An Improved Generalized Regression Neural Network for Type II Diabetes Classification

Moeketsi Ndaba[1(✉)], Anban W. Pillay[1,2], and Absalom E. Ezugwu[1]

[1] School of Mathematics, Statistics and Computer Science, University of Kwazulu-Natal, Westville Campus, Private Bag X54001, Durban 4000, South Africa
ndaba.max@gmail.com, {pillay4,ezugwuA}@ukzn.ac.za
[2] Centre for Artificial Intelligence Research (CAIR), Durban, South Africa

Abstract. This paper proposes an improved Generalized Regression Neural Network (KGRNN) for the diagnosis of type II diabetes. Diabetes, a widespread chronic disease, is a metabolic disorder that develops when the body does not make enough insulin or is unable to use insulin effectively. Type II diabetes is the most common type and accounts for an estimated 90% of cases. The novel KGRNN technique reported in this study uses an enhanced K-Means clustering technique (CVE-K-Means) to produce cluster centers (centroids) that are used to train the network. The technique was applied to the Pima Indian diabetes dataset, a widely used benchmark dataset for Diabetes diagnosis. The technique outperforms the best known GRNN techniques for Type II diabetes diagnosis in terms of classification accuracy and computational time and obtained a classification accuracy of 86% with 83% sensitivity and 87% specificity. The Area Under the Receiver Operating Characteristic Curve (ROC) of 87% was obtained.

Keywords: Diabetes classification · Artificial Neural Networks Generalized Neural Network

1 Introduction

Diabetes is one of the most challenging and widespread chronic diseases in the world [22] with an estimated 425 million diagnosed cases worldwide [8]. This number is expected to rise to 629 million by 2045 [8]. There are three types of diabetes: Type I (insulin-dependent diabetes), Type II (non-insulin-dependent diabetes) and gestational diabetes which affects pregnant women. Type II diabetes is the most prevalent form of the disease afflicting an estimated 90% of diabetics. The disease is especially devastating in low and middle income countries where two thirds of people living with diabetes have inadequate control and management of the disease due to limited continuous access to anti diabetic

O. Gervasi et al. (Eds.): ICCSA 2018, LNCS 10963, pp. 659–671, 2018.
https://doi.org/10.1007/978-3-319-95171-3_52

treatments such as insulin and limited access to quality professional health assistance [8]. In 2014 alone, approximately 4.9 million deaths have been attributed to diabetes related illnesses [8]. Late diagnosis of the disease makes it especially dangerous. 80% of complications related to type II diabetes can be prevented or delayed by early diagnosis or early identification of people at risk [4]. This demonstrates that the traditional methods for diagnosing and managing the disease are not sufficient to contain its spread.

The accelerating adoption of information systems technologies in the medical and health care sector has contributed to the increasing availability of data pertaining to diabetes; including symptoms, risk factors and socioeconomic data. These large datasets provide an opportunity for the application of data-driven approaches such as various Machine Learning (ML) techniques, to aid health professionals in making more accurate and timely diagnoses. Machine Learning (ML) techniques have been shown to be effective in quick and cost effective diagnosis. This is especially important in developing countries where the disease burden is high but health systems are generally poorly resourced.

The diagnosis of diabetes may be considered a classification problem and thus various machine learning techniques such as Artificial Neural Networks (ANNs), Logistic Regression (LR), and K-Nearest Neighbours (KNN) may be employed. These techniques are known to perform well in classification problems. ANNs are biologically inspired computer programs which simulate the way the human brain processes information [2]. ANNs have an information processing structure that is composed of a large number of interconnected processing units called neurons. ANNs use an activation function and a training algorithm such as the Back-Propagation algorithm to learn from the data. A Multi-Layer Perceptron (MLP) is a widely used ANN to solve diagnosis problems.

The KNN algorithm is a lazy-learning algorithm which classifies test instances based on their similarity with training instances in a feature space. KNN is regarded as a lazy learning technique because it stores training instances during learning and only uses them during classification. LR is a statistical regression technique that is used to model a relationship between one dependent variable and one or more independent variables. The dependent variable of the LR takes a categorical value to represent a binary outcome. This makes the algorithm more suitable to solve classification problems such as diabetes diagnosis.

In this paper, an improved Generalized Regression Neural Network (GRNN) is proposed to diagnose type II diabetes. The algorithm was improved by hybridizing it with an efficient K-Means clustering algorithm. The efficacy of the improved GRNN network was tested on the Pima Indian dataset including three additional datasets that were produced by preprocessing the Pima Indian dataset. The performance of the algorithm in these datasets was compared with other studies in literature.

This rest of the paper is organized as follows: Sect. 2 presents related work, Sect. 3 details the methodology and Sect. 4 gives the experimental results and discussion. Lastly, the conclusion and pointers to future work are given in Sect. 5.

2 Related Work

Artificial Neural Networks (ANNs) have been successfully used to replace conventional pattern recognition methods in disease diagnosis systems. Mehmet et al. [5] conducted a diabetes classification study using six different types of ANNs: Probabilistic Neural Network (PNN), Learning Vector Quantization (LVQ), Feed- Forward Networks (FFN), Cascade-Forward Networks (CFN), Distributed Time Delay Networks (DTDN) and Time Delay Networks (TDN)). They [5] also applied the artificial immune system and a Gini algorithm derived from decision tree algorithms for classification. Their studies used the Pima Indian diabetes dataset to train and test the algorithms. The best classification accuracy of 76% was achieved by the DTDN, followed by LVQ with a classification accuracy of 73%. The Gini algorithm produced the lowest classification accuracy of 66%. The remaining algorithms produced a classification accuracy between 68% and 72%.

Adeyemo and Akinwonmi [1], proposed the use of Generalized Regression Neural Network (GRNN) and the Probabilistic Neural Network to classify diabetes using patient data obtained from the Family Medicine Clinic of the Wesley Guild Unit at the University Teaching Hospital of Nigeria. Both networks used *tanh* as the activation function. Their GRNN achieved a classification accuracy of 84% while the PNN achieved an accuracy of 76%. Pradhan and Kumar Sahu [14] implemented a Multi Layer Back-propagation (MLBP) ANN with a single hidden layer with five neurons and a Genetic Algorithm (GA) for feature selection. They applied 10-fold cross validation for training and testing the network and achieved a classification accuracy of 72.2% with a Mean Square Error of 1.683.

Asha et al. [10] suggested that instead of gradient-based learning techniques for MLPBP networks, one may apply the commonly used optimization algorithms such as Genetic Algorithms, Particle Swarm Optimization (PSO), and Ant Colony Optimization (ACO) to determine the best network weights. They used a GA to initialize and optimize the connection weights of the MLBP network. The GA provided was used to determine various parameters such as the number of hidden layers, features to select, and efficient network learning rate. The GA also allowed the network to efficiently initialize the connection weights. Their study also identified features to be selected by applying a mixture of Decision Tree (DT) and GA-CFS (Correlation Feature Selection) algorithms. These algorithms (DT and GA-CFS) were used as input to the hybrid model of the MLPBP network and GA to classify diabetes. A classification accuracy of 84% was achieved by this hybrid of GA and MLPBP network.

Vijayan et al. [21] proposed the use of KNN, a K-Means clustering algorithm and an Amalgam KNN on Pima Indian dataset. The Amalgam algorithm was a hybrid of the KNN and K-Means clustering algorithm. Their KNN and K-Means clustering algorithm achieved classification accuracies of 73% and 77% respectively. The Amalgam KNN algorithm achieved an accuracy of 80%. The Amalgam KNN algorithm produced improved results by using K-Means clustering to identify and remove instances that were erroneously classified by the KNN.

KNN and K-Means clustering algorithms produced average results because the selected values of k and the dataset was not preprocessed to reduce noise.

Nai-Aruna and Moungmaia [13] used a hybrid of LR with ensemble methods. They selected Boosting and Bagging ensemble methods. Ensemble methods are designed to improve the stability and the accuracy of ML algorithms. Nai-Aruna and Moungmaia [13] applied the hybdridized LR algorithm on patient data from the Sawanpracharak Regional Hospital in Thailand. The hybrid of LR and Boosting achieved the classification accuracy of 82.308% while the hybrid of LR and Bagging achieved an accuracy of 82.318%. An accuracy of 82.308% was obtained with the standard LR algorithm. These results show that ensemble methods were not effective in significantly improving the classification accuracy of the LR model.

3 Methodology

3.1 Datasets and Data Preprocessing

The population for the dataset is the Pima Indian population in Phoenix, Arizona, USA but restricted to Pima Indian Females older than 20 years [17]. This population has been under study since 1965 by the National Institute of Diabetes and Digestive and Kidney Diseases due to the high incidence rate of diabetes. The Pima Indian females participated in standardized diabetes examinations and Type II diabetes was diagnosed using the WHO Type II diabetes diagnosis criteria [16]. A patient was considered diabetic when their two hour post-load plasma glucose was at least 200 mg/dl (11.1 mmol/l) at any survey examination [21]. The Pima Indian dataset has 768 observations with nine attributes. Five hundred of these patients were diabetic and each patient had only one record. The attributes in the dataset are given below and basic statistical properties of the dataset are given in Table 1.

1. Number of times pregnant (PREG)
2. Plasma Glucose Concentration at 2 h in an Oral Glucose Tolerance Test (GLUC)
3. Diastolic Blood Pressure in mm Hg (PRESS)
4. Triceps Skin Fold Thickness (mm) (SKIN)
5. 2-Hour Serum Insulin Uh/ml) (INSU)
6. Body Mass Index (Weight in kg/(Height in cm)) (BMI)
7. Diabetes Pedigree Function (PDF)
8. Age in years (AGE)
9. Diabetes Class Variable (0 for no diabetes or 1 for diabetes presence)

Raw data is susceptible to missing values, noise, and inconsistency [12]. These factors affect the quality of the dataset and consequently the performance of ML techniques [12]. To improve the quality of the dataset and ultimately the results of ML techniques, raw data is preprocessed. Data preprocessing deals with the preparation and transformation of the initial dataset before applying

Table 1. Statistical properties of the dataset

Feature	Minimum	Maximum	Mean	Variance	Standard deviation
Number of times pregnant	0.00	17.00	3.85	11.34	3.37
Plasma glucose concentration	0.00	199.00	120.89	1020.92	31.95
Diastolic blood pressure (mm Hg)	0.00	122.00	69.11	374.16	19.34
Triceps skin fold thickness (mm)	0.00	99.00	20.54	254.14	15.94
2-h serum insulin (Uh/ml)	0.00	846.00	79.80	13263.89	115.17
Body mass index (Weight in kg/(Height in in.))	0.00	67.10	31.99	62.08	7.88
Diabetes pedigree function	0.08	2.42	0.47	0.11	0.33
Age (years)	21.00	81.00	33.22	138.12	11.75

ML techniques. Data preprocessing techniques are divided into the following categories: Dataset Cleaning, Dataset Transformation, Dataset Reduction and Dataset Integration.

The Pima Indian dataset has approximately 10% of instances with attributes that have missing values. The Min-Max Normalization technique was employed to re-scale the values in the dataset to be in the same range [0,1] and various pre-processing techniques were applied to the original dataset to produce three additional datasets:

1. A dataset with all instances with missing values removed.
2. A dataset with missing attribute values replaced by the mean value of that attribute (except attributes: number of times pregnant and class variable).
3. A dataset with insignificant features that were excluded using a learner-based feature selection technique (Random Forest algorithm).

In this study, the original un-preprocessed dataset is referred to as A-[Unprocessed]. The dataset formed as a result of applying preprocessing method 1 is called B-[Excl Missing]. The dataset formed as a result of applying method 2 is called C-[Replaced by Mean]. Dataset D-[Extracted Features] is the dataset formed after applying feature extraction using the Random Forest algorithm.

Table 2 provides an overview of basic properties of these derived datasets including the original dataset.

Table 2. Properties of diabetes dataset A, B, C and D.

Measure	A-[Unprocessed]	B-[Excl Missing]	C-[Replaced by Mean]	D-[Extracted Features]
Number of instances	768	359	768	359
Number of positive instances	268	128	268	128
Number of negative instances	500	231	500	231
Number of instances with missing values	409	0	0	0

3.2 Improved Generalized Regression Neural Network

A Generalized Regression Neural Network (GRNN) is a special case of the Radial Basis Function Network that is based on kernel regression networks. A GRNN does not require an iterative training procedure as is the case for back propagation networks. It is known that the feedforward back-propagation network suffers from sensitivity to randomly assigned initial weights. A GRNN does not suffer from this problem because it uses the target value as a weight connection between pattern and summation layer neurons instead of using randomized weights. A GRNN provides an estimate of continuous variables by converging into either a linear or nonlinear regression surface. This type of artificial neural network (ANN) has a simple yet powerful structure of four layers: a input layer, a pattern layer, a summation layer, and an output layer. A GRNN estimates the output by using a weighted average of the outputs of the training dataset. An output weight is computed by using a distance measure between the training data and test data. If the distance between the training data and test data is small, then more weight is allocated to the output, otherwise less weight is allocated to the output. The activation function of the GRNN is given as follows:

$$y_i = \frac{\sum_{i=1}^{n} g_i . w_{ij}}{\sum_{i=1}^{n} g_i}$$

where: w_{ij} is the target output corresponding to input training vector x_i, $g_i = e^{-\frac{D_i^2}{2.\sigma^2}}$ is the output of a hidden layer neuron i. Here, $D_i^2 = (x - v_i)^T (x - v_i)$ is the squared distance between the input vector x and the training vector v. v_i is the training vector i at neuron i. σ is a constant controlling the size of the receptive region.

In a GRNN the distance D_i between the training sample and the test instance x is used to measure how well each training sample can represent the position of x. Figure 1 shows the structure of the proposed KGRNN network. This network has four layers. The first layer has nine neurons representing the nine attributes in the dataset (including the class label T). The hidden layer has 49 neurons representing 49 RBF activation functions with 49 centroids. The summation layer has two neurons that compute the weighted and unweighted sum of outputs from the hidden layer. Finally, the output neuron computes the final classification value.

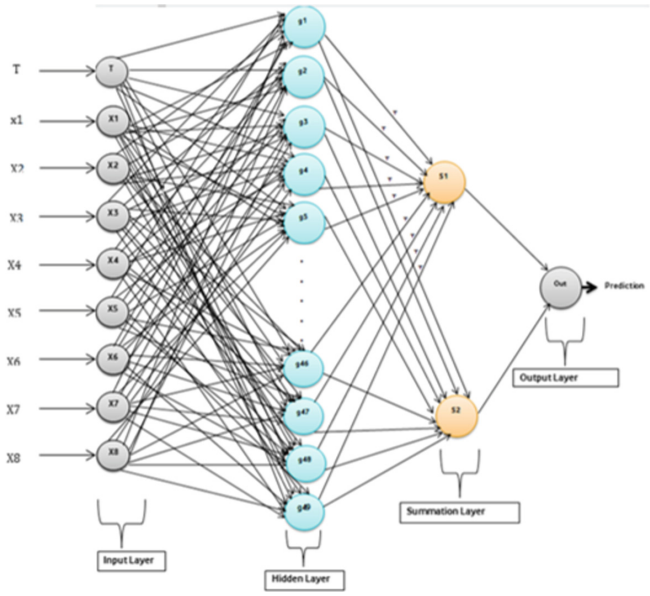

Fig. 1. KGRNN architecture

The proposed GRNN (KGRNN) uses an enhanced K-Means clustering technique to produce cluster centers (centroids) which are used as the training set for the network instead of all instances in the training set. The enhanced K-Means clustering technique (CVE-K-Means) finds quality clusters by searching for an optimal value of k within a pre-defined range of values. In order to obtain quality clusters, the CVE-K-Means algorithm also selects initial cluster centers by computing the distance between each training instance and the origin. The distance is then used to initially assign training instances to appropriate clusters based on their proximity to the origin. The pseudocode for KGRNN is given in Algorithm 1.

Algorithm 1. Pseudocode for KGRNN Algorithm

Data: A set D of n instances where $D = d_1, d_2, d_3, ..., d_i, .., d_n$ is a set of test examples. A set C of k-centroids (with $k < n$) produced by CVE-K-Means Algorithm using training set.

Result: A classification accuracy of the network given all instances from D as input

begin

Make each centroid C_i in the dataset C be the centre point for each neuron N_i in the hidden layer of GRNN network `/*` *number of neurons in the hidden layer is = number of centroids* `*/`

numClassified ← 0 `/*` *Counter of correctly classified test instances* `*/`

for *each instance* d_i *in* D **do**

input layer of the network receives d_i attributes as input

input layer passes inputs to all neurons in the hidden layer

Hidden layer receives the input from input layer

for *each neuron in the hidden layer* **do**

D_i ← distance(input,C_i)

Gaussian activation function $g_i = e^{-\frac{D_i^2}{2.\sigma^2}}$

Transfer the result of the activation function as input to neurons in the summation layer

Summation Layer

numerator ← $\sum_{i=1}^{n}(weight_i * GaussianOutput_i)$

denominator ← $\sum_{i=1}^{n}(GaussianOutput_i)$

Output Layer

Output ← numerator/denominator

if *result is less than 0.5* **then**

classification = 0

else

classification = 1

if *classification matches class label of* d_i **then**

increment numClassified

`/* Finish classifying given test instances */`

Overall accuracy = numClassified / size of D

Return classification accuracy of KGRNN

4 Results and Discussion

The KGRNN was implemented in the Java programming language on a machine with an Intel i5 vPro processor running at 2.30 GHz, with 8 GB RAM and using the JDK 1.8 software. Feature extraction using the Random Forest algorithm was implemented in the Weka Machine Learning tool [9]. Five standard performance measures were used in this study to evaluate the KGRNN. These performance measures are: Classification accuracy, Sensitivity, Specificity, Positive Predictive

Value (PPV), and Negative Predictive Value (NPV). The proposed KGRNN network was configured with 49 neurons in the pattern layer and 0.19 was used as the smoothing factor σ. the forty-nine centroids that were produced by CVE-K-Means algorithm were used as a training set.

With this configuration, the network achieved the highest classification accuracy of 86% with 83% Sensitivity and 87% Specificity using the dataset D-[Extracted Features]. The second highest classification accuracy of 85% was achieved on dataset B-[Excl Missing] using $\sigma = 0.08$ and 49 centroids as training instances. The network performed poorly on two datasets A-[Unprocessed] and C-[Replaced by Mean]. It achieved the highest classification accuracy of 66% on dataset A-[Unprocessed] using 49 centroids and $\sigma = 0.09$. On dataset C-[Replaced by Mean] it achieved the highest classification accuracy of 71% using 49 centroids as training set and $\sigma = 0.05$.

Figure 2 gives the performance results for KGRNN technique using 49 training centroids and different values of σ. The influence of different values of the smoothing factor across the four datasets was also investigated. The results of this investigation are given in Fig. 3.

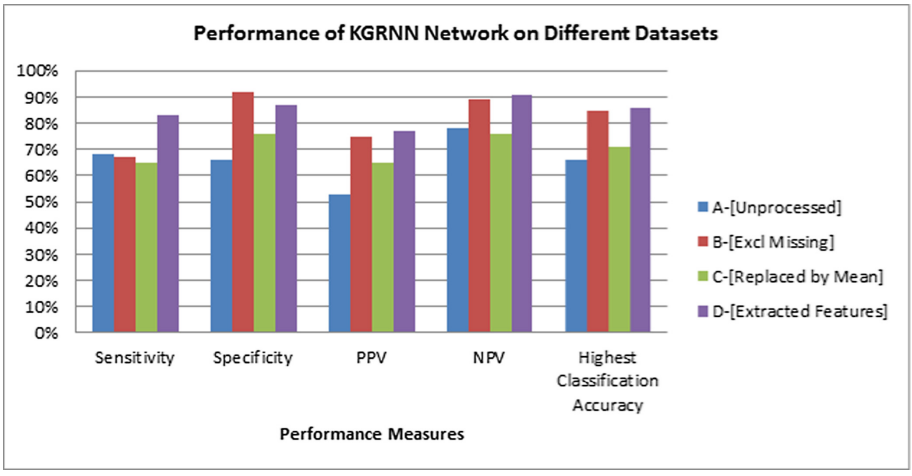

Fig. 2. Performance measures for KGRNN network.

The impact of using centroids as the training set for the KGRNN on its execution time was evaluated by recording the time the network took to train and classify diabetes as the number of centroids used increased. The performance results in Fig. 4 show that as the number of centroids increased from 5 to 140, the time it took the network to train and classify test cases also steadily increased. The results also show that it takes the network even longer to train and perform classification when data is not preprocessed.

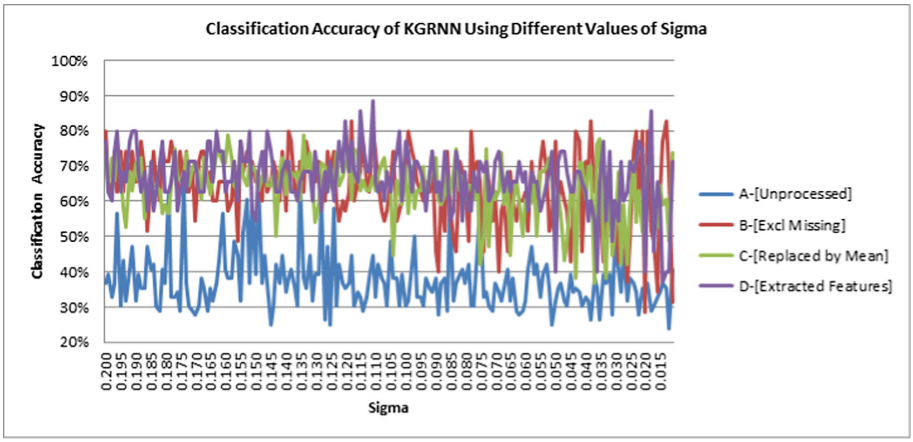

Fig. 3. KGRNN accuracy using different values of sigma and datasets.

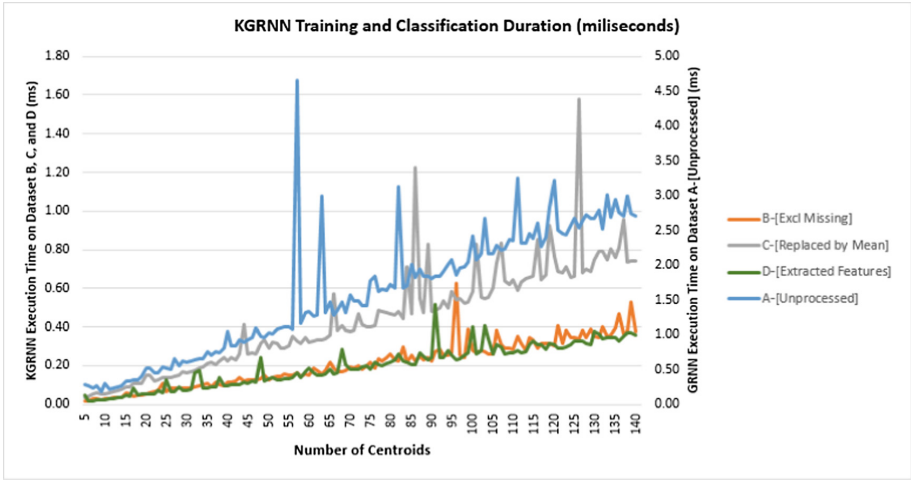

Fig. 4. KGRNN execution time for different number of centroids.

The ROC curve with AUC of 87% for KGRNN is given in Fig. 5. The ROC curve was plotted by training and testing the network using dataset D-[Extracted Features], 49 centroids as the training set and $\sigma = 1.9$. The KGRNN technique outperformed by 1% the standard GRNN technique proposed by Alby and Shivakumar [3]. The GRNN proposed by [3] obtained the highest diabetes classification accuracy known in literature for a GRNN. Table 3 compares the performance of KGRNN to other artificial neural networks in the literature.

The performance results show that the selection of relevant features using feature extraction techniques positively contributed towards the network's improved results by enhancing the quality of the dataset. The network struggled to obtain

Table 3. Accuracy comparison of KGRNN network with other ANNs

Year	Reference	Algorithm	Highest accuracy	σ
2017	This study	KGRNN	86%	0.19
2017	[3]	GRNN	85%	Not Reported
2017	[15]	RBM ANN	85%	-
2011	[1]	GRNN	84%	Not Reported
2016	[20]	CNN	83%	-
2016	[18]	PNN	81%	-
2003	[11]	GRNN	80%	2.5
2016	[7]	GA-MLPNN	79%	-
2017	[6]	RBFN	70%	-
2014	[19]	VGRNN	57%	Not reported

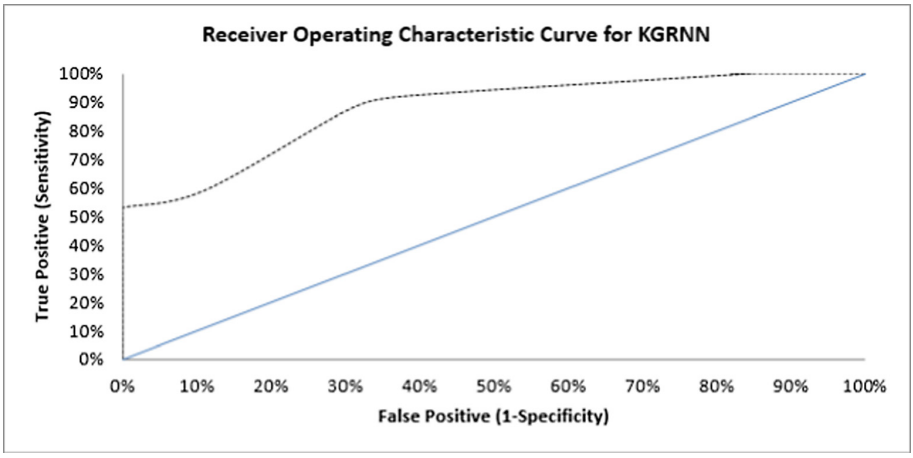

Fig. 5. ROC curve for KGRNN network

good results on dataset A-[Unprocessed] and dataset C-[Replaced by Mean]. This suggests that the network is sensitive to noise and varying attribute scales in the data. The utilization of cluster centers as the training set for the KGRNN network ensured that the network obtained improved results in less computation time compared to utilizing the entire training set. The computation time of the standard GRNN significantly increases as the number of training instances increases. This inefficiency of GRNN is effectively addressed by KGRNN.

5 Conclusion and Future Work

In this study, the diabetes diagnosis problem was addressed using an improved Generalized Regression Neural Network. The network employed the Gaussian

function as its activation function during training. To improve the computational time of the network, an improved K-Means clustering technique was employed to produce quality cluster centers that were used by the network as a training set. KGRNN was able to accurately identify 83% of diabetic individuals and 87% of non-diabetic individuals during testing. Data preprocessing proved to be another significant contributor in the good performance of KGRNN. The experimental results showed that type II diabetes can be efficiently and quickly diagnosed using KGRNN.

Future work could include employing different data preprocessing techniques and different algorithm parameters. To determine the efficacy of the proposed algorithm, it would be beneficial to apply it to other benchmark problems in the future.

References

1. Adeyemo, A., Akinwonmi, A.: On the diagnosis of diabetes mellitus using artificial neural network models **4** (2011)
2. Agatonovic-Kustrin, B.: Basic concepts of artificial neural network (ANN) modeling and its application in pharmaceutical research. J. Pharm. Biomed. Anal. **22**(5) (2000)
3. Alby, S., Shivakumar, B.: A prediction model for type 2 diabetes using adaptive neuro-fuzzy interface. J. Biomed. Res., S69–S74 (2018). Special Issue
4. Barakat, N.H., Bradley, A.P., Barakat, M.N.H.: Intelligible support vector machines for diagnosis of diabetes mellitus. Trans. Info. Tech. Biomed. **14**(4), 1114–1120 (2010). https://doi.org/10.1109/TITB.2009.2039485
5. Bozkurt, M., Yurtay, N., Yilmaz, Z., Sertkaya, C.: Comparison of difierent methods for determining diabetes **22**, 1044–1055 (2014)
6. Cheruku, R., Edla, D.R., Kuppili, V.: Diabetes classification using radial basis function network by combining cluster validity index and bat optimization with novel fitness function. Int. J. Comput. Intell. Syst. **10**, 247–265 (2017)
7. Choubey, D.K.: GA-MLP NN: a hybrid intelligent system for diabetes disease diagnosis. Int. J. Intell. Syst. Appl. **1**, 49–59 (2016)
8. Indian Dental Association 6th Edition Committee: Follow-up to the political declaration of the high-level meeting of the General Assembly on the prevention and control of non-communicable diseases, 6th edn. IDF (2013)
9. Frank, E., Hall, M.A., Witten, I.H.: The WEKA Workbench. Online Appendix for Data Mining: Practical Machine Learning Tools and Techniques, 4th edn. Morgan Kaufmann (2016)
10. Karegowda, A.G., Manjunath, A., Jayaram, M.: Application of genetic algorithm optimized neural network connection weights for medical diagnosis of pima Indians diabetes. Int. J. Soft Comput. (IJSC) **2** (2011)
11. Kayaer, K., Yildirim, T.: Medical diagnosis on pima Indian diabetes using general regression neural networks. In: Proceedings of the International Conference on Artificial Neural Networks and Neural Information Processing (2003)
12. Kotsiantis, S.B., Kanellopoulos, D., Pintelas, P.E.: Data preprocessing for supervised leaning. Int. J. Comput. Sci. **1**(2) (2006)
13. Nai-Aruna, N., Moungmaia, R.: Comparison of classifiers for the risk of diabetes prediction. In: 7th International Conference on Advances in Information Technology, vol. 69, pp. 132–142 (2015)

14. Pradhan, M., Sahu, R.K.: Predict the onset of diabetes disease using artificial neural network (ANN). Int. J. Comput. Sci. Emerg. Technol. **2** (2011)
15. Ramesh, S., Balaji, H.: Optimal predictive analytics of pima diabetics using deep learning. Int. J. Database Theor. Appl. **10**, 47–62 (2017)
16. Shanker, M., Hu, M., Hung, M.: Estimating probabilities of diabetes mellitus, using neural networks. SAR QSAR Environ. Res. **11**(2), 133–147 (2000)
17. Smith, J., Everhart, J., Dickson, W., Knowler, W., Johannes, R.: Using the ADAP learning algorithm to forecast the onset of diabetes mellitus. In: Symposium on Computer Applications and Medical Care, pp. 261–265. IEEE Computer Society Press (1988)
18. Soltani, Z.: A new artificial neural networks approach for diagnosing diabetes disease type II. Int. J. Adv. Comput. Sci. Appl. **7** (2016)
19. Sujatha, V.: An intelligent expert based system neural network for the diagnosis of type2 diabetes patient. Int. J. Innovative Res. Adv. Eng. **1**(2) (2014)
20. Tharani, S., Yamini, C.: Classification using convolutional neural network for heart and diabetics datasets. Int. J. Adv. Res. Comput. Commun. Eng. **5**(12) (2016)
21. Vijayan, V., Ravikumar, A., VeluC., M., Kashwan, K.R., Karegowda, A.G., Cios, K.J., Moore, G.W.: Study of data mining algorithms for prediction and diagnosis of diabetes mellitus (2014)
22. WHO: Global health risks global health risks who mortality and burden of disease attributable to selected major risks. World Health Organization (2009)