Using Principal Component Analysis to determine changes in Mechanical Properties

Author Akewak Jeba

Supervisor Jerker Björkqvist

May 2021

ABSTRACT

The main goal of this study is to determine changes in mechanical properties using principal component analysis. This project included the main concept of principal component analysis and its implementation in determining changes in mechanical properties over time.

The study is made using wind turbine as a user case in order to determine the changes in mechanical properties. Two Different data cases were used to address the different changes in mechanical properties failure. In this paper we used a fully functional and a broken 3MW wind turbines vibration measurements.

MATLAB software was used to carry out the project. Mathematical algorithms were programmed and tested for the result to find the necessary output. In this project statistical analysis was the basic core.

In order to find the main indicators of the changes of the specific mechanical wear, principal component analysis was used. Principal component analysis reduces the dimensionality of the datasets and retains the variation presented on the original dataset. In this paper statistical features from time domain and spectral kurtosis from frequency domain were extracted. Then Principal component analysis was used for dimension reduction and feature fusion.

This study also used the principal components as health-indicators in order to determine the remaining useful life of the wind turbine. This project also suggested more research have to be done in many other fault detections in other rotating machineries to show the importance of principal component analysis in those implementations

Key words: PCA, Principal Component Analysis, Wind turbine, Mechanical wear, Fault detection, High -speed prognosis, RUL, Remaining useful life.

Table of Contents

LIST OF ABBREVIATIONS AND SYMBOLS

PCA	Principal Component Analysis
FFT	Fast Fourier Transform
RMS	Root Mean Square
CF	Crest Factor
STFT	Shot Time Fourier Transform
HI	Health Indicator
EoL	End of Life
RUL	Remaining Useful Life
AI	Artificial Intelligence
SK	Spectral Kurtosis
CRR	Cumulative Relative Accuracy
PH	Prognostic Horizon

1. INTRODUCTION

1.1. Background

Early humans used different technologies to gather foods and to survive predators. In order to survive, humans created different tools that could be used to hunt foods and protect themselves. These tools have evolved through time. Since the early days of humans, the tools made have either worn out or become useless. This property of materials has been the major problem

.

Today, engineers who build materials or components are trying to minimise the change in mechanical properties of materials. These changes can be detected by different methods. One of them is collecting data from the machineries and analysing the data and identifying the cause of the failures.

Fault detection on rotating machineries includes diagnosis of items such as shafts, gears and pumps. There are different types of faults and the diagnoses are made in different ways. Some of the diagnosis methods include vibration analysis, model-based techniques, and statistical analysis [1].

The goal of this thesis is to use principal component analysis to determine changes in mechanical properties. The wind turbine is used as a user case in the study to determine the changes in mechanical properties. To address the various changes in mechanical properties failure, two different data cases were used. PCA is used as a dimension reduction and feature fusion tool in this work to detect faults and determine the RUL of the bearings.

For this thesis, high-speed wind turbine pinion data was used. The measurement was taken on a 3MW wind turbine with radial vibration for one week and a fault was found on the pinion gear. Similarly, other measurements were taken from two other machines and with no known faults [2].

1.2. Objective

The main goals of this thesis are as follows:

1. To discuss different fault types in rotating machineries.
2. To understand wind turbine systems and the different fault cases.
3. To use a prognostic approach to identify faults in wind turbines.
4. To show the implementation of principal component analysis to determine faults in rotating machineries by taking wind turbine high-speed bearing data for our case study.

5. To determine the remaining useful life of the wind turbine after a fault is detected.

1.3. Organisation of the thesis

The second chapter defines machine faults and introduces the various algorithms used to detect them. The third chapter describes the wind turbine system, the general fault modes in wind turbines, and health monitoring methods. The fourth chapter discusses the fault detection implementation and the results obtained. Chapter 5 contains the conclusion and additional work that has been proposed.

2. VIBRATION ANALYSIS ALGORITHMS AND ROTATING MACHINERY

2.1. Background

Rotating machineries, such as turbines, are key components to power plants and manufacturing industries. Since they are usually subjected to harsh environments, they are easily prone to defects or faults. Faults in these machineries could cause damages to the power-generating plants or other manufacturing industries. These damages could be economic, or even worse, life threatening. For example, a system failure in transportation machines such as cars, trains or airplanes could be deadly. Therefore, it is essential to identify the causes of faults and defects.

The causes of faults in rotating machineries could arise from different components of the machinery, such as the gear, the shaft or the pinions. Once the faults in these components are identified, it could be easy to identify symptoms related to the faults analysis and detect the faults before they occur.

Machine breakdowns generally are undesired for many reasons. As mentioned above, it causes additional cost for the machine owner by requiring materials and manpower. Efficiency loss, equipment downtime and customer service or comfort consume resources and even damages reputations. Faults could occur suddenly, gradually or due to incorrect set system. Preventing faults, detecting faults beforehand and designing a fault tolerant system are among the approaches taken to prevent the problem.

2.1.1. Fault prevention

Faults can be prevented through periodical maintenance during a statistically determined period of time. This periodical maintenance helps to minimise faults between those periods, ensures the correctness of the machine's settings and minimises unexpected economic and life costs. These periodic maintenances require resources and downtime incurring costs. However, making the proper maintenance in accordance with manufacturers recommendations will actually increase energy savings and boost productivity.

2.1.2. Fault detection

In order to detect faults different measurement methods are used to monitor the conditions of machineries. Acoustic, vibration and temperature measurement are among the different techniques used. The signals collected from the machineries are pre-processed and analysed for the presence of faults. Acoustic and vibration measurement methods are the most common in detecting faults in rotating machineries. Time and frequency domain vibration measurements, sound pressure and intensity, shock pulse, and acoustic emission methods are some of the methods used to measure the response from a defective machinery.

Time and frequency domain methods are the most widely used methods for fault detection. Research shows that time domain methods were being used before the birth of frequency domain methods. Some of the well-known examples of time domain methods are root mean square (RMS), crest factor, peak value and kurtosis.

2.1.3. Faults in rotating machines

Faults in rotating machineries can be described as any change in the machine components that causes the machine to work inefficiently or puts it to a complete stop. These faults can be caused in one or more components of the machine.[4] Gomaa and Khader, in their article on fault diagnosis of rotating machinery, explain the causes for the failure in a rotating machine. They mention weakness in material design, misuse and mechanical wear of components as the causes.

Faults in rotating machines are due to damaged gears, damaged bearings, bent shaft, and vibration of the machine itself. These faults should be detected as soon as they occur in order to minimise any damages and losses. For this purpose, different mechanisms are put forward by different researchers and scientists and are still being suggested.

Different condition monitoring and fault detection systems are used for monitoring and fault detection in rotating machineries. These methods use measurements such as temperature, pressure, noise, vibration and oil analysis in order to monitor the condition of the machines. However, the most effective systems, both for condition monitoring and fault detection, use vibration measurements using transducers such as accelerometers, velocity pickups and displacement probes.[5]

2.2. Signal processing techniques for fault detection

This section discusses different signal processing techniques that are applied to rotating machineries. These different methods include the most widely used feature extractions from time domain, frequency domain and time-frequency domain and find patterns that are related to faults. Similarly, other methods that are more advanced signal processing techniques such as wavelet transform (WT), classification methods, envelop analysis and cyclostationary analysis are available.

2.2.1. Time domain methods

Time domain feature extraction for fault detection is a traditional and most commonly used technique in rotating machineries. It is a simple method which involves the extraction of statistical features in order to analyse the changes in trends which is related to faults.

Root mean square (RMS) is the most common statistical feature extracted in vibrational signals. RMS can be mathematically defined as:

$$RMS = \sqrt{\frac{1}{T_2 - T_1}\int_{T_1}^{T_2}[x(t)]^2\,dt} \qquad (2.1)$$

where $x(t)$ is a signal between the instants T_1 and T_2 [17].

Energy is another statistical feature extracted from the time-frequency domain. It is the measure of true energy in the area defined by the signal curve. It is defined mathematically as

$$E = \int_{-\infty}^{\infty} |X(t)|^2 dt \qquad (2.2)$$

where $x(t)$ is the signal [17].

Peak-to-peak is another feature extracted from the time domain. It is the distance from the top of one peak to the bottom of another or it can be generally explained as the maximum voltage value in the signal [17].

Crest factor (CF) is also a feature that is derived from peak-to-peak divided with the RMS. This feature is not effective when the degradation is significant in the rolling machinery. It is mathematically expressed as follows [17]:

$$CF = \frac{|x|_{peak}}{x_{RMS}} \qquad (2.3)$$

2.2.2. Frequency domain methods

Fast fourier transform is a frequency domain spectral analysis technique that is used for detection of faults in vibrational signals. It is an extension of the fourier transform signal analysis method where we transform a time-based signal *f(t)* to a frequency-based spectrum *F(w)* including all frequencies.

$$F(w) = \int_{-\infty}^{\infty} f(t)e^{-j2\pi utdt} \tag{2.4}$$

FFT might be the most popular numerical algorithm in science and engineering, however, it has its own limitations since it is only applicable to stationary signals. Due to these limitations, many more non-stationary signals analysis methods were introduced. Short-time fourier transform (STFT) was later developed to determine the frequency content of a local section in terms of change in time [14].

Spectral kurtosis is a frequency domain feature extraction technique. It is the most powerful and most commonly used method for detecting degradation in rotating machineries. It is defined as the fourth-order spectral moment. It can be defined mathematically as:

$$SK(f) = \frac{\langle |X^4(t,f)| \rangle}{\langle X^4(t,f) \rangle^2} - 2 \tag{2.5}$$

For a signal of x(t), where $X^4(t,f)$ is the fourth order and $X^2(t,f)$ is the second order of the band-pass filtered signal x(t) [14].

2.2.3. Wavelet transform

Wavelet transform is a new and powerful technique which is applicable to nonstationary signals. Wavelet analysis can be done by multiplying a wavelet function to a signal that is going to be analysed. Then, the transform is computed for each segment created. The advantage of wavelet transform is that it helps us to use short time interval where there is a high-frequency information or a long-time interval with precise low-frequency information. This behaviour helps us to perform local analysis without losing spectral information. The disadvantage of this technique is that the frequency resolution might be poor for high-frequency signals. This simultaneous time-frequency representation of a signal helps us to identify faults in machineries.

2.2.4. Cyclostationary analysis

Rotating machinery vibration signals feature modulations; thus, the cyclic correlation and cyclic spectrum are well suited to analyse their modulation characteristics to detect faults in their components. The cyclic correlation and cyclic spectrum for Amplitude modulation and frequency modulation (AM-FM) signals are derived and summarised [18].

The cyclic statistics is used to for signals with periodicity and multi-periodicity with respect to time. Cyclic correlation and cyclic spectrum are effective methods to extract features of cyclostationary signals. The cyclic function can be defined as:

$$R_x^{\alpha}(\tau) = \int_{-\infty}^{\infty} x\left(t + \frac{\tau}{2}\right)^{*} x\left(t - \frac{\tau}{2}\right) exp(-j2\pi\alpha t)\, dt \qquad (2.6)$$

where $x(t)$ represents the signal, τ is time lag and α is the cyclic frequency [18].

2.2.5. Envelope analysis

Envelope analysis is a technique of extracting the modulating signal from amplitude of the signal and detects periodic variations in the amplitude of the signal. It is used for detection of faults in machineries, such as gearbox, turbines and induction motors where faults have an amplitude modulation effect on the characteristic frequencies. It is also a tool for diagnosing local faults like cracks and spalling in rolling element bearings. This technique is the Fast Fourier transform (FFT) frequency spectrum of the modulating signal. This technique uses the following steps:

i. Band pass filter
ii. Signal rectification
iii. Hilbert transform
iv. Spectral analysis

In general, there are many other fault detection methodologies that are suggested by different scientists and researchers. The above-mentioned methods are the most important and the most applicable ones. In this paper, only time domain and frequency domain spectral kurtosis are implemented.

2.3. Principal component analysis (PCA)

Principal component analysis (PCA) is a dimension reduction technique without losing much of the information and the variation within the original dataset. This technique is used for different purposes in high dimensional data, such as data compression, feature extraction, signal analysis, and fault diagnosis.

Data with *n* number of samples and k -attributes can be represented as a matrix X,

$$\mathrm{X} = \begin{bmatrix} x_{11} & \cdots & x_{1k} \\ \vdots & \ddots & \vdots \\ x_{n1} & \cdots & x_{nn} \end{bmatrix} \qquad (2.7)$$

Once we represent the data as a matrix shown above, we then subtract the mean from each dimension, and this would produce a dataset with a zero mean. This subtraction of the mean is called mean-centring. The subtraction of the mean helps us to remove bias.

$$\mathrm{M} = \begin{bmatrix} x_{11} - \overline{X_1} & \cdots & x_{1k} - \overline{X_k} \\ \vdots & \ddots & \vdots \\ x_{n1} - \overline{X_k} & \cdots & x_{nn} - \overline{X_k} \end{bmatrix} \qquad (2.8)$$

Then we can calculate the covariance matrix as,

$$Covariance\ matrix = \frac{M * M^T}{n} \qquad (2.9)$$

thus resulting,

$$S = \begin{bmatrix} C_{11} & \cdots & C_{1k} \\ \vdots & \ddots & \vdots \\ C_{k1} & \cdots & C_{kk} \end{bmatrix} \qquad (2.10)$$

where C_{IJ} expressed as,

$$C_{IJ} = 1/n\,\{(x_i - \bar{X}_I)(x_j - \bar{X}_j)\} \quad (i,j = 1,2,\ldots k) \qquad (2.11)$$

Different scale measures of variance and covariance will result in different results. Since these results cannot be compared, we have to normalise the data by dividing each matrix by its standard deviation. Hence this results in a normalized matrix element C_{ij},

$$C_{ij} = \frac{c_{ij}}{\sqrt{var(i)\,.var(j)}} \quad (\mathrm{I,j} = 1,2,\ldots\ldots,\mathrm{k}) \qquad (2.12)$$

where $var(i)$ and $var(j)$ are the variance of ith and jth elements respectively. Both variances are the maximum variations a variable can have, and the correlation can never exceed $\sqrt{var(i)\,.\,var(j)}$. This will result in the maximum value of covariance matrix (equal to one). For variables which are uncorrelated the covariance is zero i.e., $C_{ij} = C_{ji} = 0$

As mentioned above, the main goal of PCA is to reduce dimensions of a data set while keeping the variation of the original dataset. Covariance matrix defines both the spread (variance) and orientation (covariance) of a dataset.

3. SYSTEM OVERVIEW AND FAULTS OF WIND TURBINES

3.1. System overview

A wind turbine is an electromechanical system that converts wind energy to electrical energy. It is made of different subsystems and components such as blades, rotor, gearbox, generator, yaw, tower, controller, anemometer and break.

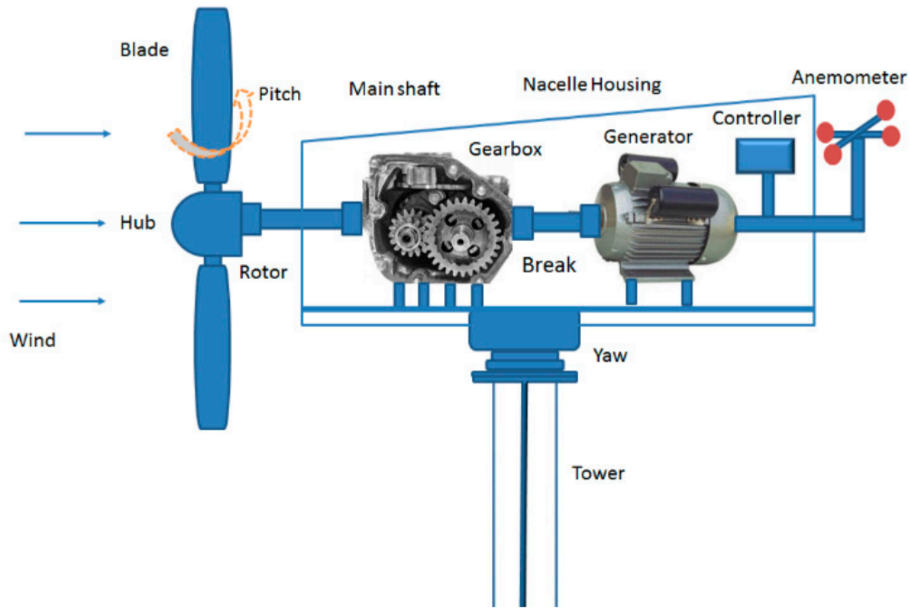

Figure 3.1 Typical wind turbine system [9]

As we can see from the figure above, a typical wind turbine is made of different parts. Its functionality can also be explained as follows. As the wind blows, it causes the blade and the rotor to run and causes the main shaft to rotate and speed up the gearbox to drive the generator and change wind energy to mechanical and finally to electrical energy. The anemometer is used to identify the wind direction to help the yaw system to align the turbine to the direction of the wind. The controller is used to make sure the wind turbine generates the needed electricity [9].

Wind turbine components are vulnerable to faults due to short-lived malfunctions or aging, causing system interruption. There is a difference between faults and failure. A fault is when a system is in a mode where it keeps its functionality, but it is in an unacceptable stage. However, failures in a system or its component mean a complete inability to function. The figure below shows the different causes of failures and their relative percentage of occurrences in wind turbines [9].

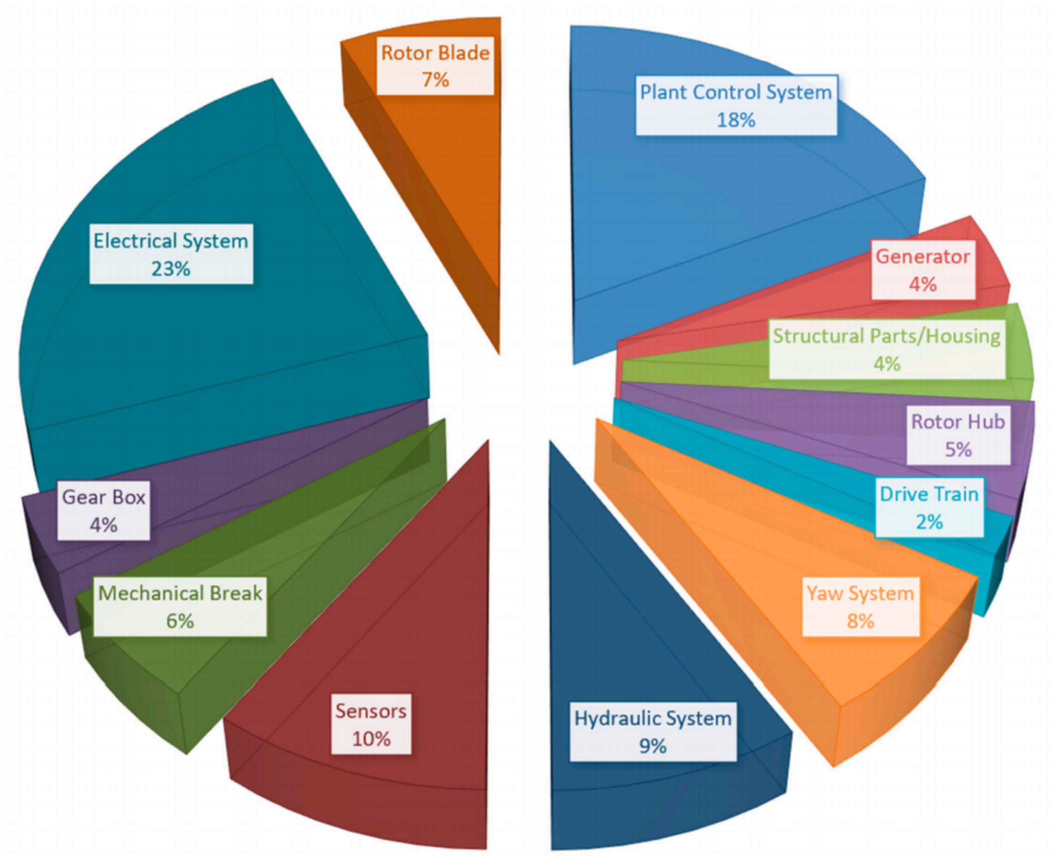

Figure 3.2 The total share of faults shown in percentage [9]

Table 3.1 Wind turbine fault types [11,12,13]

Types of Faults	Causes of Faults
Faults on blades and rotors	Corrosion of blades and hub; crack; reduced stiffness; increased surface roughness; deformation of the blades; errors of pitch angle; and imbalance of rotors, etc.
Faults on gearbox	Imbalance and misalignment of shaft; damage of shaft, bearing and gear; broken shaft; high oil temperature; leaking oil; and poor lubrication, etc.
Faults on generator	Excessive vibrations of generator; overheating of generator and bearing; abnormal noises; and insulation damage, etc.
Faults on bearing	Overheating; and premature wear caused by unpredictable stress, etc.
Faults on main shaft	Misalignment; crack; corrosion; and coupling failure, etc.
Hydraulic faults	Sliding valve blockage; oil leakage, etc.
Faults on mechanical braking system	Hydraulic failures; and wind speed exceeding the limit, etc.
Faults on tower	Poor quality control during the manufacturing process; improper installation and loading; harsh environment, etc.
Faults on electrical systems/devices	Broken buried metal lines; corrosion or crack of traces; board delamination; component misalignment; electrical leaks; and cold-solder joints, etc.
Faults on sensors	Malfunction or physical failure of a sensor; malfunction of hardware or the communication link; and error of data processing or communication software, etc.

The percentages of typical faults in wind turbines are shown in the above diagram (Figure 3.2). As one can see from the above pie chart, the faults in the wind turbines are mainly caused by mechanical and electrical components [10].

Table 3.1 lists the main causes of typical fault types in wind turbines and causes for the faults. As can be seen, all the faults mentioned are either mechanical or electrical faults. The faults occur at different parts of the wind turbine, such as the blade, gearbox, bearing, generator, tower and sensors.

3.2. Fault modes of wind turbines

Both onshore and offshore wind turbines have become one of the fastest growing Both onshore and offshore wind turbines have become one of the fastest growing sources of electricity production. There is a growing need and use of wind turbines all over the world, especially in Europe due to its clean energy production. In Europe, it has been recorded that it is the second source of energy next to gas and oil. With the increasing use of wind turbines, an evolution in quantity and size of the wind turbines has taken place. This has caused an increase in cost of maintenance due to failures in wind turbines [5].

There are two types of maintenance: corrective and preventive. These are referred to as traditional maintenance strategies by Abid et al., and they have a significant disadvantage in that the fault is experienced by the system. The other strategies are called prognostics and health management or predictive maintenance strategies that provide an advanced maintenance strategy that can increase reliability and availability while reducing unexpected fault and maintenance cost [5].

The performance of the wind turbine or the condition of components, especially critical components such as rotating machinery (e.g., generator), are used to track the health of wind turbines. Fault prognostic or predictive maintenance is a useful strategy, since we are predicting when a failure will take place, or it estimates the remaining useful life (RUL) of the wind turbine. RUL prediction estimates the time remaining between the degradation detection and failure threshold. Most of the time, the prediction of critical components is very difficult due to noise, system complexity and prediction uncertainty caused by operating conditions of the wind turbines, such as wind speed and direction [5].

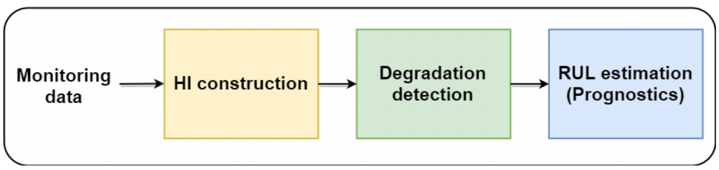

Figure 3.3 General approach of prognostic [5]

Figure 3.3 depicts a general approach to prognosis. The figure suggests the steps for the approach, i.e., health indicator (HI) construction, degradation indicator and RUL estimation (prognostics). The health indicator is constructed by the pre-processing of the data and monitoring the evolution of the system performance. The degradation indicator is then triggered as soon as the health indicator HI falls below a predefined threshold. Finally, we predict the remaining useful life by forecasting the degradation evolution and estimating the time when the system will fall below the threshold [5].

3.3. Health indicator

A change in system performance is evaluated using a health indicator. The main and the most important step toward achieving prognostic is the health indicator. If the system's performance deviates from its normal condition, we can say it is heading towards failure. These indicators are drawn by studying the system pattern from the collected data from the sensors. There are two types of health indicators: indicators based on a single feature and indicators based on multiple features. Those with single features, such as using sensor data, residuals-based features, and time domain or time-frequency features are extracted from data collected by monitoring sensors[5].

The other health indicator is based on multiple features. This health indicator method helps to identify complex degradations. These complex degradations are difficult to identify and construct a health indicator that follows the degradation over time to allow a reliable remaining useful life. Therefore, we have to fuse different features in order to identify and draw the health indicator. This fusion has its own drawbacks, such as information loss, which can lead to a lack of interpretability of health indicators to represent the virtual description of the system's health performance [5].

Abid et al. in their literature proposed several methods for health indicator construction on single and multiple features, as shown in the table below. Various methodologies are proposed in the table based on the number of features. These features, which are extracted from the vibration signals of damaged bearings, frequently exhibit varying modulation characteristics. As a result, the features extracted from one bearing may not correlate with the features extracted from another bearing, which occurs due to different trends from different cases. As a result, we must apply and validate various bearing fault indicators, such as Kurtosis and root mean square, for various applications [14].

Table 3.2: Health indicator construction methods [14]

Feature type	Computation approaches	Methods	Data
Single feature based HI	Raw signal and residuals	Raw signal of viscositv and dielectric constant	Lubrication oil (gearbox)
		Raw signal of oil debris	Lubrication oil (gearbox)
		Power residual	Generated power signal
		Temperature residual	Bearings temperature (Gearbox)
	Time domain features	Root mean square	Vibration (bearings)
		Spectral kurtosis	Vibration (bearings)
		Trigonometric functions	Vibration (bearings)
	Time-frequency features	Wavelet Packet Decomposition	Vibration (bearings)
		Hilbert huang transform	Vibration (bearings)
		Power spectral density	Current signal (gearbox)
Multiple features based HI	Dimension reduction	ISOMAP	Vibration (bearings)
		PCA	SCADA data
	Distance between classes	Euclidian and mahalanobis distance	Pitch angle (pitch system), Voltage (Converter)
		Jensen-Renyi Distance	Vibration (bearings)

Different methodologies are suggested based on the number of features, as seen in the table above. These features, which are extracted from the vibration signals of damaged bearings, frequently exhibit different modulation characteristics. As a result, the features extracted from one bearing may not correlate with the features extracted from another bearing, which occurs due to different trends from different cases. As a result, we must apply and validate various bearing fault indicators, such as Kurtosis and root mean square, for various applications [14].

3.3.1. Feature extraction

The vibration signal is employed in the computation of features in the time, frequency, and time-frequency domains. Different statistical features are generated from vibration signals to indicate faults in bearings, shafts, and gears, which are components of a gearbox. Bearings frequently operate in harsh environments and are subject to various types of faults. The advantages of the time-frequency domain have been demonstrated in numerous research papers and journals [15].

3.3.2. Time domain feature extraction

As previously stated, one of the most important methods for identifying faults in bearings is time domain analysis. It is a simple and powerful tool for identifying faults in mechanical vibrations by computing statistical features. These statistical features are implemented in this paper and are shown in the table below.

Table 3.3 Time domain features [15]

Feature	Expression
RMS	$\left(\frac{1}{N}\sum_{i=1}^{N} x_i^2\right)^{\frac{1}{2}}$
Kurtosis	$\frac{1}{N}\sum_{i=1}^{N} \frac{(x_i-\bar{x})^4}{\rho^4}$
Skewness	$\frac{1}{N}\sum_{i=1}^{N} \frac{(x_i-\bar{x})^3}{\rho^3}$
Peak to peak (P-P)	$x_{\max}-x_{\min}$
Crest factor	$\frac{x_{\max}}{RMS}$
Shape factor	$\frac{RMS}{\frac{1}{N}\sum_{i=1}^{N} \lvert x_i \rvert}$
Impulse factor	$\frac{x_{\max}}{\frac{1}{N}\sum_{i=1}^{N} \lvert x_i \rvert}$
Margin factor	$\frac{x_{\max}}{\left(\frac{1}{N}\sum_{i=1}^{N} \lvert x_i \rvert\right)^2}$
Mean	$\bar{x} = \frac{1}{N}\sum_{i=1}^{N} x_i$
Standard deviation (Std)	$\sigma = \left(\frac{1}{N}\sum_{i=1}^{N} (x_i - mean)\right)^{\frac{1}{2}}$
Energy	$\sum_{i=1}^{N} x_i^2$
Energy entropy	$-\sum_{i=1}^{N} x_i \log(x_i)$

In the table above, x represents the sampled time signal, sample index, and N represents the number of samples. As shown above, several statistical fault-indicating features are used to diagnose faults using vibration signals. The most commonly used fault indicators are RMS and Kurtosis. In this paper, we implement all of the features for prognostic analysis.

3.4. Frequency domain feature extraction

In this paper, we used frequency domain feature extraction, which is one of the most widely used methods in bearing fault diagnosis. Since it can be obscured by other causes of vibration such as gears, shafts, and other mechanical faults, spectral kurtosis (SK) is one of the most effective tools for detecting faults.

Ben Ali et al. define the SK signal as the normalised fourth-order spectral moment and can be mathematically written as:

$$SK(f) = \frac{\langle |X^4(t,f)| \rangle}{\langle X^4(t,f) \rangle^2} - 2 \qquad (3.1)$$

For x(t) signal, where $X^4(t,f)$ is the fourth order and $X^2(t,f)$ is the second order of the band-pass filtered signal x(t).

3.5. Health indicator evaluation criteria

Lei et al. proposed various criteria for assessing the suitability of the constructed health indicators for RUL estimation. Many researchers have proposed different metrics for evaluation of prognostic health indicators. Lei et.al. have summarised those in a table shown below.

Table 3.4. Summary of metrics of evaluating prognostics health indicators.

Categories	Names
Metrics depending on a single HI	• Monotonicity • Robustness
Metrics depending on a HI and time	• Trendability
Metrics depending on a HI and stage sequence	• Identifiability
Metrics depending on multiple HIs	• Consistency
Hybrid metrics	• Hybrid metrics

The most important health indicator criteria, according to Abid et al., are monotonicity, prognosability, and trendability.

3.5.1. Monotonicity

Monotonicity is the property that evaluates the positive or negative trend of a health indicator based on the assumption that the faulty component is unable to recover on its own. Based on the derivatives of the health indicator, two metrics of monotonicity were proposed. As shown below, the absolute difference between negative and positive derivatives is used to calculate the first monotonicity.

$$Mon_1(X) = \frac{1}{K-1}|No.of\ \ d/dx > 0 - No.of\ \ d/dx < 0| \quad (3.2)$$

where $X = \{x_k\}_{k=1:K}$ is the health indicator sequence, x_k is the health indicator value at time t_k, d/dx represents the derivative of the health indicator, K represents the total number of health indicator values; $d/dx = x_{k+1} - x_k$ denotes the difference of the health indicator sequence. $No.of\ \ d/dx > 0$ 0 and $No.of\ \ d/dx < 0$ represent the number of positive and negative differences respectively. The monotonicity score ranges from 0 to 1, with a higher score indicating better monotonicity performance.

The other metric is presented as

$$\begin{cases} Mon_{2+}(X) = \frac{No.of\ d/dx>0}{K-1} + \frac{No.of\ d^2/d^2x>0}{K-2} \\ Mon_{2+}(X) = \frac{No.of\ d/dx<0}{K-1} + \frac{No.of\ d^2/d^2x}{K-2} \end{cases} \quad (3.3)$$

where d^2/d^2x denotes the second order derivatives of X; $Mon_{2+}(X)$ represents positive monotonicity and $Mon_{2-}(X)$ represents negative monotonicity [6].

A function f can be monotonically increasing or decreasing depending on its values. It may be trivial to check the monotonicality of a single failure progression sample by examining the difference between consecutive points. The function is said to be non-decreasing, if all of the difference values are greater than or equal to 0. However, rather than individual sample analysis, monotonicality over all samples representing failure progression should be considered [7].

3.5.2. Prognosability

Prognosability is a measure of a feature's variability at failure in relation to the range between its initial and final values. The prognosability of each path is determined by dividing the standard deviation of the final failure values by the mean range of the path. This is exponentially weighted to achieve the best zero to one scale.

$$Prognosability = exp\left(-\frac{std(failurevalues)}{(|failurevalue-startingvalue|)}\right) \quad (3.4)$$

This metric takes into account failure values that are well clustered, i.e., failure values with a low standard deviation and wide parameter ranges [17].

3.5.3. Trendability

Trendability is a measure of the similarity of a feature's trajectories measured in multiple run-to-failure experiments. It is the measurement with the smallest absolute correlation. As compared to the other two metrics, determining the trendability of a population of parameters is even more complicated. If the same underlying functional type can model each parameter in the population, a candidate parameter is trendable. Trendability can be expressed as follows:

$$Trendability = min\left(\left|corrcoef_{ij}\right|\right) \quad (3.5)$$

The trendability is determined by the smallest absolute correlation measured across the entire population of prognostic parameters [17].

3.6. Degradation detection

As previously stated, the health indicator provides information about the system's performance or health. When there is a degradation, the health indicator changes its trend, causing the prognostic module to predict the remaining useful life. There are various methods for detecting degradation. One method is to divide the health indicator into two or more stages by using a threshold.

Table 3.5 Methods for detecting degradation and their drawbacks

Approaches	Methods	Advantage	Drawback
Threshold	Threshold value	No need for degradation data. Easy for implementation.	Difficulty to choose the threshold value in order to achieve early degradation detection and avoid false alarms. Need of an expert for the choice of the threshold value.
	Statistical threshold 3σ interval		
Classification	Logistic regression	The boundary between nominal and degraded conditions is estimated automatically.	Necessity of system degradation data. Model parameters are difficult to tune.
	SVM		

The division of the health indicator allows us to improve the reliability of the degradation and the estimation of the RUL.

3.7. Remaining useful life (RUL) estimation

When a degradation is detected using the health indicator with a change from normal performance, the remaining useful life estimation begins. In general, the RUL is the amount of time between the time we discovered the fault and the end of the useful life. According to Lei et al., this time is expressed as $l_k = t_{EOL} - t_k$, where t_{EOL} is time at the end of life, t_k is the current time, and l_k is the remaining useful time at t_k [7].

Forecasting a wind turbine's remaining useful life provides us with information on how much time remains before the wind turbine loses its operational capacity. The two main challenges in predicting the remaining useful time of wind turbines are how to predict based on the given data and the accuracy of the prediction [7].

3.8. Degradation modelling

As previously stated, estimation or prediction is accomplished through the use of various approaches. According to Abid et al., these approaches are broadly classified into two types: experience-based approaches and degradation modelling approaches. The experience-based approaches rely on maintenance and inspection data, whereas the degradation approach relies on physical or data-driven approaches. This model-based approach models degradation using physical and mathematical relationships. The wind turbine condition monitoring sensors provide us with accurate information about our system and assist us in developing data-driven approaches to predicting system degradation. Statistical and artificial intelligence tools are primarily used in these data-driven approaches [6].

Table 3.6 Different RUL prediction approaches [7]

Categories of RUL prediction approaches
Statistical approaches, model-based approaches and AI approaches
Traditional reliability approaches, physics-based models, data-driven models and integrated approaches
Knowledge-based models, aggregate reliability functions, physical models, statistical models, conditional probability models and ANN
Model based approaches, data-driven approaches and hybrid prognostic approaches
Statistical data driven approaches
Model-based methods, data-driven models and combination models
Heuristic models, statistical models, physics-based models, data-driven models and hybrid models

The table above shows that different researchers used different approaches to predict RUL. Lei et al., however, further subdivided the model-based approaches into four categories: physics model-based approaches, statistical model-based approaches, Artificial Intelligent (AI) approaches, and hybrid approaches.

3.9. Prognostic metrics

When predicting the remaining useful life, we need some metrics to compare the various approaches mentioned above. These metrics are divided into three categories because the remaining useful life prediction can be evaluated from a variety of perspectives. The first metrics categories are based on RULs from the field. The second metric is based on run-to-failure data, while the last is based on available measurements. In the table below, Lei et al. summarised some of the metrics.

Table 3.7 A summary of the RUL prediction metrics [7]

Categories	Names
Metrics depending on ground truth RULs	• RMSE • CI • Prediction horizon • α-λ accuracy • Relative accuracy • CRA • Convergence • ETA
Metrics depending on run-to-failure data	• Predictability • Mean prediction error, Std • Overall average bias • Overall average variability • Reproducibility
Metrics depending on available measurements	• Online RMSE • Online coverage • Online width

The root mean square error (RMSE) and mean absolute percentage error (MAPE) are the most commonly used metrics for RUL evaluation (MAPE). However, other new and relevant metrics for evaluating prognostics performance were proposed. Prognostic horizon, $\alpha - \lambda$ performance, relative accuracy, and cumulative relative accuracy are a few of them.

3.9.1. Root mean square error (RMSE)

The root mean square of the prediction in the interval between the first prediction period and the end of lifetime is called the root mean square error (RMSE).

$$RMSE = \sqrt{\frac{1}{n}\sum_{t=1}^{n}(r^{l}(t) - r_{*}^{l}(t))^2} \qquad (3.6)$$

where n is the number of observations in the interval from the first prediction time and end of lifetime, t is the time index, $r^{l}(t)$ is the predicted RUL, and $r_{*}^{l}(t)$ is the ground true RUL. A higher RMSE value indicates a larger prediction error [6].

3.9.2. Mean absolute percentage error (MAPE)

Similarly, the mean absolute percentage error (MAPE) is a popular metric for evaluating RUL as RMSE.

$$MAPE = \frac{100\%}{n} \sum_{t=1}^{n} \left| \frac{r^l(t) - r_*^l(t)}{r^l(t)} \right| \qquad (3.7)$$

where n denotes the number of observations between the first prediction time and end of lifetime, t denotes the time index, $r^l(t)$ denotes the predicted RUL, and $r_*^l(t)$ denotes the ground true RUL [6].

3.9.3. Prognostic horizon

The prognostic horizon (PH) or prediction horizon is the time between the detection of degradation and the end of life (EoL). It can be expressed as

$$PH = t_{EoL} - t_{i_{\alpha\beta}} \qquad (3.8)$$

where $i_{\alpha\beta}$ =is the first-time index when prediction β criteria for a given α and t_{Eol} is the predicted end of lifetime EoL [6].

3.9.4. $\alpha - \lambda$ performance

$\alpha - \lambda$ is a binary metric used to evaluate RUL prediction where the result falls within a specified limit at a given time.

$$\alpha - \lambda \text{ performance} = \begin{cases} 1 & if \;\; \pi(p(l_{t_\lambda}))|_{-\alpha}^{+\alpha} \geq \beta \\ 0 & otherwise \end{cases} \qquad (3.9)$$

where λ is a time window modifier; $\pi(p(l_{t_\lambda}))|_{-\alpha}^{+\alpha} \geq \beta$ is the probability mass of the prediction PDF in the α-bound [6].

3.9.5. Relative accuracy

Relative prediction accuracy is the relative prediction error between the predicted RUL to the actual RUL at a specific time.

$$RA_{\lambda} = 1 - \frac{\left|l^{*}_{t_{\lambda}} - l_{t_{\lambda}}\right|}{l^{*}_{t_{\lambda}}} \tag{3.10}$$

where $l^{*}_{t_{\lambda}}$ is the RUL's truth value and $l_{t_{\lambda}}$ is the predicted RUL at time λ [6].

3.9.6. Cumulative relative accuracy

Cumulative relative accuracy CRA evaluates the RUL at multiple time instances by aggregating the relative accuracy values at a specific time instance. It is computed as a weighted sum of the relative accuracies at multiple time instances.

$$CRA_{\lambda} = \frac{1}{|\Omega_{\lambda}|} \sum_{k e \Omega_{\lambda}} w(l_{t_k}) RA_{\lambda} \tag{3.11}$$

where $w(l_{t_k})$ is a weight function depending on the predicted RUL, Ω_{λ} is the set of all time indexes and $|\Omega_{\lambda}|$ is the cardinality of the set [6].

According to Lei et al., two other metrics that rely on the ground truth URLs are convergence and exponential transformed accuracy. Among the metrics that are affected by run-to-failure results are predictability, mean prediction error and standard deviation, overall average bias, overall average variability, and reproducibility. Metrics based on available measurements, such as online root mean square error, online coverage, online width, and Epilog, are also discussed.

4. WIND TURBINE HIGH SPEED BEARING PROGNOSIS

In this thesis paper, we use PCA to detect faults in a wind turbine and build an exponential degradation model to predict the remaining useful life (RUL) of the wind turbine. The chosen model helps us to detect the significant degradation of the wind turbine. Generally, the general steps of prognosis include: Data import and exploration, feature extraction, feature post processing, training data, feature importance ranking, dimension reduction and feature fusion, building model, model fitting and analysing performance.

4.1. Dataset

The dataset in this study is taken from a high-speed gear dataset by Eric Bechhoefer and is licensed under a creative Commons attribution-non-commercial share Alike 4.0 international License. The dataset has two cases: case 1 with fault and case 2/3 with good condition. The dataset is collected from a 3MW wind turbine high-speed shaft driven by a 32-tooth pinion gear with a sample rate of 97656 Hz. A vibration signal of 6 seconds was acquired in both cases.

4.2. Data Import

The data used for this project are in the form of MAT files which are a binary binary data container format that a MATLAB program uses. Therefore, in this implementation we used MATLAB and implemented MATLAB codes.

teeth	ppr	gs	sr	tach
32	8	{585936×1 double}	97656	{9406×1 double}
32	8	{585936×1 double}	97656	{9459×1 double}
32	8	{585936×1 double}	97656	{9463×1 double}
32	8	{585936×1 double}	97656	{9477×1 double}
32	8	{585936×1 double}	97656	{9517×1 double}
32	8	{585936×1 double}	97656	{9542×1 double}
32	8	{585936×1 double}	97656	{9278×1 double}
32	8	{585936×1 double}	97656	{9464×1 double}
32	8	{585936×1 double}	97656	{9491×1 double}
32	8	{585936×1 double}	97656	{9410×1 double}
32	8	{585936×1 double}	97656	{9325×1 double}
32	8	{585936×1 double}	97656	{9420×1 double}
32	8	{585936×1 double}	97656	{9555×1 double}
32	8	{585936×1 double}	97656	{9570×1 double}
32	8	{585936×1 double}	97656	{9442×1 double}
32	8	{585936×1 double}	97656	{9498×1 double}
32	8	{585936×1 double}	97656	{9499×1 double}
32	8	{585936×1 double}	97656	{9527×1 double}
32	8	{585936×1 double}	97656	{9465×1 double}
32	8	{585936×1 double}	97656	{9583×1 double}
32	8	{585936×1 double}	97656	{9507×1 double}
32	8	{585936×1 double}	97656	{9429×1 double}
32	8	{585936×1 double}	97656	{9574×1 double}
32	8	{585936×1 double}	97656	{9407×1 double}

Figure 4.1 Imported data

In the above figure, the shown dataset contains a constant vibration signal (gs), sampling rate(sr), constant angular sampling(pp), number of teeth on pinion(teeth) and tachometer signal (tach).

4.3. Data Exploration

Data exploration is one of the most important steps for data understanding, preliminary data quality evaluation and learning from the domain. The visualisations usually reveal patterns or exceptions that would suggest that there is something interesting in the dataset. In our case we explore the data in both time domain and frequency domain in order to figure out what features we need to extract for prognosis purposes.

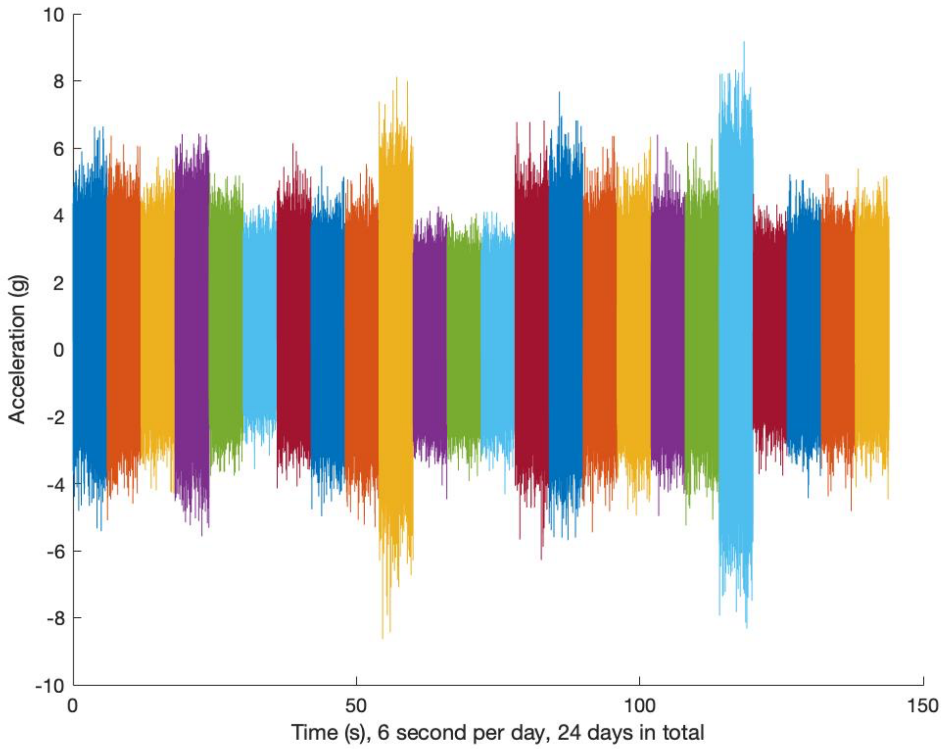

Figure 4.2 Vibration signal in time domain

The above figure shows the vibration signal in time domain. The dataset contains 24 vibration signals of 6 seconds measured on 24 different days. It shows some exceptions and trends of signal impulsiveness. From the figure above, we can suggest that indicators of statistical features such as kurtosis, crest factors, peak-to-peak value are potential indicators of impulsiveness of signals in the wind turbine bearing dataset.

Similarly, kurtosis is the most powerful tool in frequency domain wind turbine prognosis. The figure below depicts the evolution of spectral kurtosis over time.

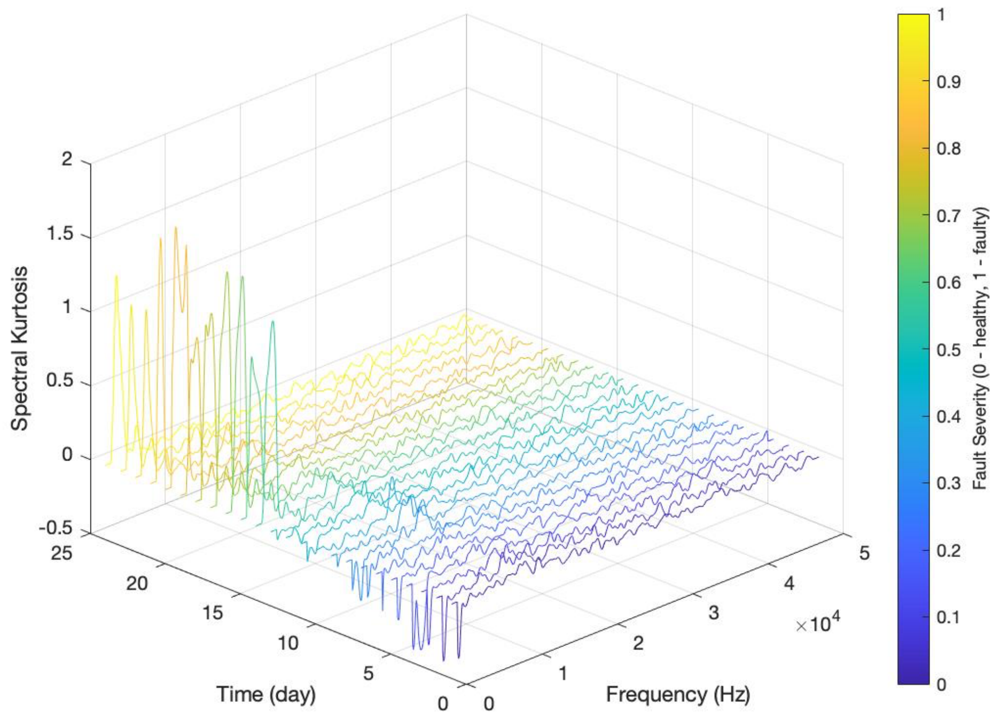

Figure 4.3 Visualisation of change in spectral kurtosis

We can see from the above frequency domain visualisation that we used a colour-bar to denote the magnitude of the fault, which is normalised into a 0 to 1 scale. The kurtosis value around 5kHz gradually increases as the machine condition deteriorates, as shown in the figure. This means that spectral kurtosis features such as mean standard deviation and others could be potential indicators of bearing degradation.

4.4. Feature extraction

We used frequency domain and time domain visualisations to analyse the vibration signals in the previous section. We have concluded that we will derive statistical features from time domain and extract spectral kurtosis from the frequency domain. The extracted statistical features are depicted in the diagram below.

feature_table

24x15 table

	1 mean	2 std	3 skewness	4 kurtosis	5 peak2peak	6 RMS	7 CrestFactor	8 ShapeFactor	9 ImpulseFactor	10 MarginFactor	11 Energy	12 SKMean	13 SKStd	14 SKSkewness	15 SKKurtosis
1	0.7638	1.3859	-0.0033	3.0284	12.9006	1.5824	4.3320	0.2308	2.0718	0.5834	1.4672e...	0.0503	0.2845	8.5011	116.0061
2	0.7626	1.3610	-0.0022	3.0048	12.4125	1.5601	4.3355	0.2307	2.0458	0.5815	1.4261e...	0.0343	0.2665	9.3768	144.7037
3	0.7702	1.2248	-0.0067	3.0064	11.8977	1.4468	4.4549	0.2245	1.8786	0.5932	1.2266e...	0.0011	0.1314	0.3161	5.7827
4	0.7221	1.5311	-0.0042	2.9569	13.8826	1.6928	4.5365	0.2204	2.3442	0.5215	1.6791e...	0.0232	0.2191	9.4375	172.2626
5	0.7594	1.1685	-0.0156	3.0219	11.8348	1.3936	4.8140	0.2077	1.8351	0.5768	1.1380e...	0.0063	0.1314	0.4127	5.1372
6	0.7909	0.9552	-0.0019	3.0143	8.7687	1.2401	4.0371	0.2477	1.5679	0.6256	9.0113e...	0.0094	0.1344	0.7197	4.9068
7	0.7442	1.1849	-0.0101	3.0022	10.9277	1.3992	4.3880	0.2279	1.8802	0.5538	1.1471e...	0.0095	0.1398	1.3105	10.5741
8	0.2245	1.2136	-0.0117	3.0152	12.6066	1.2341	5.3899	0.1855	5.4978	0.0504	8.9244e...	0.0024	0.1336	0.2352	5.3603
9	0.1981	1.2767	-0.0055	3.0093	12.3766	1.2920	4.8951	0.2043	6.5219	0.0392	9.7810e...	0.0162	0.1513	2.5592	37.2576
10	0.3107	1.8131	-0.0019	3.0480	19.9742	1.8395	5.4568	0.1851	5.9209	0.0965	1.9826e...	0.0055	0.1316	0.5595	4.0233
11	0.2110	1.0253	-0.0044	3.0171	10.7680	1.0468	5.2483	0.1905	4.9605	0.0445	6.4211e...	0.0066	0.1349	0.3923	5.5108
12	0.2352	0.9694	0.0013	2.9980	9.1121	0.9975	4.6428	0.2226	4.2409	0.0553	5.8303e...	0.0031	0.1333	0.4810	4.5991
13	0.2148	0.9873	-0.0011	3.0008	9.2496	1.0104	4.6140	0.2202	4.7030	0.0462	5.9820e...	0.0072	0.1358	0.5423	4.3736
14	0.5628	1.3287	-0.0042	3.2640	14.0192	1.4430	5.3653	0.1864	2.5639	0.3168	1.2201e...	0.0254	0.2176	4.3544	46.8549
15	0.5491	1.4610	0.0045	3.2417	16.7313	1.5608	5.7844	0.1729	2.8425	0.3015	1.4274e...	-0.0068	0.1592	-0.4152	8.7547
16	0.5542	1.2688	-0.0255	3.1865	13.8771	1.3845	5.8424	0.1712	2.4983	0.3071	1.1232e...	-0.0063	0.1528	-0.5741	7.3615
17	0.5578	1.2345	-0.0118	3.1956	13.3870	1.3546	5.9445	0.1682	2.4285	0.3111	1.0752e...	0.0174	0.2257	6.6507	93.2861
18	0.5297	1.2267	-0.0102	3.1431	12.7488	1.3362	5.1928	0.1926	2.5227	0.2805	1.0461e...	-0.0037	0.1606	-0.1072	9.1053
19	0.5363	1.2640	-0.0139	3.1121	12.1908	1.3731	4.7475	0.2106	2.5601	0.2877	1.1047e...	0.0029	0.1681	2.0436	24.7672
20	0.5481	1.8614	-0.0388	3.8175	18.9103	1.9405	5.1648	0.1936	3.5403	0.3004	2.2063e...	-0.0161	0.1686	-1.0296	7.8992
21	0.5332	0.9758	0.0034	3.0865	10.0900	1.1120	5.2333	0.1911	2.0855	0.2843	7.2454e...	0.0386	0.3062	6.9669	71.0895
22	0.5563	1.1097	0.0040	3.0583	10.5815	1.2413	4.6736	0.2140	2.2315	0.3094	9.0278e...	-0.0046	0.1441	-0.2472	7.0366
23	0.5503	1.1166	-0.0011	3.0712	11.1882	1.2449	5.0787	0.1969	2.2620	0.3029	9.0800e...	-0.0016	0.1428	-0.1825	6.9284

Figure 4.4 Statistical features extracted from both time domain and frequency domain

Mean, skewness, standard deviation, kurtosis, peak-to-peak, root mean square, crest factor, shape factor, impulse factor, margin factor, and energy are some of the time domain statistical features shown in the figure above. In contrast, we included statistical features from the frequency domain, specifically spectral kurtosis. These include mean, standard deviation, skewness and kurtosis of the spectral kurtosis.

4.5. Feature postprocessing

Typically, the above-mentioned extracted time domain and frequency domain features are associated with noise. The occurrence of noise will harm our RUL prediction. In the following section, we will discuss Monotonicity, a feature performance metric. We must use the classic moving mean filter since this metric is sensitive to noise. The feature is shown in Figure 4.5 before and after smoothing. In this paper, we implemented a casual moving mean with a lag window of 5 steps. This signal delay for smoothing can be achieved by selecting the appropriate RUL prediction threshold.

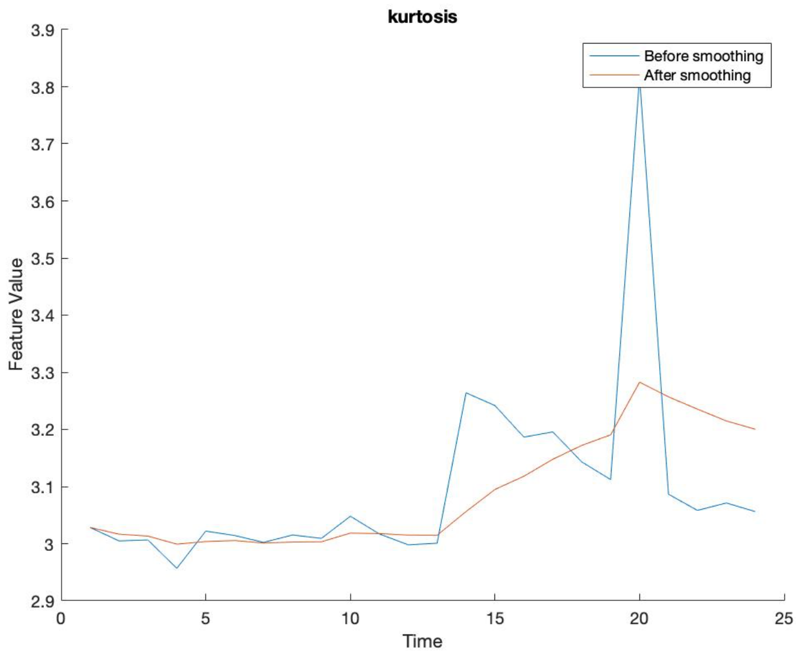

Figure 4.5 Before and after smoothing features

4.6. Training Data and feature importance ranking

When designing the prognostic algorithm, keep in mind that some data was obtained at the beginning of the entire life cycle. As a result, we must assume that some percentage of the data will be treated as training data. In this paper we have used 66.67% of the data for training.

Figure 4.6 shows that the margin factor is the most important feature based on monotonicity, and that we used a minimum of 0.3 feature importance score to select features for feature fusion.

Similarly, figure 4.7 displays features that were chosen based on a feature importance score greater than 0.3. This feature selection is critical for feature fusion following dimension reduction with principal component analysis (PCA).

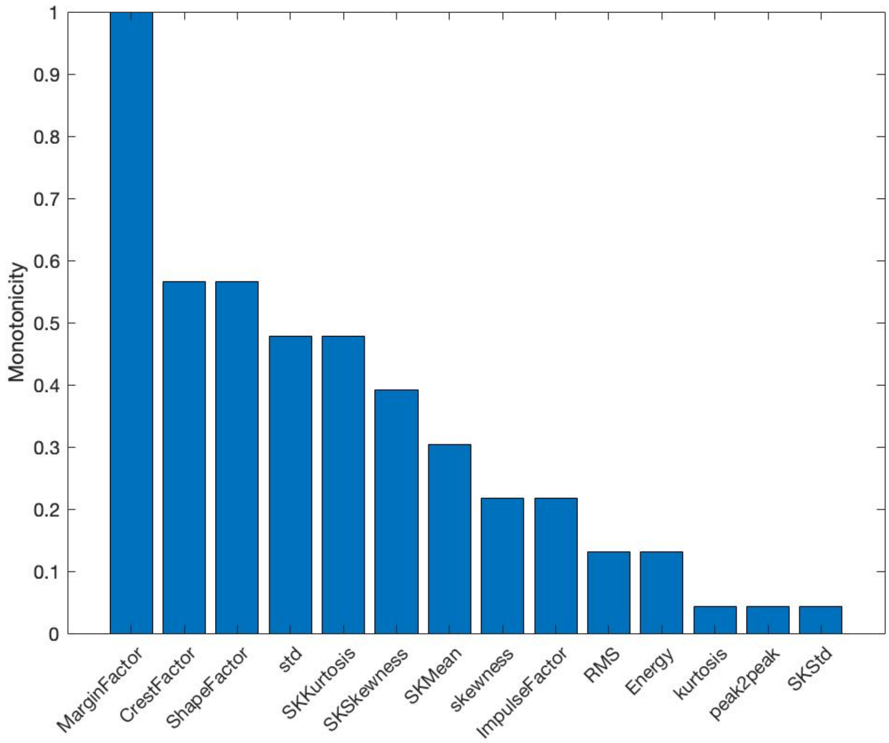

Figure 4.6 Feature selection scores based on monotonicity

```
featureSelected =

  24×7 table
```

mean	RMS	CrestFactor	ImpulseFactor	Energy	SKStd	SKSkewness
0.76379	1.5824	4.332	2.0718	1.4672e+06	0.28451	8.5011
0.76318	1.5712	4.3338	2.0588	1.4466e+06	0.27549	8.9389
0.76551	1.5298	4.3741	1.9987	1.3733e+06	0.22745	6.0647
0.75467	1.5705	4.4147	2.0851	1.4497e+06	0.22537	6.9079
0.75563	1.5352	4.4946	2.0351	1.3874e+06	0.20657	5.6088
0.76151	1.486	4.4183	1.9572	1.3063e+06	0.19455	4.794
0.75824	1.4554	4.4277	1.9253	1.253e+06	0.17043	3.5956
0.66856	1.4011	4.6034	2.5006	1.1641e+06	0.14828	2.072
0.57321	1.3753	4.6768	3.2745	1.1227e+06	0.15159	2.4458
0.50463	1.3998	4.8301	3.8707	1.1732e+06	0.13701	0.96615
0.41323	1.342	4.9025	4.3916	1.0906e+06	0.13759	0.96274
0.32061	1.3015	5.0035	4.8371	1.0376e+06	0.1374	0.92296
0.23239	1.2367	5.0411	5.3075	9.4609e+05	0.13674	0.79492
0.28878	1.2715	5.037	4.8185	1.0007e+06	0.15075	1.4815
0.34728	1.3163	5.1853	4.2053	1.0756e+06	0.15208	0.98572
0.38786	1.2405	5.2495	3.6349	9.3233e+05	0.1556	0.79679
0.44565	1.2918	5.3656	3.2129	1.0045e+06	0.17074	1.8399
0.49473	1.3483	5.4572	2.9265	1.0817e+06	0.17529	1.7418
0.54831	1.4087	5.4795	2.5694	1.1661e+06	0.18066	1.992
0.54586	1.4916	5.4461	2.7321	1.3305e+06	0.17249	1.0947
0.54321	1.4168	5.3542	2.6059	1.2133e+06	0.19699	2.3251
0.54356	1.3929	5.1594	2.5614	1.1766e+06	0.19555	2.3795
0.54232	1.3746	5.0151	2.5337	1.1487e+06	0.18174	1.2407
0.54476	1.3596	5.042	2.4947	1.1259e+06	0.17914	1.2491

Figure 4.7 Features with feature importance score larger than 0.3

4.7. Dimension reduction and feature fusion

Dimensionality reduction is a critical step in the selection of important statistical features. This is also the primary focus of this research paper. We chose a few important features in the preceding section by using a minimum feature importance score value of 0.3. This has resulted in the removal of more than half of the statistical features proposed for determining the detection of faults in wind turbine bearings. These statistical features are always correlated and contain redundant information. We have a higher computational complexity because of the correlations and redundancies among the statistical features, so we must use principal component analysis for dimensional reduction while retaining important data information, as explained in the introduction of this paper section 2.2.3.

In this experiment, we normalized the features into the same scales before implementing the principal component as dimension reduction and feature fusion. These normalized PCA coefficients, mean, and standard deviation were derived from training data and applied to the entire dataset.

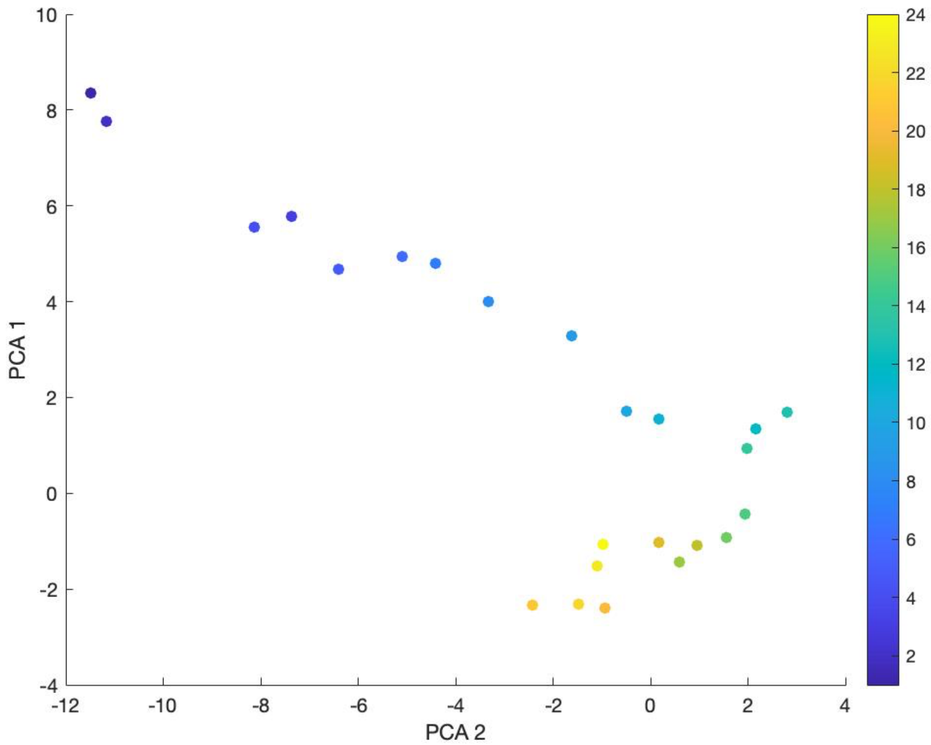

Figure 4.8 Principal components

The plot in figure 4.8 shows that the second principal component increases as the wind turbine approaches failure. Therefore, the second principal component is considered as a promising fused health indicator.

The second principal component is then visualized, as shown in figure 4.9.

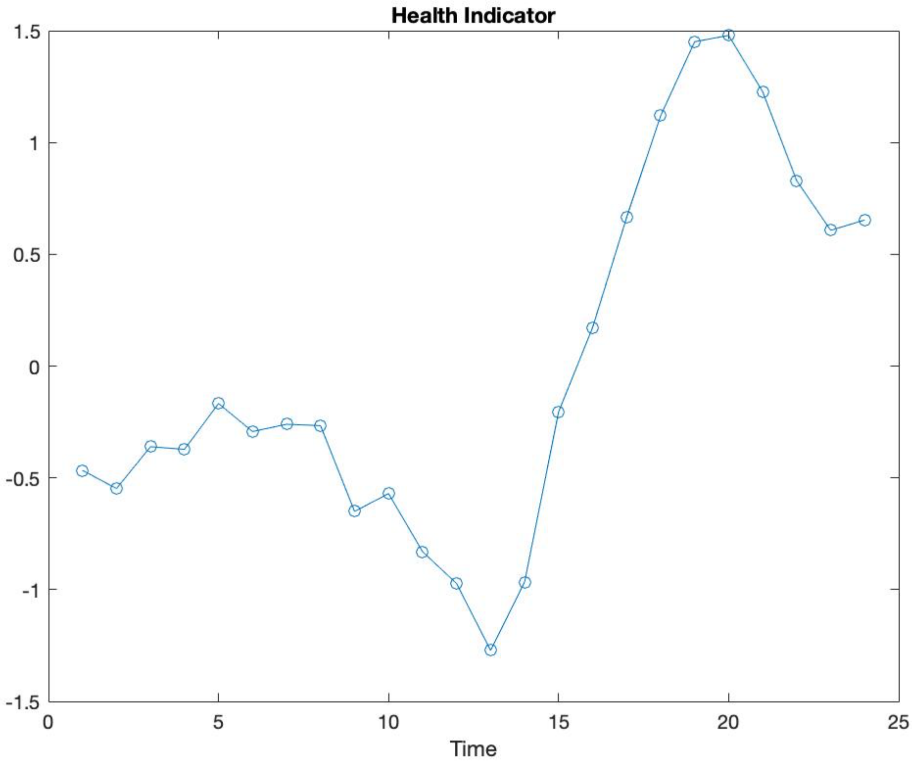

Figure 4.9 Visualisation of health indicator

4.8. Exponential degradation models for remaining useful life (RUL) estimation

The following step will be to create a model for high-speed bearing prognosis. It is the second critical step in determining the remaining useful time (RUL) after a fault has been detected. In this study, we used the exponential degradation model to demonstrate the exponential degradation process and estimate the RUL. Degradation models, in general, estimate the RUL by predicting when a signal exceeds a predefined threshold.

The exponential degradation model can be defined mathematically as follows:

$$S(t) = \phi + \theta(t)\, e^{(\beta(t) + \epsilon(t) - \frac{\varepsilon^2}{2})}$$

where ϕ is a constant model intercept that can be a lower or upper bound depending on the sign of θ. $\theta(t)$ is a random variable with theta mean and variance that is modelled as a lognormal distribution.$\beta(t)$ can also be modelled as gaussian distribution with beta mean and variance.$\epsilon(t)$ is a noise modelled as a normal distribution with zero mean and variance, $\sigma^2$is the called the noise variance [16].

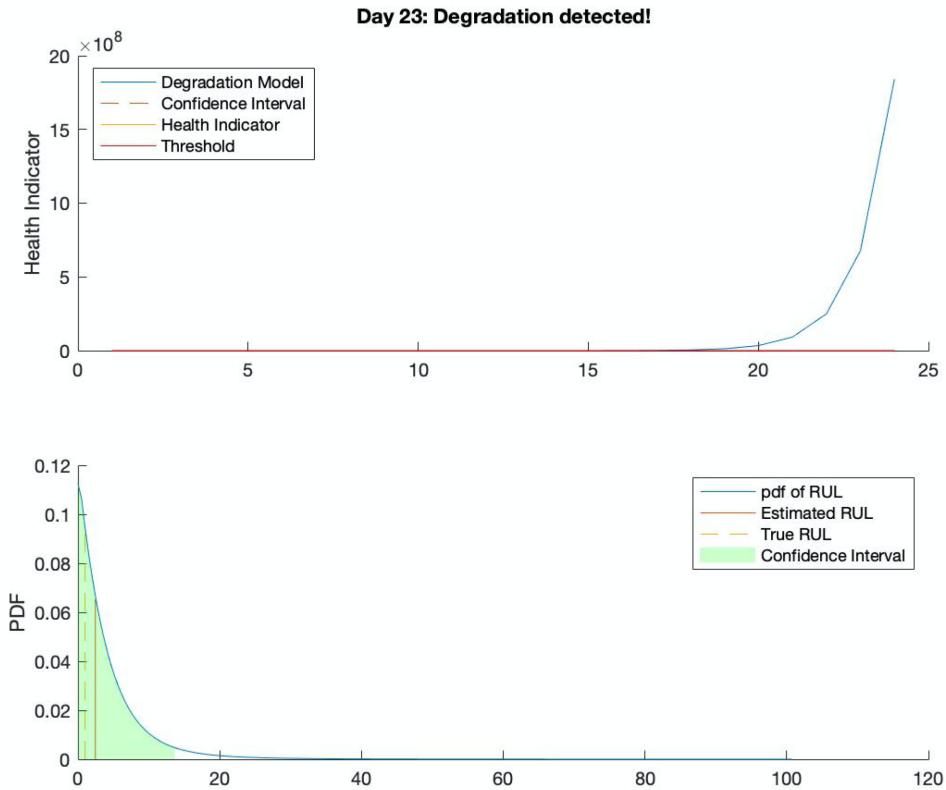

Figure 4.10 RUL estimation using exponential model

4.9. Performance analysis

There are various performance metrics used for prognostic performance analysis, as discussed in section 3.8. We used the $\alpha - \lambda$ metrics in this paper, where we set the λ to 20% and estimated the RUL within the α bound. In the figure 4.11 we have plotted the prognostic performance analysis.

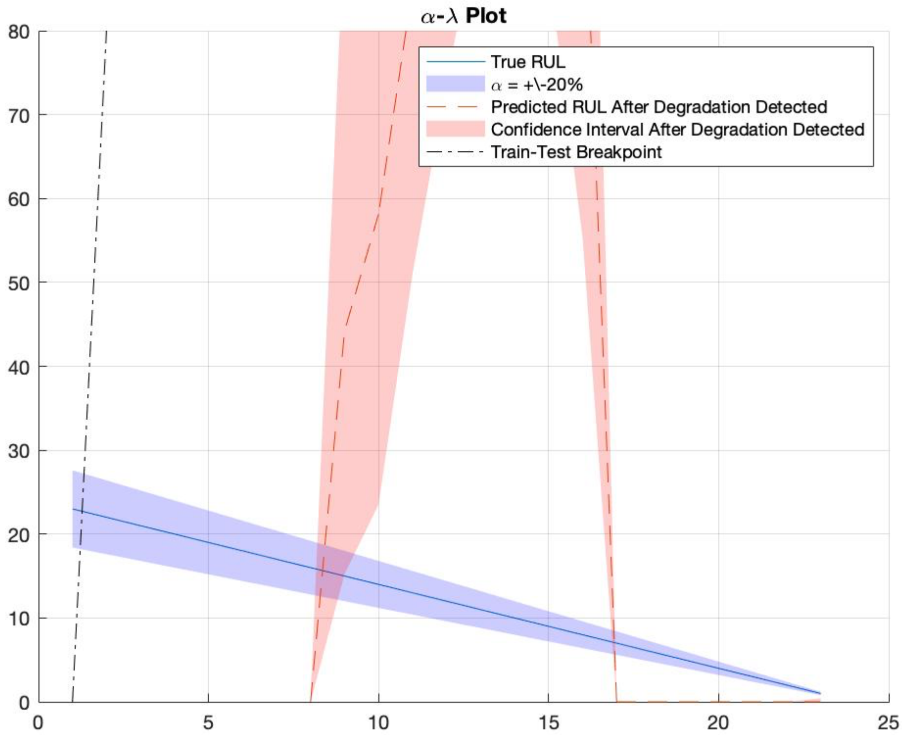

Figure 4.11 $\alpha - \lambda$ plot for prognostic performance analysis

The probability within λ to 20% bound is plotted in the figure 4.12 below. we have plotted the probability within λ to 20% bound. As more data points become available, the prediction becomes more accurate.

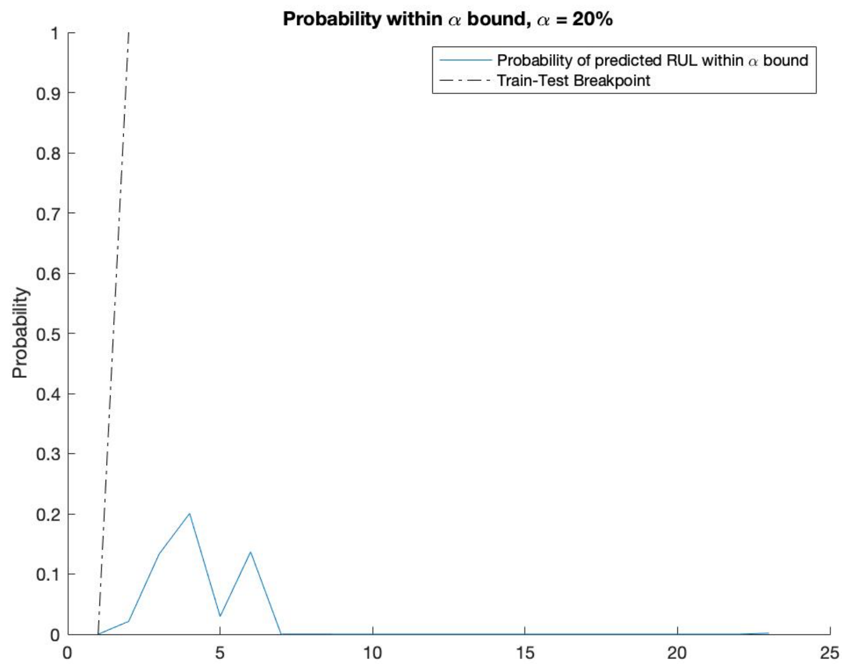

Figure 4.12 Probability of the predicted RUL within the α-bound

5. CONCLUSION AND FUTURE WORK

5.1. Summary of the research

This paper generally discussed faults and fault detection in rotating machineries, taking wind turbine high-speed bearings as a test case. In this work, PCA is used as dimension reduction and feature fusion tool to detect faults and determine the RUL of the bearings. A general prognostic approach with steps such as health indicator (HI) construction, degradation indicator and remaining useful life (RUL) estimation is discussed in section 3.2. This paper implemented this prognostic approach in real time.

In this paper, statistical features of time domain such as mean, standard deviation, peak-to-peak, crest factor, skewness, kurtosis, energy, margin factor, RMS, shape factor and impulse factor were used. Similarly, statistical features were extracted from the frequency domain, specifically spectral kurtosis. After selecting the important features, one of the features is selected as a PCA component. The selected PCA component is used as HI and as a trigger for the degradation indicator, if the HI goes below a predefined threshold.

Generally, the approach discussed in this paper is very important and applicable in different real-life situations. Future works can be done using different machineries depending on the availability of data. Further research can be done on different topics such as:

- Further investigation of the effectiveness of the implementation of PCA in large datasets of wind turbines
- Fault detection using PCA in other machinery datasets
- Condition monitoring techniques and comparison among the techniques

6. REFERENCES

[1] S.Edwards, A.W. Lees and M.I. Friswell Fault Diagnosis of rotating machinery

[2] N. Tandon, A.Choudhury A review of vibration and acoustic measurement methods for the detection of defects in rolling element bearings, TribiologyInternationsl 32(1999) 469-480

[3] http://data-acoustics.com/measurements/gear-faults/gear-1/

[4] Dr. f. R, Gomaa , Dr.k.M. Kahder, Eng M.A.Eissa Fault Diagnosis of Rotating Machinery based on vibration analysis. International Journal of Advanced Engineering and Global Technology vol-04, issue-01, January 2016ß

[5] Koceila Abid, Moamar Sayed Mochaweh, and Laurence Cornez, Fault Prognostics for the Predictive Maintenance of Wind Turbines: State of the Art, 5July 2019

[6] Yaguo Lei, Naipeng Li,Liang Guo,Ningbo Li, Tao Yan, Jing Lin, Machinary health Prognostics: A systematic review from data acquisition to RUL prediction

[7] F. Camci, K. Medjaher, N. Zerhounib, P. Nectouxb, Feature evaluation for effective bearing prognostics, Qual. Reliab. Eng. Int. 29 (2013) 477–486.

[8] Saxena, A., Celaya, J., Balaban, E., Goebel, K., Saha, B., Saha, S., Schwabacher, M.: Metrics for evaluating performance of prognostic techniques. In: Prognostics and health management, 2008. phm 2008. international conference on. pp. 1–17. IEEE (2008)

[9] Gao, Z.; Liu, X. An Overview on Fault Diagnosis, Prognosis and Resilient Control for Wind Turbine Systems. Processes 2021, 9, 300

[10] Hahn, B.; Durstewitz, M.; Rohrig, K. Reliability of Wind Turbines. Wind Energy Eng. 2007, 329–332

[11] Verbruggen, T. Wind turbine operation & maintenance based on condition monitoring. In ECN Wind Energy; Technical Report ECN-C-03-047; ECN: Petten, The Netherlands, 2003.

[12] Qiao, W.; Lu, D. A survey on wind turbine condition monitoring and fault diagnosis-part I: Components and subsystems. IEEE Trans. Ind. Electron. 2015, 62, 6536–6545. [CrossRef]

[13] Lau, B.; Ma, E.; Pecht, M. Review of offshore wind turbine failures and fault prognostic methods. In Proceedings of the IEEE Conference on Prognostics and System Health Management (PHM), Beijing, China, 23–25 May 2012; pp. 1–5.

[14] Elasha F, Shanbr S, Li X, Mba D. Prognosis of a Wind Turbine Gearbox Bearing Using Supervised Machine Learning. Sensors (Basel). 2019;19(14):3092. Published 2019 Jul 12. doi:10.3390/s19143092

[15] Ali, Jaouher Ben, et al. "Online automatic diagnosis of wind turbine bearings progressive degradations under real experimental conditions based on unsupervised machine learning." Applied Acoustics 132 (2018): 167-181.

[16] Gebraeel, Nagi. "Sensory-Updated Residual Life Distributions for Components with Exponential Degradation Patterns." IEEE Transactions on Automation Science and Engineering. Vol. 3, Number 4, 2006, pp. 382–393.

[17] Coble, J. "Merging Data Sources to Predict Remaining Useful Life - An Automated Method to Identify Prognostics Parameters." Ph.D. Thesis. University of Tennessee, Knoxville, TN, 2010.

[18] Tang, S.; Yuan, S.; Zhu, Y. Cyclostationary Analysis towards Fault Diagnosis of Rotating Machinery. Processes 2020, 8, 1217.

7. APPENDICES

Appendix 1. MATLAB code

```
filenames =
{'case2_1.mat','case2_2.mat','case2_3.mat','case2_4.mat','case2_5.ma
t','case2_6.mat','case2_7.mat','case3_1.mat','case3_2.mat','case3_3.
mat','case3_4.mat','case3_5.mat','case3_6.mat','case1_1.mat','case1_
2.mat','case1_3.mat','case1_4.mat','case1_5.mat','case1_6.mat','case
1_7.mat','case1_8.mat','case1_9.mat','case1_10.mat','case1_11.mat'};

for kk = 1:numel(filenames)
    load(filenames{kk})
end

x1=load('case2_1.mat');x2=load('case2_2.mat');x3=load('case2_3.mat')
;x4=load('case2_4.mat');x5=load('case2_5.mat');x6=load('case2_6.mat'
);
x7=load('case2_7.mat');x8=load('case3_1.mat');x9=load('case3_2.mat')
;x10=load('case3_3.mat');x11=load('case3_4.mat');x12=load('case3_5.m
at');x13=load('case3_6.mat');x14=load('case1_1.mat');x15=load('case1
_2.mat');x16=load('case1_3.mat');x17=load('case1_4.mat');x18=load('c
ase1_5.mat');x19=load('case1_6.mat');
x20=load('case1_7.mat');x21=load('case1_8.mat');x22=load('case1_9.ma
t');x23=load('case1_10.mat');x24=load('case1_11.mat');

t1=struct2table(x1,'AsArray',true);t2=struct2table(x2,'AsArray',true
);t3=struct2table(x3,'AsArray',true);t4=struct2table(x4,'AsArray',tr
ue);t5=struct2table(x5,'AsArray',true);t6=struct2table(x6,'AsArray',
true);t7=struct2table(x7,'AsArray',true);t8=struct2table(x8,'AsArray
',true);t9=struct2table(x9,'AsArray',true);t10=struct2table(x10,'AsA
rray',true);t11=struct2table(x11,'AsArray',true);t12=struct2table(x1
2,'AsArray',true);t13=struct2table(x13,'AsArray',true);t14=struct2ta
ble(x14,'AsArray',true);t15=struct2table(x15,'AsArray',true);t16=str
uct2table(x16,'AsArray',true);t17=struct2table(x17,'AsArray',true);t
18=struct2table(x18,'AsArray',true);t19=struct2table(x19,'AsArray',t
rue);t20=struct2table(x20,'AsArray',true);t21=struct2table(x21,'AsAr
ray',true);t22=struct2table(x22,'AsArray',true);t23=struct2table(x23
,'AsArray',true); t24=struct2table(x24,'AsArray',true);



T=[t1;t2;t3;t4;t5;t6;t7;t8;t9;t10;t11;t12;t13;t14;t15;t16;t17;t18;t1
9;t20;t21;t22;t23;t24]

fs = sr; % Hz

tstart = 0;
figure
hold on
for i=1:24

    v = T.gs{i};
    t = tstart + (1:length(v))/fs;
    % Downsample the signal to reduce memory usage
    plot(t(1:10:end), v(1:10:end))
    tstart = t(end);
```

```
end
hold off
xlabel('Time (s), 6 second per day, 24 days in total')
ylabel('Acceleration (g)')


hsbearing.DataVariables = ["vibration", "tach", "SpectralKurtosis"];
colors = parula(24);
figure
hold on
day = 1;
for i=1:24

    data2add = table;
    % Get vibration signal and measurement date
    v = T.gs{i};

    % Compute spectral kurtosis with window size = 128
    wc = 128;
    [SK, F] = pkurtosis(v, fs, wc);
    data2add.SpectralKurtosis = {table(F, SK)};

    % Plot the spectral kurtosis
    plot3(F, day*ones(size(F)), SK, 'Color', colors(day, :))

    % Write spectral kurtosis values
    % writetable(hsbearing, data2add);
    hsbearing=table2struct(data2add,'ToScalar',true);


    % Increment the number of days
    day = day + 1;
end
hold off
xlabel('Frequency (Hz)')
ylabel('Time (day)')
zlabel('Spectral Kurtosis')
grid on
view(-45, 30)
cbar = colorbar;
ylabel(cbar, 'Fault Severity (0 - healthy, 1 - faulty)')

k=[T.gs{1,1},T.gs{2,1},T.gs{3,1},T.gs{4,1},T.gs{5,1},T.gs{6,1},T.gs{
7,1},T.gs{8,1},T.gs{9,1},T.gs{10,1},T.gs{11,1},T.gs{12,1},T.gs{13,1}
,T.gs{14,1},T.gs{15,1},T.gs{16,1},T.gs{17,1},T.gs{18,1},T.gs{19,1},T
.gs{20,1},T.gs{21,1},T.gs{22,1},T.gs{23,1},T.gs{24,1}];

A = array2table(k);
C = table2array(A);
D = mean(C,1);
E = std(C,1);
F = skewness(C,1);
G = kurtosis(C,1);
H = peak2peak(C,1);
I = rms(k);
J= peak2rms(k,1);
L=max(k);
M=bsxfun(@rdivide,I,L);
N=abs(D);
```

```
O=bsxfun(@rdivide,I,N);
P=bsxfun(@rdivide,I,N.^2);
P=bsxfun(@rdivide,I,(N.^2));
P=N.^2;
Q=bsxfun(@rdivide,I,P);
R= sum(k.^2,1);

S
=[pkurtosis(T.gs{1,1},fs),pkurtosis(T.gs{2,1},fs),pkurtosis(T.gs{3,1
},fs),pkurtosis(T.gs{4,1},fs),pkurtosis(T.gs{5,1},fs),pkurtosis(T.gs
{6,1},fs),pkurtosis(T.gs{7,1},fs),pkurtosis(T.gs{8,1},fs),pkurtosis(
T.gs{9,1},fs),pkurtosis(T.gs{10,1},fs),pkurtosis(T.gs{11,1},fs),pkur
tosis(T.gs{12,1},fs),pkurtosis(T.gs{13,1},fs),pkurtosis(T.gs{14,1},f
s),pkurtosis(T.gs{15,1},fs),pkurtosis(T.gs{16,1},fs),pkurtosis(T.gs{
17,1},fs),pkurtosis(T.gs{18,1},fs),pkurtosis(T.gs{19,1},fs),pkurtosi
s(T.gs{20,1},fs),pkurtosis(T.gs{21,1},fs),pkurtosis(T.gs{22,1},fs),p
kurtosis(T.gs{23,1},fs),pkurtosis(T.gs{24,1},fs)];

MS= mean(S,1);
STS= std(S,1);
SKS= skewness(S,1);
KUS= kurtosis(S,1);

feature=[D.',E.',F.',G.',H.',I.',J.',M.',O.',P.',R.',MS.',STS.',SKS.
',KUS.'];
feature_table = array2table(feature);
feature_table.Properties.VariableNames = {'mean', 'std', 'skewness',
'kurtosis', 'peak2peak','RMS', 'CrestFactor', 'ShapeFactor',
'ImpulseFactor', 'MarginFactor',
'Energy','SKMean','SKStd','SKSkewness','SKKurtosis'};
hsbearing.SelectedVariables = ['mean', 'std', 'skewness',
'kurtosis', 'peak2peak','RMS', 'CrestFactor', 'ShapeFactor',
'ImpulseFactor', 'MarginFactor',
'Energy','SKMean','SKStd','SKSkewness','SKKurtosis'];
variableNames = feature_table.Properties.VariableNames;
featureTableSmooth = varfun(@(x) movmean(x, [5 0]), feature_table);
featureTableSmooth.Properties.VariableNames = variableNames;
figure
hold on
plot(feature_table.kurtosis)
plot(featureTableSmooth.kurtosis)
hold off
xlabel('Time')
ylabel('Feature Value')
legend('Before smoothing', 'After smoothing')
title('kurtosis')
trainData = featureTableSmooth(1:24, :);
featureImportance = monotonicity(trainData);
helperSortedBarPlot(featureImportance, 'Monotonicity');
trainDataSelected = trainData(:, featureImportance{:,:}>0.3);
featureSelected = featureTableSmooth(:, featureImportance{:,:}>0.3)
meanTrain = mean(trainDataSelected{:,:});
sdTrain = std(trainDataSelected{:,:});
trainDataNormalized = (trainDataSelected{:,:} - meanTrain)./sdTrain;
coef = pca(trainDataNormalized);
PCA1 = (featureSelected{:,:} - meanTrain) ./ sdTrain * coef(:, 1);
PCA2 = (featureSelected{:,:} - meanTrain) ./ sdTrain * coef(:, 2);
figure
numData = size(feature_table, 1);
scatter(PCA1, PCA2, [], 1:numData, 'filled')
```

```
xlabel('PCA 1')
ylabel('PCA 2')
cbar = colorbar;
healthIndicator = PCA2;
figure
plot( healthIndicator, '-o')
xlabel('Time')
title('Health Indicator')
healthIndicator = healthIndicator - healthIndicator(1);
threshold = healthIndicator(end);
mdl = exponentialDegradationModel(...
    'Theta', 1, ...
    'ThetaVariance', 1e6, ...
    'Beta', 1, ...
    'BetaVariance', 1e6, ...
    'Phi', -1, ...
    'NoiseVariance', (0.1*threshold/(threshold + 1))^2, ...
    'SlopeDetectionLevel', 0.05);


% Keep records at each iteration
totalDay = length(healthIndicator) - 1;
estRULs = zeros(totalDay, 1);
trueRULs = zeros(totalDay, 1);
CIRULs = zeros(totalDay, 2);
pdfRULs = cell(totalDay, 1);

% Create figures and axes for plot updating
figure
ax1 = subplot(2, 1, 1);
ax2 = subplot(2, 1, 2);

for currentDay = 1:totalDay

    % Update model parameter posterior distribution
    update(mdl, [currentDay healthIndicator(currentDay)])

    % Predict Remaining Useful Life
    [estRUL, CIRUL, pdfRUL] = predictRUL(mdl, ...
                                         [currentDay
healthIndicator(currentDay)], ...
                                         threshold);
    trueRUL = totalDay - currentDay + 1;

    % Updating RUL distribution plot
    helperPlotTrend(ax1, currentDay, healthIndicator, mdl,
threshold);
    helperPlotRUL(ax2, trueRUL, estRUL, CIRUL, pdfRUL)

    % Keep prediction results
    estRULs(currentDay) = estRUL;
    trueRULs(currentDay) = trueRUL;
    CIRULs(currentDay, :) = CIRUL;
    pdfRULs{currentDay} = pdfRUL;

    % Pause 0.1 seconds to make the animation visible
    pause(0.1)
end
```

```
alpha = 0.2;
detectTime = mdl.SlopeDetectionInstant;
prob = helperAlphaLambdaPlot(alpha, trueRULs, estRULs, CIRULs, ...
    pdfRULs, detectTime);
title('\alpha-\lambda Plot')
figure
t = 1:totalDay;
hold on
plot(t, prob)
plot([0 1], 'k-.')
hold off
ylabel('Probability')
legend('Probability of predicted RUL within \alpha bound', 'Train-
Test Breakpoint')
title(['Probability within \alpha bound, \alpha = '
num2str(alpha*100) '%'])
```