N° Ordre........../FT/UMBB/2024

PEOPLE’S DEMOCTRATIC REPUBLIC OF ALGERIA

MINISTRY OF HIGHER EDUCATION AND SCIENTIFIC RESEARCH

M’HAMED BOUGARA UNIVERSITY BOUMERDES

Faculty of Technology

Doctoral Thesis

Presented by:

MOUSSAOUI Imane

In fulfillment of a Doctorate degree in**:**

Field: Mechanical Engineering

Option: Mechatronics

Analysis of bearings defaults using machine learning techniques

Jury Members:

M.	ADJERID	Smaïl	Professor	UMBB	Chairman
M.	BENAZZOUZ	Djamel	Professor	UMBB	Thesis Director
M.	RAHMOUNE	Chemesddine	Professor	UMBB	Thesis Co-director
M	BENFDILA	Arezki	Professor	UMMTO	Examiner
M.	BENTARZI	Hamid	Professor	UMBB	Examiner
M.	GOURI	Rabah	Professor	ENST	Examiner

Academic Year: 2023/2024

Analysis of bearings defaults using machine learning techniques

Abstract

Rotating machines have become ubiquitous in contemporary industries, playing a pivotal role in various applications. The consequences of defects in these machines extend beyond mere technical issues, potentially leading to substantial economic losses and posing a significant threat to human safety. Operators often grapple with the intricacies of troubleshooting these complex systems, where a single mistake can have catastrophic consequences.

One of the most critical elements in these machines is the bearings. Consequently, numerous researchers have dedicated their time and efforts to addressing this matter.

While extensive studies have been conducted in this field, a common limitation is the focus on constant-speed scenarios. In reality, rotating machines typically operate under non-stationary conditions, making constant-speed techniques largely theoretical.

This thesis is divided into two essential parts. The first part addresses the challenges of diagnostic resolution under time-varying conditions. Given the dynamic nature of the working environment, understanding and mitigating faults in non-stationary conditions is imperative for practical applications. Our method aims to tackle the diagnostic issue under time-varying conditions. The technique was tested on a bearing database collected under time-varying conditions, containing three types of faults. Vibrational signals are initially processed using the Empirical Wavelet Transform (EWT) to extract AM-FM modes. Subsequently, a list of features is extracted from these modes. For feature selection, the Clan-Based Cultural Algorithm (CCA) is employed, and model training utilizes the Random Forest algorithm. The results demonstrate the robustness of the diagnostic process despite varying conditions.

The second part focuses on feature selection, which plays a crucial role in controlling the quality of the diagnostic system and reducing misleading factors. This area of research is increasingly attracting attention, with numerous methods developed. However, many of these techniques require in-depth domain knowledge, particularly concerning parameter tuning and result interpretation. In this work, we introduce a robust technique based on standard deviation and Random Forest methods for sequential feature selection. The method was tested on three different bearing databases, including time-varying conditions, and three signal decomposition techniques (EWT, EMD, and MODWPT). It provided promising results in terms of both quality and quantity, being user-friendly and not demanding extensive knowledge in the optimization field.

Keywords:

Rotating machines, fault classification, features selection, empirical wavelet transform, random forest, time-varying conditions, signal processing.

Analyse des défauts de roulements à l'aide des techniques d'apprentissage automatique.

Résumé

Les machines tournantes sont devenues omniprésentes dans les industries contemporaines, jouant un rôle central dans diverses applications. Les conséquences des défauts de ces machines vont au-delà des simples problèmes techniques, pouvant entraîner des pertes économiques considérables et représentant une menace importante pour la sécurité humaine. Les opérateurs se retrouvent souvent à jongler avec les complexités du dépannage de ces systèmes complexes, où une simple erreur peut avoir des conséquences catastrophiques. L'un des éléments les plus critiques dans ces machines est les roulements. Par conséquent, de nombreux chercheurs ont consacré leur temps et leurs efforts à résoudre ce problème. Bien que des études approfondies aient été menées dans ce domaine, une limitation fréquente est la focalisation sur des scénarios à vitesse constante. En réalité, les machines tournantes fonctionnent généralement dans des conditions non stationnaires, ce qui rend les techniques à vitesse constante largement théoriques.

Cette thèse est divisée en deux parties essentielles. La première partie aborde les défis de la résolution diagnostique dans des conditions variables dans le temps. Étant donné la nature dynamique de l'environnement de travail, comprendre et atténuer les défauts dans des conditions non stationnaires est impératif pour les applications pratiques. Notre méthode vise à traiter le problème diagnostique dans des conditions variables dans le temps. La technique a été testée sur une base de données de roulements collectée dans des conditions variables dans le temps, comportant trois types de défauts. Les signaux vibratoires sont d'abord traités à l'aide de la Transformée en Ondelette Empirique (EWT) pour extraire les modes AM-FM. Ensuite, une liste de caractéristiques est extraite de ces modes. Pour la sélection des caractéristiques, l'Algorithme Culturel Basé sur le Clan (CCA) est utilisé, et l'entraînement du modèle utilise l'algorithme Random Forest. Les résultats démontrent la robustesse du processus diagnostique malgré les conditions variables.

La deuxième partie se concentre sur la sélection des caractéristiques, qui joue un rôle crucial dans le contrôle de la qualité du système diagnostique et dans la réduction des facteurs trompeurs. Ce domaine de recherche attire de plus en plus l'attention, avec de nombreuses méthodes développées. Cependant, beaucoup de ces techniques nécessitent une connaissance approfondie du domaine, notamment en ce qui concerne l'ajustement des paramètres et l'interprétation des résultats. Dans ce travail, nous présentons une technique robuste basée sur l'écart type et les méthodes Random Forest pour la sélection séquentielle des caractéristiques. La méthode a été testée sur trois bases de données de roulements différentes, incluant des conditions variables dans le temps et trois techniques de décomposition du signal (EWT, EMD, et MODWPT). Elle a fourni des résultats prometteurs en termes de qualité et de quantité, étant conviviale et ne nécessitant pas de connaissances approfondies dans le domaine de l'optimisation.

Mots-clés :

Machines rotatives, classification des défauts, sélection des caractéristiques, transformée en ondelettes empiriques (EWT), forêt aléatoire (RF), conditions variables dans le temps, traitement du signal).

المساهمة في التصنيف الذكي لعيوب الآلات الدوارة.

ملخص:

أصبحت الآلات الدوارة منتشرة بشكل واسع في الصناعات المعاصرة، حيث لعبت دورًا مركزيًا في مختلف التطبيقات .إن عواقب العيوب في هذه الآلات تتجاوز المشاكل التقنية البسيطة، التي يمكن أن تؤدي إلى خسائر اقتصادية كبيرة وتشكل تهديدًا كبيرًا لسلامة الإنسان .غالبًا ما يعاني المشغلون من تعقيدات استكشاف الأخطاء وإصلاحها، حيث يمكن أن يكون لخطأ واحد عواقب وخيمة .نتيجة لذلك، كرس العديد من الباحثين وقتهم وجهدهم لتطوير مجال تشخيص الآلات الدوارة، بهدف تحديد العيوب وتحديد موقعها بسرعة عند ظهورها.

على الرغم من الدراسات مكثفة في هذا المجال، إلا أن أحد القيود الشائعة هو التركيز على سيناريوهات السرعة الثابتة .في الواقع، تعمل الآلات الدوارة بشكل عام في ظروف غير ثابتة، مما يجعل تقنيات السرعة الثابتة نظرية إلى حد كبير .تنقسم هذه الأطروحة إلى جزأين رئيسيين.

يتناول الجزء الأول تحديات حل التشخيص في ظل ظروف متغيرة بمرور الوقت .وبالنظر إلى الطابع الديناميكي لبيئة العمل، فإن فهم العيوب وتخفيفها في الظروف غير الثابتة أمر لا بد منه للتطبيقات العملية .تهدف طريقتنا إلى حل مشكلة التشخيص في ظل ظروف متغيرة بمرور الوقت .تم اختبار هذه التقنية على قاعدة بيانات للمحامل التي تم جمعها في ظل ظروف متغيرة بمرور الوقت مع ثلاثة عيوب .
تتم معالجة إشارات الاهتزاز في البداية باستخدام تحويل الموجات التجريبية (EWT) لاستخراج أوضاع .AM-FM بعد ذلك، يتم استخراج قائمة بالميزات من هذه الأوضاع .لاختيار الميزات، يتم استخدام الخوارزمية الثقافية القائمة على العشيرة(CCA) ، ويستخدم تدريب النموذج خوارزمية الغابة العشوائية .تظهر النتائج متانة عملية التشخيص على الرغم من الظروف المتغيرة.

ويركز الجزء الثاني على اختيار الخصائص، والقيام بدور حاسم في مراقبة جودة نظام التشخيص والحد من العوامل المضللة .يحظى مجال البحث هذا باهتمام متزايد، مع تطوير العديد من الأساليب . ومع ذلك، فإن العديد من هذه التقنيات تتطلب معرفة معمقة بالميدان، لا سيما فيما يتعلق بوضع المعايير
وتفسير النتائج .في هذا العمل، نقدم تقنية فعالة تعتمد على الانحراف المعياري وخوارزمية الغابة العشوائية لاختيار الميزات الأكثر وصفا للمجموعات بطريقة تسلسلية. تم اختبار الطريقة على ثلاث قواعد بيانات مختلفة تحمل، بما في ذلك ظروف مختلفة زمنيًا وثلاث تقنيات لتحلل الإشارات EWT) و EMD و .(MODWPT وقدمت نتائج واعدة من حيث النوعية والكمية، بالإضافة إلى كونها سهلة الاستعمال ولا تتطلب معرفة متعمقة في مجال التحسين الأمثل

.الكلمات المفتاحية : الآلات الدوارة، تصنيف العيوب، اختيار الميزات، الموجة التجريبية(EWT) ، الغابات العشوائية(RF) ، الظروف المتغيرة زمنيًا، معالجة الإشارات.

Acknowledgements

All praise and thanks are due to Allah, the Most Merciful and Compassionate, for granting me the strength, perseverance, and blessings that enabled me to overcome the challenges encountered on this journey.

This thesis represents the culmination of rigorous research conducted over a span of approximately five years throughout my doctoral studies at the esteemed Solid Mechanics and Systems Laboratory within the Faculty of Technology (FT). I am deeply indebted to Professor Benazzouz Djamel for affording me the invaluable opportunity to delve into this captivating research domain. His unwavering support, insightful guidance, and constructive feedback have been instrumental in shaping my academic trajectory.

I also extend my heartfelt gratitude to my esteemed supervisor, Professor Rahmoune Chemesddine, whose mentorship, expertise, and encouragement have been pivotal in navigating the complexities of this research endeavor. His profound insights and dedication to academic excellence have left an indelible mark on my scholarly pursuits.

To my beloved family, my devoted parents Mohammed and Lyakout, my dear husband Ahmed, my thoughtful and encouraging sister Ihssen, and my two supportive brothers Aymane and Anes, I owe an immense debt of gratitude for their unwavering support, boundless love, and endless patience throughout this arduous journey. Your steadfast belief in my abilities has been a constant source of motivation and strength, guiding me forward during moments of doubt and uncertainty.

Furthermore, I wish to express my sincere appreciation to my colleagues and friends whose unwavering belief in my capabilities has been a source of inspiration and encouragement.

MOUSSAOUI Imane

April; 2024

Contents

1 General Introduction **1**
1.1 Introduction . . . 2
1.2 Research objectives . . . 4
1.3 Limitations and research motivation . . . 5
1.4 Structure of the Thesis . . . 6

2 Bearing Defects and Detection Approaches **7**
2.1 Introduction . . . 8
2.2 rotating machines defects . . . 8
2.3 Bearings Generalities . . . 9
2.4 Bearing faults causes . . . 12
2.5 Fault detection and diagnosis . . . 15
2.6 Vibration Analysis for Bearing Fault Detection . . . 18
2.7 Fault detection and diagnosis techniques: State of art . . . 21
2.7.1 Model-based techniques . . . 22
2.7.2 Data-driven techniques . . . 23
2.7.3 Rule-based techniques . . . 24
2.8 Comparison of Model-based, Data-driven, and Rule-based Techniques . 25
2.9 Conclusion . . . 28

3 Signal processing and feature selection techniques **29**
3.1 Introduction . . . 30
3.2 Signal processing . . . 30
3.2.1 Time-domain techniques . . . 31
3.2.1.1 Empirical Mode Decomposition (EMD) . . . 32
3.2.2 Frequency-domain techniques . . . 34
3.2.2.1 Fourier transform . . . 34
3.2.3 Time-frequency domain techniques . . . 36
3.2.3.1 Maximal overlap discrete wavelet-packet transform . . 42
3.2.3.2 Empirical wavelet transform . . . 44
3.3 Feature extraction and selection . . . 47
3.3.1 Filter techniques . . . 49
3.3.2 Embedded techniques . . . 51

3.3.3 Wrapper techniques . . . 51
3.4 Conclusion . . . 53

4 Nature-Inspired Optimization Algorithms and Machine Learning Classification **54**
4.1 Introduction . . . 54
4.2 Nature-inspired optimization algorithms . . . 54
4.2.1 Ant colony optimization . . . 55
4.2.2 Grey wolf optimization algorithm . . . 57
4.2.3 Squirrel search algorithm . . . 58
4.2.4 Grasshopper Optimization Algorithm . . . 59
4.2.5 Simulated Annealing . . . 59
4.2.6 Firefly algorithm . . . 60
4.2.7 Binary Differential Evolution . . . 61
4.2.8 Clan-based Cultural Algorithm . . . 61
4.3 Machine learning and classification . . . 64
4.3.1 Introduction to Machine Learning . . . 64
4.3.2 Types of Machine Learning . . . 65
4.3.2.1 Supervised Learning . . . 65
4.3.2.1.1 Classification . . . 65
4.3.2.1.2 Regression . . . 73
4.3.2.2 Unsupervised Learning . . . 75
4.3.2.2.1 Clustering . . . 75
4.3.2.2.2 Dimensionality Reduction . . . 80
4.3.2.3 Reinforcement learning . . . 80
4.4 Conclusion . . . 82

5 Results and discussions **83**
5.1 Introduction . . . 84
5.1.1 Bearing fault detection under time-varying speed based on empirical wavelet transform, cultural clan-based optimization algorithm, and random forest . . . 84
5.1.1.1 Database description . . . 85
5.1.1.2 Experimental procedure . . . 87
5.1.1.3 Signal processing . . . 89
5.1.1.4 Features Selection and Optimization . . . 92
5.1.2 Rolling bearing fault feature selection based on standard deviation and random forest classifier using vibration signals . . . 95
5.1.2.1 Databases Description . . . 96
5.1.2.2 Experimental procedure . . . 99
5.1.2.3 Results and analysis . . . 101

5.1.3 Multi-fault Bearing Diagnosis Under Time-Varying Conditions Using Empirical Wavelet Transform, Gaussian Mixture Model and Random Forest Classifier 107
5.1.3.1 Database 108
5.1.3.2 Experimental procedure 108
5.1.3.3 Multi-Stage Classification 117
5.2 Experimental Evaluation 117
5.3 Results and Discussion 118
5.3.1 Conclusion 121

List of Tables

2.1 Comparison between Data-Driven, Model-Driven, and Rule-Based Approaches in Fault Detection and Diagnosis. 27

3.1 Common filter feature selection methods [1] 50
3.2 Comparison of Feature Selection Methods 52

4.1 Training dataset 70

5.1 Comparison between processed and unprocessed signal 91
5.2 Comparison between CCA, Ant, grasshopper, wolf, and firefly optimization algorithms with K-NN classifier 93
5.3 Comparison between classifiers 94
5.4 Characteristics of the second database 97
5.5 Table depicting the performance of the STD-RF algorithm with EWT. 102
5.6 Table depicting the performance of the STD-RF algorithm with EMD. 102
5.7 Table depicting the performance of the STD-RF algorithm with MOD-WPT. 102
5.8 Number of features selected and accuracies obtained by different optimization algorithms on database 1 103
5.9 Number of features selected and accuracies obtained by different optimization algorithms on database 2 104
5.10 Number of features selected and accuracies obtained by different optimization algorithms database 3 104
5.11 Table of Extracted Features 111
5.12 Binary classification results 114
5.13 Confusion matrices for the combined fault case using the 3 binary classifiers 116
5.14 Performance of optimization algorithms in Stage 1 118
5.15 Performance of optimization algorithms in Stage 2 118
5.16 Performance of optimization algorithms in Stage 3 119
5.17 Polygon Area Metric parameters 120

List of Acronyms

ABVT	**A**lignment-**B**alance-**V**ibration **T**
ACO	**A**nt **C**olony **O**ptimization
AE	**A**coustic **E**mission
AI	**A**rtificial **I**ntelligence
AM-FM	**A**mplitude **M**odulation- **F**requency **M**odulation
ANFIS	**A**daptive **N**eural **F**uzzy **I**nference **S**ystem
ASD	**A**daptive **S**ignal **D**ecomposition
AUC	**A**rea **U**nder **C**urve
BA	**B**at **A**lgorithm
BDE	**B**inary **D**ifferential **E**volution
BLSO	**B**inary **L**ocal **S**earch **O**ptimizer
CA	**C**ultural **A**lgorithm
CA	**C**lassification **A**ccuracy
CART	**C**lassification **A**nd **R**egression **T**ree
CCA	**C**lan-based **C**ultural **A**lgorithm
CFS	**C**orrelation-based **F**eature **S**election
CMFD	**C**ondition **M**onitoring and **F**ault **D**etection
CMM	**C**oordinate **M**easuring **M**achines
CS	**C**uckoo **S**earch
DR	**D**imensionality **R**eduction
DWT	**D**iscrete **W**avelet **T**ransform
EFD	**E**mpirical **F**ourier **D**ecomposition
EM	**E**xpectation-**M**aximization
EMD	**E**mpirical **M**ode **D**ecomposition
EM-GMM	**E**xpectation-**M**aximization **G**aussian **M**ixture **M**odel

EWT	**E**mpirical **W**avelet **T**ransform
FA	**F**irefly **A**lgorithm
FCBF	**F**ast **C**orrelation-**B**ased **F**ilter
FE	**F**eature **E**xtraction
FFT	**F**ast **F**ourier **T**ransform
FM	**F**-**M**easure
FMI	**F**owlkes-**M**allows **I**ndex
FN	**F**alse **N**egative
FP	**F**alse **P**ositive
FS	**F**eature **S**election
GA	**G**enetic **A**lgorithm
GMM	**G**aussian **M**ixture **M**odel
GOA	**G**rasshopper **O**ptimization **A**lgorithm
GWO	**G**rey **W**olf **O**ptimization
IG	**I**nformation **G**ain
IMF	**I**ntrinsic **M**ode **F**unction
IRT	**I**nfra**R**ed **T**hermography
JI	**J**accard **I**ndex
K	**K**appa
kNN	**K**-**N**earest **N**eighbor
LFSBSS	**L**ocalized **F**eature **S**election **B**ased on **S**catter **S**eparability
MaFaulDa	**Ma**chinery **Faul**t **Da**ta
MCFS	**M**ulti-**C**luster **F**eature **S**election
MFS	**M**achinery **F**ault **S**imulator
ML	**M**achine **L**earning
MODWPT	**M**aximal **O**verlap **D**iscrete **W**avelet **P**acket **T**ransform

MRA	**M**ulti**R**esolution **A**nalysis
NN	**N**eural **N**etwork
PAM	**P**olygon **A**rea **M**etric
PDF	**P**robability **D**ensity **F**unction
PSO	**P**article **S**warm **O**ptimization
QMF	**Q**uadrature **M**irror **F**ilter
RF	**R**andom **F**orest
RNFC	**R**emoving **N**on-bearing **F**ault **C**omponent
ROC	**R**eceiver **O**perating **C**haracteristic
SA	**S**imulated **A**nnealing
SE	**Se**nsitivity
SI	**S**warm **I**ntelligence
SNR	**S**ignal-to-**N**oise **R**atio
SOA	**S**pectroscopic **O**il **A**nalysis
SP	**Sp**ecificity
SSA	**S**quirrel **S**earch **A**lgorithm
STD	**ST**andard **D**eviation
STFT	**S**hort-**T**ime **F**ourier **T**ransform
SVM	**S**upport **V**ector **M**achine
TF	**T**otal **F**eature
TFR	**T**ime-**F**requency **R**epresentation
TN	**T**rue **N**egative
TP	**T**rue **P**ositive
VMD	**V**ariational **M**ode **D**ecomposition
WT	**W**avelet **T**ransform

List of Figures

2.1 rotating machines faults [2] 9
2.2 Principal bearing dimensions and characteristic fault frequencies . . . 14
2.3 Bearings faults spectrum with X-axis representing frequency and the Y-axis representing displacement in mm 21
2.4 Fault detection and diagnosis techniques categories 22

3.1 Envelopes Detection 33
3.2 First IMF Candidate, r_1 33
3.3 Visualization of various mother wavelet functions 41
3.4 Segmenting the Fourier axis [3] 45
3.5 Feature selection procedure 49

4.1 An experiment demonstrates how ant colonies find the shortest path. Two paths of different lengths connect the nest to the food source. In the visuals, pheromone trails are shown as dashed lines, with thickness indicating trail strength. a) There are two pathways linking the nest to the food source. b) When neither road has any pheromones, the chance of choosing either road is 50% for each. c) Ants, depicted as circles, that arrive first are more likely to opt for the shortest path. d) The shortest route will have a greater concentration of pheromones, and all ants will choose it. 56
4.2 Hierarchy of grey wolf community 57
4.3 Clan-based Cultural Algorithm 62
4.4 Machine learning categories. 65
4.5 Classification procedure. 66
4.6 A decision tree 69
4.7 Regression types. 73
4.8 Linear regression. 74
4.9 Clustering types. 76
4.10 Cluster Dendrogram. 77
4.11 The standard reinforcement-learning model. 81

5.1 SpectraQuest machinery fault simulator set-up [4] 85
5.2 Healthy bearing vibration signals 86

5.3 Vibration signals of bearings with inner race defects 86
5.4 Vibration signals of bearings with outer race defects 87
5.5 Process flowchart 88
5.6 Healthy signal decomposition using EWT 90
5.7 Inner signal decomposition using EWT 90
5.8 Outer signal decomposition using EWT 91
5.9 The plot of the classification using different classifiers. 94
5.10 Confusion matrices of classification with different classifiers. 95
5.11 The SpectraQuest machine fault simulator. 96
5.12 Bearing test rig of Case Western Reserve University Data Center. 98
5.13 Flowchart of the proposed method 99
5.14 The calculation of the centroids' coordinates 100
5.15 STD values of the selected features for the three databases processed by EWT 103
5.16 Comparison graph illustrating the percentage of selected features by different optimization methods for the first dataset. 105
5.17 Comparison graph illustrating the number of selected features by different optimization methods for the first dataset. 105
5.18 Comparison graph illustrating the number of selected features by different optimization methods for the second dataset 106
5.19 Comparison graph illustrating the number of selected features by different optimization methods for the third dataset 106
5.20 flowchart of the classification process. 109
5.21 Extracted AM-FM modes using the EWT for bearing with faulty inner race. 109
5.22 Flowchart of features selection using EM-GMM 112
5.23 Multi-stage classification 113
5.24 Polygon area metric for classification results using the selected features 120
5.25 Polygon area metric for classification results using all features. 121
5.26 Polygon area metric for classification results using the unselected features.121

Chapter 1

General Introduction

1.1 Introduction

Rotating machines are the workhorses of industry, serving as the mechanical heartbeats that power, manufacture, and propel the modern world. These versatile devices, which encompass a wide array of mechanical systems from engines to turbines, play a pivotal role in various industrial sectors. Their significance is deeply intertwined with the efficient functioning of numerous industries, each relying on these machines for specific tasks and applications.
The importance of rotating machines in industry cannot be overstated. They are indispensable in sectors such as manufacturing, energy production, and transportation, providing the necessary power and motion for a wide range of processes. Their versatility and efficiency in converting energy into mechanical work make them essential components in modern industrial operations. However, they are susceptible to failures that can disrupt operations. Bearings, on the other hand, represent one of the most critical elements in these types of machines. Bearings play a vital role in rotating equipment. The type of bearing used is determined by factors such as the application, load capacity, speed, and operating conditions. In rotating machines, bearings are positioned between the rotating shaft and the supporting machine part to facilitate smooth rotation by minimizing friction and wear. Their proper selection, maintenance, and timely replacement are essential to prevent failures and ensure the reliable performance of rotating machines. Bearings are integral to the functionality and longevity of rotating equipment, making them indispensable components in the smooth operation of various industrial processes.

Rolling bearings play a critical role within rotating machinery, functioning to minimize friction between moving components [5]. Despite their relatively small size, even minor flaws in these bearings can have far-reaching consequences, potentially disrupting the entire operation of machinery systems. Astonishingly, bearing defects are accountable for a significant share of machinery breakdowns, contributing to 40% of issues in large-scale systems and an alarming 90% in smaller-scale machinery setups [6].

The health of bearings is influenced by a myriad of factors, encompassing design irregularities, manufacturing imperfections, suboptimal mounting practices, misalignment of bearing races, variations in rolling element dimensions, inadequate lubrication, overloading, and fatigue [7]. These diverse factors can lead to substantial damage to bearing components, incurring significant losses both in terms of equipment reliability and financial expenses [6].

This situation underscores the urgent necessity to develop and deploy effective fault diagnosis methods that can ensure the smooth operation of rolling bearings and preempt unforeseen failures. These diagnostic techniques are specifically engineered to detect and identify any issues or abnormalities within the bearings' operational health. To this end, a wide spectrum of indicators is utilized, including motor current analysis, acoustic emissions, temperature measurements, and vibration analysis.

While vibration analysis and acoustic monitoring emerge as the most prevalent techniques for bearing fault diagnosis, their prominence arises from their ability to conduct diagnostics without disrupting machine operations or necessitating disassembly [8]. However, the acquisition of pristine acoustic signals can be a formidable challenge, often hindered by environmental conditions, variations in recording software configurations, and the interference of reflected acoustic signals [9].

In contrast, vibration analysis assumes the role of the preferred method for assessing machinery conditions. This technique furnishes a wealth of information about the defective component and the severity of the damage. This information is extracted from the amplitude and frequency characteristics inherent in the collected vibration signatures. It is noteworthy that more than 82% of fault diagnosis techniques rely on vibration analysis [8], which can define the non-stationary scenario precisely.

To effectively harness the insights embedded within raw vibration signals, a comprehensive processing framework is essential. This involves a multifaceted approach where the extraction of pertinent features plays a pivotal role in unveiling hidden patterns and anomalies. The feature selection step emerges as a critical juncture in this process, assuming paramount significance in refining the dataset for subsequent analysis.

In this crucial stage, the dataset undergoes meticulous curation to discern and eliminate irrelevant, redundant, or noisy information that may obscure the underlying patterns. Conversely, relevant features are carefully preserved and prioritized to optimize the efficacy of subsequent data mining algorithms. By meticulously excising extraneous data points and preserving salient features, the objective is to enhance the efficiency of predictive models, bolster the accuracy of diagnostic predictions, and render the resultant insights more interpretable and actionable.

Within the domain of bearing diagnosis, various feature selection algorithms have been extensively employed, consistently yielding high-performance outcomes. These algorithms leverage sophisticated techniques such as statistical analysis, machine learning, and signal processing to identify and prioritize features that exhibit strong correlations with fault conditions. Through rigorous experimentation and validation, these algorithms have demonstrated their efficacy in distilling complex vibration datasets into concise and informative feature sets, thereby facilitating more accurate and reliable diagnostic outcomes.

This integrative approach displays a concerted task aimed at augmenting the reliability and precision of bearing fault diagnosis across a diverse array of operational conditions and scenarios. Through the symbiotic amalgamation of state-of-the-art feature selection methodologies with advanced vibration analysis techniques, this research initiative endeavors to catalyze substantial advancements in the domain of fault detection within machinery systems.

By synergistically harnessing the analytical prowess of advanced feature selection algorithms alongside rigorous vibration analysis protocols, this research seeks to develop robust and empirically grounded methodologies for the early detection and characterization of bearing faults under non-stationary conditions. Through methodical experimentation and meticulous validation, the overarching objective is to elucidate the intricate dynamics governing the manifestation of fault signatures within vibration signals, thereby furnishing engineers and maintenance practitioners with empirically validated tools and insights essential for proactive fault mitigation strategies and the optimization of machinery reliability and performance.

As we have established the pivotal role of bearings in rotating machines and the pressing need for effective fault detection methods, it's imperative to outline the structure of our thesis. The upcoming chapters are organised as follows:

Chapter 1 offers a comprehensive introduction, delving into the motivations driving our research and its overarching objectives.

Chapter 2 provides a detailed review of bearings, covering their defects and factors influencing their condition monitoring and fault detection.

Chapter 3 presents an analysis of existing fault detection techniques, showcasing the latest advancements in the field.

Chapter 4 elucidates the procedural framework and technical tools utilized throughout our research process.

Chapter 5 encapsulates our findings, offering a thorough analysis of the data collected.

Finally, the concluding chapter ties together key insights and implications derived from our study.

1.2 Research objectives

This thesis aims to achieve several key objectives aimed at advancing the field of fault diagnostics for rotating machines. The first set of objectives focuses on enhancing diagnostic approaches to effectively address varying operational conditions. The goal is to develop innovative diagnostic techniques that can seamlessly adapt to the dynamic nature of operational scenarios experienced by rotating machines. By mitigating potential economic losses and threats to human safety resulting from defects in these machines, the research aims to overcome the common limitation identified in prior studies, where diagnostic approaches were primarily focused on constant-speed scenarios.

Simultaneously, a parallel set of objectives aims to optimize the feature selection process for improved fault diagnostics. Recognizing the challenge of identifying parameters that accurately discriminate between different fault conditions, the research seeks to explore and implement an enhanced feature selection process. This involves considering various statistical parameters commonly used in fault diagnostics and understanding the crucial role of optimal features for effective pattern recognition and fault detection. The overarching objective is to elevate the accuracy of fault diagnostics in real-world scenarios.

Ultimately, the research aims to advance the efficiency and reliability of the diagnostic process by integrating the developed diagnostic techniques for varying operational conditions with refined feature selection methods. The proposed techniques will be rigorously evaluated and validated through comprehensive testing under diverse operational scenarios and fault conditions. By achieving these objectives, the research seeks to contribute significantly to the broader field of machine health monitoring and fault detection, ensuring a comprehensive improvement in the overall efficiency and reliability of diagnostic procedures.

This integrated approach to fault diagnosis promises to deliver a holistic solution to the challenges faced by rotating machines in contemporary industries. By addressing the complexity of variable operational conditions and optimizing the feature selection process, the research aims to establish a robust framework for accurate and timely fault detection, ultimately contributing to the enhancement of equipment reliability and safety. As a result, the research will lay the groundwork for future advancements in the field of rotating machine diagnostics, setting new benchmarks for the industry and promoting sustainable growth.

1.3 Limitations and research motivation

In the realm of rotating machines, rolling bearings emerge as indispensable components crucial to their operational integrity. These bearings, fundamental to the functioning of machinery, inherently generate vibrations that serve as vital indicators of their current state, offering invaluable insights into machinery health.

This thesis tackles a key research challenge centered on achieving diagnostic precision for rotating machines. These machines, pervasive across modern industries, play pivotal roles in diverse applications. However, their operation under non-stationary conditions presents a formidable obstacle, rendering conventional diagnostic approaches tailored to stationary conditions insufficient. Addressing the complexities of troubleshooting these systems becomes a multifaceted endeavor, as defects extend beyond mere technical issues, potentially leading to substantial economic losses and posing tangible risks to human safety.

Further complicating matters is the prevalent limitation observed in prior research—a narrow focus on constant-speed scenarios. This limitation underscores the urgent need for practical diagnostic techniques customized to the dynamic operational conditions of rotating machines. The primary aim of this thesis is to develop a diagnostic procedure capable of effectively navigating time-varying conditions, ensuring precise diagnoses in real-world applications.

Another significant challenge in this diagnostic process pertains to the identification of suitable parameters capable of accurately distinguishing between different fault conditions of the bearing. The selection of optimal features represents a critical phase in pattern recognition, necessitating a deep understanding of the relevant domain. Chosen features must adeptly capture subtle behavioral changes induced by various fault conditions and exhibit robustness in efficiently describing faults under diverse operating conditions, including varying speeds and loads.

In the contemporary landscape of data-driven applications, the identification of pertinent features has emerged as pivotal for maximizing the effectiveness of data mining algorithms. Various feature selection methods proposed in the literature aim to extract the most relevant feature or feature subsets, facilitating classification and clustering objectives in real-world scenarios.

However, selecting appropriate algorithms for this purpose poses its own array of challenges. While many of these algorithms show theoretical promise, they often prove time-consuming and impractical for online activities due to the delays they introduce. Moreover, their effective utilization demands a profound comprehension of the domain to fine-tune hyperparameters and interpret results accurately. This underscores the pressing need for streamlined and efficient feature selection methods that align with the practical demands of real-world diagnostic applications.

1.4 Structure of the Thesis

This section outlines the structure of the thesis. The upcoming chapters are organized as follows:

Chapter 1: Provides a comprehensive introduction, discussing the motivations behind the research, its objectives, and the scope of the study.

Chapter 2: Presents an in-depth review of bearings, focusing on their defects and the various fault detection approaches.

Chapter 3: Explores signal processing techniques and the feature selection steps integral to the research methodology.

Chapter 4: Investigates nature-inspired algorithms and machine learning classification techniques.

Chapter 5: Summarizes the findings, offering a thorough analysis of the data collected throughout the research process.

Finally, the concluding chapter synthesizes key insights and discusses the broader implications of the study's outcomes.

Chapter 2

Bearing Defects and Detection Approaches

2.1 Introduction

In this chapter, we'll discuss rotating machines and bearings defects. We will demonstrate bearings' role in facilitating smooth rotation and operational efficiency across a wide range of machinery applications. We'll explore the significance of bearings in ensuring the seamless functioning of rotating equipment, highlighting their contribution to overall performance and reliability.

Additionally, we'll examine the various types of defects that bearings may encounter during their operational lifespan. From common issues such as wear and tear to more complex challenges like misalignment and fatigue, we'll discuss the diverse range of factors that can affect bearing performance and longevity.

Moreover, we'll explore the indicators and techniques used for monitoring the condition of bearings in real time. Through methods such as vibration analysis, thermal imaging, and lubricant analysis, we'll uncover how engineers and maintenance professionals assess the health and performance of bearings to detect potential issues before they escalate into major failures.

By delving into these topics, this chapter aims to provide a comprehensive understanding of the role of bearings in rotating machines, the challenges associated with bearing defects, and the importance of proactive condition monitoring for ensuring optimal performance and reliability.

2.2 rotating machines defects

Rotating machines are essential in various industries, their importance lies in their ability to change the state of working fluids, convey or transport fluids, extract energy, and create propulsion. They are vital for power generation, various forms of transportation, and a wide array of industrial processes. Hence, they play a vital role across diverse industries, significantly enhancing the efficiency of various processes this is why they are becoming indispensable in many factories. However, like any other mechanical system, rotating machines are susceptible to failure and downtime due to numerous factors. These factors include harsh environmental conditions, suboptimal maintenance practices, prolonged and intensive usage, and other circumstances. The intricate nature of rotating machines, involving numerous interconnected components and continuous operational demands, renders them particularly powerless to wear and tear. Exposure to extreme temperatures, contaminants, and abrasive substances in the operating environment can accelerate the deterioration of crucial components, ultimately leading to operational malfunctions. Additionally, inadequate or irregular maintenance practices, including lubrication, calibration, and component inspections, contribute significantly to the degradation of machine performance and reliability. The sheer frequency and intensity of usage, especially in industrial settings, can expedite the wear of essential parts, thereby increasing the risk of unexpected breakdowns.
rotating machines are exposed to different damages related to their different parts,

from rotor, stator, shafts, gears and bearings. Figure 2.1 represents the percentage of different defects of the rotating machines.

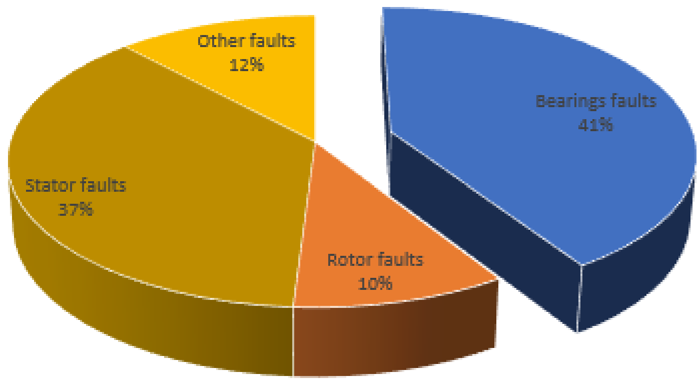

Figure 2.1: rotating machines faults [2]

In figure 2.1 bearing represents bearing faults contribute to 41% of the faults experienced by a 200 hp motor [2]. The contribution of bearing faults may vary from 40% to 90% in large to small-range machines [10]. These statistics highlight the significant impact of bearing defects on the operation of rotating machinery and the importance of detecting and diagnosing these defects. This is why focused our interest in studying, detecting and identifying bearing defects.

2.3 Bearings Generalities

Bearings, as essential mechanical components, fulfill a paramount role in the realm of machinery and equipment that encompass rotating motion. Their primary function is to facilitate relative motion between two or more parts by mitigating the detrimental effects of friction, thereby enhancing the overall efficiency of the system and prolonging the operational lifespan of the machinery. This friction-reducing attribute is pivotal, as it not only conserves energy but also mitigates wear and tear on vital components. Furthermore, bearings exhibit remarkable versatility by adeptly supporting both radial loads, which act perpendicular to the axis of rotation, and axial loads, which act parallel to the axis of rotation. This multifaceted support ensures that machinery can adeptly handle an array of force types, ranging from the complex interplay of forces in complex industrial systems to the more straightforward loads experienced by household appliances.

In essence, the function of the bearing is to enable smooth and efficient operation. They are everywhere in diverse applications, seamlessly transitioning from the inner workings of small household devices, such as washing machines and ceiling fans, to the heavy-duty demands of industrial giants like conveyor systems, construction equipment, and massive manufacturing machinery. In the industrial landscape, where precision, reliability, and productivity are paramount, bearings are indispensable companions, silently and efficiently facilitating the rotating dance of gears, shafts, and wheels, ultimately driving the wheels of progress across diverse sectors.

Types of Bearings

Bearings come in various types, each designed for specific applications and load-bearing requirements. Here is an overview of some common types:

- **Rolling Bearings:** consist of
  - **Ball Bearings:** These bearings use spherical balls to separate the moving parts and reduce friction. They are further categorized into:
    - **Deep Groove Ball Bearings:** Suitable for high-speed and radial loads, often used in applications like electric motors and appliances.
    - **Angular Contact Ball Bearings:** Designed to handle both radial and axial loads, commonly found in automotive wheels and machine tool spindles.
    - **Thrust Ball Bearings:** Primarily designed for axial loads, used in applications like thrust reversers in aircraft engines.
  - **Roller Bearings:** Roller bearings use cylindrical, tapered, or spherical rollers to distribute loads and reduce friction. They include:
    - **Cylindrical Roller Bearings:** Suitable for high radial loads, found in applications like conveyor systems and large motors.
    - **Tapered Roller Bearings:** Designed to handle both radial and axial loads, commonly used in automotive transmissions and heavy machinery.
    - **Spherical Roller Bearings:** Able to handle heavy radial and axial loads, often used in applications where misalignment might occur, such as in mining and construction equipment.
- **Plain Bearings (Bushings):** These bearings consist of a sliding surface [11], often made of materials like bronze or plastic, that allows two parts to move relative to each other smoothly. They are divided into:
  - **Sleeve Bearings:** Used in applications where a shaft rotates within the bearing, like in electric motors.
  - **Flanged Bearings:** Similar to sleeve bearings but with a flange to aid in positioning.
- **Needle Bearings:** Needle roller bearings use long, thin cylinders (needle rollers) to support high radial loads in constrained spaces. They are commonly used in automotive engines, transmissions, and motorcycle suspension systems.

- **Thrust Bearings:** These bearings are designed to handle axial loads (force along the axis of rotation) and are divided into:
  - **Ball Thrust Bearings:** Contain balls and are often used in applications like automotive transmissions.
  - **Roller Thrust Bearings:** Utilize cylindrical rollers and they are suitable for heavy-duty applications, including large industrial machinery.
- **Spherical Bearings:** Allow for misalignment between two connected parts and are used in applications where flexibility is needed. They include:
  - **Spherical Plain Bearings:** Used in articulating joints, suspension systems, and hydraulic cylinders.
  - **Spherical Roller Thrust Bearings:** Designed for applications involving heavy axial loads and misalignment.
- **Mounted Bearings:** These are pre-assembled bearings incorporated into housings for easy installation. They are commonly used in conveyor systems, agricultural equipment, and HVAC systems.
- **Linear Bearings:** Linear bearings enable motion along a straight path and are widely used in applications such as CNC machines, 3D printers, and industrial robots.
- **Air Bearings:** These use a thin film of compressed air to support and guide moving parts, offering ultra-low friction and precision. They are used in high-precision machinery like coordinate measuring machines (CMMs) and semiconductor manufacturing equipment.

The choice of bearing type depends on factors such as load capacity, speed, environmental conditions, and the specific needs of the machinery or equipment. In our study, we focused our interest on rolling bearings, being the most utilized type in the industry due to several advantages that make them suitable for a wide range of applications. Rolling bearings offer a significant load-carrying capacity. Additionally, rolling bearings contribute to increased efficiency by reducing friction and energy loss, which is crucial for optimizing the performance of machines and systems. Their precision over motion is essential in applications requiring accuracy and reliability, such as machine tools and robotics. Furthermore, rolling bearings are available in various designs, materials, and configurations, providing adaptability to different industrial settings. They benefit from standardization, as they come in standardized sizes and specifications, simplifying the selection and integration of bearings into diverse products and systems.

2.4 Bearing faults causes

Rolling bearings are mechanical components that facilitate smooth rotation between two or more parts. They are used in various industrial applications, such as automotive, aerospace, and manufacturing machinery. The three primary components of a rolling bearing are the inner ring, the outer ring, and the rolling elements [12]. The inner ring is the part that rotates along with the shaft or rotating component, while the outer ring remains stationary. The rolling elements (balls or rollers) reduce friction and distribute the load evenly between the inner and outer rings. In addition to the three primary components, most rolling bearings contain the cage, which holds the rolling elements in place and ensures that they do not come into contact among themselves. It is usually made of steel or plastic and may take various shapes depending on the design of the bearing [13].
Rolling element bearings are designed to operate for a specified service life determined during manufacturing based on expected operating conditions, load, and speed. However, despite careful design and manufacturing processes, premature bearing failures can occur due to various factors [14]. In the literature, the bearing faults are categorised based on the following:

- the fault's location: inner race, outer race, balls, and cage.
- the fault signature: single-point defects and generalized roughness [15].

Single-point defects

A single-point defect is recognized for its ability to introduce discernible and distinctive fault frequencies within the vibration spectrum of a machine. These fault frequencies, often referred to as characteristic frequencies, are not random occurrences but can be predicted. Their manifestation is intricately tied to the precise bearing surface affected by the defect. Single-point defects manifest as periodic impulses within the vibration signals of the machine. The key characteristics of these impulses, namely their amplitude and periodicity, are intimately linked to several influential factors, including the rotational speed of the machine's shaft, the precise location of the fault on the bearing surface, and the dimensions of the bearing itself. Consequently, each component of a bearing can be associated with a specific and identifiable frequency. In essence, when a single-point defect occurs within a machine's bearing, it sets in motion a chain of events that culminate in the creation of these distinct vibration frequencies. The rotational speed of the machine's shaft is a fundamental factor, as it determines the periodicity of the impulses. Meanwhile, the precise location of the fault on the bearing surface contributes to the unique spectral pattern associated with the defect. Additionally, the bearing's physical dimensions influence the amplitude of the vibration frequencies generated. This inherent relationship between single-point defects and vibration frequencies provides a valuable foundation for fault diagnosis and localization. By analyzing the vibration spectrum and identifying the specific frequencies associated with each bearing component, maintenance professionals and engineers can pinpoint the nature and location of faults in the bearing system. The primary frequency associated with the cage's motion can be mathematically defined as:

$$f_c = \frac{f_r}{2}\left(1 - \frac{d}{D}\cos(\alpha)\right) \tag{2.1}$$

This expression encapsulates the essential parameters: f_r symbolizing the rotor's speed, d representing the diameter of the ball, D denoting the pitch diameter of the bearing, and α signifying the angle of contact. Together, these elements harmonize to determine the cage's fundamental frequency.

Moving beyond the cage, another crucial frequency emerges—the ball defect frequency:

$$f_{bd} = \frac{D}{d} f_r \left(1 - \frac{d^2}{D^2}\cos^2(\alpha)\right) \tag{2.2}$$

This frequency unveils potential issues within the bearings, providing valuable insights into their health. It's a frequency intricately intertwined with the dimensions of the ball, the bearing's pitch diameter, the rotor's speed, and the angle of contact.

Now, let's delve into the inner race frequency, a key parameter in bearing analysis:

$$f_{id} = n\left(f_r - f_c\right) = \frac{n f_r}{2}\left(1 - \frac{d}{D}\cos(\alpha)\right) \tag{2.3}$$

This frequency embodies the inner race's dynamics, revealing important information about potential faults or anomalies. It relies on the number of balls, the rotor's speed, the ball diameter, the pitch diameter of the bearing, and the angle of contact.

Completing the quartet of frequencies is the frequency of the outer race:

$$f_{od} = n f_c = \frac{n f_r}{2d}\left(1 - \frac{d}{D}\cos(\alpha)\right) \tag{2.4}$$

This frequency characterizes the behavior of the outer race and plays a crucial role in identifying any irregularities. It relies on parameters such as the number of balls, the rotor's speed, the ball diameter, the pitch diameter of the bearing, and the angle of contact.

In summary, these frequency equations provide valuable tools for diagnosing bearing health and pinpointing potential defects, with each frequency shedding light on a different aspect of the bearing's condition. Understanding the interplay between these parameters is essential for effective bearing fault diagnosis [15].

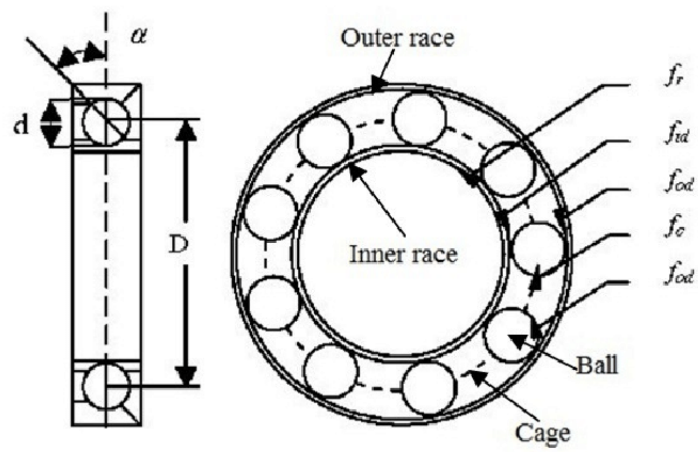

Figure 2.2: Principal bearing dimensions and characteristic fault frequencies

Generalized roughness fault

Generalized roughness is the most common type of damage that occurs to rolling bearings. It refers to a condition where the bearing surface becomes rough or uneven over a wide area due to factors such as:

- Fatigue Surface contact fatigue is a frequently encountered factor in the failure of rolling-element bearings. The degree of damage observed is affected by a combination of factors, including the contact loads, the curvature of the rolling elements, and the relative motion between the contacting surfaces. The various types of contact fatigue are as follows: Firstly, under pure rolling contact, microscopic pits are formed. These pits serve as stress concentration sites and can result in additional faults. Secondly, irregular-shaped pits are formed under rolling sliding contact, which can accelerate the damage process when geometric inhomogeneities such as corrosion pits, handling damage, and dents are present. Thirdly, flaking occurs from the progression of pits formed under rolling and rolling-sliding contact fatigue, which creates large, irregular-shaped pits. Lastly, spalling is the development of large, deep pits with sharp edges, steep sides, and flat bases or cracking at the case-core interface in case-hardened surfaces [14].

- Wear

  Wear is a prevalent factor contributing to bearing failure. Its occurrence is mainly attributed to the ingress of dirt and foreign particles into the bearing because of inadequate sealing or lubricant contamination. The abrasive nature of these foreign particles roughens the contacting surfaces, leading to a dull appearance. In severe wear cases, the raceway profile is altered, and the rolling element profile and diameter are affected, increasing the bearing clearance. Consequently, rolling friction significantly increases, which leads to high levels of slip and skidding. The culmination of these effects is a complete breakdown of the bearing [14].

- Corrosion Corrosion damage is a significant factor contributing to the failure of rolling-element bearings. It occurs when contaminants such as water, acids, and other substances enter the bearing arrangement. Various factors as damaged seals, acidic lubricants and condensation, can cause corrosion. For instance, when bearings cool from a higher operating temperature in humid air, condensation can occur, leading to rust formation on the running surfaces. Rust formation on the rolling bearings' surfaces produces uneven and noisy operation, as the rust particles interfere with the lubrication and smooth rolling action of the rolling elements. The corrosion process is affected by many factors, including the type of lubricant used, operating temperature, and relative humidity. In addition, the extent and severity of corrosion damage can be affected by the bearing material used and its surface finish [14].

- Lubrication Inadequate lubrication is a significant contributor to premature bearing failure, and it is crucial to maintain proper lubrication to ensure the longevity of the bearing. The lack of lubrication can result in several issues as skidding, slipping, increased friction, heat generation, and sticking. These problems can cause significant damage to the bearing, which leads to a complete breakdown. Insufficient lubrication can cause the contacting surfaces to weld together at the highly stressed region of Hertzian contact. As the rolling element moves on, the shells are separated, resulting in damage and increased wear. The damage caused by inadequate lubrication can result in significant costs and downtime. The three critical points of bearing lubrication include the cage-roller interface, the roller-race interface, and the cage-race interface. It is essential to ensure that lubricant is adequately supplied to these points to reduce friction and prevent damage. Furthermore, proper lubrication can help dissipate heat, reduce wear, and extend the life of the bearing [14].

- Plastic deformation Plastic deformation of the contacting surfaces in bearings is a common phenomenon observed when bearings support excessive loads while stationary or undergoing small movements. This phenomenon relates to localized plastic deformation, which causes indentation of the raceway. This plastic deformation leads to changes in the surface roughness, which alters the bearing performance, leading to excessive vibration during rotation. The deformed bearing will also experience an increase in operating temperature. This temperature degrades the material properties and results in a decrease in the bearing life. [14].

2.5 Fault detection and diagnosis

Condition monitoring and fault detection (CMFD) techniques play a vital role in ensuring the reliability and performance of bearings in diverse applications, They offer insights into the operational state of machines that enable the detection of any deterioration in its condition. Also, it allows timely implementation of preventive measures to avert catastrophic failures, these techniques empower proactive maintenance practices by swiftly identifying and diagnosing faults in their early stages, thereby minimizing costly downtime and preventing unexpected failures. Many indicators are available to provide valuable insights into the type and severity of defects [16] by monitoring various output parameters where the alterations observed in these parameters aid in identifying the emergence of faults, diagnosing their root causes, and anticipating potential failures. In the subsequent section, we will delve into the most commonly employed indicators within the CMFD process.

- **Current measurements** Defects in bearings within a mechanical system driven by an induction motor lead to changes in the spectrum of its stator current. These bearing faults cause fluctuations in load irregularities within the magnetic field, resulting in variations in mutual and self-inductance. Consequently, this gives rise to side-bands that appear alongside the line frequency [17]. it is a non-invasive technique. It offers a non-disruptive approach to bearing fault detection. It relies on the pre-existing electrical current measurements, negating the need for additional sensors or equipment. This non-intrusive nature not only simplifies the implementation process but also minimizes any potential interference with the normal operation of the machinery system. By leveraging the existing electrical current measurements, the method effectively taps into valuable information already available within the system, eliminating the need for costly and time-consuming sensor installations or modifications. [18].
  The source of nearly all important frequency components in the stator current is widely known, making it easy to eliminate components unrelated to bearing faults. Moreover, it is possible to predict most of the significant frequency components in the stator current based on fundamental information about the machine, such as its speed. This comprehensive understanding of the spectral content of the stator current enables the effective detection of most electric machine faults as soon as their characteristic patterns become prominent [18].

- **Acoustic measurements** Acoustic Emission (AE) is a technique that captures the generation of transient elastic waves due to different events, such as cyclic fatigue and fractures. In the context of bearings, the application of AE becomes particularly relevant [19]. It allows the detection of acoustic waves produced from the formation of subsurface cracks caused by the Hertzian contact stress created by the rolling action of the bearing elements on the inner and outer races [20]. Additionally, the interaction between damaged mating surfaces within the bearing leads to rubbing, which generates further acoustic emission. One of the critical advantages of AE-based analysis is its remarkable capability to detect and identify extremely low-energy signals associated with bearing failures, making it highly effective in detecting issues at an early stage or even during operations at low speeds [2]. By leveraging the sensitivity of AE to subtle acoustic emissions, this method enables proactive monitoring and assessment of bearing health, facilitating timely maintenance and reducing the risk of unexpected failures [19].

However, it is essential to notice that the propagation of Acoustic Emission (AE) signals is subject to the influence of various factors that can significantly impact its behaviour. One crucial aspect is the microstructure of the monitored material, and its grain structure of the material can also affect the generation and transmission of AE waves. The arrangement of free surfaces also plays a role in AE propagation. The geometry and surface conditions of the structure or component under inspection can affect the transmission and scattering of AE waves. Irregularities, roughness, or the presence of surface coatings can modify the path and intensity of the AE signals [20].

- **Infrared thermography** Infrared Thermography (IRT) is a highly effective non-invasive, and non-contact condition monitoring tool widely utilized in various industries. This technique employs thermal imaging to detect and analyze the temperature distribution on the surface of an object or system. With its ability to capture infrared radiation emitted by objects, IRT provides valuable insights into the thermal behaviour of components, including bearings. IRT relies on thermal energy measurements (heat) radiated from the surface of a bearing housing, which is transformed later into a thermal image or surface temperature map [21].

- **Wear Debris and Lubricating Oil Analysis** Lubricating oil is utilized in machinery to minimize friction and provide cooling. It directly interacts with bearings, which undergo gradual deterioration over time, resulting in the generation of small wear particles carried by the lubricating oil. Examining these particles, called wear debris, can yield valuable insights into the machine's condition, as it directly corresponds to the underlying source of wear and its impact on overall machine performance.

  Wear occurs due to interactions between surfaces, leading to the production of particles primarily through friction. In systems relying on oil lubrication, the debris present in the lubrication system provides valuable information about the nature and severity of faults. Analyzing the composition and properties of this debris enables us to identify specific defects and evaluate the remaining operational lifespan of the machinery [22]. This method is widely used to directly assess surface degradation in machines. It involves examining a representative sample of the lubricating fluid to analyze the concentration, size, chemical composition, colour, shape, and surface properties of the wear particles. This comprehensive analysis helps in understanding the mechanisms driving wear and assists in evaluating the overall health and performance of the machine. It is essential to obtain an oil sample from the machine for analysis to interpret the results accurately. Hence, sampling plays a crucial role in this process. These samples should be wear particles homogeneously mixed with the oil and be picked from the return line before reaching the oil filter. Wear particles can range in size from a few microns to 1000 microns.
  Various techniques are available to examine wear debris, including magnetic plugs,

ferrography, and spectroscopic oil analysis (SOA). These techniques are sensitive to different particle sizes. Magnetic plugs are effective for detecting particles larger than 50 microns, ferrography is suitable for particles ranging from 10 to 300 microns, and SOA is sensitive to particles smaller than 10 microns. Therefore, selecting the appropriate technique depends on the size range of the analyzed particles.

- **Vibration analysis** Defective bearings in machinery give rise to vibrations, which serve as valuable indicators of the bearing's condition. During operation, the condition of the bearings changes, leading to variations in the vibration patterns [23]. Each machine has a characteristic level of vibration that is considered normal or acceptable. However, there are instances when the vibration levels deviate from the expected range, indicating the presence of mechanical issues. A noticeable increase in vibration is often indicative of the occurrence of mechanical problems .

  Vibration signals are particularly informative in bearing fault detection due to the characteristic impulse components generated by localized defects or general roughness within the bearing. These impulses arise as the ball bearing passes through the points of the defects, resulting in consecutive and periodic patterns within the machine's vibration signal [24]. Vibration signals are collected using sensors. These are proximity sensors, velocity transducers and accelerometers. Accelerometers are the most used for vibration analysis due to their effectiveness in measuring and recording vibrations accurately. These collected data contain a large quantity of information that is often challenging to comprehend. As a result, the signals undergo diverse processing methods to emphasize specific aspects within the overall signal [25].

2.6 Vibration Analysis for Bearing Fault Detection

Vibration analysis is preferred over other techniques for several reasons, including its ability to monitor hard-to-access components without planned shutdowns, provide real-time insight into the condition of critical assets, and offer established standard operating procedures, methodologies, and software to simplify the analysis process. Additionally, vibration analysis can be used remotely and has niche applications beyond rotating machinery, such as monitoring the structural integrity of bridges, pipes, and other infrastructure. Furthermore, it helps identify issues like broken gear teeth early, and there are various software solutions specifically designed for vibration analysis. Moreover, vibration analysis can improve plant efficiency, reduce costs, and help avoid supply chain issues by identifying and addressing abnormal vibrations before they cause problems, thus improving machine performance and reducing the need for unnecessary maintenance. Therefore, the advantages of vibration analysis, including its versatility, real-time capabilities, and impact on cost savings, make it a preferred technique for many maintenance and reliability professionals. Numerous surveys consistently affirm the widespread adoption of vibration signatures as the predominant method for fault diagnosis, surpassing alternative techniques like temperature monitoring, current analysis, and acoustic emissions. The preeminence of vibration analysis can be attributed to several key factors.

First and foremost, the acquisition of vibration data stands out for its relative simplicity when compared to other diagnostic parameters. Vibration sensors are readily available, and the installation and setup processes are notably straightforward. This accessibility streamlines the collection of vital vibration signals from a diverse range of machinery and systems.

Moreover, vibration analysis serves as a gateway to profound insights into the origin and severity of faults. Vibration signatures encapsulate a wealth of information about the dynamic behavior and mechanical state of the observed system. By meticulously scrutinizing the frequency content, amplitude variations, and temporal patterns of vibrations, analysts are equipped to pinpoint underlying anomalies, detect the incipient stages of faults, and gauge the severity of these faults. The rich information contained within vibration signatures significantly enhances diagnostic capabilities [26].

Furthermore, in comparison to signals obtained from alternative sensors, such as acoustic emissions, vibration signals derived from accelerometers offer the distinct advantage of covering a wide dynamic range and a broad frequency spectrum [27].

The utility of vibration signals in diagnostic applications has been widely recognized, with a plethora of studies achieving remarkable accuracy. For instance, in the work by Deng et al. [28], vibration signals served as inputs for their methodology, which incorporated the Empirical Wavelet Transform (EWT) for signal decomposition. Subsequently, fuzzy entropy was employed to assess the complexity of vibration signals, effectively capturing variations in intrinsic oscillations. The fuzzy entropy values of the AM-FM (Amplitude Modulation-Frequency Modulation) components were then computed and used as inputs for training and constructing a Support Vector Machine (SVM) classifier, enabling fault pattern recognition.

In another notable study, Zarei et al. [24] utilized vibration signals along with a neural network-designed filter known as the "removing non-bearing fault component (RNFC) filter" to eliminate non-bearing fault components from the vibration signal. The filtered signal was subsequently fed into a second neural network employing pattern recognition techniques for fault classification.

Additionally, Attoui et al. [29] employed the principles of Discrete Wavelet Transform (DWT) and Fast Fourier Transform (FFT) to extract the amplitude of the primary frequencies associated with bearing defects from the vibration signal obtained from a rotating machine. These extracted parameters were subsequently utilized by the Adaptive Neural Fuzzy Inference System (ANFIS) to automate the process of fault detection and diagnosis.

Zair et al. [30] introduced a novel approach for the multi-fault diagnosis of rolling bearings using vibration signals. This method combined the utilization of fuzzy entropy derived from empirical mode decomposition for signal processing, principal component

analysis for dimensionality reduction, and a self-organizing map neural network for classification.
In yet another contribution, Altaf et al. [31] employed the second derivative of time domain signals and the power spectral density of vibration signals, along with their corresponding frequency spectrum, to obtain a set of parameters. These parameters were subsequently used in conjunction with machine learning algorithms such as K-nearest neighbor, support vector machine, and kernel linear discriminant analysis.
A method introduced by Yang et al. [32] involved detecting bearing faults through vibration analysis, utilizing the basis pursuit technique in combination with a neural network (NN). In the study by Paya et al. [33], expert systems and fuzzy logic were applied to the task of diagnosing faults in rolling element bearings by analyzing vibration characteristics.
Martin et al. [34] employed kurtosis and skewness measurements of vibration signals to identify early-stage bearing faults. Heng et al. [35] explored the utilization of statistical analysis techniques with sound pressure and vibration signals to detect defects in rolling element bearings. It's worth noting, however, that statistical analysis may have limitations in identifying bearing defects in advanced stages of development.
Furthermore, a distinct method was introduced by [22], centered on cyclostationary analysis, to examine vibration signals from bearings. In the study by Karacay et al. [36], synchronous averaging was utilized to examine the calculation of the envelope signal that emerges from the high-frequency vibrations caused by spalling damage in rolling element bearings.
The versatility of vibration analysis lies in its ability to be applied to various applications, machinery, and vibration signals, as well as in conjunction with other techniques, ultimately supporting more effective condition monitoring and fault diagnosis in various industrial applications. However, as we can see in figure 2.3 bearings defects can not be distinguished only from the vibration spectrum, hence, signal decomposition is a crucial preprocessing step in vibration analysis that allows for the extraction of useful knowledge and insight into the data and relevant underlying systems, particularly in the context of non-stationary signals.

This process is essential as it enables the separation of non-stationary signals into their constituent components, providing a better understanding of the dynamic behavior of the system. Moreover, the use of adaptive signal decomposition methods has become increasingly common in the industry for vibration-based condition monitoring and fault diagnosis, contributing to the overall increase in machine availability and reliability. Therefore, the importance of signal decomposition in vibration analysis is evident, as it enables the extraction of valuable information from non-stationary vibration signals, ultimately supporting more effective condition monitoring and fault diagnosis in various industrial applications. This step will be explained deeply in the methodology and contributions chapter.

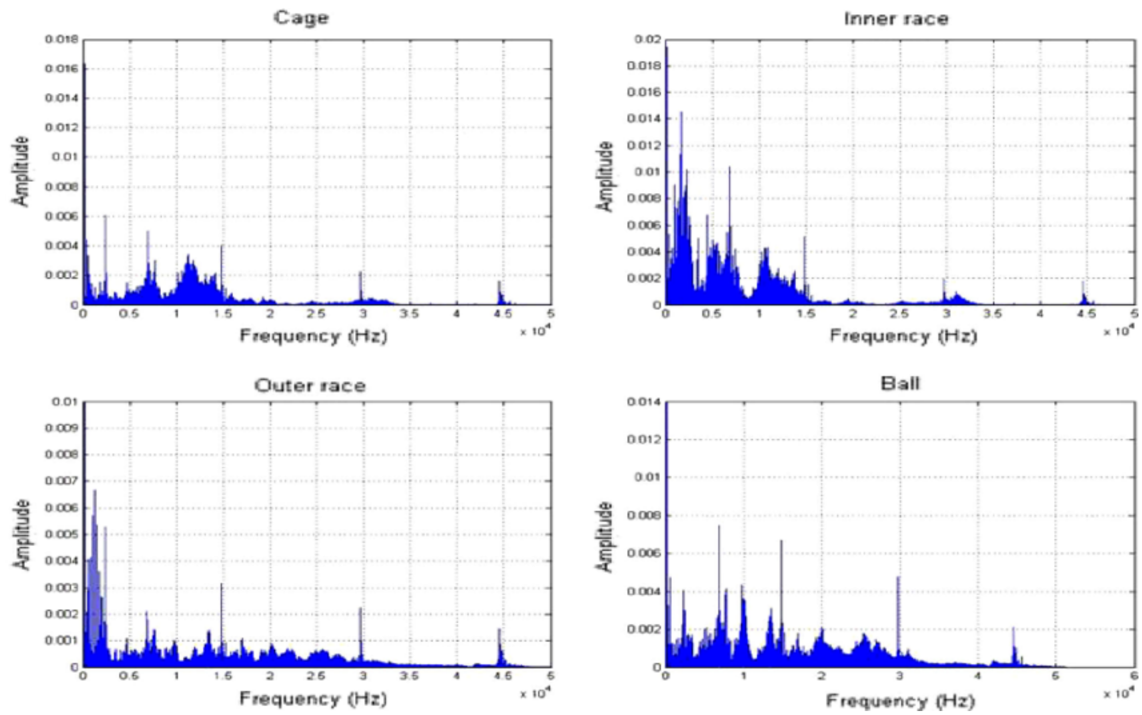

Figure 2.3: Bearings faults spectrum with X-axis representing frequency and the Y-axis representing displacement in mm

2.7 Fault detection and diagnosis techniques: State of art

Since its inception in the late 1960s, modern control theory has evolved into a cornerstone in the field of control systems, with branches like system identification, adaptive control, robust control, optimal control, variable structure control, and stochastic system theory finding diverse applications. Despite its progress and versatility, the field remains dynamic, offering numerous challenging topics for exploration from both theoretical and practical perspectives. As technological landscapes continually evolve and systems become increasingly intricate, modern control theory faces new dimensions of challenges. This evolution underscores the importance of addressing the growing complexities in systems. This brings us to the focus on fault detection and classification, which becomes paramount in ensuring the reliability and efficiency of these diverse systems. The need for robust methodologies to identify and categorize faults has grown exponentially, prompting the emergence of a myriad of methodologies, each aiming to refine the precision of fault detection processes.

Numerous methodologies have been developed to address the intricate challenge of fault detection and classification, each striving to enhance the precision of this pivotal process. These approaches broadly fall into three distinct groups (figure 2.4: model-based techniques, data-driven techniques, and rule-based techniques [37].

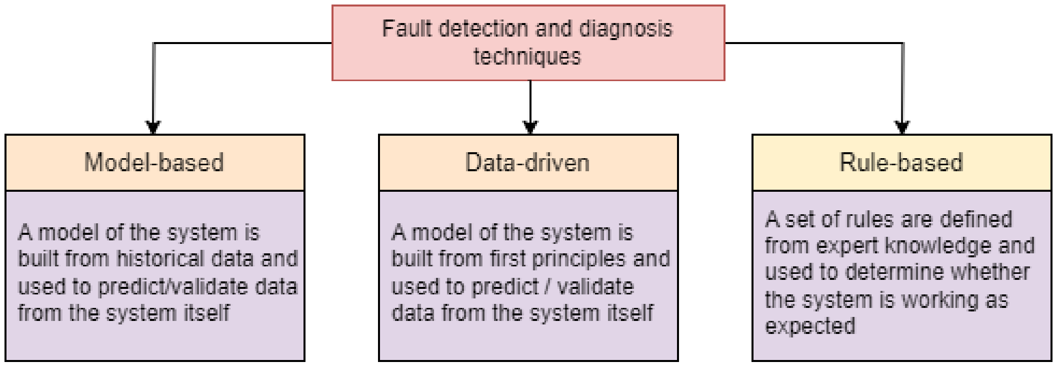

Figure 2.4: Fault detection and diagnosis techniques categories

Model-based approaches rely on theoretical models of system behavior, leveraging a comprehensive understanding of system dynamics to detect and classify faults effectively. These methods often involve mathematical models, physical simulations, or statistical representations of the system's behavior [37].
Conversely, data-driven techniques harness the power of empirical data patterns, employing advanced machine learning algorithms and statistical methods to discern anomalies and classify faults. The basis of these methods is the analysis of historical data, feature extraction, and the application of various classification algorithms.
Rule-based techniques utilize expert knowledge and predefined rules to detect and diagnose faults. These techniques often involve the use of expert systems, fuzzy logic, and rule-based reasoning.
The classification into these three distinctive groups underscores the diversity and sophistication of strategies deployed to fortify fault detection and classification mechanisms. This diversity caters to the multifaceted requirements across different domains, such as engineering, manufacturing, and process control, reflecting the dynamic nature of fault detection methodologies [38].

2.7.1 Model-based techniques

Model-based algorithms represent a category of techniques grounded in the utilization of theoretical models and expert knowledge to formulate decisions or predictions [39]. In the realm of fault detection and diagnosis, these algorithms play a crucial role in identifying and diagnosing faults in rotating machines by capitalizing on a comprehensive understanding of system dynamics. The application of model-based algorithms often encompasses the integration of mathematical models, physical simulations, or statistical representations to capture and interpret the intricate behavior of the system. Widely employed across diverse domains such as machine learning, signal processing, and fault detection, model-based algorithms prove particularly advantageous in situations where the system's behavior is well-understood and can be precisely modeled. Their versatility and efficacy make them instrumental in enhancing decision-making processes and predictive capabilities, contributing significantly to advancements in fault detection

methodologies.
Model-based algorithms play a crucial role in fault detection and diagnosis for rotating machines by capitalizing on a comprehensive understanding of system dynamics. These algorithms are widely used in various applications, including condition monitoring strategies for fault diagnosis in rotating machines [40] [41], A model-based approach to the detection and diagnosis of mechanical faults in rotating machinery is studied in [42] which utilizes a theoretical model of the system to detect and diagnose faults.
Furthermore, a model-based fault diagnosis technique for rotating machinery is proposed in [43] which employs a convolutional LSTM, Fast Fourier, and continuous wavelet transforms for fault detection and diagnosis. This technique leverages the theoretical model of the system to detect and diagnose faults by analyzing the vibration signals of the rotating machinery.
Moreover, a deep-learning-based data-driven fault diagnosis technique for rotating machinery is presented in [44], which uses a convolutional neural network (CNN) to classify the faults based on the vibration signals. This technique utilizes a wide three-axis vibration signal input as a high definition 1D image, enabling high classification accuracy.

Furthermore, model-based algorithms are highly effective in scenarios where the underlying physics or dynamics of the system are well-established. By leveraging this domain knowledge, these algorithms can provide valuable insights into the root causes of faults and deviations in system behavior. This not only enables accurate fault diagnosis but also facilitates proactive maintenance and system optimization. Additionally, the integration of model-based algorithms with real-time monitoring systems allows for continuous assessment of equipment health, enabling early detection of potential issues and preventing costly downtime.
In the context of machine learning, model-based algorithms are essential for developing predictive models that align with the underlying principles governing the system. This approach ensures that the generated models are interpretable and align with known physical laws, making them suitable for applications where transparency and explainability are paramount.

2.7.2 Data-driven techniques

Data-driven techniques in artificial intelligence (AI) are a category of approaches that rely on large datasets to learn models from data, often using machine learning techniques. These techniques are widely used in diverse domains such as machine learning, signal processing, and fault detection. They are particularly advantageous in situations where the system's behavior is complex, non-linear, and changing, and where the underlying physics or dynamics of the system are not well-established. Data-driven techniques encompass a wide range of methodologies, including supervised learning, unsupervised learning, and reinforcement learning. Supervised learning involves training a model on labeled data, where the input and output pairs are known. Unsupervised learning involves training a model on unlabeled data, where the model must identify patterns and relationships in the data. Reinforcement learning involves training a model to make decisions based on feedback from the environment. The application of data-driven techniques often encompasses the integration of various techniques and methodologies, such as fuzzy logic, fuzzy and rough sets for handling uncertainty, neural networks for approximating functions, global optimization and evolutionary computing, statistical learning theory, and Bayesian methods. These techniques have found applications in various fields, including economics, customer relations management, financial services, medicine, and the military, among others. Machine learning, a subfield of artificial intelligence, is closely related to data-driven modeling as it also focuses on using historical data to create models that can make predictions. In , Zhao discusses the significance of data-driven fault diagnosis in ensuring the reliable operation of rotating machinery. It highlights the reliance on data-driven approaches for fault diagnosis, emphasizing the role of these methods in maintaining the operational integrity of machinery.

In [45], Mushtaq et al present a comprehensive review of deep learning-aided data-driven fault diagnosis for rotatory machines, focusing on bearing fault diagnosis. It delves into the use of deep learning algorithms like Convolutional Neural Networks (CNN) for fault diagnosis, showcasing the advancements in this field and the potential for high classification accuracy.

In [46], Calabrese et al discuss a data-driven fault detection and diagnosis method based on deep representation clustering for rotating machinery. This method utilizes unsupervised data to enhance fault diagnosis in rotating machinery, showcasing the challenges and opportunities in real-world scenarios.

zhang et al in [47], address rotating machinery fault detection and diagnosis based on deep domain adaptation. It highlights the challenges of collecting large-scale supervised data for training deep networks in mechanical fault detection and emphasizes the importance of data-driven approaches in this context.

Data-driven techniques offer several advantages, including the ability to handle complex, non-linear, and changing systems, adaptability to new patterns, and the ability to handle uncertainty and unpredictability. They are also well-suited for handling large datasets and can provide unmatched data efficiency. However, they may require extensive labeled data and may struggle with highly complex or inadequately captured system behavior.

2.7.3 Rule-based techniques

Rule-based techniques in artificial intelligence (AI) are a category of approaches that rely on a set of predefined rules to make decisions or predictions. These rules are often created by experts or through trial and error and are used to represent domain-specific knowledge. Rule-based systems are widely used in diverse domains such as machine learning, signal processing, and fault detection. They are particularly advantageous in situations where the system's behavior is well-understood and can be precisely modeled. A typical rule-based system has four basic components: a list of rules or rule base, an inference engine or semantic reasoner, a database of facts, and a user interface. The rules typically take the form of an 'IF:THEN expression', where an individual rule is not in itself a model, since the rule is only applicable when its condition is satisfied. Therefore, rule-based machine learning methods typically comprise a set of rules, or knowledge base, that collectively make up the prediction model.

In [48], Zhou et al present a knowledge base system for fault detection and diagnosis of rotating equipment, showcasing a rule-based fault detection system designed to assist mechanics and engineers in dealing with fault diagnosis in rotating machinery systems. This system provides a structured approach to fault diagnosis, enhancing the efficiency and accuracy of fault identification processes

In [49], they introduced a rule-based intelligent method for fault diagnosis of rotating machinery, aiming to empower non-experts to conduct diagnosis operations effectively. This method focuses on intelligent fault identification in rotating machinery systems, emphasizing the importance of rule-based approaches in simplifying fault diagnosis procedures and in [50], Dou et al discuss a rule-based classifier ensemble for fault diagnosis of rotating machinery, highlighting the integration of base classifiers using an improved weighted voting technique. This method is praised for its transparent decision-making process, short diagnosis time, high reliability, and easy maintenance, showcasing a significant increase in accuracy compared to individual base classifiers for SKF6203 bearings.

Rule-based techniques have several advantages, including interpretability, transparency, and ease of use. They are particularly useful in situations where the decision-making process needs to be transparent and explainable, such as in medical diagnosis or legal decision-making. Rule-based systems are also easy to modify and update, as new rules can be added or existing rules can be modified to reflect changes in the system. In contrast to rule-based techniques, data-driven techniques rely on large datasets to learn models from data, often using machine learning techniques. These techniques are particularly advantageous in situations where the system's behavior is complex, non-linear, and changing, and where the underlying physics or dynamics of the system are not well-established. Data-driven techniques encompass a wide range of methodologies, including supervised learning, unsupervised learning, and reinforcement learning.

2.8 Comparison of Model-based, Data-driven, and Rule-based Techniques

Fault detection and diagnosis are critical processes across diverse domains, necessitating sophisticated methodologies for accurate anomaly identification and classification. These techniques fall into two primary categories: the data-driven and model-driven techniques, each distinguished by fundamental principles and applications.

The data-driven approach prioritizes the quality, governance, and management of data to address specific problem statements. By leveraging large datasets and relying on machine learning techniques, it extracts models directly from data, minimizing dependence on predefined models or algorithms. This approach proves practical in real-world applications, especially when dealing with high-quality and well-managed data. Its effectiveness shines in scenarios where relationships between variables may not be well-defined, such as in complex, non-linear, and changing systems.

Conversely, the model-driven approach focuses on developing new models and algorithmic manipulations to enhance performance. It relies on predefined models or algorithms, often crafted by experts or through trial and error. While efficient in some scenarios, this approach may lack adaptability to new or changing data and demands a deeper understanding of underlying problems and mathematical models. Its strength is in systems where behavior is well-understood and can be accurately modeled.

In terms of interpretability, model-based techniques offer clarity through explicit models, providing insights into system behavior. In contrast, data-driven techniques are often perceived as "black-box" models, prioritizing predictive accuracy over internal interpretability. Consequently, model-based techniques are preferable when interpretability is crucial, while data-driven techniques excel when the focus is on predictive accuracy, and understanding underlying mechanisms is less critical.

Handling changes presents another distinction: data-driven techniques demonstrate adaptability by learning from data patterns, making them well-suited for systems with evolving dynamics. In contrast, model-based techniques may require recalibration when significant changes occur due to their reliance on theoretical models that may need adjustments with shifting system dynamics.

Considering training requirements, data-driven techniques demand a substantial amount of labeled training data for algorithm training, learning from historical patterns to make predictions on new data. Conversely, model-based techniques necessitate a deep understanding of system dynamics and precise knowledge of parameters, involving the creation and fine-tuning of theoretical models. Thus, data-driven techniques require extensive labeled data, while model-based techniques rely on a profound comprehension of system dynamics.

In terms of generalization, data-driven techniques prove more adaptable to complex, non-linear, and changing systems, handling situations where variable relationships may be less well-defined. Conversely, model-based techniques may face challenges in complex systems if the theoretical model inaccurately captures intricacies. This disparity arises from the adaptive nature of data-driven techniques to new patterns, while model-based techniques may struggle with highly complex or inadequately captured system behavior.

The dependency on models is another key distinction: data-driven techniques operate without predefined models, learning patterns directly from available data. Conversely, model-based techniques depend on accurate system models, their effectiveness hinging on the fidelity of the theoretical models employed.

Based on the preceding comparison of fault diagnosis techniques, it is evident that data-driven methods surpass other categories, demonstrating superior adaptability to complex and non-linear systems. These techniques are intricately linked to the extracted features, and their effectiveness is heavily contingent upon the quality of the utilized data. Consequently, the significance of the feature selection process is underscored, a topic that will be further explored in the subsequent chapter. Moreover, delving deeper into the realm of feature selection unveils its pivotal role in enhancing model

Table 2.1: Comparison between Data-Driven, Model-Driven, and Rule-Based Approaches in Fault Detection and Diagnosis.

Aspect	**Data-Driven Approach**	**Model-Driven Approach**	**Rule-Based Approach**
Principles	Emphasizes data quality, governance, and management	Focuses on developing models and algorithmic manipulations for improved performance	Relies on expert knowledge and rules for fault detection and diagnosis
Data Usage	Leverages large datasets and machine learning techniques to learn models	Relies on predefined models or algorithms crafted by experts or through trial-and-error	May involve expert systems, fuzzy logic, and rule-based reasoning
Applicability	Practical in real-world applications with high-quality, well-managed data; Effective in complex, non-linear, and changing systems	Efficient in scenarios with well-understood behavior; May lack adaptability to new or changing data	Preferred when expert knowledge is required for fault diagnosis in less understood systems
Interpretability	Often perceived as "black-box" models, prioritizing predictive accuracy	Offers clarity through explicit models, providing insights into system behavior	May offer transparency through explicit rules and reasoning
Handling Changes	Demonstrates adaptability by learning from data patterns, well-suited for systems with evolving dynamics	May require recalibration when significant changes occur due to reliance on theoretical models	May require updates to rules and reasoning when significant changes occur in the system
Training Requirements	Requires a substantial amount of labeled training data for algorithm training, learning from historical patterns	Necessitates a deep understanding of system dynamics and precise knowledge of parameters for model creation	May require expert knowledge for creating and maintaining rules and reasoning
Generalization	More adaptable to complex, non-linear, and changing systems, handling less well-defined variable relationships	May face challenges if the theoretical model inaccurately captures intricacies in complex systems	May face challenges if the rules and reasoning inaccurately capture intricacies in complex systems
Dependency on Models	Operates without predefined models, learning patterns directly from available data	Depends on accurate system models, effectiveness hinging on the fidelity of theoretical models	Depends on accurate and up-to-date expert knowledge and rules

performance, refining predictive accuracy, and streamlining computational efficiency.

2.9 Conclusion

In this chapter, we have thoroughly explored the various fault detection categories that currently exist, providing a detailed analysis of their performance and effectiveness. We have delved into the intricacies of each category, examining their strengths, weaknesses, and suitability for different applications. Through this examination, we aim to provide readers with a comprehensive understanding of the landscape of fault detection techniques.

In the next chapter, our focus will shift to introducing the common general process adopted in our study. We will outline each step of this process in detail, providing insights of each stage. Additionally, we will introduce the technical tools and resources utilized in each step of the procedure, offering readers a glimpse into the practical implementation of our approach. By elucidating the process and tools employed, we aim to provide readers with the necessary foundation to comprehend and engage with the subsequent chapters of our study effectively.

Chapter 3

Signal processing and feature selection techniques

3.1 Introduction

In this chapter, we will explore the fundamental techniques of signal processing and feature selection, which play a crucial role in enhancing the accuracy and efficiency of our procedure. Signal processing allows us to refine and manipulate raw data, ensuring it is suitable for further analysis, while feature selection enables us to identify the most relevant variables, improving model performance and reducing complexity. By thoroughly understanding and implementing these steps, we lay the groundwork for more advanced methodologies in the subsequent sections, ultimately contributing to the overall success of our approach.

3.2 Signal processing

A signal is a functional representation that serves as a valuable conduit for conveying intricate insights into the intricate workings of a system or delineating the distinctive attributes associated with a particular phenomenon. A signal encapsulates the essence of an observed phenomenon, including its dynamics and nuances. On the other hand, signal processing is an active and dynamic endeavor that engages with an initial input signal, operates on it with a series of operations, and ultimately yields a refined and transformed output signal. This process forms a crucial bridge between raw data and actionable information, enabling us to extract meaningful knowledge, enhance signal quality, or facilitate specific outcomes based on the input signal's characteristics.
Real-world signals, are inherently non-stationary and exhibit significant complexity [51]. Unlike stationary signals, which have constant statistical properties over time[52], non-stationary signals undergo changes in their statistical characteristics, making them challenging to analyze. This inherent non-stationarity represents an obstacle in detecting concealed anomalies within the signals. Anomalies, representing deviations from expected patterns or behaviors, can be critical indicators of underlying issues or malfunctions. However, these non-stationary complex signals consist of multiple components, i.e. each component can vary over time in amplitude, phase, and frequency. It is possible to view each component as an oscillatory mode characterized by amplitude modulation and frequency modulation (AM-FM). As a result, a complex signal with multiple components is a combination or superposition of various AM-FM components [53] as expressed in equation 3.2.

$$x(t) = \sum_{i=1}^{N} c_i(t) = \sum_{i=1}^{N} a_i(t)\cos[\phi_i(t)] \tag{3.1}$$

$$= \sum_{i=1}^{N} a_i(t)\cos[\omega_c t + \int \omega_i(t)dt] \tag{3.2}$$

where $a_i(t)$ represents the instantaneous amplitude, $\phi_i(t)$ represents the instantaneous phase, ω_c denotes the carrier frequency, and $\omega_c + \omega_i(t) = \dot{\phi}_i(t)$ indicates the instanta-

neous frequency [53].
This superposition characteristic has led to the concept of decomposing the signal into its elementary components, enabling the extraction of meaningful insights from complex signals. Hence, various signal decomposition techniques have been developed. These techniques aim to break down the complex signatures of signals into simpler components, enabling a more detailed and comprehensive analysis. Consequently, analysts can effectively discern the underlying patterns and structures hidden within the complexity. This process enhances the understanding of the signal and facilitates its monitoring and analysis in various applications.
Decomposing signals into their constituent components empowers analysts to investigate anomalies more effectively. By isolating and examining specific parts, analysts can focus on the relevant aspects of the signal and gain deeper insights into the underlying causes of anomalies. This comprehensive analysis enables more informed decision-making. An appropriate signal decomposition technique is required to extract relevant information which contributes to better outcomes [54]. Signal decomposition techniques fall into three main categories: time-domain decomposition techniques, frequency-domain techniques, and time-frequency domain techniques.

3.2.1 Time-domain techniques

Time-domain techniques are foundational tools for understanding signal behavior over time. Within this domain, several techniques serve the purpose of decomposing signals into their constituent parts or analyzing specific aspects. These decomposition techniques are pivotal in extracting meaningful information from complex signals. Here are some common time-domain decomposition techniques:

1. **Empirical Fourier Decomposition (EFD)**: EFD is an accurate signal decomposition method for nonlinear and non-stationary time series analysis. It decomposes a signal into its constituent frequency components, providing insights into the underlying modal information.

2. **Variational Mode Decomposition (VMD)**: VMD is a data-driven signal decomposition technique that separates a signal into its constituent parts by exploiting its sparsity in a suitable function space. VMD has been applied in various fields, such as biomedical signal analysis and seismic signal analysis [54].

3. **Adaptive Signal Decomposition (ASD)**: ASD is a flexible and efficient signal decomposition method that adaptively decomposes a signal into its constituent parts. It can handle both stationary and non-stationary signals, making it a versatile tool for various applications **??**.

4. **Empirical Mode Decomposition (EMD)** EMD is a data-driven multiresolution technique employed to decompose a signal into components with physical significance. EMD proves useful in the analysis of non-linear and non-stationary

signals by autonomously segregating them into distinct components. The EMD algorithm iteratively extracts various resolutions directly from the data, without relying on fixed frequency filters. These extracted components in EMD are denoted as intrinsic mode functions (IMF). EMD has found applications in diverse fields, including biomedical data analysis, power signal analysis, and the examination of seismic signals.

In this signal decomposition category, we will focus on the Empirical Mode Decomposition since it is one of the most frequent time domain signal decomposition techniques utilized in condition monitoring, moreover, we are using it in our experiments.

3.2.1.1 Empirical Mode Decomposition (EMD)

In 1998, Huang and his collaborators introduced Empirical Mode Decomposition (EMD), a groundbreaking approach to signal analysis that stands apart from traditional methods. What distinguishes EMD is its remarkable adaptability. Unlike conventional techniques that rely on predefined function bases, EMD autonomously adjusts to suit the unique characteristics of the analyzed signal, denoted as $f(t)$. EMD's core objective remains consistent: to break down a signal into a finite sum of $N+1$ Intrinsic Mode Functions (IMFs), expressed as:

$$f(t) = \sum_{0}^{N} f_k(t) \tag{3.3}$$

These IMFs represent specific components of the original signal, enabling a comprehensive and meaningful decomposition.

An IMF, or Intrinsic Mode Function, can be described as a function that incorporates both amplitude modulation and frequency modulation, typically following this format:

$$f_k(t) = F_k(t)\cos(\phi(t)), \tag{3.4}$$

where: $F_k(t), \phi'_k(t) > 0$ for all t.
The primary assumption underlying EMD is that f_k and ϕ'_k change at a slower rate compared to ϕ_k. The Intrinsic Mode Function (IMF) f_k exhibits characteristics akin to those of a harmonic component. Initially, Huang and colleagues employed a purely algorithmic approach to extract these IMFs.

Envelopes Detection

To identify candidates for Intrinsic Mode Functions (IMFs), the first step is to compute the upper envelope, represented as $\phi(t)$, and the lower envelope, denoted as $\psi(t)$. This is accomplished through cubic spline interpolation, using the maxima and minima points of f. Subsequently, the mean envelope is derived by calculating:

$$m(t) = \frac{\overline{f}(t) + \underline{f}(t)}{2} \tag{3.5}$$

A visual representation of envelope detection is depicted in Figure 3.1.

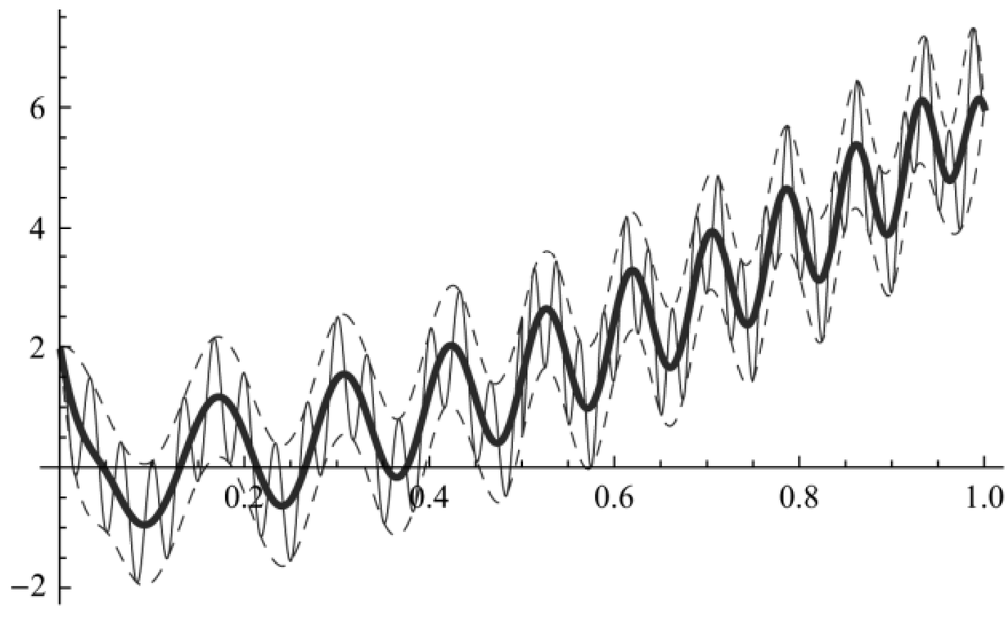

Figure 3.1: Envelopes Detection

Candidate Selection for IMFs

The candidate, $r_1(t)$, as illustrated in Figure 3.2, typically does not exhibit the characteristics of an IMF. A suitable candidate can be obtained by repeating the same procedure for r_1 and the subsequent r_k. The final retained IMF is denoted as $f_1(t) = r_n(t)$. Subsequently, the next IMF is obtained by applying the same algorithm to the signal $f(t) - f_1(t)$. The remaining IMFs can be calculated by applying this algorithm iteratively to the consecutive residues.

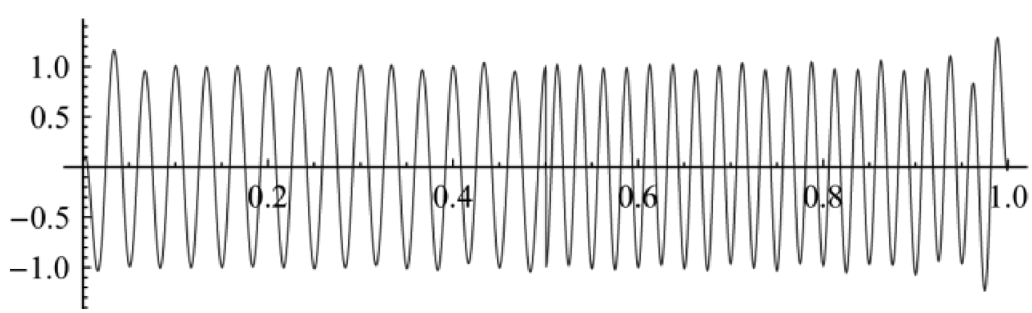

Figure 3.2: First IMF Candidate, r_1

Challenges and Adaptability

This algorithm showcases a remarkable ability to effectively capture the non-stationary aspect of the original function. However, its primary limitation stems from its reliance on an ad-hoc procedure. In other words, the EMD procedure is designed for a specific purpose and lacks a generalized model or framework that can be universally applied to all scenarios. This characteristic makes the formulation of a precise mathematical model a challenging endeavor. As a result, gaining a thorough understanding of the fundamental principles of EMD can be a complex task. For instance, complications may arise when working with signals that include noise. difficulties may emerge when dealing with signals that contain noise.

3.2.2 Frequency-domain techniques

Frequency domain techniques are a set of indispensable methods employed for signal processing and analysis purposes. In contrast to their time domain counterparts, these techniques focus on revealing and manipulating the spectral properties of signals, thus shedding light on the intricate world of frequencies. This approach is invaluable in diverse applications across science and engineering, offering profound insights into signals' fundamental constituents and attributes. Moreover, frequency domain techniques are instrumental in spectrum analysis, facilitating the comprehension of a signal's frequency content. These methods also find application in harmonic analysis, contributing to the identification of the fundamental frequency and harmonics in periodic signals. In cases where multiple signals intertwine, the frequency domain lends itself to coherence analysis, a technique that assesses the degree of relatedness between signals within this spectral realm. They are an essential toolkit, resonating across domains such as audio signal processing, image analysis and communication systems. They not only empower engineers and scientists to decipher the spectral signatures of signals but also drive innovations in noise reduction, feature extraction, and modulation and demodulation.

3.2.2.1 Fourier transform

The Fourier transform is a fundamental mathematical technique with wide-ranging applications across numerous disciplines, encompassing signal processing, mathematics, physics, engineering, and many more. Its name pays homage to the pioneering French mathematician and physicist, Jean-Baptiste Joseph Fourier. This transformative mathematical concept serves as a powerful tool for the analysis of functions and signals, unlocking a profound understanding of their frequency constituents.
In the most accessible terms, the Fourier transform performs a remarkable feat: it takes a time-domain signal, essentially a representation of how a phenomenon changes over time, and elegantly transmutes it into a frequency-domain counterpart. In this transformed representation, the data is no longer conveyed in terms of time but rather by revealing the presence of specific frequencies within the signal. In essence, it unveils how much of each frequency component contributes to the overall character of the signal.

The significance of this transformation lies in its practicality and versatility. It enables researchers, engineers, and scientists to delve into the spectral content of a signal, effectively dissecting it into its constituent frequencies. This capability is invaluable for tasks ranging from filtering and enhancing signals to decoding complex phenomena in the realms of acoustics, image processing, quantum mechanics, and various other domains.

The mathematical foundation of the continuous Fourier transform is an elegant expression that captures the essence of how a time-domain signal can be transformed into its frequency-domain counterpart. The formula at the heart of this transformation is as follows:

$$F(\omega) = \int_{-\infty}^{\infty} f(t) \cdot e^{-j\omega t} \, dt \tag{3.6}$$

Breaking this down, each element plays a pivotal role in unraveling the spectral secrets hidden within the signal:

- $F(\omega)$: This represents the result of the transformation, a complex-valued function of angular frequency (ω). It is the gateway to understanding the frequency components that constitute the original signal. The complex nature of $F(\omega)$ means that it does not only reveal the amplitude of these components but also their phase relationships. This intricate detail provides a comprehensive picture of how different frequencies contribute to the signal's overall character.

- $f(t)$: This denotes the time-domain signal itself. It is a function of time (t) that encapsulates the behavior of a physical or mathematical system over a continuous time interval. The Fourier transform allows us to transition from this time-dependent representation to one that focuses on frequency, which is often more insightful for analysis and processing.

- ω (omega): Angular frequency is a fundamental parameter that determines the frequency content of the signal in the frequency domain. It is defined as the rate of change of phase of the complex exponential function $e^{-j\omega t}$ concerning time. Different values of ω correspond to different frequencies, and as we vary ω, we traverse the entire spectrum of frequency components within the signal.

- $e^{-j\omega t}$: This component is a complex exponential function, a key element in the transformation process. The complex exponential contains information about the frequency being analyzed (ω) and its phase. By multiplying the time-domain signal $f(t)$ with this complex exponential at various ω values, we capture the signal's frequency components and their phase relationships. This step is central to extracting the underlying frequency information.

Fast Fourier transform:

The Fast Fourier Transform (FFT) stands as a cornerstone in signal processing, offering an efficient means to compute the Discrete Fourier Transform (DFT) and its inverse. This algorithm plays a pivotal role in converting signals from their original domain, such as time or space, into the frequency domain. By decomposing a sequence of values into components of different frequencies, the FFT provides a powerful tool for analyzing the frequency content of signals [55]. Efficiency lies at the heart of the FFT, as it reduces the computational complexity of the DFT from $O(n^2)$ to $O(n \log n)$, where n represents the data size. This efficiency is particularly advantageous when dealing with large datasets, enabling swift processing and analysis [56]. The FFT's ability to transform signals into the frequency domain is fundamental for tasks like filtering, spectral analysis, and feature extraction. By revealing the frequency components of a signal, the FFT empowers researchers and engineers to gain insights into the underlying characteristics of the data being analyzed. In various fields, including engineering, telecommunications, audio processing, image processing, and scientific research, the FFT finds extensive applications. Its versatility makes it indispensable for tasks such as signal analysis, system identification, image compression, and spectral analysis. The algorithm's significance extends to real-time processing applications, where speed is paramount. Industries such as audio processing, radar systems, and telecommunications benefit from the FFT's efficiency in swiftly analyzing signals and making real-time decisions based on frequency domain information. Moreover, the FFT enables detailed spectral analysis, allowing for the identification of dominant frequencies, anomaly detection, and feature extraction from signals. Its ability to efficiently compute the inverse DFT further enhances its utility, enabling the reconstruction of signals from their frequency components. In conclusion, the Fast Fourier Transform stands as a powerful and versatile algorithm in signal processing, offering a fast and efficient means to analyze signals in the frequency domain. Its impact across various industries and its efficiency in processing large datasets make it an indispensable tool for researchers and engineers alike.

3.2.3 Time-frequency domain techniques

Time-frequency analysis is a signal processing technique used to analyze and characterize signals simultaneously in both the time and the frequency domains. Unlike Fourier analysis, which provides information about a signal's frequency components but does not capture time-varying characteristics well, time-frequency analysis allows the signal's frequency content examination over time. This is useful for non-stationary signals.

Short time Fourier transform:

The Short-Time Fourier Transform (STFT) is a mathematical technique closely related to the Fourier transform, specifically designed to unveil the sinusoidal frequency components and phase information within localized sections of a signal, as these characteristics evolve [57] To compute the STFT, the standard approach involves the segmentation of a longer time-domain signal into shorter, uniform-length segments. Each of these segments is then subjected to its own Fourier transform operation, effectively uncovers the frequency spectrum peculiar to that segment. By performing this process across multiple time segments, the result is a dynamic depiction of how the signal's frequency content changes over time. This dynamic representation is commonly visualized as a spectrogram or waterfall plot, which provides a detailed, time-varying perspective on the signal's spectral characteristics. Such analyses are widely utilized in various fields, including audio and speech processing, music analysis, and image texture characterization, enabling researchers and engineers to gain valuable insights into how signals evolve both temporally and spectrally.

Wavelets

This section explores the intriguing and powerful concept of wavelets, which provide a versatile approach to approximating complex functions [58]. Wavelet analysis involves meticulously deconstructing intricate functions into manageable components known as basis functions. These basis functions, akin to puzzle pieces, are thoughtfully weighted and assembled to create a coherent representation of the original function. This methodology serves as a valuable tool, enabling us to decipher the intricacies of complex functions, enhance our understanding, and facilitate practical applications, including data compression [58].

While Fourier's pioneering work, employing sinusoidal functions as fundamental building blocks, advanced our comprehension of frequency composition, it faced inherent limitations when dealing with non-stationary signals. To address this limitation, the field of time-frequency representations (TFRs) emerged, offering a nuanced perspective on signals by considering not only their spectral content but also their temporal localization.

The transition from Fourier's frequency-based approach to TFRs marked a profound milestone in signal analysis. The introduction of the short-time Fourier transform (STFT) represented a decisive departure from Fourier's global perspective. This transformative approach allowed for the analysis of signals within localized time windows. Pioneers in the field, including luminaries like Dennis Gabor and Jean Ville, made indispensable contributions to this paradigm shift.

In the late 1970s, J. Morlet introduced a groundbreaking innovation—the concept of wavelets with compact support in both the time and frequency domains. In collaboration with A. Grossman, not only introduced these wavelets but also formalized the wavelet transform, thereby ushering in a new era of signal analysis. This pivotal transition from Fourier to STFT to wavelet analysis can be aptly described as revolutionary in the realm of signal processing and analysis.

Yves Meyer's observations in 1984 regarding the remarkable parallels between Morlet's work and Calderón's contributions were particularly noteworthy. His subsequent work in 1985 aimed at enhancing wavelet localization, a pursuit that would significantly impact the field. It is intriguing to note that J.O. Strömberg had independently discovered similar wavelets before these developments. It is essential to acknowledge that the foundation of orthonormal wavelets was laid by Alfred Haar in 1909, although it is worth noting that Haar wavelets while pioneering, had certain limitations in terms of frequency localization. Furthermore, it came to light that Haar's pioneering work had been expanded upon in the 1930s by Paul Levey, who delved into the realm of random signals of Brownian motion. Concurrently, Littlewood and Paley were engaged in exploring the localization of a function's contributing energies, contributing to the rich history of wavelet development [58].

During this period, Ingrid Daubechies, a former graduate student of Grossman at the Free University of Brussels, introduced the innovative concept of wavelet frames. This pioneering idea offered greater flexibility in selecting basis functions, albeit with some trade-offs in terms of redundancy. Alongside Stephane Mallat, she played a pivotal role in the transition from continuous to discrete signal analysis. Particularly in 1986, Mallat, a graduate student at the University of Pennsylvania, introduced the concept of multiresolution analysis (MRA) for discrete wavelet transform (DWT) in collaboration with Meyer. The concept involved the meticulous decomposition of a discrete signal into its dyadic frequency bands, accomplished through a series of lowpass and highpass filters, ultimately facilitating the computation of its DWT from the approximations at these various scales. Intriguingly, this concept had already been familiar to electrical engineers for nearly two decades under the name of quadrature mirror filters (QMF) and subband filtering, a realm pioneered by A. Croisier, D. Esteban, and C. Galand around 1976. Mallat's work seamlessly extended the concept of time localization to complement the well-established concept of frequency localization found in QMF and subband coding. Additionally, in 1988, with the development of Daubechies' orthonormal bases of compactly supported wavelets, the foundational principles of modern wavelet theory were firmly established.

In recent years, we have witnessed a significant wave of exploration in search of alternative wavelet basis functions and refinements in multiresolution analysis (MRA) algorithms. Notably, developments in compactly supported biorthogonal wavelets and wavelet packets have emerged as noteworthy milestones in the ongoing journey of advancing wavelet theory and its diverse applications [58].

Wavelet Overview

The Fourier series, which had its origins in the early 19th century, was a groundbreaking development in mathematical analysis. Initially devised for the study of continuous and periodic signals, it laid the foundation for a profound understanding of signal decomposition and representation. This revolutionary concept introduced the notion that complex functions could be expressed as a sum of simpler components, thus opening the door to a new realm of mathematical insight and practical applications.

In mathematical terms, the Fourier series is elegantly expressed as:

$$x(t) = \sum_{k=-\infty}^{\infty} c_k e^{j2\pi kt/T} \tag{3.7}$$

Here, $x(t)$ signifies the signal under scrutiny, T represents the signal's fundamental period, and c_k denotes the Fourier coefficients, given by:

$$c_k = \frac{1}{T} \int_{-T/2}^{T/2} x(t) e^{-j2\pi kt/T} dt. \tag{3.8}$$

These coefficients are obtained by integrating the product of the signal $x(t)$ and the complex exponential functions $e^{j2\pi kt/T}$ over one complete period, a process that provides the spectral representation of the signal.

However, it's important to note that the complex exponential functions associated with discrete frequencies of $2\pi jk/T$ do not have compact temporal support, as they extend infinitely in time. Nevertheless, they possess perfect frequency localization, manifesting as delta functions at their respective frequencies in the Fourier domain. As mentioned earlier, this inherent characteristic renders the Fourier representation unsuitable for the analysis of non-stationary signals. In simpler terms, due to their infinite temporal reach, complex exponentials offer a global perspective of the signal across time and only reveal the spectral components contained within it. Consequently, the Fourier representation cannot provide information about when these spectral components occur in time. While this limitation is not problematic when dealing with stationary signals, where all spectral components persist uniformly over time, it becomes a significant constraint when working with non-stationary signals that exhibit changes in their spectral characteristics over time. Unfortunately, many individuals, unaware of this limitation, have erroneously applied the Fourier representation to analyze non-stationary signals in practical scenarios, despite the fact that a majority of real-world signals, irrespective of their origin, display non-stationary behavior.

The Short-Time Fourier Transform (STFT) represented a crucial improvement, addressing the challenge of analyzing non-stationary signals by dividing them into relatively short, quasi-stationary segments and subsequently calculating the Fourier representation for each of these segments. This process is mathematically defined as:

$$S(\tau, f) = \int x(t) w(t - \tau) e^{-j2\pi ft} dt \tag{3.9}$$

Where $S(\tau, f)$ denotes the STFT at frequency f and time translation τ.

In this equation, $w(t)$ represents the windowing function, f is the frequency parameter, τ is the time translation parameter, and $*$ denotes the complex conjugate operator. Notably, for each frequency f, the temporal localization is achieved by segmenting $x(t)$

using $w(t-\tau)$, which is a windowing function centered at $t = \tau$. The Fourier transform of this segmented signal subsequently provides frequency localization, aligning with the strength of the Fourier transform.

However, this approach presents a drawback in that it maintains a consistent resolution for all frequencies due to its reliance on the same window for analyzing the entire signal. When dealing with signals that encompass high-frequency components over a short duration, optimal time resolution necessitates the use of a narrow window, which is compactly supported in time. It's important to note that narrow windows inherently result in wider frequency bands, consequently leading to diminished frequency resolution. Conversely, if the signal incorporates low-frequency components spanning a longer duration, a broader window must be employed to achieve superior frequency resolution, albeit at the cost of diminished time resolution.

The wavelet transform (WT) was developed to overcome this limitation by using windows of varying durations, allowing for variable time and frequency resolutions. The WT operates differently from the Short-Time Fourier Transform (STFT), first decomposing the signal into distinct frequency bands and subsequently analyzing them in the time domain. This process is mathematically represented by the continuous wavelet transform equation:

$$W(a,b) = \int_{-\infty}^{\infty} x(t)\psi^*_{a,b}(t)\,dt \tag{3.10}$$

Here, $\psi_{a,b}(t)$ represents the scaled and translated version of the mother wavelet $\psi(t)$, where the mother wavelet is also known as the basis function is the fundamental building block used in wavelet transform analysis. Different types of mother wavelets are designed to capture specific features or characteristics of signals or data. The choice of a mother wavelet depends on the particular application and the nature of the data being analyzed as shown in figure 3.3 and it is given:

$$\psi_{a,b}(t) = \frac{1}{\sqrt{a}}\psi\left(\frac{t-b}{a}\right) \tag{3.11}$$

In this equation 3.10, $a > 0$ and b are scale and translation parameters, respectively, ψ is the mother wavelet, and $W(a,b)$ is the continuous wavelet transform of $x(t)$. Equation 3.10 can be interpreted as an inner product of $x(t)$ with the scaled and translated versions of the basis functions ψ:

$$W(a,b) = \int x(t)\cdot\psi^*_{a,b}(t)\,dt \tag{3.12}$$

The scaled and translated versions of these basis functions are derived from a single prototype function, the mother wavelet. It's worth noting that the term 'wavelet' originates from the admissibility condition, which mandates that basis functions have finite support and an oscillatory nature, hence 'wavelet' (small wave).

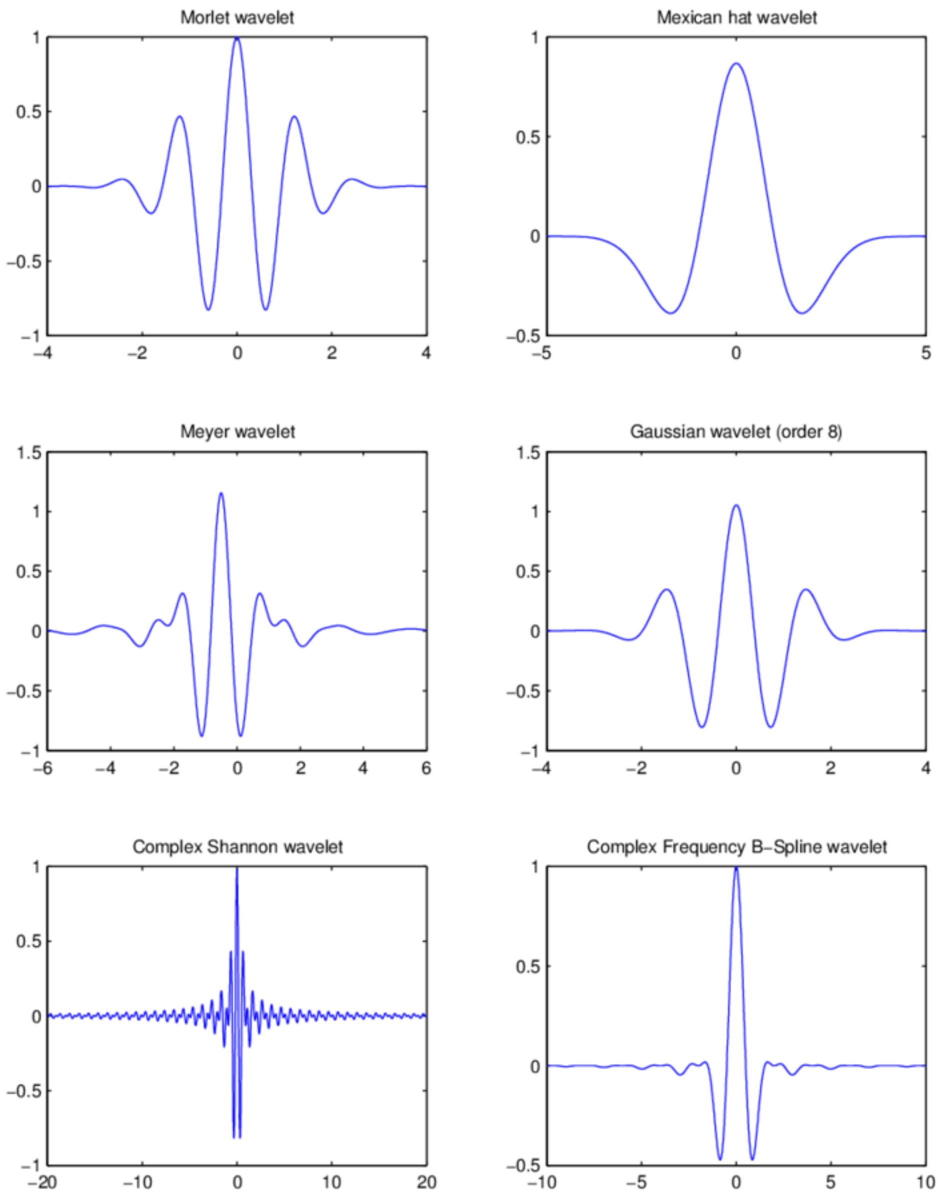

Figure 3.3: Visualization of various mother wavelet functions

To derive the Discrete Wavelet Transform (DWT), it is necessary to discretize the parameters a and b. Daubechies demonstrated that by discretizing with $a = 2^j$ and $b = 2^{jk}$, it leads to orthonormal basis functions for specific selections of ψ, often referred to as Daubechies wavelets. This discretization is expressed as follows:

$$\psi_{j,k}(t) = 2^{-\frac{j}{2}}\psi(2^{-j}t - k) \tag{3.13}$$

Mallat subsequently illustrated that Multiresolution Analysis (MRA) can be employed to obtain the DWT of a discrete signal. This process involves iteratively applying both lowpass and highpass filters and subsequently downsampling them by a factor of two. Figure 1 illustrates this procedure, featuring the highpass filter $g[n]$ and lowpass filter $h[n]$, along with discrete frequency bands for each level. At each level, this procedure computes the following equations:

$$y_k[n] = \sum_k x[n] \cdot h_k[n] \tag{3.14}$$

$$y_k[n] = \sum_k x[n] \cdot g_k[n] \tag{3.15}$$

Where:

$$h[N - 1 - n] = h[1 - n]$$

and N represents the total number of samples in $x[n]$.

The process of reconstructing the original signal can be achieved by reversing the previous steps or by performing the following computation:

$$x[n] = \sum_k (y_{low}[k] \cdot h_k[n] + y_{high}[k] \cdot g_k[n])$$

In this equation, y_{high} and y_{low} represent the outputs of highpass and lowpass filters, respectively, at each level. The summation is performed over all relevant k values. This procedure effectively combines the filtered outputs to reconstruct the original signal.

3.2.3.1 Maximal overlap discrete wavelet-packet transform

The Discrete Wavelet Transform (DWT) is a highly effective method for signal analysis because it allows the examination of an input signal in both the time and frequency domains. The DWT decomposes the input signal into approximation and detail coefficients. This process involves passing the input signal (x) through both a high-pass filter (h) and a low-pass filter (g), followed by down-sampling by a factor of 2 [59]. The low pass filter produces an approximated signal (a), and the high pass filter generates the detail coefficients (d). This iterative process can be performed multiple times, utilizing the approximated signal as input for further DWT decomposition. Each filter generates a set of coefficients that effectively represent and compress the original signal. As a result, the DWT decomposes the input signal into multiple sub-bands or levels. The DWT offers numerous benefits, including improved data compression, efficient signal reconstruction, and effective signal denoising [60]. However, one drawback of the DWT is its requirement for the sample size to be a power of 2 to perform the full transform effectively this limitation arises due to the down-sampling step involved in the DWT process. Furthermore, it exhibits limited frequency resolution when analyzing signals with high frequencies. [61].

The maximal overlap discrete wavelet packet transform (MODWPT) has been developed []. This technique does not only improve the frequency resolution, but it also liberates signal analysis from the constraints of specific sample sizes since it is a non-downsampling technique where each level of decomposition retains an equal number of wavelet coefficients. This design ensures the preservation of vital information related to bearing vibrations and maintains consistent cyclic timing between successive fault-induced impulses at various locations. Additionally, the MODWPT is an energy-conserving transformation, meaning that the total energy of the MODWPT coefficients corresponds to the energy of the original signal [62].

Let us consider a discrete time sequence $X = \{x_0, x_1, \ldots, x_{N-1}\}$ of N samples obtained by sampling a continuous-time signal $x(t)$ with a sampling frequency F_s. The MODWPT can be obtained by convolving X with a subset of the MODWPT filters. The low-pass scaling filter can be expressed as $\{g_m : m = 0, 1, \ldots, M-1\}$, and its quadratic mirror high-pass wavelet filter can be represented as $\{h_m : m = 0, 1, \ldots, M-1\}$, where M is the length of the filter and $M \leq N$. The filters g_m and h_m are related to each other as:

$$h_m = (-1)^m g_{M-m-1} \quad \text{or} \quad g_m = (-1)^m h_{M-m-1} \tag{3.16}$$

To maintain energy conservation, the MODWPT filters are scaled. The scaled filters are denoted as $\widetilde{g}_m = \frac{g_m}{\sqrt{2}}$ and $\widetilde{h}_m = \frac{h_m}{\sqrt{2}}$. The corresponding transfer functions, $\widetilde{G}(f)$ and $\widetilde{H}(f)$, can be defined as:

$$\widetilde{G}(f) = \sum_{m=0}^{M-1} \widetilde{g}_m e^{-j2\pi f m} \tag{3.17}$$

$$\widetilde{H}(f) = \sum_{m=0}^{M-1} \widetilde{h}_m e^{-j2\pi f m} \tag{3.18}$$

At the first level, $j = 1$, $W_{0,0}(= X)$ is circularly filtered by $\tilde{H}(f)$ and $\tilde{G}(f)$ to obtain correspondingly, the first level coefficients $W_{1,0} = \{W_{1,0,k} : k = 0, 1, ..., N-1\}$ and $W_{1,1} = \{W_{1,1,k} : k = 0, 1, ..., N-1\}$. Next, the filters $\tilde{h}_m$ and $\tilde{g}_m$ of the second level ($j = 2$) are expanded by adding one zero between the elements of filter coefficients to give $\tilde{g}_m = \{\tilde{g}_0, 0, \tilde{g}_1, 0, ..., \tilde{g}_{M-2}, 0, \tilde{g}_{M-1}\}$ and $\tilde{h}_m = \{\tilde{h}_0, 0, \tilde{h}_1, 0, ..., \tilde{h}_{M-2}, 0, \tilde{h}_{M-1}\}$, respectively. Moreover, their corresponding transfer functions are given by $\tilde{H}(2f)$ and $\tilde{G}(2f)$. Now, $W_{1,0}$ and $W_{1,1}$ are circularly filtered by $\tilde{H}(2f)$ and $\tilde{G}(2f)$ to give the second level vectors $W_{2,0}$, $W_{2,1}$, $W_{2,2}$, and $W_{2,3}$. For subsequent levels j of the transform, $2^{j-1}-1$ zeros are inserted between the elements of $\tilde{h}_m$ and $\tilde{g}_m$ whose transfer functions are given by $\tilde{H}(2^{j-1}f)$ and $\tilde{G}(2^{j-1}f)$, and the filtering process continues until the desired level Jo.

3.2.3.2 Empirical wavelet transform

The wavelet transform has garnered widespread adoption within the realm of signal analysis and processing, as noted in studies such as Sifuzzaman et al. (2009) . It has emerged as a potent and versatile instrument for scrutinizing signals that exhibit characteristics like nonlinearity and non-stationarity, effectively unveiling intricate patterns and anomalies within these signals. However, a notable limitation of the wavelet transform lies in its utilization of fixed basis functions, a constraint that curtails its adaptability to the multifaceted landscapes of real-world signals. These pre-defined basis functions are, at times, ill-suited to capture the intricate nuances and unique traits exhibited by various types of signals, thus impeding the full realization of the wavelet transform's potential, as highlighted in the research by Liu et al. (2019) [63]. In stark contrast, Empirical Mode Decomposition (EMD) offers a distinctive approach to signal decomposition. It employs a methodology that dissects a given signal into discrete oscillatory components referred to as modes, a concept introduced by Boudraa and Cappé (2004) [64]. These modes are instrumental in capturing and revealing specific characteristics, inherent behaviors, and unique features embedded within the original signal. EMD is celebrated for its remarkable capability to derive these fundamental functions directly from the original input signal, essentially providing a flexible and data-driven mechanism for signal decomposition, as emphasized in the study by Grasso et al. (2016) [65]. EMD excels in adapting to the distinctive attributes of each signal it encounters, making it a valuable tool for interpreting complex and diverse datasets across various domains. EMD has demonstrated its effectiveness in many applications across different fields. Its ability to accurately capture the inherent dynamics of complex signals has proven valuable in diverse domains [63]. Researchers have successfully employed EMD to extract meaningful information from signals that exhibit nonlinearity, non-stationarity, and transient behaviours. However, it is important to note that despite the achievements and practical applications of EMD, it still lacks a comprehensive mathematical theory [66]. The absence of a rigorous mathematical framework limits its theoretical understanding and places constraints on its widespread acceptance and adoption.

To address these constraints, Gilles introduced the empirical wavelets transform (EWT) [3], inspired by both the wavelet transform (WT) and empirical mode decomposition (EMD) to construct adaptive wavelets that can adjust or adapt to the characteristics of a particular signal unlike fixed basis wavelets, which have predefined shapes and properties, adaptive wavelets are intended to capture the local features and nuances of the signal being analyzed. They can extract finite AM-FM components, denoted as $f_k(t)$, or modes. These modes represent specific intrinsic behaviours or characteristics embedded within the original signal. Each mode provides valuable insights into the underlying dynamics of the signal. The relationship between the modes and the original signal can be expressed as follows:

$$f(t) = \sum_{k=0}^{N} f_k(t) \tag{3.19}$$

The key idea of the empirical wavelet transform (EWT) involves the establishment of a set of N wavelet filters, consisting of one low-pass filter and N-1 band-pass filters that correspond to the approximation and detail components, respectively [67].

We denote the real value signal and its corresponding Fourier spectrum as $f(t)$ and $f(\hat{\omega})$,respectively. Initially, the boundaries of each Fourier support in signal $f(\hat{\omega})$ are identified as per the following procedure. For the sake of convenience, our analysis is constrained to the interval $[0, \pi]$, which is subsequently partitioned into N sub-intervals Λ_n with $\Lambda_n = [\omega_n, \omega_{n+1}]$ and $U_{n=1}^{N}\ \Lambda_n = [0,\ \pi]$ as shown in 3.4, The process begins by computing the local maxima from the Fourier spectrum of $f(t)$. These local maxima serve as crucial reference points for identifying the boundaries. By locating the centres between two subsequent peaks, we divide the spectrum into segments, each representing a distinct frequency range. This step lays the foundation for constructing the empirical wavelets, which are essential for breaking down the signal into its constituent parts.

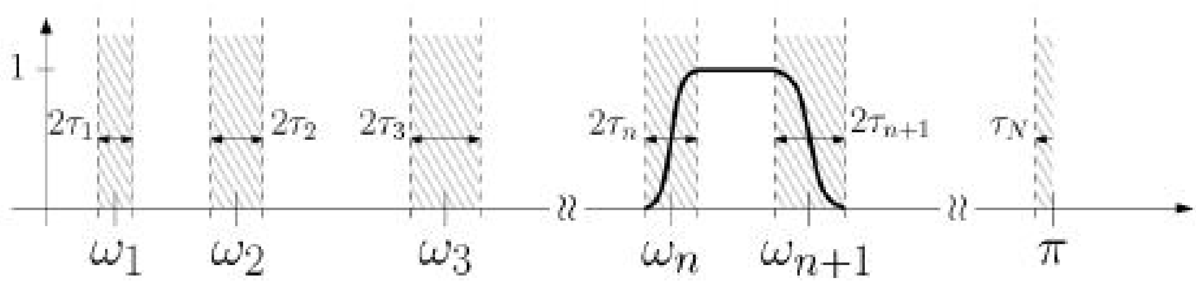

Figure 3.4: Segmenting the Fourier axis [3]

Expressions 3.20 and 3.21 define the empirical scaling function and the empirical wavelets, respectively [3]. They form the basis of the empirical wavelet transform and allow us to adaptively analyze different frequency components of the signal, providing valuable insights into its approximation and detail characteristics.

$$\hat{\phi}_n(\omega) = \begin{cases} 1 & \text{if } |\omega| \leq \omega_n - \tau_n \\ \cos\left[\frac{\pi}{2}\beta\left(\frac{1}{2\tau_n}\left(|\omega| - \omega_n + \tau_n\right)\right)\right] & \text{if } \omega_n - \tau_n \leq |\omega| \leq \omega_n + \tau_n \\ 0 & \text{otherwise} \end{cases} \tag{3.20}$$

$$\hat{\psi}_n(\omega) = \begin{cases} 1 & \text{if } \omega_n + \tau_n \leq |\omega| \leq \omega_{n+1} - \tau_{n+1} \\ \cos\left[\frac{\pi}{2}\beta\left(\frac{1}{2\tau_{n+1}}\left(|\omega| - \omega_{n+1} + \tau_{n+1}\right)\right)\right] & \text{if } \omega_{n+1} - \tau_{n+1} \leq |\omega| \leq \omega_{n+1} + \tau_{n+1} \\ \sin\left[\frac{\pi}{2}\beta\left(\frac{1}{2\tau_n}\left(|\omega| - \omega_n + \tau_n\right)\right)\right] & \text{if } \omega_n - \tau_n \leq |\omega| \leq \omega_n + \tau_n \\ 0 & \text{otherwise} \end{cases} \tag{3.21}$$

Expression 3.20 represents the empirical scaling function, denoted as $\hat{\phi}_n(\omega)$. This function captures the scaling properties of the wavelet transform and allows us to explore the low-frequency components of the signal. The function $\hat{\phi}_n(\omega)$ is defined piecewise based on the frequency scope. If the frequency $|\omega|$ is less than or equal to $\omega_n - \tau_n$, the scaling function is set to 1. If the frequency falls within the range of $\omega_n - \tau_n$ to $\omega_n + \tau_n$, the function takes the form of a cosine function with a parameter β that controls the shape. Outside these ranges, the scaling function is zero.

Expression 3.21 defines the empirical wavelets, denoted as $\hat{\psi} n(\omega)$. These wavelets capture the details and high-frequency components of the signal. Similar to the scaling function, the empirical wavelets are defined piecewise. The function $\hat{\psi} n(\omega)$ takes the value 1 in the frequency range $\omega_n + \tau_n$ to $\omega n + 1 - \tau n + 1$. Within the range of $\omega_{n+1} - \tau_{n+1}$ to $\omega_{n+1} + \tau_{n+1}$, it is a cosine function, and within the range of $\omega_n - \tau_n$ to $\omega_n + \tau_n$, it becomes a sine function. Outside these ranges, the wavelet function is zero.

The function $\beta(x)$ is an arbitrary $C^K([0, 1])$, such that

$$\beta(x) = \begin{cases} 0 \quad \text{if} \quad x \leq 0 \quad \text{and} \quad \beta(x) + \beta(x-1) = 1 \quad \forall x \in [0,1] \\ 1 \quad \text{if} \quad x \geq 1 \end{cases} \tag{3.22}$$

The expression 3.23 is frequently employed among various functions that fulfil the given properties.

$$\beta\left(x\right) = x^4\left(35 - 84x + 70x^2 - 20x^3\right) \tag{3.23}$$

τ_n is half the length of the transition phase, $\tau_n = \lambda\omega_n$. To guarantee the absence of overlap between two successive transition regions, parameter λ must satisfy the following equation:

$$\lambda < min_n \left(\frac{\omega_{n+1} - \omega_n}{\omega_{n+1} + \omega_n}\right) \tag{3.24}$$

Equation 3.24 limits the maximum value that *lambda* can take, ensuring that the transition regions between different wavelets do not overlap and thus preserving the integrity of the wavelet decomposition process.

The EWT $\mathcal{W}^{\varepsilon}{}_{f}\left(n,t\right)$ is defined as for the classical WT, the coefficients are obtained by the inner product of the original signal and the empirical wavelets [3].

$$\mathcal{W}^{\varepsilon}{}_{f}\left(n,t\right) = \langle f, \psi_n \rangle \; = \int f\left(\tau\right) \overline{\psi_n\left(\tau - t\right)}\, d\tau \tag{3.25}$$

$$= \left(\hat{f}\left(\omega\right) \overline{\hat{\psi}_n\left(\omega\right)}\right)^{\vee} \tag{3.26}$$

The approximation coefficients are calculated using equation 3.27.

$$\mathcal{W}^{\varepsilon}{}_{f}\left(0,t\right) = \langle f, \phi_1 \rangle \; = \int f\left(\tau\right) \overline{\phi_1\left(\tau - t\right)} d\tau \tag{3.27}$$

So the signal's empirical modes f_k can be calculated as follows:

$$f_0\left(t\right) = \mathcal{W}_{f}^{\varepsilon}\left(0,t\right) * \phi_1\left(t\right) \tag{3.28}$$

$$f_k\left(t\right) = \mathcal{W}_{f}^{\varepsilon}\left(k,t\right) * \psi_k\left(t\right) \tag{3.29}$$

The original signal can be reconstructed using the equation 3.30.

$$f\left(t\right) = \mathcal{W}^{\varepsilon}{}_{f}\left(0,t\right) * \phi_1\left(t\right) + \sum_{n=1}^{N} \mathcal{W}_{f}^{\varepsilon}\left(n,t\right) * \psi_n\left(t\right) \tag{3.30}$$

3.3 Feature extraction and selection

The feature selection stage serves as a crucial initial step aimed at improving the overall quality of the underlying clustering process. Not all features carry the same level of relevance in the classification process, as some may introduce more noise than others. Hence, it's crucial to eliminate these noisy and irrelevant features. It's important to note that feature selection and dimensionality reduction are closely intertwined. In feature selection, we pick specific subsets of the original features, while dimensionality reduction, as seen in techniques like principal component analysis, may involve creating linear combinations of features to refine the selection process.

The classification accuracy heavily relies on the quality of the extracted features. Therefore, selecting relevant and informative attributes that accurately describe the state of the diagnosed element is crucial in the diagnosis process. Also, a precise and deliberate selection of features can enhance the performance of the inductive learner. This improvement can be observed in various aspects, such as accelerated learning speed, increased capacity to generalize effectively, and the creation of a more simplified and manageable model. There are many feature selection techniques to optimize the feature

set for training classification models, achieve efficient data reduction, and extract the most salient features from the raw data.
A typical feature selection process comprises four essential steps to identify the most relevant subset of features. These steps include subset generation, subset evaluation, stopping criterion, and result validation as demonstrated in figure 3.5. Subset generation refers to the process of creating potential feature subsets for evaluation, employing a specific search strategy. Each candidate subset is then assessed and compared to the previously identified best subset using a designated evaluation criterion. The iteration of subset generation and evaluation continues until a predetermined stopping criterion is met [68]. The commonly employed stopping criteria are as follows:

- Completion of the search: This criterion indicates that the search process has been exhaustively performed, considering all possible subsets of features.
- Feature performance plateau: This criterion comes into play when subsequent addition or deletion of any feature fails to yield a noticeable improvement in the performance or evaluation metric. It suggests that the current subset under evaluation represents a peak or plateau in terms of performance, and further modifications are unlikely to lead to significant enhancements.
- Achievement of a sufficiently good subset: This criterion focuses on reaching a subset of features that meets predefined criteria or performance thresholds. It ensures that the feature selection process terminates once a subset that satisfies the desired level of quality or performance has been identified.
- Specified limit or bound: This criterion involves setting a predetermined limit on the number of iterations or the number of selected features. Once this limit is reached, the feature selection process concludes, even if an optimal solution has not been achieved.

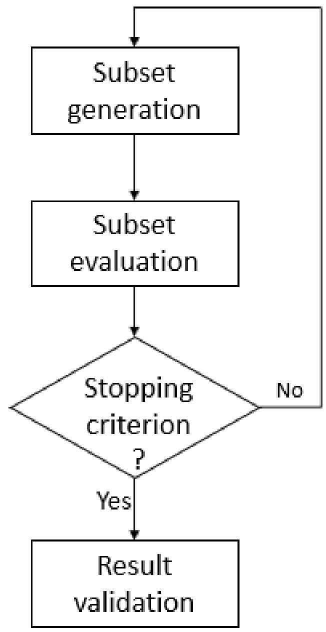

Figure 3.5: Feature selection procedure

Feature selection encompasses two primary approaches: individual evaluation and subset evaluation. Individual evaluation also referred to as feature ranking, involves assigning weights to individual features based on their relevance degrees [69]. In contrast, subset evaluation generates candidate feature subsets using a specific search strategy. Additionally, feature selection methods can be further categorized into four models: filters, wrappers, embedded, and hybrid methods[70].

3.3.1 Filter techniques

Filter feature selection techniques are methods used to select relevant features independently of a specific learning algorithm. These techniques assess the features based on their intrinsic properties, such as information, distance, consistency, similarity, and statistical measures [1]. Filter methods rank or score features individually without considering the interaction between them. Table 3.1 summarizes the most common filter methods

Examples of commonly used filter feature selection techniques include correlation-based feature selection, chi-squared test, mutual information, and variance thresholding. These techniques provide a preliminary filter to identify the most informative features before applying more complex and computationally expensive learning algorithms. Filter methods can be divided into univariate and multivariate methods. Univariate feature filters analyze and typically rank individual features, while multivariate filters assess an entire subset of features. The feature subsets' generation in multivariate filters relies on the chosen search strategy. Four common starting points for feature

subset generation: are forward selection, backward elimination, bidirectional selection, and heuristic feature subset selection.
Forward selection starts with an empty feature set and gradually adds one or more features. Backward elimination, on the other hand, begins with the entire feature set and considers removing one or more parameters. Bidirectional search simultaneously explores larger and smaller feature subsets by starting from both an empty set and a complete set. Finally, heuristic feature subset selection utilizes a specific heuristic approach to determine the most relevant feature subset.
The search strategies used in multivariate filters are divided into exponential, sequential, and randomized algorithms. Exponential algorithms evaluate subsets that exponentially increase in size with the number of features. Sequential algorithms add or remove features one at a time, potentially leading to local minima. Randomized algorithms incorporate randomness into the search process, which helps avoid local minima [71].

Table 3.1: Common filter feature selection methods [1]

Name	Filter Class	Applicable to Task
Information gain	Univariate	Classification
Gain ratio	Univariate	Classification
Correlation	Univariate	Regression
Chi-square	Univariate	Classification
Minimum redundancy, maximum relevance	Multivariate	Classification, Regression
Correlation-based feature selection (CFS)	Multivariate	Classification, Regression
Fast correlation-based filter (FCBF)	Multivariate	Classification
Fisher score	Univariate	Classification
Relief and ReliefF	Univariate	Classification, Regression
Feature selection for sparse clustering	Multivariate	Clustering
Localized Feature Selection Based on Scatter Separability (LFSBSS)	Multivariate	Clustering
Multi-Cluster Feature Selection (MCFS)	Multivariate	Clustering
Feature weighting Kmeans	Multivariate	Clustering
ReliefC	Univariate	Clustering

Filter methods utilize ranking techniques to assess the importance of each variable in the database. These methods have garnered popularity due to their simplicity and effectiveness in real-world applications. The assessment starts by ordering the variables according to a suitable criterion and then eliminating all variables falling below a predetermined threshold. Filter methods are known as ranking methods because they work by filtering out variables deemed to be irrelevant before classification. A crucial aspect of a distinguishing feature is that the variable should possess relevant information for the different classes within the database [72].

Filter methods have several advantages, including computational efficiency, the ability to avoid overfitting, and their proven effectiveness on specific databases. Unlike learning algorithms, filter methods used in feature ranking are not biased, which prevents data from being manipulated to fit a particular algorithm. However, one potential drawback of ranking methods is that they may not always yield an optimal subset of features. Additionally, some ranking methods, such as the Pearson correlation criteria and mutual information, do not distinguish between highly correlated variables, potentially resulting in a smaller subset being sufficient.

3.3.2 Embedded techniques

Embedded techniques incorporate feature selection directly into the learning algorithm [72]. The feature selection process is an inherent part of the algorithm itself. During the training phase, the embedded algorithm dynamically assesses the relevance and importance of various features. It carefully examines the input data and identifies the most informative and discriminative features that affect the model's performance. Embedded techniques typically use feature importance or regularization methods to determine relevant elements. The algorithm defines the feature importance or coefficients during training, and it selects features based on their impact on the model's performance or through specific regularization techniques. These techniques don't only consider the relationships between individual input features and the output labels but perform local searches to identify features that enable improved discrimination. They employ independent criteria to determine the best subsets for a specific cardinality then they utilize a learning algorithm to choose the ultimate optimal subset from the optimal subsets obtained for different cardinalities.
Embedded methods share similar benefits to wrapper methods when it comes to how feature selection and classification interact. Additionally, they have a superior computational complexity due to incorporating feature selection directly into the training process of the classifier [73].

3.3.3 Wrapper techniques

Wrapper techniques involve assessing the relevance of features by utilizing a classifier and selecting only the most significant subset of features [74]. Feature selection consists of three key components: the search algorithm, the induction algorithm, and the evaluation metric. The initial feature space comprises N features, while the desired target feature space is a subset of the original features, encompassing k features selected from the N available options, where k ranges from 1 to N. Given that the number of possible feature subsets is equivalent to the power set of N, the search algorithm focuses on exploring the feature subset space to obtain the target feature subset. In the wrapper model, the search algorithm identifies feature subsets, which are then subjected to training and testing using a classifier designed for evaluating the performance of the selected feature subset. This classifier is known as the induction algorithm. During the evaluation stage, the classification results obtained from the wrapper model are compared with the correct data labels. Based on the prediction error, further search iterations are determined, or the search process is halted. This evaluation step is crucial in guiding the subsequent search actions or terminating the search process [68].

Table 3.2: Comparison of Feature Selection Methods

Method	Advantage	Disadvantage
Filter	• Computationally much more efficient than wrapper and embedded • Independent of the classifier • Easy and fast to implement • Better generalization ability than wrapper and embedded • Easily scale up to very high-dimensional data	• Usually evaluates features one by one and ignores correlation among features • Ignores interaction with the classifier
Wrapper	• Interacts with classifier • Usually evaluates features jointly and considers the dependency among them	• Computationally more expensive than filter and embedded • Requires building many models which is very time-consuming • More susceptible to overfitting • Low generalization ability • Classifier-dependent selection
Embedded	• Interacts with classifier • Evaluates features jointly and considers the dependency among them • Less susceptible to overfitting than wrapper techniques • Better computational complexity than wrapper techniques	• Classifier-dependent selection

3.4 Conclusion

In conclusion, the techniques of signal processing and feature selection are foundational to the success of our procedure. By refining the data through careful processing and selecting the most relevant features, we ensure that our model remains both accurate and efficient. These crucial steps not only optimize the analysis but also pave the way for more sophisticated methodologies in future chapters. Understanding and applying these processes will provide a solid foundation for achieving more advanced and reliable outcomes in the later stages of our work. In the next chapter, we will explore nature-inspired techniques and examine the final step of our procedure: classification.

Chapter 4

Nature-Inspired Optimization Algorithms and Machine Learning Classification

4.1 Introduction

In this chapter, we will delve into nature-inspired algorithms, which draw inspiration from natural processes and phenomena to solve complex computational problems. These algorithms have gained significant attention due to their ability to efficiently explore vast solution spaces and adapt to dynamic environments. Alongside these algorithms, we will also focus on the classification process using machine learning techniques. Classification plays a pivotal role in many applications, as it involves categorizing data into predefined classes based on learned patterns. By combining nature-inspired algorithms with machine learning classification, we aim to enhance the performance, robustness, and flexibility of our models, providing a powerful toolset for tackling real-world challenges.

4.2 Nature-inspired optimization algorithms

Nature-inspired optimization algorithms constitute a remarkable category of problem-solving techniques that derive their innovative power from emulating the fundamental principles and intricate behaviors exhibited within the natural world. These algorithms serve as invaluable tools in the pursuit of discovering optimal solutions to intricate and multifaceted challenges by mirroring the highly efficient and adaptive processes witnessed in the domains of biology, physics, and various other natural phenomena. By replicating the remarkable strategies that underpin the success of living organisms and the laws governing physical phenomena, these algorithms offer a unique avenue for tackling complex problems across diverse fields. With their widespread applicability and adaptability, nature-inspired optimization algorithms have become indispensable tools for addressing optimization challenges in an ever-expanding array of domains, exemplifying the incredible potential that can be harnessed from nature's profound wisdom []. Nature-inspired optimization algorithms constitute a remarkable category of problem-solving techniques that derive their innovative power from emulating the fundamental principles and intricate behaviors exhibited within the natural world. These algorithms serve as invaluable tools in the pursuit of discovering optimal solutions to intricate and multifaceted challenges by mirroring the highly efficient and adaptive processes witnessed in the domains of biology, physics, and various other natural phenomena. By replicating the remarkable strategies that underpin the success of living organisms and the laws governing physical phenomena, these algorithms offer a unique avenue for tackling complex problems across diverse fields. With their widespread applicability and adaptability, nature-inspired optimization algorithms have become indispensable tools for addressing optimization challenges in an ever-expanding array of domains, exemplifying the incredible potential that can be harnessed from nature's profound wisdom [75].

Many algorithms were developed based on this inspiration, notably the genetic algorithm (GA) introduced by Holland in 1975 [76], and swarm intelligence (SI) based algorithms. Indeed, a diverse array of SI-based techniques has surfaced over the past few decades. This assortment encompasses influential methods like ant colony optimization (ACO), a concept pioneered by Dorigo in 1992 [77], and particle swarm optimization (PSO) [78]. Additionally, other notable approaches include the bat algorithm (BA), innovatively proposed by Yang in 2010 [79], the firefly algorithm (FA) introduced by Yang in 2009 [80], and the cuckoo search (CS) algorithm detailed by Yang and Deb in 2009 [81].

These cutting-edge algorithms are inspired by the collaborative behaviors witnessed in the natural world, where the power of evolution and the synergy of swarming organisms have been harnessed for solving complex problems. Genetic algorithms, for instance, simulate the process of natural selection to evolve optimal solutions iteratively. Ant colony optimization draws inspiration from the foraging patterns of ants to find the shortest path in complex networks. Similarly, particle swarm optimization replicates the social interactions of birds and fish to optimize problem solutions. The bat algorithm, firefly algorithm, and cuckoo search, on the other hand, are further examples of nature-inspired algorithms that utilize various natural phenomena to address intricate challenges.

4.2.1 Ant colony optimization

In the early 1990s, Marco Dorigo and his team introduced the initial ACO algorithms [82], drawing inspiration from the collective behavior of ant colonies. Ants, as social insects, prioritize the survival of their colony over individual survival. The particular behavior that served as a model for ACO pertains to how ants efficiently discover the shortest routes between their nest and food sources. When foraging, ants initially explore their surroundings randomly while leaving chemical pheromone trails along their

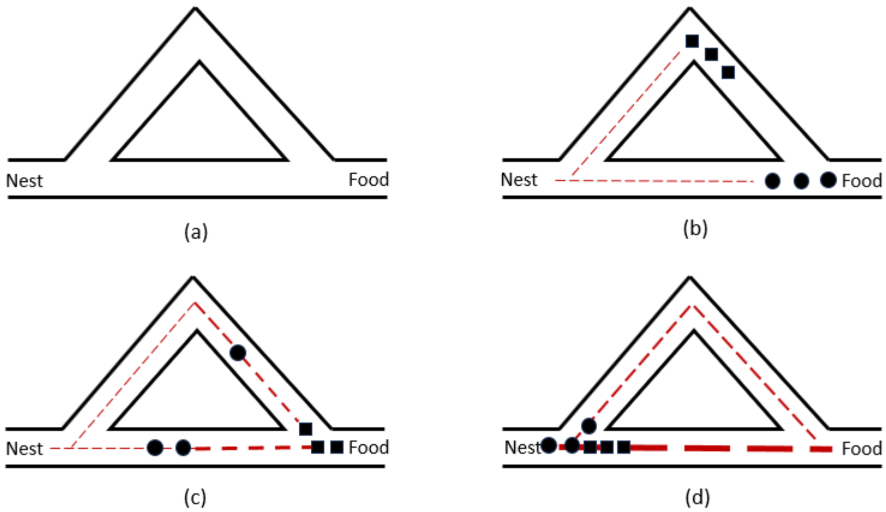

Figure 4.1: An experiment demonstrates how ant colonies find the shortest path. Two paths of different lengths connect the nest to the food source. In the visuals, pheromone trails are shown as dashed lines, with thickness indicating trail strength. a) There are two pathways linking the nest to the food source. b) When neither road has any pheromones, the chance of choosing either road is 50% for each. c) Ants, depicted as circles, that arrive first are more likely to opt for the shortest path. d) The shortest route will have a greater concentration of pheromones, and all ants will choose it.

paths. These pheromone markers are detectable by other ants, influencing their path choices, and favoring routes with higher pheromone concentrations. Upon finding a food source, an ant assesses both the quantity and quality of the food and transports some back to the nest. During this return journey, the amount of pheromone deposited on the ground may depend on the food's attributes. These pheromone trails effectively guide other ants to the food source. The indirect communication among ants through pheromone trails, a concept known as stigmergy, facilitates the discovery of the shortest paths between their nest and food sources, as depicted in an idealized setting in Figure 4.1 [reference 27]. On the other hand, feature selection involves the process of selecting a subset of relevant features (variables) from a larger set. In this context, we can draw an analogy between features and "paths" leading to a well-optimized model. Just as ants explore and exploit paths to efficiently locate food sources, feature selection algorithms explore and exploit various feature combinations to enhance the performance of a machine learning model. Feature importance measures, such as information gain, correlation, or importance scores generated by tree-based models, serve as the "pheromones" of the feature selection process. They provide guidance by assigning a significance level to each feature. Just as ants are drawn to trails marked with higher pheromone concentrations, feature selection algorithms favor features with greater importance scores, as they are deemed more likely to contribute to model success. While ant optimization algorithms continuously adjust pheromone levels and make probabilistic decisions to find optimal solutions, feature selection algorithms take on the role of iterative methods or heuristic search strategies. They systematically explore and evaluate different feature subsets, striking a balance between exploration and exploitation to maximize the utility of selected features. In both cases, the iterative nature of the process allows for continual refinement and adaptation, ultimately leading to the discovery of paths and feature subsets that offer the most efficient and successful solutions to their respective problems.

4.2.2 Grey wolf optimization algorithm

The Grey Wolf Optimization (GWO) is part of a class of algorithms known as meta-heuristic algorithms. It is a nature-inspired optimization algorithm that was introduced by Mirjalili et al in 2014 [83]. It is inspired by the social hierarchy and hunting behavior of grey wolves in the wild. The grey wolves are really good at hunting and are at the top of the food chain. They usually like living in groups, and these groups usually have about 5 to 12 wolves in them grouped into 4 classes as shown in figure 4.2. What's interesting is that they have a strict social order, like a leader, and the leaders are a male and a female called alphas. The alpha wolves make decisions about things like hunting, where to sleep, and when to wake up, and the rest of the pack follows their lead. Sometimes, though, they all work together to make decisions. When they gather, the whole group shows respect to the alphas by lowering their tails. The alphas are the only ones allowed to have babies in the group. Surprisingly, the alpha isn't always the strongest wolf, but they're the best at leading the pack. This shows that being organized and following the rules is more important than being the strongest.

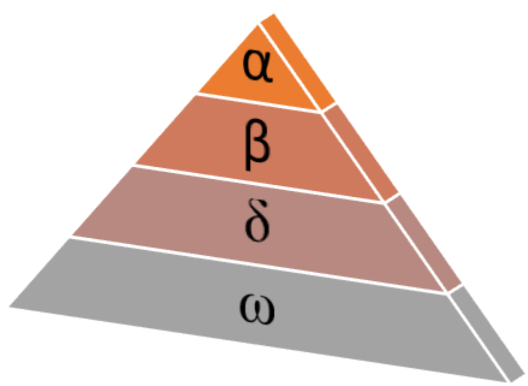

Figure 4.2: Hierarchy of grey wolf community

The second level in the group is called beta. Beta wolves help the alphas make decisions and do other things for the group. Beta wolves can be male or female, and they might become the alpha if one of the alpha wolves can't do the job anymore. Betas have to respect the alphas but also give orders to the lower-ranking wolves. They are like advisors and disciplinarians for the pack, making sure everyone follows the rules.

The lowest-ranked wolf is called omega. Omega wolves have to obey all the other higher-ranking wolves and are the last to eat. Even though they might not seem important, the group can have fights and problems if the omega is missing. This is because the omega helps release tension and frustration among the other wolves. This keeps the group together and organized. In some cases, omegas also take care of the young wolves in the pack.

If a wolf is not an alpha, beta, or omega, they are called subordinate or delta. Subordinate wolves have to listen to the alphas and betas but are in charge of the omegas. They have different roles in the group, like scouts, sentinels, elders, hunters, and caretakers. Scouts watch the territory's boundaries and alert the group to danger. Sentinels protect the pack. Elders are wise wolves who used to be alphas or betas. Hunters help find food for the pack. Caretakers look after sick or injured wolves in the group.

Apart from the social order, grey wolves have a fascinating way of hunting together. They follow a few steps: tracking, chasing, getting close to the prey, pursuing, surrounding and bothering the prey until it stops, and then attacking.

4.2.3 Squirrel search algorithm

The Squirrel Search Algorithm (SSA) stands out as a sophisticated optimization methodology drawing inspiration from the intricate workings of nature. This innovative approach intricately replicates the dynamic foraging behaviors observed in flying squirrels, particularly emphasizing their adept gliding locomotion. The art of gliding, known for its efficiency in enabling small mammals to traverse extensive distances, takes center stage as a pivotal element within the SSA [84]. At its core, SSA integrates fundamental concepts that mirror the resourceful strategies observed in squirrels. The algorithm harnesses the power of random exploration, local exploitation, and a food storage mechanism to refine and optimize solutions systematically. In this algorithmic framework, squirrels act as symbolic representations of candidate solutions, and their movements mimic the intricate exploration patterns within the search space [84].

The algorithm embarks on its optimization journey through a series of well-defined steps. It initiates the process by creating an initial population of squirrels, each representing a potential solution, with random generation adding an element of diversity to the candidate pool. As the algorithm progresses, these virtual squirrels navigate the search space, delicately balancing exploration to uncover new possibilities and exploitation to exploit promising areas [84].

Crucially, the fitness of each squirrel is meticulously assessed based on the predefined objective function, allowing the algorithm to discern the efficacy of each solution in the context of the optimization problem. The positions of the squirrels dynamically evolve as a result of the interplay between exploration and exploitation strategies, ensuring adaptability and responsiveness in the search for the optimal solution.

One distinctive feature of SOA lies in its incorporation of a memory mechanism, mirroring the food storage behavior observed in real-world squirrels. This mechanism enables the algorithm to retain and recall promising solutions, enhancing its ability to converge towards optimal outcomes over time.

In essence, the comprehensive set of steps within the SOA reflects the intricacies of nature-inspired foraging behaviors translated into a systematic and efficient optimization process. The algorithm not only encapsulates the essence of squirrel foraging but also leverages these behaviors to create a powerful optimization tool capable of addressing complex problems across diverse domains.

4.2.4 Grasshopper Optimization Algorithm

Grasshopper Optimization Algorithm (GOA) is a modern swarm intelligence algorithm that draws inspiration from grasshoppers' natural foraging and swarming behaviors [85]. Drawing on the intricate coordination and communication within a group of grasshoppers, the algorithm is structured around two fundamental concepts. Firstly, it emulates the swarming behavior of grasshoppers, where individuals collaboratively work together to discover optimal paths or solutions. Secondly, it incorporates a communication mechanism, allowing grasshoppers to exchange information about their positions and experiences, influencing the overall movement of the swarm. The algorithm unfolds through a series of well-defined steps. It commences with the initialization of a population of grasshoppers, randomly generated to represent potential solutions to the optimization problem at hand. The fitness of each grasshopper is then evaluated based on the objective function of the problem. Subsequently, grasshoppers dynamically move within the search space, adjusting their positions based on individual experiences and the information shared within the swarm. To prevent being confined to local optima, the algorithm strategically balances local exploration (exploitation of nearby solutions) and global exploration (exploration of distant solutions). Crucially, the update of positions is a dynamic process, influenced not only by individual experiences but also by the experiences of other grasshoppers within the swarm. This collaborative approach contributes to the algorithm's effectiveness in finding high-quality solutions. The communication mechanism is pivotal in fostering a collective and cooperative search strategy among grasshoppers. The GOA algorithm improves its chances of converging towards optimal solutions by sharing information about positions and the quality of solutions found.

4.2.5 Simulated Annealing

Simulated Annealing (SA) is a probabilistic approach employed to estimate the global optimum of a specified function, particularly suitable for expansive search spaces characterized by numerous local optima. It is a metaheuristic for global optimization and is inspired by the physical annealing process in materials science. The simulated annealing (SA) algorithm emulates the gradual heating of a material until it attains an annealing temperature, causing its molecular structure to weaken and become more amenable to alterations. Upon cooling, the molecular structure solidifies, becoming more resistant

to further changes [86]. The SA algorithm can be summarized in the following steps:
Initialize: Start with an initial solution $s = S_0$ and an initial temperature $t = t_0$.

Move: Perturb the current solution s through a defined move, such as displacement of a block to a new position, interchange of blocks, or orientation change for a block.

Calculate Score: Calculate the change in the objective function value (energy) between the current solution s and the perturbed solution s'.

Acceptance Probability: Calculate the probability of accepting the perturbed solution s' using the equation Prob(accepting uphill move) $\approx 1 - \exp\left(\frac{\Delta E}{kT}\right)$, where ΔE is the amount by which the objective function value is worsened (i.e., energy is increased), k is a constant related to temperature and energy, and T is a parameter analogous to temperature in an annealing system.

Temperature Reduction: At higher values of T, uphill moves are more likely to occur, while as T tends to zero, they become more unlikely. The temperature decreases according to an annealing schedule, making suboptimal moves less likely as the algorithm proceeds.

Acceptance: If the perturbed solution s' is better than the current solution s (i.e., energy is decreased), it is always accepted. If it is worse (i.e., energy is increased), it is accepted with a probability given by the acceptance probability equation.

Repeat: Repeat steps 2-6 until a stopping criterion is met, such as reaching a minimum temperature or a maximum number of iterations. Simulated Annealing (SA) proves advantageous in scenarios characterized by numerous local minima. In such cases, traditional gradient descent methods may become trapped in a local minimum, impeding their ability to reach the global minimum.

4.2.6 Firefly algorithm

The Firefly Algorithm, conceived by Xin-She Yang in 2008, draws inspiration from the flashing patterns exhibited by fireflies in the tropical summer sky. Functioning as a metaheuristic optimization algorithm, it has found applications in diverse domains such as digital image compression, image processing, feature selection, fault detection, antenna design, structural design, and scheduling [87].

The Firefly Algorithm can be summarized in the following steps [88]:

1. **Objective Function:** Define the objective function $f(x)$ for the optimization problem, where $x = (x_1, x_2, \ldots, x_d)$.

2. **Initial Population:** Generate an initial population of fireflies x_i $(i = 1, 2, \ldots, n)$.

3. **Light Intensity:** Formulate the light intensity I so that it is associated with $f(x)$ (for example, for maximization problems, $I \propto f(x)$ or simply $I = f(x)$).

4. **Absorption Coefficient:** Define the absorption coefficient n for all fireflies.

5. **Movement:** For each firefly i, compare its light intensity I_i with the light intensity I_j of other fireflies j within its communication range. If $I_j > I_i$, vary the attractiveness with distance r via $\exp(-\gamma r)$ and move firefly i towards j.

6. **Evaluation and Update:** Evaluate the new solutions and update the light intensity of the fireflies.

7. **Ranking and Update:** Rank the fireflies and find the current best solution.

8. **Repeat:** Repeat steps 5-7 until a stopping criterion is met, such as reaching a maximum number of iterations or a satisfactory solution.

4.2.7 Binary Differential Evolution

The Binary Differential Evolution (BDE) algorithm is an iterative optimization approach designed to enhance a candidate solution. This binary optimization technique integrates the principles of binary differential evolution (BDE) with a binary local search optimizer (BLSO).
The steps of the BDE algorithm can be outlined as follows [89]:

1. **Initialization:** Generate an initial population of candidate solutions (agents) x.

2. **Selection:** Select three agents x_1, x_2, and x_3 from the population.

3. **Crossover:** Generate a trial solution v by combining the selected agents using a crossover operation, such as binary crossover.

4. **Mutation:** Generate a mutant solution u by adding a scaled difference between two agents to the trial solution v.

5. **Acceptance:** Compare the trial solution v with the current agent x. If v is an improvement, update the agent x with v.

6. **Repeat:** Repeat steps 2-5 for all agents in the population.

7. **Termination:** Repeat steps 2-6 until a stopping criterion is met, such as reaching a maximum number of iterations or a satisfactory solution.

4.2.8 Clan-based Cultural Algorithm

Cultural Algorithms (CAs) represent a computational optimization approach inspired by the dynamics of human societies. At its core, a Cultural Algorithm revolves around the concept of individuals forming groups or clans within a broader community. These clans typically consist of closely related individuals who share common beliefs and values. They often engage in collective social activities, somewhat autonomously from other segments of society.

Within this societal framework, clans interact with each other, adhering to predefined rules and regulations and sometimes following scheduled events or rituals. These interactions can encompass a wide range of activities, such as trade, alliances, and collaborations, all aimed at improving the overall well-being of society.

The influence of each individual within a clan is subject to their adherence to clan-specific rules and norms. Additionally, there is a concept known as "clan influence," which represents the collective impact of all clan members. This influence extends beyond the clan's boundaries and can significantly impact the larger society. Clans with substantial influence often dominate the societal landscape, and their values and beliefs tend to become dominant.

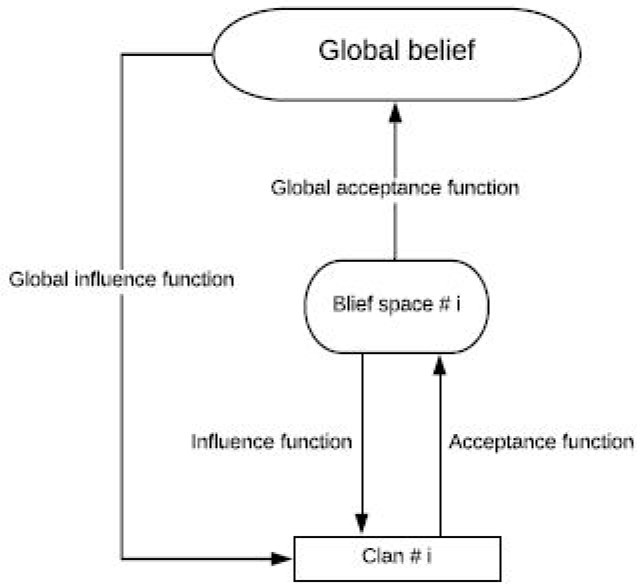

Figure 4.3: Clan-based Cultural Algorithm

The Clan-based Cultural Algorithm (CCA) is depicted in Figure 4.3. It commences by generating an initial population for each clan, denoted as P_n, where 'n' corresponds to the clan's unique identifier. Each clan is responsible for a distinct portion of the search space, allowing multiple clans to focus their search efforts on specific regions. During the initialization process, both the local belief spaces for each clan, represented as b_n, and the global belief space G are populated.

The population within each clan consists of individual solutions, denoted as $I(i, n)$. These solutions are created randomly and are presented as binary strings. The length of these binary strings aligns with the number of features in the dataset. For example, if the dataset comprises fifty features, each solution will comprise fifty binary digits. A binary '1' signifies the selection of a feature, while '0' signifies its exclusion. With a fixed population size of thirty individuals per clan, referred to as *popSize*, each clan houses thirty individual solutions, resulting in a total of 120 solutions encompassing all populations [90].

Each individual solution is subjected to evaluation using the k-nearest neighbor (kNN) classifier. This classifier determines the class label of data points based on their proximity to the 'k' nearest neighbors.

The belief space encompasses two distinct types of knowledge: situational and normative. Situational knowledge contains information about the best solution achieved within the population throughout generations. Normative knowledge defines the potential range of values that a solution can assume. Furthermore, attributes shared across all clans within the society are stored in the global belief space, denoted as G. The determination of G is facilitated through the utilization of Equation 4.1 [90].

$$G = \max(B_N) \qquad B_N = b_1, \ldots, b_n \tag{4.1}$$

The search process utilized by CCA follows an iterative approach, with each iteration considered as a "year." Each year leads to the generation of a new set of solutions, and this iterative process is depicted as a loop. In each generation, multiple clans evolve simultaneously within the societal context. At the onset of each year, the algorithm categorizes it into one of three types: migration year, global year, or normal year [90].

The algorithm checks whether the current year is a migration year. During migration years, there is an exchange of clan members within the society, fostering interactions among different clans. These inter-clan interactions serve to inject diversity into the population and expand the influence of each clan, thereby enhancing the overall search capabilities of the algorithm [9]. Although interactions can manifest in various ways, CCA models interactions as reproductive events between members of different clans. To enable reproduction, selected members from clan "n" migrate to randomly chosen destination clan "m" (Eq. 4.2), with the stipulation that the destination clan differs from the originating one [90].

$$d(I(i,n)) = p(P_m), \quad n = m; \quad n, m = 1, \ldots, c \tag{4.2}$$

The value of 'c' represents the overall count of clans within the society. The individuals chosen for migration are randomly selected from the population and are determined by equation 4.3.

$$mP_{op} = mRatio \cdot popSize \tag{4.3}$$

The migrating population, denoted as $mPop$, is determined by a migration ratio, $mRatio$, which is restricted to values below 0.6. Migrating individuals are merged with the original clan population, forming a unified mating pool. Offspring are generated through a combination of crossover and mutation operations. The crossover process employs a 3-point crossover technique at randomly selected locations, increasing the likelihood of producing offspring that differ from their parents, particularly when both parents are similar. The mutation rate is set at 10%. Subsequently, the combined population of parents and offspring undergoes evaluation, with weaker solutions being eliminated to ensure that the population retains the best solutions following the reproductive process.

At the global year, the global belief space influences all the clans. The global belief space, G, represents clan interactions at the topmost societal level. At this level, all decisions made affect all the clans. The global belief space contains the best solution across all clans (equation. 4.1) and represents the global acceptance function (Fig 4.3). This information is used to create new solutions across all clans. The global influence function creates new solutions similar to the best in the global belief space across all the clans, to direct the search process towards the global optimum. The global belief space influences the clans at every 10th generation to reduce the risk of homogeneity, and its influence affects only a percentage of the total population. This procedure ensures that the impact of the clan with the strongest influence will be propagated across all clans in the society, and thus the search will be directed towards the best solution in the search space [90].

Every member within a clan directly interacts with its dedicated local belief space. During each generation, all individual solutions are evaluated, and their fitness is calculated. Fitness is determined by the classification accuracy evaluated by the kNN classifier. Each solution's fitness is compared to the best solution stored in the local belief space's situational knowledge for the respective clan. The influence function identifies solutions with lower fitness and regenerates them within the bounds defined by the situational knowledge of the local belief space [90].

4.3 Machine learning and classification

4.3.1 Introduction to Machine Learning

Rather than encoding explicit principles into computers, machine learning (ML) attempts to acquire significant connections and patterns from instances autonomously and observations[91].
ML is a subfield of Artificial intelligence(AI). It is a revolutionary field that gathers computer science, statistics, and artificial intelligence. It has replaced the traditional way of problem-solving and decision-making by enabling computer systems to learn from data and make predictions or decisions without being explicitly programmed. Instead, ML uses algorithms and models that can improve their performance on a specific task through experience. This experience results after analyzing and learning patterns from large datasets. This ability to learn and adjust makes ML systems adaptable and valuable in various domains. However, the performance of these algorithms, in the first place, relies on the amount and the quality of data used for training, which led to the era of big data, where organizations collect and leverage vast datasets. ML algorithms ingest these data to gain insights, recognize patterns, and then use these patterns to make predictions or decisions for future data.

4.3.2 Types of Machine Learning

Machine learning algorithms are the spine of artificial intelligence. These algorithms are divided into four fundamental categories based on their learning approach and data nature, as shown in Figure 4.4.

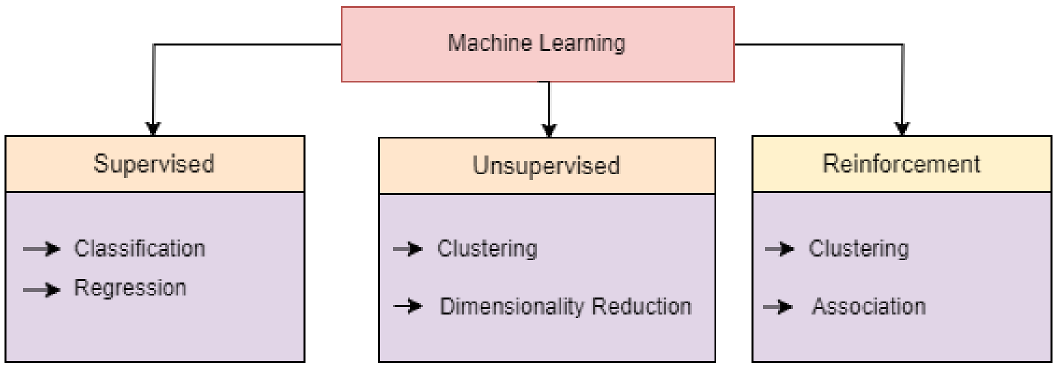

Figure 4.4: Machine learning categories.

4.3.2.1 Supervised Learning

Supervised learning is one of the fundamental categories of machine learning, where the model trains over labelled datasets. Within supervised learning, every data point in the training dataset belongs to a known and predefined target or class label. The core objective that drives supervised learning is to interpret and establish a robust mapping or correlation between the input features and their corresponding labels. Supervised learning aims to extract a mapping or relationship between the input features and their classes by processing the data and adjusting its internal parameters to minimize the difference between its predictions and the true labels, this mapping is captured by the algorithm in the form of a predictive model that allows the predictions or classifications of new data.
In essence, supervised learning forms the cornerstone of numerous real-world applications, playing a pivotal role in fields as diverse as mechanical diagnosis, healthcare diagnostics, natural language understanding, and recommendation systems.
Supervised learning can be further categorized into two main types:

4.3.2.1.1 Classification

Classification is a crucial technique in data mining and finds pervasive utility across diverse domains. In the classification procedure, data undergoes meticulous scrutiny, giving rise to definitive grouping rules, which, in turn, are harnessed to predict group associations for future data. This procedure is essential in revealing latent patterns and gaining profound insights from the expansive databases at our disposal. **Types of classification**
There are two distinct classification types [92] :

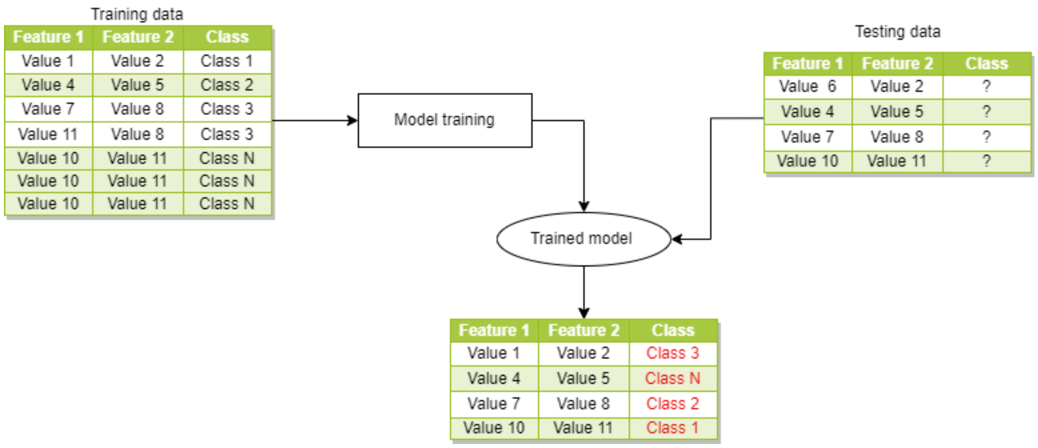

Figure 4.5: Classification procedure.

1. **Binary Classification**:
   Binary classification is supervised machine learning. It aims to categorize data into one of two possible classes or categories where the target variable can take on only two distinct values, often referred to as positive and negative.
   The primary objective of binary classification is to build a model that can learn to discriminate between the two classes based on input features. Various algorithms like logistic regression, support vector machines (SVM), decision trees, random forests, and many types of neural networks use binary classification. The performance of a binary classification model is evaluated using metrics such as accuracy, precision, recall, F1-score, and the area under the Receiver Operating Characteristic (ROC-AUC) curve.

2. **Multi-label Classification**:
   Multi-label classification belongs to supervised machine learning, where each data point can belong to one or more classes or categories. In contrast to binary classification, where each data point belongs to one of two exclusive classes, multi-label classification allows for more complex and flexible categorization.
   Multi-label classification is challenging because it requires handling dependencies between classes and addressing the potential overlap between labels. The choice of approach depends on the specific problem and dataset. Performance evaluation in multi-label classification often involves metrics like Hamming Loss, F1-score, and micro/macro-average precision and recall.

Fundamentals of Classification

Classification is a fundamental concept in machine learning as it serves as the foundation for a wide range of applications. It plays the role of a compass that guides us through unlabeled data, enabling us to categorize and uncover hidden insights from complex data. Before delving into classification, it is essential to understand its core principles. In this section, we will explain the fundamental aspects of classification.

- **Categories or Classes**: In classification, the set of predefined categories or classes you want to assign data points to. These classes represent the different possible outcomes or labels for data.

- **Features**: The set of features or attributes describes the data points. These features are the characteristics or properties of the data that the classification model uses for predictions. For example, in image classification, the features might be pixel values.

- **Training Data**: In supervised learning, data are used to build a classification model. This dataset consists of examples associated with known class labels. The model learns from this training data to make predictions. While in unsupervised learning, where we work with unlabeled data and data points lack known class labels. The model uses this training data to analyze and identify inherent patterns or groupings without the guidance of predefined categories.

- **Model building**: Machine learning algorithms, such as decision trees, support vector machines, and neural networks, are used to build classification models. These algorithms learn patterns in the training data and create a model that predicts the class of new and unseen data.

- **Prediction**: After training the classification model, it can be used to predict the class labels of new, unlabeled data points. The model analyses the features of the new data and assigns it to one of the predefined classes.

- **Evaluation**: The performance of a classification model is typically evaluated using various metrics such as accuracy, precision, recall, F1-score, and ROC-AUC, depending on the nature of the problem and the importance of false positives and false negatives. where:

  - **Accuracy**:
    Accuracy measures the proportion of correctly classified instances over the total instances in the testing dataset. It provides a general sense of the classification model's performance across all classes.

  - **Precision**:
    Precision measures the accuracy of positive predictions made by a model. It is crucial when false positives are costly.
    Formula:

$$\frac{TP}{TP + FP} \tag{4.4}$$

  - **Recall (Sensitivity or True Positive Rate)**: Recall measures the ability of a model to correctly identify all relevant instances (true positives). It is

essential when false negatives are costly.
Formula:

$$\frac{TP}{TP + FN} \tag{4.5}$$

– **F1-Score**:
The F1-Score is the harmonic mean of precision and recall. It provides a balanced measure of a model's performance, considering both false positives and false negatives. It is useful when there is an uneven class distribution or when you want to strike a balance between precision and recall.
Formula:

$$F1 - Score = \frac{2 \times (Precision \times Recall)}{Precision + Recall} \tag{4.6}$$

– **ROC-AUC (Receiver Operating Characteristic: Area Under the Curve)**:
It evaluates the performance of binary classification models. It measures the area under the Receiver Operating Characteristic curve, which plots the true positive rate (recall) versus the false positive rate at various thresholds.
A higher ROC-AUC score indicates a better ability of the model to distinguish between positive and negative instances.
Where:

  * True Positive (TP): The model accurately predicts a positive instance.
  * True Negative (TN): The model predicts a negative event with accuracy.
  * False Positive (FP): The model assigns a positive label to a data point when it is negative.
  * False Negative (FN): Occurs when the model predicts an instance to be negative when it is positive.
  * Hamming Loss: Hamming Loss is a metric used to evaluate the performance of multi-label classification models. It measures the fraction of the incorrectly predicted labels by the model for a given dataset. In essence, it quantifies the degree of disagreement between the true labels and the predicted labels across all data points. It is calculated as follows:
    1. For each data point, the true labels and the predicted labels are compared.
    2. For each label, if the true label and the predicted label match, there is no loss for that label. If they do not match, there is a loss for that label.
    3. The total number of label mismatches across all data points is divided by the total number of labels in the dataset to compute the Hamming Loss.

The mathematical equation of the hamming loss is as follows:

$$\text{Hamming Loss} = \frac{1}{NL}\sum_{i=1}^{N}\sum_{j=1}^{L}\delta(y_{ij} \neq \hat{y}_{ij}) \tag{4.7}$$

Exemples of supervised classification algorithm:
Among the supervised classification techniques, Decision Trees and Random Forest stand out as prominent examples, each with its unique approach and strengths in handling classification problems. In this section, we explore these two prominent supervised classification algorithms, which were used in our study for a better understanding.

1. **Decision Tree**
   Decision trees are foundational components of machine learning, representing versatile tools for prediction and classification tasks, they involve a series of straightforward tests as shown in figure 4.6, where each test compares either a numeric attribute to a specific threshold value or a nominal attribute to a set of potential values [93]. They hold historical significance as one of the earliest statistical algorithms implemented in the later decades of the 20th century [94].

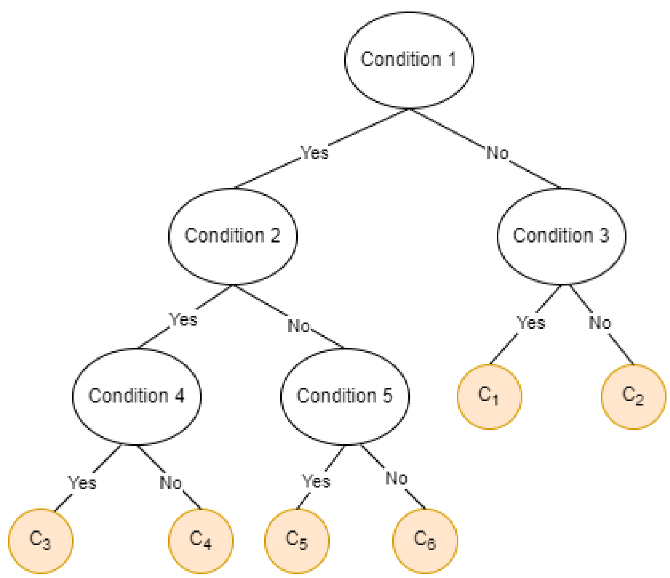

Figure 4.6: A decision tree

Condition 1	Condition 2	Condition 3	Condition 4	Condition 5	Label
No	-	Yes	-	-	C_1
No	-	No	-	-	C_2
Yes	Yes	-	Yes	-	C_3
Yes	Yes	-	No	-	C_4
Yes	No	-	-	Yes	C_5
Yes	No	-	-	No	C_6

Table 4.1: Training dataset

At their core, decision trees exhibit a distinctive characteristic: a recursive subdivision of a target dataset based on the values of relevant input features or predictors. This recursive process leads to the formation of nodes or leaves, each representing a distinct partition and associated descendant data subsets. These partitions, known as leaves or nodes, exhibit a remarkable property: they contain instances with increasingly similar target values as one delves deeper into the tree's hierarchy. Conversely, the differences in target values between different nodes or leaves become more pronounced at any given level of the tree.

Decision trees offer a unique advantage in interpretability, making them accessible and comprehensible to human users. The resulting tree structure provides transparent insights into the decision-making process, enabling users to grasp the rationale behind model predictions.

Moreover, decision trees find application across a broad spectrum of domains, encompassing classification, regression, and feature selection tasks. Beyond their standalone utility, decision trees serve as the foundational building blocks for advanced ensemble methods, including Random Forests which leverage multiple decision trees to enhance predictive accuracy and robustness.
In the process of constructing the tree and deciding which attributes to use for node splitting, three key parameters are considered: Information Gain (IG), Gain Ratio, and Gini Value.

(a) **Information Gain (IG):** Information Gain measures the reduction in uncertainty or entropy achieved by partitioning a dataset based on a specific attribute. It quantifies how much information is gained about the target variable (class labels) when the dataset is split using a particular attribute. The formula for Information Gain is typically defined as [93]:

$$IG(T, A) = H(T) - H(T|A) \tag{4.8}$$

Where:

- $IG(T, A)$: Information Gain for attribute A in tree T.

- $H(T)$: Entropy of the original dataset T.
- $H(T|A)$: Conditional entropy of T given attribute A.

A higher Information Gain indicates that the attribute is more informative for splitting the data.

(b) **Gain Ratio:**

Gain Ratio is an extension of Information Gain that takes into account the number of categories or values an attribute can take. It aims to reduce the bias towards attributes with a large number of values. The formula for Gain Ratio is [93]:

$$GainRatio(T, A) = \frac{IG(T, A)}{SplitInfo(T, A)} \tag{4.9}$$

Where:

- $GainRatio(T, A)$: Gain Ratio for attribute A in tree T.
- $IG(T, A)$: Information Gain for attribute A.
- $SplitInfo(T, A)$: Split Information, which measures the potential information generated by splitting attribute A in tree T. It is calculated as:

$$SplitInfo(T, A) = -\sum_{j=1}^{k} \frac{|T_j|}{|T|} \cdot \log_2\left(\frac{|T_j|}{|T|}\right) \tag{4.10}$$

  Where:
  - k is the number of partitions created by attribute A.
  - $|T_j|$ is the number of instances in partition T_j.
  - $|T|$ is the total number of instances in dataset T.

The Gain Ratio normalizes the Information Gain by the intrinsic information of the attribute, helping prevent overfitting when dealing with attributes that have many values.

(c) **Gini Value (Gini Index or Gini Impurity):** Gini Value measures the impurity of a dataset, indicating how often a randomly chosen element from the set would be incorrectly classified. In the context of decision trees, Gini Value is used to evaluate how well a particular attribute separates the data into classes. The formula for Gini Value is [93]:

$$Gini(T) = 1 - \sum_{i=1}^{c}(p_i)^2 \tag{4.11}$$

Where:

- $Gini(T)$: Gini Value for dataset T.
- c: The number of classes.
- p_i: The probability of an element in T belonging to class i.

A lower Gini Value implies that the attribute is better at separating the data into pure classes, meaning the data is more homogeneous when split by that attribute.

These metrics play a crucial role in the attribute selection process of decision tree algorithms (e.g., CART or C4.5) to determine the best attribute for splitting a node and building an effective decision tree for classification tasks.

2. **Random Forest**
Random Forest (RF) is a powerful ensemble classifier renowned for its robustness and effectiveness in various machine-learning tasks. This classifier is constructed from a multitude of decision trees, each generated using random vectors independently sampled from the input vector [95]. One of RF's key innovations lies in the controlled randomness it introduces during the tree-building process. Specifically, it limits the number of parameters used to determine the optimum split at each node to a carefully selected subset of the total parameters, chosen at random [96].

RF leverages Breiman's Classification and Regression Tree (CART) method as its foundation for determining splits in the training data and constructing individual trees. This process begins by producing a bootstrap sample from the training set, ensuring that each tree's training dataset is a distinct subset of the original data [97].

In a classifier tree, each node represents a specific condition or feature, and it makes a binary decision, taking either one path or another, thus giving rise to two subnodes. The critical objective at each node is to maximize the homogeneity between the two resulting subnodes. Various measures can be employed to quantify this homogeneity, but one of the most popular methods for categorical classification is the Gini Index, which serves as a measure of impurity. The Gini Index for a node p is defined by Equation 4.12 [97].

$$I_P = n_P . \sum_{c=1}^{C} \Pi_p(c)[1 - \Pi_p(c)] \tag{4.12}$$

Here, n_P represents the total number of training instances, Π_P denotes the proportion of occurrences of class c in node p, and c signifies the class. The decrease in impurity, or equivalently, the increase in homogeneity resulting from the split of nodes R and L, can be quantified using Equation 4.13.

$$\Delta I(P, L, R) = I_P - (I_L + I_R) \tag{4.13}$$

In the tree-building process, the split that maximizes $\Delta I(P, L, R)$ is selected after evaluating all alternative splits, and this process continues until it reaches the predefined maximum depth.

In the context of classification, each tree in the Random Forest provides a unit vote for the most popular class. The true power of RF emerges when these individual tree predictions are aggregated. RF makes its final prediction for each sample by taking a majority vote from all predictor trees. This ensemble approach helps RF achieve robustness and high predictive accuracy, making it a versatile choice for a wide range of machine-learning tasks.

RF offers several advantages, including its ability to handle high-dimensional data, its resistance to overfitting, and its capability to provide feature importance rankings. Additionally, RF is capable of handling missing data, which is a common challenge in real-world datasets. This makes it a valuable tool for both classification and regression.

4.3.2.1.2 Regression

Regression is a supervised machine-learning technique for modeling and analyzing the relationships between a dependent variable (target) and one or more independent variables (predictors or features). The primary goal of regression analysis is to understand how changes in the independent variables relate to changes in the dependent variable. In essence, it helps us to predict or estimate a continuous numerical outcome based on input data, while classification involves predicting discrete labels for the input. Regression is categorized into two primary types: Simple Linear and Multiple, as shown in Figure 4.7.

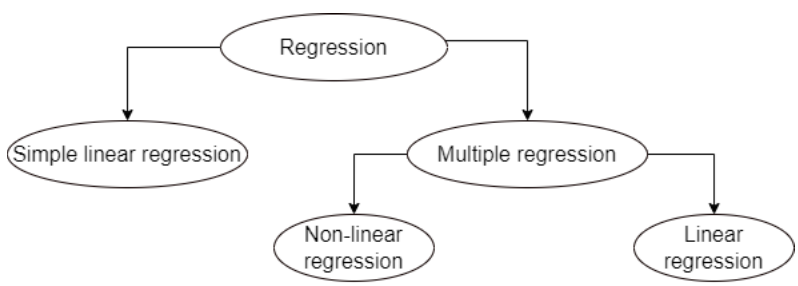

Figure 4.7: Regression types.

Simple Linear Regression

Simple Linear Regression is a fundamental concept in regression analysis used to model the linear relationship between a dependent variable (target) and a single independent variable (predictor) by finding the best-fit line (a straight line) that minimizes the difference between the predicted values (Y) and the actual data points as shown in Figure 4.8. The simple linear regression model is represented by the equation:

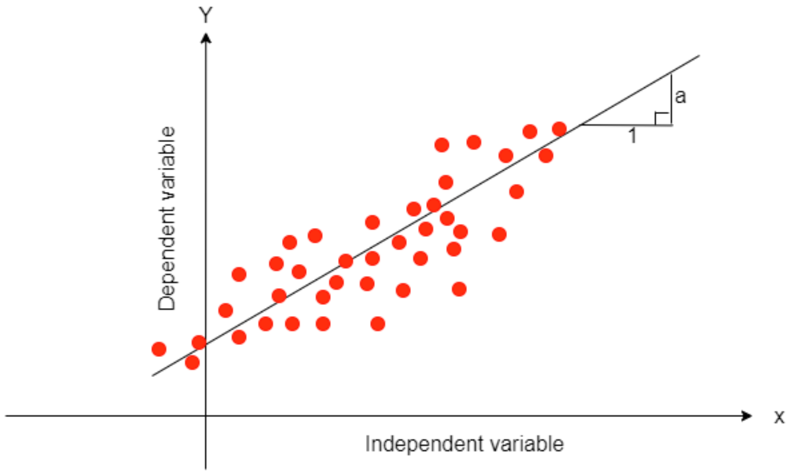

Figure 4.8: Linear regression.

$$y = ax + b \tag{4.14}$$

where:

- x: is the independent variable (the variable used for prediction)
- y: is the dependent variable (the variable we want to predict)
- a: is the slope of the line
- b: is the y-intercept, indicating the value of Y when X is zero.

Multiple regression

Multiple regression is a statistical technique used in data analysis to investigate the relationship between a dependent variable and two or more independent variables. It extends the simple linear regression concept, which examines the relationship between a dependent variable and a single independent variable, to a situation where multiple independent variables may influence the dependent variable simultaneously. In multiple regression, the goal is to create a linear equation or model that predicts the dependent variable value based on the independent variables. The model is represented by equation 4.15:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + ... + \beta_k x_k + \epsilon \tag{4.15}$$

Where:

- Y represents the dependent variable.
- X_1, X_2, ..., X_k: represent the independent variables.

- β_0: represents the intercept (the value of Y when all independent variables are zero).
- β_1, β_2, ..., β_k: coefficients represent the effect of each independent variable (X_1, X_2, ..., X_k) on the dependent variable (Y).
- ϵ: represents the error.

The coefficients β_1, β_2, ..., β_k are estimated using statistical methods that minimize the sum of squared differences between the predicted values and the actual values of the dependent variable. Once the coefficients are determined, the multiple regression model can be used to make predictions about the dependent variable based on specific values of the independent variables.

4.3.2.2 Unsupervised Learning

Unsupervised learning is a category of machine learning. It investigates how systems can learn to define specific input patterns in a way that represents the statistical structure of the entire collection of input patterns [98]. Unsupervised learning uses unlabeled data where the algorithm is not provided with a target variable to predict or classify. Instead, it tries to find patterns or structures within the data.
The goal of unsupervised learning is to reveal the underlying data structure or reduce the dimensionality of the data. There are two common types of unsupervised learning techniques:

4.3.2.2.1 Clustering

Clustering, within the domain of unsupervised learning, consists of object grouping based on shared inherent similarities. It entails organizing data patterns into subsets, where each shares some resemblances. This process results in a structured arrangement that characterizes the sampled population. In a formal and traditional context, the clustering structure is represented as a collection denoted as S, consisting of subsets labelled as S_1, S_2... S_K, such that:

$$S_1 \cap S_2 \cap ...S_K = \emptyset \tag{4.16}$$

This implies that each element within the set S (comprising S1 through Sk) is exclusively assigned to a single subset. The clustering process also finds relevance in characterizing the distinctive attributes of individuals, facilitating their recognition based on shared similarities. In a broader context, we can categorize individuals into distinct clusters based on many factors like gender, height, weight, color, voice, and other physical characteristics. Therefore, clustering encompasses many interdisciplinary fields, from mathematics and statistics to biology and genetics. Across these domains, various terminologies are employed to elucidate the topological structures derived from the application of this clustering analysis technique.
Clustering is generally regarded as a more challenging task compared to supervised

classification due to the absence of predefined labels associated with data patterns. In supervised classification, these labels serve as valuable cues for grouping data objects comprehensively. However, in clustering, the absence of such labels poses a significant challenge in determining which group a pattern should be assigned to. Additionally, the complexity is amplified by the presence of multiple parameters or features that could potentially be used for clustering. The curse of dimensionality exacerbates this situation, as high-dimensional data further compounds the difficulty [99].

Types of clustering

Various clustering methods have been suggested, each employing distinct inclusion principles. Fraley and Raftery in [100] recommended categorizing these clustering methods into two main groups: hierarchical methods and partitioning techniques as shown in Figure 4.9.

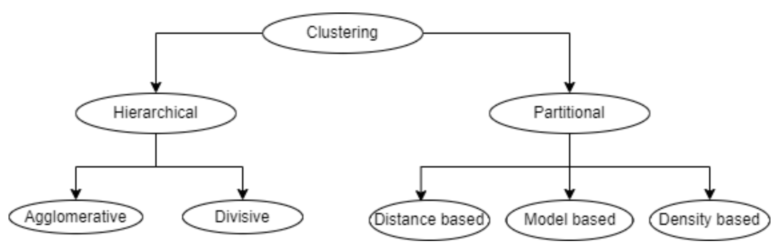

Figure 4.9: Clustering types.

Hierarchical clustering

In hierarchical clustering techniques, clusters are created by progressively dividing the data patterns using either a top-down or a bottom-up approach as illustrated in Fig.4.10. There are two variations of hierarchical methods, known as agglomerative clustering and divisive clustering [101].

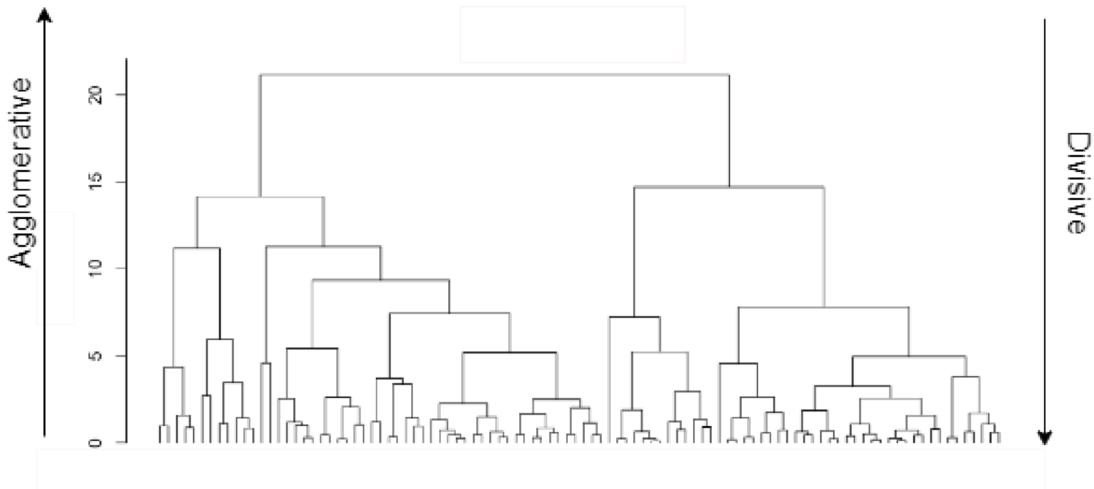

Figure 4.10: Cluster Dendrogram.

Agglomerative clustering, adopting a bottom-up strategy, begins by forming clusters with individual objects and subsequently combines these smaller clusters into progressively larger ones. This process continues until all objects form a single set or specific termination criteria are satisfied. On the other hand, divisive hierarchical clustering employs a top-down approach, starting with a single cluster encompassing all objects and then progressively subdividing it into smaller clusters. This process continues until each object forms its own cluster or termination conditions are met.

Partitional clustering

Partitional clustering, in contrast to hierarchical clustering, involves assigning data into K clusters without forming any hierarchical structure. This process is carried out by optimizing a specific criterion function[99]. This category can be divided into three main classes: distance-based, model-based and density-based.

- **Distance based**
  Distance-based clustering algorithms rely on the notion of distance between data points to create clusters. These algorithms work by defining a distance metric, the most commonly used criterion is the Euclidean distance, which identifies the minimum distance between data points and each available cluster, subsequently assigning the data point to the most suitable cluster after measuring the similarity or dissimilarity between data points. Common examples of distance-based clustering algorithms include K-Means and K-Medoids. K-Means, for instance, partitions data points into K clusters by minimizing the sum of squared distances between data points and the centroid of their assigned cluster.

- **Model-based**
  Model-based clustering algorithms assume that the data is generated from a probabilistic model. They attempt to fit statistical models to the data and use these

models to identify clusters. Expectation-Maximization (EM) is a popular model-based clustering algorithm. It iteratively tunes the model parameters until it converges to a solution that explains the observed data well.

- **Density-based**
  Density-based clustering aims to identify clusters of various shapes within spatial databases, even in the presence of noise. It forms groups based on the largest group of densely connected data points. At the heart of density-based clustering are two crucial concepts: density-reachability and density-connectivity. This approach relies on two input parameters. "ϵ" represents the radius, and "MinPts" denotes the minimum number of points necessary to constitute a cluster. The process initiates with an arbitrary, unvisited starting point. Subsequently, it retrieves the "ϵ"-neighborhood, and if this neighborhood contains a sufficient number of points, a cluster is initiated [102].

Exemple of supervised classification algorithm:
One powerful approach within the realm of unsupervised learning is the Expectation-Maximization (EM) algorithm applied to Gaussian Mixture Models (GMM) which was used for feature selection in our study.

Expectation-Maximization Gaussian Mixture Model (EM-GMM)

Model-based clustering methods offer the advantage of creating flexible, soft partitions within datasets. These methods employ mixture distributions to model the data, allowing for the assignment of probabilistic labels based on conditional probabilities. The Gaussian Mixture Model (GMM) holds a prominent position among the frequently used mixture models for clustering. In a GMM, each Gaussian density is referred to as a component of the mixture and possesses its unique mean and covariance. In various applications, these component parameters are estimated through maximum likelihood, often employing the Expectation-Maximization algorithm (EM) [103].

Gaussian Mixture Model (GMM)

The Gaussian Mixture Model (GMM) is a versatile probabilistic model used for clustering and modeling data that is assumed to be generated by a mixture of multiple Gaussian distributions. Each Gaussian density in a GMM represents a fundamental function or a "latent" unit. This modeling approach aims to provide a more comprehensive representation of data compared to a single Gaussian distribution [104].

In mathematical terms, a GMM can be defined as a linear combination of M component Gaussian densities:

$$p(x|\lambda) = \sum_{i=1}^{M} w_i g(x|\mu_i, \Sigma_i) \tag{4.17}$$

Where:

- $p(x|\lambda)$ is the probability density function (PDF) of the data vector x given the GMM parameters λ.
- M is the number of mixture components in the GMM.
- w_i represents the weight assigned to each mixture component i, with i ranging from 1 to M.
- $g(x|\mu_i, \Sigma_i)$ is the Gaussian density function for each component i, defined as:

$$g(x|\mu_i, \Sigma_i) = \frac{1}{(2\pi)^{D/2}|\Sigma_i|^{1/2}} \exp\left(-\frac{1}{2}(x-\mu_i)^\top \Sigma_i^{-1}(x-\mu_i)\right) \tag{4.18}$$

Where:

- μ_i represents the mean vector for component i.
- Σ_i is the covariance matrix for component i.
- D is the number of dimensions in the data vector x.

GMMs are widely applied in various machine learning and statistical applications, including clustering, density estimation, and classification.

Parameter Estimation with the EM Algorithm

The estimation of the characterizing parameters, including means, covariances, and mixture weights, is a crucial step in implementing GMMs. The Expectation-Maximization (EM) algorithm is an efficient tool commonly used for this purpose [105] [106].

In the context of GMMs, maximum likelihood estimation aims to find the parameter values that maximize the likelihood of the observed data. In the case of continuous data distributions, the likelihood is defined as the joint probability of the data. Assuming that each data point is independent, the likelihood of all data points is expressed as the product of the likelihood of each data point:

$$L(\Theta) = \prod_{i=1}^{n} P(x_i|\Theta) \tag{4.19}$$

Here, Θ represents the set of parameters to be estimated, including means and covariances. To find the maximum likelihood estimates, we need to maximize this likelihood function. However, directly maximizing the product of probabilities is a complex task.

As a solution, the EM algorithm simplifies this optimization by introducing the log-likelihood function. Taking the logarithm of the likelihood transforms the product of probabilities into a sum:

$$\log \prod_{i=1}^{n} P(x_i|\Theta) = \sum_{i=1}^{n} \log P(x_i|\Theta) \tag{4.20}$$

We denote the parameter values that maximize this log-likelihood as $\hat{\Theta}$:

$$\hat{\Theta} = \arg\max \sum_{i=1}^{n} \log P(x_i|\Theta) \tag{4.21}$$

Where $\arg\max$ represents the arguments of maxima, and it refers to the values of Θ that maximize the log-likelihood function.

The EM algorithm iteratively refines the parameter estimates, alternating between two steps: the Expectation (E) step, which computes the expected values of the latent variables, and the Maximization (M) step, which maximizes the expected log-likelihood for the model parameters.

In summary, Gaussian Mixture Models and the Expectation-Maximization algorithm provide a robust framework for modeling complex data distributions and estimating their underlying parameters, making them valuable tools in the fields of machine learning and statistics.

4.3.2.2.2 Dimensionality Reduction

The Dimensionality Reduction (DR) objective is to minimize the distance between distributions of various datasets within a latent space, facilitating efficient transfer learning. The outcomes reveal that when considering each device separately, the results obtained through Dimensionality Reduction (DR) are significantly more favorable. Transforming the data into a lower-dimensional representation effectively addresses the challenge posed by high dimensionality, making data analysis, processing, and visualization more accessible. This underscores the advantages of employing dimensionality reduction techniques on a dataset [107].

Typically, Dimensionality Reduction (DR) techniques fall into two fundamental categories: Feature Selection (FS) and Feature Extraction (FE). Feature Selection (FS) holds paramount importance in the context of increasing data generation rates. This methodology is crucial in mitigating substantial challenges associated with dimensionality, including effective redundancy reduction, removing extraneous data, and enhancing the result interpretability. In contrast, Feature Extraction (FE) addresses the intricate task of identifying the most distinctive and informative feature set, thereby augmenting the efficiency of both data processing and storage systems [108].

4.3.2.3 Reinforcement learning

Reinforcement learning (RL) offers qualitative and quantitative frameworks that facilitate the comprehension and modeling of adaptive decision-making processes in response to rewards and penalties [109], as shown in Figure 4.11. It is the problem faced by an agent that must learn behavior through trial-and-error interactions with a dynamic environment.

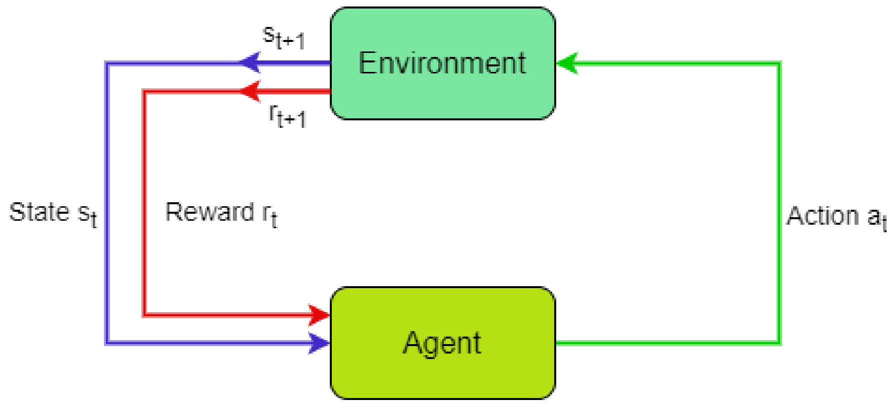

Figure 4.11: The standard reinforcement-learning model.

RL algorithms can be broadly classified into two main categories: model-based and model-free approaches, each characterized by unique optimization techniques[110].

- **Model-based RL**: In model-based reinforcement learning, experience is employed to build an internal representation of how the environment transitions and yields immediate results. Decisions regarding suitable actions are subsequently made by exploring or strategizing within this internal model of the world. This approach is statistically efficient, as it enables the storage of environmental information in a statistically accurate and computationally manageable manner. As long as continuous replanning remains feasible, this approach facilitates adaptable action selection in response to alterations in transition patterns and outcome values [110].

- **Model-free RL** Model-free RL refers to a category of reinforcement learning methods where an agent learns how to make decisions (select actions) directly from its interactions with the environment, without building an explicit model of how the environment works [110].
  Model-free approaches are statistically less effective when compared to model-based methods. This is due to the fact that model-free methods combine current information from the environment with past, and potentially incorrect, estimations or assumptions about state values, instead of utilizing information directly. Additionally, these methods store information in singular numerical values, making it difficult to later separate specific details about rewards or state transitions. Consequently, model-free methods are less capable of rapidly adjusting to alterations in environmental conditions and outcome values.[110]

4.4 Conclusion

This chapter has provided an overview of the nature inspired optimization algorithms and the machine learning classification. By outlining the step-by-step process and highlighting the significance of each stage, we have offered transparency into the methodologies employed. This chapter serves to provide clarity on the methodologies driving our study and lays the foundation for the subsequent discussion and analysis presented in the following chapter.

Chapter 5

Results and discussions

5.1 Introduction

Our primary objective is to address real-world problems and provide practical solutions in the field of diagnosis. One of the most critical challenges is to detect anomalies and pinpoint their location accurately under unstable conditions, especially in the presence of irrelevant parameters within the training model that mislead the diagnosis process and result in inaccurate results, ultimately leading to decreased efficiency and increased maintenance costs. All results obtained in subsequent work were rigorously assessed using **MATLAB 2018a** on a **HP ProBook 450 with an Intel i3 processor**, ensuring the reliability and consistency of the analysis and results.

In this thesis, our primary focus was improving the quality of the model attributes used in the diagnosis procedure. We shine a light on the feature selection step to identify the most relevant parameters that better describe the state of the diagnosed bearings. The precision of the diagnosis depends mainly on the quality and representativeness of the selected features. To address this critical issue, we proposed three novel procedures to improve the diagnosis process under time-varying conditions. Our contributions provide innovative approaches to tackle the challenges of diagnosis in real-world cases and increase the reliability and efficiency of the diagnosis process.

5.1.1 Bearing fault detection under time-varying speed based on empirical wavelet transform, cultural clan-based optimization algorithm, and random forest

Motivation

Anomaly detection stands as a fundamental imperative in a range of academic and practical domains. It assumes heightened significance due to its formidable nature, particularly when confronted with datasets sourced from varying conditions.

Our first contribution is the proposition of a novel procedural framework. This framework is strategically designed to address the intricate challenge of effective anomaly monitoring and detection within the specialized context of bearing systems.

Our approach leverages advanced techniques encompassing machine learning and sophisticated statistical modeling. This methodological amalgamation empowers our system to proficiently discern and isolate anomalies, even amidst the relentless dynamism and volatility that characterize the underlying operating conditions.

The database used in this study contains data acquired under conditions characterized by significant temporal variations. This database serves as the foundational bedrock upon which we construct a robust and adaptive methodology. It is specifically designed to navigate the inherent complexities of authentic real-world scenarios.

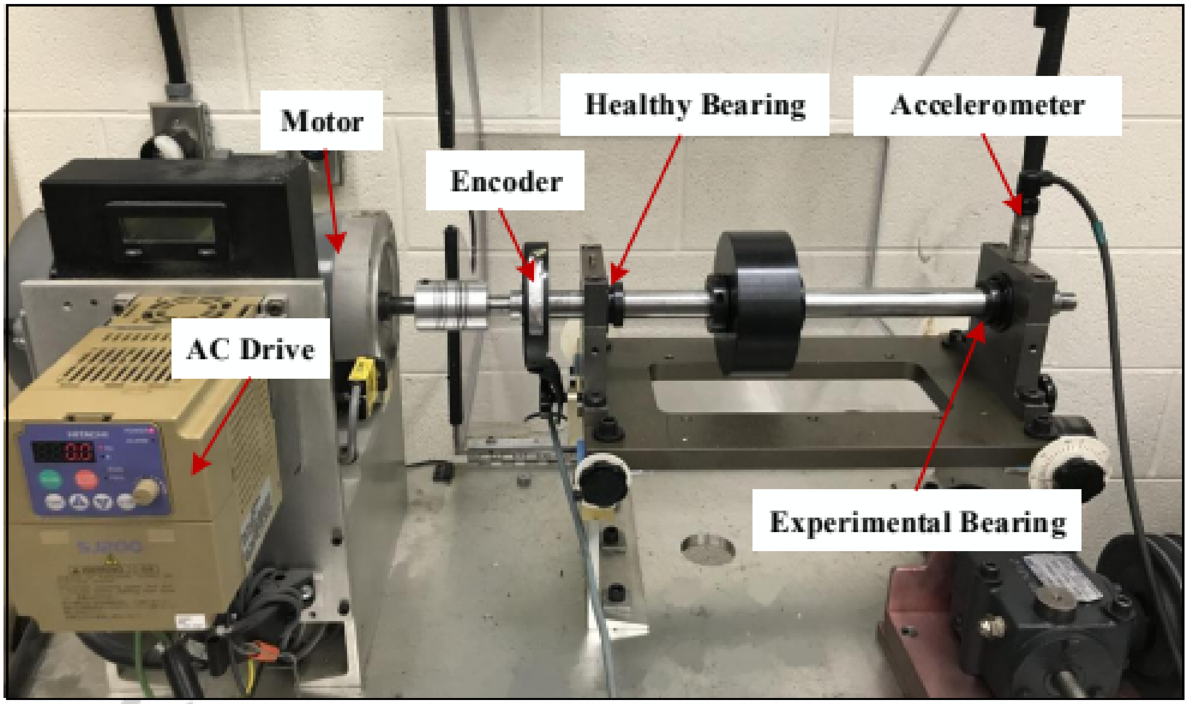

Figure 5.1: SpectraQuest machinery fault simulator set-up [4]

5.1.1.1 Database description

Vibration signals are of paramount importance in the field of condition monitoring and fault diagnosis for mechanical systems. They provide critical insights into the health and performance of various components, with bearings being a prime area of focus due to their ubiquitous presence in machinery. In this article, we specifically discuss bearing faults, a topic of great relevance for ensuring the reliability and safety of industrial equipment.

One of the key challenges in bearing failure diagnosis is addressing inherent variability in operating conditions. Bearings often operate in dynamic environments with fluctuating speeds, such as those encountered in real-world industrial applications. To mirror these complex and dynamic scenarios, we turn our attention to the dataset known as ”Bearing vibration data collected under time-varying rotational speed” a resource curated by Huang and Baddour in 2018 [4].

This dataset stands out for its meticulous curation and its ability to capture the multifaceted nature of bearing operations, the data are obtained using the set-up shown in figure 5.1. The data set encompasses a range of bearing health conditions, including the pristine state of healthy bearings and those afflicted with specific types of defects, namely inner race and outer race defects. The inclusion of various health states enables us to examine the distinctive vibration patterns associated with these conditions, which is vital for accurate fault detection and diagnosis. One of the standout features of this dataset is its incorporation of variations in operational speed. This is a critical consideration, as bearings in real-world settings frequently operate under conditions of changing rotational speed. To encapsulate the intricate dynamics of speed fluctuations, the dataset presents four specific scenarios:

1. Increasing Speed: In this scenario, the rotational speed of the bearing progressively rises. This mirrors situations where equipment accelerates during its operation.

2. Decreasing Speed: Conversely, the dataset models the case where the rotational speed steadily decreases. This scenario is representative of equipment deceleration or gradual shutdown.

3. Increasing then Decreasing Speed: Here, the rotational speed undergoes a cycle of ascent followed by descent. Such variations are common in equipment subjected to cyclic operational patterns.

4. Decreasing then Increasing Speed: This scenario mirrors situations where rotational speed initially decreases and subsequently increases, which is a pattern often seen in equipment startup and shutdown sequences.

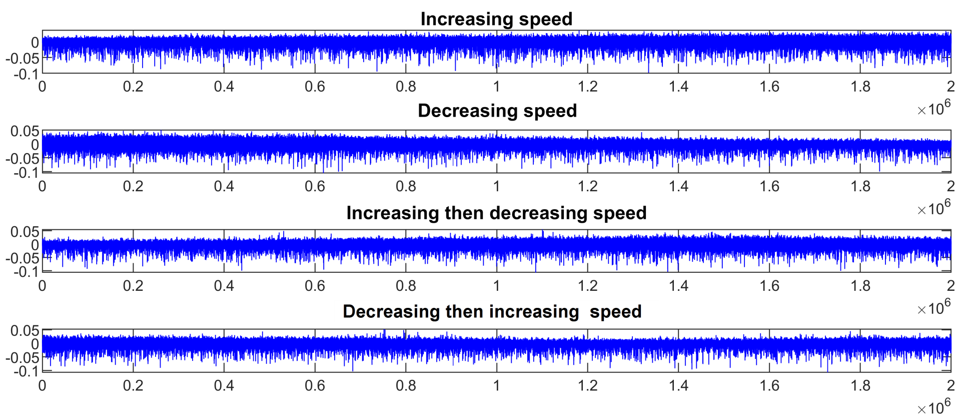

Figure 5.2: Healthy bearing vibration signals

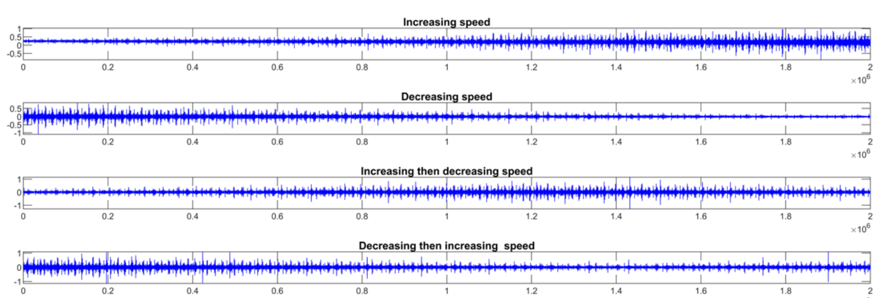

Figure 5.3: Vibration signals of bearings with inner race defects

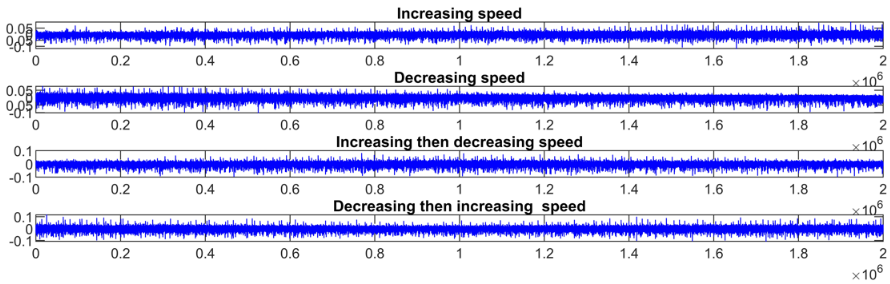

Figure 5.4: Vibration signals of bearings with outer race defects

To ensure the robustness and reliability of the dataset, each of these speed scenarios was subjected to three separate trials. This meticulous approach results in the creation of 36 subdatasets, allowing for comprehensive exploration and analysis of the various operating conditions.

The data collection process itself is worth noting. The dataset was meticulously assembled using a SpectraQuest machinery fault simulator (MFS-PK5M), a sophisticated setup that offers precise control over rotational speed through a DC drive. An accelerometer, a critical instrument in capturing vibration data, was employed to monitor the mechanical vibrations associated with bearing operations. Furthermore, an encoder was integrated into the setup to accurately measure the rotational speed of the shaft. This multi-sensor approach ensures that the dataset is well-rounded and captures essential parameters for comprehensive analysis.

Each data sample within the dataset is structured to contain two distinct channels. The first channel records the vibration data, offering valuable insights into the mechanical condition of the bearing. The second channel is dedicated to capturing the rotational speed, providing a critical parameter for assessing the dynamics of the system. Both of these channels were sampled at a high rate of 200 KHz, ensuring that detailed and time-sensitive information is available for analysis. The duration of each data point was set at 10 seconds, a decision aligned with the methodology established by Huang and Baddour in their 2018 study.

5.1.1.2 Experimental procedure

In Figure 5.20, we present the schematic representation of our novel methodology designed for bearing faults classification. This methodology is structured into three fundamental phases: signal processing, feature extraction and selection, and classification.

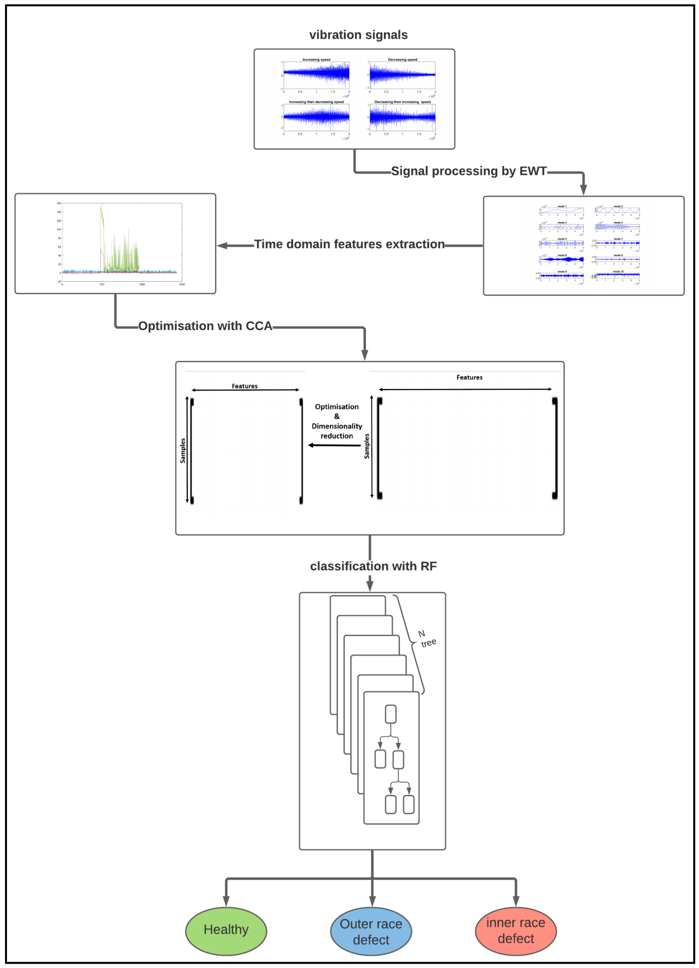

Figure 5.5: Process flowchart

The process begins with the decomposition of the vibration signals corresponding to three distinct health states. To achieve this, we employ the Empirical Wavelet Transform (EWT), a powerful technique that decomposes each signal into 10 amplitude-modulated frequency-modulated (AM-FM) modes.

The number of extracted modes is justified through a comparative analysis in which we evaluate the performance of the decomposition using different numbers of modes. By testing various modes (for example, 5, 10, and 15 modes), we determine the optimal number that provides the best trade-off between capturing relevant signal characteristics and avoiding overfitting. This comparison ensures that the chosen number of modes (10 in this case) sufficiently represents the underlying vibration patterns while maintaining computational efficiency. The decomposition step is essential, as it allows us to separate the signal into distinct components, each capturing unique characteristics of the system's vibration behavior. These components are expected to hold valuable information about the system's health condition, revealing specific changes that occur across different states.

Once the signals are decomposed into their AM-FM modes, the next step involves feature extraction. From each of the 10 modes, a set of features is calculated, including statistical measures, spectral characteristics, and other relevant signal descriptors. These features serve as a comprehensive representation of the vibration signals and are the foundation for further analysis. The extracted features provide valuable insights into the system's behavior under different health conditions.

To ensure that only the most relevant and discriminative features are utilized in the subsequent classification step, we employ a Clan-Based Cultural Algorithm (CBCA) for feature selection. CBCA is an optimization technique inspired by the process of cultural evolution, which allows for effective exploration of feature subsets. It iterates through the feature set, considering both local knowledge (individual features) and global knowledge (interrelationships between features), enabling the selection of the most informative, non-redundant features. This process reduces the dimensionality of the feature space while retaining the essential characteristics that differentiate the health states.

Once the most relevant features are selected, we move on to the final step: model training. We use the Random Forest classifier, an ensemble learning method that constructs multiple decision trees during the training phase. Random Forest is particularly suited for handling high-dimensional data, managing feature interactions, and ensuring robust classification performance. By training the model on the selected features, we aim to develop a reliable classifier capable of accurately distinguishing between the health states of the system.

To assess the effectiveness and resilience of our proposed method, we have subjected it to a rigorous validation process. This validation procedure relies on a meticulously crafted database that includes data from healthy bearings, inner race defects, and outer race defects. It contains a wide range of rotational speeds, which simulate the dynamic variations encountered in real-world scenarios. By carefully selecting this dataset for validation, we enhance our confidence in the method's ability to perform reliably and withstand the dynamic conditions commonly encountered in real-world applications.

5.1.1.3 Signal processing

Initially, distinguishing between the states of the bearings using the original vibration signatures proves to be challenging. Consequently, we initiate the process by subjecting the signals to Empirical Wavelet Transform (EWT). EWT decomposes the signals into 10 modes, as depicted in Figure 5.6, revealing substantial distinctions among the 10 modes corresponding to the three states.

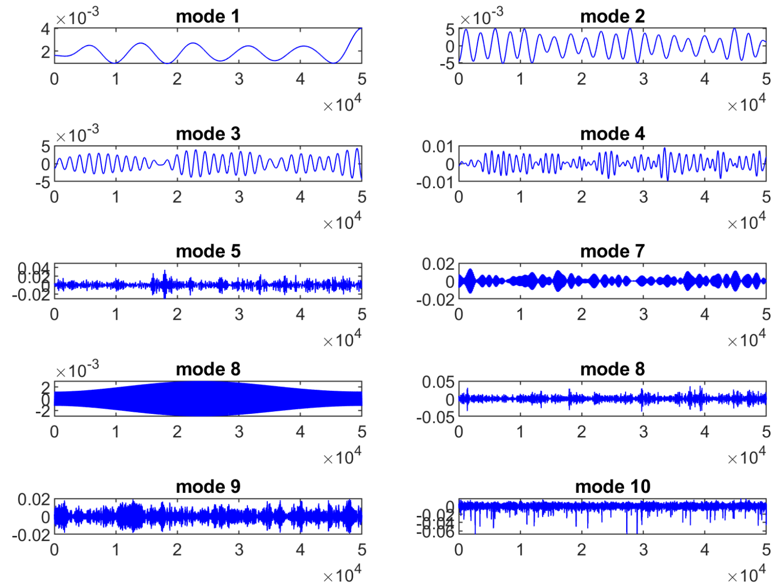

Figure 5.6: Healthy signal decomposition using EWT

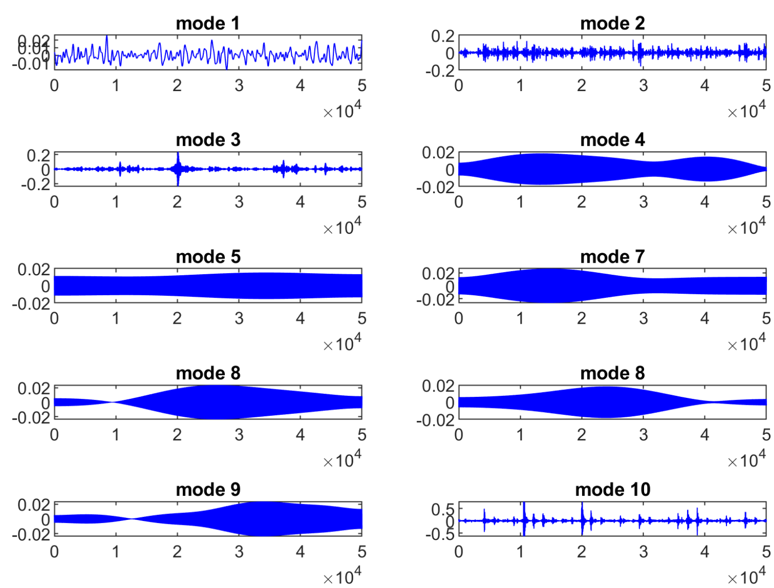

Figure 5.7: Inner signal decomposition using EWT

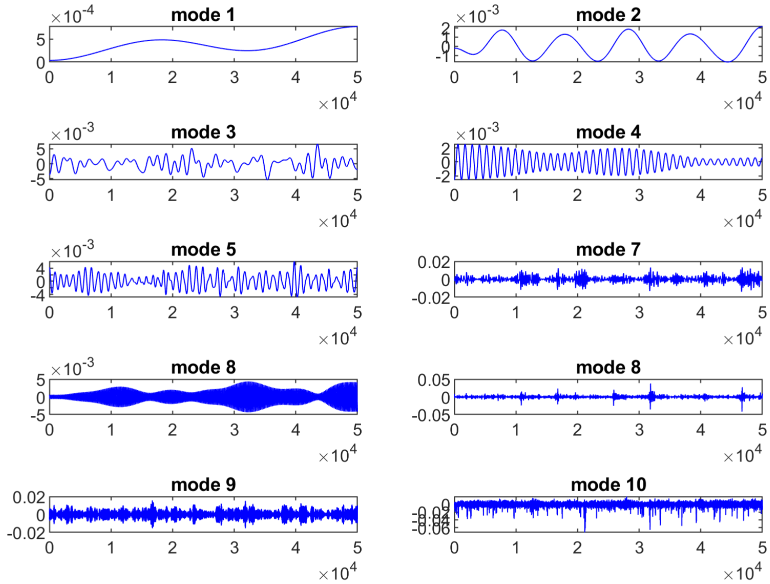

Figure 5.8: Outer signal decomposition using EWT

The selection of the number of modes is validated through a comparative analysis involving 3, 5, 7, and 10 modes, in addition to the original (unprocessed) signal. This evaluation is summarized in Table 5.1, where the results demonstrate that the decomposition into 10 modes outperforms other scenarios in terms of accuracy, achieving a remarkable accuracy rate of 100%, along with stability as indicated by a standard deviation of 0.11.

Table 5.1: Comparison between processed and unprocessed signal

			processed signal with EWT							
	Unprocessed signal		Without CCA				With CCA			
Iter	Without CCA	With CCA	3 modes	5 modes	7 modes	10 modes	3 modes	5 modes	7 modes	10 modes
1	50.35%	51.73%	97.91%	98.26%	98.26%	98.61%	98.61%	98.26%	100%	100%
2	48.96%	55.20%	98.95%	98.96%	97.22%	99.31%	98.96%	99.31%	100%	100%
3	49.31%	50.34%	98.61%	97.22%	99.65%	98.26%	97.92%	99.31%	100%	100%
4	50.35%	53.47%	98.26%	98.61%	98.61%	99.31%	98.61%	98.96%	97.92%	99.65%
5	48.26%	49.65%	98.95%	98.96%	98.61%	98.61%	98.96%	98.61%	100%	100%
6	48.96%	46.18%	97.91%	98.61%	97.91%	97.92%	97.57%	98.61%	98.61%	100%
7	53.82%	48.95%	98.61%	98.26%	99.30%	98.61	99.31%	97.57%	100%	100%
8	47.22%	55.20%	98.61%	97.57%	98.26%	97.57%	97.22%	100%	100%	100%
9	45.49%	50.69%	98.61%	98.61%	98.61%	98.96%	98.61%	99.31%	100%	100%
10	46.88%	48.26%	98.95%	98.88%	98.95%	97.57%	98.61%	98.61%	100%	100%
Mean	48.96%	50.96%	98.53%	98.39%	98.50%	98.49%	98.43%	98.86%	99.65%	99.96%
std	2.29	2.96	0.39	0.63	0.67	0.64	0.66	0.67	0.74	0.11

Subsequent to signal decomposition, we observe that the plots of the modes for the three states exhibit distinctive characteristics, with each mode encapsulating both frequency and temporal information. In our methodology, we place particular emphasis on time domain parameters. These parameters, outlined in Table 5.11, are computed for each mode.

5.1.1.4 Features Selection and Optimization

The effectiveness of any classification method relies heavily on the choice of input features. Some features may provide redundant or duplicate information, while others offer more valuable insights that carry greater weight in the classification process.

To address this issue, we've introduced an optimization step in our proposed procedure. This step is designed to ensure that we select the most informative features for our classification task. It involves the careful selection of relevant parameters and the removal of redundant ones, achieved through the Clan-based Cultural Optimization Algorithm (CCA). The CCA algorithm helps us find the best combination of features that maximize the accuracy of our classification results.

To evaluate the impact of the optimization step, we conducted a comparison between our method with optimization and the same method without optimization. The results, summarized in Table 5.1, clearly demonstrate that the optimization step leads to significant improvements in accuracy. Our proposed method with optimization consistently achieved the highest accuracy levels, highlighting the importance of eliminating redundant parameters in enhancing classification accuracy.

Furthermore, the results in Table 5.1 reveal that the optimized feature set not only improves accuracy but also enhances stability. Additionally, the optimization step reduces the number of input features required for classification, which, in turn, reduces processing time and speeds up the classification process. Based on these results, we conclude that the feature selection step is crucial for achieving high classification accuracy and improving computational efficiency. By removing redundant features, the optimization technique effectively boosts the performance of the classification method, making it more efficient and dependable.

To strengthen this conclusion, we compared the CCA algorithm with four other powerful nature-inspired optimization algorithms. This comparison involved a set of 130 features derived from 13 attributes across 10 modes. To ensure a fair comparison and visualize the effects of the algorithms, we used a K-Nearest Neighbors (K-NN) classifier, known for its simplicity and effectiveness. We conducted twenty simulations for each optimization algorithm and summarized the results in Table 5.2. This comparison provides compelling evidence of the significant improvement that the CCA algorithm brings to the classification process.

Table 5.2: Comparison between CCA, Ant, grasshopper, wolf, and firefly optimization algorithms with K-NN classifier

Algorithm	CCA		Ant		Grasshopper		Wolf		Firefly	
	Accuracy	Feat	Accuracy	Feat	Accuracy	Feat	Accuracy	Feat	Accuracy	Feat
Max	99.51%	72	98.80%	15	99.18%	66	99.17%	46	97.36%	60
Min	99.10%	51	93.20%	15	96.25%	46	96.25%	32	92.63%	60
Mean	99.40%	64	97.27%	15	98.51%	56	98.39%	46	95.63%	60
Std	0.1186		1.262		0.7164		0.8032		1.288	

The outcomes elucidated in Table 5.2 offer a revealing perspective. They showcase that the ant algorithm, while commendably compact in its feature selection, attains an accuracy of 97.27%. However, when viewed through the lens of accuracy, the Cultural Clan-based Algorithm emerges as the undisputed champion among the four algorithms. It achieves this feat by streamlining the feature pool, effectively halving the number of considerations required in the classification process, all while substantially elevating accuracy and stability levels.

This judicious culling of pertinent features begets a classification framework that is not only more efficient but also more precise—a pivotal consideration, especially in the context of industrial applications. The significance of this precision becomes all the more apparent when we recognize that classification relying on feeble classifiers driven by inconsequential features can yield erroneous outcomes, potentially misclassifying faults.

In our proposed methodology, we place our trust in the Random Forest (RF) classifier. This robust classifier is equipped to handle time domain features extracted from ten distinct modes corresponding to the three states. These features are meticulously selected by the Cultural Clan-based Optimization Algorithm. The RF classifier, reinforced by a forest of 100 trees, operates with 80% of the data dedicated to training and 20% for testing.

To substantiate the appropriateness of our RF classifier choice, we pitted it against several other well-established classifiers, including K-nearest neighbor, Decision tree, Extra tree, Extreme learning machine, and Least squares support vector machine. The comparative results are diligently laid out in the accompanying table.

Table 5.3: Comparison between classifiers

Simulation	Random Forest	K-nearest neighbor	ELM	Decision Tree	Extra Tree	LSSVM
1	100%	98.95%	50.34%	98.95%	100%	99.65%
2	100%	98.61%	74.30%	100%	99.65%	100%
3	100%	98.95%	58.31%	98.95%	99.30%	99.65%
4	100%	98.95%	66.31%	98.95%	100%	99.65%
5	100%	98.61%	61.80%	99.30%	99.30%	100%
6	100%	98.95%	83.33%	99.55%	99.30%	99.30%
7	99.65%	98.61%	55.90%	99.30%	99.65%	100%
8	100%	99.30%	56.25%	99.30%	99.65%	100%
9	100%	98.61%	63.19%	99.30%	99.65%	99.30%
10	100%	99.30%	42.70%	98.61%	100%	99.65%
mean	99.96%	98.88%	61.24%	99.22%	99.65%	99.72%
std	0.11	0.27	11.60	0.38	0.28	0.27

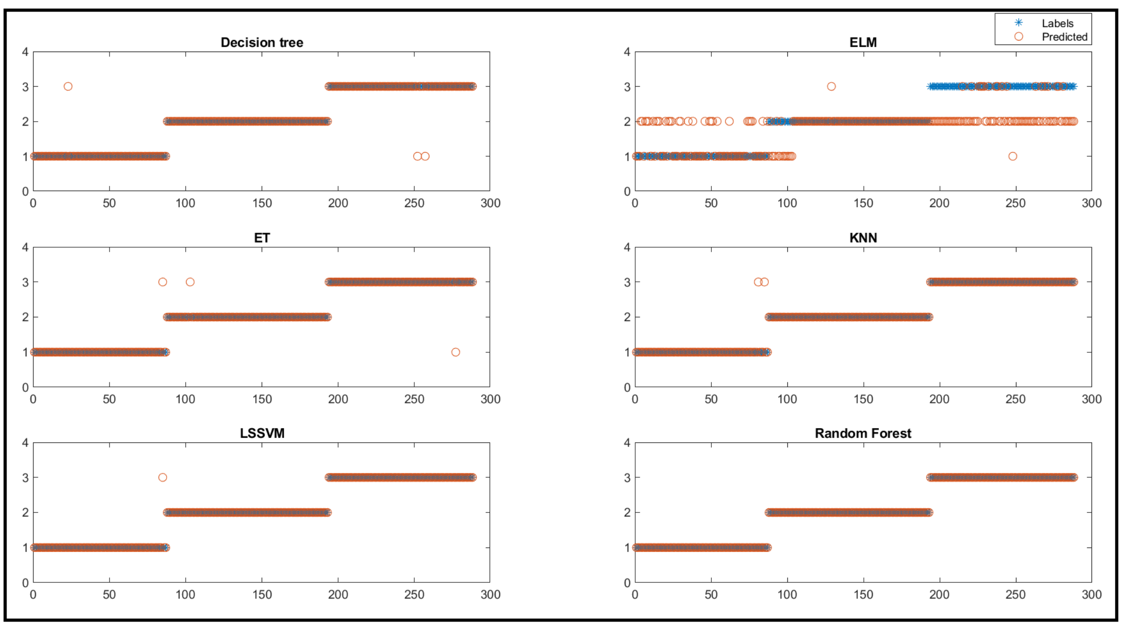

Figure 5.9: The plot of the classification using different classifiers.

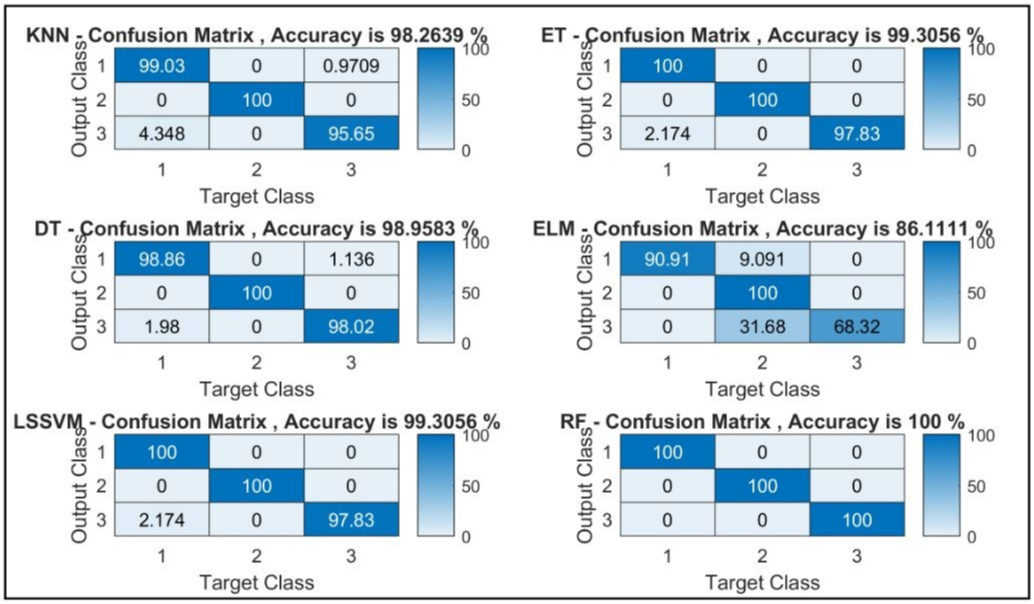

Figure 5.10: Confusion matrices of classification with different classifiers.

The results in Table 5.3 demonstrate the robustness of the "Random Forest" classifier compared to other powerful algorithms, such as ET, LSSVM, and DT. The highest accuracies belong to the RF classifier, which outperforms the rest of the classifiers regarding stability, with a standard deviation of only 0.1. Figure 5.9 depicts the predicted states versus the labels by RF and the other selected classifiers. Notably, RF achieved perfect classification with no misclassifications, as shown in the confusion matrix in Figure 5.10.

5.1.2 Rolling bearing fault feature selection based on standard deviation and random forest classifier using vibration signals

Motivation

The classification accuracy heavily depends on the quality and representativeness of the parameters used in the training process. In this context, our research introduces a novel optimization algorithm designed to enhance diagnostic accuracy by selecting the most informative and relevant parameters. The proposed algorithm synergizes the standard deviation (STD) parameter with the random forest classifier to identify the optimal parameters. The STD parameter assumes a pivotal role in ensuring the selection of parameters with the highest consistency and stability. Through the integration of this parameter with the random forest classifier, we can refine parameter selection and, consequently, elevate the accuracy of the diagnostic process.
To validate the effectiveness of our approach, we conducted experiments using three distinct databases, including one characterized by non-stationary data reflecting real-world

conditions. The non-stationary database presents a particularly formidable challenge for fault diagnosis, and our approach demonstrated superior accuracy and stability compared to existing methods. These results underscore the effectiveness and resilience of our proposed algorithm, which has the potential to substantially improve the accuracy and efficiency of fault diagnosis in real-world applications.

5.1.2.1 Databases Description

In this contribution, our proposed method was evaluated using three different databases to evaluate its effectiveness. These databases consist of vibration data collected from bearings in various health states, operating under different conditions.

Database 1

The first database utilized is **"Bearing vibration data collected under time-varying rotational speed"** used for the first contribution to assess whether our proposed algorithm can effectively handle time-varying conditions.

Database 2

The second database, MaFaulDa (Machinery Fault Data), is sourced from SpectraQuest's Machinery Fault Simulator (MFS) figure 5.11, known as Alignment-Balance-Vibration (ABVT). ABVT is a specialized system designed to provide comprehensive vibration signals along the three axes, accompanied by acoustic signals, specifically to investigate the behavior of faulty bearings with various defective parts. These defective parts include the outer track, the rolling element, and the inner track.

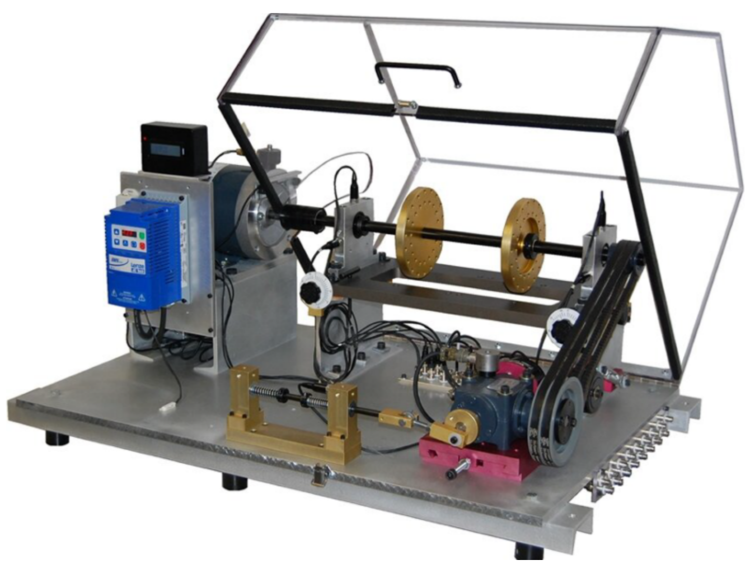

Figure 5.11: The SpectraQuest machine fault simulator.

To summarize the contents of this database, we present the following details, with distinct sequences for each bearing and two different positions:

- Bearing placed between the rotor and the motor (Underhang).
- Rotor positioned between the bearing and the motor (Overhang).

Below, a table provides an overview of the database's characteristics:

- having the bearing between the rotor and the motor (underhang).
- having the rotor between the bearing and the motor (overhang).

Table 5.4: Characteristics of the second database

State	defect element	masses	sequence
Normal	-	-	49
Underhang	outer track	0g	49
		6g	48
		20g	49
		35g	42
	Rolling element	0g	49
		6g	49
		20g	49
		35g	37
	Inner track	0g	50
		6g	49
		20g	49
		35g	38
Overhang	outer track	0g	49
		6g	49
		20g	49
		35g	41
	Rolling element	0g	49
		6g	49
		20g	49
		35g	41
	Inner track	0g	49
		6g	43
		20g	25
		35g	20

The bearings used in this database are equipped with eight balls, each with a diameter of 7.145 mm. The data were sampled at a rate of 50 kHz, with each sequence lasting

5 seconds. The operating frequency ranges from 737 to 3686 rpm, with incremental steps of approximately 60 rpm. MaFaulDa provides a rich and comprehensive dataset for studying machine failure diagnosis and condition monitoring, particularly in the context of bearing behavior under various conditions and defects.

Database 3

The third database used for the validation of our suggested algorithm is from the Case Western Reserve University Data Center website. It consists of vibration test data for ball bearings, both in normal and faulty conditions. The tests were performed using a 2 hp Reliance Electric motor, with acceleration data collected at both near and remote locations from the motor bearings. The test conditions, including motor operational parameters and bearing fault status, are carefully documented for each experiment.
Faults in the motor bearings were artificially introduced using electro-discharge machining (EDM), with fault sizes ranging from 0.007 to 0.040 inches in diameter. These faults were placed at three locations: the inner raceway, the rolling element (ball), and the outer raceway. After the introduction of the fault, the bearings were re-installed in the test motor and vibration data was recorded for motor loads ranging from 0 to 3 horsepower (motor speeds between 1797 and 1720 RPM).

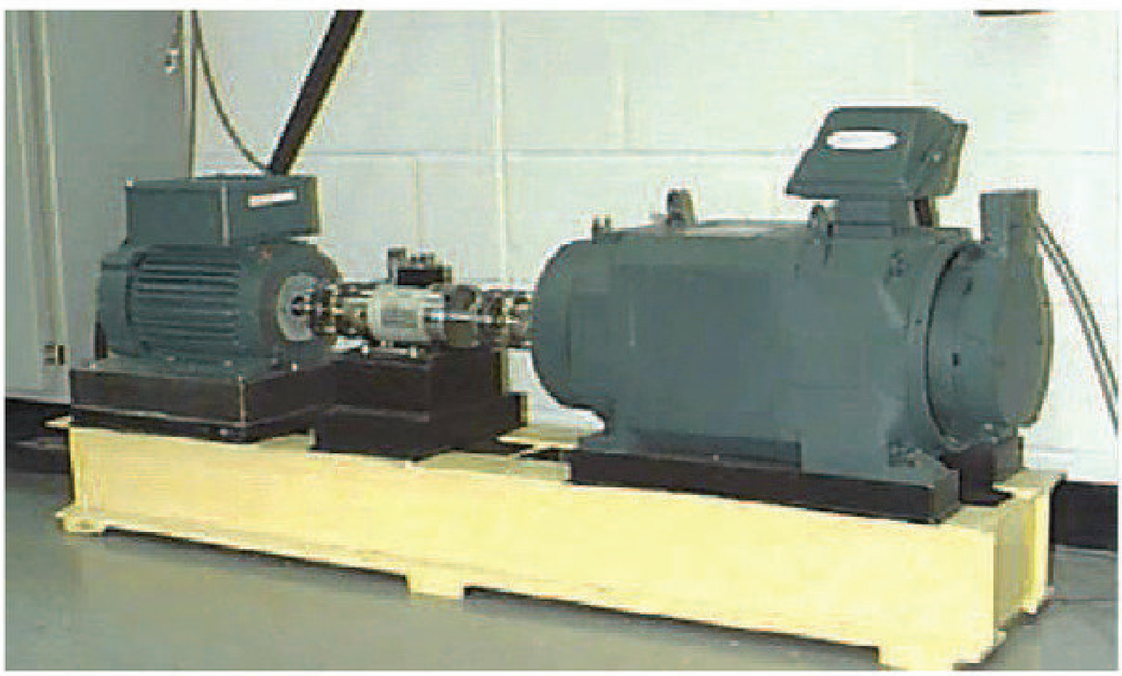

Figure 5.12: Bearing test rig of Case Western Reserve University Data Center.

5.1.2.2 Experimental procedure

The flowchart in Fig. 5.13 illustrates the method proposed for feature selection in the bearing diagnosis process.

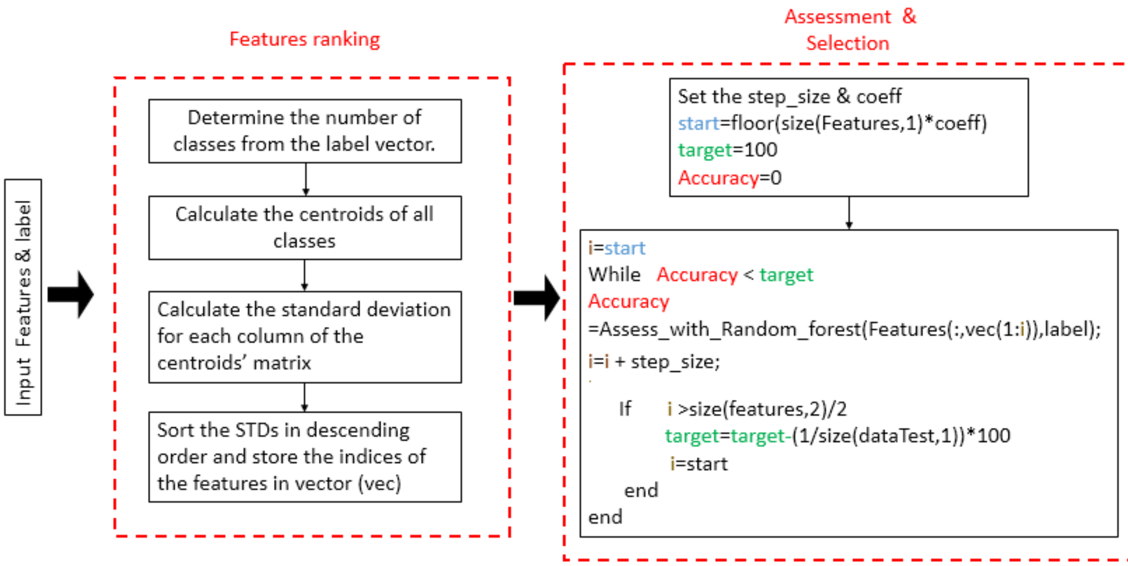

Figure 5.13: Flowchart of the proposed method

The following steps outline the proposed method:

1. Determine the number of classes and their corresponding number of samples.

2. Calculate the centroid coordinates for each class. C_K is the centroid of an arbitrary class K, and we calculate it as follows:

$$C_K = \frac{\sum Samples}{size(DataTest)} \tag{5.1}$$

Where:

$$Class_K = \begin{bmatrix} x_{1,1} & x_{1,2} & .... & x_{1,N} \\ x_{2,1} & x_{2,2} & .... & x_{2,N} \\ .... & .... & .... & .... \\ .... & .... & .... & .... \\ x_{l,1} & x_{j,2} & .... & x_{l,N} \end{bmatrix}$$

l is the size of samples in class K and N is the number of features.
we expand equation 5.1 into:

$$C_K = \frac{(x_{1,1}, ..., x_{1,N}) + ... + (x_{l,1}, ..., x_{j,N})}{l} \tag{5.2}$$

$$C_K = (\frac{x_{1,1} + ... + x_{l,1}}{l}, ..., \frac{x_{1,N} + ... + x_{l,N}}{l}) \tag{5.3}$$

$$C_K = (\frac{\sum_{j=1}^{l} x_{j,1}}{l}, ..., \frac{\sum_{j=1}^{l} x_{j,N}}{l}) \tag{5.4}$$

Then, the centroid's coordinates are equal to the means of the corresponding class's columns as shown in figure.5.14.

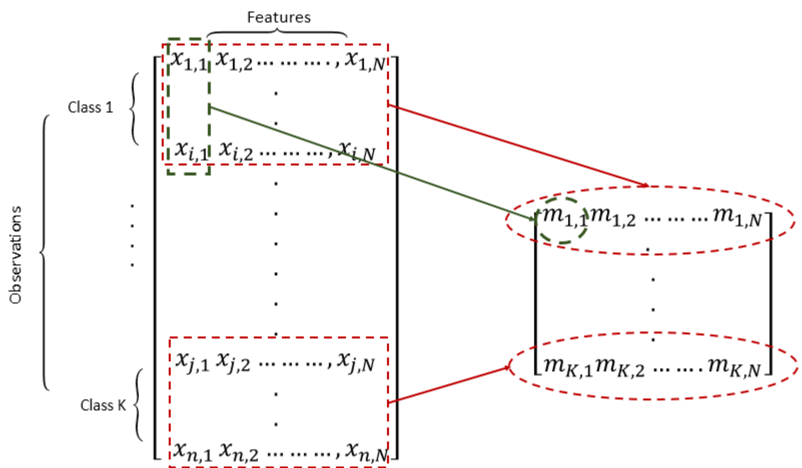

Figure 5.14: The calculation of the centroids' coordinates

3. Compute the standard deviation using equation.5.5 for each column of the centroids matrix.

$$Centroids = \begin{bmatrix} m_{1,1} & m_{1,2} & .... & m_{1,N} \\ m_{2,1} & m_{2,2} & .... & m_{2,N} \\ .... & .... & .... & .... \\ .... & .... & .... & .... \\ m_{p,1} & m_{p,2} & .... & m_{p,N} \end{bmatrix}$$

Where p is the number of classes,

$$STD_j = \sqrt{\frac{\sum_{i=1}^{p}(m_{i,j} - M_j)^2}{p}} \tag{5.5}$$

And,

$$M_j = \frac{\sum_{i=1}^{p} m_{i,j}}{p} \tag{5.6}$$

We obtain a vector **S** of N STD value.

$$S = \begin{bmatrix} STD_1 & STD_2 & .... & STD_N \end{bmatrix}$$

4. Sort the vector **S** in descending order and save the indices of the corresponding features in vector **V**.

5. Execute a sequential forward selection on the indices' vector **v** and assess the performance of the corresponding features with the Random forest classifier. The process starts from **start** and stops once the accuracy reaches the **Target**.

   - **start** is the initial index for the sequential selection. It helps to preserve time by considering the indices from 1 to **start** highly significant features.

     $$start = size(features) \times coef \tag{5.7}$$

     In our application, we set the coef=5%, assuming that the first 5% are relevant features.

   - **Target** is initially equal to 100%, it is used as a termination criterion in the selection process. If the intended accuracy is not reached with less than half of the features, the Target is adjusted using equation5.8 to provide the highest possible accuracy.

     $$Target = Target - (\frac{1}{size(DataTest)} \times 100) \tag{5.8}$$

5.1.2.3 Results and analysis

On three datasets of rolling bearings collected under different conditions, we apply three signal processing techniques: Empirical Wavelet Transform (EWT), Empirical Mode Decomposition (EMD), and Maximal Overlap Discrete Wavelet Packet Transform (MODWPT). For the resulting signals, we compute the features listed in table 5.11.
Then, we apply the STD-RF selection method to the obtained feature set. We consider the execution time, the number of features opted for, and the obtained accuracy.
We decompose the signal into 10 Amplitude Modulation-Frequency Modulation (AM-FM) modes for the EWT technique.

The number of intrinsic modes functions (IMF) for the EMD technique varies between 12 and 16 for the three databases. We chose 16 as the maximum value to adjust the features matrix without losing any information.

For the three datasets, the MODWPT technique extracts 16 terminal nodes.

Table 5.5: Table depicting the performance of the STD-RF algorithm with EWT.

TF	Dataset1		Dataset2		Dataset3	
	130		520		130	
	selected features	Execution time (s)	selected features	Execution time (s)	selected features	Execution time(s)
1	14	10.20	49	13.44	17	17.71
2	23	42.19	50	12.57	17	12.77
3	14	10.28	49	16.02	17	24.26
4	14	9.95	49	10.70	20	22.49
5	16	17.53	50	13.86	21	26.40
6	14	7.01	49	15.29	20	34.09
7	16	17.53	49	11.30	17	12.83
8	16	12.61	49	12.10	20	29.96
9	19	24.65	49	11.11	23	42.90
10	20	27.06	49	15.80	22	27.39
mean	17	17.90	49	13.21	19	25.08
STD	3.09	-	0.42	-	2.27	-
% of selected features	13.07%	-	9.42%	-	14.61%	-

Table 5.6: Table depicting the performance of the STD-RF algorithm with EMD.

TF	Dataset1		Dataset2		Dataset3	
	208		832		208	
	selected features	Execution time (s)	selected features	Execution time (s)	selected features	Execution time(s)
1	7	16.40	132	301.20	13	22.43
2	7	31.82	132	311.70	17	50.80
3	11	25.79	130	283.97	13	19.71
4	10	25.89	129	294.46	14	47.77
5	6	13.93	132	303.46	13	15.84
6	14	46.12	130	281.63	13	18.66
7	12	34.74	132	291.80	13	16.62
8	18	62.31	132	348.93	13	14.74
9	7	15.46	132	338.05	15	32.55
10	12	31.31	135	330.55	13	33.77
mean	11	27.78	132	308.57	14	27.28
STD	3.8	-	1.64	-	1.33	-
% of selected features	5.28%	-	15.86%	-	6.73%	-

Table 5.7: Table depicting the performance of the STD-RF algorithm with MODWPT.

TF	Dataset1		Dataset2		Dataset3	
	208		832		208	
	selected features	Execution time (s)	selected features	Execution time (s)	selected features	Execution time(s)
1	21	14.92	83	125.02	20	41.85
2	21	18.03	84	67.85	20	13.10
3	20	8.46	83	49.53	20	10.82
4	20	8.99	83	45.63	20	11.27
5	20	8.25	83	48.69	20	9.30
6	20	7.29	83	50.43	20	12.42
7	20	9.99	83	52.07	21	14.37
8	20	8.07	83	54.84	20	11.31
9	20	8.91	84	64.93	20	27.78
10	20	10.10	83	55.30	21	18.64
mean	20	10.30	83	61.42	20	17.08
STD	0.42	-	0.42	-	0.42	-
% of selected features	9.61%	-	9.97%	-	9.61%	-

Tables 5.5, 5.6, 5.7 present the features selected by the STD-RF method for three databases processed by EWT, EMD and MODWPT, respectively.
The tables contain the results of ten simulations using the STD-RF selection method and the execution time for each case. As we can see, our proposed method could reduce

Table 5.8: Number of features selected and accuracies obtained by different optimization algorithms on database 1

simulation	Squirrel	Wolf	BDE	GOA	SA	**STD-RF**
1	65	50	124	55	51	**16**
2	57	56	120	59	53	**17**
3	54	50	107	58	51	**15**
4	71	35	122	68	48	**20**
5	69	42	123	55	57	**14**
6	59	46	117	59	59	**16**
7	66	32	72	52	53	**20**
8	70	44	122	59	60	**15**
9	68	54	97	62	58	**16**
10	73	40	126	64	53	**21**
mean	65	46	117	60	54	**17**
Accuracy	98.61%	100%	100%	99.06%	99.02%	**100%**

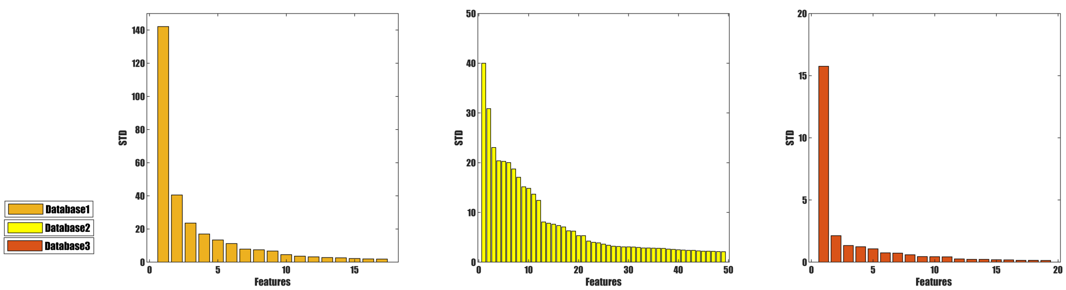

Figure 5.15: STD values of the selected features for the three databases processed by EWT

the sets of features to less than 15% using the EWT, less than 16% using EMD and less than 10% while using MODWPT and hence help to boost the diagnosis process speed. From tables 5.5, 5.6 and 5.7, we observe that the STD-RF's results remain in the same scope despite the signal decomposition technique tool involved in the data processing. Also, the number of selected features for the ten simulations affirms the stability of our method in both quantity and quality terms because of the features' ordering at the beginning of the process. We put our method in comparison with five robust optimization algorithms in the bearing diagnosis field. Our method was compared to the squirrel search algorithm, grey wolf optimization algorithm, binary coded differential evolution (BDE), Grasshopper optimization algorithm (GOA), and simulated annealing (SA).

Table 5.9: Number of features selected and accuracies obtained by different optimization algorithms on database 2

simulation	Squirrel	Wolf	BDE	GOA	SA	**STD-RF**
1	251	112	398	65	61	**49**
2	247	125	442	58	57	**49**
3	236	148	423	61	60	**50**
4	231	143	415	62	60	**49**
5	244	108	224	69	67	**50**
6	244	113	441	65	63	**49**
7	240	131	469	67	71	**49**
8	251	182	437	69	67	**49**
9	210	192	458	60	64	**49**
10	229	169	248	68	63	**49**
Mean	238	142	395	64	63	**49**
Accuracy	99.46%	99.78%	99.78 %	99.71%	99.67%	**99.89%**

Table 5.10: Number of features selected and accuracies obtained by different optimization algorithms database 3

simulation	Squirrel	Wolf	BDE	GOA	SA	**STD-RF**
1	64	38	110	54	48	**17**
2	69	45	94	60	48	**17**
3	77	50	104	52	53	**17**
4	65	37	75	62	52	**20**
5	64	44	97	58	54	**21**
6	66	32	72	52	53	**20**
7	240	131	469	67	71	**17**
8	251	182	437	69	67	**20**
9	210	192	458	60	64	**23**
10	229	169	248	68	63	**22**
66	42	96	56	53	53	**19**
Accuracy	99.46%	99.7%	99.78%	99.71%	99.67%	99.89%

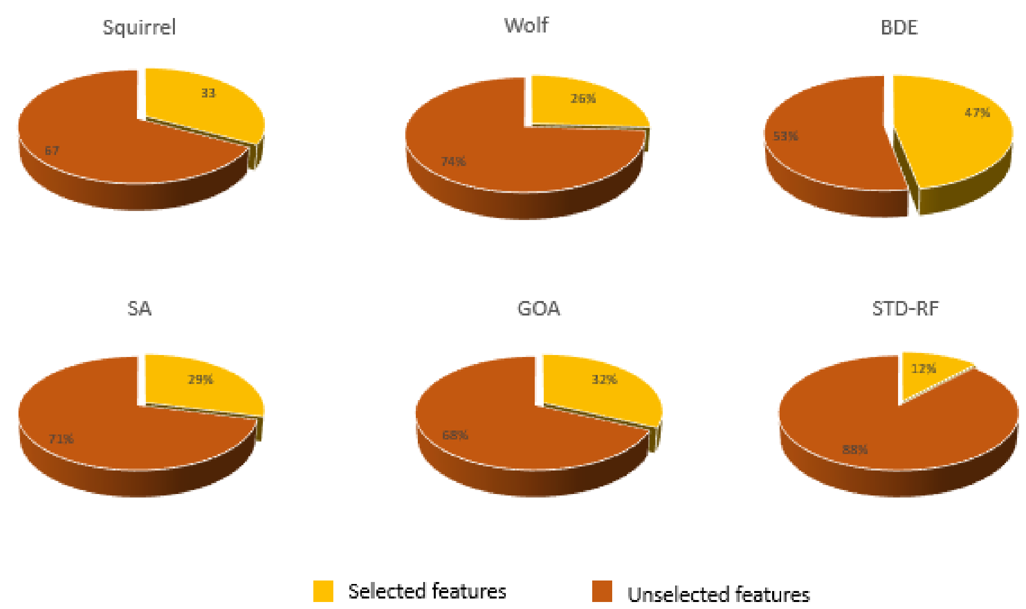

Figure 5.16: Comparison graph illustrating the percentage of selected features by different optimization methods for the first dataset.

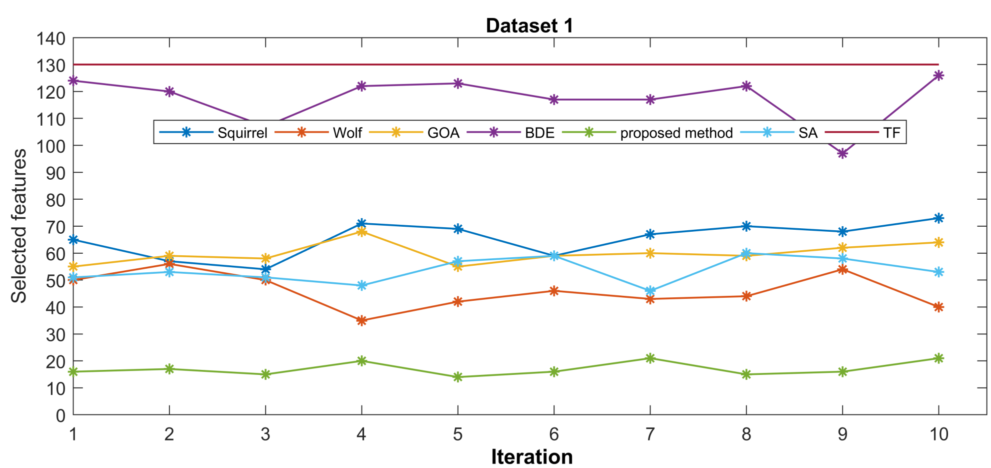

Figure 5.17: Comparison graph illustrating the number of selected features by different optimization methods for the first dataset.

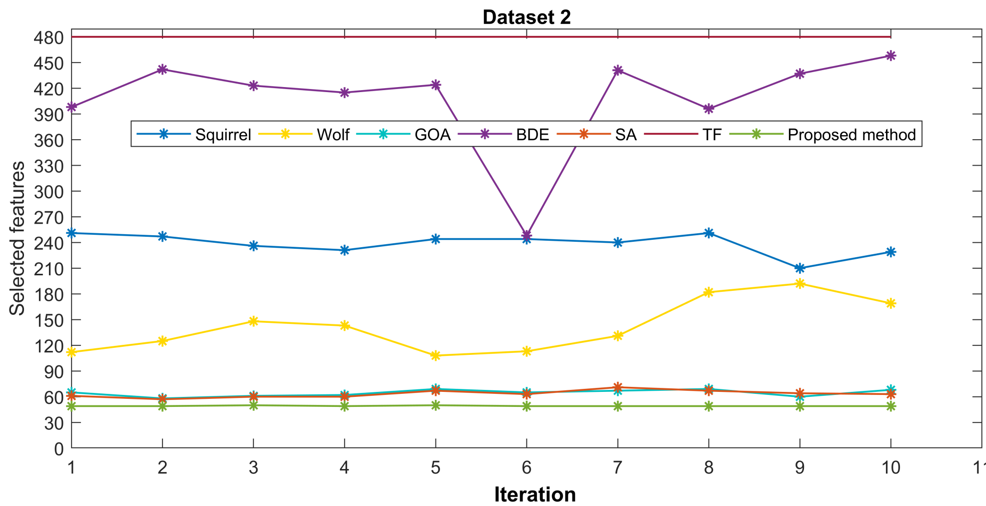

Figure 5.18: Comparison graph illustrating the number of selected features by different optimization methods for the second dataset

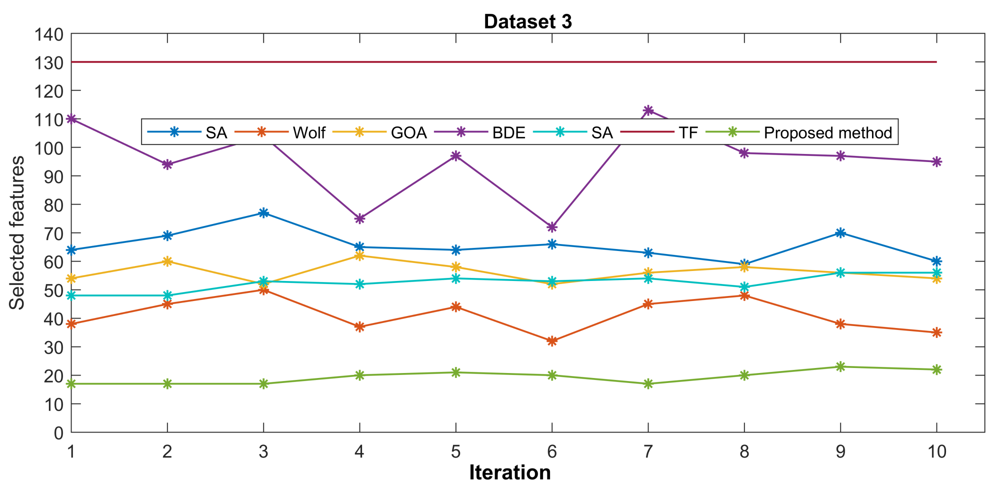

Figure 5.19: Comparison graph illustrating the number of selected features by different optimization methods for the third dataset

Tables 5.14, 5.15 and 5.16 demonstrate that the **STD-RF** selection method exhibits superior performance compared to the other algorithms concerning the accuracy and the number of selected features. Additionally, the table reveals that for the same dataset, if n simulations yield the same number of selected features, this implies that the n sets are identical, as the vector of indices **v** is consistently ordered irrespective of the initial arrangement of the data. This independence of the output from the initial data's position enhances the system's stability, unlike algorithms where the search procedure is initiated randomly and is influenced by the order of the features, leading to varying feature sets in different simulations.

Figure 5.16 provides a clear representation of the power of our proposed method in feature selection using the first database processed by the EWT technique. It reduces the number of parameters involved in the classification process to just **12%** without affecting the classification's accuracy.
The accuracies listed in TableS 5.14, 5.15 and 5.16 were assessed using the RF classifier, We have tested our proposed method using the holdout cross-validation and we repeated it 10 times as an explicit 10-fold cross-validation to detect any hidden variance between the 10 folds, and this because the k-fold cross validation provides the average of the k simulations without giving an idea about the stability of the system. We split the data randomly into 80:20 to have a larger amount of data for testing, and we repeated the process 10 times then we calculated both the average and the STD.
Figures 5.17, 5.18 and 5.19 illustrate clearly the strength of our proposed method in reducing the size of the features set compared to the total features(TF) and the outputs of strong optimization algorithms as the squirrel, grey wolf, BDE and others, without affecting the accuracy of classification as seen in tables 5.14, 5.15 and 5.16.
The accuracy of fault diagnosis can be notably enhanced by utilizing feature ranking. Figure 5.15 represents a histogram, which illustrates the selected features in the three datasets processed by EWT. These features are arranged in a particular order corresponding to their importance, which is determined based on their standard deviation (STD). The histogram visually depicts the distribution of the selected features and their relative significance.

5.1.3 Multi-fault Bearing Diagnosis Under Time-Varying Conditions Using Empirical Wavelet Transform, Gaussian Mixture Model and Random Forest Classifier

Motivation

Our research journey was motivated by a compelling need to address complex real-world challenges in fault diagnosis and classification. We turned to the EM-GMM (Expectation-Maximization Gaussian Mixture Model) clustering algorithm to infuse the unsupervised perspective into our feature selection process. This choice was driven by our desire to unravel hidden insights within our data and, in turn, enhance the effectiveness of our diagnostic approach. The unsupervised clustering facilitated by the EM-GMM algorithm revealed latent patterns and relationships that traditional supervised techniques might overlook. We recognized the potential of these insights to revolutionize our understanding of fault detection. Therefore, we selected features that emerged from this process, confident that they held the key to more accurate and robust classification.
Our commitment to addressing real-world scenarios led us to choose a database that replicated the dynamic, time-varying conditions often encountered in practical applications. This database intentionally included combined fault cases to reflect the complexity of actual operational environments.

Through our research, we aimed not only to enhance the state-of-the-art in fault diagnosis but also to empower industries facing intricate challenges. By integrating unsupervised clustering, feature selection, and classification within the context of dynamic, combined fault scenarios, our work aspired to offer practical solutions and advance the field in meaningful ways.

5.1.3.1 Database

The database selected for this study is essentially the same as the first database but includes more complex cases. The dataset consists of vibration signals collected from bearings under various health conditions, operating at time-varying rotational speeds [111]. In total, there are 60 datasets, with each dataset corresponding to two experimental settings: bearing health condition and varying speed condition. The bearing health conditions are as follows: (i) healthy, (ii) faulty with an inner race defect, (iii) faulty with an outer race defect, (iv) faulty with a ball defect, and (v) faulty with combined defects on the inner race, outer race, and ball. The rotational speed conditions include: (i) increasing speed, (ii) decreasing speed, (iii) increasing followed by decreasing speed, and (iv) decreasing followed by increasing speed. Consequently, there are 20 distinct experimental cases. To ensure data reliability, three trials are conducted for each experimental setting, resulting in a total of 60 datasets. Each dataset contains two channels: 'Channel 1', which represents vibration data measured by the accelerometer, and 'Channel 2', which represents rotational speed data measured by the encoder. All data are sampled at a frequency of 200,000 Hz, with a sampling duration of 10 seconds. The encoder's CPR (Cycles Per Revolution) is 1024.

5.1.3.2 Experimental procedure

Figure 5.20 summarizes the process of our proposed method for the classification of bearing faults. Our procedure consists of three pivotal steps. In the initial step, we employ the Empirical Wavelet Transform (EWT) to extract the AM-FM (Amplitude Modulation-Frequency Modulation) modes from the vibration signatures. This powerful technique excels at precisely decomposing the signal into a predefined number of modes. In our study, we opt for 10 modes, as visually depicted in Figure 5.21, as this choice effectively captures the intricate characteristics inherent in the vibration signatures.

Subsequently, we calculate a set of features outlined in Table 5.11 for each of these ten modes. This process results in a total of 170 attributes, providing a comprehensive and nuanced understanding of the underlying characteristics within the vibration signatures. This in-depth analysis empowers us to discern subtle variations and patterns that may carry diagnostic significance.

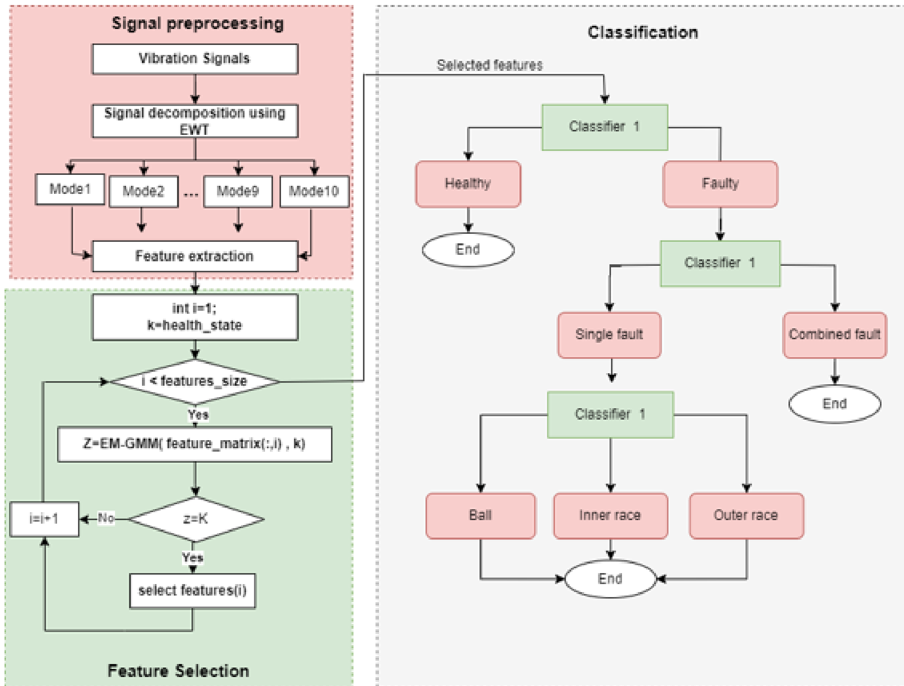

Figure 5.20: flowchart of the classification process.

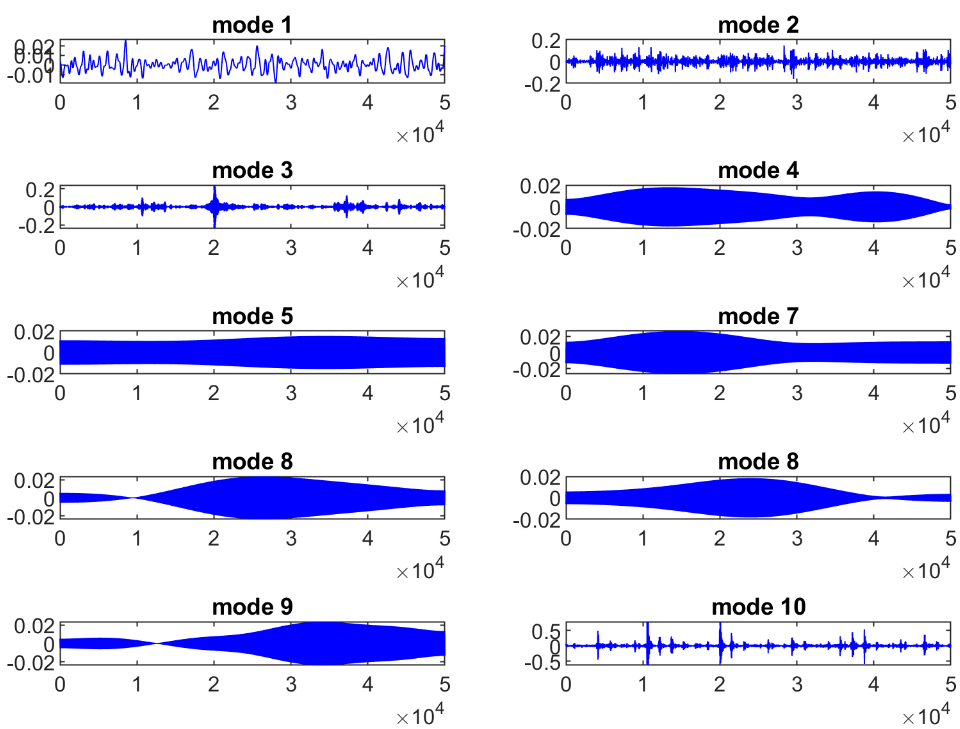

Figure 5.21: Extracted AM-FM modes using the EWT for bearing with faulty inner race.

In the second pivotal step of our procedure, we turn to the powerful EM-GMM (Expectation-Maximization Gaussian Mixture Model) clustering algorithm to unravel the most pertinent set of features for each of the three distinct stages artfully depicted in Figure 5.23. Our approach involves a meticulous examination of the data at each level independently, seeking to pinpoint the attributes that exhibit the highest discriminatory potential among the diverse classes under consideration.

As we delve into this feature selection process, we feed the extracted features sequentially to the EM-GMM algorithm. For each set of features, the algorithm is applied, and we select only those elements that can effectively distinguish the exact number of predefined groups, ensuring that the resulting features are truly relevant to the task at hand. This iterative process allows us to refine the feature set step-by-step, selecting only those features that contribute to clearly separating the data into well-defined, meaningful clusters.

The goal of this process is to identify those distinctive features that can effectively segregate the data into precisely delineated groups. This process is visually exemplified in Figure 5.22, where the features' ability to delineate the distinct clusters becomes evident. The final set of selected features is one that maximizes the algorithm's ability to distinguish between the different stages, thereby enhancing the overall performance of the subsequent modeling and analysis steps.

It's worth noting that this feature selection procedure is not a one-size-fits-all approach. Rather, it is repeated for each of the three stages, custom-tailored to the unique characteristics and intricacies of each level. This iterative and comprehensive analysis ensures that we capture the discriminative features that are most relevant and informative at every stage of our diagnostic journey.

By dedicating attention to the specifics of each stage and carefully selecting the features that best serve the diagnostic purpose, our methodology aims to enhance the precision and effectiveness of our fault identification process, ultimately contributing to more accurate and reliable outcomes.

Table 5.11: Table of Extracted Features

Feature	Definition	Equation/Description
Maximum	The highest value in the signal.	$Max = \max(x)$
Minimum	The lowest value in the signal.	$Min = \min(x)$
Median	The middle value when the data is sorted.	Calculate the median of x.
Peak to Peak	The difference between the maximum and minimum values.	$Peak2Peak = Max - Min$
Root Mean Square	The square root of the average of the squared values.	$RMS = \sqrt{\frac{1}{N}\sum_{i=1}^{N} x_i^2}$
Mean	The average value of the signal.	$Mean = \frac{1}{N}\sum_{i=1}^{N} x_i$
Standard Deviation	A measure of the dispersion of data points.	$STD = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i - Mean)^2}$
Kurtosis	A measure of the "tailedness" of the data distribution.	$Kurtosis = \frac{N\sum\limits_{i=1}^{N}(x_i-\overline{x})^4}{\left[\sum\limits_{i=1}^{N}(x_i-\overline{x})^2\right]^2}$
Crest Factor	The ratio of the peak value to the RMS value.	$Peak2RMS = \frac{Max}{RMS}$
Skewness	A measure of the asymmetry of the data distribution.	$Skewness = \frac{\sum\limits_{i=1}^{N}(x_i-\overline{x})^3}{(n-1)\sigma^3}$
Variance	The average of the squared differences from the Mean.	$VAR = \frac{1}{N}\sum_{i=1}^{N}(x_i - Mean)^2$
Root Sum of Squares	The square root of the sum of the squares of the values.	$Rssq = \sqrt{\sum_{i=1}^{N} x_i^2}$
Energy	The total energy of the signal.	$EN = \sum_{i=1}^{N} x_i^2$
Peak Frequency	The frequency corresponding to the maximum amplitude in the spectrum.	Find the frequency at which the spectrum has its maximum value.
Mean Frequency	The weighted average of the frequencies in the spectrum.	Calculate the weighted average frequency of the spectrum.
Median Frequency	The frequency below which half of the energy is contained in the spectrum.	Find the frequency such that cumulative energy is 50%.
Signal-to-Noise Ratio (SNR)	The ratio of the signal power to noise power in decibels.	$SNR_{\text{dB}} = 10 \cdot \log_{10}\left(\frac{P_s}{P_n}\right)$

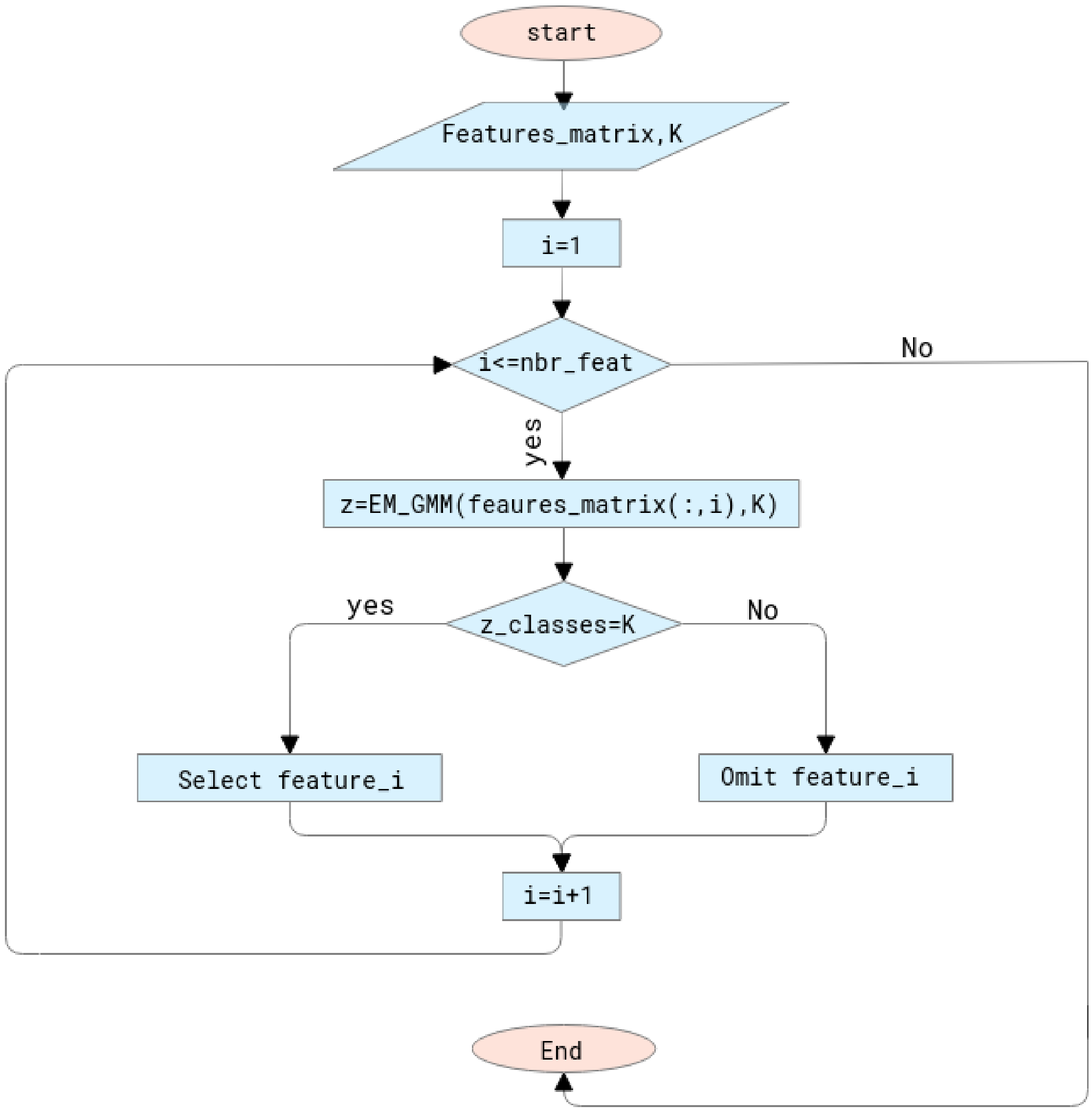

Figure 5.22: Flowchart of features selection using EM-GMM

The final step in our comprehensive process involves harnessing the three meticulously selected feature sets, denoted as SF_i, to construct three distinctive and independent classifier models, which we designate as MDL_i. These models are meticulously tailored to accommodate the unique characteristics of the data within each set and are trained using the robust Random Forest classifier. Each classifier, MDL_i, is designed to address a specific set of classes, precisely defined by $Label_i$. This strategic segregation of classes allows us to encapsulate the nuanced distinctions within the data. It's important to highlight that these classifiers are intricately nested within our overarching algorithm, as elegantly detailed in Algorithm 1. This nesting concept is visually

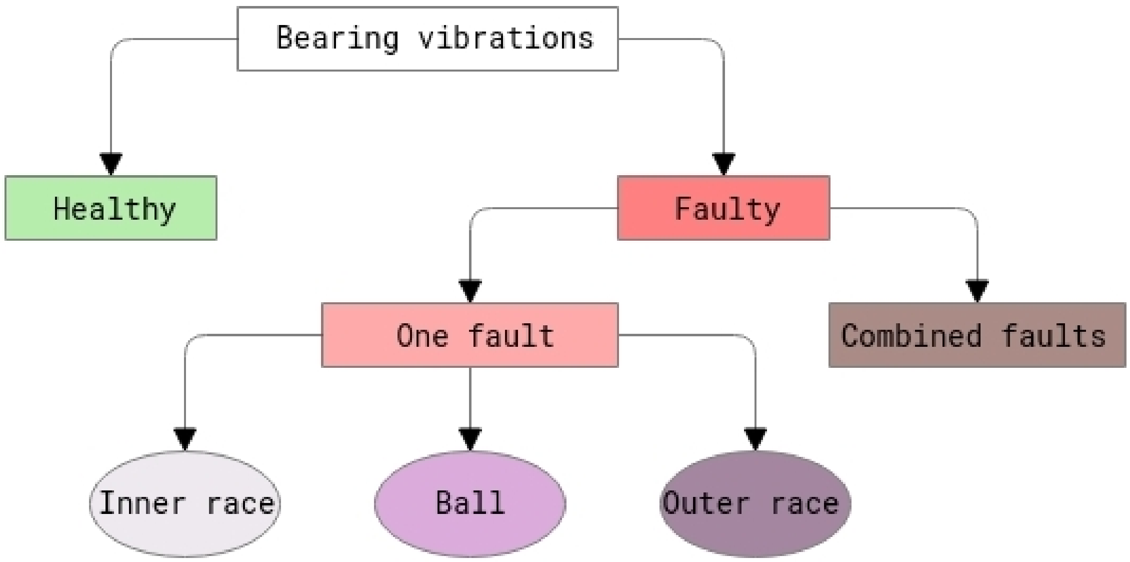

Figure 5.23: Multi-stage classification

elucidated in the illustrative Figure 5.23, where the interplay of the classifiers is represented. This multistage classification framework is designed to provide a holistic and nuanced approach to fault identification. By employing distinct classifiers tailored to different subsets of classes, we empower our methodology to excel in recognizing and distinguishing between intricate fault patterns.

Algorithm 1 Multi-stage classification pseudo-code

Input: SF_1, SF_2, SF_3, $Data\_Test$,$Label\_Test$, MDL_1, MDL_2, MDL_3

Output: *Accuracy*

```
Label_1 = [healthy, faulty]
Label_2 = [one_fault, combined_faults]
Label_3 = [inner_race, outer_race, ball]
for i = 1 : size(data_Test, 1) do
    predicted(i) = MDL_1(Data_Test(i, SF_1))
    if (predicted(i) == faulty) then
        predicted(i) = MDL_2(Data_Test(i, SF_2))
        if (predicted(i) == one_fault) then
            predicted(i) = MDL_3(Data_Test(i, SF_3))
        end if
    end if
end for
correct ← 0
for i = 1 : size(Data_Test, 1) do
    if (predicted(i) == Label_Test(i)) then
        correct ← correct + 1
    end if
end for
Accuracy ← correct/size(data_Test, 1)
```

False alarm detection using Binary Classification:

First, we perform binary classification using the Random Forest algorithm for three distinct scenarios: the inner race, outer race, and ball cases. The features selected for this classification are determined using the Gaussian Mixture Model (GMM). Subsequently, the results of this classification process are compiled and presented in a comprehensive format within table 5.12. This table serves as a concise summary, offering insights into the outcomes of our analysis for each of the aforementioned cases.

Table 5.12: Binary classification results

Element	Inner Race	Outer Race	Ball
Accuracy	100%	100%	100%

The results obtained from the binary classification process exhibit a high degree of accuracy, underscoring the effectiveness of the initial classification. In the subsequent step, we extend our analysis to encompass combined fault scenarios, which involve multiple fault types coinciding. This expanded assessment includes the combination of faulty conditions alongside healthy data. To address this complex scenario, we employ a cascading approach that integrates all three previously trained classifiers. Algorithm 2 outlines the procedure for this multi-classifier cascade.
The rationale behind employing this cascade is to leverage the collective detection capabilities of the individual classifiers. In the case of combined faults, it becomes crucial to ensure that all three parts—inner race, outer race, and ball—demonstrate some form of defect. Therefore, by employing this cascading approach, we aim to enhance the accuracy and reliability of fault detection, as all three components must be deemed defective for the combined fault scenario to be detected successfully.

Algorithm 2 Combined fault detection using binary classifiers

Input: $Inner\_race$, $Outer\_race$, $Ball$, $Label$.
Output: $Accuracy$

```
if Inner_race == defected ∧ Outer_race == defected ∧ Ball == defected then
    predicted = combined
else if Inner_race == normal ∧ Outer_race == normal ∧ Ball == normal then
    predicted = normal
else
    predicted = undefined
end if
Correct = 0
for i = 1 : size(Label) do
    if predicted(i) == Label(i) then
        Correct = Correct + 1
    end if
    Accuracy = Correct/size(Label)
end for
```

Table 5.13 provides a comprehensive overview of the results acquired through the utilization of the binary classifiers within the context of the combined fault scenario. In this extended analysis, we consider situations where multiple fault types coexist, thus presenting a more intricate and realistic assessment of the system's performance.

To obtain these results, we apply the same rigorous binary classification techniques employed previously for individual fault types (inner race, outer race, and ball). However, in this scenario, the classifiers are challenged to detect and differentiate between various fault combinations, showcasing their adaptability and robustness in the face of complex real-world conditions.

The data presented in Table 5.13 is instrumental in evaluating the overall effectiveness and reliability of our classification system when confronted with combined fault scenarios. It offers valuable insights into the system's ability to identify and classify complex faults accurately.

Table 5.13: Confusion matrices for the combined fault case using the 3 binary classifiers

Classifiers	Inner Race		Outer Race		Ball	
	Normal	Faulty	Normal	Faulty	Normal	Faulty
Normal	100%	0%	100%	0%	100%	0%
Combined	13%	87%	78%	22%	0%	100%

The precision of our three classification models when applied to healthy data was truly commendable. However, the true challenge emerged when we turned our attention to the combined fault dataset, where each classifier was tasked with identifying specific faults within distinct components of the system. Here's a detailed breakdown of their performance in this complex scenario:

Ball Classifier: The ball classifier emerged as the standout performer in our evaluation. It demonstrated impeccable precision in detecting any damage within bearing combinations that included the ball component. Its ability to identify issues in this context was virtually flawless.

Inner Race Classifier: In stark contrast, the inner race classifier encountered significant challenges. It struggled to correctly classify a substantial portion of the defective bearing combinations, accounting for a notable 13% of the total dataset. This indicates that it had difficulty distinguishing between normal and faulty conditions in the inner race component.

Outer Race Classifier: The outer race classifier, too, faced difficulties when presented with the combined fault cases. It managed to achieve accurate classifications for only 22% of the combined fault data, reflecting its limited ability to handle these complex scenarios.

To quantitatively assess the overall performance, we employed accuracy as a key metric. Accuracy is calculated by summing the number of true positives and true negatives and then dividing this sum by the total size of the predicted dataset. Regrettably, the resulting accuracy rate stands at 35%, underscoring the classification system's notable limitations in detecting combined fault cases.

This accuracy figure serves as a clear indicator of the classification system's challenges when dealing with multifaceted fault scenarios. It highlights the importance of further research, potential model enhancements, or the exploration of alternative approaches to improve fault detection accuracy in such intricate contexts.

5.1.3.3 Multi-Stage Classification

To illustrate the robustness and efficiency of our proposed multi-stage classification procedure, as depicted in Figure 5.23, aimed at simultaneously enhancing accuracy and stability for diverse diagnostic scenarios, we harnessed the power of a Random Forest classifier combined with the EM-GMM optimization algorithm for early fault detection and classification. Our approach unfolds across three pivotal stages, each contributing to a comprehensive diagnosis:

Stage 1: Healthy vs. Faulty Classification

In the initial stage, our procedure distinguishes between healthy and faulty bearings. This crucial step forms the foundation for subsequent, more nuanced assessments.

Stage 2: Single vs. Combined Fault Detection

The second stage builds upon the initial classification, aiming to pinpoint whether the fault is singular or a combination of multiple issues. This additional layer of analysis adds granularity to the diagnosis.

Stage 3: Fault Localization

The final level of our multi-stage process localizes the specific element within the bearing that exhibits defects. This precise localization is vital for targeted maintenance and repair.

Our journey commences with the selection of features for each stage, depicted in Figure 5.23, through the application of the Gaussian Mixture Model (GMM). To validate the efficacy of the GMM clustering algorithm in the feature selection process, we conducted a rigorous comparative study against commonly used optimization algorithms in the domain of bearing diagnosis.

5.2 Experimental Evaluation

Table 5.16 showcases the accuracy results obtained during the feature selection process across these three stages. We employed a variety of optimization algorithms, including EM-GMM, Simulated Annealing Algorithm (SA), Grasshopper Optimization Algorithm (GOA), Grey Wolf Optimization Algorithm (GWO), Squirrel Algorithm, and Clan-Based Cultural Algorithm (CCA). The evaluation of these results was performed using the Random Forest classifier.

Our assessment utilized the holdout cross-validation method, repeating the process ten times to uncover any potential variations across ten distinct data splits. For each iteration, the data was randomly divided into 90% for training and 10% for testing.

5.3 Results and Discussion

Table 5.16 underscores the remarkable performance of the EM-GMM method in terms of both accuracy and stability. Across all three diagnostic stages, the achieved accuracy consistently hovers around the 100% mark, while the standard deviation (STD) remains nearly negligible. This signifies the superiority of EM-GMM as a feature selection tool, reaffirming its pivotal role in elevating the precision and reliability of our diagnostic process.

Table 5.14: Performance of optimization algorithms in Stage 1

itr	EM-GMM	SA	GOA	GWO	Squirrel	CCA
1	99.58%	98.33%	98.75%	97.91%	99.17%	97.91%
2	100%	98.75%	97.08%	97.50%	98.75%	98.95%
3	99.58%	98.75%	98.33%	98.75%	99.16%	97.91%
4	100%	98.33%	96.66%	98.75%	98.75%	99.16%
5	100%	99.58%	97.08%	97.08%	100%	99.58%
6	100%	98.66%	98.33%	100%	98.75%	99.58%
7	100%	97.91%	98.75%	97.91%	99.58%	98.95%
8	99.17%	99.16%	98.75%	97.91%	98.75%	99.58%
9	100%	98.33%	97.91%	98.75%	99.58%	96.66%
10	99.58%	99.16%	98.75%	98.33%	98.75%	97.91%
Mean	99.79%	98.69%	98.03%	98.28%	99.12%	98.61%
STD	0.29	0.49	0.81	0.82%	0.45%	0.97%

Table 5.15: Performance of optimization algorithms in Stage 2

itr	EM-GMM	SA	GOA	GWO	Squirrel	CCA
1	100%	98.75%	98.75%	98.75%	100%	99.37%
2	100%	100%	99.37%	100%	98.75%	100%
3	100%	100%	99.37%	99.37%	100%	100%
4	100%	98.75%	98.75%	100%	99.37%	99.37%
5	100%	100%	100%	99.37%	100%	100%
6	100%	99.37%	99.37%	100%	99.37%	99.37%
7	100%	100%	99.37%	100%	100%	98.75%
8	100%	98.12%	100%	99.37%	98.12%	98.75%
9	100%	100%	99.37%	98.75%	98.75%	98.12%
10	100%	98.12%	98.75%	100%	100%	99.37%
Mean	100%	99.31%	99.31%	99.56%	99.43%	99.31%
STD	0	0.80	0.46	0.51	0.68	0.62

Table 5.16: Performance of optimization algorithms in Stage 3

itr	EM-GMM	SA	GOA	GWO	Squirrel	CCA
1	100%	100%	100%	100%	99.16%	100%
2	100%	100%	99.16%	100%	99.16%	100%
3	100%	100%	100%	100%	100%	100%
4	100%	99.16%	99.16%	99.16%	99.16%	100%
5	100%	100%	100%	100%	100%	99.16%
6	100%	100%	99.16%	99.16%	100%	100%
7	100%	99.16%	100%	100%	99.16%	100%
8	100%	100%	99.16%	99.16%	97.5%	100%
9	100%	100%	99.16%	100%	99.16%	99.16%
10	100%	99.16%	100%	100%	100%	100%
Mean	100%	99.48%	99.58%	99.74%	99.33%	99.83%
STD	0	0.4	0.44	0.4	0.76	0.35

After training three-stage models, we cascade them according to the diagram in Figure 5.23 to classify the five states of the bearing. This method has been able to achieve a 100% accuracy rate. However, this accuracy is significant only when each class has an equal number of samples, which is not always the case. As a result, this metric is not a reliable measure of the classifier's performance. To overcome this limitation, we employ the Polygon Area Metric (PAM) [112], which takes into account five other metrics in addition to CA, including Sensitivity (SE), Specificity (SP), Area Under Curve (AUC), Jaccard index (JI), kappa (K), and F-measure (FM). The Polygon Area Metric's parameters are defined as follows:

$$CA = \frac{TP + TN}{TP + TN + FP + FN} \tag{5.9}$$

$$SE = \frac{TP}{TP + FN} \tag{5.10}$$

$$SP = \frac{TN}{TN + FP} \tag{5.11}$$

$$JI = \frac{TP}{TP + FP + FN} \tag{5.12}$$

$$F = \frac{2TP}{2TP + FP + FP} \tag{5.13}$$

The Polygon Area Metric (PAM) serves as a comprehensive evaluation framework that encapsulates the performance of a classification system through a geometric representation. This metric draws upon a hexagonal shape, meticulously constructed to incorporate six fundamental evaluation parameters: CA (Classification Accuracy), SE (Sensitivity), SP (Specificity), AUC (Area Under the Curve), JI (Jaccard Index), and

Table 5.17: Polygon Area Metric parameters

Parameter	Polygon Area	CA	Sensitivity	Specificity	AUC	Kappa	JI	F-measure
Unselected features	0.83	0.93	0.99	0.80	0.90	0.90	0.90	0.95
All features	0.96	0.98	0.99	0.96	0.98	0.98	0.98	0.99
Selected features	1	1	1	1	1	1	1	1

FMI (Fowlkes-Mallows Index). These parameters are pivotal in assessing the accuracy, sensitivity, specificity, and overall effectiveness of a classification model.

The hexagon's design is regular, characterized by six equal sides and angles, facilitating a consistent and intuitive representation of the classification system's performance. To ensure an equitable and interpretable measure, we normalize the hexagonal area by dividing it by the value 2.59807. This normalization factor is derived from the area of a regular hexagon composed of six equilateral triangles, each with a side length of 1 unit.

Table 5.17 stands as a testament to the effectiveness of our multi-stage classification procedure. It provides a comprehensive overview of the performance evaluation using the Polygon Area Metric (PAM) across three distinct scenarios: the utilization of all available features, the employment of the selected feature subset, and the use of the unselected feature subset.

By leveraging the PAM, we gain valuable insights into how the classification system's performance varies under these different feature utilization scenarios. This nuanced analysis empowers us to make informed decisions about feature selection and optimization, ultimately contributing to the refinement and enhancement of our diagnostic methodology.

Figures 5.24, 5.25 and 5.26 provide a more comprehensive visualization of the results, clearly illustrating that the classification using the selected features achieves the highest level of performance.

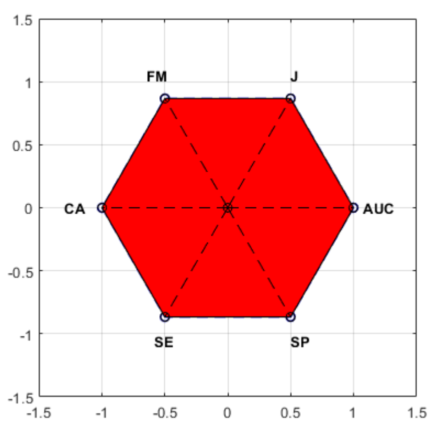

Figure 5.24: Polygon area metric for classification results using the selected features

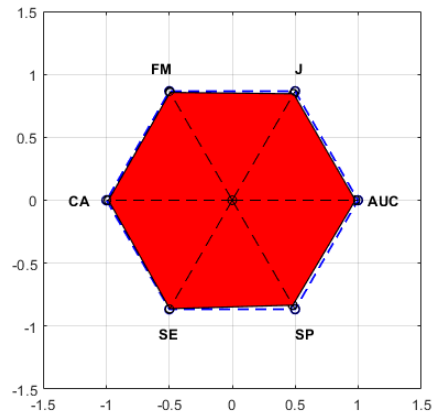

Figure 5.25: Polygon area metric for classification results using all features.

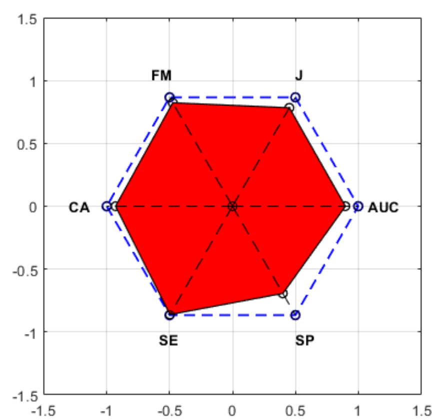

Figure 5.26: Polygon area metric for classification results using the unselected features.

5.3.1 Conclusion

Bearing fault diagnosis is an increasingly critical field, demanding the development of advanced methodologies to attain heightened accuracy in classification, particularly when faced with non-stationary conditions. This article introduces a robust multi-stage classification process tailored for the detection of combined faults in dynamic and time-varying operational settings.

Our methodology commences with the decomposition of vibration signatures sourced from the 'Bearing vibration data collected under time-varying rotational speed conditions' database. We employ the Empirical Wavelet Transform (EWT), a potent technique adept at extracting AM-FM (Amplitude-Modulated Frequency-Modulated) modes. These modes yield a richer data representation, encapsulating variations and patterns across both the amplitude and frequency domains.

From the extracted AM-FM modes, we derive an extensive array of time and frequency domain parameters. These parameters serve as the building blocks of our diagnostic process, affording us an intricate insight into the underlying characteristics of the vibration signatures.

In the subsequent stage, we meticulously select features with the Expectation-Maximization Gaussian Mixture Model (EM-GMM) clustering method. This selection process is pivotal, as it ensures the chosen features possess the discriminative power required to accurately differentiate between diverse fault classes.

Moving forward, we implement a multi-stage classification strategy employing the Random Forest classifier. Each classification model corresponds to a specific diagnostic level within our hierarchical approach. This tailored modeling aligns seamlessly with our method's tiered structure, enhancing the precision of fault identification.

To rigorously evaluate the performance of our proposed procedure, we turn to the Polygon Area Metric (PAM), encompassing six distinct evaluation parameters. This comprehensive evaluation framework scrutinizes the classification results from multiple angles, providing a holistic perspective on the effectiveness of our methodology.

The outcomes of our multi-stage classification process firmly establish the prowess of our approach, particularly in the intricate task of combined defect detection in bearings operating under rigorous and time-varying conditions. Our methodology emerges as a robust solution, bolstering the realm of fault diagnosis, and contributing significantly to the reliability and maintenance efficiency of rotating machinery across diverse industrial domains.

Conclusion

This work marks a significant advancement in the realm of rotating machine diagnostics, driven by the necessity to address the repercussions of defects in these ubiquitous components within modern industries. The core motivation behind this study originates from the critical requirement for a proactive and effective diagnostic strategy to mitigate financial losses and uphold human safety.

The comprehensive investigation commences by addressing the challenges posed by the dynamic operating conditions of rotating machines. Recognizing the limitations of constant-speed scenarios in capturing true operational dynamics, our methodology aims to bridge this gap. Through rigorous testing on a bearing database featuring time-varying conditions and three distinct faults, our proposed approach utilizes the Empirical Wavelet Transform (EWT) to extract AM-FM modes from vibrational signals. Feature extraction from these modes, supported by the Clan-Based Cultural Algorithm (CCA) for selection and Random Forest for training, highlights the resilience of our diagnostic process even in dynamic conditions. Expanding on these foundations, the subsequent phase of our study focuses on feature selection, acknowledging its pivotal role in enhancing diagnostic system quality and reducing misleading factors. Our thesis introduces a robust technique based on standard deviation and Random Forest for sequential feature selection, demonstrating effectiveness across diverse bearing databases and signal decomposition techniques. Furthering in the feature selection process we a clustering algorithm EM-GMM for feature selection. This method excels in selecting features that accurately identify the distinct health states within bearing data, encompassing five categories (healthy, inner race defect, outer race defect, ball defects, and combined faults). The data collected under time-varying speed conditions presents a challenging backdrop for classification. The classification process is intricately divided into three stages: fault detection, differentiation between simple or combined faults, and fault localization. A comparison with binary classification highlights the superior performance of the three-stage classification approach, particularly for combined faults. Beyond presenting innovative solutions, the practical implications of our research are substantial. Our diagnostic approach and feature selection methods offer tangible benefits in real-world applications by enhancing the reliability and safety of rotating machines. Implementation of these methods can facilitate proactive fault identification, thereby minimizing economic losses and ensuring human well-being. Recognizing the importance of transparency in our research, we acknowledge the limitations of our methods under specific conditions. This transparency delineates the scope of our research and suggests avenues for future development. The novelty of our work extends beyond diagnostic approaches to encompass innovative feature selection methods. The unique advantages offered by our diagnostic approaches and feature selection methods over existing techniques position our work as a notable advancement in the field. Looking ahead, this study lays a robust foundation for future research in rotating machine diagnostics. The potential extension of our research to other machine types or its integration with complementary diagnostic methods presents promising avenues for enhancing accuracy. These recommendations serve as inspiration for future advancements in the field. In essence, this work not only identifies challenges but also provides innovative solutions with practical implications while acknowledging limitations, emphasizing novelty, and showcasing promising findings from incorporating a clustering algorithm for feature selection. The research significantly propels the field forward, promising tangible benefits for industries, safety, and technological evolution. The knowledge generated sets the stage for a proactive and reliable approach to rotating machine diagnostics, shaping the trajectory of future advancements.

References

[1] Alan Jović, Karla Brkić, and Nikola Bogunović. A review of feature selection methods with applications. In *2015 38th international convention on information and communication technology, electronics and microelectronics (MIPRO)*, pages 1200–1205. Ieee, 2015.

[2] KC Deekshit Kompella, Venu Gopala Rao Mannam, and Srinivasa Rao Rayapudi. Dwt based bearing fault detection in induction motor using noise cancellation. *Journal of Electrical Systems and Information Technology*, 3(3):411–427, 2016.

[3] Jerome Gilles. Empirical wavelet transform. *IEEE transactions on signal processing*, 61(16):3999–4010, 2013.

[4] Huan Huang and Natalie Baddour. Bearing vibration data collected under time-varying rotational speed conditions. *Data in brief*, 21:1745–1749, 2018.

[5] Rudrapati Victor Daniel, Savale Amit Siddhappa, Savale Bhushan Gajanan, S Vipin Philip, and P Sam Paul. Effect of bearings on vibration in rotating machinery. In *IOP Conference Series: Materials Science and Engineering*, volume 225, page 012264. IOP Publishing, 2017.

[6] Duy Tang Hoang and Hee Jun Kang. A motor current signal-based bearing fault diagnosis using deep learning and information fusion. *IEEE Transactions on Instrumentation and Measurement*, 69(6):3325–3333, 2019.

[7] BM Vamsi Krishna and Manish Vishwakarma. A review on vibration-based fault diagnosis in rolling element bearings. *International Journal of Applied Engineering Research*, 13(8):6188–6192, 2018.

[8] Mohamad Hazwan Mohd Ghazali and Wan Rahiman. Vibration analysis for machine monitoring and diagnosis: a systematic review. *Shock and Vibration*, 2021:1–25, 2021.

[9] Adam Glowacz. Ventilation diagnosis of angle grinder using thermal imaging. *Sensors*, 21(8):2853, 2021.

[10] Abdenour Soualhi, Kamal Medjaher, and Noureddine Zerhouni. Bearing health monitoring based on hilbert–huang transform, support vector machine, and regression. *IEEE Transactions on instrumentation and measurement*, 64(1):52–62, 2014.

[11] M. A. Mukutadze and Dinara Usmanovna Khasyanova. Radial friction bearing with a fusible coating in the turbulent friction mode. *Journal of Machinery Manufacture and Reliability*, 48:421 – 430, 2019.

[12] Fabio Immovilli, Marco Cocconcelli, Alberto Bellini, and Riccardo Rubini. Detection of generalized-roughness bearing fault by spectral-kurtosis energy of vibration or current signals. *IEEE Transactions on Industrial Electronics*, 56(11):4710–4717, 2009.

[13] Lars Erik Stacke and Dag Fritzson. Dynamic behaviour of rolling bearings: simulations and experiments. *Proceedings of the Institution of Mechanical Engineers, Part J: Journal of Engineering Tribology*, 215(6):499–508, 2001.

[14] Prashant P Kharche and Sharad V Kshirsagar. Review of fault detection in rolling element bearing. *Int. J. Innov. Res. Adv. Eng*, 1(5):169–174, 2014.

[15] Ciprian Harlisca and Loránd Szabó. Bearing faults condition monitoring-a literature survey. *Journal of Computer Science and Control Systems*, 5(2):19, 2012.

[16] Zepeng Liu and Long Zhang. A review of failure modes, condition monitoring and fault diagnosis methods for large-scale wind turbine bearings. *Measurement*, 149:107002, 2020.

[17] Sukhjeet Singh, Amit Kumar, and Navin Kumar. Motor current signature analysis for bearing fault detection in mechanical systems. *Procedia Materials Science*, 6:171–177, 2014.

[18] Jason R Stack, Thomas G Habetler, and Ronald G Harley. Bearing fault detection via autoregressive stator current modeling. *IEEE Transactions on Industry Applications*, 40(3):740–747, 2004.

[19] JaeYoung Kim and Jong-Myon Kim. Bearing fault diagnosis using grad-cam and acoustic emission signals. *Applied Sciences*, 10(6):2050, 2020.

[20] A Morhain and David Mba. Bearing defect diagnosis and acoustic emission. *Proceedings of the Institution of Mechanical Engineers, Part J: Journal of Engineering Tribology*, 217(4):257–272, 2003.

[21] Anurag Choudhary, Tauheed Mian, and Shahab Fatima. Convolutional neural network based bearing fault diagnosis of rotating machine using thermal images. *Measurement*, 176:109196, 2021.

[22] BJ Roylance, JA Williams, and R Dwyer-Joyce. Wear debris and associated wear phenomena—fundamental research and practice. *Proceedings of the Institution of Mechanical Engineers, Part J: Journal of Engineering Tribology*, 214(1):79–105, 2000.

[23] Chandrabhanu Malla and Isham Panigrahi. Review of condition monitoring of rolling element bearing using vibration analysis and other techniques. *Journal of Vibration Engineering & Technologies*, 7:407–414, 2019.

[24] Jafar Zarei, Mohammad Amin Tajeddini, and Hamid Reza Karimi. Vibration analysis for bearing fault detection and classification using an intelligent filter. *Mechatronics*, 24(2):151–157, 2014.

[25] J Mathew and RJ Alfredson. The condition monitoring of rolling element bearings using vibration analysis. 1984.

[26] D Goyal and BS Pabla. The vibration monitoring methods and signal processing techniques for structural health monitoring: a review. *Archives of Computational Methods in Engineering*, 23:585–594, 2016.

[27] Bin Zhang, Georgios Georgoulas, Marcos Orchard, Abhinav Saxena, Douglas Brown, George Vachtsevanos, and Steven Liang. Rolling element bearing feature extraction and anomaly detection based on vibration monitoring. In *2008 16th Mediterranean Conference on Control and Automation*, pages 1792–1797. IEEE, 2008.

[28] Wu Deng, Shengjie Zhang, Huimin Zhao, and Xinhua Yang. A novel fault diagnosis method based on integrating empirical wavelet transform and fuzzy entropy for motor bearing. *Ieee Access*, 6:35042–35056, 2018.

[29] Issam Attoui, Nadir Boutasseta, Nadir Fergani, Brahim Oudjani, and Adel Deliou. Vibration-based bearing fault diagnosis by an integrated dwt-fft approach and an adaptive neuro-fuzzy inference system. In *2015 3rd International Conference on Control, Engineering & Information Technology (CEIT)*, pages 1–6. IEEE, 2015.

[30] Mohamed Zair, Chemseddine Rahmoune, and Djamel Benazzouz. Multi-fault diagnosis of rolling bearing using fuzzy entropy of empirical mode decomposition, principal component analysis, and som neural network. *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, 233(9):3317–3328, 2019.

[31] Muhammad Altaf, Tallha Akram, Muhammad Attique Khan, Muhammad Iqbal, M Munawwar Iqbal Ch, and Ching-Hsien Hsu. A new statistical features based approach for bearing fault diagnosis using vibration signals. *Sensors*, 22(5):2012, 2022.

[32] Hongyu Yang, Joseph Mathew, and Lin Ma. Basis pursuit-based intelligent diagnosis of bearing faults. *Journal of Quality in Maintenance Engineering*, 13(2):152–162, 2007.

[33] BA Paya, II Esat, and MNM Badi. Artificial neural network based fault diagnostics of rotating machinery using wavelet transforms as a preprocessor. *Mechanical systems and signal processing*, 11(5):751–765, 1997.

[34] HR Martin and Farhang Honarvar. Application of statistical moments to bearing failure detection. *Applied acoustics*, 44(1):67–77, 1995.

[35] RBW Heng and Mohd Jailani Mohd Nor. Statistical analysis of sound and vibration signals for monitoring rolling element bearing condition. *Applied Acoustics*, 53(1-3):211–226, 1998.

[36] Tuncay Karacay and Nizami Akturk. Experimental diagnostics of ball bearings using statistical and spectral methods. *Tribology International*, 42(6):836–843, 2009.

[37] Khaoula Tidriri, Nizar Chatti, Sylvain Verron, and Teodor Tiplica. Bridging data-driven and model-based approaches for process fault diagnosis and health monitoring: A review of researches and future challenges. *Annual reviews in control*, 42:63–81, 2016.

[38] Thanasak Wanglomklang, Phathan Chommaungpuck, Kontorn Chamniprasart, and Jiraphon Srisertpol. Using fault detection and classification techniques for machine breakdown reduction of the hga process caused by the slider loss defect. *Manufacturing Review*, 9:21, 2022.

[39] Lu Qian, Qing Pan, Yaqiong Lv, and Xingwei Zhao. Fault detection of bearing by resnet classifier with model-based data augmentation. *Machines*, 10(7):521, 2022.

[40] Rolf Isermann. Model-based fault-detection and diagnosis–status and applications. *Annual Reviews in control*, 29(1):71–85, 2005.

[41] Venkat Venkatasubramanian, Raghunathan Rengaswamy, Kewen Yin, and Surya N Kavuri. A review of process fault detection and diagnosis: Part i: Quantitative model-based methods. *Computers & chemical engineering*, 27(3):293–311, 2003.

[42] Kenneth A Loparo, Maurice L Adams, Wei Lin, M Farouk Abdel-Magied, and Nadar Afshari. Fault detection and diagnosis of rotating machinery. *IEEE Transactions on Industrial Electronics*, 47(5):1005–1014, 2000.

[43] Masoud Jalayer, Carlotta Orsenigo, and Carlo Vercellis. Fault detection and diagnosis for rotating machinery: A model based on convolutional lstm, fast fourier and continuous wavelet transforms. *Computers in Industry*, 125:103378, 2021.

[44] Davor Kolar, Dragutin Lisjak, Michał Pajak, and Danijel Pavković. Fault diagnosis of rotary machines using deep convolutional neural network with wide three axis vibration signal input. *Sensors*, 20(14):4017, 2020.

[45] Shiza Mushtaq, MM Manjurul Islam, and Muhammad Sohaib. Deep learning aided data-driven fault diagnosis of rotatory machine: A comprehensive review. *Energies*, 14(16):5150, 2021.

[46] Francesca Calabrese, Alberto Regattieri, Marco Bortolini, and Francesco Gabriele Galizia. Data-driven fault detection and diagnosis: Challenges and opportunities in real-world scenarios. *Applied Sciences*, 12(18):9212, 2022.

[47] Siyu Zhang, SU Lei, GU Jiefei, LI Ke, ZHOU Lang, and Michael Pecht. Rotating machinery fault detection and diagnosis based on deep domain adaptation: A survey. *Chinese Journal of Aeronautics*, 36(1):45–74, 2023.

[48] J-H Zhou, Louis Wee, and Z-W Zhong. A knowledge base system for rotary equipment fault detection and diagnosis. In *2010 11th International Conference on Control Automation Robotics & Vision*, pages 1335–1340. IEEE, 2010.

[49] Dongyang Dou, Jianguo Yang, Jiongtian Liu, and Yingkai Zhao. A rule-based intelligent method for fault diagnosis of rotating machinery. *Knowledge-Based Systems*, 36:1–8, 2012.

[50] Dongyang Dou, Jian Jiang, Yuling Wang, and Yong Zhang. A rule-based classifier ensemble for fault diagnosis of rotating machinery. *Journal of Mechanical Science and Technology*, 32:2509–2515, 2018.

[51] Changqing Cheng, Akkarapol Sa-Ngasoongsong, Omer Beyca, Trung Le, Hui Yang, Zhenyu Kong, and Satish TS Bukkapatnam. Time series forecasting for nonlinear and non-stationary processes: a review and comparative study. *Iie Transactions*, 47(10):1053–1071, 2015.

[52] Dany Abboud, Sophie Baudin, Jérôme Antoni, Didier Rémond, Mario Eltabach, and Olivier Sauvage. The spectral analysis of cyclo-non-stationary signals. *Mechanical Systems and Signal Processing*, 75:280–300, 2016.

[53] Zhipeng Feng, Dong Zhang, and Ming J Zuo. Adaptive mode decomposition methods and their applications in signal analysis for machinery fault diagnosis: a review with examples. *IEEE access*, 5:24301–24331, 2017.

[54] Marco Civera and Cecilia Surace. A comparative analysis of signal decomposition techniques for structural health monitoring on an experimental benchmark. *Sensors*, 21(5):1825, 2021.

[55] Pierre Duhamel and Martin Vetterli. Fast fourier transforms: a tutorial review and a state of the art. *Signal processing*, 19(4):259–299, 1990.

[56] Franz Franchetti and Markus Püschel. Fast fourier transform. *Encyclopedia of Parallel Computing. Springer*, page 51, 2011.

[57] Ervin Sejdić, Igor Djurović, and Jin Jiang. Time–frequency feature representation using energy concentration: An overview of recent advances. *Digital signal processing*, 19(1):153–183, 2009.

[58] Robi Polikar. The story of wavelets. *Physics and modern topics in mechanical and electrical engineering*, pages 192–197, 1999.

[59] Nawaf HMM Shrifan, Muhammad Firdaus Akbar, and Nor Ashidi Mat Isa. Maximal overlap discrete wavelet-packet transform aided microwave nondestructive testing. *NDT & E International*, 119:102414, 2021.

[60] Zhu Jie, Liu Xiaoli, and Li Juntao. Fresh food distribution center storage allocation strategy analysis based on optimized entry-item-quantity-abc. *Int. J. Data Sci. Technol*, pages 36–40, 2016.

[61] Fei Dong, Xiao Yu, Enjie Ding, Shoupeng Wu, Chunyang Fan, and Yanqiu Huang. Rolling bearing fault diagnosis using modified neighborhood preserving embedding and maximal overlap discrete wavelet packet transform with sensitive features selection. *Shock and Vibration*, 2018:1–29, 2018.

[62] D-M Yang. The detection of motor bearing fault with maximal overlap discrete wavelet packet transform and teager energy adaptive spectral kurtosis. *Sensors*, 21(20):6895, 2021.

[63] Wei Liu and Wei Chen. Recent advancements in empirical wavelet transform and its applications. *IEEE Access*, 7:103770–103780, 2019.

[64] AO Boudraa, JC Cexus, and Z Saidi. Emd-based signal noise reduction. *International Journal of Signal Processing*, 1(1):33–37, 2004.

[65] M Grasso, STEVEN Chatterton, P Pennacchi, and BM Colosimo. A data-driven method to enhance vibration signal decomposition for rolling bearing fault analysis. *Mechanical Systems and Signal Processing*, 81:126–147, 2016.

[66] Thomas Oberlin, Sylvain Meignen, and Valérie Perrier. An alternative formulation for the empirical mode decomposition. *IEEE Transactions on Signal Processing*, 60(5):2236–2246, 2012.

[67] Jinde Zheng, Haiyang Pan, Shubao Yang, and Junsheng Cheng. Adaptive parameterless empirical wavelet transform based time-frequency analysis method and its application to rotor rubbing fault diagnosis. *Signal Processing*, 130:305–314, 2017.

[68] Nyararai Mlambo, Wilson K Cheruiyot, and Michael W Kimwele. A survey and comparative study of filter and wrapper feature selection techniques. *International Journal of Engineering and Science (IJES)*, 5(8):57–67, 2016.

[69] Isabelle Guyon and André Elisseeff. An introduction to variable and feature selection. *Journal of machine learning research*, 3(Mar):1157–1182, 2003.

[70] S Shilan Hameed, Olutomilayo Olayemi Petinrin, A Osman Hashi, and Faisal Saeed. Filter-wrapper combination and embedded feature selection for gene expression data. *Int. J. Advance Soft Compu. Appl*, 10(1):90–105, 2018.

[71] Huan Liu and Hiroshi Motoda. *Feature extraction, construction and selection: A data mining perspective*, volume 453. Springer Science & Business Media, 1998.

[72] Girish Chandrashekar and Ferat Sahin. A survey on feature selection methods. *Computers & Electrical Engineering*, 40(1):16–28, 2014.

[73] Zhongzhe Xiao, Emmanuel Dellandrea, Weibei Dou, and Liming Chen. *ESFS: A new embedded feature selection method based on SFS*. PhD thesis, Ecole Centrale Lyon; Université de Lyon; LIRIS UMR 5205 CNRS/INSA de Lyon ..., 2008.

[74] Asha Gowda Karegowda, MA Jayaram, and AS Manjunath. Feature subset selection problem using wrapper approach in supervised learning. *International journal of Computer applications*, 1(7):13–17, 2010.

[75] Xin-She Yang. Nature-inspired optimization algorithms: Challenges and open problems. *Journal of Computational Science*, 46:101104, 2020.

[76] Anderson Rogério Faia Pinto, Antonio Fernando Crepaldi, and Marcelo Seido Nagano. A genetic algorithm applied to pick sequencing for billing. *Journal of Intelligent Manufacturing*, 29:405–422, 2018.

[77] Alberto Colorni, Marco Dorigo, and Vittorio Maniezzo. A genetic algorithm to solve the timetable problem. *Politecnico di Milano, Milan, Italy TR*, pages 90–060, 1992.

[78] James Kennedy and Russell Eberhart. Particle swarm optimization. In *Proceedings of ICNN'95-international conference on neural networks*, volume 4, pages 1942–1948. IEEE, 1995.

[79] Juan R González, Alejandro Sancho-Royo, David A Pelta, and Carlos Cruz. Nature-inspired cooperative strategies for optimization. In *Encyclopedia of Networked and Virtual Organizations*, pages 982–989. IGI Global, 2008.

[80] Xin-She Yang. Firefly algorithms for multimodal optimization. In *International symposium on stochastic algorithms*, pages 169–178. Springer, 2009.

[81] Xin-She Yang and Suash Deb. Cuckoo search via lévy flights. In *2009 World congress on nature & biologically inspired computing (NaBIC)*, pages 210–214. Ieee, 2009.

[82] Marco Dorigo. Optimization, learning and natural algorithms. *Ph. D. Thesis, Politecnico di Milano*, 1992.

[83] Seyedali Mirjalili, Seyed Mohammad Mirjalili, and Andrew Lewis. Grey wolf optimizer. *Advances in engineering software*, 69:46–61, 2014.

[84] Mohit Jain, Vijander Singh, and Asha Rani. A novel nature-inspired algorithm for optimization: Squirrel search algorithm. *Swarm and evolutionary computation*, 44:148–175, 2019.

[85] Yassine Meraihi, Asma Benmessaoud Gabis, Seyedali Mirjalili, and Amar Ramdane-Cherif. Grasshopper optimization algorithm: theory, variants, and applications. *Ieee Access*, 9:50001–50024, 2021.

[86] Balram Suman and Prabhat Kumar. A survey of simulated annealing as a tool for single and multiobjective optimization. *Journal of the operational research society*, 57:1143–1160, 2006.

[87] Nur Farahlina Johari, Azlan Mohd Zain, Mustaffa H Noorfa, and Amirmudin Udin. Firefly algorithm for optimization problem. *Applied Mechanics and Materials*, 421:512–517, 2013.

[88] X. S. Yang. Firefly algorithms for multimodal optimization. In *Stochastic algorithms: foundations and applications*, pages 169–178. Springer, 2008.

[89] X. S. Yang. Binary differential evolution: A new optimization algorithm. *IEEE Transactions on Evolutionary Computation*, 23(4):1068–1079, 2019.

[90] Oluwabunmi Oloruntoba, Georgina Cosma, and Antonio Liotta. Clan-based cultural algorithm for feature selection. In *2019 international conference on data mining workshops (ICDMW)*, pages 465–472. IEEE, 2019.

[91] Christian Janiesch, Patrick Zschech, and Kai Heinrich. Machine learning and deep learning. *Electronic Markets*, 31(3):685–695, 2021.

[92] Pratap Chandra Sen, Mahimarnab Hajra, and Mitadru Ghosh. Supervised classification algorithms in machine learning: A survey and review. In *Emerging Technology in Modelling and Graphics: Proceedings of IEM Graph 2018*, pages 99–111. Springer, 2020.

[93] Sotiris B Kotsiantis. Decision trees: a recent overview. *Artificial Intelligence Review*, 39:261–283, 2013.

[94] Barry De Ville. Decision trees. *Wiley Interdisciplinary Reviews: Computational Statistics*, 5(6):448–455, 2013.

[95] Mahesh Pal. Random forest classifier for remote sensing classification. *International journal of remote sensing*, 26(1):217–222, 2005.

[96] Anantha M Prasad, Louis R Iverson, and Andy Liaw. Newer classification and regression tree techniques: bagging and random forests for ecological prediction. *Ecosystems*, 9:181–199, 2006.

[97] Andrew J Sage, Ulrike Genschel, and Dan Nettleton. Random forest variable importance in the presence of missing data. *Random forest robustness, variable importance, and tree aggregation*, page 37, 2018.

[98] Peter Dayan, Maneesh Sahani, and Grégoire Deback. Unsupervised learning. *The MIT encyclopedia of the cognitive sciences*, pages 857–859, 1999.

[99] Amit Saxena, Mukesh Prasad, Akshansh Gupta, Neha Bharill, Om Prakash Patel, Aruna Tiwari, Meng Joo Er, Weiping Ding, and Chin-Teng Lin. A review of clustering techniques and developments. *Neurocomputing*, 267:664–681, 2017.

[100] Chris Fraley and Adrian E Raftery. How many clusters? which clustering method? answers via model-based cluster analysis. *The computer journal*, 41(8):578–588, 1998.

[101] Fionn Murtagh. A survey of recent advances in hierarchical clustering algorithms. *The computer journal*, 26(4):354–359, 1983.

[102] Glory H Shah, CK Bhensdadia, and Amit P Ganatra. An empirical evaluation of density-based clustering techniques. *International Journal of Soft Computing and Engineering (IJSCE) ISSN*, 22312307:216–223, 2012.

[103] Arthur P Dempster, Nan M Laird, and Donald B Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the royal statistical society: series B (methodological)*, 39(1):1–22, 1977.

[104] Christopher M Bishop and Nasser M Nasrabadi. *Pattern recognition and machine learning*, volume 4. Springer, 2006.

[105] Shu Kay Ng, Thriyambakam Krishnan, and Geoffrey J McLachlan. The em algorithm. *Handbook of computational statistics: concepts and methods*, pages 139–172, 2012.

[106] Maya R Gupta, Yihua Chen, et al. Theory and use of the em algorithm. *Foundations and Trends® in Signal Processing*, 4(3):223–296, 2011.

[107] Jasmin Praful Bharadiya. A tutorial on principal component analysis for dimensionality reduction in machine learning. *International Journal of Innovative Science and Research Technology*, 8(5):2028–2032, 2023.

[108] Rizgar Zebari, Adnan Abdulazeez, Diyar Zeebaree, Dilovan Zebari, and Jwan Saeed. A comprehensive review of dimensionality reduction techniques for feature selection and feature extraction. *Journal of Applied Science and Technology Trends*, 1(2):56–70, 2020.

[109] Peter Dayan and Yael Niv. Reinforcement learning: the good, the bad and the ugly. *Current opinion in neurobiology*, 18(2):185–196, 2008.

[110] Nathaniel D Daw, Yael Niv, and Peter Dayan. Uncertainty-based competition between prefrontal and dorsolateral striatal systems for behavioral control. *Nature neuroscience*, 8(12):1704–1711, 2005.

[111] Huan Huang and Natalie Baddour. Bearing vibration data under time-varying rotational speed conditions, 2019.

[112] Onder Aydemir. A new performance evaluation metric for classifiers: polygon area metric. *Journal of Classification*, 38:16–26, 2021.