Global Journal of Computer Science and Technology: D
Neural & Artificial Intelligence
Volume 19 Issue 2 Version 1.0 Year 2019
Type: Double Blind Peer Reviewed International Research Journal
Publisher: Global Journals
Online ISSN: 0975-4172 & Print ISSN: 0975-4350

An Optimized Recursive General Regression Neural Network Oracle for the Prediction and Diagnosis of Diabetes

By Dana Bani-Hani, Pruthak Patel & Tasneem Alshaikh

State University of New York at Binghamton

Abstract- Diabetes is a serious, chronic disease that has been seeing a rise in the number of cases and prevalence over the past few decades. It can lead to serious complications and can increase the overall risk of dying prematurely. Data-oriented prediction models have become effective tools that help medical decision-making and diagnoses in which the use of machine learning in medicine has increased substantially. This research introduces the Recursive General Regression Neural Network Oracle (R-GRNN Oracle) and is applied on the Pima Indians Diabetes dataset for the prediction and diagnosis of diabetes. The R-GRNN Oracle (Bani-Hani, 2017) is an enhancement to the GRNN Oracle developed by Masters et al. in 1998, in which the recursive model is created of two oracles: one within the other. Several classifiers, along with the R-GRNN Oracle and the GRNN Oracle, are applied to the dataset, they are: Support Vector Machine (SVM), Multilayer Perceptron (MLP), Probabilistic Neural Network (PNN), Gaussian Naïve Bayes (GNB), K-Nearest Neighbor (KNN), and Random Forest (RF). Genetic Algorithm (GA) was used for feature selection as well as the hyperparameter optimization of SVM and MLP, and Grid Search (GS) was used to optimize the hyperparameters of KNN and RF. The performance metrics accuracy, AUC, sensitivity, and specificity were recorded for each classifier.

Keywords: GRNN oracle, data mining, machine learning, genetic algorithm, diabetes, prediction model.

GJCST-D Classification: I.2.6

An Optimized Recursive General Regression Neural Network Oracle for the Prediction and Diagnosis of Diabetes

Dana Bani-Hani [α], Pruthak Patel [σ] & Tasneem Alshaikh [ρ]

Abstract- Diabetes is a serious, chronic disease that has been seeing a rise in the number of cases and prevalence over the past few decades. It can lead to serious complications and can increase the overall risk of dying prematurely. Data-oriented prediction models have become effective tools that help medical decision-making and diagnoses in which the use of machine learning in medicine has increased substantially. This research introduces the Recursive General Regression Neural Network Oracle (R-GRNN Oracle) and is applied on the Pima Indians Diabetes dataset for the prediction and diagnosis of diabetes. The R-GRNN Oracle (Bani-Hani, 2017) is an enhancement to the GRNN Oracle developed by Masters et al. in 1998, in which the recursive model is created of two oracles: one within the other. Several classifiers, along with the R-GRNN Oracle and the GRNN Oracle, are applied to the dataset, they are: Support Vector Machine (SVM), Multilayer Perceptron (MLP), Probabilistic Neural Network (PNN), Gaussian Naïve Bayes (GNB), K-Nearest Neighbor (KNN), and Random Forest (RF). Genetic Algorithm (GA) was used for feature selection as well as the hyperparameter optimization of SVM and MLP, and Grid Search (GS) was used to optimize the hyperparameters of KNN and RF. The performance metrics accuracy, AUC, sensitivity, and specificity were recorded for each classifier. The R-GRNN Oracle was able to achieve the highest accuracy, AUC, and sensitivity (81.14%, 86.03%, and 63.80%, respectively), while the optimized MLP had the highest specificity (89.71%).

Keywords: GRNN oracle, data mining, machine learning, genetic algorithm, diabetes, prediction model.

I. Introduction

According to the World Health Organization (WHO), the number of people with diabetes had quadrupled since 1980. Prevalence is increasing worldwide, particularly in low- and middle-income countries. It is estimated that medical costs and lost work and wages for people diagnosed with diabetes is $327 billion yearly and twice as much as those who do not have diabetes (CDC, 2018). About 422 million people worldwide have the disease. It can lead to serious complications in any part of the body such as kidney disease, blindness, nerve damage, and heart disease (Temurtas et al., 2009),and increases the risk of dying prematurely – diabetes is the seventh leading cause of death worldwide.

There are many factors to analyze to diagnose diabetes in a patient which makes the physician's job difficult. Thus, to save time, cost, and the risk of an inexperienced physician, classification models may be built to help predict and diagnose diabetes based on previous records (Polat et al., 2008). The use of machine learning in medicine has increased substantially. With the exponential growth of big data, manual efforts to analyze such data are impossible, therefore, automated techniques such as machine learning are used. Machine learning is defined as having the ability for a system to learn on its own, by extracting patterns from large raw data (Goodfellow et al., 2016).

The General Regression Neural Network Oracle (GRNN Oracle), developed by Masters et al. in 1998, combines the predictions of individually trained classifiers and outputs one superior prediction by determining the error rate for each classifier form a set of observations in order to assign weights to favor classifiers with lower error rates. The final prediction for an unknown observation is calculated by summing each classifier's prediction for that unknown observation multiplied by the classifier's weight; the classifiers with lower error rates have greater influence on the final prediction.

Because of the strong capabilities of the oracle, it has been enhanced to consist of two GRNN Oracles; one within the other. First proposed by Bani-Hani (2017), the first oracle is created through its own combination of algorithms and acts now as a classifier as it has its own predictions and error contribution to a set of unknown observations. It is then combined with other classifiers to create a new, outer oracle that has been named the Recursive General Regression Neural Network Oracle (R-GRNN Oracle). This study is applied on the Pima Indians Diabetes dataset where Genetic Algorithm (GA) is used for feature selection and hyperparameter optimization, and the proposed classifier, the Recursive General Regression Neural Network Oracle (R-GRNN Oracle), is applied along with seven other classifiers, namely Support Vector Machine (SVM), Multilayer Perceptron (MLP), Random Forest (RF), Probabilistic Neural Network (PNN), Gaussian Naïve Bayes (GNB), K-

Author α σ: Department of Systems Science and Industrial Engineering, State University of New York at Binghamton, Binghamton, NY 13902, USA. e-mail: dbaniha1@binghamton.edu

Author ρ: Department of Industrial Engineering, Jordan University of Science and Technology, Al Ramtha, Jordan.

Nearest Neighbor (KNN), and the GRNN Oracle, for the prediction and diagnosis of diabetes. The R-GRNN Oracle was able to achieve the highest accuracy and AUC (area under the Receiver Operating Characteristic (ROC) curve) performance metrics in comparison to the other classifiers used.

The remainder of this paper is organized as follows: Section 2 presents the related work regarding this study. Section 3 explains the methodology adopted in this study. Section 4 shows the experimental analysis and results. Section 5 presents the discussion. And Section 6 presents the conclusion and future work.

II. Related Work

Prediction models are vastly implemented in clinical and medical fields to support diagnostic decision-making (Zheng et al., 2015). Very few of its diagnostic applications include the prediction of Alzheimer's disease (López et al., 2009; Ramírez et al., 2013; Beheshti et al. 2017), Parkinson's disease (Gil and Manuel, 2009; Haller et al., 2012; Aich et al., 2018), and cancer such as breast cancer (Akay, 2009; Karabatak and Ince, 2009; Zheng et al., 2014; Bhardwaj and Tiwari, 2015), lung cancer (Jayasurya et al., 2010; Sun et al., 2013; Sakumua et al., 2017), and leukemia (Fang and Grzymala-Busse, 2006; Manninen et al., 2013).A plethora of studies have been carried out on the prediction of diabetes. Polat et al. (2008) used Least Square Support Vector Machine (LS-SVM) for the prediction of diabetes through Generalized Discriminant Analysis (GDA). Park and Edington (2001) applied sequential multi-layered perceptron (SMLP) with back propagation learning on 6,142 participants. The early detection of diabetes type II was conducted by Zhu et al. in 2015 in which they proposed a dynamic voting scheme ensemble. Thirugnanam et al. (2012) adopted techniques such as fuzzy logic, Neural Network (NN), and case-based reasoning as an individual approach (FNC) for the diagnosis of diabetes.

Regarding the dataset used in this study, the Pima Indian Diabetes dataset, various studies used the dataset to create prediction models for the prediction and diagnosis of diabetes. Kayaer and Yildirim (2003) applied an MLP, Radial Basis Function (RBF), and a General Regression Neural Network (GRNN) on the Pima Indian Diabetes dataset. Their highest accuracy was achieved by the GRNN at 80.21%. Carpenter and Markuzon(1998) applied several techniques on the dataset including, but not limited to, KNN, Logistic Regression (LR), the perceptron-like ADAP model, ARTMAP, and ARTMAP-IC (named for instance counting and inconsistent cases), in which the ARTMAP-IC obtained the highest accuracy at 81%. Bradley (1997) also used various classifiers on the dataset where the author's main purpose was to assess the use of the AUC as a performance metric. The author was able to achieve the highest accuracy of 78.4% using a two-layer MLP. A hybrid of Artificial Neural Network (ANN) and Fuzzy Neural Network (FNN) was proposed by Kahramanli and Allahverdi in 2008. Their approach resulted in an accuracy of 84.2%.Lekkas and Mikhailove (2010) applied Evolving Fuzzy Classification (EFC) to two datasets including Pima Indians Diabetes dataset. They were able to reach an accuracy of 79.37%.Miche et al. (2010) presented the Optimally Pruned Extreme Learning Machine (OP-ELM) and compared its performance to a MLP, SVM, and Gaussian Process (GP) on several regression and classification datasets. Regarding the dataset concerning this study, the GP had the highest accuracy among the classifiers tested with an accuracy of 76.3%. Huang et al. (2004) was able to achieve an accuracy of 77.31% using SVM, although their paper proposed an algorithm called Extreme Machine Learning (EML). Kumari and Chitra (2013) used SVM and obtained an accuracy of 78.2%. Al Jarullah (2011) also found the accuracy to be 78.2% using Decision Trees (DTs). Bradley and Mangasarian (1998) applied Feature Selection via Concave (FSC), SVM, and Robust Linear Program (RLP) in which the RLP had the highest accuracy on the Pima Indian Diabetes dataset at 76.16%. Using a novel Adaptive Synthetic (ADASYN) sampling approach, He et al. (2008) achieved an accuracy of 68.37%.Şahan et al. (2005) proposed a new artificial immune system named Attribute Weighted Artificial Immune System (AWAIS) in which they attained a classification accuracy of 75.87%.Luukka (2011) used Similarity-Based (S-Based) classifier with fuzzy entropy measures as a feature selection method and reached an accuracy of 75.97%. Using Extreme Gradient Boosting (XGBoost), Christina et al. (2018) achieved 81% accuracy. Ramesh et al. (2017) used deep learning, more specifically Restricted Boltzmann Machine (RBM), on the dataset with 81% accuracy. Vaishali et al. (2017) applied GA for feature section with a Multi Objective Evolutionary Fuzzy (MOEF) classifier and obtained an accuracy of 83.04%.

Many other studies have been carried out on the same dataset, however, due to reporting training accuracies rather than testing and validation accuracies, they have been excluded from the literature review for several reasons including, and most importantly, overfitting, as overfitting generates higher accuracies due to fitting the model too perfectly to the training set making the model not generalized enough. The other studies that have been excluded are those that obtained high accuracies but did not mention whether they obtained it from a training set or a testing or validation set making the results questionable. It is worthy to note that this study applied 4-fold cross validation to train each classifier and were tested on a validation subset that did not take part in neither the training nor testing steps.

III. Methodology

Six individual classifiers were used in this research: SVM, MLP, RF, PNN, GNB, and KNN, in which some were used to create the GRNN Oracle, and some were combined with the first oracle to create the R-GRNN Oracle. The software and language used for this study was Python 3.6 and the hardware specifications were Intel® Core™ i7-8750H CPU @ 2.20GHz with 32.0 GB RAM.

a) Individual Classifiers

Support Vector Machine: SVM is a statistical learning method proposed by Vapnik (1995). It is a widely used supervised machine learning algorithm used for both classification and regression. SVM works by finding the hyperplane that maximizes the margin between the classes in the feature space, as seen in Figure 1. Support vectors are observations that help dictate the hyperplane. It classifies new samples based on which side of the boundary they are located on.

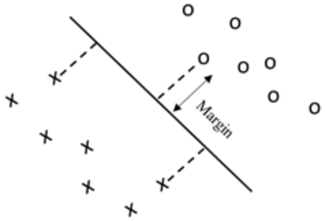

Figure 1: A simple linear SVM

Multilayer Perceptron: MLP is a feed forward artificial NN that is a modification of the standard linear perceptron. It is an algorithm that does not require a linear relationship between the independent variables and the dependent variable as it is able to solve problems that are not linearly separable through the use of activation functions located in each node. An MLP consists of an input layer, a hidden layer(s), and an output layer. It is a supervised machine learning algorithm that exploits back propagation to train itself to optimize the weights of each edge connecting two nodes. It is the most frequently used NN (Hossain et al., 2017) and is widely-used for classification, regression, recognition, prediction, and approximation tasks. Figure 2 illustrates an example of anMLP with one hidden layer with five hidden nodes.

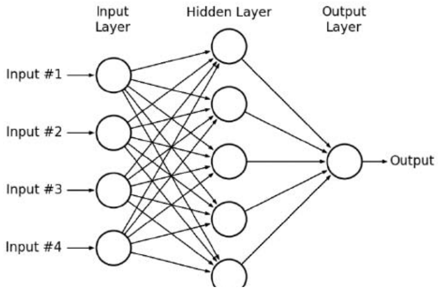

Figure 2: AnMLP NN with one hidden layer (Mohamed et al. 2015)

Random Forest: RF is an ensemble created by Ho (1995) that is used for classification and regression. It has received great attention from researchers because of its simplicity and ensemble learning characteristics (Breiman, 2001). A RF is made up of many DTs where they are created through a random sampling process with replacements (Belgiu and Drăguţ, 2016). RF uses the bagging technique to improve the model's performance by decreasing the model's variance without increasing the bias which helps overcome DTs' poor habit of overfitting.

Probabilistic Neural Network: PNN is a feedforward NN which is used for classification and pattern recognition problems. The probability density function (PDF) for each class is estimated using a Parzen window (kernel density estimation (KDE)) and a non-parametric function. It then uses the Baye's strategy for estimating the class probabilityof a new input using the PDF of each class (Karthikeyan et al., 2008). It consists of three layers: an input layer, a hidden layer, and an output layer.

Gaussian Naïve Bayes: GNB is a supervised learning algorithm that is widely used for classification problems because of its simplicity and accurate results (Farid et al., 2014). It uses Bayes theorem as its framework (Griffis et al., 2016) and has strong independence assumptions between the independent variables. One important advantage of GNB is that it could estimate the parameters necessary for classification by training on a small training set.

K-Nearest Neighbor: KNN is a non-parametric, lazy learning method for classification and regression tasks (Zhang, 2016). $k$is a user-set parameter that represents the number of known observations closest to the unknown observation mapped out in the feature space. For classification tasks, the class of the new observation is based on the majority class surrounding it; k is typically an odd number. For regression tasks, the new observation is taken as the average of its k neighbors.

b) Optimization Algorithms

Genetic Algorithm: GA is a population-based metaheuristic developed by John Holland in the 1970s (Holland, 1992). It is a widely used optimization technique inspired by nature, more specifically, evolution and survival of the fittest. It finds solutions throughout the search space using two main operators: crossover and mutation. Every solution is represented as a chromosome with several alleles encoded with genetic material that measure the fitness value of the objective function. Crossover produces two somewhat different chromosomes, called offspring, from two parents. The mutation operator is applied on the offspring at a given probability to create diversity in the population pool, which allows diversification in the search space.

Grid Search: GS is an exhaustive search optimization technique that works with user-set parameters for an algorithm. It is a traditional approach to manual hyperparameter tuning in which all possible combinations of the parameters selected are tested. It is guided by a performance metric and typically measured by cross validation on the training set or an evaluation on a validation subset (Hsu et al., 2003).

c) *GRNN Oracle*

The GRNN Oracle combines the predictive powers of several machine learning classifiers that were trained independently to form one superior prediction (Li, 2014). It determines the error rate for each classifier involved in the oracle in order to assign weights to favor classifiers with lower error rates. The final prediction for an unknown observation is calculated by summing each classifier's prediction for that unknown observation multiplied by the classifier's weight.

The steps involved in predicting a class (output) for a single observation are: first, each classifier (k) is trained on a training subset of the data and tested on another subset to obtain predictions for the observations. Second, each prediction obtained from the previous step (probability of belonging to each class) for each observation is compared to its actual prediction (actual class) and the Mean Squared Error (MSE) is calculated through Formula 1:

$$error_{i,k} = \sum_{m=1}^{num\_classes} \left(AP_m - PP_{m,k}\right)^2 / num\_classes \tag{1}$$

where $error_{i,k}$ is the mean squared error of a known observation (i) from classifier (k), $num\_classes$ is the total number of classes, AP_m is the actual probability of the known observation (i) for being class (m) and $PP_{m,k}$ is the predicted probability of being class (m) from classifier (k).Third, for a given unknown observation in the validation set (an observation that needs to be predicted), the distance between the observation and all the known samples in the testing set is calculated, and each known observation has a particular weight for the unknown observation. The distance is calculated using Formula 2 and the weight is calculated using Formula 3.

$$D(\vec{x},\vec{x}_i) = \frac{1}{p}\sum_{j=1}^{p}((x_j - x_{ij})/\sigma_j)^2 \tag{2}$$

$$weight_i = e^{-D(\vec{x},\vec{x}_i)} \tag{3}$$

where $\vec{x}$ represents the vector of features belonging to the unknown observation, [feature 1, feature 2, …, feature p], $\vec{x}_i$ is the feature vector for the known observation, x_j is the j-th feature of the unknown observation, x_{ij} is the j-th feature of the known observation, σ_j is an adjustable sigma parameter for the j-th feature and p is the total number of features. w_k is the weight (trust) of classifier (k) on the prediction of the unknown observation. Fourth, for the unknown observation, for each classifier (k), the predicted squared error is obtained through the MSE and weight of each known observation (Formula 4).

$$error_k(\vec{x}) = (\textstyle\sum_{i=1}^{n} error_{i,k} * weight_i)/\sum_{i=1}^{n} weight_i \tag{4}$$

Fifth, each classifier (k) has an amount of trust for the final prediction of the unknown observation where the higher the weight, the more influence it has on the final prediction of the unknown observation (Formula 5).

$$w_k = (1/error_k)/(\textstyle\sum_{l=1}^{L} 1/error_k) \tag{5}$$

$$\textstyle\sum_{k=1}^{L} w_k = 1 \tag{6}$$

where$L$ is the total number of classifiers, and l indicates classifier l. The sum of w_k for all classifiers (L) equals one (Formula 6). Lastly, through the amount of error each classifier (k) contributes, their trust/weight is multiplied by the unknown observation's prediction and summed up to form the final prediction for that particular unknown observation (Formula 7).

$$\hat{y} = \textstyle\sum_{k=1}^{L} w_k * q_k \tag{7}$$

where$\hat{y}$ is the prediction of the unknown observation outputted by the GRNN Oracle represented as a class membership vector and q_k is the predicted class membership vector for the unknown observation given by classifier (k).

d) *Recursive GRNN Oracle*

The best combination of classifiers that were trained and tested individually and independently was used to make the first oracle. By having predictions outputted from the oracle, it now acts as any other machine learning classifier would. The best combination of classifiers that would enhance the performance of the first GRNN Oracle is selected and this selected combination, including the first oracle, creates the second oracle, the R-GRNN Oracle. The accuracy, AUC, sensitivity, and specificity of its final predictions are taken, along with the same performance metrics of the inner GRNN Oracle and the individual classifiers for the final comparison.

IV. Experimental Analysis and Results

a) *Dataset Description*

The Pima Indians Diabetes dataset was used in this study where it was originally a study conducted by the National Institute of Diabetes and Digestive and Kidney Diseases (NIDDK) on the Pima Indian population near Phoenix, Arizona, in 1965 (Smith et al., 1988). There is a total of 768 observed patients where 268 of them have diabetes, which indicates the imbalanced property of the dataset. In this dataset, there are eight independent variables (features) and one dependent variable (outcome: diabetes or no diabetes), as presented in Table 1. More detailed attributes distributions and statistical analysis are further shown in Figure 3, where the color orange signifies patients who have diabetes. All patients recorded are females at least 21 years old of Pima Indian heritage.

Table 1: Pima Indians Diabetes dataset feature description

	Description	Type
X1	No. of Pregnancies	Discrete
X2	Plasma Glucose Concentration	Continuous
X3	Diastolic Blood Pressure	Continuous
X4	Skin Thickness	Continuous
X5	2-hr Serum Insulin	Continuous
X6	BMI	Continuous
X7	Diabetes Pedigree Function	Continuous
X8	Age	Continuous
Y	Outcome: Diabetes/No Diabetes	Discrete

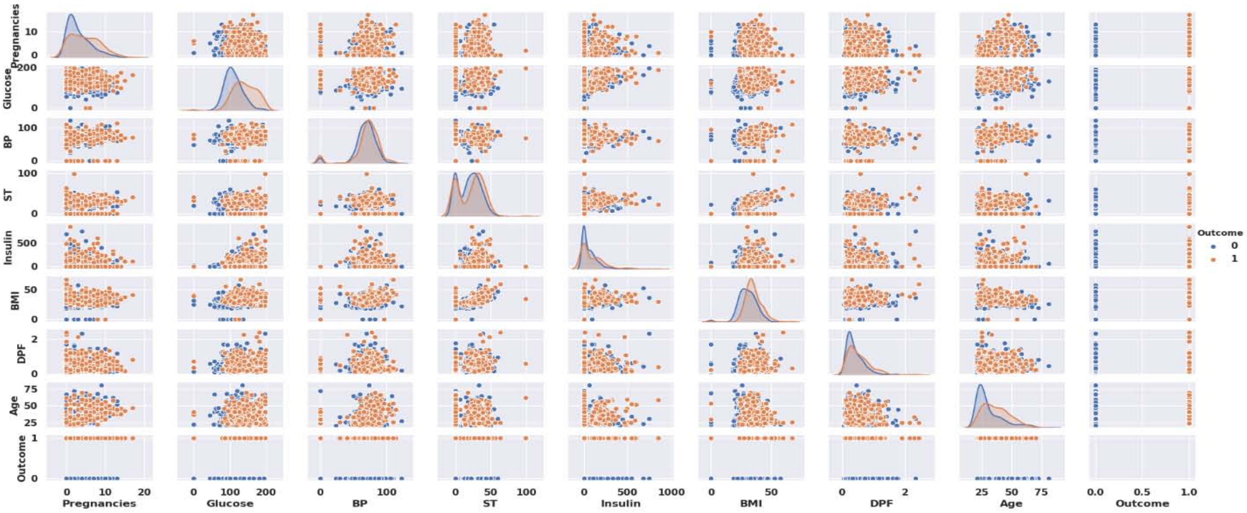

Figure 3: Attributes distributions and statistical analysis of the Pima Indians Diabetes dataset (printed in color)

b) *Data Preprocessing*

The first step taken in the data preprocessing phase was excluding outliers as they can drastically affect the model's predictive ability. Any point that was three standard deviations (3σ) away from the mean of any given feature was excluded. The original dataset had 768 patients, and after outlier removal, the new dataset contained 709 patients where 243 of them had diabetes. The next step was to correct the imbalanced property of the data. Since only 243 patients from the remaining 709 had diabetes, this is a class imbalance problem where those with diabetes only make up 34% of the data. Thus, an oversampling approach was applied to the minority class. Oversampling was favored over under sampling because the dataset's size concerning the number of observations was already small, and concerning how the R-GRNN Oracle works, it would not be a wise approach to remove observations, as the recursive oracle requires the dataset to be relatively large for the data subsets to be drawn. After this step, a normalization technique was applied to each independent variable in which the variable was scaled to a range between 0 and 1; 0 indicating the lowest value in a particular feature and 1 indicating the highest. The formula of normalization is given in Formula 8 where $\min V_i$ is the minimum value in the set of values in feature V_i, and $\max V_i$ is the maximum value in feature V_i. This is performed to ensure each feature has an equal weight so that no one feature would outweigh another before the creation of the prediction model.

$$\bar{v}_j = (v_j - \min V_i)/(\max V_i - \min V_i) \qquad (8)$$

c) *Hyperparameter Optimization*

The hyperparameters in any algorithm contributes greatly to the output of the model, therefore, determining the optimal (or near-optimal) combination of hyperparameters would yield the best result. For example, some of a NN's hyperparameters include the number of hidden layers a user sets and the number of hidden nodes in each hidden layer. Hyperparameters are defined as the properties of a model that the user can set the value to. They are different from parameters as parameters are changed internally by the model itself during training rather than set by the user before the training process. An example to a parameter is the weights of a NN, as they are adjusted through back propagation using Gradient Decent (or any other optimizer) rather than by the user.

GA was utilized to optimize the performances of the SVM and MLP, while GS was applied on KNN and RF. The reason that GS was used instead of GA was that both KNN and RF have one parameter of interest: the number of neighbors and the number of DTs, receptively. Therefore, no combinations of hyperparameters are needed which makes it a straightforward exhaustive search. SVM and MLP however have more than one hyperparameter that need to be optimized simultaneously, which also include continuous values, this is why GA is used.

Formula 9 shows the fitness function (FV) used to evaluate each chromosome (each solution). They were evaluated based on their prediction accuracies, where TP is the true positive rate, in which it indicates those who actually have diabetes and were predicted to have diabetes, TN is the true negative rate where those who do not have diabetes were predicted not having diabetes, FP is the false positive rate in which those without diabetes were falsely predicted that they do have diabetes, FN is the false negative rate, where patients have diabetes but were falsely predicted that they don't, and K is the number of folds required for the K-fold cross validation, in which it was set to four for this study.

$$FV = \frac{1}{K}\left[\sum_{k=1}^{K}\frac{TP+TN}{TP+FP+TN+FN}\right] \qquad (9)$$

The hyperparameters that were included in this study relating to SVM were c and gamma (γ), where both take on continuous values, while MLP's hyperparameters included the learning rate (α), momentum, the number of hidden layers, the number of hidden nodes in each hidden layer, and the solver, where α and momentum are continuous, the number of hidden layers and nodes are integers, and the solver is categorical. Figure 4 and Figure 5 show the encoding (genotype) for the SVM and MLP parameters, respectively, where each continuous hyperparameter was encoded with a binary chromosome with a length of 15 alleles.

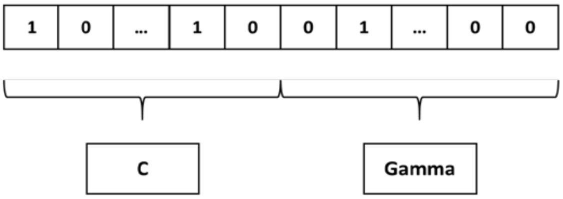

Figure 4: Chromosome encoding of the SVM parameters

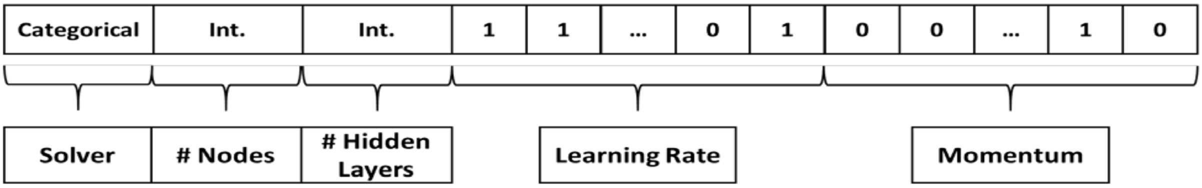

Figure 5: Chromosome encoding of the MLP parameters

SVMs can handle nonlinear classifications through transforming inputs into feature vectors with the use of kernels. The SVM kernel set for this study is the Radial Basis Function (RBF) in which it is a popular Gaussian kernel function. Some of RBF's greatest advantages are its high accuracy, its fast convergence, and its applicability in almost any dimension. $c$is a regularization hyperparameter that determines how correctly the hyperplane between the classes separates the data. It controls the trade-off between model complexity and training error (Joachims, 2002). γ has a serious impact on the classification accuracy as it defines the influence of each training observation (Tuba and Stanimirovic, 2017), with lower values meaning "far" and higher values meaning "close". It can be thought of as the inverse of the radius of influence of observations selected by the model as support vectors. Figure 6-A shows SVM's accuracy achieved by GA in each of the 100 generations run.

With MLP, the activation function set in this study was the Rectified Linear Unit (ReLU). Activation functions are operations which map an output to a set of inputs. They are used to impart non-linearity to the network structure (Acharya et al, 2017). Because ReLU returns a positive number, i.e. $max\ (x, 0)$, the two major advantages of it are sparsity and the reduced likelihood of the "vanishing gradient" problem, as adding as many hidden layers as one would like would not cause the gradient multiplication to reach a very small number that it will likely "vanish" with more layers to add. Solvers in NNs train and optimize the weights connecting the nodes between two-adjacent layers. The two solvers considered for this study are Stochastic Gradient Decent (SGD) and Adam, a variant of SGD. The other two important hyperparameters are $\alpha$and momentum. α controls how fast the network learns during training andmomentum helps to converge the data (Acharya et al, 2017). They can be thought of the stepping size and direction in the search space. Figure 6-B shows MLP's accuracy achieved by GA in each of the 100 generations run.

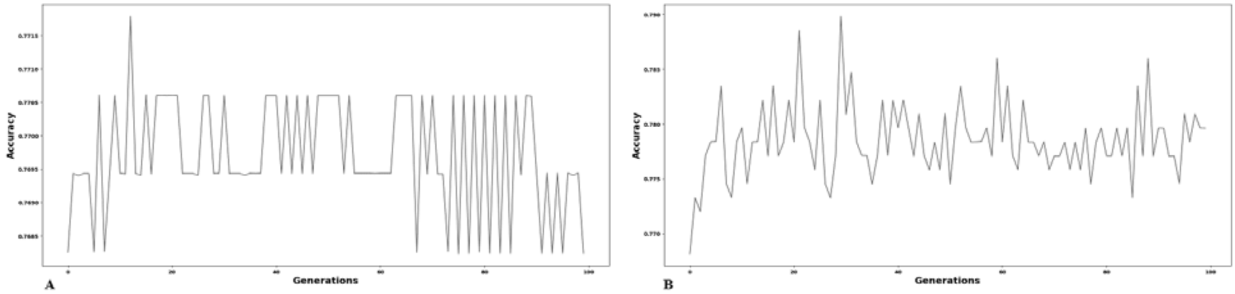

Figure 6: Prediction accuracy throughout 100 generations, A- SVM, B- MLP

d) Feature Selection

Feature selection plays an important role in classification for several reasons (Luukka, 2011). First, it can simplify the model's complexity which helps reduce computational cost, and when the model is taken for practical use fewer inputs are needed. Second, by removing redundant features from the dataset one can also make the model more transparent and more comprehensible, providing better explanation of suggested diagnosis, which is an important requirement in medical applications. Feature selection process can also reduce noise in which it may enhance classification accuracy.

GA was applied for feature selection through SVM and its optimized hyperparameters from the previous step. The solution representation for feature selection was embodied by a chromosome of eight binary values (i.e. 0's and 1's). An allele of the value 0 indicates that feature n was not included while an allele of 1 indicated that it was included; n is the nth feature in the dataset. To explain further, Figure 7 illustrates an example where the encoding of the selected features #1, #2, #3, and #6, out of eight featuresis shown. Chromosomes with a subset of features selected are then evaluated based on their accuracy. The subset of features that attained the highest accuracy was selected for further analysis. Formula 9 was also used as the fitness function for chromosome evaluation.

1	1	1	0	0	1	0	0

Figure 7: Chromosome encoding of a selected subset of features

e) Recursive GRNN Oracle

For the first GRNN Oracle (the inner oracle), the classifiers fed into it were SVM, GNB, and RF. The accuracy and AUC for SVM were 79.72% and 85.79%, respectively. GNB had 79.09% and 85.56%, respectively, and RF at 77.50% and 81.15%. The performance of the first oracle had an accuracy of 79.54%, AUC of 85.16%, sensitivity of 59.60%, and specificity of 88.51%. MLP, PNN, and KNN were not chosen because of their inferior performances when compared to the other models. All models were run 15 times and the average of the performance metrics were taken.

For the R-GRNN Oracle, the first GRNN Oracle, which now acts as a classifier with its own predictions, was combined with SVM. Since SVM had a better performance than others, itwas chosen as a match with the first oracle to create the second oracle. Figure 8 illustrates the classifiers being fed into each one of the two oracles.

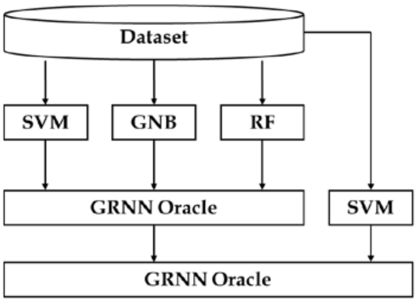

Figure 8: Overview of the Recursive GRNN Oracle

The first oracle achieved an accuracy of 79.54%, however, it was surpassed by SVM (79.72%), but the recursive model had the highest accuracy at 81.14% and highest AUC at 86.03%. Although it was able to reach the highest sensitivity too (63.80%) in comparison to the rest, MLP had the highest sensitivity (89.71%), where the recursive model came in third with 89.14% after MLP and SVM. However, since detecting TPs is of great importance (those who have diabetes), the sensitivity metric, where the R-GRNN was the highest, has a higher significance than specificity. Table 2shows the accuracy, AUC, sensitivity, and specificity of all the classifiers: six individual classifiers (performing on their own), the GRNN Oracle, and the R-GRNN Oracle. The performances can also be seen in Figure 9 and Figure 10. Figure 9 shows the recursive model's 15 runs where the best, average, and worst performance were recorded, 86.47%, 81.14%, and 76.15%, respectively. It is worthy to mention that the dataset was shuffled each time the classifiers were run to ensure the robustness of the model, as no matter how it the data was shuffled, it always yielded better performance than the rest of the classifiers. Shuffling the data is the reason behind the high variation seen in Figure 9. Also, as a reminder to what was mentioned earlier, 4-fold cross validation was applied to train and test the models, but the actual validation of each model was applied on a validation subset that was not involved in neither the training nor testing steps of each model.

Table 2: Performance metrics for the classifiers (average of 15 runs)

	Accuracy	AUC	Sensitivity	Specificity
SVM	79.72	85.79	58.43	89.31
MLP	76.88	80.75	49.11	**89.71**
RF	77.50	81.15	57.11	86.65
PNN	71.03	75.54	61.43	75.24
GNB	79.09	84.56	60.44	87.58
KNN	76.59	80.77	58.72	84.53
GRNN O.	79.54	85.16	59.60	88.51
R. GRNN O.	**81.14**	**86.03**	**63.80**	89.14

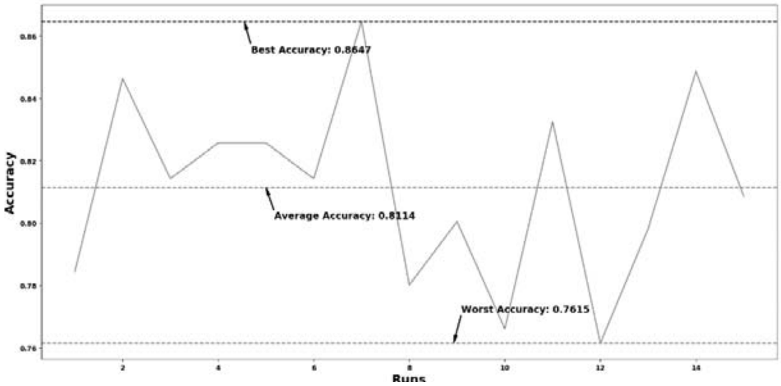

Figure 9: The best, worst, and average performances of the R-GRNN Oracle in 15 runs

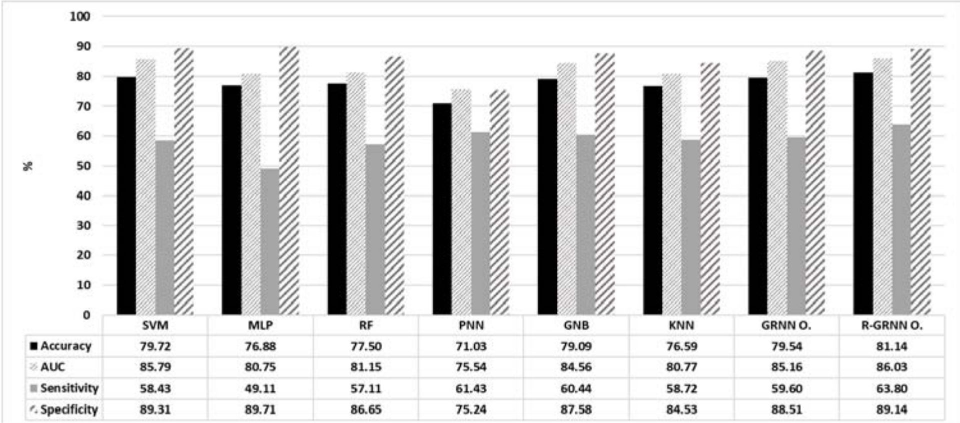

Figure 10: Graphical representation for the performance metrics for the classifiers

V. Discussion

While the accuracy of the proposed model was not the highest in the literature, it still came in third when compared to all the publications studied (Table 3). It also bested the traditional oracle, SVM, MLP, RF, PNN, GNB, and KNN. However, as a slight remark, the studies did not confirm whether their accuracies were from conducting several runs and taking the average or not. As in this study, the highest accuracy achieved by the recursive model was 86.47%; one could simply report it as the highest achieved, therefore, it is wise if several runs are conducted and the average was taken.

Table 3: Comparison of the accuracy in the literature with this study

	Method	Accuracy
Kahramanli and Allahverdi (2008)	ANN and FNN Hybrid	84.20%
Vaishali et al. (2017)	MOEF	83.04%
This Study	**R-GRNN Oracle**	**81.14%**
Carpenter and Markuzon (1998)	ARTMAP-IC	81.00%
Christina et al. (2018)	XGBoost	81.00%
Ramesh et al. (2017)	RBM	81.00%
Kayaer and Yildirim (2003)	GRNN	80.21%
Lekkas and Mikhailove (2010)	EFC	79.37%
Bradley (1997)	MLP	78.40%
Al Jarullah (2011)	DTs	78.20%
Kumari and Chitra (2013)	SVM	78.20%
Huang et al. (2004)	SVM	77.31%
Miche et al. (2010)	GP	76.30%
Bradley and Mangasarian (1998)	RLP	76.16%
Luukka (2011)	S-Based	75.97%
Şahan et al. (2005)	AWAIS	75.87%
He et al. (2008)	ADASYN	68.37%

VI. Conclusion and Future Work

This study presented the R-GRNN Oracle and was applied on the Pima Indians Diabetes dataset. It was applied along with seven other classifiers in which their final performances were compared. The other classifiers included are the traditional GRNN Oracle, SVM, MLP, RN, PNN, GNB, and KNN. GA was used to optimize the hyperparameters of SVM and MLP, and GS was used on RF and KNN. The models were run 15 times and the dataset was shuffled each run to ensure robustness. 4-fold cross validation was adopted as the validation method. Compared to the other models, the recursive oracle achieved the highest accuracy, AUC, and sensitivity at 81.14%, 86.03%, and 63.80%, respectively. It, however, came in third for specificity at 89.14% where optimized MLP had the highest at 89.71%.

Future research may include applying feature selection and hyperparameter optimization simultaneously rather than applying feature selection based on the optimized hyperparameters from all the features. It can also include using other metaheuristics, such as Particle Swarm Optimization (PSO) for hyperparameter optimization.

References Références Referencias

1. Acharya, U. R., Oh, S. L., Hagiwara, Y., Tan, J. H., & Adeli, H. (2017). Deep convolutional neural network for the automated detection and diagnosis of seizure using EEG signals. Computers in biology and medicine.
2. Aich, S., Younga, K., Hui, K. L., Al-Absi, A. A., & Sain, M. (2018, February). A nonlinear decision tree-based classification approach to predict the Parkinson's disease using different feature sets of voice data. In Advanced Communication Technology (ICACT), 2018 20th International Conference on (pp. 638-642). IEEE.
3. Akay, M. F. (2009). Support vector machines combined with feature selection for breast cancer diagnosis. Expert systems with applications, 36(2), 3240-3247.
4. Al Jarullah, A. A. (2011, April). Decision tree discovery for the diagnosis of type II diabetes. In Innovations in Information Technology (IIT), 2011 International Conference on (pp. 303-307). IEEE.
5. Bani-Hani, D. (2017). A Recursive General Regression Neural Network Oracle through Applying a Polybrid of Machine Learning Algorithms (Master's thesis, State University of New York at Binghamton).
6. Beheshti, I., Demirel, H., Matsuda, H., & Alzheimer's Disease Neuroimaging Initiative. (2017). Classification of Alzheimer's disease and prediction of mild cognitive impairment-to-Alzheimer's conversion from structural magnetic resource imaging using feature ranking and a genetic algorithm. Computers in biology and medicine, 83, 109-119.
7. Belgiu, M., & Drăguţ, L. (2016). Random forest in remote sensing: A review of applications and future directions. ISPRS Journal of Photogrammetry and Remote Sensing, 114, 24-31.
8. Bhardwaj, A., & Tiwari, A. (2015). Breast cancer diagnosis using genetically optimized neural network model. Expert Systems with Applications, 42(10), 4611-4620.
9. Bradley, P. S., & Mangasarian, O. L. (1998, July). Feature selection via concave minimization and support vector machines. In ICML (Vol. 98, pp. 82-90).

10. Breiman, L. (2001). Random forests. Machine learning, 45(1), 5-32.
11. Carpenter, G. A., & Markuzon, N. (1998). ARTMAP-IC and medical diagnosis: Instance counting and inconsistent cases. Neural Networks, 11(2), 323-336.
12. Centers for Disease Control and Prevention. Diabetes quick facts. <https://www.cdc.gov/diabetes/basics/quick-facts.html>. Accessed on 2018-09-08.
13. Christina, S. S., & Santiago, N. (2018). Decision Support System for a Chronic Disease-Diabetes.
14. Fang, J., & Grzymala-Busse, J. W. (2006, June). Leukemia prediction from gene expression data-a rough set approach. In International conference on artificial intelligence and soft computing (pp. 899-908). Springer, Berlin, Heidelberg.
15. Farid, D. M., Zhang, L., Rahman, C. M., Hossain, M. A., & Strachan, R. (2014). Hybrid decision tree and naïve Bayes classifiers for multi-class classification tasks. Expert Systems with Applications, 41(4), 1937-1946.
16. Gil, D., & Manuel, D. J. (2009). Diagnosing Parkinson by using artificial neural networks and support vector machines. Global Journal of Computer Science and Technology, 9(4).
17. Goodfellow, I., Bengio, Y., Courville, A., & Bengio, Y. (2016). Deep learning (Vol. 1). Cambridge: MIT press.
18. Griffis, J. C., Allendorfer, J. B., & Szaflarski, J. P. (2016). Voxel-based Gaussian naïve Bayes classification of ischemic stroke lesions in individual T1-weighted MRI scans. Journal of neuroscience methods, 257, 97-108.
19. Haller, S., Badoud, S., Nguyen, D., Garibotto, V., Lovblad, K. O., & Burkhard, P. R. (2012). Individual detection of patients with Parkinson disease using support vector machine analysis of diffusion tensor imaging data: initial results. American Journal of Neuroradiology.
20. He, H., Bai, Y., Garcia, E. A., & Li, S. (2008, June). ADASYN: Adaptive synthetic sampling approach for imbalanced learning. In Neural Networks, 2008. IJCNN 2008. (IEEE World Congress on Computational Intelligence). IEEE International Joint Conference on (pp. 1322-1328). IEEE.
21. Holland, John Henry. Adaptation in natural and artificial systems: An introductory analysis with applications to biology, control, and artificial intelligence. MIT press, 1992.
22. Hossain, M. S., Ong, Z. C., Ismail, Z., & Khoo, S. Y. (2017). A comparative study of vibrational response-based impact force localization and quantification using radial basis function network and multilayer perceptron. Expert Systems with Applications, 85, 87-98.
23. Hsu, C. W., Chang, C. C., & Lin, C. J. (2003). A practical guide to support vector classification.
24. Huang, G. B., Zhu, Q. Y., & Siew, C. K. (2004, July). Extreme learning machine: a new learning scheme of feed forward neural networks. In Neural Networks, 2004. Proceedings. 2004 IEEE International Joint Conference on (Vol. 2, pp. 985-990). IEEE.
25. Jayasurya, K., Fung, G., Yu, S., Dehing-Oberije, C., De Ruysscher, D., Hope, A., De Neve, W., Lievens, Y., Lambin, P. & Dekker, A. L. A. J. (2010). Comparison of Bayesian network and support vector machine models for two-year survival prediction in lung cancer patients treated with radiotherapy. Medical physics, 37(4), 1401-1407.
26. Joachims, T. (2002). Learning to classify text using support vector machines: Methods, theory and algorithms (Vol. 186). Norwell: Kluwer Academic Publishers.
27. Kahramanli, H., & Allahverdi, N. (2008). Design of a hybrid system for the diabetes and heart diseases. Expert systems with applications, 35(1-2), 82-89.
28. Karabatak, M., & Ince, M. C. (2009). An expert system for detection of breast cancer based on association rules and neural network. Expert systems with Applications, 36(2), 3465-3469.
29. Karthikeyan, B., Gopal, S., & Venkatesh, S. (2008). Partial discharge pattern classification using composite versions of probabilistic neural network inference engine. Expert Systems with Applications, 34(3), 1938-1947.
30. Kayaer, K., & Yıldırım, T. (2003, June). 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 (ICANN/ICONIP) (pp. 181-184).
31. Kumari, V. A., & Chitra, R. (2013). Classification of diabetes disease using support vector machine. International Journal of Engineering Research and Applications, 3(2), 1797-1801.
32. Lekkas, S., & Mikhailov, L. (2010). Evolving fuzzy medical diagnosis of Pima Indians diabetes and of dermatological diseases. Artificial Intelligence in Medicine, 50(2), 117-126.
33. Li, Y. (2014). A complex adaptive system for accurate detection of multiple species of pathogens using multiple machine learning techniques. State University of New York at Binghamton.
34. López, M. M., Ramírez, J., Górriz, J. M., Álvarez, I., Salas-Gonzalez, D., Segovia, F., & Chaves, R. (2009). SVM-based CAD system for early detection of the Alzheimer's disease using kernel PCA and LDA. Neuroscience Letters, 464(3), 233-238.
35. Luukka, P. (2011). Feature selection using fuzzy entropy measures with similarity classifier. Expert Systems with Applications, 38(4), 4600-4607.

36. Manninen, T., Huttunen, H., Ruusuvuori, P., & Nykter, M. (2013). Leukemia prediction using sparse logistic regression. PloS one, 8(8), e72932.
37. Masters, T., Land, W. H., & Maniccam, S. (1998, October). An oracle based on the general regression neural network. In Systems, Man, and Cybernetics, 1998. 1998 IEEE International Conference on (Vol. 2, pp. 1615-1618). IEEE.
38. Miche, Y., Sorjamaa, A., Bas, P., Simula, O., Jutten, C., & Lendasse, A. (2010). OP-ELM: optimally pruned extreme learning machine. IEEE transactions on neural networks, 21(1), 158-162.
39. Mohamed, H., Negm, A., Zahran, M., & Saavedra, O. C. (2015). Assessment of artificial neural network for bathymetry estimation using High Resolution Satellite imagery in Shallow Lakes: case study El Burullus Lake. In International water technology conference (pp. 12-14).
40. Park, J., & Edington, D. W. (2001). A sequential neural network model for diabetes prediction. Artificial intelligence in medicine, 23(3), 277-293.
41. Polat, K., Güneş, S., & Arslan, A. (2008). A cascade learning system for classification of diabetes disease: Generalized discriminant analysis and least square support vector machine. Expert systems with applications, 34(1), 482-487.
42. Ramesh, S., Balaji, H., Iyengar, N. C. S., &Caytiles, R. D. (2017). Optimal predictive analytics of pima diabetics using deep learning. diabetes, 10(9).
43. Ramírez, J., Górriz, J. M., Salas-Gonzalez, D., Romero, A., López, M., Álvarez, I., & Gómez-Río, M. (2013). Computer-aided diagnosis of Alzheimer's type dementia combining support vector machines and discriminant set of features. Information Sciences, 237, 59-72.
44. Şahan, S., Polat, K., Kodaz, H., & Güneş, S. (2005, August). The medical applications of attribute weighted artificial immune system (AWAIS): diagnosis of heart and diabetes diseases. In International Conference on Artificial Immune Systems (pp. 456-468). Springer, Berlin, Heidelberg.
45. Sakumura, Y., Koyama, Y., Tokutake, H., Hida, T., Sato, K., Itoh, T., Akamatsu, T. & Shin, W. (2017). Diagnosis by volatile organic compounds in exhaled breath from lung cancer patients using support vector machine algorithm. Sensors, 17(2), 287.
46. Smith, J. W., Everhart, J. E., Dickson, W. C., Knowler, W. C., & Johannes, R. S. (1988, November). Using the ADAP learning algorithm to forecast the onset of diabetes mellitus. In Proceedings of the Annual Symposium on Computer Application in Medical Care (p. 261). American Medical Informatics Association.
47. Sun, T., Wang, J., Li, X., Lv, P., Liu, F., Luo, Y., Y., Gao, Q., Zhu, & Guo, X. (2013). Comparative evaluation of support vector machines for computer aided diagnosis of lung cancer in CT based on a multi-dimensional data set. Computer methods and programs in biomedicine, 111(2), 519-524.
48. Temurtas, H., Yumusak, N., & Temurtas, F. (2009). A comparative study on diabetes disease diagnosis using neural networks. Expert Systems with applications, 36(4), 8610-8615.
49. Thirugnanam, M., Kumar, P., Srivatsan, S. V., &Nerlesh, C. R. (2012). Improving the prediction rate of diabetes diagnosis using fuzzy, neural network, case based (FNC) approach. Procedia engineering, 38, 1709-1718.
50. Tuba, E., & Stanimirovic, Z. (2017, June). Elephant herding optimization algorithm for support vector machine parameters tuning. In Electronics, Computers and Artificial Intelligence (ECAI), 2017 9th International Conference on (pp. 1-4). IEEE.
51. Vaishali, R., Sasikala, R., Ramasubbareddy, S., Remya, S., & Nalluri, S. (2017, October). Genetic algorithm based feature selection and MOE Fuzzy classification algorithm on Pima Indians Diabetes dataset. In Computing Networking and Informatics (ICCNI), 2017 International Conference on (pp. 1-5). IEEE.
52. Vapnik, V. (2013). The nature of statistical learning theory. Springer science & business media.
53. World Health Organization. 10 facts on diabetes. <http://www.who.int/features/factfiles/diabetes/en/ >. Accessed on 2018-09-08.
54. Zhang, Y., Lu, S., Zhou, X., Yang, M., Wu, L., Liu, B., Phillips, P. & Wang, S. (2016). Comparison of machine learning methods for stationary wavelet entropy-based multiple sclerosis detection: decision tree, k-nearest neighbors, and support vector machine. Simulation, 92(9), 861-871.
55. Zheng, B., Yoon, S. W., & Lam, S. S. (2014). Breast cancer diagnosis based on feature extraction using a hybrid of K-means and support vector machine algorithms. Expert Systems with Applications, 41(4), 1476-1482.
56. Zheng, B., Zhang, J., Yoon, S. W., Lam, S. S., Khasawneh, M., & Poranki, S. (2015). Predictive modeling of hospital readmissions using metaheuristics and data mining. Expert Systems with Applications, 42(20), 7110-7120.
57. Zhu, J., Xie, Q., & Zheng, K. (2015). An improved early detection method of type-2 diabetes mellitus using multiple classifier system. Information Sciences, 292, 1-14.