Functional principal component analysis of vibrational signal data: A functional data analytics approach for fault detection and diagnosis of internal combustion engines

By

Justin Blake McMahan

A Thesis
Submitted to the Faculty of
Mississippi State University
in Partial Fulfillment of the Requirements
for the Degree of Master of Science
in Industrial and Systems Engineering
in the Bagley College of Engineering

Mississippi State, Mississippi

December 2018

Copyright by

Justin Blake McMahan

2018

Functional principal component analysis of vibrational signal data: A functional data analytics approach for fault detection and diagnosis of internal combustion engines

By

Justin Blake McMahan

Approved:

_____

Brian Smith
(Major Professor)

_____

Mohammad Marufuzzaman
(Committee Member)

_____

Junfeng Ma
(Committee Member)

_____

Ruholla Jafari Marandi
(Committee Member)

_____

Stanley F. Bullington
(Graduate Coordinator)

_____

Jason M. Keith
Dean
Bagley College of Engineering

Name: Justin Blake McMahan

Date of Degree: December 14, 2018

Institution: Mississippi State University

Major Field: Industrial and Systems Engineering

Select Appropriate Title: Brian K. Smith

Title of Study: Functional principal component analysis of vibrational signal data: A functional data analytics approach for fault detection and diagnosis of internal combustion engines

Pages in Study 89

Candidate for Degree of Master of Science

Fault detection and diagnosis is a critical component of operations management systems. The goal of FDD is to identify the occurrence and causes of abnormal events. While many approaches are available, data-driven approaches for FDD have proven to be robust and reliable. Exploiting these advantages, the present study applied functional principal component analysis (FPCA) to carry out feature extraction for fault detection in internal combustion engines. Furthermore, a feature subset that explained 95% of the variance of the original vibrational sensor signal was used in a multilayer perceptron to carry out prediction for fault diagnosis. Of the engine states studied in the present work, the ending diagnostic performance shows the proposed approach achieved an overall prediction accuracy of 99.72 %. These results are encouraging because they show the feasibility for applying FPCA for feature extraction which has not been discussed previously within the literature relating to fault detection and diagnosis.

DEDICATION

I would like to dedicate this thesis to my parents and to my wife, Juliana. I know that none of this would have been possible without your constant love and support.

ACKNOWLEDGEMENTS

I would like to express gratitude to Jafarian et al for allowing us to use their dataset which was the subject for the work of this thesis. This gesture allowed us to develop new approaches for fault detection and diagnosis and contribute to the field of research.

Likewise, I would like to thank my committee, especially my advisor and committee chair, Brian K. Smith. While I was eager to accomplish my work, your direction, mentorship, and support helped to make the process easy and enjoyable.

I would also like to acknowledge Ruholla Jafari Marandi as a key resource for the work. He was able to identify this project for me and provide advice and suggestions which helped me identify my methodology and interpret my results.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER I

INTRODUCTION: MOTIVATION FOR FAULT DETECTION AND DIAGNOSIS

Since the advent of industrialization, humans have inherently been concerned about the functional states of their man-made machines. Initially, the process of acquiring information pertaining to the functional state of the machine was acquired through biological senses (e.g. visual inspection for any changes in shape or color, auditory inspection for any unique changes in the sound of the intensity or pitch of the machine, touching the machine to sense for excess vibration or heat, and smelling for fumes from leaks or overheating) (Gertler, 2013). These initial "sensor data" were able to provide rough insights into necessary decisions that needed to be made for fault detection and diagnosis (FDD).

Building on the biological sensory capability of humans, technological advances resulted in the development of an array of sensor devices that can measure aspects of machines and equipment that are outside the realm of biological sensory capability. Likewise, the information provided by these sensors is more plentiful, exact, and can be used as support for greater decision making.

1.1 Faults in Industrial Processes

In hierarchical engineering systems composed of processes, sub-processes, individual machines, and equipment, fault detection is a main concern. Faults can exist

anywhere within the system and can be directly associated with the technical equipment for facilities' direct operations or with the associated measurement and control equipment. Specifically, a fault is characterized as a deviation from the normal operating behavior in the machine, sub-process, process, or system of interest. As mentioned previously, faults can exist anywhere within the overall system. For convenience, the following classifications have been provided for faults (Gertler, 1998):

1. Additive process faults: These faults are the result of unknown inputs. When present, these unknown inputs cause an output response that is independent of the inputs. A leak is a classic example of an additive process fault.
2. Multiplicative process faults: These faults arise from gradual or abrupt changes in some of the operating parameters for the system of interest. The resulting change in output caused by these faults is dependent on the magnitude of the known inputs.
3. Sensor faults: These faults represent discrepancies between the measured and actual values of individual system variables. The effect of these faults can be either additive or multiplicative.
4. Actuator faults: These faults represent discrepancies between the input command of an actuator and its actual output. These faults are usually handled as additive but occasionally are better characterized as multiplicative.

Through the development of FDD systems, condition-based maintenance of engineering systems can be achieved. As the name implies, fault detection and fault diagnosis are the major components of FDD systems. With fault detection, the goal is to identify that something in the system is causing it to behave abnormally when compared to a healthy, baseline operating condition. Once a fault is detected, fault isolation, the first

stage of fault diagnosis, is performed to determine the specific location of the fault. Lastly, fault identification, the last stage of fault diagnosis, quantifies the magnitude and identity of the fault.

1.2 Benefits of FDD in Industrial Processes

A multitude of industries rely on FDD techniques to satisfy the demand for higher efficiency, performance, reliability, and safety. With the use of FDD techniques, industries can obtain detailed information on the operational system performance and carry out condition-based monitoring schemes. Specifically, FDD techniques represent a central component of abnormal event management which involves the timely detection of abnormal events. Once detected, the causes of the events are diagnosed. Appropriate diagnoses of specific faults support the decision making process to initiate countermeasures to bring the system of interest back to its normal, safe operating state (Al-Sheikh & Moubayed, 2012).

1.3 Data Driven Approach for FDD

While different approaches exist for carrying out FDD, data-driven methods for carrying out statistical process monitoring have been studied extensively and accommodate system complexity with unknown variables better than other approaches (e.g. model-based FDD approach). As well, data-driven approaches do not require perquisite mathematical models of the system and only rely on historical process data. Furthermore, the statistical nature of many processes' behaviors is well suited to be the target of data analytics for process monitoring. Influenced by the data-driven approach

for FDD, various multivariate statistical process monitoring approaches that utilize hallmark methods such as principal component analysis (PCA) and support vector machines (SVM) have been developed to further investigate their detection and diagnostic applicability (Lei, Tajammal Munir, Bao, & Young, 2016).

CHAPTER II

BACKGROUND

2.1 Data Analytics Components of FDD: Feature Extraction, Feature Selection, and Classification

Many FDD approaches utilize machine learning techniques to carry out prediction of a process's state. Prediction is dependent on having input data that adequately differentiates the dependent possible output responses. Likewise, having redundant input features that do not add to the discriminatory capability of the machine learning prediction model can increase the computational cost of the FDD approach thus making it less efficient (Kannan, 2016). In a standard FDD approach, the following steps are generally employed: feature extraction, feature selection, and fault classification/prediction.

2.1.1 Feature Extraction

Feature extraction represents one of the most essential parts of intelligent classification systems. Feature extraction deals with the transformation of data from its original space into a new feature space. Generally, these transformations can be a linear combination of continuous features which have good discriminatory power between classes. Enhancing the between-class discriminatory power provides a simpler, yet powerful representation of the original data in terms of class discriminatory capability (Khalid, Khalil, & Nasreen, 2014). Classic examples of feature extraction techniques include PCA (or some variant) as well as Linear Discriminant Analysis (LDA).

2.1.2 Feature Selection

High dimensional data poses problems for classification algorithms due to the presence of potential features that may be irrelevant, misleading, or redundant. The presence of these features increases the search space size in effect increasing the computational cost (Khalid et al., 2014). To address these extraneous features, feature selection techniques can be used.

Feature selection is the process of selecting the best features among all the classes that have strong between-class discriminatory power and omitting features that do not significantly contribute to discrimination. Techniques are generally classified as being a filter method, a wrapper method, or a hybrid method that combines components from filter and wrapper methods.

With a filter method, some ranking is applied over the features. This ranking represents the "usefulness" of each feature for classification. Once the features are ranked, a feature subset containing the best N features is extracted. The wrapper method, as the name suggests, "wraps" a classifier into a feature selection algorithm. Once a set of features is chosen, efficacy is determined for that set. A perturbation is then made, and the new set's efficacy is evaluated. The drawback of this approach is that a vast dimensional space would require a large run time to look at every possible combination of features. Thus, search heuristics are employed to determine an optimum feature set (Shardlow, 2016).

2.1.3 Classification

Supervised classification, defined as the task of predicting a function/response based on the input of labeled predictor data, is the task most frequently carried out by intelligent systems in predictive data mining and analytics applications (Sotiris B.

Kotsiantis, 2007). As mentioned, the goal of classification techniques is to develop a model that produces a distribution of class labels based on corresponding sets of features. These features are obtained after feature extraction or selection techniques to allow the classification model to perform more efficiently. The resulting distribution is then used to assign class labels to testing instances where the input predictor variables are known and the output labels are unknown (S. B. Kotsiantis, Zaharakis, & Pintelas, 2006). To carry out supervised classification, several techniques have been developed over the years. These techniques are based on artificial intelligence (logic-based techniques and perceptron-based techniques) and statistics (Bayesian networks and instance-based learning).

2.2 Functional Data Analytics and FDD

In a generalized definition, functional data analytics (FDA) describes the use of statistical methods for analyzing functional data. Functional data refers to data that is initially recorded as discrete observations that were collected across some continuum (temporal, spatial, etc.). Often, these types of data are initially viewed as multivariate since they are collected as a series of discrete observations. However, this approach completely ignores important information about the smooth functional behavior of the generating process underlying the entire series of data across its respective continuum (Ullah & Finch, 2013).

Thus, the goal of FDA is to express discrete observations that arise from a continuous series in the form of a single function that is representative of the entire series of data. From a collection of these functions, modeling and prediction information can be deduced and utilized in additional analysis. Advantages of an FDA approach include

reducing noise through curve smoothing, accommodating data with irregular sampling intervals, and providing more accurate estimates of curve parameters (Ullah & Finch, 2013). Using the functional data, analogues of multivariate statistical methods are carried out in the functional space of the functional data in contrast to the space spanned by the vectors in individual observations for multivariate data (Viviani, Grön, & Spitzer, 2005).

2.3 Internal Combustion Engines as a Subject for FDD

Internal combustion engines (ICEs) represent a well-studied system with regards to FDD and will be the focus of the proposed work. ICEs have transformed the transportation industry and have become integral components in many industrial processes and systems. Since the beginning with the basic configuration of a reciprocating ICE being introduced in the late 1800s, ICEs have improved significantly in terms of thermal efficiency, emissions levels, and overall reliability. Likewise, research in these areas is still ongoing since the improvements made towards efficiency and reliability are continuous (Ferguson & Kirkpatrick, 2015).

To satisfy the demand for higher performance, efficiency, reliability, and safety, FDD techniques have been researched as a tool for abnormal event management and condition-based monitoring in ICEs. A multitude of faults can occur in an ICE but when concerning emission levels and reliability, two of the most studied faults include cylinder misfire and abnormal valve clearance.

2.3.1 Cylinder Misfire

Misfiring is an abnormal condition for healthy ICEs. Cylinder misfires result from a lack of combustion during the engine's power stroke. The combustion within the cylinder is dependent on the air-fuel (A/F) ratio. Maintaining the proper A/F ratio enables the engine components such as the catalytic converter to operate with optimized efficiency helping to reduce exhaust emissions (Kim, Han, & Moon, 2000). The lack of combustion that accompanies cylinder misfires is usually a downstream response of a damaged ignition unit, poor fuel metering, poor compression, and so on (Lindemann et al., 2000).

When misfiring occurs, the A/F ratio is altered to an extent that results in the reduction of efficiency for the catalytic converter. Likewise, engine output can reduce by 25 % (Sharma, Sugumaran, & Babu Devasenapati, 2014a). Operating at this lower efficiency damages the catalytic converter and increases exhaust emissions. Due to the widespread use of IC engines, the collective environmental consequences of cylinder misfires have garnered the attention of governmental regulatory agencies. The agencies (e.g. California Air Resources Board) focus on regulations that require on-board monitoring of component faults that could result in increased emissions. These regulations motivate the research into FDD for cylinder misfires.

2.3.2 Abnormal Valve Clearance

Valve clearance refers to the total clearance in the timing gears of the engine. Valve clearance is critical because it accommodates for the changes in the linear dimensions of the timing gear elements due to thermal expansion. Clearance values differ on the basis of factors such as engine type, cooling mode, and timing gear design, and a given clearance value is determined experimentally (Krzywonos, 2015). As a rule of thumb, clearance

values should be minimized while allowing enough space for the valves to close tightly during engine operations

Abnormal clearance values tend to develop as a result of component wear such as the camshaft and/or fault adjustment during engine maintenance (Ftoutou, Chouchane, & Besbès, 2012). A decline in engine performance and reliability accompanies valve clearance faults initially and as the fault persists, severe malignant failures such as valve fractures and cylinder hit faults can occur (Jiang, Mao, Wang, & Zhang, 2017).

CHAPTER III

LITERATURE REVIEW

3.1 FDD in ICEs: Feature Extraction, Feature Selection, and Classification

As previously mentioned, feature extraction and selection are critical elements in data-based FDD techniques. Once appropriate feature extraction and selection techniques are determined and carried out, the resulting subset of features are input into a classification model to diagnose the class corresponding with input features.

3.1.1 Feature Extraction for ICE FDD

The first step is to apply a feature extraction technique to data acquired from the engine. The data often exists in the form of a signal which can be of an acoustic or vibrational origin for example. Likewise, these signals can also correspond with current and voltage signals or performance and condition monitoring signals.

Among the current approaches, vibrational signal processing of rotary machines and ICEs has emerged as the most popular and effective method for analyzing diagnostic fault features. Table 3.1 below provides information on feature extraction techniques used to process vibrational signals for fault diagnosis.

Table 3.1 Feature Extraction Techniques for Vibrational Signal Data

Source	Sample/Study Description	Results
(Farajzadeh-Zanjani, Razavi-Far, Saif, & Rueda, 2016)	A wavelet packet transform is used for frequency domain feature extraction. Linear discriminant analysis is then used to reduce the features to a smaller set. The feature subset is used in a fault classification module.	Experimental results verified the effectiveness of the proposed technique for the diagnosis of multiple bearing defects present in induction motors.
(P. Liu, Li, Han, & Wang, 2016)	A novel feature extraction technique (two-dimensional nonnegative matrix factorization) is used in tandem with a hybrid wrapper-filter feature selection scheme for engine fault diagnosis.	Experimental results demonstrated that the proposed feature extraction and selection scheme achieved satisfactory classification performance.
(Zhan, Shi, Shwe, & Wang, 2007)	Wavelet analysis is coupled with principal component analysis to extract distinctive features for fault diagnosis. Extracted features are input into a support vector machine for fault diagnosis.	Proposed method detects and classifies faults accurately.
(Saravanan, Cholairajan, & Ramachandran, 2009)	Statistical analysis of vibration signals was used for feature extraction and a decision tree was used for feature selection. A fuzzy classifier was used on the set of selected features to classify the various gearbox faults.	Results showed a great potential and strong ability for the proposed method to classify and identify machinery faults.
(Unal, Onat, Demetgul, & Kucuk, 2014)	Feature extraction was accomplished by Hilbert and Fast Fourier Transform, and a proposed artificial neural network (ANN) was used for fault classification.	The resulting feature extraction and fault classification network provided acceptable classification accuracy.
(A. Moosavian, Najafi, Nadimi, & Arab, 2017)	Fast Fourier Transform was used for feature extraction from vibration signals collected from a sensor on an ICE. Features were input into an ANN to estimate engine friction.	The results showed that the prediction of engine friction based on the extracted features from vibratory signals had an accuracy of 91.77% with a RMSE of 0.0523 bar.

3.1.2 Feature Selection for ICE FDD

After a specific feature extraction technique is used to process the raw vibrational signals, a feature selection technique may be used if there is a need to reduce the extracted feature set to ultimately enhance the performance of a classification algorithm for which the feature subset will be fed into. Table 3.2 below provides some current examples of feature selection techniques.

Table 3.2 Common Feature Selection Techniques

Source	Feature Selection Technique Used
(Sugumaran, Muralidharan, & Ramachandran, 2007)	Decision Tree
(Malhi & Gao, 2004)	Principal Components Analysis
(Breheny & Huang, 2011)	Penalized Regression by Coordinate Descent
(Jafari-Marandi, Khanzadeh, Smith, & Bian, 2017)	Self-Organizing Maps
(Jack & Nandi, 2000)	Optimization-Based Approach
(Chaves et al., 2009; Zhou & Wang, 2007)	Statistical Analysis

3.1.3 Classification for ICE FDD

The diagnostic power of the extracted and selected features is realized when they are used as input into a classification algorithm. An array of classification techniques exists with differing foundational bases (logic, perceptron, and statistical). Table 3.3 below provides an at-a-glance overview of popular classification techniques.

Table 3.3 Popular Classification Techniques

Algorithm Class	Approach Used	Source
Logic-Based	Decision Trees	(Saravanan et al., 2009)
	Learning Set of Rules	(Muralidharan & Sugumaran, 2013)
Perceptron-Based	Single-Layered Perceptron	(Knerr, Personnaz, & Dreyfus, 1992)
	Multi-Layered Perceptron or Artificial Neural Networks	(J. Da Wu & Liu, 2008)
	Radial Basis Function (RBF) Networks	(Wuxing, Tse, Guicai, & Tielin, 2004)
Statistical Learning	Naïve Bayes Classifiers	(Muralidharan & Sugumaran, 2012)
	Bayesian Networks	(Aminian & Aminian, 2001)
Instance-Based	k-Nearest Neighbor	(Seshadrinath, Singh, & Panigrahi, 2014)
Support Vector Machines		(Saimurugan, Ramachandran, Sugumaran, & Sakthivel, 2011)

CHAPTER IV

PROPOSED THESIS

4.1 Extension of Work

The work of this thesis will extend the efforts made by Jafarian et al (Jafarian, Mobin, Jafari-marandi, & Rabiei, 2018). Specifically, the work by Jafarian et al studied a collection of fault states in combustion engines. They analyzed isolated cylinder misfire faults in two different locations as well as analyzing these faults in combination. Likewise, they also analyzed faults associated with abnormal valve clearances. This approach resulted in four unique fault states. Additionally, the group also analyzed the operational behavior of an engine with no faults ultimately bringing the number of studied states to five.

Vibrational signal data was collected under the aforementioned operational conditions. The vibrational signal was collected on four different one-direction, piezo electric, CTC accelerometers that were placed on different locations of the engine. The original raw signal data underwent preprocessing to filter excess noise before undergoing fast Fourier Transform (FFT) to convert the signal to the frequency domain for feature extraction. The extracted features were the first two dominant frequency and amplitude pairs for each sensor that collected each raw signal. Additionally, Eigenvalue analysis was conducted on the signals to provide additional features.

Statistical analysis was performed to select a subset of features from the extracted features that possessed good discriminatory power for classification of engine fault types. These features were then used as an input set of data for three different classification techniques: ANN, SVM, and kNN. A comparison of the diagnostic accuracy to similar efforts in the literature proved the validity of the proposed method (Jafarian et al., 2018).

4.2 Proposed Work

The aim of this thesis is to expand the work of Jafarian et al. Specifically, we aim to study the applicability of a functional data analytics approach (e.g. functional principal component analysis) for feature extraction using the raw vibrational signal data obtained by Jafarian et al. We will then assess the adequacy of using functional principal component analysis for feature extraction and selection by feeding the features into a multilayer perceptron for classification. Figure 4.1 below depicts a flow chart that describes the total approach for this thesis.

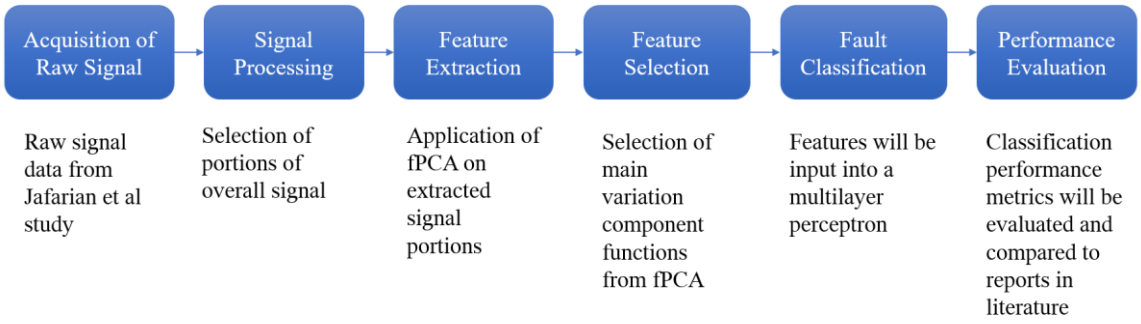

Figure 4.1 FDD framework for proposed thesis

To make the case for the importance of a functional data analytics approach, we first need to understand what is meant by functional data analytics. As previously

mentioned in 2.2, functional data analytics (FDA) describes the use of statistical methods to analyze sequences of data that can be represented by functional curves in contrast to a series of discrete observations. The analytics performed on functional data are extensions of the procedures used in the multivariate domain. One of the most popular multivariate data analytics procedure is principal component analysis.

Principal component analysis (PCA) represents a well-studied feature extraction technique in the multivariate domain. Extending PCA to the functional domain results in functional PCA (FPCA). FPCA represents a useful tool for determining common factors or trends that are present in the behavior of functions that underlie observed functional data. The main goal of FPCA is identical to multivariate PCA in the sense that both techniques aim to transform the original data by estimating components that maximize the variance observed in the data. Instead of estimating the principal components as sets of vectors that span the multi-dimensional space as is the case for multivariate PCA, FPCA estimates the principal components as functions across the continuum for which they were collected. Thus, FPCA aims to find a set of orthogonal, principal component functions that maximize the variance along each component for the functional data set.

4.3 Significance of Approach

4.3.1 Relevance of FPCA for Feature Extraction in FDD for ICEs

Indicated by Figure 4.2, a brief survey of FPCA used in literature shows that most of the research efforts have been focused towards the medical field (as indicated by the high proportion of published papers coming from biomedicine from 1995 to 2010) (Ullah & Finch, 2013).

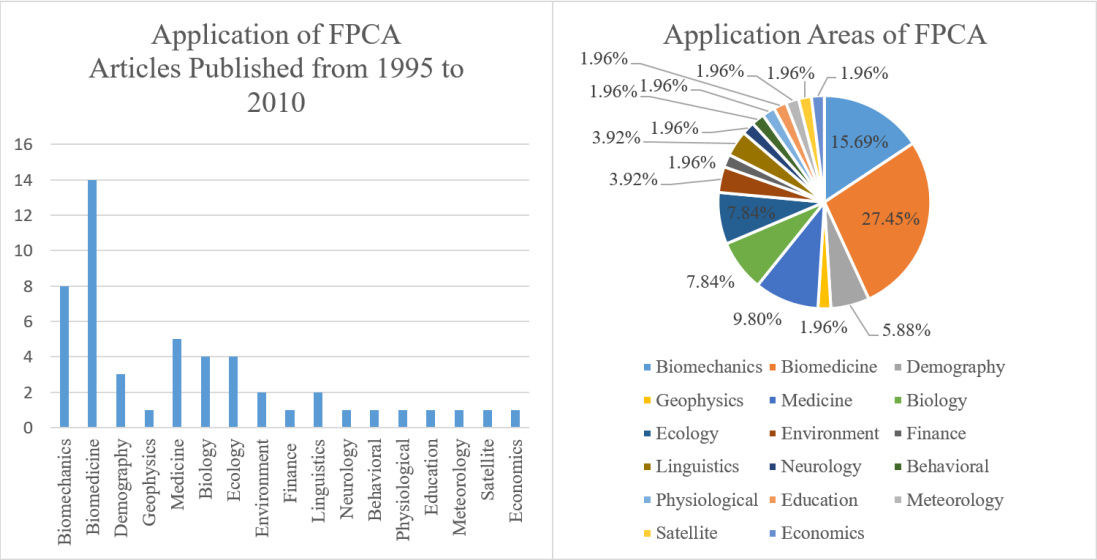

Figure 4.2 Application of FPCA articles published from 1995 to 2010. Left: bar graph Right: percentage chart

Likewise, Table 4.1 shows that the literature since 2010 has not reported extensive application of FPCA within the field of FDD.

Table 4.1 Popular FPCA applications in research since 2011

Source	Area of Application
(Coffey, Harrison, Donoghue, & Hayes, 2011)	Biomechanics
(Warmenhoven et al., 2017)	Biomechanics
(Dean et al., 2016)	Medicine
(Gong, Miller, & Scott, 2015)	Geospatial/Environmental Science
(Di Salvo, Ruggieri, & Plaia, 2015)	Environmental Science
(Nicol, 2013)	Aerospace/Aviation
(Huynh, Jacho-Chávez, Petrunia, & Voia, 2011)	Economics
(Sánchez-Sánchez et al., 2014)	Neurology & Rehabilitation
(C. Liu, Ray, & Hooker, 2017)	Statistics
(Burns, Houpt, Townsend, & Endres, 2013)	Psychology
(Moreno-Oyervides et al., 2017)	Spectroscopy
(Khanzadeh, Chowdhury, Marufuzzaman, Tschopp, & Bian, 2018)	Manufacturing

As well, advances in technology allowing for better acquisition and storage of functional data in industrial applications is creating great potential for the application of functional data analytics approaches in new areas. Thus, this thesis represents an exploratory study that aims to investigate the feasibility of using FPCA for the extraction of discriminatory features for engine state classification from vibrational signal data. The adequacy of this approach will be evaluated by coupling it with an artificial neural network (e.g. Multilayer perceptron) approach for classification and analyzing the prediction accuracy based from the features obtained from FPCA.

4.3.2 On-line Predictive Analytics for FDD for ICEs

However, the main contribution and significance with the proposed work is focused on how FPCA, when used in a FDD data-driven framework, will improve detection and prediction accuracy. As well, the computational demand required to carry out this approach will also be of interest. The goal with many FDD approaches is to provide close to real-time analytics for the process behavior. Quickly and efficiently detecting faults at or close to real time can provide significant advantages in continuous production systems when scheduling preventative maintenance, reducing unplanned downtime, and improving the organization's bottom line. Through the exploration of using FPCA for feature extraction and selection, we hope to provide support and future research directions into its application for providing real-time process analytics that can help operators, practitioners, and decision makers respond in a quicker and more correct way when faults occur in a process.

CHAPTER V

METHODOLOGY BACKGROUND

In this section, the underlying methods associated with the major analytics and machine learning techniques (FPCA and multilayer perceptrons, respectively) are introduced.

5.1 Feature Extraction – FPCA

In this research, FPCA is proposed as a novel technique for identifying the dominant modes of variation across the engines states investigated in the study. Since FPCA is an extension of multivariate PCA, an explanation of both multivariate and functional PCA is provided below. The explanation provided in all subsections of 5.1 is adapted from the explanation provided by Ramsay and Silverman (Ramsay & Silverman, 2005).

5.1.1 Multivariate PCA

The main concept exploited in multivariate statistics is the formation of linear combinations of variable values for a given dataset represented in Eq. (5.1)

$$f_i = \sum_{j=1}^{p} \beta_j x_{ij}, \;\; i = 1, \dots, N \tag{5.1}$$

where β_j is weight coefficient applied to the observed value x_i at variable j. In vector notation, Eq. (5.1) can be represented as Eq. (5.2)

$$f_i = \beta' x_i, \;\; i = 1, \dots, N \tag{5.2}$$

where β is a weight vector $[\beta_1, \dots, \beta_p]'$and x_i is the data vector $[x_{i1}, \dots, x_{ip}]'$ consisting of p variables. Applying weight coefficients allows for the transformation of the original variables. For PCA, the goal is to identify the weight coefficients that display the dominant modes of variation in the dataset.

To accomplish this, PCA proceeds via the following steps below to obtain a set of orthogonal weights that maximize the variation in the observed scores, $f_i's$:

1. Find the weight vector $\xi_1 = (\xi_{11}, \dots, \xi_{p1})'$, for which $f_{i1} = \sum_j \xi_{j1} x_{ij} = {\xi_1}' x_i$ has the largest possible mean square calculated by $N^{-1} \sum_i f_{i1}^2$ subject to the following constraint in Eq. (5.3).

$$\sum_j \xi_{j1}^2 = \|\xi_1\|^2 = 1 \tag{5.3}$$

   The weight vector ξ_1 that maximizes the mean square is the first principal component vector.

2. To calculate subsequent principal components $(\xi_2, \xi_3, \dots, \xi_m)$ where $m < p$, repeat the first step for each. For example, computing a weight vector ξ_m requires finding the weight values for the vector ξ_m that maximize the mean square of f_{im} where $f_{im} = \sum_j \xi_{jm} x_{ij} = {\xi_m}' x_i$ subject to the constraint $\|\xi_m\|^2 = 1$ and the additional $m - 1$ constraints in Eq. (5.4)

$$\sum_j \xi_{jk} \xi_{jm} = {\xi'}_k {\xi'}_m = 0 \; where \; k < m \tag{5.4}$$

The constraint in Eq. (5.4) establishes orthogonality among all the principal component vectors. This ensures that the components share no correlation with each other, and the variance explained by each component is unique to that component only. Because of the orthogonality constraint, the subsequent principal component vectors calculated beyond ξ_1 will explain less variation. Because the amount of explained variation declines with each principal component, the dominant modes of variation explained in the first few components usually account for a large percentage of the total observed variation in the original dataset.

5.1.2 FPCA

Extending the concepts of multivariate PCA to the functional domain, we see that the original vector notation of linearly combining a weight vector and data vector in Eq. (5.1) can be adapted for the functional context. Specifically, the data vector x_j becomes $x_i(s)$ such that the discrete index j in the multivariate context is converted into the continuous index s for the functional context. Likewise, the weight vector β also becomes a continuous function of s in the functional domain. Converting Eq. (5.1) into its functional analogue is displayed in Eq. (5.5)

$$\beta' x = \sum_j \beta_j x_j \quad \rightarrow \quad \int \beta x = \int \beta(s) x(s) ds. \tag{5.5}$$

Because β and x are functions of s, summations over j in the multivariate context are replaced by integrations over s in the functional context. Building on the notation described in Eq. (5.5), the corresponding score for a given function of β is

$$f_i = \int \beta x_i = \int \beta(s) x_i(s) ds \tag{5.6}$$

Observation of Eq. (5.6) allows for FPCA to be carried out in an approach identical to that of multivariate PCA. The steps for carrying out FPCA are as follows:

1. The weight function $\xi_1(s)$ is chosen such that the mean square of f_1 is maximized subject to $\int \xi_1(s)^2 ds = 1$, the continuous analogue of the unit sum of squares constraint. The calculated function of ξ_1 that maximizes $f_1's$ mean square represents the first principal component function.

2. As is the case for multivariate PCA, the calculation of all subsequent weight functions $(\xi_2(s), \xi_3(s), \ldots, \xi_m(s))$ requires maximizing the mean square of f_m subject to the continuous sum of squares constraint $\int \xi_m(s)^2 ds$ and the orthognality constraint(s) $\int \xi_k \xi_m = 0, k < m$ on subsequent steps.

Like multivariate PCA, each weight function defines the most important mode of variation of the original curves analyzed. Once again, the orthogonality constraint ensures that the variation explained in one principal component function is independent of the variation explained in all preceding and subsequent principal component functions. The orthogonality constraint also results in a decline of the explained variation by each component function as subsequent functions are calculated. Thus, the total variation of the original collection of functional curves can be well approximated by using a subset of principal component functions to transform the original functional curves.

5.1.3 PCA & Eigenanalysis

5.1.3.1 Multivariate PCA

In the multivariate context, PCA's primary task is to find the eigenvalues and eigenvectors of the covariance matrix. With this approach, matrix X is defined as a $N\ x\ p$

matrix containing all values x_{ij}, and ξ is a vector of length p that stores the linear combination of weight values that will be subjected to change for maximization of the mean square of the corresponding score, f_i. Thus, the function to maximize the mean square can be rewritten as seen Eq. (5.7).

$$\max\ N^{-1}\xi' X' X\xi\ \ such\ that\ \xi'\xi = 1 \tag{5.7}$$

Eq. (5.7) can be further modified in terms of the sample variance. Incorporating the sample variance-covariance, represented as $V = N^{-1}X'X$, Eq. (5.7) becomes Eq. (5.8) as seen below.

$$\max \xi' V\xi\ such\ that\ \xi'\xi = 1 \tag{5.8}$$

Now, this maximization problem be solved as an *eigenequation* that finds the *eigenvector,* ξ, that maximizes an *eigenvalue*, ρ.

$$V\xi = \rho\xi \tag{5.9}$$

This process yields a sequence of different eigenvalue-eigenvector pairs that satisfy Eq. (5.9). Likewise, the eigenvectors are all orthogonal to one another which satisfies the orthogonality constraint imposed by PCA.

5.1.3.2 FPCA

Extending the concepts of Eigenanalysis to the functional domain, the covariance matrix, defined as $v(s,t)$, is shown in Eq. (5.10).

$$v(s,t) = N^{-1}\sum_{i=1}^{N} x_i(s)x_i(t) \tag{5.10}$$

Proceeding with an approach like that in multivariate Eigenanalysis, similar rules are used to determine the principal component weight functions as seen in Eq. (5.11).

$$\int v(s,t)\xi(t)dt = \rho\,\xi(s) \tag{5.11}$$

The left side of the above equation represents the integral transform of the covariance operator, V, of the weight function ξ as shown in Eq. (5.12). Replacing the left side of Eq. (5.11) with Eq. (5.12) yields the eigenequation presented in Eq. (5.13).

$$V\xi = \int v(.,t)\xi(t)\,dt \tag{5.12}$$

$$V\xi = \rho\,\xi \tag{5.13}$$

In this functional representation of the eigenequation, ξ is now an eigenfunction compared to its eigenvector counterpart that was observed in the multivariate context. The obtained eigenfunction-eigenvalue pairs satisfy the eigenequation. As well, the orthogonality constraint is satisfied among all calculated eigenfunctions.

5.2 Classification for Fault Diagnosis – Multilayer Perceptron

The present study will rely on a multilayer perceptron for ICE fault diagnosis. The multilayer perceptron (MLP) is the most popular type of artificial neural network (ANNs) (Díaz-Rodríguez, Cancilla, Matute, Chicharro, & Torrecilla, 2015). MLPs belong to a general class of ANNs known as feedforward neural networks. Feedforward neural networks represent a basic type of neural network that can approximate classes of functions. As illustrated in Figure 5.1, we see a general network structure that is comprised of neurons (circles) and connecting lines that are referred to as links. Every link has a weight parameter associated with it, and every neuron receives a collective stimulus from its neighboring neurons that are connected to it. The purpose of the network is to predict a dependent variable based on the input of features deemed significant for discrimination between different dependent variable states, and the performance can be validated through available cross validation schemes.

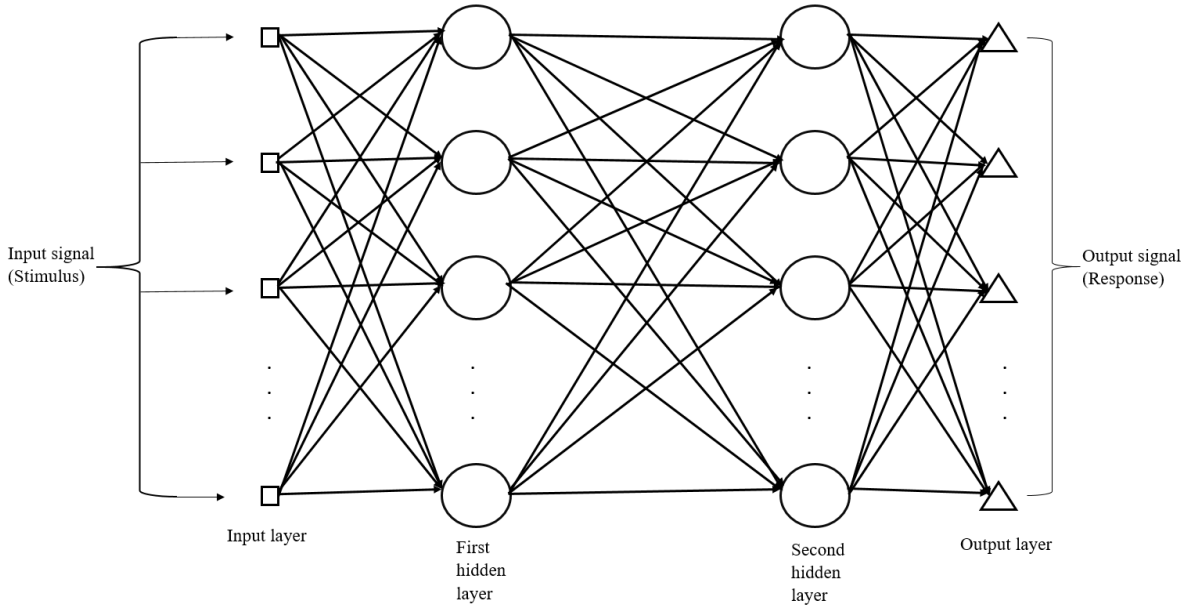

Figure 5.1 Structural representation of a multilayer perceptron with two hidden layers

Structure of the MLP

5.2.1 MLP Structure

MLPs consist of three distinct layers: an input layer, a hidden layer, and an output layer. The input layer is where feature information is fed into the network. The number of neurons in this layer will correspond to the number of selected features or independent attributes that will be used to predict the corresponding dependent attributes.

On the other end of the MLP exists the output layer. The number of neurons housed in this layer will correspond to the number of dependent class attributes. For the applications of this study, there will be five neurons in the output layer, each corresponding to the five engine states detailed in Table 6.2.

The hidden layer is the bridge that connects the independent attributes to the dependent labels. Neurons that exist in the layer(s) between the input and output layers receive stimuli from nodes in the preceding layer and transmit stimuli to nodes in the

following layer. The stimuli transduced by the neurons in the network is modulated to optimize the prediction accuracy of a dependent response based on the corresponding independent attributes.

5.2.2 MLP Mechanics

For an MLP to learn the patterns that correlate input features to a specific dependent class, training must occur. For training, many cases consisting of a set of independent features labeled with their respective dependent class labels is required.

Training begins with the feeding of the first training case into the network. The input features enter the network, and the network predicts the corresponding output. At this stage, the network has randomly assigned weights to each neuronal connection; thus, its predictive performance is entirely random. However, this establishes a baseline which will be improved upon. This baseline predictive performance is quantified by calculating the network's error (i.e. the difference between the predicted class label and the actual class label).

After the forward pass has been made through the network and the prediction error has been quantified, backpropagation occurs. Backpropagation represents the learning power of ANNs because it helps to reduce the error by altering the weights in such a way that brings the prediction of the network as close as it can to the actual class label for the input features of each case/instance (Jafari-Marandi, Davarzani, Soltanpour Gharibdousti, & Smith, 2018).

The weight updating that occurs in backpropagation relies on the partial derivative of the error based on each neuron's weight. Negative partial derivative values indicate that small positive additions to the specific neuronal weight will reduce the network's error.

Conversely, positive partial derivatives indicate that small negative additions to the specific neuronal weight will reduce the network's error (Jafari-Marandi et al., 2018).

The approach investigated in this thesis will rely on the Levenberg-Marquardt algorithm for backpropagation. The algorithm is one of the most widely used backpropagation algorithms and is an improvement to Newton's method (Hagan & Menhaj, 1994). The Levenberg-Marquardt algorithm uses the following equation detailed in Eq. (5.14) to calculate each neuronal weight change from the partial derivative and the neuronal weight change from the previous epoch.

$$\Delta\omega_{ij}(n) = \eta \frac{\delta E}{\delta \omega_{ij}} + \alpha \omega_{ij}(n-1) \tag{5.14}$$

In Eq. (5.14), E is the average of all squared errors, ω_{ij} is the weight of the connection between the i^{th} neuron and the j^{th} neuron, η is the learning rate, α is the momentum rate, and n is the epoch number.

An epoch represents one round of moving forward through the network, making a prediction, quantifying the prediction error, and propagating back through the network to alter neuronal weights to reduce the network error. The network will progress through a given number of epochs to steadily minimize the prediction error. As well, the learning rate refers to the rate at which the network changes, and the momentum rate refers to the degree of impact that past weight changes have on current weight changes (Jafari-Marandi et al., 2018)

However, the process of feeding forward and backpropagation will reach a point where the change in the error is no longer significant. At this point, the network will be

optimized for prediction based on the discriminatory power of the input features and the number of hidden nodes and hidden layer(s).

CHAPTER VI

EXPERIMENTAL DESIGN

6.1 Experimental Setup and Data Acquisition

The dataset used in this thesis was obtained by way of the experimental setup and data acquisition protocol followed by Jafarian et al (Jafarian et al., 2018).

To study and characterize different fault modes in ICEs, the authors used a 1600 cc, linear-four-cylinder, four-stroke, eight valve engine as the subject of the study. The faults investigated related to cylinder misfire and abnormal valve clearance. Specifically, three faults related to cylinder misfire (two slight misfires and one severe misfire), and one fault state was associated with abnormal valve clearance. Likewise, an engine operating in a healthy state, devoid of any misfire or clearance faults, was studied as well (Jafarian et al., 2018).

6.1.1 Fault Simulation

Cylinder misfire faults are the result of a misfiring of the spark plug. To simulate this phenomenon, the wires connected to the first and second cylinders of the engine were cut. Cutting only one of the wires represented a slight misfire (two cases: slight misfire in cylinder 1 and slight misfire in cylinder 2). Simultaneous cutting of both wires simulated a severe misfire fault (Jafarian et al., 2018).

Valve clearance represents the space between the rocker arm and valve seat as indicated by Figure 6.1. Abnormal clearance can occur when this space is too tight or too excessive respective to the manufacture's specifications. The authors focused on the occurrence of excessive valve clearance which was simulated by adjusting the exhaust valve clearance to 0.6 mm which is twice the amount of the normal clearance of 0.3 mm (Jafarian et al., 2018).

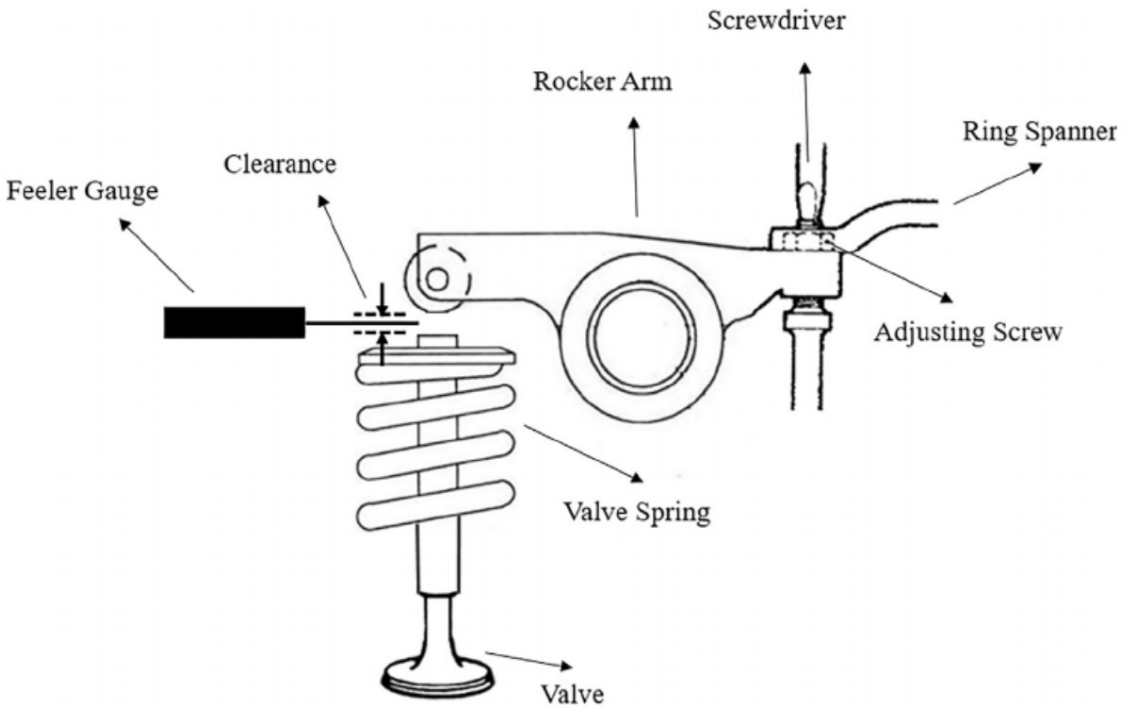

Figure 6.1 Implementation of abnormal valve clearance (Jafarian et al., 2018)

6.1.2 Vibrational Signal Monitoring and Collection

To monitor the vibration signals from the engines in the five different states mentioned in 0, four one-direction, piezoelectric, CTC accelerometers were used. For brevity, these accelerometers will be referred to as "sensors" throughout the rest of the

paper. The sensors were placed under each cylinder plug by way of a magnet. The sensors had a resonant frequency of 35 kHz; thus, 20 kHz approximately represented the upper limit of operational frequency (Jafarian et al., 2018).

Vibrational signal collection was accomplished with the ADASH4400. The ADASH4400 has four AC and DC channels. As well, the instrument had an action channel that presented vibrational wave and frequency domains and saved data in internal memory. These capabilities allowed for the collected vibrational signals to be exported as comma separated value files for additional analysis (Jafarian et al., 2018).

6.1.3 Parameter Settings for Data Acquisition

For data acquisition, literature reported a spectrum of sampling rates. A higher sampling rate of 48 kHz was reported by Flett and Bone (Flett & Bone, 2015). Using this sampling rate in Equations 6.1 and 6.2 show that the sampling rate corresponds to an encoder resolution of 1440 pulses/revolution.

$$Encoder\ Resolution = \frac{Sampling\ Rate}{\frac{N}{60}} \tag{6.1}$$

$$Encoder\ Pulse = \frac{360°}{Encoder\ Resoultion} \tag{6.2}$$

However, most of the reports in literature do not use a sampling rate as high as the one used by Flett and Bone. High sampling rates tend to produce higher encoder resolution values. Alternate approaches have used encoder pulse values that result from encoder resolution values much smaller than those reported by Flett and Bone. Thus, in this research, 2000 RPM was considered for the rotational speed of the engine's crankshaft, and a sample rate of 2 kHz was used. This resulted in an encoder resolution of 60 pulses/revolution which corresponded to an encoder pulse every 6°. The encoder resolution

of 60 pulses/revolution provided a moderate number according to the settings used in relevant literature detailed in Table 6.1.

Table 6.1 Encoder Resolution and Pulse Values Used in the Literature

Source	Encoder Resolution	Encoder Pulse
Flett and Bone (Flett & Bone, 2015)	1440 pulses/revolution	0.25°
Kiencke (Kiencke, 1999)	60 pulses/revolution	6°
Jafarian et al (Jafarian et al., 2018)	60 pulses/revolution	6°
Osburn (Osburn, Kostek, Franchek, & Franchek, 2005)	12 pulses/revolution	30°
Jung et al (Jung, Eriksson, Frisk, & Krysander, 2014)	12 pulses/revolution	30°
Naik (Naik, 2004)	4 pulses/revolution	90°

6.1.4 Experimental Setup

Employing the aforementioned data monitoring and acquisition scheme, vibrational signal data was collected for each engine state described in Table 6.2.

Table 6.2 Engine states investigated in present study

State	Abbreviation	Engines
Healthy	H	30
Slight Misfire 1	M1	30
Severe Misfire	M12	30
Slight Misfire 2	M2	30
Abnormal Valve Clearance	VC	30

In this research, 30 engines were investigated for each of the five states. On each of the engines, a one minute, 60,000 observation signal was collected on each sensor

resulting in a total of four one minute, 60,000 observation signals for each engine in the study. These signals were only one minute in duration because longer duration study periods could have resulted in malignant engine damage.

6.2 Signal Processing

For this research, 600 one minute, 60,000 observation signals were collected (1 signal per sensor, 4 sensors per engine, 150 engines). Below in Figure 6.2 is an example of a full, one-minute signal. Obviously, it is difficult to observe the waveform behavior due to the frequency of the signal.

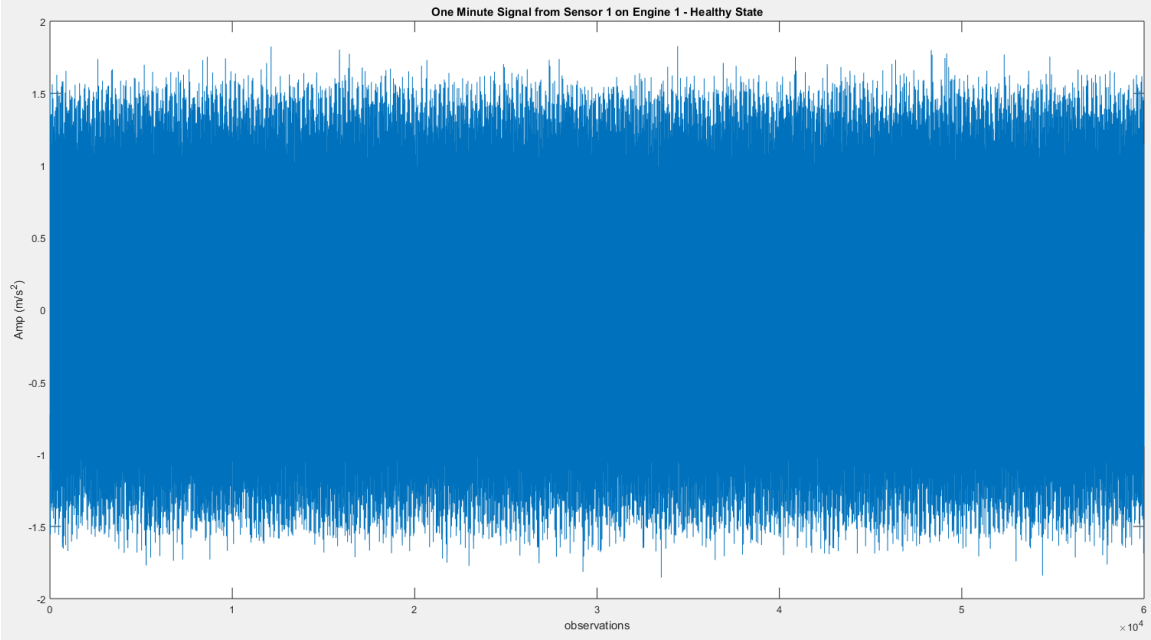

Figure 6.2 Full one-minute signal obtained from Sensor 1 on Engine 1 in a Healthy state

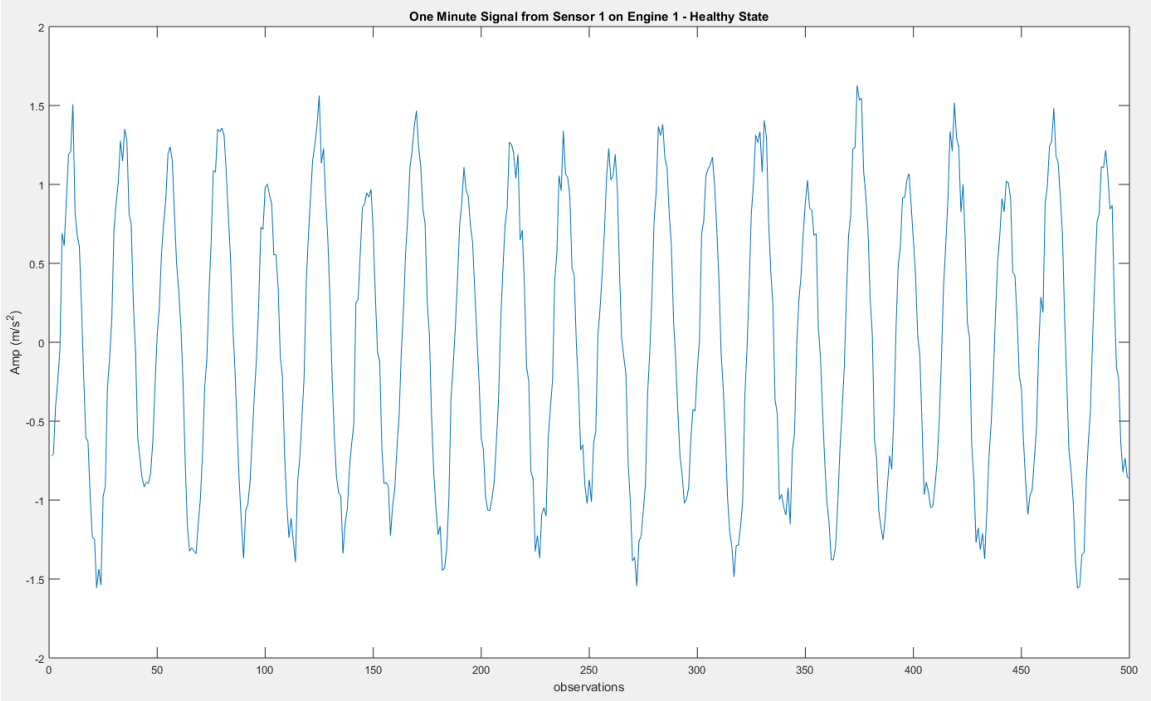

Figure 6.3 First 500 observations of signal data obtained from Sensor 1 on Engine 1 in a Healthy state

6.2.1 Signal Extraction

To better view the waveform signatures, we analyzed fractions of the signals. In Figure 6.3, we can observe a more signature behavior of the waveform by analyzing 500 observations which is equivalent to half of a second of the original 60 second signal. Expanding on this approach, we aimed to extract portions of the one minute, 60,000 observation signals to perform feature extraction and feature selection for fault detection. Specifically, we were interested in investigating the prediction accuracy that corresponded with small fractions of the overall signal. For this thesis, we investigated the feature extraction, feature selection, and classification using the following durations for extracted signal portions: 100 observations, 500 observations, 1,000 observations, 5,000

observations, and 10,000 observations. Each of these durations were extracted from the beginning of the signal for the initial phases of investigation.

In addition to using varying extracted portions of the signals, we also investigated the effect of having more cases for MLP training and testing. Table 6.3 combines the signal duration and case number combination to completely detail the scenarios investigated in this thesis.

Table 6.3 Signal Duration – Case combination scheme used in present study

Signal Duration (observations)	Signal Duration (time)	Cases	% of Total Signal
100	0.1 seconds	150	0.167 %
100	0.1 seconds	750	0.833 %
500	0.5 seconds	150	0.833 %
500	0.5 seconds	750	4.167 %
1,000	1 second	150	1.667 %
1,000	1 second	750	8.333 %
5,000	5 seconds	150	8.333 %
5,000	5 seconds	750	41.667 %
10,000	10 seconds	150	16.667 %
10,000	10 seconds	750	83.333 %

Figure 6.4 shows examples of the signal extraction used for this study. In situations where 750 cases were observed, we extracted multiple signals from the same engine in the same state. For example, as illustrated in Figure 6.4, generating 750 cases for a signal duration of 100 observations required the extraction of a sequence of the first 500 observations from one signal. Subsequently, this 500-observation sequence would be divided into five separate, 100 observation signals.

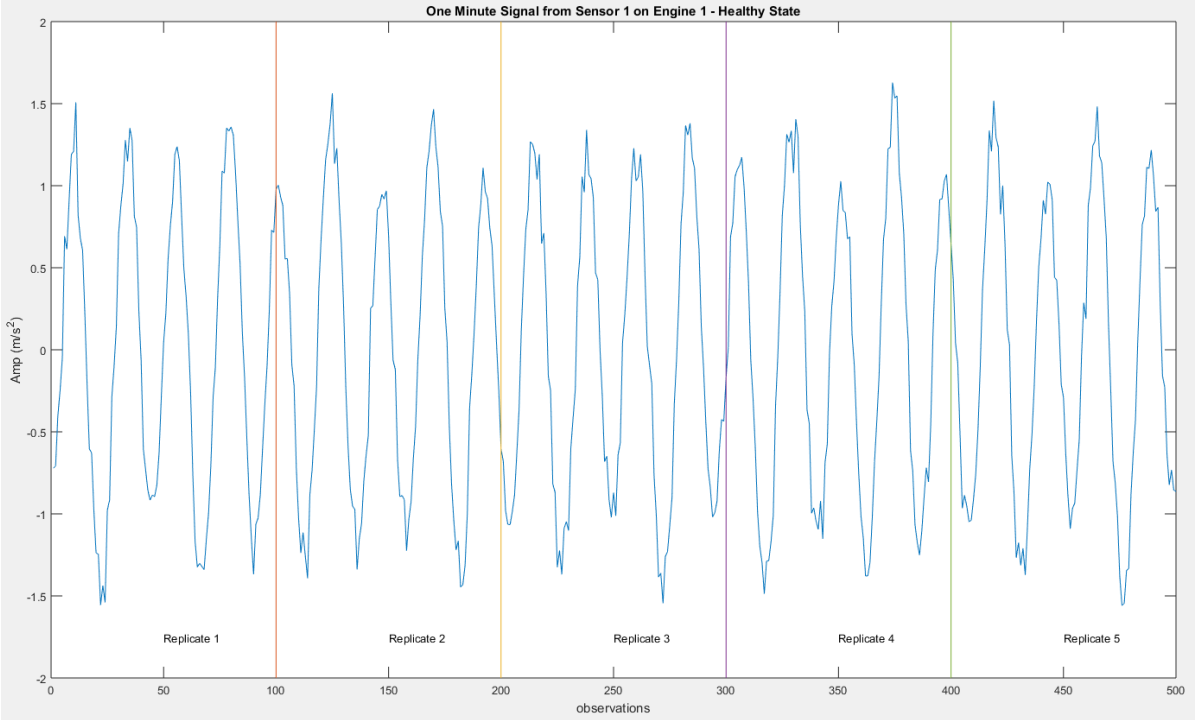

Figure 6.4 Five 100 observation signals in sequence. This approach was used to generate five replicate cases for each signal to allow for 750 total cases across all 150 engines used in the study

Using this approach was possible because all signals exhibited stationary behavior over the one-minute time span for which the signal was collected. We also implemented a random sampling scheme to identify and extract signals from random locations in the one-minute signal to verify this claim which will be discussed in further detail in the Analysis and Results chapter.

6.2.2 Curve Smoothing

For data with observable noise as seen in Figure 6.3, it is wise to employ some type of smoothing approach. One of these such approaches is using smoothing splines. A smoothing spline s is constructed based off a smoothing parameter p and specified weights w_i as depicted in Eq. (6.3).

$$p\sum_i w_i\left(y_i - s(x_i)\right)^2 + (1-p)\int((d^2s)/(dx^2))^2 \ \ dx \tag{6.3}$$

The smoothing parameter,p, is defined along the range [0, 1]. When p is 0, the resultant spline is equivalent to a least-squares straight-line fit of the data. Alternatively, when p is 1, the resultant spline is in the form of a cubic spline interpolant. According to Pollock, determining the appropriate smoothing parameter "represents the optimal predictor of the path of a certain stochastic differential equation of which the observations are affected by noise" (Pollock, 1999).

By using the smoothing splines function within MATLAB, the default smoothing parameter is selected in the "interesting range". This range is near $1/(1 + h^3/6)$ where h represents the average spacing of the collected data points. While the smoothing parameter can be manually adjusted by the user, the default smoothing parameter obtained within MATLAB produces a curve that is smoother than the interpolant but also adequately fits the data. In the case where the smoothing parameter is decreased relative to the default value, the resulting spline curve is smoother but does not fit the data well. Conversely, increasing the smoothing parameter relative to the default value results in a curve that approaches a cubic spline interpolant ("Smoothing Splines - MATLAB & Simulink," n.d.).

When the default smoothing parameter is lower than 1 (e.g. p = 0.95), major differences can be observed with divergences at the end points. However, depending on the data, the default smoothing parameter can result in a cubic spline interpolant with p = 1 or ≈ 1. For this research, the default smoothing parameter was calculated to be 1 for all the signal duration/case number combinations detailed in Table 6.3.

6.3 FPCA Application

6.3.1 Feature Extraction – Calculation of Principal Components

For the scenarios presented in Table 6.3, the corresponding collection of given signal duration curves were collected. Each instance has signals from each of the four sensors. Table 6.4 shows the matrix representation of the data once it had been processed to undergo FPCA. With this approach, FPCA would be performed on each of the sensor columns indicating that four sets of PC functions will be calculated from the four sensor collections of either 150 or 750 functional signal curves.

Table 6.4 Matrix representation of 150 case examples of 100 observation signals

Index	Engine State	S1 Signal	S2 Signal	S3 Signal	S4 Signal
1	H	100 obs curve	100 obs curve	100 obs curve	100 obs curve
2	H	100 obs curve	100 obs curve	100 obs curve	100 obs curve
…	H	100 obs curve	100 obs curve	100 obs curve	100 obs curve
30	H	100 obs curve	100 obs curve	100 obs curve	100 obs curve
31	M1	100 obs curve	100 obs curve	100 obs curve	100 obs curve
32	M1	100 obs curve	100 obs curve	100 obs curve	100 obs curve
…	M1	100 obs curve	100 obs curve	100 obs curve	100 obs curve
60	M1	100 obs curve	100 obs curve	100 obs curve	100 obs curve
61	M12	100 obs curve	100 obs curve	100 obs curve	100 obs curve
62	M12	100 obs curve	100 obs curve	100 obs curve	100 obs curve
…	M12	100 obs curve	100 obs curve	100 obs curve	100 obs curve
90	M12	100 obs curve	100 obs curve	100 obs curve	100 obs curve
91	M2	100 obs curve	100 obs curve	100 obs curve	100 obs curve
92	M2	100 obs curve	100 obs curve	100 obs curve	100 obs curve
…	M2	100 obs curve	100 obs curve	100 obs curve	100 obs curve
120	M2	100 obs curve	100 obs curve	100 obs curve	100 obs curve
121	VC	100 obs curve	100 obs curve	100 obs curve	100 obs curve
122	VC	100 obs curve	100 obs curve	100 obs curve	100 obs curve
…	VC	100 obs curve	100 obs curve	100 obs curve	100 obs curve
150	VC	100 obs curve	100 obs curve	100 obs curve	100 obs curve

6.3.2 Feature Selection – 95% explained variance

Because the present study is of an exploratory nature, we analyzed the sequence of PC functions that explained 95% of the variation present in each sensors' curve collections. As detailed later, this approached reduced the number of total PC functions due to the diminishing return of variation explanation by downstream PC functions.

6.4 Classification – Multilayer Perceptron

The selected PC functions that explained 95% of the variance in the original data were passed into an MLP to determine predictive performance. A trial and error method was employed to determine the appropriate network structure (i.e. the number of hidden layers used as well as the number of neurons present in each layer) that provided an adequate balance between prediction accuracy and variability.

6.4.1 Network Tuning – Trial & Error

A trial & error approach was used to investigate the optimal number of neurons in a MLP with two hidden layers (L1 and L2) with a maximum of 10 neurons in each layer. With this approach, all possible neuron combinations for a MLP with two hidden layers were investigated. For a given [1, 1] network (e.g. neurons in L1 = 1, neurons in L2=1), a network's performance is trained on 80% of the data for a specific signal duration/case number combination and tested on the remaining 20%. This process is repeated 50 times for a given network to generate performance statistics (e.g. prediction accuracy and variance/standard deviation). Finally, the performance statistics are then used to conduct statistical analysis via paired t-test to determine when significance exists for the observed

differences in the calculated network performance across all signal duration/case number combinations.

6.4.2 K-Fold Cross Validation & Effect of Random Initialization

Once the optimal neuron number for the hidden layers were identified, we then used the optimal parameter settings and performed k-fold cross validation with $k = 5$. With this process, we also investigated the effect of MLP weight initialization by carrying out k-fold cross validation on ten additional networks with the optimal parameter settings that underwent different initializations.

6.5 Stationary Signal Validation

Lastly, we validated that the backend MLP performance statistics were consistent with results obtained from taking the same optimal signal/duration case number portion of the total signal from other random locations within the 60,000-observation signal. We repeated this process for ten random locations and compiled the MLP performance statistics.

CHAPTER VII

ANALYSIS AND RESULTS

7.1 Signal Processing – Signal Extraction and Curve Smoothing

As described in the methodology and experimental setup, the signal processing protocol was employed for each scenario. For illustrative purposes, the scenario where 150 cases of 100 observation signal fractions are displayed in Figure 7.1 and Figure 7.2. In Figure 7.1, we see that the raw signal extracts possess a noticeable degree of noise. Thus, curve smoothing was employed to reduce this noise which is illustrated in the functional curves in Figure 7.2. This process was repeated for every combination scenario listed in Table 6.3. Once a collection of smoothed, functional curves was obtained from the extracted signals, FPCA was applied to determine the PC functions and corresponding eigenvalues. The eigenvalues were used to determine the appropriate number of PC functions to explain different percentages of the variation in the original functional curve collections.

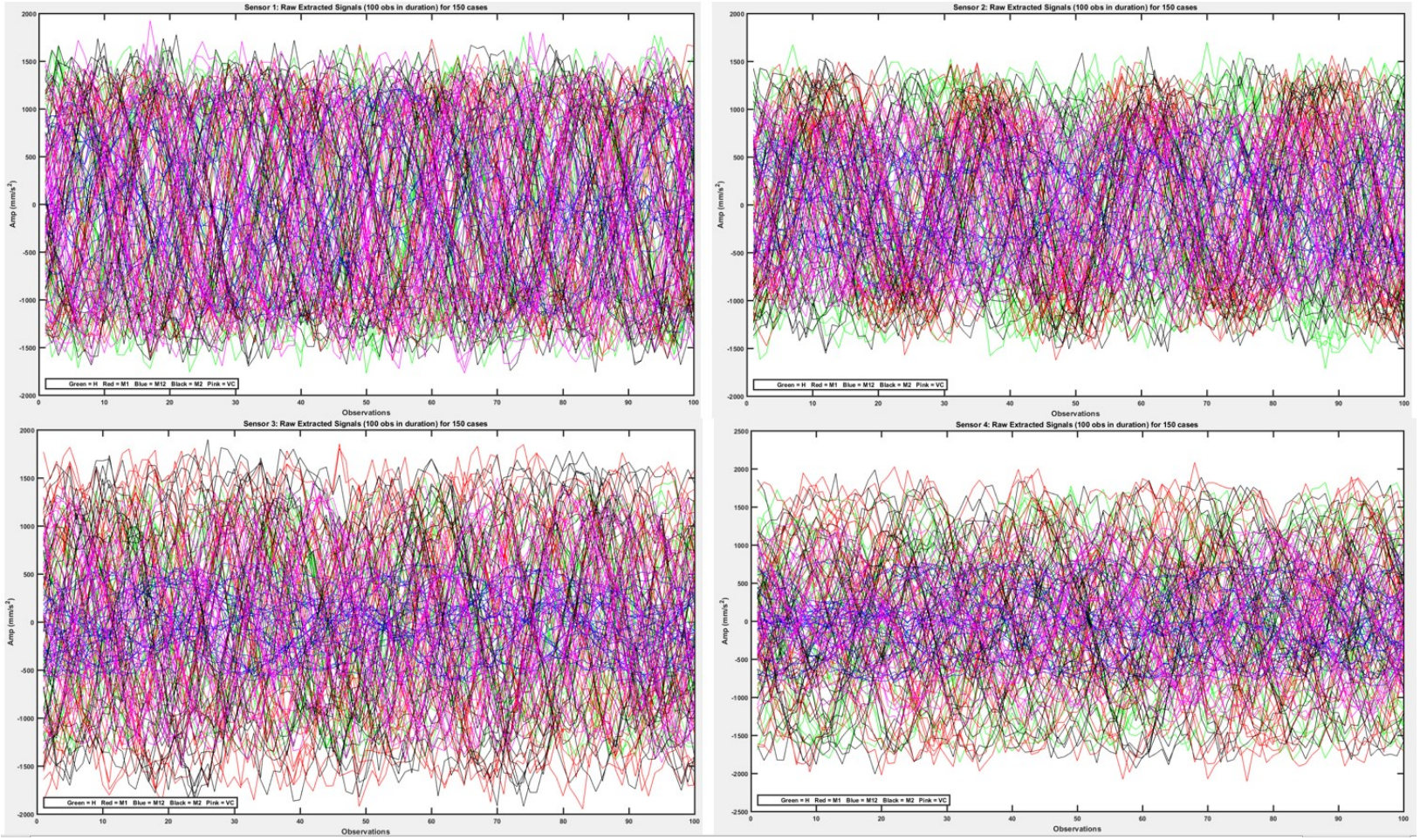

Figure 7.1 Each plot represents a sensor's collection of 150 signals (1 for each engine) where each signal is a 100-observation signal (extracted from observations 1-100 of the total 60,000 observation signal).

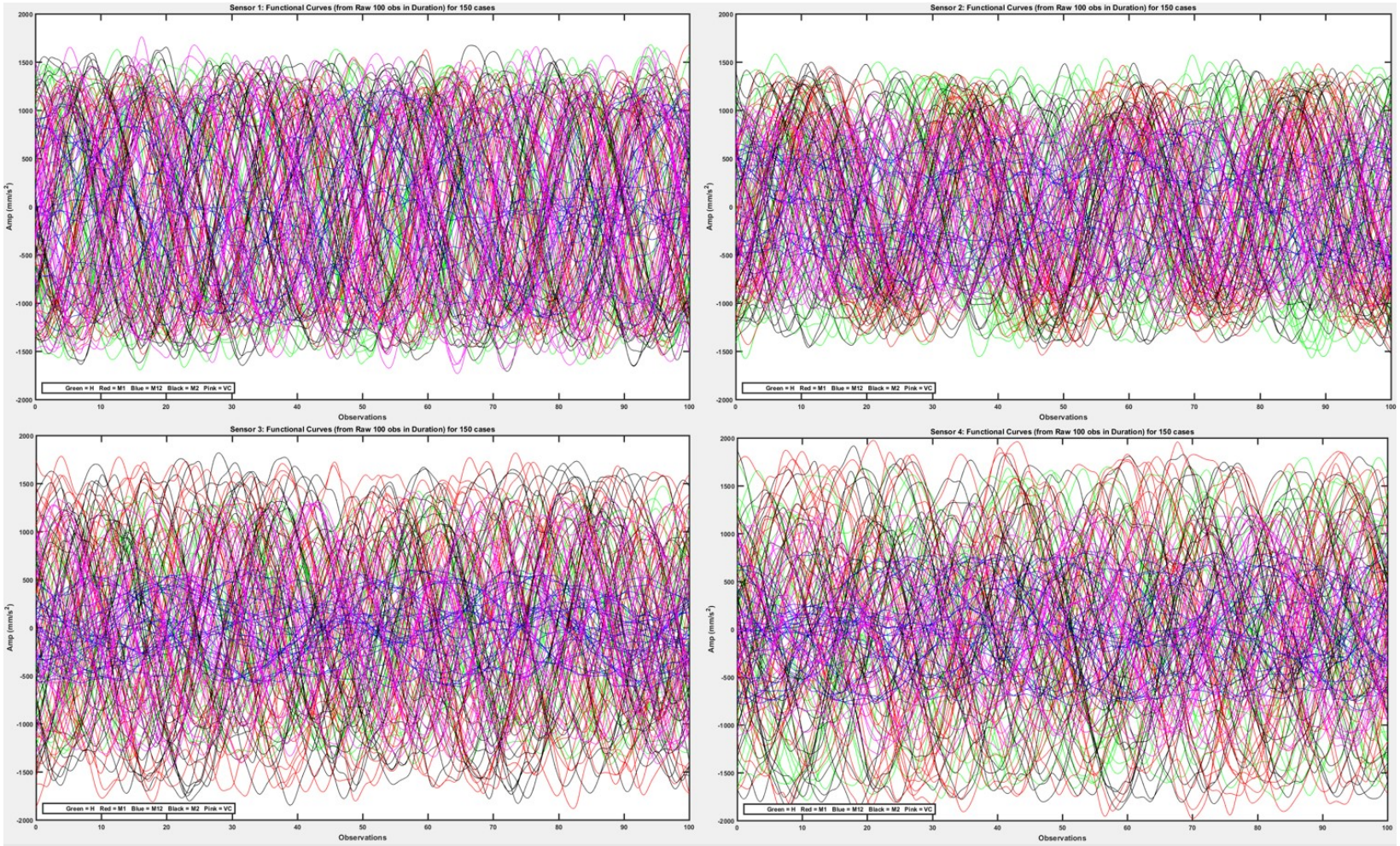

Figure 7.2 Each plot represents a collection of 150 smoothed signal curves (1 for each engine) for a sensor.

7.2 FPCA

FPCA was applied to every collection of functional curves for the scenarios investigated in this study. From FPCA, we were able to extract the PC functions and then select the first N functions (where $N <$ (# of Cases – 1)) that explain a given percentage of the variance. These results are compiled in Table 7.1.

Table 7.1 FPCA Explained Variance – PC Sequence. S1 = Sensor 1, S2 = Sensor 2, S3 = Sensor3, S4 = Sensor 4

			PC Functions 95 % Exp Variance				PC Functions 99 % Exp Variance			
Signal Duration (observations)	Signal Duration (time)	Cases	S1	S2	S3	S4	S1	S2	S3	S4
100	0.1 seconds	150	7	7	7	7	10	10	10	10
100	0.1 seconds	750	7	7	7	7	10	10	10	10
500	0.5 seconds	150	12	9	10	15	23	27	19	26
500	0.5 seconds	750	13	9	10	15	26	31	22	28
1,000	1 second	150	15	9	10	18	34	37	23	33
1,000	1 second	750	18	9	10	20	40	49	30	40
5,000	5 seconds	150	16	9	10	19	39	40	24	35
5,000	5 seconds	750	16	9	10	19	42	46	25	35
10,000	10 seconds	150	16	9	10	19	38	40	24	34
10,000	10 seconds	750	18	9	10	19	42	46	25	34

Selecting the first N components of each collection of curves for each sensor resulted in four sets of PC functions to explain the specified proportion of variance. For MLP investigation, we opted to use the selected PC functions that explained 95% of the variance. This proportion was chosen because it approximates the original data set. As well, the PC functions that explain 95% of the variance are approximately half the number of the PC functions that are necessary to explain 99% of the variance when handling many of the signal duration/case number scenarios investigated in the study.

7.3 MLP Tuning – Trial & Error Approach

The first N components for sensors 1 through 4 were used as input features for the MLP. Likewise, the corresponding class labels (i.e. engine states) were used as the output features. Table 7.2 below details the network parameters that correspond with the highest accuracy and the lowest standard deviation. Because the approach was repeated 50 times, statistical analysis was conducted to investigate if the differences observed in network performance was significant.

Table 7.2 Trial & Error MLP Optimization for Signal Duration/# of Cases Combinations

		Max. Accuracy				Min. Standard Deviation			
Signal Duration	# of Cases	L1	L2	Acc	StDev	L1	L2	Acc	StDev
100	150	8	6	87.67 %	5.81 %	8	6	87.67 %	5.81 %
100	750	10	5	97.87 %	1.52 %	8	7	96.90 %	1.40 %
500	150	10	10	89.71 %	6.35 %	9	5	88.80 %	6.35 %
500	750	10	7	99.02 %	1.03 %	8	9	98.95 %	1.00 %
1000	150	7	10	88.65 %	7.41 %	9	10	86.76 %	6.18 %
1000	750	10	5	98.93 %	1.24 %	10	9	98.91 %	1.00 %
5000	150	9	6	89.25 %	5.79 %	9	6	89.25 %	5.79 %
5000	750	8	9	99.24 %	0.94 %	10	10	99.22 %	0.86 %
10000	150	6	9	87.74 %	7.15 %	10	8	87.44 %	6.64 %
10000	750	7	10	99.17 %	1.28 %	5	6	99.02 %	0.89 %

As mentioned previously, the performance statistics compiled in Table 7.2 are the averages across 50 iterations. Statistical analysis was carried out on the data presented in Table 7.2. Analyzing the effect of the number of cases used, it was observed that at a significance level of $\alpha = 0.05$, the optimized network's performance was significantly better when 750 cases were present (i.e. 600 cases were used for training and 150 cases were used for testing). The results of this analysis are presented in Table 7.3.

Table 7.3 Statistical analysis of the significance of # of cases with performance statistics associated from the networks that produced the highest prediction accuracy for each scenario

Signal	Cases	Mean (%)	SD (%)	p (%)	Decision H_0: No significant difference between samples' predictive performance
100	150	87.67	5.81	< 0.001	Reject null
	750	97.87	1.52		
500	150	89.71	6.35	< 0.001	Reject null
	750	99.02	1.00		
1000	150	88.65	7.41	< 0.001	Reject null
	750	98.92	1.24		
5000	150	89.25	5.79	< 0.001	Reject null
	750	99.24	0.94		
10000	150	87.74	7.15	< 0.001	Reject null
	750	99.17	1.28		

Because of the performance advantage associated with using 750 cases, only scenarios involving 750 cases analyzed. Figure 7.3 shows a plot of the 95% confidence intervals of the t-distribution for each accuracy of each 750-case signal duration.

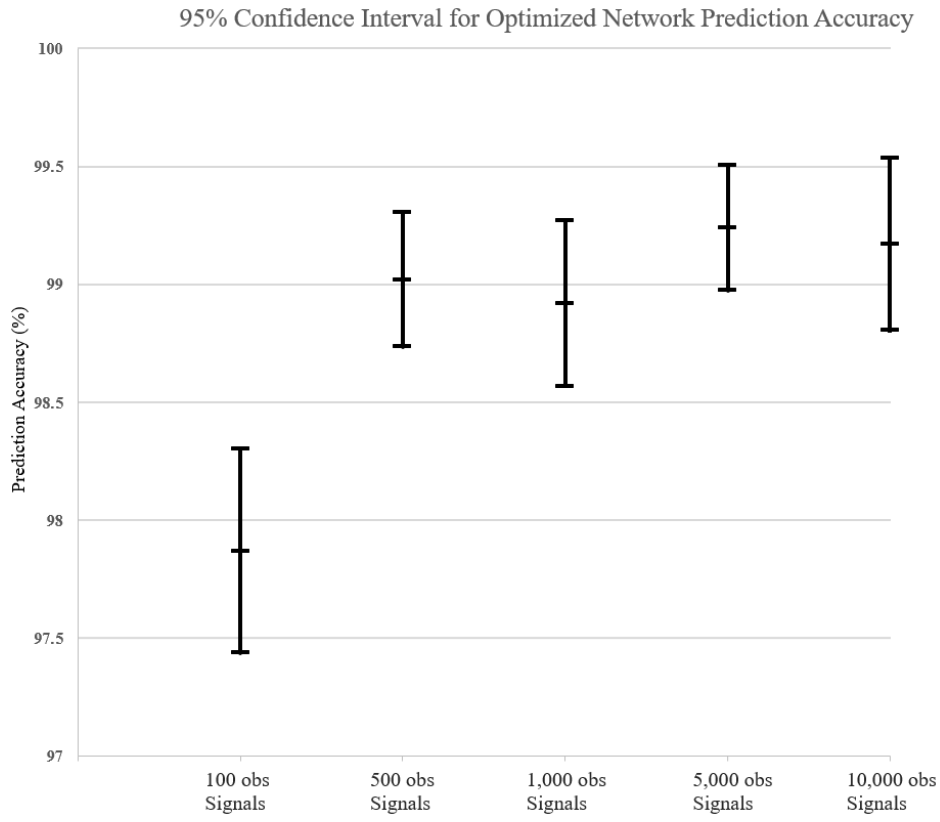

Figure 7.3 Interval plot of the 95% confidence intervals for the prediction accuracy for each signal duration studied with a case number of 750.

From Figure 7.3, there is no overlap between the 95% CI for prediction accuracy of the 100-observation signal collection when compared to all the other signal durations investigated. This indicates that the prediction performance from originally using a collection of 750, 100 observation signals is significantly lower than the other scenarios that were studied. Conversely, we see that the mutual overlapping of the remaining intervals indicates that there is not a significant difference from the observed 95% confidence intervals for prediction accuracy. This lack of significance among the prediction performance from using 500, 1000, 5000, and 10000 signal duration collections allows us to use the most convenient of these options. From a computational standpoint, operations such as curve smoothing become increasingly more complex when longer

duration signals must be smoothed. Thus, we chose the scenario of using 750 cases of 500 observation signals for the final MLP cross validation and performance evaluation phase of this research.

7.4 K-Fold Cross Validation

K-Fold Cross Validation (CV) was carried out to validate MLP prediction performance resulting from applying FPCA on scenario using 750 cases of 500 observation, smoothed signals. The network settings used are identified in Table 7.2 ([10, 7] network i.e. an MLP with 10 neurons in the first hidden layer and 7 neurons in second hidden layer). K-Fold CV proceeded by first randomly shuffling the 750 cases of data. The [10, 7] two-hidden layer MLP was then randomly initialized. Then, using a k=5, the entire dataset was split into five 150 case folds, and K-Fold CV was performed. This exact process was repeated 10 times where each time the random initializations of the MLP were different. Table 7.4 below displays the results of this process.

Table 7.4 K-Fold CV results for MLP using extracted PC function features from 750 cases of smoothed, 500 observation extracted signals

Network Index	Acc (%)	Error Count	Error ID
1	99.73	2	7, 14
2	99.07	7	14, 14, 14, 7, 7, 14, 14
3	99.60	3	7, 7, 7
4	99.87	1	14
5	99.87	1	14
6	99.73	2	7, 6
7	99.47	4	7, 7, 7, 7
8	99.73	2	14, 14
9	100.00	0	-
10	100.00	0	-

Table 7.4 shows little variation in terms of error across K-Fold CV performed on 15 differently initialized [10, 7] MLP networks. Additionally, from Table 7.4 we see the column 'Error ID'. This indicates the extent of the misclassification that produced the prediction error. Table 7.5 below represents a dictionary that defines the Error ID for each type of misclassification.

Table 7.5 Misclassification Error ID dictionary

	Misclassification	
Error ID	Actual	Predicted
1	H	M1
2	H	M12
3	H	M2
4	H	VC
5	M1	H
6	M1	M12
7	M1	M2
8	M1	VC
9	M12	H
10	M12	M1
11	M12	M2
12	M12	VC
13	M2	H
14	M2	M1
15	M2	M12
16	M2	VC
17	VC	H
18	VC	M1
19	VC	M12
20	VC	M2

To provide a clearer presentation of the data in Table 6.4, the K-Fold CV results across the 10 different initialized [10,7] MLP networks are presented in the confusion matrix below in Table 7.6.

Table 7.6 Confusion matrix for summed K-Fold CV results across 10 different initialized [10,7] MLP networks on the initial observation investigation range of 1-2500 for each collected signal

Total Predicted Cases for Each State = 1500		Predicted				
		H	M1	M12	M2	VC
Actual	H	1500 (100%)	0	0	0	0
	M1	0	1488 (99.2%)	1 (0.07%)	11 (0.73%)	0
	M12	0	0	1500 (100%)	0	0
	M2	0	10 (0.67%)	0	1490 (99.33%)	0
	VC	0	0	0	0	1500 (100%)

With the insights gained from Table 7.6, we see that the present error types are mainly either 7 or 14, corresponding to the error of classifying an M1 fault as M2 and vice versa. Likewise, there is only one occurrence of an error type of 6 which is the misclassification of an M1 fault as M12. The mode of misclassification is relevant because as indicated by Table 7.7 and Table 7.8, these faults are not considered as a Type I or Type II error.

Table 7.7 Potential Type I errors

Actual	Predicted
H	M1
H	M12
H	M2
H	VC

Table 7.8 Potential Type II errors

Actual	Predicted
M1	H
M12	H
M2	H
VC	H

7.5 Stationary Signal Validation

The results obtained thus far were the result of using the first observations (e.g. 1-2500) of the total 60,000 observation signal. This 2500 signal was then split accordingly (1-500, 501-1000, 1001-1500, 1501-2000, and 2001-2500) allowing for five 500 observation signals to be collected from 1 engine. Because 150 engines were in the study, this allowed for 750 cases.

An important signal feature that we visually observed initially was that the signal behavior did not evolve over the minute time span for which is collected. To validate this, we used the same overall approach of the study (signal extraction, curve smoothing, FPCA, and MLP) to observe the MLP prediction performance resulting from taking signals at random locations in the entire 60,000 observation signal and applying FPCA. Using the optimal signal duration (500 observations) at 750 cases, this process was carried out 10 times at 10 different locations that are detailed in Table 7.9.

Table 7.9 Random Extracted Ranges for Investigation

Random Sample	Random Location in 60,000 obs Signal	Extracted Observation Range
1	23,979	23,979 – 26,479
2	41,919	41,919 – 44,419
3	29,552	29,552 – 32,052
4	24,479	24,479 – 26,979
5	12,840	12,840 – 15,340
6	12,765	12,765 – 15,265
7	50,033	50,033 – 52,533
8	50,223	50,223 – 52,723
9	37,397	37,397 – 39,897
10	5,609	5,609 – 8,109

FPCA was applied to all the scenarios, and the corresponding PC functions that explained 95% of the variance were extracted and used as input for the MLP. K-fold CV was performed in similar manner as previously discussed in Section 7.4. The tables corresponding to the accuracy for each randomly initialized [10, 7] MLP network are located in the appendix in Table A.1, Table A.2, and Table A.3.

To provide clearer presentations of the data in Table A.1, Table A.2, and Table A.3 in the appendix, the K-Fold CV results across the 10 different initialized [10,7] MLP networks for each random signal range are presented in the confusion matrices below in Table 7.10 - Table 7.19.

Table 7.10 Confusion matrix for summed K-Fold CV results across 10 different initialized [10,7] MLP networks on the observation range of (23,979 – 26,479) of each collected signal

Total Predicted Cases for Each State = 1500		Predicted				
		H	M1	M12	M2	VC
Actual	H	1500 (100%)	0	0	0	0
	M1	0	1490 (99.33%)	0	10 (0.67%)	0
	M12	0	0	1500 (100%)	0	0
	M2	0	10 (0.67%)	0	1490 (99.33%)	0
	VC	0	0	0	0	1500 (100%)

Table 7.11 Confusion matrix for summed K-Fold CV results across 10 different initialized [10,7] MLP networks on the observation range of (41,919 – 44,419) of each collected signal

Total Predicted Cases for Each State = 1500		Predicted				
		H	M1	M12	M2	VC
Actual	H	1500 (100.00%)	0	0	0	0
	M1	0	1488 (99.20%)	1 (0.07%)	11 (0.73%)	0
	M12	0	0	1500 (100.00%)	0	0
	M2	0	9 (0.60%)	0	1491 (99.40%)	0
	VC	0	0	0	0	1500 (100.00%)

Table 7.12 Confusion matrix for summed K-Fold CV results across 10 different initialized [10,7] MLP networks on the observation range of (29,552 – 32,052) of each collected signal

Total Predicted Cases for Each State = 1500		Predicted				
		H	M1	M12	M2	VC
Actual	H	1500 (100.00%)	0	0	0	0
	M1	0	1488 (99.20%)	0	11 (0.73%)	1 (0.07%)
	M12	0	0	1500 (100.00 %)	0	0
	M2	0	15 (1.00%)	0	1485 (99.00%)	0
	VC	0	0	0	0	1500 (100.00 %)

Table 7.13 Confusion matrix for summed K-Fold CV results across 10 different initialized [10,7] MLP networks on the observation range of (24,479 – 26,979) of each collected signal

Total Predicted Cases for Each State = 1500		Predicted				
		H	M1	M12	M2	VC
Actual	H	1500 (100.00%)	0	0	0	0
	M1	0	1487 (99.13%)	0	13 (0.87%)	0
	M12	0	0	1500 (100.00%)	0	0
	M2	0	19 (1.33%)	0	1481 (98.67%)	0
	VC	0	0	0	0	1500 (100.00%)

Table 7.14 Confusion matrix for summed K-Fold CV results across 10 different initialized [10,7] MLP networks on the observation range of (12,840 – 15,340) of each collected signal

Total Predicted Cases for Each State = 1500		Predicted				
		H	M1	M12	M2	VC
Actual	H	1500 (100.00%)	0	0	0	0
	M1	0	1492 (99.46%)	0	8 (0.54%)	0
	M12	0	0	1500 (100.00%)	0	0
	M2	0	8 (0.54%)	0	1492 (99.46%)	0
	VC	0	0	0	0	1500 (100.00%)

Table 7.15 Confusion matrix for summed K-Fold CV results across 10 different initialized [10,7] MLP networks on the observation range of (12,765 – 15,265) of each collected signal

Total Predicted Cases for Each State = 1500		Predicted				
		H	M1	M12	M2	VC
Actual	H	1500 (100.00%)	0	0	0	0
	M1	0	1490 (99.33%)	0	10 (0.67%)	0
	M12	0	0	1500 (100.00%)	0	0
	M2	0	12 (0.80%)	0	1488 (99.20%)	0
	VC	0	0	0	0	1500 (100.00%)

Table 7.16 Confusion matrix for summed K-Fold CV results across 10 different initialized [10,7] MLP networks on the observation range of (50,033 – 52,533) of each collected signal

Total Predicted Cases for Each State = 1500		Predicted				
		H	M1	M12	M2	VC
Actual	H	1500 (100.00%)	0	0	0	0
	M1	0	1494.36 (99.60%)	0	6 (0.40%)	0
	M12	0	0	1500 (100.00%)	0	0
	M2	0	18 (1.20%)	0	1482 (98.80%)	0
	VC	0	0	0	0	1500 (100.00%)

Table 7.17 Confusion matrix for summed K-Fold CV results across 10 different initialized [10,7] MLP networks on the observation range of (50,223 – 52,723) of each collected signal

Total Predicted Cases for Each State = 1500		Predicted				
		H	M1	M12	M2	VC
Actual	H	1500 (100.00%)	0	0	0	0
	M1	0	1484 (98.93%)	0	16 (1.07%)	0
	M12	0	0	1500 (100.00%)	0	0
	M2	0	6 (0.40%)	0	1494 (99.60%)	0
	VC	0	0	0	0	1500 (100.00%)

Table 7.18 Confusion matrix for summed K-Fold CV results across 10 different initialized [10,7] MLP networks on observation range of (37,397 – 39,897) of each collected signal

Total Predicted Cases for Each State = 1500		Predicted				
		H	M1	M12	M2	VC
Actual	H	1500 (100.00%)	0	0	0	0
	M1	0	1492 (99.47%)	0	8 (0.53%)	0
	M12	0	0	1500 (100.00%)	0	0
	M2	0	13 (0.87%)	0	1487 (99.13%)	0
	VC	0	0	0	0	1500 (100%)

Table 7.19 Confusion matrix for summed K-Fold CV results across 10 different initialized [10,7] MLP networks on observation range of (5,609 – 8,109) of each collected signal

Total Predicted Cases for Each State = 1500		Predicted				
		H	M1	M12	M2	VC
Actual	H	1500 (100.00%)	0	0	0	0
	M1	0	1490 (99.33%)	0	10 (0.67%)	0
	M12	0	0	1500 (100.00%)	0	0
	M2	0	10 (0.67%)	0	1490 (99.33%)	0
	VC	0	0	0	0	1500 (100.00%)

By observing the ten predicted accuracies for each initialized network for each of the ten random ranges investigated detailed in Table A.1, Table A.2, and Table A.3of the appendix, we collected the average and standard deviations. From these values, a 95% CI plot was created to observe if there were any significant differences in the average prediction accuracies for the ten random samples.

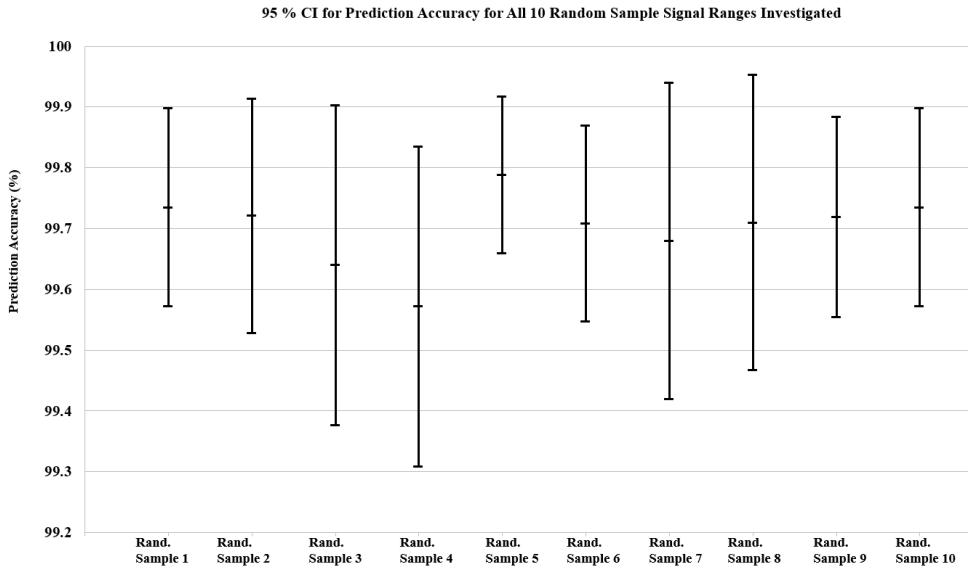

Figure 7.4 95% Confidence Interval Plot for Average Prediction Accuracy for the Ten Random Sample Ranges Investigated

From Figure 6.4, it is observed that none of the sample ranges produce prediction accuracies that were significantly different from prediction accuracies collected for the other random sample ranges investigated. This supports the hypothesis that the entire 60,000 observation signal is stationary, and the results obtained by using a given fraction of the signal at one location are consistent with the results obtained from using the same fraction of the signal at a different location.

CHAPTER VIII

DISCUSSION

8.1 Main Takeaways

8.1.1 FPCA for Feature Extraction and Dimensionality Reduction

The main aim of the presented work was to establish the efficacy of FPCA as a vibrational signal feature extraction technique because the application of FPCA within the realm of FDD is largely undocumented and the application of FPCA could provide a method that increases prediction accuracy while using less data. With our approach, we observed that analyzing 750 cases of labeled signals resulted in high performance cases across all signal duration levels as indicated by Table 7.3. Not only do the results of Table 7.3 highlight the importance of FPCA as an efficient feature extraction technique that helps to produce high MLP prediction accuracies, the table also highlights FPCA's capability of efficiently extracting significant discriminatory features for classification while using small fractions of the overall signal. The optimal signal duration fraction among the different signal duration levels investigated (500 observations long for 750 cases = 2500 observations from each raw signal) only required 2.5 seconds of the original one minute, 60,000 observation signals collected on each sensor.

Likewise, applying FPCA also helps to reduce the dimensionality of the original data. Initial acquisition of the PC functions that represent the original dimension space via

orthogonal functions that represent maximum variance functions is the first stage of reduction. Furthermore, specifying the desired explained variance level which selects a subset of the extracted PC functions reduces the dimensionality even further.

8.2 Comparison to Reports in Literature

The results presented in Table 8.1 show the overall averages of the prediction accuracy results of the present study.

Table 8.1 Confusion Matrix for Average ± SD for accuracy across all ranges investigated in study

Total Predicted Cases for Each State = 1500		Predicted				
		H	M1	M12	M2	VC
Actual	H	1500 ± 0 (100% ± 0%)	0 ± 0 (0% ± 0%)	0 ± 0 (0% ± 0%)	0 ± 0 (0% ± 0%)	0 ± 0 (0% ± 0%)
	M1	0 ± 0 (0% ± 0%)	1489.364 ± 2.767 (99.291% ± 0.184%)	0.182 ± 0.404 (0.0121% ± 0.0270%)	10.364 ± 2.656 (0.69% ± 0.177%)	0.091 ± 0.302 (0.006% ± 0.02%)
	M12	0 ± 0 (0% ± 0%)	0 ± 0 (0% ± 0%)	1500 ± 0 (100% ± 0%)	0 ± 0 (0% ± 0%)	0 ± 0 (0% ± 0%)
	M2	0 ± 0 (0% ± 0%)	11.818 ± 4.094 (0.788% ± 0.273%)	0 ± 0 (0% ± 0%)	1488.18 ± 4.094 (99.33% ± 0.273%)	0 ± 0 (0% ± 0%)
	VC	0 ± 0 (0% ± 0%)	0 ± 0 (0% ± 0%)	0 ± 0 (0% ± 0%)	0 ± 0 (0% ± 0%)	1500 ± 0 (100%)
Overall Prediction Accuracy Across All States = 99.724 Accuracy Predicting H = 100.00 % Accuracy Predicting M1 = 99.291 % Accuracy Predicting M12 = 100.00 % Accuracy Predicting M2 = 99.33 % Accuracy Predicting VC = 100.00 %						

Using the summary of the prediction accuracies presented in the bottom of Table 8.1, we expanded on the table originally presented by Jafarian et al to include additional references as well as the results we obtained with our study. Table 8.2 presents a comparison of the accuracies obtained in our study with similar reports in literature. From Table 8.2, it is shown that the approach proposed in this study outperformed most of the other reports presented in the table. The proposed approach can conclusively identify the states of a healthy engine, an engine with a dual misfire fault, and an engine with an abnormal valve clearance fault. Likewise, the approach can predict a specific single misfire fault with 99.291% accuracy for M1 faults and 99.33% accuracy for M2 faults. Because the accuracy performance is above 94 % in both cases, this approach is valid and reliable for detecting specific single misfire faults (Jafarian et al, 2018).

Table 8.2 Comparison of results to similar works in literature (Jafarian et al., 2018)

Source	Signal Type	Faults Investigated	Prediction Accuracy (%)
(Tay & Shen, 2003)	Vibrational Signal	Valve Clearance Faults	76.32
(Sharma, Sugumaran, & Babu Devasenapati, 2014b)	Vibrational Signal	Single Cylinder Misfire	89.40
(Jafari, Mehdigholi, & Behzad, 2014)	Acoustic Signal	Multiple Valve Clearance Faults	92.00
(Basir & Yuan, 2007)	Vibrational Signal Acoustic Signal Pressure Data Temperature Data	Valve Clearance Faults	94.00

Table 8.2 (continued)

(J.-D. Wu & Liu, 2009)	Acoustic Signal	Single Cylinder Misfire Dual Cylinder Misfire	92.00 97.00
(Boudaghi, Shahbakhti, & Jazayeri, 2015)	Engine Control Unit (ECU) Data	Single Cylinder Misfire Dual Cylinder Misfire	94.00 97.00
(Sharkey, Chandroth, & Sharkey, 2000)	Acoustic and Vibrational Signals	Cylinder Misfire and Valve Clearance Faults	95.83
(Ashkan Moosavian, Khazaee, Najafi, Kettner, & Mamat, 2015)	Acoustic and Vibrational Signals	Single Cylinder Misfire	98.56
(Li, Mi, Liu, & Wang, 2013)	Vibrational Signal	Valve Clearance Fault	98.20
(Yu, Junhong, Fengrong, Jiewei, & Wenpeng, 2015)	Vibrational Signal	Multiple Valve Clearance Faults	98.30
(Devasenapati., Ramachandran., & Sugumaran., 2010)	Vibrational Signal	Single Cylinder Misfire	98.40
(Shatnawi & Al-khassaweneh, 2014)	Acoustic Signal	Single Cylinder Misfire Dual Cylinder Misfire	100.00 90.00
(Flett & Bone, 2016)	Vibrational Signal	Valve Clearance Faults	99.95
(Jafarian et al.)	Vibrational Signal	Single Cylinder Misfire Dual Cylinder Misfire Valve Clearance Fault	97.34
This paper	Vibrational Signal	Single Cylinder Misfire Dual Cylinder Misfire Valve Clearance Fault	99.72

8.3 Prominent Modes of Misclassification

While the classification performance of the proposed network was highly reliable and valid, the occurrences of misclassification were largely localized to two types: misclassification of an M1 fault as an M2 and misclassification of an M2 fault as an M1. The interpretation of this phenomenon is that the PC features used in the MLP for these two classes must be similar to some degree. To provide visual support, the plots below in Figure 8.1 represent the PC scores across the PC functions that explain 95% of the original variance when collecting 750 cases of 500 observation signals from the beginning range of each signal (from observations 1-2500).

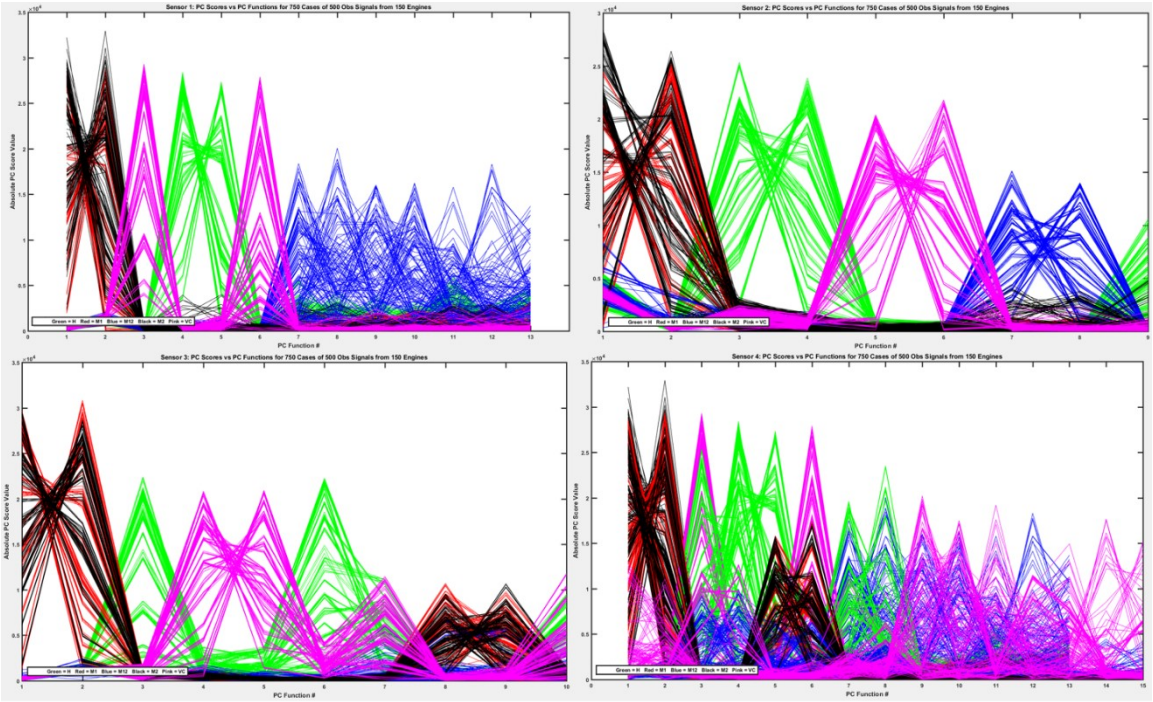

Figure 8.1 PC Scores vs PC Function # across all four sensors for each case (case number = 750)

From Figure 8.1, it is observed that the states H, M12, and VC can be differentiated from all other states based on their PC function scores. Filtering out states H, M12, and VC allows for the visualization of only states M1 and M2 in Figure 8.2.

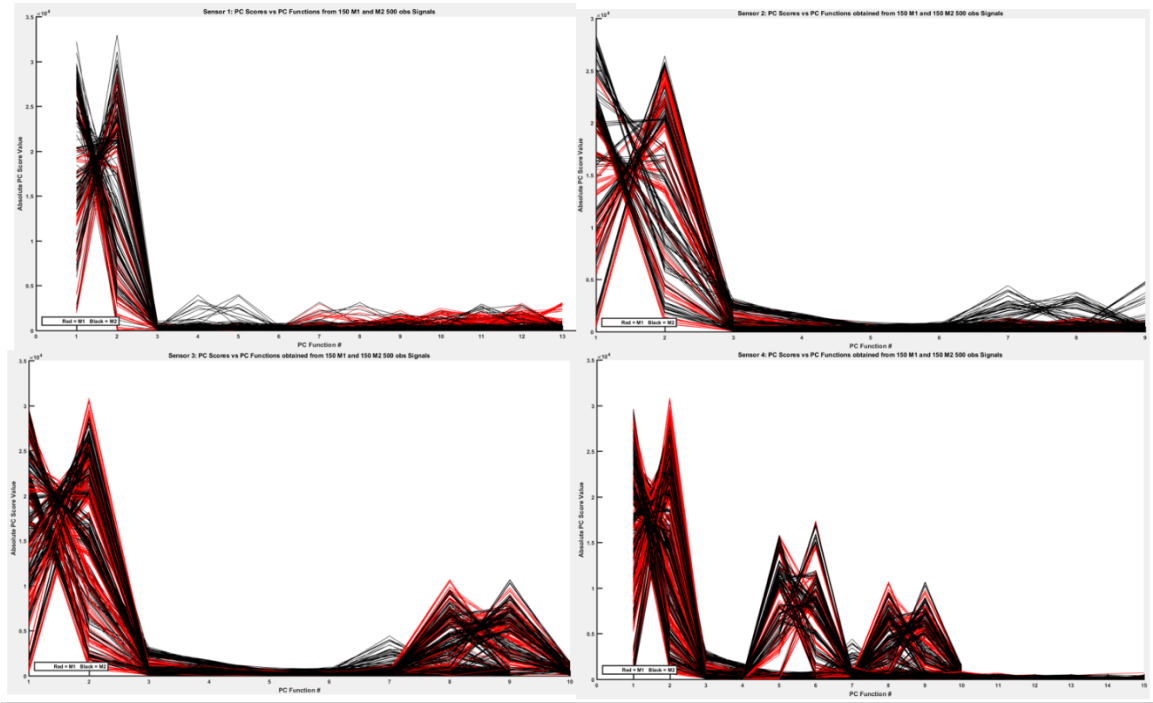

Figure 8.2 PC Scores vs PC Function # across all four sensors for each case (case number = 750)

In Figure 8.2, the PC function scores obtained for M1 and M2 signals appear to be very similar. However, according to Table 8.1, the classification performance results for predicting M1 and M2 were 99.29% and 99.33% respectively. Although the scores shown in Figure 8.2 look visually similar, the high prediction accuracies reported in Table 8.1 were possible because there were multiple significant discriminatory features captured in the PC sequences collected across all four sensors. These significant features are highlighted in Table 8.3.

Table 8.3 Paired T-test results of M1 PC_{ij} Score vs M2 PC_{ij} Score where i = PC function # and j = sensor #

PC Scores	Sensor 1 (First 13 PCs explain 95% of variance)		Sensor 2 (First 9 PCs explain 95% of variance)		Sensor 3 (First 10 PCs explain 95% of variance)		Sensor 4 (First 15 PCs explain 95% of variance)	
	P-value	Reject H_0 at $\propto$=0.05? 1: Yes 0: No	P-value	Reject H_0 at $\propto$=0.05? 1: Yes 0: No	P-value	Reject H_0 at $\propto$=0.05? 1: Yes 0: No	P-value	Reject H_0 at $\propto$=0.05? 1: Yes 0: No
1	0.960	0	**< 0.01**	**1**	0.819	0	0.143	0
2	0.526	0	0.152	0	0.435	0	**0.019**	**1**
3	0.313	0	**< 0.0001**	**1**	0.664	0	**< 0.0001**	**1**
4	**0.026**	**1**	**< 0.0001**	**1**	**< 0.0001**	**1**	**0.002**	**1**
5	**0.012**	**1**	0.060	0	0.012	1	0.152	0
6	0.483	0	**0.019**	**1**	0.431	0	**0.044**	**1**
7	0.767	0	**0.026**	**1**	**0.005**	**1**	0.802	0
8	0.056	0	0.444	0	**0.034**	**1**	**0.001**	**1**
9	0.137	0	**0.005**	**1**	0.175	0	0.636	0
10	0.405	0	-	-	0.293	0	**0.001**	**1**
11	0.223	0	-	-	-	-	**< 0.0001**	**1**
12	0.125	0	-	-	-	-	**0.004**	**1**
13	0.173	0	-	-	-	-	0.178	0
14	-	-	-	-	-	-	0.422	0
15	-	-	-	-	-	-	0.481	0

8.4 Effect of Misclassification

One of the main conclusions that can be drawn from the approach is that the occurrences of misclassifications in this study were localized to misclassification of an M1 misfire fault as an M2 misfire fault and vice versa. Most importantly, neither of these are a Type I or Type II error. Specifically, Type II errors could have severe implications in industrial applications by leading to higher costs associated with permanent engine damage

due to the failure to detect a fault. Therefore, it is up to the decision maker or practitioner that is responsible for applying cost coefficients and risks associated with each type of misclassification to ultimately determine their importance or relevance.

CHAPTER IX

FUTURE RESEARCH & CONCLUSION

9.1 Future Research

The approach presented in this thesis represents an exploratory study that investigated the feasibility of FPCA as a method for feature extraction for vibrational signals acquired from a collection of ICEs. The main aim of the thesis was to provide evidence to support that FPCA could be applied for the extraction of significant discriminatory features for each specific engine state investigated in the work. As indicated by the high prediction accuracy reported in Table 8.1, the approach investigated in this work was proven to be valid and reliable for the application. The results obtained from this study are extremely encouraging and support future research to expand on the current findings. Because of the exploratory nature of the proposed approach, all aspects associated with the approach can be investigated in more depth.

9.1.1 Optimization of Signal Extraction

The first aspect of the proposed approach that can be further investigated is the signal extraction aspect of the approach. Specifically, efforts can be devoted to establishing an approach to analyze signal parameters of interest such as the frequency, amplitude, and periodicity and determine the effect that these parameters have on the optimal size of the extracted portions to be used in FPCA.

9.1.2 Reducing Computational Bottlenecks of Curve Smoothing/Functional Data Conversion Operations

As mentioned previously, the goal of the approach proposed with this thesis is to achieve a method that allows for close to real-time analytics. Producing close to real-time analytics could allow for better on-line monitoring of process behavior which would allow operators overseeing the process to quickly respond when a fault has been detected. Acknowledging the importance that computational speed has on the end goal of achieving real-time analytics, it is critical to improve the computational bottlenecks in the proposed FDD approach.

In its current state, the process of curve smoothing where the raw data is converted into a functional state represents the largest computational bottleneck in the approach. If we observe Figure 9.1 and Figure 9.2, it is clear that increasing the signal duration results in an exponential increase in computation time.

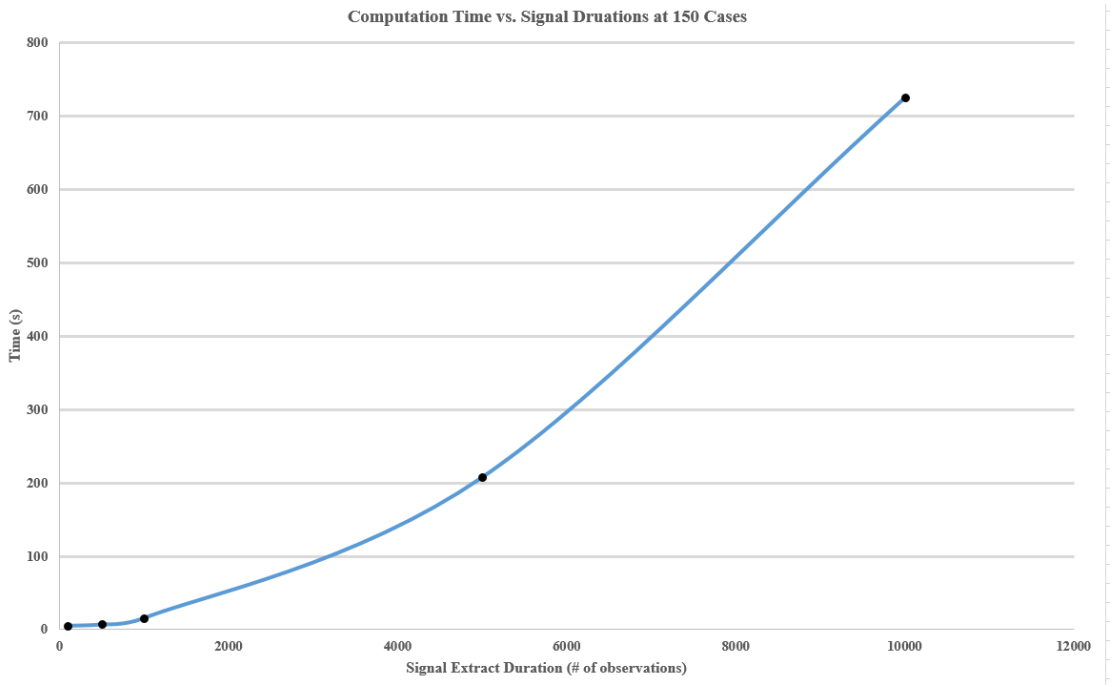

Figure 9.1 Computation Time vs. Signal Extract Duration (at 150 Cases)

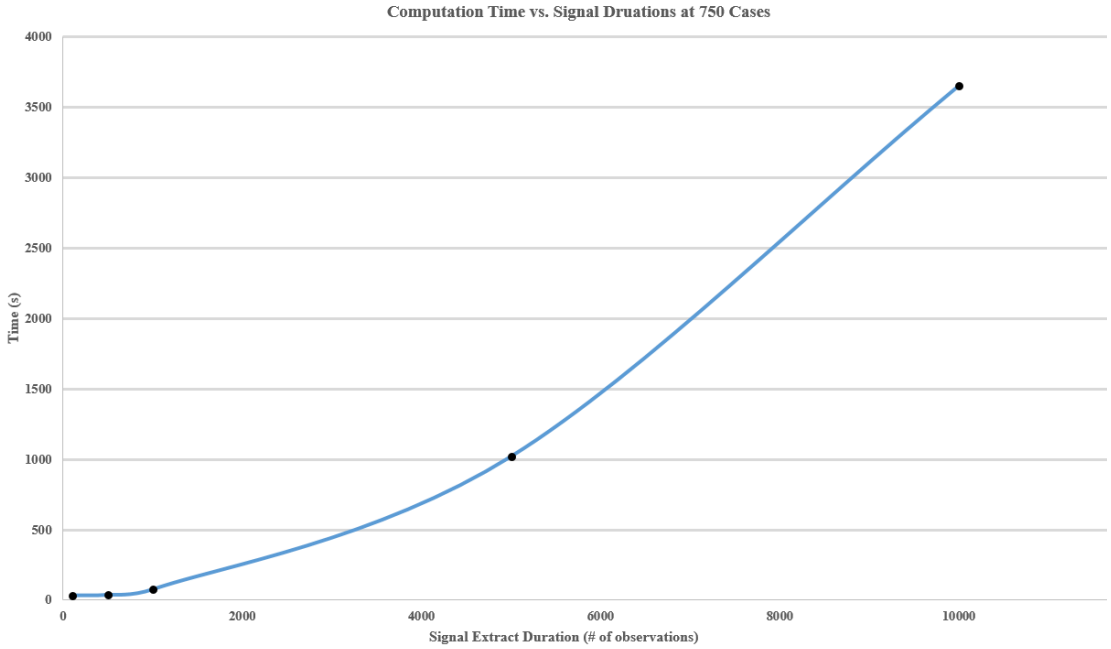

Figure 9.2 Computation Time vs. Signal Extract Duration (at 750 Cases)

The computational inefficiency was realized when signal extracts of size 5,000 observations and above were investigated. This phenomenon is reported in literature and has already been the subject for improvement (Ma, Huang, & Zhang, 2015; Xu & Wang, 2017; Yue, Simpson, Lindgren, & Rue, 2012) which shows potential for further investigation to optimize the curve fitting protocol followed in the present work.

9.1.3 Further Investigation of FPCA for Feature Extraction

As for the main subject of the study, the application of FPCA for feature extraction can be investigated further as well. One extension that can be made is to observe the effect of different explained variance ranges by performing a sensitivity analysis on the prediction accuracy based on the explained variance that is captured by the corresponding sequential range of PC functions across each sensor for each engine.

9.1.4 Functional Discriminant Analysis for Feature Extraction

One interesting avenue that can be developed more extensively is to investigate the feasibility of using a functional analogue of discriminant analysis for feature extraction for the vibrational signal data used in this work. An aspect of PCA in both the multivariate and functional context is that both forms represent unsupervised methods. This means that the methods ignore class labels when determining the directions or functions that maximize the variance in a data set. Because of this element, a calculated eigenvector or eigenfunction

that maximizes the variance in a dataset will not necessarily be an adequate measure for differentiating between classes as indicated in Figure 9.3.

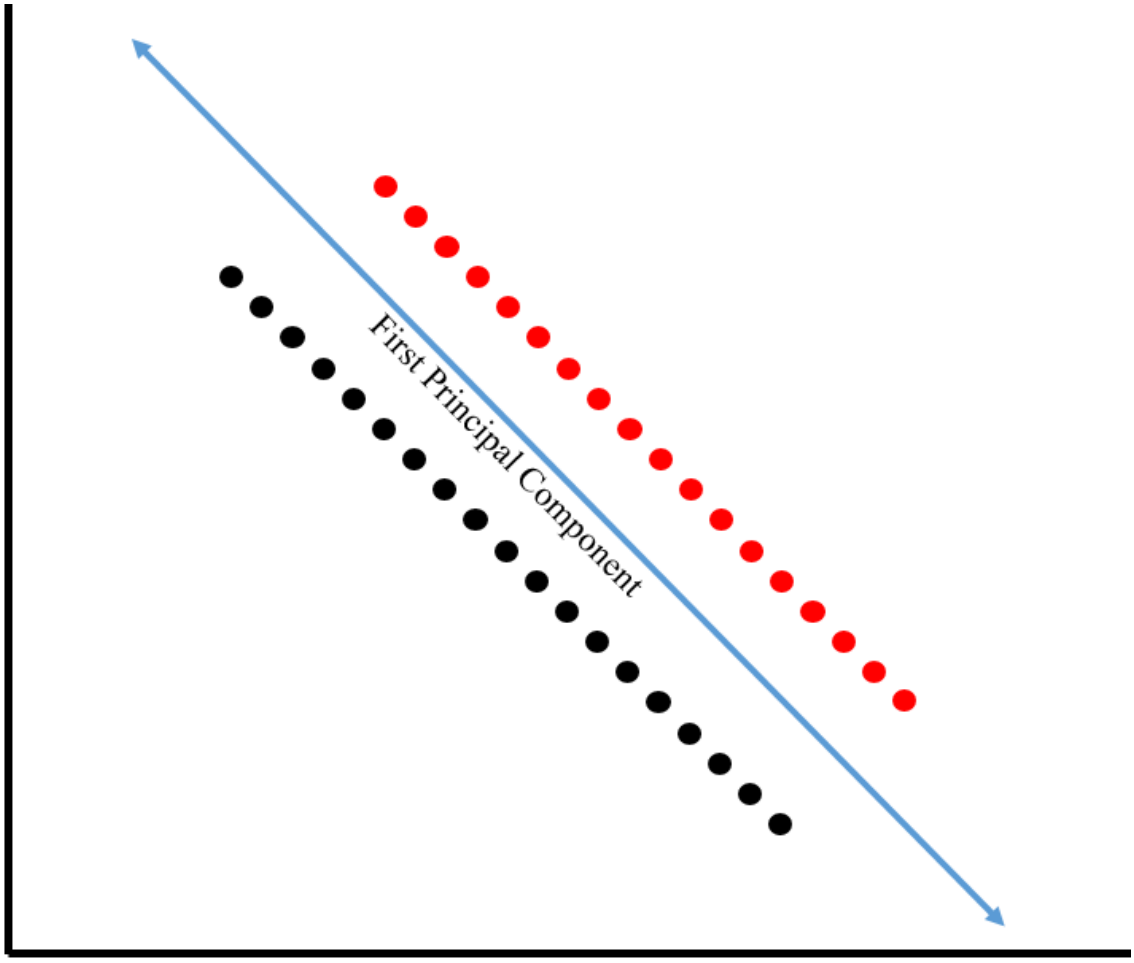

Figure 9.3 Example of the First Principal Component for an arbitrary dataset

On the other hand, discriminant analysis represents a supervised method that uses class labels when finding directions or functions that maximize the variance in the original dimensional space of the dataset. But because discriminant analysis accounts for class labels, the directions or functions that are determined are calculated in a way that

maximizes the variance between classes of data while minimizing the variation among data points that belong to the same class as indicated in Figure 9.4.

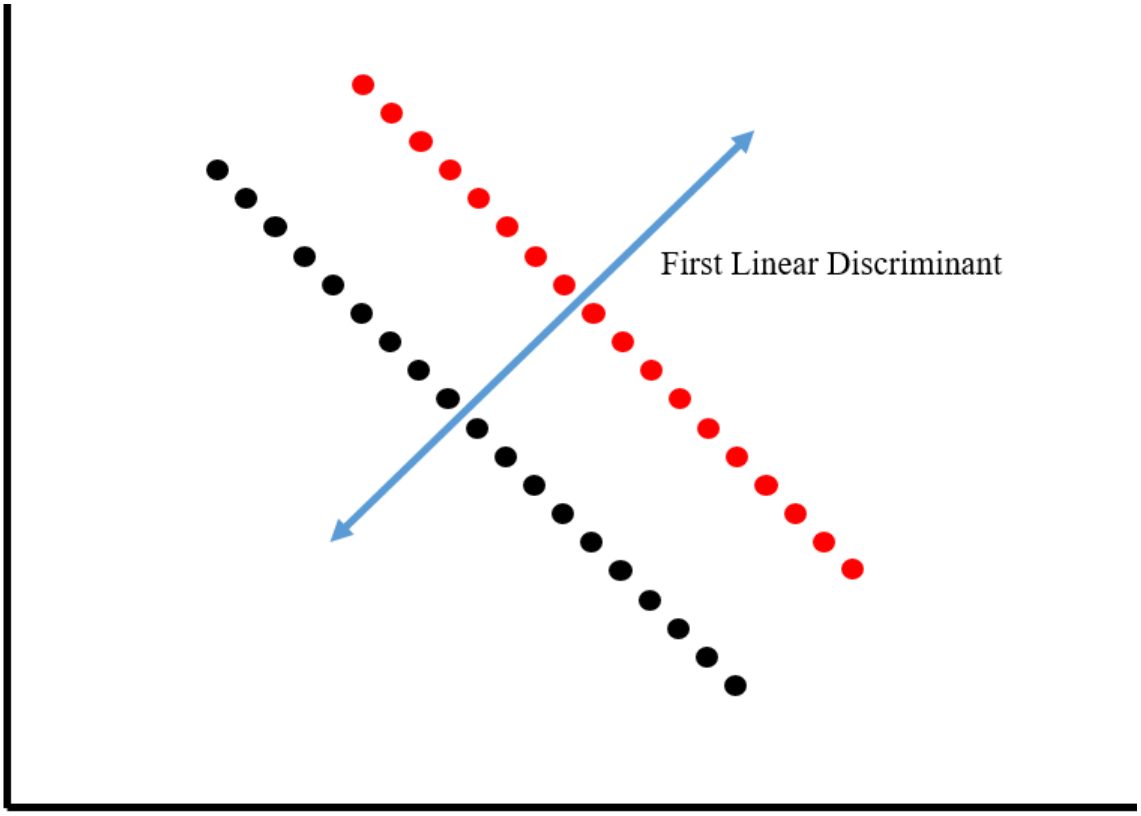

Figure 9.4 Example of the First Linear Discriminant for an arbitrary dataset

While FPCA performed exceptionally well for the application used in this research, the supervised nature of discriminant analysis may present discriminant directions or functions in sequence according to their between class discriminatory power in a more

convenient and concise manner for feature extraction. As well, additional efforts can be devoted to combining the power of FPCA and a functional form of discriminant analysis.

9.1.5 Alternate Classification Algorithms and Self-Organizing Maps

As the last aspect for the proposed data-driven approach for FDD, further investigation of alternate classification algorithms is possible. While the results obtained with the optimized MLP proved highly valid and reliable, it would be interesting to compare performance metrics with other algorithms similar to the approaches used by other authors (Jafarian et al., 2018). As well, utilization of Kohonen Self-Organizing Maps could provide visualization for the artificial neural network's learning and predictive capabilities (Kohonen, 1982). Analysis of the prominent plotting characteristics within self-organizing maps could provide deeper insights into the prominence and likelihood for all potential modes of misclassification.

However, proving the breadth of applicability for this approach towards other application for FDD is of main importance. Additional effort should be devoted to studying and validating the performance of this approach with data acquired from other applications (e.g. acoustic, electrical, and thermal).

9.2 Conclusion

The results collected from this study were highly encouraging. Applying FPCA to functional vibrational signals resulted in the extraction of significant features that predicted an engine' state out of five potential states with 99.72% accuracy. Likewise, this accuracy was obtained by using 4.167% of the original one-minute signal highlighting the approach's capability for quick detection. Similarly, FPCA provided an element of

dimensionality reduction by representing each functional curve as a vector of significant PC scores for classification.

The exploratory nature of the work allows for optimization of all the aspects discussed in the thesis which include but are not limited to the following: signal extraction, curve smoothing, feature extraction, feature selection, and classification. As well, confirming broad applicability of this approach for different applications with different forms of functional data would show support for industrial applications, specifically in a production environment where detecting deviations from normal process behavior in a correct and timely manner are of paramount importance.

REFERENCES

Al-Sheikh, H., & Moubayed, N. (2012). Fault detection and diagnosis of renewable energy systems: An overview. *2012 International Conference on Renewable Energies for Developing Countries (REDEC)*, 1–7. https://doi.org/10.1109/REDEC.2012.6416687

Aminian, F., & Aminian, M. (2001). Fault Diagnosis of Analog Circuits Using Bayesian Neural Networks with Wavelet Transform as Preprocessor. *Journal of Electronic Testing: Theory and Applications*, *17*, 29–36.

Basir, O., & Yuan, X. (2007). Engine fault diagnosis based on multi-sensor information fusion using Dempster–Shafer evidence theory. *Information Fusion*, *8*(4), 379–386. https://doi.org/10.1016/j.inffus.2005.07.003

Boudaghi, M., Shahbakhti, M., & Jazayeri, S. A. (2015). Misfire Detection of Spark Ignition Engines Using a New Technique Based on Mean Output Power. *Journal of Engineering for Gas Turbines and Power*, *137*(9), 091509. https://doi.org/10.1115/1.4029914

Breheny, P., & Huang, J. (2011). Coordinate descent algorithms for nonconvex penalized regression, with applications to biological feature selection. *Annals of Applied Statistics*, *5*(1), 232–253. https://doi.org/10.1214/10-AOAS388

Burns, D. M., Houpt, J. W., Townsend, J. T., & Endres, M. J. (2013). Functional principal components analysis of workload capacity functions. *Behavior Research Methods*, *45*(4), 1048–1057. https://doi.org/10.3758/s13428-013-0333-2

Chaves, R., Ramírez, J., Górriz, J. M., López, M., Salas-Gonzalez, D., Álvarez, I., & Segovia, F. (2009). SVM-based computer-aided diagnosis of the Alzheimer's disease using t-test NMSE feature selection with feature correlation weighting. *Neuroscience Letters*, *461*(3), 293–297. https://doi.org/10.1016/j.neulet.2009.06.052

Coffey, N., Harrison, A. J., Donoghue, O. A., & Hayes, K. (2011). Common functional principal components analysis: A new approach to analyzing human movement data. *Human Movement Science*, *30*(6), 1144–1166. https://doi.org/10.1016/j.humov.2010.11.005

Dean, J. A., Wong, K. H., Gay, H., Welsh, L. C., Jones, A.-B., Schick, U., … Gulliford, S. L. (2016). Functional Data Analysis Applied to Modeling of Severe Acute Mucositis and Dysphagia Resulting From Head and Neck Radiation Therapy. *International Journal of Radiation Oncology*Biology*Physics*, *96*(4), 820–831. https://doi.org/10.1016/j.ijrobp.2016.08.013

Devasenapati., B. S., Ramachandran., K. ., & Sugumaran., V. (2010). Misfire Detection in a Spark Ignition Engine using Support Vector Machines. *International Journal of Computer Applications*, *5*(6), 25–29. https://doi.org/10.5120/917-1295

Díaz-Rodríguez, P., Cancilla, J. C., Matute, G., Chicharro, D., & Torrecilla, J. S. (2015). Inputting molecular weights into a multilayer perceptron to estimate refractive indices of dialkylimidazolium-based ionic liquids—A purity evaluation. *Applied Soft Computing*, *28*, 394–399. https://doi.org/10.1016/j.asoc.2014.12.004

Di Salvo, F., Ruggieri, M., & Plaia, A. (2015). Functional principal component analysis for multivariate multidimensional environmental data. *Environmental and Ecological Statistics*, *22*(4), 739–757. https://doi.org/10.1007/s10651-015-0317-8

Farajzadeh-Zanjani, M., Razavi-Far, R., Saif, M., & Rueda, L. (2016). Efficient feature extraction of vibration signals for diagnosing bearing defects in induction motors. *Proceedings of the International Joint Conference on Neural Networks*, *2016-Octob*, 4504–4511. https://doi.org/10.1109/IJCNN.2016.7727789

Ferguson, C., & Kirkpatrick, A. (2015). *Internal Combustion Engines: Applied Thermosciences, 3ed* (3rd ed.). John Wiley & Sons.

Flett, J., & Bone, G. M. (2015). Fault detection and diagnosis of diesel engine valve trains. https://doi.org/10.1016/j.ymssp.2015.10.024

Flett, J., & Bone, G. M. (2016). Fault detection and diagnosis of diesel engine valve trains. *Mechanical Systems and Signal Processing*, *72–73*, 316–327. https://doi.org/10.1016/j.ymssp.2015.10.024

Ftoutou, E., Chouchane, M., & Besbès, N. (2012). Internal combustion engine valve clearance fault classification using multivariate analysis of variance and discriminant analysis. *Transactions of the Institute of Measurement and Control*, *34*(5), 566–577. https://doi.org/10.1177/0142331211408492

Gertler, J. (1998). *Fault detection and diagnosis in engineering systems*. Marcel Dekker.

Gertler, J. (2013). Fault Detection and Diagnosis. In J. Baillieul & T. Samad (Eds.), *Encyclopedia of Systems and Control* (pp. 1–7). London: Springer London. https://doi.org/10.1007/978-1-4471-5102-9_223-1

Gong, M., Miller, C., & Scott, M. (2015). Functional PCA for Remotely Sensed Lake Surface Water Temperature Data. *Procedia Environmental Sciences*, *26*, 127–130. https://doi.org/10.1016/j.proenv.2015.05.015

Hagan, M. T., & Menhaj, M. B. (1994). Training feedforward networks with the Marquardt algorithm. *IEEE Transactions on Neural Networks*, *5*(6), 989–993. https://doi.org/10.1109/72.329697

Huynh, K. P., Jacho-Chávez, D. T., Petrunia, R. J., & Voia, M. (2011). Functional Principal Component Analysis of Density Families With Categorical and Continuous Data on Canadian Entrant Manufacturing Firms. *Journal of the American Statistical Association*, *106*(495), 858–878. https://doi.org/10.1198/jasa.2011.ap10111

Jack, L., & Nandi, A. (2000). Genetic algorithms for feature selection in machine condition monitoring with vibration signals. *Vision, Image and Signal Processing, IEEE Proceedings*, *147*(3), 205–212. https://doi.org/10.1049/ip-vis:20000325

Jafari, S. M., Mehdigholi, H., & Behzad, M. (2014). Valve Fault Diagnosis in Internal Combustion Engines Using Acoustic Emission and Artificial Neural Network. *Shock and Vibration*, *2014*, 1–9. https://doi.org/10.1155/2014/823514

Jafarian, K., Mobin, M., Jafari-marandi, R., & Rabiei, E. (2018). Misfire and Valve Clearance Faults Detection in the Combustion Engines Based on a Multi-Sensor Vibration Signal Monitoring. *Measurement*, *128*(November 2017), 527–536. https://doi.org/10.1016/j.measurement.2018.04.062

Jafari-Marandi, R., Davarzani, S., Soltanpour Gharibdousti, M., & Smith, B. K. (2018). An optimum ANN-based breast cancer diagnosis: Bridging gaps between ANN learning and decision-making goals. *Applied Soft Computing*, *72*, 108–120. https://doi.org/10.1016/j.asoc.2018.07.060

Jafari-Marandi, R., Khanzadeh, M., Smith, B. K., & Bian, L. (2017). Self-Organizing and Error Driven (SOED) artificial neural network for smarter classifications. *Journal of Computational Design and Engineering*, *4*(4), 282–304. https://doi.org/10.1016/j.jcde.2017.04.003

Jiang, Z., Mao, Z., Wang, Z., & Zhang, J. (2017). Fault diagnosis of internal combustion engine valve clearance using the impact commencement detection method. *Sensors (Switzerland)*, *17*(12). https://doi.org/10.3390/s17122916

Jung, D., Eriksson, L., Frisk, E., & Krysander, M. (2014). Development of misfire detection algorithm using quantitative FDI performance analysis $. https://doi.org/10.1016/j.conengprac.2014.10.001

Kannan, kavita &. (2016). An Efficient Framework for Heart Disease Classification using Feature Extraction and Feature Selection Technique in Data Mining. *Ieee*, 1–5. https://doi.org/10.1109/ICETETS.2016.7603000

Khalid, S., Khalil, T., & Nasreen, S. (2014). A survey of feature selection and feature extraction techniques in machine learning. *2014 Science and Information Conference*, 372–378. https://doi.org/10.1109/SAI.2014.6918213

Khanzadeh, M., Chowdhury, S., Marufuzzaman, M., Tschopp, M. A., & Bian, L. (2018). Porosity prediction: Supervised-learning of thermal history for direct laser deposition. *Journal of Manufacturing Systems*, *47*, 69–82. https://doi.org/10.1016/j.jmsy.2018.04.001

Kiencke, U. (1999). *Engine misfire detection*. *Control Engineering Practice* (Vol. 7).

Kim, C., Han, S., & Moon, S. (2000). The Effect of Engine Misfire on Catalytic Converter and vehicle Emissions. 1) ,2),3), 2–6.

Knerr, S., Personnaz, L., & Dreyfus, G. (1992). Handwritten digit recognition by neural networks with single-layer training. *IEEE Transactions on Neural Networks / a Publication of the IEEE Neural Networks Council*, *3*(6), 962–968. https://doi.org/10.1109/72.165597

Kohonen, T. (1982). Self-organized formation of topologically correct feature maps. *Biological Cybernetics*, *43*(1), 59–69. https://doi.org/10.1007/BF00337288

Kotsiantis, S. B., Zaharakis, I. D., & Pintelas, P. E. (2006). Machine learning: A review of classification and combining techniques. *Artificial Intelligence Review*, *26*(3), 159–190. https://doi.org/10.1007/s10462-007-9052-3

Kotsiantis, Sotiris B. (2007). Supervised Machine Learning: A Review of Classification Techniques. *Informatica*, *31*, 249–268. https://doi.org/10.1115/1.1559160

Krzywonos, L. (2015). Application of vibration signal in the diagnosis of IC engine valve clearance, (October 2017).

Lei, Q., Tajammal Munir, M., Bao, J., & Young, B. (2016). A Data-driven Fault Detection Method Based on Dissipative Trajectories. *IFAC-PapersOnLine*, *49*(7), 717–722. https://doi.org/10.1016/j.ifacol.2016.07.266

Li, B., Mi, S., Liu, P., & Wang, Z. (2013). Classification of time-frequency representations using improved morphological pattern spectrum for engine fault diagnosis. *Journal of Sound and Vibration*, *332*(13), 3329–3337. https://doi.org/10.1016/j.jsv.2013.01.017

Lindemann, M., Filbert, D., Automatisierungstechnik, M.-, Sekretariat, E., Berlin, T. U., & Berlin, D.-. (2000). SIGNAL COMPONENT ANALYSIS FOR MISFIRE DETECTION. *XVI IMEKO World Congress*.

Liu, C., Ray, S., & Hooker, G. (2017). Functional principal component analysis of spatially correlated data. *Statistics and Computing*, *27*(6), 1639–1654. https://doi.org/10.1007/s11222-016-9708-4

Liu, P., Li, B., Han, C., & Wang, F. (2016). Feature Extraction and Selection Scheme for Intelligent Engine Fault Diagnosis Based on 2DNMF, Mutual Information, and NSGA-II. *Shock and Vibration*, *2016*, 1–13. https://doi.org/10.1155/2016/3975285

Ma, P., Huang, J. Z., & Zhang, N. (2015). Efficient computation of smoothing splines via adaptive basis sampling. *Biometrika*, *102*(3), 631–645. https://doi.org/10.1093/biomet/asv009

Malhi, A., & Gao, R. X. (2004). PCA-based feature selection scheme for machine defect classification. *IEEE Transactions on Instrumentation and Measurement*, *53*(6), 1517–1525. https://doi.org/10.1109/TIM.2004.834070

Moosavian, A., Najafi, G., Nadimi, H., & Arab, M. (2017). Estimation of engine friction using vibration analysis and artificial neural network. *2017 International Conference on Mechanical, System and Control Engineering, ICMSC 2017*, 130–135. https://doi.org/10.1109/ICMSC.2017.7959457

Moosavian, Ashkan, Khazaee, M., Najafi, G., Kettner, M., & Mamat, R. (2015). Spark plug fault recognition based on sensor fusion and classifier combination using Dempster–Shafer evidence theory. *Applied Acoustics*, *93*, 120–129. https://doi.org/10.1016/j.apacoust.2015.01.008

Moreno-Oyervides, A., Lopez, J. P., Martin-Mateos, P., Walla, F., Criado, R., Krozer, V., & Acedo, P. (2017). Use of functional principal components analysis in CW subTHz spectroscopy for hydrocarbon emulsified water assessment. In *2017 42nd International Conference on Infrared, Millimeter, and Terahertz Waves (IRMMW-THz)* (pp. 1–2). Cancun, Mexico: IEEE. https://doi.org/10.1109/IRMMW-THz.2017.8067185

Muralidharan, V., & Sugumaran, V. (2012). A comparative study of Naïve Bayes classifier and Bayes net classifier for fault diagnosis of monoblock centrifugal pump using wavelet analysis. *Applied Soft Computing Journal*, *12*(8), 2023–2029. https://doi.org/10.1016/j.asoc.2012.03.021

Muralidharan, V., & Sugumaran, V. (2013). Rough set based rule learning and fuzzy classification of wavelet features for fault diagnosis of monoblock centrifugal pump. *Measurement: Journal of the International Measurement Confederation*, *46*(9), 3057–3063. https://doi.org/10.1016/j.measurement.2013.06.002

Naik, S. (2004). Advanced misfire detection using adaptive signal processing. *International Journal of Adaptive Control and Signal Processing*, *18*(2), 181–198. https://doi.org/10.1002/acs.789

Nicol, F. (2013). Functional principal component analysis of aircraft trajectories. *ISIATM 2013*, 17.

Osburn, A. W., Kostek, T. M., Franchek, M. A., & Franchek, M. A. (2005). Mechanical Systems and Signal Processing Residual generation and statistical pattern recognition for engine misfire diagnostics. https://doi.org/10.1016/j.ymssp.2005.06.002

Pollock, D. S. G. (1999). Smoothing with Cubic Splines. *Handbook of Time Series Analysis, Signal Processing, and Dynamics*, 293–322. https://doi.org/10.1016/B978-012560990-6/50013-0

Ramsay, J., & Silverman, B. W. (2005). *Functional Data Analysis* (2nd ed.). New York: Springer-Verlag. Retrieved from //www.springer.com/us/book/9780387400808

Saimurugan, M., Ramachandran, K. I., Sugumaran, V., & Sakthivel, N. R. (2011). Multi component fault diagnosis of rotational mechanical system based on decision tree and support vector machine. *Expert Systems with Applications*, *38*(4), 3819–3826. https://doi.org/10.1016/j.eswa.2010.09.042

Sánchez-Sánchez, M., Belda-Lois, J.-M., Mena-del Horno, S., Viosca-Herrero, E., Gisbert-Morant, B., Igual-Camacho, C., & Bermejo-Bosch, I. (2014). Functional principal component analysis as a new methodology for the analysis of the impact of two rehabilitation protocols in functional recovery after stroke. *Journal of NeuroEngineering and Rehabilitation*, *11*(1), 134. https://doi.org/10.1186/1743-0003-11-134

Saravanan, N., Cholairajan, S., & Ramachandran, K. I. (2009). Vibration-based fault diagnosis of spur bevel gear box using fuzzy technique. *Expert Systems with Applications*, *36*(2 PART 2), 3119–3135. https://doi.org/10.1016/j.eswa.2008.01.010

Seshadrinath, J., Singh, B., & Panigrahi, B. K. (2014). Investigation of vibration signatures for multiple fault diagnosis in variable frequency drives using complex wavelets. *IEEE Transactions on Power Electronics*, *29*(2), 936–945. https://doi.org/10.1109/TPEL.2013.2257869

Shardlow, M. (2016). An Analysis of Feature Selection Techniques. *The University of Manchester*, 1–7.

Sharkey, A. J. C., Chandroth, G. O., & Sharkey, N. E. (2000). Acoustic emission, cylinder pressure and vibration: a multisensor approach to robust fault diagnosis. In *Proceedings of the IEEE-INNS-ENNS International Joint Conference on Neural Networks. IJCNN 2000. Neural Computing: New Challenges and Perspectives for the New Millennium* (pp. 223–228 vol.6). Como, Italy: IEEE. https://doi.org/10.1109/IJCNN.2000.859400

Sharma, A., Sugumaran, V., & Babu Devasenapati, S. (2014a). Misfire detection in an IC engine using vibration signal and decision tree algorithms. *Measurement: Journal of the International Measurement Confederation*, *50*(1), 370–380. https://doi.org/10.1016/j.measurement.2014.01.018

Sharma, A., Sugumaran, V., & Babu Devasenapati, S. (2014b). Misfire detection in an IC engine using vibration signal and decision tree algorithms. *Measurement*, *50*, 370–380. https://doi.org/10.1016/j.measurement.2014.01.018

Shatnawi, Y., & Al-khassaweneh, M. (2014). Fault Diagnosis in Internal Combustion Engines Using Extension Neural Network. *IEEE Transactions on Industrial Electronics*, *61*(3), 1434–1443. https://doi.org/10.1109/TIE.2013.2261033

Smoothing Splines - MATLAB & Simulink. (n.d.). Retrieved September 17, 2018, from https://www.mathworks.com/help/curvefit/smoothing-splines.html

Sugumaran, V., Muralidharan, V., & Ramachandran, K. I. (2007). Feature selection using Decision Tree and classification through Proximal Support Vector Machine for fault diagnostics of roller bearing. *Mechanical Systems and Signal Processing*, *21*(2), 930–942. https://doi.org/10.1016/j.ymssp.2006.05.004

Tay, F. E. H., & Shen, L. (2003). Fault diagnosis based on Rough Set Theory. *Engineering Applications of Artificial Intelligence*, *16*(1), 39–43. https://doi.org/10.1016/S0952-1976(03)00022-8

Ullah, S., & Finch, C. F. (2013). Applications of functional data analysis: A systematic review. *BMC Medical Research Methodology*, *13*(1). https://doi.org/10.1186/1471-2288-13-43

Unal, M., Onat, M., Demetgul, M., & Kucuk, H. (2014). Fault diagnosis of rolling bearings using a genetic algorithm optimized neural network. *Measurement*. https://doi.org/10.1016/j.measurement.2014.08.041

Viviani, R., Grön, G., & Spitzer, M. (2005). Functional principal component analysis of fMRI data. *Human Brain Mapping*, *24*(2), 109–129. https://doi.org/10.1002/hbm.20074

Warmenhoven, J., Cobley, S., Draper, C., Harrison, A., Bargary, N., & Smith, R. (2017). Considerations for the use of functional principal components analysis in sports biomechanics: examples from on-water rowing. *Sports Biomechanics*, *0*(0), 1–25. https://doi.org/10.1080/14763141.2017.1392594

Wu, J. Da, & Liu, C. H. (2008). Investigation of engine fault diagnosis using discrete wavelet transform and neural network. *Expert Systems with Applications*, *35*(3), 1200–1213. https://doi.org/10.1016/j.eswa.2007.08.021

Wu, J.-D., & Liu, C.-H. (2009). An expert system for fault diagnosis in internal combustion engines using wavelet packet transform and neural network. *Expert Systems with Applications*, *36*(3), 4278–4286. https://doi.org/10.1016/j.eswa.2008.03.008

Wuxing, L., Tse, P. W., Guicai, Z., & Tielin, S. (2004). Classification of gear faults using cumulants and the radial basis function network. *Mechanical Systems and Signal Processing*, *18*(2), 381–389. https://doi.org/10.1016/S0888-3270(03)00080-3

Xu, D., & Wang, Y. (2017). Divide and Recombine Approaches for Fitting Smoothing Spline Models with Large Datasets. *Journal of Computational and Graphical Statistics*, *0*(0), 1–7. https://doi.org/10.1080/10618600.2017.1402775

Yu, L., Junhong, Z., Fengrong, B., Jiewei, L., & Wenpeng, M. (2015). A fault diagnosis approach for diesel engine valve train based on improved ITD and SDAG-RVM. *Measurement Science and Technology*, *26*(2), 025003. https://doi.org/10.1088/0957-0233/26/2/025003

Yue, Y. R., Simpson, D., Lindgren, F., & Rue, H. (2012). Bayesian Adaptive Smoothing Spline using Stochastic Differential Equations. *ArXiv:1209.2013 [Math, Stat]*. Retrieved from http://arxiv.org/abs/1209.2013

Zhan, Y. L., Shi, Z. Bin, Shwe, T., & Wang, X. Z. (2007). Fault diagnosis of marine main engine cylinder cover based on vibration signal. *Proceedings of the Sixth International Conference on Machine Learning and Cybernetics, ICMLC 2007*, *2*(August), 1126–1130. https://doi.org/10.1109/ICMLC.2007.4370313

Zhou, N., & Wang, L. (2007). A Modified T-test Feature Selection Method and Its Application on the HapMap Genotype Data. *Genomics, Proteomics and Bioinformatics*, *5*(3–4), 242–249. https://doi.org/10.1016/S1672-0229(08)60011-X

APPENDIX A

K-FOLD CROSS VALIDATION RESULTS FOR RANDOM RANGES INVESTIGATED IN STATIONARY SIGNAL VALIDATION PHASE OF ANALYSIS

Table A.1 K-Fold CV results of Random Ranges 1-4

	Range 1 (23,979 – 26,479)			Range 2 (41,919 – 44,419)			Range 3 (29,552 – 32,052)			Range 4 (24,479 – 26,979)		
Network Index	Acc. (%)	Error Count	Error ID	Acc. (%)	Error Count	Error ID	Acc. (%)	Error Count	Error ID	Acc.	Error Count	Error ID
1	99.87	1	14	99.87	1	14	99.33	5	14, 7, 14, 7, 14	99.20	6	14, 7, 7, 14, 14, 14
2	99.47	4	7, 7, 14, 7	99.60	3	14, 7, 7	99.20	6	14, 14, 7, 7, 7, 7	100.00	0	-
3	99.73	2	14, 7	99.60	3	7, 14, 14	99.73	2	14, 14	99.33	5	7, 14, 14, 7, 7
4	100.00	0	-	100.00	0	-	100	0	-	100.00	0	-
5	99.33	5	14, 14, 7, 7, 7	100.00	0	-	99.87	1	14	98.40	12	14, 7, 7, 14, 14, 14, 7, 14, 7, 14, 14, 14
6	99.47	4	7, 7, 14, 14	99.20	6	7, 7, 7, 14, 7, 7	99.60	3	14, 7, 14	99.73	2	7, 7
7	99.87	1	14	99.60	3	14, 7, 7	99.87	1	7	100.00	0	-
8	99.87	1	14	99.87	1	14	99.87	1	14	99.73	2	14, 14
9	99.87	1	14	100.00	0	-	100.00	0	-	99.73	2	14, 7
10	99.87	1	7	99.47	4	14, 6, 7, 14	98.93	8	14, 7, 14, 14, 7, 8, 7, 14	99.60	3	7, 14, 14

Table A.2 K-Fold CV results of Random Ranges 5-8

	Range 5 (12,840 – 15,340)			Range 6 (12,765 – 15,265)			Range 7 (50,033 – 52533)			Range 8 (50,223 – 52,723)		
Network Index	Acc. (%)	Error Count	Error ID	Acc. (%)	Error Count	Error ID	Acc. (%)	Error Count	Error ID	Acc.	Error Count	Error ID
1	99.60	3	14, 14, 14	99.60	3	14, 14, 14	99.60	3	14, 14, 14	99.87	1	14
2	99.87	1	7	99.33	5	7, 7, 14, 7, 14	98.93	8	7, 14, 14, 14, 7, 7, 14, 14	99.33	5	14, 7, 7, 7, 7
3	99.87	1	7	99.87	1	14	99.73	2	7, 14	99.73	2	7, 14
4	100.00	0	-	100.00	0	-	100.00	0	-	100.00	0	-
5	99.73	2	14, 7	99.47	4	14, 14, 7, 14	99.87	1	7	99.60	3	14, 7, 7
6	99.47	4	7, 7, 7, 7	99.47	4	7, 7, 14, 7	99.87	1	14	100.00	0	-
7	99.87	1	14	99.73	2	7, 14	100.00	0	-	99.73	2	7, 7
8	99.87	1	14	99.87	1	14	99.60	3	14, 7, 14	98.93	8	14, 7, 14, 7, 7, 7, 7, 7
9	100.00	0	-	99.87	1	7	100.00	0	-	100.00	0	-
10	99.60	3	14, 7, 14	99.87	1	7	99.20	6	14, 14, 14, 14, 14, 14	99.87	1	7

Table A.3 K-Fold CV results of Random Ranges 9 and 10

	Range 9 (37,397 – 39,897)			Range 10 (5,609 – 8,109)		
Network Index	Acc. (%)	Error Count	Error ID	Acc. (%)	Error Count	Error ID
1	100.00	0	-	99.87	1	14
2	99.60	3	7, 7, 7	99.47	4	7, 7, 14, 7
3	99.73	2	14, 14	99.73	2	14, 7
4	100	0	-	100	0	-
5	99.33	5	14, 14, 14, 7, 14	99.33	5	14, 14, 7, 7, 7
6	99.60	3	7, 7, 7	99.47	4	7, 7, 14, 14
7	99.73	2	14, 14	99.87	1	14
8	100.00	0	-	99.87	1	14
9	99.73	2	14, 7	99.87	1	14
10	99.47	4	14, 14, 14, 14	99.87	1	7