DOCTORAL THESIS

Failure Diagnostics Using Support Vector Machine

Yuan Fuqing

LULEÅ UNIVERSITY OF TECHNOLOGY

DOCTORAL THESIS

Failure Diagnostics using Support Vector Machine

YUAN FUQING

Luleå University of Technology
Division of Operation and Maintenance Engineering

Printed by Universitetstryckeriet, Luleå 2011

ISSN: 1402-1544
ISBN 978-91-7439-366-8

Luleå 2011

www.ltu.se

PREFACE

Support Vector Machine (SVM) is a multidisciplinary technique which includes mathematics and computer science. The research on SVM covers statistics, functional analysis, matrix theory, programming, algorithm design, and pattern recognition. Combining such subjects is challenging. Philosophically, the aim of such research is simple: how to use available knowledge to predict a future event. It assumes the available knowledge holds information on the future and the future is predictable.

The research on the SVM is progressing very quickly and the new advances of SVM have gone far beyond its initial. SVM is closely related to machine learning, a hot topic in the field of information technology. My research on SVM is not for IT purposes but for failure diagnostics. IT purposes focus more on fast training algorithms but failure diagnostics focuses more on accuracy. Using a technique without knowing the principle is risky, especially in the case where safety has a high priority. Therefore, in this thesis lots efforts have been put on SVM's theoretical foundation.

My research is ongoing on but I have to stop temporarily to write this thesis. I would like to express my gratitude to my supervisor Professor Uday Kumar for giving me the opportunity to pursue my doctorate; I greatly appreciate his guidance and help. Thanks also to assistant supervisor Diego Galar for his willingness to discuss a topic in which we are both interested. Thanks to Professor Krishna B. Misra for his suggestions, guidance and especially his faith in me. I am grateful to Trafikverket for supporting this research. I would also like to thank Aditya, Ramin, and Bezhad for their supervision in the first year of my Ph.D. I appreciate the help of Ali, Rajiv, Stephen, Andi and all other colleagues in the division of operation and maintenance. Thanks to Mr.Xiao, Mr.Dong. And finally, I have to thank my wife Dr. Lu Jinmei for her support and her cooking.

ABSTRACT

Failure diagnostics is an important part of condition monitoring aiming to identify existing or impending failures. Accurate and efficient failure diagnostics can guarantee that the operator makes the correct maintenance decision, thereby reducing the maintenance costs and improving system availability. The Support Vector Machine (SVM) is discussed in this thesis with the purpose of accurate and efficient failure diagnostics.

The SVM utilizes the kernel method to transform input data from a lower dimensional space to a higher dimensional space. In the higher dimensional space, the hitherto linearly non separable patterns can be linearly separated, without compromising the computational cost. This facilitates failure diagnostics as in the higher dimensional space, the existing failure or incipient failure is more identifiable. The SVM uses the maximal margin method to overcome the "overfitting" problem. This problem makes the model fit special data sets. The maximal margin method also makes it suitable for solving small sample size problems.

In this thesis, the SVM is compared with another well known technique, the Artificial Neural Network (ANN). In the comparative study, the SVM performs better than the ANN. However, as the performance of the SVM critically depends on the parameters of the kernel function, this thesis proposes using an Ant Colony Optimization (ACO) method to obtain the optimal parameters. The ACO optimized SVM is applied to diagnose the electric motor in a railway system. The Support Vector Regression (SVR) is an extension of the SVM. In this thesis, SVR is combined with a time-series to forecast reliability. Finally, to improve the SVM performance, the thesis proposes a multiple kernel SVM.

The SVM is an excellent pattern recognition technique. However, to obtain an accurate diagnostics performance, one has to extract the appropriate features. This thesis discusses the features extracted from the time domain and uses the SVM to diagnose failure for a bearing. Another case in this thesis is presented, namely failure diagnostics for an electric motor installed in a railway's crossing and switching system; in this case, the features are extracted from the power consumption signal.

In short, the thesis discusses the use of the SVM in failure diagnostics. Theoretically, the SVM is an excellent classifier or regressor possessing a solid theoretical foundation. Practically, the SVM performs well in failure diagnostics, as shown in the cases presented. Finally, as failure diagnostics critically relies on feature extraction, this thesis considers feature extraction from the time domain.

Keywords: *Support Vector Machine; Failure Diagnostics; Neural Network; Kernel method; Multi-kernel Support vector machine; Time Domain; Feature Extraction; Kernel Parameter Optimization*

LIST OF APPENDED PAPERS

PAPER I: **Y. Fuqing**, U. Kumar and D. Galar, "Reliability Prediction using Support Vector Regression," *International Journal of Systems Assurance Engineering and Management,* vol. 1, No. 3 pp. 263-268, 2010.

PAPER II: **Y. Fuqing**, U. Kumar and D. Galar, "Fault Diagnosis of Railway Assets using Support Vector Machine and Ant Colony Optimization Method," *International journal of COMADEM,* (Accepted for Publication).

PAPER III: **Y. Fuqing**, U. Kumar and D. Galar, "An Adaptive Multiple-kernel Method based Support Vector Machine for Classification," *International Journal of Condition Monitoring*.(Submitted)

PAPER IV: **Y. Fuqing**, U. Kumar and D. Galar, "A Comparative Study of Artificial Neural Networks and Support Vector Machine for Fault Diagnosis," presented at the CM 2011 and MFPT 2011, Cardiff, UK, 2011. Improved version has been submitted to *International Journal of Performability Engineering.*

PAPER V: **Y. Fuqing**, U. Kumar and D. Galar, " Fault Diagnosis on time domain for Rolling Element Bearings using Support Vector Machine," *Reliability Engineering & System Safety*.(Submitted)

LIST OF RELATED PUBLICATIONS

[1] **Y. Fuqing** and U. Kumar, "A General Imperfect Repair Model Considering Time-Dependent Repair Effectiveness," *IEEE Transactions on Reliability*, 2012. 03 (Accepted).

[2] **Y. Fuqing** and U. Kumar, "A Cost Model for Repairable System Considering Multi-failure types over Finite Time Horizon," *International Journal of Performability Engineering*, vol. 7, pp. 121-129, 2011.

[3] **Y. Fuqing** and U. Kumar, "Complex System Reliability Evaluation using Support Vector Machine for Incomplete Data-set," *International Journal of Performability Engineering*, vol. 7, pp. 32-42, 2011.

[4] **Y. Fuqing** and U. Kumar, "Kernelized Proportional Intensity Model for Repairable System considering Piecewise Operating Condition," *IEEE Transactions on Reliability*.(Second Revision)

[5] **Y. Fuqing**,U.Kumar, C. Rocco, K.B.Misra., "Complex System Reliability Evaluation using Support Vector Machine," presented at the SMRLO10, Israel, 2009.

[6] **Y. Fuqing** and U. Kumar, "Replacement policy for repairable system under various failure types with finite time horizon," presented at the MMR2009, Moscow, 2009.

[7] **Y. Fuqing** and U. Kumar, "Predicting Time to Failure using Support Vector Regression," presented at the E-Maintenance 2010, Luleå, 2010

CONTENTS

1. Introduction

1.1 Background

No matter how well a system is designed, products deteriorate over time, since they are operating under stress or loads in the real environment, often involving randomness (Jardine *et al.*, 2006). Therefore, proper maintenance is necessary to sustain the system at a satisfactory level. Maintenance is defined as the combination of all the technical and administrative actions, including supervisory actions, intended to retain an item in, or restore it to, a state where it can perform a required function (BSI, 1984). Maintenance increases the life length and reduces the number of failures and degradation rate.

Maintenance can be categorized as corrective maintenance, scheduled maintenance and Condition Based Maintenance (CBM) (De Silva, 2005, Martin, 1994), as illustrated in Figure 1.1. Corrective maintenance is a strategy whereby maintenance, in the form of repair work or replacement, is only performed when machinery has failed. Scheduled maintenance is undertaken when specific maintenance tasks are performed at set time intervals in order to maintain a significant margin between machine capacity and actual duty. CBM is a maintenance program that recommends maintenance actions based on the information collected through Condition Monitoring.

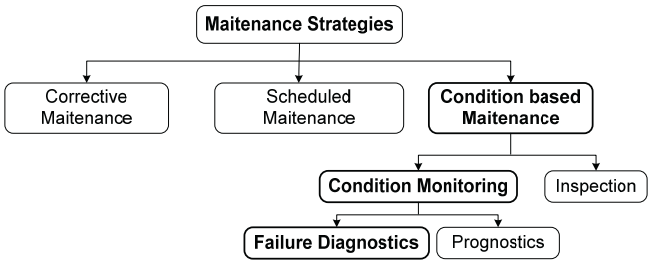


Figure 1.1 Maintenance Strategies

Corrective maintenance is undertaken in situations where the failure consequence is not serious and a quick repair or replacement is possible. Scheduled maintenance is carried out with fixed time intervals regardless of the real machine condition. This maintenance strategy leads to unnecessary maintenance, making it an expensive maintenance strategy; nevertheless, the maintenance interval can be optimized by analyzing its reliability (De Silva, 2005, Barabady and Kumar, 2008, Kumar *et al.*, 1989). CBM attempts to avoid unnecessary maintenance tasks by performing maintenance actions only when there is evidence of abnormal behaviour of a physical asset occurred. Properly implemented CBM can significantly reduce maintenance costs by reducing unnecessary scheduled preventive maintenance operations (Jardine *et al.*, 2006).

CBM can be based on condition monitoring or on the results of regular inspections. Condition monitoring is defined as a technique or a process of monitoring the operating characteristics of a machine in such a way that changes and trends in the monitored characteristics can be used to predict the need for maintenance before serious deterioration or breakdown occurs and/or to estimate the machine's "health" (Han and Song, 2003).

Failure diagnostics is an important aspect of condition monitoring as it determines the state of the system (faulty or normal) as well as the type of faults (Akbaryan and Bishnoi, 2001). Failure diagnostics may find incipient failures, so that action can be taken before a catastrophic failure occurs. Recently, researchers have focused on developing an effective and efficient failure diagnostics method; in this state-of-the-art research, failure diagnostics methods have been devised, improved upon, or adopted from other fields.

1.2 Failure Diagnostics Techniques

Failure diagnostics techniques include analytical, knowledge based and data driven models. As the analytical model, also called model-based, requires the full understanding of interactions inside machines, it is machine specific. Knowledge based models use expert domain knowledge in a computer program with an automated inference engine to perform reasoning (Jardine *et al.*, 2006, Ebersbach and Peng, 2008). The knowledge based model can therefore be categorized as rule-based, case-based and model-based, as illustrated in Figure 1.2 (Saunders *et al.*, 2000). Data driven models diagnose the failure from the available data, including condition monitoring and operating data. The dependency between the machine condition and the available data is quantified by using probability, statistical or self-learning methods. The data driven model can be further categorized as a probability and statistics-based model or a non-probability and statistics-based model.

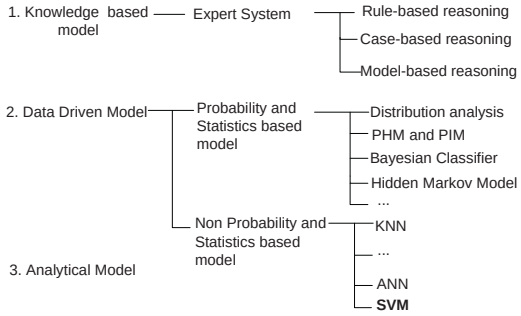


Figure 1.2 Failure Diagnostics Techniques

In failure distribution analysis, the probability and statistics-based method assumes the failure of a system is random and follows a special statistical distribution, e.g. Weibull distribution (Barlow and Proschan, 1965). The parameters in the distribution are evaluated from the observed data. The accuracy of the assumed distribution can be

checked by goodness-of-fit tests or graph methods (Klefsjo and Kumar, 1992). The Proportional Hazard Models (PHM) and the Proportional Intensity Model (PIM) are other statistical models (Cox, 1972, Lawless, 1987, Klefsjo and Kumar, 1992, Kumar, 1995). Both consider the condition monitoring measurement as a covariant and can evaluate the dependency between reliability and the covariant (Jardine, 2001, Jardine *et al.*, 1997, Jardine *et al.*, 1999).

Some failure diagnostics methods use statistical pattern recognition techniques. An important example is the Bayesian Classifier (Theodoridis and Koutroumbas, 2006). This classifier models each variable using a specified distribution and estimates the conditioned probability by measuring the dependence between the variables and a specified failure type. The distribution can be univariate or multivariate depending on the dimension of the measurements. The Bayesian inference requires fewer data sets due to the incorporation of prior information.

The non-probability and statistics-based model identifies failure based on the geometric distance between data sets. This method or learning algorithm measures the similarity or dependency of distances between data sets. These include the Euclidean distance, Riemannian distance, Mahalanobis distance, or Kullback-Leibler distance (Jardine *et al.*, 2006). The K-Nearest Neighbor (KNN) algorithm is a typical Euclidean distance-based algorithm (Theodoridis and Koutroumbas, 2006). The kernel method used by the SVM is a Riemannian distance method (Amari and Wu, 1999). As the SVM, along with the Artificial Neural Network (ANN) possess self-learning ability, they are also called learning algorithms.

Fuzzy logic is also used in failure diagnostics. It is generally used to measure the uncertainty of rules and measurement inputted into knowledge based models or self learning models such as the ANN and the SVM (Hong and Hwang, 2003, Pfeufer and Ayoubi, 1997, Lin and Wang, 2002). And heuristic methods, such as the genetic or ant colony model, are mostly used in failure diagnostics to optimize parameters for the ANN or SVM models; for example, they are used to optimize the ANN structure, and used to find the optimal parameters for the SVM (Chen, 2007).

1.3 Learning Algorithms for Failure Diagnostics

A key issue in failure diagnostics is the ability to detect failures automatically, accurately and efficiently. High accuracy means fewer false alarms; this is important, as shutting down machines can be costly. Efficiency is especially important for online condition monitoring. A slow response to a newly changed situation will not allow early warning. Automation is required when the data are too large to be treated manually. A large data set results when many sensors are mounted to systems, as in complex modern systems like aircraft, spacecraft, and high speed trains. When Computerized Maintenance Management Systems (CMMS) are used, the data generated daily are huge. Fusing these maintenance data with condition monitoring data for failure diagnostics is a challenge. Inversely, there are situations when information is lacking, missing, or incomplete

(Fuqing *et al.*, 2011). Having insufficient information increases the risk of poor decision making. Reducing such risks is another challenge.

1.3.1 Artificial Neural Network for Failure Diagnostics

ANN is an artificial technique with self learning ability and can adapt themselves with the data automatically. The ANN has been extensively investigated for applications (Amari *et al.*, 1994, Cheng and Titterington, 1994), and it has numerous variants and extensions covering pattern recognition, forecasting, and function approximation (Kermit *et al.*, 2000, Hippert *et al.*, 2001, Kahraman and Oral, 2001, Maier and Dandy, 2000, Rowley *et al.*, 1998, Sugisaka and Fan, 2005). The ANN has been widely applied to failure diagnostics. Chen and Lee (2002) have proposed an ANN method to identify failure patterns for F-16 aircraft. Thomas *et al.* have proposed a hybrid of fuzzy logic and ANN to perform failure diagnostics (Pfeufer and Ayoubi, 1997). Castro *et al.* (2005b, 2005a, 2005c) have used the ANN to diagnose transformer failures. Tarng *et al.* (1994) have used it to diagnose milling failures. In spite of the ANN's wide applications and its popularity in academia, it is criticized for certain weaknesses, including its "overfitting" and the time-consuming training process (Tu, 1996, Theodoridis and Koutroumbas, 2006).

1.3.2 Support Vector Machine for Failure Diagnostics

The SVM is learning algorithm developed after the ANN (Shawe-Taylor and Cristianini, 2004, Vapnik, 1995, Vapnik, 1998). It claims it can prevent the ANN's "overfitting" problem. The SVM uses the kernel function to measure the similarities between data, and the decision function is represented by an expansion of the kernel function (Bennett and Campbell, 2000, Noble, 2006). The SVM has been extensively used for data classification and diagnostics in the medical sciences and bio technology (Li and Gui, 2004, Li and Luan, 2003, Noble, 2006). It is gradually finding application in condition monitoring for rolling element bearings, gear boxes, induction motors, machine tools, pumps, compressors, valves and turbines, engine knock, autonomous underwater vehicles and so on (Widodo and Yang, 2007). In such applications, the SVM is used as a data-driven classifier.

As a classifier, the SVM as is further divided into the following: one-class classifiers, binary classifiers and multi-class classifiers. The multi-class classifier is commonly used in state-of-the-art failure diagnostics. Sugumaran *et al.* (2008) use the multi-class SVM to diagnose failures in roller bearings; Widodo *et al.* (2009) use it to diagnose failures in low speed bearings; Yuan and Chu (2007) to diagnose failures in turbo-pump rotors; and Antonelli *et al.* (2004) to diagnose autonomous underwater vehicle failures. The one-class classifier is called novelty detection.

Onoda *et al.* (2008) use the SVM to detect unusual conditions in hydroelectric power plants by analyzing the temperature of the room, oil cooler and etc. and by analyzing the vibration signal from the generator shaft and turbine. Hayton *et al.* (2001) use the SVM to detect abnormal aspects of the vibration signature of jet engine vibration spectra. Finally, Davy *et al.* (2006) use the SVM to detect abnormal events online for gear boxes.

1.4 Problem Description

With the automation of the data acquisition process, a large amount of condition monitoring and maintenance data is collected, making it almost impossible to manually extract and analyze valuable maintenance knowledge. Learning algorithms like the Artificial Neural Networks (ANN) and Support Vector Machine (SVM) can be used to extract information efficiently. If they are properly implemented, accurate failure diagnostics can be performed based on maintenance and condition monitoring data.

This SVM implementation can overcome the "overfitting" problem, compared with the ANN and other failure diagnostics methods. However, it has two major problems: internal parameter selection and the time-consuming training involved with large scale data sets. The latter problem has been solved by using sequential minimal optimization (SMO) (Schölkopf *et al.*, 1999), but the former remains unsolved. This thesis investigates this problem and attempts to find solution to overcome problems associated with internal parameter selection.

1.5 Purpose of the Research

The main aim of the present research is to explore the suitability of the Support Vector Machine (SVM) for failure diagnostics using condition monitoring data in maintenance contexts and suggests improvement of internal parameter selection approach so as to improve failure diagnostics. The research also aims to evaluate the performance of the SVM in failure diagnostics and suggests improvement in standard SVM model.

1.6 Research Objectives

To fulfil the research purpose, the following objectives have been formulated:

a. Study the principle of SVM for its application in failure diagnostics context and suggest methods for improving the SVM so that it can be used in accurate and efficient failure diagnostics.
b. Develop a methodology to determine the optimal SVM parameter to achieve maximum accuracy and improved diagnostics.
c. Evaluate the performance of the SVM in failure diagnostics in terms of computational cost, complexity, accuracy and stability.

1.7 Scope and Limitations of the Study

This study covers the investigation of the theoretical foundation of support vector machine, applying the SVM to failure diagnostics. The study is performed on some railway assets. The limitations of the thesis relate to classical SVM. Other techniques for example combining the SVM with statistics is beyond the scope of the study.

1.8 Structure of the Thesis

This thesis consists of ten chapters and five appended papers. Its structure is illustrated in Figure 1.3.

Chapter 1 introduces the research, giving the background of the use of the support vector machine in reliability data analysis, especially failure diagnostics. It presents a brief survey of the literature on failure diagnostics. The chapter also discusses the problem existing, research purpose, the research objectives, and the scope and limitations.

Chapter 2 discusses the procedure used to implement failure diagnostics. It examines condition monitoring data acquisition and collection, as well as data processing techniques, such as feature extraction and selection. Failure diagnostics in railway system is discussed in the last section.

Chapter 3 looks at the method to induce the support vector machine using the maximal margin and the kernel method. It introduces the support vector classifier and support vector regression.

Chapter 4 discusses some basic concepts of the learning theory, looking closely at the generalization error. The chapter considers both distribution based bounds and distribution free based bounds. It looks at the capacity of admissible functions and suggests how to obtain a good generalization error bound.

Chapter 5 discusses the kernel method and shows how it can improve the performance of classification. It briefly describes a kernel function and provides a geometrical explanation. Some widely used kernel functions are described and the advantage of kernel method is summarized.

Chapter 6 discusses the application of the SVM to reliability data analysis and failure diagnostics. For each application, it presents a brief example of how the SVM can be used.

Chapter 7 summarizes the appended papers and highlights the important findings for appended papers.

Chapter 8 discusses the important issues on failure diagnostics using SVM and presents the suggested solution for each of these issues.

Chapter 9 presents summary of the findings from this research and give suggestions for implementing the SVM further.

Chapter 10 summarizes the research contributions of the thesis and presents the scope of future research within this filed.

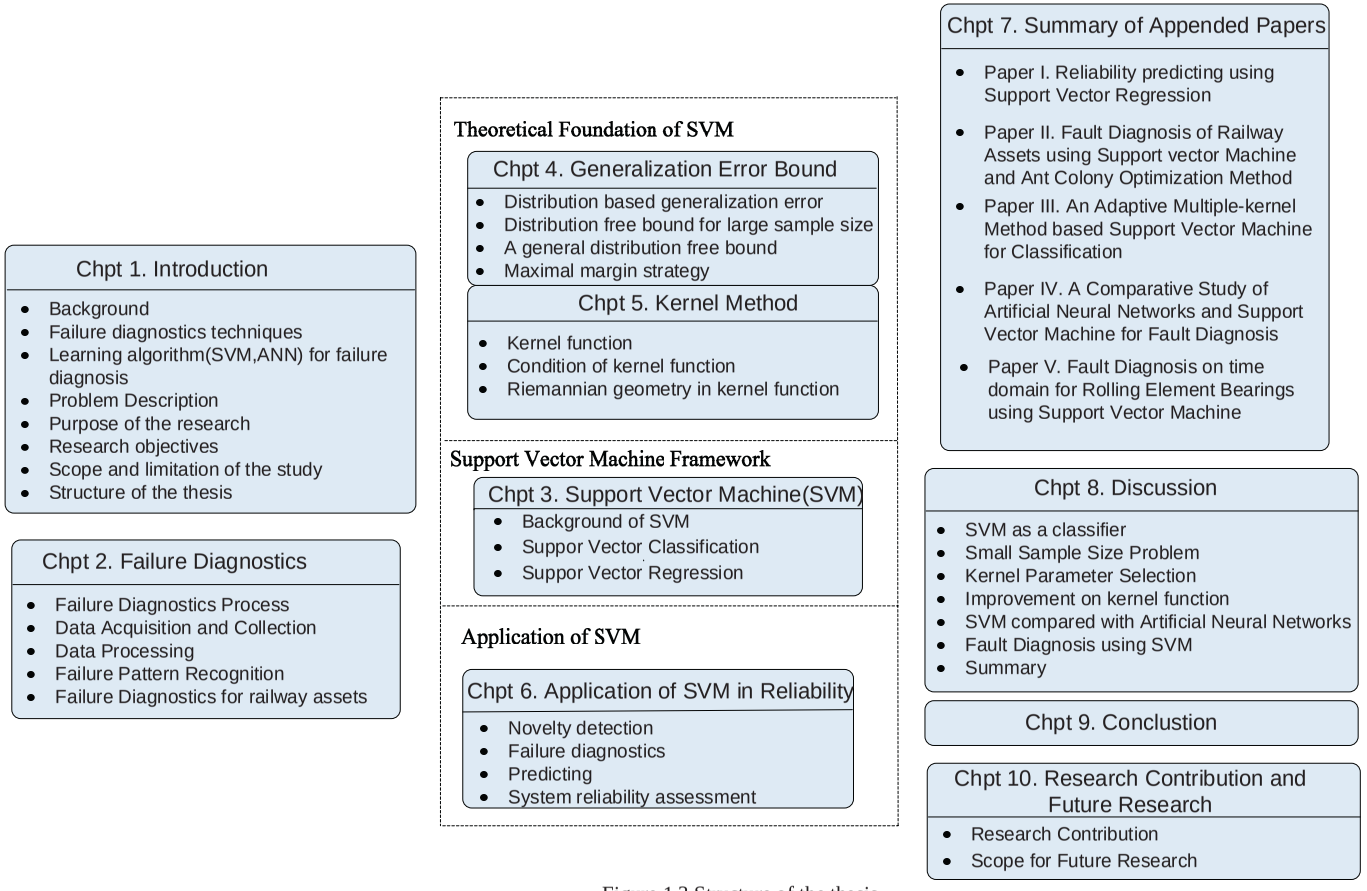


Figure 1.3 Structure of the thesis

2. Failure Diagnostics

An efficient and effective failure diagnostics can give an accurate early warning to the incipient failure, thus the maintenance strategy, spare parts, maintenance tools, personnel, and etc can be scheduled in advance, and the unplanned stoppage can be prevented due to maintenance action taken earlier.

2.1 Failure Diagnostics Process

Failure diagnostics methods vary dramatically according to the monitored system and the type of failure. Methods include vibration analysis, oil analysis, infrared analysis, current analysis, power analysis and so on. For rotary machinery, such as bearings and gears, failure diagnostics can be performed by analyzing the machine's vibration signal. For reciprocating machines, such as diesel engines and reciprocating compressors, the machine's cylinder pressure signal can be analyzed throughout a cycle. Electrical machines can be analyzed through their power consumption, while the analysis of electronic devices can draw on the machine's heat distribution. Despite the differences, all failure diagnostics consist of three main steps, as illustrated in Figure 2.1: data and signal acquisition and collection; data processing; and failure pattern recognition.

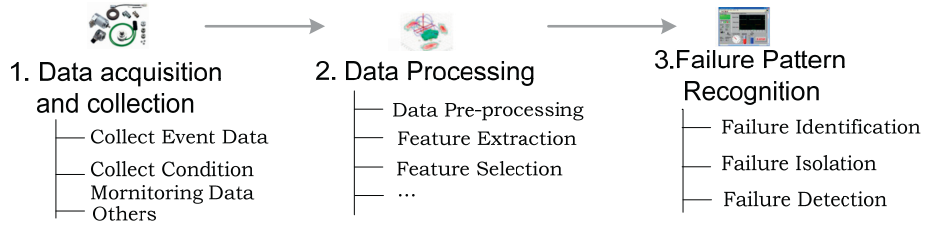


Figure 2.1 Failure Diagnostics Process

2.2 Data Acquisition and Collection

In data acquisition, data are collected from sensors mounted on the system. These include displacement sensors, velocity sensors, and accelerometer sensors. Each sensor measures a specified signal; sometimes several identical sensors are installed in various locations to measure the same signal to obtain the system's health information from several perspectives. The data collected from sensors are called condition monitoring data in this thesis.

Other data are probably available, such as historical failure data and manufacturer information, and these can help to diagnose failure. These data are commonly called event data. For example, the Swedish railway asset information system BIS and failure reporting system 0felia, as shown in Figure 2.2, are databases containing a huge amount of event data. BIS collects Switches and Crossings (S&C) data, including track section, S&C type, year put in place and so on. 0felia collects data on date and time of reported

failures, time of maintenance, failure symptoms and so on. Collecting as many data as possible can provide more system information for failure diagnostics.

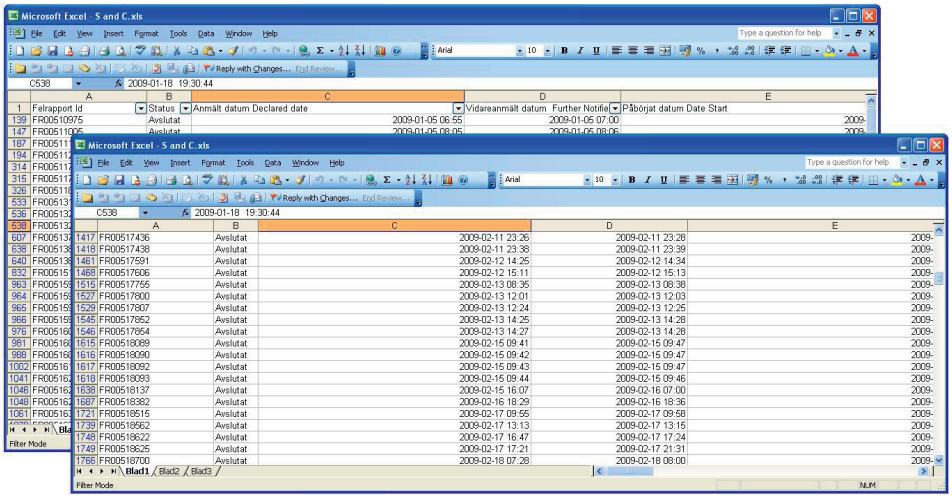


Figure 2.2 Event Data of S&C

2.3 Data Processing

2.3.1 Data Pre-Processing

The raw signal may contain noise or irrelevant signals. Eliminating noises or irrelevant signals is necessary for reliable failure diagnostics. Take the bearing for example. In the early failure stage, the noise signal is dominant, and performing failure analysis without de-noising will lead to a false alarm. Pre-processing a signal covers outlier removal, data normalization, noise removal and irrelevant signal removal.

2.3.2 Feature Extraction

Features are the individual measurable heuristic properties of the phenomena being observed. They are usually numeric, as for example, the mean, variance and peak of the series of a signal (Theodoridis and Koutroumbas, 2006). Feature extraction is the process of extracting features with understandable information about the health of the component (Theodoridis and Koutroumbas, 2006). Features can be extracted from the time-domain, the frequency domain, or other domains.

Time-domain features are used for non periodical signals or when the periodicity of a signal is not significant, for example, early stage bearing fault signals. Time domain features cover, for example, mean, variance, minimum, maximum, or polynomial coefficients of the signal (Mathew and Alfredson, 1984, Y.Kim *et al.*, 2007, B.Sreejith *et al.*, 2008, Zhang and Randall, 2009).

For the periodical signal, as in the defect signal of a bearing or gear, the feature can be extracted from the frequency domain, for example, by the Fast Fourier Transform (FFT) (Mathew and Alfredson, 1984). The amplitude of a frequency can be a feature (Theodoridis and Koutroumbas, 2006). However, the Fourier Transform is only suitable for transforming a stationary signal. For non-stationary signal the Short-time Fourier Transform (Zhu *et al.*, 2007, Griffin and Lim, 1984), or the Wavelet transform (Daubechies, 1990) can be used.

Feature extraction is domain specific and signal specific. To ensure the right features obtained, there are a variety of methods available to evaluate feature performance. The classical test statistics such as t-test, F-test, Chi-squared test and etc can be applied to test the performance of each individual feature (Theodoridis and Koutroumbas, 2006), and the relief algorithm is another classical method (Kira and Rendell, 1992). The disadvantage of these methods ignores correlation between features. Fortunately, there are some methods available to measure the cross-correlation between features which can be used to remove highly correlated features.

Another category for feature performance measure is correlation coefficients assessing the degree of dependence of individual variables with the target pattern. The Pearson correlation coefficient is a classical method of them which uses relevance index for individual feature (Guyon and Elisseeff, 2006). The feature separability can be also used to measure feature performance. Qiue and Joe (2006) defines a separability based on distance between features from diffident patterns. Other separability measures such as: Bayesian inference based divergence, Chernoff bound distance, Bhattacharyya distance, Fisher's Discriminant Ratio (FDR), can be used to measure feature performance as well (Theodoridis and Koutroumbas, 2006). These measures convey information on the discriminatory capability related to the features.

2.3.3 Feature Selection

Intuitively, extracting as many features as possible is always better, as more features can provide more information. However, the presence of irrelevant and redundant features complicates the diagnostics model, and increases the computational cost. Most importantly, having a large number of features could degrade the ability of the diagnostics model to generalize. For a finite number of data sets, a good model with good performance usually has a higher ratio between the data sets and number of features.

Figure 2.3 shows that performance does not always improve with an increased number of features (G.V.Trunk, 1979). In this scenario, increasing the number of features can only improve the performance initially, but after a critical number of features, the performance decreases. This is called the "peaking phenomenon" (Theodoridis and Koutroumbas, 2006). The figure also shows that only for infinite data sets or sufficiently large data sets, increasing the number of features can improve the performance of the diagnostics model. But creating infinite data sets or even sufficiently large data sets is not possible in most situations.

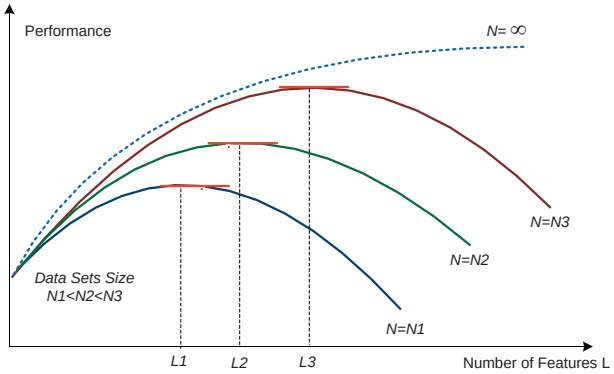


Figure 2.3 Peaking Phenomenon

Feature selection to reduce the number of features to a sufficient level is necessary to improve the model performance. There are two general methods of feature selection, although removing the irrelevant and redundant features depends on the specified problem. These methods can be categorized as individual feature selection and subset feature selection. In individual feature selection, each feature can be ranked according to its importance, and the less important features can be removed (Yu and Liu, 2004). For the SVM, each feature can be weighted in the input space, and these weights can be evaluated during the training process (Nguyen and de la Torre, 2010). The less important weights will have smaller or zero weight so their influence can be weakened or removed.

Methods of subset features selection search for a minimum subset of features that satisfies a goodness measure by removing irrelevant and redundant features. This method is effective, but the computational cost is very high, as one must exhaustively search all the feature subsets (Devroye *et al.*, 1996). For a problem with feature number d , the size of the subsets is 2^d . For each subset, one must run one computation; for instance, for the ANN or the SVM, the computational cost training process must be run. The high computational cost leads to the use of heuristic methods, such as branch and bound (NARENDRA and FUKUNAGA, 1977), genetic algorithm (Siedlecki and Sklansky, 1989), Tabu search (Zhang and Sun, 2002) and etc, to reduce the computational cost.

Principle Component Analysis (PCA) is an effective way to reduce the corrected and redundant features, as it can reduce the number of features without losing information. Eker and Camci have compared the feature selection method using PCA with the statistical t-test, where the non-significant feature is removed after the t-test. In their case study, the accuracy of the support vector machine using PCA is much higher than the feature selection using the t-test (O.F.Eker and F.Camci, 2010). The disadvantage of this PCA method is it requires an extra computation to perform data transformation. Kernel component analysis is the corresponding PCA for the kernel method to perform PCA nonlinearly (Schölkopf and Smola, 2002). PCA can be also considered a feature extraction method, as it can extract new features from existing features. But the new

features are generally not interpretable as they are extracted from a mathematical perspective.

Feature selection selects a sufficient minimum number of features containing sufficient information to ensure the best performance of the diagnostics model. This performance can be measured by the following: classifier error rate measures; distance measures; information measures; dependence measures; and consistency measures (Dash and Liu, 2003). For the most part, this thesis uses classifier error rate measures for the SVM model.

2.4 Failure Pattern Recognition

The features extracted from the data represent the characteristic status of the machine. A feature's value above a predefined threshold may imply a possible failure; the degree of the deviation may imply the severity of the failure. One challenge is how to determine this predefined threshold.

For some machinery, the threshold or boundary which can differentiate a normal state from failure or different types of failure can be defined by experience. For example, by experience, the Kurtosis feature value of a rotary bearing's vibration signal is 3 in its normal state. Therefore, the value 3 can be defined as a threshold to discriminate a normal bearing from a faulty bearing. However, in practice, for most situations, this cannot be done due to a lack of historical information or the existence of varying thresholds among specified machines or operating environments.

The threshold or boundary can be obtained automatically from available data using classical pattern recognition techniques, such as Bayesian classifier and k-nearest neighbour (Theodoridis and Koutroumbas, 2006), or by using a learning algorithm. For learning algorithm, when the feature values and the corresponding machine state are known, the learning algorithm is called supervising learning; if the machine state is not known, the algorithm is called unsupervised learning. The learning algorithm uses a decision function to discriminate different patterns; learning algorithm finds the optimal decision function automatically from the available data.

The simplest learning algorithm is the linear classifier; the nonlinear classifier includes the ANN and the SVM. This thesis proposes using the latter for failure diagnostics. Figure 2.4 uses synthesis data to show the SVM decision function; different colours denote the different patterns. The decision function is evidently nonlinear and flexible. This flexibility facilitates the ability to discriminate patterns.

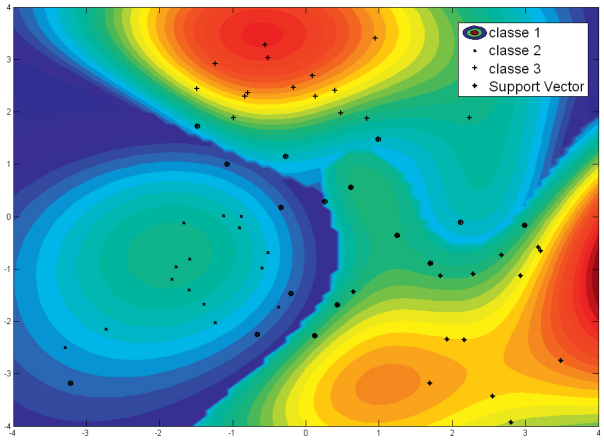


Figure 2.4 Nonlinear Patterns Recognition

The SVM can be used in both supervised and unsupervised learning. Supervised learning pairs the input and output data. The input can be the feature vector extracted from the signal, or the original raw signal when the size of the data points in the signal is small. Using raw data without feature representation as the input, the feature extraction step is skipped, but the SVM will be sensitive to the noise in the signal. The output of SVM can be the corresponding machine states as shown in Figure 2.5. Supervising learning selects internal coefficients by minimizing the predicted output and real output during its training. The optimal decision is the training result. The performance of the trained SVM can be further evaluated by using a set of test data as shown in Figure 2.5.

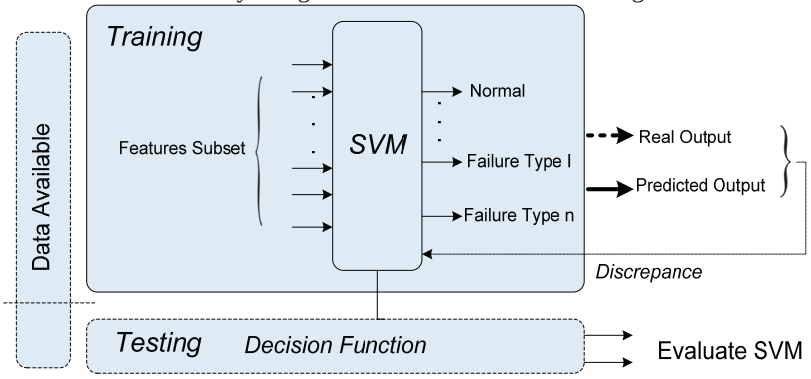


Figure 2.5 Schematic Diagram of Implementing SVM

The unsupervised learning such as novelty detection can be used to detect the abnormal events. A schema to implement the novelty detection shows in Figure 2.6. The optimal feature subset is used as input of the SVM. The data used to train the SVM is feature vectors and the training result is a boundary defined by these data. When a new data comes, using the decision function calculates if the new data is in the boundary. If it is not in the boundary, the abnormality may occur; otherwise, no abnormality detected.

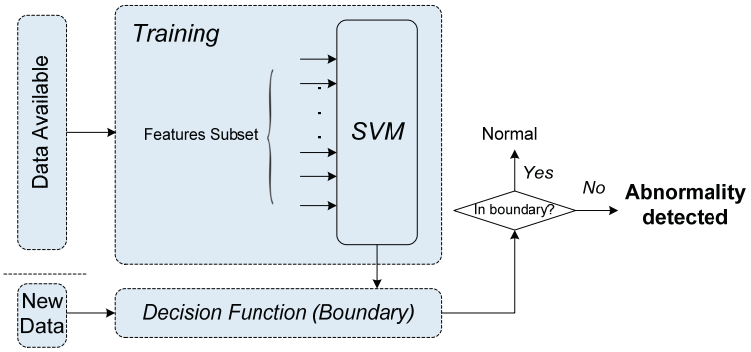


Figure 2.6 Schema of Implementing Novelty Detection

2.5 Failure Diagnostics for Railway Assets

2.5.1 Condition Monitoring on Railway

The railway is an important means of transportation for both freight and passengers. Improving its reliability, availability and safety will benefit society and reduce costs. The concept of condition monitoring has been implemented in the railway system in both rolling stock and infrastructure. A European company reports that its broken springs have decreased 90% since the implementation of condition monitoring, and the Canadian National Railway (CNR) reports a dramatic reduction in bearing failure after the installation of a condition monitoring system (Lagnebäck, 2007). The effectiveness of condition monitoring is evident.

In condition monitoring, sensors are mounted on selected asset to detect the asset's condition. Condition monitoring can enable the maintainer, in this case the railway, to move away from "find and fix" to "predict and prevent" (Bint, 2008). The railway system is complex, with a large geographical distribution and many personnel. Therefore, its condition monitoring system is also complex. Figure 2.7 illustrates a typical wayside condition monitoring system. In this figure, the sensors are mounted under the track to measure the temperature of the rail and the force of a train's passage. Sensors are also mounted in the bogies to measure the acoustic emission of the wheelset.

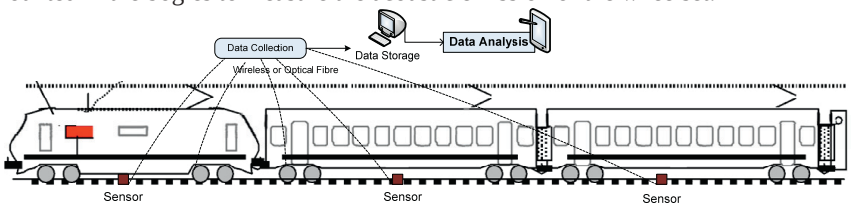


Figure 2.7 A typical Railway Condition Monitoring System

The measurements from the sensors are transmitted to a nearby collector, or to a data centre on the vehicle, or directly to a nearby data station. The data can be transmitted by means of optical fibre or by wireless. If the latter is chosen, one must consider reducing the disturbances on the existing railway signalling system. A set of automatic failure diagnostics algorithms or the engineers will analyze these data to find current or incipient failures.

The condition monitoring systems are essentially information technology infrastructures which enable collection, storage, and analysis of the health of the asset. Table 2.1 lists some measurements collected by railway condition monitoring.

Table 2.1 Diagnostics System

Category	Type of measurement	Category	Type of measurement
Track measurement	Track Geometry Rail Profile Rail Corrugation Ballast Profile	Vision Systems	Automatic rail surface defects detection Automatic overhead line defects detection
Overhead Line Measurement	Overhead line geometry Contact wire wear Pantograph interaction Arc Detection Overhead line electric parameters	Video inspection	Railway section and surroundings Track surfaces Overhead Line Platforms Way side
Vehicle dynamics measurement	Ride quality Body,bogie,axle boxes accelerations Wheel-rail interaction forces Wheel-rail contact	Others	Signalling Telecommunication quality Environmental Temperature Tunnel detection system Power consumption

Recently, some CM systems have been proposed which integrates the sensor information with internal train control information, train monitoring information, and passenger information (K.Liu *et al.*, 2008). This integrated information could increase the accuracy of failure diagnostics but at the same time, it could interfere with the operation of the train. Therefore, few CM systems are implementing this schema.

2.5.2 Switches and Crossings

Switches and crossings (S&C) are mechanical installations enabling trains to be guided from one track to another at a junction (Nissen, 2009) and allowing slower trains to be overtaken. They are an important part of the railway system. According to Swedish railway statistics, the railway infrastructure in Sweden has 17,000 km of railway and about 12,000 switches and crossings. The S&C are reported as frequently failing components by railway operators. S&C failures occur more frequently in Sweden due to severe winter weather. According to the event records of a Swedish railway database, the failures of Swedish S&C that were directly attributable to snow and ice was 17.4% for 2009-2010. Figure 2.8 illustrates the number of S&C failures during this period.

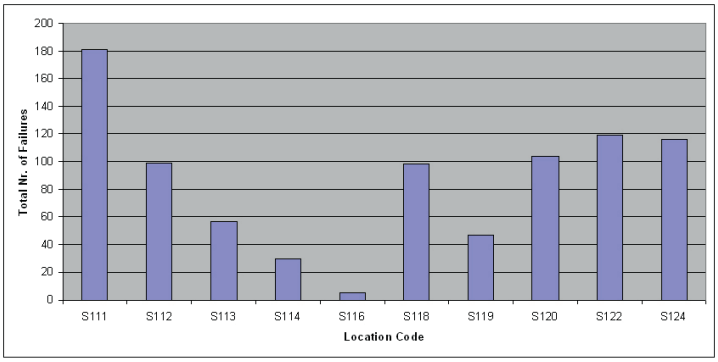


Figure 2.8 Number of S&C failures in Sweden

The total number of failures ranged from a high of 181 to a low of 5. S&C failures caused numerous delays, in fact, S&C related failures constituted 14% of all causes of train delays and S&C failure costs equalled at least 13% of the total maintenance cost. Clearly, this is an important functional and financial problem (Nissen, 2009).

There are two types of S&C: manual and automatic. Figure 2.9 illustrates a simplified automatic switch (F.Zhou *et al.*, 2001). As the figure shows, the switch is a complex system with many mechanical and electromechanical components. It has two movement directions, either pushing out ("reverse" movement) or pulling in ("normal" movement). The lock blade is used to fix the position of the rail. The movement of the switch takes the following steps: the motor torque is transferred to the clutch, then to the belt and the ballscrew, changing the rotating torque to an axial direction force. Using the crank, the force direction is changed by 90º to drive the switch rails.

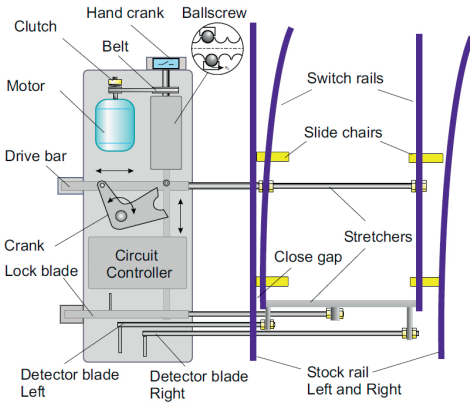


Figure 2.9 Switches in Railway Systems (F.Zhou *et al.*, 2001)

Figure 2.10 illustrates a layout of sensors to detect different failure modes for this switch described in Figure 2.9. These sensors measure the rail temperature, voltage, current in

the motor, and displacement. Their signals are transmitted to a local logger or to a local computer for analysis.

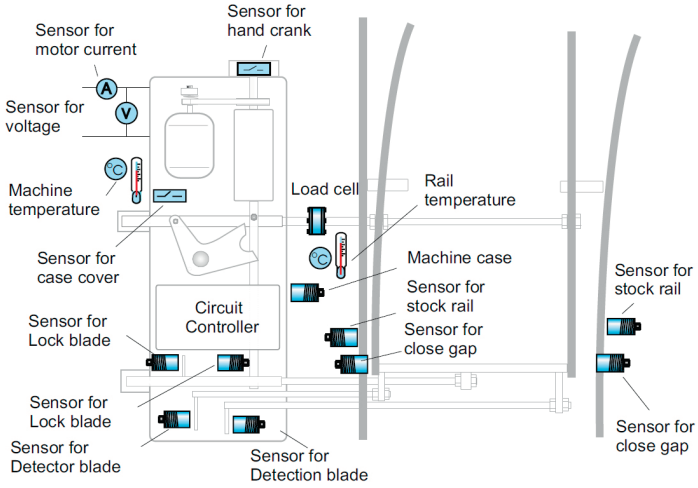


Figure 2.10 A Simplified Switch and the Installed Sensors

There are several in state-of-art methods available to analyze these signal data. Eker and Camci use the Support vector machine to determine whether the drive rod is out of adjustment (O.F.Eker and F.Camci, 2010). Chamroukhi *et al.* (2008) propose a method using mixture discriminant analysis to diagnose the failure in S&C electric motors by analyzing the consumed power (obtained by reading voltage and current sensors). Roberts uses neuro-fuzzy networks to discriminate various failures (Roberts *et al.*, 2002). Paper II appended to this thesis proposes the use of SVM to diagnose the lubrication level by analyzing the consumed power collected from the electric motor.

3. Support Vector Machine (SVM)

3.1 Background of Support Vector Machine

Support Vector Machine (SVM) is a learning algorithm which can automatically estimate dependency between data. The SVM is a classification problem when the dependency assigns labels to objects, and it is a regression problem when the dependency estimates the relationship between explanatory variables and predictive variables. In state-of-the-art research, the SVM is mostly used as a nonlinear classifier technique, while as a classifier, it can be explained from a geometrical point of view (Noble, 2006). The SVM has been successfully applied to a number of applications ranging from particle identification, face identification, and text categorization, to engine knock detection, bioinformatics, and database marketing (Bennett and Campbell, 2000).

V. Vapnik considers the SVM to be representative of the statistical learning theory (Vapnik, 1995). He claims the SVM is a further development of the original ANN type of learning algorithm, as it focuses on mathematical fundamentals (Vapnik, 1998). Recently, state-of-the-art research has featured numerous variants of the SVM (Li and Luan, 2003, Zhu and Hastie, 2005, Trafalis and Gilbert, 2006). However, there are two characteristics at the core of the SVM: the maximal margin and the kernel method. The next section of this chapter and Chapters 4 and 5 will discuss these in more detail.

3.2 The Framework of Support Vector Machine

The SVM incorporates the maximal margin strategy and the kernel method. Figure 3.1 illustrates the architecture of a classical SVM.

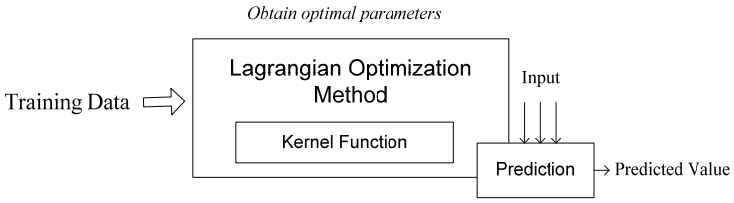


Figure 3.1 Architecture of SVM

The decision function of the SVM is an expansion of the kernel function. The Lagrangian optimization method is used to obtain this optimal decision function from the training data (Luenberger and Ye, 2008). The decision function is used to predict the output for a given input; this is the "prediction" shown in Figure 3.1. The maximal margin method is applied to improve the accuracy of the prediction.

Essentially, the SVM provides a general framework to learn from data. The dependence between data can be estimated using this framework. One can define a specific SVM based on this framework (Camci *et al.*, 2008, Camci and Chinnam, 2008). The support

vector classifier and support vector regressor discussed in the next section are applications of this framework.

In general, the SVM framework consists of the following components:

a. Use of the maximal margin to reduce the VC dimension, thus reducing the upper bound of the SVM and improving the generalization ability.
b. Kernel trick. The kernel function defines the similarity between two sample data. It can transform the problem from a lower dimension to a higher dimension, while the computation complexity remains the same. Transforming the problem from a lower to a higher dimension makes the approximation function more flexible with its data, reducing the risk of empirical error.
c. Sparseness. With fewer SVs (support vectors, data taking effect) the generalization ability is improved. Furthermore, as the decision function is comprised of SVs, having fewer SVs can reduce the computation complexity.
d. Convex optimization. The optimal solution of the SVM is achieved by the use of a quadratic optimization problem. The convex property of the formulation makes the solution unique. The SVM utilizes the Lagrangian optimization method to solve this problem.

3.3 Support Vector Classifier

The larger margin tends to a smaller generation error, as discussed in Section 4.7. Thus, maximizing the margin becomes the optimization objective. To demonstrate this, this section uses a simple binary classification problem as an example. As shown in Figure 3.2, the aim of the classifier is to classify the two classes of dots. Evidently, any straight lines located between these two classes of dots are able to separate them. The task is to decide which is optimal. The SVM uses the straight line which has the maximal margin as the optimal one; as discussed in Section 4.7, the maximal margin is prone to have a better performance. This optimal line is labelled L^* in Figure 3.2.

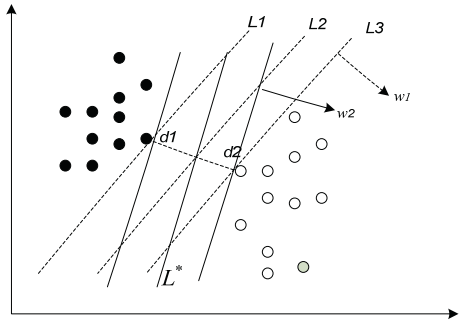


Figure 3.2 Binary Support Vector Classifier

Geometrically, the maximum margin can be obtained from the following constrained optimization formula:

$$\min_{w,b} \frac{1}{2}\|w\|^2$$
$$s.t. \quad y_i(< w, x_i > +b) \geq 1 \qquad i = 1,2,...,m. \tag{3.1}$$

where *w* denotes the normal lines perpendicular to the decision function line (*e.g.*, L^* in Figure 3.2), *b* denotes the bias, x_i denotes a input data set and y_i denotes the output, which is labelled by, e.g. 1 for the dark dot and -1 for the white dots. All data in Figures 3.2 correspond to a constraint in Formula (3.1).

Formula (3.1) is called the primal problem. Usually the SVM does not use the primal problem to obtain the optimal line; instead, it uses the simpler dual problem. By introducing the Lagrangian multiplier α_i, the dual problem of Formula (3.1) is written as

$$\max_{\alpha} \quad -\frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m}\alpha_i\alpha_j y_i y_j < x_i, x_j > +\sum_{k=1}^{m}\alpha_k$$
$$s.t. \ \ 0 \leq \alpha_i, \quad i = 1,2,3...,m \tag{3.2}$$
$$\sum_{i=0}^{m}\alpha_i y_i = 0$$

where α_i represents the Lagrangian multiplier which corresponds to the data set x_i. The $< x_i, x_j >$ can be further rewritten as $K(x_i, x_j)$, the kernel function discussed in Chapter 5.

The above Figure 3.2 shows a problem named the separable problem as all data sets can be linearly separated. For a problem which cannot be linearly separated, the SVM introduces slack variables ξ_i for Formula (3.1) to tolerate misclassification. The margin for the non separable problem is named the soft margin. The primal problem with the soft margin is formulated as:

$$\min \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{m}\xi_i$$
$$s.t. \ \ y_i(< w, x_i > +b) \geq 1 - \xi_i \tag{3.3}$$
$$\zeta_i \geq 0, \ \ i = 1,2,....,m.$$

The corresponding dual problem is:

$$\max_{\alpha} \quad -\frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m}\alpha_i\alpha_j y_i y_j < x_i, x_j > +\sum_{k=1}^{m}\alpha_k$$
$$s.t. \ \ 0 \leq \alpha_i \leq C, \quad i = 1,2,3...,m \tag{3.4}$$
$$\sum_{i=0}^{m}\alpha_i y_i = 0$$

where *C* is the penalty parameter used to penalize the misclassification.

The decision function for the classification is an expansion of the kernel function as shown in the following Formula (3.5). The coefficients of the expansion are obtained from Formula (3.4).

$$f(x) = \text{sgn}(\sum_{j=1}^{m} \alpha_j y_j K(x, x_j) + b) \tag{3.5}$$

The data set with $\alpha_i \neq 0$ is the Support Vector (SV).

3.4 Support Vector Regression

Support Vector Regression (SVR) is an extension of the support vector classifier which estimates the continuous function of certain training data sets. As shown in Figure 3.3, the data above the regression function $f(x)$ are considered class 1 data, and the data below $f(x)$ are considered class 2. In this sense, SVR transforms the regression problem into a special classification problem. Moreover, like the support vector classifier, the SVR uses soft margins to tolerate misclassification. Finally, SVR uses a tactic named ε-insensitive loss function (Schölkopf and Smola, 2002) to balance the approximate accuracy and computation complexity.

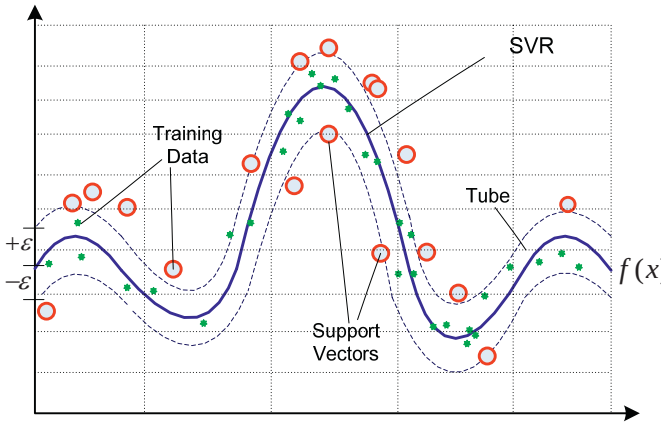


Figure 3.3 Support Vector Regression

As shown in Figure 3.3, the ε-insensitive function defines a tube with size of ε. Inside of the tube, there is no penalty on the deviation. However, outside of the tube, the penalty is imposed. Introducing the slack variable ξ_i and ξ_i^* and considering the regression problem as a binary classification problem, the primal problem of the SVR is written as follows:

$$\begin{aligned} &\min \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{m}(\xi_i + \xi_i^*) \\ &s.t.\ \ f(x_i) - y_i \leq \varepsilon + \xi_i \\ &\quad y_i - f(x_i) \leq \varepsilon + \xi_i^* \\ &\quad \xi_i \geq 0, \xi_i^* \geq 0.\ i = 1,2,3,....m. \end{aligned} \tag{3.6}$$

By introducing Lagrangian multipliers α_i and α_i^* for each inequation in Formula (3.6), a dual problem of Formula (3.6) is written as:

$$
\begin{aligned}
&max \quad W(\alpha, a^*) = -\varepsilon \sum_{i=1}^{m} (\alpha_i + \alpha_i^*) + \sum_{i=1}^{m} (\alpha_i^* - \alpha_i) y_i - \frac{1}{2} \sum_{j=1}^{m} (\alpha_i^* - \alpha_i)(\alpha_j^* - \alpha_j) \left\langle x_i, x_j \right\rangle \\
&s.t. \quad \alpha_i \geq 0, \alpha_j \geq 0; \\
&\qquad \alpha_i^* \leq C, \alpha_j^* \leq C; \\
&\qquad \sum_{i=1,j}^{m} (\alpha_i^* - \alpha_i) = 0, \; i = 1, 2, ..., m
\end{aligned}
\tag{3.7}
$$

The $<x_i,x_j>$ can be substituted by a kernel function $K<x_i,x_j>$. The desired function $f(x)$, which is also the decision function, is approximated as follows:

$$
f(x) = \sum_{j=1}^{m} (\alpha_i^* - \alpha_i) K(x, x_j) + b \tag{3.8}
$$

In SVC and SVR, the inner product $<x_i,x_j>$ is replaced directly by a kernel function without changing other parts of Formula (3.4) or (3.7). When a nonlinear kernel function is used, the optimal decision function can be obtained in the same way it is obtained in the simple inner product $<x_i,x_j>$, which is essentially a linear kernel function. From this point of view, the SVM solves a nonlinear problem in a linear way.

4. Generalization Error Bound

The SVM is a learning algorithm which can be considered as an implementation of statistical learning theory (Vapnik, 1995, Cherkassky and Mulier, 2007). A learning algorithm is illustrated in Figure 4.1., where x_i, y_i denote the observed input and output data respectively. For a failure diagnostics problem, x_i can be a feature vector; y_i can be the label, e.g. 1 and -1, of the pattern (e.g. failure or normal). The learning algorithm tries to minimize the margin between the actual output y_i and the predicted y_i.

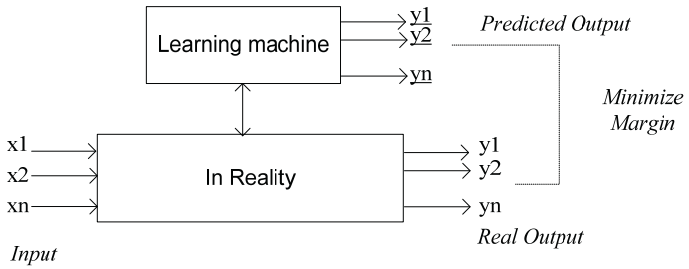


Figure 4.1 Leaning Algorithm

One major performance measure of a learning system is how closely its real output approximates its desired output (Sarkar, 1996). The generalization error which measures the margin of the real output and predicted output is often used to measure the generalization performance (Wang and Shen, 2006). Reducing the generalization error bound can improve the performance of the learning algorithm. Therefore, estimating the generalization error bound lays a solid theoretical foundation for developing an efficient learning algorithm.

4.1 Generalization Error for Data Known Distribution

Take the simplest binary classifier as example. Suppose the observations are picked up randomly from $x \times y \in \{x_1, x_2, ...x_n\} \times \{1, -1\}$. The hypothesis $h : x \to y$ from the hypothesis space H assigns a label from y for each element in x. These independent and identically distributed (i.i.d) observations are essentially drawn according to the distribution $P(x, y) = P(x)P(y \mid x)$ as:

$$(x_1, y_1), ..., (x_i, y_i), ..., (x_l, y_l) \qquad (4.1)$$

In this case, the generalization error can be implied by the probability of misclassification. Provided the data sets are (x_i, y_i), the probability of misclassification is $P(x_i, y \neq y_i) = P(x_i)P(y \neq y_i \mid x_i)$. If the distributions $P(x)$ and $P(y \mid x)$ are known, $P(x_i, y \neq y_i)$ can be derived easily. For example, assume

$P(x) \sim U[a,b]$, $P(y=1|x) \sim N(u_1, \sigma_1^2)$, and $P(y=-1|x) \sim N(u_2, \sigma_2^2)$. If the unknown label for x is the y with $\max\{P(y=1|x), P(y=-1|x)\}$, as the Bayesian classifier assumes (Theodoridis, 2010), the probability of misclassification, which is the generalization error, is

$$P(x_i, y \neq y_i) = \frac{1}{b-a}\int_{-\infty}^{x_0} \frac{1}{\sqrt{2\pi}\sigma_1} e^{-\frac{(s-u_1)}{2\sigma_1^2}} ds + \frac{1}{b-a}\int_{x_0}^{+\infty} \frac{1}{\sqrt{2\pi}\sigma_2} e^{-\frac{(s-u_2)}{2\sigma_2^2}} ds \qquad (4.2)$$

The misclassification $P(x_i, y \neq y_i)$ is illustrated as the shadow area in Figure 4.2. In the smaller $P(x_i, y \neq y_i)$, the classification is more accurate and reliable.

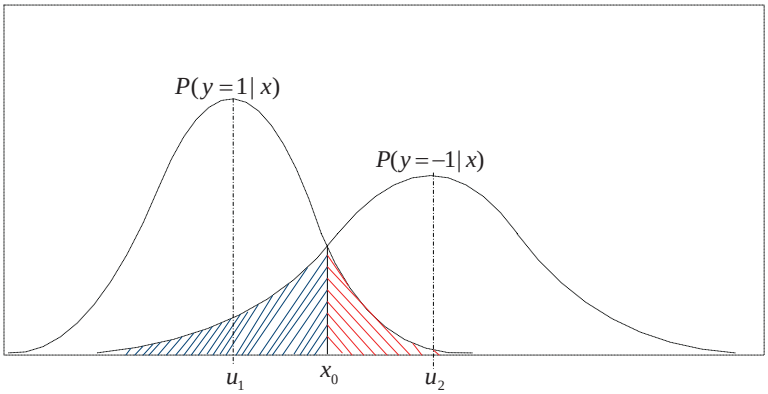


Figure 4.2 Generalization Error for Known Distribution

4.2 A Distribution Free Bound for Large Sample Size

When the distribution of $P(x)$ and $P(y|x)$ is known, the generalization bound can be easily derived, as the example in the previous section shows. However, the distribution based method requires a large sample size. For a small sample size, the Bayesian method can be used; this requires a predefined distribution, but the distribution is sometimes unknown. In this case, one can use another method to obtain the upper bound of the generalization error.

A loose generalization bound can be derived using large number theory when the observations assume i.i.d. Using the same data sets as in Section 4.1, the probability of misclassification is $P(x_i, y \neq y_i)$, which we denote as ε for simplicity. We define a loss function as:

$$L(x_i) = \frac{1}{2}|f(x_i) - y_i| \qquad (4.3)$$

where (x_i, y_i) are observation data, $f(x)$ are functions defined on x and $f(x_i) \in \{1,-1\}$. Obviously, $L(x_i)$ is a random variable; for correct classification, $L(x_i) = 0$; for the contrary $L(x_i) = 1$. $L(x_i)$ can be considered a Bernoulli trial as it contains binary events.

Coincidently, for Bernoulli trials, the number of misclassifications for l observations is $\sum L(x_i)$. Therefore, one can establish an empirical misclassification rate, which is also called empirical risk, as $\frac{\sum L(x_i)}{l}$. $\frac{\sum L(x_i)}{l}$ can be used as an estimate of $P(x_i, y \neq y_i)$. Thereafter, a distribution free bound of generalization error can be derived by the Chernoff inequality (Chernoff, 1952) as,

$$P\left\{\left|\frac{\sum L(x)}{l} - \frac{E(L(x))}{l}\right| \geq \varepsilon\right\} \leq 2\exp(-2l\varepsilon^2) \qquad (4.4)$$

$E(L(x))/l = P(x_i, y \neq y_i)$ for Bernoulli trials; hence, one can write Formula (4.4) in PAC (Probably Approximately Correct) style (Cristianini and Shawe-Taylor, 2000). An upper bound of $P(x_i, y \neq y_i)$ is as follows: with the probability σ

$$P(x_i, y \neq y_i) \leq \frac{\sum L(x)}{l} + \sqrt{-\frac{\ln\frac{\sigma}{2}}{2l}} \qquad (4.5)$$

This bound suggests that when $l \to \infty$, the second part of the right hand of Eq (4.5) will be zero, i.e. the empirical risk is close to the real risk $P(x_i, y \neq y_i)$. Figure 4.3 shows the bound of error probability decreasing with sample size and tending to a constant when the sample size increases. This constant is the real risk $P(x_i, y \neq y_i)$.

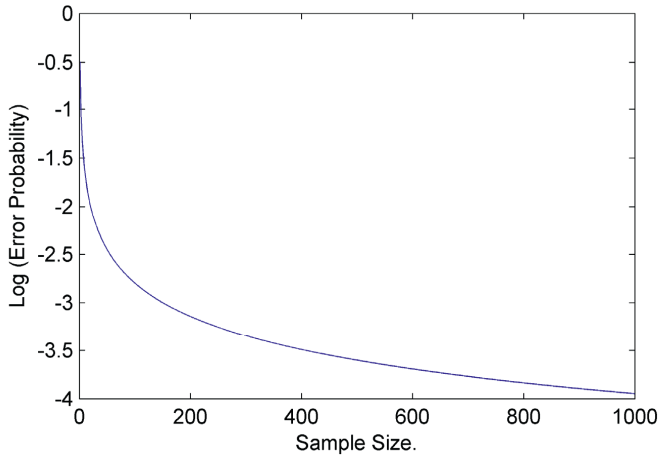


Figure 4.3 Error Probability Bound vs Sample Size

On the other hand, Inequality (4.5) suggests that reducing the number of misclassification $\sum L(x_i)$ can reduce the overall misclassification rate when many observations are available. Selecting a function that minimizes the empirical misclassification rate can achieve a low misclassification rate. The criterion defining an optimal function as the lowest empirical misclassification rate is called the Empirical Risk Minimization (ERM). The ANN employs this criterion to optimize its internal parameters and obtain an optimal decision function. However, this criterion is only suitable for larger sample size problems.

4.3 Bias-Variance Dilemma

Using a more complex function, say a function with more parameters, can lead to fewer errors. As suggested in Inequality (4.5), having fewer errors will lead to the lower bound. However, this argument is only valid if there is a sufficiently large number of data sets. Finite data sets can lead to the "Bias-Variance Dilemma" phenomenon (Geman *et al.*, 1992) whereby more complex model does not always lead to better performance. The expected Mean-Squared Error can be expressed as:

$$E_D\left[(f(x;D)-E(y\mid x))^2\right]=\left[E_D f(x;D)-E(y\mid x)\right]^2+E_D\left[f(x;D)-E_D f(y\mid x)\right]^2 \quad (4.6)$$

The first part on the right hand of Eq (4.6) is "Bias", the second part is "Variance". Let the $f(x;D)$ be the estimating function on the target function $f(x)$ given data set D. This equation suggests that the overall mean square error tradeoffs bias and variance. A more complex $f(x;D)$ with more free parameters can fit a data set D perfectly, i.e. the bias can be zero. But its variance may increase. A less complex model has less variance but has larger bias.

The Figure 4.4 shows a figure demonstrating the "Bias Variance Dilemma". The dashed curve $f(x)$ is the objective function. The data sets are drawn from $y=f(x)+\varepsilon$ where the ε denotes random noise. Firstly, arbitrarily selecting the estimating function $f(x,D)$ as a straight line independent of the noise ε, so the variance of this function is zero. However, its bias is large, as the straight line deviates from the objective $f(x)$ greatly; On the other side, the $f_1(x)$ and $f_2(x)$ are flexible functions which can well fit the data sets D_1 and D_2 respectively. But the variance is large, as the estimating function differs from each other if using different data sets, in spite the target function $f(x)$ is the same.

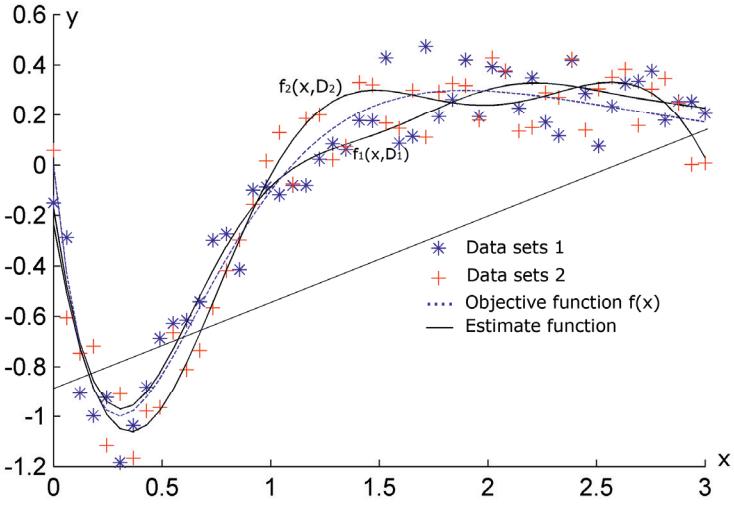


Figure 4.4 Linear Functions and Flexible Nonlinear Function

For a learning algorithm, a complex model will fit the available training data sets perfectly but possibly have large deviation for other unseen data sets. This scenario is also called “overfitting” problem. A simpler model, for example linear model, will have less variance but it has larger bias. This scenario is also called “underfitting”. To build a model with a good performance one has to trade-off the bias and variance.

4.4 Selection of Optimal Function

By using a larger sample size, one can avoid the “Bias-Variance Dilemma”. However, in some situations, obtaining a larger sample size is impossible or costly. If obtaining a larger sample size is impossible, a small sample size problem may result. In this case, Inequality (4.5) is still applicable. But when l is small, the right side of Inequality (4.5) increases, i.e. the deviation between the real risk and the empirical risk becomes larger, and the ERM criterion will not be suitable. Essentially, for a small sample size problem, even using the simpler low order polynomial or linear model, a numerous, even infinite, number of functions could result in the lowest empirical error. But the function with the lowest empirical error cannot guarantee the lowest generalization error.

Equation (4.6) explains this dilemma but cannot be used to build a good model, as the variance depends on the distribution of data D. The distribution of D is not available in most situations.

The Structural Risk Minimization (SRM) inductive principle is a practical method which can be used to balance bias and variance (Vapnik, 1998). The SRM principle tries to minimize a confidence interval of the real risk to select the optimal function. This

confidence interval is derived from a set of admissible functions. The optimal function is obtained from tradeoffs between empirical risk and the size of the admissible functions. As the complex model usually has a larger number of admissible functions, controlling the size of the admissible functions in some way is essential to control the model's complexity.

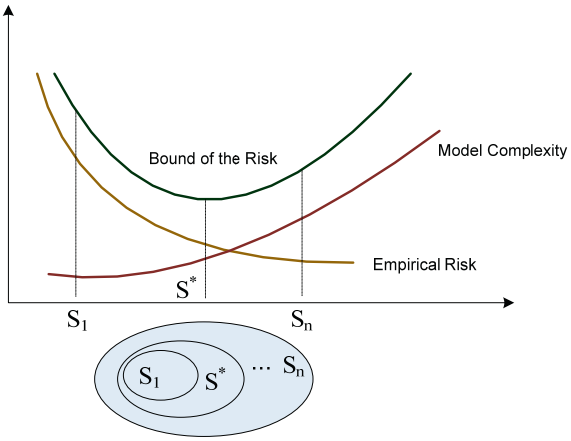


Figure 4.5 Risk Bound of Learning Algorithm

Figure 4.5 illustrates the relationship among real risk bound (e.g. real misclassification error), empirical risk (or empirical error) and model complexity. The complex model usually has more admissible functions and hence more easily finds the lower empirical risk. However, the ultimate real risk bound can increase with model complexity as well. For example, using the ANN for classification, the training error becomes lower if the number of neurons is increased, i.e. creating a more complex model with more admissible functions, but the test error may increase accordingly. The optimal function with the optimal risk bound is a trade off between empirical risk and model complexity (Vapnik, 1998). The remaining sections of this chapter introduce a bound which explains this trade off theoretically.

4.5 A General Distribution-Free Risk Bound

The above mentioned bounds do not consider the complexity of the function. This section derives a bound which incorporates the ERM and the complexity of the function. This bound is the foundation of the learning algorithm such as that found in the SVM. In a consistent function, the empirical error and real risk will approach zero as sample size l approaches infinity. For a set of functions F, Vapnik and Chervonenkis define consistence as (Vapnik, 1995)

$$\lim_{l\to\infty} P\left\{\sup_{f\in F}(R(f) - R^{l}_{emp}(f) > \varepsilon\right\} = 0 \qquad (4.7)$$

The superscript l in Equation (4.7) denotes the sample size. Equation (4.7) can be bounded by the following inequality using the union theorem:

$$P\left\{\sup_{f\in F}(R(f)-R^{l}_{emp}(f)>\varepsilon\right\}\le\sum_{f\in F}P\left\{(R(f)-R^{l}_{emp}(f)>\varepsilon\right\} \quad (4.8)$$

The right hand of Inequality (4.8) depends on the number of function f. For an infinite set of functions such as those with a continuous function, the size of F is infinite, and the bound derived from Inequality (4.8) is impossible. A tactic introduced by Vapnik (Vapnik, 1998) can be used to bound (4.8) for a infinite number of functions. Each function $f\in F$ can be bounded by the probability that it differs by more than $\varepsilon/2$ from the empirical risk on a second and equal size l. This tactic, called Symmetrization (Schölkopf and Smola, 2002), is the following for $\varepsilon^2 l\ge 2$:

$$P\left\{(R(f)-R^{l}_{emp}(f)>\varepsilon\right\}\le 2P_{2l}\left\{(R^{l}_{emp}(f)-R^{l'}_{emp}(f)>\varepsilon/2\right\} \quad (4.9)$$

In Inequality (4.9), l and l' are equal. The bound on the right hand side of Inequality (4.9) can be obtained by considering each sample as a Bernoulli trial. The bound on the right hand side of Inequality (4.9) can be obtained by using the modified Chernoff inequality (Chernoff, 1952) as:

$$P_{2l}\left\{(R^{l}_{emp}(f)-R^{l'}_{emp}(f)>\varepsilon/2\right\}\le 2\exp\left(-\frac{l\varepsilon^2}{8}\right) \quad (4.10)$$

When Inequality (4.8), (4.9) and (4.10) are joined, Inequality (4.8) can be rewritten as:

$$P\left\{\sup_{f\in F}(R(f)-R_{emp}(f)>\varepsilon\right\}\le 4N(F,2l)\exp\left(-\frac{l\varepsilon^2}{8}\right) \quad (4.11)$$

where $N(F,2l)$ denotes the number of admissible functions for the $2l$ samples. $N(F,2l)$ is finite, as the maximum number of $N(F,2l)$ for $2l$ samples is 2^{2l}.

From Inequality (4.11), we can derive a bound for the $R(f)$ in a PAC style. The following inequality satisfies the probability at least $1-\sigma$:

$$R(f)\le R_{emp}(f)+\sqrt{\frac{8}{l}[\ln N(F,2l)+\ln(\frac{4}{\delta})]} \quad (4.12)$$

This is called the confidence interval for $R(f)$. This inequality is important. It implies that the bound of $R(f)$ not only depends on empirical error, but also on the size of functions $N(F,2l)$. The objective of one learning algorithm can be the minima of Inequality (4.12), as is done by the Support Vector Machine.

Remarkably, the bound for $R(f)$ is not unique, and the bound in Inequality (4.12) is not the lowest bound. Other bounds for $R(f)$ have been obtained. For example, Cristianini and Shawe-Taylor have derived (Cristianini and Shawe-Taylor, 2000):

$$R(f) = P(x_i, y \neq y_i) \leq \frac{2}{l}[\ln\frac{2}{\delta} + \ln N(F,l)] \qquad (4.13)$$

This bound also suggests the misclassification error depends on the size of the admissible functions.

4.6 Capacity of Admissible Functions

$N(F,2l)$ in Inequality (4.13) and (4.14) is the size of the admissible functions. For sample size l, the capacity of this function can be measured by a quantity known as the growth function. The growth function either satisfies the equality (Vapnik, 1998)

$$N(F,l) = 2^l \qquad (4.14)$$

or is bounded by

$$N(F,l) \leq (\frac{el}{d})^d \qquad (4.15)$$

The d is the Vapnik-Chervonenkis (VC) dimension, which measures the complexity of the admissible functions. Notably, when $d < l$, the growth function monotonically increases with d.

For function $f(x) \in \{1,-1\}$, the VC dimension is the maximum number of l which can be separated into two classes in all 2^l possible ways using the set of function F (Vapnik, 1998). For example, for a 2-dimensional dot as shown in Figure 4.6, three dots can be separated no matter what the label (1 or -1) of the dot may be. However, for four dots, the dots cannot be separated, as shown on the right hand side of Figure 4.6. So for the linear function of the 2-dimension, the VC dimension is 3.

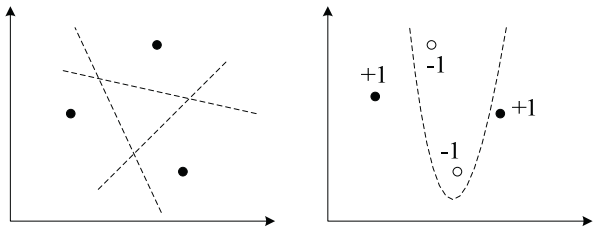


Figure 4.6 VC Dimension of Linear Functions

The above example shows that the VC dimension of the 2-dimensional linear function is 3. Essentially, for n-dimensional linear functions, the VC dimension is $n+1$ (Burges, 1998). This result is important for the SVM.

4.7 Maximal Margin Strategy

Numerous or even infinite admissible functions can lead to zero empirical misclassification rates. A straightforward problem is how to select the optimal function from these zero error functions. The error bound depends on the capacity of admissible functions. As discussed, a smaller set of functions will have a smaller bound. Using a maximal margin strategy can effectively reduce the size of admissible functions. Figure 4.7 illustrates a binary classifier; any line between these two classes of dots can separate them linearly, i.e. any straight line between them is a candidate line with a corresponding linear function.

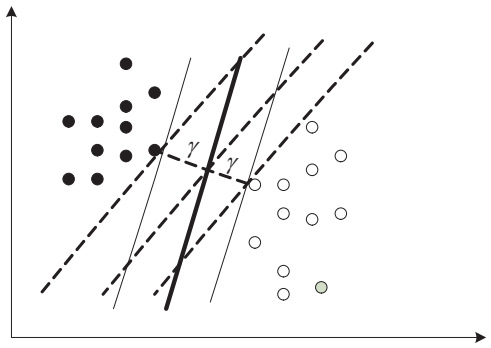


Figure 4.7 Maximum Margin Strategy

Intuitively, the most reasonable line is the bold solid line shown in Figure 4.7, as the separator will more robustly separate the dots with larger margin. We will explain this theoretically in what follows.

The margin of a set of samples $(x_i, y_i) \in R^n \times \{1,-1\}$, where n is the dimension of x_i, for a function $f \in F$ is defined as the quantity

$$\gamma_i = y_i f(x_i) \qquad (4.16)$$

Note that $\gamma_i \geq 0$ means a correct classification of the point (x_i, y_i) ; $\gamma_i < 0$ means misclassification; $f(x) = 0$ is the separator as the straight line shown in Figure 4.7. For a separable case, the margin of a set of samples is defined as the minimal γ_i. Written in a mathematical form, it appears as:

$$\gamma = \min\{\gamma_i = y_i f(x_i) : i = 1,...,l\} \qquad (4.17)$$

The γ is as shown in Figure 4.7. Employing margin strategy, the VC dimension can be obtained from an important finding by Vapnik (Schölkopf and Smola, 2002) as:

For the linear function, consider linear hyperplanes $< w, x >= 0$ *with* $| w | \leq B$ *and the sample* x_i *defined within a sphere of radius* R*, i.e.* $| x_i | \leq R$*; then the VC dimension is satisfying.*

$$VC_l(\gamma) \leq (\frac{BR}{\gamma})^2 \qquad (4.18)$$

This finding implies that with larger γ, the VC dimension is smaller. Using Inequality (4.18), Inequality (4.12) can be rewritten as:

$$R(f) \leq R_{emp}(f) + \sqrt{\frac{8}{l}[(\frac{BR}{\gamma})^2 \ln \frac{2el\gamma^2}{(BR)^2} + \ln(\frac{4}{\delta})]} \qquad (4.19)$$

Inequality (4.19) shows that once the VC dimension is decreased, the error bound decreases accordingly. The SVM utilizes this strategy to select the optimal decision function.

5. Kernel Method

Kernel method is a class of algorithms developed for multivariate data analysis and pattern recognition measures the similarity between pairwise data sets. The notion of "similarity" can be geometrically interpreted as the distance between data sets. The distance between two samples can be Euclidean or Riemannian. Examples of the kernel method include the Support Vector Machine (SVM), Gaussian Process, kernel principle analysis etc. The kernel method obtains its name because it uses the kernel function.

5.1 Kernel function

A linear classifier can be used to introduce the concept of kernel function. This linear classifier is called the linear perceptron in the ANN. Suppose $x \in R^n$, i.e. in an n-dimensional space. The decision function for a classifier is defined as (Cristianini and Shawe-Taylor, 2000):

$$f(x) = \text{sgn}(\sum_{i=1}^{n} w_i x^i + b) \qquad (5.1)$$

where sgn(.) denotes 1 or -1. The coefficients w can be calculated from a combination of training data sets as:

$$w = \sum_{i=1}^{l} \alpha_i y_i x_i \qquad (5.2)$$

Hence, Equation (5.1) can be rewritten as:

$$f(x) = \text{sgn}(\sum_{i=1}^{l} \alpha_i y_i < x_i, x > + b) \qquad (5.3)$$

Now transform the n-dimension x into another dimension using the following transformation function:

$$x = (x_1, x_2, ...x_n) \mapsto \Phi(x) = (\phi_1(x), ..., \phi_i(x), ..., \phi_m(x)) \qquad (5.4)$$

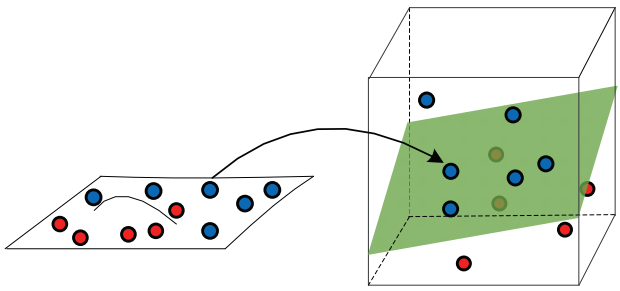


Figure 5.1 From 2-dimension to 3-dimension

The new number of dimensions is represented by m and is called the feature space. As in Equation (5.1), the corresponding decision function in the new space is

$f(x) = \text{sgn}(\sum_{i=1}^{m} w_i \phi_i(x) + b)$. As $w_i = \sum_{j}^{l} \alpha_i y_i \phi_i(x_j)$, the decision function in the new space can be written as

$$f(x) = \text{sgn}(\sum_{j}^{l} \alpha_i y_i < \Phi(x_j).\Phi(x) > + b) \qquad (5.5)$$

The decision function is represented by an expansion of the inner product. Since in the decision function the only thing of interest is the value of the inner product, one obvious solution is to replace this inner product with function $k(x_i, x)$. This function is called the kernel function. Suppose the dimension of the new space is infinite. The kernel function in Eq (5.5) can be rewritten as:

$$k(x_j, x) = \sum_{i=1}^{\infty} \lambda_i \phi_i(x_j) \phi_i(x) \qquad (5.6)$$

where λ_i is the weight for each dimension in the new space. By using Eq (5.6), we have converted a computation using the inner product to a computation using a function. In the kernel method, the decision function is represented by a combination of kernel functions. Equation (5.5) is thus rewritten as:

$$f(x) = \text{sgn}(\sum_{j=1}^{m} \alpha_j y_j k(x, x_j) + b) \qquad (5.7)$$

where α_j denotes the coefficient of data sets x_j, and b denotes bias.

5.2 Condition of Kernel function

The kernel definition given in Eq (5.6) shows how a feature space defines a kernel function. But what makes a function a legitimate kernel function? First, the function must be symmetric; second, it must satisfy the Cauchy-Schwarz inequality. Any function $k(x, z)$ which satisfies the Mercer Theorem is legitimate as a kernel function.

Mercer Theorem: Suppose $k(x, z)$ is a continuous symmetric function such that the integral operator $T_k : L_2(X) \to L_2(X)$ with $(T_k f)(.) = \int_X k(., x) f(x) dx$ is positive; that is

$$\int_{X \times X} k(x, z) f(x) f(z) dx dz \geq 0 \qquad (5.8)$$

for all $f \in L_2(X)$. Then one can expand $k(x, z)$ in a uniformly convergent series in terms of T_k's eigen-functions $\Phi_j \in L_2(X)$, normalized with $\|\Phi_j\| = 1$ and its corresponding eigenvalues $\lambda_i \geq 0$. The kernel:

$$k(x, z) = \sum_{j=1}^{\infty} \lambda_j \Phi_j(x) \Phi_j(z) \qquad (5.9)$$

The mercer theorem suggests that once the function can be represented as Eq(5.9), the function has a positive definition and can be used as kernel function.

5.3 Some Kernel Functions

There are many available kernel functions. Common ones are the Gaussian function, Polynomial function, Fourier Series kernel and so on (Gunn, 1998).

The form of the Gaussian function is:

$$k(x,z) = \exp(-\frac{\|x-z\|}{2\sigma^2}) \tag{5.10}$$

The form of the polynomial function is:

$$k(x,z) = (< x,z > +1)^d \quad \text{or} \quad k(x,z) = (< x,z >)^d \tag{5.11}$$

The form of the Fourier Series is:

$$k(x,z) = \frac{\sin(N+0.5)(x-z)}{\sin(0.5(x-z))} \tag{5.12}$$

The kernel function can be a continuous function like Equation (5.10)(5.11)(5.12). The kernel can be in other forms such as string kernel, tree kernel, graph kernel (Gartner, 2008). A kernel function can be designed for a specific application.

5.4 Kernel Function in Riemannian Geometry

The kernel function transforms input space to feature space. The feature space $\phi_i(x)$ explains the kernel function theoretically, but it is difficult to obtain explicitly. In practice, for example, to improve the performance of the kernel function, one can explain the kernel function using Riemannian geometry.

As noted above, the distance between two points in the kernel function is mostly not Euclidean distance but Riemannian distance. Figure 5.2 visualizes a surface with a 2-d input space for a $d = 2$ polynomial function. The curvature of this figure suggests that the distance in the feature space is not Euclidean distance.

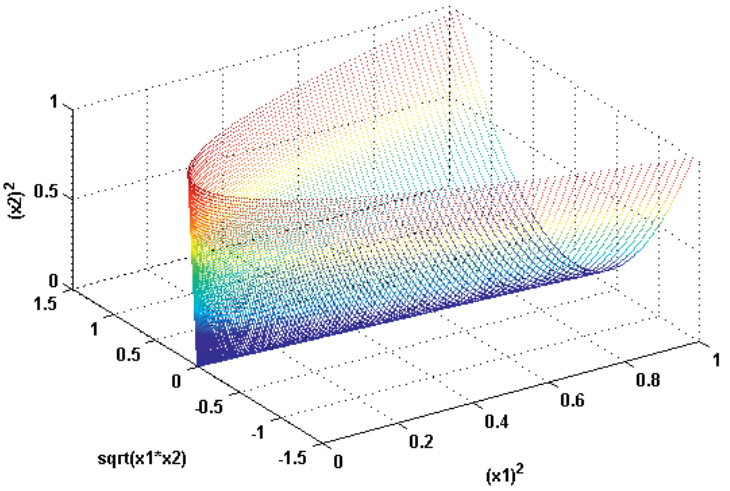


Figure 5.2 Curvature using Polynomial Function

Figure 5.3 illustrates the difference between Euclidean distance and Riemannian distance. On surface *S*, the distance between AB is II not I. The I is Euclidean distance, and the II is Riemannian distance.

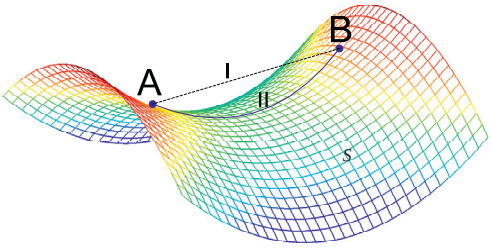


Figure 5.3 Riemannian Distance

For a kernel function, this line element of the Riemannian distance is (Amari and Wu, 1999):

$$ds^2 = \sum_{i,j} g_{ij}(x) dx_i dx_j \qquad (5.13)$$

The g_{ij} in Eq (5.13) is the Riemannian tensor metric induced in the feature space as

$$g_{ij}(x) = \frac{\partial}{\partial x_i}\frac{\partial}{\partial y_j} k(x,y)\Big|_{x=y} \qquad (5.14)$$

For a Euclidean distance, $g_{ij} = \delta_{ij}$, where δ_{ij} is a function defined as the following: for $i = j$, $\delta_{ij} = 1$, otherwise $\delta_{ij} = 0$.

For the Gaussian kernel function, $g_{ij} = \frac{1}{\sigma^2}\delta_{ij}$. For the polynomial function (Wu and Amari, 2002),

$$g_{ij} = \delta_{ij} d + x_i x_j d(d-1) \qquad (5.15)$$

The use of Riemannian geometry to explain the kernel function facilitates the kernel function's improvement. The Gaussian and polynomial kernel functions have been improved by modifying the tensor metric g_{ij} for a particular application (Schölkopf *et al.*, 1999).

5.5 Advantage of Kernel Function

One advantage of the kernel function is that it renders the computational cost independent of the input data dimension. For the classical linear regression, the complexity of the regression increases as the dimension of the input data increases. When the dimension is high, the regression will be complex, and the computational cost will increase. However, in kernel method, the computation of the inner product will be transferred into the kernel function. This means the computation cost depends only on the kernel function. Therefore, even with a high dimension of input data, the computation cost remains the computation of kernel function.

The kernel method is also an efficient way to perform data fusion. Data fusion is necessary when the data come from various data sources. For example, a condition monitoring system has mounted several sensors which measure humidity, temperature, pressure and so on. One can also extract features from the time-domain, frequency-domain or the time frequency domain. Thus, the number of features is large. If one is using the classical regression techniques or its variant, the regression model will be complex. For this situation, one can use the kernel method, merging all the features into a kernel function. To illustrate, we take the schematic example shown in Figure 5.4. We suppose the data are from n sensors. In sensor 1, features are extracted from both time and frequency domains. In sensor n, the features are only from the time domain. We incorporate these features into a kernel function, as shown in Figure 5.4. The data from various sensors are thus fused into the kernel function.

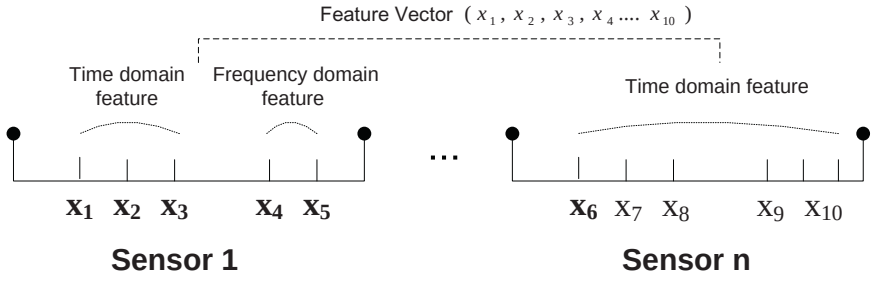


Figure 5.4 Sensor Data Fusion

Another advantage of the kernel method is that it can transform input data x from a lower dimension to a higher dimension implicitly. In the higher dimensional space, the pattern will be more distinct. Moreover, using the kernel function in the higher dimension, the computational cost will not improve significantly. In this sense, the kernel function is a shortcut, as in the classical way, one would find a transformation function first and then compute the inner product as shown in Equation (5.5). But the kernel function skips this step.

6. Application of SVM in Reliability

The application of Support Vector Machine has been mostly in the field of medicine, bioinformatics, meteorology, and etc (Noble, 2006), either using classical SVM or its variants. Recently, some literature has applied the Support Vector Machine in reliability. In general, these applications can be classified into two categories: failure diagnostics such as novelty detection and multi-failure discrimination; and secondly reliability data analysis, such as reliability prediction and system reliability assessment.

6.1 Novelty Detection

An important application of the SVM to reliability is novelty or anomaly detection, also called the one class classification problem (Davy *et al.*, 2006). This SVM is named the support vector data description (SVDD) (Tax and Duin, 2004). The aim of the one class classification is to define a boundary separating one kind of data from another, e.g. separating normal data from abnormal data, failure data from functional data, and normal data from incipient failure data. The SVM as a means of novelty detection is illustrated in Figure 6.1. In this figure, the normal data are inside the boundary. Once data are found locating outside, i.e. outlier data (Red dot in 6.1), the suspected abnormality has occurred.

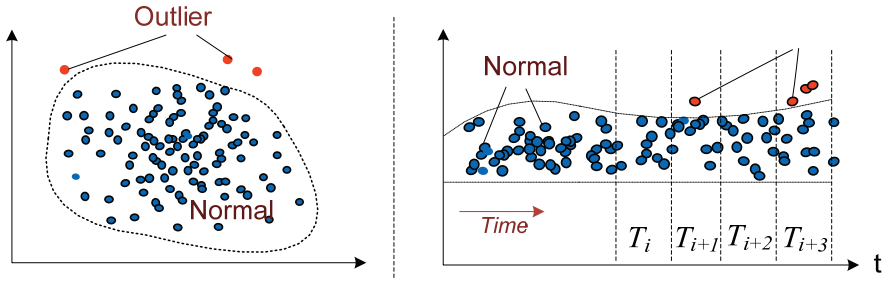


Figure 6.1. SVM for Novelty Detection

The SVM has a theoretical advantage over other novelty detection techniques based on statistics.

- As earlier discussed, the SVM is a distribution free technique. The boundary for novelty is defined with no assumptions on specified distribution.
- The novelty boundary can be linear or nonlinear depending on the kernel function. For nonlinear problems, the Gaussian or high order Polynomial function can be the kernel function. For linear problems, the first order Polynomial function can be the kernel function.
- The novelty using the SVM is adaptive to new data. For an online condition monitoring system, the boundary of the novelty can be adjusted automatically by the newly added data, i.e. it is a continuous learning process.

The disadvantage of the SVM as a novelty technique is the high computational cost. Some state-of-the-art applications use the SVM or its variants for the novelty technique. For example, Sortiris *et al.* (2010) have developed a Bayesian SVM as a novelty technique, and Camci uses the SVM as a quality control technique (Camci and Chinnam, 2008, Camci *et al.*, 2008).

6.2 Failure Diagnostics

The SVM has been used to diagnose failure in rolling element bearings, induction motors, machine tools, pumps, compressors, valves, turbines, HVAC and various other machines (Widodo and Yang, 2007). In these applications, the SVM is used as classifier, called the Support Vector Classifier (SVC). Initially, the SVC is a binary classifier but can be extended to become a multi-class classifier. The multi-class classifier is composed of a group of binary classifiers. According to the organization of the binary classifiers, the multi-class classifier belongs to one of the following categories (Hsu and Lin, 2002):

- One-against-all
- One-against-one
- Direct Acyclic Graph (DAG).

Hsu and Lin discuss the performance of the three multi-class SVMs (Hsu and Lin, 2002). They observe no significant differences among these multi-class classifiers in terms of accuracy. However, in terms of training time, the DAG method is best.

Figure 6.2 shows an example using the Multi-Class SVC to diagnose failure for a bearing. The kernel function used in the figure is the first order linear polynomial kernel function, so the Multi-Class SVC is a linear classifier. The upper figure in Figure 6.2 uses the feature Normalized Normal Negative Likelihood and the Shannon entropy. It shows that the two features can separate the normal pattern, namely, the inner race and outer race defect, linearly. The lower figure in Figure 6.2 tries to separate more patterns, including normal patterns and patterns with different defect sizes; the features used are the Crest and Weibull negative likelihood. These patterns can also be linearly separated by the SVC. The definition of these features can be found in Paper V appended to this thesis.

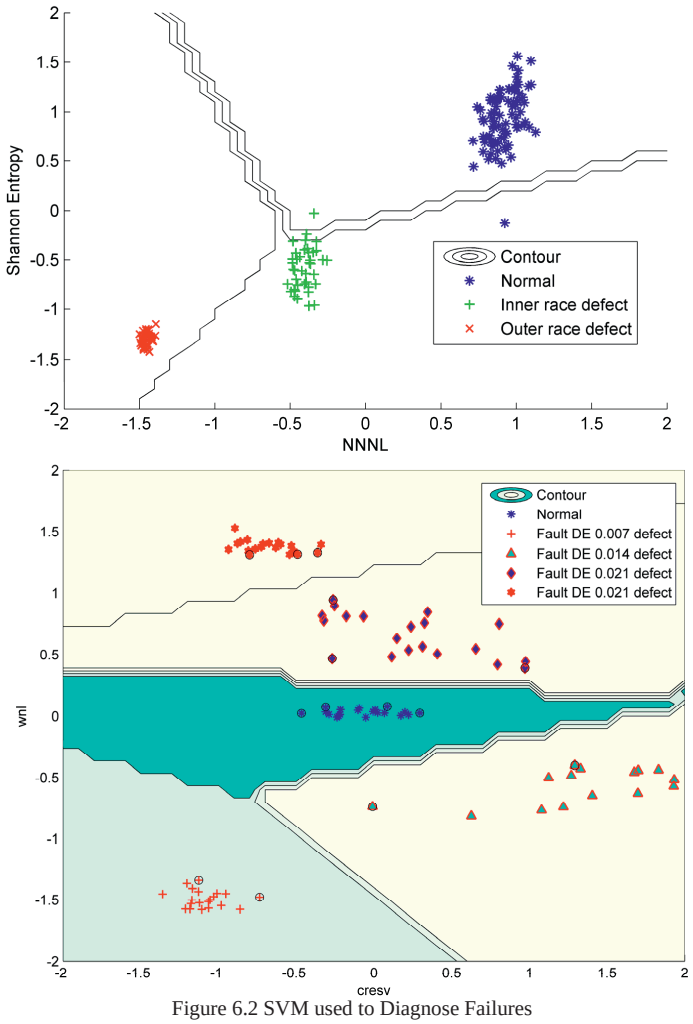


Figure 6.2 SVM used to Diagnose Failures

6.3 Predicting

Support Vector Regression (SVR) has been widely used for predicting (Sapankevych and Sankar, 2009). The prediction error of SVR is very small if a proper kernel function and proper parameters are selected (Sapankevych and Sankar, 2009, Thiessen *et al.*, 2003). For example, Chen (2007) has proposed a methodology to incorporate SVR with time series and genetic algorithms to predict reliability. Chen uses genetic algorithms to estimate SVM parameters. SVR is then used to establish the relationship between failure

data and system reliability performance (Chen, 2007). This paper shows that the SVM is more accurate than the ANN. Fuqing (2010) proposes SVR with a time series to predict reliability for a repairable system. This paper compares the accuracy of the classical NHPP model and SVR.

The following example demonstrates the use of the SVM as forecasting technique. In this example, the SVR is combined with the time series. Suppose the time sequential event are $x_1,\ x_2,\ x_3 \ldots x_n$. A *m*-lag time series is composed from these event data as the input of the SVR and the next event time x_i as output. The input output pair for the SVM is shown in Table 6.1.

Table 6.1 n-step lagged input vector and its corresponding output

Time Series	x(input)					y(output)
s_1	x_1	x_2	….	x_{m-1}	x_m	x_{m+1}
s_2	x_2	x_3	….	x_m	x_{m+1}	x_{m+2}
…	…	…	….	…	…	…
s_{n-m}	x_{n-m}	x_{n-m+1}	….	x_{n-2}	x_{n-1}	x_n

The scheme combining the SVM with the time series for forecasting is illustrated in Figure 6.3.

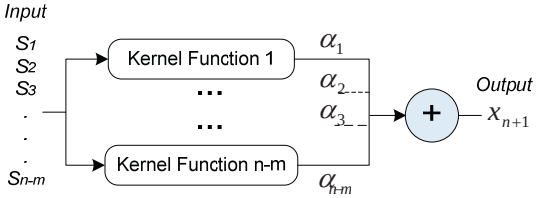


Figure 6.3 SVM Combined with Time Series

The prediction accuracy of SVM is higher than that of the ANN (Fuqing *et al.*, 2011a, Chen, 2007).

6.4 System Reliability Assessment

The SVM has also been used to evaluate system reliability. Claudio has proposed a series of models using the SVM to evaluate system reliability based on the Monte Carlo simulation (Rocco and Moreno, 2002). In these models, the SVM plays a role as a learning system. The SVM is trained by the state of the system and the state vector of its constituent components to perform as learning. After training, the logic configuration of the system is built into the SVM and the Monte Carlo simulation is used to access system reliability (Rocco and Moreno, 2002, Rocco and Muselli, 2005). Fuqing *et al.* also use the SVM to access system reliability. One of their papers accesses system reliability from the derived path sets and cuts sets using SVM, thereafter evaluating system reliability using these path sets and cut sets (Fuqing *et al.*, 2009). Another paper on system reliability considers using the SVM to make up missing or incomplete data (Fuqing; *et al.*, 2011).

The following example demonstrates the use of the SVM to access system reliability. The configuration of the network is shown in Figure 6.4. When 1 denotes the functional state of the element or system, 0 denotes the failed state. The vector of the element states, e.g. <1,0,1,1,1>, is proposed as the input of the SVM; the corresponding output is the state of the whole system, i.e. 0 or 1. After training the SVM, the logic configuration of the system is built into the decision function (Fuqing *et al.*, 2011) in the SVM.

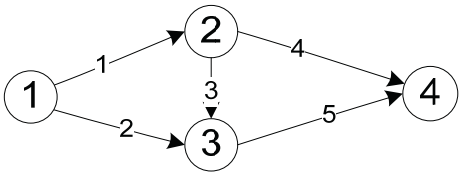


Figure 6.4 A Simple Bridge System

Thereafter, the element state is simulated by Monte Carlo simulation. The corresponding system state of the simulated element vector is predicted by the SVM decision function. After repeating this procedure one thousand or ten thousand times, the reliability is estimated by the ratio of number of functional system states out of the total number of iterations. The error of this method is very low when the number of iterations is sufficiently large.

7. Summary of Appended Papers

7.1 Paper I

Reliability predicting using Support Vector Regression

This paper uses Support Vector Regression (SVR) and a time series to predict time to next failure (TTF). A n-step lagged time series is constructed to accommodate the historical data and is used as the SVR input. As the kernel parameter is crucial to the SVR performance, this paper proposes to simplify the widely used but computationally costly Leave-One-Out (LOO) method to find the optimal kernel parameter. It proposes a Non-Homogeneous Poisson Process (NHPP) based confidence interval in order to bound the predicted TTF from SVR. The confidence interval is used because there is no guarantee that the best parameters of SVR have been obtained. This interval prevents unreasonable predictions.

A numerical case study is presented to demonstrate the approach. In this case study, the step size of 1 to 4 of the time series is used to predict the TTF. In terms of the prediction error, the best is the 3-step lagged time series. For comparison, this paper also predicts the TTF using the NHPP based method. It finds that the 3-step lagged SVR has lower prediction error than the NHPP based method. Thus, the simplified LOO can effectively find the optimal parameters, but the computational cost is high.

7.2 Paper II

Fault Diagnosis of Railway Assets using Support Vector Machine and Ant Colony Optimization Method

The Support Vector Machine (SVM) is an excellent classifier to discriminate patterns. This paper uses the extension of the SVM, multi-class SVM, to identify the failure patterns of electric motor in a railway system. The signal analyzed in this paper is the power consumption by the electric motor installed in the Switch and Crossing system. The paper identifies three motor states: normal state; minor lack of lubrication; and critical lack of lubrication. As this is a 3-class classification problem, the multi-class SVM is used to discriminate patterns.

When there is a lack of lubrication in a switch operation, the action taking place in the “translation” stage of the electric motor consumes more power. Therefore, the power consumption in the “translation” stage can be used as a condition indicator of the switch and diagnose the “lack of lubrication” phenomenon in its early stages. In this paper, the features extracted from the signal are the following: maximum value, minimum value, mean, and deviation of the power consumption, as well as the coefficients of polynomial fit on the shape of the signal. These features are used as input of the 3-class SVM.

As the parameters in the kernel function significantly influence the performance of the SVM classifier, this paper proposes the heuristic Ant Colony Optimization (ACO) algorithm to find the optimal parameters. The ACO mimics the method by which ants find the optimal route to a food source. They release pheromones in the travelled route; the route with the shortest path will end up with the most pheromones. This shortest path corresponds to the best solution. In this paper, the optimal route is the optimal parameter in the kernel function.

A case study using the simulation data illustrates the proposed method. Using the optimal parameters obtained from the ACO, we find that the accuracy of the performed fault diagnosis on the electric motor is higher than when we use other parameter. In the simulation case study, the diagnosis accuracy reached over 98%.

7.3 Paper III

An Adaptive Multiple-kernel Method based Support Vector Machine for Classification

This paper has developed a multiple kernel function method SVM. This multiple kernel function need not predefine the parameter in the kernel function so it is named the adaptive multiple kernel function SVM (AMK-SVM). The multiple kernel method is suitable for a situation with a mixture of continuing, categorical, or logical features. A single kernel function to fit such features mixture is infeasible, but the AMK-SVM can accommodate such heterogeneous features naturally.

The SVM is a representative of the kernel method. This paper begins by discussing the principle of the kernel method. The kernel method is a tactic used to transform the original input data from a lower dimensional space into a higher dimensional space. The paper discusses how the data sets in the higher dimensional space are represented. Linearly non separable data sets in the original space may become linearly separable in the higher dimensional space, without compromising the computational cost.

The kernel function of a kernel method with a better separability is prone to induce a better classier. Kernel Alignment (KA) is a method which can be used to measure the separability of a kernel function. It measures the similarity between the matrix produced by the kernel function and the perfect target matrix. Because the multiple kernel method combines several kernel functions, it can achieve a higher KA and hence be used to improve the performance of the support vector classifier. Theoretically, multiple kernel functions can outperform the single kernel function as the latter is a special case of the multiple kernel function.

However, in practise, the MK method performs well only when the MK is combined appropriately. Optimizing the coefficients for each kernel is necessary. Furthermore, the kernel parameters significantly affect the performance of SVM. This paper shows how to automatically select the optimal kernel parameters using the KA method. The selection of these parameters and coefficients is a challenge. Fortunately, kernel alignment does not

require a time-consuming training process. Based on the kernel alignment, the paper proposes an optimization program to obtain the optimal parameters and coefficients.

In this paper, a numerical example using benchmarking data demonstrates the feasibility of the developed approach and tests its performance. Two single kernel SVMs are applied to the same data sets to compare their performance with the AMK-SVM. In terms of accuracy, the developed AMK-SVM performs better than the single kernel SVM. However, in terms of computational cost, the single kernel SVM is superior. Therefore, in applications where accuracy is prioritized over computational cost, the AMK-SVM is best. Inversely, for applications when computational cost is prioritized, the single kernel SVM is better.

7.4 Paper IV

A Comparative Study of Artificial Neural Networks and Support Vector Machine for Fault Diagonis

This paper compares the performance of Artificial Neural Networks (ANN) and Support Vector Machine (SVM) in failure diagnostics. This paper only compares the standard SVM with the two-layer ANN, as they have similar complexity. Theoretically, both techniques can establish a decision function automatically from data, and moreover, the SVM has similar form of decision function with regularized ANN. But they differ in the following ways:

- The ANN is a black box method, while the SVM is transparent. The decision function of the ANN is more complex than that of the SVM.
- The ANN minimizes empirical error, but SVM minimizes the structural risk as shown in Chapter 3 in this thesis.
- The optimal ANN is a local minimum, and the optimal SVM is global minimum. The local minimum suggests the performance of ANN is not stable.

The measurements selected to measure the performance are the following: accuracy of fault diagnosis, computational cost, and performance stability. The case study shows that the accuracy the SVM allows it to outperform the two-layer ANN. In terms of computational cost, in this case, the SVM is also better than the ANN. In terms of stability, the ANN exhibits a random performance for each run with same training data. The SVM exhibits a stable performance as long as the data have not changed. Thus, the SVM has more stable performance.

7.5 Paper V

Fault Diagnosis on time domain for Rolling Element Bearings using Support Vector Machine

The rolling element bearing is important in industry. This paper analyzes the bearing's vibration signal from the time domain. Several existing time-domain statistical feature such as Kurtosis, Crest Factor, Normal Negative Likelihood (NNL), etc are discussed.

The time-domain feature has been argued to be computationally more efficient, more understandable, more easily implemented, and less sensitive to bearing load and speed variance.

In this paper, the existing NNL feature has been extended to a new non-dimensional feature: the Normalized NNL (NNNL). This feature is independent of the power of the signal. The paper also proposes two additional features: the Mean by Variance Ratio (MVR) and the Symbolized Sequence Shannon Entropy (SSSE). The MVR is also a non-dimensional feature. The SSSE is adopted from physics and can detect weak signals. It also considers the spatial distribution of the samples in the signal.

An Extended Separation Index (ESI), as the name suggests, is extended from an existing SI and is used here to remove the noise features. Thereafter, an exhaustive feature subset selection is used to select optimal subsets. The selected significant feature vector is used as the input of the Multi-Class Support Vector Machine (MSVM) for failure diagnostics.

A numerical case is presented to demonstrate the proposed features and approach. In this case, three different types of bearing defects are discriminated: defects in the inner race, the outer race and the bearing ball. The results show that by using the time domain features, these various types of failure can be discriminated with high accuracy. The proposed features MVR, SSSE, NNNL are in the subsets with high accuracy. The results also show that the new ESI is effective as an indicator of separability; the higher SI is prone to be highly accurate when single feature are used for diagnosis. In this case study, the linear polynomial kernel function is selected for diagnostics, as the linear kernel function shows the best performance. This result verifies that the more complex higher order model sometimes cannot outperform the simpler lower order model.

8. Discussion

8.1 Support Vector Machine as a classifier

As a failure diagnostics technique, the SVM is mostly used as a classifier in state-of-the-art research. As a classifier, the SVM requires fewer assumptions, thereby facilitating its ability to perform classification automatically. However, it implicitly assumes observations to be statistical independent. In failure diagnostics problems, this assumption has to be justified before using the SVM for classification. In the case study presented in Papers II and V, the signals can be considered approximately independent if they are measured at different times.

A classical classifier has to assume the decision function follows a specific form; for instance, it could be a linear or nonlinear form of the second or third order. Using a specific form with strict assumptions is possible only when one has sufficient knowledge of the problem. For instance, for a wheel motor, in order to determine failure when data on the shaft speed, oil sediment, viscosity and etc are available (Jardine, 2001), one can use the proportional hazard model which assumes that the covariant is linear with the logarithm of hazard rate (E.Love and R.Guo, 1991, Kumar, 1995, Love and Guo, 1991, Lugtigheid *et al.*, 2004).This linear assumption is strong and not realistic in some situations. The advantage of using the SVM to automatically diagnose failure is that it skips linear and nonlinear assumptions.

8.2 Small Sample Size Problem

In Vapnik's view, a small sample size is a problem (Vapnik, 1995) when the ratio of sample size by the dimension of the data $l/n < 20$. A small sample size is prone to "overfitting". A general framework to solve the small size problem is as follows (Bickel and Li, 2006):

$$\min_{f \in H} \left[\sum_{i=1}^{l} L(y_i, f(x_i)) + \gamma J(f) \right] \qquad (8.1)$$

Classical regression, ridge regression, and Lasso regression are implementations of this framework. In this framework, the loss function $L(...)$ can be any user-defined function, such as likelihood function, squared loss function, and Huber function. $J(f)$ is the penalty function. The SVM is one implementation of this framework. In the SVM, $J(f) = \frac{1}{2}\|w\|^2$. This is essentially the Maximal margin method.

The SVM is suitable for small problems, especially for $l/n \le 1$ due to its use of the kernel method. In the kernel method, the computational cost depends on the sample size l instead of the data dimension n. Therefore, for the problem where $l/n \le 1$, the computational cost can be reduced using the kernel method. The problem $l/n \le 1$ is common in medicine (Li and Gui, 2004, Noble, 2006), but could also occur in condition

monitoring. For instance, when several sensors are mounted in a machine measuring its vibration, one can extract dozens of features from each signal if the amplitudes of some harmonies are used as features. In this sense, the SVM can not only avoid the "overfitting" problem but can be computationally efficient.

The SVM also works well in larger sample size problems. This has been verified by using benchmarking data (Li and Gui, 2004, Sigillito, 1989). The case presented in Paper III is a large size problem as $l/n = 26$; here, the SVM performs well.

8.3 Kernel Parameter Selection

The parameters of the kernel function play an important role in the SVM. As mentioned in Section 4, the kernel function transforms data from a lower dimension to an implicit higher dimension. This implicit higher dimension space is determined by the parameters of the kernel function. As the SVM classifies the pattern in the implicit higher space, changing the kernel parameters will change the SVM performance. Selecting the appropriate parameters in the kernel function is an important step in the kernel method as this leads to good generalization ability, i.e. a low misclassification rate.

In addition to the parameters in the kernel function, as discussed in Section 5, the penalty C is critical to the SVM. In practice, the parameter C should be optimized along with the parameters in the kernel function. Figure 8.1 shows two figures from Paper II illustrating the influence of the kernel parameter and C to the misclassification rate. This figure empirically shows that the misclassification rate is sensitive to the kernel parameters and the penalty C. Remarkably, there are several minima in this figure so obtaining optimal parameters is a challenge.

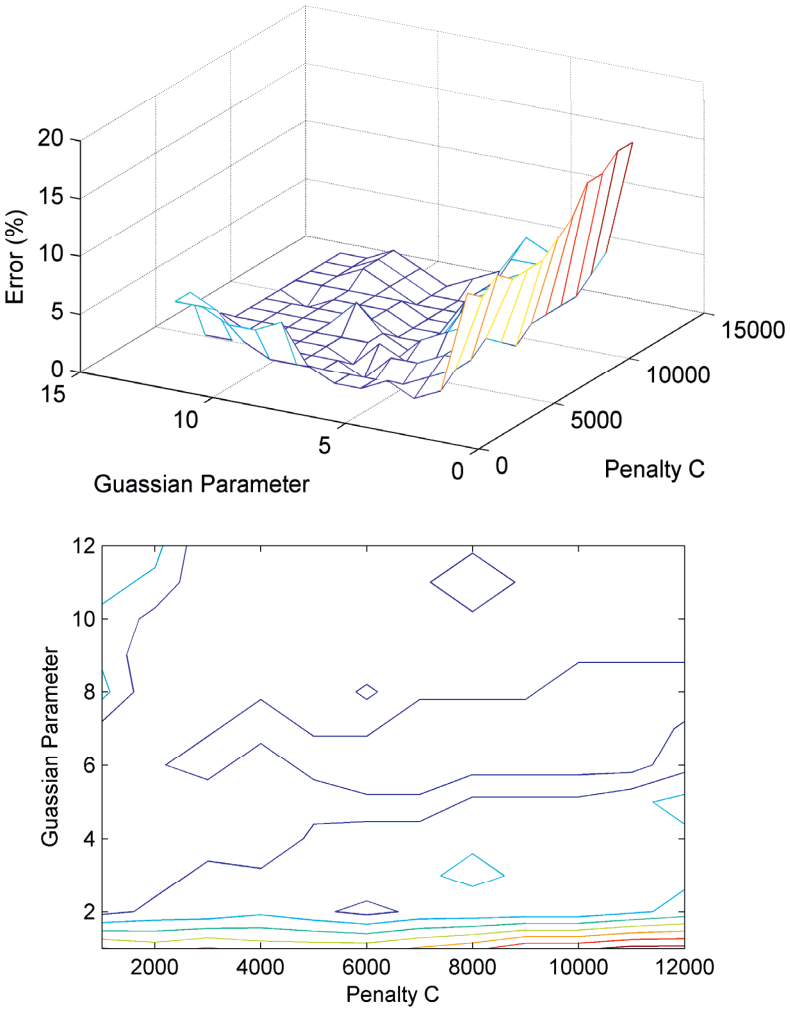


Figure 8.1 Misclassification error vs kernel parameter and C

The procedure to find the optimal parameters consists of two steps: finding an efficient method to evaluate the misclassification rate and finding the optimal parameters to minimize the misclassification rate. The important methods for selecting the evaluation misclassification rate are: the cross validation method; the support vector count; Xi-Alpha bound; the approximate span bound; the VC bound; the radius-margin method (Duan *et al.*, 2003); and the kernel alignment method (Lanckriet *et al.*, 2004). Cross validation divides the training data sets into two parts: one for training, the other for validating. The CV method is effective but computationally expensive. For large data sets, it is infeasible.

It is the basic method for parameter selection. The frequently used Leave-one-out (LOO) method is a special case of the CV but has a high computational cost. Paper I proposes a method to simplify the LOO method for regression. The case study shows the LOO is effective, but the computational cost is high.

The support vector count method is proved ineffective by our experiment and by experiments carried out by others. The VC bound method minimizes the bound discussed in Section 2.5. This theoretical bound is quite loose and therefore not an effective method. The approximate span bound has been proved ineffective by Duan *et al.* (2003). The Xi-alpha method is suitable only when the penalty value C is low. The Radius-margin is widely used; however, this method obtains a local minimum instead of a global minimum, i.e. the Radius-margin method cannot guarantee that a global minimal misclassification is obtained. The kernel alignment (KA) which is discussed in Paper III, assumes the higher KA value leads to higher generation ability. This method has computational efficiency, but the higher KA is not sufficient to guarantee a lower misclassification rate, as it could lead to an "overfitting" problem. In short, the most effective method for evaluating the misclassification rate is the CV method.

To select optimal parameters, the grid search is widely used. However, this method is exhaustive with an extremely high computational cost. There are some heuristic methods which improve the computational cost, such as genetic algorithm, simulated annealing and so on (Zhao *et al.*, 2011, Lin *et al.*, 2008). A heuristic method selects an initial solution heuristically and refines the solution iteratively until a local optimal solution is obtained. It then selects a new initial point, repeating the process until the predefined criterion is satisfied. This method cannot guarantee a global optimal solution but the computational efficiency is improved.

Paper II uses the heuristic Ant colony optimization (ACO) method to find the minimal misclassification rate, and the CV method is used to measure the misclassification rate. The ACO is inspired by how ants find the shortest path to a food source (Dorigo and Blum, 2005). As the case study in Paper II shows, the ACO is suitable for mid-scale data sets. However, for a large scale data set problem, this method is inefficient.

For large scale data sets, the KA method or the Radius-margin method should be used to find the optimal parameters as they can avoid the time-consuming training process. The selection of the SVM parameter is a trade-off between accuracy and computational efficiency.

8.4 Improvement on kernel function

The kernel function significantly influences the performance of the SVM, as it implicitly transforms the original input space to a higher dimensional space. Therefore, modifying the kernel function is one way to improve the performance of the SVM. The literature notes that the performance of the SVM can be improved by modifying the kernel function using the multiple kernel method (Gönen and Alpaydın, 2011, Kandola *et al.*, 2002, Subrahmanya and Shin, 2010, Rakotomamonjy *et al.*, 2008), the spectral method

(Argyriou *et al.*, 2010) and methods increasing the Riemannian geometrical distance (Amari and Wu, 1999).

The multiple kernel method combines several kernel functions linearly to improve the classification ability (Lanckriet *et al.*, 2004). As discussed in Paper III, the multiple kernel function can outperform the single kernel function. For failure diagnostics, the multiple kernel function can also fuse data from different data sources naturally, as demonstrated in Figure 8.2 (Gönen and Alpaydın, 2011).

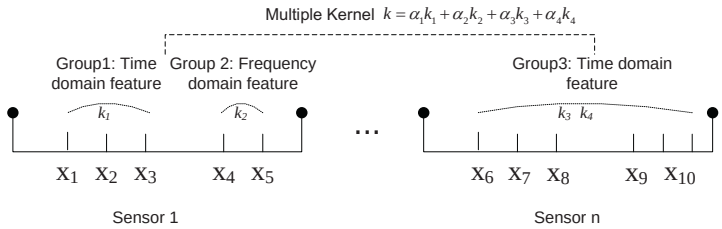


Figure 8.2 Using multiple kernel to combine various data sources

Moreover, the classification ability of the multiple kernel functions can be improved compared with single kernel function. The kernel alignment value which suggests the separability of kernel function for multiple kernel can be much higher than single kernel function. However, this improvement conditioned on the appropriate parameter estimation. For instance, the parameter α_i for each kernel function in Figure 8.2. The estimation of these parameters introduces extra computational costs. In a situation where a fast response is required, for example, in an online learning system, the multiple kernel method may not be appropriate.

8.5 Support Vector Machine compared with Artificial Neural Networks

The SVM and the Artificial Neural Network (ANN) are similar techniques and are generally applicable to the same problems. There are numerous variants of the ANN and the SVM. In this section, the ANN refers to the classical Feed Forward Neural Networks and the SVM refers to the SVM defined in Chapter 3. Paper IV compares the similarities and differences in their principles and some of their characteristics, as shown in Tables 8.1 and 8.2.

Table 8.1 Similarities of ANN and SVM in principle

Similarities	Description
Adaptability	Automatically learning from data
Error Minimization Leading	Need to minimize empirical error
Implementation	Computed by computer
Flexibility	Can be used nonlinearly

Table 8.2 Differences of ANN and SVM in principle

Differences	ANN	SVM
Transparency	Black box	Transparency
Optimization objective	Minimize empirical error	Structural risk minimization
Optimum Solution	Local minimum	Global minimum
Robustness	Low	High

Their performance is diagrammatically shown in Figure 8.3. In general, the accuracy and stability of the SVM is better. In term of computational cost, as shown in Paper IV, the SVM is lower. The computational cost depends on the size of the problem. For a small scale data set problem, the SVM can be faster than the ANN. Similar results have been shown by Samanta *et al.* (2003) and Samanta (2004).

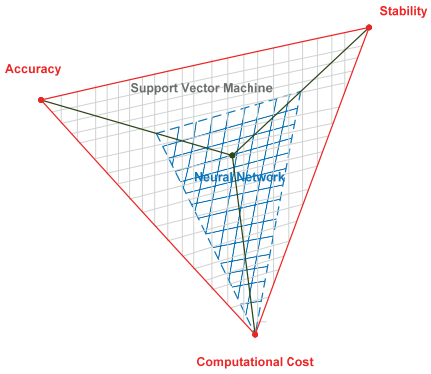


Figure 8.3 Performance of SVM vs Neural Networks

8.6 Failure Diagnostics using Support Vector Machine

One reason to use the SVM for failure diagnostics is that it can automatically obtain a decision function from the data, and this decision function can adapt to new data. This is called the self-learning ability in artificial intelligence. In an “automatic” method, fewer assumptions are required and there are fewer man interruptions. In this sense, the SVM is suitable in a scenario where one has little knowledge of the data. Another reason to use the SVM for failure diagnostics is the flexibility of its decision function. The decision function, which can be linear or nonlinear, is flexible when it is used to separate patterns, and it can automatically adjust to data. This is useful, as non-linear separable patterns frequently occur in failure diagnostics.

The SVM only deals with the problem of separating data; in failure diagnostics, selecting the input data is also very important. Support vector machine used for failure diagnostics can either use the original raw data or the feature vector as input. The raw data are normally huge; for example, the bearing signal in the case studied in Paper V has 12,000 points for a single second. These data are large, so the computational cost is extremely high if one uses the raw data as SVM input. The SVM is sensitive to noise data. Therefore, using the feature vector instead of the raw signal is a better option.

Feature extraction and selection is a crucial step in failure diagnostics. The feature can be extracted from the time domain, frequency domain or others. The frequency domain is effective for analyzing periodic signals, such as the bearing defect signal. When a signal does not exhibit frequency characteristics, one must extract features from the time domain as in Paper II. For feature selection, the SVM is special as it does not discriminate the patterns in the original feature space but in a transformed implicit space induced by the kernel function. This characteristic suggests that some feature selection methods performed in the original feature space are not feasible. Paper V uses the exhaustive feature subsets selection method; it selects the subset with the highest classification rate as optimal. This method is crude but efficient. The only disadvantage is its high computational cost.

Finally, the results of case study presented in Papers II, IV and V suggest using the simpler SVM model to perform failure diagnostics, as it has lower computational costs and could be more accurate. A more complex model, for example, the higher order polynomial SVM, has more a flexible decision function. However, as shown in Paper V, the more flexible SVM does not perform well.

8.7 Summary

Feature extraction and selection is crucial for failure diagnostics. This thesis discusses the feature extraction from time domain in both paper II and V. Using feature vector as input, the SVM is used as classifier for failure diagnostics as shown in paper II, IV, V and used for regression in the prediction in Paper I. The theoretical foundation of SVM is solid and thus the accuracy is high and the performance is stable. The sensitivity of kernel parameter entail for using heuristics method such as ant colony optimization to select optimal parameters. Utilizing multiple kernel function is necessary when the accuracy is prioritized over the computational efficiency, and when the different kernel is used to accommodate different data source. Moreover, in some cases both the accuracy and computational cost of SVM can outperform neural networks. Therefore, using the SVM for failure diagnostics is effective and efficient. This is the reason why the SVM is used for data mining in IT industry where huge data sets available. In engineering application, SVM has its distinct advantage for small sample size problem, as the regularization of SVM can control the “overfitting” problem when small sample size presented.

9. Conclusion

The accurate diagnostics of failure depends on both the feature extraction and use of the SVM as a pattern recognition technique. Feature extraction determines how precisely the raw signal is represented by the features. The SVM determines how accurately the decision function can separate the patterns. This thesis investigates the time domain feature extraction for vibration signals in rolling bearings and the power consumption signal in an electrical motor. It develops a new feature which extends the existing Normal Negative likelihood feature for bearing defects. As Paper V shows, this new feature can be used to diagnose bearing defects. Other features such as mean to variance, polynomial coefficients, Mean, Max and so on are also used to discriminate the patterns. The case study in Paper II shows that these features are effective in failure diagnostics for the presented case.

In the SVM, pattern recognition is performed in the feature space induced by the kernel function. The parameters of the kernel function determine the induced space; thus, these parameters significantly influence the performance of the SVM. As state-of-the-art research and the results show, the best parameter selection method is still the crude cross-validation method. This method is computationally costly, but this problem can be solved by using heuristic methods. This thesis uses the heuristic Ant Colony Optimization (ACO) to find the optimal parameters. As the case study in Paper II shows, the accuracy of failure diagnostics improves by using ACO.

This thesis proposes improving failure diagnostics by using multi-kernel method. In the multi-kernel SVM, the kernel parameters and the coefficient of each kernel are automatically adapted to the training data using the kernel alignment. This multi-kernel method can be used to naturally fuse data from various data sources. More importantly, as the case study in Paper III shows, the multi-kernel method can outperform the single kernel method in term of accuracy.

The more complex SVM model, for example the higher order polynomial kernel function or the multi-kernel method with more kernel functions, has a more flexible decision function, but the desired generalization ability does not always increase with the complexity of the SVM. In the case presented in Paper V, the less complex model has greater accuracy. Therefore, due to its computational efficiency and accuracy, a simpler SVM model is preferred.

This thesis presents the theoretical basis for the support vector machine, which shows the SVM has a solid theoretical foundation. The thesis compares the classical ANN with the SVM. Although performance depends on a specific problem, in several comparisons, the ANN shows a more random performance than the SVM. The randomness of the ANN is partially due to the local minima solution; however, the SVM has a global unique solution. The SVM also shows higher accuracy when the complexity of the ANN and the SVM are similar. In terms of the computational cost, the results show the SVM's cost can be lower than the ANN's for small and median scale data sets.

The SVM is a flexible framework in which the model complexity can be controlled by the parameters in the kernel function. In this framework, one can design a specific kernel function; one can even design one's own SVM by using the regularization strategy on the kernel function. In this sense, the SVM provides a generic theoretical framework for one to adapt a specific SVM to a specific application.

10. Research Contribution and Future Research

10.1 Research contribution

The research contributions of this thesis, in short, can be described as:

- A simplified Leave One Out (LOO) method has been proposed for parameter selection in support vector regression. A confidence interval is derived by using the classical non-homogenous Poisson Process (NHPP); this interval can complement the SVM forecasting (Paper I).

- Development of a parameter selection method using the Ant Colony Optimization (ACO). The ACO together with the SVM is applied to identify the failure patterns of an electric motor in a railway system (Paper II).

- Development of a multiple kernel support vector machine (MK-SVM). This new MK-SVM can be adapted to the data automatically and all the parameters in the MK-SVM need not be predefined (Paper III).

- A comparative study evaluating the effectiveness of SVM over ANN, demonstrates that SVM is more effective for failure diagnostics. These two well-known techniques are compared both in efficiency and effectiveness (Paper IV).

- Development of a new statistical feature from the time domain for bearing. A new separation index is developed to test the significance of difference between two signals. The support vector machine is used to identify fault (Paper V).

10.2 Scope for future research

Theoretically, the support vector machine is a kernel method technique. In this thesis, we do not develop a new kernel function; instead, we use the common Gaussian kernel function and the polynomial function. In future research, new kernel function will be developed for special applications. Additionally, the SVM is a method that can overcome the "overfitting" problem to some extent. As "overfitting" still exists in some situations, future research will introduce a new regularization method to overcome "overfitting".

Feature extraction is an important step of failure diagnostics. This study discusses only the features from the time domain. Future research will look for new features in the frequency domain or the time-frequency domain. These features will be used as the input for the SVM to improve the accuracy of failure diagnostics.

References

AKBARYAN, F. & BISHNOI, P. R. 2001. Fault diagnosis of multivariate systems using pattern recognition and multisensor data analysis technique. *Computers & Chemical Engineering,* 25**,** 1313-1339.

AMARI, S., BARRON, A. R., BIENENSTOCK, E., GEMAN, S., BREIMAN, L., MCCLELLAND, J. L., RIPLEY, B. D., TIBSHIRANI, R., CHENG, B. & TITTERINGTON, D. M. 1994. Neural Networks - a Review from a Statistical Perspective - Comments and Rejoinders. *Statistical Science,* 9**,** 31-54.

AMARI, S. & WU, S. 1999. Improving support vector machine classifiers by modifying kernel functions. *Neural Networks,* 12**,** 783-789.

ANTONELLI, G., CACCAVALE, F., SANSONE, C. & VILLANI, L. Year. Fault Diagnosis for AUVs Using Support Vector Machines. *In:* Proceedings of the 2004 IEEE, 2004 New Orleans, LA, USA.

ARGYRIOU, A., MICCHELLI, C. A. & PONTIL, M. 2010. On Spectral Learning. *Journal of Machine Learning Research,* 11**,** 935-953.

B.SREEJITH, A.K.VERMA & A.SRIVIDYA 2008. Fault diagnosis of rolling element bearing using time-domain features and neural networks. *Proceedings of Third IEEE International Conference on Industrial and Information Systems ICIIS 2008.* Kharagpur,India.

BARABADY, J. & KUMAR, U. 2008. Reliability analysis of mining equipment: A case study of a crushing plant at Jajarm Bauxite Mine in Iran. *Reliability Engineering & System Safety,* 93**,** 647-653.

BARLOW, R. E. & PROSCHAN, F. 1965. *Mathematical theory of reliability,* New York,, Wiley.

BENNETT, K. P. & CAMPBELL, C. 2000. Support Vector Machines: Hype or Hallelujah? *SIGKDD* 2.

BICKEL, P. J. & LI, B. 2006. Regularization in statistics. *Test,* 15**,** 271-303.

BINT, M. D. 2008. The role of remote condition monitoring in a modern railway. *The 4th IET International Conference on Railway Condition Monitoring.* Derby,UK.

BSI, B. S. I. 1984. BS3811 Glossary of maintenance terms in Terotechnology. London: BSI.

BURGES, C. J. C. 1998. A tutorial on Support Vector Machines for pattern recognition. *Data Mining and Knowledge Discovery,* 2**,** 121-167.

CAMCI, F. & CHINNAM, R. B. 2008. General support vector representation machine for one-class classification of non-stationary classes. *Pattern Recognition,* 41**,** 3021-3034.

CAMCI, F., CHINNAM, R. B. & ELLIS, R. D. 2008. Robust kernel distance multivariate control chart using support vector principles. *International Journal of Production Research,* 46**,** 5075-5095.

CASTRO, A. R. G. & MIRANDA, V. 2005a. Improving the IEC table for transformer failure diagnosis with knowledge extraction from neural networks. *Ieee Transactions on Power Delivery,* 20**,** 2509-2516.

CASTRO, A. R. G. & MIRANDA, V. 2005b. An interpretation of neural networks as inference engines with application to transformer failure diagnosis. *International Journal of Electrical Power & Energy Systems,* 27**,** 620-626.

CASTRO, A. R. G. & MIRANDA, V. 2005c. Knowledge discovery in neural networks with application to transformer failure diagnosis. *Ieee Transactions on Power Systems,* 20**,** 717-724.

CHEN, K. Y. 2007. Forecasting systems reliability based on support vector regression with genetic algorithms. *Reliability Engineering & System Safety,* 92**,** 423-432.

CHEN, Y. M. & LEE, M. L. 2002. Neural networks-based scheme for system failure detection and diagnosis. *Mathematics and Computers in Simulation,* 58**,** 101-109.

CHENG, B. & TITTERINGTON, D. M. 1994. Neural Networks - a Review from a Statistical Perspective. *Statistical Science,* 9**,** 2-30.

CHERKASSKY, V. S. & MULIER, F. 2007. *Learning from data : concepts, theory, and methods,* Hoboken, N.J., IEEE Press : Wiley-Interscience.

CHERNOFF, H. 1952. A Measure of Asymptotic Efficiency for Tests of a Hypothesis Based on the sum of Observations. *The Annals of Mathematical Statistics* 23**,** 493-507.

COX, D. R. 1972. Regression models and life-tables. *Biometrika,* 62**,** 187-219.

CRISTIANINI, N. & SHAWE-TAYLOR, J. 2000. *An introduction to Support Vector Machines : and other kernel-based learning methods,* Cambridge, Cambridge University Press.

DASH, M. & LIU, H. A. 2003. Consistency-based search in feature selection. *Artificial Intelligence,* 151**,** 155-176.

DAUBECHIES, I. 1990. The Wavelet Transform, Time-Frequency Localization and Signal Analysis. *Ieee Transactions on Information Theory,* 36**,** 961-1005.

DAVY, M., DESOBRY, F., GRETTON, A. & DONCARLI, C. 2006. An online support vector machine for abnormal events detection. *Signal Processing,* 86**,** 2009-2025.

DE SILVA, C. W. 2005. *Vibration and shock handbook,* Boca Raton, Taylor & Francis.

DEVROYE, L., GYÖRFI, L. & LUGOSI, G. B. 1996. *A probabilistic theory of pattern recognition,* New York, Springer.

DORIGO, M. & BLUM, C. 2005. Ant colony optimization theory: A survey. *Theoretical Computer Science,* 344**,** 243-278.

DUAN, K., KEERTHI, S. S. & POO, A. N. 2003. Evaluation of simple performance measures for tuning SVM hyperparameters. *Neurocomputing,* 51**,** 41-59.

E.LOVE, C. & R.GUO 1991. Using proportional hazard modeling in plant maintenance. *Quality and reliability engineering international,* 7**,** 7-17.

EBERSBACH, S. & PENG, Z. 2008. Expert system development for vibration analysis in machine condition monitoring. *Expert Systems with Applications,* 34**,** 291-299.

F.CHAMROUKHI, A.SAME, P.AKNIN & M.ANTONI. Year. Switch mechanism diagnosis using a pattern recognition approach. *In:* Railway Condition Monitoring, 2008 4th IET International Conference on, 18-20 June 2008 Derby 1-4.

F.ZHOU, M.DUTA, M.HENRY, S.BAKER & C.BURTON. Year. Condition monitoring and validation of railway point machines *In:* Intelligent and Self-Validating Instruments -- Sensors and Actuators (Ref. No. 2001/179), IEE Seminar on, 2001. 1-7.

FUQING, Y., KUMAR, U. & MISRA, K. B. 2011. Complex System Reliability Evaluation using Support Vector Machine for Incomplete Data-set. *International journal of Performability Engineering,* 7**,** 32-42.

FUQING, Y., KUMAR, U., S., C. M. R. & MISRA, K. B. 2009. Complex System Reliability Evaluation using Support Vector Machine. *SMRLO10.* Israel.

FUQING;, Y., KUMAR;, U. & GALAR;, D. 2010. Reliability Prediction using Support Vector Regression. *international journal of systems assurance engineering and management*.

FUQING;, Y., KUMAR;, U. & MISRA;, K. B. 2011. Complex System Reliability Evaluation using Support Vector Machine for Incomplete Data-set. *International journal of Performability Engineering,* 7**,** 32-42.

G.V.TRUNK 1979. A problem of dimensionality: a simple example. *Ieee Transactions on Pattern Analysis and Machine Intelligence,* 1**,** 306-307.

GARTNER, T. 2008. *Kernels for structured data,* Hackensack, NJ ; London, World Scientific.

GEMAN, S., BIENENSTOCK, E. & DOURSAT, R. 1992. Neural Networks and the Bias Variance Dilemma. *Neural Computation,* 4**,** 1-58.

GRIFFIN, D. W. & LIM, J. S. 1984. Signal Estimation from Modified Short-Time Fourier-Transform. *Ieee Transactions on Acoustics Speech and Signal Processing,* 32**,** 236-243.

GUNN, S. R. 1998. Support Vector Machines for Classification and Regression. SOUTHAMPTON: UNIVERSITY OF SOUTHAMPTON.

GUYON, I. & ELISSEEFF, A. 2006. An introduction to feature extraction. *In:* GUYON, I., GUNN, S., NIKRAVESH, M. & ZADEH, L. (eds.) *Feature Extraction, Foundations and Applications* Berlin: Springer.

GÖNEN, M. & ALPAYDıN, E. 2011. Multiple Kernel Learning Algorithms *Journal of Machine Learning Research,* 12**,** 2211−2268.

HAN, Y. & SONG, Y. H. 2003. Condition monitoring techniques for electrical equipment - A literature survey. *Ieee Transactions on Power Delivery,* 18**,** 4-13.

HAYTON, P., SCH¨OLKOPF, B., TARASSENKO, L. & ANUZIS, P. 2001. *Support vector novelty detection applied to jet engine vibration spectra,* MIT Press.

HIPPERT, H. S., PEDREIRA, C. E. & SOUZA, R. C. 2001. Neural networks for short-term load forecasting: A review and evaluation. *Ieee Transactions on Power Systems,* 16**,** 44-55.

HONG, D. H. & HWANG, C. H. 2003. Support vector fuzzy regression machines. *Fuzzy Sets and Systems,* 138**,** 271-281.

HSU, C. W. & LIN, C. J. 2002. A comparison of methods for multiclass support vector machines. *Ieee Transactions on Neural Networks,* 13**,** 415-425.

JARDINE, A. K. S. 2001. Optimizing a mine haul truck wheel motors' condition monitoring program Use of proportional hazards modeling. *Journal of Quality in Maintenance Engineering,* 7**,** 286.

JARDINE, A. K. S., BANJEVIC, D. & MAKIS, V. 1997. Optimal replacement policy and the structure of software for condition-based maintenance. *Journal of Quality in Maintenance Engineering,* 3**,** 109 - 119.

JARDINE, A. K. S., JOSEPH, T. & BANJEVIC, D. 1999. Optimizing condition-based maintenance decisions for equipment subject to vibration monitoring. *Journal of quality in maintenance engineering,* 5**,** 192 - 202.

JARDINE, A. K. S., LIN, D. M. & BANJEVIC, D. 2006. A review on machinery diagnostics and prognostics implementing condition-based maintenance. *Mechanical Systems and Signal Processing,* 20**,** 1483-1510.

K.LIU, W.H.SIEW, R.W.STEWART & Y.WANG 2008. Smart wireless railway monitoring system *The 4th IET International Conference on Railway Condition Monitoring.* Derby, UK.

KAHRAMAN, F. & ORAL, M. 2001. Artificial neural-network-based face detection in colour and gray-scale image in complex scenes. *Perception,* 30**,** 79-79.

KANDOLA, J., SHAWE-TAYLOR, J. & CRISTIANINI, N. 2002. optimizing kernel alignment over combinations of kernels. University of London.

KERMIT, M., EIDE, A. J., LINDBLAD, T. & WALDEMARK, K. 2000. Treatment of obstructive sleep apnea syndrome by monitoring patients airflow signals. *Pattern Recognition Letters,* 21**,** 277-281.

KIRA, K. & RENDELL, L. A. Year. A practical approach to feature selection. *In:* ML92 Proceedings of the ninth international workshop on Machine learning, 1992 San Francisco, CA, USA. Morgan Kaufmann

KLEFSJO, B. & KUMAR, U. 1992. Goodness-of-Fit Tests for the Power-Law Process Based on the Ttt-Plot. *Ieee Transactions on Reliability,* 41**,** 593-598.

KUMAR, D. 1995. Proportional Hazards Modeling of Repairable Systems. *Quality and Reliability Engineering International,* 11**,** 361-369.

KUMAR, U., KLEFSJO, B. & GRANHOLM, S. 1989. Reliability Investigation for a Fleet of Load Haul Dump Machines in a Swedish Mine. *Reliability Engineering & System Safety,* 26**,** 341-361.

LAGNEBÄCK, R. 2007. *Evaluation of wayside condition monitoring technologies for condition-based maintenance of railway vehicles.* Licetiate, Luleå University of Technology.

LANCKRIET, G. R. G., CRISTIANINI, N., BARTLETT, P., EL GHAOUI, L. & JORDAN, M. I. 2004. Learning the kernel matrix with semidefinite programming. *Journal of Machine Learning Research,* 5**,** 27-72.

LAWLESS, J. F. 1987. Regression Methods for Poisson-Process Data. *Journal of the American Statistical Association,* 82**,** 808-815.

LI, H. & GUI, J. 2004. Partial Cox regression analysis for high-dimensional microarray gene expression data. *Bioinformatics,* 20 Suppl 1**,** i208-15.

LI, H. & LUAN, Y. 2003. Kernel Cox regression models for linking gene expression profiles to censored survival data. *Pac Symp Biocomput***,** 65-76.

LIN, C. F. & WANG, S. D. 2002. Fuzzy support vector machines. *Ieee Transactions on Neural Networks,* 13**,** 464-471.

LIN, S. W., LEE, Z. J., CHEN, S. C. & TSENG, T. Y. 2008. Parameter determination of support vector machine and feature selection using simulated annealing approach. *Applied Soft Computing,* 8**,** 1505-1512.

LOVE, C. E. & GUO, R. 1991. Application of Weibull proportional hazards modeling to bad-as-old failure data. *Quality and reliability engineering international,* 7**,** 149-157.

LUENBERGER, D. G. & YE, Y. 2008. *Linear and nonlinear programming,* New York, Springer.

LUGTIGHEID, D., BANJEVIC, A. & JARDINE, A. K. S. 2004. Modeling repairable system reliability with explanatory variables and repair and maintenance actions. *IMA Journal of Management Mathematics,* 15**,** 89-110.

MAIER, H. R. & DANDY, G. C. 2000. Neural networks for the prediction and forecasting of water resources variables: a review of modelling issues and applications. *Environmental Modelling & Software,* 15**,** 101-124.

MARTIN, K. F. 1994. A Review by Discussion of Condition Monitoring and Fault-Diagnosis in Machine-Tools. *International Journal of Machine Tools & Manufacture,* 34**,** 527-551.

MATHEW, J. & ALFREDSON, R. J. 1984. The Condition Monitoring of Rolling Element Bearings Using Vibration Analysis. *Journal of Vibration Acoustics Stress and Reliability in Design-Transactions of the Asme,* 106**,** 447-453.

NARENDRA, P. M. & FUKUNAGA, K. 1977. Branch and Bound Algorithm for Feature Subset Selection. *IEEE TRANSACTIONS ON COMPUTERS,* 26**,** 917-922.

NGUYEN, M. H. & DE LA TORRE, F. 2010. Optimal feature selection for support vector machines. *Pattern Recognition,* 43**,** 584-591.

NISSEN, A. 2009. *Development of Life Cycle Cost Model and Analyses for Railway Switches and Crossings.* Doctoral Thesis, Luleå University of Technology.

NOBLE, W. S. 2006. What is a support vector machine. *Nature Biotechnology,* 24**,** 1565-1567.

O.F.EKER & F.CAMCI 2010. Failure Diagnostics on Railway Turnout Systems using Support Vector Machine. *E-Maintenance 2010.* Luleå,Sweden.

ONODA, T., ITO, N. & HIRONOBU, Y. 2008 Unusual condition monitoring based on support vector machines for hydroelectric power plants *IEEE Congress on Evolutionary Computation, 2008.* Hong Kong.

PFEUFER, T. & AYOUBI, M. 1997. Application of a hybrid neuro-fuzzy system to the fault diagnosis of an automotive electromechanical actuator. *Fuzzy Sets and Systems,* 89**,** 351-360.

QIU, W. L. & JOE, H. 2006. Separation index and partial membership for clustering. *Computational Statistics & Data Analysis,* 50**,** 585-603.

RAKOTOMAMONJY, A., BACH, F. R., CANU, S. & GRANDVALET, Y. 2008. SimpleMKL. *Journal of Machine Learning Research,* 9**,** 2491-2521.

ROBERTS, C., DASSANAYAKE, H. P. B., LEHRASAB, N. & GOODMAN, C. J. 2002. Distributed quantitative and qualitative fault diagnosis: railway junction case study. *Control Engineering Practice,* 10**,** 419-429.

ROCCO, C. M. & MORENO, J. A. 2002. Fast Monte Carlo reliability evaluation using support vector machine. *Reliability Engineering & System Safety,* 76**,** 237-243.

ROCCO, C. M. & MUSELLI, M. 2005. Approximate multi-state reliability expressions using a new machine learning technique. *Reliability Engineering & System Safety,* 89**,** 261-270.

ROWLEY, H. A., BALUJA, S. & KANADE, T. 1998. Neural network-based face detection. *Ieee Transactions on Pattern Analysis and Machine Intelligence,* 20**,** 23-38.

SAMANTA, B. 2004. Gear fault detection using artificial neural networks and support vector machines with genetic algorithms. *Mechanical Systems and Signal Processing,* 18**,** 625-644.

SAMANTA, B., AL-BALUSHI, K. R. & AL-ARAIMI, S. A. 2003. Artificial neural networks and support vector machines with genetic algorithm for bearing fault detection. *Engineering Applications of Artificial Intelligence,* 16**,** 657-665.

SAPANKEVYCH, N. L. & SANKAR, R. 2009. Time Series Prediction Using Support Vector Machines: A Survey. *Ieee Computational Intelligence Magazine,* 4**,** 24-38.

SARKAR, D. 1996. Randomness in generalization ability: A source to improve it. *Ieee Transactions on Neural Networks,* 7**,** 676-685.

SAUNDERS, C., GAMMERMAN, A. & HOLLOWAY, R. Year. Application of Support Vector Machines to Fault Diagnosis and Automated Repair. *In:* Proc. Of the 11 th Int. Workshop on Principles of Diagnosis, June 8-10 2000.

SCHÖLKOPF, B., BURGES, C. J. C. & SMOLA, A. J. 1999. *Advances in kernel methods : support vector learning,* Cambridge, Mass., MIT Press.

SCHÖLKOPF, B. & SMOLA, A. J. 2002. *Learning with kernels : support vector machines, regularization, optimization, and beyond,* Cambridge, Mass., MIT Press.

SHAWE-TAYLOR, J. & CRISTIANINI, N. 2004. *Kernel methods for pattern analysis,* Cambridge, Cambridge University Press.

SIEDLECKI, W. & SKLANSKY, J. 1989. A Note on Genetic Algorithms for Large-Scale Feature-Selection. *Pattern Recognition Letters,* 10**,** 335-347.

SIGILLITO, V. 1989. *Ionosphere Data Set* [Online]. Laurel,U.S.A. Available: http://archive.ics.uci.edu/ml/datasets/Ionosphere [Accessed 11.20 2010].

SOTIRIS, V. A., TSE, P. W. & PECHT, M. G. 2010. Anomaly Detection Through a Bayesian Support Vector Machine. *Ieee Transactions on Reliability,* 59**,** 277-286.

SUBRAHMANYA, N. & SHIN, Y. C. 2010. Sparse Multiple Kernel Learning for Signal Processing Applications. *Ieee Transactions on Pattern Analysis and Machine Intelligence,* 32**,** 788-798.

SUGISAKA, M. & FAN, X. J. 2005. An effective search method for neural network based face detection using particle swarm optimization. *Ieice Transactions on Information and Systems,* E88d**,** 214-222.

SUGUMARAN, V., SABAREESH, G. R. & RAMACHANDRAN, K. I. 2008. Fault diagnostics of roller bearing using kernel based neighborhood score multi-class support vector machine. *Expert Systems with Applications,* 34**,** 3090-3098.

TARNG, Y. S., HWANG, S. T. & HSEIH, Y. W. 1994. Tool Failure Diagnosis in Milling Using a Neural-Network. *Mechanical Systems and Signal Processing,* 8**,** 21-29.

TAX, D. M. J. & DUIN, R. P. W. 2004. Support vector data description. *Machine Learning,* 54**,** 45-66.

THEODORIDIS, S. 2010. *Introduction to pattern recognition a MATLAB approach* [Online]. Burlington, MA: Academic Press. Available: http://site.ebrary.com/lib/mitlibraries/docDetail.action?docID=10391584 [Accessed].

THEODORIDIS, S. & KOUTROUMBAS, K. 2006. *Pattern recognition,* San Diego, CA, Academic Press.

TRAFALIS, T. B. & GILBERT, R. C. 2006. Robust classification and regression using support vector machines. *European Journal of Operational Research,* 173**,** 893-909.

TU, J. V. 1996. Advantages and disadvantages of using artificial neural networks versus logistic regression for predicting medical outcomes. *Journal of Clinical Epidemiology,* 49**,** 1225-1231.

WANG, J. H. & SHEN, X. T. 2006. Estimation of generalization error: Random and fixed inputs. *Statistica Sinica,* 16**,** 569-588.

VAPNIK, V. N. 1995. *The nature of statistical learning theory,* New York, Springer.

VAPNIK, V. N. 1998. *Statistical learning theory,* New York, Wiley.

WIDODO, A., KIM, E. Y., SON, J. D., YANG, B. S., TAN, A. C. C., GU, D. S., CHOI, B. K. & MATHEW, J. 2009. Fault diagnosis of low speed bearing based on relevance vector machine and support vector machine. *Expert Systems with Applications,* 36**,** 7252-7261.

WIDODO, A. & YANG, B. S. 2007. Support vector machine in machine condition monitoring and fault diagnosis. *Mechanical Systems and Signal Processing,* 21**,** 2560-2574.

WU, S. & AMARI, S. I. 2002. Conformal transformation of kernel functions: A data-dependent way to improve Support Vector Machine classifiers. *Neural Processing Letters,* 15**,** 59-67.

Y.KIM, E., C.C.TAN, A., YANG, B.-S. & KOSSE, V. 2007. Experimental Study on Condition Monitoring of Low Speed Bearings:Time domain Analysis. *5th Australasian Congress on Applied Mechanics,ACAM2007.* Brisbane,Australia.

YU, L. & LIU, H. 2004. Efficient feature selection via analysis of relevance and redundancy. *Journal of Machine Learning Research,* 5**,** 1205-1224.

YUAN, S. F. & CHU, F. L. 2007. Fault diagnosis based on support vector machines with parameter optimisation by artificial immunisation algorithm. *Mechanical Systems and Signal Processing,* 21**,** 1318-1330.

ZHANG, H. B. & SUN, G. Y. 2002. Feature selection using tabu search method. *Pattern Recognition,* 35**,** 701-711.

ZHANG, Y. X. & RANDALL, R. B. 2009. Rolling element bearing fault diagnosis based on the combination of genetic algorithms and fast kurtogram. *Mechanical Systems and Signal Processing,* 23**,** 1509-1517.

ZHAO, M. Y., FU, C., JI, L. P., TANG, K. & ZHOU, M. T. 2011. Feature selection and parameter optimization for support vector machines: A new approach based on genetic algorithm with feature chromosomes. *Expert Systems with Applications,* 38**,** 5197-5204.

ZHU, J. & HASTIE, T. 2005. Kernel logistic regression and the import vector machine. *Journal of Computational and Graphical Statistics,* 14**,** 185-205.

ZHU, X. L., BEAUREGARD, G. T. & WYSE, L. L. 2007. Real-time signal estimation from modified short-time Fourier transform magnitude spectra. *Ieee Transactions on Audio Speech and Language Processing,* 15**,** 1645-1653.

APPENDED PAPERS

PAPER I: **Y. Fuqing**, U. Kumar and D. Galar, "Reliability Prediction using Support Vector Regression," *International Journal of Systems Assurance Engineering and Management*, Vol. 1, No. 3 pp. 263-268, 2010.

Int J Syst Assur Eng Manag (July-Sept 2010) 1(3):263–268
DOI 10.1007/s13198-011-0040-2

ORIGINAL PAPER

Reliability prediction using support vector regression

Yuan Fuqing · Uday Kumar · Diego Galar

Received: 11 October 2010 / Revised: 2 March 2011 / Published online: 19 April 2011

Abstract Reliability prediction of machinery is crucial to schedule overhauls, spare parts replacement and maintenance interventions. Many AI tools have been used in order to provide these predictions for the industry. Support vector regression (SVR) is a nonlinear regression technique extended from support vector machine. SVR can fit data flexibly and it has a wide variety of applications. This paper utilizes SVR combining time series to predict the next failure time based on historical failure data. To solve the parameter selection problem a method has been proposed. This method approximates the widely used leave-one-out method. To bound the prediction error, a confidence interval is proposed based on the non-homogeneous poisson process. A numerical case from the mining industry is presented to demonstrate the developed approach.

Keywords Support vector regression · Reliability prediction · Parameter selection · Time series · Non-homogeneous poisson process

1 Introduction

Support vector machine (SVM) is an artificial intelligence technique which can be used for prediction. The SVM has been verified as a good prediction method (Radhika and Shashi 2009). In general, SVM possesses characteristics as:

a. Uses maximum margin method: The basic of SVM is a binary classifier. Maximum margin aims to find a decision function to separate two classes of data as far as possible that is why SVM can predict the class of a new data more accurately.
b. Uses kernel method: Using a kernel function to transform the problem from a lower dimension to a higher dimension makes the decision function of SVM more flexible.
c. Possesses Sparseness property: Support Vectors (SVs) are a group of special datasets exerting an effect on the decision function. A smaller number of SVs tends to enable SVM to achieve better prediction ability.
d. Constructs convex optimization to obtain unique solution: The convex property makes the SVM to obtain its optimal solution easily.

The detailed principle and development of SVM is described in (Vapnik 1995, 1998).

SVR is the extension of SVM for regression analysis. The regression function of a SVR is an expansion of kernel functions. When a nonlinear kernel function is used, the regression function of SVR is accordingly nonlinear. This characteristic enables the SVR able to fit the data flexibly. Gonzalez et al. (2010) uses SVR to process image data. Paul et al. (2010) uses SVR to perform face detection. Yang and Shieh (2010) propose a method to predict consumers affective responses.

Reliability prediction is important in reliability engineering. The objective of reliability prediction varies with the stage of products life cycle in order to predict the remaining useful life. According to Foucher et al. (2002),

Y. Fuqing (✉) · U. Kumar · D. Galar
Division of Operation and Maintenance Engineering, Luleå University of Technology, 971 87 Luleå, Sweden
e-mail: yuan.fuqing@ltu.se

U. Kumar
e-mail: uday.kumar@ltu.se

D. Galar
e-mail: diego.galar@ltu.se

reliability prediction is to help achieving a reliable manufacturing process, establishing baseline for logistic support requirement. Accurate reliability prediction is crucial for maintenance policy optimization and system availability improvement. On the application of SVR to reliability data analysis, there are also some applications. Chen (2007) incorporated the SVM and genetic algorithms to predict reliability. Xing and Guo (2005) proposed a method to predict software reliability.

In this paper, a method using SVR incorporated with a time-series technique to perform reliability prediction for complex repairable system is proposed. Section 2 describes the principle of SVR. Section 3 discusses the development of SVR model, the procedure to perform reliability prediction using SVR, and the confidence interval for the SVR. Section 4 presents a numerical case study to illustrate the proposed method. Section 5 presents discussion and future work.

2 Support vector regression

As mentioned before, SVR is an extension of the SVM. SVR uses a loss function to define the degree of penalty when the estimated value deviates from the real value. The loss function of the SVR is usually called ε-insensitive function (Schölkopf and Smola 2002). This ε-insensitive function defines a tube: inside the tube, there is no penalty for deviation; while outside of the tube, a penalty occurs for any deviation. ε defines the size of this tube which is used to balance the accuracy of approximation and the computational complexity.

The primal problem of ε-insensitive SVR is defined as follows (Schölkopf and Smola 2002):

$$\begin{aligned} &\min \frac{1}{2}||w||^2 + C\sum_{i=1}^{m}(\xi_i + \xi_i^*) \\ &s.t.\ f(x_i) - y_i \le \varepsilon + \xi_i \\ &\quad y_i - f(x_i) \le \varepsilon + \xi_i^* \\ &\quad \xi_i \ge 0,\ \xi_i^* \ge 0.\ i = 1, 2, 3, \ldots m. \end{aligned} \tag{1}$$

We transform the primal problem into a Lagrangian dual problem, which is:

$$\begin{aligned} \max\ W(\alpha, \alpha^*) = &-\varepsilon\sum_{i=1}^{m}(\alpha_i + \alpha_i^*) + \sum_{i=1}^{m}(\alpha_i^* - \alpha_i)y_i \\ &-\frac{1}{2}\sum_{i,j=1}^{m}(\alpha_i^* - \alpha_i)(\alpha_j^* - \alpha_j)\langle x_i, x_j\rangle \\ s.t.\ &\sum_{i=1}^{m}(\alpha_i - \alpha_i^*) = 0 \\ &\alpha_i^*, \alpha_i \in [0, C],\ i = 1, 2, 3, \ldots m \end{aligned} \tag{2}$$

In dual problem, kernel functions $K\ \langle x_i, x_j\rangle$ is used to substitute $\langle x_i, x_j\rangle$. The desired regression function is then:

$$f(x) = \sum_{i=1}^{m}(\alpha_i^* - \alpha_i)K(x, x_i) + b \tag{3}$$

The performance of SVR heavily relies on the kernel function. Some authors have proposed many different kernel functions (Schölkopf and Smola 2002; Vapnik 1995). The most common kernel functions are the Gaussian and the polynomial function. Equation of the Gaussian kernel function is (Burges 1998):

$$K(x, x') = \exp\left(-\frac{||x - x'||}{2\sigma^2}\right) \tag{4}$$

where the parameter σ defines the sensitivity of the Gaussian function. The equation of the polynomial function is:

$$K(x, x') = (\langle x, x'\rangle + 1)^d \tag{5}$$

where the parameter d is a discrete integer.

3 SVR model development

In this section, firstly input of SVR is discussed, and later the parameter selection and the development of confidence interval for the SVR will be discussed.

3.1 Input of SVR

The input of this SVR is a time series. Time series is able to accommodate historical data. Suppose that the datasets are $x_1, x_2, x_3 \ldots x_n$. From these datasets, for a m-step lagged time series, number of $n - m$ time series can then be formulated, as shown in Table 1.

3.2 Parameter selection process

Parameter selection is critical in SVR. An inappropriate SVR model will lead to an unreasonable regression function. Most used method for parameter selection found in literature survey is the leave-one-out (LOO) method (Cawley and Talbot 2003). This method takes out one portion of data to train the SVM, while using the remaining datasets to validate the model. The LOO method has been proven effective, however it is not considered efficient. In

Table 1 n-step lagged input array and its corresponding output

Time series	x(input)					y(output)
s_1	x_1	x_2	…	x_{m-1}	x_m	x_{m+1}
s_2	x_2	x_3	…	x_m	x_{m+1}	x_{m+2}
…	…	…	…	…	…	…
s_{n-m}	x_{n-m}	x_{n-m+1}	…	x_{n-2}	x_{n-1}	x_n

this section a more efficient parameter selection method is proposed.

Suppose that Gaussian function is used as the kernel function and one ε-insensitive function is used as the loss function. The parameters considered in SVR are:

- C (penalty parameter in SVR)
- ε (tube size of the loss function)
- σ (width of the Gaussian function).

The three parameters are crucial in prediction performance. Specially if you try to predict such a sensitive parameter as reliability then a minimal prediction error has to be achieved.

The analytical relationship between the parameters and the minimal prediction error cannot be explicitly expressed. Through some experimental results, we found that their relationship is nonlinear with numerous local optimums (Fuqing et al. 2009). Search the global optimal solution for such a problem is NP-hard (Hochbaum 1997). One simple solution is to generate parameter values heuristically and iteratively and then select the one with minimal prediction error from them.

Proposed method is as follows:

- A set of parameters (C, ε, σ) using heuristic algorithm such as ant colony or genetic algorithm is generated.
- Within the time series ($s_1 s_2 \ldots s_{n-m}$), we take the first h datasets to train the SVR and use the next $h + 1$th datasets to validate the trained SVR.
- Prediction error between real output and predicted output is calculated.
- Again, we use the first $h + 1$ of datasets to train the SVM and the $h + 2$th datasets as validation datasets, and so on.

The prediction error of a set of parameters is the sum of each step's (from h to n – m − 1) prediction error. Iterating this procedure at a predefined times, the optimal set of parameters is the set with the minimal prediction error.

The detailed parameter selection process is as follows:

1. Initialization: Define the number of iterations.
2. Generate a set of heuristic values for parameter C, ε and σ.
3. Take out the first s_h, e.g. (s_1, s_2, s_3) in ($s_1 s_2$ s_3;s_4), as shown in Fig. 1, to train the SVR, and use the data s_{h+1}, e.g. s_4 in ($s_1 s_2$ s_3;s_4), for validation.
4. Train the SVR and predict the $h + 1$th failure time $\widehat{x}_{h+1}$ using s_{h+1}. Compute the discrepancy between the predicted $\widehat{x}_{h+1}$ and the real x_{h+1}.
5. Set h = h + 1 and go to Step 3. Repeat this until h < n − m − 1.
6. Summarize all the discrepancies of steps from h to $n - m - 1$.

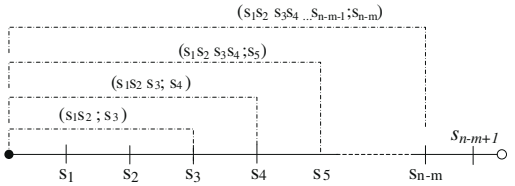


Fig. 1 Computing prediction error

7. Repeat step 2 to step 6 until the predefined number of iterations is reached.
8. Take the C, ε and σ with the minimal sum of discrepancies as the optimal parameters.

The LOO method is to take each time series out once. The taken out time series are used for validation. The remaining time series are used to train the SVR. As the number of time series is n-m, using the LOO method, one need to run the SVM for all the combinations of these time series. Therefore, the number of runs comes to $(n - m)$. Shown method has simplified the above-mentioned LOO. In this method, the number of runs needed is only $n - m - h$ so the computational complexity has been reduced.

3.3 Predicting reliability using SVR

A set of optimal parameter (C, ε, σ) can be obtained from the parameter selection process described in the previous section. These parameters are used to train the SVR. After training, one optimal decision function, which is defined in Formula (3), is obtained. The x and x_i in Formula (3) is time series s_i described in Table 1. One can use the decision function to predict the next failure time. The process to predict the reliability is illustrated in Fig. 2.

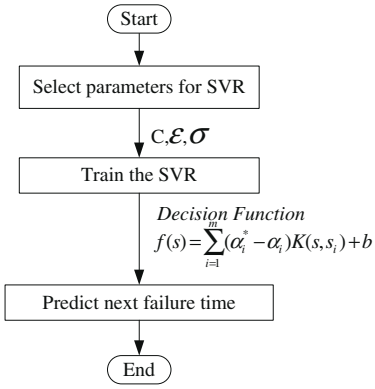


Fig. 2 Flowchart of reliability prediction

3.4 Confidence interval for SVR

Essentially, neither the LOO method nor the methodology proposed in this paper can guarantee that the optimal parameter can be obtained. The prediction error of the SVR is not controllable. To bound the prediction error, a non-homogeneous poisson process (NHPP) based method is introduced, when the considered datasets are failure time data as you can see in Table 1.

The NHPP is a popular approach to depicting the failure process of repairable systems (Barlow and Proschan 1965; Yanez et al. 2002). One of the NHPP models is the Power Law Process model, whose intensity rate is described as (Rausand and Høyland 2004; Klefsjo and Kumar 1992):

$$\lambda(t) = \frac{\beta}{\theta}\left(\frac{t}{\theta}\right)^{\beta-1} \tag{6}$$

Let t_i denote the previous failure time, and t_{i+1} denote the next failure time. The expected number of failures occurring between t_i and t_{i+1} is then:

$$\Lambda(t_i, t_{i+1}) = \left(\frac{t_{i+1}}{\theta}\right)^{\beta} - \left(\frac{t_i}{\theta}\right)^{\beta} \tag{7}$$

We use Formula (7) to predict the next failure time. The time to the next failure is when the expected number of failures between $[t_i, t_{i+1}]$ is one. The desired next failure time t_{i+1} is therefore:

$$t_{i+1} = \theta\left[1 + \left(\frac{t_i}{\theta}\right)^{\beta}\right]^{1/\beta} \tag{8}$$

The θ and β can be estimated from a maximum likelihood estimator: (Yanez et al. 2002):

$$\widehat{\theta} = \frac{t_n}{n^{1/\beta}} \tag{9}$$

$$\widehat{\beta} = \frac{n}{\sum_{i=1}^{n} \ln\left(\frac{t_n}{t_i}\right)} \tag{10}$$

From t_i to $t \geq t_i$, the probability of failure occurrence is:

$$P[t_{i+1} \geq t_i] = 1 - e^{-\int_{t_i}^{t_{i+1}} \lambda(t)dt} \tag{11}$$

Appendix A presents the proof of the Formula (11). Given the probability P, inversing Formula (11) one can estimate the next failure time. The upper bound of the next failure time at the significance level α is then:

$$t_U = \theta\left[\left(\frac{t_i}{\theta}\right)^{\beta} - \ln\frac{\alpha}{2}\right]^{\frac{1}{\beta}} \tag{12}$$

The corresponding lower bound is:

$$t_L = \theta\left[\left(\frac{t_i}{\theta}\right)^{\beta} - \ln\left(1 - \frac{\alpha}{2}\right)\right]^{\frac{1}{\beta}} \tag{13}$$

With a probability of $1 - \alpha$, the next failure time is bounded between $[t_L, t_U]$. When using SVR to predict the next failure time, the predicted value should be located in $[t_L, t_U]$.

4 Numerical case study

A Load-Haul-Dump (LHD) machine is for loading and transportation of ore/minerals in underground mines. This machine is important in underground mines and contributes a high maintenance cost. Figure 3 shows the role of LHD in mining process.

As described by Gustafson et al. (2008), in the underground mining process, the ore is initially excavated and loaded onto LHD. Then the ore is transported to the vertical shafts that are placed along the ore body and then dumped into the shaft by LHD.

The LHD is crucial but also experiences frequent failures. Figure 4 shows the sub systems that compose a typical LHD. The size of each block denotes the number of failures. It shows the hydraulic system contributes the largest number of failures.

In this numerical example, the whole LHD is considered, and developed methodology to predict the next Failure Time (FT) for the LHD is used. Forty five failure data were collected from all subsystems involved in the LHD. As you can see in Fig. 4, all these sub-systems are serial configured so each occurring failure stops the machine that is why all breakdowns contribute with the same severity to the whole reliability of the equipment. These data are tabulated in Table 2 (Kumar 1990).

These data have been divided into two groups. First, one set of 25 samples is used to train the SVM and the remaining 20 to test the model. The procedure for testing the proposed method was as follows:

1. First 25 datasets were taken out to obtain the optimal parameters using method described in Sect. 3.2,
2. These 25 datasets were also used to train the SVR.
3. Thereafter the trained SVR was used to predict the 26th failure.
4. We repeat this procedure again using the first 27th, 28th, etc. to train SVR and predict the next failure.

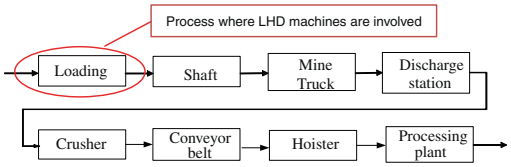


Fig. 3 Flowchart from loading to processing plant, Adapted from (Gustafson et al. 2008)

Courtesy from Sandvik

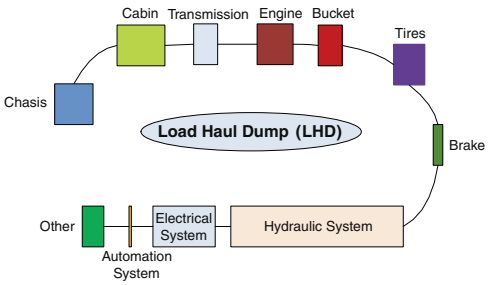


Fig. 4 Subsystems of a LHD

Table 2 Failure time (FT, in h) of a LHD

No.	FT	No.	FT	No.	FT	No.	FT	No.	FT	No.	FT	No.	FT
1	3	8	46	15	99.2	22	133.7	29	168.2	36	183.7	43	211
2	7.8	9	46.6	16	100.5	23	137.7	30	172.2	37	184.7	44	212
3	8.1	10	67.3	17	114.2	24	141.5	31	173.1	38	191.7	45	212.7
4	22.7	11	70.6	18	120.2	25	148.1	32	176.1	39	195.7		
5	32.7	12	77.6	19	123.2	26	151.3	33	177	40	196.7		
6	40.7	13	83.5	20	127.8	27	164.4	34	180.9	41	207.9		
7	45.2	14	95.1	21	129.7	28	166.8	35	182.2	42	208.9		

Time series with several lag sizes (1, 2, 3, and 4) have been used to predict failure. The results using the different time series are shown in Table 3 in terms of the total error and the Normalized Mean Square Error (NMSE).

As shown in Table 3, the best performance is that of the time series with Lag = 3. As shown in Fig. 5, the time series with Lag = 3 is closer to the real data than that with Lag = 2.

In Sect. 3.4, NHPP-based method to bound the prediction error was proposed. Using the method developed in that section, an upper bound and a lower bound at a 95% confidence interval were derived for this case study. The results are shown in Fig. 6. The figure shows that the predicted values of Lag = 3 are all within the bounds of a 95% confidence interval.

Figure 6 also shows comparison between the predicted values from the NHPP-based method and those from the SVR. It is shown that the prediction performance of the SVR with the Lag = 3 time series is better than performed by NHPP-based method, so the prediction results from the SVR are closer to the real data than those of the NHPP-based method. The SVR combining with time series achieves a good prediction performance.

Table 3 Performance of different time series

	Lag = 1	Lag = 2	Lag = 3	Lag = 4
Total error (h)	63.50	61.38	47.19	72.20
NMSE	0.0229	0.0215	0.0191	0.0289

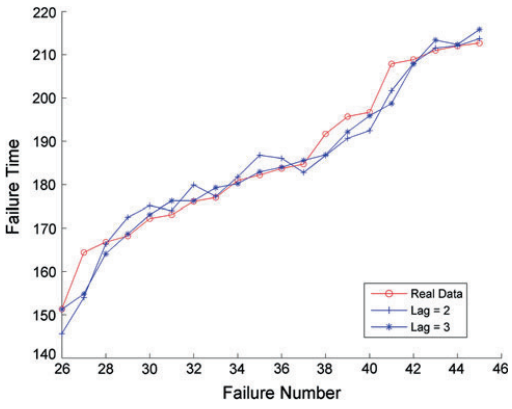


Fig. 5 Performance comparison with different time series

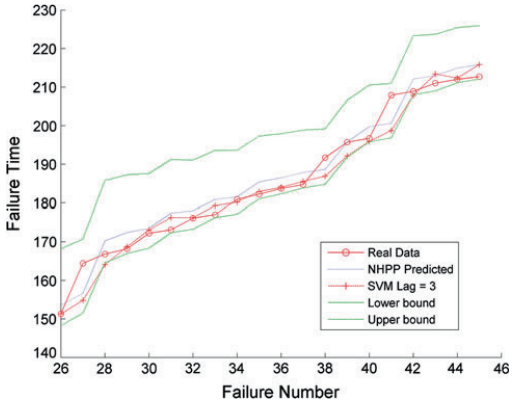


Fig. 6 Confidence interval of 95%

5 Conclusions

In this paper a method to predict the next failure time using SVR and time series has been proposed. A parameter selection method is used to approximate the LOO method. Based on the results of the case study, it is shown that the predicted values from the SVR are within the NHPP-based bounds and close to the real data.

Moreover, no guarantee of the best parameters of SVR has been obtained however a NHPP-based method is proposed to bound the predicted value within a confidence interval. This interval prevent an unreasonable prediction occurred. Weakness of this method is that this confidence interval is still too large so future work on investigating a narrower confidence interval is recommended.

Appendix

Assuming that one failure occurs at time t_i, the probability of at least one failure occurring during time t to time t_{i+1} is:

$$P[N(t) \geq 1] = 1 - P[N(t) = 0] \quad \text{(A.1)}$$

where $N(t)$ denotes the number of failures between t and t_i. As the number of failures in the interval $[t, t_{i+1}]$ is Poisson-distributed, by using the Poisson theorem:

$$P[N(t) = 0] = e^{-\int_{t_i}^{t_{i+1}} \lambda(t)dt} \quad \text{(A.2)}$$

Therefore, the probability of failure during $[t, t_{i+1}]$ is:

$$P[t \geq t_i] = 1 - e^{-\int_{t_i}^{t_{i+1}} \lambda(t)dt} \quad \text{(A.3)}$$

References

Barlow RE, Proschan F (1965) Mathematical theory of reliability. Siam series in applied mathematics. Wiley, New York

Burges CJC (1998) A tutorial on support vector machines for pattern recognition. Data Min Knowl Disc 2(2):121–167

Cawley GC, Talbot NLC (2003) Efficient leave-one-out cross-validation of kernel fisher discriminant classifiers. Pattern Recogn 36(11):2585–2592

Chen KY (2007) Forecasting systems reliability based on support vector regression with genetic algorithms. Reliab Eng Syst Safe 92(4):423–432. doi:10.1016/j.ress.2005.12.014

Foucher B, Boullie J, Meslet B, Das D (2002) A review of reliability prediction methods for electronic devices. Microelectron Reliab 42(8):1155–1162. doi:PiiS0026-2714(02)00087-2

Fuqing Y, Kumar U, CMR S, Misra KB (2009) Complex system reliability evaluation using support vector machine. Paper presented at the SMRLO10, Israel, February 8–11

Gonzalez S, Vega J, Murari A, Contributors J-E (2010) Analysis of video-movies using support vector regression. Fusion Sci Technol 58(3):763–770

Gustafson A, Parida A, Nissen A (2008) Optimizing productivity through performance measures for underground mining industry. In: Paper presented at the Proceedings of 5th MASSMIN Conference 2008, Luleå,Sweden, 08–11 June

Hochbaum DS (1997) Approximation algorithms for np-hard problems. PWS Pub. Co, Boston

Klefsjo B, Kumar U (1992) Goodness-of-fit tests for the power-law process based on the ttt-plot. IEEE T Reliab 41(4):593–598

Kumar U (1990) Development of maintenance policies for mines: a statistical approach. Luleå University of Technology, Luleå

Paul PP, Monwar MM, Gavrilova ML, Wang PSP (2010) Rotation invariant multiview face detection using skin color regressive model and support vector regression. Int J Pattern Recogn 24(8):1261–1280. doi:10.1142/S0218001410008391

Radhika Y, Shashi M (2009) Atmospheric temperature prediction using support vector machines. Int J Comput Theory Eng 1(1):1793–8201

Rausand M, Høyland A (2004) System reliability theory: models, statistical methods, and applications. Wiley series in probability and statistics. Applied probability and statistics, 2nd edn. Wiley-Interscience, Hoboken, NJ

Schölkopf B, Smola AJ (2002) Learning with kernels: support vector machines, regularization, optimization and beyond. Adaptive computation and machine learning. MIT Press, Cambridge

Vapnik VN (1995) The nature of statistical learning theory. Springer, New York

Vapnik VN (1998) Statistical learning theory. Adaptive and learning systems for signal processing, communications, and control. Wiley, New York

Xing F, Guo P (2005) Support vector regression for software reliability growth modeling and prediction. Advances in Neural Networks. Isnn 2005, Pt 1, Proceedings 3496:925–930

Yanez M, Joglar F, Modarres M (2002) Generalized renewal process for analysis of repairable systems with limited failure experience. Reliab Eng Syst Safe 77(2):167–180. doi:PiiS0951-8320(02)00044-3

Yang CC, Shieh MD (2010) A support vector regression based prediction model of affective responses for product form design. Comput Ind Eng 59(4):682–689. doi:10.1016/j.cie.2010.07.019

PAPER II: **Y. Fuqing**, U. Kumar and D. Galar, "Fault Diagnosis of Railway Assets using Support Vector Machine and Ant Colony Optimization Method," *International journal of COMADEM*, 2012 (Accepted for publication).

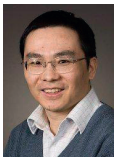
Fault Diagnosis of Railway Assets using Support Vector Machine and Ant Colony Optimization Method

Yuan Fuqing[1], Uday Kumar[1], Diego Galar[1]

[1]Division of Operation and Maintenance Engineering,

Luleå University of Technology, SE-971 87 Luleå, Sweden

Tel.: +46 920 49 1682, Email: yuan.fuqing@ltu.se



Yuan Fuqing obtained his M.Tech. in System Engineering at Beijing University of Aeronautics and Astronautics, China, in the year 2007. He joined the Division of Operation and Maintenance Engineering, Luleå University of Technology, Sweden, in September 2007 to study for the degree of PhD. His area of research deals with reliability data analysis and statistical learning theory.

Dr Uday Kumar obtained his B.Tech. in India during the year 1979. After working for 6 years in Indian mining companies, he joined the postgraduate programme of Luleå University of Technology, Luleå, Sweden, and obtained the degree of PhD in the field of Reliability and Maintenance during 1990. Afterwards, he worked as a senior lecturer and an associate professor at Luleå University of Technology from 1990-1996. In 1997, he was appointed Professor of Mechanical Engineering (Maintenance) at University of Stavanger, Stavanger, Norway. Presently, he is Professor of Operation and Maintenance Engineering at Luleå University of Technology, Luleå, Sweden. His research interests are equipment maintenance, equipment selection, reliability and maintainability analysis, system analysis, etc. He has published more than 170 papers in international journals and conference proceedings.

Dr Diego Galar is a telecommunications engineer and possesses a PhD in Manufacturing from the University of Saragossa. He has been a professor at several universities, including the University of Saragossa and the European University of Madrid. He has also been the Technical Director of ANANTASA, and the Director of Academic Innovation and subsequently Pro-Vice-Chancellor at Universidad San Jorge. He has been a researcher at the Department of Design and Manufacturing Engineering at the University of Saragossa and a researcher at I3A, the Institute of Engineering Research in Aragon. At present he is a senior researcher at Luleå University of Technology (LTU).

Abstract

Support Vector Machine (SVM) is an excellent technique for pattern recognition. This paper uses a multi-class SVM as a classifier to solve a multi-class classification problem for fault diagnosis. As the pre-defined parameters in the SVM influence the performance of the classification, this paper uses the heuristic Ant Colony Optimization (ACO) algorithm to find the optimal parameters. This multi-class SVM and ACO are applied to the fault diagnosis of an electric motor used in a railway system. A case study illustrates how efficient the ACO is in finding the optimal parameters. By using the optimal parameters from the ACO, the accuracy of the performed diagnosis on the electric motor is found to be highest.

Keywords

Support Vector Machine (SVM); Ant Colony Optimization (ACO); Fault Diagnosis, Electric Motor.

1 INTRODUCTION

Fault diagnosis is a procedure for mapping the information obtained in the measurement space to the machine faults space. This mapping process is called pattern recognition [1]. In condition monitoring, pattern recognition tries to discriminate failure patterns from normal patterns, avoiding unexpected breakdowns or false alarms, or to identify one failure pattern from another failure pattern based on the obtained signal, commonly known as diagnosis.

Fault diagnosis has been widely used to diagnose railway systems. Eker and Camci proposed a method to diagnose the "drive-rod out of adjustment" failure using Support Vector Machine [2]. Chamroukhi et al. used a pattern recognition method to detect the failure of an electric motor mounted in the switch of a railway track [3]. Feldmann et al. proposed a method to diagnose the deterioration of track by using the Karhunen–Loeve transformation [4]. Skarlatos et al. used fuzzy-logic to diagnose railway wheels [5].

In recent decades, Artificial Intelligence (AI) techniques such as Artificial Neural Network (ANN) [6] and Support Vector Machine (SVM) [7] have been widely used to perform fault diagnosis [8]. The ANN and SVM are similar techniques, but the SVM has been claimed to be more accurate than the ANN [9-11]. A detailed review of fault diagnosis using the SVM was carried out by Yang and Widodo [10]. To solve real problems, the SVM is usually combined with other techniques, such as Bayesian Theory, Wavelet Analysis, Component Analysis, Fuzzy Logic, and Rough Set Theory [12].

This paper uses the multi-class SVM combined with Ant Colony Optimization (ACO) to perform fault diagnosis. In the remaining sections, Section 2 presents the principle of the multi-class SVM, discusses the functionality of the kernel function, and presents the procedure for using the SVM for failure diagnosis. Section 3 presents a heuristic algorithm called Ant Colony Optimization (ACO) to optimize the parameters of the SVM. Section 4 illustrates a case study of fault diagnosis for railway systems to demonstrate the developed methodology.

2 MULTI-CLASS SUPPORT VECTOR CLASSIFIER

2.1 Multi-Class Classifier

The SVM was initially developed to classify two classes of objects. Only one decision function needed to be found for such binary classification [13]. However, there are a great number of applications where the number of classes is more than two. For example, one may need to discriminate between the various states of a machine, e.g. different failure stages or functional states, or one may need to discriminate different failures from each other based on the obtained signal. In terms of pattern recognition, this is a multi-class classification problem. To accommodate this multi-class problem, several SVMs can be merged together. The one-against-all multi-class SVM is one of the tricks used to merge several binary SVMs. For the sake of simplicity, this paper calls the multi-class SVM the SVM hereafter.

Suppose there are k classes of objects to be separated. Moreover, assume that there are l training data sets $(x_1, y_1),...,(x_l, y_l)$, which are used as input for the SVM. Then $y_i \in \{1,2,3,..k\}$ is the output of the SVM and it is the category (class) to which a data set belongs. The one-against-all method transformed the multi-class problem into n sub-binary classification problems. The i^{th} sub-binary classification problem labels the indicator of the data sets belonging to the i^{th} class with "1" and labels all the remaining data sets with "-1". The mathematical formula for this i^{th} binary classification is [14]:

$$\min \ \frac{1}{2}(\omega^i)^T \omega^i + C\sum_{j=1}^{l}\xi_j^i(\omega^i)^T$$
$$(\omega^i)^T \phi(x_j) + b^i \geq 1 - \xi_j^i, \ if \ y_j = i$$
$$(\omega^i)^T \phi(x_j) + b^i \leq -1 + \xi_j^i, \ if \ y_j \neq i$$
$$\xi_j^i \geq 0, \ j = 1,....,l \quad (1)$$

Each sub-classification has one decision function. For all the k classes, one can obtain k decision functions:

$$(\omega^1)^T \phi(x) + b^1$$
$$\cdots$$
$$(\omega^k)^T \phi(x) + b^k \quad (2)$$

The predicted class of x is regarded as the class with the largest decision function value. In terms of mathematics, it is:

$$i = \arg\max((\omega^i)^T \phi(x) + b^i) \quad (3)$$

2.2 Role of the Kernel Function

In the case of the SVM, the decision function formulated as Eq (3) is usually described in the form of kernel functions, i.e.:

$$f(x) = b + \sum_{i=1}^{l}\alpha_i k(x, x_i) \quad (4)$$

where $k(x, x_i)$ is a kernel function, α_i is the coefficient corresponding to the kernel function, and b is a constant. The kernel function plays a key role in the SVM, in defining and measuring the similarity of two data sets. It can perform data transformation from a lower dimension to a higher dimension explicitly.

The performance of the SVM is intrinsically and significantly affected by its kernel function. The selection of the kernel function and the selection of the internal parameter of this function control the SVM's classification capacity and classification accuracy. The most common kernel functions in the literature are the Gaussian function and the polynomial function.

The form of the Gaussian function is:

$$K(x, x') = \exp(-\frac{\|x - x'\|}{2\sigma^2}) \quad (5)$$

whereas the form of the polynomial function is:

$$K(x, x') = (< x, x' > +1)^d \quad (6)$$

2.3 Fault Diagnosis using the SVM

The SVM can be a flexible classifier. Since the parameter C in Formula (1), the parameter σ in Eq (5), and d in Eq (6) influence the performance of the SVM significantly, one has to determine their optimal values before using the SVM. This paper uses ACO to find these optimums, and Section 3 addresses this problem in detail. The Gaussian function is used as the kernel function herein. The parameters of interest are described as (C, σ) for the sake of simplicity. The proposed model for fault diagnosis combining ACO and the SVM is shown in Figure 1. There are three steps to take in performing the classification.

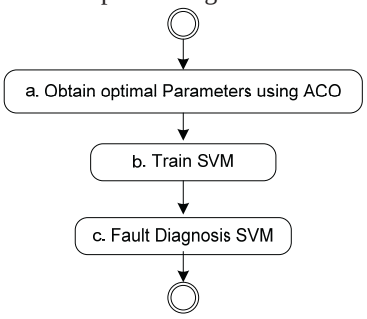


Figure 1: Procedure for fault Diagnosis

a. Parameter Optimization using ACO: A better set of parameters (C, σ) can improve the performance of the SVM greatly. The optimal (C, σ) is found by ACO.

b. Train the SVM. The training process is the process of finding the optimal solution for Eq (3). This step is time-consuming. For small-scale data sets, the active-set method can be used. As a large-scale method, the Sequential Minimal Optimization (SMO) algorithm can be used [13].

c. Fault Diagnosis using the SVM. Given a new signal obtained from the monitored machine, the identification of an existing failure among all the failures considered is performed by using Eq (3).

3 ANT COLONY OPTIMIZATION

As mentioned before, the SVM is sensitive to its parameter values. The appropriate selection of these values is a major issue for the performance of the SVM as a classifier.

As a heuristic optimization technique, ACO mimics the way in which natural ants find the optimal route for food. Ants release pheromones to mark the travelled path for the ants following them. Ants frequently move between their nest and food source and they prefer the path with a larger amount of pheromones. Therefore, after a time period and gradually, the shorter path will gain more pheromones. The paths will finally vary in their amount of pheromones. The route having the largest amount of pheromones will be the shortest (optimal) route. This procedure has inspired the new heuristic optimization approach entitled ACO. ACO was initially proposed to solve combinatorial problems [15]. Most of the application areas of ACO involve finding the optimal route, i.e. solving discrete problems such as the Travelling Salesperson Problem (TSP) [16]. Currently, ACO has been extended to solve continuous problems [15, 17-18]. A detailed survey of ACO has been performed by Dorigo and Blum [19].

3.1 ACO for Continuous Problems

Each feasible solution in ACO is designed to be associated with a probability which is used as an amount of pheromones. This probability implies the possibility of the solution being selected as optimal. For the combinatorial problem, this probability is discrete. For the continuous problem, it is continuous, along with its continuous solution space. This paper uses a methodology developed by Socha and Dorigo [15] to solve the continuous problem, and the methodology is called ACO herein for the sake of simplicity.

In an *n*-dimensional problem, ACO uses for each dimension a weighted superposition of Gaussian functions, which acts as ACO's probability (pheromones). The SVM in this paper considers a two-dimensional problem, as only two parameters (C, σ) are of interest. This weighted superposition biases the search path towards the optimal solution. The sum of the Gaussian functions for the *i*th dimension is defined as follows:

$$G^i = \sum_{l=1}^{k} \omega_l \frac{1}{\sigma_l^i \sqrt{2\pi}} e^{-\frac{(x-u_l^i)^2}{2\sigma_l^{i2}}} \quad (6)$$

One Gaussian function possesses only one minimum or maximum (see G1, G2, etc. in Figure 2), and therefore it can only suit the monotonic problem. A superposition of Gaussian functions can possess more than one local minimum and maximum, as shown in Figure 2. Hence they can deal with a problem with several minimums or maximums.

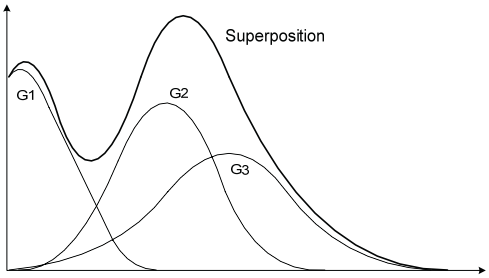


Figure 2: Superposition of Gaussian Functions

Let the n-dimensional problem be denoted by $f(s) = f(s_1, s_2, s_3, ..., s_n)$. The objective is to find the s which minimize or maximize $f(s)$. A *k*-row archive table is defined to store the recent solutions, with each row corresponding to a Gaussian function. As shown in Table 1, ω_l denotes the weight of the Gaussian function, $f(s_l)$ denotes the *l*th objective function value and s_l^i denotes the *i*th dimension of the *l*th solution.

Table 1: Archive of Partial Solutions

No.	ω_i	f(s)	Solution				
			D 1	…	D i	…	D n
1	ω_1	$f(s_1)$	s_1^1	…	s_1^i	…	s_1^n
2	ω_2	$f(s_2)$	s_2^1	…	s_2^i	…	s_2^n
…	…	…	…	…	…	…	…
l	ω_l	$f(s_l)$	s_l^1	…	s_l^i	…	s_l^n
…	…	…	…	…	…	…	…
k	ω_k	$f(s_k)$	s_k^1	…	s_k^i	…	s_k^n

The size of the table is fixed to k (which means that a superposition of k Gaussian functions is used as a pheromone). Notably, as Socha and Dorigo suggest, the size of the table should be larger than the number of dimensions. For the problem of minimizing the objective function, these objective values *f(s)* are placed in descending order. For the problem of maximizing the objective function, they are placed in ascending order.

The weight ω_l is updated using the following formula:

$$\omega_l = \frac{1}{qk\sqrt{2\pi}} e^{-\frac{(l-1)^2}{2q^2k^2}} \quad (7)$$

where q is a predefined parameter which is used to control the diversification of ACO.

The objective is to control the search direction in order to obtain optimal solutions in the solution space. As was mentioned before, a superposition of k Gaussian functions (described in Eq (6)) is used to guide the search. The mean u_l^i of the Gaussian functions is defined as follows:

$$\mu_1^i = s_1^i;\ \mu_2^i = s_2^i; ...; \mu_k^i = s_k^i \quad (8)$$

Further on, ACO defines the average distance from the mean s_l^i to other $s_j^i (j \neq l)$ in the same dimension as the standard deviation σ of the Gaussian function in Eq (6). The σ in Eq (6) is estimated using the following formula:

$$\sigma_l^i = \xi \sum_{j=1}^{k} \frac{\left| s_j^i - s_l^i \right|}{k-1} \quad (9)$$

where ξ is a predefined parameter which is adopted to adjust the convergence speed of the algorithm.

After the Gaussian functions have been defined, ACO then uses these functions to search for new solutions. The probability with which the lth Gaussian function can be selected is defined as in Formula (10):

$$p_l = \frac{\omega_l}{\sum_{j=1}^{k} \omega_j} \quad (10)$$

Suppose that the lth Gaussian function is selected, whose mean μ and standard deviation σ are defined by Eq (8) and (9). Then use this Gaussian function to sample a new value for this dimension. If the new solution is better than the best one so far in the archive table, add the new solution to the archive table.

3.2 Algorithm for Optimizing Parameters in the SVM

The objective of ACO in this paper is to minimize the classification error of the SVM in the diagnosis process: $Error_{SVM}$. The $Error_{SVM}$ can be described as a function of (C, σ), i.e. $Error_{SVM} = f(C, \sigma)$. Eq (11) is an approximation of this function:

$$Error_{SVM} = \frac{\sum \sigma_i (P_k \neq k)}{l} \quad (11)$$

where l is the total number of data sets, P_k is the data set from the k-class, and σ_i is a function of P_k (when $P_k \neq k$, then $\sigma_i = 1$ and when $P_k = k$, then $\sigma_i = 0$). The parameters (C, σ) with the minimal $Error_{SVM}$ will be considered as the optimum.

Using ACO, the detailed algorithm described below is used to find the optimal C, σ:

Step 1: Initialization. Initialize the number of ants, *nAnt*, the number of iterations for each ant, *nIteration*, and the size of the archive table, *nSize*. Moreover, initialize the coefficient parameters: q, described in Eq (7), and ξ, described in Eq (9).

Step 2: Initialize the archive table. Generate feasible solutions randomly according to the uniform distribution. Calculate the objective function values $Error_{SVM}$ and place the $Error_{SVM}$ in descending order. Calculate the

weight ω_l and the probability p_l using Eq (7) and Eq (10), respectively.

Step 3: Start the iteration. Generate a new solution. For each dimension of the solution, select a Gaussian function out of the k functions according to the probability p_l. Calculate the standard deviation, using Eq (9), for the selected Gaussian function and take the current values of s_l^i as the mean. Generate a random real number according to the Gaussian function. This real number is the new value of this dimension. Repeat this for all the dimensions and then a new solution comes out. Repeat step 3 *nAnt* times. Therefore, the number of new solutions coming out will be equal to *nAnt*.

Step 4: Update the archive table. Calculate the $Error_{SVM}$ for each new solution generated in Step 3 and select the solution with the minimal $Error_{SVM}$ as the best. Insert the best solution into the ordered archive table.

Step 5: Stop criteria. If $Error_{SVM}$ is less than the predefined value, stop the iteration, or if the maximum number of iterations has been reached, stop the iteration (which means that the optimal solution has not been found); otherwise go to Step 3.

Step 6: Take out the optimum solution. The last solution in the archive table is the optimum solution if the stop criteria have been satisfied.

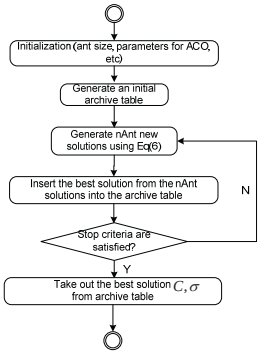


Figure 3: Flowchart of ACO

4 NUMERICAL CASE STUDY

4.1 A Numerical Example for Railway Turnouts

Railway turnouts are composed of elements known as crossings and switches. They enable vehicles to be diverted from one track to another and consist of a pair of switches and a crossing, connected by closure rails [20], as shown in Figure 4.

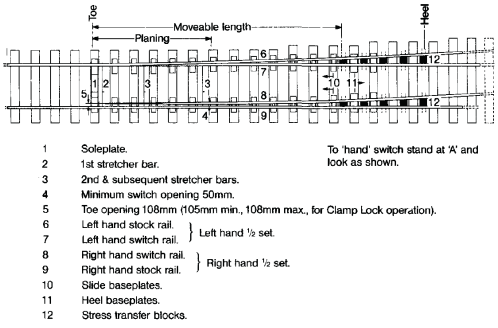


Figure 4: Components of Turnouts [20]

Turnouts are important devices along a railway track. For example, Sweden has a railway infrastructure consisting in total of 17,000 km of track and about 12,000 switches and crossings. Turnout-related failures cause 14% of all train delay time. That means an approximate delay time of 15 minutes per turnout and year, assuming that there is one turnout per 2 km of main track and that the rail traffic consists of 50 trains per day. Furthermore, the maintenance cost for turnouts represents at least 13% of the total maintenance cost. Consequently, the turnout plays an important role in the railway industry both from a functional and a financial point of view [21].

The motors and mechanical drive systems in the turnout must perform safely and reliably over a wide range of temperatures and other environmental conditions [22]. Figure 5 shows a kind of switch which is driven by an electric motor. The failure of the electric motor leads to the failure of the switch and the track, thereby compromising the safety of the entire system. One root cause of failure is a “lack of lubrication”. To exemplify the use of the SVM

for fault diagnosis, we now attempt to identify this failure mode, so that the failure can be detected incipiently.

Figure 5: Electric Motor in Switch [3]

According to [23], power consumption and dissipation are effective parameters for the condition monitoring of railway points. When there is a lack of lubrication in a switch operation, the action taking place in the "translation" stage consumes a much higher amount of power, as illustrated in Figure 6 [3]. Based on these findings, the power consumption signal can be assumed to be a condition indicator of the switch and can be used to diagnose the "lack of lubrication" phenomenon in its early stages.

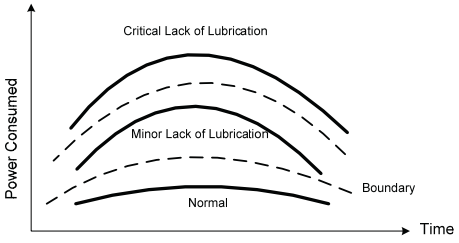


Figure 6: Patterns in the Power Consumption Signal

In simple terms, the different statuses of the electric motor in the switch of a turnout can be classified into three patterns:

- Normal.
- Showing a minor lack of lubrication. Maintenance is suggested.
- Showing a critical lack of lubrication. Maintenance must be carried out.

For this fault diagnosis problem, our target is to find the optimal boundaries separating the three patterns. Similar to Chamroukhi's approach [3], several features are extracted from the power consumption signal: the maximum, minimum, mean, deviation, and the coefficients of the polynomial fit. The number of coefficients of the polynomial fit depends on the order of the fitting. For an n-order polynomial fit, the approximate function is as follows:

$$f(x) = a_n x^n + a_{n-1} x^{n-1} + ... + a_1 x^1 + a_0 \quad (12)$$

The higher the order n is, the more flexible is the polynomial fit. The coefficients a_i are used as features. These features compose the input of the multi-class SVM.

4.2 Results Analysis

In this numerical case study, simulated data are used to demonstrate the proposed methodology. For each pattern mentioned above, 120 data sets are simulated, as shown in Figure 7. These 120 sample data sets for each pattern are divided into three different subsets: the first 60 data sets are used to train the SVM. The next 30 data sets are used to find the optimal parameters in the SVM. The remaining 30 data sets are used to validate the developed SVM and to obtain the accuracy of the SVM.

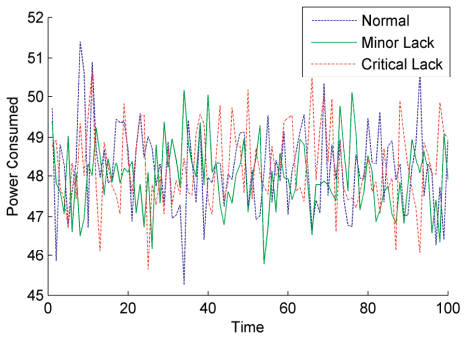


Figure 7: Simulated Data

The optimal parameters in the SVM are optimized using the methodology proposed in Section 3. Figure 8 and Figure 9 plot the figures for the error with the penalty parameter and the parameter in the kernel function, respectively. They show that there are a great number of local minimums in the function $Error = f(C, \sigma)$. The ant colony

algorithm is useful for solving the problem with a great number of local minimums.

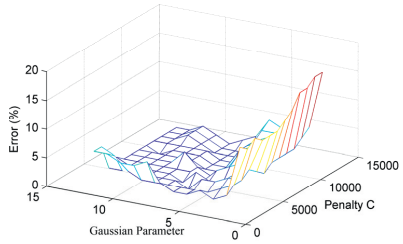


Figure 8: The Function $Error = f(C,\sigma)$

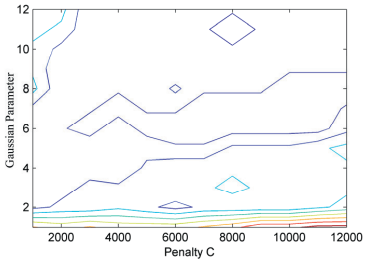


Figure 9: Contour of $Error = f(C,\sigma)$

The optimal parameters obtained by ACO are used by the SVM and several orders of polynomial fit are used for the power consumption signal. The results are tabulated in Table 2.

Table 2: Error with Various Polynomial Orders

n	C	σ	Training Error (%)	Validating Error (%)	Test Error (%)
1	1,035	3.79	3.89	7.78	5.56
2	7,902	6.18	1.67	5.56	**1.11**
3	14,626	2.69	1.67	6.67	5.56
4	3,969	3.66	2.22	8.89	7.78
5	3,910	5.44	3.33	8.89	5.56

Table 2 shows that the lowest test error occurs for n=2, that the corresponding penalty is $C = 7902$ and that the kernel function parameter is $\sigma = 6.18$. Therefore, the 2nd order polynomial fit are selected as kernel function to diagnose whether or not the DC motor lacks lubrication. The results using the simulated data show that the accuracy of diagnosis can reach 100%-1.11%= 98.89%.

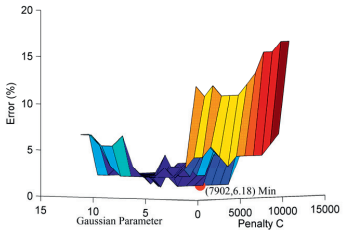


Figure 10: Minimum Point of $Error = f(C,\sigma)$

To verify if the minimum point of error is achieved with $C = 7902$ and $\sigma = 6.18$, Figure 10 plots $Error = f(C,\sigma)$ by including the point $C = 7902$ and $\sigma = 6.18$. It is shown that this point is the point with the lowest error.

5 CONCLUSIONS

This paper uses the multi-class SVM to diagnose incipient failures. ACO is used to find the optimal parameters for the multi-class SVM. The numerical case study on fault diagnosis of the electric motor of a railway switch shows that ACO is efficient in finding these optimal parameters. The diagnosis accuracy is highest using the parameters obtained from ACO. Consequently, using the SVM and ACO for fault diagnosis is feasible and practical.

Our future work will focus on the collection of condition monitoring data from switches. The SVM will be applied to the fault diagnosis to improve the diagnosis accuracy.

REFERENCES

1. Jardine, A.K.S., Lin, D.M., and Banjevic, D., "A review on machinery diagnostics and prognostics implementing condition-based maintenance" Mechanical Systems and Signal Processing, vol.20,no.7, pp 1483-1510, 2006
2. Eker, O.F. and Camci, F., "Failure Diagnostics on Railway Turnout Systems using Support Vector Machine" E-Maintenance 2010, 2010
3. Chamroukhi, F., et al., "Switch mechanism diagnosis using a pattern recognition approach" Railway

Condition Monitoring, 2008 4th IET International Conference, 2008
4. Feldmann, U., Kreuzer, E., and Pinto, F., "Dynamic diagnosis of railway tracks by means of the Karhunen-Loeve transformation" Nonlinear Dynamics, vol. 22, no. 2, pp 193-203, 2000
5. Skarlatos, D., Karakasis, K., and Trochidis, A., "Railway wheel fault diagnosis using a fuzzy-logic method" Applied Acoustics, vol. 65, no. 10, pp 951-966, 2004
6. Hassoun, M.H., Fundamentals of artificial neural networks, MIT Press, 1995
7. Vapnik, V.N., The nature of statistical learning theory, Springer, 1995
8. Roberts, C., Chen, J., and Weston P., "Fault detection and diagnosis for railway track circuits using neuro-fuzzy systems"Control Engineering Practice, vol.16, no. 5, pp. 585-596. 2008
9. Chen, K.Y., "Forecasting systems reliability based on support vector regression with genetic algorithms" Reliability Engineering & System Safety, vol. 92, no. 4, pp.423-432. 2007
10. Yang, B.S. and Widodo, A., "Support vector machine in machine condition monitoring and fault diagnosis. Mechanical Systems and Signal Processing", vol. 21, no. 6, pp. 2560-2574, 2007
11. Fuqing, Y., Kumar, U., and Galar, D., "Performance of Failure Detection using Support Vector Machine and Artificial Neural Networks: A Comparative Study" CM 2011 and MFPT 2011. 2011
12. Ramesh, R., et al., "Thermal error measurement and modelling in machine tools. Part II. Hybrid Bayesian Network - support vector machine model" International Journal of Machine Tools & Manufacture, vol. 43, no. 4. pp. 405-419, 2003
13. Schölkopf, B. and Smola, A.J., Learning with kernels : support vector machines, regularization, optimization, and beyond, MIT Press, 2002.
14. Hsu, C.W. and Lin, C.J., "A comparison of methods for multiclass support vector machines" Ieee Transactions on Neural Networks, vol. 12, no.2, pp. 415-425, 2002
15. Socha, K. and Dorigo, M., "Ant colony optimization for continuous domains" European Journal of Operational Research, vol. 185, pp. 1155–1173, 2008
16. Guntsch, M. and Middendorf, M, "Applications of Evolutionary Computing: A population based approach for ACO" Proceedings of the Applications of Evolutionary Computing on EvoWorkshops, Springer-Verlag, 2002.
17. G.BiLCHEV and PARMEE, I.C., "The Ant Colony Metaphor for Searching Continuous Design Spaces" Lecture Notes in Computer Science, vol. 993, pp. 25-39, 1995
18. Mathur, M., et al., "Ant colony approach to continuous function optimization" Industrial & Engineering Chemistry Research, vol. 39, no. 10, pp. 3814-3822, 2000
19. Dorigo, M. and Blum, C., "Ant colony optimization theory: A survey" Theoretical Computer Science, vol. 344, no. 2-3, pp. 243-278, 2005
20. Bonnett, C.F., Practical railway engineering, 2nd Edition, Imperial College Press, 2005
21. Nissen, A., "Development of Life Cycle Cost Model and Analyses for Railway Switches and Crossings" Doctral thesis in Luleå University of Technology, 2009
22. Marquez, F.P.G. and Schmid, F., "A digital filter-based approach to the remote condition monitoring of railway turnouts" Reliability Engineering & System Safety, vol, 92, no. 6, pp. 830-840, 2007
23. Pabst, M. "Remote monitoring of points based on effective power dissipation" Condition Monitoring for Rail Transport Systems, 1998

PAPER III: **Y. Fuqing**, U. Kumar and D. Galar, "An Adaptive Multiple-kernel Method based Support Vector Machine for Classification," *International Journal of Condition Monitoring*, 2011.(Submitted)

An Adaptive Multiple-kernel Method based Support Vector Machine for Classification

Yuan Fuqing[a*], Uday Kumar[b], Diego Galar[c]

[a,b,c]*Division of Operation and Maintenance Engineering, Luleå University of Technology, SE-971 87 Luleå, Sweden*
Email: Yuan.Fuqing@ltu.se

Abstract:

Classification is an important technique in pattern recognition, data mining and knowledge discovery. It is also a common tool that can be used in decision support systems. Intensive researches have been done on the classification and plenty of techniques have been developed. Support Vector Machine (SVM) as one of the classifier techniques possesses an excellent classification capacity and has been widely used in literature. Support Vector Machine (SVM) is a machine learning algorithm based on the Statistical Learning Theory (SLT), which can get good classification effects with a few learning samples with low computational cost. The effectiveness of SVM depends on the selection of the kernel function so in order to maximize performance; this paper proposes an Adaptive Multiple Kernel SVM (AMK-SVM) for classification. Using AMK, large number and heterogeneous features such as continuous, categorical, logic, etc. can be merged together. Instead of predefining the parameters in kernel functions as other Multiple Kernel SVM, this method can adapt its parameters to data automatically by using Kernel Alignment. A numerical example with benchmarking data is presented to demonstrate the feasibility of the developed approach and test its performance. A three-layer neural network and two single kernel SVMs are also applied to the same data sets to compare their performance with AMK-SVM.

Keywords: Support Vector Machine (SVM); Classification; Adaptive MK; Kernel Alignment.

1. Introduction

Classification target consists of the clustering of given data to a specific class. Areas where classifiers are used cover all aspects of business, technology or sciences. For instance, an email server should decide if the incoming mails are spam. In this case performance and processing speed are crucial because thousands of emails can be received simultaneously by the system, and they could degrade or disable the system. In face recognition, the human face should be discriminated from other objects, and medical image processing uses these classifiers to detect known patterns in X-ray pictures, etc... In fact medicine is a popular area for classifiers utilization; Automatic diagnosis cab be

[*] Corresponding author. Tel.:+46 920 49 1682; fax: +46 920 49 1935.
E-mail: yuan.fuqing@ltu.se
Postal Address: Div of Operation and Maintenance Engineering, Luleå University of Technology, SE-971 87 Luleå, Sweden.

done in function of patient's symptoms in order to decide the proper treatment. Business areas for banking risk analysis are a recent interesting topic because bankers can evaluate their customers to decide if they give loans or no. Finally, many industrial applications require classifiers to diagnose possible failures in machinery, extracting this information from known states. In these applications, decision making based on the probability of occurrence of a scenario is crucial to take proper further actions. Obviously, these areas are different but the above mentioned problems can be formulated in a similar way, using the same tools, as a mathematical classification problem.

In terms of mathematics, classification tries to create a model that shows the dependence between observed data and its class. Due to its wide occurrence in the filed of pattern recognition, data mining, knowledge discovery, cancer diagnosis, fraud detection and, etc., intensive researches have been done in this field and plenty of techniques have been developed. Most popular methods are Logistic regression [1], Naive-Bayes Classifier [2], Recursive Partitioning Methods [3], Linear Discriminant classifier [4], the k-nearest neighbor [5], Artificial Neural Network (ANN) [6],Support Vector Machine (SVM)[7-8], etc. For all these techniques, many variations have been developed for specific applications trying to improve the performance in concrete areas.

Logistic regression and Naive-Bayes Classifier assume the observed data follows a specified distribution. Using these data, optimal parameter of these distributions can be estimated. User can decide the class of new date based on the distribution where this date most probably belongs to. The other techniques above mentioned are non-parameters techniques. Such techniques don't need to consider the distribution of the data sets. Linear Discriminant classifier is the simplest model with the lowest computational cost among them. This model is easily to be understood and interpreted. However, its performance is poor due to the strict linear assumption on the decision function. K-nearest neighbor is to perform classification based on closest training data sets. Such as method is also simple yet its computational cost is high.

ANN, SVMs are later developed techniques. As intelligence techniques, both techniques are flexible and nonlinear in data classification. ANN and SVM are similar because they try to minimize the error in the classification process, but they differ at their basis principle. The differences of performance of both methods have also been widely discussed in literature. Scholkopf etc. proved SVM can outperform ANN by using USPS database of handwritten digits [9]. Chen also proved SVM can outperform ANN for his case[10].

As a classifier, SVM has thousands of applications, which cover image process [11], failure diagnosis [12], cancer diagnosis [13], cancer issue validation [14], text classification [15], credit risk evaluation [16], speech recognition [17], spam email detection [18], financial forecasting [19], etc. SVM is characterized by its using of kernel function. Normally, SVM uses a single kernel. Recently, in order to improve prediction accuracy, SVM has been developed from single kernel to multiple kernels. Chapelle et al. developed a multiple kernel SVM. According to the author, different attribute or element in data use different kernel parameter. The final kernel function is a combination of all basic kernels [20]. Similar to Chapelle's model, Rakotomamonjy et al. developed a multiple kernel method called SimpleMKL [21]. In SimpleMKL, each attribute can use different kernels and for the whole set of attributes multiple kernels can be used as well.

Subrahmanya and Shin developed another methodology to use multiple kernels. Their proposal weights the regularization parameter for features from different data source instead of do it with the individual kernel function [22].

The remaining part of this paper is structured as follows: Section 2 presents the concept and characteristics of some preliminary theory of kernel method. Section 3 presents the developed Adaptive Multiple Kernels SVM (AMK-SVM) and proposes a parameter optimization method for the AMK-SVM. Section 4 presents a case study and finally, Section 5 presents discussion and future work.

2. Preliminary theory

The measurements used for classification are called features. SVM uses kernel method to transform original input space into a higher dimensional feature space. In the higher dimensional space, the pattern is more identifiable. In this section, we introduce the concept of kernel method and describe some property of kernel function related to our proposed AMK-SVM.

2.1 Kernel Method

Kernel method is a class of algorithms developed for multivariate data analysis and pattern recognition. The major characteristic of kernel method is its use of kernel function, which is denoted as $k(z,x)$. It calculates the similarity between two observations (data sets) z and x . The notion "similarity" can be geometrically interpreted as the distance of between the data set z and x .

Kernel function plays a key role in SVM. It settles the performance of SVM. To show the principle of kernel function mathematically, we present the classical method to calculate similarity of two data sets.

The distance or similarity between data sets z and x can be calculated by inner product of z and x .

$$k(z,x) = \langle z, x \rangle \tag{1}$$

As in a higher dimensional space, the difference of two data sets may be more distinct. So we transform the original data sets to a higher dimensional space by using transformation function $\Phi(z)$, as shown in following Equation (2).

$$z = (z_1, z_2, ...z_n) \mapsto \Phi(z) = (\phi_1(z), ..., \phi_i(z), ...) \tag{2}$$

In the higher dimension, the patterns tend to be more identifiable. Figure 1 shows the schema of this advantage. In the Figure1 a, the circle and crossing are not linearly separable in two-dimension space. By using of the transformation Φ, the original two-dimension data are transformed into a higher three-dimension data, as shown in Figure 1 b. In the higher three-dimension space, the two classes of dots can be separated more easily.

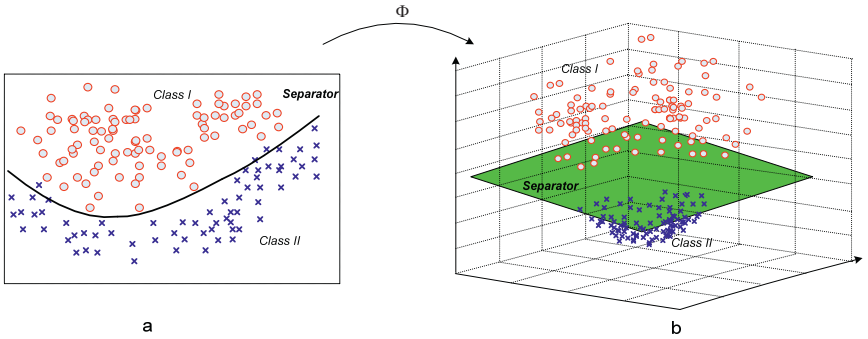


Fig. 1. Transformation of Data Set

In the transformed higher dimensional space, the similarity quantity of data sets z and x is calculated by:

$$k(z,x) = \langle \Phi(z), \Phi(x) \rangle \tag{3}$$

As the transformation $\Phi(z)$ is an expansion of function $\phi_i(z)$, the kernel function calculates similarity in Equation (3) is equivalent with:

$$k(x,z) = \sum_{i=1}^{\infty} \lambda_i \phi_i(x) \phi_i(z) \tag{4}$$

The above mentioned method is the classical method to calculate similarity. The objective is to find the similarity quantity $k(x,z)$. Instead of using the above procedure to calculate similarity of two data sets, SVM usually use a kernel function to directly calculate $k(z,x)$, where the transformation function Φ is implicitly defined in that kernel function.

The common kernel function used in literature survey is Gaussian and polynomial kernel function. The form of Gaussian function is:

$$k(x,z) = \exp\left(-\frac{\|x-z\|}{2\sigma^2}\right) \tag{5}$$

And polynomial function is:

$$k(x,z) = (< x, z > +1)^d \tag{6}$$

The advantages of kernel function are one need not define the transformation function Φ explicitly, and the computational cost is also low.

When kernel function is used to define the similarity of data sets, the corresponding decision function is thus represented by a combination of kernel functions. For example, the decision function for a binary classifier of SVM is:

$$f(x) = \mathrm{sgn}\left(\sum_{j=1}^{m} \lambda_j y_j k(x, x_j) + b\right) \tag{7}$$

where λ_j denotes coefficient of data sets x_j, b denotes a constant bias. This decision function can be nonlinear depending on the kernel function $k(x, x_j)$. Figure 2 shows the decision function when Gaussian kernel function is used. The "1 0 -1" are values of decision function. Line with "0" is the optimal separator to classify the two classes of dots. Evidently, this separator is nonlinear and it is flexible to fit data.

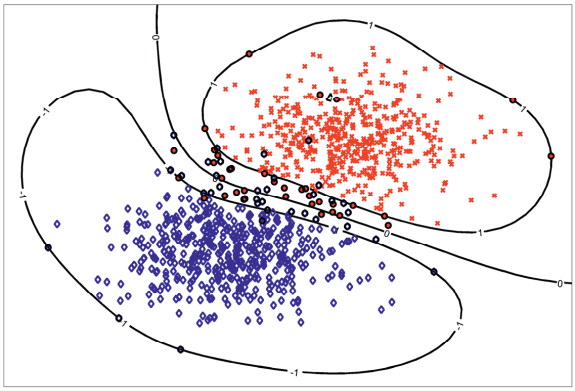


Fig. 2. Decision function using Gaussian function

2.2 Additive property of kernel function

All functions satisfy Mercy theory can be a kernel function [23]. As the linear combination of kernel function also satisfies mercy theory, so this combination is still a kernel function. We call this property additive property. This property lays the foundation of multiple kernel method. One can obtain an optimal kernel function via the optimization on the linear combination of kernel function.

The additive property is described mathematically as: Suppose k1 and k2 are kernel functions, given $\alpha_1 \geq 0$, $\alpha_2 \geq 0$, the $\alpha_1 k_1 + \alpha_2 k_2$ is also a kernel function.

The combination of kernel function sometimes can improve the performance of SVM because special characteristics of these kernels are added. For instance, Gaussian function is local function, which means it can fit a nearby datasets perfectly, whereas it neglects the influence of other far data sets. On the other hand global kernels such as polynomial kernel consider the classification problem globally while lacking of local flexibility. To obtain benefits from both local fit and global influence of kernel, Use a combination of kernel functions can be one solution.

2.3 Kernel Alignment

The performance of SVM is sensitive to the parameter value in kernel function, such as σ in Gaussian kernel function. So Parameter estimation on these parameters is important to the performance of SVM. Some existing parameter estimation methods such as crossing-validating are effective but their computational cost is expensive [20]. As for each tentative parameter, one needs to train the SVM at least once. However, the training

of SVM is computational cost, so these methods are inefficient. Kernel alignment provides a method to tune the parameter without training of SVM and obtaining important savings.

Kernel alignment tries to measure the distance (or similarity) between two kernel matrixes [24]. Kernel matrix is a matrix composed by kernel function value. Suppose the data sets are denoted by

$$S = \{(x_1, y_1), (x_2, y_2), \ldots, (x_i, y_i)...\} \quad (8)$$

where the xi as an input corresponds an output y_i.

The kernel matrix defined on the S is defined as:

$$K = \begin{bmatrix} k(x_1, x_1)\ k(x_1, x_2) \ldots\ k(x_1, x_n) \\ k(x_2, x_1)\ k(x_2, x_2) \ldots\ k(x_2, x_n) \\ \ldots \\ k(x_n, x_1)\ k(x_n, x_2) \ldots k(x_n, x_n) \end{bmatrix} \quad (9)$$

The definition of kernel alignment is:

Definition: Let K_1 and K_2 are kernel matrixes by two kernel functions on data sets S. Assume the K_1 and K_2 are same size. The alignment between K_1 and K_2 then as follows:

$$A(K_1, K_2, S) = \frac{< K_1, K_2 >_F}{\|K_1\|_F \|K_2\|_F} \quad (10)$$

The F in Formula (8) denotes Frobenius inner product. Frobenius inner product of the matrixes is defined as $< K_1, K_2 >= Trace(K_1 K_2)$ [24].

In SVM, it has been known the optimal kernel matrix is the kernel matrix constructed by YTY [26], where Y=(y_1,y_2,..y_i,..).. The y_i is the indicator of classes for i^{th} data set. For binary classification, the y_i is defined as $y_i \in \{1, -1\}$. This best kernel matrix provides a target to design kernel. One can construct a new kernel to make its corresponding kernel matrix as similar as possible to the optimal kernel matrix YTY. The corresponding kernel alignment is:

$$A(K, Y^T Y) = \frac{< K, Y^T Y >_F}{\|K\|_F \|Y^T Y\|_F} \quad (11)$$

This kernel alignment implies the separability of the kernel function. When one design kernel function or optimize a linear combination of kernel functions, the maximum kernel alignment can be of its optimization objective.

3. Adaptive Multiple Kernel Method

The use of single kernel function in SVM is rigid to fit data, especially when the input features are extracted from different data sources. Multiple kernel functions facilitate SVM to fit data more flexibly. The present paper proposes a multiple kernel method based on the kernel alignment above mentioned.

3.4 AMK Development

For features from different data source or everyone feature, one can select a special kernel to fit their characteristics. For example, for continuous feature, Gaussian kernel can be selected; for feature extracted from text expression, string kernel can be selected. On the other side, for a feature, one can use more than one kernel. For instance, as mentioned in Section 2.2, both Gaussian and Polynomial kernels can be used to a continuous feature by lineally combining them.

To illustrate the above mentioned concept, we propose a schematic example as shown in Figure 3. Suppose the data are from n data sources. In source 1, features are extracted from both time and frequency domain. Let the features from time domain and frequency domain feature accommodate in kernel function k_1 and k_2 respectively. For data source n, the features are only from the time domain. We accommodate these features in kernel function k_3 and k_4. The k_3 and k_4 can be same kind of kernel but with different parameter, or using a different kind of kernel function, e.g. Gaussian and Polynomial function. The final kernel function is then a linear combination.

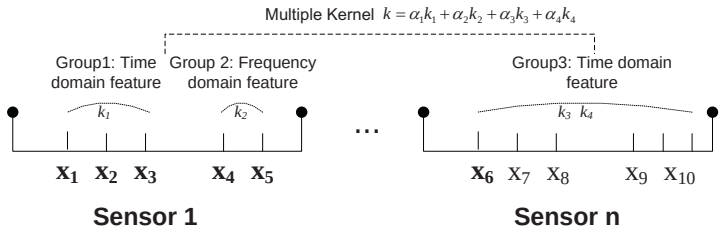


Fig. 3. Multiple Kernel Method

When the parameters of kernel function, such as σ in Gaussian function, has been predefined. One multiple kernel function is as follows [21]:

$$k_M(\alpha) = \alpha_1 k_1 + \alpha_2 k_2 + \alpha_3 k_3 + \alpha_4 k_4 + \alpha_5 k_5 + ... \qquad (12)$$

where α_i is the weight of kernel function k_i.

Our AMK considers these parameters in kernel functions as variable, which depend on the data, i.e. it is adaptive to the data. The corresponding multiple kernel function is as follows:

$$\begin{aligned} k_M(\alpha,\sigma) = {} & \alpha_1 k_1(\sigma_1) + \alpha_2 k_2(\sigma_2) + \alpha_3 k(\sigma_3) + \\ & + ...\alpha_i k_i(\sigma_i) + \alpha_{i+1} k_{i+1} + ... \end{aligned} \qquad (13)$$

In multiple kernel method, the value $k(z,x)$ for different kernel function may differ drastically. For example, using the polynomial kernel function, for d=1, when the $k(z,x)$ is 5, that for d= 10 is then 3125, i.e. the value of higher order polynomial will much higher than that of lower order. This problem will make the smaller kernel function ineffective when multiple kernel functions are used. For such problem, it is necessary to

normalize the $k(z,x)$. One effective normalization method is the method proposed by Shawe-Taylor and Cristianini [25], which is shown as:

$$k'(z,x) = \frac{k(z,x)}{\|k(z,z)\|.\|k(x,x)\|} \tag{14}$$

Notably, the normalization on kernel function will improve the computational cost considerably. Sometimes one can avoid such normalization by selecting other kernel function.

3.5 Parameter Estimation

In our AMK, both the coefficients of each kernel function and the parameters in kernel functions are variable. For simplicity, in this paper, we call coefficient as parameter as well. Such large number of parameters is a challenge for parameter optimization. The improper parameter will lead to poor performance both in terms of computational cost and accuracy. Determination of such parameters has been a hot topic. Crossing-validation method is the effective method to solve this problem [8]. However, the computational cost of this method is high, as given each heuristic parameter value, the SVM should be trained at least once. However, training the SVM is a time consuming process. If the number of training data sets is large then the crossing-validation method will be almost infeasible.

The kernel alignment, as mentioned in Section 2, provides a solution to estimate parameters where time-consuming training process can be avoided. Without training the SVM, the parameter estimation will be more computational efficient. Our proposed parameter estimation method based on kernel alignment. It sets the maximum kernel alignment between kernel matrix K_M generated by multiple kernels and the target matrix Y^TY. The optimization programming formulation of this problem is as follows:

$$\begin{aligned} &\max \frac{< K_M(\alpha,\sigma), Y^TY >_F}{\|K_M\|_F \|Y^TY\|_F} \\ &\alpha \geq 0; \\ &\sigma \geq 0 \end{aligned} \tag{15}$$

As discussed in paper [26], the above Formula (15) is equivalent to

$$\begin{aligned} &\max_{\alpha} \sum_k \alpha_k Y^T K_k(\sigma_k) Y - \sum_{kl} \alpha_k \alpha_l < K_k(\sigma_k), K_l(\sigma_l) >_F \\ &s.t. \quad \alpha_i \geq 0; \sigma_i \geq 0 \end{aligned} \tag{16}$$

where K_k denotes the kernel matrix of k^{th} kernel, α_k denotes its corresponding coefficient, σ_k denotes parameter of the k^{th} kernel.

Formula (16) can be transformed into a non constraint optimization problem. We replace the coefficient α_i by $\ln e_i$ and σ_i by $\ln \sigma_i$. The equivalent of Formula (16) without constraint is as follows:

$$\max_{e,\sigma} \sum_k \ln e_i Y^T K_k(\ln \sigma_k) Y - \sum_{kl} \ln e_k \ln e_l < K_k(\ln \sigma_k), K_l(\ln \sigma_k) >_F \qquad (17)$$

Some constraint or unconstraint optimization algorithm can be utilized to obtain the optimal parameters. However, when the parameters in kernel method are adaptive, these methods will also be computationally inefficient, as for each heuristics parameter, the kernel matrix needs to be recalculated once. The computational cost on kernel matrix is high. One computationally cheaper method is to use following two-step approximate method:

Step 1: For each kernel function k_i, use kernel alignment method as follows to find its optimal parameter $\hat{\sigma}_i$.

$$\begin{aligned} &Max \frac{< K_i(\sigma), Y^T Y >_F}{\left\| K_i(\sigma) \right\|_F \left\| Y^T Y \right\|_F} \\ &\sigma \geq 0; \end{aligned} \qquad (18)$$

Step 2: Fix the $\hat{\sigma}_i$ in kernel function. Optimize the coefficient α_i of each kernel function using following Formula.

$$\begin{aligned} &\underset{\alpha}{Max} \sum_k \alpha_k Y^T K_k(\hat{\sigma}_k) Y - \sum_{kl} \alpha_k \alpha_l < K_k(\hat{\sigma}_k), K_l(\hat{\sigma}_l) > \\ &s.t. \quad \alpha_i \geq 0 \end{aligned} \qquad (19)$$

This algorithm decomposes the parameter estimation into two sub problems. Each sub problems is easily to be solved. Notably, the parameter obtained from the above two-step method is an approximate of that obtained from Formula (16). But this approximate is reasonable as it can improve the computational efficiency.

After the optimal coefficient has been obtained, some coefficient could be insignificant. Such insignificant coefficient implies the irrelevance of the corresponding features. We can remove the features with a small contribution. In this way, we have performed a kind of dimension reduction implicitly.

3.6 AMK-SVM

We apply this proposed AMK to SVM. This is called AMK-SVM. The dual problem of a 2-norm binary SVM classifier is as follows [23]:

$$\begin{aligned} \max_{\alpha} \quad & -\frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{m} \lambda_i \lambda_j y_i y_j k_M < x_i, x_j > + \sum_{k=1}^{m} \lambda_k \\ s.t. \quad & 0 \leq \lambda_i \leq C, \quad i = 1,2,3...,m \\ & \sum_{i=0}^{m} \lambda_i y_i = 0 \end{aligned} \qquad (20)$$

where λ_i represents Lagrangian multiplier which corresponds a data sets x_i. C is penalty parameter which trade-offs classification accuracy and computation complexity.

$k_M < x_i, x_j >$ is the multiple kernel function which equivalent to $k_M(x_i, x_j) = \alpha_1 k_1(x_i, x_j) + \alpha_2 k_2(x_i, x_j) + \alpha_3 k_3(x_i, x_j) + \ldots.$.

Using optimization method such as active set methods [27] to obtain optimal solution (values of λ_i) for Formula (16), we get the following decision function to predict class for the new data set x.

$$f(x) = \text{sgn}(\sum_{j=1}^{m} \lambda_j y_j k_M(x, x_j) + b) \qquad (21)$$

3.7 Features Subset Selection by Kernel alignment

Feature selection is to remove the insignificant feature. The motivation of feature selection mostly is to reduce the computational complexity and to improve generalization performance. Using kernel method, the major motivation of feature selection is to improve generation performance.

As we mentioned before, higher KA value implies better separability of a kernel function. This separability depends on both the kernel function and the data sets for training. When we aim to perform feature selection, we fix the kernel function and compare the kernel alignment for a different subset of features. Suppose a number of n features present initially. There will be 2n combinations (subset) by these features. We calculate the kernel alignment for each of these combinations. The optimal feature combination is that combination with highest kernel alignment. The detail procedure to perform feature selection is as shown in Figure 4.

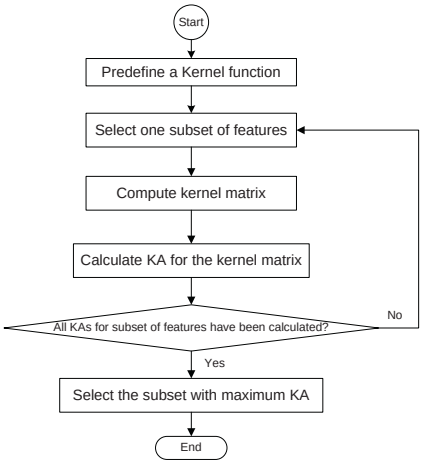


Fig. 4. Feature Selection Procedure

The major advantage of this feature selection method is its computational efficient, as KA can be obtained without the time consumption for the training process of SVM. When the number of n is large, the abovementioned method enumerating all subsets is impossible.

One can employ some heuristic method such as Genetic Algorithm, Tabu Search to obtain the optimal combinations [28].

4. Numerical Example

The data of this numerical example is the widely used benchmarking ionosphere dataset. These data is collected by a radar system [29]. The data is for a binary classification problem, which contains 33 inputs (i.e. 33 features), a single binary output with 351 data sets. Positive output 1 returns shows the evidence of structure in the ionosphere, negative output -1 returns shows no structure. Figure 5 shows the original raw data in 2-dimensional space for the 2nd and 3rd features. It shows the two patterns (1,-1) are mixed together geometrically. In order to classify the two patterns efficiently, nonlinear classifier such as ANN, SVM should be introduced. The line "0" in Figure 5 is the decision function of SVM using single Gaussian function. It shows the performance of this function is low, as lots data are misclassified.

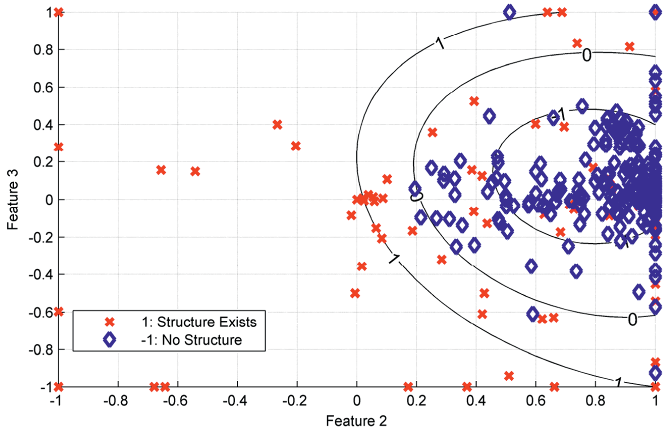


Fig.5. Pattern in 2-D space

The proposed AMK-SVM is used to classify the two patterns and predict if structure exists in the ionosphere. In order to compare performance, an artificial neural network and single kernel SVMs is also applied to the same data sets. The AMK is selected as a linear combination of one Gaussian kernel and one Polynomial kernel, where the parameters for each kernel are adaptive. The parameter of the single kernel SVMs is also adaptive. ANN selects 3-layer with 24 hidden neurons. The data has been normalized. And all these methods are implemented by Matlab. The algorithm to train the SVM is Active Set Method. The optimal coefficient in AMK-SVM is obtained by the built-in fminunc function in Matlab. This function includes several optimization methods such as Quasi-Newton and trust-region method. The ANN used here is from the pattern recognition algorithm in Neural Network Toolbox.

The total 351 data sets have been divided into two groups. The first group is used to train the SVM or ANN, the other group is used to validate the predicting results. Out of the 351 data sets, 50% and 80% data sets are selected for training and the remaining data sets for validating. These 50% or 80% data sets are selected randomly. Twenty times runs for each division have been carried out, thus 20 training sets have been selected for each division. This random selected training sets are not replicate, as using various training sets to test the proposed approach is more reasonable.

4.1 Accuracy

The predicting accuracy of using these techniques is tabulated in Table 1. Most of the 20 runs, the AMK-SVM can achieve highest accuracy and the Gaussian adaptive is second highest for both 50% and 80% data division.

Table 1. Accuracy using Various Techniques

50% Training					80% Training				
No.	Gaussian	Poly	AMK-SVM	ANN	No.	Gaussian	Poly	AMK-SVM	ANN
1	90,34	85,80	94,32	87,16	*1*	87.32	83.10	91.55	85.77
2	89,20	86,36	92,61	85,06	*2*	94.37	85.92	94.37	89.15
3	89,20	82,39	95,45	86,82	*3*	90.14	85.92	95.77	91.55
4	90,91	84,09	92,05	80,63	*4*	88.73	83.10	94.37	86.76
5	89,20	82,95	94,32	88,18	*5*	90.14	87.32	94.37	77.61
6	92,61	83,52	95,45	82,27	*6*	95.77	88.73	98.59	91.55
7	90,91	86,36	94,89	86,70	*7*	90.14	83.10	97.18	90.85
8	87,50	84,66	93,18	86,76	*8*	90.14	85.92	97.18	88.45
9	92,61	85,23	93,18	84,77	*9*	88.73	84.51	92.96	85.49
10	88,07	82,39	92,61	86,14	*10*	90.14	88.73	100	89.44
11	89,20	86,93	92,05	87,95	*11*	90.14	83.10	91.55	82.68
12	89,20	76,70	93,18	88,58	*12*	90.14	78.87	92.96	88.31
13	88,07	85,23	95,45	82,84	*13*	90.14	81.69	91.55	87.46
14	93,75	86,93	93,75	88,41	*14*	91.55	85.92	92.96	88.45
15	83,52	85,80	92,05	86,31	*15*	91.55	85.92	94.37	90.00
16	84,09	75,57	91,48	83,24	*16*	91.55	88.73	95.77	89.44
17	89,20	81,25	91,48	86,19	*17*	94.37	88.73	95.77	91.41
18	90,34	88,07	93,75	80,74	*18*	92.96	87.32	91.55	92.82
19	88,07	86,36	91,48	86,99	*19*	91.55	85.92	92.96	86.48
20	86,93	84,66	93,18	85,74	*20*	87.32	74.65	91.55	81.97

The data in Table 1 is listed in the Figure 6. For the 50% training data, all the accuracy of AMK-SVM are higher than that of the others. For the 80% training data, AMK-SVM achieves higher accuracy for most runs, where only 1 out of the 20 runs can not be better than that of other.

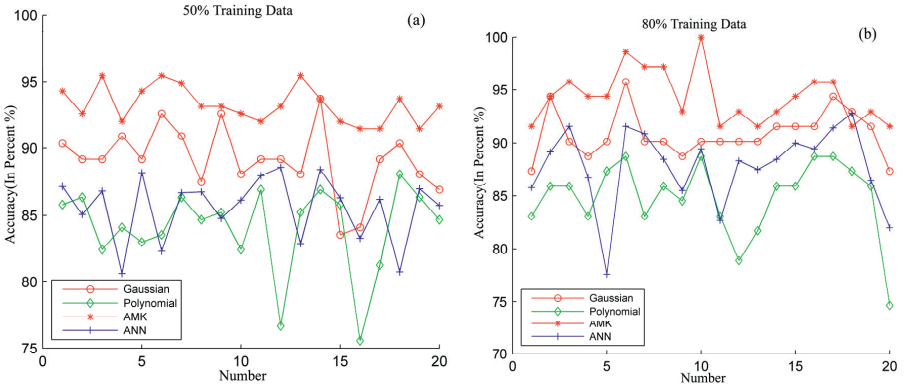


Fig. 6. Comparison of Accuracy

Table 2 below is a summary of predicting error, where mean error is the mean of error from the 20 runs. Deviation is the standard deviation of error from the 20 runs. In terms of mean error, Table 2 shows the error of AMK-SVM is lower than single kernel SVMs and ANN. This result shows interestingly the AMK-SVM can achieve better results than that from its constituent single kernel function.

Table 2. A Summary of Predicting Error (In Percent %)

	Gaussian	Polynomial	AMK-SVM	ANN
80% Mean Error	9.15	15.14	**5.63**	12.22
80% Deviation	2.21	3.56	**2.50**	3.74
50% Mean	10.85	15.94	**6.70**	14.43
50% Deviation	2.54	3.24	**1.34**	2.43

4.2 Computational Cost

As shown above, the AMK-SVM has a higher accuracy. However, compared with the single kernel SVMs and ANN, the computational cost of AMK-SVM is also high. Figure 7 plots the computational time for AMK-SVM, single kernel method and ANN. For both the 50% and 80% data sets, the computational time of AMK-SVM is highest.

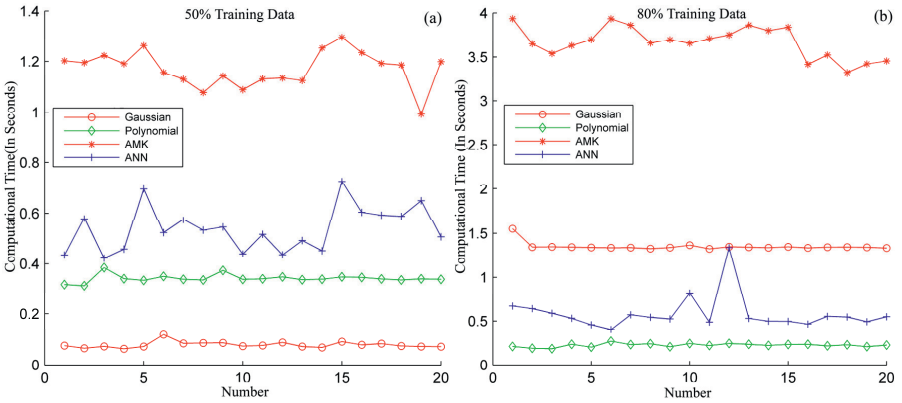


Fig.7. Comparison of Computational Cost

Table 3 shows the mean of the computational time.

Table 3. A Summary of Computational time (In Second)

Training Data	Gaussian	Polynomial	AMK-SVM	ANN
50%	0.3418	0.077	1.1702	0.5376
80%	1.3507	0.2233	3.6676	0.5857

Evidently, the mean of computational time of AMK is also higher than others. This computational time is acceptable. The neck bottle of the computational cost is the optimization on coefficient, i.e. step 2 in Section 3.5. Some efficient method should be implemented to fasten this computation. Nevertheless, the proposed approach in this paper is still efficient than the classical crossing-validating method. We don't discuss it in detail since it is not the main concern of this paper.

5. Conclusion and Discussion

Numerical examples provided in Section 4 show that the proposed methodology is feasible to perform classification. In terms of accuracy, developed AMK-SVM shows a good performance for the demonstrated case. However, in terms of computational cost, the AMK-SVM cannot outperform ANN and single kernel SVM for some applications such as cancer diagnosis, where accuracy prioritizes over computational cost, the AMK-SVM is the optimum. Nevertheless, some applications as real time control where a quick response is required, the single kernel SVM or ANN is better.

However, in some situation the multiple kernel method is inevitable. For example, some features are a mixture of continues, categorical, or logic. Some feature even is a text expression. Using a uniform kernel function to fit such a mixture feature is infeasible. Even so, the AMK-SVM can accommodate such heterogeneous features naturally. More importantly, proposed AMK-SVM does not need to predefine the parameter for each kernel function heuristically, as our AMK-SVM can automatically adapt its parameter to fit the data. This advantage can avoid the heuristic determination of the kernel parameters where an improper value will result to a poor performance on the SVM. Moreover, using

kernel alignment method, our AMK-SVM is computationally efficient compared with the classical time-consuming crossing-validating method.

Summarizing, AMK-SVM is for binary classification. This model can be extended to solve multi-class classification problems. The only thing needs to be improved is to change the calculation of kernel alignment to accommodate multi-classes classification. A limit of the developed method is: when a large number of kernels presents, the procedure to obtain the optimal parameters is time consuming. The improperly estimated coefficients will lead to poor performance as well, even it is worse than some single kernel method. The future research will focus on developing an effective optimization programming method to obtain optimal parameters.

References

[1] D.W. Hosmer, S. Lemeshow, Applied logistic regression, 2nd ed. ed.,Wiley, New York, Chichester, 2000.

[2] C.C. Hsu, Y.P. Huang, K.W. Chang, Extended Naive Bayes classifier for mixed data, Expert Syst Appl, 35(3) (2008) 1080-1083.

[3] P. Flajolet, Methods in the Analysis of Algorithms - Evaluations of a Recursive Partitioning Process, Lecture Notes in Computer Science, 158(1983) 141-158.

[4] N. Cristianini, J. Shawe-Taylor, An introduction to Support Vector Machines : and other kernel-based learning methods, Cambridge University Press, Cambridge, 2000.

[5] S. Theodoridis, K. Koutroumbas, Pattern recognition, 3rd ed., Academic Press, San Diego, CA, 2006.

[6] H.A. Rowley, S. Baluja, T. Kanade, Neural network-based face detection, Ieee T Pattern Anal, 20(1) (1998) 23-38.

[7] V.N. Vapnik, Statistical learning theory, Wiley, New York, 1998.

[8] V.N. Vapnik, The nature of statistical learning theory, Springer, New York, 1995.

[9] B. Scholkopf, K.K. Sung, C.J.C. Burges, F. Girosi, P. Niyogi, T. Poggio, V. Vapnik, Comparing support vector machines with Gaussian kernels to radial basis function classifiers, Ieee T Signal Proces, 45(11) (1997) 2758-2765.

[10] K.Y. Chen, Forecasting systems reliability based on support vector regression with genetic algorithms, Reliab Eng Syst Safe, 92(4) (2007) 423-432.

[11] F. Bovolo, L. Bruzzone, L. Carlin, A Novel Technique for Subpixel Image Classification Based on Support Vector Machine, Ieee T Image Process, 19(11) (2010) 2983-2999.

[12] A. Widodo, B.S. Yang, Support vector machine in machine condition monitoring and fault diagnosis, Mechanical Systems and Signal Processing, 21(6) (2007) 2560-2574.

[13] R. Bieda, A. Switonski, S. Kwiatek, W. Latos, G. Cieslar, A. Sieron, K. Wojciechowski, Classification by support vector machine aided the diagnosis cancer domains in multispectral endoscopic imaging, Prz Elektrotechniczn, 86(12) (2010) 13-16.

[14] T.S. Furey, N. Cristianini, N. Duffy, D.W. Bednarski, M. Schummer, D. Haussler, Support vector machine classification and validation of cancer tissue samples using microarray expression data, Bioinformatics, 16(10) (2000) 906-914.

[15] S. Tong, D. Koller, Support vector machine active learning with applications to text classification, J Mach Learn Res, 2(1) (2002) 45-66.

[16] Y.Q. Wang, S.Y. Wang, K.K. Lai, A new fuzzy support vector machine to evaluate credit risk, Ieee T Fuzzy Syst, 13(6) (2005) 820-831.

[17] S.K. Kim, J.H. Chang, Speech/Music Classification Enhancement for 3GPP2 SMV Codec Based on Support Vector Machine, Ieice T Fund Electr, E92a(2) (2009) 630-632.
[18] J. Moon, T. Shon, J. Seo, J. Kim, J. Seo, An approach for spam E-mail detection with support vector machine and n-gram indexing, Lect Notes Comput Sc, 3280(2004) 351-362.
[19] C.J. Lu, T.S. Lee, C.C. Chiu, Financial time series forecasting using independent component analysis and support vector regression, Decis Support Syst, 47(2) (2009) 115-125.
[20] O. Chapelle, V. Vapnik, O. Bousquet, S. Mukherjee, Choosing multiple parameters for support vector machines, Machine Learning, 46(1-3) (2002) 131-159.
[21] A. Rakotomamonjy, F.R. Bach, S. Canu, Y. Grandvalet, SimpleMKL, J Mach Learn Res, 9(2008) 2491-2521.
[22] N. Subrahmanya, Y.C. Shin, Sparse Multiple Kernel Learning for Signal Processing Applications, Ieee T Pattern Anal, 32(5) (2010) 788-798.
[23] B. Schölkopf, A.J. Smola, Learning with kernels : support vector machines, regularization, optimization, and beyond, MIT Press, Cambridge, Mass., 2002.
[24] G.R.G. Lanckriet, N. Cristianini, P. Bartlett, L. El Ghaoui, M.I. Jordan, Learning the kernel matrix with semidefinite programming, J Mach Learn Res, 5(2004) 27-72.
[25] J. Shawe-Taylor, N. Cristianini, Kernel methods for pattern analysis, Cambridge University Press, Cambridge, 2004.
[26] J. Kandola, J. Shawe-Taylor, N. Cristianini, optimizing kernel alignment over combinations of kernels, in, University of London, 2002.
[27] D.G. Luenberger, Y. Ye, Linear and nonlinear programming, 3rd ed. ed., Springer, New York, 2008.
[28] D.T. Pham, D. Karaboga, Intelligent optimisation techniques : genetic algorithms, tabu search, simulated annealing and neural networks, Springer, London, 2000.
[29] V. Sigillito, Ionosphere Data Set in http://archive.ics.uci.edu/ml/datasets/Ionosphere (assesed on 2011.02.10), Laurel,U.S.A., 1989.

PAPER IV: **Y. Fuqing**, U. Kumar and D. Galar, "A Comparative Study of Artificial Neural Networks and Support Vector Machine for Fault Diagnosis," presented at the CM 2011 and MFPT 2011, Cardiff, UK, 2011. Improved version has been submitted to *International Journal of Performability Engineering*.

A Comparative Study of Artificial Neural Networks and Support Vector Machine for Fault Diagnosis

YUAN FUQING, UDAY KUMAR AND DIEGO GALAR

Division of Operation and Maintenance
Luleå University of Technology,
SE-971 87Lulea, Sweden

Abstract: Fault diagnosis is a crucial step in condition based maintenance requiring. The importance of fault diagnosis necessitates an efficient and effective failure pattern identification method. Artificial Neural Networks (ANN) and Support Vector Machine (SVM) emerging as prospective pattern recognition techniques in fault diagnosis have been showing its adaptability, flexibility and efficiency. Regardless of variants of the two techniques, this paper discusses the principle of the two techniques, and discusses their theoretical similarity and difference. Eventually using the commonest ANN, SVM, a case study is presented for fault diagnosis using a wide used bearing data. Their performances are compared in terms of accuracy, computational cost and stability.

Keywords*: Failure Pattern Recognition, Artificial Neural Networks (ANN), Support Vector Machine (SVM), Fault Diagnosis.*

1. Introduction

As artificial intelligence techniques, the Support Vector Machine (SVM) and the Artificial Neural Networks (ANN) can be used for failure diagnosis. Both techniques have the ability to automatically learn and remember relationships among input data. This learning process can be supervised or unsupervised. Supervised learning requires a set of empirical data (x,y), paired up as input and output, to estimate the dependency between x, y. To achieve the learning, the ANN and the SVM adjust their internal parameters to minimize the margin between real output y and predicted $\hat{y}$. In Artificial Intelligence, this learning process is called training. When the minimal error has been determined, the relationship between x and y is established and the tool is used to make further predictions.

The ANN and the SVM are the most popular artificial intelligence techniques for fault diagnosis. Yang et al. (2002) propose a method to diagnose bearing failures using Neural Networks to classify data, whereby features are extracted from spectrum analysis [1]. Samanta et al. (2003) diagnose bearing failures using time domain features; the ANN is also used as a classifier [2]. Hajnayeb et al. (2008) use the ANN to perform feature selection for ball-bearings [3]. The SVM is also widely used for condition monitoring in state-of-the-art research [4]. Abbasion et al. (2007) use the SVM as a classifier to diagnose rolling element bearings [5]. Sugumaran et al. (2008) use the multi-class SVM to diagnose bearing faults using features extracted from the time domain [6].

The paper compares the performance of the SVM and the ANN in fault diagnosis and is an improved version of Yuan et al (2011) [7]. Section 2 explains the principles of the ANN and the SVM, while Section 3 discusses their computational complexity. Section 4 provides a numerical study, and Section 5 presents the discussion.

2. ANN and SVM in principle

The ultimate objective of the ANN and the SVM is to find a learning algorithm maximizing the generalization ability. The generalization ability can be bounded by a formula described as [8]:

$$R(\theta) = R_{emp}(\theta) + \Phi(\frac{l}{h}) \qquad (1)$$

In this formula, $R(\theta)$ denotes the real risk of the function with parameters θ and it is the quantitative measurement of the generalization ability. $R_{emp}(\theta)$ is the empirical risk. $\Phi(l/h)$ denotes the confidence of $R_{emp}(\theta)$, where l is the number of data sets and h is the VC dimension of the admissible functions. More complex functions have a higher $\Phi(l/h)$ and a lower $R_{emp}(\theta)$. Formula (1) suggests that a good learning algorithm minimizing $R(\theta)$ should balance the empirical risk and the confidential interval.

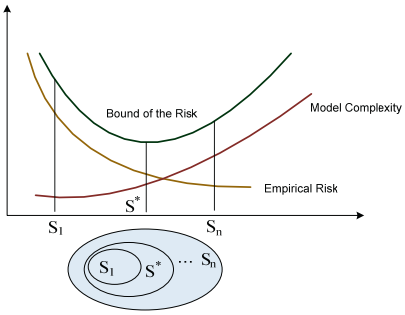


Figure 1 Risk Bound of learning algorithm

The ANN and the SVM achieve this goal in different ways: The ANN fixes $\Phi(l/h)$ but minimizes $R_{emp}(\theta)$; the SVM fixes $R_{emp}(\theta)$ but minimizes $\Phi(l/h)$. The different ways to accomplish the same goal result in myriad learning algorithms in various forms. In this sense, Formula (1) unifies ANN and SVM. The next two sections describe the principles of the ANN and the SVM.

2.1 Artificial Neural Networks

Artificial neural networks are simplified artificial models based on the biological learning process of the human brain. The ANN has been widely used in recent years for many non-linear applications, including classification, function approximation, control filter, pattern recognition etc. Numerous attempts have been made to use the ANN for machinery fault diagnosis. For example, an application of the ANN to pre-process, compress and classify vibration spectrum and time signals for bearing faults

is demonstrated by Hajnayeb et al. (2008) [3]. Li et al. (2000) use ANN to diagnose motor rolling bearing faults [9]. Raheem et al. (2010) propose an approach to diagnose bearing failures using ANN and Laplace wavelet [10]. Liu et al. (1993) use the neural network to detect bearing defects [11].

There are numerous neural networks available with different implementation methods, and thus having different complexity and performance. The simplest ANN is McCulloch-Pitts's neuron [12] shown in Figure 2. $x_1, x_2, ...x_n$ are the inputs of the neuron and the weights w_i are adjustable parameters. The optimum of w_i is obtained by minimizing the discrepancy between the desired output and the real output. $\sum$ denotes the addition of these weighted inputs, $z = \sum w_i x_i$. $f = (\Sigma)$ is the activation function that defines the output of the neuron.

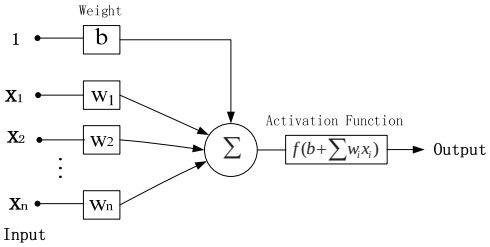


Figure 2. McCulloch-Pitts's Neuron

The ANN uses numerous neurons to construct a multi-input and multi-output neural network. In condition monitoring, the multi-layer neural networks are the most commonly used type of ANN. The topology of this kind of neural network is shown in Figure 3.

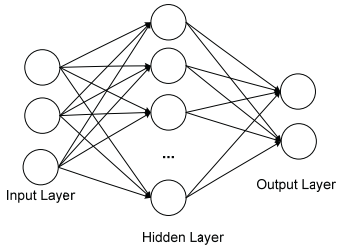


Figure 3. Three-layer ANN

This type of ANN is called "feed forward ANN;" the data flow is strictly forward and not cyclic, so there is not feedback connection, and the loop is not closed. The feed forward ANN is the most common type of ANN used in engineering.

2.2 Support Vector Machine

The SVM was originally developed for classification. Figure 4 provides a simple illustration of the classifier. Suppose there are two classes of dots (black and white). The solid line shown in Figure 4 separates them. This line is named "separator" or

"Hyperplane". The SVM tries to find the optimal line which maximizes the distance between these two classes of dots.

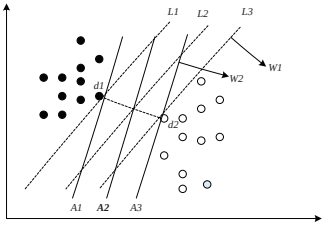


Figure 4. Binary Classifier

To find the optimal separator, the SVM employs the architecture shown in Figure 5. Given a set of training data, the SVM employs the Lagrangian method to obtain the optimal decision function based on the kernel function [8, 13, 14]. The kernel function is used to measure the distance between two vectors; it can transform the input vector to a higher dimension space where the pattern is more identifiable. The SVM has a nonlinear decision function when its kernel function is nonlinear.

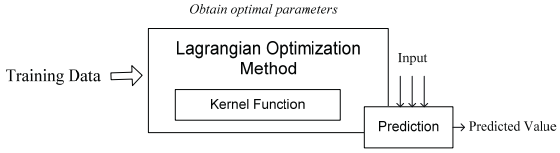


Figure 5. Architecture of SVM

The decision function of the SVM is

$$f(x) = \sum_{i=1}^{N} \alpha_i k(x, x_i) + b \qquad (2)$$

where x_i is the training data sets, $k(.,.)$ is the kernel function and b is the bias. The mathematical formula to obtain the optimal solutions is omitted here, which is readily available elsewhere [8, 14].

2.3 Regularized Neural Networks

As mentioned in Section 2, the neural networks fix the complexity of the model and minimize the empirical risk. This empirical risk is normally measured by the mean square error. The "overfitting" phenomenon is more likely occur when the model is complex and the available data set is small. To prevent this phenomenon the ANN usually uses the early stopping technique or the regularization technique. This paper focuses on the latter. The optimization objective in the regularization framework for the ANN is in the form of [15, 16]:

$$F = \beta E_d + \alpha E_w \qquad (3)$$

where E_d is the expected mean square error, E_w is the regularizer controlling the weights in the neural networks, and α and β are coefficients. This framework can prevent the "overfitting" problem with the proper α and β. And inversely, the improper α and β will degrade the performance of the ANN.

The Bayesian regularization determines α and β adaptively using the Bayesian inference. The regularized ANN has been described as equivalent to the SVM [15]. For the neural network with one hidden layer, the decision function is $f_{FNN}(x) = b_0 + \sum_{k=1}^{N} \lambda_k \varphi(\omega_k, x)$. N is the number hidden neurons. This form is similar to the SVM decision function. The difference is that in the ANN (ω_k, x) is the activation function, whereas in the SVM it is the kernel function.

2.4 Similarities and differences of ANN and SVM

Both the ANN and the SVM are adaptable. Adaptability means that the dependency between data is automatically established and can be updated when new data are available. Adaptability to data gives the ANN and the SVM a learning ability. Differences between them are summarized in Table 1.

Table 1. Differences of ANN and SVM in principle

	ANN	**SVM**
Transparency	Black box	Transparency
Optimization objective	Minimize empirical error	Structural risk minimization
Optimum Solution	Local minimum	Global minimum
Stability	Low	High

Stability refers to performance stability in this paper. Both the ANN and the SVM show random performance when the selection of internal parameters is random or the selection of training sets is random. In principle, the stability of SVM should be better than the ANN because the SVM solution is global while the ANN solution is local. Different initial values in the ANN in the configuration and the input data sets lead to different local optimums and result in different performance. The numerical example studied in this paper shows that the randomness of the ANN is evident.

Both the SVM and the ANN are able to solve nonlinear problems. However, the ANN attempts to minimize empirical errors, so it has an "overfitting" problem, which means it tends to fit specified training data. The improved ANN, such as regularized ANN, can avoid the "overfitting" problem to some extent by bounding the weight of parameters [17], for example, the Bayesian Regularized (BR) ANN. The case study in this paper shows the effectiveness of this regularization process of the BR ANN. To avoid the "overfitting" problem, the SVM employs structural risk minimization [8], which is also a regularization framework.

3. Computational complexity of ANN and SVM

Computational complexity of the learning algorithm depends on the particular training algorithm. The training algorithms of the ANN and the SVM differ dramatically due to their different principles. For the ANN, this paper uses the following three training algorithms to get benchmarks points; n denotes the number of training data sets [18]:

- Scaled Conjugate Gradient (SCG) Algorithm, complexity $o(n^2)$
- Levenberg-Marquardt (LM), complexity $o(n^3)$

- BFGS quasi-Newton method with complexity $o(n^2)$

In the case of the SVM, for small and median scale data that do not require to be chunked during training, the training of the SVM is a simple quadratic optimization problem whose computational depends on the penalty C. Let n_s denote the number of support vectors and d_l denote the dimension of the training data. If the C is large, the coefficients of a few data sets in SVM can reach C,

- the complexity is $o(n_s^3 + n_s^2 n + n_s d_l n)$ if $n_s / n << 1$, or
- the complexity is $o(n_s^3 + n_s n + n_s d_l n)$ if $n_s / n \approx 1$

When C is small, the computational complex can be reduced:

- if $n_s / n << 1$, the complexity is $o(n_s^2 + n_s d_l n)$ and
- if $n_s / n \approx 1$, the complexity is $o(d_l n^2)$ [19]

Although the computational cost depends on the particular algorithm, the computational complexity of the SVM and the ANN are theoretically close. However, in practice, for small and medium scale data sets, the computational cost of SVM can be much less than the ANN [2, 20].

4. Numerical Comparison

The case study uses the public data on bearings obtained from a test rig in Case Western Reverse Lab [21]. In this rig two bearings are installed on this rig, one at the end of the driver, and the other at the end of the fan. Vibration data are collected by accelerometers attached to the housing with magnetic bases. The data used here are from the bearing located at the driver end. The data are collected when the shaft speed is 1797 rpm. Five signals are obtained for different defect sizes in the inner race of the bearing. The signal is sampled at 12k per second. The defect sizes and the corresponding total data points are shown in Table 2.

Table 2. Defect sizes of inner race

	Normal	Inner defect 1	Inner defect 2	Inner defect 3	Inner defect 4
Defect Size	0	0.007 inches	0.014 inches	0.021 inches	0.028 inches
Data Points	243938	121265	121846	122136	120801

Figure 6 shows that the patterns of normal and defect signals differ in shape and amplitude. In this study, both the ANN and the SVM are used to discriminate them automatically.

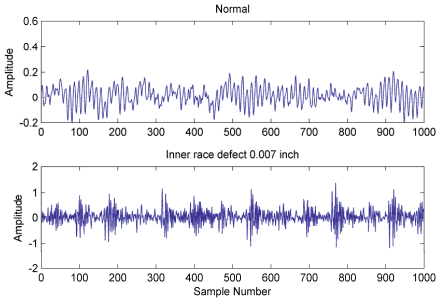


Figure 6. Normal and Defect Signals

Feature for vibration signal can be extracted from time domain, frequency domain or other domain [22]. The following features are extracted from the time domain: Kurtosis, Impulse Factor, and Normal Negative Likelihood value (NNL) [23]. The SVM and the ANN diagnose fault with five inner race defects. Their performances are compared in terms of accuracy, computational complexity and stability.

Each category in Table 2 corresponds to a signal and each signal is divided into 20 slots. Hence, 100 data sets are obtained. Out of these 100 data sets, 60% are randomly selected as training data sets; the remaining 40% are selected as validation data sets.

4.1 Fault Diagnosis using ANN

A two-layer ANN with multiple hidden neurons and one output neuron is used. The selection of this ANN is because its complexity similar to that of the SVM [24]. The structure of the ANN shown in Figure 7.

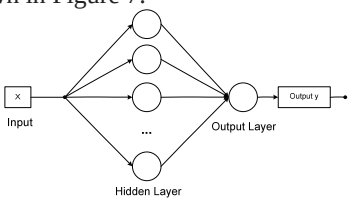


Figure 7. Structure of ANN used

Various numbers of neurons from 1 to 30 are used for this two-layer ANN. Three training algorithms are used to test its computational cost: Scaled Conjugate Gradient (SCG), Levenberg-Marquardt (LM), and BFGS quasi-Newton method (BFGS). The computational complexity is discussed in Section 3.

For the ANN, the training target is selected as the Mean Square Error (MSE). This predefined MSE significantly influences the performance of the ANN. To demonstrate this, we arbitrarily select several MSEs to show the MSE influence. Figure 8 shows the accuracy (left side) and elapsed time (right side) for various numbers of hidden neurons. The accuracy is highest for MSE=0.001, the accuracy of MSE=0.01 follows. Despite the high level of accuracy for MSE=0.001, however, the computational cost of this MSE is also high, as shown in the right figure in Figure 8. Trading off accuracy and computational cost, MSE=0.01 is selected as the training target.

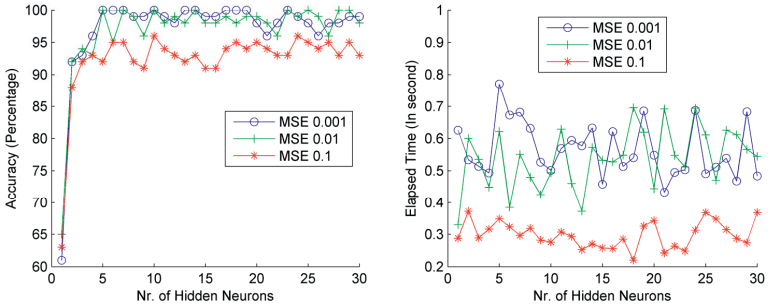


Figure 8. Accuracy and elapsed time for various MSEs

As mentioned, the training algorithms can significantly affect the computational cost. To compare the computational cost and their accuracy, several popular algorithms is used to train the ANN. As shown in Figure 9, when the number of neurons exceeds 10, the performances of all the training algorithms stabilize. The regularized ANN (BR in Figure 9) has most outstanding accuracy. Essentially, the BR shown in Figure9 is not a training algorithm but a Bayesian Reutilization ANN using the LM algorithm for training. The BR is presented here to facilitate the comparison. In terms of computational cost, the LM has a much lower computational cost and the BR follows. Trading off computational cost and accuracy, the BR is evidently the best ANN for this case.

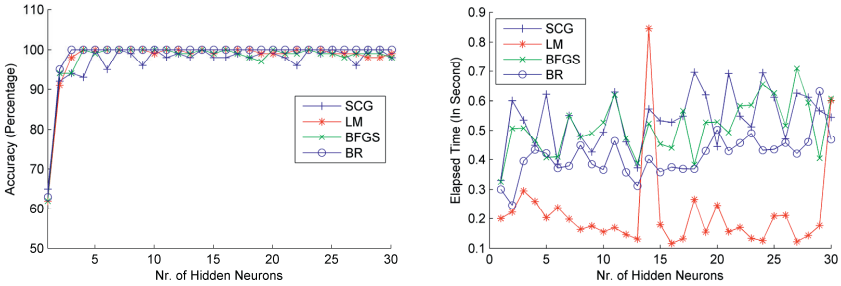


Figure 9. Accuracy and time Elapsed for various algorithms

The ANNs are using fixed 10 neurons as hidden layers. Each algorithm has been run 10 times. As discussed, the performance of the ANN could be random, as the selection of initial weights is random. The mean, maximum, and minimum for each of the 10 runs are shown in Table 3. Evidently, the performance of ANN is random, as the accuracy varies even when the same data sets used. Out of these training algorithms, the BR has the best performance.

Table 3. Accuracy with 10 hidden neurons

Algorithm	Mean Accuracy	Max Accuracy	Min Accuracy	Mean CPU Time	Mean Time Elapsed
SCG	92.4	100	88	0.69	0.60
LM	79.6	100	20	0.33	0.19
BFGS	76.9	100	20	0.46	0.41
BR	99.2	100	97	0.72	0.37

As mentioned in Section 2, the ANN is a nonlinear learning algorithm. Therefore, the decision function of the BR ANN is nonlinear. This nonlinear is evident as demonstrated in Figure 10.

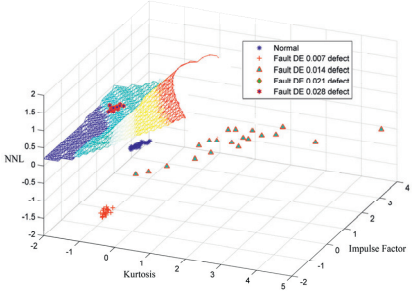


Figure 10. Decision function of BR ANN

4.2 Fault Diagnosis using SVM

The SVM uses both the Gaussian and the polynomial kernel functions to diagnose faults. For the Gaussian kernel function, the parameters range from 0.01 to 10 with a step size of 0.05. Results shows an SVM using all these parameters can detect faults with 100% accuracy, but the computational cost varies, as shown in the left figure in Figure 11, where the elapsed time varies with various parameters. However, for the most part, it is around 0.1 seconds. When SVMs using the polynomial kernel function with parameters from 1 to 20 are applied to this problem, results show that faults can be also detected with 100% accuracy as well. Their computational cost is shown on the right hand side of Figure 11. The elapsed time is below 0.1 seconds, which is lower than the Gaussian kernel function.

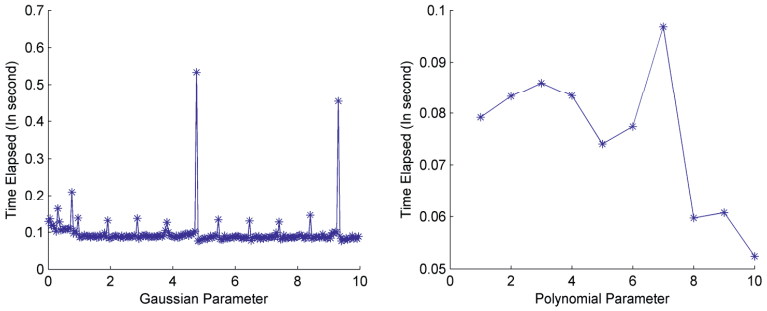


Figure 11. Computational cost of SVM

The SVM can be a linear or nonlinear classifier. Decision functions using a polynomial function with a parameter of 1 is linear, as shown in Figure 12. Here, the decision is a plane which means it is linear.

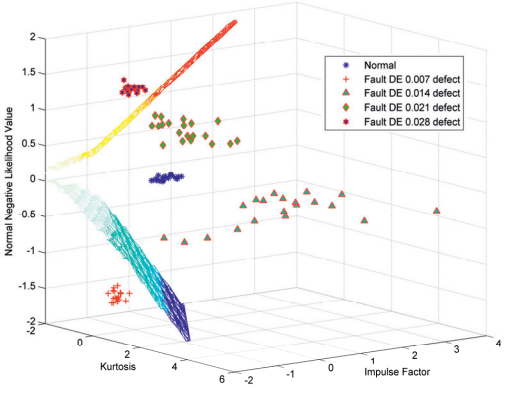


Figure 12. Linear SVM

4.3 ANN vs SVM

The BR ANN has been found to be the best among the ANNs discussed in the previous section. Therefore, only the BR ANN is compared to the SVM. In this case,

the predication accuracy of the SVM is 100% for all the Gaussian and Polynomial parameters used, while the mean accuracy of the BR ANN is 99.2. Thus, the SVM has better accuracy. In terms of computational cost, the SVM is also more efficient than the BR ANN as shown in Table 4.

Table 4. ANN vs SVM

Techniques	Mean Accuracy	Max Accuracy	Min Accuracy	Time Elapsed	CPU Time
SVM Poly	100	100	100	0.075	0.127
SVM Gaussian	100	100	100	0.096	0.179
BR	99.2	100	97	0.72	0.37

The performance of the ANN and the SVM depends on the internal parameters, such as the MSE in the ANN, and the parameter in the kernel function in SVM. Both ANN and SVM show random performance, when the selection of internal configuration parameters is random and the selection of training sets is random. The cases study in this paper clearly shows the randomness of ANN performance. As shown in Table 4, for the same data sets, the accuracy of the BR ANN is ranged between 97% and 100%. The SVM is more stable; when the parameters are changed, the accuracy remains 100%.

5. Conclusions

This paper compares the performance of the ANN and the SVM with respect to fault diagnosis, using measurements of accuracy, computational cost, and stability. The case study shows the SVM can outperform the two-layer ANN in all three measurements. The regularized ANN has the best performance among the ANNs but cannot outperform SVM, even though in theory, the two-layer regularized ANN has the same model complexity as the SVM.

This selection of techniques for failure diagnosis is a trade off between time efficiency and accuracy. For example, for the ANN, the smaller MSE leads to higher accuracy, but consumes more training time, as shown in Figure 8. For fault diagnosis, the simpler model seems a better option. One might, for example, select fewer neurons in the ANN or use a lower order polynomial kernel function in the SVM, as the less complex model could demonstrate similar or better performance while incurring less computational cost.

Acknowledgements

Authors are thankful to Swedish Railway Administration Trafikverket and Swedish Strategic Mining Research Center CAMM for financially supporting the study.

References

[1] D. M. Yang, *et al.*, "Third-order spectral techniques for the diagnosis of motor bearing condition using artificial neural networks," *Mechanical Systems and Signal Processing*, vol. 16, pp. 391-411, Mar-May 2002.

[2] B. Samanta, "Gear fault detection using artificial neural networks and support vector machines with genetic algorithms," *Mechanical Systems and Signal Processing*, vol. 18, pp. 625-644, May 2004.
[3] A. Hajnayeb, *et al.*, "Design and implementation of an automatic condition-monitoring expert system for ball-bearing fault detection," *Industrial Lubrication and Tribology*, vol. 60, pp. 93-100, 2008.
[4] A. Widodo and B. S. Yang, "Support vector machine in machine condition monitoring and fault diagnosis," *Mechanical Systems and Signal Processing*, vol. 21, pp. 2560-2574, Aug 2007.
[5] S. Abbasion, *et al.*, "Rolling element bearings multi-fault classification based on the wavelet denoising and support vector machine," *Mechanical Systems and Signal Processing*, vol. 21, pp. 2933-2945, Oct 2007.
[6] V. Sugumaran, *et al.*, "Fault diagnostics of roller bearing using kernel based neighborhood score multi-class support vector machine," *Expert Systems with Applications*, vol. 34, pp. 3090-3098, May 4 2008.
[7] Y. Fuqing, *et al.*, "Performance of Failure Detection using Support Vector Machine and Artificial Neural Networks: A Comparative Study," presented at the CM 2011 and MFPT 2011, Cardiff, UK, 2011.
[8] V. N. Vapnik, *The nature of statistical learning theory*. New York: Springer, 1995.
[9] B. Li, *et al.*, "Neural-network-based motor rolling bearing fault diagnosis," *Ieee Transactions on Industrial Electronics*, vol. 47, pp. 1060-1069, Oct 2000.
[10] K. F.Al-Raheem and W. Abdul-Karem, "rolling bearing fault diagnostics using artificial neural networks based on laplace wavelet analysis," *International Journal of engineering, science and technology*, vol. 2, pp. 278-290, 2010.
[11] T. I. Liu and N. R. Iyer, "Diagnosis of roller bearing defects using neural networks," *International Journal of Advanced Manufacturing Technology*, vol. 8, pp. 210-215, 1993.
[12] M. H. Hassoun, *Fundamentals of artificial neural networks*. Cambridge, Mass.: MIT Press, 1995.
[13] Y. Fuqing, *et al.*, "Complex System Reliability Evaluation using Support Vector Machine for Incomplete Data-set," *International journal of Performability Engineering*, vol. 7, pp. 32-42, 2011.
[14] B. Schölkopf and A. J. Smola, *Learning with kernels : support vector machines, regularization, optimization, and beyond*. Cambridge, Mass.: MIT Press, 2002.
[15] P. Andras, "The equivalence of support vector machine and regularization neural networks," *Neural Processing Letters*, vol. 15, pp. 97-104, 2002.
[16] D. J. C. Mackay, "A Practical Bayesian Framework for Backpropagation Networks," *Neural Computation*, vol. 4, pp. 448-472, May 1992.
[17] T. Hastie, *et al.*, *The elements of statistical learning : data mining, inference, and prediction*, 2nd ed. ed. New York: Springer, 2009.
[18] M. F. Moller, "A Scaled Conjugate-Gradient Algorithm for Fast Supervised Learning," *Neural Networks*, vol. 6, pp. 525-533, 1993.
[19] C. J. C. Burges, "A tutorial on Support Vector Machines for pattern recognition," *Data Mining and Knowledge Discovery*, vol. 2, pp. 121-167, Jun 1998.

[20] B. Samanta and K. R. Al-Balushi, "Artificial neural network based fault diagnostics of rolling element bearings using time-domain features," *Mechanical Systems and Signal Processing*, vol. 17, pp. 317-328, Mar 2003.
[21] (2011, 04.04). *Bearing data centre*. Available: http://www.eecs.cwru.edu/laboratory/bearing
[22] S. S. AGASHE and P. P. REGE, "Vibration Signal Analysis for Fault Identification of a Control Component," *International journal of Performability Engineering*, vol. 5, pp. 357-366, 2010.
[23] R. B. W. Heng and M. J. M. Nor, "Statistical analysis of sound and vibration signals for monitoring rolling element bearing condition," *Applied Acoustics*, vol. 53, pp. 211-226, Jan-Mar 1998.
[24] E. Romero and D. Toppo, "Comparing support vector machines and feedforward neural networks with similar hidden-layer weights," *Ieee Transactions on Neural Networks*, vol. 18, pp. 959-963, May 2007.

Yuan Fuqing obtained his M.Tech. in System Engineering at Beijing University of Aeronautics and Astronautics, China, in the year 2007. He joined the Division of Operation and Maintenance Engineering, Luleå University of Technology, Sweden, in September 2007 to study for the degree of PhD. His area of research deals with reliability data analysis and statistical learning theory.

Dr Uday Kumar obtained his B.Tech. in India during the year 1979. After working for 6 years in Indian mining companies, he joined the postgraduate programme of Luleå University of Technology, Luleå, Sweden, and obtained the degree of PhD in the field of Reliability and Maintenance during 1990. Presently, he is Professor of Operation and Maintenance Engineering at Luleå University of Technology, Luleå, Sweden. His research interests are equipment maintenance, equipment selection, reliability and maintainability analysis, system analysis, etc. He has published more than 170 papers in international journals and conference proceedings.

Dr Diego Galar is a telecommunications engineer and possesses a PhD in Manufacturing from the University of Saragossa. He has been a professor at several universities, including the University of Saragossa and the European University of Madrid. He has also been the Technical Director of ANANTASA, and the Director of Academic Innovation and subsequently Pro-Vice-Chancellor at Universidad San Jorge. He has been a researcher at the Department of Design and Manufacturing Engineering at the University of Saragossa and a researcher at I3A, the Institute of Engineering Research in Aragon. At present he is a senior researcher at Luleå University of Technology (LTU).

PAPER V: **Y. Fuqing**, U. Kumar and D. Galar, "Fault Diagnosis on time domain for Rolling Element Bearings using Support Vector Machine," *Reliability Engineering & System Safety*, 2011.(Submitted)

Fault Diagnosis on time domain for Rolling Element Bearings using Support Vector Machine

Yuan Fuqing[*], Diego Galar , Uday Kumar and Matti Rantatalo

Division of Operation and Maintenance Engineering
Luleå, SE-971 87, Sweden

Abstract: Feature extraction is crucial to efficiently diagnose fault. This paper discusses a number of time-domain statistical features, including Kurtosis or the Crest Factor, the Mean by Variance Ratio (MVR), and Symbolized Sequence Shannon Entropy (SSSE). The SSSE reflects the spatial distribution of the signal which is complementary with the statistical features. A new feature, Normalized Normal Negative Likelihood (NNNL), is used to improve the Normal Negative Likelihood (NNL). A Separation Index (SI) called the Extended SI (ESI) evaluates the performance of each feature and to remove noise feature. The Multi-Class Support Vector Machine (MSVM) recognizes bearing defect patterns. A numerical case is presented to demonstrate these features, their selection method and the pattern recognition method. The MSVM is used to detect three different types of bearing defects: defects in the inner race, outer race and bearing ball.

Keywords: *Fault Diagnosis, Time domain; Feature extraction; Extended Separation Index (ESI); Multi-class Support Vector Machine (MSVM).*

1. Introduction

Rolling element bearings are critical components in industrial machinery and thus play an important role in industry [1]. Because unexpected failures in the bearings lead to costly downtime [2], much research has considered their prevention. Fault diagnosis aims to find the incipient failure so that the catastrophic failure can be prevented. Within this paper, fault diagnosis refers to determining the state of the system (faulty or normal) as well as the type of faults [3].

The fault diagnosis of rolling element bearings is generally done manually based on practical experience. This is an intuitive and subjective method; therefore, when numerous patterns are involved, this method is inefficient. More recently, pattern recognition has been used to diagnose fault automatically. There are four steps in automatic fault diagnosis: signal acquisition, feature extraction, feature selection and pattern recognition. Signal acquisition installs sensors and collects the signal for analysis. Feature extraction is to extract the numerical measurements of the signal that

[*]*To correspond with author: Tel.: +46920491682; fax: +46920491935*
E-mail: yuan.fuqing@ltu.se
Postal address: Div of Operation and Maintenance Engineering, Luleå University of Technology, Luleå, SE-971 87, Sweden.

is more interpretable than the raw signal [4]. Feature extraction generates measurements, and feature selection chooses a subset of optimal features based on those measurements. Finally, pattern recognition techniques such as classification or cluster algorithms can be used to discriminate among different bearing states, for example discriminating failure patterns from normal patterns [4].

Signal acquisition in the case of the rolling element bearing focuses on collecting the vibration signal. The presence of a defect in a bearing produces impulses from the contact between metal surfaces when the bearing is running. These impulses are periodically produced with a frequency determined by the location of the defect and its size [5]. Accelerometer sensors can be mounted on the bearing house to measure the vibration signal which contains the defect information. As the vibration signal can be collected online, by analyzing this signal the status of the bearing, e.g. the location of the defect and possibly the severity of the problem, can be diagnosed without stopping the machine. This is the advantage of vibration analysis.

Feature extraction is very important as the raw vibration signal is normally too chaotic to find defect directly. A feature can be considered to represent the signal and can therefore be used as the indicator of the bearing state. Features can be extracted either from time domains, such as Kurtosis, Crest Factor, etc., or from frequency or time-frequency domains. Most analyses of vibration signals use the frequency domain or time-frequency domains. Nevertheless, the use of time domains has been argued to be computationally efficient [6], understandable and more easily implemented. Additionally, as argued by Tao et al., the time domain feature is less sensitive to bearing load and speed variance [7]. Finally, in the early stages of the defect, the frequency features are not significant, and frequency domain analysis is inefficient.

Extracting a large number of features may provide more information on the signal, but it adversely affects the computational cost inversely and could degrade the performance of diagnosis. In feature selection, a set of optimal features are chosen. State-of-the-art feature selection methods are individual feature selection and subset feature selection. The Separation Index (SI) can be used to measure the classifiability of an individual feature or a subset feature and thus can be used to perform feature selection [8, 9]. For example, Zio et al. propose an SI called the classifiability evaluation function to select a subset of features. Qiu et al. propose a method based on the geometry distance. Kim et al. develop one SI to evaluate the significance of an individual feature. This paper extends Kim's SI [10] by improving the statistical property and using it to remove noise feature.

A classification algorithm can be used to discriminate patterns; the traditional algorithms used are the linear classifier and Bayesian classifier [4]. More recently, researchers have advocated classification algorithms such as neural networks and support vector machine as an optimal way to automatically discriminate between faulty and normal patterns. These advanced methods can adapt their decision to new data automatically, i.e. self learning. More importantly, the advanced algorithms can perform classification flexibly. Chen and Lee have proposed a neural network method to identify failure patterns for F-16 aircraft [11]. Pfeufer and Ayoubi proposed a hybrid of fuzzy logic and neural networks to perform fault diagnosis [12]. Sugumaran et al. used Multi-class SVM to diagnose rolling bearings using features from the time

domain [13]. Abbasion et al. used Wavelet analysis and SVM to detect different types of faults [6] .

This paper discusses state-of–the-art time domain features and proposes some new ones to represent vibration signal more accurately; these include features adopted from another field and improvements on existing ones. A separation index is proposed to remove noise feature, and a commonly used subset selection method chooses the optimal subset of features. The optimal subset features are used as SVM input to diagnose fault. The multi-class SVM is selected to discriminate fault patterns, as the SVM can be linear or nonlinear depending on the kernel function chosen.

Section 2 of the paper discusses state-of-the-art time domain features and the proposed new feature. Section 3 presents a feature selection method. Section 4 discusses the multi-class SVM as a classifier of time domain features for diagnosis and prognosis purposes. Section 5 presents a numerical example using public test-rig data. Section 6 discusses the findings.

2. Feature Extraction from Time Domain

2.1 State-of-the-art features in the time domain

In the early stage of fault development, the bearing is not significantly damaged and the defective signal is masked by the noise. As the periodicity of the occurrence of the signal is not significant, the spectral analysis is ineffective. Even when the periodicity is significant, using the time domain feature is still recommended because normal and defective signals differ in their statistical characteristics.

Some time domain features found in the literature shows in Table 1. Kurtosis is an important and popular feature used in rolling element machines. It defines the peakedness of the amplitude of the signal. Beta parameters are the shape and scale parameters in the Beta distribution when the amplitude of the signal is assumed to follow a Beta distribution. This is a flexible distribution and most signals can fit it. Since the parameters in Beta distribution for a normal vibration signal (bearings without defects) and a defective signal (bearings with defects) differ, they can be used to differentiate between types of defects [14]. However, some critics, such as Heng and Nor, argue that the Beta method has no significant advantage over using the Kurtosis and Crest factor for rolling element bearings [14].

Table 1. State-of-the-art time domain features

	Feature	*Definition*		*Feature*	*Definition*
1	Peak value	Pv=(1/2)[max(xi)-min(xi)]	6	Clearance factor	Clf = $\frac{Pv}{(\frac{1}{n}\sum_{i=1}^{n}\sqrt{\lvert x_i\rvert})^2}$
2	RMS	RMS = $\sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i)^2}$	7	Impulse Factor	Imf = $\frac{Pv}{\frac{1}{n}\sum_{i=1}^{n}\lvert x_i\rvert}$
3	Standard Deviation	Std= $\sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i-\bar{x})^2}$	8	Shape factor	Shf = $\frac{RMS}{\frac{1}{n}\sum_{i=1}^{n}\lvert x_i\rvert}$
4	Kurtosis Value	Kv= $\frac{\frac{1}{n}\sum_{i=1}^{n}(x_i-\bar{x})^4}{RMS^4}$	9	Normal Negative Likelihood value	NNL= $-\ln L$; $L=\prod_{i=1}^{N} f(x_i,u,\sigma)$
5	Crest factor	Crf = Pv/ RMS	10	Beta Parameter	The estimated parameter values (α,β) in beta function $B(\alpha,\beta)=\int_0^1 x^{\alpha-1}(1-x)^{\beta-1}dx$ are used as features

The Kurtosis, Crest and Impulse factors are non-dimensional features and are independent of the magnitude of the signal power. RMS, Peak value, standard deviation, and Normal Negative Likelihood (NNL) value are fully dependant on the signal power. Some nuisance factors such as the quality of the sensors and the location where they are mounted can influence the power of the signal. The main advantage of non-dimensional features is that they are more immune from nuisance factors than dimensional features. RMS is an important feature in signal processing. It measures the power of the signal and can be used to normalize the signal. Therefore, some features are normalized by RMS. Certain other features used in the past are normalized by RMS, as for example, Beta-Kurtosis [15], Weibull negative likelihood value [16], Kurtosis Ratio [17], etc. They are not discussed here as the focus is on commonly used time domain features.

2.2 Normalized NNL

Normal Negative Likelihood (NNL) has been used by some researchers to diagnose fault [16]. In NNL, the amplitudes of the signal are assumed to follow Normal distribution. The parameters u and σ are calculated using the maximum likelihood estimator method. This paper proves that the performance of NNL is equivalent to a much simpler feature.

Let the amplitudes of the signal denoted by a series $x_1, x_i, \ldots. x_n$ discretely. When parameters u and σ are unknown, the negative likelihood function of this series is:

$$f(x_1,x_2,...,x_n\,|\,\mu,\sigma^2)=\prod_{i=1}^{n} f(x_i\,|\,\mu,\sigma^2)=(\frac{1}{2\pi\sigma^2})^{n/2}\exp(-\frac{\sum_{i=1}^{n}(x_i-\mu)^2}{2\sigma^2}) \qquad (1)$$

The maximum likelihood estimator of u and σ is:

$$\hat{u} = \bar{x} = \sum_{i=1}^{n} x_i / n \text{ and } \hat{\sigma}^2 = \frac{\sum_{i=1}^{n} (x_i - \hat{\mu})}{n} \quad (2)$$

Substituting Formula (2) into Formula (1) and simplifying it, the following equation is obtained:

$$f(x_1, x_2, ..., x_n | \mu, \sigma^2) = (\frac{1}{2\pi})^{n/2} \exp(-n/2).\sigma^{-n} \quad (3)$$

Thus the negative likelihood is:

$$Ln = \ln\left((\frac{1}{2\pi})^{n/2} \exp(-n/2) \right) - n \ln \sigma \quad (4)$$

It can be concluded from the above that NNL is essentially equivalent to $\ln\sigma$ and obviously, $\ln\sigma$ is not non-dimensional. In order to make the feature independent of power, as Kurtosis does, we normalize it by using RMS in the following way:

$$NNNL = \ln \frac{\sigma}{RMS} \quad (5)$$

This new feature is called Normalized Normal Negative Likelihood Value (NNNL).

Essentially, the old NNL is not a stable feature as it non-necessarily depends on the number of sample size *n* as shown in Formula (4). For a scenario where two signals are identical but differ in length, the NNL values will differ. This is evidently not reasonable and is the major disadvantage of this feature. The new feature, NNNL, is not only independent of the sample size but independent of the signal power.

2.3 Mean Variance Ratio

The distribution of amplitude in the signal samples differs from normal and defective rolling element bearing signal. The normal signal without defect is comprised by some noise signals and the shape of the signal thus tends to be peak. The distribution of defective signal has more wide amplitude so the variance is bigger than that of normal. It can be shown from Figure 1. The left figure in Figure 1 is from normal signal of a bearing and the other one from defective signal from the same bearing. It is evident that the defective signal differs from normal signal and it has wider variance.

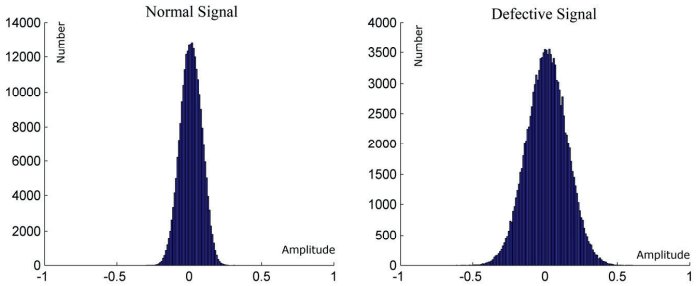


Figure 1. Normal and Defective signal

Therefore, it is straightforward to be reminded that the Mean and Variance Ratio (MVR) could be a feature to discriminate both defective and normal condition signals. The definition of MVR is:

$$\mathrm{MVR} = \frac{\frac{1}{n}\sum_{i=1}^{n} x_i}{\sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^2}} \tag{6}$$

Obviously, MVR is also a non-dimensional feature independent of signal power. MVR implies the degree of scatter for the distribution of signal amplitude.

2.4 Symbolized Sequence Shannon Entropy

All the features described in Table 1 are statistical features. These features consider statistical characteristics of the amplitude distribution; however, in all of these features, the information on the spacious pattern of the amplitude is lost. For example, for rolling element bearing, when defect exists, the amplitude tends to be periodic and however this periodicity can not be reflected in the statistical features. Figure 2 shows a simple example to verify this argument. This figure is comprised by 100 samples. The amplitude of each sample is comprised by {1,2,3,4} and each value has identical probability of appearance. The upper figure and the lower figure in Figure 2 are plotted by same samples, but with different spacious distribution. In the upper figure, the signal is periodic where the amplitude is distributed deterministically with a sequence of 1234, 1234,… iteratively. In the lower figure, the amplitudes are randomly distributed.

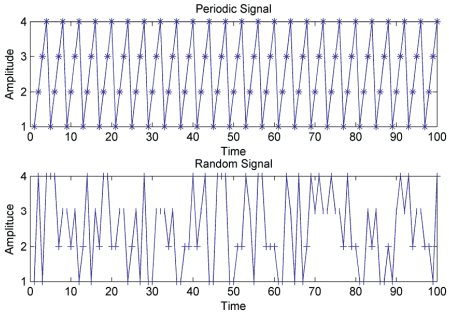


Figure 2. Periodic Signal and Random Signal

From the Figure 2, it is evident that both signals are different. However their statistical features are the same, i.e. the statistical features are not able to discriminate them.

The Shannon entropy has been known as a parameter capable to measure the uncertainty of a random process. Rolling element bearing without defect tends to generate a more random signal, while the machine with existing defect usually tends to have more deterministic signal, i.e. their Shannon entropy will be different. To extract the periodicity in the signal, a feature named Symbolized Sequence Shannon Entropy (SSSE) is used. In this feature, the signal is firstly symbolized and then the Shannon entropy is used. This SSSE has been used to detect weak signal in other research fields [18, 19]. This paper uses the SSSE to diagnose fault for vibration signal for rolling element bearing. The procedure for SSSE calculation is:

1. Discretize the signal. A threshold is predefined. The amplitude below the threshold is coded as 0 and the above is coded as 1. Thus the signal is discretized into a binary sequence, which is denoted by
$$b_1, b_2, b_3, ...b_i, ...$$
2. Segment the binary signal with equal length L. For example, segment the binary sequence 110010010 into 110, 010, 010 with length L=3. Calculate the decimal value of each segment. It is “6”, “2” , “2” in this example.
3. Calculate the probability of each segment. The probability is considered as the frequency. For “6” in this example, it is 1/3 and for “2” it is 2/3.
4. Calculate the entropy using the following Shannon entropy formula:
$$\mathrm{H} = -\frac{1}{\log N}\sum_i p_i \log p_i \qquad (7)$$

The N is the total number of unique segmented binary sequence. The p_i is the probability of the i^{th} kind of the unique sequence.

In a periodic signal, some sequences will occur frequently so the Shannon entropy will be lower. Therefore, the Shannon entropy values vary with different acquired data so it can be used as feature to measure the characteristics of a signal. For a pure random data, the Shannon entropy value is 1. For deterministic signal, the entropy is between 0 and 1 [19]. The more deterministic is the signal, the lower its SSSE value. Using the above procedure, for the example of Figure 2, the SSSE of periodic signal (upper in Figure 2) is 0 and that of the random signal is 0.905 (lower in Figure 2). These two signals can be significantly discriminated. Similarly different defect in rolling element bearing can be discriminated attending to the randomness existing in the acquired signals.

2.5 Simulation of feature performance

Similarly to the method discussed by Heng and Nor [14], where the authors discuss different features by using simulation signals, this paper simulates various signals to test the performance obtained with different features. The simulated signals are shown in Figure 3.

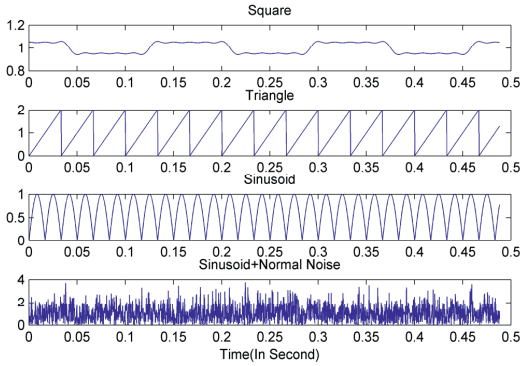


Figure 3. Simulated Signals

The feature values are shown in Table 2 for the signals shown in Figure 3. For the different signals, their feature values differ. This means these features can

discriminate these signal. The feature SSSE is excellent to discriminate deterministic signal and noise signal. The SSSE shows a small value around 0.1 for the deterministic signal, such as Square, Sin, Triangle, but for the for random signals, such Gaussian random and uniform random, the SSSE values are large above 0.9.

Table 2. Time Domain Feature Values for Simulated Signals

Signals	***Kurtosis***	***Crest factor***	***NNNL***	***MVR***	***SSSE***	***Shape***	***Clear***
Square	1.0866	0.059	3.02	20.42	0.12	1.00	0.06
Triangle	1.8	0.87	0.69	1.73	0.19	1.15	1.12
Sin	1.93	0.71	0.83	2.07	0.26	1.11	0.86
N(0,1)	3.91	2.08	0.50	1.32	0.90	1.25	3.07
U(0,1)	1.79	0.86	0.70	1.74	0.91	1.15	1.12
Sin+N(0,0.1)	1.97	0.95	0.81	2.02	0.46	1.11	1.17
Sin+N(0,1)	3.33	1.78	0.52	1.35	0.90	1.24	2.60

Note: N(0,1) denote normal distribution with mean 0 and variance 1, U(0,1) denotes uniform distribution ranged in [0,1].

To test noise sensitivity, one experiment has been done with the sinusoidal signal. Two white Gaussian noise N (0,0.1) and N(0,1) have been added to the same sinusoidal signal respectively. The results, shown in Table 2, show that all the time domain features differ so it unfortunately implies that time domain is sensitive to noise.

3. Feature Selection

3.1 Noise feature removal using Separation Index

This paper extracts several features from time domain. Some features may not contribute to the fault diagnosis and even degrade the performance of the diagnosis. In order to remove these non significant features or noise features, an index named Separation Index is used to define the significance of features [16].

For the sake of simplicity, each feature value is called a sample in this section. If two signals are presented to be compared, let $\overline{m}_d$ and $\overline{m}_h$ denotes the mean of samples, S_d and S_h denotes standard deviation. Kim et al. developed one Separation Index (SI) as [10]

$$SI = \frac{\left|\overline{m}_d - \overline{m}_h\right|}{S_d + S_h} \tag{8}$$

This separation index is used defining the separability of features. This paper has improved this SI to gain a better statistical property. The new SI, which is called Extended SI (ESI), as:

$$SI = \frac{\left|\overline{m}_d - \overline{m}_h\right|}{\sqrt{S_d^2 + S_h^2}} \tag{9}$$

Assume the samples are normal distribution and the number of sample size from each signal is equal with n. The quantity $SI \times \sqrt{n}$ can be test its significance using t-test. That is, when

$$SI \times \sqrt{n} > t_{\sigma,v} \tag{10}$$

the $\overline{m}_d$ and $\overline{m}_h$ has significantly difference, where σ is significance level and v is degree of freedom. The v in Inequality (10) is:

$$v = \frac{(1+u)^2(n-1)}{1+u^2} \tag{11}$$

where $u = S_d^2 / S_h^2$. The statistical proof of this can be found in reference [20].

The t-test proposed above assumes that the sample size is equal from both sides. For a problem with unequal sample size, one can use the same Separation Index but the t-test is different from Formula (11). One can refer to the reference [20] for more detail for the unequal sample size t-test.

The significance of the difference between two signals then has more statistical foundation by using ESI. This is the advantage of the new proposed separator index. This ESI can be used to remove noise feature. For example, when the feature values from two signals are tested significantly different, the feature will be retained; otherwise they should be excluded from further consideration.

3.2 Feature Subset Selection

The ESI can be used to remove some noise feature in the initial step of feature selection. Feature subset selection is to select the compact optimal feature set. The feature subset selection is necessary as features are possibly correlated or redundant. There are numerous methods available to perform subset selection [21] however the simplest one is the exhaustive method. This method enumerates all the subsets and selects the one with highest performance, for example the highest fault diagnosis accuracy, as the optimal. This exhaustive method is computational cost, as for a number of n features, there have as much as 2^n subsets, therefore, the exhaustive method is only suitable for small number of features. The advantage of the exhaustive method is a global optimal solution can be obtained. The detailed procedure of performing feature selection in this paper is shown in Figure 4.

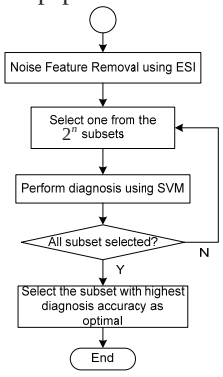


Figure 4. Feature Selection Procedure

4. Pattern Recognition using SVM

The standard SVM is a binary classifier classifying two classes of objects. To accommodate the multi-classes problems, as the problem shown in Figure 5, one has to extend the standard SVM. One solution is to combine several binary SVMs together. One-Against-All multi-class SVM is one of them.

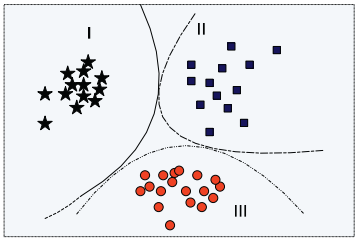


Figure 5. Three-Classes Classification Problem

The One-Against-All method transforms the k-class problem into a k sub binary classification problem. The i^{th} sub binary classification problem labels the indicator of data sets belong to the i^{th} class with 1 and label all the remaining data sets with -1 so the binary SVM is formulated as [22]:

$$\begin{aligned} &\min \frac{1}{2}(\omega^i)^T \omega^i + C\sum_{j=1}^{l}\xi_j^i(\omega^i)^T \\ &(\omega^i)^T \phi(x_j) + b^i \geq 1 - \xi_j^i, if\ y_j = i \\ &(\omega^i)^T \phi(x_j) + b^i \leq -1 + \xi_j^i, if\ y_j \neq i \\ &\xi_j^i \geq 0, j = 1,....,l \end{aligned} \qquad (12)$$

where $(x_1, y_1),...,(x_l, y_l)$ denotes the training data, l is the total number of training data sets. The x_i is the input of the SVM, which is the feature vector in this paper. The $y_i \in \{1,2,3,..k\}$ is the output of the SVM which is the indicator of the category (class). The vector ω^i is the coefficient for vector $\phi(x_j)$. b is the threshold.

Each sub binary SVM has one decision function. For a k-Classes SVM, one hence can obtain k decision functions:

$$\begin{aligned} &(\omega^1)^T \phi(x) + b^1 \\ &\cdots \\ &(\omega^k)^T \phi(x) + b^k \end{aligned} \qquad (13)$$

The predicted class for x is the class with largest decision function value, as illustrated in Figure 6. It is:

$$i = \arg\max((\omega^i)^T \phi(x) + b^i) \qquad (14)$$

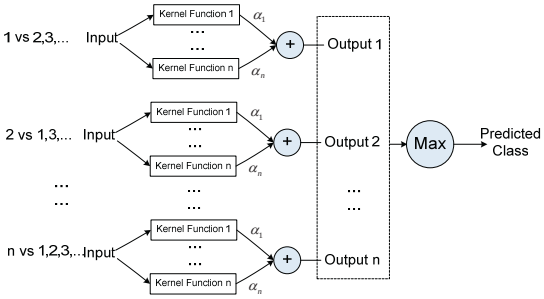


Figure 6. Multi-Class SVM

SVM is a kernel method which depends on the kernel function [23, 24]. Before one uses the SVM for pattern recognition, one should select a proper kernel function. The discussion on kernel function is omitted as it is not the concern of this paper. The motivation to select the SVM for fault diagnosis is because the decision function is flexible as it can be linear or nonlinear depending on the kernel function.

5. Numerical Case

5.1 Data Description

This numerical case uses the public bearing data which is collected from a test rig in Case Western Reverse Lab [25]. In this rig two bearings are installed which are located at the end of the driver and fan respectively. Artificial defect are introduced to the inner race, outer race and ball in the bearings. Figure 7 shows a bearing picture with defect in inner race.

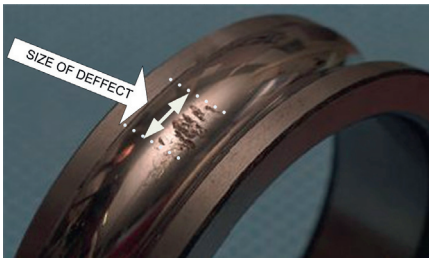


Figure 7. Defect in Inner Race

The vibration data is collected by accelerometers attached to the housing with magnetic bases. The data using in this case is from the bearing located in the driver end which is a signal sampled at 12 kHz as shown in Figure 8.

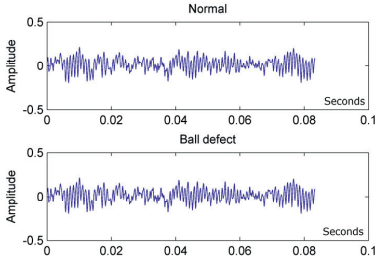


Figure 8. Normal and Ball Defect Signal

5.2 Performance of individual feature

The raw signal is divided into several non-overlapping segments using a fixed window size. The window size used in the case study is 3000 for each segment which is more than the minimal requirement $w_{len} = 296$ suggested by following Inequality [17]:

$$w_{len} \geq \frac{4 \cdot f_s}{f_{BDF}} \tag{15}$$

where f_s denotes the sampling frequency and f_{BDF} denotes the fault frequency, for example the ball pass frequency on the inner race.

The signals used are a normal bearing signal and ball defective signal shown in Figure 8. Features extracted from time domain are: MVR, SSSE, NNNL, Kurtosis, Crest Factor, Clear Factor, Impulse Factor, Shape factor. All these features are non-dimensional features. By using the t-test described in Section 3.1, the feature SSSE is significantly equal for the two signals so it is removed for consideration in this comparison. The ESI value is computed by Eq(9) and is compared with the accuracy obtained by simplest linear polynomial function SVM, as shown in Table 3, where the 0.8, 0.6, 0.5 means the ratio selected as training data, the remaining data sets are test datasets. The selection of the data sets is random and each ratio is repeated 30 times. The number listed in Table 3 is the mean of the 30 runs.

Table 3. Accuracy using Single Feature

Features	*Kurtosis*	*Shape*	Clear	NNNL	MVR	Impulse	Crest
SI	1.17	1.01	0.92	1.01	1.01	0.89	0.84
Accuracy(%) for Ratio 0.8	88.17	84.6	85	83.75	81.75	74.33	64.84
Accuracy(%) for Ratio 0.6	86.29	84	84.04	82.75	80.79	74.71	68.83
Accuracy(%) for Ratio 0.5	86.14	83.91	83.80	82.71	80.27	75.24	68.03

As shown in Table 3, the higher ESI implies the accuracy tends to be higher. The Kurtosis has the highest ESI, so the accuracy is also the highest. The ESI of the Crest is the lowest, the corresponding accuracy also lowest. This alignment implies the efficiency of ESI as a feature performance indicator.

5.3 Fault Diagnosis using MSVM

There are tree types of defects introduced in the inner race, out race, ball of the bearing. Therefore, including normal state (bearing without defect), four patterns are needed to be discriminated: normal, inner race defect, outer racer defect, and ball bearing defect. Multi-SVM is used to recognize these patterns. Segment the signal obtained from accelerometer sensors and use the t-test proposed in Section 3.1 to filter noise feature. The test results show that except feature SSSE, all the other features are significantly difference. The SSSE fails to pass the t-test between normal and ball defect. However, the SSSE is significant different between other signals. Therefore this feature is still kept in the feature selection. As a consequence, the input of SVM is the vector of feature values: MVR, NNNL, Crest Factor, Clear Factor, Impulse Factor,

Shape factor, Kurtosis, SSSE. The output of the SVM is the status of the bearing: Normal, Inner Race defect, Ball defect, Outer race defect.

5.3.1 Pattern recognition using MSVM

The polynomial function as written follows is selected as kernel function [26, 27]:

$$K(x, x') = (< x, x' > +1)^d \qquad (16)$$

The parameter d above is a predefined parameter. When d=1, the SVM is a linear classifier; When $d > 1$ it is nonlinear. The higher of d, the more flexible of the SVM classifies the data; whereas a too flexible classifier is prone to occur "overfitting" phenomenon [28], which means a classifier has low train error but have a high prediction error. The Figure 9 shows the "overfitting" phenomenon. In this figure, the training accuracy is always increasing with d. This means increasing the order can always improve the training accuracy as the classifier becomes more flexible. After d>4, the training accuracy reaches 100%. However, the test accuracy is not increasing with d, inversely it deceases with the order when $d \leq 4$. The higher order d turns out to have lower test accuracy. This is the "overfitting" phenomenon. The highest test accuracy in the figure is order d=1. Therefore, in this case, d=1 is selected for the polynomial kernel function.

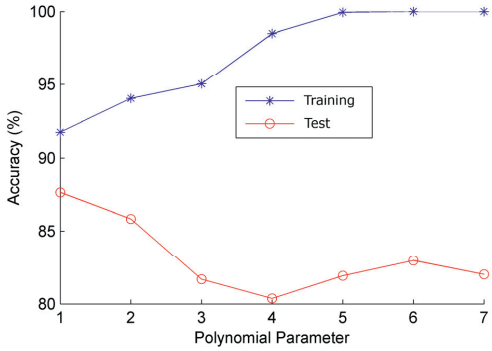


Figure 9. Performance of Various Kernel Parameters

Divide the data extracted from signals at ratio 0.8, 0.6, 0.5 respectively and the first part is used for training, the second for test. To reduce randomness, the division of data for training and test are also random and each subset has been ran 30 times. Utilizing the approach mentioned in Section 3.2 selects the optimal subset features. The resulting optimal subset and its diagnosis accuracy, that is the mean of the 30 runs, are shown in Table 4.

Table 4. Optimal Subsets and Their Accuracy

Features	Kurtosis	Crest	Clear	Impulse	Shape	NNNL	MVR	SSSE	Accuracy%
Ratio 0.6	√	√	√	√	/	/	√	√	98.71
	√	√	√	√	√	√	/	√	98.30
Ratio 0.5	√	/	√	√	√	/	/	√	98.55
	√	√	√	/	/	√	/	√	97.95
Ratio 0.8	√	/	√	√	√	/	/	√	99.50
	√	√	√	/	√	√	√	√	99.25

Note: "√" denote the feature in the subset, "/" denote not in the subset

All the optimal subsets include feature Kurtosis, SSSE, Impulse and Shape. These four features are common features in the subsets having highest accuracy. The feature MVR is also in the optimal subset of Ratio 0.6, which means it is a good candidate feature for time domain fault diagnosis. Moreover, the proposed new NNNL is also important. The second row for each ratio in Table 4 list the subsets containing NNNL, which have accuracy very close to the highest accuracy. Nonetheless, the Kurtosis and SSSE are the most incredible in this case.

All the statistical features contain only the amplitude distribution of the signal. The information about the spacious distribution of the signal has lost. This lost can be proved in the simulated example in Section 2.4. However, the feature SSSE considers the spacious distribution information but it loses the amplitude distribution information. In this sense, the SSSE and the statistical features are complementary. Therefore, combining the statistical features and the SSSE could achieve higher diagnosis accuracy, which can be verified from the results in Table 4.

6. Conclusion

The fault diagnosis on rolling element bearing is mostly on frequency domain. This paper proposes an approach on time domain. The feature on time domain is less dependent on the machine load and rotation speed, and when the defect is in its early stage, the defect frequency is insignificant and diagnosis on time domain is necessary. The numerical case shows the diagnosis on time domain is feasible.

The feature SSSE shows incredible in the numerical case thus it shows its effectiveness used for fault diagnosis. The new feature NNNL has also shown its importance as using this feature the diagnosis accuracy is high as well. The proposed MVR is in the optimal subset so it is a feature of importance for the diagnosis. The SSSE is complementary with the other statistical features and thus combining them can have a good fault diagnosis result. This can be verified from the bearing case.

The proposed Extended Separation Index shows its efficiency in the bearing case. In the bearing case, the higher ESI implies the higher diagnosis accuracy. Moreover, the MSVM used in this paper shows its flexibility by adapting itself to data. By tuning the order in the Polynomial kernel function, the SVM can be linear or nonlinear. In the bearing case, it finally uses the simplest linear Polynomial kernel function that shows better performance than the more complex nonlinear Polynomial kernel function.

References

[1] Su WS, Wang FT, Zhu H, Zhang ZX, Guo ZG. Rolling element bearing faults diagnosis based on optimal Morlet wavelet filter and autocorrelation enhancement. Mechanical Systems and Signal Processing. 2010;24:1458-72.

[2] Zio E, Gola G. A neuro-fuzzy technique for fault diagnosis and its application to rotating machinery. Reliability Engineering & System Safety. 2009;94:78-88.

[3] Akbaryan F, Bishnoi PR. Fault diagnosis of multivariate systems using pattern recognition and multisensor data analysis technique. Comput Chem Eng. 2001;25:1313-39.

[4] Theodoridis S, Koutroumbas K. Pattern recognition, fourth edition. 4th ed. Burlington, Mass.: Academic Press; 2009.
[5] Mcfadden PD, Smith JD. Model for the Vibration Produced by a Single Point-Defect in a Rolling Element Bearing. J Sound Vib. 1984;96:69-82.
[6] Abbasion S, Rafsanjani A, Farshidianfar A, Irani N. Rolling element bearings multi-fault classification based on the wavelet denoising and support vector machine. Mechanical Systems and Signal Processing. 2007;21:2933-45.
[7] Tao B, Zhu LM, Ding H, Xiong YL. An alternative time-domain index for condition monitoring of rolling element bearings - A comparison study. Reliability Engineering & System Safety. 2007;92:660-70.
[8] Qiu WL, Joe H. Separation index and partial membership for clustering. Computational Statistics & Data Analysis. 2006;50:585-603.
[9] Zio E, Baraldi P, Roverso D. An extended classifiability index for feature selection in nuclear transients. Annals of Nuclear Energy. 2005;32:1632-49.
[10] Y.Kim E, C.C.Tan A, Yang B-S, Kosse V. Experimental Study on Condition Monitoring of Low Speed Bearings:Time domain Analysis. 5th Australasian Congress on Applied Mechanics,ACAM2007. Brisbane,Australia2007.
[11] Chen YM, Lee ML. Neural networks-based scheme for system failure detection and diagnosis. Mathematics and Computers in Simulation. 2002;58:101-9.
[12] Pfeufer T, Ayoubi M. Application of a hybrid neuro-fuzzy system to the fault diagnosis of an automotive electromechanical actuator. Fuzzy Sets and Systems. 1997;89:351-60.
[13] Sugumaran V, Sabareesh GR, Ramachandran KI. Fault diagnostics of roller bearing using kernel based neighborhood score multi-class support vector machine. Expert Syst Appl. 2008;34:3090-8.
[14] Heng RBW, Nor MJM. Statistical analysis of sound and vibration signals for monitoring rolling element bearing condition. Applied Acoustics. 1998;53:211-26.
[15] Wang WQ, Ismail F, Golnaraghi MF. Assessment of gear damage monitoring techniques using vibration measurements. Mech Syst Signal Pr. 2001;15:905-22.
[16] B.Sreejith, A.K.Verma, A.Srividya. Fault diagnosis of rolling element bearing using time-domain features and neural networks. Proceedings of Third IEEE International Conference on Industrial and Information Systems ICIIS 2008. Kharagpur,India2008.
[17] Vass J, Smid R, Randall RB, Sovka P, Cristalli C, Torcianti B. Avoidance of speckle noise in laser vibrometry by the use of kurtosis ratio: Application to mechanical fault diagnostics. Mechanical Systems and Signal Processing. 2008;22:647-71.
[18] Tang XZ, Tracy ER, Boozer AD, Debrauw A, Brown R. Symbol Sequence Statistics in Noisy Chaotic Signal Reconstruction. Physical Review E. 1995;51:3871-89.
[19] Finney CEA, Green JB, Daw CS. symbolic time-series analysis engine combustion measurements. Society of Automotive Engineers. 1998.
[20] Ruxton GD. The unequal variance t-test is an underused alternative to Student's t-test and the Mann-Whitney U test. Behav Ecol. 2006;17:688-90.
[21] Kohavi R, John GH. Wrappers for feature subset selection. Artificial Intelligence. 1997;97:273-324.
[22] Hsu CW, Lin CJ. A comparison of methods for multiclass support vector machines. Ieee T Neural Networ. 2002;13:415-25.
[23] Vapnik VN. Statistical learning theory. New York: Wiley; 1998.

[24] Fuqing Y, Kumar U, Galar D. Reliability prediction using support vector regression International Journal of Systems Assurance Engineering and Management 2011;1:263-8.
[25] Bearing data centre. Case western reserve university.
[26] Schölkopf B, Burges CJC, Smola AJ. Advances in kernel methods : support vector learning. Cambridge, Mass.: MIT Press; 1999.
[27] Vapnik VN. The nature of statistical learning theory. New York: Springer; 1995.
[28] Hastie T, Tibshirani R, Friedman JH. The elements of statistical learning : data mining, inference, and prediction. 2nd ed. ed. New York: Springer; 2009.