

# Résumé

Dans cette étude, nous avons mis en place un système en deux phases pour classifier les signaux cardiaques, en se concentrant sur la détection des arythmies. La première phase consiste en l'extraction des caractéristiques des signaux, réalisée à l'aide de la modélisation auto-régressive. Cette étape permet de traduire les signaux en informations significatives. Dans la deuxième phase, nous utilisons des réseaux de neurones, notamment l'algorithme de convolution (CNN), pour effectuer la classification des signaux.

Nous évaluons la performance de notre système en utilisant la base de données "MIT-BIH Arrhythmia" et en effectuant des tests en variant les tailles des fenêtres temporelles et le nombre de coefficients d'auto-régression. Cette approche nous permet d'exploiter pleinement les informations contenues dans les signaux cardiaques et d'obtenir une classification précise. Dans cette étude, l'utilisation de l'algorithme de convolution (CNN) nous a permis d'atteindre une précision de 98.6%. Cette technique d'apprentissage automatique a démontré sa capacité à apprendre des modèles complexes et à classifier les signaux cardiaques de manière précise et efficace.

**Mots clés:** Arythmies, Apprentissage profond, réseaux de neurones, classification des signaux, base de données "MIT-BIH Arrhythmia", CNN, ECG.

---

# Abstract

In this study, we have implemented a two-phase system to classify cardiac signals, focusing on the detection of arrhythmias. The first phase involves extracting features from the signals using autoregressive modeling. This step translates the signals into meaningful information. In the second phase, we employ neural networks, specifically the convolutional neural network (CNN) algorithm, to perform signal classification.

We evaluate the performance of our system using the "MIT-BIH Arrhythmia" database and conduct tests by varying the sizes of temporal windows and the number of autoregressive coefficients. This approach allows us to fully leverage the information contained in the cardiac signals and achieve accurate classification. In this study, the use of the convolutional neural network (CNN) algorithm has enabled us to achieve an accuracy of 98.6%. This machine learning technique has demonstrated its ability to learn complex patterns and classify cardiac signals accurately and effectively.

**Keywords:** Arrhythmias, Deep learning, Neural networks, Signal classification, "MIT-BIH Arrhythmia" database, CNN, ECG

---

## Table des matières

Liste des Tableaux .....	10
Liste Abréviation .....	11
Introduction générale .....	12
1. Contexte .....	12
2. Problématique et objectifs .....	12
Chapitre1:Le signal Electrocardiogramme (ECG).....	14
1. Introduction .....	15
2. Le fonctionnement du cœur .....	15
3. Anatomie du cœur .....	15
3.1. Les couches du cœur .....	16
3.2. L'activité électrique du cœur .....	16
3.2.1. Le rythme cardiaque .....	16
3.2.2. Le circuit électrique .....	17
4. L'électrocardiographie .....	18
4.1. Électrocardiographe .....	18
4.2. L'électrocardiogramme .....	18
4.3. Les ondes du signal ECG .....	18
4.4. Les segments et intervalles qui caractérisent un ECG normal .....	19
5. Dérivations .....	20
5.1. Dérivations dans le plan frontal .....	20
5.1.1. Les dérivations bipolaires .....	20
5.1.2. Les dérivations unipolaires .....	20
5.2. Dérivations dans le plan horizontal .....	21
6. Différentes pathologies cardiaques .....	22
6.1. Le rythme sinusal .....	22
6.2. Les arythmies cardiaques .....	23
7. Conclusion .....	24
Chapitre02:L'apprentissage automatique et l'apprentissage profond .....	25
1. Introduction .....	26
2. L'apprentissage Automatique .....	26
2.1. Définition .....	26
2.2. Les types d'apprentissage automatique .....	27
2.2.1. Apprentissage supervisé .....	27

---

2.2.2. Apprentissage non supervisé .....	28
2.2.3. L'apprentissage semi supervisée .....	28
2.2.4. L'apprentissage par renforcement .....	28
2.3. Les Algorithmes d'apprentissage .....	29
3. L'apprentissage Profond .....	30
3.1. Définition .....	30
3.2. Les réseaux de neurones artificiels «RNA» .....	31
3.2.1. Le neurone biologique .....	32
3.2.2. Le neurone formel .....	32
3.3. Les différentes architectures d'apprentissage profond .....	33
3.3.2. Les réseaux de neurones profonds .....	34
4. Quelques systèmes existants de prediction des maladies cardiovasculaires .....	39
5. Conclusion .....	40
Chapitre 3: Système réalisé .....	42
1. Introduction .....	43
2. L'objectif et la motivation .....	43
3. Architecture générale .....	43
4. Présentation détaillée du système .....	44
4.1. Base de données utilisée .....	45
1. Exemple d'un ECG normal et anormal .....	46
4.2. Prétraitement des données .....	47
4.3. Choix du modèle .....	48
4.4. Etape de classification .....	49
5. Résultat et discussion .....	54
5.1. Évolution des performances .....	57
5.1.1. Les courbes de perte .....	57
5.1.2. La matrice de confusion .....	58
5.2. L'analyse comparative des systèmes de prédiction des maladies cardiaques ....	60
6. Conclusion .....	61
Conclusion générale .....	62
1. Bilan .....	62
2. Perspectives .....	62
Bibliographie .....	64
Webographie .....	69

---

Annexe :Outils d’implimentation du système .....	70
1. Introduction .....	71
2. Présentation des outils utilisés .....	71
2.1. Google collab .....	71
2.2. Google drive .....	71
2.3. Python .....	72
2.4. Pytorch .....	73
2.5. Pandas .....	73
2.6. Numpy .....	74
2.8. Matplotlib .....	75
2.9. Seaborn .....	75
3. Etapes d’implémentation .....	75
3.1. Importation des bibliothèques et chargement de l’ensemble de données .....	75
3.2. Extraction caractéristiques et Labels .....	76
3.3. Prétraitement des données .....	77
3.4. L’architecture du CNN utilisé .....	78
3.5. Apprentissage .....	79
3.6. Évaluation du modèle .....	80
4. Conclusion .....	81

---

# Liste des Tableaux

Table 1 :ajustement du nombre de couches dans un modèle. ....	54
Table 2 :Comparaison des performances en utilisant les fonctions d'activation GELU et ReLU .....	55
Table 3 :l'évaluation de l'impact de la normalisation des données. ....	55
Table 4 :les résultats obtenus en utilisant deux types de couches de pooling. ....	56
Table 5 :Résultats expérimentaux pour déterminer le taux de Dropout. ....	57
Table 6 :Tableau récapitulatif des précisions globales. ....	60

---

# Liste Abréviation

IA	Intelligence artificielle
ECG	Electrocardiogramme
PVC	Premature Ventricular Contraction
APC	Atrial Premature Contraction
RBB	Right Bundle Branch
PAB	Premature Atrial Beat
ML	Machine Learning
DL	Deep Learning
KNN	K-Nearest Neighbors
SVM	Support Vector Machines
RNA	Réseau de Neurones Artificiels
PMC	Perceptron Multicouche
RNN	Réseau de Neurones Récurrents
CNN	Réseau de Neurones Convolutifs
LR	Learning Rate

# Introduction générale

## 1. Contexte

Actuellement, le secteur médical nécessite l'utilisation de nouvelles techniques et technologies afin d'évaluer l'information de manière objective. Cela est dû aux récents développements dans le domaine de l'électronique, qui ont poussé l'informatique à un niveau avancé. Ces avancées ont permis d'avoir des machines de plus en plus performantes capables d'exécuter des algorithmes complexes et d'explorer de nouvelles approches de l'intelligence artificielle (IA), ce qui était auparavant impossible.

La médecine, tout comme de nombreux autres domaines, a bénéficié de cette révolution informatique, en particulier de l'IA. Dans ce contexte, l'informatique est devenue un outil indispensable dans la pratique médicale moderne en général, et plus particulièrement en tant que support d'aide au diagnostic. De nombreuses techniques d'IA telles que la logique floue, les réseaux de neurones et les algorithmes génétiques sont couramment utilisées dans les applications médicales afin d'améliorer les performances des systèmes d'aide au diagnostic médical.

Les maladies cardiaques sont l'une des principales causes de décès dans le monde aujourd'hui, et leur diagnostic revêt une importance vitale. Les services de cardiologie des hôpitaux utilisent largement l'électrocardiogramme (ECG) pour analyser et diagnostiquer les troubles cardiaques, notamment les arythmies. L'ECG fournit une représentation graphique des signaux électriques générés par le cœur, en détectant les variations du potentiel électrique à la surface du corps. La classification précise des signaux ECG joue un rôle crucial en offrant des avantages cliniques significatifs et en facilitant une prise en charge adaptée pour les patients.

## 2. Problématique et objectifs

Dans le monde, la plupart des patients décèdent en raison de la découverte de leur maladie à un stade avancé en raison du manque de précision des outils utilisés ou de l'absence d'un cardiologue dans l'hôpital pour interpréter les résultats des tests et diagnostiquer la maladie. Les maladies cardiovasculaires sont considérées comme un défi critique dans le domaine de l'analyse des données cliniques.

Notre objectif est de développer un système de prédiction précis des maladies cardiovasculaires en utilisant des techniques avancées d'apprentissage profond. En analysant les données médicales pertinentes, nous visons à identifier les individus à haut risque de

développer ces maladies, ce qui permettrait une détection précoce et une prise en charge adaptée pour réduire les taux de mortalité.

Cet objectif général peut être détaillé dans les sous-objectifs spécifiques suivants :

- Concevoir un modèle CNN pour reconnaître l'état normal du cœur et détecter quatre types d'arythmies à partir des signaux électrocardiogramme (ECG) en entrée.
- Une étude comparative visant à démontrer l'efficacité des réseaux de neurones convolutifs (CNN) dans le développement de systèmes de détection des maladies cardiovasculaires.

### **3. Plan de mémoire**

Ce mémoire est structuré en trois chapitres principaux, ainsi qu'une introduction, une conclusion générale et une annexe.

- Le premier chapitre, présente des généralités sur les maladies cardiaques.
- Dans le deuxième chapitre, nous avons abordé l'apprentissage automatique et l'apprentissage profond, en définissant leurs différents types et en examinant les algorithmes associés à la création de modèles.
- Le troisième chapitre présente en détail notre proposition (solution), en décrivant les différentes étapes essentielles que nous avons suivies. Ce chapitre comprend également une discussion approfondie des résultats obtenus.
- L'annexe présente les outils utilisés et décrit l'implémentation spécifique du système.

# Chapitre1:Le signal Electrocardiogramme (ECG)

## **1. Introduction**

Le cœur est un organe vital qui assure la circulation sanguine en pompant le sang à travers les vaisseaux sanguins. Pour diagnostiquer les problèmes cardiaques, les professionnels de santé utilisent des outils tels que l'électrocardiogramme (ECG), un test non-invasif qui mesure l'activité électrique du cœur [1]. L'ECG peut révéler des informations importantes telles que la fréquence cardiaque, le rythme cardiaque et la présence d'anomalies électriques. Alors dans ce chapitre on va répondre aux questions suivantes , quel est l'électrocardiogramme ? Comment est-il testé ? Quels sont les éléments les plus importants ?

## **2. Le fonctionnement du cœur**

Le cœur est un muscle vital qui fonctionne comme une pompe en contractant rythmiquement pour faire circuler le sang dans tout le corps. Il est capable de pomper en permanence une quantité de sang allant jusqu'à 4 à 5 litres depuis la naissance jusqu'à la mort. Chaque jour, le cœur effectue en moyenne 100 000 contractions, pompe 8 000 litres de sang, et fait en moyenne 2 milliards de battements au cours d'une vie [web 1].

## **3. Anatomie du cœur**

Le cœur est un muscle de la taille d'un poing situé dans le thorax, dont le rôle est celui d'une pompe à sang. Il est constitué de 4 cavités : les oreillettes gauche et droite forment la partie supérieure du cœur, tandis que les ventricules gauche et droit correspondent à la partie inférieure. Ces 4 cavités communiquent entre elles à travers des ouvertures appelées valves, qui assurent la circulation du sang dans une seule direction. En effet, l'oreillette droite reçoit le sang désoxygéné du reste du corps à travers les veines caves supérieure et inférieure, et l'injecte dans le ventricule droit en traversant la valve tricuspide. De l'autre côté, l'oreillette gauche reçoit le sang oxygéné des poumons à travers les veines pulmonaires, puis l'envoie vers le ventricule gauche en passant par la valve mitrale. Ensuite, le sang est pompé du ventricule droit vers les poumons à travers les artères pulmonaires gauche et droite, dont le flux est contrôlé par la valve pulmonaire. De même, le sang est pompé du ventricule gauche vers le reste du corps en passant par l'aorte à travers la valve aortique[2]. La Figure 1 présente une illustration détaillée de l'anatomie du cœur et des vaisseaux associés.

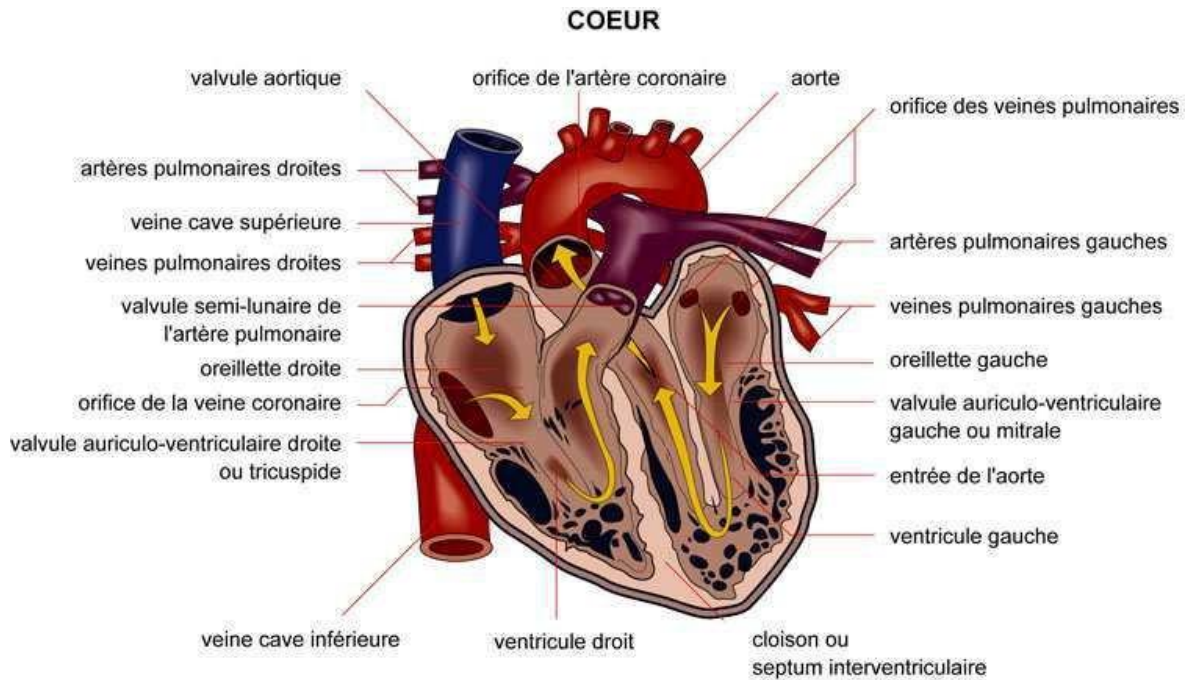


Figure 1:anatomie du cœur et des vaisseaux associés [ 3].

### 3.1. Les couches du cœur

Le cœur est composé de trois couches distinctes, chacune ayant un rôle spécifique dans la fonction cardiaque. La couche externe est fine et sert à protéger le cœur contre les frottements lorsqu'il se contracte. La couche intermédiaire, qui est épaisse, est responsable du pompage cardiaque et se contracte régulièrement. La couche interne, quant à elle, est la couche intérieure de la paroi cardiaque. Ensemble, ces trois couches permettent au cœur de se contracter et de se dilater efficacement [Web2].

### 3.2. L'activité électrique du cœur

#### 3.2.1. Le rythme cardiaque

Le rythme cardiaque est constitué de deux éléments distincts :

Une composante mécanique, représentée par le cycle cardiaque, qui comprend les différentes étapes de contraction (systole) pour propulser le sang hors du ventricule gauche, ainsi que les phases de relaxation (diastole) qui permettent le remplissage de la cavité cardiaque.

Une composante électrique qui joue un rôle essentiel dans la coordination et la synchronisation de la phase mécanique du rythme cardiaque. Ces deux composantes travaillent en parfaite harmonie pour assurer le bon fonctionnement du cœur[Web3].

### 3.2.2. Le circuit électrique

Le fonctionnement électrique du cœur est essentiel pour assurer son rythme cardiaque régulier et coordonné.

Le système de conduction cardiaque se compose des éléments suivants : le nœud sinusal (ou nœud SA), le nœud auriculo-ventriculaire (nœud AV), le faisceau de Hys, les branches du faisceau et les fibres de Purkinje. Le nœud SA, considéré comme le stimulateur anatomique, initie le processus en déclenchant la contraction des muscles auriculaires. Le signal électrique se propage ensuite du nœud SA vers le nœud AV à travers le faisceau de Hys, puis se divise dans les branches du faisceau et se diffuse dans les fibres de Purkinje, ce qui entraîne la contraction des ventricules. Ce signal électrique génère un courant qui peut être enregistré et visualisé sur un graphique appelé électrocardiogramme (ECG). Les médecins utilisent l'ECG pour surveiller l'activité électrique du système de conduction cardiaque dans le cœur [4]. La Figure 2 illustre de manière détaillée le courant électrique du cœur.

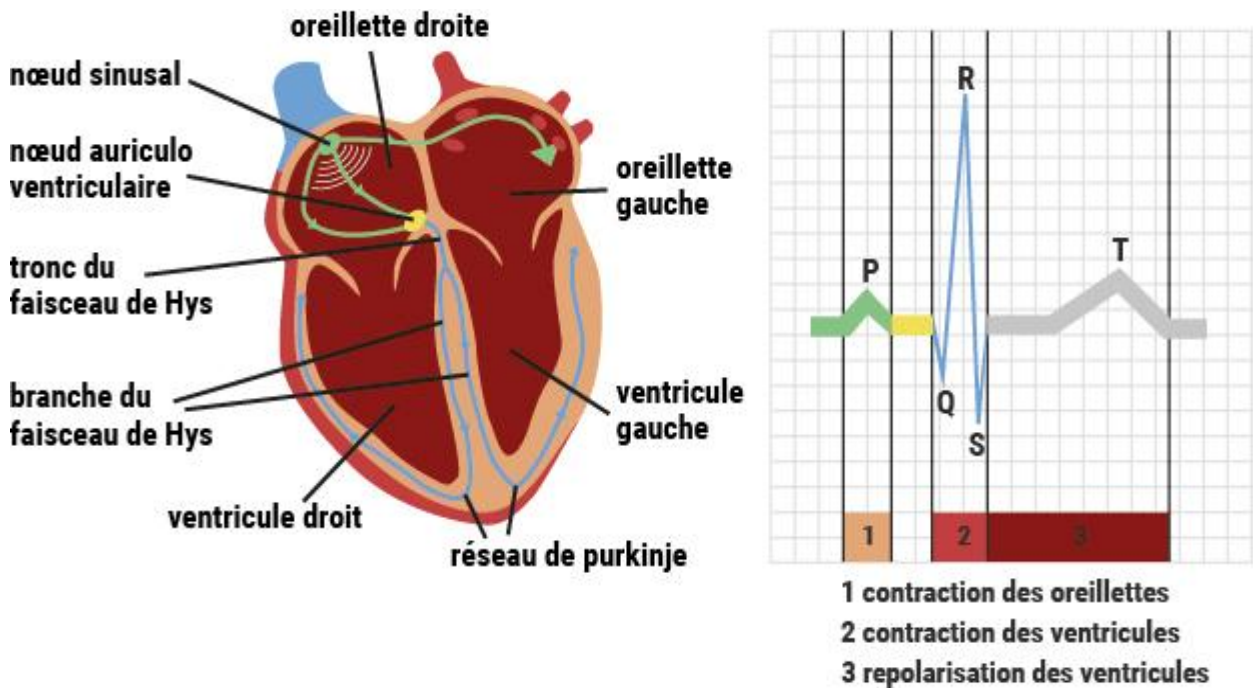


Figure 2:le courant électrique du cœur[Web3].

**Le faisceau de Hys** : long de un à deux centimètres, c'est un centre d'automatisme secondaire. Non seulement il propage l'influx de l'étage auriculaire à l'étage ventriculaire mais aussi, il est capable de décharger spontanément des impulsions de 40 à 60 battements par minute [5].

## 4. L'électrocardiographie

### 4.1. Électrocardiographie

Un électrocardiographe est un appareil qui mesure et enregistre l'activité électrique qui traverse le cœur [6] :

- Le tracé de cet enregistrement s'appelle un électrocardiogramme (souvent noté ECG).
- Cet appareil a été inventé par Willem Einthoven en 1895 (Prix Nobel en 1924).
- L'électrocardiogramme est aujourd'hui l'examen le plus courant pour étudier le fonctionnement du cœur et pour déceler les troubles cardiaques.

### 4.2. L'électrocardiogramme

L'examen ECG est un outil non invasif effectué par le médecin en vue d'explorer le fonctionnement du cœur par l'emploi des électrodes externes mises en contact de la peau. Il s'agit d'un signal qui reflète l'activité électrique du cœur. Les signaux électrocardiogrammes d'ECG peuvent être vus comme la périodisation d'un motif composé de plusieurs ondes successives notées P, QRS et T. L'analyse de ces ondes permet le diagnostic de certaines pathologies cardiaques et leur détection et caractérisation s'avèrent importantes[7].

### 4.3. Les ondes du signal ECG

Le signal ECG est composé de lettres de l'alphabet, telles que "P, Q, R, S et T", qui représentent l'activité électrique des différentes parties du cœur. Ces lettres sont associées à des ondes spécifiques qui indiquent la morphologie et l'amplitude des signaux électriques générés par le cœur. Toutefois, la durée de ces ondes peut varier en fonction des dérivations ECG utilisées[8].

**L'onde P** : Elle représente la dépolarisation auriculaire, se produisant avant le complexe QRS. Il s'agit d'une petite déviation positive (ou négative) qui se propage à partir du nœud SA et se propage dans les cellules de l'oreillette par des jonctions intermédiaires.

**Le complexe QRS** : Il est composé de déviations positives et négatives qui correspondent à la contraction des ventricules. Dans des conditions normales, sa durée est inférieure à 0,12 seconde et son amplitude varie entre 5 et 20 milli Volte.

- ◆ **L'onde Q** : C'est la première déviation initiale vers le bas ou "négative" du complexe QRS.

- ◆ **L'onde R** : C'est la déviation suivante vers le haut du complexe QRS, à condition qu'elle traverse la ligne isoélectrique et devienne "positive".
- ◆ **L'onde S** : C'est la déviation suivante vers le bas du complexe QRS, à condition qu'elle traverse la ligne isoélectrique et devienne brièvement négative avant de revenir à la ligne de base isoélectrique.
- ◆ **L'onde T** : Elle correspond à la repolarisation des ventricules. L'onde T normale a une amplitude plus faible que le complexe QRS .

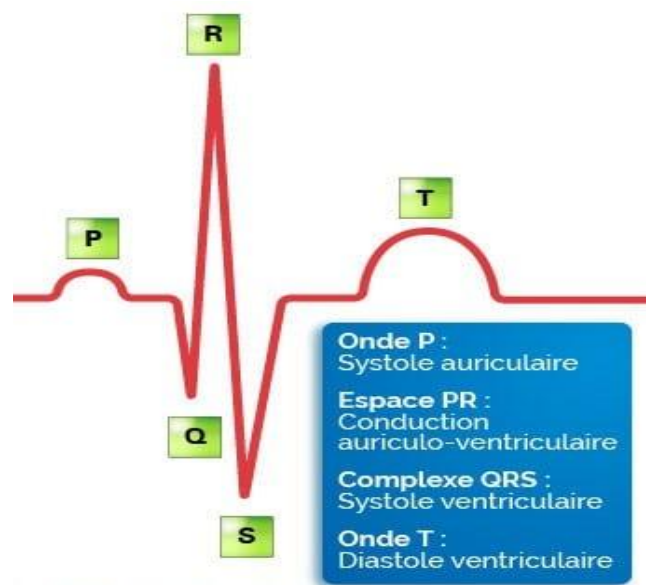


Figure 3: Les ondes du signal ECG [8].

#### 4.4. Les segments et intervalles qui caractérisent un ECG normal

En plus des différentes ondes, il existe certains intervalles et segments dans un signal ECG qui fournissent des informations importantes sur la vitesse de conduction de l'impulsion électrique dans les différentes parties du cœur. Ces intervalles et segments sont essentiels pour la caractérisation complète d'un ECG[9].

**Intervalle RR** : C'est l'intervalle de temps entre deux dépolarisations successives des ventricules. Il est utilisé pour calculer la fréquence cardiaque.

**Segment PR** : Il représente la pause au niveau du nœud auriculo-ventriculaire (AV) entre la fin de la dépolarisation des oreillettes et le début de la dépolarisation des ventricules. C'est le temps pendant lequel l'onde de dépolarisation est retardée au niveau du nœud AV.

Intervalle PR : Cet intervalle mesure la durée de propagation de l'onde de dépolarisation depuis le nœud sinusal jusqu'aux cellules myocardiques ventriculaires. Il indique le temps de conduction auriculo-ventriculaire.

Intervalle QT : Il représente la durée de la systole ventriculaire, c'est-à-dire le temps écoulé depuis le début de l'excitation des ventricules jusqu'à la fin de leur relaxation.

Segment ST : Ce segment correspond à la phase pendant laquelle toutes les cellules ventriculaires sont dépolarisées. Pendant cette phase, le segment est isoélectrique, c'est-à-dire qu'il présente une valeur de potentiel électrique constante .

## **5. Dérivations**

L'ECG enregistre l'activité électrique du cœur à partir de plusieurs points (dérivations). Les courbes obtenues varient en fonction de la position des électrodes. On utilise 12 dérivations de base, dont 6 sur le plan frontal et 6 sur le plan horizontal. Les dérivations bipolaires forment un triangle (triangle d'Einthoven) et les dérivations bipolaires et unipolaires dans le plan frontal constituent le système He axial de Bailey [10].

### **5.1. Dérivations dans le plan frontal**

#### **5.1.1. Les dérivations bipolaires**

Il y a six axes dans le plan frontal pour les dérivations bipolaires I, II et III, également connues sous le nom de dérivations d'Einthoven[11] .

La dérivation I enregistre les variations de potentiel électrique entre le poignet droit et le poignet gauche ;

La dérivation II enregistre les variations de potentiel électrique entre le poignet droit et la jambe gauche ;

La dérivation III enregistre les variations de potentiel électrique entre le poignet gauche et la jambe gauche .

#### **5.1.2. Les dérivations unipolaires**

Il y a également trois dérivations unipolaires dans le plan frontal connues sous le nom de dérivations de Wilson : aVR (pour "right", droite), aVL (pour "left", gauche) et aVF (pour "foot", pied).

La dérivation aVL enregistre les variations de potentiel électrique pour l'avant-bras gauche, tandis que la dérivation aVR enregistre les variations de potentiel électrique pour l'avant-bras droit. Enfin, la dérivation aVF enregistre les variations de potentiel électrique pour la jambe gauche[10]. La Figure 4 présente la position des électrodes pour les dérivations bipolaire et unipolaire utilisées dans l'électrocardiogramme (ECG), montrant leur placement spécifique sur le corps pour enregistrer l'activité électrique du cœur .

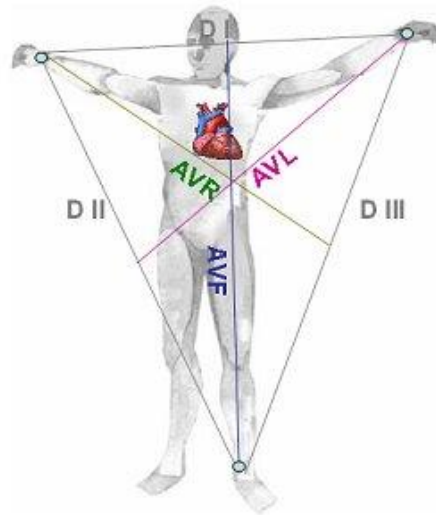


Figure 4:position des électrodes des dérivations bipolaire et unipolaire[11].

## 5.2. Dérivations dans le plan horizontal

Les dérivations du plan horizontal, également connues sous le nom de dérivations de Wilson, comprennent six dérivations unipolaires : V1 à V6. Ces dérivations présentent une polarité positive à l'endroit où l'électrode exploratrice est placée et une polarité négative dans la région opposée. L'électrode exploratrice est positionnée sur différents points précordiaux. En plus des dérivations V1 à V6, il est parfois possible d'utiliser des dérivations droites (V1R à V4R) ou gauches (V5 à V6) dans certaines situations [11]. La figure 5 montre les dérivations précordiales, des positions spécifiques des électrodes pour l'ECG.

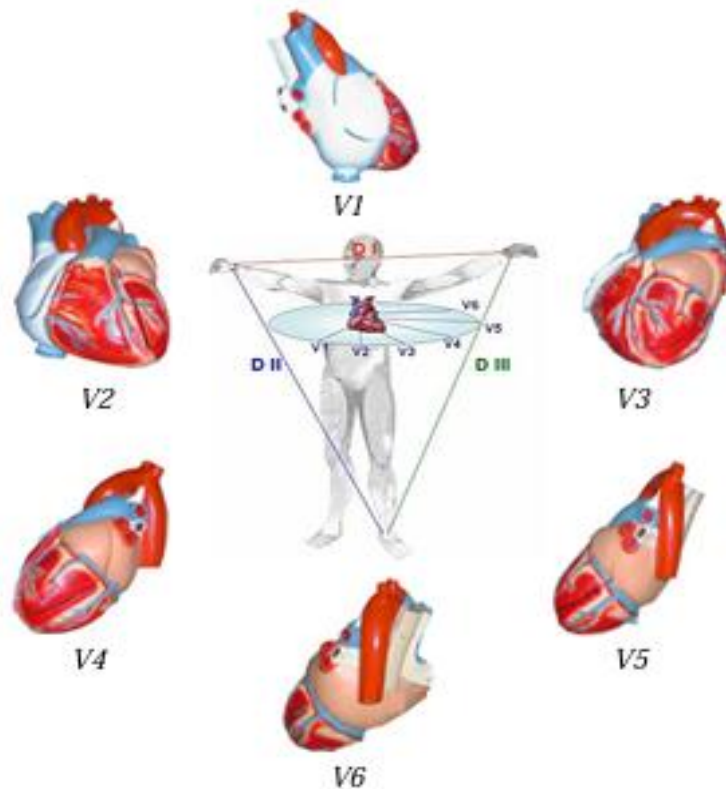


Figure 5:les dérivation précordiales[11].

## 6. Différentes pathologies cardiaques

Dans cette partie, notre objectif n'est pas d'effectuer une analyse approfondie des origines des pathologies cardiaques, de leurs conséquences sur le fonctionnement cardiaque, ou de décrire les traitements nécessaires pour ces pathologies. Nous nous concentrons plutôt sur la mise en relation de certaines observations anormales du tracé ECG avec ces pathologies. Parmi les maladies les plus courantes, on trouve les arythmies cardiaques, qui affectent le rythme cardiaque. Cependant, avant d'aborder les arythmies, il est important de comprendre les caractéristiques du rythme cardiaque normal, également connu sous le nom de rythme sinusal[12].

### 6.1. Le rythme sinusal

Le rythme sinusal est considéré comme le rythme cardiaque normal. Il est caractérisé par une activation physiologique des oreillettes, suivie des ventricules, qui débute au niveau du nœud sinusal. Le rythme sinusal se situe généralement entre 60 et 80 battements par minute, avec des intervalles réguliers entre chaque battement normal .

## **6.2. Les arythmies cardiaques**

Les arythmies sont des anomalies qui affectent le rythme cardiaque normal. Il existe plusieurs types d'arythmies, dont certains ne posent aucun problème de santé significatif, mais peuvent causer des symptômes gênants tels que des étourdissements ou des douleurs thoraciques. D'autres formes d'arythmies sont plus dangereuses car elles peuvent compromettre l'apport sanguin et nécessitent donc une prise en charge médicale appropriée[12]: .

Dans notre projet, nous allons nous concentrer sur l'utilisation de quatre types spécifiques d'arythmies:

### **6.2.1. Les contractions ventriculaires prématurées (PVC)**

Les PVC sont des battements cardiaques anormaux qui se produisent dans les ventricules du cœur. Elles résultent d'une irritabilité électrique du muscle cardiaque ventriculaire, ce qui perturbe le rythme cardiaque normal et peut provoquer une sensation de basculement ou de battement sauté dans la poitrine. Les PVC sont assez fréquents et peuvent survenir chez les personnes âgées, les personnes hypertendues et les personnes souffrant de maladies cardiaques .

### **6.2.2. La contraction auriculaire prématurée (APC)**

Les APC est une dépolarisation précoce qui se produit dans les oreillettes du cœur, généralement suivie d'une propagation aux ventricules, bien que cela ne se produise pas toujours. La morphologie de l'onde P lors d'une extrasystole auriculaire diffère de celle de l'onde P normale et dépend de l'emplacement du foyer auriculaire ectopique .

### **6.2.3. Bloc de branche droit (RBB)**

Lorsqu'un bloc de branche droite (RBB) est présent, le ventricule droit ne reçoit pas directement les impulsions électriques qui se propagent à travers la branche droite du faisceau de Hys. En revanche, le ventricule gauche est normalement activé par la branche gauche. Les impulsions électriques se propagent alors à travers le muscle cardiaque, du ventricule gauche vers le ventricule droit, pour finalement l'activer. La prévalence du bloc de branche droite tend à augmenter avec l'âge.

### **6.2.4. Le battement du rythme (Paced Beat : PAB)**

est généralement un signe de dysfonctionnement du muscle cardiaque. Il peut survenir à tout âge, mais est souvent le résultat d'un infarctus du myocarde. Le PAB se caractérise par une

désynchronisation des battements cardiaques, ce qui entraîne une diminution de l'efficacité de pompage sanguin . La Figure 6 présente les différents types de résultats d'un ECG (1 normal, 4 anormaux).

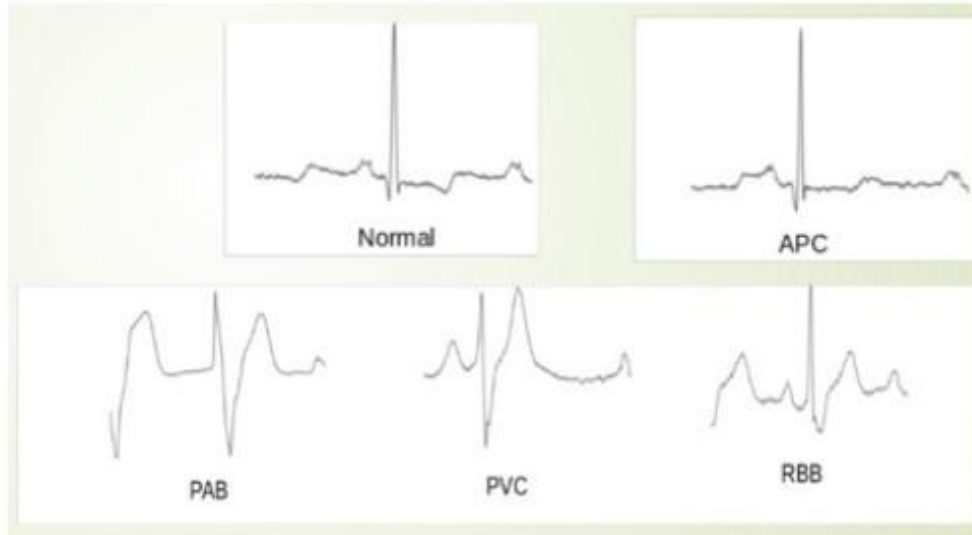


Figure 6:Les types des résultats ECG (1 normal, 4 anormaux)[12].

## 7. Conclusion

Dans ce chapitre, nous avons présenté une explication simplifiée du fonctionnement du muscle cardiaque et de la régulation électrique du cœur, ainsi que les méthodes de mesure de son activité. Nous avons également examiné le sens des différentes ondes qui indiquent l'état de santé du cœur.

# Chapitre02:L'apprentissage automatique et l'apprentissage profond

## 1. Introduction

L'apprentissage automatique est un domaine de l'informatique et de l'intelligence artificielle qui se concentre sur le développement d'algorithmes et de modèles capables de permettre aux ordinateurs d'apprendre et de s'améliorer à partir de l'expérience, sans être explicitement programmés. L'objectif de l'apprentissage automatique est de créer des modèles qui peuvent reconnaître des schémas dans les données et utiliser ces schémas pour faire des prévisions ou prendre des actions.

Les algorithmes d'apprentissage automatique sont capables de traiter de grandes quantités de données, d'extraire des modèles et des informations à partir de celles-ci, et d'appliquer ces connaissances à de nouveaux problèmes.

L'apprentissage automatique a de nombreuses applications pratiques, notamment la reconnaissance d'images, la reconnaissance de la parole, le traitement du langage naturel, la détection de fraudes, les systèmes de recommandation et la maintenance prédictive. Il est de plus en plus utilisé dans de nombreux domaines, tels que la finance, la santé, l'industrie et les technologies de l'information, pour aider à résoudre des problèmes complexes et améliorer les résultats.

## 2. L'apprentissage Automatique

### 2.1. Définition

L'apprentissage automatique est une technique de l'intelligence artificielle qui permet aux systèmes informatiques d'apprendre à partir de données et de s'améliorer sans être explicitement programmés à cet effet. Les programmes informatiques de l'apprentissage automatique sont conçus pour accéder à des données et pour utiliser ces données pour apprendre de manière autonome. Pour comprendre comment fonctionne l'apprentissage automatique, il est important de savoir comment formuler un problème d'apprentissage. Selon la définition de Tom Mitchell dans son livre "Machine Learning", un programme informatique apprend à partir de l'expérience  $E$  pour une classe de tâches  $T$  et un objectif de performance  $P$  lorsque ses performances sur les tâches de la classe  $T$ , évaluées par la mesure  $P$ , s'améliorent au fil de l'expérience  $E$ [13].

## 2.2. Les types d'apprentissage automatique

L'apprentissage automatique implique différents types d'apprentissage, en fonction de la structure du problème. Les plus courants dans les domaines d'application dominants sont l'apprentissage supervisé et non supervisé, ainsi que l'apprentissage semi-supervisé et par Renforcement

### 2.2.1. Apprentissage supervisé

L'apprentissage supervisé est une méthode d'apprentissage automatique où un algorithme est entraîné à partir d'un ensemble de données d'apprentissage pré-étiquetées. Les données d'apprentissage sont constituées d'exemples de cas réels traités et validés, où chaque exemple est composé d'un ensemble de variables explicatives (aussi appelées descripteurs ou caractéristiques) et d'une variable à prédire (aussi appelée étiquette ou label). L'objectif de l'apprentissage supervisé est d'inférer la relation entre les variables explicatives et la variable à prédire, afin de prédire avec précision la variable à prédire pour de nouvelles entrées pour lesquelles la variable à prédire est inconnue. L'algorithme d'apprentissage supervisé est généralement choisi en fonction de la nature des données et du type de variable à prédire. Les deux types d'algorithmes les plus couramment utilisés en apprentissage supervisé sont les algorithmes de régression et les algorithmes de classification.

Les algorithmes de **régression** sont utilisés pour prédire une variable continue, telle que le prix d'une maison ou le nombre de ventes d'un produit. Les algorithmes de **classification** quant à eux, sont utilisés pour prédire une variable catégorielle, telle que le type de fleur ou la catégorie de spam d'un courriel[14].

Par exemple :

- Prédire si un e-mail est un spam ( $y = 1$ ) ou non ( $y = 0$ ) selon le nombre de liens présent dans l'e-mail ( $x$ );
- Prédire si une tumeur est maligne ( $y = 1$ ) ou bénigne ( $y = 0$ ) selon la taille de la tumeur ( $x_1$ ) et l'âge du patient ( $x_2$ ).

Dans le cas d'un problème de classification, on représente souvent les classes par des symboles, plutôt que par leur valeur numérique (0, 1, ...). La Figure 7 illustre la Classification et la Régression.

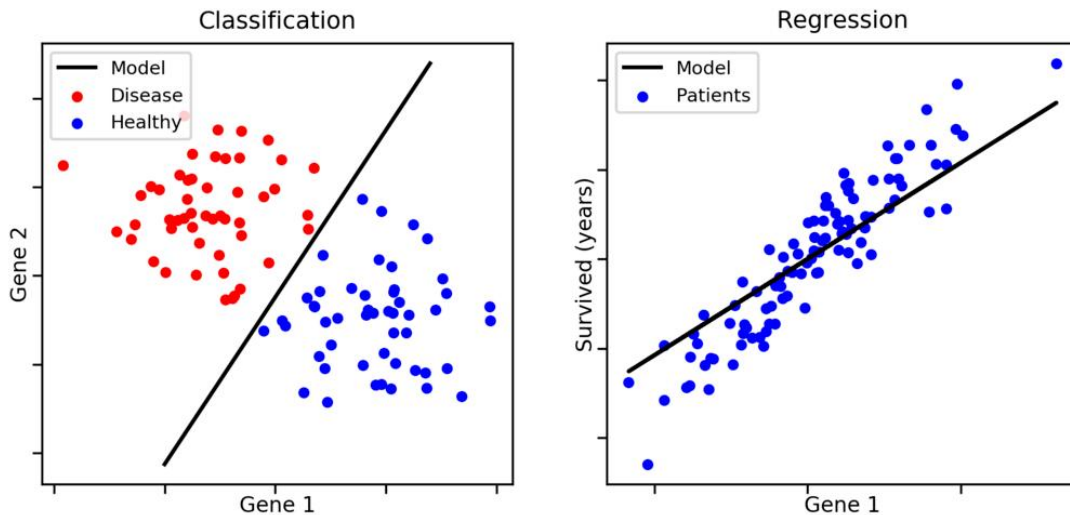


Figure 7:la Classification et la Régression[web4] .

### 2.2.2. Apprentissage non supervisé

l'apprentissage non supervisé, les données ne sont pas étiquetées, ce qui signifie qu'il n'y a pas de variable cible ou de réponse à prédire. Les algorithmes d'apprentissage non supervisé sont conçus pour trouver des modèles ou des structures cachées dans les données d'entrée, sans avoir de connaissances préalables sur les étiquettes ou les catégories.

Les algorithmes d'apprentissage non supervisé sont très utiles pour des applications où les données non étiquetées sont plus abondantes que les données étiquetées, comme dans la recherche d'information. Ils peuvent également être utilisés pour le clustering, qui est une technique qui consiste à regrouper des données similaires en groupes homogènes. Dans ce contexte, l'apprentissage non supervisé est utilisé pour apprendre les caractéristiques ou les propriétés des données brutes, ce qui permet de regrouper les données similaires[15].

### 2.2.3. L'apprentissage semi supervisée

une combinaison des deux précédentes méthodes (l'apprentissage supervisé et non supervisé). Il s'agit d'utiliser une petite quantité de données étiquetées conjointement avec une masse importante de données non-étiquetées.

### 2.2.4. L'apprentissage par renforcement

L'apprentissage par renforcement est une méthode d'apprentissage où un agent autonome doit apprendre à prendre les meilleures décisions possibles dans un environnement donné en

recevant des récompenses positives ou négatives pour chaque décision prise. L'objectif est de trouver une stratégie optimale qui maximise les récompenses obtenues.

Par exemple, dans un jeu de football comme FIFA, l'agent doit apprendre à prendre les bonnes décisions en fonction de la situation sur le terrain, comme faire une passe ou un tir, pour marquer un but et recevoir une récompense positive. L'agent doit alors trouver une stratégie qui maximise les récompenses obtenues tout en explorant de nouvelles options pour améliorer sa stratégie.

En résumé, l'apprentissage par renforcement consiste à trouver la meilleure stratégie pour maximiser les récompenses obtenues dans un environnement donné en explorant de nouvelles options pour améliorer la stratégie existante[16]. La Figure 8 offre une représentation de l'apprentissage par renforcement.

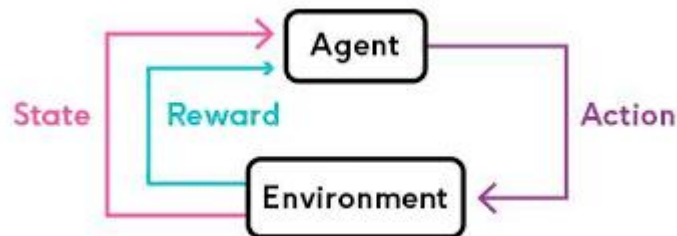


Figure 8:apprentissage par renforcement[web 5] .

### 2.3. Les Algorithmes d'apprentissage

Les algorithmes d'apprentissage automatique sont un ensemble de méthodes qui permettent à un ordinateur d'apprendre de nouveaux concepts à partir de données. Il y a plusieurs types d'algorithmes et ils peuvent être combinés pour créer différentes façons d'apprendre. Le choix de l'algorithme à utiliser dépendra de la tâche que l'on souhaite accomplir, comme classer des objets ou estimer des valeurs. Dans la suite, nous allons donner quelques exemples de ces algorithmes:

#### 2.3.1. Algorithme de plus proche voisin (KNN)

Plus connu en anglais sous le nom K-nearest neighbor (KNN). est une méthode d'apprentissage automatique qui ne nécessite pas la construction d'un modèle prédictif à partir des données d'entraînement. Au lieu de cela, il stocke simplement les données en mémoire et pour prédire la classe d'un nouveau cas, il recherche les K exemples les plus similaires dans

l'ensemble de données d'entraînement. La méthode utilise deux paramètres, le nombre K et une fonction de similarité, pour trouver les exemples les plus proches. La prédiction finale est basée sur la classe la plus fréquente parmi ces K exemples. Il est important de choisir des valeurs appropriées pour K et la fonction de similarité car cela peut affecter considérablement les résultats. Bien que l'apprentissage soit rapide car il ne nécessite pas la construction d'un modèle, la classification d'un nouveau cas peut être coûteuse car il faut comparer ce cas à tous les exemples stockés en mémoire[17].

### 2.3.2. Arbre de décision

Pour un usage général, les arbres de décision sont utilisés pour représenter visuellement les décisions et montrer ou éclairer la prise de décision. Lorsqu'on travaille avec l'apprentissage automatique et l'exploration de données, les arbres de décision sont utilisés comme modèle prédictif. Ces modèles cartographient les observations de données et tirent des conclusions sur la valeur cible des données.

L'objectif de l'apprentissage par arbre de décision est de créer un modèle qui prédira la valeur d'une cible en fonction de variables d'entrée. Dans le modèle prédictif, les attributs des données qui sont déterminés par l'observation sont représentés par les branches, tandis que les conclusions sur la valeur cible des données, sont représentées dans les feuilles [18].

### 2.3.3. Les machines à vecteurs de support (SVM)

Les SVMs ( Support Vector Machine ou Machine à vecteurs de support ), développés dans les années 1990, sont une famille d'algorithmes d'apprentissage automatique qui permettent de résoudre des problèmes de classification. Les SVMs ont pour but de séparer les données en classes à l'aide d'une frontière, de telle façon que la distance entre les différents groupes de données et la frontière qui les sépare soit maximale. Cette distance est aussi appelée « marge » et les SVMs sont ainsi qualifiés de « séparateurs à vaste marge », les « vecteurs de support » étant les données les plus proches de la frontière [19].

## 3. L'apprentissage Profond

### 3.1. Définition

le deep learning est une approche avancée de l'apprentissage automatique qui utilise des réseaux neuronaux artificiels profonds pour apprendre et effectuer des tâches complexes en

exploitant de grandes quantités de données. Il a révolutionné de nombreux domaines de l'IA et continue d'être une méthode de pointe pour résoudre des problèmes difficiles liés à la perception, à la compréhension et à la prise de décision [20]. La Figure 9 compare l'apprentissage automatique (Machine Learning - ML) et l'apprentissage profond (Deep Learning - DL).

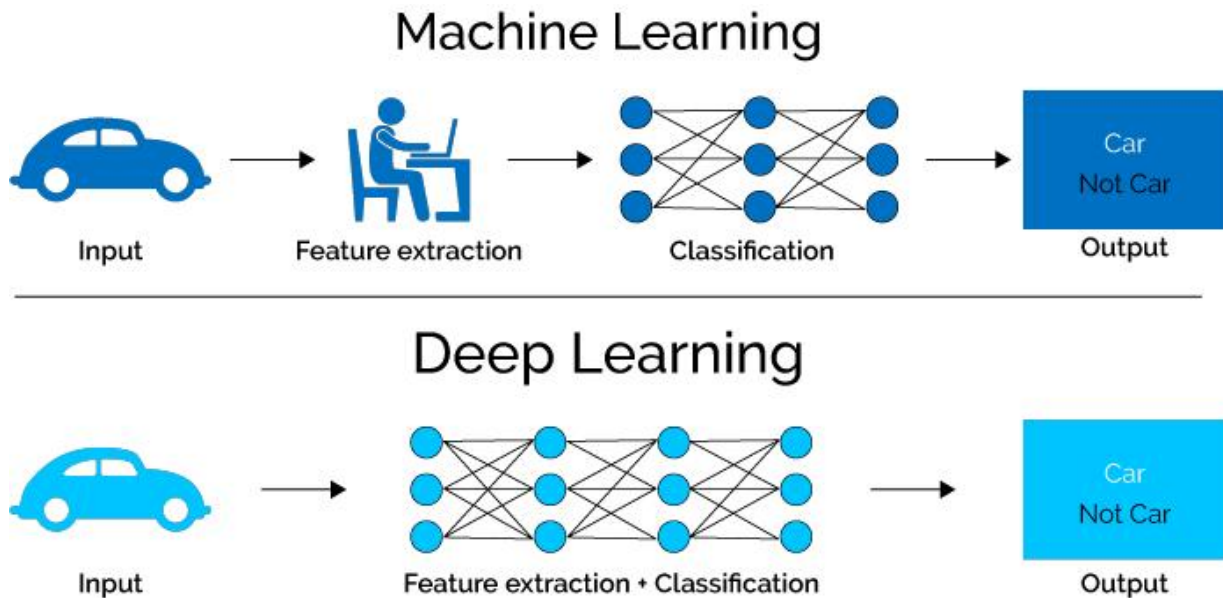


Figure 9:ML versus DL [20].

### 3.2. Les réseaux de neurones artificiels «RNA»

Les réseaux de neurones artificiels sont considérés comme une technologie innovante dans les domaines de l'informatique et de la neuro-informatique. Ils sont conçus pour apprendre automatiquement et permettre aux ordinateurs de résoudre des problèmes en utilisant des techniques d'intelligence artificielle[21].

Les réseaux de neurones artificiels (RNA) sont constitués d'un ensemble de processeurs élémentaires connectés les uns aux autres de manière dense, travaillant en simultané. Chaque processeur élémentaire calcule une sortie en fonction des informations qu'il reçoit. Tout réseau qui possède une structure hiérarchique est considéré comme un réseau de neurones[22].

### 3.2.1. Le neurone biologique

Le neurone est une cellule fondamentale du système nerveux qui se distingue par sa capacité à transmettre des informations. Bien qu'il puisse varier en termes de structure et de fonction d'une espèce à l'autre, il possède des caractéristiques communes qui le définissent en tant que cellule nerveuse. Le neurone est composé de trois parties principales : les dendrites, le noyau et l'axone[23]. La Figure 10 présente le neurone biologique . Cette illustration met en évidence la structure et les composants essentiels d'un neurone.

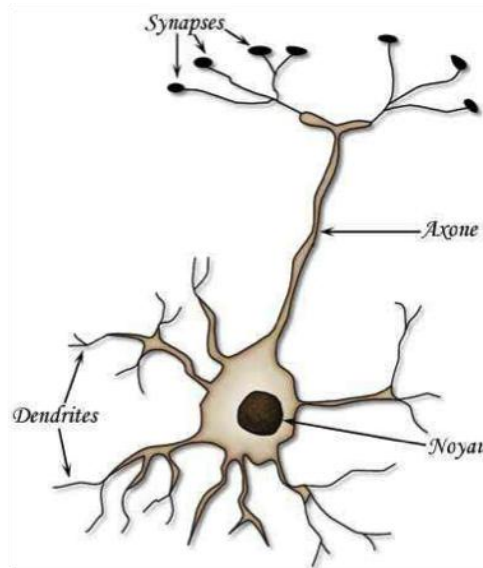


Figure 10 :Le neurone biologique[web 6].

### 3.2.2. Le neurone formel

Les réseaux neuronaux artificiels (RNA, en Anglais Artificial Neural Network) cherche à simuler ces réseaux et à faire en sorte que les ordinateurs agissent comme des cellules cérébrales interconnectées. Différentes parties du cerveau humain sont responsable du traitement de différentes informations, lorsque l'information entre dans le cerveau, chaque niveau de neurones traite l'information, et les transmet à la couche suivante, plus élevée. C'est cette approche en couches du traitement de l'information et de la prise de décision que les RNA tentent de simuler. Dans sa forme la plus simple, un RNA ne peut avoir que trois couches de neurones[24] :

- La couche d'entrée : ou les données entrent dans le système.
- La couche cachée : ou les informations sont traitées.
- La couche de sortie : ou le système décide de ce qu'il doit faire en fonction des donnée.

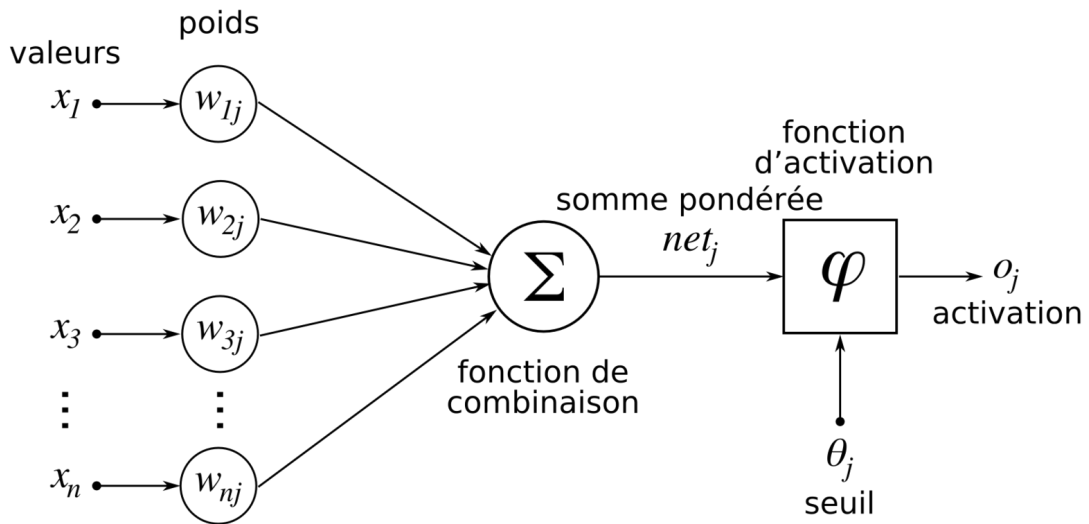


Figure 11 : Modèle du neurone formel[web 7].

### 3.3. Les différentes architectures d'apprentissage profond

#### 3.3.1. Réseau de neurones feed-forward

- **Le perceptron monocouche** : est une architecture de réseau neuronal très simple qui se compose d'une seule couche d'entrée et d'une seule couche de sortie. Il est capable de résoudre des problèmes linéairement séparables, tels que la fonction logique 'OU' ou 'ET'. Dans ce type de réseau, chaque valeur d'entrée est connectée à tous les neurones de la couche de sortie, comme illustré dans le schéma suivant[25] :

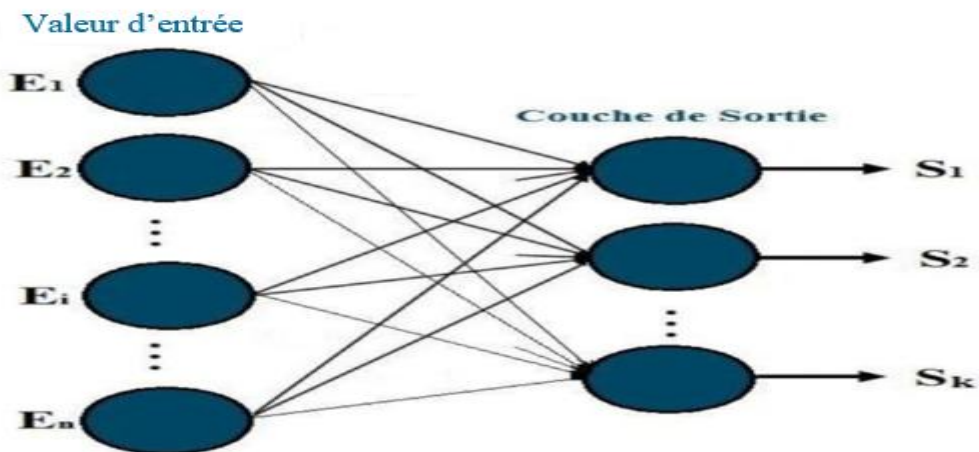


Figure 12:Le perceptron monocouche[25].

$E_1, \dots, E_i, \dots, E_n$  : Représentent les n valeurs d'entrée.

$S_1, S_k$ : Représentent les k valeurs de Sortie.

- **Le Perceptron MultiCouche ( PMC )**:contient des couches cachées entre la couche d'entrée et celle de sortie, ce réseau résout des problèmes nonlinéairement séparables ( ex : fonction logique 'XOR' ); et là aussi, chaque neurone des couches cachées est connecté à tout les neurones de la couche qui la précède et celle qui la suit. La figure ci-dessous représente un PMC)[25] :

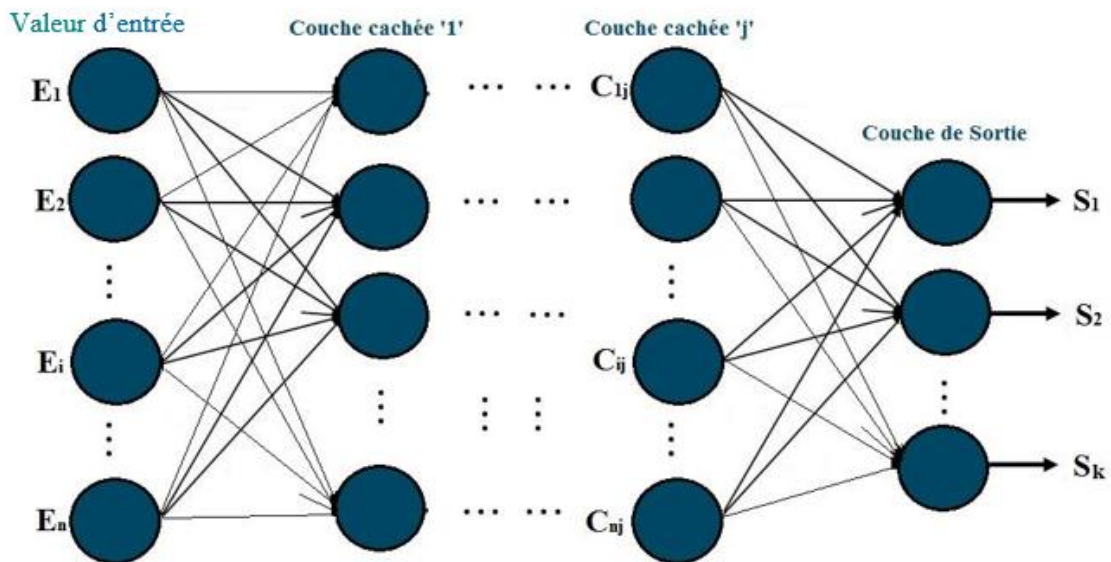


Figure 13:Le perceptron multicouche[25].

$E_1, \dots, E_i, \dots, E_n$  : Représentent les n valeurs d'entrée.

$C_{1j}, \dots, C_{mj}$ : Représentent les m valeurs de la couche cachée j.

$S_1, S_k$ : Représentent les k valeurs de Sortie.

### 3.3.2. Les réseaux de neurones profonds

Les réseaux de neurones profonds sont des réseaux de neurones artificiels qui ont plusieurs couches cachées, ce qui leur permet de capturer des relations complexes entre les entrées et les sorties. En utilisant des réseaux de neurones profonds, les modèles peuvent être entraînés pour effectuer des tâches telles que la classification d'images, la reconnaissance de la parole, la traduction automatique, etc [26].

- **Les réseaux de neurones récurrents (RNN)**

Le Réseau Neuronal Récurrent (RNN) est une classe d'apprentissage profond .Les RNN sont reconnus pour leur capacité à traiter et à obtenir des informations à partir de données

séquentielles. Par conséquent, l'analyse vidéo, le sous-titrage d'images, le traitement du langage naturel (NLP, en anglais pour Natural Language Processing) et l'analyse musicale dépendent tous des capacités des réseaux neuronaux récurrents[27].

Contrairement aux réseaux neuronaux standard qui supposent l'indépendance entre les points de données, les RNN capturent activement les dépendances séquentielles et temporelle entre les données. La Figure 14 présente un réseau neuronal récurrent (RNN). Elle illustre la structure et le fonctionnement du RNN.

### Recurrent Neural Networks

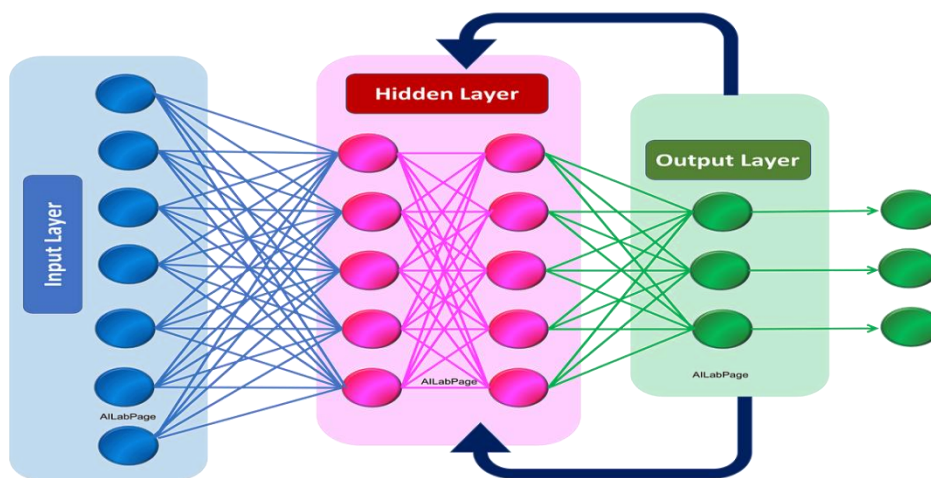


Figure 14:Réseau Neuronal Récurrent[web 7].

#### ➤ Les réseaux de neurones convolutifs (CNN ou ConvNet)

Les CNN (Convolutional Neural Networks) sont une famille de réseaux neuronaux multicouches spécialement conçus pour être utilisés sur des données bidimensionnelles, telles que des images et des vidéos. Les CNN sont la première approche d'apprentissage profond véritablement réussie dans laquelle de nombreuses couches d'une hiérarchie ont été entraînées de manière robuste. Un CNN est un choix d'architecture qui exploite les relations spatiales et temporelles pour réduire le nombre de paramètres à apprendre et améliorer ainsi l'entraînement général en propagation avant-arrière. Les CNN ont été proposés comme un cadre d'apprentissage profond motivé par des exigences minimales de prétraitement des données.

Dans les CNN, de petites parties de l'image sont traitées en tant qu'entrées de la couche la plus basse de la structure hiérarchique. L'information se propage généralement à travers les

différentes couches du réseau, où à chaque couche, un filtrage numérique est appliqué afin d'obtenir des caractéristiques saillantes des données observées. La méthode confère un certain niveau d'invariance aux décalages, aux échelles et aux rotations, car le champ réceptif local permet au neurone ou à l'unité de traitement d'accéder à des caractéristiques élémentaires telles que des contours orientés ou des coins[28].

### 1. Architecture globale

Une architecture typique de réseau de neurones convolutionnelle (CNN) est composée de couches de convolution et de pooling alternées, suivies de une ou plusieurs couches entièrement connectées à la fin. Le réseau CNN est divisé en deux parties importantes, chacune ayant un objectif différent. La première partie du réseau est chargée d'extraire les caractéristiques et est constituée de couches convolutionnelles et de pooling. La deuxième partie effectue la classification en utilisant les fonctionnalités extraites à l'aide de couches entièrement connectées. L'agencement des différentes couches dans une architecture CNN joue un rôle essentiel dans la conception de nouvelles architectures et dans l'amélioration des performances du modèle. Cette section offre une brève explication du rôle de chaque couche dans une architecture CNN [29]. La Figure 15 présente un réseau neuronal convolutif (CNN) elle met en évidence les différentes couches convolutives.

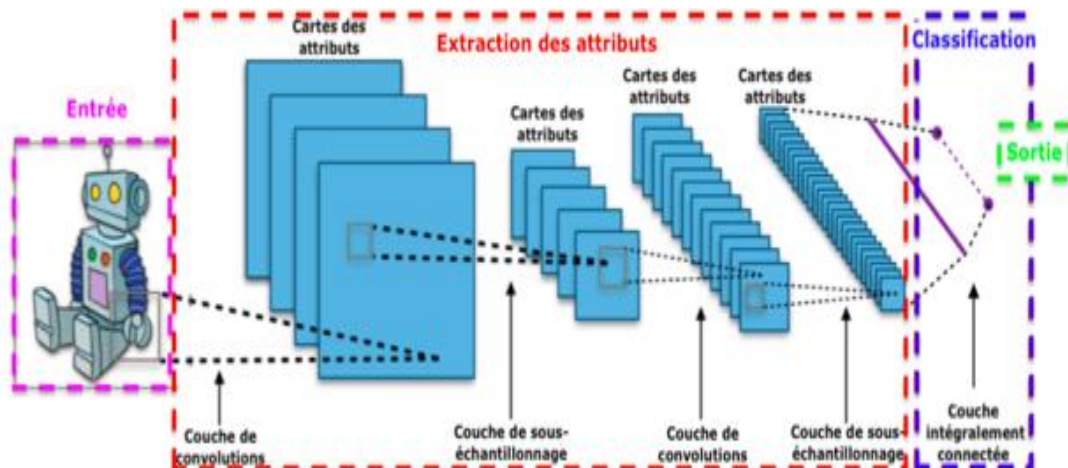


Figure 15:Réseau neuronal convolutif[web 8].

- **Couche de Convolution**

La couche convolutionnelle est l'élément central d'un CNN, où la plupart des calculs sont effectués. Elle est composée de plusieurs filtres optimisables qui transforment l'entrée  $X$  en une sortie  $y$ , également appelée carte de caractéristiques. Le nombre de

filtres détermine la profondeur de la couche convolutionnelle. L'opération de convolution est effectuée localement et de manière itérative, consistant essentiellement à faire glisser le filtre sur la couche et à calculer le produit scalaire entre le filtre et les valeurs de la couche. Le résultat est une nouvelle couche de produits scalaires pour chaque filtre, qui est également une carte de caractéristiques. Les premières cartes de caractéristiques d'un CNN peuvent inclure des motifs simples tels que des coins, des cercles ou des bords, tandis que dans les couches plus profondes, la convolution révèle généralement des motifs plus abstraits et des concepts de niveau supérieur. Les cartes de caractéristiques dérivées sont ensuite utilisées en tant qu'entrée pour la couche suivante[30].

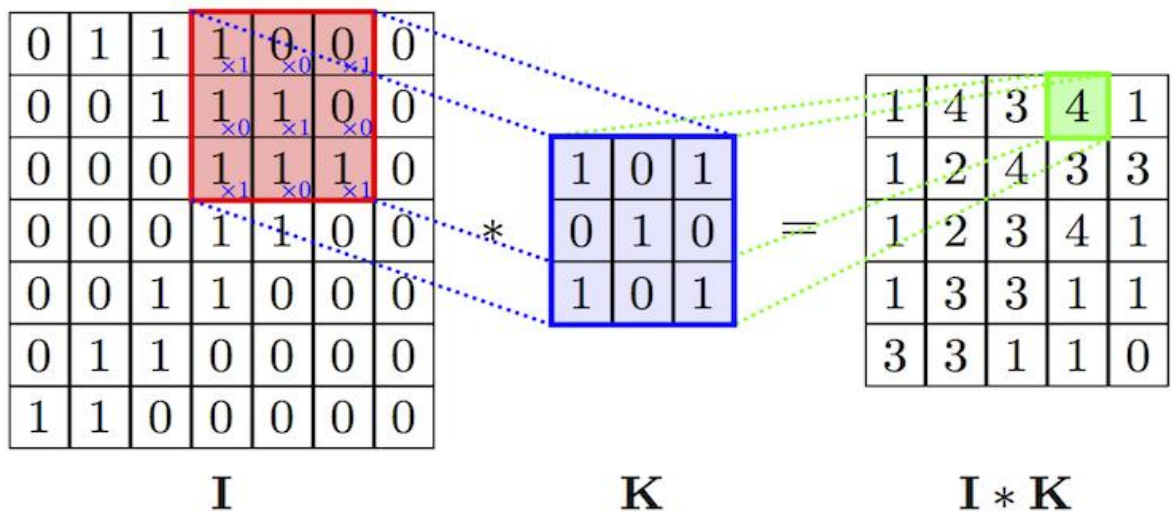


Figure 16:Un exemple de l'opération de convolution[31].

- **Couche de Pooling (POOL)**

Le pooling, ou sous-échantillonnage, consiste simplement à réduire la taille d'une image. Il prend une petite région de la sortie de la convolution en tant qu'entrée et la sous-échantillonne pour produire une seule sortie. Différentes techniques de pooling sont disponibles, telles que le pooling maximal (max pooling), le pooling moyen (mean pooling), le pooling moyen (average pooling), etc. Le max pooling prend la valeur maximale des pixels d'une région, comme illustré dans la figure 3. Le pooling réduit le nombre de paramètres à calculer mais rend le réseau invariant aux translations en termes de forme, de taille et d'échelle.

En résumé, le pooling est une opération qui permet de réduire la taille des caractéristiques en prenant une région de la sortie de la convolution et en en sélectionnant une seule valeur représentative. Cela réduit le nombre de paramètres à calculer tout en rendant le réseau invariant aux translations de forme, de taille et d'échelle[31].

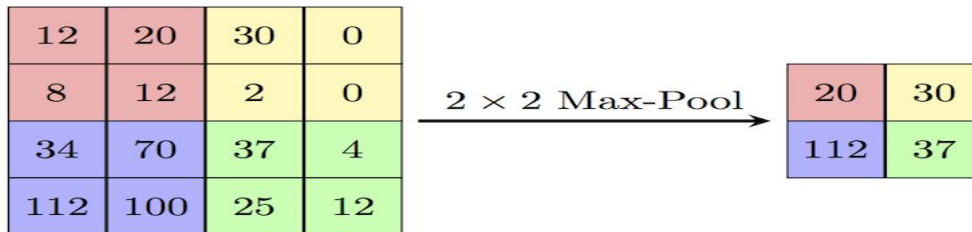


Figure 17:Un exemple de l'opération de Max pooling[31].

- **Couche Fully-Connected (FC)**

la couche fully-connected est une couche dans un réseau de neurones qui connecte chaque neurone de la couche précédente à tous les neurones de la couche actuelle. Elle est utilisée pour la classification ou la prédiction finale et permet d'apprendre des relations complexes entre les caractéristiques extraites et les étiquettes de sortie[31].

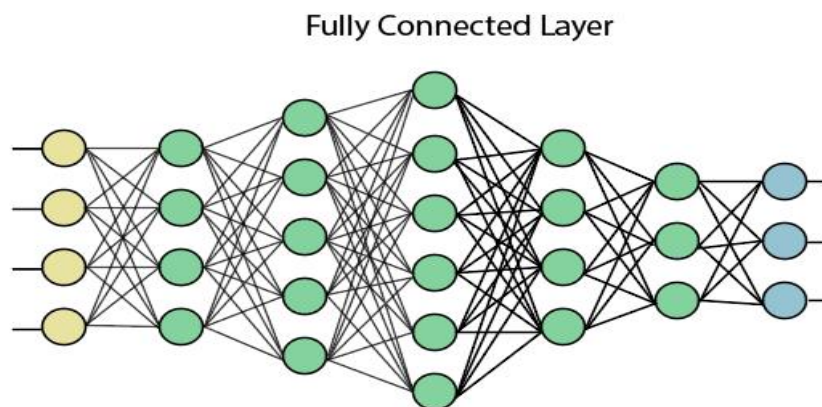


Figure 18: Couche Fully-Connected [Web 9].

## 4. Quelques systèmes existants de prediction des maladies cardiovasculaires

Dans cet article[32], les modèles utilisés sont un réseau de neurones à convolution (CNN) et un réseau de neurones convolutifs à mémoire à court et long terme (ConvLSTM). Les signaux ECG d'entrée sont transformés en images 2D (spectrogrammes) qui sont ensuite utilisées comme entrée pour les modèles CNN et ConvLSTM.

Les résultats finaux obtenus sont les suivants :

- Sur les spectrogrammes de l'ensemble de données MIT-BIH, une précision globale de 97 % a été atteinte.
- Sur les spectrogrammes de l'ensemble de données MIT-BIH compressé, une précision globale de 98 % a été obtenue.
- Sur l'ensemble de données PhysioNet 2016, une précision globale de 94 % a été réalisée.
- Sur l'ensemble de données PhysioNet 2018, une précision globale de 91 % a été obtenue.

Ces résultats démontrent que le cadre proposé, utilisant à la fois le CNN et le ConvLSTM, est robuste et efficace pour la détection automatique des arythmies, même dans le cas de données bruitées.

Dans cet article[33], les chercheurs ont développé un système de classification automatisée basé sur les réseaux de neurones artificiels (ANN) pour les arythmies cardiaques en utilisant des enregistrements ECG multi-canaux. Dans cette étude, l'objectif principal est de produire des résultats de classification d'arythmie à forte confiance pour les systèmes de soutien à la décision diagnostique. Un modèle de réseau neuronal avec l'algorithme de rétropropagation est utilisé pour classifier les cas d'arythmie en classes normales et anormales. Les modèles de réseaux sont entraînés et testés sur l'ensemble de données MIT-BIH arrhythmie. Les différentes structures d'ANN ont été entraînées avec un mélange de données arythmiques et non arythmiques de patients. Les performances de classification sont évaluées à l'aide des mesures suivantes : sensibilité, spécificité, précision de classification, erreur quadratique moyenne (MSE), courbes caractéristiques de fonctionnement du récepteur (ROC) et aire sous la courbe (AUC).

Les résultats expérimentaux montrent une précision de 96,77% sur la base de données MIT-BIH et de 96,21% sur une base de données préparée en incluant également la base de données

Dans cet article[34], une approche de classification des signaux d'électrocardiogramme (ECG) basée sur l'intelligence artificielle est développée en utilisant l'apprentissage profond (Deep Learning ou DL) à l'aide d'un réseau de neurones convolutifs (CNN).

Le CNN permet d'apprendre automatiquement les caractéristiques à partir des signaux ECG du domaine temporel, sans nécessiter d'extraction manuelle de ces caractéristiques. Le CNN est entraîné et testé sur un ensemble de données ECG provenant de la base de données MIT-BIH, qui est une référence dans le domaine. Les signaux ECG sont utilisés pour classer sept types d'arythmies, ce qui permet aux cardiologues d'identifier efficacement les patients atteints de maladies cardiaques. Le système proposé est comparé avec différentes fonctions d'activation en faisant varier le nombre d'époques. Les résultats obtenus montrent que la fonction d'activation ELU donne de meilleurs résultats avec une précision de 93,6 % et une perte de 0,2.

Les arythmies sont des modifications irrégulières du rythme cardiaque normal et leur identification manuelle efficace nécessite beaucoup de temps et dépend de l'expérience des cliniciens. [35] Cet article propose une nouvelle approche basée sur les réseaux neuronaux convolutifs (CNN) en 2D pour la classification précise de cinq types d'arythmie différents. Les performances de l'architecture proposée ont été testées sur des signaux d'électrocardiogramme (ECG) provenant de la base de données de référence des arythmies MIT-BIH. Les signaux ECG ont été segmentés en battements cardiaques et chacun des battements cardiaques a été converti en images en niveaux de gris en 2D, qui ont été utilisées comme données d'entrée pour la structure du CNN. La précision de l'architecture proposée a été de 97,42% sur les résultats d'entraînement, ce qui a révélé que l'architecture CNN en 2D proposée avec des images d'ECG transformées peut atteindre la plus grande précision sans prétraitement, extraction de caractéristiques et étapes de sélection de caractéristiques pour les signaux ECG.

## 5. Conclusion

Dans ce chapitre, nous avons d'abord défini l'apprentissage automatique, puis nous avons exploré les différents types d'apprentissage et leurs algorithmes, ainsi que leurs mécanismes.

Nous avons également abordé l'apprentissage profond. De plus, nous avons discuté des méthodes d'évaluation des performances de ces approches.

# Chapitre 3: Système réalisé

## **1. Introduction**

Les maladies cardiaques représentent l'une des principales causes de décès dans le monde entier, engendrant un fardeau considérable pour la santé publique. La prévention, la détection précoce et la prise en charge efficace de ces maladies sont d'une importance primordiale pour améliorer les résultats cliniques et réduire les taux de mortalité. Dans cet objectif, l'utilisation de systèmes intelligents de prédiction des maladies cardiaques se révèle être une approche prometteuse pour la prédiction précoce.

Ce chapitre est dédié à la conception d'un système intelligent de prédiction des maladies cardiaques, basé sur les avancées de l'intelligence artificielle. Nous proposons d'exploiter les puissantes capacités de l'apprentissage automatique et de l'apprentissage profond pour analyser les données médicales pertinentes et fournir des prédictions fiables concernant des maladies cardiaques chez les individus.

## **2. L'objectif et la motivation**

Notre objectif principal est de développer un système de prédiction précis des maladies cardiovasculaires. Pour cela, nous avons choisi une approche basée sur les techniques avancées d'apprentissage profond. Les maladies cardiovasculaires représentent un problème de santé majeur, et il est crucial de les détecter précocement et de mettre en place une prise en charge adaptée pour réduire les taux de mortalité. En utilisant des modèles basés sur l'apprentissage profond, notre objectif est d'analyser les données médicales pertinentes afin d'identifier les individus à haut risque de développer des maladies cardiovasculaires.

Nous réaliserons des expérimentations approfondies pour déterminer les meilleures configurations de modèles et les paramètres optimaux, afin d'obtenir des résultats de prédiction fiables. Nous évaluerons notre système en le comparant à d'autres approches existantes et en le confrontant aux travaux de recherche les plus récents. Nous cherchons ainsi à démontrer l'efficacité de notre approche basée sur l'apprentissage profond et sa capacité à améliorer la détection et la prévention des maladies cardiovasculaires.

## **3. Architecture générale**

Notre système est basé sur une architecture de réseau neuronal profond (dans notre cas un réseau convolutif uni-dimensionnel approprié au signal ECG). Il est constitué de plusieurs couches de neurones interconnectées qui permettent d'extraire des caractéristiques pertinentes

à partir des données d'entrée. Cette approche nous permet de modéliser de manière efficace les relations complexes entre les données d'entrée et les classes de sortie. La Figure 19 présente l'architecture globale de la classification des signaux ECG. Elle illustre les différentes étapes du processus.

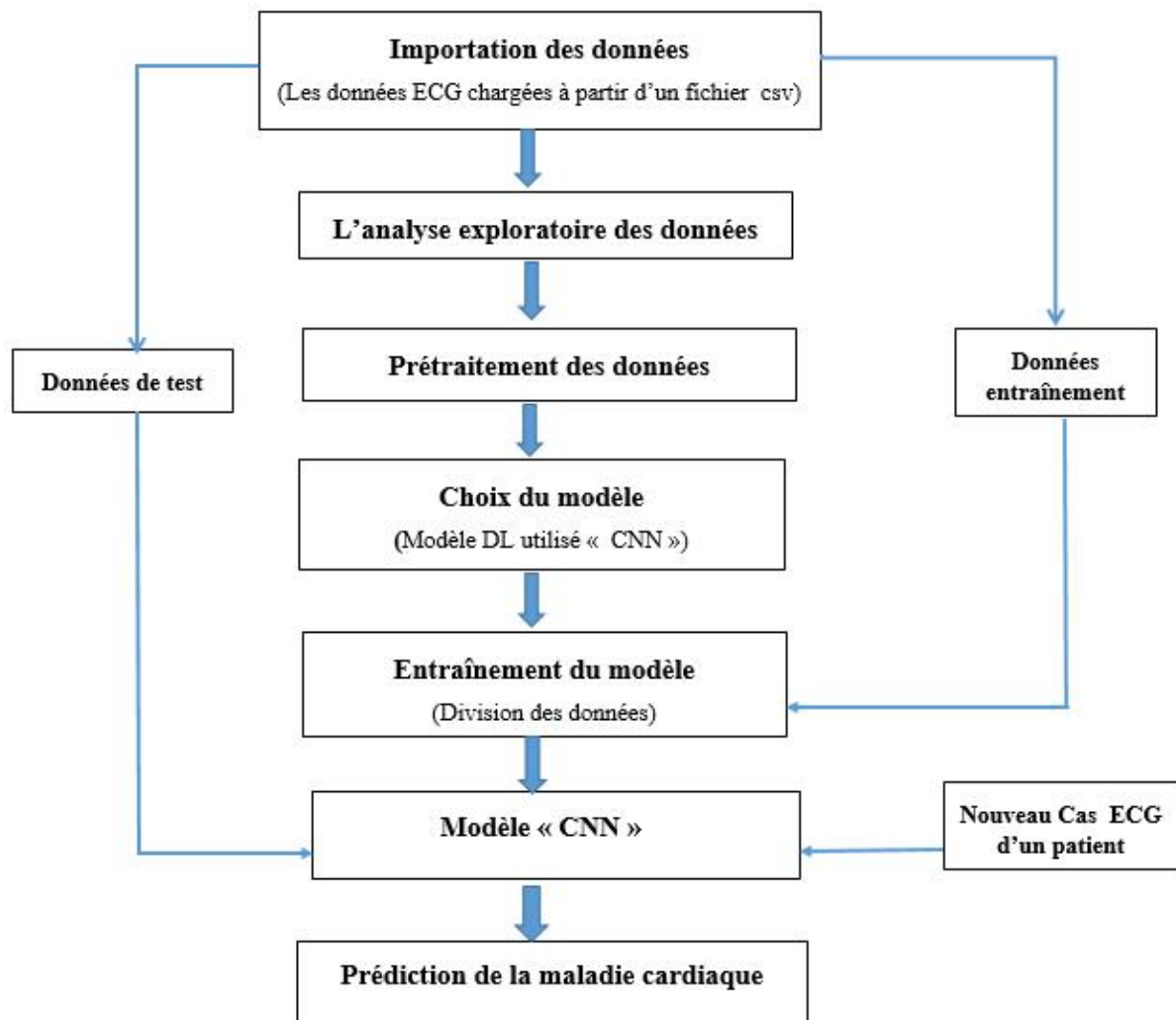


Figure 19: L'architecture globale de la classification des signaux ECG.

#### 4. Présentation détaillée du système

Dans cette section, nous allons fournir une description détaillée des différentes phases qui composent notre système.

### 4.1. Base de données utilisée

#### 4.1.1. présentation de la base

Nous avons utilisé une base de données d'arythmies MIT-BIH de Physionet [36]. Cette base de données est composée de signaux d'électrocardiogramme (ECG) représentant les battements de cœur. Elle contient à la fois des cas normaux et des cas affectés par différentes arythmies et infarctus du myocarde. Les signaux ont été prétraités et segmentés pour obtenir des segments correspondant à chaque battement de cœur.

La base de données est organisée sous forme de fichiers CSV, où chaque fichier contient une matrice de données. Chaque ligne de la matrice représente un exemple de la base de données, et le dernier élément de chaque ligne indique la classe à laquelle cet exemple appartient.

En utilisant cette base de données, nous avons réalisé des expériences pour étudier la classification des battements de cœur en utilisant des architectures de réseaux neuronaux profonds.

La base de données utilisée dans notre projet est caractérisée par les éléments suivants :

- Nombre d'échantillons : 109446
- Nombre de catégories : 5
- Fréquence d'échantillonnage : 125 Hz
- Classes : {0: 'N', 1: 'PVC', 2: 'APC', 3: 'RBB', 4: 'PAB'}

Où : 'N' représente l'état normale du coeur.

'PVC ,APC,RBB,PAB' représente les 4 types d'arythmie.

La base est divisée en deux parties: 80% de l'ensemble des données pour l'apprentissage (87554 échantillon) et 20% pour le test (21892 échantillon).

#### 4.1.2. L'analyse exploratoire des données

L'exploration des données joue un rôle essentiel dans la compréhension approfondie de la nature des données et de la structure de base. En analysant les ensembles de données de manière systématique, elle permet de découvrir des informations précieuses sur les types de données présents, tels que les données numériques, catégoriques, textuelles ou temporelles.

### ➤ Aperçu sur la base de données

La base de données se compose de 87554 exemples, chacun ayant 188 attributs (features). Les attributs vont de 0 à 186, et la colonne 187 représente la classe cible (label) de la base de données. Dans notre cas de classification multi-classe, les valeurs possibles de la classe cible sont 0, 1, 2, 3 et 4. La Figure 20 présente les cinq premières lignes ou enregistrements de la base de données MIT-BIH.

```
train_df.head()
```

	0	1	2	3	4	5	6	7	8	9	...	178	179	180	181	182	183	184	185	186	187
0	0.977941	0.926471	0.681373	0.245098	0.154412	0.191176	0.151961	0.085784	0.058824	0.049020	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
1	0.960114	0.863248	0.461538	0.196581	0.094017	0.125356	0.099715	0.088319	0.074074	0.082621	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2	1.000000	0.659459	0.186486	0.070270	0.070270	0.059459	0.056757	0.043243	0.054054	0.045946	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
3	0.925414	0.665746	0.541436	0.276243	0.196133	0.077348	0.071823	0.060773	0.066298	0.058011	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
4	0.967136	1.000000	0.830986	0.586854	0.356808	0.248826	0.145540	0.089202	0.117371	0.150235	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

5 rows x 188 columns

Figure 20 : Les 5 premières lignes de la base de données MIT-BIH.

### ➤ Visualisation des signaux

La visualisation des signaux dans le domaine temporel permet d'observer leur forme et leur comportement, révélant des tendances, des motifs périodiques ou des variations aléatoires. C'est une étape cruciale pour comprendre la nature des signaux.

#### 1.Exemple d'un ECG normal et anormal

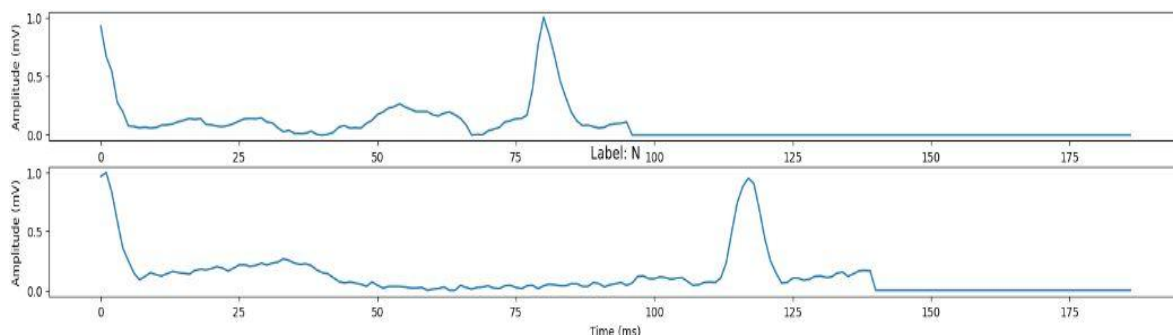


Figure 21: ECG normal.

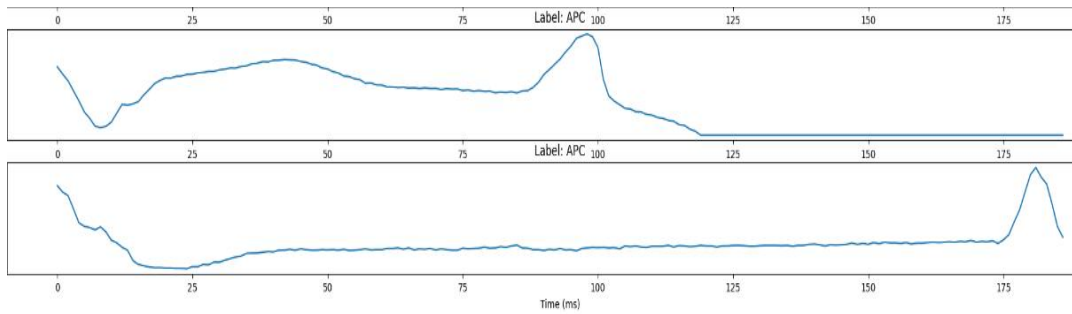


Figure 22: ECG anormal.

### 4.2. Prétraitement des données

Le prétraitement des données fait référence aux transformations et aux manipulations appliquées sur les données brutes afin de les rendre adaptées à l'entraînement du modèle.

Dans notre cas, le prétraitement des données comprend la normalisation, l'ajout d'une dimension supplémentaire et la conversion des données.

#### 4.2.1. La normalisation

Avant la normalisation, les données présentaient une distribution avec une moyenne non nulle et un écart-type significatif. Après la normalisation, les données ont été transformées pour avoir une moyenne très proche de zéro et un écart-type proche de un. Cette normalisation a permis d'éliminer les écarts d'échelle entre les caractéristiques, facilitant ainsi la comparaison et l'apprentissage du modèle de manière plus équilibrée.

#### 4.2.2. L'ajout d'une dimension

Après avoir appliqué la normalisation aux données, nous avons réorganisé la représentation des données pour inclure une dimension supplémentaire. La nouvelle forme des données est  $(87554, 1, 187)$ , où 87554 représente le nombre d'exemples, 1 représente la dimension ajoutée, et 187 correspond à la longueur de chaque séquence. Cette modification de forme permet d'exploiter davantage la structure et les relations entre les points de la séquence, contribuant ainsi à une meilleure modélisation et à des résultats plus précis.

#### 4.2.3. La conversion des données

Enfin, la conversion des données a également été effectuée dans le cadre du prétraitement. Cette étape vise à s'assurer que les données sont dans le format approprié pour l'apprentissage.

### 4.3. Choix du modèle

Le système conçu dans ce travail est basé sur une architecture de réseau neuronal profond ou plus précisément un réseau convolutif uni-dimensionnel. L'avantage de ce réseau est qu'il est adéquat au traitement des données séquentielles uni-dimensionnelles comme dans notre cas (signal ECG).

Ce type de réseau a une architecture flexible et ajustable en utilisant plusieurs hyperparamètres. Ces derniers jouent un rôle crucial dans la bonne conduite du processus d'apprentissage. Le choix judicieux de ces hyperparamètres est essentiel pour obtenir de bonnes performances du modèle.

Le premier hyperparamètre est le "Batch Size" (bs), qui spécifie le nombre d'exemples d'entraînement utilisés pour mettre à jour les poids du modèle à chaque itération. Dans notre cas, nous avons mis "Batch Size" à 1024 comme une valeur initiale.

Ensuite, nous avons le "Learning Rate" (LR) ou la vitesse d'apprentissage, qui détermine l'amplitude des ajustements des poids lors de l'optimisation.  $(LR) = 1e-3$ . Cela signifie que les poids du notre modèle seront ajustés de manière itérative avec une petite valeur fixée à 0.001.

Enfin, le nombre d'epochs (nombre d'itérations complètes sur l'ensemble des données) a été fixé à 50. Chaque epoch permet au modèle d'améliorer progressivement ses performances en ajustant ses poids sur l'ensemble des données d'entraînement.

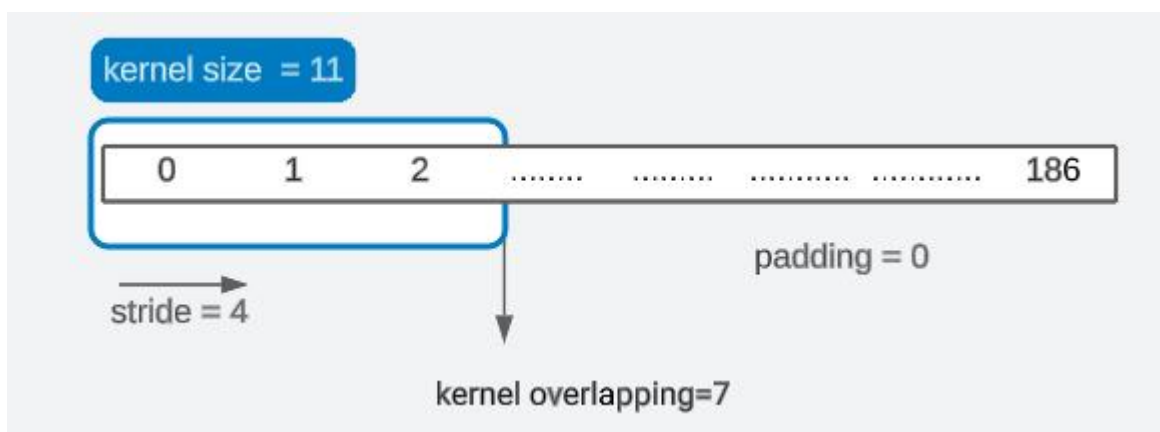


Figure 23: les paramètres du bloc de convolution.

La figure représente les paramètres clés du bloc de convolution dans le modèle. Ce bloc de convolution est utilisé pour extraire des caractéristiques à partir des données d'entrée. Voici une explication des paramètres utilisés

- **Kernel Size (taille du noyau)** Le paramètre kernel size représente la taille du noyau de convolution utilisé dans l'opération.
- **Stride (pas)** Le paramètre strides représente le pas de déplacement du noyau lors de la convolution.
- **le paramètre "padding"** représente le remplissage appliqué aux bords de l'entrée avant d'appliquer la convolution. Plus précisément, il indique le nombre de zéros ajoutés autour des bords.

Lorsque "padding = 0", cela signifie qu'aucun zéro n'est ajouté et il n'y a donc pas de remplissage.

En revanche, "padding = 1", cela signifie qu'un seul zéro est ajouté de chaque côté de l'entrée. Cela permet de conserver la taille spatiale de la sortie par rapport à l'entrée d'origine.

- **Le "kernel overlapping"** (chevauchement du noyau) se réfère à la façon dont le noyau de convolution se déplace lors de l'opération de convolution. Normalement, lorsqu'un noyau de convolution se déplace sur l'entrée, il se déplace sans chevauchement, c'est-à-dire qu'il se déplace de la taille du stride à chaque étape. Cependant, avec le chevauchement du noyau, le noyau peut se déplacer avec un certain chevauchement entre deux pas de stride.

### 4.4. Etape de classification

#### 4.4.1. Phase d'apprentissage

Dans la phase d'apprentissage, nous utilisons les 87 554 échantillons de données pour générer un modèle CNN. Ce modèle sera ensuite utilisé dans la phase suivante. Cette phase d'apprentissage représente 80% de l'ensemble des données.

#### ➤ Génération du modèle

La génération du modèle CNN (Convolutional Neural Network) constitue une étape clé de notre système de classification des maladies cardiovasculaires basé sur les signaux ECG. Le modèle CNN est une architecture de réseau neuronal profond spécialement conçue pour extraire des caractéristiques pertinentes à partir des données d'entrée et réaliser des tâches de classification.

Comme mentionné précédemment dans le chapitre 2, les architectures CNN suivent généralement des principes de conception similaires. Elles appliquent une couche de

convolution aux données d'entrée, effectuent du sous-échantillonnage (Max Pooling) sur les dimensions spatiales et augmentent progressivement le nombre de cartes de caractéristiques. On retrouve également des couches entièrement connectées (Fully Connected), des fonctions d'activation et des fonctions de Dropout.

Les couches de convolution, de pooling et entièrement connectées sont considérées comme le cœur d'un CNN, car elles jouent un rôle clé dans l'extraction de caractéristiques et la classification des données. Cependant, des couches supplémentaires peuvent être introduites pour améliorer les performances, comme la couche Softmax qui est souvent utilisée pour améliorer la précision de la détection d'images[37]. Il convient de noter que le nombre de couches dans un CNN peut varier en fonction de l'application, de ses besoins spécifiques et même de la langue de programmation utilisée. Il n'y a pas de nombre prédéfini de couches pour un CNN, car cela dépendra de la complexité des données, de la taille de l'ensemble de données et des objectifs de classification. La Figure 24 montre l'ordre des couches convolutives, de max pooling adaptatif et flatten dans l'architecture globale du modèle.

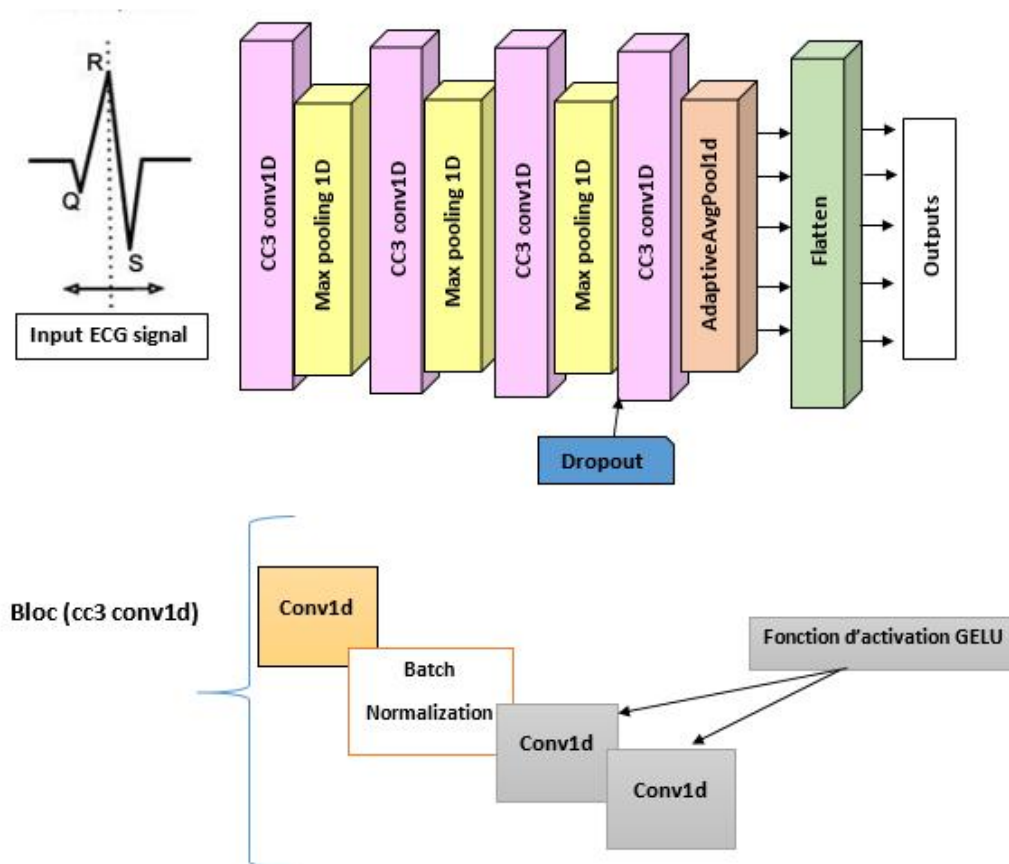


Figure 24:: L'architecture globale du modèle.

Voici l'architecture générale de notre modèle utilisant l'architecture "Network-in-Network" pour traiter les signaux en utilisant des couches de convolution 1D de taille 1x1 et une opération de pooling global average à la fin :

**Entrée du signal** La première couche représente l'entrée du signal brut.

**Couche de convolution** : cette couche permet d'extraire des caractéristiques discriminantes des signaux en appliquant des filtres de convolution de taille 1x1, de réduire la dimensionnalité et d'introduire de la non-linéarité dans le modèle.

Dans les couches de convolution, une couche de normalisation par lots est créée pour les activations en une dimension (1D). Cette couche est appliquée après la première couche de convolution dans chaque bloc.

**La fonction GELU** (Gaussian Error Linear Unit) est une fonction d'activation utilisée dans les blocs de convolution est appliquée après la normalisation par lots (Batch Normalization) dans chaque bloc de convolution. Cela permet d'introduire de la non-linéarité dans les activations intermédiaires du modèle, ce qui aide à apprendre des caractéristiques plus complexes et à améliorer la représentation des données.

**Couche de Pooling** : La couche de pooling est utilisée pour réduire la dimension spatiale des cartes de caractéristiques, préserver les informations discriminantes et réduire la complexité du modèle. Cela est réalisé en appliquant une opération de pooling, telle que le max pooling ou average pooling, sur des régions du signal.

**Pooling global average** : Une fois que les couches de convolution 1D ont extrait les caractéristiques pertinentes, une opération de pooling global average est appliquée.

Le pooling global average réduit les dimensions spatiales des caractéristiques en calculant la moyenne de chaque canal.

Cette opération agrège les informations spatiales en une seule valeur pour chaque canal, ce qui permet de réduire la quantité de données à traiter.

**Couche de Flatten**: la couche "Flatten" est la dernière couche du réseau. Cette couche est permet de traiter les signaux avec un réseau de convolution 1D sans avoir besoin de couches entièrement connectées supplémentaires. Cette couche est utilisée dans notre modèle pour transformer les sorties de la couche précédente en une forme

adaptée pour la classification des 5 classes. Elle permet d'obtenir les résultats finaux du modèle, où chaque élément du vecteur représente la probabilité ou la sortie associée à une classe spécifique.

### ➤ **Entraînement du modèle**

Pendant l'entraînement, le modèle itère sur les données d'entraînement pour un certain nombre d'époques prédéfini. À chaque itération, les données d'entraînement sont passées au modèle, qui génère des prédictions. En utilisant une fonction de perte appropriée, telle que la perte d'entropie croisée, la perte est calculée en comparant les prédictions avec les étiquettes réelles. Ensuite, la rétropropagation du gradient est effectuée pour mettre à jour les poids du modèle en utilisant l'optimiseur avec le pas d'apprentissage spécifié. Pendant l'entraînement, les métriques d'évaluation, telles que la précision, sont suivies pour évaluer les performances du modèle. Ce processus d'itération, de calcul de la perte, de rétropropagation et de mise à jour des poids se répète jusqu'à ce que toutes les époques soient terminées, permettant ainsi au modèle d'apprendre à associer les caractéristiques extraites aux classes correspondantes des maladies cardiovasculaires.

### ➤ **Optimisation du modèle**

L'optimisation d'un modèle CNN (Convolutional Neural Network) implique plusieurs étapes clés pour améliorer ses performances. Voici les principales techniques d'optimisation utilisées pour notre modèle :

- **Initialisation des poids** : Pour notre modèle, nous avons opté pour une initialisation aléatoire des poids. Cela signifie que les poids sont attribués à des valeurs aléatoires lors de la création du modèle. Cette approche permet d'introduire de la diversité dans les poids initiaux, favorisant ainsi l'apprentissage et l'adaptation du modèle aux données d'entraînement.
- **Fonction de perte (Loss function)** : La fonction de perte utilisée est la `CrossEntropyLoss` (perte de l'entropie croisée). Cette fonction de perte est spécialement conçue pour les tâches de classification où il y a plusieurs classes à prédire. Elle permet de mesurer l'écart entre la distribution de probabilités prédite par le modèle et les étiquettes réelles des classes. L'utilisation de la `CrossEntropyLoss` est courante car elle permet d'optimiser efficacement les modèles de classification et d'encourager des prédictions précises sur toutes les classes disponibles.

- **L'algorithme d'optimisation** : L'algorithme d'optimisation utilisé pour ajuster les poids de notre modèle pendant l'entraînement afin de minimiser la fonction de perte est Adam. Adam est un algorithme populaire qui combine les avantages de la descente de gradient stochastique (SGD) et de l'algorithme RMSprop. Il utilise des estimations adaptatives du premier moment (moyenne des gradients) et du second moment (moyenne mobile des carrés des gradients) pour mettre à jour les poids de manière efficace
- **Le pas d'apprentissage (Learning rate)** : Le pas d'apprentissage contrôle l'ajustement des poids lors de l'optimisation. Trop élevé, il peut provoquer une instabilité, trop faible, il ralentit l'apprentissage.
- **Régularisation**: la régularisation est une technique utilisée pour prévenir le surapprentissage et améliorer la capacité de généralisation du modèle. Dans notre modèle, nous avons appliqué la normalisation comme technique de régularisation.

➤ **Normalisation**

La normalisation, telle que la normalisation de lot (Batch normalization), permet de normaliser les activations des couches intermédiaires du modèle, ce qui stabilise et régularise l'apprentissage. Cela aide à réduire les effets nuisibles du surapprentissage en favorisant une meilleure généralisation du modèle aux nouvelles données. La normalisation contribue ainsi à améliorer les performances et la stabilité de notre modèle CNN.

➤ **Dropout**

Le dropout est une technique couramment utilisée dans l'optimisation des modèles de réseaux neuronaux, tels que les réseaux de neurones convolutifs (CNN) ou les réseaux de neurones profonds (DNN). Son objectif principal est de régulariser le modèle en réduisant le surajustement (overfitting). Lors de l'entraînement de notre modèle, le dropout est utilisé pour désactiver aléatoirement 10% de neurones à chaque étape d'apprentissage.

### 4.4.2. Phase de test

Dans la phase de test, notre modèle est utilisé pour effectuer des prédictions sur les 21892 échantillons de données, ce qui représente 20% de l'ensemble des données disponibles. Ces échantillons de données de test sont réservés exclusivement à l'évaluation des performances et de la capacité de généralisation de notre modèle.

Notre modèle est soumis aux échantillons de données de test, où il effectue des prédictions en se basant sur les connaissances acquises lors de la phase d'apprentissage. Les prédictions générées par notre modèle sont ensuite comparées aux étiquettes ou aux vérités terrain correspondantes pour évaluer sa précision et ses performances.

Dans cette phase, un cas d'électrocardiogramme (ECG) d'un patient est utilisé pour évaluer les performances du modèle de prédiction. Le cas concerne un patient présentant des symptômes d'arythmie cardiaque. Le signal ECG du patient est prétraité et passé à travers le modèle de prédiction entraîné précédemment.

## 5. Résultat et discussion

Afin d'optimiser les performances de notre système, nous avons procédé à une série d'expérimentations en ajustant les paramètres du modèle, notamment le nombre de couches convolutives, les fonctions d'activation, les taux de Dropout et avec des données normalisées et non normalisées.

- Lors de l'expérimentation, nous avons étudié l'impact du nombre de couches sur les performances du modèle. Nous avons observé que lorsque nous avons augmenté le nombre de couches de 3 à 5, la précision du modèle a légèrement diminué, passant de 98.6% à 98.5%. De même, lorsque nous avons réduit le nombre de couches de 3 à 2, la précision est restée à 98.5%.

Table 1 :ajustement du nombre de couches dans un modèle.

<b>Nombre de couches</b>	<b>précision globale</b>
3	98.6%
2	98.5%
5	98.5%

Ces résultats suggèrent qu'il existe un point d'équilibre dans la complexité du modèle en termes de nombre de couches. Une augmentation excessive du nombre de couches peut entraîner une augmentation de la complexité et de la capacité de représentation du modèle, ce qui peut potentiellement conduire à une légère baisse de la précision. De même, une

diminution du nombre de couches peut réduire la capacité du modèle à extraire des caractéristiques pertinentes, ce qui peut également affecter légèrement les performances.

- Les résultats indiquent clairement que l'utilisation de la fonction d'activation GELU a conduit à une meilleure précision par rapport à la fonction d'activation ReLU. La précision globale du modèle avec la fonction GELU a été de 98.6%, tandis qu'avec la fonction ReLU, elle a été de 85.2%.

Table 2 : Comparaison des performances en utilisant les fonctions d'activation GELU et ReLU

<b>Fonction d'activation</b>	<b>précision globale</b>
GELU	98.6%
ReLU	85.2%

La fonction d'activation joue un rôle crucial dans la capacité du modèle à apprendre des représentations complexes et à capturer les non-linéarités des données. Dans ce cas, la fonction GELU semble mieux convenir à la tâche de classification des maladies cardiovasculaires, car elle a permis d'obtenir une meilleure précision.

- Lors de notre étude, nous avons évalué l'impact de la normalisation des données sur les performances du modèle de prédiction des maladies cardiovasculaires. Nous avons comparé les résultats obtenus en utilisant la normalisation des données à ceux obtenus sans cette étape de prétraitement.

Table 3 : l'évaluation de l'impact de la normalisation des données.

<b>Normalisation des données</b>	<b>précision globale</b>
<b>Sans Normalizatio</b>	88.5%
<b>Avec Normalization</b>	98.6%

Sans la normalisation, la précision globale du modèle a été de 88.5%. Cependant, en incorporant la normalisation des données, nous avons observé une augmentation significative de la précision globale, atteignant 98.6%.

Ces résultats confirment l'importance de considérer la normalisation des données lors de la construction du modèles. La normalisation permet d'améliorer significativement les performances du modèle, en assurant une meilleure adéquation aux données et une meilleure capacité de généralisation.

- Après avoir mené différentes expérimentations, nous avons remplacer la couche `nn.AdaptiveAvgPool1d`( Adaptive Average Pooling ) par `nn.MaxPool1d` dans l'architecture de notre modèle. Cette modification a entraîné une légère diminution de la précision globale, passant de 98.6% à 97.6%.

Table 4: les résultats obtenus en utilisant deux types de couches de pooling.

<b>Couche de polling</b>	<b>précision globale</b>
<b>AdaptiveAvgPool1d</b>	98.6%
<b>MaxPool1d</b>	97.6%

La couche `nn.AdaptiveAvgPool1d(1)` effectue une opération de mise à l'échelle adaptative en calculant la moyenne adaptative des caractéristiques spatiales. Elle a été utilisée précédemment dans notre modèle et a permis d'obtenir de bons résultats en termes de précision.

- Pour régulariser le modèle, nous avons utilisé des taux de Dropout. En testant différents taux de Dropout, nous avons évalué leur effet sur la capacité du modèle à généraliser et à réduire le surapprentissage en désactivant aléatoirement certaines activations pendant l'entraînement.

Table 5: Résultats expérimentaux pour déterminer le taux de Dropout.

<b>Taux de Dropout</b>	<b>précision globale(%)</b>
0.1	98.7
0.15	98.6
0.25	98.6
0.3	98.5
0.4	92.2

Les résultats expérimentaux présentés dans le tableau confirment clairement que le taux de Dropout de 0.10 a donné les meilleurs résultats. Avec ce taux, qui signifie que seulement 10 % des neurones sont désactivés à chaque époque d'entraînement. Les taux de 0.15 et 0.25 ont également donné des précisions élevées de 98.6%. Cependant, à partir d'un taux de 0.30, la précision diminue légèrement à 98.5%, et 0.40 conduit à une précision plus basse de 92.2%. Il est important de choisir le taux de Dropout approprié pour éviter le surapprentissage.

## **5.1. Évolution des performances**

### **5.1.1. Les courbes de perte**

Les courbes de perte d'entraînement (train) et de validation (val) avec des lignes lissées sont tracées dans le graphique. La courbe "train" représente l'évolution de la perte d'entraînement au fil des époques, tandis que la courbe "val" représente l'évolution de la perte de validation. Les axes du graphique sont étiquetés "Epochs" pour l'axe horizontal et "Loss" pour l'axe vertical, indiquant le nombre d'époques et la valeur de la perte, respectivement.

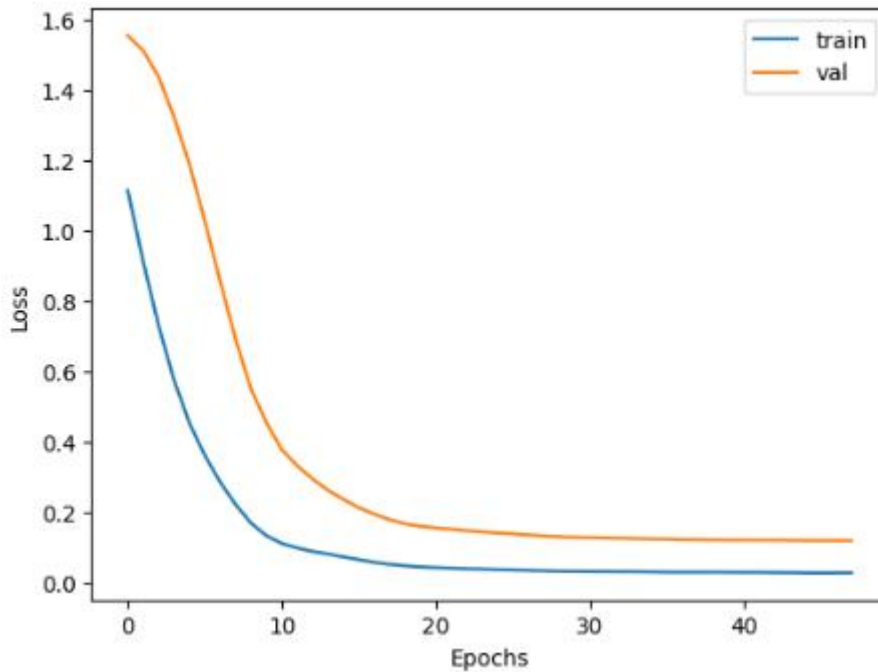


Figure 25: Analyse des performances du modèle avec lissage des pertes

la diminution de la perte pour l'entraînement et la validation est un indicateur positif de l'amélioration des performances du modèle. Cela suggère que notre modèle apprend et s'ajuste mieux aux données au fil du temps.

En plus de surveiller la diminution de la perte d'entraînement et de validation, il est essentiel d'examiner d'autres métriques de performance pour obtenir une évaluation complète des performances du modèle, notamment la matrice de confusion

### 5.1.2. La matrice de confusion

La matrice de confusion est une représentation tabulaire des performances d'un modèle de classification. Elle compare les prédictions du modèle avec les valeurs réelles des données, permettant ainsi d'évaluer la précision de la classification. Elle est utile pour calculer des mesures telles que la précision, le rappel, le score F1 et l'exactitude, et pour visualiser les erreurs de classification spécifiques.

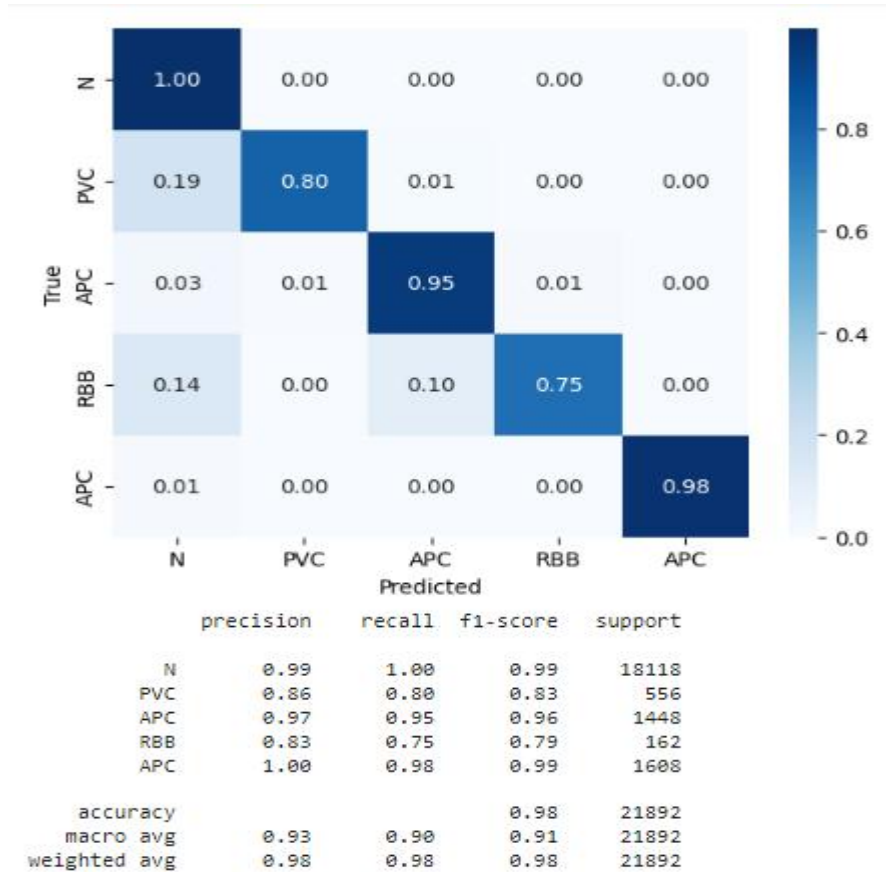


Figure 26: Matrice de confusion

Les résultats de l'évaluation de notre modèle de classification d'ECG sont présentés ci-dessus,

Voici ce que chaque mesure signifie :

- **Précision (precision)** : Il s'agit de la proportion des prédictions positives correctes parmi toutes les prédictions positives faites par le modèle. Pour la classe "N", la précision est de 0,99, ce qui signifie que 99 % des prédictions positives pour cette classe étaient correctes. Pour les autres classes, les précisions sont également élevées, mais légèrement plus basses.
- **Rappel (recall)** : Il s'agit de la proportion d'exemples positifs correctement identifiés parmi tous les exemples positifs réels. Pour la classe "N", le rappel est de 1,00, ce qui signifie que le modèle a correctement identifié tous les exemples positifs de cette classe. Pour les autres classes, les rappels sont légèrement inférieurs, mais toujours élevés.
- **Score F1 (F1-score)** : Il s'agit d'une mesure combinée de la précision et du rappel, calculée comme une moyenne pondérée harmonique entre les deux. Le score F1 est une mesure globale de la performance du modèle qui prend en compte à la fois la précision et

le rappel. Les valeurs de score F1 pour toutes les classes sont élevées, indiquant une bonne capacité du modèle à classer les différentes classes d'ECG.

- **Support** : Il s'agit du nombre d'exemples réels dans chaque classe. Par exemple, la classe "N" a un support de 18 118, ce qui signifie qu'il y a 18 118 exemples réels de cette classe dans l'ensemble de données.
- **Accuracy (exactitude)** : Il s'agit de la proportion d'exemples correctement classés parmi tous les exemples. L'exactitude globale de notre modèle est de 0,98, ce qui indique que 98 % des exemples ont été classés correctement.

## 5.2. L'analyse comparative des systèmes de prédiction des maladies cardiaques

Voici un tableau comparatif des différents modèles d'apprentissage profond utilisés dans diverses études pour prédire les maladies cardiaques

Table 6: Tableau récapitulatif des précisions globales.

<b>Etude</b>	<b>Modèle utilisés</b>	<b>BDD utilisée</b>	<b>Précision globale</b>
Article [32]	un réseau de neurones à convolution (CNN) et un réseau de neurones convolutifs à mémoire à court et long terme (ConvLSTM)	MIT-BIH	97%
		MIT-BIH compressé	98%
		Ensemble de données Physionet 2016.	94%
		Ensemble de données Physionet 2018	91%
Article [33]	Réseau de Neurones Artificiel (ANN)	MIT-BIH.	96,77%
		Base de données préparée incluant MIT-BIH	96,21%
Article [34]	un réseau de neurones à convolution (CNN)	MIT-BIH.	93.6%
Article [35]	Réseau de Neurones Convolutifs 2D (CNN2D)	MIT-BIH.	97.42%

Le tableau récapitulatif des performances des différents modèles dans les articles cités permet une comparaison directe avec notre système. Les résultats varient en fonction des modèles et des bases de données utilisées, avec des précisions globales allant de 91% à 98%. Notre système se situe dans cette fourchette, démontrant ainsi des performances comparables aux travaux précédents dans la classification des signaux cardiaques.

Lors de nos expérimentations, nous avons mené différentes modifications sur notre modèle de classification afin d'améliorer le taux de reconnaissance. Parmi toutes ces expérimentations, un cas a retenu notre attention particulière, celui où la précision globale atteint 98.7%. Ce cas spécifique correspond à l'utilisation de trois couches convolutives, des données normalisées, une fonction d'activation GELU et un taux de Dropout de 0.1. De plus, une couche d'AdaptiveAvgPool1d a été ajoutée à la fin du modèle pour une meilleure agrégation des informations. Ces ajustements combinés ont permis d'atteindre une précision remarquable de 98.7%, démontrant l'efficacité de notre approche pour la classification des signaux. Il convient de noter que lors de notre analyse comparative, le système de référence avait un taux de Dropout de 0.2. Nous avons ensuite testé plusieurs valeurs de taux de Dropout, et après évaluation, nous avons choisi un taux de 0.1 comme le plus optimal pour améliorer les performances du modèle.

## 6. Conclusion

Dans ce chapitre, nous avons présenté comment mettre en œuvre les différentes étapes pour développer un modèle de prédiction des maladies cardiovasculaires, où nous avons précédemment expliqué en détail la conception de ces étapes. Nous avons également effectué une simple comparaison avec des travaux récents. De plus, nous avons présenté nos expérimentations visant à améliorer les performances du modèle.

# Conclusion générale

## 1. Bilan

L'objectif de ce mémoire était d'explorer l'application des outils d'intelligence artificielle pour la classification des battements cardiaques, en particulier en utilisant une approche neuronale. Dans cette étude, nous nous sommes concentrés sur l'utilisation des réseaux de neurones artificiels, qui ont démontré leur efficacité dans la classification des arythmies cardiaques.

Nous avons utilisé un réseau de neurones convolutionnel à une dimension (CNN 1D) pour la classification des signaux. Notre modèle a été spécifiquement conçu pour un apprentissage supervisé. Nous avons utilisé les classes suivantes : "N" pour les battements cardiaques normaux, "pvc", "APC", "RBB", et "PAB" pour les autres types d'arythmies.

Lors du processus d'apprentissage, le modèle a été exposé à des exemples de battements cardiaques préalablement étiquetés avec leurs classes correspondantes. Le modèle a été entraîné à reconnaître les motifs caractéristiques dans les signaux temporels de l'ECG associés à chaque classe.

En nous basant sur les résultats obtenus, notre modèle a prouvé sa capacité à classer le signal ECG comme normal ou inhabituel avec une précision élevée. Nous avons obtenu une précision de 98,7% lors de la classification, ce qui démontre que le CNN est une technique efficace et précise pour cette tâche.

Dans un avenir proche, cette recherche sera améliorée pour fournir des solutions plus efficaces, opérationnelles et abordables permettant d'identifier les patients à risque de mort subite d'origine rythmique. Cela permettra de prendre des mesures préventives adaptées à chaque situation, ce qui pourrait potentiellement éviter des millions de décès.

## 2. Perspectives

Comme perspectives nous pouvons citer :

- Créer une API et intégrer le modèle prédictif à l'intérieur.
- Étendre la classification à d'autres types d'arythmies cardiaques pour une détection plus complète et précise des anomalies.

- Explorer d'autres techniques de prétraitement et d'extraction de caractéristiques pour améliorer encore les performances du modèle.

Tester notre modèle sur d'autres ensembles de données pour évaluer leur généralisation et leur performance dans différents contextes.

# Bibliographie

- [1] Halimi, R., & Hammouya, Y. Classification d'un signal ECG par RNA (RBF).
- [2] Karoui, A. (2021). Méthodes numériques pour la résolution de problèmes inverses en électrocardiographie (Doctoral dissertation, Bordeaux).
- [3] Laleg, T. M., Crépeau, E., & Sorine, M. (2005). Identification d'un compartiment vasculaire et des caractéristiques de la pression et du débit aortiques
- [4] Iaizzo, P. A. (Ed.). (2010). Handbook of cardiac anatomy, physiology, and devices. Springer Science & Business Media.
- [5] Khaddoumi, B. (2005). Analyse et modélisation de l'activité électrique du coeur dans le cas de pathologies ventriculaires (Doctoral dissertation, Université Nice Sophia Antipolis).
- [6] Goldberger, A. L. (2012). Clinical Electrocardiography E-Book: A Simplified Approach. Elsevier Health Sciences.
- [7] TEMIMI, Lahouari, et al. "Conception et Réalisation D'un électrocardiographe «ECG» numérique." (2014).
- [8] MEZIANE, H. (2003). Acquisition de signaux Electrocardiogrammes (ECG) à l'aide de la carte DSPACE. mémoire de n d'études, Université Abou Bekr Belkaid–Tlemcen, le, 2.

- [9] Dekkiche, L. (2020). Classification des arythmies ECG avec des méthodes de Machine Learning et de Deep Learning (Doctoral dissertation, Université Mouloud Mammeri).
- [10] [http://medweb1.unige.ch/enseignement/apprentissage/module2/circ/apprentissage/intranet/cours/intro\\_ecg.pdf](http://medweb1.unige.ch/enseignement/apprentissage/module2/circ/apprentissage/intranet/cours/intro_ecg.pdf).
- [11] Houamed, I. (2011). Détection de l'Onde P dans un Signal ECG (Doctoral dissertation, Université de Batna 2).
- [12] D.ATTIAS, N.LELLOUCHE « ikb CARDIOLOGIE VASCULAIRE »Edition VG,7e edition 2016.
- [13] Tom, M. (1997). Mitchell: Machine Learning. 1997 Burr Ridge, 45(37), 870-877.
- [14] Mifdal, R. (2019). Application des techniques d'apprentissage automatique pour la prédiction de la tendance des titres financiers (Doctoral dissertation, École de technologie supérieure).
- [15] Moïse, K. K. Revue de littérature sur au moins trois techniques de machine learning.
- [16] Abbas, J., & Hamdad, A. (2020). Apprentissage automatique du dialecte algérien (Doctoral dissertation, Université Mouloud Mammeri).
- [17] Marref, N. (2013). Apprentissage Incrémental & Machines à Vecteurs Supports (Doctoral dissertation, Université de Batna 2).
- [18] Ferecatu, M., & Crucianu, M. Apprentissage, réseaux de neurones et modèles graphiques (RCP209).
- [19] Benbahria, R., & Megri, A. F. (2021). Prédiction Des Systèmes Par Apprentissage Automatique Dans Un Environnement Imprécis.

- [20] Mimoune, Z., & Ouahab, A. (2019). Développement d'une Architecture Basée sur l'Apprentissage Profond (Deep Learning) pour la Détection d'Intrusion dans les Réseaux (Doctoral dissertation, Université Ahmed Draïa-Adrar).
- [21] Anderson, D., & McNeill, G. (1992). Artificial neural networks technology. Kaman Sciences Corporation, 258(6), 1-83.
- [22] Touzet, C. (1998). Des réseaux de neurones artificiels à la robotique coopérative (Doctoral dissertation, Aix Marseille Université).
- [23] Hachemaoui, A. (2004). Application des réseaux de neurones artificiels dans la prédiction des propriétés des hydrocarbures purs et des fractions pétrolères (Doctoral dissertation).
- [24] Abdoune, A., & Amenouche, M. (2021). Simulation d'une pompe triplex avec simecape.(Doctoral dissertation, Université Mouloud Mammeri Tizi Ouzou).
- [25] Terbiche, Z., Mekkid, M., & Khaouane, L. (2021). Modélisation de la viscosité des liquides par intelligence computationnel.
- [26] Kim, P. (2017). Matlab deep learning with machine learning, neural networks and artificial intelligence. by Phil Kim.
- [27] Grégoire, F. (2018). Extraction de phrases parallèles à partir d'un corpus comparable avec des réseaux de neurones récurrents bidirectionnels.
- [28] Affi, M., & Latiri, C. (2022, October). Arabic Named Entity Recognition Using Variant Deep Neural Network Architectures and Combinatorial Feature Embedding Based on CNN, LSTM and BERT. In Proceedings of the 36th Pacific Asia Conference on Language, Informa

- [29] Khan, A., Sohail, A., Zahoor, U., & Qureshi, A. S. (2019). A survey of the recent architectures of deep convolutional neural networks. arXiv 2019. arXiv preprint arXiv:1901.06032.
- [30] Kattenborn, T., Leitloff, J., Schiefer, F., & Hinz, S. (2021). Review on Convolutional Neural Networks (CNN) in vegetation remote sensing. ISPRS journal of photogrammetry and remote sensing, 173, 24-49.
- [31] Sultana, F., Sufian, A., & Dutta, P. (2018, November). Advancements in image classification using convolutional neural network. Fourth International Conference on Research in Computational Intelligence and Communication Networks (ICRCICN) (pp. 122-129). IEE
- [32] Hammad, M., Abd El-Latif, A. A., Hussain, A., Abd El-Samie, F. E., Gupta, B. B., Ugail, H., & Sedik, A. (2022). Deep learning models for arrhythmia detection in IoT healthcare applications. Computers and Electrical Engineering, 100, 108011.
- [33] Vishwa, A., Lal, M. K., Dixit, S., & Vardwaj, P. (2011). Classification of arrhythmic ECG data using machine learning techniques. IJIMAI, 1(4), 67-70.
- [34] Rajkumar, A., Ganesan, M., & Lavanya, R. (2019, March). Arrhythmia classification on ECG using Deep Learning. In 2019 5th international conference on advanced computing & communication systems (ICACCS) (pp. 365-369). IEEE.
- [35] Izci, E., Ozdemir, M. A., Degirmenci, M., & Akan, A. (2019, October). Cardiac arrhythmia detection from 2d ecg images by using deep learning technique. In 2019 medical technologies congress (TIPTEKNO) (pp. 1-4). IEEE.
- [36] kaggle datasets download -d shayanfazeli/heartbeat

[37] Rosyda, S. S., & Purboyo, T. W. (2018). A review of various handwriting recognition methods. *International Journal of Applied Engineering Research*, 13(2), 1155-1164.

# Webographie

**[Web1]** Définition coeur –coeur, groupe madeinfutura, Retrieved may 9,2023, septembre 2021, website:<https://www.futura-sciences.com/sante/definitions/biologie-coeur-6849/#fonction-du-cur>.

**[Web2]** Jawad.A.sabha , August 23, 2018,Retrieved may 9,2023from website:<https://www.youtube.com/watch?v=C4u09wSNufk>

**[Web3]** l'activité électrique du cœur, Retrieved may 13, 2023, Publication date: August 23, 2016, Modified on May 21, 2021 from website: <https://fedecardio.org/je-m-informe/l-activite-electrique-du-coeur/>.

**[Web4]** Apprentissage Supervisé Vs. Non Supervisé,Zakariyaa ISMAILI ,Retrieved may 17, 2023,janvier 28, 2019,from website:<https://brightcape.co/apprentissage-supervise-vs-non-supervise/>

**[Web5]** What is Reinforcement Learning?: Components, Approaches, Markov Decision Process, Formalisation using MDP,Retrieved may 17, 2023,janvier 28, 2019,from website:<https://medium.com/@vishnuvijayanpv/what-is-reinforcement-learning-e5dc827c8564>

**[Web6]** Les neurones biologiqueset artificiels,Retrievedmay17,2023,from website:<https://iatpe2015.wordpress.com/le-fonctionnement/le-reseau-de-neurones-artificiel/les-neurones-biologiques-et-artificiels/>

**[Web7]** Intelligence mécanique,Ikram Chraibi Kaadoud,Retrieved may 2, 2023,from website:<https://scilogs.fr/intelligence-mecanique/reprenons-bases-neurone-artificiel-neurone-biologique/>

**[Web8]** RECURRENT NEURAL NETWORK (RNN),Retrieved may 30, 2023,from website:<https://medium.com/mllearning-ai/recurrent-neural-network-rnn-20a619190586>

**[Web 9]** Réseau \_neuronal\_ convolutif,Retrieved Juin 1,2023,from website:[https://fr.wikipedia.org/wiki/R%C3%A9seau\\_neuronal\\_convolutif](https://fr.wikipedia.org/wiki/R%C3%A9seau_neuronal_convolutif)

# Annexe :Outils d'implimentation du système

### 1. Introduction

Ce chapitre se concentre sur l'implémentation concrète du système intelligent de prédiction des maladies cardiaques. Nous aborderons ici les différents outils et bibliothèques utilisés pour développer et mettre en œuvre ce système. L'objectif de cette partie est de fournir un aperçu détaillé des ressources et des technologies qui ont été mobilisées pour créer un modèle fonctionnel et performant.

### 2. Présentation des outils utilisés

#### 2.1. Google collab

Google Collab est une plateforme de développement basée sur le cloud qui permet d'écrire, d'exécuter et de partager des notebooks Jupyter. Avec des fonctionnalités telles que l'accès au cloud, les librairies préinstallées, les ressources GPU et TPU gratuites, la collaboration en temps réel et le stockage intégré, Colab est devenu populaire pour l'apprentissage automatique, la science des données et le développement de logiciels.

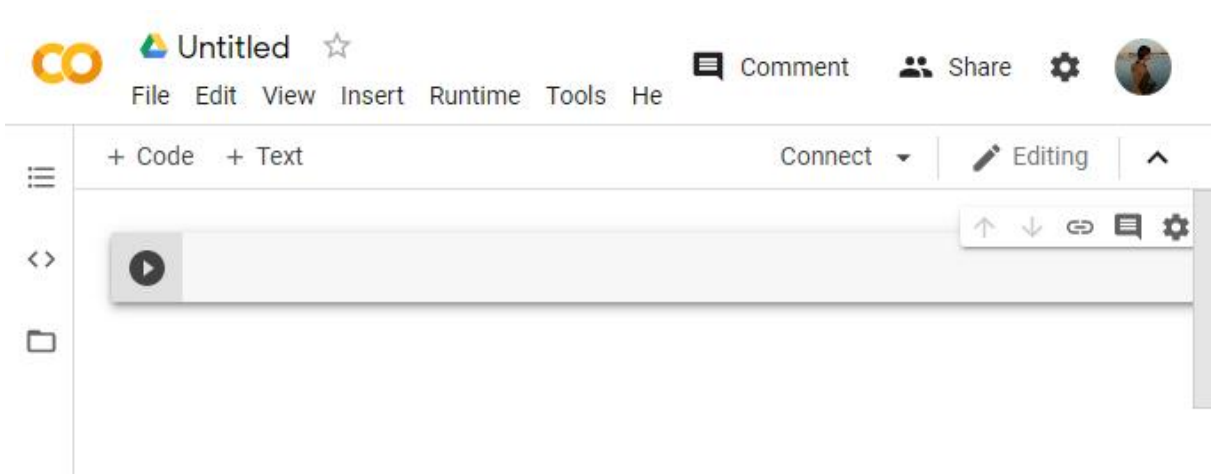


Figure 27:interface Google collab.

#### 2.2. Google drive

Google Drive est un service de stockage en ligne qui permet de sauvegarder et d'accéder à des fichiers et des documents depuis n'importe quel appareil connecté à Internet. Vous pouvez utiliser Google Drive pour stocker vos notebooks Jupyter, fichiers de code source, ensembles de données et autres ressources nécessaires à votre projet.

## Annexe :Outils d'implimentation du système

Google Drive et Google Colab sont des outils complémentaires qui permettent de stocker, partager et exécuter des notebooks Jupyter en ligne, facilitant ainsi la collaboration et le développement de projets de programmation.

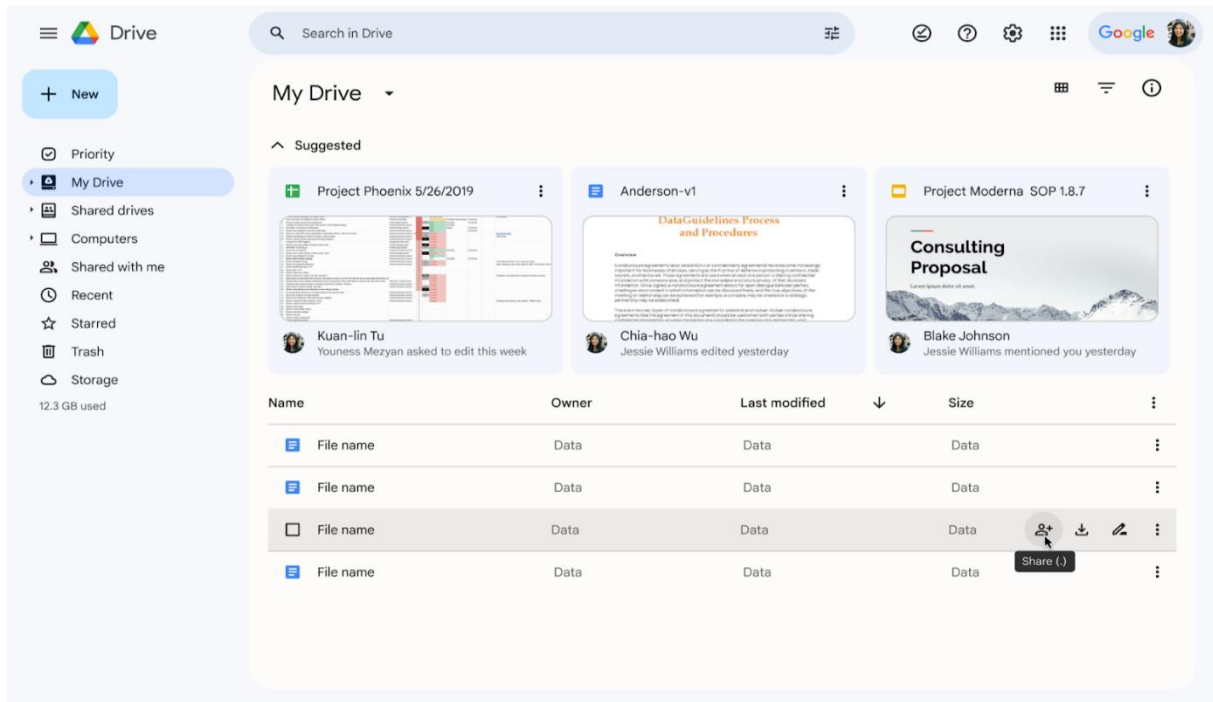


Figure 28:interface Google Drive

### 2.3. Python

Python est un langage de programmation interprété largement utilisé dans divers domaines. Il est fréquemment enseigné dans les lycées et les collèges comme langage de programmation d'introduction en raison de sa facilité d'apprentissage. Cependant, Python est également adopté par des professionnels du développement de logiciels dans des organisations prestigieuses telles que Google, la NASA et Lucas film Ltd.



Figure 29:Logo Python

### 2.4. Pytorch

PyTorch est une bibliothèque open source de logiciels utilisée pour le flux de données et la programmation différentielle dans diverses tâches. Elle est largement utilisée dans l'apprentissage par les réseaux de neurones. Développé en 2016 par Facebook, PyTorch est devenu rapidement populaire grâce à sa flexibilité et à sa facilité d'utilisation. La bibliothèque est conçue pour fonctionner sur plusieurs processeurs et est disponible sur différentes plateformes, y compris les appareils mobiles. Le nom "PyTorch" fait référence aux tableaux multidimensionnels appelés tenseurs (Tensor) utilisés couramment dans les réseaux de neurones. Aujourd'hui, PyTorch est utilisé par de grandes entreprises et organisations telles que Microsoft, Tesla, et l'Université de Stanford.



Figure 30:Logo PyTorch.

### 2.5. Pandas

Pandas est une bibliothèque open source de logiciels utilisée pour la manipulation et l'analyse des données dans diverses tâches. Elle est largement utilisée dans le domaine de la science des données et de l'analyse statistique. Développée en 2008 par Wes McKinney, Pandas offre une gamme complète d'outils et de structures de données puissantes pour traiter et manipuler des données tabulaires. La bibliothèque est conçue pour faciliter les opérations de nettoyage, de transformation et de visualisation des données, ainsi que pour effectuer des calculs statistiques avancés. Le nom "Pandas" est dérivé de "panel data", un terme utilisé en économie pour désigner des données multidimensionnelles. Aujourd'hui, Pandas est largement utilisé par des professionnels de la science des données, des chercheurs, des universitaires et des entreprises du monde entier pour analyser et explorer leurs données de manière efficace.



Figure 31 : pandas logo.

### 2.6. Numpy

NumPy est une bibliothèque open source de logiciels utilisée pour la manipulation et le calcul de tableaux multidimensionnels dans le langage de programmation Python. Elle est largement utilisée dans le domaine de la science des données, du calcul scientifique et de l'apprentissage automatique. Développée en 2005, NumPy offre une large gamme de fonctionnalités pour travailler avec des tableaux, y compris des opérations mathématiques avancées, des transformations de données, des fonctions statistiques et bien plus encore. La bibliothèque est conçue pour être rapide, efficace et polyvalente, permettant aux utilisateurs de manipuler et d'analyser de grandes quantités de données de manière optimisée. Le nom "NumPy" est dérivé de "Numeric Python", soulignant son objectif principal de fournir des outils numériques puissants pour le langage Python. Aujourd'hui, NumPy est largement adopté par les chercheurs, les scientifiques des données, les ingénieurs et les développeurs du monde entier pour effectuer des calculs numériques avancés et des manipulations de données de manière efficace.



Figure 32: Numpy logo .

### 2.7. Scikit-learn

Scikit-learn est une bibliothèque open source de machine Learning pour Python. Elle offre des outils simples et efficaces pour effectuer des tâches telles que la classification, la régression, le regroupement et la réduction de dimensionnalité. Scikit-learn est largement utilisée dans le domaine de la science des données et est appréciée pour sa facilité d'utilisation et sa documentation complète.



Figure 33: Scikit-learn logo.



## Annexe :Outils d'implimentation du système

développer, entraîner et évaluer notre modèle, ainsi que pour manipuler et visualiser les données avec facilité.

```
import pandas as pd
import numpy as np
import torch
import torch.nn as nn
import torch.nn.functional as F
import torch.optim as optim
from torch.utils.data import TensorDataset, DataLoader, WeightedRandomSampler
from sklearn.model_selection import train_test_split
import matplotlib.pyplot as plt
import seaborn as sns
from copy import deepcopy
from sklearn.metrics import confusion_matrix
from sklearn.metrics import classification_report
from torch.optim.lr_scheduler import StepLR
```

Figure 36:un morceau de code permettant d'importer les bibliothèques nécessaires.

- Nous avons utilisé la fonction `pd.read_csv ()` de la bibliothèque pandas pour charger un fichier CSV dans un DataFrame. Dans notre cas, le fichier CSV que nous avons chargé s'appelle "mitbih\_train.csv". Il est localisé dans le chemin spécifié.

```
#load csv files to pandas dataframe
train_df = pd.read_csv('/content/drive/MyDrive/dataset/mitbih_train.csv', header=None)
```

Figure 37:Chargement d'un fichier CSV dans un DataFrame avec pandas.

### 3.2. Extraction caractéristiques et Labels

- La dernière colonne du DataFrame `train_df` contient les étiquettes de données, qui sont décrites comme des catégories numérotées de 0 à 4. Nous pouvons utiliser cette colonne comme variable cible pour notre modèle.

## Annexe :Outils d'implimentation du système

```
#the last column is the data label described as category from 0 to 4
#so we can use it as a target
y_df = train_df.iloc[:, -1]
y_df = y_df.astype('int')
y_df.head()

0    0
1    0
2    0
3    0
4    0
Name: 187, dtype: int64
```

Figure 38: Extraction de la colonne de données cible du DataFrame.

- Les colonnes restantes du DataFrame `train_df` contiennent les données ECG, qui serviront de caractéristiques pour notre modèle

```
#the rest are the ECG datapoints so we use them as features
X_df = train_df.iloc[:, :-1]
X_df.head ()
```

Figure 39:Extraction des caractéristiques (features) du DataFrame.

### 3.3. Prétraitement des données

- Pour préparer les caractéristiques pour l'apprentissage du modèle, nous allons calculer certaines statistiques sur les données.

```
#perfect, we now have the input and target in the correct shape
X_mean = X_np.mean()
X_std = X_np.std()
X_max = X_np.max()
X_min = X_np.min()
print(f'mean before normalization: {X_mean}')
print(f'std: {X_std}')
print(f'max: {X_max}')

mean before normalization: 0.17428295254906354
std: 0.22632738719052128
max: 1.0
```

Figure 40:Calcul des statistiques des caractéristiques avant la normalisation.

## Annexe :Outils d'implimentation du système

```
print(X_norm.shape)
print(f'new mean is {X_norm.mean()}')
print(f'new std is {X_norm.std()}')

(87554, 1, 187)
new mean is 1.5088439287128576e-15
new std is 1.0000000000000004
```

Figure 41:Affichage de la forme des caractéristiques normalisées et des nouvelles statistiques.

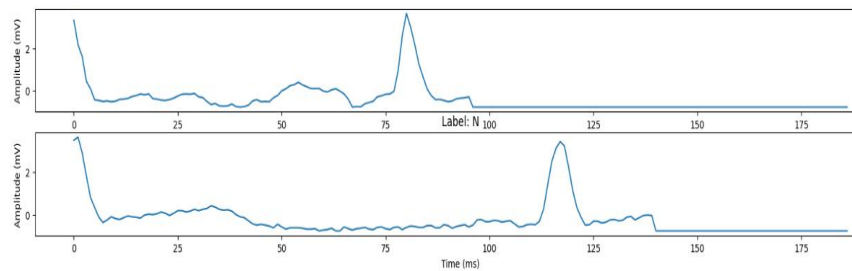


Figure 42:Visualisation des Données Normalisées par Échantillon.

### 3.4. L'architecture du CNN utilisé

- La fonction `nin_block` représente un bloc de convolution 1D. Ce bloc est composé d'une couche de convolution 1D suivie d'une normalisation par lots, d'une activation GELU et de deux couches de convolution supplémentaires avec des noyaux de taille 1.

```
#make some convolutional blocks to use in the model
def nin_block(in_channels, out_channels, kernel_size, padding, strides):
    return nn.Sequential(
        nn.Conv1d(in_channels, out_channels, kernel_size, strides, padding),
        nn.BatchNorm1d(out_channels),
        nn.GELU(),
        nn.Conv1d(out_channels, out_channels, kernel_size=1), nn.GELU(),
        nn.Conv1d(out_channels, out_channels, kernel_size=1), nn.GELU()
    )
```

Figure 43: Construction de Blocs de Convolutions pour le Modèle.

- l'architecture du modèle retourné par la fonction `get_model`. Il s'agit d'un modèle CNN avec plusieurs couches de convolutions, de max pooling et de couches linéaires pour la classification des 5 classes.

## Annexe :Outils d'implimentation du système

```
def get_model():
    return nn.Sequential( #input shape: (batch_size, 1, 187)
        nin_block(1, 48, kernel_size=11, strides=4, padding=0), #output shape: (batch_size, 48, 44)
        nn.MaxPool1d(3, stride=2), #output shape: (batch_size, 48, 21)
        nin_block(48, 128, kernel_size=5, strides=1, padding=2), #output shape: (batch_size, 128, 21)
        nn.MaxPool1d(3, stride=2), #output shape: (batch_size, 128, 10)
        nin_block(128, 256, kernel_size=3, strides=1, padding=1), #output shape: (batch_size, 256, 10)
        nn.MaxPool1d(3, stride=2), #output shape: (batch_size, 256, 4)
        nn.Dropout(0.4),
        #last layers for classification of 5 classes
        nin_block(256, 5, kernel_size=3, strides=1, padding=1), #output shape: (batch_size, 5, 4)
        nn.AdaptiveAvgPool1d(1), #output shape: (batch_size, 5, 1)
        nn.Flatten() #output shape: (batch_size, 5)
    )
```

Figure 44: Architecture du modèle à base de blocs de convolution 1D pour la classification.

### 3.5. Apprentissage

- La fonction `train` a pour but d'entraîner le modèle sur un nombre spécifié d'époques en utilisant les données d'entraînement.

```
def train(model, criterion, optimizer, train_dl, val_dl, epochs, lr_scheduler: StepLR=None):
    """trains the model for the given number of epochs
```

Figure 45: Fonction d'entraînement du modèle.

- Ce code effectue l'entraînement du modèle en utilisant les données d'entraînement et de validation, enregistrant les pertes et les meilleures performances obtenues.

```
train_losses, val_losses, best_model_params, best_accuracy = train (model,
criterion, optimizer, train_dl, val_dl, hyper_params['epochs'], lr_scheduler)
```

Figure 46:Entraînement et évaluation du modèle avec enregistrement des résultats.

- Ces deux instructions permettent de sauvegarder le modèle avec ses paramètres optimaux, ce qui permet de le charger ultérieurement pour une utilisation ou une évaluation ultérieure.

## Annexe :Outils d'implimentation du système

```
#load the best model parameters
model.load_state_dict(best_model_params)
#save the model to a file
torch.save(model.state_dict(), f'best_model{best_accuracy:.2f}.pt')
```

Figure 47:Chargement et sauvegarde du meilleur modèle.

### 3.6. Évaluation du modèle

- Les données de test n'ont pas été impliquées dans l'apprentissage, il s'agit donc de nouvelles données pour notre modèle.

```
#this section loads the test data and calculates the accuracy on the test set
test_df = pd.read_csv('/content/drive/MyDrive/dataset/mitbih_test.csv', header=None)
test_df.head()
```

Figure 48: Chargement des données de test.

Après cette étape, nous pouvons utiliser les données de test pour évaluer la performance du modèle, en utilisant différentes métriques telles que l'exactitude (accuracy), la précision (precision), le rappel (recall), le score F1 (F1 Score).

- La fonction `plot_cm` est appelée pour générer la matrice de confusion en utilisant le modèle entraîné (`model`), les données de test (`test_dl`) et les noms des classes (`class_names`).

```
#calculate the sensitivity and specificity for each class using scikit-learn
plot_cm(model, test_dl, class_names)
```

Figure 49:Fonction de matrice de confusion.

La matrice de confusion affiche les prédictions du modèle par rapport aux vraies étiquettes pour chaque classe. Cela permet d'évaluer la performance du modèle.

## Annexe :Outils d'implimentation du système

	precision	recall	f1-score	support
N	0.99	1.00	0.99	18118
S	0.89	0.81	0.84	556
V	0.97	0.95	0.96	1448
F	0.86	0.75	0.80	162
Q	1.00	0.99	0.99	1608
accuracy			0.98	21892
macro avg	0.94	0.90	0.92	21892
weighted avg	0.98	0.98	0.98	21892

Figure 50:Évaluation des performances du modèle.

La précision (accuracy) obtenue est environ : 98%.

## 4. Conclusion

Dans cette annexe, nous avons exploré en détail le développement de notre système, en fournissant une description complète de sa mise en œuvre. Nous avons examiné les différentes composantes et étapes clés qui ont contribué à la réalisation du système.

## **Annexe :Outils d'implimentation du système**