INTELLIGENT ARRHYTHMIA CLASSIFICATION BASED ON SUPPORT VECTOR MACHINES

by

Aslı Uyar Özkaya

B.S., Control and Computer Engineering, Istanbul Technical University, 2003

Submitted to the Institute for Graduate Studies in
Science and Engineering in partial fulfillment of
the requirements for the degree of
Master of Science

Graduate Program in Computer Engineering
Boğaziçi University

2006

ACKNOWLEDGEMENTS

I would like to thank my supervisor, Prof. Fikret Gürgen, for many insightful conversations during the development of the ideas in this thesis, and for helpful comments on this research, as well as for the many discussions carried on.

I also thank Prof. Ethem Alpaydın, for the courses and the materials which were referenced in this thesis. I am also grateful to him for his many helpful suggestions and comments on our experimental results.

My sincere appreciation goes out to Asst. Prof. Borahan Tümer, my committee member.

I also thank to my friends Yasemin and Evren for their moral support and technical assistance when I needed.

I am also grateful to my husband, Fatih. Throughout my thesis, he provided encouragement, sound advice, and lots of good ideas. I also thank him for his incredible patience and for his continued moral support. So, I thank him for all he has done for me.

Finally, I am forever indebted to my sisters Hüsne and Yıldız for their understanding, endless patience and encouragement when it was most required. Lastly, and most importantly, I wish to thank my small and sweet baby, Eren, for his innocent motivation. To him I dedicate this thesis.

ABSTRACT

INTELLIGENT ARRHYTHMIA CLASSIFICATION BASED ON SUPPORT VECTOR MACHINES

The main objective of this study is to provide automatic recognition of arrhythmic cardiac pathologies from the classification of ECG recordings. ECG is a graphical signal which is the result of electrical tension of heart and is the most important biosignal used by cardiologists for diagnostic purposes. The difficulty faced in interpretation of ECG signals forced researchers to study about automatic detection of cardiac arrhythmia disorders. Using intelligent data analysis techniques, computer programs could easily interpret complex ECG signals, predict presence or absence of cardiac arrhythmia and provide real-time analysis and diagnosis. In this study Support Vector Machines (SVM) technique has been applied to ECG dataset for intelligent arrhythmia classification. The dataset used in this study have been obtained from UCI repository. PCA and ICA methods have been used for dimensionality reduction of high dimensional ECG data. Parameter selection is very critical for SVM since its performance is greatly influenced by the model parameters. The results of the standard SVM classifier improved by parameter selection, dimension reduction and a threshold based rejection method to avoid false predictions for ambiguous patterns. The proposed threshold method provides uncertainty management and could be used for suppressing false alarms. As a comparison, k-Nearest Neighbor and Decision Tree algorithms have been tested on the arrhythmia dataset. According to experimental results improved SVM results shown to outperform competing classification results.

ÖZET

DESTEK VEKTÖR MAKİNELERİ KULLANILARAK ARİTMİ SINIFLANDIRMASI

Bu tezde yapılan çalışmanın temel amacı EKG işaretlerinin sınıflandırılması ile aritmik kardiyak hastalıkların otomatik olarak teşhisini sağlamaktır. EKG, kalpteki elektriksel gerilimin bir sonucu olan grafik şeklinde bir işarettir ve kardiyologlar tarafından teşhis amacıyla kullanılan en önemli veridir. EKG işaretlerinin yorumlanmasında karşılaşılan güçlükler araştırmacıları kardiyak aritmi bozuklukların otomatik olarak belirlenmesi konusunda çalışma yapmaya sevk etmiştir. Günümüzde akıllı veri analizi yöntemlerini kullanan bilgisayar programları karmaşık EKG işaretlerini kolayca yorumlayabilmekte, kardiyak aritmi varlığı konusunda öngörüde bulunabilmekte ve gerçek zamanlı analiz ve teşhis sağlamaktadır. Bu çalışmada, akıllı aritmi sınıflandırması için diğer sınıflandırma yöntemlerine göre daha yeni bir yöntem olan Destek Vektör Makineleri (DVM) kullanılmaktadır. Deneylerde kullanılan veriler UCI Aritmi Veritabanı'ndan elde edilmiş olup çok boyutlu EKG veri kümesinde boyut indirgeme için PCA ve ICA yöntemleri kullanılmıştır. DVM algoritmasının performansı model parametrelerine bağlı olduğundan parametre seçimi oldukça önemlidir. Yapılan çalışmada, parametre seçimi, boyut indirgeme ve eşik tabanlı bir reddetme yöntemi kullanılarak standart DVM sınıflandırıcısının performansı arttırılmıştır. Önerilen eşik tabanlı reddetme yöntemi ile belirsizlik yönetimi sağlanırken sınıflandırmadaki yanlış alarmların da ortadan kaldırılması hedeflenmektedir. Karşılaştırma amacıyla k-NN ve karar ağaçları yöntemleri de EKG veri kümesine uygulanmış ve deney sonuçlarına göre geliştirilmiş DVM yönteminin daha iyi sonuç verdiği belirlenmiştir.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

ANN	Artificial Neural Networks
CV	Cross Validation
DT	Decision Tree
ECG	Electrocardiogram
FP	False Positive
FR	False Rate
ICA	Independent Component Analysis
k-NN	k-Nearest Neighborhood
MLP	Multi Layer Perceptron
PCA	Principal Component Analysis
RBF	Radial Basis Functions
SVM	Support Vector Machines

1. INTRODUCTION

Bioinformatics integrates computational sciences and engineering principles to study medicine. In recent years, computer engineers advances informatics approaches for the prevention, diagnosis and treatment of diseases, for patient rehabilitation and for improving health.

Intelligent data analysis methods in medicine have attracted many researchers. Several researches and applications of intelligent systems in medical applications have been listed [1,2]. Artificial Neural Networks [3], Fuzzy Systems [4], Statistical Approaches [5] and in recent years a newer method Support Vector Machines have been implemented in medical applications [6]. All of these researches present several advantages over manual analysis in medical applications. Computer programs could easily interpret complex patient related data, predict future indicator values based on past data, provide automated real-time analysis and diagnosis and enables rapid identification and classification of input data. Intelligent "machine learning" methods are expected to be powerful tools to enhance current medical diagnostic techniques [1].

This study presents the results of application of Support Vector Machines for the classification of arrhythmic disorders. Cardiac arrhythmia could be determined via analyzing the Electrocardiogram (ECG) signals.

Signal processing and interpretation in medicine involves a complex analysis of signals, graphic representations and pattern classification [1]. The difficulty faced in interpretation of ECG signals, forced researchers to study about automatic detection of cardiac arrhythmic disorders. Most frequently Artificial Neural Networks have been used for arrhythmia classification [3,7,8].

In this study arrhythmia classification procedure involves, data acquisition, dimensionality reduction for the high dimensional input data, classification and interpretation of results. Complex ECG signals have been transformed to numeric data and created arrhythmia databases using ECG records of lots of patients. Two most popular Arrhythmia Databases could be obtained from UCI Repository [9] and MIT-BIH database.

The dataset used in this study has been obtained from UCI Arrhythmia Database. Because of the complex structure of ECG signals, the input data have too many attributes. Consequently, Principal Component Analysis (PCA) and Independent Component Analysis (ICA) have been applied for dimensionality reduction. Finally SVM classification method applied to obtain the predicted results. In addition, k-Nearest Neighbor Algorithm and Decision Tree classification methods have been applied to the arrhythmia dataset and the results have been compared to SVM. The results of classification task have also been presented in terms of medical biostatistics.

1.1. Motivation

In the present scenario, one of the main causes of death in developed and in emerging countries is the cardiovascular related disease [7]. It is suggested that most of these deaths could be avoided if there was a pre-monitoring and a pre-diagnostic of these cardiac arrhythmias by using an ECG tool. This enforces the researchers to study about automatic detection of cardiac arrhythmia disorders.

Automatic ECG analysis is critical for diagnosis and treatment of critically ill patients. Modeling and simulation of ECG under various conditions are very important in understanding the functioning of cardiovascular system as well as in the diagnosis of heart diseases. Computer-assisted arrhythmia recognition enables more reliable management of cardiac disorders. Various techniques have been utilized to classify arrhythmias. The reliable detection of arrhythmias constitutes a challenge for a cardiovascular diagnostic system. Consequently, significant amount of researches have focused on the development of algorithms for accurate diagnosis of arrhythmias.

The objective of this study is to classify certain cardiac arrhythmias using Support Vector Machines classification technique. Most of the techniques involve significant amounts of computation and processing time for extraction of features and classification task. The other disadvantage is the small number of arrhythmias classified using a given technique with most techniques being used to classify two to three arrhythmias. Most of the researches results between 70% to 80% accuracy [1].There is a need for extending a particular technique for improving the accuracy of classification. As a new intelligent data

analysis method, Support Vector Classifier applied to ECG records for arrhythmia classification, and the results of this application expected to meet these necessities.

1.2. Outline

Chapter 1 is the introductory part of this thesis. The motivation of the research and the outline is given.

Detailed information about the cardiac arrhythmia disorders and ECG signals are given in Chapter 2. The characteristics of dataset, which include ECG records of patients, are explained.

Chapter 3 presents what we have done in this study. The proposed arrhythmia classification system, applied dimensionality reduction techniques, Support Vector Machines as a classification algorithm, implementation of SVM to arrhythmia dataset and k-NN and DT classification algorithms have been explained in this chapter.

Chapter 4 includes the experimental results of this application. Performance analyses of the proposed model and comparison of the results with other classification techniques have been discussed. The results are given and analyzed in the means of medical biostatistics.

The conclusion part of the thesis is Chapter 5, a summary is given about the proposed model. Possible future research directions are also discussed in this chapter.

2. BACKGROUND

2.1. Medical Data Analysis

Intelligent data analysis methods in medicine have attracted many researchers. Several researches and applications of intelligent systems in medical applications have been listed [1,2].

In the last years, data mining and machine learning techniques have been remarkably successful in extracting valuable knowledge from ever growing medical databases. In this context, neural networks, decision trees, hidden markov models and recently support vector machines have been applied to medical diagnosis, bioinformatics as well as other application areas [10]. In addition the comparison of these machine learning techniques for medical decision making is an important issue. The classification models, k-Nearest Neighbor (k-NN), Decision Trees, Multi Layer Perceptrons (MLP), Radial Basis Functions (RBF) and Support Vector Machines (SVM) have been applied to different medical data sets and as a result MLP, RBF and SVM classifiers found to be valuable in medical decision making applications when enough data are available. The advantage of SVM structure is presented as the ability to control the sensitivity of the classification. “In many medical applications, medical specialists do not want to miss the False Positive: FP (Decision system labels it as positive while it is actually negative) cases. This makes SVM based system useful in many medical decision making applications.” [11]

2.1.1. Applications of SVMs to Medical Decision Support

The application of SVMs to medical decision support is an important issue for researchers [2,10,12]. Most of the researches propose applications of SVMs to automated diagnosis. The objective of such studies is to show that SVM classifiers are good candidates in this area [12].

An interesting research about rule-extraction from support vector machines for medical diagnosis provides an explanation capability to the black box model of the SVM

classifier [10]. This capability is important for the acceptance of this machine learning technique, especially for applications such as medical diagnosis. The main objective of this study is to extract high quality rules from trained SVMs.

Support Vector Machines have also been used for the recognition of Atrial and Ventricular depolarization in Holter ECG recordings. The study presents a new approach to the ECG holter analysis which enables the signal shape recognition including P, QRS and T waves. The recordings are filtered and segmented into single heartbeats. Then SVMs have been applied for approximation and classification. The researchers claim that they have obtained excellent performance on automatic recognition of normal vs. pathological heartbeats [13].

2.2. Arrhythmia Classification

Automatic detection of arrhythmia disorders is critical for diagnosis. Due to the large number of patients in intensive care units and the need for continuous observation of them, several methods for automated arrhythmia detection have been developed in recent years to attempt simplify the monitoring task.

An interesting study about real-time arrhythmia classification propose that the new advances in sensor technology, PDAs and wireless communications favor the development of a new type of monitoring systems that can provide patients with assistance anywhere at any time [14]. In the paper, the authors present MOLEC: a PDA-based system that performs local real-time classification and detects the ECG anomalies. This solution allows a real time classification anywhere and at any time where the PDAs can analyze ECG signals, detect anomalies, and make use of wireless communications in order to send those anomalous situations to the control center. The set of rules used to classify the beats have been extracted via applying techniques based on decision trees [14].

For cardiologists, the problem of ECG arrhythmia is to discriminate different kind arrhythmias from a normal cardiac rhythm. An automatic ECG analyzer will provide a cardiologist with a tool allowing a faster and more accurate diagnosis.

Automatic recognition of cardiac pathologies from the investigation of Holter ECG recordings is usually based on the analysis of heart rate variability and mainly on the shape of QRS complexes and ST segment [13].

2.2.1. ECG Signals

Electrocardiography is an important tool in diagnosing the condition of the heart. Electrocardiogram (ECG) is a graphical signal produced by the electrocardiograph which records the electrical tension of the heart activity. ECG signal is the result of sequential electrical depolarization that are coupled to contraction of the muscular heart chambers. The depolarization is detectable via electrodes placed on the body surface.

The ECG is the most important biosignal used by cardiologists for diagnostic purposes. The ECG signal provides key information about the electrical activity of the heart. Continuous ECG monitoring permits observation of cardiac variations over an extended period of time. Detection of abnormal ECG signals is a critical step in administering aid to patient. Early detection of heart diseases can prolong life and enhance the quality of living through appropriate treatment.

A typical ECG graphic consists of repeating P wave, PR interval, QRS complex, ST segment and T wave. A normal ECG is given in Figure 2.1. To recognize electrocardiographic abnormalities the range of normal wave patterns must be understood [15].

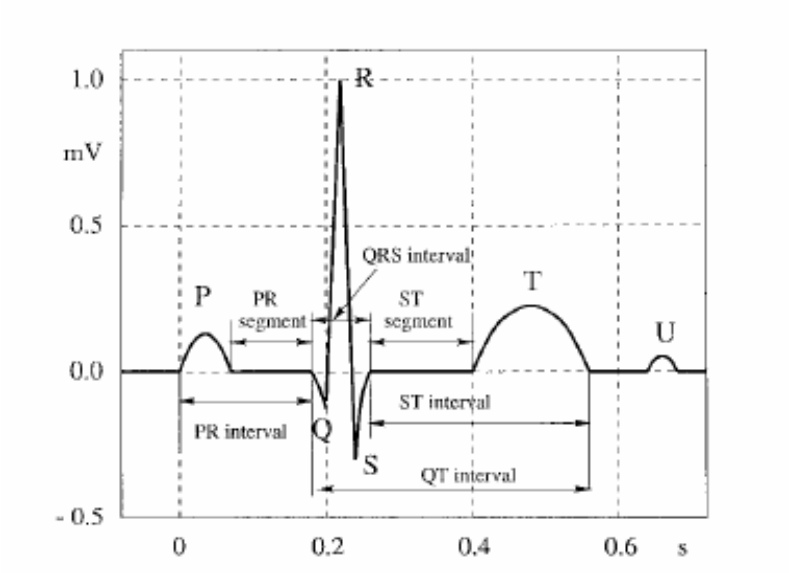

Figure 2.1. Normal ECG signal [3]

P wave: Atrial depolarization produces the P wave on the electrocardiogram. The duration of the P wave should not exceed 0.12 s.

PR interval: The PR interval is the time between the onset of atrial depolarization and the onset of ventricular depolarization and it is measured from the beginning of the P wave to the first deflection of the QRS complex. The normal duration of the PR interval is 0.12 s to 0.20 s.

QRS complex: The QRS complex represents the electrical forces generated by ventricular depolarization. The duration of the QRS complex should not exceed 0.10 s.

ST segment: The ST Segment lies between the QRS complex and the beginning of the T wave, and represents the period between the end of ventricular depolarization and the beginning of repolarization.

T wave: Ventricular repolarization produces the T wave. The normal T wave is asymmetrical, the first halve having a more gradual slope than the second half.

QT interval: The QT interval is measured from the beginning of the QRS complex to the end of the T wave and represents the total time taken for depolarization and repolarization

of the ventricles. The QT interval increases slightly with age and tends to be longer in women than in men.

*U wave***:** The U wave is a small deflection that follows the T wave. Many electrocardiograms have no discernible U waves. Prominent U waves may be found in athletes and are associated with hypokalaemia and hypercalcaemia.

The various parameters electrocardiographics (ECG) are basic in characterization of the forces generated during the cardiac activity. Actually, it is an essential tool for the diagnosis of cardiac abnormalities. The analysis consist of the measure of the amplitudes, various segment the durations and the morphologies of the P, QRS and T waves [16].

2.2.2. Arrhythmia Disorders

An arrhythmia is a common term for any cardiac rhythm, which deviates from normal sinus rhythm. The shape and size of P-QRS-T wave, the time intervals between its various peaks, etc. may contain useful information about the nature of disease afflicting the heart. However, these subtle details cannot be directly monitored by the human observer. Besides, since biosignals are highly subjective, the symptoms may appear at random in the time scale. Therefore, the signal parameters, extracted and analyzed using computers are highly useful in diagnostics.

Automated systems for arrhythmia detection have been using since the early 1960s. These systems mostly use QRS complex and P-R interval to group arrhythmias into ventricular and supra ventricular categories. Then ventricular arrhythmias can be analyzed further. Supra ventricular arrhythmias however need detection of P wave in addition to the QRS complex. The QRS complexes are detected with specialized algorithms and then classified with several techniques [17].

Different kinds of arrhythmia disorders can be summarized as Ischemic Changes (Coronery Artery Disease), Old Arterior Myocardial Infarction, Old Interior Myocardial Infarction, Sinus Tachcardy, Sinus Bradycardy, Ventricular Premature Contraction,

Supraventricular Premature Contraction, Left Bundle Branch Block, Right Bundle Branch Block, Left ventricule hypertrophy, Atrial Fibrillation or Flutter.

2.2.3. UCI Arrhythmia Database

The Arrhythmia Database used in this study is obtained from UCI Repository [9]. The dataset includes 452 patient records which are described by 279 feature values. There are 16 groups of records which class 01 refers to normal ECG, classes 02-15 refers to the different kinds of arrhythmia disorders and class 16 refers to the unclassified records. The detailed class distribution is given in the table.

Table 2.1. Class distribution of arrhythmia database

Class Code	Class	# of Instances
01	Normal	245
02	Ischemic Changes (Coronary Artery Disease)	44
03	Old Anterior Myocardial Infarction	15
04	Old Inferior Myocardial Infarction	15
05	Sinus Tachycardy	13
06	Sinus Bradycardy	25
07	Ventricular Premature Contraction (PVC)	3
08	Supraventricular Premature Contraction	2
09	Left Bundle Branch Block	9
10	Right Bundle Branch Block	50
11	1. degree AtrioVentricular block	0
12	2. degree AV block	0
13	3. degree AV block	0
14	Left ventricule hypertrophy	4
15	Atrial Fibrillation or Flutter	5
16	Others	22

Each record contains clinical measurements, from ECG signals, such as QRS duration, P-R interval and Q-T intervals and some other personal information such as sex, age, weight together with the decision of cardiologist. There are a total of 279 attributes per patient in a record. Diagnosis of the cardiologist is either normal or one of 15 different classes of arrhythmia.

The first 9 features are f_1: Age; f_2: Sex; f_3: Height; f_4: weight; f_5: the average QRS duration in msec.; f_6: the average duration between onset of P and Q waves in msec.; f_7: the average duration between onset of Q and offset of T waves in msec.; f_8: the average duration between two consecutive T waves in msec.; f_9: the average duration between two consecutive P waves in msec. The whole definition of the attributes is given in Appendix A.

About 0.33 % of the feature values in the dataset are missing. Class distribution of this dataset is very unfair and instances of classes 11, 12 and 13 do not exist in the dataset. Class 01 (normal) is the most frequent one.

Although the ECG of some patients shows the characteristics of more than one arrhythmia, it is assumed that no patient has more than one cardiac arrhythmia [5].

Because of the missing attribute values and unclassified records the classification accuracy of any model built using this dataset cannot be perfect, however such characteristics make it more similar to the real-world situations [3].

The study “A Supervised Machine Learning Algorithm for Arrhythmia Analysis” [5] proposes the VFI5 algorithm diagnosis of cardiac arrhythmia from standard 12 lead ECG recordings. (The authors of this paper are the owners of the UCI Arrhythmia data set at the same time.) Classification in VFI5 is based on a majority voting among the class predictions (votes) made by each feature separately. If a feature includes unknown or missing value that feature does not participate in the classification process. Hence the features containing missing values are simply ignored. The VFI5 algorithm achieved an accuracy of 62% on the data set.

Another study “ANN Based Diagnostic System for Arrhythmia with ECG Signals” [3] uses the same UCI arrhythmia dataset and presents a diagnostic system for cardiac arrhythmias from ECG data, using an artificial neural network classifier based on a Bayesian framework. The ECG signals used as system input include five parameters: QRS duration, PR interval, QT interval, T interval and P interval. The personal information available includes age, height, weight and sex. The data set is divided into two groups, labeled as Normal and Abnormal. There are 245 cases in the normal group and 207 cases in abnormal group. As the result of the classification task 75% test accuracy obtained. After applying “dual threshold” method, 91% test accuracy obtained. The capability of uncertainty management with the dual threshold method could be used as a control strategy and suppress false alarm signals [3].

3. METHODOLOGY

According to published results, existing approaches generally tend to suffer from problems those results from high sensitivity to noise included in the data and unreliability in dealing with new or ambiguous patterns. In clinical domains one must to face the problem of developing classifiers that are able to deal with nonlinear discrimination between classes, incomplete or ambiguous input patterns and suppression of false alarm signals.

It is necessary to develop new detection schemes with a high level accuracy to be useful in practical applications. Since Support Vector Machines are nonlinear and can deal with high dimensional data, such techniques are useful in the area of ECG.

3.1. System Overview

The proposed intelligent arrhythmia classification system consists of three main stages. First stage is “Data Acquisition” that consists of Handling Missing Values, Scaling and Dimensionality Reduction of the high dimensional Arrhythmia dataset.

The second part of the system is classification. Although the aim of this study is to apply Support Vector Machine classification for arrhythmia classification, Decision Tree and k-Nearest Neighbor algorithms have been applied in order to compare these techniques with SVM. Finally, the results of classifications have been evaluated both in terms of machine learning and medical biostatistics. Detailed information about system components is given in the next sections.

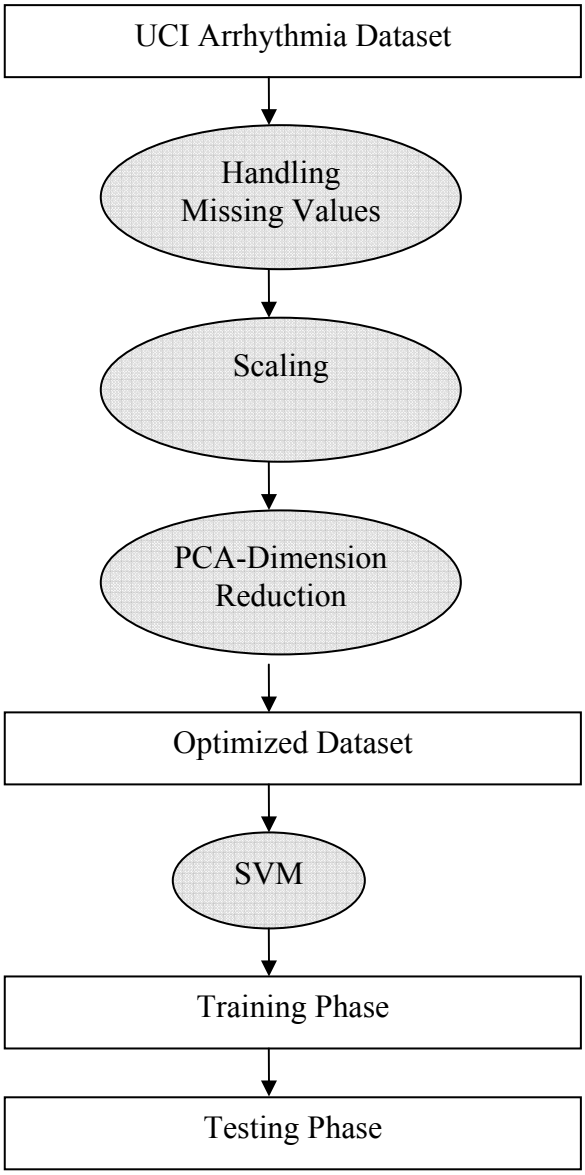

Figure 3.1. Schematic representation of presented system

3.2. Data Acquisition

The dataset used in the study consists of 452 ECG recordings from the UCI Arrhythmia Database. This database stores digitized ECG signal instead of graphical representation. These stored digital data can than be processed to detect various kinds of arrhythmia.

The dataset includes about 0.33% missing attribute values and 22 unclassified instances, so the prediction accuracy of any model built sing it cannot be perfect [3]. However such characteristics make it more similar to the real world dynamic environment. Each record consists of a set of clinical parameters measured on ECG signals and some personal information about the patients.

The dataset includes 16 groups of records which class 01 refers to normal ECG, classes 02-15 refers to the different kinds of arrhythmia disorders and class 16 refers to the unclassified records.

Class distribution of this dataset is very unfair and instances of classes 11, 12 and 13 do not exist in the dataset. Class 01 (normal) is the most frequent one. Because of the unfair class distribution, multi-class classification would make unreliable predictions, so the dataset is divided into two groups labeled as Normal and Abnormal. There are 245 cases in the normal group and 207 cases in the abnormal group. In all of the classification and data preprocessing tasks, this two-class dataset have been used.

3.3. Handling Missing Values

Missing data frequently occur in applied statistical data analysis. There are several reasons why the data may be missing. They may be missing because equipment malfunctioned; observations become incomplete due to people or observations which are not entered correctly.

Three general methods have been mainly used for handling missing values in statistical analysis. One is the "***complete case analysis***" which ignores the observations with missing values and bases the analysis on the complete case data. The disadvantages of this approach are the loss of efficiency due to discarding the incomplete observations and biases in estimates when data are missing in a systematic way. The second approach for handling missing values is the ***imputation method***, which imputes values for the missing covariates and carries out the analysis bias of the complete case analysis, but lead to additional bias in multivariate analysis if the imputation fails to control for all multivariate

relationships. The third approach is to assume some models for the covariates with missing values and then use a maximum likelihood approach to obtain estimates for the models.

A study about handling missing values in Support Vector Machine classifier proposes “an alternative approach where no attempt is made to reconstruct the values which are missing, but only the impact of the missingness on the outcome and the expected risk is modeled explicitly” [18].

3.4. Dimensionality Reduction

Dimension reduction techniques are frequently used as a preprocessing step to machine learning. Dimension reduction is the process of choosing a reduced set of original features and has been an important field of research and development since 1970s and shown very effective in removing irrelevant and redundant features, increasing efficiency in learning tasks, improving learning performance like predictive accuracy [19].

In recent years, data have become increasingly larger both rows (i.e., number of instances) and columns (i.e., number of features) in many areas such as medical applications. This enormity may cause serious problems to many machine learning algorithms with respect to scalability and learning performance. For example, high dimensional data (i.e., datasets with hundreds or thousands of features), can contain high degree of irrelevant and redundant information which may greatly degrade the performance of learning algorithms. Therefore, dimension reduction becomes very necessary for machine learning tasks when facing high dimensional data nowadays [19].

Independent and principal components are well known techniques of pattern recognition and intelligent systems. For Independent Component Analysis (ICA), statistically independent features are selected from the data. With Principal Component Analysis (PCA), maximally variant or diagonalized covariance features are selected. Both ICA and PCA were employed in various applications such as robust speaker verification [20]. Datasets with Gaussian distributions can be represented with principal components while data sets with non-Gaussian, can be decomposed to independent components.

3.4.1. Principal Component Analysis (PCA)

The main objective of principal component analysis is to reduce the dimensionality of the data set and to identify new uncorrelated variables. PCA involves a mathematical procedure that transforms a number of possibly correlated variables into a smaller number of uncorrelated variables called *principal components*. The first principal component accounts for as much of the variability in the data as possible, and each succeeding component accounts for as much of the remaining variability as possible [21].

The mathematical technique used in PCA is called eigen analysis. A solution for the eigenvalues and eigenvectors of a square symmetric matrix with sums of squares and cross products is carried out. The eigenvector associated with the largest eigenvalue has the same direction as the first principal component. The eigenvector associated with the second largest eigenvalue determines the direction of the second principal component. The sum of the eigenvalues equals the trace of the square matrix and the maximum number of eigenvectors equals to the number of rows (or columns) of this matrix [21].

Using an eigenvector rotation, it would be possible to decompose the X matrix into a series of loadings and scores. PCA is typically conducted using covariance matrix or auto scaled data. It is then diagonalized to the eigenvector rotation [21]. Typically, the largest eigenvectors (based on the size of the eigenvalues) are the most important ones.

3.4.2. Independent Component Analysis (ICA)

ICA is a very general-purpose statistical technique in which observed random data are linearly transformed into components that are maximally independent from each other [22]. Simply, ICA is a method for finding underlying factors or components from multi dimensional data. What distinguishes ICA from other methods is that it looks for components that are statistically independent or non-Gaussian.

A first step in many ICA algorithms is to remove any correlations in the data. The value of any one of the components gives no information on the values of other components.

Given a set of observations of random variables $(x_1(t), x_2(t),..., x_n(t))$, where *t* is the time or sample index, assume that they are generated as a linear mixture of independent components:

$$X = AS \tag{3.1}$$

where **A** is some unknown matrix. Independent component analysis now consists of estimating both the matrix and $s_i(t)$, when we only observe the $x_i(t)$.

ICA could be defined alternatively as follows: find a linear transformation given by a matrix **W,** so that the random variables $y_i, i = 1,..., n$ are as independent as possible [23]. After estimating **A** its inverse gives **W**.

Since ICA separates sources by maximizing their non-Gaussianity, perfect Gaussian sources can not be separated. Even when the sources are not independent, ICA finds a space where they are maximally independent.

Applications of ICA can be found in many different areas such as audio processing, biomedical signal processing, image processing, telecommunications and econometrics [22].

3.5. Support Vector Machines (SVM)

SVM are fast emerging as a powerful machine learning tool for pattern recognition, decision-making and classification. SVM can be used to extract valuable information from datasets and construct fast classification algorithms for massive data.

SVMs map data points to a high dimensional feature space where a separating hyperplane can be found. This mapping can be carried on by applying the kernel trick which implicitly transforms the input space into high dimensional feature space. The separating hyperplane is computed by maximizing the distance of the closest patterns, i.e. margin maximization.

SVM is an inductive machine learning technique based on the structural risk minimization which aims at minimizing the true error. SVMs generate black box models which lack the explanation capability on how to reach a decision [10].

SVMs have been applied in many real world problems and in several areas: Pattern Recognition, Regression, Multimedia, Bio-informatics, Artificial Intelligence etc. [6]. Many techniques, such as decision trees, neural networks, genetic algorithms, etc. have been used in these areas; however what distinguishes SVMs is its solid mathematical foundation which is based on the statistical learning theory.

Rather than minimizing the training error (empirical risk), SVMs minimize the structural risk which expresses an upper bound on the generalization error, i.e., the probability of an erroneous classification on unseen examples.

In most classification tasks, SVM generalization performance either matches or is significantly better than competing methods [24]. Also, as mentioned in the experimental results, SVM classifier can deal with high dimensional data. Since medical datasets are large especially in number of attributes, there is a need for such a classification technique. Several studies about applications of SVMs to medical decision support have been listed in recent years.

In this study, it is shown that the results of standard SVM classifier have been improved with applying PCA for dimensionality reduction and the classification results can be controlled according to the estimated probabilities of test instances.

3.5.1. Linear SVMs

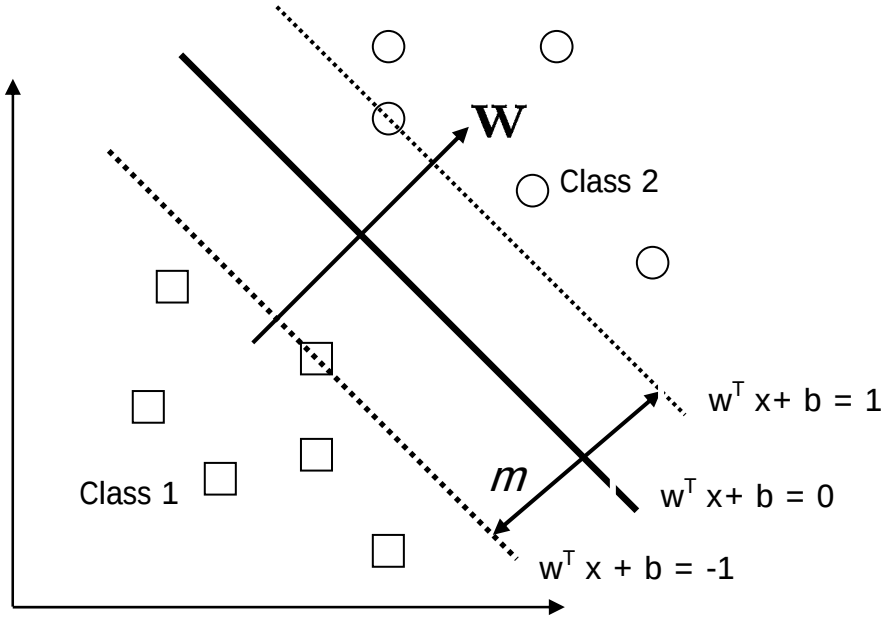

Figure 3.2. Decision boundary for two classes

Let $(x_1, ..., x_n)$ be our data set and $y_i \in (1,-1)$ be the class label of x_i Assuming all data are at least distance 1 from the decision boundary, the following two constraints follow for a training set (x_i, y_i):

$$w^T x_i + b \geq 1 \quad \text{if } y_i = 1 \tag{3.2}$$

$$w^T x_i + b \leq -1 \text{ if } y_i = -1 \tag{3.3}$$

Equations 3.2 and 3.3 can be rewritten as

$$y_i (w^T x_i + b) \geq 1 \tag{3.4}$$

The equation 3.4 implies the instances to be some distance away from the hyperplane for better generalization [25]. The decision boundary should be as far away from the instances of both classes as possible. We should maximize the margin, m, which is the distance between the hyperplane and the instances closest to it.

To maximize the margin we should minimize ||w||. So we can define the problem as

$$\text{Minimize } \frac{1}{2} \| w \|^2 \tag{3.5}$$

$$\text{Subject to } \quad y_i \, (\mathbf{w^T x_i} + b) \geq 1 \quad \forall i \tag{3.6}$$

This is a standard quadratic optimization problem, and the solution involves constructing a dual problem where a Lagrange multiplier α_i is associated with every constraint in the primary problem.

$$\max \ \mathrm{W}(\alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1, j=1}^{n} \alpha_i \alpha_j y_i y_j x_i^T x_j \tag{3.7}$$

$$\text{subject to } \alpha_i \geq 0, \ \sum_{i=1}^{n} \alpha_i y_i = 0 \tag{3.8}$$

w can be recovered as

$$\mathrm{w} = \sum_{i=1}^{n} \alpha_i y_i x_i \tag{3.9}$$

The size of the dual depends on N, sample size, and not on d, the input dimension [25].

The solution suggests that many of the α_i are zero. w is a linear combination of a small number of data. x_i with non-zero α_i are called support vectors (SV). Examples closest to the hyperplane are support vectors and the decision boundary is determined only by the support vectors.

During testing, instead of using margin, we can calculate $g(x) = w^T x + b$ and choose according to the sign of $g(x)$ [25]. Choose C_1 (class 1) if $g(x) > 0$ and C_2 (class 2) otherwise.

3.5.2. Soft Margin SVMs

If the two classes are not linearly separable, we look for the separating hyperplane that incurs least error. We allow error ξ_i "slack variable" in classification which stores the deviation from the margin.

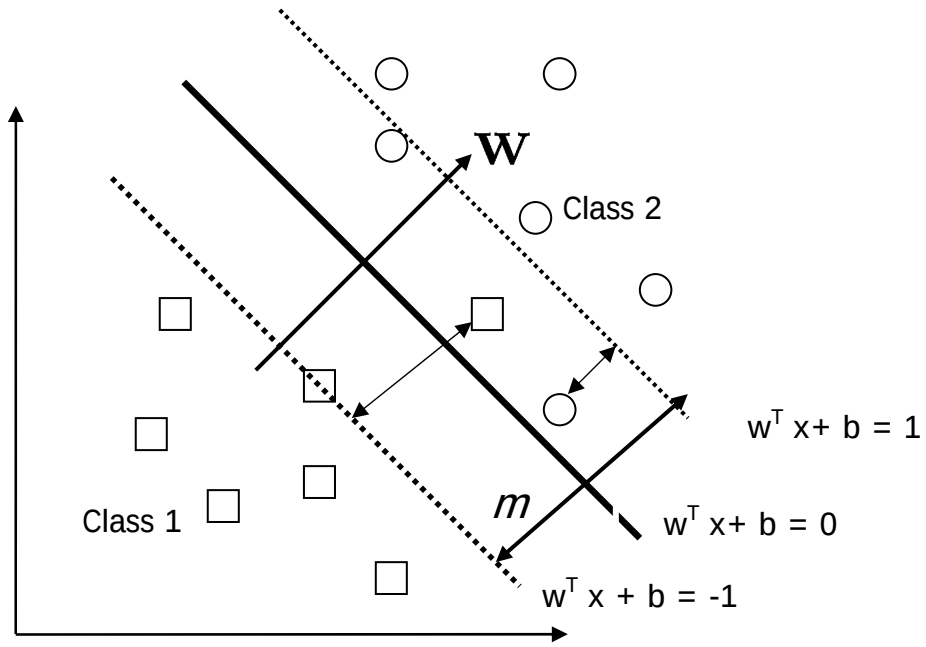

Figure 3.3. Soft-Margin decision boundary for two classes

There are two types of deviation: An instance may lie on the wrong side of the hyperplane and be misclassified. Or, it may be on the right side but may lie in the margin that is not sufficiently away from hyperplane. Then the equation 3 becomes:

$$y_i\ (\mathbf{w^T x_i} + b) \geq 1 - \xi_i \qquad (3.10)$$

If $\xi_i = 0$, there is no problem, If $0 < \xi_i < 1$, x_i is correctly classified but it is in the margin. If $\xi_i > 1$, x_i is misclassified. Now we want to minimize:

$$\frac{1}{2} \| w \|^2 + C \sum_{i=1}^{n} \xi_i \tag{3.11}$$

C is the tradeoff parameter between error and margin can be defined as the penalty factor which is the tradeoff between the complexity (number of support vectors) and the data misclassified [25]. Again the quadratic optimization problem is solved.

3.5.3. Non-Linear SVMs and Kernel Functions

If the two classes are not linearly separable, instead of fitting a nonlinear function, the solution may be mapping the data to a higher dimensional space.

The key idea with the non-linear SVMs is that the original input space can always be mapped to some higher dimensional feature space where the training set is separable. In such a case we are interested in a method whose complexity does not depend on the input dimensionality but depends on the number of training instances [25].

Linear operation in feature space is equivalent to the non-linear operation in original input space. We use soft margin hyperplane because the problem may not be linearly separable in the new feature space. It is critical here to choose a proper C, the penalty factor. If it is too large, we have high penalty for non-separable points and we may store many support vectors and overfit, if it is too small we may have underfit [25].

As the key idea of non-linear SVMs, kernel functions are used for mapping data to a higher dimensional space. The most popular kernel functions are:

Linear: $K(x_i, x_j) = x_i^T x_j$

Polynomial of degree p : $K(x_i, x_j) = (1 + x_i^T x_j)^p$

Radial Basis Function: $$K(x_i, x_j) = \exp\left[-\frac{\|x_i - x_j\|^2}{\sigma^2}\right]$$

Sigmoidal Function: $$K(x_i, x_j) = \tanh(2x_i^T x_j + 1)$$

3.5.4. Multi-Class SVMs

SVM is basically a two class classifier. Multiclass categorization problems are typically solved by reformulating the multiclass problems with M classes into L binary classification problems. More commonly, the dataset is divided into two parts "intelligently" in different ways and a separate SVM is trained for each way of division. Multiclass classification is done by combining the output of all SVM classifiers.

One vs. all: A multiclass approach that constructs a classifier for each class that separates that class from the rest classes of the data.

One vs. one: A multiclass approach that constructs a classifier for each pair of classes that separates those classes. Test data are classified by all the classifiers, and will belong to the class with largest number of positive outputs from pair of classifiers.

3.5.5. Combining SVMs with Various Techniques

SVM have been extensively studied and have shown remarkable success in many applications. However, the success of SVM is very limited when it is applied to the problem of learning from imbalanced datasets in which negative instances heavily outnumber the positive instances [26].

Application areas such as gene profiling, medical diagnosis and credit card fraud detection have highly skewed datasets with a very small number of positive instances which are hard to classify correctly, but important to detect nevertheless[26]. Classifiers generally perform poorly on imbalanced datasets because they are designed to generalize from sample data and output the simplest hypothesis that best fit to data [26].

Making the classifier too specific may make it too sensitive to noise and more prone to learn an erroneous hypothesis. Certain algorithms specifically modify the behavior of existing algorithms to make them more immune to noisy instances, such as the IB3 algorithm for k-NN, or pruning of decision trees, or soft margin SVMs. But, the positive instances can be treated as noises and ignored completely by the classifier [26].

A popular approach towards solving these problems is to bias the classifier so that it pays more attention to the positive instances. This can be done, for instance, by increasing the penalty associated with misclassifying the positive class relative to the negative class. Another approach is to preprocess the data by over sampling the majority class or under sampling the minority class, in order to create a balanced data set [26].

Usually SVMs suffer from a large number of features. Studies about combining SVMs with various feature selection strategies present increased test accuracy [27]. These methods select important features first and then SVM is applied for classification.

A support vector classifier (SVC) provides information about the significance of each feature vector. The feature vectors and the principal and independent component bases are modified to obtain classification error and better generalization than can be obtained by the SVC on the row data and its PCA or ICA subspace representation [28].

3.6. Decision Trees

Decision Trees (DT) are one of the most popular approaches for both classification and regression type predictions. DT are in the form of a tree structure where each node is either a *leaf node* that indicates the value of the target class of examples or a *decision node* that specifies test on a single attribute value with one branch and sub-tree for each possible outcome of the test [29].

A DT is typically constructed recursively in a top-down manner. If a set of labeled instances is sufficiently pure, then the tree is a leaf with the assigned label being that of the most frequently occurring class in that set. Otherwise, a test is constructed and placed into

an internal node that constitutes the tree so far. A branch is created for each block of the partition, and a tree is constructed recursively for each block.

The main problem of the decision tree growing algorithms is selecting which attribute to test at each node in the tree. The concept of ***entropy*** is used for the selection of the attribute with the most inhomogeneous class distribution. Entropy characterizes the impurity of an arbitrary collection of examples. ***Information gain*** uses entropy to measure how well a given attribute separates the training examples according to their target classification [29].

Given a set of *S*, containing two class of examples, the entropy of *S* is defined as

$$Entropy\,(S) = -p_1 \log_2 p_1 - p_2 \log_2 p_2 \tag{3.12}$$

Where p_1 is the proportion of class 1 in *S* and p_2 is the proportion of class 2 (*0log0* is assumed to be 0). The entropy is 0 if all members of *S* belong to the same class; the entropy is 1 if the classes include equal number of examples. In other cases the entropy is between 0 and 1. When generalized to multiclass case:

$$Entropy(S) = \sum_{i=1}^{c} - p_i \log_2 p_i \tag{3.13}$$

where p_i is the proportion of *S* belonging to class *i.*

the information gain *Gain(S,A)* of an attribute *A* is defined as:

$$Gain(S, A) = Entropy(S) - \sum_{v \varepsilon Value(A)} \frac{|S_v|}{|S|} Entropy(S_v) \tag{3.14}$$

Where *Values(A)* is a set of all possible values for attribute *A*, and *Sv* is the subset of *S* for which attribute *A* has value *v*. If the attribute values are continuous, we should to define new discrete-valued attributes that partition the continuous attribute value into a set of discrete intervals.

DT are able to generate understandable rules that it is possible to define each path from the root to a leaf node as a set of IF-THEN rules [25].

3.7. k-Nearest Neighbor Classifier

K-Nearest Neighbor (k-NN) is a distance based classification algorithm and is based on the minimum distance from the test instances to the training samples to determine k-nearest neighbors [30]. Then the class of the test instance is predicted according to majority of these nearest neighbors. In other words, we sort the distances of all training samples to the test instance and determine the k-th minimum distance.

The Nearest Neighbor Algorithm (1-NN) mean the class of a test instance is decided to be the same as the class of its nearest neighbor. k-NN means the class of a test instance is decided to be the same as the class appearing most frequently among k-neighborhood of the test data. By intuition, it seems that it is a reasonable choice to assume that observations which are close together will have the same classification. If the number of training samples is large it would give better results to use the majority vote of the nearest k neighbors instead of the single nearest neighbor. The number k should be large enough to minimize the probability of misclassifying, small enough so that the samples are close enough to give an accurate prediction.

Most generally, the distance metric of k-NN algorithm is Euclidean Distance. Assume $p_1(x_1, y_1)$ and $p_2(x_2, y_2)$ are two points in two-dimensional space, the Euclidean distance $d(p_1, p_2)$ between these two points is:

$$d(p_1, p_2) = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \tag{3.15}$$

If we generalize equation to N dimensional space:

$$d(p, q) = \sqrt{\sum_{i=1}^{N} (p_i - q_i)^2} \tag{3.16}$$

where p_i and q_i are the coordinates in dimension i.

The steps for k-NN classification task can be summarized as follows:

- Determine parameter k: number of neighbors.
- Calculate the distance between test instance and all the training samples
- Sort the distances and determine nearest neighbors based on the k-th minimum distance
- Use simple majority of the class of nearest neighbors as the prediction value of the test instance

k-NN algorithm is simple to implement, robust to noisy training data and effective if the training data are large. However, it needs to determine best value of parameter k. Distance based learning is not clear which type of distance to use and which attributes to use to produce the best results. Computation cost is quite high, because wee need to compute distance of each test sample to all training samples.

k-NN and SVM are both distance based algorithms. As mentioned in the study [31], and can be seen in the experimental results of this study, k-NN algorithm often performs more poorly than SVMs on classification tasks.

4. EXPERIMENTAL RESULTS

The LIBSVM software have been used for SVM classification tasks [32]. "A Practical Guide to Support Vector Classification" [33] proposes the following procedure:

- Transform data to the format of an SVM software
- Conduct simple scaling on the data
- Consider the RBF kernel
- Use cross validation to find the best parameter C and γ
- Use the best parameter C and γ to train the whole training set
- Test

While applying SVM classification to arrhythmia dataset, the proposed procedure has been followed and higher accuracy obtained with scaling and parameter selection.

The *prestd* function of MATLAB software is used for scaling the data. The p*restd* function processes the data set by normalizing the inputs so that they have means of zero and standard deviations of 1.

The main advantage of scaling is to avoid attributes in greater numeric ranges dominate those in smaller numeric ranges. Another advantage is to avoid numerical difficulties during the calculation. Because kernel values usually depend on the inner products of feature vectors large attribute values may cause numerical problems [33].

To measure the classification accuracy and compare different classification methods, 10 fold cross validation technique is used in the experiments that is the whole dataset is partitioned into 10 subsets. The 9 of the subsets is used as the training set and the 10th is used as the test set. This process is repeated 10 times once for each subset being the test set. Classification accuracy is the average of these 10 runs. This technique ensures that the training and the test sets are disjoint

4.1. Handling Missing Values

The UCI Arrhythmia database consists of 0.33% missing attribute values. It is an important issue to handle missing values in machine learning community. Mean Imputation method and Complete Case Analysis are two most popular techniques for handling missing values. In the experiments, we applied both techniques, and classified the complete dataset using SVM classifier. In both cases the same SVM parameters have been used which C = 4 and Gamma = 0.001.

In the Mean Imputation method, the mean of the attribute values have been imputed instead of the missing values and achieved a classification accuracy of **79.35%.**

We achieved an accuracy of **78.26%** when we applied Complete Case Analysis method and ignored the attributes which contain missing values. In such case 11^{th}, 12^{th}, 14^{th} and 15^{th} attributes have been removed, so the remaining data set includes 275 attributes.

According to the results, the complete dataset that has been formed after the mean imputation method and this complete dataset that includes 279 attributes is used in all the classification tasks.

4.2. Parameter Selection

There are four common kernels which are mentioned in Section 3.5.3. The guide [33] suggests that in general the RBF kernel is a reasonable first choice. “The RBF kernel non-linearly maps samples into a higher dimensional space so it can handle the case when the relation between class labels and attributes are non-linear.” [33]. The second reason is mentioned as the number of hyperparameters which influences the complexity of model. The RBF kernel has only one hyperparameter γ (gamma).

While using RBF kernels The SVM has two parameters, C and γ. It is not known which C and γ are the best. So, some kind of parameter search must be done. A simple “grid search” is recommended [33]. Basically pairs of (C, γ) are tried and the one with

best accuracy is picked. It is found that trying exponentially growing sequences of C and γ is a practical method to identify good parameters. (For example $C = 2^{-5}, 2^{-3}, ...2^{15}, \gamma = 2^{-15}, 2^{-13}, ...2^{3}$)

In a previous study, using another SVM tool, we found best parameters as $C = 2^{15}, \gamma = 2^{-7}$ [34]. Again a grid search method applied and the following results obtained.

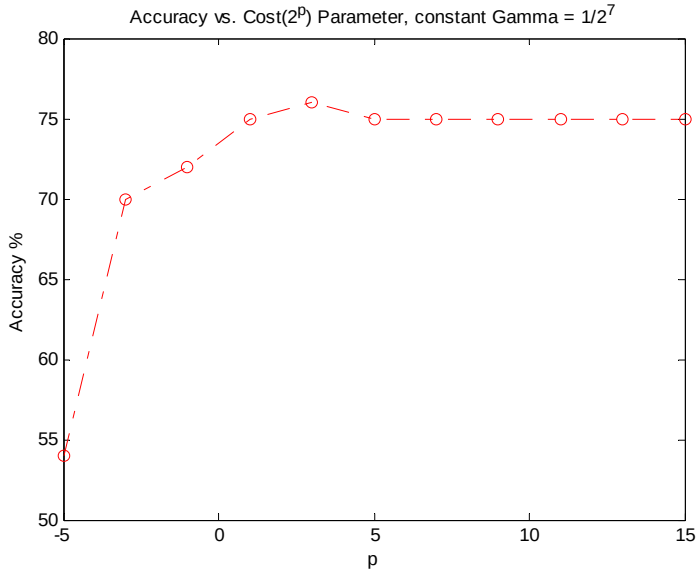

Figure 4.1. SVM test accuracy depending on cost (C) with constant γ

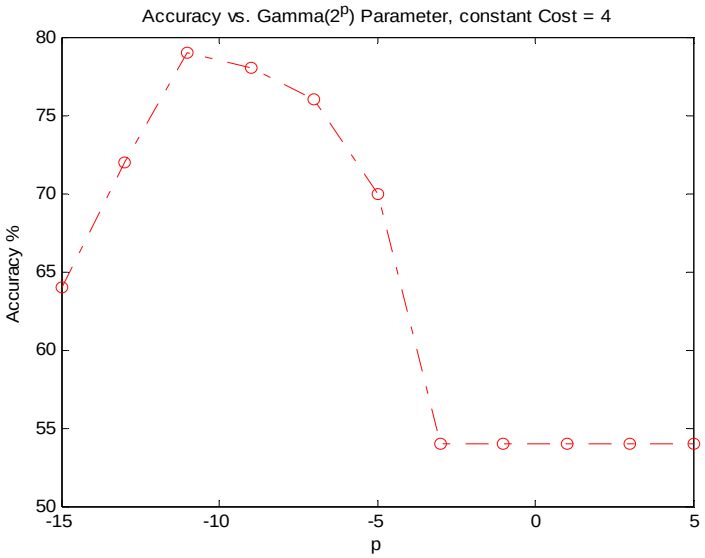

Figure 4.2. SVM test accuracy depending on γ with constant cost (C)

After identifying a better region on the grid, a finer grid search on that region can be conducted. We apply a finer grid search on the neighborhood of $(2^3,2^{-11})$. Finally best C and γ found to be as:

$$C = \mathbf{4} \text{ and } \gamma = \mathbf{0.001}$$

These parameters have been used in the rest of the experiments.

Different kernel types have been applied for the SVM classification. The results are given in the table.

Table 4.1. Test accuracy of SVM classification according to different kernel types

Kernel Type	Test Accuracy
Linear Kernel	0.69
Polynomial of degree 2	0.73
Polynomial of degree 3	0.66
RBF	0.79
Sigmoid	0.74

The RBF kernel gives best results for arrhythmia dataset, so RBF kernel have been used in following experiments.

Scaling and parameter selection is critical for SVM classification. The improved results is given below:

Default SVM classifier parameters:

Kernel Type: Default RBF
Cost : Default 1
Gamma: Default 0

Original set with default parameters: 54.3 %
Scaled set with default parameters: 78.2 %
Scaled set with parameter selection: 79.4 %

Selected parameters: $C = 4$ and $\gamma = 0.001$, RBF kernel.

4.3. Dimensionality Reduction Results

Dimension reduction techniques such as Principal Component Analysis (PCA) and Independent Component Analysis (ICA) are frequently used as a preprocessing step to machine learning. Dimension reduction is the process of choosing a reduced set of original features. These techniques are suggested to increase efficiency and improve performance such predictive accuracy.

Because of UCI arrhythmia dataset is a high dimensional dataset with 279 attribute values, dimensionality reduction is expected to be efficiency. PCA and ICA techniques have been applied and results are given.

4.3.1. PCA Results

Matlab tool is used for applying PCA to arrhythmia dataset. PREPCA preprocesses the network input training set by applying a principal component analysis. This analysis transforms the input data so that the elements of the input vectors will be uncorrelated. In addition, the size of the input vectors may be reduced by retaining only those components which contribute more than a specified fraction (min_frac) of the total variation in the data set.

To measure the effect of PCA method to SVM classification, we have measured test accuracy for different numbers of Principal Components and the case with best accuracy have been chosen to be compared with the complete case SVM classification result. In the test phase, the whole data set is divided into training and test subsets, where the training subset contains 360 ECG recordings and test set contains the remaining 92 instances.

Table 4.2. Results of principal component analysis

min_frac	# of Principal Components	SVM Test Accuracy (%)
0.02	10	0.72
0.015	13	0.74
0.01	20	0.77
0.009	22	0.78
0.008	25	0.78
0.007	28	0.78
0.006	33	0.82
0.005	40	0.80
0.003	63	0.77
0.0025	74	0.77
0.002	87	0.83
0.0015	104	0.83
0.001	128	0.83
0.0009	132	0.83
0.0008	137	0.82
0.00075	140	0.80
0.0007	143	0.84
0.0006	149	0.82

In the table above SVM test accuracy depending on number of principal components is given. Maximum test accuracy obtained with 0.9993 correlation and 143 principal components. As we can see from the Table 4.2, test accuracy increased from 0.79 to 0.84. Although this result is not satisfactory for medical classification applications, PCA provides a significant improvement for SVM classification.

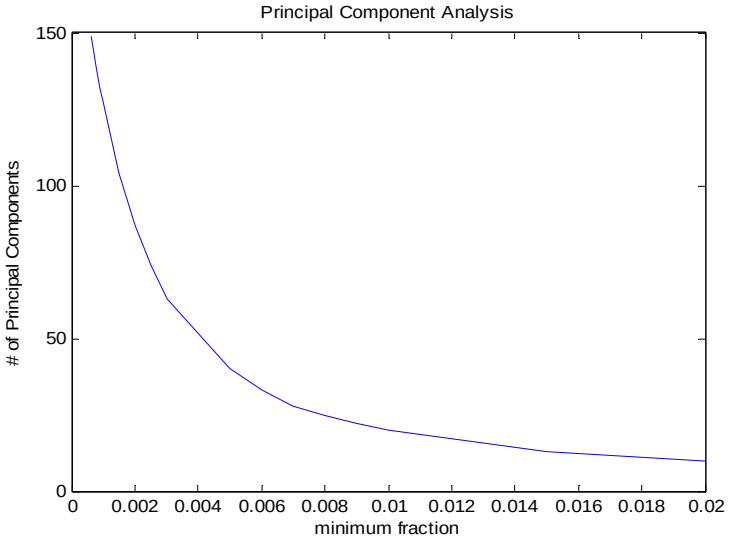

Figure 4.3. Number of principal components depending on minimum fraction

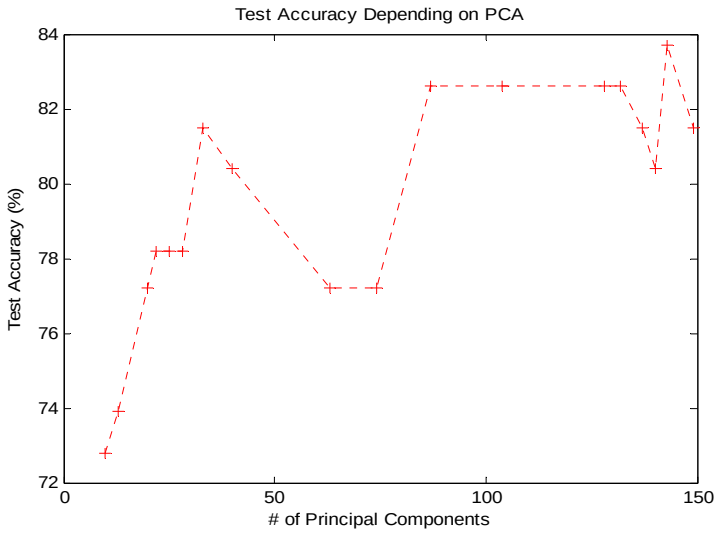

Figure 4.4. SVM test accuracy of principal components

4.3.2. ICA Results

Fast Independent Component Analysis, FASTICA [A12] estimates the independent components from given multidimensional signals; a MATLAB GUI for FASTICA is used in this study to apply ICA technique to arrhythmia dataset. FASTICA uses the fixed-point algorithm developed by Aapo Hyvarinen [35].

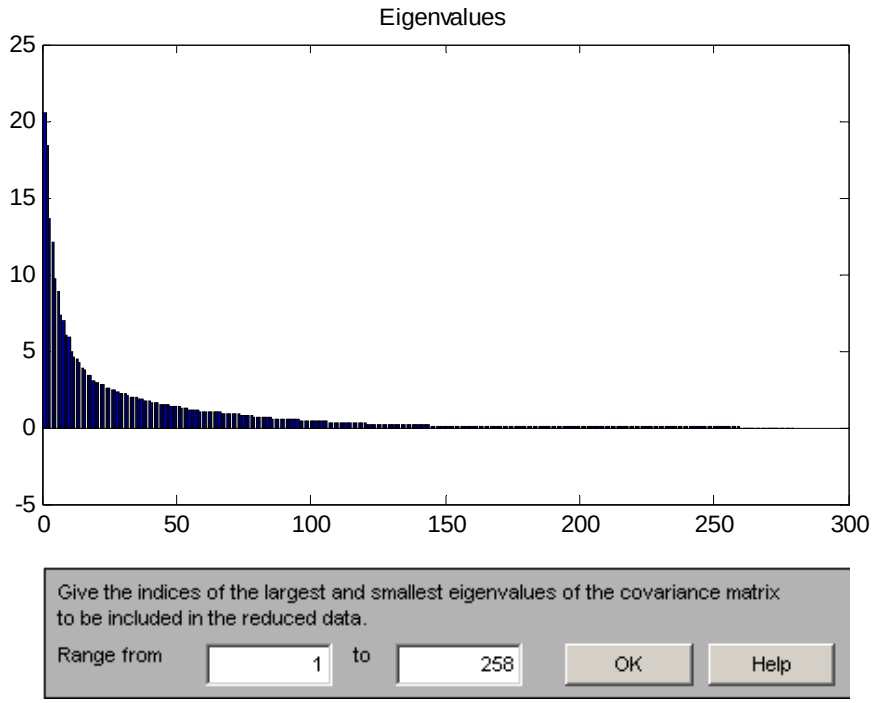

Figure 4.5. Eigenvalue structure of the covariance matrix of the data

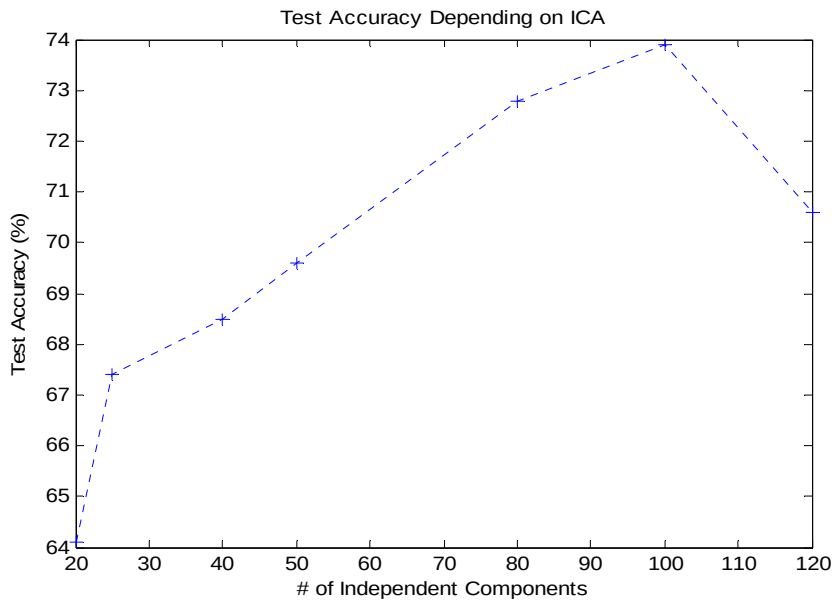

Figure 4.6. SVM test accuracy of independent components

Maximum test accuracy as the result of SVM classification of independent components is 0.74. Consequently PCA found to be much more effective for dimension reduction.

4.4. Classification Results

For supervised learning tasks the classification accuracy of the classifier is one measure of performance. The most commonly used metric for classification accuracy is the percentage of correctly classified test instances over all test instances.

4.4.1. SVM Results

The scaled complete Arrhythmia dataset with 452 recordings and 279 attributes, have been classified using MATLAB Interface of LIBSVM, with best $C = 4$ and $\gamma = 0.001$ parameters. 10-fold-cross-validation technique applied and 77% accuracy obtained. The Table 4.3 shows the test accuracy in each fold.

Table 4.3. 10-Fold-Cross-Validation results for SVM

Fold No	Test Accuracy (%)	# of Correctly Classified Samples
Fold1	0.76	34/45
Fold2	0.67	30/45
Fold3	0.73	33/45
Fold4	0.78	35/45
Fold5	0.82	37/45
Fold6	0.87	39/45
Fold7	0.78	35/45
Fold8	0.67	30/45
Fold9	0.78	35/45
Fold10	0.83	39/47
Average	0.77	347/452

With the same parameters, as a comparison to a previous study [3], 360 instances of the dataset is used as training set, and the remaining 92 is used as test set and an accuracy

of **79.3%** obtained when all the 279 attributes have been used. The table shows the misclassification rate for test phase of the Support Vector Classifier.

Table 4.4. Arrhythmia diagnosis results for SVM

		Real Case	Test Results		
			Normal	Abnormal	FR %
Test Set	Normal	50	45	5	0.10
	Abnormal	42	14	28	0.33
	Total	92	59	33	0.20

If only first 9 attributes used for classification, 76% test accuracy obtained. The first 9 features are:

f_1: Age;
f_2: Sex;
f_3: Height;
f_4: weight;
f_5: the average QRS duration in msec.;
f_6: the average duration between onset of P and Q waves in msec.;
f_7: the average duration between onset of Q and offset of T waves in msec.;
f_8: the average duration between two consecutive T waves in msec.;
f_9: the average duration between two consecutive P waves in msec.

To simplify the data set, we may combine Weight and Height as BMI (Body Mass Index); and combine Age and Sex as Average Age.

$$BMI = \frac{Weight(kg)}{Height(m)^2}$$

$$AverageAge = \frac{Age}{ExpectedAvr.Age(AccordingtoSex)}$$

We wanted to reduce the dimension of input without any information loss. The reduced 7 attributes are:

f_1: Average Age

f_2: Body Mass Index

f_3: the average QRS duration in msec.;

f_4: the average duration between onset of P and Q waves in msec.;

f_5: the average duration between onset of Q and offset of T waves in msec.;

f_6: the average duration between two consecutive T waves in msec.;

f_7: the average duration between two consecutive P waves in msec.

The following table includes the classification results of SVM method, with various numbers of attributes. The results show that SVM gives better results for high dimensional data. This is critical for large medical datasets. Also, the effect of model parameters on the performance of SVM classifier can be seen in the results. The best parameters should be selected in each case.

Table 4.5. SVM results for special cases

# of attributes	# of training instances	# of test instances	C (Cost)	γ	Accuracy (%)
279	360	92	4	0.001	79.3
9	360	92	4	0.001	76
7	360	92	4	0.001	54
7	360	92	2^{15}	2^{-7}	76

4.4.2. Improved SVM with PCA

It is shown that Principal Component Analysis improves the accuracy of SVM classifier. The table shows the test results of SVM, using 143 principal components. An accuracy of 83.7 % obtained as the result of classification with reduced dimensions.

Table 4.6. Arrhythmia diagnosis results for SVM
(After PCA, the dataset includes 143 attributes)

		Real Case	Test Results		
			Normal	Abnormal	FR %
Test Set	Normal	50	46	4	0.08
	Abnormal	42	11	31	0.26
	Total	92	57	35	0.16

4.4.3. False Alarm Suppression

A dual threshold method is proposed to suppress false alarms and to notice uncertain predictions. The uncertainty criterion is introduced for high risk classification outputs lying between upper and lower thresholds. We define a classification response of “uncertain” between probabilities of -0.4 and 0.4 meaning that such cases should be reviewed manually by the supervising physician for a final decision.

When the thresholds are varied, different levels of classification accuracy and uncertainty are obtained. This can be used to control the diagnosis strategy. When suitable thresholds are chosen, an optimum balance between false rate and uncertainty may be found. The dual threshold method presented here could be applied to suppress false alarm signals.

Previously, ANNs have been implemented to perform arrhythmia detection because of its capability to reject unknown or ambiguous patterns. For this purpose, the uncertainty criterion is introduced and evaluated.

In this study, the uncertainty criterion can be defined in terms of "estimated probabilities", which stores the distance to the margin in Support Vector Classifier. The figure 4.7 shows the distribution of the estimated probabilities of test instances. We want to choose the best thresholds that we can classify as much as possible instances with the best accuracy. The dotted lines shows the best region **[-0.4, +0.4]** for false alarm suppression.

In such an alarm case, classifier rejects to classify the patterns whose estimated probabilities are between the Upper Threshold +0.4 and Lower Threshold -0.4. These patterns are assumed to be critical and suggested to be classified manually by a cardiologist (or another classification technique may be applied to the critical patterns, as discussed in the conclusion).

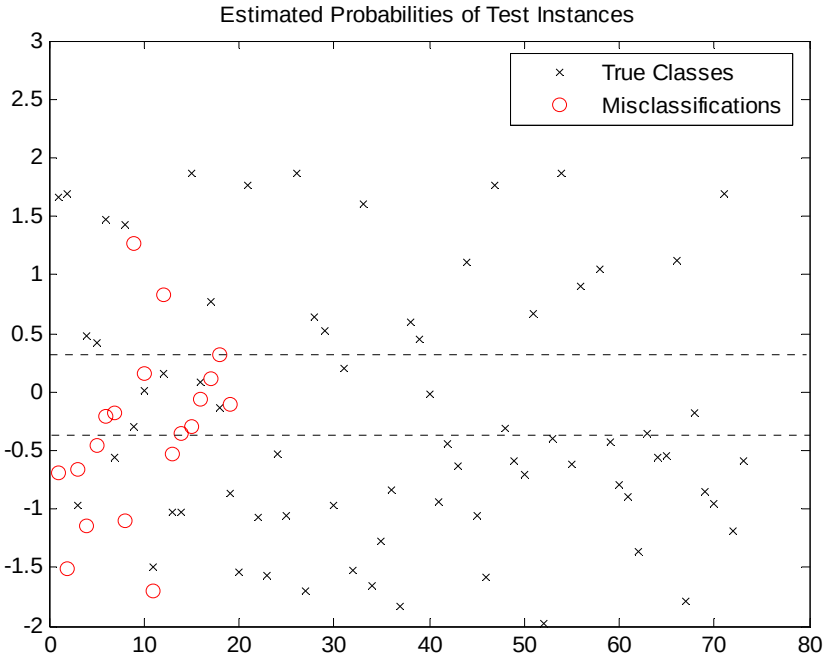

Figure 4.7. Estimated SVM probabilities of test instances

As the result of the False Alarm suppression, the SVM test accuracy increases to 89 %. Finally, using Support Vector Machines Classifier and thresholds for false alarm

suppression, **79 %** of the instances are classified with the accuracy of **89 %.** The table gives the test results with uncertain predictions.

Table 4.7. Arrhythmia diagnosis results for SVM
(After false alarm suppression)

		Real Case	Test Results			
			Normal	Abnormal	Uncertain	FR %
Test Set	Normal	50	39	2	9	0.04
	Abnormal	42	8	24	10	0.19
	Total	92	47	25	19	0.11

4.4.4. Evaluation of Results in Terms of Medical Biostatistics

Clinical research often investigates the statistical relationship between symptoms (or test results) and the presence of disease. Crosscheck can be applied to the data in the form of a 2 by 2 table in order to assess the statistical significance of the association between the presence or absence of a symptom and the presence or absence of the disease under study. When significant associations are found, it is useful to express the data in ways which are clinically relevant. In order to communicate the results in a consistent manner, the following definitions have been developed [36].

Table 4.8. Crosscheck of test results

		Test Results	
		Negative	Positive
Disease	Absent	a	b
	Present	c	d

Sensitivity: The probability that a symptom is present (or screening test is positive) given that the person has the disease d/(c+d). This is also known as the true positive rate.

Specificity: The probability that a symptom is not present (or screening test is negative) given that the person does not have the disease a/(a + b). This is also known as true negative rate.

Predictive value positive (PVP): the probability that a person has the disease given a positive test result d / (b + d).

Predictive value negative (PVN): the probability that a person does not have the disease given a negative test a / (a + c)

False negative: a person who tests as negative but who is actually positive

False positive: a person who tests as positive but who is actually negative

Table 4.9. Crosscheck for SVM results (279 attributes)

		Test Results	
		Negative	Positive
Disease	Absent	45	5
	Present	14	28

Table 4.9 and Table 4.10 give the confusion matrix for SVM classification result before and after dimensionality reduction, respectively. According to these results sensitivity, specificity, PVP and PVN values are given in Table 4.11.

Table 4.10. Crosscheck for SVM results (after PCA)

		Test Results	
		Negative	Positive
Disease	Absent	46	4
	Present	12	30

Table 4.11. SVM results in terms of medical biostatistics

	Sensitivity	Specificity	PVP	PVN
Complete Dataset	0.66	0.90	0.85	0.76
Dataset after PCA	0.71	0.92	0.88	0.79

4.4.5. k-Nearest Neighbor Results

The only hyperparameter of *k*-NN algorithm is *k*, the number of neighbors which are active in decision.

10-fold-cross-validation method applied to the Arrhythmia dataset for separately 279, 9, and 7 attributes and for different *k* values. The maximum average test accuracy is obtained with 7 features. The variation of test accuracy depending on k values is given in the table and figures.

Table 4.12. 10-Fold-CV results for k-NN with different k parameters

	Test Accuracy (%)					
# of attributes	*k = 3*	*k = 5*	*k = 7*	*k = 9*	*k = 11*	*k=13*
279	0.65	0.64	0.65	0.64	0.62	0.61
9	0.65	0.67	0.67	0.68	0.69	0.68
7	0.63	0.66	0.68	0.72	0.71	0.70

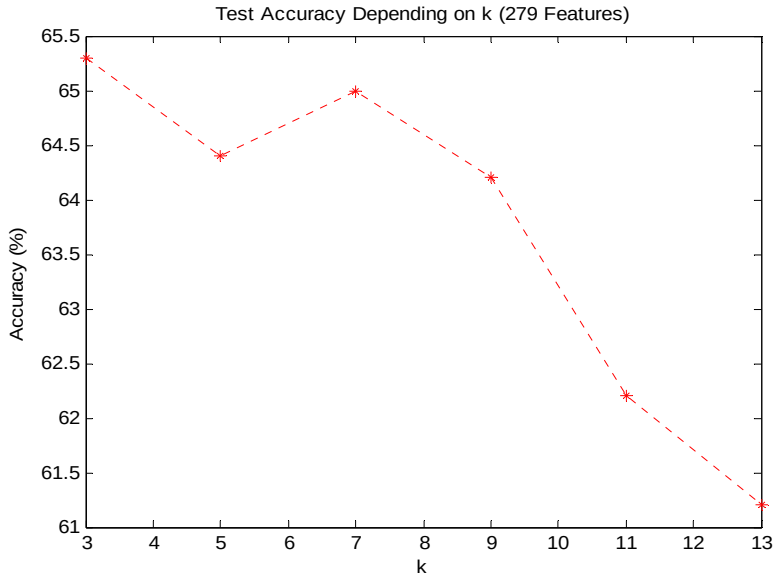

Figure 4.8. k-NN test accuracy depending on k using 279 features

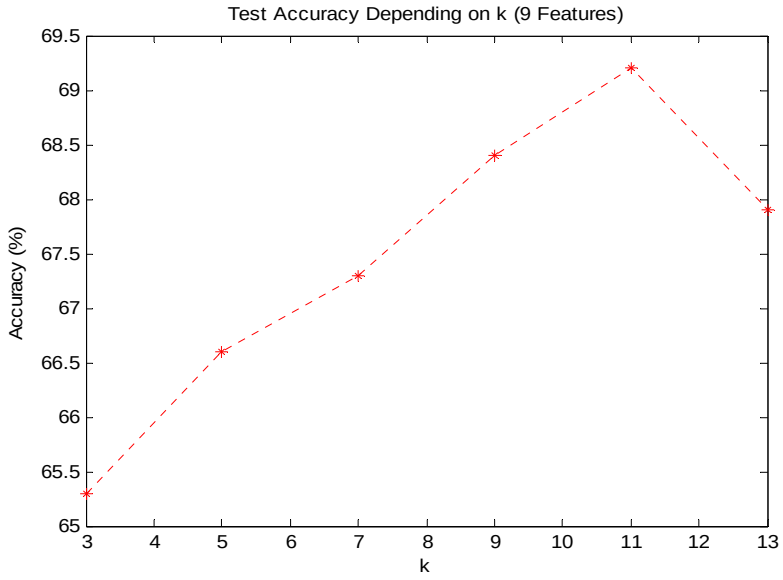

Figure 4.9. k-NN test accuracy depending on k using 9 features

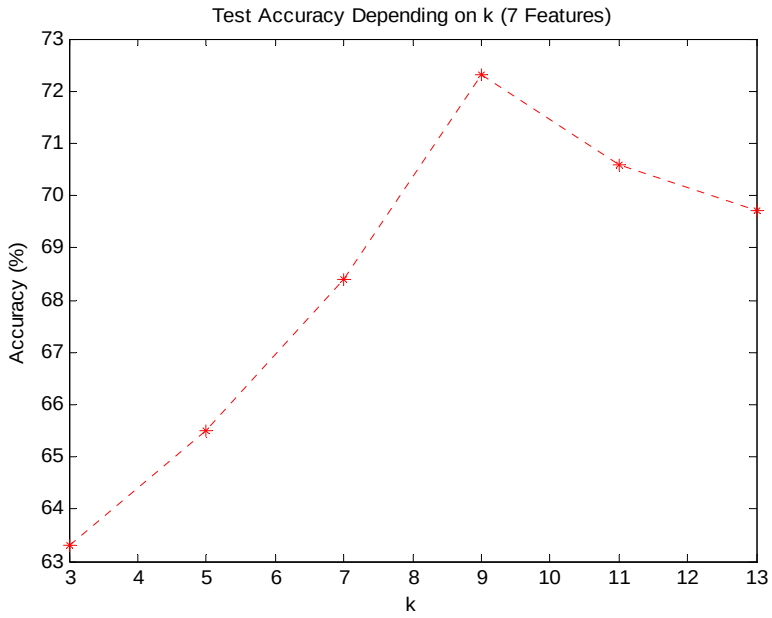

Figure 4.10. k-NN test accuracy depending on k using 7 features

The results show that, in k-NN classification method, smaller k values work better for higher dimensions. Although, both k-NN and SVM are distance based classification techniques, k-NN works better with reduced number of attributes, SVM can deal with large number of attributes. As a result, in such cases, k-NN outperforms SVM. These classifiers can be combined for increasing the test accuracy as a future work.

Table 4.13. k-NN results for special cases

			Test Accuracy (%)					
# of attributes	# of training instances	# of test instances	*k = 3*	*k = 5*	*k = 7*	*k = 9*	*k = 11*	*k=13*
279	360	92	0.74	0.70	0.66	0.66	0.66	0.62
9	360	92	0.74	0.78	0.76	0.76	0.76	0.75
7	360	92	0.73	0.78	0.79	0.82	0.83	0.82

4.4.6. Decision Tree Results

Table 4.14. DT results for special cases

# of attributes	# of training samples	# of test samples	Accuracy (%)
279	360	92	0.62
9	360	92	0.57
7	360	92	0.65

Table 4.14. 10-fold-cross-validation results for DT

Fold No	Test Accuracy (%)	# of Correctly Classified Samples
Fold1	0.62	28/45
Fold2	0.71	32/45
Fold3	0.58	26/45
Fold4	0.53	24/45
Fold5	0.53	24/45
Fold6	0.69	31/45
Fold7	0.56	25/45
Fold8	0.62	28/45
Fold9	0.58	26/45
Fold10	0.77	36/47
Average	0.62	280/452

The average accuracy of 10-fold-cross-validation is **62%**.

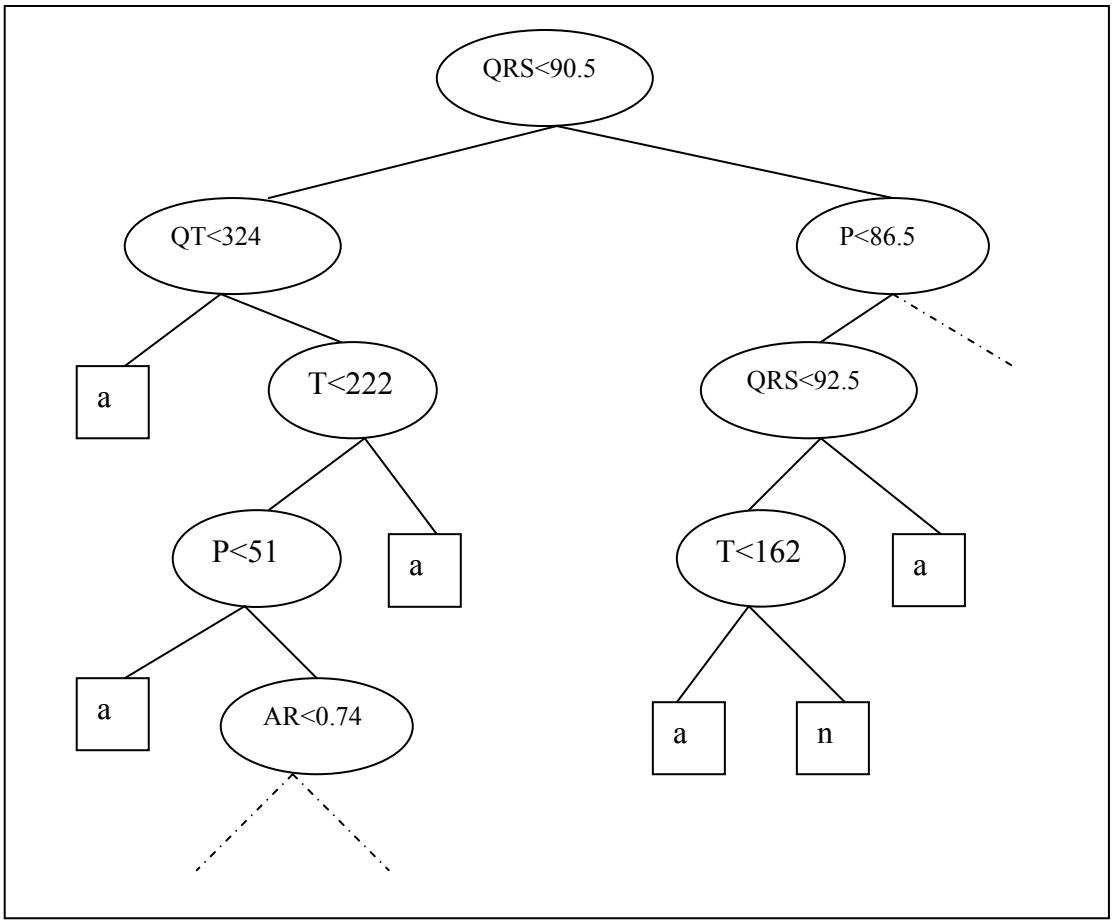

Figure 4.11. Tree representation of DT classification result

QRS: the average QRS duration in msec.

QT: the average duration between onset of Q and offset of T waves in msec.

P: the average duration between two consecutive T waves in msec.

T: the average duration between two consecutive T waves in msec.

AR: Average age

We can extract simple rules from the decision tree results. Class label ‘n’ refers to Normal ECG and ‘a’ refers to Abnormal ECG. The part of the decision tree in Figure 4.11 can be written as:

R1: IF (QRS < 90.5) AND (QT<324)THEN y = abnormal

R2: IF (QRS < 90.5) AND (QT<324) AND (T≥222)THEN y = abnormal

R3: IF (QRS < 90.5) AND (QT<324) AND (T≥222) AND (P<51) THEN

y = abnormal

R4: IF(QRS≥90.4) AND (P<86.5) AND (QRS < 92.5) AND (T≥162) THEN Y = normal

4.5. Comparison of Classifiers

To measure the classification accuracy, 10 fold cross validation technique is used in the experiments that is the whole dataset is partitioned into 10 subsets. The 9 of the subsets is used as the training set and the 10th is used as the test set. This process is repeated 10 times once for each subset being the test set. Classification accuracy is the average of these 10 runs. This technique ensures that the training and the test sets are disjoint

Table 4.16. 10-fold CV results of k-NN, DT and SVM

	Average Test Accuracy (%)		
# of attributes	*kNN (k=3)*	*DT*	*SVM*
279	0.65	0.62	0.77

As a comparison to a previous study [3], 360 instances of the dataset is used as training set, and the remaining 92 is used as test set. The results for ANN [I], k-NN, DT and SVM classifications are given in the table.

Table 4.17. Test accuracy for ANN, k-NN, DT and SVM classification

			Test Accuracy (%)			
# of attributes	# of training instances	# of test instances	*ANN*	*k-NN* (k=5)	*DT*	*SVM*
9	360	92	0.75	0.78	0.57	0.76

Classifiers give different test accuracy for different number of attributes. These accuracy values are given in the table. Best parameters for each classifier have been used in all cases.

Although Nearest Neighbor Classifier works better in lower dimensions, the results show that, Decision Tree and Support Vector Classifiers give better result in higher dimensions.

Table 4.18. Comparison of classifiers depending on number of attributes

			Test Accuracy (%)		
# of attributes	# of training instances	# of test instances	*k-NN*	*DT*	*SVM*
279	360	92	0.74 (k=3)	0.62	0.79
9	360	92	0.78 (k=5)	0.57	0.76
7	360	92	0.83 (k=11)	0.65	0.76

5. CONCLUSIONS

In this study an intelligent arrhythmia classification system based on SVM classifier si presented. The prediction performance is evaluated by measuring both the False Rate and Prediction Accuracy. The proposed system consists of data preprocessing and classification with SVM. First the missing attribute values have been completed. Then dimension reduction techniques have been applied.

Principal Component Analysis and Independent Component Analysis methods have been examined for dimensionality reduction of high dimensional arrhythmia dataset. PCA has shown to outperform ICA according to prediction accuracy of classification of reduced dimensional data subspace.

The results of the standard SVM classifier improved by using PCA and a threshold based rejection method to avoid false predictions for ambiguous instances. The proposed threshold method provides uncertainty management and could be used to control diagnosis strategy and suppressing false alarms.

According to experimental results, parameter selection is very critical for SVM classifier. The performance of the SVM depends on the choice of model parameters. It is shown that by using best parameters, the prediction accuracy of SVM is significantly increased.

As a comparison of classification techniques, k-Nearest Neighbor and Decision Tree algorithms have been tested on the arrhythmia dataset. All of the classifiers have been performed for different number of training and test samples with various parameters. SVM seems to be more consistent.

As a future work, the performance of SVM classifier on different medical datasets would be analyzed to generalize the performance of this technique on medical applications. Another future direction of the study would be combining classifiers. One of the most important issues in machine learning community is to find the optimal architecture of a

classification system. Since it is very difficult to find the best architecture, methods are proposed to combine different architectures. Specifically, in the pattern recognition community, combinations of classifiers are proposed to improve the classification performance of single classifiers.

APPENDIX A: CARDIAC ARRHYTHMIA DATABASE

Title: Cardiac Arrhythmia Database

Original owners of Database:

1. H. Altay Guvenir, PhD., Bilkent University,
Department of Computer Engineering and Information Science,
06533 Ankara, Turkey, Phone: +90 (312) 266 4133
Email: guvenir@cs.bilkent.edu.tr

2. Burak Acar, M.S., Bilkent University, EE Eng. Dept.
06533 Ankara, Turkey
Email: buraka@ee.bilkent.edu.tr

3. Haldun Muderrisoglu, M.D., Ph.D., Baskent University,
School of Medicine, Ankara, Turkey

Donor: H. Altay Guvenir

Date: January, 1998

Past Usage:
H. Altay Guvenir, Burak Acar, Gulsen Demiroz, Ayhan Cekin
"A Supervised Machine Learning Algorithm for Arrhythmia Analysis"
Proceedings of the Computers in Cardiology Conference, Lund, Sweden, 1997.

The aim is to determine the type of arrhythmia from the ECG recordings.

Relevant Information:
This database contains 279 attributes, 206 of which are linear valued and the rest are nominal. Concerning the study of H. Altay Guvenir: "The aim is to distinguish between the presence and absence of cardiac arrhythmia and to classify it in one of the 16 groups. Class

01 refers to 'normal' ECG classes 02 to 15 refers to different classes of arrhythmia and class 16 refers to the rest of unclassified ones. For the time being, there exists a computer program that makes such a classification. However there are differences between the cardiolog's and the programs classification. Taking the cardiolog's as a gold standard we aim to minimise this difference by means of machine learning tools."

The names and id numbers of the patients were recently removed from the database.

Number of Instances: 452
Number of Attributes: 279

Attribute Information:

-- Complete attribute documentation:

1 Age: Age in years , linear
2 Sex: Sex (0 = male; 1 = female) , nominal
3 Height: Height in centimeters , linear
4 Weight: Weight in kilograms , linear
5 QRS duration: Average of QRS duration in msec., linear
6 P-R interval: Average duration between onset of P and Q waves in msec., linear
7 Q-T interval: Average duration between onset of Q and offset of T waves in msec., linear
8 T interval: Average duration of T wave in msec., linear
9 P interval: Average duration of P wave in msec., linear
Vector angles in degrees on front plane of:, linear
10 QRS
11 T
12 P
13 QRST
14 J
15 Heart rate: Number of heart beats per minute ,linear
Of channel DI:
Average width, in msec., of: linear

16 Q wave

17 R wave

18 S wave

19 R' wave, small peak just after R

20 S' wave

21 Number of intrinsic deflections, linear

22 Existence of ragged R wave, nominal

23 Existence of diphasic derivation of R wave, nominal

24 Existence of ragged P wave, nominal

25 Existence of diphasic derivation of P wave, nominal

26 Existence of ragged T wave, nominal

27 Existence of diphasic derivation of T wave, nominal

Of channel DII:

28 .. 39 (similar to 16 .. 27 of channel DI)

Of channels DIII:

40 .. 51

Of channel AVR:

52 .. 63

Of channel AVL:

64 .. 75

Of channel AVF:

76 .. 87

Of channel V1:

88 .. 99

Of channel V2:

100 .. 111

Of channel V3:

112 .. 123

Of channel V4:

124 .. 135

Of channel V5:

136 .. 147

Of channel V6:

148 .. 159

Of channel DI:

Amplitude , * 0.1 milivolt, of

160 JJ wave, linear

161 Q wave, linear

162 R wave, linear

163 S wave, linear

164 R' wave, linear

165 S' wave, linear

166 P wave, linear

167 T wave, linear

168 QRSA , Sum of areas of all segments divided by 10,
(Area= width * height / 2), linear

169 QRSTA = QRSA + 0.5 * width of T wave * 0.1 * height of T wave. (If T is diphasic then the bigger segment is considered), linear

Of channel DII:

170 .. 179

Of channel DIII:

180 .. 189

Of channel AVR:

190 .. 199

Of channel AVL:

200 .. 209

Of channel AVF:

210 .. 219

Of channel V1:

220 .. 229

Of channel V2:

230 .. 239

Of channel V3:
240 .. 249
Of channel V4:
250 .. 259
Of channel V5:
260 .. 269
Of channel V6:
270 .. 279

Missing Attribute Values: Several. Distinguished with '?'.

Class Distribution:
Database: Arrhythmia

Class code :	Class :	Number of instances:
01	Normal	245
02	Ischemic changes (Coronary Artery Disease)	44
03	Old Anterior Myocardial Infarction	15
04	Old Inferior Myocardial Infarction	15
05	Sinus tachycardy	13
06	Sinus bradycardy	25
07	Ventricular Premature Contraction (PVC)	3
08	Supraventricular Premature Contraction	2
09	Left bundle branch block	9
10	Right bundle branch block	50
11	1. degree AtrioVentricular block	0
12	2. degree AV block	0
13	3. degree AV block	0
14	Left ventricule hypertrophy	4
15	Atrial Fibrillation or Flutter	5
16	Others	22

APPENDIX B: DOCUMENT FOR MATLAB INTERFACE OF LIBSVM

Introduction

This tool provides a simple interface to LIBSVM, a library for support vector machines (http://www.csie.ntu.edu.tw/~cjlin/libsvm). It is very easy to use as the usage and the way of specifying parameters is the same as that of LIBSVM.

On Windows systems, pre-built 'svmtrain.dll' and 'svmpredict.dll' are included in this package, so no need to conduct installation.

Usage

matlab> model = svmtrain (training_label_vector, training_instance_matrix, [,'libsvm_options']);

-training_label_vector:
An m by 1 vector of training labels.
-training_instance_matrix:
An m by n matrix of m training instances with n features.
It can be dense or sparse.
-libsvm_options:
A string of training options in the same format as that of LIBSVM.

matlab> [predicted_label, accuracy, decision_values/prob_estimates] = svmpredict(testing_label_vector, testing_instance_matrix, model [,'libsvm_options']);

-testing_label_vector:
An m by 1 vector of prediction labels. If labels of test
data are unknown, simply use any random values.
-testing_instance_matrix:

An m by n matrix of m testing instances with n features.
It can be dense or sparse.

-model:
The output of svmtrain.

-libsvm_options:
A string of testing options in the same format as that of LIBSVM.

Returned Model Structure

The 'svmtrain' function returns a model which can be used for future prediction. It is a structure and is organized as [Parameters, nr_class, totalSV, rho, Label, ProbA, ProbB, nSV, sv_coef, SVs]:

-Parameters: parameters
-nr_class: number of classes; = 2 for regression/one-class svm
-totalSV: total #SV
-rho: -b of the decision function(s) wx+b
-Label: label of each class; empty for regression/one-class SVM
-ProbA: pairwise probability information; empty if -b 0 or in one-class SVM
-ProbB: pairwise probability information; empty if -b 0 or in one-class SVM
-nSV: number of SVs for each class; empty for regression/one-class SVM
-sv_coef: coefficients for SVs in decision functions
-SVs: support vectors

If you do not use the option '-b 1', ProbA and ProbB are empty matrices. If the '-v' option is specified, cross validation is conducted and the returned model is just a scalar: cross-validation accuracy for classification and mean-squared error for regression.

More details about this model can be found in LIBSVM FAQ (http://www.csie.ntu.edu.tw/~cjlin/libsvm/faq.html) and LIBSVM implementation document (http://www.csie.ntu.edu.tw/~cjlin/papers/libsvm.pdf).

Result of Prediction

The function 'svmpredict' has three outputs. The first one, predictd_label, is a vector of predicted labels. The second output, accuracy, is a vector including accuracy (for classification), mean squared error, and squared correlation coefficient (for regression). The third is a matrix containing decision values or probability estimates (if '-b 1' is specified). If k is the number of classes, for decision values, each row includes results of predicting k(k-1/2) binary-class SVMs. For probabilities, each row contains k values indicating the probability that the testing instance is in each class. Note that the order of classes here is the same as 'Label' field in the model structure.

REFERENCES

1. Hussain, W. and W. Ishak, *The Potential of Neural Networks in Medical Applications*, http://www.generation5.org/content/2004/NNAppMed.asp, 2004.

2. Veropoulos, K., N. Cristianini and C. Campbell, "The Applications of Support Vector Machines to Medical Decision Support: A Case Study", *ACAI99,* Chania, 1999.

3. Gao, D., M. Madden, M. Schukat, D. Chambers and G. Lyons, "ANN Based Diagnostic System for Arrhythmia with ECG Signals*", Proc. 23rd IASTED International Multi-Conference Artificial Intelligence and Applications*, Innsbruck, 2005.

4. Acharya, U. R., P. S. Bhat, S. S. Iyengar, A. Rao and S. Dua, "Classification of Heart Rate Data Using Artificial Neural Network and Fuzzy Equivalence Relation", *Pattern Recognition*, Vol. 36, pp. 61-68, 2003.

5. Guvenir, H.A., B. Acar, G. Demiroz and A. Cekin, "A Supervised Machine Learning Algorithm for Arrhythmia Analysis", *In: Proc. Computers in Cardiology Conference*, Lund, Vol. 24, pp. 433-436, 1997.

6. Awad, M. and L. Khan, *Applications and Limitations of Support Vector Machines*, University of Texas at Dallas, 2004.

7. Vargas, F., M. Castro, M. Macarthy and D. Lettnin, "Electrocardiogram Pattern Recognition by Means of MLP Network and PCA: A Case Study on Equal Amount of Input Signal Types", *VII Brazilian Symposium on Neural Networks (SBRN'02)*, p. 200, 2002.

8. Silipo, R., and C. Marchesi, "Artificial Neural Networks for Automatic ECG Analysis", *IEEE Transactions on Signal Processing*, Vol. 46, No. 5, pp. 1417-1425, 1998.

9. Blake, C. and C. Merz, *UCI Repository of Machine Learning Databases,* http://www.ics.uci.edu/~mlearn/MLRepository.html, 1998.

10. Barakat, N., "Rule-Extraction from Support Vector Machines for Medical Diagnosis – Prediction and Explanation", *ITEE Seminar*, The University of Queensland, Australia, 2005.

11. Guler, N. and F. S. Gurgen, "The Effects of Data Properties on Local, Piecewise, Global, Mixture of Experts and Boundary-Optimized Classifiers for Medical Decision Making", *International Symposium of Computer and Internet Sciences* (ISCIS2004), pp. 51-61, 2004.

12. Veropoulos, K., "Report on Workshop Support Vector Machine Theory and Applications", *ACAI99*, Chania, 1999.

13. Jankowski, S., J. Tijink, G. Vumbaca, M. Balsi and G. Karpinski, "Support Vector Machine for the Recognition of Atrial and Ventricular Depolarization in Holter ECG Recordings", *International Journal of Bioelectromagnetism*, Vol. 4, No. 2, pp. 343-344, 2002.

14. Rodriguez, J., A. Goni, L. Dranca and A. Burgos, *MOLEC: Anywhere and at Anytime Arrhythmia Classification*, University of the Basque Country, Donostia, 2006.

15. Morris, F., *ABC of Clinical Electrocardiography,* BMJ Publishing Group, London, 2003.

16. Mokrani, K. and A. Allice, *ECG Parameters Estimation Using Time Scale Analysis,* University of Bejaja Algeria, 2003.

17. Dilmac, O. and F. Gurgen,, "An Arrhythmia Classification Algorithm Using Dynamic Time Warping", *Turkish Symposium on Artificial Intelligence and Neural Networks (TAINN)*, 2003.

18. Pelcksman, K., J. De Brabanter, J. A. K. Suykens and B. De Moor, "Handling Missing Values in Support Vector Machine Classifier", *Neural Networks*, Vol. 18, pp. 684-692, 2005.

19. Yu, L. and H. Liu, "Feature Selection for High Dimensional Data: A Fast Correlation-Based Filter Solution", *20th International Conference on Machine Learning,* Washington, 2003.

20. Senturk, A. and F. S. Gurgen, "The Usage of Independent Component Analysis for Robust Speaker Verification", *Bogazici University, Department of Computer Engineering*, 2006.

21. Vidal, R., Y. Ma and S. Sastry, "Generalized Principal Component Analysis (GPCA)", *IEEE Computer Society Conference,* 2003.

22. Hyvarinen, A., "Independent Component Analysis: Algorithms and Applications", *Neural Networks*, Vol. 13, pp. 411-430, 2000.

23. Hyvarinen, A. and O. Karhunen, *Independent Component Analysis*, John Wiley & Sons, 2001.

24. Burges, C.J.C., "A Tutorial on Support Vector Machines for Pattern Recognition", *Data Mining and Knowledge Discovery*, Vol. 2, pp. 121-167, 1998.

25. Alpaydın, E., *Introduction to Machine Learning*, MIT Press, 2005.

26. Akbani, R., S. Kwek and N. Japkowicz, "Applying Support Vector Machines to Imbalanced Datasets", *XVth European Conference on Machine Learning(ECML04)*, 2004.

27. Chen, Y. W. and C. J. Lin, *Combining SVMs with various Feature Selection Strategies*, Department of Computer Science, National Taiwan University, 2005.

28. Fortuna, J. and D. Capson, "Improved Support Vector Classification Using PCA and ICA Feature Space Modification", *Pattern Recognition*, Vol. 37, pp. 1117-1129, 2004.

29. Hamilton, H., E. Gurak, L. Findlater and W. Olve, *Overview of Decision Trees*, http://www.cs.uregina.ca/~dbd/cs831/notes/ml/dtrees/4_dtrees1.html, 2001.

30. Tenomo, K., *Nearest Neighbor Tutorial*, http://people.reoledu.com/kard./tutorial/KNN, 2001.

31. Vincent, P. and Y. Bengio, *K-Local Hyperplane and Convex Distance Nearest Neighbor Algorithms*, Universite de Montreal, Technical Report 1197, 2001.

32. Chang, C. C. and C. J. Lin, *LIBSVM : A Library For Support Vector Machines*, http://www.csie.ntu.edu.tw/~cjlin/libsvm, 2001.

33. Hsu, C. W., C. C. Chang and C. J. Lin, *A Practical Guide to Support Vector Classification*, Department of Computer Science and Information Engineering, National Taiwan University, Taipei, 2001.

34. Özkaya, A. U., M. E. Kaya, F. Gurgen, "Destek Vektör Makineleri Kullanılarak Akıllı Aritmi Sınıflandırması", *BIYOMUT*, Istanbul, 2005.

35. Gävert, H., J. Hurri, J. Särelä and A. Hyvärinen, *FastICA for Matlab 7.x and 6.x, Version 2.5*, 2005.

36. Willoughby, T. L., *Medical Biostatistics*, http://research.med.umkc.edu/tlwbiostats /sens_specif_predval.html, 1997.

37. Ji, C. and S. Ma, "Combinations of Weak Classifiers", *IEEE Transactions on Neural Networks*, Vol. 8, No. 1, 1997.