

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

**MINISTRE DE L'ENSEIGNEMENT SUPERIEURE ET DE LA
RECHERCHE SCIENTIFIQUE**

Université 20 Aout 1955- Skikda

Faculté des Science

Département informatique



MEMOIRE

En vue de l'obtention du diplôme de Master

Filière : Informatique

Spécialité : Intelligence Artificielle

Thème

**Prédiction de la mortalité dans les unités de soins
intensifs (USI) à l'aide d'algorithmes
d'apprentissage automatique**

Présenté par :

Kharbega SALIMA

Boudouab ZAHRA

Encadré par :

Dr. Lahsasna ADEL

2024-2025

Remerciement

Tout d'abord, nous remercions Allah le Tout-Puissant de nous avoir donné la force, la patience et la santé nécessaires pour accomplir ce travail

الحمد لله

*Nous exprimons notre profonde gratitude à notre superviseur, le Dr **Lahsasna Adel**, pour son soutien continu, ses précieux conseils, sa patience et sa motivation.*

Ses vastes connaissances et ses orientations ont été inestimables et nous ont beaucoup aidés à toutes les étapes de la recherche et de la rédaction de cette thèse.

Nous sommes fiers d'avoir eu l'opportunité de travailler sous sa direction et nous chérissons les leçons précieuses que nous avons apprises de lui.

Nous tenons également à remercier sincèrement les membres du jury pour avoir accepté d'examiner et d'évaluer notre travail, ainsi que tous les enseignants qui nous ont encouragés et soutenus tout au long de nos années d'études.

Nous remercions profondément nos chères familles, en particulier nos parents, pour leur soutien spirituel et moral constant tout au long de ce parcours. Nous remercions également nos amis pour leur amitié indéfectible et leur soutien inconditionnel.

Dédicaces



Grace à Dieu qui m'a éclairé le chemin vers cette
réussite

Je dédie ce modeste travail :

A mes chers parents « *Kharbega Aissa & Guerfi Hakima* »
qui ont sacrifié leurs bons moments afin de me
construire une vie harmonieuse pleine de réussite,

A mon cher frère ; *Fatah*,

A mes chères sœurs ; *Amira, Abir et Rachida*,

A ma grand-mère et mon grand-père

A ma nièce ; *Soudjoud*,

A mes amis ; *Ahlem, Soulef, Asma*,

A tous ceux qui ont participé de près ou de loin à la
réalisation de ce travail.

Salima...

Dédicaces



Louange à Dieu, par Sa grâce, les bonnes choses s'accomplissent, et par Sa bénédiction, les rêves se réalisent.

Aux personnes les plus chères de ma vie...

À mes parents bien-aimés, merci pour tout l'amour, les soins et le soutien que vous m'avez apportés. Vous êtes le soutien et la force sur lesquels je compte toujours.

À mes chers sœurs et frère, vous êtes mes amis éternels et les proches de mon cœur. Rien n'égale la force de la relation qui nous unit.

À mon cher mari, merci d'être mon partenaire de vie et mon meilleur ami. Votre présence dans ma vie rend tout meilleur.

À la famille de mon cher mari, merci de m'avoir accepté comme membre de votre famille et pour tout l'amour et le soutien que vous me témoignez.

À vous tous, je dédie cette note pour exprimer ma gratitude et mon profond amour pour vous.

Zahra ...

Résumé

Cette recherche se concentre sur l'utilisation des algorithmes d'apprentissage automatique pour prédire la mortalité dans les unités de soins intensifs (USI). L'objectif principal est d'améliorer la précision des prédictions en utilisant divers modèles et en évaluant leurs performances sur un ensemble de données de patients en USI. La méthodologie inclut la collecte et le prétraitement des données, l'application de différents modèles et l'évaluation de leurs performances.

Les résultats indiquent que le modèle de Forêt Aléatoire (Random Forest) offre la meilleure performance avec une précision de 87,93 % avant la sélection des caractéristiques et de 87,80 % après la sélection. Ce modèle surpasse les autres, tels que les Réseaux de Neurones Artificiels (ANN), les Machines à Vecteurs de Support (SVM), et les Arbres de Décision, en termes de précision de prédiction des décès en USI. Les caractéristiques les plus importantes pour la prédiction incluent NT-proBNP, la créatine kinase, la production d'urine, l'azote uréique, les plaquettes, les leucocytes, les lymphocytes, le bicarbonate, le gap anionique et l'acide lactique.

Cette recherche vise à fournir un outil efficace pour diagnostiquer les patients et aider les professionnels de la santé à offrir des traitements appropriés, tout en identifiant les facteurs de risque significatifs associés à la mortalité en USI.

Abstract

This study focuses on the use of machine learning algorithms to predict mortality in intensive care units (ICU). The primary objective is to improve prediction accuracy by utilizing various models and evaluating their performance on a dataset of ICU patients. The methodology includes data collection and preprocessing, application of different models, and performance evaluation.

The results indicate that the Random Forest model provides the best performance with an accuracy of 87.93% before feature selection and 87.80% after feature selection. This model outperforms other models such as Artificial Neural Networks (ANN), Support Vector Machines (SVM), and Decision Trees in terms of predicting mortality in ICUs. The most important features for prediction include NT-proBNP, creatine kinase, urine output, urea nitrogen, platelets, leukocytes, lymphocytes, bicarbonate, anion gap, and lactic acid.

This research aims to provide an effective tool for diagnosing patients and assisting healthcare professionals in delivering appropriate treatments, while identifying significant risk factors associated with mortality in ICUs.

ملخص

تركز هذه الدراسة على استخدام خوارزميات التعلم الآلي للتنبؤ بالوفيات في وحدات العناية المركزة (USI). الهدف الرئيسي هو تحسين دقة التنبؤات باستخدام نماذج مختلفة وتقييم أدائها على مجموعة بيانات لمرضى في وحدات العناية المركزة. تشمل المنهجية جمع البيانات ومعالجتها، وتطبيق النماذج المختلفة، وتقييم أدائها.

أظهرت النتائج أن نموذج الغابة العشوائية (Random Forest) يوفر أفضل أداء بدقة تصل إلى 87.93% قبل اختيار الميزات و87.80% بعد اختيار الميزات. يتفوق هذا النموذج على النماذج الأخرى مثل الشبكات العصبية الاصطناعية (ANN) وآلات متجهات الدعم (SVM) والأشجار القرار (Decision Trees) من حيث دقة التنبؤ بالوفيات في وحدات العناية المركزة. تشمل أهم الميزات للتنبؤ NT-proBNP، كيناز الكرياتين، إنتاج البول، النيتروجين اليوريا، الصفائح الدموية، الكريات البيض، الخلايا للمفاوية، البيكربونات، الفجوة الأيونية وحمض اللاكتيك.

تهدف هذه الدراسة إلى توفير أداة فعالة لتشخيص المرضى ومساعدة المهنيين الصحيين على تقديم العلاجات المناسبة، مع تحديد عوامل الخطر الهامة المرتبطة بالوفيات في وحدات العناية المركزة.

TABLE DES MATIERES

Remerciements.....	I
Dédicace.....	II
Résumé.....	IV
Abstract.....	V
ملخص	VI
Chapitre 01: Introduction	
1.1 Contexte et importance	1
1.2 Enoncé du problème	1
1.3 Objectifs de l'étude	2
1.4 Questions de Recherche	2
1.5 Portée et limitations	3
1.6 Structure de la thèse	4
Chapitre 02 : Revue de la littérature	
2.1 Introduction	5
2.2 Aperçu de l'environnement des soins intensifs	5
2.2.1 Le rôle et l'importance de la prédiction de la mortalité	6
2.2.2 Défis dans la Gestion des patients en USI	7
2.3 Prédiction de la mortalité en USI	8
2.3.1 Systèmes de notation traditionnels	8
2.3.2 Les approches d'apprentissage automatique	9
2.3.2.1 Apprentissage supervisé.....	10
2.3.2.2 Apprentissage non supervisé.....	11

2.3.2.3 Apprentissage semi-supervisé	11
2.3.2.4 Apprentissage par renforcement.....	12
2.3.3 Apprentissage profond (Deep Learning)	13
2.4 Etude précédentes sur la prédiction de la mortalité en USI utilisant l'apprentissage automatique	13
2.5 Conclusion	17
Chapitre 03 : Méthodologie	
3.1 Introduction	18
3.2 Collecte et prétraitement des Données	19
3.2.1 Description des ensembles de données utilisés	19
3.2.2 Les étapes de Prétraitement	23
3.3 Modèles d'apprentissage automatique	24
3.3.1 Modèles d'apprentissage automatique traditionnel	24
3.3.1.1 Les réseaux de neurones (Artificial Neural Networks)	24
3.3.1.2 Arbre de décision	25
3.3.1.3 Machine à vecteur de support (Support Vector Machines)	25
3.3.1.4. Forêt aléatoire (Random Forest)	26
3.3.1.5 K-plus proches voisins (KNN)	26
3.3.1.6 Régression logistique	27
3.3.1.7 Extreme Gradient boosting (XGBoost)	27
3.3.2 Modèles d'apprentissage profond	27
3.3.2.1 Les réseaux neuronaux profonds (DNN)	28
3.3.2.2 Les réseaux de neurones récurrents (RNN)	28

3.3.2.3 Le Long Short Term Memory (LSTM)	28
3.3.2.4 Les réseaux de neurones convolutifs (CNN)	29
3.4 Entraînement et validation du modèle.....	29
3.4.1 La méthode de validation croisée	29
3.5 Métriques d'évaluation.....	31
3.6 Analyse de l'importance des Caractéristiques	31
3.7 Conclusion	33
Chapitre 04 : Implémentation du Système	
4.1 Introduction	34
4.2 Aperçu de l'architecture du système	34
4.2.1 L'interface principale	34
4.2.2. Le module de comparaison	35
4.2.3 Le module de déploiement	38
4.3 Diagramme de bloc illustrant les principaux composants	40
4.4 Technologies et Outils.....	40
4.4.1 Langages de Programmation	40
4.4.1.1 Python	40
4.4.1.2 Pourquoi Python est-il si populaire ?.....	41
4.4.2 Frameworks et bibliothèques	42
4.4.2.1 Frameworks utilisés	42
4.4.2.2 Bibliothèques utilisés	42
4.4.3 Environnement de développement	44
4.4.3.1 Matériel utilisé	44

4.4.3.2 Logicielle de l'environnement de développement.....	44
4.5 Interface Utilisateur	45
4.5.1 Considérations de Conception pour l'Interface Utilisateur	45
4.5.2 Captures d'Écran de l'Interface Utilisateur	47
4.5.2.1 Le module de comparaison	47
4.5.2.2 Le module de déploiement	47
4.5.3 Description des Fonctionnalités Fournies par l'Interface Utilisateur	49
4.6 Conclusion	51
Chapitre 05 : Résultats et discussion
5.1 Introduction	52
5.2 Présentation de la performance du modèle sur les ensembles de données de test	52
5.3 Analyse comparative avec les performances des modèles	52
5.4 Discussion des résultats	53
5.5 Implication pour la pratique clinique	55
Chapitre 06 : Conclusion et Perspectives
6.1 Conclusion	59
6.2 Limitations	59
6.3 Perspectives Futures	59
Bibliographie	61

LISTE DES FIGURES

Figure 2.1 Les différents types d'apprentissage automatique.....	10
Figure 2.2 Schéma d'un modèle supervisé.....	11
Figure 2.3 Schéma d'un modèle non supervisé.....	11

Figure 2.4 Schéma d'un modèle semi-supervisé.....	12
Figure 2.5 Schéma d'un modèle par renforcement.....	13
Figure 3.1 Un diagramme des étapes utilisées pour développer notre système de prédiction..	19
Figure 3.2 Structure générale du réseau de neurones artificiels avec deux couches cachées...	24
Figure 3.3 Exemple de validation croisée	30
Figure 4.1 Diagramme de bloc illustrant les principaux composants.....	40
Figure 4.2 L'interface d'accueil pour le système de prédiction de la mortalité dans les USI...	47
Figure 4.3 Interface du module de comparaison de prédiction de la mortalité dans les USI..	48
Figure 4.4 Interface du module de déploiement.....	49

LISTE DES TABLEAUS

Tableau 3.1 Les caractéristiques spécifiques de base de données.....	20
Tableau 5.1 La performance du modèle sur les ensembles de données de test	52

Chapitre 01 : Introduction

1.1 Contexte et importance

La prédiction de la mortalité dans les unités de soins intensifs (USI) revêt une importance cruciale pour les professionnels de la santé et les systèmes de santé. Les USI sont des environnements critiques où les patients présentent souvent des conditions médicales graves qui nécessitent une surveillance étroite et des interventions intensives. La capacité à prédire avec précision la probabilité de mortalité des patients admis en USI peut avoir un impact significatif sur les décisions cliniques, telles que le niveau de soins à fournir, l'allocation des ressources et les discussions avec les patients et les familles sur les options de traitement et les pronostics.

Les avancées dans les techniques d'apprentissage automatique offrent de nouvelles possibilités pour développer des modèles de prédiction de la mortalité en USI. En utilisant des algorithmes sophistiqués et des ensembles de données riches en informations médicales, il devient possible de créer des outils de prédiction précis et personnalisés, adaptés aux besoins spécifiques des patients et des environnements de soins.

Cependant, malgré les progrès réalisés, il reste des défis à relever dans le domaine de la prédiction de la mortalité en USI, notamment la sélection des caractéristiques pertinentes, la gestion des données manquantes, la validation des modèles et leur intégration dans les flux de travail cliniques.

1.2 Enoncé du problème

La problématique centrale de cette étude consiste à déterminer quel algorithme d'apprentissage automatique est le plus efficace pour prédire la mortalité dans les unités de soins intensifs (USI). Nous examinerons des algorithmes couramment utilisés, notamment les réseaux de neurones artificiels (ANN), les machines à vecteurs de support (SVM), les arbres de décision, les forêts aléatoires, les k-plus proches voisins (KNN), Régression logistique, XGBoost, Les réseaux neuronaux profonds (DNN), Les réseaux de neurones récurrents (RNN), Le Long Short-Term Memory (LSTM) et Les réseaux de neurones convolutifs (CNN).

L'objectif est déterminer quel modèle offre les performances les plus précises et fiables pour prédire la mortalité des patients dans les USI, afin d'améliorer la prise en charge clinique et les résultats des patients. En parallèle, il sera essentiel d'identifier les facteurs de risque significatifs associés à la mortalité en USI, afin d'optimiser les interventions médicales et de réduire les taux de mortalité dans ces environnements critiques.

1.3 Objectifs de l'étude

Les objectifs de cette étude sont d'évaluer un ensemble d'algorithmes pour sélectionner l'algorithme de prédiction le plus précis pour prédire la mortalité, et d'utiliser ce dernier pour développer un système efficace. De plus, l'étude vise à identifier les caractéristiques les plus importantes liées à la prédiction, afin de fournir des informations sur les facteurs de risque pouvant être pris en considération pour atténuer la mortalité.

1.4 Questions de Recherche

Cette étude vise à explorer l'efficacité des algorithmes d'apprentissage automatique dans la prédiction de la mortalité dans les unités de soins intensifs (USI). Voici les questions de recherche que nous aborderons pour atteindre cet objectif :

1. Quels sont les algorithmes d'apprentissage automatique les plus efficaces pour prédire la mortalité dans les unités de soins intensifs (USI) ?
2. Comment ces algorithmes peuvent-ils être comparés en termes de précision et de fiabilité des prédictions ?
3. Quelles caractéristiques ou variables sont les plus importantes pour la prédiction de la mortalité en USI ?
4. Quels sont les défis et les solutions pour la gestion des données manquantes dans le contexte des modèles de prédiction en USI ?

1.5 Portée et limitations

Portée : Le projet vise à explorer et évaluer divers algorithmes d'apprentissage automatique et profond pour la prédiction de la mortalité des patients dans les unités de soins intensifs (USI). Les principales activités du projet incluent :

1. Collecte et Préparation des Données : Recueillir des ensembles de données riches en informations médicales pertinentes, nettoyer et préparer ces données pour l'analyse.
2. Sélection des Caractéristiques : Identifier et sélectionner les caractéristiques médicales pertinentes pour la prédiction de la mortalité en USI.
3. Implémentation des Algorithmes : Mettre en œuvre les algorithmes d'apprentissage automatique sélectionnés.
4. Évaluation des Modèles : Évaluer la performance de chaque algorithme en utilisant le taux de précision de la classification.
5. Sélection du Meilleur Modèle : Identifier l'algorithme offrant les performances les plus précises et fiables pour la prédiction de la mortalité.
6. Développement d'un Système de Prédiction : Utiliser l'algorithme sélectionné pour développer un système de prédiction opérationnel.
7. Analyse des Facteurs de Risque : Identifier et analyser les facteurs de risque significatifs associés à la mortalité en USI pour fournir des recommandations cliniques.

Limites : Le projet est délimité par les facteurs suivants :

1. Ensembles de Données : Les analyses sont limitées aux données disponibles et leur qualité. Les résultats dépendent fortement des ensembles de données utilisés.
2. Algorithmes d'Apprentissage : L'étude se concentre uniquement sur les algorithmes spécifiés (ANN, SVM, arbres de décision, forêts aléatoires, KNN, régression logistique, xgBoost, DNN, RNN, LSTM, CNN) et n'inclut pas d'autres techniques d'apprentissage automatique ou profond ou d'IA.

3. Contextes Cliniques : L'applicabilité des résultats est limitée aux contextes des USI et peut ne pas s'étendre à d'autres environnements médicaux ou patients en dehors des USI.

4. Validation des Modèles : La validation se fait à l'aide de techniques standards telles que la validation croisée.

1.6 Structure de la thèse

A l'exception de l'introduction, ce mémoire se compose de cinq autres chapitres organisés comme suit:

Le deuxième chapitre passe en revue la littérature concernant les USI, la prédiction de la mortalité à l'aide des systèmes de notation traditionnels et de l'apprentissage automatique, ainsi que les études précédentes dans ce domaine.

Le troisième chapitre détaille la méthodologie, y compris la collecte et le prétraitement des données, les modèles utilisés, l'entraînement et l'évaluation, et l'analyse de l'importance des caractéristiques.

Le quatrième chapitre aborde la mise en œuvre du système, avec une description de l'architecture, des technologies et des outils utilisés, ainsi que de l'interface utilisateur.

Le cinquième chapitre présente les résultats des modèles et discute de leur performance.

Enfin, le sixième chapitre résume les résultats, identifie les limitations et propose des perspectives futures.

Chapitre 02 : Revue de la littérature

2.1 Introduction

Dans ce chapitre, Nous commencerons par une vue d'ensemble de l'environnement des soins intensifs et du rôle de la prédiction de la mortalité dans l'amélioration de la gestion des patients. Nous discuterons des défis auxquels sont confrontés les médecins et les infirmiers dans la prise en charge des patients en USI, ainsi que de l'importance d'une prédiction précise de la mortalité comme outil pour améliorer les soins de santé. Nous examinerons également les systèmes traditionnels d'évaluation des patients et comment les méthodes d'apprentissage automatique peuvent contribuer à améliorer ces processus d'évaluation et de prédiction.

Nous examinerons les études antérieures qui ont utilisé des méthodes d'apprentissage automatique pour améliorer la prédiction de la mortalité en USI, ouvrant ainsi la voie à la présentation de notre cadre de travail pour cette étude et des méthodes utilisées pour atteindre ses objectifs.

2.2 Aperçu de l'environnement des soins intensifs

L'unité de soins intensifs est un service complexe et riche en informations. Les patients admis dans les unités de soins intensifs nécessitent une surveillance étroite et continue en raison de la gravité élevée de la maladie et du potentiel de progression rapide de la maladie. Les patients en soins intensifs sont également hétérogènes, souffrant souvent de multiples problèmes concomitants et moins nombreux que les patients se présentant dans des spécialités à système unique. La recherche est donc nécessairement limitée à la fois par l'hétérogénéité et par le petit nombre d'échantillons. Pour cette raison et d'autres encore, les données probantes sur la pratique des soins intensifs sont moins bien développées que pour certaines autres spécialités aiguës. La combinaison unique de riches sources de données issues de la surveillance et d'une population de patients complexe et hétérogène rend le contexte des soins intensifs particulièrement bien adapté à la mise en œuvre d'un système assistant basé sur les données qui analyse de grandes quantités de données brutes qui pourraient être négligées par les experts humains. L'utilisation des données des soins intensifs pour la prévision précoce de la mortalité constitue un domaine d'investigation attrayant, à la fois pour des raisons de qualité et de coût.

2.2.1 Le rôle et l'importance de la prédiction de la mortalité

Le taux de mortalité est l'un des résultats les plus couramment utilisés pour mesurer la qualité des soins et, à ce titre, la tâche de prévision de la mortalité a maintenu une présence substantielle dans le secteur de la santé. La prédiction de la mortalité en unité de soins intensifs remplit diverses fonctions pour les différentes parties prenantes des soins de santé, comme expliqué ci-dessous.

Au niveau individuel

Il cherche à fournir une estimation empirique impartiale du risque pour la prise de décision clinique. Les médecins disposant d'estimations quantitatives de la probabilité de mortalité peuvent prioriser les soins aux patients et préparer un plan d'action et un pronostic ciblés. La probabilité de mortalité peut également fournir une mesure approximative de l'efficacité du traitement. Un score de physiologie aiguë plus élevé le troisième jour en soins intensifs plutôt que le premier jour peut identifier des soins potentiellement inefficaces chez les patients entrés avec une probabilité de mortalité élevée.

Il peut également être utilisé pour évaluer l'adéquation des patients à de nouveaux traitements. De plus, la probabilité de mortalité fournit au médecin et aux patients et/ou à leur famille une base pour décider de l'agressivité du traitement ou de l'arrêt du maintien de la vie.

Au niveau organisationnel

La probabilité de mortalité est utilisée pour quantifier la gravité de la maladie aux fins de l'allocation des ressources. Les soins intensifs étant une source de coûts importante, la probabilité de mortalité doit être prise en compte pour garantir une allocation efficace des ressources. Les affirmations que l'identification des patients qui pourraient ne pas survivre à leur séjour aux soins intensifs sera efficace, des économies considérables. Il facilite également les tâches administratives et indique qu'un score de risque de mortalité peut aider à catégoriser les patients pour les registres ou les systèmes de remboursement et va même jusqu'à anticiper un rôle pour lui dans l'accréditation des unités individuelles de soins intensifs. La probabilité de

mortalité sert également de base à la stratification des patients. La stratification du risque basée sur le score de gravité de la maladie et la probabilité de mortalité fournit des normes de comparaison entre les groupes pour les évaluations de la qualité et les études de traitement par les équipes de recherche impliquées dans l'USI.

Au niveau des infrastructures sanitaires régionales et nationales

La qualité des hôpitaux est évaluée et par conséquent récompensée par les agences régionales et nationales à l'aide de diverses mesures, notamment les taux de mortalité ajustés au risque (RAMR). Le RAMR d'un hôpital est son taux de mortalité brut qui a fait l'objet d'un ajustement statistique du risque pour tenir compte de toute anomalie dans la population prévue de l'hôpital. Pour calculer l'ajustement statistique du risque en fonction du taux de mortalité, un rapport entre la mortalité observée et prévue est utilisé. L'Agence pour la recherche et la qualité des soins de santé (AHRQ) considère un ensemble de 101 mesures de qualité, dont 11 ont été approuvées par le National Quality Forum (un organisme à but non lucratif qui s'occupe de mesurer et de rendre compte de la qualité des soins de santé). Au moins 8 des 11 paramètres étaient liés aux taux de mortalité ajustés en fonction du risque. De plus, les Centres for Medicare et Medicaid Services (CMS) utilisent trois types de taux de mortalité ajustés au risque dans le cadre de leur évaluation de la qualité des soins de santé dans le cadre de leur programme d'achat basé sur la valeur des hôpitaux (qui récompense les hôpitaux pour la meilleure qualité des soins fournis aux bénéficiaires de Medicare). Cela signifie qu'une prévision précise de la mortalité est essentielle à un taux de mortalité représentatif ajusté au risque. La prévision de la mortalité joue donc un rôle dans la formulation d'un indicateur illustratif de la qualité des soins qui fournit un moyen juste et équitable de comparaison et d'évaluation des hôpitaux [1].

2.2.2 Défis dans la Gestion des patients en USI

La prise en charge des patients en unité de soins intensifs (USI) présente divers défis que les professionnels de la santé doivent relever. Voici un résumé des défis mentionnés :

1. Augmentation du nombre de patients âgés : Cela nécessite une formation spécifique pour les intensivistes afin de comprendre les changements physiologiques liés au

vieillesse et de prendre en compte les résultats fonctionnels et la morbidité en plus de la mortalité [2].

2. Impact de la pandémie de COVID-19 : La pandémie a révélé des problèmes tels que la gestion inefficace des ressources, les faiblesses organisationnelles et le stress mental chez les prestataires de soins de santé dans les unités de soins intensifs [3].

3. Augmentation du nombre de survivants nécessitant un soutien organique prolongé après une maladie aiguë : Connu sous le nom de maladie critique chronique (CCI), cela pose des défis uniques en raison de complications telles que les infections secondaires, la myopathie et la neuropathie, nécessitant une approche multidisciplinaire pour des soins individualisés [4].

4. Gestion de la douleur en unité de soins intensifs : Cela nécessite des outils d'évaluation appropriés et une connaissance des différents types de douleur afin de se conformer aux directives des meilleures pratiques [5].

2.3 Prédiction de la mortalité en USI

2.3.1 Systèmes de notation traditionnels

La prédiction de la mortalité peut apparemment ressembler à un problème difficile de classification binaire, mais, en réalité, elle est plus probabiliste. La probabilité de mortalité est dérivée d'un continuum de valeurs qui sont positivement corrélées à l'aggravation de l'état d'un patient et, à ce titre, la plupart des systèmes de notation de « gravité de la maladie » se sont révélés être une mesure fiable de la probabilité de mortalité. Les systèmes de notation de gravité sont des méthodes ancrées dans la médecine factuelle et peuvent être générales ou spécifiques à une maladie. Certains systèmes de notation généraux utilisés sont APACHE II, APACHE III, SAPS II et MPM II. Le score de gravité de la maladie est généralement obtenu en résumant certaines caractéristiques basées sur des mesures et des attributs physiologiques du patient et/ou de son admission et de la probabilité de mortalité est dérivé de ce score de gravité final. Cependant, certains de ces systèmes sont plus fondés statistiquement sur les pondérations de certaines caractéristiques sont déterminées par régression logistique multiple.

Les systèmes susmentionnés sont principalement utilisés au niveau individuel et organisationnel. La mortalité prévue pour la déclaration des taux de mortalité ajustés en fonction du risque à des niveaux plus élevés est calculée en utilisant des protocoles établis par le ministère de la Santé et des Services sociaux des États-Unis (l'organisme directeur du CMS et de l'AHRQ). Les protocoles impliquent pour la plupart systématiquement facteurs de risque choisis en combinaison avec une régression logistique qui a été formulés à travers études empiriques subjectives.

L'évaluation des performances de la prévision de la mortalité est généralement décrite par la discrimination et l'étalonnage. La discrimination fait référence à la capacité du prédicteur à différencier ceux qui vont mourir de ceux qui survivront et l'étalonnage fait référence à la précision des probabilités prédites par rapport aux probabilités réelles (qualité de l'ajustement) [1].

2.3.2 Les approches d'apprentissage automatique

L'apprentissage automatique c'est un domaine d'étude qui confère aux ordinateurs la capacité d'apprendre sans être explicitement programmés. Il fait partie de l'intelligence artificielle. Ces algorithmes construisent des modèles basés sur des données d'entraînement, également appelées "données d'entraînement", afin de réaliser des prédictions ou des prises de décisions sans avoir besoin d'une programmation explicite à cet effet. Les algorithmes d'apprentissage automatique sont largement utilisés dans divers domaines tels que la médecine et la vision par ordinateur.

L'apprentissage automatique peut être divisé en quatre catégories principales: L'apprentissage supervisé, l'apprentissage non supervisé, l'apprentissage semi-supervisé et l'apprentissage par renforcement [6].

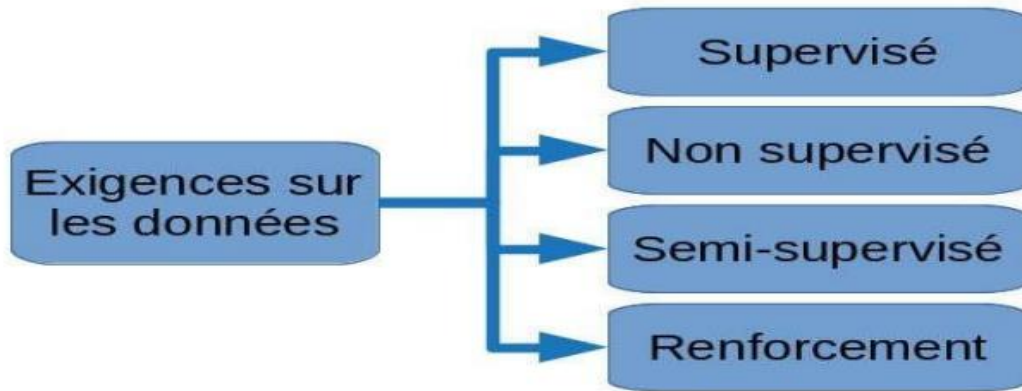


Figure 2.1 Les différents types d'apprentissage automatique

2.3.2.1 Apprentissage supervisé

L'apprentissage supervisé est la tâche d'apprentissage automatique qui consiste à apprendre une fonction qui associe une entrée à une sortie sur la base d'exemples de paires d'entrée-sortie. Il déduit une fonction à partir de données d'entraînement étiquetées, qui sont un ensemble d'exemples d'entraînement. Les algorithmes d'apprentissage supervisé sont ceux qui nécessitent une assistance externe. L'ensemble de données d'entrée est divisé en un ensemble d'entraînement et un ensemble de test. L'ensemble d'entraînement contient une variable de sortie qui doit être prédite ou classée. Tous les algorithmes apprennent certains types de schémas à partir de l'ensemble de données d'entraînement et les appliquent à l'ensemble de données de test pour la prédiction ou la classification. Il existe deux principaux domaines dans lesquels l'apprentissage automatique supervisé est utile: les problèmes de classification et les problèmes de régression. La classification consiste à prendre une valeur d'entrée et à l'affecter à une valeur discrète. Dans les problèmes de classification, la sortie est généralement une classe ou une catégorie [7].

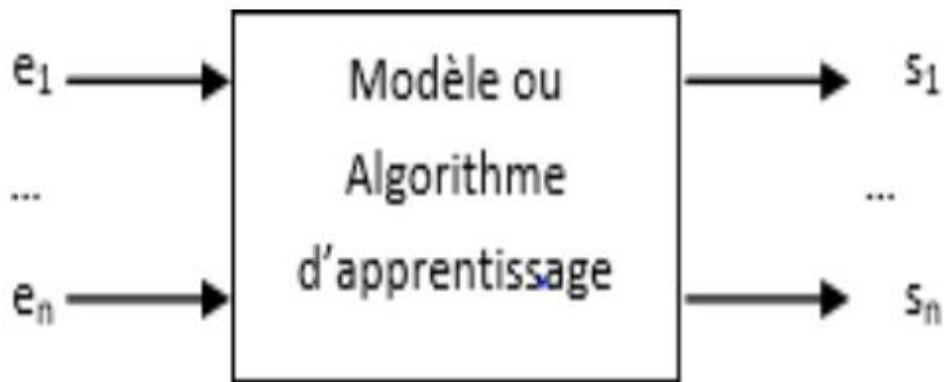


Figure 2.2 Schéma d'un modèle supervisé

2.3.2.2 Apprentissage non supervisé

Il s'agit d'un apprentissage qui se fait sans l'utilisation d'étiquettes de données. Dans ce cas, apprend les représentations internes ou les fonctionnalités importantes afin de découvrir des relations ou une structure inconnue dans les données d'entrée. Le clustering, la réduction de la dimensionnalité et les techniques génératives sont souvent considérés comme des approches d'apprentissage non supervisé [8].

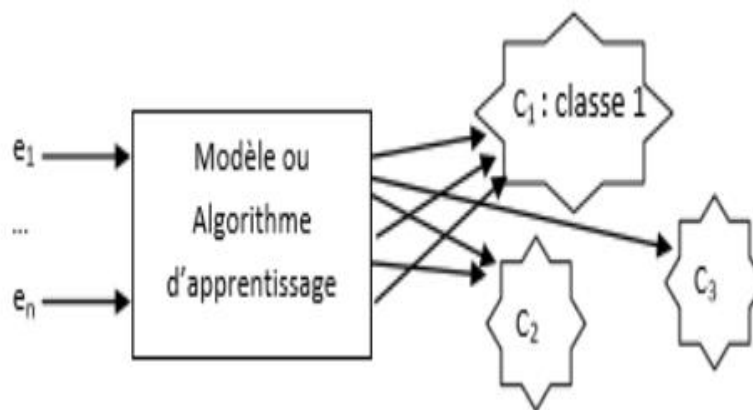


Figure 2.3 Schéma d'un modèle non supervisé

2.3.2.3 Apprentissage semi-supervisé

L'apprentissage semi-supervisé ou semi-supervised learning se situe entre le supervisé et le non supervisé. En effet, dans ce type d'apprentissage, nous avons un ensemble de données étiquetées et non étiquetées. Ce qui est intéressant avec cette méthode est que nous laissons une certaine liberté à notre modèle. En effet, nous lui

donnons des données sur lesquelles il va pouvoir extraire différentes caractéristiques et sur d'autres, il va devoir trouver des caractéristiques qui lui permettent de s'améliorer.

Ce type d'apprentissage est intéressant lorsque nous avons énormément de données. En effet, cela peut être fastidieux d'étiqueter l'intégralité de nos données. Cette méthode d'apprentissage nous permet d'éviter ce problème [8].

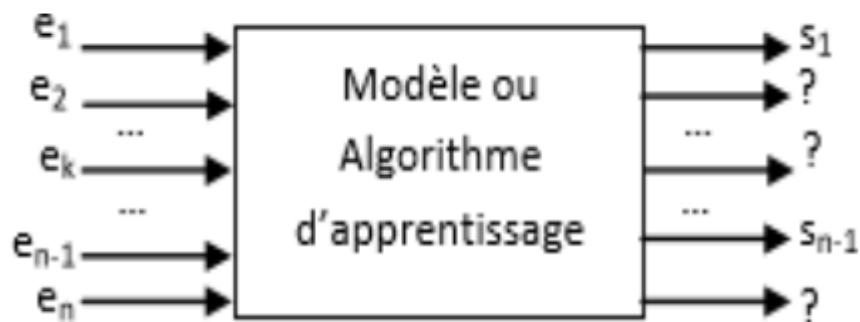


Figure 2.4 Schéma d'un modèle semi-supervisé

2.3.2.4 Apprentissage par renforcement

L'apprentissage par renforcement est un quatrième paradigme d'apprentissage automatique, aux côtés de l'apprentissage supervisé et de l'apprentissage non supervisé. Un des défis qui se posent dans l'apprentissage par renforcement, et pas dans d'autres types d'apprentissage, est le compromis entre exploration et exploitation. Pour obtenir beaucoup de récompenses, un agent d'apprentissage par renforcement doit préférer les actions qu'il a déjà essayées et trouvées efficaces pour produire des récompenses. Mais pour découvrir de telles actions, il doit essayer des actions qu'il n'a pas sélectionnées auparavant. L'agent doit exploiter ce qu'il sait déjà pour obtenir des récompenses, mais il doit également explorer pour améliorer ses sélections d'actions à l'avenir. Le dilemme est que ni l'exploration ni l'exploitation ne peuvent être poursuivies exclusivement sans échouer dans la tâche. L'agent doit essayer une variété d'actions et privilégier progressivement celles qui semblent être les meilleures. Une autre caractéristique clé de l'apprentissage par renforcement est qu'il considère explicitement l'ensemble du problème d'un agent dirigé vers un objectif interagissant avec un environnement incertain [8].

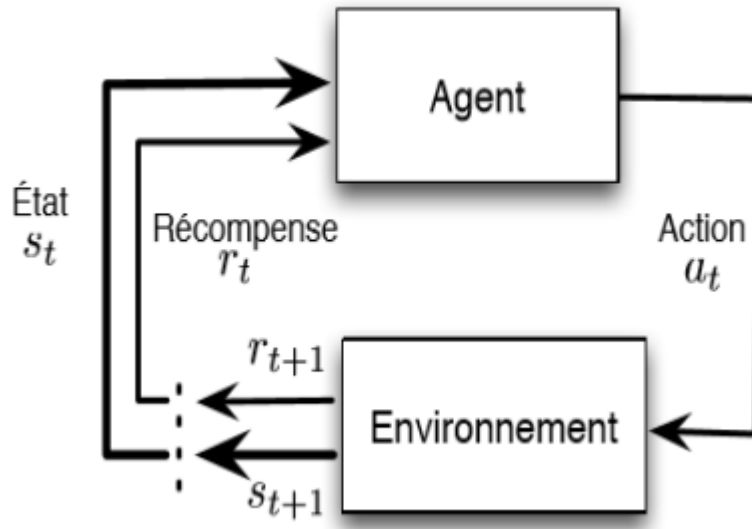


Figure 2.5 Schéma d'un modèle par renforcement

2.3.3 Apprentissage profond (Deep Learning)

L'apprentissage profond est un sous-ensemble de l'apprentissage automatique, utilise des Réseaux de Neurones Artificiels qui sont inspirés par la biologie du cerveau humain, et cela permet de créer des algorithmes très efficaces pour résoudre les problèmes de classification. En général, le Deep Learning est un sous-domaine de l'Intelligence Artificielle qui repose sur l'utilisation de réseaux de neurones complexes et de grandes quantités de données précises, afin de permettre aux machines d'apprendre de manière autonome, comme le ferait un être humain. Le Deep Learning est considéré comme étant à la pointe de l'IA et est l'une des technologies les plus passionnantes des dernières décennies [9].

2.4 Etude précédentes sur la prédiction de la mortalité en USI utilisant l'apprentissage automatique

Dans cette section, nous citons quelques travaux connexes dans le domaine de prédiction la mortalité en USI:

Algorithmes d'arbre de décision (DT)

Dans ce travail Crawford et al. [10] ont conclu qu'un algorithme d'arbre de décision fournit un résultat d'exploration cliniquement acceptable pour prédire la propagation des ganglions lymphatiques chez les hommes atteints de carcinome de la prostate localisé sur le plan clinique. En outre, Kim et al. [11] ont comparé l'utilisation de DT, des réseaux neuronaux artificiels (ANN) et SVM avec APACHE III pour prédire la mortalité, et ont rapporté que DT a donné la meilleure performance

Algorithmes de forêt aléatoire (RF)

Dans ce travail Sadeghi et al. [12] ont tenté de prédire la mortalité précoce à l'hôpital. Cette étude présente une nouvelle méthode pour prédire la mortalité en utilisant 12 caractéristiques extraites des signaux cardiaques des patients au cours de la première heure d'admission en soins intensifs en utilisant la base de données MIMIC-III. Les caractéristiques extraites sont alimentées dans huit classificateurs : arbre de décision, discriminant linéaire, régression logistique, SVM, forêt aléatoire, arbres boostés, SVM gaussien et KNN. L'algorithme de forêt aléatoire a estimé à la fois la précision et l'interprétabilité mieux que les autres algorithmes. De plus, Ghose et al. [13] ont utilisé RF, SVM et LR, comparés à SAPS, et ont conclu que RF atteint une aire sous la courbe (AUC) de 87 % ; ils ont également constaté que RF surpasse certains des modèles prédictifs de pointe basés sur SVM et LR. En outre, Awed et al. [14] ont conçu un modèle d'ensemble pour la prédiction de la mortalité, qui comprend RF, DT et Naive Bayes (NB). Ils ont appliqué leur modèle d'ensemble sur 20 caractéristiques extraites au cours des 6 premières heures de l'admission du patient. Il atteint un score AUROC de 82 %.

Algorithmes de SVM

Dans ce travail, Calvert et al. [15] ont utilisé des variables cliniques collectées à partir des dossiers de santé électroniques pour prédire la mortalité en unité de soins intensifs, obtenant une forte précision prédictive de 80 %. De plus, Ghose et al. [13] ont utilisé RF, SVM et LR, comparés à SAPS, et ont conclu que RF a atteint une aire sous la courbe (AUC) de 87 %, surpassant certains des modèles prédictifs de pointe basés sur SVM et LR. En outre, Member et al. [16] ont illustré l'utilisation des

résultats SVM pour améliorer la prédiction de la durée de séjour du patient en utilisant des règles d'association.

Algorithmes de régression logistique (LR)

Dans ce travail Ball et al [17]. Ont utilisé LR pour développer une approche de prédiction clinique de la mortalité basée uniquement sur les caractéristiques de la fréquence cardiaque. Leur travail fournit des résultats acceptables pour les patients âgés canadiens qui souffrent de problèmes cardiaques. De plus, Anand et al. [18] Ont recommandé l'utilisation de LR pour prédire la mortalité des patients adultes admis pour une chirurgie cardiaque et dans l'unité de soins coronariens.

Algorithmes de réseaux neuronaux artificiels (ANN)

Dans ce travail Kim et al. [19] Ont comparé l'utilisation de DT, des réseaux neuronaux artificiels (ANN) et SVM avec APACHE III pour prédire la mortalité, et ont rapporté que DT a donné la meilleure performance. En outre, Dybowski et al. [12] Ont adopté un système basé sur les réseaux neuronaux artificiels (ANN) pour prédire la mortalité et ont comparé ses performances à celles des modèles LR et SVM. Ils ont rapporté que l'ANN était plus précise que le SVM et le LR.

Algorithmes d'ensemble

Dans ce travail Johnson et al. [20] Ont fourni un ensemble pour la prédiction de la survie en utilisant un schéma d'ensemble bayésien qui se compose de 500 apprenants faibles (DT) ; il a atteint une aire sous la courbe ROC (AUC) de 86 %. De plus, Awed et al. [11] Ont conçu un modèle d'ensemble pour la prédiction de la mortalité, qui comprend RF, DT et Naive Bayes (NB). Ils ont appliqué leur modèle d'ensemble sur 20 caractéristiques extraites au cours des 6 premières heures de l'admission du patient. Il atteint un score AUROC de 82 %. En outre Xia et al. [21] Ont proposé un modèle d'ensemble basé sur la technique de mémoire à long terme et courte terme (LSTM) pour la prédiction de la mortalité. L'idée derrière ce travail est d'utiliser deux couches LSTM basées sur 50 caractéristiques extraites au cours des premières 24 heures. Ils ont obtenu un score AUC de 85,5 %.

Algorithmes d'apprentissage profond

Dans ce travail Caicedo-Torres et al. [22] Ont proposé un modèle d'apprentissage profond appelé ConvNet pour la prédiction de la mortalité. Leur système a atteint 87,3 % en termes d'AUC, et ils ont ajouté d'autres étapes en utilisant ConvNet pour gérer à la fois les données statiques et dynamiques. De plus, Ding et al. [23] Ont tenté de surmonter cette lacune en développant un cadre en deux étapes - une pour le regroupement et l'autre pour la prédiction de la mortalité. Leur méthode utilise l'algorithme d'apprentissage juste à temps (JITL) pour collecter des échantillons pertinents, après quoi le processus de prédiction est mené localement. En outre, Miao et al. [24] OÙ les données de laboratoire ont été utilisées pour prédire le risque de mortalité chez les patients souffrant d'insuffisance cardiaque. Cependant, un inconvénient de l'utilisation des mesures de laboratoire est que ces tests peuvent ne pas être disponibles pour tous les patients en USI, et tous les cas critiques ne peuvent pas être prédits à l'aide des résultats des tests de laboratoire. Les auteurs ont construit un modèle d'apprentissage profond utilisant uniquement les notes infirmières pour prédire la mortalité.

2.5 Conclusion

Les prédictions des taux de mortalité dans les unités de soins intensifs sont essentielles pour améliorer la gestion des patients et allouer les ressources de manière efficace. Les techniques d'apprentissage automatique augmentent la précision des prédictions, contribuant ainsi à améliorer la qualité des soins de santé. Les progrès continus dans ce domaine offrent de grandes opportunités pour fournir des soins de santé plus efficaces et fiables dans les unités de soins intensifs.

Chapitre 03 : Méthodologie

3.1 Introduction

Dans ce chapitre, nous commencerons par décrire la structure des données disponibles dans MIMIC-III et discuterons les méthodes utilisées pour nettoyer et préparer les données, ainsi que de l'utilisation des tests statistiques pour sélectionner les caractéristiques clés influençant la prédiction des décès. Nous présenterons différents modèles utilisés dans ce contexte, tels que le modèle logistique, les réseaux neuronaux artificiels, et les arbres aléatoires, ainsi que la manière dont ils sont entraînés et évalués.

La figure 3.1 présente les principales étapes utilisées pour développer notre système de prédiction. Avant de commencer à présenter les étapes en détail, il est important de comprendre le processus complet que nous suivons lors de l'entraînement d'un modèle d'apprentissage automatique. La formation d'un modèle comprend plusieurs phases essentielles, allant de la préparation des données jusqu'au déploiement du modèle. Voici les étapes en détail :

- Prétraitement des données : il s'agit de nettoyer et de préparer les données pour la formation et l'évaluation. Cela peut impliquer de supprimer le bruit, de traiter les valeurs manquantes et de transformer les données dans un format adapté au modèle.
- Sélection du modèle : il s'agit de choisir le bon modèle pour la tâche à accomplir. Il existe plusieurs modèles différents, chacun avec ses propres forces et faiblesses. Le meilleur modèle pour une tâche particulière dépendra de la nature des données et du résultat souhaité.
- Entraînement du modèle : il s'agit d'alimenter le modèle en données et de lui permettre d'apprendre les relations entre les caractéristiques et la variable cible. La quantité de données d'apprentissage requises dépendra de la complexité du modèle et de la taille de la variable cible.
- Évaluation du modèle : cela implique de tester le modèle sur un ensemble de données retenu qui n'a pas été utilisé pour la formation. Cela permet d'évaluer les performances du modèle sans sur-ajustement aux données d'apprentissage.
- Déploiement du modèle : cela implique de rendre le modèle disponible pour utilisation. Cela peut impliquer le déploiement du modèle dans un environnement de production ou sa mise à disposition en tant que service.

- Analyse des caractéristiques (facteurs de risque) : Identifier et analyser les facteurs de risque significatifs associés à la mortalité en USI pour fournir des recommandations cliniques.

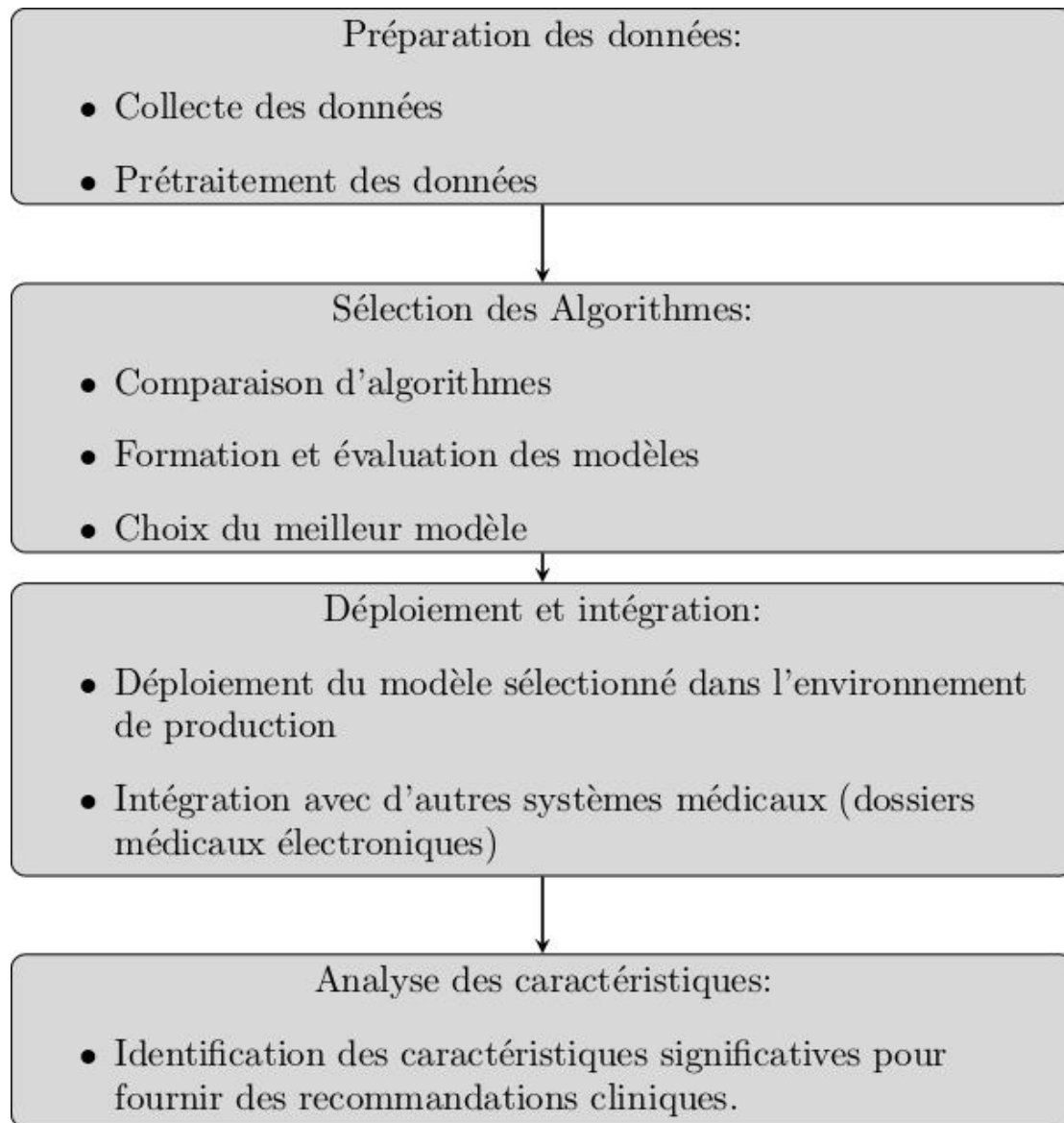


Figure 3.1 Un diagramme des étapes utilisées pour développer notre système de prédiction

3.2 Collecte et prétraitement des Données

3.2.1 Description des ensembles de données utilisés

Dans notre travail, nous utilisons la base de données MIMIC-III (Medical Information Mart for Intensive Care III) pour prédire la mortalité en USI. MIMIC-III est une base de données de soins critiques accessible au public, mise à jour par le Laboratoire de physiologie computationnelle du Massachusetts Institute of

Chapitre 03: Méthodologie

Technology (MIT). Cette base de données intègre des données cliniques détaillées et non identités de patients admis dans une unité de soins intensifs du Beth Israël Deaconess Medical Center (BIDMC) à Boston, Massachusetts, entre 2001 et 2012 [25].

Zhou, Jingmin et al. Ont extrait une version compatible avec l'apprentissage automatique et l'ont publiée sur le site Kaggle sous le nom "In-Hospital Mortality Prediction" au lien suivant : <https://www.kaggle.com/datasets/saurabhshahane/in-hospital-mortality-prediction>. Nous avons utilisé cet ensemble de données dans notre étude [26].

Les données se composent de 51 caractéristiques variées, incluant des informations démographiques. Ces caractéristiques fournissent un ensemble complet d'informations nécessaires pour développer et améliorer les modèles prédictifs liés aux résultats des patients dans les hôpitaux. Les caractéristiques mesurées dans le tableau suivant :

Group	Groupe d'étude.
ID	Identifiant du patient.
outcome	Résultat (0 = le patient n'est pas décédé, 1 = le patient est décédé).
age	Âge du patient.
gendera	Sexe du patient (1 = Homme, 2 = Femme).
BMI	Indice de masse corporelle (Body Mass Index).
hypertensive	Si le patient est hypertendu (1 = Oui, 0 = Non).
atrialfibrillation	Si le patient souffre de fibrillation auriculaire (1 = Oui, 0 = Non).
CHD with no MI	Maladie coronarienne sans infarctus du myocarde (1 = Oui, 0 = Non).
diabetes	Si le patient est diabétique (1 = Oui, 0 = Non).
deficiencyanemias	Anémies par carence (1 = Oui, 0 = Non).

Chapitre 03: Méthodologie

depression	Si le patient est dépressif (1 = Oui, 0 = Non).
Hyperlipemia	Hyperlipidémie (1 = Oui, 0 = Non).
Renal failure	Insuffisance rénale (1 = Oui, 0 = Non).
COPD	Bronchopneumopathie chronique obstructive (1 = Oui, 0 = Non).
heart rate	Fréquence cardiaque.
Systolic blood pressure	Pression artérielle systolique.
Diastolic blood pressure	Pression artérielle diastolique.
Respiratory rate	Fréquence respiratoire.
temperature	Température corporelle.
SP O2	Saturation en oxygène du sang.
Urine output	Volume urinaire.
hematocrit	Hématocrite.
RBC	Nombre de globules rouges.
MCH	Teneur moyenne en hémoglobine.
MCHC	Concentration corpusculaire moyenne en hémoglobine.
MCV	Volume globulaire moyen.
RDW	Écart-type de distribution des globules rouges.
Leucocyte	Nombre de leucocytes.
Platelets	Nombre de plaquettes.
Neutrophils	Pourcentage de neutrophiles.
Basophils	Pourcentage de basophiles.
Lymphocyte	Pourcentage de lymphocytes.
PT	Temps de prothrombine.

Chapitre 03: Méthodologie

INR	Ratio international normalisé (International Normalized Ratio).
NT-proBNP:	Protéine B-Natriurétique.
Creatine kinase	Enzyme créatine kinase.
Creatinine	Niveau de créatinine dans le sang.
Urea nitrogen	Niveau d'azote uréique dans le sang.
glucose	Niveau de glucose dans le sang.
Blood potassium	Niveau de potassium dans le sang.
Blood sodium	Niveau de sodium dans le sang.
Blood calcium	Niveau de calcium dans le sang.
Chloride	Niveau de chlorure dans le sang.
Anion gap	Écart anionique.
Magnesium ion	Niveau de magnésium dans le sang.
PH	pH du sang.
Bicarbonate	Niveau de bicarbonate dans le sang.
Lactic acid	Niveau d'acide lactique dans le sang.
PCO2	Pression partielle de dioxyde de carbone dans le sang.
EF	Fraction d'éjection cardiaque.

Tableau 3.1 Les caractéristiques spécifiques de base de données.

Ces caractéristiques fournissent des informations complètes sur l'état de santé des patients, les signes vitaux et les résultats de laboratoire, ce qui aide à analyser les données et à prédire les résultats des patients à l'aide de modèles analytiques. Ces variables permettent aux chercheurs de développer et de tester des modèles prédictifs pour évaluer la probabilité de décès des patients en unité de soins intensifs. Ces modèles peuvent être utilisés pour améliorer la prise de décision clinique et la gestion des ressources en soins intensifs.

3.2.2 Les étapes de Prétraitement

Nous avons effectué le prétraitement des données et comprend les étapes suivantes:

1. Les colonnes de groupe et d'identifiant ne sont pas nécessaires, donc elles doivent être supprimées.
2. Nous constatons qu'il y a 1 ligne dont le résultat est NA, nous allons supprimer cette ligne. Les autres colonnes qui incluent des valeurs NA représentent une quantité significative, donc nous ne pouvons pas supprimer ces lignes. D'autre part, ces colonnes sont très importantes pour la décision, donc nous ne pouvons pas non plus exclure l'ensemble des colonnes du processus d'entraînement.
3. Nous pouvons observer à partir des résultats ci-dessus que pour les lignes ayant une fréquence cardiaque NA, les autres signes vitaux sont également NA. Comme le nombre de lignes est faible, il est préférable de supprimer ces lignes.
4. Pour les valeurs manquantes restantes, remplaçons-les par une métrique statistique, les moyennes de colonnes spécifiquement.
5. La sélection des 10 meilleures caractéristiques pour l'entraînement est nécessaire en raison du grand nombre de caractéristiques présentes dans ces données. Nous avons utilisé trois méthodes différentes car chacune d'elles donne des résultats différents. Ensuite, nous prenons la moyenne de ces résultats pour choisir les caractéristiques. Ces méthodes sont les suivantes :

Le test du Chi-carré (Chi-square) : est une méthode supervisée de sélection de features, utilisée pour tester l'indépendance de deux variables statistiques (en trouvant des valeurs de test basées sur la relation entre le terme t et la variable cible c), par calcul de la corrélation du terme t avec la classe C . Le score du Ch2 utilise le test d'indépendance pour évaluer si le feature est indépendant de de la classe [27].

L'Information Mutuelle (Mutual Information) : est une mesure statistique de la quantité d'informations qu'une variable aléatoire possède sur une autre variable. IM peut être utilisée pour identifier les features pertinents pour une catégorie particulière [28].

La valeur F d'ANOVA (ANOVA F-value) : est utile pour comprendre l'influence des paramètres d'entrée donnés à partir d'une série de résultats expérimentaux, elle permet également de fournir une interprétation des données de sortie. La signification statistique des modèles quadratiques de prédiction est évaluée par P-value et F-value d'ANOVA [29].

3.3 Modèles d'apprentissage automatique

3.3.1 Modèles d'apprentissage automatique traditionnel

Il existe plusieurs modèles d'apprentissage automatique traditionnel. Dans cette section, nous présenterons de manière concise les modèles de classification les plus couramment utilisés et populaires.

3.3.1.1 Les réseaux de neurones (Artificial Neural Networks)

Les réseaux de neurones artificiels (ANN) sont des architectures informatiques modélisées d'après le cerveau. Ils sont construits à partir d'une série de "neurones" organisés en couches. Ces neurones présentent un comportement global déterminé par les connexions établies entre les différents éléments de traitement et les paramètres associés au sein de l'architecture du réseau neuronal. Chaque connexion qui relie les neurones dans des couches consécutives est pondérée. Le poids représente la force de la connexion entre le jème neurone d'une couche et le neurone de la couche suivante du réseau. La structure d'un réseau neuronal se compose d'une couche "d'entrée", d'une ou plusieurs couches "cachées" et d'une couche "de sortie". Le nombre de couches cachées et le nombre de neurones dans chacune de ces couches dépendent de la complexité du système considéré. La figure 3.1 montre une architecture ANN typique avec deux couches cachées [30].

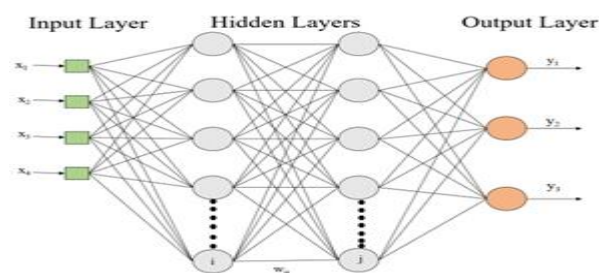


Figure 3.2 Structure générale du réseau de neurones artificiels avec deux couches cachées

Le perceptron multi-couches (MLP)

Un modèle largement utilisé appelé perceptron multi-couches (MLP) est le type le plus courant d'ANN. Le MLP se compose d'une couche d'entrée, d'une ou plusieurs couches cachées et d'une couche de sortie. Chaque couche emploie plusieurs neurones, et chaque neurone d'une couche est connecté aux neurones de la couche adjacente avec différents poids, ce qui permet de déterminer dans quelle mesure une unité influencera l'autre. La tâche d'un MLP est d'obtenir la sortie souhaitée en fonction des entrées de certains types. Dans cette optique, l'apprentissage du perceptron est réalisé. Les entrées et les sorties désirées sont chargées dans le schéma, et l'erreur de la réponse est déterminée. Les paramètres du système, les poids dans les interconnexions, sont modifiés pendant la phase d'apprentissage de l'ANN afin de réduire la différence entre la sortie désirée et la sortie réelle. Alors que le MLP se concentre davantage sur la résolution de problèmes non linéairement séparables, la classification et les fonctions continues approximatives [31].

3.3.1.2 Arbre de décision

Un arbre de décision est une structure semblable à un organigramme qui utilise une série de questions oui/non pour parvenir à une décision. Les arbres de décision sont un choix populaire pour la prise de décision car ils sont faciles à comprendre et à utiliser. Ils peuvent être utilisés pour prendre des décisions sur un large éventail de sujets, de ce qu'il faut porter aujourd'hui à quel investissement faire, dans un arbre de décision, commencez par le nœud racine et posez la première question. En fonction de la réponse à la question, vous suivrez la branche appropriée jusqu'au nœud suivant. Ce processus se poursuit jusqu'à ce que vous atteigniez un nœud feuille, qui représente la décision finale [32].

3.3.1.3 Machine à vecteur de support (Support Vector Machines)

Les SVMs sont un type d'algorithme d'apprentissage supervisé qui peut être utilisé pour la classification, la régression et la détection des valeurs aberrantes. Ils fonctionnent en trouvant l'hyperplan qui sépare le mieux les points de données en deux classes ou plus, les SVMs trouvent d'abord l'hyperplan à marge maximale, qui est l'hyperplan le plus éloigné de l'un des points de données. Cet hyperplan n'est pas toujours unique, donc les SVMs introduisent également une variable d'écart pour

permettre à certains points de données d'être du mauvais côté de l'hyperplan. L'objectif est de minimiser la variable d'écart tout en maximisant la marge [33].

3.3.1.4. Forêt aléatoire (Random Forest)

La forêt aléatoire est une méthode d'apprentissage d'ensemble pour la classification, la régression et d'autres tâches qui combine plusieurs arbres de décision. C'est une méthode robuste et précise qui est souvent utilisée dans les compétitions d'apprentissage automatique. L'algorithme de forêt aléatoire fonctionne en construisant un certain nombre d'arbres de décision sur différents sous-ensembles de données d'entraînement. Chaque arbre de décision est construit à l'aide d'un sous-ensemble aléatoire des caractéristiques. Cela aide à réduire la variance des arbres de décision individuels et rend la forêt aléatoire plus robuste au surajustement. La prédiction finale de la forêt aléatoire est faite en faisant la moyenne des prédictions des arbres de décision individuels. Cela aide à réduire le biais des arbres de décision individuels et rend la forêt aléatoire plus précise [34].

3.3.1.5 K-plus proches voisins (KNN)

L'algorithme K-Nearest Neighbors (KNN) est un algorithme d'apprentissage automatique non paramétrique qui peut être utilisé à la fois pour les tâches de classification et de régression. Cela fonctionne en trouvant les k instances les plus similaires (voisins) à une nouvelle instance, puis en prédisant l'étiquette de la nouvelle instance en fonction des étiquettes de ses voisins [35].

L'algorithme KNN fonctionne selon les étapes suivantes :

- Choisir la valeur de k.
- Calculez la distance entre la nouvelle instance et toutes les instances de l'ensemble d'apprentissage.
- Trier les instances en fonction de leur distance à la nouvelle instance.
- Sélectionnez les k instances les plus proches de la nouvelle instance.
- Prédire l'étiquette de la nouvelle instance en fonction des étiquettes des k plus proches voisins [35].

3.3.1.6 Régression logistique

La régression logistique est le plus utilisé des algorithmes de classification. Elle a un temps d'apprentissage rapide. Par conséquent, il est utilisé comme classificateur de base [25].

3.3.1.7 Extreme Gradient boosting (XGBoost)

XGBoost signifie "Extreme Gradient Boosting". est une méthode d'apprentissage supervisé qui utilise une stratégie d'ensemble basée sur le gradient boosting. Les data scientists utilisent fréquemment ce système évolutif de boosting d'arbres de bout en bout pour obtenir des résultats de pointe sur une variété de défis en apprentissage automatique. Il peut résoudre efficacement les problèmes de classification et de régression tout en produisant de meilleurs résultats. Les machines de Gradient Boosting ont été utilisées pour mettre en œuvre la première version de cet algorithme. Après avoir rendu ce travail open-source, une vaste communauté de data science a commencé à contribuer aux projets XGBoost, ce qui a aidé à améliorer davantage l'algorithme. XGBoost est devenu une bibliothèque logicielle et est maintenant directement installable dans nos systèmes grâce à une communauté aussi incroyable. XGBoost est une bibliothèque de gradient boosting distribuée qui a été optimisée pour être très efficace, adaptable et portable. Il utilise le cadre du Gradient Boosting pour implémenter des algorithmes d'apprentissage automatique. Il offre un boosting d'arbres parallèle pour résoudre rapidement et précisément une variété de problèmes de data science. La comparaison est effectuée en termes de précision et de vitesse d'entraînement [36].

3.3.2 Modèles d'apprentissage profond

Il existe plusieurs modèles d'apprentissage profond. Dans cette section, nous présenterons de manière concise les modèles de classification les plus couramment utilisés et populaires.

3.3.2.1 Les réseaux neuronaux profonds (DNN)

Les réseaux neuronaux profonds également appelés réseaux neuronaux profonds, sont des types spécifiques de réseaux de neurones qui sont composés de plusieurs couches de nœuds. Ces réseaux utilisent différentes architectures et techniques pour résoudre des problèmes complexes dans divers domaines, tels que la vision par ordinateur, le traitement du langage naturel, la reconnaissance de motifs, etc. Les réseaux neuronaux profonds ont montré des performances impressionnantes dans de nombreuses applications et ont contribué aux avancées récentes de l'apprentissage en profondeur [37].

3.3.2.2 Les réseaux de neurones récurrents (RNN)

Les réseaux de neurones récurrents (RNN) sont des modèles d'apprentissage profond spécialement conçus pour résoudre des problèmes liés aux séries temporelles. Les RNN excellent dans divers domaines tels que le traitement du signal, le traitement du langage naturel et la reconnaissance de la parole. Ils sont capables de prendre en compte les dépendances temporelles des données, en utilisant les sorties précédentes comme entrées pour les étapes suivantes. Cette capacité est due à leur mémoire interne qui leur permet de conserver des informations contextuelles. Cette prise en compte du contexte temporel permet aux RNN d'obtenir de meilleures performances dans des problèmes où la dimension temporelle est cruciale. Cependant, les RNN standard rencontrent des difficultés à apprendre des dépendances temporelles supérieures à quelques pas de temps discrets (environ 5 à 10) entre les événements d'entrée pertinents et les signaux cibles [38]. Le principal problème des réseaux de neurones récurrents (RNN) réside dans leur difficulté à capturer des dépendances à long terme dans les séquences. En raison de la nature récurrente de leur architecture, les RNN ont tendance à souffrir de ce que l'on appelle le "problème du gradient qui disparaît" ou le "problème du gradient explosif".

3.3.2.3 Le Long Short Term Memory (LSTM)

Les réseaux LSTM (Long Short-Term Memory) ont été développés pour remédier à la perte de mémoire à long terme des réseaux de neurones récurrents (RNN). Ils sont conçus pour modéliser des dépendances à longue distance dans une séquence en utilisant une mémoire interne appelée "cell state" dans chaque neurone. Dans un

LSTM, lorsque la sortie de l'unité neuronale précédente arrive à l'unité courante, elle passe par trois portes : porte d'oubli, la porte d'entrée et la porte de sortie. Ces portes utilisent des fonctions d'activation telles que des sigmoïdes et des tangentes hyperboliques pour gérer l'information [39].

3.3.2.4 Les réseaux de neurones convolutifs (CNN)

Les réseaux de neurones convolutifs (CNN) tirent leur nom de l'utilisation de l'opération mathématique de convolution. Ils sont une forme spécifique de réseaux neuronaux qui utilisent la convolution à la place de la multiplication matricielle générale dans au moins une de leurs couches. Les CNN sont considérés comme l'un des meilleurs algorithmes d'apprentissage pour effectuer la convolution, ce qui permet d'extraire des caractéristiques pertinentes à partir de données localement corrélées. Les noyaux convolutifs produisent une sortie qui est ensuite passée à une unité de traitement non linéaire, également appelée fonction d'activation. Cette fonction non linéaire facilite l'apprentissage des abstractions et intègre la non-linéarité dans l'espace des caractéristiques. Grâce à cette non-linéarité, différentes réponses génèrent différents modèles d'activations, ce qui facilite l'apprentissage des différences sémantiques dans les images [40].

3.4 Entraînement et validation du modèle

3.4.1 La méthode de validation croisée

La validation croisée est une technique précieuse en apprentissage automatique et en analyse statistique pour évaluer la performance de généralisation des modèles et sélectionner le meilleur modèle en fonction des métriques de performance. En validation croisée, les données sont divisées de manière répétée et plusieurs modèles sont entraînés. La version la plus couramment utilisée de la validation croisée est la validation croisée en k-plis, où k est un nombre spécifié par l'utilisateur, généralement 5 ou 10 [41].

Lors de la réalisation d'une validation croisée en 10 plis voici les étapes :

1. Diviser les données : L'ensemble de données est divisé aléatoirement en 10 parties de taille égale.

2. Boucle de validation : Une boucle est effectuée 10 fois, chaque fois en utilisant un pli différent comme ensemble de test et les 9 autres plis comme ensemble d'entraînement.
3. Entraînement du modèle : Le modèle est entraîné sur l'ensemble d'entraînement, c'est-à-dire les 9 plis.
4. Evaluation : Le modèle est évalué sur l'ensemble de test, c'est-à-dire le pli restant, et la performance du modèle est enregistrée.
5. Mesure de la performance : Une mesure de performance, telle que la précision de la classification est calculée pour évaluer la performance du modèle sur chaque pli de test.
6. Moyenne des performances : Une fois que la boucle d'évaluation est terminée, les performances obtenues sur les 10 plis de test sont généralement moyennées pour obtenir une estimation globale de la performance du modèle.

L'avantage d'utiliser la validation croisée en k-plis est qu'elle fournit une estimation plus robuste de la performance d'un modèle en utilisant plusieurs ensembles d'entraînement et de validation. Elle aide à réduire le biais et la variance dans l'estimation de la performance par rapport à une simple division entraînement-test. De plus, la validation croisée permet une évaluation plus approfondie de la capacité du modèle à se généraliser à des données non vues, car elle teste le modèle sur différents sous-ensembles des données [41].

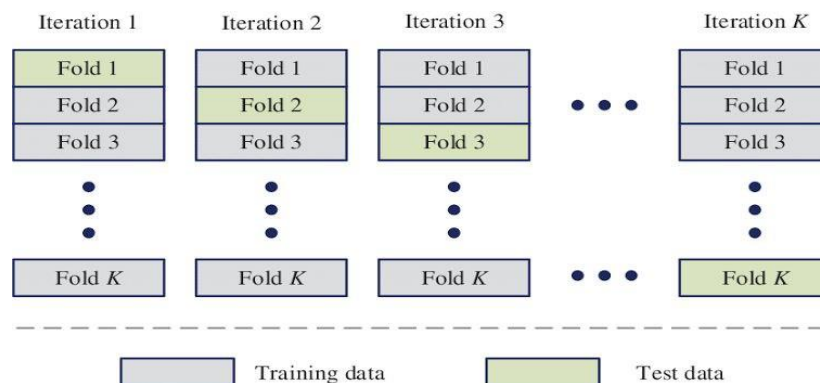


Figure 3.3 Exemple de validation croisée

3.5 Métriques d'évaluation

Précision : La mesure de précision de la classification est une mesure couramment utilisée pour évaluer la performance d'un modèle de classification. Elle est utilisée pour déterminer dans quelle mesure un modèle est capable de classer correctement les données d'un ensemble de test. Cette mesure est souvent appelée "taux de précision". La précision de la classification est calculée en comparant les prédictions faites par le modèle aux étiquettes réelles des données de test. Plus précisément, elle est calculée en divisant le nombre de prédictions correctes par le nombre total de prédictions. La formule pour calculer la précision de la classification est la suivante : $\text{Taux de précision} = (\text{Nombre de prédictions correctes} \times 100) / (\text{Nombre total de prédictions})$ La précision de la classification est exprimée en pourcentage et peut varier de 0 à 100 % [42].

3.6 Analyse de l'importance des Caractéristiques

La méthode de forêts aléatoires (Random Forest) est couramment utilisée pour identifier les caractéristiques les plus importantes dans un ensemble de données. Voici comment la méthode Random Forest peut être utilisée pour cette tâche [43]:

1. Construction d'un ensemble d'arbres : La méthode Random Forest construit un ensemble d'arbres de décision en utilisant des échantillons aléatoires du jeu de données d'entraînement et un sous-ensemble aléatoire de caractéristiques à chaque nœud de l'arbre. Cela permet de créer plusieurs arbres différents qui capturent différentes relations entre les caractéristiques et la variable cible.
2. Calcul de l'importance des caractéristiques : Une fois que l'ensemble d'arbres est construit, l'importance des caractéristiques peut être évaluée en se basant sur la réduction de l'impureté (généralement mesurée par le critère de gain d'information ou de gini) obtenue par chaque caractéristique lors de la construction des arbres. Plus précisément, l'importance d'une caractéristique est calculée en mesurant la diminution de l'impureté moyenne résultant de l'utilisation de cette caractéristique pour diviser les arbres de la forêt.
3. Classement des caractéristiques : Les importances des caractéristiques sont normalisées pour former un classement des caractéristiques, où celles avec des valeurs

plus élevées sont considérées comme plus importantes. Ce classement permet de déterminer les caractéristiques les plus discriminantes pour la prédiction de la variable cible.

Il convient de noter que la méthode Random Forest est robuste face au surapprentissage et fournit généralement des estimations fiables de l'importance des caractéristiques. De plus, elle est capable de prendre en compte les interactions entre les caractéristiques, ce qui peut améliorer la précision de l'identification des caractéristiques les plus importantes.

3.7 Conclusion

Cette étude met en lumière l'importance de la base de données MIMIC-III pour prédire les décès en unités de soins intensifs. En utilisant une analyse approfondie des données et en appliquant des modèles d'apprentissage automatique, cette recherche contribue à améliorer la prise de décision clinique et la gestion des ressources de manière efficace. Cela renforce la qualité des soins aux patients et améliore leurs résultats de santé.

Chapitre 04 :

Implémentation du Système

4.1 Introduction

Dans ce chapitre, traite des éléments fondamentaux d'un système donné, y compris l'architecture du système, les outils et technologies utilisés, et l'environnement de développement. Ces points sont illustrés par la présentation des principaux langages de programmation comme Python, des bibliothèques importantes telles que Pandas et Scikit-learn, ainsi que des détails sur le matériel et les logiciels utilisés pour le développement. Le passage donne également un aperçu des considérations de conception de l'interface utilisateur et des fonctionnalités disponibles.

4.2 Aperçu de l'architecture du système

Le système de la prédiction de la mortalité en USI se compose de deux modules et d'une interface principale.

Cette dernière est le point de départ de l'application, où vous pouvez trouver deux boutons, l'un pour accéder au premier module et l'autre pour accéder au second module. Ce qui suit est une description de la partie de l'implémentation de notre système.

4.2.1 L'interface principale

L'interface principale comprend les parties suivantes.

Bibliothèques utilisées

- tkinter : pour créer l'interface graphique (GUI).
- PIL (de la bibliothèque Pillow) : pour la manipulation des images.
- subprocess : pour exécuter des fichiers Python externes.

Ouverture des modules externes

- open_comp_module() : Une fonction qui lance le premier module qui sert à comparer les performances des algorithmes de classification avant et après la sélection des caractéristiques afin de choisir le meilleur classifieur. Cette fonction ouvre le module de comparaison (comparaison.py).
- open_deploy_module() : Une fonction qui lance le deuxième module qui sert à entraîner Random Forest afin de l'utiliser comme un classifieur pour le système de prédiction de la mortalité. Cette fonction ouvre le module (deployment.py)

Création de l'interface utilisateur :

1. Configuration de la fenêtre principale :

- Création d'une fenêtre intitulée "Système de la prédiction de la mortalité dans les unités de soins intensifs (USI)".
- Définition des dimensions de la fenêtre (620x600) et désactivation de la redimensionnement.

2. Ajout de l'arrière-plan :

- Chargement et redimensionnement de l'image depuis le chemin spécifié.
- Définition de l'image comme arrière-plan de la fenêtre.

Ajout des boutons

- Bouton pour lancer le module de comparaison, avec un style spécifique, un fond bleu et du texte en blanc.
- Bouton pour lancer le déploiement du système, avec le même style.

Fonctionnement de l'interface :

- Lors de l'exécution du programme, une fenêtre s'ouvre avec une image d'arrière-plan et deux boutons au centre de l'écran.
- En cliquant sur l'un des boutons, le module correspondant (comparaison.py ou deployment.py) est exécuté en utilisant subprocess.

4.2.2. Le module de comparaison

Le module de comparaison comprend les parties suivantes.

Bibliothèques utilisées

- .- tkinter : bibliothèque pour créer des interfaces graphiques.
- PIL (Pillow) : bibliothèque pour le traitement des images.
- pandas : pour l'analyse et la manipulation des données.
- scikit-learn : pour l'application des modèles de machine learning, la validation croisée et la sélection des caractéristiques.
- xgboost : pour appliquer le modèle XGBoost.

- tensorflow et scikeras : pour créer et entraîner différents réseaux de neurones.

La classe MortalityPredictionApp

Représente la classe principale pour l'application de prédiction de la mortalité. Elle contient plusieurs fonctions pour créer l'interface utilisateur, charger les données, entraîner les modèles et afficher les résultats.

La fonction init Initialise la fenêtre de l'application et définit les variables de base.

La fonction create_gui Crée les éléments de l'interface utilisateur tels que les boutons et la zone de texte.

Les fonctions pour créer des modèles de réseaux de neurones

- **create_dnn_model** : pour créer un modèle de réseau de neurones profond (DNN).

- **create_cnn_model** : pour créer un modèle de réseau de neurones convolutif (CNN).

- **create_rnn_model** : pour créer un modèle de réseau de neurones récurrent (RNN).

- **create_lstm_model** : pour créer un modèle LSTM.

La fonction select_features

Sélectionne les meilleures caractéristiques des données en utilisant trois méthodes différentes et combine les résultats pour obtenir les 10 meilleures caractéristiques.

La fonction upload_file Charge un fichier CSV et stocke les données dans une variable.

La fonction on_train_click Nettoie les données, sélectionne les caractéristiques, entraîne les modèles et affiche les résultats.

La fonction train_models Entraîne différents modèles et calcule la précision de chaque modèle.

La fonction calculate_accuracy Calcule la précision du modèle en utilisant la validation croisée.

La fonction display_results Exécute l'application et affiche l'interface graphique.

Les étapes principales

1. Création de l'interface graphique (GUI) :

- La fenêtre de l'application est créée à l'aide de la bibliothèque Tkinter.
- Le titre et les dimensions de la fenêtre sont définis.
- Une image de fond est chargée et affichée.
- Des boutons sont créés pour charger un fichier CSV et calculer le taux de prédiction de la mortalité.

2. Chargement et traitement des données :

- Les données sont chargées à partir d'un fichier CSV à l'aide de pandas.
- Les données sont nettoyées en supprimant les colonnes inutiles et en gérant les valeurs manquantes.
- Les meilleures caractéristiques sont sélectionnées en utilisant trois méthodes différentes : Chi2, F-classif et Mutual Information.

3. Entraînement des différents modèles :

- Une gamme variée de modèles est entraînée, y compris ANN, SVM, Random Forest, Logistique Régression, Décision Tree, KNN et XGBoost.
- Les modèles de réseaux de neurones différents tels que DNN, CNN, RNN et LSTM sont entraînés en utilisant TensorFlow.

4. Calcul de la précision des modèles :

- La validation croisée est utilisée pour calculer la précision de chaque modèle.
- Le meilleur modèle est choisi en fonction de la précision.

5. Affichage des résultats :

- Les résultats sont affichés dans l'interface graphique.
- La précision de chaque modèle et les meilleures caractéristiques sélectionnées sont affichées.

Interaction avec l'utilisateur

- L'utilisateur peut charger un fichier CSV contenant les données.

- L'utilisateur peut appuyer sur le bouton "Calculer le taux de prédiction de la mortalité" pour démarrer le processus de traitement des données et d'entraînement des modèles.

- Les résultats sont affichés dans une zone de texte dans l'interface graphique.

4.2.3 Le module de déploiement

Le module de déploiement comprend les parties suivantes.

Bibliothèques utilise

- pandas : pour la manipulation des données.

- sklearn : pour la sélection des caractéristiques et les modèles de machine learning.

- tkinter et PIL : pour créer l'interface utilisateur graphique (GUI).

- messagebox et filedialog : pour les boîtes de dialogue et la sélection de fichiers.

Fonctions de traitement et d'entraînement du modèle

Fonction `train_model`

- Entrées : X (caractéristiques) et y (étiquettes).

1. Divise les données en ensembles d'entraînement (90%) et de test (10%).
2. Entraîne un classifieur de forêt aléatoire sur les données d'entraînement.
3. Affiche une boîte de message informant l'utilisateur de la fin de l'entraînement.

Fonction `classify_model`

1. Récupère les valeurs des caractéristiques saisies par l'utilisateur.
2. Crée un nouveau point de données avec ces valeurs.
3. Utilise le modèle de forêt aléatoire entraîné pour prédire le résultat.
4. Affiche le résultat de la prédiction et les importances des caractéristiques dans une zone de texte.

Fonction `upload_file`

1. Ouvre une boîte de dialogue pour sélectionner un fichier CSV.
2. Charge le fichier CSV dans un DataFrame pandas.

3. Affiche une boîte de message informant l'utilisateur du chargement réussi du jeu de données.

Interface utilisateur graphique (GUI)

Création de la fenêtre principale

- Dimensions et titre : Définition des dimensions de la fenêtre et du titre.
- Image de fond : Chargement et redimensionnement d'une image de fond.

Boutons et zones d'entrée

- Bouton de téléchargement : Permet de charger un fichier CSV.
- Bouton d'entraînement : Entraîne le modèle avec les données chargées.
- Bouton de classification : Prédit le résultat pour un nouveau patient.
- Zones d'entrée : Création de boîtes d'entrée pour les caractéristiques du patient, telles que NT proBNP, Créatine kinase, etc.

Zone de texte pour les résultats

- Affichage des résultats : Zone de texte pour afficher le résultat de la classification et les importances des caractéristiques.

Fonctions supplémentaires

Fonction `train_button_click`

1. Nettoie les données en supprimant les colonnes inutiles et en gérant les valeurs manquantes.
2. Sélectionne les caractéristiques les plus importantes.
3. Entraîne le modèle en appelant la fonction `train_model`.

Fonction `select_features`

1. Sélectionne les meilleures caractéristiques en utilisant trois méthodes : Chi2, F-classif et Mutual Information.
2. Calcule la moyenne des scores pour chaque méthode.
3. Retourne les indices et les noms des meilleures caractéristiques.

Fonction `classify_button_click`

1. Sélectionne les caractéristiques les plus importantes.
2. Classifie un exemple en utilisant la fonction `classify_model`.

4.3 Diagramme de bloc illustrant les principaux composants

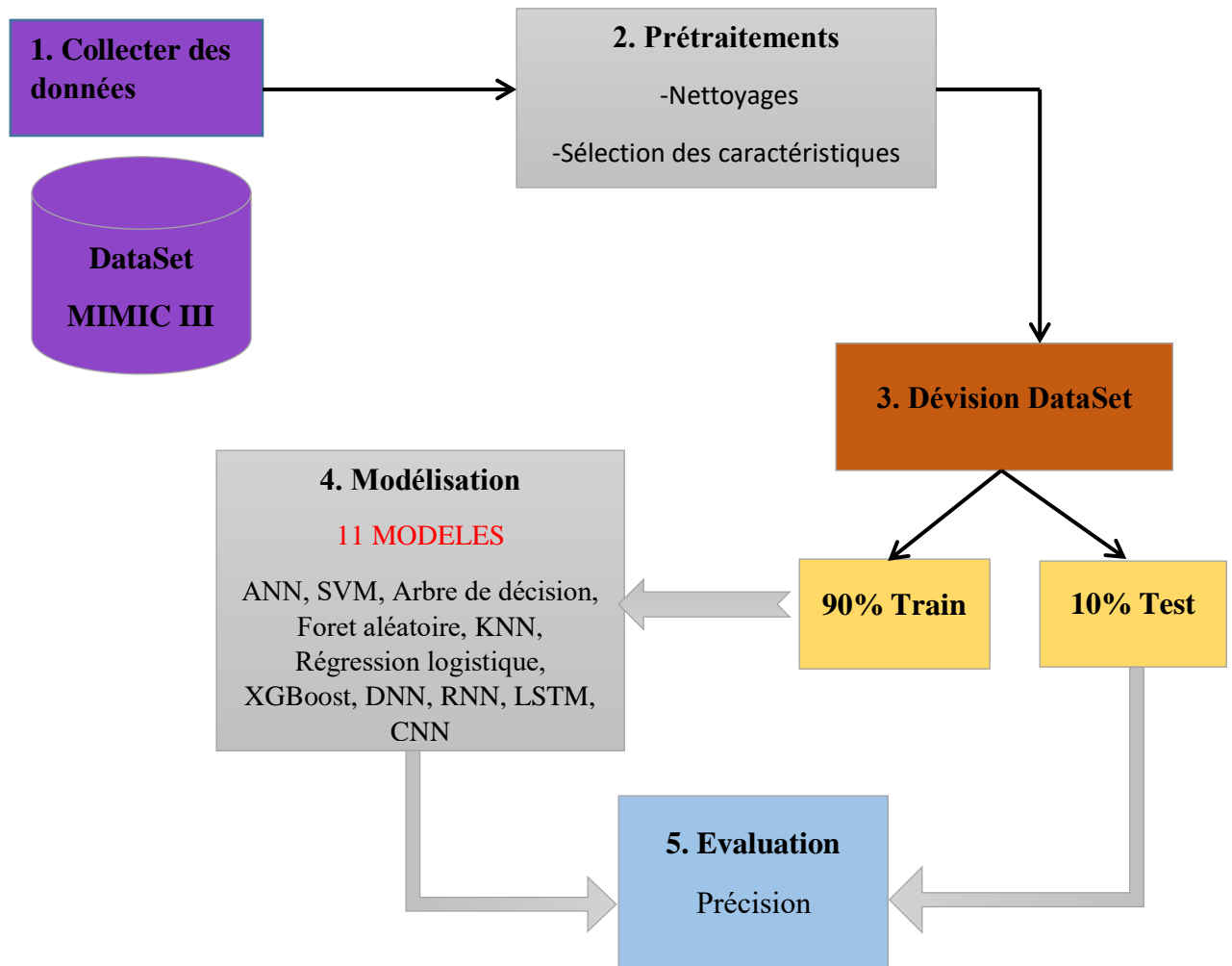


Figure 4.1 Diagramme de bloc illustrant les principaux composants

4.4 Technologies et Outils

4.4.1 Langages de Programmation

4.4.1.1 Python : est un langage de programmation de haut niveau interprété pour la programmation à usage général. Créé par Guido van Rossum, et publié pour la première fois en 1991. Python repose sur une philosophie de conception qui met l'accent sur la lisibilité du

code, notamment en utilisant des espaces significatifs. Il fournit des constructions permettant une programmation claire à petite et grande échelle [43].

Python propose un système de typage dynamique et une gestion automatique de la mémoire. Il prend en charge plusieurs paradigmes de programmation, notamment orienté objet, impératif, fonctionnel et procédural, et dispose d'une bibliothèque standard étendue et complète.

Python est un langage de programmation open-source et de haut niveau, développé pour une utilisation avec une large gamme de systèmes d'exploitation. Il est qualifié de langage de programmation le plus puissant en raison de sa nature dynamique et diversifiée. Python est facile à utiliser avec une syntaxe super simple très encourageante pour les apprenants débutants, et très motivante pour les utilisateurs chevronnés [43].

Il est le langage de programmation open source le plus couramment utilisé par les informaticiens. Ce langage se démarque dans les domaines de la gestion d'infrastructures, de l'analyse de données ou du développement de logiciels. En fait, dans ses qualités, Python permet aux développeurs de se concentrer sur ce qu'ils font, et non sur la façon dont ils le font. Il libère les développeurs des contraintes formelles des anciens langages qui tourmentaient leur époque. Par conséquent, développer du code en Python est plus rapide que dans d'autres langages [43].

Il est également disponible pour les débutants si nous passons du temps à nous lancer. De nombreux tutoriels sont également disponibles sur des sites dédiés. Les réponses aux questions peuvent toujours être trouvées sur les forums informatiques, car de nombreux professionnels l'utilisent [43].



4.4.1.2 Pourquoi Python est-il si populaire ?

Python est populaire pour un certain nombre de raisons. Voici un aperçu plus approfondi de ce qui le rend si polyvalent et si facile à utiliser pour les programmeurs.

Sa syntaxe simple imite le langage naturel, ce qui facilite la lecture et la compréhension. Il est donc plus rapide de construire des projets et de les améliorer.

Il est polyvalent. Python peut être utilisé pour de nombreuses tâches différentes, du développement web à l'apprentissage automatique.

Il est convivial pour les débutants, ce qui le rend populaire auprès des codeurs débutants. Il s'agit d'un logiciel libre, ce qui signifie que son utilisation et sa distribution sont gratuites, même à des fins commerciales.

Les archives de modules et de bibliothèques de Python - des ensembles de codes créés par des utilisateurs tiers pour étendre les capacités de Python - sont vastes et ne cessent de croître. Python dispose d'une communauté importante et active qui contribue à l'ensemble des modules et des bibliothèques de Python et agit comme une ressource utile pour les autres programmeurs. La vaste communauté de soutien signifie que si les programmeurs se heurtent à une pierre d'achoppement, il est relativement facile de trouver une solution ; Quelqu'un a forcément déjà rencontré le même problème [44].

4.4.2 Frameworks et bibliothèques

4.4.2.1 Frameworks utilisés

Keras

Keras est une bibliothèque de réseaux neuronaux de haut niveau, écrite en Python et capable de s'exécuter sur TensorFlow ou Theano. Il a été développé dans le but de permettre une expérimentation rapide. Être en mesure de passer de l'idée au résultat le plus rapidement possible est la clé pour faire de la recherche [45].

4.4.2.2 Bibliothèques utilisés

Pandas

Pandas est une autre bibliothèque python utilisée pour la manipulation et l'analyse de données. Pandas fonctionne avec des Dataframes. Il a de nombreux outils intégrés qui pourraient être utilisés pour une multitude de fonctions. Une fonction importante est le nettoyage des données. Selon les analyses d'IBM, environ 80 % du temps consacré à un projet d'apprentissage automatique est consacré au prétraitement des données. Beaucoup de jeux de données disponibles ont des champs vides et cela peut grandement affecter négativement un modèle [44].



Scikit-learn (sklearn)

Scikit-learn est une bibliothèque gratuite pour le calcul scientifique et l'apprentissage automatique en langage python. Il contient divers algorithmes de régression, de classification et de clustering [44].



Tkinter

Tkinter est le module Python spécifiques aux interfaces graphiques (GUI) fourni par défaut. Il dérive de Tk (ToolKit) une extension graphique du langage de script Tcl (Tool Command Language) [46].



PIL (Python Imaging Library)

La bibliothèque d'imagerie Python ajoute des capacités de traitement d'images à votre interpréteur Python. Cette bibliothèque offre un support étendu des formats de fichiers, une représentation interne efficace et des capacités de traitement d'images assez puissantes. La bibliothèque d'images principale est conçue pour un accès rapide aux données stockées dans quelques formats de pixels de base. Elle doit fournir une base solide pour un outil de traitement d'images général [47].



Tensorflow

TensorFlow est une bibliothèque logicielle open source pour le calcul numérique utilisant des graphes de flux de données [48].



4.4.3 Environnement de développement

4.4.3.1 Matériel utilisé

Notre travail s'est effectué sur Labtop ASUS ayant les caractéristiques suivantes :

- Processeur : Intel(R) Core(TM) i3-5005U CPU @ 2.00GHz 2.00 GHz
- ID de périphérique : 7F3F0432-B63F-44FB-98CF-B2DF661FBDDA
- ID de produit : 00325-80595-24825-AAOEM
- Mémoire RAM installée : 4,00 GO
- Type de Système : Windows10, 64 bits, Processeur x64

4.4.3.2 Logicielle de l'environnement de développement

Visual studio code

Visual Studio Code (VS Code) est un éditeur de code source multiplateforme gratuit développé par Microsoft. Le logiciel prend en charge la coloration syntaxique, la complétion de code (alias IntelliSense) et la refactorisation du code. Il dispose d'outils de ligne de commande intégrés et d'un système de contrôle de version Git. Les utilisateurs peuvent installer des extensions pour étendre les fonctionnalités de VS Code via la boutique

d'extensions intégrée. Visual Studio Code supporte de nombreux langages de programmation par défaut, notamment JavaScript, TypeScript, CSS et HTML. Il est un excellent outil de développement pour les langages de développement autres que Java et Python.

Après que Yuzhao et nous ayons organisé une réunion avec l'équipe BI4PEOPLE, nous avons su que certains membres ont développé leurs applications en Node.js.

Alors, nous avons commencé notre développement d'application en Node.js afin d'aligner sur leur développement. Pour programmer plus efficacement, nous avons choisi Visual Studio Code à implémenter la deuxième version de notre application [49].



4.5 Interface Utilisateur

4.5.1 Considérations de Conception pour l'Interface Utilisateur

1. Interface Intuitive et Ergonomique

L'application vise à être conviviale et intuitive, facilitant ainsi la navigation et l'utilisation par les utilisateurs, qu'ils soient experts ou non en science des données.

2. Organisation des Fonctionnalités

Les principales fonctionnalités (téléchargement de fichiers CSV, entraînement du modèle, saisie des données du patient, prédiction des résultats) sont clairement organisées à l'aide de boutons et de zones de texte dédiées.

- Boutons d'Action : Les boutons comme "Télécharger un fichier CSV", "Entraîner le système" et "Prédire" sont placés de manière stratégique pour une accessibilité maximale et une progression logique des actions de l'utilisateur.

3. Gestion des Erreurs et des Messages Utilisateur

L'application inclut des mécanismes pour gérer les erreurs potentielles, telles que le chargement de données incorrectes ou manquantes, en affichant des messages d'erreur clairs à l'utilisateur.

- Boîtes de Dialogue et Messages Contextuels : Des boîtes de dialogue et des messages contextuels sont utilisés pour informer l'utilisateur sur le succès ou l'échec des opérations telles que le chargement des données, l'entraînement du modèle, etc.

4. Affichage des Résultats et de l'Importance des Caractéristiques

- Zone de Texte pour les Résultats : Une zone de texte est dédiée pour afficher les résultats de prédiction, indiquant si le patient est décédé ou non.

- Affichage de l'Importance des Caractéristiques : Les caractéristiques importantes utilisées par le modèle sont présentées dans la même zone de texte après la prédiction, permettant à l'utilisateur de comprendre quelles variables ont influencé la prédiction.

5. Esthétique et Design Visuel

L'interface utilise une image de fond attrayante pour améliorer l'esthétique générale et rendre l'expérience utilisateur plus engageante.

- Image de Fond Personnalisée : Une image personnalisée est intégrée à l'interface pour améliorer l'attrait visuel, tout en ne compromettant pas la lisibilité des éléments interactifs.

4.5.2 Captures d'Écran de l'Interface Utilisateur

Le système de prédiction de la mortalité se compose de deux modules et une interface principale.

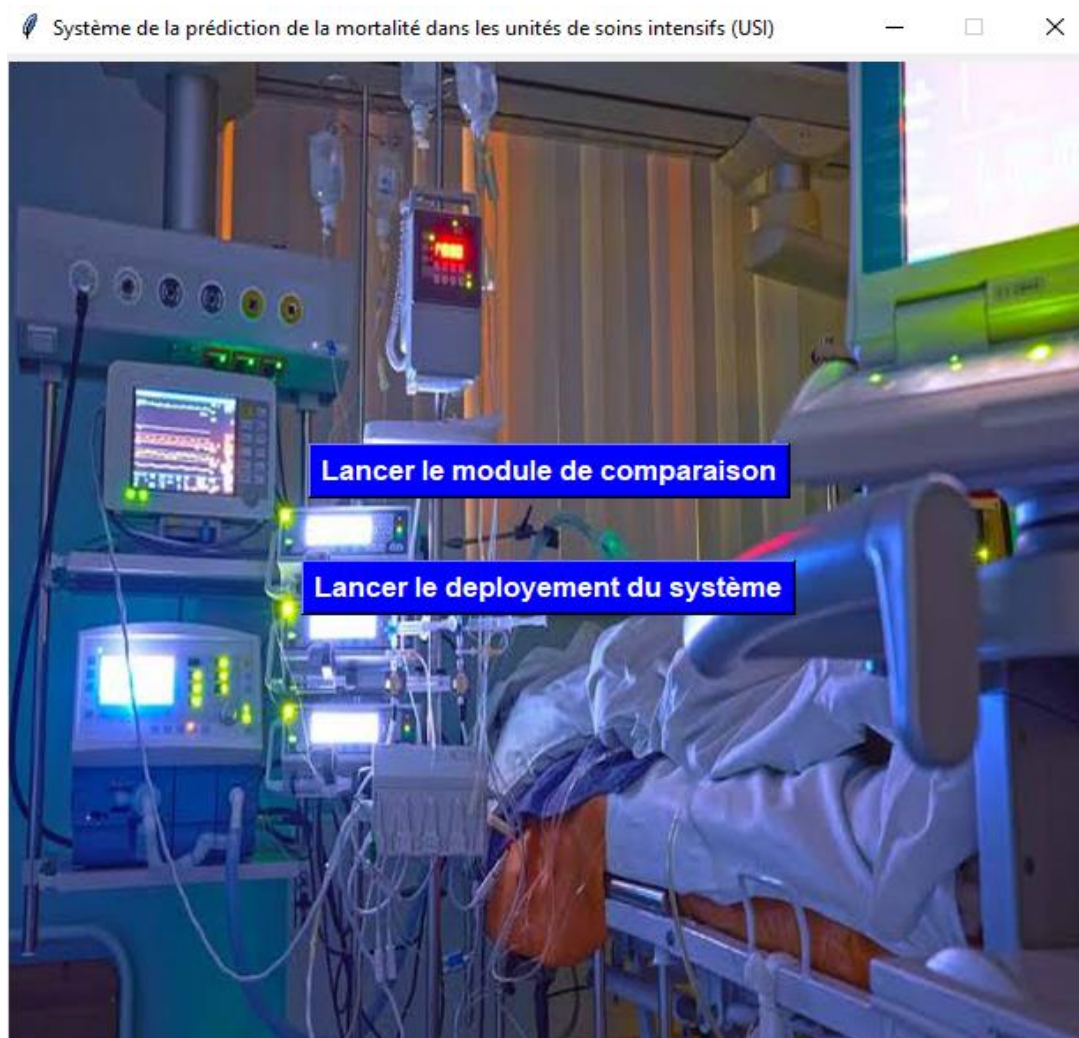


Figure 4.2 : L'interface principale pour le système de prédiction de la mortalité dans les USI

4.5.2.1 Le module de comparaison

Ce module vise à comparer la précision de classification de 11 algorithmes. La première étape est importer le fichier de jeu de données MIMIC III. Ensuite dans la deuxième étape, on lance le processus d'évaluation en cliquant sur le bouton « Calculer le Taux Précision de la Mortalité ». Le résultat d'évaluation s'affiche dans la fenêtre de résultats.

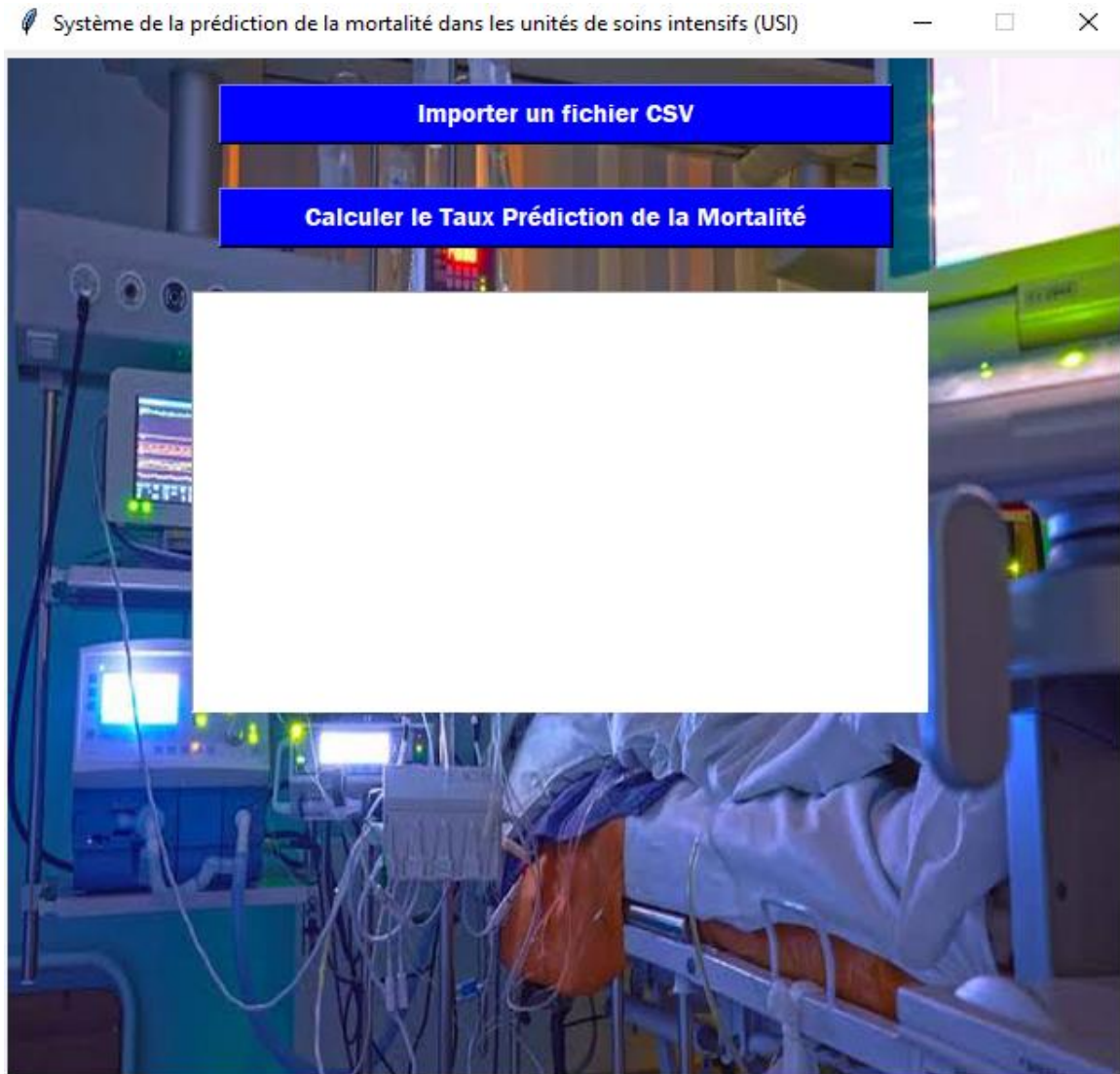


Figure 4.3 Interface du module de comparaison de prédiction de la mortalité dans USI

4.5.2.2 Le module de déploiement

Le meilleur algorithme est sélectionné (Random Forest dans notre cas). Cet algorithme est utilisé pour les processus de diagnostic. Pour utiliser ce module, on importe le fichier csv qui contient le jeu de données en cliquant sur le bouton « Télécharger un fichier CSV » et ensuite, sur le bouton « Entraîner le système » pour construire le système de classification en utilisant l'algorithme « Random Forest ». Après l'entraînement, le système est prêt pour diagnostiquer de nouveaux patients. L'utilisateur peut taper les valeurs des caractéristiques et ensuite clique sur le bouton « prédire ».

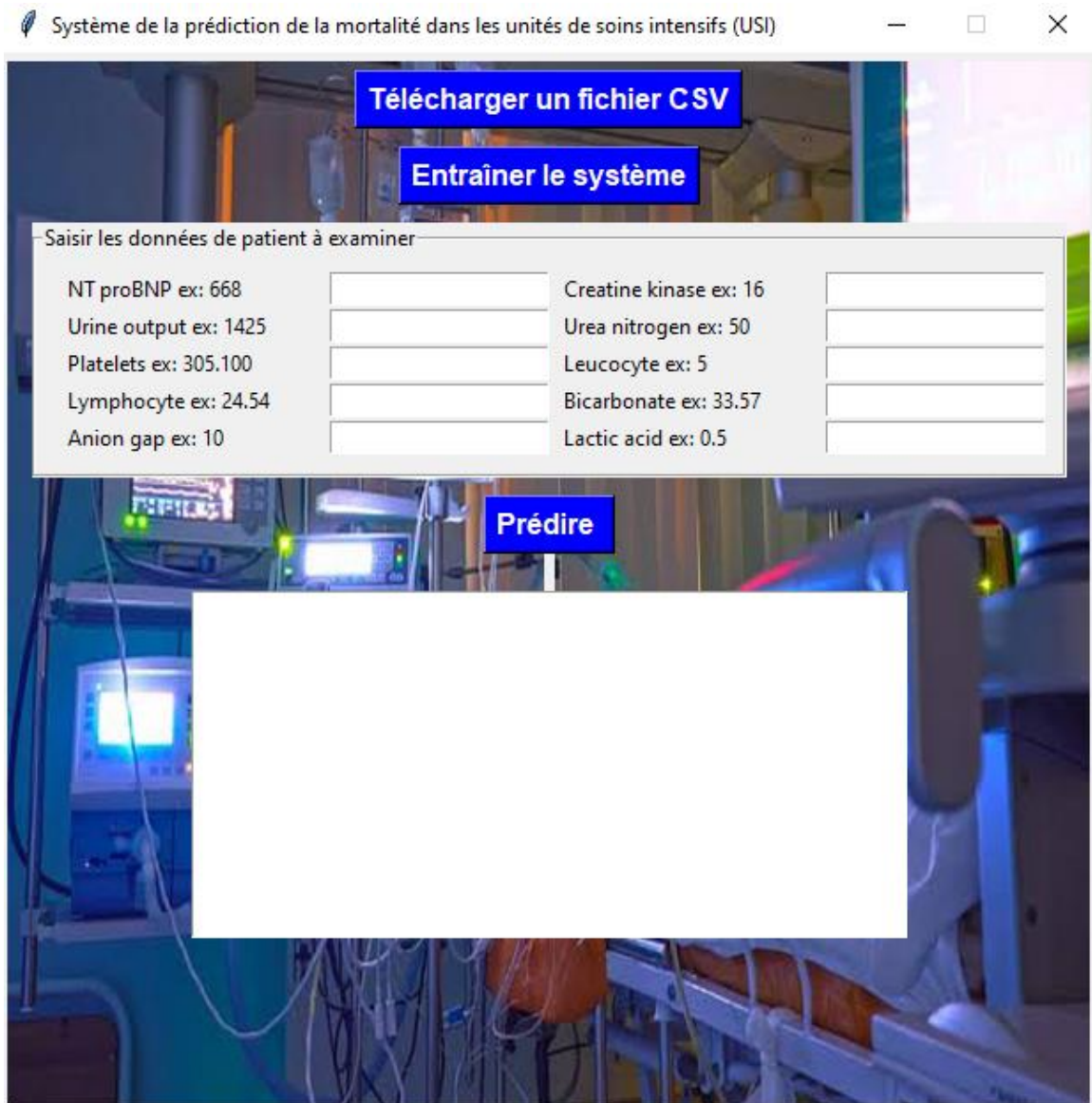


Figure 4.4 : Interface du module de déploiement

4.5.3 Description des Fonctionnalités Fournies par l'Interface Utilisateur

L'application de prédiction de la mortalité dans les unités de soins intensifs (USI) offre plusieurs fonctionnalités interactives via son interface utilisateur, permettant aux utilisateurs de télécharger des fichiers de données, d'entraîner des modèles de classification, et de prédire les résultats. Voici une description détaillée des principales fonctionnalités fournies :

Téléchargement de Fichiers CSV

L'interface utilisateur permet aux utilisateurs de télécharger des fichiers CSV contenant les données des patients à travers un bouton dédié :

- Bouton "Télécharger un fichier CSV" : Ce bouton ouvre une boîte de dialogue pour sélectionner et télécharger un fichier CSV. Le fichier sélectionné est ensuite chargé dans l'application pour un traitement ultérieur.

Entraînement du Modèle

L'application offre la possibilité d'entraîner un modèle de prédiction basé sur les données téléchargées :

- Bouton "Entraîner le système" : Après avoir téléchargé les données, l'utilisateur peut cliquer sur ce bouton pour lancer le processus d'entraînement du modèle. L'application nettoie les données, sélectionne les caractéristiques les plus importantes et entraîne un modèle de classification de forêt aléatoire (Random Forest).

Saisie des Données du Patient

Pour prédire la mortalité, l'utilisateur doit saisir les données spécifiques du patient :

- Cadre d'Entrée des Données : Un cadre est fourni pour l'entrée des caractéristiques du patient, telles que NT proBNP, Créatine kinase, Débit urinaire, etc. Chaque caractéristique a une étiquette descriptive et un champ de saisie où l'utilisateur peut entrer les valeurs correspondantes.

Prédiction et Visualisation des Résultats

Une fois les données du patient saisies, l'application peut prédire la mortalité et afficher les résultats :

- Bouton "Prédire" : En cliquant sur ce bouton, l'application utilise le modèle entraîné pour prédire si le patient est décédé ou non. Les résultats sont affichés dans une zone de texte dédiée.

- Zone de Texte des Résultats : Cette zone affiche le résultat de la prédiction ainsi que l'importance des caractéristiques utilisées par le modèle. Les caractéristiques sont triées par ordre d'importance pour donner une idée de leur influence sur la prédiction.

Affichage des Résultats

L'application fournit également un moyen d'afficher et de visualiser les résultats et l'importance des caractéristiques de manière claire et organisée :

- Zone de Texte pour l'Importance des Caractéristiques : Après la prédiction, l'importance des différentes caractéristiques est affichée, permettant à l'utilisateur de comprendre quelles caractéristiques ont le plus contribué à la prédiction.

4.6 Conclusion

Ce passage offre une compréhension complète des éléments essentiels utilisés dans le développement du système, en mettant l'accent sur les outils et technologies qui contribuent à l'efficacité et à la performance. En utilisant des langages de programmation puissants et des bibliothèques polyvalentes, les développeurs peuvent créer un système intégré répondant aux besoins des utilisateurs. De plus, l'environnement de développement avancé facilite le processus de programmation et améliore la qualité des résultats.

Chapitre 05 : Résultats et Discussion

5.1 Introduction

Dans ce chapitre, nous avons examiné les performances de plusieurs modèles d'apprentissage automatique sur des ensembles de données de test, en analysant l'impact de la sélection des caractéristiques sur la précision de ces modèles. Une comparaison détaillée des résultats a été présentée, en mettant en évidence les modèles qui ont montré une amélioration, ainsi qu'une analyse approfondie des facteurs influençant la performance de chaque modèle.

5.2 Présentation de la performance du modèle sur les ensembles de données de test

	La précision avant la sélection des caractéristiques %	La précision après la sélection des caractéristiques %	Gain
ANN	77.56	78.81	+1.25
SVM	86.43	86.43	aucune différence
Random forest	87.93	87.80	-0.13
Regression logistique	86.77	87.63	+0.86
Decision tree	80.53	81.54	+1.01
KNN	85.39	85.39	aucune différence
XGBoost	86.69	85.31	-1.38
Deep Neural Network	78.45	79.67	+1.22
CNN	86.60	85.80	-0.80
RNN	86.53	86.57	+0.04
LSTM	81.23	84.19	+2.96

Tableau 5.1 La performance du modèle sur les ensembles de données de test.

5.3 Analyse comparative avec les performances des modèles

Les résultats montrent que le modèle Random Forest a la précision la plus élevée, suivi de près par la régression logistique et le SVM. Les autres modèles, tels que le RNN, le CNN, le KNN et XGBoost, présentent également des précisions élevées.

L'arbre de décision, les réseaux de neurones profonds et artificiels montrent des performances appréciables, mais légèrement inférieures par rapport aux autres modèles

Les modèles ayant le plus grand gain de précision après la sélection des caractéristiques sont le LSTM (+2.96), l'Arbre de décision (+1.01), et le Deep Neural Network (+1.22).

Le SVM et le KNN montrent une stabilité totale, sans gain ni perte de précision.

Certains modèles ont une légère baisse de performance après la sélection des caractéristiques, notamment le Random Forest (-0.13), le XGBoost (-1.38), et le CNN (-0.80).

La sélection des caractéristiques semble avoir un impact positif sur la plupart des réseaux neuronaux et les arbres de décision.

La sélection des caractéristiques améliore généralement les performances des modèles de réseaux neuronaux et des arbres de décision, tout en ayant peu ou pas d'effet sur les modèles comme le SVM et le KNN, et un effet mixte sur les autres.

5.4 Discussion des résultats

Améliorations significatives de performance

1. Long Short-Term Memory (LSTM) a montré une amélioration significative, indiquant que la sélection des caractéristiques a aidé à éliminer les données non nécessaires et à améliorer la focalisation du modèle sur les motifs importants. En particulier, LSTM bénéficie des données propres et organisées en raison de sa structure adaptée au traitement des séquences temporelles.
2. Arbre de décision (Decision Tree) L'amélioration de la précision de l'arbre de décision suggère que la sélection des caractéristiques a permis d'identifier les caractéristiques les plus influentes, réduisant la complexité et augmentant l'efficacité du modèle pour segmenter les données en fonction des informations les plus importantes.
3. Réseau de neurones profonds (Deep Neural Network) L'amélioration de la précision du réseau de neurones profonds suggère que la sélection des caractéristiques

a aidé à réduire le bruit et à renforcer l'apprentissage du modèle. Ces réseaux bénéficient des ensembles de données plus concentrés pour obtenir de meilleures performances.

4. Réseau de neurones artificiels (ANN) L'amélioration de la précision de l'ANN indique que la sélection des caractéristiques a contribué à améliorer les performances du modèle, bien que dans une moindre mesure par rapport aux autres modèles.

Stabilité de la performance

1. Machine à vecteurs de support (SVM) et k plus proches voisins (KNN) La stabilité de la précision de SVM et KNN indique que ces modèles bénéficiaient déjà de l'ensemble initial de caractéristiques et que la sélection des caractéristiques n'a pas eu un impact significatif sur leurs performances. Cela reflète la capacité de ces modèles à gérer des ensembles de caractéristiques de haute dimension et leur efficacité avec l'ensemble de caractéristiques original.

Légère baisse de performance

1. XGBoost La baisse de précision de XGBoost suggère que la sélection des caractéristiques a peut-être supprimé des caractéristiques importantes contribuant aux performances du modèle, soulignant la nécessité d'une analyse approfondie des caractéristiques lors de l'utilisation de cette technique.

2. Réseau de neurones convolutif (CNN) La légère baisse de précision du CNN suggère que certaines caractéristiques supprimées étaient utiles pour reconnaître les motifs dans les données.

3. Forêt aléatoire (Random Forest) La légère baisse de précision de la forêt aléatoire indique que le modèle était capable de gérer le grand nombre de caractéristiques sans problème majeur, mais que la suppression de certaines caractéristiques a pu entraîner la perte d'informations importantes.

Résultats mixtes

1. Régression logistique et Réseau neuronal récurrent (RNN) L'amélioration légère de la précision de la régression logistique suggère que la sélection des caractéristiques a aidé à améliorer les performances du modèle en se concentrant sur les caractéristiques

les plus influentes. Quant à l'augmentation minimale de la précision du RNN, elle indique que la sélection des caractéristiques a eu un impact limité sur le modèle, reflétant la capacité du RNN à gérer efficacement des ensembles de données variés.

Les résultats montrent que le modèle Random Forest a constamment montré les meilleures performances. Cela souligne la robustesse de ce modèle dans le traitement de données médicales complexes et multidimensionnelles, capable de gérer efficacement les interactions non linéaires entre les variables.

Il est évident que la sélection des caractéristiques n'a pas eu un impact significatif sur la précision globale des modèles. Cela suggère que les caractéristiques éliminées n'avaient pas une influence notable sur la capacité des modèles à prédire avec précision. Dans le contexte de la prédiction des décès en unité de soins intensifs, cela signifie que les données de base utilisées étaient déjà riches en informations nécessaires.

5.5 Implication pour la pratique clinique

Nous utilisons la méthode de Random Forest pour entraîner le système de la prédiction de la mortalité sur le même ensemble de données. Une fois l'entraînement terminé, le système sera prêt à être utilisé pour diagnostiquer de nouveaux patients. Pour tester le fonctionnement du système, nous avons sélectionné de manière aléatoire un exemple du jeu de données.

Les valeurs des caractéristiques de cet exemple sont les suivantes :

- NT-proBNP: 668
- Creatine kinase: 16
- Urine output: 1425
- Urea nitrogen: 50
- Platelets: 305.100
- Leucocyte: 5
- Lymphocyte: 24.54

- Bicarbonate: 33.57
- Anion gap: 10
- Lactic acid: 0.5

Les caractéristiques mentionnées sont des indices vitaux qui peuvent aider à évaluer l'état des patients en soins intensifs et à prédire les résultats. Voici une explication simplifiée de chacune :

Bicarbonate : Le bicarbonate est un paramètre clé dans l'équilibre acido-basique de l'organisme. Un niveau bas de bicarbonate peut indiquer une acidose métabolique, souvent observée chez les patients gravement malades, et est associé à une mortalité plus élevée.

Anion gap : L'anion gap mesure la différence entre les cations et les anions dans le plasma. Un anion gap élevé peut indiquer une acidose métabolique, ce qui est souvent un marqueur de gravité de la maladie et peut être prédictif de résultats défavorables.

Lactic acid : L'acide lactique est un indicateur de l'oxygénation tissulaire. Des niveaux élevés d'acide lactique (lactatémie) peuvent signaler une hypoperfusion tissulaire et un choc septique, et sont souvent associés à un risque accru de mortalité en soins intensifs.

Urine output : La production d'urine est un indicateur important de la fonction rénale et de l'état d'hydratation. Une réduction de la diurèse peut indiquer une insuffisance rénale aiguë, un mauvais pronostic hydrique, et est un indicateur de mauvais pronostic.

Leucocyte : Le nombre de leucocytes (globules blancs) peut indiquer la présence d'une infection ou d'une inflammation. Une leucocytose (augmentation des leucocytes) ou une leucopénie (diminution des leucocytes) peuvent toutes deux être associées à une infection sévère ou à une réponse inflammatoire systémique, augmentant le risque de mortalité.

Lymphocyte : Les lymphocytes jouent un rôle crucial dans la réponse immunitaire. Un faible nombre de lymphocytes (lymphopénie) peut indiquer une

immunosuppression, fréquente chez les patients en soins intensifs et est associé à une issue défavorable.

Urea nitrogen : L'azote uréique sanguin (BUN) est un indicateur de la fonction rénale. Des niveaux élevés de BUN peuvent indiquer une insuffisance rénale, souvent corrélée avec une maladie critique et une mortalité accrue.

Platelets : Les plaquettes sont essentielles pour la coagulation sanguine. Une thrombocytopénie (faible nombre de plaquettes) peut indiquer un risque accru de saignement et est souvent associée à des troubles graves et une mortalité plus élevée.

NT-proBNP : Le NT-proBNP est un marqueur de l'insuffisance cardiaque. Des niveaux élevés de NT-proBNP peuvent indiquer une dysfonction cardiaque, fréquente chez les patients en soins intensifs, et sont associés à une augmentation de la mortalité.

Creatine kinase : La créatine kinase (CK) est un enzyme qui peut indiquer une lésion musculaire ou cardiaque. Des niveaux élevés de CK peuvent être observés dans des conditions telles que l'infarctus du myocarde ou la rhabdomyolyse, conditions graves souvent corrélées avec une mortalité accrue en soins intensifs.

Le système a correctement donné le résultat suivant : ce patient va probablement survire, suivi par un classement des caractéristiques selon leur importance ou leur contribution dans la classification du diagnostic. Voici les résultats obtenus (les caractéristiques sont énumérées avec leur importance respective):

Importance des caractéristiques :

- Bicarbonate: 0.12
- Anion gap: 0.12
- Lactic acid: 0.11
- Urine output: 0.10
- Leucocyte: 0.10
- Lymphocyte: 0.10

- Urea nitrogen: 0.09
- Platelets: 0.09
- NT-proBNP: 0.07
- Creatine kinase: 0.07

Ces valeurs représentent l'importance relative de chaque caractéristique dans la classification. Plus la valeur est élevée, plus la caractéristique est considérée comme importante pour la prédiction. Dans ce cas, les caractéristiques la plus importante pour la classification sont "Bicarbonate" et "Anion gap" avec une valeur d'importance de (0.12). Il est important de noter que ces valeurs d'importance sont spécifiques à ce modèle de classification par Random Forest et peuvent varier d'un modèle à l'autre.

Chapitre 06 : Conclusion et Perspectives

6.1 Conclusion

Cette étude visait à explorer l'utilisation des algorithmes d'apprentissage automatique pour la prédiction de la mortalité dans les unités de soins intensifs (USI). Le problème de recherche se concentrait sur l'amélioration de la précision des prédictions. Nous avons adopté une méthodologie comprenant la collecte de données de patients en USI, le prétraitement de ces données, l'application de divers modèles, et l'évaluation de leurs performances.

Les résultats ont montré que le meilleur modèle en termes de performance était celui des Forêts Aléatoires (Random Forest), qui a atteint une précision de 87,93 % avant la sélection des caractéristiques et de 87,80 % après la sélection. Ce modèle a clairement démontré une supériorité par rapport aux autres modèles testés, ce qui en fait le plus efficace pour prédire les décès dans les unités de soins intensifs.

Les caractéristiques les plus importantes qui ont contribué à améliorer la précision des prédictions comprenaient NT-proBNP, la créatine kinase, Urine output, Urea nitrogen, Platelets, leucocytes, lymphocytes, bicarbonate, Anion gap et lactic acid.

6.2 Limitations

Malgré les avancées réalisées, cette étude présente certaines limitations :

- 1. Échantillon de Données Limité :** Les ensembles de données utilisés proviennent d'un nombre restreint d'institutions, ce qui peut limiter la généralisation des résultats à d'autres contextes cliniques.
- 2. Complexité des Modèles :** Les modèles plus complexes nécessitent des ressources computationnelles significatives et peuvent être difficiles à interpréter pour les cliniciens.

6.3 Perspectives Futures

Comme perspectives, nous pouvons citer:

- Avec les avancées continues dans les techniques d'apprentissage automatique et profond, nous envisageons d'atteindre une précision prédictive plus élevée dans la prévision des décès en unité de soins intensifs. Nous compterons sur un ensemble de données cliniques et biologiques plus large pour améliorer nos modèles, ce qui aidera les médecins à prendre des décisions thérapeutiques plus efficaces et opportunes.

- À l'avenir proche, nous prévoyons de déployer notre modèle sur une application web et d'utiliser d'autres techniques pour prédiction.

Bibliographie

- [1] M. H. Khan, «A CNN-LSTM for predicting mortality in the ICU,» The University of Tennessee, Knoxville, 2019.
- [2] Lucille B Bruncker¹, Christina S Boncyk^{1,2}, Kimberly F Rengel^{1,2}, Christopher G Hughes^{1,2}, «A CNN-LSTM for predicting mortality in the ICU,» *Clinical Interventions in Aging*, 2023.
- [3] «Identifying the challenges of Intensive Care unit Caregiving in the Face of COVID-19,» *Iranian Red Crescent Medical Journal*, 2023.
- [4] L. Bruncker, Christina S Boncyk, Kimberly F. Rengel, et Christopher G. Hughes, «Elderly Patients and Management in Intensive Care Units (ICU),» *Clinical Interventions in Aging*, vol. 18, 2023.
- [5] Eniko Holndonner-Kirst, Gellért Karvaly, J. Gál et Csaba Hermann, «Challenge in modern intensive care: chronic critical illness - pathophysiology and therapeutic options,» *Orvosi Hetilap*, 2023.
- [6] z. anfal et boussireb abir, «Vision-Transformers pour la classification des images histopathologiques du cancer du sein,» Université 20 août 1955 - Skikda, skikda, 2023.
- [7] D. S. Bourekkouk Samira, «techniques d'apprentissage Automatique pour la segmentation d'Images Médicales(IRM),» Université 20 août 1955 - Skikda, skikda, 2023.
- [8] C.Houassine, «segmentation d'images par une approche biomimétique hybride,» Université m'hamed bougara, boumerdes, 2012.
- [9] A. K. e. G. Rekha, «Challenges of applying deep learning in real-world applications,» *Challenges and applications for implementing machine learning in computer vision.*, pp. 92-118, 2020.

- [10] Crawford, E. D.; Batuello, J. T.; Snow, P.; Gamito, E. J.; McLeod, D. G.; Partin, A. W.; Stone, N.; Montie, J.; Stock, R.; Lynch, J.; Brandt, J., The use of artificial intelligence technology to predict lymph node spread in men with clinically localized prostate carcinoma. *Cancer* 2000, 88, (9), 2105-9.
- [11] S. Kim, W. Kim, and R. Woong Park, "A comparison of intensive care unit mortality prediction models through the use of data mining techniques," *Healthc. Inform. Res.*, vol. 17, no. 4, pp. 232–243, 2011.
- [12] R. Sadeghi, T. Banerjee, and W. Romine, "Early hospital mortality prediction using vital signals," *Smart Health*, vols. 9–10, pp. 265–274, Dec. 2018.
- [13] S. Ghose, J. Mitra, S. Khanna, and J. Dowling, "An improved patient-specific mortality risk prediction in ICU in a random forest classification framework," *Stud. Health Technol. Inform.*, vol. 214, pp. 56–61, Aug. 2015.
- [14] Xia, S. Pan, M. Zhu, G. Cai, M. Yan, Q. Su, J. Yan, and G. Ning, "A long-short-term memory ensemble approach for improving the outcome prediction in intensive care unit," *Comput. Math. Methods Med.*, vol. 2019, pp. 1–10, Nov. 2019.
- [15] Calvert, J.; Mao, Q.; Hoffman, J. L.; Jay, M.; Desautels, T.; Mohamadlou, H.; Chettipally, U.; Das, R., Using electronic health record collected clinical variables to predict medical intensive care unit mortality. *Ann Med Surg (Lond)* 2016, 11, 52-7.
- [16] C.-W. Cheng, N. Chanani, J. Venugopalan, K. Maher, and M. D. Wang, "IcuARM—An ICU clinical decision support system using association rule mining," *IEEE J. Transl. Eng. Health Med.*, vol. 1, 2013, Art. no. 4400110
- [17] I. M. Ball, S. M. Bagshaw, K. E. A. Burns, D. J. Cook, A. G. Day, P. M. Dodek, D. J. Kutsogiannis, S. Mehta, J. G. Muscedere, H. T. Stelfox, A. F. Turgeon, G. A. Wells, and I. G. Stiell, "A clinical prediction tool for hospital mortality in critically ill elderly patients," *J. Crit. Care*, vol. 35, pp. 206–212, Oct. 2016.

- [18] R. S. Anand, P. Stey, S. Jain, D. R. Biron, H. Bhatt, K. Monteiro, E. Feller, M. L. Ranney, I. N. Sarkar, and E. S. Chen, “Predicting mortality in diabetic ICU patients using machine learning and severity indices,” *AMIA Summits Transl. Sci. Proc.*, vol. 2017, pp. 310–319, May 2018
- [19] S. Kim, W. Kim, and R. Woong Park, “A comparison of intensive care unit mortality prediction models through the use of data mining techniques,” *Healthc. Inform. Res.*, vol. 17, no. 4, pp. 232–243, 2011
- [20] A. E. W. Johnson, N. Dunkley, L. Mayaud, A. Tsanas, A. A. Kramer, and G. D. Clifford, “Patient specific predictions in the intensive care unit using a Bayesian ensemble,” *Comput. Cardiol.*, vol. 39, pp. 249–252, Jan. 2012.
- [21] Xia, S. Pan, M. Zhu, G. Cai, M. Yan, Q. Su, J. Yan, and G. Ning, “A long short-term memory ensemble approach for improving the outcome prediction in intensive care unit,” *Comput. Math. Methods Med.*, vol. 2019, pp. 1–10, Nov. 2019.
- [22] W. Caicedo-Torres and J. Gutierrez, “ISeeU: Visually interpretable deep learning for mortality prediction inside the ICU,” *J. Biomed. Informat.*, vol. 98, Oct. 2019, Art. no. 103269.
- [23] Y. Ding, Y. Wang, and D. Zhou, “Mortality prediction for ICU patients combining just-in-time learning and extreme learning machine,” *Neurocomputing*, vol. 281, pp. 12–19, Mar. 2018.
- [24] F. Miao, Y.-P. Cai, Y.-X. Zhang, X.-M. Fan, and Y. Li, “Predictive modeling of hospital mortality for patients with heart failure by using an improved random survival forest,” *IEEE Access*, vol. 6, pp. 7244–7253, 2018.
- [25] A. ABRAICH, «L'apprentissage profond sur MIMIC-III : Prédiction de la mortalité sous 24 h,» université evry val-d'essonne, paris, 2018.
- [26] Zhou, Jingmin et al. (2021). Prediction model of in-hospital mortality in intensive care unit patients with heart failure: machine learning-based, retrospective analysis of the MIMIC-III database [Dataset]. Dryad. <https://doi.org/10.5061/dryad.0p2ngf1zd>

- [27] S. Aya, «Un schéma de pondération et de sélection des termes pertinents basé sur la distribution des fréquences des documents et des termes entre catégories,» Université de 8 Mai 1945, Guelma, 2022.
- [28] S. B. e. al, «Feature selection using an improved Chi-square for Arabic text classification,» Journal of King Saud University-Computer and Information Sciences , pp. 225-231, 2020.
- [29] B. Chaded, «APPLICATION DE L'ANALYSE DE RÉGRESSION POUR LA PRÉDICTION DE L'EFFORT DE COUPE LORS TOURNAGE,» Université Larbi tébessi, Tébessa, 2021.
- [30] A. Moghaddamia, «“Evaporation estimation using artificial neural networks and adaptive neuro-fuzzy inference system techniques,» Advances in Water Resources, pp. . 88-97., 2009.
- [31] F. Ortellado et Blas Manuel, «Applications of artificial neural networks in three agro-environmental systems,» microalgae production, nutritional characterization of soils and meteorological variables management, 2019.
- [32] B. Samah et Boussekine Marwa, «Développement d'un système de diagnostic médical du diabète à l'aide d'algorithmes d'apprentissage automatique,» Université 20 Aout 1955, skikda, 2022.
- [33] m. Christopher, «Support Vector Machines,» Pattern Recognition., 2006.
- [34] P. Recognition., « the elements of statistical learning .,» 2009.
- [35] A. Ng, « An Introduction to Support Vector Machines.,» 2004.
- [36] A. P. |. D. P. |. S. S. |. S. T. |. H. Rathod, «XGBoost A lgorithm and Its Comparative Analysis,» INTERNATIONAL JOIRNAL OF NOVEL RESEARCH AND DEVELOPMENT (IJNRD), vol. 7, pp. 2456-4184, 2022.
- [37] N. BENMENIA et Khaoula SMAILI, «Approche DL pour l'identification des classes d'ARN codants et non codants,» Université Frères Mentouri

Constantine, Constantine, 2022.

[38] J. Brownlee, «A gentle introduction to long short-term memory networks by the experts,» Machine Learning Mastery 19., 2017.

[39] M. Macary, «Analyse de données massives en temps réel pour l'extraction d'informations sémantiques et émotionnelles de la parole,» Le Mans Université., 2022.

[40] G. Altan, «Deep learning-based mammogram classification for breast cancer,» International Journal of Intelligent Systems and Applications in Engineering, pp. 171-176, 2020.

[41] A. BOUAKKAZ, «DESIGN AND IMPLEMENTATION OF HOME ENERGY MANAGEMENT SYSTEM ADVISOR,» University 20 Aout 1955-Skikda, SKIKDA, 2023.

[42] Y. Zhang, «Diabetes Technology & Therapeutics,» "Deep learning for diabetes diagnosis: A systematic review., 2021.

[43] Y. Qi, « Random forest for bioinformatics.,» Ensemble machine learning: Methods and applications, pp. 307-323, 2012.

[44] Y. Derfoufi, « Programmation en langage Python,» CRMEF OUJDA, 2019.

[45] A. Djouama, «Un système de détection d ' intrusion pour la Cybersécurité,» University 20 Aout 1955-Skikda, SKIKDA, 2022.

[46] c. d. s. overflow, APPRENEZ Keras, Creative Commons BY-SA.

[47] C. Nguyen, «La GUI tkinter,» Université de Toulon et du Var.

[48] M. E. Fredrik Lundh, «Python Imaging Library Overview,» 2002.

[49] J. G. M.-T. L. P. W. Danijar Hafner, TensorFlow for Machine Intelligence.

[50] Y. CHEN, «Mise en place d'une application pour la création automatique d'entrepôt de données,» université toulouse 1, 2 Rue du Doyen-Gabriel-Marty 31042 TOULOUSE, 2022.