CLASSIFICATION OF BEARING FAULTS USING EXTREME LEARNING MACHINE ALGORITHMS

TEH CHOON KEONG

ELECTRICAL AND ELECTRONIC ENGINEERING

UNIVERSITI TEKNOLOGI PETRONAS

JANUARY 2017

CLASSIFICATION OF BEARING FAULTS USING EXTREME LEARNING MACHINE ALGORITHMS

By

Teh Choon Keong
18355

Dissertation submitted in partial fulfillment of

the requirement for the

Bachelor of Engineering (Hons)

(Electrical And Electronic)

JANUARY 2017

Universiti Teknologi PETRONAS

Bandar Seri Iskandar

32610 Seri Iskandar

Perak Darul Ridzuan

Malaysia

CERTIFICATION OF APPROVAL

CLASSIFICATION OF BEARING FAULTS USING EXTREME LEARNING MACHINE ALGORITHMS

By

Teh Choon Keong

18355

A project dissertation submitted to the

Electrical and Electronic Engineering Programme

Universiti Teknologi PETRONAS

in partial fulfillment of the requirement for the

BACHELOR OF ENGINEERING (HONS)

(ELECTRICAL AND ELECTRONIC)

Approved by,

____

(Dr. Azrina binti Abd Aziz)

UNIVERSITI TEKNOLOGI PETRONAS

BANDAR SERI ISKANDAR, PERAK

January 2017

CERTIFICATION OF ORIGINALITY

This is to certify that I am responsible for the work submitted in this project, that the original work is my own except as specified in the references and acknowledgements, and that the original work contained herein have not been undertaken or done by unspecified sources or persons.

Teh Choon Keong

ABSTRACT

Roller element bearing fault diagnosis is crucial for industry to maintain machine in good condition so that there is no delay of work due to machine breaks down. This project implements the bearing fault diagnosis that classifies the bearing data into four classes which are healthy bearing, inner race defect bearing, outer race defect bearing, and roller element defect bearing. Most of existing bearing fault diagnosis are done using Back Propagation (BP) algorithm which take a long time to train the neural network resulting in inefficiency of training the Single Hidden Layer Feedforward Neural Network (SLFN). Therefore, this project introduces three learning algorithms which are Extreme Learning Machine (ELM), Finite Impulse Response Extreme Learning Machine (FIR-ELM) and Discrete Fourier Transform Extreme Learning Machine (DFT-ELM) to improve the bearing fault diagnosis accuracy and shorten the time used to train and test the neural network. These learning algorithms perform significantly better than Back Propagation (BP) and take shorter time to train neural network compared to BP due to different computation method is used. In this project, there are four learning algorithms used to train the SLFN such as BP, ELM, FIR-ELM, and DFT-ELM. The performance comparison of each learning algorithm is evaluated in term of accuracy, error rate, sensitivity, and precision. The result shows DFT-ELM has the smallest training error rate and testing error rate as compared to FIR-ELM, ELM and BP while BP has the highest training and testing error rate. DFT-ELM, FIR-ELM, and ELM required lesser time to train the neural network while it also able to achieve higher accuracy than BP.

ACKNOWLEDGEMENT

First and foremost, the author would like to thank Dr. Azrina binti Abd Aziz and Associate Professor Dr Irraivan Elamvazuthi for supervising me for my Final Year Project. Furthermore, I appreciate all the guidance given from them on how to conduct a good research project and writing a report.

Last but not least, the author would like to appreciate the help from Center on Intelligent Maintenance System (IMS) University of Cincinnati, USA, for providing the bearing data set online.

TABLE OF CONTENT

List of Figures

List of Tables

Abbreviation

1.	ANN	Artificial Neural Network
2.	SLFN	Single Hidden Layer Feedforward Neural Network
3.	BP	Back Propagation
4.	ELM	Extreme Learning Machine
5.	FIR-ELM	Finite Impulse Response Extreme Learning Machine
6.	DTF-ELM	Discrete Fourier Transform Extreme Learning Machine
7.	EMD	Empirical Mode Decomposition
8.	IMF	Intrinsic Mode Functio n

Nomenclatures

c	Intrinsic mode function
SD	Standard deviation between two signals
R	Residual of the vibration signal
E	Energy of IMF
E_T	Total energy of all IMF
H_{en}	Energy Entropy
p	Normalized energy of IMF
μ_k	Mean of feature vector
θ^l	Output weight of BP
$\delta^{(l+1)}$	Error difference
α	Learning rate
H	Hidden layer output
β	Output weight
W	Input weight
T	Target output
H^+	Moore Penrose Pseudo inverse of Hidden layer output matrix
λ	Regularization Parameter
N	Number of hidden nodes
x	Input
O	Output of SLFN
$\overline{A}$	Transformation matrix

Y_d Desired response for hidden layer in DFT-ELM

I Identity matrix

CHAPTER 1
INTRODUCTION

1.1 Background

Bearing failure is one of the common reasons why machine breaks down. The function of bearing is to provide support and lubrication to the shaft so that the shaft can rotate in a smooth manner. However, if there is any failure in the bearing, it might affect the rotating shaft. Therefore, bearing diagnosis research is very crucial in industry. The conventional preventive and corrective method have been replaced by the diagnosis method such as deep learning, support vector machine, recurrent neural network, SLFN and so on. Artificial neural network (ANN) able to detect the fault before the machine break down and anticipation of problem in time, so that there is no delay of work due to machine broke down. A fault diagnosis is considered as pattern recognition problem. A suitable features attraction and classifier method play a big role in obtaining higher diagnosis accuracy. Thus, investment on bearing fault diagnosis with the aim to diagnose bearing fault has been carried out.

The bearing used in this project is a roller element bearing as shown in Figure 4. Roller element bearing can be divided into four different parts which are outer race, inner race, roller element, and cage. There are few categories of bearing faults are considered in this bearing data set, such as outer race defects as shown in Figure 3, inner race defects as shown in Figure 1, and roller element defects as shown in Figure 2. On the other hand, healthy bearing is as shown in Figure 4. Bearing data collected based on vibration signal is complex, nonlinear and non-stationary. Hence, a suitable feature extraction method should be used to simplify the bearing data to improve the classification result of a classifier (ANN). ANN able to process information like human brains. This makes them a useful tool in solving problem like pattern recognition, which can be done by human brain. Therefore, it has high capabilities in determining the relationship in input data that cannot be done easily by common analytic techniques.

Figure 1: Inner race defects

Figure 2: Roller element defects

Figure 3: Outer race defects

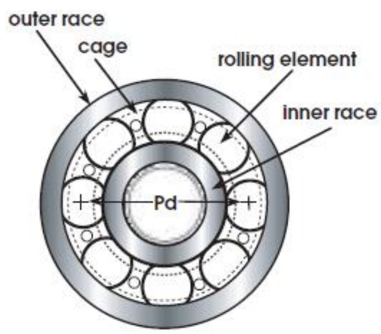


Figure 4: Elements in roller element bearing

Problem caused by BP and Solution to solve the problem

Based on previous work [1] and [13], the bearing fault diagnosis is done by using ANN trained by Back Propagation algorithm (BP) and the features extraction method is based on EMD. The limitation of using back propagation algorithm to train ANN is the training time is too long. The training time used by BP is very long due to the gradient descent method used by BP. Gradient descent method is used to optimize the input weight and output weight of the ANN. It takes a lot of iterations to optimize the input weight and output weight. In order to overcome this problem, Single Hidden Layer Feedforward Neural Network (SLFN) is used and trained by ELM, FIR-ELM and DFT-ELM. In this project, SLFN will be trained by four learning algorithms which are Back Propagation (BP), Extreme Learning Machine (ELM), Finite Impulse Response Extreme Learning Machine (FIR-ELM) and Discrete Fourier Transform Extreme Learning Machine (DFT-ELM). The author will prove that ELM, FIR-ELM and DFT-ELM can overcome the slow training speed of BP. Besides that, ELM, FIR-ELM, and DFT-ELM will perform significantly better and robust as compared to BP.

How SLFN is implemented?

Rolling element bearing (REBs) is commonly used in industrial and domestic application. REB is used in almost all rotating machinery and one of the main reasons of machine failure is the bearing failure. In this research, feature attraction will be done on the bearing data set to extract the important features and group it into a feature vector. Next, the feature vector will be used to train ANN on classifying the bearing fault and to compare with the desired output response. Next, a testing data (different from the training data) will go through feature extraction and then fed into the ANN for testing purposes. The output of the ANN will determine the accuracy of bearing fault diagnosis. There are few bearing faults to be classified by ANN, such as inner race defect, roller element defect, outer race failure, and healthy roller element bearing. To achieved an optimal result, a suitable feature extraction method and classifier must be chosen. Empirical Mode Decomposition (EMD) is used as a feature extraction method and SLFN is used as a classifier.

1.2 Problem Statement

In the bearing vibration data set collected by Centre on Intelligent Maintenance (IMS) University of Cincinnati, USA is very complex. Classification result of SLFN can be affected by the complexity, nonlinearity, and non-stationary characteristics of bearing vibration data if it is not properly processed. Therefore, a suitable feature extraction method must be used to extract the important features from the bearing vibration data set to improve the performance of SLFN. Based on the existing bearing fault diagnosis using ANN, the accuracy of fault is low. In [1], BP takes a very long time to train a ANN to perform a classification problem which result in inefficient training of ANN. Therefore, this report will focus on how to improve the fault diagnosis accuracy, and training speed. Besides, a suitable feature extraction method will be used to produce a better diagnosis accuracy.

1.3 Objectives

The aims of this research are as below: -

1. To investigate four learning algorithms for classifying motor bearing faults
   - Back Propagation (BP)
   - Extreme Learning Machine (ELM)
   - Finite Impulse Response Extreme Learning Machine (FIR-ELM)
   - Discrete Fourier Transform Extreme Learning Machine (DFT-ELM)
2. To develop a suitable model based on the four learning algorithms (BP, ELM, FIR-ELM and DFT-ELM)
3. To compare the performance of learning algorithms

1.4 Scope of Study

In this research, Single Hidden Layer (SLFN) is the main focus. SLFN is trained by 4 learning algorithms which are BP, ELM, FIR-ELM and DFT-ELM. This research project is working on the bearing fault data which is collected based on vibration signal to train the SLFN. The designed SLFN must have the capability to diagnose the bearing fault by using the bearing vibration data. Besides, a suitable feature extraction method will be used to extract the important features from the vibration data due to its complexity. Last but not least, the diagnosis accuracy will be compared to the previous research result.

1.5 Dissertation Outline

The rest of the dissertation is written as the following:

Chapter 2 provides discussion on bearing fault diagnosis, data acquisition, feature extraction method, fundamental of artificial neural network, and learning algorithms used to train the SLFN. Besides that, a critical analysis is done based on the existing work.

Chapter 3 provides discussion on the step by step methods to train the SLFN to achieve the objective of this project and solve the problem stated in the problem statement section. The design of SLFN and parameter used to design SLFN architecture will be discussed as well. Besides that, data set management and arrangement for data set and arrangement for training and testing the SLFN are discussed as well.

Chapter 4 provides discussion on original vibration signal, Intrinsic Mode Function (IMFs), energy entropy of IMFs of healthy bearing, inner race defect bearing, roller element defect bearing and outer race defect bearing. On the other hand, classification result, error rate, mean error rate, standard deviation and accuracy of each learning algorithm are explained and analysed in detail. Next, performance comparison of each learning algorithm is analysed by varying the number of hidden nodes, and the value of regularization parameter. Last but not least, a graphical user interface for the SLFN is shared in this chapter, besides, the training error rate and testing error rate of four learning algorithms are being analysed based on 5 different data sets.

And, finally, Chapter 5 provides a conclusion on the result and the future works.

CHAPTER 2

LITERATURE REVIEW

In this chapter, bearing fault diagnosis, vibration data, feature extraction method, artificial neural network and learning algorithms will be explained in detail.

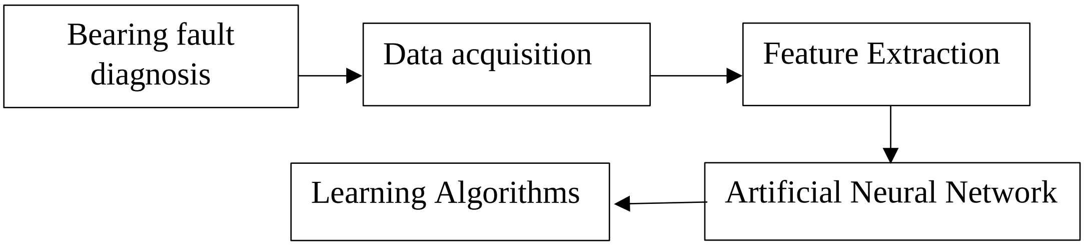


Figure 5: The flow of literature review

2.1 Bearing fault diagnosis

In past decade, bearing fault diagnosis in rotating machinery was done subjectively using a professional's experience or knowledge. They are able to distinguish the types of bearing fault based on their experience and knowledge about the vibration signal [12]. Vibration signal contains meaningful information about the bearing and at the same time the signal is very complex due to the noise in the vibration signal. Besides, bearing vibration signal is complex because it has nonlinear and non-stationary characteristics. The meaningful feature in the vibration signal must be extracted for SLFN to classify different types of bearing fault. In old times, bearing fault diagnosis is done by using artificial inference. The bearing fault diagnosis using artificial inference is done based on the frequency spectrum, because different bearing fault has different frequency spectrum distribution [12]. Since the vibration signal are complex, it is difficult for human to distinguish the type of bearing fault based on their experience and knowledge about the vibration data.

There are few traditional diagnosis methods are used to perform the bearing fault diagnosis. The traditional diagnosis methods are fuzzy inference and closeness method. However, the accuracy of this method can be influenced by a professional's experience. This is because this method requires human to design the value of the weighting parameter. Recently, a new method has been discovered, which is the Artificial Neural Network (ANN). ANN is normally applied in pattern classification and recognition. ANN is preferable to be applied in fault diagnosis because of its ability to automatically the vibration signal. Besides, the weight of the network will be modified automatically unlike the traditional methods. Therefore, the ANN is able to diagnose the fault better than the traditional method. In order to obtain high diagnosis accuracy, the ANN must be trained and tested. In training phase, the weights of the ANN will be changed automatically based on the difference between output of ANN and the targeted output. After successful training of the ANN, the trained ANN is then ready to be tested with a different kinds of vibration signal as an input to the trained ANN. In order to achieve a high diagnosis accuracy, the testing accuracy must be must be as high as possible.

2.2 Bearing Fault Diagnosis using ANN

2.2.1 Data acquisition

The bearing vibration data set used in this research was produced by the NSF I/UCR Center for Intelligent Maintenance Systems and Rexnord Corp. in Milwaukee [6]. During data acquisition, a test rig was setup as in Figure 6.

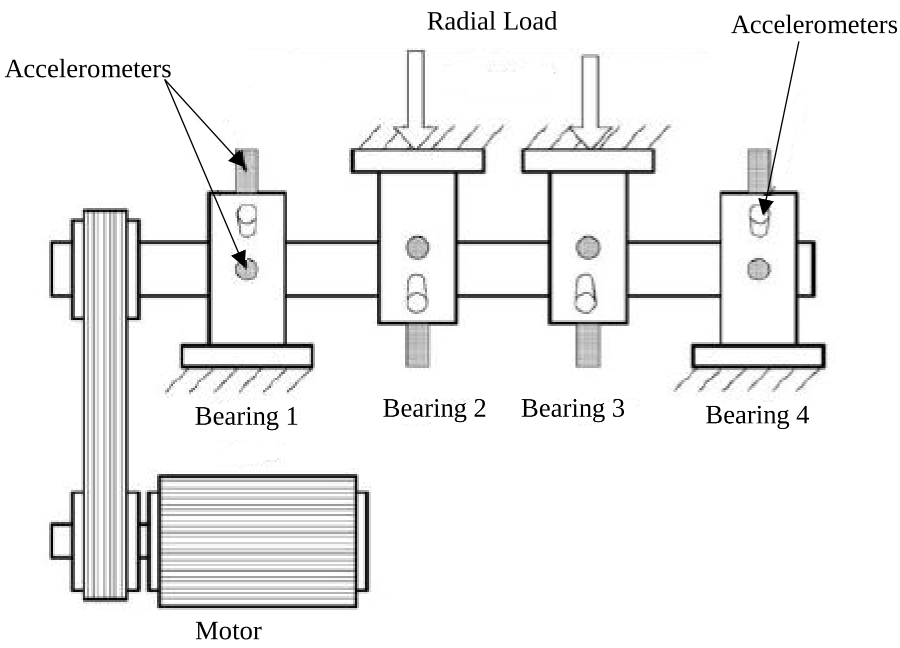


Figure 6: Bearing test rig

There are four bearings have been installed onto the shaft. The PCB 353B33 accelerometer was placed at the bearing 1, radial load of 6000 lbs was placed on bearing 2 and bearing 3 and lastly, a thermocouples was placed on the bearing 4. The shaft is rotated at a constant speed of 2000 rpm by using the motor. After exceeding the designed life time (100 million revolutions) for the bearing, the failure of the bearing occurred. This test took 35 days to observe the failure of bearing so that run to failure data set can be obtained with the presence of bearing defects.

2.2.2 Bearing Data set

In the data set provided by NSF I/UCR Center for Intelligent Maintenance System (IMS), there are 3 tests of bearing data. There are 4 bearings involve in each of the test. As mentioned, 3 sets of test data are collected in a run-to-failure experiment. So, each set consist of individual file that are 1-s vibration signal snapshots recorded at every 10 minutes [6]. There are 20480 points in each individual file with the sampling frequency of 20 kHz [6]. This vibration signal consists of a non-stationary and non-linear characteristic [1]. Furthermore, the noise present in the vibration signal bring a lot of trouble for a researcher to study this type of signal. Artificial bearing fault is created by the operator to collect the synthetic bearing data samples. In real life, corrosion, produced too much of heat due to the lack of lubrication, bearing misalignment and bearing mounting defects will greatly affect the bearing fault in industrial place [1]. Due to the Complexity of the vibration data set, the famous feature extraction method, Empirical Mode Decomposition(EMD) is introduced. EMD method was introduced by Huang et al [23]. Intrinsic Mode Function is an output of EMD where it is decomposed from the original signal. The IMFs extracted by using EMD have the local characteristics information of the actual vibration signal. There is not much paper working on a run-to-failure vibration signal. This is because run-to-failure vibration signal contain noise that make it difficult to detect the degradation of bearing when the signal to noise ratio (SNR) is less than 1.

2.2.3 Feature Extraction

Feature extraction is an important method to deal with a complicated data set. The main function of feature extraction is to extract the important data from an original data. These extracted features will have the same property as the original signal. The only difference is that the extracted features are simpler. By using feature extraction method, it is able to speed up the generalization process of an ANN, besides, it also improves the overall accuracy of the ANN. The works in [1] and [12] use EMD and ANN to diagnose different bearing fault. In [6], a statistical method is used to extract the important feature from the vibration signal and then the extracted features are feed into Support Vector Machine (SVM) and ANN.

The function of EMD is to decompose a signal $x(t)$ into intrinsic mode decomposition as follows:

1. All local minima must be identified and form a cubic spine line by using the local minima as the lower envelope.
2. All local maxima must be identified and form a cubic spine line by using the local maxima as the upper envelope.
3. Determine the mean $m_1(t)$ of upper envelope and lower envelope.
4. Calculate the difference between the signal $x(t)$ and the mean $m_1(t)$. The difference between them is known as the first component of the IMF, $I_1(t)$.
5. If $I_1(t)$ is not an IMF, it will be treated as an original signal and repeat step 1 to step 4 to determine the first IMF, which is known as sifting process

   $I_{1k}(t) = I_{(1(k-1))}(t) - m_{1k}(t)$ …………… (1.1)

   $c_1(t) = I_{1k}(t)$ …………………………..… (1.2)

   There are stoppage criteria for the sifting process. The number of sifting process will be determined by the stoppage criteria.

   $$SD_k = \sum_{t=0}^{T} \frac{[I_{k-1}(t) - I_k(t)]^2}{I_{k-1}^2(t)} \text{ ……..…… (1.3)}$$

   Once the standard deviation of a IMF component is less than a pre-determined standard deviation, the sifting process stops.
6. Once the first IMF $c_1(t)$ is obtained by using the stoppage criteria, then subtract the first IMF $c_1(t)$ from the original signal.

   $r_1 = x(t) - c_1(t)$ ………………………..… (1.4)

   $r_n = r_{n-1} - c_n(t)$ ………………………… (1.5)

The sifting process will finally stop when the residual r_n becomes a monotonic function which mean there is no IMF can be extracted.

The feature extraction for empirical mode decomposition is as follows:

1. Bearing signal is decomposed into IMFs
2. Compute total energy for each IMF component

   $E_i = \sum_{j=1}^{a} c_{ij}$ ……………………………… (1.6)

   a denotes the length of each IMF.

   i denoted number of IMF component.
3. Calculate today energy of all IMF

   $E_T = \sum_{i=1}^{n} E_i$ ……………………………………………….……… (1.7)

. n denotes total number of IMF.

4. Construct feature vector

   $\left[H_{en}, \frac{E_1}{E_T}, \frac{E_2}{E_T}, \frac{E_3}{E_T}, \dots, \frac{E_n}{E_T}\right] = [H_{en}, H_{en}IMF1, H_{en}IMF2, \dots, H_{en}IMFn]$….. (1.8)

   H_{en} is known as the EMD energy entropy. It can be calculated as follows:

   $H_{en} = -\sum_{i=1}^{n} p_i \log(p_i)$ ……………………………….....… (1.9)

   $p_i = \frac{E_i}{E}$ ……………………………………………………….… (1.10)

In [1], a statistical analysis is used to determine the best features from the feature vector. It is as follows:

1. Compute the mean of a feature vectors of the kth class

   $\mu_k = \frac{1}{N} \sum_{n=1}^{N} x_{k,n}$ ………...…… (1.11)

   $x_{k,n}$ represents the nth extract of the kth class
2. Total average features vector can be calculated as follows:

   $\mu_c = \frac{1}{K} \sum_{k=1}^{K} \mu_k$ ……………………………………….…..… (1.12)
3. Intra-class variance matrix of average dispersion between different class can be calculated as follows:

   $V_{intra} = \frac{1}{KN} \sum_{k=1}^{K} \sum_{n=1}^{N} (x_{k,n} - \mu_k)(x_{k,n} - \mu_k)^t$ …………… (1.13)
4. Inter-class variance matrix of average dispersion between different classes:

   $V_{inter} = \frac{1}{K} \sum_{k=1}^{K} (\mu_k - \mu_c)(\mu_k - \mu_c)^t$ ……………………….…… (1.14)

5. Lastly, the J degree can be computed as follows:

$$J = trace(S_{intra}^{-1} S_{inter}) \quad \ldots \quad (1.15)$$

The higher the value of J degree, the better the feature becomes.

Before start doing the statistical analysis for the feature vector, the feature must be normalized. It can be done as follows:

$$x_{k,n} = \frac{x_{k,n} - mean(x_{1:K,n})}{\sqrt{var(x_{1:K,n})}} \quad \ldots \quad (1.16)$$

2.3 Artificial Neural Network(ANN)

ANN was produced to model in some way that the functionality can be represents the biological neural network. Biological neuron is made up of cell body, dendrites, axon and synapse as shown in Figure 7.

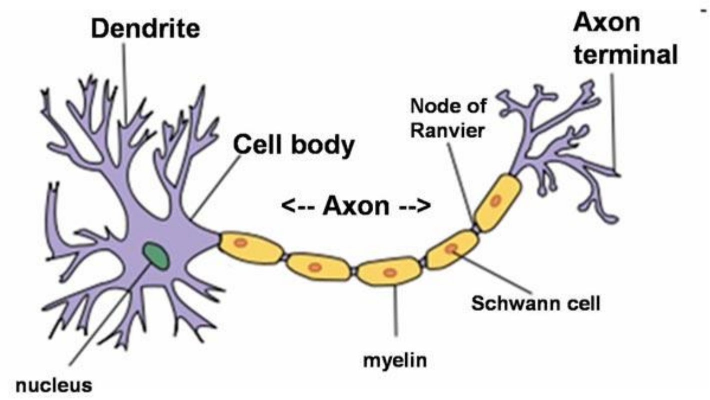


Figure 7: Biological neuron

Neurons are used to process information. Cell body contains a nucleus that contain information. A signal is received by neurons as an Impulse from other neurons through dendrites as a receiver. After the body cell generates a signal, the signal is then ready to be channelled to the synapse through Axon. Artificial neural network was inspired by biological neurons. An ANN consists of inputs, input weights, hidden nodes, output, and output weights as shown in Figure 8.

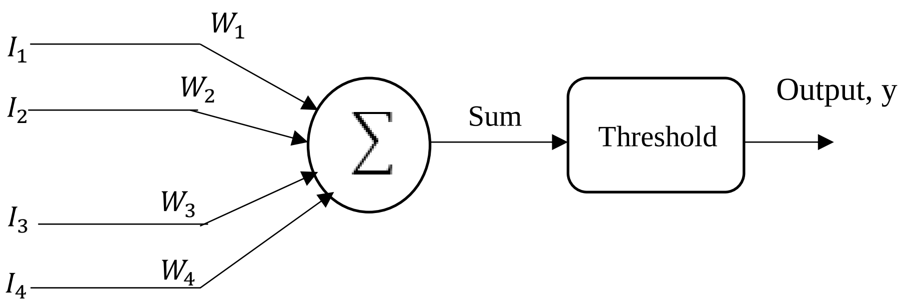


Figure 8: Artificial Neuron [21]

The output of the ANN is ‘1’, when the sum of outputs of the hidden layer is more that the threshold value. Conversely, the output of ANN is ‘0’, when the sum of outputs of the hidden layer is less than the Threshold value. Besides, the ANN can be trained by doing adjustment on the weights in the network to obtain output that close to desired output. The adjustment on the weights can be easily explained as the memory to the neural network [21]. The weight will be modified during the training session until it achieved the lowest error rate [21].

2.4 Learning Algorithms

2.4.1 Back Propagation algorithm (BP)

BP algorithm can be considered as one of the oldest learning algorithms to train a neural network. Gradient decent method is normally used by the BP algorithm to optimize the weight for training a neural network [8]. SLFN is able to learn the input output mapping by using the BP algorithm. Input data at the input layer will be fed forward to the hidden layer of the SLFN by multiplying the input weight and then the output of the hidden layer is propagated to the output layer by multiplying the output weight. Back propagation is implemented, after it feedforward from input layer to the output layer to generate an output response. It back propagates difference (error) between output and desired output to the hidden layer. This error can be said as a ‘responsibility’. It is to measure how much that specific node is “responsible” for the error in the output [3]. After back propagating the “responsibility” of output nodes to hidden nodes, it is then ready to perform weight updates for input weight and output weight. Next, it performs feedforward again to the output layer to generate the new output pattern. The error between the output and the desired output will be reduced. This training process is repeated until the difference between the real output and desired output converges to zero. The disadvantage of BP algorithm is it takes a long time for the error to converge to zero. Therefore, it is time consuming.

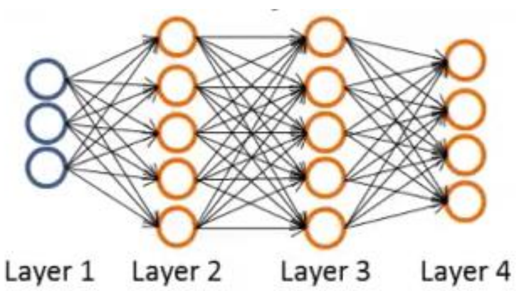


Figure 9: Artificial Neural Network

The first step in Back Propagation algorithm is to perform the forward propagation. The following equations are used.

$a^{(1)} = x \ add \ bias \ a_0^{(1)} \quad (layer\ 1)$………………………. (2.1)

$z^{(2)} = \theta^{(1)} a^{(1)}$…………………………………………. (2.2)

$a^{(2)} = g\left(z^{(2)}\right) \ add \ bias \ a_0^{(2)} \quad (layer\ 2)$………………... (2.3)

$z^{(3)} = \theta^{(2)} a^{(2)}$ …………………………………………. (2.4)

$a^{(3)} = g\left(z^{(3)}\right) \ add \ bias \ a_0^{(3)} \quad (layer\ 3)$ ………………... (2.5)

$z^{(4)} = \theta^{(3)} a^{(3)}$…………………………………..…... (2.6)

$a^{(4)} = g(z^{(4)}) \quad (layer\ 4)$ …………………………...…... (2.7)

The second step in Back Propagation algorithm is to perform the back propagation:

1. Compute the error difference for each output unit in output layer (layer 4) to the target output.

   $\delta_j^{(4)} = a_j^{(4)} - y_j$ ………………………………….. (2.8)

2. Back propagate the error or "the responsible" from 4th layer to 2nd layer:

   $\delta^{(3)} = (\theta^{(3)})^T \delta^4 .* g'(z^3)$ ………………………... (2.9)

   $\delta^{(2)} = (\theta^{(2)})^T \delta^3 .* g'(z^2)$ ………………………... (2.10)

3. The gradient or "derivative of cost function" can be calculated as below: -

   $\Delta^l = a^{(l)} \delta^{(l+1)}$ ……………………………...…... (2.11)

   Take note that the computation is without regularization term.

   The gradient with regularization term can be define as follows:-

   $\frac{\partial}{\partial \theta^l} J(\theta^l) = \frac{1}{m} \Delta^l + \lambda \theta^l$ ……………………….…... (2.12)

   Δ^l and θ^l have been vectorized.

4. Update the weight

$$\theta^l = \theta^l - \alpha \frac{\partial}{\partial \theta^l} J(\theta^l) \quad \text{..................................... (2.13)}$$

Take note that the α is the learning rate where it can not be too big in value as it will worsen the learning speed of gradient descent.

2.4.2 Extreme Learning Machine(ELM)

ELM was proposed by Guan Bin Huang in [5]. ELM was introduced to overcome the problem caused by BP algorithm. ELM has a faster learning speed as compared to BP algorithm and at the same time obtaining better generalization performance. In ELM algorithm, the input weight and hidden layer biases of a SLFN are randomly assigned within a certain range. Take note that, the input weight and hidden layer biases must be randomly assigned in the same range. The input weight can only be randomly assigned if the activation function of hidden layer is infinitely differentiable [5]. After randomly assigned the input weight and hidden layer biases, the SLFN is ready to feedforward to obtain the output of hidden layer. Next, the mapping between hidden layer output and the output of the network is assumed to be linear relationship. Therefore, Least Squared Estimator method (LSE) with Moore Penrose Pseudo Inverse of the hidden layer output matrix is used to compute the output weight of the SLFN. After that, the SLFN can feedforward from hidden layer to the output layer to obtain the output pattern. In additional, lowest error rate and smallest norm of weight can be achieved by using ELM algorithm to train the SLFN [5]. Besides, it can be easily implemented as compared to BP algorithm due to simple calculation. Furthermore, ELM has a faster training and testing speed as compared to BP because ELM only needs 1 iteration to obtain the optimized output weight whereas BP requires many iterations to optimize the output weight. The random assignment of input weight and hidden layer biases will cause a large change to the hidden layer output and this will lead to a large change in the output weight matrix. Thus, it will affect the outcome of the SLFN, which will result in SLFN with higher empirical risk and structural risk [7].

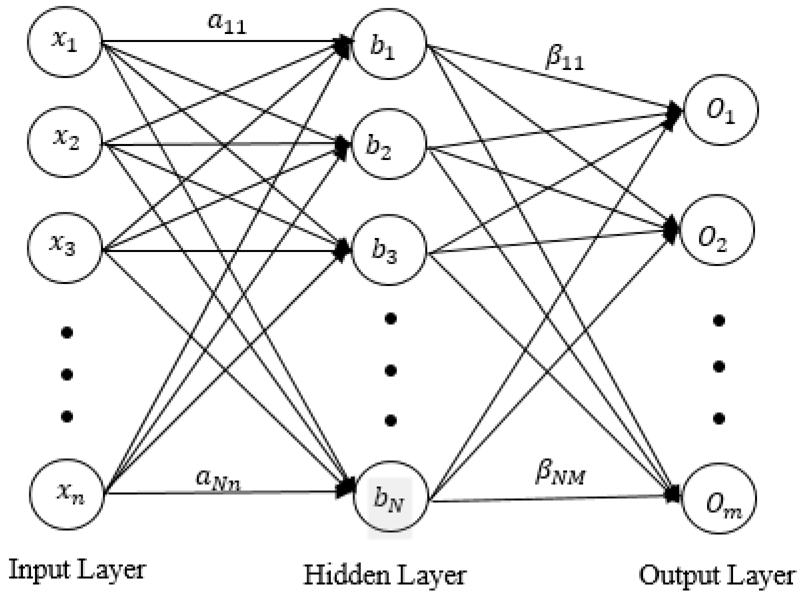


Figure 10: Extreme Learning Machine (ELM)

The general model of Extreme Learning algorithm is as shown below: -

$\sum_{i=1}^{N} \beta_i f_i(x_j) = \sum_{i=1}^{N} \beta_i f_i(a_i \cdot x_j + b_i) = t_j \; for \; j = 1 \dots M$ .... (3.1)

Where N represents number of hidden nodes, f(x) represents activation function, a_i represents input weight vector $a_i = [a_{i1}, a_{i2}, a_{i3}, \dots, a_{in}]$ connecting ith hidden nodes and the input nodes, output weight vector $\beta_i = [\beta_{i1}, \beta_{i2}, \dots, \beta_{im}]^T$ connecting ith hidden nodes with output nodes, b_i represents the hidden layer bias, x_j is the input samples. The activation function can be sigmoid, Radial Basis Function or Sine Function.

Hidden layer output, H is as shown below: -

$$H = \begin{bmatrix} f(a_1 \cdot x_1 + b_1) & \cdots & f(a_N \cdot x_1 + b_N) \\ \vdots & \cdots & \vdots \\ f(a_1 \cdot x_M + b_i) & \cdots & f(a_N \cdot x_M + b_N) \end{bmatrix}_{MxN}$$

The general formula can be written as below: -

$H\beta = T$ .... (3.2)

The output weight, β matrix is as shown below: -

$$\beta = \begin{bmatrix} \beta_1^T \\ \vdots \\ \beta_N^T \end{bmatrix}_{NxM} \quad \text{.................................} \; (3.3)$$

The target matrix, T is as shown below: -

$$T = \begin{bmatrix} t_1^T \\ \vdots \\ t_N^T \end{bmatrix}_{NxM} \quad \text{.................................} \; (3.4)$$

The output weight matrix was determined by using the least square estimated (LSE) method. This because LSE method can increase the accuracy of the output weight matrix(optimized). The output weight matrix will be optimized by using formula below: -

$\beta = H^{+}T$ ……….……................... (3.5)

$H^{+} = (H^{T}H)^{-1}H^{T}$ represents the Moore Penrose Pseudo Inverse.

$H^{T}H$ may not always be a square matrix and thus this method will not always perform well.

In order to overcome the problem, least square estimator with regularization theory is introduced as below: -

$\beta = (\lambda I + H^{T}H)^{-1}HT$ …………….. (3.6)

λ is the regularization parameter.

The Extreme Learning algorithm with a training set X, activation function $g(x)$ and a hidden node number N works as follows: -

1. Randomly assignment of input weight w_i and hidden layer bias b_i with a given range for $i = 1 \dots N$.
2. Compute the hidden layer output matrix $\boldsymbol{H}$.
3. Calculate the output weight β by using Least Square Estimator method:

$$\beta = H^{+}T$$

Where T is the target matrix and $H^{+} = (H^{T}H)^{-1}H^{T}$

4. Compute the output.

Lastly, [4] mentioned that, there are several disadvantages in ELM: -

1. ELM algorithm only minimized the empirical risk but never minimized the structural risk.
2. If there are outliers present in the input data, the performance of the ELM will be affected.
3. Unable to deal with noisy data set.
4. The random initialization of input weight affects the performance of SLFN because it causes a large change in hidden layer output and output weight matrix.

2.4.3 Finite Impulse Response Extreme Learning Machine(FIR-ELM)

Finite Impulse Response ELM (FIR-ELM) was proposed by Zhihong Man [8]. FIR-ELM was introduced to overcome the problems cause by SLFN trained with ELM. The random assignment of input weight and hidden layer biases have cause a large change to the hidden layer output and this will lead to a large change in the output weight matrix which will affect the accuracy of the SLFN. The second problem is ELM could not deal with a noisy input data. The performance of SLFN trained by ELM will be affected if there is noise in the input data. In FIR-ELM, all the linear hidden nodes are FIR filter which mean the hidden nodes work as a pre-processor to the input data [8]. The FIR filter can be any FIR filters such as low pass filter, high pass filter, band pass filter, band stop filter, rectangle filter and many more. FIR filter is chosen based on the input data. The function of FIR filter (pre-processor) is to filter out the noise and undesired frequency components. The FIR filter able to reduce the disturbance in input data, the structural risk and empirical risk of the SLFN [8]. The hidden layer biases of the FIR-ELM are randomly assigned within a given range which is the same as the ELM. However, the input weight of FIR-ELM is not randomly assigned like input weight in ELM. In FIR- ELM, input weight is designed by using the FIR filter so that the hidden layer can of the SLFN able to works as a pre-processor. Next, the output of hidden layer can be calculated by feedforward from input layer to hidden layer. Next, the output weight matrix can be calculated by using the Least Squared Estimator with Moore Penrose pseudo inverse of the output of hidden nodes and then the output can be obtained by feedforward from hidden layer to the output layer by multiplying the output weight to observe the output pattern.

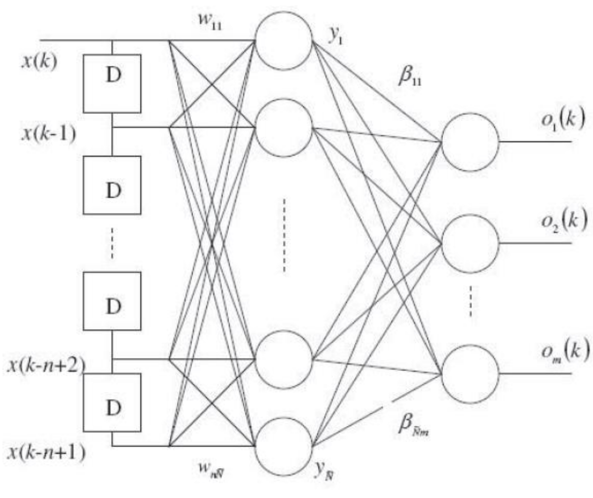


Figure 11: Finite Impulse Response ELM

The output of the hidden nodes can be calculated as follows:

$y_i(k) = \sum_{j=1}^{n} w_{ij} x(k-j+1) = w_i^T x(k) \quad for\ i = 1, \dots, N$ ……………. (4.1)

As you can see the structure of the formula looks similar to the formula structure of a FIR filter, where the input weight w_{ij} can represents the filter coefficients or impulse response coefficient. Thus, it can be said that the output $y_i(k)$ is the convolution sum of the filter coefficient and the input data.

Assumed that the desired frequency response of hidden nodes can be represented as below:

$H_{id}(\omega) = \sum_{k=0}^{\infty} h_{id}[k] e^{-j\omega k}$ ……………………………………..…. (4.2)

It can be converted to time domain by inverse DTFT:

$h_{id}[k] = \frac{1}{2\pi} \int_{-\pi}^{\pi} H_{id}(\omega)\, e^{j\omega k}$ ……………………………………..……. (4.3)

The length of the filter must be the same as the input data. Based on the formula of the output of hidden nodes, the input data starts at $x(k-(n-1))$ to $x(k)$ which mean it takes n-1 past data for input data. Therefore, the length of the filter must be n. Let's consider a rectangle filter $z[k]$ for the input data to allow n desired frequency responses :

$$z[k] = \begin{cases} 1 & for\ k = 0,1,2,\dots,(n-1) \\ 0 & otherwise \end{cases}$$

The Fourier transform of $d[k]$ is as shown below:

$Z(\omega) = \sum_{k=0}^{n-1} z[k]\, e^{-j\omega k} = e^{-j\omega(\mathrm{n}-1)/2} \frac{\sin(\frac{\omega \mathrm{n}}{2})}{\sin(\frac{\omega}{2})}$ ……………..…..……. (4.4)

After converting input data and rectangle filter from time domain into frequency domain, then the convolution can be done as follow:

$\widehat{H_{id}}(\omega) = \frac{1}{2\pi} \int_{-\pi}^{\pi} H_{id}(v) Z(\omega - v)\, dv$ ……………………………..…. (4.5)

If the researcher chose Low pass filter to be performed in the hidden nodes, the frequency response is as shown below:

$$H_{id}(\omega) = \begin{cases} 1e^{-j\omega(\mathrm{n}-1)/2} & for\ |\omega| \le \omega_c \\ 0 & for\ \omega_c < |\omega| < \pi \end{cases}$$

ω_c denotes as the cut off frequency. This low pass filter preserves the low frequency component while the high frequency component is eliminated. The low pass filter formula can be converted into time domain as follow:

$\widehat{h_{id}}[k] = \frac{1}{2\pi} \int_{-\omega_c}^{\omega_c} e^{-j\omega(\mathrm{n}-1)/2}\, e^{j\omega k}\ d\omega = \frac{\sin[\omega_c\left(k-\frac{n-1}{2}\right)]}{\pi\left(k-\frac{n-1}{2}\right)}$ ……………….. (4.6)

$\widehat{h_{id}}[k] = \widehat{h_{id}}[n-k+1]$ ……………………………………..……. (4.7)

The $\widehat{h_{id}}[k]$ value can then be assigned to the input weight as follow:

$w_{i1} = \widehat{h_{id}}[0], w_{i2} = \widehat{h_{id}}[1], \dots, w_{in} = \widehat{h_{id}}[n-1]$ ……………….…. (4.8)

The working principle of FIR-ELM learning algorithm can be summarized in few steps: -

1. Set the n desired frequency responses to the input weight w_{ij}
2. Compute the hidden layer output matrix $\boldsymbol{H}$ as in ELM.
3. Compute the output weight matrix $\boldsymbol{\beta}$ as in ELM.
4. Compute the output of ANN.

Output of hidden layer can be computed as follows: -

$H_i = \sum_{j=1}^{n} w_{ij} x(k-j+1) = w_i^T x(k) \mid for\ i = 1, \dots, \widetilde{N}$ …………… (4.9)

Calculate the output weight β by using Least Square Estimator method:

$\beta = H^{+}T$ ……………………………………………………….……. (4.10)

Where T is the target matrix and $H^{+} = (H^T H)^{-1} H^T$

$H^T H$ may not always be a square matrix and thus this method will not always perform well.

In order to overcome the problem, least square estimator with regularization theory is introduced as below: -

$\beta = (\lambda\mathrm{I} + H^T H)^{-1} HT$ …………………………………..….….….… (4.11)

The output of the SLFN can be computed as follows: -

$O_i(k) = \sum_{p=1}^{\widetilde{N}} \beta_p w_p^T x(k) \mid for\ i = 1, \dots, m$ ………………..….….….… (4.12)

2.4.4 Discrete Fourier Transform Extreme Learning Machine(DFT-ELM)

The DFT-ELM is proposed by Zhihong Man to further improve the robustness of FIR-ELM [7]. The working principle of DFT-ELM is almost same as FIR-ELM and ELM. DFT-ELM was introduced to further improve the mapping between input and output, and reduce the empirical and structural risk. If a nonlinearly separable input data is used, the input weight of SLFN will be trained in a way that the hidden layer able to assign DFT of all feature vector to the desired position is the frequency domain [7]. In DFT-ELM, the input weight is trained by the regularization theory to reduce the error between the frequency component of the desired feature vectors and features vectors from the hidden layer of SLFN. The working principle of the DFT-ELM is the same as the FIR-ELM, the only different is the input weight of the DFT-ELM is trained by the regularization theory and frequency component from discrete fourier transform while the input weight from FIR-ELM is designed by using FIR filter. As a result, robustness of the SLFN trained with the DFT-ELM will significantly increase when dealing input data with noise compared to SLFN trained with BP, ELM and FIR-ELM. The regularization technique is applied to optimize the output weight to further reduce the structural and empirical risk,. With this, Single Hidden Layer Neural Network Trained by DFT-ELM results a better robustness property with respect of input disturbance.

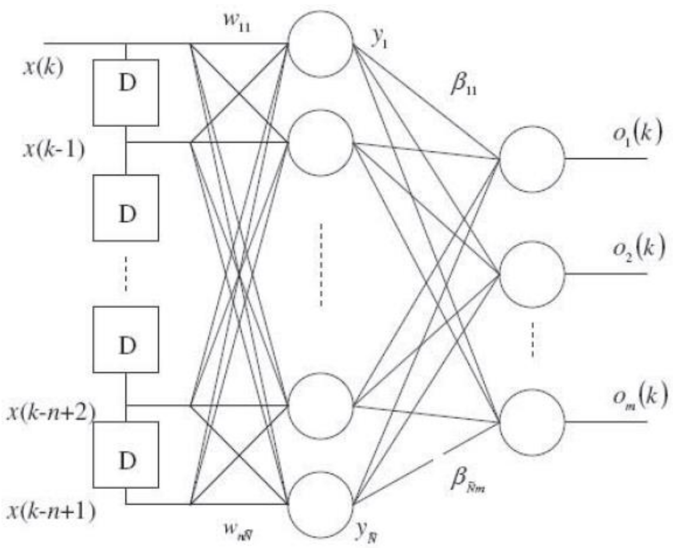


Figure 12: Discrete Fourier Transform ELM [8]

The DTF algorithm can be summarized into 5 steps:

(1) Obtain the frequency- spectrum sample matrix from DFT $\boldsymbol{Y_d}$.

(2) Compute the transformation matrix $\bar{\mathbf{A}}$.

(3) Compute the optimal input weight matrix w.

(4) Compute the hidden layer output matrix **H**.

(5) Compute the optimal output weight matrix $\boldsymbol{\beta}$.

(6) Compute the output of ANN.

$$Y_d = [Y_1 \quad Y_2 \quad Y_3 \quad \cdots \quad Y_{dN}] \;\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \;(5.1)$$

$$\overline{A} = \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & e^{-j2\pi/\widetilde{N}} & e^{-j4\pi/\widetilde{N}} & \cdots & e^{-j2\pi(\widetilde{N}-1)/\widetilde{N}} \\ 1 & e^{-j4\pi/\widetilde{N}} & e^{-j8\pi/\widetilde{N}} & \cdots & e^{-j4\pi(\widetilde{N}-1)/\widetilde{N}} \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ 1 & e^{-j2\pi(\widetilde{N}-1)/\widetilde{N}} & e^{-j4\pi(\widetilde{N}-1)/\widetilde{N}} & \cdots & e^{-j2\pi(\widetilde{N}-1)(\widetilde{N}-1)/\widetilde{N}} \end{bmatrix} \ldots\ldots\; (5.2)$$

$$w = \frac{1}{\widetilde{N}}\left(\frac{d_1}{\gamma_1 \widetilde{N}}\boldsymbol{I} + XX^T\right)^{-1} XY_d^T\overline{A} \;\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \;(5.3)$$

$$H = w^T X \;\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \;(5.4)$$

$$\beta = (\boldsymbol{\lambda}\mathbf{I} + H^T H)^{-1} H^T T \;\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \;(5.5)$$

X denotes input samples.

T denotes desired output.

$\widetilde{N}$ denotes number of hidden layer.

$\frac{d_1}{\gamma_1} = \boldsymbol{\lambda}$ denotes the regularization theory.

2.5 Related works on bearing fault diagnosis

The research projects in table 1 focus on the bearing fault diagnosis using vibration data set. Table 1 shows the analysis on the merits and demerits of the learning algorithm used for each research projects.

Table 1: Related work on bearing fault diagnosis

No	Author	Year	Title	Feature Extraction	Learning Algorithm	Application	Merits	Demerits
1	Jaouher Ben Ali, Nader Fnaiech, Lotfi Saidi, Brigitte Chebel-Morello, Farhat Fnaiech.	2014	Application of empirical mode decomposition and artificial neural network for automatic bearing fault diagnosis based on vibration signals	Empirical Mode Decomposition (EMD)	ANN (Back Propagation algorithm)	Bearing fault diagnosis based on vibration signal	The ANN has 93% of diagnosis accuracy.	Back Propagation takes a long time to train the ANN.
2	P.K. Kankar, Satish C. Sharma, S.P. Harsha	2011	Fault diagnosis of ball bearing using machine learning methods.	Statistical Method	Support Vector Machine (SVM), ANN (Back Propagation algorithm)	Bearing fault diagnosis based on vibration signal	The SVM has a higher diagnosis accuracy as compared to ANN.	The maximum diagnosis accuracy is not good enough. Which is 93% accuracy.
3	Yang Yu, YuDejie, Cheng junsheng	2006	A roller bearing fault diagnosis method based on EMD energy entropy and ANN.	EMD	ANN (Back Propagation algorithm)	Bearing fault diagnosis based on vibration signal	The ANN has around 95% for the diagnosis accuracy.	Back Propagation takes a long time to train the ANN.
4	Chun-Chieh Wang, Yuan kang, Ping-Chen Shen, Yeon-Pun Chang, Yu-Liang Chung	2010	Application of fault diagnosis in rotating machinery by using time series analysis with neural network	Autoregressive (AR)	ANN (Back Propagation algorithm)	Bearing fault diagnosis based on vibration signal	The difference value of AR coefficient with BPNN was superior to AR coefficient with BPNN	BPNN takes too long to train a ANN.
5	Jacek Dybala, Radoslaw Zimroz	2014	Rolling bearing diagnosing method based on Empirical Mode Decomposition of machine vibration signal	Empirical Mode Decomposition	-	Bearing fault diagnosis based on vibration signal	Successfully identify the defect at bearing outer race	Only consider bearing fault during early stage
6	Jinde Zheng, Junsheng Cheng, Yu Yang	2013	Generalized empirical mode decomposition and its application to rolling element bearing fault diagnosis	Generalized EMD	-	Bearing fault diagnosis based on vibration signal	Improved the time-frequency analysis for non-stationary and non0linear data.	DEMD required more calculated as compared to EMD.
7	Diego Fernandez-Francos, David Martinez-Rego, Oscar Fontenlaa-Romero, Amparo Alonso-Betanzos	2013	Automatic Bearing fault diagnoses based on one-class v-SVM	-	One-class v-SVM	Bearing fault diagnosis based on vibration data	Able to detect the bearing fault automatically based on the vibration spectrum.	Complex calculation.
8	Guang-Bin Huang, Qin-Yu Zhu, Chee-Kheong Siew	2006	Extreme Learning Machine: Theory and application	-	SLFN (Extreme Learning Machine	Handwritten Character Classification	Faster and simpler Computation.	Unable to deal with noisy input which will result in higher empirical and structural risk.

								Random initialization of input weight cause large change at the output matrix.
9	Zhihong Man, Kevin Lee, Dianhui Wang, Zhenwei Cao, Chunyan Miao	2011	A new robust training algorithm for a class of single hidden layer feedforward neural network	-	SLFN (Finite Impulse Response Extreme Learning Machine)	Classification	It able to remove the effect of input disturbance and reduce the empirical and structural risk.	Complex computation as compared to ELM.
10	Zhihong Man, Kevin Lee, Dianhui Wang, Zhenwei Cao, and Suiyang Khoo	2012	Robust Single Hidden Layer Feedforward Network-Based Pattern Classifier	-	SLFN (Discrete Fourier Transform Extreme Learning Machine	Hand writtern character classification	Result in lower empirical and structural risk as compared to FIR-ELM. Input weight is Optimized	Complex computation.

2.6 Critical Analysis

Based on Table 1, every group of researchers are using different types of feature extraction. Most of the feature extraction works the same which is to extract the important feature and remove the unwanted features. In [1] and [12], EMD is used as a feature extraction to the vibration data set. However, In [6], the useful features are extracted from the original vibration data set by using statistical method. Furthermore, in [11], autoregressive method is used as a feature extraction to the vibration signal. Feature extraction method is chosen based on one's objective.

Besides that, all work in Table 1 use the same learning algorithm which is BPNN. In [6], the paper shows the comparison between Support Vector Machine and BPNN. In [1] and [12], both of these papers are using the same learning algorithm (BPNN) to train the neural network in bearing fault diagnosis application. In addition, the research projects above are dealing with time series data set which means the input layer is time-tapped delay line to allow time series data to be an input to the SVM or BPNN.

In [14], the bearing fault diagnosis based on vibration data is done by using the EMD and IMF identification and aggregation. The EMD is used to break the original vibration data into few intrinsic mode functions (IMFs). Next, IMF identification and aggregation will identify the noise-only signal, signal-only part of signal and trend-only part of signal. After identifying the signal, kurtosis analysis is used to identify the bearing fault. If the kurtosis value of noise-only part of signal is high, it means that some bearing fault occurs.

In [15], a new algorithm, Generalized Empirical Mode decomposition (GEMD) has been proposed. The function of GEMD is the same as EMD but the decomposed signal is known as generalized intrinsic mode function (GIMF). The GEMD algorithm have improved the time-frequency analysis for the non-stationary and non-linear vibration data. The result shown is significantly better as compared to the result in [14]. The disadvantage of the GEMD is it requires more calculation than the EMD but the result shown by the GEMD is much more accurate than EMD [15].

On the other hand, ELM is developed by Guang-Bin Huang in [5]. This learning algorithm is proposed to overcome the slow learning speed problem caused by Back Propagation algorithm (BP). The learning speed can be said 10 times faster compared to BP. The disadvantage of ELM is it is unable to deal with noisy data [5]. The disturbance in noisy data will increase the empirical and structural risk of ELM

[5]. The other demerit of ELM is the random assignment of input weight caused a large change in the hidden layer output matrix resulting in large change in output weight matrix which will affect the output of a ANN. This result in less robustness of the learning algorithm.

The FIR-ELM is introduced to overcome the less robustness of ELM caused by the random initialized input weight. In FIR-ELM, the input weight is assigned based on FIR filter coefficient which is different from the ELM [8]. The hidden layer of the FIR-ELM works as a pre-processor to the input data which filter the input disturbance [8]. This results in a lower empirical and structural risk. In order to further improve the robustness of SLFN, DFT-ELM is proposed. In DFT-ELM, the input weight is trained with regularization theory and the desired feature vector which will result in lower empirical risk and structural risk as compared to FIR-ELM [7]. The training method for the output weight for ELM, FIR-ELM, and DFT-ELM is the same which is the similarity between them.

2.7 Summary

The bearing data set used in this project is collected based on vibration signal. Therefore, this bearing vibration signal is complex because it has nonlinear and non-stationary characteristics. This is why feature extraction is used in this project. The feature extraction method used in this project is Empirical Mode Decomposition, where it extracts the important feature and remove the unwanted features. This will significantly improve the performance of SLFN. Extracted features will be fed into SLFN as input and trained with 4 learning algorithms which are BP, ELM, FIR-ELM and DFT-ELM.

CHAPTER 3

METHODOLOGY

3.1 Project Methodology

Figure 13 shows the steps in classifying bearing faults using ANN as a classifier.

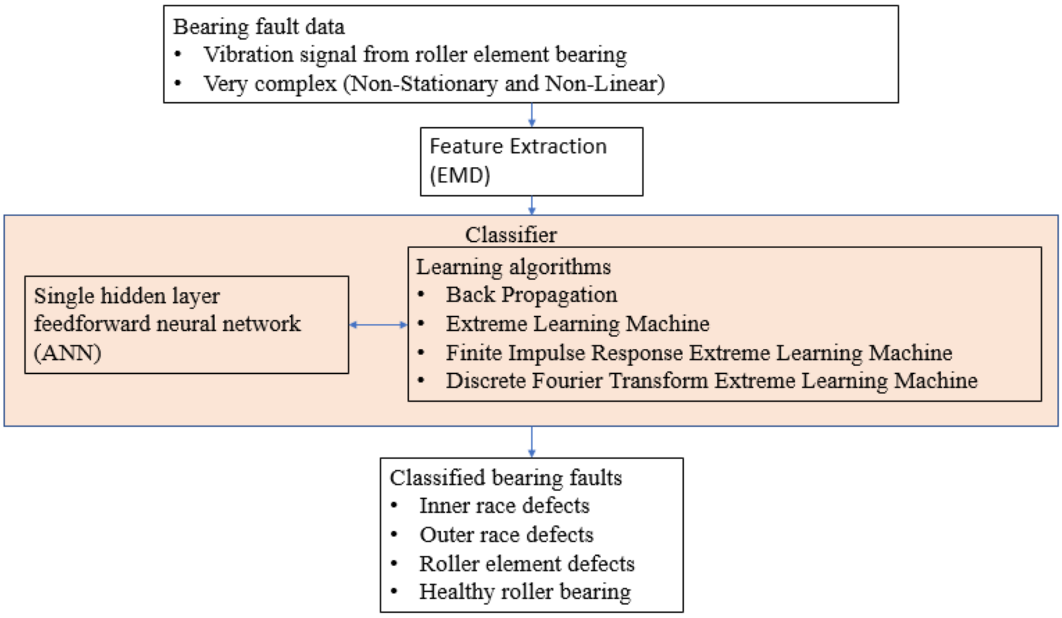


Figure 13: Methods used to classify bearing faults

<u>Bearing Fault Data</u>

The bearing vibration data set provided by NSF I/UCP Centre for Intelligent Maintenance is very complex (non-stationary and nonlinear). Therefore, a suitable feature extraction method must be used to extract the important features.

<u>Feature Extraction</u>

Empirical mode decomposition (EMD) is used as a feature extraction to the complex vibration data. EMD is used because it is better in dealing with non-stationary and nonlinear signal as it extracts the frequency component of an original signal. The function of EMD is to break the original bearing vibration signal into Intrinsic Mode Functions(IMFs). The formulas of EMD method can be referred to equation (1.1) to (1.5).

After extracting IMFs from the original bearing vibration data, it is ready to form a feature vector that consists of the energy entropy of the IMFs and energy entropy of the whole signal. Energy entropy of IMFs can be a feature vector of the original vibration data because the sum of the energy of each IMF is equal to the energy of the original bearing vibration data. The formulas of computing the energy entropy of IMF signals can be referred to (1.6) to (1.10). The feature vector is as shown in equation (1.8)

<u>Classifier</u>

Next, the feature vector is used as an input to the SLFN. A simplified feature vector will improve the overall performance, time taken for training and time taken for testing of a SLFN. After setting the feature vector as an input to the SLFN, few learning algorithms are used to train the SLFN with respect to its desired output response. The learning algorithms are BP, ELM, FIR-ELM, and DFT-ELM.

<u>What is the output of the classifier?</u>

The desired output response can be divided into four groups which are inner race defects, outer race defects, roller element defects and healthy bearing. Since the SLFN is only required to classify four groups of bearing fault, therefore, four bits is used to represents different group of bearing faults as the desired output response. [1 0 0 0] represents healthy bearing, [0 1 0 0] represents inner race defects, [0 0 1 0] represents roller element defects, [0 0 0 1] represents outer race defect. After the training the SLFN, the process is repeated for the testing phase with different input data. In testing phase, a new input data is fed into the EMD to extract its IMFs and then form a simplified feature vector. Next, the feature vector is used to evaluate the performance of the SLFN which is trained by different learning algorithms. Besides, there is no desired output response in testing phase. The trained SLFN model must generate its own output response using the new input data and then the testing accuracy is computed based on the output response.

3.2 Research Methodology

The research methodology will be explained in detail in the following section.

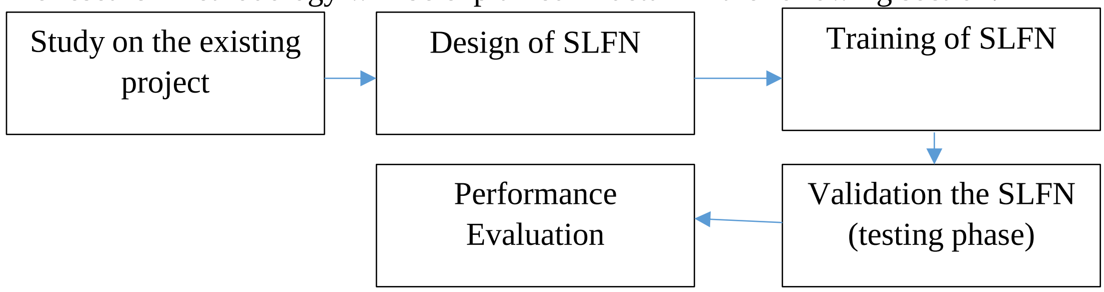


Figure 14: Research Methodology

3.2.1 Study on existing project.

During this stage, the author starts exploring all the basic understanding about ANN and learning algorithms to train the ANN. The exploration is done by reading the existing paper and understand the working principle of the project. All the basic knowledge about the ANN and learning algorithms are stated in the Literature Review.

3.2.2 Design of SLFN

Single Hidden Layer Feedforward Neural Network (SLFN) consists of 3 layers which are input layer, hidden layer and output layer. Besides, there are weights connecting between input layer and hidden layer and between hidden layer and output layer as shown in Figure 15

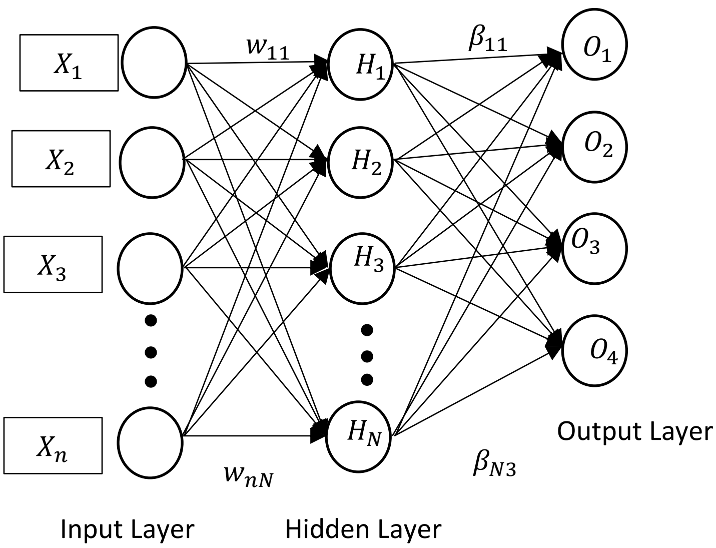


Figure 15: ANN Architecture

- X represents input.
- W represents input weight (connections between input layer and hidden layer).
- β represents output weight (connections between hidden layer and output layer).
- n represents number of input layer.
- N represents number of hidden layer.

Table 2: Parameters used to model the SLFN

Parameters	Value
Input layer, X	14
Input Weight for BP and ELM	Randomly assigned with a given range.
Input Weight for FIR-ELM	Assigned based on filter coefficient
Input Weight for DFT-ELM	Trained with regularization theory with respect to the desired frequency response in the hidden layer.
Hidden Nodes for BP	40
Hidden nodes for ELM	100
Hidden nodes for FIR-ELM	50
Hidden nodes for DFT-ELM	50
Output nodes	4

Table 3: Specification of Laptop

Laptop Specification	
Laptop	Acer Aspire v5
Processor	Intel CORE i5
RAM	4GB

In this project, there are 4 categories of bearing to be classified which are Healthy bearing, Inner race defects, Outer race defects and roller element defects. Furthermore, the author is using an Acer laptop with i5 processor and 4 GB RAM which will result in taking even longer time to process the data. Therefore, 10000 bearing vibration data points are processed by EMD and result in 14 IMFs. This is due to the time consumption for EMD to extract the IMF from the raw vibration signal. When more IMFs need to be extracted, the time taken will increase as well. Hence, 14 IMFs are extracted from each category of vibration data (10000 data points) set using EMD. Hence, there are 4 sets of 14 IMFs, each set representing different category of bearing. The IMFs matrix is as shown below: -

$$IMF = \begin{bmatrix} imf_1^1 & \cdots & imf_{10000}^1 \\ imf_1^2 & \cdots & imf_{10000}^2 \\ \vdots & \cdots & \vdots \\ imf_1^{14} & \cdots & imf_{10000}^{14} \end{bmatrix} \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (6.1)$$

Next, all the IMFs will be simplified into a feature vector consist of the energy entropy of all IMFs and energy entropy of the whole signal. The feature vector is as shown below: -

$$\left[H_{en}, \frac{E_1}{E}, \frac{E_2}{E}, \frac{E_3}{E}, \ldots, \frac{E_{14}}{E}\right] = [H_{en}, H_{en}IMF1, H_{en}IMF2, \ldots, H_{en}IMF14] \ldots\ldots\ldots\ldots (6.2)$$

Based on the feature vector, if 14 IMFs is extracted from the bearing vibration data, energy of each IMF must be computed. Hence, there is 14 energy entropies $\left(\frac{E_1}{E}, \frac{E_2}{E}, \frac{E_3}{E}, \ldots, \frac{E_{14}}{E}\right)$ for 14 IMFs. Hence, an energy entropy feature vector is formed for each class of bearing. Since there are four types of bearing to be classified using the SLFN, 4 feature vectors are combined to form a matrix and this matrix will be an input to the SLFN. Hence, the number of input node in input layer of the SLFN is 14 so that the input layer can accommodate 14 extracted energy entropies for each class of bearing. Number of hidden node is set to 50 nodes. Output nodes is 4 for 4 classes of classification.

3.2.3 Training of SLFN

In training phase, the feature matrix is set to be an input to the SLFN and it is ready to be feedforward to the output layer to obtain the real output response. The real output response will be compared to the desired output response for evaluating the training accuracy. As for the ways in computing the output response of SLFN, it is different for each learning algorithm. It is explained in detail in Literature Review chapter. In training phase, overfitting must be avoided. Overfitting can be defined as when a model tends to fit the training data completely which produces approximately 100% training accuracy which in turn worsen the performance of a SLFN in testing phase. This can be explained when SLFN fit very well with the training data which will cause it unable to fit the testing data and result in the output response to be incorrect and less accurate. To overcome this problem, regularization theory is used. The function of regularization theory is to overcome overfitting problem. Regularization theory in considered in all the learning algorithms. When the value of regularization parameter increases, the model will be moving toward underfitting. On the other hand, when the value of regularization parameter decreases, the model will be moving toward overfitting. Therefore, the value of regularization parameter must be chosen wisely. The aim of the model is to achieve a “just nice” fitting to the data set.

3.2.4 Validation of SLFN

In validation of SLFN, new data is used to validate the accuracy of the trained SLFN. The new data will undergo feature extraction (EMD) and obtain 14 IMFs for each class. Next, the energy entropy of each IMF and energy entropy of the whole signal are required to form a feature vector for testing. After the feature vector is formed, it is ready to evaluate the SLFN with different learning algorithms. All the feature vectors are combined to form a feature matrix. This feature matrix(input) will be feedforward to the output layer to obtain an output response. From this output response, the fault diagnosis accuracy can be calculated by comparing it to the desired output response. The fault diagnosis is measured by using mean squared error (MSE). MSE can be defined as the mean of squared difference between real output and desired output. The formula of MSE is as shown below: -

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(O-T)^2 \quad (6.3)$$

Where T represents desired output response, O represents real output response.

3.2.5 Performance Evaluation

The diagnosis accuracy obtained in testing phase will be compared to the result of existing project to evaluate the performance of model. The diagnosis accuracy from SLFN is expected to be higher than the existing work because most of the exist work the fault diagnosis is done by using Back Propagation algorithm. Back Propagation algorithm used a long time to train a SLFN whereas in this project ELM, FIR-ELM, and DFT-ELM are used where the training time is 10 times faster than Back Propagation algorithm.

3.3 Data Set Management

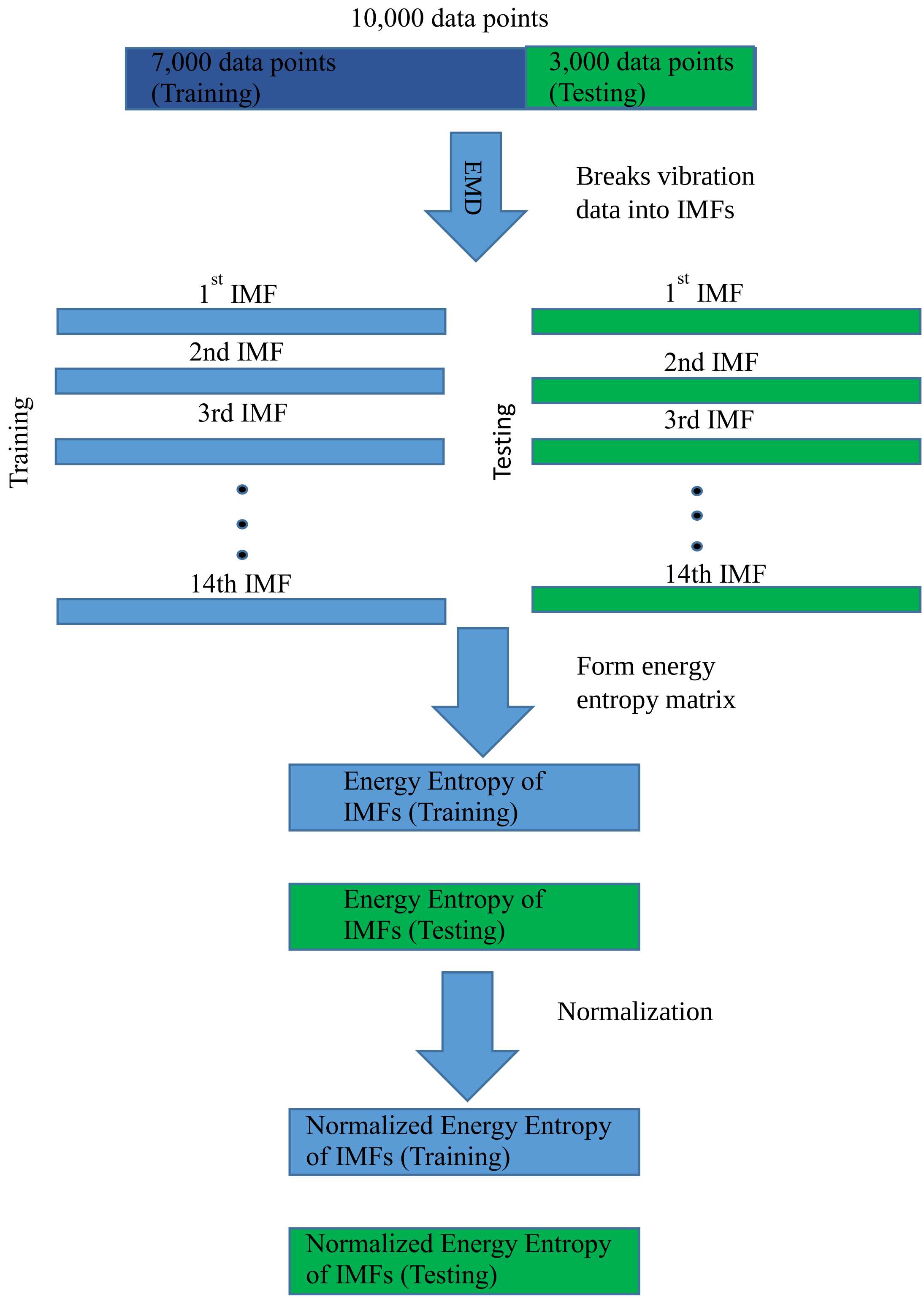


Figure 16: Steps in Data Processing

In this section, Figure 16 describes the process of extracting the feature from the raw vibration signal. The vibration signal is divided into two parts which are training vibration data and testing vibration data. 70% of the raw vibration data (7000 data points) are used for training and 30% of the raw vibration data (3000 data points) are used for testing.

Next, both vibration data sets will undergo a feature extraction process called Empirical Mode Decomposition. This process breaks the vibration data into 14 different Intrinsic Mode Function (IMF) for training and testing data. In order to simplify the data, each of the IMFs is the summed up to produce one energy entropy that representing a particular IMF. Therefore, there are 14 energy entropies for training and testing data.

Lastly, normalization process is essential to normalize the energy entropy because each energy entropy has a large value.

3.4 Data set and Target set Arrangement

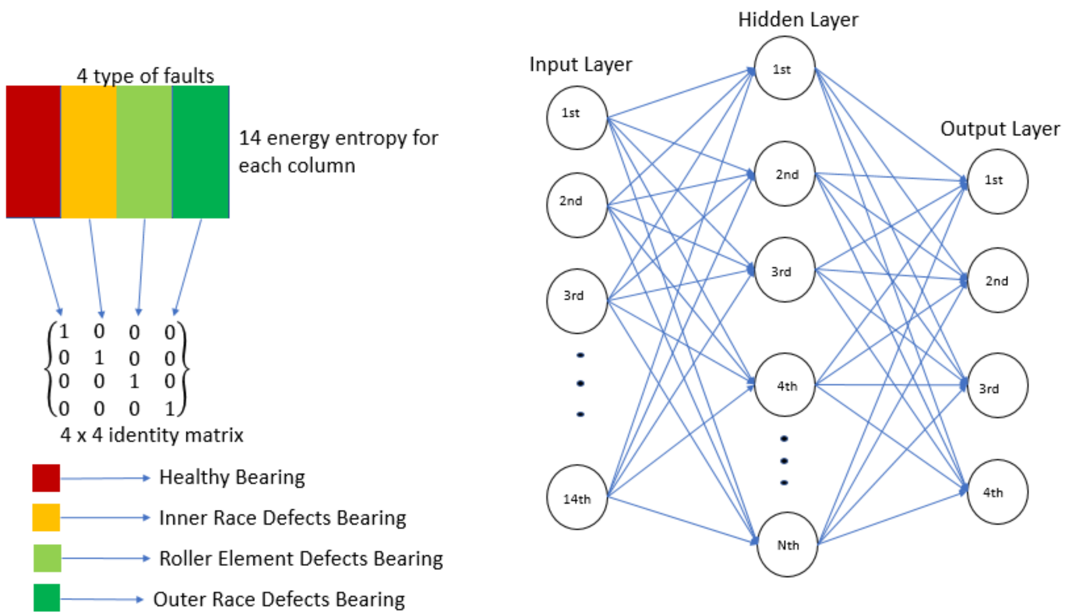


Figure 17: Data set and Target set Arrangement for ANN model

As shown in Figure 17, the training set and testing set are arranged into four classes where red represents Healthy bearing, Orange represents Inner race defects bearing, leaf green represent Roller element defects bearing and dark green represents Outer race defects bearing.

Each class in training set has 14 energy entropies. Each energy entropy is calculated from the sum of one series of IMF. Take note that, the energy entropies have been normalized. Since there are 14 energy entropies in each class, so 14 nodes for input layer are required. Besides that, there are 4 classes to be classified in the output layer. Therefore, 4 nodes are required for the output layer.

As for the number of hidden nodes, the higher the number of hidden nodes, the longer it takes to train SLFN and result the better the accuracy. Conversely, when a low number of hidden nodes is used, the SLFN can be trained very quickly but the accuracy is low.

3.5 Summary

Feature Extraction(EMD) method is very important to simplify the bearing vibration data which is very complex. After feature extraction process, the extracted feature must be arranged properly as shown in Figure 17 and then it can be used as an input to the SLFN and then trained with BP, ELM, FIR-ELM and DFT-ELM with respect to the target output (Supervised learning). The validation of SLFN is done by using a new set of data to feed into the SLFN and then evaluate the error rate of the output. The error rate is calculated by using the Mean Squared Error and then performance evaluate will be done within 4 learning algorithms

CHAPTER 4

Result and Discussion

4.1 Vibration signals

The designed SLFN in this project is required to classify four classes such as healthy bearing, inner race defects, outer race defects, and roller element defects. Different types of bearing fault will result in different vibration signal. The vibration signal of healthy bearing, inner race defect bearing, roller element defect bearing and outer race defect bearing are shown in Figure 18, Figure 19, Figure 20, and Figure 21 respectively.

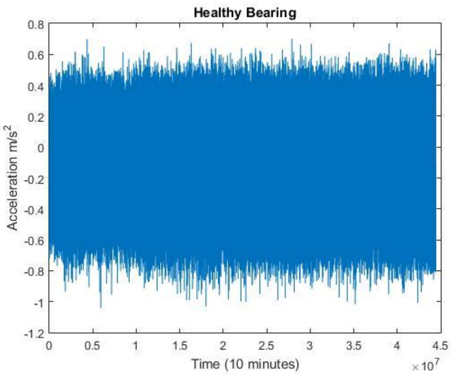


Figure 18: Vibration signal of healthy bearing

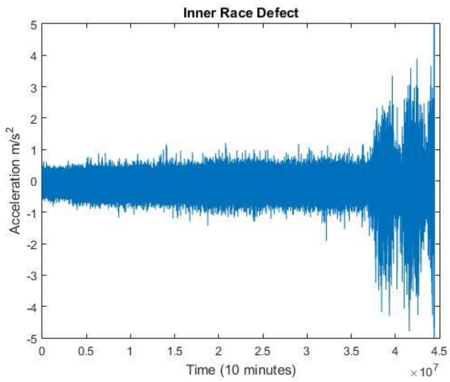


Figure 19: Vibration signal of inner race defects bearing

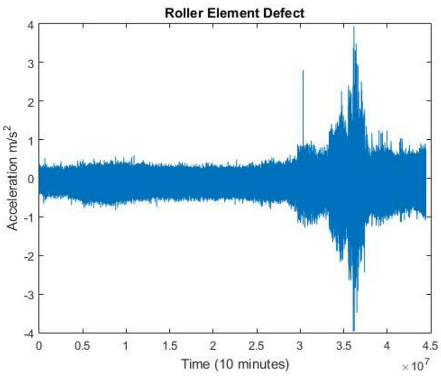


Figure 20: Vibration signal of roller element defects bearing

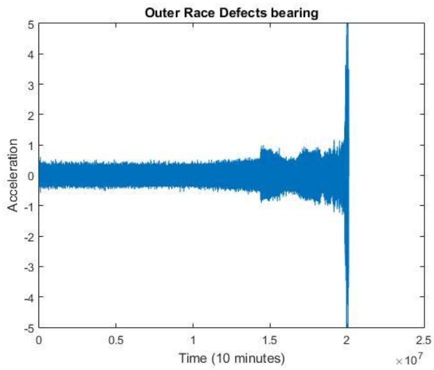


Figure 21: Vibration signal of outer race defects bearing

Based on the vibration signal in Figure 18, a healthy bearing will maintain constant acceleration. However, the defects in bearing can be identified in Figure 19 to Figure 21 from the sudden gain in acceleration. Take note that 70% of the vibration signal is used for training and remaining 30% used for testing.

4.2 Intrinsic Mode Function

Identification of the intrinsic oscillatory mode in each time location from a signal is the aim of using EMD. EMD is able to break a complex signal into numbers of simplified signal which is named as Intrinsic Mode Function. IMF signal consists of oscillatory mode in any time location. A signal is considered as IMF if the number of extrema and number of zero-crossings are the same [23]. Basically, EMD extracts different frequency component from the vibration signal into IMFs. Therefore, IMF represents the frequency component of the original signal. For example, if 20 IMFs is extracted, the 20^{th} IMF will have less important frequency compared to the first 7 IMFs (the signal at 20^{th} IMF is monotonic) because there is not much frequency component to be extracted after 19 IMFs have been extracted. Therefore, the higher number of IMF can be eliminated because it does not affect the result.

4.3 Energy Entropy of IMFs

Energy entropy of IMFs is computed after all the IMFs are successfully decomposed from the original bearing vibration signal. The energy entropy of IMF and EMD energy entropy can be computed based on the equation from (1.6) to (1.10). Based on the existing works, most of them are using the EMD to exacts IMF and then compute the energy entropy of all IMFs and EMD energy entropy. EMD energy entropy is the energy entropy of the original signal. Energy entropy is computed to further simplify the IMF signal into a feature vector. This can improve the overall performance of SLFN and training time of the SLFN.

For the preliminary result, four different classes have been used for classification problem using Single Layer Feedforward Neural Network (SLFN). The classes are healthy bearing, inner race defects, outer race defects, and roller element fault. The vibration signal representing these classes are broken up into IMF by using EMD.

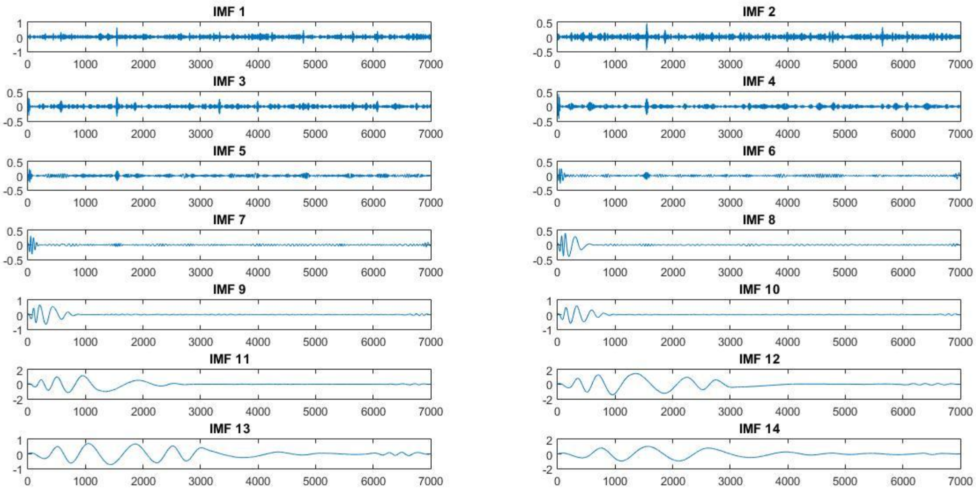


Figure 22: 14 IMFs extracted from bearing vibration signal

14 IMFs have been extracted for each class. Based on Figure 22, 14^{th} IMF signal is close to a monotonic signal. The extracted IMFs is not fit to be an input to the SLFN because the data size is too big. A IMF has 10000 data point, 14 IMFs will have 140000 data point for one class which will worsen the classification performance. Therefore, energy entropy of each IMF and energy entropy of the whole signal will be calculated to form a feature vector that will be an input to the SLFN.

$$\left[H_{en}, \frac{E_1}{E}, \frac{E_2}{E}, \frac{E_3}{E}, \dots, \frac{E_n}{E}\right] = [H_{en}, H_{en}IMF1, H_{en}IMF2, \dots, H_{en}IMFn]$$

Where n = 14 because 14 IMFs have been extracted.

Calculation formula can be referred to 1.6 to 1.10.

4.4 Classification results of each learning algorithm

To ease the classification problem, 4 bits are used to classify the four different classes of bearing:

I. [1 0 0 0] represents healthy bearing.
II. [0 1 0 0] represents Inner race defects.
III. [0 0 1 0] represents Roller element defects.
IV. [0 0 0 1] represents Outer race defects.

Table 4: Classification result using BP algorithm

BP	
Training	Testing
$\begin{bmatrix} 0.9882 & 0 & 0.0137 & 0.0001 \\ 0.0004 & 0.9739 & 0.0164 & 0.0125 \\ 0.0109 & 0.0245 & 0.9783 & 0.0001 \\ 0.0051 & 0.0094 & 0 & 0.9899 \end{bmatrix}$	$\begin{bmatrix} 0.9916 & 0 & 0.0259 & 0.0001 \\ 0.0001 & 0.8530 & 0.0105 & 0.0795 \\ 0.0010 & 0.3994 & 0.7231 & 0.0013 \\ 0.0026 & 0.0523 & 0 & 0.9927 \end{bmatrix}$
Error Rate: 0.01182	Error Rate: 0.1644
Training and Testing Time: 45.32 seconds	

Table 5: Classification result using ELM algorithm

ELM	
Training	Testing
$\begin{bmatrix} 0.9964 & 0.0014 & 0.0013 & 0.0009 \\ 0.0014 & 0.9965 & 0.0014 & 0.0007 \\ 0.0013 & 0.0014 & 0.9961 & 0.0012 \\ 0.0009 & 0.0007 & 0.0012 & 0.9972 \end{bmatrix}$	$\begin{bmatrix} 0.9855 & 0.0161 & 0.0143 & 0.0158 \\ 0.1059 & 0.9348 & 0.0209 & 0.1921 \\ 0.0551 & 0.0156 & 0.9482 & 0.0913 \\ 0.0012 & 0.01 & 0.0039 & 0.9950 \end{bmatrix}$
Error Rate: 0.0021	Error Rate: 0.0633
Training and Testing Time: 0.002225 second	

Table 6: Classification result using FIR-ELM algorithm

FIR-ELM	
Training	Testing
$\begin{bmatrix} 0.9909 & 0.0030 & 0.0028 & 0.0031 \\ 0.0030 & 0.9909 & 0.0028 & 0.0032 \\ 0.0028 & 0.0028 & 0.9909 & 0.0036 \\ 0.0031 & 0.0032 & 0.0036 & 0.9901 \end{bmatrix}$	$\begin{bmatrix} 0.9918 & 0.0024 & 0.0027 & 0.0021 \\ 0.0262 & 1.0137 & 0.0162 & 0.0177 \\ 0.0073 & 0.0097 & 0.9942 & 0.0096 \\ 0.0067 & 0.0007 & 0.0008 & 0.9847 \end{bmatrix}$
Error Rate: 0.001623	Error Rate: 0.01891
Training and Testing Time: 0.009862 second	

Table 7: Classification result using DFT-ELM algorithm

DFT-ELM	
Training	Testing
$\begin{bmatrix} 0.9971 & 0.0004 & 0.0016 & 0.0008 \\ 0.0004 & 0.9980 & 0.0005 & 0.0008 \\ 0.0016 & 0.0005 & 0.9973 & 0.0005 \\ 0.0008 & 0.0008 & 0.0005 & 0.9977 \end{bmatrix}$	$\begin{bmatrix} 0.9982 & 0.0002 & 0.0006 & 0.0010 \\ 0.0113 & 0.9767 & 0.0163 & 0.0027 \\ 0.0034 & 0.0031 & 0.9942 & 0.0002 \\ 0.0124 & 0.0313 & 0.0222 & 1.0497 \end{bmatrix}$
Error Rate: 0.001434	Error Rate: 0.01870
Training and Testing Time: 0.00419 second	

Table 8: Result comparison between 4 classifiers (algorithms)

Classifiers				
	BP	ELM	FIR-ELM	DFT-ELM
Training Error Rate	0.01181	0.0021	0.001623	0.001434
Testing Error Rate	0.1644	0.0633	0.01891	0.01870
Training and Testing Time(second)	45.32	0.002225	0.009862	0.00419

Table 4 to Table 7 show the classification result obtained using MATLAB simulation for BP, ELM, FIR-ELM and DFT-ELM respectively. In order to compare the error rate of four learning algorithms, the training error rate, testing error rate and time taken have been tabulated in Table 8. Based on Table 8, DFT-ELM achieved the lowest training error rate and testing error rate and then followed by FIR-ELM, ELM, and BP. ELM able to achieve a lower error rate compared to BP because different computation method is used. BP uses gradient descent method to optimize the input and output weight. The disadvantage of this method is that there is possibility where cost function converges into local minimum instead of global minimum. Cost function can be defined as the squared difference between target value and actual value. As cost function converges to local minimum, it is not optimized and this creates a poor input and output weight optimization. Besides, in ELM, the SLFN is treated as a linear system where the output weight can be optimized by using a generalized inverse of hidden layer output matrix. Although ELM performs better than BP, but it shown poor robustness when dealing with a input with noise disturbance. As shown in Table 8, FIR-ELM achieves a lower error rate compared to ELM because in FIR-ELM, the hidden layer is treated as a filter to filter out the noise and improves the structural risk and empirical risk. Lastly, DFT-ELM achieves the lowest training error rate and testing error rate as compared to BP, ELM and FIR-ELM because the input weight and output weight are trained with regularization theory unlike ELM and FIR-ELM.

Table 9: Mean Error Rate and Standard Deviation of each learning algorithm

	BP		ELM		FIR-ELM		DFT-ELM	
	Mean	Std. Dev	Mean	Std. Dev	Mean	Std. Dev	Mean	Std. Dev
Training	0.01181	0.00099	0.00202	0.00012	0.00162	0.00061	0.00143	9.16E-05
Testing	0.16438	0.09198	0.06325	0.00946	0.01890	0.00757	0.01869	0.00723

For further analysis, the performance comparisons of SLFN classifier trained with BP, ELM, FIR-ELM, and DFT-ELM are carried out in term of average Mean Squared Error (MSE) over 50 iterations and the standard deviated of 50 iterations has been calculated as well. Based on Table 9, BP has the highest sensitivity because it has the largest mean MSE and standard deviation and then followed by ELM, FIR-ELM and DFT-ELM. Higher standard deviation means that the error rate deviate a lot based on the 50 iteration. As a result, DFT-ELM has the lowest sensitivity which mean it has a high precision in diagnosing bearing fault, besides, DFT-ELM has the highest accuracy as well because it has the lowest mean error rate.

Table 10: Accuracy of each learning algorithm in bearing fault diagnosis

Learning algorithms	BP	ELM	FIR-ELM	DFT-ELM
Training Accuracy (%)	98.82	99.79	99.44	99.85
Testing Accuracy (%)	83.56	93.67	98.11	98.13

Table 10 shows the accuracy of bearing fault diagnosis of four learning algorithms. Based on Table 10, DFT-ELM achieves the highest accuracy, followed by FIR-ELM, ELM, and BP respectively.

4.5 Performance Comparison of Learning Algorithms

There are two variables to be examined on how they affect the performance of SLFN. These variables are number of hidden nodes, and regularization parameter as shown in Table 11. First, training error rate, testing error rate, and time taken to train the SLFN will be examine while varying the number of hidden layer.

Table 11: list of parameters

Parameters	Explanations
Hidden nodes	Hidden nodes is the intermediate layer where the input nodes are map to the hidden nodes and hidden nodes are map to the output nodes.
Regularization parameter	Regularization parameter is a tool used to overcome the overfitting problem.

4.5.1 Varying number of hidden nodes

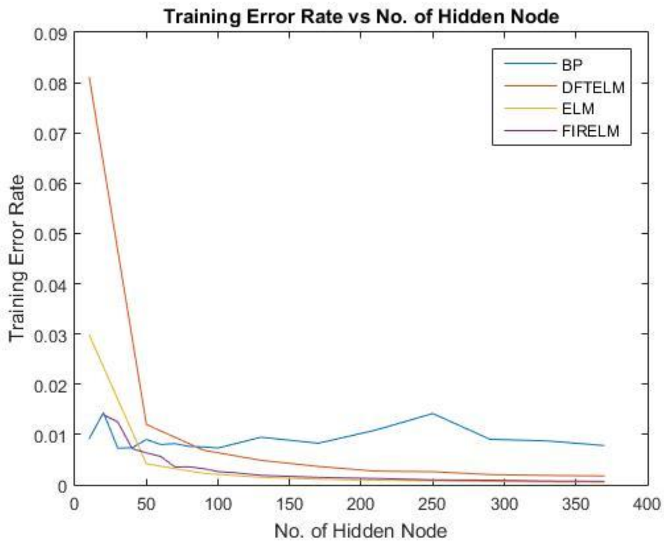


Figure 23: Changes in the training error rate when varying number of hidden nodes

As shown in Figure 23, as the number of hidden layer increases, the training error rate reduces.

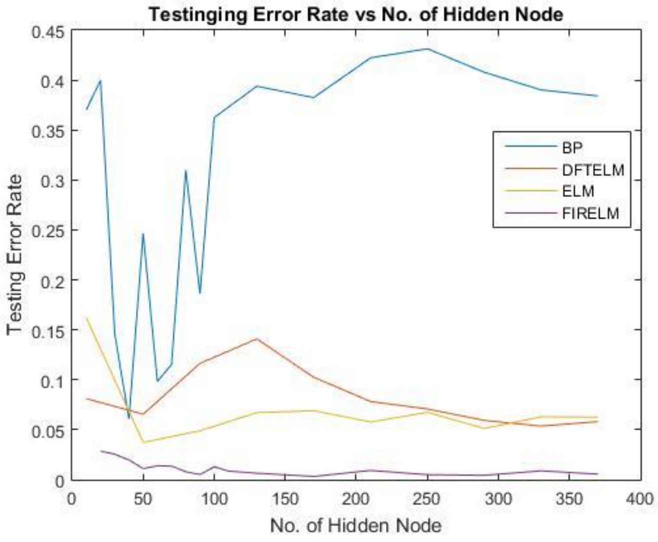


Figure 24: Changes in testing error rate when varying number of hidden nodes

As shown in Figure 24, for DFT-ELM, FIR-ELM and ELM, the testing error rate is reduced when the number of hidden node increases. However, BP will lost the classification capability when the hidden layer got bigger. This is because BP used the gradient descent method to optimize the input and output weight. So, when the hidden layer is large, the input weight and output weight matrix become large as well which will disturb the optimization process of input weight and output weight. Besides, the classification capability of DFT-ELM, FIR-ELM and ELM improved when the number of hidden nodes increase. This is due to the simple computation method is used as compared to BP.

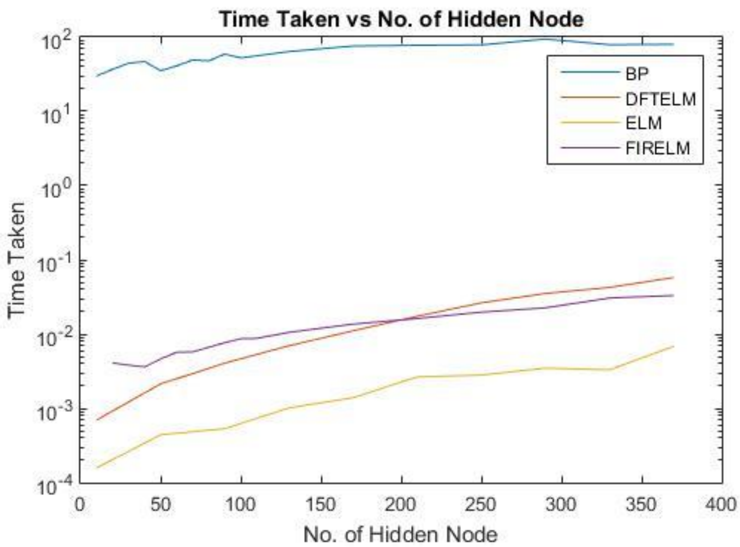


Figure 25: Time taken for a SLFN(BP) to be trained and tested when varying number of hidden nodes

Figure 25 shows that the time taken to train and test SLFN increases when the number of hidden nodes increases. BP takes the longest time to train SLFN followed by DFT-ELM, FIR-ELM and ELM. BP takes the longest time to train SLFN because it uses gradient descent method to optimize the input and output weight which requires a lot of iterations. Whereas, DFT-ELM requires a longer time to train SLFN as compared to FIR-ELM when the hidden layer is very big, because input weight in DFT-ELM need to be trained by using regularization theory but the input weight in FIR-ELM is just an assignment from the FIR filter coefficient. ELM requires the least amount of time to train and test the SLFN because it only randomly assigns the input weight within a given range without training the input weight like DFT-ELM and simple computation method as compared to BP, FIR-ELM and DFT-ELM.

4.5.2 Varying Regularization Parameter

On the other hand, by varying value of regularization parameter, it does affect the performance of the SLFN. The function of regularization parameter is to overcome overfitting problem. When the value of regularization parameter increases, it will adjust the fitting toward underfitting. Conversely, when the value of regularization parameter become smaller, it will adjust the fitting toward overfitting. The following tables, will describe how regularization parameter affects the performance of SLFN.

Table 12: Performance of BP when varying value of regularization parameter

	BP						
Regularization	0	5.00E-07	1.00E-05	0.0001	0.001	0.1	1
Training Error Rate	0.006446	0.007778	0.017508	0.047911	0.2041	0.612497	0.790569
Testing Error Rate	0.281163	0.172589	0.118071	0.187381	0.250553	0.612497	0.790569
Cost Function	2.37E-05	0.000125	0.001429	0.006737	0.047765	0.216886	0.310863
Time Taken	71.82436	47.39408	70.13692	58.34547	65.59499	38.7786	48.67443

Table 13: Performance of ELM when varying value of regularization parameter

	ELM						
Regularization	0	5.00E-07	1.00E-05	1.00E-04	0.001	0.1	1
Training Error Rate	2.27E-11	1.05E-03	0.019772	0.131847	0.35928	0.432068	0.432916
Testing Error Rate	1.102058	0.081601	0.061845	0.157224	0.363336	0.432129	0.432923
Time taken	0.001315	0.001457	0.00044	0.000453	0.000445	0.000464	0.00045

Table 14: Performance of FIR-ELM when varying value of regularization parameter

	FIR-ELM						
Regularization	0	1.00E-07	1.00E-06	1.00E-05	0.001	0.01	1
Training Error Rate	1.45E-15	1.74E-07	2.26E-06	1.86E-05	0.001925	0.017316	0.347464
Testing Error Rate	0.04054	0.02223	0.01312	0.00928	0.007	0.03099	0.349
Time taken	0.00676	0.00455	0.00471	0.00452	0.00451	0.00496	0.00468

Table 15: Performance of DFT-ELM when varying value of regularization parameter

	DFT-ELM						
Regularization	0	1.00E-07	1.00E-05	0.001	0.01	0.1	1
Training Error Rate	1.45E-14	6.57E-07	5.82E-05	0.005334	0.055768	0.238282	0.403626
Testing Error Rate	1.090047	0.073474	0.113918	0.133984	0.175884	0.255111	0.398413
Time taken	0.137262	0.001438	0.001434	0.001607	0.001617	0.001598	0.002271

In order to show the effect of overfitting and underfitting on the training and testing error rate, the above tables (Table 12 to Table 15) are used to best represent the result. As shown in the tables, as the regularization parameter and the training error rate are approaching zero, the testing error rate to increases. This is the overfitting problem, which means the SLFN fit the training data perfectly until it unable to fit the testing data. However, as the regularization parameter approaching one, the training error rate and testing error rate will increase because SLFN underfits the training data and as a result error rate become high in both training and testing. Therefore, the value of regularization parameter must be chosen properly between zero to one to obtain the high classification accuracy.

Besides that, by varying value of regularization parameter, it does not affect much on the time taken for training and testing and SLFN as shown in the tables above.

4.6 Graphical User Interface

A graphical user interface (GUI) has been created to demonstrate the selection of number of hidden layer, learning algorithm, training data and testing data. The graphical user interface is as shown in Figure 26.

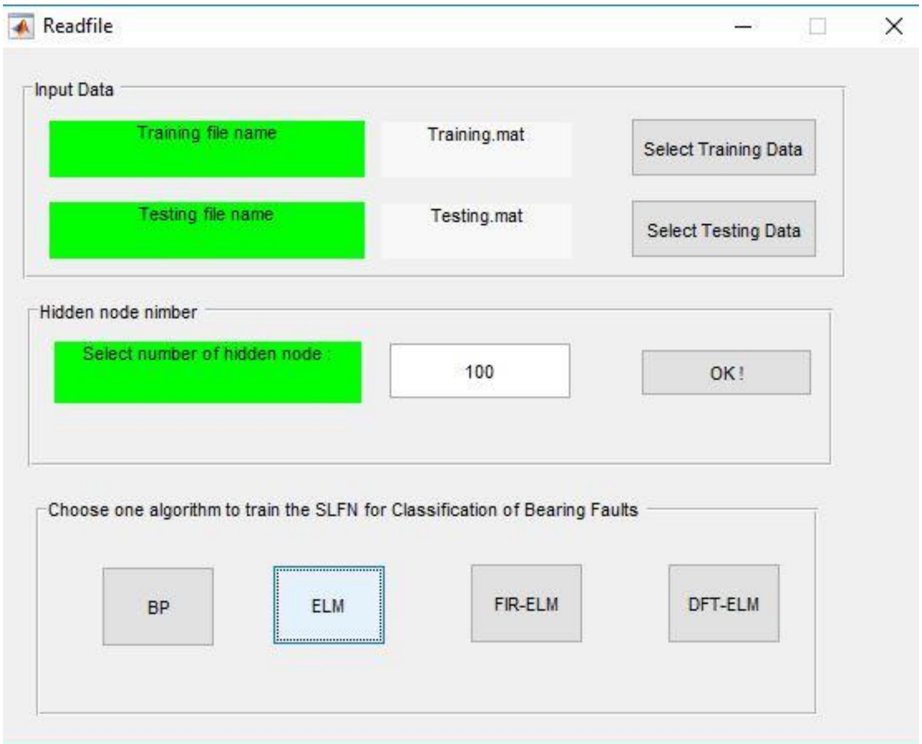


Figure 26: Graphical User Interface

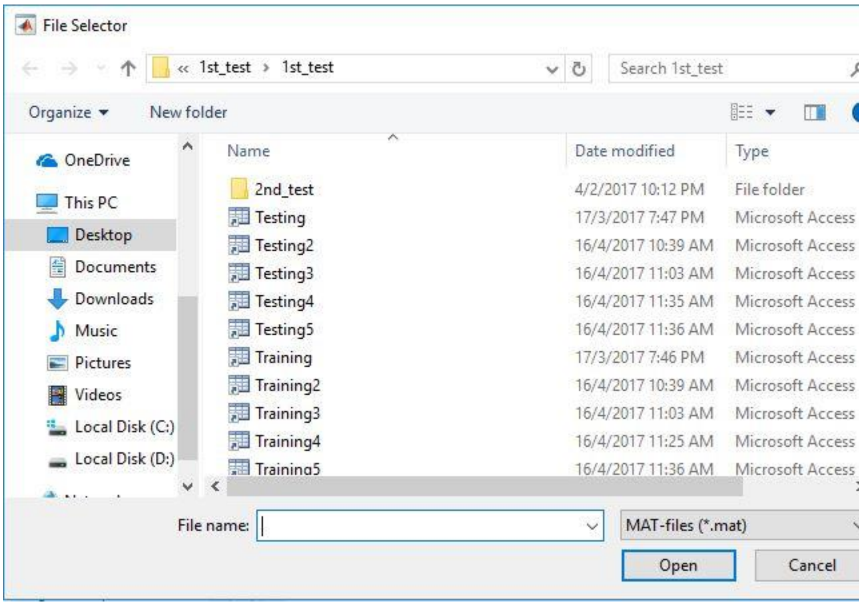


Figure 27: Selection of training data and testing data from 5 sets of data.

```
Elapsed time is 0.001780 seconds.

output =

    'HB'      [0.9957]    [     0.0011]    [     0.0019]    [     0.0014]
    'IRDB'    [0.0011]    [     0.9967]    [9.7741e-04]    [     0.0012]
    'REDB'    [0.0019]    [9.7741e-04]    [     0.9966]    [5.8172e-04]
    'ORDB'    [0.0014]    [     0.0012]    [5.8172e-04]    [     0.9968]


output_testing =

    'HB'      [ 0.9988]    [-0.0082]    [ 0.0080]    [0.0015]
    'IRDB'    [-0.0666]    [ 0.9419]    [ 0.0334]    [0.0911]
    'REDB'    [-0.0156]    [ 0.0343]    [ 0.9316]    [0.0496]
    'ORDB'    [-0.0096]    [ 0.0501]    [-0.0176]    [0.9771]


TrainingErrorRate =

    0.0021


TestingErrorRate =

    0.0428
```

Figure 28: Example Result

HB represents healthy bearing.

IRDB represents inner race defect bearing.

REDB represents roller element bearing defect.

ORDB represents outer race defect bearing.

As shown in Figure 27, there are 5 sets of data for training and testing the SLFN. The performance is evaluated in the following Figures.

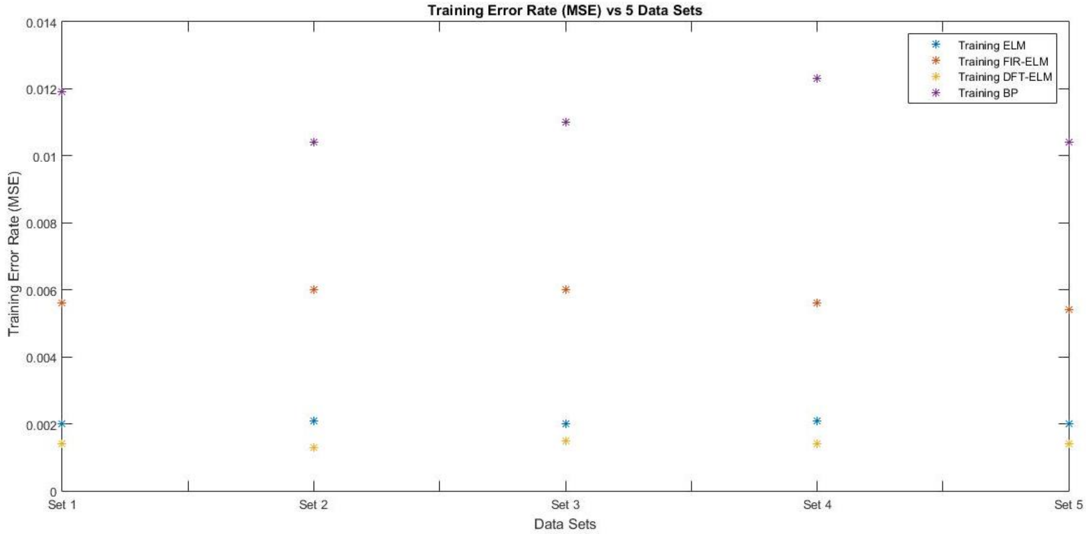


Figure 29: Training error rate of four learning algorithms over 5 different data sets

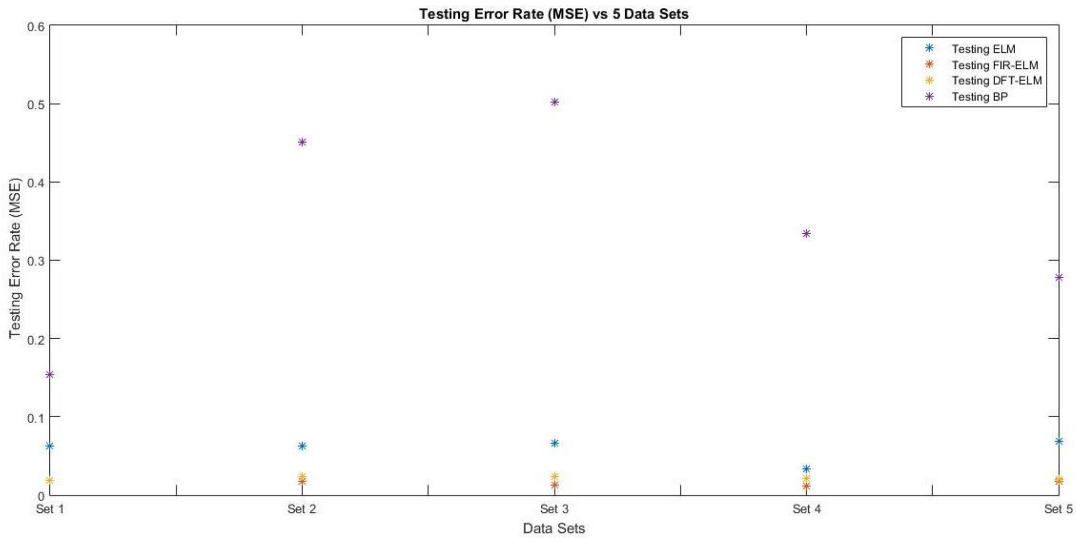


Figure 30: Testing error rate of four learning algorithms over 5 different data sets.

Based on Figures 29 and Figure 30, DFT-ELM has the lowest training error rate and followed by ELM, FIR-ELM and DFT-ELM. On the other hand, FIR-ELM has slightly lower testing error rate as compare to DFT-ELM. ELM ranked as the third lowest testing error rate and BP is the highest testing error rate. The testing error rate of BP deviate a lot over the 5 different data set. Therefore, BP is very sensitive therefore, BP is less robust as compared to ELM, FIR-ELM and DFT-ELM. As shown in Figure 30, the testing error rate of ELM, FIR-ELM and DFT-ELM only have a little deviation over the 5 sets of data. In conclusion, SLFN trained by DFT-ELM and FIR-ELM have high robustness as compared to SLFN trained by ELM and BP.

CHAPTER 5

Conclusion and Recommendation

5.1 Conclusion

Four types of bearing faults have been successfully classified by SLFN trained by four algorithms BP, ELM, FIR-ELM and DFT-ELM. EMD is used as a feature extraction method to extract the Intrinsic Mode Function (IMF) from bearing vibration signal. Next, the energy entropy of each IMF and energy entropy of the whole signal are computed to form a feature vector. This feature vector is used as an input to SLFN. The combination between EMD and SLFN techniques trained by BP, ELM, FIR-ELM and DFT-ELM have been explored and the result shows that it improves the bearing fault diagnosis. Based on the result, DFT-ELM gives the highest classification accuracy and then followed by FIR-ELM, ELM and BP. The findings of this work also prove that BP takes the longest time to train the SLFN. The key contributions of this work are DFT-ELM, FIR-ELM and ELM improved the bearing fault diagnosis accuracy as compared to BP.

5.2 Recommendation

In future work, this work will be used for other application such as medical diagnosis, tidal prediction, gear box fault diagnosis and stator fault diagnosis. Besides, more data will be used to evaluate the performance of SLFN trained by BP, ELM, FIR-ELM and DFT-ELM in the future.

Reference

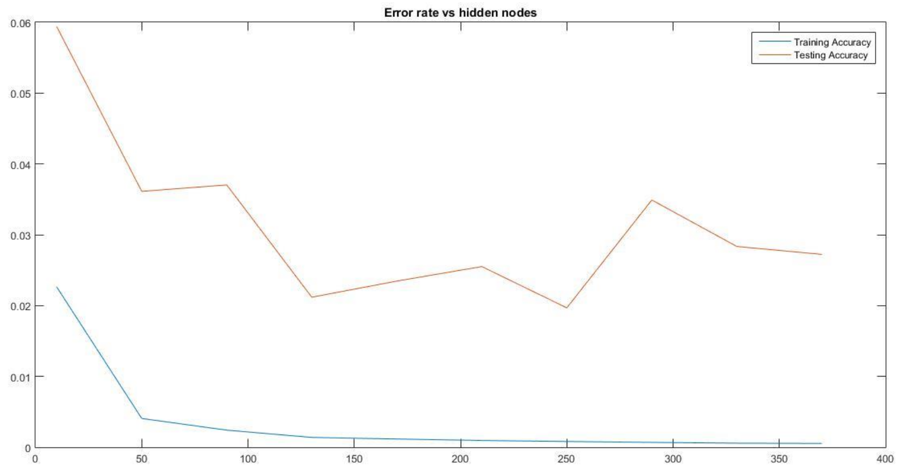
1. Ali, J. B., et al. (2015). "Application of empirical mode decomposition and artificial neural network for automatic bearing fault diagnosis based on vibration signals." Applied Acoustics **89**: 16-27.
2. Andrew. Ng, (n.d). Machine Learning Tutorial. Retrieved from https://www.coursera.org/.
3. Andrew. Ng, J.Q. Ngiam, C.Y. Foo, Y.F. Mai, Caroline Suen. (n.d). Machine Learning Course: Deep learning tutorial: Back Propagation algorithm. Retrieved from http://deeplearning.stanford.edu /wiki/index.php/ UFLDL_Tutorial
4. Ding, S., et al. (2015). "Extreme learning machine: algorithm, theory and applications." Artificial Intelligence Review **44**(1): 103-115.
5. Huang, G.-B., et al. (2006). "Extreme learning machine: theory and applications." Neurocomputing **70**(1): 489-501.
6. IMS bearing data; 2013. <https://ti.arc.nasa.gov/tech/dash/pcoe/prognostic-data-repository/#bearing>
7. Kankar, P., et al. (2011). "Fault diagnosis of ball bearings using machine learning methods." Expert Systems with Applications **38**(3): 1876-1886.
8. Man, Z., et al. (2012). "Robust single-hidden layer feedforward network-based pattern classifier." IEEE transactions on neural networks and learning systems **23**(12): 1974-1986.
9. Man, Z., et al. (2011). "A new robust training algorithm for a class of single-hidden layer feedforward neural networks." Neurocomputing **74**(16): 2491-2501.
10. S. Haykin, *Neural Networks and Learning Machines*, 3rd ed. Englewood Cliffs, NJ: Prentice-Hall, 2009.
11. S. Haykin, *Adaptive Filter Theory*, 3rd ed. Englewood Cliffs, NJ: Prentice-Hall, 1996.
12. Wang, C.-C., et al. (2010). "Applications of fault diagnosis in rotating machinery by using time series analysis with neural network." Expert Systems with Applications **37**(2): 1696-1702.

13. Yu, Y. and C. Junsheng (2006). "A roller bearing fault diagnosis method based on EMD energy entropy and ANN." Journal of sound and vibration **294**(1): 269-277.
14. Dybala J, ZImroz R. Rolling bearing diagnosing method based on empirical mode decomposition of machine vibration signal. Appl Acoust 2014;77:195-203.
15. Zheng J, Cheng J, Yang Y. Generalized empirical mode decomposition and its application to rolling element bearing fault diagnosis, Mech Syst Signal Process 2013;40:136-53
16. FernáNdez-Francos, D., et al. (2013). "Automatic bearing fault diagnosis based on one-class ν-SVM." Computers & Industrial Engineering **64**(1): 357-365.
17. Zhang Y, Randall RB. Rolling element bearing fault diagnosis based on the combination of genetic algorithms and fast kurtogram. Mech Syst Signal Process 2009;23:1509–17.
18. Qiu H, Lee J, Lin J, Yu G. Wavelet filter-based weak signature detection method and its application on rolling element bearing prognostics. J Sound Vib 2006;289:1066–90.
19. Xu H, Chen G. An intelligent fault identification method of rolling bearings based on LSSVM optimized by improved PSO. Mech Syst Signal Process 2013;35:167–75.
20. Tandon N, Choudhury A. A review of vibration and acoustic measurement methods for the detection of defects in rolling element bearings. Tribol Int 1999;32:469–80.
21. Stergiou.C, Siganos.D.(n.d). Neural Networks. Retrieved from https://www.doc.ic.ac.uk/~nd/surprise_96/journal/vol4/cs11/report.html
22. An, X. and D. Jiang (2014). "Bearing fault diagnosis of wind turbine based on intrinsic time-scale decomposition frequency spectrum." Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability **228**(6): 558-566.
23. N.E. Huang, Z. Shen, S.R. Long, M.C. Wu, H.H. Shih, Q. Zheng, N.-C. Yen, C.C. Tung, H.H. LiuThe. Empirical Mode Decomposition and the Hilbert Dpectrum for nonlinear and non-stationary time series analysis.

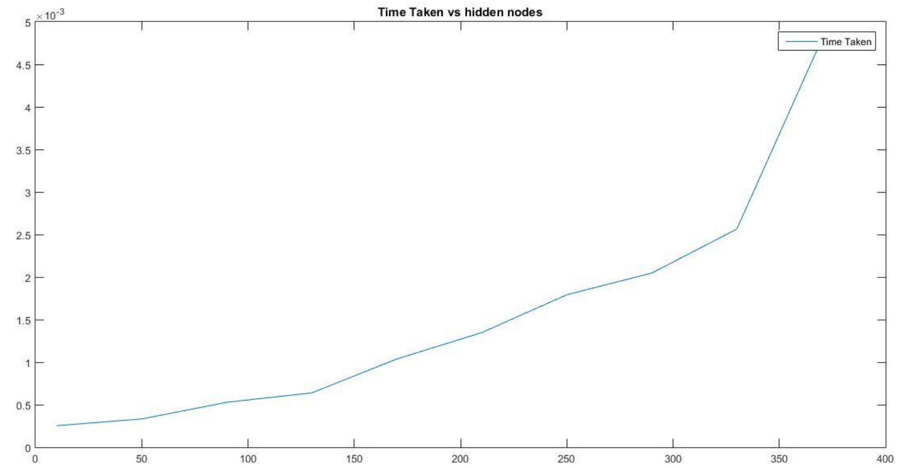
APPENDICES

Appendix A

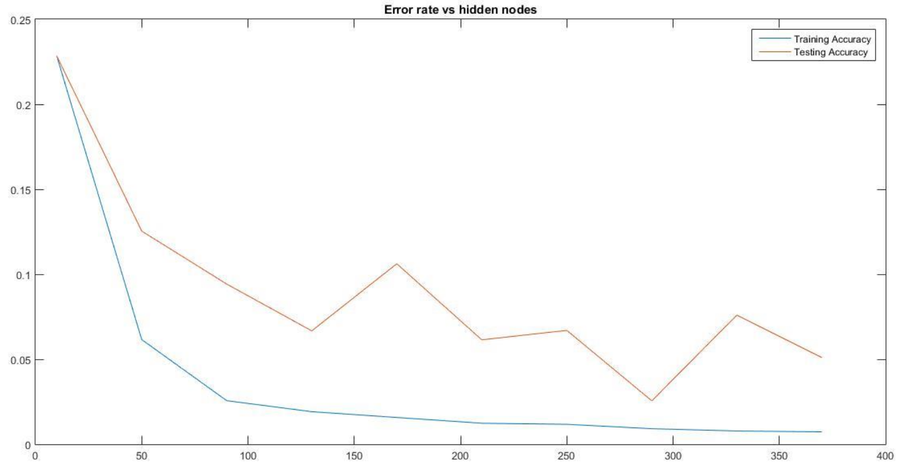
I. Error rate of ELM reduces as hidden node increases.



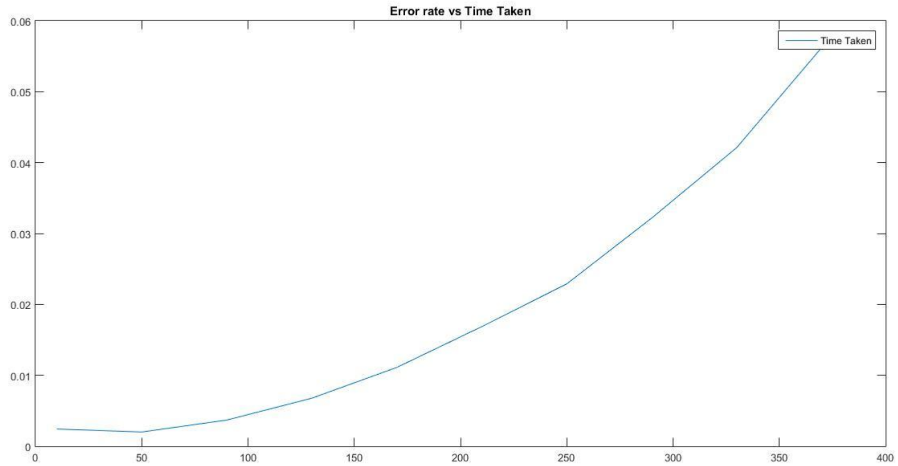
II. Time taken to train and test SLFN using ELM increases as number of hidden nodes increases.



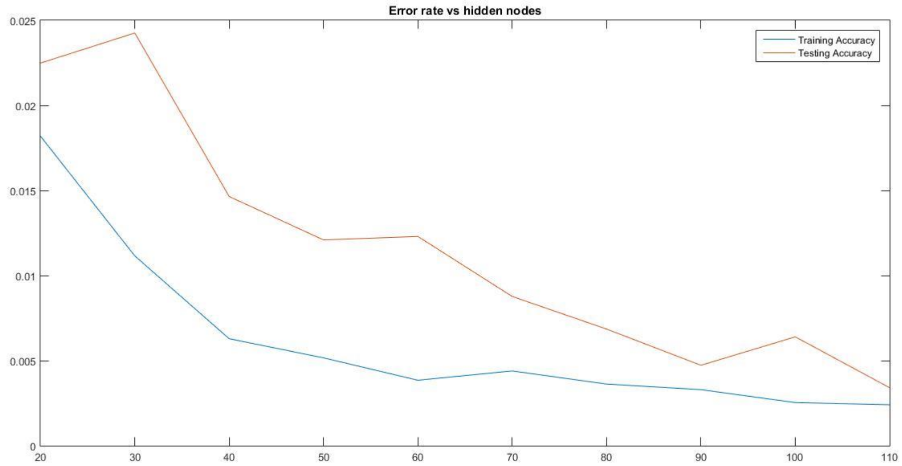
III. Error rate of DFT-ELM reduces as number hidden node increases.



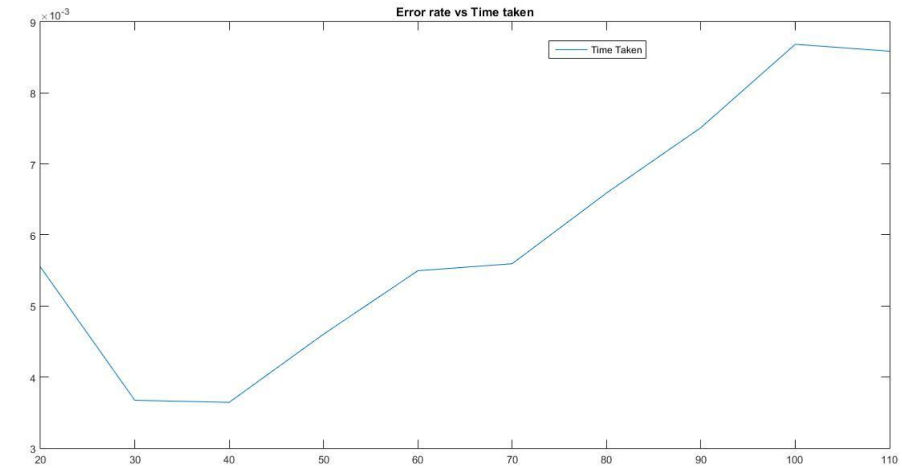
IV. Time taken to train and test SLFN using DFT-ELM increases as number of hidden nodes increases.



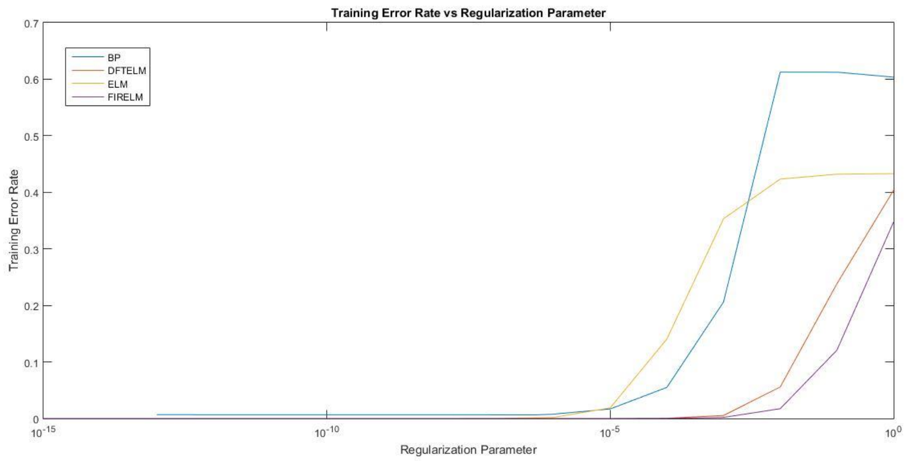
V. Error rate of FIR-ELM reduces as number hidden node increases.



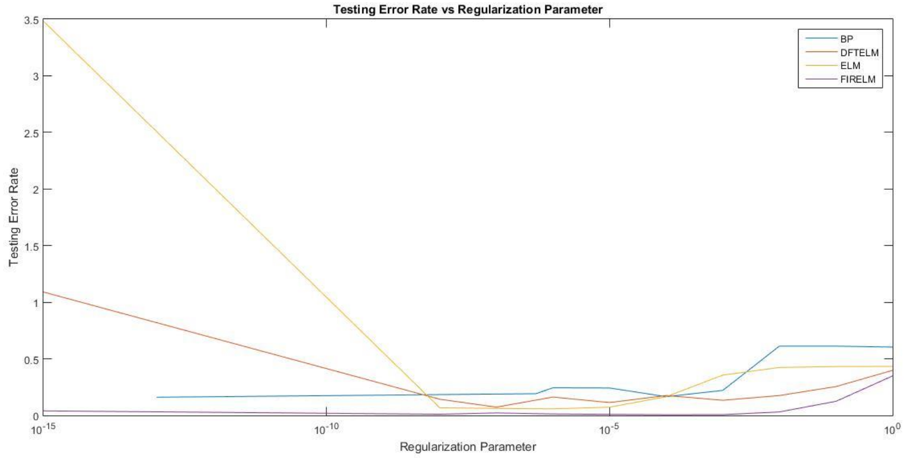
VI. Time taken to train and test SLFN using FIR-ELM increases as number of hidden nodes increases.



VII. Training error rate increases as value of regularization parameter increases.



VIII. Value of regularization parameter must be chosen properly to avoid underfitting and overfitting problem.



Appendix B

I. Graphical User Interface