Intelligent Condition Assessment of Power Transformers

UC
UNIVERSITY OF CANTERBURY
Te Whare Wānanga o Waitaha
CHRISTCHURCH NEW ZEALAND

Abdolrahman Peimankar

Department of Electrical and Computer Engineering

University of Canterbury

A thesis submitted for the degree of

Doctor of Philosophy

2017

To Fatemeh
and
my family, Florya, Jamil, and Vahedeh.

Abstract

This thesis begins by providing an introduction to different transformer failures and the most effective condition monitoring techniques. Different failures are introduced and their corresponding fault diagnosis methods are listed to have a better understanding of failure modes and their consequence effects. An investigation into monitoring major failures of transformers using dissolved gas analysis is then presented. Various conventional, dissolved gas analysis based, fault diagnosis techniques are presented and the drawbacks of these methods are discussed. Intelligent fault diagnosis methods are introduced to overcome the problems of the conventional techniques. An overview of statistical and machine learning algorithms applied in this research is also described.

Preliminary research results on transformer load tap changers fault classification are reported. A hierarchical fault diagnosis algorithm for transformer load tap changers using support vector machines is used, in which, for each fault class, a unique single support vector machine algorithm is employed. However, while the developed algorithm is reasonably accurate, the shortcomings of applying single learning algorithms are discussed and a proposal for developing a more robust and generalised transformers condition assessment algorithm is made.

An intelligent power transformer fault diagnosis algorithm is then developed to classify faults of transformers. The proposed fault diagnosis algorithm is an ensemble-based approach which uses different statistical and machine learning algorithms. In the first phase of the proposed algorithm the most relevant features (dissolved gases) corresponding to each fault class are first determined. Then, selected features are used to classify transformer faults. The results of this algorithm show a significant improvement, in terms of classification.

A time-series forecasting algorithm is developed to predict future values of dissolved gases in transformers. The dataset for this algorithm was collected from a transformer for a period of six months which consisted of seven dissolved gases, a loading history, and three measured, ambient, oil, and winding, temperatures of transformer. The correlation coefficients between these 11 time series are then calculated and a non-linear principle component analysis is used to extract an effective time series from highly correlated variables. The proposed multi-objective evolutionary time series forecasting algorithm selects the most accurate and diverse group of forecasting methods among various implemented time series forecasting algorithms. The proposed method is also compared with other conventional time series forecasting algorithms and the results show the improvements over the different forecasting horizons.

List of Publications

The following is a list of papers that have been published/submitted for publication during the course work of this research.

Journal Papers

Abdolrahman Peimankar, Stephen John Weddell, Thahirah Jalal, Andrew Craig Lapthorn. ‘Evolutionary Multi-Objective Fault Diagnosis of Power Transformers’, *Swarm and Evolutionary Computation*, ISSN 2210-6502, 2017. (https://doi.org/10.1016/j.swevo.2017.03.005)

Abdolrahman Peimankar, Stephen John Weddell, Thahirah Jalal, Andrew Craig Lapthorn. ‘Multi-Objective Ensemble Forecasting with an Application to Power Transformers’, submitted to *Applied Soft Computing - Elsevier* on 10/08/2017.

Conference Papers

Abdolrahman Peimankar, Stephen John Weddell, Thahirah Jalal, Andrew Craig Lapthorn. ‘Ensemble Classifier Selection Using Multi-Objective PSO for Fault Diagnosis of Power Transformers’, In *Proceedings of the 2016 IEEE Congress on Evolutionary Computation (CEC)*, 2016, pp. 3622-3629. (https://doi.org/10.1109/CEC.2016.7744248)

Abdolrahman Peimankar, Andrew Craig Lapthorn. ‘Condition Assessment of Transformers Load Tap Changers Using Support Vector Machine’, In *the 19th International Symposium on High Voltage Engineering (ISH 2015)*, Pilsen, Czech Republic, 23-28 August 2015.

Other Presentations

A poster presented in the Optimisation and Statistics in Data Science (OSDS) workshop titled 'Multi-Objective Ensemble Forecasting: An Application to Power Transformers', University of Canterbury, Christchurch, New Zealand, 22 November 2016.

An oral presentation titled 'Intelligent Transformer Management' at the Unison Networks Limited (UNL) post-graduate students workshop, Hastings, New Zealand, 31 March 2016.

An oral presentation titled 'Ensemble Classifier Selection Using Multi-Objective PSO for Fault Diagnosis of Power Transformers' at the 2016 IEEE World Congress on Computational Intelligence, Vancouver BC, Canada, 28 July 2016.

A poster presented in the Electric Power Engineering Centre (EPECentre) R&D Expo titled 'A General Intelligent Asset Management Tool for Transformers.', University of Canterbury, Christchurch, New Zealand, 23 September 2015.

Acknowledgements

I would like to express my sincere gratitude to my supervisor Dr Andrew Lapthorn for his priceless support, guidance, and advice through the course of my research.

I would also like to thank my co-supervisor Dr Stephen Weddell for his motivating remarks and constructive advices. His office door has been always open for our enjoyable discussions.

I must acknowledge the financial support of Ministry of Business, Innovation, and Employment, Callaghan Innovation, and Unison Networks Limited (R&D Fellowship Grant UNIS1203) without which this work would not have been possible. My special thanks also goes to my industry advisor, Dr Thahirah Jalal, for her useful advice and all the arrangements for my research and to the Unison Asset Management team for hosting me two months during my research.

I am indebted to Professor Pat Bodger for his time and support during my first year at UC. His kind words in our first meeting back then 2014 are still in my mind. In one sentence: I wish I could meet such a legend many years ago.

My appreciation to all of my friends, fellow researchers, and staff at the Department of Electrical and Computer Engineering who made my time at the beautiful city of Christchurch very pleasant.

Last but not least, I would like to thank my wife, Fatemeh Ashari Ghomi, for her unconditional and heartfelt support through the ups and downs of this journey and also my family in Iran whose smiles in our video chats are the most precious in my life.

Contents

List of Figures

List of Tables

Glossary

Abbreviations

H_2	Hydrogen
CH_4	Methane
C_2H_2	Acetylene
C_2H_4	Ethylene
C_2H_6	Ethane
CO	Carbon monoxide
CO_2	Carbon dioxide
O_2	Oxygen
N_2	Nitrogen
2-ADOPT	Two Step Algorithm for Fault Diagnosis of Power Transformers
ACO	ANT Colony Optimisation
ADASYN	Adaptive Synthetic Over-Sampling Technique
ARIMA	Autoregressive Integrated Moving Average
ANFIS	Adaptive Network-Based Fuzzy Inference System
AUC	Area Under Curve
CART	Classification and Regression Trees
CC	Correlation Coefficients
CD	Crowding Distance
CFNN	Cascade Forward Neural Network
CV	Cross Validation
DGA	Dissolved Gas Analysis
DP	Degree of Polymerisation
DRM	Dielectric Response Analysis
DST	Dempster-Shafer Theory
EA	Evolutionary Algorithms
ED	Electrical Discharge
EMO	Evolutionary Multi-Objective Optimisation
ESN	Echo State Network
FFNN	Feedforward Neural Network

FKNN	Fuzzy K-Nearest Neighbor
FN	False Negative
FP	False Positive
FPR	False Positive Rate
FRA	Frequency Response Analysis
GA	Genetic Algorithm
GMDH	Group Methods of Data Handling
IEC	International Electrotechnical Commission
IEEE	Institute of Electrical and Electronic Engineers
KNN	K-Nearest Neighbor
KRIDGE	Kernel Ridge Regression
LTCs	Load Tap Changers
MAPE	Mean Absolute Percentage Error
MCC	Matthews Correlation Coefficient
MLP	Multi-Layer Perceptron
MOPSO	Multi-Objective Particle Swarm Optimisation
MSE	Mean Square Error
NB	Naive Bayes
NLPCA	Non-Linear Principle Component Analysis
NNs	Number of Neighbors
NSGA-II	Non-Dominated Sorting Genetic Algorithm II
OHF	Over Heating Faults
ORF	Oblique Random Random Forests
PACF	Partial Autocorrelation Function
PCA	Principle Component Analysis
PD	Partial Discharge
PER	Persistence Model
PPV	Positive Predictive Value
PSO	Particle Swarm Optimisation
RBF	Radial Basis Function
RF	Random Forests
RMSE	Root Mean Square Error
RT	Regression Trees
ROC	Receiver Operating Characteristics

RVFL	Random Vector Functional Link
SES	Simple Exponential Smoothing
SPEA-II	Strength Pareto Evolutionary Algorithm II
SVC	Support Vector Classification
SVM	Support Vector Machines
SVR	Support Vector Regression
TN	True Negative
TP	True Positive
TPR	True Positive Rate

Nomenclature

exp	the exponential function
$tanh$	the hyperbolic tangent function
$\|\mathbf{x}\|$	the L2 norm of $\mathbf{x}$
$\langle \mathbf{x}, \mathbf{y} \rangle$	inner product
P(A|B)	conditional probability
$\boldsymbol{A}^T$	the transpose of $\boldsymbol{A}$
$\boldsymbol{A}^{\textit{-1}}$	the inverse of $\boldsymbol{A}$
$A_{i,j}$	the (i, j) element of $\mathbf{A}$ $\boldsymbol{A}$
$\mathbb{R}$	the set of integer number
$\hat{y}$	the output vector of a model
$\boldsymbol{x}^*$	a Pareto optimal solution
$E\{\cdot\}$	expectation of a random variable
$\mathbf{x}$, $\mathbf{y}$, $\mathbf{w}$	vectors
x_i, y_i, w_i	the ith element of a vector
$\mathbf{x_1}, \cdots, \mathbf{x_m}$	a sequence of m vectors
$\frac{\partial L(\omega)}{\partial \omega}$	the partial derivative of a function

Chapter 1

Introduction

1.1 General overview

Power transformers are one of the most significant and expensive pieces of equipment in electrical networks. Monitoring the condition of these assets in order to ensure reliable operation is of great interest to electric utilities and power companies. Thus, transformer condition assessment plays an important role in a transformer asset management scheme. An optimum condition assessment can help power companies to manage their transformer fleet economically. In addition, there is a large social and environmental impact, because an optimum condition assessment activity can enhance the remaining useful lifetime of transformers and consequently, can prevent widespread power outages and defer expenditure.

There are useful conventional standards and transformer fault diagnosis methods which help to interpret the actual faults of transformers. However, they sometimes suffer from the lack of interpretability and accuracy which leads to an incorrect or non-detectable fault diagnosis using these methods. For example, Bacha et al. (2012) reported a 23% (7/30) and 26.7% (8/30) non-detection rate for key gas and ratio methods, which are two widely used conventional techniques described in Chapter 2, respectively. Furthermore, in order to effectively apply these methods, electric utilities and power companies should also consider the size, type, and environmental conditions of their transformer fleet which is very challenging is some cases.

In the early 1990's, new expert systems were used to diagnose incipient faults of transformers (Lin et al., 1993; Tomsovic et al., 1993; Zhang et al., 1996). These studies focused on addressing drawbacks of conventional methods. Since then, different intelligent algorithms have been proposed to assess the condition of transformers and to overcome the disadvantages of conventional methods, such as uncertainty in fault interpretation. With the growth in Machine Learning, Statistical Learning and Artificial Intelligence fields, it is now possible to learn from the historical data of transformers and predict their faults and future status.

Over the last decade, different machine learning methods have been applied in power systems applications; especially transformer condition assessment. A comprehensive review of these studies are given in Section 5.2 and Section 6.2. In most of these studies a single intelligent expert has been applied to diagnose transformer faults. However, based on the *no free launch* theorem, selecting the best algorithm is not always a straightforward process. The performance depends on the available dataset and it can vary extensively for different electricity networks.

To develop a reliable and general intelligent transformer condition assessment model, various intelligent single algorithms can be considered in order to create an ensemble of the best algorithms. Thus, an intelligent data-driven approach can be adopted, which can be used "in house" by electric utilities and power companies, regardless of the transformer type, size and technical conditions. However, it should be noted that using these kind of models require a depth of knowledge on how the developed algorithm functions in order to tune it properly.

1.2 Thesis objective

The objective of this thesis is to develop an intelligent data-driven transformer condition assessment model. For this purpose, an ensemble technique was used to develop an accurate and data-driven intelligent condition monitoring model to be able to select the best group of statistical and machine learning algorithms, automatically.

In order to achieve this objective, various classification and time series forecasting algorithms were developed, and different evolutionary multi-objective optimisation

(EMO) algorithms were used to select the most accurate and diverse ensemble of algorithms.

1.3 Thesis contribution

The work in this thesis has led to the development of a state-of-the-art intelligent transformer condition assessment tool. For this purpose, two different algorithms have been developed;

- An intelligent, dissolved gas analysis (DGA) based, transformer fault diagnosis algorithm was developed using various statistical and machine learning algorithms. All these algorithms were trained using an available DGA dataset. Subsequently, an evolutionary multi-objective optimisation algorithm was used to select a group of the most accurate and diverse classifiers/algorithms to classify transformer faults on the new DGA samples.
- To forecast the value of dissolved gases in transformers, a forecasting model was developed. The DGA dataset, along with some of the transformer's operating characteristics for a period of 6 months, were used to forecast the dissolved gases one, two, three, and four days ahead.

1.4 Thesis outline

Chapter 2 provides an overview of transformer fault diagnosis, based on DGA. This overview includes the importance of dissolved gas analysis in incipient fault diagnosis. The possibility of on-line DGA monitoring, as an advantage to diagnose or indicate abnormal operation of transformers, is also investigated. In addition, a list of different transformer failures which could be diagnosed using DGA are given. Furthermore, the drawbacks of conventional DGA based fault diagnosis methods are explained.

Chapter 3 describes the basic theory behind the statistical and machine learning algorithms used in this thesis. Two types of algorithms have been used in this research; classification and time series forecasting. In general, most of the classification algorithms used for classifying faults of transformers were tailored to be applied in dissolved gas forecasting.

Chapter 4 presents an example using support vector machines for fault classification of transformer load tap changers. The preliminary results show a promising prospect of using statistical and machine learning algorithm on condition assessment of power transformers. The shortcomings of single learning algorithm, compared with an ensemble learning system, are also discussed.

Chapter 5 details developed evolutionary multi-objective fault diagnosis of transformers. The algorithm is presented step by step and the obtained results presented in detail. A comprehensive performance comparison between the proposed algorithm and other conventional methods is also given in this chapter.

Chapter 6 presents the details of the developed multi-objective ensemble transformers' dissolved gas forecasting model. The detail of selecting best forecasting algorithms is discussed and the results of the forecasting model to predict the future value of dissolved gases are presented. In addition, the results of the proposed model are compared with other traditional time-series forecasting techniques.

Chapter 7 concludes the thesis and discusses some possible directions for future studies.

Chapter 2

Dissolved Gas Analysis of Power Transformers

2.1 Overview

This chapter begins by introducing different transformer condition monitoring and condition assessment techniques. This is followed by an overview of the various failure modes in power transformers and the importance of dissolved gas analysis technique in diagnosing these failure modes. Various conventional methods for interpreting dissolved gas results are then presented. A discussion on the drawbacks of these methods to clarify the motivation behind this research is also presented.

2.2 Introduction

Today, power companies can deliver higher quality services to their clients by performing intelligent asset management activities and reducing operating costs. One of the most critical asset classes to deliver electric power is power and distribution transformers where the risk of failure increases with ageing (Zhang and Gockenbach, 2008). A transformer failure usually results in a widespread outage in the network. Replacing a power transformer is expensive. A unit can cost up to 1 million dollars and long lead times are typical (Wang et al., 2002). It is therefore imperative for any electricity company to manage such assets effectively.

Electricity companies require new approaches, such as an intelligent fault diagnosing system, to reduce the operating costs and the failure rate of their assets (Abu-Elanien and Salama, 2010).

Reducing operating costs, enhancing the reliability, and improving the quality of services to clients are the major concerns for electric utilities. There is a high risk to leave assets, such as distribution and power transformers, in service without sufficient monitoring, as the probability of losing equipment with the ageing of these assets increases. By changing approaches to achieve new techniques of condition monitoring, condition assessment, and end-of-life estimation, electricity utilities are working on reducing their operating costs and improving the reliability of their assets.

Many transformer asset management activities have been developed during recent years and different techniques have been introduced to deal with this issue. The three main steps of a general asset management activity are failure modes and mechanisms analysis, condition monitoring and condition assessment, and scheduling appropriate maintenance plans. These techniques can be used for different equipment such as power transformers, circuit breakers, cables, etc. In Figure 2.1, common condition monitoring and their corresponding condition assessment techniques are shown for a power transformer (Abu-Elanien and Salama, 2010; Zhang and Gockenbach, 2008). This research focuses on transformer asset management using dissolved gas analysis technique. However, brief explanations of other condition assessment techniques are given below:

Thermal analysis: Since, a change in the thermal behavior of a transformer is a common phenomenon during abnormal operation, thermal analysis can provide useful insights about the condition of transformer. Overloading is one of the most important abnormal conditions in transformers that can be detected by thermal analysis (IEEE, 2012).

Vibration analysis: Vibration analysis is one of the relatively new methods for transformer condition assessment. Vibration analysis is usually done on three main parts of transformers, such as core, winding, and on-load tap changers (Rivas et al., 2009; Shengchang et al., 2001).

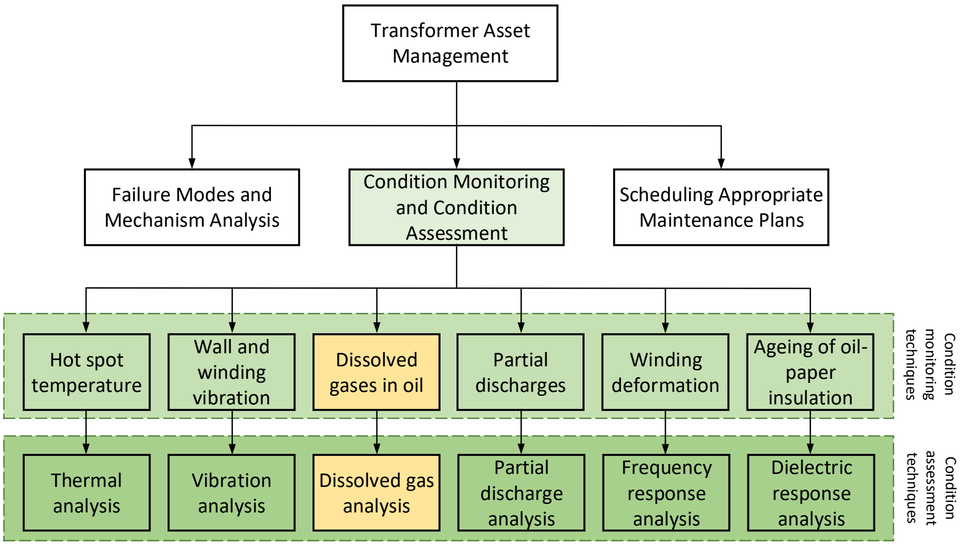

Transformer Asset Management
Failure Modes and Mechanism Analysis
Condition Monitoring and Condition Assessment
Scheduling Appropriate Maintenance Plans
Hot spot temperature
Wall and winding vibration
Dissolved gases in oil
Partial discharges
Winding deformation
Ageing of oil-paper insulation
Condition monitoring techniques
Thermal analysis
Vibration analysis
Dissolved gas analysis
Partial discharge analysis
Frequency response analysis
Dielectric response analysis
Condition assessment techniques

Figure 2.1: Schematic diagram of transformer asset management (Abu-Elanien and Salama, 2010).

Partial discharge: Partial discharge is the result of the exceeded electric field of the dielectric breakdown strength in the insulation medium of transformers. Consistent partial discharge leads to major failures in the dielectric properties of the transformer's insulation (Judd et al., 2002; Strachan et al., 2005). There are different very well-known techniques for detecting and measuring partial discharge such as using ultra high frequency sensors (Judd et al., 2002, 2005), acoustic sensors (Lundgaard, 1992; Najafi et al., 2013), and optical fiber sensors (Zargari and Blackburn, 1998).

Frequency response analysis: Mechanical stresses in transformers are due to fault currents which leads to winding movement and deformations. Different types of failures, such as mechanical deformation, short-circuited turn-to-turn, short-circuit-to-ground, ungrounded core, open-circuited, high contact resistance, bulk movement, loose clamping structure etc., can be detected by measuring electrical transfer functions of transformers over a wide frequency range using frequency response analysis method. This method works based on the comparison between frequency responses results of transformers before and after a failure (Islam, 2000; Wang et al., 2002, 2005; Yousof et al., 2015a,b).

Dielectric response analysis: Dielectric response analysis is one of the useful methods for measuring the content of moisture in transformer oil-paper insulation medium. Moisture can move into the oil-paper insulation system from ambient during the installation or repairing of transformers and can cause severe failures. Generally, determining the moisture content can be used for end-of-life assessment of transformers. Dielectric spectroscopy technique can be used in time and frequency domain for estimating the quality of insulation systems of transformers (Saha, 2003). Frequency domain spectroscopy is one of the most common techniques for quality assessment of transformer insulation system (Yousof et al., 2015a; Zaengl, 2003).

2.3 The importance of dissolved gas analysis

Dissolved gas analysis (DGA) is one the most useful and common techniques for condition monitoring and condition assessment of power transformers. A wide range of transformer failures can be detected and diagnosed using this technique. The three major stress categories that a transformer may encounter during its lifetime are thermal, electrical, and mechanical. Table 2.1 lists the common failures of transformers caused by these stresses and whether these failures can be detected by DGA. Nowadays, real time monitoring is of great importance for electric utilities and power companies helping them to manage their fleet more effectively. As illustrated in Table 2.1, almost all of the transformer failure modes can be monitored online using the DGA technique. Therefore, electric utilities and power companies use DGA as a convenient method for monitoring and incipient fault diagnosis of transformers. Once the faults are confirmed for further investigation, the most optimum maintenance process can then be planned. Further details of each failure mode are given in the following subsections.

2.3.1 Insulation degradation

There are many different factors which effect transformer insulation degradation. However, two major reasons are thermal and electrical stresses in the insulation medium of transformers (cellulose and mineral oil).

Table 2.1: Failure modes and corresponding condition monitoring techniques (CIG, 2003).

Failure mode	Condition monitoring	Online monitoring
Paper degradation	• DGA • Furan analysis • Power factor • Insulation resistance • Dielectric response analysis • Moisture analysis • Degree polymerisation	• Yes • Yes • No • No • No • Yes • No
Oil degradation	• DGA • Oil conductivity • Power factor • Insulation resistance • Dielectric response analysis • Moisture analysis • Degree polymerisation	• Yes • Yes • No • No • No • Yes • No
Partial discharge (PD)	• DGA • PD analysis	• Yes • Yes
Contact resistance	• DGA • Frequency response analysis • Winding resistance test	• Yes • No • No
Load tap changers failure	• DGA • Internal inspection	• Yes • No
Short circuit turn to turn	• DGA • Winding resistance test • Frequency response analysis • Transformer turns ratio • Excitation current	• Yes • No • No • No • No
Short circuit to ground	• DGA • Power factor • Frequency response analysis • Insulation resistance	• Yes • No • No • No

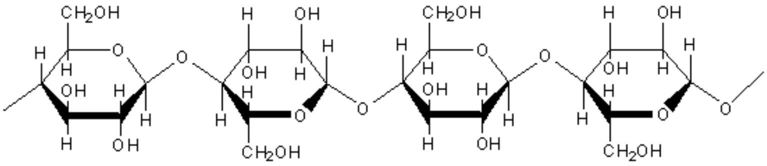


Figure 2.2: The cellulose molecule.

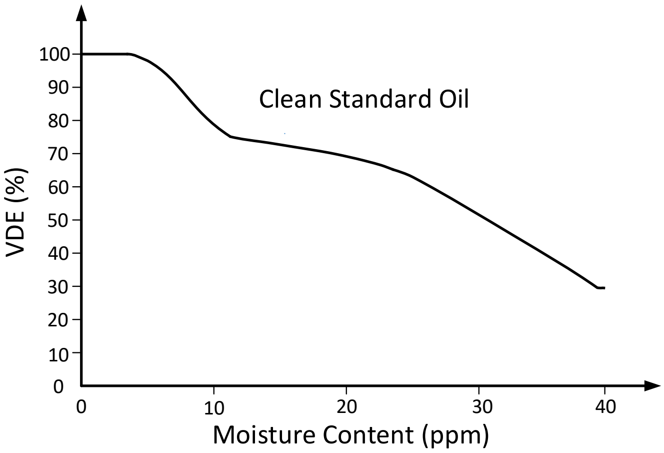


Figure 2.3: The effect of increasing moisture on voltage dielectric strength in transformer oil (Miners, 1982).

Cellulose is a macro molecule which consists of interconnected glucose rings as shown in Figure 2.2. The number of glucose rings per chain is called the degree of polymerisation (DP). In normal condition, the number of glucose rings in the chain can vary between 300 to over 1000. These long glucose chains may be broken under thermal and electrical stresses and other ageing processes. The condition of paper is deemed not acceptable for use in a power transformer if the number of glucose rings is less than 200 (DP<200) because the paper loses its mechanical properties and becomes brittle (Saha, 2003). Furthermore, cellulose oxidization produces water in the paper and, as a consequence, the voltage dielectric strength (VDE) of the paper is reduced significantly (Miners, 1982). Figure 2.3 shows the effect of moisture on the VDE.

Transformer oil consists of different hydrocarbon molecules. When electrical or thermal stress occurs inside a transformer, these hydrocarbon molecules are broken into carbon-hydrogen and carbon-carbon bonds. Different gases are formed, based on the amount of energy and temperature produced by the faults inside the transformer. Dissolved gas analysis (DGA) is a common method for interpretation of the produced gases in oil and different standards and techniques are available for this purpose, such as the IEEE C57.104 and IEC 60599 standards (IEC, 2007; IEEE, 2009). Transformer oil contains dissolved gases, even during normal operation when no faults occur in transformer. The level of these gases increases when a fault occurs in transformers. The increasing amount/rate of these gases depends on two important factors. The first is type and the second is the location of the fault in transformer (IEC, 2007; IEEE, 2009). The generated gases can be divided into three different groups:

- Hydrogen and hydrocarbons: H_2, CH_4, C_2H_2, C_2H_4, and C_2H_6.
- Carbon oxides: CO and CO_2.
- No-fault gases: O_2 and N_2.

Research has shown there is also correlation between faults and dissolved gas concentration (Emsley and Stevens, 1994; IEC, 2007; IEEE, 2009; Singh and Bandyopadhyay, 2010). Arcing faults produce relatively large quantities of H_2 and C_2H_2. Temperatures in excess of 500 °C are required for the generation of C_2H_2. Thermal decomposition of oil leads to increased concentration of C_2H_4, in combination with CH_4. The temperature required for generation of these gases is lower than 250 °C. An increase in concentration of H_2 and CH_4 is a sign for partial discharge in the transformer's oil. Generation of CO_2 and CO indicates thermal ageing in the cellulose insulation. The presence of H_2 and O_2 in the transformer oil, without other hydrocarbon gases, verifies the presence of water (IEC, 2007). Figure 2.4 illustrates the generation rate of the most relevant gases to each fault type for different temperatures.

In addition to the aforementioned problems, insulation degradation can cause several other faults, such as short circuits, extra heating, partial discharge or arcing.

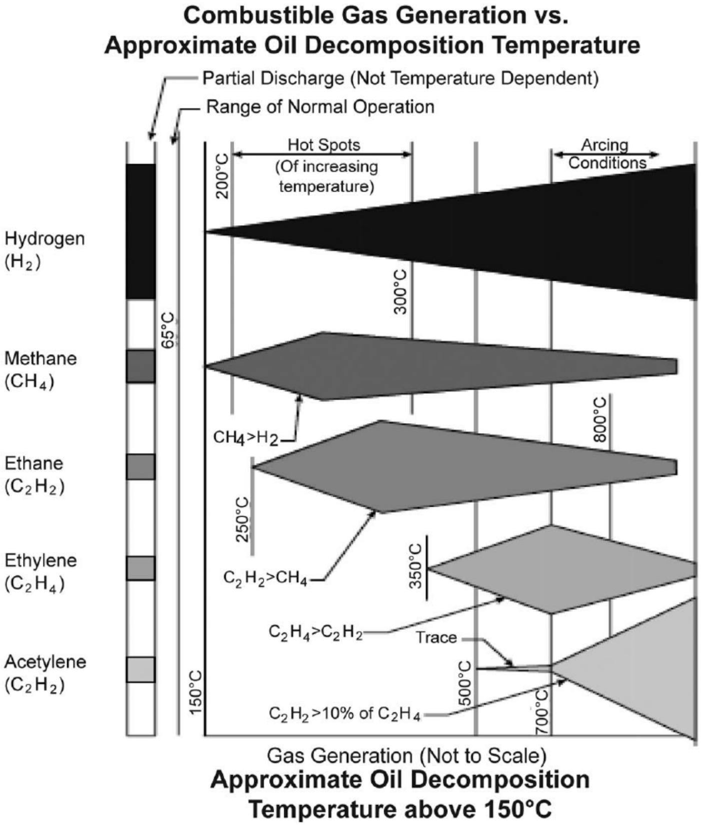


Figure 2.4: Schematic diagram of dissolved gases generation in different temperatures (Singh and Bandyopadhyay, 2010).

2.3.2 Partial discharge

Partial bridging of a transformer's insulation system can be a simple definition of partial discharge (PD). There are some phenomena which induce partial discharge, such as degradation of a transformer insulation during its life time and temporary over-voltage. Different defects in transformers, which may result in partial discharges, are as follows (Boggs, 1990; Morshuis, 2005):

- Floating component.
- Bad contact.

- Suspended particles.
- Rolling particles.
- Surface discharge.
- Protrusions.

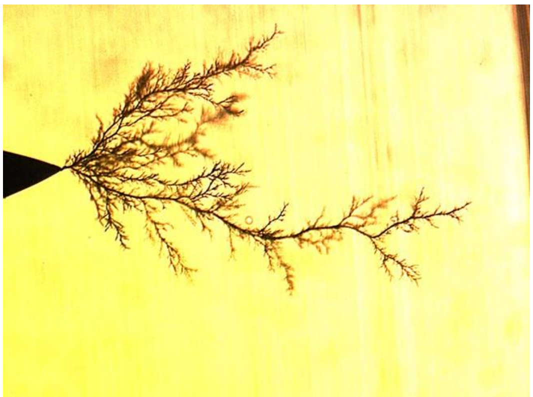


Figure 2.5: An example of electrical treeing in power transformer insulation (Treeing).

PD can deteriorate the electrical properties of the insulation, since PD pulses cause formation of a carbonized channel in cellulose insulation, which long term may lead to complete breakdown inside the dielectric. Figure 2.5 shows an electrical breakdown of polymeric insulation of power transformers. This condition can also affect the quality of oil insulation by producing chemical byproducts such as, gases, acids, and water, which drastically reduce oil withstand strength (Ghaffarian Niasar, 2015; Liao et al., 2011a).

As mentioned in Section 2.2, there are different PD detection methods. However, DGA as a cheap and straightforward technique can be also used to diagnose PD in power transformers. In DGA based fault diagnosis techniques, which are introduced in Section 2.4, hydrogen plays an important role as the key gas for diagnosing PD. In Chapter 5, an intelligent method is proposed and implemented to select the most relevant gases for diagnosing PD in power transformers.

2.3.3 Load tap changer failure

Due to the mechanical mechanisms of load tap changers (LTCs), the failure rate of LTCs is higher than other transformer parts, such as windings, bushing, and the core (Zhang and Gockenbach, 2008). A common problem of LTCs is contact cocking, which may cause increasing contact resistance and overheating (Duval, 2008). Although normal operation of LTCs produces dissolved gases due to arcing during normal operation of LTCs, the levels of these gases are usually higher than faulty transformers. Therefore, DGA can be used as an important measure for LTCs fault diagnosis. The details of an intelligent LTCs fault diagnosis model, which was developed during the course of this research, are presented in Chapter 4.

2.3.4 Other failure modes

The failure modes of transformers are not limited to the above-mentioned categories. There are also other failures, which have a low probability of occurrence, but they can cause severe damage to a transformer. Some of these failure modes are loss of sealing, blocking of pressure relief devices, and loss of core-clamping, which can result in an insulation problem, explosion due to accumulated combustible gases, or extra heat. In addition, to avoid extra heat within a transformer, fans, pumps, and radiators should work without any problem to transfer heat properly.

2.4 Transformer incipient fault diagnosis using DGA

Thermal and electrical stresses are two main reasons that result in the degradation of a transformer insulation and lead to the release of dissolved gases inside transformers. The type of fault in the transformer can be determined by analysing these gases. In order to classify the transformer incipient faults, several standards and methods have been developed, such as IEEE (2009) and IEC (2007) standards. Several intelligent algorithms have been also introduced to improve the reliability of diagnosing faults using conventional methods. For example, fuzzy logic and neuro-fuzzy systems (Duraisamy et al., 2007; Hooshmand et al., 2012; Huang et al., 1997; Tomsovic et al., 1993), artificial neural network (Huang, 2003; Miranda and Castro, 2005; Sarma and Kalyani, 2004), and statistical learning, such as Support Vector Machine (Ganyun et al., 2005; Mirowski and LeCun, 2012; wei Fei and bin Zhang, 2009; wei Fei et al., 2009) are the common machine learning methods, which have been applied to diagnose faults of in-service transformers. Although, these algorithms are very powerful, they have some drawbacks too. For example, in the fuzzy logic method, sometimes it is not easy to define the rules or using neural networks requires a comprehensive and reliable dataset to train the network.

2.4.1 DGA based fault diagnosis methods

In general, transformer faults are classified into four major classes as follows (IEC, 2007; IEEE, 2009):

1. Electrical arcing.
2. Electrical corona.
3. Overheating of cellulose.
4. Overheating of oil.

Table 2.2 shows the corresponding causes of these four major fault classes. As is clear from Table 2.2, some of these fault classes have more than one cause in transformers. Three major conventional transformer fault diagnosis techniques, based on DGA, are briefly explained in the following subsections.

2.4.1.1 Ratio methods

There are different ratio based methods which use a group of defined dissolved gas ratios. The most important ratios used in these methods are listed in Table 2.3.

Table 2.2: Power transformers fault classes and their corresponding causes.

Causes	Fault classes			
	Arcing	Corona	Overheating of paper	Overheating of oil
Short circuit turn to turn	✓		✓	
Open circuit	✓		✓	
Overloading			✓	✓
Moisture	✓	✓		
Floating particles	✓	✓		
Cooling system malfunction				✓
Load tap changers operation	✓			
Winding displacement		✓	✓	

Table 2.3: Dissolved gas ratios used in DGA ratio based methods.

Ratio	CH_4/H_2	C_2H_2/C_2H_4	C_2H_2/CH_4	C_2H_6/C_2H_2	C_2H_4/C_2H_6
Abbreviation	R_1	R_2	R_3	R_4	R_5

To evaluate the performance of the conventional ratio methods, three dissolved gas samples are considered and the interpretation of each method is presented on these samples in the following subsections. Table 2.4 shows the value of dissolved gases for these samples and their corresponding actual faults, which are partial discharge (PD), no fault (NF), and energy discharge (ED). The proposed method in Chapter 5 was tested on this dataset to show the capability of the developed intelligent transformer fault classification algorithm in this research.

Table 2.4: The diagnostic results of the conventional ratio methods on the three dissolved gas samples.

No.	Actual fault	Dissolved gases [ppm]					Diagnostic methods				
		H_2	CH_4	C_2H_4	C_2H_6	C_2H_2	IEC	Rogers	Doernenburg	Duval	Key gas
1	PD	1076	95	4	71	231	Not diagnosed	Not diagnosed	Not diagnosed	ED	PD
2	NF	2501	1428	4963	4622	6998	PD	Not diagnosed	PD	ED	Not diagnosed
3	ED	1565	93	34	47	0	PD	PD	Not diagnosed	TF	PD

2.4.1.2 Doernenburg's ratio method

In this method four ratios are used to classify three fault classes as listed in Table 2.5. To apply Doernenburg ratio method, three steps can be considered as follows:

Table 2.5: Fault diagnosis using Doernenburg's ratio method.

Fault class	R_1	R_2	R_3	R_4
Thermal	>1.0	<0.75	<0.3	>0.4
Low energy partial discharge (corona)	<0.1	Non-significant	<0.3	>0.4
High energy partial discharge (arcing)	>0.1 & <1.0	>0.75	>0.3	<0.4

Table 2.6: Dissolved gases concentration limit for Doernenburg's ratio validation check.

Dissolved gas	H_2	CH_4	CO	C_2H_2	C_2H_4	C_2H_6
Concentration limit (ppm)	100	120	350	35	50	65

- The first step is called validity check. For this purpose, the level of at least one gas used in the ratios in Table 2.5 should be twice the limits listed in Table 2.6 and one of the other three dissolved gases should reach these limits.
- If the Doernenburg ratio is valid for the transformer, then the four ratios (R_1, R_2, R_3, and R_4) can be computed.
- The calculated ratios should be checked whether they fall into the given ranges in Table 2.5.

One of the main drawbacks of Doernenburg's ratio technique is its high rate of non-diagnosed cases, as this method can only be applied when a validation test is passed (Bacha et al., 2012). As an example, Case 3 in Table 2.4 does not pass the validation check and Doernenburg ratio method cannot be used for this DGA sample. On the other hand, Case 1 falls into the highlighted area (A) in Figure 2.6, which is actually an uncertainty (blank) zone, and Doernenburg's ratio method is not able to diagnose the corresponding fault. Lastly, Case 2 is incorrectly classified as high energy PD using this method.

2.4.1.3 Rogers ratio method

Rogers ratio method is one of the most commonly used ratio methods. This method is mainly recognised for better diagnosing of thermal fault class compared to Doernenburg ratio method (Doernenburg and Strittmatter, 1974). Four gas

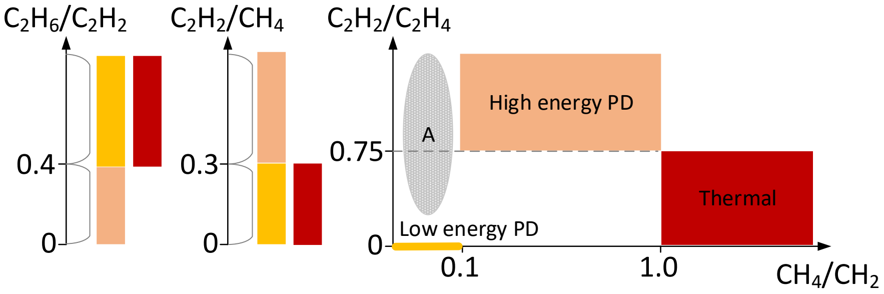


Figure 2.6: Schematic diagram of Doernenburg ratio method for transformers fault classification.

Table 2.7: Rogers ratio codes.

Ratio	Range	Code
R_1	≤0.1	5
	>0.1 & <1.0	0
	≥1.0 & <3.0	1
	≥3.0	2
R_2	<0.5	0
	≥0.5 & <3.0	1
	≥3.0	2
R_4	<1.0	0
	≥1.0	1
	>3.0	2
R_5	<1.0	0
	≥1.0 & <3.0	1
	≥3.0	2

ratios are used in this method as listed in Table 2.7. To apply this method, first a code is defined corresponding to each gas ratio level as shown in Table 2.7, then, Table 2.8 can be used to make a final decision on the transformer's fault.

The Rogers ratio method was so popular such that IEC 60599 standard (IEC, 2007) was proposed, based on this technique. However, this method is unable to diagnosed faults correctly in some cases, which increases the uncertainty rate (Bacha et al., 2012). In addition, the calculated ratios can be outside the defined ranges in Table 2.7, which results in non-diagnosable cases and consequently higher

Table 2.8: Fault diagnosis using Rogers ratio method.

Fault class	R_1	R_2	R_4	R_5
No fault	0	0	0	0
Partial discharge	5	0	0	0
Thermal fault ($T < 150\,^{\circ}$C)	1-2	0	0	0
Thermal fault ($150\,^{\circ}$C $< T < 200\,^{\circ}$C)	1-2	0	1	0
Thermal fault ($200\,^{\circ}$C $< T < 300\,^{\circ}$C)	0	0	1	0
General conductor overheating	0	0	0	1
Winding circulating current	1	0	0	1
Core and tank circulating currents, overheated joints	1	0	0	2
Flashover without power follow through	0	1	0	0
Arc with power follow through	0	1-2	0	1-2
Continuous sparking to floating potential	0	2	0	2
Partial discharge with tracking (note CO)	5	1-2	0	0

rate of uncertainty. As given in Table 2.4, it is not possible to interpret DGA for two DGA samples (Case 1 and Case 2) using Rogers ratio method. In addition, case 3 is incorrectly classified as PD, while the actual fault is ED.

2.4.1.4 IEC ratio method

The IEC ratio is derived from the Rogers ratio method. The main difference between these two methods is the number of ratios used in these methods. In IEC method only three gas ratios (R_1, R_2, and R_5) are used to diagnose six fault classes. The IEC ratio codes and the interpretation of the IEC ratio codes are summarized in Table 2.9 and Table 2.10, respectively (IEC, 2007). In general, the accuracy of the IEC method is higher than the Rogers ratio and Doernenburg ratio methods. For example, Muhamad et al. (2007) reported a 66% accuracy for the IEC method compared to 45% and 41% for the Rogers and Doernenburg ratio methods, respectively, for a dataset consists of 92 dissolved gas samples. In other research, Ghoneim et al. (2016) reported a 49% accuracy for the IEC method compared to 45% and 41% for the Rogers and Doernenburg ratio methods, respectively. The DGA dataset used in their study consists of 418 samples.

To show the drawbacks of the IEC method, as given in Table 2.4, this method is not able to interpret the DGA for Case 1 and the other two DGA samples

Table 2.9: IEC ratio codes.

Ratio	Range	Code
R_1	<0.1	1
	≥0.1 & <1.0	0
	≥1.0 & <3.0	2
	≥3.0	2
R_2	<0.1	0
	≥0.1 & <1.0	1
	≥1.0 & <3.0	1
	≥3.0	2
R_5	<0.1	0
	≥0.1 & <1.0	0
	≥1.0 & <3.0	1
	≥3.0	2

Table 2.10: Interpretation of IEC ratio codes.

Fault class	R_1	R_2	R_5
No fault	0	0	0
Low energy partial discharge	1	Non-significant	0
High energy partial discharge	1	1	0
Low energy discharge	0	1-2	1-2
High energy discharge	0	1	2
Thermal ($T < 150\,^\circ$C)	0	1	2
Thermal ($150\,^\circ\text{C} < T < 300\,^\circ$C)	2	0	0
Thermal ($300\,^\circ\text{C} < T < 700\,^\circ$C)	2	0	1
Thermal ($T > 700\,^\circ$C)	2	0	2

are incorrectly diagnosed as PD. Figure 2.7 illustrates the interpretation of the IEC ratio method for classifying three major fault classes, partial discharges (PD), low/high energy discharges (DL & DH), and low, medium, and high thermal faults (TL, TM, and TH) (IEC, 2007). It is clear that there are some blank zones in Figure 2.7 and it is not possible to diagnose the correct fault of transformer if a DGA sample were to fall into these blank zones such as Case 1 in the above example.

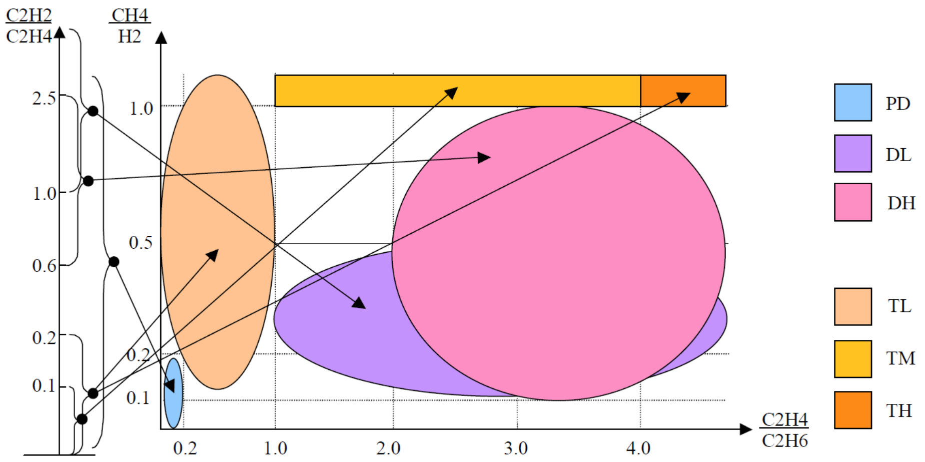


Figure 2.7: Schematic diagram of IEC ratio method for transformers fault classification (IEC, 2007; Wang, 2000).

2.4.2 Key gas and total dissolved combustible gas methods

In this method the main gases relevant to each fault type are used to diagnose the fault of transformer. As shown in Figure 2.4, the quantity of the generated dissolved gases in the transformer's oil is different at varying temperatures. This method uses the percentage of the key gases in the transformers to diagnose faults of transformer. Table 2.11 summarizes the four major fault classes and their corresponding key gas (Gray, 2009; Kelly, 1980).

The performance of the key gas technique is comparable with other conventional methods. This method diagnoses case 1 in Table 2.4 as PD correctly, while case 3 is incorrectly classified as PD and case 2 is not diagnosable using the key gas method.

Table 2.11: Fault diagnosis using key gas method.

Fault class	Key gas	Gas proportion
Arcing in oil	C_2H_2	H_2 (60%), C_2H_2 (30%), CH_4 (5%), C_2H_4 (3%), C_2H_6 (2%)
Corona in oil	H_2	H_2 (85%), CH_4 (13%), C_2H_4 (1%), C_2H_6 (1%)
Thermal in oil	C_2H_4	C_2H_4 (63%), C_2H_6 (19%), CH_4 (16%), H_2 (2%)
Thermal in cellulose	CO	CO (92%)

Table 2.12: Dissolved gas concentration limits (ppm) used TDCG method.

Dissolved gas	H_2	CH_4	C_2H_2	C_2H_4	C_2H_6	CO	CO_2	TDCG
Condition 1	100	120	1	50	65	350	2500	720
Condition 2	101-700	121-400	2-9	51-100	66-100	351-570	2500-4000	721-1920
Condition 3	701-1800	401-1000	10-35	101-200	101-150	571-1400	4001-10000	1921-4630
Condition 4	>1800	>1000	>35	>200	>150	>1400	>10000	>4630

In IEEE standard C57.104 (IEEE, 2009), a different key gas approach called the total dissolved combustible gas (TDCG) method was introduced. This method considers the summation of the dissolved gases and the value of the individual gases simultaneously to evaluate the condition of transformer. As stated in IEEE (2009), it can be difficult to classify between normal and faulty condition using concentration of dissolved gases. The four steps TDCG method is especially useful when there are no historical DGA records for the transformer. In this method, four different conditions of transformer based on the level of individual dissolved gases and TDCG are defined as shown in Table 2.12. The CO_2 value in Table 2.12 is not considered in TDCG value. The interpretation of transformer condition evaluation using this method is given as follows:

- *Condition 1*: If TDCG is below the 720 ppm, the transformer is in a healthy condition. However, immediate investigation would be required if the value of any individual dissolved gas exceeds the defined levels in Table 2.12.
- *Condition 2*: TDCG between 721 ppm and 1920 ppm indicates greater than normal dissolved gas concentrations. If any individual dissolved gas exceeds the specified levels, an immediate investigation of the transformer is necessary and a trend check is also required.
- *Condition 3*: TDCG within this range is a symptom of a high level decomposition. A prompt investigation is required if any individual dissolved gas exceeds the thresholds in Table 2.12. The probability of existing fault(s) in the transformer is high and a trend check action should be done immediately.

- *Condition 4*: TDCG within this range is a symptom of an excessive decomposition. If the transformer remains in-service, it could leads to a complete failure of the transformer.

2.4.3 Duval Triangle method

One of the most reliable methods for diagnosing faults in transformers is the Duval Triangle, which was introduced by Michel Duval in 1974 (Duval, 1974). This is a visual interpretation technique for DGA and it is based on using three different hydrocarbon gases (CH_4, C_2H_2, C_2H_4). Figure 2.8 shows the Triangle used for diagnosing faults and its distinct zones corresponding to each fault class. Each side of the Triangle represents the relative proportions of the three gases ($\frac{CH_4}{CH_4+C_2H_2+C_2H_4} \times 100$, $\frac{C_2H_2}{CH_4+C_2H_2+C_2H_4} \times 100$, and $\frac{C_2H_4}{CH_4+C_2H_2+C_2H_4} \times 100$).

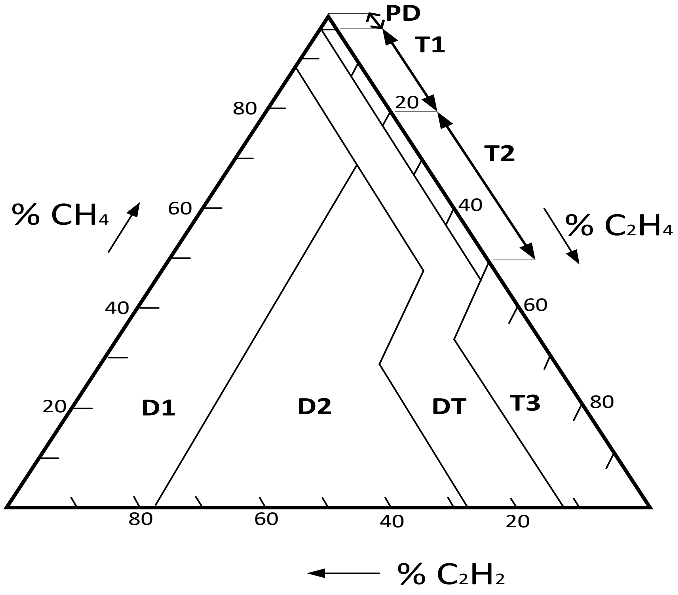


Figure 2.8: Duval Triangle.

Three major fault types can be diagnosed using this method, i.e., partial discharge, high and low energy arcing, and overheating (thermal faults) of three different temperature ranges. An additional zone is also considered in the Triangle which

Table 2.13: Dissolved gases concentration and generation limit for Duval Triangle validation check.

Dissolved gas	H_2	CH_4	C_2H_2	C_2H_4	C_2H_6	CO	CO_2
Concentration limit (ppm)	100	75	3	75	75	700	7000
Generation limit (ppm per month)	50	38	3	38	38	350	3500

is called the intermediate zone and symbolized by DT for a mixture of electrical and thermal faults in transformers (Duval, 2002). The main drawback in applying the Duval Triangle method is the validation step, as it is crucial to confirm that at least one of the gases has reached its minimum and increasing rate limits, which are listed in Table 2.13 (Duval, 2008; Muhamad et al., 2007).

The reported accuracies of the Duval triangle method in the literature are generally higher than other conventional ratio methods. The accuracy of the Duval triangle method reported in Muhamad et al. (2007) and Ghoneim et al. (2016) are 89% and 78.9%, respectively, which are higher than the accuracies of other ratio methods mentioned in the previous sections for these two studies.

2.4.4 Intelligent DGA based fault diagnosis methods

Over the past decade, various intelligent power transformer condition assessment methods have been developed using artificial intelligence, statistical, and machine learning models. In these systems, data-driven approaches are used to extract knowledge from the raw historical data in order to overcome the drawbacks of the conventional methods. For example, Shintemirov et al. (2009) compared three different classification algorithms and the overall accuracy of their method was 92.11%. A relatively high diagnostic accuracy is reported for another two intelligent transformer fault diagnosis methods where Ghoneim and Taha (2016) designed an algorithm with 92.91% accuracy and Bacha et al. (2012) proposed an intelligent fault classification method with 90% accuracy on their dataset. A comprehensive literature review on the intelligent models is given in Section 5.2 and Section 6.2 respectively. However, before introducing the proposed state-of-the-art transformer fault diagnosis and dissolved gas forecasting algorithms, a brief review on applied statistical and machine learning algorithms in this research will be presented in Chapter 3.

2.5 Discussion

In this chapter different traditional DGA interpretation techniques were introduced and the drawbacks of these methods were listed. The drawbacks and shortcomings of these methods for transformers fault diagnosis were investigated using a case study. Table 2.4 summarises the performance of different traditional fault diagnosis methods on the three dissolved gas samples. As is clear from Table 2.4, most of these methods are not able to interpret DGA or diagnose faults for these case studies, which is mainly because of the limited defined ranges of the dissolved gas ratios used in these methods. To overcome this problem, an intelligent transformer fault classification algorithm is proposed in this research. The fault classification algorithm is able to define soft and non-linear boundaries between the fault classes regardless of wherever the dissolved gas samples were to fall in the space. In other words, this algorithm can classify and interpret all the dissolved gas samples with high accuracy and without any validation check step, which there was in some traditional DGA methods. Another important feature of the developed fault classification algorithm is assigning probabilities to each diagnosed case, which provides further insights to transformer fault diagnosis practice. Sometimes, collecting DGA data is very expensive and sometimes a set of DGA sensors can cost more than one hundred thousand dollars. The dataset used for developing an intelligent data-driven algorithm in Chapter 5 is collected from different power transformers published in previous studies (Ganyun et al., 2005; Gao et al., 1998; Sarma and Kalyani, 2004; Vanegas et al., 1997; Zhang et al., 1996). This helps to develop a general fault diagnosing algorithm regardless of type and size of transformers. Therefore, electric utilities can use the developed algorithms in this study using their own dataset.

Chapter 3

An Overview of Statistical and Machine Learning Algorithms

3.1 Overview

An overview of some of the well-known statistical and machine learning algorithms used in this research is presented in this chapter. These algorithms are used in the following chapters to diagnose power transformer faults and to predict the value of dissolved gases inside transformers. The chapter begins by introducing machine learning frameworks, followed by a brief description of the algorithms used, and concludes with a discussion on the merits and disadvantages of these algorithms.

The algorithms described in this chapter, which are then used in an ensemble framework in the next chapters (Chapter 5 and Chapter 6), were chosen from different machine learning algorithms families such that there is at least one candidate from each category. In addition, there are various well-developed packages that can be used to implement these algorithms. Table 3.1 lists the algorithms used in this research and their corresponding categories.

3.2 Machine Learning Frameworks

These methods are categorised as intelligent algorithms which can learn from the dataset and perform classification, regression, clustering, and time series prediction

Table 3.1: List of the used algorithms in this research and their corresponding categories.

Category	Algorithm(s)
Statistics based methods	Decision Trees
Kernel based methods	Support Vector Machines
Probabilistic models	Naive Bayes
Distance based methods	K-Nearest Neighbours
Regularization based methods	Kernel Ridge Regression
Rule based methods	Adaptive Network-based Fuzzy Inference Systems
Artificial Neural Networks	Feedforward Neural Networks Radial Basis Function Networks Cascade Forward Neural Networks Echo State Networks Random Vector Functional Link Group Method of Data Handling

tasks properly. Generally, machine learning problems can be divided into three main groups, which are:

- Supervised learning: In these problems some previously solved examples are presented to the machine and the goal is to find a function (rule) that maps inputs to outputs. The machine (algorithm) can learn from the training dataset during the learning process and then predicts on a new example. This is very similar to the learning process in the real world. For example, students are given a set of examples and the corresponding answers in class to practice, then they are asked about new questions in the exam, which are similar to what they have learned in the class but not exactly the same.

- Unsupervised learning: In these cases the machine is left to find the hidden pattern behind the data without any given examples.

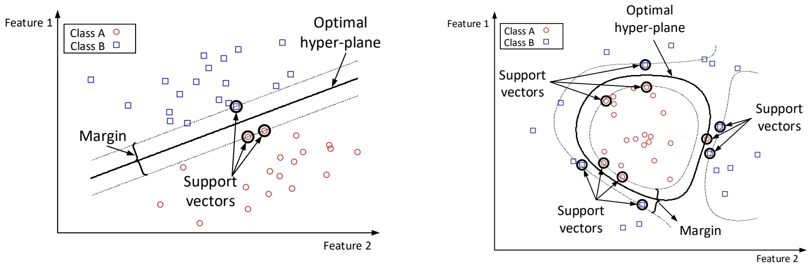
- Reinforcement learning: The machine can interact with an external dynamic environment, and accordingly, the system is either rewarded or punished from the environment in order to learn an appropriate task over time.

In this research, statistical and machine learning algorithms were used in a supervised learning framework. Therefore, the following algorithms are described using this assumption, i.e., a set of examples are first presented to the algorithm to learn, which is called the training phase, then the trained algorithm is tested on an unknown example.

3.3 Support Vector Machines

There are two different but similar Support Vector Machines (SVM) used for classification and regression problems (Friedman et al., 2001). Mathematically these two algorithms, Support Vector Classification (SVC) and Support Vector Regression (SVR), are identical except for some minor differences. The main difference between them is that the outputs of SVC are binary valued vectors of the predicted class indices, while the outputs of SVR are real values of the predicted function or time series data.

SVM is a learning algorithm based on the statistical learning theory which seeks optimum hyper-planes in order to separate a dataset into different classes or approximate a function. An example of linear and non-linear classification for a two class problem using SVM are shown in Figure 3.1.



(a) (b)

Figure 3.1: An example of: (a) linear and (b) non-linear support vector classifiers.

Suppose that we have a dataset of N inputs and N targets as:

$$Z = \{(\mathbf{x}_1, t_1), (\mathbf{x}_2, t_2), \ldots, (\mathbf{x}_N, t_N)\}, \tag{3.1}$$

$$\text{s.t.} \quad \mathbf{x}_n \in \mathrm{I\!R}^{\mathrm{m}}, \mathrm{t_n} \in \mathrm{I\!R}, \tag{3.2}$$

where $\boldsymbol{x}_i$ and $\boldsymbol{t}_i$ are inputs vectors and targets, respectively.

The aim of this algorithm is to use this dataset to find the function $f(\mathbf{x})$, which maps inputs to targets, as follows (Cortes and Vapnik, 1995):

$$f(\mathbf{x}) \simeq \sum_{n=1}^{N} (\mathbf{w}^{\mathrm{T}} \cdot \mathbf{x}_n + b), \tag{3.3}$$

where $\mathbf{x}_n$ and $\mathbf{w}^{\mathrm{T}}$ denote inputs and transpose of the weights vectors, respectively, and b is the bias term.

Figure 3.2 shows the core concept in SVM, which is to find $\mathbf{w}$. This keeps the error of prediction such that it will be less than ε, which is called the margin. So, if a sample like a is outside the acceptable error region, ε, it is penalized as follows (Cortes and Vapnik, 1995):

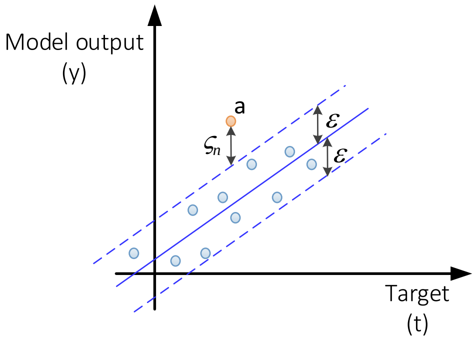


Figure 3.2: A graphical representation of penalizing a sample (a) that falls outside of an acceptable margin (ε).

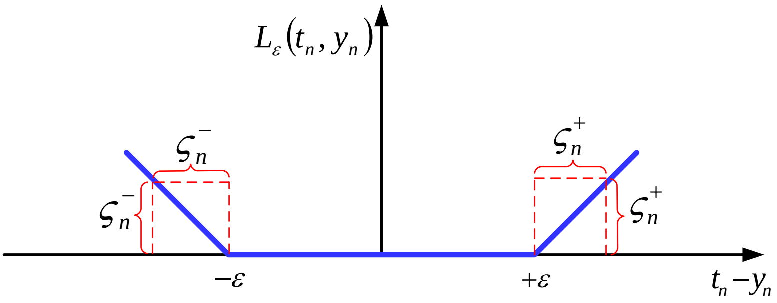


Figure 3.3: A graphical representation of Vapnik loss function.

$$L_\varepsilon(t_n, y_n) = \begin{cases} 0, & \text{if } |t_n - y_n| \le \varepsilon. \\ |t_n - y_n| - \varepsilon, & \text{otherwise.} \end{cases} \tag{3.4}$$

where L_ε is called Vapnik loss function, and y_n are the outputs. The $|t_n - y_n| \le \varepsilon$ is considered as the acceptable region and L_ε can be graphically shown in Figure 3.3. The ζ_n are called positive slack variables.

According to Figure 3.3 and Equation 3.4, the loss function can be rewritten as follows (Cortes and Vapnik, 1995):

$$\begin{cases} \zeta_n^+ + \zeta_n^- = L_\varepsilon(t_n, y_n), \\ \zeta_n^+ \zeta_n^- = 0, \\ \zeta_n^+ \ge 0, \zeta_n^- \ge 0. \end{cases} \tag{3.5}$$

Therefore, in the SVM algorithm, an empirical risk should be minimized as (Cortes and Vapnik, 1995):

$$\min \quad \frac{1}{2}\|\omega\|^2 + C\sum_{n=1}^{N}(\zeta_n^+ + \zeta_n^-), \tag{3.6}$$

subject to

$$-\varepsilon - \zeta_n^- \leq t_n - y_n \leq \varepsilon - \zeta_n^+, \quad \forall n \tag{3.7}$$

$$\zeta_n^+ \geq 0, \quad \forall n \tag{3.8}$$

$$\zeta_n^- \geq 0, \quad \forall n, \tag{3.9}$$

where C controls the trade-off between maximizing the margin and minimizing the training error. Applying the Lagrangian principle to the defined problem in Equation 3.6 results in:

$$\begin{aligned} L_P(\mathbf{w}, b, \alpha) &= \frac{1}{2}\mathbf{w}^{\mathrm{T}}\mathbf{w} + C\sum_{n=1}^{N}(\zeta_n^+ + \zeta_n^-) \\ &- \sum_{n=1}^{N}\alpha_n^+(-t_n + y_n + \varepsilon + \zeta_n^+) \\ &- \sum_{n=1}^{N}\alpha_n^-(t_n - y_n + \varepsilon + \zeta_n^-) \\ &- \sum_{n=1}^{N}\mu_n^+\zeta_n^+ - \sum_{n=1}^{N}\mu_n^-\zeta_n^-, \end{aligned} \tag{3.10}$$

where α_i are the Lagrangian coefficients. The optimal conditions is achieved by (Cortes and Vapnik, 1995):

$$\begin{cases} \dfrac{\partial L_P}{\partial \mathbf{w}} = \mathbf{w} - \displaystyle\sum_{n=1}^{N}(\alpha_n^+ - \alpha_n^-)\mathbf{x}_n = 0. \\ \dfrac{\partial L_P}{\partial b} = \displaystyle\sum_{n=1}^{N}(\alpha_n^+ - \alpha_n^-) = 0. \\ \dfrac{\partial L_P}{\partial \zeta_n^+} = \alpha_n^+ + \mu_n^+ = C. \\ \dfrac{\partial L_P}{\partial \zeta_n^-} = \alpha_n^- + \mu_n^- = C. \end{cases} \tag{3.11}$$

By substituting Equation 3.11 in Equation 3.10, the dual problem is:

$$\begin{aligned} \max \quad & - \frac{1}{2} \sum_{n,m=1}^{N} (\alpha_n^+ - \alpha_n^-)(\alpha_m^+ - \alpha_m^-) \ \mathbf{x}_n^{\mathrm{T}} \mathbf{x}_m \\ & + \sum_{n=1}^{N} (\alpha_n^+ - \alpha_n^-) t_n \ - \ \varepsilon \sum_{n=1}^{N} (\alpha_n^+ + \alpha_n^-), \end{aligned} \tag{3.12}$$

subject to

$$\begin{cases} \sum_{n=1}^{N} (\alpha_n^+ - \alpha_n^-) = 0, \\ 0 \leq \alpha_n^+ \leq C, \\ 0 \leq \alpha_n^- \leq C. \end{cases} \tag{3.13}$$

The set of support vectors are defined as follows:

$$S = \{n | 0 < \alpha_n^+ + \alpha_n^- < C \ \wedge \ \alpha_n^+ \alpha_n^- = 0\}. \tag{3.14}$$

From Equation 3.14 and Equation 3.11:

$$\mathbf{w} = \sum_{n=1}^{N} (\alpha_n^+ - \alpha_n^-) \mathbf{x}_n. \tag{3.15}$$

Subsequently, one can determine the bias term (b) in Equation 3.3 by:

$$t_n \ = \ \overbrace{\mathbf{w}^T \mathbf{x}_n + b}^{f(\boldsymbol{x})} \ + \ [sign(\alpha_n^+ - \alpha_n^-)] \varepsilon, \tag{3.16}$$

and,

$$b \ = \ \frac{1}{|S|} \sum_{n \in S} (t_n - \mathbf{w}^T \mathbf{x}_n - [sign(\alpha_n^+ - \alpha_n^-)]) \varepsilon.$$

Now, by replacing the inner product, $\langle \mathbf{x}_n, \mathbf{x}_m \rangle$, in Equation 3.12 with a kernel function $K(\mathbf{x}_n, \mathbf{x}_m)$ and following the same procedure described above, a non-linear predictor can be obtained and Equation 3.12 can be rewritten as follows:

$$
\begin{aligned}
\max \qquad & -\frac{1}{2}\sum_{n,m=1}^{N}(\alpha_n^+ - \alpha_n^-)(\alpha_m^+ - \alpha_m^-)\ K(\mathbf{x}_n, \mathbf{x}_m) \\
& + \sum_{n=1}^{N}(\alpha_n^+ - \alpha_n^-)t_n \ - \ \varepsilon\sum_{n=1}^{N}(\alpha_n^+ + \alpha_n^-),
\end{aligned}
\tag{3.17}
$$

Some of the most common kernel functions are listed in Equation 3.18 (Cortes and Vapnik, 1995; Schölkopf and Smola, 2002).

$$
\begin{cases}
\text{Linear kernel :} & K(\mathbf{x}_n, \mathbf{x}_m) = \mathbf{x}_n^{\mathrm{T}}\mathbf{x}_m, \\
\text{Polynomial kernel :} & K(\mathbf{x}_n, \mathbf{x}_m) = (\mathbf{x}_n^{\mathrm{T}}\mathbf{x}_m + 1)^P, \\
\text{Gaussian kernel :} & K(\mathbf{x}_n, \mathbf{x}_m) = \exp\left[\dfrac{-\|\mathbf{x}_n - \mathbf{x}_m\|^2}{2\sigma^2}\right], \\
\text{Sigmoid kernel :} & K(\mathbf{x}_n, \mathbf{x}_m) = \tanh[\gamma\mathbf{x}_n^{\mathrm{T}}\mathbf{x}_m + \beta].
\end{cases}
\tag{3.18}
$$

3.4 Group Method of Data Handling

The group Method of Data Handling (GMDH) was introduced by Ivakhnenko (1971). This method is also known as a polynomial neural network. In GMDH, the relationship between multiple inputs and outputs of the network can be modelled as:

$$
\widehat{Y}(\mathbf{x}) = a_0 + \sum_{i=1}^{m} a_i f_i(\mathbf{x}), \tag{3.19}
$$

where $\mathbf{x}$ is the input vector, Y is the output, a_i are coefficients, f_i are the elementary functions, and m is the number of base function components in the GMDH network.

In the GMDH algorithm, various subsets of Equation 3.19, which are called partial-models, are defined. Then, the coefficients of these partial-models are determined using least-squares techniques (Ivakhnenko, 1971). The core concept of GMDH is to find a model (network) with optimal complexity by gradually increasing the partial-models. This research uses one of the most well-known base functions in GMDH algorithms called the Kolmogorov-Gabor polynomial:

$$\widehat{Y}(\mathbf{x}) = a_0 + \sum_{i=1}^{n} a_i \mathbf{x}_i + \sum_{i=1}^{n} \sum_{j=i}^{n} a_{ij} \mathbf{x}_i \mathbf{x}_j + \sum_{i=1}^{n} \sum_{j=i}^{n} \sum_{k=j}^{n} a_{ijk} \mathbf{x}_i \mathbf{x}_j \mathbf{x}_k + \dots . \tag{3.20}$$

In order to show the complexity of the network, consider the following; if the number of inputs is considered equal to 10, the number of coefficients, and subsequently, the number of elementary functions, is equal to 1024. To overcome this problem, the aforementioned partial-models are created in an intelligent self-organizing process. Figure 3.4 shows an example of the process of creating partial-models in the GMDH algorithm. For the sake of simplicity, and without loss of generality, only four inputs ($\mathbf{x}_1 - \mathbf{x}_4$) are considered here. In the first hidden layer, a set of elementary functions are defined. The elementary functions are considered as a Kolmogorov-Gabor polynomial with the degree of two, as:

$$z_{ij} = c_1 + c_2 \mathbf{x}_i + c_3 \mathbf{x}_j + c_4 \mathbf{x}_i^2 + c_5 \mathbf{x}_j^2 + c_6 \mathbf{x}_i \mathbf{x}_j, \tag{3.21}$$

where $\mathbf{x}_i$ and $\mathbf{x}_j$ are two selected inputs and the c's are coefficients which are determined using the least squares technique in the training phase. Then, the defined z_{ij}'s in layer one is sorted and selected using the *external criterion*. One popular *external criterion* is called Criterion of Regularity, which is a minimization of least squares of the partial models using a separate part of a dataset which is not used for estimating of coefficients. This process continues until a stopping criterion is met.

3.5 Multi-Layer Perceptron, Radial Basis Function, and Cascade Forward Neural Network

A Multi-Layer Perceptron (MLP) is a fully connected feed-forward neural network (Rumelhart et al., 1985). The basic structure of a MLP is shown in Figure 3.5.

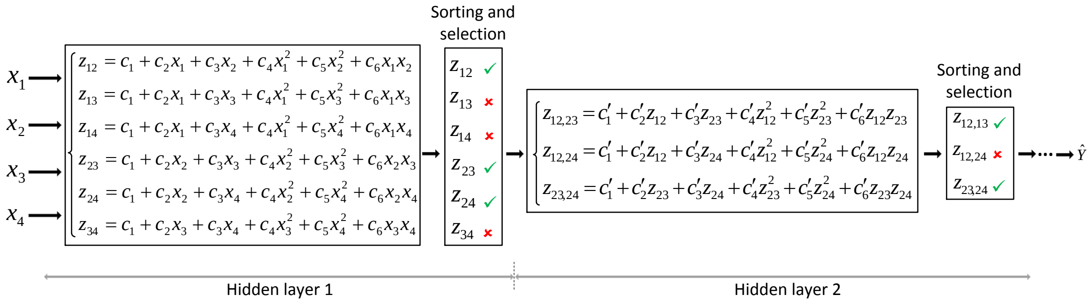

x_1
x_2
x_3
x_4
$z_{12} = c_1 + c_2x_1 + c_3x_2 + c_4x_1^2 + c_5x_2^2 + c_6x_1x_2$
$z_{13} = c_1 + c_2x_1 + c_3x_3 + c_4x_1^2 + c_5x_3^2 + c_6x_1x_3$
$z_{14} = c_1 + c_2x_1 + c_3x_4 + c_4x_1^2 + c_5x_4^2 + c_6x_1x_4$
$z_{23} = c_1 + c_2x_2 + c_3x_3 + c_4x_2^2 + c_5x_3^2 + c_6x_2x_3$
$z_{24} = c_1 + c_2x_2 + c_3x_4 + c_4x_2^2 + c_5x_4^2 + c_6x_2x_4$
$z_{34} = c_1 + c_2x_3 + c_3x_4 + c_4x_3^2 + c_5x_4^2 + c_6x_3x_4$
Sorting and selection
z_{12} ✓
z_{13} ✗
z_{14} ✗
z_{23} ✓
z_{24} ✓
z_{34} ✗
$z_{12,23} = c_1' + c_2'z_{12} + c_3'z_{23} + c_4'z_{12}^2 + c_5'z_{23}^2 + c_6'z_{12}z_{23}$
$z_{12,24} = c_1' + c_2'z_{12} + c_3'z_{24} + c_4'z_{12}^2 + c_5'z_{24}^2 + c_6'z_{12}z_{24}$
$z_{23,24} = c_1' + c_2'z_{23} + c_3'z_{24} + c_4'z_{23}^2 + c_5'z_{24}^2 + c_6'z_{23}z_{24}$
Sorting and selection
$z_{12,13}$ ✓
$z_{12,24}$ ✗
$z_{23,24}$ ✓
$\hat{Y}$
Hidden layer 1
Hidden layer 2

Figure 3.4: An example of GMDH network. In layer 1, first the polynomial with the degree of two are created and then some of these are sorted and selected. This process continues in the subsequent layers until a stopping criterion is met and the output $(\widehat{Y})$ is reported.

The network aims to map inputs to targets properly. In Figure 3.5, y_j denotes the outputs of the j^{th} neuron and s_j is the weighted sum of the inputs for the jth neuron. The nodes in hidden layers are called neurons and each node has a non-linear activation function $f(\cdot)$. The two most popular non-linear activation functions, called *log-sigmoid* and *tan-sigmoid*, are shown in Figure 3.6. The "weights" and biases of the network are first initialised and then optimized to improve the performance of the network. To evaluate the network performance, the mean square error

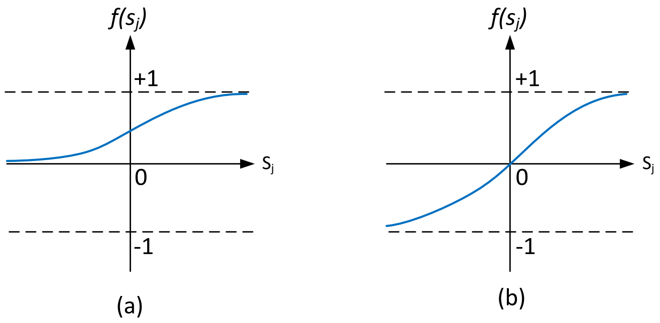


Figure 3.6: Two commonly used non-linear activation functions. (a) Log-sigmoid. (b) Tan-sigmoid

(Equation 3.22) is used as a cost function.

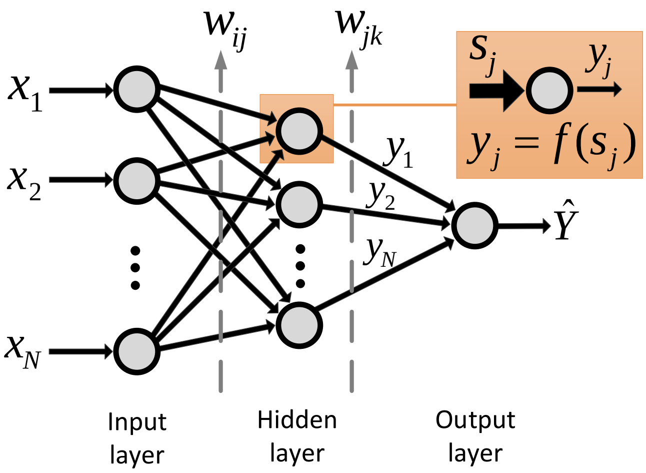


Figure 3.5: An example of MLP network.

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(\hat{Y} - Y)^2, \tag{3.22}$$

where Y and $\hat{Y}$ are the target and network output, respectively, and N is the size of the dataset or the number of samples in the dataset.

To tune the network parameters (weights and biases), the optimisation method can use the gradient of the network performance with respect to the weights. The *backpropagation* algorithm is one of the most popular techniques used in the training phase of the MLP (Neural Networks). This algorithm minimises the cost function in Equation 3.22 in each iteration by adopting the new parameters (weights and biases) to the network. The network parameters are chosen by the *backpropagation* algorithm in such way that the cost function has a maximum decrease (Leung and Haykin, 1991).

The Radial Basis Function (RBF) is also a neural network and its structure is very similar to MLP (Broomhead and Lowe, 1988). The main difference between a RBF and a MLP is in the type of the activation function. In a RBF network,

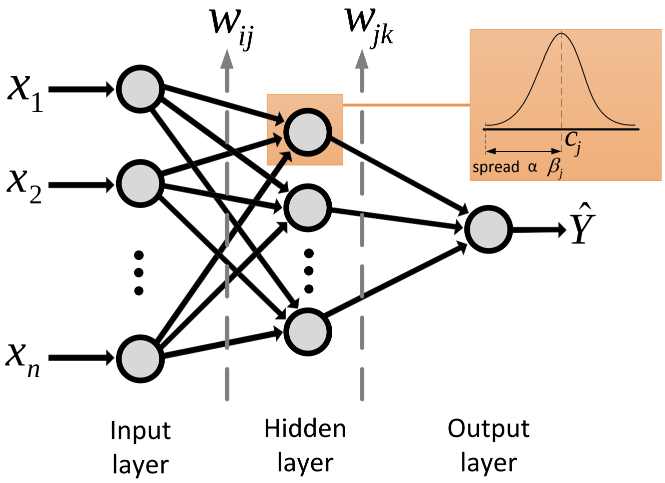


Figure 3.7: An example of RBF network.

a non-linear radial basis activation function is used. The most popular activation function in a RBF is Gaussian, which is defined as:

$$f(\boldsymbol{s}) = \exp(-\beta_j\|\mathbf{s}-\mathbf{c}_j\|^2), \tag{3.23}$$

where β_j is controlling the effectiveness of the jth neuron by adjusting the width of the bell curve, c_j is the centre of the jth neuron, and s_j is the weighted sum of the jth neuron's inputs as shown in Figure 3.7 (Broomhead and Lowe, 1988). The aim in a RBF network is to optimize β_j, c_j, and the network's weights, in order to minimize the error between the network's output and the actual value (target). The parameters of RBF network can be tuned using the *backpropagation* algorithm, similar to MLP.

The cascade forward neural network (CFNN) was first introduced by Fahlman and Lebiere (1989). The structure in these types of neural networks is similar to a Feedforward Neural Network (FFNN) or MLP with an additional direct connection from the previous layers to the output layer as shown in Figure 3.8. The network parameters for updating is similar to MLP and RBF networks, which use the *backpropagation* learning algorithm to find the optimum weights of the network.

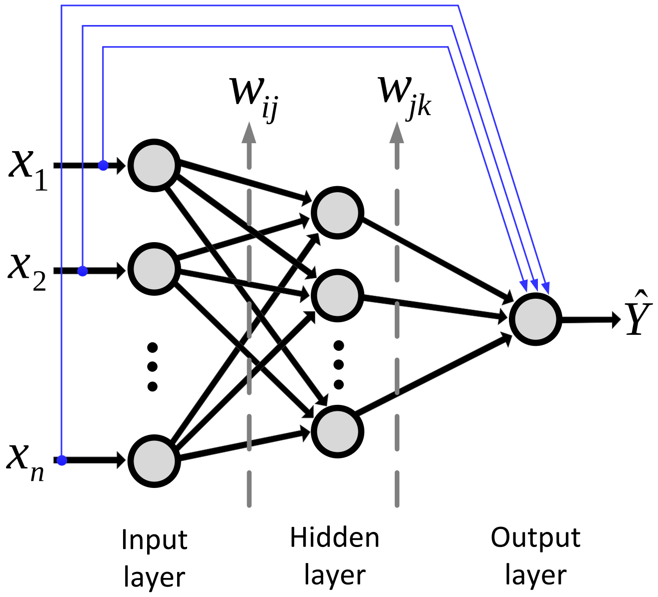


Figure 3.8: An example of CFNN network.

3.6 Classification and Regression Trees

The classification and regression trees (CART) algorithm (Breiman et al., 1984) is one the most powerful and well-known data mining methods. These trees find a model to predict the targets from the inputs, such that each time a new set of decisions is followed from the root node down to leaf node as shown in Figure 3.9 to partition the dataset. The decision trees are called regression trees if the responses are numerical and they are considered as classification trees if the responses are categorical.

In the developed CART algorithm by Breiman et al. (1984), which is a recursive method, the decision trees are constructed by splitting the training set using predictors to create two leaf nodes repeatedly. To choose the best predictor, Breiman et al. (1984) uses Gini impurity, which measures the chance of incorrect classification of a randomly chosen sample from the set as follows:

$$G(n) = \sum_{x \neq y} p(\mathbf{x}|n)p(\mathbf{y}|n), \tag{3.24}$$

where $G(n)$ is the Gini impurity index at node n, and $p(\mathbf{x}|n)$ and $p(\mathbf{y}|n)$ are the relative frequency of two classes (categories) at node n, respectively. The Gini impurity index is equal to zero if all samples at one node belongs to one class. On the other hand, in regression trees, the mean squared error (MSE) measure is used as the impurity index. Therefore, the CART algorithm for regression problems utilizes a split to the minimize MSE of predictions, compared to the training dataset.

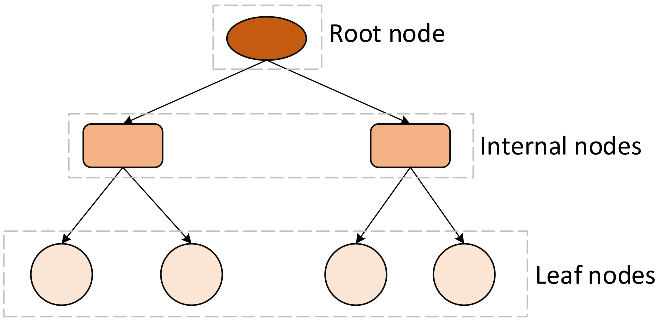


Figure 3.9: Decision tree diagram.

3.7 Naive Bayes Classifier

Naive Bayes (NB) classifier uses Bayes' theorem to classify the given samples (data point) into different classes (Friedman et al., 2001). Bayes' theorem can be defined as:

$$P(A|B) = \frac{P(B|A) \cdot P(A)}{P(B)}, \tag{3.25}$$

where $P(A|B)$ and $P(B|A)$ are posterior probability and likelihood, respectively, $P(A)$ is the prior probability, and $P(B)$ is called evidence. Equation 3.25 presents how often A happens, given that B happens, if it is known that how often B happens, given that A happens, and how likely A and B are happening independently.

The NB classifiers simply assumes that the input features (predictors) are independent. Therefore, the algorithm estimates the probability of each predictor given the corresponding class (Friedman et al., 2001):

$$P(C_j|\mathbf{x}_n) = \frac{P(C_j) \cdot \prod_{m=1}^{M} P(\mathbf{x}_{mn}|C_j)}{\sum_{j=1}^{J} P(C_j) \cdot \prod_{m=1}^{M} P(\mathbf{x}_{mn}|C_j)}, \tag{3.26}$$

where $\mathbf{x_n} = \{x_{1n}, x_{2n}, x_{Mn}\}$, which are the input features of the nth sample, and C_j is the class label of $\mathbf{x_n}$. It is worth nothing that the class label of $\mathbf{x_n}$, C_L, can be determined by:

$$\begin{aligned} x_N \in C_L \Leftrightarrow L &= \underset{j}{\operatorname{argmax}}\{P(C_j|\mathbf{x}_n)\} \\ &= \underset{j}{\operatorname{argmax}}\left\{P(C_j) \cdot \prod_{m=1}^{M} P(\mathbf{x}_{mn}|C_j)\right\}. \end{aligned} \tag{3.27}$$

3.8 Fuzzy K-Nearest Neighbor Classifier

The K-Nearest Neighbor (KNN) is categorized as one of the least complex but important data mining algorithms. In a traditional KNN algorithm (Cover and Hart, 1967) the inputs (sample data points) of the algorithm are set of predictors and the outputs are class labels. In this algorithm, the distance between the new sample and its K-nearest neighbors are calculated and a class is assigned using the majority voting technique. So, the new sample belongs to the same class as the majority of its K-nearest neighbors. This algorithm can also be used in regression problems and the predicted value of the new sample is the average of its K-nearest neighbors. There are some improved versions of the KNN algorithm, such as weighted KNN and fuzzy KNN (Stone, 1977). In a simple version of a weighted KNN algorithm, some weights are assigned to the nearest neighbors and these nearest neighbors will have more contribution in the classification or regression task. A common weighting assignment technique utilizes a weighting factor equal to $\frac{1}{d}$, where d is the distance between the sample and its corresponding nearest neighbor.

In this research, a Fuzzy KNN (FKNN) algorithm was used. The main difference between KNN and FKNN is that in FKNN, a class membership is assigned to a new sample instead of simply assigning a binary class label (Keller et al., 1985). This approach can be easily used in multi-class classification problems and the value of each class membership can be considered a certainty measure. The higher the certainty measure (membership value) of a class, the more likely the new sample belongs to this class. As an example, in a classification with three classes, if the membership values are 0.8, 0.1, and 0.1, one can confidently conclude that the winning class is the first class. Alternatively, if the membership values are 0.4, 0.5, and 0.1, further investigation is needed between the first and second classes to assign the correct class to the sample confidently.

To develop this algorithm, first the membership value of a new sample should be determined. Suppose a dataset of N labeled samples is given as $Z = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N\}$ and u_{np} is the assigned membership value to the nth labelled sample for the pth class. There are several ways to assign membership values to labelled sampled (Keller et al., 1985), but one of the most popular and routine approaches will be described here. The complete membership value can be assigned to the labelled samples in their correct class and zero membership value in other classes (Keller et al., 1985). The membership value of the new unlabelled sample can be calculated as (Keller et al., 1985):

$$U_P(\mathbf{x}) = \frac{\sum_{n=1}^{K} u_{np} \cdot \left(\frac{1}{\|\mathbf{x}-\mathbf{x}_n\|^{\frac{2}{m-1}}} \right)}{\sum_{n=1}^{K} \left(\frac{1}{\|\mathbf{x}-\mathbf{x}_n\|^{\frac{2}{m-1}}} \right)}, \tag{3.28}$$

where K is the number of nearest neighbors and m controls the weight of the distance between labelled samples and new unlabelled sample, which calculates the membership value of the new sample. If $m = 2$, the membership value of each K-nearest neighbor is weighted by the inverse distance of a corresponding member from the new unlabelled sample.

3.9 Adaptive Network-based Fuzzy Inference System

The adaptive Network-based Fuzzy Inference System (ANFIS) network was introduced by Jang (1993). It is actually a fuzzy system which has a network structure. The two main learning algorithms used in ANFIS are called back propagation and a hybrid algorithm. These learning algorithms search a feasible space to find the best parameters of the network iteratively by minimising the cost function (Equation 3.22).

The ANFIS network consists of a number of nodes which are connected together by directional links through different layers (Jang, 1993). The structure of the ANFIS network is shown in Figure 3.10. There are two different types of nodes: adaptive and fixed. The output of a fixed node is only dependent on the output of the previous layer, i.e., the nodes of Layer 2, 3 and 5, whereas the output of an adaptive node is also dependent on its input parameters, i.e., the nodes of Layer 1 and 4.

In general, the ANFIS network consists of five layers, connecting n inputs to one output y. For the sake of simplicity and without loss of generality, we will assume that the fuzzy inference system used in the ANFIS network has only two inputs x_1

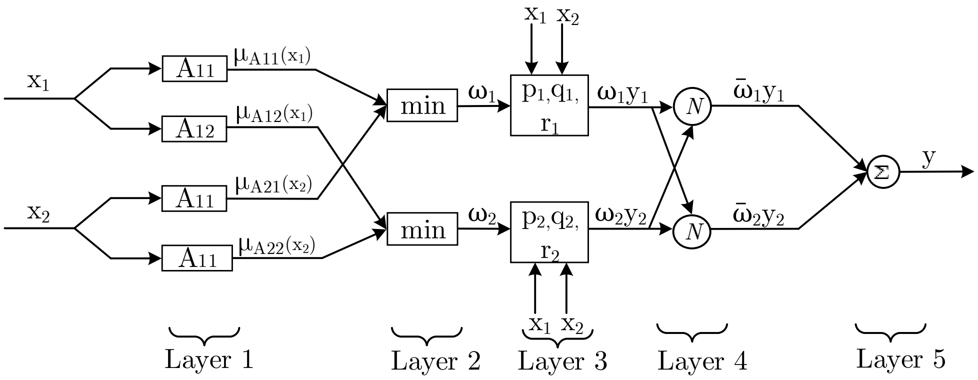


Figure 3.10: An example of ANFIS network.

and x_2 and one output y as shown in Figure 3.10. The Takagi-Sugeno type fuzzy inferences used in the ANFIS network are as follows:

$$\text{Rule 1: if } x_1 \text{ is } A_{11} \text{ and } x_2 \text{ is } A_{21}, \text{ then } y_1 = p_1 \cdot x_1 + q_1 \cdot x_2 + r_1, \tag{3.29}$$

$$\text{Rule 2: if } x_1 \text{ is } A_{12} \text{ and } x_2 \text{ is } A_{22}, \text{ then } y_2 = p_2 \cdot x_1 + q_2 \cdot x_2 + r_2, \tag{3.30}$$

where A_{11}, A_{12}, A_{21}, and A_{22} are membership functions, x_1 and x_2 are the inputs and p's, q's, and r's are the tunable network parameters.

The most commonly used membership functions are increasing, decreasing, triangular and trapezoidal functions. Figure 3.11 represents a schematic diagram of the fuzzy inference in the ANFIS network. The ω_1 and ω_2 are the t-norms of the two pairs of membership values $\{\mu_{A_{11}}(x_1),\ \mu_{A_{21}}(x_2)\}$ and $\{\mu_{A_{12}}(x_1),\ \mu_{A_{22}}(x_2)\}$, respectively.

In the ANFIS network, shown in Figure 3.10, the output of the first layer, called the fuzzification layer, are the membership values. The membership functions assign a value from the interval $[0, 1]$ to each input. A membership function can be defined as:

$$\mu_F : U \to [0, 1] \tag{3.31}$$

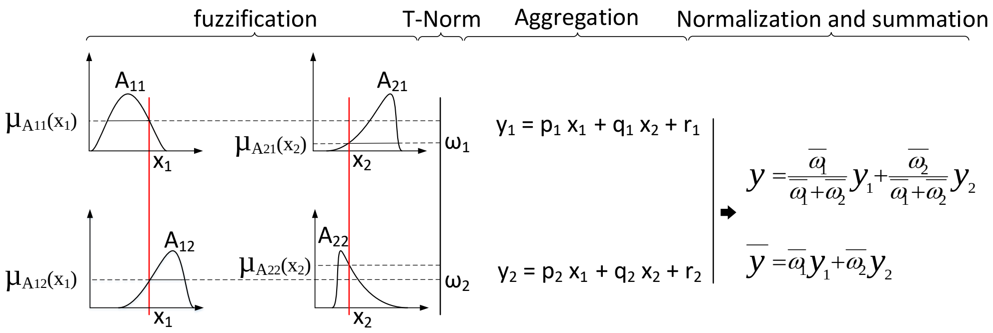

fuzzification
T-Norm
Aggregation
Normalization and summation
A_{11}
A_{21}
$\mu_{A11}(x_1)$
$\mu_{A21}(x_2)$
x_1
x_2
ω_1
$y_1 = p_1 x_1 + q_1 x_2 + r_1$
$y = \frac{\overline{\omega_1}}{\omega_1+\omega_2} y_1 + \frac{\overline{\omega_2}}{\omega_1+\omega_2} y_2$
A_{12}
A_{22}
$\mu_{A12}(x_1)$
$\mu_{A22}(x_2)$
x_1
x_2
ω_2
$y_2 = p_2 x_1 + q_2 x_2 + r_2$
$\overline{y} = \overline{\omega_1} y_1 + \overline{\omega_2} y_2$

Figure 3.11: Schematic diagram of an ANFIS network.

In the implication layer (layer 2) the t-norms of the membership values are determined as illustrated in Figure 3.11. In the third layer (aggregation layer) y_1 and y_2 are defined using three tunable parameters (p, q, and r) and are multiplied by the t-norms of the membership values (ω_1 and ω_2). The outputs of the aggregation layer are then normalised in layer four (normalisation layer). Lastly, the normalised y_1 and y_2 are summed in the summation layer.

3.10 Random Vector Functional Link Network

The Random Vector Functional Link (RVFL) network was first introduced by Pao et al. (1992a). The main drawbacks of feedforward neural networks, which use back propagation optimisation algorithms, are slow to converge and can be easily trapped in local minima. In the RVFL network, the weights from input layer to the hidden layer (enhancement layer) can be randomly selected from a feasible domain and are kept unchanged during the learning stage. The RVFL network structure is shown in Figure 3.12. In the RVFL network, there are direct links between the inputs and outputs, which help to improve the performance of the network.

The random weights r_{ij} from the input layer to the hidden layer, as suggested in (Alhamdoosh and Wang, 2014), are randomly generated from a uniform distribution, [-S, S], where S is called scale factor and should be set based on the training

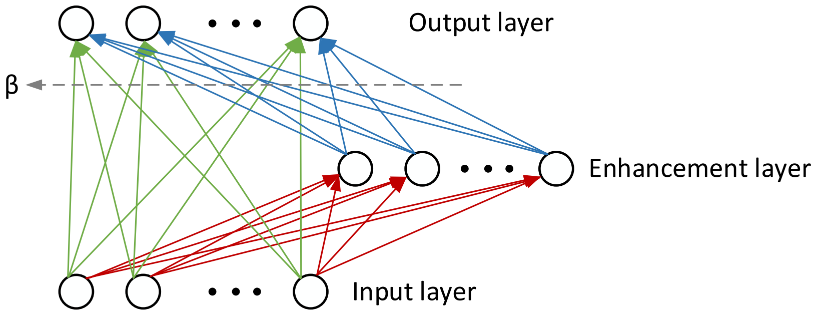


Figure 3.12: An example of RVFL network.

dataset. This ensures that the activation functions $f(r_j^{\mathrm{T}}x + b_j)$ will not saturate. Thus, in the RVFL network, the output weights β should be determined during the training stage. These output weights can be found by solving Equation 3.32.

$$y_i = f_i^{\mathrm{T}}\beta, \quad i = 1, 2, \ldots, m \tag{3.32}$$

where m is the number of samples in the dataset, y is the target and f is the vector of generated random weights and inputs.

In practice, regularized least squares is used to solve Equation 3.32. There are two main reasons for using regularized least squares instead of the ordinary least squares. The first one is that sometimes the number of variables are higher than the number of samples, such that, the ordinary least squares problem is considered as an *ill-posed* problem and the optimisation problem has infinite solutions (Hansen, 1998). Regularized least squares is also used to improve the generalisation of the model by forcing the optimisation problem to find more *sparse* solution.

In general, there are two types of RVFL networks, iterative and closed-form. The implemented RVFL network in this research is a closed-form RVFL network. In a closed-form RVFL network, pseudo-inverse (Igelnik and Pao, 1995; Pao and Phillips, 1995) approaches can be used to find a solution in a single learning step. One of the most commonly used pseudo-inverse methods is called the Moore-Penrose pseudo-inverse, which solves Equation 3.32 as (Pao and Phillips, 1995):

$$\beta = F^{+}Y, \tag{3.33}$$

where F is the concatenated vector of generated random, weights and inputs for all data samples and Y are the targets vector of all samples. The '+' represents the Moore-Penrose pseudo-inverse. An alternative approach to solve Equation 3.32 is using ridge regression (or $L2$ norm regularized least square), which tends to solve (Murphy, 2012),

$$\sum_i (y_i - f_i^{\mathrm{T}}\beta)^2 + \lambda \parallel \beta \parallel, \quad i = 1, 2, \ldots, m \tag{3.34}$$

$$\therefore \beta = F(F^{\mathrm{T}}F + \lambda I)^{-1}Y, \tag{3.35}$$

where λ is the regularisation parameter which needs to be tuned properly.

3.11 Kernel Ridge Regression

Ridge Regression was first introduced by Hoerl and Kennard (1970). It is categorised as a shrinkage method because it imposes a constraint on the regression coefficients and prevents them from being very large. In fact, in this method the ridge coefficients ($\hat{\beta}_{ridge}$) minimise,

$$\underset{\beta}{\operatorname{argmin}} \left\{ \sum_{i=1}^{N} (y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j)^2 + \lambda \sum_{j=1}^{p} \beta_j^2 \right\}, \tag{3.36}$$

where $\lambda \geq 0$ and is called the penalty factor, and x and y are simply inputs and targets, respectively.

If we rewrite the Equation 3.36 in matrix form, the residuals sum of squares (RSS) is as follows (Hoerl and Kennard, 1970):

$$\mathrm{RSS}(\lambda) = (\mathbf{y} - \mathbf{X}\beta)^{\mathrm{T}}(\mathbf{y} - \mathbf{X}\beta) + \lambda\beta^{\mathrm{T}}\beta, \tag{3.37}$$

and the solution of Equation 3.37 is given by (Hoerl and Kennard, 1970),

$$\hat{\beta}_{ridge} = (\mathbf{X}^{\mathrm{T}}\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^{\mathrm{T}}\mathbf{y}, \tag{3.38}$$

where I is the identity matrix. A geometric representation of the ridge regression method is illustrated in Figure 3.13. Each ellipse represents its corresponding RSS. The smaller the ellipse, the smaller the RSS. The circle shows the constraints in

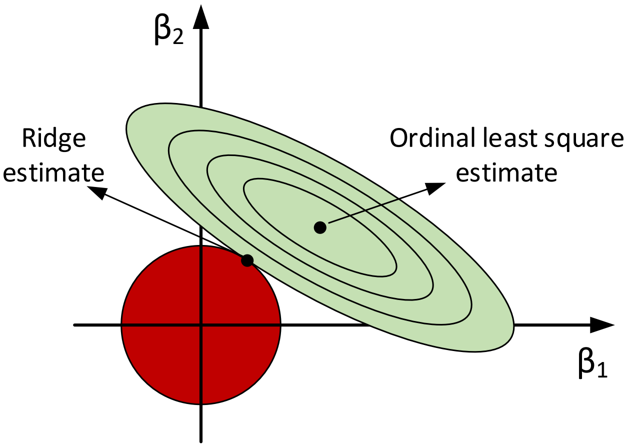


Figure 3.13: A geometric representation of ridge regression.

ridge regression. The ridge estimate point in the figure is achieved by minimising the size of the circle and ellipses, simultaneously.

Ridge regression can be kernelised by replacing $\mathbf{X}\mathbf{X}^{\mathrm{T}}$ with an appropriate kernel function $(\mathbf{K})$ in Equation 3.38. A list of commonly used kernel functions are given in Equation 3.18. The kernelised form of Equation 3.38 is as follows Murphy (2012):

$$\hat{\beta}_{ridge} = \mathbf{X}^{\mathrm{T}}(\mathbf{K} + \mathbf{I})^{-1}\mathbf{y}. \tag{3.39}$$

However, the $\mathbf{X}^{\mathrm{T}}$ term still present in Equation 3.39. In order to remove this term, we define a variable α as (Murphy, 2012):

$$\boldsymbol{\alpha} \triangleq (\mathbf{K} + \lambda\mathbf{I})^{-1}\mathbf{y} \tag{3.40}$$

and Equation 3.39 can be rewritten as:

$$\hat{\beta}_{ridge} = \mathbf{X}^{\mathrm{T}}\boldsymbol{\alpha} = \sum_{i=1}^{N} \alpha_i \mathbf{x}_i. \tag{3.41}$$

Finally, a closed form for kernel ridge regression (KRIDGE) can be obtained by looking at the predicted value, which is,

$$\hat{f}(\mathbf{x}) = \hat{\beta}_{ridge}^{\mathrm{T}}\mathbf{x} = \sum_{i=1}^{N} \alpha_i x_i^{\mathrm{T}} x = \sum_{i=1}^{N} \alpha_i \kappa(x, x_i) \tag{3.42}$$

3.12 Echo State Network

In Section 3.5, MLP was briefly described as one of the most commonly used feedforward neural networks (FFNN). As illustrated in Figure 3.5, there are only forward connections between the neurons. There is another type of neural network referred to as the Recurrent Neural Network (RNN). The main difference between the FFNN and RNN is the cyclical connections in the hidden layer of RNN as shown in Figure 3.14. Different types of RNNs have been proposed, such as the Echo State Network (ESN) (Jaeger, 2001), the Elman Network (Elman, 1990), the Time Delayed Network (Lang et al., 1990), and Jordan Network (Jordan, 1990).

In this study, an Echo State Network (ESN) was used as a time series forecasting algorithm. The ESN was first introduced by Jaeger (2001) and it is actually a RNN with a fixed non-trainable sparse reservoir part and a linear readout.

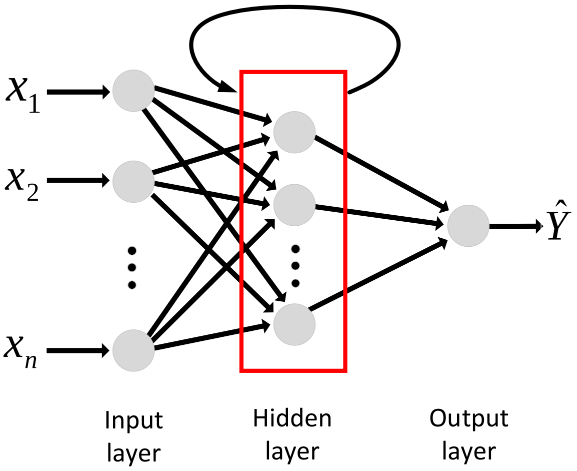


Figure 3.14: An example of a RNN.

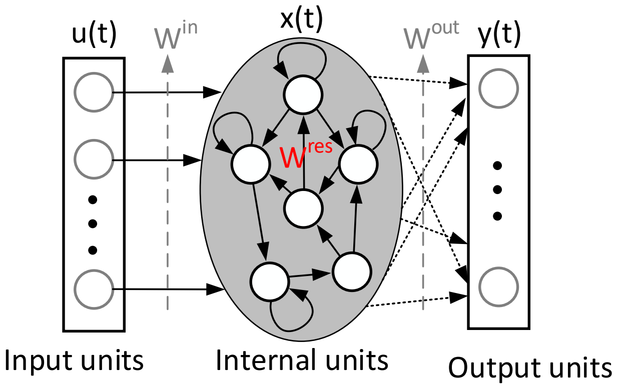


Figure 3.15: An example of an ESN.

The ESN has K input units, N internal units (reservoir), and L output units. Figure 3.15 represents the basic architecture of an ESN. It should be noted that, for the sake of simplicity and without loss of generality, there are neither feedback connections from the output units to the internal units nor from the inputs to the outputs. Activations of input, internal, output units at time step t are given as (Jaeger, 2001):

$$\mathbf{u}(t) = (u_1(t),\ u_2(t), \dots,\ u_K(t)), \tag{3.43}$$

$$\mathbf{x}(t) = (x_1(t),\ x_2(t), \dots,\ x_N(t)), \tag{3.44}$$

$$\mathbf{y}(t) = (y_1(t),\ y_2(t), \dots,\ y_L(t)). \tag{3.45}$$

In addition, there are three weight matrices, input-internal, internal-internal, and internal-output, which are denoted by $\mathbf{W}^{\text{in}}_{N\times K}$, $\mathbf{W}^{\text{res}}_{N\times N}$, and $\mathbf{W}^{\text{out}}_{L\times N}$, respectively. Equation 3.46 and 3.47 formulate the update phase of the internal units and compute the linear readout, respectively (Jaeger, 2001).

$$\mathbf{x}(t+1) = f(\mathbf{W}^{\text{in}} \cdot \boldsymbol{u}(t+1) + \mathbf{W}^{\text{res}} \cdot \mathbf{x}(t)), \tag{3.46}$$

$$\mathbf{y}(t+1) = \mathbf{W}^{\text{out}}\mathbf{x}(t+1). \tag{3.47}$$

where f is the reservoir activation function. The two most popular and commonly used reservoir activation functions are *tanh* and *sig*.

As mentioned earlier, the elements of two matrices, $\mathbf{W}^{\mathrm{in}}$ and $\mathbf{W}^{\mathrm{res}}$, are fixed, with random numbers from a uniform distribution assigned before the network starts training. In ESN the $\mathbf{W}^{\mathrm{res}}$ matrix is scaled as follows (Jaeger, 2001):

$$\mathbf{W}^{\mathrm{res}} \leftarrow \frac{\alpha \cdot \mathbf{W}^{\mathrm{res}}}{|\lambda_{max}|}, \tag{3.48}$$

where $|\lambda_{max}|$ is the spectral radius of $\mathbf{W}^{\mathrm{res}}$ and $0 < \alpha < 1$ which should be fine-tuned. To train the ESN, first the $\mathbf{W}^{\mathrm{res}}$ is scaled by α, then the $\mathbf{W}^{\mathrm{out}}$, which is the trainable part of ESN, is computed using a simple linear regression model as follows:

$$\mathbf{y} = \mathbf{X}\mathbf{W}^{out} + \boldsymbol{\varepsilon}, \tag{3.49}$$

where

$$\begin{aligned}\mathbf{X} &= [\mathbf{x}(l),\ \mathbf{x}(l+1),\ \ \ldots,\ \mathbf{x}(l+N-1)]^{\mathrm{T}},\\ \mathbf{y} &= [y(l),\ y(l+1),\ \ \ldots,\ y(l+N-1)]^{\mathrm{T}},\end{aligned}$$

and l is the index of the first training sample since the first $(l-1)$ samples are not considered valid. Such initial transient are "washed" out of the network by not including these values in processing, $\boldsymbol{\varepsilon}$ is zero mean Gaussian noise, and N is the size of training set. One possible way to solve Equation 3.49 is to use the pseudoinverse,

$$\mathbf{W}^{\mathrm{out}} = \mathbf{X}^{\dagger}\mathbf{y} = (\mathbf{X}^{\mathbf{T}}\mathbf{X})^{-1}\mathbf{X}^{\mathrm{T}}\mathbf{y}. \tag{3.50}$$

The pseudo inverse was also applied in Section 3.11 (ridge regression), and such regularisation methods can be used to achieve good generalisation. Therefore, Equation 3.50 can be rewritten as:

$$\mathbf{W}^{out} = (\lambda\mathbf{I} + \mathbf{X}^{\mathrm{T}}\mathbf{X})^{-1}\mathbf{X}^{\mathrm{T}}\mathbf{y}, \tag{3.51}$$

where λ is the regularisation parameter. It should be noted that a cross-validation technique can be used to estimate λ. The details of cross validation technique can be found in Friedman et al. (2001).

3.13 Discussion

All the individual algorithms introduced in this chapter have some advantages and disadvantages.

For example, SVM can approximate complex nonlinear functions and works very well with small training set but it suffers from a slow training phase on large training sets. This is due to the computation complexity of the matrices of kernel values in the training phase. In addition, there is no direct SVM for multi-class classification problem and two-class SVMs must be combined to deal with these kinds of problems.

As an advantage, unlike many other algorithms, the KNN algorithm makes few assumptions about the dataset. The only assumption is proximity, which means similar instances (samples) should have similar labels. This method is also a non-parametric approach, which means there is no need to fit a distribution to the dataset. Downsides include in the situation of missing values, the algorithm does not work is sensitive to outliers irrelevant attributes. Furthermore, the KNN algorithm is computationally expensive because there is no learning phase for this algorithm and the algorithm is just storing all the training instances and then doing comparisons at the time of testing, which typically needs lots of storage and time to do the comparisons. The rate of incorrect classification for this algorithm on high dimensional problem is also high.

Naive Bayes classifier is one of the simplest algorithms to implement and it is also easy to understand, however the main drawback of this method is the unrealistic assumption of feature independence. Another problem is due to an imbalanced dataset, which results in skewed probabilities (Rennie et al., 2003). To obtain desirable results, and to be comparable with other algorithms, a large training set should be used, which is sometimes impossible to collect in practice.

The main advantages of neural networks are the ability of this method to approximate almost all nonlinear functions and its robustness to outliers. However, it is sometimes difficult to fine-tune an algorithm and to avoid the overfitting problem.

Finding the optimum topology of the network is not always a straightforward process. Since neural networks require a lot of data for training, its training phase takes a long time.

To overcome the shortcomings of single learning algorithms, and to make a robust classification or time series forecasting algorithm, an ensemble approach is introduced in Chapter 5 and Chapter 6 to develop fault classification and dissolved gas forecasting algorithms. Figure 3.16 shows a generic diagram of an ensemble learning model with N learners. Each learner is trained individually using input $\mathbf{X}$ to estimate the function f such that minimize the error of the prediction. Then, the outputs of all ensemble members are combined to make a decision. There are several approaches to build and combine an ensemble (Polikar, 2006; Ren et al., 2016).

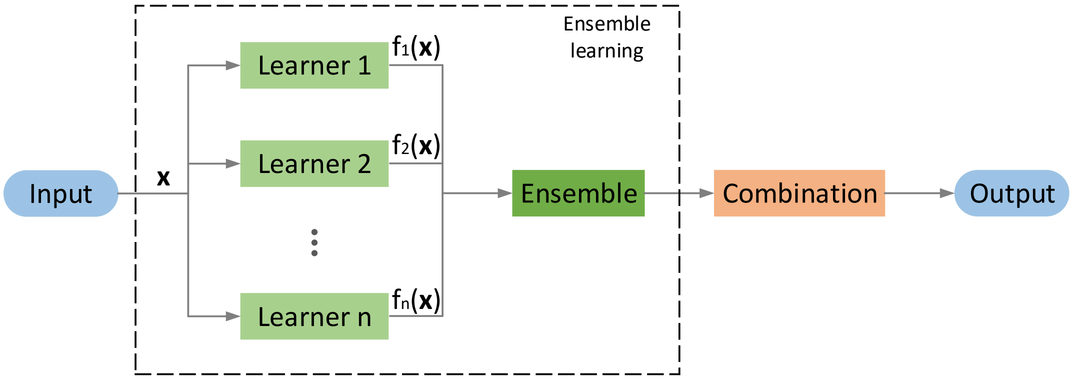


Figure 3.16: A generic diagram of an ensemble learning model.

Chapter 4

Condition Assessment of Transformers Load Tap Changers Using Support Vector Machine

4.1 Overview

In this chapter, a single classifier is used to classify the faults of transformer load tap changers using dissolved gas analysis. A SVM classifier is used to diagnose the faults of transformer load tap changers. The results of the developed algorithms are compared with a well-known transformer load tap changers fault diagnosis technique called modified Duval Triangle, which is similar to the original Duval Triangle technique discussed in Section 2.4.3.

4.2 Introduction

As mentioned in Section 2.3, one of the most widely used tests to diagnosis incipient faults in transformer load tap changers (LTCs) is dissolved gas in oil analysis (DGA). Several standards and diagnosing techniques (i.e. International Electrotechnical Commision (IEC) 60599 (IEC, 2007) and Institute of Electrical and Electronics Engineers (IEEE) C57.104 standards (IEEE, 2009)) have been developed to detect faults in transformers, but not specifically in oil-type LTCs. The main problem in applying these conventional methods for assessing the condition of LTCs is due to the arcing in oil through the normal operation of load tap changers. The arcing tends to produce hydrocarbons in the oil such as hydrogen and acetylene, which leads to incorrect diagnosing by these methods. Duval (2008) proposed an alternative method for this purpose by developing the modified Duval Triangle method to diagnose faults in LTCs. In this chapter, one of the most powerful machine learning algorithms for classification problems, called support vector machine (SVM), is applied to the modified Duval Triangle method to classify LTCs faults. The developed algorithms are first trained using DGA samples for LTCs and then the trained models are used for diagnosing LTCs faults on the testing set. The main motivation for using SVM in this study was the size of the DGA dataset. The DGA dataset for LTCs was not large enough to reliably train an ANN to an acceptable degree of accuracy or to extract comprehensive rules using fuzzy logic methods. So, because of the capability of the SVM in dealing with classification problems with small training set, this method was chosen to classify faults of LTCs.

4.3 Transformers load tap changers fault diagnosis

As mentioned in Section 2.4.3, one of the most commonly used conventional methods for diagnosing faults of transformers is the Duval Triangle method (Duval, 1974). A modified version of this technique for LTCs has been proposed in (Duval, 2008). This is also a visual technique for LTCs fault diagnosis using DGA. Figure 4.1 shows the modified Triangle used for diagnosing faults and the distinct zones corresponding to each fault. Each Triangle coordinate is determined as follows:

$$\%\mathrm{CH}_4 = \frac{x}{x+y+z} \times 100, \tag{4.1}$$

$$\%\mathrm{C}_2\mathrm{H}_2 = \frac{y}{x+y+z} \times 100, \tag{4.2}$$

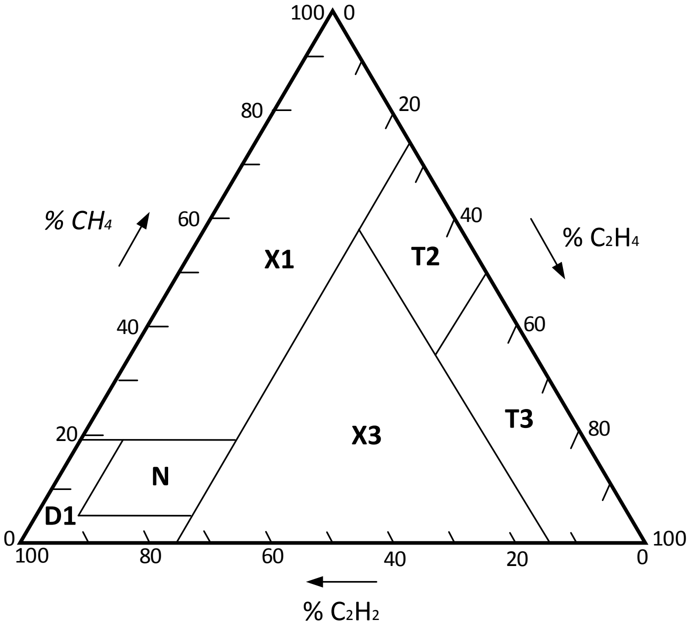


Figure 4.1: Modified Duval Triangle for LTCs (Duval, 2008).

$$\%C_2H_4 = \frac{z}{x+y+z} \times 100, \tag{4.3}$$

where x, y, and z are the amount of CH_4, C_2H_2, and C_2H_4 in ppm, respectively.

Table 4.1 indicates the fault types corresponding to each zone. Four major fault types can be diagnosed using this method, i.e., the discharge of low energy (D1), the discharge of high energy (D2), and overheating (thermal faults) over two different temperature ranges: T2 ($300\,^{\circ}\mathrm{C} \leqslant \mathrm{T} \leqslant 700\,^{\circ}\mathrm{C}$) and T3 ($\mathrm{T} \geqslant 700\,^{\circ}\mathrm{C}$).

In this study, the DGA test results for LTCs have been extracted from (Duval, 2002). The distribution of the DGA samples across all the classes are given in Table 4.2.

In this study, three different SVM structures are used to classify faults of transformers LTCs. The two well-known (default) SVM modes, one-versus-one and one-versus-all, are compared with a rather complicated method as shown in Figure 4.2. To identify the five states (normal, T3, T2, D1, and D2), four SVM

Table 4.1: Fault zones and corresponding fault types in modified Duval Triangle for LTCs.

Fault zone	Fault type
N	Normal operation
T3	Thermal fault (T $\geqslant 700\,^{\circ}$C), severe coking
T2	Thermal fault($300\,^{\circ}$C $\leqslant$ T $\leqslant 700\,^{\circ}$C), light coking
X3	Thermal faults (T3 and T2) are in progress, severe arcing (D2)
D1	Discharge of low energy
X1	Discharge of low energy (D1) and thermal fault are in progress

Table 4.2: Number of DGA samples for each fault class.

Fault types	No fault	Thermal (T2, T3, X3)	Arcing (D1, D2, X3)
# of samples	6	31	16

classifiers are used in Figure 4.2. The order of tree based SVM method (Figure 4.2) for transformers LTCs fault diagnosis is inspired by conventional DGA fault diagnosis methods. The implemented fault diagnosis algorithms using SVM consists of six main steps, which are described as follows:

1. *Normalization*: All the DGA samples are first normalized to zero mean and unit standards deviation.

2. *Divide dataset*: The DGA dataset is then divided into training and test set. The test set is set aside and it is not used for anything except reporting the accuracy of the models.

3. *Select the best kernel function*: As listed in Equation 3.18 (page 33), there are four popular kernel functions. Since the Gaussian kernel function performed a better classification task on the available dataset, this kernel was chosen to be used in the SVM algorithms to classify faults of transformers LTCs.

4. *Find the best parameters*: There are two parameters in the SVM algorithms that need to be fine tuned properly. It should be mentioned that all the SVM algorithms are implemented in MATLAB 2016b using `fitcsvm` function which applies a heuristic procedure using subsampling to find the optimum values for hyper-parameters. These are the C parameter in Equation

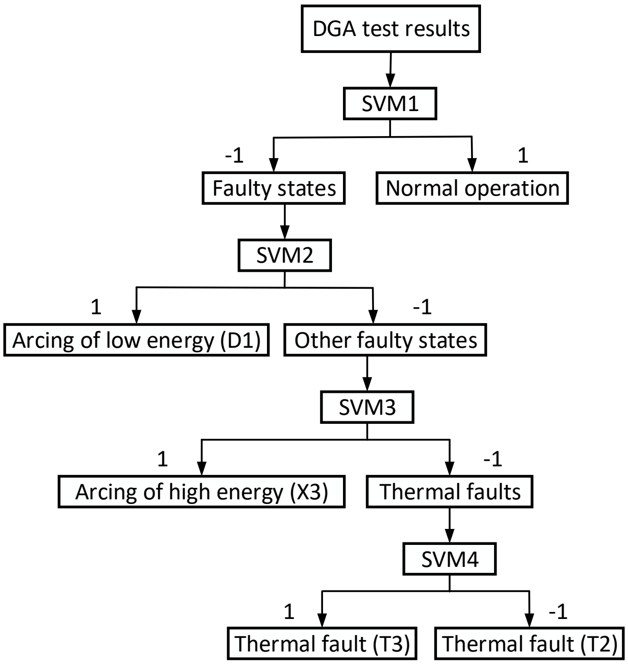


Figure 4.2: SVM classifiers for LTCs fault diagnosis.

3.6 (page 30) and the σ parameter of the Gaussian kernel function (Equation 3.18) are selected using a heuristic approach. The heuristic procedure used here is very similar to Randomized Parameters Optimization in Python *scikit-learn* package which uses a randomized search over feasible set of parameters (Pedregosa et al., 2011).

5. *Train SVM algorithms*: The SVM algorithms are trained using selected parameters on the training set.

6. *Test SVM algorithms*: Lastly, the trained SVM algorithms are examined on the testing set and the classification accuracy are reported.

The aforementioned procedure is applied for one-versus-rest and one-versus-one SVM algorithms and for training all four SVM classifiers in Figure 4.2. As illustrated, SVM1 is trained to diagnose the normal operation state from other faulty states and the output of SVM1 is set to +1 for normal operation and -1 for other cases. SVM2 and SVM3 are separately trained for diagnosing discharge of low and high energy classes, respectively. The output of these two SVMs are set to +1 for D1 and D2 cases in SVM2, and SVM3 and -1, otherwise. SVM4 is also trained to diagnose thermal faults of T2 from T3 in a similar way.

4.4 Comparison of different SVM models

Fifteen DGA samples of LTCs, which are given in Table 4.3, were considered as a testing set. In the following, the capability of the four SVMs in Figure 4.2 for classifying faults of these samples is investigated.

Table 4.3: The actual value of dissolved gas samples as testing set.

No.	Actual fault	Dissolved gases [ppm]				
		H_2	CH_4	C_2H_4	C_2H_2	C_2H_6
1	No fault	1215	5386	6400	35420	963
2	No fault	43	8	11	61	2
3	Low energy arcing	1084	188	166	769	8
4	Low energy arcing	47	12	17	144	31
5	Low energy arcing	1317	608	2278	8739	841
6	High energy arcing	391	164	293	736	14
7	High energy arcing	9083	3279	9606	8527	1136
9	Thermal fault T2	69	450	329	41	137
10	Thermal fault T2	859	843	3574	5155	843
11	Thermal fault T3	591	6088	11433	193	2626
12	Thermal fault T3	1312	39981	120319	774	35146
13	Thermal fault T3	19700	117000	142000	3490	44600
14	Thermal fault T3	217	749	1754	33	171
15	Thermal fault T3	2217	53434	235024	1633	55535

All SVMs were tested for classifying different classes and the output of the classifiers are summarised in Table 4.4. SVM1 was tested for diagnosing the normal operation cases from other faulty ones. This classifier (SVM1) was able to diagnose all the normal cases correctly, and these instances are represented by 1 in the output pattern of the algorithm. However, there is one false positive case where

Table 4.4: Outputs of four SVM classifiers on testing set.

Fault type	Output pattern of SVM classifiers	DC[a]	DI[b]	ND[c]
Normal	SVM1=[1 1 -1 1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1]	2	1	0
D1	SVM2=[1 1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1]	2	0	1
D2	SVM3=[1 1 1 1 -1 -1 -1 -1 -1 -1]	4	0	0
T3	SVM4=[1 1 1 1 -1 -1]	4	0	0

[a] Diagnosed correctly
[b] Diagnosed incorrectly
[c] Not diagnosed

a low energy arcing (D1) faulty case was classified incorrectly as a normal operation case. Further investigation revealed that this sample is actually located very close to the D1 fault zone on the Duval Triangle and this misclassification may be addressed by using a larger training set, which leads to a more accurate decision boundary. Figure 4.3 graphically shows the Duval triangle fault diagnosis on the testing set. As is clear from Figure 4.3, the misclassified case in SVM1 (case 5) lies very close to the boundary between D1 and normal zones.

SVM2 classifies the low energy arcing (D1) from other faulty cases. As it is shown in Table 4.4, there is only one case which is not diagnosed correctly as D1. This is the case number 6 on Figure 4.3, which is very close to the normal operation zone and SVM2 was not able to classify it correctly.

SVM3 was tested to diagnose high energy discharge (D2) cases and the output pattern of this classifier in Table 4.4 shows a successful fault classification. The output of SVM4 for diagnosing thermal faults of T2 and T3 confirms that a successful fault classification of these two classes can be obtained by using this classifier.

Table 4.5 compares the overall classification performance of transformers LTCs using different SVM models with the modified Duval Triangle technique. The classification accuracy is calculated as:

$$Accuracy = \frac{TP + TN}{Number\ of\ test\ samples}, \tag{4.4}$$

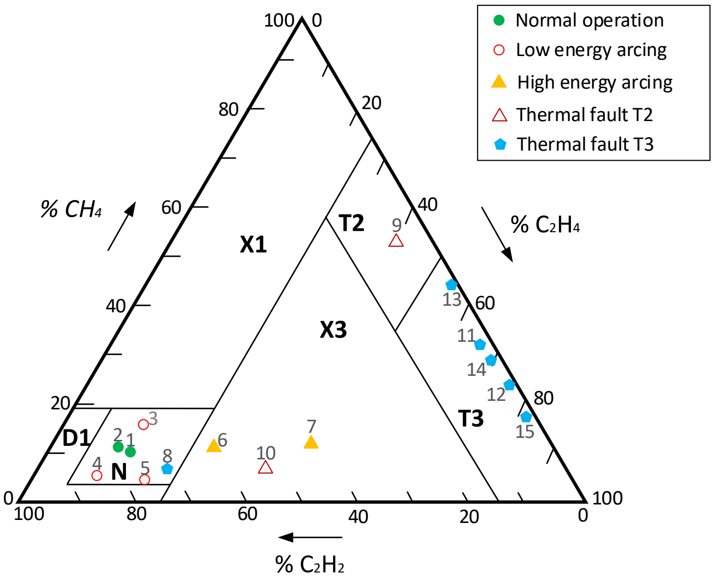


Figure 4.3: Graphical representation of the testing dataset on the Duval triangle. It should be noted that sample 8 is for a LTC with large numbers of operations resulting in a severe hot spot.

Table 4.5: Comparison between classification accuracy of SVM models and the modified Duval Triangle method.

Method	Overall classification accuracy (%)
Modified Duval triangle	73.33
Tree based SVM	86.67
one-versus-all SVM	**90.67**
one-versus-one SVM	80

where TP and TN are the number of true positives and true negatives respectively, which represent the capability of the classifier in correctly classifying DGA samples either they belong to the "positive" class or the "negative" class.

As given in Table 4.5, all the developed algorithms outperform the Modified Duval triangle technique. The one-versus-all SVM fault diagnosis method shows a higher classification accuracy compared to other SVM structures. The main shortcoming of the Duval Triangle technique to diagnose transformers LTCs is in diagnosing normal operation from low energy discharge cases (D1) (Figure 4.1), which has the maximum error in diagnosing D1 fault class. In addition, Figure 4.3 shows that the D1 zone on the triangle is small and very close to the normal operation zone, which results in misclassification between D1 and Normal fault classes.

4.5 Summary and discussion

In this chapter, a statistical method, which is called support vector machine (SVM), has been used to diagnose faults of transformers load tap changers, intelligently. First, the algorithm was trained by a training dataset based on the modified Duval Triangle method and then the algorithm was tested using a separate dataset. The numerical results show the capability of the SVM to improve the interpretation accuracy, compared with traditional methods. Fault diagnosis of transformer LTCs using DGA depends on various factors, such as type of LTCs and other environmental conditions, and it may be different from utility to utility. So, applying intelligent methods such as SVM can be a reliable method to improve the accuracy of applied diagnostic techniques.

Although, the fault classification accuracy of the simple proposed algorithms discussed here showed some improvements over other conventional techniques, there are still some questions which needed to be further investigated, namely:

- What if we have a large noisy/imbalanced dataset?
- How can the proposed model be effectively generalised to deal with a new dataset?
- Is it possible to select the best learning algorithm among the various statistical and machine learning algorithms to achieve optimal results?

One of the most popular approaches to achieve high accuracy within a generalised learning system is Ensemble Learning (EL). Ensemble learning enables us to take advantage of using different learning algorithms and to create a more accurate and reliable learning model. There are some very well known ensemble learning algorithms such as Random Forests and AdaBoost. In this research, an evolutionary multi-objective ensemble learning approach was developed to overcome the shortcomings of a single learning algorithm. In the next two chapters, the details of the two developed EL algorithms for fault classification and time series forecasting with an application to power transformers are presented.

Chapter 5

Evolutionary Multi-Objective Fault Diagnosis of Power Transformers

5.1 Overview

In this chapter a two step algorithm for fault diagnosis of power transformers (2-ADOPT) is introduced using a binary version of the multi-objective particle swarm optimization (MOPSO) algorithm. Feature subset selection and ensemble classifier selection are implemented to improve the diagnosing accuracy for dissolved gas analysis (DGA) of power transformers. First, the proposed method selects the most effective features in a multi objective framework and the optimum number of features, simultaneously, which are used as inputs to train classifiers in the next step. The input features are composed of DGA performed on the oil of power transformers along with the various ratios of these gases. In the second step, the most accurate and diverse classifiers are selected to create a classifier ensemble. Finally, the outputs of selected classifiers are combined using the Dempster-Shafer combination rule in order to determine the actual faults of power transformers. In addition, the obtained results of the proposed method are compared to five other scenarios: 1) multi-objective ensemble classifier selection without any feature selection step which takes all the features to train classifiers and then applies MOPSO algorithm to find the best ensemble of classifiers, 2 & 3) a well-known classifier ensemble technique called random forests with standard axis align decision tree splits and KNN as the weak classifiers, 4) another powerful decision tree ensemble which is called oblique random forests, and 5) an ensemble method called AdaBoost with decision stumps as the weak classifier. The comparison results were favourable to the proposed method and showed the high reliability of this method for power transformers fault classification.

5.2 Introduction

Currently, most electricity companies rely on expert individuals to analyse data gathered from transformers and to make a decision about the status of their transformers using conventional methods. This can be difficult when the experts concerned are unavailable. Besides, conventional methods are sometimes unable to generate comprehensive results. Thus, we are developing an intelligent fault diagnosing system that will help electricity companies manage their transformer fleet intelligently (Peimankar and Lapthorn, 2015).

Up to now, most power transformers fault diagnosis and condition assessment models have placed emphasis on single classification algorithms (learning algorithms). Ganyun et al. (Ganyun et al., 2005) used a multi-layer support vector machine (SVM), that consists of three SVM classifiers, to diagnose faults of transformers using the relative content of the five dissolved gases, plus the amount of the most abundant gas, as an input feature vector. Fei et al. (wei Fei and bin Zhang, 2009) proposed a Genetic Algorithm (GA)-based SVM to detect faults of power transformers, which can tune the parameters of a support vector machine. In (wei Fei and bin Zhang, 2009) and (wei Fei et al., 2009) the possibility of forecasting the ratios of dissolved gases has been studied by applying GA-based SVM and PSO-based SVM, respectively. These two studies can enhance the reliability of transformers by providing useful information about the rate of failures in a short and medium period of time. Illias et al. (Illias et al., 2015) proposed a successful PSO based artificial neural network algorithm to diagnose faults of transformers based on DGA. In another study, Illias et al. (Illias et al., 2016) implemented an artificial neural network based method for classifying faults of transformers called hybrid modified evolutionary particle swarm optimization-time varying acceleration coefficient-artificial neural network (MEPSO-TVAC-ANN). In this study, they modified the particle swarm optimization algorithm to achieve a better searching behavior. Souahila et al. (Souahlia et al., 2012) developed a fault diagnosis algorithm using a multi-layer perceptron artificial neural network. They applied a cross validation (Golub et al., 1979) technique to determine the parameters of the model using the value of dissolved gases as inputs. In (Wang et al., 1999) the authors combined a feedforward neural network with an expert system to diagnose the fault of power transformers. They have implemented a two level detection system in which they first classified normal/abnormal cases, and then diagnosed the faults of abnormal transformers. Prior to this, Lin et al. (Lin et al., 1993) had developed a rule-based expert system using fuzzy logic. Another research using fuzzy logic technique for fault diagnosis of power transformers is reported in (Su et al., 2000), which defines several fuzzy rules corresponding to each fault class. In (Guardado et al., 2001) a neural network was trained using five different set of ratios of DGA as input features. Each network was trained twice with two different number of neurons in the hidden layer. Flores et al. (Flores et al., 2011) designed an expert system for fault diagnosis of power transformers using type-2 fuzzy logic systems. In their algorithm, besides the value of dissolved gases, the oil chemical characteristics are also considered as inputs to achieve more comprehensive knowledge about the status of the transformer. Ma et al. (Ma et al., 2015) developed a multi-agent system to monitor and assess the condition of transformers. Their study reported that an SVM classifier has better interpretation accuracy for DGA of power transformers, compared to a radial basis function network. Ashkezari et al. (Ashkezari et al., 2014) investigated the effect of feature selection techniques on improving the classification accuracy of an SVM. Two different feature selection techniques, called correlation based and minimum-redundancy-maximum-relevance, were used to select the most correlated features and assign a health index to each transformer using SVM.

All of the aforementioned works implemented a single objective framework to diagnose faults of power transformers. Although the aforementioned diagnosing algorithms have been well trained, there are still some questions that need to be investigation such as: 1) How the diagnosing algorithm can be generalized to deal with a new dataset, and 2) how can we choose the most accurate classification algorithm which results in maximizing the accuracy?. The purpose of this chapter is to develop an intelligent multi objective framework using machine learning techniques to design a reliable fault diagnosis system that will overcome inaccuracies and uncertainties that exist in conventional diagnosis methodologies.

In machine learning, feature selection techniques are commonly used for dimensionality reduction and finding the most relevant features in order to enhance classification capability (Liu and Motoda, 2007). They have been used in a wide range of real-world applications such as biomedical studies (Mohapatra et al., 2016), face recognition (Panda et al., 2011), and medicine (Bellazzi and Zupan, 2008). In recent years, evolutionary algorithms (EA) have been of great interest to researchers for use as a search algorithm to find the best subset of features in feature selection problems (Alba et al., 2007). Traditionally, most of the feature subset selection approaches use a single objective search algorithm (Li et al., 2004). In this chapter, feature selection is dealt with as a multi-objective optimization problem (Deb, 2001). There is not a single solution for a multi-objective optimization problem that could optimize all objectives simultaneously. Therefore, in multi-objective optimization problems the strategy is not finding an optimal solution but selecting efficient solutions which are called non-dominated solutions in the objective space. Non-dominant solutions have superior performance in all objectives over all other solutions. A single non-dominated solution can be found in each simulation run of a multi-objective algorithm. Since it is desired to find several non-dominated solutions in each run, population-based EAs is one of the best choices for solving multi-objective optimization problems.

Particle swarm optimization (PSO) is categorised as a population-based metaheuristic algorithm developed by Kennedy and Eberhart (Eberhart and Kennedy, 1995). Generally, swarm intelligence predicates agents that are not able to handle a problem individually and try to achieve a unique goal in a swarm. Unlike other evolutionary algorithms, such as the genetic algorithm (GA) (Goldberg and Holland, 1988) and the Ant Colony Optimization algorithm (ACO) (Dorigo and Gambardella, 1997), the mechanism of PSO gives the ability to make a well-balance between local and global optima to achieve an efficient exploration and exploitation in shorter computation time compared to its counterparts. However, one of the drawbacks of PSO is the high sensitivity of this algorithm in terms of parameters, which need to be fine tuned. Some research was done to address this problem and to suggest a way for a better convergence of PSO algorithm (Ghosh et al., 2012; Jiang et al., 2007; van den Bergh and Engelbrecht, 2006). However, the single objective PSO algorithm has been successfully applied in power systems engineering applications (Chaturvedi et al., 2008), fault diagnosis (Chenglin et al., 2011), and reliability engineering (Nieto et al., 2015).

A multi-objective version of PSO, named MOPSO, has been applied to multi-objective optimization problems Coello and Lechuga (2002). In a subsequent study, an archive based MOPSO is introduced by Coello et al. (2004) in 2004. This algorithm is inspired by a traditional PSO algorithm (Eberhart and Kennedy, 1995) to deal with multi-objective problems. Since its introduction, the literature continues to show MOPSO improvements which handle multi-objective problems (Elhossini et al., 2010; Leong and Yen, 2008; Mostaghim and Teich, 2003, 2004; Tripathi et al., 2007; Wang and Yang, 2009; Zhao and Suganthan, 2011). The MOPSO algorithm has shown competitive performance in multi-objective optimization problems compared to the non-dominated sorting genetic algorithm (Deb et al., 2002b), which is a multi-objective version of GA, multi-objective evolutionary algorithm based on decomposition (Zhang and Li, 2007), and the strength Pareto evolutionary algorithm (Zitzler et al., 2001).

In the first phase of the proposed method (2-ADOPT), the multi-objective PSO selects the best subset of features corresponding to each fault class of power transformer. Then, in the second stage, advantage is taken of ensemble learning systems to classify actual faults of transformers. Using ensemble learning increases the chance of selecting more accurate classifiers by avoiding selection of a single weak classifier (Polikar, 2006). Ensemble learning systems are frequently used for decision making in various applications, such as financial (West and Qian, 2005), biomedical (Shi and Qian, 2011), and power engineering (Abraham and Das, 2010; Hu et al., 2012; Peimankar et al., 2016). Generally, all ensemble learning systems consist of three main steps (Polikar, 2006):

1. Sampling from a dataset to make a training set,
2. training a group of classifiers,
3. combining the output of classifiers.

There are five major techniques for classifier selection which are: i) Classifier Fusion, ii) Static Classifier Selection (Kuncheva et al., 2000), iii) Static Ensemble Selection (Yang, 2011), iv) Dynamic Classifier Ensemble (Woods et al., 1996), and v) Dynamic Ensemble Selection (Ko et al., 2008). In this chapter a Static Ensemble Selection approach using the MOPSO algorithm is applied to diagnose faults of power transformers. To classify faults of transformers, two criteria are considered to design a diverse classifier ensemble to classify faults of transformers. First, three types of neural networks (NN) as unstable classifiers, which can define different decision boundaries by selecting different parameters, are used in the ensemble (Brown et al., 2005a). Second, different classifiers are used as base learners, which are Support Vector Machine (SVM) (Cortes and Vapnik, 1995), Fuzzy K-Nearest Neighbour (FKNN) (Keller et al., 1985), Naive Bayes (NB) (Rish, 2001), Kernel Ridge Regression Classifier (KRIDGE) (Murphy, 2012), Random Vector Functional Link (RVFL) (Pao et al., 1992b, 1994; Zhang and Suganthan, 2016b) , Cascade-forward Neural Network (CFNN) and Feed-forward Neural Network (FFNN) (Hornik et al., 1989). Each of these unique classifiers is trained with different parameter settings and training functions. The ensemble is therefore composed of thirty classifiers. A list of classifiers used in this research is given in section 5.3.2. In addition, Dempster-Shafer theory is used as a combination rule for combining the outputs of the classifiers.

The remainder of this chapter consists of 9 sections. In section 5.3, feature subset selection and ensemble classifier selection using MOPSO are explained. Pareto optimality in multi-objective optimization and MOPSO algorithm are explained in section 5.4 and section 5.5, respectively. Section 5.6 gives a brief explanation about Dempster-Shafer theory for combining outputs of classifiers. The two phase proposed method for diagnosing faults of power transformers is introduced in section 5.7. Common performance metrics to evaluate binary classification are listed in section 5.8. Section 5.9 presents experimental results, and lastly, Section 5.10 provides a summary of this chapter.

5.3 Multi-objective feature subset selection and ensemble classifier selection

5.3.1 Feature subset selection

Fourteen different features (dissolved gases and their ratios) are defined to classify a fault in transformers, which are listed in Table 5.1. The solutions to each multi-objective feature subset selection are binary vectors whose lengths equal the number of features. Figure 5.1 shows an arbitrary particle in which the selected features are shown with 1's, while 0's represents the corresponding features are not selected.

Table 5.1: Feature used for fault diagnosis of power transformers.

Features #	Dissolved gases and ratios				
F_1-F_5	H_2[a]	CH_4	C_2H_4	C_2H_6	C_2H_2
F_6-F_{10}	C_2H_2/C_2H_4	CH_4/H_2	C_2H_4/C_2H_6	C_2H_6/C_2H_2	C_2H_2/CH_4
F_{11}-F_{13}	CH_4/TGC[b]	C_2H_4/TGC	C_2H_2/TGC		
F_{14}	$H_2+CH_4+C_2H_4+C_2H_6+C_2H_2$				

[a] all gas values are in [ppm]

[b] (TGC = $CH_4+C_2H_4+C_2H_2$)

In multi-objective feature subset selection, we try to minimse the error of fault classification by selecting the best subset and the optimum number of features. In order to calculate a reliable error estimation, a 5-fold cross validation technique is applied. The details of the cross validation technique can be found in (Polikar, 2006). Generally, the classification ability can be measured by a fitness function which is defined as follows:

$$Fitness = \frac{1}{k}\sum_{i=1}^{k}(\frac{1}{n}\sum_{j=1}^{n}(\widehat{y}_{ij} - y_{ij})^2), \tag{5.1}$$

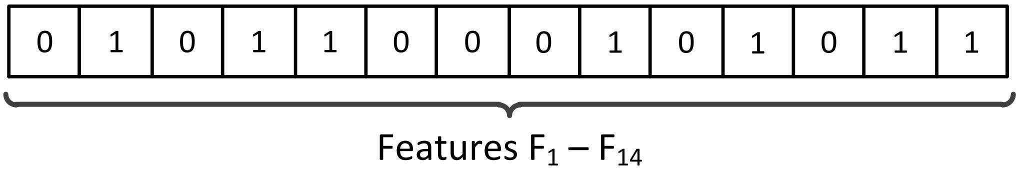


Figure 5.1: An example of a particle for feature subset selection in MOPSO. The 1's represent the corresponding selected features.

where k is the number of folds for cross validation, n is the number of samples, $\widehat{y}$ is the target value for each sample, and y is the binary output of the diagnosing algorithm.

The multi-objective feature subset selection model is

$$\text{Minimize} \quad Err \quad = (\omega_{tr} \cdot Fitness_{tr}) + [(1 - \omega_{tr}) \cdot Fitness_{val}], \tag{5.2}$$

$$\text{Minimize} \quad N_f, \tag{5.3}$$

where ω_{tr} is a weighting factor for training set in cross validation and is set to 0.8 here. It should be noted that to achieve a reliable weighted fitness function, ω_{tr} should be at least equal to 0.63. The term $Fitness_{tr}$ gives the classification error on the training set, $Fitness_{val}$ refers to the classification error on the validating set, and N_f is the number of selected features. Equation 5.2 is applied to minimize the classification error on the selected subset of features. Equation 5.3 is applied to minimize the number of selected features to reduce the complexity of the fault diagnosing system.

5.3.2 Ensemble classifier selection

Thirty different classification algorithms have been used to classify faults of transformers . A list of the classification algorithms used to create a diverse classifier

Table 5.2: The list of the classifiers used in the ensemble.

No.	Classifier	Description
1-6	FFNN	Feedforward neural network classifiers trained using Levenberg-Marquardt, scaled conjugate gradient, and Bayesian regularization optimization algorithm with 10 and 20 hidden layer size.
7-12	CFNN	Cascade-forward neural network classifiers trained using Levenberg-Marquardt, scaled conjugate gradient, and Bayesian regularization optimization algorithm with 10 and 20 hidden layer size.
13, 14, and 15	SVM	Support vector machine classifiers with radial basis, linear and polynomial kernel functions. The kernel scale parameters are selected using a heuristic approach during the training step of classifiers.
16, 17, and 18	FKNN	Fuzzy K-nearest neighbours classifiers trained using 2, 5, and 10 nearest neighbours parameters, respectively.
19	NB	Naive Bayes classifier with standard normal kernel density function and a probability density function.
20-22	KRIDGE	Kernel Ridge Regression classifier with radial basis, polynomial, and linear kernel functions.
23-30	RVFL	Random Vector Functional Link classifier trained using "sigmoid" and "hardlim" activation functions and with Moore-Penrose pseudoinverse and ridge regression for computing of the output weights. Each network is also trained with direct link from input to output layer and with/without bias in the output layer (Zhang and Suganthan, 2016a).

ensemble is given in Table 5.2. Each of these classifiers is trained with the selected input features from the feature selection phase.

In a multi-objective ensemble learning system the best group of classifiers is selected based on two important factors, which are considered as objective functions. One is selecting diverse classifiers and the second factor is accuracy (correct classification rate) which also needs to be taken into account to achieve a more accurate ensemble selection (Ren et al., 2016).

There are two different approaches to measure the diversity of the selected classifiers: pairwise and non-pairwise (Kuncheva and Whitaker, 2003). In this study, a pairwise measure is used which is called Q-statistic and is calculated by Equation 5.4 (Kuncheva and Whitaker, 2003):

$$Q_{ij} = \frac{(tt)(ff) - (tf)(ft)}{(tt)(ff) + (tf)(ft)}, \tag{5.4}$$

where *tt* is the number of correctly classified samples by the pair of classifiers i and j; *ff* is the number of incorrectly classified samples by the pair of classifiers

i and j; *tf* is the portion of dataset correctly classified by the ith classifier and incorrectly classified by the jth classifier; *ft* is the reverse of *tf*. The range of this Q-statistic measure is between -1 and 1. The lower the measure, the better the diversity is. The value of Q-statistic measure is equal to 0 for statistically independent classifiers. It should be noted that the Q-statistic measure is positive if an instance is classified into the same class (Kuncheva and Whitaker, 2003).

To achieve a reliable error estimation and, consequently, an accurate model, a 5-fold cross validation is also applied in the ensemble classifier selection stage. Therefore, all 30 classifiers are trained with the selected non-dominated subset of features corresponding to each fault class using 5-fold cross validation.

The multi-objective ensemble classifier selection model is to:

$$\text{Minimize} \quad Fitness_{tr} = \frac{1}{k}\textstyle\sum_{i=1}^{k}(\frac{1}{n}\sum_{j=1}^{n}(\widehat{y}_{tr} - y_{tr})^2), \tag{5.5}$$

$$\text{Maximize} \quad Div = exp(\frac{2}{L(L-1)}\textstyle\sum_{i=1}^{L-1}\sum_{j=i+1}^{L}Q_{ij}), \quad L \geq 2, \tag{5.6}$$

where k is the number of folds for cross validation, n is the number of training samples, $\widehat{y}_{tr}$ is the target value for each training sample, and y_{tr} is the binary output of the diagnosing algorithm, L is the number of selected classifiers by multi-objective ensemble classifier selection algorithm and Q_{ij} is calculated by Equation 5.4. Equation 5.5 is applied to minimize the classification error on training set. Equation 5.6 (Kuncheva and Whitaker, 2003) is applied to maximize the diversity measure of the selected group of classifiers. Note that the number of selected classifiers (L) should be always greater than 1. The procedure of the proposed algorithm is discussed in section 5.7 in detail.

5.4 Pareto optimality in multi-objective optimization

Pareto optimality (efficiency) is the most important concept in multi-objective optimization (Jin and Sendhoff, 2008). Therefore, it is necessary to introduce Pareto optimality briefly before presenting the multi-objective framework for fault diagnosing of power transformers.

Mathematically, a multi-objective optimization problem with P objectives and N constraints can be formulated as follows (Coello et al., 2004; Deb, 2001; Zitzler and Thiele, 1999):

$$\text{Minimize} \qquad \boldsymbol{H}(\boldsymbol{x}) = [h_1(\boldsymbol{x}), h_2(\boldsymbol{x}), \ldots, h_P(\boldsymbol{x})]^T, \tag{5.7}$$

$$\text{s.t.} \qquad g_n(\boldsymbol{x}) \leq 0, \quad n = 1, 2, \ldots, N, \tag{5.8}$$

where $\mathbf{x}$ is a m-dimensional decision variable vector from a feasible region, $\mathbf{H}(\boldsymbol{x})$ is the vector of P objective functions, and $g_n(\mathbf{x})$ are the N inequality constraints. Objective functions may be any linear or nonlinear function.

In almost all multi-objective optimization problems, multiple objectives are in conflict. To satisfy the contradiction between the objectives, multi-objective optimization problems determine *Pareto optimal* solutions which are called non-dominated solutions (*efficient* solutions). To clarify the concepts of dominance and Pareto optimality, they are mathematically defined for a minimization problem as follows (Deb, 2001; Zitzler and Thiele, 1999):

- *Dominance*: A vector $\mathbf{u} = (u_1, u_2, \ldots, u_L)$ is said to dominate vector $\mathbf{v} = (v_1, v_2, \ldots, v_L)$ (denoted by $\mathbf{u} \preceq \mathbf{v}$) if and only if $\forall i \in \{1, 2, \ldots L\}, u_i \leq v_i \ \wedge \ \exists i \in \{1, 2, \ldots L\} : \ u_i < v_i$.

- *Pareto optimal*: A solution $\boldsymbol{x}^* \in \Theta$ is said to be a Pareto optimal (non-dominate solution) if and only if there is no $\mathbf{x} \in \Theta$ for which $\mathbf{H}(\mathbf{x}) = [h_1(\mathbf{x}), h_2(\mathbf{x}), \ldots, h_P(\mathbf{x})]$ dominates $\mathbf{H}(\mathbf{x}^*) = [h_1(\mathbf{x}^*), h_2(\mathbf{x}^*), \ldots, h_P(\mathbf{x}^*)]$

Figure 5.2 graphically represented the Pareto solutions for an arbitrary multi-objective optimization problem which belongs to two different Pareto sets (1 and 2). Solution P$_{23}$ is not dominated by any other members of both Pareto sets. Thus, P$_{11}$, P$_{12}$, P$_{23}$, and P$_{13}$ are the Pereto optimal solutions (non-dominated solution) on the Pareto front.

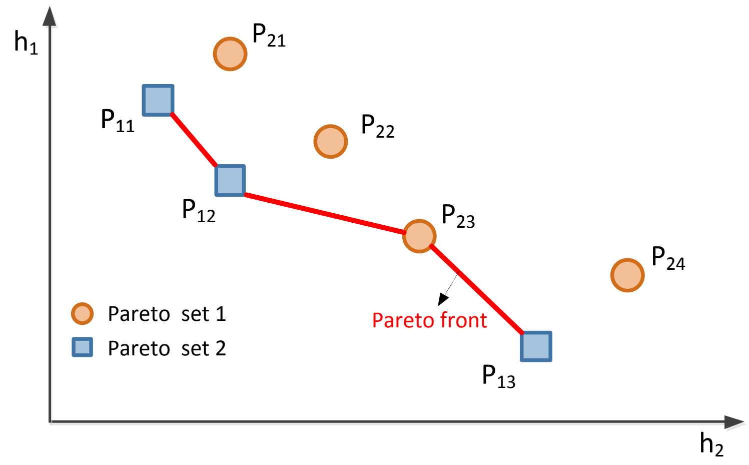


Figure 5.2: An example of a Pareto front.

5.5 Multi-objective PSO optimization

To implement a multi-objective feature subset selection and ensemble classifier selection, a MOPSO algorithm is used in the proposed power transformer fault diagnosis system. Understanding MOPSO requires some background about the PSO algorithm. In the PSO algorithm, each particle represents a possible solution for an optimization problem and every movement of the particles towards a new position within a defined space could be a new solution. In each iteration the PSO algorithm is updated, based on 3 rules: 1) continue in the same direction of the latest movement (inertia term); 2) move towards the best personal solution (nostalgia term); and 3) move towards the best solution which has been found so far by all the particles (global best). These three rules used for updating the position of the particles in PSO are formulated as follows (Kennedy, 2011):

$$\mathbf{v}_i\left(k\right) = \overbrace{\omega \cdot \mathbf{v}_i\left(k-1\right)}^{\text{inertia term}} + \overbrace{c_1 \cdot r_1 \cdot \left(\mathbf{x}_{pbest_i} - \mathbf{x}_i(k)\right)}^{\text{nostalgia term}} + \overbrace{c_2 \cdot r_2 \cdot \left(\mathbf{x}_{gbest_i} - \mathbf{x}_i(k)\right)}^{\text{global term}}, \tag{5.9}$$

$$\mathbf{x}_i\left(k\right) = \mathbf{x}_i\left(k-1\right) + \mathbf{v}_i\left(k\right), \tag{5.10}$$

where ω is the inertia weight, $\mathbf{x}_{pbest_i}$ is the best position that particle $\mathbf{x}_i$ has experienced so far, $\mathbf{x}_{gbest_i}$ is the position of the best particle among the swarm, r_1 and r_2 are two uniformly distributed random numbers in the range of [0,1], and c_1 and c_2 are the learning factors to control the effect of nostalgia and global terms, respectively. In this study $c_1 = c_2 = 2$ and $\omega = 0.8$ (Shi et al., 2001).

The general pseudo-code for PSO can be described as follows:

- Initialise the position and the velocity of the swarm.
- Select the best particle (x_{gbest}) among the swarm as leader.
- Repeat the following steps while the terminate criteria has not been reached.
  - Update velocity (Equation 5.9).
  - Update position (Equation 5.10).
  - Find new $\mathbf{x}_{pbest}$ for each particle.
  - Find new $\mathbf{x}_{gbest}$ (leader).
  - Evaluate fitness function.
- Report the best found particle as an optimum solution for the problem.

The main difference between PSO and MOPSO algorithms is the *global best* concept. However instead of using *global best*, in the MOPSO algorithm, a repository ("hall of fame") (Engelbrecht, 2007) is defined, which stores an archive of the non-dominated solutions. The repository also approximates the best Pareto optimal (Alvarez-Benitez et al., 2005). So, unlike the PSO algorithm, the *global best* is not unique and particles can select members from the repository as a leader without any preferences, since they are all non-dominated solutions. Although this approach is simple to implement, it may decrease the convergence rate of the algorithm. To tackle this problem, a region-based selection system (Coello and Lechuga, 2002; Coello et al., 2004; Knowles and Corne, 2000) is used which divides the search space into several subregions. Then, the least number of non-dominated solutions in a subregion, the more likely the *global best* is selected from that subregion. This selection approach helps to increase the diversity in selecting non-dominated solutions as a *global best*. The region-based selection is performed in the following way:

$$n_i \leq n_j \Leftrightarrow P_i \geq P_j, \tag{5.11}$$

$$s.t. \quad P_k = \frac{\exp(-\beta n_k)}{\sum\limits_k (\exp(-\beta n_k))}, \quad k = 1, 2, \ldots, K \tag{5.12}$$

where n_i and n_j are the number of non-dominated solutions in the ith and jth subregion, respectively, P_i and P_j are the selection probabilities of the ith and jth subregion, K is the number of subregions, and β is called the selection pressure parameter. The larger the β, the higher the diversity of selecting the leader (*global best*) is. Note that if there is more than one non-dominated solution in the selected subregion, one of them is randomly selected as the *global best*.

The fast speed of convergence is one of the main advantages of the PSO algorithm. So, in order to avoid a premature convergence and, consequently, selecting a false Pareto optimal, a mutation operator was implemented that has been described in (Coello et al., 2004) in detail.

The pseudo-code of the MOPSO algorithm used in this research is shown in Algorithm 1. First, each particles position and velocity are randomly initialised. The first Pareto optimal set is created from the non-dominated particles. Then, each particle selects a leader (*global best*) using region-based selection and the position and velocity of each particle are updated using Equation 5.10 and Equation 5.9. In addition, the mutation operator is applied. After the mutation, the *pbest* of each particle is checked whether it is dominated by the mutated or new particles. The non-dominated particles are added to the Pareto optimal set. To avoid exceeding the predetermined size of the repository (Pareto optimal set), only non-dominated leaders are kept. Obviously, the size of subregions is updated, too. In line 1 of the MOPSO algorithm (Algorithm 1) some parameters need to be set. These are as follows:

- *nPop*; population size,

- *MaxIt*; maximum number of iteration,
- *nRep*; repository size,
- μ; mutation rate,
- β; leader selection pressure,
- *nRegion*; number of subregions,
- c_1 and c_2; learning factor (coefficient),
- ω; inertia weight.

Algorithm 1: MOPSO algorithm.

```
Set the values of MOPSO parameters
Initialize the position and velocity of the swarm
Evaluate objective values on initialized particles
Select non-dominated solutions as leader gbest
for $it \leftarrow 0$ to $MaxIt$ do
    for $n \leftarrow 0$ to $nPop$ do
        Select a leader for particle $n$
        Update the velocity and position of particle $n$
        Apply mutation on particles' position
    end
    Evaluate objective values
    Add non-dominated particles to the repository
    Determine domination of new repository members
    Keep only non-dominated members in the repository
    Remove members from occupied sub-region if repository is full (nRep)
end
Report Pareto optimal set (non-dominated solutions)
```

5.6 Dempster-Shafer combination rule

In order to determine the degree of certainty of the proposed fault classification method, the outputs of the classifier ensemble need to be combined. The best approach is to achieve a single probability value that shows how likely the fault occurs inside the transformers.

The Dempster-Shafer theory (DST) is a powerful method for combining information from different sources, which is an extension of Bayesian inference (Shafer et al., 1976). One advantage of this method is the capability of capturing and combining whatever certainty exists in the information sources (Klein, 2004). An overview of DST is briefly given here.

There are three main functions in DST: a mass probability function (m), a belief function (Bel), and a plausibility function (Pl). The mass probability function is the most important function in the DST as the rule of combination, which meets the following conditions (Klein, 2004):

$$\begin{aligned} m : 2^X \rightarrow [0,1]\,, \\ m\left(\emptyset\right) = 0, \\ \sum_{A \subseteq X} m(A) = 1, \end{aligned} \tag{5.13}$$

where X is the universal set and $\emptyset$ is the empty set. For this application, the universal (X) is X = {No fault, partial discharge, energy discharge, overheating fault}.

DST is able to combine independent evidences (mass probability functions), m_1 and m_2, to produce more informative evidence, which is shown by $m_1 \bigoplus m_2$ and is calculated as follows:

$$(m_1 \bigoplus m_2)(A) = \frac{1}{1-K} \sum_{B \cap C = A \neq \emptyset} m_1(B) \cdot m_2(C),$$

$$\text{where} \qquad K = \sum_{B \cap C = \emptyset} m_1(B) m_2(C). \tag{5.14}$$

It should be noted that the outputs of the classification algorithms used in the 2-ADOPT algorithm (described in the following sections) are types of normalised mass functions between 0 and 1, so it is possible to consider them as mass probability functions.

5.7 Two phase MOPSO transformer fault diagnosis framework

Figure 5.3 shows the flowchart of the proposed fault classification method, which consists of two main phases (feature subset selection and ensemble classifier selection). The proposed framework utilizes the advantages of a MOPSO algorithm to select the best subset of features in the first phase. The non-dominated solutions of the first phase, which are the best selected features corresponding to each fault class, are considered as inputs to train the classifiers in the second phase. Then, in the second phase of 2-ADOPT, the MOPSO algorithm is used again to select the most accurate and diverse group of classifiers.

2-ADOPT algorithm is described in ten main steps as follows:

1. *Normalization*: All input features are first normalized to zero mean and unit standard deviation.

2. *Separate the testing set from a non-testing set*: The DGA dataset is randomly divided into two sets; a non-testing dataset to train and validate the model, and a testing dataset to test the proposed model.

3. *Create a synthetic dataset*: Since transformers are well-maintained during their operation the fault rate of these assets is generally low. Thus, labelled data for training the classification algorithms are not sufficient for some classes. So, the probability of biased classification using this imbalanced dataset increases, which in turns leads to a higher error rate on the minority fault classes (He et al., 2008). To tackle an imbalanced dataset, adaptive synthetic over-sampling technique (ADASYN) is applied to enable the classification algorithms to achieve their desirable performance He et al. (2008). The ADASYN algorithm comprises three major steps: i) compute the degree of class imbalance to calculate the number of synthetic samples for the minority class; ii) calculate Euclidean distance to find the K nearest neighbours in a minority class; and iii) generate the synthetic dataset for the minority class by Equation 5.15.

$$d_i = x_i + (x_{ki} - x_i) \times \lambda, \tag{5.15}$$

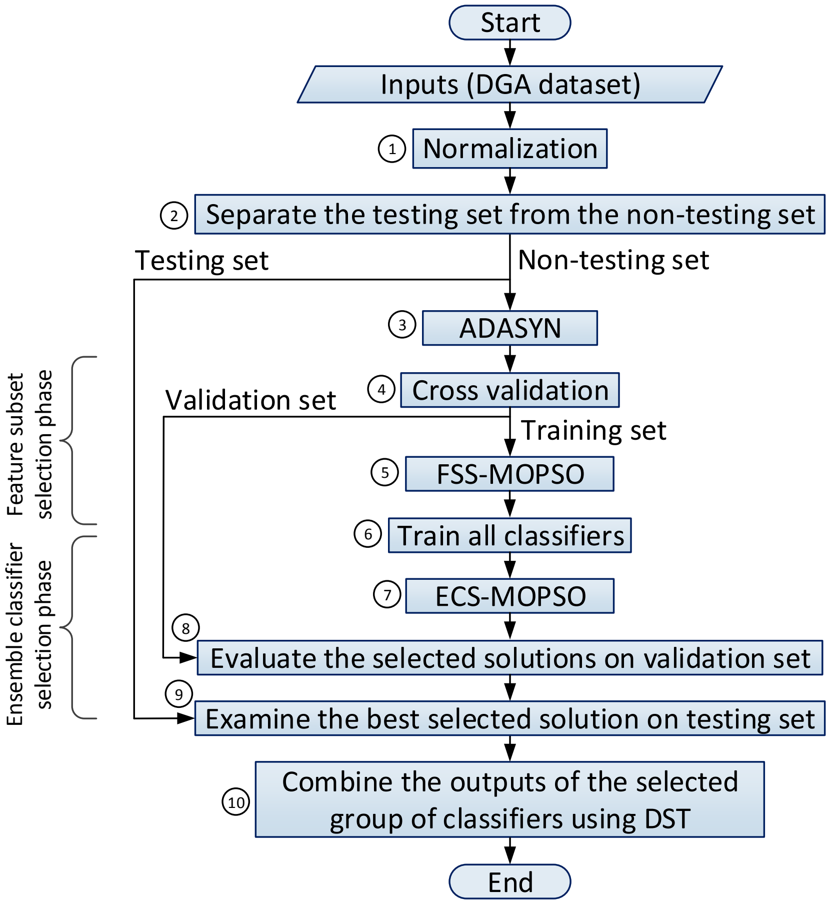

Start
Inputs (DGA dataset)
1 Normalization
2 Separate the testing set from the non-testing set
Testing set
Non-testing set
3 ADASYN
4 Cross validation
Validation set
Training set
5 FSS-MOPSO
6 Train all classifiers
7 ECS-MOPSO
8 Evaluate the selected solutions on validation set
9 Examine the best selected solution on testing set
10 Combine the outputs of the selected group of classifiers using DST
End
Feature subset selection phase
Ensemble classifier selection phase

Figure 5.3: Flowchart of 2-ADOPT.

where x_i is a minority sample, x_{ki} is a randomly chosen sample from the determined K nearest neighbors, and λ is a random number in the range of $[0,\ 1]$.

4. *Cross validation*: A 5-fold cross validation is performed to estimate a reliable error for the model.

5. *Using MOPSO to find non-dominated solutions for the first phase (FSS-MOPSO)*: In this step, as described in section 5.3.1 a multi-objective feature subset selection is applied using MOPSO.

6. *Train all classifiers using selected features*: From this step, the second phase of the algorithm starts. In each iteration, all the classifiers are trained with the selected non-dominated feature vectors on the Pareto optimal set (repository) of the first phase. Then, the MOPSO algorithm is called to select the most accurate and diverse group of classifiers. This procedure is repeated for all non-dominated selected input feature vectors.

7. *Use MOPSO to select the best group of classifiers (ECS-MOPSO)*: Following the multi-objective ensemble classifier selection approach in Section 5.3.2, the use of MOPSO enables us to find the most accurate and diverse group of classifiers.

8. *Evaluate the best selected solutions on the validation set*: In this step, all non-dominated solutions on the Pareto optimal set for the ensemble classifier selection phase are tested on the validation set to rank them. Then, the non-dominated solution with the highest performance within the validation set is selected.

9. *Examine the best solution within the testing set*: So far, a group of the best classifiers have been selected. In this step, the test set is provided to each selected classifier to make predictions. The outputs of the classifiers on the test samples are actually assigned probabilities corresponding to each fault class. As an example, assume two classifiers have been selected and they have assigned probabilities of 0.9 and 0.85 to the test sample number one to be a NF class, respectively. Now, these two assigned probabilities should be combined to make a final prediction on this test sample (# 1) using a combination method, which is explained in the next step.

10. *Combine the outputs of the classifiers*: The DST is used to combine the assigned probabilities to each test sample as described in Section 5.6. For instance, in the example of Step 9, the two assigned probabilities to the test sample number one are combined using Equation 5.14. Therefore, the final prediction of the 2-ADOPT algorithm for this test sample is NF class with the probability of 0.981. This procedure is repeated to compute the prediction probabilities of all samples in the test set.

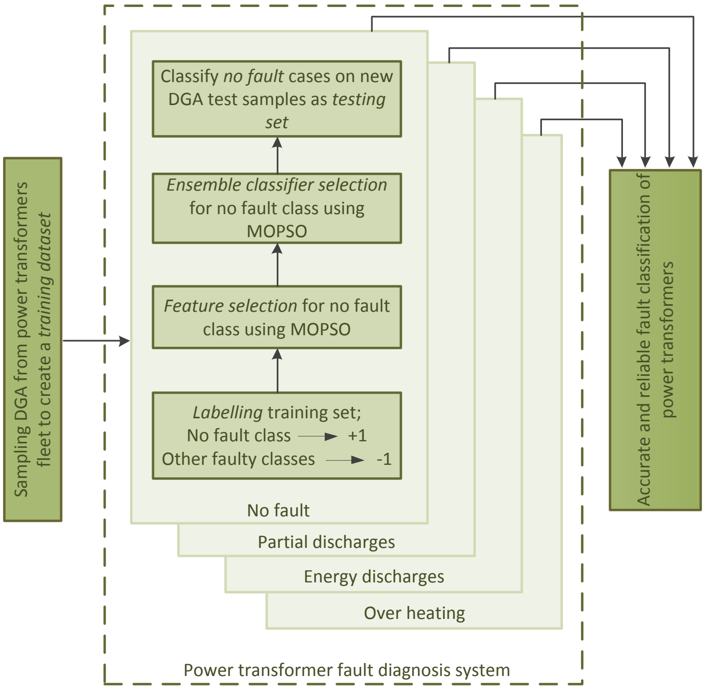

Sampling DGA from power transformers fleet to create a training dataset
Classify no fault cases on new DGA test samples as testing set
Ensemble classifier selection for no fault class using MOPSO
Feature selection for no fault class using MOPSO
Labelling training set;
No fault class → +1
Other faulty classes → -1
No fault
Partial discharges
Energy discharges
Over heating
Power transformer fault diagnosis system
Accurate and reliable fault classification of power transformers

Figure 5.4: Schematic diagram of a generic power transformer fault diagnosis system.

The proposed method is applied for all four fault classes as illustrated in Figure 5.4. Performing the two phase multi-objective fault diagnosis method for each class separately results in selecting the best subset of features and the most accurate and diverse classifiers corresponding to each fault class.

5.8 Binary classification performance metrics

One of the key issues in evaluating the performance of a classification approach is the capability of correct classification of new examples. The classification performance of two class problems, as in this case, can be interpreted in a confusion matrix as shown in Table 5.3.

The most commonly used measure to evaluate the performance of a classifier is

Table 5.3: Confusion matrix for a two class problem.

	Predicted positive	Predicted negative
Actual positive	True positive (TP)	False negative (FN)
Actual negative	False positive (FP)	True negative (TN)

accuracy rate as given in Equation 5.16.

$$Acc = \frac{TP + TN}{TP + FN + FP + TN}. \tag{5.16}$$

Other measures that can be derived from Table 5.3 to evaluate the performance of classification algorithms are listed as follows (Baldi et al., 2000):

- True positive rate or recall: $\mathrm{TPR} = \frac{\mathrm{TP}}{\mathrm{TP+FN}}$.
- False positive rate: $\mathrm{FPR} = \frac{\mathrm{FP}}{\mathrm{FP+TN}}$.
- Positive predictive value (Precision): $\mathrm{PPV} = \frac{\mathrm{TP}}{\mathrm{TP+FP}}$.

In addition, two other important metrics for evaluating binary classification can be deduced from Table 5.3 which are called F-score and Matthews correlation coefficient.

- F-score (Baldi et al., 2000): As a weighted harmonic mean of recall and precision, F-score (F-measure) considers both the precision and recall measures to analyse the accuracy of a binary classification (Equation 5.17).

$$F = \left(1 + \beta^2\right) \cdot \frac{precision \cdot recall}{(\beta^2 \cdot precision) + recall}. \tag{5.17}$$

when β is equal to 1 the measure is called balanced F-score (F_1 score) which is the harmonic mean of precision and recall and takes both precision and recall into account equally.

- Matthews correlation coefficient (MCC) (Baldi et al., 2000): MCC can be used as a quantitative measure of the quality of a binary classification. In statistics, it is also known as the phi coefficient. Actually, this measure interprets the correlation between the target and prediction in a two class classification. The value of MCC is between -1 and +1 in which +1 shows the highest classification ability and -1 represents the lowest classification ability or total conflict between prediction and target. MCC can be formulated by Equation 5.18:

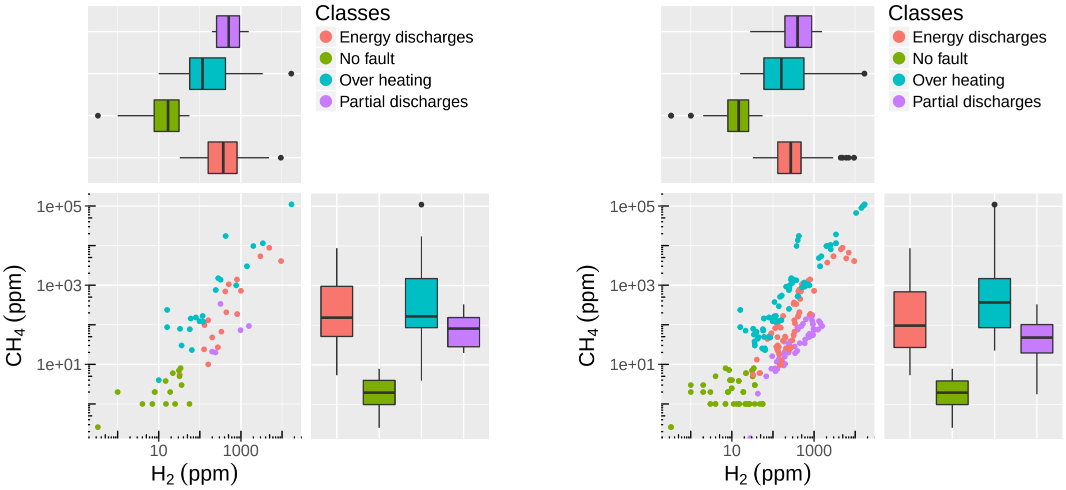
$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}}. \tag{5.18}$$

- Receiver operating characteristics (ROC) curve (Fawcett, 2006): ROC curve is used as a qualitative and quantitative evaluation measure. It shows the true positive rate (TPR) versus false positive rate (FPR) regarding different threshold settings (cutoff points) which graphically visualises the trade-off between TPR and FPR. In fact, the ROC curve tries to plot the cumulative distribution function of known probability distribution for both true and false detected cases in the y-axis against x-axis, respectively. Furthermore, one can evaluate the classification ability by calculating the area under the curve (AUC) as a scalar measure. The higher the value of the area under each curve, the better the classification is.

The performance of the proposed fault classification method was evaluated using the above mentioned metrics, then the proposed method is compared with three other ensemble fault diagnosis techniques reported in section 5.9.2.

5.9 Experimental validation

In this study, the imbalanced DGA dataset is composed of 101 samples from confirmed field data for transformers which are given in (Ganyun et al., 2005; Gao et al., 1998; Sarma and Kalyani, 2004; Vanegas et al., 1997; Zhang et al., 1996). The number of instances for the four classes are presented in Table 5.4. These four classes are typically used by electrical utilities to assess the condition of their transformers fleet based on DGA. After creating synthetic data, the number



(a) (b)

Figure 5.5: Scatter plots of DGA dataset for two arbitrary features in logarithmic scale. (a) Imbalanced. (b) Synthetically balanced.

of cases for all classes were distributed equally (Table 5.4). Distribution of the imbalanced and balanced DGA dataset for two arbitrary features (H_2 and CH_4) are plotted in Figure 5.5a and Figure 5.5b, respectively. This shows how synthetic data are distributed regarding two dissolved gases (H_2 and CH_4). In addition, Figure 6.2 shows to what extent the classification problem would be challenging, if, like conventional techniques and standards, one only used prespecified features to classify power transformer faults.

Table 5.4: Number of DGA samples for balanced and imbalanced datasets.

Fault classes	No fault	Over heating	Partial discharges	Energy discharges
Number of cases (imbalanced)	56	21	6	18
Number of cases (balanced)	71	70	69	72

As mentioned in Section 5.3, there are 14 different features. In the Multi-objective feature subset selection phase, we find non-dominated feature vectors corresponding to each fault class, which can be used as input features to train the classifiers. For instance, the fault diagnosis procedure of energy discharge (ED) class is described in the following section.

5.9.1 Verifying the performance of 2-ADOPT algorithm for diagnosing ED fault class

The five non-dominated solutions (S_1, S_2, S_3, S_4, and S_5) for the feature subset selection phase are presented in Figure 5.6a. Each of these five solutions is a vector of the best features to classify energy discharge fault class with a high accuracy on testing set. In the second phase, firstly, each of the five selected feature vectors are used to train the classifiers, then MOPSO selects a group of the most accurate and diverse classifiers. In Table 5.5 the fault classification error using 5-fold cross validation for each of the five non-dominated feature vectors (S_1, S_2, S_3, S_4, and S_5) on the validation set is given.

Table 5.5: Four non-dominated feature vectors and their corresponding error on the validation set.

Non-dominated feature vectors	Error on the validation set
S_1	0.1015
S_2	0.0782
S_3	**0.0451**
S_4	0.093
S_5	0.1508

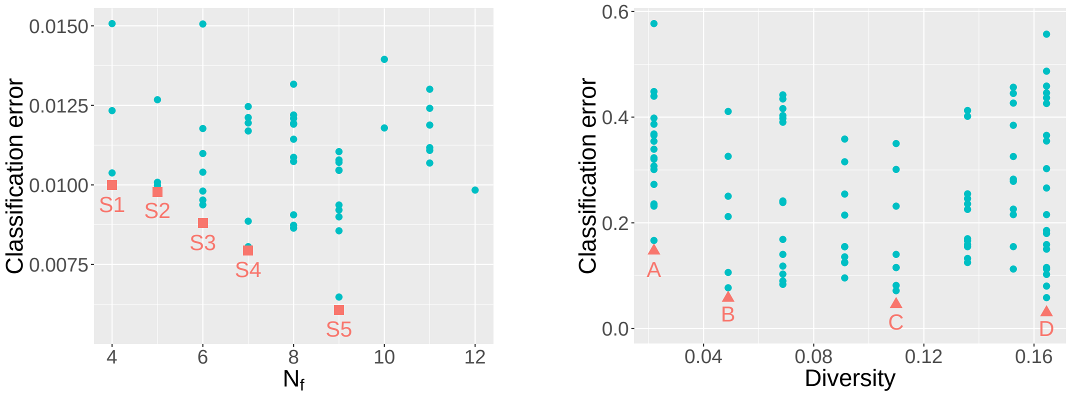
Here, the ensembles created using S_3 as a feature vector resulted in better fault diagnosing performance on the validation set. Therefore, solution S_3 was selected as the best feature vector for classifying ED fault class. Table 5.6 lists the best features corresponding to each fault class. Next, we need to choose one of the solutions on Figure 5.6b, i.e., Pareto front (A, B, C, and D) as the best ensemble. To do this, all four solutions were evaluated on the validation set and, as reported in Table 5.7, solution C had a better classification performance on the validation set. As a result solution C was nominated to diagnose ED faults. The selected

Table 5.6: Selected features for each fault class.

Fault class	Selected feature vector
No fault	F_1, F_2, F_3, F_4, F_6, F_7, F_9, F_{12}
Partial discharges	F_1, F_4, F_5, F_6, F_{12}
Energy discharges	F_2, F_4, F_5, F_6, F_{10}, F_{11}
Over heating	F_1, F_2, F_3, F_4, F_5, F_7, F_8, F_{11}

Table 5.7: Three non-dominated solutions and their corresponding error on the validation set for ED fault class.

Solutions	Selected classifiers #	Error
A	2, 3, 4, 5, 7, 10, 13, 17, 19, 20, 22, 27, 28, 29	0.266
B	4, 5, 6, 8, 10, 11, 12, 15, 18, 24, 29, 30	0.0922
C	**1, 4, 6, 11, 12, 13, 14, 15, 17, 19, 21, 27, 28, 31**	**0.063**
D	3, 4, 8, 9, 11, 12, 13, 18, 19, 20, 25, 28, 29	0.078



(a)

(b)

Figure 5.6: Pareto optimal set for (a) feature selection phase and (b) ensemble classifier selection phase. Solutions in red represent the non-dominated solutions.

classifiers corresponding to the four non-dominated ensembles are also given in Table 5.7. Clearly, the highest classification accuracy on the validation set belongs to solution C. Although, for the training set (Figure 5.6b) the accuracy of solutions D is higher than C, solution C performs better on the validation set. In addition, computationally, there is no preference between solution C and D because both use approximately equal number of classifiers. Also, the value of diversity measure (Q-statistics) for solution C is equal to 0.11 which represents a diverse selected group of classifiers.

5.9.2 Comparison with other ensemble approaches

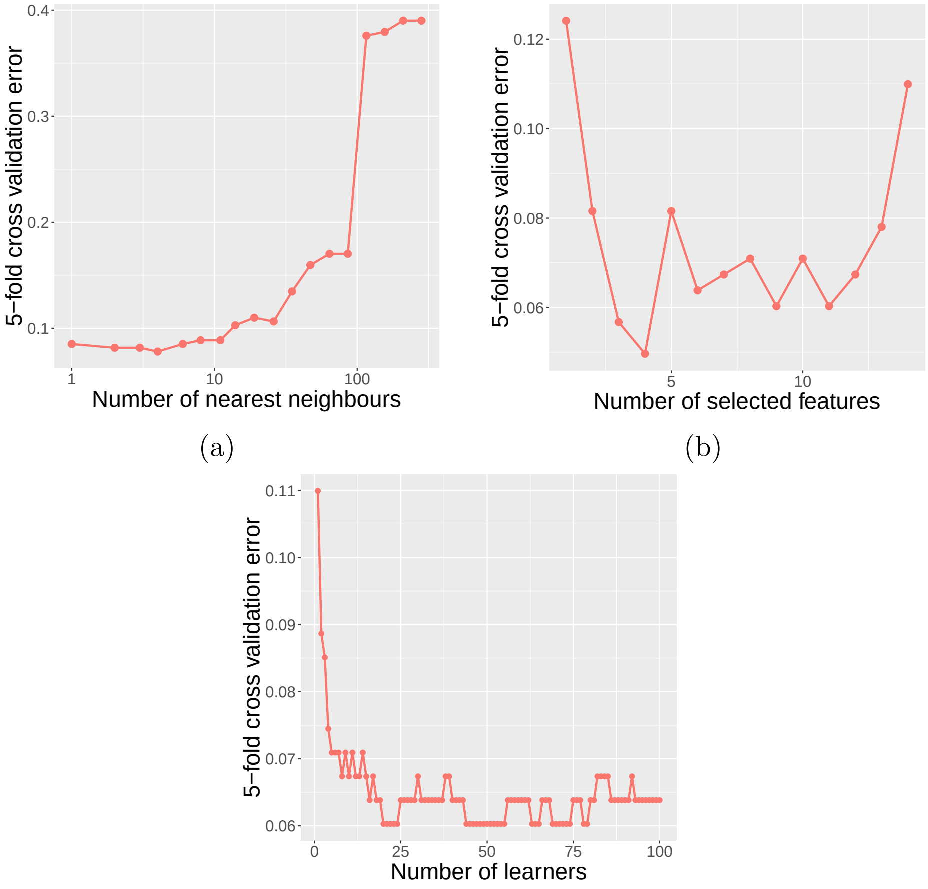
The performance of the proposed method (2-ADOPT) was compared with that of multi-objective ensemble classifier selection using MOPSO without feature subset selection phase (MOECS) and four other common ensemble learning techniques:

random forests with KNN and axis align decision tree as the weak learners, AdaBoost, and oblique random forests (Breiman, 2001; Friedman et al., 2001; Menze et al., 2011; Zhang and Suganthan, 2015).

In the case of MOECS the same type of classifiers as utilized in 2-ADOPT and listed in Table 5.2 are trained, and MOPSO selects the most accurate and diverse group of them to classify the fault of the transformers. The main difference between MOECS and 2-ADOPT is training the classifiers with all fourteen features (Table 5.1), instead of applying a feature subset selection phase.

In the case of K-NN random forests (KNN-RF), the input feature vector also consisted of all fourteen features. Here, the K-nearest neighbour (K-NN) (Friedman et al., 2001) and random subspace (Ho, 1998) are used as the classifier and ensemble algorithm, respectively. A 5-fold cross validation is also applied to find the optimum number of nearest neighbours, input features, and classifiers in the ensemble. Figure 5.7a shows the number of nearest neighbours against the estimated classification error using 5-fold cross validation. The minimum cross validation error has been achieved with four nearest neighbors (NNs). Furthermore, ensembles were created for 4-NN classifiers with a different number of features, to find the desired number of features as represented in Figure 5.7b. Clearly, the ensembles that use four features result in the lowest cross validation error, which is equal to 0.05. Finally, the optimum number of classifiers in an ensemble using 4-NN and four predictors, which lead to the lowest cross validation error, was investigated. Figure 5.7c confirms that it is possible to have good classification accuracy with 50 classifiers. Therefore, the final ensemble was constructed using the optimum parameters: 4-NN, four selected features, and 50 classifiers.

For axis align-aligned RF (AA-RF), a 5-fold cross validation is also applied to find the optimum number of trees and maximum number splits. In Figure 5.8, each plot shows the 5-fold cross validation errors for versus number of tree for various tree complexity levels (MaxNumSplits). For example, as shown in Figure 5.8, the model with 22 trees and *MaxNumSplits* equals 9 results in minimum cross validation error for NF class. Therefore, the final ensemble for NF class was constructed using the following parameters: 22 trees and maximum 9 splits. This procedure were repeated for all other classes.



(a) (b)

(c)

Figure 5.7: Adjusting the optimum numbers of (a) nearest neighbours, (b) features (predictors), and (c) learners for random forests ensemble method using 5-fold cross validation error.

On the other hand, the oblique random forests (ORF) method was implemented in R (R Core Team, 2013) using the *obliqueRF* package (Menze and Splitthoff, 2012). The following parameters were used for training the oblique random forests algorithm (Menze et al., 2011):

- Number of selected features used in each node of decision trees = $max(sqrt(\#offeatures), 2)$,
- Training method = *ridge regression*,
- Number of trees in the ensemble = 50.

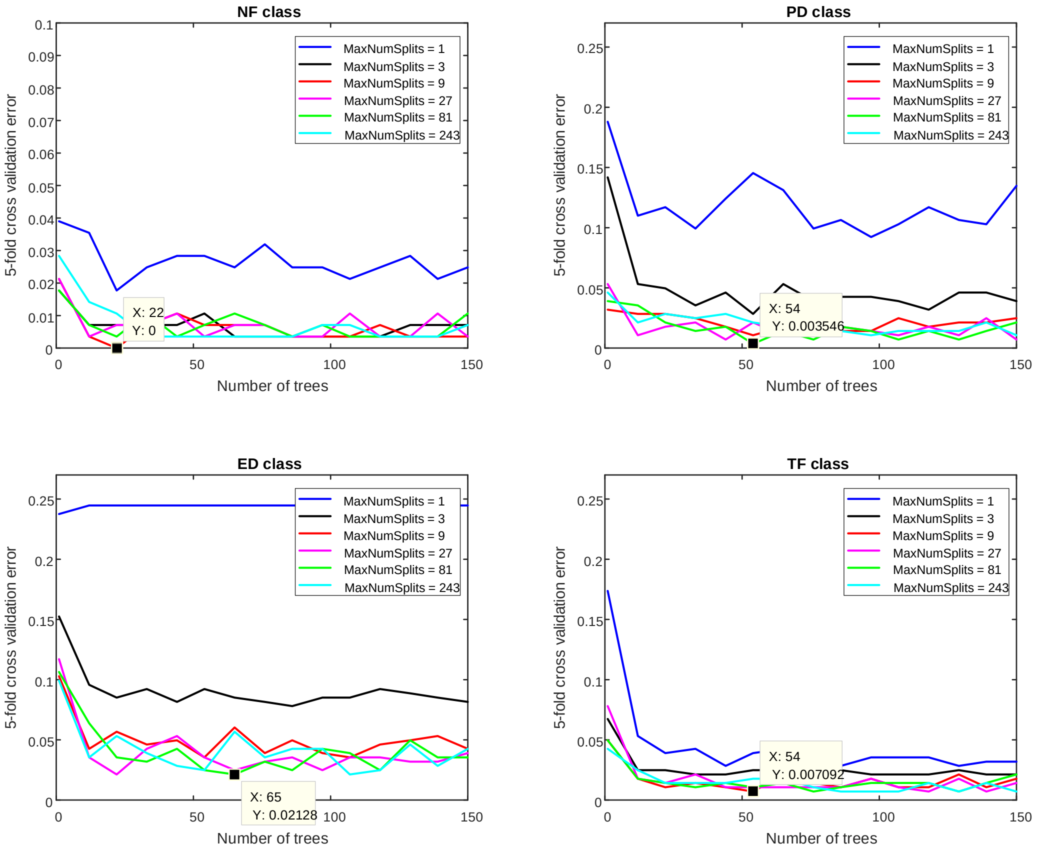


Figure 5.8: Adjusting the optimum parameters of axis aligned RF for four different fault classes.

A 5-fold cross validation is also used to estimate the classification accuracy of the oblique random forests.

For finding the optimum number of learning cycles in AdaBoost algorithm, a 5-fold cross validation is used. Figure 5.9 represents the optimum number of learning cycles corresponding to each fault class. As an example, the lowest cross validation error for PD fault class is obtained with 76 learning cycles.

The classification ability of these six methods (2-ADOPT, MOECS, KNN-RF, AA-RF, ORF, and AdaBoost) are evaluated using the measures listed in section 5.8. The overall classification accuracies, F_1 score, and MCC measures of these three methods for diagnosing faults of transformers are given in Table 5.8.

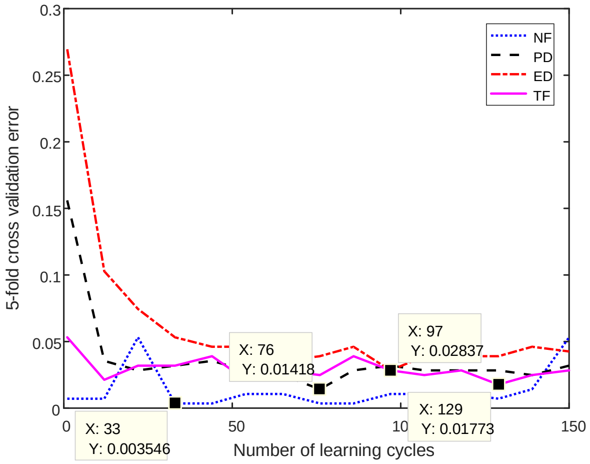


Figure 5.9: Adjusting the optimum learning cycles of AdaBoost for four different fault classes.

Table 5.8: Comparison of the classification measures at 0.9 cutoff point among 2-ADOPT, MOECS, KNN-RF, AA-RF, ORF and AdaBoost.

Method	Accuracy rate (%)				F_1 score				MCC			
	NF	PD	ED	OHF	NF	PD	ED	OHF	NF	PD	ED	OHF
2-ADOPT	**100**	**97**	**94**	**100**	**1**	**0.92**	**0.81**	**1**	**1**	**0.97**	**0.8**	**0.99**
MOECS	94	88	88	94	0.95	0.8	0.61	0.93	0.85	0.62	0.6	0.91
KNN-RF	91	91	85	79	0.91	0	0.43	0.43	0.85	0	0.52	0.25
AA-RF	**100**	94	**94**	**100**	**1**	0.9	**0.81**	**1**	**1**	0.97	**0.8**	0.91
ORF	97	94	85	85	0.96	0.2	0.43	0.625	**1**	0.1	0.5	0.59
AdaBoost	**100**	91	91	**100**	**1**	0.33	0.88	**1**	0.96	0.9	0.5	0.91

KNN-RF and ORF show almost random behaviour for classifying the PD fault class with the F_1 score equal to 0 and 0.2, respectively, and a poor classification for the ED fault class, with the F_1 score equal 0.43. However, the 2-ADOPT algorithm is capable of boosting the F_1 score for both classes to 0.92 and 0.81, respectively. On the other hand, AA-Rf algorithm is comparable with 2-ADOPT algorithms which is only outperformed for PD fault class. This can also be concluded from MCC values, where the values corresponding to, the 2-ADOPT algorithm are closer to 1. The performance of AdaBoost algorithm is also comparable with 2-ADOPT and AA-RF algorithms on NF and TF fault classes. In addition, the accuracy rate, F_1 scores, and MCC in different cutoffs are compared in Figure 5.10. The

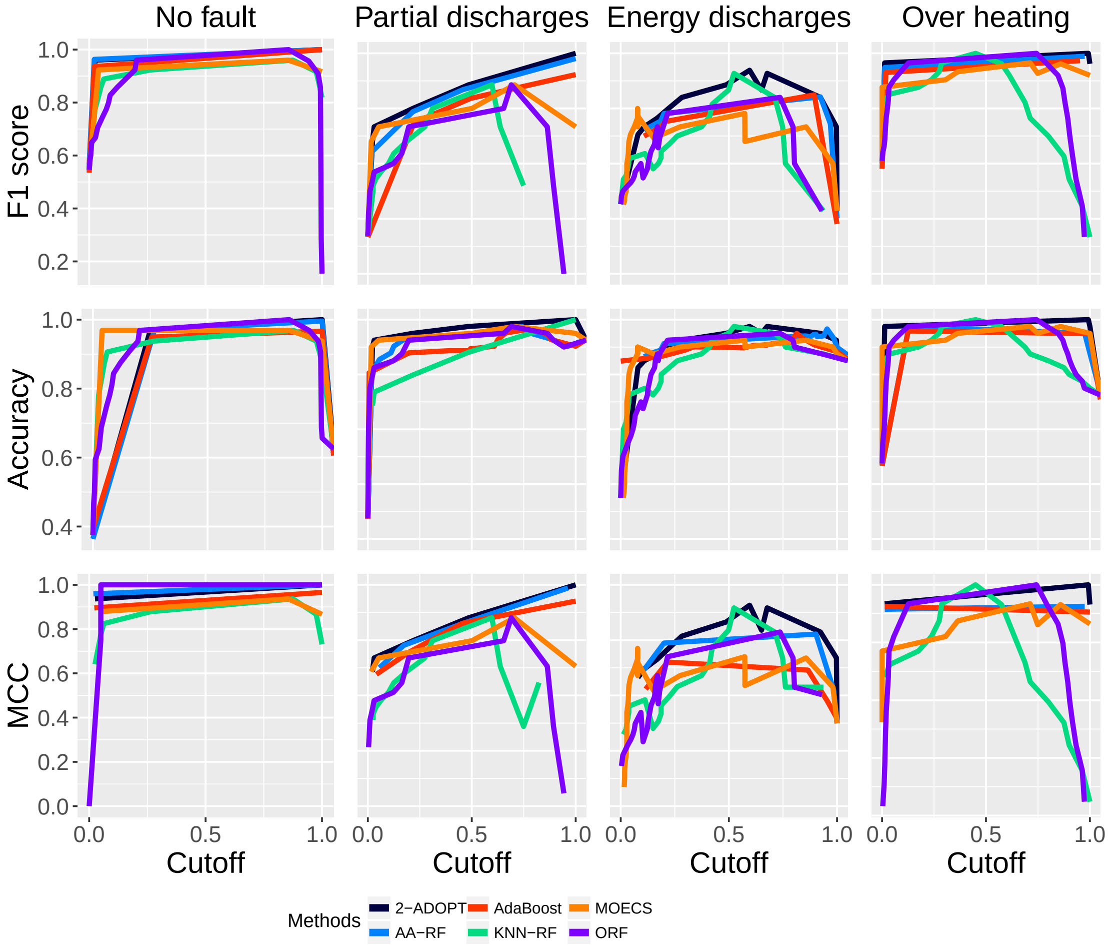


Figure 5.10: Comparison of classification accuracy, F_1 score, and MCC measure at different cutoff points.

graphs confirm that the 2-ADOPT algorithm has better classification capability in almost all cutoffs.

The computed class probabilities of diagnosed faults on the testing set are compared in Figure 5.11. Here, diagnosing probabilities can be considered as certainty measures for the diagnosed faults of transformers; the black dashed threshold line indicates the 0.9 cutoff point. Considering this, the threshold point led to 8, 19, 4, 15, and 21 misclassification cases for MOECS, KNN-RF, AA-RF, ORF, and AdaBoost respectively, while there are only two false negative (FN) diagnoses for

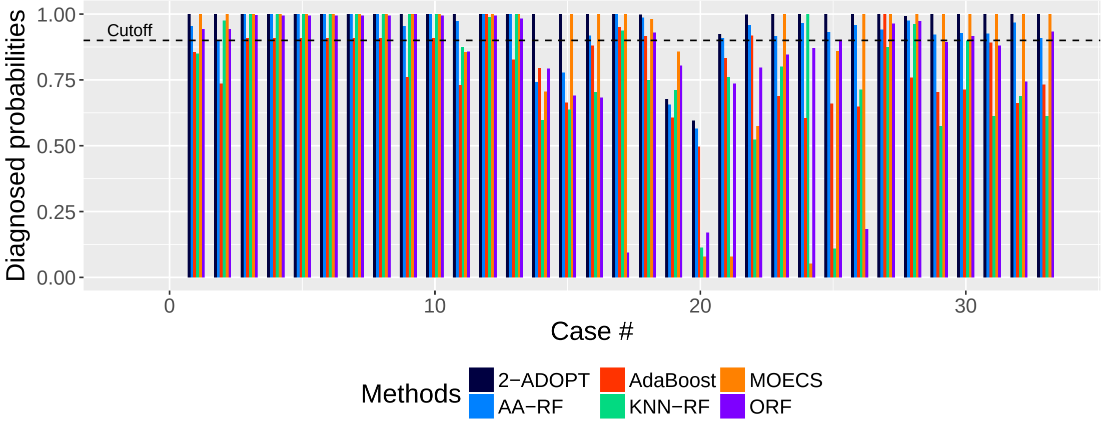


Figure 5.11: Comparison of diagnosing probabilities on the testing set.

2-ADOPT (case 19 and 20). In addition, the results of diagnosed probabilities of the proposed algorithm (2-ADOPT) on the testing set for all four fault classes are given in Table 5.9. At 0.9 cutoff point, there are two false negative (FN) cases for energy discharges fault class and one true negative (TN) case for partial discharges fault class, which are shown in bold in Table 5.9. Overall, 2-ADOPT performs very well on no fault and over heating cases, while there are minor uncertainties when diagnosing the other two fault classes.

The four aforementioned fault diagnosis algorithms were evaluated on a Windows 8 PC with Intel Core i7 CPU and 8GB RAM. The CPU processing time of these algorithms are compared in Table 5.10. Although the processing time of the proposed method is longer than the other diagnosing algorithms, the accuracy of the diagnosing algorithm is much more important than its speed for utilities and power companies, especially when it comes to assess one of the most critical assets such as power transformers.

To compare the results of the proposed method with the previously developed intelligent fault diagnosing algorithms, the results of some of these algorithms are reported in Table 5.11. Since the DGA dataset used in these studies are not available to the public, there is no opportunity for benchmarking here. However, the proposed method in this chapter can be compared, in terms of number of fault classes and the overall accuracy of the diagnosing algorithm. Generally speaking,

Table 5.9: The actual value of 33 dissolved gas samples as testing set and their corresponding diagnosed probabilities.

No.	Actual fault	Dissolved gases [ppm]					Diagnosed probabilities				Predicted fault
		H_2	CH_4	C_2H_4	C_2H_6	C_2H_2	NF	PD	ED	OHF	
1	NF	14.7	3.8	10.5	2.7	0.2	1	0	0	0	NF
2	NF	8.5	7.2	4.3	3.9	3.51	1	0	0	0	NF
3	NF	22	0	0	14	0	1	0	0	0	NF
4	NF	3	1	1	2	1	1	0	0	0	NF
5	NF	27	30	2.4	23	0.1	1	0	0	0	NF
6	NF	0	19.3	0	57.2	0	1	0	0	0	NF
7	NF	9	4	0	11	0	1	0	0	0	NF
8	NF	561	389	365	238	273	1	0	0	0	NF
9	NF	2501	1428	4983	4622	6996	1	0	0	0	NF
10	NF	5	21	63	19	0	1	0	0	0	NF
11	NF	20	18	3	4	0	1	0	0	0	NF
12	NF	218	965	682	75	309	1	0	0	0	NF
13	NF	11	11	58	17	1	1	0	0	0	NF
14	PD	240	20	5	28	96	0	1	0.08	0	PD
15	PD	650	53	34	20	0	0	1	0	0.0125	PD
16	PD	1076	95	4	71	231	0	1	0	0	PD
17	ED	1565	93	34	47	0	0	0.03	1	1	ED
18	ED	300	240	14	160	140	0	0	0.997	0	ED
19	ED	212	38	15	47	0	0	0.317	**0.677**	0	Not diagnosed
20	ED	24	13	5	43	319	0	0.484	**0.596**	0	Not diagnosed
21	ED	858	1324	208	2793	7672	0	0	0.923	0	ED
22	ED	1249	370	56	606	1371	0	**0.862**	0.998	0	Not diagnosed
23	TF	199	770	217	1508	72	0	0	0.109	1	OHF
24	TF	2754	16615	3657	31476	613	0	0	0	1	OHF
25	TF	266	584	328	862	1	0	0	0.651	1	OHF
26	TF	80	619	326	2480	0	0	0	0.169	1	OHF
27	TF	231	3997	1726	5584	0	0.025	0	0.221	1	OHF
28	TF	65	61	16	143	3	0	0.022	0	0.993	OHF
29	TF	137	369	144	1242	16	0	0	0.488	1	OHF
30	TF	56	286	96	928	7	0	0	0	1	OHF
31	TF	86	110	18	92	7.4	0	0	0	1	OHF
32	TF	42	97	157	600	0	0	0	0	1	OHF
33	TF	73	520	140	1200	6	0	0	0.282	1	OHF

Table 5.10: Average CPU processing time for 25 runs of 2-ADOPT, MOECS, KNN-RF, AA-RF, ORF, and AdaBoost algorithms.

Algorithm	2-ADOPT	MOECS	KNN-RF	AA-RF	ORF	AdaBoost
Time (s)	680.59	388.91	194.65	238.57	86.8	375.27

overall accuracy of the proposed method is higher than the listed algorithms in Table 5.11. Furthermore, the number of diagnosed fault classes in the proposed method (2-ADOPT) is more than most of the methods in Table 5.11.

In order to graphically compare the TPR and FPR between the proposed method

Table 5.11: Comparison of the recent state-of-the-art transformers fault diagnosis algorithms based on DGA.

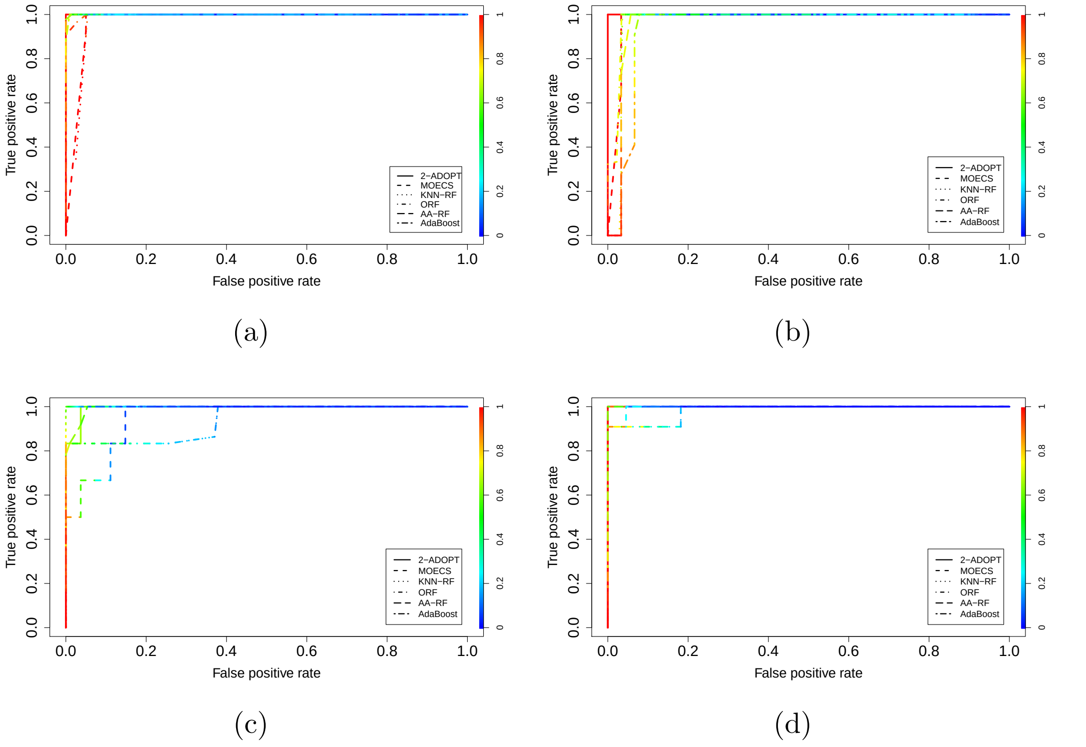
Reference	Number of samples				Accuracy (%)			
	NF	PD	ED	OHF	NF	PD	ED	OHF
(Dong et al., 2008)		60[a]				88.3[b]		
(Ghoneim et al., 2016)	56	32	146	184	48.2	75	97.3	94.8
(Shintemirov et al., 2009)	26	18	54	69		92.11[b]		
(Morais and Rolim, 2006)	180	10[c]		22	85.56	50[c]		63.63
(Ghoneim and Taha, 2016)		240[a]				92.91[b]		
(Tang et al., 2008)		168[d]				80.2[d]		
(Bacha et al., 2012)		30[a]				90[b]		
(Souahlia et al., 2012)		40[a]				85[b]		

[a] all classes
[b] overall
[c] (PD + ED)
[d] (OHF + ED)

Table 5.12: Comparison of the AUC and pAUC among 2-ADOPT, MOECS, RF, and ORF.

Method	AUC				pAUC			
	NF	PD	ED	OHF	NF	PD	ED	OHF
2-ADOPT	**1**	**1**	**0.99**	**1**	**0.1**	**0.1**	**0.09**	**0.1**
MOECS	0.97	0.97	0.95	0.98	0.07	0.07	0.06	0.08
KNN-RF	0.97	0.96	0.9	0.96	0.07	0.06	0.08	0.08
AA-RF	**1**	0.96	**0.99**	**1**	**0.1**	0.07	**0.09**	**0.1**
ORF	0.99	0.97	0.91	0.97	**0.1**	0.07	0.07	0.09
AdaBoost	**1**	0.94	0.95	**1**	**0.1**	0.05	**0.09**	**0.1**

and other methods, receiver operating characteristics (ROC) curves are plotted in Figure 5.12. These colorised ROC curves helps to give an informative view of TPR versus FPR at various cutoff points. For example, the first false positive of ED fault class in 2-ADOPT (Figure 5.12c) occurs at the probability equal to 0.6. Moreover, the area under curves (AUCs) and partial AUC at FPR equal to 0.1 are reported in Table 5.12. It should be noted that the maximum value of AUC and pAUC for a perfect classification are equal to 1 and 0.1, respectively.



(a)

(b)

(c)

(d)

Figure 5.12: ROC curves with colorized cutoff points to compare the proposed method (2-ADOPT), and two other ensemble methods (MOECS and random forests) for: (a) No fault, (b) Partial discharge (c) Energy discharge, and (d) Over heating fault classes.

5.10 Summary

Fault diagnosis of transformers depends on various factors, such as the type of transformers and environmental conditions, and these may differ from utility to utility. Therefore, applying intelligent methods to diagnose the faults of transformers increases reliability and accuracy of applied diagnostic techniques. The DGA dataset in this study was collected from various ranges and types of transformers. So, unlike other conventional methods, the proposed method is not highly dependent on the transformer type, and environmental and technical conditions. In other words, the proposed algorithm is an intelligent data-driven method. In this research, a two phase evolutionary multi-objective technique to diagnose faults of power transformers was proposed. First, a feature subset selection using the MOPSO algorithm was carried out to select the most relevant features for each fault class. The feature vectors on the best Pareto optimal were considered as inputs to train the classification algorithms in the second phase, accordingly. Subsequently, the MOPSO algorithm was again applied to find the best group of classifiers among 30 single trained algorithms, in terms of accuracy and diversity measures as objective functions. The selected solutions (group of classifiers) by MOPSO were examined on a validation set of DGA to find the best ensemble (solution) to classify power transformer faults. In addition, the proposed method was compared with three other scenarios; a multi-objective PSO based ensemble classifier selection without feature subset selection, random forests and oblique random forests ensemble techniques, where it consistently outperformed these scenarios over several performance metrics.

The proposed method can also be used “in house” by electric utilities and power companies to diagnose faults of their assets.

Chapter 6

Multi-Objective Ensemble Forecasting of Dissolved Gases in Power Transformer

6.1 Overview

In this chapter an ensemble time series forecasting algorithm using evolutionary multi-objective optimization algorithms to predict dissolved gas contents in power transformers is presented. In this method, the correlation between each individual dissolved gas and other transformer features, such as temperature characteristics and loading history, is first determined. Then, a non-linear principal component analysis (NLPCA) technique is applied to extract the most effective time series from the highly correlated features. Subsequently, the forecasting algorithms that support a cross validation technique are used for training. In addition, evolutionary multi-objective optimization algorithms are used to select the most accurate and diverse group of forecasting algorithms to construct an ensemble. Finally, the selected ensemble is examined to predict the value of the dissolved gases on the testing set. The results of one day, two day, three day, and four day ahead forecasting are presented, which show higher accuracy and reliability of the proposed method compared with other statistical methods.

6.2 Introduction

Thus far, most dissolved gas prediction models have placed emphasis on single forecasting algorithms. In (Fei and Sun, 2008) a Support Vector Machine (SVM) algorithm has been used to forecast the ratio of dissolved gases. A genetic algorithm (GA) has been also applied to find the optimum hyper parameters of the SVM through its training procedure. In another research, Fei et al. (Fei et al., 2009) investigated the forecasting of dissolved gases using SVM and using a particle swarm optimization (PSO) algorithm to adjust the hyper parameters of the SVM algorithm. In (Wang, 2004) the possibility of forecasting incipient faults of power transformers using grey-extension method has been studied. Ghunem et al. (Ghunem et al., 2012) applied multi-layer perceptron feed forward neural networks to predict the transformer oil furan contents. They also used a stepwise regression method to select the most effective inputs (features) using neural networks. In Liao et al. (2011b) the authors proposed a least squares support vector regression method to forecast dissolved gases in power transformers. A parameter tuning procedure using a PSO algorithm was used during the training phase of the least squares support vector regression algorithm. Furthermore, there are various proposed methods in the context of artificial intelligence and statistical modelling for forecasting time series (Chandra, 2015; Kavousi-Fard et al., 2016; Li et al., 2012; Miranian and Abdollahzade, 2013; Quan et al., 2014; Taieb and Atiya, 2016).

All of the aforementioned studies implemented a single objective framework to forecast the dissolved gas contents in power transformers. Although all these forecasting algorithms have been well trained, there are some questions that need to be further investigated:

- How can a prediction (forecasting) algorithm can be generalized to deal with new data sets?
- Which prediction (forecasting) algorithm is the most accurate method to apply for predicting dissolved gases in power transformer ?

In this chapter, a method to address these questions will be presented, which uses a multi-objective ensemble selection technique to overcome inaccuracies and uncertainties that exist in conventional DGA forecasting methodologies.

In machine learning, ensemble learning techniques are widely used to enhance the capability and accuracy of the classification and regression algorithms by avoiding the selection of a single weak technique (Polikar, 2006). In recent years, ensemble learning has been of great interest to researchers and is used in classification, regression, time series forecasting, and remaining useful lifetime estimation applications (Araujo and New, 2007; Kim and Kang, 2010; Kourentzes et al., 2014; Lariviere and Vandenpoel, 2005; Lim et al., 2017; Peimankar et al., 2016, in press; Wang and Chiang, 2011; Yan, 2012; Zhang et al., 2016). A key question however is how to effectively create a diverse and accurate ensemble in ensemble learning (Ren et al., 2016).

Three different Evolutionary Multi-Objective Optimization (EMO) algorithms were used to select the most accurate and diverse time series forecasting algorithm, among a group of previously trained algorithms. These include, Multi-Objective Particle Swarm Optimization (MOPSO) (Coello et al., 2004), Non-dominated Sorting Genetic Algorithm (NSGA-II) (Deb et al., 2002a), and the Strength Pareto Evolutionary Algorithm (SPEA-II) (Zitzler et al., 2001). Firstly, Non-Linear Principal Component Analysis (NLPCA) (Scholz et al., 2005) was used to extract an informative time series from highly correlated inputs. Secondly, different time series forecasting algorithms were trained as base predictors, such as Support Vector Regression (SVR) (Cortes and Vapnik, 1995), Regression Trees (RT) (Breiman et al., 1984), Group Method of Data Handling (GMDH) (Ivakhnenko, 1971), Radial Basis Function (RBF) (Chen et al., 1991), Adaptive Network-based Fuzzy Inference System (ANFIS) (Jang, 1993), Echo State Networks (ESN) (Jaeger, 2001), Kernel Ridge Regression (KRIDGE) (Murphy, 2012), Cascade Forward Neural Network (CFNN) (Fahlman and Lebiere, 1989), and Feed Forward Neural Network (FFNN) (Hagan et al., 1996). Each of these unique time series forecasting algorithms was trained with different parameter settings and training functions. Thirdly, an EMO algorithm selects the most accurate and diverse group of algorithms. Lastly, the outputs of the selected algorithms in the ensemble are combined to forecast the content of dissolved gases in power transformers.

6.3 Time series extraction using NLPCA

The data for this study were collected from sensors installed on the power transformers. Our dataset consisted seven measured dissolved gases (H_2, CH_4, C_2H_2, C_2H_4, C_2H_6, CO, and CO_2) from the insulation oil of a power transformer, followed by the load history, ambient temperature, oil temperature, and winding temperature of the transformer. Loading history and, as a consequence, the thermal characteristics of transformers, have significant effects on the level of dissolved gases in power transformers. Therefore, it becomes very important to consider these factors when forecasting the dissolved gases. Furthermore, a dissolved gas in transformer oil, as a member of the hydrocarbon gases, is sometimes correlated with the level of other dissolved gases. The pairwise Pearson's correlation coefficients (CC) were calculated (Dowdy et al., 2011) of these input time series. As general rule, two variables are said to be positively correlated if $0.5 \leq CC \leq 1$ and are negatively correlated if $-1 \leq CC \leq -0.5$. However, there is not a general rule to set this threshold accurately and the best solution is using a statistical test to confirm if there is a significant correlation between variables (time series). For this purpose, a post hoc right tailed test is used for testing the null hypothesis of no correlation, against the alternative of significant positive correlation between variables. Figure 6.1 illustrates the Pearson's pairwise correlation coefficients. The scatter plots of the pair variables are also shown and the slopes of the least squares fitted lines is equal to the correlation coefficient. It should be noted that the time series (variables) are normalized to zero mean and unit standard deviation. The histograms of each time series are also plotted. The asterisks indicate if the correlation between variables are statistically significant. The larger and higher number of asterisks show that the two corresponding variables are more significantly correlated. Therefore, corresponding to each variable, the positively correlated variables with three large asterisks are selected as highly correlated variables. Since there are multiple comparisons, a correction method called Bonferroni correction is performed to adjust the significant level in Figure 6.1. However, the adjusted significance values are not changed much.

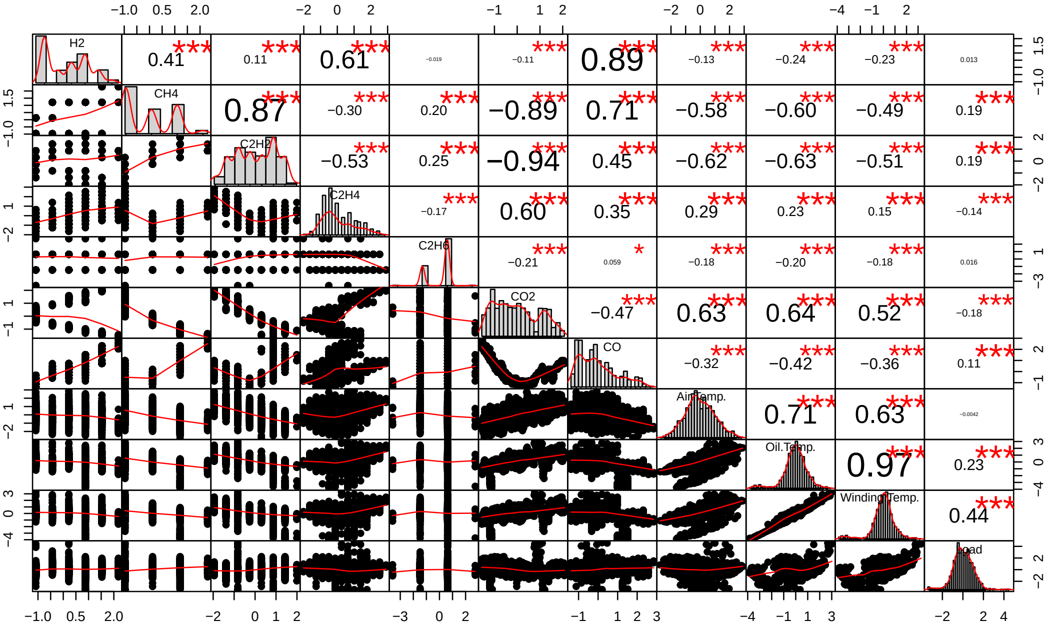


Figure 6.1: The Pearson's pairwise correlation coefficients along the scatter plots of the pair variables and the histogram of the variables. The slopes of the least squares fitted lines are equal to the correlation coefficient. The larger and more asterisks show the more significant correlation between variables.

After determining the highly correlated time series, NLPCA was used to extract an effective time series from the highly correlated variables. Principle Component Analysis (PCA) is a well-known technique in statistics and machine learning to extract the best features (principle components). PCA is a linear approach but there are different non-linear PCA techniques, such as kernel based PCA and auto-associative based PCA. An auto-associative based NLPCA is also applied, which is shown in Figure 6.2 (Scholz et al., 2005). The $\mathbf{x_i}$ and $\mathbf{Z}$ are the inputs and extracted time series, respectively. The network of the extraction phase (solid lines) is stored to be used in the generation phase (dashed lines) in the next step of the algorithm to reconstruct the targeted time series. The NLPCA approach is implemented using the nonlinear PCA toolbox in MATLAB (Scholz, 2014).

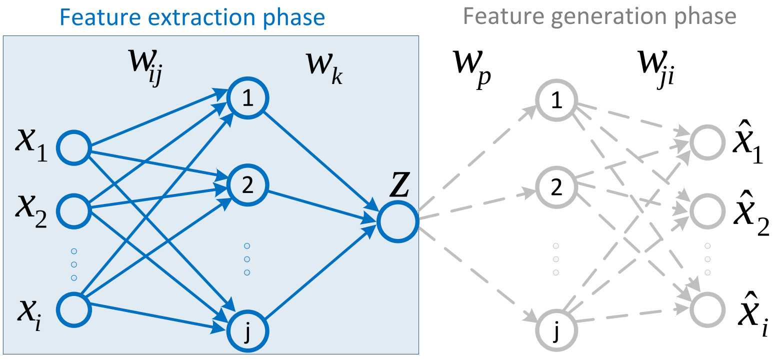


Figure 6.2: Architecture of NLPCA. The $\mathbf{x}$ and $\mathbf{Z}$ are the correlated time series as inputs and the extracted most informative time series (principle component), respectively. The feature generation network (dashed lines) is used to reconstruct the predicted time series $\widehat{\mathbf{x}}$.

6.4 Stationarity analysis

Developing a successful time series forecasting model requires the provision of a model with a stationary series as input. In general, a time series consists of three distinct components: 1) non-seasonal trend component, 2) seasonal component, and 3) stochastic component. To make a non-stationary series stationary, one needs to remove the trend and seasonality components from the series. In other words, the desired time series should have a zero mean and variance such that only the stochastic term remains.

The DGA time series in this study are non-stationary series with non-seasonal trend components. One of the most common techniques to ensure a time series is stationary is to apply a differencing method. Differencing is defined as (Bowerman et al., 2005):

$$s(t) = s\left(t_0\right) - s\left(t_0 - 1\right), \tag{6.1}$$

where $s(t)$ and $s(t_0)$ are the original and differenced time series, respectively. The differenced time series $(s(t))$ along with the autocorrelation function (ACF) and

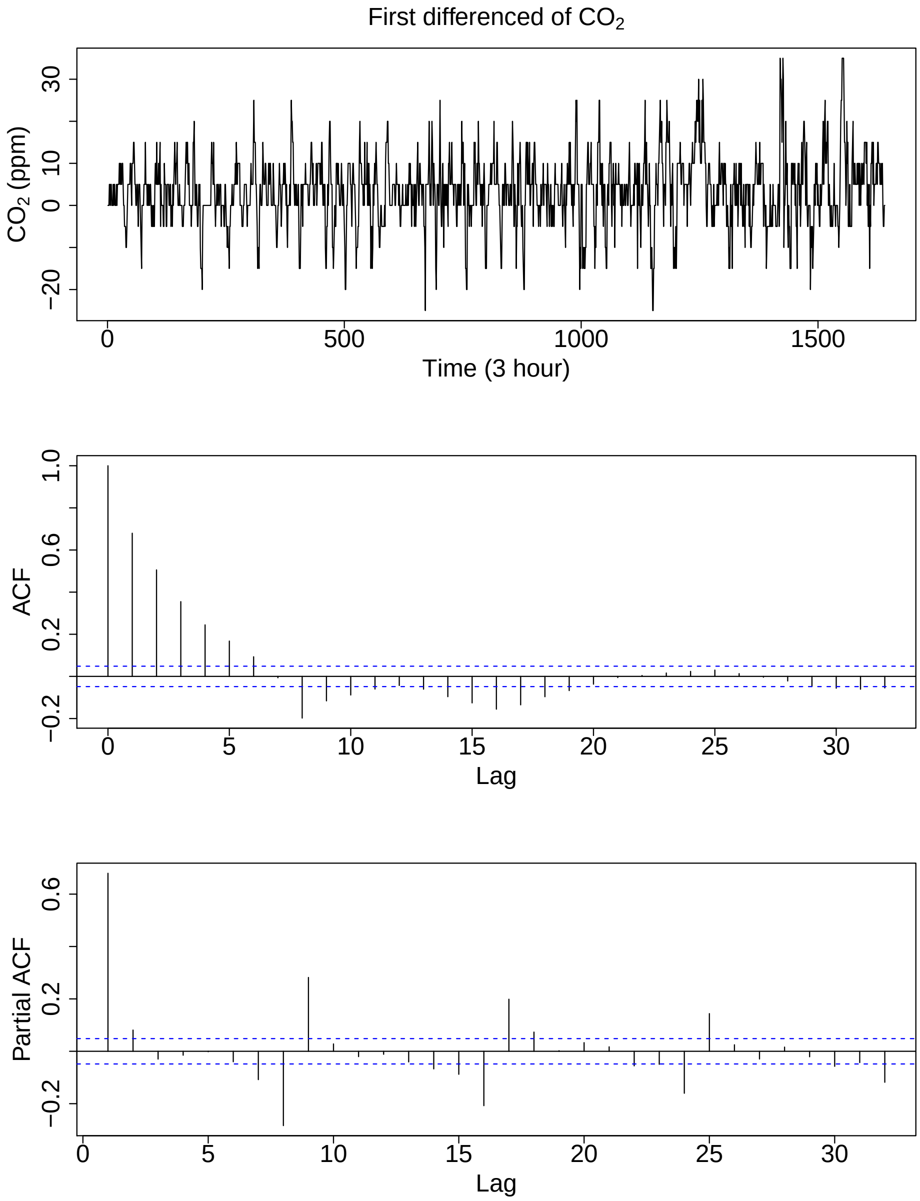
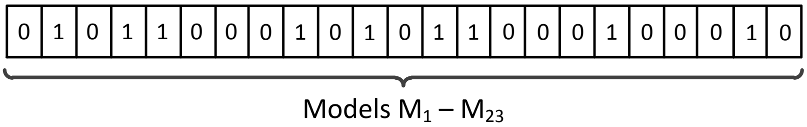


Figure 6.3: First differenced of CO$_2$ time series with its ACF and PACF analyses over a time period of six months (July 2015 - January 2016).

partial autocorrelation function (PACF) are plotted in Figure 6.3. From the ACF and PACF (Figure 6.3), the appropriate inputs for the time series forecasting model can be determined. With the trial and error method, the daily DGA time series, which consists of eight lags (every three hours), was selected as inputs for the forecasting algorithms. In addition, the ACF and PACF were used to determine the orders (p and q) of the autoregressive integrated moving average (ARIMA(p,d,q)) model in Section 6.8 to compare with the proposed algorithm discussed in this chapter.

6.5 Multi-objective ensemble time series forecasting

There are 23 time series forecasting algorithms, as listed in Table 6.1, which are trained using the extracted time series data by NLPCA. In the multi-objective optimization algorithm, solutions are found which are binary vectors whose lengths equal the number of selected time series forecasting algorithms. An example of a solution for the multi-objective optimization algorithm is shown in Figure 6.4. In this binary vector, 1's represent the selected algorithms to forecast the time series, while 0's show that the corresponding algorithms are not selected.



0	1	0	1	1	0	0	0	1	0	1	0	1	1	0	0	0	1	0	0	0	1	0

Figure 6.4: An example of a solution in a multi-objective evolutionary optimization for selecting time series forecasting algorithms. The 1's represent the corresponding selected forecasting algorithms.

In multi-objective ensemble learning, we need to carefully define our objective functions to select a group of the most accurate and diverse algorithms. Diversity can significantly improve the performance of the ensemble by alleviating the over-fitting problem and create a generalized model. Although the accuracy can be easily formulated, the diversity measure is still an open issue in classification,

Table 6.1: The list of the time series forecasting algorithms used to create the ensemble.

No.	Model	Description
1	rTree	Binary regression decision tree.
2-4	SVR	Support vector regression with radial basis function, linear and Gaussian kernel functions. The kernel scale parameters are tuned using a heuristic approach during the training phase.
5-7	GMDH	Group method of data handling with 5 layers and 10, 20, and 50 maximum neuron size in hidden layers, respectively.
8-10	RBF	Radial basis network with 10, 50, and 100 neurons in the RBF layer, respectively.
11	ANFIS	Adaptive network-based fuzzy inference system with Sugeno type fuzzy inference and using fuzzy c-means clustering (Bezdek, 2013) to generate clusters. The optimum number of clusters are determined using the subtractive clustering technique (Chiu, 1994).
12-14	ESN	Echo state network with 10, 50, and 100 internal units, respectively.
15-17	KRIDGE	Kernel Ridge Regression with radial basis, polynomial, and linear kernel functions.
18-20	CFNN	Cascade forward neural network using Levenberg-Marquardt as optimization algorithm with 10, 50, and 100 hidden layer size, respectively.
21-23	FFNN	Feedforward neural network using Levenberg-Marquardt as optimization algorithm with 10, 50, and 100 hidden layer size, respectively.

regression, and time series forecasting ensemble learning (Kuncheva and Whitaker, 2003; Ren et al., 2016). For classification problems, different diversity measures have been defined (Kuncheva and Whitaker, 2003). Whereas, for regression/time series forecasting problems, the diversity of the ensemble can be meet by considering the covariance between the base predictors (Brown et al., 2005b).

Suppose a dataset of N input and target vectors is given by,

$$z = \{(\mathbf{x}_1, t_1), (\mathbf{x}_2, t_2), \ldots, (\mathbf{x}_N, t_N)\}. \tag{6.2}$$

It should be noted that each data point is sampled from an unknown distribution $p(\mathbf{x}, t)$. The problem is to find an estimator g that maps the inputs to targets in order to minimize the cost function (Friedman et al., 2001),

$$err(g) = \int (g(\mathbf{x}, \mathbf{w}) - t)^2 p(\mathbf{x}, t) d(\mathbf{x}, t). \tag{6.3}$$

where $\boldsymbol{\omega}$ are the parameters of the estimator function $g(.)$. Since $p(\mathbf{x}, t)$ is an unknown distribution, Equation 6.3 should be inevitably substituted with a summation,

$$err(g) \approx \frac{1}{N} \sum_{n=1}^{N} (g(\mathbf{x}_n, \boldsymbol{w}) - t_n)^2, \quad (\mathbf{x}_n, t_n) \in z. \tag{6.4}$$

There are two important issues that need to be taken into consideration. First, if the $\boldsymbol{\omega}$ parameters of the estimator g are tuned to achieve an absolute zero value for err, the model will suffer from over-fitting which leads to poor performance on a future dataset because the true distribution of $p(\mathbf{x}, t)$ is unknown. On the other hand, if the $\boldsymbol{\omega}$ parameters are not tuned properly to their optimum values, the model will suffer from under-fitting which again leads to poor performance on future datasets. Geman et al. (1992) formulated Equation 6.4 in a bias-variance decomposition,

$$\begin{aligned} E\{(g(\mathbf{x}_n, \boldsymbol{w}) - t)^2\} &= (E\{g(\mathbf{x}_n, \boldsymbol{w})\} - t)^2 + E\{(g(\mathbf{x}_n, \boldsymbol{w}) - E\{g(\mathbf{x}_n, \boldsymbol{w})\})\} \\ &= \text{bias}(g(\mathbf{x}_n, \boldsymbol{w}))^2 + \text{variance}(g(\mathbf{x}_n, \boldsymbol{w})), \end{aligned} \tag{6.5}$$

where $E\{.\}$ is the expectation operator, which is used as a substitute for summation in Equation 6.4. In an ensemble with M members, the bias-variance decomposition can be defined as follows:

$$\begin{aligned} E\{(\overline{g} - t)^2\} = (E\{\overline{g}\} - t)^2 + E\{(\overline{g} - E\{\overline{g}\})\}, \\ s.t. \quad \overline{g} = \frac{1}{M} \sum_{i=1}^{M} g_i(\mathbf{x}, \boldsymbol{w}_i). \end{aligned} \tag{6.6}$$

To achieve a trade-off between accuracy and diversity in an ensemble, a bias-variance-covariance decomposition was defined by Ueda and Nakano (1996). For this purpose, three terms: average bias, average variance, and average covariance of the ensemble were defined, as follows:

$$\overline{\text{Bias}} = \frac{1}{M}\sum_{i=1}^{M}(E\{g_i\} - t). \tag{6.7}$$

$$\overline{\text{Var}} = \frac{1}{M}\sum_{i=1}^{M}E\{(g_i - E\{g_i\})^2\}. \tag{6.8}$$

$$\overline{\text{Covar}} = \frac{1}{M(M-1)}\sum_{i=1}^{M}\sum_{j=i+1}^{M}E\{(g_i - E\{g_i\})(g_j - E\{g_j\})\}. \tag{6.9}$$

Subsequently, the bias-variance-covariance decomposition can be formulated as:

$$E\{(\overline{g} - t)^2\} = \overline{\text{Bias}}^2 + \frac{1}{M}\overline{\text{Var}} + (1 - \frac{1}{M})\overline{\text{Covar}}. \tag{6.10}$$

In addition, Krogh et al. (1995) introduced another approach to provide diversity in an ensemble, which is called *ambiguity decomposition*, and is formulated as follows:

$$(\overline{g} - t)^2 = \overbrace{\frac{1}{M}\sum_{i=1}^{M}(g_i - t)^2}^{\text{average error of predictors}} - \overbrace{\frac{1}{M}\sum_{i=1}^{M}(g_i - \overline{g})}^{\text{ambiguity term}}. \tag{6.11}$$

From Equation 6.10 and Equation 6.11, we can also derive:

$$E\left\{\frac{1}{M}\sum_{i=1}^{M}(g_i - t)^2 - \frac{1}{M}\sum_{i=1}^{M}(g_i - \overline{g})\right\} = \overline{\text{Bias}}^2 + \frac{1}{M}\overline{\text{Var}} + (1 - \frac{1}{M})\overline{\text{Covar}}. \tag{6.12}$$

In other research, Brown et al. (2005b) showed that the two terms in Equation 6.12 can be divided as follows:

$$\begin{aligned} E\left\{\frac{1}{M}\sum_{i=1}^{M}(g_i - t)^2\right\} &= \overline{\text{Bias}}^2 \\ &+ \overbrace{\overline{\text{Var}} + \frac{1}{M}\sum_{1}^{M}(E\{g_i\} - E\{\overline{g}\})^2}^{\Theta}.[15pt] \end{aligned} \tag{6.13}$$

$$E\left\{\frac{1}{M}\sum_{i=1}^{M}(g_i-\overline{g_i})^2\right\} = \overbrace{\overline{\text{Var}}+\frac{1}{M}\sum_{1}^{M}(E\{g_i\}-E\{\overline{g}\})^2}^{\Theta} - \left[\frac{1}{M}\overline{\text{Var}}+(1-\frac{1}{M})\overline{\text{Covar}}\right]. \qquad (6.14)$$

The term Θ confirms the fact that the diversity measure (Equation 6.14) cannot be individually maximized without any effect on the average mean squared error of the ensemble (Equation 6.13). Therefore, Equation 6.13 and Equation 6.14 are considered as objective functions for the multi-objective optimization evolutionary algorithms. It is proposed that finding an optimum trade-off between these two objective functions will guarantee the most accurate and diverse ensemble of forecasting algorithms for predicting the value of dissolved gases in power transformers.

To minimise the forecasting error, whilst simultaneously maximising the diversity of the selected ensemble, a multi-objective time series model with the following constraints is applied:

$$\text{Minimize} \quad CF_1 = \tfrac{1}{M}\textstyle\sum_{i=1}^{M}(g_i-t)^2. \qquad (6.15)$$

$$\text{Maximise} \quad CF_2 = \tfrac{1}{M}\textstyle\sum_{i=1}^{M}(g_i-\overline{g_i})^2. \qquad (6.16)$$

Equation 6.15 is applied to minimize the forecasting error, and simultaneously, Equation 6.16 is used to maximize the diversity of the selected ensemble. The proposed multi-objective time series selection method for the forecasting of power transformer's dissolved gases is discussed in Section 6.7.

6.6 Evolutionary multi-objective optimization algorithms

The two objective functions to find the most accurate and diverse group of time series forecasting algorithms were formulated in Section 6.5. In this section, the three evolutionary multi-objective optimization algorithms, which are used to achieve an optimum trade-off between accuracy and diversity, are described. These algorithms are categorised into population-based algorithms. MOPSO is a multi-objective version of the PSO algorithm (Eberhart and Kennedy, 1995) which is a meta-heruristic algorithm developed by Coello et al. (2004) in 2004. Generally, in swarm intelligence, agents are appointed to handle a problem and to achieve a unique goal, which is not able to be handled by individual agents. The details of MOPSO algorithm can be found in Section 5.5. The NSGA-II is also a very well-known evolutionary multi-objective optimization algorithm, which was introduced by Deb et al. Deb et al. (2002a). Similarly, this algorithm is inspired by the single objective Genetic Algorithm (Goldberg and Holland, 1988). The third evolutionary multi-objective optimization algorithm is called SPEA-II (Zitzler et al., 2001). In this algorithm an index is assigned to each solution that measures the strength of the corresponding solution compared with others in order to find the non-dominated solutions. A common concept among all the aforementioned evolutionary multi-objective optimization algorithms is called Pareto optimality, which is explained in Section 5.4. The details of NSGA-II and SPEA-II algorithms are described in the following section.

6.6.1 NSGA-II algorithm

As mentioned in Section 6.6, NSGA-II is a multi-objective version of the GA. So, in order to have a deeper understanding, first we review the main steps in GA which are: 1) create an initial main population; 2) use binary tournament selection to create a parent population; 3) apply crossover and mutation operators to the parent population to produce child and mutated populations, respectively; 4) create a new population from main, child, and mutated populations.

The main difference between GA and NSGA-II algorithms is in step 4, where a new population should be created. Figure 6.5 illustrates the procedure of the GA over one iteration. The sorting concept in NSGA-II is revised using a two step sorting technique. These steps are called non-dominated sorting and crowding distance. In non-dominated sorting, each solution in the population is compared with others to count the number of dominations (n_{dom}) of each solution. Solutions

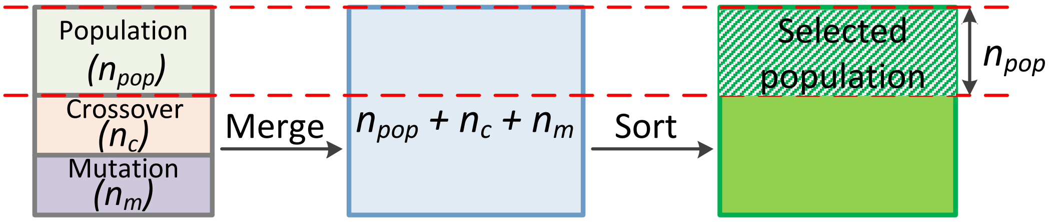


Figure 6.5: Schematic diagram of GA.

with $n_{dom} = 0$ are considered as the first Pareto front (F_1). Then, the solutions on F_1 are removed from the population and a similar procedure is repeated to find the F_2. This procedure continues until all solutions are categorized into their corresponding Pareto fronts. The details of the non-dominated sorting algorithm are given in Deb et al. (2002a). Figure 6.6 illustrates how the solutions are sorted using a non-dominated sorting algorithm.

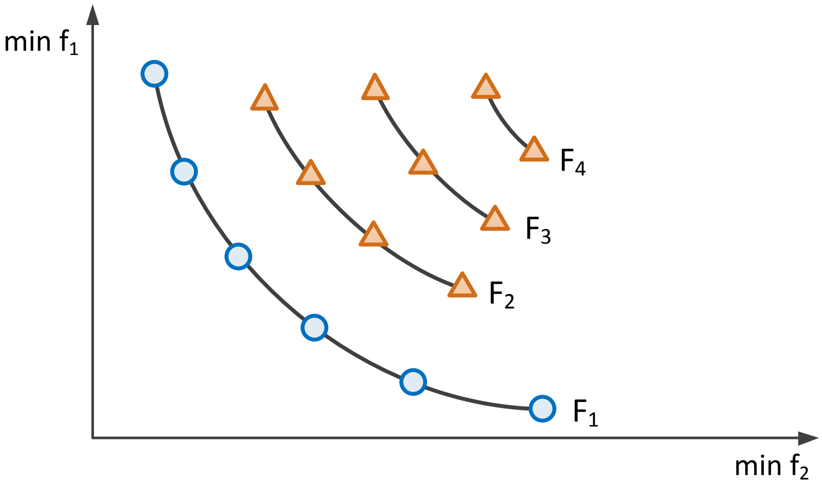


Figure 6.6: Schematic diagram of non-dominated sorting.

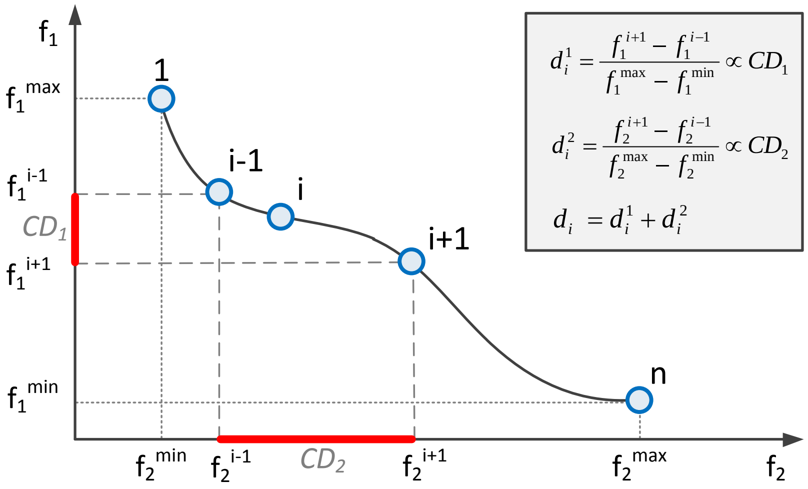

f_1
$d_i^1 = \frac{f_1^{i+1} - f_1^{i-1}}{f_1^{\max} - f_1^{\min}} \propto CD_1$
$d_i^2 = \frac{f_2^{i+1} - f_2^{i-1}}{f_2^{\max} - f_2^{\min}} \propto CD_2$
$d_i = d_i^1 + d_i^2$
1
f_1^{max}
i-1
i
f_1^{i-1}
CD_1
i+1
f_1^{i+1}
n
f_1^{min}
f_2^{min}
f_2^{i-1}
CD_2
f_2^{i+1}
f_2^{max}
f_2

Figure 6.7: Schematic diagram of CD.

Similar to other evolutionary multi-objective optimization algorithms, the non-dominated solutions may not be able to be selected based on only one factor, e.g., non-dominated sorting ranking, because of the archive size limitations. So, the crowding distance is used as a second ranking criteria in the NSGA-II algorithm. The crowding distance helps to select non-dominated solutions from the more sparse solutions to increase the exploration of the algorithm. The crowding distance (CD) concept is graphically shown in Figure 6.7 and is also formulated as,

$$\begin{aligned} d_i^j &= \frac{|f_j^{i-1} - f_j^{i+1}|}{f_j^{max} - f_j^{min}}, \\ d_i = d_i^1 + d_i^2 + \ldots + d_i^M &= \sum_{j=1}^{M} d_i^j, \end{aligned} \tag{6.17}$$

where d_i^j is the CD of the ith solution for the jth objective function. Figure 6.8 illustrates the sorting and non-dominated solutions selection steps in NSGA-II. As

Algorithm 2: Pseudo-code of NSGA2 algorithm.

```
Set the values of NSGA-II parameters
Initialize the population $pop$
Evaluate objective values on initialized populations
Select non-dominated solutions as leader $gbest$
for $it \leftarrow 1$ to $MaxIt$ do
    for $it_c \leftarrow 1$ to $n_c$ do
        Apply crossover to create $pop_c$
    end
    for $it_m \leftarrow 1$ to $n_m$ do
        Apply mutation to create $pop_m$
    end
    Merge $n_{Pop}$, $n_m$, and $n_c$
    for $it_{Pop} \leftarrow 1$ to $n_{Pop}$ do
        Apply non-dominated sorting
    end
    Calculate crowding distance of solutions on each front
    Sort population
end
Report Pareto optimal set (non-dominated solutions)
```

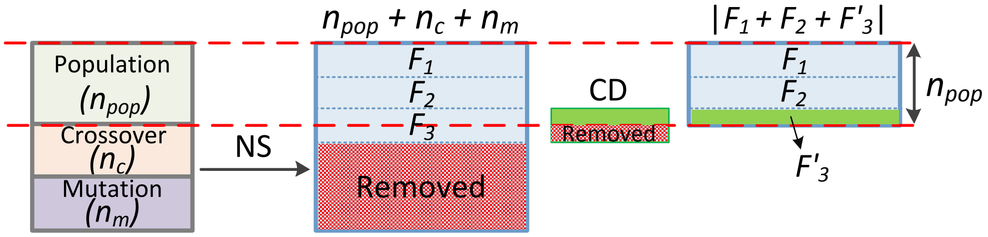


Figure 6.8: Schematic diagram of NSGA-II.

shown, the solutions on F_1 and F_2 are directly selected after NS ranking but some of the solutions in F_3 are rejected to enter the archive set after applying CD sorting on them. The pseudo-code of the NSGA-II algorithm is given in Algorithm (2).

The two main operators of any GA algorithms are crossover and mutation which help searching new solutions and generate a new set of solutions for the optimisation problem. The crossover technique used in this study is called single point crossover to generate a new solution (a “child”) from a pair of “parent” (Srinivas and Patnaik, 1994). For the mutation operator, a unique process is done such that some of the values in the chromosome are randomly replaced. For example, if length of chromosome is 20 and the mutation rate is set to 20%, four values in the chromosome are randomly replaced. The chromosomes here consist of a binary string which represent the selected forecasting algorithms as shown in Figure 6.4. The chances of occurring crossover and mutation are also given using crossover and mutation probabilities. Before the main loop of the algorithm some parameters should be set, which are:

- population size ($nPop$) = 300,
- maximum number of iteration ($MaxIt$) = 100,
- crossover probability (p_c) = 0.7,
- the number of solutions for crossover (n_c) $= 2 \times round(\frac{p_c \times nPop}{2})$,
- mutation probability (p_m) = 0.4,
- the number of solutions for mutation (n_m) $= round(p_m \times nPop)$,
- mutation rate (μ) = 0.2,

6.6.2 SPEA-II algorithm

The Improved Strength Pareto Evolutionary Algorithm (SPEA-II) was introduced by Zitzler et al. (2001) in 2001. This multi-objective evolutionary based algorithm also utilizes genetic operators such as mutation and crossover. The strength Pareto term is the key concept of this algorithm. This is a relative index that shows to what degree a solution is close to being a non-dominated solution or a member of the Pareto optimal set. This index is defined as follows (Zitzler et al., 2001):

$$S(i) = |\{j \mid j \in P_t \cup \overline{P_t} \wedge i \preceq j\}|, \quad i \in P_t \cup \overline{P_t} \tag{6.18}$$

where i and j are two arbitrary solutions, P_t and $\overline{P_t}$ are the population and archive sets at the tth iteration, respectively, and the $|.|$ is the cardinality operator such that $|P_t| = N$ and $|\overline{P_t}| = \overline{N}$. The next definition is called the raw fitness and is calculated as follows (Zitzler et al., 2001):

$$R(i) = \sum_{k \in P_t \cup \overline{P_t}, k \preceq i} S(k), \quad i \in P_t \cup \overline{P_t}, \tag{6.19}$$

where the $R(i)$ is always an integer number. The smaller the $R(i)$, the better the solution.

Ideally, the corresponding $R(i)$ for non-dominated solutions is equal to zero. Figure 6.9 shows an example of the assigned $R(i)$ in the tth iteration using the SPEA-II algorithm. For instance, solution A, which is a non-dominated solution, dominates two other solutions (K and I), so $S(A) = 2$ and $R(A) = 0$. On the other hand, solution K is dominated only by solution A and $R(K) = 2$.

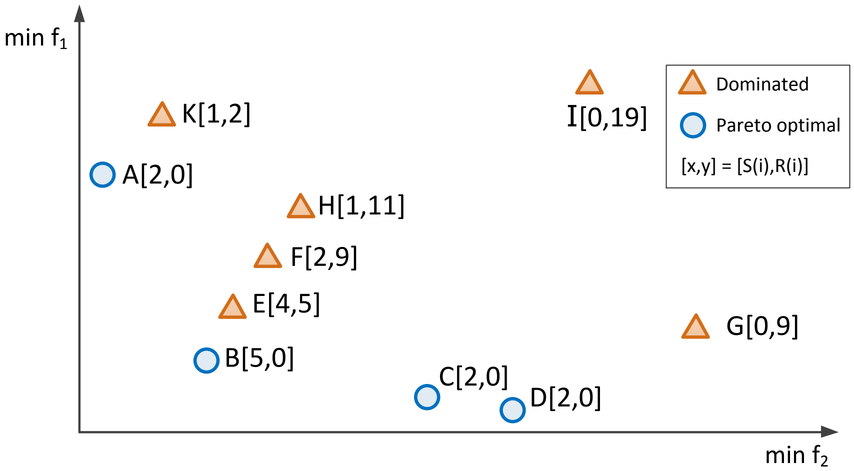


Figure 6.9: An example of assigned $R(i)$ and $S(i)$ in SPEA-II algorithm.

Two scenarios can be considered here: 1) if $|\overline{P}_{t+1}| < \overline{N}$, some of the dominated solutions should also be included in the archive set at the $(t+1)$th iteration; 2) if $|\overline{P}_{t+1}| > \overline{N}$, the additional member of the archive should be truncated, such that the unique solutions, which are in a more sparse space, are kept to increase the exploration of the algorithm. For example, let us firstly assume $\overline{N} = 7$. So, the first seven selected solutions are [A, B, C, D, E, K] and one needs to choose either solution F or solution G as the last archive member. In this example, the raw fitness of both dominated solutions (G and F) are equal to 9. Therefore, it is not possible to choose a solution from these two pairs based on only raw fitness, and another selection criterion should be taken into consideration. In the second scenario, we assume $\overline{N} = 3$. So, three out of four non-dominated solutions should be selected in the $(t + 1)$th iteration. Since the raw fitness for all non-dominated solutions is equal to 0, similar to the first case, another selection criterion is necessary to be considered here. To tackle this problem, a density function $D(i)$ is defined:

$$D(i) = \frac{1}{\sigma_i^k + 2}, \quad 0 < D(i) \leq \frac{1}{2}. \tag{6.20}$$

where σ_i^k is the distance of the ith solution from its kth nearest neighbor. The value of k, as recommended in the K-NN algorithm (Silverman, 1986), is set to $\sqrt{N + \overline{N}}$. The density function is calculated for all solutions. Subsequently, a new fitness function can be formulated as follows:

$$F(i) = R(i) + D(i). \tag{6.21}$$

Algorithm 3: Pseudo-code of SPEA-II algorithm.

```
Set the values of SPEA-II parameters
Initialize a population ($P_0$) and allocate an empty set as archive ($\overline{P}_0$)
Evaluate the objective values on initialized population and rank them
for $t \leftarrow 1$ to $T$ do
    for $n \leftarrow 1$ to $(N + \overline{N})$ do
        Calculate the fitness values for all solutions ($P_t \cup \overline{P}_t$) using Equation 6.21
        Select non-dominated solutions of $P_t \cup \overline{P}_t$
        if $|\overline{P}_{t+1}| < \overline{N}$ then
            Choose some of the dominated solutions to fill the archive
        else if $|\overline{P}_{t+1}| > \overline{N}$ then
            Remove additional solutions from non-dominated solutions (Pareto optimal) using Equation 6.22
        else
            Copy all non-dominated solutions to $\overline{P}_{t+1}$
        end
    end
    Check the stopping criteria; Use binary tournament selection to choose solutions from $\overline{P}_{t+1}$ for the mating pool
    Apply mutation and crossover operators to the mating pool and create $P_{t+1}$
end
Report Pareto optimal set (non-dominated solutions)
```

Subsequently, in case 1, $D(G) = 0$ and $D(F) = 2$, which results in $F(G) = 9 < F(F) = 11$, and solution G is selected. In case 2, to select from non-dominated solutions, a removal procedure is conducted using Equation 6.22 which is also based on selecting the solutions from the more sparse space to increase the exploration of the optimization algorithm. Therefore, solutions A, B and D are selected here. The non-dominated solutions selection process is formulated as:

$$\begin{aligned} i \preceq j \Leftrightarrow \exists 1 \leq k \leq |\overline{P}_{t+1}| \colon [(\forall 1 \leq l \leq k \colon \sigma_i^l = \sigma_j^l) \wedge \sigma_i^k < \sigma_j^k], \\ \vee \quad \forall\, 1 \leq k \leq |\overline{P}_{t+1}| \colon \sigma_i^k = \sigma_j^l, \end{aligned} \tag{6.22}$$

where i is the non-dominated solution, which is chosen to be removed from the archive set at the $(t+1)$th iteration, and j is a member of non-dominated solutions

(Pareto optimal) at the $(t+1)$th iteration. The pseudo-code of the SPEA-II algorithm is described in Algorithm (3). The parameters of SPEA-II algorithm should be first set, which are: 1) population size (N); 2) archive size $(\overline{N})$; and 3) the maximum number of iterations (T).

6.7 Proposed methodology

Figure 6.10 shows the schematic diagram of the proposed ensemble method for forecasting of dissolved gas contents in power transformers. The proposed method takes advantage of EMO algorithms to create an accurate and diverse ensemble of time series forecasting algorithms. The proposed method is described in ten main steps as follows:

1. *Normalization*: All input time series are first normalized to zero mean and unit standard deviation. This step should be done before running the NLPCA algorithm as it is very sensitive to the scale of the data.

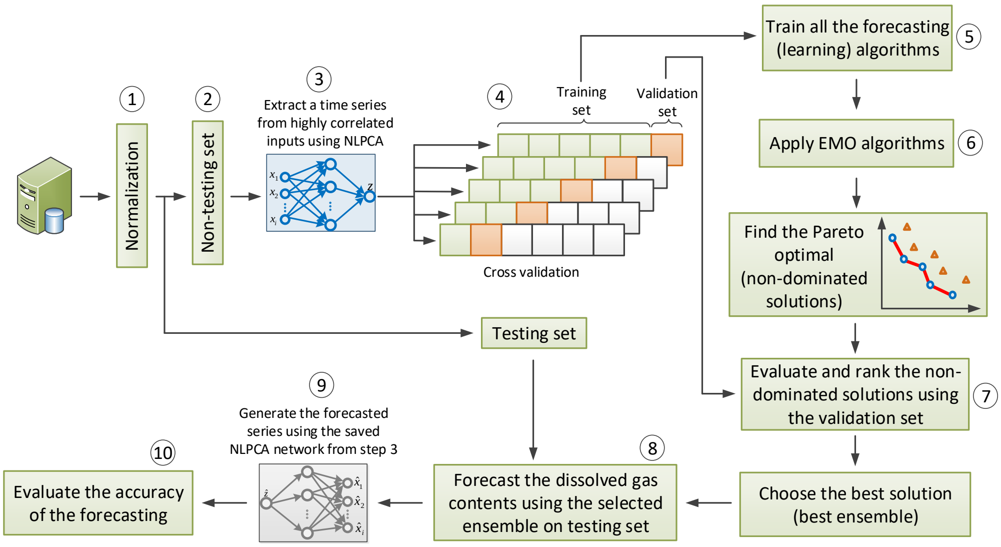

1
Normalization
2
Non-testing set
3
Extract a time series from highly correlated inputs using NLPCA
4
Training set
Validation set
Cross validation
Testing set
5
Train all the forecasting (learning) algorithms
6
Apply EMO algorithms
Find the Pareto optimal (non-dominated solutions)
7
Evaluate and rank the non-dominated solutions using the validation set
Choose the best solution (best ensemble)
8
Forecast the dissolved gas contents using the selected ensemble on testing set
9
Generate the forecasted series using the saved NLPCA network from step 3
10
Evaluate the accuracy of the forecasting

Figure 6.10: Schematic diagram of the proposed forecasting method.

3. *Separate the testing set from a non-testing set*: The collected time series dataset is divided into two sets. The non-testing set is used to train and validate the forecasting models and the testing set is used to evaluate the proposed algorithm.

4. *Extract time series*: The NLPCA is applied to extract a higher level of time series from the highly correlated time series inputs.

5. *Cross validation (CV)*: A rolling window CV technique is performed to achieve a reliable error estimation. As it is shown in Figure 6.10, the non-testing dataset is divided into six folds and each time one fold is added to the training set and the last fold is considered as validation set. Since the DGA dataset is for a period of six months (July 2015 - January 2016), this type of CV can help to consider the seasonal effect in training the forecasting algorithms.

6. *Train all the single forecasting algorithms using extracted time series*: All the listed forecasting algorithms in Table 6.1 are trained and their training and validation errors are reported.

7. *Apply evolutionary multi-objective algorithms to select the best group of forecasting algorithms*: The evolutionary multi-objective algorithms using two objective functions defined in Section 6.5 utilize the most accurate and diverse ensembles.

8. *Evaluate the solutions on Pareto optimal/archive set*: Each solution on the Pareto front is a vector of selected single forecasting algorithms (Figure 6.4). They are all non-dominated solutions and one of them should be considered to forecast dissolved gas contents on the testing set. In this step, all non-dominated solutions are evaluated and ranked on the validation set.

9. *Forecast the dissolved gas contents using the selected ensemble on testing set*: The best non-dominated solution selected in the previous step is used to forecast the dissolved gas contents on the testing set.

10. *Generate time series*: The forecasted time series is used as an input for the stored NLPCA network in step 3 to generate the targeted time series.

11. *Evaluate the performance*: The accuracy of the proposed forecasting method is evaluated using two metrics discussed in Section 6.8.1.

6.8 Results and discussion

In this chapter three multi-objective ensemble approaches were used to forecast the dissolved gas contents in power transformers: MOPSO based ensemble time series forecasting, NSGA-II based ensemble time series forecasting, and SPEA-II based ensemble time series forecasting. In addition, these methods are compared with four different techniques: 1) weighted ensemble method which assigns a normalised weight to each forecasting algorithm using validation accuracy; 2) autoregressive integrated moving average (ARIMA) (Box et al., 2015); 3) simple exponential smoothing (SES) (Holt, 2004); and 4) the persistence model (PER) (Polikar, 2006). All seven aforementioned approaches are evaluated on a collected dataset of dissolved gas contents, load history, and three temperature readings over a period of six months (July 2015-January 2016). The dissolved gas contents were measured every 3 hours, while temperatures and load history were measured every 3 minutes. Figure 6.11 presents the proportion of the data used to train, validate, and test the forecasting methods.

Four forecasting time horizons were considered which are 8 steps ahead (one-day), 16 steps ahead (two-day), 24 steps ahead (three-day), and 32 steps ahead (four-day).

6.8.1 Time series forecasting performance metrics

Two metrics are employed to evaluate the performance of the time series forecasting methods: 1) RMSE (Root Mean Squared Error); and 2) MAPE (Mean Absolute Percentage Error). These two error measures are calculated as follows:

$$RMSE = \sqrt{\frac{\sum_{t=1}^{T}(y_t - \hat{y}_t)^2}{T}}, \tag{6.23}$$

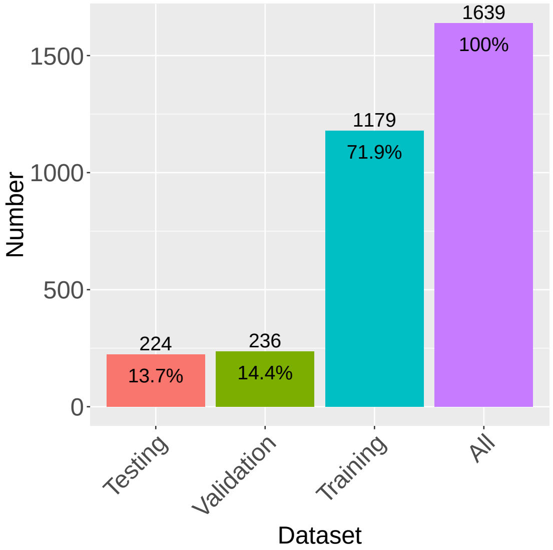


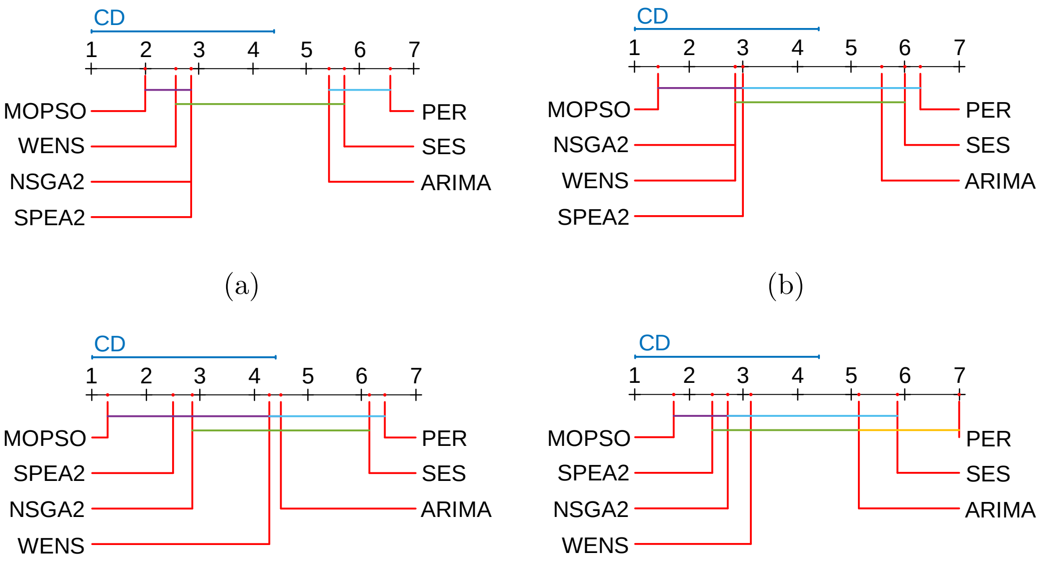
Figure 6.11: Number of samples used in training, validation, and testing phase.

$$MAPE = \frac{1}{n} \sum_{t=1}^{T} |\frac{y_t - \hat{y}_t}{y_t}|, \quad y_t \neq 0, \tag{6.24}$$

where T is the number of samples, and y_t and $\hat{y}_t$ are the actual and the forecasted values of the dissolved gases, respectively.

6.8.2 Performance comparison of dissolved gases forecasting models

The multi-objective based ensemble methods were compared with four benchmark models using a non-parametric statistical test called the Friedman test, as shown in Figure 6.12. The Friedman test ranks the forecasting models on predicting different dissolved gases and the Nemenyi post hoc test determines if there is a significant difference between these forecasting models. Figure 6.12 shows the results of the Friedman test and the average ranks of forecasting models. In this figure the models without significant difference are connected. For this purpose, the Friedman's critical value q_α, at a 0.05 significance Demšar (2006) level for the seven forecasting models and employing the seven dissolved gases dataset, was calculated the 2.948. Based on the obtained critical value, there are some significant



(a) (b)

(c) (d)

Figure 6.12: Comparison of forecasting methods using Nemenyi post hoc test and the average rank of different methods for: (a) 8 steps ahead (one day), (b) 16 steps ahead (two day), (c) 24 steps ahead (three day), and (d) 32 steps ahead (four day) dissolved gas contents forecasting. The methods without significance performance difference are connected.

differences between the forecasting methods especially when we go further into the future. For example, the mean rank difference between the MOPSO based ensemble forecasting method, and three benchmark models (ARIMA, SES, and PER) for all time horizons, are greater than the 3.4041 and there is also a significant performance difference between MOPSO and WENS for four day ahead forecasting. However, the performances of the three multi-objective approaches are comparable.

To group the methods between those that showed no significant differences, a *post hoc* Nemenyi test is employed. The MATLAB toolbox for Nemenyi post hoc test was obtained from (Kourentzes, 2013). A critical distance (CD) is then calculated as follows :

$$CD = q_{\alpha}\sqrt{\frac{k(k+1)}{6b}}, \tag{6.25}$$

where k and b are the number of methods and datasets, respectively (both equal to 7 here), and $q_\alpha = 2.948$. Therefore, in this experiment, $CD = 3.4041$. In Figure 6.12, the methods with the mean rank difference smaller than the calculated CD are connected. Therefore, there is no significant performance difference between them.

The two computed error metrics, MAPE and RMSE, of the forecasting methods over all time horizons, are given in Table 6.2. The lowest MAPE and RMSE values in Table 6.2 are in bold, which confirms the better performance of the multi-objective approaches. According to Figure 6.12, three of the four benchmark models (ARIMA, SES, and PER) have the lower mean rank in almost all four forecasting time horizons, and among them, the persistence model, which states no change for the future steps compared to the last observed point that has the lowest rank. From Table 6.2, the multi-objective based ensemble forecasting methods outperform the WENS, ARIMA, SES, and persistence models as we go further into the future (three and four day ahead forecasting horizons). From Figure 6.12, the MOPSO-based ensemble algorithm achieved the highest rank among all the models to forecast the dissolved gases. Therefore, the percentage improvement of the MOPSO-based ensemble algorithm, compared with WENS, ARIMA, SES, and persistence models, was investigated. The percentage improvement is calculated as follows:

$$\frac{\text{Benchmark performance} - \text{proposed performance}}{\text{Benchmark performance}} \times 100 \tag{6.26}$$

The percentage improvement of MOPSO-based ensemble performance compared with four benchmarking models (WENS, ARIMA, SES, and PER) over all seven dissolved gas time series forecasting are presented in Figure 6.13. In almost all cases, the proposed method improved forecasting compared with the three benchmark models ARIMA, SES, and PER by more than 50%, while the average percentage improvement over all DGA datasets for weighted ensemble model (WENS) is more than 20%.

Multi-objective based ensemble methods utilize more than one algorithm to forecast the dissolved gas contents, which improves the forecasting accuracy significantly. Figure 6.14 shows the selected time series forecasting algorithms in the

Table 6.2: Comparison of the forecasting performance among proposed ensemble methods and other prediction techniques.

	Method	RMSE							MAPE						
		H_2	CH_4	C_2H_2	C_2H_4	C_2H_6	CO_2	CO	H_2	CH_4	C_2H_2	C_2H_4	C_2H_6	CO_2	CO
one day ahead	MOPSO	0.1842	0.0011	0.0967	0.0967	0.1373	5.1327	**1.7273**	1.3113	0.0631	**1.8333**	**0.9468**	19.9581	1.3334	1.1617
	NSGA2	0.1937	0.0171	0.0963	0.0963	0.1379	**5.0078**	1.7462	1.4064	0.0952	1.8835	0.9610	20.4483	**1.2744**	1.1749
	SPEA2	0.1951	0.0121	0.0957	**0.0957**	0.1381	5.0129	1.7462	1.4531	0.6713	1.8368	0.9580	20.5290	1.2835	1.1599
	ARIMA	0.5677	0.0304	0.4955	0.8936	0.7363	26.1733	5.1733	2.1400	1.5216	2.1100	1.8589	23.6278	1.8650	1.5395
	SES	0.5678	0.0304	0.4936	0.9025	0.7377	26.3668	5.1986	2.1393	1.5214	2.0263	1.8962	24.0000	1.9034	1.8650
	PER	0.6598	0.0300	0.4982	0.9325	0.7433	32.5397	5.5704	3.1322	1.5000	2.1955	2.0090	25.6845	2.5684	1.6170
	WENS	**0.1799**	**0.0007**	**0.0926**	0.1057	**0.1336**	5.8201	1.7728	**1.2099**	**0.0210**	2.1260	1.0306	**18.5329**	1.4011	**1.1466**
two day ahead	MOPSO	0.1848	0.0023	**0.0994**	**0.2559**	**0.1383**	**5.1649**	1.8045	1.4624	0.1277	**1.8983**	**0.9681**	**20.5325**	**1.3484**	1.2170
	NSGA2	0.1989	0.0183	0.1001	0.2639	0.1394	6.5217	**1.7971**	1.5078	1.0174	1.9634	1.0125	21.2580	1.3717	1.2214
	SPEA2	0.1953	0.0109	0.1001	0.2639	0.1397	6.5487	1.8019	1.5146	0.9654	1.9634	1.0125	21.4040	1.3712	1.2239
	ARIMA	0.6666	0.0304	0.5044	0.9446	0.7452	31.3530	5.5569	3.3300	1.5211	2.2109	2.0224	25.7144	2.3290	1.6276
	SES	0.6604	0.0304	0.4982	0.9447	0.7434	32.2720	5.5789	3.1546	1.5216	2.4421	2.0112	26.0680	2.5306	1.7552
	PER	0.7532	0.0300	0.4964	0.9746	0.7535	32.5397	5.9063	4.1144	1.5000	2.1367	2.1902	28.5119	2.5684	2.3290
	WENS	**0.1825**	**0.0021**	0.1069	0.2783	0.1593	6.7232	1.8613	**1.3563**	**0.0870**	2.2047	1.5968	23.2452	1.8961	**1.2162**
three day ahead	MOPSO	0.1798	0.0015	**0.0964**	0.2747	**0.1385**	**5.1654**	**1.7934**	**1.3569**	0.0858	**1.7954**	**1.0570**	**20.4411**	**1.3582**	**1.1995**
	NSGA2	0.1801	0.0010	0.0967	0.2759	0.1419	6.0548	1.8006	1.3762	0.0534	1.8559	1.0747	21.7090	1.3585	1.2067
	SPEA2	0.1797	**0.0003**	0.0969	**0.2744**	0.1415	6.1029	1.7973	1.3811	**0.0175**	1.8737	1.0636	21.6947	1.3686	1.2011
	ARIMA	0.7684	0.0304	0.5017	0.9659	0.7501	40.7643	5.8706	4.1372	1.5211	2.1521	2.0345	26.6643	2.8227	1.7656
	SES	0.7539	0.0304	0.4963	0.9946	0.7536	41.2016	5.9153	4.6700	1.5216	2.3488	2.3062	28.5448	3.1432	2.1137
	PER	0.8250	0.0300	0.5077	0.9943	0.7564	47.4896	6.3274	5.0884	1.5186	2.4408	2.1292	28.7351	3.8503	2.8227
	WENS	**0.1786**	0.0024	0.1246	0.3761	0.1711	7.8553	1.8502	1.3681	0.1348	2.2318	2.8389	25.08763	1.9208	1.2017
four day ahead	MOPSO	0.1822	**0.0018**	0.0943	0.2808	**0.1294**	**10.0261**	1.8462	1.2894	**0.1002**	**1.7273**	**1.0585**	**19.1230**	**1.3854**	1.2433
	NSGA2	**0.1783**	0.0133	**0.0929**	0.2836	0.1318	12.3067	**1.8371**	**1.2333**	0.7389	1.7540	1.0791	19.8659	1.3887	1.2446
	SPEA2	0.1794	0.0064	**0.0929**	**0.2801**	0.1321	12.3658	**1.8371**	1.2394	0.3530	1.7527	1.0641	20.0570	1.3929	**1.2420**
	ARIMA	0.7684	0.0304	0.5068	1.0013	0.7501	55.9129	5.8706	5.9800	2.9211	4.5540	4.2169	32.4407	4.4054	4.1231
	SES	0.8162	0.0304	0.5060	1.0271	0.7588	57.4911	6.3494	6.7120	2.9215	4.4546	4.2980	32.7688	4.8013	4.4468
	PER	0.9050	0.0304	0.5069	1.0329	0.7576	68.0696	6.6596	7.2624	2.9224	5.9232	4.3731	33.0982	5.5232	5.4054
	WENS	0.1825	0.0029	0.1291	0.5717	0.1757	10.3064	1.9986	1.2648	0.1606	2.3469	3.1070	25.4819	1.9217	1.2218

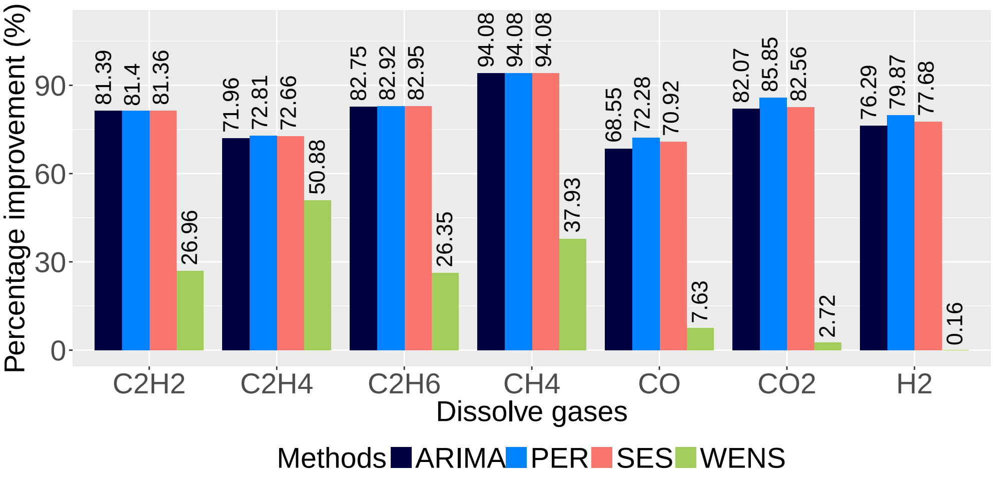


Figure 6.13: MOPSO-based ensemble forecasting percentage improvement compared with three benchmark models for four day ahead forecasting.

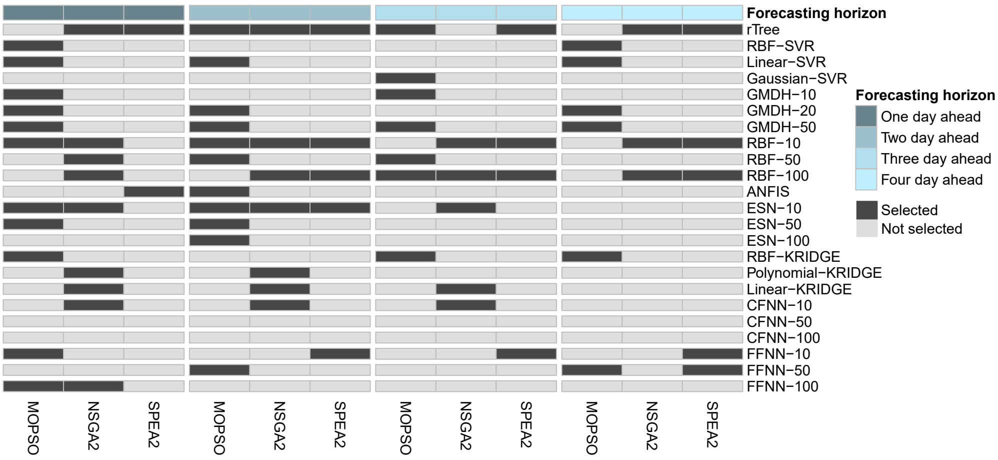


Figure 6.14: Selected time series forecasting algorithms for each ensemble using evolutionary multi-objective algorithms. The numbers in forecasting algorithms names represent the maximum number of neurons in the hidden layer except for ESN which is the size of internal units.

multi-objective based ensemble approaches. The outputs of selected forecasting algorithms in the chosen ensemble are averaged on each sample to predict the value of dissolved gases.

Since the MOPSO based ensemble method performs better than other methods,

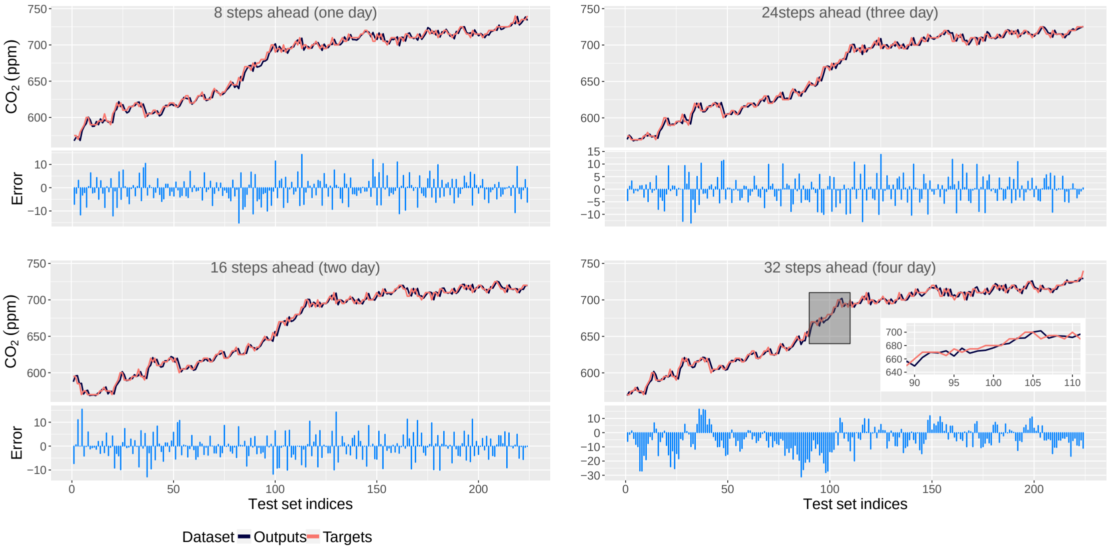


Figure 6.15: The predicted values of CO$_2$ dissolved gas and their corresponding actual values for four forecasting horizons using MOPSO based ensemble forecasting. The bar plots of errors are also represented for each forecasting horizon.

the results of this algorithm for forecasting CO$_2$ are presented to verify the performance of the proposed multi-objective ensemble dissolved gas forecasting method. Figure 6.15 shows the forecasted values (outputs) using the MOPSO based ensemble method on the test set. The patterns of the forecasting errors at the bottom of each subfigure in Figure 6.15 show that as over a longer time horizon is forecasted, the error increases. The histograms of the errors are represented in Figure 6.16 overlaid by density curves. The error mean and the error standard deviation (StD) of each time horizon forecasting are also given in Figure 6.16. The error mean of one day, two day, and three day ahead forecasts are close to zero (perfect forecasting). Moreover, the error mean of four day ahead forecast are actually within a reasonable range for this relatively long-term forecasting.

Furthermore, another metric was used to show how well the MOPSO based ensemble method forecasts the CO$_2$ contents, which is called coefficient of determination (R^2) Draper and Smith (2014). Figure 6.17 illustrates how close the forecasted values are to the actual values of CO$_2$. The higher the R^2 value, the better the forecasting method. The R^2 value can vary between 0 and 1 where $R^2 = 1$ for per-

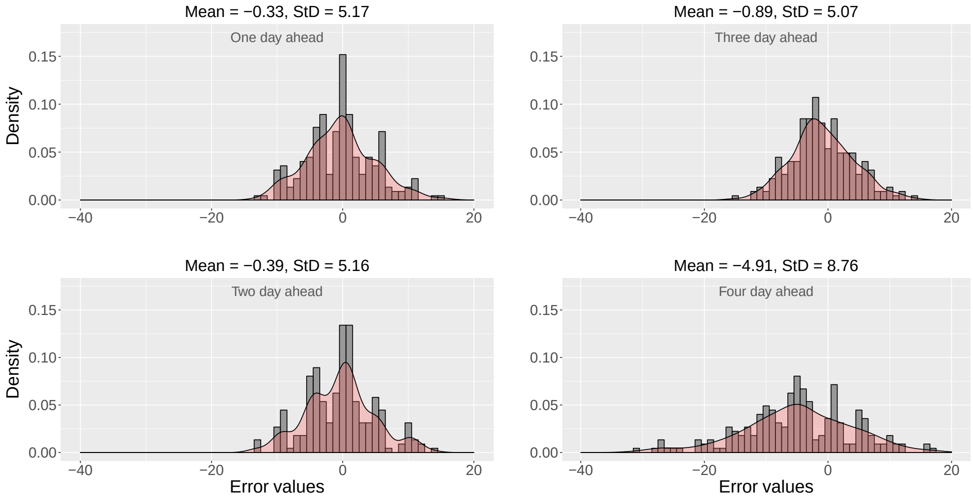


Figure 6.16: Density curve and histogram of error values of CO_2 dissolved gas forecasting using MOPSO based ensemble forecasting for four forecasting horizons. The error mean and error standard deviation for each forecasting horizon are also reported.

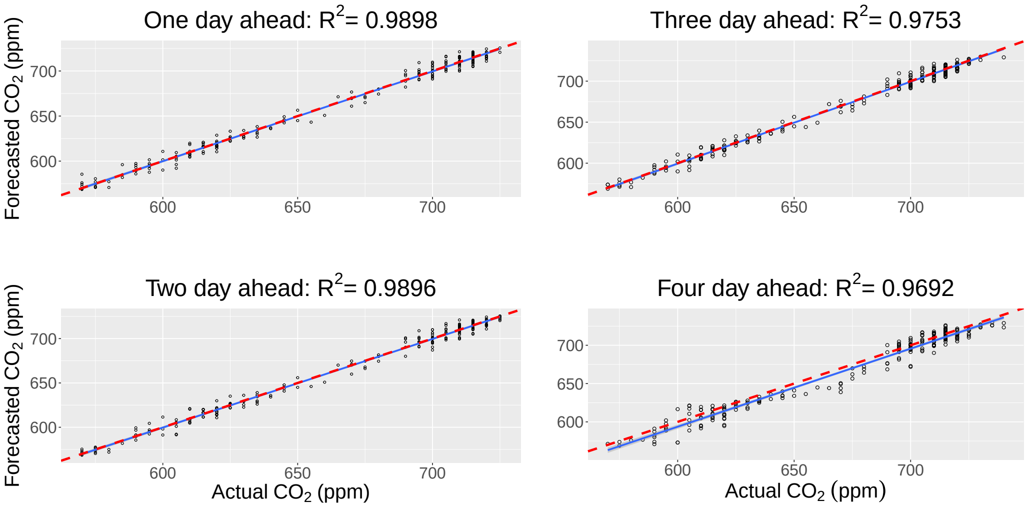


Figure 6.17: Linear regression of forecasted values relative to actual values of CO_2 dissolved gas forecasting using MOPSO based ensemble forecasting for four forecasting horizons. The red dashed lines represent the perfect forecasting where the forecasted values are exactly the same as actual values.

fect forecasting. The red dashed lines in Figure 6.17 represent perfect forecasting. If the fitting line (solid blue line) perfectly masks the dashed line, the forecasting method has a maximum accuracy (100%).

6.9 Summary

Forecasting dissolved gases of power transformers depends on different factors, such as the value of dissolved gas itself, the load history of the power transformers, and the ambient, oil, and winding temperatures. Considering all these factors, the contents of each dissolved gas helps to create an accurate and reliable forecasting model. In addition, the type and environmental conditions of power transformers vary widely. Therefore, utilizing an intelligent framework to forecast the dissolved gases is of great interest to electric utilities and power companies in order to achieve a better predictive based maintenance scheme. An effective time series from input variables was first extracted using a non-linear PCA method to train the forecasting algorithms. Then, evolutionary multi-objective optimization algorithms are applied to find the most accurate and diverse group of the forecasting algorithms among 23 trained algorithms. Subsequently, the selected non-dominated solutions were examined on the validation set to rank them and choose the best solution (group of the algorithms). The obtained results of the proposed method on the testing set were also compared with other conventional techniques. The proposed method outperformed the conventional methods in all forecasting time horizons. In addition, among three multi-objective ensemble forecasting methods (SPEA2, NSGA2, and MOPSO), the performance of the MOPSO algorithm was slightly higher. The prototype dissolved gas forecasting method can be used “in house” by electric utilities to accurately predict the trend of dissolved gases and to diagnose incipient faults of transformers.

Chapter 7

Conclusions and Future Work

7.1 Conclusions

In this thesis, firstly, some basic background about the different transformers faults were presented and their corresponding condition monitoring and condition assessment techniques were investigated. One of the most commonly used condition monitoring techniques in practice, which is called dissolved gas analysis, was reviewed in detail. This method is widely used by power companies and electric utilities to assess the condition of their transformer fleet. In addition, some of the most important conventional DGA based fault diagnosis methods were introduced and the main drawbacks of each method were discussed. The uncertainty of the traditional dissolved gas analysis base methods in classifying the correct faults of transformers was the main motivation of this research. To overcome these shortcomings, an intelligent condition assessment method was proposed using various statistical and machine learning techniques.

Some of the basic theory of statistical and machine learning algorithms used in this research were presented and discussed in Chapter 4. Each algorithm was explained to make it possible for the interested readers to understand these methods. Most of these algorithms were to be fine-tuned and implemented in both developed algorithms in this research for fault classification and dissolved gas forecasting in Chapter 5 and Chapter 6.

An intelligent load tap changers fault diagnosis algorithm was first developed using a single classifier learning system. In this algorithm a support vector machine classifier was used. Although, the preliminary results of this algorithm showed some improvements over other conventional techniques such as the modified Duval triangle for load tap changer fault diagnosis, there were still some concerns about the limitations of the proposed algorithm as listed in Section 4.5. The performance of the proposed hierarchical fault diagnosis algorithm for a classification between normal and faulty cases was better than the conventional method. However, the size of the available load tap changers dataset was small and the reported diagnostic accuracies may change on a larger dataset. The shortcomings and challenges of the single classification algorithm were the main motivations of developing an ensemble fault diagnosis algorithm for power transformers.

A model was developed for classifying faults of power transformers. The proposed algorithm used different classification algorithms in a multi objective ensemble to identify incipient faults of power transformers using dissolved gases in transformer oil. A multi objective particle swarm optimisation algorithm was utilised to select the most accurate and diverse group of classification algorithms and also the most relevant dissolved gases to each fault class. The chosen group of classifiers were then tested on unknown DGA samples to evaluate the accuracy of the proposed method. The accuracy of the proposed fault classification algorithm was comparable with the previous reported studies. The proposed algorithm is actually a data-driven method which is able to classify faults of transformers regardless of the environmental and technical conditions of transformers. The DGA samples used in this algorithm were collected from different published studies and from various ranges and types of transformers. The results of this method were also compared with other ensemble approaches which showed some improvements over these methods.

In addition to the fault classification algorithm, a time series forecasting algorithm was developed in Chapter 6, which utilised a multi objective ensemble to predict the future state of the dissolved gases in power transformers. For this purpose, three evolutionary multi-objective optimisation algorithms were applied to choose the most accurate and diverse group of time series forecasting algorithms. The result of this study confirmed that the multi-objective particle swarm optimisation algorithm performed better on selecting an ensemble of the best single forecasting algorithms compared with other optimisation approaches. The predicted dissolved gases using the proposed algorithm were also benchmarked against other traditional time series forecasting methods and showed some improvements over the desired forecasting horizons. Finally, a number of options for future research and development were presented.

7.2 Future Work

The research presented in this thesis can be further investigated in some aspects:

- Adding more intelligent condition assessment modules using various condition monitoring techniques.
- Developing an anomaly detection agent to alarm the abnormal operation of the transformers.
- Improving accuracy and reliability of the DGA forecasting algorithm using more advanced machine learning methods.
- Building a general asset management tool to estimate the remaining useful life of power transformers.

Following are the possible ways for further development of the listed suggestions. As shown in Figure 2.1, despite the DGA based fault classification, which was studied in this research, it is also possible to develop five more condition assessment algorithms. The algorithm would be a general condition assessment tool for power transformers. For this purpose, different single intelligent algorithms can be implemented and trained using the available dataset obtained from each condition monitoring technique. The main challenge is to collect historical data for each condition monitoring technique. Then, based on the decision of each condition assessment agent, a reliable and comprehensive decision can be made on the transformer faults.

The anomaly detection agent can be developed using online measured data from the sensors installed on the transformer. Some the anomaly detection units are as follows:

- *Top oil temperature monitoring*: An intelligent algorithm that receives loading history and the ambient temperature of the transformer as inputs can be developed to predict the top oil temperature. The agent can send an alarm in the case of abnormal operation when the value of top oil temperature is higher than a fixed threshold.
- *Dissolve gasses trend monitoring*: The trend of the dissolved gasses can be monitored to identify the sudden increase or decrease which may be a symptom of occurring faults inside the transformer.

The main problem in applying traditional machine learning techniques for time series forecasting tasks is to choose the most appropriate delays in the time series as inputs for the learning algorithm. For this purpose, in the proposed algorithm in Chapter 6, a maximum delay of eight was chosen by trial and error. In addition, the ESN algorithm as an architecture for recurrent neural network, which provides a short-term memory in the reservoir units, was also used in the ensemble of forecasting algorithm and helped to improve the forecasting results. However, using deep learning for forecasting tasks is very popular nowadays (Kur, 2014; Hu2, 2016; Qiu et al., 2014). One of the most promising deep learning architectures is called Long-Short Term Memory (LSTM), which is also a recurrent neural network (Gers et al., 2000; Hochreiter and Schmidhuber, 1997). LSTM has not only the short-term memory feature like other RNN architectures, but also make it possible to remembering time series values for a longer period of time. These properties can enhance the dissolved gas forecasting accuracies.

Since different failure modes (faults) can be determined by different condition assessment techniques, each diagnostic method can assign a value that shows its degree of certainty about a specific failure mode. These values can be considered as diagnostic probabilities of different condition assessment methods for each failure mode. The overall assessment can be done based on the assign values of different diagnostic methods. The current status of the transformer (e.g., normal, minor fault, major fault, and failed) can be determined based on the vector of assigned diagnostic probabilities by different diagnostic methods. Furthermore, a probabilistic approach such as hidden Markov model (HMM) can be used to estimate the failure rate and consequently remaining useful life of the transformer.

Appendix A

Ensemble Classifier Selection Using Multi-Objective PSO for Fault Diagnosis of Power Transformers

The content of this appendix is based on the published conference paper during the course work of this research in the 2016 IEEE World Congress on Computational Intelligence (IEEE CEC 2016).

A. Peimankar, S. J. Weddell, T. Jalal, A. C. Lapthorn

Ensemble classifier selection using multi-objective PSO for fault diagnosis of power transformers

2016 IEEE Congress on Evolutionary Computation (CEC), p. 3622-3629

Abstract: This paper presents a binary version of the Multi-Objective Particle Swarm Optimization (bi-MOPSO) algorithm to classify the faults of power transformers. The proposed method selects the most accurate and diverse classifiers, simultaneously. Then, the selected classifiers are combined to diagnose the actual faults of power transformers using dissolved gas analysis (DGA) performed on the oil of power transformers. The obtained results are compared to other scenarios such as combining the outputs of all classifiers or using only the most accurate classifier to diagnose the faults. The comparison reveals that the proposed method is highly reliable and useful for diagnosing the faults of power transformers.

doi=10.1109/CEC.2016.7744248

Bibliography

Life management techniques for power transformers. *CIGRE WG A2.18*, June 2003.

Time series forecasting using a deep belief network with restricted boltzmann machines. *Neurocomputing*, 137:47 – 56, 2014. Advanced Intelligent Computing Theories and MethodologiesSelected papers from the 2012 Eighth International Conference on Intelligent Computing (ICIC 2012).

Transfer learning for short-term wind speed prediction with deep neural networks. *Renewable Energy*, 85:83 – 95, 2016.

Ajith Abraham and Swagatam Das. *Computational intelligence in power engineering*, volume 302. Springer, 2010.

Ahmed E. B. Abu-Elanien and M. M. A. Salama. Asset management techniques for transformers. *Electric Power Systems Research*, 80(4):456–464, April 2010.

Enrique Alba, José García-Nieto, Laetitia Jourdan, and El-Ghazali Talbi. Gene selection in cancer classification using pso/svm and ga/svm hybrid algorithms. In *Evolutionary Computation, 2007. CEC 2007. IEEE Congress on*, pages 284–290. IEEE, 2007.

Monther Alhamdoosh and Dianhui Wang. Fast decorrelated neural network ensembles with random weights. *Information Sciences*, 264:104 – 117, 2014. Serious Games.

Julio E Alvarez-Benitez, Richard M Everson, and Jonathan E Fieldsend. A MOPSO algorithm based exclusively on pareto dominance concepts. In *Evolutionary Multi-Criterion Optimization*, pages 459–473. Springer, 2005.

M Araujo and M New. Ensemble forecasting of species distributions. *Trends in Ecology & Evolution*, 22(1):42–47, January 2007.

A. D. Ashkezari, H. Ma, T. K. Saha, and Y. Cui. Investigation of feature selection techniques for improving efficiency of power transformer condition assessment. *IEEE Transactions on Dielectrics and Electrical Insulation*, 21(2), 2014.

Khmais Bacha, Seifeddine Souahlia, and Moncef Gossa. Power transformer fault diagnosis based on dissolved gas analysis by support vector machine. *Electric Power Systems Research*, 83(1):73–79, feb 2012.

Pierre Baldi, Søren Brunak, Yves Chauvin, Claus AF Andersen, and Henrik Nielsen. Assessing the accuracy of prediction algorithms for classification: an overview. *Bioinformatics*, 16(5):412–424, 2000.

Riccardo Bellazzi and Blaz Zupan. Predictive data mining in clinical medicine: Current issues and guidelines. *International Journal of Medical Informatics*, 77 (2):81 – 97, 2008.

James C Bezdek. *Pattern recognition with fuzzy objective function algorithms.* Springer Science & Business Media, 2013.

S. A. Boggs. Partial discharge: overview and signal generation. *IEEE Electrical Insulation Magazine*, 6(4):33–39, July 1990.

Bruce L. Bowerman, Richard T. O'Connell, and Anne B. Koehler. *Forecasting, time series, and regression: an applied approach.* Thomson Brooks/Cole, Belmont, CA, 4th edition, 2005.

George EP Box, Gwilym M Jenkins, Gregory C Reinsel, and Greta M Ljung. *Time series analysis: forecasting and control.* John Wiley & Sons, 2015.

Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

Leo Breiman, Jerome Friedman, Charles J Stone, and Richard A Olshen. *Classification and regression trees.* CRC press, 1984.

David S Broomhead and David Lowe. Radial basis functions, multi-variable functional interpolation and adaptive networks. Technical report, DTIC Document, 1988.

Gavin Brown, Jeremy Wyatt, Rachel Harris, and Xin Yao. Diversity creation methods: a survey and categorisation. *Information Fusion*, 6(1):5–20, 2005a.

Gavin Brown, Jeremy L. Wyatt, and Peter Tio. Managing Diversity in Regression Ensembles. *Journal of Machine Learning Research*, 6(Sep):1621–1650, 2005b.

R. Chandra. Competition and collaboration in cooperative coevolution of elman recurrent neural networks for time-series prediction. *IEEE Transactions on Neural Networks and Learning Systems*, 26(12):3123–3136, Dec 2015.

Krishna Teerth Chaturvedi, Manjaree Pandit, and Laxmi Srivastava. Self-organizing hierarchical particle swarm optimization for nonconvex economic dispatch. *Power Systems, IEEE Transactions on*, 23(3):1079–1087, 2008.

S. Chen, C.F.N. Cowan, and P.M. Grant. Orthogonal least squares learning algorithm for radial basis function networks. *IEEE Transactions on Neural Networks*, 2(2):302–309, March 1991.

Zhao Chenglin, Sun Xuebin, Sun Songlin, and Jiang Ting. Fault diagnosis of sensor by chaos particle swarm optimization algorithm and support vector machine. *Expert Systems with Applications*, 38(8):9908–9912, 2011.

Stephen L Chiu. Fuzzy model identification based on cluster estimation. *Journal of Intelligent & fuzzy systems*, 2(3):267–278, 1994.

C. A. Coello Coello and M. S. Lechuga. MOPSO: a proposal for multiple objective particle swarm optimization. In *Evolutionary Computation, 2002. CEC '02. Proceedings of the 2002 Congress on*, volume 2, pages 1051–1056, 2002.

C.A.C. Coello, G.T. Pulido, and M.S. Lechuga. Handling multiple objectives with particle swarm optimization. *IEEE Transactions on Evolutionary Computation*, 8(3):256–279, jun 2004.

Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine Learning*, 20(3):273–297, September 1995.

T. Cover and P. Hart. Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, 13(1):21–27, January 1967.

K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, 6(2):182–197, April 2002a.

Kalyanmoy Deb. *Multi-objective optimization using evolutionary algorithms*, volume 16. John Wiley & Sons, 2001.

Kalyanmoy Deb, Amrit Pratap, Sameer Agarwal, and TAMT Meyarivan. A fast and elitist multiobjective genetic algorithm: NSGA-II. *Evolutionary Computation, IEEE Transactions on*, 6(2):182–197, 2002b.

Janez Demšar. Statistical comparisons of classifiers over multiple data sets. *Journal of Machine learning research*, 7(Jan):1–30, 2006.

Doernenburg and Strittmatter. Monitoring oil-cooled transformers by gas analysis. *Brown Boveri Review*, 61:238–247, 1974.

Lixin Dong, Dengming Xiao, Yishan Liang, and Yilu Liu. Rough set and fuzzy wavelet neural network integrated with least square weighted fusion algorithm based fault diagnosis research for power transformers. *Electric Power Systems Research*, 78(1), jan 2008.

M. Dorigo and L. M. Gambardella. Ant colony system: a cooperative learning approach to the traveling salesman problem. *IEEE Transactions on Evolutionary Computation*, 1(1):53–66, Apr 1997.

Shirley Dowdy, Stanley Wearden, and Daniel Chilko. *Statistics for research*, volume 512. John Wiley & Sons, 2011.

Norman R. Draper and Harry Smith. *Applied regression analysis.* Wiley, New York, 3rd edition, 2014. ISBN 9780471170822;0471170828;.

V. Duraisamy, N. Devarajan, D. Somasundareswari, A. Antony Maria Vasanth, and S.N. Sivanandam. Neuro fuzzy schemes for fault detection in power transformer. *Applied Soft Computing*, 7(2):534 – 539, 2007.

M. Duval. Fault gases formed in oil-filled breathing ehv power transformers- the interpretation of gas analysis data. In *IEEE PAS Conference*, 1974.

M. Duval. A review of faults detectable by gas-in-oil analysis in transformers. *IEEE Electrical Insulation Magazine*, 18(3):8–17, May 2002.

M. Duval. The duval triangle for load tap changers, non-mineral oils and low temperature faults in transformers. *IEEE Electrical Insulation Magazine*, 24(6): 22–29, November 2008.

Russ C Eberhart and James Kennedy. A new optimizer using particle swarm theory. In *Proceedings of the sixth international symposium on micro machine and human science*, volume 1, pages 39–43. New York, NY, 1995.

Ahmed Elhossini, Shawki Areibi, and Robert Dony. Strength Pareto particle swarm optimization and hybrid EA-PSO for multi-objective optimization. *Evolutionary Computation*, 18(1):127–156, 2010.

Jeffrey L Elman. Finding structure in time. *Cognitive science*, 14(2):179–211, 1990.

A.M. Emsley and G.C. Stevens. Review of chemical indicators of degradation of cellulosic electrical paper insulation in oil-filled transformers. *Science, Measurement and Technology, IEE Proceedings -*, 141(5):324–334, Sep 1994.

Andries P Engelbrecht. *Computational intelligence: an introduction.* John Wiley & Sons, 2007.

Scott E Fahlman and Christian Lebiere. The cascade-correlation learning architecture. 1989.

Tom Fawcett. An introduction to ROC analysis. *Pattern Recognition Letters*, 27 (8):861 – 874, 2006. ROC Analysis in Pattern Recognition.

Sheng-Wei Fei and Yu Sun. Forecasting dissolved gases content in power transformer oil based on support vector machine with genetic algorithm. *Electric Power Systems Research*, 78(3):507 – 514, March 2008.

Sheng-wei Fei, Ming-Jun Wang, Yu-bin Miao, Jun Tu, and Cheng-liang Liu. Particle swarm optimization-based support vector machine for forecasting dissolved gases content in power transformer oil. *Energy Conversion and Management*, 50(6):1604–1609, June 2009.

Wilfredo C. Flores, Enrique E. Mombello, Jos. A. Jardini, Giuseppe Ratt, and Antonio M. Corvo. Expert system for the assessment of power transformer insulation condition based on type-2 fuzzy logic systems. *Expert Systems with Applications*, 38(7):8119–8127, July 2011.

Jerome Friedman, Trevor Hastie, and Robert Tibshirani. *The elements of statistical learning*, volume 1. Springer series in statistics Springer, Berlin, 2001.

L.V. Ganyun, Cheng Haozhong, Zhai Haibao, and Dong Lixin. Fault diagnosis of power transformer based on multi-layer {SVM} classifier. *Electric Power Systems Research*, 74(1):1 – 7, 2005.

N Gao, GJ Zhang, Z Qian, Z Yan, and DH Zhu. Diagnosis of DGA based on fuzzy and ANN methods. In *Electrical Insulating Materials, 1998. Proceedings of 1998 International Symposium on*, pages 767–770. IEEE, 1998.

Stuart Geman, Elie Bienenstock, and René Doursat. Neural networks and the bias/variance dilemma. *Neural computation*, 4(1):1–58, 1992.

Felix A. Gers, Jrgen Schmidhuber, and Fred Cummins. Learning to forget: Continual prediction with lstm. *Neural Computation*, 12(10):2451–2471, 2000.

Mohamad Ghaffarian Niasar. *Mechanisms of Electrical Ageing of Oilimpregnated Paper due to Partial Discharges*. PhD thesis, KTH Royal Institute of Technology, 2015.

Sherif S. M. Ghoneim and Ibrahim B. M. Taha. A new approach of DGA interpretation technique for transformer fault diagnosis. *International Journal of Electrical Power & Energy Systems*, 81:265–274, oct 2016.

Sherif S. M. Ghoneim, Ibrahim B. M. Taha, and Nagy I. Elkalashy. Integrated ANN-based proactive fault diagnostic scheme for power transformers using dissolved gas analysis. *IEEE Transactions on Dielectrics and Electrical Insulation*, 23(3):1838–1845, jun 2016.

Sayan Ghosh, Swagatam Das, Debarati Kundu, Kaushik Suresh, and Ajith Abraham. Inter-particle communication and search-dynamics of lbest particle swarm optimizers: An analysis. *Information Sciences*, 182(1):156 – 168, 2012. Nature-Inspired Collective Intelligence in Theory and Practice.

R. A. Ghunem, K. Assaleh, and A. H. El-hag. Artificial neural networks with stepwise regression for predicting transformer oil furan content. *IEEE Transactions on Dielectrics and Electrical Insulation*, 19(2):414–420, April 2012.

David E Goldberg and John H Holland. Genetic algorithms and machine learning. *Machine learning*, 3(2):95–99, 1988.

Gene H Golub, Michael Heath, and Grace Wahba. Generalized cross-validation as a method for choosing a good ridge parameter. *Technometrics*, 21(2):215–223, 1979.

IAR Gray. A guide to transformer oil analysis. *Transformer Chemistry Service*, 2009.

JL Guardado, JL Naredo, P Moreno, and CR Fuerte. A comparative study of neural network efficiency in power transformers diagnosis using dissolved gas analysis. *IEEE Transactions on Power Delivery*, 16(4):643–647, 2001.

Martin T Hagan, Howard B Demuth, Mark H Beale, and Orlando De Jesús. *Neural network design*, volume 20. PWS publishing company Boston, 1996.

Per Christian Hansen. *Rank-deficient and discrete ill-posed problems: numerical aspects of linear inversion.* SIAM, 1998.

Haibo He, Yang Bai, E.A. Garcia, and Shutao Li. ADASYN: Adaptive synthetic sampling approach for imbalanced learning. In *Neural Networks, 2008. IJCNN 2008. (IEEE World Congress on Computational Intelligence). IEEE International Joint Conference on*, pages 1322–1328, June 2008.

Tin Kam Ho. The random subspace method for constructing decision forests. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 20(8):832–844, 1998.

Sepp Hochreiter and Jrgen Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 1997.

Arthur E Hoerl and Robert W Kennard. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12(1):55–67, 1970.

Charles C. Holt. Forecasting seasonals and trends by exponentially weighted moving averages. *International Journal of Forecasting*, 20(1):5 – 10, 2004.

R. A. Hooshmand, M. Parastegari, and Z. Forghani. Adaptive neuro-fuzzy inference system approach for simultaneous diagnosis of the type and location of faults in power transformers. *IEEE Electrical Insulation Magazine*, 28(5):32–42, September 2012.

Kurt Hornik, Maxwell Stinchcombe, and Halbert White. Multilayer feedforward networks are universal approximators. *Neural Networks*, 2(5):359 – 366, 1989.

Chao Hu, Byeng D. Youn, Pingfeng Wang, and Joung Taek Yoon. Ensemble of data-driven prognostic algorithms for robust prediction of remaining useful life. *Reliability Engineering & System Safety*, 103:120 – 135, 2012. ISSN 0951-8320.

Yann-Chang Huang. Evolving neural nets for fault diagnosis of power transformers. *IEEE Transactions on Power Delivery*, 18(3):843–848, July 2003.

Yann-Chang Huang, Hong-Tzer Yang, and Ching-Lien Huang. Developing a new transformer fault diagnosis system through evolutionary fuzzy logic. *IEEE Transactions on Power Delivery*, 12(2):761–767, Apr 1997.

IEC. guide to the interpretation of dissolved and free gases analysis. *IEC Std 60599-2007*, Feb 2007.

IEEE. guide for the interpretation of gases generated in oil-immersed transformers. *IEEE Std C57.104-2008*, pages 1–36, Feb 2009.

IEEE. guide for loading mineral-oil-immersed transformers and step-voltage regulators. *IEEE Std C57.91-2011 (Revision of IEEE Std C57.91-1995)*, pages 1–123, March 2012.

B. Igelnik and Yoh-Han Pao. Stochastic choice of basis functions in adaptive function approximation and the functional-link net. *IEEE Transactions on Neural Networks*, 6(6):1320–1329, Nov 1995.

Hazlee Azil Illias, Xin Rui Chai, Ab Halim Abu Bakar, and Hazlie Mokhlis. Transformer Incipient Fault Prediction Using Combined Artificial Neural Network and Various Particle Swarm Optimisation Techniques. *PLOS ONE*, 10(6):e0129363, June 2015.

Hazlee Azil Illias, Xin Rui Chai, and Ab Halim Abu Bakar. Hybrid modified evolutionary particle swarm optimisation-time varying acceleration coefficient-artificial neural network for power transformer fault diagnosis. *Measurement*, 90:94–102, August 2016.

S. M. Islam. Detection of shorted turns and winding movements in large power transformers using frequency response analysis. In *2000 IEEE Power Engineering Society Winter Meeting. Conference Proceedings (Cat. No.00CH37077)*, volume 3, pages 2233–2238 vol.3, Jan 2000.

A. G. Ivakhnenko. Polynomial theory of complex systems. *IEEE Transactions on Systems, Man, and Cybernetics*, 1(4):364–378, 1971.

Herbert Jaeger. The echo state approach to analysing and training recurrent neural networks-with an erratum note. *Bonn, Germany: German National Research Center for Information Technology GMD Technical Report*, 148:34, 2001.

J.-S.R. Jang. ANFIS: adaptive-network-based fuzzy inference system. *IEEE Transactions on Systems, Man, and Cybernetics*, 23(3):665–685, June 1993.

M. Jiang, Y.P. Luo, and S.Y. Yang. Stochastic convergence analysis and parameter selection of the standard particle swarm optimization algorithm. *Information Processing Letters*, 102(1):8 – 16, 2007.

Y. Jin and B. Sendhoff. Pareto-based multiobjective machine learning: An overview and case studies. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 38(3):397–415, May 2008.

Michael I. Jordan. Artificial neural networks. chapter Attractor Dynamics and Parallelism in a Connectionist Sequential Machine, pages 112–127. IEEE Press, Piscataway, NJ, USA, 1990.

M. D. Judd, S. D. J. McArthur, J. R. McDonald, and O. Farish. Intelligent condition monitoring and asset management. partial discharge monitoring for power transformers. *Power Engineering Journal*, 16(6):297–304, Dec 2002.

M. D. Judd, Li Yang, and I. B. B. Hunter. Partial discharge monitoring of power transformers using uhf sensors. part i: sensors and signal interpretation. *IEEE Electrical Insulation Magazine*, 21(2):5–14, March 2005.

A. Kavousi-Fard, A. Khosravi, and S. Nahavandi. A new fuzzy-based combined prediction interval for wind power forecasting. *IEEE Transactions on Power Systems*, 31(1):18–26, Jan 2016. doi: 10.1109/TPWRS.2015.2393880.

J.M. Keller, M.R. Gray, and J.A. Givens. A fuzzy K-nearest neighbor algorithm. *Systems, Man and Cybernetics, IEEE Transactions on*, SMC-15(4):580–585, July 1985.

J. J. Kelly. Transformer fault diagnosis by dissolved-gas analysis. *IEEE Transactions on Industry Applications*, IA-16(6):777–782, Nov 1980.

James Kennedy. Particle swarm optimization. In *Encyclopedia of machine learning*, pages 760–766. Springer, 2011.

Myoung-Jong Kim and Dae-Ki Kang. Ensemble with neural networks for bankruptcy prediction. *Expert Systems with Applications*, 37(4):3373–3379, April 2010.

Lawrence A Klein. *Sensor and data fusion: a tool for information assessment and decision making*. Spie Press Bellingham WA, 2004.

Joshua D Knowles and David W Corne. Approximating the nondominated front using the Pareto archived evolution strategy. *Evolutionary computation*, 8(2): 149–172, 2000.

Albert HR Ko, Robert Sabourin, and Alceu Souza Britto Jr. From dynamic classifier selection to dynamic ensemble selection. *Pattern Recognition*, 41(5): 1718–1731, 2008.

Nikolaos Kourentzes. Forecasting research. http://kourentzes.com/forecasting/, 2013.

Nikolaos Kourentzes, Devon K. Barrow, and Sven F. Crone. Neural network ensemble operators for time series forecasting. *Expert Systems with Applications*, 41(9):4235–4244, July 2014.

Anders Krogh, Jesper Vedelsby, et al. Neural network ensembles, cross validation, and active learning. *Advances in neural information processing systems*, 7:231–238, 1995.

Ludmila Kuncheva et al. Clustering-and-selection model for classifier combination. In *Knowledge-Based Intelligent Engineering Systems and Allied Technologies, 2000. Proceedings. Fourth International Conference on*, volume 1, pages 185–188. IEEE, 2000.

Ludmila I Kuncheva and Christopher J Whitaker. Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy. *Machine learning*, 51(2):181–207, 2003.

Kevin J. Lang, Alex H. Waibel, and Geoffrey E. Hinton. A time-delay neural network architecture for isolated word recognition. *Neural Networks*, 3(1):23 – 43, 1990.

B Lariviere and D Vandenpoel. Predicting customer retention and profitability by using random forests and regression forests techniques. *Expert Systems with Applications*, 29(2):472–484, August 2005.

Wen-Fung Leong and Gary G Yen. PSO-based multiobjective optimization with dynamic population size and adaptive local archives. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, 38(5):1270–1293, 2008.

Henry Leung and Simon Haykin. The complex backpropagation algorithm. *IEEE Transactions on Signal Processing*, 39(9):2101–2104, 1991.

D. Li, M. Han, and J. Wang. Chaotic time series prediction based on a novel robust echo state network. *IEEE Transactions on Neural Networks and Learning Systems*, 23(5):787–799, May 2012.

Tao Li, Chengliang Zhang, and Mitsunori Ogihara. A comparative study of feature selection and multiclass classification methods for tissue classification based on gene expression. *Bioinformatics*, 20(15):2429–2437, 2004.

Ruijin Liao, Jiaming Yan, Lijun Yang, Mengzhao Zhu, and Bin Liu. Study on the relationship between damage of oil-impregnated insulation paper and evolution of phase-resolved partial discharge patterns. *European Transactions on Electrical Power*, 21(8):2112–2124, 2011a.

Ruijin Liao, Hanbo Zheng, Stanislaw Grzybowski, and Lijun Yang. Particle swarm optimization-least squares support vector regression based forecasting model on dissolved gases in oil-filled power transformers. *Electric Power Systems Research*, 81(12):2074–2080, December 2011b.

P. Lim, C. K. Goh, K. C. Tan, and P. Dutta. Multimodal degradation prognostics based on switching kalman filter ensemble. *IEEE Transactions on Neural Networks and Learning Systems*, 28(1):136–148, Jan 2017.

C. E. Lin, J. M. Ling, and C. L. Huang. An expert system for transformer fault diagnosis using dissolved gas analysis. *IEEE Transactions on Power Delivery*, 8(1):231–238, Jan 1993.

Huan Liu and Hiroshi Motoda. *Computational methods of feature selection*. CRC Press, 2007.

L. E. Lundgaard. Partial discharge. xiv. acoustic partial discharge detection-practical application. *IEEE Electrical Insulation Magazine*, 8(5):34–43, Sept 1992.

H. Ma, T. K. Saha, C. Ekanayake, and D. Martin. Smart transformer for smart grid–intelligent framework and techniques for power transformer asset management. *IEEE Transactions on Smart Grid*, 6(2):1026–1034, March 2015.

Bjoern Menze and Nico Splitthoff. *obliqueRF: Oblique Random Forests from Recursive Linear Model Splits*, 2012. R package version 0.3.

Bjoern H Menze, B Michael Kelm, Daniel N Splitthoff, Ullrich Koethe, and Fred A Hamprecht. On oblique random forests. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 453–469. Springer, 2011.

K. Miners. Particles and moisture effect on dielectric strength of transformer oil using vde electrodes. *IEEE Transactions on Power Apparatus and Systems*, PAS-101(3):751–756, 1982.

V. Miranda and A. R. G. Castro. Improving the IEC table for transformer failure diagnosis with knowledge extraction from neural networks. *IEEE Transactions on Power Delivery*, 20(4):2509–2516, Oct 2005.

A. Miranian and M. Abdollahzade. Developing a local least-squares support vector machines-based neuro-fuzzy model for nonlinear and chaotic time series prediction. *IEEE Transactions on Neural Networks and Learning Systems*, 24(2): 207–218, Feb 2013.

P. Mirowski and Y. LeCun. Statistical machine learning and dissolved gas analysis: A review. *Power Delivery, IEEE Transactions on*, 27(4):1791–1799, Oct 2012.

P. Mohapatra, S. Chakravarty, and P.K. Dash. Microarray medical data classification using kernel ridge regression and modified cat swarm optimization based gene selection system. *Swarm and Evolutionary Computation*, 28:144 – 160, 2016.

D. R. Morais and J. G. Rolim. A hybrid tool for detection of incipient faults in transformers based on the dissolved gas analysis of insulating oil. *IEEE Transactions on Power Delivery*, 21(2):673–680, apr 2006.

P. H. F. Morshuis. Degradation of solid dielectrics due to internal partial discharge: some thoughts on progress made and where to go now. *IEEE Transactions on Dielectrics and Electrical Insulation*, 12(5):905–913, Oct 2005.

Sanaz Mostaghim and Jürgen Teich. Strategies for finding good local guides in multi-objective particle swarm optimization (MOPSO). In *Swarm Intelligence Symposium, 2003. SIS'03. Proceedings of the 2003 IEEE*, pages 26–33. IEEE, 2003.

Sanaz Mostaghim and Jürgen Teich. Covering Pareto-optimal fronts by subswarms in multi-objective particle swarm optimization. In *Evolutionary Computation, 2004. CEC2004. Congress on*, volume 2, pages 1404–1411. IEEE, 2004.

N. A. Muhamad, B. T. Phung, T. R. Blackburn, and K. X. Lai. Comparative study and analysis of dga methods for transformer mineral oil. In *2007 IEEE Lausanne Power Tech*, pages 45–50, July 2007.

Kevin P Murphy. *Machine learning: a probabilistic perspective.* MIT Press, 2012.

S. A. Mahmood Najafi, A. Peimankar, H. Saadati, E. Gockenbach, and H. Borsi. The influence of corona near to the bushing of a transformer on partial discharge measurement with an acoustic emission sensor. In *2013 IEEE Electrical Insulation Conference (EIC)*, pages 295–298, June 2013.

P.J. Garca Nieto, E. Garca-Gonzalo, F. Snchez Lasheras, and F.J. de Cos Juez. Hybrid PSO–SVM-based method for forecasting of the remaining useful life for aircraft engines and evaluation of its reliability. *Reliability Engineering & System Safety*, 138:219 – 231, 2015.

Rutuparna Panda, Manoj Kumar Naik, and B.K. Panigrah. Face recognition using bacterial foraging strategy. *Swarm and Evolutionary Computation*, 1(3):138 – 146, 2011.

Yoh-Han Pao and Stephen M. Phillips. The functional link net and learning optimal control. *Neurocomputing*, 9(2):149 – 164, 1995. Control and Robotics, Part {II}.

Yoh-Han Pao, Stephen M. Phillips, and Dejan J. Sobajic. Neural-net computing and the intelligent control of systems. *International Journal of Control*, 56(2): 263–289, 1992a.

Yoh-Han Pao, Stephen M Phillips, and Dejan J Sobajic. Neural-net computing and the intelligent control of systems. *International Journal of Control*, 56(2): 263–289, August 1992b.

Yoh-Han Pao, Gwang-Hoon Park, and Dejan J. Sobajic. Learning and generalization characteristics of the random vector functional-link net. *Neurocomputing*, 6(2):163–180, April 1994.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

A. Peimankar and A.C. Lapthorn. Condition assessment of transformers load tap changers using support vector machine. In *Proceedings of the Nintheenth International Symposium on High Voltage Engineering, Pilsen, Czech Republic, August 23–28*, 2015.

A. Peimankar, S. J. Weddell, T. Jalal, and A. C. Lapthorn. Ensemble classifier selection using multi-objective PSO for fault diagnosis of power transformers. In *2016 IEEE Congress on Evolutionary Computation (CEC)*, pages 3622–3629, July 2016.

A. Peimankar, S. J. Weddell, T. Jalal, and A. C. Lapthorn. Evolutionary multi-objective fault diagnosis of power transformers. *Swarm and Evolutionary Computation*, in press.

R. Polikar. Ensemble based systems in decision making. *IEEE Circuits and Systems Magazine*, 6(3):21–45, 2006.

X. Qiu, L. Zhang, Y. Ren, P. N. Suganthan, and G. Amaratunga. Ensemble deep learning for regression and time series forecasting. In *2014 IEEE Symposium on Computational Intelligence in Ensemble Learning (CIEL)*, pages 1–6, Dec 2014.

H. Quan, D. Srinivasan, and A. Khosravi. Short-term load and wind power forecasting using neural network-based prediction intervals. *IEEE Transactions on Neural Networks and Learning Systems*, 25(2):303–315, Feb 2014.

R Core Team. *R: A Language and Environment for Statistical Computing.* R Foundation for Statistical Computing, Vienna, Austria, 2013.

Y. Ren, L. Zhang, and P. N. Suganthan. Ensemble classification and regression-recent developments, applications and future directions. *IEEE Computational Intelligence Magazine*, 11(1):41–53, February 2016.

Jason D Rennie, Lawrence Shih, Jaime Teevan, David R Karger, et al. Tackling the poor assumptions of naive bayes text classifiers. In *ICML*, volume 3, pages 616–623. Washington DC), 2003.

Irina Rish. An empirical study of the naive bayes classifier. In *IJCAI 2001 workshop on empirical methods in artificial intelligence*, volume 3, pages 41–46. IBM New York, 2001.

E. Rivas, J. C. Burgos, and J. C. Garcia-Prada. Condition assessment of power oltc by vibration analysis using wavelet transform. *IEEE Transactions on Power Delivery*, 24(2):687–694, April 2009.

David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. Learning internal representations by error propagation. Technical report, DTIC Document, 1985.

T. K. Saha. Review of modern diagnostic techniques for assessing insulation condition in aged transformers. *IEEE Transactions on Dielectrics and Electrical Insulation*, 10(5):903–917, October 2003.

DVSS Siva Sarma and GNS Kalyani. ANN approach for condition monitoring of power transformers using DGA. In *TENCON 2004. 2004 IEEE Region 10 Conference*, volume 100, pages 444–447. IEEE, 2004.

Bernhard Schölkopf and Alexander J Smola. *Learning with kernels: support vector machines, regularization, optimization, and beyond*. MIT press, 2002.

Matthias Scholz. Nonlinear PCA. http://nlpca.org/, 2014.

Matthias Scholz, Fatma Kaplan, Charles L. Guy, Joachim Kopka, and Joachim Selbig. Non-linear PCA: a missing data approach. *Bioinformatics*, 21(20):3887–3895, 2005.

Glenn Shafer et al. *A mathematical theory of evidence*, volume 1. Princeton university press Princeton, 1976.

Ji Shengchang, Shan Ping, Li Yanming, Xu Dake, and Cao Junling. The vibration measuring system for monitoring core and winding condition of power transformer. In *Proceedings of 2001 International Symposium on Electrical Insulating Materials (ISEIM 2001). 2001 Asian Conference on Electrical Insulating Diagnosis (ACEID 2001). 33rd Symposium on Electrical and Ele*, pages 849–852, 2001.

Xi-L. Weng M. Shi, L. and J. Qian. A novel ensemble algorithm for biomedical classification based on ant colony optimization. *Applied Soft Computing*, 11(8):5674–5683, Dec 2011.

Yuhui Shi et al. Particle swarm optimization: developments, applications and resources. In *evolutionary computation, 2001. Proceedings of the 2001 Congress on*, volume 1, pages 81–86. IEEE, 2001.

A. Shintemirov, W. Tang, and Q. H. Wu. Power transformer fault classification based on dissolved gas analysis by implementing bootstrap and genetic programming. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 39(1):69–79, jan 2009.

Bernard W Silverman. *Density estimation for statistics and data analysis*, volume 26. CRC press, 1986.

S. Singh and M. N. Bandyopadhyay. Dissolved gas analysis technique for incipient fault diagnosis in power transformers: A bibliographic survey. *IEEE Electrical Insulation Magazine*, 26(6):41–46, November 2010.

Seifeddine Souahlia, Khmais Bacha, and Abdelkader Chaari. MLP neural network-based decision for power transformers fault diagnosis using an improved combination of Rogers and Doernenburg ratios DGA. *International Journal of Electrical Power & Energy Systems*, 43(1):1346–1353, dec 2012.

M. Srinivas and L. M. Patnaik. Genetic algorithms: a survey. *Computer*, 27(6): 17–26, June 1994.

Charles J Stone. Consistent nonparametric regression. *The annals of statistics*, pages 595–620, 1977.

S. M. Strachan, S. D. J. McArthur, M. D. Judd, and J. R. McDonald. Incremental knowledge-based partial discharge diagnosis in oil-filled power transformers. In *Proceedings of the 13th International Conference on, Intelligent Systems Application to Power Systems*, pages 6 pp.–, Nov 2005.

Q. Su, C. Mi, L. L. Lai, and P. Austin. A fuzzy dissolved gas analysis method for the diagnosis of multiple incipient faults in a transformer. *IEEE Transactions on Power Systems*, 15(2):593–598, May 2000.

S. B. Taieb and A. F. Atiya. A bias and variance analysis for multistep-ahead time series forecasting. *IEEE Transactions on Neural Networks and Learning Systems*, 27(1):62–76, Jan 2016.

W. H. Tang, J. Y. Goulermas, Q. H. Wu, Z. J. Richardson, and J. Fitch. A Probabilistic Classifier for Transformer Dissolved Gas Analysis With a Particle Swarm Optimizer. *IEEE Transactions on Power Delivery*, 23(2):751–759, apr 2008.

K. Tomsovic, M. Tapper, and T. Ingvarsson. A fuzzy information approach to integrating different transformer diagnostic methods. *IEEE Transactions on Power Delivery*, 8(3):1638–1646, July 1993.

Electrical Treeing. Electrical breakdown in high voltage insulation systems, Department of Engineering, University of Leicester. http://www2.le.ac.uk/departments/engineering/research/electrical-power/electrical-insulation-and-dielectric-phenomena.

Praveen Kumar Tripathi, Sanghamitra Bandyopadhyay, and Sankar Kumar Pal. Multi-objective particle swarm optimization with time variant inertia and acceleration coefficients. *Information Sciences*, 177(22):5033 – 5049, 2007.

N. Ueda and R. Nakano. Generalization error of ensemble estimators. In *1996 IEEE International Conference on Neural Networks*, volume 1, pages 90–95, June 1996.

F. van den Bergh and A.P. Engelbrecht. A study of particle swarm optimization particle trajectories. *Information Sciences*, 176(8):937 – 971, 2006.

O. Vanegas, Y. Mizuno, K. Naito, and T. Kamiya. Diagnosis of oil-insulated power apparatus by using neural network simulation. *IEEE Transactions on Dielectrics and Electrical Insulation*, 4(3):290–299, Jun 1997.

B. Wang and H. D. Chiang. Elite: Ensemble of optimal input-pruned neural networks using trust-tech. *IEEE Transactions on Neural Networks*, 22(1):96–109, Jan 2011.

M. Wang, A. J. Vandermaar, and K. D. Srivastava. Review of condition assessment of power transformers in service. *IEEE Electrical Insulation Magazine*, 18(6): 12–25, November 2002.

M. Wang, A. J. Vandermaar, and K. D. Srivastava. Improved detection of power transformer winding movement by extending the fra high frequency range. *IEEE Transactions on Power Delivery*, 20(3):1930–1938, July 2005.

M. H. Wang. Grey-extension method for incipient fault forecasting of oil-immersed power transformer. *Electric Power Components and Systems*, 32(10):959–975, October 2004.

Yujia Wang and Yupu Yang. Particle swarm optimization with preference order ranking for multi-objective optimization. *Information Sciences*, 179(12):1944 – 1959, 2009. Special Section: Web Search.

Z. Wang, Y. Liu, and P. J. Griffin. A combined ann and expert system tool for transformer fault diagnosis. In *Power Engineering Society 1999 Winter Meeting, IEEE*, volume 1, pages 339–347 vol.1, Jan 1999.

Zhenyuan Wang. *Artificial intelligence applications in the diagnosis of power transformer incipient faults.* PhD thesis, Virginia Polytechnic Institute and State University, 2000.

Sheng wei Fei and Xiao bin Zhang. Fault diagnosis of power transformer based on support vector machine with genetic algorithm. *Expert Systems with Applications*, 36(8):11352 – 11357, 2009.

Sheng wei Fei, Ming-Jun Wang, Yu bin Miao, Jun Tu, and Cheng liang Liu. Particle swarm optimization-based support vector machine for forecasting dissolved gases content in power transformer oil. *Energy Conversion and Management*, 50(6):1604 – 1609, 2009.

Dellana S. West, D. and J. Qian. Neural network ensemble strategies for financial decision applications. *Computers and Operations Research*, 32(10):2543–2559, Oct 2005.

K. Woods, K. Bowyer, and Jr. Kegelmeyer, W.P. Combination of multiple classifiers using local accuracy estimates. In *Computer Vision and Pattern Recognition, 1996. Proceedings CVPR '96, 1996 IEEE Computer Society Conference on*, pages 391–396, Jun 1996.

W. Yan. Toward automatic time-series forecasting using neural networks. *IEEE Transactions on Neural Networks and Learning Systems*, 23(7):1028–1039, July 2012.

Liying Yang. Classifiers selection for ensemble learning based on accuracy and diversity. *Procedia Engineering*, 15:4266–4270, 2011.

M. F. M. Yousof, C. Ekanayake, and T. K. Saha. Examining the ageing of transformer insulation using fra and fds techniques. *IEEE Transactions on Dielectrics and Electrical Insulation*, 22(2):1258–1265, April 2015a.

M. F. M. Yousof, C. Ekanayake, and T. K. Saha. Frequency response analysis to investigate deformation of transformer winding. *IEEE Transactions on Dielectrics and Electrical Insulation*, 22(4):2359–2367, August 2015b.

W. S. Zaengl. Dielectric spectroscopy in time and frequency domain for hv power equipment. i. theoretical considerations. *IEEE Electrical Insulation Magazine*, 19(5):5–19, Sept 2003.

A. Zargari and T. R. Blackburn. Acoustic detection of partial discharges using non-intrusive optical fibre sensors [current transformers]. In *Conduction and Breakdown in Solid Dielectrics, 1998. ICSD '98. Proceedings of the 1998 IEEE 6th International Conference on*, pages 573–576, Jun 1998.

C. Zhang, P. Lim, A. K. Qin, and K. C. Tan. Multiobjective deep belief networks ensemble for remaining useful life estimation in prognostics. *IEEE Transactions on Neural Networks and Learning Systems*, PP(99):1–13, 2016.

Le Zhang and P.N. Suganthan. A comprehensive evaluation of random vector functional link networks. *Information Sciences*, 367-368:1094–1105, November 2016a.

Le Zhang and P.N. Suganthan. A survey of randomized algorithms for training neural networks. *Information Sciences*, 364-365:146–155, October 2016b.

Le Zhang and Ponnuthurai N. Suganthan. Oblique Decision Tree Ensemble via Multisurface Proximal Support Vector Machine. *IEEE Transactions on Cybernetics*, 45(10):2165–2176, October 2015.

Q. Zhang and H. Li. Moea/d: A multiobjective evolutionary algorithm based on decomposition. *IEEE Transactions on Evolutionary Computation*, 11(6):712–731, Dec 2007.

X. Zhang and E. Gockenbach. Asset-Management of Transformers Based on Condition Monitoring and Standard Diagnosis [Feature Article]. *IEEE Electrical Insulation Magazine*, 24(4):26–40, July 2008.

Y. Zhang, X. Ding, Y. Liu, and P. J. Griffin. An artificial neural network approach to transformer fault diagnosis. *IEEE Transactions on Power Delivery*, 11(4): 1836–1841, Oct 1996.

S. Z. Zhao and P. N. Suganthan. Two-*lbests* based multi-objective particle swarm optimizer. *Engineering Optimization*, 43(1):1–17, 2011.

Eckart Zitzler and Lothar Thiele. Multiobjective evolutionary algorithms: a comparative case study and the strength pareto approach. *IEEE transactions on Evolutionary Computation*, 3(4):257–271, 1999.

Eckart Zitzler, Marco Laumanns, and Lothar Thiele. SPEA2: Improving the strength pareto evolutionary algorithm. Technical report, 2001.