Int. J. Nonlinear Anal. Appl. 15 (2024) 2, 39–46
ISSN: 2008-6822 (electronic)
http://dx.doi.org/10.22075/ijnaa.2022.29575.4196

Hybrid of particle swarm optimization algorithm and fuzzy system for diabetes diagnosis

Reza Ghabousian[a], Yousef Farhang[b,*], Kambiz Majidzadeh[a], Amin Babazadeh Sangarh[a]

[a]*Department of Computer Engineering, Urmia Branch, Islamic Azad University, Urmia, Iran*

[b]*Department of Computer Engineering, Khoy Branch, Islamic Azad University, Khoy, Iran*

(Communicated by Asadollah Aghajani)

Abstract

Diabetes is a dangerous disease in which the body is incapable of controlling blood sugar due to inadequate insulin hormone levels. This chronic disease increases blood sugar in patients. Therefore, if it is not controlled, it will cause many complications. A considerable number of people in the world suffer from this disease owing to its damage and lack of its initial diagnosis. The patient visits the doctor frequently to diagnose his/her illness and conducts various tests that are boring and costly. Increasing machine learning approaches through heuristics, and novel methods can somewhat decrease the problems. The current study aims to propose a model that can predict diabetes in patients with high accuracy. The paper introduces a new method based on the assortment of metaheuristic algorithms of a particle swarm and fuzzy inference system. The proposed method utilizes fuzzy systems to binary the particle swarm algorithm. The achieved model is applied to the diabetes dataset and then evaluated using a neural network classifier. The results indicate an increase in classification accuracy to 95.47% compared to other existing methods.

Keywords: Diabetes, PSO algorithm, neural networks, fuzzy systems, meta-heuristic algorithms
2020 MSC: 68Txx

1 Introduction

As one of the most prevalent diseases, diabetes results in disability and death worldwide, particularly in developing countries. The number of people with the disease is estimated to rise by 48% to 629 million by 2045 [5]. However, diabetes is mostly preventable and can be prevented by altering lifestyles. The change can also reduce the probability of developing heart disease and cancer. Thus, there is a dire need for a prognostic tool that can help physicians in the early diagnosis of this disease. As a result, doctors will be able to advise the lifestyle changes needed to prevent the progression of the fatal disease [3]. In recent decades, artificial intelligence methods such as fuzzy logic, artificial neural networks, genetic programming, the combination of neural and fuzzy systems, and regression have been used in dealing with many issues. The disease in question is of three types. Type 1 diabetes, also known as insulin-dependent diabetes, is mostly to be seen in children. Type 2 diabetes or non-insulin-dependent diabetes occurs for 90 to 95 percent of diabetics and type III diabetes, which is mostly observed in pregnant women, turns into type II diabetes after pregnancy [2].

*Corresponding author
Email addresses: ghabousian.reza@gmail.com (Reza Ghabousian), yfarhang@yahoo.com (Yousef Farhang), k.majidzadeh@iaurmia.ac.ir (Kambiz Majidzadeh), bsamin2@liveutm.onmicrosoft.com (Amin Babazadeh Sangarh)

Received: July 2022 *Accepted:* December 2022

In this study, a new model is proposed for forecasting type 2 diabetes based on data mining technique. The combined Particle Swarm Optimization (PSO) and Fuzzy system are used to evaluate a set of medical data relating to a diabetes diagnosis challenge.

2 Review of literature

Currently, factors such as environmental conditions, sedentary lifestyle, and genetics cause certain diseases, the most famous of which is known as ”diabetes”, which has become the leading chronic disease. The main goal of this work is to provide an efficient diagnostic tool for the diagnosis of diabetes, even though there are several existing techniques that have been used to diagnose diabetes. Diabetes is a disease that increases blood glucose known as hyperglycemia to a degree that affects the body to a great extent. The main reason for the symptoms of hyperglycemia is the lack of insulin, in which the production of insulin in the pancreas by beta cells fails.

Sisodia et al. applied three classification algorithms namely SVM, Decision Tree, and Naïve Bayes to devise a model that could provide the highest accuracy. They assessed their proposed method on the PIDD dataset. The researchers used accuracy criteria, F and Recall criteria, and ROC chart analysis in evaluating the results. The results demonstrated that the Naïve Bayes classifier has higher accuracy than other classifiers [12]. Khanam et al have utilized data mining, machine learning (ML), and neural network (NN) algorithms on Pima Indian Diabetes (PID) dataset in their studies. Their researches indicate that the model works well in predicting diabetes by logistic regression (LR) and support vector machine (SVM). It also provided the NN model with two hidden layers of 88.6% accuracy [9]. Hasan et al. have presented a framework for predicting diabetes in which outlier rejection, filling the missing values, data standardization, feature selection, K-fold cross-validation, and different Machine Learning (ML) classifiers (k-nearest Neighbor, Decision Trees, Random Forest, AdaBoost, Naive Bayes, and XGBoost) and Multilayer Perceptron (MLP) have been used. The weight composition of various ML models has been suggested to enhance the prediction of diabetes. A ROC curve (receiver operating characteristic curve) was applied as a performance criterion and all experiments were carried out using the Pima diabetes dataset. Results of the proposed model based on the criteria namely sensitivity, specificity, false omission rate, diagnostic odds ratio, and AUC stand at 0.789, 0.934, 0.092, 66.234, and 0.950, respectively [7]. Singh et al. have put forward a method named eDiaPredict in which a group of algorithms including XGBoost, Random Forest, Support Vector Machine, Neural Network, and Decision tree are used. The results of their model were evaluated through the criteria of sensitivity, specificity, Gini Index, precision, the area under the curve, the area under the convex hull, minimum error rate, and minimum weighted coefficient. The results of 95% accuracy were obtained on the Pima diabetes dataset [11]. Pradhan et al. provided a method for effective enhancement of Naïve Bayes, decision tree, and vector machine techniques to predict diabetes. In their proposed model, an analysis of collected data from diabetic patients including a list of factors causing diabetes (more affected age groups, work style, and food habits) based on an artificial neural networks algorithm is presented. The results on the Pima dataset show test and accuracy of 85.09% [10]. Kannadasan et al. have proposed a method based on deep neural networks. In the recommended method, autoencoders stacked encoders on diabetes data have been used. These features are extracted from the dataset by means of automated stack encoders, and the datasets are classified by the softmax layer. To evaluate the proposed model, some criteria such as precision, recall, specificity, and F1-score have been used [8]. In order to diagnose type 2 diabetes, PK Singh and Mukesh used the basic notions of soft computation and neural networks in their research using a hybrid binary classification model. They achieved 85 percent of the diagnosis by a combined neural network [6]. Bozkurt et al. used six various neural networks including PNN, LVQ, FFN, CFN, DTDN, and TDN to diagnose type 2 diabetes [4]. Afroz et al. used the decision tree to classify type 2 diabetes patients by studying 1506 datasets on the artificial immune system and the Gini algorithm. They applied varied techniques in the research by comparative study. The results indicated that mortality has a direct relationship with age and rates of cardiovascular disease have an impact on type 2 diabetes [1].

3 Proposed method

In this study, the proposed method is based on selecting effective characteristics for the classification of diabetic and non-diabetic patients. In the method, a binary particle swarm metaheuristic algorithm is used to select characteristics. A fuzzy system is applied to binary this algorithm. To evaluate the obtained results, a neural network classifier is utilized. The proposed method aims to find a model for the classification of diabetic patients. To achieve the desired objective, a new method based on the particle swarm algorithm is presented in which fuzzy systems are used to binary the algorithm. The proposed method is shown in the first part of the flowchart and the details of the method are also presented. The flowchart of the proposed method is shown in the figure 1.

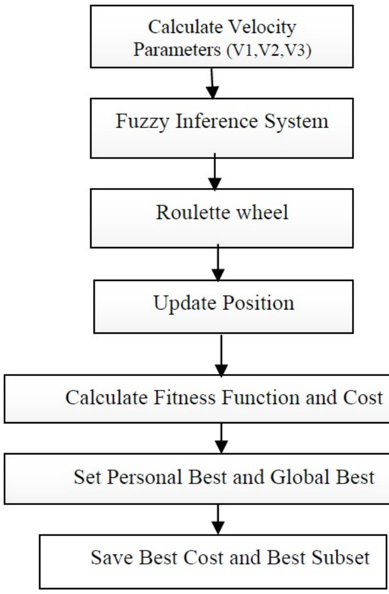

Figure 1: The flowchart of the proposed method

3.1 Data normalization

Normalizing data could be carried out by some varied approaches. The researchers have applied the Min-Max normalization method to this end. In this method, assuming set A in the interval $[A_{\min}, A_{\max}]$, we want to convert it to the new interval $[New_{\min}, New_{\max}]$. For this purpose, each initial value such as V will be converted into a new value V' in the new interval. The normalization based on this method is given in the relation.

$$\frac{V-(A_{\min})}{(A_{\max})-(A_{\min})} = \frac{V'-(New_{\min})}{(New_{\max}-New_{\min})} \tag{3.1}$$

3.2 Bainery particle swarm optimization

Particle Swarm Optimization (PSO) is based on the principle that any particle can refer to a solution in the swarm. Any swarm has a position and the fitness value related to that will be studied and optimized by using the proportion function. The movement of particles from one position to another depends on their latest velocity vector. This velocity vector will be determined by taking into account the performance of the particle itself and other particles and by using the best positions of the particle in a swarm. For N particles, $x_i^t = (x_{i1}^t, x_{i2}^t, \ldots, x_{ij}^t, \ldots, x_{ik}^t)$ is given in which x_i^t represents particle i in the time period t with the number of features K. The particle vector with the number of K attributes in which each property is $x_{ij}^t \in \{0,1\}$. Furthermore, K represents the number of whole attributes in the main dataset. The number of subset features is $k \leq K$. Therefore, the particle swarm at movement t could be shown as $(x_1^t, x_2^t, ..., x_N^t)$. Also, the best general position of G^t, which represents the best position for the whole particle, is shown as $G = (g_1^t, g_2^t, ..., g_j^t, ..., g_k^t)$ and $g_j^t \in \{0,1\}$. For each particle, the best position ever will be as $P_i^t = (p_{i1}^t, p_{i2}^t, ..., p_{ij}^t, ..., p_{ik}^t)$. The velocity of the particle according to the relevance is updated as follows.

$$V_i^{t+1} = w * v_i^t + c_1 * r_1(P_i^t - X_i^t) + c_2 * r_2(G^t - X_i^t) \tag{3.2}$$

in the above formula, $V_i^t = (v_{i1}^t, v_{i2}^t, ..., v_{ij}^t, ..., v_{ik}^t)$, represents the velocity vector in the previous repetition, w represents internal weight, c_1 represents the weight factor for the best local solution, c_2 represents weight factor for the best general solution and r_1 and r_2 represent random numbers that are placed between [0,1]. The weights $c_1 c_2$ represent learning rates that indicate how much inertia weight can affect the memory of the new velocity. The inertia weight is updated according to the below equation:

$$W^{t+1} = \text{Velocity}_{\max} - \frac{(\text{Velocity}_{\max} - \text{Velocity}_{\min})t}{T} \tag{3.3}$$

In the above equation, t represents repetition t and T represents the number of repetitions of the algorithm PSO. Unler and Murat [13] in their proposed binary PSO method, have first transferred the velocity vector to the probability vector through the following Sigmoid function:

$$S_{ij}^t = \frac{1}{1+e^{-v_{ij}^t}} \tag{3.4}$$

in the above function, S_{ij}^t represents the jth probability of a feature in x_i^t. Then, the particle status in binary PSO is updated as follows:

$$x_{ij}^t = \begin{cases} 1, & \text{if } \delta < S_{ij}^t; \\ 0, & \text{otherwise.} \end{cases} \qquad j = 1, 2, ..., k \tag{3.5}$$

in the above figure, δ represents the random number between 0 and 1.

3.3 Evaluating function

In this section, features are calculated in each subset using mutual information as an evaluating function of relevance and redundancy. Calculating the relevance between features and class is presented according to the relevance.

$$R_1 = \sum_{x \in X} I(x; c) \tag{3.6}$$

in the above figure, X denotes the selected subset while C denotes the class label. Moreover, $I(x; c)$ calculates the amount of mutual information of the feature x with c. For each feature vector, redundancy is obtained based on mutual information from the relevance.

$$R_2 = \sum_{i=1}^{n} \sum_{j=i}^{n} I(x_i; x_j) \tag{3.7}$$

after that, the R_1 and R_2 are the achieved value of the evaluating function from the relevance obtained.

$$\text{fitness}_1 = R_1 / R_2 \tag{3.8}$$

3.4 Designing a fuzzy system based on fuzzy inference system

A new method for the BPSO algorithm named BPSO-Fuzzy is presented. In the method, instead of using the Sigmoid function to binarize the feature vector in the next iterations, a fuzzy inference system is designed, which based upon the three parameters used in the calculation of particle velocity, the feature selection probability is attained in a fuzzy form. Based on the probability obtained and Roulette Wheel algorithm, the results of the proposed method on the dataset investigated by this paper and the standard dataset illustrate that the proposed method has high classification capability. For each set of X, the membership function of the X set is a function of X relative to the interval [1,0]. The definition of a membership function is as follows [5].

$$\mu_A(x) = \begin{cases} 0, & x \in A \\ 0, & x \notin A \end{cases} \tag{3.9}$$

To calculate the movements of the particles in PSO, the equation (3.4) is applied. The calculated value is derived from three vectors. The result of multiplying the inertia weight at the particle velocity in the previous motion shows the position of the particle relative to the best position it has ever had and also the position of the particle relative to the best general position among all particles in the current repetition. The three values are entered as inputs to the fuzzy inference system. For three values V_1, V_2 and V_3 in the fuzzy inference system, membership functions have been presented. In fact, the membership functions in the input fuzzification section are tools or mappings that map numbers from the set of real numbers to the interval [0,1]. The functions are selected according to the problem and conditions of the system under study.

$$\begin{aligned} v_1^{t+1} &= w * v_i^t \\ v_2^{t+1} &= c_1 * r_1 (P_i^t - X_i^t) \\ v_3^{t+1} &= c_2 * r_2 (g^t - X_i^t) \end{aligned} \tag{3.10}$$

The functions used to fuzzy the values of these three inputs are the PI-shaped function. The function has been selected based on experience. The variation range of inputs has been determined in the range of [-10,10] and the severity of changes in each input is controlled using the min-max function in the implementation stage. Two PI-shaped membership functions are considered for each input to determine the fuzzy values of inputs according to the positive and negative input values.

Figure 2 shows an overview of the definition part of the membership function.

Figure 3 PI-shaped membership function defined for the inputs

After defining membership functions for all three inputs, the output of the fuzzy inference system is also defined. The output represents the probability based on which we will decide whether or not to select the feature. For the output of this system, the functions of the PI-shaped and triangular membership functions are used. Four membership functions for the output with 'high', 'ALhigh', 'Allow' and 'low' indicating respectively 'the high amount', 'a little high', 'a little low', and 'low' have been determined.

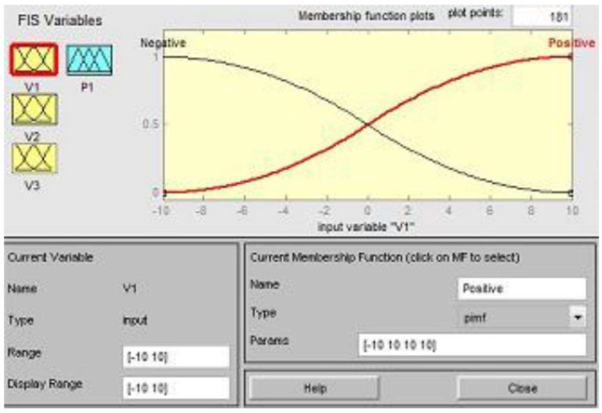

Figure 2: illustrates the output status of the fuzzy inference system

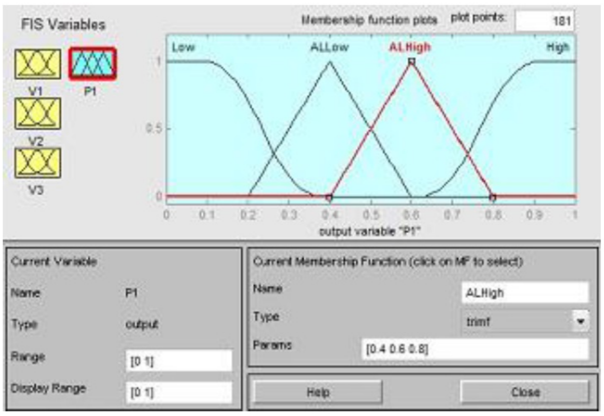

Figure 3: illustrates the output status of the fuzzy inference system.

The triangular membership function is shown as following.

$$\mu_A(x) = \begin{cases} 0, & x \leq a \\ \frac{x-a}{m-a}, & a < x \leq m \\ \frac{b-x}{b-m}, & m < x \leq b \\ 0, & x \geq b \end{cases} \tag{3.11}$$

The above formula could be also presented as following.

$$f(x; a, b, c) = \max\left(\min\left(\frac{x-a}{b-a}, \frac{c-x}{c-b}\right), 0\right). \tag{3.12}$$

The membership functions designed for the output of the system define the defuzzification operation by performing a mapping from the fuzzy set B, in $V \subset R$, which is the output of the fuzzy inference engine, to a definite point $y^* \in V$.

Up to this point, we entered the parameters of particle movement velocity as inputs with real values. The system converts the real values to fuzzy ones based on the relationship. It also converts the actual value between [0,1] which is the probability value by inferring the rules we have defined by non-fuzzy in accordance with the relationship. The value of this probability will determine the status of the feature selection. Fuzzy rules can usually be extracted according to the experiences of the human operator or the theoretical model. These rules, expressed as if-then rules, actually formulate the necessary conditions for decision-making to be graded by fuzzy logic and are presented in the following general form.

$$\text{If x is } A_i \text{ and y is } B_i \text{ then z is } C_{ij} \tag{3.13}$$

Each of these rules can be converted into a fuzzy R relationship. In the above rule, the A_i, B_i, C_{ij} are known as linguistic variables. In this stage, considering the number of inputs, eight rules are defined by the researchers. These rules are defined based on input and output membership functions. Based on defined rules, inputs converted to fuzzy numbers are then converted to non-fuzzy values. The obtained output was converted to a binary value of one or zero using a Roulette Wheel. This process was conducted for all the features in each particle. This makes new alternatives of features according to the evaluating function. The output of the fuzzy inference system is in accordance with the following formula:

$$S_{ij}^t = \text{evaluate } FIS(v_1, v_2 v_3) \tag{3.14}$$

To evaluate membership functions and defined rules for inputs and outputs, the resulting level can be considered in two-dimensional form for inputs. For all three inputs in the figure 4 the levels of change are illustrated in terms of rules and inputs.

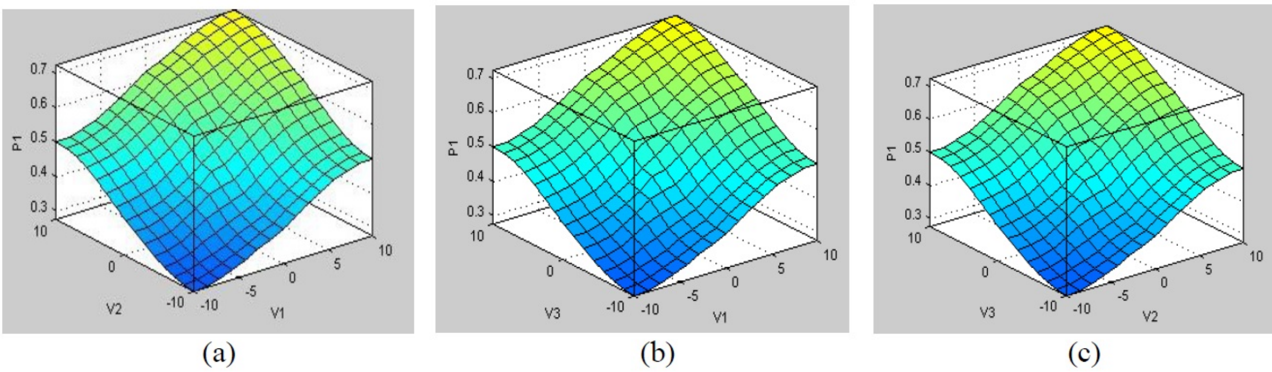

Figure 4: The output level indicates how the output changes are defined in terms of inputs and rules. a: For two input values V_1, V_2. b: For two input values V_1, V_3. c: For two input values V_2 and V_3.

3.5 Implementation and application

MATLAB software was used to implement the proposed method. First, the data were normalized by coding in the MATLAB environment. Then, the PSO algorithm was implemented in MATLAB. Also, MATLAB fuzzy toolbox section was used to design and implement a fuzzy inference system. To calculate the accuracy of the classification by the neural network, the nprtool was utilized. In order to implement diverse parts of the process, several functions including the main function of the algorithm, evaluation function, and functions related to the calculation of classification accuracy for classifications were written by the researchers of the study.

3.6 Evaluation and comparison

In the proposed method, a fuzzy inference system is used to update the position of the particles. Indians Pima dataset: PID Diabetes available in UCI Machine Learning data repository is used [3]. This database contains 768 records collected from Indian women, of which 500 were healthy women while 268 were type 1 diabetic. The minimum and maximum ages of the examined people are 21 and 81 years, respectively. Table 1 demonstrates 8 diagnostic factors for each person.

The proposed method has been applied a number of times to this dataset.

Also, instead of the fuzzy part in the method, the Sigmoid function has been used. In both cases, the algorithm was able to select 6 properties as an optimal subset. The number of chosen particles was 40 and the number of repetitions

Table 1: Database Properties for Diabetes

Dataset	feature	sample	class
pima-indians-diabetes	10	785	2

Table 2: Risk of Diagnostic Factors of Diabetes

Number	Attribute Name
1	Number of Times Pregnant
2	Plasma Glucose Concentration
3	Diastolic Blood Pressure
4	Triceps Skinfold Thickness
5	2-hour Serum Insulin
6	Body Mass Index (BMI)
7	Diabetes Record
8	Age

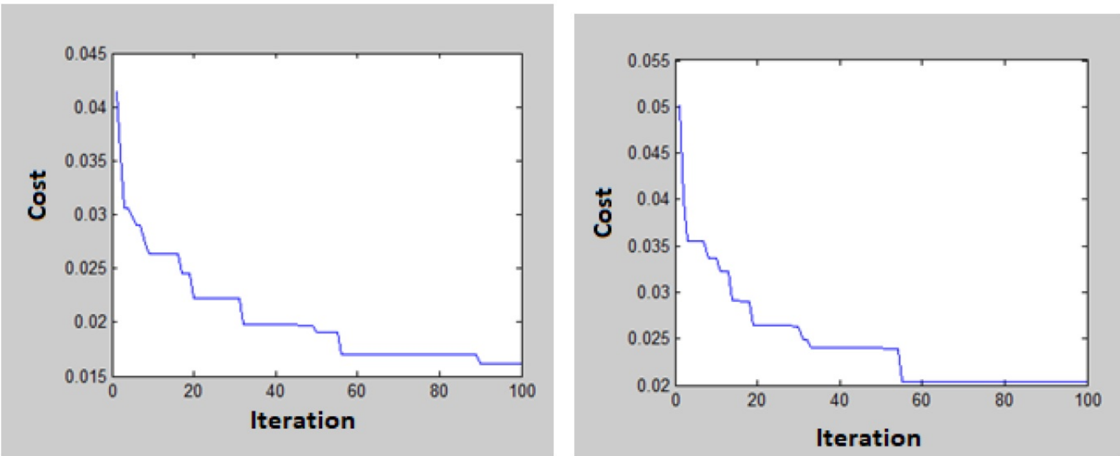

Figure 5: a) Convergence Status by Selecting 6 Features by Fuzzy PSO Method. b) Convergence Status by Selecting 6 Features by Binary PSO Method

was 100. In the proposed method, the convergence speed was higher than in other methods. The property is shown in table 3.

The neural network is one of the classifiers used in the current study. The neural network designed to train and test diabetes data includes two layers, an output layer and a hidden layer with 4 neurons. The results of the proposed method and its comparison with other methods are presented in the table 3 and figure 6.

Table 3: Results of Proposed Method and Comparison with Other Methods on Diabetes Data

Method	Data Selection Method	Features	Accuracy	Error Ratio	Recall	Precision
PSO Binary	Leave-one-out	6	0.9203	0.0780	0.8909	0.93
PSO Fuzzy	BootStrap	6	0.9509	0.0480	0.9215	0.9519
PSO Fuzzy	30% Test and 70% Train	6	0.9547	0.0475	0.9269	0.96
PSO Fuzzy	Fold-10	6	0.9532	0.0478	0.9241	0.9589

4 Conclusion

To diagnose diabetic patients, a model based on a metaheuristic algorithm for particle swarm optimization is presented. In the proposed model, the number of features in order to increase the predictive speed was decreased. This increases the speed and accuracy of classification. The binary version of the particle swarm optimization algorithm has been used, except that a fuzzy system is utilized to binary the algorithm. A neural network classifier was used to evaluate the proposed model and its accuracy reached 95.47%. The proposed model enjoys higher accuracy and speed compared to other methods.

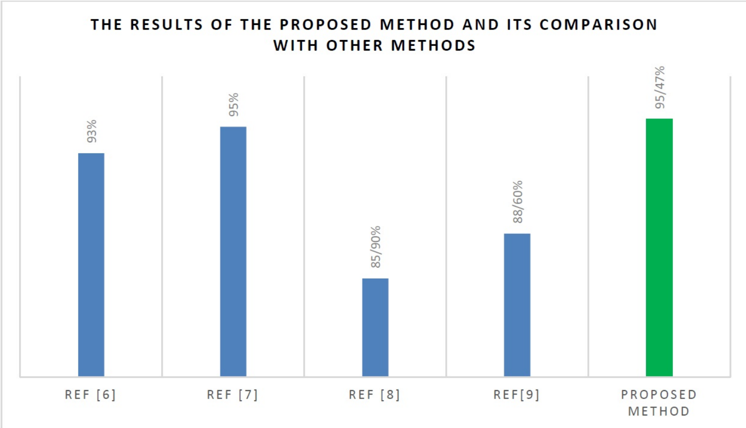

Figure 6: The results of the proposed method and its comparison with other methods.

References

[1] A. Afroz, M.J. Alramadan, M.N. Hossain, L. Romero, K. Alam, D.J. Magliano and B. Billah, *Cost-of-illness of type 2 diabetes mellitus in low and lower-middle income countries: a systematic review*, BMC Health Serv. Res. **18** (2018), no. 1, 1–10.

[2] American Diabetes Association, *Diagnosis and classification of diabetes mellitus*, Diabetes care **37** (2014), no. Supplement_1, S81–S90.

[3] F. Arroyave, D. Montaño and F. Lizcano, *Diabetes mellitus is a chronic disease that can benefit from therapy with induced pluripotent stem cells*, Int. J. Molecular Sci. **21** (2020), no. 22, 8685.

[4] M.R. Bozkurt, N. Yurtay, Z. Yilmaz and C. Sertkaya, *Comparison of different methods for determining diabetes*, Turkish J. Electr. Eng. Comput. Sci. **22** (2014), no. 4, 1044–1055.

[5] N.G. Forouhi and N.J. Wareham, *Epidemiology of diabetes*, Med. **47** (2019), no. 1, 22–27.

[6] D. Gupta, A. Choudhury, U. Gupta, P. Singh and M. Prasad, *Computational approach to clinical diagnosis of diabetes disease: a comparative study*, Multimedia Tools Appl. **80** (2021), no. 20, 30091–30116.

[7] M.K. Hasan, M.A. Alam, D. Das, E. Hossain and M. Hasan, *Diabetes prediction using ensembling of different machine learning classifiers*, IEEE Access **8** (2020), 76516–76531.

[8] K. Kannadasan, D.R. Edla and V. Kuppili, *Type 2 diabetes data classification using stacked autoencoders in deep neural networks*, Clinic. Epidemiol. Glob. Health **7** (2019), no. 4, 530–535.

[9] J.J. Khanam and S.Y. Foo, *A comparison of machine learning algorithms for diabetes prediction*, ICT Express **7** (2021), no. 4, 432–439.

[10] N. Pradhan, G. Rani, V.S. Dhaka and R.C. Poonia, *Diabetes prediction using artificial neural network*, Deep Learning Techniques for Biomedical and Health Informatics, Academic Press, 2020.

[11] A. Singh, A. Dhillon, N. Kumar, M.S. Hossain, G. Muhammad and M. Kumar, *eDiaPredict: An ensemble-based framework for diabetes prediction*, ACM Trans. Multimedia Comput. Commun. Appl. **17** (2021), no. 2s, 1–26.

[12] D. Sisodia and D.S. Sisodia, *Prediction of diabetes using classification algorithms*, Proc. Comput. Sci. 132 (2018), 1578–1585.

[13] A. Unler and A. Murat, *A discrete particle swarm optimization method for feature selection in binary classification problems*, Eur. J. Oper. Res. **206** (2010), no. 3, 528–539.