Democratic and Popular Republic of Algeria
Ministry of Higher Education and Scientific Research
Echahid Cheikh Larbi Tebessi University- Tebessa
Faculty of Sciences and Technology
Electrical Engineering Department

Fault Detection and Diagnosis of an Industrial Process

By

SOUAIDIA Chouaib

A Dissertation Submitted to the Electrical Engineering Department

Affiliated to Electrical Engineering Laboratory (LABGET)

In Partial Fulfillment of the Requirement for the Degree of

Doctorate of Philosophy LMD

In

Automatic

Defended Date: July 15th, 2023

Board of Examiners:

Jury members	Grade	Designation	University
Dr. Tarek BENTAHAR	MCA	Chairman	Echahid Cheikh Larbi Tebessi University- Tebessa
Dr. Djamel OUNNAS	MCA	Member	Echahid Cheikh Larbi Tebessi University- Tebessa
Pr. Abderazzak LACHOURI	Professor	Member	University of 20th August 1955 Skikda
Pr. Khaled KHELIL	Professor	Member	Mohamed Sherif Messaadia University – SoukAhras
Pr. Salah CHENIKHER	Professor	Supervisor	Echahid Cheikh Larbi Tebessi University- Tebessa
Dr. Tawfik THELAIDJIA	MCA	Co-Supervisor	Mohamed Cherif Messaadia University – SoukAhras

Academic Year: 2022 / 2023

ملخص:

يهدف العمل المقترح في هذه الأطروحة إلى تطوير مساهمة جديدة لتشخيص الأخطاء في عملية صناعية ، وتحديداً في المحامل ، بناءً على طرق معالجة الإشارات والتعرف على الأنماط. يركز العمل المقدم في هذه الرسالة على كشف وتشخيص عيوب الحمل باستخدام تحليل الاهتزاز والتعلم الآلي.

في المرحلة الأولى ، يتم الحصول على البيانات من نظام أو منصة اختبار ليتم دراستها مع مراعاة أنواع العيوب المختلفة. في هذا العمل ، تم استخدام مجموعتي بيانات مختلفتين تحتويان على إشارات اهتزاز لتحمل تشخيص الأعطال. بعد ذلك ، سيتم تطبيق تقنيات معالجة الإشارات مثل طرق فصل المصادر العمياء لتحليل الإشارة. من بين العديد من طرق فصل المصادر العمياء ، يتم استخدام تحليل المتجه المستقل (IVA) لتحليل إشارة الاهتزاز كطريقة لفصل مصادر الاهتزاز عن الإشارات المرصودة ، ولتقليل التداخل والضوضاء. بعد ذلك ، قم بتطوير تقنية فعالة لاستخراج الميزات باستخدام ميزات المجال الزمني لتقليل أبعاد البيانات ، وإزالة المعلومات غير ذات الصلة أو الزائدة عن الحاجة ، وتعزيز جدوى البيانات وإمكانية تفسيرها. ثم سيتم تطبيق طرق اختيار الميزة لتقليل الأبعاد وتعقيد الميزة المستخرجة مما يؤدي إلى تسريع خوارزمية التعلم وتحسين الدقة التنبؤية لخوارزمية التصنيف ، وتجنب الإفراط في التركيب والضوضاء.

المرحلة الثانية هي تصنيف الحالة بناءً على خوارزميات التعلم الآلي. ستقوم هذه المرحلة باستكشاف ومقارنة خوارزميات التعلم الخاضعة للإشراف المختلفة مثل آلات المتجهات الداعمة والغابات العشوائية والشبكات العصبية الاصطناعية وآلات التعلم المتطرفة لتحديد النهج الأنسب لتصنيف الأخطاء. أخيرًا ، إجراء اختبارات صارمة والتحقق من صحة نماذج التعلم الآلي المطورة على الآلات الصناعية في العالم الحقيقي لضمان فعاليتها وعمليتها في بيئة الإنتاج.

تم التحقق من فعالية الطرق المقترحة في هذه الأطروحة من خلال إشارات محاكاة وبيانات تجريبية.

الكلمات الرئيسية: *تحمل تشخيص الأعطال. تحليل الاهتزاز تحليل النواقل المستقلة خوارزمية الخفافيش الثنائية تحسين سرب الجسيمات الثنائية ؛ ثنائي رمادي الذئب الأمثل ؛ دعم آلات النواقل ؛ غابات عشوائية الشبكات العصبية الاصطناعية؛ آلات التعلم المتطرفة.*

Abstract:

The suggested work in this thesis aims to develop a new contribution for fault diagnosis in an industrial process, specifically in bearings, based on signal processing and pattern recognition methods. The work presented in this thesis focuses on the detection and diagnosis of bearing defects by using vibration analysis and machine learning.

In the first stage, data is acquired from a system or a test rig to be studied taking into account the various types of faults. In this work, two different datasets containing vibration signals are used for bearing fault diagnosis. Then, signal processing techniques such as Blind Source Separation methods will be applied for signal analysis. Among many methods of Blind Source Separation, the Independent vector analysis (IVA) is used for vibration signal analysis as a way to separate the sources of vibration from the observed signals, and to reduce the interference and noise. Next, develop an effective feature extraction technique using time-domain features to reduce the dimensionality of the data, remove irrelevant or redundant information, and enhance the meaningfulness and interpretability of the data. Then Feature Selection methods will be applied to reduce the dimensionality the complexity of the extracted feature, which will speed up a learning algorithm and improve the predictive accuracy of a classification algorithm, and avoid overfitting and noise.

The second stage is condition classification based on machine learning algorithms. This stage will explore and compare various supervised learning algorithms like Support Vector Machines, Random Forests, Artificial Neural Networks and Extreme Learning Machines to determine the most suitable approach for bearing fault classification. Finally, conducting rigorous testing and validation of the developed machine learning models on real-world industrial machinery to ensure their effectiveness and practicality in a production environment.

The effectiveness of the proposed methods in this thesis has been validated by simulated signals and experimental data.

Keywords: *Bearing Fault Diagnosis; Vibration Analysis; Independent Vector Analysis; Binary Bat Algorithm; Binary Particle Swarm Optimisation; Binary Grey Wolf Optimisation; Support Vector Machines; Random Forests; Artificial Neural Networks; Extreme Learning Machines.*

Résumé:

Le travail proposé dans cette thèse vise à développer une nouvelle contribution pour le diagnostic de défauts dans un processus industriel, spécifiquement dans les roulements, basée sur des méthodes de traitement du signal et de reconnaissance de formes. Le travail présenté dans cette thèse porte sur la détection et le diagnostic des défauts de roulements en utilisant l'analyse vibratoire et l'apprentissage automatique.

En premier étape, des données sont acquises à partir d'un système ou d'un banc d'essais à étudier en tenant compte des différents types de défauts. Dans ce travail, deux ensembles de données différents contenant des signaux de vibration sont utilisés pour le diagnostic des défauts de roulement. Ensuite, des techniques de traitement du signal telles que les méthodes de séparation aveugle de la source seront appliquées pour l'analyse du signal. Parmi de nombreuses méthodes de séparation aveugle des sources, l'analyse vectorielle indépendante (IVA) est utilisée pour l'analyse des signaux de vibration comme moyen de séparer les sources de vibration des signaux observés et de réduire les interférences et le bruit. Ensuite, développez une technique d'extraction de caractéristiques efficace à l'aide de caractéristiques du domaine temporel pour réduire la dimensionnalité des données, supprimer les informations non pertinentes ou redondantes et améliorer la pertinence et l'interprétabilité des données. Ensuite, des méthodes de sélection de caractéristiques seront appliquées pour réduire la dimensionnalité et la complexité de la caractéristique extraite, ce qui accélérera un algorithme d'apprentissage et améliorera la précision prédictive d'un algorithme de classification, et évitera le surajustement et le bruit.

La deuxième étape est la classification des conditions basée sur des algorithmes d'apprentissage automatique. Cette étape explorera et comparera divers algorithmes d'apprentissage supervisé tels que les machines à vecteurs de support, les forêts aléatoires, les réseaux de neurones artificiels et les machines d'apprentissage extrêmes afin de déterminer l'approche la plus appropriée pour la classification des défauts de roulement. Enfin, effectuer des tests et une validation rigoureuse des modèles d'apprentissage automatique développés sur des machines industrielles réelles pour garantir leur efficacité et leur praticité dans un environnement de production.

L'efficacité des méthodes proposées dans cette thèse a été validée par des signaux simulés et des données expérimentales.

Mots-clés : *Diagnostic De Défaut De Roulement ; Analyse Vibratoire ; Analyse Vectorielle Indépendante ; Algorithme De Chauve-souris Binaire ; Optimisation D'essaim De Particules Binaires ; Optimisation Binaire Du Loup Gris ; Soutenir Les Machines Vectorielles ; Forêts Aléatoires ; Réseaux De Neurones Artificiels; Machines D'apprentissage Extrêmes.*

ACKNOWLEDGEMENTS

I would like to express my sincere gratitude to the following individuals who have contributed to the completion of this thesis:

First and foremost, I am deeply grateful to my thesis supervisor, Pr. Salah CHENIKHER, for his unwavering guidance, continuous support, and valuable feedback throughout the research process. His expertise and encouragement have been instrumental in shaping this work.

I also express my special gratitude to my co-supervisor, Dr. Tawfik THELAIDJIA, for his guidance throughout my PhD study. His immense knowledge, professional suggestions and comments have been essential for this thesis. His great passion for research also motivated me to complete this thesis.

I extend my thanks to the members of my academic committee, Pr. Abderazzak LACHOURI, Pr. Khaled KHELIL, Dr. Djamel OUNNAS and Dr. Tarek BENTAHAR, for their time and insightful comments, helped improve the quality of this thesis and their hard questions motivated me to widen my research.

I would like to thank my friends and fellow students who engaged in meaningful discussions, provided feedback, and offered words of encouragement throughout this academic journey.

Lastly, I extend my gratitude to all the experts and researchers in the Electrical Engineering Department who generously shared their knowledge, expertise and support, enriching this thesis. To all those who have contributed in any way, big or small, to this endeavour, I offer my heartfelt thanks.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

GLOSSARY

AE	Acoustic Emission
AI	Artificial Intelligence
ANN	Artificial Neural Network
BBA	Binary Bat Algorithm
BF	Ball Fault
BGWO	Binary Grey Wolf Optimization
BPFI	Ball Pass Frequency Inner race
BPFO	Ball Pass Frequency Outer race
BPSO	Binary Particle Swarm Optimization
BSF	Ball Spin Frequency
CNN	Convolutional Neural Network
CSDF	Class Separation And Domain Fusion
CWRU	Case Western Reserve University
CWT	Continuous Wavelet Transform
DBN	Deep Belief Networks
DIRG	Dynamic and Identification Research Group
DL	Deep Learning
DNN	Deep Neural Network
DWT	Discrete Wavelet Transform
EEMD	Ensemble Empirical Mode Decomposition
ELM	Extreme Learning Machine
EMD	Empirical Mode Decomposition
FNN	Feed-Forward Neural Networks
FFT	Fast Fourier Transform
FTF	Fundamental Train Frequency
GA	Genetic Algorithms
GAN	Generative Adversarial Network
GMM	Gaussian Mixture Model

GMM	Gaussian Mixture Model
HHT	Hilbert-Huang Transform
HTY	Healthy
Hz	Hirtz
ICA	Independent Component Analysis
IF	Inner Race Fault
IRT	Infrared thermography
IVA	Independent Vector Analysis
KNN	K-Nearest Neighbor
Kurt	Kurtosis
LDA	Linear Discriminant Analysis
LSTM	Long Short-Term Memory
MD	Mahalanobis Distance
ML	Machine Learning
MLP	Multi-Layer Perceptron
MSST	Multi-synchro-squeezing Transform
OF	Outer Rave Fault
PCA	Principal Component Analysis
PSD	Power Spectral Density
RBFN	Radial Basis Function Networks
ReLU	Rectified Linear Unit
RF	Random Forest
RMS	Root Mean Square
RNN	Recurrent Neural Network
ROC	Receiver Operating Characteristic
RPM	Round Per Minute
SAE	Stack Autoencoder
SD	Standard Deviation
SKF	Svenska Kullagerfabriken (Swedish Ball Bearing Factory)
SOM	Self-Organizing Map

STFT	Short Time Fourier Transform
SVD	Singular Value Decomposition
SVM	Support Vector Machine
TFCF	Time-frequency Compression Fusion
TMSST	Time-reassigned multi-synchro-squeezing Transform
VAE	Variational Autoencoder
VMD	Variational Mode Decomposition
WPT	Wavelet Packet Transform

GENERAL INTRODUCTION

In the industry field, mechanical machinery is exposed to damages and failures for many reasons such as poor maintenance, overloading, material defects, operator error, and environmental factors **[1]**. Each one of these can lead to serious consequences, including production downtime, loss of revenue, damage to equipment, and even injury or loss of life **[2]**.

Implementing periodic maintenance and inspection programs that include routine maintenance, lubrication changing, vibration monitoring, and replacement of worn parts is necessary to prevent or minimize these failures **[3]**. Typically, the fault detection, diagnosis, and prognosis process are referred to as failure prevention:

Fault detection: The measured system data and system status information are observed and compared to a typical range of observed characteristics to discover if any measurements are outside the range reflecting the system's healthy state **[4]**.

Diagnosis: The procedure at hand comprises establishing the reasons for the failure and the status of the failing components **[5]**.

Prognosis: Predicting imminent component failures or aberrant system states and calculating the usable lifespan of such components **[6]**.

In rotating equipment, the bearing is the key mechanical element that is utilized to minimize friction between two moving parts. It permits for smooth and effective rotation or movement of one portion concerning another. it may be found in a wide range of applications, from machinery and equipment to cars and home goods **[7]**. The failure of this item can lead to excessive vibrations, noise, and, in severe circumstances, machine failure.

Effective bearing problem detection is vital for preserving the dependability and lifetime of machinery, as well as for reducing costly downtime. Several techniques are utilized for bearing failure identification and condition monitoring, including vibration analysis, acoustic emission analysis, oil analysis, thermography, motor current analysis, and ultrasound analysis **[8]**.

Analysis of vibration is a common technique for diagnosing bearing flaws as it enables early fault diagnosis and predictive maintenance **[9]**. Vibration analysis comprises monitoring the vibration signals produced by the bearing and detecting fluctuations in frequency and amplitude.

Bearing vibration signals are intricate and encompass a large array of frequencies. The kind of trouble in the bearing dictates the vibration signal's frequencies. A damaged outer raceway, for example, will provide vibration signals at the outer raceway frequency, but a damaged rolling element will offer vibration indications at the rolling element frequency **[10]**.

An essential area of study in mechanical engineering and machine maintenance is bearing fault diagnosis. because they are dynamic and complex. It is essential to create efficient methods for the early identification and diagnosis of bearing faults. Machine learning classification algorithms can be used to diagnose bearing faults based on vibration analysis to ascertain whether the bearing is sound or faulty **[11]**. Support vector machines, artificial neural networks, decision trees, and random forests are just a few examples of machine learning algorithms that have been used for bearing fault diagnosis. The algorithm selected will depend on the particular application, the size, and the complexity of the data set.

Over the years, numerous methods for diagnosing bearing faults have been proposed, including vibration analysis, acoustic emission analysis, temperature analysis, and oil analysis. One of the most widely used methods for identifying bearing faults is vibration analysis **[12]**. Determining the type, severity, and location of the fault involves measuring the vibration signals produced by the bearing and analysing them. Another method involves identifying the high-frequency noise produced by the bearing while it is in use, called acoustic emission analysis. Less popular methods that monitor the temperature and oil quality to look for changes linked to bearing faults include temperature analysis and oil analysis **[13]**.

Pattern recognition diagnostic is a technique used to identify and diagnose bearing faults in machinery. Pattern recognition diagnostic approaches are based on two main steps: the extraction of a vector of attributes also called features and the choice of detection rules that allow the classification of observations. For the first phase, many research works were carried out to extract the most appropriate features **[14]**.

S. Fang and W. Zijie presented a new method based on wavelet analysis and RBF-type and neural networks for the diagnosis of defects in ball bearings **[15]**, in this method, the results application of two types of mother wavelet were compared, namely the db8 wavelet and that

of sym8, the results show that the sym8 wavelet gives better performance. Unfortunately in this last work, only two mother wavelets were considered, while other types of wavelets can offer better results. The optimal level of decomposition is not so justified.

S. Fu, K. Liu, Y. Xu, and Y. Liu doing a study for bearing condition monitoring based on statistical parameters to detect incipient defects **[16]**. The results obtained show that this method offers good performance. Nevertheless, the statistical parameters are very sensitive to noise due to the acquisition system.

Several studies have shown that using a combined feature can help with better representation. The employment of this method, however, unfortunately, increases the input vector's dimension, making the classification step more difficult. To fix these issues, it will be necessary to choose a few characteristics that accurately describe the bearing's state. In general, parameter selection methods can be divided into two categories: filtering methods, which obtain the pertinent attributes based on the evaluation criteria regardless of the classification system, and wrapping methods (wrapper), which choose the pertinent parameters based on the classifier's output.

Y. Lei, Z. Hea, Y. Zia and Q. Hua have proposed a new approach for fault diagnosis **[17]**, this approach is based on the combination of certain pre-processing techniques, such as filtering, demodulation and Empirical Mode Decomposition (EMD), for the extraction of the global features. Six sets of features including time domain and frequency domain statistical features were obtained. Finally, the improved distance evaluation technique is proposed and it is with it that six attributes among the six sets are selected.

C. Shen, D. Wang, F. Kong and P. W. Tse presented a new fault diagnosis scheme based on the extraction of statistical parameters from the coefficients obtained by the application of the wavelet packet transform, the Distance evaluation technique (DET) is subsequently applied for the selection of the most relevant parameters **[18]**.

I. Rashedul, A. K. Sheraz and K. Jong-Myon in their study proposed a method for monitoring and diagnosing faults in bearings **[19]**. In this study, the original shape vector is a set of hybrid statistical parameters calculated from temporal analysis, frequency domain analysis and envelope spectrum analysis of acoustic emission. Genetic algorithms are then used to select the set of optimal attributes.

R. Ziani, A. Felkaoui and R. Zegadi have developed a new method based on the extraction of parameters in the time domain, the spectral domain and the time-spectral domain. Binary particle swarm optimization is then used for the selection of relevant attributes **[20]**.

This work is a thorough investigation into the use of machine learning-based automatic fault diagnostic techniques to non-stationary signals obtained from bearings. It is presented in the next sections of this thesis.

The goal of this research is to provide the development of an effective and reliable technique using one of the machine learning techniques for more accuracy in the detection, efficiency, and diagnosis of incipient bearing faults. The main objectives of this research are:

1_ To select the suitable bearing fault dataset based on the vibration signal, and also to present and discuss an overview of rolling element bearings, including bearing types and components, their applications, and their failure. In addition, the importance of maintaining the bearing's condition.

2_ Signal pre-processing, this phase depends on three techniques. The first is to apply a filtering technique to this dataset, among many techniques, the Independent Vector Analysis (IVA) has been chosen. The main advantage of IVA is that it can separate the individual sources and remove any noise or interference that may be present without requiring any prior knowledge of the sources themselves. The second is feature extraction, many techniques in signal processing were developed and applied in feature extraction for bearing faults diagnosis, which enabled features to be extracted effectively. In particular, statistical parameters in time domain-based methods were investigated in this study. The third technique is feature selection which is a crucial technique in machine learning. In feature selection, a smaller subset of pertinent features is chosen from a larger set of features that are present in the dataset. The purpose of feature selection is to decrease the dimensionality of the input data and remove unnecessary or redundant features to enhance the performance of a machine-learning algorithm.

3_ The last objective is to assess a classification methodology to determine whether the bearing is healthy or faulty and specify the type of defect(s) as well as the level of the defect severity by applying machine learning techniques. This classification was achieved to some extent by appropriate feature extraction from vibration signals.

The rest of this thesis is divided into four chapters namely:

Chapter 1 is the state the art of bearing fault diagnosis and related research.

Chapter 2 focuses on feature extraction and selection for pattern recognition using feature extraction, time domain parameters, frequency domain parameters, feature selection, binary optimization algorithms, and Conclusion.

Chapter 3, fault classification and diagnosis techniques are discussed such as artificial neural networks, random forests, support vector machines, and extreme learning machines, and at the end the conclusion.

Chapter 4 presents bearing fault diagnosis based on pattern recognition, this chapter contains the following outlines, Introduction. Rolling element bearing: structure and types. Degradation modes of a rolling element bearing. Rolling element-bearing fault detection and diagnostics. Influence of defects on the signal structure. Dynamic and Identification Research Group (DIRG) Dataset. Proposed method. Obtained results and Conclusion.

CHAPTER I

BEARING FAULT DIAGNOSIS: THE STATE OF THE ART

I.1. Introduction:

In the industrial field, rotating machines play a crucial role in manufacturing, each one of them contains many components, among them the bearing. The bearing is a mechanical part and it is the most vital component of a rotating machine, which maintains the shaft's steady position while rotating and minimizes friction and noise as much as possible. There are several types of bearings available with different features like shape, strength, and field of use. Commercially, there are some common bearings such as plain bearings that are utilized for sliding, and rotating motion, and for high dimensional and geometrical precision. The fluid film bearings are widely used in weighty rotating machines in which the stationary and rotating parts are isolated by a thin film of lubricants. Magnetic bearings, these types are suitable for fields that require high speeds and low vibration and no need for lubrication. The rolling element bearings come in different types like ball bearings, cylindrical roller bearings, spherical roller bearings, tapered roller bearings, needle roller bearings, and thrust bearings, they are used for both radial and thrust loads, can resist a small amount of weight, have low friction and also, they are the cheapest to manufacture.

In general, a rolling element bearing consists of the inner race, which holds the rotating shaft and rotates along with it. The outer race fitted on the housing and stays stationary. The rolling elements come between the inner & the outer race. The cage holds the space between rolling elements. in our case, we are interested in a type of widely used bearing, namely rolling element bearings, precisely the ball bearing, this type is with one row of balls and with radial contact. **Figure I.1** shows typical rolling elements bearing components.

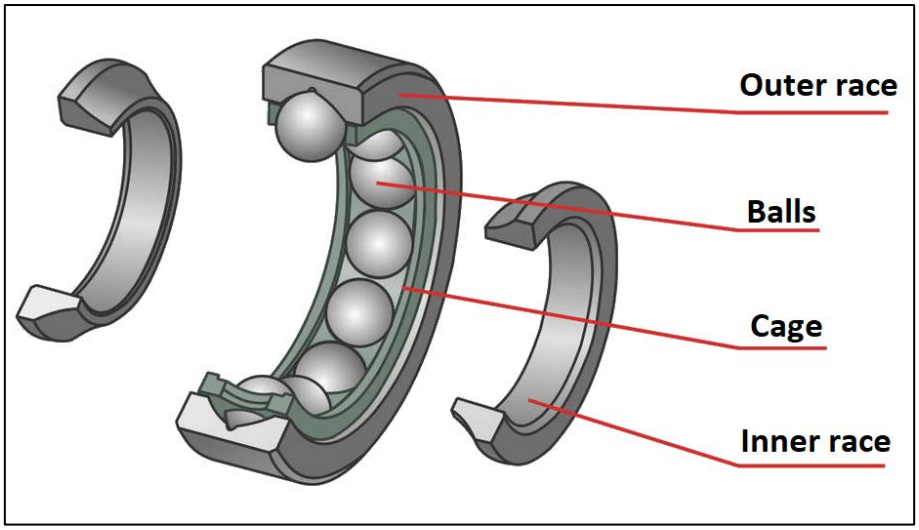

***Figure I. 1**. Structure of a rolling element bearing*

Rolling element bearings failures have been a subject of recent studies, due to their significance and their effects on the machine's performance. The failure occurs for many reasons such as high dynamic loads, improper or lack of lubrication, small debris side, flaking…etc. Once the failure occurs, it will lead to the instability of a machine which causes vibrations. The vibration signal carries information that helps us in fault detection and diagnosis by applying different algorithms in signal processing and data classification.

The damage to the bearing components creates one or more characteristic failure frequencies in the frequency spectrum that allow us to identify them quickly and easily. The four possible bearing failure rates are **[21]**:

1. BPFO (Ball Pass Frequency Outer): Ball Pass Frequency for the outer ring is the frequency at which a rolling element in a bearing passes a fixed point on the outer ring. It is calculated as the product of ball speed and the number of balls in the bearing. The BPF of the outer ring affects its operating performance, with higher BPF leading to greater dynamic loading and increased potential for failure. It is important to consider BPF in the design and selection of bearings to ensure reliable operation and prevent premature failure.

2. BPFI (Ball Pass Frequency Inner): Ball pass frequency, also known as BPF, in a bearing refers to the number of times a ball rotates in one minute (RPM) divided by the number of balls in the bearing. It is a measure of the stability and uniformity of the bearing. The BPF of an inner race is determined by the number of balls, their diameter, and the speed of rotation. High BPF can indicate a high-quality bearing, while low BPF can indicate bearing issues such as wear or misalignment.

3. BSF (Ball Spin Frequency): The ball spin frequency in a bearing is the number of Rounds per minute (RPM) that a ball makes as it rotates within the bearing. It is influenced by several factors, including the rotational speed of the shaft, the size of the ball, the dimensions of the bearing, and the load on the bearing. The ball spin frequency helps determine the overall performance and efficiency of a bearing system.

4. FTF (Fundamental Train Frequency): The fundamental train frequency (FTF) in bearings refers to the natural frequency at which the bearing vibrates. It is a function of the bearing's geometric parameters (such as the inner and outer diameter, the number of rolling elements, and the bearing stiffness) and operating conditions (such as the speed, load, and lubrication). The FTF can provide valuable information about the health and condition of a bearing and is used in machinery vibration analysis to detect early signs of bearing wear and fatigue.

I.1. Formulas for the calculation of the bearing fundamental failing frequencies

Bearing fundamental frequencies are the natural frequencies of vibration that are inherent to a bearing and its supporting structure. These frequencies are determined by the size, shape, and material properties of the bearing and its housing, as well as the loads and operating conditions. Understanding and accurately predicting bearing fundamental frequencies are important in the design and operation of rotating machinery, as they can influence the lifespan and performance of the bearing and contribute to excessive noise and vibration levels **[22]**.

The fundamental fault frequencies (in Hz) can be calculated using Equations **(1)** to **(4)**:

$$\boldsymbol{BPFO = \frac{n \times N}{2}\left(1 - \frac{D_b}{D_p}\cos(\theta)\right)} \quad (1)$$

$$BPFI = \frac{n \times N}{2}\left(1 + \frac{D_b}{D_p}\cos(\theta)\right) \quad (2)$$

$$BSF = \frac{n \times N}{2}\left[1 - \left(\frac{D_b}{D_p}\cos(\theta)\right)^2\right] \quad (3)$$

$$FTF = \frac{n}{2}\left(1 - \frac{D_b}{D_p}\cos(\theta)\right) \quad (4)$$

$$D_P = \frac{D_1 + D_2}{2} \quad (5)$$

With:

N: Rotation frequency [Hz];

n: Number of balls;

D_p: Mean diameter [mm];

D_b: Ball diameter [mm];

θ: Contact angle [degree]. (See **Figure I. 2**).

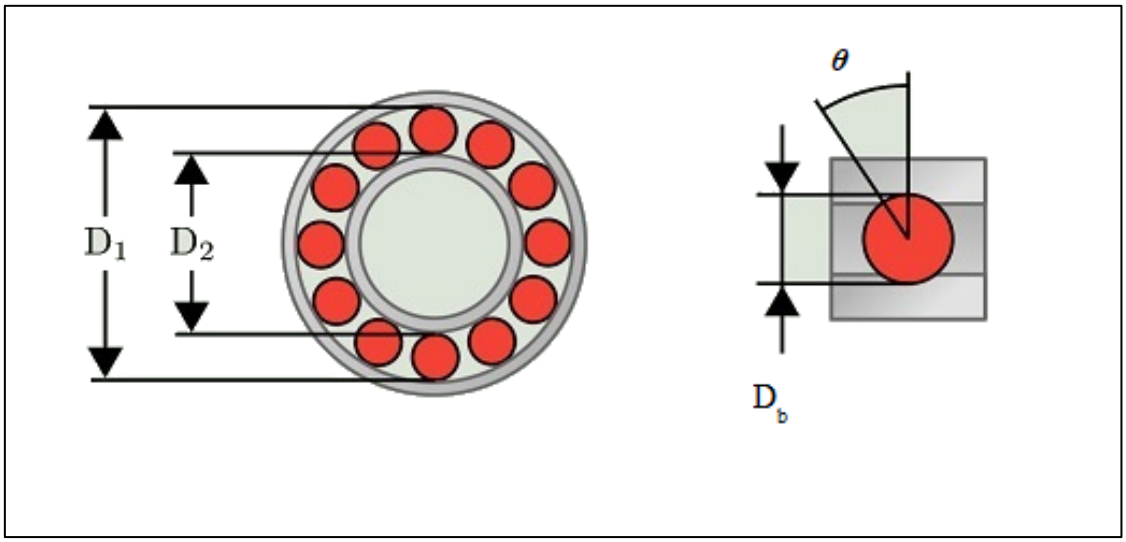

Figure I. 2 Dimensions of a bearing

I.2. Independent Vector Analysis (IVA)-based rolling element-bearing signal pre-processing:

There are two rolling element-bearing fault severity datasets used in this work. One was collected at Case Western Reserve University (CWRU) Bearing Center **[23]**, and the second has been obtained from the Dynamic and Identification Research Group (DIRG) in the Department of Mechanical and Aerospace Engineering at Politecnico di Torino, Italy **[24]**. These two datasets have different fault severity and they had been collected from different rig sets up.

I.2.1. Vibration-based

A common technique for determining the severity of problems in rolling element bearings is vibration analysis. This method measures the vibration produced by the faulty bearing and compares it to a baseline measurement taken from a healthy bearing **[25]**. Vibration analysis is based on the fact that a faulty bearing produces a unique pattern of vibration that can be used to identify the type and severity of the fault. The vibration measurement can be obtained using accelerometers or other types of sensors that are placed on or near the bearing **[26]**.

I.2.1.1. Case Western Reserve University (CWRU) Bearing Centre dataset:

This dataset has been provided by CWRU Bearing Centre **[27]**, it contains several vibration signals of various conditions of bearings and it has been collected using the test rig, which consists of a 2-horsepower induction motor, a dynamometer, a transducer. as shown in **Figure I.3**.

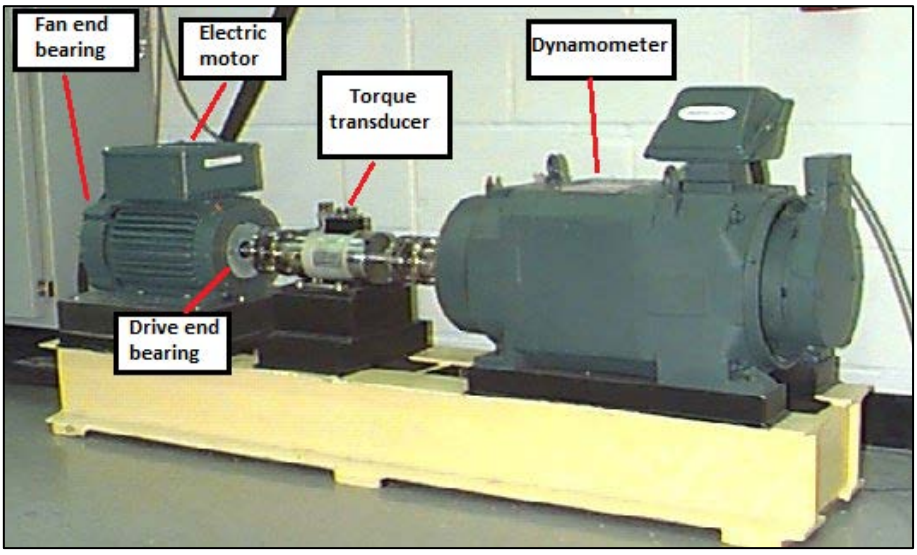

Figure I. 3 *CWRU bearing test rig [23] for collecting vibration signals*

The vibration signals of the artificially planted faults are acquired using accelerometers, which are mounted on the top of the housing at the Drive-End (DE) of the motor. These signals are with a sampling frequency of 12 000 samples per second using a 16-channel data recorder and with different applied loads and the speed range of the motor is 1722-1797 RPM. Several types of conditions of the bearings have been used which are Normal Type (HTY), Ball Type (BF), inner Race Type (IF), and Outer Race Type (OF). The bearings are SKF deep-grove ball of 6205-2RSJEM type for the Drive-End. The details of this dataset are listed in **Table 1**.

Table I. 1 Information about the dataset obtained from the CWRU-bearing data bank

Dataset	**Health Type**		**Crack Size (mm)**	**Speed (RPM)**	**Load**	
Dataset 1	Ball Fault	BF1	0.1778	1797	0	0% of the nominal load (Unloaded case)
		BF2	0.5334		0	
	Inner Race Fault	IF1	0.1778		0	
		IF2	0.5334		0	
	Outer Race Fault	OF1	0.1778		0	
		OF2	0.5334		0	
	Healthy	HTY	/		0	
Dataset 2	Ball Fault	BF1	0.1778	1772	1	50% of the nominal load (Half loaded case)
		BF2	0.5334		1	
	Inner Race Fault	IF1	0.1778		1	
		IF2	0.5334		1	
	Outer Race Fault	OF1	0.1778		1	
		OF2	0.5334		1	
	Healthy	HTY	/		1	
Dataset 3	Ball Fault	BF1	0.1778	1750	2	100% of the nominal load (Full loaded case)
		BF2	0.5334		2	
	Inner Race Fault	IF1	0.1778		2	
		IF2	0.5334		2	
	Outer Race Fault	OF1	0.1778		2	
		OF2	0.5334		2	
	Healthy	HTY	/		2	

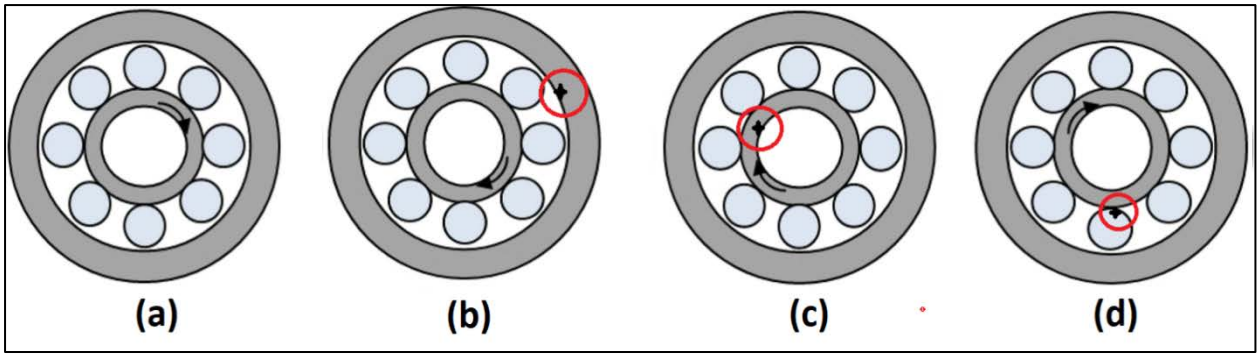

Figure I. 4 *Schematic of fault place in bearing parts: (a) healthy bearing (HTY), (b) bearing with outer race fault (OF), (c) bearing with inner race fault (IF), and (d) bearing with ball fault (BF).*

Table I. 2 The outer ring's frequencies at various motor speeds [27]

Inside diameter mm	Outside diameter mm	Ball diameter mm	Thickness mm	Pitch diameter mm	Contact angle °	Ball number
25.001	51.999	7.940	15.001	39.040	0°	9

Table I. 3 The outer ring's frequencies at various motor speeds [27]

Motor Speed (RPM)	1797	1750	1724
Defect frequency (Hz)	107.36	104.56	103.00

I.2.1.2. Dynamic and Identification Research Group (DIRG) dataset:

The provided dataset by the DIRG laboratory in the Department of Mechanical and Aerospace Engineering at Politecnico di Torino has been collected over a rig set up **[24]**, for testing high-speed aerospace bearings, their acceleration measurements at variable speed, radial load, and degree of damage. The test bench is depicted in **Figure I.5**, and it contains a high-speed spindle conducting a hollow shaft supported by identical roller bearings B1 and B3. The considered bearing is B1, where a tri-axial accelerometer is mounted on its support.

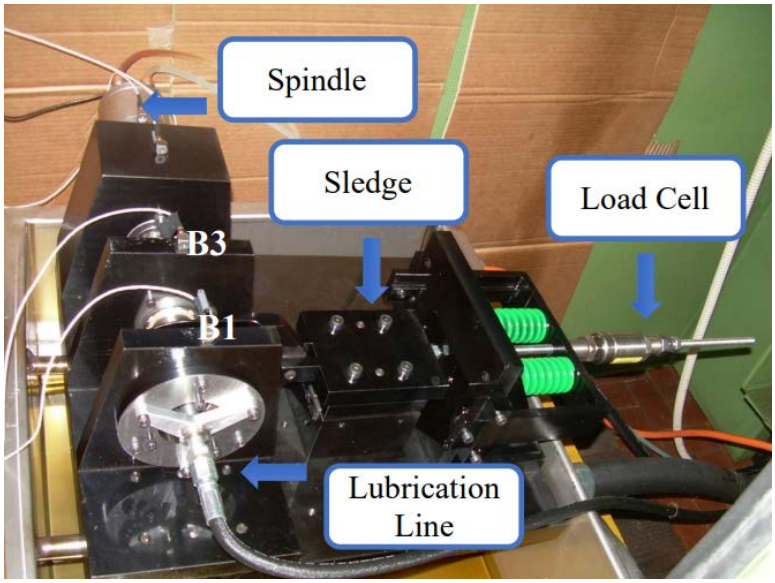

Figure I. 5 *The test bench of DIRG*

The base of the test bench carries in positions B1 and B3 a duo of supports for the outer race of two identical roller bearings. The inner race of these bearings is connected to a very short and strong hollow shaft, which is specially designed for speeds of up to 35000 rpm. The shaft was originally part of a complete gearbox and carried a spur gear that drives the rotation. Due to the torque applied, the spur gear generated a contact pressure with radial and tangential direction, which existed on a pair of roller bearings. The outer race of the bearing B2 is attached to a precision sledge. The principal geometrical properties of the three bearings, particularly manufactured for this high-speed aeronautical application, are listed in **Table I.4**.

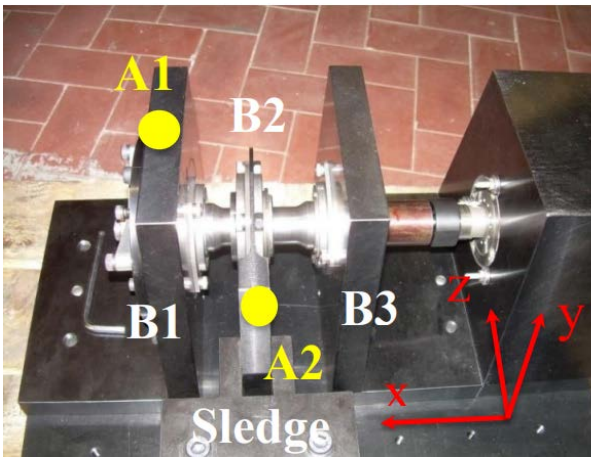

Figure I. 6 *The positions of the two accelerometers*

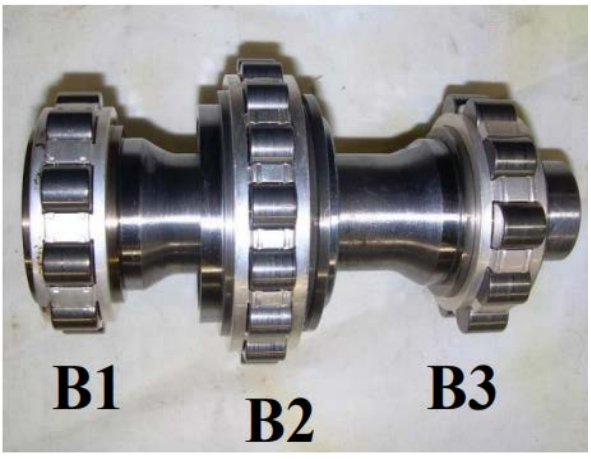

Figure I. 7 *The three roller bearings on the shaft*

Table I. 4 The primary characteristics of roller bearings

	Pitch diameter D (mm)	**Roller's diameter d (mm)**	**Contact angle ϕ (°)**	**Rolling elements Z**
B1 & B3	40.5	9.0	0	10
B2	54.0	8.0	0	16

The bearing in position B1 **Figure I.7** is designed in such a way that it can be disassembled from its support in a very simple way to be able to check the reaction of the system when mounting bearings with different types and degrees of damage. **Table I. 5** contains the names of the damaged items (1A to 6A), with 0A indicating the undamaged case.

Table I. 5 A Roller's list of the various bearings mounted in position B1's known defects

Name	Fault	Dimension(µm)
0A	The inner ring's indentation's diameter	450
1A	The inner ring's indentation's diameter	250
2A	The inner ring's indentation's diameter	150
3A	The roller's indentation's diameter	450
4A	The roller's indentation's diameter	250
5A	The roller's indentation's diameter	150

The acquired dataset has been collected with sampling frequency fs = 51200 Hz for a duration of T = 10 s. Each file contains an array with the same name as the file (except for the .m) with 819200 rows (time samples) and 6 columns (one for each channel). Every recorded file in the dataset whose name has the following format: **CnA_fff_vvv_m.mat.**

- **C**: The root of the file name shared with all files;
- **n**: Integer value from 0 to 6, indicating the type of the defect, e.g., 0A, 1A, …, 6A (Table I. 5);
- **fff**: Integer value from 100 to 500, denoting the nominal speed of the shaft (Hz);
- **vvv:** Integer value corresponding to the voltage of the load cell (mV), denoting the applied load;
- **m**: Integer value, denoting if the measurement has been duplicated (m=2) or not (m=1);
- mat: MATLAB® file extension.

I.2.2. Acoustic emission (AE) based

Bearing fault diagnosis using AE involves analysing sounds generated by bearings during their operation to identify specific types of faults, such as cracks, looseness, or wear **[28]**.

AE signals are generated by the release of energy from the friction, impact, or deformation of the material in the bearing. This energy is converted into acoustic waves that propagate through the bearing and can be picked up by sensors attached to the bearing housing **[29]**.

The signals generated by the bearing faults are analysed to extract features that are indicative of specific faults, such as cracks, looseness, or wear. Machine learning algorithms among them ANN **[30]**, RF **[31]**, and SVM **[32]**. These algorithms are commonly used to classify the signals into different fault categories **[33]**.

This is a non-destructive method of detecting faults in rotating machinery components, including bearings.

Acoustic Emission (AE) is a very precise and efficient technology for detecting and monitoring faults, leaks, and fatigue. AE-based analysis can detect very low-energy signals generated by bearing failures early in the process or during slow-speed operation. AE signals have several benefits over other sensor signals, such as vibration signals, in terms of capturing and reflecting both local and global bearing defect aspects **[33]**. Signal processing and useful feature extraction are critical steps in employing AE sensors for machine failure diagnostics. The AE, unlike vibration, is less impacted by noise and structural vibration. For example, vibration sensors have a more difficult time capturing the high-frequency resonances of the structure of a bearing fitted in a mechanical system than AE sensors **[34]**.

AE-based analysis may recognise very low-energy signals generated by defective bearings early in the process or while in slow-speed operation. AE signals have several benefits over other sensor signals, such as vibration signals, in terms of capturing and reflecting both local and global bearing defect aspects. Signal processing and useful feature extraction are critical steps in employing AE sensors for machine failure diagnostics **[28]**. AE-based bearing fault diagnosis is a promising method for detecting faults in rotating machinery components. It is a non-invasive and cost-effective method that provides valuable information about the condition of the bearings.

I.2.3. Infrared thermography based

Infrared (IR) thermography is a non-contact, non-destructive method of diagnosing faults in bearings. The technique involves using a thermal imaging camera to detect changes in temperature in the bearing, which can indicate the presence of a fault **[35]**.

A healthy bearing will have a uniform temperature distribution, while a faulty one will exhibit hot spots due to increased friction and energy loss. The temperature anomalies detected by

infrared thermography can be used to diagnose various types of faults, including outer race defects, inner race defects, ball defects, and cage defects **[36]**.

IR thermography can be performed while the machine is in operation, making it a valuable tool for predictive maintenance. The method is easy to use and does not require any disassembly of the machine, reducing the risk of further damage. in conclusion, R thermography is a useful method for detecting faults in bearings and can provide important information for predictive maintenance.

I.2.4. Oil analysis based

Oil analysis is a diagnostic method used to detect faults in bearings by analysing the properties of the lubricating oil. The principle behind this technique is that the lubricating oil circulates through the bearing, picking up metal particles, wear debris, and other contaminants, which can provide valuable information about the condition of the bearing **[37]**.

Oil analysis can be performed using various techniques, such as spectrometry, particle counting, and viscosity measurements. Spectrometry can be used to detect the presence of metal particles and determine their elemental composition, while particle counting can be used to quantify the number and size of particles in the oil **[38]**. Viscosity measurements can provide information about the overall health of the lubricating oil and can indicate changes in the oil that may be due to bearing wear.

To diagnose bearing faults using oil analysis, the oil must be sampled and analysed periodically, and the results must be compared to established norms. Any deviations from the norms can indicate the presence of a fault, such as excessive wear or corrosion, in the bearing.

In summary, oil analysis is a valuable diagnostic tool for monitoring the condition of bearings and predicting potential failures **[39]**. By analysing the properties of the lubricating oil, this technique can provide valuable information about the health of the bearing and help prevent unexpected failures.

I.2.5. Bearing current analysis based

Bearing current analysis is a method that detects the presence of stray electrical currents flowing in the bearing metal, which can indicate incipient fault conditions such as damage or wear. These currents can be measured using specialized sensors and analysed to determine the presence of faults **[40].**

The basic principle behind bearing current analysis is that as a bearing begins to fail, it generates electrical currents that can flow along the metal components of the machinery. These currents can be measured and analysed to determine the type of fault that occurred **[41]**. The analysis typically involves filtering the measured current signals to remove noise and other interference and then comparing the resulting signal to known fault signatures.

Bearing current analysis has several advantages, including its ability to detect incipient faults before they become serious, its non-intrusive nature, and its ability to work even in harsh industrial environments **[42]**. Additionally, bearing current analysis can be used in conjunction with other diagnostic techniques such as vibration analysis to provide a more exhaustive understanding of the health of the machinery **[43]**.

It is important to note that to get accurate results from bearing current analysis, the equipment and sensor setup must be properly calibrated, and the analysis should be performed by trained professionals who understand the technique and can interpret the results correctly **[44]**. The advantages of bearing fault diagnosis using current analysis **[45]**:

Early Detection: The approach can discover bearing defects early on, allowing maintenance crews to intervene before a small issue escalates into a significant issue.

Non-Intrusive: Current analysis is a non-intrusive approach for assessing equipment health since it does not involve direct contact with the bearings.

Cost Savings: By allowing predictive maintenance, firms may minimise unplanned downtime, save repair costs, and prolong equipment lifespan

Real-time Monitoring: Continuous monitoring of the present signature enables real-time monitoring of the machine's state.

Data-Driven Insights: Data obtained may be analysed over time to detect trends and patterns in bearing health, assisting in decision-making and maintenance planning.

In conclusion, bearing defect detection based on current analysis provides various advantages over previous techniques, including lower cost and non-invasiveness. Traditional approaches, however, might make it difficult to discern the distinctive frequency of bearing problems in the current spectrum. Deep learning and information fusion strategies have been presented in recent research to increase the performance of this approach, and these methods are useful in detecting and diagnosing bearing defects.

I.3. Independent Vector Analysis (IVA)-based rolling element-bearing signal pre-processing:

A set of mixed signals can be separated into their original source signals using the blind source separation technique known as Independent Vector Analysis (IVA). Finding a set of statistically independent vectors that together form a linear mixing matrix and can be used to decouple the original source signals from their mixed observations and estimate the transformation matrix, which is the aim of IVA **[46]**.

It is possible to separate the mixed signal into its underlying independent sources after the transformation matrix has been estimated. The sources can then, as needed, be further analysed or processed after being separated. The separated sources, for instance, can be used to enhance speech comprehensibility and noise reduction in speech recognition, for sound source localization or music separation in audio processing, and can be used in biomedical signal analysis to monitor vital signs or to diagnose a variety of illnesses **[47]**.

IVA does not need to know anything about the source signals or the mixing matrix beforehand, unlike other blind source separation methods. Even when sources are highly correlated or have non-Gaussian distributions, it has been demonstrated that it can still effectively separate them **[48]**. Compared to the independent component analysis technique independent component Analysis (ICA), it is more useful and efficient **[49]**.

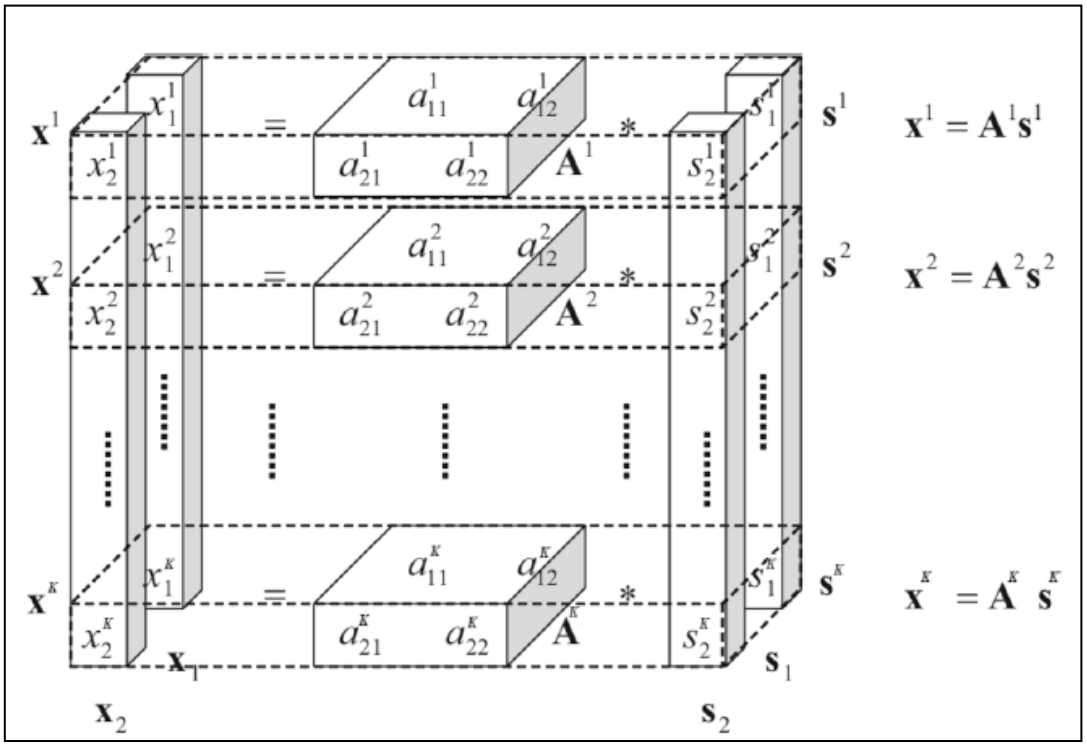

Figure I.8 Independent vector analysis (IVA)'s mixture model

According to **[50]**, the IVA mixing and separation models are as follows:

$$x^{[k]}(t) = A^{[k]}(t)s^{[k]}(t) + n(t)\ ;\ \forall k\ \in\ \{1, \dots, K\} \quad \textbf{(6)}$$

Denotes:

The observation vector, x[k](t), contains the mixtures of independent sources. The mixing matrix is denoted by A[k], the source vector is denoted by s[k](t), K stands for the number of datasets, and the additive Gaussian noise is represented by n(t).

The Blind Source Separation (BSS) approach determines K demixing matrices and source vector estimations for each dataset, with the kth ones being labelled as W[k] and Y[k] = W[k] X[k], and each one is identifiable by the letters W[k] and X[k]. Thus, we have:

$$y_n^{[k]} = \left(w_n^{[k]}\right)^T x^{[k]} \quad \textbf{(7)}$$

The IVA cost function, which can be represented as **[51]**, maximises the mutual information within each source component vector (SCV).

$$\mathcal{J}_{IVA} = \sum_{n=1}^{N}\left(\sum_{k=1}^{K}\mathcal{H}\left[y_n^{[k]}\right] - I[y_n] - \sum_{k=1}^{K} log\left(\left|det\left(W^{[k]}\right)\right| - C_1\right)\right) \quad \textbf{(8)}$$

Where:

W denotes the set of all de-mixing matrices,

$\boldsymbol{W^{[k]}}$ a value to be estimated,

$\mathcal{H}$ stands for the entropy of the source estimates,

$\mathcal{J}_{IVA}$ represents the mutual information within an SCV,

$\boldsymbol{C_1}$ signify a constant.

I.4. Rolling element-bearing fault diagnosis based on pattern recognition:

Pattern recognition plays an important role in diagnostics and can be used as a generic term. It's a segment of machine learning that focuses on spotting patterns and regularities in data. It's also known as data classification based on prior learning or statistical data pulled from patterns and/or their representation. This type of detection can be performed for different types of input, e.g., biometric recognition, colours, image recognition, and face recognition **[52]**. The structure process for a pattern recognition system is as follows **[53]:**

- **Sensing:** In this step, sensors are used to receive data and measure properties like vibration, pressure, temperature…etc.
- **Segmentation:** This step is to divide the received data of a case into several segments or sub-signals to gather so much information on a case and to make sure that the sensed objects are separated.
- **Feature extraction:** In this step, several useful information have been extracted and computed from a signal or an image. These features distinguish one case from another and they represent the input of a classification algorithm. They could be in the time domain like statistical parameters or frequency domain like Power Spectral Density (PSD).
- **Classification:** This phase makes a connection between the extracted features with their correct category or label using a specific model that has prior training. Many machine learning algorithms provide data classification such as statistical algorithms, Structural algorithms, Neural network-based algorithms, Template matching algorithms, Fuzzy-based algorithms, and Hybrid algorithms.
- **Post-processing:** Here, additional reviews are made before a conclusion is made.

Bearing fault pattern recognition is the process of identifying and diagnosing faults that occur in bearings. Bearing fault pattern recognition techniques have been studied in recent years to enhance the reliability of error detection and diagnosis. These techniques can identify bearing conditions even when the computations are not entirely accurate as long as all the computations are based on a similar approximation **[54]**. Several machine learning classifier algorithms have been used in bearing fault pattern recognition such as ANN **[55]**, SVM **[56]**, ELM **[57]**, and RF **[58]**. The selection of the appropriate pattern recognition algorithm can increase the accuracy of the system **[59]**.

I.5. Related Researches:

Bearing fault diagnosis using machine learning is a rapidly growing field that leverages the capabilities of machine learning algorithms to detect and diagnose faults in bearings. The use of vibration analysis in combination with various machine learning techniques is effective in detecting and diagnosing bearing faults. Some of the machine learning techniques used for this purpose include traditional methods such as envelope analysis and wavelet transform, as well as more recent approaches such as deep learning.

Sun, M. et al. (2022) [60]: They suggested a stack autoencoder transfer learning method (SAE-CSDF) based on class separation and domain fusion to overcome several challenges in fault detection, such as low accuracy or the necessity for some labelled data containing fault information in the new machine. The results demonstrate that the algorithm's accuracy may approach 97% when transferring data across computers, even when the new machine lacks tagged fault data.

Choudhary, A., Goyal, D. and Letha, S. S. (2021) [61]: In this work, an emergent two-dimensional discrete wavelet transforms (2D-DWT) based Infrared thermography (IRT) approach for identifying various bearing defects in induction motors (IM) was developed. The Mahala Nobis distance (MD) was used in feature selection to produce the ideal feature set. A support vector machine was used to classify the errors. The experimental results show that the SVM-based approach may be used to design a proactive robust system for defect identification. This paves the way for condition-based maintenance, which will aid in the prevention of catastrophic failures and the elimination of unanticipated breakdown costs.

Hou, J. et al. (2020) [62]: The vibration signal was decomposed in this study using Ensemble Empirical Mode Decomposition (EEMD). The entropy feature vector was then calculated by multiplying the permutation entropy values of each modal component. Following that, Linear Discriminant Analysis (LDA) was used to redact features. Experiments with data reveal that the suggested fault diagnostic approach may obtain acceptable clustering rates. This suggests that the error detection approach provided in this study has the benefit of giving superior compactness within the class of clustering results when compared to previous method combination procedures.

Zhang, X., Zhao, B. and Lin, Y. (2021) [63]: In this work, the CWRU dataset was employed. The present mainstream rolling bearing has been researched first. Following that, machine

learning-based bearing failure detection has been separated into three stages. The initial process is feature extraction, followed by feature selection, and finally classifier detection. Many machine learning techniques, such as SVM and KNN, were employed in the classification phase, as well as deep learning classifiers such as Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM). According to the report, the SVM had the best classification effect.

Sun, G. et al. (2022) [64]: The purpose of this study is to optimise the input of time-frequency images and intelligent diagnosis algorithms by thoroughly analysing the advanced time-frequency analysis algorithms multi-synchro-squeezing transform (MSST) and time-reassigned multi-synchro-squeezing transform (TMSST). Finally, time-frequency compression fusion (TFCF) has been suggested for superposing and splicing two time-frequency images to form dual-channel images. The proposed diagnostic model not only solves the issue of diagnosis in normal working conditions, but it also maintains acceptable performance with a small sample size, brief sampling time, and low sampling frequency, and it has a broad range of application possibilities.

Toma, R. N., Prosvirin, A. E. and Kim, J.-M. (2020) [65]: This paper describes a hybrid approach to bearing failure diagnostics based on motor current data that employs statistical characteristics, genetic algorithms (GA), and machine learning models. The statistical characteristics of the motor current signals are obtained. Second, GA is used to reduce the number of features in the feature database and prioritise them. Finally, three unique classification methods, namely K-Nearest Neighbours Algorithm (KNN), decision tree, and random forest, are designed and evaluated utilising these attributes to evaluate the bearing defects. The experimental results show that the three classifiers achieve an accuracy of more than 97%.

Li, J. et al. (2019) [66]: To satisfy the expectations for an effective evaluation of varied fault types and severities with real-time computing performance, the authors proposed a new diagnostic technique for rolling bearing failures based on multi-dimensional feature extraction and evidence fusion theory. To acquire health status feature vectors, vibration signals are initially processed using a multidimensional feature extraction approach based on entropy characteristics, Holder coefficient characteristics, and enhanced generalised box-counting dimension characteristics. Furthermore, using the recovered feature vectors, a grey relation method is employed to construct the basic belief assignments (BBAs), and the BBAs are fused

using the Yager algorithm to recognise bearing fault patterns. Based on a small number of training samples, the findings show that the diagnosis success rate for bearing defective circumstances is 100%, and the overall diagnostic success rate is close to 99.09%. In contrast to existing intelligent diagnostic approaches, the suggested approach enhanced fault diagnostic accuracy and may be suited for online bearing fault diagnosis.

Khan, S. A. and Kim, J.-M. (2016) [67]: Based on an automated approach for defect diagnostics in bearings, this work offers a 2D analysis of vibration acceleration data under changing speed situations. The pictures obtained from the vibration signals for each defect have different textures that vary only a little with shaft speed. These photos are used to generate unique fault signatures for each type of problem, which may then be used to identify those faults at different speeds. The k-nearest neighbour classifier is trained to utilise fault signatures created for one operating speed to discover problems at all other operating speeds. The authors determined that because the basic fault frequencies are dependent on the nonstationary shaft speed, shaft speed changes are inescapable. To validate the suggested approach, this study employed fault pictures at four distinct operating speeds. After training with pictures for one operating speed, the classification performance of a kNN classifier was tested by testing it with fault images for the remaining three operating speeds. The classifier's average classification accuracy of 99.74% illustrates that variations in shaft speed do not affect the proposed technique.

Li, W. et al. (2016) [68]: This study includes a unique feature in the form of visuals. The spectrum images are produced using only the rapid Fourier transformation. Two-dimensional principal component analysis (2DPCA) is used to decrease the dimensionality of these pictures. The bearing defects are then classified using the minimal distance approach. The effectiveness of the proposed technique was demonstrated using experimental vibration signals. The pictures might considerably improve the efficacy of defect detection by providing a new view of the FFT spectrum. Even with a short training sample, the suggested approach may achieve high accuracy.

Jiang, Y. and Xie, J. (2022) [69]: VMD and RP are used in the suggested bearing fault diagnostic technique based on VMD-RP-CSRN to maximise the retention of fault features in the original signal while emphasising the signal's hidden attributes. The recommended channel split operation gathers concealed features while allowing additional fault features to participate in the feature extraction process of the diagnosis model by selecting the major operational

channel of the three-channel feature picture. The experimental results show that the suggested technique beats the comparison method by at least 1.2% in terms of noise immunity. Although the excellent performance of VMD has been demonstrated in the literature, there are still some concerns about its use in bearing problem diagnostics.

I.6. Conclusion:

This chapter described the modes of failure of the rolling bearing components, their causes, and the standard methods for identifying bearing defects. Methods for measuring the severity of rolling element-bearing faults have been demonstrated. Based on pattern recognition, the rolling element-bearing fault diagnosis has been mentioned. Also, the used datasets in this study have been described.

CHAPTER II

FEATURE EXTRACTION AND SELECTION FOR PATTERN RECOGNITION

II.1. Introduction:

In the field of artificial intelligence and data analysis, feature extraction is a crucial concept. It is frequently used in practice to solve real-world problems. The extracted features from the data are fed into ML algorithms, which can then be trained to perform tasks like classification, regression, clustering, and more. The outcomes of ML models can be used to make decisions, make predictions, or gain insights into data.

Feature selection is a critical feature extraction component and plays an essential role in ML and data analysis. Selecting a subset of the most relevant and informative features from a high-dimensional data set to use as inputs to ML algorithms or other predictive models is known as feature selection, and it is critical for improving the performance and interpretability of ML algorithms and predictive models.

II.2. Feature extraction:

Feature extraction is a critical step in pattern recognition, which is identifying patterns and regularities in data. It involves selecting a subset of relevant information from the raw data and transforming it into features suitable for the specific machine-learning algorithm or task. The goal of feature extraction is to extract meaningful information from representative data that can effectively distinguish between different classes or patterns. Good feature extraction can result in improved accuracy, faster training, and reduced overfitting. This method is necessary for many machine-learning jobs and can have a considerable impact on the performance of the final model, making it a vital stage in the machine-learning process. In addition, the complexity of the data is decreased, making it easier to handle and analyse. Techniques for feature extraction include linear discriminant analysis, principal component analysis, and wavelet transform **[70]**.

The steps of feature extraction in ML typically involve the following **[71]**:

1. Data Preprocessing: This step involves cleaning and preparing the data, handling missing values, and normalizing the data if necessary.

2. Feature Selection: This step involves choosing the most relevant and informative features from the dataset. Feature selection helps to reduce the number of features, reduce overfitting, and increase the efficiency of the machine-learning algorithm.

3. Feature Transformation: This step involves transforming the selected features into a format that can be used by the machine-learning algorithm. This may involve creating new features, aggregating existing features, or converting continuous features into categorical features.

4. Dimensionality Reduction: This step entails reducing the number of features to a more manageable number. Methods such as principal component analysis (PCA), linear discriminant analysis (LDA), and singular value decomposition can be used to accomplish this (SVD).

5. Feature Scaling: This step involves scaling the features so that they are all on the same scale, which can help some ML algorithms perform better.

Generally, the purpose of feature extraction is to extract the most important information from the signal and describe it in a shape that the machine-learning algorithm can understand. These stages can be repeated many times to enhance the feature extraction process.

The advantages of feature extraction in ML include **[72]**:

1. Improved Model Performance: By identifying and extracting the most relevant features from a dataset, feature extraction can lead to improved performance of the machine-learning model.

2. Reduced Overfitting: By reducing the number of features, feature extraction can reduce the risk of overfitting, where the model is too closely fit to the training data and does not generalize well to new data.

3. Increased Efficiency: By reducing the number of features, feature extraction can make the training and prediction process more efficient and faster.

4. Better Interpretability: By reducing the number of features, feature extraction can make the results of the machine-learning model more interpretable and easier to understand.

5. Improved Data Visualization: By transforming the features into a more manageable format, feature extraction can make it easier to visualize and explore the relationships between features and the target variable.

II.3. Time domain features:

Bearing fault diagnosis using time-domain features involves extracting relevant features from the vibration signal generated by a faulty bearing and using these features to identify the type and severity of the fault. Some common time domain parameters used for bearing fault diagnosis include:

II.3.1. Root Mean Square (RMS):

has been used as a standard statistical metric in ML for evaluating the performance of a regression or classification model. The smaller the RMS value, the better the model is at making accurate predictions and detections. It is usually used as a loss function in optimization algorithms to train ML models, also used in time series forecasting or image classification and in conjunction with other metrics, like mean absolute error (MAE) or mean squared error (MSE), to obtain the full picture of the performance of an ML model. The choice of the evaluation metric counts on the characteristic requirements and goals of the problem being solved **[73]**.

The Root Mean Square is defined as follows **[74]**:

$$RMS = \sqrt{\left(\frac{\sum_{i=1}^{N}(x_i)^2}{N}\right)} \quad \textbf{(9)}$$

With:

i: Is a sample,

N: Is the number of data points (samples) of *x(i)*.

II.3.2. Kurtosis:

It is a typical statistical metric used to analyse the peakedness or flatness of data, similar to a normal distribution. A higher kurtosis value indicates a more peaked distribution, while a lower kurtosis value indicates a flatter distribution. Kurtosis can be used in experimental applications to discover outliers and specify the presence of a distributional skew, which might affect the performance of specific algorithms. While preprocessing and converting data in ML tasks, it is critical to include kurtosis as well as other statistical variables such as mean, standard deviation, and skewness **[75].**

In the field of bearing fault diagnosis, kurtosis is often used as a feature to detect abnormal conditions in rotating machinery. The vibration signals collected from bearings can contain information about the health of the bearings, and the kurtosis of these signals can be used to identify different types of faults, such as inner race faults, outer race faults, and ball faults **[76]**. It is defined as follows **[77]:**

$$\boldsymbol{Kur} = \frac{\frac{\mathbf{1}}{\boldsymbol{N}}\sum_{\boldsymbol{i=1}}^{\boldsymbol{N}}(\boldsymbol{x(i)} - \overline{\boldsymbol{x}})^{\mathbf{4}}}{\left[\frac{\mathbf{1}}{\boldsymbol{N}}\sum_{\boldsymbol{i=1}}^{\boldsymbol{N}}(\boldsymbol{x(i)} - \overline{\boldsymbol{x}})^{\mathbf{2}}\right]^{\mathbf{2}}} \quad \textbf{(10)}$$

where x(n) is a signal series for n = 1, 2, …, N. N is the number of data points. There are three types of kurtosis **[78]**:

1. Mesokurtic: This refers to a kurtosis value that is equal to the kurtosis of a normal distribution, which is 3. A mesokurtic distribution has a similar shape to a normal distribution, with relatively few outliers and a balanced distribution of data.

2. Leptokurtic: This refers to a kurtosis value that is greater than 3, indicating a more peaked distribution with a larger concentration of data in the tails. This type of distribution has a higher likelihood of extreme values or outliers.

3. Platykurtic: This refers to a kurtosis value that is less than 3, indicating a flatter distribution with fewer extreme values. A platykurtic distribution has a lower concentration of data in the tails compared to a normal distribution.

II.3.3. Crest Factor:

It's a measure used to quantify the peakiness or impulsiveness of a signal. In the domain of ML, it is used to evaluate the performance of models for signal-processing tasks. In addition, it is described as the ratio between the maximum amplitude of a signal and its root mean square (RMS) value. A lower crest factor indicates a smoother signal, while a higher crest factor indicates a more peaky signal **[79]**.

The crest factor can be used as a metric for evaluating the quality of audio signals and for comparing different models. It is also used in the field of mechanical engineering for bearing fault diagnosis. In this context, the crest factor is used to analyze the vibration signals generated by rotating machinery to detect and diagnose faults in bearings **[80]**. It is computed as the ratio of the maximum value to the root mean square (RMS) value of the vibration signal.

An increase in the crest factor can indicate an increase in the impulsive or non-sinusoidal component of the vibration signal, which is often associated with bearing faults such as pitting, spalling, or misalignment. By analyzing the crest factor, engineers can identify the presence of bearing faults and diagnose the type and severity of the fault. It is defined as follows **[81]**:

$$\boldsymbol{CRF} = \frac{\boldsymbol{max}|\boldsymbol{x}(\boldsymbol{n})|}{\sqrt{\left(\frac{\sum_{i=1}^{N}(x_i)^2}{N}\right)}} \quad \textbf{(11)}$$

A high crest factor indicates that the vibration signal has high peaks and a low average value, which is characteristic of a fault in the bearings. By Incorporating the Crest Factor Into the Feature Set of the ML Model, the Model Can Better Differentiate Between Normal and Faulty Signals, Leading to Improved Accuracy in Fault Diagnosis.

II.3.4. Impulse Factor:

In ML, the quick and unexpected effect that fresh data can have on a model's predictions is known as the Impulse Factor This can happen when the model is compelled to modify its predictions in response to the introduction of fresh data points. The phrase is frequently used in time series data modelling, where new data points stand in for the most recent observations **[82]**.

In some circumstances, outliers, or data points that differ noticeably from the other data points in the data collection, might provide the push factor. They can significantly affect the model's predictions since they can skew the data's distribution and make the model forecast things incorrectly. In other instances, an abrupt shift in the data distribution may be to blame for the gain factor. For instance, the model may lose its ability to correctly forecast future observations based on historical data if the underlying mechanism that generates the data changes **[83]**.

The impulse factor can be introduced into a bearing defect diagnosis system by abrupt changes in the vibration signals produced by the bearing. These alterations can be brought on by the existence of a problem, but they can also be brought on by outside variables such as variations in load or speed **[84].** To establish an appropriate diagnosis, the model utilised in the diagnosis system must be able to separate between these many sources of the impulse.

It is crucial to carefully preprocess the data and eliminate any potential outliers to lessen the impact of the impulsive component. It may also be necessary to use robust statistical methods, such as median filtering, to smooth the data and reduce the impact of the impulse factor **[85]**. It is defined as follows **[81]:**

$$\boldsymbol{IMF} = \frac{\boldsymbol{max}|\boldsymbol{x}(\boldsymbol{n})|}{\frac{1}{N}\sum_{i=1}^{N}|x_i|} \quad \textbf{(12)}$$

In brief, because it can significantly affect the accuracy of the diagnosis, the impulse factor is a crucial aspect to take into account when diagnosing bearing faults. It is conceivable to lessen the impact of the impulse component and boost the diagnostic precision by thoroughly preprocessing the data and utilising strong statistical procedures **[86]**.

II.3.5. Skewness:

In ML, Skewness describes the form of the target variable's dataset distribution. The performance of ML models can be significantly impacted if the target variable is skewed or biased in one way **[87]**. For instance, the target variable is assumed to have a normal distribution by several common ML techniques, including linear regression and decision trees. If the target variable is heavily skewed, these algorithms may produce biased results. This is because the mean, which is used as a key metric in these algorithms, can be influenced by extreme values and outliers in the skewed distribution **[88]**.

In the field of bearing fault diagnosis, skewness is often used as a statistical feature for pattern recognition and classification tasks. Bearings are a crucial component in many mechanical systems, and their failure can result in significant downtime and repair costs. The vibration signals generated by faulty bearings can contain valuable information about the fault type and severity **[89]**.

Skewness, along with other statistical features such as kurtosis and standard deviation, is commonly extracted from the vibration signals and used as inputs to ML algorithms **[90]**. The skewness feature captures the asymmetry of the vibration signal and can help distinguish between different types of faults, such as inner race faults, outer race faults, and rolling element faults. It is defined as follows **[91]:**

$$SK = \frac{\frac{1}{N}\sum_{i=1}^{N}(x(i) - \overline{x})^3}{\left(\sqrt{\frac{1}{N}\sum_{i=1}^{N}(x(i) - \overline{x})^2}\right)^3} \quad \textbf{(13)}$$

In conclusion, skewness plays an important role in the early detection of bearing faults, helping to ensure the reliable operation of mechanical systems and reducing the costs associated with downtime and repairs.

II.3.6. The Standard Deviation:

In statistics, the standard deviation is a measure of the amount of variation or dispersion of a set of values. It is a measure of how much the data deviates from the mean. A low standard deviation indicates that the values tend to be close to the mean (also called the expected value) of the set, while a high standard deviation indicates that the values are spread out over a wider range **[92]**.

The standard deviation is calculated as the square root of variance **[92]**. The formula for variance is:

$$\sigma^2 = \frac{\sum_{i=1}^{N}(x(i) - \bar{x})^2}{N - 1} \quad \textbf{(14)}$$

Where:

σ^2: is variance.

x_i is each value in the dataset,

$\bar{x}$ is the mean of all values in the dataset

N is the number of values in the dataset.

The formula uses $N-1$ in the denominator instead of N when calculating the standard deviation for a sample. This adjustment is known as Bessel's correction and helps to provide an unbiased estimate of the population standard deviation based on a sample.

Standard deviation is critical in machine learning for understanding data distribution and analysing the variety of features within a dataset **[93]**. It is a statistical metric that provides insights into the spread and dispersion of data points, which may be useful for a variety of machine learning activities such as data preparation, model validation, and outlier detection. In addition, the standard deviation can be used to measure confidence in a model's statistical conclusions **[94]**.

II.4. Frequency domain features:

Feature extraction in the frequency domain refers to the process of extracting relevant information from signals or data by transforming them into the frequency domain. Techniques like Fast Fourier Transform (FFT) and Short-Time Fourier Transform (STFT) are frequently used to accomplish this **[95]**.

Frequency domain features can be used in various applications, including speech and audio processing, image processing, and signal analysis. These features can provide insight into the spectral content of signals and can be used to identify patterns, extract features, and classify signals. Examples of frequency domain features include spectral power, spectral centroid, spectral roll-off, and spectral flatness **[96]**.

Types of Feature extraction in the frequency domain:

II.4.1. The Power Spectral Density (PSD):

PSD is a commonly used method in the field of vibration analysis and bearing fault diagnosis. The PSD provides information about the distribution of power in a signal over different frequencies, which can be used to identify the presence of a fault and diagnose its type. In bearing fault diagnosis, the PSD is used to analyze the vibration signals generated by rotating machinery **[97]**. By comparing the PSD of a healthy bearing with the PSD of a faulty bearing, it is possible to identify the presence of a fault and determine its type. To perform a PSD analysis, the vibration signals are first transformed into the frequency domain using techniques such as the Fast Fourier Transform (FFT). The resulting spectra are then used to calculate the PSD, which provides information about the distribution of power in the signal over different frequencies **[98].**

In conclusion, the Power Spectral Density is a useful tool for bearing fault diagnosis, as it provides valuable information about the distribution of power in the vibration signals generated by rotating machinery. By analyzing the PSD, it is possible to identify the presence of a fault and diagnose its type, which can then be used to make informed decisions about maintenance and repair **[99]**. Which is defined as follows **[100]**:

$$S(f) = \int_{-\infty}^{+\infty} R_x(t)\, e^{-j2\pi ft} dt \qquad (15)$$

Where :
S(f): is Power Spectral Density.
$R_x(t)$: is the autocorrelation function of a random process ***X(t)***.

II.4.2. Spectral Centroid: The spectral centroid is a commonly used feature in the diagnosis of bearing faults. In essence, it is a measure of the centre of gravity of the frequency spectrum of a vibration signal and can provide information about the distribution of energy across different frequencies **[101]**.

In healthy bearings, the vibration signals typically have energy across a range of frequencies, with the majority of the energy concentrated in the low-frequency range. If a bearing fault occurs, it can cause changes in the vibration signals, such as increased energy at higher frequencies or a shift in the spectral centroid towards higher frequencies. These changes can be used to identify the presence of a fault and diagnose the type of fault that has occurred **[102]**.

In conclusion, the spectral centroid is a useful feature for the diagnosis of bearing faults, as it provides information about the distribution of energy across different frequencies in the vibration signals from a bearing. This information can be used to identify changes that are indicative of a fault and to diagnose the type of fault that has occurred. The spectral centroid is calculated as described in **[103]**:

$$SC = \frac{\sum_{k=b_1}^{b_2} f_k s_k}{\sum_{k=b_1}^{b_2} s_k} \quad \textbf{(16)}$$

Where:

f_k is the frequency in **Hz** corresponding to bin k.

s_k is the spectral value at bin k.

b_1 and b_2 are the band edges, in bins, over which to calculate the spectral centroid.

II.4.3. Entropy: is a statistical measure that provides information about the randomness or disorder of a signal in the domain of bearing fault diagnosis. It is commonly used in this context to extract features from vibration signals that can be used to diagnose bearing faults **[104]**.

A signal's entropy can be calculated using a variety of algorithms, including the Shannon entropy **[104]**, sample entropy **[105]**, and permutation entropy **[106]**. These algorithms process the vibration signal in different ways, but the main idea is to quantify the amount of uncertainty or randomness in the signal. For example, a healthy bearing will typically produce a vibration signal with low entropy, while a faulty bearing may produce a signal with higher entropy due to increased randomness or disorder in the vibration pattern **[107]**. By computing the entropy of a vibration signal, it is possible to extract features that can be used as input to machine learning (ML) algorithms for fault diagnosis. For example, the entropy of a signal could be

used as a feature in a decision tree or a support vector machine to classify different types of bearing faults **[108]**.

It is important to note that entropy is just one of many possible features that can be used for bearing fault diagnosis and that the choice of feature extraction method will depend on the specific application and the characteristics of the vibration signal. In general, it is recommended to use a combination of different features and algorithms for robust and accurate fault diagnosis **[109]**. Which is defined as follows **[110]**:

$$\boldsymbol{Ent = -\sum_{i=1}^{N} x_i \log_2(x_i)} \quad \textbf{(17)}$$

Where:
$\boldsymbol{x_i}$: The frequentist probability of an element/class 'i'
$\boldsymbol{N}$: The total number of classes.

II.4.4. Wavelet decomposition: is a signal processing technique that divides a signal into frequency sub-bands. The wavelet decomposition can be used to extract information from vibration signals collected from a faulty bearing in the aspect of bearing fault diagnosis. The vibration signals contain rich information about the bearing condition, including frequency content, time domain behaviour, and non-stationary patterns **[111]**. By decomposing the vibration signals into multiple frequency sub-bands using wavelet transforms, it is possible to isolate and analyze the fault-related information and detect faults in the early stages before they lead to catastrophic failure.

The wavelet transform provides a multi-resolution representation of the vibration signal, which allows the identification of both low- and high-frequency fault features. For example, the presence of high-frequency spikes in the wavelet coefficients can indicate the presence of surface faults, such as cracks or pitting, while low-frequency coefficients can reveal sub-surface faults, such as inner race or ball defects **[112]**.

In conclusion, wavelet decomposition is a powerful tool for bearing fault diagnosis and is widely used in industry and academia. It provides a flexible and effective approach to analyzing vibration signals and detecting faults in rotating machinery.

Continuous Wavelet Transforms (CWT) **[113]**:

$$\boldsymbol{T(a,b) = \frac{1}{\sqrt{a}} \int_{-\infty}^{+\infty} x(t) \Psi^* \left(\frac{t-b}{b} \right) dt} \quad \textbf{(18)}$$

Discrete Wavelet Transforms (DWT) **[114]**:

$$T_{m,n} = \int_{-\infty}^{+\infty} x(t)\Psi_{m,n}(t)dt \quad \textbf{(19)}$$

Where:
Ψ(t): is the mother wavelet.
x(t): a finite energy signal.
a: sets the scale of the wavelet.
b: defines the location of the wavelet.

II.4.5. Power Spectrum Ratio: The Power Spectrum Ratio (PSR) is a commonly used method for diagnosing bearing faults in rotating machinery. The PSR compares the power spectral density of two signals, one from a healthy bearing and one from a faulty bearing. By analyzing the differences in the spectral content of these signals, the PSR can identify the presence of a fault and its type **[115]**. To perform a PSR analysis, the vibration signals from the healthy and faulty bearings are first transformed into the frequency domain using techniques such as the Fast Fourier Transform (FFT). The resulting spectra are then used to calculate the PSR by dividing the spectrum of the faulty bearing by the spectrum of the healthy bearing. A significant increase in the amplitude of certain frequency components in the PSR may indicate the presence of an inner race fault or an outer race fault **[116]**. A decrease in the amplitude of certain frequency components in the PSR may indicate the presence of a rolling element fault. In general, the PSR is a useful tool for bearing fault diagnosis because it provides valuable information about the spectral content of the vibration signals. By analyzing the PSR, one can quickly identify the presence of a fault and diagnose its type, which can then be used to make informed decisions about maintenance and repair **[117]**. Which is defined as follows **[118]**:

$$PSR = \frac{p_0}{p} = \frac{\int_{f_0-n}^{f_0+n} p(f)df}{\int_{-\infty}^{+\infty} p(f)df} \quad \textbf{(20)}$$

Where :
PSR: is the power spectrum ratio,
p(f): is the power spectrum density function,
$\mathbf{f_0}$: is the frequency corresponding to the maximum power spectrum,
n: is the integral range.

II.5. Dimensionality Reduction:

Dimensionality reduction is a method for lowering the number of variables in a dataset while maintaining as much information as possible. It is commonly used in ML and pattern recognition. In the aspect of bearing fault diagnosis, it is used to simplify the high-dimensional vibration signals typically obtained from sensors installed on rotating machinery **[119]**. The goal is to convert complex and noisy signals into a more compact and interpretable representation that can be used for further analysis and diagnosis.

Several popular dimensionality reduction techniques are commonly used in bearing fault diagnosis, including:

II.5.1. Principal Component Analysis (PCA): is one of the techniques for reducing dimensionality that is popularly used in data analysis and ML. The goal of PCA is to transform a collection of potentially correlated observations into a collection of linearly uncorrelated variables called principal components. The first principal component has the greatest variance and captures the majority of the information in the original set of variables. The second principal component captures additional information and has the second highest variance. It is orthogonal to the first principal component. This procedure is repeated until all of the principal components have been calculated **[120]**.

PCA is useful in a variety of applications, including image compression, noise reduction, feature extraction, high-dimensional data visualization, and others.

The steps for performing PCA are as follows **[121]**:

Data Pre-processing: Ensure that the data is cleaned and in the correct format for PCA analysis. This may include dealing with missing values, transforming variables if necessary, and scaling the variables.

Data Matrix Preparation: Create a data matrix $\mathbf{X} = \{x_1, x_2, \dots, x_m\}$ from the pre-processed data, where each row represents an observation, and each column represents a variable.

Mean Centering: Subtract the mean of each variable from its respective values to obtain the mean-centred data.

$$\boldsymbol{x_{new} = x - \mu} \qquad \textbf{(21)}$$

Normalize the data: Normalize the data before achieving PCA. This will guarantee that each feature has a mean = 0 and variance = 1

$$z = \frac{x - \mu}{\sigma} \tag{22}$$

Where **σ**and**μ**are the Mean and standard deviation, respectively.

Covariance Matrix Calculation: Calculate the covariance matrix of the mean-centred data, which is a measure of the variability of the variables and their relationships.

$$Cov(X, Y) = \frac{1}{m}\sum_{i=1}^{m}(x_i - \sigma_x)(y_i - \sigma_y) \tag{23}$$

Eigenvalue and Eigenvector Calculation: Calculate the eigenvalues and eigenvectors of the covariance matrix, which represent the magnitude and direction of the principal components, respectively.

Principal Component Sorting: Sort the eigenvalues in descending order and select the top k eigenvectors, where k is the number of dimensions desired in the transformed data.

Transformation Matrix Calculation: Calculate the transformation matrix as the matrix of the selected eigenvectors.

Data Transformation: Transform the original data into the new, lower-dimensional space by multiplying the mean-centred data matrix by the transformation matrix.

Interpretation: Interpret the results of the PCA analysis, including the magnitude and direction of each principal component, the explained variance, and the transformed data.

The fact that PCA is a linear technique and is sensitive to the scale of the features are just two examples of its drawbacks. The complexity of huge datasets can still be reduced with this method, especially when paired with other dimensionality reduction methods.

II.5.2. Independent Component Analysis (ICA): Using this statistical technique, one can disentangle a multivariate signal into independent, non-Gaussian components; Finding a linear representation of the data in which each component is as independent of the others as possible is the basic goal of ICA. When trying to separate signals that are combined or overlaid in some way, such as in the picture and audio processing, this might be helpful **[122]**.

The steps involved in ICA, along with their mathematical formulation, are as follows **[123]**:

Whitening: The first step is to pre-process the data by converting it into a white noise signal, which has zero, mean and unit covariance.

Decorrelation: In this step, the decorrelation matrix is applied to the whitened data to make it statistically independent. The decorrelation matrix is computed using eigenvalue decomposition or singular value decomposition.

Centring: The de-correlated data is then centred to have zero mean, which helps in finding the underlying independent components.

Immixing: In this step, the immixing matrix is estimated that separates the independent components from the mixture of signals. This can be done using various algorithms such as FastICA.

Demixing: The final step is to apply the immixing matrix to the centred data to obtain the estimated independent components.

Note that the accuracy of the results obtained from ICA depends on various factors such as the number of independent components, the distribution of the signals, and the choice of the algorithm used for immixing **[124]**.

II.5.3. Linear Discriminant Analysis (LDA): is a method for reducing dimensionality that is frequently applied in the fields of ML and pattern recognition; A high-dimensional dataset can be converted into a lower-dimensional space using LDA while preserving the data's most important details; LDA aims to optimize the separation between various classes in the data by projecting the data onto a lower-dimensional space **[125]**.

Algorithm 1 LDA Algorithm [126]

Require Data matrix $X \in R^{d\times N}, x_i \in R^{d\times 1}$ is the i-th column of X. Lavel vector $y_i \in \{1, 2, \dots, C\}, i = 1, \dots, N, N_c, c = 1, \dots, C$ is the amount of samples in each class.

Ensure The projection matrix $P \in R^{p\times d}$

1: Compute the mean vector for each class : $m_c = \frac{\sum_{i=1}^{N_c} x_i}{N_c} \in R^{d\times l}$

2: Compute total mean vector: $m = \frac{\sum_{i=1}^{N} x_i}{N} \in R^{d\times l}$

3: Compute within-class scatter: $S_w = \sum_{c=1}^{C}\sum_{j=1}^{N_c}(x_j - m_c)(x_j - m_c)^T \in R^{d\times d}$

4: Compute between-class scatter: $S_b = \sum_{c=1}^{C} N_c(m_c - m)(m_c - m)^T \in R^{d\times d}$

5: Eigen-decomposition: $[V, D] = eig(S_w^{-1}S_b)$

6: Getting the projection matrix: The matrix P is composed of the top-p eigenvectors corresponding to the largest eigenvalues.

In conclusion, dimensionality reduction using LDA is a powerful technique for reducing the complexity of the data while preserving its information content. It is widely used in pattern recognition, ML, and data mining and is a valuable tool for solving real-world classification problems. It can improve the performance of classification algorithms by reducing the noise in the data and making the data more separable. It can also reduce the computational cost of subsequent algorithms, which is particularly important for large datasets.

II.5.3. Autoencoder: By encoding high-dimensional input into a lower-dimensional representation (latent space) and then decoding that representation back into a near approximation of the original high-dimensional data, autoencoders are a form of neural network that can be used to reduce dimensionality. The objective is to develop a concise representation of the data that captures the most crucial relationships and features while eliminating extraneous or redundant material **[127]**.

The encoder and decoder are the two parts that make up an autoencoder. The input data is converted by the encoder into a lower-dimensional latent space, and the latent representation is then converted by the decoder back into the original high-dimensional space. The goal is to minimise the reconstruction error or the difference between the original and reconstructed data. The weights of the encoder and decoder are adjusted during training to minimise reconstruction error **[128]**.

There are, basically, 7 types of autoencoders **[129]**:

II.5.4. Denoising autoencoder(DAE): this is a type of deep learning architecture that is used to remove noise from data; It is an extension of the traditional autoencoder, which is a neural network architecture used for unsupervised learning.

II.5.5. Sparse Autoencoder: a sparse autoencoder is a type of deep learning architecture that learns a compact, low-dimensional representation of the data while encouraging sparsity in the activations of the hidden layer; This results in a sparse, efficient representation of the data, which can be useful for various applications.

II.5.6. Deep Autoencoder: a deep autoencoder is a type of deep learning architecture that learns a compact, low-dimensional representation of the input data by stacking multiple layers of neurons in the encoding and decoding parts of the network; By doing so, the network can learn more complex representations of the data, which can be useful for various applications.

II.5.7. Contractive Autoencoder: a contractive autoencoder is a type of deep learning architecture that adds a regularization term to the reconstruction loss to enforce a contractive property, where small changes in the input data result in small changes in the hidden representations; By doing so, the network can learn more robust and useful hidden representations, which can be useful for various applications.

II.5.8. Under-complete Autoencoder: an under-complete autoencoder is a type of autoencoder where the number of neurons in the hidden layer is smaller than the number of neurons in the input layer; By forcing the network to learn a compressed representation of the data, the under-complete autoencoder can capture the essential features of the data, which can be useful for various applications.

II.5.9. Variational Autoencoder: a Variational Autoencoder (VAE) is a type of deep learning architecture that combines the idea of an autoencoder with probabilistic modelling. The goal

of a VAE is to learn a compact, low-dimensional representation of the data while also being able to generate new samples from that representation; VAEs have many applications, including generative modelling, unsupervised representation learning, and anomaly detection.

II.5.10. Convolutional Autoencoder: a convolutional autoencoder is a type of deep learning architecture that uses convolutional layers in the encoding and decoding parts of the network to process image data; By doing so, the network can learn more complex representations of the data, which can be useful for various applications, especially for image data.

Autoencoders have been applied to a wide range of applications, including image and audio compression, anomaly detection, and feature learning. They are also a popular tool for unsupervised learning, as they can learn useful representations of data without the need for labelled data.

II.6. Feature Selection:

The process of selecting a subset of relevant features, or characteristics, from a larger set of features, to improve the performance of a ML model is known as feature selection. The goal of feature selection is to reduce data dimensionality by removing irrelevant, redundant, or noisy features. Feature selection can also improve model interpretability and comprehension **[130]**.

Feature selection is an important step in the process of bearing fault diagnosis. It involves selecting a subset of the available features, or characteristics, of the bearing data that are most relevant for the task of detecting and diagnosing faults. The goal of feature selection is to reduce the dimensionality of the data and improve the accuracy, efficiency, and interpretability of the diagnosis system **[131]**.

Several feature selection methods can be used, including **[132]**:

II.6.1. Filter methods: Filter methods evaluate each feature individually based on a statistical measure, such as correlation or mutual information, with the target variable. The features with the highest scores are selected.

II.6.2. Wrapper methods: Wrapper methods evaluate the performance of the ML model trained on subsets of the features. The features that result in the best performance are selected.

II.6.3. Embedded methods: Embedded methods combine feature selection and model training into a single process. For example, regularization in linear models can be used to reduce the magnitude of the coefficients and select important features.

II.6.4. Hybrid methods: Hybrid methods combine multiple feature selection methods to get the best results.

It is vital to note that the approach used to choose features will be determined by the individual problem and data, and the results may differ based on the method utilised. Furthermore, feature selection is frequently an iterative process in which different methods are tried and the results are compared to discover the best collection of features **[133]**.

Dimensionality reduction and feature selection are two distinct strategies for preprocessing and optimising ML models' performance. They do, however, have different purposes and approaches. In conclusion, feature selection prioritises the most relevant and useful aspects, whereas dimensionality reduction prioritises reducing the number of features while maintaining the most critical information. Both strategies can be used in tandem to increase a machine-learning model's performance **[134]**.

II.7. Binary Optimization algorithms:

Binary optimization is a type of mathematical optimization problem where the variables can only have binary values, i.e., either 0 or 1; These kinds of problems are frequently seen in decision-making situations, such as those in which the objective is to select the best combination of items from a list of possible options **[135]**.
There are several algorithms for solving binary optimization problems, including:

II.7.1. Binary Bat Algorithm (BBA): is a metaheuristic optimization algorithm inspired by bat echolocation. It is a population-based algorithm for solving complex optimization problems. The BBA is a relatively new algorithm that is effective in solving a variety of optimization problems in recent years **[136]**.

The BBA is based on bat echolocation behaviour. Echolocation is used by bats to navigate and find food in their environment. They emit sound waves and then listen for the echoes that are reflected from objects in their environment. By analyzing the echoes, they can determine the location and size of objects in their environment. The BBA uses this same principle to solve optimization problems **[137]**.

The BBA works by creating a population of solutions to the optimization problem. Each solution is represented by a bat. The bats then emit sound waves and listen for the echoes that are reflected from the objective function. The bats then adjust their solutions based on the echoes they receive. This process is repeated until the bats find a solution that is close to the optimal solution **[138]**.

The BBA outperforms other optimization techniques in various ways. It is straightforward to build and can be used to address a wide range of optimization problems. It is also quite quick and can uncover solutions in a relatively short period. Furthermore, the BBA can find solutions that are near the optimal answer **[139]**.

Overall, the Binary Bat Algorithm is a powerful and efficient feature selection algorithm inspired by bat echolocation behaviour that can be used to identify the most important features in a dataset. It is computationally efficient, able to identify non-linear relationships between features, and can be used to identify the most important features in a dataset. Furthermore, it is quite fast and can identify solutions that are close to the idea **[140]**.

Each artificial bat in the binary bat algorithm contains a position vector X_i, a velocity vector V_i, which are the i^{th} bat individual in the t^{th} iteration, as well as a variable loudness A and a frequency vector F_i that can be modified throughout redundancy. Each bat moves in the manner described below:

$$V_i(t+1) = V_i + (X_i(t) - G_{best}) \times F_i \quad (24)$$

$$X_i(t+1) = X_i(t) + V_i(t+1) \quad (25)$$

$$F_i = F_{min} + (F_{max} - F_{min}) \times \beta \quad (26)$$

Where:

- **G_{best}: represents the current optimal solution.**
- **β: denotes is uniformly distributed between [0, 1].**
- ***max*: stands for the iteration maximum number.**
- ***rand*: represents a uniform random number between 0 and 1.**
- ***minfit*: signifies the minimum fitness value.**

Algorithm 2: The binary bat algorithm **[141]**

Initialize pulse rates r_i and loudness A_i
Initialize the bat population x_i ($i = 1, 2, \ldots, n$) and v_i
Define pulse frequency f_i and x_i
While ($t < max$) **do**
Adjust frequency and velocity and Calculate the transfer function
If ($rand > r_i$) **then**
Select a Solution G_{best} among the best solutions randomly.
Adjust some of the dimensions of the position vector with some of the dimensions of G_{best}
End if
$Fitness1 = f(x_i)$
$Fitness2 = f(G_{best})$
If ($fitness1 > fitness2$ & $rand < A_i$) **then**
Update initial bat and reduce loudness, increase pulse rate
End if
If ($fitness2 < min_{fit}$) **then**
Update G_{best}
Reduce loudness A_i, increase pulse rate r_i.
End if
Classify the bats and locate the current G_{best}
End while

II.7.2. Binary particle swarm optimisation: Binary Particle Swarm Optimization (BPSO) is an evolutionary computation technique used for optimization problems and feature selection. It is a hybrid meta-heuristic combining features of the social and cognitive behaviour of particles in a swarm with binary search techniques. This technique enables the optimization of a problem by using a population of particles that move in the search space. Each particle is associated with a binary vector, with each element in the vector representing a feature. The particles move in the search space and interact with each other as they try to find a globally optimum solution. BPSO has several advantages over other feature selection algorithms, such as its ability to avoid local optima, its ability to converge quickly, and its low computational complexity. Additionally, BPSO is suitable for large-scale feature selection problems, as its computational complexity does not increase with the number of features **[142]**. Thus, BPSO offers a useful and efficient tool for feature selection and provides an alternative to other, more traditional methods.

Binary Particle Swarm Optimization (BPSO) is a promising optimization algorithm for feature selection due to its ability to efficiently search for optimal subsets of features **[143]**. Unfortunately, there are a few obstacles to its adoption. To begin, BPSO necessitates a significant number of iterations to achieve an ideal subset of features, which raises the computational cost of the process. Second, it can quickly become caught in local optima, producing a suboptimal subset of features. This can be mitigated by raising the number of rounds, however, this increases the computing cost even further. Finally, the BPSO solution's accuracy is typically sensitive to the algorithm's parameter values, making it challenging to optimise the parameters. Finally, it may struggle to pick features in high-dimensional datasets because of the curse of dimensionality, which arises when the number of features exceeds the number of samples. Despite these challenges, it remains a promising feature selection optimization method due to its ability to efficiently search for the optimal subset of features **[144]**.

Due to its potential use in feature selection, the technique known as Binary Particle Swarm Optimization (BPSO) is growing in favour lately. The potential of BPSO was examined by AD Li, B Xue, and M Zhang **[145]** in their article that was published in Applied Soft Computing. They specifically ran tests to demonstrate how well the BPSO algorithm selected critical features from a dataset. They also evaluated how well the BPSO method performed in comparison to other algorithms like the Support Vector Machine and the Genetic Algorithm.

The outcomes demonstrated that the BPSO method was more precise and effective in choosing crucial features for a dataset. Furthermore, the authors discovered that the BPSO algorithm could identify the most relevant features in a dataset in fewer rounds, saving time and resources. Additionally, the authors proposed that BPSO be employed in a variety of applications, including data mining, decision-making, and ML. This study exemplifies how BPSO can be utilised in feature selection and its potential applications in a variety of sectors.

In conclusion, Binary Particle Swarm Optimization (BPSO) is a feature selection algorithm that efficiently replaces exhaustive search techniques by searching for an optimal subset of features. BPSO uses the fundamental ideas of swarm intelligence, which can combine stochastic and deterministic methods to optimise. This method can be used to implement additional solutions in various fields and is superior to genetic algorithms for feature selection processes **[146]**.

The BPSO only accepts binary values (0 or 1), and each particle progresses through the problem space by following the most recent optimal particles $\boldsymbol{X_i}$ **[147]**. Given that there are $\boldsymbol{N}$ particles in the particle swarm and that each has a diameter of $\boldsymbol{D}$, the particles are represented by:

$$\boldsymbol{X_i} = \left(\boldsymbol{x_i^1}, \boldsymbol{x_i^2}, \boldsymbol{x_i^3}, \dots, \boldsymbol{x_i^d}, \dots, \boldsymbol{x_i^D}\right) \quad \textbf{(27)}$$

$$\boldsymbol{V_i} = \left(\boldsymbol{v_i^1}, \boldsymbol{v_i^2}, \boldsymbol{v_i^3}, \dots, \boldsymbol{v_i^d}, \dots, \boldsymbol{v_i^D}\right) \quad \textbf{(28)}$$

for $i = 1, 2, 3, \dots, N$ and $d = 1,2, \dots, D$

The velocities and positions of particles are randomly initialized and updated as follows **[148]**:

$$\boldsymbol{v_i^d(t+1) = w(t) * v_i^d(t) + c_1 * r_1 * (pbest_i^d(t) - x_i^d(t)) + c_2 * r_2 * (gbest^d(t) - x_i^d(t))} \quad \textbf{(29)}$$

$$\boldsymbol{x_i^d(t+1)} = \begin{cases} \boldsymbol{1}, & \boldsymbol{if\ S\left(v_i^d(t+1)\right) > r_3} \\ \boldsymbol{0}, & \boldsymbol{otherwise} \end{cases} \quad \textbf{(30)}$$

$$\boldsymbol{S\left(v_i^d(t+1)\right) = \frac{1}{1 + e^{-v_i^d(t+1)}}} \quad \textbf{(31)}$$

$$\boldsymbol{w = w_{max} - (w_{max} - w_{min})\left(\frac{t}{T_{max}}\right)} \quad \textbf{(32)}$$

Where:

$\boldsymbol{r_1}$ and $\boldsymbol{r_2}$ are two independent random vectors in **[**0,1].

$\boldsymbol{r_3}$ is a random number distributed between 0 and 1.

$\boldsymbol{w}$ is the inertia weight.

c_1 and c_2 are acceleration coefficients.

$pbest_i^d(t)$ is the personal best position of the ith particle in the d^{th} dimension.

$gbest^d(t)$ is the best position in the d^{th} dimension found so far by a swarm.

i is the order; d is the dimension of search space; and t is the number of iterations.

$S\left(v_i^d(t+1)\right)$ is the velocity transformed into the probability value using the sigmoid function.

Algorithm 3: Binary Particle Swarm Optimization **[148]**

Begin: initialize the parameters, N, T_{max}, c_1, c_2, v_{max}, v_{min}
Initialize a population of particles, X
Evaluate the fitness of particles $f(x_i^d(t))$
Set ***pbest*** and ***gbest***
for $t = 1$ to the maximum number of iterations T_{max}
Compute the inertia weight w as in Equation **(32)**
for $i = 1$ to the number of particles N
for $d = 1$ to the number of dimensions D
Update the velocity of the particle, $v_i^d(t+1)$ using Equation **(29)**
Convert the velocity into probability value as in Equation **(31)**
Update the position of a particle, $x_i^d(t+1)$ using Equation **(30)**
next d
Evaluate the fitness of new particles, $f(x_i^d(t+1))$
next i
Update the ***pbest*** and ***gbest***
next t
End

II.7.3. Binary grey wolf optimisation: Binary Grey Wolf Optimization (BGWO) is a powerful feature selection technique for bearing fault diagnosis. It is a meta-heuristic optimization algorithm based on the behaviour of grey wolves in nature **[149]**. The algorithm is inspired by the cooperative hunting behaviour of grey wolves, which is characterized by a leader, a follower, and a scout. The leader and follower wolves search for prey in a certain area, while the scout wolf searches for a better area with more prey **[150]**.

The wisdom of leadership and hunting behaviours of grey wolves in nature serve as the primary inspiration for GWO. Wolves often live in packs with a rigid hierarchy; the leader of the pack is known as the alpha (α) and is in charge of directing the entire pack in hunting, migration, and feeding. The second level of the hierarchy is occupied by a beta (β) wolf, who takes over as leader of the pack if it becomes ill or dies, and is followed by delta wolves (δ). The remaining members of the pack are known as omegas (ω) **[151]**.

The steps of BGWO in feature selection are as follows **[152]**:

1_Initialization: The first step in the BGWO process is to populate the population of solutions. This is accomplished by generating a set of binary strings at random that represent the dataset's features.

2_Fitness Evaluation: The following step is to assess the fitness of each solution. This is accomplished by computing the fitness value of each solution to the objective function.

3_Selection: The third step is to choose the best solutions from among the available options. This is accomplished by choosing solutions with the highest fitness values.

4_Mutation: The fourth step is to apply mutation to the solutions that have been chosen. This is accomplished by randomly flipping the bits of the chosen solutions.

5_Crossover: The fifth step is to cross-reference the mutated solutions. This is accomplished by randomly swapping bits between two solutions.

6_Local Search: The sixth step involves performing a local search on the mutated and crossed solutions. This is accomplished by selecting a solution at random and then looking for a better solution in its vicinity.

7_Update: The seventh stage is to inform the population about the new solutions. This is accomplished by replacing the previous solutions with new solutions.

8_Ending: The algorithm is finished when the closure requirements are met. If the maximum number of iterations is reached or the best solution has not changed for a specific number of iterations.

BGWO has been used successfully to diagnose bearing faults. Other feature selection techniques, such as genetic algorithms and particle swarm optimization, have been shown to outperform them **[153]**. BGWO is also computationally efficient, as finding the optimal feature subset requires fewer iterations. Furthermore, BGWO can identify the most important features for bearing fault diagnosis, which can improve the classifier's accuracy **[154]**.

Algorithm 4: Binary Grey Wolf Optimization (BGWO) **[155]**

Begin
Randomly initialize the population of grey wolves, $\boldsymbol{X}$
Initialize the parameters $\boldsymbol{a}$, $\boldsymbol{A}$ and $\boldsymbol{C}$
Evaluate the fitness of wolves, **F(X)**
Set $\boldsymbol{X_\alpha}$ = The position of the best wolf
$\boldsymbol{X_\beta}$ = The position of the second-best wolf
$\boldsymbol{X_\delta}$ = The position of the third-best wolf
for $\boldsymbol{t}$ **= 1** to the maximum number of iterations, $\boldsymbol{T}$
for i = 1 to the number of the wolfs, **N**
Compute $\boldsymbol{X_1}$, $\boldsymbol{X_2}$ & $\boldsymbol{X_3}$ using Eq. **(33)**, **(34)** and **(35)**.
Generate $\boldsymbol{X_{new}}$ by applying Eq. (**36**).
next $\boldsymbol{i}$
Evaluate the fitness of all grey wolves, **F(X_{new})**
Update the position of $\boldsymbol{\alpha}$, $\boldsymbol{\beta}$ and $\boldsymbol{\omega}$.
Update the parameters $\boldsymbol{a}$, $\boldsymbol{A}$ and $\boldsymbol{C}$.
next $\boldsymbol{t}$
End

Where $\mathbf{X_1}$, $\mathbf{X_2}$, and $\mathbf{X_3}$ are calculated as follows:

$$\boldsymbol{X_1 = |X_\alpha - A_1 * D_\alpha|} \quad \textbf{(33)}$$

$$\boldsymbol{X_2 = |X_\beta - A_2 * D_\beta|} \quad \textbf{(34)}$$

$$\boldsymbol{X_3 = |X_\delta - A_3 * D_\delta|} \quad \textbf{(35)}$$

$$\boldsymbol{X^d(t+1) = \begin{cases} 1, & S\left(\frac{X_1^d + X_2^d + X_3^d}{3}\right) > r_3 \\ 0, & otherwise \end{cases}} \quad \textbf{(36)}$$

$$\boldsymbol{S(x) = \frac{1}{1 + e^{(-10*(x-0.5))}}} \quad \textbf{(37)}$$

$$\boldsymbol{A = 2 * a * r_1 - a} \quad \textbf{(38)}$$

$$\boldsymbol{C = 2 * r_2} \quad \textbf{(39)}$$

$$\boldsymbol{D = \left| C * X_p(t) - X(t) \right|} \quad \textbf{(40)}$$

$$\boldsymbol{a = 2 - 2 * \left(\frac{t}{T} \right)} \quad \textbf{(41)}$$

Where :

- Alpha α: indicates the fittest solution,
- Beta β: represents the second fittest solution,
- Delta δ: usually takes the third fittest solutions,
- Omega ω: indicates the rest of the solutions,
- t indicates the current iteration,
- T is the maximum number of iterations,
- A and C are coefficient vectors,
- X_p is the position vector of the prey,
- X indicates the position vector of a grey wolf,
- r_1 and r_2 are two independent random numbers uniformly distributed between [0,1],
- a is the encircling coefficient, it is linearly decreasing, from 2 to 0,
- $S(x)$ is the sigmoid function.

Conclusion:

Research on bearing fault diagnosis has been conducted extensively in recent years, due to their importance in industrial applications and with the availability of advanced technologies, feature extraction and feature selection plays an important role in bearing fault diagnosis.

Feature extraction and selection techniques are essential in bearing fault diagnosis. These techniques are used to extract and select the most relevant features from the vibration signals collected from the bearing faults and use them in diagnosis.

Feature extraction techniques involve the extraction of features from raw signals, such as frequency, time domain and statistical parameters. The process of feature extraction contains two main components. The first is Signal pre-processing is a key step as it allows for the reduction of noise, as well as the removal of redundant data. The second is Feature extraction, on the other hand, involves using algorithms to extract features from the signal that are relevant to the fault diagnosis.

Feature selection techniques involve the selection of a subset of relevant features from the extracted features to improve the accuracy and reliability of the diagnosis. it is then performed to select the most relevant features and discard those that are irrelevant. This process is complicated by the fact that there is no universal set of features that are relevant to all fault diagnoses. Each fault diagnosis requires its own set of features that are relevant to that specific diagnosis.

Feature extraction and selection is a crucial step in bearing fault diagnosis. By extracting suitable characteristics from the data and selecting meaningful ones, the accuracy of bearing fault diagnosis is maximized. In conclusion, feature extraction and selection approaches are effective and important for bearing fault diagnosis.

CHAPTER III

FAULT CLASSIFICATION AND DIAGNOSIS USING MACHINE LEARNING

III.1. Introduction:

In current years, machine learning techniques have been widely utilised for bearing fault diagnosis. These methods analyse data from sensors placed on bearings using a variety of machine-learning algorithms to detect and diagnose faults **[156]**. The sensor data can be used to train machine learning algorithms to recognise patterns in data that indicate a bearing fault.

Supervised learning algorithms such as support vector machines (SVMs), random forests, and artificial neural networks are the most commonly used machine learning algorithms for bearing fault diagnosis. These algorithms are trained using labelled data, which has been labelled with the appropriate fault type **[157]**. The tagged data for training the algorithm to identify data patterns that indicate a bearing fault. The algorithm, once trained, can detect and diagnose bearing faults in new data.

Lastly, using machine learning techniques to detect and diagnose bearing faults is an effective and efficient way to detect and diagnose bearing faults. It is quicker and more precise than classic methods, and it can be used to detect and diagnose faults in non-visible bearings. This makes it an attractive choice for multiple industries that rely on bearings to function **[158]**.

III.2. Artificial Neural Networks:

III.2.1. Briefs information on Artificial Neural Networks (ANN):

ANN is one of the machine-learning algorithms that is based on the biological neural networks of a human brain. The human brain consists of several neurons. These neurons are connected in various layers of the networks known as nodes in ANN. Each node has a specific function and is responsible for processing a specific piece of data **[159]**.

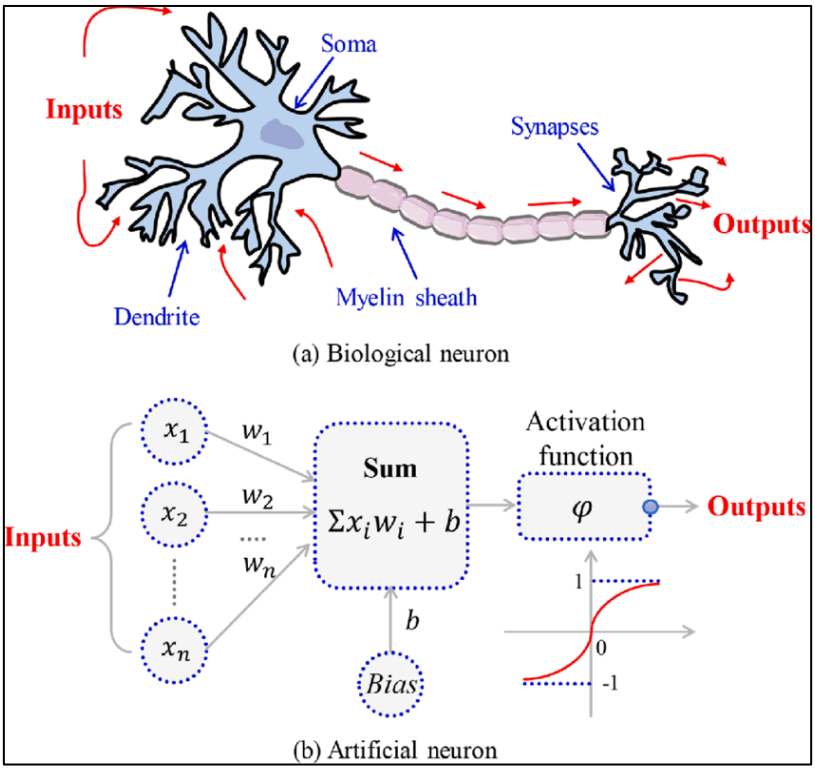

Figure III. 1 Biological and Artificial Neural Network [160].

ANN are computer algorithms that can be used to describe a system in terms of input-output relationships. They represent an alternative method of describing systems when analytical approaches are difficult or impossible to use. They have been used in a wide range of manufacturing applications **[161]**.

Generally, an artificial neural network is organised into three layers. These layers are made up of many interconnected nodes and each one of them has an activation function. The three layers of a neural network are as follows **[162]**:

Input Layer: As the name suggests, this layer receives the raw input data in several different formats provided by the programmer, such as images or text, and passes it on to the next layer

for processing. The number of input nodes in an input layer is ordinarily the same as the number of explanatory variables.

Hidden Layers: They are located between the input and output layers and are responsible for processing the input data by performing complex mathematical operations and the actual processing is done via a system of weighted 'connections', which are then added to produce a single number.

Output Layer: This layer produces the final output of the model, which could be a single value or a vector of values representing a classification or regression task.

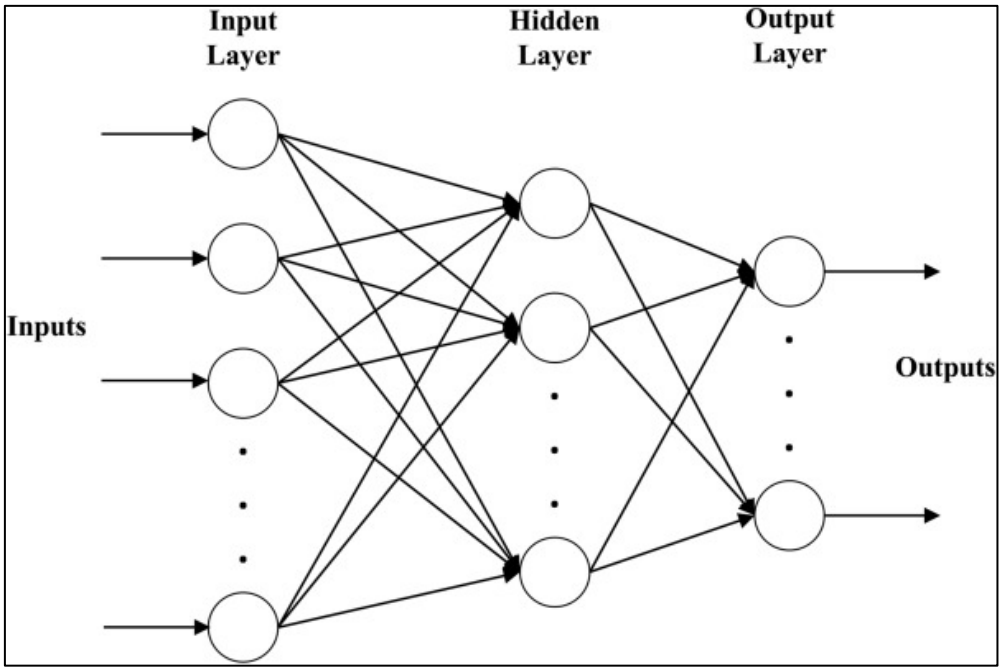

Figure III. 2 The architecture of an artificial neural network [163]

ANN have several different types such as feed-forward neural networks, recurrent neural networks, and convolution neural networks, the most common type is the feed-forward neural network and which consists of a series of layers, with each layer consisting of multiple neurons, in the training process, the weights of the connections between neurons are adjusted to minimise the difference between the predicted and actual output. In addition to the layers themselves **[164]**, ANN also have parameters that are learned during the training process. These include the weights and biases of the neurons, which are updated iteratively using optimization algorithms such as gradient descent.

III.2.2. Types of Artificial Neural Networks (ANN):

III.2.2.1. Feed-forward Neural Networks (ANN): A feed-forward artificial neural network, or FNN, only permits information to move in one direction, from the input layer via any potential hidden layers to the output layer. In a feed-forward neural network, the hidden layer, which comes after the input layer, connects every neuron to every other neuron in the input layer. There are connections between every neuron in the hidden layer, which may have one or more layers and every neuron in the layer above it. After receiving outputs from the concealed layer or layers, the output layer provides the final output [**165]**.

In order to apply a nonlinear activation function, each layer's neurons first quickly sum up the input data, which is commonly done by weighting the inputs. During training, the weights and biases of the neurons are learned via a technique called backpropagation. Feed-forward neural networks are capable of performing a wide range of tasks, such as speech recognition, natural language processing, picture recognition, and many more. One benefit of feed-forward neural networks is their capacity to learn intricate non-linear correlations between inputs and outputs. This ability makes feed-forward neural networks a powerful tool for solving problems that are difficult or impossible to solve using conventional techniques **[166]**.

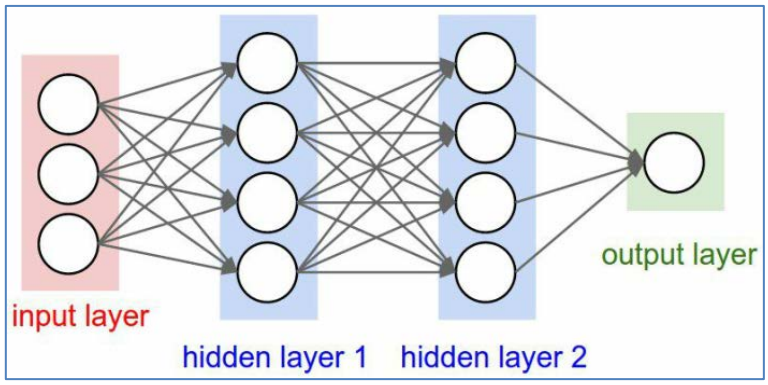

Figure III. 3 Feed-forward Neural Networks architecture [165].

III.2.2.2. Recurrent Neural Networks: An artificial neural network called a recurrent neural network (RNN) is made to interpret sequential data, such as time series or natural language. RNNs may accept variable-length sequences as input and produce output at each point in the sequence, in contrast to typical feedforward neural networks, which have a set number of input and output nodes **[167]**.

Recurrent connections, which enable the network to maintain a hidden state that can be updated at each time step based on the current input and the prior hidden state, are the core characteristic of RNNs. Then, using this hidden state, a prediction or output can be created for the current time step and the hidden state can be updated for the following time step **[168]**.

The vanishing gradient problem, which happens when the gradient of the loss function concerning the network parameters becomes very small, is one of the main issues with RNNs. As a result, training RNNs over lengthy periods may be challenging. The Long Short-Term Memory (LSTM) network, one of the most popular types of RNNs, employs specialised memory cells and gating mechanisms to enable the network to selectively recall or forget information across lengthy sequences **[169]**.

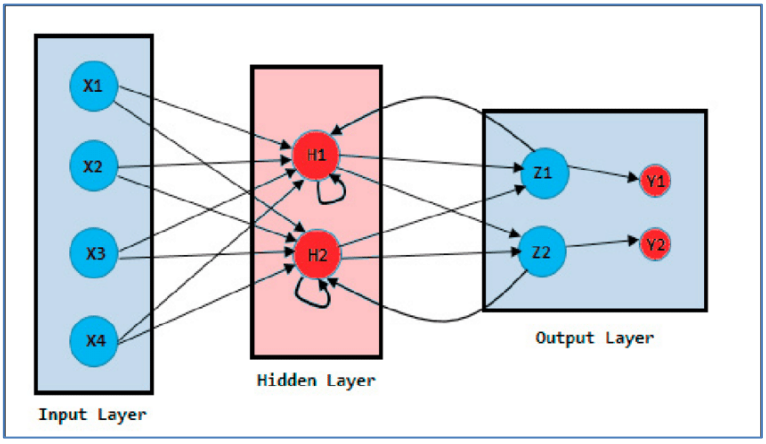

Figure III. 4 Recurrent Neural Networks architecture [167].

III.2.2.3. Convolutional Neural Networks (CNNs): Deep learning neural networks such as convolutional neural networks (CNNs) are extremely effective in analysing photos and videos. The capacity of CNNs to recognise spatial patterns in an image by using convolutional processes is their distinguishing feature **[170]**. The convolutional layer is the fundamental component of a CNN. It transforms a portion of the input picture or feature map into a new feature map that captures specific visual patterns or characteristics by applying a series of learnable filters. Pooling layers frequently come after these convolutional layers, downsampling the feature maps by averaging or picking the maximum value across small regions **[171]**.

Fully connected layers, which employ the characteristics extracted by the convolutional layers to generate predictions, are another option for CNNs. All of these layers' parameters are learned through backpropagation during training, where the network modifies its settings to reduce a

specific loss function, often depending on the discrepancy between the anticipated output and the true label **[172]**.

CNNs have become the preferred approach for performing numerous computer vision tasks, including picture classification, object recognition, and segmentation. They've also been used on other sorts of data, such as audio and text **[173]**.

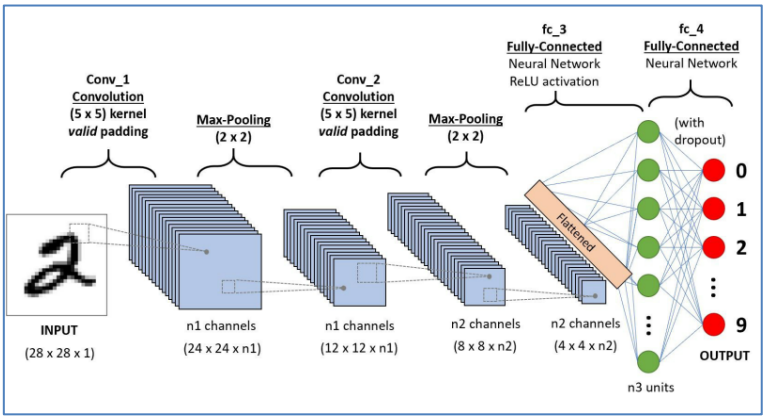

Figure III. 5 Convolutional Neural Networks architecture [170].

III.2.2.4. Self-Organizing Maps (SOMs): An artificial neural network (ANN) called a self-organizing map (SOM) employs unsupervised learning to produce a low-dimensional representation of high-dimensional data. Teuvo Kohonen, a Finnish mathematician, developed SOMs in the 1980s [**174]**.

A grid of nodes or neurons that each represent a prototype or codebook vector makes up the SOM algorithm. The weights of these neurons are changed during training such that they more closely resemble the input data vectors that are fed into the network. During training, neurons compete to react to particular input vectors while cooperating with other neurons to represent similar input vectors **[175]**.

SOMs can be applied to many different tasks, such as feature extraction, clustering, and data visualisation. They are especially helpful for examining and analysing high-dimensional data, such as gene expression or picture data, and for discovering undiscovered patterns and connections between the data **[176]**.

SOMs have been used in a variety of domains, including bioinformatics, image processing, natural language processing, data mining, and image processing. They are also employed in the creation of anomaly detection and recommendation systems **[177]**.

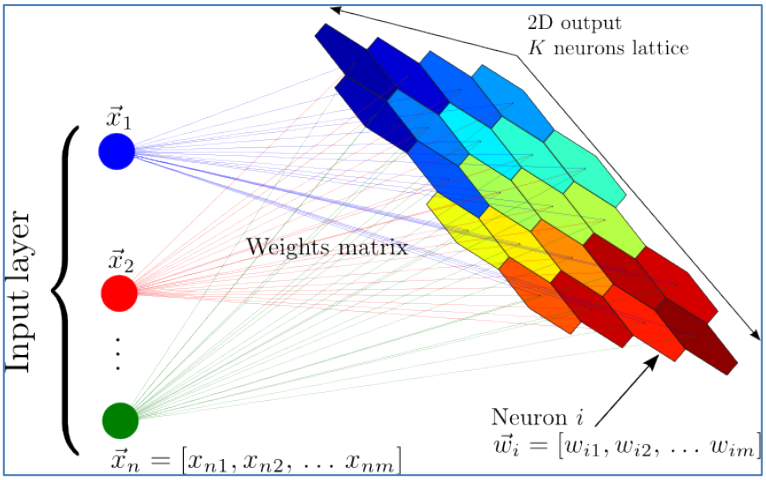

Figure III. 6 Self-Organizing Maps Architecture [178].

III.2.2.5. Radial Basis Function Networks (RBFNs): RBF networks, a form of artificial neural network, are frequently used for pattern recognition, classification, and function approximation applications. An input layer, a hidden layer, and an output layer are the traditional three layers that make them up **[179]**.

Radial basis functions are used by neurons in the hidden layer of an RBF network to map the input data onto a high-dimensional space. The most popular radial basis function is the Gaussian function, which gives each input a weight based on how far it is from the centre. The Multiquadric, Inverse Multiquadric, and Thin Plate Spline functions are further useful radial basis functions. The output is produced using a linear combination of the basis functions after the input data has been mapped onto the high-dimensional space. The output layer, which can consist of a single node for regression tasks or many nodes for classification tasks, receives this data next **[180]**.

RBF networks provide several benefits, including the capacity to handle noisy and imperfect input, quick training times, and strong generalisation skills. Numerous applications, such as speech recognition, image classification, and financial forecasting, have made use of them **[181]**.

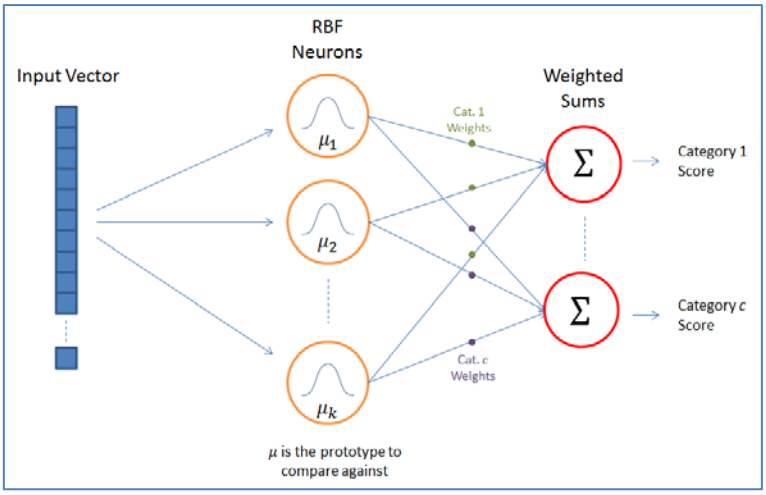

Figure III. 7 Radial Basis Function Networks architecture [182].

III.2.2.6. Hopfield Networks: John Hopfield first described Hopfield networks, a subset of recurrent neural networks, in 1982. They bear his name and were created to store and retrieve data in the form of patterns **[183]**. A Hopfield network's fundamental principle is to employ neuronal feedback connections to establish a collection of stable states that are related to particular input patterns. Each neuron in a Hopfield network is connected to every other neuron, and each connection has a corresponding weight. The patterns that the network has discovered are kept in these weights **[184]**.

The neurons in the network are updated synchronously whenever a new input pattern is provided to it, and this process continues until the network reaches a stable state that matches the input pattern. The network may retrieve previously acquired patterns using this associative memory method, even if the patterns are only partially present in the input [**185]**.

Applications for Hopfield networks include associative memory tasks, optimisation issues, and the recognition of speech and images. They do, however, have certain drawbacks, such as the fact that the number of patterns that can be stored in the network is constrained by the number of neurons and that the retrieval process may be sensitive to input noise **[186]**.

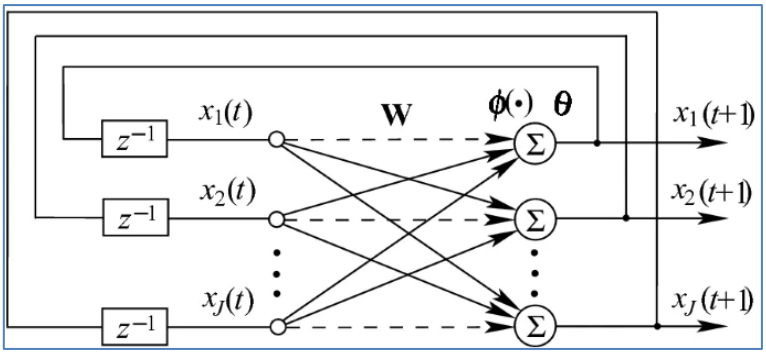

Figure III. 8 Hopfield Networks architecture [187].

III.2.2.7. Deep Belief Networks (DBNs): An example of an unsupervised learning job that Deep Belief Networks (DBNs) are frequently employed for is feature extraction, dimensionality reduction, and data clustering. They are made up of numerous interconnected layers of nodes, and they employ a technique called layerwise pretraining to discover hierarchical representations of the input data **[188]**.

DBNs normally go through two main levels of training. Each layer is trained in the first stage as a restricted Boltzmann machine (RBM), a kind of generative stochastic neural network that simulates the probability distribution of the visible and hidden units together. Backpropagation, which modifies the weights of the connections between the layers to minimise a predetermined cost function, is used in the second stage to fine-tune the entire network **[189]**.

Several applications, such as speech recognition, image recognition, and natural language processing, have made use of DBNs. They are renowned for their capacity to generalise well to new data and for automatically learning valuable features from raw data without the need for manual feature engineering **[190]**.

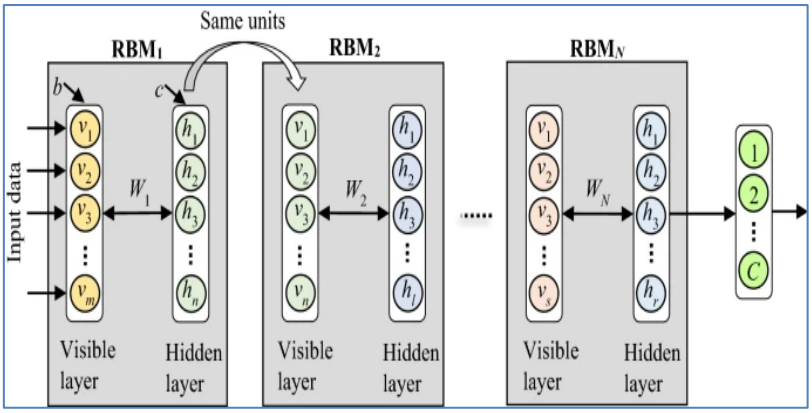

Figure III. 9 Deep Belief Networks (DBNs) architecture [191].

III.2.2.8. Generative Adversarial Network (GAN): A deep learning architecture called a generative adversarial network (GAN) consists of two neural networks: a discriminator and a generator. While the discriminator network tells the difference between real and fake data, the generator network creates artificial data. The generator network creates synthetic data that is meant to resemble actual data from input of random noise **[192]**. The discriminator network predicts whether the data is authentic or phoney by taking both actual and synthetic data as input. The discriminator network seeks to accurately identify real and synthetic data during training while the generator network attempts to create synthetic data that is indistinguishable from genuine data **[163]**.

In a procedure known as adversarial training, the two networks are trained concurrently and optimised in opposite ways. As a result, the discriminator network gets better at telling the difference between real and phoney data whereas the generator network gets better at producing realistic data. This procedure is repeated until the generator network can produce data that can hardly be distinguished from actual data. GANs have been utilised successfully in a variety of applications, including the creation of images, videos, voices, and texts. They are employed in a variety of industries, including fashion, entertainment, and the arts **[194]**.

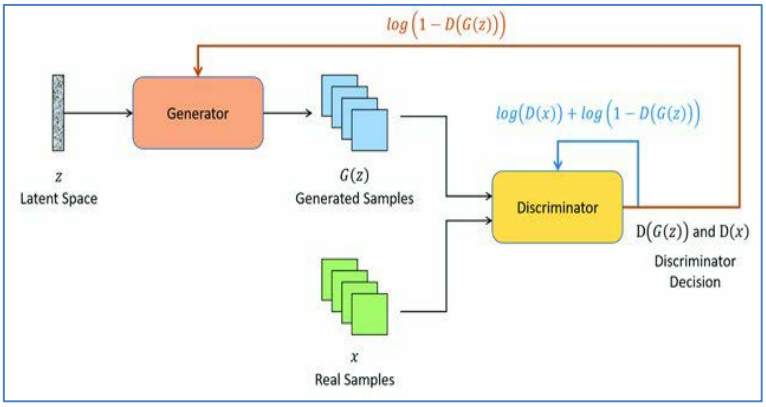

Figure III. 10 Generative Adversarial Network Architecture [195].

III.2.3. Types of Neural Networks Activation Functions:

Activation functions play a crucial part in neural networks as they introduce non-linearity to the model, and decide whether a neuron should be activated or not. In other means, they determine whether the neuron's input to the network is essential or not in the process of prediction using simpler mathematical operations. Here are some commonly used activation functions in neural networks **[196]**:

III.2.3.1. Sigmoid or Logistic Activation Function: The sigmoid function maps any input to a value between 0 and 1. It is often used in the output layer of binary classification problems. The function has the following mathematical expression **[197]**:

$$f(x) = \frac{1}{1 + e^{-x}} \quad \textbf{(42)}$$

III.2.3.2. Rectified Linear Unit (ReLU): The ReLU function returns 0 for any negative input and returns the input value for any non-negative input. The function is defined as **[197]** :

$$f(x) = \max(0, x) \quad \textbf{(43)}$$

III.2.3.3. Leaky ReLU: The leaky ReLU function is similar to ReLU, but it returns a small negative value for any negative input, instead of 0. This helps to overcome the problem of dead neurons in ReLU, where a neuron could become unresponsive to any input during training. The function is defined as **[197]** :

$$f(x) = \max(0.1x, x) \quad \textbf{(44)}$$

III.2.3.4. Tanh or hyperbolic tangent Activation Function: The tanh function is similar to the sigmoid function but maps input to a range between -1 and 1. It is often used in the hidden layers of neural networks. The function is defined as **[197]** :

$$f(x) = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}} \quad \textbf{(45)}$$

III.2.3.5. Softmax: The softmax function is used in the output layer of multi-class classification problems, where the goal is to predict the probability distribution of the input belonging to each class. The function maps the input to a probability distribution that sums up to 1. The function is defined as **[197]** :

$$f(x)_i = \frac{e^{x_i}}{\sum_{j=1}^{n} e^{x_j}} \quad \textbf{(46)}$$

III.2.4. Artificial Neural Networks In Bearing Fault Diagnosis:

ANN has been successfully applied in bearing fault diagnosis. The basic idea behind using ANN for this task is to train a model to learn the patterns of normal and faulty bearings based on vibration signals, acoustic signals or other types of data collected from the bearings **[198]**.

In practice, the process of using ANN for bearing fault diagnosis involves several steps. The first step is to collect data from the bearings, which can be done using sensors such as accelerometers for vibration signals, tachometers for speed, or microphones for an acoustic signal. The data can be in the form of time-domain signals, frequency-domain signals, or time-frequency domain signals.

The second step is to preprocess the data to extract relevant features that can be used to train the ANN. Feature extraction can involve techniques such as statistical parameters, wavelet transform, Fourier transform, or time-domain statistical analysis.

The third step is to train the ANN using the preprocessed data. This involves selecting an appropriate architecture for the ANN, choosing an appropriate learning algorithm, and setting the hyperparameters of the model. The training data should include both healthy and faulty bearing data to enable the ANN to learn the patterns of normal and faulty bearings.

The fourth step is to evaluate the performance of the ANN on a test set of data that it has not seen before. This involves calculating metrics such as accuracy, sensitivity, specificity, and receiver operating characteristic (ROC) curve.

One of the advantages of using ANN in bearing fault diagnosis is their ability to learn from both labelled and unlabeled data. This means that ANN can be trained on a small labelled dataset and then fine-tuned on a larger unlabeled dataset to improve their performance. Also, ANN has proven to be a powerful tool for bearing fault diagnosis, with high accuracy and robustness to noise and variability in the data.

III.3. Random forests:

III.3.1. Random Forests Overview:

A random forests algorithm is a machine-learning technique that belongs to the family of ensemble methods. It is a combination of multiple decision trees, where each tree is trained on a random subset of the training data and a random subset of the features. The output of the random forests is determined by combining the predictions of all the trees in the forest **[199].** The random forests algorithm has two tasks, classification and regression tasks.

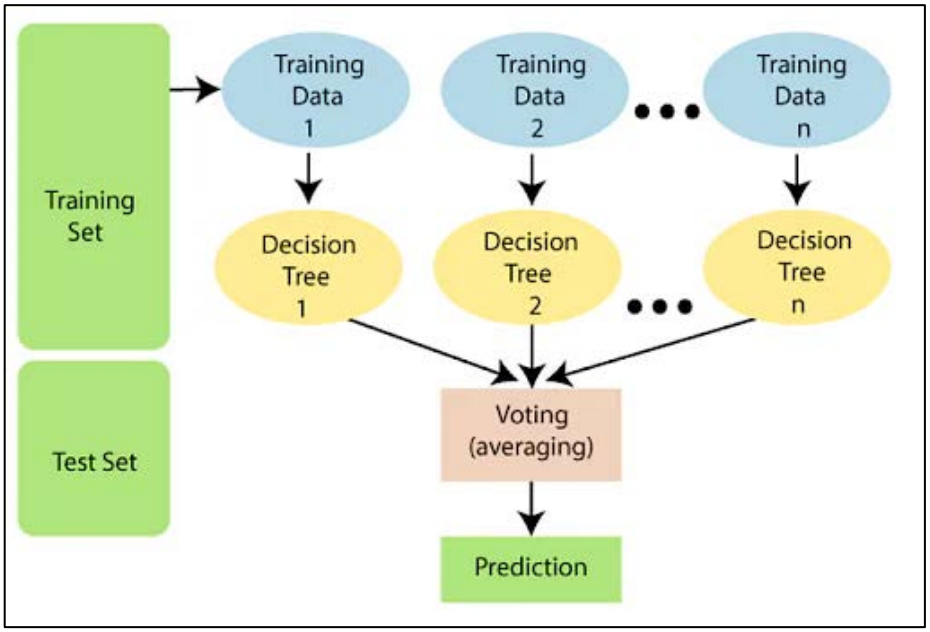

Figure III. 11 Random Forest Algorithm [200]

III.3.2. Classification and regression steps of the random forests:

In classification tasks, the output of the random forests is determined by majority voting among the trees, Here are the steps for performing classification using Random Forest **[201]**:

1. Starting by selecting samples from the training dataset allowing for a replacement to create bootstrap samples.

2. For each bootstrap sample construct a decision tree using a subset of the features. The number of features used is typically set to a value the square root of the total number of features.

3. Repeating steps 1 and 2 to generate a collection or "forest" of decision trees.

4. When classifying a data point make predictions using each decision tree in the forest. The predicted class, for the data point, is determined by majority vote, among all the decision trees.

This approach allows random forest classification to provide predictions based on input data points. Here are a few advantages of utilizing forests for classification;

- Random forests are known for their accuracy particularly when dealing with complex data.
- They have a resistance, to overfitting making them reliable in scenarios.
- Random forests offer ease of interpretation and explanation making it easier to understand and convey the results.
- This algorithm can handle both classification and regression problems effectively.

However there are also some limitations when using forests for classification;

- Training a forest model can be computationally demanding, especially with large datasets.
- The performance of the model can be sensitive to hyperparameter choices, such as the number of trees and features used per tree.
- Achieving optimal performance, by tuning the hyperparameters can be challenging.

In regression tasks, the output is determined by averaging the predictions of the trees. Here are the general steps for performing regression using Random Forest **[201]**:

1. Take samples from the training dataset allowing for duplicates to create bootstrap samples.

2. For each bootstrap sample build a decision tree using a subset of the features. Typically we use a fixed number of features, such, as the root of the number of features.

3. Repeat steps 1 and 2 times to create a collection of decision trees known as a forest.

4. When predicting the value of a data point make predictions using each decision tree in the forest. The average of these predictions becomes the predicted value for the data point.

Some advantages associated with using Random Forests for regression;

- Random Forests tend to provide accurate results particularly when dealing with high dimensional data.
- This algorithm is relatively robust against overfitting issues.
- It can be used effectively for both classification and regression problems.
- Additionally, Random Forests can offer estimates regarding prediction uncertainties.

However, it's important to consider drawbacks when utilizing Random Forests for regression;

- Training Random Forests can be computationally expensive especially when working with large datasets.
- The performance may vary depending on hyperparameter choices such, as the number of trees and features used per tree.
- Tuning these hyperparameters to achieve performance might prove challenging.

III.3.3. Random forest techniques:

In the aspect of ensemble learning, which includes random forests algorithm, the terms bagging and boosting are frequently used **[202]**.

III.3.3.1. Bagging: also known as Bootstrap Aggregation, is a method of building numerous decision tree models using distinct subsets of the training data. Each tree is based on a bootstrap sample of the original data, which is generated by picking occurrences at random with replacement. This strategy is used to reduce model variance and avoid overfitting. Bagging is used in random forests to produce numerous decision trees, each trained on a distinct subset of the training data.

III.3.3.2. Boosting: Using the concept of boosting weak models are iteratively trained and then combined to create a stronger model. Each succeeding weak model focuses more on the instances that the preceding model misclassified since the weak models are trained on various subsets of the training data. This method is used to lessen the model's bias and increase its overall accuracy. Boosting can be used in random forests to enhance each decision tree's performance.

In conclusion, bagging and boosting are two methods used in random forests algorithm to increase precision and decrease overfitting. Whereas boosting is used to lessen the model's bias, bagging is used to lessen the model's variance.

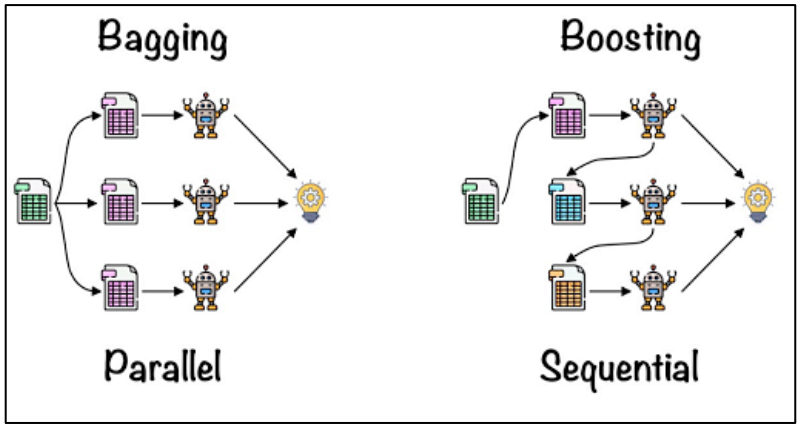

Figure III. 12 Bagging and Boosting Techniques [202]

III.3.4. Bearing fault diagnosis using random forests:

Bearing faults can be detected by analyzing the vibration signals generated by the bearings during their operation. These signals can be collected using accelerometers or other sensors **[203]**. Using random forests in bearing fault diagnosis is a popular method in machine learning for predicting the health of industrial equipment. To identify issues, with bearings using forests the usual steps involve **[204]**:

1. Gathering vibration data from the bearing.
2. Preparing the data by eliminating any noise.
3. Extracting features from the data.
4. Training a forest model using the collected data.
5. Utilizing the model to assess and diagnose the condition of the bearing.

Adjusting the number of decision trees in a forest can enhance model accuracy. While a larger number of trees generally results in accuracy it also increases training time. Random forests are an option for diagnosing bearing faults due to their characteristics;

1. Noise Resistance: Random forests employ decision trees trained on subsets of data making them less susceptible to noise in datasets.
2. Accuracy: With datasets, random forests can achieve high levels of accuracy in diagnosing bearing faults.
3. Scalability: Random forests can effectively handle amounts of vibration data when diagnosing bearings.

III.3.5. Conclusion:

In conclusion, Bearing fault diagnosis using random forests is an effective technique that can help detect faults in bearings early, prevent catastrophic machinery failures, and ultimately save costs associated with unplanned downtime and repair, but this effect depends on several factors, including the quality of the data, the selection of features, and the tuning of hyper-parameters. However, with proper data preparation and algorithm configuration, random forests are a reliable and accurate method for diagnosing bearing faults.

III.4. Support Vector Machine (SVM):

III.4.1. Support Vector Machine (SVM)

A popular supervised machine-learning technique for both classification and regression tasks is the Support Vector Machine (SVM) **[205]**. Due to its precision and efficiency in resolving challenging issues in high-dimensional environments, SVM is a well-known method. SVM, or support vector machines, is a binary classification technique that seeks the optimal hyperplane to divide the data points into two groups. The margin, also known as the hyperplane, is the decision boundary that optimizes the separation between the nearest points from each class. The margin displays the degree of categorization confidence we have **[206]**.

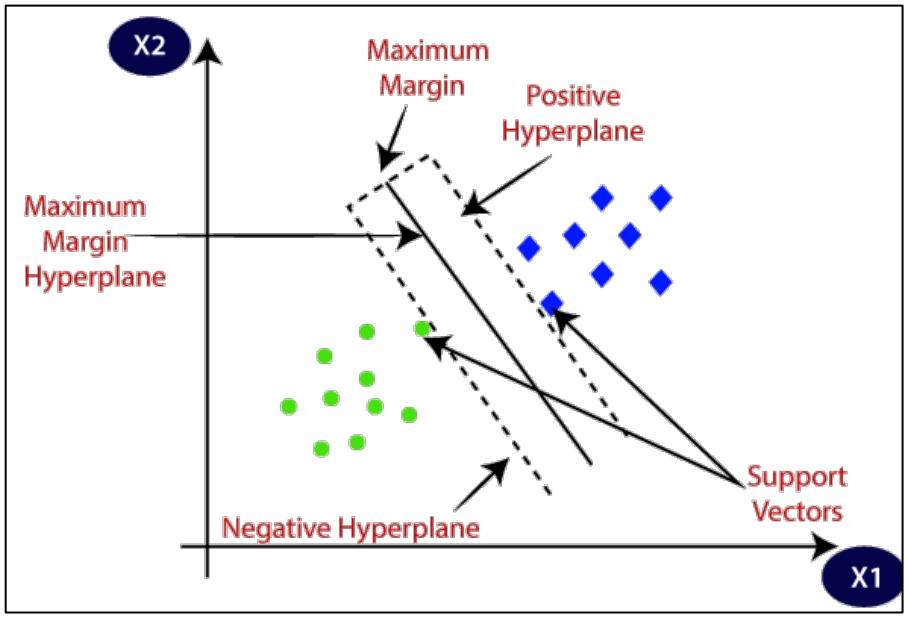

Figure III. 13 Hyperplanes and Support Vectors [206]

SVM optimises the margin according to some limitations to identify the best hyperplane. The restrictions ensure that the hyperplane properly divides the data into classes. SVM transforms the input data into a higher-dimensional space where linear separation is possible using a kernel function **[207]**. In higher-dimensional space, the kernel function computes the dot product of the input data and a fixed reference vector.

III.4.2. Major Kernel Functions in Support Vector Machine (SVM):

To find the optimum hyperplane, SVM optimizes the margin based on a few constraints. The limitations make sure that the data is properly divided into classes by the hyperplane. Using a kernel function, SVM converts the input data into a higher-dimensional space where linear

separation is achievable **[208]**. The dot product of the input data and a fixed reference vector is computed in higher-dimensional space by the kernel function.

Several major kernel functions are commonly used in SVM **[209]** :

III.4.2.1. Linear Kernel: The linear kernel is the simplest and most commonly used kernel function. It is a dot product between two vectors, and it works well for linearly separable data.

$$k(x, y) = x^T . y \quad \textbf{(47)}$$

III.4.2.2. Polynomial Kernel: The polynomial kernel is a popular kernel function that is used when the data is not linearly separable. It maps the data into a higher-dimensional space using a polynomial function, which can help to create a separating hyperplane.

$$k(x, y) = (x^T, y)^P = (x^T . y + \tau)^p \quad \textbf{(48)}$$

III.4.2.3. Radial Basis Function (RBF) Kernel: The RBF kernel is another popular kernel function that is used when the data is not linearly separable. It maps the data into a higher-dimensional space using a Gaussian function, which can help to create a separating hyperplane.

$$k(x, y) = \exp(-\|x - y\|^2/\sigma^2) \quad \textbf{(49)}$$

III.4.2.4. Sigmoid Kernel: This function is equivalent to a two-layer neural network perceptron model, which is utilised as an activation function for artificial neurons. It maps the data into a higher-dimensional space using a sigmoid function, which can help to create a separating hyperplane.

$$k(x, y) = tanh(k_1 x^T y + k_2) \quad \textbf{(50)}$$

III.4.2.5. Laplacian Kernel: The Laplacian kernel, like the RBF kernel, is a non-linear kernel function. It uses a Laplacian function to map the data into a higher-dimensional space, which can aid in the creation of a separating hyperplane.

$$k(x, y) = \exp(-\|x - y\|/\sigma) \quad \textbf{(51)}$$

III.4.3. The importance of SVM parameters:

The SVM algorithm involves the selection of different parameters that can affect the performance of the model. The choice of these parameters can have a significant impact on the accuracy and efficiency of the SVM algorithm **[210]**.

The most important parameters in SVM are **[211]** :

III.4.3.1. Kernel function: The kernel function determines the shape of the decision boundary. SVM can use different kernel functions such as linear, polynomial, Gaussian RBF, sigmoid, etc. The choice of kernel function depends on the characteristics of the data and the problem at hand.

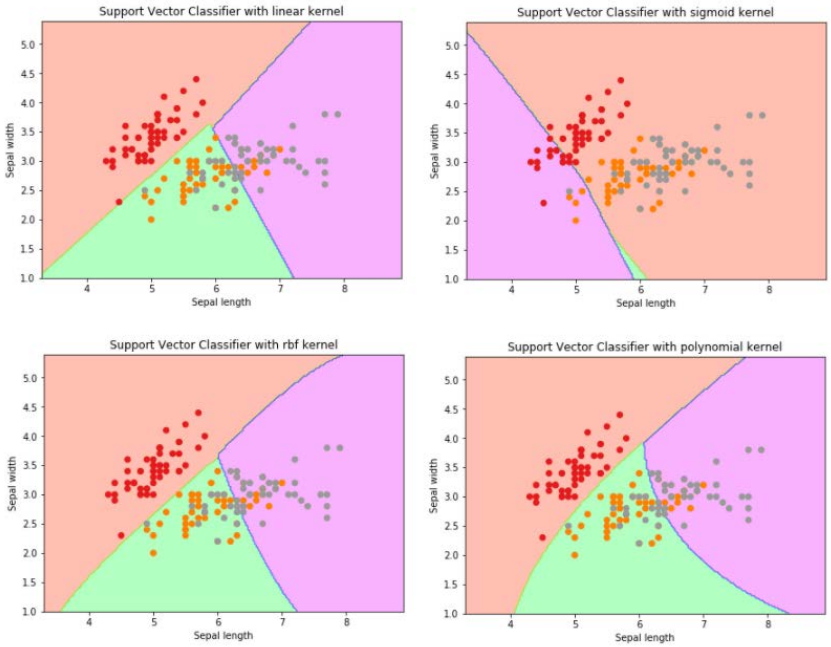

Figure III. 14 The kernel functions [212]

III.4.3.2. C parameter: The C parameter controls the trade-off between maximizing the margin and minimizing the classification error. A smaller value of C will result in a wider margin and more tolerance for misclassified data points, while a larger value of C will result in a smaller margin and less tolerance for misclassified data points.

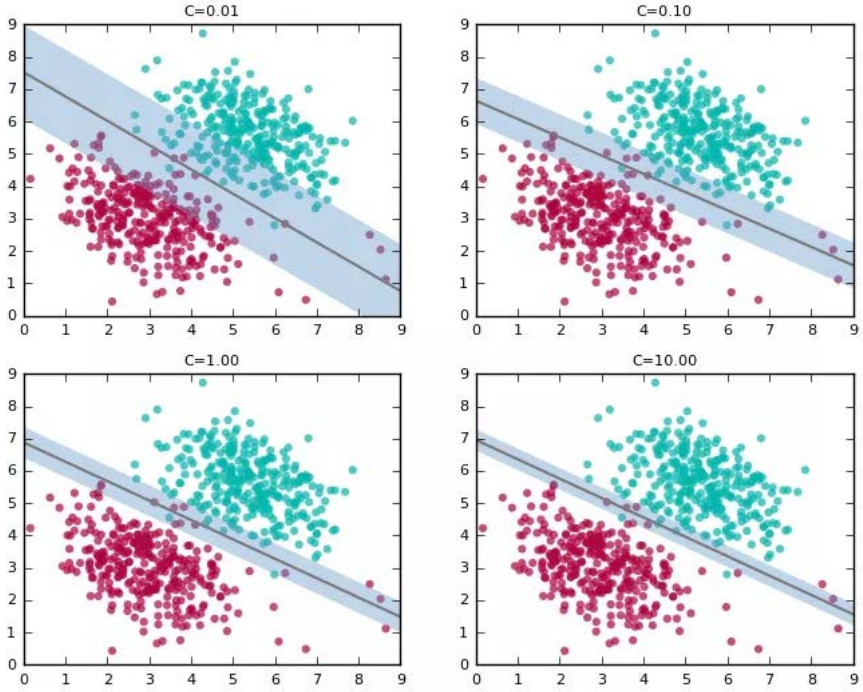

Figure III. 15 The effect of the C parameter in SVM [213]

III.4.3.3. Gamma parameter: The gamma parameter controls the form of the decision boundary for non-linear kernels. A higher gamma value results in a more complex and wavy choice border, whereas a lower gamma value results in a smoother decision boundary.

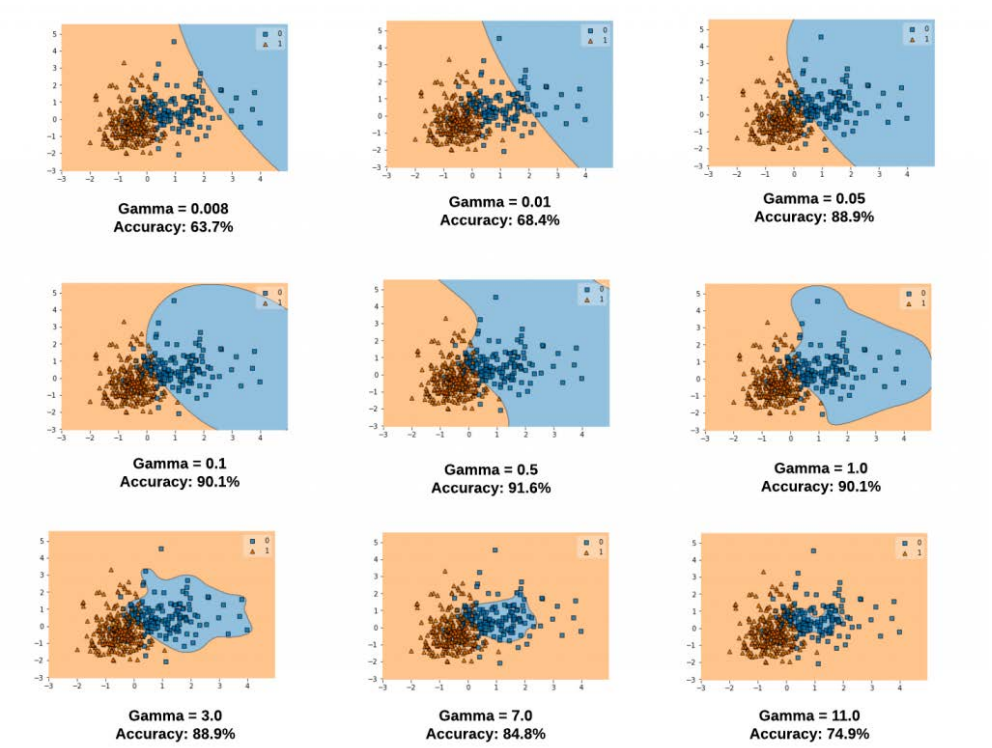

Figure III. 16 The effect of the gamma parameter [214]

III.4.3.4. Degree parameter: The degree parameter is used for polynomial kernels and determines the degree of the polynomial function used for mapping the input data into a higher-dimensional space.

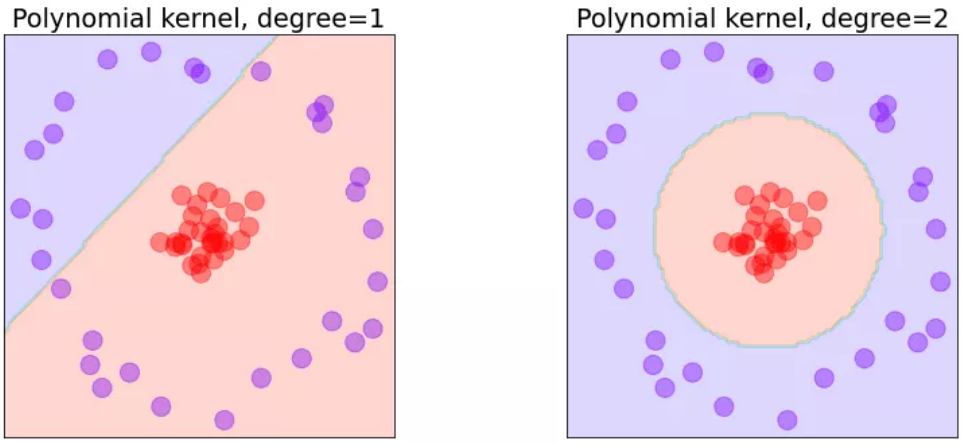

Figure III. 17 The polynomial degree parameter [215]

III.4.3.5. Class weights: The class weights parameter is used to handle a class imbalance in the data. It assigns higher weights to the minority class to give it more importance during the training process.

Several methods, such as grid search, random search, or Bayesian optimization, can be used to choose these parameters. Finding the set of settings that produces the best performance on a validation set is the objective. It is crucial to remember that choosing the right parameters can be difficult and time-consuming, particularly for huge datasets or complex situations. Thus, it is advised to use automated methods or specialist knowledge to direct the selection process **[216]**.

III.4.4. Types of Classification in Support Vector Machine:

In terms of classification, there are mainly three types of SVM classification **[217]**:

III.4.4.1. Binary classification: In binary classification, the SVM algorithm learns a hyperplane that separates the data into two classes. The hyperplane is chosen in such a way that it maximizes the margin, which is the distance between the hyperplane and the nearest data points from each class. The data points that lie on the margin are called support vectors.

III.4.4.2. Multiclass classification: In multiclass classification, the SVM algorithm learns a hyperplane that separates the data into more than two classes. There are two main approaches for multiclass classification in SVM:

III.4.4.2.a. One-vs-One (OVO): In this approach, the SVM algorithm trains multiple binary classifiers for all possible pairs of classes. For example, if there are n classes, then n(n-1)/2 binary classifiers are trained. The final prediction is made by combining the predictions of all the binary classifiers.

III.4.4.2.b. One-vs-All (OVA): In this approach, the SVM algorithm trains a binary classifier for each class, where each classifier is trained to distinguish that class from all the other classes. The final prediction is made by selecting the class with the highest score among all the binary classifiers.

III.4.4.3. Non-linear classification: SVM seeks a non-linear boundary that divides the data in non-linear classification. SVM accomplishes this by translating the initial data into a higher-dimensional space where a hyperplane that divides the classes can be located. The kernel trick, a method for doing this without explicitly computing the transformation, enables SVM to implicitly translate the data into a higher-dimensional space **[218]**. The radial basis function (RBF) kernel, polynomial kernel, and sigmoid kernel are a few common kernel functions utilised in SVM.

III.4.5. Bearing fault diagnosis using Support Vector Machine:

Support Vector Machines (SVM) is a wide-use machine learning algorithm that can be used for fault diagnosis of mechanical systems such as bearing faults. Bearing faults can lead to severe machinery failure and unexpected downtime, which can have significant economic and safety consequences **[219]**. Therefore, early detection of bearing faults is crucial in ensuring efficient and safe machinery operation. In the case of bearing fault diagnosis, SVM can be used for classification, where it can predict whether a bearing is faulty or not based on the features extracted from the vibration signals **[220]**.

The process of using SVM for bearing fault diagnosis typically involves the following steps **[221]** :

1. Data Collection: Collecting the vibration signals from the bearings using an appropriate sensor.

2. Feature Extraction: Extracting features from the vibration signals using signal processing techniques. These features can include time-domain features, frequency-domain features, and time-frequency features.

3. Data splitting: Split the dataset into training and testing sets. The training set is used to train the SVM model, while the testing set is used to evaluate the performance of the model.

4. Training: Training the SVM model using the training dataset. The SVM algorithm tries to find the best hyperplane that separates the data points into different classes.

5. Model Evaluation: Evaluating the performance of the SVM model using the testing dataset. Metrics like accuracy and precision can be used to assess the model's performance.

6. Making a Decision: Using the trained SVM model to predict the health status of the bearings.

III.4.6. Conclusion:

It has been demonstrated that early-stage bearing faults can be detected and the remaining usable life of the bearings can be predicted using SVM-based bearing fault diagnosis. However, the selection of the hyperparameters and kernel function, as well as the calibre of the training data, can have an impact on how well the SVM model performs. To create a reliable and accurate SVM-based bearing fault diagnosis model, careful consideration must be given to the kernel function and hyperparameter selection, as well as the gathering and preprocessing of the training data **[222]**.

III.5. Extreme learning machine (ELM):

III.5.1. Extreme learning machine Overview:

The ELM is a kind of machine learning algorithm that may be used to solve supervised learning problems like classification and regression. Additionally, it is a simple and efficient one-hidden-layer feed-forward neural network (SLFN) **[223]**. The hidden layer's output is combined linearly into the ELM's output layer, whose weights are established by the least-squares technique. The ELM also offers a network training analytical solution. The thresholds of the hidden layer and the weights of the input layer in ELM are set at random and are static after generation **[224]**. Once the hidden layer activation function and node count have been chosen, the training data can be used to determine the single best solution. The original structure of the ELM is depicted below. The training data can be used to identify the single optimal solution once the hidden layer activation function and node count have been specified. Below is an illustration of the ELM's original structure **[225]**.

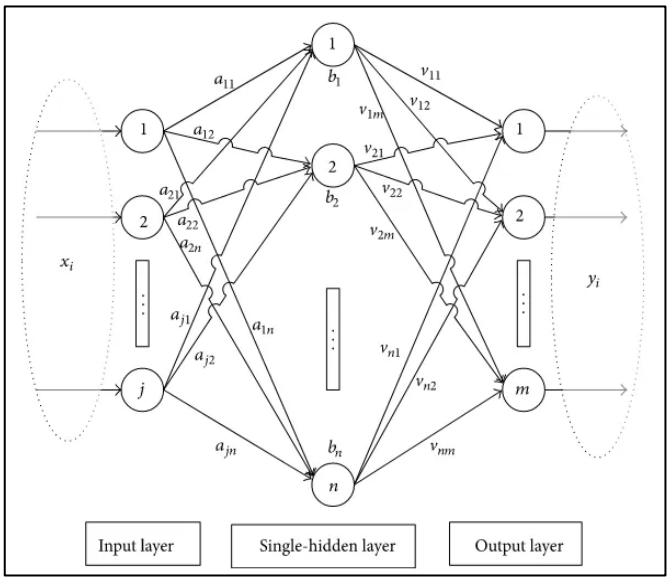

Figure III. 18 The structure of ELM

III.5.2. Extreme learning machine Model:

Where n and m are the dimensions of the input matrix and output matrix, respectively, and X and T are the inputs and outputs of the ELM. The following is how they are expressed **[226]**:

$$X = \begin{bmatrix} x_{11} & \dots & x_{1Q} \\ x_{n1} & \dots & x_{nQ} \end{bmatrix} \text{ and } T = \begin{bmatrix} t_{11} & \dots & t_{1Q} \\ t_{m1} & \dots & t_{mQ} \end{bmatrix} \quad \textbf{(52)}$$

Between the input layer and the hidden layer, the weights have been assigned at random:

$$W = \begin{bmatrix} w_{11} & \cdots & w_{1m} \\ w_{n1} & \cdots & w_{nm} \end{bmatrix} \quad \textbf{(53)}$$

Where w_{ij} represents the weights between the jth input layer neuron and ith hidden layer neuron.

The ELM takes into account the following weights between the hidden layer and the output layer:

$$\beta = \begin{bmatrix} \beta_{11} & \cdots & \beta_{1m} \\ \beta_{k1} & \cdots & \beta_{km} \end{bmatrix} \quad \textbf{(54)}$$

In this instance, ***ij*** stands for the weights between the j^{th} hidden layer neuron and the k^{th} output layer neuron. The bias of the hidden layer neurons is randomly adjusted by the ELM:

$$B = [b_1 \quad b_2 \quad \ldots \quad b_n]' \quad \textbf{(55)}$$

The network activation function is determined by the ELM, g(x). **Figure III. 18** shows how the output matrix T can be expressed.

$$T = \begin{bmatrix} t_{11} & \cdots & t_{1Q} \\ t_{m1} & \cdots & t_{mQ} \end{bmatrix}_{m \times Q} \quad \textbf{(56)}$$

Each column vector of the output matrix *T* is as follows:

$$t_j = \begin{bmatrix} t_{1j} \\ t_{2j} \\ \vdots \\ t_{mj} \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^{l} \beta_{i1} g(w_j x_j + b_i) \\ \sum_{i=1}^{l} \beta_{i2} g(w_j x_j + b_i) \\ \vdots \\ \sum_{i=1}^{l} \beta_{im} g(w_j x_j + b_i) \end{bmatrix} ; (j = 1, 2, \ldots, Q) \quad \textbf{(57)}$$

From **(56)** and **(57)**, we obtain:

$$H\beta = T' \quad \textbf{(58)}$$

Where *H* is the output of the hidden layer and T' is the transpose of *T*. We compute the weight matrix values of β using the least square method to get the unique answer with the least amount of error.

here is T' the transpose of *T* and *H* is the output of the hidden layer. To obtain the unique solution with minimum error, we use the least square method to calculate the weight matrix values of β.

$$\beta = H^{+}T' \quad \textbf{(59)}$$

Where: H+ is the generalized Moore-Penrose inverse of matrix H.

We add a regularisation term to the β **[227]** to increase the network's capacity for generalisation and stabilise the results. β can be written as follows when there are fewer hidden layer neurons than training samples:

$$\beta = (\frac{1}{\lambda} + H'H)^{-1}H'T' \quad \textbf{(60)}$$

Where: $\beta'\beta = I$ and λ are the regularisation coefficients used to balance the network's complexity and training accuracy **[228]**. When there are more hidden layer nodes than training samples, the formula for β is **[229]**:

$$\beta = H'(\frac{I}{\lambda} + HH')^{-1}T' \quad \textbf{(61)}$$

The ELM algorithm can be summarized as follows **[230]**:

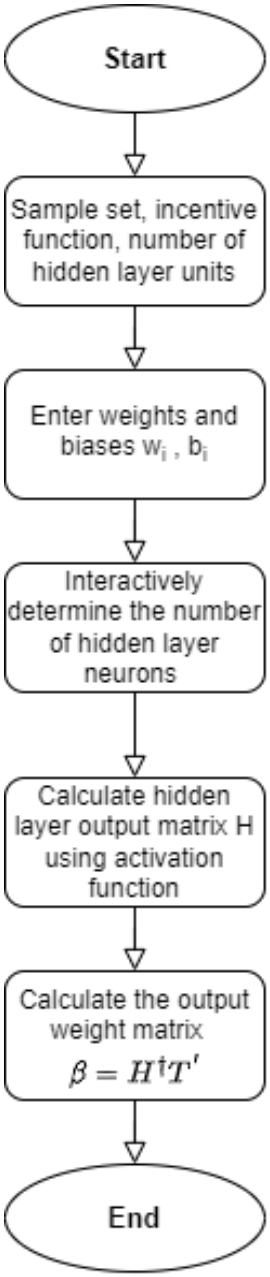

Figure III. 19 The ELM algorithm [230]

III.5.3. Conclusion:

Extreme Learning Machine (ELM) has several advantages in classification. According to ScienceDirect Topics **[231]**, some of these advantages include: Fast and efficient learning speed, Fast convergence, Good generalization ability, Smallest training time and error, Better generalization performance, Simple algorithm, no need to decide the number of hidden layers, learning rate, and other hyperparameters and Can outperform support vector machines in both classification and regression applications.

III.6. Conclusion:

Machine learning algorithms can be trained on data collected from sensors that monitor vibration and acoustic signals from bearings. The data collected is used to create models that can accurately detect and classify different types of errors **[232]**. Models can be trained using different types of machine learning techniques, e.g. decision trees, neural networks, support vector machines and random forests. It has a significant impact on bearing fault diagnosis by improving the accuracy and efficiency of the diagnostic process. Here are some specific ways machine learning can benefit bearing fault diagnosis **[233]**:

Improved accuracy: Machine learning algorithms can identify subtle changes in vibration signals that may be missed by human analysts, improving the accuracy of bearing fault diagnosis.

Faster diagnosis: Machine learning algorithms can process large amounts of data quickly, reducing the time needed to diagnose a bearing fault. This is especially important in industrial settings where equipment downtime can be costly.

Early detection: Machine learning algorithms can identify the early stages of bearing faults, allowing for preventative maintenance to be performed before a failure occurs.

Reduced costs: By detecting bearing faults early, machine learning can reduce the costs associated with unplanned downtime and equipment repair/replacement.

Automation: Machine learning can automate the entire bearing fault diagnosis process, reducing the need for human intervention and freeing up resources for other tasks.

CHAPTER IV

PATTERN RECOGNITION IN BEARING FAULT DIAGNOSIS

IV.1. Bearing fault diagnosis based on Linear discrimination analysis (LDA) and support vector machine (SVM):

In this work, a unique approach is suggested for diagnosing bearing defects. The suggested approach is composed of two key steps. In the first step, linear discriminant analysis is used to reduce the dimensionality after statistical parameters are produced from vibration data received using a tri-axes accelerometer. The optimum feature subset is then created using the LDA components. The second step incorporates an enhanced support vector machine based on a whale optimization technique to categorize defects. Finally, the suggested methodology is tested and evaluated using real-time vibration signals.

IV.2.1. The suggested approach

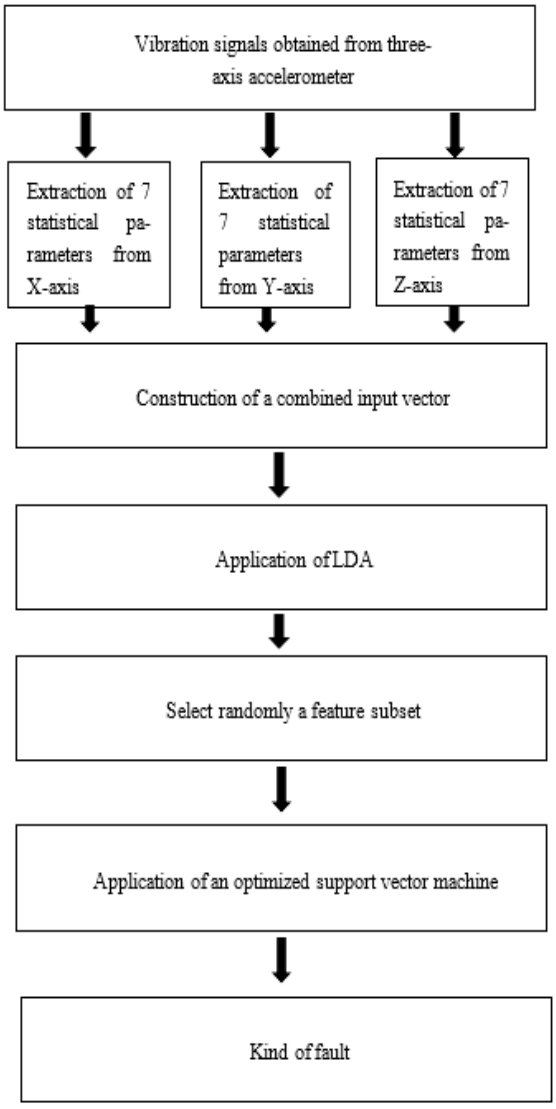

Figure IV. 1: The suggested approach

The suggested approach can be decorticated as follows:

Step 1. Extraction of 7 statistical parameters from each of the three vibration signals of the accelerometer. As a result, an input vector that contains 21 parameters is obtained for each vibration signal. The extracted statistical parameters are the variance, the kurtosis, the mean, the standard deviation, the skewness, the moment, and the covariance.

Step 2. Application of linear discrimination analysis to reduce the dimensionality of the constructed vector.

Step 3. Randomly selected feature sub-set and used it as input of SVM.

Step 4. Application of an optimized support vector machine based on a whale optimization algorithm to automate the classification phase.

Step 5. Calculate the test classification rate.

Step 6. Repeat steps 3 to step 5 for all combinations and find the most superior classification rate and the salient feature subset.

IV.2.2. Test rig and simulation results

The used dataset in this study was acquired from a test bench developed at The “DIRG Lab in the Department of Mechanical and Aerospace Engineering at Politecnico di Torino” **[24]**. The test bench is presented in **Figure IV.2** and **Figure IV.3**, and it contains a high-speed spindle conducting a hollow shaft supported by identical roller bearings B1 and B3. The considered bearing is B1 in this study, where a tri-axial accelerometer is mounted on its support.

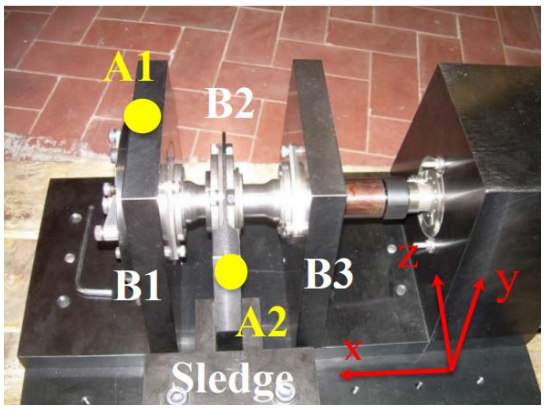

Figure IV. 2: The positions of the two accelerometers

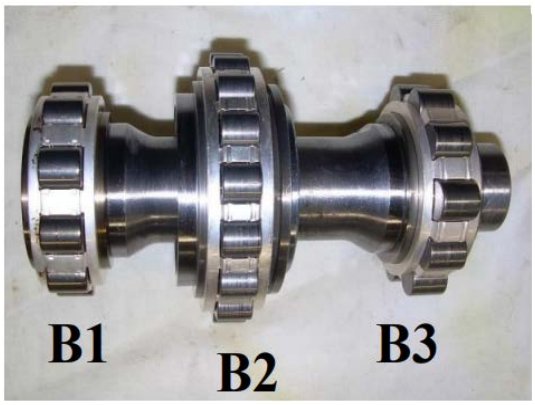

Figure IV. 3: The three roller bearings on the shaft

This experiment extracts vibration signals from the three axes accelerometers placed in point A1. 250 signals are acquired for each class, 180 signals are employed for training, and 70 for test purposes.

Table IV.1 summarizes some obtained results using various feature subsets. It also gives the optimal C and δ parameters obtained by applying the whale optimization algorithm.

Table IV.1 Obtained results using LDA

Sub-set	Training rate	Test rate	optimal C	optimal δ
The two first components of LDA	99.60	100	631.5481	15.2808
2^{nd} and 3^{rd} component of LDA	99.44	100	228.0672	10.0502
3^{rd} and 4^{th} component of LDA	75.7143	77.4603	10000	35.7

After applying the proposed methodology, it has been found that the combination of only the first and the second components gives good results with a test rate equal to 100%.

Figure IV.4 and **Figure IV.5** illustrate the distribution of the input training and test data. We notify you that the different classes are well separated and ready for classification.

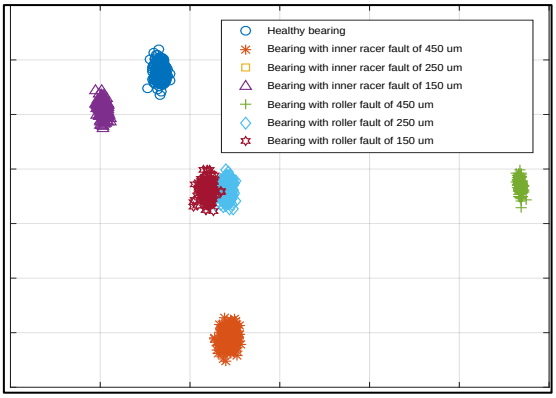

Figure IV. 4: The two first components were obtained from LDA (Training data)

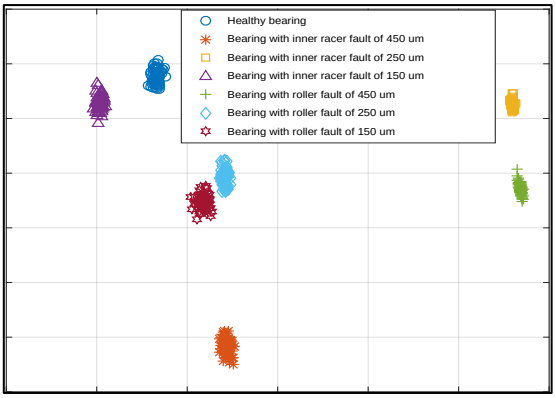

Figure IV. 5: The two first components were obtained from LDA (Test data)

IV.2.3. Conclusion:

This study develops a novel method for diagnosing severity faults. In this method, the tri-axis accelerometer's three vibration signals are used to extract statistical parameters. Linear discrimination analysis is then used to reduce the input vector's dimension. Following that, many subsets are randomly selected and used as an input vector for the SVM classifier. Finding the ideal SVM parameters also involves successfully applying the whale optimisation algorithm. The obtained results demonstrate that the suggested method can successfully extract the ideal feature subset and produce accurate classification outcomes.

IV.2. Independent vector analysis and extreme learning machine (ELM)-based bearing fault diagnosis:

Using IVA, feature selection, and ELM classifiers, a novel bearing defect diagnosis architecture is developed in this study. The following phases make up the recommended process for using vibration signals: The IVA is first developed in order to identify the various vibration signal components from one another. Second, data from each source that was gathered is used to create statistical parameters. Three binary optimisation techniques, including the binary bat algorithm, binary particle swarm optimisation, and binary grey wolf optimisation, are then utilized individually for feature selection. Three classifiers based on ELM, ANN, and RF are then used to finish the classification process.

The results show that IVA produces an ideal input vector with only five features for each sample and an extremely low misdiagnosis rate of 0.76%, whereas feature selection based on BGWO and classification using an ELM also yields optimal input vectors. The collected results further show that the suggested methodology offers the best classification outcomes and great visibility when compared to the other evaluated alternatives.

IV.2.1. The suggested architecture:

Figure IV.6 shows the suggested architecture, which includes the following steps:

Step 1: Each data sample contains three signals, each of which is a vibration signal acquired from one of the three unique directions of x, y, and z using a tri-axis accelerometer.

Step 2: For each data sample, the IVA approach is applied using the three vibration signals as input. Vibration signal components are separated from one another using IVA. After applying IVA, three independent sources are gathered.

Step 3: After the IVA algorithm has been used, statistical parameters are gathered from the three independent sources to build a rich input vector for classification. However, because they have different classification sensitivity factors, not all of the retrieved attributes are necessary to have a good classification performance. The feature subset with the relevant parameters should be chosen as the classification input vector as a result. To complete this objective, swarm intelligent feature selection techniques will be employed. Binary BAT, Binary PSO,

and Binary GWO are the three different binary swarm algorithms we employ. Then, consideration is given to the feature subset with the best objective function.

Step 4: Utilising FNN, RF, and ELM, the classification step is ensured.

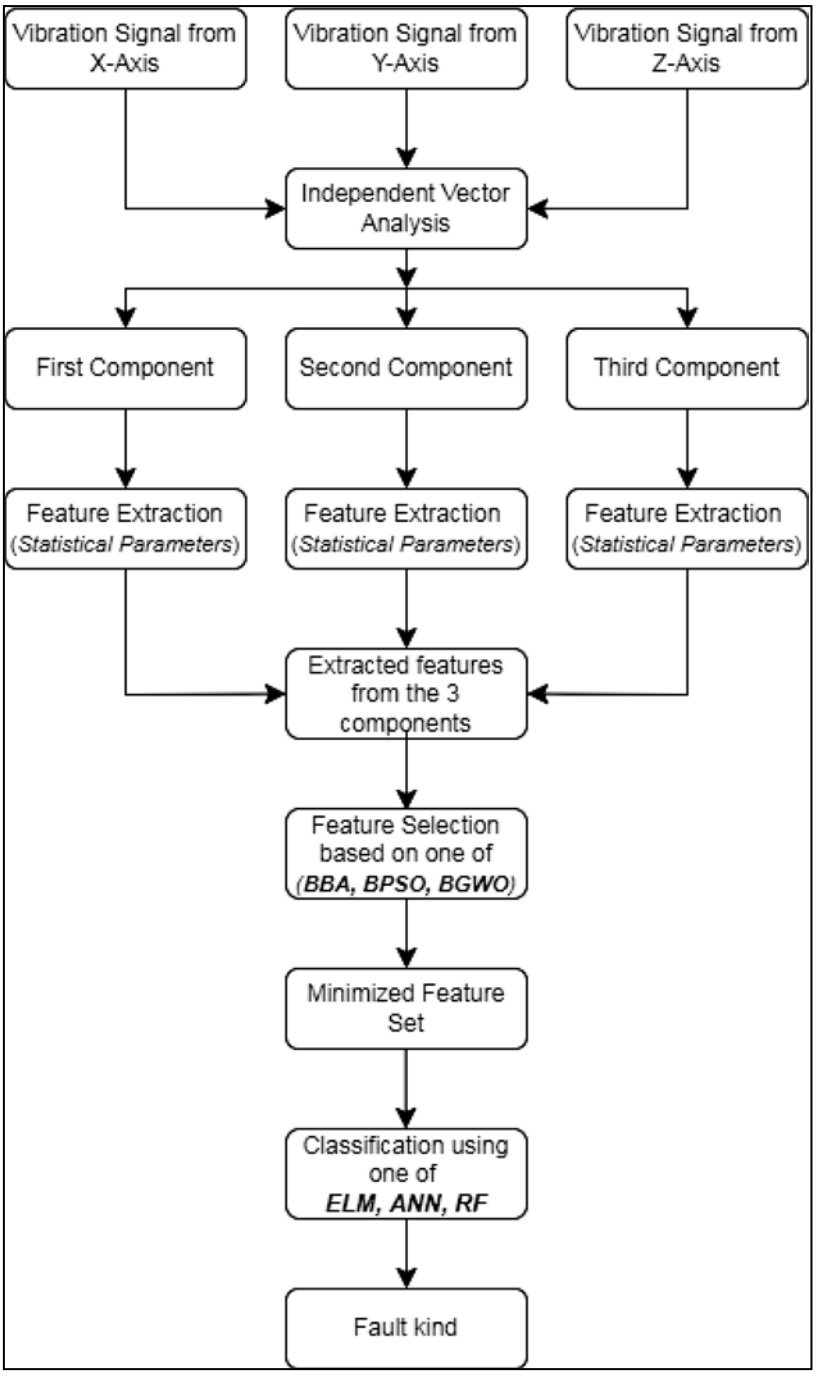

Figure IV. 6: The suggested architecture

IV.2.2. Experimental setup:

For experimental verification and to evaluate our suggested method, we used the bearing fault data from the "Dynamic and Identification Research Group (DIRG) at the Department of Mechanical and Aerospace Engineering at Politecnico di Torino, Italy", in this study **[24]**. Tri-axial accelerometers are used to record vibration signals with a sampling frequency of fs = 51200 Hz for T = 10 s. The various bearing states are listed in **Table I. 4**, with A0 being the healthy case and (A1-A6) denoting the damaged cases. There are 875 samples in total, 125 for each class, in the database. The database is split into two sections: training data makes up 70 % of the database, while test data makes up 30 %.

IV.2.3. The obtained results:

The IVA application for recovering independent sources is the first step. The three sources for a healthy bearing and a defective bearing are shown in **Figure IV.7** and **Figure IV.8**, respectively. **Figure IV.9** and **Figure IV.10** show that the independent sources collected vary from one bearing state to another.

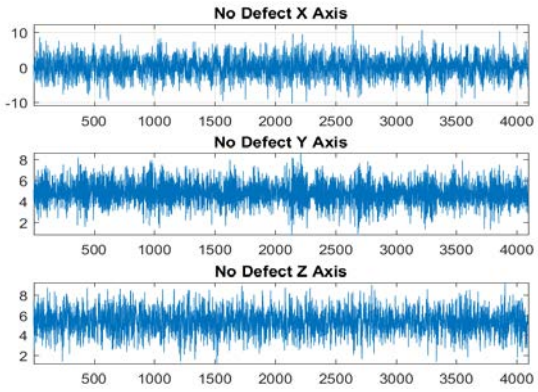

Figure IV. 7: Vibration signals of healthy bearing

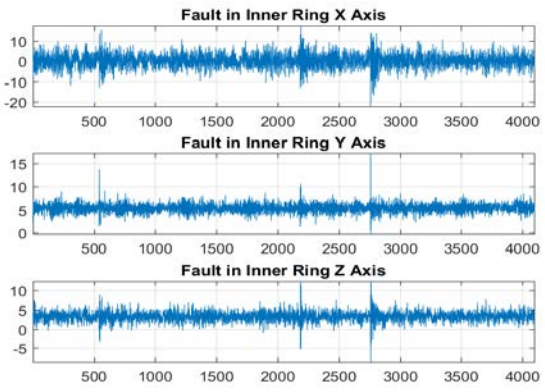

Figure IV. 8: Vibration signals of defective bearing

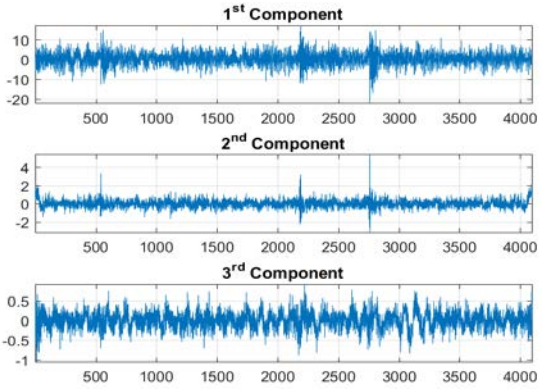

Figure IV. 9: Healthy bearing after using IVA

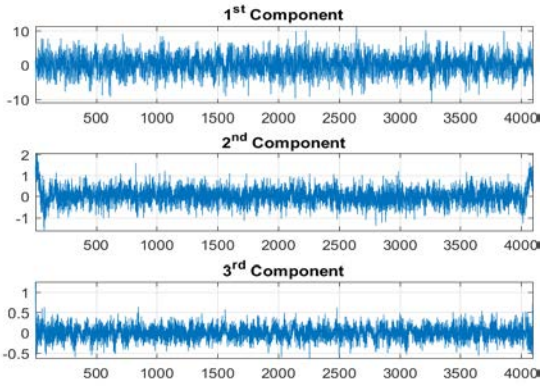

Figure IV. 10: Defected bearing after using IVA

Following the application of the IVA, step 2 involves feature extraction. It is required to define a specific feature in order to use the whole data set for troubleshooting. The three independent sources are used to derive the seven (07) statistical parameters for each bearing condition that were defined in Chapter II.

As a result, for each bearing situation, an input vector with 21 features is obtained. These qualities don't have the same level of sensitivity. To choose the appropriate feature subset, three binary optimisation algorithms—BPSO, BGWO, and BBA are individually used.

The tuning parameters used for the BBA are listed in **Table IV.2**. There are 20 particles that are sufficient to solve the typical tasks, however, more particles may be needed to tackle some challenging issues. The particle number is chosen to be 30 for this reason.

Table IV.2 Tuning parameters for BBA

Parameter	**Values**
Maximum iterations	100
Number of particles (NoP)	30
Number of dimensions (Nov)	21
Loudness (A)	[1 - 2]
Pulse rate (r)	[0 - 1]
Minimum frequency (Qmin)	20
Maximum frequency (Qmax)	50
Initial frequency (for each particle)	0
Initial velocity (for each particle)	0
Initial position (for each particle)	0

Each individual's loudness (A_i) and heart rate (r_i) are both randomly initialised in the [0-1] and [1-2] ranges, respectively.

The tuning parameters utilised for the BPSO are listed in **Table IV.3**. The choice of numerous parameters, including the particle number, the cognitive parameter c1, the social parameter c_2, and the inertia weight *w*, affects the results of PSO.

Table IV.3 The BPSO variables utilized in this study

Parameter	Values
Maximum iteration	100
Number of particles	10
Cognitive parameter, c_1 and social parameter, c_2	2
Velocity [V_{min}, V_{max}]	[-6, 6]
Inertia weight, *w*	[0.4, 0.9]

The PSO parameters in this study are as follows: c1 = c2 = 2. Equation **(31)** is used to compute the inertia weight.

The tuning settings used for the BGWO are listed in **Table IV.4**. The benefit of BGWO is that it converges to an optimal solution independent of initialization and randomization used.

Table IV.4 Setting the BGWO parameters

Parameter	Values
Numbers of iterations	100
Numbers of search agents	10
Problem dimension	21
Search domain	[0, 1]

There aren't many factors to adjust, it's easy to strike a good balance between exploration and exploitation, and it can lead to a beneficial convergence. GWO is also straightforward, user-friendly, adaptable, and scalable. The convergence of the objective functions of BBA, BPSO, and BGWO are depicted in **Figure IV.11, 12**, and **13**, respectively. The objective function is provided in this case as follows:

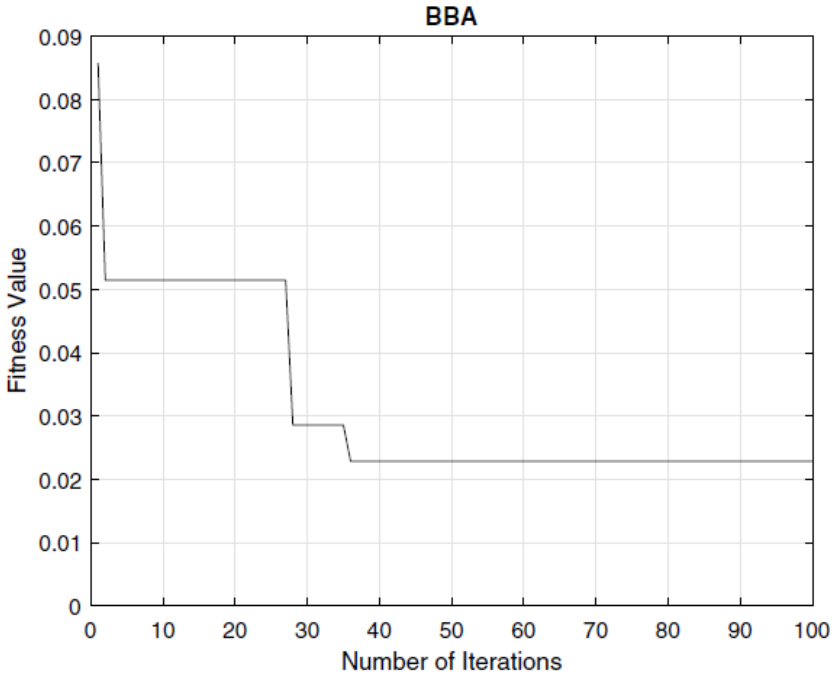

Figure IV. 11: Convergence of the BBA-based cost function

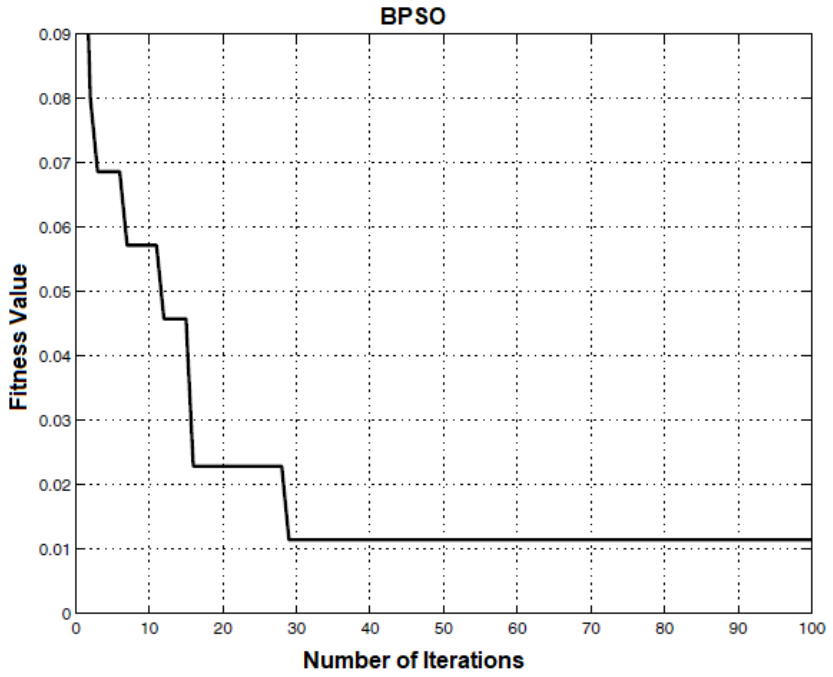

Figure IV. 12: Convergence of the BPSO-based cost function

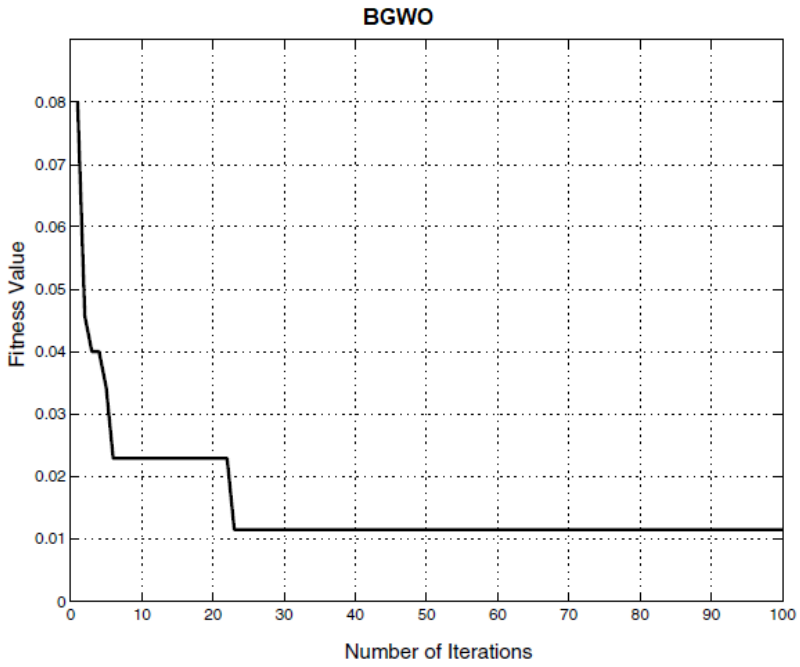

Figure IV. 13: Convergence of the BGWO-based cost function

Cf = 1 Tacc, where Tacc is the training accuracy of a k closest neighbour unsupervised classifier. We can see that the BGWO algorithm converges more quickly than the BBA and BPSO algorithms from **Figure IV.11, 12**, and **13**. We can also observe that the BGWO algorithm, with Cf = 0.011429, has attained the smallest objective function. The RMS and SD of the first independent component, the PPV and CRF of the second independent component, and the PPV of the third independent component make up the ideal feature subset that is produced by using the BGWO method.

Three classifiers namely, the extreme learning machine classifier, feed-forward neural network, and random forest—are used after choosing the salient feature set to guarantee the classification step. Different architectures for feed-forward neural networks are tested. The neural network with one hidden layer and 10 neurons, which we discovered, provides the best test accuracy, which is equal to 98.10%. Based on the number of trees, the random forest classifier is put through a lot of tests. The best outcome was reached with 10 trees, and its test accuracy was 97.34%.

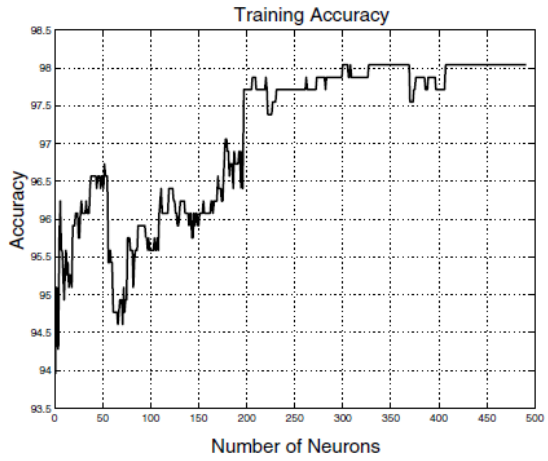

Figure IV. 14: Training ELM accuracy with feature selection based on BBA

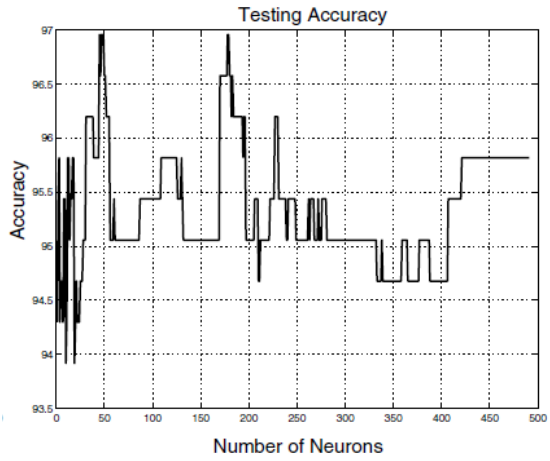

Figure IV. 15: Testing ELM accuracy with feature selection based on BBA

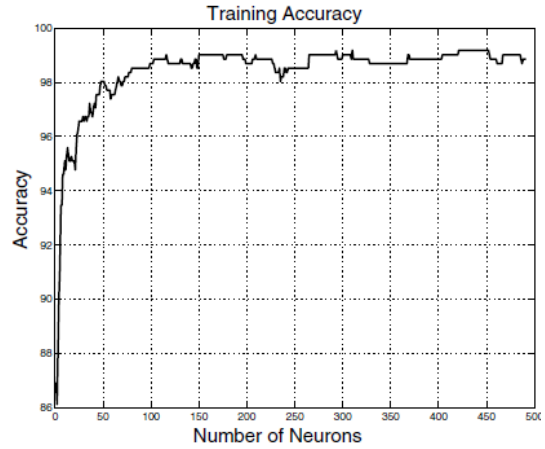

Figure IV. 16: Training ELM accuracy with feature selection based on BPSO

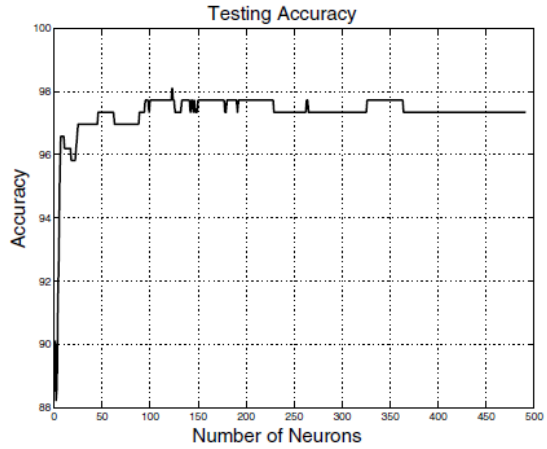

Figure IV. 17: Testing ELM accuracy with feature selection based on BPSO

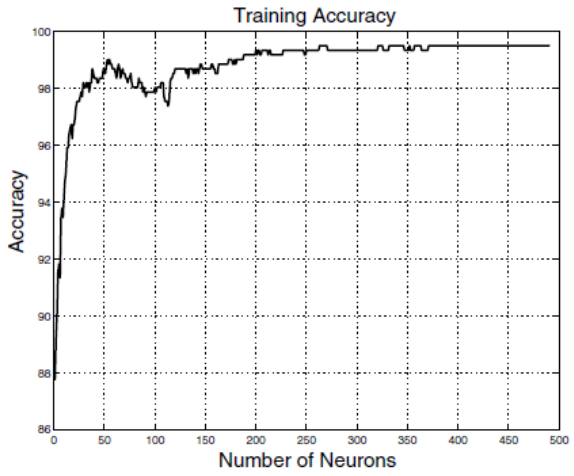

Figure IV. 18: Training ELM accuracy with feature selection based on BGWO

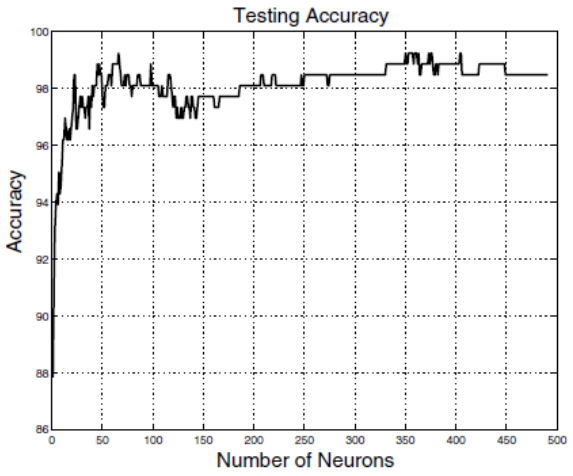

Figure IV. 19: Testing ELM accuracy with feature selection based on BGWO

The input weights, bias value, and the number of hidden neurons are the three main parameters that affect how well ELM performs classification. The input weights and thresholds of the hidden layer neurons of the ELM classifier are initialised at random in this investigation between the values [-1, 1]. Equations **(60)** and **(61)** can be used to calculate the input weights of the output layer.

The sigmoid function is used as an activation function in this thesis. The ELM is extensively useful in many disciplines since it has the advantages of not easily slipping into a local minimum and having a better capacity for generalisation than conventional approaches.

According to **Figure IV.14–19**, the ideal number of hidden neurons is chosen based on the highest test accuracy. **Figure IV.14–19** illustrates how the number of neurons in the hidden layer based on BBA, BGWO, and BPSO fluctuates depending on training and test accuracy. According to **Figure IV.14–19**, the optimum architecture was created using the BGWO feature selection and the ELM classifier, which has 405 neurons in the hidden layer and has a test accuracy of **99.24%**.

Table IV.5 compares the test accuracy obtained using the three classifiers and the three feature selection algorithms, and the results show that the extreme learning classifier based on grey wolf feature selection provides the highest test accuracy.

Figures IV.20 to **22** show, respectively, the confusion matrix of the test results based on the BBA, BPSO and BGWO, and excessive learning. **Figures IV.20–22** show that the proposed technique, which is based on BGWO and extreme learning, produces the greatest results, with the majority of samples being correctly identified and only two examples being incorrectly classified. However, there are 08 and 05 samples that are incorrectly classified, respectively, when we use BBA and BPSO.

Table IV.5 Various classifiers used to determine classification accuracy

	Classifier					
	Extreme Learning Machine		**Artificial Neural Networks**		**Random Forests**	
Feature Selection Technique	**Train (%)**	**Test (%)**	**Train (%)**	**Test (%)**	**Train (%)**	**Test (%)**
The BBA	98.04	96.96	85.29	85.55	99.67	96.58
The BPSO	99.18	98.10	85.62	85.17	99.67	97.34
The BGWO	99.51	99.24	97.88	98.10	99.84	97.34

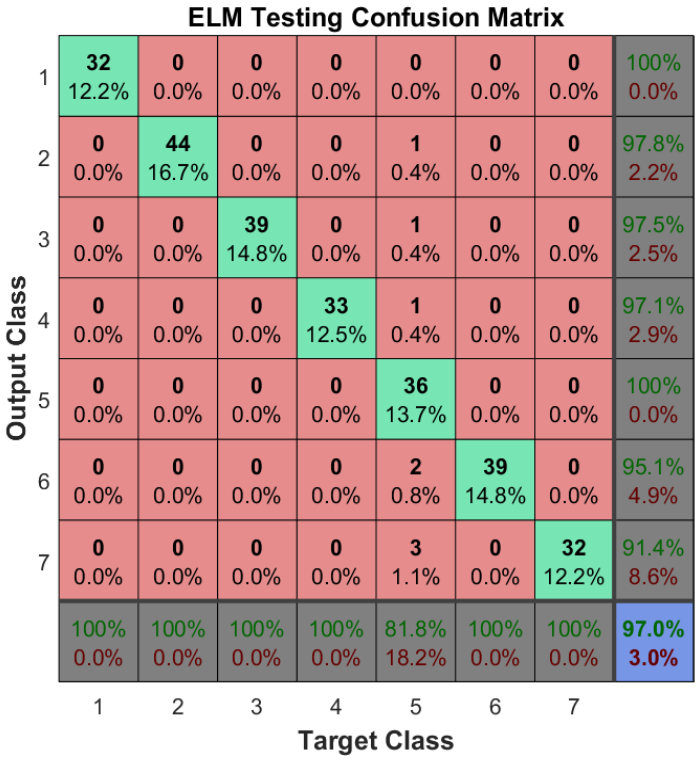

Figure IV. 20: 7-class confusion matrix for ELM and BBA

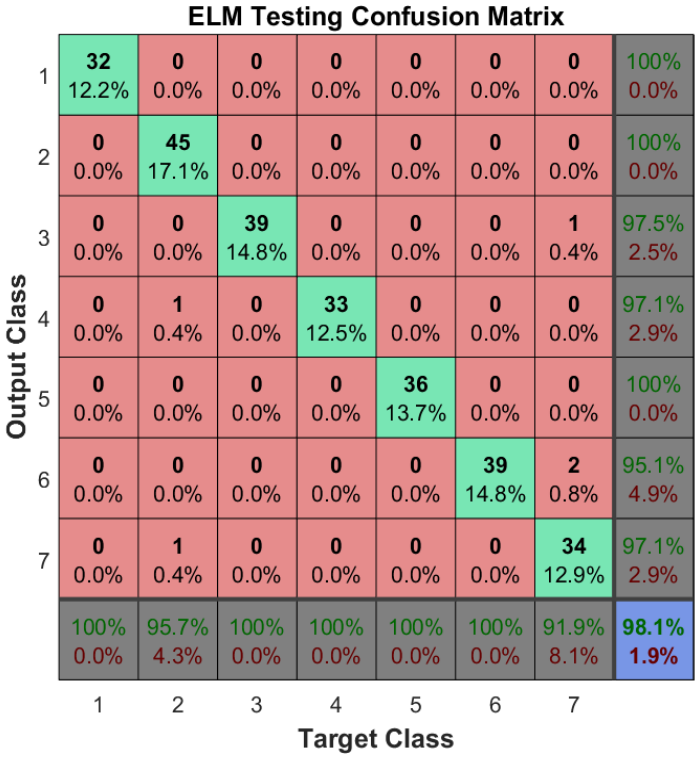

Figure IV. 21: 7-class confusion matrix for ELM and BPSO

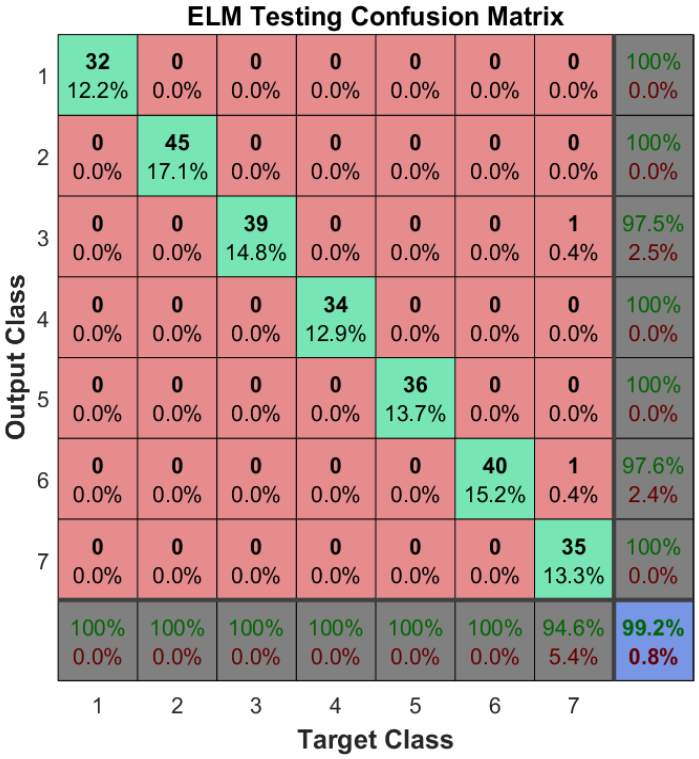

Figure IV. 22: 7-class confusion matrix for ELM and BGWO

IV.2.4. Conclusion

Using IVA, feature selection, and ELM, a novel bearing fault diagnosis approach was developed in this study. The IVA has first been utilized to isolate independent sources from the mixed signals obtained from a three-axis accelerometer, and then statistical parameters are recovered from each component that results. The retrieved features have been optimized using three binary optimisation algorithms: the binary bat algorithm, the binary PSO, and the binary grey wolf optimisation. This stage seeks to select the optimal feature subset, lower the high dimensionality of the collected features, and boost classification accuracy even more. The best feature subset is then used to train the ELM, ANNs, and RF classifiers. This stage seeks to select the optimal feature subset, lower the high dimensionality of the collected features, and boost classification accuracy even more. The best feature subset is then used to train the ELM, ANNs, and RF classifiers. The findings suggest that the best classification outcomes are achieved when IVA, binary grey wolves, and severe learning are combined. Additionally, this combination enables the identification of bearing faults.

GENERAL CONCLUSION

Bearing fault diagnostics using vibration signals is a typical technique in predictive maintenance. The technique entails examining vibration signals created by rotating machinery such as motors, turbines, and bearings to discover any anomalies that may indicate a bearing issue. Sensors such as accelerometers, proximity probes, and velocity transducers can be used to measure vibration signals. Following the recording of the vibration signals, they can be evaluated using various signal processing techniques such as time-domain analysis, frequency-domain analysis, and time-frequency analysis.

These vibration signals need to be analysed. Among many techniques, feature extraction is one effective technique. The primary idea behind feature extraction is to convert raw bearing vibration signals into a set of features that can be utilised to detect the existence of problems. These characteristics are often determined from the signals' time-domain, frequency-domain, or time-frequency domain representations.

Once the features are extracted from the vibration signal, various machine-learning algorithms can be used to classify the fault type and severity based on the extracted features. Many machine learning algorithms have been used in bearing fault diagnoses such as random forests, support vector machines, and artificial neural networks. These algorithms can improve bearing fault diagnosis by providing early detection, and accurate diagnosis, reducing maintenance costs, improving reliability, and increasing safety. The accuracy of the classification depends on the quality of the features and the choice of machine learning technique.

In conclusion, the importance of ML in bearing fault diagnosis cannot be overstated. ML algorithms can improve the accuracy and speed of fault diagnosis, reduce costs, and enable predictive maintenance, leading to more efficient and reliable equipment operations.

REFERENCES

[01] Maqsood, M., Mehmood, I., Kharel, R., Muhammad, K., Lee, J. and Alnumay, W.S.J.H.C.C.I.S., 2021. Exploring the role of deep learning in industrial applications: a case study on coastal crane casting recognition. vol, 11, pp.1-14.

[02] Arunraj, N.S. and Maiti, J., 2009. A methodology for overall consequence modeling in chemical industry. Journal of hazardous materials, 169(1-3), pp.556-574.

[03] Zhu, X., Zhong, C. and Zhe, J., 2017. Lubricating oil conditioning sensors for online machine health monitoring–A review. Tribology International, 109, pp.473-484.

[04] Bode, G., Thul, S., Baranski, M. and Müller, D., 2020. Real-world application of machine-learning-based fault detection trained with experimental data. Energy, 198, p.117323.

[05] Lee, J., Wu, F., Zhao, W., Ghaffari, M., Liao, L. and Siegel, D., 2014. Prognostics and health management design for rotary machinery systems—Reviews, methodology and applications. Mechanical systems and signal processing, 42(1-2), pp.314-334.

[06] PECHT, S.K.M., 2010. Modeling approaches for prognostics and health management of electronics. International Journal of Performability Engineering, 6(5), p.467.

[07] Malla, C. and Panigrahi, I., 2019. Review of condition monitoring of rolling element bearing using vibration analysis and other techniques. Journal of Vibration Engineering & Technologies, 7, pp.407-414.

[08] Márquez, F.P.G., Tobias, A.M., Pérez, J.M.P. and Papaelias, M., 2012. Condition monitoring of wind turbines: Techniques and methods. Renewable energy, 46, pp.169-178.

[09] Popescu, T.D., Aiordachioaie, D. and Culea-Florescu, A., 2022. Basic tools for vibration analysis with applications to predictive maintenance of rotating machines: an overview. The International Journal of Advanced Manufacturing Technology, pp.1-17.

[10] Zmarzły, P., 2020. Multi-dimensional mathematical wear models of vibration generated by rolling ball bearings made of AISI 52100 bearing steel. Materials, 13(23), p.5440.

[11] Toma, R.N., Prosvirin, A.E. and Kim, J.M., 2020. Bearing fault diagnosis of induction motors using a genetic algorithm and machine learning classifiers. Sensors, 20(7), p.1884.

[12] Karabacak, Y.E., Özmen, N.G. and Gümüşel, L., 2022. Intelligent worm gearbox fault diagnosis under various working conditions using vibration, sound and thermal features. Applied Acoustics, 186, p.108463.

[13] Mohanty, J.K., Dash, P.R. and Pradhan, P.K., 2020. FMECA analysis and condition monitoring of critical equipments in super thermal power plant. International Journal of System Assurance Engineering and Management, 11(3), pp.583-599.

[14] Yongbo, L.I., Xiaoqiang, D.U., Fangyi, W.A.N., Xianzhi, W.A.N.G. and Huangchao, Y.U., 2020. Rotating machinery fault diagnosis based on convolutional neural network and infrared thermal imaging. Chinese Journal of Aeronautics, 33(2), pp.427-438.

[15] S. Fang et W. Zijie. Rolling bearing fault diagnosis based on wavelet packet and rbf neural network. Proceedings of the 26th Chinese Control Conference July 26-31, , Zhangjiajie, Hunan, China (2007).

[16] S. Fu, K. Liu, Y. Xu, et Y. Liu. Rolling bearing diagnosing method based on time domain analysis and adaptive fuzzy means clustering. Shock and Vibration (2016).

[17] Y. Lei, Z. He, Y. Zi et Q. Hu . Fault diagnosis of rotating machinery based on multiple anfis combination with ga. Mechanical Systems and Signal Processing 21, pages 2280–2294 (2007).

[18] C. Shen, D. Wang, F. Kong, et P.W. Tse. Fault diagnosis of rotating machinery based on the statistical parameters of wavelet packet paving and a generic support vector regressive classifier. Measurement 46, pages 1551–1564 (2013).

[19] I. Rashedul, A. K. Sheraz et K. Jong-Myon . Maximum class separability-based discriminant feature selection using a ga for reliable fault diagnosis of induction motors. Advanced Intelligent Computing Theories and Applications pages 526–537 (2015).

[20] R. Ziani, A. Felkaoui et R. Zegadi . Bearing fault diagnosis using multiclass support vector machines with binary particle swarm optimization and regularized fishers criterion. Journal of Intelligent Manufacturing 28(9), pages 405–417 (2017).

[21] Fernandez, A. (2020) Rolling element bearing components and failing frequencies, Power. Power-MI. Available at: https://power-mi.com/content/rolling-element-bearing-components-and-failing-frequencies (Accessed: January 31, 2023).

[22] Graney, Brian & Starry, Ken. (2012). Rolling Element Bearing Analysis. Materials Evaluation. 70. 78-85.

[23] Case Western Reserve University Bearing Data Center, 2009, https://csegroups.case.edu/bearingdatacenter/pages/welcome-case-western-reserve-university-bearing-data-center-website

[24] Daga A, Fasana A, Marchesiello S, Garibaldi L (2019) The politecnico di torino rolling bearing test rig: description and analysis of open access data. Mech Syst Signal Process 120:252–273. https:// doi.org/10.1016/j.ymssp.2018.10.010

[25] Niculescu, Dan & Ghionea, Adrian & Adrian, Olaru. (2013). Diagnosis and Predictive Maintenance of Machinery and Equipment, by Measuring Vibration. Applied Mechanics and Materials. 325-326. 186-191. 10.4028/www.scientific.net/AMM.325-326.186.

[26] Soliman, Mohammed, Vibration Basics and Machine Reliability Simplified: A Practical Guide to Vibration Analysis (October 12, 2020). Book, ISBN-13: 979-8696217345 , Available at SSRN: https://ssrn.com/abstract=3752458

[27] Liu, H., Han, S., Yang, J., & Liu, S. (2017). Improving the weak feature extraction by adaptive stochastic resonance in cascaded piecewise-linear system and its application in bearing fault detection. In Journal of Vibroengineering (Vol. 19, Issue 4, pp. 2506–2520). JVE International Ltd. https://doi.org/10.21595/jve.2017.17727

[28] Wang, Y., Li, Y., & Zhang, X. (2020). Review of bearing fault diagnosis using acoustic emission technique. Journal of Sound and Vibration, 503, 114510. https://doi.org/10.1016/j.jsv.2019.114510

[29] Ferrando Chacon, J. L. et al. (2016) 'An experimental study on the applicability of acoustic emission for wind turbine gearbox health diagnosis', Low Frequency Noise & Vibration. SAGE Publications, 35(1), pp. 64–76. doi: 10.1177/0263092316628401.

[30] Noori, N. S. et al. (2022) 'Data driven seal wear classifications using acoustic emissions and Artificial Neural Networks', PHM Society European Conference. PHM Society, 7(1), pp. 368–375. doi: 10.36001/phme.2022.v7i1.3327.

[31] Tian, J., Liu, L., Zhang, F., Ai, Y., Wang, R. and Fei, C., 2019. Multi-domain entropy-random forest method for the fusion diagnosis of inter-shaft bearing faults with acoustic emission signals. Entropy, 22(1), p.57. doi: 10.3390/e22010057.

[32] Ali, S.M., Hui, K.H., Hee, L.M. and Leong, M.S., 2018. Automated valve fault detection based on acoustic emission parameters and support vector machine. Alexandria engineering journal, 57(1), pp.491-498.

[33] Kim, JaeYoung, and Jong-Myon Kim. 2020. "Bearing Fault Diagnosis Using Grad-CAM and Acoustic Emission Signals" Applied Sciences 10, no. 6: 2050. https://doi.org/10.3390/app10062050.

[34] Ibarra-Zarate, D., Tamayo-Pazos, O. & Vallejo-Guevara, A. Bearing fault diagnosis in rotating machinery based on cepstrum pre-whitening of vibration and acoustic emission. Int J Adv Manuf Technol 104, 4155–4168 (2019). https://doi.org/10.1007/s00170-019-04171-6

[35] Mehta, A., Goyal, D., Choudhary, A., Pabla, B.S. and Belghith, S., 2021. Machine learning-based fault diagnosis of self-aligning bearings for rotating machinery using infrared thermography. Mathematical Problems in Engineering, 2021, pp.1-15.

[36] Bagavathiappan, S., Lahiri, B.B., Saravanan, T., Philip, J. and Jayakumar, T., 2013. Infrared thermography for condition monitoring–A review. Infrared Physics & Technology, 60, pp.35-55.

[37] Senanayaka, J.S.L., Kandukuri, S.T., Van Khang, H. and Robbersmyr, K.G., 2017, April. Early detection and classification of bearing faults using support vector machine algorithm. In 2017 IEEE Workshop on Electrical Machines Design, Control and Diagnosis (WEMDCD) (pp. 250-255). IEEE.

[38] Laborda, F., Gimenez-Ingalaturre, A.C., Bolea, E. and Castillo, J.R., 2019. Single particle inductively coupled plasma mass spectrometry as screening tool for detection of particles. Spectrochimica Acta Part B: Atomic Spectroscopy, 159, p.105654.

[39] Jin, X., Chen, Y., Wang, L., Han, H. and Chen, P., 2021. Failure prediction, monitoring and diagnosis methods for slewing bearings of large-scale wind turbine: A review. Measurement, 172, p.108855.

[40] Saucedo-Dorantes, J.J., Delgado-Prieto, M., Ortega-Redondo, J.A., Osornio-Rios, R.A. and Romero-Troncoso, R.D.J., 2016. Multiple-fault detection methodology based on vibration and current analysis applied to bearings in induction motors and gearboxes on the kinematic chain. Shock and Vibration, 2016.

[41] Singleton, R.K., Strangas, E.G. and Aviyente, S., 2016. The use of bearing currents and vibrations in lifetime estimation of bearings. IEEE Transactions on Industrial Informatics, 13(3), pp.1301-1309.

[42] Mehrjou, M.R., Mariun, N., Marhaban, M.H. and Misron, N., 2011. Rotor fault condition monitoring techniques for squirrel-cage induction machine—A review. Mechanical Systems and Signal Processing, 25(8), pp.2827-2848.

[43] Barai, V., Ramteke, S.M., Dhanalkotwar, V., Nagmote, Y., Shende, S. and Deshmukh, D., 2022, October. Bearing fault diagnosis using signal processing and machine learning techniques: A review. In IOP Conference Series: Materials Science and Engineering (Vol. 1259, No. 1, p. 012034). IOP Publishing.

[44] Liang, X., Ali, M.Z. and Zhang, H., 2019. Induction motors fault diagnosis using finite element method: a review. IEEE Transactions on Industry Applications, 56(2), pp.1205-1217.

[45] Mbaabu, Onesmus. "Understanding Pattern Recognition in Machine Learning." *Engineering Education (EngEd) Program | Section*, 31 Mar. 2021, www.section.io/engineering-education/understanding-pattern-recognition-in-machine-learning.

[46] Liang, Y., Harris, J., Naqvi, S.M., Chen, G. and Chambers, J.A., 2014. Independent vector analysis with a generalized multivariate Gaussian source prior for frequency domain blind source separation. Signal Processing, 105, pp.175-184.

[47] Kitamura, D., Ono, N., Sawada, H., Kameoka, H. and Saruwatari, H., 2016. Determined blind source separation unifying independent vector analysis and nonnegative matrix factorization. IEEE/ACM Transactions on Audio, Speech, and Language Processing, 24(9), pp.1626-1641.

[48] Brendel, A., Haubner, T. and Kellermann, W., 2020. A unified probabilistic view on spatially informed source separation and extraction based on independent vector analysis. IEEE Transactions on Signal Processing, 68, pp.3545-3558.

[49] Uddin Z, Nebhen J, Altaf M, Orakzai FA (2021) Independent vector analysis inspired amateur drone detection through acoustic signals. IEEE Access 9:63456–63462. https:// doi. org/ 10. 1109/ ACCESS. 2021. 30749 66

[50] Lee IT (2009) Machine learning algorithms for independent vector analysis and blind source separation. PhD thesis, USA. AAI3373454

[51] Anderson M, Adali T, Li X-L (2012) Joint blind source separation with multivariate gaussian model: algorithms and performance analysis. Trans Sig Proc 60(4):1672–1683. https:// doi. org/ 10. 1109/ TSP. 2011. 21818 36

[52] SINGH, AMAN, editor. "Types of Pattern Recognition Algorithms." Types of Pattern Recognition Algorithms - Global Tech Council, 15 Feb. 2021, www.globaltechcouncil.org/machine-learning/types-of-pattern-recognition-algorithms.

[53] Zarei, J. (2012) 'Induction motors bearing fault detection using pattern recognition techniques', Expert systems with applications. Elsevier BV, 39(1), pp. 68–73. doi: 10.1016/j.eswa.2011.06.042.

[54] Chen, P. (2001) 'Bearing condition monitoring and fault diagnosis'. University of Calgary. DOI: 10.11575/PRISM/23398.

[55] Samanta, B., Al-Balushi, K.R. and Al-Araimi, S.A., 2004. Bearing fault detection using artificial neural networks and genetic algorithm. EURASIP Journal on Advances in Signal Processing, 2004, pp.1-12

[56] Wang, Z., Yao, L., Cai, Y. and Zhang, J., 2020. Mahalanobis semi-supervised mapping and beetle antennae search based support vector machine for wind turbine rolling bearings fault diagnosis. Renewable Energy, 155, pp.1312-1327.

[57] He, C., Wu, T., Gu, R., Jin, Z., Ma, R. and Qu, H., 2021. Rolling bearing fault diagnosis based on composite multiscale permutation entropy and reverse cognitive fruit fly optimization algorithm–extreme learning machine. Measurement, 173, p.108636.

[58] Roy, S.S., Dey, S. and Chatterjee, S., 2020. Autocorrelation aided random forest classifier-based bearing fault detection framework. IEEE Sensors Journal, 20(18), pp.10792-10800.

[59] Parajuli, N., Sreenivasan, N., Bifulco, P., Cesarelli, M., Savino, S., Niola, V., Esposito, D., Hamilton, T.J., Naik, G.R., Gunawardana, U. and Gargiulo, G.D., 2019. Real-time EMG based pattern recognition control for hand prostheses: A review on existing methods, challenges and future implementation. Sensors, 19(20), p.4596.

[60] Sun, M. et al. (2022) 'Stack autoencoder transfer learning algorithm for bearing fault diagnosis based on class separation and domain fusion', IEEE transactions on industrial electronics (1982). Institute of Electrical and Electronics Engineers (IEEE), 69(3), pp. 3047–3058. doi: 10.1109/tie.2021.3066933.

[61] Choudhary, A., Goyal, D. and Letha, S. S. (2021) 'Infrared thermography-based fault diagnosis of induction motor bearings using machine learning', IEEE sensors journal. Institute of Electrical and Electronics Engineers (IEEE), 21(2), pp. 1727–1734. doi: 10.1109/jsen.2020.3015868.

[62] Hou, J. et al. (2020) 'A novel intelligent method for bearing fault diagnosis based on EEMD permutation entropy and GG clustering', Applied sciences (Basel, Switzerland). MDPI AG, 10(1), p. 386. doi: 10.3390/app10010386.

[63] Zhang, X., Zhao, B. and Lin, Y. (2021) 'Machine learning based bearing fault diagnosis using the case western reserve university data: A review', IEEE access: practical innovations, open solutions. Institute of Electrical and Electronics Engineers (IEEE), 9, pp. 155598–155608. doi: 10.1109/access.2021.3128669.

[64] Sun, G. et al. (2022) 'Rolling bearing fault diagnosis based on time-frequency compression fusion and residual time-frequency mixed attention network', Applied sciences (Basel, Switzerland). MDPI AG, 12(10), p. 4831. doi: 10.3390/app12104831.

[65] Toma, R. N., Prosvirin, A. E. and Kim, J.-M. (2020) 'Bearing fault diagnosis of induction motors using a genetic algorithm and machine learning classifiers', Sensors (Basel, Switzerland). MDPI AG, 20(7), p. 1884. doi: 10.3390/s20071884.

[66] Li, J. et al. (2019) 'Research on rolling bearing fault diagnosis based on multi-dimensional feature extraction and evidence fusion theory', Royal Society open science. The Royal Society, 6(2), p. 181488. doi: 10.1098/rsos.181488.

[67] Khan, S. A. and Kim, J.-M. (2016) 'Automated bearing fault diagnosis using 2D analysis of vibration acceleration signals under variable speed conditions', Shock and vibration. Hindawi Limited, 2016, pp. 1–11. doi: 10.1155/2016/8729572.

[68] Li, W. et al. (2016) 'Bearing fault diagnosis based on spectrum images of vibration signals', Measurement science & technology. IOP Publishing, 27(3), p. 035005. doi: 10.1088/0957-0233/27/3/035005.

[69] Jiang, Y. and Xie, J. (2022) 'VMD–RP–CSRN based fault diagnosis method for rolling bearings', Electronics. MDPI AG, 11(23), p. 4046. doi: 10.3390/electronics11234046.

[70] Kumar, B. (2018) 'Feature extraction using principal component analysis and discrete wavelet transform for image classification', International journal of computer sciences and engineering. ISROSET: International Scientific Research Organization for Science, Engineering and Technology, 6(8), pp. 582–586. doi: 10.26438/ijcse/v6i8.582586.

[71] Forecast Global and Taylor, S. (2022) Feature engineering, Corporate Finance Institute. Available at: https://corporatefinanceinstitute.com/resources/data-science/feature-engineering/ (Accessed: February 3, 2023).

[72] Dhahri, H., Al Maghayreh, E., Mahmood, A., Elkilani, W. and Faisal Nagi, M., 2019. Automated breast cancer diagnosis based on machine learning algorithms. Journal of healthcare engineering, 2019.

[73] Bouktif, S. et al. (2018) 'Optimal deep learning LSTM model for electric load forecasting using feature selection and genetic algorithm: Comparison with machine learning approaches', Energies. MDPI AG, 11(7), p. 1636. doi: 10.3390/en11071636.

[74] Yan, X., Liu, Y. and Jia, M., 2019. A feature selection framework-based multiscale morphological analysis algorithm for fault diagnosis of rolling element bearing. IEEE Access, 7, pp.123436-123452.

[75] Legendre, P., & Legendre, D. L. F. J. (2012). Numerical Ecology: Volume 24 (3rd ed.). Elsevier Science.

[76] Zhang, C. et al. (2014) 'Periodic impulsive fault feature extraction of rotating machinery using dual-tree rational dilation complex wavelet transform', Journal of manufacturing science and engineering. ASME International, 136(5), p. 051011. doi: 10.1115/1.4027839.

[77] Thelaidjia, T., 2017. Conception d'un système intelligent de diagnostic des défauts dans les machines tournantes. PhD. GUELMA: Université 8 mai 1945 - GUELMA.

[78] Rai, V. (2021) Skewness And Kurtosis In Machine Learning - Vivek Rai. Available at: https://vivekrai1011.medium.com/skewness-and-kurtosis-in-machine-learning-c19f79e2d7a5.

[79] Qu, Aaron & Bechhoefer, Eric & He, David & Zhu, Junda. (2013). A New Acoustic Emission Sensor Based Gear Fault Detection Approach. International Journal of Prognostics and Health Management. 4. 10.36001/ijphm.2013.v4i3.2141.

[80] Abou El Anouar, Bouchra. (2017). A comparative experimental study of different methods in detection and monitoring bearing defects. International Journal of Advanced Scientific and Technical Research. volume 1. 409-423.

[81] Hui, K.H., Ooi, C.S., Lim, M.H., Leong, M.S. and Al-Obaidi, S.M., 2017. An improved wrapper-based feature selection method for machinery fault diagnosis. PloS one, 12(12), p.e0189143.

[82] Z. Huo, Y. Zhang, P. Francq, L. Shu and J. Huang, "Incipient Fault Diagnosis of Roller Bearing Using Optimized Wavelet Transform Based Multi-Speed Vibration Signatures," in IEEE Access, vol. 5, pp. 19442-19456, 2017, doi: 10.1109/ACCESS.2017.2661967.

[83] Liang, L. et al. (2019) 'Feature extraction of impulse faults for vibration signals based on sparse non-negative tensor factorization', Applied sciences (Basel, Switzerland). MDPI AG, 9(18), p. 3642. doi: 10.3390/app9183642.

[84] Dequan, Yu & Wenbo, Wu & Hongyong, Fu & Yang, Wang & Ke, Wang. (2019). Fault Diagnosis Technology of Rolling Bearing in Space Station. Journal of Physics: Conference Series. 1325. 012015. 10.1088/1742-6596/1325/1/012015.

[85] Meng, Z., Shi, G. and Wang, F. (2020) 'Vibration response and fault characteristics analysis of gear based on time-varying mesh stiffness', Mechanism and machine theory. Elsevier BV, 148(103786), p. 103786. doi: 10.1016/j.mechmachtheory.2020.103786.

[86] Wang, X. et al. (2015) 'Bearing fault diagnosis based on statistical locally linear embedding', Sensors (Basel, Switzerland). MDPI AG, 15(7), pp. 16225–16247. doi: 10.3390/s150716225.

[87] Altaf, M. et al. (2022) 'A new statistical features based approach for bearing fault diagnosis using vibration signals', Sensors (Basel, Switzerland). MDPI AG, 22(5), p. 2012. doi: 10.3390/s22052012.

[88] Rai, V. (2021) Skewness and kurtosis in machine learning - Vivek Rai, Medium. Available at: https://vivekrai1011.medium.com/skewness-and-kurtosis-in-machine-learning-c19f79e2d7a5 (Accessed: February 10, 2023).

[89] Experimental study based on acoustic emission and vibration signals to detect the presence of defects in a rolling element bearing., hazim alsadoon, sedat baysec, necle togun international journal of scientific research : volume-8 | issue-9 | september-2019

[90] Xu M., Shang P. Analysis of financial time series using multiscale entropy based on skewness and kurtosis. Phys. A Stat. Mech. Its Appl. 2018;490:1543–1550. doi: 10.1016/j.physa.2017.08.136.

[91] Bhende, A.R., Awari, G.K. and Untawale, S.P., 2014. Comprehensive bearing condition monitoring algorithm for incipient fault detection using acoustic emission. Jurnal Tribologi, 2, pp.1-30.

[92] Bardes, A., Ponce, J. and LeCun, Y., 2021. Vicreg: Variance-invariance-covariance regularization for self-supervised learning. arXiv preprint arXiv:2105.04906.

[93] Nestor, B., McDermott, M.B., Boag, W., Berner, G., Naumann, T., Hughes, M.C., Goldenberg, A. and Ghassemi, M., 2019, October. Feature robustness in non-stationary health records: caveats to deployable model performance in common clinical machine learning tasks. In Machine Learning for Healthcare Conference (pp. 381-405). PMLR.

[94] Wong, L.P., Alias, H., Wong, P.F., Lee, H.Y. and AbuBakar, S., 2020. The use of the health belief model to assess predictors of intent to receive the COVID-19 vaccine and willingness to pay. Human vaccines & immunotherapeutics, 16(9), pp.2204-2214.

[95] Christopher Schweitzer, Jan 1, 2001, Hearingreview.com. Available at: https://hearingreview.com/hearing-products/hearing-aids/ite/temporal-pattern-processing-of-hearing-instruments (Accessed: February 7, 2023).

[96] Khan, S. I. and Ahmed, V. (2016) "Study of effectiveness of stockwell transform for detection of coronary artery disease from heart sounds," in 2016 2nd International Conference on Contemporary Computing and Informatics (IC3I). IEEE.

[97] Hasan, M.J., Islam, M.M. and Kim, J.M., 2019. Acoustic spectral imaging and transfer learning for reliable bearing fault diagnosis under variable speed conditions. Measurement, 138, pp.620-631.

[98] AdSimuTec - Structural mechanics: PSD Power spectral density analysis (no date) Adsimutec.com. Available at: https://adsimutec.com/en/fem-cfd-simulation-engineering-services/structural-mechanics/psd-power-spectral-density-analysis (Accessed: February 10, 2023).

[99] Wescoat, E., Mears, L., Goodnough, J. and Sims, J., 2020. Frequency energy analysis in detecting rolling bearing faults. Procedia Manufacturing, 48, pp.980-991.

[100] Pishro-Nik, H. (2014) Introduction to probability, statistics, and random processes. Kappa Research.

[101] Arun, P., Lincon, S. A. and Prabhakaran, N. (2017) "Detection and characterization of bearing faults from the frequency domain features of vibration," IETE journal of research, pp. 1–14. doi: 10.1080/03772063.2017.1369369

[102] Shen Fei, et al."The application of spectrophic migration in the diagnosis of alteration bearing failure." Instrument Instrument Journal 40.05(2019):99-108. doi:10.19650/j.cnki.cjsi.J1905038.

[103] Peeters, G. "A Large Set of Audio Features for Sound Description (Similarity and Classification) in the CUIDADO Project." Technical Report; IRCAM: Paris, France, 2004

[104] H. M. Liu, C. Lu, and J. C. Zhang, "Bearing fault diagnosis based on Shannon entropy and wavelet package decomposition," Vibroengineering PROCEDIA, Vol. 4, pp. 223–228, Nov. 2014.

[105] Ni, Q. et al. (2017) 'A case study of sample entropy analysis to the fault detection of bearing in wind turbine', Case studies in engineering failure analysis. Elsevier BV, 9, pp. 99–111. doi: 10.1016/j.csefa.2017.10.002.

[106] Wu, S.-D. et al. (2012) 'Bearing fault diagnosis based on multiscale permutation entropy and support vector machine', Entropy (Basel, Switzerland). MDPI AG, 14(8), pp. 1343–1356. doi: 10.3390/e14081343.

[107] Mai, N.-D., Lee, B.-G. and Chung, W.-Y. (2021) 'Affective computing on machine learning-based emotion recognition using a self-made EEG device', Sensors (Basel, Switzerland). MDPI AG, 21(15), p. 5135. doi: 10.3390/s21155135.

[108] M. Gashnikov, "Decision-Tree-Based Interpolation for Multidimensional Signal Compression," 2020 8th International Symposium on Digital Forensics and Security (ISDFS), Beirut, Lebanon, 2020, pp. 1-5, doi: 10.1109/ISDFS49300.2020.9116450.

[109] Ciaburro, G. (2022) 'Machine fault detection methods based on machine learning algorithms: A review', Mathematical biosciences and engineering: MBE. American Institute of Mathematical Sciences (AIMS), 19(11), pp. 11453–11490. doi: 10.3934/mbe.2022534.

[110] Wasif, H., Fahimi, F., Brown, D. and Axel-Berg, L., 2012, March. Application of multi-fuzzy system for condition monitoring of liquid filling machines. In 2012 IEEE International Conference on Industrial Technology (pp. 906-912). IEEE.

[111] Zhang, M., Schwemmer, M.A., Ting, J.E. et al. Extracting wavelet based neural features from human intracortical recordings for neuroprosthetics applications. Bioelectron Med 4, 11 (2018). https://doi.org/10.1186/s42234-018-0011-x

[112] Morgan, D., 1996. Multiresolution Signal Analysis and Wavelet Decomposition. Embedded Systems Programming, 9, pp.30-49.

[113] Sadowsky, J., 1994. The continuous wavelet transform: A tool for signal investigation and understanding. Johns Hopkins APL Technical Digest, 15, pp.306-306.

[114] 'The discrete wavelet transform' (2002) in The Illustrated Wavelet Transform Handbook. Taylor & Francis. doi: 10.1201/9781420033397.ch3.

[115] Frid, A. and Manevitz, L.M., 2018. Features and machine learning for correlating and classifying between brain areas and dyslexia. arXiv preprint arXiv:1812.10622.

[116] Phinyomark, A., Phukpattaranont, P. and Limsakul, C., 2012. Feature reduction and selection for EMG signal classification. Expert systems with applications, 39(8), pp.7420-7431.

[117] Y. Wang, X. Zhang, J. Zhao and C. He, "Pattern recognition of electromyography applied to Exoskeleton Robot," 2010 3rd International Congress on Image and Signal Processing, Yantai, China, 2010, pp. 3802-3805, doi: 10.1109/CISP.2010.5646759.

[118] Wang, Y., Zhang, X., Zhao, J. and He, C., 2010, October. Pattern recognition of electromyography applied to exoskeleton robot. In 2010 3rd International Congress on Image and Signal Processing (Vol. 8, pp. 3802-3805). IEEE.

[119] Brito, L. C. et al. (2021) 'Fault detection of bearing: An unsupervised machine learning approach exploiting feature extraction and dimensionality reduction', Informatics (MDPI). MDPI AG, 8(4), p. 85. doi: 10.3390/informatics8040085.

[120] B. Ghojogh, M.N. Samad, S.A. Mashhadi, T. Kapoor, W. Ali, F. Karray, M. Crowley, Feature selection and feature extraction in pattern analysis: A literature review, 2019, arXiv preprint arXiv:1905.02845.

[121] Zhao, H., Zheng, J., Xu, J. and Deng, W., 2019. Fault diagnosis method based on principal component analysis and broad learning system. IEEE Access, 7, pp.99263-99272.

[122] Shchurenkova, E., 2017. Dimension reduction using Independent Component Analysis with an application in business psychology (Doctoral dissertation, University of British Columbia).

[123] Hyvärinen, A. and Oja, E. (2000) 'Independent component analysis: algorithms and applications', Neural networks: the official journal of the International Neural Network Society. Elsevier BV, 13(4–5), pp. 411–430. doi: 10.1016/s0893-6080(00)00026-5.

[124] Xie, X. et al. (2009) 'Independent Component Analysis', in Encyclopedia of Biometrics. Boston, MA: Springer US, pp. 735–741. doi: 10.1007/978-0-387-73003-5_305.

[125] Ghosh, J. and Shuvo, S.B., 2019, July. Improving classification model's performance using linear discriminant analysis on linear data. In 2019 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT) (pp. 1-5). IEEE.

[126] Gareth, J., Daniela, W., Trevor, H. and Robert, T., 2013. An introduction to statistical learning: with applications in R. Spinger.

[127] Mao, W., He, J., Li, Y. and Yan, Y., 2017. Bearing fault diagnosis with auto-encoder extreme learning machine: A comparative study. Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science, 231(8), pp.1560-1578.

[128] Pramoditha, Rukshan. "11 Dimensionality Reduction Techniques You Should Know in 2021." Medium, 2021, towardsdatascience.com/11-dimensionality-reduction-techniques-you-should-know-in-2021-dcb9500d388b.

[129] Simplilearn (2022) What are Autoencoders? Introduction to Autoencoders in Deep Learning, Simplilearn.com. Simplilearn. Available at: https://www.simplilearn.com/tutorials/deep-learning-tutorial/what-are-autoencoders-in-deep-learning (Accessed: February 12, 2023).

[130] Chandrashekar, G. and Sahin, F. (2014) 'A survey on feature selection methods', Computers & electrical engineering: an international journal. Elsevier BV, 40(1), pp. 16–28. doi: 10.1016/j.compeleceng.2013.11.024.

[131] Yang, Y., Liao, Y., Meng, G. and Lee, J., 2011. A hybrid feature selection scheme for unsupervised learning and its application in bearing fault diagnosis. Expert Systems with Applications, 38(9), pp.11311-11320.

[132] Feature selection techniques in machine learning (2021) GeeksforGeeks. Available at: https://www.geeksforgeeks.org/feature-selection-techniques-in-machine-learning/ (Accessed: February 12, 2023).

[133] Khaire, U. M. and Dhanalakshmi, R. (2022) 'Stability of feature selection algorithm: A review', Journal of King Saud University - Computer and Information Sciences. Elsevier BV, 34(4), pp. 1060–1073. doi: 10.1016/j.jksuci.2019.06.012.

[134] Mendes Junior, J. J. A. et al. (2020) 'Feature selection and dimensionality reduction: An extensive comparison in hand gesture classification by sEMG in eight channels armband approach', Biomedical signal processing and control. Elsevier BV, 59(101920), p. 101920. doi: 10.1016/j.bspc.2020.101920.

[135] Yuan, G. and Ghanem, B. (2016) 'Binary optimization via Mathematical Programming with Equilibrium Constraints'. arXiv. doi: 10.48550/ARXIV.1608.04425.

[136] Mirjalili, S., Mirjalili, S. M. and Yang, X.-S. (2014) 'Binary bat algorithm', Neural computing & applications. Springer Science and Business Media LLC, 25(3–4), pp. 663–681. doi: 10.1007/s00521-013-1525-5.

[137] Diebold, C. A., Salles, A. and Moss, C. F. (2020) 'Adaptive echolocation and flight behaviors in bats can inspire technology innovations for sonar tracking and interception', Sensors (Basel, Switzerland). MDPI AG, 20(10), p. 2958. doi: 10.3390/s20102958.

[138] Huang, J. and Ma, Y. (2020) 'Bat algorithm based on an integration strategy and Gaussian distribution', Mathematical problems in engineering. Hindawi Limited, 2020, pp. 1–22. doi: 10.1155/2020/9495281.

[139] Chawla, M. and Duhan, M. (2015) 'Bat algorithm: A survey of the state-of-the-art', Applied artificial intelligence: AAI. Informa UK Limited, 29(6), pp. 617–634. doi: 10.1080/08839514.2015.1038434.

[140] Nakamura, R.Y., Pereira, L.A., Costa, K.A., Rodrigues, D., Papa, J.P. and Yang, X.S., 2012, August. BBA: a binary bat algorithm for feature selection. In 2012 25th SIBGRAPI conference on graphics, patterns and images (pp. 291-297). IEEE.

[141] Zhao, D., He, Y. A novel binary bat algorithm with chaos and Doppler effect in echoes for analog fault diagnosis. Analog Integr Circ Sig Process 87, 437–450 (2016). https://doi.org/10.1007/s10470-016-0728-y

[142] Chuang, L.-Y., Tsai, S.-W. and Yang, C.-H. (2011) 'Chaotic catfish particle swarm optimization for solving global numerical optimization problems', Applied mathematics and computation. Elsevier BV, 217(16), pp. 6900–6916. doi: 10.1016/j.amc.2011.01.081.

[143] Yang, C.S., Chuang, L.Y., Li, J.C. and Yang, C.H., 2008, June. Chaotic maps in binary particle swarm optimization for feature selection. In 2008 IEEE conference on soft computing in industrial applications (pp. 107-112). IEEE.

[144] Kotsiantis, S. B. (2014) 'RETRACTED ARTICLE: Feature selection for machine learning classification problems: a recent overview', Artificial intelligence review. Springer Science and Business Media LLC, 42(1), pp. 157–157. doi: 10.1007/s10462-011-9230-1.

[145] Li, A.-D., Xue, B. and Zhang, M. (2021) 'Improved binary particle swarm optimization for feature selection with new initialization and search space reduction strategies', Applied soft computing. Elsevier BV, 106(107302), p. 107302. doi: 10.1016/j.asoc.2021.107302.

[146] Venkateswaran, K. et al. (2016) 'Performance analysis of GA and PSO based feature selection techniques for improving classification accuracy in remote sensing images', Indian journal of science and technology. Indian Society for Education and Environment, 9(16). doi: 10.17485/ijst/2016/v9i16/87457.

[147] Qiaorong, Z. and Guochang, G., 2008, September. Path planning based on improved binary particle swarm optimization algorithm. In 2008 IEEE Conference on Robotics, Automation and Mechatronics (pp. 462-466). IEEE.

[148] Too, J., Abdullah, A.R., Mohd Saad, N. and Tee, W., 2019. EMG feature selection and classification using a Pbest-guide binary particle swarm optimization. Computation, 7(1), p.12.

[149] Mirjalili, S., Mirjalili, S. M. and Lewis, A. (2014) 'Grey wolf optimizer', Advances in engineering software (Barking, London, England: 1992). Elsevier BV, 69, pp. 46–61. doi: 10.1016/j.advengsoft.2013.12.007.

[150] Niu, P. et al. (2019) 'The defect of the Grey Wolf optimization algorithm and its verification method', Knowledge-based systems. Elsevier BV, 171, pp. 37–43. doi: 10.1016/j.knosys.2019.01.018.

[151] Chantar, H. et al. (2020) 'Feature selection using binary grey wolf optimizer with elite-based crossover for Arabic text classification', Neural computing & applications. Springer Science and Business Media LLC, 32(16), pp. 12201–12220. doi: 10.1007/s00521-019-04368-6.

[152] Roayaei, M. (2021) 'On the binarization of grey wolf optimizer :A novel binary optimizer algorithm', Research Square. Research Square. doi: 10.21203/rs.3.rs-235817/v1.

[153] Lee, C.-Y., Le, T.-A. and Lin, Y.-T. (2022) 'A feature selection approach hybrid grey wolf and heap-based optimizer applied in bearing fault diagnosis', *IEEE access: practical innovations, open solutions*. Institute of Electrical and Electronics Engineers (IEEE), 10, pp. 56691–56705. doi: 10.1109/access.2022.3177735.

[154] Shi, H. et al. (2021) 'Intelligent fault identification for rolling bearings fusing average refined composite multiscale dispersion entropy-assisted feature extraction and SVM with multi-strategy enhanced swarm optimization', Entropy (Basel, Switzerland), 23(5). doi: 10.3390/e23050527.

[155] Too, J. et al. (2018) 'A new competitive binary grey wolf optimizer to solve the feature selection problem in EMG signals classification', Computers. MDPI AG, 7(4), p. 58. doi: 10.3390/computers7040058.

[156] Kankar, P. K., Sharma, S. C. and Harsha, S. P. (2011) 'Fault diagnosis of ball bearings using machine learning methods', Expert systems with applications. Elsevier BV, 38(3), pp. 1876–1886. doi: 10.1016/j.eswa.2010.07.119.

[157] Ciaburro, G. (2022) 'Machine fault detection methods based on machine learning algorithms: A review', Mathematical biosciences and engineering: MBE. American Institute of Mathematical Sciences (AIMS), 19(11), pp. 11453–11490. doi: 10.3934/mbe.2022534.

[158] Poulose, Jasmine and Prasad S R, Vishnu and Sadique, Anwar, Ball Bearings Fault Detection with Machine Learning of Vibration Signals (March 3, 2022). Proceedings of the International Conference on Aerospace & Mechanical Engineering (ICAME 21), Available at: http://dx.doi.org/10.2139/ssrn.4048616

[159] Analytics Vidhya. (2021). ANN for Data Science | Basics Of Artificial Neural Network. [online] Available at: https://www.analyticsvidhya.com/blog/2021/07/understanding-the-basics-of-artificial-neural-network-ann/.

[160] Wang, X., Liu, Y. and Xin, H. (2021) 'Bond strength prediction of concrete-encased steel structures using hybrid machine learning method', Structures. Elsevier BV, 32, pp. 2279–2292. doi: 10.1016/j.istruc.2021.04.018.

[161] Amazon Web Services, Inc. (n.d.). What is a Neural Network? AI and ML Guide - AWS. [online] Available at: https://aws.amazon.com/what-is/neural-network/.

[162] Pyo, S. et al. (2017) 'Predictability of machine learning techniques to forecast the trends of market index prices: Hypothesis testing for the Korean stock markets', PloS one. Public Library of Science (PLoS), 12(11), p. e0188107. doi: 10.1371/journal.pone.0188107.

[163] Martinez, M. et al. (2011) 'C i the use of artificial neural networks for identification and location of defective insulators in power lines through current transformers', in. Unpublished. doi: 10.13140/2.1.3695.4884.

[164] www.xenonstack.com. (n.d.). Artificial Neural Networks Applications and Algorithms. [online] Available at: https://www.xenonstack.com/blog/artificial-neural-network-applications.

[165] Rathi, M. (n.d.). FeedForward Neural Networks. [online] mukulrathi.com. Available at: https://mukulrathi.com/demystifying-deep-learning/feed-forward-neural-network/.

[166] Hadiyan, P.P., Moeini, R. and Ehsanzadeh, E., 2020. Application of static and dynamic artificial neural networks for forecasting inflow discharges, case study: Sefidroud Dam reservoir. Sustainable Computing: Informatics and Systems, 27, p.100401.

[167] Khosla, E. et al. (2018) 'RNNs-RT: Flood based Prediction of Human and Animal deaths in Bihar using Recurrent Neural Networks and Regression Techniques', Procedia computer science. Elsevier BV, 132, pp. 486–497. doi: 10.1016/j.procs.2018.05.001.

[168] Nah, S., Son, S. and Lee, K.M., 2019. Recurrent neural networks with intra-frame iterations for video deblurring. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (pp. 8102-8111).

[169] Golroudbari, A.A. and Sabour, M.H., 2023. End-to-end deep learning framework for real-time inertial attitude estimation using 6dof imu. arXiv preprint arXiv:2302.06037.

[170] Nadeem, M. W. et al. (2020) 'Bone age assessment empowered with deep learning: A survey, open research challenges and future directions', Diagnostics (Basel, Switzerland). MDPI AG, 10(10), p. 781. doi: 10.3390/diagnostics10100781.

[171] Nirthika, R., Manivannan, S., Ramanan, A. and Wang, R., 2022. Pooling in convolutional neural networks for medical image analysis: a survey and an empirical study. Neural Computing and Applications, 34(7), pp.5321-5347.

[172] Zhang, W., Yu, Y., Qi, Y., Shu, F. and Wang, Y., 2019. Short-term traffic flow prediction based on spatio-temporal analysis and CNN deep learning. Transportmetrica A: Transport Science, 15(2), pp.1688-1711.

[173] Danelljan, M., Hager, G., Shahbaz Khan, F. and Felsberg, M., 2015. Convolutional features for correlation filter based visual tracking. In Proceedings of the IEEE international conference on computer vision workshops (pp. 58-66).

[174] Miljković, D., 2017, May. Brief review of self-organizing maps. In 2017 40th international convention on information and communication technology, electronics and microelectronics (MIPRO) (pp. 1061-1066). IEEE.

[175] Abdelsamea, M., Mohamed, M.H. and Bamatraf, M., 2015. An effective image feature classiffication using an improved som. arXiv preprint arXiv:1501.01723.

[176] Sarkar, S., Ejaz, N. and Maiti, J., 2018, March. Application of hybrid clustering technique for pattern extraction of accident at work: a case study of a steel industry. In 2018 4th International Conference on Recent Advances in Information Technology (RAIT) (pp. 1-6). IEEE.

[177] Peek, N., Combi, C., Marin, R. and Bellazzi, R., 2015. Thirty years of artificial intelligence in medicine (AIME) conferences: A review of research themes. Artificial intelligence in medicine, 65(1), pp.61-73.

[178] Crespo, R. et al. (2020) 'A spatially explicit analysis of chronic diseases in small areas: a case study of diabetes in Santiago, Chile', International journal of health geographics. Springer Science and Business Media LLC, 19(1), p. 24. doi: 10.1186/s12942-020-00217-1.

[179] Foqaha, M.A.M., 2016. Email spam classification using hybrid approach of RBF neural network and particle swarm optimization. International Journal of Network Security & Its Applications, 8(4), pp.17-28.

[180] Kim, J., Han, D., Tai, Y.W. and Kim, J., 2015. Salient region detection via high-dimensional color transform and local spatial support. IEEE transactions on image processing, 25(1), pp.9-23.

[181] Aslani, M., Mesgari, M.S. and Wiering, M., 2017. Adaptive traffic signal control with actor-critic methods in a real-world traffic network with different traffic disruption events. Transportation Research Part C: Emerging Technologies, 85, pp.732-752.

[182] Lazzaretti, A.E., Guerra, F.A., Riella, R.J., Neto, H.V. and dos Santos Coelho, L., 2009. COMBINED APPROACH OF RBF NEURAL NETWORKS, GENETIC ALGORITHMS AND LOCAL SEARCH AND ITS APPLICATION TO THE IDENTIFICATION OF A NON-LINEAR PROCESS. In Proceedings of COBEM 2009, 20th International Congress of Mechanical Engineering.

[183] Diederich, N., Bartsch, T., Kohlstedt, H. and Ziegler, M., 2018. A memristive plasticity model of voltage-based STDP suitable for recurrent bidirectional neural networks in the hippocampus. Scientific reports, 8(1), pp.1-12.

[184] Fard, A.T., Srihari, S., Mar, J.C. and Ragan, M.A., 2016. Not just a colourful metaphor: modelling the landscape of cellular development using Hopfield networks. NPJ systems biology and applications, 2(1), pp.1-9.

[185] Kasihmuddin, M.S.M., Mansor, M.A. and Sathasivam, S., 2017. Hybrid Genetic Algorithm in the Hopfield Network for Logic Satisfiability Problem. Pertanika Journal of Science & Technology, 25(1).

[186] Sahoo, R.C. and Pradhan, S.K., A Comparative Study on Hopfield Network with LBP, PCA and LDA for Face Recognition in Distorted Face Images.

[187] Du, K.-L. and Swamy, M. N. S. (2019) 'Hopfield networks, simulated annealing, and chaotic neural networks', in Neural Networks and Statistical Learning. London: Springer London, pp. 173–200. doi: 10.1007/978-1-4471-7452-3_7.

[188] Huda, S., Miah, S., Yearwood, J., Alyahya, S., Al-Dossari, H. and Doss, R., 2018. A malicious threat detection model for cloud assisted internet of things (CoT) based industrial control system (ICS) networks using deep belief network. Journal of Parallel and Distributed Computing, 120, pp.23-31.

[189] Voulodimos, A., Doulamis, N., Doulamis, A. and Protopapadakis, E., 2018. Deep learning for computer vision: A brief review. Computational intelligence and neuroscience, 2018.

[190] Liu, W., Wang, Z., Liu, X., Zeng, N., Liu, Y. and Alsaadi, F.E., 2017. A survey of deep neural network architectures and their applications. Neurocomputing, 234, pp.11-26.

[191] Ronoud, S. and Asadi, S. (2019) 'An evolutionary deep belief network extreme learning-based for breast cancer diagnosis', Soft computing. Springer Science and Business Media LLC, 23(24), pp. 13139–13159. doi: 10.1007/s00500-019-03856-0.

[192] Zhu, F., Ye, F., Fu, Y., Liu, Q. and Shen, B., 2019. Electrocardiogram generation with a bidirectional LSTM-CNN generative adversarial network. Scientific reports, 9(1), pp.1-11.

[193] Xu, D., Yuan, S., Zhang, L. and Wu, X., 2018, December. Fairgan: Fairness-aware generative adversarial networks. In 2018 IEEE International Conference on Big Data (Big Data) (pp. 570-575). IEEE.

[194] Na, T., Ko, J.H. and Mukhopadhyay, S., 2017. Cascade adversarial machine learning regularized with a unified embedding. arXiv preprint arXiv:1708.02582.

[195] Vint, D. et al. (2021) 'Automatic target recognition for Low Resolution Foliage Penetrating SAR images using CNNs and GANs', Remote sensing. MDPI AG, 13(4), p. 596. doi: 10.3390/rs13040596.

[196] Chng, Z.M. (2022). Using Activation Functions in Neural Networks. [online] Machine Learning Mastery. Available at: https://machinelearningmastery.com/using-activation-functions-in-neural-networks/.

[197] Sharma, S., Sharma, S. and Athaiya, A., 2017. Activation functions in neural networks. Towards Data Sci, 6(12), pp.310-316.

[198] Seldon. (2022). Neural Network Models Explained. [online] Available at: https://www.seldon.io/neural-network-models-explained.

[199] Ren, S., Cao, X., Wei, Y. and Sun, J., 2015. Global refinement of random forest. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 723-730).

[200] Swapna, Mudrakola, Uma Maheswari Viswanadhula, Rajanikanth Aluvalu, Vijayakumar Vardharajan, and Ketan Kotecha. 2022. "Bio-Signals in Medical Applications and Challenges Using Artificial Intelligence" Journal of Sensor and Actuator Networks 11, no. 1: 17. https://doi.org/10.3390/jsan11010017

[201] Reference List Logunova, I. (2022) Guide to random forest classification and regression algorithms, Serokell Software Development Company. Serokell. Available at: https://serokell.io/blog/random-forest-classification (Accessed: March 6, 2023).

[202] Simplilearn (2020) Random Forest Algorithm, Simplilearn.com. Simplilearn. Available at: https://www.simplilearn.com/tutorials/machine-learning-tutorial/random-forest-algorithm (Accessed: March 6, 2023).

[203] McInerny, S.A. and Dai, Y., 2003. Basic vibration signal processing for bearing fault detection. IEEE Transactions on education, 46(1), pp.149-156.

[204] Tang, G., Pang, B., Tian, T. and Zhou, C., 2018. Fault diagnosis of rolling bearings based on improved fast spectral correlation and optimized random forest. Applied sciences, 8(10), p.1859.

[205] Abdiansah, A. and Wardoyo, R., 2015. Time complexity analysis of support vector machines (SVM) in LibSVM. International journal computer and application, 128(3), pp.28-34.

[206] Shetty, Rishika & Kasbe, Shweta & Jorwekar, Kimaya & Kamble, Dharti & Velankar, Makarand. (2015). Study of Emotion Detection in Tunes Using Machine Learning.

[207] Stitson, M.O., Weston, J.A.E., Gammerman, A., Vovk, V. and Vapnik, V., 1996. Theory of support vector machines. University of London, 117(827), pp.188-191.

[208] Howley, T. and Madden, M.G., 2005. The genetic kernel support vector machine: Description and evaluation. Artificial intelligence review, 24, pp.379-395.

[209] Kavzoglu, T. and Colkesen, I., 2009. A kernel functions analysis for support vector machines for land cover classification. International Journal of Applied Earth Observation and Geoinformation, 11(5), pp.352-359.

[210] Aljarah, I., Al-Zoubi, A.M., Faris, H., Hassonah, M.A., Mirjalili, S. and Saadeh, H., 2018. Simultaneous feature selection and support vector machine optimization using the grasshopper optimization algorithm. Cognitive Computation, 10, pp.478-495.

[211] Syarif, I., Prugel-Bennett, A. and Wills, G., 2016. SVM parameter optimization using grid search and genetic algorithm to improve classification performance. TELKOMNIKA (Telecommunication Computing Electronics and Control), 14(4), pp.1502-1509.

[212] Jamil, Mutiullah & Rehman, Hafeez & SaleemUllah, & Ashraf, Imran & Ubaid, Saqib. (2023). Smart Techniques for LULC Micro Class Classification Using Landsat8 magery. Computers, Materials & Continua. 74. 5545-5557. 10.32604/cmc.2023.033449.

[213] Yang, Jun, Zhengmin Ma, and Tao Shen. 2021. "Multi-Time and Multi-Band CSP Motor Imagery EEG Feature Classification Algorithm" Applied Sciences 11, no. 21: 10294. https://doi.org/10.3390/app112110294

[214] Kumar, A. (2020) SVM RBF kernel parameters with code examples, dzone.com. DZone. Available at: https://dzone.com/articles/using-jsonb-in-postgresql-how-to-effectively-store-1 (Accessed: March 6, 2023).

[215] Prosise, J. (2021) *Support-vector machines*, *Atmosera*. Available at: https://www.atmosera.com/wintellect-blogs/support-vector-machines/ (Accessed: March 6, 2023).

[216] Putatunda, S. and Rama, K., 2019, December. A modified bayesian optimization based hyper-parameter tuning approach for extreme gradient boosting. In *2019 Fifteenth International Conference on Information Processing (ICINPRO)* (pp. 1-6). IEEE.

[217] Shi, Yong & Tian, Yingjie & Kou, Gang & Peng, Yi & Li, Jianping. (2011). Support Vector Machines for Multi-class Classification Problems. 10.1007/978-0-85729-504-0_3.

[218] Pinheiro Jr, M. and Dral, P.O., 2023. Kernel methods. In Quantum Chemistry in the Age of Machine Learning (pp. 205-232). Elsevier.

[219] Chen, X., Wang, S., Qiao, B. et al. Basic research on machinery fault diagnostics: Past, present, and future trends. Front. Mech. Eng. 13, 264–291 (2018). https://doi.org/10.1007/s11465-018-0472-3

[220] Jha, R.K. and Swami, P.D., 2021. Fault diagnosis and severity analysis of rolling bearings using vibration image texture enhancement and multiclass support vector machines. *Applied Acoustics*, *182*, p.108243.

[221] Xue, Q., Xu, B., He, C., Liu, F., Ju, B., Lu, S. and Liu, Y., 2021. Feature extraction using hierarchical dispersion entropy for rolling bearing fault diagnosis. IEEE Transactions on Instrumentation and Measurement, 70, pp.1-11.

[222] Palkhiwala, S., Shah, M. & Shah, M. Analysis of Machine learning algorithms for predicting a student's grade. J. of Data, Inf. and Manag. 4, 329–341 (2022). https://doi.org/10.1007/s42488-022-00078-2

[223] Gao K, Deng X, Cao Y (2019) Industrial process fault classification based on weighted stacked extreme learning machine. 2019 CAA Symposium on Fault Detection, Supervision and Safety for Technical Processes (SAFEPROCESS), pp. 328–332. https:// doi. org/ 10. 1109/ SAFEP ROCES S45799. 2019. 92133 17

[224] Li L, Zeng J, Jiao L, Liang P, Liu F, Yang S (2020) Online active extreme learning machine with discrepancy sampling for polsar classification. IEEE Trans Geosci Remote Sens 58(3):2027–2041. https:// doi. org/ 10. 1109/ TGRS. 2019. 29522 36

[225] Schreuder, Martijn & Blankertz, Benjamin & Tangermann, Michael. (2010). A New Auditory Multi-Class Brain-Computer Interface Paradigm: Spatial Hearing as an Informative Cue. PloS one. 5. e9813. 10.1371/journal.pone.0009813.

[226] Huang G-B (2014) An insight into extreme learning machines: random neurons, random features and kernels. Cogn Comput 6:376–390. https:// doi. org/ 10. 1007/ s12559- 014- 9255-2

[227] Wei J, Liu H, Yan G, Sun F (2017) Robotic grasping recognition using multi-modal deep extreme learning machine. Multidimens Syst Signal Process. https:// doi. org/ 10. 1007/ s11045- 016- 0389-0

[228] Fan Q, Liu T (2020) Smoothing l0 regularization for extreme learning machine. Math Probl Eng 2020:1–10. https:// doi. org/ 10. 1155/ 2020/ 91751 06

[229] Xiao D, Li B, Mao Y (2017) A multiple hidden layers extreme learning machine method and its application. Math Probl Eng 2017:1–10. https:// doi. org/ 10. 1155/ 2017/ 46701 87

[230] Miao, Anzhu & Liu, Feiping. (2021). Application of human motion recognition technology in extreme learning machine. International Journal of Advanced Robotic Systems. 18. 172988142098321. 10.1177/1729881420983219.

[231] Cen Chen, Kenli Li, Mingxing Duan, Keqin Li, Chapter 6 - Extreme Learning Machine and Its Applications in Big Data Processing, Editor(s): Hui-Huang Hsu, Chuan-Yu Chang, Ching-Hsien Hsu, In Intelligent Data-Centric Systems, Big Data Analytics for Sensor-Network Collected Intelligence, Academic Press, 2017, Pages 117-150, ISBN 9780128093931, https://doi.org/10.1016/B978-0-12-809393-1.00006-4.

[232] Toma, Rafia Nishat, Alexander E. Prosvirin, and Jong-Myon Kim. 2020. "Bearing Fault Diagnosis of Induction Motors Using a Genetic Algorithm and Machine Learning Classifiers" Sensors 20, no. 7: 1884. https://doi.org/10.3390/s20071884

[233] Mushtaq, S., Islam, M.M. and Sohaib, M., 2021. Deep learning aided data-driven fault diagnosis of rotatory machine: A comprehensive review. Energies, 14(16), p.5150.

Appendix

Dataset and Test Rig description of Case Western Reserve University (CWRU) Bearing Center

Table A1.1: Drive end bearing: 6205-2RS JEM SKF, deep groove ball bearing

Inside Diameter	Outside Diameter	Thickness	Ball Diameter	Pitch Diameter
0.9843"	2.0472"	0.5906"	0.3126"	1.537"

Table A1.2: Defect frequencies: (multiple of running speed **in Hz**)

Inner Ring	Outer Ring	Cage Train	Rolling Element
5.4152"	3.5848"	0.39828"	4.7135"

Table A1.3: Fan end bearing: 6203-2RS JEM SKF, deep groove ball bearing

Inside Diameter	Outside Diameter	Thickness	Ball Diameter	Pitch Diameter
0.6693"	1.5748"	0.4724"	0.2656"	1.122"

Table A1.4: Defect frequencies: (multiple of running speed **in Hz**)

Inner Ring	Outer Ring	Cage Train	Rolling Element
4.9469"	3.0530"	0.3817"	3.9874"

Table A1.5: Fault Specifications

Bearing	Fault Location	Diameter (")	Depth (")	Bearing Manufacturer (")
Drive End	Inner Raceway	.007	.011	SKF
Drive End	Inner Raceway	.014	.011	SKF
Drive End	Inner Raceway	.021	.011	SKF
Drive End	Inner Raceway	.028	.050	NTN
Drive End	Outer Raceway	.007	.011	SKF
Drive End	Outer Raceway	.014	.011	SKF
Drive End	Outer Raceway	.021	.011	SKF
Drive End	Outer Raceway	.040	.050	NTN
Drive End	Ball	.007	.011	SKF
Drive End	Ball	.014	.011	SKF
Drive End	Ball	.021	.011	SKF
Drive End	Ball	.028	.150	NTN
Fan End	Inner Raceway	.007	.011	SKF
Fan End	Inner Raceway	.014	.011	SKF
Fan End	Inner Raceway	.021	.011	SKF
Fan End	Outer Raceway	.007	.011	SKF
Fan End	Outer Raceway	.014	.011	SKF
Fan End	Outer Raceway	.021	.011	SKF
Fan End	Ball	.007	.011	SKF
Fan End	Ball	.014	.011	SKF
Fan End	Ball	.021	.011	SKF

Table A1.6: 12k Drive End Bearing Fault Data

* = Data not available

Fault Diameter	Motor Load (HP)	Approx. Motor Speed (rpm)	Inner Race	Ball	Outer Race Position Relative to Load Zone (Load Zone Centered at 6:00)		
					Centred @6:00	Orthogonal @3:00	Opposite @12:00
0.007"	0	1797	IR007_0	B007_0	OR007@6_0	OR007@3_0	OR007@12_0
	1	1772	IR007_1	B007_1	OR007@6_1	OR007@3_1	OR007@12_1
	2	1750	IR007_2	B007_2	OR007@6_2	OR007@3_2	OR007@12_2
	3	1730	IR007_3	B007_3	OR007@6_3	OR007@3_3	OR007@12_3
0.014"	0	1797	IR014_0	B014_0	OR014@6_0	*	*
	1	1772	IR014_1	B014_1	OR014@6_1	*	*
	2	1750	IR014_2	B014_2	OR014@6_2	*	*
	3	1730	IR014_3	B014_3	OR014@6_3	*	*
0.021"	0	1797	IR021_0	B021_0	OR021@6_0	OR021@3_0	OR021@12_0
	1	1772	IR021_1	B021_1	OR021@6_1	OR021@3_1	OR021@12_1
	2	1750	IR021_2	B021_2	OR021@6_2	OR021@3_2	OR021@12_2
	3	1730	IR021_3	B021_3	OR021@6_3	OR021@3_3	OR021@12_3
0.028"	0	1797	IR028_0	B028_0	*	*	*
	1	1772	IR028_1	B028_1	*	*	*
	2	1750	IR028_2	B028_2	*	*	*
	3	1730	IR028_3	B028_3	*	*	*

Table A1.7: 48k Drive End Bearing Fault Data

Fault Diameter	Motor Load (HP)	Approx. Motor Speed (rpm)	Inner Race	Ball	Outer Race Position Relative to Load Zone (Load Zone Centered at 6:00)		
					Centred @6:00	Orthogonal @3:00	Opposite @12:00
0.007"	0	1797	IR007_0	B007_0	OR007@6_0	OR007@3_0	OR007@12_0
	1	1772	IR007_1	B007_1	OR007@6_1	OR007@3_1	OR007@12_1
	2	1750	IR007_2	B007_2	OR007@6_2	OR007@3_2	OR007@12_2
	3	1730	IR007_3	B007_3	OR007@6_3	OR007@3_3	OR007@12_3
0.014"	0	1797	IR014_0	B014_0	OR014@6_0	*	*
	1	1772	IR014_1	B014_1	OR014@6_1	*	*
	2	1750	IR014_2	B014_2	OR014@6_2	*	*
	3	1730	IR014_3	B014_3	OR014@6_3	*	*
0.021"	0	1797	IR021_0	B021_0	OR021@6_0	OR021@3_0	OR021@12_0
	1	1772	IR021_1	B021_1	OR021@6_1	OR021@3_1	OR021@12_1
	2	1750	IR021_2	B021_2	OR021@6_2	OR021@3_2	OR021@12_2
	3	1730	IR021_3	B021_3	OR021@6_3	OR021@3_3	OR021@12

Table A1.8: 12k Fan End Bearing Fault Data

* = Data not available

Fault Diameter	Motor Load (HP)	Approx. Motor Speed (rpm)	Inner Race	Ball	Outer Race Position Relative to Load Zone (Load Zone Centered at 6:00)		
					Centred @6:00	Orthogonal @3:00	Opposite @12:00
0.007"	0	1797	IR007_0	B007_0	OR007@6_0	OR007@3_0	OR007@12_0
	1	1772	IR007_1	B007_1	OR007@6_1	OR007@3_1	OR007@12_1
	2	1750	IR007_2	B007_2	OR007@6_2	OR007@3_2	OR007@12_2
	3	1730	IR007_3	B007_3	OR007@6_3	OR007@3_3	OR007@12_3
0.014"	0	1797	IR014_0	B014_0	OR014@6_0	OR014@3_0	*
	1	1772	IR014_1	B014_1	*	OR014@3_1	*
	2	1750	IR014_2	B014_2	*	OR014@3_2	*
	3	1730	IR014_3	B014_3	*	OR014@3_3	*
0.021"	0	1797	IR021_0	B021_0	OR021@6_0	*	*
	1	1772	IR021_1	B021_1	*	OR021@3_1	*
	2	1750	IR021_2	B021_2	*	OR021@3_2	*
	3	1730	IR021_3	B021_3	*	OR021@3_3	*

The Case Western Reserve University Bearing Fault Dataset can be downloaded from:

https://engineering.case.edu/bearingdatacenter/download-data-file

Dataset and Test Rig description of the Dynamic and Identification Research Group (DIRG)

The dataset used in this research is from a high-speed aeronautical bearing test rig created by the Dynamic & Identification Research Group (DIRG) at Politecnico di Torino's Department of Mechanical and Aerospace Engineering. The rig is completely described in **[24]**, although the important points are summarised below]. A single direct-drive rotating shaft is supported by two identical high-speed aeronautical roller bearings (B1 and B3 in **Figure I.7**) to form the rig. B3 is known to be healthy, however, B1 has been purposefully injured with indentations of varying sizes in various regions of the bearing (Rolling Element and Inner Ring), as shown in **Table A2. 3**. The third central bearing B2 is installed on a sledge that is designed to load the shaft with increasing forces of 0, 1000, 1400, and 1800 N, as the speed decreases from 470 to 0 Hz (run-down acquisitions). The operational circumstances are summarised in **Table A2. 1**. Two tri-axial accelerometers, one on the B1 bearing support (accelerometer A1) and the other on the loading sledge (accelerometer A2). The acquisitions last around T = 50 s and have a sampling frequency of fs = 102400 Hz. To perform a significant analysis, the five selected features root mean square, skewness, kurtosis, peak value, and crest factor are extracted on one hundred independent chunks (approximately 0.5 s each) for each of the 6 channels of the 4 original acquisitions in all 7 health conditions (from 0A, healthy, to 6A). Finally, 100 observations in a 30-dimensional space (6 channels, 5 features) are gathered for each condition.

Table A2. 1. The different loads while the speed is decreasing from 470 to 0 Hz (rundown acquisitions).

Label	1	2	3	4
F **[kN]**	0	1	1.4	1.8

Table A2. 2: File names for the eight bearings with different damages, from 0A to 6A

1	C0A_100_000_1.mat	C1A_100_000_2.mat	C2A_100_000_1.mat	C3A_100_000_1.mat
2	C0A_100_505_1.mat	C1A_100_502_2.mat	C2A_100_506_1.mat	C3A_100_505_1.mat
3	C0A_100_706_1.mat	C1A_100_702_2.mat	C2A_100_701_1.mat	C3A_100_699_1.mat
4	C0A_100_899_1.mat	C1A_100_898_2.mat	C2A_100_901_1.mat	C3A_100_906_1.mat
5	C0A_200_505_1.mat	C1A_200_502_2.mat	C2A_200_506_1.mat	C3A_200_505_1.mat
6	C0A_200_706_1.mat	C1A_200_702_2.mat	C2A_200_701_1.mat	C3A_200_699_1.mat
7	C0A_200_899_1.mat	C1A_200_898_2.mat	C2A_200_901_1.mat	C3A_200_906_1.mat
8	C0A_300_505_1.mat	C1A_300_502_2.mat	C2A_300_506_1.mat	C3A_300_505_1.mat
9	C0A_300_706_1.mat	C1A_300_702_2.mat	C2A_300_701_1.mat	C3A_300_699_1.mat
10	C0A_300_899_1.mat	C1A_300_898_2.mat	C2A_300_901_1.mat	C3A_300_906_1.mat
11	C0A_400_505_1.mat	C1A_400_502_2.mat	C2A_400_506_1.mat	C3A_400_505_1.mat
12	C0A_400_706_1.mat	C1A_400_702_2.mat	C2A_400_701_1.mat	C3A_400_699_1.mat
13	C0A_500_505_1.mat	C1A_500_502_2.mat	C2A_500_506_1.mat	C3A_500_505_1.mat

1	C4A_100_000_1.mat	C5A_100_000_1.mat	C6A_100_000_1.mat
2	C4A_100_496_1.mat	C5A_100_498_1.mat	C6A_100_500_1.mat
3	C4A_100_702_1.mat	C5A_100_700_1.mat	C6A_100_705_1.mat
4	C4A_100_895_1.mat	C5A_100_900_1.mat	C6A_100_909_1.mat
5	C4A_200_496_1.mat	C5A_200_498_1.mat	C6A_200_500_1.mat
6	C4A_200_702_1.mat	C5A_200_700_1.mat	C6A_200_705_1.mat
7	C4A_200_895_1.mat	C5A_200_900_1.mat	C6A_200_909_1.mat
8	C4A_300_496_1.mat	C5A_300_498_1.mat	C6A_300_500_1.mat
9	C4A_300_702_1.mat	C5A_300_700_1.mat	C6A_300_705_1.mat
10	C4A_300_895_1.mat	C5A_300_900_1.mat	C6A_300_909_1.mat
11	C4A_400_496_1.mat	C5A_400_498_1.mat	C6A_400_500_1.mat
12	C4A_400_702_1.mat	C5A_400_700_1.mat	C6A_400_705_1.mat
13	C4A_500_496_1.mat	C5A_500_498_1.mat	C6A_500_500_1.mat

The time histories of the six channels (**Table A2. 2**) have been collected with sampling frequency fs=51200 Hz for a duration of T=10s. Data are recorded in files whose names have the following format: **CnA_fff_vvv_m.mat.**

C: the root of the file name, common to all files;

n: integer value from 0 to 6, indicating the kind of the defect, e.g., 1A, …, 6A (**Table A2. 3**);

fff: integer value from 100 to 500, indicating the nominal speed of the shaft (Hz);

vvv: integer value corresponding to the voltage of the load cell (mV), indicating the applied load;

Table A2. 3: List of the defects of the various bearings mounted in position B1.

Name	Defect	Dimension (μm)
0A	NO DEFECT	----
1A	The diameter of an indentation on the inner ring	450
2A	The diameter of an indentation on the inner ring	250
3A	The diameter of an indentation on the inner ring	150
4A	The diameter of an indentation on a roller	450
5A	The diameter of an indentation on a roller	250
6A	The diameter of an indentation on a roller	150

The entire procedure took around 30 minutes, and due to the inverter's restricted power, higher rotation speeds with higher loading circumstances were not possible. The speed-load combinations are listed in **Table A2. 4**.

Table A2. 4: List of the tested load and speed cases.

Nominal load (N)	**Nominal speed (Hz)**				
0	100	200	300	400	500
1000	100	200	300	400	500
1400	100	200	300	400	----
1800	100	200	300	----	----

The related time records can be downloaded from: ftp://ftp.polito.it/people/DIRG_BearingData/