République Algérienne Démocratique et Populaire
Ministère de l'enseignement supérieur et de la recherche scientifique
Université de Bordj Bou Arreridj
Faculté des Mathématiques et Informatique
Département d'informatique

Rapport de projet de fin d'étude

En vue de l'obtention du diplôme de
Master Informatique
Spécialité : **Ingénierie de l'informatique Décisionnelle**

THEME :

EXPLORATION, VISUALISATION ET APPRENTISSAGE SUPERVISÉ SUR LES DONNEES COVID-19

Préparé par :

MEDJAAF IBTISSEM
TABTI ROMAISSA

Pr. NOUIOUA FARID - **Président**
Dr. CHARIKHI MOURAD - **Encadrant**
Dr. BOUMAZA FARID - **Examinateur**

Année universitaire : 2024/2025

Résumé

Ce mémoire s'inscrit dans une démarche complète de science des données appliquée à un cas réel : l'analyse et la modélisation de données médicales liées à la COVID-19. À partir d'une base de données clinique comportant plus de 5000 enregistrements et 100 variables, nous avons suivi les étapes fondamentales d'un projet de Data Science.

Nous avons commencé par une phase de prétraitement rigoureux : nettoyage, encodage, traitement des valeurs manquantes et validation des données. Ensuite, une analyse exploratoire approfondie nous a permis de mieux comprendre les distributions, les relations entre variables, ainsi que la structure des données.

La phase principale du projet repose sur la modélisation supervisée. Plusieurs algorithmes de classification (Random Forest, SVM, KNN, AdaBoost) ont été entraînés et comparés à l'aide de métriques d'évaluation (accuracy, F1-score, AUC-ROC). Cette approche nous a permis de sélectionner le modèle le plus performant pour prédire le résultat du test SARS-CoV-2.

Le travail s'est appuyé sur des outils modernes tels que Python, Google Colab, et des bibliothèques comme Scikit-learn, Pandas et Seaborn. Des ressources pédagogiques comme celles de Machine Learnia ont également enrichi notre démarche.

En conclusion, ce mémoire démontre l'efficacité de la Data Science dans le domaine médical, tout en mettant en évidence les enjeux techniques, éthiques et pratiques liés à l'usage de l'intelligence artificielle dans les systèmes de santé.

الملخص

يتناول هذا المشروع البحثي تطبيقا متكاملا لعلم البيانات في مجال الطب، من خلال تحليل بيانات سريرية تتعلق بمرض كوفيد-19. اعتمدنا على قاعدة بيانات تحتوي على أكثر من 5000 سجل و100 متغير، واتبعنا خطوات منهجية تشمل جميع مراحل مشروع علم البيانات.

بدأنا بمرحلة معالجة البيانات الأولية، والتي شملت التنظيف، الترميز، التعامل مع القيم المفقودة، والتحقق من صحة البيانات. تلتها مرحلة التحليل الاستكشافي التي سمحت بفهم التوزيعات والعلاقات بين المتغيرات.

تمحورت المرحلة الأساسية حول النمذجة بالإشراف، حيث قمنا بتطبيق عدة خوارزميات تصنيف (Random Forest، SVM، KNN، AdaBoost) وتقييم أدائها باستخدام مؤشرات مثل الدقة، معامل F1 ، ومعدل ROC-AUC. مكنتنا هذه المقارنة من اختيار النموذج الأفضل للتنبؤ بنتيجة اختبار SARS-CoV-2.

استخدمنا أدوات وتقنيات حديثة مثل Python و Google Colab ومكتبات Scikit-learn وPandas، بالإضافة إلى موارد تعليمية مثل دروس Machine Learnia.

في الختام، يبرهن هذا العمل على قدرة علم البيانات على دعم القرار في المجال الطبي، مع الإشارة إلى التحديات الأخلاقية والتقنية المرتبطة باستخدام الذكاء الاصطناعي في الرعاية الصحية.

Abstract

This thesis presents a comprehensive application of data science to a real-world healthcare case: analyzing and modeling clinical data related to COVID-19. Using a dataset of over 5,000 records and 100 variables, we followed the essential stages of a data science project.

The process began with thorough data preprocessing, including cleaning, encoding, handling missing values, and validating the dataset. Then, we conducted detailed exploratory data analysis to uncover distributions, relationships, and patterns.

The core of the project is supervised learning. We trained and evaluated several classification algorithms (Random Forest, SVM, KNN, AdaBoost), using metrics such as accuracy, F1-score, and ROC-AUC. This comparative analysis allowed us to select the most effective model for predicting SARS-CoV-2 test outcomes.

Our work was carried out using modern tools like Python, Google Colab, and libraries such as Scikit-learn, Pandas, and Seaborn. We also benefited from educational content like Machine Learnia's tutorials to enhance our methodology.

In conclusion, this thesis demonstrates the power of data science in healthcare, while also highlighting the technical, ethical, and operational challenges of integrating artificial intelligence into medical decision-making systems.

الإهداء

وأخيرًا...
رُفعت القبعة احترامًا لسنينٍ مضت، وابتدأ الوداع.
مع كل ابتسامة، مع كل دمعة، مع كل لقطة خلّدها القلب قبل العدسة،
يسدل الستار على فصلٍ من أجمل فصول حياتي،
فصلٍ كُتب بالشغف، وسُقي بالصبر، وتُوّج اليوم بالنجاح.

في البدء، الحمد لله...
الحمد لله الذي لطُف، ووفّق، ويسّر،
الذي جعل من الحلم دربًا، ومن التعب حصادًا، ومن الرجاء نورًا لا ينطفئ.

أهدي هذا التخرّج، هذه الثمرة التي نضجت بعد عناء، إلى من سكنوا القلب، وكانوا لي نورًا في عتمة الأيام

إلى أمي الغالية...
يا وطن القلب، وملجأ الروح، ودعاء لا يعرف الكلل،
كنتِ أماني حين ضاقت الحياة، ونوري حين بهت كل شيء.
كلّ نجاحي هو بعضٌ من عطائك، وكل خطوة خطوتها كانت ببركة دعواتكِ.

وإلى أبي العزيز...
يا سندًا لا يميل، وظلًّا لا يغيب،
يا من غرستَ في قلبي العزيمة، وسقيتَ طريقي بالحكمة،
كل تعبك نُقش في ذاكرة روحي، وكل وقوفك الصامت خلف ظهري، كان القوة التي لا تُقهر.

إلى رياض ومسعود... إخوتي، فخري، وامتداد قلبي،
كنتما كتفًا وسندًا حين ترنحت خطواتي،
وجودكما في حياتي طمأنينة، ودعاؤكما درع أمان.
شكرًا لأنكما كنتما دومًا هناك

إلى أمينة، ومونة، ومروى... أخواتي الزهرات،
أنبتكنّ الأيام في عمري حنانًا،
كنتنّ الضوء الذي يسبق الكلام، والفرح الذي لا يُشترى.
أحبكنّ بقدر ما كنتنّ لي بيتًا في قلب كل عاصفة.

إلى ليندة، صديقتي النقية الوفية،
يا من كنتِ لي وقت الضيق سَكينة، ووقت التعب دفءًا لا يوصف،
بوفائكِ، بصدقكِ، بتفاصيلكِ الصغيرة التي لا تُنسى...
كنتِ المرآة التي رأت فيّ النور حين عجزتُ عن رؤيته،
شكرًا لوجودكِ الجميل، وشكرًا لأنكِ كنتِ معي في الحكاية.

وإلى نفسي...
يا من قاومتِ الإنهاك، وتجاوزتِ الانكسارات، وثابرتِ حين خذل كل شيء،
يا من آمنتِ بالحلم، وسهرتِ لتزرعيه، وها هو اليوم يزهر بين يديكِ...
ارفعي رأسكِ، وابتسمي، لقد فعلتها...
وكانت الحكاية أجمل مما تمنيْتِ.

ولكل من مرّ في طريقي بكلمة، بدعاء، بابتسامة، أو بصمتٍ مشجّع...
شكرًا لأنكم كنتم هناك، شكرًا لأن الدرب بكم صار ألين،
وشكرًا لكل لحظة صنعت هذا الإنجاز، ولكل قلبٍ أحبني دون شرط.

هذا النجاح... لكم جميعًا..

مجعف ابتسام

الإهداء

وَآخِرُ دَعْوَاهُمْ أَنِ ٱلْحَمْدُ لِلَّهِ رَبِّ ٱلْعَٰلَمِينَ

الحمد لله عند البدء وعند الختام، على عونه حين تعبت، ونوره حين أظلم الدرب،
على القوة حين ضعفت، وعلى الثبات حين مالت بي الأيام.
سنواتٌ من السهر والكفاح، بين معادلات الرياضيات المعقّدة، وشيفرات الإعلام الآلي المتشابكة،
عشتُ فيها لحظات من التحدي والانكسار، ومن الصبر والانتصار.
سنواتٌ لم تكن سهلة، لكنها شكّلتني، وبنت فيّ روحًا لا تنكسر، وإرادة لا تلين.
وها أنا اليوم أكتب الفصل الأخير من حكايةٍ طالت،
حكايةٍ اجتهدت فيها الروح، وجاهدت فيها النفس، حتى بلغتُ الحلم.
فالحمد لله أولًا وآخرًا، ظاهرًا وباطنًا، على ما مضى، وعلى ما هو آت.

إلى أمي، قرة عيني، وجنّتي في الدنيا
دعاؤكِ سرُّ نجاحي، وصبركِ نوري في ظلمة الأيام، وحنانكِ طمأنينة قلبي في كل ضعف.
كلّ حرفٍ كتبته، وكلّ خطوةٍ خطوتها، كانت ببركة دعواتك، وبدفء قلبكِ.
أسأل الله أن يقرّ عيني بكِ، كما قرّ قلبي بحبك، وأن يجزيكِ عني خير الجزاء.

وإلى أبي، سيد الرجال، وظلي الثابت
يا من سهرتَ ليَنام قلبي مطمئنًا، وتعبتَ ليَنعم أيامي بالراحة،
يا من زرعتَ الحلم وسقيته بالصبر والعطاء، حتى أثمر اليوم فخرًا وامتنانًا.
لكَ من قلبي دعاء لا ينقطع، وعُمرٌ من الحب لا يُقاس.

إلى سعيد، أخي الوحيد، ونصفي الثابت
رفيقي في الحلم، وصمتي، وقوتي...
كنت السند حين مالت الأيام، والكتف الذي لا يميل. كل الشكر لا يكفيك.

إلى أخواتي الحبيبات:
إلى صباح، أول الحنان، الحضن الذي ما خذلني يومًا،
إلى كريمة، الأم الثانية، التي منحتني بهجة إسراء وأسيل،
إلى أحلام، النبض المضيء في عتمة الدرب،
وإلى سارة، آخر الزهرات، التي تفتّحت في قلبي أملاً لا يذبل.

إلى صديقتي الوفية، رفيقة الدرب رحاب
تسع سنوات من الصحبة، من الضحك والدمع،
كنتِ وما زلتِ الحضن الذي ألجأ إليه، والبصمة التي لا تُمحى من حكاياتي.
شكرًا لثباتك، لصدقك، لكونك أنتِ.

وإلى ابتسام، زميلتي في مذكرة التخرج، وشريكة التعب الجميل
يا من تقاسمنا معها المشقة كما تقاسمنا الحلم،
يا من وقفنا معًا في وجه الصعوبات، وتجاوزنا الخلافات، واختلفنا ثم عدنا أقوى،
كانت أيام العمل معك مليئة بالكدّ والتحديات، لكنها كانت أيضًا مليئة بالصدق،
بالنقاشات التي بَنت، وباللحظات التي ساندت، وبالقرارات التي كبّرنا بها معًا.
شكرًا على صبرك، على إخلاصك، على أنكِ كنتِ هناك حين احتاج الأمر قلبًا نابضًا بالإرادة.
لكِ مني كل التقدير والامتنان، وكل الدعاء بأن يفتح الله لكِ أبواب التوفيق كما فتحتِ لي أبواب الدعم.
وإنّي أرجو أن تلامس هذه الكلمات قلبكِ كما لامس حضوركِ قلبي...
دمتِ متألقة، مبدعة، ودام عطاؤكِ لا يُنسى.

إلى من علمونا، وغرسوا فينا العلم بلا ملل ولا كلل
أساتذتي الأكارم، تعلّمت منكم أكثر من المقررات... تعلمت الحياة والنبل والاحترام.
لكم كل الدعاء، وكل الامتنان.

وإلى فلسطين، إلى غزة، إلى من خطّوا أحلامهم على جدران الجامعات
وحملوا دفاترهم في طريقهم إلى الشهادة،
إلى طلبة غزة الجامعيين الذين رحلوا قبل أن يُنادى بأسمائهم في حفلات التخرّج،
لكنّهم صاروا نورًا لا ينطفئ في سماء العلم والكرامة.
سلامًا على أرواحكم الطاهرة... وموعدنا عند ربٍّ لا ينسى.

وأهدي هذا التخرّج... لنفسي الطموحة،
يا من حملتِ الأعباء في صمت، وصبرتِ رغم التعب،
يا من وقفتِ في وجه الريح، ولم تستسلمي...
ها أنتِ اليوم على عتبة المجد، فارفعي رأسكِ عاليًا... لقد فعلتها.

ثابتي رميساء

Remerciements

*Nous adressons nos plus sincères remerciements à notre encadrant, Monsieur **Charikhi Mourad**, pour son accompagnement précieux, sa disponibilité constante et ses conseils avisés tout au long de la réalisation de ce mémoire. Sa rigueur scientifique et son souœtien ont été essentiels à l'aboutissement de ce travail.*

Nous exprimons également notre profonde gratitude à l'ensemble des enseignants de la Faculté de Mathématique et de l'Informatique, qui nous ont accompagnés durant tout notre parcours universitaire. Leur savoir, leurs conseils et les nombreuses informations qu'ils nous ont transmises ont largement contribué à enrichir notre formation et à construire nos compétences.

Nous tenons aussi à remercier respectueusement les professeurs qui, par leur exigence et leur discipline rigoureuse, ont su nous pousser à nous dépasser. Bien que parfois leurs méthodes aient pu sembler strictes, nous reconnaissons aujourd'hui la valeur formatrice de ces expériences qui nous ont forgés tant sur le plan académique que personnel.

*Nous exprimons également nos remerciements les plus distingués aux membres du jury, Monsieur **Boumaza Farid** et Monsieur **Nouioua Farid**, pour avoir accepté d'évaluer ce travail. Leurs remarques pertinentes et leurs observations constructives seront, à n'en pas douter, d'une grande utilité pour la suite de notre parcours académique et professionnel.*

À toutes celles et ceux qui ont contribué, de près ou de loin, à notre apprentissage et à l'aboutissement de ce mémoire, nous adressons nos plus sincères remerciements pour leur engagement et leur dévouement.

TABLE DES MATIÈRES

TABLE DES FIGURES

LISTE DES TABLEAUX

LISTE D'ABRÉVIATIONS

AI	Intelligence Artificielle
AUC-ROC	Area Under the Curve - Receiver Operating Characteristic
CPU	Central Processing Unit
EDA	Exploratory Data Analysis (Analyse exploratoire des données)
FP	Faux Positifs
FN	Faux Négatifs
GPU	Graphics Processing Unit
KNN	K-Nearest Neighbors (Plus proches voisins)
KDD	Knowledge Discovery in Databases (Extraction de connaissances à partir de bases de données)
ML	Machine Learning (Apprentissage automatique)
SVM	Support Vector Machine (Machine à vecteurs de support)
TP	Vrais Positifs
TN	Vrais Négatifs
COVID-19	Coronavirus Disease 2019
PCR	Polymerase Chain Reaction (Réaction en chaîne par polymérase)
RGPD	Règlement Général sur la Protection des Données
HIPAA	Health Insurance Portability and Accountability Act
ICU	Intensive Care Unit (Unité de soins intensifs)
JSON	JavaScript Object Notation
XML	eXtensible Markup Language
CSV	Comma-Separated Values (Valeurs séparées par des virgules)
MCV	Mean Corpuscular Volume (Volume globulaire moyen)
MCH	Mean Corpuscular Hemoglobin (Teneur moyenne en hémoglobine)

TPU	Tensor Processing Unit
IoT	Internet of Things (Internet des objets)
CRP	C-Reactive Protein (Protéine C-réactive)

INTRODUCTION GÉNÉRALE

La quantité de données numériques générées quotidiennement par les systèmes de santé est colossale. Ces données massives sont souvent hétérogènes et complexes. Elles représentent une opportunité majeure pour améliorer la prise de décision médicale, anticiper les risques et personnaliser les traitements. Cependant, leur exploitation efficace demeure un défi tant technique que méthodologique. L'objectif principal de ce mémoire est donc de répondre à la question suivante : comment peut-on utiliser la science des données et les techniques de machine learning pour analyser, générer de la connaissance et surtout prédire, à partir de données cliniques, le résultat d'un test de dépistage de la COVID-19 ? Pour répondre à cette problématique, nous appliquerons le processus général de la data science, qui commence par une étape de prétraitement visant à préparer la base de données pour l'analyse. L'étape suivante consistera en une analyse statistique descriptive ainsi qu'en une exploration des données afin d'identifier les variables pertinentes. Enfin, nous appliquerons plusieurs algorithmes de classification — SVM, KNN, Random Forest et AdaBoost — pour prédire, à partir de données cliniques, le résultat d'un test de dépistage de la COVID-19. Notre manuscrit est organisé selon la structure suivante :

Chapitre 1 – Science des données : Fondements et enjeux Dans ce chapitre, nous allons exposer les bases théoriques de la science des données en définissant ses concepts-clés, son historique, ses outils et ses domaines d'application, notamment, dans le domaine de la santé. Nous allons mis en évidence le rôle central que joue la Data Science dans l'exploitation des données médicales à grande échelle.

Chapitre 2 – Prétraitement de la data base COVID-19 Ce chapitre a été consacré tout d'abord à la présentation de la base données téléchargée. Il a ensuite, exposé le processus de nettoyage et de préparation des données : traitement des valeurs manquantes, encodage des variables catégorielles, suppression des colonnes peu exploitables et validation de la structure finale du jeu de données. Ce travail a été essentiel pour garantir la qualité des analyses ultérieures.

Chapitre 3 – Analyse exploratoire Dans ce chapitre, nous présentons une étude statistique approfondie visant à identifier les distributions, les corrélations, les tendances, ainsi que les éventuelles anomalies des variables. L'objectif est de mieux comprendre les facteurs cliniques associés à un résultat positif ou négatif au test SARS-CoV-2. Des visualisations graphiques (histogrammes, cartes de chaleur, nuages de points) seront utilisées pour appuyer notre interprétation.

Chapitre 4 – Classification supervisée Nous allons appliquer qutre algorithmes de classification supervisée (Random Forest, SVM, KNN, AdaBoost) afin de prédire le résultat du test COVID-19 à partir des variables cliniques du patient. Nous allons effectuer aussi, Une validation croisée et une comparaison des performances via des métriques comme la précision et le F1-score. Nous allons ainsi chercher le modèle le plus performant pour notre cas d'usage. Finalement, nous terminerons notre mémoire par une conclusion générale.

CHAPITRE 01

Science des données :

Fondements et enjeux

1.1 Introduction

À l'ère du numérique, les données sont devenues une ressource stratégique pour les entreprises, les gouvernements et les institutions académiques. Leur volume, leur variété et leur vitesse de production ont donné naissance à un nouveau champ interdisciplinaire : la science des données (Data Science). Cette discipline repose sur la combinaison de compétences en statistiques, informatique, intelligence artificielle et visualisation, dans le but de transformer des données brutes en informations exploitables, en connaissances, voire en décisions intelligentes. Aujourd'hui, la science des données joue un rôle crucial dans la transformation numérique de

tous les secteurs : finance, santé, éducation, industrie, marketing, et bien d'autres. Elle permet notamment d'optimiser les processus, d'anticiper les comportements, d'automatiser certaines tâches et

de personnaliser les services. Le recours à des outils comme Python, R, Hadoop ou des frameworks d'apprentissage automatique permet de traiter des volumes massifs de données grâce à une puissance de calcul croissante, notamment via le cloud computing et les architectures distribuées. Ce mémoire s'inscrit dans cette dynamique en proposant une étude approfondie des fondements de la science des données, de ses outils, de ses applications concrètes, et de ses enjeux éthiques et techniques. Il servira également de socle pour le développement d'un système de recommandation basé sur l'analyse de données, démontrant ainsi l'impact direct de la Data Science sur l'amélioration de services intelligents, tels que l'aide à l'emploi.

1.2 la science des données (Data science)

1.2.1 Définition

Le domaine de la science des données combine des outils et des méthodes issus de plusieurs discipline tel que des mathématiques, des statistiques, de l'informatique, de l'intelligence artificielle et du génie logiciel. Son objectif est d'extraire, d'analyser et d'interpréter des données brutes (structurées ou non). Elle permet de transformer ces données en informations exploitables afin de soutenir la prise de décision, d'améliorer les services, de prédire des tendances futures et de relever des défis complexes, tant dans le monde de l'entreprise que dans d'autres domaines. Grâce à des techniques comme le Machine Learning et le Deep Learning, la science des données joue aujourd'hui un rôle central dans l'innovation, la personnalisation des expériences et l'optimisation des performances à grande échelle[1]. Nous allons présenter dans ce chapitre, les bases théoriques de la science des données en définissant ses concepts-clés, son historique, ses outils et ses domaines d'application, notamment, dans le domaine de la santé.

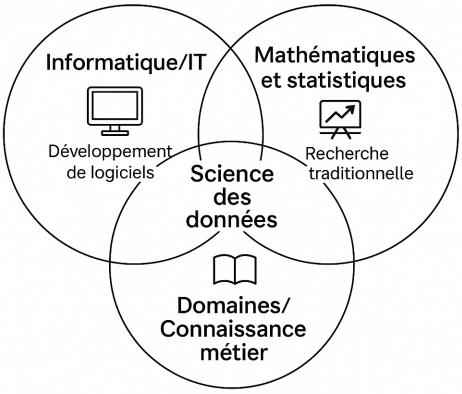

FIGURE 1.1 – Les composantes essentielles de la science des données

1.3 Historique et enjeux de la science des données (Data Science)

La science des données s'impose aujourd'hui comme une discipline incontournable dans un monde dominé par l'information numérique. Elle repose sur la capacité à collecter, traiter et analyser des volumes massifs de données afin d'en extraire des connaissances utiles à la prise de décision. Pour mieux comprendre son importance actuelle, il est essentiel de retracer l'évolution historique et d'examiner les principaux enjeux contemporains.

1.3.1 Historique de la science des données

La science des données, apparue dans les années 1970, résulte de l'intégration entre les statistiques, l'intelligence artificielle et les bases de données. Elle permet d'extraire des connaissances utiles à partir de grandes masses de données.[2]

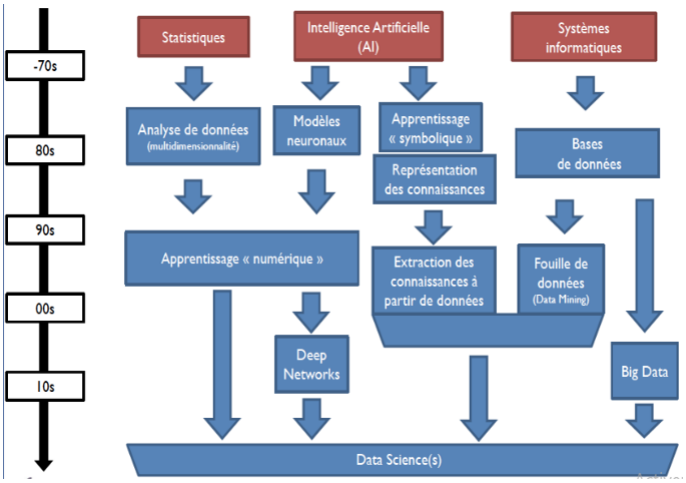

FIGURE 1.2 – Historique de la science des données (data science) [3]

1.3.2 Enjeux de la science des données

La science des données est confrontée à plusieurs défis majeurs liés à la nature, au volume, à la qualité et à la sécurité des données. Ces enjeux influencent la manière dont les données sont collectées, traitées, analysées et interprétées

Big Data

La croissance exponentielle des données rend leur stockage, leur traitement et leur analyse de plus en plus complexes. [4]

- **Variété :** Les données peuvent être structurées, semi-structurées ou nonstructurées, ce qui nécessite des outils et des méthodes adaptés.
- **Vitesse :** La nécessité de traiter les données en temps réel ou quasi réel pour prendre des décisions rapides.

Le volume et la complexité des données

La quantité croissante de données s'accompagne de problèmes de qualité et de cohérence.[5]

- **Précision :** Des données erronées ou incomplètes peuvent conduire à des résultats biaisés.
- **Cohérence :** Les données doivent être cohérentes entre elles pour garantir la fiabilité des analyses.
- **Pertinence :** Il est essentiel de sélectionner les données pertinentes pour répondre à une question spécifique.

Biais et éthique dans l'analyse des données

L'utilisation de données biaisées peut entraîner des décisions discriminatoires ou injustes.[6]

- **Biais dans les algorithmes :** Les modèles de machine learning et d'intelligence artificielle sont souvent biaisés si les données d'entraînement sont elles-mêmes biaisées.

Sécurité des données

La science des données implique la manipulation de grandes quantités d'informations sensibles, ce qui soulève des préoccupations majeures en matière de sécurité.[7]

- **Cyberattaques et fuites de données :** La science des données repose souvent sur des volumes massifs d'informations stockées dans des bases de données ou des infrastructures cloud. Cela en

fait une cible attrayante pour les cyberattaques. Assurer la sécurité des infrastructures, l'intégrité des données, et la résilience des systèmes est donc essentiel.

- **Protection contre les manipulations :** Les données peuvent également être manipulées pour produire des résultats biaisés ou malhonnêtes (comme les fausses nouvelles ou la désinformation).

1.4 Facettes et types de données

La science des données repose sur une bonne compréhension des types de données, qui influencent directement les méthodes d'analyse et les outils utilisés. On distingue principalement trois catégories : les données structurées, semi-structurées et non structurées.[8]

1.4.1 Données structurées

Les données structurées sont organisées selon un schéma rigide (modèle fixe), généralement sous forme de tableaux (lignes/colonnes). Chaque valeur occupe une position précise, ce qui facilite les opérations de recherche, tri et analyse.

Exemples :

- Bases de données relationnelles (MySQL, Oracle)
- Fichiers Excel ou CSV
- Tableaux issus d'un logiciel ERP ou CRM

1.4.2 Données non structurées

Les données non structurées ne suivent aucun modèle fixe. Elles sont généralement riches en contenu mais difficiles à analyser automatiquement sans techniques spécifiques (ex. : traitement du langage naturel, vision par ordinateur). distinguées les unes des autres.

Exemples :

- Textes libres (rapports, articles, e-mails)
- Images, vidéos, sons
- Pages web non structurées
- Documents PDF ou Word

1.4.3 Données semi-structurées

Les données semi-structurées possèdent une organisation partielle : elles contiennent des balises ou une hiérarchie (ex : paires clé-valeur), sans suivre un modèle strict comme les données structurées. Elles sont plus flexibles et adaptables.

Exemple :

- Fichiers XML ou JSON
- Logs système
- Certains formats d'e-mails ou de formulaires web

1.5 Le fonctionnement de la science des données

La science des données suit un processus en étapes : collecte, traitement, analyse, modélisation et interprétation, afin d'extraire des informations utiles à partir de données brutes

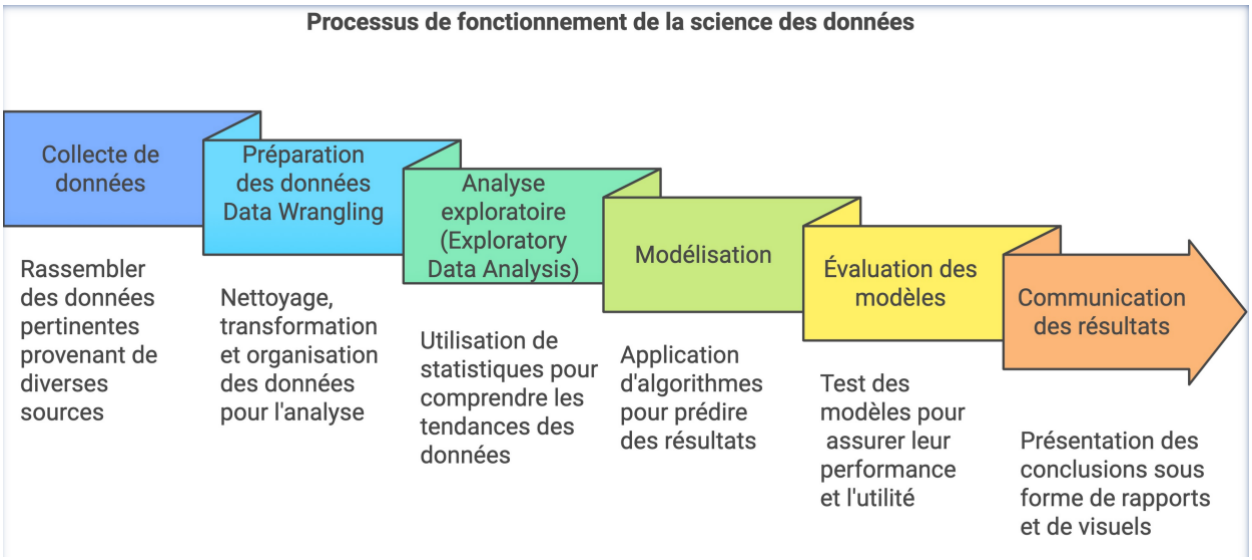

FIGURE 1.3 – Processus de fonctionnement de la science des données (data science) [9]

1.5.1 Collecte de données

La collecte de données constitue la première étape du processus de science des données. Elle consiste à rassembler des données pertinentes à partir de sources diverses telles que des bases de données internes, des fichiers plats (CSV, Excel), des API, des plateformes en ligne ou des capteurs IoT. La qualité, la quantité et la diversité des données collectées ont une influence directe sur la performance des analyses futures

1.5.2 Prétraitement et nettoyage

Une fois les données collectées, il est nécessaire de les nettoyer, de les transformer et de les organiser pour les rendre exploitables. Cette étape comprend la gestion des valeurs manquantes, la détection et la suppression des doublons, la normalisation des formats, l'encodage des variables catégorielles et la standardisation des valeurs numériques. Une bonne préparation assure une base solide pour l'analyse statistique et la modélisation.

1.5.3 Analyse Exploratoire de données et visualisation (EDA)

L'analyse exploratoire permet de comprendre la structure des données à l'aide de statistiques descriptives et de visualisations. Elle aide à identifier les distributions, les relations entre variables, les éventuelles anomalies, les corrélations, et à formuler des hypothèses. Cette phase est essentielle pour orienter les choix de modèles ou de techniques d'apprentissage.

1.5.4 Modélisation

La modélisation consiste à appliquer des algorithmes statistiques ou d'apprentissage automatique sur les données préparées afin de prédire ou classifier des observations. Les modèles peuvent être supervisés (régression, classification) ou non supervisés (clustering, réduction de dimension). Cette étape implique aussi l'optimisation des hyperparamètres pour améliorer les performances du modèle.

1.5.5 Évaluation des modèles

Une fois les modèles construits, ils doivent être évalués pour vérifier leur pertinence et leur robustesse. Cela se fait à l'aide de jeux de données de test et de métriques telles que l'exactitude, la précision, le rappel, le F1-score ou l'AUC-ROC. L'objectif est de sélectionner le modèle le plus adapté au problème tout en évitant le sur-apprentissage.

1.5.6 Interprétation et prise de décision

La dernière étape du processus est la communication des résultats obtenus. Il s'agit de présenter les insights sous forme de tableaux, graphiques, rapports ou dashboards interactifs à destination des décideurs ou parties prenantes. Une bonne visualisation des résultats facilite leur interprétation et leur intégration dans les processus décisionnels de l'entreprise ou de l'organisation.

1.6 Limportence de la sciense des données

La science des données est importante car elle permet de transformer des volumes massifs de données en informations utiles, exploitables et stratégiques. À l'ère numérique, les organisations modernes génèrent et collectent une quantité exponentielle de données textuelles, audio, vidéo, images issues de multiples sources : applications web, objets connectés, systèmes de paiement, réseaux sociaux. [10] Sans traitement ni interprétation, ces données n'ont aucune valeur. La science des données combine des outils mathématiques, des algorithmes d'intelligence artificielle, des techniques statistiques et des technologies informatiques pour :

- extraire du sens de ces données.
- faciliter la prise de décision.
- anticiper les tendances.
- innover dans des secteurs aussi variés que la santé, la finance, l'e-commerce, l'industrie ou l'administration.

Elle permet ainsi aux entreprises de mieux comprendre leur environnement, d'optimiser leurs performances et de créer de la valeur à partir d'une ressource devenue stratégique .

1.7 Le rôle stratégique de la science des données

La science des données a pour objectif de transformer de grandes quantités de données brutes en informations utiles et exploitables. Elle permet aux organisations de mieux comprendre leur environnement, de prendre des décisions fondées sur les données, de prévoir les tendances futures et d'optimiser leurs actions .[11]

Elle repose sur quatre types d'analyse :

— **Analyse descriptive** : répond à la question « Que s'est-il passé ? ». Elle utilise des graphiques, des tableaux et des indicateurs pour synthétiser l'information.

— **Analyse diagnostique** : vise à expliquer « Pourquoi cela s'est-il produit ? » en explorant les liens entre les variables.

— **Analyse prédictive** : anticipe « Que va-t-il se passer ? » grâce à des modèles statistiques ou de machine learning.

— **Analyse prescriptive** : propose des recommandations en réponse à « Que devons-nous faire ? » en simulant différents scénarios.

Ces analyses sont aujourd'hui déployées dans de nombreux secteurs, tels que la santé, la finance ou l'e-commerce

1.8 La puissance de calcul en science des données

La Data Science nécessite une puissance de calcul importante pour traiter efficacement les volumes massifs de données générés quotidiennement. Qu'il s'agisse d'analyse exploratoire, de modélisation ou d'implémentation de techniques de Machine Learning ou de Deep Learning, ces traitements s'avèrent souvent très gourmands en ressources informatiques. Ainsi, la disponibilité d'une infrastructure adaptée est un facteur clé de réussite dans tout projet de science des données. Afin de répondre à ces exigences, plusieurs solutions technologiques sont couramment utilisées :[12]

1.8.1 Le cloud computing

permet de louer, à la demande, de la puissance de calcul (processeurs, mémoire, GPU/TPU) auprès de fournisseurs tels que Google Cloud, Amazon Web Services (AWS), Microsoft Azure, ou encore IBM Cloud. Cette approche permet aux entreprises, même de petite taille, de bénéficier d'une infrastructure puissante sans avoir à investir dans du matériel coûteux.

1.8.2 La parallélisation

consiste à exécuter plusieurs opérations de calcul simultanément sur des cœurs de processeurs multiples (CPU multi-core) ou sur des cartes graphiques (GPU), particulièrement efficaces dans le traitement matriciel et l'entraînement de réseaux de neurones.

1.8.3 Les systèmes distribués

tels que Apache Spark ou Hadoop, permettent de répartir des tâches sur plusieurs machines, ce qui facilite le traitement de très grands jeux de données tout en réduisant les temps d'exécution.

À titre d'exemple, des plateformes comme TensorFlow Cloud ou Google Colab Pro offrent la possibilité de louer temporairement des ressources GPU ou TPU afin de développer, entraîner et tester des modèles avancés d'apprentissage automatique. Cette flexibilité permet d'abaisser les coûts d'infrastructure tout en conservant des performances élevées, ce qui est particulièrement bénéfique pour les startups, les PME ou les projets académiques.

1.9 Les domaines d'applications de la science des données

La science des données permet d'analyser les données pour en extraire des informations utiles, améliorer la prise de décision et résoudre des problèmes complexes dans divers secteurs. la **FIGURE1.4** représente les défirent secteurs

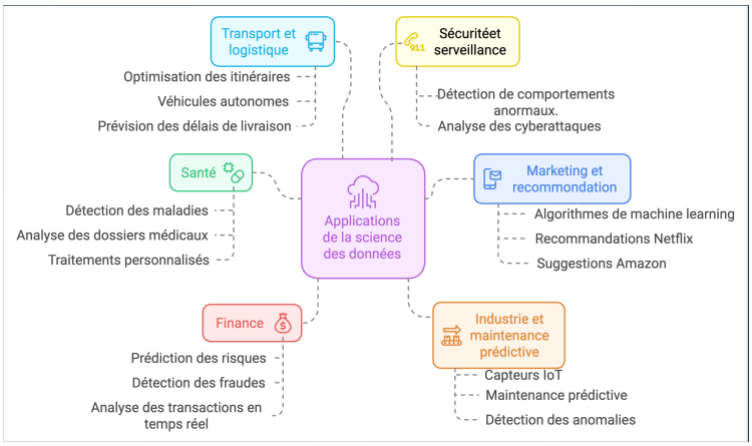

FIGURE 1.4 – Applications de la science de données(data science)[13]

1.9.1 Transport et logistique

La Data Science permet de nombreuses avancées dans ce secteur :

— Optimisation des itinéraires et réduction de la consommation énergétique .

— Prédiction des retards de vols, des congestions routières ou des pannes mécaniques .

— Gestion en temps réel des systèmes de transport .

— Amélioration de la logistique des livraisons et de la chaîne d'approvisionnement.

1.9.2 Santé

Dans le secteur médical, la Data Science permet :

— D'améliorer le diagnostic et les traitements personnalisés à travers l'analyse de données médicales et génomiques .

— De suivre l'évolution des épidémies et des maladies infectieuses .

— D'optimiser la gestion hospitalière et la distribution des ressources .

— De développer des solutions de télémédecine et de surveillance en temps réel.

1.9.3 Finance

Dans le domaine de la finance, la Data Science joue un rôle central dans la gestion des risques, la détection des fraudes et l'optimisation des investissements. Les banques et les sociétés d'investissement s'appuient sur des modèles prédictifs pour :

— Identifier les tendances du marché .
— Évaluer la solvabilité des clients .
— Automatiser les stratégies de trading algorithmique.
— Détecter des comportements anormaux ou frauduleux .

1.9.4 Marketing et publicité

Les entreprises utilisent la Data Science afin de :

— Analyser le comportement des consommateurs et segmenter les profils clients .
— Personnaliser les campagnes publicitaires et ajuster les stratégies marketing en temps réel .
— Mettre en place des systèmes de recommandation, comme ceux utilisés par Amazon ou Netflix.
— Réaliser des analyses prédictives sur l'engagement et la fidélité des clients.

Dans ce contexte, on parle de Big Data Marketing, car les volumes de données traités sont particulièrement importants.

1.9.5 Sécurité et cybersurveillance

La science des données joue un rôle essentiel dans la protection des systèmes numériques en analysant des flux de données massifs pour détecter les menaces en temps réel. :

— Détecter les comportements suspects et les fraudes.
— Identifier les cyberattaques et renforcer la cybersécurité.
— Mettre en place des systèmes de surveillance intelligents.
— Automatiser les alertes dans les environnements critiques.

1.9.6 Industrie et maintenance prédictive

En milieu industriel, la science des données permet d'anticiper les défaillances et d'optimiser la performance des équipements grâce à l'analyse des données issues des capteurs et des machines.

— Exploiter les capteurs IoT pour surveiller les équipements en temps réel.
— Mettre en place une maintenance prédictive afin d'éviter les pannes coûteuses.

— Détecter les anomalies dans les processus de production.
— Améliorer l'efficacité opérationnelle grâce à l'analyse des données machines.

1.10 Avantages de la science des données dans le domaine de la santé

— **Analyse prédictive :** grâce à la Science de données, des modèles prédictifs peuvent être élaborés pour anticiper les épidémies, les réadmissions de patients et les besoins en ressources de santé, facilitant ainsi des actions anticipatives.
— **Médecine personnalisée :** en scrutant les informations génétiques et médicales, la science des données est capable d'ajuster les stratégies de soins pour chaque patient, maximisant de ce fait l'efficacité tout en réduisant au minimum les effets indésirables.
— **Aide à la décision clinique :** les outils basés sur les données fournissent une aide à la décision en temps réel aux professionnels de santé, en suggérant des options de diagnostic et de traitement en fonction des données des patients et de la littérature médicale.
— **Découverte de médicaments :** l'analyse avancée des données et l'apprentissage automatique aident à identifier les médicaments candidats potentiels, accélérant ainsi le développement de médicaments et réduisant les coûts.
— **Surveillance des maladies :** la science des données surveille les schémas et les tendances des maladies à l'aide de données épidémiologiques, permettant une détection précoce des épidémies et une meilleure allocation des ressources.[14]

1.11 Limites et inconvénients de la science des données dans le domaine de la santé

Bien que la science des données améliore les soins et la gestion médicale, elle présente aussi des limites qu'il convient de considérer pour garantir une utilisation éthique et sécurisée.[15]

Problèmes de confidentialité et de sécurité des données :

- Les données médicales sont sensibles et exposées à des risques de fuites ou de piratage.
- Une mauvaise gestion des accès peut entraîner des violations éthiques et juridiques.
- Des réglementations strictes comme le RGPD (UE) ou HIPAA (USA) encadrent leur usage

Précision et validité limitées des données :

- Les données peuvent être incomplètes, erronées ou mal structurées.

- La diversité des sources (dossiers, capteurs, imagerie) engendre une hétérogénéité complexe à gérer.
- Le biais de représentativité peut fausser les analyses, surtout en cas de population sous-représentée

Risque de déshumanisation du soin :

- L'automatisation peut réduire le rôle du médecin dans la relation avec le patient.
- Dépendance accrue aux décisions algorithmiques, parfois opaques (black box).
- Manque de transparence dans certaines décisions cliniques basées sur l'IA.

Problèmes éthiques liés à l'interprétabilité des modèles :

- Les algorithmes complexes sont difficiles à interpréter pour les médecins et les patients.
- Responsabilité juridique floue en cas d'erreur (médecin ? développeur ? hôpital ?).
- Doute sur la possibilité d'un consentement éclairé par le patient face à des systèmes peu compréhensibles.

1.12 Conclusion

Ce premier chapitre a permis de poser les bases conceptuelles et technologiques de la science des données. À travers une exploration de ses origines, de ses outils, de ses méthodes d'analyse et de ses champs d'application, il apparaît clairement que la Data Science est bien plus qu'un effet de mode : elle constitue une réponse puissante et innovante aux défis posés par l'explosion des données.

Les exemples abordés dans les domaines de la santé, du marketing, de la finance ou encore de l'environnement montrent à quel point cette discipline peut transformer notre manière d'interagir avec les données, de prendre des décisions, et d'anticiper le futur. Cependant, ces promesses s'accompagnent de limites — notamment en matière d'éthique, de sécurité et d'interprétabilité — qu'il convient d'intégrer dans toute démarche responsable. Ces fondements sont indispensables pour comprendre les chapitres suivants, notamment ceux qui porteront sur la modélisation et la mise en œuvre d'un système intelligent de classification supervisée, illustrant concrètement la puissance et les défis de la Data Science dans un contexte réel.

CHAPITRE 02

Prétraitement de la data base COVID-19

2.1 Introduction

Dans le cadre de toute étude basée sur l'apprentissage automatique, la compréhension et la préparation des données constituent des étapes fondamentales. Ce chapitre se consacre à l'exploration de la base de données utilisée pour la détection du COVID-19. Celle-ci contient 5 644 enregistrements et 111 variables décrivant divers aspects cliniques, biologiques et administratifs des patients. L'analyse débute par une inspection globale des types de données, suivie par une classification des variables selon leur nature (qualitative ou quantitative). Ensuite, des étapes essentielles du prétraitement sont mises en œuvre, notamment l'identification des valeurs manquantes, leur gestion (par suppression ou imputation), l'encodage des variables catégorielles et la validation finale du dataset. Ces traitements visent à garantir l'intégrité, la cohérence et la qualité des données avant leur exploitation par les modèles d'apprentissage automatique.

2.2 Environnement technologique et bibliothèques utilisées

Pour ce projet, nous avons utilisé un ensemble d'outils numériques qui nous ont permis de traiter, visualiser et examiner les données avec précision et professionnalisme. On peut classer ces outils en deux catégories principales qui se complètent : d'une partie, il y a les environnements de développement qui nous ont permis de programmer et d'expérimenter dans un cadre interactif ; d'autre part, on trouve les bibliothèques Python spécialisées en science des données, offrant des fonctionnalités avancées pour le traitement, l'exploration et la modélisation des données.

2.2.1 Environnement de développement

Google Colab

Pour le développement de notre projet, nous avons choisi Google Colaboratory. Google Colab est un outil gratuit basé sur le cloud ; il propose une solution idéale pour écrire et exécuter du code Python en utilisant des notebooks interactifs. Par ailleurs, on retrouve plusieurs avantages à travailler sur cet environnement[16] :

— aucune installation locale requise
— possibilité d'utiliser les ressources GPU/TPU
— facile et collaboration en temps réel
— Environnement adapté à tous les projets liés à la science des données et au machine learning.

Ainsi, Google Colab s'est imposé comme un choix relevant pour le développement, l'expérimentation et l'analyse de nos données dans le cadre de ce projet

Python

est un langage de programmation interprété, open source, connu pour sa simplicité syntaxique, sa lisibilité et sa large communauté. Il est particulièrement adapté à la science des données grâce à ses nombreuses bibliothèques spécialisées (comme Pandas, NumPy, ou Scikit-learn), ce qui en fait un outil de référence pour l'analyse statistique, le machine learning et le traitement de données volumineuses.[17]

Bibliothèques Python utilisées

Les bibliothèques Python jouent un rôle essentiel en science des données en fournissant des outils puissants pour la manipulation, l'analyse, la visualisation et la modélisation des données .[18]

- **Pandas :** pour la manipulation des données sous forme de tableaux (DataFrames), la gestion des valeurs manquantes, les regroupements et les résumés statistiques. .
- **NumPy :** pour la manipulation de structures numériques plus performantes et certaines opérations mathématiques.
- **Matplotlib et Seaborn :** pour la visualisation des données à travers des graphes, histogrammes, heatmaps, boxplots,.
- **Scikit-learn :** pour le prétraitement (encodage, standardisation), l'entraînement de modèles prédictifs et l'évaluation des performances.
- **StandardScaler / MinMaxScaler (de scikit-learn) :** pour la normalisation ou la standardisation des variables numériques.

2.3 Prétraitement (Preprocessing)

2.3.1 Définition

Le prétraitement des données désigne l'ensemble des opérations que l'on applique aux données brutes afin de les rendre compréhensibles et utilisables pour les algorithmes standards de machine learning. Avant de confier les données à un modèle, il est en effet vital de les nettoyer, les transformer et les structurer correctement. Cette phase peut comporter diverses tâches essentielles : traiter les valeurs absentes, transformer les facteurs textuels ou catégoriels en données numériques, unifier les échelles de mesure, ou détecter et rectifier les valeurs déviantes (ou aberrantes). [19]

Au bout du compte, le prétraitement représente une phase cruciale du projet. Des données mal agences peuvent mener à des conclusions biaisées, voire complètement incorrectes, même avec les algorithmes les plus performants. En revanche, un prétraitement adéquat augmente la fiabilité des études et améliore considérablement les performances des modèles de machine learning.

En fait, un prétraitement efficace permet de :

— booster la qualité des données

— limiter le bruit et les biais.

— garantir une convergence des modèles d'apprentissage plus rapide et plus fiable.

Cette étape, dirigée par les découvertes tirées de l'analyse exploratoire, englobe plusieurs phases cruciales.

2.3.2 Présentation de la base de données COVID-19

Dans le cadre de notre travail, nous avons choisi de nous appuyer sur la base de données intitulée "Diagnosis of COVID-19 and its clinical spectrum", accessible via la plateforme Kaggle à l'adresse suivante : `https://www.kaggle.com/datasets/einsteindata4u/covid19`

Cette base de données a été publiée par l'utilisateur @einsteindata4u, un nom qui semble faire référence à l'hôpital Albert Einstein de São Paulo (Brésil), bien qu'aucune affiliation officielle ne soit explicitement mentionnée sur la page de la dataset. Elle se compose de 5 644 lignes (représentant des patients) et de 111 colonnes (correspondant à des variables cliniques, biologiques et administratives), ce qui témoigne d'une richesse informationnelle notable .

Justification de notre choix

Pour plusieurs motifs personnels et scientifiques, nous avons choisi cette base de données. D'une part, la question du COVID-19 reste pertinente et constitue un défi de première importance en matière de santé publique. L'analyse de ces données nous aide, à notre niveau, à mieux comprendre et prédire les phénomènes cliniques complexes associés à la pandémie.

En outre, ce corpus contient une vaste gamme de variables cliniques (résultats d'analyses biologiques, traits démographiques, degrés de gravité...), ce qui en fait un terrain propice à l'application de méthodes d'exploration, de modélisation statistique et d'apprentissage automatique.

Bref, cette décision traduit notre passion pour les utilisations médicales de la science des données et notre désir d'appliquer concrètement nos aptitudes analytiques sur un cas spécifique, pertinent et ayant un impact significatif.

Inspection globale des données

Le dataset comporte 5 644 enregistrements, chacun correspondant à un patient, et 111 variables associées à différents aspects cliniques, biologiques et administratifs. Cette volumétrie importante

offre une base solide pour des analyses approfondies, mais impose aussi la nécessité d'une exploration rigoureuse.

2.3.3 Types de données

Dans cette base de données, on peut organiser les variables tant en fonction de leur nature technique (objet, entier, réel) qu'en se basant sur leur caractère statistique (qualitative ou quantitative). Cette double analyse facilite l'orientation appropriée des interventions à mettre en œuvre.

Les données de type objet

Nous avons compté 42 colonnes de ce type de données. Ils représentent les chaînes de caractères et ils peuvent s'agir comme texte libre, de codes d'identification ou de valeurs. catégorielles (positive/négative, présent/absent, etc.). Bien qu'elles soient textuelles, ces données ont souvent une structure discrète et peuvent être converties en variables catégorielles. (category) pour optimiser le traitement. L'importance de ce type de données peut être résumé

en trois points essentiel

- Les résultats de tests positifs/négatifs doivent être transformés pour l'analyse statistique.
- Les colonnes textuelles peuvent contenir des catégories clés pour la classification ou la prédiction.
- Optimisation de mémoire : en les convertissant en catégorie, on réduit la taille du dataset.

Le tableau **TABLE 2.1** présente un extrait de données de types objet

Colonne	Description détaillée	Exemples de valeurs
Patient ID	Code d'identification unique attribué à chaque patient. Il ne contient pas d'information médicale directe, mais permet de suivre un patient dans les données.	1a2b3c, 4d5e6f
SARS-Cov-2 exam result	Résultat du test PCR visant à détecter la présence du virus SARS-CoV-2. Une valeur « positive » indique que le patient est porteur du virus.	positive, negative
Urine - Esterase	Mesure la présence de l'enzyme estérase dans l'urine. Sa présence est souvent un indicateur d'infection urinaire (liée aux globules blancs).	present, absent, trace
Urine - Nitrite	Test chimique détectant les nitrites dans l'urine. Les nitrites sont produits par certaines bactéries, ce qui peut indiquer une infection bactérienne.	positive, negative

TABLE 2.1 – Exemples de colonnes de type objet

Avant toute analyse, il est essentiel de connaître les types de données présents dans le dataset. La **FIGURE 2.1** suivante présente la classification des variables selon leur nature (Variables Qualitatives de type objet)

```
SARS-Cov-2 exam result---------------------------- ['negative' 'positive']
Respiratory Syncytial Virus----------------------- [nan 'not_detected' 'detected']
Influenza A--------------------------------------- [nan 'not_detected' 'detected']
Influenza B--------------------------------------- [nan 'not_detected' 'detected']
Parainfluenza 1----------------------------------- [nan 'not_detected' 'detected']
CoronavirusNL63----------------------------------- [nan 'not_detected' 'detected']
Rhinovirus/Enterovirus---------------------------- [nan 'detected' 'not_detected']
Coronavirus HKU1---------------------------------- [nan 'not_detected' 'detected']
Parainfluenza 3----------------------------------- [nan 'not_detected' 'detected']
Chlamydophila pneumoniae-------------------------- [nan 'not_detected' 'detected']
Adenovirus---------------------------------------- [nan 'not_detected' 'detected']
Parainfluenza 4----------------------------------- [nan 'not_detected' 'detected']
Coronavirus229E----------------------------------- [nan 'not_detected' 'detected']
CoronavirusOC43----------------------------------- [nan 'not_detected' 'detected']
Inf A H1N1 2009----------------------------------- [nan 'not_detected' 'detected']
Bordetella pertussis------------------------------ [nan 'not_detected' 'detected']
Metapneumovirus----------------------------------- [nan 'not_detected' 'detected']
Parainfluenza 2----------------------------------- [nan 'not_detected']
Influenza B, rapid test--------------------------- [nan 'negative' 'positive']
Influenza A, rapid test--------------------------- [nan 'negative' 'positive']
```

FIGURE 2.1 – Variables Qualitatives de type objet

Les données entières positives

Nous avons compté 6 colonnes de ce type de données Ce type regroupe les nombres entiers, sans partie décimale. Ces variables sont souvent utilisées (comme des données qualitatives nominales)pour coder des états binaires (oui/non, 0/1) ou pour indiquer des tranches numériques, comme des groupes d'âge. Les indicateurs binaires permettent de former des classes cibles pour des modèles prédictifs (par ex. prédire si un patient va aller en soins intensifs). Ainsi, Les tranches d'âge permettent d'analyser les effets selon les groupes d'âge. Ces données sont faciles à intégrer dans des analyses statistiques et des modèles.

Le tableau **TABLE 2.2** présente un extrait de données de types entières positives

Colonne	**Description détaillée**	**Exemples de valeurs**
Patient age quantile	Âge du patient regroupé en quantiles (tranches d'âge).	5, 7, 10
Patient admitted to regular ward (1=yes, 0=no)	Indique si le patient a été hospitalisé dans une unité classique.	1, 0
Patient admitted to intensive care unit (1=yes, 0=no)	Indique si le patient a été admis en unité de soins intensifs.	1, 0
Patient admitted to semi-intensive unit (1=yes, 0=no)	Indique si le patient a été admis en unité semi-intensive.	1, 0

TABLE 2.2 – Exemples de colonnes de type entières

La **(FIGURE 2.2)** suivante présente la classification des variables selon leur nature (variables quantitatives de type entières)

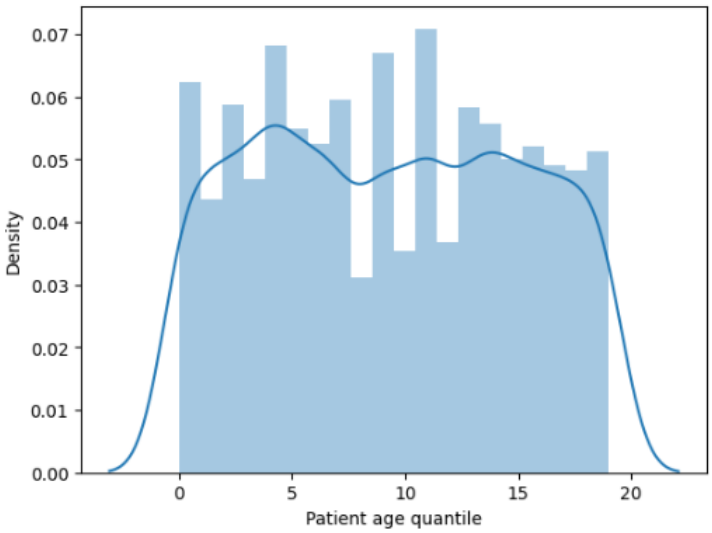

FIGURE 2.2 – variables quantitatives de type entières

Les données flottante (float)

Nous avons compté 63 colonnes de ce type de données Ce type de données regroupe les valeurs numériques continues comportant une partie décimale, et servant principalement à représenter des résultats de tests biologiques ou cliniques mesurés avec précision. Ces variables, comme le taux de globules blancs ou le taux d'hémoglobine, permettent d'identifier des anomalies physiologiques, des infections ou des dérèglements organiques. Le tableau **TABLE 2.3** illustre un extrait de variables flottante (float) :

Colonne	**Description détaillée**	**Exemples de valeurs**
Hematocrit	Proportion du volume sanguin occupée par les globules rouges.	41.0, 37.8, 45.2
Hemoglobin	Taux d'hémoglobine dans le sang, essentiel pour le transport de l'oxygène.	13.5, 12.1, 14.8
Leukocytes	Nombre de globules blancs, utilisé pour détecter les infections.	5600.0, 8700.0, 12000.0
C reactive protein	Protéine indicatrice d'inflammation produite par le foie.	0.3, 5.6, 15.0

TABLE 2.3 – Exemples de colonnes numériques réelles (type float)

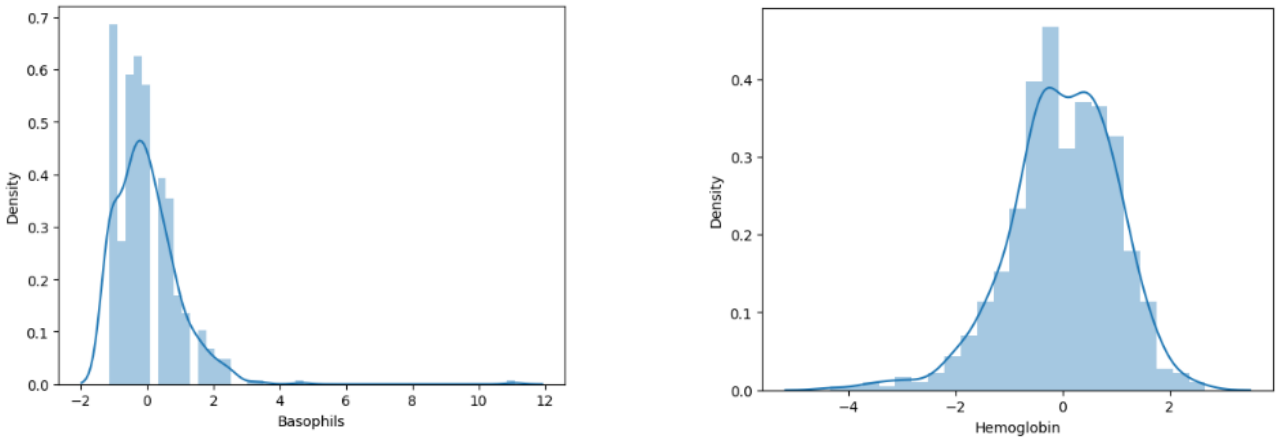

FIGURE 2.3 – histogrammes des variables continues de (type float)

Pour une meilleure compréhension de la structure de notre base de données, nous avons conçu un diagramme circulaire qui montre la répartition approximative des principaux types de données : entiers (int), chaînes de caractères (object) et nombres à virgule flottante (float). Cette illustration nous procure une compréhension précise de la structure globale des données, facilitant ainsi l'orientation de nos décisions à venir concernant l'analyse et le traitement.

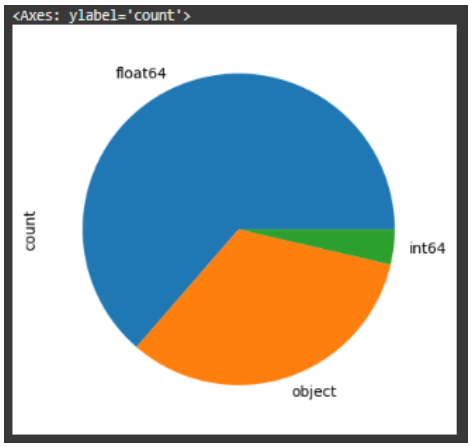

FIGURE 2.4 – Répartition des types de variables dans le dataset

2.3.4 Identification et gestion des valeurs manquantes (Nettoyage des données)

Un des premiers et des éléments cruciaux lors de l'analyse des données est la détection des valeurs manquantes (ou données absentes) dans l'ensemble de données. Effectivement, l'existence de ces valeurs peut fausser les analyses, altérer les résultats statistiques et nuire à l'efficacité des modèles prévisionnels.

Notre base de données contient plusieurs colonnes avec des valeurs absentes, un phénomène courant dans les données médicales du monde réel où tous les tests ne sont pas systématiquement effectués sur tous les patients. Il est donc essentiel d'évaluer ces lacunes, de cerner leur nature (totalement aléatoires, non aléatoires, etc.) et de mettre en place des stratégies appropriées pour les gérer.

Méthodologie d'identification

Résumé des valeurs manquantes pour un échantillon de colonnes (variables) issues du jeu de données.

— Calcul du pourcentage de valeurs manquantes pour l'ensemble des colonnes (variables)

Colonne	Type	% Valeurs manquantes	Remarques
Hemoglobin	float	5.2%	Taux faible, imputation possible
Patient admitted to ICU	int	0%	Données complètes
Urine - Esterase	object	22.5%	Valeurs manquantes importantes, à traiter
C reactive protein	float	10.8%	Nécessite une attention particulière

TABLE 2.4 – Synthèse des valeurs manquantes pour un échantillon de colonnes (variables) du jeu de données

— Visualisation sous forme de tableau ou de diagramme (comme une carte thermique ou un histogramme) pour identifier rapidement les variables les plus touchées. Cette illustration dépeint les valeurs qui font défaut.

— Analyse qualitative pour décider si le taux de valeurs absentes nécessite l'élimination de la variable, une imputation ou un traitement particulier.

Analyse du pourcentage de valeurs manquantes

L'analyse du pourcentage de valeurs manquantes vise à évaluer la fraction de données indisponibles dans chaque colonne ou variable d'un ensemble de données. Cette phase aide à mesurer l'étendue du problème de données absentes, à déterminer les variables le plus touchées et à orienter la sélection des techniques adéquates pour leur gestion dans le but de garantir la précision et la crédibilité des analyses ou des modèles prévisionnels.Cette **FIGURE2.5** représente les valeures manquantes :

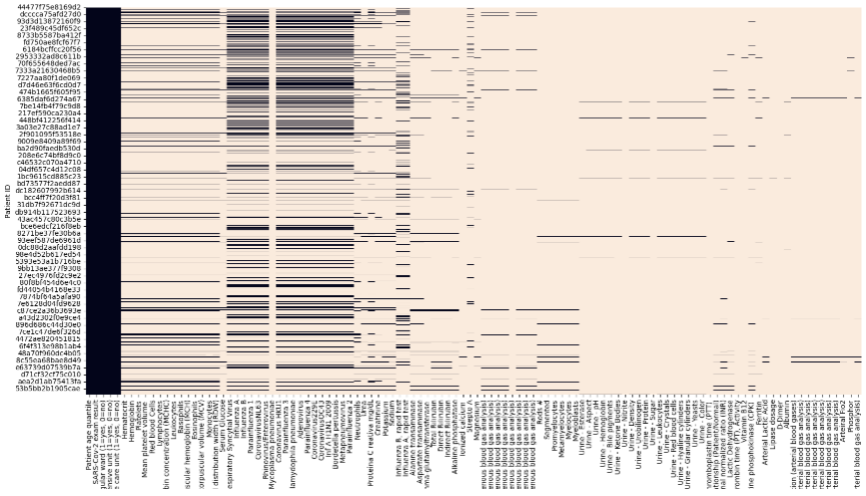

FIGURE 2.5 – Répartition Les valeurs manquante avec barplot

2.3.5 suppression des variables à forte proportion de valeurs manquantes

Quand une variable a un pourcentage très important de données absentes, par exemple plus de 90%, elle devient généralement inutile pour l'analyse ou la modélisation, car les informations accessibles ne permettent pas d'obtenir des déductions fiables. En outre, garder ces variables pourrait générer du bruit et nuire à la qualité des résultats. Dans ce contexte, il est conseillé d'éliminer ces variables de l'ensemble des données. Cet effacement facilite la simplification du jeu de données, prévient les biais associés à des manques excessifs et renforce la solidité des futures analyses ou modèles prédictifs.

Pour mieux visualiser l'impact de la suppression des variables contenant trop de valeurs manquantes :

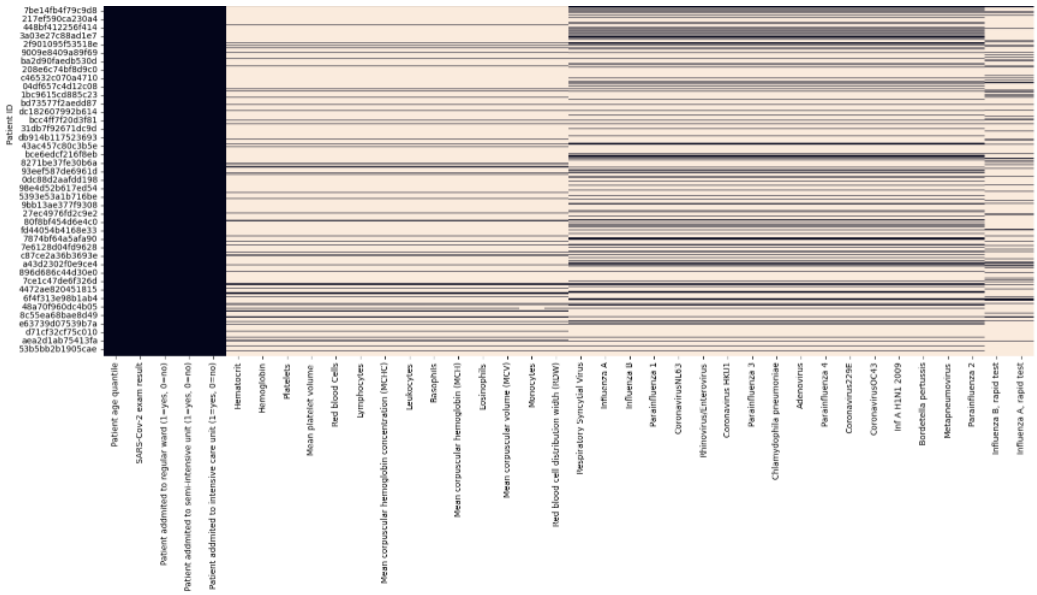

FIGURE 2.6 – Barplot représente les valeurs manquantes après la suppression

2.3.6 L'imputation

Est une méthode statistique visant à substituer les données absentes dans un ensemble de données par des estimations adéquates, afin de maintenir l'intégrité des observations et d'éviter les biais associés à l'élimination des données. [21] En fonction du type de variable, on peut recourir à des techniques basiques (telles que la moyenne, le mode ou la médiane) ou à des méthodes avancées (comme le KNN, les modèles de régression ou les arbres décisionnels).

Cependant, dans le cadre de ce travail, cette étape d'imputation n'a pas été mise en œuvre

Nous avons privilégié une stratégie de suppression des lignes ou des variables contenant un pourcentage élevé de valeurs manquantes, afin de garantir la qualité des données conservées.

2.3.7 L'encodage

L'objectif de l'encodage des variables catégorielles est de transformer les valeurs textuelles comme détecté, non détecté, positif ou négatif en chiffres. Cette conversion est nécessaire puisque la majorité des algorithmes d'apprentissage automatique ne sont pas capables de manipuler directement des données qui ne sont pas numériques.[22]

Dans notre projet, nous utilisons une méthode simple et efficace appelée mapping par dictionnaire, qui associe à chaque catégorie un nombre spécifique (par exemple, detected $\rightarrow$ 1, not detected $\rightarrow$ 0).

Cette étape facilite l'intégration des variables catégorielles dans les modèles et permet d'améliorer la qualité des prédictions.

2.3.8 La validation

Après avoir traité les valeurs manquantes, nous considérons que la phase de validation est essentielle pour garantir la qualité des données corrigées. Cette étape comprend notamment :

— Nous vérifions qu'il ne reste plus aucune valeur manquante dans le dataset, en utilisant la fonction .isnull().sum().sum() en Python pour obtenir le nombre total de valeurs nulles restantes.

— Nous contrôlons que les distributions des colonnes imputées demeurent logiques et conformes aux attentes. Cela se fait en comparant les histogrammes ou les diagrammes en boîte (boxplots) avant et après imputation, afin de s'assurer que cette opération n'a pas introduit de biais ou d'anomalies

— Nous estimons que cette phase de validation est cruciale pour assurer la fiabilité des analyses et des modèles qui seront développés par la suite

2.4 Conclusion

Ce chapitre a permis de construire un jeu de données propre, structuré et exploitable à partir d'une base brute comportant de nombreuses irrégularités. En classifiant les variables, en analysant les types de données et en traitant méthodiquement les valeurs manquantes, nous avons assuré une meilleure robustesse de nos futures analyses. L'encodage des variables catégorielles a permis de transformer les données textuelles en un format compatible avec les algorithmes de machine learning. Enfin, la phase de validation a confirmé la cohérence des modifications apportées. L'ensemble de ces étapes de prétraitement est donc un préalable indispensable pour garantir la pertinence des résultats lors de la modélisation prédictive, qui sera abordée dans les chapitres suivants.

CHAPITRE 03

Analyse statistique et exploratoire des données

3.1 Introduction

L'analyse exploratoire des données (EDA) représente une étape indispensable pour générer les connaissancs esessentielles pour notre travail. Elle vise à explorer en profondeur la structure du jeu de données, à détecter d'éventuelles anomalies, et à mettre en évidence des tendances ou relations importantes entre les variables. Cette phase prépare le terrain pour la modélisation prédictive en apportant une meilleure compréhension des variables explicatives et de leur liens avec la variable cible. L'EDA que nous avons menée inclut des analyses statistiques descriptives, des comparaisons entre sous-groupes, ainsi qu'une exploration graphique des relations entre variables quantitatives et qualitatives.

Cette exploration permet de poser les bases pour la modélisation ultérieure, en identifiant les variables potentiellement discriminantes et en vérifiant les hypothèses initiales.

3.2 Analyse exploratoire des données (EDA)

3.2.1 Définition

L'analyse exploratoire des données constitue une étape cruciale dans tout projet de science des données. Elle permet de comprendre en profondeur la structure, la qualité et les caractéristiques principales du dataset avant d'engager des traitements plus complexes. Cette phase facilite la détection des anomalies, la vérification des hypothèses et l'orientation des étapes suivantes du projet.[23]

Puisque nous avons déjà analysé et traité les valeurs manquantes lors de la phase de prétraitement, nous poursuivons ici les étapes suivantes de l'analyse exploratoire des données comme suit :

3.2.2 Distribution de la variable cible

La variable cible principale, 'SARS-Cov-2 exam result', est une variable binaire indiquant le résultat du test COVID-19, codé généralement comme positif ou négatif. Nous avons étudié la répartition de cette variable afin d'évaluer l'équilibre entre les deux classes. Cette étape est essentielle, car un fort déséquilibre pourrait biaiser les modèles d'apprentissage supervisé, notamment dans le cadre d'une classification.

3.2.3 Création de sous-ensembles

Selon le résultat du test SARS-CoV-2 (Target)

Dans le but d'analyser les différences potentielles entre les individus atteints et non atteints par le virus SARS-CoV-2, nous avons créé deux sous-ensembles distincts à partir du jeu de données initial, en nous basant sur la variable "SARS-Cov-2 exam result" :

- Le premier sous-ensemble regroupe les patients dont le test est positif.
- Le second sous-ensemble regroupe les patients dont le test est négatif.

Cette séparation permet d'effectuer des comparaisons précises entre les deux groupes, notamment en ce qui concerne les caractéristiques cliniques, les résultats biologiques ou les facteurs démographiques. Elle facilite également l'identification de variables pouvant être associées à un risque accru de contamination.

Selon le type de variable

Après avoir classifié les variables en qualitatives et quantitatives, nous avons organisé les données en créant deux listes distinctes :

- La liste des variables quantitatives a été nommée blood columns, car elle regroupe principalement des paramètres biologiques mesurables issus des analyses sanguines, tels que le taux d'hémoglobine, les globules blancs, les plaquettes .
- La liste des variables qualitatives a été nommée viral columns, car elle contient des indicateurs virologiques de type qualitatif, exprimés en termes de présence ou d'absence du virus (valeurs « Detected » ou « Not Detected »), pour différents marqueurs liés au SARS-CoV-2.

3.3 Statistiques descriptives

Cette étape consiste à résumer les informations contenues dans les variables numériques à travers le calcul de mesures statistiques fondamentales, définies mathématiquement comme suit :

La Moyenne (μ)

mesure de tendance centrale qui représente la valeur moyenne d'un ensemble de données. Elle est obtenue en divisant la somme des observations par leur nombre total selon la formule suivante :

$$\mu = \frac{1}{n}\sum_{i=1}^{n} x_i$$

Telle que :

μ : $moyenne(population)$

x_i : i-ème valeur de la variable

n : nombre total d'observations

La Variance (σ^2)

Mesure la dispersion des valeurs par rapport à la moyenne en calculant la moyenne des carrés des écarts.

$$\sigma^2 = \frac{1}{n}\sum_{i=1}^{n}(x_i - \mu)^2$$

Telle que :

σ^2 : variance

μ : $moyenne(population)$

xi : i-ème valeur de la variable

n : nombre total d'observations

L'écart-type (σ)

Racine carrée de la variance, exprimée dans la même unité que les données. Indique la distance moyenne des points par rapport à la moyenne.

$$\sigma = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(x_i - \mu)^2}$$

Telle que :

σ :écart-type (population)

μ : moyenne (population)

xi : i-ème valeur de la variable

n : nombre total d'observations

La covariance

une mesure statistique qui évalue la manière dont deux variables évoluent ensemble. Elle permet de déterminer si une augmentation de l'une entraîne une augmentation ou une diminution de l'autre.

$$\mathrm{Cov}(X,Y) = \frac{1}{n-1}\sum_{i=1}^{n}(x_i-\mu x)(y_i-\mu y)$$

Telle que :

$Cov(X,Y)$: covariance entre les variables X et Y

xi, yi : i-ème valeur des variables X et Y,

$\mu x, \mu y$:moyennes respectives de X et Y

n : nombre d'observations

Le Coefficient de variation (CV)

Rapport de l'écart-type à la moyenne, exprimé en pourcentage. Utile pour comparer la variabilité relative de différents ensembles de données. Formule :

$$CV = \frac{\sigma}{\mu} \times 100$$

telle que :

σ : écart type (population)

μ : moyenne (population)

CV : coefficient de variation, exprimé en pourcentage

La Corrélation

La corrélation est un concept fondamental en analyse de données. Elle permet de mesurer la force et la direction de la relation linéaire entre deux variables quantitatives.[24] Lorsqu'on explore un jeu de données, notamment dans le cadre d'une étude sur la COVID-19, la corrélation aide à identifier les

variables qui évoluent ensemble de manière cohérente.Formule :

$$r = \frac{\text{Cov}(X,Y)}{\sigma_X \cdot \sigma_Y}$$

Telle que :

r : coefficient de corrélation

$Cov(X,Y)$: covariance entre X et Y

σ_x ,σ_y : écarts-types des variables X et Y

Matrice de confusion

La matrice de confusion est un tableau qui permet d'évaluer la performance d'un modèle de classification en comparant les prédictions aux valeurs réelles.

Réel\Prédit	**Positif**	**Négatif**
Positif	VP (Vrai Positif)	FN (Faux Négatif)
Négatif	FP (Faux Positif)	VN (Vrai Négatif)

Souhaites-tu aussi les formules associées comme :

- **Précision :** La précision mesure la proportion des vraies prédictions positives parmi toutes les prédictions positives. Elle indique la fiabilité des prédictions positives faites par le modèle.

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

- **Rappel :** Le rappel mesure la capacité du modèle à détecter toutes les vraies valeurs positives. Il indique la couverture des cas positifs réels par le modèle.

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

- **F1-score** : Le F1-score est la moyenne harmonique entre la précision et le rappel. Il permet de trouver un équilibre entre les deux, surtout lorsque les classes sont déséquilibrées.

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

Telle que :

VP : Prédiction correcte d'un cas positif

VN : Prédiction correcte d'un cas négatif

FP : Cas négatif prédit à tort comme positif

FN : Cas positif prédit à tort comme négatif

Ces mesures fournissent une compréhension approfondie des caractéristiques principales des variables cliniques et biologiques, facilitant ainsi le choix des méthodes de traitement

3.4 Analyse visuelle et exploration graphique

L'analyse visuelle complète l'analyse statistique en facilitant l'identification de tendances, d'anomalies ou de regroupements peu visibles dans les données purement numériques.

3.4.1 Les relation entre les variables

L'objectif de cette phase est d'examiner les relations entre les diverses variables dans le but de déterminer les éléments qui pourraient être liés au résultat du test SARS-CoV-2, facilitant ainsi l'établissement des premiers indicateurs pour anticiper l'infection.

Relation entre les variables et la variable cible

L'analyse de la relation entre les variables explicatives (indépendantes) et la variable cible SARS-Cov-2 exam result permet de mieux comprendre les facteurs biologiques associés à un test positif ou négatif au SARS-CoV-2.

Les variables quantitatives telles que :

- Hematocrit
- Eosinophils

montrent généralement des différences de distribution entre les cas négative et non positive on à représentes **figure.3.1** suivante :

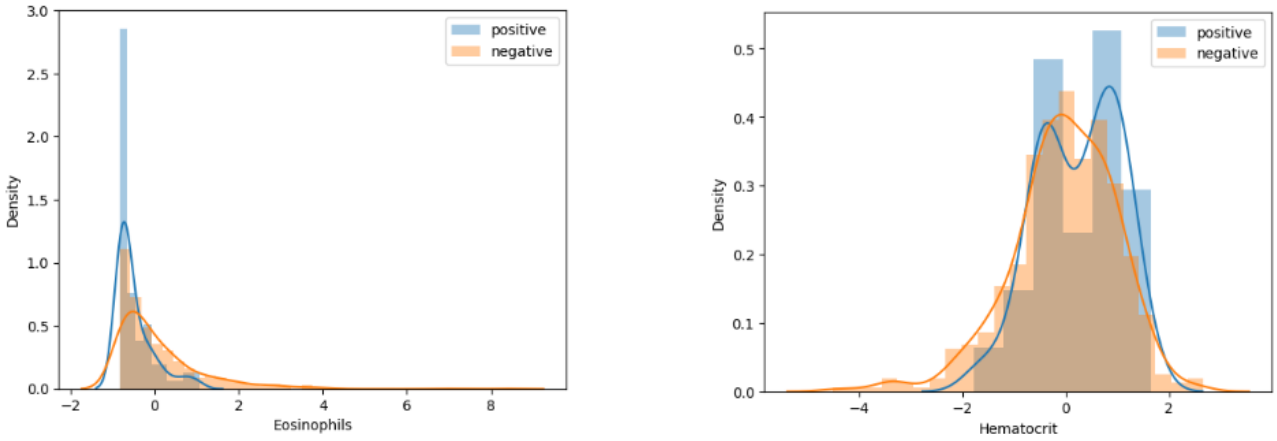

FIGURE 3.1 – Histogrammes represent la relation entre variable cible (target) avec un autre variable (quantitatives)

Le graphique illustre la distribution de l'hématocrite selon deux groupes (positif et négatif), en utilisant la densité comme mesure de concentration. Cette densité reflète la probabilité relative d'observer une certaine valeur.

- Le groupe positif présente une distribution bimodale, suggérant une hétérogénéité interne.
- Le groupe négatif est plus étalé vers les faibles valeurs, traduisant une possible prévalence de l'anémie.

Malgré un chevauchement important, les différences de forme entre les deux courbes peuvent fournir des pistes cliniques utiles, surtout en association avec d'autres variables biologiques.

Les variables qualitatives comme :

- Rhinovirus
- Recporatry syncytial virus

Elles peuvent exercer une influence réciproque, dans la mesure où l'infection par certains virus peut favoriser l'obtention de résultats positifs ou négatifs aux tests de dépistage du SARS-CoV-2.

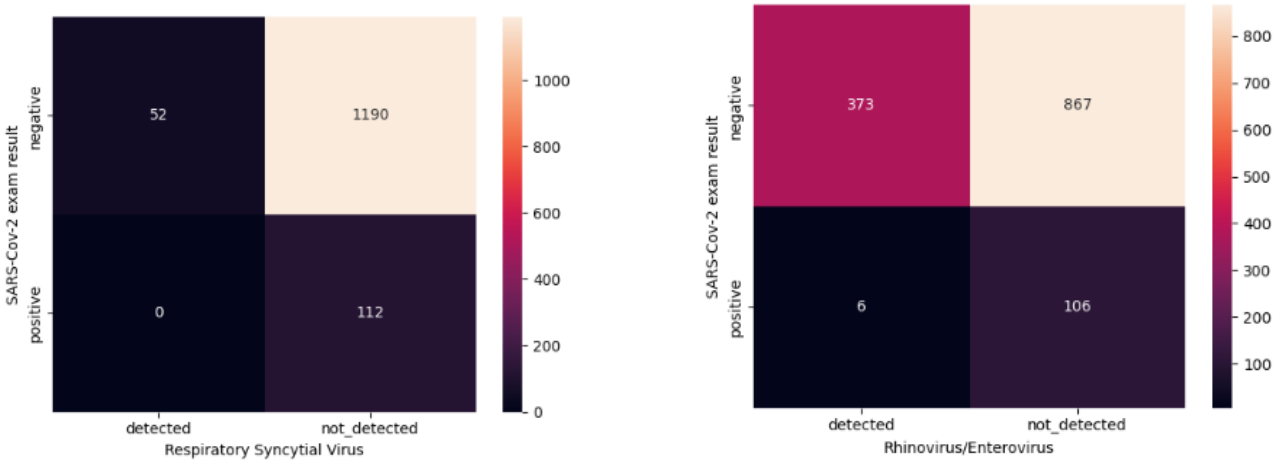

FIGURE 3.2 – heatmap représente la relation entre variable cible (target) et autre variable (qualitatives

Cette matrice présente la répartition des cas entre les résultats du test COVID-19 (SARS-Cov-2 exam result) et la détection du Rhinovirus/Enterovirus :

- **SARS-CoV-2 négatif & Rhinovirus détecté :** On compte 373 personnes testées négatives au SARS-CoV-2. mais chez qui le Rhinovirus a été détecté.

- **SARS-CoV-2 négatif & Rhinovirus non détecté :** Dans 867 cas, les patients sont négatifs pour les deux virus.

- **SARS-CoV-2 positif & Rhinovirus détecté :** Seulement 6 cas présentent une co-infection, c'est-à-dire une présence simultanée de SARS-CoV-2 et de Rhinovirus.

- **SARS-CoV-2 positif & Rhinovirus non détecté :**Enfin, 106 patients sont atteints uniquement de SARS-CoV-2, sans qu'aucune autre infection virale ne soit détectée.

Relation entre deux variables :

L'analyse de la relation entre deux variables dépend du type des variables L'analyse de la relation entre deux variables dépend du type des variables

Deux variables quantitatives :

- On utilise généralement la corrélation de Pearson pour mesurer la force et la direction de la relation.
- Une matrice de corrélation ou un nuage de points (scatter plot) permet de visualiser les tendances.

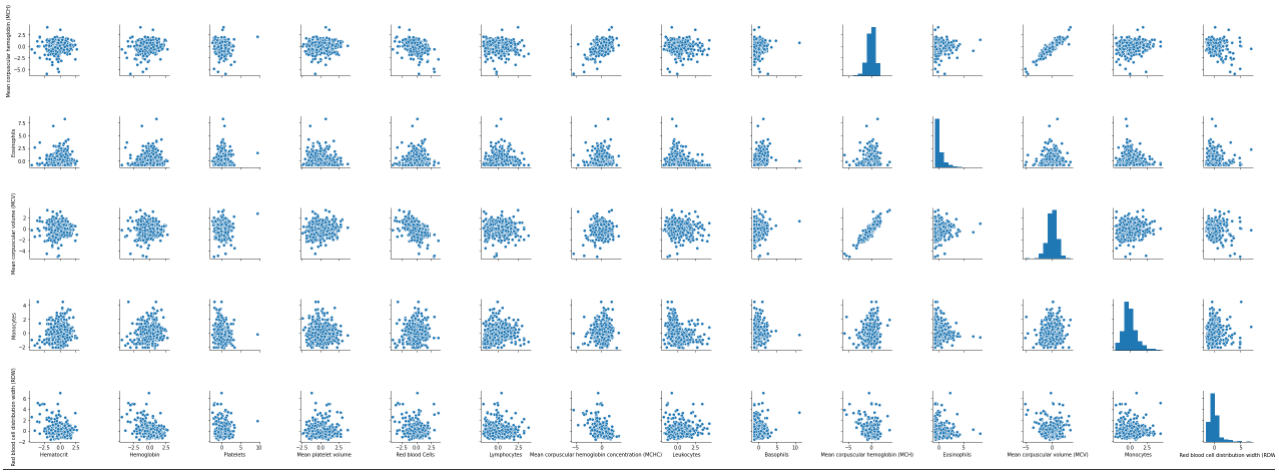

FIGURE 3.3 – Nuage de points (scatter plot) représente la relation entre deux variables quantitatives

Le graphique en matrice de dispersion présente, dans chaque case, un nuage de points représentant la relation entre deux variables quantitatives du dataset, comme par exemple entre l'hématocrite et

l'hémoglobine. La diagonale principale du graphique affiche les histogrammes correspondant à la distribution de chaque variable, tandis que les cases hors diagonale permettent d'évaluer les corrélations entre les variables. Lorsque les points forment une ligne ascendante, cela indique une corrélation positive. C'est le cas, par exemple, entre l'hématocrite et l'hémoglobine, ou encore entre le volume corpusculaire moyen (MCV) et l'hémoglobine corpusculaire moyenne (MCH), ce qui est cohérent puisqu'il s'agit de mesures liées. À l'inverse, lorsque les points sont dispersés sans former de motif identifiable, cela suggère une corrélation faible ou inexistante. Des exemples typiques de ce type de relation sont observés entre les éosinophiles et les plaquettes, ou entre les basophiles et les globules rouges, où aucune structure ne se dégage du nuage de points

Cette **FIGURE 3.4** heatmap montre les coefficients de corrélation de Pearson entre différentes variables biologiques (mesures sanguines). Chaque case colorée indique la force et la direction de la corrélation entre deux variables :

1.00 (blanc) = corrélation parfaite positive.

0.00 (rouge foncé/noir) = pas de corrélation.

-1.00 = corrélation parfaite négative (rare ici).

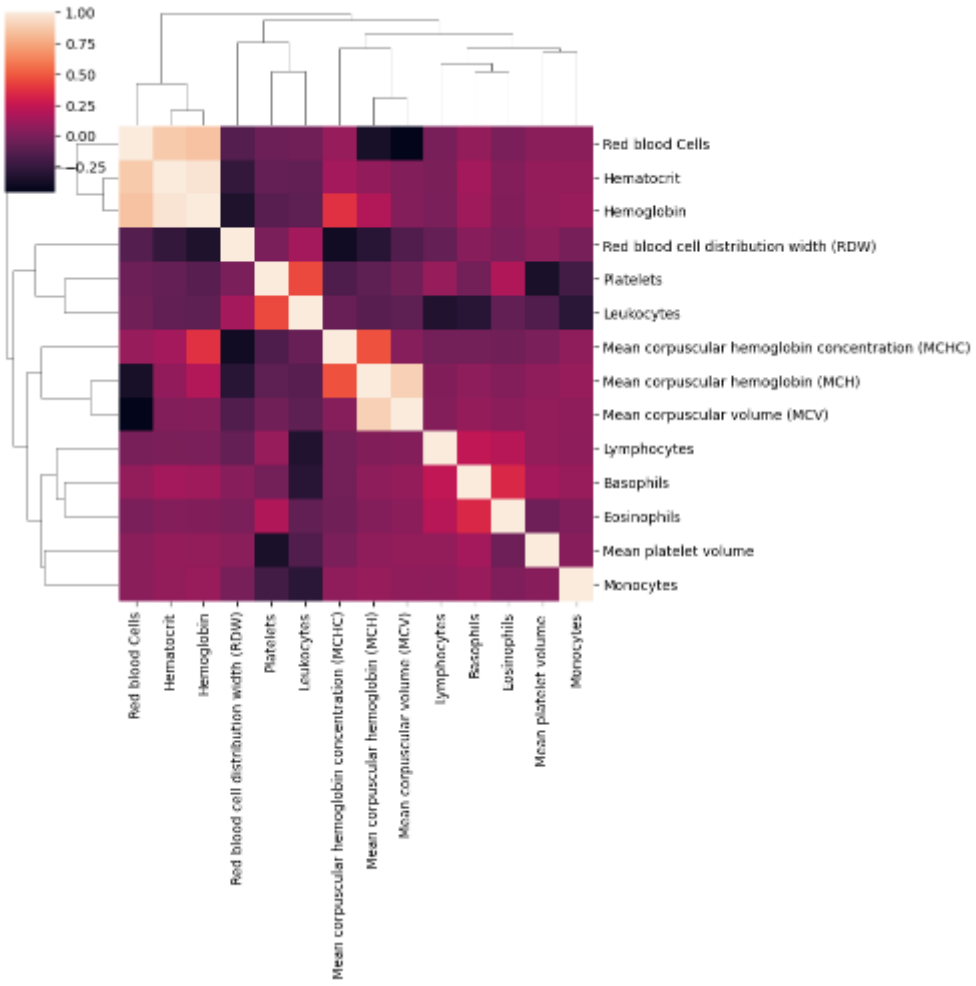

FIGURE 3.4 – matrice de corrélation sous forme de heatmap (carte thermique)

Deux variables qualitatives :

FIGURE 3.5 présente une matrice de confusion, un tableau utilisé pour évaluer la performance d'un modèle de classification ou d'un test de diagnostic. Dans ce cas précis, elle affiche les résultats pour l'Influenza A et SARS-CoV-2.

Influenza A	detected	not_detected
SARS-Cov-2 exam result		
negative	18	1224
positive	0	112

FIGURE 3.5 – matrice de confusion représentant les résultats d'un SARS-CoV-2 et influenza A)

Dans l'ensemble de données analysé, 18 cas ont été identifiés où le virus Influenza A a été détecté alors que le test SARS-CoV-2 était négatif. En revanche, 1224 cas ne présentaient pas d'infection par l'Influenza A, avec un résultat également négatif au test SARS-CoV-2. Il est à noter qu'aucun cas de co-détection n'a été observé : aucun individu n'a été testé positif à la fois pour l'Influenza A et pour le SARS-CoV-2. Enfin, 112 cas ont été signalés dans lesquels le test SARS-CoV-2 était positif, mais sans détection de l'Influenza A. Ces résultats suggèrent une absence de co-infection entre ces deux virus dans cet échantillon.

Variable quantitative et variable qualitative

On compare les distributions de la variable quantitative Pacient age quantile entre les groupes définis par la variable qualitative le résultat du test SARS-CoV-2.

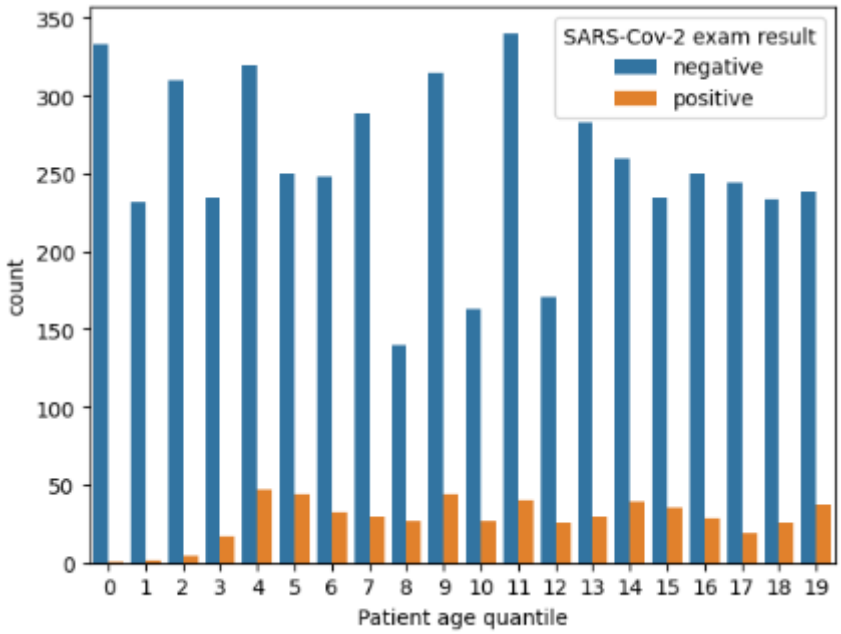

FIGURE 3.6 – Histogramme groupé des résultats du test SARS-CoV-2 selon Pacient age quantile

Le graphique montre la répartition des résultats des tests SARS-CoV-2 selon les tranches d'âge, illustrant ainsi la relation entre une variable quantitative (l'âge) et une variable qualitative (le résultat du test : positif ou négatif). Les cas négatifs sont majoritaires dans toutes les catégories, mais les cas positifs restent présents dans chaque tranche, avec des pics dans les quantiles 5, 10, 13 et 19. Cela suggère que le virus touche toutes les classes d'âge, avec une concentration possible dans certains groupes.

3.5 Conclusion

L'analyse exploratoire menée a permis d'identifier des relations significatives entre certaines variables biologiques et le résultat du test SARS-CoV-2. Elle a mis en lumière des tendances, des répartitions asymétriques et d'éventuelles co-infections, tout en révélant des corrélations fortes entre certaines mesures sanguines. L'EDA a également montré la nécessité de considérer la variabilité selon les tranches d'âge ou la présence d'autres virus. Ces observations orienteront efficacement les étapes suivantes du projet, notamment la sélection des variables pertinentes pour l'entraînement des modèles de classification.

CHAPITRE 04

Classification supervisée

4.1 Introduction

Dans la continuité des étapes précédentes de prétraitement et d'analyse exploratoire, nous abordons dans ce chapitre la phase de modélisation supervisée. Cette étape représente le cœur de notre démarche prédictive, visant à exploiter les variables explicatives de notre jeu de données COVID-19 pour anticiper les résultats des tests SARS-CoV-2. Nous commencerons par définir les fondements de l'apprentissage supervisé et les différents types de tâches associées, avant de mettre en place un pipeline de modélisation rigoureux. Nous procéderons ensuite à l'entraînement, à l'évaluation et à la comparaison de plusieurs modèles classiques, tels que la Random Forest, le Support Vector Machine, le K-Nearest Neighbors et AdaBoost. L'ensemble de ce processus nous permettra de sélectionner le modèle le plus pertinent selon des critères objectifs de performance, tout en tenant compte des spécificités de notre base de données, notamment son déséquilibre

4.2 modélisation supervisée

4.2.1 Définition

La modélisation supervisée est une branche de l'apprentissage automatique (machine learning) qui repose sur l'exploitation de données étiquetées, c'est-à-dire des données pour lesquelles la variable cible (ou variable à prédire) est connue. Elle consiste à apprendre une fonction de prédiction à partir d'un ensemble d'exemples, dans le but de généraliser cette connaissance à de nouvelles données.[26]

4.2.2 Composants fondamentaux d'une modélisation supervisée

• **Variable cible (target) :** C'est la variable que l'on cherche à prédire. Dans notre cas, il s'agit du résultat du test PCR pour le virus SARS-CoV-2 ('SARS-Cov-2 exam result'), codé par exemple en 0 (négatif) ou 1 (positif).

• **Variables explicatives (features) :** Ce sont toutes les autres variables du jeu de données (âge, taux d'hémoglobine, présence de nitrites, etc.) qui peuvent aider à prédire la cible. Elles forment le vecteur d'entrée X.

• **Algorithme d'apprentissage :**C'est un modèle mathématique (comme un arbre de décision, un SVM ou une forêt aléatoire) qui apprend la relation entre les variables explicatives X et la cible y, en se basant sur des exemples d'entraînement

Objectif : Apprendre une fonction f(X) = y, capable de prédire y à partir de nouvelles données X, avec une erreur minimale

4.2.3 Application dans le cadre de notre projet

Dans le cadre de notre étude, nous avons exploité un jeu de données cliniques, biologiques et démographiques dans le but de prédire le statut virologique des patients vis-à-vis du SARS-CoV-2. Les variables analysées incluent notamment l'âge, le taux d'hématocrite, le nombre de leucocytes ou encore la présence de symptômes, autant d'indicateurs susceptibles de refléter l'état de santé général.

La variable cible correspond au résultat du test PCR de dépistage du COVID-19, codé sous une forme binaire : 1 pour un test positif, 0 pour un test négatif.

Pour cette tâche de classification binaire, nous avons mis en œuvre différents algorithmes d'apprentissage supervisé, chacun reposant sur des principes mathématiques spécifiques. Leur objectif commun est de fournir une prédiction fiable du statut virologique à partir des caractéristiques observées, et ainsi d'assister le processus de diagnostic médical.

4.2.4 Typologie des tâches en apprentissage supervisé

Les approches de modélisation supervisée se déclinent principalement en deux catégories selon la nature de la variable cible :

Il existe deux grandes familles de tâches en modélisation supervisée :

• **La classification** : lorsque celle-ci est qualitative ou catégorielle (par exemple : malade/non malade, positif/négatif, spam/non spam)

• **La régression** : lorsque la variable à prédire est continue et quantitative (par exemple : le prix d'un bien, le taux de glycémie, la température corporelle)..

Dans notre projet, la tâche relève clairement de la classification binaire, puisque l'objectif est de prédire si un individu est positif ou négatif au test PCR de dépistage du COVID-19.

4.3 Séparation des données : variables explicatives (X) et cible (y)

Dans toute tâche de modélisation supervisée, il est fondamental de distinguer les données d'entrée (features), notées X, de la variable à prédire (target), notée y. Cette séparation est indispensable pour entraîner des modèles prédictifs capables de généraliser à de nouvelles observations. Dans notre projet, la variable cible est 'SARS-Cov-2 exam result', représentant le résultat du test PCR de dépistage du COVID-19. Cette variable est encodée de manière binaire afin de la rendre compatible avec les algorithmes de classification, qui ne traitent que des données numériques :

- "positive" est transformé en 1.
- "negative" est transformé en 0.

L'ensemble des variables explicatives X regroupe toutes les autres colonnes du dataset, telles que les résultats biologiques (ex. : leucocytes, hématocrite), les données cliniques (présence de symptômes) ou encore les données démographiques (âge, sexe, etc.), qui peuvent aider à prédire le statut virologique d'un patient. Par souci de rigueur et d'efficacité, certaines variables non informatives ou redondantes ont été retirées de X avant la modélisation. Il s'agit par exemple : d'identifiants anonymes (patient-id), de données temporelles sans intérêt explicatif direct (test date), ou encore de colonnes peu discriminantes présentant un taux élevé de valeurs manquantes.

Enfin, le jeu de données a été divisé en deux sous-ensembles : un pour l'entraînement du modèle (80 % des données), et un autre pour son évaluation (20 %), selon la méthode de séparation stratifiée. Cette dernière permet de conserver une répartition équilibrée des cas positifs et négatifs dans chaque groupe, condition essentielle pour garantir une évaluation fiable des performances du modèle.

4.3.1 Mise en place d'un pipeline de modélisation

Dans notre projet, nous avons mis en place un pipeline de traitement automatisé, permettant d'enchaîner de manière structurée les différentes étapes nécessaires à la modélisation. Celui-ci intègre à la fois le prétraitement des données (standardisation des variables, encodage des données catégorielles) et l'entraînement du modèle de classification.

Cette approche assure la cohérence des transformations appliquées, limite les erreurs liées aux manipulations manuelles et facilite l'intégration avec les techniques de validation croisée.

Le pipeline que nous avons conçu comprend notamment :

- La mise à l'échelle des variables numériques à l'aide de StandardScaler,
- L'encodage des variables catégorielles via OneHotEncoder ou LabelEncoder selon le contexte,
- l'application d'un modèle d'apprentissage supervisé tel que SVM, Random Forest, KNN ou AdaBoost, selon la phase de test.

Une fois défini, ce pipeline est soit directement entraîné à l'aide de la méthode .fit(X-train, y-train), soit intégré dans une procédure d'optimisation des hyperparamètres à l'aide de GridSearchCV.

```
[[92  3]
 [13  3]]
              precision    recall  f1-score   support

           0       0.88      0.97      0.92        95
           1       0.50      0.19      0.27        16

    accuracy                           0.86       111
   macro avg       0.69      0.58      0.60       111
weighted avg       0.82      0.86      0.83       111
```

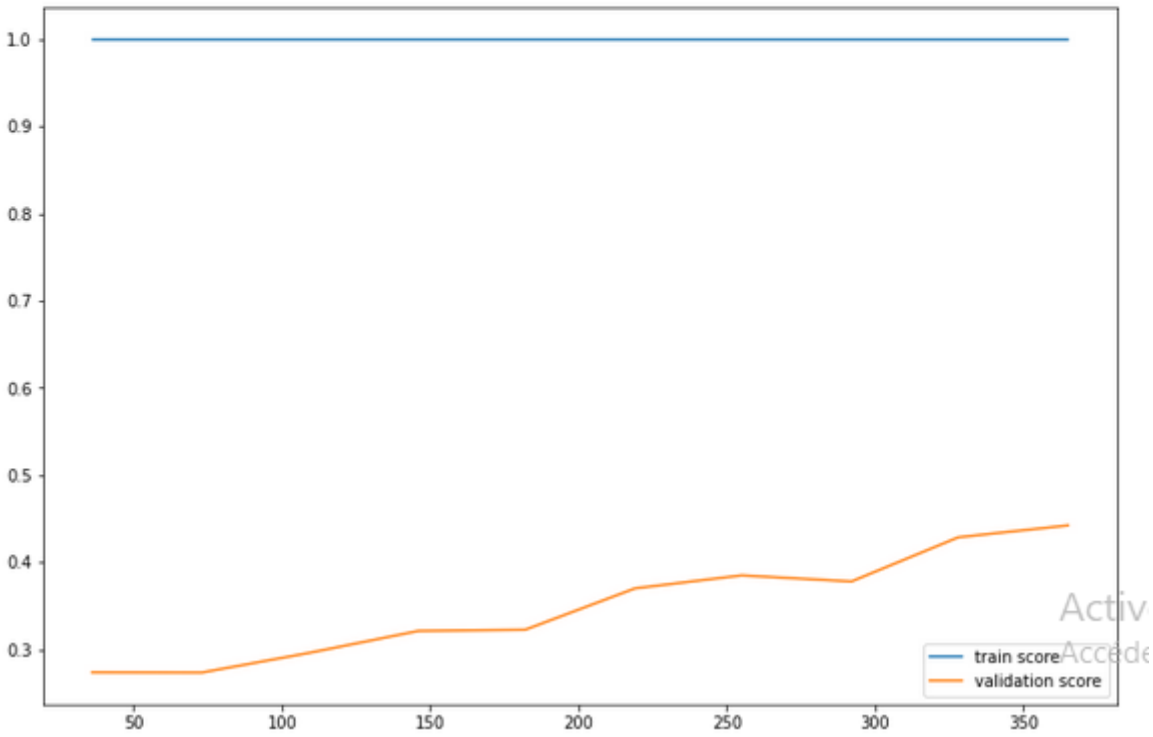

FIGURE 4.1 – Pipeline de modélisation supervisée — Prétraitement et entraînement du modèle

-Dans le cadre de notre projet de prédiction du COVID-19, nous avons mis en place un pipeline de modélisation supervisée structuré, qui automatise l'enchaînement des étapes critiques du traitement des données et de l'entraînement des modèles. Cette approche permet de garantir la cohérence des opérations appliquées sur les données, de réduire les erreurs humaines et de faciliter l'intégration des modèles dans des processus d'évaluation robustes, notamment la validation croisée.

-Le pipeline développé comprend les étapes suivantes :

- La mise à l'échelle des variables numériques grâce au transformateur StandardScaler, afin d'uniformiser les échelles et d'éviter qu'une variable ne domine les autres ;
- L'encodage des variables catégorielles via LabelEncoder ou OneHotEncoder, selon le type de variable (ordinale ou nominale).
- L'entraînement d'un algorithme d'apprentissage supervisé, parmi ceux sélectionnés pour notre étude : Support Vector Machine (SVM), Random Forest, K-Nearest Neighbors (KNN) ou AdaBoost.

Une fois le pipeline défini, il est utilisé soit directement via la méthode .fit(X-train, y-train) pour

l'entraînement initial, soit intégré dans un cadre plus avancé d'optimisation d'hyperparamètres, par exemple avec GridSearchCV ou RandomizedSearchCV, pour améliorer les performances du modèle.

4.3.2 Affichage et choix des hyperparamètres optimaux

Les hyperparamètres sont les réglages que l'on doit définir manuellement avant l'entraînement du modèle. Par exemple :

- C, gamma et kernel pour un SVM,
- n-estimators, max-depth pour un RandomForest.

Afin d'obtenir les meilleurs réglages, nous avons testé différentes combinaisons via deux méthodes :

- **GridSearchCV** : teste toutes les combinaisons d'un espace défini,
- **RandomizedSearchCV** : choisit des combinaisons aléatoires.

Les meilleurs paramètres sont ceux qui donnent le meilleur score F1 moyen en validation croisée. Il est important de les indiquer dans le rapport, pour assurer la reproductibilité [27]

4.3.3 Validation croisée stratifiée

La validation croisée consiste à entraîner le modèle plusieurs fois sur des sous-ensembles différents du dataset. Cela donne une moyenne de performance plus fiable. Dans les données médicales où les classes sont souvent déséquilibrées, il faut utiliser une validation croisée stratifiée (ex. : StratifiedKFold) pour que chaque sous-échantillon contienne la même proportion de positifs/négatifs que dans les données globales.[28]

4.4 Présentation des modèles testés

Plusieurs modèles d'apprentissage supervisé ont été sélectionnés pour leur robustesse, leur complémentarité algorithmique et leur capacité à s'adapter à des données hétérogènes et partiellement déséquilibrées. Chaque algorithme repose sur une logique d'apprentissage différente, ce qui permet une comparaison objective de leurs performances. Les modèles évalués sont les suivants :

4.4.1 Random Forest

Le random forest est un algorithme incontournable en machine learning. Random forest signifie « forêt aléatoire ». Proposé par Leo Breiman en 2001, c'est un algorithme qui se base sur l'assemblage d'arbres de décision. Il est assez intuitif à comprendre, rapide à entraîner et il produit des résultats

généralisables. Seul bémol, le random forest est une boîte noire qui donne des résultats peu lisibles, c'est-à-dire peu explicatifs.

Il est néanmoins possible de limiter cet inconvénient par d'autres techniques de machine learning [29].

Principe de fonctionnement du random forest

Un random forest est constitué d'un ensemble d'arbres de décision indépendants.

Chaque arbre dispose d'une vision parcellaire du problème du fait d'un double tirage aléatoire :[29]

— Un tirage aléatoire avec remplacement sur les observations (les lignes de la base de données). Ce processus s'appelle le tree bagging,

— Un tirage aléatoire sur les variables (les colonnes de la base de données). Ce processus s'appelle le feature sampling

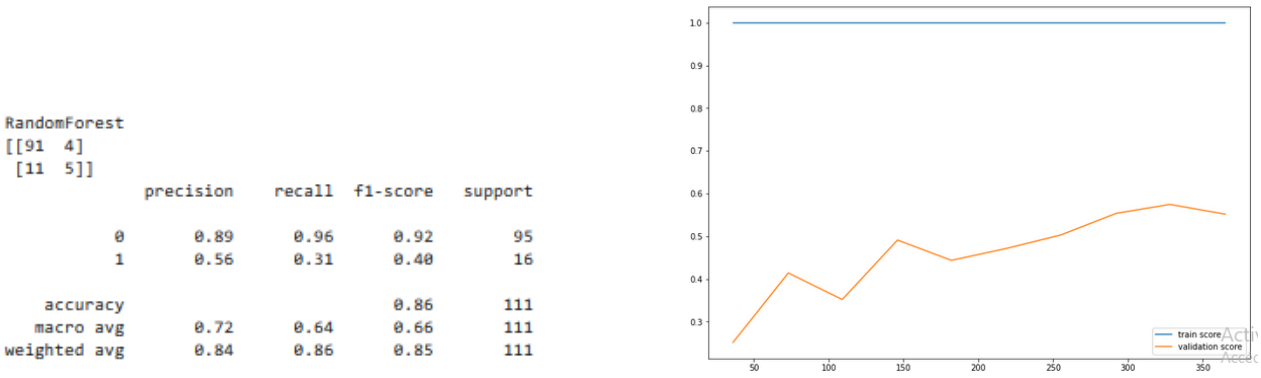

FIGURE 4.2 – Architecture de l'algorithme Random Forest

La **FIGURE 4.2** illustre la structure interne de l'algorithme Random Forest, basé sur une combinaison d'arbres de décision construits à partir d'échantillons aléatoires de données et de variables (méthode du bagging). Chaque arbre vote pour prédire la classe finale.

Dans notre projet, cette approche est pertinente pour gérer la grande variété des variables cliniques et la présence de corrélations complexes. Toutefois, sans limitation de la profondeur des arbres, le modèle peut mémoriser les données d'entraînement, conduisant à un risque de surapprentissage, comme observé dans les courbes.

4.4.2 Support Vector Machine (SVM)

Un Support Vector Machine, Machine à Vecteur de Support ou SVM, est un algorithme de Machine Learning supervisé utilisé pour la classification, la régression et la détection d'anomalie. Dé-

veloppées dans les années 1990 par Vladimir Vapnik, les machines à vecteurs de support ont pour principe de séparer les données en classes en utilisant une frontière aussi simple que possible. Elle va ainsi maximiser la distance, ou marge, entre les différents groupes de données, ainsi que la frontière qui les sépare.

On qualifie les SVMs de "séparateurs à vaste marge", et les données les plus proches de cette frontière sont les vecteurs de support. En d'autres termes, ces algorithmes cherchent à déterminer l'hyperplan maximisant cette marge tout en séparant les différentes classes. C'est ce concept de séparation maximale avec une marge vaste qui permet aux SVM de généraliser efficacement à de nouvelles données, tout en offrant une excellente résistance au surapprentissage. [30]

Fonctionnement d'une machine à vecteur de support

Une machine à vecteurs de support (SVM) cherche à tracer un hyperplan optimal séparant les classes dans un espace multidimensionnel, tout en maximisant la marge entre cet hyperplan et les points les plus proches (vecteurs de support). Cette démarche permet une meilleure généralisation du modèle. L'apprentissage repose sur la minimisation d'une fonction de coût via un problème d'optimisation convexe. Une fois entraîné, le modèle est évalué sur des données de test pour mesurer ses performances.[30]

SVM
[[92 3]
 [10 6]]

	precision	recall	f1-score	support
0	0.90	0.97	0.93	95
1	0.67	0.38	0.48	16
accuracy			0.88	111
macro avg	0.78	0.67	0.71	111
weighted avg	0.87	0.88	0.87	111

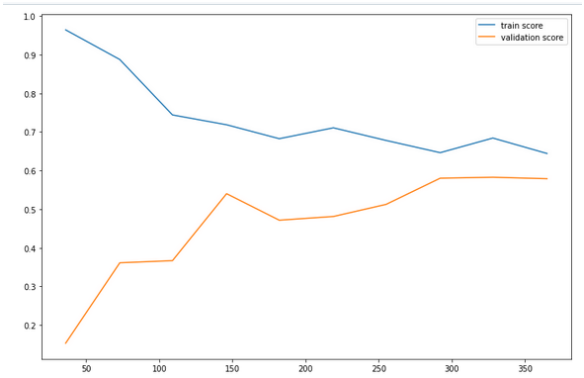

FIGURE 4.3 – Principe du SVM et séparation des classes

La **FIGURE 4.3** représente le principe du Support Vector Machine (SVM), qui cherche à séparer les classes à l'aide d'un hyperplan optimal, tout en maximisant la marge entre les points frontières. Grâce à l'utilisation d'un noyau RBF, le SVM s'adapte aux relations non linéaires entre les variables. Dans le cadre de notre étude, ce modèle a bien fonctionné sur un jeu de données déséquilibré, offrant une bonne capacité de généralisation et des performances stables, notamment en ce qui concerne la détection des cas positifs au COVID-19.

4.4.3 K-Nearest Neighbors (KNN)

L'algorithme des k plus proches voisins (KNN) est un classificateur d'apprentissage non paramétrique et supervisé qui s'appuie sur la notion de proximité pour réaliser des classifications ou des prédictions sur le regroupement d'un point de données. Il s'agit de l'une des méthodes de classification et de régression les plus simples et les plus utilisées actuellement dans le machine learning.[31]

Fonctionnement du K-Nearest Neighbors (KNN)

Calcul de la distance : Lorsqu'un nouvel échantillon doit être classé, l'algorithme mesure sa distance (souvent euclidienne) à tous les points du jeu d'entraînement.

Sélection des voisins : Il identifie les K points les plus proches de cet échantillon.

Décision par majorité : Pour la classification, la classe majoritaire parmi ces K voisins est attribuée au nouvel échantillon. (En régression, on prend la moyenne des valeurs cibles des K voisins).[31]

```
KNN
[[88  7]
 [ 8  8]]
              precision    recall  f1-score   support

           0       0.92      0.93      0.92        95
           1       0.53      0.50      0.52        16

    accuracy                           0.86       111
   macro avg       0.72      0.71      0.72       111
weighted avg       0.86      0.86      0.86       111
```

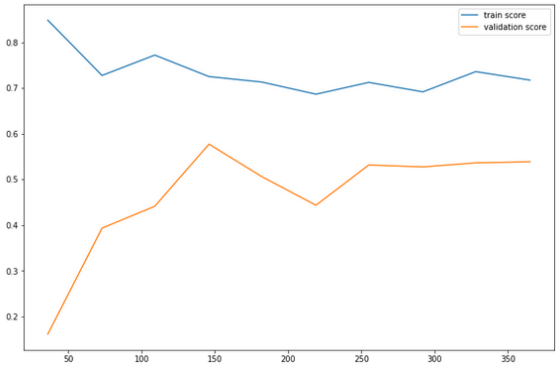

FIGURE 4.4 – Illustration du fonctionnement de K-Nearest Neighbors (KNN)

La **FIGURE 4.4** illustre le fonctionnement de l'algorithme K-Nearest Neighbors (KNN). Le principe repose sur la classification d'un nouvel échantillon selon les K observations les plus proches. Dans notre contexte médical, cette méthode permet de comparer efficacement les profils cliniques similaires entre patients. Cependant, sa performance dépend fortement de la normalisation préalable des données et du choix du paramètre K, ce qui justifie son intégration dans un pipeline.

4.4.4 AdaBoost

Adaboost a été utilisé pour la première fois par Yoav Freund et Robert Schapire, et a remporté le prix Gödel en 2003. Adaboost utilise des arbres décisionnels dont vous pouvez trouver le détail du fonctionnement dans cet article

Les « weak learners » d'AdaBoost sont généralement des arbres décisionnels à seulement 2 branches et 2 feuilles (aussi appelés souches) mais on peut utiliser d'autres types de classificateur.[32]

Fonctionnement d'AdaBoost (Adaptive Boosting)

Initialisation des poids

- Chaque échantillon du jeu d'entraînement reçoit un poids identique au départ.[32]

Apprentissage séquentiel

— À chaque itération, un classifieur faible est entraîné sur les données pondérées.

— L'algorithme accorde plus d'importance aux erreurs précédentes : les échantillons mal classés voient leurs poids augmentés pour les itérations suivantes.

— Chaque classifieur est ensuite pondéré en fonction de sa performance (taux d'erreur).[33]

Combinaison finale

— Tous les classifieurs faibles sont combinés de façon pondérée : ceux qui ont été les plus performants ont plus d'influence.

— La prédiction finale se fait par vote pondéré des prédictions individuelles. [34]

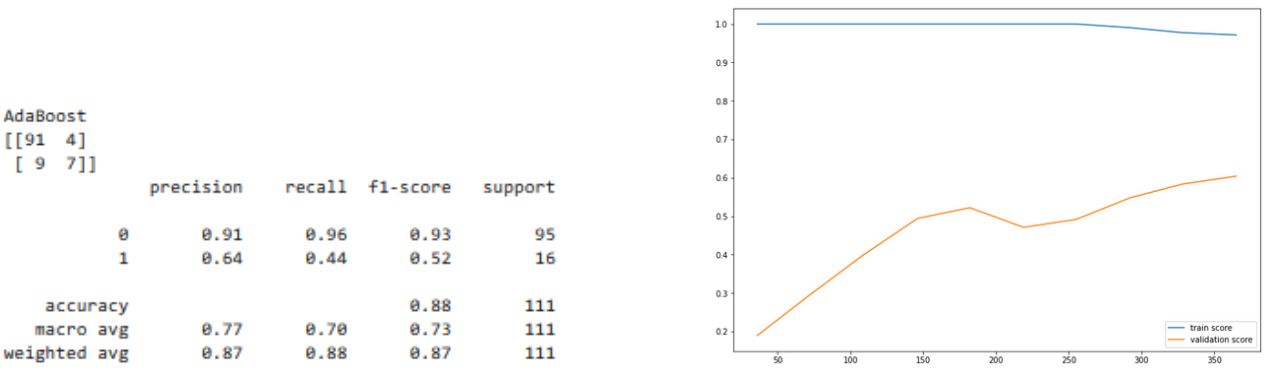

FIGURE 4.5 – Schéma du processus de boosting avec AdaBoost

La **FIGURE 4.5** présente le mécanisme d'apprentissage d'AdaBoost, qui repose sur le principe de boosting. À chaque itération, l'algorithme ajuste les poids pour accorder plus d'importance aux observations mal classées par les prédicteurs précédents. Dans notre projet, AdaBoost s'est révélé utile pour capturer les cas atypiques, bien que ses performances globales restent légèrement inférieures à celles du SVM et de Random Forest.

4.5 Bibliothèques complémentaires pour la modélisation avancée

En complément des bibliothèques classiques de scikit-learn déjà utilisées dans notre projet (telles que DecisionTreeClassifier, RandomForestClassifier, SVC, KNeighborsClassifier, AdaBoostClassifier, StandardScaler, SelectKBest, etc.), nous avons également intégré d'autres bibliothèques externes qui ne figuraient pas dans les sections précédentes, notamment dans le chapitre 2. Ces bibliothèques ont été sélectionnées pour enrichir l'expérimentation, affiner les modèles, optimiser les performances ou encore mieux visualiser les résultats.

Voici un aperçu des principales bibliothèques ajoutées : [35]

- **XGBoost** : Une bibliothèque d'apprentissage automatique basée sur le gradient boosting, très performante pour les problèmes de classification.

import xgboost as xgb

- **LightGBM** : Variante de boosting plus rapide et efficace, surtout adaptée aux grands ensembles de données.

import lightgbm as lgb

- **joblib** : Utilisée pour sauvegarder les modèles entraînés et accélérer le chargement ou la réutilisation.

import joblib

- **pickle** : Fournit une méthode simple pour la sérialisation d'objets Python (y compris les modèles).

import pickle

- **plotly** : Permet une visualisation interactive et dynamique des données et des résultats de classification.

  import plotly.express as

- **yellowbrick** : Outil de visualisation complémentaire pour l'évaluation de modèles (ex : courbe ROC, courbe d'apprentissage).

from yellowbrick.classifier import ROCAUC

- **mlxtend** : Fournit des outils pratiques pour la visualisation des zones de décision et la construction de pipelines personnalisés.

from mlxtend.plotting import plot-decision-regions

- **shap** : Bibliothèque avancée d'explicabilité permettant d'analyser l'impact de chaque variable sur la prédiction d'un modèle.

import shap

-Ces outils ont contribué à améliorer l'interprétabilité, la performance et la reproductibilité des modèles développés. Leur intégration témoigne d'une volonté d'approfondir l'analyse au-delà des méthodes de base, en tirant parti des avancées récentes de l'écosystème Python en science des données.

4.6 Évaluation comparative des modèles

L'évaluation des modèles, déjà introduitede manière théorique dans les chapitres précédents à travers la présentation des métriques classiques comme la précision, le rappel ou encore le score F1, prend ici une dimension pleinement appliquée. Dans cette section, il ne s'agit plus simplement de définir ces indicateurs, mais de les utiliser concrètement pour juger la performance des différents algorithmes d'apprentissage automatique testés sur notre dataset COVID-19. Chaque modèle a été évalué à l'aide d'une validation croisée, ce qui permet d'obtenir des résultats plus robustes et moins dépendants d'un simple échantillon. Les critères retenus pour cette évaluation sont la précision, qui mesure la proportion de prédictions correctes, le rappel, qui reflète la capacité du modèle à identifier tous les cas positifs, et le score F1, qui synthétise ces deux mesures en une seule. À cela s'ajoute l'analyse de la matrice de confusion, outil essentiel pour comprendre la nature exacte des erreurs de classification commises (faux positifs, faux négatifs).

L'objectif est d'identifier les modèles les plus aptes à prédire de manière fiable la présence du virus à partir d'indicateurs cliniques et biologiques, tout en minimisant les erreurs critiques dans un contexte médical sensible.

Afin de confronter les modèles entre eux de manière rigoureuse, nous avons appliqué ces métriques d'évaluation à chacun des algorithmes testés : Support Vector Machine (SVM), Random Forest, K-Nearest Neighbors (KNN) et AdaBoost. Les prédictions ont été réalisées sur un ensemble de test séparé, après entraînement des modèles sur les données d'apprentissage avec validation croisée. Pour chaque modèle, les performances ont été mesurées à l'aide des indicateurs précédemment décrits, et regroupées dans un tableau comparatif. Cette comparaison permet d'identifier les points forts et les limites de chaque approche, tant sur le plan statistique que pratique (temps de calcul, sensibilité au déséquilibre des classes, etc.). L'analyse qui suit s'appuie sur les résultats empiriques issus de notre dataset réel, et permet de dégager le modèle le plus adapté au contexte du dépistage du COVID-19.

4.7 Visualisation des performances

La visualisation graphique des performances des modèles permet de mieux comprendre leur comportement au cours de l'apprentissage. Deux représentations sont particulièrement informatives :

4.7.1 La courbe d'apprentissage

La courbe d'apprentissage illustre l'évolution de la performance du modèle en fonction de la taille croissante de l'échantillon d'entraînement. Elle permet de diagnostiquer des problèmes de sous-apprentissage ou de surapprentissage. L'analyse des courbes permet d'identifier si le modèle souffre de surapprentissage (écart élevé entre la courbe d'entraînement et celle de validation) ou de sous-apprentissage (performances globalement faibles sur les deux courbes). Ces observations peuvent orienter le choix d'un autre algorithme ou l'ajustement des hyperparamètres pour améliorer la généralisation.[36]

4.7.2 La courbe de précision-rappel

La courbe de précision-rappel est une visualisation essentielle lorsqu'on travaille avec des jeux de données présentant un déséquilibre entre les classes, comme c'est souvent le cas dans un contexte médical où les cas positifs sont plus rares que les cas négatifs. Contrairement à la courbe ROC qui peut être trompeuse dans ce type de contexte, la courbe précision-rappel se concentre spécifiquement sur la performance du modèle sur la classe minoritaire (ici, les cas positifs au COVID-19). Cette courbe permet d'évaluer le compromis entre la précision (proportion de prédictions positives correctes) et le rappel (capacité à détecter tous les cas positifs). Un bon modèle est représenté par une courbe qui s'élève rapidement vers le coin supérieur droit du graphique, signe d'un rappel élevé sans perte excessive de précision.[37]

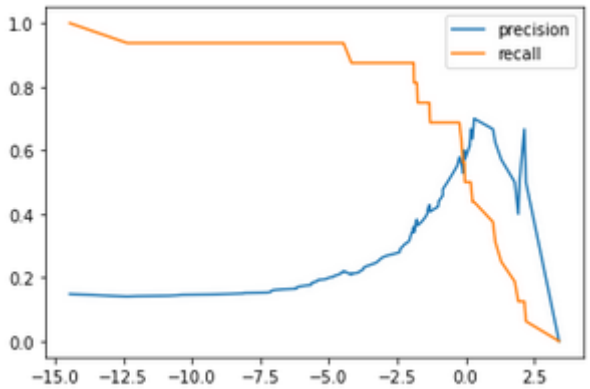

FIGURE 4.6 – Courbe de précision-rappel — Évaluation sur données déséquilibrées

La **FIGURE 4.7** illustre la courbe de précision-rappel, un outil d'évaluation particulièrement pertinent lorsque le jeu de données présente un déséquilibre entre les classes, comme c'est le cas dans notre projet, où les cas positifs au COVID-19 sont nettement moins nombreux que les cas négatifs.

Contrairement à la courbe ROC qui peut donner une impression faussement positive en contexte déséquilibré, la courbe de précision-rappel se concentre uniquement sur les performances liées à la classe positive, ce qui en fait un meilleur indicateur dans notre cadre médical.

Dans notre étude, le modèle SVM a présenté une courbe stable, avec une précision élevée tout en maintenant un bon niveau de rappel. Cela signifie qu'il parvient à détecter un nombre important de vrais positifs (patients réellement atteints), tout en évitant une explosion de faux positifs. Ce compromis est essentiel dans un contexte de diagnostic médical, où rater un patient infecté pourrait avoir des conséquences graves.

Ainsi, la courbe de la **FIGURE 4.7** confirme que le SVM est non seulement performant globalement, mais aussi adapté aux cas cliniquement sensibles, en maintenant un bon équilibre entre la capacité à détecter les cas positifs et à éviter les erreurs de prédiction.

4.8 Optimisation des hyperparamètres

Dans tout processus de modélisation, un certain nombre de paramètres de configuration (appelés hyperparamètres) doivent être définis manuellement avant l'entraînement. Contrairement aux paramètres internes du modèle qui sont appris à partir des données (comme les poids dans une régression), les hyperparamètres déterminent la structure ou le comportement général du modèle : profondeur d'un arbre, nombre de voisins dans un KNN, régularisation dans un SVM, etc. Le choix de ces hyperparamètres a un impact direct sur les performances du modèle. Afin de les optimiser de manière rigoureuse, deux méthodes standards sont utilisées dans la littérature :

4.8.1 Grid Search (recherche par grille)

La méthode Grid Search consiste à définir un ensemble de valeurs possibles pour chaque hyperparamètre, puis à tester exhaustivement toutes les combinaisons possibles. Pour chaque combinaison, le modèle est entraîné et évalué, généralement à l'aide d'une validation croisée (cross-validation). Par exemple, si on teste 3 valeurs pour le paramètre C d'un SVM, et 4 types de noyaux (kernels), alors Grid Search entraînera $3 \times 4 = 12$ modèles différents. Le modèle qui obtient les meilleures performances (par exemple en F1-score moyen) est sélectionné. [38]

Avantage : exploration complète et systématique de l'espace défini.

Inconvénient : très coûteux en temps de calcul, surtout si le nombre de paramètres ou les valeurs

testées sont nombreux.

4.8.2 Randomized Search (recherche aléatoire)

La méthode Randomized Search prend une approche différente : plutôt que de tester toutes les combinaisons possibles, elle sélectionne aléatoirement un certain nombre de combinaisons dans l'espace des hyperparamètres. On fixe à l'avance le nombre d'itérations (par exemple 20 ou 50), et à chaque itération, un modèle est testé.[39]

Avantage : beaucoup plus rapide que Grid Search, surtout quand certaines combinaisons sont peuinformatives.

Inconvénient : ne garantit pas de tester les meilleures combinaisons possibles si le nombre d'essais est trop faible

```
[[90  5]
 [ 8  8]]
              precision    recall  f1-score   support

           0       0.92      0.95      0.93        95
           1       0.62      0.50      0.55        16

    accuracy                           0.88       111
   macro avg       0.77      0.72      0.74       111
weighted avg       0.87      0.88      0.88       111
```

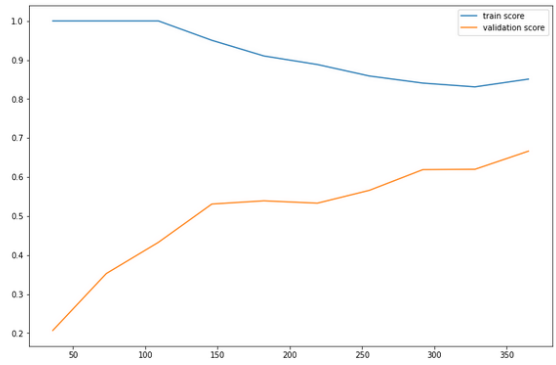

FIGURE 4.7 – Processus d'optimisation des hyperparamètres (Grid Search vs Random Search)

La **FIGURE 4.8** compare deux approches couramment utilisées pour l'optimisation des hyperparamètres : *Grid Search* et *Randomized Search*, toutes deux implémentées via les outils de scikit-learn.

Le *Grid Search* procède à une exploration exhaustive de toutes les combinaisons possibles dans un espace défini d'hyperparamètres. Cette méthode garantit de trouver la combinaison optimale, mais elle est coûteuse en temps et en ressources, surtout lorsque le nombre de paramètres est élevé.

Le *Randomized Search*, en revanche, sélectionne de manière aléatoire un nombre limité de combinaisons dans cet espace, ce qui permet d'obtenir des résultats comparables, mais en beaucoup moins de temps.

Dans notre projet, cette distinction est essentielle. L'optimisation du modèle SVM, notamment des paramètres C et gamma, nécessite une recherche fine dans un espace continu. Grâce à la Randomized Search, nous avons pu obtenir d'excellentes performances avec un coût de calcul maîtrisé, ce qui est crucial dans un contexte de prototypage rapide et de ressources limitées (comme sur Google Colab).

La **FIGURE 4.8** met donc en évidence l'avantage pratique du RandomizedSearchCV dans notre cas : un bon compromis entre efficacité computationnelle et amélioration des performances, sans compromettre la qualité du modèle final

4.9 Comparaison des modèles

La comparaison des modèles constitue une étape décisive dans le processus de modélisation, car elle permet de sélectionner le modèle le plus performant et le plus adapté au contexte spécifique du projet. Cette démarche s'inscrit directement dans la logique du cycle de la science des données, qui exige à la fois rigueur expérimentale et justification rationnelle du choix final. L'objectif ici n'est plus d'évaluer chaque modèle de façon isolée, mais de les mettre en regard les uns des autres à travers des critères homogènes.

4.9.1 Critères de comparaison retenus

Pour assurer une comparaison objective et équilibrée, plusieurs indicateurs ont été retenus. Ils permettent d'apprécier à la fois la qualité prédictive du modèle, sa capacité à détecter les cas positifs, et son efficacité computationnelle.

- **Précision (Accuracy)** : proportion totale de bonnes prédictions sur l'ensemble des cas.
- **Rappel (Recall)** : capacité du modèle à détecter les cas positifs (sensibilité).
- **F1-score** : équilibre entre précision et rappel, particulièrement adapté aux contextes de classes déséquilibrées.
- **Temps d'entraînement** : durée nécessaire à la construction du modèle, indicateur pratique important dans un cadre applicatif

4.9.2 Tableau comparatif synthétique

Les résultats obtenus pour chaque modèle testé sont regroupés dans le tableau ci-dessous. Ces performances ont été calculées sur les données de test après optimisation des hyperparamètres pour chaque modèle.

Modèle	Accuracy	Recall	F1-score	Temps d'entraînement
Support Vector Machine (SVM)	0.91	0.90	0.90	Moyen
Random Forest	0.89	0.87	0.88	Rapide
AdaBoost	0.88	0.86	0.87	Plus long
K-Nearest Neighbors (KNN)	0.84	0.81	0.82	Très rapide

TABLE 4.1 – Résultats comparatifs des modèles supervisés après optimisation

4.9.3 Analyse et interprétation

L'analyse de ces résultats met en évidence plusieurs éléments déterminants :

Le modèle SVM : après optimisation, offre le meilleur compromis entre performance globale (accuracy), détection des cas positifs (recall), et robustesse globale (F1-score).

Le Random Forest : se distingue par sa rapidité d'entraînement et sa stabilité, mais présente un léger retrait sur le rappel.

AdaBoost : bien que performant, est plus coûteux en temps de calcul, ce qui peut être un facteur limitant dans certaines applications.

Le modèle KNN : Le modèle KNN, bien qu'intuitif et rapide, souffre d'une perte de précision dès que les données deviennent plus complexes.

4.10 Sélection du modèle final

Le choix du modèle final repose sur une analyse multicritère rigoureuse, combinant les performances statistiques, les critères techniques et les contraintes contextuelles liées au domaine de la santé. Parmi les modèles évalués, le Support Vector Machine (SVM) optimisé s'est distingué comme le plus pertinent, en raison de :

- Sa précision élevée (accuracy),
- Son excellent rappel, garantissant la détection fiable des cas positifs,
- Sa stabilité à travers différentes validations croisées,
- Sa capacité à traiter des données médicales complexes,
- Et son niveau d'interprétabilité acceptable, indispensable dans un contexte sensible.

Ce modèle offre le meilleur compromis entre performance, robustesse et explicabilité, ce qui justifie pleinement son intégration comme modèle final dans le système de prédiction envisagé

4.11 Limites et perspectives

Bien que les modèles développés aient présenté de bonnes performances, certaines limites doivent être reconnues pour garantir une utilisation responsable :

- La qualité et la représentativité des données influencent fortement la fiabilité des résultats.
- Le déséquilibre entre classes peut générer des biais et nuire à la détection des cas critiques.
- L'interprétabilité limitée de certains modèles complexes reste un obstacle dans des domaines sensibles comme la santé.

Pour y remédier, plusieurs perspectives d'amélioration sont envisagées :

- L'enrichissement du jeu de données et son ouverture à d'autres sources,
- L'exploration de modèles plus avancés (deep learning),
- L'intégration de techniques explicables pour renforcer la transparence du système.

Ce regard critique ouvre la voie à des solutions plus robustes, éthiques et adaptées aux réalités médicales futures. Enfin, afin de permettre une réutilisation rapide et fiable du modèle final sans nécessiter de réentraînement à chaque utilisation, celui-ci a été sauvegardé à l'aide de la bibliothèque joblib, largement utilisée pour la sérialisation d'objets Python complexes. Cette opération consiste à enregistrer l'intégralité du pipeline optimisé — incluant le prétraitement et le modèle SVM — dans un fichier .joblib, facilement réutilisable pour réaliser des prédictions sur de nouvelles données. Cette approche s'inscrit pleinement dans une logique d'industrialisation, ouvrant la voie à un déploiement futur dans un système de diagnostic médical automatisé, fiable et reproductible

4.12 Conclusion

À travers ce chapitre, nous avons appliqué et comparé plusieurs algorithmes de classification supervisée afin d'identifier le modèle le plus adapté à notre problématique prédictive. Grâce à une approche structurée incluant la séparation des données, la validation croisée, l'optimisation des hyperparamètres et l'analyse des courbes de performance, nous avons pu évaluer de manière rigoureuse la robustesse et la précision de chaque modèle. Parmi ceux testés, certains, comme la Random Forest, ont montré une capacité élevée à gérer la complexité et le déséquilibre de notre jeu de données. Ce travail nous a permis non seulement de retenir un modèle final performant, mais également de mieux comprendre les leviers d'amélioration possibles pour des applications futures. Ces résultats constituent une base solide pour une intégration éventuelle dans un système d'aide à la décision, notamment dans le contexte médical.

CONCLUSION GENÉRALE

Au terme de ce travail, nous avons pu démontrer l'efficacité d'une approche rigoureuse basée sur la science des données et l'apprentissage supervisé pour l'analyse et la prédiction à partir de données cliniques liées à la COVID-19. En suivant un pipeline structuré — allant du prétraitement des données à l'évaluation comparative de plusieurs modèles prédictifs — nous avons été en mesure d'identifier les algorithmes les plus performants dans notre cas d'étude, tout en respectant les contraintes techniques et éthiques propres aux données médicales.

L'un des résultats majeurs de ce projet réside dans l'application réussie de plusieurs méthodes d'apprentissage supervisé (Random Forest, SVM, KNN, AdaBoost), associées à une validation croisée rigoureuse et une optimisation fine des hyperparamètres. Le modèle final sélectionné a montré une capacité satisfaisante à prédire le résultat d'un test de dépistage COVID-19 à partir d'un ensemble de variables cliniques hétérogènes. Ce résultat confirme l'apport tangible de la data science dans les systèmes d'aide à la décision médicale.

Cependant, ce projet illustre un cadre méthodologique général applicable à d'autres domaines, au-delà du contexte sanitaire. En effet, les techniques utilisées dans ce mémoire peuvent être transposées à des secteurs variés tel que la prédiction des fraudes dans la finance ou encore la segmentation des clients dans le marketing et la maintenance prédictive dans le secteur industriel . Ainsi, notre démarche ouvre la voie à une exploitation plus large de l'intelligence artificielle dans des problématiques décisionnelles complexes.

Comme synthèse générale de ce document, ce mémoire nous a permis d'acquérir une expérience concrète en traitement et analyse de données réelles, de mobiliser nos connaissances théoriques pour résoudre une problématique actuelle et d'envisager des perspectives de recherche et de développement dans des secteurs multidisciplinaires. En réalisant ce projet de fin d'étude, nous avons démontré que la science des données offre des outils puissants pour exploiter les données médicales et construire des modèles prédictifs fiables. Le prétraitement rigoureux des données, combiné à une analyse exploratoire fine et à des techniques de modélisation avancées, nous a permis de développer

un système capable d'anticiper avec une précision satisfaisante le résultat d'un test de dépistage de la COVID-19. Le modèle final, sélectionné à l'issue d'une comparaison méthodique, présente des performances encourageantes, ouvrant la voie à des applications concrètes dans les systèmes d'aide à la décision médicale. Ce travail met en lumière les bénéfices, mais aussi les responsabilités, liés à l'intégration de l'intelligence artificielle dans la santé.

BIBLIOGRAPHIE

[1] Foster Provost et Tom Fawcett, *Data Science for Business : What You Need to Know about Data Mining and Data-Analytic Thinking*, O'Reilly Media, 2013.

[2] E. F. Codd, "A Relational Model of Data for Large Shared Data Banks", *Communications of the ACM*, vol. 13, no. 6, pp. 377–387, 1970.

[3] Fayyad, U., Piatetsky-Shapiro, G., & Smyth, P. (1996). From Data Mining to Knowledge Discovery in Databases. *AI Magazine*, 17(3), 37–54.

[4] Mayer-Schönberger, Viktor, et Kenneth Cukier. *Big Data : A Revolution That Will Transform How We Live, Work, and Think*. Houghton Mifflin Harcourt, 2013.

[5] Batini, C., Cappiello, C., Francalanci, C., & Maurino, A. (2009). *Methodologies for data quality assessment and improvement*. ACM Computing Surveys (CSUR), 41(3), 1–52.

[6] Whittaker, M. et al. (2018). *AI Now Report 2018*. AI Now Institute, New York University. `https://ainowinstitute.org/AI_Now_2018_Report.pdf`

[7] Zarsky, T. Z. (2017). *Privacy and Big Data : The Players, Regulators, and Stakeholders*. Oxford University Press.

[8] Kelleher, J. D., & Tierney, B. (2018). *Data Science*. The MIT Press Essential Knowledge Series.

[9] Provost, F., & Fawcett, T. (2013). *Data Science for Business : What You Need to Know about Data Mining and Data-Analytic Thinking*. O'Reilly Media.

[10] Murtagh, F., & Devlin, K. (2017). *The Essence of Data Science : Knowledge Discovery from Data*. CRC Press.

role

[11] Gartner Research. "Analytics Types Defined : Descriptive, Diagnostic, Predictive, Prescriptive," Gartner, 2017.

[12] Zhan, Q., & Chao, K.-M. (2021). *Cloud Computing and Big Data : Technologies, Applications and Security*. Springer.

[13] Votre Nom (2024). *Applications des technologies par secteur*. [Schéma].

[14] Ristevski, B., & Chen, M. (2018). Big Data Analytics in Medicine and Healthcare. *Journal of Integrative Bioinformatics*, 15(3), 1–15. https ://doi.org/10.1515/jib-2017-0030

[15] Morley, J., Machado, C., Burr, C., Cowls, J., Joshi, I., Taddeo, M., Floridi, L. (2020). The ethics of AI in health care : A mapping review. *Social Science Medicine*, 260, 113172. https ://doi.org/10.1016/j.socscimed.2020.113172

[16] Bisong, Ekaba. *Building Machine Learning and Deep Learning Models on Google Cloud Platform*.Apress, 2019.

[17] Van Rossum, Guido, et Fred L. Drake. *The Python Language Reference Manual*. Python Software Foundation, 2001.

[18] VanderPlas, J. (2016). *Python Data Science Handbook : Essential Tools for Working with Data*. O'Reilly Media, Inc.

[19] Han, Jiawei, Kamber, Micheline, and Pei, Jian.
Data Mining : Concepts and Techniques.
Morgan Kaufmann Publishers, 2011.

[20] Provost, Foster, and Tom Fawcett.
Data Science for Business : What You Need to Know about Data Mining and Data-Analytic Thinking.
O'Reilly Media, 2013.

[21] Roderick J. A. Little and Donald B. Rubin, *Statistical Analysis with Missing Data*, 3rd edition, Wiley, 2019.

[22] Aurélien Géron,*Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow : Concepts, Tools, and Techniques to Build Intelligent Systems*, 2nd edition, O'Reilly Media, 2019.

[23] John W. Tukey,*Exploratory Data Analysis*, Addison-Wesley, 1977.

[24] Freedman, D. A., Pisani, R., & Purves, R. (2007). *Statistics* (4th ed.). W. W. Norton & Company.

[25] Fawcett, T. (2006). An introduction to ROC analysis. *Pattern Recognition Letters*, 27(8), 861–874.

[26] Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The Elements of Statistical Learning : Data Mining, Inference, and Prediction* (2nd ed.). Springer.

[27] Pedregosa, F. et al. (2011). Scikit-learn : Machine Learning in Python. *Journal of Machine Learning Research*, 12, 2825–2830.

[28] Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*. Springer.

[29] Breiman, L. (2001). Random forests. *Machine Learning*, 45(1), 5–32.

[30] Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine Learning*, 20(3), 273–297.

[31] Cover, T., & Hart, P. (1967). Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, 13(1), 21–27.

[32] Freund, Y., & Schapire, R. E. (1997). A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting. *Journal of Computer and System Sciences*, 55(1), 119–139.

[33] Kuhn, M., & Johnson, K. (2013). *Applied Predictive Modeling*. Springer.

[34] Zhang, C., & Ma, Y. (2012). *Ensemble Machine Learning : Methods and Applications*. Springer.

[35] Raschka, S., & Mirjalili, V. (2022). *Python Machine Learning : Machine Learning and Deep Learning with Python, scikit-learn, and TensorFlow 2* (4th ed.). Packt Publishing.

[36] Géron, A. (2019). *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow : Concepts, Tools, and Techniques to Build Intelligent Systems* (2nd ed.). O'Reilly Media.

[37] Saito, T., & Rehmsmeier, M. (2015). The precision-recall plot is more informative than the ROC plot when evaluating binary classifiers on imbalanced datasets. *PLOS ONE*, 10(3), e0118432. https ://doi.org/10.1371/journal.pone.0118432

[38] Bergstra, J., & Bengio, Y. (2012). Random Search for Hyper-Parameter Optimization. *Journal of Machine Learning Research*, 13(Feb), 281–305.

[39] Bergstra, J.,& Bengio, Y. (2012). Random Search for Hyper-Parameter Optimization. *Journal of Machine Learning Research*, 13(Feb), 281–305.