BIBLIOTHÈQUE
CÉGEP DE L'ABITIBI-TÉMISCAMINGUE
UNIVERSITÉ DU QUÉBEC EN ABITIBI-TÉMISCAMINGUE

Mise en garde

La bibliothèque du Cégep de l'Abitibi-Témiscamingue et de l'Université du Québec en Abitibi-Témiscamingue (UQAT) a obtenu l'autorisation de la personne autrice de ce document afin de diffuser, dans un but non lucratif, une copie de son œuvre dans Depositum, site d'archives numériques, gratuit et accessible à tous. L'autrice ou l'auteur conserve néanmoins ses droits de propriété intellectuelle, dont son droit d'auteur, sur cette œuvre.

Warning

The library of the Cégep de l'Abitibi-Témiscamingue and the Université du Québec en Abitibi-Témiscamingue (UQAT) obtained the permission of the author to use a copy of this document for nonprofit purposes in order to put it in the open archives Depositum, which is free and accessible to all. The author retains ownership of the copyright on this document.

Université du Québec en Abitibi-Témiscamingue

MAINTENANCE PRÉDICTIVE EN TEMPS RÉEL DE L'APPAREILLAGE DYNAMIQUE DANS LES USINES INTELLIGENTES

MÉMOIRE
PRÉSENTÉ
COMME EXIGENCE PARTIELLE
DE LA MAÎTRISE EN INGÉNIERIE

Par
MARWEN CHOUK

[Mars] [2026]

REMERCIEMENTS

Avant d’entamer ce rapport, c’est avec un grand plaisir que je réserve ces quelques lignes en signe de gratitude et de profonde reconnaissance à tous ceux qui, de près ou de loin, ont contribué à l’aboutissement de ce travail.

Je souhaite exprimer ma profonde gratitude envers mon directeur de recherche, le **Pr. FOUAD SLAOUI HASNAOUI**, pour sa guidance précieuse, sa motivation infaillible et le temps qu'il a généreusement consacré tout au long de cette période.

Sans son soutien inestimable, ses judicieuses recommandations et ses conseils, la réalisation de ce mémoire aurait été impossible.

Je tiens à témoigner toute ma reconnaissance à tous le personnel et la Direction de l'UQAT pour m'avoir offert l'opportunité de vivre cette expérience enrichissante.

Merci pour l’excellent accueil que vous m’avez réservé et merci d’avoir veillé directement à mon intégration et au bon déroulement de mes études. Je tiens également à remercier chaleureusement tous les professeurs du programme de maîtrise en ingénierie, concentration en génie électrique, pour leur formation de qualité et leurs conseils précieux au quotidien. Merci d’avoir grandement contribué à la réussite de mon projet de recherche.

Mes remerciements sont adressés également aux membres du jury pour leur bienveillance à vouloir évaluer mon travail, tout en espérant qu’ils trouveront dans ce rapport les qualités de clarté et de motivation qu’ils attendent.

Un grand merci à ma famille, pour l’amour qu’elle m’apporte au quotidien, les conseils ainsi que le soutien inconditionnel, à la fois moral et économique, m’ont permis de réaliser les études que j’ai toujours souhaité poursuivre.

DÉDICACE

À mes chers parents Lotfi et Ibtissem,

Je ne saurai exprimer mon respect, mon amour et ma considération à travers une simple dédicace. Vos prières et vos bénédictions m'ont été d'un grand secours aux différents stades de ma vie. Autant de phrases aussi expressives soient-elles ne sauraient montrer le degré d'amour et d'affection que j'éprouve pour vous.

Vous m'avez comblé avec votre tendresse et affection tout au long de mon parcours. Puisse le tout puissant vous donner santé, bonheur et longue vie. Que ce travail soit l'exaucement de vos vœux tant allégués, le fruit de vos innombrables sacrifices et de votre soutien infaillible.

À ma sœur Safa,

En signe de l'affection et du grand amour que je te porte, les mots sont insuffisants pour exprimer ma profonde affection et mon attachement indéfectible. Je te dédie ce travail avec mes vœux de réussite, de prospérité et de bonheur. Que Dieu te garde et te protège.

À mes chers amis,

En souvenir des moments merveilleux que nous avons passés et aux liens solides qui nous unissent. Un grand merci pour votre soutien, vos encouragements, votre aide.

J'ai trouvé en vous le refuge de mes chagrins et mes secrets. Avec toute mon affection et estime, je vous souhaite beaucoup de réussite et de bonheur, autant dans votre vie professionnelle que privée.

À ma famille au Canada,

Aux personnes qui sont devenues une famille pour moi, je tiens à vous remercier pour votre amour et votre soutien. Je vous suis toujours reconnaissante de tout cœur.

À tous mes professeurs,

À vous chers lecteurs,

TABLE DES MATIÈRES

REMERCIEMENTS II
DÉDICACE III
TABLE DES MATIÈRES IV
LISTE DES FIGURES VIII
LISTE DES TABLEAUX X
LISTE DES SIGLES ET DES ABRÉVIATIONS XI
RÉSUMÉ XII
ABSTRACT XIII
INTRODUCTION 14
0.1 Mise en contexte 14
0.2 Problématique 16
0.3 Objectifs 17
0.3.1 Objectif général 17
0.3.2 Objectifs spécifiques 17
0.4 Hypothèses 18
0.5 Originalité de la recherche 19
0.6 Méthodologie 20
0.7 Organisation du mémoire 21
1. CONCEPTS DE BASE ET REVUE DE LA LITTÉRATTURE 22
1.1 Introduction 22
1.2 Concepts de base 22
1.2.1 L'évolution industrielle et l'industrie 4.0 22
1.2.2 La maintenance industrielle : définition et typologies 23
1.2.3 Analyse vibratoire au service de la maintenance prédictive 23
1.2.4 Processus de mise en œuvre d'une maintenance prédictive basée sur les données 24
1.3 Revue de la littérature 25
1.3.1 Maintenance prédictive 25
1.3.2 Analyse des données vibratoires 26

1.3.3 L'intelligence artificielle : Apprentissage Automatique (Machine Learning) 30
1.3.4 L'intelligence artificielle : Apprentissage Profond (Deep Learning) 35
1.3.5 ADASYN (Adaptive Synthetic Sampling) 36
1.3.6 CTGAN (Conditional Tabular Generative Adversarial Network) 37
1.4 Conclusion 39
2. MÉTHODOLOGIE DE RECHERCHE 41
2.1 Introduction 41
2.2 Démarche de recherche 41
2.2.1 Collecte des données 42
2.2.2 Analyse des données vibratoires 44
2.2.3 Prédiction du défaut en utilisant l'apprentissage automatique 45
2.2.4 Prédiction du défaut en utilisant l'apprentissage profond 46
2.3 Conclusion 48
3. DIAGNOSTIC DE DÉFAUT AVEC L'ANALYSE TEMPORELLE, FRÉQUENTIELLE ET TEMPS-FREQUENCE 49
3.1 Introduction 49
3.2 Description des données 49
3.3 Analyse temporelle 53
3.3.1 Visualisation d'une composante temporelle 53
3.3.2 Extraction des descripteurs statistiques 54
3.3.3 Discussion 55
3.4 Analyse fréquentielle 56
3.4.1 Estimation des densités spectrales de puissance 56
3.4.2 Spectres de transformée de Fourier (FFT) 60
3.4.3 Discussion 61
3.5 Analyse temps-fréquence par transformée en ondelettes continues 62
3.5.1 Spectre CWT 62
3.5.2 Scalogrammes temps-fréquence 63
3.5.3 Discussion 65
3.6 Conclusion 65

4. PRÉDICTION DU DÉFAUT EN UTILISANT L'APPRENTISSAGE AUTOMATIQUE 67
4.1 Introduction 67
4.2 Étapes de construction des modèles de prédiction 67
4.3 Construction des modèles de prédiction 71
4.4 Résultats des modèles de prédiction 76
4.4.1 Matrice de confusion 76
4.4.2 F1-Score 79
4.4.3 Le rappel 80
4.4.4 Le ROC AUC 81
4.4.5 La précision 82
4.4.6 L'exactitude 83
4.5 Discussion 84
4.6 Conclusion 86
5. PRÉDICTION DU DÉFAUT EN UTILISANT L'APPRENTISSAGE PROFOND 87
5.1 Introduction 87
5.2 Étapes de construction des modèles de prédiction 87
5.3 Construction des modèles de prédiction 89
5.4 Résultats des modèles de prédiction 94
5.4.1 Matrice de confusion 94
5.4.2 F1-Score 95
5.4.3 Le rappel 96
5.4.4 Le ROC AUC 97
5.4.5 La précision 98
5.4.6 L'exactitude 99
5.5 Discussion 99
5.6 Déploiement dans une maintenance prédictive 101
5.7 Conclusion 103
CONCLUSION ET TRAVAUX FUTURS 106
ANNEXE A –CAPTEUR BOSCH DATASHEET 112

ANNEXE B –EXPLOITATION DE LA BASE DES DONNÉES 114
ANNEXE C –LA CONFIGURATION PYTHON UTILISÉE POUR L'ANALYSE VIBRATOIRE (PARAMÈTRE TOOL_ID).....115
ANNEXE D –SPECTRES FFT ET MARQUEURS D'AMPLITUDE POUR LES AXES Y ET Z DE LA MACHINE M01 (OUTIL 07) 116
ANNEXE E –SPECTRES CWT ET SCALOGRAMMES POUR LES AXES Y ET Z DE LA MACHINE M01 (OUTIL 07) 117
ANNEXE F –BIBLIOTHÈQUE PYTHON ET CHARGEMENT DES DONNÉES POUR L'APPRENTISSAGE AUTOMATIQUE 118
ANNEXE G –BIBLIOTHÈQUE PYTHON ET CHARGEMENT DES DONNÉES POUR L'APPRENTISSAGE PROFOND.....119
ANNEXE H –L'INTERFACE DE L'APPLICATION À DÉPLOYER 120
LISTE DE RÉFÉRENCES 121

LISTE DES FIGURES

Figure 1 Industrie 4.0 : panorama des piliers technologiques 14
Figure 2 Les quatre révolutions industrielles 22
Figure 3 Chaîne de traitement d'une maintenance prédictive basée sur les données 25
Figure 4 Schémas simplifié de l'approche basée sur l'apprentissage automatique 46
Figure 5 Schémas simplifié de l'approche basée sur l'apprentissage profond 47
Figure 6 Schéma simplifié du système d'acquisition des données 50
Figure 7 Exemple pour une répartition des données 52
Figure 8 Les signaux vibratoires avant et après équilibrage 53
Figure 9 PSD médianes des états Good et Bad machine M01, axe X, OP 07 57
Figure 10 Figure similaire pour M02 M03 57
Figure 11 Zoom autour de 200, 400, 600 et 800 Hz pour M01, axe X, OP 07 58
Figure 12 Spectres FFT d'amplitude des signaux de vibration pour l'OP 07, machine M01, axe X 60
Figure 13 Spectre moyen CWT Good/Bad pour M01, OP 07, axe X 62
Figure 14 Scalogrammes Good et Bad pour M01, OP 07, axe X 64
Figure 15 Pipeline global du modèle de prédiction ML 69
Figure 16 Matrice de confusion générale pour une classification binaire 70
Figure 17 Exemple de minorité de données "Bad" 71
Figure 18 Rééchantillonnage des données avec ADASYN 73
Figure 19 Validation croisée stratifiée (K=5) 74
Figure 20 Lecture d'état d'apprentissage des modèles 75
Figure 21 La chaîne de construction des modèles de prédiction 76
Figure 22 Matrice de confusion RF et XGBoost sans équilibrage 77
Figure 23 Matrice de confusion des modèles avec rééchantillonnage 78
Figure 24 F1-score des modèles d'apprentissage automatique 79
Figure 25 Rappel des modèles d'apprentissage automatique 80
Figure 26 ROC AUC des modèles d'apprentissage automatique 81
Figure 27 Précision des modèles d'apprentissage automatique 82
Figure 28 Exactitude des modèles d'apprentissage automatique 83

Figure 29 Radar des performances des six classificateurs testés 85
Figure 30 Pipeline global de modèle de prédiction DL 89
Figure 31 Début de chargement et fenêtrage des données brutes..... 89
Figure 32 Nombre des fenêtres et taille des données d'entraînement\Test..... 90
Figure 33 Équilibrage des données avec CTGAN..... 90
Figure 34 L'architecture du modèle sans équilibrage 91
Figure 35 L'architecture du modèle avec ADASYN..... 92
Figure 36 L'architecture du modèle avec CTGAN..... 93
Figure 37 La chaine de construction des modèles d'apprentissage profond 94
Figure 38 Matrice de confusion de 3 approches 95
Figure 39 Comparaison du F1-Score pour les 3 versions 96
Figure 40 Comparaison du rappel des 3 modèles..... 97
Figure 41 Comparaison ROC AUC des 3 versions 97
Figure 42 Comparaison de précision des 3 modèles 98
Figure 43 Comparaison d'exactitude des 3 versions du modèle 99
Figure 44 Radar de performance des 3 variantes de modèle..... 100
Figure 45 Prédiction du défaut (Good) à travers l'interface 102
Figure 46 Prédiction du défaut (Bad) à travers l'interface..... 103

LISTE DES TABLEAUX

Tableau 1 Démarche de recherche 42
Tableau 2 Opérations d'usinage collectées à partir des machines M01, M02 et M03 51
Tableau 3 Indicateurs temporels par machine axe X OP 07 (Good vs Bad) 54
Tableau 4 Indicateurs temporels par machine axe Y OP 07 (Good vs Bad) 54
Tableau 5 Indicateurs temporels par machine axe Z OP 07 (Good vs Bad) 55
Tableau 6 Marqueurs spectraux des pics pour M01, axe X, OP 07 59
Tableau 7 Marqueurs spectraux issus de la FFT pour l'OP 07, machine M01, axe X 61
Tableau 8 Marqueurs CWT pour M01, OP 07, axe X 63
Tableau 9 Tableau comparatif entre les différents modèles 84
Tableau 10 Tableau comparatif entre les différentes versions 99

LISTE DES SIGLES ET DES ABRÉVIATIONS

TCM	: Tool Condition Monitoring
CNC	: Computer Numerical Control
MEMS	: Micro Electro Mechanical Systems
IA	: Intelligence Artificielle
IoT	: Internet of Things
RF	: Random Forest
KNN	: K Nearst Neighbors
DT	: Decision Tree
ANN	: Artificial Neural Network
ML	: Machine Learning, apprentissage machine
RMS	: Root Mean Square
CF	: Crest Factor
CWT	: Continuous Wavelet Transform
ADASYN	: Adaptive Synthetic Sampling
CTGAN	: Conditional Tabular Generative Adversarial Network
SDV	: Synthetic Data Vault
TSTR	: Train Synthetic Test Real
PSD	: Power Spectral Density

RÉSUMÉ

Aujourd'hui, des technologies comme l'automatisation, l'Internet des objets et l'apprentissage machine transforment en profondeur les systèmes de production. Dans le contexte de l'Industrie 4.0, elles permettent aux entreprises de mieux exploiter leurs chaînes de fabrication, de réduire les arrêts non planifiés et de prolonger la durée de vie des équipements. Au cœur de cette évolution, la maintenance prédictive vise à anticiper les pannes avant qu'elles ne surviennent, en s'appuyant sur l'analyse de données capteurs. Parmi les signaux les plus utilisés, les vibrations des machines rotatives constituent une source d'information privilégiée pour détecter l'usure d'outils et les défauts mécaniques, à condition de disposer d'une chaîne de traitement et d'analyse adaptée. Dans ce contexte, ce mémoire propose une démarche de diagnostic et de détection précoce de défauts sur des équipements dynamiques (machines-outils de type CNC) à partir de mesures vibratoires issues de capteurs MEMS tri-axes. On met en place une chaîne complète allant de l'acquisition des signaux bruts jusqu'au diagnostic automatique de l'état "good/Bad" des outils de coupe. Les signaux sont d'abord segmentés en fenêtres temporelles, puis décrits par des indicateurs temporels et fréquentiels (RMS, kurtosis, facteurs de crête, spectre FFT, etc.). Sur ces données, on compare plusieurs familles d'algorithmes d'intelligence artificielle, en particulier des méthodes d'apprentissage machine (SVM, Random Forest, KNN, régression logistique, XGBoost) et des architectures d'apprentissage profond de type réseaux convolutifs et LSTM, afin d'identifier les modèles les plus performants pour une détection fiable en quasi-temps réel. Les résultats obtenus montrent qu'une combinaison judicieuse d'analyse vibratoire et de modèles d'IA permet d'améliorer significativement la sensibilité au défaut tout en restant compatible avec les contraintes opérationnelles des usines intelligentes

Mots-clés : maintenance prédictive, usine intelligente, usure d'outils, analyse vibratoire, apprentissage profond.

Keywords: predictive maintenance, smart factory, tool wear, vibration analysis, deep learning.

ABSTRACT

Today, technologies such as automation, the Internet of Things and machine learning are deeply transforming production systems. In the context of Industry 4.0, they help companies make better use of their manufacturing lines, reduce unplanned downtime and extend equipment lifetime. At the heart of this evolution, predictive maintenance aims to anticipate failures before they occur by exploiting sensor data. Among the available signals, vibrations of rotating machines are a key source of information for monitoring tool wear and detecting mechanical faults, provided that an appropriate processing chain is available. In this framework, this thesis proposes a diagnosis and early fault detection approach for dynamic equipment (CNC machine tools) based on vibration measurements acquired with tri-axial MEMS sensors. A complete pipeline is implemented, from raw signal acquisition to automatic classification of the tool condition (“good/bad”). The signals are segmented into time windows and characterized using indicators from time, frequency and time–frequency analysis (RMS, kurtosis, crest factor, FFT spectrum, etc.). On top of these features, several models of artificial intelligence are compared, machine-learning methods (SVM, Random Forest, KNN, logistic regression, XGBoost) and deep-learning architectures based on convolutional and LSTM networks, combined with imbalance-handling techniques such as ADASYN and CTGAN to cope with the strong class imbalance between healthy and faulty states. The results show that a suitable combination of vibration analysis and AI models significantly improves defect sensitivity while remaining compatible with the operational constraints of smart factories.

Keywords: predictive maintenance, smart factory, tool wear, vibration analysis, deep learning.

INTRODUCTION

0.1 Mise en contexte

Les différences notables entre la maintenance proactive classique et la maintenance prédictive sont les suivantes, la maintenance prédictive vise à prévoir les pannes avant qu'elles ne surviennent. Pour ce faire, elle exploite les nouvelles technologies pour collecter d'importants volumes de données en temps réel directement à partir des capteurs attachés aux machines. Ces données permettent d'arrêter instantanément la machine avant qu'une panne ne puisse se produire. En revanche, les approches classiques de maintenance s'appuyaient sur des modèles basés sur des distributions de données historiques, nécessitant des cycles d'analyse hors ligne, ce qui engendrait des délais significatifs entre la collecte des données et la prise de décision.

Cette approche moderne, qui intègre les données des capteurs directement dans le processus de décision, constitue un élément clé de l'Industrie 4.0. Cette révolution technologique transforme les entreprises, qu'elles soient manufacturières ou de services, en utilisant des avancées telles que l'Internet des Objets (IoT), le Big Data et l'Intelligence Artificielle (IA). Ces technologies permettent de connecter les équipements, d'analyser des volumes massifs de données et de prendre des décisions presque instantanément, réduisant ainsi les temps d'arrêt et rapprochant les entreprises de l'objectif de zéro panne. Une dynamique illustrée à la figure 1.



Figure 1
Industrie 4.0 : panorama des piliers technologiques
https://www.shutterstock.com/search/industrial-revolution-infographic

L'Industrie 4.0 réalise son potentiel grâce à l'IA, surtout le *machine Learning* et le *Deep Learning*, pour l'analyse des capteurs, la maintenance prédictive, le contrôle qualité et l'optimisation en temps réel. Côté science, la dynamique est massive, une analyse bibliométrique sur Scopus (une base de données de résumés et de citations scientifiques) recense 398 782 publications en machine Learning sur 2013–2022, organisées en 26 axes de recherche [1]. Côté déploiement, 78 % des entreprises déclarent utiliser l'IA dans au moins une fonction en 2025, en hausse nette par rapport à 2024. Dans les usages Deep Learning (ex. imagerie), quelques architectures dominent, les 9 principales concentrent 94,1 % des articles analysés (2012–2020), signe d'une standardisation rapide des pratiques [2].

Dans ce paysage marqué par la généralisation des données et de l'IA, les procédés d'usinage constituent un terrain d'application privilégié où la maintenance prédictive prend tout son sens. Les machines-outils sont en effet au cœur de la chaîne de valeur industrielle, et leur disponibilité comme la qualité des pièces produites dépendent directement de la capacité à anticiper les défaillances plutôt qu'à simplement les subir.

Au sein de ces procédés d'usinage, l'outil coupant paraît n'être qu'un consommable parmi d'autres, alors qu'en réalité il constitue un levier majeur de la performance globale, des études industrielles montrent que les outils représentent seulement 3 à 4 % du coût direct d'usinage, mais conditionnent la quasi-totalité des autres postes (temps machine, énergie, rebuts, retouches, etc.). L'usure de l'outil se trouve ainsi au croisement des enjeux de productivité, de qualité et de fiabilité, ce qui en fait une cible prioritaire pour les approches de maintenance prédictive. Dans certains travaux de recherche, on estime même que les outils peuvent aller jusqu'à 25 % du coût total d'un procédé et être responsables de près de 20 % des temps d'arrêt machine, ce qui veut dire qu'une mauvaise gestion de l'usure se traduit directement en heures de production perdues [3]. Sur le terrain, une statistique montre qu'un atelier tournant à environ 60 000 changements d'outil par an, avec un temps moyen de 18 s, perd autour de 300 h de machine par an rien que pour ces micro-arrêts liés au changement d'outils. Les chercheurs montrent d'ailleurs que des systèmes de suivi d'usure et de changement d'outil optimisé peuvent réduire les coûts liés aux outils de l'ordre de 10

à 40 %, en allongeant la durée de vie utile sans sacrifier la qualité et en évitant des défaillances catastrophiques [3].

En résumé, l'essor de l'Industrie 4.0, de l'IA et de la maintenance prédictive ne vise pas seulement à "faire moderne", mais à répondre très concrètement à un enjeu clé, réduire au minimum les arrêts de production et les rebuts liés à l'usure des outils coupants. En connectant les machines, en exploitant en continu les données de capteurs et en modélisant finement l'état des outils, il devient possible d'anticiper leur défaillance, de planifier leur remplacement au bon moment et d'éviter à la fois les pannes imprévues et le sur-usage des arêtes de coupe. L'objectif est clair transformer la gestion de l'outil, longtemps réactive et empirique, en un processus piloté par les données, qui soutient la continuité de production, la stabilité de la qualité et la réduction des temps d'arrêt non planifiés.

0.2 Problématique

La mise en œuvre de la maintenance prédictive dans le cadre de l'Industrie 4.0 soulève une problématique complexe, bien que cette approche promette de révolutionner la gestion des équipements en réduisant les pannes et en optimisant les performances, elle reste difficilement accessible à de nombreuses entreprises. Le recours à des technologies avancées telles que les capteurs connectés, le Big Data, l'intelligence artificielle et le concept émergent des jumeaux numériques requiert non seulement des investissements financiers considérables, mais aussi une reconfiguration des infrastructures existantes et une expertise technique approfondie. Ces barrières limitent l'adoption de ces technologies, en particulier pour les petites et moyennes entreprises, et mettent en lumière la nécessité d'une recherche scientifique ciblée pour développer des solutions plus accessibles et mieux adaptées aux besoins et capacités de différentes organisations.

Dans ce cadre, l'usure des outils de coupe en usinage CNC est un excellent exemple de ce phénomène, un outil émoussé ou endommagé affecte la précision dimensionnelle, la qualité de surface et peut, en cas de rupture, abîmer la pièce ou la machine et mettre en jeu la sécurité. D'où la nécessité d'une surveillance en temps

réel pour décider du bon moment de remplacement et éviter les arrêts non planifiés, à l'inverse d'un suivi hors ligne (inspection après usinage), la surveillance en ligne repose sur des capteurs et des algorithmes embarqués qui infèrent l'état d'usure pendant l'usinage, sans interrompre le processus.

Un autre défi majeur réside dans la nécessité de mettre à jour en permanence les modèles de détection des défauts pour prédire avec précision le comportement futur des équipements physiques. Les conditions opérationnelles peuvent évoluer rapidement, et les modèles doivent être capables de s'adapter en temps réel pour rester pertinents et fiables. Cela soulève des questions complexes liées à la collecte, à l'analyse et à l'intégration des données en temps réel pour alimenter ces mises à jour de modèle [4].

0.3 Objectifs

0.3.1 Objectif général

Proposer une méthode innovante de maintenance prédictive en temps réel des appareillages dynamiques dans les usines intelligentes, basée sur l'utilisation de l'intelligence artificielle (Artificiel Intelligence), tels que l'apprentissage automatique (Machine Learning) et l'apprentissage profond (Deep Learning), afin d'améliorer la disponibilité des équipements, réduire les coûts de maintenance et minimiser les interruptions imprévues dues aux pannes. Cette méthode exploitera les algorithmes des prédictions pour créer des modèles virtuels précis et dynamiques des équipements, permettant une surveillance en temps réel de leur état de santé et une analyse approfondie post-pronostic. Cette approche facilitera la planification proactive et ciblée des actions de maintenance, garantissant ainsi un fonctionnement optimal des appareillages tout en répondant aux exigences d'efficacité et de durabilité de l'Industrie 4.0.

0.3.2 Objectifs spécifiques

- Prédiction des défauts des machines CNC précisément l'usure des outils en se basant sur différentes bases de données.

- Sélection des méthodes d'apprentissage automatique (Machine Learning) et d'apprentissage profond (Deep Learning) pour la prédiction des défauts.
- Implémentation des algorithmes en Python pour détecter et anticiper les défaillances.
- Évaluation comparative des modèles d'apprentissage afin de sélectionner celui offrant les meilleures performances.
- Déploiement applicatif réalisation d'une application de démonstration (desktop), pouvant éventuellement s'intégrer à une ligne de production plus tard.

0.4 Hypothèses

- Étant donné que le fonctionnement des machines CNC est étroitement lié aux vibrations et aux variations des signaux captés par les capteurs, on part du principe que des anomalies dans ces vibrations indiquent des défaillances au niveau de l'outil des machines CNC.
- Les écarts de vibration par rapport au profil nominal (ex dérives de RMS, kurtosis, énergie fréquentielle par bandes) sont des indicateurs précoces de défaillances sur machines CNC
- Pour des raisons de confidentialité l'ordre des opérations des outils a été délibérément modifié et que seule une partie du flux de production est présente dans le jeu de données, on suppose que chaque opération répertoriée correspond à un processus spécifique effectué par un outil distinct avec des paramètres uniques.
- Faute d'informations du contrôleur, une segmentation temporelle par fenêtres de taille fixe permet une détection d'anomalies par processus suffisante, sans recourir à l'identifiant du processus.
- L'augmentation des données de la classe BAD via ADASYN ou CTGAN expose le modèle à plus de variantes difficiles et améliore la détection des défaillances rares. Après un simple réglage du seuil de décision, on s'attend à capter plus tôt ces cas sans multiplier les fausses alertes.

0.5 Originalité de la recherche

L'originalité de ce travail tient d'abord à l'architecture multi-niveau adoptée pour analyser les vibrations dans un contexte industriel réel. On ne se limite pas à une machine isolée ni à un seul scénario d'usinage, on compare systématiquement un même outil de coupe utilisé sur plusieurs centres d'usinage, ce qui permet de mettre en évidence l'influence de la machine, de sa chaîne cinématique et de son état propre sur la signature vibratoire. Cette comparaison multi-machines, conduite à conditions de coupe comparables, offre une vision beaucoup plus riche des comportements dynamiques que les approches classiques limitées à un seul banc d'essai ou à une cellule de laboratoire.

Sur le plan du traitement du signal, l'originalité vient de l'approche hybride qui combine trois modes d'analyse complémentaires, temporel, fréquentiel et temps–fréquence. Les indicateurs temporels suivent les tendances globales et les changements de régime, l'analyse fréquentielle révèle les bandes de résonance et les harmoniques, tandis que l'analyse temps–fréquence suit l'évolution de ces phénomènes au cours du temps. En combinant ces trois regards plutôt que de se limiter à un seul domaine, on obtient une description plus complète des vibrations, mieux adaptée à la détection de comportements anormaux et de l'usure progressive.

Un autre aspect important réside dans l'usage de données industrielles réelles issues de capteurs MEMS montés sur des machines de production, sur une période prolongée et sur plusieurs équipements. Il existe très peu de jeux de données CNC issus d'environnements de production réels, alors que ces conditions sont essentielles pour construire des modèles robustes et généralisables. En travaillant directement sur les mesures brutes en atelier, on se rapproche des contraintes et des variabilités du terrain industriel.

Enfin, l'approche d'apprentissage combine des méthodes de machine Learning et des réseaux de neurones profonds pour automatiser la détection d'anomalies et la prédiction des défaillances futures. Les algorithmes supervisés exploitent les descripteurs extraits dans les trois domaines, tandis que les modèles profonds

apprennent directement des fenêtres vibratoires la frontière entre états “Good” et “Bad”. Pour traiter le fort déséquilibre entre classes, on intègre ADASYN et CTGAN, adapté aux données tabulaires, afin de générer des exemples artificiels réalistes pour la classe rare.

0.6 Méthodologie

Afin d’atteindre les objectifs de cette étude, on utilise des capteurs MEMS et des outils d'analyse des vibrations afin de surveiller les machines industrielles. L'analyse des vibrations permet de détecter les anomalies et d'optimiser la maintenance prédictive.

On a d’abord mené une revue de littérature (articles, conférences, ouvrages, sources spécialisées) pour cerner les méthodes pertinentes, leurs atouts et limites, et fixer une démarche claire.

Il s’agit d’un projet de recherche appliquée fondé sur des données de vibrations réelles, les mesures proviennent de trois fraiseuses CNC instrumentées par des capteurs MEMS. La méthodologie retenue s’articule en trois étapes, brièvement présentées dans cette section et dont les détails seront traités dans le Chapitre 2.

La première étape consiste à faire une analyse temporelle-fréquentielle unifiée des signaux, avec extraction de descripteurs temporels (RMS, kurtosis, facteur de crête, écart-type, etc.) et fréquentiels, afin de constituer un jeu de variables robuste.

La deuxième étape comprend l’utilisation de modèles de machine et Deep Learning en suivant deux voies complémentaires, comparer un même outil sur plusieurs machines (multi-machines) et entraîner des modèles globaux (régression logistique, K-NN, SVM, arbres de décision, Random Forest, XGBoost, réseaux de neurones), pour la détection et la prédiction des anomalies. Pour cela, on a utilisé la programmation en Python

Finalement, la troisième étape consiste à utiliser, comparer et évaluer ces modèles sur des jeux séparés, puis la préparation au déploiement industriel pour reconnaître de manière fiable les signaux « bons » et « mauvais » en fonctionnement réel.

0.7 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, détections des défauts et l'usure d'outils 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 les méthodes d'analyse vibratoire couramment utilisées dans le domaine du diagnostic et de la détection des défauts qui sont l'analyse temporelle, l'analyse fréquentielle et l'analyse temps-fréquence. Cette étude est nécessaire pour comprendre l'étape d'extraction des caractéristiques et savoir différencier les bons signaux des mauvais.

Le quatrième chapitre expose l'étape de développement des modèles de prédiction dans le but de détecter les défauts liés à l'usure de l'outils. On aboutit dans ce chapitre l'approche un seul outil multi machines. Dans cette section, une comparaison entre les algorithmes de classification a été proposée (SVM, Logistic Regression, KNN, DT, RF, XGBoost)

Le cinquième chapitre présente le développement de modèles d'apprentissage profond pour détecter l'usure des outils d'où l'exploitation de toutes les données disponibles afin d'entraîner et comparer des architectures adaptées aux signaux vibratoires pour capter à la fois les motifs locaux et les dépendances temporelles.

Enfin, une discussion générale de la problématique ainsi que les contributions ce mémoire, les limites et les suggestions des travaux futurs sont présentées dans la conclusion.

1. CONCEPTS DE BASE ET REVUE DE LA LITTÉRATTURE

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

1.2.1 L'évolution industrielle et l'industrie 4.0

L'idée d'Industrie 4.0 émerge en 2011 pour répondre aux besoins d'efficacité, d'agilité et de personnalisation croissante dans le manufacturier. Portée par la numérisation et des technologies comme l'IoT, les données massives et l'IA, elle prolonge trois vagues majeures : la mécanisation (vapeur), l'électrification et la production de masse, l'automatisation avec l'informatique et l'électronique [5]. La figure 2 ci-dessous synthétise cette trajectoire et situe l'apparition d'Industrie 4.0 [6].

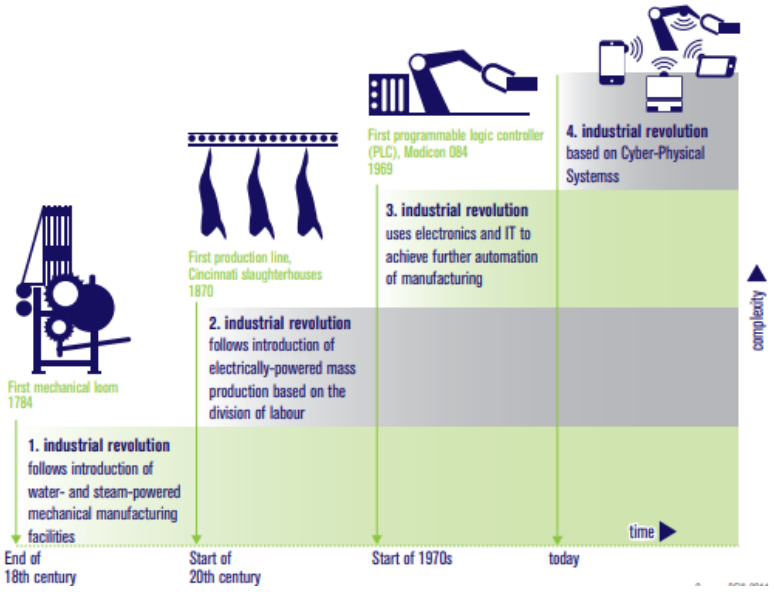


Figure 2
Les quatre révolutions industrielles

1.2.2 La maintenance industrielle : définition et typologies

La maintenance industrielle regroupe l'ensemble des opérations destinées à maintenir ou à rétablir un bien dans l'état requis afin d'assurer le service attendu, tout en visant un coût optimal. Dans tous les secteurs, négliger le coût et l'impact des pannes soudaines est risqué, une grande part de la productivité dépend directement de la qualité du dispositif de maintenance. On distingue trois types principaux :

- Corrective : déclenchée après la panne pour remettre l'équipement en service
- Préventive : planifiée et régulière, destinée à détecter et corriger les problèmes avant qu'ils n'apparaissent
- Prédictive : fondée sur la surveillance de l'état réel des machines et l'analyse de la tendance de dégradation afin d'anticiper le point de défaillance et d'intervenir au bon moment.

En bref, la corrective répare, la préventive évite, la prédictive anticipe.

1.2.3 Analyse vibratoire au service de la maintenance prédictive

On part du principe que l'érosion de l'arête de coupe modifie les forces de coupe, et donc la signature vibratoire lors du passage des dents. Des accéléromètres MEMS placés près de la broche révèlent ces évolutions sous forme d'augmentations d'amplitude, de modulations, et d'un transfert d'énergie vers des bandes de fréquence plus élevées [7]. On exploite ces changements par une analyse temporelle afin de capter l'apparition d'impulsions et de dérives dans le signal. En parallèle, une analyse fréquentielle du signal vibratoire s'articule autour du spectre d'amplitude unilatéral on superpose ensuite les spectres obtenus pour un « outil sain » et un « outil usé » afin d'en visualiser clairement les différences caractéristiques. L'ensemble de ces indicateurs, temporels et fréquentiels, alimente ensuite nos modèles d'apprentissage profond, ce qui permet de distinguer automatiquement un outil sain d'un outil usé et de suivre sa dégradation dans le temps.

1.2.4 Processus de mise en œuvre d'une maintenance prédictive basée sur les données

Pour mettre en place une stratégie de maintenance prédictive, il faut d'abord des capteurs connectés qui mesurent en continu les paramètres de fonctionnement des équipements. Les données sont agrégées et transmises via l'IoT vers un moteur d'analyse [8]. Ce moteur croise les mégadonnées avec l'historique d'interventions, apprend les corrélations entre signatures de capteurs et pannes, puis anticipe les défaillances, dès que des motifs déjà associés à un défaut réapparaissent, il propose une action de maintenance pour éviter l'arrêt non planifié.

Pour la sélection des technologies à chaque étape, Radhya et al [9] proposent des lignes directrices utiles aux décideurs. Une fois ce choix acté, on passe au développement du modèle de maintenance prédictive autour de quatre temps forts :

- Collecter et préparer les données des équipements puis distinguer états « normaux » et « non normaux ».
- Modéliser les schémas de panne, à la fois par détection d'anomalies et par classification multi-catégories fondée sur l'historique.
- Entraîner le modèle à reconnaître de nouveaux événements/défaillances.
- Adapter le système pour qu'il se mette à jour au fil des nouvelles observations.

On entraîne le modèle sur des exemples sans défaut et avec défaut, afin qu'il identifie l'état courant de l'équipement. Classiquement, l'ensemble est scindé en données d'entraînement et données de test, on apprend sur les premières et on vérifie la performance sur les secondes. Le flux global de l'acquisition à l'évaluation est résumé dans la figure 3 adapté de [10].

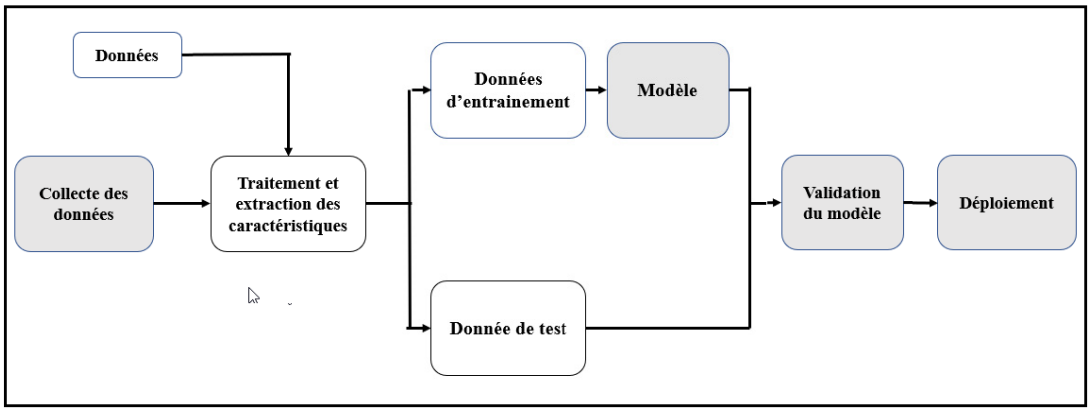


Figure 3
Chaîne de traitement d'une maintenance prédictive basée sur les données

1.3 Revue de la littérature

1.3.1 Maintenance prédictive

La maintenance prédictive s'appuie sur la collecte continue de données capteurs (vibrations, température, courant, etc.) et sur des analyses avancées (IA) pour anticiper les défaillances et planifier au plus juste les interventions, elle est aujourd'hui largement documentée pour ses gains potentiels en disponibilité, qualité et fiabilité [11].

Dans le cadre de l'industrie 4.0, elle nécessite une intégration cohérente de de l'acquisition, de l'infrastructure, du stockage, de la sécurité et de l'intelligence analytique, tout en mobilisant diverses techniques éprouvées, analyse d'huile, acoustique, suivi thermique/température, analyse du courant moteur, et surtout analyse vibratoire. Cette dernière est couramment employée pour déceler rapidement l'usure des outils de coupe à partir de signatures temporelles et fréquentielles (RMS, kurtosis, facteurs spectraux) qui sont mesurées par des accéléromètres lors des opérations d'usinage [12]. En réalité, la surveillance en ligne, contrairement aux vérifications hors ligne, combine capteurs et algorithmes pour évaluer en permanence l'état et s'intègre dans les modèles modernes diagnostic–pronostic–décision, qui organisent les méthodes de maintenance en catégories complémentaires.

1.3.2 Analyse des données vibratoires

Analyse temporelle

Le suivi indirect de l'usure outil repose sur l'utilisation de capteurs mesurant des grandeurs physiques corrélées à l'état de l'arête de coupe. À mesure qu'un outil s'use, il a tendance à engendrer des changements mesurables dans le processus de coupe, par exemple une augmentation des efforts, des vibrations ou de la température de coupe [13].

L'analyse vibratoire est un outil éprouvé en maintenance prédictive pour diagnostiquer de nombreux problèmes machine (déséquilibres, roulements défectueux, jeux excessifs, résonances, etc.)

Dans un premier temps, on peut analyser les signaux temporels mesurés pendant l'usinage pour en déduire des descripteurs simples. Parmi les indicateurs couramment utilisés figurent la valeur moyenne, la valeur efficace (RMS), l'amplitude crête (peak), le facteur de crête (ratio peak/RMS), l'écart-type, la kurtosis, etc. Par exemple, l'augmentation progressive de la composante moyenne de force ou de courant au fil du temps d'usinage indique généralement l'usure progressive de l'outil (outil qui nécessite plus d'effort moyen pour couper) [14]. De façon similaire, l'instabilité croissante du signal (fluctuations plus importantes, donc écart-type ou RMS en hausse) peut indiquer une détérioration du tranchant. Ces indicateurs scalaires peuvent être comparés à des seuils fixes ou flottants. Dans l'industrie, une stratégie simple est de définir un seuil d'alarme sur un indicateur par exemple, une limite sur la hausse du courant moteur appelée "limite émoussée" au-delà de laquelle on considère que l'outil est trop usé. Cette méthode est facile à implémenter et fonctionne dans bien des cas (typiquement, la majorité des signaux augmentent avec l'usure). Cependant, elle présente des limites, certains signaux ne varient pas de manière monotone avec l'usure (ils peuvent diminuer puis remonter, selon le mécanisme d'usure en cours). Un même niveau d'usure peut produire des signatures vibratoires très différentes d'un cas à l'autre, ce qui complique l'établissement de seuils ou de critères universels, ce qui rend un simple seuil peu fiable [14].

L'analyse du domaine temporel consiste à examiner les signaux de vibration dans l'ordre chronologique afin de détecter des anomalies et de diagnostiquer l'état de santé d'une machine. Elle permet de quantifier l'amplitude et la variabilité des signaux au fil du temps, et donc de repérer les événements inhabituels ou caractéristiques de défauts mécaniques. Les caractéristiques extraites dans ce domaine sont utilisées comme descripteurs statistiques pour différencier les états « sains » et « défectueux » des équipements. Ci-dessous, les principales métriques utilisées sont détaillées.

- La valeur quadratique moyenne (RMS) calculé avec la formule (1), indique l'énergie globale d'un signal. Elle est couramment utilisée pour détecter des déséquilibres dans les machines tournantes.

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

- Le facteur de crête (Crest Factor, CF) calculé avec la formule (2), mesure le rapport entre la valeur maximale et la valeur RMS. Il est utilisé pour identifier les impulsions anormales dans les signaux, fréquentes en cas de défauts localisés comme la dégradation des roulements.

$$CF = \frac{max(x_i(i))}{V_{RMS(x_i(i))}} \tag{2}$$

- Le kurtosis calculé avec la formule (3), mesure l'aplatissement de la distribution du signal. C'est un indicateur pertinent pour détecter des pics impulsifs dans les données de vibration, typiques des pannes localisées.

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

- L'écart type (STD) calculé avec la formule (4) représente la dispersion des valeurs autour de la moyenne. Il est utile pour estimer la stabilité du comportement vibratoire d'un composant.

$$STD = \sqrt{\frac{1}{L}\sum_{i=1}^{L}\left|y_i - \mu\right|^2} \quad (4)$$

Néanmoins, l'analyse temporelle de base reste un point de départ indispensable, souvent couplé avec des techniques fréquentielles pour améliorer le diagnostic.

Analyse fréquentielle

L'analyse fréquentielle consiste à étudier le spectre du signal de coupe, obtenu typiquement via une Transformée de Fourier (FFT), afin d'identifier des fréquences caractéristiques qui évoluent avec l'usure. Par exemple, l'apparition ou l'amplification de vibrations à des fréquences particulières peut trahir un phénomène de broutement induit par un outil usé. De même, un outil endommagé (petit éclat) peut engendrer une rotation excentrée ou des impacts périodiques, se manifestant par des composantes fréquentielles à la fréquence de rotation de l'outil ou ses harmoniques. L'analyse spectrale est largement utilisée, d'après une revue, la FFT fait partie des techniques de signal les plus populaires en usinage, aux côtés de la transformée en ondelettes et de la transformée de Hilbert-Huang [15].

La FFT permet de passer du domaine temporel, souvent complexe, à une vue où certaines composantes se détachent. Toutefois, elle suppose un signal stationnaire ou quasi-stationnaire sur la fenêtre analysée, ce qui n'est pas toujours le cas en usinage (les signaux peuvent évoluer en quelques millisecondes lors d'une casse par exemple). L'analyse fréquentielle aide aussi à filtrer le signal autour des bandes intéressantes (par ex. filtrer autour d'une fréquence de résonance d'outil qui change d'amplitude avec l'usure, ou au contraire éliminer la composante à la fréquence de rotation si elle n'apporte pas d'info pertinente).

Dans la surveillance de l'état des outils et la détection de défauts mécaniques en usinage la transformée de Fourier rapide (FFT) est un outil classique pour analyser les signaux vibratoires ou acoustiques. Elle permet de faire ressortir les fréquences dominantes et les composantes harmoniques dans le spectre, fournissant des

indications sur les mécanismes sous-jacents d'usure ou de défaillance [16]. Par exemple, en fraisage, l'intensification de certaines bandes de fréquences spécifiques dans le spectre vibratoire a été corrélée à la progression de l'usure de l'outil [16]. Toutefois, la FFT classique, en fournissant une analyse purement globale en fréquence, montre ses limites face aux signaux transitoires ou non stationnaires typiques des vibrations d'usinage [17]. Incapable de localiser précisément l'apparition d'événements brefs dans le temps, cette transformée peut manquer certaines signatures éphémères d'usure ou de défaut. Ces limites justifient donc le recours à des méthodes temps-fréquence plus avancées (par exemple la transformée en ondelettes) pour une détection plus fine des phénomènes transitoires.

Analyse temps-fréquence

L'analyse par ondelettes s'est imposée pour les signaux d'usinage, intrinsèquement non stationnaires, elle offre une lecture temps–fréquence multi-échelle qui localise transitoires, modulations et résonances bien mieux qu'une FFT seule, d'où son adoption massive en diagnostic d'état machine et en suivi d'usure des outils. Des revues de référence soulignent ce rôle central en maintenance (présentation des CWT/DWT/WPT, débruitage, extraction de signatures, singularités), et documentent ses gains sur vibrations et AE en contexte industriel [18].

Étant donné le caractère hautement non stationnaire des signaux d'usinage, les ondelettes constituent un outil d'analyse particulièrement précieux. La transformée en ondelettes réalise une décomposition multi-échelle du signal et fournit une localisation temporelle et fréquentielle fine des phénomènes [19]. Fang et al [20] ont montré qu'une analyse par ondelettes des vibrations en tournage permet de mieux débruiter et mettre en évidence les effets de l'usure de l'arête de coupe qu'une FFT classique, en particulier lors d'usinages à grande vitesse. De son côté, Cao et al [21] ont appliqué la WPT (décomposition par paquets d'ondelettes) suivie de la transformée de Hilbert–Huang pour détecter le phénomène de broutement en fraisage, démontrant ainsi l'efficacité de l'analyse par ondelettes sur ces signaux vibratoires non stationnaires. Par ailleurs, l'extraction de paramètres fréquentiels par ondelettes sur chaque sous-

bande, combinée à un algorithme de classification, a permis d'identifier l'état d'usure de l'outil avec une précision élevée (jusqu'à ~96 % de taux de reconnaissance dans une étude) [19]. En fraisage, des travaux exploitent la CWT pour suivre l'évolution d'énergie par bandes et refléter la dégradation de l'outil, y compris avec des schémas de séparation de sources qui améliorent la lisibilité des indicateurs d'usure [22].

Enfin, des approches hybrides récentes intègrent les ondelettes dans des schémas d'apprentissage profond [19]. Plus récemment, des approches hybrides marient ondelettes et Deep Learning, par exemple, des Derived Wavelet Frames alimentent un CNN 2D et offrent une surveillance robuste de l'usure en fraisage, avec des performances supérieures aux pipelines classiques [19].

Historiquement, la littérature sur l'usinage reliait clairement le contenu vibratoire aux états de coupe et montrait que les ondelettes facilitent le diagnostic en cours de procédé [23].

1.3.3 L'intelligence artificielle : Apprentissage Automatique (Machine Learning)

Avec l'essor du Big Data et l'augmentation exponentielle de la puissance de calcul, l'apprentissage automatique (Machine Learning) est devenu un outil indispensable pour réaliser des prédictions et optimiser efficacement des systèmes industriels complexes [24]. Contrairement à la programmation classique à base de règles fixes, le Machine Learning apprend en identifiant automatiquement des motifs et similarités au sein de grandes quantités de données, ce qui lui permet de généraliser ces connaissances à de nouvelles informations sans instructions explicites [25]. Aujourd'hui, ces techniques s'illustrent dans de nombreux domaines tels que le traitement du langage naturel, la vision par ordinateur ou les systèmes de recommandation. Parmi les approches de Machine Learning, on distingue généralement quatre familles principales, l'apprentissage supervisé, non supervisé, semi-supervisé et par renforcement [24, 25]. Chacune de ces familles est adaptée à des types de problèmes et de données spécifiques. L'apprentissage supervisé, par exemple, exploite des jeux de données étiquetés pour entraîner un modèle capable de prédire avec précision des résultats futurs, tandis que l'apprentissage non

supervisé vise à découvrir des structures cachées dans des données non étiquetées, sans connaissances préalables des résultats attendus [24].

Support Vector Machines : Les machines à vecteurs de support (Support Vector Machines, SVM) sont des classificateurs supervisés issus de la théorie de l'apprentissage statistique développée par Vapnik et ses collègues au début des années 1990. Leur principe est de trouver un hyperplan qui sépare au mieux deux classes en maximisant la marge entre les exemples des deux côtés, ce qui favorise une bonne généralisation. Cortes et Vapnik [26] ont formalisé cette approche pour les problèmes de classification binaire.

K-Plus Proches Voisins (k-NN) : Il s'agit d'un algorithme non paramétrique simple où la prédiction pour une nouvelle instance se fait en regardant les K échantillons d'entraînement les plus proches (selon une métrique de distance, typiquement la distance euclidienne) et en effectuant une majorité de vote (pour la classification) ou une moyenne (pour la régression). Malgré sa simplicité, k-NN peut offrir de bonnes performances si les données sont bien structurées. En revanche, il est peu efficace sur des données de grande dimension ou volumineuses. En pratique industrielle, k-NN a pu servir à détecter des anomalies en considérant qu'un point de données très éloigné de ses voisins est potentiellement anormal [27].

Arbres de décision : Un arbre de décision est un modèle supervisé qui prédit une variable cible en segmentant récursivement l'espace des données, à chaque nœud, un test sur un attribut sépare les observations en sous-ensembles de plus en plus homogènes, jusqu'aux feuilles qui portent une classe ou une valeur numérique. Facile à interpréter grâce à sa représentation en règles « si… alors… », il permet d'identifier clairement les facteurs qui conduisent, par exemple, à une défaillance. En revanche, un arbre unique reste peu robuste de petites variations dans les données d'apprentissage peuvent modifier sa structure et dégrader la stabilité des prédictions. [28].

Les forêts aléatoires : Les forêts aléatoires (Random Forests) combinent un grand nombre d'arbres de décision construits sur des sous-échantillons aléatoires des

données et des variables. Cette approche d'ensemble améliore la précision et réduit le risque de surapprentissage en moyennant les prédictions des arbres. Dans le contexte de la maintenance prédictive, les forêts aléatoires sont utilisées pour identifier les facteurs les plus influents sur l'occurrence de pannes et pour prédire le risque de défaillance d'un équipement à partir de multiples capteurs [29].

XGBoost : L'algorithme XGBoost (« eXtreme Gradient Boosting ») est une méthode avancée d'ensemble qui combine plusieurs arbres décisionnels faibles de manière séquentielle afin d'améliorer progressivement la prédiction finale [30]. Chaque nouvel arbre corrige les erreurs résiduelles des arbres précédents, ce qui permet à XGBoost d'obtenir des performances prédictives exceptionnelles. Dans la maintenance prédictive industrielle, XGBoost est particulièrement apprécié pour sa capacité à traiter de grands volumes de données complexes et à identifier efficacement les facteurs critiques de défaillance. Par exemple, il est largement utilisé pour prédire précisément la durée de vie restante des équipements industriels (RUL), ainsi que pour classifier les types de pannes à partir des données issues de multiples capteurs.

Régression Logistique : La régression logistique est un modèle statistique souvent utilisé pour les problèmes de classification binaire, bien qu'il puisse être étendu à des classifications multi classe. Il modélise la probabilité qu'une instance appartienne à une classe particulière (par exemple, panne ou absence de panne) à partir des valeurs des variables prédictives [31]. Ce modèle produit une valeur entre 0 et 1 représentant la probabilité d'appartenance à la classe positive (par exemple, « panne imminente »). En maintenance prédictive, la régression logistique est fréquemment employée pour anticiper les défaillances d'équipements industriels en estimant le risque de panne à court terme à partir d'indicateurs clés comme les niveaux de vibration, les variations de température, ou encore l'évolution de la consommation énergétique [8].

En milieu industriel, le Machine Learning trouve une application majeure dans la maintenance prédictive, qui vise à anticiper les défaillances et à optimiser la gestion du cycle de vie des équipements. Dans les systèmes d'usinage CNC, par exemple, la surveillance de l'usure des outils de coupe est un cas emblématique, l'intégration de

capteurs (vibrations, acoustiques, forces de coupe, température, images) sur les machines permet de collecter en continu un volume massif de données sur l'état de l'outil pendant la production. L'analyse de ces données par ces algorithmes d'apprentissage automatique révèle des motifs caractéristiques de l'usure et fournit une estimation en temps réel du niveau de dégradation de l'outil.

Les efforts de recherche récents se concentrent sur l'application des techniques d'intelligence artificielle. Dans ce contexte Hoppenstedt et al [32] 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 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 [8].

La prédiction de l'usure des outils de coupe constitue un enjeu majeur en maintenance prédictive, et de nombreuses études explorent l'application de l'apprentissage automatique dans ce domaine. Par exemple, Wu et al [33] se sont intéressés à l'estimation de l'usure d'outils de fraisage à l'aide de différents algorithmes d'apprentissage supervisé. Ces auteurs [33] ont comparé les performances de trois méthodes couramment utilisées – les réseaux de neurones artificiels (ANN), les machines à vecteurs de support (SVM) et les forêts d'arbres décisionnels aléatoires (RF) – pour prédire l'état d'usure des outils à partir de données expérimentales issues de 315 essais de fraisage. Pour chaque essai, un ensemble de caractéristiques statistiques a été extrait de divers signaux capteurs liés au processus d'usinage, notamment les signaux de force de coupe, de vibration et d'émission acoustique (AE).

Les résultats obtenus par Wu et al [33] mettent en évidence une nette supériorité du modèle fondé sur les forêts aléatoires pour la prédiction de l'usure. En effet, le classifieur RF s'est avéré capable de fournir les estimations d'usure les plus précises, surpassant celles obtenues par les modèles fondés sur les réseaux de neurones ou sur les SVM. Autrement dit, dans le contexte spécifique du suivi d'usure des outils de

coupe, les forêts d'arbres décisionnels ont offert une meilleure exactitude prédictive que les approches basées sur les ANN ou les SVM. Ces conclusions suggèrent que les méthodes d'ensemble comme les forêts aléatoires peuvent capter plus efficacement les relations complexes entre les caractéristiques signal et l'usure de l'outil, par rapport aux algorithmes plus traditionnels, du moins pour les données et conditions de fraisage étudiées par Wu et al [33].

Plus récemment, Jones et Cao [13] ont proposé une architecture de fusion de données multi-capteurs combinée à des modèles d'apprentissage automatique (forêt aléatoire, réseau de neurones, XGBoost), atteignant un coefficient de détermination R^2=0.955 pour la prédiction de l'usure de flanc en fraisage CNC. Pour renforcer le vecteur de caractéristiques Tahir et al [34], on ajouter des indicateurs temporels simples et efficaces, afin de stabiliser les estimations et d'alimenter les classifieurs avec des signaux résumés mais complémentaires des marqueurs fréquentiels. Cette combinaison fournit un socle robuste pour distinguer **Good (Sain) /Bad (Défectueux)** et suivre l'évolution de l'usure.

Ces travaux illustrent donc le potentiel des algorithmes d'apprentissage automatique avancés pour améliorer le suivi de l'état des outils, en offrant des prédictions d'usure plus fiables et en permettant d'anticiper le remplacement ou la maintenance des outils de coupe avec une plus grande précision.

Dans la littérature TCM, les approches supervisées "classiques" (SVM, RF, k-NN, régression logistique, XGBoost) sont la référence de départ avant d'évaluer des modèles profonds (CNN, LSTM ou hybrides), ce qui permet de mesurer l'apport réel du Deep Learning [35]. En pratique, ces modèles de référence restent très efficaces et lisibles, mais lorsque le volume de données explose et que la non-stationnarité des signaux (vibrations, AE) devient dominante, on passe aux réseaux profonds (p. ex. CNN-LSTM) qui exploitent des représentations temps-fréquence pour capter les transitoires et les variations de conditions en usinage, notamment en fraisage CNC [36].

1.3.4 L’intelligence artificielle : Apprentissage Profond (Deep Learning)

L’apprentissage profond (Deep Learning), une branche avancée de l'apprentissage automatique fondée sur les réseaux neuronaux profonds composés de nombreuses couches, permet aux machines de réaliser des tâches particulièrement complexes avec une efficacité remarquable. Les travaux récents ont montré que des approches de Deep Learning pouvaient détecter automatiquement l’usure en combinant l’analyse d’images de l’outil et de signaux vibratoires, ainsi, l’utilisation conjointe de la vision par ordinateur et de capteurs vibratoires, couplée à des réseaux de neurones convolutifs, permet d’identifier avec précision l’état d’usure d’un outil pendant l’usinage [37].

Depuis 2018, l’essor de l’apprentissage profond permet de traiter directement les signaux bruts ou de les transformer automatiquement en représentations optimales. En particulier, des réseaux de neurones convolutifs (CNN) sont exploités pour extraire automatiquement les motifs caractéristiques de l’usure dans les données vibratoires. Une approche courante consiste à convertir le signal 1D en une image temporelle-fréquentielle (par exemple via une transformée de Fourier à court terme, ou une transformée ondelette continue) et à appliquer un CNN sur cette image comme s’il s’agissait d’une empreinte visuelle de l’usure [38]. De même, Zhang et al [38] ont montré sur des signaux de force de coupe que cette technique améliore la reconnaissance d’état d’usure par rapport aux réseaux classiques, tandis que Li et al [39] ont appliqué un CNN à des spectrogrammes issus de vibrations mesurées, ce qui a permis de prédire la durée de vie résiduelle d’un outil de façon précise. L’intérêt du Deep Learning est de pouvoir apprendre des caractéristiques complexes sans avoir à les définir manuellement, et de modéliser des relations non linéaires entre le signal et l’usure, ce qui correspond bien à la nature du procédé de fraisage [38].

D’autres réseaux profonds comme les LSTM (Long Short-Term Memory) ont été employés pour exploiter la dimension temporelle des données vibratoires et suivre l’évolution progressive de l’usure. Une étude a combiné un réseau ResNet (résiduel) et un LSTM en entrée de gamme pour fusionner des signaux multi-capteurs (vibrations, force de coupe, courant) et prédire l’usure, ce modèle hybride ResNet-LSTM a surpassé les modèles individuels, atteignant une précision de 98,3 % dans la

prédiction du niveau d'usure [40]. Ces résultats très élevés montrent le potentiel du Deep Learning, mais ils exigent en contrepartie de larges volumes de données pour l'entraînement. Avec la multiplication des capteurs sur les machines modernes, il devient possible de collecter de grands jeux de données en production pour alimenter ces modèles.

Parallèlement, les réseaux de neurones artificiels (ANN) sont parmi les plus populaires pour ces tâches, par leur capacité à approximer des relations non linéaires complexes. Une revue de 77 articles récents indique que les ANN figurent parmi les méthodes prédominantes en détection d'usure, notamment pour les applications en tournage [35]. Les réseaux profonds (Deep Learning) ont fait leur entrée avec succès par exemple, des réseaux convolutifs (CNN) couplés à des représentations temps-fréquence des signaux permettent d'atteindre des précisions de classification très élevées. Une étude a ainsi obtenu 99,2% d'exactitude pour classer des outils en trois états (neuf, moyen, usé) à partir de spectrogrammes de signaux acoustiques traités par un CNN 3 couches [41]. De même, des architectures récurrentes (LSTM) ou hybrides (Conv LSTM) ont été proposées pour exploiter la dynamique temporelle des signaux de surveillance. Par exemple, Rajakannu et al [41] combinent la décomposition de signal CEEMDAN et un réseau Conv LSTM pour détecter l'usure sur des forêts CNC à partir de signaux d'émission acoustique « airain » (air borne) en obtenant d'excellents résultats de classification temporelle.

1.3.5 ADASYN (Adaptive Synthetic Sampling)

Introduit par He, Bai, Garcia et Li en 2008 [42], ADASYN est une méthode de sur-échantillonnage « adaptatif » qui génère davantage d'exemples synthétiques là où la classe minoritaire est la plus difficile (régions proches de la frontière et dominées par la majoritaire), afin de déplacer/affiner la frontière de décision et d'améliorer la détection de la classe rare. C'est le principe fondateur, une pondération par difficulté locale guide la quantité de synthèse autour de chaque point minoritaire.

ADASYN ajuste le nombre d'exemples synthétiques générés pour chaque échantillon minoritaire en fonction du degré de difficulté qu'il représente pour le modèle

d'apprentissage. Privilégiés minoritaires situés dans des régions fortement dominées par la classe majoritaire, considérés comme difficiles à classifier, sont privilégiés par l'algorithme. Cette stratégie permet ainsi d'affiner la frontière décisionnelle et d'améliorer la performance des modèles sur les classes minoritaires.

Des revues récentes confirment cette lecture, Jafarigol & Trafalis (2023) résument qu'ADASYN, dérivé de SMOTE, utilise un critère de densité/difficulté pour cibler les zones problématiques un avantage conceptuel par rapport aux schémas uniformes [43]. Côté résultats, plusieurs comparatifs empiriques montrent qu'ADASYN peut surpasser des techniques classiques (p. ex. SMOTE simple) dans des scénarios fortement déséquilibrés ou hétérogènes, Brandt et al [44] observe un gain d'ADASYN sur la plupart des métriques (avec arbres de décision), tout en rappelant que l'avantage dépend du modèle et des données, Mujahid et al [45], sur un large benchmark, positionnent ADASYN parmi les meilleures variantes quand on vise un meilleur rappel de la minoritaire.

Cela dit, l'issue reste dépendante des données, certaines études trouvent SMOTE (ou d'autres variantes) plus stables, d'où l'intérêt de tester des alternatives ciblées comme Borderline-SMOTE (sur-échantillonnage uniquement près de la frontière) ou des hybrides "sur-échantillonnage + nettoyage" type SMOTE-ENN / SMOTE-Tomek pour réduire le bruit et les chevauchements après synthèse [46, 47].

1.3.6 CTGAN (Conditional Tabular Generative Adversarial Network)

Présenté en 2019 par Xu et al [48] ont introduit CTGAN et posé ses bases expérimentales. CTGAN désigne un Conditional Tabular Generative Adversarial Network, un GAN conditionnel conçu pour les données tabulaires hétérogènes (colonnes continues souvent multimodales + catégories parfois déséquilibrées).

CTGAN s'utilise directement via SDV (Bibliothèque Python conçue pour centraliser la création de données synthétiques tabulaires), fournit le prétraitement tabulaire la génération conditionnelle (explicitement la classe minoritaire), et des outils d'évaluation. La doc et les billets officiels de SDV montrent l'entraînement et le suivi de la progression d'un modèle CTGAN (import, fit, sample, visualisations), ainsi que

l’échantillonnage conditionnel pour cibler une valeur rare quand on veut rééquilibrer une classe [48].

Conçu pour les tableaux “réels” (mélange de colonnes continues souvent multimodales et de catégories déséquilibrées), CTGAN reformule la génération de données tabulaires comme un GAN conditionnel qui apprend la distribution jointe des lignes et peut ensuite échantillonner des enregistrements plausibles, y compris pour les modalités rares. Son cœur de méthode tient à deux briques très spécifiques, une normalisation spécifique aux modes (mode-specific normalization) basée sur un mélange de variables continus non gaussiennes, et un schéma de training-by-sampling qui rééquilibre l’exposition du générateur aux catégories rares et lui force à voir suffisamment de catégories peu fréquentes pendant l’entraînement [48].

C’est précisément formulé par Xu et al [48], les données déséquilibrées entraînent également des possibilités d'entraînement insuffisantes pour les classes minoritaires, et CTGAN utilise un générateur conditionnel et un entraînement par échantillonnage pour traiter les colonnes discrètes déséquilibrées.

La littérature récente corrobore l’usage de CTGAN pour rééquilibrer, des études montrent que conditionner la génération sur la classe et sur-échantillonner la minoritaire avec CTGAN améliore souvent l’AUC/F1 par rapport aux sur-échantillonnages classiques (SMOTE/ROS) sur données tabulaires, et des variantes hybrides (p. ex. CTGAN-ENN) sont proposées spécifiquement pour le déséquilibre et le chevauchement entre classes [49].

Dans la pratique, CTGAN s’emploie comme un sur-échantillonneur génératif pour rééquilibrer des tableaux fortement déséquilibrés avant l’entraînement d’un classifieur, on entraîne le générateur sur les données réelles, puis on échantillonne conditionnellement la modalité rare (p. ex. label = Bad) afin d’augmenter la minoritaire sans casser les dépendances entre colonnes exactement le cas d’usage prévu par Xu et al [48] , qui cible explicitement les colonnes discrètes très déséquilibrées pour offrir davantage d’exposition aux catégories rares lors de l’apprentissage.

Dans des scénarios fortement déséquilibrés, plusieurs études rapportent que l'usage de CTGAN (ou d'un CGAN tabulaire proche) améliore les performances de classification par rapport à des schémas d'over sampling classiques (SMOTE/ROS), notamment sur l'AUC, la F1 et le rappel de la minoritaire, des comparatifs récents montrent même des gains nets lorsque le déséquilibre est sévère [50, 51].

Pour l'évaluation et la "gouvernance" des données synthétiques, l'idée n'est pas seulement de comparer des histogrammes, on vérifie surtout l'utilité aval avec le protocole TSTR on entraîne le modèle prédictif sur les données générées puis on le teste sur un lot réel tenu à part, ce qui mesure directement si les synthétiques portent bien les motifs discriminants utiles à la tâche (et on compare au TRTR comme ligne de base). C'est une pratique recommandée dans les guides d'évaluation de la donnée synthétique et largement documentée [52].

Des usages concrets de CTGAN en maintenance prédictive par exemple, dans l'industrie pâte-papier, un cadre "smart manufacturing" a utilisé CTGAN pour sur-échantillonner les rares "paper breaks", ce qui a amélioré nettement les métriques de détection après augmentation (jusqu'à +30 %/+20 %/+90 % selon le classifieur), l'étude cadrant explicitement l'objectif comme de la prédictive maintenance face à des événements rares [53].

Enfin, côté usure d'outil (CNC), si beaucoup d'articles emploient des GANs "séries temporelles" plutôt que CTGAN strictement, le principe est identique, équilibrer/augmenter les classes rares pour améliorer la prédiction d'usure ou de RUL. Un exemple multi-capteurs sur l'usure d'outil montre qu'un GAN pour l'augmentation des signaux améliore la classification par rapport au jeu déséquilibré initial [54].

1.4 Conclusion

Cette revue de la littérature a permis d'approfondir la thématique et de se familiariser avec les techniques phares mobilisés en maintenance prédictive d'usinage, en particulier pour la détection de l'usure outil en fraisage CNC à partir de vibrations. Elle éclaire l'évolution des approches, des indicateurs temporels et spectraux (RMS, kurtosis, FFT) vers des analyses temps-fréquence (ondelettes, scalogrammes) et

montre comment l'IA (ML puis Deep Learning) s'y greffe pour automatiser le diagnostic et améliorer la précision. Au regard des alternatives (analyse d'huile, thermique/IR, courant), l'analyse vibratoire apparaît comme la plus sensible et réactive pour capter les signatures fines de dégradation à la dent.

Par ailleurs, la littérature confirme l'intérêt d'une approche basée sur les données pour deux raisons principales :

- La maturité des capteurs/IoT et des chaînes d'acquisition facilite la collecte et le stockage de volumes massifs de données exploitables par des outils d'analyse toujours plus performants.
- La complexité des procédés et la non-stationnarité des signaux rendent les modèles physiques/simulations délicats, alors que les méthodes d'apprentissage apprennent directement les motifs discriminants à partir des données.

À l'issue de cette revue, la problématique se cadre comme un problème de classification visant à identifier l'état de l'outil (sain / usé) à partir de fenêtres vibratoires. On comparera des modèles de référence (k-NN, régression logistique, DT, RF, XGBoost) avec des architectures profondes (CNN/LSTM) alimentées par des représentations temps-fréquence. Compte tenu du déséquilibre de classes fréquent en contexte industriel, on intégrera des stratégies dédiées (ADASYN pour le sur-échantillonnage adaptatif, CTGAN pour la synthèse tabulaire conditionnelle) et adopterons des métriques pertinentes (précision/rappel, F1, AUC) pour sélectionner la méthodologie la plus adaptée à notre cas d'étude et atteindre un niveau de performance fiable en production.

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 destine à identifier l'état de l'outil en mode opérationnel. C'est-à-dire, de prédire si la machine fonctionne dans les conditions normales ou bien elle présente un problème avec l'outil.

2.2 Démarche de recherche

Pour atteindre les objectifs du projet, la démarche s'est inspirée des travaux étudiés dans le chapitre précédent. La mise en œuvre s'articule autour de quatre étapes complémentaires :

- Une étape de contextualisation : Cette phase est essentielle pour la compréhension et la familiarisation du thème lié à notre projet de recherche. Une revue de la littérature ciblée et des pratiques en TCM, afin de cerner précisément le périmètre de notre étude.
- Une étape d'analyse des besoins et de formulation de la problématique concevoir un système de maintenance prédictive capable d'identifier à temps l'état de l'outil en opération (fonctionnement normal vs problème lié à l'outil) pour éviter rebuts, dérives qualité et arrêts non planifiés.
- Une troisième étape de conception de la solution : définition de l'architecture et de la méthodologie de prédiction (chaîne d'acquisition, prétraitement et fenêtrage, choix des représentations du signal, sélection des modèles et des critères d'évaluation) menant au modèle retenu.
- Une dernière étape de validation : expérimentation sur données réelles, tests et mesures des performances des modèles pour vérifier que la méthode atteint les objectifs fixés.

Le tableau 1 ci-après récapitule ces quatre étapes, en précisant pour chacune les objectifs visés et les actions associées.

Tableau 1
Démarche de recherche

Étapes	Objectifs	Action
Contextualisation	• Établir un cadre conceptuel et opérationnel solide afin de délimiter précisément le périmètre (outil, capteurs, signaux, métriques) et de formuler les questions/hypothèses de recherche.	• Revue de la littérature.
Analyse des besoins	• Analyse et interprétations des points constatés à l'étape précédente. • Élaboration de la problématique industrielle et revue de la littérature.	• Documentation. • Base de données.
Conception de la solution	• Solution proposée. • Analyse vibratoire. • Conception des modèles de prédiction.	• Revue des techniques existantes. • Définition des étapes et élaboration de la méthode.
Test et évaluation	• Choix de la solution qui répond à nos exigences.	• Comparaison entre plusieurs techniques. • Evaluation 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.

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, L'usure de l'outil influence le comportement vibratoire du processus d'usinage. Un outil usé tend à générer plus de vibrations en raison d'efforts fluctuants, de phénomènes de broutage ou d'un moindre tranchant qui accroît les irrégularités de coupe. La mesure des vibrations via un accéléromètre monté sur la broche, le porte-outil ou la pièce est une méthode indirecte répandue pour la surveillance en temps réel, car les capteurs de vibration sont peu coûteux, faciles à installer et utilisables sur tout type de machine [14].

Pour notre application, on retient des accéléromètres MEMS tri-axiaux Bosch CISS (Les spécifications détaillées du capteur Bosch CISS sont fournies en Annexe A [55]). Ce choix privilégie la mesure d'accélération, particulièrement sensible aux transitoires de coupe, micro-impacts, débuts d'écaillage, variations rapides liées à l'usure. Le capteur est posé au plus près de la broche, côté palier avant, sur une surface plane avec fixation rigide pour éviter toute résonance parasite. Les trois axes sont orientés de façon cohérente avec la cinématique de la machine, un axe radial, un axe dans la direction d'avance, un axe axial.

Le jeu de données provient d'un site de production et couvre trois machines CNC (M01, M02, M03) opérant sur des pièces en aluminium. Il rassemble quinze opérations/outils récurrents, exécutés sur les trois machines à différentes périodes, de façon à capturer la variabilité process et les dérives inter-machines au fil du temps (comme illustré dans l'Annexe B). Le taux d'échantillonnage tri-axial retenu génère un volume d'environ 4,14 Go par jour, ce qui dépasse rapidement les capacités d'un traitement entièrement sur site. Dans ce contexte, l'infrastructure de données s'appuie sur une chaîne organisée (collecte, stockage hiérarchisé, annotation, prétraitements, apprentissage) capable d'absorber des flux continus tout en conservant la traçabilité des opérations. Cette architecture garantit que l'on conserve l'information utile à la détection d'usure tout en maîtrisant les coûts de stockage et de calcul. Elle facilite en outre la comparaison entre machines et entre outils [56, 57].

2.2.2 Analyse des données vibratoires

Analyse temporelle

L'analyse temporelle des formes d'onde vibratoires enregistrées chronologiquement met en évidence quand surviennent les événements anormaux et avec quelle intensité, à partir d'indicateurs comme la valeur quadratique moyenne (RMS), l'écart-type, l'amplitude crête, la kurtosis, le facteur de crête, etc. Cette lecture dans le temps offre une vue globale de l'état de l'outil. Au cœur du procédé, l'extraction de caractéristiques transforme le signal brut en des descripteurs représentatifs capables de distinguer les conditions saines des conditions dégradées. Concrètement, après acquisition et prétraitements, les signaux sont convertis en caractéristiques temporelles dont leur évolution sert à détecter les changements d'état et à suivre l'usure de l'outil au fil des opérations.

Analyse Fréquentielle

L'analyse fréquentielle consiste à passer du temps → fréquence, pour rendre lisible ce que le signal brut cache encore. Dans notre cadre, on calcule des spectres d'amplitude à partir des vibrations et l'on compare systématiquement l'outil sain à l'outil dégradé, superposition des courbes, lecture des écarts, observation des transferts d'énergie vers certaines zones spectrales. On complète cette lecture par des zooms autour de fréquences jugées sensibles et par l'extraction d'un petit jeu d'indicateurs spectraux qui servent ensuite à la classification. Le tout est décliné sur les trois axes et pour chaque machine, afin de comparer proprement les campagnes et de suivre l'évolution des signatures dans le temps. En pratique, cette approche condense l'information, met en évidence les phénomènes répétitifs et fournit des marqueurs simples à surveiller, tout en restant complémentaire de l'analyse temporelle. Lorsque les signaux deviennent très non stationnaires, on bascule naturellement vers une lecture temps–fréquence (ondelettes) pour garder la localisation temporelle sans perdre la richesse spectrale.

Analyse temps-fréquence

L'analyse par ondelettes prolonge la lecture classique en fréquence en la rendant temps–fréquence et multi-échelle, on "déplie" le signal pour localiser dans le temps les composantes spectrales, suivre des transitoires (éclats, chocs), des modulations et des débuts de broutement qui échappent souvent à une FFT pure. Dans le cadre de ce travail, on calcule une transformée en ondelettes continue pour obtenir des scalogrammes (cartes temps–fréquence) et un spectre global par bande (amplitude moyenne par fréquence) permettant une comparaison claire entre outil sain et outil dégradé, superposition des courbes, lecture des écarts, mise en évidence de transferts d'énergie sur certaines sous-bandes. Des zooms ciblent les zones sensibles et l'on extrait quelques indicateurs temps–fréquence simples qui alimentent ensuite la classification et un suivi par seuils. Déclinée sur les trois axes et toutes les machines, cette démarche complète l'analyse temporelle et la FFT, elle reste lisible, mais surtout robuste à la non-stationnarité, ce qui est décisif lorsque les conditions d'usinage évoluent au sein d'une même passe.

2.2.3 Prédiction du défaut en utilisant l'apprentissage automatique

Dans ce travail, une approche d'apprentissage supervisé a été mises en œuvre à partir des signaux bruts issus des capteurs triaxiaux.

Comme on fait la prédiction du défaut sur l'outil on se concentre sur un outil choisi (observé sur M01, M02, M03), on charge les CSV, on nettoie, on étiquette (Good=0, Bad=1) et on agrège en un jeu unique conservant les trois axes (x, y, z) et trois descripteurs compacts (RMS, kurtosis, magnitude |v|). Les échantillons/fenêtres sont standardisés, et le déséquilibre est corrigé uniquement sur les plis d'entraînement via ADASYN. On entraîne ensuite un panel de modèles (Random Forest, KNN, Decision Tree, XGBoost, Logistic Regression, SVM). L'évaluation s'appuie sur l'exactitude, Précision, Rappel, F1, ROC AUC, avec matrices de confusion agrégées et un contrôle simple du sur/sous-apprentissage. Le "meilleur" modèle est retenu via un tri priorisant le F1, puis le Recall et le ROC AUC, et le pipeline complet (scaler + ADASYN +

modèle) est sauvegardé pour la suite. La figure 4 présente le schéma simplifié de cette approche.

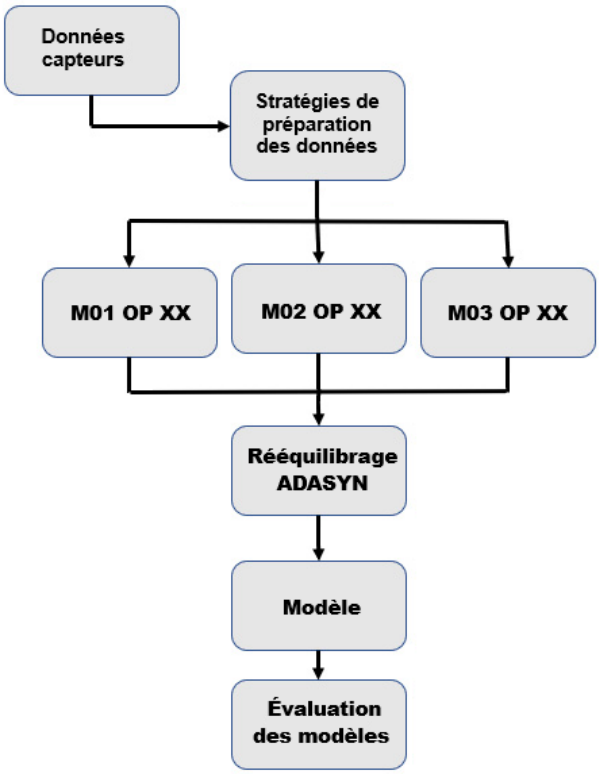


Figure 4
Schémas simplifié de l'approche basée sur l'apprentissage automatique

2.2.4 Prédiction du défaut en utilisant l'apprentissage profond

Dans la deuxième approche, un modèle d'apprentissage profond a été développé à partir des signaux bruts de vibration collectés sur les trois machines (M01, M02 et M03) et les quinze outils de coupe. Les signaux triaxiaux (axes X, Y et Z) sont d'abord segmentés en fenêtres temporelles de longueur fixe, normalisés et étiquetés en deux classes (« good » et « Bad »), de façon à constituer un jeu de données séquentiel exploitable par un réseau neuronal. Sur cette base, trois scénarios de préparation des données sont étudiés :

- Un premier scénario où le réseau est entraîné directement sur les données déséquilibrées.
- Un deuxième où le déséquilibre entre les classes est corrigé par la méthode ADASYN, qui génère des exemples synthétiques de la classe minoritaire.
- Un troisième scénario où un générateur CTGAN est utilisé pour produire des séquences supplémentaires de la classe minoritaire avant l'entraînement.

Le modèle est évalué sur un jeu de test séparé pour chacun des trois scénarios à l'aide des métriques classiques (exactitude, précision, rappel, F1-score, courbe ROC et matrice de confusion), ce qui permet de comparer l'impact d'ADASYN et de CTGAN sur les performances du réseau CNN–LSTM pour le diagnostic d'usure des outils, comme le montre la figure 5.

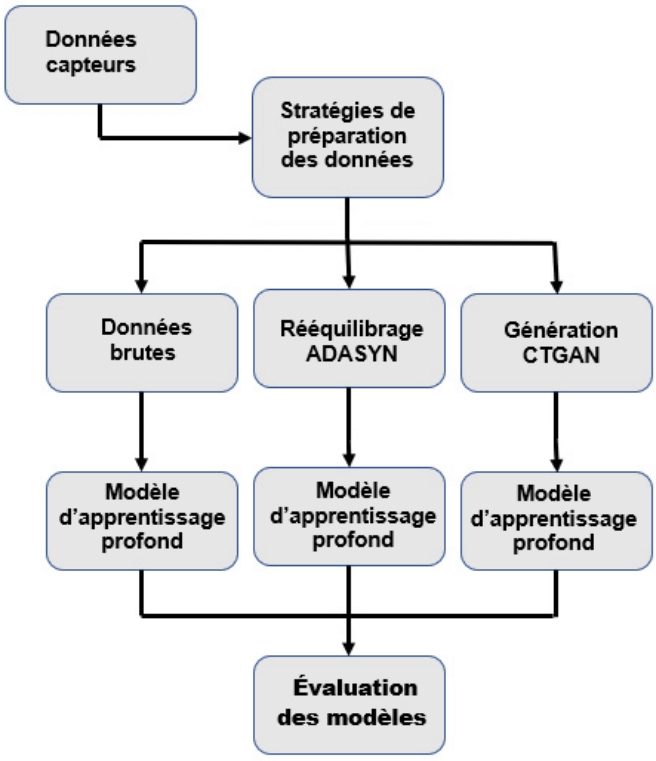


Figure 5
Schémas simplifié de l'approche basée sur l'apprentissage profond

2.3 Conclusion

Dans ce chapitre, on a présenté en détail la méthodologie adoptée pour la réalisation du projet de maintenance prédictive appliquée à l'usure des outils de coupe. Dans un premier temps, il a été nécessaire de bien comprendre le contexte industriel et la problématique d'usure, afin de préciser les objectifs. Ensuite, D'une part, l'analyse des signaux dans les domaines temporel et fréquentiel, complétée par l'analyse temps–fréquence. D'autre part, une approche d'apprentissage supervisé classiques ont été développées pour comparer différents algorithmes de classification largement utilisés en maintenance prédictive. Enfin, une deuxième approche basée sur l'apprentissage profond, et plus précisément sur un réseau CNN–LSTM.

3. DIAGNOSTIC DE DÉFAUT AVEC L'ANALYSE TEMPORELLE, FRÉQUENTIELLE ET TEMPS-FREQUENCE

3.1 Introduction

Dans ce chapitre, on va mettre en œuvre les approches retenues pour la détection des défauts dans les machines rotatives en appuyant sur une base de données réelle issue d'une fraiseuse CNC brownfield.

On va commencer par décrire les données collectées, puis on va présenter et visualiser les signaux disponibles afin d'évaluer la nécessité d'éventuels prétraitements avant analyse. On va appliquer ensuite les méthodes classiques d'analyse vibratoire, en particulier l'analyse temporelle, l'analyse fréquentielle et l'analyse temps-fréquence afin de caractériser l'état de fonctionnement de la machine et des outils de coupe. Enfin, on tire quelques conclusions de ces approches, ce qui permettra de motiver les méthodes plus avancées développées dans les chapitres suivants.

3.2 Description des données

L'acquisition des données est réalisée dans un environnement industriel réel, Les données utilisées dans ce travail proviennent de l'ensemble de données publié par [56, 57], impliquant trois machines-outils CNC à quatre axes utilisés pour l'usinage horizontal de pièces en aluminium. Ces machines sont équipées de capteurs MEMS Bosch CISS tri-axiaux, utilisés pour collecter indirectement les données d'accélération. Ces capteurs sont installés à l'arrière du boîtier de broche des machines CNC, une position stratégique choisie car elle est protégée des conditions extrêmes présentes dans la zone d'usinage (présence de copeaux, lubrifiants, vibrations importantes, etc.). La position des capteurs garantit des mesures cohérentes et une précision optimale dans la détection des vibrations comme le montre la figure 6.

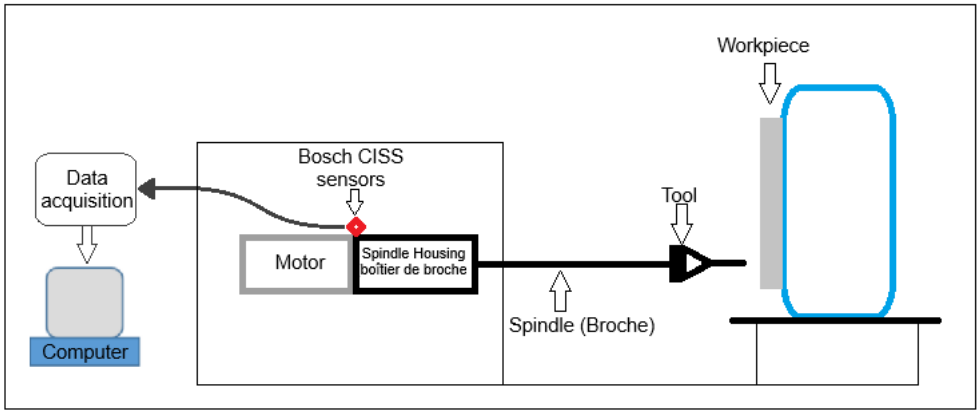


Figure 6
Schéma simplifié du système d'acquisition des données

Les capteurs MEMS collectent des données d'accélération sur les trois axes orthogonaux (X, Y, Z), échantillonnées à une fréquence élevée de 2 kHz. Cette fréquence d'échantillonnage est soigneusement choisie pour assurer la détection fiable des anomalies liées à l'usinage, couvrant notamment des fréquences critiques situées entre 75 Hz et 1 kHz, fréquences typiques associées aux défauts tels que les déséquilibres mécaniques, le mauvais alignement des outils ou encore les problèmes liés à l'usure de l'outil. La haute résolution temporelle de l'échantillonnage génère un volume conséquent de données, de l'ordre de 4,14 Go par jour [57].

Avec une fréquence d'échantillonnage de 2 kHz, les capteurs enregistrent 2000 points de mesure par seconde pour chaque axe (X, Y, Z), d'où on a le temps total calculé avec la formule (5) :

$$Temps\ total = \frac{nombre\ total\ de\ points}{Fréquence\ d'échantillonage} \quad (5)$$

Chaque point de mesure est enregistré à un intervalle régulier de temps, calculé comme suit (6) :

$$Intervalle\ d'échantillonage = \frac{1}{\text{Fréquence d'échantillonage}} = 0{,}5\ \text{ms} \quad (6)$$

Ainsi, les données sont enregistrées à intervalles de 0,5 milliseconde, ce qui garantit une capture précise et détaillée des événements dynamiques survenant lors de l'usinage.

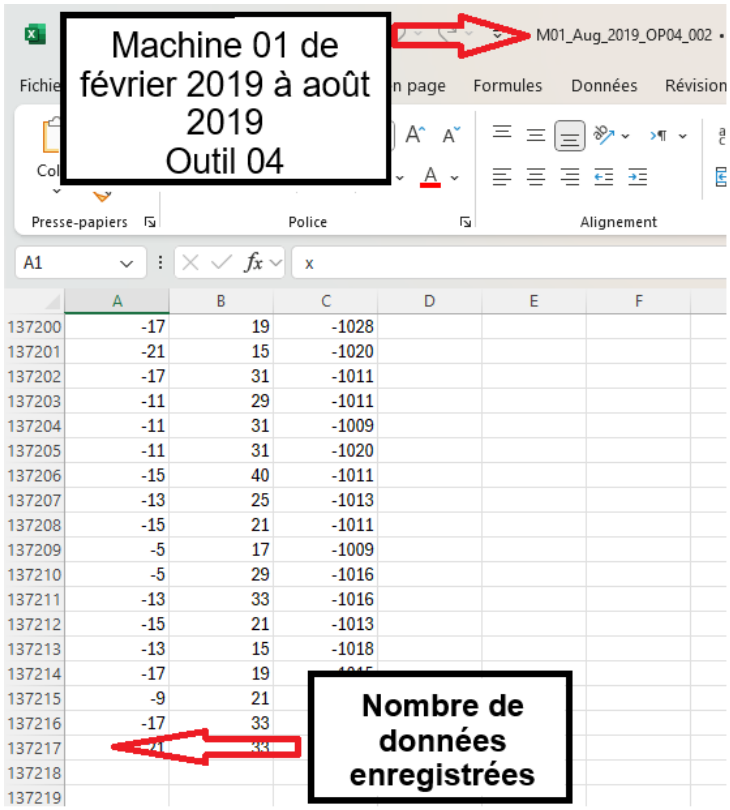
Les données ont été collectées régulièrement dans une usine de production à partir de trois machines CNC différentes (M01, M02 et M03) pendant la période d'octobre 2018 à août 2021. Chaque intervalle temporel est étiqueté sous la forme « Mois Année » et représente une fenêtre de 6 mois précédant cette date. Par exemple, l'étiquette « Août 2019 » correspond à la période allant de février 2019 à août 2019. Le jeu de données est constitué de 15 opérations d'usinage différentes réalisées sur les trois machines, à des périodes distinctes. Le Tableau 2 adapté de [57] présente un aperçu détaillé des caractéristiques de chacune de ces opérations.

Tableau 2
Opérations d'usinage collectées à partir des machines M01, M02 et M03

Opération	Description	Vitesse [Hz]	Avance [mm/s]	Durée [s]
OP00	Step Drill	250	≈ 100	≈ 132
OP01	Step Drill	250	≈ 100	≈ 29
OP02	Drill	200	≈ 50	≈ 42
OP03	Step Drill	250	≈ 330	≈ 77
OP04	Step Drill	250	≈ 100	≈ 64
OP05	Step Drill	200	≈ 50	≈ 18
OP06	Step Drill	250	≈ 50	≈ 91
OP07	Step Drill	200	≈ 50	≈ 24
OP08	Step Drill	250	≈ 50	≈ 37
OP09	Straight Flute	250	≈ 50	≈ 102
OP10	Step Drill	250	≈ 50	≈ 45
OP11	Step Drill	250	≈ 50	≈ 59
OP12	Step Drill	250	≈ 50	≈ 46
OP13	T-Slot Cutter	75	≈ 25	≈ 32
OP14	Step Drill	250	≈ 100	≈ 34

De plus, pour chaque opération d'usinage, le nombre total de points de données collectés peut être estimé en appliquant une simple règle temporelle. En multipliant la durée totale de l'opération (en secondes) par la fréquence d'échantillonnage du

capteur (2 kHz), on obtient le nombre total de points mesurés par axe. Par exemple, pour l'opération OP04 d'une durée approximative de 64 secondes, cela correspond à 64 × 2000 = 128 000 points pour un seul axe. La figure 7 montre un exemple de répartition de données pour un outil choisis au hasard.



	A	B	C
137200	-17	19	-1028
137201	-21	15	-1020
137202	-17	31	-1011
137203	-11	29	-1011
137204	-11	31	-1009
137205	-11	31	-1020
137206	-15	40	-1011
137207	-13	25	-1013
137208	-15	21	-1011
137209	-5	17	-1009
137210	-5	29	-1016
137211	-13	33	-1016
137212	-15	21	-1013
137213	-13	15	-1018
137214	-17	19	
137215	-9	21	
137216	-17	33	
137217	-21	33	
137218			
137219			

Figure 7
Exemple pour une répartition des données

Dans la suite du mémoire, on se concentre sur l'outil 07, qui sert de cas d'étude principal. Les scripts d'analyse restent toutefois génériques en modifiant simplement le paramètre de configuration tool_id (ainsi que la liste des machines si nécessaire), on peut appliquer exactement la même chaîne de traitement à n'importe quel autre outil de la base de données. Un extrait de cette configuration Python, montrant la sélection de l'outil et des machines analysés, est présenté en Annexe C.

3.3 Analyse temporelle

Afin de comparer le comportement vibratoire de la machine pour les états Good (outil en bon état) et Bad (outil usé), on commence par analyser les signaux dans le domaine temporel, puis on extrait plusieurs descripteurs statistiques classiques du domaine temporel afin de voir s'ils permettent de distinguer les deux états, avant de passer aux analyses fréquentielles et temps–fréquence.

3.3.1 Visualisation d'une composante temporelle

Pour illustrer l'impact de l'usure de l'outil sur le niveau de vibration, on extrait d'abord une composante harmonique particulière à partir du signal brut. Un filtre passe-bande étroit centré à 200 Hz est appliqué afin d'isoler la composante associée à l'harmonique dite 13X. La Figure 8 montre l'évolution temporelle de cette composante pour les états Bad et Good sur l'axe X de la machine M01.

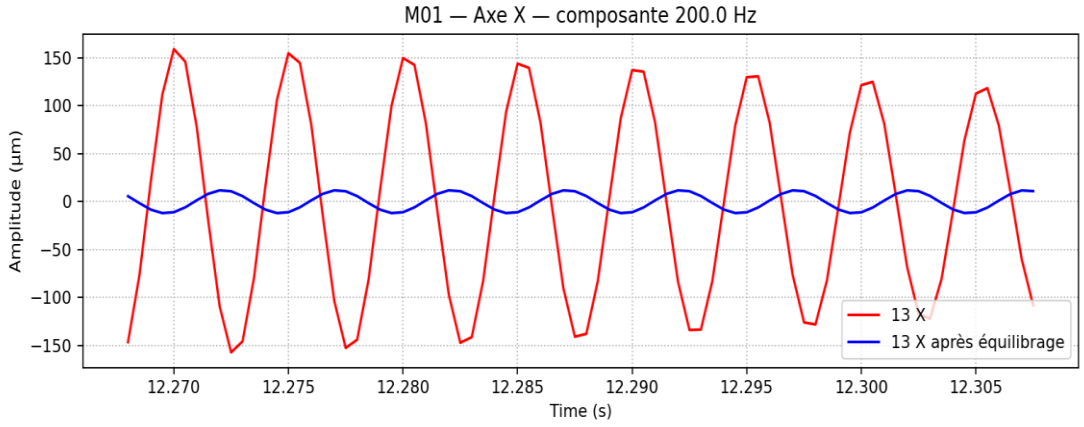


Figure 8
Les signaux vibratoires avant et après équilibrage

On voit que l'amplitude crête de la composante 13X est nettement plus élevée en état Bad (courbe rouge) qu'en état Good (courbe bleue). Après équilibrage ou remplacement de l'outil, la vibration à cette fréquence est fortement réduite et la forme d'onde devient plus régulière. On comprend déjà ici que l'amplitude de cette composante harmonique est directement liée à la condition de l'outil.

3.3.2 Extraction des descripteurs statistiques

Pour aller au-delà de cet exemple visuel, on extrait ensuite plusieurs descripteurs statistiques à partir des signaux temporels bruts pour chaque machine (M01 à M03), chaque axe de mesure (X, Y, Z) et pour les deux états Good et Bad. On retient notamment :

- La valeur efficace (RMS)
- L'écart-type (STD)
- La kurtosis
- Le facteur de crête (CrestFactor).

Tableau 3
Indicateurs temporels par machine axe X OP 07 (Good vs Bad)

Machine	Status	x_RMS	Δ%RMS	x_STD	Δ%STD	x_Kurtosis	x_CrestFactor
M01	Bad	530.236	+28.600	530.235	+28.600	5.468	5.752
	Good	412.168		412.156		9.014	6.260
M02	Bad	501.112	+23.800	500.849	+23.700	5.111	4.241
	Good	404.791		404.785		9.015	5.630
M03	Bad	525.112	+16.000	525.079	+16.100	5.760	5.511
	Good	452.573		452.434		8.424	19.128

Sur l'axe X comme le montre le tableau 3, on observe une augmentation systématique de la valeur RMS entre les états Good et Bad. Pour la machine M01, la RMS passe d'environ 412 µm à 530 µm, soit une hausse d'environ 28,6 %. Des augmentations comparables apparaissent pour M02 (+23,8 %) et M03 (+16,0 %). L'écart-type suit la même tendance, ce qui montre qu'en état Bad le niveau de vibration global est plus élevé et plus dispersé dans le temps.

Tableau 4
Indicateurs temporels par machine axe Y OP 07 (Good vs Bad)

Machine	Status	y_RMS	Δ%RMS	y_STD	Δ%STD	y_Kurtosis	y_CrestFactor
M01	Bad	317.886	+28.600	316.347	+70.300	8.790	12.665
	Good	188.759		185.754		11.253	14.929
M02	Bad	337.562	+122.100	337.331	+122.000	7.203	7.717
	Good	152.010		151.944		5.722	10.269
M03	Bad	304.165	+40.300	303.899	+40.700	6.557	9.952
	Good	216.738		216.048		10.560	14.995

Tableau 5
Indicateurs temporels par machine axe Z OP 07 (Good vs Bad)

Machine	Status	z_RMS	Δ%RMS	z_STD	Δ%STD	z_Kurtosis	z_CrestFactor
M01	Bad	1078.678	+5.100	339.666	+142.900	6.846	3.745
	Good	1026.803		139.824		4.411	1.902
M02	Bad	1092.284	+4.600	321.439	+107.000	4.086	2.852
	Good	1044.568		155.321		4.779	2.403
M03	Bad	1053.560	+0.800	258.990	+17.200	5.052	2.025
	Good	1044.989		221.060		7.104	3.575

Les axes Y et Z présentent des comportements similaires, avec une hausse marquée de la RMS et de l'écart-type en état Bad, en particulier sur l'axe Y de la machine M02 où la RMS est fortement augmentée (Tableaux 4 et 5). On peut donc dire que la dégradation de l'outil s'accompagne d'une augmentation nette du niveau vibratoire sur l'ensemble des directions mesurées.

En revanche, la kurtosis et le facteur de crête n'évoluent pas de façon aussi simple. Sur l'axe X, la kurtosis est plus élevée en état Good (valeurs autour de 8–9) qu'en état Bad (5–6), et le facteur de crête diminue également lorsque l'outil se dégrade, notamment pour la machine M03. Des tendances proches se retrouvent sur les axes Y et Z. On en déduit que, pour ce jeu de données, l'usure ne se traduit pas par l'apparition de chocs isolés très marqués, mais plutôt par une augmentation plus continue du niveau de vibration. Dans ce contexte, la RMS et l'écart-type offrent des indicateurs temporels plus robustes que la kurtosis ou le facteur de crête pour distinguer les états Good et Bad.

3.3.3 Discussion

À partir de cette analyse temporelle, on voit que l'état Bad de l'outil est associé à :

- Une augmentation significative du niveau de vibration global (RMS, écart-type)
- Une composante harmonique 13X beaucoup plus marquée en amplitude.

La RMS reste un indicateur simple et largement utilisé en surveillance vibratoire, car elle est directement liée à l'énergie du signal et permet de suivre l'évolution globale de la machine. En revanche, la kurtosis et le facteur de crête se révèlent moins stables

dans ce cas et ne suffisent pas, à eux seuls, pour caractériser finement l'usure ou identifier le type de défaut.

On en conclut que les indicateurs temporels donnent déjà une première séparation entre les états Good et Bad, mais qu'ils restent limités pour une détection précoce et pour l'interprétation détaillée des mécanismes physiques. C'est pourquoi on complète cette étude par une analyse plus poussée dans le domaine fréquentiel, puis dans le domaine temps–fréquence, afin d'exploiter la structure spectrale des signaux de vibration.

3.4 Analyse fréquentielle

Dans cette section, on analyse les signaux de vibration dans le domaine fréquentiel pour mieux mettre en évidence l'effet de l'usure de l'outil sur le contenu spectral. On commence par examiner la répartition de l'énergie en fréquence à l'aide des densités spectrales de puissance (PSD), puis on s'intéresse plus précisément à certaines composantes harmoniques mises en évidence par la transformée de Fourier (FFT). Enfin, on utilise des marqueurs spectraux définis autour de fréquences d'intérêt afin de quantifier les différences entre les états Good et Bad et d'identifier les zones fréquentielles les plus sensibles à l'usure.

3.4.1 Estimation des densités spectrales de puissance

Dans ce travail, on s'appuie sur les PSD pour repérer les zones fréquentielles où la différence entre les états Good et Bad est la plus marquée. Les pics observés autour de 200, 400, 600 et 800 Hz, ainsi que leurs bandes voisines, sont retenus comme fréquences d'intérêt et serviront de base aux marqueurs spectraux utilisés par la suite.

Pour chaque machine et chaque axe, la PSD est calculée sur l'ensemble des fichiers disponibles, puis on prend la médiane des PSD obtenues afin de limiter l'influence d'éventuels signaux atypiques.

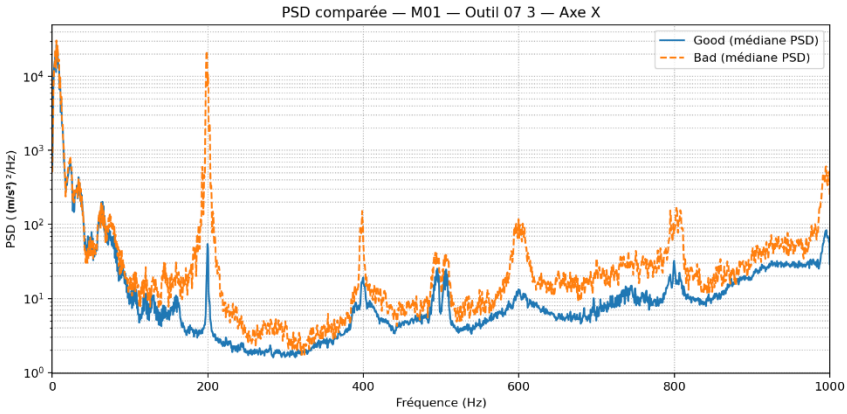


Figure 9
PSD médianes des états Good et Bad machine M01, axe X, OP 07

La Figure 9 illustre, à titre d'exemple, la comparaison des PSD médianes pour la machine M01 sur l'axe X OP 07. On voit que l'état Bad présente un niveau spectral globalement plus élevé que l'état Good, avec des différences particulièrement autour de certaines fréquences distinctes. Cette augmentation de l'énergie spectrale est cohérente avec la hausse de la RMS observée dans le domaine temporel.

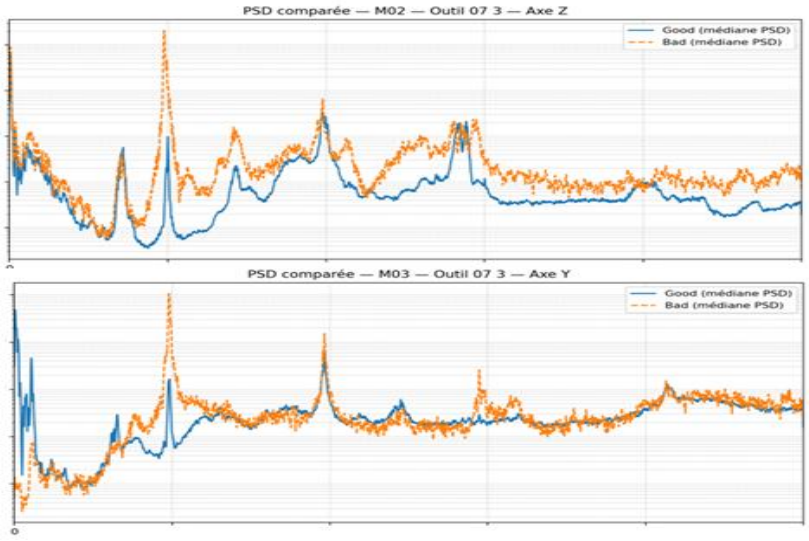


Figure 10
Figure similaire pour M02 M03

Des comportements similaires comme le montre la figure 10, sont observés sur les machines M02 et M03, ainsi que sur les axes Y et Z, même si l'intensité des écarts varie selon la machine et la direction de mesure.

Pour quantifier plus précisément les différences entre les états Good et Bad, on réalise des zooms sur des bandes étroites centrées autour de 200, 400, 600 et 800 Hz. La Figure 11 montre, pour la machine M01 et l'axe X, que les pics spectraux situés dans ces bandes sont nettement plus élevés en état Bad.

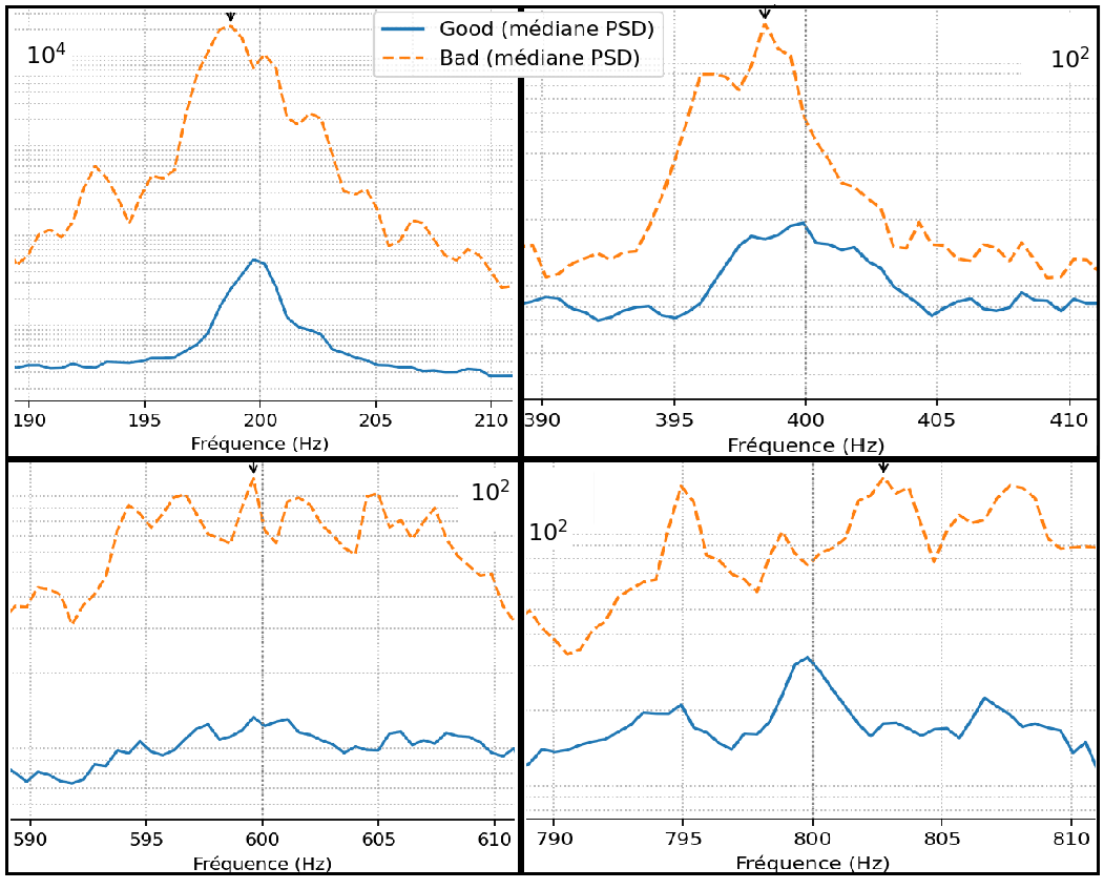


Figure 11
Zoom autour de 200, 400, 600 et 800 Hz pour M01, axe X, OP 07

À partir des zooms de PSD, on extrait pour chaque bande centrée autour de f_0(200, 400, 600 et 800 Hz) l'amplitude du pic spectral pour les deux états, notée $PSD_{Good}(f_{pk})$

et $PSD_{Bad}(f_{pk})$. Le tableau 6 récapitule ces valeurs pour chaque machine et chaque axe. À partir de ces pics, on calcule ensuite à l'aide de l'équation 7, pour chaque fréquence d'intérêt, un ratio en décibels [58, 59] défini par :

$$ratio_peak_dB(f_0) = 10\ log_{10}(\frac{PSD_{Bad}(f_{pk})}{PSD_{Good}(f_{pk})}) \quad (7)$$

Cette relation permet de quantifier directement, en dB, l'écart de niveau spectral entre l'outil usé (Bad) et l'outil sain (Good) à la fréquence considérée.

Tableau 6
Marqueurs spectraux des pics pour M01, axe X, OP 07

Tool	Machine	Axis	f_0(Hz)	f_{pk}(Hz)	PSD_Peak_{Good}	PSD_Peak_{Bad}	$ratio_peak_dB$
07	M01	X	200	198.730	25.5562	22041.5	29.3574
07	M01	X	400	398.438	16.2226	151.867	9.71342
07	M01	X	600	599.609	13.3126	117.760	9.46737
07	M01	X	800	802.734	17.6647	167.885	9.77904

Les valeurs obtenues montrent que la bande centrée à 200 Hz est la plus fortement impactée par l'usure :

Pour la machine M01, axe X, le ratio de pic atteint environ +30 dB, ce qui correspond à un rapport d'énergie proche de 10^3 entre les états Bad et Good. Des augmentations notables, de l'ordre de 7 à 10 dB, sont également observées autour de 400, 600 et 800 Hz. Les autres axes présentent la même tendance, avec des pics systématiquement plus élevés en état Bad. En combinant les zooms de PSD et les ratios en dB issus du tableau, on voit donc clairement que l'usure de l'outil se traduit par un renforcement très marqué de certaines composantes fréquentielles localisées.

À partir de ces résultats en densité spectrale de puissance, on s'intéresse ensuite au spectre d'amplitude obtenu par la transformée de Fourier (FFT), afin d'observer plus finement les raies harmoniques associées à ces bandes fréquentielles et de comparer directement leurs amplitudes entre les états Good et Bad.

3.4.2 Spectres de transformée de Fourier (FFT)

En complément de la PSD, on exploite le spectre d'amplitude FFT unilatéral pour analyser plus finement certaines composantes harmoniques. Pour chaque axe et chaque machine, on calcule la FFT d'un signal représentatif en appliquant une fenêtre de Hann et une correction de gain cohérente. La Figure 12 présente, pour M01 et l'axe X, les spectres d'amplitude des états Good et Bad.

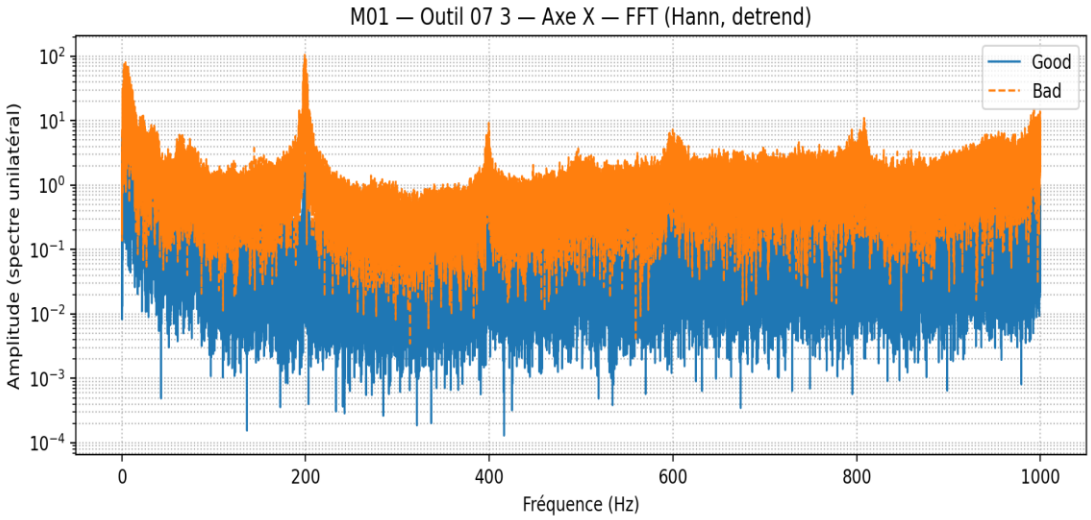


Figure 12
Spectres FFT d'amplitude des signaux de vibration pour l'OP 07, machine M01, axe X

On observe que les raies situées à proximité de 200, 400, 600 et 800 Hz sont plus marquées en état Bad, ce qui est cohérent avec les résultats obtenus sur les PSD. Les marqueurs FFT extraits autour de ces fréquences (amplitude au bin fréquentiel le plus proche et ratio Bad/Good en dB) sont regroupés dans le Tableau 7. On voit en particulier qu'autour de 200, 600 et 800 Hz, l'amplitude des raies est multipliée par un facteur compris entre 8 et 30 lorsque l'outil est usé, alors que la raie à 400 Hz n'augmente que faiblement. Ces composantes apparaissent ainsi comme fortement sensibles à l'usure de l'outil.

Tableau 7
Marqueurs spectraux issus de la FFT pour l'OP 07, machine M01, axe X

Tool	Machine	Axis	f0_hz	amp_good	amp_bad	ratio_amp_db
07	M01	X	200	0.207296	6.06347	29.3226
07	M01	X	400	0.163656	0.218836	2.52377
07	M01	X	600	0.217671	3.70583	24.6217
07	M01	X	800	0.28401	2.31462	18.2229

On se limite ici à la présentation détaillée de la machine M01 sur l'axe X. Les résultats analogues obtenus pour les axes Y et Z, qui montrent la même tendance avec des raies plus marquées en état Bad, sont regroupés en Annexe D.

3.4.3 Discussion

L'analyse fréquentielle a montré que l'usure de l'outil ne se limite pas à une simple augmentation globale du niveau vibratoire, mais qu'elle amplifie fortement certaines composantes bien localisées en fréquence. Les PSD médianes mettent en évidence des différences marquées entre les états Good et Bad, en particulier autour de 200 Hz, puis de 600 et 800 Hz. Les marqueurs spectraux issus de la FFT confirment ces observations :

Pour la machine M01, l'amplitude des raies à 200, 600 et 800 Hz augmente de plusieurs dizaines de décibels lorsque l'outil est usé, ce qui traduit un renforcement très important de ces harmoniques. Les axes Y et Z présentent la même tendance (voir Annexe D) ce qui confirme la robustesse de ces fréquences comme indicateurs de l'usure.

Cependant, la PSD et la FFT restent des outils essentiellement "moyennés" dans le temps, elles décrivent bien quelles fréquences sont excitées par l'outil usé, mais pas comment ces composantes évoluent au cours de l'usinage (démarrage, régime établi, éventuelles variations transitoires). Pour mieux suivre l'apparition et la dynamique de ces harmoniques, on complète donc cette étude par une analyse temps–fréquence basée sur la transformée en ondelettes, qui permet de représenter simultanément le contenu fréquentiel et son évolution temporelle.

3.5 Analyse temps-fréquence par transformée en ondelettes continues

3.5.1 Spectre CWT

Dans cette partie, on montre comment la transformée en ondelettes continues (CWT) complète l'analyse temporelle et fréquentielle classique et renforce le diagnostic d'usure. Pour un signal vibratoire $x(t)$, la CWT s'écrit, à l'échelle a et au temps b :

$$W_x(a,b) = \int_{-\infty}^{+\infty} x(t)\,\psi^*_{a,b}(t)\,dt, \quad \psi_{a,b}(t) = \frac{1}{\sqrt{a}}\psi\left(\frac{t-b}{a}\right) \tag{8}$$

où ψ est ici une ondelette de Morlet complexe. Le scalogramme correspond alors à la densité d'énergie temps–fréquence $|W_x(a,b)|^2$, utilisée pour visualiser l'évolution des bandes fréquentielles au cours de l'usinage. Cette approche permet de voir à la fois quelles bandes fréquentielles sont excitées (par exemple autour de 200, 400, 600 et 800 Hz) et comment leur niveau évolue au cours du temps pendant l'usinage.

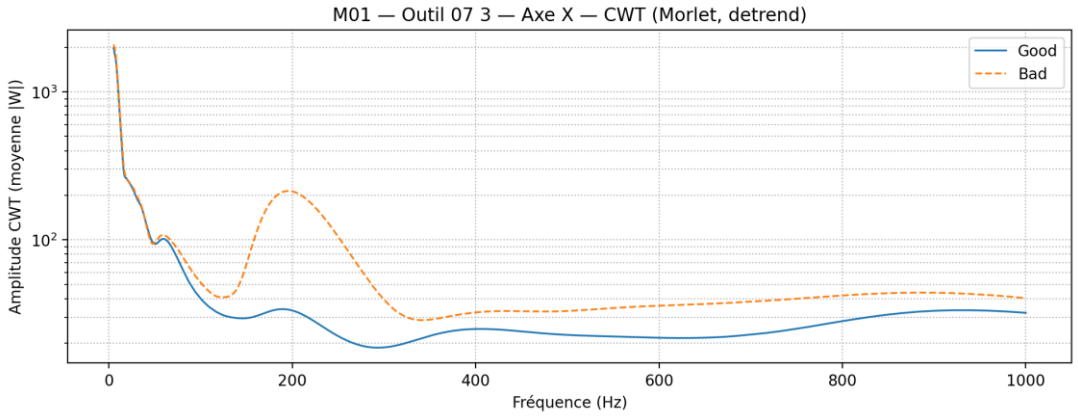


Figure 13
Spectre moyen CWT Good/Bad pour M01, OP 07, axe X

La Figure 13 présente le spectre moyen CWT pour la machine M01, axe X, en états Good et Bad. On voit que, sur l'ensemble de la bande 5–1000 Hz, la courbe Bad reste au-dessus de la courbe Good, avec une différence plus marquée dans certaines zones fréquentielles.

Tableau 8
Marqueurs CWT pour M01, OP 07, axe X

Tool	Machine	Axis	f0_hz	amp_good	amp_bad	ratio_amp_db
07	M01	X	200	33.0033	211.016	16.1152
07	M01	X	400	24.9619	32.3281	2.24603
07	M01	X	600	21.8051	35.7511	4.29463
07	M01	X	800	27.9136	41.8524	3.5181

Le Tableau 8 quantifie ces écarts autour des fréquences d'intérêt 200, 400, 600 et 800 Hz. Autour de 200 Hz, l'amplitude moyenne de la CWT passe d'environ 33 en état Good à plus de 211 en état Bad, soit un gain d'environ 16 dB. Les bandes situées autour de 400, 600 et 800 Hz présentent aussi des hausses, plus modérées (entre 2 et 4 dB). Ces résultats confirment que l'état Bad concentre davantage d'énergie dans les mêmes zones spectrales que celles mises en évidence par la PSD et la FFT, tout en montrant que cette augmentation reste présente en moyenne sur toute la durée du signal.

3.5.2 Scalogrammes temps-fréquence

Les scalogrammes de la machine M01, axe X, permettent de visualiser directement l'évolution temps–fréquence des vibrations pour les états Good et Bad. À partir de ce scalogramme, on calcule un spectre moyen CWT en moyennant l'amplitude sur toute la durée d'observation. En discret, pour une fréquence (ou échelle) f_k et N instants t_n, on a par exemple :

$$S_x(f_k) = \frac{1}{N}\sum_{n=1}^{N} | W_x(f_k, t_n) | \tag{9}$$

Le spectre moyen CWT obtenu pour M01, OP 07, axe X montre que, sur la bande 5–1000 Hz, la courbe Bad reste systématiquement au-dessus de la courbe Good, avec un écart particulièrement marqué autour de 200 Hz. Cet écart est quantifié en décibels qui atteint ici environ $16\ \mathrm{dB}$ à 200 Hz et reste positif autour de 400, 600 et 800 Hz. Les scalogrammes temps–fréquence confirment cette tendance en montrant que ces bandes sont excitées de manière plus intense, plus large et plus persistante dans le

temps en état Bad. On n’observe pas seulement des pics ponctuels, mais des zones d’énergie récurrentes qui suivent l’usinage. Ce qui est présenté par la figure 14 :

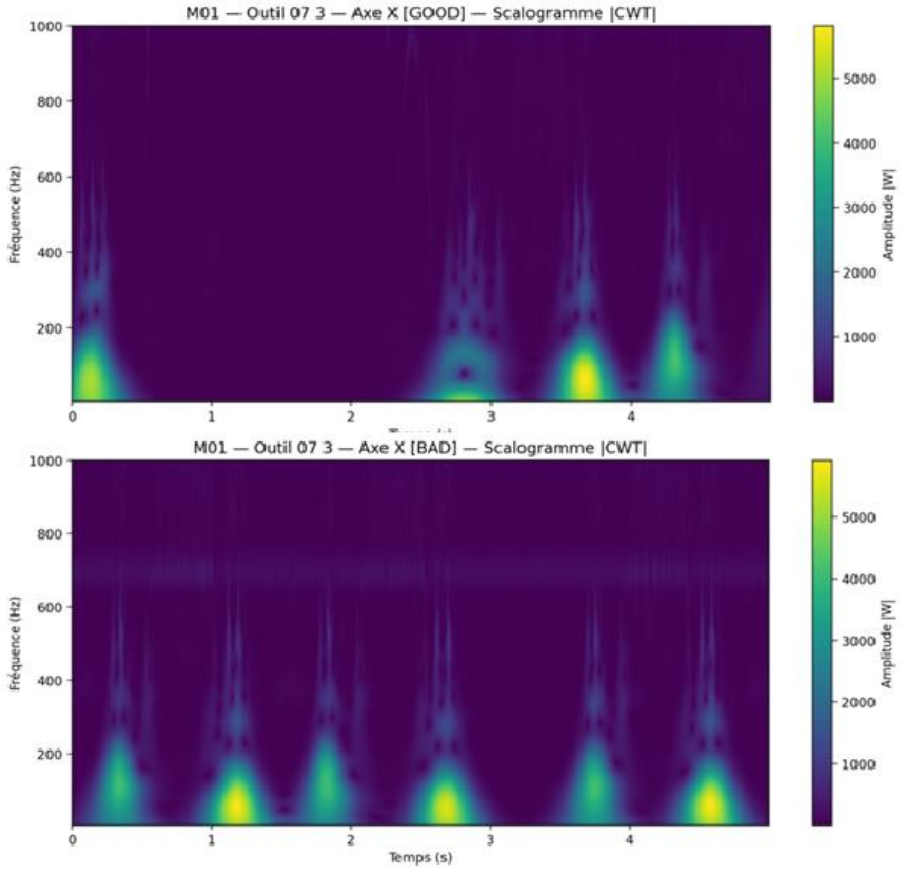


Figure 14
Scalogrammes Good et Bad pour M01, OP 07, axe X

On distingue des zones d’énergie renforcée sous forme de bandes quasi horizontales, en particulier autour de 200 Hz et, dans une moindre mesure, autour de 400, 600 et 800 Hz. En état Bad, ces bandes sont nettement plus intenses et plus larges que pour l’état Good, ce qui montre que l’outil usé excite ces fréquences de manière plus continue au cours de l’usinage, et pas seulement à un instant isolé.

Ces observations complètent bien les analyses PSD et FFT, les mêmes fréquences ressortent comme critiques, mais la CWT montre en plus que leur niveau reste élevé

sur toute la fenêtre temporelle considérée. Cela confirme que l'usure de l'outil se manifeste par un comportement vibratoire anormal stable dans le temps. Les scalogrammes des axes Y et Z, qui présentent une tendance similaire, sont regroupés en annexe E.

3.5.3 Discussion

Les résultats CWT confirment d'abord ce qu'on avait observé avec la PSD et la FFT, les bandes situées autour de 200, 400, 600 et 800 Hz sont systématiquement plus énergétiques en état Bad qu'en état Good. Le spectre moyen CWT montre que, sur toute la durée étudiée, l'amplitude moyenne $|W|$reste plus élevée pour l'outil usé, en particulier autour de 200 Hz, ce qui va dans le même sens que les écarts en dB obtenus sur les spectres de puissance.

Les scalogrammes apportent une information supplémentaire importante, on voit que ces bandes ne sont pas activées ponctuellement, mais reviennent de manière répétée au fil du temps, sous forme de "taches" d'énergie plus intenses et plus étalées en état Bad. Autrement dit, l'usure de l'outil se manifeste par un excès d'énergie fréquentielle qui est à la fois localisé en fréquence et persistant dans le temps. Cela suggère que les fréquences déjà identifiées (en particulier autour de 200 Hz) peuvent servir de bons indicateurs pour un suivi en ligne, et que des descripteurs temps–fréquence dérivés de la CWT pourraient enrichir encore la détection automatique de l'usure dans les travaux futurs.

3.6 Conclusion

Dans ce chapitre, on a appliqué plusieurs niveaux d'analyse vibratoire sur les signaux issus de l'outil 07 afin de caractériser l'effet de l'usure. Dans le domaine temporel, les indicateurs simples (RMS, valeur maximale, écart-type, facteur de crête) montrent déjà une augmentation nette du niveau de vibration entre les états Good et Bad. Ces grandeurs restent cependant globales et ne permettent pas d'identifier précisément quelles composantes sont affectées.

L'analyse fréquentielle par PSD et FFT a ensuite permis d'isoler des bandes beaucoup plus sensibles à l'usure, principalement autour de 200 Hz, puis de 600 et 800 Hz. Les

marqueurs spectraux (ratios en dB entre états Good et Bad) ont montré des écarts très importants sur ces raies, ce qui en fait de bons candidats pour le suivi de l'état de l'outil. Enfin, l'analyse temps–fréquence par CWT a confirmé ces observations tout en montrant que ces bandes restent fortement excitées au cours du temps en état Bad, ce qui suggère un comportement vibratoire anormal stable plutôt qu'un simple phénomène transitoire.

On dispose ainsi d'un ensemble cohérent d'indicateurs issus des domaines temporel, fréquentiel et temps–fréquence, capables de distinguer de façon robuste les états Good et Bad. Cependant, dans l'état actuel du travail, ces descripteurs sont surtout utilisés lorsque l'usure est déjà présente, afin d'analyser et de comparer les deux états.

Les bandes fréquentielles mises en évidence et les marqueurs associés constituent néanmoins une base intéressante pour développer, à terme, une approche plus automatique de détection de l'usure en temps réel. En intégrant ces indicateurs dans un système de surveillance en ligne et dans des algorithmes de décision, on pourrait suivre l'évolution de l'usure avant qu'elle ne devienne critique et planifier le changement d'outil de manière proactive, en limitant autant que possible les arrêts non planifiés de la machine.

4. PRÉDICTION DU DÉFAUT EN UTILISANT L'APPRENTISSAGE AUTOMATIQUE

4.1 Introduction

L'apprentissage automatique occupe aujourd'hui une place centrale dans la science des données et dans la maintenance prédictive. En s'appuyant sur des méthodes statistiques et des algorithmes d'optimisation, il permet d'extraire automatiquement de l'information pertinente à partir de signaux capteurs pour effectuer des tâches de classification ou de prédiction. Dans ce chapitre, on exploite ces principes pour s'attaquer à un problème concret de maintenance prédictive, la détection de l'usure d'outils de coupe montés sur des machines-outils CNC, à partir de mesures vibratoires issues d'un accéléromètre tri-axial Bosch CISS. L'objectif est de développer un modèle capable de distinguer un fonctionnement normal d'un fonctionnement dégradé afin de contribuer à la mise en place d'une stratégie de maintenance plus proactive dans le contexte des usines intelligentes.

Dans la première partie du chapitre, on présente la méthodologie adoptée pour le développement du modèle, préparation et segmentation des données, extraction des indicateurs, équilibrage des classes et mise en forme des jeux d'apprentissage et de test. Dans la deuxième partie, on applique cette méthodologie en détaillant les algorithmes d'apprentissage supervisé retenus ainsi que leur configuration. Enfin, on discute les performances obtenues et la capacité des modèles à discriminer correctement les états de santé des outils. Dans tout ce qui suit, la classe 0 correspond à un état de fonctionnement sain de l'outil (« Good »), tandis que la classe 1 représente un état dégradé ou usé (« Bad »).

4.2 Étapes de construction des modèles de prédiction

Dans cette partie, on va étudier des algorithmes d'apprentissage machine. 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. Dans une première approche, on travaille sur un même outil utilisé sur trois machines (M01, M02, M03), pour chaque couple machine/état (good/Bad), tous les CSV sont chargés, étiquetés (0=good, 1=Bad) et agrégés en un seul jeu, la colonne machine restant uniquement pour la traçabilité (voir Annexe F pour la bibliothèque Python et le chargement des données).

On scinde d'abord en apprentissage/test (80/20) avec stratification, on ajuste le Standard Scaler sur le train uniquement et on enrichit les variables via une fenêtre glissante (taille contrôlée) en ajoutant à (X, Y, Z) des descripteurs temps robustes, RMS et kurtosis par axe, ainsi que la magnitude, afin de mieux capter l'intensité et l'impulsivité liées à l'usure.

On applique ensuite ADASYN sur le train (taux ajustable) pour générer des exemples minoritaires dans les zones difficiles, tout en laissant le test dans sa distribution naturelle pour éviter toute fuite d'information. Ensuite, on entraîne un panel de classifieurs hétérogènes (SVM, Random Forest, KNN, Arbre de décision, Régression logistique, XGBoost). Pour rendre l'évaluation robuste, on utilise une validation croisée stratifiée à K = 5 plis.

Puis on évalue avec Accuracy, Precision, Recall, F1, ROC-AUC et la matrice de confusion (TN, FP, FN, TP). On agrège les matrices de confusion. On contrôle l'état d'ajustement via ΔF1 train–test (fit OK / sous / sur-apprentissage) et on classe les modèles selon le critère principal retenu, tout en reportant l'ensemble des scores pour une comparaison transparente.

La figure 15 donne un schéma simplifié des étapes de construction du modèle :

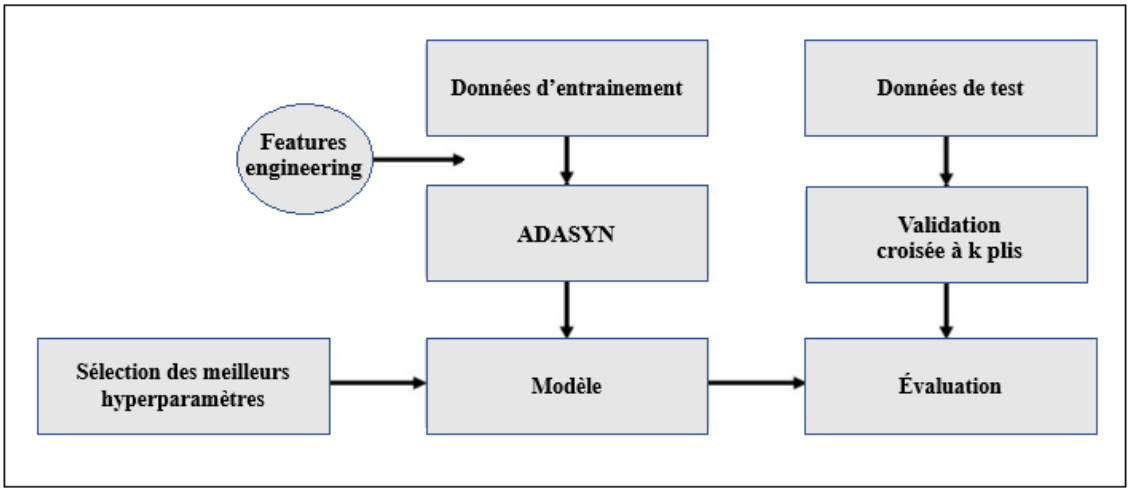


Figure 15
Pipeline global du modèle de prédiction ML

L'évaluation de la performance d'un modèle de classification débute par un résumé des résultats en quatre catégories. Le premier groupe comprend les vrais positifs (TP). Ce collectif illustre les prévisions faites dans une classe, qui appartiennent en réalité à cette classe. Le groupe deux sont les vrais négatifs (TN). Il illustre les prévisions observées comme n'appartenant pas à une classe, qui ne font en réalité pas partie de cette classe. Le troisième groupe comprend les faux positifs (FP) observations prévues comme appartenant à une catégorie, mais qui ne relèvent en réalité pas de celle-ci classe. Enfin, les erreurs de type faux négatifs (FN). Ce groupe montre les nombres des observations anticipées comme ne pas appartenir à une classe, mais en vérité fait partie de cette classe. Ces résultats sont clairement affichés dans une matrice de confusion. C'est une œuvre picturale qui permet d'observer les performances d'un modèle de classification. La matrice d'erreur est également employée pour évaluer les indicateurs de performance d'un modèle. La figure 16 [60] montre une matrice de confusion de dimension 2 × 2 en termes abstraits.

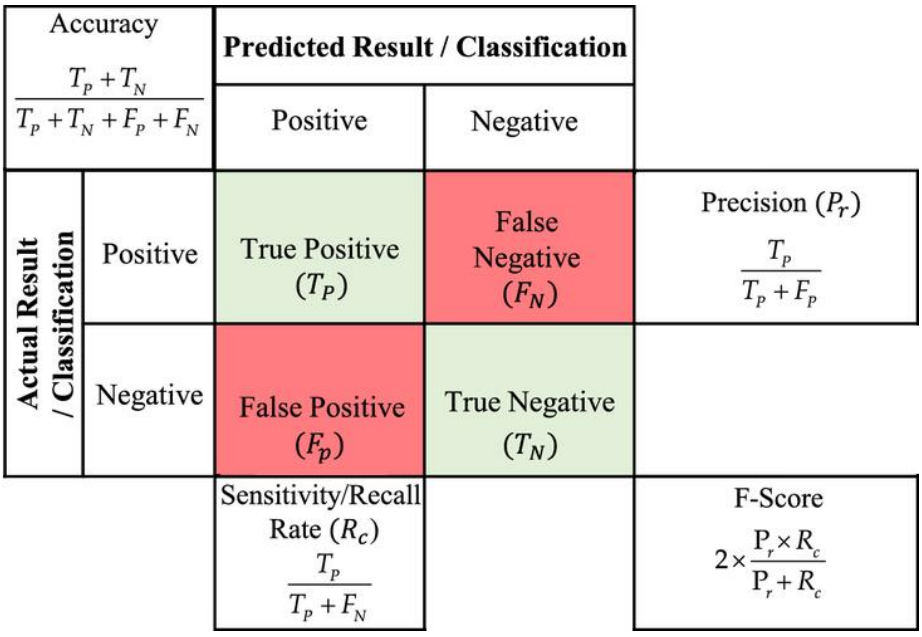


Figure 16
Matrice de confusion générale pour une classification binaire

Les métriques dans les modèles de classification en apprentissage automatique sont employées pour mesurer l'efficacité des algorithmes de classification de l'apprentissage automatique dans un contexte spécifique. Ces indicateurs de performance comprennent l'exactitude la précision, le rappel, ROC-AUC et le score « F1 ». Dans cette recherche, les quatre critères sont choisis :

- F1-score : Moyenne harmonique précision–rappel, pratique en déséquilibre de classes. Donne un seul chiffre qui pénalise les modèles déséquilibrés (ex. précision très haute mais rappel bas). Bon critère de tri “généraliste”.
- Recall (Rappel) : Parmi tous les vrais “Bad”, combien sont détectés ? Utile quand manquer une panne est très coûteux/dangereux. Haut rappel ⇒ peu de ratés.
- ROC-AUC : Aire sous la courbe ROC pour tous les seuils. Assez robuste au déséquilibre, mais peut masquer des problèmes quand les “Bad” sont très rares. À compléter par F1 ou par la courbe PR.

- Precision (Précision) : Parmi ce que le modèle marque “Bad”, combien le sont vraiment ? Utile quand le coût des fausses alertes est élevé. Haute précision ⇒ on ne déclenche rarement pour rien.
- Accuracy (exactitude) : Part de prédictions justes, Simple, mais peut être trompeuse si la classe “Bad” est rare. À utiliser surtout quand les classes sont proches en taille.

4.3 Construction des modèles de prédiction

Dans cette étape, on part de mesures vibratoires issues de trois machines CNC (M01, M02, M03), organisées par opérations OP00–OP14 et étiquetées en “good”/“Bad”, avec un fort déséquilibre en faveur du normal comme le montre la figure 17 :

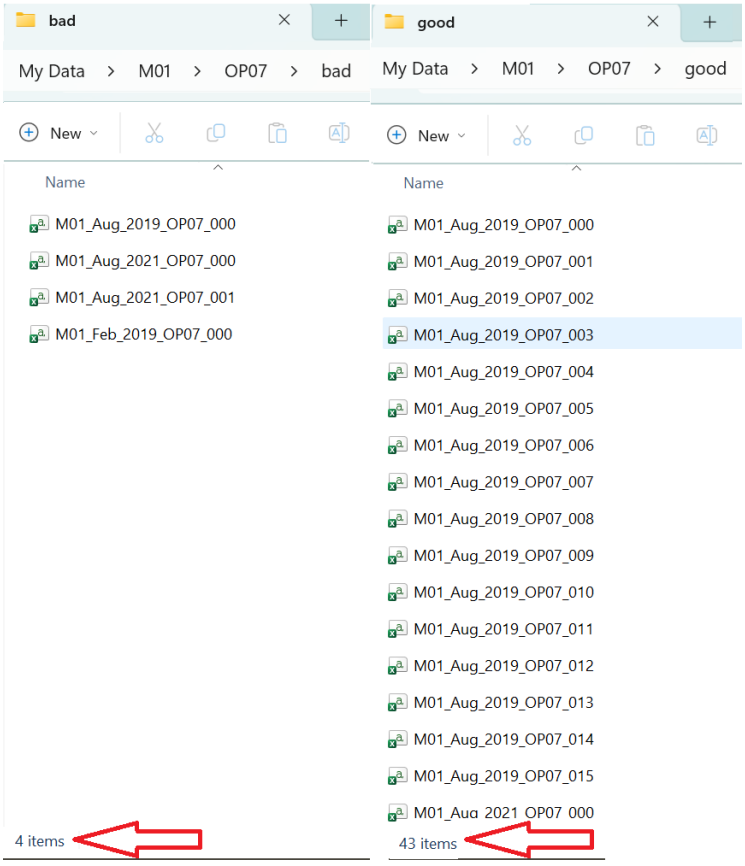


Figure 17
Exemple de minorité de données "Bad"

On segmente ensuite chaque série en fenêtres temporelles de taille W=200 échantillons (glissant configurable = 100 par défaut, non-chevauchantes possibles à H=200). Pour ce travail, on a choisi de ne pas alimenter les modèles avec chaque ligne brute du signal, mais avec des fenêtres de longueur fixe (W = 200 échantillons). D'abord, la fenêtre permet d'intégrer un contexte temporel, au lieu de regarder un instant isolé, on observe l'évolution du signal sur un court intervalle (montée en vibration, petites oscillations, micro-impacts, etc.). Or beaucoup de phénomènes liés à l'usure ou au défaut n'apparaissent pas sur un seul échantillon, mais sur une petite séquence. Ensuite, une ligne isolée est très bruitée et peu informative, un seul point peut être perturbé par un parasite (bruit électrique, micro-choc ponctuel, erreur de mesure) sans refléter l'état réel de la machine. En travaillant par fenêtres, on moyenne ce bruit et on extrait des indicateurs plus stables (RMS, kurtosis, magnitude moyenne…), qui résument le comportement global sur quelques millisecondes au lieu d'un point instantané. Le fenêtrage a aussi un effet pratique, à partir de longues séries temporelles, il génère beaucoup plus d'exemples d'apprentissage tout en réduisant la dimension brute (on passe de dizaines de milliers d'échantillons à quelques centaines de fenêtres avec 6–7 caractéristiques par fenêtre). Enfin, la taille W = 200 a été choisie comme compromis, assez longue pour contenir plusieurs périodes de vibration et rendre les descripteurs statistiques pertinents, mais suffisamment courte pour conserver une bonne résolution temporelle et rester compatible avec les contraintes de calcul et de validation croisée.

Puis on extrait par fenêtre des descripteurs compacts RMS et les kurtosis par axe et la magnitude moyenne. Pour éviter toute fuite d'information, on scinde d'abord en apprentissage/test (80/20, stratifié). Le StandardScaler est ajusté uniquement sur le train, et l'équilibrage est appliqué uniquement au train via ADASYN, qui génère des exemples synthétiques près des zones difficiles comme le montre la figure 18 :

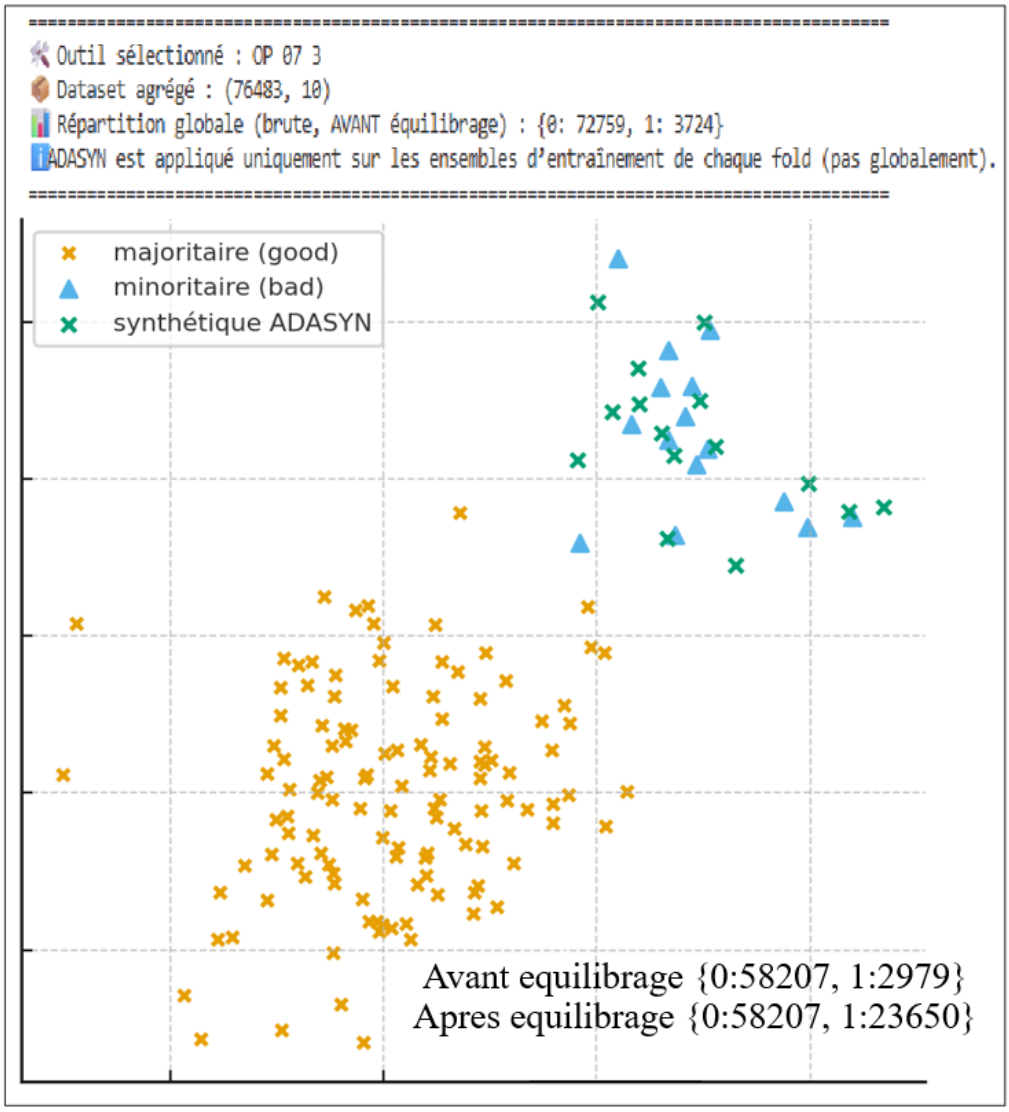


Figure 18
Rééchantillonnage des données avec ADASYN

La mise au point des modèles passe par une recherche d'hyperparamètres intégrée sur ce train standardisé et rééchantillonné. En effet, la majorité des modèles d'apprentissage automatique doivent être adéquatement paramétrés pour donner les meilleurs résultats. Plutôt qu'une grille exhaustive, on a utilisé une recherche aléatoire (Random Search) multi-modèles (scikit-learn, ParameterSampler) qui échantillonne des combinaisons pertinentes pour RF, XGBoost, KNN, arbre de décision et

régression logistique. Comme les meilleurs hyperparamètres de modèle Random Forest : (“n estimators = 400”, “min samples split = 5”, “min samples leaf = 5”, “min impurity decrease = 0.0005”, “max features = 0.8”, “max depth = 14”, “class weight = None”, “bootstrap = True”).

Pour rendre l’évaluation vraiment robuste et éviter qu’un simple découpage chanceux du jeu fausse la lecture on adopte une validation croisée stratifiée à K = 5. Concrètement, on découpe les données en cinq plis qui conservent le ratio good/bad, à chaque tour, un pli sert de test tenu à l’écart, les quatre autres forment l’entraînement, comme le montre la figure 19 :

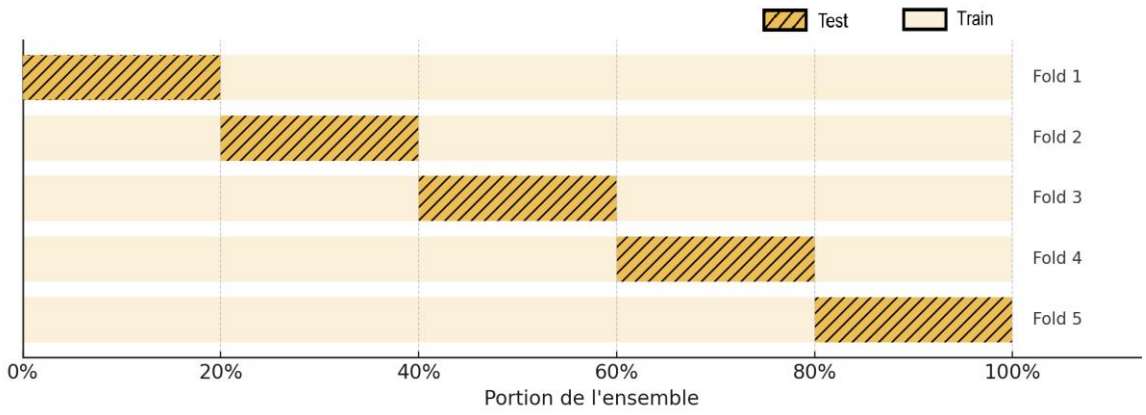


Figure 19
Validation croisée stratifiée (K=5)

Juste après cette évaluation, on lit l’ajustement via $\Delta F1_k = F1_{\text{train},k} - F1_{\text{test},k}$ pour chaque pli k. Si :

- ΔF1 proche de 0 → fit correct.
- ΔF1 trop positif (au-dessus de la bande) → sur-apprentissage : le modèle performe bien sur le train mais se dégrade sur le test.
- ΔF1 très négatif → sous-apprentissage : le modèle n’arrive pas à apprendre assez sur le train

On rapporte la moyenne et la dispersion des $\Delta F1_k$ et on les compare à une petite bande de tolérance $0{,}10, 0{,}20]$ pour juger la stabilité globale, Comme l'illustre la figure 20, on voit la dynamique de $F1_{\text{train}}$, $F1_{\text{test}}$ et des $\Delta F1_k$ dans la bande de tolérance.

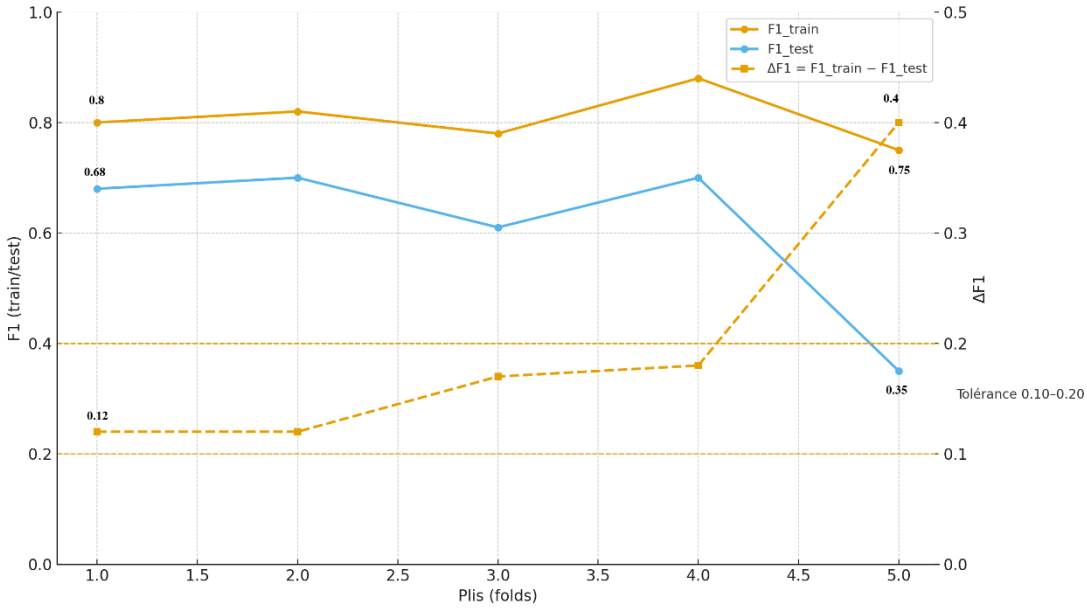


Figure 20
Lecture d'état d'apprentissage des modèles

Pour clore cette partie, on illustre sur la Figure 21 l'ensemble de la chaîne de construction des modèles, depuis les signaux bruts jusqu'au classificateur final. On y voit successivement le chargement et le fenêtrage des données, l'extraction des descripteurs temporels, l'équilibrage par ADASYN dans les plis, puis la phase de validation croisée avec recherche aléatoire d'hyperparamètres sur plusieurs familles de modèles.

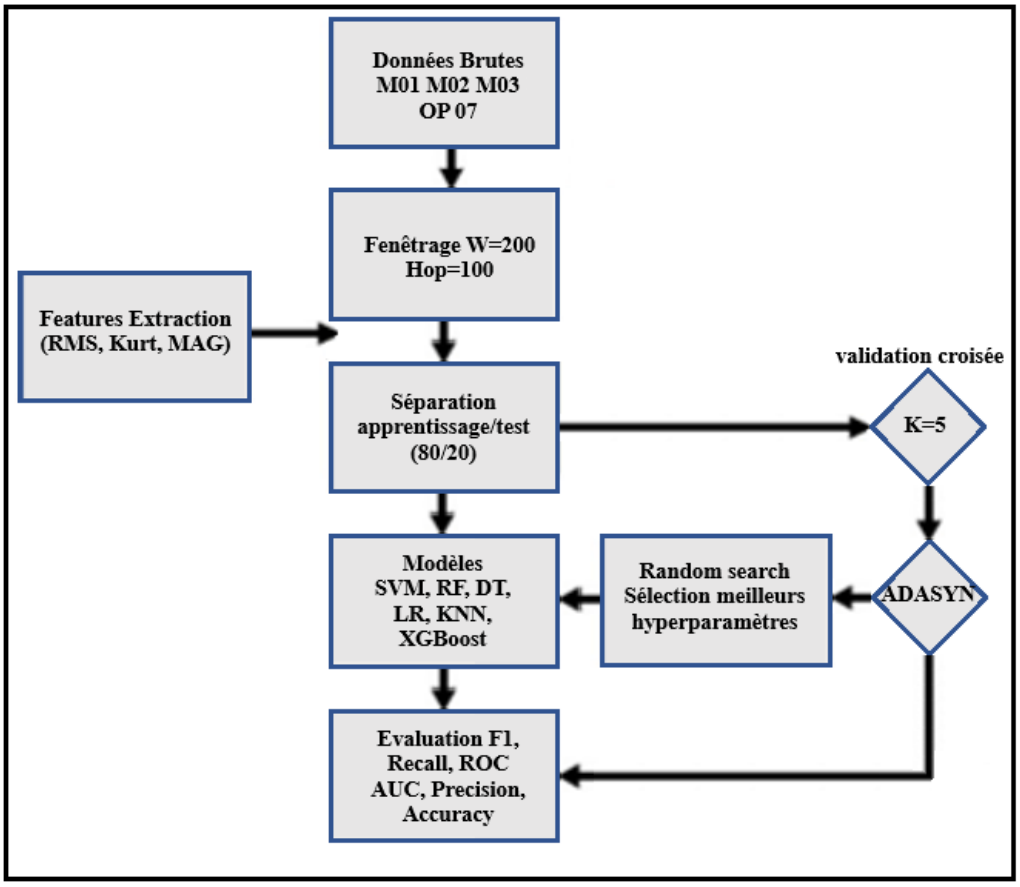


Figure 21
La chaîne de construction des modèles de prédiction

4.4 Résultats des modèles de prédiction

On présente ici les performances des classifieurs entraînés sur notre jeu vibratoire. On retient le F1 comme critère principal, complété par le rappel, l'AUC, l'exactitude, la précision, et la matrice de confusion agrégée. On commente aussi l'ajustement via ΔF1 train–test pour qualifier le fit (correct, sur- ou sous-apprentissage).

4.4.1 Matrice de confusion

La matrice de confusion condense ce que le modèle prédit face à la réalité, elle répartit les observations en vrais positifs/négatifs et faux positifs/négatifs, ce qui montre où il se trompe, pas seulement combien. Sur un jeu déséquilibré, elle est cruciale, un taux

élevé de FN (classe 1 = Bad non détectés) signale un risque opérationnel, alors qu'un excès de FP coûte en fausses alertes.

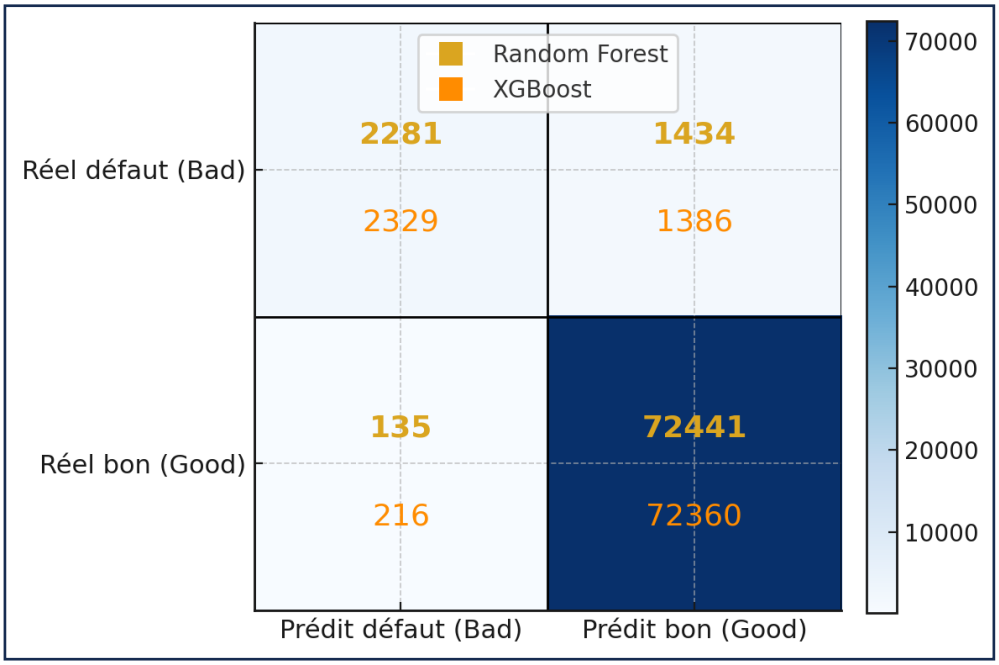


Figure 22
Matrice de confusion RF et XGBoost sans équilibrage

La figure 22 présente la matrice de confusion combinée de deux modèles sans équilibrage des données. À première vue, les métriques globales (Rappel, F1, ROC AUC) sont plutôt encourageantes, par exemple les métriques pour RF (exactitude = 0.9794, F1 = 0.7440, Précision = 0.9441, Rappel = 0.6140, ROC AUC = 0.9558) mais en regardant la matrice de confusion on voit que le modèle laisse encore passer un nombre non négligeable de défauts, c'est-à-dire des états Bad prédits comme good (FN), alors que les fausses alarmes (FP) restent relativement limitées. Or, en contexte industriel, on préfère largement déclencher trop d'alertes (des good prédits Bad) plutôt que manquer un outil réellement dégradé. C'est précisément pour corriger ce déséquilibre qu'on a recours au rééchantillonnage des données, l'objectif est de réduire les FN même si, en contrepartie, certaines métriques globales se dégradent légèrement, ce qui reste un compromis acceptable au regard du risque opérationnel.

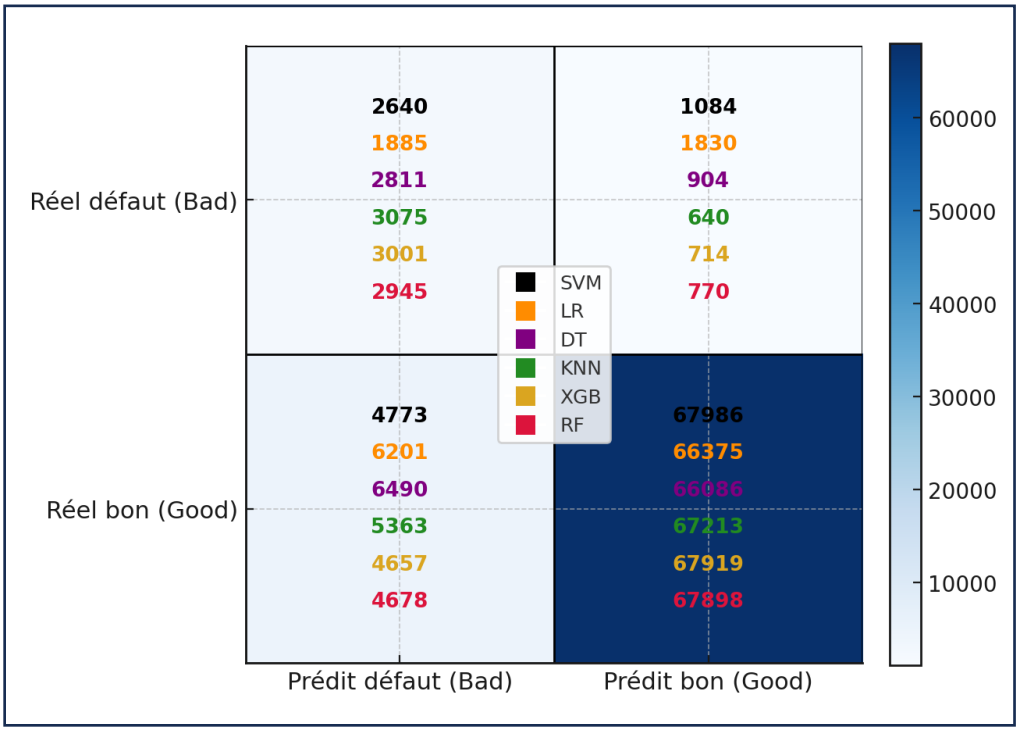


Figure 23
Matrice de confusion des modèles avec rééchantillonnage

Comme illustré la figure 23, la matrice de confusion met bien en évidence le comportement comparé des modèles, TP (réel défaut, prédit défaut), KNN, XGBoost et RF récupèrent le plus de défauts, avec respectivement ~3075, 3001 et 2945 vrais positifs, contre 2640 pour le SVM et seulement 1885 pour la régression logistique. Ce sont donc eux qui ratent le moins de défauts, comme le confirment FN (réel défaut, prédit bon) (640 pour KNN, 714 pour XGBoost, 770 pour RF, contre 1084 pour SVM et 1830 pour LR). Pour FP (réel bon, prédit défaut), on regarde cette fois les faux positifs, la régression logistique et l'arbre de décision sur-signalent les défauts (plus de 6200–6500 FP), alors que XGBoost et Random Forest restent nettement plus raisonnables (environ 4650–4700 FP), avec le SVM légèrement au-dessus et KNN un peu plus élevé (~5360). Finalement, XGBoost apparaît comme le compromis le plus équilibré, il maintient un rappel élevé sur les défauts (beaucoup de TP, peu de FN) tout en ayant le plus faible niveau de faux positifs, Random Forest étant très proche derrière, puis KNN qui pousse encore un peu le rappel au prix d'un surcroît de FP. À l'inverse, la régression logistique ressort comme le modèle le moins adapté dans ce

contexte, puisqu'il laisse passer beaucoup de défauts tout en générant de nombreuses fausses alarmes, tandis que l'arbre de décision améliore légèrement la détection des défauts mais reste très « bruyant » côté faux positifs.

4.4.2 F1-Score

Moyenne harmonique précision–rappel. Il monte seulement si les deux sont bons, et pénalise fort dès que l'un chute. Pratique en déséquilibre de classes pour avoir un seul chiffre "équilibré" de détection utile, d'où son usage comme critère principal de tri.

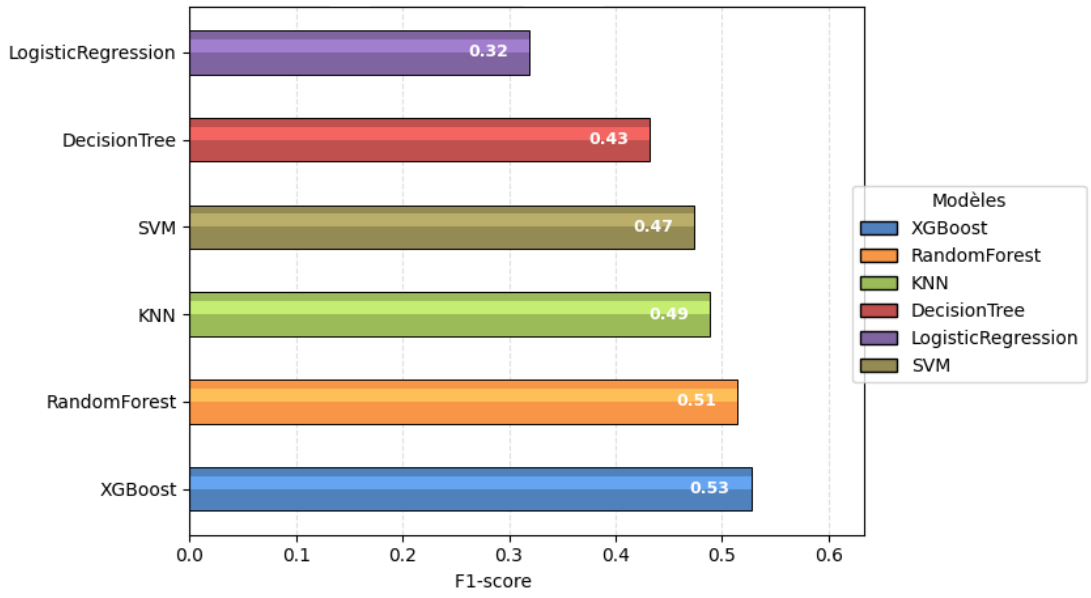


Figure 24
F1-score des modèles d'apprentissage automatique

Le graphique à barres du F1-score dans la figure 24 valide la hiérarchie précédemment repérée dans les matrices de confusion. XGBoost atteint le meilleur équilibre global avec un F1=0,53, suivi de près par la forêt aléatoire (0,51) et le KNN (0,49), qui composent le trio de modèles les plus efficaces pour cette tâche de détection de défauts. Le SVM se positionne juste après (0,47) avec des performances satisfaisantes mais un peu inférieures, tandis que l'arbre de décision simple affiche un

F1=0,43, ce qui indique une dégradation plus significative lorsque le déséquilibre des classes et la complexité des frontières de décision augmentent. Enfin, la régression logistique est clairement en retrait (0,32).

4.4.3 Le rappel

Parmi tous les vrais Bad, quelle part est détectée ? C'est la métrique critique si rater une panne coûte cher (FN). On l'améliore en abaissant le seuil ou en rééquilibrant mieux les données (ex. ADASYN), quitte à accepter plus de fausses alertes.

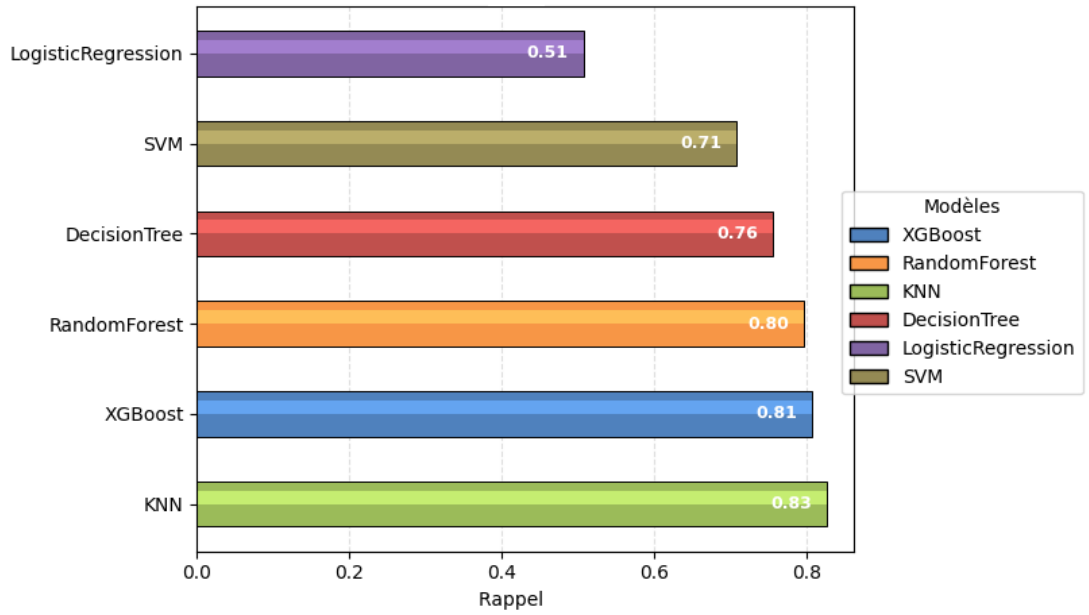


Figure 25
Rappel des modèles d'apprentissage automatique

Le diagramme de rappel, illustré par la figure 25, démontre de manière évidente l'aptitude des modèles à identifier les outils défectueux (classe Bad). KNN obtient le meilleur rappel avec 0,83, juste derrière XGBoost (0,81) et la forêt aléatoire (0,80), ces trois modèles présentent donc un faible nombre de défauts, ce qui est en accord avec les faibles nombres de FN relevés dans les matrices de confusion. L'arbre de décision demeure valide avec un rappel de 0,76, tandis que le SVM chute à 0,71 et la régression logistique à 0,51, indiquant qu'elle laisse passer quasiment un défaut sur

deux. Dans un environnement industriel où l'objectif majeur est d'éviter de laisser échapper un outil véritablement défectueux sans avis, cette illustration démontre que KNN, XGBoost et RF sont les plus appropriés si l'on se concentre uniquement sur le critère de rappel.

4.4.4 Le ROC AUC

Surface sous la courbe ROC, qui résume en un seul nombre la capacité du modèle à ranger les Bad avant les good sur toute la gamme des seuils. Comme elle ne dépend pas d'un seuil précis et reste robuste en cas de déséquilibre de classes, on l'utilise comme indicateur global de "pouvoir de séparation" du classificateur.

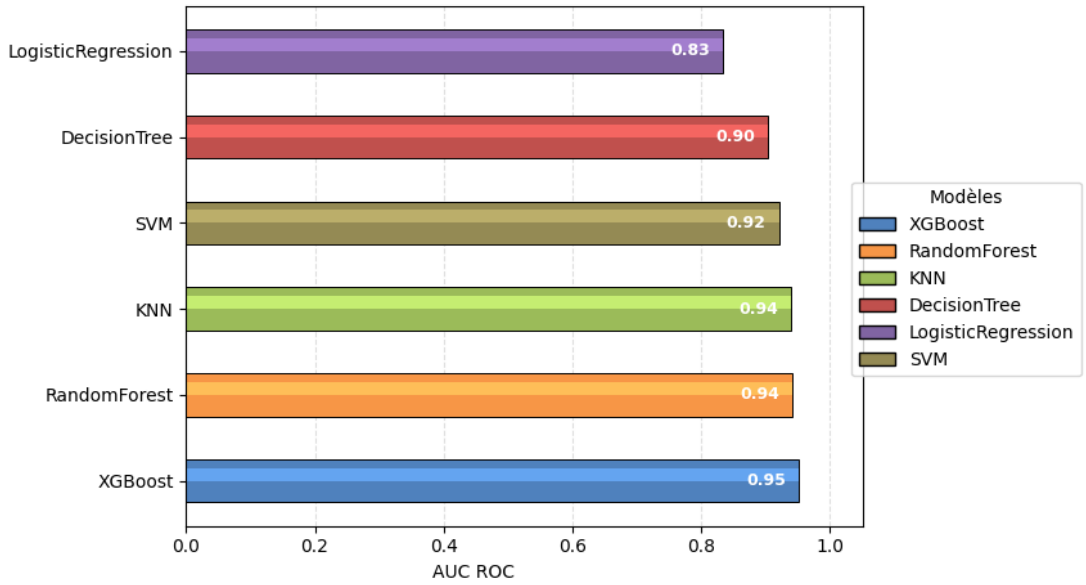


Figure 26
ROC AUC des modèles d'apprentissage automatique

Le graphique de l'AUC ROC représenté par la figure 26 illustre que tous les modèles parviennent à bien distinguer les classes, mais avec des marges variées, XGBoost se classe en premier avec une AUC de 0,95, suivi de près par la forêt aléatoire et le KNN (0,94), puis par le SVM (0,92) et l'arbre de décision (0,90), tandis que la régression logistique est bien en retrait (0,83). Étant donné que l'AUC ROC évalue l'aptitude du modèle à bien classer les fenêtres "défaut" par rapport aux fenêtres "bon état" pour

tous les seuils possibles, des valeurs proches de 0,9 indiquent que XGBoost, RF et KNN offrent une excellente séparation statistique entre les deux états, tandis que la régression logistique a plus de difficultés à les différencier. Cette illustration complète donc efficacement les précédentes, bien que certaines métriques telles que la précision ou le F1 soient modestes en raison du déséquilibre entre les classes et du choix du seuil, le potentiel discriminant global des modèles les plus complexes demeure élevée.

4.4.5 La précision

Parmi ce que le modèle marque Bad, quelle part est vraiment Bad ? Utile quand on veut limiter les fausses alertes (FP) qui mobilisent l'équipe pour rien. On augmente souvent la précision en relevant le seuil, au prix d'un rappel plus bas.

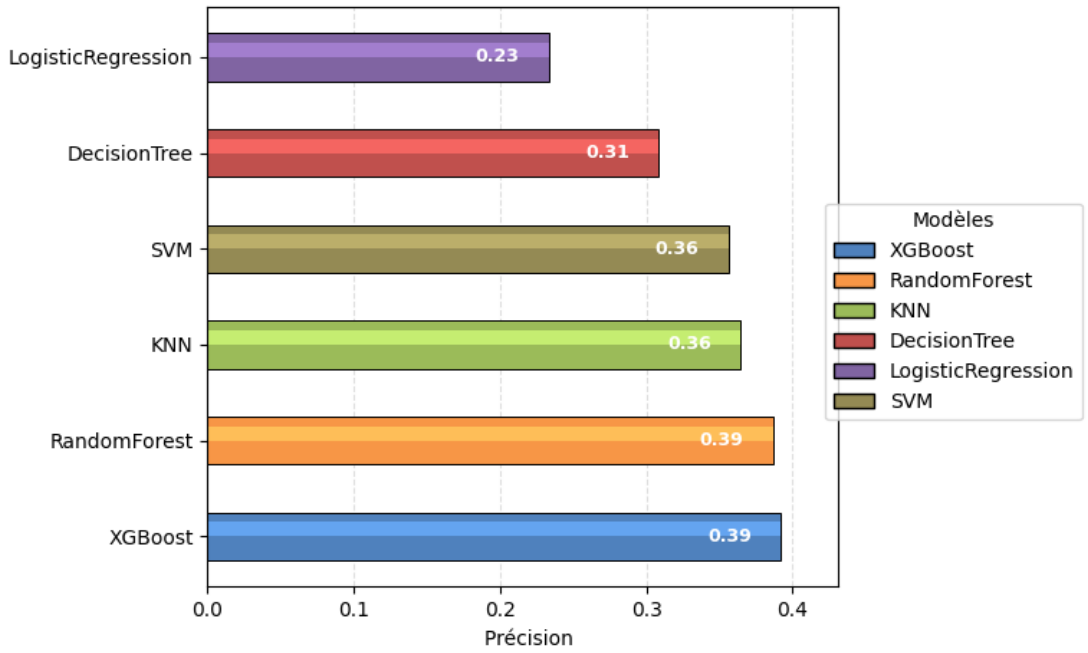


Figure 27
Précision des modèles d'apprentissage automatique

Le graphique de précision, illustré par la figure 27, souligne que malgré une bonne séparation des classes par les modèles, une part significative des alarmes demeure erronée. XGBoost et RF obtiennent la meilleure performance, avoisinant 0,39. En

pratique, sur toutes les fenêtres classées “défaut”, seulement quatre sur dix correspondent effectivement à un réel défaut, ce qui reste cependant supérieur à KNN et SVM (0,36) ainsi qu’à l’arbre de décision (0,31). La régression logistique est de nouveau en retrait, affichant une précision d'environ 0,23, indiquant qu'elle produit beaucoup plus d'alertes erronées. Ce niveau de précision globalement faible est en accord avec l'important déséquilibre des classes et avec l'optique d'un seuil davantage axé sur le rappel (éviter de rater des défauts).

4.4.6 L’exactitude

Part des prédictions correctes. C’est simple à lire, mais trompeur quand la classe Bad est rare, on peut afficher 95 % d’exactitude tout en ratant beaucoup de pannes. On l’utilise surtout quand les classes sont proches en taille et que les coûts d’erreur se valent.

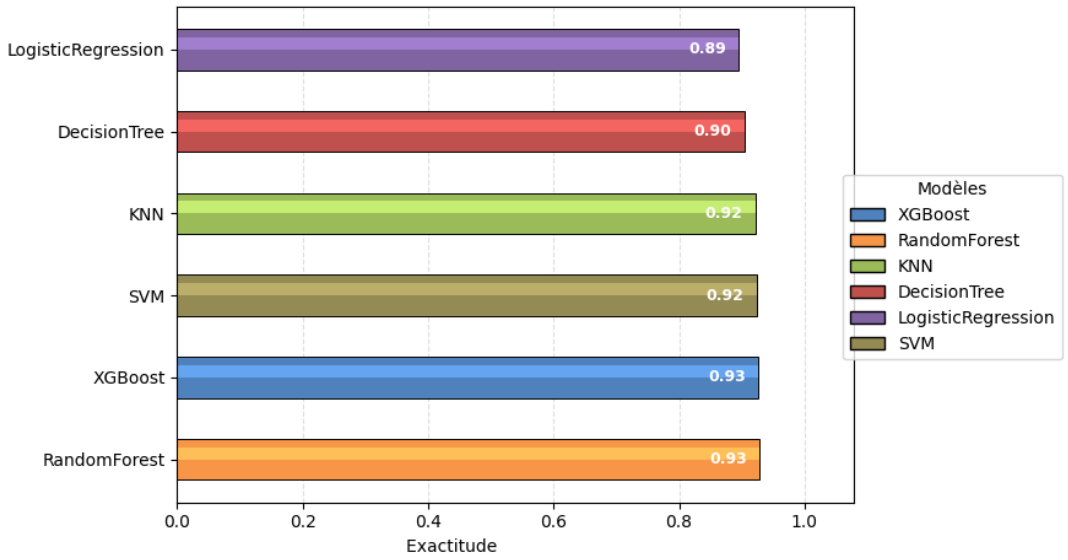


Figure 28
Exactitude des modèles d’apprentissage automatique

Le graphique d’exactitude, figure 28, montre que tous les modèles obtiennent des scores très élevés, variant entre 0,89 et 0,93. RF et XGBoost se distinguent en premier (0,93), suivis de près par KNN et SVM (0,92), puis par l'arbre de décision (0,90) et la régression logistique (0,89). Cette similarité des scores indique qu’au niveau de la

proportion totale de fenêtres correctement classées, les variations entre les modèles demeurent limitées. Cependant, dans un environnement très déséquilibré où la classe "positive" prend le dessus, une exactitude avoisinante 0,9 peut être trompeuse, un modèle peut montrer une bonne précision tout en laissant passer un nombre significatif d'erreurs ou en produisant de nombreuses fausses alertes. Cette illustration confirme que l'exactitude à elle seule ne permet pas d'évaluer la performance des classifieurs, et qu'il est nécessaire de l'analyser en relation avec le rappel, la précision, le F1-score et les matrices de confusion.

4.5 Discussion

Les résultats des mesures de performance pour les classificateurs obtenus dans la section précédente sont regroupés dans le Tableau 9 (Rappelons que le SVM était le plus long de tous les modèles plus que 2 heures d'entraînement). En effet, le modèle SVM sur un ensemble de données volumineux prend comparativement plus de temps à s'entrainer.

Tableau 9
Tableau comparatif entre les différents modèles

Métrique	XGBoost	RF	KNN	SVM	DT	LR
F1 Score	0.528	0.519	0.507	0.474	0.432	0.32
Rappel	0.808	0.793	0.828	0.709	0.757	0.507
ROC_AUC	0.951	0.944	0.94	0.921	0.904	0.834
Précision	0.392	0.386	0.364	0.356	0.308	0.233
Exactitude	0.93	0.929	0.921	0.923	0.904	0.895
Statut	Fit OK	Fit OK	Fit OK	Fit OK	Fit OK	Fit OK
ΔF1	0.024	0.022	0.099	0.011	0.016	0.002
Temps(s)	21.746	319.552	126.254	7376.549	7.751	4.423

Tout d'abord, il est apparent que l'ensemble des classifieurs obtient une précision élevée (0,89–0,93), toutefois, cette métrique est peu indicative dans un contexte aussi déséquilibré. C'est principalement le duo rappel / F1-score, enrichi par l'AUC ROC, qui permet une comparaison efficace des méthodes. Le tableau montre aussi que l'audit de l'état d'apprentissage est rassurant, le ΔF1 moyen reste faible pour tous les modèles et le statut est "Fit OK", ce qui signifie que la validation croisée ne met pas

en évidence de sur-apprentissage massif malgré la recherche aléatoire et la richesse des hyperparamètres.

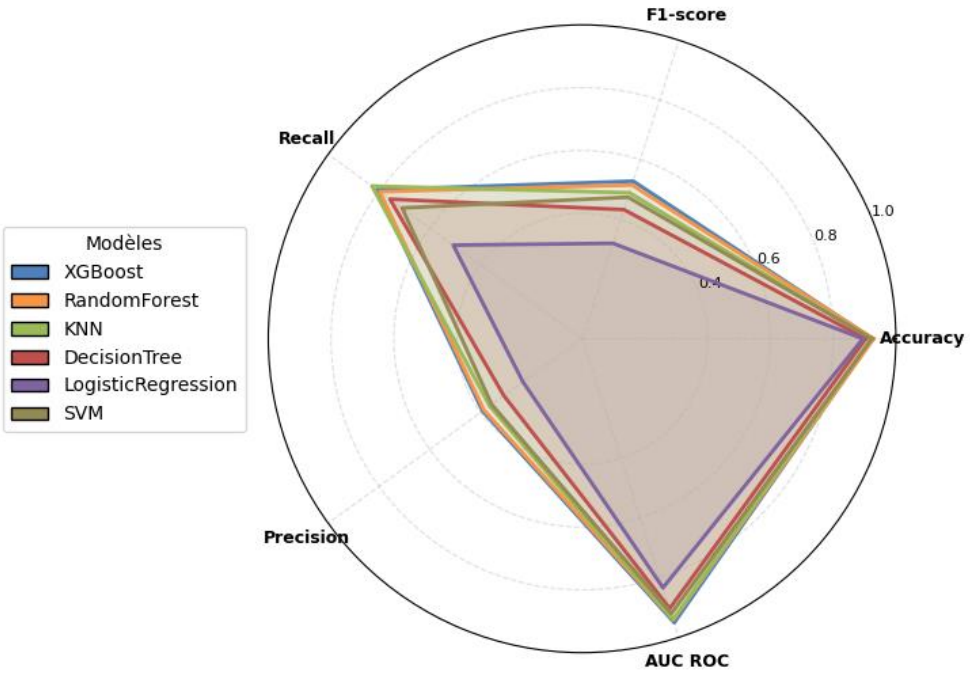


Figure 29
Radar des performances des six classificateurs testés

Sur le radar, figure 29, XGBoost apparaît comme le meilleur compromis global. Il combine le plus haut F1-score (0,53), un rappel élevé sur la classe “défaut” (0,81) et la meilleure AUC ROC (0,95), avec une précision autour de 0,39. La forêt aléatoire suit de très près avec des performances très proches sur toutes les métriques, ce qui en fait une alternative robuste et plus facilement interprétable. KNN se distingue par le rappel le plus élevé (0,83) mais au prix d’une légère dégradation de la précision et du F1-score, ce qui revient à déclencher un peu plus d’alarmes inutiles pour maximiser la détection des défauts. Le SVM et l’arbre de décision se situent dans une zone intermédiaire, ils offrent un bon rappel mais une précision plus faible, ce qui se traduit par davantage de faux positifs et un F1-score plus modeste. Enfin, la régression logistique joue bien son rôle de ligne de base linéaire, son AUC ROC (0,83), son

rappel (0,51) et surtout son F1-score (0,32) montrent qu'un modèle linéaire est insuffisant pour capturer la complexité de la séparation entre états "good" et "Bad".

En résumé, en combinant lecture du tableau numérique et du radar multi-métriques, on voit que les modèles à base d'arbres, et en particulier XGBoost, offrent le meilleur équilibre entre capacité de détection des défauts (rappel élevé), qualité globale de classification (F1, AUC ROC) et stabilité de l'apprentissage (ΔF1 faible).

4.6 Conclusion

Pour conclure, ce chapitre a permis de construire et d'auditer un premier système de classification basé sur l'apprentissage supervisé classique à. Après un premier diagnostic sans équilibrage, qui donnait des métriques globales flatteuses mais des matrices de confusion moins rassurantes.

On a introduit une approche avec rééchantillonnage ADASYN sur les données d'entraînement uniquement pour six modèles (régression logistique, arbre de décision, KNN, SVM, RF et XGBoost). Les métriques (F1, rappel, précision, AUC ROC, exactitude), complétés par la matrice de confusion, montrent que les modèles offrent le meilleur compromis entre détection des défauts. Au passage, l'audit ΔF1 et la prise en compte du temps d'entraînement ont permis de vérifier que tous les modèles retenus sont en état "Fit OK", mais certains, comme le SVM, deviennent très coûteux en temps de calcul pour un gain limité par rapport aux forêts d'arbres.

Malgré ces résultats encourageants, cette approche reste fortement dépendante du choix des descripteurs, de l'ingénierie de caractéristiques et d'un seuil de décision fixe. Ces limites justifient de passer à une modélisation plus expressive, capable de travailler directement sur des séquences multicanales et de capturer des motifs dynamiques plus complexes. C'est précisément l'objet du chapitre suivant, où on va développer et évaluer des architectures de type Conv1D–LSTM, intégrées à un pipeline complet de rééchantillonnage, calibration de seuil et diagnostic du fit.

5. PRÉDICTION DU DÉFAUT EN UTILISANT L'APPRENTISSAGE PROFOND

5.1 Introduction

Ce chapitre est consacré à la prédiction des défauts en s'appuyant cette fois sur l'apprentissage profond. Après avoir exploré, dans les chapitres précédents, les approches plus classiques de la maintenance prédictive (analyse des signaux temporels et fréquentiels, extraction de descripteurs, puis utilisation de classificateurs supervisés), on cherche ici à tirer parti de modèles capables d'apprendre directement des représentations complexes à partir des signaux bruts. L'idée générale est de laisser au réseau la responsabilité d'extraire lui-même des caractéristiques pertinentes liées à l'usure et aux défauts, plutôt que de se limiter à un ensemble restreint d'indicateurs prédéfinis.

Dans la suite de ce chapitre, on présente d'abord la préparation spécifique des données pour l'apprentissage profond (segmentation, normalisation, organisation des fenêtres). On décrit ensuite l'architecture retenue, en justifiant les choix de couches et d'hyperparamètres par rapport aux contraintes du problème. Enfin, on détaille la procédure d'entraînement, les métriques d'évaluation utilisées et on discute les performances obtenues en les comparant aux méthodes d'apprentissage supervisé plus classiques étudiées précédemment.

5.2 Étapes de construction des modèles de prédiction

Dans cette partie, on décrit la chaîne de traitement adoptée pour la prédiction des défauts par apprentissage profond. Comme au chapitre précédent, les scripts sont développés en Python, mais en s'appuyant cette fois principalement sur la bibliothèque « TensorFlow/Keras » pour la construction et l'entraînement des réseaux de neurones, en complément des bibliothèques numériques et scientifiques « NumPy et SciPy ». On travaille toujours sur trois machines (M01, M02, M03), pour chaque couple machine/état (Good/Bad), l'ensemble des fichiers CSV est chargé, converti en valeurs numériques, étiqueté (0 = Good, 1 = Bad) et agrégé dans un seul jeu de données, la colonne machine étant conservée uniquement pour la traçabilité (voir Annexe G pour les scripts de préparation et de chargement des données).

Les signaux triaxiaux sont ensuite segmentés en fenêtres temporelles de taille fixe, organisées sous forme de tenseurs éventuellement enrichis par quelques descripteurs simples (par exemple magnitude ou indicateurs statistiques) afin d'alimenter correctement le réseau.

Pour analyser l'impact du déséquilibre de classes sur les performances du modèle profond, on entraîne la même architecture sur trois variantes du jeu de données. Dans un premier cas, on conserve la distribution originale sans aucun rééquilibrage (« sans équilibrage »), ce qui permet de mesurer la capacité du réseau à gérer directement une classe "Bad" minoritaire. Dans un deuxième cas, on applique ADASYN sur les représentations tabulaires associées aux fenêtres d'apprentissage, de façon à générer des exemples synthétiques supplémentaires pour la classe "Bad" dans les zones difficiles, tout en laissant intacte la base de test pour éviter toute fuite d'information. Enfin, dans un troisième cas, on utilise un générateur de données tabulaires de type CTGAN afin de produire des fenêtres synthétiques réalistes pour la classe minoritaire à partir de la distribution observée. Dans les trois configurations, on procède à une séparation apprentissage/test (80/20) stratifiée, à une normalisation adaptée des entrées, puis à un entraînement du réseau de neurones en suivant le même protocole d'évaluation que pour les modèles classiques, calcul de l'exactitude, de la Précision, du Rappel, du F1-score et du ROC-AUC, ainsi que l'analyse des matrices de confusion agrégées pour comparer de manière transparente l'apport de chaque stratégie de rééquilibrage. Toutes ces étapes sont simplifiées dans la figure 30.

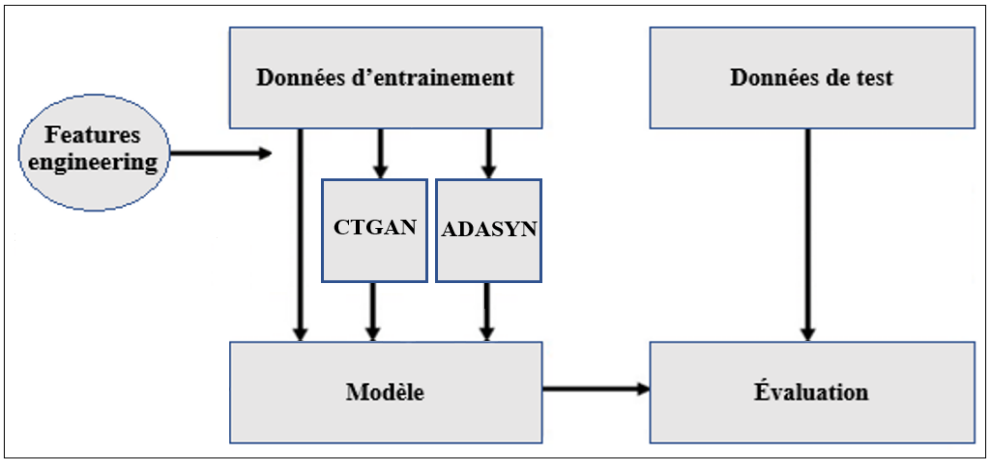


Figure 30
Pipeline global de modèle de prédiction DL

5.3 Construction des modèles de prédiction

On commence par une phase de chargement et fenêtrage aléatoire. On parcourt les mêmes dossiers (M01, M02, M03), on lit tous les fichiers dans Good et Bad, on garde les colonnes X, Y, Z (indices 0,1,2) et on construit des fenêtres de longueur « WINDOW_SIZE=200 » extraites de façon aléatoire « SEQS_PER_FILE=100 » par fichier. On refuse les couples (machine, outil) qui n'ont pas au moins un fichier good et un fichier Bad comme le montre la figure 31 :

```
INFO: Step1 (load+window) – démarrage…
WARNING: Skip M01/OP00: nécessite good & bad (good=29, bad=0)
WARNING: Skip M01/OP13: nécessite good & bad (good=43, bad=0)
WARNING: Skip M02/OP06: nécessite good & bad (good=30, bad=0)
WARNING: Skip M02/OP09: nécessite good & bad (good=43, bad=0)
WARNING: Skip M02/OP13: nécessite good & bad (good=51, bad=0)
WARNING: Skip M03/OP00: nécessite good & bad (good=28, bad=0)
WARNING: Skip M03/OP03: nécessite good & bad (good=13, bad=0)
WARNING: Skip M03/OP05: nécessite good & bad (good=38, bad=0)
WARNING: Skip M03/OP06: nécessite good & bad (good=35, bad=0)
```

Figure 31
Début de chargement et fenêtrage des données brutes

Dans tous les scénarios, on active les canaux dérivés magnitude + RMS à partir de X, Y, Z. À la fin de cette étape, on a dans les trois cas un tenseur X de taille (200, 5)

et un vecteur y (0=good, 1=Bad), plus un tableau groups qui mémorise le chemin du fichier source pour chaque fenêtre. Ensuite, les trois scripts utilisent le même principe un split train/test (80%\20%), en garantissant qu'un même fichier CSV ne se retrouve jamais à la fois dans le train et dans le test comme illustré dans la figure 32 :

```
Total windows: 121000, each window=(200 x 5) | features=['X', 'Y', 'Z', 'MAG', 'RMS']
RMS params: kind=centered, window=16, padding=edge
Step1 (load+window) — durée: 0:03:42.874577
Step2 (group split train/test) — démarrage…
Train+Val (pre-valsplit): samples=96800 | files=968 | good=91200 | bad=5600 | bad_ratio=0.0579
Test: samples=24200 | files=242 | good=22800 | bad=1400 | bad_ratio=0.0579
```

Figure 32
Nombre des fenêtres et taille des données d'entraînement\Test

La troisième étape, c'est l'équilibrage des classes, et c'est là que les scénarios divergent vraiment. Dans le premier scénario (Sans équilibrage), on ne fait aucun équilibrage à ce stade. Dans le deuxième scénario (ADASYN), on applique ADASYN uniquement sur le sous-ensemble d'entraînement interne, on rééchantillonne la classe BAD jusqu'à un ratio=0.6. Dans le script CTGAN, la logique est plus sophistiquée, on crée une trame de données avec les caractéristiques et la colonne Target, puis on isole les BAD, puis on échantillonne des BAD synthétiques jusqu'à atteindre un ratio GOOD BAD plafonné (CTGAN_RATIO_CAP_GOOD_TO_BAD=1.6:1). On les concatène avec les données réelles pour obtenir un train équilibré. La figure 33 ci-dessous représente l'architecture décrite précédemment

```
INFO: Step3 (CTGAN balance on train via PCA) — démarrage…
INFO: DF for CTGAN (pre-balance): 96800 rows, columns=1001
INFO: CTGAN(PCA): +51400 BAD synth. (final good=91200, bad=57000)
INFO: [BALANCE SUMMARY] before: good=91200, bad=5600 | after: good=91200, bad=57000
INFO: After balance: total=148200 | good=91200, bad=57000
INFO: Step3 (CTGAN balance on train via PCA) — durée: 0:24:20.780457
```

Figure 33
Équilibrage des données avec CTGAN

On passe alors à la construction du modèle Conv1D+LSTM. Les architectures sont proches, mais pas identiques.

L'algorithme sans équilibre utilise un modèle plus simple, on commence par une couche Conv1D (32 filtres, noyau 3, ReLU) immédiatement suivie d'un MaxPooling1D (2) pour réduire de moitié la longueur temporelle et concentrer l'information sur les activations les plus marquantes. Un premier Dropout (0,3) vient ensuite limiter le sur-apprentissage en éteignant aléatoirement une partie des cartes de caractéristiques pendant l'entraînement. Les séquences ainsi compressées sont transmises à un LSTM de 64 unités, puis à un deuxième Dropout (0,3). Enfin, une couche Dense (32, ReLU) joue le rôle de couche cachée pleinement connectée avant la couche de sortie Dense (1, sigmoïde), qui fournit une probabilité associée à l'état de défaut de la machine. La figure 34 illustre l'architecture dont on parle :

Layer (type)	Output Shape	Param #
conv1d (Conv1D)	(None, 198, 32)	512
max_pooling1d (MaxPooling1D)	(None, 99, 32)	0
dropout (Dropout)	(None, 99, 32)	0
lstm (LSTM)	(None, 64)	24,832
dropout_1 (Dropout)	(None, 64)	0
dense (Dense)	(None, 32)	2,080
dense_1 (Dense)	(None, 1)	33

Figure 34
L'architecture du modèle sans équilibrage

Le script ADASYN utilise une architecture, figure 35, encore plus "musclée" avec régularisation. L'architecture proposée commence par deux blocs convolutionnels 1D composés chacun d'une couche Conv1D à 128 filtres avec régularisation L2, suivie de son activation non linéaire appliquée séparément, puis d'un MaxPooling1D qui réduit la longueur temporelle et d'un SpatialDropout1D qui éteint aléatoirement des cartes complètes de caractéristiques pour limiter le sur-apprentissage. Ces cartes résumées alimentent ensuite un LSTM à 160 unités, dont les poids récurrents sont également régularisés afin de stabiliser l'apprentissage des dépendances temporelles. On ajoute un Dropout (0,2) sur les sorties du LSTM pour renforcer la robustesse du modèle, avant de passer dans une couche Dense de 64 neurones

(ReLU) avec régularisation L2, qui joue le rôle de couche cachée pleinement connectée. Enfin, la couche de sortie Dense (1, sigmoïde) fournit une probabilité associée à la classe « défaut ».

Layer (type)	Output Shape	Param #
conv1d (Conv1D)	(None, 200, 128)	3,328
activation (Activation)	(None, 200, 128)	0
max_pooling1d (MaxPooling1D)	(None, 100, 128)	0
conv1d_1 (Conv1D)	(None, 100, 128)	49,280
activation_1 (Activation)	(None, 100, 128)	0
spatial_dropout1d (SpatialDropout1D)	(None, 100, 128)	0
lstm (LSTM)	(None, 160)	184,960
dropout (Dropout)	(None, 160)	0
dense (Dense)	(None, 64)	10,304
dense_1 (Dense)	(None, 1)	65

Figure 35
L’architecture du modèle avec ADASYN

Le script CTGAN suit la chaîne suivante, figure 36, une couche Conv1D (48 filtres, noyau 3, ReLU) extrait des motifs locaux des signaux de vibration, puis une Batch Normalization stabilise les activations sur les 48 cartes de caractéristiques. Un MaxPooling1D de facteur 2 réduit ensuite la longueur temporelle tout en conservant les activations les plus significatives, avant un premier Dropout (taux 0,25) qui limite le sur-apprentissage. La couche LSTM à 96 unités modélise les dépendances temporelles plus longues entre les fenêtres, suivie d’un second Dropout (0,25) pour renforcer la régularisation. Enfin, une couche Dense de 48 neurones (ReLU) projette ces informations dans un espace de caractéristiques plus compact et la couche de sortie Dense (1 neurone, sigmoïde) fournit une probabilité associée à l’état de fonctionnement de la machine (good/Bad).

Layer (type)	Output Shape	Param #
conv1d (Conv1D)	(None, 198, 48)	768
batch_normalization (BatchNormalization)	(None, 198, 48)	192
max_pooling1d (MaxPooling1D)	(None, 99, 48)	0
dropout (Dropout)	(None, 99, 48)	0
lstm (LSTM)	(None, 96)	55,680
dropout_1 (Dropout)	(None, 96)	0
dense (Dense)	(None, 48)	4,656
dense_1 (Dense)	(None, 1)	49

Figure 36
L'architecture du modèle avec CTGAN

La phase d'entraînement principal suit ensuite, avec des stratégies différentes. Enfin, tous les scripts terminent par une évaluation sur le set de test en appliquant tel quel le seuil calibré sur validation. On calcule la matrice de confusion (TN, FP, FN, TP), la précision, le rappel, le F1, le FPR, l'exactitude et l'AUC ROC.

Pour terminer cette section, la Figure 37 synthétise l'ensemble de la chaîne de construction des modèles, depuis les signaux bruts jusqu'au classificateur final. On y distingue le chargement et le découpage en fenêtres, l'extraction des descripteurs temporels, l'équilibrage des classes, puis l'entraînement des modèles enfin l'évaluation de ces derniers.

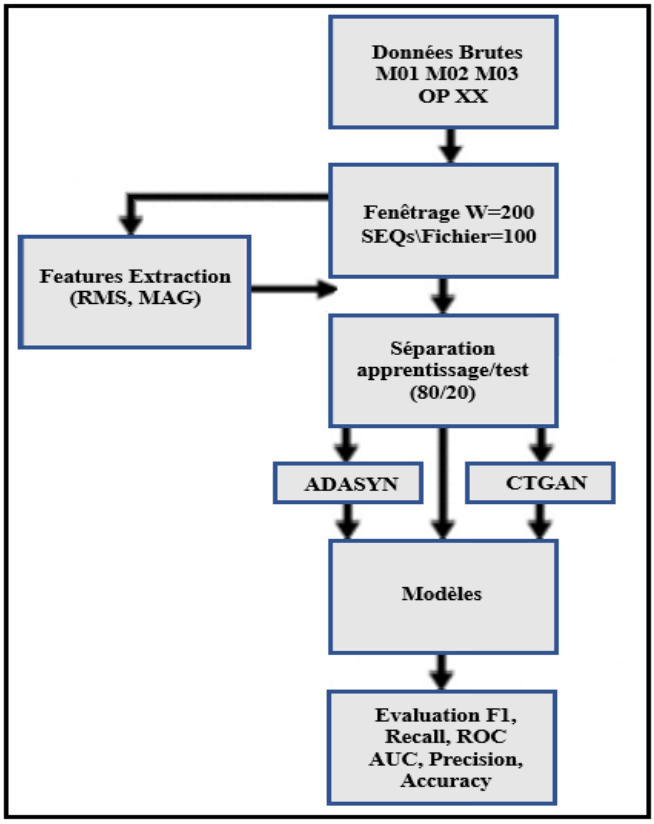


Figure 37
La chaine de construction des modèles d'apprentissage profond

5.4 Résultats des modèles de prédiction

5.4.1 Matrice de confusion

Pour analyser les performances des trois modèles, on commence par examiner les matrices de confusion, qui résument pour chaque script le nombre de fenêtres correctement et incorrectement classées entre les états good et Bad. Cet outil permet de distinguer clairement les faux négatifs (défauts manqués) des faux positifs (faux défauts signalés), ce qui est essentiel en maintenance prédictive où le coût d'un défaut non détecté n'est pas le même que celui d'une fausse alarme. Dans la suite, on présente pour chaque variante (Sans équilibre, ADASYN, CTGAN) sa matrice de confusion et on compare l'équilibre obtenu entre détection des défauts et limitation des erreurs de classification d'où la figure 38 :

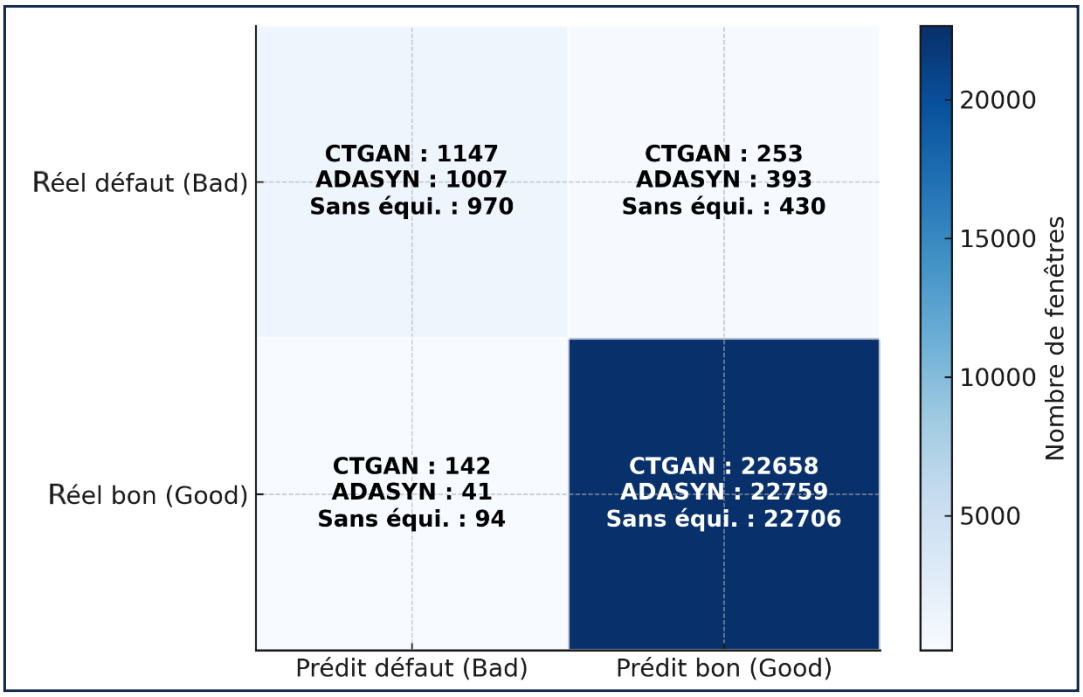


Figure 38
Matrice de confusion de 3 approches

On observe que CTGAN réduit le nombre de défauts manqués par rapport aux deux autres scripts (moins de FN), au prix d'un peu plus de fausses alertes. À l'inverse, ADASYN minimise fortement les faux positifs mais laisse passer davantage de défauts réels, ce qui traduit un choix plus conservateur sur la classe Bad. La version sans équilibrage se situe entre les deux, elle génère plus de faux positifs qu'ADASYN et plus de faux négatifs que CTGAN, ce qui montre l'intérêt d'introduire une stratégie d'équilibrage dédiée lorsque l'on cherche à piloter finement le compromis entre rappel et précision en maintenance prédictive.

5.4.2 F1-Score

La figure 39 compare les valeurs de F1-score obtenues pour les trois variantes du modèle Conv1D+LSTM (sans équilibrage, ADASYN et CTGAN) sur le jeu de test.

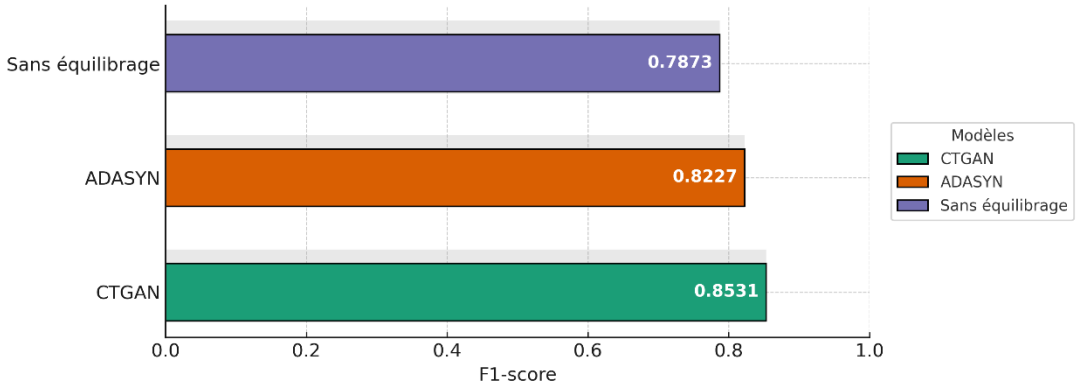


Figure 39
Comparaison du F1-Score pour les 3 versions

La figure montre que le modèle entraîné avec CTGAN obtient la meilleure performance globale, avec un F1=0,8531, devant la version ADASYN (0,8227) et le script sans équilibrage (0,7873). On voit donc que l'introduction d'un équilibrage par données synthétiques améliore nettement le compromis global précision/rappel, et que CTGAN apporte un gain supplémentaire par rapport à ADASYN. En pratique, ce résultat confirme que le modèle CTGAN est celui qui exploite le mieux les fenêtres rares de défaut tout en conservant un bon contrôle des fausses alarmes.

5.4.3 Le rappel

Pour compléter l'analyse, on représente aussi le rappel, c'est-à-dire la proportion de fenêtres réellement défectueuses correctement détectées par le modèle. La figure 40 ci-dessous compare directement les valeurs de rappel obtenues par la version sans équilibrage, la variante ADASYN et le pipeline CTGAN.

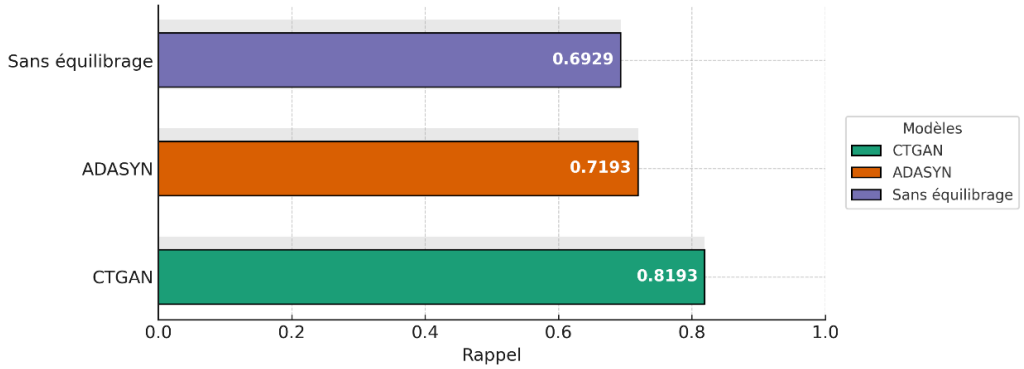


Figure 40
Comparaison du rappel des 3 modèles

On constate que le CTGAN obtient le rappel le plus élevé (0,8193), devant ADASYN (0,7193) et le modèle sans équilibrage (0,6929). Autrement dit, le pipeline CTGAN est celui qui laisse passer le moins de défauts (moins de FN), ce qui en fait la variante la plus favorable lorsqu'on cherche à maximiser la probabilité de détecter un comportement anormal, quitte à accepter un peu plus de fausses alarmes.

5.4.4 Le ROC AUC

On regarde aussi la performance globale en termes d'AUC ROC, qui mesure la capacité du modèle à bien séparer les fenêtres "bon" et "défaut" sur tout l'intervalle des seuils comme le montre la figure 41 :

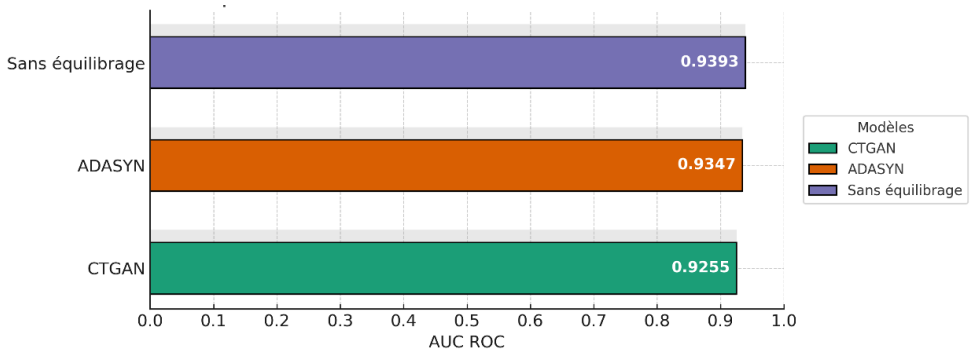


Figure 41
Comparaison ROC AUC des 3 versions

On observe ici que les trois scripts atteignent des AUC très élevés (tous > 0,92), avec un léger avantage pour le modèle sans équilibrage (0,9393), suivi d'ADASYN (0,9347) puis de CTGAN (0,9255). Cela montre que, en termes de séparation globale des classes, l'architecture de base reste déjà très robuste, et que l'introduction de CTGAN vise surtout à améliorer le rappel sur la classe rare plutôt qu'à maximiser l'AUC brute.

5.4.5 La précision

La figure 42 résume la précision obtenue par les trois variantes du modèle. Comme précédemment, on compare la version sans équilibrage, le script ADASYN et le pipeline CTGAN sur le même jeu de test. L'objectif est ici de mesurer, parmi les fenêtres prédites "défaut", quelle proportion correspond réellement à un défaut.

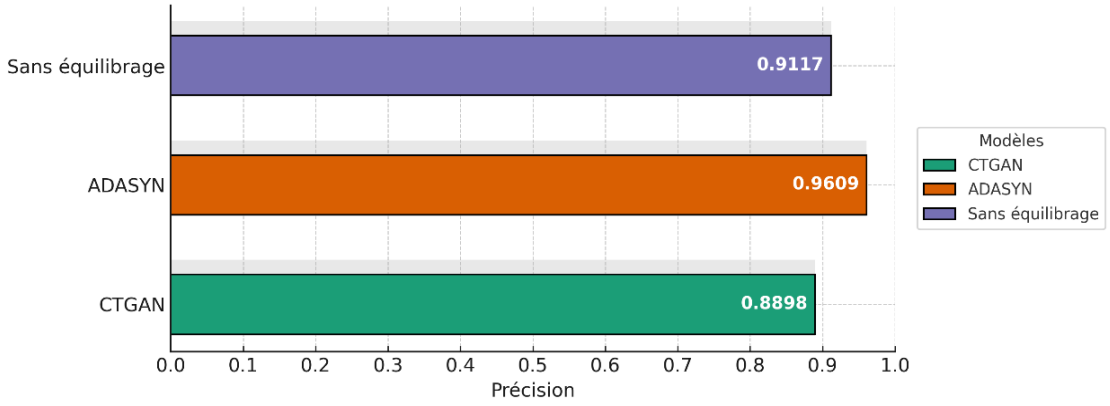


Figure 42
Comparaison de précision des 3 modèles

On observe que la variante ADASYN atteint la meilleure précision (0,9609), devant le modèle sans équilibrage (0,9117) et la version CTGAN (0,8898). Concrètement, ADASYN génère le moins de fausses alertes, mais cette amélioration en précision se fait au détriment du rappel, alors que CTGAN accepte une légère baisse de précision pour mieux capturer les fenêtres réellement défectueuses.

5.4.6 L'exactitude

La Figure 43 compare l'exactitude globale des trois variantes évaluées sur le même jeu de test. On y retrouve la version sans équilibrage, le script ADASYN et le pipeline CTGAN, toujours dans les mêmes conditions d'entraînement.

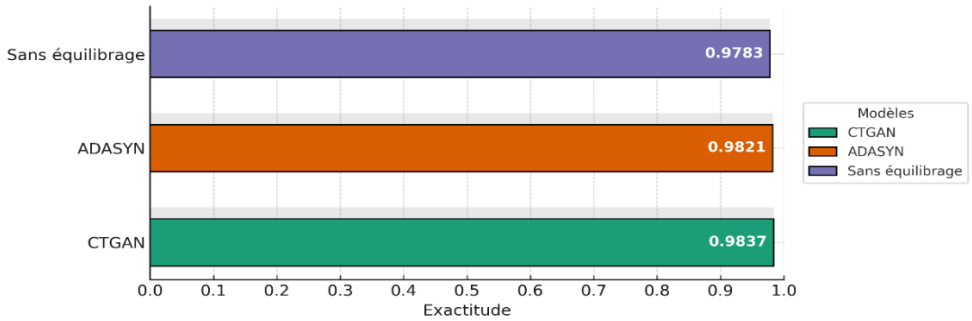


Figure 43
Comparaison d'exactitude des 3 versions du modèle

On observe que les trois configurations atteignent une exactitude très élevée, supérieure à 0,97, avec un léger avantage pour CTGAN (0,9837), suivi de ADASYN (0,9821) puis du modèle sans équilibrage (0,9783). Cela confirme que, malgré leurs différences de rappel et de précision, les trois scripts restent globalement fiables sur la classification binaire bon/défaut, et que le pipeline CTGAN parvient à améliorer le rappel sans dégrader l'exactitude globale.

5.5 Discussion

Le tableau 10 ci-dessous récapitule les principales métriques obtenues pour les trois algorithmes (sans équilibrage, ADASYN et CTGAN) sur le même jeu de test.

Tableau 10
Tableau comparatif entre les différentes versions

Métrique	CTGAN	ADASYN	Sans équilibrage
Exactitude	0.9837	0.9821	0.9783
Précision	0.8898	0.9609	0.9117
Rappel	0.8193	0.7193	0.6929
F1-Score	0.8531	0.8227	0.7873
AUC ROC	0.9255	0.9347	0.9393
Temps entraînement	10h07min39s	14h22min43	1h31min24s

En partant du tableau, on voit d'abord que les trois scripts atteignent une exactitude et une AUC très élevées, donc l'architecture Conv1D+LSTM fonctionne bien dans tous les cas. Mais dès qu'on regarde les colonnes F1-score et rappel, CTGAN prend l'avantage, il combine un F1 plus élevé et un rappel nettement supérieur, ce qui veut dire qu'on rate moins de fenêtres "Bad". ADASYN, lui, pousse très fort la précision (moins de fausses alarmes), mais au prix d'un rappel plus faible, donc plus de défauts non détectés. Le script sans équilibrage reste correct partout, mais n'est le meilleur sur aucune de ces métriques clés. Et en plus, on voit que ce gain de performance avec CTGAN et ADASYN se paie en temps d'entraînement, beaucoup plus long que pour la version sans équilibrage.

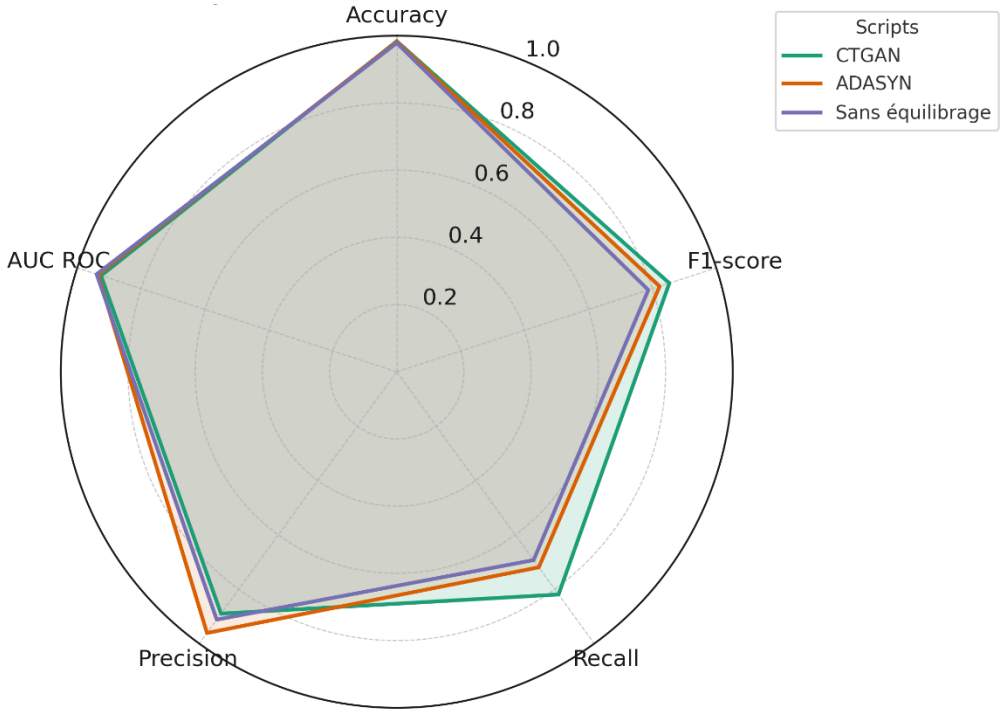


Figure 44
Radar de performance des 3 variantes de modèle

La figure 44 radar vient illustrer visuellement exactement ce constat, la courbe CTGAN enveloppe globalement les autres sur les axes rappel et F1, tout en restant très proche en exactitude et AUC, ce qui montre un comportement plus équilibré. ADASYN forme

un profil très “pointu” du côté de la précision, mais “creusé” du côté du rappel, ce qui traduit son côté plus conservateur. Le modèle sans équilibrage garde une bonne forme sur l’AUC, mais avec une surface globale plus réduite. En combinant ce que montre le tableau et ce que suggère le radar, on justifie donc le choix du script CTGAN comme configuration principale, c’est celui qui offre le meilleur compromis pratique pour la détection des défauts dans un contexte de maintenance prédictive.

5.6 Déploiement dans une maintenance prédictive

Après avoir retenu le pipeline Conv1D+LSTM avec CTGAN comme configuration de référence, on a développé une petite application dédiée pour l’exploiter en pratique. Cette interface (Comme illustré dans l’annexe H) permet de charger des séries vibratoires (au format CSV ou en saisie manuelle), d’appliquer automatiquement le même prétraitement que durant l’entraînement (fenêtrage, magnitude, RMS, normalisation) et de calculer, pour chaque fenêtre et pour chaque fichier, la probabilité de défaut associée. L’objectif est de rapprocher le modèle du terrain en offrant à l’opérateur un outil simple pour tester différents scénarios et visualiser en temps réel les décisions du classificateur dans un cadre de maintenance prédictive.

Pour illustrer le déploiement concret du modèle, la figure 45 montre l’interface de l’application lors du chargement d’un fichier de vibration réel au format CSV. L’opérateur sélectionne ici un fichier issu de la machine M01, opération OP01, dans le dossier good, puis lance la prédiction.

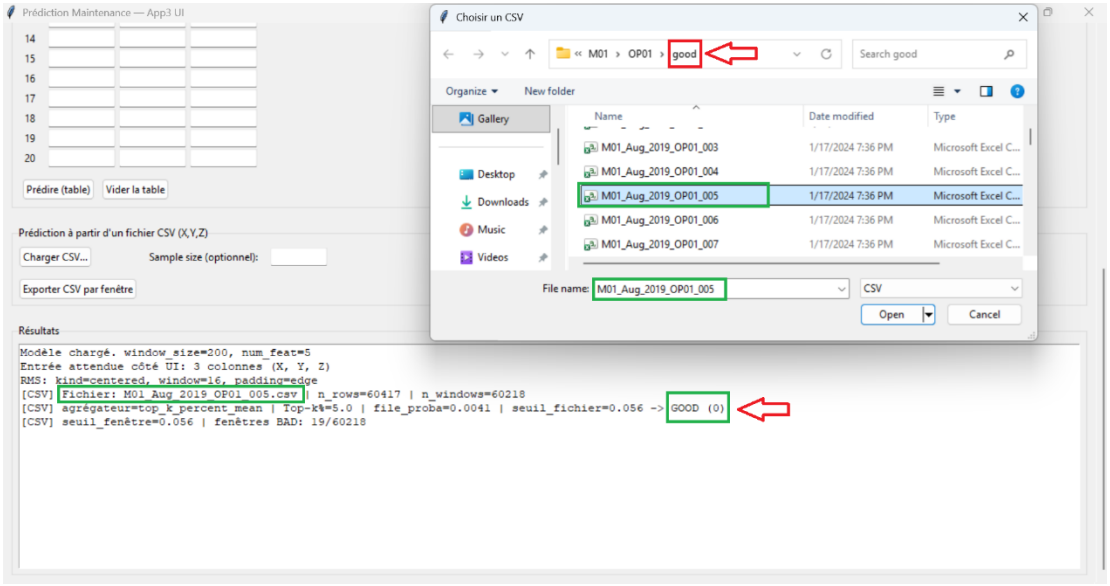


Figure 45
Prédiction du défaut (Good) à travers l'interface

On voit dans la zone « Résultats » que le fichier M01_Aug_2019_OP01_005.csv est segmenté en plusieurs fenêtres, puis agrégé en une probabilité globale de défaut (file_proba=0.0041) qui reste bien en dessous du seuil calibré (seuil_fichier=0.056). Le modèle classe donc ce cas comme GOOD (0), ce qui est cohérent avec l'étiquette attendue du dossier de départ et montre que l'application restitue correctement la décision finale au niveau fichier dans un contexte d'usage réel.

La deuxième figure (figure 46) illustre le cas symétrique, où l'opérateur sélectionne cette fois un fichier issu du dossier Bad (machine M03, opération OP10). Comme précédemment, le CSV est chargé dans l'interface, prétraité et passé au modèle CTGAN via le même agrégateur et les mêmes seuils.

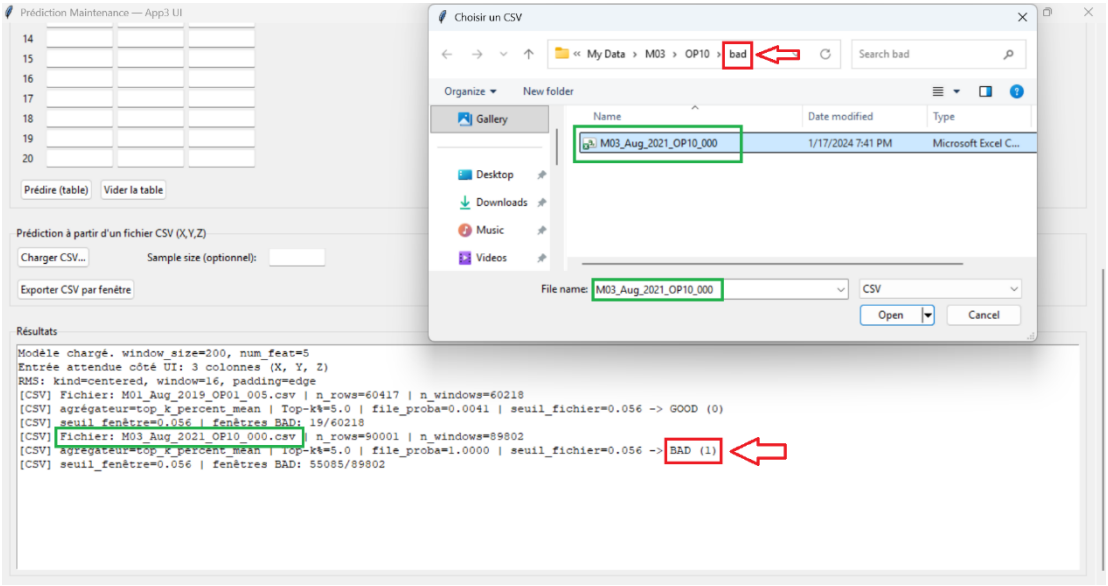


Figure 46
Prédiction du défaut (Bad) à travers l'interface

Dans la zone « Résultats », on voit que le fichier M03_Aug_2021_OP10_000.csv présente une probabilité de défaut agrégée file_proba=1.0000, très largement au-dessus du seuil seuil_fichier=0.056, avec une forte proportion de fenêtres marquées BAD. Le système classe donc clairement ce cas en BAD (1), ce qui montre que le modèle est capable d'identifier un outil dégradé de façon nette et cohérente avec l'étiquette du dossier d'origine.

5.7 Conclusion

Dans ce chapitre, on s'est intéressé à la prédiction automatique du défaut en s'appuyant cette fois sur des modèles d'apprentissage profond, plus précisément des architectures Conv1D+LSTM adaptées à l'analyse de séries temporelles vibratoires. On a d'abord rappelé les grandes étapes de construction de ces modèles. Cette chaîne de traitement garantit que le modèle voit le même type d'entrée en apprentissage et en test, tout en évitant les biais optimistes liés au mélange de fenêtres provenant d'un même fichier dans le train et le test.

Sur cette base commune, on a décliné trois scripts Conv1D+LSTM qui se distinguent par la stratégie de gestion du déséquilibre de classes et, en partie, par l'architecture réseau et le coût de calcul :

- Une version sans équilibrage, avec un réseau relativement compact.
- Une version ADASYN, doté d'une architecture plus "musclée" et fortement régularisée.
- Un script CTGAN, qui combine un réseau intermédiaire avec un équilibrage par données synthétiques CTGAN.

L'analyse des matrices de confusion a d'abord permis de vérifier que les trois modèles restent globalement cohérents, le nombre de vrais négatifs est très élevé pour toutes les variantes, ce qui traduit une bonne capacité à reconnaître les situations "bon état". En revanche, la proportion de vrais positifs et de faux négatifs varie fortement selon le script, ce qui montre que la stratégie d'équilibrage influe directement sur la capacité du modèle à détecter les fenêtres "Bad". Le modèle sans équilibrage tend à être plus conservateur et laisse passer un nombre non négligeable de défauts, alors que les versions ADASYN et surtout CTGAN détectent davantage de cas dégradés, au prix de quelques fausses alarmes supplémentaires.

Les métriques globales (F1-score, rappel, précision, AUC ROC, exactitude) confirment cette hiérarchie. Le pipeline CTGAN obtient le meilleur F1-score (0,8531) et le rappel le plus élevé (0,8193), avec une précision proche de 0,89 et une AUC de 0,9255. ADASYN atteint au contraire la meilleure précision (0,9609) mais avec un rappel plus limité (0,7193) et un F1-score intermédiaire (0,8227), ce qui reflète une stratégie orientée vers la réduction des fausses alertes plutôt que vers la maximisation de la détection. Le script sans équilibrage reste légèrement en retrait sur le F1-score (0,7873) et le rappel (0,6929), même s'il conserve une AUC très correcte (0,9393) et une exactitude proche de 0,98. La figure radar synthétise bien ce comportement, la courbe CTGAN enveloppe globalement les deux autres sur les axes rappel et F1, tout en restant très proche en exactitude et AUC, tandis qu'ADASYN présente un profil

très pointu sur la précision mais creusé sur le rappel, et que le modèle sans équilibrage affiche une surface globale plus réduite.

Un autre élément important de cette comparaison concerne le coût de calcul et la complexité des architectures. Le modèle sans équilibrage est de loin le plus rapide à entraîner (≈ 1 h 31 min), ce qui en fait une solution légère et facile à réentraîner. À l'inverse, le script ADASYN, avec son réseau plus large, demande plus de 14 heures d'entraînement, et le pipeline CTGAN environ 10 heures, à cause de ses étapes sophistiquées. On a donc un compromis clair, plus on renforce l'architecture et la stratégie d'équilibrage, plus les performances sur la classe minoritaire s'améliorent, mais au prix d'un temps de calcul nettement plus élevé.

La partie discussion a montré que, dans un contexte de maintenance prédictive où le coût d'un défaut non détecté n'est pas le même que celui d'une fausse alarme, la priorité n'est pas simplement d'optimiser l'exactitude globale, mais de trouver un équilibre acceptable entre rappel et précision, en tenant compte du coût opérationnel des erreurs. CTGAN apparaît comme le meilleur candidat lorsque l'objectif est de réduire au maximum le risque de défauts non détectés (FN) tout en gardant un taux de fausses alertes raisonnable. ADASYN reste intéressant dans un scénario où l'on souhaite minimiser les alarmes injustifiées, mais il est moins adapté si le coût d'un défaut manqué est très élevé. Le modèle sans équilibrage peut servir de base de comparaison ou de solution "rapide" pour des itérations préliminaires, mais il atteint ses limites dès qu'on cherche à traiter sérieusement la classe minoritaire.

Enfin, la partie déploiement a montré comment ce modèle peut être intégré dans une application opérationnelle. On a développé une interface capable de charger des fichiers CSV d'appliquer automatiquement le même prétraitement que lors de l'entraînement puis de calculer une probabilité de défaut par fenêtre et par fichier. Cette application montre que le pipeline n'est pas seulement un exercice de laboratoire, mais qu'on peut l'intégrer dans un outil d'aide à la décision utilisable par un opérateur.

CONCLUSION ET TRAVAUX FUTURS

Conclusion

Ce travail s'inscrit dans le cadre de l'Industrie 4.0, où la disponibilité des équipements et la maitrise de la qualité deviennent des défis essentiels pour les usines intelligentes. L'objectif était de concevoir, d'analyser et d'évaluer une chaîne complète de maintenance prédictive pour l'usure des outils de coupe en fraisage CNC, en exploitant des signaux vibratoires tri-axiaux issus de capteurs MEMS Bosch CISS installés sur trois machines (M01, M02, M03) et quinze processus d'usinage. L'approche suggérée fusionne l'analyse vibratoire traditionnelle, divers Algorithmes de machine Learning et des modèles d'apprentissage profond, jusqu'à l'intégration dans une application de test centrée sur l'opérateur.

La première partie a établi le cadre théorique général en revenant sur l'évolution vers l'Industrie 4.0, les diverses typologies de maintenance et le rôle particulier de la maintenance prédictive dans les usines intelligentes. On a démontré que, parmi les différentes techniques de surveillance de l'état des machines (analyse d'huile, mesures thermiques, suivi de courant, acoustique), l'analyse vibratoire occupe une place privilégiée pour surveiller l'usure des outils, surtout lorsqu'elle est associée à des approches temps-fréquence telles que les ondelettes ou la transformée de Hilbert–Huang et à des méthodes d'intelligence artificielle, qu'il s'agisse de machine Learning ou de Deep Learning. La revue de littérature concernant le Tool Condition Monitoring (TCM) a révélé deux tendances principales. D'une part, l'essor des techniques d'ensemble, comme les forêts aléatoires et XGBoost, qui constituent désormais des références très robustes. D'autre part, l'évolution rapide des architectures profondes telles que CNN, LSTM ou hybrides, axées sur des représentations temps-fréquence pour tirer parti de manière plus directe de la structure complexe des signaux de machine.

La deuxième partie a détaillé la démarche méthodologique suivie tout au long du projet. On a d'abord replacé la problématique dans son contexte industriel, avant de définir l'architecture cible de la solution et la stratégie de validation expérimentale. Le

travail s’appuie sur un jeu de données industriel réel, tri-axial, recueillies sur trois machines et quinze opérations, avec un échantillonnage à 2 kHz qui génère des données par jour. On a justifié le choix de la vibration comme variable principale à surveiller, la sélection d’accéléromètres MEMS tri-axiaux Bosch CISS, ainsi que la mise en place d’une chaîne d’acquisition structurée, capable de gérer ce volume important tout en conservant la traçabilité des mesures.

La troisième partie a montré qu’avant même d’entraîner des modèles d’IA, on peut extraire une information très riche des signaux vibratoires. Dans le domaine temporel, des indicateurs relativement simples, comme la valeur RMS, l’écart-type, la valeur maximale, la kurtosis ou le facteur de crête, révèlent déjà une montée significative du niveau vibratoire entre les états Good et Bad, avec notamment une hausse systématique de la RMS et du STD lorsque l’outil est usé. Dans le domaine fréquentiel, les densités spectrales de puissance (PSD) et les spectres FFT ont permis d’identifier des bandes particulièrement sensibles à l’usure, notamment autour de 200 Hz, puis de 600 et 800 Hz, avec des ratios en décibels qui montrent des différences significatives entre les états Good et Bad. L’analyse temps–fréquence par CWT a validé ces constatations en ajoutant une dimension essentielle, les bandes précédemment repérées restent durablement plus énergétiques en état Bad, ce qui suggère un comportement vibratoire anormal stable plutôt qu’un simple transitoire sporadique. On retient ainsi un premier résultat important, les états Good et Bad présentent des signatures vibratoires nettement distinctes, à la fois en niveau global (RMS) et sur des bandes fréquentielles ciblées, ce qui fournit une base solide pour alimenter des modèles de classification automatique.

La quatrième partie a introduit une première approche de maintenance prédictive reposant sur des algorithmes d’apprentissage supervisé “classiques”. On considère un outil donné observé sur les trois machines M01, M02 et M03. Plusieurs classifieurs sont entraînés et comparés, parmi lesquels la régression logistique, les arbres de décision, KNN, SVM, Random Forest et XGBoost. L’évaluation s’appuie sur la F1, le rappel, la précision, l’AUC ROC et l’exactitude, avec un audit du ΔF1 entre train et test pour surveiller le sur-apprentissage. Les résultats indiquent que l’exactitude globale

est élevée (environ 0,89–0,93) pour tous les modèles, mais cette métrique reste trompeuse compte tenu du déséquilibre des classes. Les méthodes à base d'arbres se distinguent, XGBoost et Random Forest offrent le meilleur compromis entre F1, rappel et AUC, avec par exemple pour XGBoost une F1 d'environ 0,53, un rappel proche de 0,81 et une AUC autour de 0,95, tandis que la RF obtient des performances très proches. KNN permet d'augmenter encore un peu le rappel, au prix de davantage de fausses alertes, et la régression logistique se révèle clairement insuffisante pour capter la complexité des signaux. Cette première approche fournit donc un classifieur robuste, interprétable et raisonnablement coûteux en calcul, qui sert de référence pour la suite du travail.

La cinquième partie s'est concentrée sur le développement de modèles profonds Conv1D+LSTM, alimentés directement par des fenêtres de signaux bruts sur les axes X, Y et Z, enrichies par la magnitude et la RMS. Trois scénarios ont été étudiés en fonction de la stratégie de gestion du déséquilibre. Un entraînement sans équilibrage sur la distribution naturelle, une variante avec ADASYN et un modèle plus large fortement régularisé, puis un pipeline complet combinant Conv1D+LSTM et sur-échantillonnage génératif via CTGAN. Les trois configurations atteignent des AUC et des exactitudes très élevées, mais elles se différencient sur la F1 et le rappel. Le pipeline CTGAN ressort comme la solution la plus équilibrée, avec une F1 autour de 0,85, un rappel proche de 0,82 et une exactitude d'environ 0,98, au prix d'un temps d'entraînement plus long, de l'ordre de dix heures. La variante ADASYN maximise la précision, qui atteint environ 0,96, mais avec un rappel plus faible, autour de 0,72, ce qui implique un risque plus important de défauts non détectés. Le modèle sans équilibrage reste correct mais en retrait sur la F1 et le rappel, même si son AUC demeure très satisfaisante. L'architecture profonde retenue a finalement été intégrée dans une application de prédiction, l'interface permet de charger des fichiers CSV ou des saisies manuelles, applique les mêmes prétraitements (fenêtrage, calcul de la magnitude et de la RMS, normalisation), puis affiche une probabilité de défaut par fenêtre et par fichier, avec une décision Good/Bad en fonction d'un seuil calibré. Les

tests réalisés sur des exemples réels Good et Bad montrent une bonne cohérence entre l'étiquette attendue et la décision du modèle.

Les deux approches ne s'opposent pas mais elles se complètent. L'apprentissage automatique classique, avec des modèles comme la Random Forest ou XGBoost, repose sur des descripteurs déjà extraits à partir des signaux. Cette approche est rapide à entraîner, relativement simple à interpréter (importance des variables, règles implicites) et très pertinente pour disposer rapidement d'un premier système de maintenance prédictive fiable avec un coût de calcul maîtrisé. L'apprentissage profond, au contraire, exploite des fenêtres quasiment brutes et apprend lui-même des représentations adaptées au problème. Le pipeline Conv1D+LSTM combiné à CTGAN permet d'atteindre des métriques globales plus élevées, notamment en F1 et en rappel, et offre une meilleure capacité à capturer la non-stationnarité et les motifs transitoires dans les signaux. Il gère également le déséquilibre de manière plus fine grâce au sur-échantillonnage génératif. En contrepartie, ces modèles sont plus lourds en entraînement et en ressources, et moins transparents du point de vue de l'interprétation. On peut ainsi considérer les classifieurs classiques comme une base robuste et interprétable, et la solution Conv1D+LSTM + CTGAN comme une version "haut de gamme" davantage adaptée à un déploiement industriel lorsque l'enjeu principal est de minimiser les défauts non détectés, même si cela implique une complexité accrue.

Au terme de ce mémoire, on dispose d'une véritable chaîne complète de maintenance prédictive pour l'usure d'outils en fraisage CNC. Cette chaîne va de l'acquisition vibratoire tri-axiale en environnement réel à l'analyse multi-domaine (temps, fréquence, temps–fréquence), puis à la construction et à la comparaison de modèles de classification, qu'ils soient basés sur la machine Learning ou sur le Deep Learning, avec une gestion avancée du déséquilibre de classes. Elle se prolonge jusqu'au déploiement dans une application opérationnelle de diagnostic destinée à un opérateur. L'ensemble montre qu'il est possible, à partir de données existantes, de mettre en place une surveillance automatique de l'usure qui s'intègre dans une

stratégie de maintenance prédictive et prépare le terrain pour des usines plus intelligentes, plus réactives et plus robustes.

Perspectives de recherche et défis actuels

Une première perspective consiste à améliorer la qualité des données et à combiner plusieurs signaux. Un diagnostic reste limité par la qualité du capteur, un signal faible, saturé ou très bruité pose des problèmes, même avec des modèles profonds.

Une deuxième perspective consiste à passer du simple couple Good/Bad à une vision pronostique de l'usure. Cela implique d'introduire des niveaux d'usure intermédiaires, puis d'estimer une RUL en nombre de pièces restantes. Coupler des modèles de dégradation avec les architectures profondes développées ici permettrait de mieux planifier les changements d'outil.

Enfin, une troisième perspective est de replacer la surveillance d'usure dans une logique d'optimisation globale coût–qualité. On peut coupler l'état d'usure avec la qualité pièce et adapter les conditions de coupe en temps réel, tout en visant des solutions abordables pour les PME (capteurs low-cost, modèles allégés en Edge, briques logicielles ouvertes).

En résumé, le mémoire montre qu'à partir de signaux vibratoires existants, on peut construire un pipeline complet de maintenance prédictive jusqu'à l'application opérateur, et ouvre la voie à des travaux futurs sur de meilleurs capteurs, des modèles plus génériques, le pronostic RUL, l'intégration usine et la démocratisation des solutions.

Contributions scientifiques issues du mémoire

Les travaux réalisés dans le cadre de ce mémoire ont donné lieu à trois contributions scientifiques, qui traduisent l'évolution progressive de la recherche menée.

Une première contribution a fait l'objet d'une publication dans la revue International Journal of Innovative Technology and Exploring Engineering (IJITEE), sous le titre Generalizable Tool-Wear Monitoring in Brownfield CNC Milling via Time, Frequency, and Time-Frequency Vibration Features, auteurs : Marwen Chouk, Fouad Slaoui Hasnaoui et Abdelhak Mehadjbia, Volume 15, numéro 3, février 2026, https://doi.org/10.35940/ijitee.F8336.15030226.

Une deuxième contribution, intitulée CNC Fault Detection from Raw Vibration Signals: A Comparison of Class-Imbalance Handling Strategies in Deep Learning, auteurs : Marwen Chouk, Fouad Slaoui Hasnaoui et Abdelhak Mehadjbia, est actuellement en cours d'évaluation au Journal of Mechanical Engineering, avec URL de soumission (https://journal.uitm.edu.my/ojs/index.php/JMechE/dashboard/mySubmissions?workflowSubmission Id=10959.

Enfin, une troisième contribution, intitulée A Comparison of Class-Imbalance Handling Strategies in Machine Learning from Raw Vibration Signals, est en cours de preparation, auteurs : Marwen Chouk, Fouad Slaoui Hasnaoui et Abdelhak Mehadjbia.

Dans leur ensemble, ces travaux témoignent de la portée scientifique du mémoire et de sa contribution au domaine de la maintenance prédictive appliquée aux machines CNC, en particulier pour l'analyse des signaux vibratoires bruts et la gestion du déséquilibre de classes dans les approches d'apprentissage automatique et profond.

ANNEXE A – CAPTEUR BOSCH DATASHEET

Bosch Connected Devices and Solutions GmbH | Datasheet

Connected Industrial Sensor Solution | CISS

The Robust Acceleration, Vibration & Condition Detector

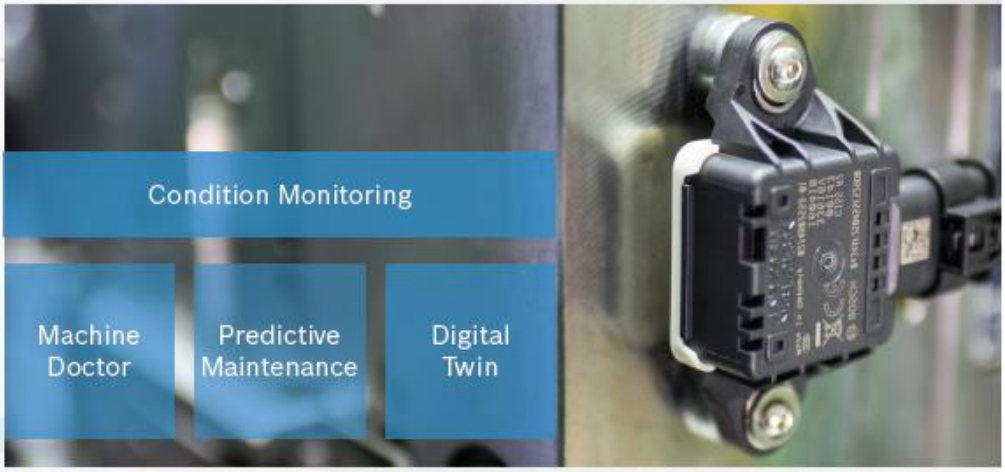
BOSCH
Invented for life

The CISS is a multi-sensor device detecting acceleration and vibration as well as environmental conditions. The robust housing and the small outline makes it perfectly suitable for industrial retrofit applications such as condition monitoring and predictive maintenance. Configuring the device enables the customer to address a broad variety of use cases by interpreting the sensor data by smart algorithms.

BUILT-IN SENSORS

Accelero-meter | Temperature sensor | Humidity sensor | Digital light sensor | Gyroscope | Magneto-meter | Pressure sensor | Micro-phone

USE CASES

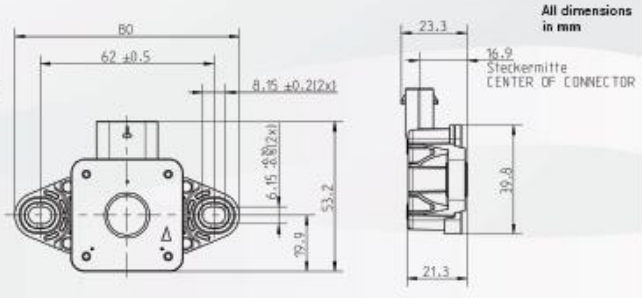


SCOPE OF DELIVERY

- CISS device with a small form factor
- USB cable (2m)
- Fasteners (2 screws, 2 washers, and 2 magnetic bases)
- Quick Start Guide

DOWNLOADS

- Example Python script
- Demo App: Virtual CISS
- Windows Driver
- Firmware Updates



MAIN COMPONENTS

- Bluetooth Low Energy (BLE) radio
- 32-Bit microcontroller (ARM Cortex M3), 1MB Flash, 128 kB RAM
- User data memory 2MB
- Accelerometer BMA280, Gyroscope BMG160, Magnetometer BMC150, Temperature sensor, Humidity sensor, Air pressure sensor BME280, Light sensor MAX44009, Microphone AKU340

BOSCH
Invented for life

OPERATING CONDITIONS

Operating temperature range	-20 °C – 80 °C
Humidity range	10 – 90 %rH (non-condensing)
Pressure range	300 – 1100 hPa
IP rating	IP54
Supply voltage	5 V DC

MEASUREMENT RANGES & ACCURACIES

Sensor	Measurement Range	Accuracy
Accelerometer	± 2, 4, 8, 16 g (14 bit resolution)	± 50 mg
Gyroscope	± 2000 °/s	± 1 °/s
Magnetometer	±1300 µT (X,Y-Axis); ±2500µT (Z-Axis)	0.06 x M ± 25µT
Temperature	-20 °C – 80 °C	max. ±2 °C + 3% T °C
Humidity	20 – 90% (non-condensing)	max. ±7% at +20 °C, max. ±10% at -20 °C
Pressure	300 - 1100 hPa	± 1.5 hPa
Light	0 - 2112800 Lux	± 15 %

SAMPLING RATE (USB)*

Inertial sensors Accelerometer, Gyroscope, Magnetometer	≤ 100 Hz
Environmental sensors Temperature sensor, Humidity sensor, Air pressure sensor, Light sensor, Microphone	≤ 1 Hz

* Inertial sensor special mode: 2kHz streaming Accelerometer (without Gyroscope, Magnetometer and environmental data)

Hints: BLE sampling rate depends on the quality of the BLE connection; Microphone data can only be transferred via BLE.

INTERFACE

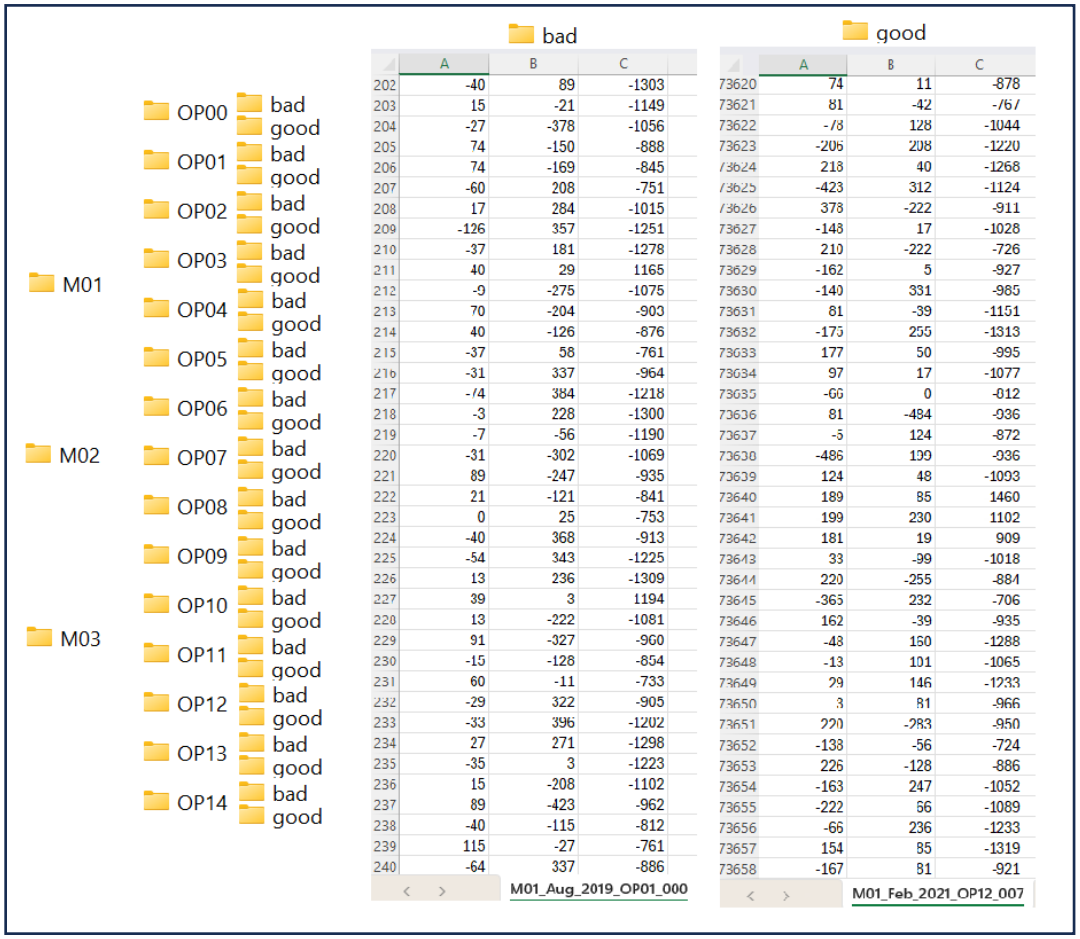
- USB communication protocol
- Bluetooth Low Energy communication protocol

GET IN CONTACT WITH US!

E-Mail: *support@bosch-connectivity.com*
Website: *www.bosch-connectivity.com/CISS*

Technical data subject to modification without notice.
© bosch-connectivity.com and solutions GmbH 2017. All rights reserved, also regarding any disposal, exploitation, reproduction, editing, distribution, as well as in the event of application for industrial property rights. July, 04, 2017

ANNEXE B – EXPLOITATION DE LA BASE DES DONNÉES



ANNEXE C – LA CONFIGURATION PYTHON UTILISÉE POUR L'ANALYSE VIBRATOIRE (PARAMÈTRE TOOL_ID)

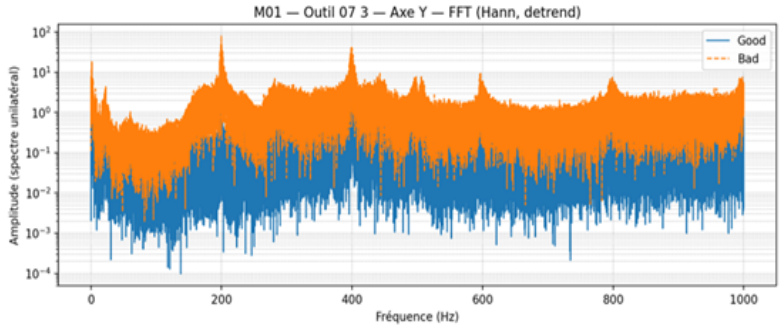
Configuration pour l'analyse temporelle

```
15  import os                  # gestion des fichiers et dossiers
16  import re                  # expressions régulières
17  import numpy as np         # calcul numérique
18  import pandas as pd        # lecture des CSV
19  import matplotlib.pyplot as plt   # affichage des courbes
20  from typing import List, Dict, Tuple   # annotations de types
21
22  tool_id = "07 3"          # identifiant de l'outil analysé
23  base_path = r"C:\Users\chouk\OneDrive\Bureau"   # dossier principal des données
24  machines = ["M01", "M02", "M03"]   # machines étudiées
25  sampling_rate = 2000      # fréquence d'échantillonnage en Hz
```

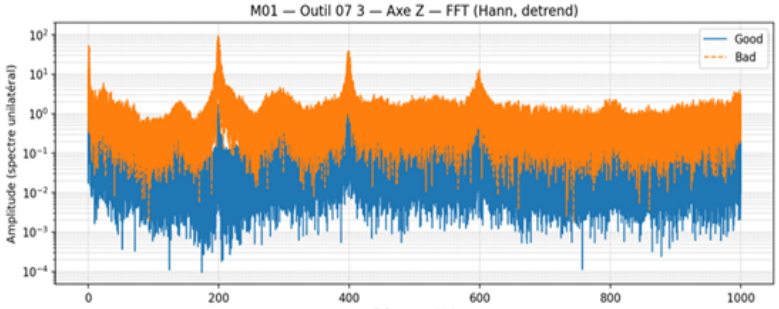
Configuration pour l'analyse fréquentielle

```
1   import os, re                          # fichiers/dossiers + expressions régulières
2   import numpy as np                     # calcul numérique
3   import pandas as pd                    # lecture des données
4   import matplotlib.pyplot as plt        # tracé des figures
5   from scipy.signal import detrend       # suppression de la tendance
6   from numpy.fft import rfft, rfftfreq   # FFT et axe fréquentiel
7   from scipy.signal import welch, find_peaks, butter, filtfilt
8
9   tool_id = "07 3"                       # identifiant de l'outil
10  base_path = r"C:\Users\chouk\OneDrive\Bureau"   # chemin des données
11  machines = ["M01", "M02", "M03"]       # machines comparées
12  sampling_rate = 2000                   # fréquence d'échantillonnage
13  fmax = 1000                            # fréquence max analysée
```

ANNEXE D – SPECTRES FFT ET MARQUEURS D'AMPLITUDE POUR LES AXES Y ET Z DE LA MACHINE M01 (OUTIL 07)

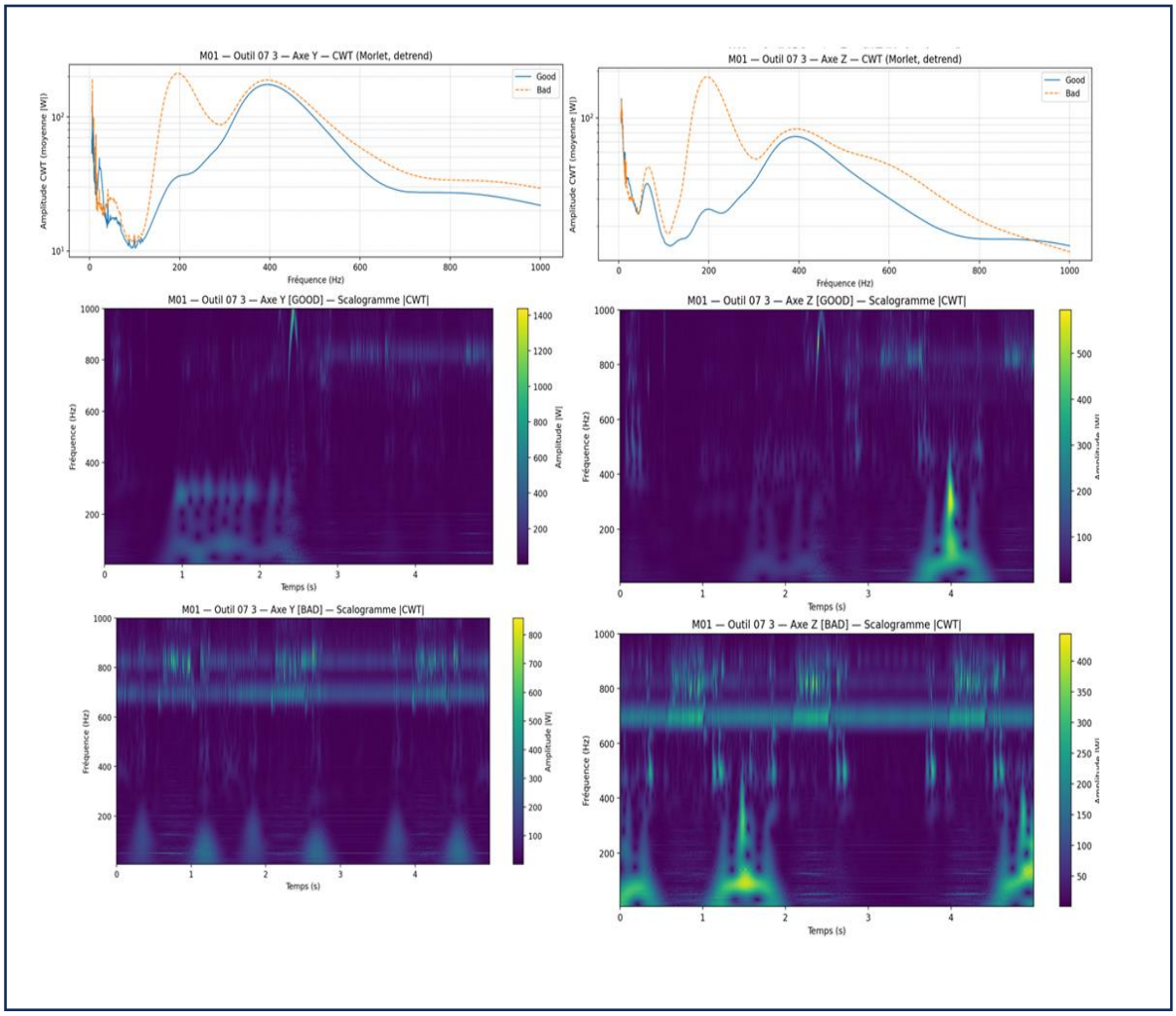


Tool	Machine	Axis	f0_hz	amp_good	amp_bad	ratio_amp_db
07	M01	Y	200	0.100036	6.53968	36.308
07	M01	Y	400	2.32173	8.26555	11.0292
07	M01	Y	600	0.239554	3.96723	24.3817
07	M01	Y	800	0.160646	2.69076	24.4801



Tool	Machine	Axis	f0_hz	amp_good	amp_bad	ratio_amp_db
07	M01	Z	200	0.118844	4.68882	31.9217
07	M01	Z	400	1.24002	9.06449	17.2783
07	M01	Z	600	0.102776	3.80468	31.3685
07	M01	Z	800	0.144149	0.478753	10.426

ANNEXE E – SPECTRES CWT ET SCALOGRAMMES POUR LES AXES Y ET Z DE LA MACHINE M01 (OUTIL 07)



ANNEXE F – BIBLIOTHÈQUE PYTHON ET CHARGEMENT DES DONNÉES POUR L'APPRENTISSAGE AUTOMATIQUE

```python
import os, re, time                  # gestion fichiers, motifs texte, temps d'exécution
from datetime import datetime        # date et heure
from collections import Counter      # comptage des classes
import numpy as np                   # calcul numérique
import pandas as pd                  # lecture des CSV
import joblib                        # sauvegarde/rechargement d'objets
from scipy.stats import kurtosis     # calcul de la kurtosis

from sklearn.model_selection import StratifiedKFold, ParameterSampler
# validation croisée stratifiée + tirage aléatoire d'hyperparamètres

from sklearn.preprocessing import StandardScaler
# normalisation des variables

from sklearn.metrics import accuracy_score, f1_score, precision_score, recall_score, roc_auc_score, confusion_matrix
# métriques d'évaluation

# Classifieurs
from sklearn.svm import SVC
from sklearn.ensemble import RandomForestClassifier
from sklearn.neighbors import KNeighborsClassifier
from sklearn.naive_bayes import GaussianNB
from sklearn.tree import DecisionTreeClassifier
from xgboost import XGBClassifier
from sklearn.linear_model import LogisticRegression

# Rééquilibrage
from imblearn.over_sampling import ADASYN
# suréchantillonnage adaptatif de la classe minoritaire

# CONFIG
BASE_PATH = r"C:\Users\chouk\OneDrive\Bureau\UQAT\Seminaire\My Data\Approche 1"  # chemin des données
MACHINES = ["M01", "M02", "M03"]   # machines étudiées
TOOL_ID = "07 3"                   # outil/opération analysé(e)
AXES_TO_USE = ("x", "y", "z")      # axes du capteur
SAMPLE_SIZE_PER_FILE = 5000        # nombre de points pris par fichier
WINDOW_SIZE = 200                  # taille d'une fenêtre
HOP_SIZE = 100                     # pas entre fenêtres (50 % de chevauchement)
APPLY_ADASYN = True                # active le rééquilibrage ADASYN
ADASYN_RATIO = 0.40                # ratio cible pour la classe minoritaire
```

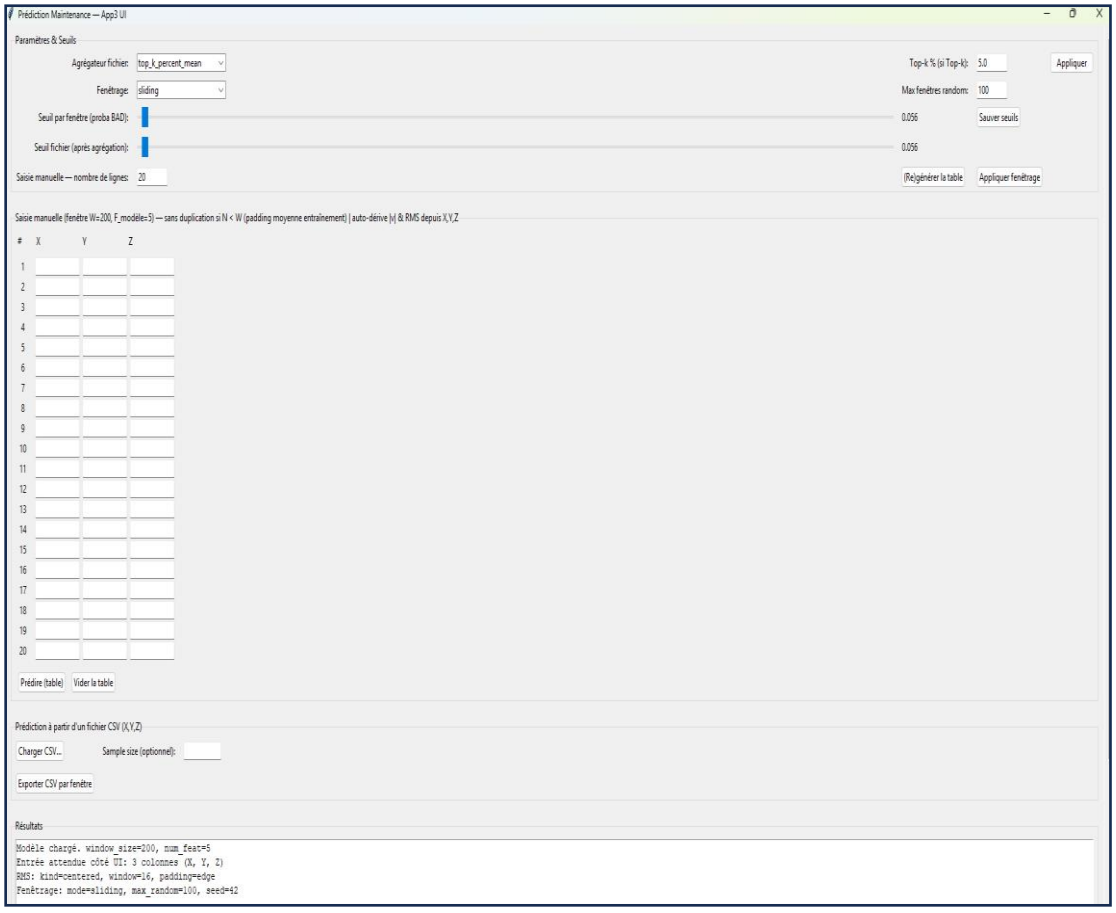
ANNEXE G – BIBLIOTHÈQUE PYTHON ET CHARGEMENT DES DONNÉES POUR L'APPRENTISSAGE PROFOND

```python
# ---------------------- Imports ----------------------
import os                    # gestion des fichiers et dossiers
import glob                  # recherche de fichiers par motif
import json                  # lecture / écriture JSON
import argparse              # gestion des arguments en ligne de commande
import datetime              # date et heure
import numpy as np           # calcul numérique
import pandas as pd          # lecture et manipulation des données
import pickle                # sauvegarde / chargement d'objets Python
import logging               # journalisation des messages
import warnings              # gestion des avertissements
from typing import List      # annotation de type liste
from contextlib import contextmanager  # création de blocs contextuels
from sklearn.model_selection import GroupShuffleSplit
# séparation train/test par groupes pour éviter la fuite de données
from sklearn.preprocessing import StandardScaler, RobustScaler
# StandardScaler : normalisation classique
# RobustScaler   : normalisation plus robuste aux valeurs extrêmes
from sklearn.utils.class_weight import compute_class_weight
# calcul des poids de classes pour compenser le déséquilibre
from sklearn.metrics import accuracy_score, roc_auc_score, confusion_matrix
# métriques d'évaluation : accuracy, ROC-AUC, matrice de confusion
from sklearn.decomposition import PCA
# réduction de dimension par analyse en composantes principales
import tensorflow as tf
# bibliothèque principale pour le deep learning
from tensorflow.keras.models import Sequential
# modèle séquentiel Keras
from tensorflow.keras.layers import Input, Conv1D, MaxPooling1D, LSTM, Dropout, Dense, BatchNormalization
# couches du réseau :
# Input            : entrée du modèle
# Conv1D           : convolution 1D sur les signaux
# MaxPooling1D     : réduction de dimension
# LSTM             : apprentissage des dépendances temporelles
# Dropout          : régularisation contre le surapprentissage
# Dense            : couche entièrement connectée
# BatchNormalization : stabilisation de l'entraînement

from sdv.single_table import CTGANSynthesizer
# générateur de données synthétiques par CTGAN
from sdv.metadata.single_table import SingleTableMetadata
# métadonnées de la table pour SDV / CTGAN
# ---------------------- DEFAULT CONFIG ----------------------
BASE_PATH   = r"C:\Users\chouk\OneDrive\Bureau\UQAT\Seminaire\My Data"  # chemin principal des données
MACHINES    = ["M01", "M02", "M03"]                                     # machines étudiées
TOOL_IDS    = [f"{i:02d}" for i in range(15)]                           # outils 00 à 14

FEATURE_COLS = [0, 1, 2]   # colonnes X, Y, Z dans les fichiers CSV
USE_DERIVED  = True        # ajout de variables dérivées : magnitude + RMS
```

ANNEXE H – L'INTERFACE DE L'APPLICATION À DÉPLOYER



LISTE DE RÉFÉRENCES

1. Lundberg, L., et al., *Bibliometric Mining of Research Trends in Machine Learning.* AI, 2024. **5**(1): p. 208-236.
2. Becker, A.S., *Evolution of deep learning trends between 2012 and 2020: A perspective from the EJR editorial board.* European journal of radiology, 2022. **155**: p. 110462.
3. Geovanni, D.-S., et al. *Methodology for Tool Wear Detection in CNC Machines Based on Fusion Flux Current of Motor and Image Workpieces*. Machines, 2023. **11**, 480 DOI: 10.3390/machines11040480.
4. Rohit, D. and S. Vijender Kumar *An IoT Based Predictive Connected Car Maintenance Approach*. 2017. **4**, 16-22 DOI: 10.9781/ijimai.2017.434.
5. Lasi, H., et al., *Industry 4.0.* Business & Information Systems Engineering : The International Journal of WIRTSCHAFTSINFORMATIK, 2014. **6**(4): p. 239-242.
6. Helbig, H.K.W.W.J., *Recommendations for implementing the strategic initiative INDUSTRIE 4.0.* 2014.
7. Qun, W., et al. *Overview of Tool Wear Monitoring Methods Based on Convolutional Neural Network*. Applied Sciences, 2021. **11**, 12041 DOI: 10.3390/app112412041.
8. Carvalho, T.P., et al., *A systematic literature review of machine learning methods applied to predictive maintenance.* Computers & Industrial Engineering, 2019. **137**.
9. Sahal, R., J.G. Breslin, and M.I. Ali, *Big data and stream processing platforms for Industry 4.0 requirements mapping for a predictive maintenance use case.* Journal of Manufacturing Systems, 2020. **54**: p. 138-151.
10. Peter, H., N. Ghani, and K. Kalegele, *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, 2017. **179**(1): p. 45-54.
11. Scaife, A.D., *Improve predictive maintenance through the application of artificial intelligence: A systematic review.* Results in Engineering, 2024. **21**.
12. Kuntoğlu, M., et al., *A Review of Indirect Tool Condition Monitoring Systems and Decision-Making Methods in Turning: Critical Analysis and Trends.* Sensors (Basel, Switzerland), 2020. **21**(1).
13. Jones, T. and Y. Cao, *Tool wear prediction based on multisensor data fusion and machine learning.* The International Journal of Advanced Manufacturing Technology, 2025. **137**(9): p. 5213-5225.
14. Bombiński, S., J. Kossakowska-Skajster, and K. Jemielniak, *New developments and future prospects in commercial tool condition monitoring systems.* Measurement, 2025. **255**.
15. Li, Z., R. Liu, and D. Wu, *Data-driven smart manufacturing: Tool wear monitoring with audio signals and machine learning.* Journal of Manufacturing Processes, 2019. **48**: p. 66-76.
16. Wang, L., et al., *Analysis and selection of eigenvalues of vibration signals in cutting tool milling.* Advances in Mechanical Engineering, 2022. **14**: p. 168781402210751.

17. Babouri, M.K. and N. Ouelaa, *Analyse vibratoire de l'usure des outils de coupe Surveillance de l'évolution de l'usure d'un outil de coupe en tournage*. 2018, Éditions universitaires européennes: Saarbrücken.
18. Peng, Z.K. and F.L. Chu, *Application of the wavelet transform in machine condition monitoring and fault diagnostics: a review with bibliography.* Mechanical Systems and Signal Processing, 2004. **18**(2): p. 199-221.
19. Xincheng, C., et al. *An Intelligent Milling Tool Wear Monitoring Methodology Based on Convolutional Neural Network with Derived Wavelet Frames Coefficient*. Applied Sciences, 2019. **9**, 3912 DOI: 10.3390/app9183912.
20. Fang, N., et al., *Wavelet-Based Denoising of Vibration Signals for Tool-Edge Wear Detection in High Speed Machining of Inconel 718.* Advanced Materials Research, 2012. **415-417**: p. 1512-1515.
21. Cao, H., Y. Lei, and Z. He, *Chatter identification in end milling process using wavelet packets and Hilbert–Huang transform.* International Journal of Machine Tools and Manufacture, 2013. **69**: p. 11-19.
22. Benkedjouh, T., N. Zerhouni, and S. Rechak, *Tool wear condition monitoring based on continuous wavelet transform and blind source separation.* The International Journal of Advanced Manufacturing Technology, 2018. **97**(9): p. 3311-3323.
23. Luo, G.Y., D. Osypiw, and M. Irle, *REAL-TIME CONDITION MONITORING BY SIGNIFICANT AND NATURAL FREQUENCIES ANALYSIS OF VIBRATION SIGNAL WITH WAVELET FILTER AND AUTOCORRELATION ENHANCEMENT.* Journal of Sound and Vibration, 2000. **236**(3): p. 413-430.
24. Hariri-Ardebili, M.A., et al., *The role of artificial intelligence and digital technologies in dam engineering: Narrative review and outlook.* Engineering Applications of Artificial Intelligence, 2023. **126**.
25. Wedel, F. and S. Marx, *Application of machine learning methods on real bridge monitoring data.* Engineering Structures, 2022. **250**.
26. Cortes, C. and V. Vapnik, *Support-vector networks.* Machine Learning, 1995. **20**(3): p. 273-297.
27. Chandola, V., A. Banerjee, and V. Kumar, *Anomaly detection: A survey.* ACM Computing Surveys, 2009. **41**(3).
28. Breiman, L., et al., *Classification and regression trees*. 2017: Chapman and Hall/CRC.
29. Breiman, L., *Random Forests.* Machine Learning, 2001. **45**(1): p. 5-32.
30. Chen, T., et al., *XGBoost: A scalable tree boosting system.* Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2016. **13-17-August-2016**: p. 785-794.
31. Hosmer, D.W., S. Lemeshow, and R.X. Sturdivant, *Applied logistic regression*. 2013, Wiley: Hoboken, New Jersey.
32. Burkhard, H., et al. *Techniques and Emerging Trends for State of the Art Equipment Maintenance Systems—A Bibliometric Analysis*. Applied Sciences, 2018. **8**, 916 DOI: 10.3390/app8060916.
33. Wu, D., et al., *A Comparative Study on Machine Learning Algorithms for Smart Manufacturing: Tool Wear Prediction Using Random Forests.* J. Manuf. Sci. Eng, 2017. **139**(7).

34. Tahir and et al., *Extracting accurate time domain features from vibration signals for reliable classification of bearing faults.* International Journal of ADVANCED AND APPLIED SCIENCES, 2018. **5**(1): p. 156-163.
35. Munaro, R., A. Attanasio, and A. Del Prete *Tool Wear Monitoring with Artificial Intelligence Methods: A Review*. Journal of Manufacturing and Materials Processing, 2023. **7**, DOI: 10.3390/jmmp7040129.
36. Si, S., D. Mu, and Z. Si, *Intelligent tool wear prediction based on deep learning PSD-CVT model.* Scientific Reports, 2024. **14**(1): p. 20754.
37. Hofmann, J., et al. *Development of Robust Machine Learning Models for Tool-Wear Monitoring in Blanking Processes Under Data Scarcity*. Applied Sciences, 2025. **15**, DOI: 10.3390/app151910323.
38. Zhang, Y., et al. *Tool Wear Condition Monitoring Method Based on Deep Learning with Force Signals*. Sensors, 2023. **23**, DOI: 10.3390/s23104595.
39. Li, X., W. Zhang, and Q. Ding, *Deep learning-based remaining useful life estimation of bearings using multi-scale feature extraction.* Reliability Engineering and System Safety, 2019. **182**: p. 208-218.
40. Wang, K., et al. *Machine Tool Wear Prediction Technology Based on Multi-Sensor Information Fusion*. Sensors, 2024. **24**, DOI: 10.3390/s24082652.
41. Rajakannu, A., et al., *Airborne acoustic emission-based intelligent wear detection for spatiotemporal data in CNC drill bits using CEEMDAN and ConvLSTM.* Sensors and Actuators: A. Physical, 2025. **396**.
42. Haibo, H., et al., *ADASYN: Adaptive synthetic sampling approach for imbalanced learning*, in *2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence)*. 2008. p. 1322-1328.
43. Jafarigol, E. and T. Trafalis, *A Review of Machine Learning Techniques in Imbalanced Data and Future Trends*. 2023.
44. Brandt, J. and E. Lanzén, *A Comparative Review of SMOTE and ADASYN in Imbalanced Data Classification*. 2021, Uppsala universitet, Statistiska institutionen.
45. Mujahid, M., et al., *Data oversampling and imbalanced datasets: an investigation of performance for machine learning and feature engineering.* Journal of Big Data, 2024. **11**(1): p. 87.
46. Han, H., W.-Y. Wang, and B.-H. Mao. *Borderline-SMOTE: A New Over-Sampling Method in Imbalanced Data Sets Learning*. in *Advances in Intelligent Computing*. 2005. Berlin, Heidelberg: Springer Berlin Heidelberg.
47. Sasada, T., et al., *A Resampling Method for Imbalanced Datasets Considering Noise and Overlap.* Procedia Computer Science, 2020. **176**: p. 420-429.
48. Xu, L., et al., *Modeling Tabular data using Conditional GAN*. 2019.
49. Eom, G. and H. Byeon *Searching for Optimal Oversampling to Process Imbalanced Data: Generative Adversarial Networks and Synthetic Minority Over-Sampling Technique*. Mathematics, 2023. **11**, DOI: 10.3390/math11163605.
50. Khoirunnisa, N. and M. Rosyda, *A comparative study on SMOTE, CTGAN, and hybrid SMOTE-CTGAN for medical data augmentation.* Science in Information Technology Letters, 2025. **6**(1): p. 44-54.

51. Priya, B.S., G. Chitra, and R. Ramalakshmi, *Performance comparison of sampling techniques with machine learning algorithms for churn prediction in telecommunication.* Franklin Open, 2025. **13**: p. 100402.
52. Learning, A.M. *Train-Synthetic-Test-Real (TSTR) Methodology* 2025; Available from: https://apxml.com/courses/evaluating-synthetic-data-quality/chapter-3-evaluating-ml-utility/tstr-methodology.
53. Khosravi, H., et al., *Strategic Data Augmentation with CTGAN for Smart Manufacturing: Enhancing Machine Learning Predictions of Paper Breaks in Pulp-and-Paper Production*. 2023.
54. Jiang, Y., B. Drescher, and G. Yuan, *A GAN-based Multi-Sensor Data Augmentation Technique for CNC Machine Tool Wear Prediction.* IEEE Access, 2023. **PP**: p. 1-1.
55. BOSCH. *Connected Industrial Sensor Solution | CISS Datasheet*. 2017; Available from: www.bosch-connectivity.com/CISS.
56. Mohamed-Ali Tnani, M.F., Klaus Diepoldb. *Dataset*. 2022; Available from: https://github.com/boschresearch/CNC_Machining.
57. Tnani, M.-A., M. Feil, and K. Diepold, *Smart Data Collection System for Brownfield CNC Milling Machines: A New Benchmark Dataset for Data-Driven Machine Monitoring.* Procedia CIRP, 2022. **107**: p. 131-136.
58. Heinzel, G., A. Rüdiger, and R. Schilling, *Spectrum and spectral density estimation by the Discrete Fourier transform (DFT), including a comprehensive list of window functions and some new at-top windows*.
59. Oppenheim, A.V., A.S. Willsky, and S.H. Nawab, *Signals & systems = Signals and systems*. 2. ed. Prentice-Hall signal processing series. 1997, London: Prentice-Hall.
60. Kaur, M., et al., *Computational deep air quality prediction techniques: a systematic review.* Artificial Intelligence Review : An International Science and Engineering Journal, 2023. **56**(Suppl 2): p. 2053-2098.