Chapter 3

MACHINE LEARNING METHODS IN GEAR FAULT DIAGNOSIS

3.1 INTRODUCTION

With advances in computer technology, it is possible to acquire, store and process a large amount of data. Most of the data acquisition systems are capable of logging real time data in digital form reliably. The technological development that goes into the memory devices makes it possible to reduce the cost and size required to store large data and they are much more reliable than the older ones. The processing speed of the processor has shown exponential growth and the availability of high speed processors today allows engineers to solve complex problems. Many of the machine learning methods is iterative in nature and they require high speed processors. The aforesaid developments facilitated the application of machine learning methods to real time problems. Fault diagnosis is one of the application areas, where machine learning methods are widely used. This chapter briefly describes some of the machine learning methods used for gear fault diagnosis purpose.

3.2 GEAR FAULTS AND DIAGNOSIS

Gears are one of the widely used components in the rotary machineries and a bevel gear box is chosen for investigation in the present study. There are several faults possible in a gear assembly like face wear, broken tooth, root crack, pitting, spalling, scoring, gear meshing etc. However only the following conditions of the gear were considered for the study:

a) Gear in good condition (Good)

b) Gear tooth breakage (GTB)

c) Gear with crack at root (GTC)
d) Gear with face wear (TFW)

In the present study, four bevel pinion wheels were used. One was used in good condition. In the other pinion wheels, defects were created; Wire-cut Electric Discharge Machining (EDM) was used to ensure precisely defined small defect. The size of gear tooth breakage was 8 mm, gear with crack at root was 0.8 mm × 0.5 mm × 20 mm and gear with face wear is 0.5mm. Diagnosis of the faults in gears was done using a number of techniques. They can be grouped under four categories as follows:

a) Traditional Approach  
b) Time domain Analysis  
c) Frequency domain Analysis  
d) Wavelet Analysis.  
e) Machine Learning Approach

3.3 TRADITIONAL APPROACH

The initial research in the area of transmission damage detection was focused on vibration signal analysis, i.e. the analysis of transmission vibration signals using the various tools available in the signal processing community. At first, the statistical characteristics of the signal in the time domain were the primary focus of study. However, the field quickly expanded to include spectral analysis, time–frequency analysis and wavelet analysis. This field is continuing to grow. As new signal processing techniques emerge, they are applied to the transmission damage detection problem and their merit as potential damage detection techniques is assessed.

3.3.1 Statistical damage detection metrics

3.3.1.1 Theory

The traditional techniques for vibration-based transmission damage detection are typically based on some statistical measurement of the energy of the vibration signal. For example, it was observed that the rms energy of the vibration signal may change in the presence of damage. However, in the mid 1970s, Stewart [24] began to investigate more
rigorously investigate the changes in a transmission vibration signal due to gear damage. He made the observation that under no fault conditions, a transmission vibration signal tends to be dominated by the mesh frequency and its harmonics, referred to as the regular meshing components of the signal, and that a noise component is present which can generally be assumed to be Gaussian. He also noted the presence of sidebands about the regular meshing components, spaced at the rotational frequency of the gear of interest. Based on these observations, McFadden [119] proposed a basic mathematical model of the transmission vibration signal. He stated that a signal from a perfect transmission can be approximated by

\[ x_{\text{perfect}}(t) = \sum_{n=0}^{\infty} P_n \cos(n\omega t + \phi_n) + w(t) \]  

where \( P_n \) is the amplitude of the \( n \)th harmonic, \( \omega \) is the mesh frequency given by \( \omega = 2\pi f_r \), \( f_r \) is the gear rotation frequency, \( T \) is the number of teeth, \( \phi_n \) is the phase angle of the \( n \)th harmonic, and \( w(t) \) is the Gaussian white noise. However, imperfections in the manufacturing and assembly of the transmission lead to amplitude and phase modulation of the signal. These modulations have the effect of creating the sidebands about the regular meshing components. Thus, McFadden noted that the amplitude modulation function \( a_n(t) \) and the phase modulation function \( b_n(t) \) are periodic with the gear rotational frequency \( f_r \) and can be written as a sum of its harmonics as

\[ a_n(t) = \sum_{m=0}^{\infty} A_{mn} \cos(m\omega_r t + \alpha_{mn}), \quad b_n(t) = \sum_{m=0}^{\infty} B_{mn} \cos(m\omega_r t + \beta_{mn}) \]  

Where \( \omega_r = 2\pi f_r \). Substituting these modulation functions into \( x_{\text{perfect}}(t) \) yields a more accurate model of a transmission vibration signal, given as

\[ x(t) = \sum_{n=0}^{\infty} P_n \left[ 1 + a_n(t) \right] \cos(n\omega t + \phi_n + b_n(t)) + w(t) \]
Stewart [24] observed that in a healthy transmission, the amplitude of the sidebands, and thus the amplitude of the modulation functions, was small. However, he noted that the presence of a fault superimposes additional dynamics upon this signal and that different types of faults have different effects on the vibration signal. In major tooth faults, for example, the peak-to-peak value of \( x(t) \) tends to increase. However, for heavily distributed damage, the peak-to-peak value can remain rather constant while the amplitude of the mesh frequency and its harmonics, \( P_n \), decrease. He also made the important observation that the additional dynamics caused by the appearance of a fault tend to significantly increase the amplitude of the sidebands, in particular, in the presence of a local defect such as a single tooth failure. This corresponds to an increase in the amplitude of the modulation functions in McFadden’s model.

To better observe the changes in the amplitude of the modulation functions, Stewart proposed the formation of a difference signal \( d \); the statistical properties of which could then be analyzed. The difference signal is given as

\[
d = x(t) - y_d(t)
\]  

(3.4)

where \( x(t) \) is the original time synchronous signal and \( y_d(t) \) is the signal containing the mesh frequencies, their harmonics and their first-order sidebands. Thus, according to McFadden’s model, \( d \) is composed of the higher-order sidebands and Gaussian noise. Later, it was proposed that the first-order sidebands need not be removed [25]. This led to the formation of the residual signal, \( r \), given as

\[
r = x(t) - y_r(t)
\]  

(3.5)

where, in this case, \( y_r(t) \) is the signal containing only the mesh frequencies and their harmonics. These observations of the nature of the changes in the transmission vibration signal in the presence of damage form the foundation of the statistical transmission damage detection metrics which follow.
3.3.1.2 Diagnostic techniques

3.3.1.2.1 Root mean square value

The Root Mean Square value (RMS) for the velocity vibration signal is defined in Equation 3.6. In comparison with the definition of kinetic energy, it is obvious that the RMS value computed from the velocity of the vibration signal describes the energy content of the signal.

\[ v_{rms} = \sqrt{\frac{1}{t} \int v^2(t) \, dt} \]  \hspace{1cm} (3.6)

where \( v_{rms} \) is the root mean square value of the velocity of the vibration signal,

\( t \) is integration time, \( v \) is the velocity of the moving object.

From the definition of RMS it is obvious that the RMS value does not increase with the isolated peaks in the signal, consequently this parameter is not sensitive to incipient tooth failure. Its value increases as tooth failure progresses. Generally the RMS value of the vibration signal is a very good descriptor of the overall condition of the tested gearboxes. This parameter is sensitive to gearbox load and speed changes. The main usage of this parameter is to monitor the overall vibration level.

3.3.1.2.2 Peak value

This value is the maximum value of the signal in a selected time interval. This parameter is usually not used alone.

3.3.1.2.3 Crest factor

This parameter indicates the damage in an early stage. It is defined as the peak value of the signal divided by the RMS value of the signal.

\[ CF = \frac{S_{\text{peak-peak}}}{S_{\text{rms}}} \]  \hspace{1cm} (3.7)
Where, $CF$ is the crest factor, $s_{\text{peak-peak}}$ is the peak to peak value of the signal, $s_{\text{rms}}$ is the root mean square value of the vibration signal. When only one tooth is damaged, there is no change in the RMS value of the vibration signal during one rotation of the drive shaft where the damaged gear is located, while the peak value increases. Therefore, the crest factor increases its value. As the damage progresses, the root mean square value of the vibration signal increases its value and the crest factor decreases. This parameter enables very tiny surface damages to be discovered, as experiments show. The crest factor is often used in gearbox quality monitoring devices.

### 3.3.1.2.4 Energy operator

The energy operator ($EO$) is computed as the normalized kurtosis from the signal where each point is computed as the difference of two squared neighborhood points of the original signal [28].

$$EO = \frac{N^2 \sum_{i=0}^{N} (\Delta x_i - \Delta \bar{x})^4}{(\sum_{i=0}^{N} (\Delta x_i - \Delta \bar{x})^2)^2}$$

(3.8)

Where, $EO =$ the energy operator, $\Delta \bar{x} =$ the mean values of the signal, $\Delta x_i = s_{i+1}^2 - s_i^2$, $N =$ the number of point on dataset $x$.

### 3.3.1.2.5 Kurtosis

The shape of the amplitude distribution is often used as a data descriptor. Kurtosis describes how peaked or flat the distribution is. If a vibration signal contains sharp peaks with a higher value, then its distribution function will be sharper. We can assume that these types of signals will be produced by a damaged gearbox. Therefore, the kurtosis value will be higher for a damaged gearbox than for a gearbox in good condition.

A mathematical definition of kurtosis is given by Eq. 3.9.

$$Kurt = \frac{N \sum_{i=0}^{N} (s_i - \bar{s})^4}{(\sum_{i=0}^{N} (s_i - \bar{s})^2)^2}$$

(3.9)
Where, \(Kurt\) is kurtosis, \(N\) is the number of points in the time history of signal \(s\), \(s_i\) is the \(i\)-th point in the time history of signal \(s\).

### 3.3.1.2.6 Energy ratio

The energy ratio \((ER)\) is defined as the ratio between the energy of the difference signal and the energy of the regular meshing component [29].

\[
ER = \frac{\sigma(d)}{\sigma(r)}
\]  

(3.10)

Where, \(ER\) is energy ratio, \(\sigma(d)\) is the standard deviation of the difference signal, \(\sigma(r)\) is the standard deviation of the regular signal. The basic idea of this indicator is that the energy is transferred from the regular meshing component to the rest of the signal as the wear progresses. This parameter is a very good indicator for heavy wear, where more than one tooth on the gearing is damaged.

### 3.3.1.2.7 FM4 parameter

The \(FM4\) parameter [28] is a simple measure if the amplitude distribution of the difference signal is peaked or flat. The parameter assumes that a gearbox in good condition has a difference signal with a Gaussian amplitude distribution, whereas a gearbox with defective teeth produces a difference signal with a major peak or a series of major peaks resulting in a less peaked amplitude distribution. If more then one tooth is defective, the data distribution becomes flat and the kurtosis value decreases.

\[
FM4 = \frac{N \sum_{i=0}^{N} (d_i - \bar{d})^4}{\left(\sum_{i=0}^{N} (d_i - \bar{d})^2\right)^2}
\]  

(3.11)

Where, \(N\) is the number of points in the time history of signal \(s\), \(d_i\) is the \(i\)-th point in the time history of signal \(s\).
3.3.1.2.8 NA4 parameter

The NA4 [28] parameter was developed to improve the behavior of the FM4 parameter when more than one tooth is damaged. The first difference between NA4 and FM4 is that NA4 uses a residual signal to compute kurtosis. The second difference is that we use an average value of variance. Thus if the gear damage spreads from one tooth to another tooth the value of the average variance increases slowly and allows the NA4 parameter to grow. The second reason why the NA4 parameter increases its value is that the residual signal contains the first order sidebands, which increase when tooth damage occurs. The NA4 parameter is defined by Eq. 3.12.

\[
NA4 = \frac{N \sum_{i=1}^{N} (r_i - \bar{r})^4}{\left( \frac{1}{M} \sum_{j=1}^{M} \left( \sum_{i=1}^{N} (r_{ij} - \bar{r}_j)^2 \right) \right)^2}
\]  

(3.12)

Where, N is the number of points in the time history of signal s, \( r_i \) is the i-th point in the time history of residual signal, \( r_{ij} \) is the i-th point in the j-th time record of the residual signal, j is the current time record, I is the data point number per reading and M is the current time record in the run ensemble.

3.4 TIME DOMAIN ANALYSIS

In the late 1980s, investigations into the joint time–frequency characteristics of transmission vibration signals were begun [120]. It was observed that local gear faults such as spalling on a limited number of gear teeth produce sharp transients in the vibration signal of a transmission. The statistical damage detection metrics tend to have an underlying assumption of signal stationarity and thus provide information about a signal which is averaged over time. As a result, the effect of local transient phenomena can be lost. On the other hand, nonstationary techniques provide information about the local time-domain properties of a signal. In general, a transmission vibration signature consists of three significant components: a sinusoidal component due to time varying loading, a broad-band impulsive component due to impact, and random noise. For an undamaged transmission, the sinusoidal components dominate. However, as damage propagates through the system, the sinusoidal components exhibit both modulation and
reduction in amplitude [121, 67]. In addition, both the broad-band impulsive components and the random noise become more prevalent [122]. It can be seen that the trends exhibited by the sinusoidal components are most visible in the frequency domain; the trends exhibited by the broad-band impulsive components are most visible in the time domain. Thus, in order to capture these trends, non-stationary, time–frequency analysis techniques are deemed appropriate [123]. Initial research into the use of joint time–frequency signal processing techniques for transmission damage detection was focused primarily on two of the standard time–frequency analysis techniques available in the signal processing community, the spectrogram and the Wigner–Ville distribution (WVD). The spectrogram is the squared magnitude of the short-time Fourier transform (STFT) and provides the energy density spectrum of the signal as a function of time [124]. To investigate the frequency domain properties of a signal about time \( t \), the signal \( x(t) \) is first multiplied by a window function \( h(t) \), centered at \( t \) this leads to the formation of a windowed signal given as

\[
x_w(t) = x(\tau)h(\tau - t)
\]  

(3.13)

The window function is chosen such that

\[
x_w(t) \sim \begin{cases} 
  x(\tau), & \text{for } \tau \text{ near } t \\
  0, & \text{for } \tau \text{ far from } t 
\end{cases}
\]  

(3.14)

The STFT about time \( t \) is the Fourier transform of \( x_w(t) \) given as \( X_w(\omega) \), and the energy density spectrum about time \( t \) is the squared magnitude of \( X_w(\omega) \). Thus, the energy density spectrum about time \( t \) is given as

\[
P_{sp}(t, \omega) = |X_w(\omega)|^2 = \left| \frac{1}{\sqrt{2\pi}} \int e^{-j\omega \tau} x(\tau)h(\tau - t) d\tau \right|^2
\]  

(3.15)

For each time \( t \), a different spectrum is obtained. These spectra can be combined to form the time–frequency distribution \( P_{SP} \) referred to as the spectrogram.
3.5 FREQUENCY DOMAIN ANALYSIS

The parameters or measures computed from the frequency domain signals are called ‘frequency domain features’. Frequency analysis of gear box is generally required for fault diagnosis of gear box. A number of frequency domain features are in use and most popular ones are discussed in following sub-sections.

3.5.1 Enveloping

Enveloping is used to monitor the high-frequency response of the mechanical system to periodic impacts of gear faults. An impulse is produced each time a faulty gear tooth makes contact with another tooth. This impulse has an extremely short duration compared to the interval between the pulses. The energy from the defect pulse will be distributed at a very low level over a wide range of frequencies. It is this wide distribution of energy that makes gear defects so difficult to detect by conventional spectrum analysis when they are in the presence of vibrations from other machine components. Fortunately, the impact usually excites a resonance in the system at a much higher frequency than the vibration generated by the other components. This structural energy is usually concentrated into a narrow band that is easier to detect than the widely distributed energy of the gear defect frequencies. With tooth wear and breakage, the side band activity near critical frequencies such as the output shaft frequency is expected to increase. The entire spectrum contains very high periodic signals associated with the gear mesh frequencies. The envelope or high frequency technique focuses on the structure resonance to determine the health of a gear or the type of failure in a gear. This technique consists of processing structure resonance energy with an envelope detector. The structure resonance is obtained by band-pass filtering the data around the structure resonance frequency. The band-pass filtered signal is then processed by an envelope detector, which consists of a half-wave (or full-wave) rectifier and a peak-hold and smoothing section. The center frequency of the band-pass filter should be selected to coincide with the structure resonance frequency being studied. The bandwidth of the filter should be at least double the highest characteristic defect frequency. This will ensure that the filter will pass the carrier frequency and at least one pair of modulation
sidebands. In practice, the bandwidth should be somewhat greater to accommodate the first two pairs of modulation sidebands around the carrier frequency. The rectifier in the envelope detector turns the bipolar filtered signal into a unipolar waveform. The peak-hold smoothing section will then remove the carrier frequency by smoothing/filtering the fast transitions in the signal. The remaining signal will then consist of the defect frequencies. This feature produces several figures of merit for analysis use. The primary figure of merit is the peak frequency and amplitude in the power spectral density of the enveloped data.

3.5.2 Demodulation

During a normal gear roll, one tooth is essentially pushing another without sliding. When the teeth wear, sliding occurs. The energy that went into pushing before will now go into pushing and sliding, thus resulting in a change of amplitude or amplitude modulation of the vibrations at the gear mesh frequency (GMF) and its harmonics. Demodulation processing identifies periodicity in modulation of the carrier. The carriers used in this processing were the GMF and 2*GMF. Demodulation techniques detect the amplitude modulation components induced by gear wear in the region of a single frequency, in this case the GMF or 2* GMF. This differs from enveloping which detects the combined effects over a range of frequencies. To implement the demodulation technique, the raw data is high-passed filtered at 85%*GMF and then low-passed filtered at 115%*GMF. The power spectral density of the filtered signal is searched to obtain the actual carrier frequency (GMF). The actual carrier is used to amplitude demodulate the filtered carrier signal. The power spectral density of the resulting signal is searched within +/- five percent of the output shaft frequency. The figures of merit extracted for this technique are the frequency of the peak and the magnitude squared amplitude.

3.5.3 NB4

NB4 is similar to NA4 except that instead of using the residual signal, NB4 uses the envelope of a band-passed segment of the time synchronous averaged signal. The idea behind this method is that a few damaged gear teeth will cause transient load fluctuations that are different from the normal tooth load fluctuations. The theory suggests that these
fluctuations will be manifested in the envelope of a signal which is band-pass filtered about the dominant meshing frequency. The dominant meshing frequency is either the primary meshing frequency or one of its harmonics whichever appears to give the most robust group of sidebands. Some suggest that the width of the band-pass filter depends on the location of the meshing frequency to other meshing frequency harmonics, while others suggest using a bandwidth giving the maximum amount of sidebands even if the sidebands interfere with those from other harmonics. The reasoning of the latter method is to assume that the interference from other sidebands is negligible and includes as many of the primary modulating sidebands as plausible. The envelope of the band-passed signal is the magnitude of the complex (i.e., analytic) signal obtained by applying the Hilbert transform to the band-passed signal:

\[ E(t) = \sqrt{(A(t))^2 + H[A(t)]^2} \]  \hspace{1cm} (3.16)

where \( E(t) \) is the envelope of the band-passed signal, \( A(t) \) is the band-passed signal, and \( H[A(t)] \) is the Hilbert transform of the band-passed signal. The analytic signal is \( A(t) + iH[A(t)] \).

### 3.6 WAVELET ANALYSIS

The advent of wavelet analysis in the mid 1980s provided a new and powerful tool for the analysis of the joint time–frequency characteristics of transmission vibration signals. The wavelet transform is a time–frequency analysis method similar to the WVD, though it is independent of time and thus can truly describe local behavior. Although wavelet analysis falls under the general heading of time–frequency analysis, it merits an independent section due to the large volume of research into its use as a diagnostic technique. Unlike the Fourier transform, where stationary complex basis functions are used to map the temporal signal into the frequency domain, the wavelet transform uses a new class of real and complex nonstationary basis functions, termed wavelets, which can be independently dilated and shifted as a function of time, to create a unique time–frequency map. The advantage of this method is that the frequency content of the signal can be analyzed without the loss of vital time-domain information. In addition, time and
frequency are independent. Thus, the time–frequency representation is not affected by cross-terms.

Wavelet means ‘small wave’. Short duration finite energy functions are called as wavelets. By definition it is a small wave, hence, it should be time limited. Finite energy functions are those which will represent a phenomenon like a spring oscillation due to an initial stimulus. It exists only for a short duration and releases finite amount of energy.

Wavelet Transform is defined as the integral of the signal $s(t)$ multiplied by scaled, shifted versions of a basic wavelet function $\psi(t)$—a real-valued function whose Fourier Transform satisfies the admissibility criteria:

$$C(a,b) = \int_{\mathbb{R}} s(t) \frac{1}{\sqrt{a}} \psi\left(\frac{t-b}{a}\right) dt$$

$$a \in \mathbb{R}^+ - \{0\}, b \in \mathbb{R}.$$  \hspace{1cm} (3.17)

where,

$C$ - Continuous wavelet coefficients

$\psi$ - time

$a$ - scaling parameter,

$b$ - time localisation parameter.

Both ‘$a$’ and ‘$b$’ can be continuous or discrete variables. Multiplication of each coefficient by an appropriately scaled and shifted wavelet yields the constituent wavelets of the original signal. For signals of finite energy, continuous wavelet synthesis provides the reconstruction formula:

$$s(t) = \frac{1}{K_{\psi}} \int_{\mathbb{R}} \int_{\mathbb{R}} C(a,b) \frac{1}{\sqrt{a}} \psi\left(\frac{t-b}{a}\right) da \, db$$

$$\hspace{1cm} (3.18)$$

Use of Discrete Wavelet Transform (DWT) for analysis retains effectiveness without sacrificing accuracy. In this scheme, ‘$a$’ and ‘$b$’ are given by:

$$(j,k) \in \mathbb{Z}^2 : a = 2^j, b = k2^j, \mathbb{Z} = \{0, \pm 1, \pm 2\}$$

Defining

$$(j,k) \in \mathbb{Z}^2 : \psi_{j,k}(t) = 2^{-j/2} \psi\left(2^{-j} t - k\right)$$
\[ \phi_{j,k}(t) = 2^{-j/2} \phi(2^{-j} t - k) \]  

(3.19)

A wavelet filter with impulse \( g \), plays the role of the wavelet \( \psi \) and a scaling filter with impulse response \( h \), plays the role of scaling function \( \phi \). \( g \) and \( h \) are defined on a regular grid \( \Delta \mathbb{Z} \), where \( \Delta \) is the sampling period (here, without loss of generality, set \( \Delta = 1 \)). Discrete wavelet analysis yields

\[ C(a,b) = c(j,k) = \sum_{n \in \mathbb{Z}} s(n) g_{j,k}(n) \]

\[ a = 2^j, b = k2^j, j \in \mathbb{N}, k \in \mathbb{N} \]  

(3.20)

Corresponding synthesis relation is:

\[ s(t) = \sum_{j \in \mathbb{Z}} \sum_{k \in \mathbb{Z}} c(j,k) \psi_{j,k}(t) \]  

(3.21)

The detail at level \( j \) is defined as:

\[ D_j(t) = \sum_{k \in \mathbb{Z}} c(j,k) \psi_{j,k}(t) \]  

(3.22)

And the approximation to \( s(t) \) at any level is:

\[ A_{j-1} = \sum_{j > j} D_j \]  

(3.23)

The following equations hold:

\[ A_{j-1} = A_j + D_j \]  

(3.24)

\[ s = A_j + \sum_{j > j} D_j \]  

(3.25)

In practice, the decomposition can be determined iteratively, with successive approximations being computed in turn, so that a signal is decomposed into many lower-resolution components. This is known as the wavelet decomposition tree.

The discrete wavelet transform (DWT) provides an effective method for generating features. The collection of all such features forms the feature vector. A feature vector is given by

\[ v_{dwt} = \left\{ v_{1 \text{\_dwt}}, v_{2 \text{\_dwt}}, \ldots, v_{12 \text{\_dwt}} \right\}^T \]  

(3.26)
A component $v_i^{dwt}$ in the feature vector is related to the individual resolutions by the following equation

$$v_i^{dwt} = \frac{1}{n_i} \sum_{j=1}^{n_i} w_{i,j}^2, \quad i = 1,2,...12$$

(3.27)

where, $n_i = 2^{12}$, $n_2 = 2^{11}$, ..., $n_{12} = 2^0$.

$v_i^{dwt}$ is the $i^{th}$ feature element in a DWT feature vector. $n_i$ is the number of samples in an individual sub-band, $w_{i,j}^2$ is the $j^{th}$ coefficient of the $i^{th}$ sub-band. Research into the use of the wavelet transform for transmission diagnostics has focused primarily on the assessment of the performance of various basis functions and various forms of the transform. Typically, a set of vibration signals is analyzed by the technique under consideration and the resulting distribution is visually inspected to determine whether changes due to damage are evident. However, research is beginning to focus on more intelligent applications of wavelets in an attempt to exploit the flexibility offered by the transform.

### 3.7 MACHINE LEARNING APPROACH

Machine fault diagnosis is a procedure of mapping the information obtained in the measurement space and/or features in the feature space to machine faults in the faults space. The mapping process is also called **machine learning or pattern recognition**. Traditionally, pattern recognition is done manually with auxiliary graphical tools such as power spectrum graph, phase spectrum graph, cepstrum graph, AR spectrum graph, spectrogram, wavelet scalogram, wavelet phase graph, etc. However, manual pattern recognition requires expertise in the specific area of the diagnostic application. Thus, highly trained and skilled personnel are needed. Therefore, automatic pattern recognition is highly desirable. This can be achieved by classification of signals based on the information features extracted from the signals using machine learning techniques. In the following sub-sections, different machine learning techniques are discussed.
3.7.1 Introduction to Machine Learning

Machine learning is the mechanisation of ‘learning’ and ‘labelling’. Learning is the process of acquiring knowledge of or skill in a domain through study or by being taught. Labelling is the process of identifying the object as a member of a class to which it belongs. Class means a category, or a condition, or a group of items which shares a common name. For example, rose, jasmine, lotus belong to a class of ‘flowers’ (Shares a common name - flower), similarly, apple, banana, mango, grapes belong to a class of ‘fruits’. The definition of a class is context dependent; one has to define a class based on a specific problem. Fig 3.1 shows the flow chart about machine learning approach.

Machine learning has three important phases; feature extraction, feature selection and feature classification. They are discussed in the following sub-sections.

3.7.2 Feature extraction

Feature extraction constitutes computation of specific measures which characterise the signal. Vibration signals were taken from the gearbox at various conditions and the
features were extracted from the signals which form the input to the classifier. Classifier is an algorithm which maps the points from input space to the fault space. In fault space, there are four regions corresponding to fault conditions. The input is made up of 8192 samples of digitized vibration signal. Most of the classifiers will find it difficult to manage such a large number of variables. To reduce the number of inputs, certain measures of the input signal characterizing the conditions, are used. These measures are called as features. The process of computing such measures is called as ‘feature extraction’. The features are application dependent and one has to choose a set of good features for better classification on case by case basis. This process of choosing the prominent features from the given set of features is known as the feature selection.

3.7.2.1 Time Domain Features

The parameters or measures computed directly from the acquired time domain signals are called ‘time domain features’. The following time domain features were widely used in fault diagnosis of machine components:

a) Statistical features

b) Histogram features

3.7.2.1.1 Statistical Features

The role of Statistical analysis of vibration signals is explained in depth in section 3.3.1.2.

3.7.2.1.2 Histogram features

Observing the magnitude of the time domain signal, it is found that the range of vibration amplitude varies from class to class. A better graph to show the range variation is the histogram plot. The information derived from a histogram plot can be used as features in the fault diagnosis. A representative sample from each gear box condition (class) is taken and the histogram is plotted. The selection of bin involves two criteria.

(a) The bin range should accommodate the amplitude range of signals obtained from all conditions of gear box being analyzed. The vibration amplitude range is divided into
set of sub ranges starting from minimum value to maximum value of the vibration signal. The sub-ranges are called bins and form the x-axis of the histogram. The number of data points whose vibration amplitude value falls in a particular bin is counted and the count forms the y-axis of the histogram. The objective here is to find out the bins whose y-axis values are same for a particular class and different from other classes. To do this, one needs to take into account the minimum and maximum amplitude of different classes while deciding the bin range. These values for a particular condition of the gearbox may be small and for another condition it may be large. The bin range selected should accommodate all conditions. i.e., the bin range should start from the minimum of minimum amplitude of different classes and go up to maximum of maximum amplitude of different classes.

(b) The width of the bin should be fixed such that the height of bins is different for different condition of the gear box. It need not be true for all width of bins, but at least few of them should follow this criterion so that it can be used as feature for distinguishing various conditions (classes). Bin width is set of boundary values that should be in ascending order so that the program counts the number of data points between the current bin number and the adjoining higher bin, if any. The bin width need not be constant always. However, in this study a constant bin width is used.

Following the above criteria, the bin width and bin range were selected and histogram was plotted. As it is seen from Fig. 3.2, there are significant differences in some of the feature values (bin ranges) for different types of faults. Selecting those features is crucial for effective classification and doing it manually demands more expertise; however, the effectiveness of the features is not guaranteed. Selecting the good features through suitable algorithm will yield better classification results. A popular method for doing this
is Principle Component Analysis (PCA) which is discussed in section 3.7.3.1.

3.7.3 Feature selection

The features can be any measure of data points or the signal; but the relevance of them will depend on how well they help in the process of classification. The process of selecting the good features from a pool of features is called ‘feature selection’. The good feature will have feature values with minimum variation within a class and maximum
variation between the classes. Many techniques are used for feature selection; among them Principle Component Analysis (PCA) is widely used. Decision tree also can be used for this purpose.

### 3.7.3.1 Principal Component Analysis (for features selection)

Given a set of n dimension feature vectors $x_i$ ($i = 1, 2 \ldots m$), generally nom,

Let

$$\mu = \frac{1}{m} \sum_{t=1}^{m} x_t$$ ..................................................(3.28)

Then, the covariance matrix of feature vectors is

$$C = \frac{1}{m} \sum_{t=1}^{m} (x_t - \mu)(x_t - \mu)^T$$ ..................................................(3.29)

The principal components (PCs) are computed by solving the eigen value problem of covariance matrix $C$,

$$C_{vi} = \lambda_i v_i$$ ..................................................(3.30)

where $\lambda_i$ ($i=1,2,\ldots n$) are the eigen values and they are sorted in descending order, $v_i$ ($i = 1,2,\ldots n$) are the corresponding eigenvectors.

To represent the raw feature vectors with low-dimensional ones, what needs to be done is to compute the first $k$ eigenvectors ($k \leq n$) which correspond to the $k$ largest eigen values. In order to select the number $k$, a threshold $\gamma$ is introduced to denote the approximation precision of the $k$ largest eigenvectors.

$$\frac{\sum_{i=1}^{k} \lambda_i}{\sum_{i=1}^{n} \lambda_i} \geq \theta$$ ..................................................(3.31)

Given the precision parameter $\gamma$, the number of eigenvectors $k$ can be decided.
\begin{align}
V = [v_1, v_2, ..., v_k], \quad \Lambda = \text{diag}[\lambda_1, \lambda_2, ..., \lambda_k]
\end{align}  \tag{3.32}

After the matrix \( V \) is decided, the low-dimensional feature vectors, named PC, of raw ones are determined as follows:

\begin{align}
P = V^T x_i
\end{align}  \tag{3.33}

The PCs of PCA have three properties [125]

(1) They are uncorrelated;

(2) They have sequentially maximum variances;

(3) The mean-squared approximation error in the representation of the original vectors by the first several PCs is minimal.

3.7.3.2 Decision Trees

A Decision Tree is a tree based knowledge representation methodology used to represent classification rules. A standard tree induced with c5.0 (or possibly ID3 or c4.5) consists of a number of branches, one root, a number of nodes and a number of leaves. One branch is a chain of nodes from root to a leaf; and each node involves one attribute. The occurrence of an attribute in a tree provides the information about the importance of the associated attribute [126]. The procedure of forming the Decision Tree and exploiting the same for feature selection is characterized by the following.

1. The set of features available at hand forms the input to the algorithm; the output is the Decision Tree.

2. The Decision Tree has leaf nodes, which represent class labels, and other nodes associated with the classes being classified.

3. The branches of the tree represent each possible value of the feature node from which they originate.
4. The Decision Tree can be used to classify feature vectors by starting at the root of the tree and moving through it until a leaf node, which provides a classification of the instance, is identified.

5. At each decision node in the Decision Tree, one can select the most useful feature for classification using appropriate estimation criteria. The criterion used to identify the best feature invokes the concepts of entropy reduction and information gain – discussed in the following sub section. A sample of the of the decision tree is shown in Fig. 3.3

### 3.7.3.2.1 Information gain and Entropy Reduction

Information gain measures how well a given attribute separates the training examples according to their target classification. The measure is used to select among the candidate features at each step while growing the tree. Information gain is the expected reduction in entropy caused by portioning the samples according to this feature.

Information gain \((S, A)\) of a feature \(A\) relative to a collection of examples \(S\), is defined as:

\[
Gain(S, A) = Entropy(S) - \sum_{v \in Values(A)} \frac{|S_v|}{|S|} Entropy(S_v) \tag{126}
\]

where, \(Values(A)\) is the set of all possible values for attribute \(A\),

\(S_v\) is the subset of \(S\) for which feature \(A\) has value \(v\) (i.e., \(S_v = \{s \in S \mid A(s) = v\}\)).

Note the first term in the equation for \(Gain\) is just the entropy of the original collection \(S\) and the second term is the expected value of the entropy after \(S\) is partitioned using feature \(A\). The expected entropy described by the second term is simply the sum of the entropies of each subset \(S_v\), weighed by the fraction of samples \(|S_v|/|S|\) that belong to \(S_v\). \(Gain(S,A)\) is therefore the expected reduction in entropy caused by knowing the value of feature \(A\). Entropy is a measure of homogeneity of the set of examples and it is given by
\[ \text{Entropy}(S) = \sum_{i=1}^{c} - P_i \log_2 P_i \]  

(3.34)

Where,

- \( c \) is the number of classes
- \( p_i \) is the proportion of \( S \) belonging to class ‘\( i \)’.

### 3.8 MACHINE LEARNING METHODS

Classifier is as defined earlier, a function which maps a set of inputs from feature space to its corresponding classes. In the present study, from set of extracted features the classifier maps them to the condition of the gear box. There are many classifiers in use for pattern classification. Amongst them artificial neural network, fuzzy logic, support vector machines and proximal support vector machines are very popular ones. The following sections describe briefly about the commonly used classifiers.
3.8.1 Artificial Neural Networks

Artificial neural networks (ANNs) are formed of cells simulating the low level functions of biological neurons. In ANN, knowledge about the problem is distributed in neurons and connection weights of links between neurons. The neural network has to be trained to adjust the connection weights and biases in order to produce the desired mapping. At the training stage, the feature vectors are applied as input to the network and the network adjusts its variable parameters, the weights and biases, to capture the relationship between the input patterns and outputs. ANNs are particularly useful for complex pattern recognition and classification tasks. The capability of learning from examples, the ability to reproduce arbitrary nonlinear functions of input, and the highly parallel and regular structure of ANN make them especially suitable for pattern classification tasks. ANN’s are characterized by their topology, weight vector and activation functions. They have three layers namely an input layer, which receives signals from the external world, a hidden layer, which does the processing of the signals and an output layer, which gives the result back to the external world. Various neural network structures are available. The reviews of literature reveals that both supervised learning and unsupervised learning have been applied in similar problems.

3.8.1.1 Multi-Layer Perceptron (MLP)

This is an important class of neural networks, namely the feed forward networks. Typically, the network consists of a set of input parameters that constitute the input layer, one or more hidden layers of computation nodes and an output layer of computation.
nodes (Fig.3.4). The input signal propagates through the network in a forward direction on a layer-by-layer basis.

MLPs have been applied to solve some difficult and diverse problems by training them in a supervised manner with a highly popular algorithm known as the error back-propagation algorithm. Each neuron in the hidden and output layer consists of an activation function, which is generally a non-linear function like the logistic function, which is given by

\[
f(x) = \frac{1}{1 + e^{-x}},
\]

(3.35)

Where \( f(x) \) is differentiable and

\[
x = \sum_{i=1}^{I} W_{ij} \xi_i + \theta_j
\]

(3.36)

Where, \( W_{ij} \) is the weight vector connecting the \( i^{th} \) neuron of the input layer to the \( j^{th} \) neuron of the hidden layer, \( \xi_i \) is the input vector and \( \theta_j \) is the threshold of the \( j^{th} \) neuron of the hidden layer. Similarly, \( W_{jk} \) is the weight vector connecting \( j^{th} \) neuron of the hidden layer with the \( k^{th} \) neuron of the output layer. \( i \) – represents the input layer, \( j \) - represents the hidden layer and \( k \)-represents the output layer. The weights that are important in predicting the process are unknown. The weights of the network to be trained are initialized to small random values. The choice of value selected obviously affects the rate of convergence. The weights are updated through an iterative learning process known as ‘Error Back Propagation (BP) algorithm’. Error Back Propagation process consists of two passes through the different layers of the network; a forward pass in which input patterns are presented to the input layer of the network and its effect propagates through the network layer by layer. Finally, a set of outputs is produced as the actual response of the network. During the forward pass the synaptic weights if the networks are all fixed. The error value is then calculated, which is the mean square error (MSE) given by

\[
E_{tot} = \frac{1}{n} \sum_{n=1}^{n} E_n
\]

(3.37)
\[ E_n = \frac{1}{2} \sum_{k=1}^{m} (\xi_k^n - O_k^n)^2 \]

Where,

\[ \xi_k^n \] is the \( k \)th component of the desired or target output vector and

\[ O_k^n \] is the \( k \)th component of the output vector.

The weights in the links connecting the output and the hidden layer \( W_{jk} \) are modified as follows:

\[ \Delta W_{jk} = \eta (-\partial E / \partial W_{jk}) = \eta \delta_j y_j, \]

where \( \eta \) is the learning rate. Considering the momentum term \( \alpha \)

\[ \Delta W_{jk} = \alpha \eta \delta_j y_j \] and \( W_{jk}^{new} = W_{jk}^{old} + \Delta W_{jk} \).

Similarly the weights in the links connecting the hidden and input layer \( W_{ij} \) are modified as follows:

\[ \Delta W_{ij} = \alpha \eta \delta_j \delta_i, \]

\[ \delta_j = y_j (1 - y_j) \sum_{k=1}^{m} \delta_k W_{jk} \]

Where,

\[ W_{ij}^{new} = W_{ij}^{old} + \Delta W_{ij} \]

\[ \delta_k = (\xi_k^n - O_k^n) O_k (1 - O_k) \] for output neurons and

\[ \delta_j = y_j (1 - y_j) \sum_{k=1}^{m} \delta_k W_{jk} \] for hidden neurons.

The training process is carried out until the total error reaches an acceptable level (threshold). If \( E_{tot} < E_{min} \) the training process is stopped and the final weights are stored, which is used in the testing phase for determining the performance of the developed network. The training mode adopted was ‘batch mode’, where weight updating was performed after the presentation of all training examples that constitutes an epoch.
### 3.8.2 Fuzzy Logic

Fuzzy Logic provides a precise approach for dealing with uncertainty. Fuzzy inference is a method that interprets the values in the input vector and, based on a set of rules, assigns values to the output vector. The point of fuzzy logic is to map an input space to an output space, and the primary mechanism for doing this is a list of ‘if-then’ statements called rules. Rules are the inputs for building a fuzzy inference engine. All rules are evaluated in parallel, and the order of the rules is unimportant. The real world data do not have sharply defined boundaries where information is often incomplete or sometimes unreliable. In quest for precision, scientists have generally attempted to manipulate the real world into artificial mathematical models that make no provision for gradation. Because Fuzzy Logic provides the tools to classify information into broad, coarse categorizations or groupings, it has infinite possibilities for application which have proven to be much cheaper, simpler and more effective than other systems in handling complex information [127]. A set of ‘if-then’ rules defined using membership functions form the knowledge base for the fuzzy inference engine.

A membership function (MF) is a curve that defines how each point in the input space is mapped to a membership value (or degree of membership) between 0 and 1. The range of values of a feature (e.g. standard error, stderr) is represented using a set of membership functions. Here, for the purpose of example, a feature called standard error is represented using a membership functions namely stderr as shown in Fig. 3.5. Then using these membership functions, ‘if-then’ rules are formed. For example, If (Stderr is not stderr) then (Output1 is GTC). These set of ‘if-then’ rules form the knowledge base of the fuzzy inference engine.
Using this knowledge base, fuzzy inference engine performs classification.

3.8.3 Support Vector Machines (SVM)

The SVM is a new generation learning system based on statistical learning theory. SVM belongs to the class of supervised learning algorithms in which the learning machine is given a set of features (or inputs) with the associated labels (or output values). Each of these features can be looked upon as a dimension of a hyper-plane. SVMs construct a hyper-plane that separates the hyper-space into two classes (this can be extended to multi-class problems as in section (3.7.6). While doing so, SVM algorithm tries to achieve maximum separation between the classes (See Fig. 3.6). Separating the classes with a large margin minimizes the expected generalization error. By ‘minimum generalization error’, we mean that when a new set of features (that is data points with unknown class values) arrive for classification, the chance of making an error in the prediction (of the class to which it belongs) based on the learned classifier (hyper-plane) should be minimum. Intuitively, such a classifier is one, which achieves maximum separation-margin between the classes. The above process of maximizing separation leads to two hyper-planes parallel to the separating plane, on either side of it. These two can have one or more points on them. The planes are known as ‘bounding planes’ and the distance between them is called as ‘margin’. By SVM ‘learning’, we mean, finding a hyper-plane, which maximizes the margin and minimizes the misclassification error. The points lying beyond the bounding planes are called support vectors. The data points $P_1,$

![Membership function for “Standard Error “](image-url)
$P2, P3, P4,$ and $P5$ belonging to A- are support vectors (See Fig. 3.6), but $P6, P7$ are not. Same facts hold good for class A+. These points play a crucial role in the theory and hence the name Support Vector Machines. Here, by ‘machine’, we mean an algorithm.

![Separating Plane](image)

**Fig. 3.6. Standard SVM Classifier**

The notations used by Fung [128] have been followed. In the formulation, ‘$A$’ is a $m \times n$ matrix whose elements belong to real space, ‘$D$’ is $m \times 1$ matrix representing class label (+1 and –1), ‘$e$’ is a vector of ones and ‘$\nu$’ is a control parameter that defines the weight of error minimization and bounding plane separation in the objective function. ‘$w$’ is orientation parameter and ‘$\gamma$’ is location parameter (location relative to origin) of separating hyper plane.

$$\min_{\{w, \gamma, y\} \in \mathbb{R}^{n+1+m}} \left\{ w^T e + \frac{1}{2} w^T w \right\}$$

$$\text{s.t.} \ D(Aw - e\gamma) + y \geq e$$

(3.42)
\[ y \geq 0 \quad \text{Where,} \quad A \in \mathbb{R}^{m \times n}, D \in \{-1, +1\}^{m \times 1}, e = 1^{m \times 1} \]

Vapnik [97] has shown that if the training features are separated without errors by an optimal hyper-plane, the expected error rate on a test sample is bounded by the ratio of the expectation of the support vectors to the number of training vectors. The smaller the size of the support vector set, more general the above result. Further, the generalization is independent of the dimension of the problem. In case such a hyper-plane is not possible, the next best is to minimize the number of misclassifications whilst maximizing the margin with respect to the correctly classified features.

### 3.8.4 Proximal Support Vector Machine (PSVM)

Recently a much simpler classifier, proximal support vector machine [129], was implemented wherein each class of points is assigned to the closer of two parallel planes (in input or feature space) that are pushed apart as far as possible. This formulation leads to a fast and simple algorithm for generating a classifier - (linear or nonlinear) that is obtained by solving a single system of linear equations.

PSVM is a modified version of Support Vector Machine (SVM). The point of departure from SVM is that, the optimization problem given by equation (3.42) is replaced by the following problem:

\[
\min_{(w, \gamma, y)} \frac{1}{2} \|y\|^2 + \frac{1}{2} (w^T w + \gamma^2)
\]

Subject to \[ D(Aw - e\gamma) + y = e \]  \hspace{1cm} (3.43)
Referring to Fig. 3.7, $y$ represents deviation (scaled by $1/\|w\|$) of the point from the plane passing through the centroid of the data cluster (A+ or A−) to which the point belongs. Hence, there is no non-negativity constraint on $y$. Further the 2-norm of the error vector $y$ is minimized instead of the 1-norm; the margin between the bounding planes is maximized with respect to both orientation $w$ and relative location $\gamma$ to the origin.

Extensive computational experience, indicates that the formulation (3.43) is almost as good as the classical formulation (3.42) with some added advantages such as strong convexity of the objective function. The key idea in this formulation is to make computation simple, by replacing the inequality constraint by equality. The modification, even though simple, changes the nature of optimization problem significantly. An explicit exact solution can be written to the problem in terms of the problem data. It is impossible to do that in the previous formulations because of their combinatorial nature.

Geometrically the formulation obtained by (3.43) can be interpreted as follows. The planes $x^Tw - \gamma = \pm 1$ are not bounding planes anymore, but can be thought of as “proximal” planes, around which the points of each class are clustered and which are pushed as far apart as possible by the term $w^Tw + \gamma^2$ in the objective function; in fact this term is the reciprocal of the 2-norm distance squared between the two planes in the
\((w, \gamma)\) space. The interpretation, however, is not based on the idea of maximizing the margin, the distance between the bounding parallel planes, which is a key feature of support vector machines.

After training, for any the new set of features prediction of its class is possible using the decision function as given below which is a function of ‘w’ and ‘\(\gamma\)’. It is called testing.

\[
f(x) = \text{sign}(w^Tx - \gamma)
\] (3.44)

If the value of \(f(x)\) is positive then new set of features belongs to class A+ otherwise it belongs to class A-. Classifying multiple classes is commonly performed by combining several binary SVM classifiers in a tournament manner, either one-against-all or one-against-one, the latter approach required substantially more computational effort.

### 3.8.5 Multi-Class Classification from One Class SVM

Multi-class SVM [130-131] classification, which will classify multiple classes in single shot, is of great use. In this section, the method of using one class support vector machine to do the multi-class classification is described. Each class is considered independently for training. The one class methodology is applied to each of the classes separately. Thus the multi-class problem is effectively reduced to multiple one-class problems. Each class is learnt separately in the following manner:

1. The Gaussian Kernel for each class is computed. The Gaussian Kernel basically gives the similarity measure between a set of points on a scale of 0 to 1.
2. The system of equations \(Qu = e\) is solved. Where \(Q\) is the kernel matrix, \(u\) is the vector of unknowns and \(e\) is a vector of ones.
3. The solution vector \(u\) is the set of Lagrangian multipliers for the particular class. These Lagrangian multipliers would be positive, negative or zero depending on the constraints. Only the positive values are to be considered and from the input data, the points corresponding to the indices of the positive values are extracted.
These points are the support vectors. The support vectors are the points, which lie on the hyper-sphere. These points contain all the information related to that class.

4. This process is repeated for all the classes and thus the representative points for each class are obtained. This completes the training process.

Depending upon the $\gamma$ that is used in the algorithm, the size of the hyper-sphere would shrink or grow, thus leaving out or enclosing more data. This has a direct effect on the number of support vectors that are obtained. In essence, Training leaves us with those data points that are absolutely essential for future classifications.

In the testing phase the accuracy of the classifier is determined. A set of supervised data points are selected which have not been used in the training process. The Kernel is computed with the test points and the support vectors of each class. The kernel matrices represent the distance relationship of the test points from the support vectors of each class.

Next, a decision matrix is formed, the columns of which are the row wise summed up kernel matrices. The maximum element of each row is computed and its index is extracted. This index corresponds to the class label to which the particular point belongs. In the decision matrix, the number of columns represents the number of classes and number of rows represents the number of points. The individual kernel matrices that are previously computed have been summed up row wise to obtain a vector. Thus the vectors are now become the columns of the decision matrix. The first row thus has a kind of score of the first data point with respect to each of the classes. The highest score wins.

This is compared with the input labels and the accuracy is determined. The training is repeated with a different $\gamma$ value if the accuracy is not good. Thus it makes use of Gaussian kernel to find the neighborhood score to perform multi-class classification and hence, the method is called kernel based neighborhood score MSVM.

3.9 SUMMARY

In this chapter, commonly used features along with the ones that are used for this study are discussed. Two feature selection techniques and commonly used classifiers for fault
diagnosis have been described briefly. The experimental set up and experimental procedures are discussed in the next chapter.