A Comparative Study of Multivariate Approach with Neural Networks and Support Vector Machines for Arrhythmia Classification

Sandeep Raj and Kailash Chandra Ray
Department of Electrical Engineering
Indian Institute of Technology Patna
Patna, India 800013
Email: {srp, kcr}@iitp.ac.in

***Abstract*—This paper presents a comparative study of multivariate approach i.e. principal component analysis (PCA) for ECG signal analysis with support vector machine (SVM) and back propagation neural network for classification. Here, the combination of different sets of feature extraction and classification algorithms are analyzed and compared with each other to yield the best performance in terms of accuracy and other performance metrics. The experiment is performed to classify six classes of ECG beats and evaluated using the MIT-BIH database. The results show that the kernel based PCA with support vector machine performs better with an average overall accuracy, sensitivity, specificity and positive predictivity of 98.96%, 98.90%, 99.79% and 98.98% respectively.**

Keywords*—*Arrhythmia beats, R-peak, Multivariate, Backpropagation Neural Network, Support Vector Machines

I. Introduction

Cardiovascular disease is the leading cause of death worldwide [1] amongst all the diseases. According to a recent data, 25% of deaths in the age group of 25-69 yrs occur because of heart diseases in India. Another important factor is that every year 130000 children in India are born with congenital heart disease and can still live a healthy life with timely intervention. Computerized electrocardiography is a well established technique to analyze the status of heart and facilitates for the diagnosis against cardiovascular diseases.

The state of heart of a person is analyzed by an Electrocardiograph (ECG) which represents the graphical recordings of the heart electrical activities. As a result of these electrical activities, P-QRS-T waves are generated which represent one cycle (single beat) and carries significant amount of information about the state of heart. The subtle changes in the amplitude and duration of these waves depict the cardiac abnormality. Since decades, ECG has served as a non-invasive diagnostic clinical tool employed for diagnosis and monitoring of heart abnormalities. The ECG is used prominently in detecting the abnormalities of heart i.e arrhythmias (abnormal ECG beats) which causes occurrence of diseases due to a long term effect. These arrhythmias are not always potentially dangerous but can lead to sudden death without a proper treatment. The arrhythmias are non-stationary in nature [2] i.e the symptoms may not show all the time while patients visiting the hospitals. Even for the same patient the ECG varies depending on their lifestyle and work. Thus, detection and classification of these arrhythmias is a challenge in clinical cardiology which is even faced by the experienced cardiologist. In view of these ambiguities, a computer-aided diagnosis system is required which shall help the physicians in diagnosis of cardiovascular diseases (CVD's).

In recent years, numerous works have been reported in literature for automated classification of these arrhythmias. A few of them includes ECG morphology [3], [4], heartbeat interval techniques [5], frequency based analysis [6], wavelet transform [7] for feature extraction purposes. Moreover, the frequency based analysis does not provide the timing information about the occurrence of frequency components. While in case of wavelet transform, its performance is degraded under noisy condition. In the work, multivariate analysis algorithm i.e PCA and k-PCA algorithms are employed to reduce the complexity of different classifiers to yield best classification accuracy. Principal Component Analysis (PCA) is a mathematical tool to reduce the dimensionality of dataset linearly while Kernel-PCA (KPCA) is an extension to nonlinear form of PCA [8]. K-PCA computes the principal components in a high-dimensional feature space, related to the input space by some nonlinear map to extract nonlinear principal components. Kernel methods are utilized in various different research areas, such as noisy interpolation and pattern recognition [9]–[12].

In this work, multivariate analysis techniques are used for feature extraction with different classifiers i.e backpropagation neural network and support vector machines. The study aims to describe the significance of proper selection of feature extraction and classification algorithms for automated classification of six classes of ECG beats namely Normal, left bundle branch block, right bundle branch block, preventricular contraction, paced and atrial preventricular beats when evaluated on MIT-BIH database [13]. The ECG signals classification comprises of three stages i.e R-peak detection, feature extraction and classification stages.

The rest of this paper is organized as follows: The Section II discusses the different feature extraction and classification algorithms employed. The Section III discusses the proposed methodology while results and performance analysis are discussed in Section IV. Finally Section V concludes the paper.

II. Feature Extraction and Classification Algorithms

This section presents the feature extraction and classification algorithms which are employed to detect the ECG signals. The stages involved for the classification of ECG beats is presented in Fig. 1.

978-1-4678-6503-1/15/$31.00 © 2015 IEEE

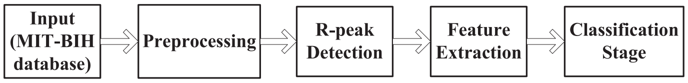

Fig. 1: Stages Involved in ECG beat classification

A. *Principal component Analysis(PCA)*

Principal Component Analysis lies within a branch of statistics known as multivariate analysis. In multivariate analysis, the multiple variables are often represented as single vector variable which includes the different variables. It aims to find transformations of multivariate data which make the data set smaller and easier to understand. Linear transformation is applied to multivariate data to produce new data sets that are more meaningful or can be condensed to fewer variables. The data set is reduced by transforming the original data into new set where some of the new variables have values that are quite small compared to others. Since these variables are small in value they can be eliminated as it does not contribute much information to overall data set. The goal of this approach to data reduction is to find a matrix that will produce such a transformation so that the new variables are uncorrelated.

For a given m-dimensional data set X, the p principal axes $T_1, T_2, \cdots, T_p$, where $1 \leq p \leq m$, are orthonormal axes onto which the retained variance is maximum in the projected space. Generally, $T_1, T_2, \cdots, T_p$ can be given by the p leading eigenvectors of the sample covariance matrix $S = 1/N \Sigma_{i=1}^{N} (x_i - \mu)^T (x_i - \mu)$ where $x_i \epsilon X$, μ is is the sample mean and N is the number of samples, so that:

$$ST_i = \lambda_i T_i, \qquad i \epsilon 1, \cdots, p. \tag{1}$$

where λ_i is the i^{th} largest eigenvalue of S. The p principal components of a given observation vector $x \epsilon X$ are given by

$$[y = y_1, y_2, \cdots, y_p] = [T_1^T x, T_2^T x, \cdots, T_p^T x] = [T^T x] \tag{2}$$

The p principal components of x are decorrelated in the projected space. In multi-class problems, the variations in data are determined on a global basis, i.e the principal axes are derived from a global covariance matrix:

$$\hat{S} = 1/N \Sigma_{j=1}^{K} \Sigma_{i=1}^{N_j} (x_i - \mu)(x_i - \mu)^T \tag{3}$$

where $\hat{\mu}$ is the global mean of all the samples, K is the number of classes, N_j is the number of samples in class j; $N = \Sigma_j^K N_j$ ve x_i represents the ith observation from class j. The principal axes $T_1, T_2, \cdots, T_p$ are therefore the p leading eigenvectors of $\widehat{S}$:

$$\widehat{ST}_i = \widehat{\lambda}_i T_i, \qquad i \epsilon 1, \cdots, p. \tag{4}$$

where $\widehat{\lambda}_i$ is the $i_t h$ largest eigenvalue of $\widehat{S}$. An assumption made for feature extraction and dimensionality reduction by PCA is that most information of the observation vectors is contained in the subspace spanned by the first p principal axes, where $p < m$. Therefore, each original data vector can be represented by its principal component vector with dimensionality p.

In signal processing, PCA is employed to time samples of a input data and if the signal is repeating like ECG, the signal is broken down into segments and presented in the statistical domain. Initially, PCA consists of computation of data covariance matrix. Then, the covariance matrix is decomposed using eigen value decomposition to obtain eigen values and eigen vectors. The eigen vectors are sorted in the descending order of eigen values. Finally, the original data is projected in the direction of these sorted eigen vectors. The first few components include the highest variability present in the data and rest will contain less. Hence, only first few components are chosen such that are mutually uncorrelated.

B. *Kernel based PCA*

Since PCA is a linear algorithm, it unable to extract the non-linear structure in data. It is here where kernel-PCA comes into existence. The kernel-PCA is performed for data x_1, x_2, $\cdots$, $x_n \in \Re^n$ is mapped into a feature space $\mathcal{F}$ and the covariance matrix is computed.

K-PCA algorithm:

Step 1: The gram matrix is computed as:

$K_{ij} = k(x_i, x_j)$, $i, j = 1, 2, \cdots, N$.

Step 2: Eigenvalue and Eigenvector of K is computed as:

$(\alpha^l, \lambda_l), l = 1, 2, \cdots, N.$

Step 3: Eigen vectors are normalized as: $\alpha^l \longleftarrow \frac{\alpha^l}{\lambda_l}$

Thus, an eigen vector w^l of C is now represented as:

$w^l = \Sigma_{k=1}^{N} . \alpha^l . \phi(x_k)$

Then,

$\phi(x)^T . w^l = \phi(x)^T (\Sigma_{k=1}^{N} . \alpha_k^l . \phi(x_k)) = \Sigma_{k=1}^{N} . \alpha_k^l . k(x_k, x).$

C. *Artificial Neural Network(ANN)*

A Multi-layer Perceptron (MLP) network is employed and trained using backpropagation algorithm. The backpropagtion algorithm is a non-parametric technique which performs estimation and classification tasks and belongs to the class of supervised networks [14]. It is designed using three layers i.e input layer, hidden layer and output layers. Neurons in the input layers acts as buffers for distributing the input signals x_i to the neurons in the hidden layers. Each neuron j in the hidden layer sums upon its input signals x_i after weighting them with respective connections w_{ji} from the input layer. Then it computes its output y_i by passing the sum through a non-linear function represented as $y_i = f(\sum_i w_{ji} x_i)$ where f is non-linear function. The backpropagation algorithm is a gradient descent algorithm. The BPNN algorithm is summarized in the following steps:

Step 1. *Declaration*: To start the training, the values of all weights and biases are initialized to a minute real random value.

Step 2. *Setting input and target*: The input layer is given the input vectors $x(1), x(2), x(3)....x(N)$ and the desired output vectors as $t(1), t(2), ...t(N)$ one pair at a time, where N is the number of training patterns.

Step 3. *Calculation of output at neurons*: The equation which is used to calculate the output at each neuron is

$$z_i = \sigma(\sum_{j=1}^{N_{M-1}} w_{ij}^{(M-1)} x_j^{(M-1)} + \theta_i^{(M-1)}),\ i = 1, ..., N_{M-1} \quad (5)$$

Step 4. *Updation of weights(w_{ij}) and biases(b_i)*:

$$\Delta w_{ij}^{(l-1)}(n) = \mu.x_j(n).\delta_i^{(l-1)}(n) \quad (6)$$

$$\Delta\theta_i^{(l-1)}(n) = \mu.\delta_i^{(l-1)}(n) \quad (7)$$

where

$$\delta_i^{(l-1)}(n) = \begin{cases} \sigma'(O_i^{(l-1)})[t_i - z_i(n)], & l = M \\ \sigma'(O_i^{(l-1)}) \sum_k w_{ki}.\delta_k^{(l)}(n), & 1 <= l <= M \end{cases} \quad (8)$$

Here, $x_j(n)$ = output of node j at iteration n, l is layer, k is the number of nodes of output network, M is output layer, and σ is the activation function. The learning rate is represented by μ .

After the training is complete, the weights are fixed and can be further used for testing.

D. Support Vector Machine(SVM)

In machine learning, support vector machine is supervised learning model associated with learning algorithms which analyze the input data and recognize patterns used for classification and regression analysis. A support vector machine constructs a hyperplane or set of hyperplanes in a high- or infinite-dimensional space [15], which can be used for classification, regression, or other tasks (which assumes that data must be linearly separable). Moreover, a good separation is achieved by the hyperplane that has the largest distance to the training support vectors of a particular class (so-called functional margin), since in general the larger the margin the lower the generalization error of the classifier. In SVM analyses, the classes are assumed as $\{-1, +1\}$. The construction of a hyperplane is given by the equation $w^T x + b = 0$ (w is the vector of hyperplane coefficients, b is a bias term) so that the margin between the hyperplane and the nearest data points (i.e support vectors) can be maximized and is posed as the quadratic optimization problem. The optimal hyperplane defined by w and b is the one that minimizes a cost function, which is combination of two criteria: margin maximization and error minimization and represented as $\phi(w, \xi) = \frac{1}{2}||w||^2 + c\sum_{i=1}^{1} \xi_i$ where C is a given value determining the trade-off between minimizing training errors and model complexity term $||w||^2$. The above minimization problem can be posed as constrained minimization quadratic programming (QP) problem. The solution gives rise to the decision function of the form $f(x) = sgn[\sum_{i=1}^{1} y_i\alpha_i(x.x_i)b]$. This quadratic complexity is feasible for low-dimensional data which results in quadratic increase in memory usage and time. Initially, support vector machines were designed for binary classification [16] but later it extended to multiclass classification [17].

Since originally SVM's classify the data in linear case, in the non-linear case SVM's do not achieve the classification tasks. To overcome this limitation, proper kernel function is chosen to train an SVM and that choice is completely data dependent and empirical. Usually two approaches are generally used to implement multi-class SVM. They are: one-against-one and one-against-all. One-against-all approach was the earliest implementation of multi-class SVM which constructs 'm' binary SVM models where m is the number of class. This approach trains the i^{th} SVM by considering all the data sets of i^th class as one space and the other data sets as another space. The one-against-one (OAO) is a better approach for implementing multi-class SVM. This method constructs $k(k-1)/2$ classifiers where each one of them is trained on data from two classes. Literature shows that OAO is superior and achieves better accuracy for multi-class classification [18], [19] which is an obvious choice to use in this study.

III. Steps for Experiments

A. ECG Signal Database

In this experiment, 25 records are selected from the MIT-BIH arrhythmia database [13] for feature extraction and classification purpose. The signals recorded with ML-II (Mason-Likar leads) system at a sampling frequency of 360 Hz and a length of about 30 minutes are used in the study. The sample values with an annotation fixed for every beat is used in this study to classify the ECG signals into one of the six categories : normal (N), left bundle branch block (LBBB), right bundle branch block (RBBB), premature ventricular contraction (PVC) , paced beat (PB), atrial premature beat (APB). The beats taken from the corresponding files of the database are summarized in Table II.

TABLE I. ECG Beats Taken from MIT-BIH Database

Beat type	Record number	Training (Beats)	Testing (Beats)
N	105, 113, 115, 122, 202, 210, 220	4000	4000
LBBB	109, 111, 207, 214	2500	2500
RBBB	118, 124, 212, 231	3000	3000
PVC	119, 221, 200, 208, 233	2000	2000
PB	107, 217	700	700
APB	209, 222, 232	3000	3000
Total		15200	15200

B. Preprocessing and R-peak detection

A R-peak detection algorithm described in [20] is employed in this study. The Pan-Tompkins [20] is implemented due to its proven sensitivity of 99.69% and positive prediction of 99.77% when evaluated with the MIT-BIH arrhythmia database [13]. The algorithm is based on morphological properties of ECG signals on which the preprocessing and decision stage is applied. The recorded ECG signals taken from the database are contaminated with various kinds of noise during their acquisition, so preprocessing is required in order to remove them and use them for analysis. The preprocessing stage performs linear and non-linear filtering of the ECG signal and produces a set of periodic vectors that describe the events. In the linear filtering stage, a cascaded low-pass and high-pass filters is employed. This digital bandpass filter improves the SNR and allows the use of lower thresholds than would be possible on the unfiltered ECG. Secondly, the filtered ECG is passed through a five-point derivative to obtain the slope of R wave. The squaring stage identifies the slope of the frequency response curve of the derivative and restricts false positives caused by T waves with higher spectral energy. Thenafter, a Moving Window Intergrator (MWI) is applied to obtain waveform feature information in addition to the slope of the R wave. The output of MWI stage determines the size of QRS complex with the varying nature of QRS complexes. This approach increases the overall detection sensitivity and reduces the number of false positives. After the detection of R-peak, a moving window of -300ms to 400ms is taken to represent each ECG beat.

C. *Feature Extraction Using PCA and k-PCA*

The features contain significant amount of information about the behavior and characteristics of the input data. Thus, a proper choice of feature vector is essential and integral part of the classification system which directly affects the accuracy and complexity of the classifier. For this purpose, PCA and k-PCA is employed as a feature extraction tool. PCA and kernel based PCA allows the decomposition of an signal into a number of principal components (PC's). Each PC represent the inherent and actual information of the signal analyzed. Since the number of PC's of each ECG signal is different, 20 PC's are thus extracted as features to represent each class of heartbeat considered. Similarly, features are also extracted for other beats which is then given as input to the classifier for further classification. While, in case of kernel PCA, a polynomial kernel of degree 12 over sigmoid kernel based on trail and error is selected to yield better performance. Gaussian kernel based PCA provides an accuracy measure close to that of linear PCA in this case; hence it is not discussed here. All these components are concatenated into a single feature vector to represent the number of beats considered i.e original beat of vector dimension of 252 is compressed into 20 and hence reduces the computation complexity of the classifier.

D. *Classification Using Multiclass SVM's*

The purpose of training SVM is either to minimize the generalization error or to maximize the classification accuracy. A widely used statistical method called m-fold cross-validation is used for this purpose. The training and testing datasets are divided randomly to estimate the generalization error or overall accuracy. In this work, 1/2 fold cross-validation has been implemented on the whole dataset (including training and testing). As discussed in the earlier section, the one-against-one approach is employed to implement multi-class SVM classification. Since SVM is implemented using the quadratic programming approach, thus it's optimization for better performance is necessary for which a maximum number of 1000 iterations at a time are employed. Each beat is matched to one of the input classes with all possible combinations of one-against-one classes and the same number of binary SVM models are initialized. Every pair of classes is considered one-against-one and the corresponding binary SVM model is trained only using the training instances belonging to that particular pair of classes. Simultaneously, the partitioned testing data set is input to the binary SVM classifier trained above to predict it's belonging to one of the six possible classes. Here, SVM classifier has been trained with the linear, polynomial, radial basis function, multi-layer perceptron (mlp) kernels functions. The performance of each kernel function is evaluated to find the relatively high overall performance in terms of accuracy. The results yielded with the multiclass-SVM based on radial basis function are compared and found better than those achieved by the linear, polynomial and mlp kernels which is chosen on the basis of empirical studies for both the feature extraction techniques. Once the training and testing of each binary SVM model is done, vote counting for classified testing data set is run. For each feature of testing data set, the class which they belong to is found by counting the maximum number of votes casted to each class in the classification process. Once the testing of each class of dataset is performed, the results are presented in the form of confusion matrix.

E. *Classification Using BPNN*

The backpropagation algorithm is performed for 20-dimensional input features representing each beat in the form of features yielded as a result of PCA decomposition and mapped into six outputs for each features sets. The output of a particular class (e.g normal, Right Bundled Block (RBBB), Left Bundled Block (LBBB), Preventricular Contraction (PVC), Atrial Premature Contraction (APC), Paced Beat) is represented by the binary coded desired class i.e 1 in the target class while 0 for the rest class of beats. The MLP is trained several times using different number of hidden neurons until the best accuracy is achieved. The number of hidden neurons are set to 19 and 11 in case of k-PCA. However, there is no significant increase in the performance is observed for higher number of hidden neurons in both cases. In the hidden layer 'tansig' function is used as an activation function. 'Trainlm' is a network training function that updates weight and bias values according to Levenberg-Marquardt optimization and is used for training the network. The value of learning rate parameter in training of BPNN is varied from 0.3 to 1.0 and selected 0.6 for PCA and 0.8 for k-PCA based on best accuracy obtained from the empirical studies.

The networks are trained with a total of 15200 beats from the corresponding files of the database as mentioned in Table II. Once the classifier is trained, the testing data set is fed into the classifier so as to formulate the classification performance in terms of accuracy. The MLP network is tested with the same data which is used to the SVM classifier.

F. *Performance Evaluation*

The performance of the different approach is estimated in terms of performance metrics i.e Sensitivity (S_e), Specificity (S_p) and Positive Predicitvity (P_p). Sensitivity refers to the correctly detected events among total no. of events. Specificity refers to the rate of correctly classified non-events whereas positive predictivity refers to the rate of correctly classified events in all detected events. Using these definitions, sensitivity, specificity and positive predictivity can be defined as:

$$Sensitivity(S_e) = \frac{TP}{TP + FN} \times 100 \tag{9a}$$

$$Specificity(S_p) = \frac{TN}{TN + FP} \times 100 \tag{9b}$$

$$PositivePredictivity(P_p) = \frac{TP}{TP + FP} \times 100 \tag{9c}$$

The overall accuracy of the classifier is defined as the ratio of total number of correctly classified events over the total number of events. The performance metrics of each class of heartbeats is calculated on the basis of confusion matrix and discussed in the next section.

IV. Simulation Results and Discussion

The experiments are performed on a total of 30400 ECG beats taken from the MIT-BIH database to classify them into six class of heart beats. All the parameters used for performing various experiments are computed using MATLAB software [R2012a; Version 7.14.0.739 installed on Windows 7 platform, 3.33GHz, i5 CPU] package. It is worth to mention that beats of a particular class has been tested while rest are ignored for estimating the efficiency of proposed algorithm. The classification of each class of heartbeats is presented in the form of confusion matrix. This matrix maps the classification of classified and misclassified heartbeats into subsequent classes.

TABLE II. Accuracy with Different Approaches

Different Approach	Network	Accuracy (%)
PCA + BPNN	20-19-6 (Trainlm)	91.29
k-PCA + BPNN	20-21-6 (Trainlm)	94.58
PCA + SVM	rbf kernel ($C = 0.9$ and $\sigma = 1.0$)	96.27
k-PCA + SVM	**rbf kernel ($C = 0.7$ and $\sigma = 1.0$)**	**98.86**

Various experiments are carried out to yield better accuracy for the classification of heartbeats with different sets of algorithms. The performance of different sets of algorithms employed for classification in terms of accuracy are summarized in Table II. From Table II it can be concluded that k-PCA with SVM outperforms the rest of different sets of algorithms employed for classification and thus presented in detail.

TABLE III. A Summary of SVM implemented with different kernel functions with k-PCA

Kernels Functions	Accuracy (%)	Time (s)	No. of iterations
Linear	92.46	44.06	1000
Polynomial	95.79	47.18	1000
MLP	97.87	51.58	1000
Radial Basis Function	98.96	58.02	1000

Table III shows the performance reported for multi-class support vector machines with different kernel functions in case of k-PCA-SVM alongwith the timing parameters. The classification results of k-PCA are presented in Table IV for detailed study and consequently the performance metrics are calculated which is presented in Table V.

The confusion matrix in Table IV validates the performance of k-PCA with support vector machines by classifying the beats into subsequent classes. Each row of the matrix in Table IV represents the number of beats in the actual class taken from

TABLE IV. Confusion Matrix of k-PCA with SVM approach

Correctly Classified Instances			**15042**	**Accuracy**		**98.96%**	
Incorrect Classified Instances			**158**	**Error Rate**		**1.04%**	
Pred./Act.	**Normal**	**LBBB**	**RBBB**	**PVC**	**Paced**	**APB**	**Total**
Normal	3971	6	11	2	9	1	4000
LBBB	9	2484	2	4	1	0	2500
RBBB	13	4	2963	5	9	6	3000
PVC	3	9	5	1977	4	2	2000
APB	17	3	9	14	2957	0	3000
Paced	5	0	0	3	2	690	700
Total	4018	2506	2990	2005	2982	699	15200

MIT-BIH database i.e for example, In table IV a total 4000 normal beats (first row) taken for testing, out of which 3971 are correctly classified into normal whereas 6 into LBBB, 11 into RBBB, 2 in PVC, 9 in Paced and 1 in APB are misclassified. Similarly, each column represents the number of beats in the predicted class i.e for example, in table IV normal class in first column, 3971 are correctly classified into normal whereas 9, 13, 3, 17,5 beats of LBBB, RBBB, PVC, Paced and APB class of beats are misclassified into normal class. Thus, total number of 4018 beats out of 15200 beats are predicted in normal class by k-PCA approach.

TABLE V. Performance Analysis of k-PCA-SVM approach

Class of beats	TP	TN	FP	FN	S_e	S_p	P_P
Normal	3971	11153	47	29	99.27	99.58	98.83
LBBB	2484	12678	22	16	99.36	99.83	99.27
RBBB	2963	12173	27	37	98.77	99.78	99.10
PVC	1977	13172	28	23	98.85	99.79	98.60
APB	2957	12175	25	43	98.57	99.80	99.16
Paced	690	14491	9	10	98.57	99.94	98.71

The performance metrics of a particular class of beat is computed on the basis of Confusion Matrix presented in Table IV and Eq. (9) and presented in Table V. The k-PCA with SVM approach yields an overall sensitivity, specificity and positive predictivity of 98.90%, 99.79% and 98.98% respectively.

A comprehensive summary for automated classification of ECG beats with other research works using MIT-BIH arrhythmia database is presented in Table VI. The conclusion

TABLE VI. Comparison Table With Existing Literature

Work Reference	Beats	Method	Accuracy (%)
Hu et al [21]	4	Mixture of Experts	94.00
Prasad et. al [7]	12	WT + NN	96.77
Osowski [22]	12	HER/HOS	95.91
Proposed	**6**	**k-PCA + SVM**	**98.96**

can be drawn from this table that k-PCA-SVM with optimized parameters outperforms the rest of approaches for the classification of ECG signals and can be considered as best amongst all.

V. Conclusion

This study presents a comparative performance analysis of the classifiers i.e BPNN and SVM with PCA and kernel based PCA algorithms for the classification of six classes of ECG signals evaluated on MIT-BIH database. The results clearly justifies the significance of a classifier with a proper feature extraction approach in terms of accuracy and other performance metrics. The multiclass SVM alonwith k-PCA with the optimized parameters has shown very good classification results on the six classes of ECG signals with an accuracy of 98.96% and average sensitivity, specificity and positive predictivity of 98.90%, 99.79% and 98.98% respectively which justifies the proposed approach and can be used in medical decision making systems for heartbeat classification.

References

[1] "World Health Organization factsheet N°317 Updated March 2013, available at: http://www.who.int/mediacentre/factsheets/fs317/en/."

[2] Y. Hu, L. Giovangrandi, and W. Tompkins, "A patient-adaptable ecg beat classifier using a mixture of experts approach," *IEEE Trans. on Biomedical Engineering*, vol. 44, no. 9, pp. 891–900, Sept. 1997.

[3] P. de Chazal, M. O08Dwyer, and R. B. Reilly, "Automatic Classification of Heartbeats Using ECG Morphology and Heartbeat Interval Features," *IEEE Trans. Biomed. Eng.*, vol. 51, no. 7, pp. 1196–1206, July 2004.

[4] F. d. Chazal and R. B. Reilly, "A patient adapting heart beat classifier using ECG morphology and heartbeat interval fatures," *IEEE Trans. Biomed. Eng.*, vol. 53, no. 12, pp. 2535–2543, Dec. 2006.

[5] Y. Hu, W. Tomkins, J. L. Urrusti, and V. X. Afonso, "Application of artificial neural networks for ECG signal detection and classification," *J. Electrocardiol.*, vol. 26, pp. 66–73, 1993.

[6] K. Minami, H. Nakajima, and T. Toyoshima, "Real-time discrimination of ventricular tachyarrhythmia with fourier-transform neural network," *IEEE Trans. on Biomed. Eng.*, vol. 46, no. 2, pp. 179–185, Feb. 1999.

[7] G. Prasad and J. Sahambi, "Classification of ecg arrhythmias using multi-resolution analysis and neural networks," in *TENCON 2003. Conference on Convergent Technologies for the Asia-Pacific Region*, vol. 1, Oct 2003, pp. 227–231 Vol.1.

[8] A. S. B. Scholkopf and K. R. Muller, "Nonlinear component analysis as a kernel eigenvalue problem," *Neural computing Journal*, vol. 10, no. 5, pp. 1299–1319, 1998.

[9] J. Lu, K. Plataniotis, and A. Venetsanopoulos, "Face recognition using kernel direct discriminant analysis algorithms," *Neural Networks, IEEE Transactions on*, vol. 14, no. 1, pp. 117–126, Jan 2003.

[10] F. R. Bach and M. I. Jordan, "Kernel independent component analysis," *Machine Learning Research Journal*, vol. 3, no. 9, pp. 41–48, 2002.

[11] D. Zhou and Z. Tang, "Kernel-based improved discriminant analysis and its application to face recognition," *Soft Comput Journal*, vol. 14, 2010.

[12] A. Khalifa, T. Fergany, R. Ammar, and M. Tolba, "Dynamic on-line allocation of independent task onto heterogeneous computing systems to maximize load balancing," in *Signal Processing and Information Technology, 2008. ISSPIT 2008. IEEE International Symposium on*, Dec 2008, pp. 418–425.

[13] G. B. Moody and R. G. Mark, "The impact of mit-bih arrhythmia database," *IEEE Trans. Biomed. Eng.*, vol. 20, no. 3, pp. 45–50, May-June 2001.

[14] S. Haykin, *Neural Networks: A comprehensive Foundation*. 1 Ed. New York,1994.

[15] B. Scholkopf, K. Tsuda, and J. Vert, "Kernel methods in computational biology," in *MIT Press series on Computational Molecular Biology. MIT Press*, 2004.

[16] V. N. Vapnik, "Statistical learning theory," in *New York: John Wiley and Sons*, 1998.

[17] J. Weston and C. Watkins, "Multi-class support vector machines," in *Citeseer, Tech. Rep.*, 1998.

[18] J. Platt, N. Cristianini, and J. Taylor, "Large margin dags for multiclass classification," *Adv Neural Inform Process Syst*, vol. 12, Mar. 2000.

[19] C. Hsu and C. Lin, "A comparison of methods for multiclass support vector machines," *IEEE Trans. Neural Network*, vol. 13, no. 2, pp. 415–425, Mar. 2002.

[20] J. Pan and W. J. Tompkins, "A real-time qrs detection algorithm," *Biomed. Eng., IEEE Trans. on*, vol. BME-32, no. 3, pp. 230–236, March 1985.

[21] Y. H. Hu, S. Palreddy, and W. J. Tomkins, "A patient adaptable ECG beat classifier using a mixture of expert approach," *IEEE Trans. Biomed. Eng.*, vol. 44, no. 9, pp. 891–900, Sept. 1997.

[22] S. Osowski, L. T. Hoai, and T. Markiewicz, "Support vector machine-based expert system for reliable heartbeat recognition," *Biomed. Eng., IEEE Trans. on*, vol. 51, no. 4, pp. 582–589, April 2004.