An Optimized and Efficient Radial Basis Neural Network using Cluster Validity Index for Diabetes Classification

Ramalingaswamy Cheruku, Damodar Edla, and Venkatanareshbabu Kuppili
Department of Computer Science and Engineering, National Institute of Technology Goa, India

Abstract: *This Radial Basis Function Neural Networks (RBFNNs) have been used for classification in medical sciences, especially in diabetes classification. These are three layer feed forward neural network with input layer, hidden layer and output layer respectively. As the number of the training patterns increases the number of neurons in the hidden layer of RBFNNs increases, simultaneously network complexity increases and classification time increases. Although various efforts have been made to address this issue by using different clustering algorithms like k-means, k-medoids, and Self Organizing Feature Map (SOFM) etc. to cluster the input data of diabetic to reduce the size of the hidden layer. Though the main difficulty of determination of the optimal number of neurons in the hidden layer remains unsolved. In this paper, we present an efficient method for predicting diabetics using RBFNN with optimal number of neurons in the hidden layer. This study mainly focuses on determining the number of neurons in hidden layer using cluster validity indexes and also find out the weights between output layer and a hidden layer by using genetic algorithm. The proposed model was used to solve the problem of detection of Pima Indian Diabetes and gave an accuracy of 73.50%, which was better than most of the commonly known algorithms in the literature. And also proposed methodology reduced the complexity of the network by 90% in terms of number of connections, furthermore reduced the classification time of new patterns.*

Keywords: *Radial basis function networks, classification, medical diagnosis, diabetes, optimal number of clusters, genetic algorithm.*

Received February 13, 2016; accepted February 8, 2017

1. Introduction

Diabetes Mellitus (DM), generally referred as diabetes. Diabetes is really a condition where the body fails to utilize the glucose properly. This is due to lack of sufficient insulin hormone in the body. It has symptoms like frequent urination, increased hunger, increase thirst and high blood sugar. Diabetes is the fastest rising long-term illness, condition that impacts lots of people globally. The excess blood sugar within the blood vessels can harm the blood vessels, this kind of situation leads to various complications like cardiovascular damage, kidney damage, nerve damage, eye damage and stroke [2, 39].

Classification systems are actually trusted in the health care sector to explore hidden patterns in the patient's data. These systems aid medical professionals to enhance their diagnosis, prognosis along with remedy organizing techniques. A lot of studies revealed that RBFNNs are helpful for classification and pattern recognition tasks. The performance of these neural networks is also on par with the more widely used Multi-Layer Perceptron Neural Network (MLPNN) model and the classical logistic regression. It utilizes fairly few numbers of locally tuned units known as neurons, and it is adaptive in nature. RBFNNs are based on supervised learning, these networks are good at modelling nonlinear data [31]. MLPNNis most popular for classification and it uses iterative process for training, since its iterative nature most of researchers proposed Radial Basis Function Neural Networks (RBFNNs) for classification task as an alternative to MLPNN. Unlike MLPNN, RBFNNs are trained in single iteration and also learn the given application quickly. The RBFNN is a distinct type of neural networks with a number of distinct capabilities. Since its first proposal, the RBFNN drawn a great attention in research areas. The RBFNN is made of three layers, specifically input layer, hidden layer and output layer [7, 20].

The size of the input layer is determined by the dimensionality of training patterns and output layer is by number of distinct classes in training patterns. To figure out number of neurons in the middle hidden layer, the simplest method is to assign a neuron for each training pattern. Even this simple approach is not possible practically as most of the applications are having large training patters with high-dimensionality. So usually it is a good practice to cluster the training patterns first to create a reasonable number of groups by employing clustering techniques like *k*-means, *k*-medoids, Self Organizing Feature Map (SOFM), etc., Once we create groups we can assign a neuron to each group (cluster) [7, 10].

As a way to identify the middle layer of an RBFNN we need to fix a number of cluster center locations in the hidden layer along with their basis function characteristics. Normally these basis functions are Gaussian functions. A Gaussian is usually characterized by means of the center location and shape (spread).To find center locations for Gaussian functions earlier so many attempts made by using clustering techniques. Mostly the *k*-means clustering process is used to locate a set of *k* Gaussian function centers because of its simplicity to implement and also it runs only in *0 (nkt)*, where *n* is the size of the data, *k* is the number for clusters and *t* is the number of iterations needed to algorithm convergence. These clustering algorithms partition the input data into *k* disjoint clusters. Once membership of all data points determined, average of cluster elements treated as the center location of that cluster. These center locations are used in Gaussian functions (basis function) of RNFNNs and the shape (spread) of Gaussian functions is determined by each cluster co-variance matrix.

In this paper, we used a Gaussian function as a kernel function and we are proposing the Optimized Radial Basis Neural Network (ORBFNN) based on cluster validity indexes to predict the diabetes mellitus.

1.1. RBF Network Model

The RBFNN [7, 18, 31] is a three layer feed forward architecture as shown in Figure 1. The construction of this type of network involves determination of number of neurons in 3 layers. The input layer is made up of *D* neurons where *D* is the dimensionality of input vector. The input layer is usually completely linked to hidden layer of size *H* neurons. These hidden layer neurons are complexly linked with output layer of size C_N neurons.

The output layer provides the response of the network for given patterns present at the input layer. There is no transformation happen at the input layer, this layer simply forward the whatever inputs are present to it. But at hidden layer it is a nonlinear transformation because of Gaussian activation functions and at output layer it is a linear transformation because the response of output layer neuron is a weighted sum of hidden layer outputs [7, 18, 31].

1. *Input layer*: This layer contains *D* number of neurons, where *D* is input vector dimensionality.
2. *Hidden layer*: This layer is made up of *H (H<N)* number of neurons, where *N* is the number of training samples.

Every neuron is mathematically described by a normalized radial basis function

$$\varphi_i(a) = \varphi(\|a - \mu_i\|), \quad i = 1,2,\dots,H \tag{1}$$

$$\varphi_i(a) = \frac{1}{\sqrt{(2\pi)^M |R_i|}} e^{-(a-\mu_i)^T R_i^{-1}(a-\mu_i)}, \tag{2}$$

$$R_{ij} = \sum_l (a_{li} - \mu_i)(a_{lj} - \mu_j), i,j = 1,2\dots,D \tag{3}$$

Where,

μ_i is the mean vector of cluster points determined from given data by clustering, *R* is the cluster covariance matrix and *l* is the index for sample pattern in the cluster.

The hidden layer is having mostly Gaussian functions as activation functions. These Gaussian functions are characterized by their mean vectors (centers)μ_i and shape (spread). The μ_i is the center for Gaussian function, and the vector *a* is the pattern presented at the input layer. The links joining the input layer neurons to the hidden layer neurons are direct connections with no weights [7, 18, 31].

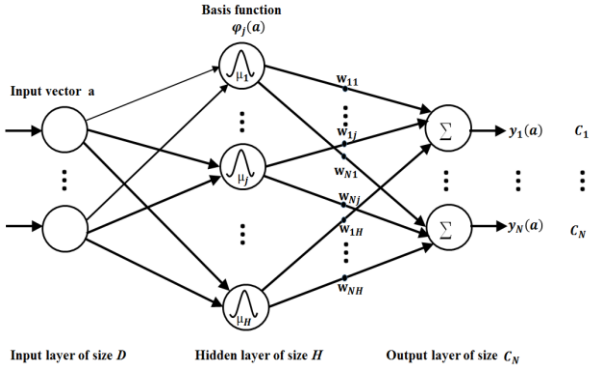

Figure 1. RBFNN architecture for pattern classification task.

3. *Output layer*: Size of this layer is very small. As shown in the Figure 1, the RBFNN structure consists of C_N neurons and with linear activation functions. Size of this layer is determined by number of distinct classes in the training data. Output of the j^{th} neuron in output layer given by

$$y_j(a) = \sum_j w_{ij}\, \varphi_i(a), i = 1,2,\dots,H, j = 1,2,\dots,C_N \tag{4}$$

Where,

w_{ij} is the weight between i^{th} unit in output layer and j^{th} unit in hidden layer.

4. *Next*, it is necessary to fix the class label for input pattern vector *a*. It is assigned to *j* Where

$$\arg \max_j y_j(a), j = 1,2,\dots,C_N \tag{5}$$

1.2. Cluster Validity Indexes

Commonly how many clusters are unknown within provided data. In *k*-means criteria, it is really hard to pre-determine value of *k*. So we need a metric for the partitioning result in order to find the perfect number of clusters. Commonly, that clustering outcome is actually tested with a qualifying measure called cluster validity index. Any validity index states precisely how very well this clustering split up this provided data sets [29, 33].

Several indexes were introduced in literature. These indexes are commonly merged into the clustering technique to have the overall finest intra-compact clusters and inter-separated clusters. Making use of

the intra-cluster and also the inter-cluster distances Ray and Turi planned a simple validity index to search for the optimal quantity of clusters inside color image segmentation [29, 33].We can define intra-inter-validity index as

$$\text{Intra} - \text{Inter Validity} = \frac{\text{Intra cluster distance}}{\text{Inter cluster distance}} \quad (6)$$

Where,

$$\text{Intra cluster distance} = \frac{1}{N}\sum_{i=1}^{k}\sum_{x\in c_i}\|x - z_i\|^2, \quad (7)$$

$$\text{Inter cluster distance} = \min_{i,j}\left(\|z_i - z_j\|^2\right) \quad (8)$$

Where,
i = 1, 2, …, k-1,
j = i + 1, i + 2, … , k,

Z_i denotes the center of the cluster c_i, k is the number of the clusters and N is the number of data points.

To find intra compact clusters and well separated inter clusters we have to minimize the intra cluster distance, i.e., distances between the points in the cluster and their cluster center, and maximize the inter cluster distance (by considering the minimum value for inter cluster distance), i.e., distance between cluster centers respectively. Overall, we have to minimize the intra-inter validity index for better clusters. It means that the minimum value for validity indicates intra compact clusters and inter well separated clusters. Another cluster index is Dunn index, it may be treated as a modified version of intra-inter index. Dunn index is defined in below Equation (15).

$$D_{nc} = \min_{i=1,\ldots,nc}\left\{\min_{j=i+1,\ldots nc}\left(\frac{d(c_i,c_j)}{\max_{k=1,\ldots nc}\text{diam}(c_k)}\right)\right\} \quad (9)$$

Where $d(c_i,c_j)$ is the dissimilarity function between two clusters c_i and c_j is given by

$$d(c_i, c_j) = \min_{x\in c_i, y\in c_j} d(x, y) \quad (10)$$

Cluster diameter diam (c) is defined in Equation (11), this can be treated as dispersion measure of cluster.

$$\text{diam}(C) = \max_{x,y\in C} d(x, y) \quad (11)$$

To identify compact and well-separated clusters we need to maximize Dunn index, unlike minimizing the intra–inter ratio. In other words, the maximum value for Dunn index indicates a good estimation of fine-tune cluster number for given data.

The above two indices are simple to implement but there are some disadvantages. The first one is, they are very sensitive to noise in the datasets [29, 33]. The second one is, in case of complex data sets like DNA microarray dataset, the definition of the minimal or maximal inter-distance used in the above two indices need not be compatible with the exact structure of the original data set. In case of complex datasets the geometry of clusters is arbitrary and it is hard to find well separated clusters [33]. By overcoming above two problems author in [33] has proposed new index called Dynamic Validity Index (DVI) to get a perfect cluster number. The new dynamic validity index represented as

$$\text{DVIndex} = \min_{k=1,..K}\{\text{IntraRatio}(k) + \gamma * \text{InterRatio}(k)\} \quad (12)$$

Where,

$$\text{IntraRatio}(k) = \frac{\text{Intra}(k)}{\text{MaxIntra}}, \quad (13)$$

$$\text{InterRatio}(k) = \frac{\text{Inter}(k)}{\text{MaxInter}}, \quad (14)$$

$$\text{Intra}(k) = \frac{1}{N}\sum_{i=1}^{k}\sum_{x\in c_i}\|x - z_i\|^2, \quad (15)$$

$$\text{MaxIntra} = \max_{i=1,..K}(\text{Intra}(i)), \quad (16)$$

$$\text{Inter}(k) = \frac{\max_{i,j}\left(\|z_i - z_j\|^2\right)}{\min_{i\neq j}\left(\|z_i - z_j\|^2\right)}\sum_{i=1}^{k}\frac{1}{\sum_{j=1}^{k}\left(\|z_i - z_j\|^2\right)}, \quad (17)$$

$$\text{MaxInter} = \max_{i=1,..K}(\text{Inter}(i)), \quad (18)$$

Where,
Z_i is the center of the cluster *Ci*, *N* is the number of data points, and *K* is the upper bound on number of clusters.

In Equation (13), *Intra* term in representing the overall compactness of clusters and in Equation (17) *Inter* term represents overall separateness of clusters. As the number of clusters increases intra term value decreases whereas inter term value increases. For the purpose of comparison author have done normalization for both terms using *MaxIntra* and *MaxInter* terms respectively to obtain *IntraRatio* and *InterRatio* terms. In [33] author also used a modelling parameter called γ, usually it set to 1 if there is no noise in the data. If there is some noise, we can set it by less than 1. In some special case we can set it greater than 1 if we intend to more compact clusters rather than well separated clusters. In other words, for which cluster number the DVIndex reaches a minimum value that indicates the optimal number of clusters for a given data set [33].

The organization of the rest of the paper is as follows. In section 2, we present a background and literature survey related to the problem. In section 3, we present in detail about the proposed model for the determination of the optimal neurons (units) needed in RBFNN hidden layer and also explained construction of Optimal Radial Basis Function Neural Network (ORBFNN). In section 4, we presented some experimental outcomes that confirm the performance of the proposed methodology. In section 5, we have drawn some final conclusions and provided some extension works that can further improve my model in future scope.

2. Background and Related Work

Classification and decision support systems have been using extensively by medical domain for disease diagnosis. This will help doctors to improve their diagnosis procedure and to provide better planning for treatment. In recent years, many studies have been

performed in literature for the diagnosis of diabetic disease. Several statistical methods have also been used. In [34] authors have used Artificial Neural Networks (ANNs) along with Bee colony optimization for Magnetic Resonance Imaging (MRI) brain image classification. In [24] a logistic regression model was used to predict diabetic status. In [30] authors were conducted experiments on Pima Indian Diabetes (PID) dataset using three different classifiers namely MLPNN, Naive Bayes (NB) Classifier and J.48. SVM has used for classification of DM patients in [4]. In [35] authors have proposed robust version of SVM called VaR-SVM. In [26] authors recommended multiple knot spline SSVM for classification problems. To estimate the efficiency of their technique, they tested on PID dataset. In [3] authors performed a comparative analysis on diabetes classification techniques. In [3, 16] different neural networks, such as Cascade-Forward Networks (CFN), Probabilistic Neural Network (PNN), Time Delay Networks (TDN), Distributed Time Delay Networks (DTDN), used for DM patients classification.

A classification algorithm based on Ant Colony Optimization (AC0), Fuzzy systems and ANNs techniques was proposed in [9, 40]. A fuzzy logic approach for diabetes classification is introduced in [28]. In [19] authors have used MDL-based decision tree for the classification of diabetes. In [14] authors have used genetic algorithm for finding the neural network weights, which are used for diabetes classification.

A hybrid binary classification model based on the concepts like ANN and soft computing techniques was proposed for classification of type2 diabetes patients in [15]. In [13] a comparison study has been performed for binary classification problems using different neural networks. In [25] RBFNN along with novel kernel density estimation function was used for data classification. In [37] authors have described about application of RBFNN in analysis of diabetes and also compared performance of RBFNN with MLPN and logistic regression.

As there is problem of hidden layer size in RBFNN, which is same as number of training samples. To tackle this problem authors have proposed clustering of input data. In literature RBFNN centres were obtained by many clustering algorithms, such as fuzzy *c*-means [5], enhanced LBG [32], *k*-means [18] and others [6, 22] etc.The clustering procedure gets the cluster centers by trying to minimize the total squared error incurred in representing the data set by the different cluster centers. So many authors made attempt in this direction to find intra compact clusters as well as inter well separated clusters by proposing different cluster validity measures given in [29, 33]. In [38] an analysis of fuzzy cluster validity indices is presented. In [1] authors investigated relationship between hypertension and diabetes. They have used Oracle Data Miner (ODM) tool for dataset analysis.

3. Proposed Work

In this paper, we present a new method to figure out the number of basis centers needed to RBFNNs for the classification of diabetes. This method uses a cluster validity index measure of clustering to fine-tune the clusters and calculates the intra similarity and inter dissimilarity of every cluster. The actual output of the proposed method trying to concentrate on optimal clusters in those input regions where the cluster validity index is more or less depending on kind of cluster validity index using, thus attempting to maximize intra similarity and inter dissimilarity of every cluster.

Once we figure out the number of basis centers in the hidden layer, next it is necessary to fix the weights between the hidden layer neurons and the output layer neurons. As we have linear activation neurons in the output layer, using matrix inversion, these weights can be find directly. Matrix inversion is computational expensive as the size of training patterns grows, so it is hard to find the weights in RBFNN. So, instead of a matrix inversion, here we are proposing genetic algorithm based technique to find the suitable weights. The block diagram of the proposed model is presented in Figure 2, and the pseudo codes of the proposed model summarized in Algorithms 1 and 2.

Pseudo code shown in Algorithm 1 is used for construction of ORBFNN classifier by obtaining the optimal number of cluster center locations and pseudo codein Algorithm 2 is used to prediction of class label for new inexperienced patterns present to ORBFNN classifier.

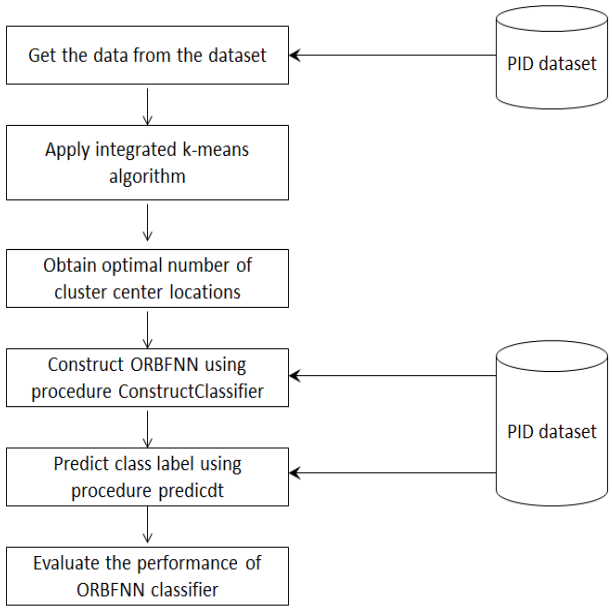

Figure 2. Block diagram of proposed model.

Algorithm 1: The pseudo codefor the construction of ORBFNN classifier.

Procedure ConstructClassifier
Input: A set of training patternsS = {s1, s2, …, sn};

Dimensionality of training pattern D;
No of distinct classes in training patterns C_N;
Genetic algorithm parameter values Popsize, Mutrate, Selection, Nbitsfrom Table 1;
Noise parameter for DV index γ=1;
Output: Optimal RBF network.
Begin
Step1: Determination of number of basisfunctions
Let Sbe the set of training patterns and run integrated k-means algorithm (k-means along with cluster validity index) with Sas input;
For each μ_i ε OptCluLoc
LetOptCluLoc be the set of optimal cluster clusters for input data as a output ofIntegrated k-means, H is the total number of optimal clusters needed for all classes; And $i = 1,2, \dots, H$;
Compute

$$\varphi_{\mathrm{i}}(\mathrm{a}) = \frac{1}{\sqrt{(2\pi)^{M}|R_i|}} e^{-(a_i-\mu_i)^T R_i^{-1}(a_i-\mu_i)} \quad (19)$$

Compute

$$R_i = \sum_{l} (a_{li} - \mu_i)\,(a_{lj} - \mu_j), \quad i,j = 1,2 \dots., D \quad (20)$$

Where,
c_i is the ith cluster of input data,
$\varphi_i(a)$ is basis function value ofc_i,
M is the dimentionality of training patterns, μ_i is the center of cluster c_i,
R_i is the covariance matrix of cluster c_i,
a_i is the training sample of c_i ε OptCluLocandi is the cluster number,
Endfor
Construct a fully connected, feed forward optimal RBFN network with D input layer units,H hidden layer units and C_N output units;
Step 2: Determination of output layer weights
Obtain the weights of output layer using Genetic Algorithm;
End

Algorithm 2: The pseudo codefor prediction of class label.

Procedure predict
Input: an input pattern a is fed into RBF network constructed with the procedure presented in Procedure ConstructClassifier;
Output: Classlabel prediction forthe input pattern a;
Begin
Let Tbe the set that consists of testingpatterns;
Threshold = 0;
For each a ε T
Compute the value of $y_j(a) = \sum_j w_{ij}\varphi_i(a)$with below equations

$$\varphi_{\mathrm{i}}(\mathrm{a}) = \frac{1}{\sqrt{(2\pi)^{\mathrm{M}}|\mathrm{R_i}|}} \mathrm{e}^{-(\mathrm{a}-\mu_{\mathrm{i}})^{\mathrm{T}}\mathrm{R_i}^{-1}(\mathrm{a}-\mu_{\mathrm{i}})}, \quad \mathrm{i} = 1,2, \dots, \mathrm{H}$$

$R_i = \sum_l (a_{li} - \mu_i)\,(a_{lj} - \mu_j), i,j = 1,2 \dots., D$
Where,
l is the index of sample a,
μ_i is the ithcenter of cluster,
D is th dimensionality of training pattern.
Endfor
If ($y_j(a)$>Threshold) then
Classlabel= 1;
Else
Classlabel = 0;
End If
Return (Classlabel);
End

3.1. Determination of Output Layer Weights by Genetic Algorithm

Conventional matrix inversion method for weights calculation between output and hidden layers is computationally expensive and feature matrix singularity problem arises, as the number of training patterns increases. Hence, Genetic Algorithm (GA) is used for overcoming these problems [23]. We can perceive (finding weights) this problem as a system of linear equations. Systems of equations are functions of at least two variables (weights). In order to find a solution for the system of equations the coefficient matrix should be non-singular. A system of linear equations shown below are

$$\left.\begin{array}{l} H_{11}\mathrm{W}_1 + H_{12}\mathrm{W}_2 + \cdots H_{1j}\mathrm{W_j} = \mathrm{Y}_1 \\ H_{21}\mathrm{W}_1 + H_{22}\mathrm{W}_2 + \cdots H_{2j}\mathrm{W_j} = \mathrm{Y}_2 \\ \vdots \\ H_{i1}\mathrm{W}_1 + H_{i2}\mathrm{W}_2 + \cdots H_{ij}\mathrm{W_j} = \mathrm{Y}_i \end{array}\right\}$$

Which is represented in matrix form as HW=Y
Where,
H is the coefficient matrix (feature vector matrix),
W is unknown weight matrix,
Y is output matrix(Y_i can be either 0 or +1),
j is the number of features in each training vector and
i is the number of training samples.

For any given system of simultaneous linear equations, it was noticed that the GA was really effective in finding out all possible sets of answers that are appropriate. Whereas, a single set of solutions are produced by conventional Gaussian elimination method. The GA approach follows survival of fittest concept. The solutions are evaluated over generations by a fitness function to find good solution to the problem [21].

In GA another important information needed is defining the fitness function. At every generation using objective function, we can evaluate the fitness value of a possible solution. Therefore, in order to define the fitness function for systems of simultaneous equations we have expressed them as follows set of equations must be all at minimum.

$$\left.\begin{array}{l} \mathrm{Fun}_1(\mathrm{W}_1, \mathrm{W}_2, \dots, \mathrm{W_j}) - \mathrm{Y}_1 = 0 \\ \mathrm{Fun}_2(\mathrm{W}_1, \mathrm{W}_2, \dots, \mathrm{W_j}) - \mathrm{Y}_2 = 0 \\ \vdots \\ \mathrm{Fun_i}(\mathrm{W}_1, \mathrm{W}_2, \dots, \mathrm{W_j}) - \mathrm{Y_i} = 0 \end{array}\right\}$$

Where, *Y* is output matrix(Y_i can be either 0 or 1), *i* is the number of training samples and *j* is the number of features in each training vector.

A correct solution has to satisfy all of the above equations. These sets of equations constituted the actual objective function for Genetic Algorithm to solve systems of linear equations.

In order to find the unknown weights in the above equations, we randomly generated data to represent each unknown weight in a system of simultaneous

equations, which constitutes each chromosome. Several of the chromosomes initially generated from the initial generation. In finding the chromosome fitness, the randomly generated vector of values (genes) for all the unknowns were used to evaluate each line of the equation. The results obtained were subtracted from the Right Hand Side (RHS) values of the original equation. The differences obtained were summed and the summation squared, that is, *(sum of difference)*^{2} [12].

The RHS values of each original equation were also summed and the summation also squared, that is, *(sum of RHS)*^{2}. The each chromosome fitness calculated using, the concept of coefficient of multiple determination was used. This concept is also called the squared multiple correlation coefficient which is obtained by

$$\text{SMCC} = \frac{((\text{Sum of RHS})^2 - (\text{Sum of Difference})^2)}{(\text{Sum of RHS})^2} \quad (21)$$

The values of *SMCC* range between 0.0 and 1.0 (i.e., $0.0 \leq SMCC \leq 1.0$); and the fitness values that are closer to 1 imply a better fitness while a fitness value of 1.0 gives the best fitness that produces the most accurate solution to the equations. A well fitted set of generation forms the initial population for the next generation and subsequently until the stopping criterion of fitness of 1.0 or very close to 1 is obtained or the maximum generation indicated in the program has been reached [12, 17].

4. Experimental Results

4.1. Experimental Setup

The ORBFNN model has been developed for the classification of diabetes. These experiments were conducted using Matlab R2015a on 4GB RAM, Intel i3 processor (3.40GHz) system. The Pima Indians Diabetes (PID)dataset is stored in a text document and read directly using Matlab R2015a.We have used parameters like classification accuracy, sensitivity, specificity and also classification time to estimate the performance of the developed model. Parameters values listed in Table 1 are used in experiments. Except γ parameter rest of the parameters used in initialization of genetic programming and γ is used DV index as a noise parameter.

Table 1. Various parameter value used in experiments.

S. No	Parameter	Value	Explanation
1	Γ	1	Noise parameter for DV index
2	PopSize	16	Initial population size
3	MutRate	0.14	Mutation rate
4	Selection Rate	0.6	Fraction of population that survive after every generation
5	Nbits	8	Number of bits in each parameter
6	MaxGen	1000	Maximum Number of generations

4.2. Diabetes Disease Dataset

The Pima Indians, Native Americans who live around Arizona, are the most intense type-2 diabetic population in the world. Since it is a homogeneous group, the data taken from these people are the subject of intense studies in diabetics. Pima dataset is a collection of 768 female patients medical reports of which 500 cases in class 0 and 268 cases in class 1 [36]. Table 2 shows the attributes of the dataset. Accordingly, 9 attributes (8 input and 1 output) were studied. Output information or class values are indicated as 0: no diabetes (negative) and 1: diabetes (positive).

In order to assess the performance of the proposed ORBFNN model, we have partitioned the PID dataset into two sets called training and testing data sets. The entire PID database having a total of 768 patients (records) of data Training data set consisted around 68% records (518) and testing data set consisted around 32% PID records (250).

Table 2. Pima Indians diabetes dataset attributes.

Attribute no.	Attribute
1	Age
2	Number of times pregnant
3	Concentration of plasma glucose
4	2-h serum insulin (mu U/mL)
5	Triceps skin-fold thickness (mm)
6	Diabetes pedigree function
7	Body mass index (kg/m2)
8	Diastolic blood pressure (mmHg)
9	Class 0 or 1

The training data set is used for training the proposed model and the testing data set is used to measure the model performance. In the training phase the proposed model is used training records to create ORBFNN (to determine optimal hidden layer units and weights of output layer).In the testing phase, our proposed ORBFNN model is fed by unseen records from testing data set.

4.3. Performance Analysis

The proposed model experimented on PID dataset. We set the maximum number of cluster centers in input data to 100.The input for the integrated *k*-means algorithm is fed with a total of 518 records and was run over 100 times (to find cluster centers for 2 to 100 clusters) using three different cluster validity indexes namely intra-inter ratio validity index, DV index and Dunn index (to find the optimal cluster number and center locations).

We found that integrated *k*-means using intra-inter ratio validity index given minimum ratio value at number of clusters equal to 48.So, we have considered 48 as optimal cluster number and its corresponding cluster mean values are optimal cluster center locations for the input data. Similarly, we run integrated *k*-means with DV index we found that 39 was the optimal cluster number and also run integrated *k*-means using Dunn index unlike other 2 validity indexes we considered the maximum value

corresponds to optimal cluster so we found this was 83.All the results are given in Table 3 and also simulated outputs are shown in Figure 3.

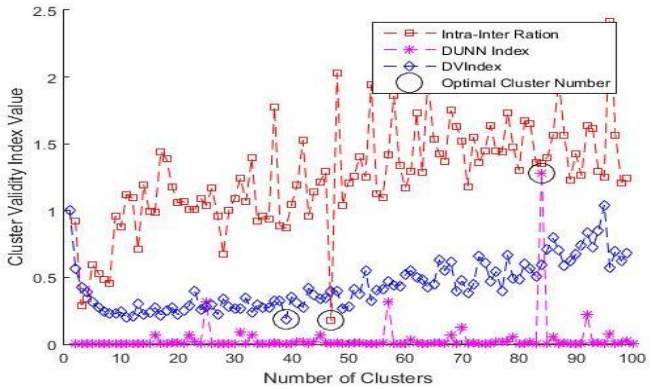

Figure 3. Comparison of various indexes.

Once we determined the optimal center locations of clusters, we have created the ORBFNN classifier with the optimized center locations found using integrated *k*-means and Gaussian kernel activation function parameter of each cluster which are also found using integrated *k*-means. Next we found weight values of output layer by using genetic algorithm (after proper tuning of parameters listed in Table 1). The execution was carried over 1000 generations.

Table 3. Optimal number of clusters determined by validity indexes.

	Optimal No of Clusters
Intra-Inter Ratio	48
Dunn	83
Dynamic Validity Index	39

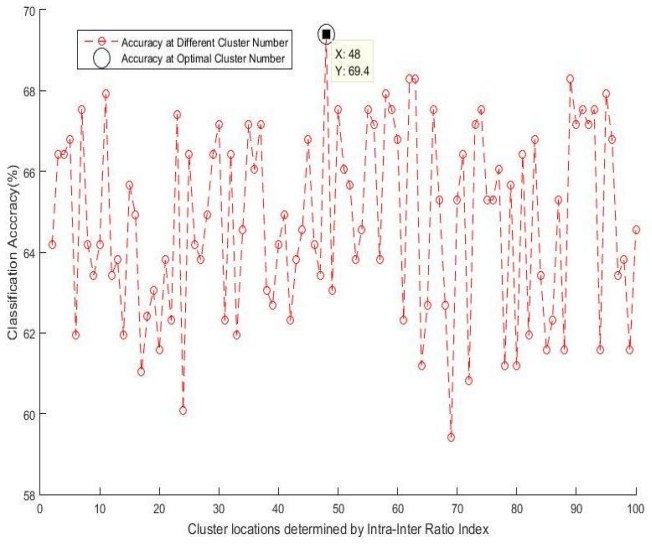

Figure 4. Performances at various cluster locations determined by Intra-Inter ratio index.

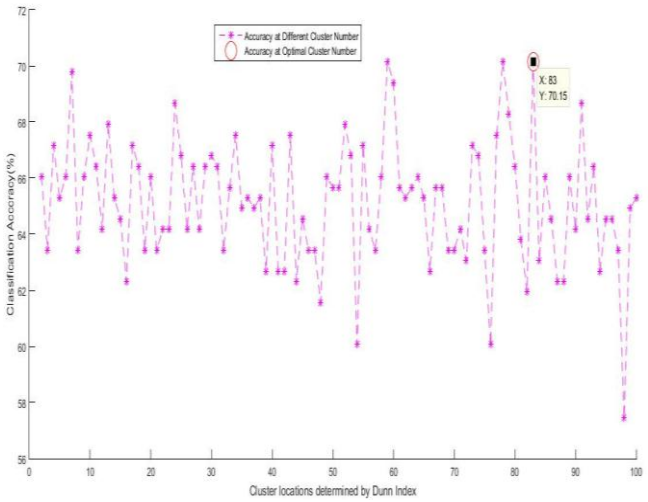

Figure 5. Performances at Various Cluster Locations Determined by DUNN Index.

Next, we have collected the model output and compared against the standard output. This determines the accuracy rate or classification accuracy of the system. The classification accuracy calculated using Equation (22) [11]. Where TP is the True Positive count represents the number of patients that the model classified to have diabetes among the patients detected with diabetes by a medical doctor, TNIs the True Negative count represents the number of patients that the model classified to be non-diabetic among the patients detected as non-diabetic by a medical doctor, FP is the False Positive count represents the number of patients that the model classified to have diabetes among the people detected as non-diabetic by a medical doctor, and FN is the False Negative count represents the number of patients that the model classified to be non-diabetic among the patients detected with diabetes by a medical doctor [11] .

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{22}$$

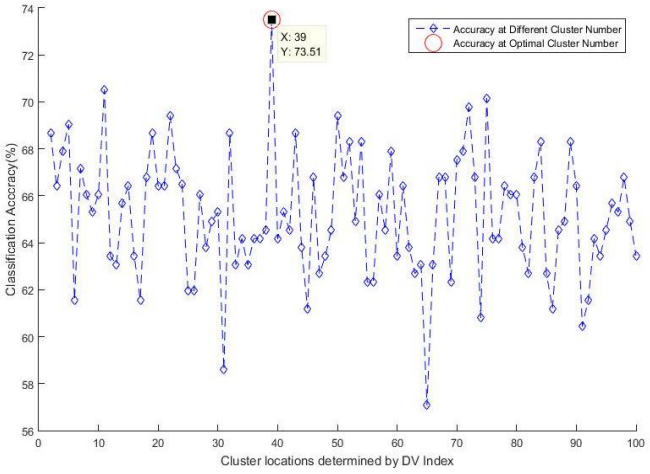

Figure 6. Performances at various cluster locations determined by DV Index.

Performances values of the proposed ORBFNN at optimal number clusters determined by three validity indexes namely intra-inter ratio, Dunn index, DV index were captured and presented in the Table 4 and also simulated outputs showed in Figures 4, 5, and 6 respectively.

Table 4. Performance comparison of three validity indexes.

	Optimal number of Clusters	Classification Accuracy Achieved
Intra-Inter Ratio	**48**	**69.40 %**
Dunn	**83**	**70.15 %**
Dynamic Validity Index	**39**	**73.50 %**

Next, we have compared proposed ORBFNN with three validity indexes against the conventional RBFNN. Experimental results proved that proposed model with three validity indexes was achieved the best accuracy as compared with conventional RBFNN and also reduced complexity of network drastically. This in turn reduced the classification time (time for classifying single unknown pattern by model) to 5.17 seconds, which is very less time compare to conventional RBFNN which has taken 574 .75 seconds. Thus we can classify the unknown patterns very quickly. These comparison results are provided in the Table 5. We got best results using DV index, best values highlighted in the Table 5.

We have calculated confusion matrix for proposed model, it is shown in Table 6. Also, we have calculated sensitivity and specificity parameters [11] from confusion matrix using Equations (23), and (24) in order to compare our proposed ORBFNN with other older models like PNN,CFN, TDN, Feed Forward Network (FFN), decision tree based model GINI and Artificial Immune System (AIS) experimented on PID dataset. These comparison results were presented in the Table 7 and best values are highlighted. These experimental results proved that our proposed ORBFNN has achieved more accuracy than models mentioned above. And also balanced the both network complexity and performance.

$$\text{Sensitivity} = \frac{\text{TP}}{\text{TP+FN}} \tag{23}$$

$$\text{Specificity} = \frac{\text{TN}}{\text{TN} + \text{FP}} \tag{24}$$

Table 5. Comparison of complexity of network, number of hidden layer neurons and classification accuracy between proposed model and conventional RBFNN.

	Conventional RBFNN	ORBFNN With Intra-Inter Ratio Index	ORBFNN With Dunn Index	ORBFNN with DV Index
# of Hidden Layer Neurons	**768**	48	83	**39**
# of Links (Complexity of network)	**7680**	480	830	**390**
Classification Accuracy	**68.53 %**	69.40 %	70.15 %	**73.50 %**
% reduction in network complexity	**0%**	93.75%	89.19%	**94.9%**
Classification Time for single sample in seconds	**574.75**	7.26	45.96	**5.17**

Table 6. Confusion matrix for the ORBFNN.

		Predicted Class		
		Yes	**No**	**Total**
Actual Class	**Yes**	**TP = 58**	**FP = 34**	**92**
	No	**FN = 32**	**TN = 126**	**158**
	Total	**90**	**160**	**250**

Table 7. Comparison of various models against proposed model.

Model	Dataset	Accuracy (%)	Sensitivity (%)	Specificity (%)
FFN	PID	68.80 %	54.44 %	76.88 %
CFN	PID	68.00 %	62.22 %	71.25 %
PNN	PID	72.00 %	63.33 %	76.88 %
TDN	PID	66.80 %	41.11 %	81.25 %
GINI	PID	65.97 %	44.71 %	77.78 %
AIS	PID	68.80 %	52.22 %	78.13 %
Proposed Model	**PID**	**73.50 %**	**64.44 %**	**78.75 %**

Finally, the proposed ORBFNN has been compared with existing standard algorithms like Memetic Elitist Pareto non dominated sorting genetic algorithm based RBFN (MEPGAN) f1-f3 [27], MEPGANf1f2 [27], Bee-RBF [8] and Bat-RBFN [7]. These methods are improvements of RBFNN in the literature. These comparison results are shown in terms of accuracy, sensitivity and specificity in Table 8. The proposed method results are highlighted in Table 8. It is observed from table results that the proposed model achieved highest accuracy and balanced sensitivity and specificity when compared with other standard algorithms.

Table 8. Comparison of proposed model and other methods in the literature.

Model	Accuracy (%)	Sensitivity (%)	Specificity (%)	Year [Ref]
MEPGANf1–f3	68.35	20.37	94.00	2013 [26]
MEPGANf1f2	72.78	45.20	87.11	2013 [26]
Bee-RBF	71.13±1.06	--	--	2016 [9]
Bat-RBFN	70..00	77.34	56.25	2017 [8]
Proposed Model	**73.50 %**	**64.44 %**	**78.75 %**	**This study**

5. Conclusions and Future Work

This paper proposed a new classification model for classifying diabetes patients. The proposed model integrates cluster validity index with *k*-means clustering algorithm. The proposed model was comprised of two main stages which are determination of optimal cluster center locations and classification. Our model was used to classify diabetes patients into one of two classes (positive/negative). Cluster validity index integrated with *k*-means algorithm to guarantee the optimal cluster locations. Optimizing cluster centers minimized the classification time by reducing network complexity. The proposed model was experimented on PID dataset of UCI repository. The average classification accuracy of model is 73.50% with DV index at 39 optimal cluster number is the best while compared with conventional RBFNN performance of 68.53%. And further this reduced the network complexity by 90% and classification time to

5.17 Seconds. The proposed method also achieved highest accuracy and balanced sensitivity, and specificity, when compared with other improved versions of RBFNN. As a future work, hybrid particle swarm optimization can be used for determining weights. Also, we can apply other kernel functions like poly harmonic spline, inverse quadratic etc., in the classification phase.

References

[1] Aljumah A. and Mohammad S., "Data Mining Perspective: Prognosis of Life Style on Hypertension and Diabetes," *The International Arab Journal of Information Technology*, vol. 13, vol. 1, pp. 93-99, 2016.

[2] American Diabetes Association, Diabetes complication, available at: www.diabetes.org/living-with-diabetes/complications, Last Visited, 2015.

[3] Bozkurt M., Yurtay N., Yilmaz Z., and Sertkaya C., "Comparison of Different Methods for Determining Diabetes Disease," *Turkish Journal of Electrical Engineering and Computer Sciences*, vol. 22, pp. 1044-1055, 2014.

[4] Barakat N., Bradley A., and Barakat M., "Intelligible Support Vector Machines for Diagnosis of Diabetes Mellitus," *IEEE Transactions on Information Technology in Biomedicine*, vol. 14, no. 4, pp. 1114-1120, 2010.

[5] Bezdek J., Ehrlich R., and Full W., "FCM: The Fuzzy C-Means Clustering Algorithm," *Computers and Geosciences*, vol. 10, no. 2, pp. 191-203, 1984.

[6] Cabestany J., Alberto P., and Sandoval F., "Computational Intelligence and Bio-inspired Systems," *in Proceedings of the 8th International Work-Conference on Artificial Neural Networks*, Barcelona, 2005.

[7] Cheruku R., Edla D., and Kuppili V., "Diabetes Classification using Radial Basis Function Network by Combining Cluster Validity Index and BAT Optimization with Novel Fitness Function," *International Journal of Computational Intelligence Systems*, vol. 10, no. 1, pp. 247-265, 2017.

[8] Cruz D., Maia R., Silva L., and Castro L., "BeeRBF: a Bee-Inspired Data Clustering Approach to Design RBF Neural Network Classifiers," *Neurocomputing*, vol. 172, pp. 427-437, 2016.

[9] Fiuzy M., Qarehkhani A., Haddadnia J., and Vahidi J., "Introduction of A Method to Diabetes Diagnosis According to Optimum Rules in Fuzzy Systems Based on Combination of Data Mining Algorithm (D-T), Evolutionary Algorithms (Aco) and Artificial Neural Networks (nn)," *The Journal of Mathematics and Computer Science*, vol. 6, no. 4, pp. 272-285, 2013.

[10] Halkidi M., Batistakis Y., and Vazirgiannis M., "Clustering Algorithms and Validity Measures," *in Proceedings of 3th International Conference on Scientific and Statistical Database Management*, Fairfax, pp. 3-22, 2001.

[11] Han J., Pei J., and Kamber M., *Data Mining: Concepts and Techniques*, Elsevier, 2011.

[12] Ikotun A., Olawale L., and Adebowale A., "The Effectiveness of Genetic Algorithm in Solving Simultaneous Equations," *International Journal of Computer Applications*, vol. 14, no. 8, pp. 38-41, 2011.

[13] Jeatrakul P. and Wong K., "Comparing the Performance of Different Neural Networks for Binary Classification Problems," *in Proceedings of the 8th International Symposium on Natural Language Processing*, Bangkok, pp. 111-115, 2009.

[14] Karegowda A., Manjunath A., and Jayaram M., "Application of Genetic Algorithm Optimized Neural Network Connection Weights for Medical Diagnosis of Pima Indians Diabetes," *International Journal on Soft Computing*, vol. 2, no. 2, pp. 15-23, 2011.

[15] Khashei M., Eftekhari S., and Parvizian J., "Diagnosing Diabetes Type Ii Using A Soft Intelligent Binary Classification Model," *Review of Bioinformatics and Biometrics*, vol. 1, no. 1, pp. 9-23, 2012.

[16] Koklu M. and Unal V., "Analysis of a Population of Diabetic Patients Databases with Classifiers," *International Journal of Biomedical and Biological Engineering*, vol. 7, no. 8, pp. 481-483, 2013.

[17] Leon S., *Linear Algebra with Applications*, Macmillan, 1980.

[18] Mashor M., "Improving the Performance of K-Means Clustering Algorithm to Position the Centers of RBF Network," *International Journal of the Computer, The Internet and Management*, vol. 6, no. 2, pp. 121-124, 1998.

[19] Mehta M., Rissanen J., and Agrawal R., "MDL-Based Decision Tree Pruning," *in Proceedings of the 1st International Conference on Knowledge Discovery and Data Mining*, Montréal, pp. 216-221, 1995.

[20] Mitchell T., *Machine Learning*, McGraw-Hill, 1997.

[21] Mitchell M., *an Introduction to Genetic Algorithms*, MIT Press, 1998.

[22] Moody E. and Darken C., "Fast Learning in Networks of Locally Tuned Processing Units," *Computer Journal Neural Computation*, vol. 1, no. 2, pp. 281-294, 1989.

[23] Mardle S. and Pascoe S., "An Overview of Genetic Algorithms for The Solution of Optimization Problems," *Computers in Higher*

Education Economics Review, vol. 13, no. 1, pp. 6-20, 1999.

[24] Morteza A., Nakhjavani M., Asgarani F., Carvalho F., Karimi R., and Esteghamati A., "Inconsistency in Albuminuria Predictors in Type2 Diabetes: A Comparison between Neural Network and Conditional Logistic Regression," *Translational Research*, vol. 161, no. 5, pp. 397-405, 2013.

[25] Oyang Y., Hwang S., Ou Y., Chen C., and Chen Z., "Data Classification with Radial Basis Function Networks Based on A Novel Kernel Density Estimation Algorithm," *IEEE Transactions on Neural Networks*, vol. 16, no. 1, pp. 225-236, 2005.

[26] Purnami S., Embong A., Zain J., and Rahayu S., "A New Smooth Support Vector Machine and Its Applications in Diabetes Disease Diagnosis," *Journal of Computer Science*, vol. 5, no. 12, pp. 1003-1008, 2009.

[27] Qasem N., Shamsuddin S., Hashim S., Darus M., and Al-Shammari E., "Memetic Multi Objective Particle Swarm Optimization-Based Radial Basis Function Network for Classification Problems," *Information Sciences*, vol. 239, pp. 165-190, 2013.

[28] Radha R. and Rajagopalan S., "Fuzzy Logic Approach for Diagnosis of Diabetes," *Information Technology Journal*, vol. 6, no. 1, pp. 96-102, 2007.

[29] Ray S. and Turi R., "Determination of Number of Clusters In K-Means Clustering and Application in Colour Image Segmentation," *in Proceedings of the 4th International Conference on Advances in Pattern Recognition and Digital Techniques*, Calcutta, pp. 137-143, 1999.

[30] Rahman R. and Afroz F., "Comparison of Various Classification Techniques Using Different Data Mining Tools for Diabetes Diagnosis," *Journal of Software Engineering and Applications*, vol. 6, no. 3, pp. 85-97, 2013.

[31] Rizvan E., Oğulata S., Şahin C., and Alparslan Z., "A Radial Basis Function Neural Network (RBFNN) Approach for Structural Classification of Thyroid Diseases," *Journal of Medical Systems*, vol. 32, no. 3, pp. 215-220, 2008.

[32] Russo M. and Patanè G., "Improving the LBG Algorithm," *in Proceedings of International Work-Conference on Artificial Neural Networks*, Alicante, pp. 621-630, 1999.

[33] Shen J., Chang S., Lee E., Deng Y., and Brown S., "Determination of Cluster Number in Clustering Microarray Data," *Applied Mathematics and Computation*, vol. 169, no. 2, pp. 1172-1185, 2005.

[34] Subramaniam S. and Radhakrishnan M., "Neural Network with Bee Colony Optimization for MRI Brain Cancer Image Classification," *The International Arab Journal of Information Technology*, vol. 13, no. 1, pp. 118-124, 2016.

[35] Tsyurmasto P., Zabarankin M., and Uryasev S., "Value-at-Risk Support Vector Machine: Stability to Outliers," *Jornal of Combinatorial Optimization*, vol. 28, no. 1, pp. 218-232, 2014.

[36] UCI Machine Learning, Pima Indians Diabetes Data Set, Irvine, CA, USA, University of California Irvine, available at http://archive.ics.uci.edu/ml/machine-learning-databases/pima-indians-diabetes, Last Visited, 2015.

[37] Venkatesan P. and Anitha S., "Application of a Radial Basis Function Neural Network for Diagnosis of Diabetes Mellitus," *Current Science*, vol. 91, no. 9, pp. 1195-1199, 2005.

[38] Wang W. and Yunjie Z., "On Fuzzy Cluster Validity Indices," *Fuzzy Sets and Systems*, vol. 158, no. 19, pp. 2095-2117, 2007.

[39] World Health Organization, Global report on Diabetes, available at: http://www.who.int/diabetes/global-report/en/, Last Visited, 2015.

[40] Zhou Q., Purvis M., and Kasabov N., "A Membership Function Selection Method for Fuzzy Neural Networks," *Information Science Discussion Papers Series No. 97/15)*, University of Otago, 1997.

Ramalingaswamy Cheruku is presently working as Full-time research scholar in Department of Computer Science and Engineering (CSE) at National Institute of Technology Goa, India. He received B.Tech. Degree in CSE from JNT University, Kakinada in 2008, M.Tech. Degree from ABV Indian Institute of Information Technology, Gwalior in 2011. He has served as developer in Tata Consultancy Services for 2 years.

Damodar Edla is an Assistant Professor and Head of the CSE Departmentat National Institute of Technology Goa. He received M.Sc. Degree from University of Hyderabad in 2006, M.Tech. and PhD Degree in CSE from Indian School of Mines Dhanbad in 2009 and 2013 respectively.

Venkatanareshbabu Kuppili is an assistant professor in department of CSE, National Institute of Technology Goa, India. He was with EvalueservePvt. Ltd, as a Senior Research Associate. He received M.Tech. and Ph. D. Degree in CSE from Indian Institute of Technology Delhi, India. He has authored a number of research papers published in reputed international journals in the area of neural networks, classification, and clustering.