

Fault Diagnosis of Gearbox Based Pitch Drives in Wind Turbines

Fault Detection by Support Vector Machine
Using Motor Current Signal Analysis

JØRGEN VOLLSTAD
VEGARD SVENSLI ÅSVESTAD

SUPERVISOR
van Khang Huynh
Surya Teja Kandukuri

University of Agder, 2020
Faculty of Engineering and Science
Department of Engineering Sciences

Abstract

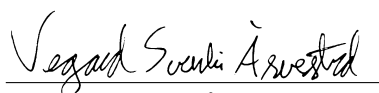
The growing dependence on wind power in recent years has increased the demand for reliant wind turbines. The pitch system of a wind turbine is one of the components with the highest failure rates. The most common way of diagnosing pitch system faults is currently through vibration analysis, which requires the installation of vibration sensors. This thesis presents a non-intrusive method for fault detection of the planetary gearbox in an electric wind turbine pitch system. The method is based on using the three-phase motor currents from the induction motor of the pitch system to calculate a DC offset using Extended Park's vector approach (EPVA). Basic statistical formulas are used to extract features from both the time- and frequency-domain of the DC offset, where fast Fourier transform (FFT) is used to find the frequency-domain values. These features, along with the amplitudes of the characteristic frequencies of the planetary gearbox and its bearing, are used in the principal component analysis (PCA) to generate features that are used to train a support vector machine (SVM) classifier. This method is validated by using labeled data from the induction motor of a pitch system test bench to classify three health conditions. One of the health conditions are a healthy system, and the two other are artificially seeded faults in the system's two-stage gearbox. These faults are a partially cracked tooth in one of the first stage planet gears, and an outer race fault in the bearing at the input shaft. The results indicate that the proposed method is capable of classifying each of the three health conditions.


Preface

This thesis concludes the renewable energy engineering master education at the University of Agder (UiA), Grimstad. This thesis is a result of the final course in the master education, Master Thesis Renewable Energy (ENE500), which is written in the spring semester of 2020 and corresponds to 30 ECTS-credits.

The subject of this thesis was chosen from a list of projects proposed by the University of Agder. We chose this subject due to our interest in wind energy and electrical systems and found this subject to be intriguing and challenging. Both Jørgen and Vegard received a B.Sc degree in renewable energy from the University of Agder, Grimstad, Norway, in 2018, and is now pursuing a M.Sc degree in renewable energy from the University of Agder, Grimstad, Norway, since 2018.

We would like to thank our supervisors van Khang Huynh and Surya Teja Kandukuri, for your guidance and valuable inputs regarding the thesis. We are especially grateful for your useful insight in finding us a new task when we lost access to the laboratory due to COVID-19. An extra thanks to Surya for providing us with the data used in this thesis. We would also like to thank the University staff for working to provide us with a place to work toward the end of our thesis.


Vegard Svensli Årvestad


Jørgen Vollstad

Mandatory Declaration

1.	I/We hereby declare that my/our report is my/our own work and that I/We have not used any other sources or have received any other help than mentioned in the report.	<input checked="" type="checkbox"/>
2.	I/we further declare that this report: - has not been used for another exam at another department/university/university college in Norway or abroad; - does not refer to the work of others without it being stated; - does not refer to own previous work without it being stated; - have all the references given in the literature list; - is not a copy, duplicate or copy of another's work or manuscript.	<input checked="" type="checkbox"/>
3.	I/we am/are aware that violation of the above is regarded as cheating and may result in cancellation of exams and exclusion from universities and colleges in Norway, see Universitets- og høyskoleloven §§4-7 og 4-8 og Forskrift om eksamen §§ 31.	<input checked="" type="checkbox"/>
4.	I/we am/are aware that all submitted reports may be checked for plagiarism.	<input checked="" type="checkbox"/>
5.	I/we am/are aware that the University of Agder will deal with all cases where there is suspicion of cheating according to the university's guidelines for dealing with cases of cheating.	<input checked="" type="checkbox"/>
6.	I/we have incorporated the rules and guidelines in the use of sources and references on the library's web pages.	<input checked="" type="checkbox"/>

Publishing Agreement

Authorization for electronic publishing of the report.

Author(s) have copyrights of the report. This means, among other things, the exclusive right to make the work available to the general public (Åndsverkloven. §2). All theses that fulfill the criteria will be registered and published in Brage Aura and on UiA's web pages with author's approval. Reports that are not public or are confidential will not be published.

I hereby give the University of Agder a free right to make the task available for electronic publishing:

YES NO

Is the report confidential?

YES NO

(confidential agreement must be completed and signed by the Head of the Department)

- If yes:

Can the report be published when the confidentiality period is over?

YES NO

Is the task except for public disclosure?

YES NO

(contains confidential information. see Offl. §13/Fvl. §13)

Contents

Abstract	i
Preface	iii
Mandatory Declaration	v
Publishing Agreement	vii
List of Figures	xiii
List of Tables	xv
Notation	xvii
Abbreviations	xix
1 Introduction	1
1.1 Problem definition	2
1.2 Thesis structure	3
2 Theoretical background	5
2.1 Literature review	5
2.2 Pitch system	7
2.3 Characteristic fault frequencies	8
2.4 Statistical features	9
2.5 Extended Park’s vector approach	10
2.6 Machine learning	11
2.6.1 Support vector machine	12
2.6.2 Cross-validation	16
2.6.3 Hyperparameter optimization	16
2.6.4 Principal component analysis	17
2.6.5 Receiver operating characteristic curve	18

3	Method	19
3.1	Laboratory setup	19
3.2	Data acquisition	21
3.3	Data overview	21
3.4	Data preparation	22
3.5	Feature selection	23
3.6	SVM model	24
4	Results and discussion	25
4.1	Preliminary analysis	25
4.1.1	Frequency spectrum overview	25
4.1.2	Motor drive harmonics	25
4.1.3	Characteristic frequencies	26
4.1.4	Statistical features	29
4.1.5	High-frequency spectrum analysis	29
4.1.6	Summary of the preliminary results	30
4.2	Results calculated using all samples	31
4.2.1	Feature correlation	31
4.2.2	Mean motor drive harmonics and characteristic frequencies amplitudes	31
4.2.3	Mean statistical features of all samples	32
4.3	PCA	33
4.4	SVM Results	34
4.5	Discussion	36
5	Conclusion	37
6	Further Work	39
	Reference	41
	Appendices	45
A	Specifications	45

B	Additional results	46
C	Overview of raw data	47
D	Calculations	48
E	Codes	49

List of Figures

1	Chart showing the distribution of cumulative downtime of 350 WTs.	1
2	Three single electric pitch drive mounted to the WT hub [21].	7
3	The two reference frames in a Park’s transformation.	10
4	Diagram of the three basic machine learning types, as well as their sub types and some use case examples.	11
5	Shows an example of underfitting, overfitting and a good fit [30].	12
6	Illustration showing how samples are divided by the decision boundary, where positive samples are represented by red dots, and negative samples by blue dots. 13	13
7	Showing the class matrix for (a) OVA and (b) OVO, where the white squares represent a positive label, the black represents a negative label, and the gray squares are ignored.	15
8	Figure comparing grid search, random search, and Bayesian optimization. Each point corresponds to a set of hyperparameters. The color of the contour plot is the classification accuracy of the classifier, with warmer colors indicating higher accuracy [38].	16
9	Illustrating the concept of PCA transformation.	17
10	Example of an ROC curve.	18
11	A schematic representation of the laboratory setup showing the scaled pitch drive and the load along with the current sensors.	20
12	Shows the artificially seeded fault in (a) the bearing and (b) the planet gear [44].	21
13	Flowchart with three sections representing the data processing system.	22
14	Plot showing the the difference between the frequency readout from the motor drive in samples 1 and 2.	22
15	Plot showing the RMS value of i_p of samples 1 to 350 with vertical lines indicating when a new run begins.	22
16	Flowchart of the feature selection process.	23
17	Plots showing the FFT of i_p of the three samples, where (a) is the three samples of the preliminary analysis, and (b) is the two faulty samples compared with an HLT sample with a load of 69%.	25
18	Showing a plot of the FFT of i_p of the two fault conditions in the region 0-400 Hz, where the black dashed lines show the position of the motor drive harmonics.	26
19	Bar plot comparing the amplitudes of the motor drive harmonics of the preliminary samples.	26

20	Showing a plot of the FFT of i_p of the two fault conditions in the region 0-400 Hz. The black dashed lines indicate the position of the CFs shown in Table. 5.	27
21	Sections of Fig. 20 in the region of: (a) f_{p1} and f_{s1} , (b) the 1st harmonic and f_{m2} , and (c) f_{bpor} and the 4th harmonic.	27
22	Bar plot showing the amplitudes of the CFs of the samples in the preliminary analysis.	28
23	Bar graph showing the values of the statistical features of the samples used in the preliminary analysis in (a) the time-domain and (b) the frequency-domain.	29
24	Plot comparing the high-frequency peaks of the preliminary samples, where the two fault conditions are compared in (a), and where (b) and (c) are the STFT of the all the samples in the PLT and BRG runs, respectively.	30
25	Close-up of the 4 kHz region of Fig. 24.	30
26	Bar plot displaying all the features average correlation with speed and load.	31
27	Bar plot showing the average amplitudes of the CFs of all samples.	32
28	Bar plot comparing the average amplitudes of the motor drive harmonics of all samples.	32
29	Bar graph of the average values of (a) the time-domain and (b) frequency-domain statistical features of all samples.	33
30	Plot showing the variance of each PC, and the cumulative variance of the PCs. The red dashed line indicate where the cumulative variance reaches 90%.	33
31	Confusion matrix of (a) the training set and (b) the testing set.	35
32	Plot showing decision boundaries of an SVM trained on the first two PCs, and the clustering of the samples with the same health condition.	35
33	ROC curve of the test set. The red diagonal line indicates the performance of random classification.	36
B.1	Showing the complete correlation matrix for all variables. The colorbar indicated the absolute value of the correlation coefficient.	46

List of Tables

1	Characteristic fault frequencies of a planetary gearbox.	8
2	Statistical features derived from the time-domain and frequency-domain of signals.	9
3	Parameters of the two-stage planetary gearbox.	19
4	Showing the number of samples per health condition.	22
5	Showing the values of the CFs when the operating speed is 1200 RPM, arranged in descending order.	28
6	Showing the hyperparameters used for the SVM.	34
A.1	Current sensor specifications.	45
A.2	Specifications of the pitch and load motor.	45
C.1	Overview of the raw data.	47

Notation

b	- The planes offset from the origin along \vec{w}
B_d	- Ball diameter
BRG	- Bearing fault health condition
C	- Penalty parameter
d	- Polynomial degree
$f_1 - f_7$	- RMS value, standard deviation, variance, 3rd order central moment, skewness, kurtosis and crest factor, respectively, in the frequency-domain.
f_{bpir}	- Ball pass inner race fault frequency
f_{bpor}	- Ball pass outer race fault frequency
f_{bsf}	- Ball spin frequency
f_{c1}	- First stage carrier frequency
f_{c2}	- Second stage carrier frequency
f_{ftf}	- Fundamental train frequency
f_{in}	- Input shaft frequency
f_{m1}	- First stage meshing frequency
f_{m2}	- Second stage meshing frequency
f_{p1}	- First stage planet frequency
f_{p2}	- Second stage planet frequency
f_s	- Fundamental supply frequency
f_{s1}	- First stage sun frequency
f_{s2}	- Second stage sun frequency
FN	- Number of false negatives
FP	- Number of false positives
g	- Binary loss function
H	- Decision boundary
HLT	- Healthy health condition
i_a, i_b, i_c	- Three-phase currents
i_d, i_q	- Direct and quadrature currents, respectively
i_p	- DC offset
k	- Number of classes
\hat{k}	- Predicted class
l	- Number of support vector machines
$m_{k,l}$	- (k,l) element of matrix
n	- Number of features of the sample
N_b	- Number of rolling elements in the bearing
N_p	- Number of planet gears
P_d	- Pitch diameter of the bearing
PLT	- Planet gear fault health condition
r	- Polynomial coefficient
\bar{s}	- Mean in the frequency-domain
$s(k)$	- The frequency spectrum
s_l	- Prediction score of a positive class of SVM, l
$t_1 - t_7$	- RMS value, standard deviation, variance, 3rd order central moment, skewness, kurtosis and crest factor, respectively, in the time-domain.
TN	- Number of true negatives
TP	- Number of true positives
\vec{u}	- n -dimensional vector containing the features of the unknown sample.
\vec{w}	- The n -dimensional normal vector of the plane
\vec{x}	- Set of features
\bar{x}	- Mean in the time-domain
$x(n)$	- The signal time series
y_i	- Labeling variable
z_p	- Number of planet gear teeth
z_r	- Number of ring gear teeth
z_s	- Number of sun gear teeth
α_i	- Lagrangian multiplier
$\phi()$	- Kernel function
σ	- Free parameter
θ	- Contact angle
ξ_i	- Slack variable

Abbreviations

AC	-	Alternating current
AUC	-	Area under curve
BPH	-	Bevel-planetary-helical
CF	-	Characteristic frequency
CWT	-	Continuous wavelet transform
DC	-	Direct current
DTW	-	Dynamic time warping
DWT	-	Discrete wavelet transform
ECOC	-	Error-correcting output codes
EPVA	-	Extended Park's vector approach
FFT	-	Fast Fourier transform
FOC	-	Field oriented control
FPR	-	False positive rate
IM	-	Induction motor
MCSA	-	Motor current signature analysis
NI DAQ	-	National Instruments data acquisition board
OVA	-	One vs. all
OVO	-	One vs. one
PC	-	Principal component
PCA	-	Principal component analysis
PVA	-	Park's vector approach
RBF	-	Radial basis function
RMS	-	Root mean square
ROC	-	Receiver operating characteristic
RPM	-	Revolutions per minute
STFT	-	Short time Fourier transform
SVM	-	Support Vector Machine
TPR	-	True positive rate
UIA	-	University of Agder
VFD	-	Variable frequency drive
WT	-	Wind Turbine

1 Introduction

Wind energy is one of the fastest-growing sources of renewable energy, due to its wide availability, low environmental impact, and low operating cost. The global wind energy capacity exceeded 650 GW in 2019, according to the annual report on global wind industry by the Global Wind Energy Council [1]. This report further states that in 2019, a total of 60.4 GW of wind energy capacity was installed globally, which was an increase of 19% compared to 2018. As the size and capacity of wind turbines (WTs) are steadily increasing, the downtime of each turbine has substantially higher consequences and associated costs. A survey conducted by the ReliaWind program [2], collected data on the downtime and maintenance of 350 WTs. The study found that 15% of all failures were pitch system related, and further that these faults were responsible for about 20% of the total downtime, as depicted in Fig. 1.

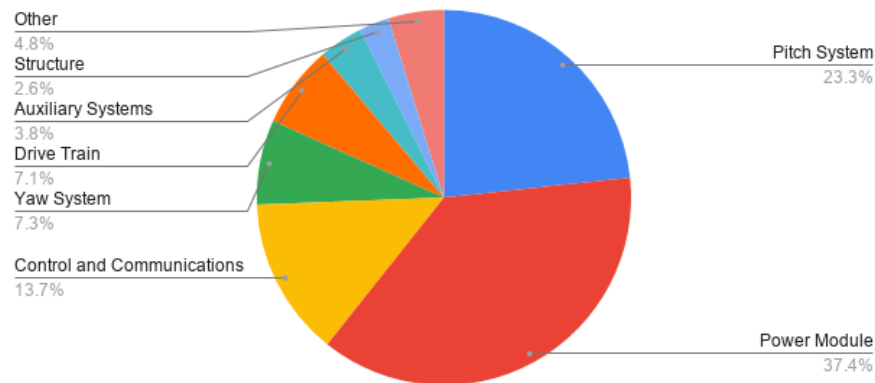


Fig. 1: Chart showing the distribution of cumulative downtime of 350 WTs.

Fault diagnosis based on vibration signals is currently the most commonly studied technique for fault diagnosis of gearboxes, as vibration signals represent the dynamic behavior of rotating machinery directly [3, 4, 5]. However, this requires vibration sensors to be installed, which adds to the complexity of the system, and introduces another component that has the potential to fail [6]. A non-intrusive method that has gained popularity in recent years is motor current signal analysis (MCSA). This method can utilize the current-sensing ability of the variable frequency drive already installed in the system, removing the need for additional components [7]. This method has been proven to be a reliable method for detecting faults on the induction motor (IM) of the pitch system, whereas using this method for fault detection of other components of the system is still in the early phases of research [8].

Fault signatures in the motor current produced by mechanical faults are orders of magnitudes smaller compared with fault signatures of vibration sensors. This has led to the rise in popularity of using machine learning to identify the minuscule changes in the current spectrum. In recent years, support vector machine (SVM) classifiers have grown in popularity due to its good classification performance and computational efficiency. For this reason, have it been extensively used in classifications of induction motor faults, as well as other components [9].

1.1 Problem definition

This thesis considers fault detection of the first stage of a two-stage planetary gearbox, which is a part of a pitch system test bench. Calculations will be based on current-data from the induction motor that supplies mechanical power to the system. This current-data is labeled with the health condition of the gearbox. Three health conditions will be analysed, being an outer race bearing fault, chipped tooth of the planet gear, and a healthy system. An SVM classifier will be used to attempt to classify each condition correctly.

Based on this, has the following goal been established:

- Determine whether or not MCSA can be used in conjunction with an SVM classifier to correctly classify faults in the two-stage planetary gearbox of a pitch system test bench.

Some sub-goals were established as a method to reach the main goal:

- Attempt to find good discriminatory features for classification.
- Determine if the characteristic frequencies of the planetary gearbox and the bearing can be observed in the motor current.
- Attempt to identify artifacts in the current-spectrum that have the potential to indicate the gearbox health condition.

The method for achieving these goals are laid out in the methodology section.

1.2 Thesis structure

In this section, a summary of each section in the thesis will be presented.

- **Section 1** - Introduction
This section presents an overall background for the proposed problem in this thesis. Further on, the problem statement for this thesis is presented.
- **Section 2** - Theoretical background
This section starts with a literature review to gain a scientific perspective on relevant research on the subject, followed by a fundamental theoretical background for the thesis. The theory describes the machine learning techniques used in his thesis. Characteristic fault frequencies and statistical features were also presented.
- **Section 3** - Method
This section describes the methodology of the thesis. The laboratory setup the data was collected from is presented. The three health conditions of the planetary gearbox are presented. The architecture of the data processing system is also described, along with the SVM classifier.
- **Section 4** - Results and Discussion
In this section, the results from the analysis and the fault diagnosis are presented. First, a preliminary analysis was performed. The findings from this analysis were then compared with the average value of all the samples. Last, the results from the principal component analysis (PCA) and SVM are presented. The results from all these analyses are also discussed in this section.
- **Section 5** - Conclusion
In this section, the concluding remarks of the work are presented.
- **Section 6** - Further work
In this section, recommendations for further work are presented.

2 Theoretical background

2.1 Literature review

This thesis is an extension of the study by Kandukuri et al. [10], which studied a two-stage fault detection and classification scheme for electric motor drives in WT pitch systems. In the first stage of the fault detection and classification scheme, Extended Park's vector approach (EPVA) was calculated for each pitch system of the WT. Three characteristic frequencies (CFs) were calculated along with several statistical features both on time- and frequency-domain representation of the EPVA, which led to a total of 19 features. It was shown that the characteristic fault frequency of the bearing was absent in the EPVA. However, the statistical features were compared across four health conditions, where one of the findings was that the 3rd- and 4th-order moments of the frequency spectrum had substantially higher amplitudes for a motor with a bearing fault, compared to a healthy one. This showed that statistical features could be good features for the SVM classifier, which was the second stage of the fault detection and classification scheme.

Fault diagnosis of a geared drive train system is usually based on vibration monitoring. However, this can be difficult to implement in planetary gearboxes due to the complexity of the measured vibration signal. This is caused by the planet's rotation around the sun gear, and the varying phase angle [11]. Kar and Mohanty [12] studied MCSA as a replacement for vibration signature analysis, to detect faults in a conventional operating transmission gearbox, and to measure its load fluctuations. Higher frequencies of steady current signals were analysed, where discrete wavelet transform (DWT) was first applied, followed by a fast Fourier transform (FFT) analysis of the decomposed signals. Continuous wavelet transform (CWT) was also applied for comparison to DWT, where it was concluded that DWT was more effective than CWT for detecting faults in the gearbox. Hong et al. [13] presented a new time-domain fault detection algorithm based on MCSA for planetary gear faults. This method combined fast dynamic time warping (DTW) and correlated kurtosis techniques to process the current signals data to detect damaged planetary gears and to locate its position. By simulation and analysis, the results showed that the method provided an effective and easy implementation of the time-domain approach. In another study, by Zhang et al. [14], the resonance residual technique was applied for the first time to MCSA to detect planetary gearbox faults. This approach was applied through both simulations and experiments on an electromechanical drive train, consisting of a motor connected to a load generator through a back-to-back planetary gearbox. The results from this study showed that the fault indicator, determined by the approach, was more sensitive to the presence of gearbox faults compared to existing MCSA-based approaches under different operating conditions. It was also pointed out that the motor current-signals are complex, weak faults are challenging to identify, and CFs are hard to extract, from complex components such as planetary gearboxes. Due to these complicated modulation characteristics of the current signal, Suo et al. [15] proposed the use of EPVA and a demodulation method based on three-phase current transformation for planetary gearbox fault diagnosis. The proposed method was compared with traditional planetary gearbox fault diagnosis methods through simulations and experiments. It was demonstrated that the proposed method could utilize three-phase current signals comprehensively, and extract modulated weak frequency components effectively. Lu et al. [16] analysed the principle of using non-stationary stator current signals to detect faults in a multistage gearbox connected

to a generator in varying speed conditions. Based on this analysis, the CFs of gear faults in the frequency spectrum of the current signals were identified. An adaptive re-sampling algorithm with fault feature extraction and fault detectors were added to the method to deal with time-varying frequencies.

Gangsar and Tiwari [17] studied the prediction of mechanical and electrical faults in IMs by vibration and current monitoring. The investigation concluded that an SVM was able to successfully predict all mechanical faults with only the use of vibration signals. However, for an effective prediction of electrical faults, it was more advantageous to use the current signals alone to perform SVM classifications. Mehala and Dahiya [18] did a comparative study for IM fault diagnostic analysis. The spectrum was obtained by using FFT, which was performed on the signal under analysis. However, FFT may give inaccurate results in cases of non-constant load torque. The paper proposed two other signal processing methods, namely, Short Time Fourier Transform (STFT) and wavelet transform. In the STFT technique, a spectrogram was used to estimate the frequency content of a signal. The wavelet transform showed changes in harmonics amplitude and distribution. Wavelet transform allows signal representation simultaneously in time- and frequency-domains.

Bearing faults are the main reason for failures in rotating machinery, where early fault detection is crucial to prevent critical system failures. Senanayaka et al. [19] applied two SVM algorithms for early detection and classification of bearing faults. The two SVM-classifiers applied were a linear SVM and a quadratic SVM, where their comparison showed that both classifiers have high accuracy. In another study, by Singh et al. [20], the detection of bearing faults in a mechanical system using MCSA was investigated. FFT was first employed for the comparison between a healthy and defective bearing. CWT was then used for the detection and occurrence of outer race faults in the bearings through MCSA. For this, six wavelets were considered, out of which three were real-valued, and the remaining three were complex-valued.

2.2 Pitch system

The pitch system of a WT is crucial to ensure safe and efficient operation and is often subjected to varying loads and unfavorable operational environments, which can lead to increased wear on the system, and eventually, failure. In an electric pitch system, three failures that commonly occur are control failures, electrical failures on the motor and converter, and mechanical failures on the gearbox and drive train [21]. While operating, the control system of the WT adjusts the pitch of the blades to keep the rotor speed within its operating limit. For example, in a high wind speed area where the rotor speed limit is reached, the pitch can be controlled to keep the rotor from exceeding its limit. If a failure is detected in one of the critical components of a WT, the pitch system can act as an aerodynamic brake by pitching the blades to ensure that the WT automatically stops. The blade pitch control system is essential for WT operation, as pitching helps enhance energy capture, reducing operation load, stalling, and aerodynamic braking [22].

The electric pitch drive system consists of six main components: electric motor, converter, gearbox and drive pinion gear, rotor blade, blade rotary joint, and drive control unit [21]. The drive control unit receives the desired pitch angle reference from the WT pitch control system, which then calculates the motor's drive signal. A pinion is mounted on the output shaft which meshes with the teeth of the rotary joint of the blade [23]. A three-bladed WT with an electric pitch system normally has a pitch drive for each blade, mounted in the hub, where each adjusts the pitch independently [24]. A simplified scheme of the electric pitch system is shown in Fig. 2.

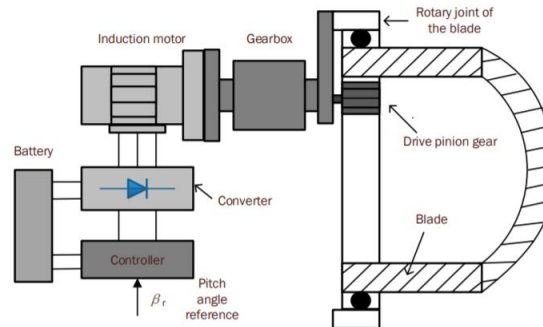


Fig. 2: Three single electric pitch drive mounted to the WT hub [21].

2.3 Characteristic fault frequencies

Bearing elements in the industry often operate under non-ideal conditions and are exposed to ambient vibrations, incorrect lubrication, overheating, moisture, shaft overload, misalignment, etc. Bearings exposed to these conditions may get small defects, such as micro cracks and dents. These defects may develop into significant flaws, which can generate detectable vibrations and increase the noise level. Bearings are an important component in rotary machines as they reduce the friction between moving parts, and can reduce the axial or radial loads on the shaft. Bearing faults can be classified as inner raceway, outer raceway, ball defect, and cage defect, which are the main sources for machine vibration [25]. One common way to detect the type of bearing fault is to see if the characteristic fault frequencies are present in the frequency spectrum of the current or the vibration signal. The CFs are given by:

$$f_{bpir} = \frac{N_b}{2} \left(1 + \frac{B_d}{P_d} \cos\theta\right) f_{in} \quad (1)$$

$$f_{bpor} = \frac{N_b}{2} \left(1 - \frac{B_d}{P_d} \cos\theta\right) f_{in} \quad (2)$$

$$f_{bsf} = \frac{P_d}{2d} \left(1 - \frac{B_d^2}{P_d^2} \cos^2\theta\right) f_{in} \quad (3)$$

$$f_{ftf} = \frac{1}{2} \left(1 - \frac{B_d}{P_d} \cos\theta\right) f_{in} \quad (4)$$

where f_{bpir} is the ball pass inner race fault frequency, f_{bpor} is the ball pass outer race fault frequency, f_{bsf} is the ball spin frequency and f_{ftf} is the fundamental train frequency. f_{in} is the input shaft frequency, N_b is the number of rolling elements in the bearing, B_d is the ball diameter, P_d is the pitch diameter of the bearing and θ is the contact angle.

The CFs of the sun gear (f_{s1}), carrier plate (f_{c1}), fundamental gear mesh (f_{m1}) and planet gear (f_{p1}) of a single-stage planetary gearbox are defined in Table. 1. z_s and z_r are the number of gear teeth on the sun gear and the ring gear respectively, and N_p is the number of planet gears.

Table 1: Characteristic fault frequencies of a planetary gearbox.

Parameter	Value
Sun gear (s_1)	$f_{s1} = \frac{N_p z_r}{z_s + z_r} f_{in}$
Carrier plate (c_1)	$f_{c1} = \frac{z_s}{z_s + z_r} f_{in}$
Fundamental gear mesh	$f_{m1} = \frac{z_r z_s}{z_s + z_r} f_{in} = z_r f_{c1}$
Planet gear ($p_{1,i}, i = 1, 2, 3$)	$f_{p1} = \frac{4z_s z_r}{z_r^2 - z_s^2} f_{in} = \frac{4z_r}{z_r - z_s} f_{c1}$

These equations are also valid for a two-stage planetary gearbox. However, the frequency of the first stage planet carrier f_{c1} replaces the shaft frequency f_{in} on the second stage.

2.4 Statistical features

Determining the characteristic features of a signal is a type of feature extraction, which is an essential preprocessing technique used for fault diagnosis. Motor currents can be contaminated by inverter harmonics and exogenous disturbances from the power supply, which may cause the diagnostic decision not to be reliable [10]. Statistical features may be extracted additionally to the CFs for more reliable fault diagnosis. The statistical features can be extracted from current and vibration signals, and are often calculated on both the time- and frequency-domain. Some common time-domain features are RMS (t_1), standard deviation (t_2), variance (t_3), 3rd order central moment (t_4), skewness (t_5), kurtosis (t_6) and crest factor (t_7). Similar statistical features can also be extracted from the frequency-domain ($f_1 - f_7$). Equations of these statistical features are summarized in Table 2.

Table 2: *Statistical features derived from the time-domain and frequency-domain of signals.*

Time-domain	Frequency-domain
$t_1 = \sqrt{\sum_{n=1}^N x(n)^2 / N}$	$f_1 = \sqrt{\sum_{k=1}^K s(k)^2 / K}$
$t_2 = \sqrt{\sum_{n=1}^N (x(n) - \bar{x})^2 / N}$	$f_2 = \sqrt{\sum_{k=1}^K (s(k) - \bar{s})^2 / K}$
$t_3 = \sum_{n=1}^N (x(n) - \bar{x})^2 / N$	$f_3 = \sum_{k=1}^K (s(k) - \bar{s})^2 / K$
$t_4 = \sum_{n=1}^N (x(n) - \bar{x})^3 / N$	$f_4 = \sum_{k=1}^K (s(k) - \bar{s})^3 / K$
$t_5 = \frac{\sum_{n=1}^N (x(n) - \bar{x})^3}{(N-1)t_2^3}$	$f_5 = \frac{\sum_{k=1}^K (s(k) - \bar{s})^3}{(K-1)f_2^3}$
$t_6 = \frac{\sum_{n=1}^N (x(n) - \bar{x})^4}{(N-1)t_2^4}$	$f_6 = \frac{\sum_{k=1}^K (s(k) - \bar{s})^4}{(K-1)f_2^4}$
$t_7 = \frac{\max(x(n))}{t_1}$	$f_7 = \frac{\max(s(k))}{f_1}$
where $x(n)$ is the signal time series, $n = 1, 2, \dots, N$ and \bar{x} is the mean.	where $s(k)$ is the frequency spectrum, $k = 1, 2, \dots, K$ and \bar{s} is the mean.

Moments are defined as the specific quantitative measure of the shape of a function. If this function is a probability distribution, the second-order central moment of the distribution is known as variance, which is a measure of how far each value in the data set is from the mean. Standard deviation is a measure of how far the signal fluctuates from the mean and is simply the positive square root of the variance. The standard deviation of current measurements only measures the AC portion of a signal. The RMS value measures both the AC and DC components, so if a signal has no DC components, the RMS value is identical to the standard deviation. The crest factor corresponds to the ratio between the RMS value and the peak value of a signal. Central moments normalized by the standard deviation raised to the order of the considered moment is known as standardized moments [26]. The 3rd order standardized moment, usually referred to as skewness, is a measure of symmetry, or more precisely, the lack of symmetry of a signal. The 4th order standardized moment, also referred to as kurtosis, is a measure that describes the shape of a distribution's tail relative to the tail of a normal distribution.

2.5 Extended Park's vector approach

The traditional Park's Vector Approach (PVA) relies on the monitoring of the three-phase currents (i_a , i_b , i_c) of the motor. By performing a Park transformation, the time-domain components of a three-phase system in an abc reference frame is converted to direct and quadrature (dq) components in a rotating reference frame, as shown in Fig. 3. The transformation converts AC signals into DC signals, which reduces the computational effort and simplifies control design and analysis of three-phase machines. The advantage of using the PVA is that sinusoidal signals with angular frequency will be seen as constant signals in the dq reference frame.

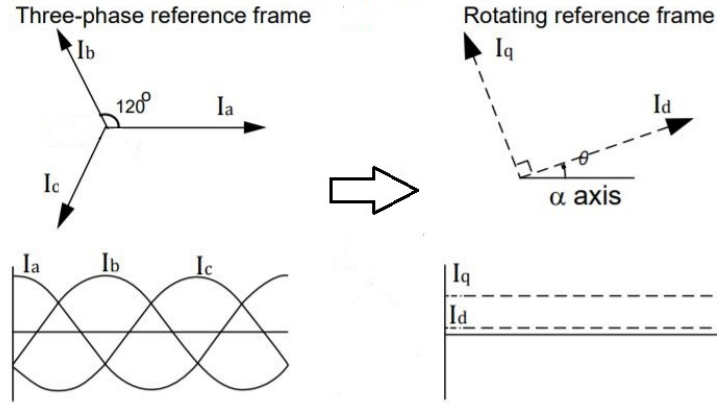


Fig. 3: The two reference frames in a Park's transformation.

The motor current's Park's vector components, i_d and i_q , are calculated by:

$$\begin{aligned} i_d &= \sqrt{\frac{2}{3}}i_a - \sqrt{\frac{1}{6}}i_b - \sqrt{\frac{1}{6}}i_c \\ i_q &= \sqrt{\frac{1}{2}}i_b - \sqrt{\frac{1}{2}}i_c \end{aligned} \quad (5)$$

The Extended Park's vector approach relies on the spectral analysis of the AC level of the PVA. Under healthy conditions, the EPVA signature will be clear from any spectral component, that is, only a DC value is present in the current Park's vector approach [27]. The modulus of the quadrature axis current-space vector, where i_p is a DC offset under healthy conditions [28], the EPVA is given by:

$$i_p = |i_d + ji_q| = \sqrt{i_d^2 + i_q^2} \quad (6)$$

Under abnormal conditions, for instance, in the presence of bearing faults, i_p will no longer be a DC offset, because the EPVA signature will contain spectral components. The EPVA is beneficial for diagnostics because it automatically eliminates the fundamental supply frequency f_s , which is the major frequency component of the line current [10].

2.6 Machine learning

Over the years, there has been developed a wide range of different machine learning algorithms for various applications. There are essentially three types of machine learning algorithms: supervised, unsupervised, and reinforced learning algorithms. Supervised learning is when the training data is labeled, where an example of this could be blood samples from patients, where the labels could be cancerous or non-cancerous patients. Two subclasses within supervised learning are classification and regression. In regression, an unknown value is predicted based on trends in the data, and in classification, the class of an unknown data point is predicted. Unlike supervised learning, unsupervised learning does not have labeled data, and the goal is to find underlying structures in the data. Two subclasses within unsupervised learning are clustering and dimensionality reduction. Dimensionality reduction makes the data more interoperable by extracting useful data from large data sets, or by creating new parameters which are a combination of numerous other parameters. In clustering, the goal is to find parameters that have similar attributes. In reinforcement, an algorithm is trained by rewarding actions in an environment to maximize a cumulative reward. An overview of these base types can be seen in Fig. 4, as well as some typical use cases.

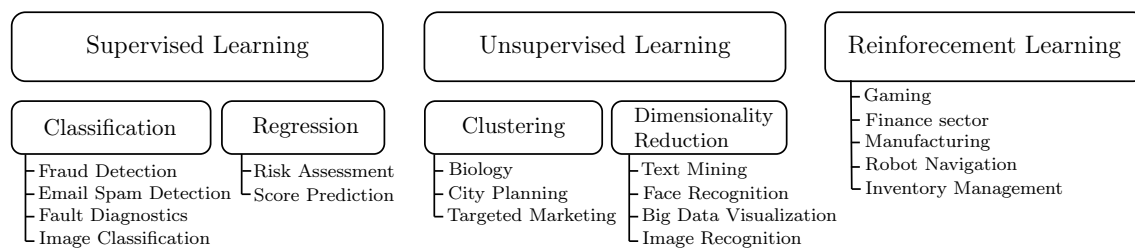


Fig. 4: *Diagram of the three basic machine learning types, as well as their sub types and some use case examples.*

In machine learning are training sets, testing sets, and validation sets common terminology used to describe the parts of the data and its intended use. The training set is the data used for training an algorithm, while the testing set is the data often held out of the training set and used only to validate the algorithm’s performance. The validation set is often a subset of the training data used to optimize the performance of the algorithm during training.

Overfitting and underfitting are two common challenges during training, where a graphical depiction of what the terms describe are shown in Fig. 5. Overfitting refers to a model that fits the training data too well, often resulting from an overly complicated model. When a model fits more data than it needs, it will start learning from the noise and inaccuracies of the data set, which will negatively impact the model’s ability to generalize. An overfitted model has good performance on the training data ,and poor performance on the test set since the model will memorize the training data instead of learning from it. Underfitting refers to a model that can neither model the training data nor the new data, which often is a result of a too simple model. Signs of underfitting is poor performance on the training data and poor predictions on new data sets. Underfitting is often not discussed as an underfit model will have poor performance on the training data, making it easy to detect. Overfitting, on the other hand, is not that easy to detect and sometimes goes unnoticed, which leads to unrealistic prediction accuracy [29].

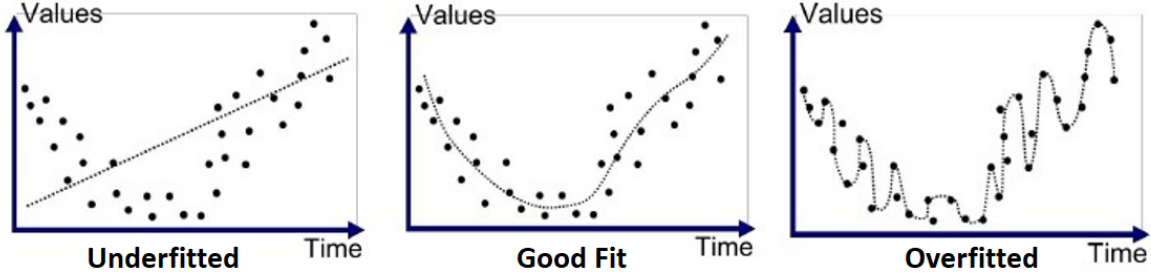


Fig. 5: Shows an example of underfitting, overfitting and a good fit [30].

2.6.1 Support vector machine

An SVM is a supervised learning model developed by Vladimir Vapnik, and the method described in this section is derived from his paper on the subject [31]. An SVM is a binary classifier, meaning that it can only distinguish between two different classes. The classification of a sample is defined by its position relative to an n -dimensional hyperplane, H , called the decision boundary, where n is the number of features of the sample. H is defined as the set of points \vec{x} that satisfies the following equation:

$$H(\vec{w}, b) = \vec{w} \cdot \vec{x} + b = 0 \quad (7)$$

where \vec{w} is the n -dimensional normal vector of the plane, and b is the planes offset from the origin along \vec{w} . Training an SVM is the process of finding the values for \vec{w} that creates the largest margin between the two classes of samples in the training data, referred to as positive and negative samples in this section. The separation of the two classes is defined by two parallel hyperplanes called the marginal planes, as seen in Fig. 6. The marginal planes are bound by the following equations:

$$\begin{aligned} \vec{w} \cdot \vec{x} + b &\geq 1 && \text{for positive samples} \\ \vec{w} \cdot \vec{x} + b &\leq -1 && \text{for negative samples} \end{aligned} \quad (8)$$

These equations can be combined to get Eq. (9) by introducing a labeling variable, y_i , which is ± 1 corresponding to the class of the i -th sample. The equations also exclude any samples in the training data from being within the marginal lines. However, this can be allowed by introducing a slack variable, ξ_i . The variable describes the normal distance of a sample with respect to its corresponding marginal hyperplane. It only describes the distance of samples on the wrong side of the marginal planes, as it is limited to be a positive number, which is also illustrated in Fig. 6.

$$y_i (\vec{w}_i \cdot \vec{x}_i + b) \geq 1 - \xi_i, \quad \xi_i \geq 0 \quad (9)$$

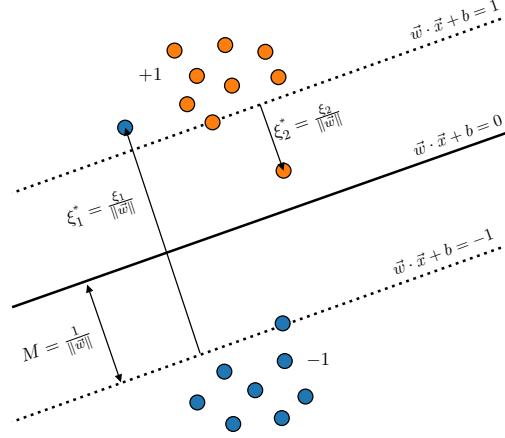


Fig. 6: Illustration showing how samples are divided by the decision boundary, where positive samples are represented by red dots, and negative samples by blue dots.

The margin, which is the distance between the marginal planes, is inversely proportional to the magnitude of \vec{w} . This gives the following optimization problem, subject to the constraint in Eq. (9):

$$\min_{\vec{w}, b, \xi} \left\{ \frac{1}{2} \|\vec{w}\|^2 + C \sum_{i=1}^N \xi_i \right\} \quad (10)$$

where C is the penalty strength for samples being on the wrong side of the marginal planes. It determines the balance between the allowed number of misclassifications in the training data and the magnitude of the margin. A large value creates a stiff model prone to overfitting, and a small value makes a less accurate model. This parameter is typically tuned using hyperparameter optimization.

The optimization problem, shown in Eq. (10), is solved with the use of Lagrangian multipliers, giving the following equation:

$$\begin{aligned} \max f(\alpha_1 \dots \alpha_n) &= \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n (\alpha_i \alpha_j y_i y_j (\vec{x}_i \cdot \vec{x}_j)) \\ \text{subject to: } &\sum_{i=1}^n (\alpha_i y_i) = 0, \text{ and } 0 \leq \alpha_i \leq C \end{aligned} \quad (11)$$

where α_i are the Lagrangian multipliers. This equation can be efficiently solved with a quadratic programming algorithm [32]. It can also be seen that the penalty parameter C determines the upper bound of the multipliers. This is why the parameter is often referred to as the box constraint, as it confines the Lagrangian multipliers to be within 0 to C .

$$\vec{w} = \sum_{i=1}^n (\alpha_i y_i \vec{x}_i) \quad (12)$$

Eq. (12), found by taking the partial differential of the Lagrangian function, shows that \vec{w} can be represented as a sum of linear equations dependent on a_i . Most of the Lagrangian multipliers have a maximum of zeros, so \vec{w} is defined by only a few samples. It is these samples with non-zero Lagrangian multipliers that are referred to as the support vectors. By using these support vectors, and the definition of the decision boundary, an unknown sample can be classified by checking the sign of Eq. (13), where \vec{u} is an n -dimensional vector containing the features of the unknown sample.

$$\sum_{i=1}^n (\alpha_i y_i \vec{x}_i) \cdot \vec{u} + b \quad (13)$$

The method described above is only able to produce a linear decision boundary, making the classifier unsuitable for classifying non-linearly separable data. However, with the use of a kernel function, shown in Eq. (14), SVMs can be used to separate non-linearly separable samples as well. This works by using a transform function, $\phi()$, to transform the samples from what is referred to as input space into feature space where the points become linearly separable. The kernel function returns the dot product of the two vectors from feature space without having to transform the vectors into that space.

$$K(\vec{x}_i, \vec{x}_j) = \varphi(\vec{x}_i) \cdot \varphi(\vec{x}_j) \quad (14)$$

One of the simplest kernels is the polynomial kernel shown in Eq. (15), where r is the coefficient of the polynomial, and d is the polynomial degree.

$$K(\vec{x}_i, \vec{x}_j) = (\vec{x}_i \cdot \vec{x}_j + r)^d \quad (15)$$

A common kernel to use is the Gaussian kernel, also known as the radial basis function (RBF) kernel, shown in Eq. (16), where σ is a free parameter. This kernel is commonly used because it compares the relationship between two points in infinite dimensions. This can be proven by taking the Taylor series expansion of the function with the kernel scale, γ , set to 0.5, this is shown in Appendix D. It can then be seen that each term in the Taylor series is equivalent to a polynomial kernel, shown in Eq. (15), with the parameter r being set to 1 and d ranging from zero to infinity.

$$K(\vec{x}_i, \vec{x}_j) = e^{-\gamma \|\vec{x}_i - \vec{x}_j\|^2}, \quad \gamma = \frac{1}{2\sigma^2} \quad (16)$$

Normally, it is only possible to use an SVM to distinguish between two classes as it is a binary classifier, however, it can also be used for multiclass classification by using it in an error-correcting output codes (ECOC) model. ECOC is a way to solve a multiclass problem by splitting the problem into multiple binary classification problems so that an SVM or another binary classifier can be used. Different ways of splitting up the multiclass problem have been devised, with the most common being one vs. all (OVA) and one vs. one (OVO) [33]. Both methods work by constructing a k -by- l matrix, where k is the number of classes, and l is the number of SVMs. When using OVO, the matrix is filled such that each column contains one positive class while the others are negative, as shown in Fig. 7a. When using OVA, the matrix is filled such that each column contains one positive and one negative class, while the others are ignored, as shown in Fig. 7b.

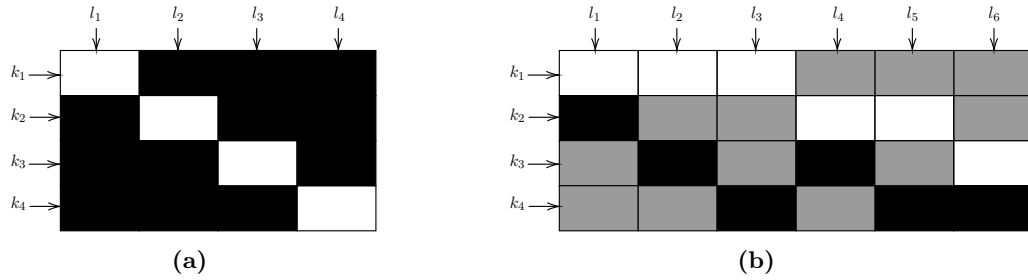


Fig. 7: Showing the class matrix for (a) OVA and (b) OVO, where the white squares represent a positive label, the black represents a negative label, and the gray squares are ignored.

The class of a new sample can be assigned by using Eq. (17), where \hat{k} is the predicted class, $m_{k,l}$ is the (k, l) element of the matrix, s_l is the prediction score of a positive class of SVM l , and g is the binary loss function. The binary loss function is most commonly a quadratic loss function [34].

$$\hat{k} = \underset{k}{\operatorname{argmin}} \frac{\sum_{l=1}^L |m_{k,l}| g(m_{k,l}, s_l)}{\sum_{l=1}^L |m_{k,l}|} \quad (17)$$

2.6.2 Cross-validation

Cross-validation is an important part of training a machine learning algorithm. Cross-validation ensures that the algorithm is trained on multiple combinations of the data to fully utilize the variation in the data and to minimize the risk of overfitting. There are multiple ways to perform cross-validation. One method is leave-one-out cross-validation, which explores all possible combinations of training and testing data. However, this is rarely a viable option, as the computational load quickly rises with the number of samples. K-fold cross-validation is a more common method, where the data is split into sets of samples called folds. This is done by first randomizing the data and then splitting it into k number of equally sized folds. One of these folds is treated as the testing data, and the rest is treated as the training data. This is repeated until all the k folds have been used as a test set. The accuracy of the algorithm is determined by the average of the accuracy of all the runs. The simplest form of cross-validation is holdout-cross-validation, where the data is simply divided into a test and a training set. A common ratio is 70/30 for training and testing, respectively [35]. A more versatile approach is a combination of holdout and k-fold, where the data is first divided into a test and a training set, and then the algorithm is trained on the training set using k-fold [36].

2.6.3 Hyperparameter optimization

In machine learning are parameters that determine the learning behavior of the model called hyperparameters. These hyperparameters can affect the model's accuracy, training time, risk of overfitting, etc. For an SVM are the hyperparameters, the box constraint, and the kernel scale. The kernel function and multiclass method are also hyperparameters, but these are usually determined beforehand and kept constant for the optimization. The most common optimization strategies are grid search, random search, and Bayesian optimization. Grid and random searches are simple models that distribute a number of points where the coordinates of the points correspond to the values of the hyperparameters, as seen in Fig. 8. The SVM is tested with the hyperparameters of each point, and the one with the highest accuracy is chosen. Bayes optimization is an adaptive model that distributes points based on the results of the previous points. This generally gives a better model than grid and random search [37]. It also allows for the optimization of additional aspects such as training time and improvement rate, as it is an iterative process.

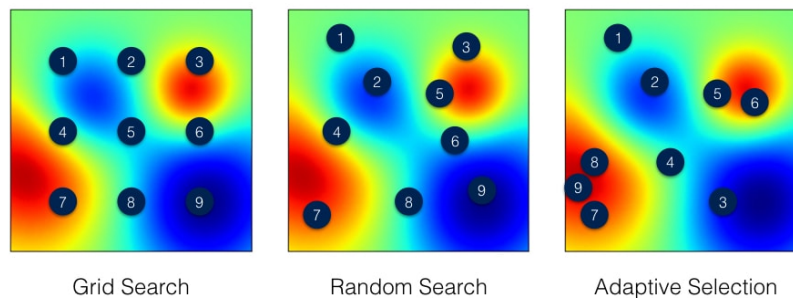


Fig. 8: Figure comparing grid search, random search, and Bayesian optimization. Each point corresponds to a set of hyperparameters. The color of the contour plot is the classification accuracy of the classifier, with warmer colors indicating higher accuracy [38].

2.6.4 Principal component analysis

PCA is a data transformation technique commonly used for dimensionality reduction and as a preprocessing step for machine learning [39]. PCA is performed by separating the features of the data into principal components (PCs), where the first PC contains most of the variance in the data, and each succession contains less variance. The PC with the most variance tends to be the most useful when trying to cluster or classify the data. It can, therefore, be beneficial to remove the last PCs to reduce the number of features. It is common to exclude the last PCs, after the cumulative variance of the first PCs reaches 90% [40]. In a strongly correlated data set, as shown in Fig. 9, where most of the variance is present in PC1, one could remove PC2 and reduce the data set from a 2-D to a 1-D data set without a substantial loss in useful information in the data.

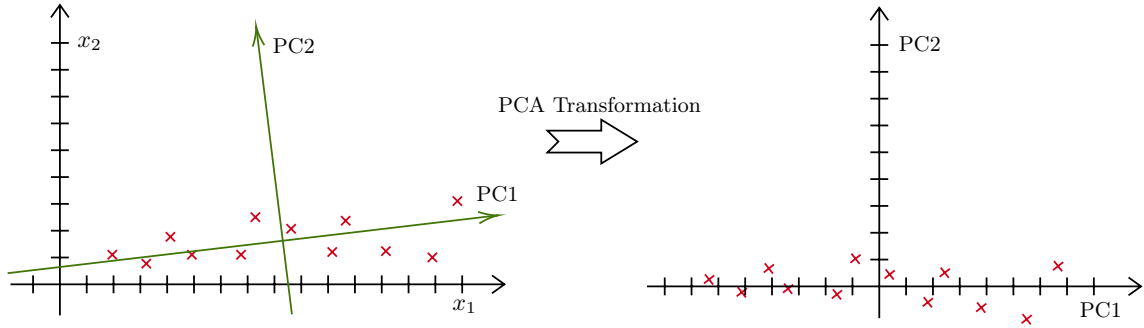


Fig. 9: *Illustrating the concept of PCA transformation.*

All data must be standardized before it is used in PCA, so the mean of all features has to be centered around zero. This is done to ensure that the first PC describes the direction of most variance and not just the mean of the data, and to minimize the root mean square [41].

The PCs are found by first creating a covariance matrix denoted as Σ , seen in Eq. (18), where (Cov) is the covariance between the n features. The variance of each feature can be found along the diagonal. The eigenvalues and eigenvectors are then calculated for the matrix. The eigenvalues indicate the magnitude of the spread in the data, and the eigenvectors indicate the direction. The first PC is the eigenvector with the largest eigenvalue. The eigenvectors will always be orthogonal to each other since they come from a symmetrical matrix.

$$\Sigma = \begin{bmatrix} \text{Cov}(x_1, x_1) & \text{Cov}(x_1, x_2) & \dots & \text{Cov}(x_1, x_n) \\ \text{Cov}(x_2, x_1) & \text{Cov}(x_2, x_2) & \dots & \text{Cov}(x_2, x_n) \\ \vdots & \vdots & \ddots & \vdots \\ \text{Cov}(x_n, x_1) & \text{Cov}(x_n, x_2) & \dots & \text{Cov}(x_n, x_n) \end{bmatrix} \quad (18)$$

2.6.5 Receiver operating characteristic curve

A receiver operating characteristic (ROC) curve is used to compare the true positive rate (TPR) and false positive rate (FPR) as the criterion changes for binary classifiers. Each point on a ROC curve represents a threshold and the calculated TPR and FPR of the threshold. The formulas for TPR and FPR are shown in equations (19) and (20), respectively, where TP is the number of true positives, FN is the number of false negatives, FP is the number of false positives, and TN is the number of true negatives. Starting from the right of the ROC plot, shown in Fig. 10, the threshold is set low, such that all samples are classified as positive. This means that all the positive samples are classified correctly giving a TPR of 1, which also means that all negative samples are classified incorrectly giving an FPR of 1. The threshold is then increased towards the left of the plot until the threshold is set so high that all samples are classified as negative. This means that all positive samples are classified incorrectly giving a TPR of 0. It also means that all negative samples are classified correctly giving an FPR of 0. The plot is useful to determine what the decrease in accuracy would be, for example, if it was more important for the classifier to have a false negative or false positive rate of close to zero. This is, for instance, useful when classifying a disease where the rate of false negatives is more important than overall classification accuracy. The curve is also a good indicator of the overall performance of the classifier. A perfect classifier would have a curve that lies on the upper horizontal and left vertical axis, and a random guess would produce a curve close to the diagonal of the plot. This means that the area under the curve (AUC) is a good indicator of a classifier's performance [42].

$$\text{TPR} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad (19)$$

$$\text{FPR} = \frac{\text{FP}}{\text{FP} + \text{TN}} \quad (20)$$

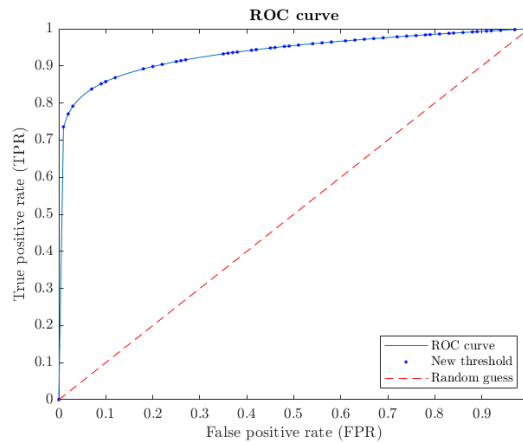


Fig. 10: Example of an ROC curve.

3 Method

3.1 Laboratory setup

A laboratory setup was built to study faults in the motor and planetary gearbox of the pitch system. A schematic representation of the laboratory setup, where the main components are the 'pitch drive', 'load', and data acquisition equipment, is shown in Fig. 11. This schematic is based on the research of Kandukuri et al. [43]. The 'pitch drive' system consists of a 1.1 kW three-phase 4-pole IM, referred to as the pitch motor, which is coupled with a two-stage planetary gearbox with a gear ratio of 1:48.1. The design parameters of this gearbox are shown in Table 3. The gearbox has a single-row rolling element bearing of type 'SKF 16012' at the input shaft with 17 rolling elements, 77.495 mm pitch diameter, 7.938 mm rolling element diameter, and a 0° contact angle. For control of the pitch motor, a commercial variable frequency drive (VFD) with field oriented control (FOC) is connected to the pitch motor. Three Hall-effect current sensors are connected to the cable from the VFD to the pitch motor for motor diagnostics. The key specifications for the current sensors are shown in Table A.1 in Appendix A. The 'load' system consists of a prime-mover, referred to as the 'load motor', coupled with a bevel-planetary-helical (BPH) gearbox with a gear ratio of 1:27.1. The BPH gearbox is coupled with the pitch drive, and is used to apply variable load torque on the pitch drive. The 'load' system consists of a 3 kW three-phase 8-pole IM, which is also controlled using a commercial VFD with FOC. Further specifications for the pitch and load motor can be found in Table A.2 in Appendix A. The signals from the frequency drives and the signals from the current sensors were measured with a 6211 National Instruments data acquisition board (NI DAQ) and logged with a PC running Matlab. A 5 V power supply was used to supply the sensors, since the current rating of the 5 V output of the DAQ was insufficient to drive multiple sensors.

Table 3: *Parameters of the two-stage planetary gearbox.*

Parameter	Value
No. of stages	2
No. of planets per stage	3
No. of ring gear teeth (z_r)	95
No. of planet gear teeth (z_p)	38
No. of sun gear teeth (z_s)	16
Gear ratio per stage	6.9375
Overall gear ratio	48.1289

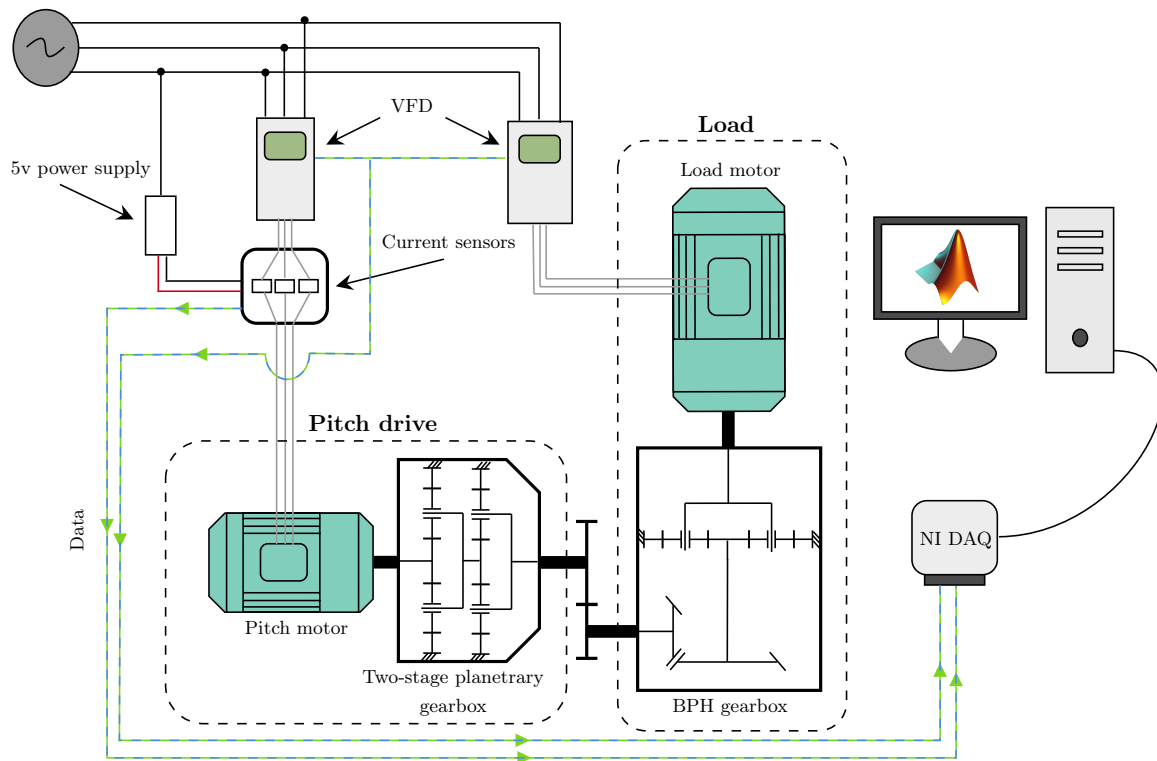


Fig. 11: A schematic representation of the laboratory setup showing the scaled pitch drive and the load along with the current sensors.

For condition monitoring of the two-staged planetary gearbox, two artificially seeded fault conditions and a healthy condition were chosen. One of the artificially seeded fault conditions was an outer race fault in the bearing at the input shaft, where a 2 mm hole was drilled through the outer race of the bearing, as depicted in Fig. 12a. An outer race fault was chosen, as it is more likely for damage to occur on the outer race than the inner race. This is because the inner race of the bearing is rotating while the outer race is fixed, which causes a centrifugal force acting on the roller elements, pushing them towards the outer race [10]. The other artificially seeded fault condition chosen for monitoring was a partially cracked tooth in one of the first stages of the planet gear, as illustrated in Fig. 12b. This was chosen because planet gears are among the components with a high failure rate, which also experiences the most complex motion. Two planetary gearboxes were used for the condition monitoring, one healthy gearbox where the healthy bearing could be exchanged with the artificially seeded faulty bearing, and one gearbox with a planet gear fault. These health conditions will be referred to as HLT, BRG, and PLT for healthy, bearing fault and planet gear fault, respectively.

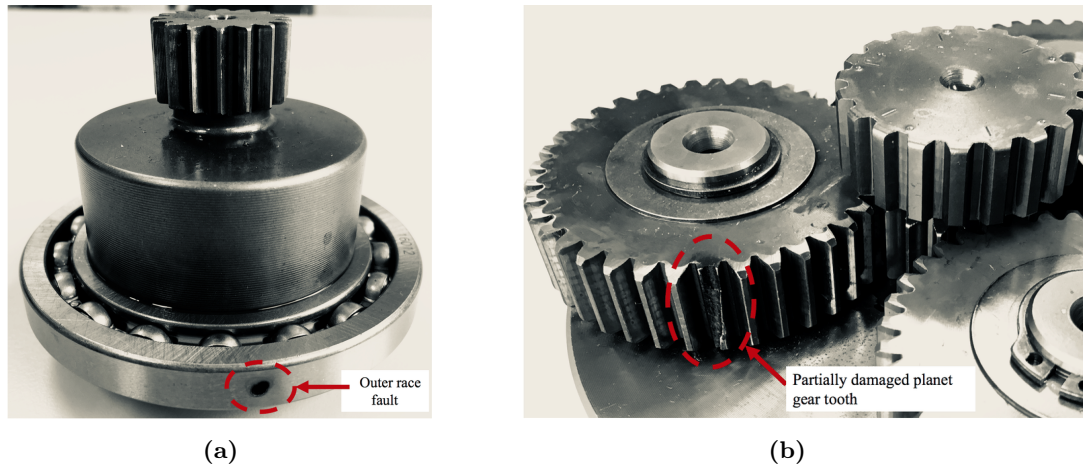


Fig. 12: Shows the artificially seeded fault in (a) the bearing and (b) the planet gear [44].

3.2 Data acquisition

The planetary gearbox with healthy conditions was tested at various steady speeds, from 600 to 1425 RPM at the pitch motor, and constant loads ranging 3 to 83% of the pitch motor rated load. The tests on the healthy gearbox were performed with a sampling frequency of 20 kHz where the data for each case was collected for a period of 240 seconds.

The two artificially seeded fault conditions, mentioned in Section 3.1, were tested similarly to the healthy gearbox. The gearbox with bearing fault was tested at various steady speeds, ranging from 1050 to 1400 rpm at the pitch motor and constant loads ranging from 0 to 50% of the rated motor load. The gearbox with planet gear fault was tested at various steady speeds, ranging from 800 to 1400 RPM at the pitch motor and constant load ranging from 25 to 65% of the rated motor load. Both of the gearboxes with an artificially seeded fault condition were tested with a sampling frequency of 30 kHz, where the data for each case was collected for a period of 60 seconds. For the gearboxes with a fault condition, most tests were performed with a constant load of 25% of the rated motor load. However, the constant loads for the healthy gearbox were chosen randomly. The speeds for healthy and fault conditions were chosen evenly within the operational limits.

3.3 Data overview

A flowchart representing the data processing system is shown in Fig. 13. The outline of the boxes represent the structure of the data being used, where dashed outlines are complete run files, solid outlines are samples, and dotted outlines are features. The flowchart is divided into three sections, where each section represents how the data is being processed. In the data preparation section, which will be described further in Section 3.4, raw data was collected, standardized, and divided into samples for calculations. In the calculation and feature selection section, additional time- and frequency-domain parameters were calculated for each sample, which is described in greater detail in Section 3.5. The collected features from all the samples were used in PCA and then split into testing and training sets before being used to train the SVM in the SVM model section. This will be further explained in Section 3.6.

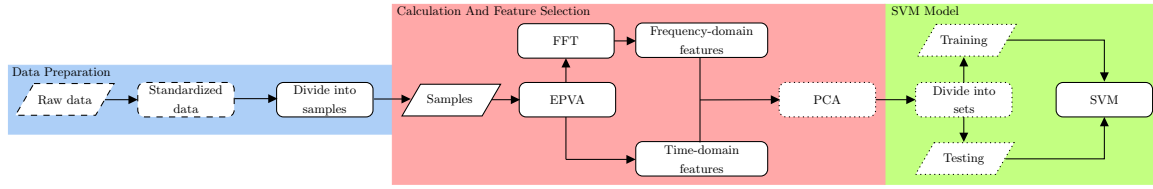


Fig. 13: Flowchart with three sections representing the data processing system.

3.4 Data preparation

The raw data collected using the method described in Section 3.2 was imported into Matlab to standardize its structure to make it easier to manipulate. The standardized data was then divided into 5-second samples, with a total of nearly 800 samples. The distribution of samples for each health condition is shown in Table 4. The sample time of 5 seconds was chosen to include at least one rotation of the output shaft of the gearbox. Since the slowest rotational speed for the tests was 600 rpm, and given the gearing ratio for each step in the two-stage gearbox, the output shaft rotational speed was 0.2 rev/s, or 4.8 s/rev. The sample time also gave an adequate frequency resolution of 0.2 Hz, which was found by dividing the sampling frequency by the number of samples. The data was then analysed to look for missing or erroneous data, as well as outliers. It was found that some runs had an uneven operating speed at the start of the run, as can be seen in Fig. 14. This led the start of some runs to have wildly different values than the rest of the run, as shown in Fig. 15. The samples highlighted in the figure were removed from the data set.

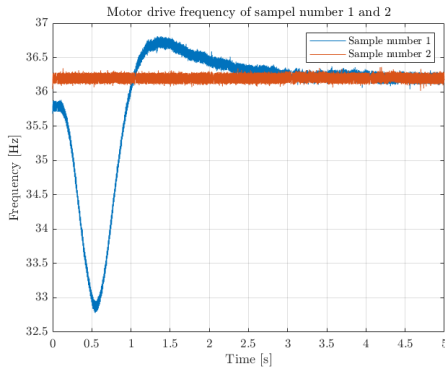


Fig. 14: Plot showing the the difference between the frequency readout from the motor drive in samples 1 and 2.

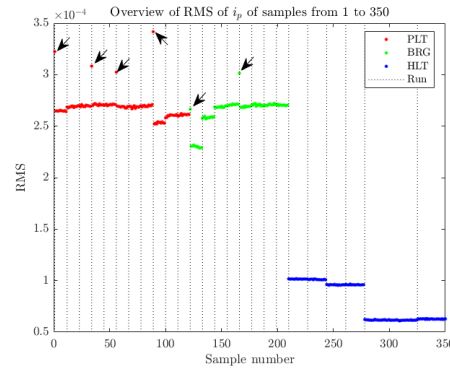


Fig. 15: Plot showing the RMS value of i_p of samples 1 to 350 with vertical lines indicating when a new run begins.

Health condition	Number of samples
PLT	121
BRG	88
HLT	585

Table 4: Showing the number of samples per health condition.

3.5 Feature selection

The flowchart shown in Fig. 16, illustrates how features were chosen through an iterative process. Methods from selected papers of the literature review were tested on a set of preliminary samples. This was done as it would otherwise be difficult to compare health conditions, because the current spectrum changes with speed and load. The samples chosen for the preliminary samples had an operating speed of 1200 RPM, and a load of 25% for the two fault conditions. The healthy sample had the same operating speed, but a load of 16%, as this was the closest matching. The results from the preliminary analysis were compared with the results from all samples to see if the features gave consistent results.

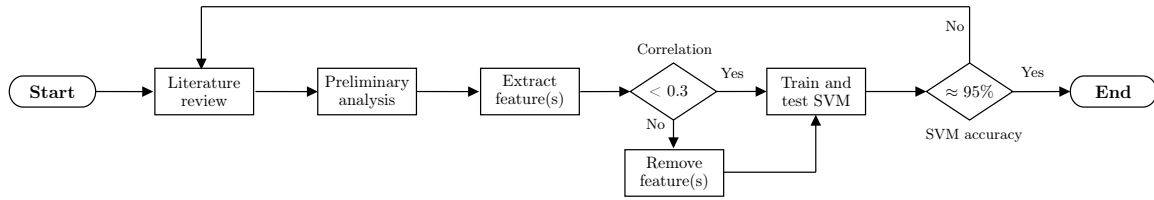


Fig. 16: Flowchart of the feature selection process.

The following paragraphs describe how the features were extracted from the data in the 'Extract feature(s)' node of the flowchart. The i_p of each sample was calculated using the EPVA method described in Section 2.5. EPVA was used as it achieves multiple operations at once. It acts as a notch filter, removing the fundamental supply frequency of the motor drive, as well as removing the modulation of the motor drive on the CFs. This was beneficial as some of the CFs lie close to the drive frequency, and would otherwise be lost in the noise surrounding the drive frequency. EPVA also works as a form of dimensionality reduction, capturing the information of the three phases into one variable.

FFT was used to find the frequency spectrum of i_p of the samples in the preliminary study, which were further analysed to look for differences between the three health conditions. The harmonics of the motor drive was compared with the spectrum to see if they were present in i_p , and to what degree. The CFs corresponding to the operating speed were calculated, and compared with the peaks of the frequency spectrum, to see if the origin of some of the peaks could be attributed to the CFs. The statistical features listed in Table 2, were also calculated for i_p of the three samples, to see if there were any differences. It was early discovered that the energy in the spectrum of the healthy sample was substantially lower than the faulty ones, by comparing the spectrums of the three health conditions. This was expected based on the literature review [10]. The majority of the analysis was, for this reason, focused on finding differences to distinguish between the two faulty conditions. The methods used in the preliminary analysis were repeated for all samples to see if the findings could be generalized for all samples. The magnitude of the CFs was captured using a series of narrowband filters, each centered around a CF, where each narrowband had a width of 1.5 Hz to accommodate for inaccuracies in the shaft speed readings. This was also done for the harmonics of the motor drive.

It was important to ensure that features used in the SVM were independent of load and speed, as the motor running at a specific load or RPM does not indicate a fault. None of the healthy runs were performed with loads equal to the faulty runs, and only some of the speeds were similar, this can be seen in the overview of the runs, seen in Table C.1 in Appendix C. An SVM classifier could, therefore, distinguish between a healthy and faulty sample, based solely on RPM and load. A correlation matrix was used to ensure that the chosen features had a low correlation with load and speed. Any features with a more than weak correlation, meaning a correlation coefficient greater than 0.3, were discarded [45]. This is illustrated as the first decision node on the flowchart in Fig. 16. A Pearson correlation matrix was used, as it was assumed that a feature's correlation with load and speed would be linear.

The list of features was relatively large compared to the number of samples, introducing the risk of overfitting [46]. Therefore, PCA was used to remove the least useful features by using only the first PCs responsible for 90% of the variance. These PCs were then used to train and test the SVM classifier, where the goal was to achieve an accuracy of around 95%.

3.6 SVM model

The training of the SVM was performed by using two cross-validation methods, holdout and k-fold. Holdout cross-validation was used to create a training and testing set, with the split being 70/30. This was done by first randomizing the features and then selecting 70% for training, while the rest was used for testing. K-fold cross-validation was also used during training to create test and validation sets for hyperparameter optimization. The hyperparameter optimization was performed using Bayesian search to optimize the coding matrix, box constraint, and kernel scale.

Not all the samples from the healthy runs were used in the training, as there was a disproportionate amount of healthy samples compared to faulty ones. A subset of 90 healthy samples was, therefore, randomly chosen to be used in the training instead. As mentioned earlier, it was discovered that the SVM could easily classify healthy samples. The main focus was, therefore, finding features that distinguished the two fault conditions.

The relatively small amount of data became a problem during the training of the SVM, as the problem of overfitting the model to the run became apparent because samples from the same run were almost identical, meaning that samples in the training and testing set were very similar. All samples from at least one of the runs were held out of the training set to make sure the model is not overfitted to the data. The low number of samples meant that almost 50% of the test set would consist of samples from one run, which is not optimal either. An alternative solution was chosen to overcome this issue, where all samples from one run of the HLT and PLT health conditions were held out from the training set, as there were more samples available compared to BRG. A mix of the two was then used in the test set to get an even distribution of samples from all runs. There would be a clear unevenness in misclassifications of PLT compared to BRG if the model was, in fact, overfitted due to this issue. It was thought unlikely that this would affect the misclassification of HLT because the fault condition was so easy to classify.

4 Results and discussion

4.1 Preliminary analysis

This section will present and discuss the results from the preliminary analysis.

4.1.1 Frequency spectrum overview

The plot shown in Fig. 17a shows the frequency spectrum of the three samples chosen for the preliminary analysis. It is apparent in the plot that the frequency spectrum of the healthy sample has a significantly lower amplitude across all frequencies, while the two faulty samples are almost indistinguishable. It would be reasonable to assume that this could be due to the load of the healthy sample being lower than the faulty samples. However, Fig. 17b shows that this is not the case, as it displays a HLT sample with an operating speed of 1425 RPM and a load of 69%, and the amplitude is still lower. Even though the overall amplitude is noticeably larger compared with the 16% sample, it is still significantly lower than the amplitude of the faulty samples, despite the load being more than doubled. The spectrum of the healthy sample has a frequency range of 0 and 10 kHz, while the faulty samples have a frequency range of 0 to 15 kHz. This is due to the difference in sampling frequencies between the healthy and faulty samples, where the healthy sample has a sampling frequency of 20 kHz, and the fault samples have a sampling frequency of 30 kHz.

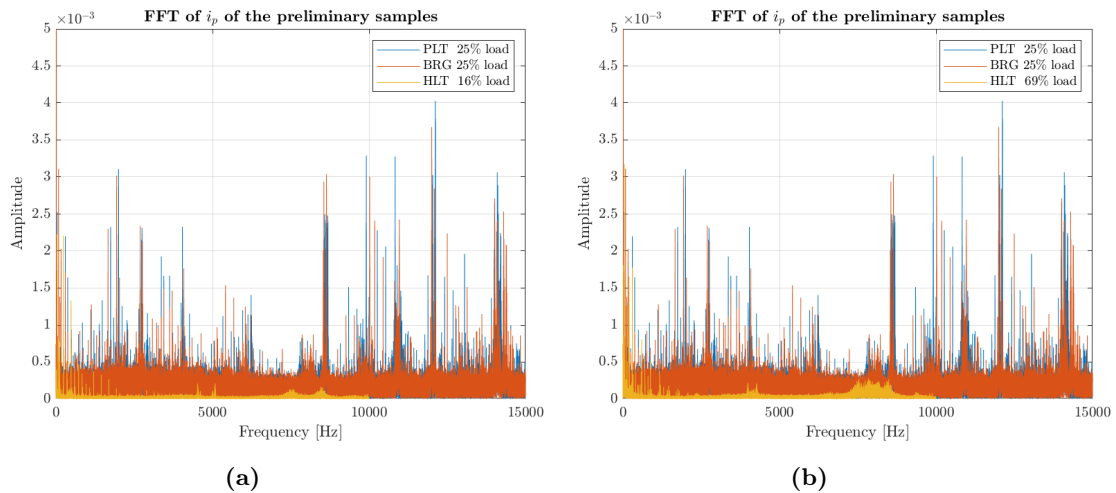


Fig. 17: Plots showing the FFT of i_p of the three samples, where (a) is the three samples of the preliminary analysis, and (b) is the two faulty samples compared with an HLT sample with a load of 69%.

4.1.2 Motor drive harmonics

The frequency spectrums of i_p of the two faulty samples are compared in Fig. 18, where the plot displays a section ranging from 0 to 400Hz. The harmonics of the motor drive is overlaid on the plot indicated by the black lines. The vertical axis of the BRG plot is reversed so that the spectrums are easier to compare, as it would otherwise be hard to distinguish the two, due to the peaks overlapping. It can be observed that the motor drive harmonics are visible

up to the 6th harmonic, where five of the harmonics have an associated peak. The amplitude of the harmonics in the PLT sample has a slightly higher amplitude, than the BRG sample, with the exception of the 2nd harmonic. This is better illustrated in Fig. 19, which compares the amplitude at the harmonics of the three samples. A single peak can be seen around 300 Hz, which is close to the first stage fundamental meshing frequency, f_{m1} . However, it is not believed that this CF is the cause of the peak, as it is present on other runs regardless of load and speed parameters. It is for this reason believed that the peak stems from extraneous noise. A potential source is thought to be ripple currents from the 5V DC power supply used to power the current sensors.

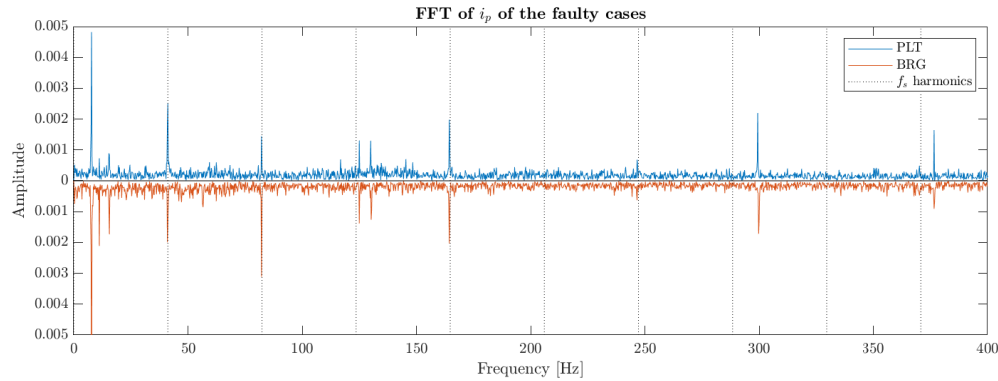


Fig. 18: Showing a plot of the FFT of i_p of the two fault conditions in the region 0-400 Hz, where the black dashed lines show the position of the motor drive harmonics.

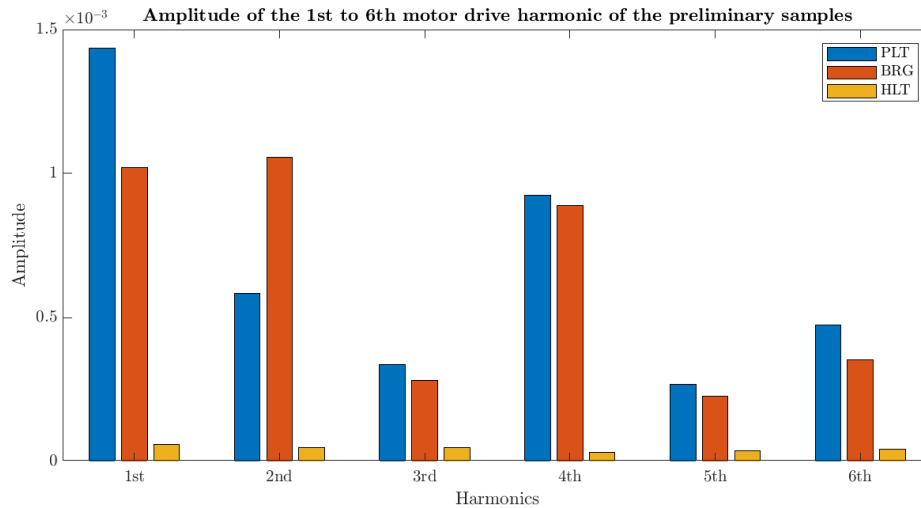


Fig. 19: Bar plot comparing the amplitudes of the motor drive harmonics of the preliminary samples.

4.1.3 Characteristic frequencies

The plot in Fig. 20 displays the same section as Fig. 18, but with the CFs, shown in Table 5, overlaid on the plot. A closeup of the region 0 to 15 Hz is also shown to display the low

frequency CFs. It can be seen that none of the unidentified peaks can be attributed to the CFs of neither the gearbox nor the bearing, besides the peak around 8 Hz. This was expected as the torque ripples produced by the bearing are small, and it was believed that they would be lost in the noise of the signal [47]. There can be observed two peaks that lie close to the CFs of both the first stage planetary gear, f_{p1} , and the second stage carrier gear, f_{c2} . The difference between these two CFs is 0.2 Hz which is close to the frequency resolution of the spectrum. A higher frequency resolution spectrum was created by increasing the sample size to 15 seconds, to identify if these peaks were a combination of the two CFs. The higher resolution plots can be seen in Figs. 21a, b, and c. The figures indicate that CF f_{p1} , and the 1st and 4th motor drive harmonic are the cause of the peaks, as their values are the closest to the peaks. In the case of the motor drives are their position known with a greater degree of certainty than the CFs, as the frequency of the motor drive is present in the data, while the shaft speed is assumed from the test run description, further supporting this. Additionally, the double peak that would be present if there were two frequencies overlapping is not visible, indicating that there is only one cause for the peaks.

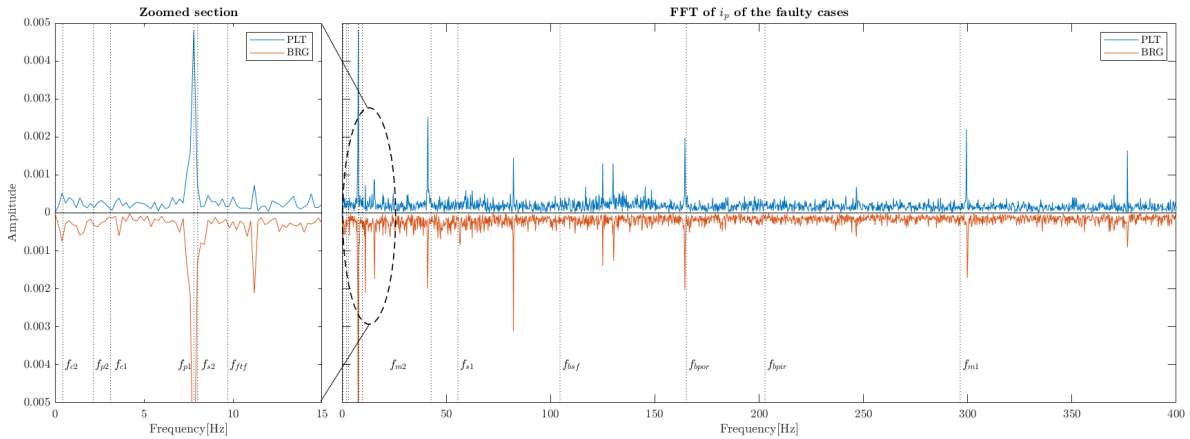


Fig. 20: Showing a plot of the FFT of i_p of the two fault conditions in the region 0-400 Hz. The black dashed lines indicate the position of the CFs shown in Table. 5.

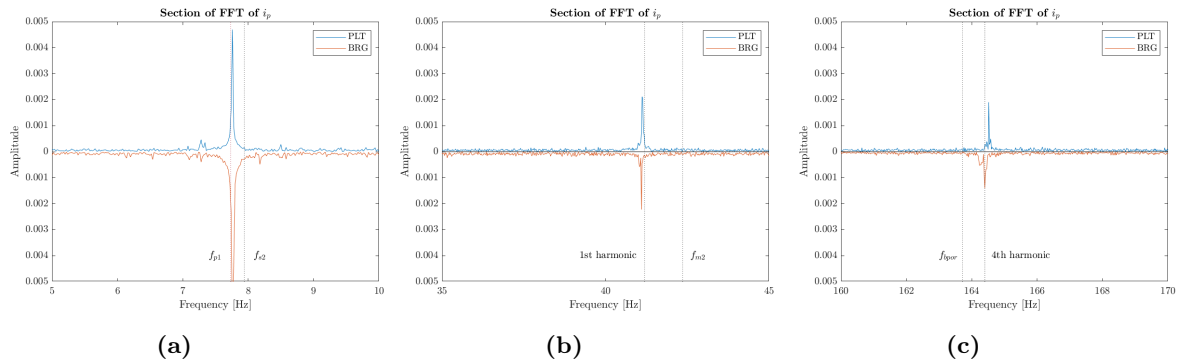


Fig. 21: Sections of Fig. 20 in the region of: (a) f_{p1} and f_{s1} , (b) the 1st harmonic and f_{m2} , and (c) f_{bpdr} and the 4th harmonic.

A bar graph, shown in Fig. 22, visualizes the differences in amplitudes of the peaks discussed in the previous paragraph. The similarities in the peaks of f_{p1} and f_{s1} are due to their similar values as mentioned earlier, leading to an overlap in the reading of the measured amplitudes. It can be seen that the f_{p1} amplitude of the BRG sample is almost twice the amplitude of the PLT sample, which seems counter-intuitive, as it would be expected that a fault in the planet gear would lead to an increased amplitude at the CF of the planet gear.

Characteristic frequency	Symbol	Value (Hz)
Second stage carrier frequency	f_{c2}	0.450
Second stage planet frequency	f_{p2}	2.163
First stage carrier frequency	f_{c1}	3.120
First stage planet frequency	f_{p1}	7.802
Second stage sun frequency	f_{s2}	8.011
Fundamental train frequency	f_{ftf}	9.714
Second stage meshing frequency	f_{m2}	42.724
First stage sun frequency	f_{s1}	55.575
Ball spin frequency	f_{bsf}	104.547
Ball pass outer race frequency	f_{bpor}	165.137
Ball pass inner race frequency	f_{bpir}	202.828
First stage meshing frequency	f_{m1}	296.400

Table 5: Showing the values of the CFs when the operating speed is 1200 RPM, arranged in descending order.

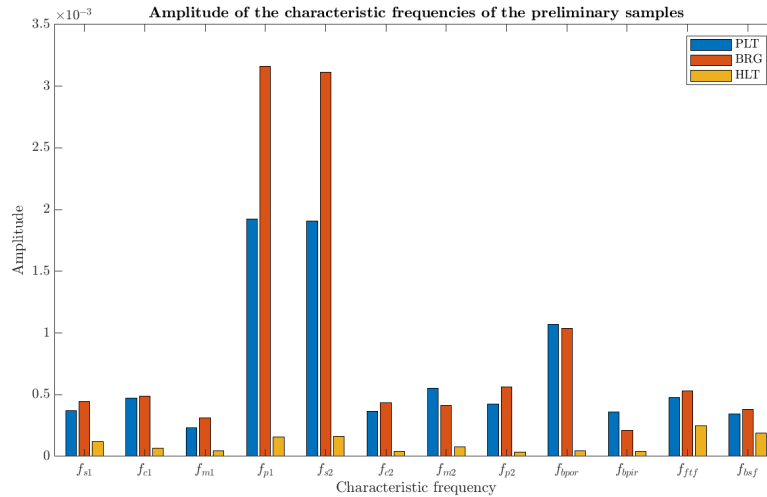


Fig. 22: Bar plot showing the amplitudes of the CFs of the samples in the preliminary analysis.

4.1.4 Statistical features

The statistical features calculated on the preliminary samples are shown in Figs. 23a and 23b, which display the time- and frequency-domain results, respectively. The results from the statistical features, t_{3-5} and f_{3-6} , have been scaled so the statistical features can be more easily compared. A clear distinction between the healthy and faulty samples can be observed, as also seen in previous results. However, there are few distinctions between the PLT and BRG sample, with the exception of f_{t4} , f_{f6} , and f_{f7} . The similarities in relative amplitude of f_{3-4} and t_{3-4} was expected as they are central moments with different orders.

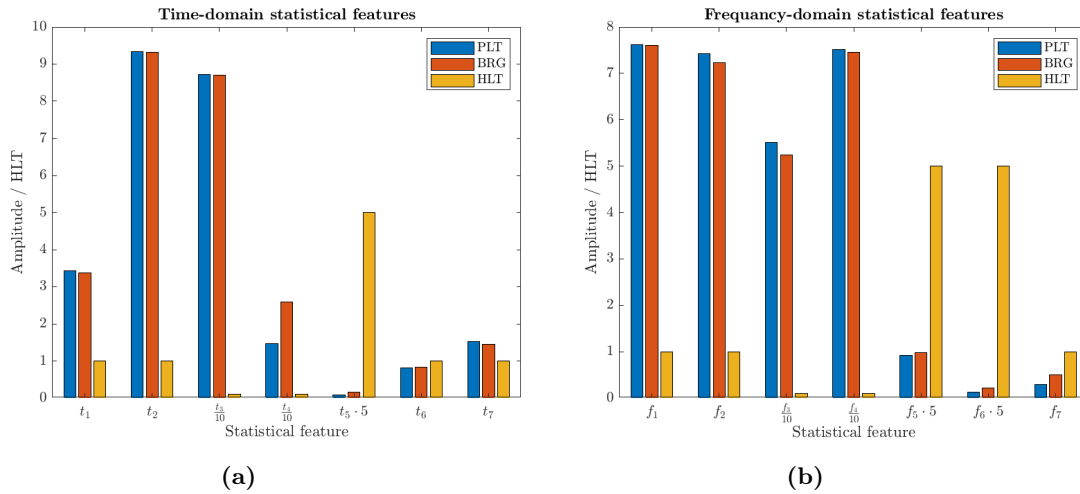


Fig. 23: Bar graph showing the values of the statistical features of the samples used in the preliminary analysis in (a) the time-domain and (b) the frequency-domain.

4.1.5 High-frequency spectrum analysis

The full frequency range of the two faulty preliminary samples is compared in Fig. 24a, where some differences in the position of the higher frequency peaks can be seen in the spectrum. These higher frequency peaks are changing over time, as seen in Fig. 24b and 24c, where the color indicates the amplitude at a specific frequency, with blue being a weak amplitude and yellow being a strong amplitude. The figure shows the FFT of all samples of the two runs containing the preliminary samples. This essentially works as an STFT with a window of 5 seconds and zero overlaps. The drift of the peaks can be difficult to spot due to the broad frequency range. However, it is clearly visible in the close-up section of the plot, displayed in Fig. 25, that shows the peak at 4.1 kHz drifting almost 50 Hz during the 60-second run. The exact cause of this is undetermined, but it can possibly be due to the temperature of the motor, gearbox, lubricant, and/or the capacitor bank of the VFD. These peaks, therefore, contribute little to the classification of the fault condition of the sample. However, their relative position to one another might be a good indicator, as well as their amplitude, but this was not investigated further.

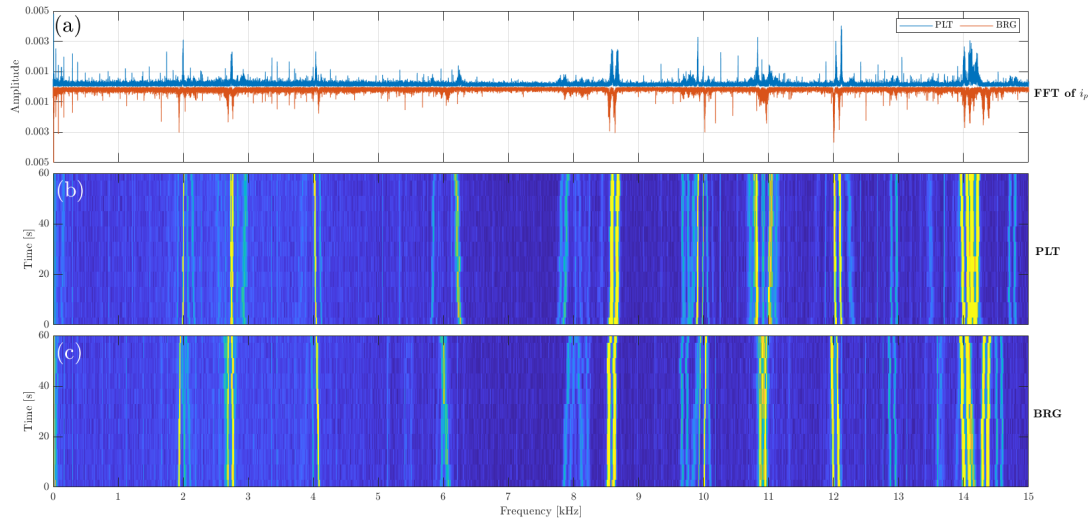


Fig. 24: Plot comparing the high-frequency peaks of the preliminary samples, where the two fault conditions are compared in (a), and where (b) and (c) are the STFT of the all the samples in the PLT and BRG runs, respectively.

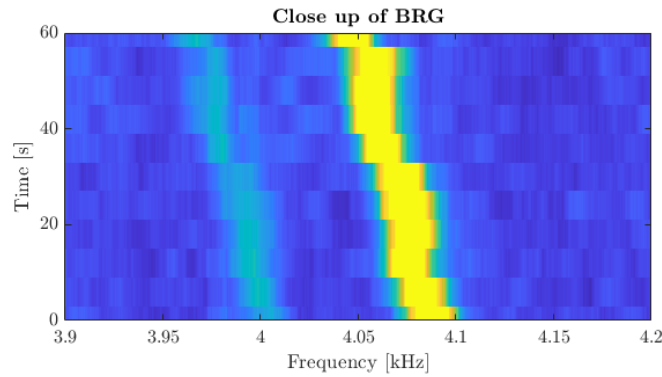


Fig. 25: Close-up of the 4 kHz region of Fig. 24.

4.1.6 Summary of the preliminary results

The preliminary analysis found substantial differences between the frequency spectrum of the healthy and faulty samples. It was also found that the harmonics of the motor drive is visible up to the 6th harmonic, and that the amplitude is higher for the PLT sample compared with the BRG sample, with the exception of the 2nd harmonic. The CFs of neither the bearing nor the gearbox are visible in the spectrum, except for the first stage planet gear (f_{p1}), which was found to be the most likely cause for the peak at 8 Hz. The results from the statistical calculations suggest that the time-domain calculations of 3rd order central moment (t_4), and the frequency-domain calculations of kurtosis (f_6) and crest factor (f_7) are good discriminatory features.

4.2 Results calculated using all samples

This section compares the findings from the preliminary analysis with the mean of all samples.

4.2.1 Feature correlation

The chart displayed in Fig. 26 shows a portion of the correlation matrix, which presents the feature's correlation coefficients with speed and load. The complete correlation matrix can be seen in Fig. B.1 in Appendix B. The red dashed line marks the threshold of 0.3, where features with a value above it will be discarded. The correlation of the load and speed are displayed on top of each other, because whether or not the correlation coefficient is above 0.3 is more important than the value of the correlation coefficient. It can be seen that 8 out of the 32 features are above the threshold and are, for this reason, not used to train or test the SVM. None of the features that were deemed as 'good' features during the preliminary analysis, were above the threshold.

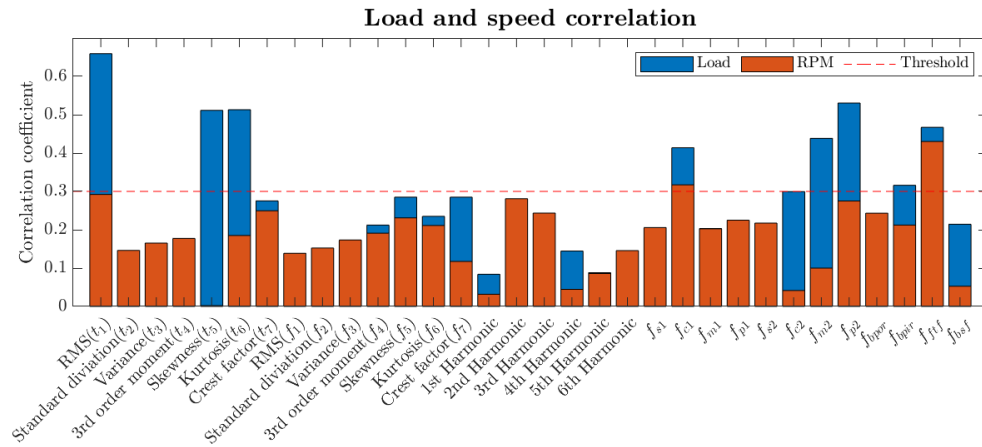


Fig. 26: Bar plot displaying all the features average correlation with speed and load.

4.2.2 Mean motor drive harmonics and characteristic frequencies amplitudes

The mean amplitudes of the CFs across all samples, shown in Fig. 27, are significantly higher for the HLT samples compared with the samples used in the preliminary analysis. This is likely due to the majority of HLT samples having a higher operating load than the sample used in the preliminary analysis. The amplitude of the CFs f_{s1} , f_{m1} , f_{p1} , f_{s2} , f_{frf} , and f_{bsf} seems relatively unchanged compared with the results from the preliminary analysis. The mean amplitude of the CF, f_{c1} , of the PLT samples is slightly higher compared with the preliminary sample. This might be due to the relatively high correlation to load, and the fact that the average load of the PLT samples is higher than the BRG samples. The same phenomenon might explain why the amplitudes of f_{m1} , f_{p2} , and f_{bpor} of the PLT samples are lower than the preliminary analysis, as the PLT data contains three runs that have an RPM of 800, and the BRG does not. It seems that f_{p1} is the only feature that consistently shows a difference between the fault conditions. However, all the CFs are still included, as there might be underlying relationships useful for classification.

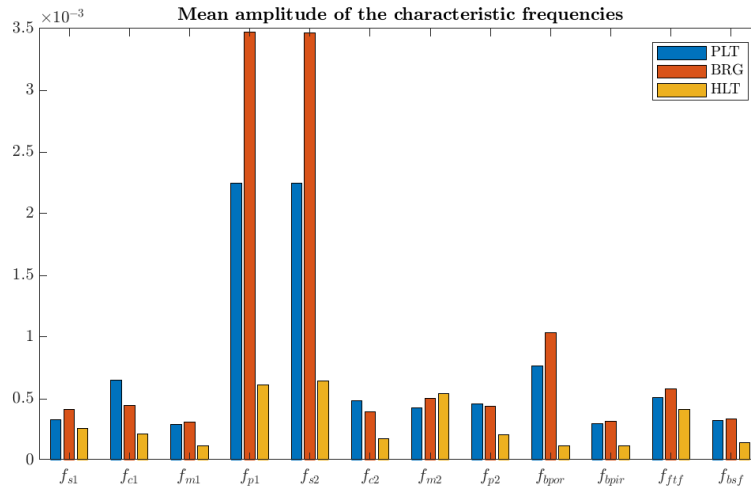


Fig. 27: Bar plot showing the average amplitudes of the CFs of all samples.

A similar rise in amplitude of the HLT samples compared with the preliminary analysis can be seen in the bar plot showing the average amplitude of the harmonics, shown in Fig. 28. It can be seen that the difference in amplitude between the two faulty health conditions is still present in the 2nd harmonic. However, the generally higher amplitude of the PLT sample does not seem to be present. This might be because the average load of the PLT samples is different from the average load of the BRG samples.

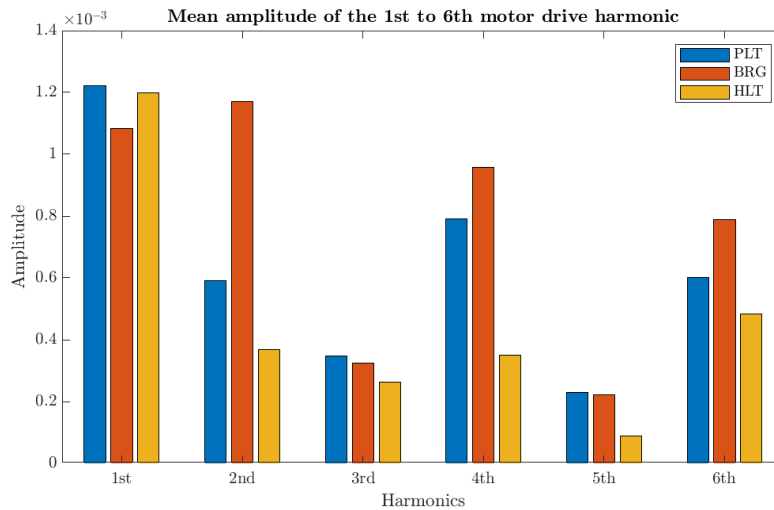


Fig. 28: Bar plot comparing the average amplitudes of the motor drive harmonics of all samples.

4.2.3 Mean statistical features of all samples

The results of the average amplitude of the statistical features, shown in Fig. 29a and Fig. 29b, agree with the findings from the preliminary analysis, as the relationship between the health

conditions is similar to the ones seen in the preliminary analysis. The major difference is that the amplitude of the statistical features for HLT are larger than the ones in the preliminary analysis, as also seen with the mean of the other features.

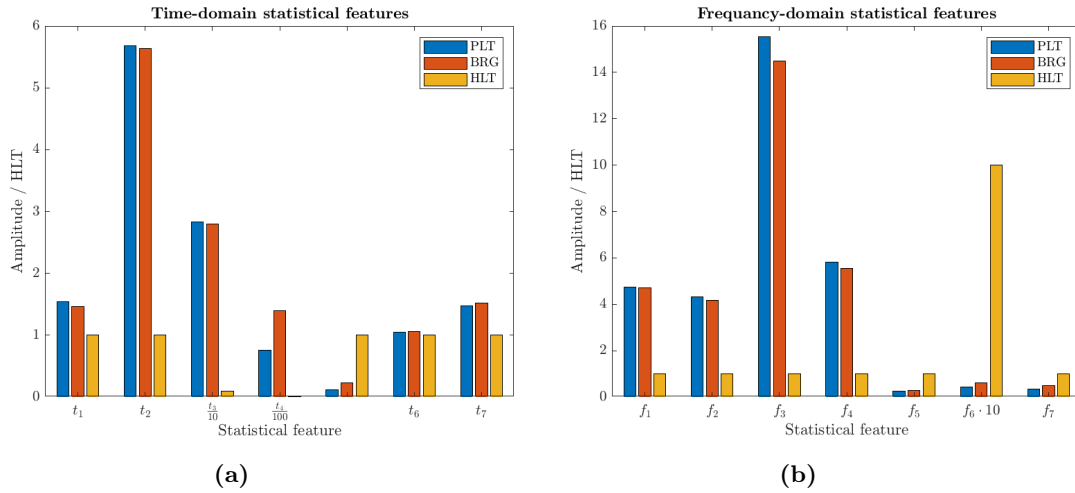


Fig. 29: Bar graph of the average values of (a) the time-domain and (b) frequency-domain statistical features of all samples.

4.3 PCA

The plot displayed in Fig. 30 shows the results from the PCA, where the blue line shows the portion of the variance of the data contained in each PC, and the orange line shows the cumulative variance. The horizontal axis corresponds to the PC number. It can be seen that almost half of the variance in the data is obtained by the first two PCs. As mentioned in Section 3.5, it was decided to discard the PCs responsible for the last 10% of the cumulative variance. This cut-off point is shown as a dashed red line on the plot. The cut-off point was at 9.7, so this was rounded up to 10 as the number of PC has to be an integer number.

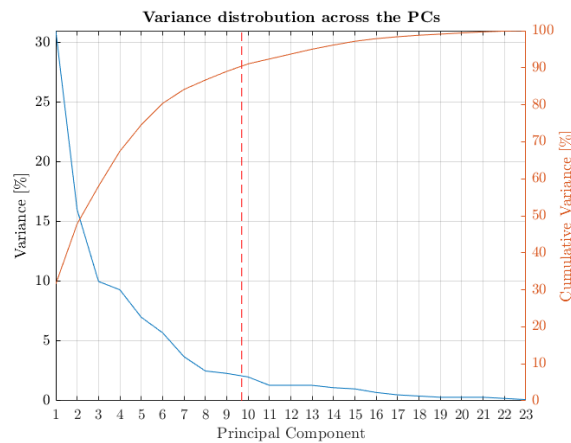


Fig. 30: Plot showing the variance of each PC, and the cumulative variance of the PCs. The red dashed line indicate where the cumulative variance reaches 90%.

4.4 SVM Results

The results from repeated hyperparameter optimizations gave fairly consistent values of the hyperparameters, finding that the best coding matrix was OVA, meaning that there would be one binary classifier for each fault condition. The box constraint and kernel scale was consistently given the value of around 121 and 0.27. These results are summarized in Table 6. The performance of the SVM classifier trained with these hyperparameters and the features from the PCA, are shown in the confusion matrices in Figs. 31a and 31b. Confusion matrices are commonly used to display the performance of a classifier. The predicted classes of the classifier are given on the horizontal axis, while the true classes are given on the vertical axis. Correctly classified classes are displayed along the diagonal, while the off-diagonal displays the misclassified samples. The percentage of correctly and misclassified samples are shown on the right side of the plot. The purpose of showing the performance of the training set is to identify whether or not the model is overfitted to the training data. If the performance of the training data differs greatly from the performance of the test data, then that is a strong indicator that the model is overfitted.

Table 6: *Showing the hyperparameters used for the SVM.*

Hyperparameter	Value
Coding matrix	OVA
Kernel scale	0.273
Box Constraint	121.321

The confusion matrix showing the performance of the training set, seen in Fig. 31a, shows an accuracy of 96%. The accuracy of the classification of HLT samples is 100%, while the classification accuracy of the BRG and PLT classification being 96.9% and 91.1%, respectively. The confusion matrix of the test set shown in Fig. 31b, shows a slightly worse accuracy of 93.7%, which is to be expected as the classifier is optimized to the testing set. The classification accuracy of the HLT samples was the same as the accuracy in the training set, supporting the claim that the classifier is good at distinguishing between healthy and faulty samples. The performance of the BRG classification is slightly worse than on the testing set, with an accuracy of 80%. The classification of the PLT samples shows better accuracy than on the training set, which was unexpected because there were samples in the PLT training data from a run that was not included in the training data. The lower accuracy of the classification of BRG compared with PLT in the testing data, might be because of the lower number of samples. However, if that was the case, it would be expected to see a similar performance difference on the training set. It should be noted that the data was performed with close to ideal conditions. There were few sources for external vibration as the test setup was ruggedly set up, and bolted to a concrete floor. In a real-world environment is there expected to be some extraneous vibrations from other components, such as the generator system. The load in the test were constant and stable. In a real-world setup, it would be expected to see load fluctuation as the blades are pitched, due to turbulence from the wind [48]. These factors will most likely contribute to lower real-world performance.

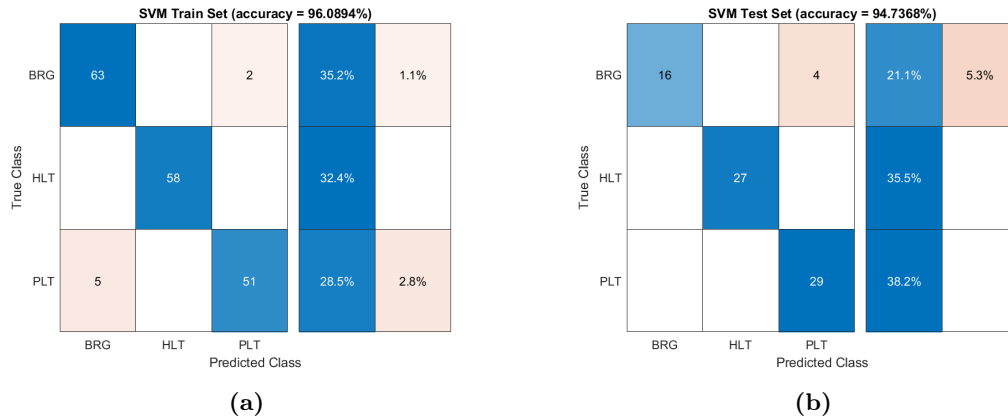


Fig. 31: Confusion matrix of (a) the training set and (b) the testing set.

The plot displayed in Fig. 32 shows the reason for the good classification accuracy of the HLT samples in both the training and testing set. The plot displays the decision boundary of an SVM classifier trained on the first two PCs from the PCA. The decision boundary is displayed in the plot as colored regions, where the color corresponding to the class of samples within that region. Although the classification accuracy of this classifier was significantly lower than the SVM trained with more than two PCs, it still had an accuracy of 100% for HLT. The reason for this is apparent when looking at the plot, as the cluster of HLT samples, shown in red, are far from away the cluster of faulty samples, shown in green and blue, representing BRG and PLT samples, respectively.

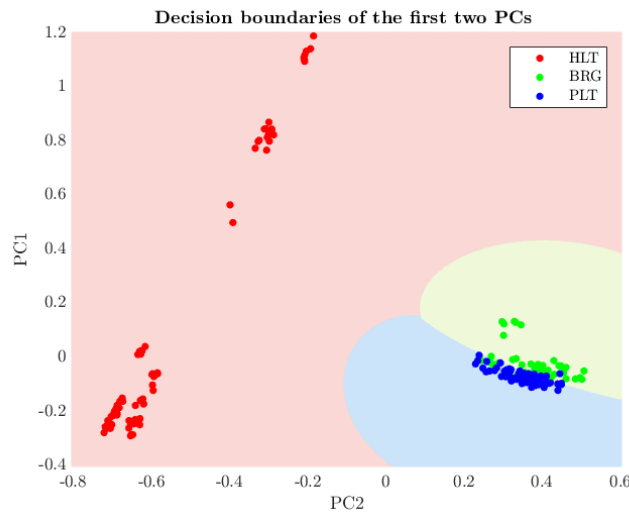


Fig. 32: Plot showing decision boundaries of an SVM trained on the first two PCs, and the clustering of the samples with the same health condition.

The plot shown in Fig. 33 displayed the ROC curves of the three binary SVM classifiers that were used to classify the health conditions. It can be seen that the ROC of the HLT classifier has an AUC of 1, indicating a perfect classifier. This is because there were no misclassified HLT samples in the test set nor the training set. The ROC of both the PLT and BRG classifiers also indicates good performance as they both have AUCs of close to 1. It can also be seen that the TPR remains 1 up until around a FPR of 0.1, and does not start to rapidly decline before approximately 0.05.

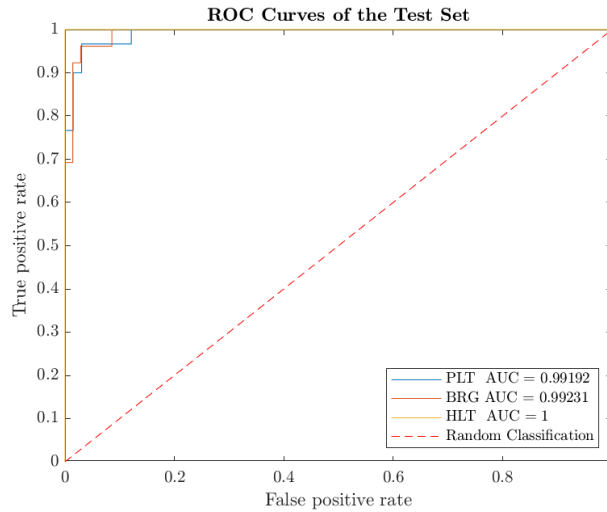


Fig. 33: ROC curve of the test set. The red diagonal line indicates the performance of random classification.

4.5 Discussion

There were challenges in pinpointing whether or not a change was due to different fault conditions, or different operation parameters, as multiple parameters were changes for each run. A preferred strategy would be to have a large data set with constant operating parameters that could be used to identify changes in the current spectrum. The SVM could then be trained on these features, and then tested on a testing set of new data with random operating parameters. This would limit the possibility of the SVM being trained on underlying features that are correlated with the speed and load of the motor. It could then be easier to draw conclusions about the SVMs actual ability to classify fault, and not the operating parameters of the motor.

5 Conclusion

The results show that the use of MCSA and an SVM classifier is a viable method for classifying faults in the input shaft bearing and the first stage planet gear of a pitch system. The classifier had an accuracy exceeding 90% on the test data, which included data from one run that was excluded from the training data. However, the classifier's real-world performance is not expected to be as good as what was observed in this thesis, as the test setup was a best-case scenario, with little external noise. Great care was put into making sure that the classifier only classified faults and not operating conditions of the test. However, it can not be completely ruled out that some of the performance of the classifier can be attributed to this, as all features had some correlation to load and speed. There were 8 out of 32 features in total that had to be discarded due to their high correlation with load and speed, as the different operating conditions of the test meant that the classifier could classify a fault condition based on load and speed alone. There were not enough samples to concretely conclude how realistic the performance of the classifier was.

The analysis found that healthy samples had a vastly different spectrum compared with the faulty samples, where the faulty spectrums had a lot higher average amplitudes. It was found that the only CF that was clearly visible in the spectrum was the first stage planet gear (f_{p1}). It was deemed as a good feature to distinguish between the two fault conditions, as the amplitude of the BRG fault was almost twice as large as the PLT fault. The higher frequency section of the spectrum was analysed using STFT, but no discriminatory features were identified, due to the low number of samples with similar operation conditions that were needed for comparison. The statistical features with consistently different values for each fault condition, and therefore, deemed good discriminative features were: time-domain 3rd order central moment (t_4), and frequency-domain kurtosis (f_6) and crest factor (f_7).

Hyperparameter optimization found that the classifier performed best using OVA as the coding matrix, and a value of 0.273 and 121.321 for kernel scale and box constraint, respectively. The SVM classifier performed well on both the training set and test set, with an accuracy of 96% for the training set, and 93.7% for the test set. The discriminatory ability of the classifier was tested using ROC, which indicated good discriminatory ability, with the AUC of the HLT classifier being 1 and the AUC of the PLT and BRG classifiers being close to 1.

The final conclusion of this thesis is that the performance of the classifier is most likely unrealistic in a real-world environment, with only the basic features used. However, the results indicate that an SVM classifier has the potential to be used for the classification of externally connected machinery, which is in agreement with the findings from the literature review. More advanced filtering and additional features might be needed to distinguish between specific faults, such as a bearing fault or a planet gear fault. However, the results indicate that the classification between a faulty and healthy system is possible with the use of basic features alone.

6 Further Work

Some topics to be further studied can be summarized accordingly.

- The proposed method in this thesis only deals with constant load and speed operation, and could not be used for samples with changing operating parameters. Therefore, future research should be conducted in more realistic settings, perhaps where the load motor is used to simulate real-life torque fluctuations caused by blade turbulence.
- Similarly to the recommendation above, future research could look at real-world data and analyse the extraneous vibration from other components. These findings could be used to get a more realistic test bench.
- An interesting topic for future work could be to look at changes between varying sizes of the pitch drive system, as this thesis only studied a scaled model of the pitch system.
- A study into the requirements for sampling rates and sensor accuracy might be beneficial for the real-world implementation of the proposed method.
- Expanding the range of possible fault conditions to accommodate other components with high rates of failure, such as the sun gear, carrier plate, etc., is also required for real-world implementations.

Several additional points of interest were originally found to be explored in this thesis. However, they were unable to be explored due to the inaccessibility of the test bench due to COVID-19. These points are summarized accordingly:

- A peak around 300 Hz was found in the frequency spectrum. A further investigation of the background noise of the test setup would have been interesting, in order to locate the source of this peak.
- Further improvements to the proposed method could be achieved with the use of more samples, as the number of samples was relatively low compared to the number of features.
- Further analysis of the higher frequency artifacts was intended to be carried out, where a single variable could be changed each time to analyse the changes in the spectrum.

References

- [1] Global Wind Energy Council. Global Wind Report 2019. <https://gwec.net/global-wind-report-2019/>, 2020. Online; accessed June 2020.
- [2] Michael Wilkinson, B Hendriks, F Spinato, K Harman, E Gomez, H Bulacio, J Roca, P Tavner, Y Feng, and H Long. Methodology and results of the reliawind reliability field study. In *European wind energy conference and exhibition 2010, ewec 2010*, volume 3, pages 1984–2004. Sheffield, 2010.
- [3] Lucio Ciabattoni, Francesco Ferracuti, Alessandro Freddi, and Andrea Monteriu. Statistical spectral analysis for fault diagnosis of rotating machines. *IEEE Transactions on Industrial Electronics*, 65(5):4301–4310, 2017.
- [4] Tianyang Wang, Qinkai Han, Fulei Chu, and Zhipeng Feng. Vibration based condition monitoring and fault diagnosis of wind turbine planetary gearbox: A review. *Mechanical Systems and Signal Processing*, 126:662–685, 2019.
- [5] T Praveenkumar, M Saimurugan, and KI Ramachandran. Comparison of vibration, sound and motor current signature analysis for detection of gear box faults. *International Journal of Prognostics and Health Management*, 8(2):1–10, 2017.
- [6] Qinkai Han, Tianyang Wang, Zhuang Ding, Xueping Xu, and Fulei Chu. Magnetic equivalent modeling of stator currents for localized fault detection of planetary gearboxes coupled to electric motors. *IEEE Transactions on Industrial Electronics*, 2020.
- [7] Hossein Davari Ardakani, Zongchang Liu, Jay Lee, Inaki Bravo-Imaz, and Aitor Arnaiz. Motor current signature analysis for gearbox fault diagnosis in transient speed regimes. In *2015 IEEE Conference on Prognostics and Health Management (PHM)*, pages 1–6. IEEE, 2015.
- [8] Feng Li, Xinyu Pang, and Zhaojian Yang. Motor current signal analysis using deep neural networks for planetary gear fault diagnosis. *Measurement*, 145:45–54, 2019.
- [9] Nassim Laouti, Nida Sheibat-Othman, and Sami Othman. Support vector machines for fault detection in wind turbines. *IFAC Proceedings Volumes*, 44(1):7067–7072, 2011.
- [10] Surya Teja Kandukuri, Jagath Sri Lal Senanyaka, Kjell G Robbersmyr, et al. A two-stage fault detection and classification scheme for electrical pitch drives in offshore wind farms using support vector machine. *IEEE Transactions on Industry Applications*, 55(5):5109–5118, 2019.
- [11] Liu Hong and Jaspreet Singh Dhupia. Vibration signal modulation of equally spaced planetary gear-set with gear tooth faults. In *Eighth International Conference on Structural Dynamics EURO-DYN*, 2011.
- [12] Chinmaya Kar and AR Mohanty. Monitoring gear vibrations through motor current signature analysis and wavelet transform. *Mechanical systems and signal processing*, 20(1):158–187, 2006.
- [13] Liu Hong and Jaspreet Singh Dhupia. A time-domain fault detection method based on an electrical machine stator current measurement for planetary gear-sets. In *2013*

- IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, pages 1631–1636. IEEE, 2013.
- [14] Jidong Zhang, Jaspreet S Dhupia, and Chandana J Gajanayake. Stator current analysis from electrical machines using resonance residual technique to detect faults in planetary gearboxes. *IEEE Transactions on Industrial Electronics*, 62(9):5709–5721, 2015.
- [15] Lin Suo, Fei Liu, Guanghua Xu, Zhenyu Wang, Wenqiang Yan, and Ailing Luo. Improved park’s vector method and its application in planetary gearbox fault diagnosis. In *2018 IEEE International Conference on Prognostics and Health Management (ICPHM)*, pages 1–7. IEEE, 2018.
- [16] Dingguo Lu, Wei Qiao, and Xiang Gong. Current-based gear fault detection for wind turbine gearboxes. *IEEE Transactions on Sustainable Energy*, 8(4):1453–1462, 2017.
- [17] Purushottam Gangsar and Rajiv Tiwari. Comparative investigation of vibration and current monitoring for prediction of mechanical and electrical faults in induction motor based on multiclass-support vector machine algorithms. *Mechanical Systems and Signal Processing*, 94:464–481, 2017.
- [18] Neelam Mehala and Ratna Dahiya. A comparative study of fft, stft and wavelet techniques for induction machine fault diagnostic analysis. In *Proceedings of the 7th WSEAS international conference on computational intelligence, man-machine systems and cybernetics, Cairo, Egypt*, volume 2931, 2008.
- [19] Jagath Sri Lal Senanayaka, Surya Teja Kandukuri, Huynh Van Khang, and Kjell G Robbersmyr. Early detection and classification of bearing faults using support vector machine algorithm. In *2017 IEEE Workshop on Electrical Machines Design, Control and Diagnosis (WEMDCD)*, pages 250–255. IEEE, 2017.
- [20] Sukhjeet Singh, Amit Kumar, and Navin Kumar. Motor current signature analysis for bearing fault detection in mechanical systems. *Procedia Materials Science*, 6:171–177, 2014.
- [21] Jiangsheng Zhu, Kuichao Ma, Amin Hajizadeh, Mohsen Soltani, and Zhe Chen. Fault detection and isolation for wind turbine electric pitch system. In *2017 IEEE 12th International Conference on Power Electronics and Drive Systems (PEDS)*, pages 618–623. IEEE, 2017.
- [22] Bin Lu, Yaoyu Li, Xin Wu, and Zhongzhou Yang. A review of recent advances in wind turbine condition monitoring and fault diagnosis. In *2009 IEEE power electronics and machines in wind applications*, pages 1–7. IEEE, 2009.
- [23] Horst Schulte, Michal Zajac, and Patrick Gerland. Takagi-sugeno sliding mode observer design for fault diagnosis in pitch control systems of wind turbines1. *IFAC Proceedings Volumes*, 45(20):546–551, 2012.
- [24] ZL Dou, MZ Cheng, ZB Ling, and X Cai. An adjustable pitch control system in a large wind turbine based on a fuzzy-pid controller. In *SPEEDAM 2010*, pages 391–395. IEEE, 2010.

- [25] Surya Teja Kandukuri. Health monitoring of pitch and yaw systems in offshore wind farms, 2018.
- [26] Marco Cocconcelli, Giuseppe Curcurú, and Riccardo Rubini. Statistical evidence of central moment as fault indicators in ball bearing diagnostics. In *The International Conference Surveillance 9*. MAR, 2017.
- [27] LMR Oliveira and AJM Cardoso. Extended park’s vector approach-based differential protection of three-phase power transformers. *IET electric power applications*, 6(8):463–472, 2012.
- [28] Surya Teja Kandukuri, Huynh Van Khang, and Kjell G Robbersmyr. Multi-component fault detection in wind turbine pitch systems using extended park’s vector and deep autoencoder feature learning. In *2018 21st International Conference on Electrical Machines and Systems (ICEMS)*, pages 1002–1007. IEEE, 2018.
- [29] John Mueller and Luca Massaron. *Machine learning for dummies*. John Wiley & Sons, Inc., 2016.
- [30] Anup Bhande. What is underfitting and overfitting in machine learning and how to deal with it. *GreyAtom, Mar*, 11, 2018.
- [31] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine learning*, 20(3):273–297, 1995.
- [32] Michael C Ferris and Todd S Munson. Interior-point methods for massive support vector machines. *SIAM Journal on Optimization*, 13(3):783–804, 2002.
- [33] Johannes Fürnkranz. Round robin classification. *J. Mach. Learn. Res.*, 2:721–747, March 2002.
- [34] S. Escalera, O. Pujol, and P. Radeva. On the decoding process in ternary error-correcting output codes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(1):120–134, 2010.
- [35] Devon K Barrow and Sven F Crone. Crogging (cross-validation aggregation) for forecasting—a novel algorithm of neural network ensembles on time series subsamples. In *The 2013 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE, 2013.
- [36] Sanjay Yadav and Sanyam Shukla. Analysis of k-fold cross-validation over hold-out validation on colossal datasets for quality classification. In *2016 IEEE 6th International conference on advanced computing (IACC)*, pages 78–83. IEEE, 2016.
- [37] James S Bergstra, Rémi Bardenet, Yoshua Bengio, and Balázs Kégl. Algorithms for hyper-parameter optimization. In *Advances in neural information processing systems*, pages 2546–2554, 2011.
- [38] Lisha Li, Kevin Jamieson, Giulia DeSalvo, Afshin Rostamizadeh, and Ameet Talwalkar. Hyperband: A novel bandit-based approach to hyperparameter optimization. *The Journal of Machine Learning Research*, 18(1):6765–6816, 2017.

- [39] Samina Khalid, Tehmina Khalil, and Shamila Nasreen. A survey of feature selection and feature extraction techniques in machine learning. In *2014 Science and Information Conference*, pages 372–378. IEEE, 2014.
- [40] Alethea Rea and William Rea. How many components should be retained from a multivariate time series pca? *arXiv preprint arXiv:1610.03588*, 2016.
- [41] Abhilash Alexander Miranda, Yann-Aël Le Borgne, and Gianluca Bontempi. New routes from minimal approximation error to principal components. *Neural Processing Letters*, 27(3):197–207, 2008.
- [42] John A Swets. *Signal detection theory and ROC analysis in psychology and diagnostics: Collected papers*. Psychology Press, 2014.
- [43] Surya Teja Kandukuri. Health monitoring of pitch and yaw systems in offshore wind farms, 2018.
- [44] Surya Teja Kandukuri, Andreas Klausen, Kjell G Robbersmyr, et al. Fault diagnostics of wind turbine electric pitch systems using sensor fusion approach. In *Journal of Physics: Conference Series*, volume 1037. IOP Publishing, 2018.
- [45] Bruce Ratner. The correlation coefficient: Its values range between $+1/-1$, or do they? *Journal of targeting, measurement and analysis for marketing*, 17(2):139–142, 2009.
- [46] Carlton Chu, Ai-Ling Hsu, Kun-Hsien Chou, Peter Bandettini, ChingPo Lin, Alzheimer’s Disease Neuroimaging Initiative, et al. Does feature selection improve classification accuracy? impact of sample size and feature selection on classification using anatomical magnetic resonance images. *Neuroimage*, 60(1):59–70, 2012.
- [47] Sukhjeet Singh and Navin Kumar. Detection of bearing faults in mechanical systems using stator current monitoring. *IEEE Transactions on Industrial Informatics*, 13(3):1341–1349, 2016.
- [48] Fan Zhang, Juchuan Dai, Deshun Liu, Linxing Li, and Xin Long. Investigation of the pitch load of large-scale wind turbines using field scada data. *energies*, 12(3):509, 2019.

Appendices

A Specifications

Table A.1: *Current sensor specifications.*

Type	Name	Specification	Value
Current sensor	LEM LTS-6NP	Supply voltage	5V
		Nominal current	6A
		Accuracy	$\pm 0.2\%$
		Range	0 ± 20

Table A.2: *Specifications of the pitch and load motor.*

Quantity	Pitch Motor	Load Motor
Model	ABB M3AR 90S4	ABB M3AA 132M8
Rated speed (rpm)	1420	715
Rated torque (Nm)	7.5	40
Rated current (A)	2.6	7.7
Voltage (V)	400, Y	400, Y
Ventilation	self-ventilated	external cooling fan

B Additional results

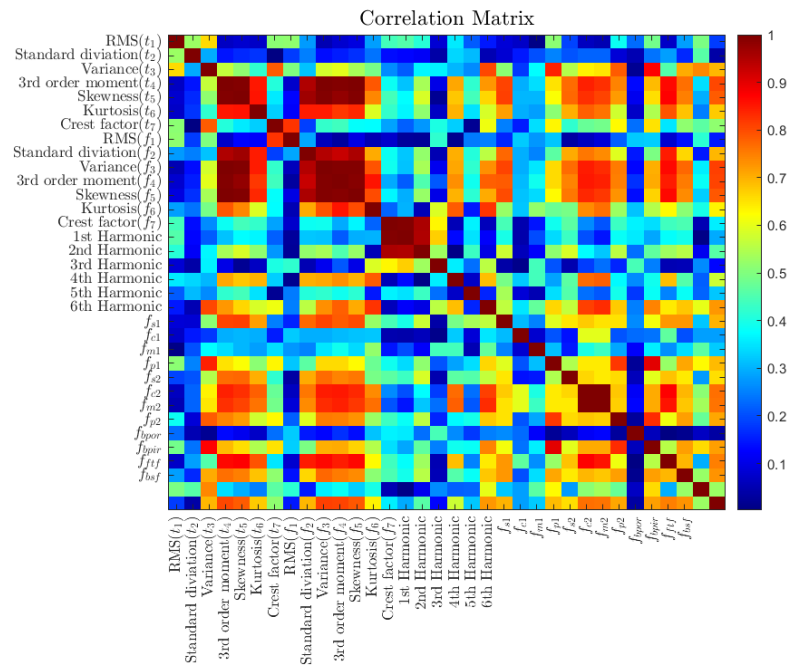


Fig. B.1: Showing the complete correlation matrix for all variables. The colorbar indicated the absolute value of the correlation coefficient.

C Overview of raw data

Table C.1: *Overview of the raw data.*

FileName	Type	Time(s)	NumDataPoints	NumVariables	RPM	Load%	NumSamples	FS sampling freq
Data-Cur-Freq-04102017_1050rpm_25L1.mat	PLT	60	1801802	4	1050	25	12	30030
Data-Cur-Freq-04102017_1200rpm_25L2.mat	PLT	60	1801802	4	1200	25	12	30030
Data-Cur-Freq-04102017_1200rpm_25L3.mat	PLT	60	1801802	4	1200	25	12	30030
Data-Cur-Freq-04102017_1300rpm_25L1.mat	PLT	60	1801802	4	1300	25	12	30030
Data-Cur-Freq-04102017_1300rpm_25L2.mat	PLT	60	1801802	4	1300	25	12	30030
Data-Cur-Freq-04102017_1400rpm_25L1.mat	PLT	60	1801802	4	1400	25	12	30030
Data-Cur-Freq-04102017_1400rpm_25L2.mat	PLT	60	1801802	4	1400	25	12	30030
Data-Cur-Freq-04102017_1400rpm_25L3.mat	PLT	60	1801802	4	1400	25	12	30030
Data-Cur-Freq-04102017_800rpm_25L1.mat	PLT	60	1801802	4	800	25	12	30030
Data-Cur-Freq-04102017_800rpm_50L1.mat	PLT	60	1801802	4	800	50	12	30030
Data-Cur-Freq-04102017_800rpm_65L1.mat	PLT	60	1801802	4	800	65	12	30030
Data-Cur-Freq-21092017_1050rpm_0L1.mat	BRG	60	1801802	4	1050	0	12	30030
Data-Cur-Freq-21092017_1050rpm_25L3.mat	BRG	60	1801802	4	1050	25	12	30030
Data-Cur-Freq-21092017_1200rpm_25L4.mat	BRG	60	1801802	4	1200	25	12	30030
Data-Cur-Freq-21092017_1200rpm_50L1.mat	BRG	60	1801802	4	1200	50	12	30030
Data-Cur-Freq-21092017_1300rpm_25L1.mat	BRG	60	1801802	4	1300	25	12	30030
Data-Cur-Freq-21092017_1400rpm_25L2.mat	BRG	60	1801802	4	1400	25	12	30030
Data-Cur-Freq-21092017_1400rpm_25L3.mat	BRG	60	1801802	4	1400	25	12	30030
Data-Cur-Freq-21092017_1400rpm_25L4.mat	BRG	60	1801802	4	1400	25	12	30030
Data-Test_1155rpm_39p2Hz_1.mat	HLT	90	2250000	4	1125	Unknown	18	25000
Data-Test_1155rpm_39p2Hz_2.mat	HLT	90	2250000	4	1125	Unknown	18	25000
Data-Test_1350rpm_45p5Hz_1.mat	HLT	90	2250000	4	1350	Unknown	18	25000
Data-Test_1350rpm_45p5Hz_2.mat	HLT	90	2250000	4	1350	Unknown	18	25000
data_21p3Hz_600rpm_83Load.mat	HLT	240	4800000	4	600	83	48	20000
data_27p9Hz_800rpm_76Load.mat	HLT	240	4800000	4	800	76	48	20000
data_33.6Hz_1000rpm_3p3Load.mat	HLT	240	4800000	4	1000	3	48	20000
data_34.5Hz_1000rpm_65Load.mat	HLT	240	4800000	4	1000	65	48	20000
data_34p3Hz_1000rpm_15Load.mat	HLT	240	4800000	4	1000	15	48	20000
data_34p5Hz_1000rpm_72Load.mat	HLT	240	4800000	4	1000	72	48	20000
data_40.2Hz_1200rpm_3p4Load.mat	HLT	240	4800000	4	1200	3	48	20000
data_41Hz_1200rpm_15Load.mat	HLT	240	4800000	4	1200	15	48	20000
data_47.7Hz_1425rpm_3p6Load.mat	HLT	240	4800000	4	1425	4	48	20000
data_48.5Hz_1425rpm_16Load.mat	HLT	240	4800000	4	1425	16	48	20000
data_48p6Hz_1425rpm_69Load.mat	HLT	240	4800000	4	1425	69	48	20000

D Calculations

Proof of RBF kernel comparing sample relationships in infinite dimensions

\vec{x}_i and \vec{x}_j have been replaced with a and b, respectively, for simplicity in this section.

Expansion of the RBF kernel:

$$K_{(\vec{a}, \vec{b})} = e^{-\gamma \|a-b\|^2} = e^{-\gamma(a^2+b^2-2ab)} = e^{-\gamma(a^2+b^2)} e^{\gamma(2ab)} \quad (\text{D.1})$$

The RBF kernel with kernel scale, γ , set to 0.5:

$$K_{(\vec{a}, \vec{b})} = e^{-\frac{1}{2}(a^2+b^2)} e^{ab} \quad , \gamma = \frac{1}{2} \quad (\text{D.2})$$

Taylor series of e^{ab} :

$$\begin{aligned} e^{ab} &= 1 + \frac{1}{1!}(ab) + \frac{1}{2!}(ab)^2 + \frac{1}{3!}(ab)^3 + \dots + \frac{1}{\infty!}(ab)^\infty \\ &= a^0b^0 + a^1b^1 + a^2b^2 + a^3b^3 + \dots + a^\infty b^\infty \end{aligned} \quad (\text{D.3})$$

Formula for polynomial kernel with r set to zero:

$$K_{(\vec{a}, \vec{b})} = (ab)^d = a^d b^d \quad , r = 0 \quad (\text{D.4})$$

Infinite sum of polynomial kernels with r set to zero:

$$\sum_{d=0}^{\infty} (a^d b^d) = a^0 b^0 + a^1 b^1 + a^2 b^2 + a^3 b^3 + \dots + a^\infty b^\infty \quad (\text{D.5})$$

Dot product of an infinite sum of polynomial kernels:

$$\sum_{d=0}^{\infty} (a^d b^d) = (1, \sqrt{\frac{1}{1!}}a, \sqrt{\frac{1}{2!}}a^2, \sqrt{\frac{1}{3!}}a^3, \dots, \sqrt{\frac{1}{\infty!}}a^\infty) \cdot (1, \sqrt{\frac{1}{1!}}b, \sqrt{\frac{1}{2!}}b^2, \sqrt{\frac{1}{3!}}b^3, \dots, \sqrt{\frac{1}{\infty!}}b^\infty) \quad (\text{D.6})$$

The dot product of the Taylor series expansion of Eq. (D.2) is given by the following equation:

$$e^{-\gamma \|a-b\|^2} = e^{-\gamma(a^2+b^2)} \left[(1, \sqrt{\frac{1}{1!}}a, \sqrt{\frac{1}{2!}}a^2, \sqrt{\frac{1}{3!}}a^3, \dots, \sqrt{\frac{1}{\infty!}}a^\infty) \cdot (1, \sqrt{\frac{1}{1!}}b, \sqrt{\frac{1}{2!}}b^2, \sqrt{\frac{1}{3!}}b^3, \dots, \sqrt{\frac{1}{\infty!}}b^\infty) \right] \quad (\text{D.7})$$

Eq. (D.7) can be simplified by introducing s , as follows:

$$e^{-\gamma \|a-b\|^2} = (s, s\sqrt{\frac{1}{1!}}a, s\sqrt{\frac{1}{2!}}a^2, s\sqrt{\frac{1}{3!}}a^3, \dots, s\sqrt{\frac{1}{\infty!}}a^\infty) \cdot (s, s\sqrt{\frac{1}{1!}}b, s\sqrt{\frac{1}{2!}}b^2, s\sqrt{\frac{1}{3!}}b^3, \dots, s\sqrt{\frac{1}{\infty!}}b^\infty) \quad , s = \sqrt{e^{-\gamma(a^2+b^2)}} \quad (\text{D.8})$$

It can be seen that the Taylor expansion of e^{ab} shown in Eq. (D.3) is the same as the infinite sum of the polynomial kernel with $r = 0$, shown in Eq. (D.5). These calculations prove that the dot product of the RBF kernel, shown in Eq. (D.8), is given by an infinite sum of polynomial kernels multiplied with s .

E Codes

Listing 1: Code used for standardizing the runs

```

1 function [Data] = myLoadData()
2
3 PLT = ls('Data\PLT');
4 BRG = ls('Data\BRG');
5 HLT = ls('Data\Healthy');
6
7 PLT = PLT(3:end,:);
8 BRG = BRG(3:end,:);
9 HLT = HLT(3:end,:);
10
11 %Loading files containg the load and speed of each run.
12 load('Data\LoadSpeedData.mat');
13
14 %Loding a file that contains the frequency of the HLT runs, as the hlt
15 % runs did not contain VFD frequency info as the PLT and BRG did
16 load('Data\HLTFreq.mat');
17
18 %% Loading Data
19 disp("Loading PLT...")
20
21 PLT_load=run_load(:,1);
22 PLT_RPM=run_speeds(:,1);
23
24 % Adding the data the raw data to a structure called Data with the
25 % structure [Data,Type,FileName,RPM,Load]
26 for i=1:size(PLT,1)
27     SampleName=strcat("RunNr",num2str(i));
28
29     Data.(SampleName)= load(strcat('Data\PLT\',PLT(i,:)));
30     Data.(SampleName).Type = "PLT";
31     Data.(SampleName).FileName = PLT(i,:);
32     Data.(SampleName).RPM =PLT_RPM(i);
33     Data.(SampleName).Load =PLT_load(i);
34
35 end
36
37 disp("Loading BRG...");
38
39 currSize=size(fieldnames(Data),1);
40 BRG_load=run_load(:,2);
41 BRG_RPM=run_speed(:,2);
42
43 for i=1:size(BRG,1)

```

```
44     SampleName=strcat("RunNr",num2str(i+currSize));
45
46     Data.(SampleName)= load(strcat('Data\BRG\ ',BRG(i,:)));
47     Data.(SampleName).Type = "BRG";
48     Data.(SampleName).FileName = BRG(i,:);
49     Data.(SampleName).RPM =BRG_RPM(i);
50     Data.(SampleName).Load =BRG_load(i);
51
52 end
53
54 disp("Loading HLT...")
55 currSize=size(fieldnames(Data),1);
56
57 HLTfreq=run_hltfreq;
58 HLTfreq=(HLTfreq+0.3287)/25.16; %converting to voltage so it is similar to
59                               % the PLT and BRG runs.
60 HLT_load=run_load(:,3);
61 HLT_RPM=run_speed(:,3);
62
63 for i=1:size(HLT,1)
64     SampleName=strcat("RunNr",num2str(i+currSize));
65
66     Data.(SampleName)= load(strcat('Data\Healthy\ ',HLT(i,:)));
67     Data.(SampleName).data(:,4)=HLTfreq(i);
68     Data.(SampleName).Type = "HLT";
69     Data.(SampleName).FileName = HLT(i,:);
70     Data.(SampleName).RPM =HLT_RPM(i);
71     Data.(SampleName).Load =HLT_load(i);
72
73 end
74
75 disp("Done.")
76
77 end
```


Listing 2: Code used for dividing the runs into samples

```

1 clear;close all;
2
3 %% Load Data
4 Data = myLoadData;
5
6 %% Devide into Samples
7 clearvars -except Data
8 timePerSample = 5; % Setting the amount of time per samples
9
10 disp("Deviding into Samples");
11
12 for c=1:size(fieldnames(Data))
13
14     % Fining the number of datapoint corresopnds to the sample time
15     totTime = (Data."RunNr"+num2str(c)).time(end);
16     numOfSamples= totTime/timePerSample;
17     dataPointsPerSample = uint32(floor(length(Data."RunNr"+num2str(c)).data)/
18         numOfSamples));
19
20     if c == 1
21         totSample = 0;
22     else
23         totSample = length(fieldnames(Samples_all));
24     end
25
26     % Finding where the data from the run should be split to make
27     % the samples.
28     for i = 1:numOfSamples
29         sampNr= i+totSample;
30         dataA = 1+dataPointsPerSample*(i-1);
31         dataB = dataPointsPerSample*i;
32
33         Samples_all.("Sample"+num2str(sampNr)).Data =...
34             Data.("RunNr"+num2str(c)).data(dataA:dataB,:);
35
36         Samples_all.("Sample"+num2str(sampNr)).Time =...
37             Data.("RunNr"+num2str(c)).time(dataA:dataB)...
38             -Data.("RunNr"+num2str(c)).time(dataA);
39
40     % Saving the samples in a new structure
41     Samples_all.("Sample"+num2str(sampNr)).Type = Data.("RunNr"+num2str(c))
42         .Type;
43     Samples_all.("Sample"+num2str(sampNr)).FileName = Data.("RunNr"+num2str
44         (c)).FileName;

```

```

42     Samples_all("Sample"+num2str(sampNr)).RPM = Data("RunNr"+num2str(c)).
        RPM;
43     Samples_all("Sample"+num2str(sampNr)).Load = Data("RunNr"+num2str(c))
        .Load;
44
45     % Overview of the running condition of the samples
46     switch Data("RunNr"+num2str(c)).Type
47         case "PLT"
48             Type_id= 1;
49         case "BRG"
50             Type_id= 2;
51         otherwise
52             Type_id= 3;
53     end
54
55     % Generating a file containg the information in each sample
56     % with the structure [Type,RPM,Load,SampleNumber]
57     Samples_info(sampNr,1)=Type_id;
58     Samples_info(sampNr,2)=Data("RunNr"+num2str(c)).RPM;
59     Samples_info(sampNr,3)=Data("RunNr"+num2str(c)).Load;
60     Samples_info(sampNr,4)=c;
61 end
62 end
63 %% Removing erroneous samples
64
65 rm_samples=[1,34,56,89,122,166]; % Sample number that are to be removed
66 rm_samples_idx=setdiff(1:length(fieldnames(Samples_all)),rm_samples);
67
68 % Creating a new structure without the erroneous samples
69 for i = 1:length(rm_samples_idx)
70     Samples("Sample"+num2str(i))=Samples_all("Sample"+num2str(rm_samples_idx(
        i)));
71     Samples_info(i)=Samples_info(rm_samples_idx(i));
72 end
73 Samples_info((length(rm_samples_idx)+1):end,:)=[];
74
75 clearvars -except Samples Samples_info
76 %% Adding Additional parameters
77 f=waitbar(0,'Processing');
78
79 %Bearing Specifications
80 bearing_N=17;
81 bearing_bd=7.938;
82 bearingfff_pd=77.495;
83
84 %Gear Specifications

```

```

85 N_p = 3;
86 Z_r = 95;
87 Z_p = 38;
88 Z_s = 16;
89
90 for i = 1:length(fieldnames(Samples))
91     waitbar(i/length(fieldnames(Samples)),f,'Processing');
92
93     % Convert from voltage to frequency
94     sample_f =Samples.("Sample"+num2str(i)).Data(:,4);
95     freq= 25.16* sample_f-0.3287; % Conversion
96     freq=movmean(freq,7); % Taking the moving averige
97
98     % Data from each sample
99     type=Samples.("Sample"+num2str(i)).Type;
100    t=Samples.("Sample"+num2str(i)).Time;
101    ia=Samples.("Sample"+num2str(i)).Data(:,1);
102    ib=Samples.("Sample"+num2str(i)).Data(:,2);
103    ic=Samples.("Sample"+num2str(i)).Data(:,3);
104    RPM =Samples.("Sample"+num2str(i)).RPM;
105
106    % Alpha Beta Transformation
107    Ialpha=2/3*(ia-0.5*ib-0.5*ic);
108    Ibeta=2/3*(sqrt(3)/2*ib-sqrt(3)/2*ic);
109
110    % dq Transformation
111    Iphase=atan(Ibeta./Ialpha);
112    Ip=sqrt(Ialpha.^2+Ibeta.^2);
113
114    % Saving Iq,Id,Ip
115    Samples.("Sample"+num2str(i)).Id=Id;
116    Samples.("Sample"+num2str(i)).Iq=Iq;
117    Samples.("Sample"+num2str(i)).Ip=Ip;
118
119    % Calculationg FFT
120    Fs=1/(t(2)-t(1)); %Finding sampeling frequency
121    [FFT_f,FFT_y]=myFFT(ip,Fs); %FFT iP
122    [FFT_f_ia,FFT_y_ia]=myFFT(ia,Fs); %FFT ia
123
124
125    % Finding the peaks in the FFT (Peaks larger than 4 x mean)
126    Cutoff=(mean(FFT_y)*4);
127    [pks1,pks2]=findpeaks(FFT_y,'MinPeakDistance',5,'MinPeakHeight',Cutoff);
128    FFT_peaks=zeros(length(FFT_y),1);
129    FFT_peaks(pks2)= 1;%pks1
130

```

```

131 Samples("Sample"+num2str(i)).FFT_peaks=FFT_peaks;
132 Samples("Sample"+num2str(i)).FFT_f=FFT_f;
133 Samples("Sample"+num2str(i)).FFT_y=FFT_y;
134 Samples("Sample"+num2str(i)).FFT_max_ia=FFT_f(find(FFT_y_ia==max(FFT_y_ia)
    ));
135
136 % Calculation Characteristic Frequencies
137 f_in=RPM/60; %Shaft speed
138
139 f_s1= (N_p*Z_r)/(Z_s+Z_r)*f_in;
140 f_c1= (Z_s/(Z_s+Z_r))*f_in;
141 f_m1= (Z_r*Z_s)/(Z_r+Z_s)*f_in;
142 f_p1= ((Z_s*Z_r)/((Z_s+Z_r)*Z_p))*f_in;
143
144 f_s2= (N_p*Z_r)/(Z_s+Z_r)*f_in*(Z_s/(Z_s+Z_r));
145 f_c2= (Z_s/(Z_s+Z_r))*f_in*(Z_s/(Z_s+Z_r));
146 f_m2= (Z_r*Z_s)/(Z_r+Z_s)*f_in*(Z_s/(Z_s+Z_r));
147 f_p2= (4*Z_s*Z_r)/(Z_r^2-Z_s^2)*f_in*(Z_s/(Z_s+Z_r));
148
149 f_bpfo= bearing_N*f_in/2*(1-bearing_bd/bearingfff_pd);
150 f_bpfi= bearing_N*f_in/2*(1+bearing_bd/bearingfff_pd);
151 f_bptf= f_in/2*(1-bearing_bd/bearingfff_pd);
152 f_bpbfbf= f_in*bearingfff_pd/(bearing_bd*2)*(1-(bearing_bd/bearingfff_pd)^2);
153
154
155 characteristic_freqs=[f_s1,f_c1,f_m1,f_p1,f_s2,f_c2,f_m2,f_p2,f_bpfo,f_bpfi
    ,f_bptf,f_bpbfbf];
156 Samples("Sample"+num2str(i)).characteristic_freqs=characteristic_freqs;
157
158 end
159 close(f);
160 clearvars -except Samples Samples_info

```

Listing 3: Code used for calculating features

```

1 % Preallocating for the features
2 feature= zeros(length(fieldnames(Samples)),33);
3 Load=zeros(length(fieldnames(Samples)),1);
4 RPM=zeros(length(fieldnames(Samples)),1);
5 feature_label= strings(length(fieldnames(Samples)),1);
6 p_range=0.75;
7
8 for i=1:length(fieldnames(Samples))
9
10     %Loading data from the sample structure
11     FFT=Samples.("Sample"+num2str(i)).FFT_y;
12     FFT_f=Samples.("Sample"+num2str(i)).FFT_f;
13     feature_label(i)= Samples.("Sample"+num2str(i)).Type;
14
15     %Statistical Features Time
16     input_data= Samples.("Sample"+num2str(i)).Ip;
17
18     feature(i,1)=rms(input_data);
19     feature(i,2)=var(input_data);
20     feature(i,3)=std(input_data);
21     feature(i,4)=kurtosis(input_data);
22     feature(i,5)=peak2rms(input_data);
23     feature(i,6)=skewness(input_data);
24     feature(i,7)=moment(input_data,3);
25
26     %Statistical Features Frequency
27     input_data= Samples.("Sample"+num2str(i)).FFT_y;
28
29     feature(i,8)=rms(input_data);
30     feature(i,9)=var(input_data);
31     feature(i,10)=std(input_data);
32     feature(i,11)=kurtosis(input_data);
33     feature(i,12)=peak2rms(input_data);
34     feature(i,13)=skewness(input_data);
35     feature(i,14)=moment(input_data,3);
36
37     % Motor Dirve Harmonics
38     f_s= Samples.("Sample"+num2str(i)).FFT_max_ia;
39     harmonics= (1:6)*f_s;
40
41     feature(i,15)= bandpower(FFT,FFT_f,[harmonics(1)-p_range harmonics(1)+
42         p_range], 'psd');
43     feature(i,16)= bandpower(FFT,FFT_f,[harmonics(2)-p_range harmonics(2)+
44         p_range], 'psd');

```

```

43     feature(i,17)= bandpower(FFT,FFT_f,[harmonics(3)-p_range harmonics(3)+
44         p_range], 'psd');
45     feature(i,18)= bandpower(FFT,FFT_f,[harmonics(4)-p_range harmonics(4)+
46         p_range], 'psd');
47     feature(i,19)= bandpower(FFT,FFT_f,[harmonics(5)-p_range harmonics(5)+
48         p_range], 'psd');
49     feature(i,20)= bandpower(FFT,FFT_f,[harmonics(6)-p_range harmonics(6)+
50         p_range], 'psd');
51
52     %Characteristic Frequency Features
53     cf= Samples.("Sample"+num2str(i)).characteristic_freqs;
54     feature(i,21)= bandpower(FFT,FFT_f,[abs(cf(1)-p_range) cf(1)+p_range], 'psd'
55         );
56     feature(i,22)= bandpower(FFT,FFT_f,[abs(cf(2)-p_range) cf(2)+p_range], 'psd'
57         );
58     feature(i,23)= bandpower(FFT,FFT_f,[abs(cf(3)-p_range) cf(3)+p_range], 'psd'
59         );
60     feature(i,24)= bandpower(FFT,FFT_f,[abs(cf(4)-p_range) cf(4)+p_range], 'psd'
61         );
62     feature(i,25)= bandpower(FFT,FFT_f,[abs(cf(5)-p_range) cf(5)+p_range], 'psd'
63         );
64     feature(i,26)= bandpower(FFT,FFT_f,[abs(cf(6)-p_range) cf(6)+p_range], 'psd'
65         );
66     feature(i,27)= bandpower(FFT,FFT_f,[abs(cf(7)-p_range) cf(7)+p_range], 'psd'
67         );
68     feature(i,28)= bandpower(FFT,FFT_f,[abs(cf(8)-p_range) cf(8)+p_range], 'psd'
69         );
70     feature(i,29)= bandpower(FFT,FFT_f,[abs(cf(9)-p_range) cf(9)+p_range], 'psd'
71         );
72     feature(i,30)= bandpower(FFT,FFT_f,[abs(cf(10)-p_range) cf(10)+p_range], '
73         psd');
74     feature(i,31)= bandpower(FFT,FFT_f,[abs(cf(11)-p_range) cf(11)+p_range], '
75         psd');
76     feature(i,32)= bandpower(FFT,FFT_f,[abs(cf(12)-p_range) cf(12)+p_range], '
77         psd');
78
79     RPM(i)=Samples.("Sample"+num2str(i)).RPM;
80     Load(i)=Samples.("Sample"+num2str(i)).Load;
81 end
82 corrmat=[feature RPM Load]; %Adding load and speed to the corrMat
83
84 CM= abs(corr(corrmat)); %Finding the absolute linear correlation
85 CM=CM([end-1 end],:); % Removing the self correlation of load and speed
86
87 % Removing the features that have a correlation coefficient greater than
88 % 0.3

```

```
73 rm_features=unique([find(CM(1,:)>0.3) find(CM(2,:)>0.3)]);
74 feature_idx=setdiff(1:size(feature,2),rm_features);
75 feature=feature(:,feature_idx);
76
77 %% Names of the included features
78 featureNames=["RMS\_t", "Var\_t", "STD\_t", "Kurt\_t", "Crest Factor\_t"...
79             , "Skewness\_t", "3rd Order Moment\_t", ...
80             "RMS\_f", "Var\_f", "STD\_f", "Kurt\_f", "Crest Factor\_f"...
81             , "Skewness\_f", "3rd Order Moment\_f", ...
82             "1st Harmonic", "2nd Harmonic", "3rd Harmonic", ...
83             "4th Harmonic", "5th Harmonic", "6th Harmonic", ...
84             "f\_s1", "f\_c1", "f\_m1", "f\_p1", "f\_s2", "f\_c2", ...
85             "f\_m2", "f\_p2", "f\_bpfo", "f\_bpfi", "f\_bptf", "f\_bpbfb"];
86 featureNames=featureNames(feature_idx);
87
88 clearvars -except featureNames feature Samples Samples_info feature_label
```

Listing 4: Code used for training and testing the SVM

```

1 %% Load Features
2 FeatureExtraction;
3
4 %% Creating test and train sets
5 X=feature;
6 Y=feature_label;
7
8 [coeff,score,~,~,explained,mu] = pca(X);
9
10 i=0; % Finding how many PCs are required to get <90% variance
11 while totVar < 0.9
12     i=i+1;
13     totVar=sum(explained(1:i));
14 end
15
16 %Making making the PCs the features
17 X=score(:,1:i);
18
19 %% Finding index od the run which is held out
20 % Find idx of PLT samples
21 holdout_idx = find(Samples_info(:,1) == 1);
22 % Find idx of PLT samples with speed of 1200
23 holdout_idx = find(Samples_info(holdout_idx,2) == 1200);
24 % Find index of PLT run with speed = 1200 and load = 25
25 holdout_idx = find(Samples_info(holdout_idx,3) == 25);
26
27
28 % Devide into eavanly sized collections of samples of the same type
29 plt_idx = [find(Y == "PLT",1,'first') find(Y == "PLT",1,'last')];
30 %Removing the holdout samples from the plt idx
31 plt_idx_mod = setdiff(plt_idx,holdout_idx);
32
33
34 brg_idx = [find(Y == "BRG",1,'first') find(Y == "BRG",1,'last')];
35 hlt_idx = [find(Y == "HLT",1,'first') find(Y == "HLT",1,'last')];
36
37 % Randomizing the data
38 plt_idx = randperm(plt_idx(2)-plt_idx(1))+plt_idx(1)-1;
39 brg_idx = randperm(brg_idx(2)-brg_idx(1))+brg_idx(1)-1;
40 hlt_idx = randperm(hlt_idx(2)-hlt_idx(1))+hlt_idx(1)-1;
41
42 % Resizing the sample sets
43 X = X([plt_idx(1:115),brg_idx(1:85),hlt_idx(1:120)],:);
44 Y = Y([plt_idx(1:115),brg_idx(1:85),hlt_idx(1:120)]);

```



```

45
46 rand_num=randperm(size(X,1));
47
48 %Deviding into trainig and testing data 70/30 split
49 train_idx = rand_num(1:round(0.7*length(X)));
50 test_idx=rand_num((round(0.7*length(X))+1:length(X)));
51
52 % Finding the plt samples of the test set
53 test_plt_idx=find(Y(test_idx)=="PLT");
54 %Number of sample to be replaced with the PLT in the holdout set
55 numOfHoldout=round(0.5*length(test_plt_idx));
56
57 %Replacing the indexes with half of the holdout indexes
58 test_plt_idx_holdout(1:numOfHoldout) =holdout_idx(randperm(numOfHoldout));
59
60 % Replacing the PLT indeces in of the test indexes
61 test_idx(test_plt_idx)=test_plt_idx_holdout;
62
63 % Relabeling the test and training sets.
64 X_train=X(train_idx,:);
65 Y_train=Y(train_idx);
66
67 X_test=X(test_idx,:);
68 Y_test=Y(test_idx);
69
70 %% Train SVM
71 % Deviding the training data into testing and validation sets using k-fold
72 % cross-validation with 5-folds
73 c = cvpartition(X_train,'k',5);
74
75 %Options for the traioning of the classifier
76 options = statset('UseParallel',true);
77 t= templateSVM('KernelFunction','gaussian');
78
79 %Training the classifier
80 Mdl= fitcecoc(X_train,Y_train,'Learners',t,'CVPartition',c,'ClassNames',...
81 {'PLT','BRG','HLT'},'Options',options,'OptimizeHyperparameters','auto',...
82 'HyperparameterOptimizationOptions',struct('AcquisitionFunctionName',...
83 'expected-improvement-plus'));
84
85 close all;
86 %% Training set confution matrix
87 f1=figure;
88 %Predicting the lable of the training data
89 pred_lable= predict(Mdl,X_train);
90

```

```

91 test_acc= sum(((grp2idx(pred_lable)-grp2idx(cellstr(Y_train))))...
92     ~=0)/length(Ytrain);
93
94 %Plotting confusion matrix
95 ConfMat = confusionchart(cellstr(Y_train),pred_lable,'RowSummary'...
96     ,'total-normalized');
97
98 title("SVM Train Set (accuracy = "+num2str(100*(1-test_acc))+"\%");
99
100 %% Test set confution matrix
101 f2=figure;
102
103 %Predicting the lable of the test data
104 pred_lable= predict(Mdl,X_test);
105
106 %Finding the total accuracy
107 test_acc= sum(((grp2idx(pred_lable)-grp2idx(cellstr(Y_test))))...
108     ~=0)/length(Y_test);
109
110 %Plotting confusion matrix
111 ConfMat = confusionchart(cellstr(Y_test),pred_lable,'RowSummary',...
112     'total-normalized');
113 title("SVM Test Set (accuracy = "+num2str(100*(1-test_acc))+"\%");
114
115 %% ROC Curve
116 f3=figure;
117 [~,~,score_svm] = predict(Mdl,X(:,fs));
118
119 Y_log=grp2idx(cellstr(Y)); % True lables
120
121 plt_y_grp=Y_log(find(Y=="PLT",1,'first'));
122 brg_y_grp=Y_log(find(Y=="BRG",1,'first'));
123 hlt_y_grp=Y_log(find(Y=="HLT",1,'first'));
124
125
126 [Xsvm1,Ysvm1,~,ACU1] = perfcurve(Y_log,score_svm(:,1),plt_y_grp);
127 [Xsvm2,Ysvm2,~,ACU2] = perfcurve(Y_log,score_svm(:,2),brg_y_grp);
128 [Xsvm3,Ysvm3,~,ACU3] = perfcurve(Y_log,score_svm(:,3),hlt_y_grp);
129
130 plot(Xsvm1,Ysvm1)
131 hold on
132 plot(Xsvm2,Ysvm2)
133 plot(Xsvm3,Ysvm3)
134
135 plot([0 1],[0 1],'r','LineStyle','—')
136

```

```
137 legend("PLT \ AUC = "+num2str(ACU1),"BRG  AUC = "+num2str(ACU2)...
138       ,"HLT \ AUC = "+num2str(ACU3),"Random Classification",...
139       'Location','southeast')
140
141 xlabel('False positive rate'); ylabel('True positive rate');
142 title('\textbf{ROC Curves of the Test Set}')
143
144 %% Save figures
145 saveas(f1, 'Plots/ConfMat_TrainSet.png');
146 saveas(f2, 'Plots/ConfMat_TestSet.png');
147 saveas(f3, 'Plots/ROC_Curve.png');
```