Maintenance prédictive pour une turbine de puissance en utilisant des algorithmes d'apprentissage machine

par

Mariem JEMMALI

MÉMOIRE PRÉSENTÉ À L'ÉCOLE DE TECHNOLOGIE SUPÉRIEURE
COMME EXIGENCE PARTIELLE À L'OBTENTION DE
LA MAITRISE EN GÉNIE DE LA PRODUCTION AUTOMATISÉE
M. Sc. A.

MONTRÉAL, LE 17 DÉCEMBRE 2021

ÉCOLE DE TECHNOLOGIE SUPÉRIEURE
UNIVERSITÉ DU QUÉBEC

PRÉSENTATION DU JURY

CE MÉMOIRE A ÉTÉ ÉVALUÉ

PAR UN JURY COMPOSÉ DE :

M. Amin Chaabane, directeur de mémoire
Département de génie des systèmes à l'École de technologie supérieure

M. Antoine Tahan, président du jury
Département de génie mécanique à l'École de technologie supérieure

M. Michel Rioux, membre du jury
Département de génie des systèmes à l'École de technologie supérieure

Mme Katherine Schmidt, examinatrice externe
Siemens Energy

IL A FAIT L'OBJET D'UNE SOUTENANCE DEVANT JURY ET PUBLIC

LE 9 DÉCEMBRE 2021

À L'ÉCOLE DE TECHNOLOGIE SUPÉRIEURE

REMERCIEMENTS

C’est avec un réel plaisir que je résume ces quelques lignes en signe de gratitude et de profonde reconnaissance à tous ceux qui, de près ou de loin, ont participé à la réalisation et à l’accomplissement de mon projet de maitrise. Un grand merci à tous ceux qui m'ont apporté le soutien moral, intellectuel et technique dont j'avais besoin, principalement mes chers parents et ma sœur à qui je dédie ce modeste travail en récompense de leurs sacrifices et leur clairvoyance qui m'a servi et me servirait tout au long de ma vie.

Je remercie plus particulièrement mon directeur de recherche, M. Amin CHABAANE, pour m’avoir encadrée et épaulée tout au long de cet ambitieux projet. Je lui adresse ma profonde reconnaissance pour l’intérêt avec lequel il a suivi la progression de mon travail. Les conseils, la disponibilité et la patience qu’il m’a accordée m’ont permis de mener à terme mes travaux.

Mes vifs remerciements s’adressent également aux membres du jury. Je remercie M. Antoine TAHAN de m’avoir fait l’honneur de présider le jury. Je remercie également M. Michel RIOUX d’avoir accepté d’être rapporteur et d’avoir consacré du temps à examiner mon travail.

J’adresse ma gratitude à l’organisme MITACS et à SIEMENS ENERGY pour leur soutien financier et pour leur aide à m’intégrer dans le monde professionnel. Je remercie particulièrement Mme Katherine SCHMIDT et M. Jean-François FERLAND qui ont bien assuré le suivi du travail et le projet. Je remercie également, M. Rick OCONNOR et la compagnie INDUSTRIAL TURBINE COMPANY (R-U) LIMITED pour sa contribution et sa collaboration.

Je voudrais également remercier tous les membres du Laboratoire de recherche NUMERIX à École de technologie supérieure (ÉTS), mes amis et collègues qui m’ont apporté tout le soutien moral nécessaire pendant ces deux années de maitrise. Merci!

Maintenance prédictive pour une turbine de puissance en utilisant des algorithmes d'apprentissage machine

Mariem JEMMALI

RÉSUMÉ

Aujourd'hui, des technologies telles que l'automatisation et l'apprentissage machine s'introduisent à plein régime au sein de plusieurs secteurs industriels. Cela permet aux entreprises de tirer un meilleur profit de leurs chaînes de production. L'un des plus grands atouts de cette avancée technologique est de permettre aux équipements de fonctionner plus longtemps sans l'apparition de panne. Ceci est une conséquence de la maintenance prédictive appliquée au système industriel et réalisée avec des systèmes d'analyse de données.

La maintenance prédictive consiste à prévoir les pannes des machines avant qu'elles ne surviennent et à éviter les pauses non programmées. L'une des principales techniques utilisées dans le cadre de la maintenance prédictive est l'extraction des données vibratoires des machines. Les informations extraites de ces données permettent d'identifier, de prévoir et de prévenir les pannes des machines rotatives. Dans ce contexte, ce mémoire propose une méthode de diagnostic et pronostic du défaut de déséquilibre d'une turbine de puissance à l'aide des techniques de l'intelligence artificielle et l'apprentissage machine en mettant en exergue leurs avantages par rapport aux anciennes techniques de diagnostic principalement l'analyse temporelle et l'analyse fréquentielle des signaux.

Cette étude de recherche a été réalisée en collaboration avec un partenaire industriel, Siemens Energy, qui est basé à Montréal. Elle a pour but d'améliorer et d'automatiser leur processus de maintenance quant à la détection de l'état de leurs machines. L'objectif de ce travail est donc de proposer une démarche de maintenance prédictive pour suivre et détecter en temps réel si la turbine de puissance est sujette d'un défaut de balourd ou elle est en état de fonctionnement normal à l'aide des techniques de l'intelligence artificielle. Pour cela, une étude comparative de différentes approches analytiques et de plusieurs algorithmes d'apprentissage machine tel que le SVM (*Support Vector Machine*), le RF (*Random Forest*) et le KNN (*K Nearst Neighbors*), a été réalisée afin de choisir l'approche la plus efficiente en ce qui concerne la détection du défaut.

Mots-clés : Maintenance prédictive, industrie 4.0, diagnostic, détection des défauts, déséquilibre, turbine de puissance, machine rotative, analyse vibratoire, apprentissage machine.

Predictive maintenance system for a power turbine using machine learning algorithms

Mariem JEMMALI

ABSTRACT

Today, automation and machine learning technologies are making their way into the industry, allowing companies to get the most out of their production chains. One of the greatest strengths of this technological advancement is that it enables equipment to function longer without failure. This is a consequence of the predictive maintenance applied to the industrial system and carried out with data analysis systems.

Predictive maintenance is about predicting machine failures before they happen and avoiding unscheduled downtime. One of the main techniques generally used in predictive maintenance is the extraction of vibration data from machines. The information extracted from this data allows us to identify, predict, and prevent rotating machinery breakdowns. In this context, this thesis proposes a method of diagnosis and prognosis of the imbalance fault of a power turbine using the techniques of artificial intelligence and machine learning by putting their advantages over the old diagnostic procedures, mainly temporal analysis and frequency analysis of signals.

This research study is designed to collaborate with our industrial partner, Siemens Energy, based in Montreal. It aims to improve and automate its machine condition maintenance process. Therefore, this work aims to propose a predictive maintenance approach to monitor and detect in real-time whether the power turbine is subject to an imbalance or it is in normal operating condition using the techniques of artificial intelligence. For this, a comparative study of different analytical approaches and several machine learning algorithms, such as SVM (Support Vector Machine), RF (Random Forest), and KNN (K Nearest Neighbors), was made to choose the most efficient method in terms of defect detection.

Keywords: Predictive maintenance, industry 4.0, diagnostics, fault detection, imbalance, power turbine, rotary machine, vibration analysis, machine learning.

TABLE DES MATIÈRES

Page

LISTE DES TABLEAUX

Page

LISTE DES FIGURES

Page

LISTE DES ABRÉVIATIONS, SIGLES ET ACRONYMES

IoT	*Internet of Things*
RF	*Random Forest*
MP	Maintenance Prédictive
SVM	*Support Vector Machine*
IA	Intelligence Artificielle
KNN	*K Nearst Neighbors.*
ANN	*Artificial Neural Network*
ML	*Machine Learning,* apprentissage machine
RMS	*Root Mean Square*
RUL	*Remaining Useful Life*
SVR	Support Vector Regression
CF	*Crest Factor*, Facteur de crête
SMOTE	*Synthetic Minority Oversampling Technique*
FFT	*Fast Fourier Transform*
CWT	*Continuous Wavelet Transform*
CWD	*Discrete Wavelet Transform*
Fr	Fréquence de rotation
DT	*Decision Tree*

INTRODUCTION

0.1 Mise en contexte

L'industrie manufacturière a subi des changements majeurs, déclenchés par le progrès rapide des technologies numériques et une réduction des coûts associés aux technologies. De même, des efforts de recherche intensifs sont concentrés sur la découverte des modèles d'adoption et des résultats de performance des usines de fabrication dans « Industrie 4.0» (2019; Morgan R. Frank, 2019).

Dans les usines, les fonctions de maintenance font également de leur mieux pour répondre à ce changement. En effet, la concurrence croissante entre les industries a conduit à l'émergence de nombreux outils et méthodes notamment les techniques de l'intelligence artificielle et les approches analytiques. L'objectif d'utiliser ces différents outils pour améliorer la prise de décision opérationnelle et en temps réel de manière efficiente.

De ce fait, l'industrie 4.0 révolutionne la façon dont la maintenance est effectuée et perçue aujourd'hui. La maintenance prédictive est en train de devenir l'un des moyens les plus applicables avec des cas d'utilisation potentiels dans de nombreux secteurs. Cette période de transformation numérique ouvre la voie à une ère de changements rapides et de nouveaux défis. En effet, l'industrie 4.0 permet aux entreprises manufacturières d'atteindre leurs objectifs avec beaucoup de succès. Dans un environnement où de nombreux capteurs sont connectés et envoient des informations en temps réel, l'Internet des objets contribue à l'intégration de nouvelles opportunités dans l'atelier. L'intelligence artificielle (IA) et l'apprentissage automatique (ML) sont l'une des principales technologies intégrées dans cette vague. L'accès aux données de production est devenu une ressource extrêmement importante et il est peu coûteux de capturer et de stocker ces informations. En outre, la maintenance des processus représente une part importante du coût d'exploitation. Prédire le comportement d'une machine est devenu une fonction courante chez les fabricants pour réduire ce type de dépenses. Certaines entreprises mettent en œuvre une maintenance préventive via le système de

supervision SCADA « *Supervisory Control And Data Acquisition* » (système de contrôle et d'acquisition de données), où les paramètres et les règles d'alerte statiques sont créés par l'homme (Kamat & Sugandhi, 2020). Cette procédure manuelle, dans la plupart des cas, ne prend pas en compte les comportements complexes des machines ou la caractéristique du processus dans son ensemble. De cette façon, les alertes sont déclenchées par erreur avec de « faux positifs », ou ne détectent pas les pannes réelles, « faux négatifs ». C'est exactement là qu'interviennent les techniques d'apprentissage machine (ML). Les algorithmes d'apprentissage automatique utilisent des données d'atelier telles que des capteurs, des superviseurs, des scanneurs, ainsi que des données de production. Ils apprennent à partir de ces données en permettant à l'algorithme de détecter les anomalies, de fournir une corrélation entre les données, de développer une compréhension et de prendre des décisions (appen, 2021). Par conséquent plus les données d'entrainement sont bonnes, meilleures sont les performances du modèle.

En ML, le processus d'entrainement des données a la capacité d'analyser une grande masse de données en temps réel le rendant ainsi, l'un des principaux facteurs d'utilisation des techniques d'apprentissage machine. En effet, La santé et le comportement des ressources de production sont analysés en permanence pour prévoir et détecter les pannes .

Dans ce cadre, un projet de collaboration avec Siemens Energy a été mis en place afin d'étudier leurs besoins en termes de maintenance et d'en faire un projet de recherche visant à améliorer leur processus de détection des défauts, principalement les défauts d'équilibrage dans les turbines de puissance. Durant ce travail, on s'est basé sur la littérature pour élaborer notre démarche de maintenance prédictive et effectuer une étude comparative entre plusieurs techniques d'analyse prédictive et apprentissage machine pour mieux répondre aux besoins du partenaire industriel.

0.2 Problématique

Les vibrations se retrouvent un peu partout dans les machines tournantes. Elles sont généralement le résultat de défauts mécaniques, notamment le déséquilibre de masse, le désalignement de l'accouplement, le desserrage mécanique et de nombreuses autres causes (Saleem, Garikapati, & Rama Surya Satyanarayana, 2012). Le déséquilibre du rotor est la raison la plus courante des vibrations causant la plupart des problèmes des machines tournantes. Le déséquilibre peut même détruire des pièces critiques de la machine, telles que les roulements, les joints, les engrenages et les accouplements (Saleem et al., 2012). En effet, une très petite quantité de balourd peut causer de graves problèmes dans les machines tournantes à grande vitesse et à mesure que leurs états se détériorent, le niveau de vibration augmente.

Pour faire face à cette problématique, Siemens Energy s'intéresse de plus en plus à implémenter une approche de maintenance prédictive pour prédire et détecter à temps les défauts qui peuvent apparaître dans leurs systèmes rotatifs principalement les turbines de puissance. En effet, ces dernières, sont fréquemment sujettes au problème de balourd. Celui-ci représente la source de vibrations la plus courante dans les machines tournantes, mais aussi la plus destructrice (Kim, Velas, & Lee, 2006). Il est donc important de détecter ce défaut rapidement avant que l'état de la turbine devienne critique et y remédier par équilibrage. En fait, le balourd qui est synonyme du déséquilibre, est défini par la norme internationale ISO 1925 "Vibrations mécaniques - Équilibrage - Vocabulaire" comme étant « *l'état dans lequel se trouve un rotor quand, à la suite de forces centrifuges, une force ou un mouvement vibratoire est communiqué à ses paliers »(ISO-1925, 2001)*. La figure 0.1 illustre ce phénomène. Les causes les plus communes du balourd sont la non-homogénéité des matériaux, les tolérances de fabrication et d'assemblage ainsi qu'une modification physique du rotor en marche.

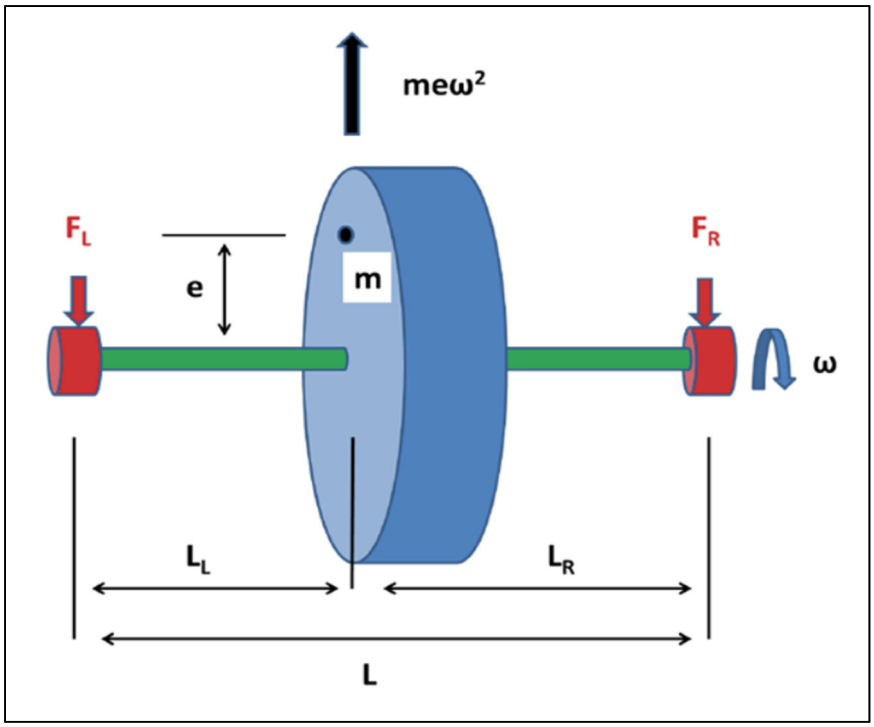


Figure 0.1 Masse de déséquilibre
Tirée de Smith (2014)

En mesurant et en surveillant les vibrations produites par une machine, on obtient un indicateur idéal de son état. En effet, si l'augmentation des vibrations de la machine induit la présence d'un défaut, l'analyse des caractéristiques de vibration peut en identifier la cause. Par conséquent, on peut régler le problème avant qu'il ne devienne critique et engendre l'arrêt total de la machine.

0.3 Objectifs et contributions

L'objectif ultime de ce projet est de proposer une approche pour détecter automatiquement l'état de balancement des turbines de puissance. Il s'agit de détecter si la turbine est en état de déséquilibre ou si elle fonctionne dans ses conditions normales. Le système vise à détecter, pronostiquer et prévoir le dysfonctionnement de la turbine à partir des signatures vibratoires mesurées.

La valeur ajoutée de ce travail est d'explorer et d'évaluer plusieurs méthodes et algorithmes de classification et d'apprentissage machine utilisés dans le cadre de la maintenance prédictive appliquée aux machines rotatives, principalement les turbines de puissance.

Cela va permettre de :

- Réduire les temps d'arrêt des interventions de maintenance ainsi que les déplacements coûteux causés par les temps d'arrêt non planifiés.
- Augmenter la productivité. Cette dernière ne sera plus affectée par les interruptions inattendues.
- Améliorer la sécurité des travailleurs. En effet, certains dysfonctionnements comme les grandes vibrations soudaines entrainent des conditions de travail néfastes.

0.4 Méthodologie

La formulation de la problématique de notre recherche s'est basée sur le choix du thème à étudier et motivée principalement par un besoin industriel.

On a commencé par faire une étude bibliographique pour identifier les avantages et les inconvénients des méthodes existantes dans la littérature traitant le même sujet pour en sélectionner les plus pertinentes. La finalité de cette étape est d'identifier ce dont on a besoin comme outils de travail et d'élaborer une démarche claire pour réaliser le projet.

Il s'agit d'un projet de recherche appliquée basé sur des données de vibrations réelles. Ainsi, la méthodologie qu'on propose est définie en trois grandes étapes décrites brièvement dans cette section et dont les détails seront traités dans le Chapitre 3.

La première étape consiste à collecter suffisamment de données nécessaires pour mettre en œuvre une approche de maintenance prédictive basée sur les données et les algorithmes

d'apprentissage machine en comparant plusieurs méthodes. L'ensemble des données utilisé dans ce projet a été fourni par Siemens Energy et issu d'une turbine de puissance.

La deuxième étape comprend le traitement et l'analyse des données afin d'en sélectionner les plus pertinents pour la détection du déséquilibre. Pour cela, on a utilisé la programmation en Python en utilisant « *Pycharm*. » On a également utilisé la combinaison des techniques de traitement de signaux et le débruitage par la transformée d'ondelette. Par la suite, on a fait une analyse temporelle et fréquentielle pour l'extraction des informations utiles pour le diagnostic et la détection du balourd.

Finalement, la troisième étape consiste à utiliser et comparer différentes méthodes de classifications basées sur l'apprentissage automatique pour la reconnaissance du défaut de déséquilibre présent dans la machine. Le but est de reconnaitre les bons et les mauvais signaux à partir des données vibratoires collectées en temps réel d'une turbine de puissance en marche.

0.5 Organisation du mémoire

Le mémoire a été divisé en cinq chapitres en plus de la conclusion. Le premier chapitre porte sur la revue de la littérature. Cette dernière met en exergue les anciens travaux de recherches réalisés dans les domaines du diagnostic et détections des défauts principalement dans les machines rotatives. Dans ce chapitre, on souligne les différentes techniques les plus employées dans la littérature en présentant quelques définitions de base liées à l'industrie 4.0 et à la maintenance prédictive en général. Dans le deuxième chapitre, on développe et explique la méthodologie suivie afin d'aboutir à la réalisation du projet et répondre au besoin industriel défini précédemment. Dans le troisième chapitre, on propose une comparaison entre deux méthodes d'analyse vibratoire couramment utilisées dans le domaine du diagnostic et de la détection des défauts qui sont l'analyse temporelle et l'analyse fréquentielle. Cette étude est nécessaire pour comprendre l'étape d'extraction des caractéristiques et savoir différencier les bons signaux des mauvais. On montre aussi comment ces méthodes sont utilisées dans l'industrie pour détecter les anomalies. Le quatrième chapitre expose l'étape de développement

des modèles de prédiction dans le but de détecter le déséquilibre des turbines de puissance. Dans cette section, une comparaison entre trois algorithmes de classification a été proposée : la méthode des K plus proches voisins (KNN), la méthode des machines à vecteurs de support (SVM) et la méthode des forêts aléatoires (RF). Dans le cinquième chapitre, un test de validité externe a été réalisé dans le but de détecter les faiblesses du modèle et d'y apporter des améliorations. Enfin, une discussion générale de la problématique ainsi que les contributions de ce mémoire, les limites et les suggestions des travaux futurs sont présentées dans la conclusion.

CHAPITRE 1

CONCEPTS DE BASE ET REVUE DE LA LITTÉRATURE

1.1 Introduction

Dans ce chapitre, une revue de la littérature abordant la thématique de recherche et ses différents aspects est proposée. On commence par montrer l'évolution industrielle et l'apparition de l'industrie 4.0. Ensuite, on met l'accent sur les notions liées à la maintenance prédictive, ses différentes approches et les multiples techniques et outils utilisés dans les travaux de recherche antérieurs en lien avec la problématique.

1.2 Concepts de base

Cette section expose les différents concepts de base en lien avec la maintenance prédictive.

1.2.1 L'évolution industrielle et l'industrie 4.0

L'idée de l'industrie 4.0 a été connue pour la première fois en 2011 pour améliorer l'efficacité de l'industrie manufacturière. Avec la complexité croissante, les demandes et la concurrence sur le marché international, les exigences de produits personnalisés sont les principaux défis des entreprises. L'industrie 4.0 peut potentiellement avoir un impact sur les systèmes de fabrication grâce aux progrès rapides de la numérisation et aux technologies telles que l'Internet des objets, les données massives et l'intelligence artificielle y compris l'apprentissage machine. Auparavant, trois transformations modernes ont eu lieu (Sufiyan Sajida, 2021): 1re révolution, caractérisée par la mécanisation, 2e révolution définie par l'électrification, et 3e révolution marquée par l'automatisation et la mondialisation. La figure 1.1 illustre ces révolutions.

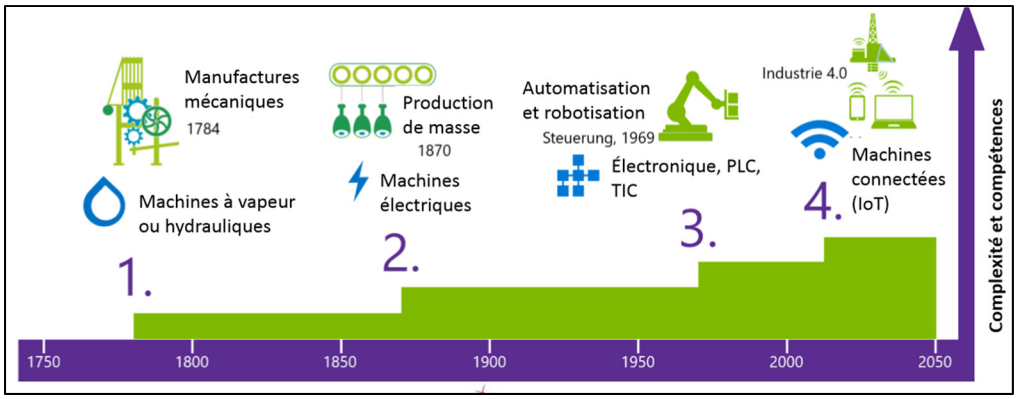


Figure 1.1 La révolution industrielle tirée de Rioux (2016)

Dans l'industrie, l'évolution vers l'industrie 4.0 peut se faire à plusieurs niveaux, y compris les opérations de maintenance des équipements.

1.2.2 Définition de la maintenance industrielle

Selon la norme AFNOR (NF X 60 010), la maintenance est un « ensemble des actions permettant de maintenir ou de rétablir un bien dans un état spécifié ou en mesure d'assurer un service déterminé ». Bien maintenir, c'est assurer ces opérations au coût optimal. En effet, quel que soit le domaine d'activité, les industriels ne doivent pas négliger les coûts et les différents impacts qu'une défaillance soudaine pourrait entrainer. Leur productivité repose en grande partie sur leurs processus de maintenance.

1.2.3 Les différents types de maintenance industrielle

Il existe trois types de maintenance industrielle. Le premier type est la maintenance corrective. Elle sert à réparer la défaillance de tout équipement dès son arrivée. Elle vise à remettre l'équipement défaillant en état de fonctionnement après la panne. C'est une maintenance corrective ou appelée aussi maintenance curative. Le deuxième type de maintenance est la maintenance préventive. Cette dernière vise à détecter et à résoudre les problèmes avant qu'ils ne surviennent. Elle est généralement réalisée sous la forme d'inspections régulières et planifiées. Le troisième type est celui de la maintenance prédictive. Selon le dictionnaire Cambridge (Cambridge University Press, 2021), l'adjectif prédictif est défini comme « dire

qu'un événement ou une action se produira dans le futur ». L'entretien est défini comme « les travaux nécessaires pour maintenir une machine en bon état ». En d'autres termes, la maintenance prédictive peut être définie comme un outil, qui permet de prédire le futur point de défaillance d'un composant de machine ou système avant qu'il ne tombe en panne. En fait, la maintenance corrective est appliquée après la panne, alors que la maintenance préventive utilise des mesures de précaution pour éviter d'éventuels problèmes. La maintenance prédictive évalue l'état de l'équipement existant et, sur la base d'une tendance projetée du processus de détérioration, les pannes sont prédites et les mesures appropriées sont prises (Matthew P. Stephens, 2014).

1.2.4 Démarche d'une approche de maintenance prédictive basée sur les données

Pour pouvoir mettre en œuvre une stratégie de maintenance prédictive, on doit d'abord avoir des capteurs connectés qui mesurent continuellement des paramètres de fonctionnement de l'équipement concerné. Ces données sont collectées massivement puis transmises à l'aide de l'IoT à un moteur d'intelligence (Thyago P et al., 2019). Ce moteur analyse les mégadonnées et les croiser avec les rapports d'intervention effectuée sur les mêmes équipements. Le modèle prédictif va ainsi peu à peu mettre en évidence des corrélations entre certaines informations transmises et les pannes. Il pourra donc apprendre que telles valeurs mesurées précèdent tel type de défaillance. Ainsi dès que ces valeurs seront à nouveau mesurées, il pourra planifier une action de maintenance et éviter l'apparition de la panne.

Pour cela, plusieurs outils sont disponibles pour chaque phase de construction, et chaque outil est adapté à des caractéristiques bien précises qui dépendent de l'équipement et du domaine d'activité de l'industrie. Dans ce contexte, Radhya, John, & Muhammad Intizar (2020) ont proposé un ensemble de lignes directrices aux décideurs afin de les guider dans la sélection des technologies les plus appropriées pour répondre à leurs besoins. Une fois les différentes technologies à utiliser ont été sélectionnées, il faut passer à l'étape suivante qui est le développement du modèle de maintenance prédictive. Pour cela, quatre étapes clés ont été identifiées. En effet, la première étape consiste à collecter les données relatives aux

équipements, puis les analyser pour les classer en état normal et état non normal. Ensuite, la deuxième phase se traduit par la modélisation des schémas de panne à l'aide des algorithmes d'identification des anomalies d'une part et, d'autre part, de pouvoir les classer en plusieurs catégories en se basant sur un historique des pannes. L'étape suivante est de développer le modèle prédictif en l'apprenant à reconnaitre les nouveaux événements et défaillances lorsqu'ils surviennent. Enfin, ce qui est intéressant est de pouvoir adapter le système pour qu'il puisse mettre à jour sa base de données en fonction des nouvelles informations collectées sur le matériel.

Les systèmes IoT permettent d'affiner les données collectées ce qui offre une nouvelle valeur à l'opérateur du système IoT. Pour obtenir de nouvelles informations, le processus qui est détecté doit être modélisé. Ce processus de modélisation et d'extraction de connaissances à partir d'ensembles de données est appelé apprentissage automatique (Kapil & Kiran, 2018). Cela peut être fait, si une quantité suffisante de données est disponible. En plus des données brutes, selon le cas d'utilisation, une description détaillée des données peut être requise. Si, par exemple, un modèle prédisant les défauts est souhaité, il est nécessaire d'apprendre au modèle à détecter les défauts à la fois des données de fonctionnement sans défaut et de fonctionnement défectueux. Le modèle peut ensuite être entrainé avec les données et détecter l'état actuel du dispositif (Kapil & Kiran, 2018). Le développement de modèles peut être divisé en cinq (5) étapes comme le montre le diagramme de la figure 1.2.

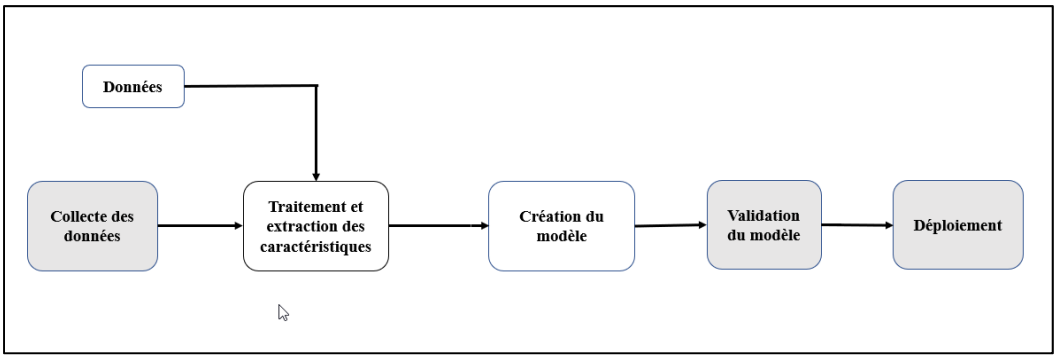


Figure 1.2 Les étapes de création d'un modèle de prédiction
Adapté de Wanga, Ghani, & Kalegele (2017)

Avant que le modèle ne soit créé avec un algorithme d'apprentissages automatique dans la phase de création du modèle, l'ensemble de données est généralement divisé en deux sous-ensembles : « données d'entrainement » et « données » de test. Seul le sous-ensemble d'entrainement est utilisé pour créer un modèle. Le sous-ensemble, données de test, est ensuite utilisé pour tester le fonctionnement du modèle créé. Le flux des travaux à réaliser pour créer un modèle d'apprentissage automatique est présenté dans la figure 1.3 ci-dessous.

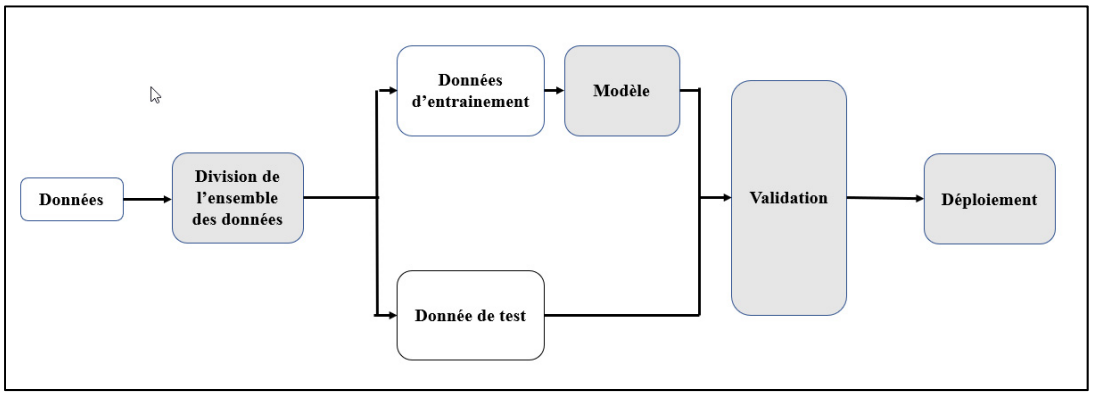


Figure 1.3 Création d'un modèle d'apprentissage automatique
Adapté deWanga (2017)

1.2.4.1 Prétraitement des données et extraction des caractéristiques

Une exigence cruciale pour la modélisation concerne les données. Elles doivent être collectées avant que toute analyse puisse être mise en œuvre. L'étape suivante consiste à traiter les données et extraire les caractéristiques nécessaires à la phase d'apprentissage. Cette étape de prétraitement traite et transforme les données afin qu'elles puissent être traitées efficacement par le modèle ML. Elle comprend la transformation des données par exemple, la normalisation, le nettoyage des données, le traitement des données manquantes, la suppression des valeurs aberrantes et la réduction des données (Thyago P et al., 2019). Par la suite, une phase d'analyse des données est nécessaire. L'objectif de cette étape est de découvrir les tendances possibles, d'utiliser des tests statistiques appropriés et de résumer les données graphiquement et numériquement.

1.2.4.2 Modèle d’apprentissage machine

Dépendamment du type des données disponibles pendant la phase de création du modèle, l’apprentissage machine est qualifié de différentes manières. En effet, après avoir s’être familiarisé avec les données, l’étape suivante consiste à appliquer un modèle pour prédire le type de défaut. La plupart des modèles appliqués à la maintenance prédictive sont basés sur des statistiques ou sur l'intelligence artificielle. Ces modèles sont capables de traiter et de saisir des relations complexes entre les données. Un point clé des modèles d'apprentissage automatique est leur processus d'apprentissage et dépend de l'application, de l'objectif et des données disponibles pour le système (Russel & Norvig, 2012). Il existe deux types d’apprentissages machine comme montré dans la figure 1.4 (Jaber, 2018).

On parle d’apprentissage supervisé quand les données utilisées dans l’entrainement d’une machine sont labellisées. C’est-à-dire des données qui ont déjà été étiquetées avec le bon « label », appelé aussi classe. Cet apprentissage, connaissant déjà la classe, permet de prédire par la suite le label de données nouvelles non étiquetées. En effet, les modèles sont entrainés en mettant des données d'entrainement en entrée, et le résultat d'intérêt est connu. La plupart des articles classent la régression et la classification dans cette approche d'apprentissage. En régression, le résultat est numérique, alors que pour la classification, le résultat est un exemple catégorique, « oui » ou « non ». Des algorithmes possibles sont les statistiques bayésiennes, l'apprentissage par arbre de décision, ou la forêt aléatoire (M. Mohri, 2018).

Contrairement à l’apprentissage supervisé qui tente de trouver un modèle depuis des données labellisées, l’apprentissage non supervisé utilise des données sans label. Il doit faire émerger automatiquement les catégories à associer aux données qu’on lui soumet pour les reconnaitre en essayant de trouver des modèles qui les caractérisent. Donc, ici, le résultat d'intérêt est inconnu ou non étiqueté pour l'ensemble de données donné. Les principales méthodes utilisées sont l' « agrégation » et « la réduction de dimensionnalité ».

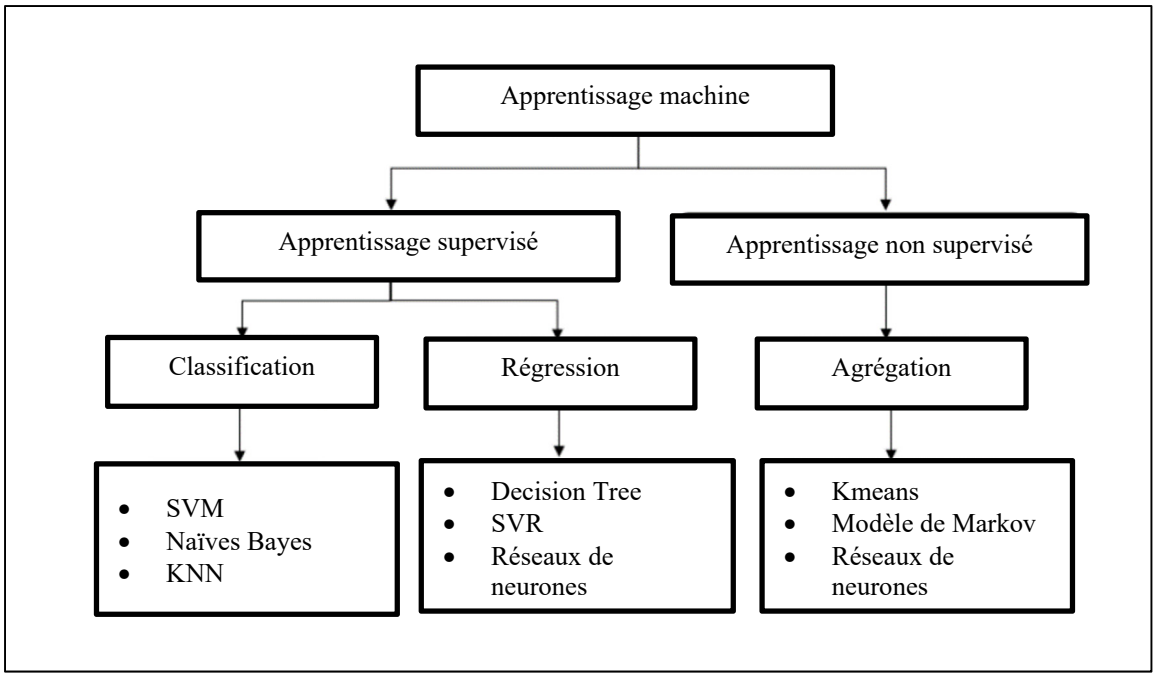


Figure 1.4 Taxonomie des modèles d'apprentissages automatiques tirée de Jaber (2018)

1.2.4.3 Analyse et validation du modèle

Après avoir obtenu les résultats des modèles algorithmiques, les techniques basées sur les données sont combinées avec des techniques basées sur les connaissances pour prendre de meilleures décisions et stratégies (Sufiyan et al., 2021). Des experts expérimentés dans l'industrie examinent les modèles et les résultats, ce qui conduit à des améliorations des procédures d'exploitation, de maintenance, de surveillance, de test et d'audit pour garantir des actions plus sûres et plus efficaces. Ensuite, une phase de déploiement peut être faite et qui représente la dernière étape du flux de travail pour concevoir un système de classification. Elle représente la mise en exploitation du système.

1.3 Revue de la littérature

Cette section présente les différents travaux de recherches appliquées dans le cadre de la maintenance prédictive des machines rotatives. Les principaux travaux ont été regroupés dans l'ANNEXE VI.

1.3.1 Maintenance prédictive et machines tournantes

La maintenance prédictive gagne en importance dans les groupes de recherche multidisciplinaires, proposant la création et l'intégration des systèmes liés à l'acquisition de données, à l'infrastructure, au stockage, à la distribution, à la sécurité et à l'intelligence. En effet, des efforts considérables ont été consacrés au développement de plusieurs méthodes fiables pour la détection des défauts. Les techniques qui ont fait leurs preuves comprennent l'analyse de l'huile de lubrification, le signal acoustique, la surveillance de la température, l'analyse du courant du moteur électrique et le plus populaire aujourd'hui, l'analyse des vibrations. Malheureusement, aucune technique n'est capable de détecter tous les défauts de la machine. Cependant, il a été suggéré de favoriser la mesure des vibrations, qui est la technique la plus utilisée dans la maintenance prédictive. Elle peut identifier avec précision 90% de toutes les pannes des machines par le changement des signaux de vibration qu'elles produisent et le niveau de signal peut donner une prédiction précise d'échec futur et par conséquent suivre l'état de l'équipement.

Dans ce cadre, Qiao & Lu (2015) pensent que la meilleure façon de détecter un défaut, son type et son degré est de faire une analyse du signal issu de la machine concernée. Dans le même contexte , Jin, Zhao, Liu, Lee, & He (2014) et Jie, Hou, & Yongguang (2015) ont utilisé le signal vibratoire dans la détection des défauts. Ils pensent que la surveillance des vibrations est le meilleur outil pour identifier les défaillances. Cependant, d'autres chercheurs comme (Mobley, 2002) trouvent que l'analyse vibratoire ne peut fournir toute l'information permettant le diagnostic des défauts mécanique et suggèrent d'exploiter les signaux électriques comme le courant et la tension (Lu, Gong, & Qiao, 2012) et (Mendonca et al., 2015). Également Kedadouche, Thomas, & Tahan (2012) ont fait une étude comparative des résultats fournis par

les émissions acoustiques et les signaux vibratoires pour des équipements à basses vitesses et ont démontré que les émissions acoustiques sont plus fiables quant à la détection des fréquences des défauts à de basses vitesses.

En outre, plusieurs techniques d'IA ont été développées afin de pouvoir diagnostiquer les fautes survenant dans un système provoquant des défaillances et de mieux suivre son état en collectant un maximum de données. Cependant, il est à noter que, selon Deutsch & He (2018) et Wang, Di Maioa, & Zio (2017), les approches de maintenance qui sont capables de surveiller l'état des équipements à des fins de diagnostic et de pronostic peuvent être regroupées en trois catégories principales.

1.3.1.1 Les approches fondées sur les connaissances et les expériences

On retrouve les méthodes basées sur des règles, les méthodes basées sur les cas et les méthodes basées sur les connaissances. Ces dernières ont été discutées dans les travaux (Luo, Hu, Ye, Zhang, & Wei, 2020; Montero, Jimenez, Rob, Grabot, & Bernard, 2020; Sufiyan et al., 2021). Les approches basées sur les connaissances se présentent sous la forme de « Si-Alors » et elles sont obtenues à partir d'expériences antérieures ou de situations problématiques concrètes. Les inconvénients de cette approche sont la difficulté d'obtenir des connaissances précises à partir de l'expérience et un accès limité à des experts ayant des connaissances. Par conséquent, il en résulte une faible précision de prédiction. Cependant, cette approche est actuellement utilisée dans les techniques d'exploration de données pour extraire les connaissances requises de la base de données.

1.3.1.2 Les approches basées sur des modèles physiques

Ces modèles se caractérisent principalement par la modélisation mathématique avec des réflexes dans l'état d'un composant, nécessitant la précision de l'état et la mesure de la défaillance, et des méthodes statistiques pour limiter ces indices (Wu, Jennings, Terpenny, Gao, & Kumara, 2017) et (Lei et al., 2016). Bien que la méthode basée sur un modèle physique puisse révéler la logique des défauts du système sans collecter beaucoup de données, elle a

besoin du soutien d'experts pour concevoir et établir le modèle. La plupart des équipements sont généralement des systèmes mécaniques et électriques complexes, et par conséquent les modèles de dégradation correspondants sont difficiles à établir avec précision en raison de l'ignorance du mécanisme de dégradation.

1.3.1.3 Les approches basées sur les données

Ces approches sont les plus trouvées dans l'évolution actuelle des solutions de maintenance prédictive et elles sont basées sur les statistiques, la reconnaissance ou l'intelligence artificielle (IA), et les algorithmes d'apprentissage automatique. Dans ce contexte, Baptista et al. (2018) comparent un certain nombre d'approches de l'intelligence artificielle à une approche statistique pour prédire quand un équipement risque de tomber en panne à l'avenir et les résultats suggèrent que les approches d'intelligence artificielle surpassent les approches statistiques. Selon plusieurs chercheurs, il est préférable d'utiliser l'approche basée sur les données lorsque les modèles de système ne sont pas disponibles, mais à la place, des données de surveillance du système sont disponibles. L'apprentissage automatique (ML), au sein de l'intelligence artificielle, est devenu aussi, un outil puissant pour développer des algorithmes prédictifs intelligents dans de nombreuses applications. Ils ont la capacité de gérer des données de grande dimension et multivariées et d'extraire des relations cachées au sein des données dans des environnements complexes et dynamiques, tels que les environnements industriels (Wuest, Weimer, Irgens, & Thoben, 2016). Ainsi, l'apprentissage automatique fournit des approches prédictives puissantes pour les applications de maintenance prédictive. Dans cette approche, les données peuvent être collectées à partir d'appareils en cours d'exécution, et un modèle précis d'évolution des pannes ou un processus de dégradation des performances n'est pas nécessaire (Canizo, Onieva, Conde, Charramendieta, & Trujillo, 2017).

Le modèle autorégressif, le réseau de neurones artificiels (Fan et al., 2019) et la machine à vecteurs de support sont couramment utilisés pour l'analyse des mégadonnées dans le domaine de la maintenance prédictive (Fan et al., 2019). L'approche basée sur les données est très utilisée pour les applications de maintenance prédictive. Cependant, la performance dépend du

choix correct de la technique. Plusieurs chercheurs ont contribué à l'amélioration des différentes approches de la maintenance prédictive.

Récemment, dans les travaux de , une approche hybride pilotée par Digital Twin (DT) est étudiée. Elle est une combinaison entre l'approche basée sur les données et l'approche basée sur le modèle crée un pont entre le monde physique et le monde numériques. Cependant, plusieurs travaux suggèrent des méthodes de pronostic basées sur les données, basées sur des algorithmes d'apprentissage automatique classiques ainsi que des systèmes développés d'acquisition de données et de surveillance des processus. Dans ce contexte Hsu, Wang, Lin, Chen, & Hsu (2020) ont appliqué des techniques de contrôle statistique des processus et d'apprentissage automatique pour diagnostiquer les défaillances des éoliennes et prévoir les besoins de maintenance en analysant 2,8 millions de données de capteurs collectées sur 31 éoliennes de 2015 à 2017 à Taiwan. Contrairement aux études précédentes qui ne s'appuyaient que sur des données historiques sur les éoliennes, cette étude a analysé les données des capteurs avec la perspicacité des praticiens en incorporant des éléments de liste de contrôle de maintenance dans les processus d'exploration de données. Ils ont utilisé les analyses de Pareto, les diagrammes de dispersion et le diagramme de cause à effet pour regrouper et classer les types de défaillance des éoliennes. Par contre, Thyago P et al. (2019) montrent dans leur article que les signaux de vibration sont les données les plus couramment utilisées pour concevoir des modèles de maintenance prédictive avec une préférence pour les données réelles afin de construire des modèles fiables, notamment pour les machines rotatives.

1.3.2 Analyse vibratoire et intelligence artificielle au service de la maintenance prédictive

L'élément rotatif de la machine crée des vibrations qui sont fonction de la dynamique de la machine. Le niveau de vibration de la machine est mesuré à l'aide des capteurs tels que les capteurs de proximité, les transducteurs de vitesse et les accéléromètres. L'analyse des vibrations permet de savoir s'il y a un désalignement ou un déséquilibre du rotor, des roulements et des engrenages détériorés ou défectueux ou si les vibrations dans certains éléments de la machine sont amplifiées par la résonance. Cela se fait grâce à l'analyse des

amplitudes à certaines fréquences ou des tendances dans les mesures statistiques. À cet égard, des approches de surveillance basées sur des signaux vibratoires ont été présentées dans de nombreuses études (Salem, Abu-Siada, & Islam, 2014); (Liu et al., 2015). Des catégories d'approches axées sur l'extraction de caractéristiques utiles pour indiquer des anomalies à partir de signaux de vibration bruyants ont été adoptées. En effet, les techniques d'extraction de caractéristiques peuvent localiser certains composants dans les signaux pour aider à détecter les défauts de la machine. Hongyu, Joseph, & Lin, (2003), McFadden & Smith (1985) et Kim Y W (1995) ont inclus l'analyse spectrale non paramétrique classique, l'analyse en composantes principales, l'analyse temps-fréquence conjointe et la transformée en ondelettes discrètes.

Lebold, Mcclintic , Campbell , Byington & Maynard (2000) ont étudié la méthode statistique d'extraction des caractéristiques pour le diagnostic des défauts de la boîte de vitesses. Tahsin & Jens (2008) ont examiné l'extraction des caractéristiques du domaine temporel pour diagnostiquer les petits défauts des roulements à rouleaux. Dans ce sens, plusieurs chercheurs ont mis l'accent sur l'efficacité du Kurtosis quant à la détection des défauts de roulements. Il s'agit d'un indicateur représentant les impulsions dans un signal. À titre d'exemple, dans J. Mathew (1998), il a été démontré que l'utilisation du Kurtosis pour la détection des défauts par rapport aux autres indicateurs conventionnels comme le PEAK (crête) ou le RMS (*Root mean square*) est plus efficace. Cet indicateur donne une idée sur l'état du roulement sans avoir beaucoup d'information par rapport à l'historique de l'équipement.

Feng et Liang (2014) ont proposé une méthode d'analyse temps-fréquence pour découvrir les composantes fréquentielles constitutives des signaux non stationnaires pour la surveillance des boîtes de vitesses des éoliennes. Également, l'utilisation de la transformée de Fourier rapide est l'un des outils traditionnels largement utilisés pour étudier le spectre et certains éléments fréquentiels afin d'en extraire les caractéristiques. Cependant, au cours des 20 dernières années, la transformée en ondelettes a connu une croissance explosive dans des applications passionnantes d'analyse du signal et l'analyse numérique, et de nombreuses autres applications sont à l'étude (Yang., 2003). En raison de sa forte capacité dans le domaine temporel et

fréquentiel, cette technique est appliquée récemment par de nombreux chercheurs sur des machines tournantes.

Aussi, le développement de la transformée en ondelettes a conduit à la technique de la transformée en ondelettes continues (CWT) (G. Y Luo et al., 2003) et de la transformée en ondelettes discrètes (DWT). Ces techniques ont réussi à traiter la détection de défauts de signaux non stationnaires. En effet, dans le travail de , la transformation de Fourier rapide (FFT) et CWT ont été utilisés pour le traitement des données vibratoires. Le signal de vibration du système de machines a été extrait en le simulant dans différentes conditions de fonctionnement comme un arbre sain, désaligné, déséquilibré et fissuré. Ensuite, ces données ont été traitées à l'aide d'outils de traitement du signal tels que FFT et CWT pour comprendre les caractéristiques de chaque condition de fonctionnement, une fois celles-ci connues, un outil de classification et de détection des défauts est développé. L'outil basé sur un diagramme de plan de phase utilise le signal des quatre accéléromètres pour classer le défaut tandis que la méthode KNN basée sur l'accélération maximale locale utilise les informations de fréquence pour classer le défaut. De même, une méthode basée sur la logique floue est proposée pour classer le défaut et identifier la gravité de l'état de la machine. La méthode proposée a montré une plus grande précision et efficacité dans l'identification et la classification de l'état de la machine.

Les efforts de recherche récents se concentrent sur l'application des techniques d'intelligence artificielle. Dans ce contexte Hoppenstedt et al (2018) ont appliqué une analyse bibliométrique au domaine des systèmes de maintenance des équipements pour fournir un aperçu complet des techniques établies et des tendances émergentes pour les systèmes de maintenance des équipements. En effet, ils ont constaté que les techniques ML, telles que SVM «*support vector machine*», RF «*Random Forest*», ANN «*Artificial neural network*», l'apprentissage profond et «*k-means»*, sont les plus appliquées et avec succès pour concevoir des applications de la maintenance prédictive (Thyago P et al., 2019).

En effet, afin d'extraire des informations utiles des signaux bruts des capteurs, un modèle de moyenne mobile autorégressif avec six algorithmes pilotés par les données a été utilisé dans pour formuler les horaires de maintenance des compagnies aériennes, et le SVM a obtenu les meilleurs résultats globaux. Selon Shafi, Safi, Shahid, Ziauddin, & Saleem (2018), quatre classificateurs ont été comparés, à savoir SVM, DT, RF et KNN. Parmi eux, le classificateur SVM a obtenu les meilleures performances sur quatre systèmes d'exploitation et la précision du SVM était de 96,6%, 98,7%, 98% et 96,6%.

Également, la transformation de Hilbert-Huang a été utilisée dans pour extraire les indicateurs de santé d'un signal de vibration; un modèle SVM a ensuite été utilisé comme solution efficace pour évaluer les états de dégradation, et un modèle SVR a été utilisé pour estimer la durée de vie utile restante (RUL) des roulements dégradés. En revanche, l'extraction des caractéristiques a été effectuée par (O. R. Seryasat, 2010.) à partir du domaine temporel des signaux de vibration, et un SVM multi classe a été déployée pour détecter les défauts de roulement. Dans les travaux de Canizo et al. (2017), la méthode RF est utilisée pour générer dynamiquement des modèles prédictifs. Ce travail propose une amélioration de l'article de Kusiak & Verma (2011) où un suivi des éoliennes est réalisé. Pour ce faire, des données d'état (alarmes activées et désactivées) et des données opérationnelles (sur les performances des éoliennes) sont utilisées pour concevoir le modèle RF. Les principales contributions de l'article sont la vitesse de traitement des données, l'évolutivité et l'automatisation (Canizo et al., 2017). Wu et al. (2017) ont évalué les performances de trois algorithmes d'apprentissage automatique, dont les ANN, SVM et RF, pour la prédiction de l'usure des outils à l'aide des données recueillies à partir de 315 tests de fraisage. Un ensemble de caractéristiques statistiques a été extrait des canaux de force de coupe, de vibration et de signal AE. Les résultats expérimentaux ont montré que le modèle prédictif formé par les RF surpasse les ANN et SVM.

1.4 Conclusion

Cette revue de la littérature nous a permis d'approfondir sur la thématique de notre recherche et de nous familiariser avec les différentes techniques utilisées dans ce domaine. Ce chapitre

nous a permis aussi d'évaluer le degré d'avancement des recherches déjà établies et de comprendre les différentes méthodes et les approches utilisées dans les différents travaux pour sélectionner la méthodologie la plus adaptée au besoin de Siemens. En effet, pour pouvoir caractériser et détecter les défauts des machines rotatives, l'analyse vibratoire s'avère la plus pertinente par rapport à d'autres techniques comme l'analyse des huiles, analyse de températures et la thermographie infrarouge. De plus, pour mettre en œuvre un modèle de maintenance prédictive, il existe plusieurs approches utilisées dans les anciens travaux, mais la plupart des chercheurs considèrent que l'approche basée sur les données est plus intéressante et fiable, et ceci pour deux raisons:

1) L'évolution technologique des capteurs et des systèmes d'acquisition des données ont permis de collecter des données massives liées aux machines et de les stocker pour pouvoir les analyser par la suite grâce aux différents outils d'analyse de données qui n'ont pas cessé de se développer de jour en jour, permettant au chercheur de comprendre et d'extraire les informations les plus utiles à son application.
2) Aujourd'hui, les systèmes industriels sont des systèmes complexes, ce qui rend leurs modélisations et leurs simulations parfois difficiles.

À partir de la revue de la littérature, on a pu catégoriser notre problématique. En effet, il s'agit d'un problème de classification simple visant à identifier l'état de la turbine de puissance. Cependant, il existe plusieurs techniques d'apprentissage automatique employées dans les problèmes de classifications. C'est pour cette raison qu'on va comparer entre les plus recommandées dans les travaux de recherches pour pouvoir choisir celle qui est la plus adaptée à notre cas d'étude et qui permet d'aboutir à un résultat satisfaisant.

CHAPITRE 2

MÉTHODOLOGIE DE RECHERCHE

2.1 Introduction

Dans ce chapitre, on expose la démarche qu'on a suivie pour mettre en œuvre la solution proposée. L'objectif est de concevoir un modèle de maintenance prédictif destiné à identifier l'état d'une turbine de puissance en mode opérationnel. C'est à dire, de prédire si la machine fonctionne dans les conditions normales ou bien elle présente un problème de balourd.

2.2 Démarche de recherche

Pour atteindre les objectifs du projet, cette recherche s'est inspirée des travaux étudiés dans le chapitre précédent. La réalisation du projet comporte quatre étapes nécessaires:

- Une étape de contextualisation: Cette phase est essentielle pour la compréhension et la familiarisation du thème lié à notre projet de recherche.
- Une étape d'identification: Cette étape a été consacrée à l'assimilation du besoin de notre partenaire industrielle et donc à l'identification de la problématique de recherche qui consiste à établir un système de maintenance prédictive capable d'identifier à temps l'état de la turbine de puissance quant à son balancement afin d'éviter les arrêts brusques de la machine dus à un déséquilibre grave.
- Une autre étape liée à la solution proposée c'est-à-dire à la conception du modèle de prédiction et de notre méthodologie à suivre pour résoudre la problématique.
- Une dernière étape d'évaluation consacrée aux tests des modèles de prédiction via une étude de cas avec des données réelles afin de valider notre méthodologie quant à la réalisation des objectifs qu'on a fixés avec Siemens.

Le tableau 2.1 ci-dessous regroupe les 4 étapes de notre démarche de recherche avec les objectifs et les actions à suivre pour chacune des étapes.

Tableau 2.1 Démarche de recherche

Étapes	Objectifs	Action
Contextualisation	• Compréhension de la thématique et du contexte du projet de recherche.	• Effectuer des réunions avec le responsable industriel • Revue de la littérature.
Identification	• Analyse et interprétations des points constatés à la phase contextuelle, • Identification du besoin et élaboration de la problématique industrielle et revue de la littérature	• Documentation • Réunions avec le partenaire industriel
Proposition	• Solution proposée • Conception des modèles de prédiction	• Revue des techniques existants et qui peuvent répondre à notre problématique • Définition des étapes et élaboration de la méthode
Test et évaluation	• Choix de la solution qui répond au besoin de Siemens Energy	• Comparaison entre plusieurs techniques • Évaluation des résultats obtenus

2.2.1 Collecte des données

Dans la partie d'acquisition de données, avec l'avancée de la technologie de l'Internet des objets (IoT), divers types de données, également appelées données volumineuses industrielles, sont capturées à partir de l'équipement surveillé, et sont stockés pour une analyse plus approfondie. Ces données sont principalement des données de surveillance de l'état, y compris le signal de vibration. Pour le cas du partenaire industriel, l'entreprise possède son propre système d'acquisition et d'analyse des signaux vibratoire (EVA: Enhaced Vibration Analytics) (O'connor, 2021).

La vibration a trois caractéristiques mesurables: le déplacement, la vitesse et l'accélération. Bien qu'elles soient liées mathématiquement, ce sont trois caractéristiques différentes. Il est nécessaire de sélectionner une mesure de vibration et un type de capteur qui mesure la vibration la plus susceptible de révéler les caractéristiques de défaillance attendues.

L'accélération est le taux de changement de vitesse. En effet, on trouve les vibrations en termes d'accélération et qui sont mesurées avec des accéléromètres. Également, la vitesse qui est mesurée en in/sec ou mm/sec. Elle mesure le taux de changement de déplacement du signal de vibration. C'est la mesure de vibration de machine la plus courante. Historiquement, le capteur de vitesse a été l'un des premiers capteurs électriques utilisés pour la surveillance de l'état des machines.

La vibration est caractérisée aussi par le déplacement qui est mesuré en mils ou en micromètres. Il est le changement de distance ou position d'un objet par rapport à une référence. Le déplacement est généralement mesuré avec un capteur communément appelé sonde de déplacement. Une sonde de déplacement est un appareil sans contact qui mesure la distance relative entre deux surfaces. Les sondes de déplacement surveillent le plus souvent les vibrations de l'arbre ou du rotor par rapport au boîtier de la machine. Généralement, on utilise deux capteurs montés à des angles de 90° pour obtenir les axes X et Y comme montré dans la figure 2.1.

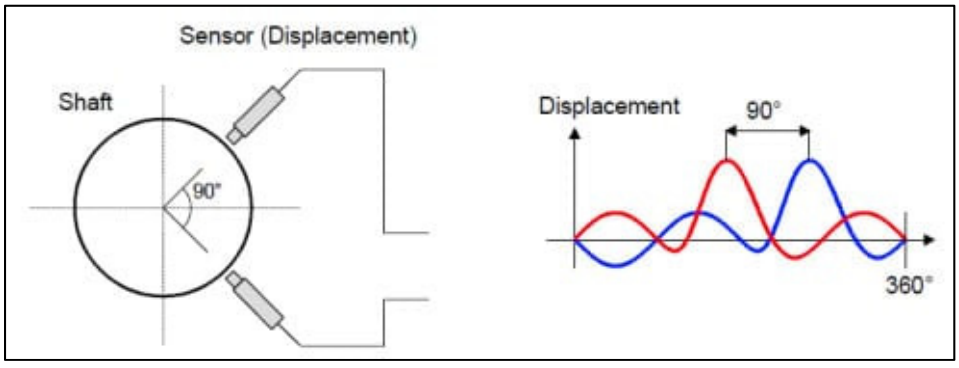


Figure 2.1 Capteurs de déplacement tiré de DataLogger Inc (2016)

On a choisi de collecter et d'analyser les mesures de déplacements pour deux raisons. Les mesures de déplacement ne sont généralement utilisées que pour examiner l'ensemble des

vibrations mécaniques. Elles sont utilisées pour détecter un déséquilibre dans une pièce en rotation en raison d'une quantité importante de déplacement aux fréquences de rotation de l'arbre de la machine. Aussi, Siemens Energy utilise des sondes de déplacement pour les mesures des vibrations.

2.2.2 Analyse des données vibratoires

L'analyse des vibrations est un processus qui surveille les niveaux de vibration et étudie les modèles de signaux de vibration. Elle est couramment menée à la fois sur les formes d'onde temporelles du signal de vibration directement, ainsi que sur le spectre de fréquence, qui est obtenue en appliquant la transformée de Fourier sur la forme d'onde temporelle. On va donc étudier ces deux techniques et les comparer.

2.2.2.1 Analyse temporelle

L'analyse du domaine temporel sur des formes d'onde de vibration enregistrée chronologiquement révèle quand et à quel point les événements de vibration anormaux se produisent en étudiant plusieurs paramètres, y compris la moyenne quadratique (RMS), l'écart type, l'amplitude de pic, le kurtosis, la crête facteur et bien d'autres. L'analyse du domaine temporel est capable d'évaluer l'état général des cibles surveillées. En fait, l'extraction des caractéristiques est le moyen de trouver et de cartographier des entités à partir des données brutes. Certaines fonctions sont utilisées pour transformer ces données en un ensemble de caractéristiques représentatives qui sont utilisées pour identifier les conditions saines et défectueuses de la machine. Les données brutes sont collectées, puis l'extraction des caractéristiques est effectuée à l'aide de l'analyse du domaine temporel qui fournit les valeurs caractéristiques pour déterminer les changements en définissant les tendances. On propose donc d’étudier ces caractéristiques pour pouvoir tirer quelques conclusions quant à l'état de la turbine.

- Skewness : L'asymétrie est une mesure de la distorsion de la distribution symétrique. Il mesure l'écart de la distribution donnée d'une variable aléatoire par rapport à une

distribution symétrique, tel que la distribution normale. Une distribution normale est sans asymétrie, car elle est symétrique des deux côtés.(CFI Education Inc).

$$Skewness = \frac{\overline{X} - M_0}{S} \tag{2.1}$$

Où:

$\overline{X}$= valeur moyenne

M_0 = valeur de mode

S = écart type des données de l'échantillon

- Valeur maximale : Le maximum est la plus grande valeur de l'ensemble de données.

- Valeur minimale : Le minimum est la plus petite valeur de l'ensemble de données.

- Valeur quadratique moyenne des valeurs instantanées dans une certaine durée (RMS) : il s'agit de la puissance de l'onde. Dans le domaine temporel, la valeur RMS est la méthode la plus basique utilisée pour détecter les défauts dans la machine tournante, mais elle n'est pas capable de détecter les défauts lorsque le problème est à un stade précoce. La valeur RMS est très utile pour détecter un déséquilibre dans les machines tournantes.

$$RMS = \sqrt{\frac{1}{N} * \sum_{i=1}^{N} x_i(i)^2} \tag{2.2}$$

- Facteur de crête (CF): indique une relation entre la valeur de crête du signal et la valeur RMS indiquant les premiers signes de dommages, en particulier lorsque les signaux de vibration ont des caractéristiques impulsives. Il est utilisé pour déterminer la détérioration des roulements par comparaison relative.

$$CP = \frac{max(x_i(i))}{V_{RMS(x_i(i))}} \tag{2.3}$$

- Le Kurtosis : Le kurtosis est une mesure de l'aplatissement, et c'est donc un bon indicateur de l'impulsivité du signal dans le contexte de la détection de défauts pour les composants rotatifs. Il est exprimé comme suit :

$$Kurtosis\ (x) = \frac{E\{(x-\mu)^4\}}{\delta^4} - 3 \tag{2.4}$$

Où μ et sont respectivement la moyenne et l'écart type des séries temporelles *x*, tandis que *E* est l'opération d'espérance. Le « moins 3 » à la fin de cette formule doit rendre le kurtosis de la distribution normale égal à zéro.

- Écart type (STD) : L'écart type est la racine carrée positive de la variance pour mesurer la variation des données de l'équation.

$$STD = \sqrt{\frac{1}{L} * \sum_{i=1}^{L} |y_i - \mu|^2} \tag{2.5}$$

Où μ est la moyenne et *L* est la taille de l'échantillon.

2.2.2.2 Analyse fréquentielle

Dans l'analyse du domaine fréquentiel, les vibrations peuvent être caractérisées par l'amplitude et la fréquence. L'amplitude représente la force du signal et la fréquence représente le taux d'oscillation. Pour la détection des défauts, on va effectuer la FFT (Fast Fourier Transformation) du signal. Au lieu d'observer les données dans le domaine temporel, avec des analyses mathématiques de fréquence, on décompose les données temporelles en séries d'ondes sinusoïdales. On peut aussi dire que l'analyse fréquentielle vérifie la présence de certaines

fréquences fixes. On va donc vérifier cette propriété afin de détecter les fréquences liées au défaut du balourd dont la vibration devrait se manifester à la fréquence de rotation Fr. Elle représente alors le pic le plus élevé avec des pics d'amplitudes plus faibles sur les harmoniques de Fr comme montré dans la figure 2.2 ci-dessous.

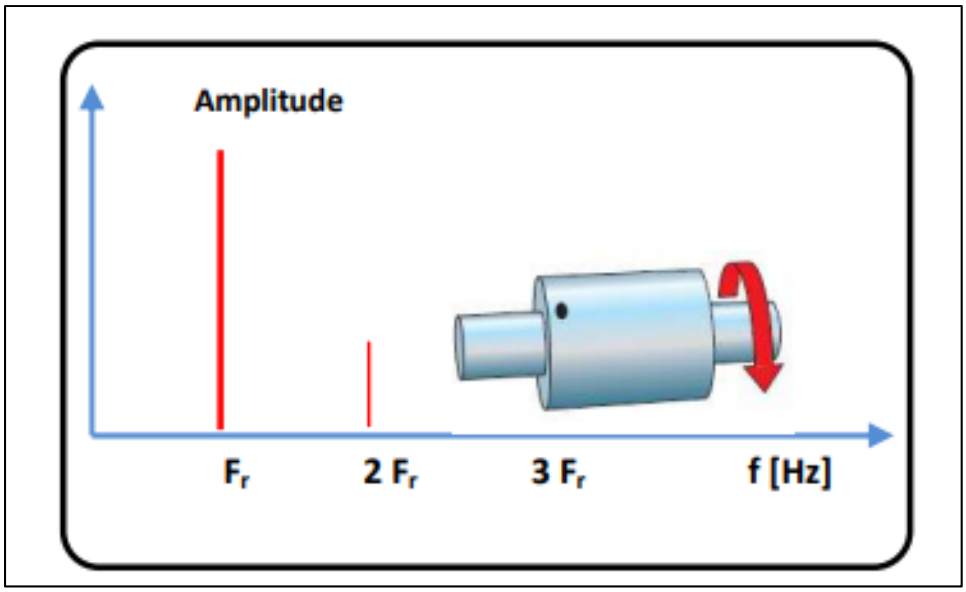


Figure 2.2 Spectre théorique d'un défaut de balourd
tirée de Landolsi (s.d.)

2.2.3 Prétraitement des données

L'utilisation de techniques de prétraitement des données rend la base de données plus complètes et plus précises. En prétraitant les données, on en facilite l'interprétation et l'utilisation. Ce processus élimine les incohérences ou les doublons dans les données, qui peuvent autrement affecter négativement la précision d'un modèle. Le prétraitement des données garantit également qu'il n'y a pas de valeurs incorrectes ou manquantes en raison d'une erreur humaine ou autre.

2.3 Détection du défaut en utilisant l'apprentissage automatique

Dans ce travail, une étude comparative a été faite pour identifier la performance des fonctionnalités avec les classificateurs RF, KNN et SVM. Ces classificateurs sont largement

utilisés dans la littérature pour diagnostiquer les défauts des machines rotatives notamment le SVM qui semble être un bon classificateur éprouvé par rapport à d'autres classificateurs conventionnels. L'outil de programmation utilisé est « *Pycharm ».* C'est un environnement de développement intégré utilisé pour la programmation en Python. On a choisi « *Pycharm* » parce qu'il est le plus recommandé pour la programmation Python. Il permet l'analyse de code et contient un débogueur graphique. Il permet également la gestion des tests unitaires, l'intégration de logiciels de gestion de versions, prend en charge le développement Web avec Django et dispose de Data Science avec Anaconda.

Les bibliothèques Python utilisées durant ce projet sont les suivants (voir ANNEXE I):

- **NumPy** est un module Python numérique. Il fournit des fonctions mathématiques rapides. Utilisé pour travailler avec des tableaux et il a également des fonctions pour travailler dans le domaine de l'algèbre linéaire, de la transformée de Fourier et des matrices.
- **Pandas** fournit des structures de données robustes pour un calcul efficace de tableaux et de matrices multidimensionnels. On a utilisé pandas pour lire et manipuler les fichiers de données dans des tableaux.
- **Scikit-Learn** qui est une bibliothèque d'apprentissage automatique et comprend divers algorithmes utilisés dans l'apprentissage machine. Les fonctions suivantes sont employées:
    - train_test_split,
    - KNeighboursClassifier, SVC, RandomForestClassifier, LinearSVC,
    - Les algorithmes d'évaluation des modèles: f1_score, exactitude_score, Rappel_score, plot_precision_Rappel_curve, confusion_matrix., precision_score,
    - MinMaxScaler pour la normalisation et la standardisation des données,
    - GridSearchCV pour la recherche des paramètres optimaux.

L'apprentissage machine comprend plusieurs étapes à savoir : la collecte de données, le prétraitement, la construction des modèles, la comparaison des modèles et enfin l'évaluation. Le processus de recherche qu'on a suivi est illustré par la figure 2.3.

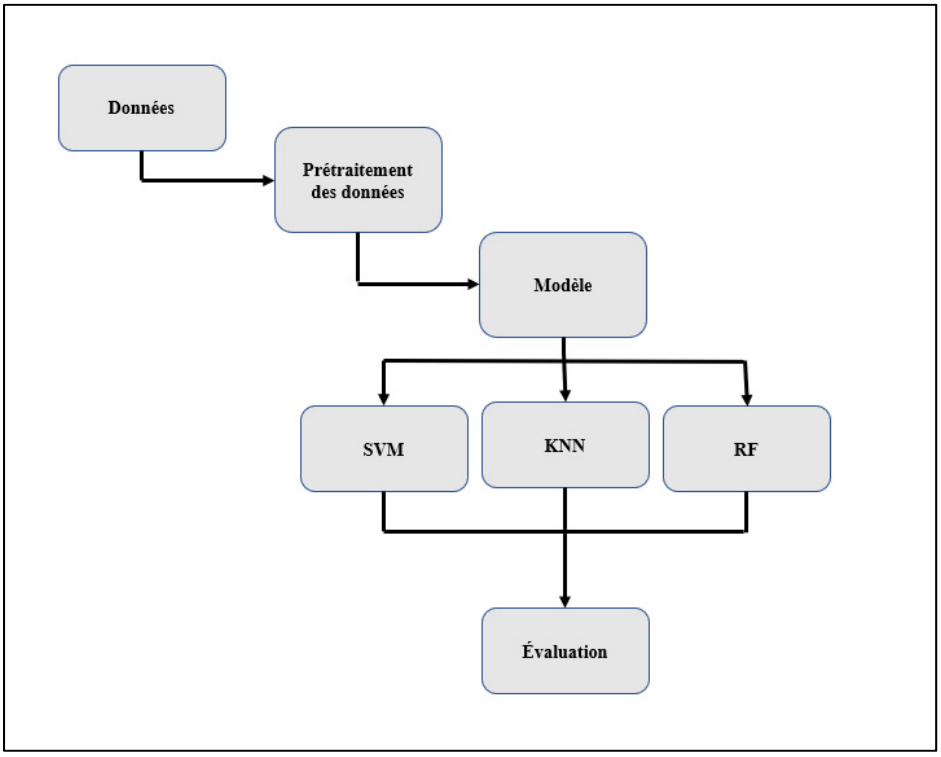


Figure 2.3 Étape de développement du modèle

2.4 Conclusion

Dans ce chapitre, on a expliqué la méthodologie adoptée pour la réalisation de ce projet. Pour résumer, la première phase primordiale est de comprendre le contexte du projet ainsi que la problématique ce qui va faciliter le développement de la méthodologie. Ensuite, il faut commencer à collecter les données adéquates pour bien mener le projet. Par la suite, on propose de comparer entre plusieurs approches utilisées dans la maintenance prédictive et la détection des défauts dans les machines rotatives, à savoir l'analyse temporelle, l'analyse fréquentielle et en fin le développement du modèle prédictif. Pour cela on ferait l'étude du SVM, du KNN et la RF.

CHAPITRE 3

DIAGNOSTIC DE LA TURBINE DE PUISSANCE AVEC L'ANALYSE TEMPORELLE ET FRÉQUENTIELLE

3.1 Introduction

Dans ce chapitre, on va mettre en pratique les approches adoptées pour détecter les défauts dans les machines rotatives en utilisant une base de données réelle d'une turbine de puissance, fournie par le partenaire industriel Siemens Energy.

On va commencer par exposer la machine étudiée ainsi que les outils utilisés dans l'extraction des signaux vibratoires. Ensuite, on va présenter et visualiser les données qu'on dispose dans le but de savoir si les signaux nécessitent un prétraitement avant l'analyse. Par la suite, on va appliquer les anciennes techniques d'analyse vibratoire à savoir l'analyse temporelle et l'analyse fréquentielle dans le but d'étudier son état. À la fin de ce chapitre, on tire quelques conclusions à propos de ces méthodes.

3.2 Machine étudiée

La machine étudiée est une turbine à gaz industrielle à écoulement axial dans la bande de puissance 6-7 MW (voir figure 3.1). Elle est une unité éprouvée pour la production d'électricité, y compris la cogénération et l'entrainement de charge mécaniques, la compression et le pompage pour une utilisation dans les secteurs de la production d'électricité industrielle du pétrole et du gaz (Siemens Industrial, 2005).

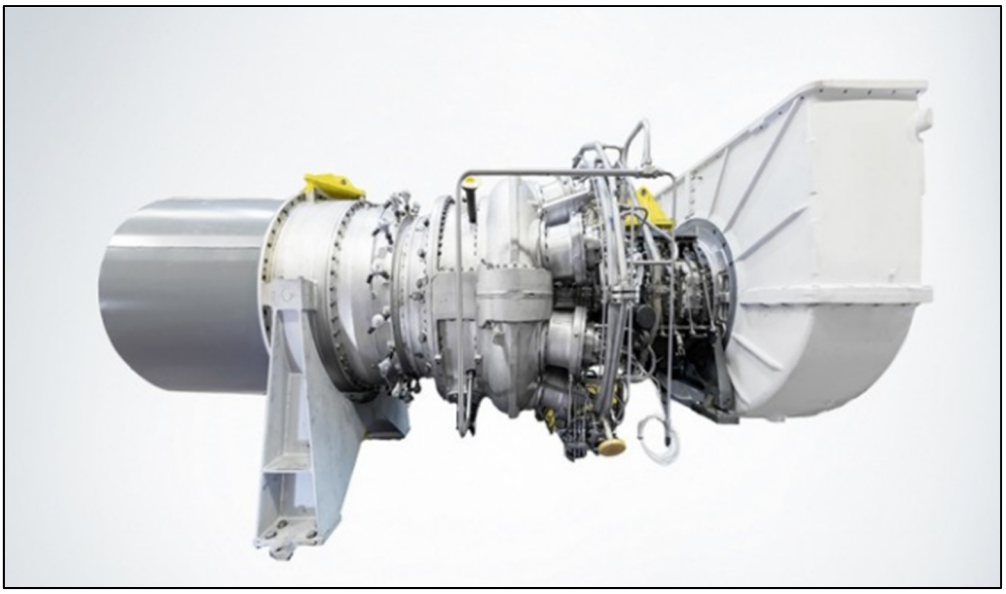


Figure 3.1 Turbine à gaz tirée de Siemens Industrial (2005)

3.3 Description des données

Comme mentionné dans le Chapitre 3, les données utilisées dans cette étude sont fournies par Siemens Energy et ils étaient collectées directement sur la machine SGT -200. Toutes les mesures ont été prises par des capteurs de déplacement. Les données acquises sont l'amplitude de la vibration (en μm) en fonction du temps (en secondes). En fait, le système rotatif étudié comprend un générateur de gaz, une turbine de puissance et le couplage Générateur de gaz/Turbine de puissance. Pour prendre les mesures nécessaires dans ce travail, des capteurs de déplacement ont été placés à l'entrée et la sortie du générateur de gaz (Capteur 10 et 11) et à l'entrée et à la sortie de la turbine (12 et 13). La figure 3.2 illustre les emplacements des capteurs utilisés.

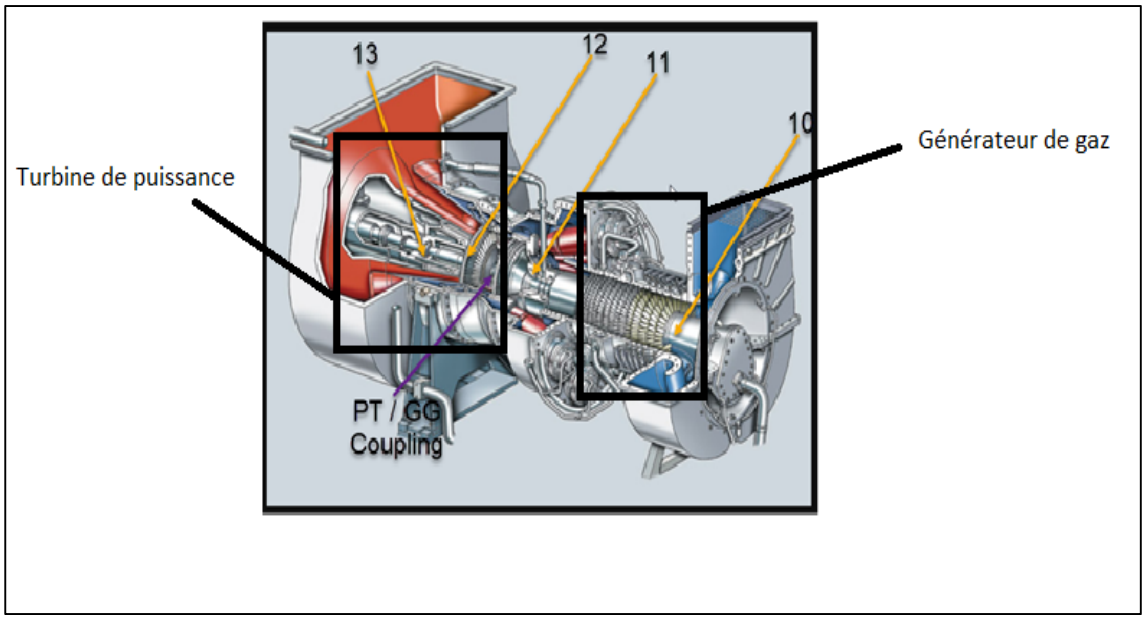


Figure 3.2 Les points de mesure des données vibratoires adaptée de Siemens Industrial (2005)

Les tableaux ci-dessous décrivent les sondes de déplacement utilisées ainsi que leurs positions de part et d'autre sur la machine. Le Tableau 3.1 correspond aux sondes au niveau du gaz générateur. Le tableau 3.2 correspond aux sondes au niveau de la turbine.

Tableau 3.1 Les capteurs utilisés au niveau du générateur de gaz

Générateur de gaz			
Canal	**Nom**	**Sonde**	**Position**
1	UD10X	GG Inlet X	45° gauche
2	UD10Y	GG Inlet Y	45° droite
3	UD11X	GG Exit X	127° droite
4	UD11Y	GG Exit Y	143° gauche

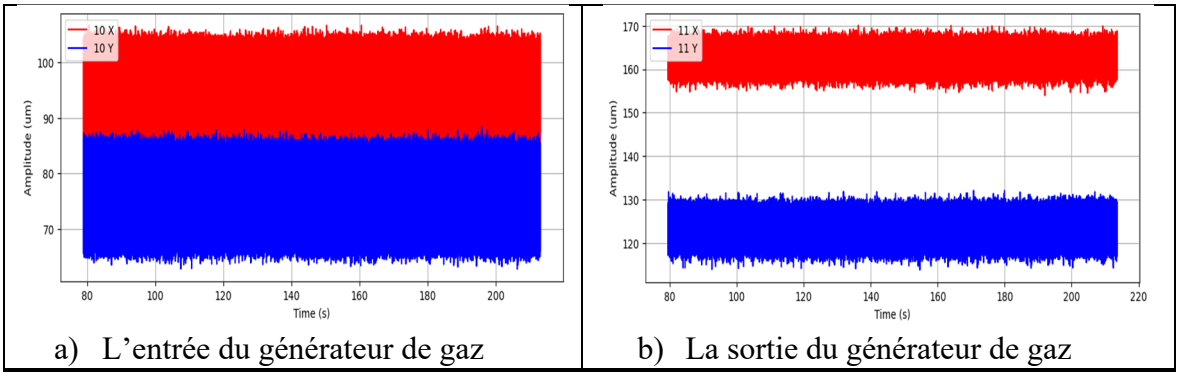
Tableau 3.2 Les capteurs utilisés au niveau de la turbine de puissance

Turbine de puissance			
Canal	**Nom**	**Sonde**	**Position**
5	UD12X	PT Inlet X	45° gauche
6	UD12Y	PT Inlet Y	45° droite
7	UD13X	PT Exit X	35° gauche
8	UD13Y	PT Exit Y	55° droite

Au cours de l'acquisition des données, des signaux de vibration consécutifs ont été enregistrés avant l'équilibrage et après l'équilibrage de la machine à une vitesse de rotation égale à 11067.5 RPM. Chaque signal dure T=84 secondes et est enregistré avec une fréquence d'échantillonnage de 12.5 kHz et on possède 8 capteurs qu'on a mentionnés précédemment. Ainsi, le nombre total des points analysés est égal à 83.88576*12 500 =1 048 572 * 8 points.

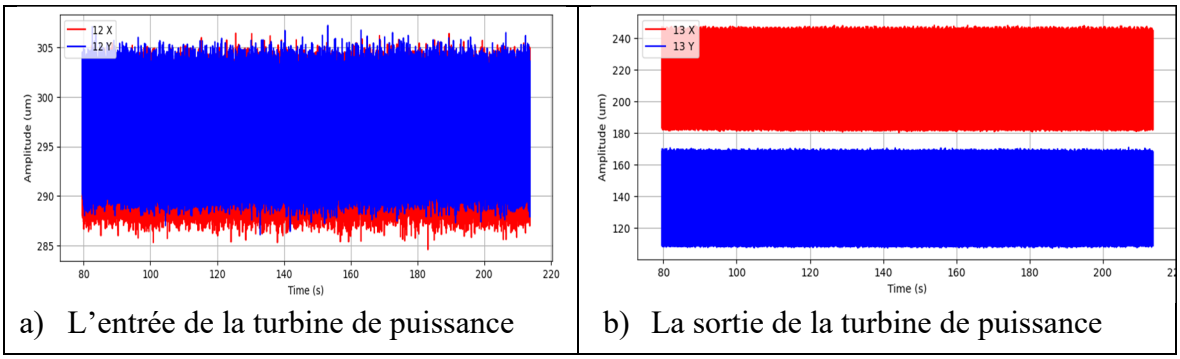
3.4 Visualisation et traitement des données

Les tracés de la forme d'onde temporelle ci-dessus illustrent comment le signal d'une sonde de déplacement apparaît lorsqu'il est représenté graphiquement en amplitude au fil du temps. Cependant, on remarque que ces formes d'onde nécessitent un peu de traitement pour être bien interprétées comme le filtrage du bruit. De plus, comme on peut le voir sur les figures 3.3 et 3.4, les formes d'onde obtenues représentent la vibration en tant que deux paramètres importants AC et DC. La composante AC représente la vibration tandis que la composante DC représente à quelle distance la moyenne de l'AC est de zéro. Mais lorsque l'on regarde les vibrations pures, on n'a besoin que du contenu AC et donc on doit soustraire les valeurs moyennes pour retrouver les vibrations pures pour pouvoir les exploiter correctement. On a appliqué aussi le filtrage de bruit pour lisser les données. Pour cela, le débruitage par ondelettes a été appliqué à nos signaux (Çiğdem & Mehmet, 2018), et comme montré sur l'exemple de la figure 3.5.



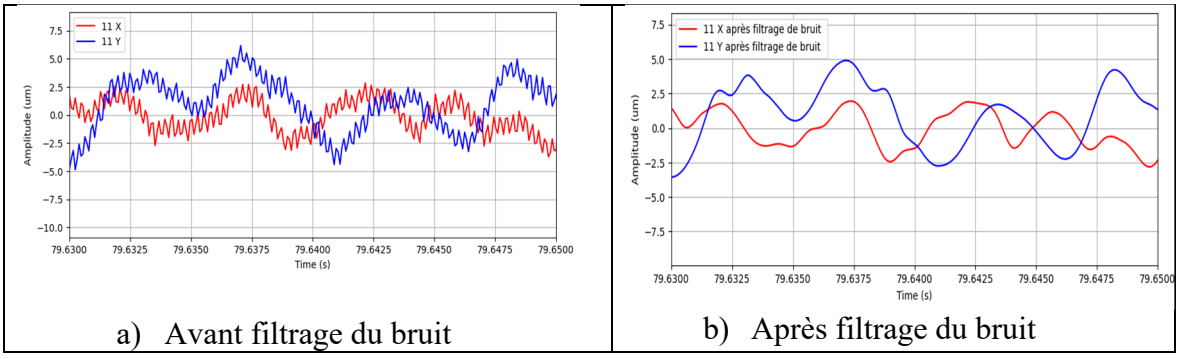
a) L'entrée du générateur de gaz b) La sortie du générateur de gaz

Figure 3.3 Visualisation des données temporelles avant traitement dans une machine avec défaut



a) L'entrée de la turbine de puissance b) La sortie de la turbine de puissance

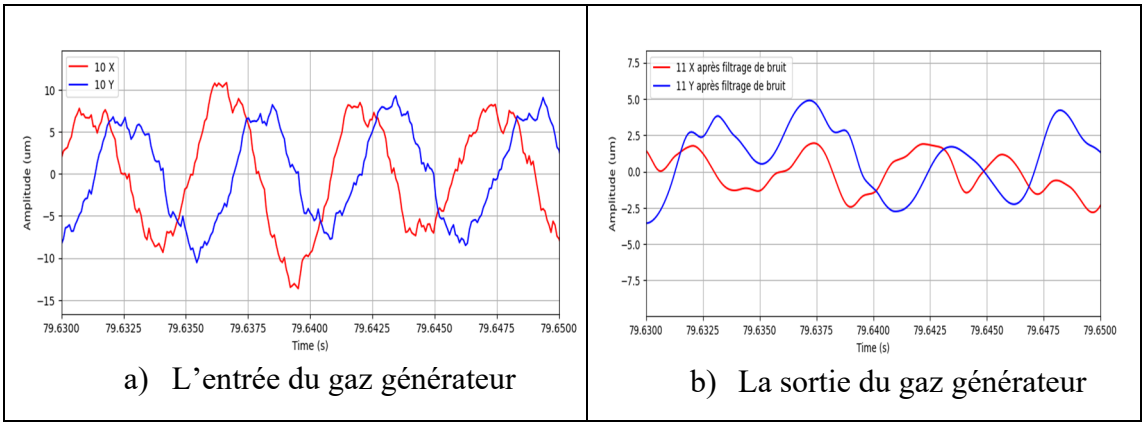
Figure 3.4 Visualisation des données temporelles avant traitement des données dans une machine avec défaut



a) Avant filtrage du bruit b) Après filtrage du bruit

Figure 3.5 Signal bruité et signal filtré en appliquant le débruitage par ondelettes

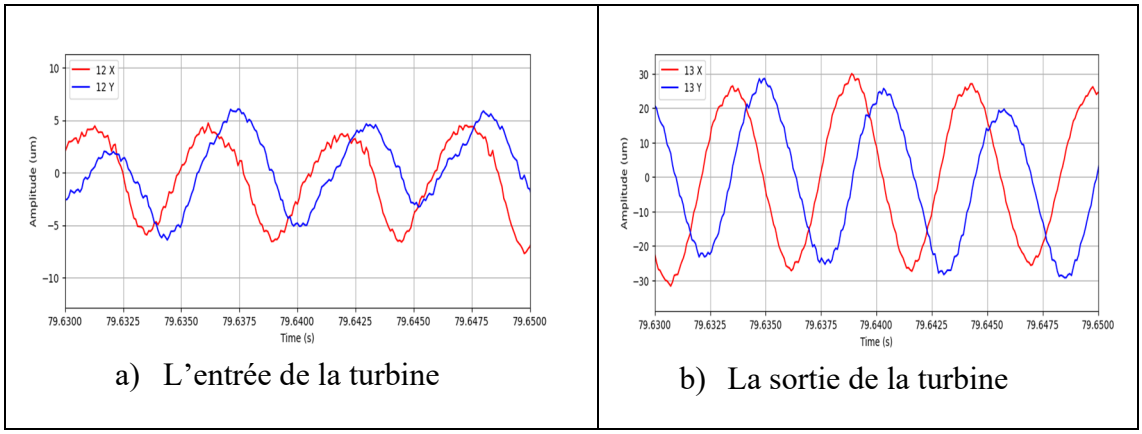
L'algorithme de débruitage de base comporte trois étapes. En effet, en partant du signal, on le décompose d'abord sur une base d'ondelettes orthogonales en utilisant la transformée discrète. Ensuite, on sélectionne une partie du coefficient par seuillage et gardons intacts les coefficients d'approximation d'un niveau convenablement choisi. Enfin, à l'aide de coefficients seuillés, on reconstruit un signal en leur appliquant la transformée discrète inverse. Le signal obtenu est le plus lisse. Cet algorithme est appliqué par la fonction « *wavelet denoise* » qui est une fonction intégrée dans la bibliothèque Python « *Wand ImageMagick* » pour supprimer le bruit en appliquant une transformation en ondelettes. La figure 3.6 représente les signaux collectés au niveau du générateur de gaz après avoir effectué le traitement nécessaire. La figure 3.7 représente les signaux collectés au niveau de la turbine de puissance après avoir effectué le traitement nécessaire.



a) L'entrée du gaz générateur

b) La sortie du gaz générateur

Figure 3.6 Visualisation des données du générateur de gaz après traitement pour une machine avec défaut.



a) L'entrée de la turbine

b) La sortie de la turbine

Figure 3.7 Visualisation des données de la turbine après pour une machine avec défaut.

De la même manière, ce traitement a été appliqué à tous les signaux recueillis avant et après le balancement de l'équipement. Ce type de tracé de vibration est également appelé tracé ou graphique du domaine temporel. Les formes d'onde temporelles affichent un échantillon de courte durée de la vibration brute. Bien qu'elle ne soit généralement pas aussi utile que d'autres formats d'analyse, l'analyse de la forme d'onde temporelle peut fournir des indices sur l'état de la machine qui n'est pas toujours évidente dans le spectre de fréquences et, lorsqu'elle est disponible, doit être utilisée dans le cadre de notre programme d'analyse. Dans la section qui suit, on va montrer comment ces formes d'onde peuvent nous renseigner sur l'état de la machine.

3.5 Analyse temporelle

Pour réaliser une étude comparative entre les comportements vibratoires des deux états de la machine, on a choisi de tracer les deux signaux dans le même graphe. En effet, la figure 3.8 illustre le comportement vibratoire de la turbine de puissance avant et après son équilibrage.

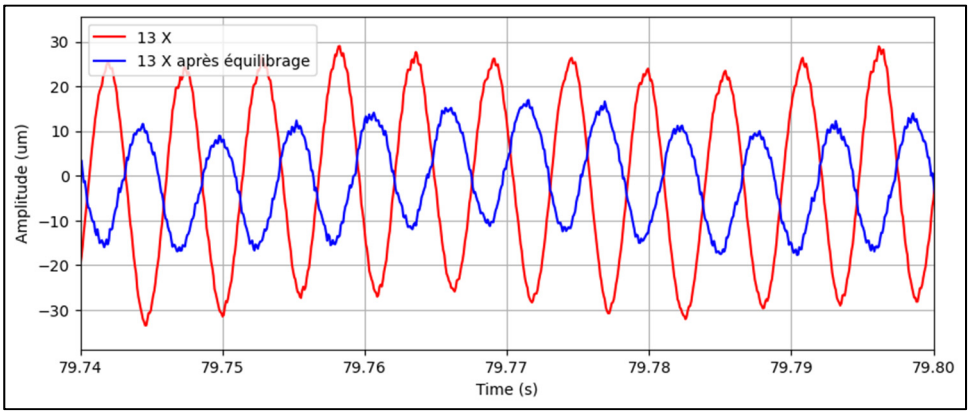


Figure 3.8 Les signaux vibratoires avant et après équilibrage

D'après la figure 3.8, il est clair qu'apr l'équilibrage de la turbine, les amplitudes des vibrations ont baissé considérablement. En effet, plus l'amplitude est élevée, plus la vibration est élevée, plus le problème est important. L'amplitude de la vibration peut donc être un bon révélateur de la condition de la machine et la gravité du défaut. Il existe plusieurs autres caractéristiques qu'on peut extraire du signal temporel.

3.5.1 Extraction des descripteurs statistiques

Plusieurs recherches utilisent diverses caractéristiques du domaine temporel pour identifier et caractériser les défauts dans les machines rotatives, telles que : la moyenne quadratique (RMS), l'écart type (SD), l'aplatissement (KU), et la moyenne (MEAN). En effet, l'extraction des caractéristiques décrit en fait l'état de la machine, ils ont un rôle important dans la classification des défauts en introduisant un ensemble de descripteurs statistiques comme entrées dans le système de classification pour l'auto-reconnaissance des défauts (Jaouher, Nader, Saidi, & al, 2015). On veut donc étudier ces caractéristiques et voir si on peut identifier le balourd de la turbine. Toutes les caractéristiques inscrites dans les tableaux ci-dessous ont été calculées en langage Python (Tableau 3.3 et Tableau 3.4).

Tableau 3.3 Caractéristiques temporelles pour un état normal

Caractéristique	10 X	10 Y	11 X	11 Y	12 X	12 Y	13 X
Kurtosis	-1,23539	-0,93390	-0,93390	-0,16407	-1,01649	-1,37799	-1,31390
RMS	4.83	2.95	4.11	1.71	4.243	5.75	9.90
Skewness	-0.117	0,16522	0.14	-0,09769	-0,19925	-0,00201	-0,04179
Moyenne	-0.132	-0.056	0.055	-0.0118	-0.0311	-0.057	-0.37
Min	-13.89	-8.22	-11.27	-7.56	-12.00	-11.98	-20.37
Max	10.76	9.30	11.15	6.95	10.33	13.57	19.12
Crest Factor	2.22	3.14	2.70	4.04	2.43	2.35	1,91
Form Factor	-36.64	-52.54	-74.63	-145.63	-136.394	-100.07	-26.66

Tableau 3.4 Caractéristiques temporelles pour un état de déséquilibre

caractéristique	10 X	10 Y	11 X	11 Y	12 X	12 Y	13 X
Kurtosis	-1,41560	-1,38406	-0,11408	-0,45391	-1,03450	-0,93056	-1,46357
RMS	6.34	5.42	5.86	2.39	3.86	3.38	19.49
Skewness	-0,01954	-0,009	-0,07841	-0,22090	-0,21591	-0,12345	-0.055
Moyenne	-0.21	-0.19	-0.010	-0.0231	-0.025	-0.019	-0.88
Min	-15.24	-13.49	-8.02	-9.95	-11.00	-10.64	-35.48
Max	13.26	11.59	7.38	8.04	10.07	10.22	32.30
Crest Factor	2.09	2.13	3.86	3.36	2.60	1,03438	1,65966
Form Factor	-29.47	-28.12	-174.74	-103.43	-153.33	-175.92	-22.14

À partir des tableaux ci-dessus, on a constaté qu'en effet certaines caractéristiques présentent une différence entre l'état normal et anormal notamment le RMS, la valeur maximale. Ces caractéristiques peuvent détecter la présence d'une anomalie.

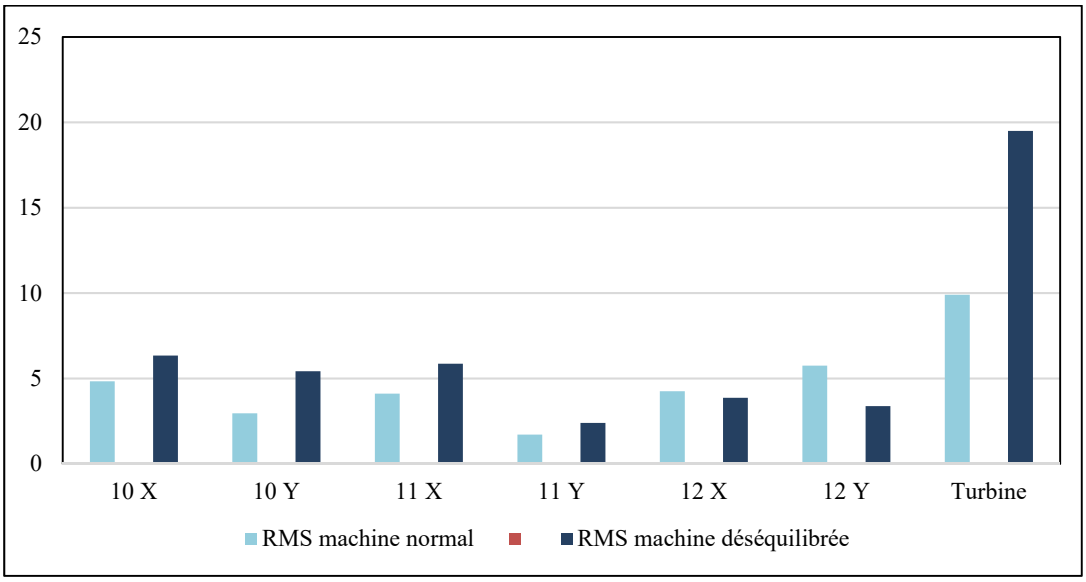


Figure 3.9 Comparaison de RMS entre machine normale et machine déséquilibrée

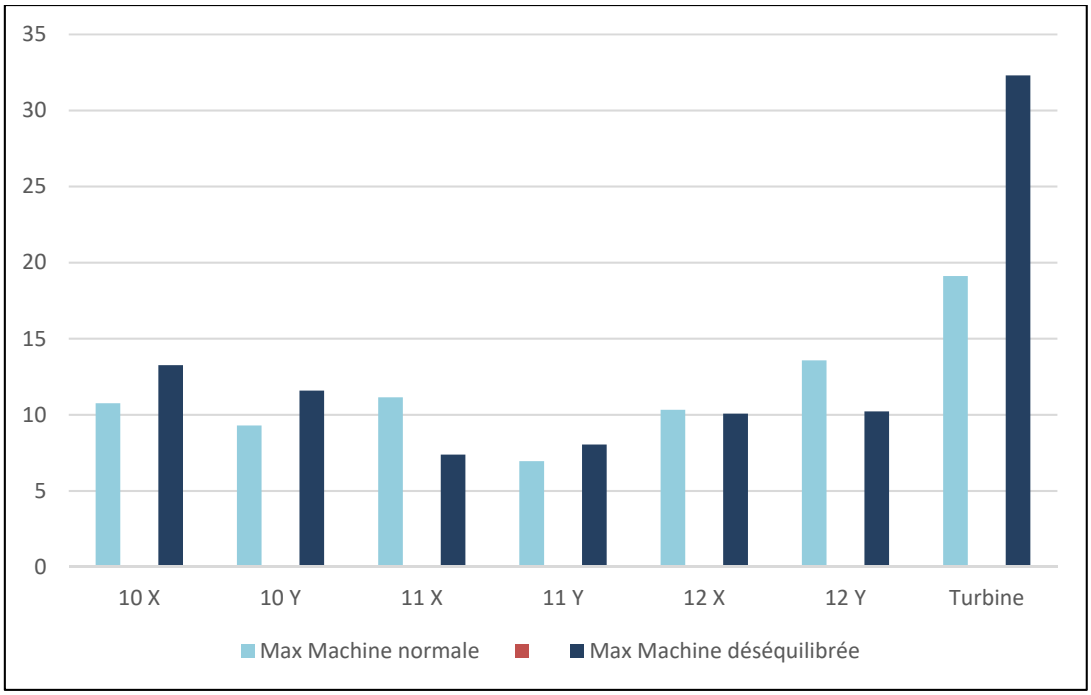


Figure 3.10 Comparaison de la valeur maximale entre machine normale et machine déséquilibrée

3.5.2 Discussion

Les deux histogrammes montrés précédemment illustrent bien la différence entre l'état normal et l'état anormal de la turbine surtout au niveau de la moyenne quadrique RMS. En effet c'est un indicateur couramment utilisé, car il est directement lié au niveau d'énergie du signal de vibration et il fournit un moyen cohérent de mesurer les vibrations causées par le déséquilibre, le désalignement et d'autres problèmes (Donovan, 2019). Cependant, il n'est pas capable de détecter les défauts lorsque le problème est à un stade précoce (Manish Vishwakarma, 2017). Également, ces caractéristiques ne peuvent pas identifier ou détecter le type de défaut.

3.6 Analyse fréquentielle

Théoriquement, l'effet du déséquilibre pur fait osciller le système d'une manière harmonique qui se manifeste comme une onde sinusoïdale dans le domaine temporel et un seul pic dans le domaine fréquentiel à la fréquence de rotation de la machine (Desouki, Sassi, Renno, & Gowid,

2020). Afin de vérifier cela, on va transformer les signaux temporaux de la machine équilibrée et de la machine présentant un balourd au domaine fréquentiel en effectuant la transformée de Fourier (FFT) à l'aide de la fonction FFT de la bibliothèque « *Scipy* » en Python. Les figures ci-dessous illustrent le résultat obtenu.

Le premier spectre (voir figure 3.13) représente la FFT de la machine sujette d'un déséquilibre et le deuxième (figure 3.14) correspond à celui de la machine dans son état normal après avoir effectué l'action de balancement. Rappelons que la vitesse de rotation de la machine est égale 11 067 RPM comme le montre la figure 3.11. Ce qui correspond à 184.45 Hz fréquence de rotation de la machine.

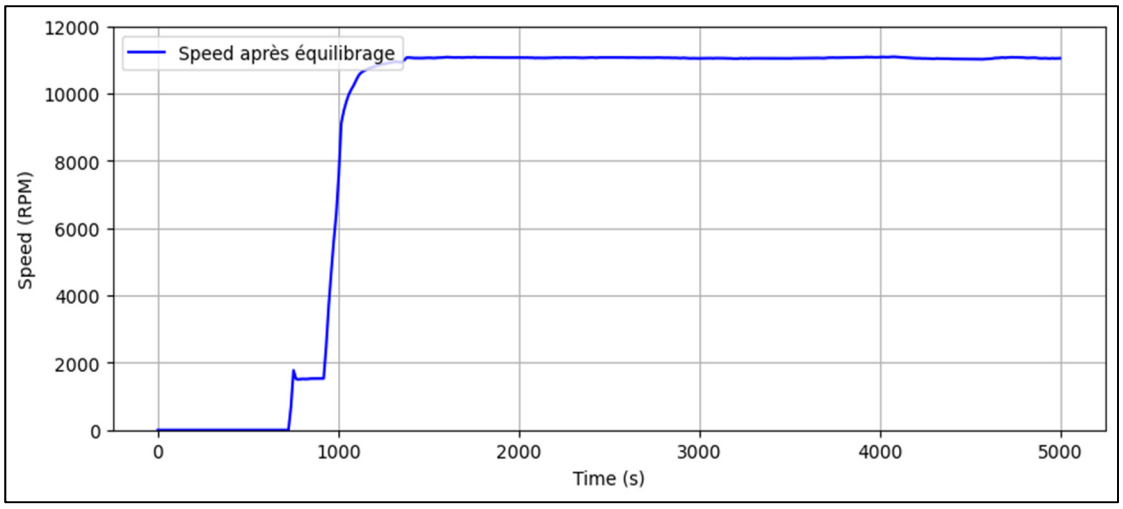


Figure 3.11 Vitesse de rotation de la machine (11067 RPM)

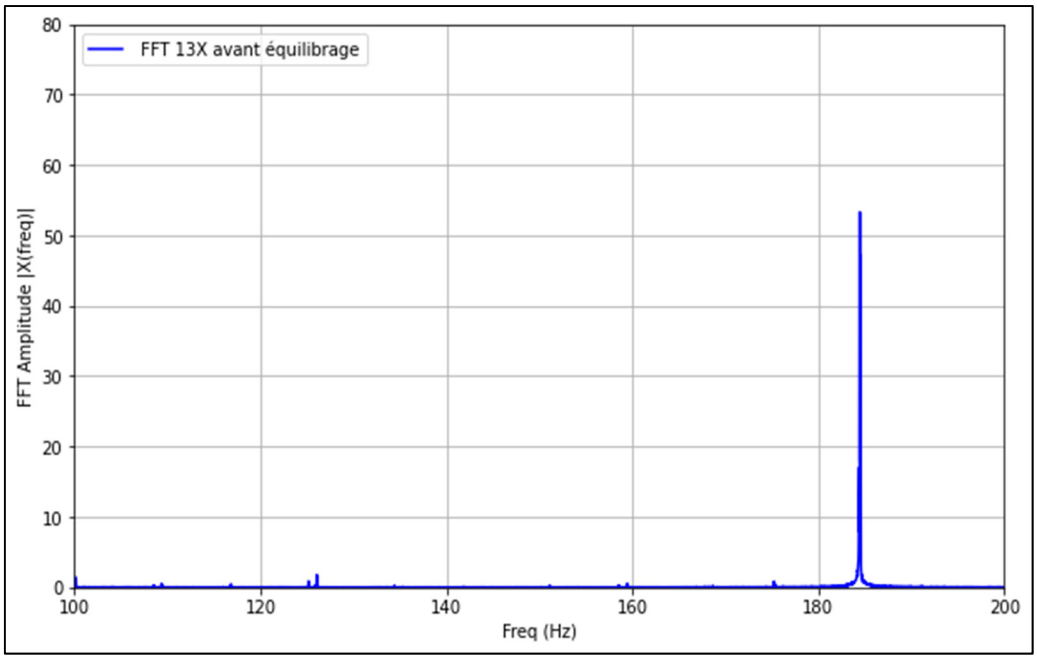


Figure 3.12 FFT du signal vibratoire de la turbine avec défaut de balourd

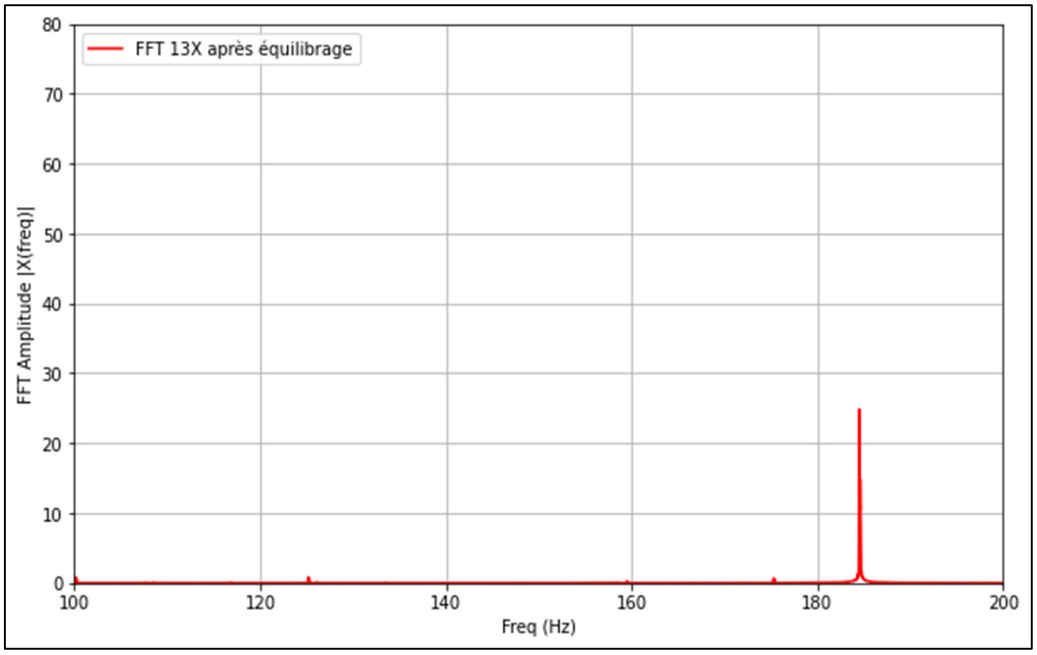


Figure 3.13 FFT du signal vibratoire de la turbine de puissance à son état normal

L'analyse du premier spectre montre qu'il y a un pic à la 1X fréquence de rotation de la machine (184.45 Hz) et d'amplitude égale 54 µm, alors que le deuxième spectre, qui représente la machine après équilibrage, présente un pic à la même fréquence du signal 184.45 Hz, mais d'amplitude égale à 25 µm. Donc, l'amplitude de la vibration a considérablement baissé après l'équilibrage.

De ce fait, on peut conclure que l'énergie de la vibration au point de fréquence fondamentale est la plus forte lorsque le défaut de déséquilibre du rotor apparaît. La fréquence caractéristique du défaut de vibration causée par un déséquilibre est la même que la fréquence fondamentale et cette conclusion est conforme à la caractéristique de vibration de déséquilibre des machines tournantes. Ainsi, on peut dire que l'analyse fréquentielle des vibrations peut en effet identifier la source des vibrations puisque chaque défaut a sa propre fréquence caractéristique.

3.7 Conclusion

Dans ce chapitre, on a expérimenté deux techniques de surveillance et diagnostic des défauts dans les machines rotatives à savoir l'analyse des vibrations dans le domaine temporel et l'analyse des vibrations dans le domaine fréquentiel. Cette étude comparative entre les deux techniques nous a permis de tirer quelques conclusions. En effet, pour l'analyse temporelle, on a pu confirmer que cette technique permet en effet de détecter un problème dans le fonctionnement de la machine rotative en surveillant les niveaux de vibrations. L'analyse temporelle peut même classer les défauts par leurs gravités selon les amplitudes des vibrations. Plus le défaut est grave, plus l'amplitude est grande. Cette méthode de diagnostic est capable aussi de définir des seuils acceptables et non acceptables des vibrations de fonctionnement en référant à des normes ou bien en exploitant l'historique des pannes à long terme et créer des alarmes quand les vibrations de la machine dépassent le seuil acceptable. Cependant, la limite de l'analyse temporelle est qu'elle ne permet pas d'identifier les problèmes et la cause des vibrations. De même, les signaux de vibration dans le domaine temporel ont certaines limites pour détecter la génération précoce de défauts.

En ce qui concerne l’analyse fréquentielle, cette technique peut être bonne puisqu’elle permet non seulement de détecter l’existence d’un défaut, mais aussi d’identifier la cause en surveillant les fréquences caractéristiques des défauts. Contrairement à la forme d’onde temporelle. L’analyse fréquentielle permet aussi d'évaluer l'intensité des défauts en fonction de l'amplitude à l’aide de la FFT, outil fondamental de toute analyse vibratoire. Ceci permet de confirmer que les caractéristiques du domaine fréquentiel sont généralement plus efficaces pour détecter les défauts que les caractéristiques du domaine temporel. En outre, étant donné que la fréquence caractéristique des vibrations de défaut des machines tournantes est liée à la fréquence fondamentale du rotor, on peut obtenir la raison du défaut en effectuant une analyse du spectre fréquentiel. En effet, l'analyse fréquentielle joue un rôle important dans la détection et le diagnostic des défauts de la machine. Dans le domaine temporel, les contributions individuelles (par ex. balourd, engrenages, etc.) à la vibration globale de la machine sont difficiles à reconnaitre. Dans le domaine fréquentiel, elles deviennent beaucoup plus faciles à identifier et peuvent donc être facilement mises en relation avec des sources individuelles de vibration.

CHAPITRE 4

DÉTECTION DU DÉFAUT DE BALOURD PAR DES TECHNIQUES D'APPRENTISSAGE MACHINE

4.1 Introduction

L'apprentissage automatique est une composante importante du domaine de la science des données. Grâce à l'utilisation de méthodes statistiques, plusieurs algorithmes sont formés pour effectuer des classifications ou des prédictions. En effet, dans ce chapitre, on compare trois classificateurs supervisés différents pour classer l'état de la machine et répondre au besoin de Siemens Energy qui consiste à développer un modèle capable de détecter si la machine, la turbine de puissance, fonctionne dans ses conditions normales ou présente un défaut de déséquilibre.

Dans la première partie du chapitre, on expose la méthodologie qu'on a adoptée lors du développement du modèle. Ensuite, dans la deuxième section, on applique la méthodologie en expliquant les algorithmes de l'apprentissage machine utilisés et en fin on expose les résultats obtenus. Dans tout ce qui suit, la classe 1 caractérise les l'état normal de la machine et la classe 2 caractérise l'état de déséquilibré de la machine.

4.2 Étapes de construction des modèles

Dans cette étude, nous avons étudié trois algorithmes d'apprentissage machine, à savoir le KNN, le SVM, et le RF. Les outils utilisés sont le langage Python et la bibliothèque « *Scikit-learn* » qui est une bibliothèque d'apprentissage machine. Elle comporte divers algorithmes de classification, de régression et d'agrégation, et est conçue pour interagir avec les bibliothèques numériques et scientifiques Python comme « *NumPy* » et « *SciPy »* qui sont aussi utilisés dans le code développé (voir ANNEXE I).

Comme mentionné dans le Chapitre 3, le développement d'un modèle d'apprentissage machine repose sur plusieurs étapes. Ainsi, dans notre développement on a suivi les étapes suivantes.

La collecte de données en combinant toutes les données en une seule avec les mêmes attributs. Les données utilisées sont les mesures issues des capteurs: 10X 10Y (les données mesurées à l'entrée du générateur de gaz), 11X 11Y (les données mesurées à la sortie du générateur de gaz), 12X 12Y (les données mesurées à la turbine de puissance), 13X (les données mesurées à la sortie de la turbine). Pour cela, un code Python (voir ANNEXE II) a été développé. Ce code lit les fichiers (*.csv) et les combine en une seule table de données.

Le prétraitement des données est effectué pour améliorer les données avant de créer un modèle d'apprentissage automatique. En effet, la plupart du temps, les données ont des ordres de grandeur différents et parfois même des valeurs manquantes. Ces problèmes peuvent impacter négativement les performances du modèle. Pour pallier cela, on a effectué quelques traitements préparatoires sur les données tels que la suppression des valeurs nulles et la mise à l'échelle qui comprend la standardisation et la normalisation. On a aussi rajouté une colonne « *Label* » qui prend la valeur « 1 » pour dire que la machine est en état normal et « 2 » pour dire que la machine est en état de déséquilibre. Le prétraitement inclut également la division des données en données d'apprentissage (80%) et en données de test (20%). Les données d'entrainement sont utilisées pour entrainer les modèles. Ces derniers seront testés par la suite à l'aide des données de test pour déterminer l'exactitude de la prédiction. La figure 4.1 résume les étapes suivies avant la construction des modèles.

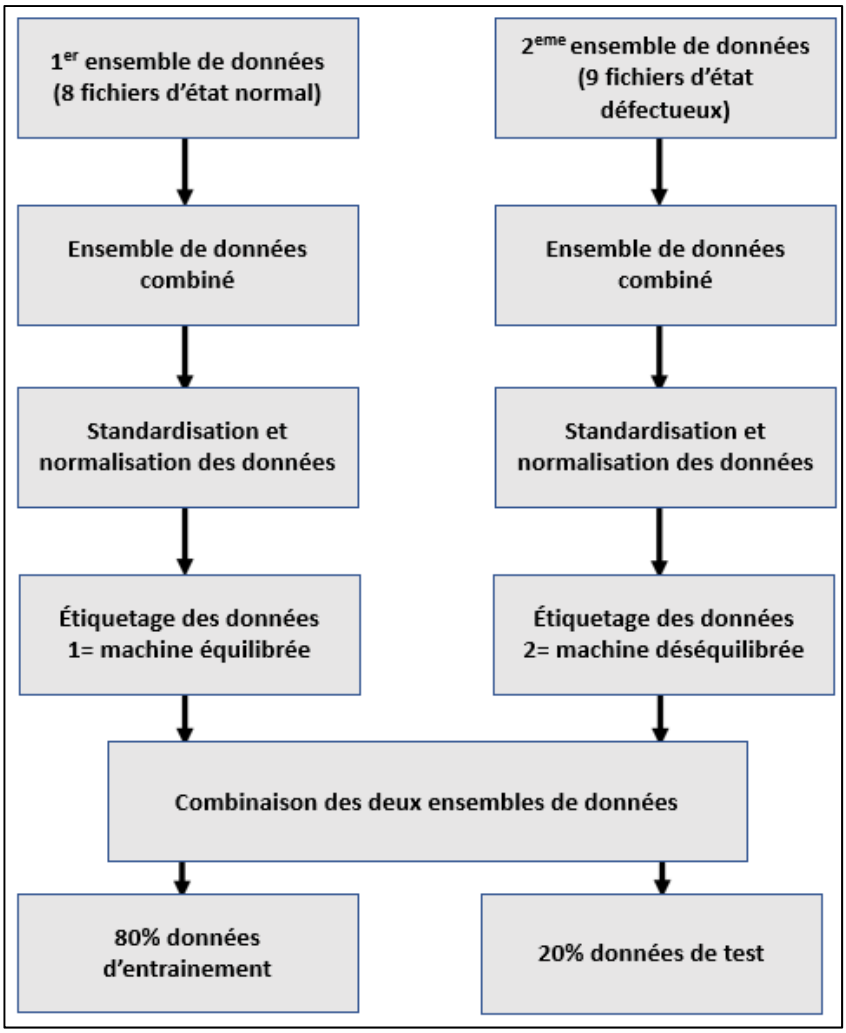


Figure 4.1 Les étapes de prétraitement des données

La construction des modèles doit comprendre la recherche des paramètres optimaux qui permettront à l'algorithme d'atteindre sa performance maximale. En effet, la majorité des modèles d'apprentissage automatique doivent être adéquatement paramétrés pour donner les meilleurs résultats. Ainsi, pour sélectionner les valeurs des hyperparamètres des algorithmes choisis, on a implémenté le « *gridsearch* » en Python avec « *sklearn.model_selection.GridSearchCV*» et dans lequel on a essayé de couvrir toutes les valeurs de ces hyperparamètres qu'on juge pertinent. Il s'agit d'une méthode d'optimisation des hyperparamètres qui permet de tester une série de paramètres et de comparer les performances pour en déduire les meilleures. Enfin, l'évaluation qui est la dernière étape qui sert à évaluer les

résultats obtenus et à comparer la performance de plusieurs modèles pour recommander la meilleure méthode qui permet de prédire avec précision l'état de la turbine. Le processus est illustré dans la figure 4.2.

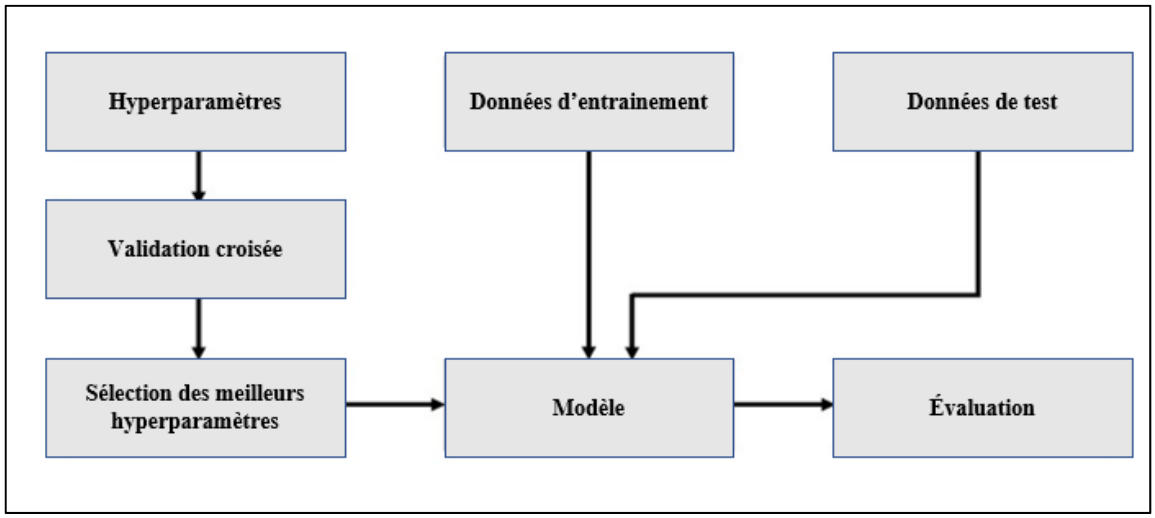


Figure 4.2 Construction du modèle de prédiction

L'évaluation de la performance d'un modèle de classification commence par résumer les résultats dans quatre groupes. Le premier groupe est celui des vrais positifs (TP). Ce groupe représente les observations prédites dans une classe, qui font en fait partie de la classe. Le deuxième groupe est les vrais négatifs (VN). Il représente les observations prédites comme ne pas faire partie d'une classe, qui ne font en réalité pas partie de la classe. Le troisième groupe est les faux positifs (FP) : observations prédites comme faisant partie d'une classe, qui en fait ne font pas partie de la classe. Et enfin, les faux négatifs (FN). Ce groupe indique les nombres des observations prédites comme ne pas faire partie d'une classe, mais en réalité fait partie de cette classe.

Ces résultats sont représentés visiblement dans une matrice de confusion. Il s'agit d'un tableau qui permet de visualiser les performances d'un modèle de classification. La matrice de confusion est aussi utilisée pour calculer les mesures de performance d'un modèle. La figure 4.3 présente une matrice de confusion 2 × 2 en termes abstraits.

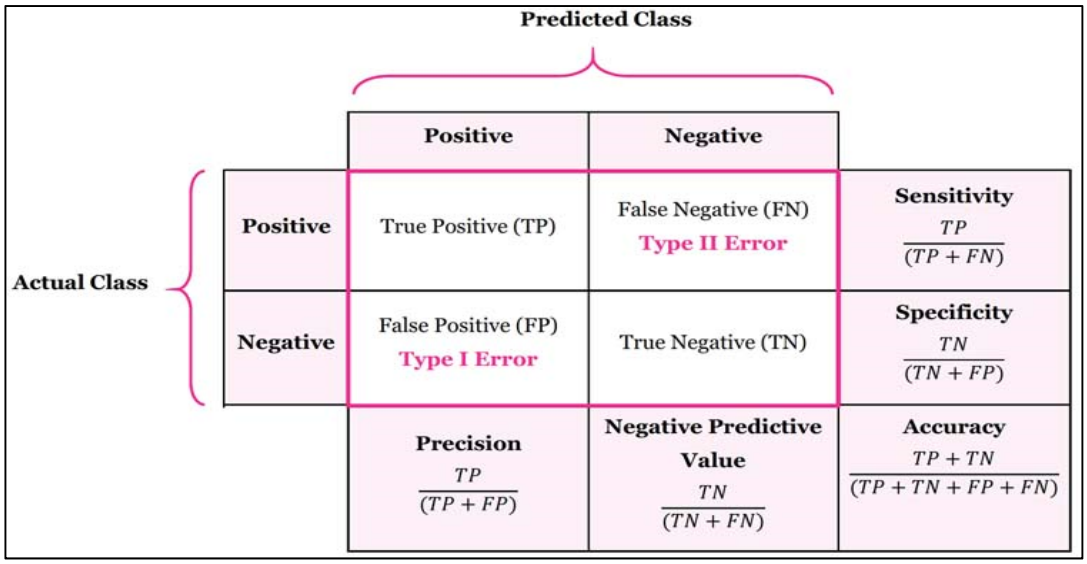


Figure 4.3 Matrice de confusion pour une classification binaire tirée de Data Science (2019)

Les mesures de performance dans les modèles de classification de l'apprentissage automatique sont utilisées pour évaluer les performances des algorithmes de classification de l'apprentissage automatique dans un contexte donné. Ces mesures de performance incluent la précision, le rappel et le score « F1 », et aident à comprendre les forces et les limites de ces modèles lors de la réalisation de prédictions dans de nouvelles situations. Dans cette étude, les quatre paramètres suivants sont sélectionnés.

- L'exactitude: c'est une métrique de performance du modèle d'apprentissage automatique qui est définie comme le rapport des vrais positifs et des vrais négatifs à toutes les observations positives et négatives. En d'autres termes, elle indique à quelle fréquence le modèle d'apprentissage automatique prédit correctement un résultat sur le nombre total de fois où il a fait des prédictions. Mathématiquement, il représente le rapport de la somme des vrais positifs et des vrais négatifs sur toutes les prédictions.

$$Exactitude = \frac{(TP + TN)}{(TP + FN + TN + FP)} \quad (4.1)$$

- La précision: cette mesure tente de répondre à la question suivante: quelle proportion d'identifications positives était réellement correcte? Cette métrique donne une idée sur la part des faux négatifs.

$$Précision = \frac{TP}{TP + FP} \tag{4.2}$$

- Le Rappel: Il représente la capacité du modèle à prédire correctement les points positifs à partir des points positifs réels qui existent dans un ensemble de données. Plus le score de rappel est élevé, meilleur est le modèle d'apprentissage automatique pour identifier les exemples positifs et négatifs. Le score de rappel est une mesure utile du succès de la prédiction lorsque les classes sont très déséquilibrées. Mathématiquement, il représente le rapport des vrais positifs à la somme des vrais positifs et des faux négatifs.

$$Rappel = \frac{TP}{FN + TP} \tag{4.3}$$

- Le Score F1 : Il représente le score du modèle en fonction de la précision et du score de rappel. F-score est une métrique de performance de modèle d'apprentissage automatique qui donne un poids égal à la précision et au rappel pour mesurer ses performances en termes de précision, ce qui en fait une alternative aux métriques de précision (il ne on oblige pas à connaître le nombre total d'observations). Il est souvent utilisé comme une valeur unique qui fournit des informations de haut niveau sur la qualité de sortie du modèle. Mathématiquement, il peut être représenté comme une moyenne harmonique de précision et un score de rappel.

$$Score\ F1 = \frac{2 * \ Score\ de\ precision\ * \ Score\ de\ recall}{Score\ de\ precision\ +\ Score\ de\ recall)} \tag{4.4}$$

4.3 Construction des modèles de prédiction

4.3.1 Machine à vecteurs de support

4.3.1.1 Principe et algorithme

En règle générale, les machines à vecteurs de support (SVM) sont considérées comme une approche de classification, mais elles peuvent être utilisées dans les problèmes de classification et de régression. SVM construit un hyperplan dans un espace multidimensionnel pour séparer au mieux les différentes classes. L'idée centrale du SVM est de trouver un hyperplan marginal maximum (MMH) qui divise au mieux l'ensemble de données en classes et pour pouvoir classer les nouveaux points par la suite.

En fait, l'objectif principal est de séparer l'ensemble de données de la meilleure façon possible. La distance entre les deux points les plus proches est connue sous le nom de marge. L'objectif est de sélectionner un hyperplan avec la marge maximale possible entre les vecteurs de support dans l'ensemble de données. Pour chercher, l'hyperplan marginal maximal, le SVM commence par générer des hyperplans qui séparent au mieux les classes. Ensuite, il sélectionne l'hyperplan droit avec la ségrégation maximale des points de données les plus proches, comme indiqué dans la figure ci-dessous.

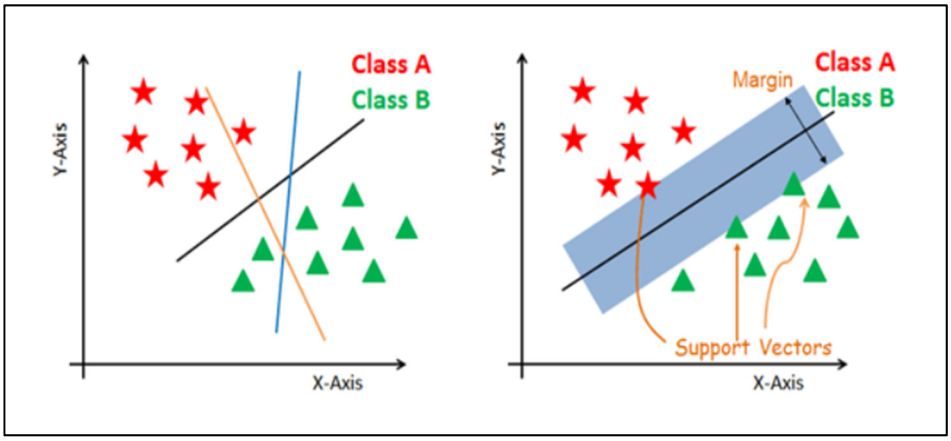


Figure 4.4 Support Vector Machine (SVM)
tirée de Navlani (2018)

4.3.1.2 Implémentation en Python

Pour implémenter l'algorithme SVM, il y a quatre étapes. La première étape consiste à faire un prétraitement des données. Par la suite, il faut ajuster l'algorithme SVM à l'ensemble d'apprentissages. La troisième étape est celle de la prédiction de la classe de l'ensemble test. Ensuite, on crée une matrice de confusion pour pouvoir évaluer le modèle. Enfin, la dernière étape consiste à visualiser et interpréter le résultat.

Pour réaliser ces étapes, il existe de nombreuses bibliothèques disponibles en langage Python. Il suffit d'appeler les fonctions avec les paramètres adéquats choisis en fonction des besoins. Dans cette étude, au départ, on a utilisé directement la fonction SVC et qui prend comme entrée les valeurs des trois paramètres du noyau, des hyperparamètres gamma et C (classifier = SVC (C=1, gamma=0.05, kernel='linear', random_state=0) (voir ANNEXE IV).

Le premier paramètre est la régularisation. C'est le paramètre C dans « *Scikit-le*arn » de python utilisé pour maintenir la régularisation. Il est le paramètre de pénalité, qui représente une mauvaise classification ou un terme d'erreur. Ce terme indique à l'optimisation SVM combien d'erreurs sont supportables. Le deuxième paramètre représente la fonction principale du noyau (Kernel). Il sert à transformer les données d'entrée du jeu de données dans la forme requise. Il existe différents types de fonctions telles que la fonction de base linéaire, polynomiale et radiale (RBF). Le troisième paramètre est appelé Gamma. Une valeur inférieure de Gamma s'adaptera vaguement à l'ensemble de données d'entrainement, tandis qu'une valeur plus élevée de gamma s'adaptera exactement à l'ensemble de données d'entrainement, ce qui entraine un sur ajustement. En d'autres termes, une faible valeur de gamma ne prend en compte que les points proches dans le calcul de la ligne de séparation, tandis que la valeur élevée de gamma prend en compte tous les points de données dans le calcul de la ligne de séparation (Navlani, 2018).

4.3.1.3 Résultat de la classification avec SVM

Le modèle SVM a pris plus que quatre heures à rouler sans donner un résultat. Ceci signifie que le modèle SVM n'est pas vraiment adapté à nos données. Ce résultat est expliqué par la grande

quantité de données injectée au modèle. En effet, la matrice d'entrainement est de taille (3 516 144, 7) ce qui est énorme. En plus, la figure 4.5 montre que les deux classes se chevauchent rendant le SVM difficile à résoudre le problème et différencier entre les deux classes.

En effet, dans le cas d'un grand ensemble de données, ces méthodes nécessitent l'ajout d'énormes quantités de mémoire et d'un temps CPU long aux ressources déjà importantes utilisées dans la formation SVM (Adankon, 2005). Pour remédier à ce problème, il existe la classification des vecteurs de support linéaire. Similaire à SVC avec le paramètre kernel='linear', mais implémenté en termes de liblinear plutôt que de libsvm, il a donc plus de flexibilité dans le choix des pénalités et des fonctions de perte et devrait mieux s'adapter à un grand nombre d'échantillons. En effet (Te-Ming Huang et Vojislav Kecman, 2009) ont montré que le SVM linéaire performe les modèles de SVM surtout quand il s'agit de quantité de données énorme.

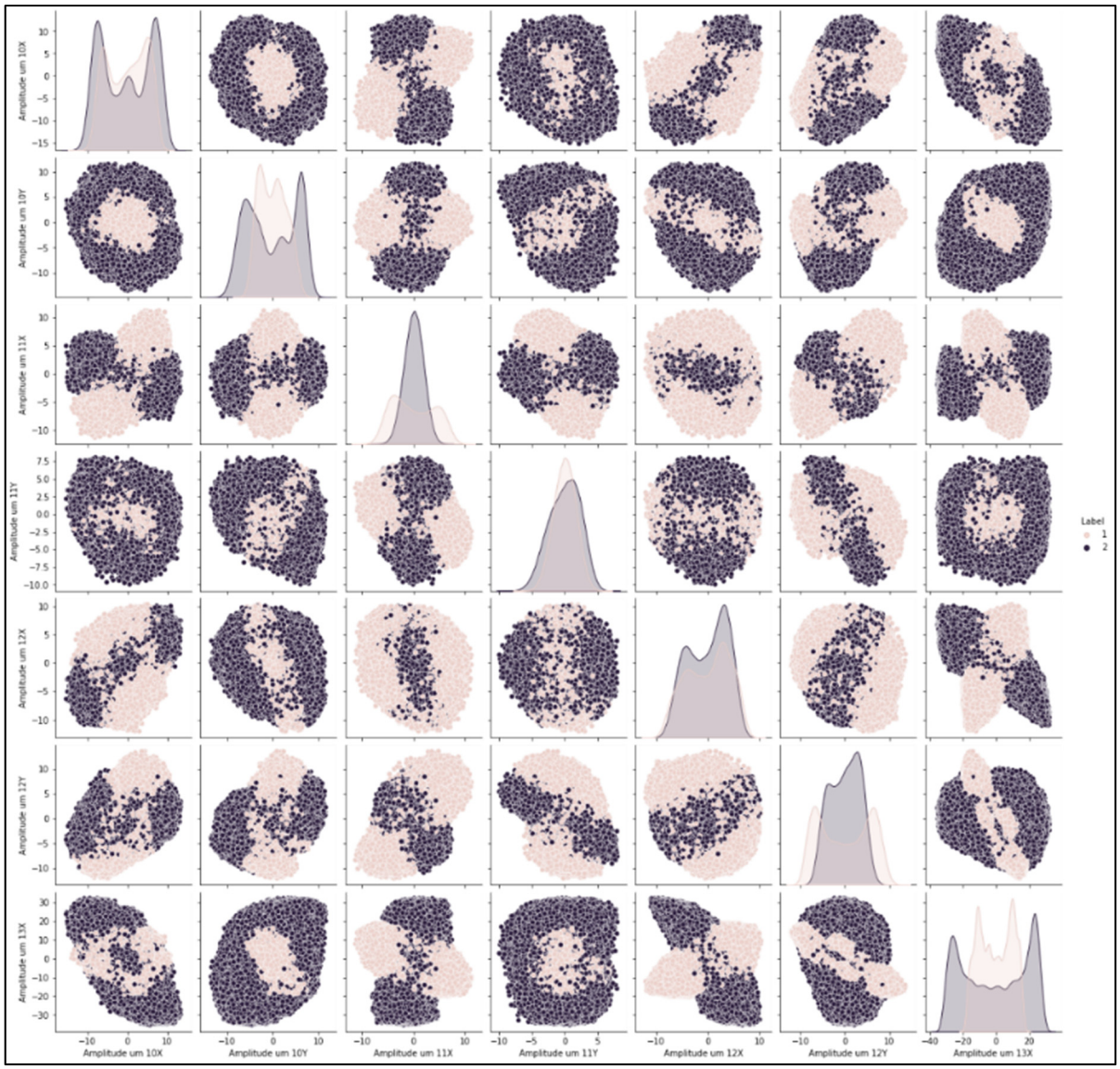


Figure 4.5 Distribution des deux classes

Ainsi, on a testé deux modèles de *LinearSVC* avec *tol=0.0001* et après *tol=0.1*. Le paramètre *tol* représente la tolérance pour les critères d'arrêts. Cela indique à *scikit* d'arrêter de rechercher un minimum (ou un maximum) une fois qu'une certaine tolérance est atteinte. La figure 4.6 représente la matrice de confusion de la classification pour les deux tolérances. En effet, le critère de tolérance n'a pas d'influence sur la performance du modèle par contre cela a permis d'avoir un résultat plus rapidement que le SVC avec le paramètre kernel='linear'.

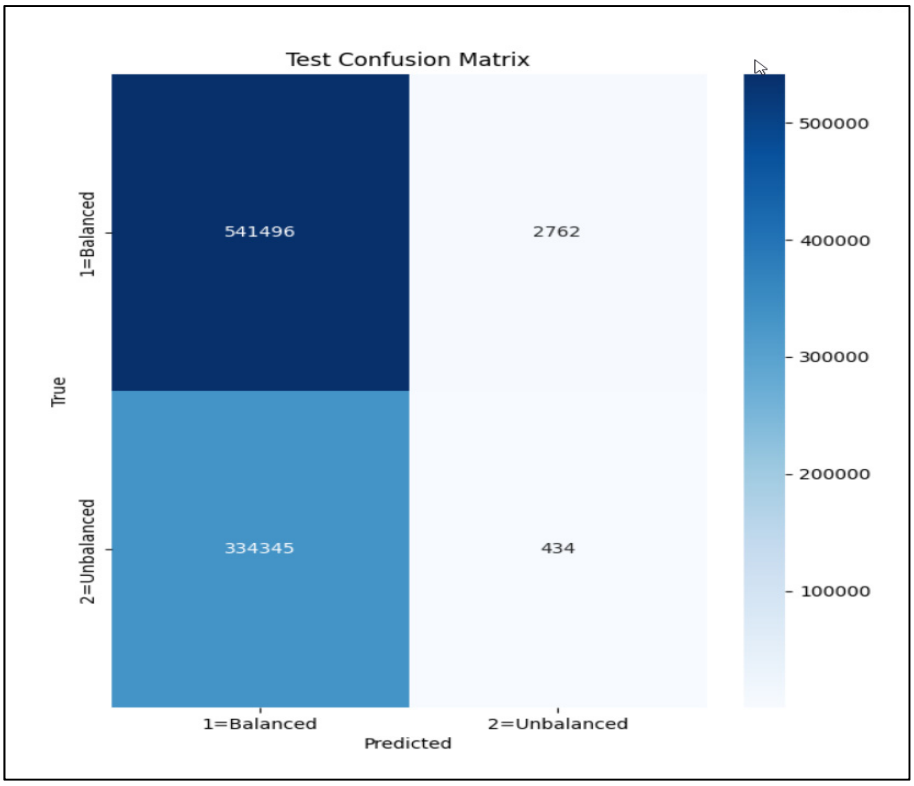


Figure 4.6 Matrice de confusion avec le modèle LinearSVC
(random_state=0, tol=0.0001 ; ET tol=0.1)

Analysons le résultat obtenu. La matrice comprend quatre classes :

- Vrai Positif (TP) : la classe réelle est positive et prédite positive. Ici, sur (541 496 + 2762) positifs réels, 541 496 sont correctement prédits positifs.
- Vrai Négatif (TN) : La classe réelle est négative et prédite négative. Ici, sur (334 345 + 434) négatifs réels, 434 sont correctement prédits négatifs.
- Faux positif (FP) : la classe réelle est négative, mais prédite comme positive. Sur (334 345 + 434) négatifs réels, 334 345 sont faussement prédits comme positifs. Ainsi, la valeur de Faux Positif est 334 345.
- Faux négatif (FN) : la classe réelle est positive, mais prédite comme négative. Sur (541 496 + 2762) positifs réels, 2762 sont faussement prédits comme négatifs. Ainsi, la valeur de faux négatif est 2762.

Les mesures de performance de cette classification sont regroupées dans le tableau 4.6

Tableau 4.1 Mesures de performance du modèle LinearSVM

Mesures	Tol=0.0001=0.1
Exactitude %	61.65
Rappel %	99.49
Précision	61.82

L'utilisation du LinearSVM avec une tolérance d'arrêt rend l'entrainement des données plus rapide. Le modèle donne une précision égale à 61.65%, par contre on remarque que le nombre des points FP appartenant réellement à la classe 2 et qui sont prédit dans la classe 1 reste très élevé. Le modèle donne un Rappel égal 99.49% et qui donne une indication sur la part du faux négatif Plus le Rappel est proche de 100% plus le classificateur est bon et il est capable de reconnaitre les positifs. Le modèle donne une précision égale 61.82% et elle donne une indication sur la part des faux positifs. C'est-à-dire la capabilité du système à reconnaitre les vrais négatifs.

4.3.2 Algorithme des K plus proches

4.3.2.1 Principe et algorithme

L'algorithme des K plus proches est l'un des algorithmes d'apprentissage le plus simple basé sur la technique d'apprentissage supervisé et peut être utilisé pour la régression ainsi que pour la classification, mais il est principalement utilisé pour les problèmes de classification. L'algorithme KNN suppose la similitude entre le nouveau cas/données et les cas disponibles et place le nouveau cas dans la catégorie la plus similaire aux catégories disponibles. Il est également appelé algorithme d'apprentissage paresseux, car il n'apprend pas immédiatement de l'ensemble d'entrainement, au lieu de cela, il stocke l'ensemble de données et au moment de la classification, il effectue une action sur l'ensemble de données. Ainsi, lorsqu'il obtient de nouvelles données, il classe ces données dans une catégorie très similaire aux nouvelles données. Le fonctionnement de l'algorithme KNN peut être expliqué sur la base de quatre grandes étapes. On commence par sélectionner le nombre K des voisins dont on calcule la distance euclidienne. Par la suite, on prend les K voisins les plus proches selon cette distance, et on compte le nombre

de points de données dans chaque catégorie. Finalement, on affecte les nouveaux points de données à une catégorie pour laquelle le nombre de voisins est maximum.

Comme on l'a discuté, le principe de l'algorithme du classificateur KNN consiste à trouver K nombre prédéfini d'échantillons d'apprentissage les plus proches de la distance du nouveau point et à prédire son étiquette. La valeur minimale de K est 1. Cela signifie utiliser un seul voisin pour la prédiction. Le maximum est le nombre de points de données dont on dispose. La recherche a montré qu'aucun nombre optimal de voisins ne convient à tous les types d'ensembles de données. Chaque jeu de données possède ses propres exigences. Dans le cas d'un petit nombre de voisins, le bruit aura une plus grande influence sur le résultat, et un grand nombre de voisins rend le calcul coûteux. Ainsi pour pouvoir le définir, on a généré le modèle sur différentes valeurs de K (de 1 à 10) et vérifié leurs performances. Dans le code qu'on a développé, le programme exécute l'algorithme pour chaque valeur de K, compare les différentes performances obtenues en termes de précision et retourne la valeur choisie avec son score (voir ANNEXE III).

4.3.2.2 Résultat de la classification du classificateur KNN

La valeur K optimale estimée par le code est égale à 9. En effet la figure 4.7 montre bien que pour K=9, on a une erreur minimale et donc on aura une performance maximale. La figure 4.8 représente la matrice de confusion du classificateur KNN obtenue avec un nombre K de voisins égal à 9.

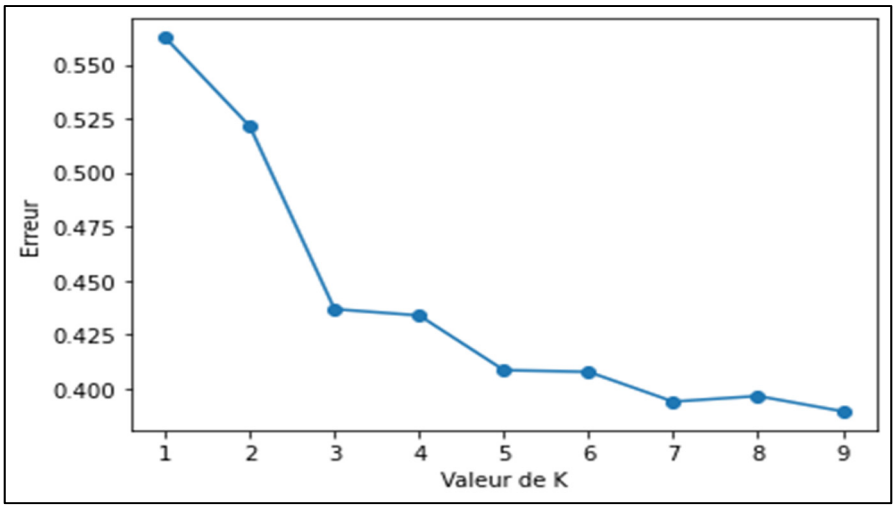


Figure 4.7 Recherche de la valeur de K pour le classificateur KNN

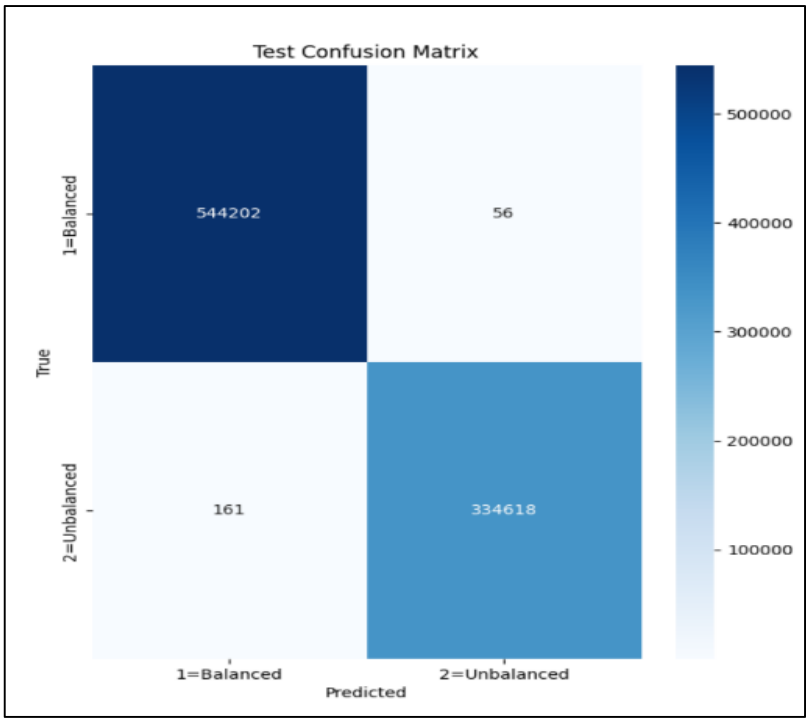


Figure 4.8 Matrice de confusion pour le classificateur KNN

La matrice de confusion comprend quatre classes:

- Vrai Positif (TP): la classe réelle est positive et prédite positive. Ici, sur le nombre total de données positives réelles (544 202 + 56), 544 202 sont correctement prédits positifs.
- Vrai Négatif (TN): la classe réelle est négative et prédite négative. Ici, sur (334 618 + 161) négatifs réels, 334 618 sont correctement prédits négatifs.
- Faux positif (FP) : la classe réelle est négative, mais prédite comme positive. Sur (334 618 + 161) négatifs réels, 161 sont faussement prédits comme positifs. Ainsi, la valeur de Faux Positif est 161.
- Faux négatif (FN) : la classe réelle est positive, mais prédite comme négative. Sur (544 202 + 56) positifs réels, 56 sont faussement prédits comme négatifs. Ainsi, la valeur de faux négatif est 56.

4.3.2.3 Évaluation du modèle KNN et discussion

Le Tableau 4.2 regroupe les mesures de performance du classificateur KNN avec un nombre de voisins égale à 9.

Tableau 4.2 Mesures de performance du modèle KNN avec K=9

Mesures	**KNN avec K= 9**
Exactitude %	99.97
Rappel %	99.98
Précision %	99.97
F1 score %	99.97

Les taux élevés des mesures de performance du modèle KNN pour un nombre des voisins proches K= 9 montrent que notre modèle est capable de bien prédire l'état de la machine. C'est-à-dire si la machine est balancée ou elle présente un balourd avec une précision de (99.97%). Une valeur de rappel élevée signifie qu'il y avait très peu de faux négatifs et que le classificateur est plus permissif dans les critères de classification de quelque chose comme positif. Plus le score

F1 est élevé, plus le modèle est bon. Cela signifie que la précision et le rappel sont élevés et donc le modèle est performant.

4.3.3 Forêt aléatoire

4.3.3.1 Principe et algorithme

La forêt aléatoire (RF) est un algorithme d'apprentissage supervisé. Elle peut être utilisée à la fois pour la classification et la régression. Les forêts aléatoires créent des arbres de décision sur des échantillons de données sélectionnés au hasard, obtiennent la prédiction de chaque arbre et sélectionnent la meilleure solution au moyen d'un vote. En effet, c'est une méthode d'ensemble (basée sur l'approche diviser pour régner) d'arbres de décision générés sur un ensemble de données divisé de manière aléatoire. Cette collection de classificateurs d'arbres de décision est également connue sous le nom de forêt. Les arbres de décision individuels sont générés à l'aide d'un indicateur de sélection d'attributs tels que le gain d'informations, le rapport de gain et l'indice de Gini pour chaque attribut. Chaque arbre dépend d'un échantillon aléatoire indépendant. Dans un problème de classification, chaque arbre vote et la classe la plus populaire est choisie comme résultat final. La figure 4.9 illustre le processus. L'algorithme est décrit en quatre étapes. La première consiste à sélectionner les échantillons aléatoires à partir d'un ensemble de données donné. Dans la deuxième étape, on construit un arbre de décision pour chaque échantillon pour avoir ensuite un résultat de prédiction à partir de chaque arbre de décision. Par la suite, on effectue un vote pour chaque résultat prévu. Et enfin, la sélection du résultat de la prédiction finale se fait selon la prédiction possédant le plus de votes. Ce processus est illustré par la figure 4.9.

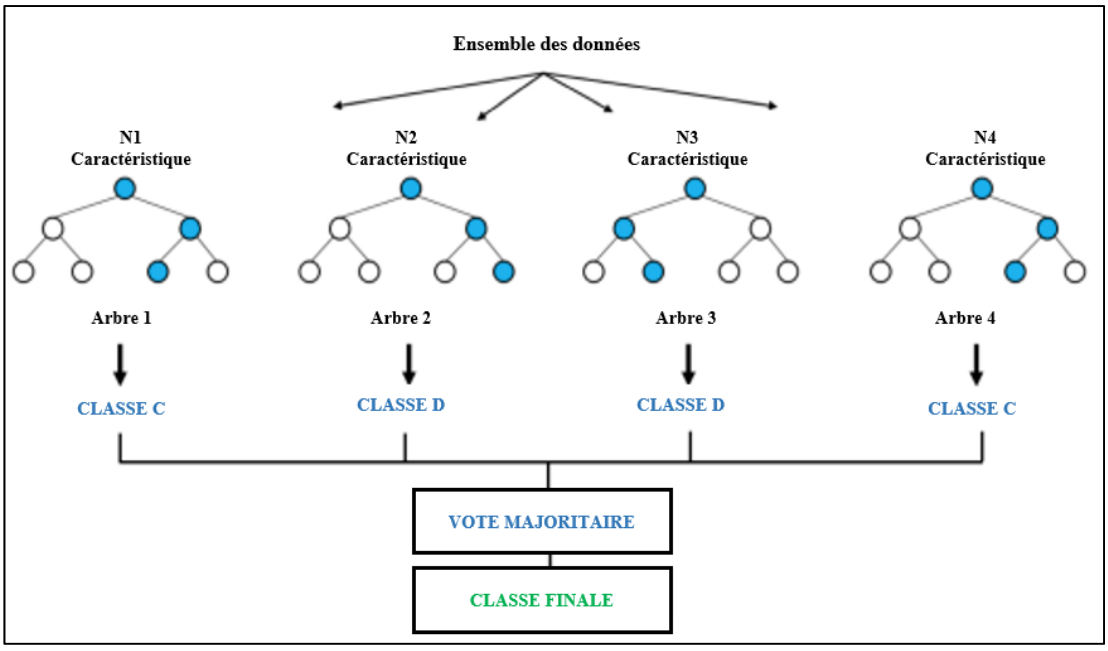


Figure 4.9 Principe des forêts aléatoires adapté de Kirasich, Smith, & Sadler (2018)

4.3.3.2 Implémentation en Python

Pour implémenter l'algorithme RF, on commence par un prétraitement des données. On ajuste ensuite l'algorithme RF à l'ensemble d'apprentissages. On prédit le résultat de l'ensemble test. Après, on crée la matrice de confusion pour évaluer la performance du modèle utilisé. Enfin. On visualise et on interprète le résultat obtenu. Pour cela, on examinera l'algorithme « *RFclassifier*» se trouvant dans la bibliothèque « *sklearn*». Cependant, pour la RF, on doit choisir le nombre d'arbres à créer et le nombre de variables à utiliser à chaque division d'un nœud. Les paramètres du modèle sont donc :

- max_depth : le max_depth d'un arbre dans la forêt aléatoire est défini comme le chemin le plus long entre le nœud racine et le nœud feuille.
- n_estimators : C'est le nombre d'arbres utilisés.

Au départ, le modèle de RF avec un nombre d'arbres égale à 10 et une profondeur maximale égale à 2 a été construit et ensuite la fonction « *GridSearch* » a été employée pour optimiser le

modèle et trouver les paramètres optimaux qui vont donner une précision de classification maximale (voir ANNEXE V).

4.3.3.3 Résultat de classification pour le RF

Avant la recherche des paramètres optimaux avec « *GridSearch* », la figure 4.10 montre que le résultat obtenu lors de la classification avant de lancer la recherche des paramètres optimaux. La matrice représente le résultat du classificateur: *Classifier = RandomForestClassifier (max_depth=2, n_estimators=10, random_state=0).*

En analysant le résultat obtenu, on remarque que la matrice de confusion comprend quatre classes:

- Vrai Positif (TP) : la classe réelle est positive et prédite positive. Ici, on a 543 436 points qui appartiennent à la classe 1 et qui sont prédits correctement. Sur (543 436 + 822) positifs réels, 543 436 sont correctement prédits positifs.
- Vrai Négatif (TN) : La classe réelle est négative et prédite négative. Ici, on a 181 798 points qui appartenant à la classe 2 et qui sont prédits ainsi. Sur (181 798 + 152 981) négatifs réels, 181 798 sont correctement prédits négatifs.
- Faux positif (FP) : la classe réelle est négative, mais prédite comme positive. Sur (181 798 + 152 981) négatifs réels, 152 981 sont faussement prédits comme positifs. Ainsi, la valeur de faux positif est 152 981.
- Faux négatif (FN) : la classe réelle est positive, mais prédite comme négative. Sur (543 436 + 822) positifs réels, 822 sont faussement prédits comme négatifs. Ainsi, la valeur de faux négatif est 822.

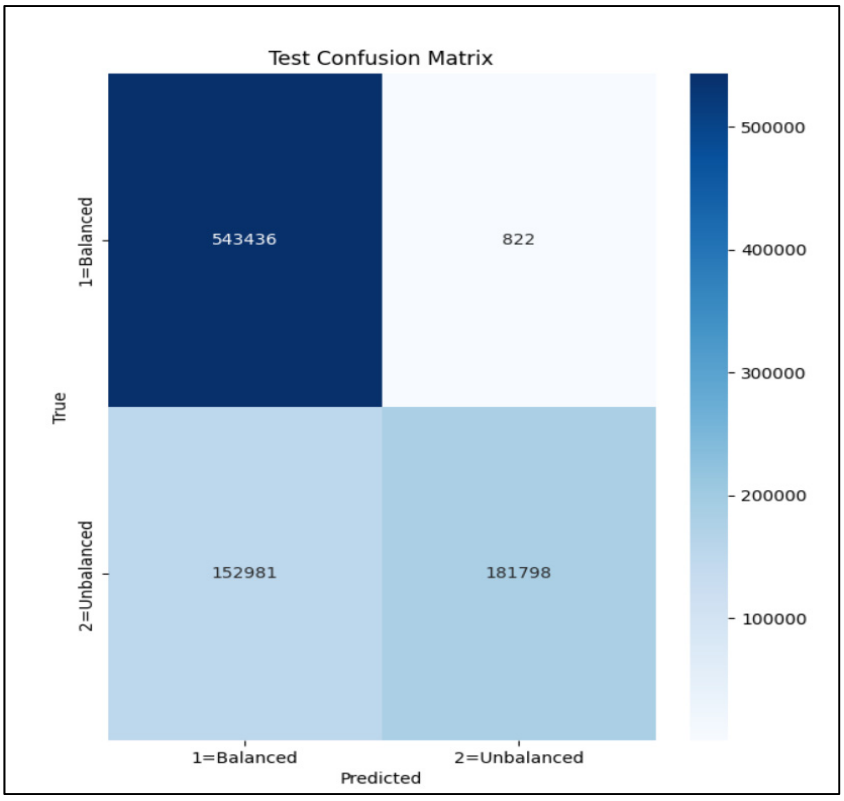


Figure 4.10 Matrice de confusion de RF avec (max_depth=2, n_estimators=10, random_state=0)

L'étape suivante consiste à tester la performance du modèle avec les paramètres estimés par « *GridSearch* » et qui semblent être les meilleurs dans la plage des valeurs entrées. La figure 4.11 montre le résultat obtenu lors de la classification après la recherche des paramètres optimaux. C'est-à-dire, cette matrice représente le résultat du classificateur:
Classifier = RandomForestClassifier(max_depth=20, n_estimators=15, random_state=0).

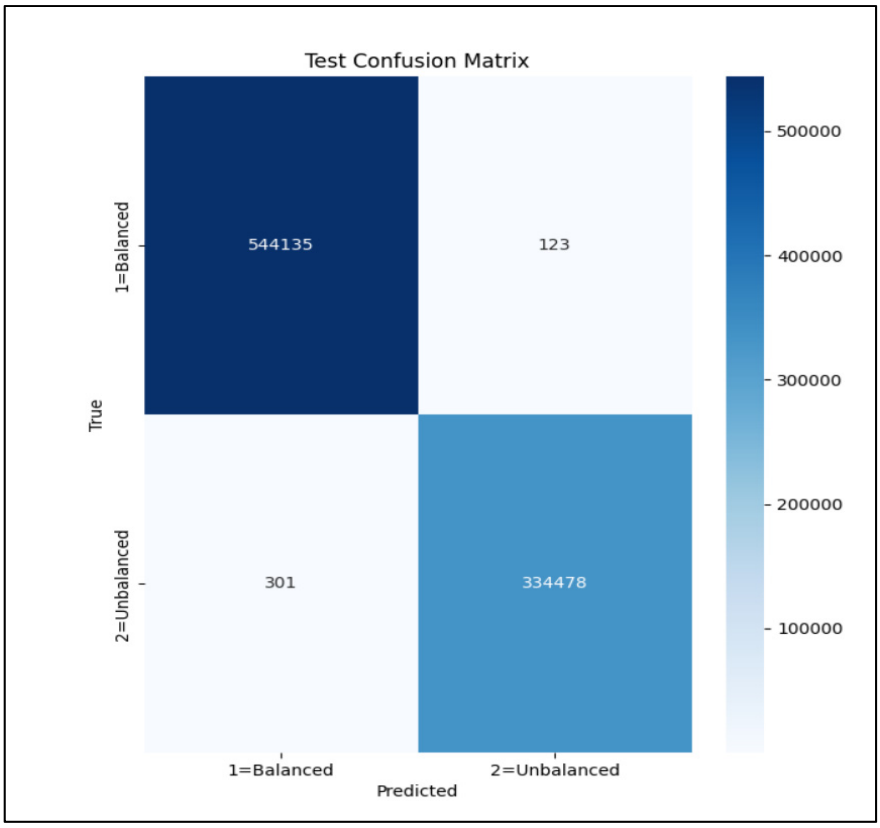


Figure 4.11 Matrice de confusion pour le modèle RF (max_depth=20, n_estimators=15, random_state=0

La matrice de confusion comprend quatre classes:

- Vrai Positif (TP) : la classe réelle est positive et prédite positive. Ici, sur (544 135+ 123) positifs réels, 544 135 sont correctement prédits positifs.
- Vrai Négatif (TN) : La classe réelle est négative et prédite négative. Ici, sur (334 478 + 301) négatifs réels, 334 478 sont correctement prédits négatifs.
- Faux positif (FP) : la classe réelle est négative, mais prédite comme positive Sur (334 478 + 301) négatifs réels, 301 sont faussement prédits comme positifs. Ainsi, la valeur de Faux Positif est 301.
- Faux négatif (FN) : la classe réelle est positive, mais prédite comme négative. Sur (544 135+ 123) positifs réels, 123 sont faussement prédits comme négatifs. Ainsi, la valeur de Faux négatif est 123.

4.3.3.4 Évaluation du classificateur RF et discussion

Les Tableaux 4.3 et 4.4 regroupent les mesures de performance du classificateur RF pour deux paramètres différents.

Tableau 4.3 Mesures de performance du modèle RF avec depth_max= 2 et n_estimators=10

Mesures	**RF**
Précision %	78.03
Rappel %	99.85
F1 score %	87.59
Exactitude	82.50

Tableau 4.4 Mesures de performance du modèle RF avec depth_max= 20 et n_estimators=15

Mesures	**RF**
Précision %	99.94
Rappel %	99.97
F1 score %	99.95
Exactitude	99.95

D'après les résultats obtenus, on remarque lorsqu'on a augmenté le nombre des arbres à 15 et la profondeur maximale à 20, le modèle est devenu plus performant avec une précision égale à 99.94% contre 87.59% quand on a pris une profondeur maximale égale à 2 et le nombre d'arbres égal à 10. Également, le Rappel et le F1 score ont augmenté. Cela signifie qu'il y avait très peu de faux négatifs en les comparant aux faux négatifs trouvés issus du premier classificateur. Pareillement, la valeur des faux positifs a largement diminué et donc la précision du modèle augmente. Donc il est clair qu'en augmentant le nombre d'arbres et la profondeur maximale, le modèle devient plus performant et prédit mieux l'état de la machine. Par contre en choisissant plus d'arbres, la complexité temporelle du modèle RF augmente également ce qui est un peu coûteux.

4.4 Évaluation et conclusion

L'évaluation des méthodes d'apprentissage automatique est l'étape suivante. Les résultats des mesures de performance pour les trois classificateurs obtenus dans la section précédente sont regroupés dans le Tableau 4.5. Rappelons que le SVM n'a généré aucun résultat pendant plusieurs heures (plus que 6 heures d'entrainement). En effet, le modèle SVM sur un ensemble de données volumineux prend comparativement plus de temps à s'entrainer. Par conséquent, dans notre contexte il n'est pas le bon et il a été remplacé par le SVM linéaire qui est plus adapté à des ensembles de données suffisamment grands.

Tableau 4.5 Tableau comparatif entre les différents modèles

Mesures	**RF (20,15)**	**RF (2,10)**	**KNN (9)**	**SVM**	**SVM linéaire**
Exactitude %	99.95	82.50	99.97	-	61.65
Rappel %	99.97	99.84	99.98	-	99.49
Précision %	99.94	78	99.97		61.82
F1 score %	99.95	87.59	99.97	-	61.73
Temps d'entrainement (s)	160.21	20.76	190.78	Infini	97.65
Temps d'inférence(s)* e-5	0.16	0.0444	9.756	-	0.00226

Entre les deux classificateurs RF (max_depth= 20 n_estimators=15) et RF (max_depth= 2, n_estimators=10), le premier est plus performant avec une exactitude égale 99.95% contre 82.50% pour la RF (max_depth= 2, n_estimators=10) par contre il prend plus de temps à s'entrainer et à prédire une nouvelle donnée et c'est normal, car le nombre d'arbres est plus grand.

Le KNN (K= 9) surpasse le SVM linéaire malgré que ce dernier prenne beaucoup moins de temps à s'entrainer et à prédire les nouvelles données. Toutes les métriques calculées pour le KNN sont supérieures à celles calculées pour le SVM linéaire 99.95% d'exactitude pour le KNN contre 61.65% pour le SVM linéaire. Remarquons aussi, que le rappel du SVM est supérieur par

rapport à la précision, ceci prouve que le modèle du SVM linéaire reconnait bien vrais positifs alors que la précision est égale 61.82% ce qui indique qu'il y a quand même un bon nombre de faux positifs.

Donc si on donne la priorité au temps de prédiction et donc de la prise de décision, et que les faux positifs coûtent plus que les faux négatifs, on choisira le SVM linéaire. Or, ce n'est pas vrai, car ignorer les faux négatifs engendrerait des arrêts de maintenance non essentiels voire, qui ne devraient pas avoir lieu et cela va coûter à l'entreprise. Donc le SVM linéaire est à écarter.

Les deux classificateurs KNN et RF (max_depth=20, n_estimators=15) semblent avoir les mêmes performances. Les deux classificateurs reconnaissent les vrais positifs et les vrais négatifs (une exactitude élevée égale à 99.95% pour le RF et 99.97 % pour le KNN). Les deux modèles minimisent également les faux négatifs (le Rappel est égal à 99.97 % pour le RF et 99.98% pour le KNN) et les faux positifs (la précision égale à 99.94 % pour le RF et 99.97% pour le KNN).

En fait, un bon classificateur maintiendra un F1score élevé. Cette métrique tient compte à la fois de la précision et du rappel en essayant d'équilibrer les deux métriques (Steen, 2020).
Ainsi, les deux classificateurs sont bons pour notre problème de classification. Cependant, KNN a tendance à être plus lent avec des ensembles de données volumineux, car il analyse l'ensemble de données entier pour faire la prédiction et il ne généralise pas les données à l'avance. Donc sur le plan rapidité, on peut dire que le RF est mieux.
Une autre façon d'évaluer les modèles est de tracer la courbe de précision-rappel. Elle est utilisée pour évaluer les performances des algorithmes de classification binaire et fournit une représentation graphique des performances d'un classificateur à travers de nombreux seuils, plutôt qu'une seule valeur (Steen, 2020).

La figure 4.12 montre le tracé de la courbe précision-rappel de certains classificateurs théoriques. La ligne pointillée grise représente un classificateur Baseline. Ce classificateur prédirait simplement que toutes les instances appartiennent à la classe positive. La ligne violette représente un classificateur idéal avec une précision et un rappel parfait à tous les seuils. Dans

le cas réel, les classificateurs se situeront quelque part entre ces deux lignes. Ils ne représentent pas le classificateur parfait, mais ils fournissent de meilleures prédictions que le classificateur Baseline.

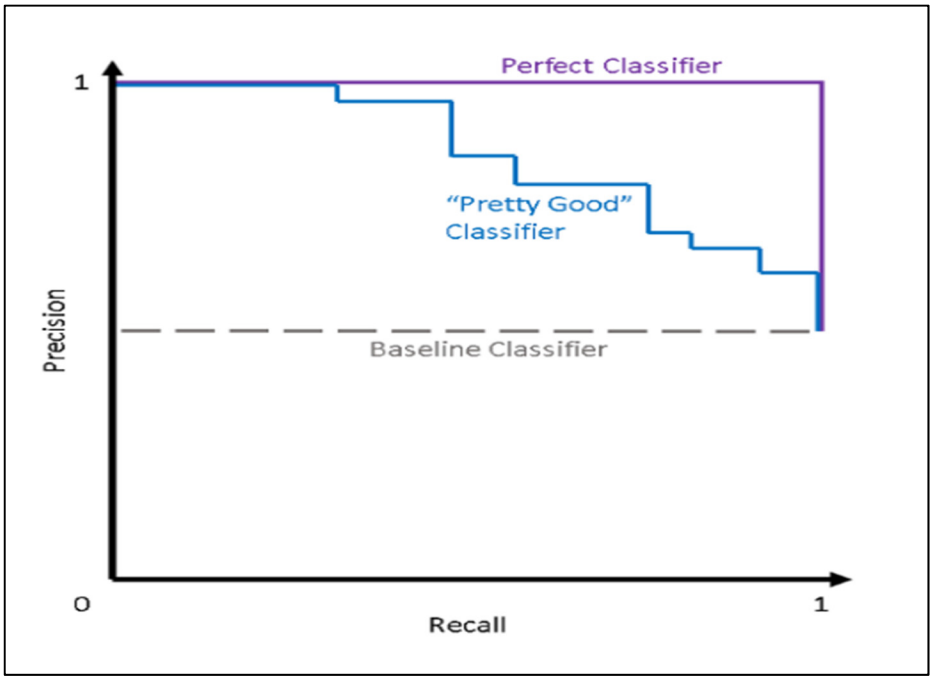


Figure 4.12 Courbe précision-Rappel pour un classificateur théorique tirée de Steen (2020)

À l'aide de la fonction « *plot_precision_Rappel_curve (classsifier, x_test, y_test*) » on a tracé les courbes précision_Rappel des 4 modèles. Voici les schémas obtenus.

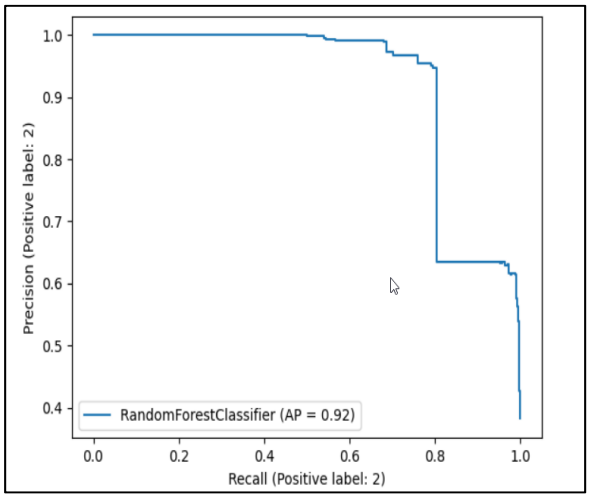


Figure 4.13 Courbe Précision/ Rappel pour RF (2,10)

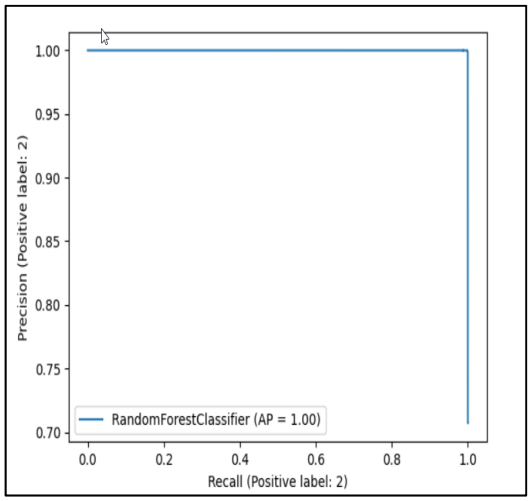


Figure 4.14 Courbe Précision/ Rappel pour RF (20,15)

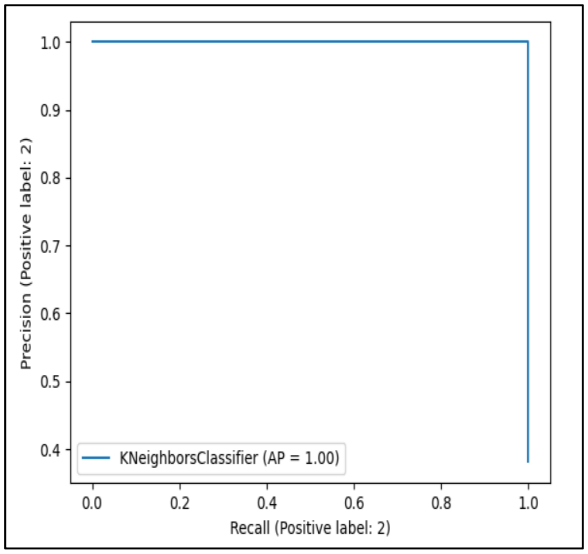


Figure 4.15 Courbe Précision/Rappel pour le KNN

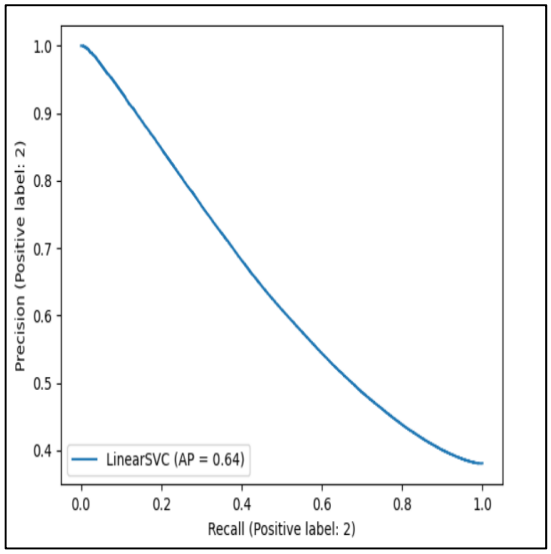


Figure 4.16 Courbe Précision/ Rappel pour le LinearSVM

En comparant les courbes Précisions / Rappel ci-dessus avec celle présentant le classificateur théorique, on remarque que le Modèle RF avec n_estimators = 15 et max_depth égale à 20 correspond à un modèle parfait et il s'adapte à notre problème. Ce classificateur permet de détecter le problème de balourd et classer la machine en machine balancée ou pas balancée avec un F1 score égal à 99.95%. Cependant, il faut prendre en considération que plus le nombre d'arbres est grand, plus il nécessite beaucoup de puissance de calcul ainsi que des ressources

pour la construction des arbres et la combinaison des sorties. Le RF nécessite également beaucoup de temps pour la formation, car il combine de nombreux arbres de décisions pour pouvoir identifier la classe. Les résultats montrent aussi qu'avec un nombre de voisins proches égal à 9, le modèle KNN semble être performant. En effet, il est capable de détecter le défaut du balourd avec une précision égale à 99.98 %, qui est une très bonne précision. Cependant, le temps d'exécution du modèle est un peu long par rapport au RF, car à chaque fois il faut calculer le nombre de k qui peut être complexe et coûteux. Quant au LinearSVM, d'après la courbe, on ne pourra jamais avoir les deux mesures, la précision et le rappel, élevés. Il faut augmenter soit la précision, soit le rappel alors qu'un bon classificateur doit avoir une précision élevée et un rappel élevé également.

4.5 Conclusion

Dans ce chapitre, trois algorithmes d'apprentissage supervisé ont été comparés pour classer l'état de la machine quant à son état d'équilibre, le LinearSVM, le RF et le KNN. D'abord, des successions d'étapes de prétraitement de données ont été réalisées afin de mettre les données brutes en une seule forme simple, propre et prête à être exploitée par le classificateur.

Les paramètres accordés à chacun des modèles jouent un rôle important dans la production de résultats de haute précision. Chaque classificateur a différentes étapes de réglage et paramètres réglés. Pour chaque classificateur, on a testé une série de valeurs pour le processus de réglage avec les paramètres optimaux déterminés en fonction de la précision de classification globale la plus élevée.
Le modèle a été entrainé avec une seule machine et une seule gravité de déséquilibre rendant le problème à un problème de classification binaire simple ayant deux classes. Machine équilibrée pour la classe 1 et machine non équilibrée pour la classe 2.

Dans cette étude, les résultats obtenus sous les paramètres optimaux de chaque classificateur ont été utilisés pour comparer les performances des classificateurs. Les deux modèles RF et KNN donnent les meilleurs résultats avec un F1 score respectivement égale à 99,95% et 99,97 %. Pour

le LinearSVM, on a obtenu un score F1 égale à 61.73%. Ce résultat s'explique par le fait qu'il est tolérant aux mauvaises classifications (FN et FP) et par conséquent, le classificateur LinearSVM n'est pas très adapté à notre problème de classification malgré qu'il soit le plus rapide dans son entrainement ainsi que sa prise de décision par rapport à l'état de la machine, car une fausse classification pourrait être très coûteuse dans un contexte de maintenance industrielle.

CHAPITRE 5

AMÉLIORATION DU MODÈLE DE PRÉDICTION

5.1 Introduction

Dans ce chapitre, on va étudier le comportement de nos modèles vis-à-vis des nouvelles données. Le but de cette étude est de vérifier la validité externe des algorithmes étudiés précédemment et d'y apporter des améliorations. Ainsi, deux bases de données issues de deux turbines de puissance différentes ont été introduites à notre modèle comme étant des données de test. Les deux machines ont des problèmes de vibrations distincts. Dans cette partie, la classe « 1 » désigne les machines équilibrées. La classe « 2 » désigne les machines déséquilibrées, alors que la classe « 3 » désigne les machines ayant un déséquilibre sévère. Finalement, la classe « 4 » désigne que la machine a un autre défaut.

5.2 Description des nouvelles données

Dans ce chapitre, de nouvelles données sont obtenues pour d'autres turbines. Il s'agit de deux ensembles de données caractérisant les vibrations de deux turbines de puissance. Le premier ensemble de données contient cinq signaux issus de cinq capteurs. La machine présente un problème de vibration lié à un déséquilibre très sévère. Chaque signal dure 15.484 secondes et est enregistré avec une fréquence d'échantillonnage de 1.25 kHz et on a 5 capteurs. Ainsi, le nombre total des points analysés est égal à 15.484*1250 =19 355 * 5 points.

Le deuxième ensemble de données contient cinq signaux issus de cinq capteurs. La machine présente un problème de vibration qui n'est pas lié à un déséquilibre, mais à un autre problème. Chaque signal dure 35.1448 secondes et est enregistré avec une fréquence d'échantillonnage de 1.25 kHz et on a 5 capteurs. Ainsi, le nombre total des points analysés est égal à 35.1448 *1250 = 43 931 * 5 points.

5.3 Méthodologie

Pour tester la validité externe du modèle, on va utiliser les trois modèles testés précédents et déjà entrainés avec le premier jeu de donnée. Les deux nouveaux ensembles seront des données de test. Pour cela, il est primordial de faire un prétraitement des données et mettre les nouvelles données sous la même forme que celle utilisée pour l'entrainement. À cet égard, on a commencé par combiner tous les fichiers CSV afin de les mettre dans un seul fichier. Par la suite, on a éliminé toutes les valeurs nulles, standardisées et normalisé les données pour enfin construire la matrice des caractéristiques.

On a aussi remarqué que la première matrice utilisée lors de l'entrainement comprend sept colonnes (sept capteurs) alors que la nouvelle matrice qu'on va utiliser lors de la validation en comprend cinq seulement. Pour cette raison on a utilisé la méthode de sélection des fonctionnalités avec les meilleures valeurs « *F ANOVA »* à l'aide de la classe « *SelectKBest Feature Selection* » de la bibliothèque « *Scikit-learn »* de Python pour extraire les meilleures caractéristiques du premier ensemble avec la variance la plus explicative et rendre les deux matrices ayant le même format. On a passé deux paramètres, l'un est la métrique de notation qui est f_classif et l'autre est la valeur de N qui signifie le nombre d'entités qu'on veut dans l'ensemble de données final. On a utilisé fit_transform pour ajuster et convertir l'ensemble de données actuel dans l'ensemble de données souhaité. Toutes les étapes ont été exécutées avec le langage de programmation Python. La figure 5.1 résume les différentes étapes décrites ci-dessus.

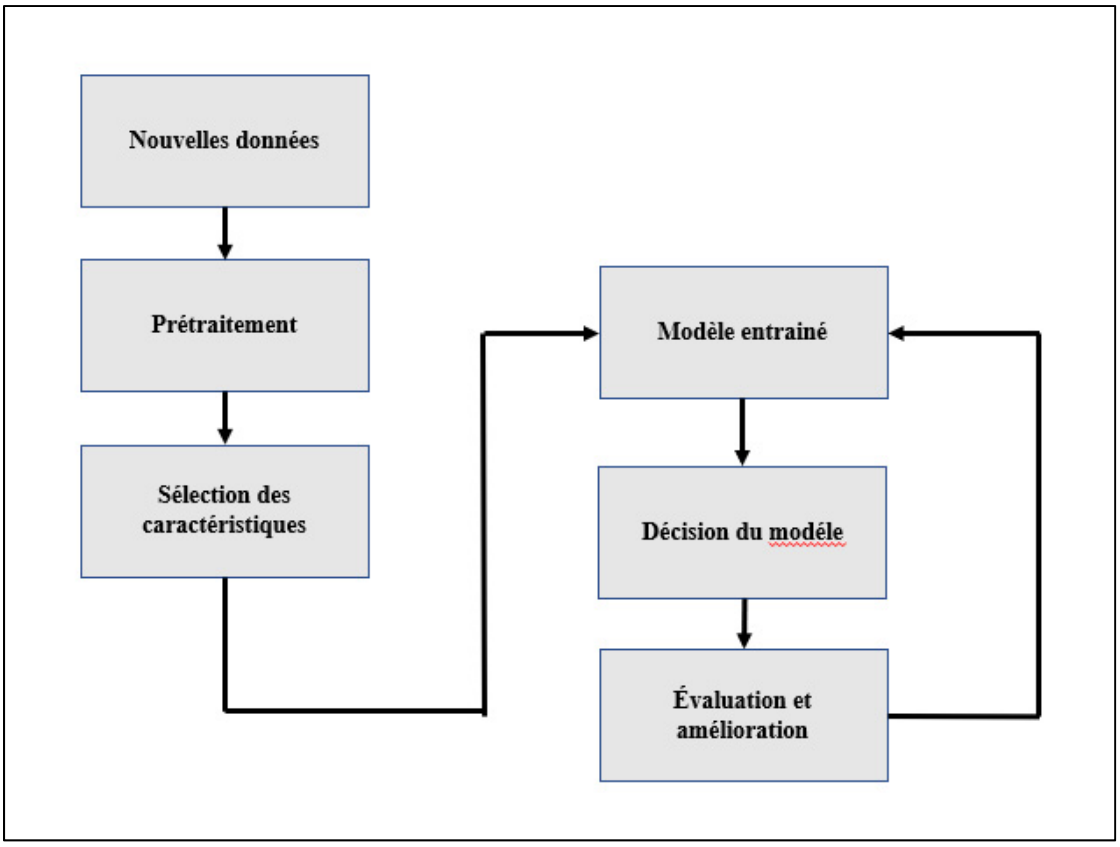


Figure 5.1 Étapes de test de validité externe du modèle

5.3.1 Résultat avec les trois classificateurs

5.3.1.1 Première machine

Les figures 5.2, 5.3, 5.4 représentent les matrices de confusions avec les trois modèles sélectionnés dans la première partie qui sont le RF (max_depth=20, n_estimators=15), le KNN (K=9) et le LinearSVM.

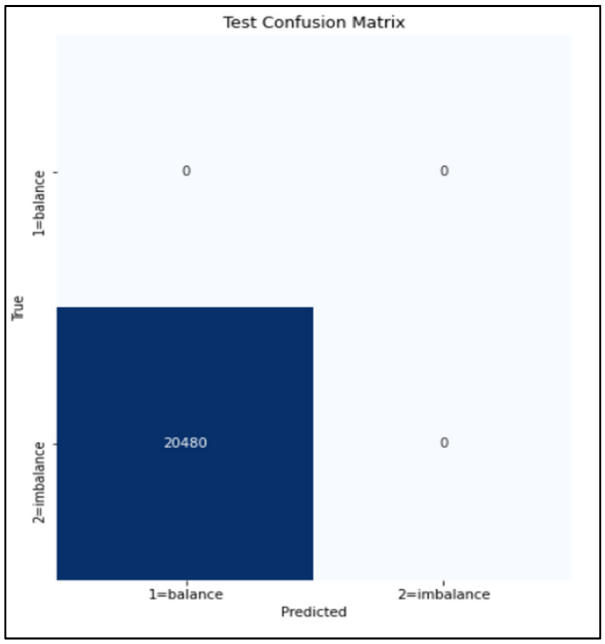


Figure 5.2 Matrice de confusion Linear SVM

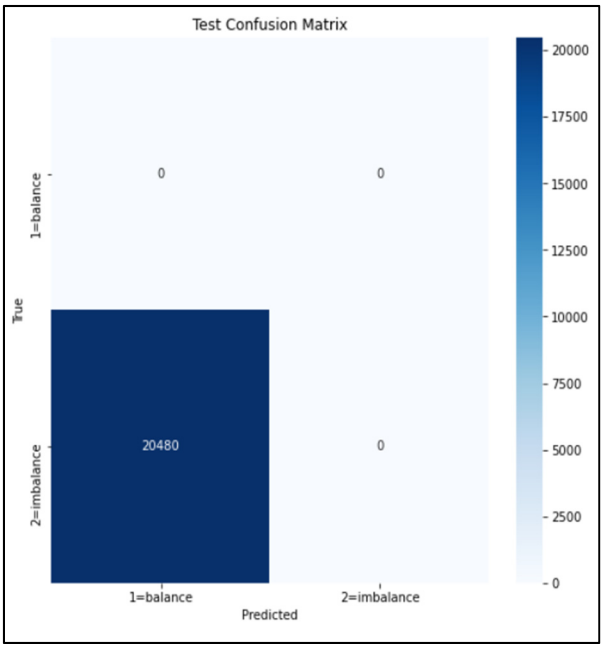


Figure 5.3 Matrice de confusion KNN

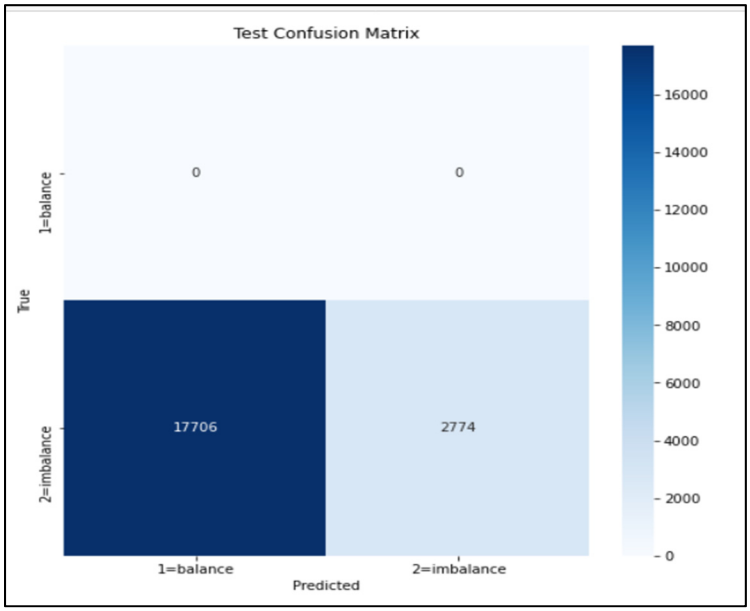


Figure 5.4 Matrice de confusion pour le RF

Le Tableau 5.1 représente le résultat de la classification des trois modèles pour la première machine.

Tableau 5.1 Résultat du 1er test de validité externe

	RF	**KNN**	**LinearSVM**
Décision du modèle	Machine balancée	Machine balancée	Machine balancée
Exactitude par rapport à l'état réel	13.54%	0%	0%

Les résultats montrent que les trois classificateurs LinearSVC (to=0.0001), KNN (k=9) et RF (max_depth=20, n_estimators=15) ont prédit que la machine testée est balancée alors qu'en réalité, elle présente un problème de balourd très sévère. Le KNN et le LinearSVM ont classé tous les points étant des données issues d'une machine saine alors que pour le RF 2774 points ont été classés anormaux sur 20 480 points qui sont réellement anormaux.

5.3.1.2 Deuxième machine

Les figures 5.5, 5.6, et 5.7 représentent les matrices de confusions avec les trois modèles sélectionnés dans la première partie qui sont le RF (max_depth=20, n_estimators=15), le KNN (K=9) et le LinearSVM.

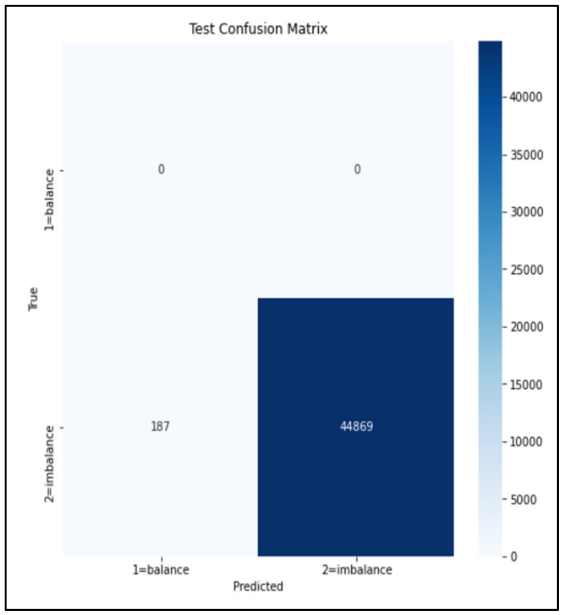


Figure 5.5 Matrice de confusion (Machine 2) KNN

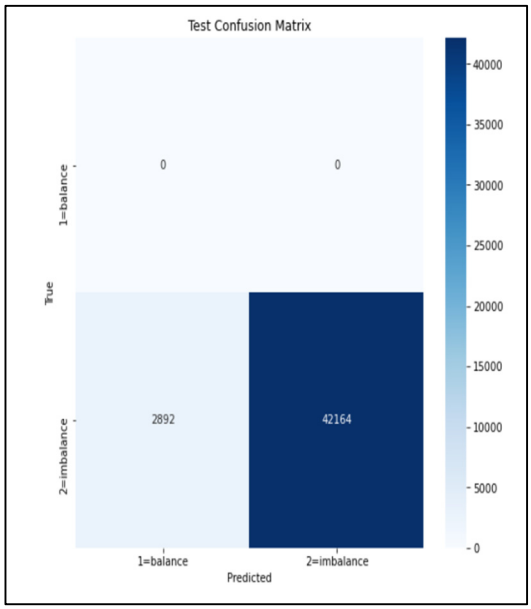


Figure 5.6 Matrice de confusion (Machine 2) RF

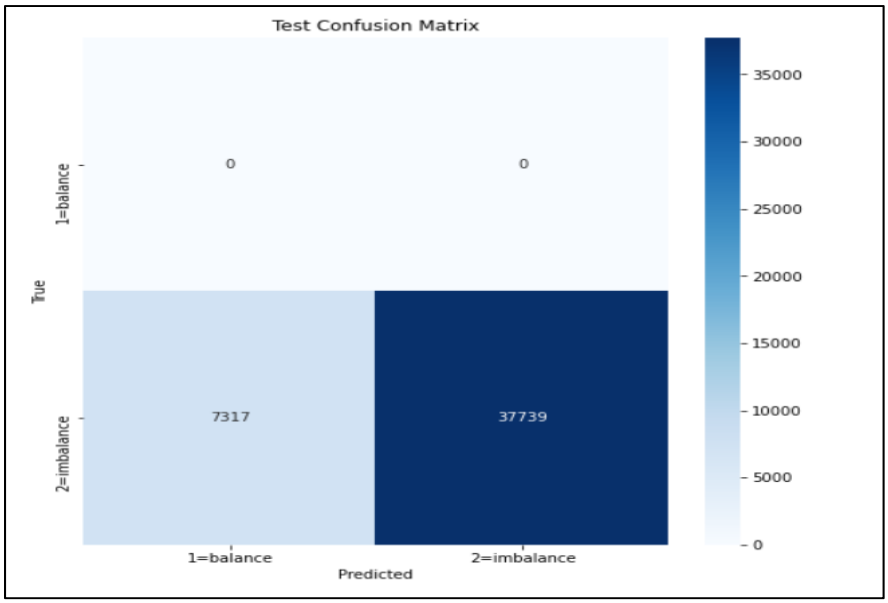


Figure 5.7 Matrice de confusion (Machine 2) LinearVSM

Le Tableau 5.2 regroupe les décisions des trois modèles quant à l'état de la deuxième machine.

Tableau 5.2 Résultat du 2eme test de validité externe

	RF	**KNN**	**LinearSVM**
Décision du modèle	Déséquilibre	Déséquilibre	Déséquilibre
Exactitude	93.58%	99.58%	83.76

Le comportement vibratoire de la deuxième machine est très similaire au comportement vibratoire que le modèle a appris. Bien que la cause de ces vibrations ne soit pas le déséquilibre, cette machine a été classée comme étant une machine présentant un problème de balourd. En effet, les trois classificateurs ont jugé que la machine est déséquilibrée avec des précisions élevées. 99.58% pour le KNN, 93.58% pour le RF et 83.76% pour le LinearSVM.

5.3.2 Analyse des résultats et conclusions

Les résultats des trois classificateurs sont plus au moins attendus. En effet, pour la première machine, les amplitudes des vibrations sont très élevées par rapport aux amplitudes que notre modèle connait et donc pour le modèle, ce sont des caractéristiques en dehors de l'intervalle qu'il a fixé pour dire que la machine présente un balourd. Pareillement pour la deuxième machine, les amplitudes des vibrations sont dans le même intervalle que le modèle ce qui explique le résultat obtenu.

De ce résultat, on peut conclure que ce modèle ne peut pas être généralisé et appliqué à toutes les machines dont les vibrations sont soupçonnables, car comme on a vu les causes des vibrations dans les machines rotatives sont diverses et elles ne sont pas forcément liée à un déséquilibre. En plus, le modèle a été entrainé pour un seul degré de déséquilibre et donc il ne va pas reconnaitre les autres sévérités. Pour y remédier, il faut élargir les données d'entrainement. Il faut aussi intégrer des données issues de différentes machines avec différents problèmes et de multiples sévérités.

5.4 Amélioration du modèle

Afin d'améliorer le modèle et le rendre plus général, fonctionnel à plusieurs contextes, on propose d'intégrer les nouvelles données dans les données d'entrainement. De ce fait, le problème de départ n'est plus un problème de classification binaire, à deux classes qui sont machine équilibrée ou déséquilibrée, mais plutôt, il est devenu un problème de classification multi classes, principalement 4. La classe 1 est pour une machine équilibrée. La classe 2 est pour une machine présentant un déséquilibre. La classe 3 représente une machine ayant un déséquilibre sévère et finalement, la classe 4 est pour une machine présentant un autre défaut. Ainsi, une option possible est de combiner les deux ensembles de données et de les mélanger au hasard. Ensuite, divisez l'ensemble de données résultant en ensembles d'entrainement et de test comme illustré dans la figure suivante.

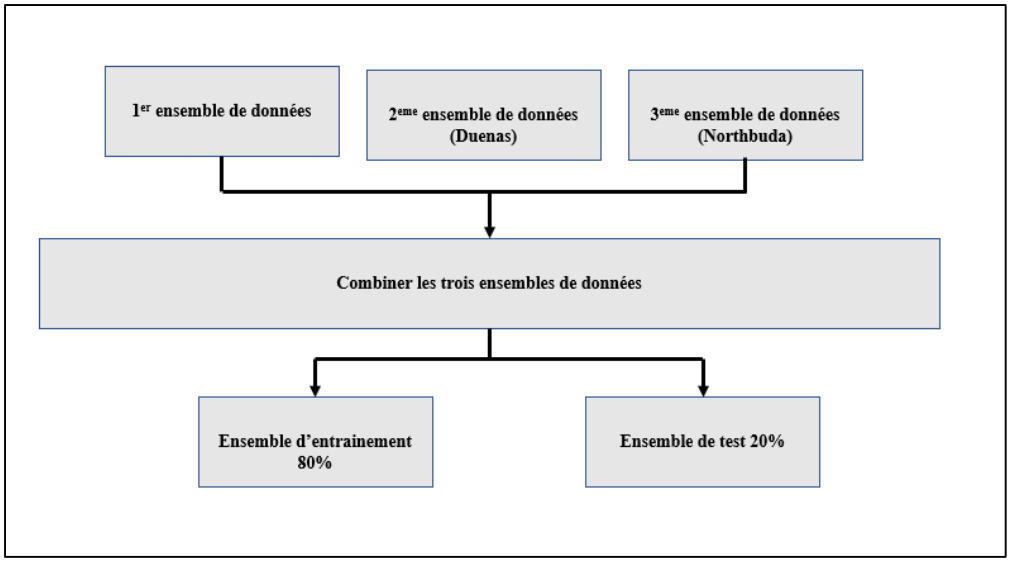


Figure 5.8 Méthode d'amélioration du modèle

5.4.1 Résultats obtenus

Dans le cas du classificateur KNN, on commence par chercher le nombre des voisins proches de K optimal. Le tracé de la figure 5.9 représente l'erreur en fonction de la valeur de K. Le résultat montre que l'erreur est minimale pour un nombre de k voisins proches est égale à 9.

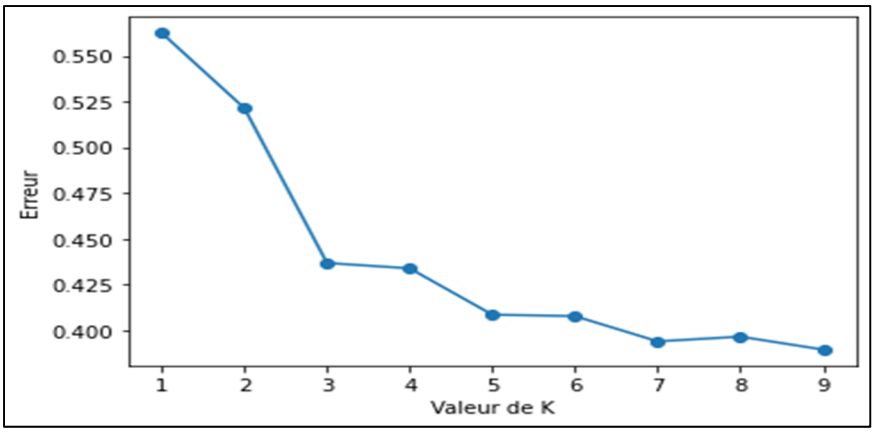


Figure 5.9 L'erreur estimée en fonction de la valeur de K KNN

La figure 5.10 représente la matrice de confusion en utilisant le classificateur KNN (K=9).

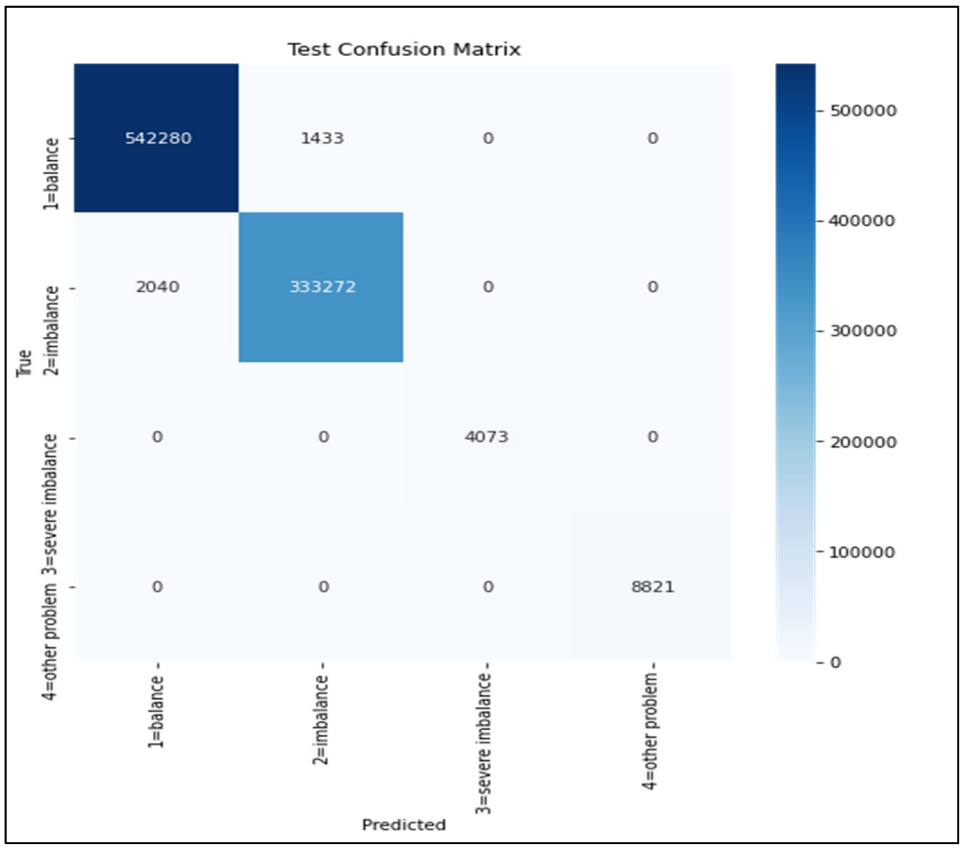


Figure 5.10 Matrice de confusion pour les 4 classes pour le classificateur KNN (K=9)

- Sur (542 280 + 1433) points appartenant à la classe 1, 542 280 sont correctement prédits alors que 1433 sont faussement prédits comme classe2.
- Sur (334 272 + 2040) points à la classe 2, 334 72 sont correctement prédits 2040 sont faussement prédits comme classe1.
- Sur (4073) points appartenant à la classe 3, 4073 sont correctement prédits.
- Sur (8821) points appartenant à la classe 4, 8821 sont correctement prédits.

Pour le classificateur RF, la figure 5.11 présente le résultat de la classification obtenu.

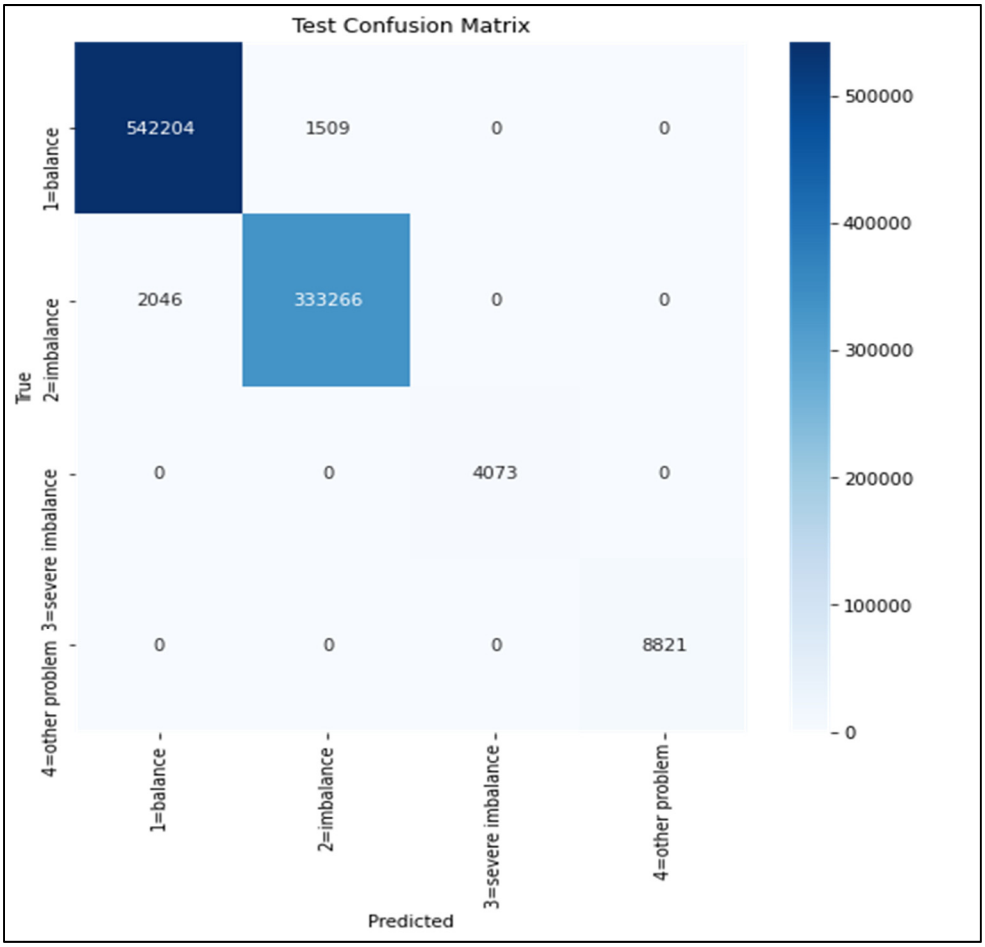


Figure 5.11 Matrice de confusion pour le modèle amélioré avec RF

- Sur (542204 + 1509) points appartenant à la classe 1, 542204 sont correctement prédits alors que 1509 sont faussement prédits comme classe2.
- Sur (333266 + 2046) points à la classe 2, 333266 sont correctement prédits et 2046 sont faussement prédits comme classe1.
- Sur (4073) points appartenant à la classe 3, 4073 sont correctement prédits.
- Sur (8821) points appartenant à la classe 4, 8821 sont correctement prédits.

Pour le LinearVSM, la figure 5.12 présente le résultat de la classification obtenu.

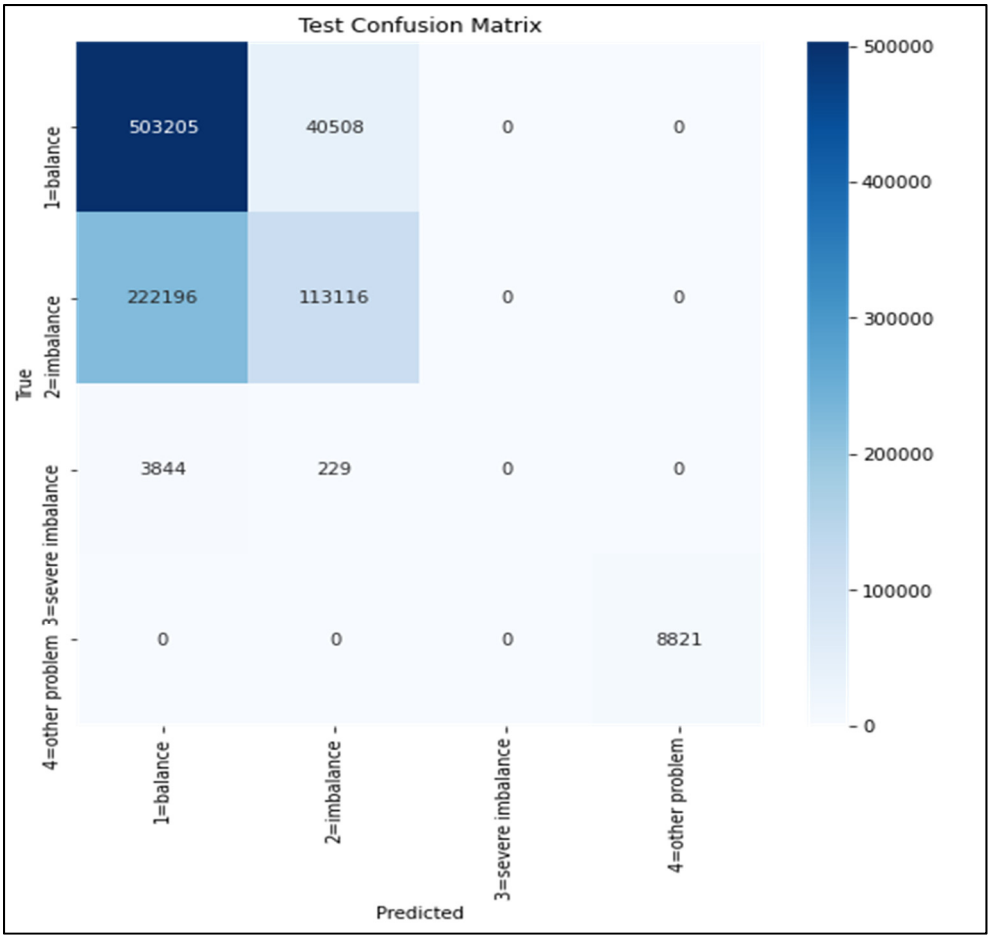


Figure 5.12 Matrice de confusion pour le LinearVSM

- Sur (503205 + 40508) points appartenant à la classe 1, 503205 sont correctement prédits alors que 40508 sont faussement prédits comme classe2.
- Sur (113116 + 222196) points à la classe 2, 113 116 sont correctement prédits et 222 196 sont faussement prédits comme classe1.
- Sur (3844+229) points appartenant à la classe 3, 0 point est correctement prédit.
- Sur (8821) points appartenant à la classe 4, 8821 points sont correctement prédits.

5.4.2 Discussion et conclusion

D'après les matrices de confusion obtenues, il est clair qu'il y a un grand déséquilibre entre les classes. Ceci est illustré par la figure ci-dessous.

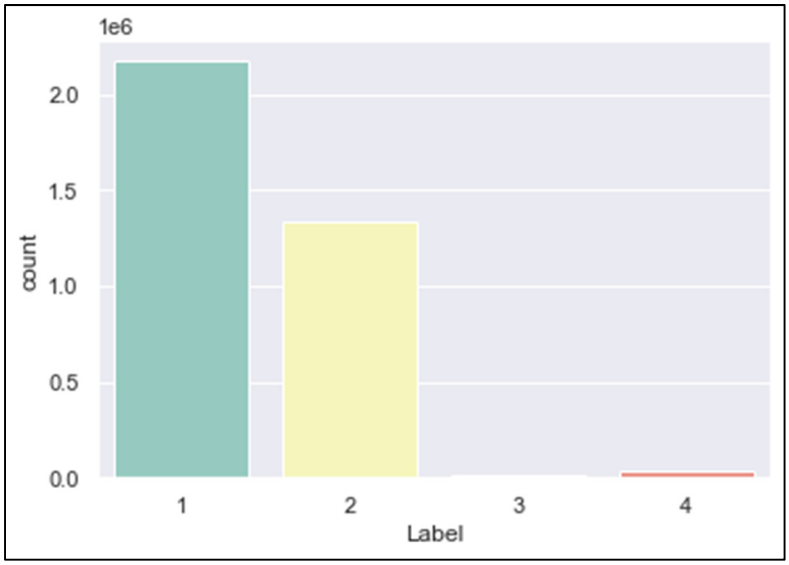


Figure 5.13 Déséquilibre entre les classes

Dans le cas d'un ensemble de données déséquilibrées, il ne faut pas utiliser la métrique « Exactitude », cela peut conduire à des conclusions erronées. Mais d'autres paramètres de rendement doivent être utilisés, comme le score F1. En effet, un ensemble de données déséquilibré peut conduire à des résultats inexacts même lorsque des modèles brillants sont utilisés pour traiter ces données. Si les données sont biaisées, les résultats le seront également (Borad, 2019).

Tableau 5.3 Comparaison entre les trois classificateurs

	RF	**KNN**	**LinearSVM**
Exactitude %	99.60	99.61	70.1
F1 Score	99.78	99.79	56.32

D'après les matrices de confusions et les F1 Score, on peut conclure que les deux classificateurs KNN et RF peuvent être utilisé dans les cas des ensembles déséquilibrés et peuvent donner de bons résultats. Pour le RF, le score F1 est égal à 99.72% et pour le KNN est égal à 99.75%.

5.4.3 Technique de SMOTE pour équilibrer un ensemble de données

L'une des approches les plus populaires consiste à équilibrer l'ensemble de données à l'aide de méthodes de suréchantillonnage ou sous-échantillonnage (Seyyed Mohammad Javadi & Asadollah, 2021) . Ainsi, dans cette partie, on propose d'équilibrer nos données en créant des points de données synthétiques en utilisant l'algorithme SMOTE. « *Synthetic Minority Oversampling technique* ». C'est une technique de suréchantillonnage et permet de créer des points de données synthétiques pour la classe minoritaire. On peut appliquer SMOTE sur l'ensemble de données d'entrainement à l'aide de la bibliothèque *imblearn*. La figure ci-dessous montre que l'équilibrage de l'ensemble des données d'entrainement s'est fait.

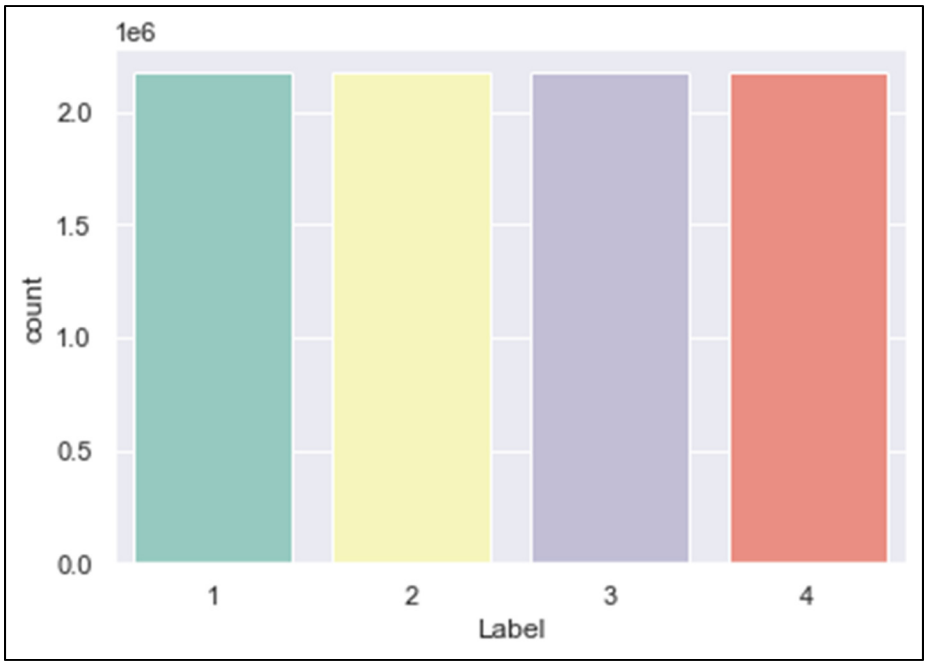


Figure 5.14 Ensemble de données équilibré

L'étape suivante consiste à entrainer à nouveau les modèles en utilisant le nouvel ensemble de données. Les résultats obtenus sont inscrits dans le tableau ci- dessous.

Tableau 5.4 Comparaison entre les trois classificateurs après équilibrage des classes

	RF	**KNN**	**LinearSVM**
Exactitude %	99.60	99.58	47.03%
F1 Score	99.78	99.77	51.00%

Les résultats obtenus sont en accord avec les conclusions qu'on a tirées précédemment. En effet, l'exactitude et le F1 score des deux classificateurs KNN et RF sont similaires à ceux obtenus quand il y avait un déséquilibre entre les classes. Ceci confirme nos conclusions précédentes.

Selon plusieurs recherches, la forêt aléatoire est un algorithme idéal pour traiter le déséquilibre extrême dans un ensemble de données (Hussai, Chan, & Patanwala, 2021). En effet, la

possibilité d'incorporer des pondérations de classe dans le classificateur de forêt aléatoire le rend sensible aux coûts et donc pénalise le classement erroné de la classe minoritaire. En outre, il combine la technique d'échantillonnage et l'apprentissage d'ensemble, par conséquent, en sous-échantillonnant la classe majoritaire et en faisant pousser des arbres sur un ensemble de données plus équilibré. Pour le LinearSVM, on remarque que l'exactitude a baissé de 70% à 47% après équilibrage des données, par contre le F1 score est proche de 56% contre 51%.

Nos résultats confirment donc que l'exactitude n'est pas une métrique fiable pour évaluer la performance des différents modèles surtout quand on a un ensemble de données où les classes sont déséquilibrées. Par contre, le F1 score est une bonne métrique pour évaluer les modèles d'apprentissage automatique. C'est la moyenne harmonique de précision et de rappel et donne une meilleure mesure des cas mal classés que la métrique exactitude. D'après l'analyse des résultats, les deux classificateurs KNN et RF sont bon pour notre problème de classification et donnent de bons résultats

5.5 Conclusion

Dans ce chapitre, un test de validité externe a été fait en introduisant aux trois modèles sélectionnés, des données de test issues des deux machines différentes et dont le modèle ignore l'état. L'algorithme a classé la première comme étant une machine saine alors qu'en réalité elle présente un balourd très sévère. Ceci s'explique par le fait, que le modèle a été entrainé par des intervalles d'amplitudes de vibrations inférieurs à celles caractérisant un balourd sévère et donc celles-ci ne correspondent pas aux caractéristiques que le modèle a choisies pour la classe 2 qui représente la classe des machines déséquilibrée.

Quant à la deuxième machine, elle avait un autre problème qui causait les mêmes intensités de vibrations que le modèle connaisse ce qui justifie sa décision par rapport à l'état de la machine et qui est la présence du déséquilibre. Ces résultats montrent que le modèle était limité à une seule machine avec une intensité de déséquilibre bien définie. Pour y remédier et améliorer les modèles, on les a entrainés avec ces nouvelles données pour les rendre plus générales. Ainsi,

le problème n'est plus un problème de classification binaire, mais plutôt un problème de classification à 4 classes à savoir, les machines équilibrées, les machines présentant un balourd, les machines présentant un balourd sévère, et les machines ayant un autre défaut. Cependant, en rajoutant les nouvelles données, on a créé un déséquilibre entre les classes. Ce problème a été résolu par la méthode de suréchantillonnage à travers la technique SMOTE dont l'objectif est de compléter l'ensemble des données original par des observations synthétiques des classes minoritaires en utilisant les instances existantes.

L'évaluation des trois classificateurs montre que le KNN et le RF sont les meilleurs à prédire correctement l'état de la machine avec un score F1 respectivement égale à 99.77% et 99.78% contre 51% pour le LinearSVM.

CONCLUSION ET TRAVAUX FUTURS

6.1 Conclusion

Le problème de balourd est l'une des principales causes de panne des machines rotatives et il est souvent à l'origine de vibrations graves engendrant des défaillances et des arrêts coûteux à l'industrie. Par conséquent, une détection précoce du déséquilibre des machines tournantes est nécessaire pour éviter l'arrêt forcé pour une action de maintenance fréquente dans les processus industriels. Diverses procédures de surveillance et de détection de l'état sont utilisées pour diagnostiquer les défauts des machines tournantes sur la base de l'analyse de la signature vibratoire.

Ainsi, en collaboration avec Siemens Energy, la finalité de ce mémoire est d'établir un modèle de diagnostic et pronostic d'une turbine de puissance capable de détecter automatiquement son état de santé, c'est-à-dire de pouvoir prédire le plus rapidement possible si la machine est saine ou elle présente un déséquilibre. Dans cette perspective, on a porté un grand intérêt à la maintenance prédictive et notamment aux techniques d'apprentissage machine afin de réaliser les objectifs du projet. En effet, avec l'essor technologique et la complexité des systèmes industriels, le besoin d'apprentissage automatique augmente de jour en jour. Grâce à ces techniques, on peut former des algorithmes d'apprentissage automatique en leur fournissant l'énorme quantité de données et en les laissant explorer les données, construire les modèles et prédire automatiquement la sortie requise.

Dans ce mémoire, une étude comparative des techniques de surveillance de l'état pour le diagnostic des turbines de puissance a été entreprise y compris l'analyse temporelle et l'analyse fréquentielle des signaux vibratoires en montrant leurs limites.

Dans le Chapitre 2, une étude de la littérature et une exploration des concepts de base ont été menées dans le but d'identifier les techniques les plus utilisées pour mettre en œuvre un système de maintenance prédictive et de surveillance des machines. À partir de cette étude, on a établi notre méthodologie de recherche qu'on a expliquée dans le chapitre 3. Également, trois

modèles d'apprentissage automatique ont été sélectionnés pour une étude comparative, à savoir le KNN, le SVM et le RF. Dans la majorité des recherches étudiées, le SVM semble avoir les meilleurs résultats. Cependant, ceci n'a pas été validé pour notre étude de cas, car l'ensemble de données qu'on a utilisé était très large pour le SVM puisse fonctionner. Ce dernier a été donc remplacé par un SVM plus rapide qui est le LinearSVM.

Dans le Chapitre 4, on a exploité les outils de traitement de données et les anciennes méthodes de diagnostic et de détection des défauts à savoir l'analyse temporelle et l'analyse fréquentielle. Ces méthodes nous ont permis d'acquérir plus de connaissances sur les caractéristiques temporelles et fréquentielles des signaux vibratoires pour les deux états de la machine, équilibrée et non équilibrée.

Pour l'analyse temporelle, des descripteurs tels que le kurtosis, le RMS, la valeur maximale et d'autres ont été calculés et on a montré que celles-ci peuvent, en effet, indiquer que la machine présente un problème. Parmi les indicateurs étudiés, le RMS est le meilleur, ceci est en concordance avec les anciens travaux. Cependant, la limite de cette technique est qu'elle est incapable de donner des informations sur le type de défaut ou son origine. Pour l'analyse fréquentielle, on a appliqué la FFT. Étant donné que chaque défaut se manifeste à une fréquence caractéristique dans le spectre fréquentiel, cette technique est capable d'identifier la présence du balourd dans une machine rotative.

Dans le Chapitre 5, une étude comparative des trois algorithmes de classification supervisés a été faite à savoir Le LinearSVM, le RF et le KNN. Dans cette partie, les résultats obtenus ont montré que les algorithmes d'apprentissage automatique sont des outils puissants pour visualiser, représenter et détecter les défauts rapidement et automatiquement, à partir des données brutes. Les résultats ont démontré que le KNN et le RF ont eu des métrique de précision et de rappel élevés et donc sont plus performants que le LinearSVM, car ce dernier tolère les mauvaises classifications contrairement aux autres.

À partir des différents résultats obtenus lors de ce projet, on a pu confirmer que chaque classificateur a ses propres paramètres accordés. Ces derniers jouent un rôle très important dans la production des résultats de haute précision. Il faut donc chercher les paramètres optimaux pour pouvoir construire un bon classificateur. En ce qui concerne les différents algorithmes testés, le SVM est une technique puissante pour la classification des données. Malgré ses bonnes bases théoriques, sa grande précision de classification, et ses divers sucées constatés dans les travaux de recherche précédents, le SVM normal ne convient pas à la classification de grands ensembles tel est le cas de notre projet. En effet, la complexité d'entrainement du SVM dépend fortement de la taille de l'ensemble de données. La métrique d'évaluation des modèles la plus fréquemment rapportée est l'exactitude (Exactitude). Cette métrique peut être trompeuse lorsque les données sont déséquilibrées. Dans de tels cas, d'autres paramètres d'évaluation doivent être pris en compte tels que la matrice de confusion, la précision, le Rappel et la spécificité. Ainsi, lorsqu'il s'agit de données déséquilibrées, la décision finale par rapport à la sélection du modèle devrait envisager une combinaison de différentes mesures au lieu de se fonder sur une seule mesure.

On a aussi remarqué que pour les problèmes de classification dont l'ensemble des données est déséquilibré, les algorithmes de classification standard ne fonctionnent pas comme le cas du SVM. En effet, ces algorithmes tentent de minimiser le taux d'erreur plutôt que de se concentrer sur la classe minoritaire, ce qui donne une classification biaisée. La forêt aléatoire est un algorithme idéal pour traiter le déséquilibre extrême pour deux raisons principales. Premièrement, la possibilité d'incorporer des pondérations de classe dans le classificateur de forêt aléatoire le rend sensible aux coûts ; elle pénalise donc le classement erroné de la classe minoritaire. Deuxièmement, il combine la technique d'échantillonnage et l'apprentissage d'ensemble, par conséquent, en sous-échantillonnant la classe majoritaire et en faisant pousser des arbres sur un ensemble de données plus équilibré.

En outre, au départ, le modèle a été entrainé sur une seule base de données contenant deux classes seulement (équilibrée ou non équilibrée). Par la suite, on s'est rendu compte qu'il faut améliorer le modèle et le rendre plus général, en introduisant deux autres classes (déséquilibre

sévère et autres défauts). Cependant, cela a engendré un grand déséquilibre au niveau des classes. Pour remédier à ce problème, on a utilisé la méthode SMOTE. Cet algorithme crée de nouvelles instances de la classe minoritaire en créant des combinaisons convexes d'instances voisines. Le résultat montre que le KNN et le RF sont de bons classificateurs. De cette manière, on a réussi à concevoir un modèle de détection du déséquilibre des turbines de puissance, capable de détecter 4 classes qui sont : état équilibré, état déséquilibré, état de déséquilibre sévère et autre défaut ainsi on peut conclure qu'on a pu remplir les objectifs du projet.

6.2 Limites et travaux futurs

Le modèle n'a pas été testé sur d'autres scénarios, où la machine présente d'autres défauts bien spécifiques ayant plusieurs degrés de sévérité ce qui limite la performance du modèle quand on introduit de nouvelles données ne caractérisant pas les mêmes défauts avec lesquels le modèle a été entrainé et donc il faut entrainer à nouveau le modèle avec les nouvelles données. En effet, il a fallu réentrainer le modèle quand on a introduit les données de la machine ayant un déséquilibre sévère et la machine présentant un autre défaut autre que le déséquilibre. Ce qui n'est pas toujours pratique et prend du temps. Par conséquent, on suggère comme futur travail l'utilisation de l'apprentissage par transfert, en essayant essentiellement d'exploiter ce qui a été appris dans une tâche pour améliorer la généralisation dans une autre. En effet, l'apprentissage par transfert permet de réutiliser les connaissances afin que l'expérience acquise une fois puisse être appliquée à plusieurs reprises. Si on applique continuellement l'apprentissage par transfert dans la pratique d'apprentissage automatique, on peut obtenir un système d'apprentissage automatique tout au long de la vie qui peut tirer des connaissances d'une succession d'expériences de résolution de problèmes, à la fois sur une longue période et à partir d'une variété de tâches (Yang, Zhang, Dai, & Pan 2020). Également, estimer la durée de vie de l'équipement peut être une bonne perspective à envisager. Cependant, le défi de la prédiction RUL est que les données d'entrainement ne sont pas principalement étiquetées, et par conséquent, les algorithmes d'apprentissage machine supervisé ne seront plus applicable dans ce cas.

ANNEXE I

LES BIBLIOTHÉQUES PYTHON

```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestClassifier
from sklearn.model_selection import GridSearchCV
from sklearn.neighbors import KNeighborsClassifier
from sklearn.svm import SVC
from sklearn.metrics import confusion_matrix
import matplotlib.pyplot as plt
import numpy as np
import pandas as pd
import seaborn as sns
from sklearn.metrics import plot_precision_recall_curve
from sklearn.metrics import recall_score, f1_score, accuracy_score
from sklearn.svm import LinearSVC
from sklearn.preprocessing import MinMaxScaler
from time import time

before_path = "C:\Data_before_balance\combined.csv"
after_path = "C:\Data_after_balance\combined.csv"
model = "rfc"
use_finetuned_params = True
do_grid_search = False
plot_confusion = True

if model.lower() not in ["knn","svm","rfc"]:
    print("model name not supported try knn or svm or rfc")
    exit()
```

ANNEXE II

TRAITEMENT DES DONNÉES

```python
import pandas as pd
import glob
import os

data_dir = r'C:\Data_before_balance'
output_path = f"{data_dir}\combined.csv"

if os.path.exists(output_path): os.remove(output_path)

all_files = glob.glob(data_dir + "/*.csv")
dfs, columns_names = [], []
print("reading csv ...")
for fname in all_files:
    if fname.split("\\")[-1].split(".")[0][-3:].lower() == "13y": continue
    print(fname)
    df = pd.read_csv(fname)
    df.drop([0,1], axis=0, inplace=True)
    lst_columns = list(df.columns)
    lst_columns.remove("MANEOUVRE")
    df.drop(lst_columns,axis=1, inplace=True)
    df = df.reset_index(drop=True)
    dfs.append(df)
    columns_names.append(fname.split("\\")[-1].split(".")[0][-3:])
print("concatenating csv ...")
df_concatenated = pd.concat(dfs,axis=1)
df_concatenated.columns = columns_names
print("dumping file ..")
df_concatenated.to_csv(output_path,index=False)
```

ANNEXE III

MODELE KNN

```
elif model.lower() == "knn":

    print("Using knn ...")
    t0 = time()
    best_k, best_score = None, None
    for k in range(8,10):
        classifier = KNeighborsClassifier(n_neighbors=k, weights='uniform', algorithm='auto', leaf_size=30, p=2,
                                  metric='minkowski', metric_params=None, n_jobs=-1)
        classifier.fit(x_train, y_train)
        accKnn = classifier.score(x_test, y_test)
        if best_k is None or best_score < accKnn:
            best_k, best_score = k, accKnn
    print(f"score of best knn: {best_score} using {best_k} nearest neighobrs")
    print(f"train time : {time() - t0}")
    t0 = time()
    y_pred_knn = classifier.predict(x_test)
    print(f"inference time {(time() - t0) / len(x_test)}")
    print('Accuracy: %.3f' % accuracy_score(y_test, y_pred_knn, normalize=True))
    print('F1 Score pour la classe 2: %.3f' % f1_score(y_test, y_pred_knn, average="binary", pos_label='2' ))
    print('F1 Score pour la classe 1: %.3f' % f1_score(y_test, y_pred_knn, average="binary", pos_label='1' ))
    print('Recall pour la classe 2: %.3f' % recall_score(y_test, y_pred_knn, average="binary", pos_label='2'))
    print('Recall pour la classe 1: %.3f' % recall_score(y_test, y_pred_knn, average="binary", pos_label='1'))
    # Plotting Precision-Recall Curve
    disp = plot_precision_recall_curve(classifier, x_test, y_test)
```

ANNEXE IV

MODELE SVM

```
elif model.lower() == "svm":
    print("Using svm ...")

    if use_finetuned_params:

        #classifier = SVC(C=1, gamma=0.05, kernel='linear',
random_state=0)
        t0 = time()
        classifier = LinearSVC(random_state=0, tol=0.0001)
        classifier.fit(x_train, y_train)
        accRfc = classifier.score(x_test, y_test)
        print(f"score : {accRfc}")
        print(f"temps d'entrainement : {time() - t0}")
        t0 = time()
        y_pred_svm = classifier.predict(x_test)
        print(f"temps d'inférence {(time() - t0) / len(x_test)}")
        print('Exactitude: %.3f' % exactitude_score(y_test,
y_pred_svm, normalize=True))
        print('F1 Score pour la classe 2: %.3f' % f1_score(y_test,
y_pred_svm, average="binary", pos_label='2'))
        print('F1 Score pour la classe 1: %.3f' % f1_score(y_test,
y_pred_svm, average="binary", pos_label='1'))
        print('Rappel pour la classe 2: %.3f' % Rappel_score(y_test,
y_pred_svm, average="binary", pos_label='2'))
        print('Rappel pour la classe 1: %.3f' % Rappel_score(y_test,
y_pred_svm, average="binary", pos_label='1'))
        disp = plot_precision_Rappel_curve(classifier, x_test,
y_test)
    if do_grid_search:
        svc = SVC()
        parameters = {
            "C": [1, 10, 50],
            'gamma': [0.05, 0.1, 0.5],
            'kernel': ["rbf", "linear" , "poly"]
        }
        classifier = GridSearchCV(svc, parameters, n_jobs=-1, cv=4)
        classifier.fit(x_train, y_train)
        print(f"score after grid search {classifier.best_score_}")
        print(f"best params {classifier.best_params_}")
        print(f"temps d'entrainement : {time() - t0}")
```

ANNEXE V

MODELE RF

```
if model.lower() == "rfc":
    print("Using random forest classifier ...")
    if use_finetuned_params:
        t0 = time()
        classifier = RandomForestClassifier(max_depth=20, n_estimators=15, random_state=0)
        classifier.fit(x_train , y_train)
        accRfc = classifier.score(x_test,y_test)
        print(f"score : {accRfc}")
        print(f"train time : {time()-t0}")
    if do_grid_search:
        param_grid = {
            'n_estimators': [2 , 5, 10, 15],
            'max_depth': [2, 9, 15 , 20]
        }
        rfc = RandomForestClassifier()
        classifier = GridSearchCV(rfc, param_grid, cv=5)
        t0 = time()
        classifier.fit(x_train, y_train)
        print(f"score after grid search {classifier.best_score_}")
        print(f"best params {classifier.best_params_}")
        print(f"train time : {time()-t0}")
    t0 = time()
    y_pred_rfc = classifier.predict(x_test)
    print(f"inference time {(time()-t0)/len(x_test)}")
    print('Accuracy: %.3f' % accuracy_score(y_test, y_pred_rfc , normalize = True))
    print('F1 Score pour la classe 2: %.3f' % f1_score(y_test, y_pred_rfc, average="binary", pos_label='1'))
    print('F1 Score pour la classe 1: %.3f' % f1_score(y_test, y_pred_rfc, average="binary", pos_label='1'))
    print('Recall pour la classe 2: %.3f' % recall_score(y_test, y_pred_rfc, average="binary", pos_label='1'))
    print('Recall pour la classe 1: %.3f' % recall_score(y_test, y_pred_rfc, average="binary", pos_label='1'))
    # Plotting Precision-Recall Curve
```

ANNEXE VI

MATRICE DE CONFUSION

```
y_pred = classifier.predict(x_test)

test_confu_matrix = confusion_matrix(y_test, y_pred)
# Test confusion matrix.
state = ['1=Balanced', '2=Unbalanced']
plt.figure(2, figsize=(18, 8))
plt.suptitle(f"Plot of {model.upper()}")
plt.subplot(121)
sns.heatmap(test_confu_matrix, annot=True, fmt="d",
            xticklabels=state, yticklabels=state, cmap="Blues")
plt.title('Test Confusion Matrix')
plt.xlabel('Predicted')
plt.ylabel('True')
plt.subplot(122)
sns.heatmap(test_confu_matrix / 75, annot=True,
            xticklabels=state, yticklabels=state, cmap="Blues")
plt.title('Test Confusion Matrix (in %age)')
plt.xlabel('Predicted')
plt.ylabel('True')

plt.show()
```

ANNEXE VII

TABLEAU RÉSUMANT LA REVUE DE LA LITTÉRATURE

Auteurs	Thèmes	Approches	Techniques
(Qiao & Lu, 2015) (Jin et al., 2014) (Jie et al., 2015) (Salem et al., 2014) (Liu et al., 2015) (Thyago P et al., 2019)	Détection des défauts dans les machines tournantes	Approche basée sur les données	Signal vibratoire
(Mobley, 2002) (Lu et al., 2012) (Mendonca et al., 2015)	Diagnostic des défauts mécanique	Approche basée sur les données	Signal électrique
(Kedadouche et al., 2012)	Détection des défauts pour équipements à bases vitesses	Approche basée sur les données	Signal acoustique
(Tahsin & Jens, 2008) (J. Mathew, 1998)	Diagnostic des petits défauts des roulements à rouleaux.	Approche basée sur les données	Analyse temporelle Kurtosis RMS PEAK
(Hongyu et al., 2003) (McFadden & Smith, 1985) (Kim Y W, 1995) (Feng et Liang, 2014)	Diagnostic des défauts de la boîte de vitesses	Approche basée sur les données	Analyse spectrale non paramétrique classique, l'analyse en composantes principales, l'analyse temps-fréquence conjointe. La transformée en ondelettes discrète.
(Luo et al., 2020; Montero et al., 2020; Sufiyan et al., 2021).	Diagnostic et Pronostic	Approche basée sur les connaissances et les expériences	
(Wu et al., 2017) (Lei et al., 2016).	Détection des défaillances	Approche basée sur les modèles physiques	Modélisation mathématique et statique des défaillances

Auteurs	Thèmes	Approches	Techniques
(Wuest et al., 2016) (Canizo et al., 2017) (Baptista et al., 2018)	Détection des défauts	Approche basée sur les données	Techniques d'intelligence artificielle et apprentissage machine
(Luo et al., 2020)	Diagnostic et pronostic	Approche hybride	Digital Twin (DT)
(Hsu et al., 2020)	Diagnostiquer les défaillances des éoliennes	Approche basée sur les données	Techniques de contrôle statistique des processus et d'apprentissage automatique
(Paudyal, 2019)	Identification et classification de l'état des machines.	Approche basée sur les données	KNN, FFT et CWT
(Baptista et al., 2018) (U. Shafi et al., 2018)	Planification des horaires de maintenance	Approche basée sur les données	SVM, KNN, DT, RF
(Soualhi, Medjaher, & Zerhouni, 2015)	Évaluer des états de dégradation et estimation de la RUL des roulements dégradés	Approche basée sur les données	Signal vibratoire+ analyse fréquentielle+ SVM+ SVR
(O. R. Seryasat, 2010.)	Détecter les défauts de roulement	Approche basée sur les données	Signal vibratoire + analyse temporelle+ SVM multi-classe
(Canizo et al., 2017), (Kusiak & Verma, 2011),	Diagnostic des éoliennes	Approche basée sur les données et approches basées sur les expériences	RF
(Wu et al., 2017)	Prédiction de l'usure des outils	Approche basée sur les données	Signal vibratoire + signal acoustique ANN, SVM et RF

LISTE DE RÉFÉRENCES BIBLIOGRAPHIQUES

Adankon, M. M. (2005). *Optimisation de ressources pour la sélection de modele des svm, (Mémoire de Maitrise,* (École de technologie supérieure, Montréal, QC).

appen. (2021). What is Training Data? Repéré à https://appen.com/blog/training-data/

Baptista, M., Sankararaman, S., de Medeirosc, I. P., Nascimento Jr, C., Prendingere, H., & Henriquesa, E. M. P. (2018). Forecasting fault events for predictive maintenance using data-driven techniques and ARMA modeling *Comput. Ind. Eng, 115*, 41-53.

Borad, A. (2019). addressing challenges associated with imbalanced datasets in machine learning. Repéré

Cambridge University Press. (2021). Prédictif. Dans *Dictionnaire Cambridge*

Canizo, M., Onieva, E., Conde, A., Charramendieta, S., & Trujillo, S. (2017). Real-time predictive maintenance for wind turbines using Big Data frameworks. *2017 IEEE International Conference on Prognostics and Health Management (ICPHM)*. doi: 10.1109/ACCESS.2020.2968615

CFI Education Inc. Skewness. Repéré à https://corporatefinanceinstitute.com/resources/knowledge/other/skewness/

Çiğdem, P. D., & Mehmet, S. Ö. (2018). Wavelet transform and signal denoising using Wavelet method. (1-4). doi: 10.1109/SIU.2018.8404418

Data Science. (2019). What is Confusion Matrix and Advanced Classification Metrics? Repéré à https://manisha-sirsat.blogspot.com/2019/04/confusion-matrix.html

DataLogger Inc. (2016). Basic Techniques of Vibration Measurement and Diagnosis. Repéré à https://www.dataloggerinc.com/resource-article/basic-techniques-of-vibration-measurement-and-diagnosis/

Desouki, M., Sassi, S., Renno, J., & Gowid, S. A. (2020). Dynamic Response of a Rotating Assembly under the Coupled
Effects of Misalignment and Imbalance. *Shock and Vibration*.

Deutsch, J., & He, D. (2018). Using Deep Learning-Based Approach to Predict Remaining Useful Life of Rotating Components. *EE Transactions on Systems, Man, and Cybernetics: Systems*.

Donovan, R. (2019). Why is True-peak measurement a better method for detecting bearing damage? Repéré

Fan, C., Xiao, F., Yan, C., Liu, C., Li, Z., & Wang, J. (2019). A novel methodology to explain and evaluate data-driven building energy performance models based on interpretable machine learning. *Applied Energy, 235*, 1551 - 15601.

Feng, Z., & Liang, M. (2014). Fault diagnosis of wind turbine planetary gearbox under nonstationary conditions via adaptive optimal kernel time–frequency analysis. *Renewable Energy*, 468-477.

Hongyu, Y., Joseph, M., & Lin, M. (2003). Vibration feature extraction techniques for fault diagnosis of rotating machinery : a
literature survey. *Asia-Pacific Vibration Conference*.

Hoppenstedt, B., Pryss, R., Stelzer, B., Meyer-Brötz, F., Kammerer, K., Treß, A., & Reichert, M. (2018). Techniques and Emerging Trends for State of the Art Equipment Maintenance Systems—A Bibliometric Analysis. *Appl. Sci* doi: https://doi.org/10.3390/app8060916

Hsu, J.-Y., Wang, Y.-F., Lin, K.-C., Chen, M.-Y., & Hsu, J. (2020). Wind Turbine Fault Diagnosis and Predictive Maintenance Through Statistical Process Control and Machine Learning. *Computer Science IEEE Access, 8*.

Hussai, B., Chan, H. W. E., & Patanwala, S. (2021). Surviving in a Random Forest with Imbalanced Datasets. Repéré

ISO-1925. (2001). *Vibrations mécaniques - Équilibrage - Vocabulaire*.

Jaber, W. (2018). Detection et diagnostic des defaillances des procedes chimiques a l'aide des reseaux neuronaux artificiels v2.0. doi: 10.13140/RG.2.2.24623.38567

Jaouher, B. A., Nader, F., Saidi, L., & al, e. (2015). Application of empirical mode decomposition and artificial neural network for automatic bearing fault diagnosis based on vibration signals. *Applied Acoustics, 89*, 16-27.

Jie, B., Hou, L., & Yongguang, M. (2015). Machine fault diagnosis using industrial wireless sensor networks and support vector machine *Proc. 12th IEEE Int. Conf. Electron. Meas. Instrum, 1*, 153-158.

Jin, C., Zhao, W., Liu, Z., Lee, J., & He, X. (2014). A vibration-based approach for diesel engine fault diagnosis. *Proc. Int. Conf. Prognostics Health Manage*, 1-9.

Kamat, P., & Sugandhi, R. (2020). Anomaly Detection for Predictive Maintenance in Industry 4.0- A survey. *E3S Web Conf., 170*, 02007. Repéré à https://doi.org/10.1051/e3sconf/202017002007

Kapil, B., & Kiran, B. (2018). Considerations for artificial intelligence and machine learning: Approaches and use cases. *IEEE Aerospace Conference*. doi: 10.1109/AERO.2018.8396488

Kedadouche, M., Thomas, M., & Tahan, A. (2012). Surveillance des roulements par emission acoustique: Étude comparative avec les techniques vibratoires pour détection précoce

Kim, B.-H., Velas, J. P., & Lee, K. Y. (2006). Semigroup Based Neural Network Architecture for Extrapolation of Enthalpy in a Power Plant. *IEEE Aerospace Conference, 39*(7), 309-314,. doi: https://doi.org/10.3182/20060625-4-CA-2906.00058.

Kim Y W, e. a. (1995). Analysis and processing of shaft angular velocity signals in rotating machinery for diagnostic applications. in Acoustics,
Speech, and Signal Processing. *ICASSP-95, vol.5*, 2971-2974.

Kirasich, K., Smith, T., & Sadler, B. (2018). Random Forest vs Logistic Regression: Binary Classification for Heterogeneous Datasets. *SMU Data Science Review, 1*.

L.S. Dalenogare, G. B. B., N.F. Ayala, A.G. Frank. (2019). The expected contribution of Industry 4.0 technologies for industrial performance.

Landolsi, F. (s.d.). *Etude-des-principaux-defauts [Note de cours]*. Repéré à http://silanus.fr/sin/formationSTI2D/ET22A-B/ET22A/Ressources/Etude-des-principaux-defauts.pdf

Lebold, M., Mcclintic , K., Campbell , R., Byington , C., & Maynard, K. (2000). Review of vibration analysis methods for gearbox diagnostics and prognostics. *54th meeting of the society for machinery failure prevention technology*, 623-634.

Lei, Y., Li, N., Gontarz, S., Lin, J., Radkowski, S., & Dybala, J. (2016). A model-based method for remaining useful life prediction of machinery
IEEE Trans. Reliab, 65, 1314-1326.

Liu, W. Y., Tang, B., Han, J., Lu, X. N., Hu, N. N., & He, Z.-x. (2015). The structure healthy condition monitoring and fault diagnosis methods in wind turbines: A review. *Renewable & Sustainable Energy Reviews, 466-472*.

Lu, D., Gong, X., & Qiao, W. (2012). Current-based diagnosis for gear tooth breaks in wind turbine gearboxes. *Proc. Energy Convers. Congr. Expo, 11*, 3780-3786.

Luo, W., Hu, T., Ye, Y., Zhang, C., & Wei, Y. (2020). A hybrid predictive maintenance approach for CNC machine tool driven by Digital Twin. *Robotics and Computer-Integrated Manufacturing, 65*, 101974.

M. Mohri, A. R., A. Talwalkar. (2018). Foundations of Machine Learning (Second Edition éd.). London, England: The MIT Press.

Manish Vishwakarma, R. P., V. Harshlata, P. Rajput,. (2017). Vibration Analysis & Condition Monitoring for Rotating Machines: A Review. *Materials Today: Proceedings, 4*(2), 2659-2664. doi: https://doi.org/10.1016/j.matpr.2017.02.140. Repéré à https://www.sciencedirect.com/science/article/pii/S2214785317303401

McFadden, P. D., & Smith, J. D. (1985). The vibration produced by multiple point defects in a rolling element bearing. *Journal of Sound and Vibration,, 98(2)*, 263-273.

.

Mendonca, P., Bonaldi, E., De Oliviera, L., Torres, G. L., Borged Da Silva, J., Borges Da Silva, L., . . . Santana, W. (2015). Development of a reduced-model laboratory for testing predictive fault system in internal combustion engines. *Proc. IEEE 10th Int. Symp. Diagnostics Elect. Mach. Électron de puissance. Drives,*, 428-434.

Mobley, R. K. (2002). An Introduction to Predictive Maintenance. *Elsevier*.

Montero, J. J., Jimenez, S., Rob, S., Grabot, M. S., & Bernard, V. (2020). Towards multi-model approaches to predictive maintenance: A systematic literature survey on diagnostics and prognostics.

Morgan R. Frank, V. O. P. A., James E. Bessen, Erik Brynjolfsson, Manuel Cebrian, David J. De. (2019). Toward understanding the impact of artificial intelligence on labor.

Navlani, A. (2018). Understanding Random Forests Classifiers in Python. Repéré à https://www.datacamp.com/community/tutorials/random-forests-classifier-python

O'connor, R. (2021). Duenas EVA Survey , Machine Health.

O. R. Seryasat, M. A. S., F. Honarvar et A. Rahmani,. (2010.). Multi-fault diagnosis of ball bearing based on features extracted from time-domain and multi-class support vector machine (MSVM. *Proc. IEEE Int. Conf. Syst. Man Cybern*, 4300-4303.

Paudyal, S. (2019). *Classification Of Rotating Machinery Fault Using Vibration Signal* (University of North Dakota, Theses and Dissertations).

Qiao, W., & Lu, D. (2015). A survey on wind turbine condition monitoring and fault diagnosis — Part II: Signals and signal processing methods. *IEEE Trans. Ind. Electron., 62*, 6546-6557.

Radhya, S., John, G. B., & Muhammad Intizar, A. (2020). Big data and stream processing platforms for Industry 4.0 requirements mapping for a predictive maintenance use case. *Journal of Manufacturing Systems, 54*, 138-151.

Rioux, S. (2016). L'industrie 4.0, c'est quoi ? Repéré à https://www.andromediatech.com/lindustrie-4-0-cest-quoi/

Russel, S., & Norvig, P. (2012). Artificial intelligence—A modern approach, 3rd edition.

Saleem, M., Garikapati, D., & Rama Surya Satyanarayana, M. (2012). Detection of Unbalance in Rotating Machines Using Shaft Deflection Measurement during Its Operation. *IOSR Journal of Mechanical and Civil Engineering., 3*(3), 2278-1684

Salem, A., Abu-Siada, A., & Islam, S. (2014). Condition Monitoring Techniques of the Wind Turbines Gearbox and Rotor. *International Journal of Electrical Energy*, 53-56.

Seyyed Mohammad Javadi, M., & Asadollah, N. (2021). A novel imbalanced data classification approach using both under and over sampling. *Bulletin of Electrical Engineering and Informatics, 10*. doi: 10.11591/eei.v10i5.2785

Siemens Industrial. (2005). SGT-200 Industrial Gas Turbine *Industrial application*. Repéré à china-power-contractorsite Web |URL| doi:DOI

Smith, S. (2014). Balancing rotating elements in machines. *Cutting Tool Engineering*.

Soualhi, A., Medjaher, K., & Zerhouni, N. (2015). Bearing health monitoring based on Hilbert-Huang transform support vector machine and regression. *IEEE Trans. Instrum. Meas., 64*, 52-62.

Steen, D. (2020). Precision-Recall curves. Repéré à https://medium.com/@douglaspsteen/precision-recall-curves-d32e5b290248

Sufiyan, S., Abid, H., Shashi, B., Mohd, J., Tarun, G., & Manoj, M. (2021). Data science applications for predictive maintenance and materials science in context to Industry 4.0. *Materials Today: Proceedings 45*, 4898-4905.

Sufiyan Sajida, A. H., Shashi Bahlb, Mohd Javaida, Tarun Goyalc, Manoj Mittalc. (2021). Data science applications for predictive maintenance and materials science in context to Industry 4.0. *Materials Today: Proceedings 45*, 4898-4905.

Tahsin, D., & Jens, S. (2008). New Time Domain Method for the Detection of Roller Bearing Defects
Proceedings of International Conference on Condition Monitoring & Machinery Failure Prevention Technologies CM

Thyago P, C., Fabrízzio, S., Roberto, V., Roberto da P, F., João P, B., & Alcalá, S. G. S. (2019). A systematic literature review of machine learning methods applied to predictive maintenance. *Computers & Industrial Engineering November 2019, 106024, 137*.

U. Shafi et al. (2018). Vehicle remote health monitoring and prognostic maintenance system *J. Adv. Transp, 1*, 1-10.

Wang, W., Di Maioa, F., & Zio, E. (2017). Three-loop Monte Carlo simulation approach to Multi-State Physics Modeling for system reliability assessment. *Reliability Engineering & System Safety, 167*, 276-289.

Wanga, H. P., Ghani, N., & Kalegele, K. (2017). On the Development of Machine Learning – Based Application Framework for Enhancing Performance of Livestock Mobile Application Systems in Poor Internet Service Areas. *International Journal of Computer Applications 179*.

Wu, D., Jennings, C., Terpenny, J., Gao, R., & Kumara, S. (2017). A Comparative Study on Machine Learning Algorithms for Smart Manufacturing: Tool Wear Prediction Using Random Forests. *ASME J. Manuf. Sci. Eng, 139*, 071018.

Wuest, T., Weimer, D., Irgens, C., & Thoben, K.-D. (2016). Machine learning in manufacturing: advantages, challenges, and applications. *Production & Manufacturing Research*. doi: 10.1080/21693277.2016.1192517

Yang, Q., Zhang, Y., Dai, W., & Pan , S. J. (2020). *Transfer Learning*.

Yang., J. (2003). An anti-aliasing algorithm for discrete wavelet transform. *Mechanical Systems and Signal Processing, 17*, 945-954.