Improved K-Means with Adaptive Divergence Weight Bat Algorithm (IKM-ADWBA) and Feature Selection for Type 2 Diabetes Mellitus Prediction

M. Ashok Kumar, I. Laurence Aroquiaraj

Abstract--- Increase in blood glucose (hyperglycaemia) leads to Diabetes Mellitus. There are two kinds of Diabetes mellitus: (Type 1 Diabetes Mellitus (T1DM) and (Diabetes Mellitus (T2DM), then former one is dependent on insulin and the latter one is independent of insulin. Various factors make it difficult to diagnose it. SO the author focuses at binging-in and analyzing the method for making a novel robust diagnosis system using data mining methods. Complete datasets is necessary for data mining techniques, but these techniques doesn't give accurate results with missing values and all features. So, for prediction, Handling Missing value replacement and selection of important features are becomes a major issue. Hence, Adaptive Neuro Fuzzy Inference System (ANFIS) were proposed to acquire the missing value in dataset and to rectify the above mentioned issue. Then for an effective seed selection in Improved K-means algorithm, Enhanced Inertia Weight Binary Bat Algorithm (EIWBBA) is proposed, which results in high convergence speed. This research work proposed for feature selection with the help of Improved Distributed Kernel based Principal Component analysis (IDKPCA) with less time, after minimizing the entire feature space to the best features set. Then for classification of clustered samples, the author brought-in the Support Vector Machine (SVM). The experimental result confirms that the proposed algorithm gives the best classification accuracy rate when compared with other methods. From Pima Indians Diabetes, the data set has been considered and the experiment is done with the help of MATLAB for examining the Knowledge and the results were distinguished with other outcomes using appropriate toolkits.

Key words--- Diabetes Mellitus Prediction, Adaptive Neuro Fuzzy Inference System (ANFIS), Improved Distributed Kernel based Principal Component Analysis (IDKPCA), Improved K-Means Algorithm with Enhanced Inertia Weight Binary Bat Algorithm (IKM-EIWBBA), Support Vector Machine (SVM) Classification.

I. INTRODUCTION

Different chronic diseases were distributed throughout the world, both in the developing and developed country. Among them diabetes mellitus is one of the chronic diseases in the world which cut human life at early age [1]. The chronic metabolic disorder diabetes mellitus is a fast-growing global problem with huge social, health, and economic consequences. The prediction says that in 2010 there were globally 285 million people (approximately 6.4% of the adult population) suffering from this disease. This number is measured, which is increase to 430 million in the absence of better control or cure [2]. This results in the trend of urbanization and life style changes, including a "western-style" diet. This is because of less awareness [3]. An ageing population and obesity are two main reasons for the increase.

Diabetes mellitus is just termed as diabetes and it is a syndrome of disordered metabolism, generally because of a combination of hereditary and environmental causes, resulting in abnormally high blood sugar levels (hyperglycemia). The tedious interaction of multiple chemicals and hormones in the body, including the hormone insulin made in the beta cells of the pancreas [3] controls the blood glucose level.

High blood glucose level causes diabetes mellitus, which occurs because of either insulin secretion or insulin action. Frequent urination, increased thirst, weight loss, slow-healing in wound, giddiness, increased hunger etc, were the symptoms of this issue and it result in serious health complications including heart disease, blindness, kidney failure and low-extremity amputations [4]. It is classified into two types: the one which is dependent on insulin (Type 1 Diabetes Mellitus (T1DM) and the one which does not depend on the insulin (Type 2 Diabetes Mellitus (T2DM).

Type1 Diabetes is also termed as juvenile-onset diabetes and it includes Autoimmune, genetic, and environmental factors. Type1 mostly occurs in young people who are below 30 years [5] and it affects the children or adults, but majority of this diabetes affects children. Here, insulin secretions were increased by beta cells of the pancreas, which is diminished by autoimmune system.

Type 2 is also known as adult-onset diabetes [6], here the insulin produced by pancreas is not enough for the body's needs, or the body's cells are resistant to it. Older age, obesity, family history of diabetes, prior history of gestational diabetes, impaired glucose tolerance, physical inactivity, and race/ethnicity are the risk factors here.

Diabetes is also a hereditary disease which follows for generations. Gestational diabetes is the 3rd type of diabetes, which affects the women at the time of the gestation period. Pregnant women will produces high blood sugar level without the previous history of diabetes [7], but it disappears after delivery.

Manuscript received September 16, 2019.

M. Ashok Kumar, Research Scholar, Dept. of Computer Science, Periyar University, Salem, T.N, India. E-mail: williamashok@gmail.com

Dr.I. Laurence Aroquiaraj, Assistant Professor, Dept. of Computer Science, Periyar University, Salem, T.N, India. (E-mail: laurence.raj@gmail.com)

Proper medication will cure this issue or it will become type 2 diabetes. Irrespective of age group, it affects all people. Normal health issues like Blood pressure, plasma glucose also leads to diabetes.

Frequent urination, being thirsty always, weight reduction, hunger and irritations [8] are some symptoms of diabetic patients, which, in turn leads to other complications like visual impairment, cardiovascular diseases and kidney diseases and these makes to diagnose it in the later stage. But if it is recognized in the earlier stage, we can reduce the risk of diabetes and also other diseases caused by this. To make it happen we need to make use of advanced information technology.

Data mining technique which is also known as Knowledge Discovery in Databases (KDD), helps to recognize this problem in the early stage, and it is defined as the computational process of discovering patterns in large datasets involving methods at the intersection of artificial intelligence, machine learning, statistics, and database systems.

Pattern recognition, prediction, association, and clustering [9] were the main purposes of this technique and it comprises of series of steps disposed automatically or semi-automatically, for extracting and discovering the interesting, unknown, hidden features from large quantities of data. It has two important features: high quality of data and the properly applied method. In health care systems this technique helps to the core and it reduces the cost. If the person is affected by diabetes, a predictive analysis is done for detection.

For classifying occurrence of diabetes in the human being, numerous computational techniques were established. Machine learning in the medical information system has proved to be beneficial as it maximizes the diagnostic accuracy, reduces cost and increases the number of successful treatments for diabetes mellitus [11]. Diabetic database is necessary for automating the overall process of diabetes prediction and severity estimation. This repository of diabetic database assists in recognizing the impact of diabetes on various human organs.

Introducing and analyzing the method for making a novel robust intelligent diagnosis system based on training data with missing values and lower dimension of the clinical attributes is the primary target of this research work. The proposed model is subjected to various pre-processing techniques and the model comprises of the Improved K-means algorithm with EIWBBA (IKM-EIWBBA) for seed selection and feature selection were performed by Improved Distributed Kernel based Principal Component Analysis (IDKPCA).

Finally the Support Vector Machine gives the best classification accuracy rate of the proposed intelligent diagnosis system for estimating the diabetes Mellitus (DM) at an early stage to save human life.

The rest of the research discussion is organized as follows: Section 1. Describe the introduction about the significance of prediction of diabetes mellitus using data mining techniques, Section 2. Explain a review of the related technical literature and its classification methodologies, Section 3 Gives the brief explanation of data preprocessing and classification methods used in the proposed hybrid model, Section 4. Experimental results and discussion to investigate the effectiveness of the proposed method are shown. Finally, Section 5 contains a summary of conclusions.

II. LITERATURE REVIEW

A hybrid algorithm of Modified-Particle Swarm Optimization and Least Squares Support Vector Machine for the classification of type II DM patients was established by Soliman et al [12]. For classification LS-SVM algorithm is utilized by recognizing the optimal hyper-plane which separates various classes.

Modified-PSO algorithm is used as an optimization technique for LS-SVM parameters, because LS-SVM is so sensitive to the changes of its parameter values and it assures the robustness of the hybrid algorithm by searching for the optimal values for LS-SVM parameters. Pima Indians Diabetes Data set from UCI repository of machine learning databases helps in executing and computing the suggested algorithm.

A new Hybrid Genetic Classifier Model (HGCM) for the prediction of type 2 diabetes was proposed by Sreedevi et al [13] which integrate different distance methods as fitness function in GA Classifier & feature Selection method for getting classification accuracy. Two rules are generated for the prediction of type 2 diabetes, by HGCM. Then for recognizing the optimal solution by cleaning out the worse gene strings based on a fitness function, we make use of Genetic algorithm (GA) is considered to be an optimal search algorithm. Unsupervised data classification was rectified by GA.

The idea of modified extreme learning machine was analyzed by Priyadarshini et al [14] for recognizing the patients of being diabetic or non-diabetic basing on some previously provided data which assist the medical people for detecting whether someone is affected by diabetes or not. The application of two popular machine learning methods: Back propagation neural network and modified Extreme learning machine were examined and distinguished and it helps as binary classifiers to mention the diabetes prediction problem. To classify patients affected by diabetes a method was suggested by Mercaldo et al [15], which make use of a set of characteristic selected in according to Hoeffding Tree algorithm. Hence it has the ability to distinguish among diabetes affected patients and not affected ones using machine learning algorithm. The Pima Indian population near Phoenix in Arizona computes this method on a real-world data. Training the model using six different classification algorithms, acquires a precision equal to 0.757 and the recall of 0.762 after the best features selection step. Clinical decision support systems was proposed by Perveen et al [16], which includes several data mining techniques for diabetes prediction and course of progression.

This proposed method follows the adaboost and bagging ensemble techniques with the help of J48 (c4.5) decision tree as a base learner along with standalone data mining technique J48 to categorize the patients with diabetes mellitus through diabetes risk factors.

Across three different ordinal adults groups in Canadian Primary Care Sentinel Surveillance network, we classify this. Experimental result confirms that the performance of adaboost ensemble method is better when compared with bagging as well as standalone J48 decision tree.

A Deep Neural Network framework for diabetes data classification using stacked auto encoders was proposed by Kannadasan et al [17]. Features were extracted from the dataset through stacked auto encoders and the dataset is classified with the help of softmax layer. In supervised fashion with the training dataset, fine tuning of the network is performed through backpropagation. But, the medical diagnosis includes the risk factors of wrong prediction; hence used evaluation metrics like precision, recall, specificity and F1 - score for the evaluation of this model and have achieved better results. With Pima Indians Diabetes data, the experiments were performed and it has 768 patient records with 8 attributes for each record. This method accomplishes classification accuracy of 86.26%.

A novel model based on data mining techniques for predicting type 2 diabetes mellitus (T2DM) was proposed by Wu et al [18]. The model is comprised of two parts, the improved K-means algorithm and the logistic regression algorithm, according to the series of preprocessing procedures. Here we make use of Pima Indians Diabetes Dataset and the Waikato Environment for Knowledge Analysis toolkit. The experimental analysis shows that the model obtains a 3.04% higher accuracy of prediction when compared with existing techniques. Moreover, this model assures that the dataset quality is sufficient. As a result, the model is shown to be useful for the realistic health management of diabetes.

A model was designed by Sisodia et al [19], which can prognosticate the probability of diabetes in patients with maximum accuracy. Hence, three machine learning classification algorithms such as Decision Tree, SVM and Naive Bayes were used in this experiment to recognize the diabetes at an early stage. In Pima Indians Diabetes Database (PIDD) the experiments were performed, which is the source from UCI machine learning repository.

The performances of all the three algorithms were measured on different measures such as Precision, Accuracy, F-Measure, and Recall. The experimental analysis shows that Naive Bayes outperforms with the highest accuracy of 76.30% comparatively other algorithms. These results were confirmed with the help of Receiver Operating Characteristic (ROC) curves in a proper and systematic manner.

A model was brought-in by Samant et al [20], which tries to compute the diagnostic validity of an old complementary and alternative medicine technique, iridology for diagnosis of type-2 diabetes through soft computing methods. Over a close group of total 338 subjects (180 diabetic and 158 nondiabetic) the analysis were done and then the infra-red images of both the eyes were captured simultaneously. The region of interest from the iris image was cropped as zone with respect to the position of pancreas organ according to the iridology chart. From the region of interest, statistical, texture and discrete wavelength transformation features were extracted and the result confirms best classification accuracy of 89.63% computed from RF classifier. Maximum specificity and sensitivity were absorbed as 0.9687 and 0.988, correspondingly. Results disclose the effectiveness and diagnostic significance of proposed model for non-invasive and automatic diabetes diagnosis.

A potentially useful alternative approach according to the support vector machine (SVM) techniques to classify persons with diabetes and pre-diabetes was presented by Yu, et al [21].

In order to established and confirm the SVM models we make use of two classification schemes: Classification Scheme I (diagnosed or undiagnosed diabetes vs. pre-diabetes or no diabetes) and Classification Scheme II (undiagnosed diabetes or pre-diabetes vs. no diabetes). To choose the sets of variables that would yield the best classification of individuals into these diabetes categories, the SVM models were utilized. Based on the area under the receiver operating characteristic (ROC) curve, were 83.5% and 73.2%, respectively, the discriminative abilities of the SVM models for Classification Schemes I and II were done. Support vector machine modeling is a promising classification approach for recognizing the persons with common diseases like diabetes and pre-diabetes.

Orabi et al [22], brought-in a way to help people by raising an alert for precautions. It is a prediction system for the diabetes disease, which estimate whether to be a candidate and at what age. The datasets are for Egyptian diabetes patients, 2/3 were utilized for training and 1/3 for testing. This system works according to the machine learning concept, and it makes use of decision tree technique. This contribution was new in the prediction system, by adding a regression technique with a randomization code to predict the age. The results were capable; the system estimates the diabetes incidents at what age, with accuracy 84 %.

Support Vector Machines (SVMs) was brought by Barakat et al [23], for examining diabetes. Specifically, we utilize an extra explanation module which turns the “black box” model of an SVM into an intelligible representation of the SVM’s diagnostic (classification) decision. Results on a real life diabetes data set confirms that intelligible SVMs gives a promising tool for the prediction of diabetes where a comprehensible rule set have been developed, with prediction accuracy of 94%, sensitivity of 93% and specificity of 94% .

III. PROPOSED METHODOLOGY

Intelligent system was proposed for estimating the diabetes mellitus and normal patient from the dataset. The proposed method comprises of three stages: data preprocessing, feature selection and classification.

The missing value analysis in data preprocessing stage is significant parts of robust pattern diagnosis. If the missing value doesn't manage properly, it results in an inaccurate inference about real class of the patients. SO, we make use of ANFIS model to acquire the missing value. Then for an effective seed selection in Improved K-means algorithm, Enhanced Inertia Weight Binary Bat Algorithm (EIWBBA) is proposed.

The model will be more complex, if the dimension of dataset is not reduced by selecting the important features. Hence, the accuracy of classifier is reduced through class detection for rectifying the above mentioned problem, the proposed Improved Distributed Kernel based Principal Component Analysis (IDKPCA) helps for feature selection. In the classification stage, Support Vector Machine (SVM) is enforced to categorize the diabetic disease with the effective results.

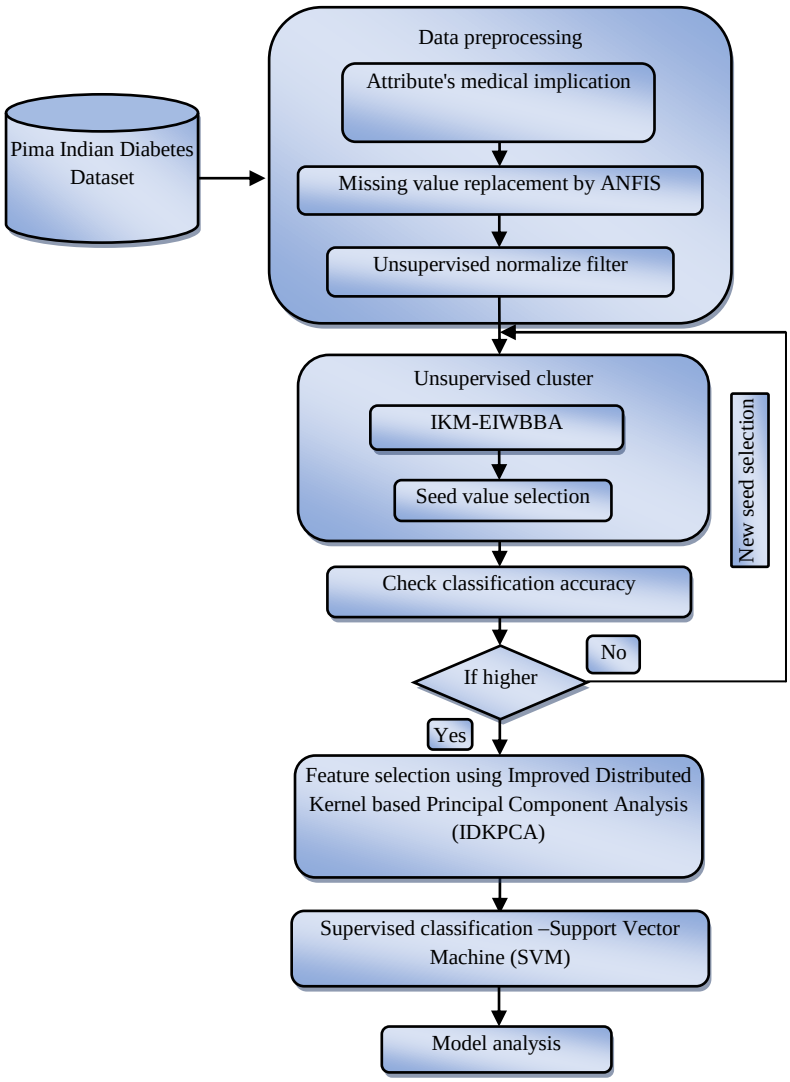

Figure 1: Block Diagram Of the Proposed Intelligent System for Prediction of Diabetes Disease

Dataset Description

Pima Indian Diabetes Dataset is assumed here and it has 768 medical records out of which 268 tested positive and the other negative instances coming down from a population from the regions of Phoenix, Arizona, USA [24]. Positive one represents that the patient is diabetic and the negative represents that the patient is not diabetic. Every instance has eight attributed and all are numeric data type and these values contain personal medical related data and the data acquired from the results of medical examinations. The entire details of the attributes helps in the dataset are Number of times pregnant (preg), Plasma glucose concentration at 2 h in an oral glucose tolerance test(plas), Diastolic blood pressure (pres), Triceps skin fold thickness (skin), 2-h serum insulin (Insu), Body mass index (BMI), Diabetes pedigree function (Pedi), Age (age) and Class variable (class).

The predictions works on the data quality and it makes the preprocessing a significant task which cannot be neglected in the model [22]. The MATLAB is equipped with various filters which favor the preprocessing and most appropriate methods were chosen for the optimization of the original dataset. Initially, the medical implication of every attribute is examined in correlation to the DM. The attribute "no of pregnancies "is defined to have low impact on the DM[14] and this numeric value is changed to a nominal value by assigning 0 for non-pregnant and 1 for pregnant. This resulted in the minimization of the data complexity.

Missing Value Replacement using Adaptive Neuro Fuzzy Inference System (ANFIS)

The deregulations causes incorrect and some missing values in the dataset, which is main reason for various in-correct results in most of the experiments are identified and removed. For example: the value cannot be 0 for the diastolic blood pressure and body mass index and if so in dataset, it points to the real value is missing [25]. Adaptive Neuro Fuzzy Inference System (ANFIS) helps to rectify this hindrance by recognizing the missing values.

The proposed ANFIS makes use of a hybrid learning procedure, which builds an input-output mapping according to the stipulated input-output data pairs and human knowledge (in the form of fuzzy if-then rules).

Working of ANFIS

The architecture and learning procedure fundamental ANFIS (Adaptive Network- based Fuzzy Inference System) is given, which is a fuzzy inference system executed in the framework of adaptive networks [25].

Building the model: The model is build, by creating, training, and testing Adaptive Network-based Fuzzy Inference System (ANFIS). The following tasks were performed.

1. **Data pre-processing**: In this step randomly generating training data and testing data in the Input Pima Dataset for examining the results.
2. **Loading the data:** The initial step in training the ANFIS is loading the data set which comprises of desired input/output data of the model which is considered. The structure of the data set to be loaded must be an array in which data is arranged in column vectors and the last column with output data. And also loaded the testing data.
3. **Generating the Initial ANFIS Structure**: The Initial ANFIS Structure is produced, before initiating the ANFIS training, generating
4. **Training the ANFIS:** Begin the training the ANFIS. The number of training Epochs (Epochs means number of iterations) is over 1000, after loading the training data and generating the initial ANFIS structure
5. **Validating the Trained ANFIS:** Finally, to test the data against the trained ANFIS. After the ANFIS is trained, validate the model with the help a testing data that varies from the one used to train the ANFIS. When test the testing data against the ANFIS, it looks satisfactory.

Concept and structure: For acquiring the good reasoning in quality and quantity [26], ANFIS is a combination of the intelligent approaches in neural network and fuzzy logic. Hence, the network acquired through reasoning in quality and quantity [26]. therefore the network acquired through fuzzy logic has extra ordinary capacity of training by virtue of neural networks and linguistic interpretation of variables. The both of them encode the information in parallel and distribute architecture in a numerical framework.

Rule: if x is A1 and y is B1 then f (x) = px + qy + r

Where x and y are the inputs, A and B are the fuzzy sets, f are the output, p, q and r are the design parameters that defined at the time of the training process. This network is constructed with the help of five layers and each of these layers is having several nodes.

Layer 1: implements a fuzzification process which represents membership functions (MFs) to each input. In this work choose Gaussian functions as membership functions:

$$O_i^1 = \mu_{A_i} = \exp\left[\frac{-(x-c)^2}{\sigma^2}\right] \quad (1)$$

Layer 2: implements the fuzzy AND of antecedents part of the fuzzy rules

$$O_i^2 = W_i = \mu_{A_i}(X_1) \times \mu_{B_i}(X_2), \qquad i = 1,2,3,4 \quad (2)$$

Layer 3: normalizes the Membership Function (MFs)

$$O_i^3 = \overline{W_i} = \frac{W_i}{\sum_{j=1}^{4} W_j}, \qquad i = 1,2,3,4 \quad (3)$$

Layer 4: implements the conclusion part of fuzzy rules

$$O_i^4 = \overline{W_i}Y_i = \overline{W_i}\left(\propto_1^i x_1 + \propto_2^i x_2 + \propto_3^i x_3\right), \quad i = 1,2,3,4 \quad (4)$$

Layer 5: measures the output of fuzzy system by summing up the outputs of the fourth layer which is the de-fuzzification process.

Seed Selection using Improved K-Means Clustering Algorithm with Enhanced Inertia Weight Binary Bat Algorithm (IKM-EIWBBA)

K-means is considered as the familiar algorithms for performing the cluster and it is a conventional distance based method and the distance is the measure of the similarity which define the short distance that tend to high similarity to Show all objects. Select K from the given N as the number of initial cluster center [27]. Measure the distance among each object and cluster center 'm'. Cluster every object to the nearest cluster works according to the distance using (5)

$$S_i^{(t)} = \left\{\forall j, \left|\left|x_p - m_i^{(t)}\right|\right|^2 \leq \left|\left|x_p - m_j^{(t)}\right|\right|^2 \forall j, 1 \leq j \leq k\right\} \quad (5)$$

Recalculate every cluster center to confirm whether they are changed using (6).

$$m_i^{(t+1)} = \frac{1}{\left|S_i^{(t)}\right|} \sum_{x_j \in S_i^{(t)}} x_j \quad (6)$$

Until the new cluster center is the same as the original one, i.e., convergence and end of the algorithm is circulated with the aboe mentioned step.

Here, the value of K cause the variable class has t=couple of results. It utilizes the data which are processed and only after the preprocessing. The significant demerits while using the K means algorithm is that the random production of the initial seed values but it requires being set based on the experience. The value of the seed affects the results in a straight forward manner.

Few steps were considered to assure that there is no deviation of the results after experiments that are caused because of the randomness of the value of the seed.

Insert a program is the basic step, which can sort and record the value referred as Within cluster sum of squared errors' in an ascending order. Here, a seed has a definite value known as 'within cluster sum of squared errors'. Higher will be the accuracy in the results, if this value is smaller. Ten thousand values are recorded with respect to the same value of the seed in a range between one and ten thousand. The seeds that have a high quality will be initially placed to utilize in the upcoming step. The initial value of the seed is performed by Binary Bat Algorithm (BBA).

A loop to be set at the last part of the algorithm, we will indulge on the second step and it assist in elimination of the data that are clustered in a non appropriate manner and it make use of the formula given in (7). If the computed rate is more than 75%, the same is put in the next level. Else the loop is exited and other seed value is selected. The most proximate rate helps when there is a situation of no need value having the rate greater than 75% after all the loops been executed and or after 60 seconds.

Rate = Remaining data / Sum (7)

Enhanced Inertia Weight Binary Bat Algorithm (EIWBBA)

The bat algorithm is stimulated by the echolocation behavior of bats. They will reduce the loudness and increase the frequency of emitted ultrasonic sound, when bats chase preys. These characteristics of real bats help in establishing the BA [27]. These basic steps of BA have been mathematically described as follows**.** In the BA, every bat has three vectors, along with the frequency vector, a velocity vector, and a position vector that are updated at time step t as (8), (9), and (10):

$$V_i(t+1) = V_i(t) + (X_i(t) - Gbest)F_i \quad (8)$$

$$X_i(t+1) = X_i(t) + V_i(t+1), \quad (9)$$

where $Gbest$ indicates the best position obtained so far and F_i indicates the frequency of ith bat which is updated as follows:

$$F_i = F_{min} + (F_{max} - F_{min})\beta, \quad (10)$$

where β in the range of [0, 1] is a random vector drawn from a uniform distribution. From (8) and (10), it is understandable that various frequencies endorse the exploration capability of bats to the optimal solution.

These equations, to a certain extent, assure the exploitation capability of the BA. But, a random walk operation has also been employed to execute the intensification better, as follows:

$$X_{new} = X_{old} + \varepsilon At, \quad (11)$$

where X_{old} means one solution selected randomly between the current best solutions, ε is a randomly selected number in the range of [−1, 1], and A represents the average loudness of all bats at this time step. Note that rand is a random number uniformly distributed in the range [0, 1]. To an extent, BA assumed as a balanced combination of global and intensive local search.

The pulse emission rate (r) and loudness (A) control the balancing among these two search techniques. As A increases, artificial bats tend to execute a diversification rather than intensification. These two parameters addressed above are updated as follows:

$$A_i(t+1) = \alpha A_i(t) \quad (12)$$

$$r_i(t+1) = r_i(0)[1 - \exp(-\gamma t)], \quad (13)$$

where α and γ are constants and α has the same meaning of the cooling factor. To assure that the artificial bats are navigating toward the optimal solutions, both loudness and emission rate are updated when the better solutions were recognized. But, the first and second items of the equation affect the algorithm so that it executes the global and local search, correspondingly.

It is confirmed that the first item of (8) may minimize the convergence rate rapidly and the second item of (8) may result in premature convergence problem. BBA is effectively the same as the original BA, for rectifying this problem,

The binary bat algorithm (BBA) was proposed to rectify optimization problems with binary search space.

The structure of BBA is almost the same as the original BA in which the velocity and frequency are determined in continuous space. BBA makes two modifications to the original BA:

(i)The vector of position is no longer a continuous valued vector but a bit string.

(ii) The random operation demonstrated by (11) is no longer suitable to binary search space. Instead, a simpler operation is adopted.

The position update equation for BBA changes to

$$x_i^k(t+1) = \begin{cases} (x_i^k(t))^{-1} & rand \le f\left(v_i^k(t+1)\right) \\ x_i^k(t) & rand > f\left(v_i^k(t+1)\right) \end{cases} \quad (14)$$

Where

$$f\left(v_i^k(t)\right) = \left|\frac{2}{\pi}\arctan\left(\frac{\pi}{2}v_i^k(t)\right)\right| \quad (15)$$

and $x_i^k(t)$ and $v_i^k(t)$ represent the position and velocity of ith artificial bat at iteration t in kth dimension and $(x_i^k(t))^{-1}$ indicates the complement of $x_i^k(t)$.

The operation demonstrated by (4) for BBA changes to

$$X_{new} = X_{old} \quad (16)$$

Inertia Weight Strategies

Where X_{old} indicates a solution selected randomly from the current best solutions. So inertia weight strategy plays a significant part in the process of maintaining the balance among the global search and local search process.

In existing step, the inertia weight strategy defines the contribution proportion of old velocity to its new velocity.

Hence, Inertia weight strategies which oversees the search situation and adjust the inertia weight value according to one or more feedback parameters.

It is obvious that this equation consists of two parts, when the velocity update equation (8) is analyzed.

The first item ($V_i(t)$) indicates the velocity of population and the second item ($((t) - Gbest)$) controls the velocity of the ith position ((t)) with guidance of the global best solution ($Gbest$).

To give a better solution, the guidance of the neighbor bat (kth solution) is utilized. So, the velocity update equation of original BBA is modified as follows:

$$V_i(t+1) = \omega\big(V_i(t)\big) + (X_i(t) - Gbest)F_i\delta_1 + (X_i(t) - X_k(t))F_i\delta_2 \quad (17)$$

$$\delta_1 + \delta_2 = 1 \quad (18)$$

where w represents the inertia weight factor which balances local and global search intensity of the ith solution by controlling the value of old velocity (t), X_k indicates one of the best solutions randomly selected from the population ($i \neq k$), δ_1 is self-adaptive learning factor of global best solution ($Gbest$) ranging from 0 to 1, and therefore, δ_2, which is a learning factor of kth solution, ranges from 1 to 0. Since the kth solution information helps to guide the ith solution, the algorithm avoids the local minima. As δ_1 is increased, the effect of the global best solution ($Gbest$) becomes superior when compared with the kth neighbor solution (X_k) an

$$\delta_1 = 1 + (\delta_{init} - 1)\left(\frac{iter_{max} - iter}{iter_{max}}\right)^n, \quad (19)$$

where δ_{init} represents the initial impact factor of δ_1, $\text{iter}_{\max}$ represents the maximum number of iterations, iter represents the current number of iterations, and n represents a nonlinear modulation index. As iter is increased, δ_1 will increase from δ_{init} to 1 nonlinearly and δ_2 will decrease from $(1 - \delta_{\text{init}})$ to 0 correspondingly. With a small δ_1 and a large δ_2, bats are allowed to fly around the search space, rather than flying toward. Else, a large δ_1 and a small δ_2 allow the bats to converge to the global optimum solution in the latter stages of the search process. Hence, the proposed approach can effectively manage the global search and improve the convergence to the global best solution at the time of the latter part of the optimization.

The inertia weight strategy helps to manage the magnitude of the velocity. This strategy is illustrated as follows:

$$w = w_{max} * \exp\left(-m * \left(\frac{iter}{iter_{max}}\right)^m\right), \quad (20)$$

where $\text{iter}_{\max}$ represents the total number of iterations, iter indicates the current number of iterations, maximal inertia values are represented by$w_{\max}$, and m is a constant larger than1. The advantages of the proposed improved binary bat algorithm (IBBA) are that it contributes to the dispersion of the solutions into binary search space. In addition, more accurate results can be acquired.

Feature selection using Improved Distributed Kernel based Principal Component Analysis (IDKPCA)

The number of attributes can be minimized by the suggested Feature selection methods, which get-rid off the redundant features using improved distributed kernel based Principal Component analysis (IDKPCA) is a powerful technique for extracting structure from either high dimensional data set. Principal Component Analysis (PCA), measure the low dimensional subspace capturing as much of the variance of the union of their point sets as possible. The components in lower dimensional space are also known as principal components. These principle components must be independent when the data set is normally distributed. PCA is generally sensitive to the scaling of the original variables. The dimensionality of input feature vector gets modified by assuming the small eigen values as zeros. The accuracy of the acquired results works according to the selected threshold value of an eigen value. The significant properties of an approximate improved distributed PCA algorithm are its communication cost and computational efficiency for a provided desired accuracy is less accurate while enforcing the covariance matrix. The distributed PCA only permits the linear dimensionality reduction. But, if the data has more complicated structures which can't be well indicated in a linear subspace, improved distributed PCA isn't useful. Fortunately, kernel PCA allows to improved distributed PCA to a nonlinear dimensionality reduction.

Improved Distributed kernel based Principal Component Analysis

In the distributed setting, consider a set of s nodes V = $\{v_i, 1 \leq i \leq s\}$, each of which can inteact with a central coordinator v_0. There is a local data matrix $P_i \in R^{ni \times d}$ having n_i data points in d dimension ($n_i > d$), on each node v_i,. The global data $P \in R^{n \times d}$ is then a concatenation of the local data matrix, i.e. $P^T = [P_1^T, P_2^T, \ldots\ldots P_s^T]$ and $n = \sum_{i=1}^{s} n_i$. Let p_i represent the i-th row of P. Here consider that the data points are centered to have zero mean, i.e., $\sum_{i=1}^{s} p_i = 0$. Uncentered data requires a rank-one modification to the algorithms, whose communication and computation costs are dominated by those in the other steps.

Constructing the Kernel Matrix

Assume a nonlinear transformation $\phi(x)$ from the original D-dimensional feature space to an M-dimensional feature space, where usually M≫D. Then each data point x_i is projected to a point $\phi(x_i)$. Perform standard PCA in the new feature space, but it is costly and inefficient. Fortunately, we make use of kernel methods to make simpler the computation.

First, consider that the projected new features have zero mean,

$$\frac{1}{N}\sum_{i=1}^{N}\phi(x_i) = 0 \quad (21)$$

The covariance matrix of the projected features is M × M, calculated by

$$C = \frac{1}{N}\sum_{i=1}^{N}\phi(x_i)\phi(x_i)^T \quad (22)$$

Its eigen values and eigenvectors are provided by

$$Cv_k = \lambda_k v_k \quad (23)$$

where k = 1, 2, · · · , M. From Eq. (22) and Eq. (23), have

$$\frac{1}{N}\sum_{i=1}^{N}\phi(x_i)\{\phi(x_i)^T v_k\} = \lambda_k v_k \quad (24)$$

which can be rewritten as

$$v_k = a_{ki}\phi(x_i) \quad (25)$$

Now by substituting v_k in Eq. (24) with Eq. (25), have

$$\frac{1}{N}\sum_{i=1}^{N}\phi(x_i)\phi(x_i)^T\sum_{j=1}^{N}a_{kj}\phi(x_j) = \lambda_k\sum_{i=1}^{N}a_{ki}\phi(x_i) \quad (26)$$

The kernel function is defined as,

$$k(x_i, x_j) = \phi(x_i)^T\phi(x_j) \quad (27)$$

and multiply both sides of Eq. (27) by $(x_l)^T$, have

$$\frac{1}{N}\sum_{i=1}^{N}k(X_l, X_i)\sum_{j=1}^{N}a_{kj}k(X_i, X_j) = \lambda_k\sum_{i=1}^{N}a_{ki}k(X_l, X_i) \quad (28)$$

In this use the matrix notation as,

$$K^2a_k = \lambda_k NKa_k \quad (29)$$

Where

$$K_{i,j} = k(X_i, X_j) \quad (30)$$

and a_k is the N-dimensional column vector of a_{ki}:

$$a_k = [a_{k1}\ a_{k2}\ \dots\ a_{kN}]^T \quad (31)$$

a_k can be solved by

$$Ka_k = \lambda_k Na_{k,} \quad (32)$$

and the resulting kernel principal components can be calculated using

$$y_k(X) = \phi(X)^T v_k = \sum_{i=1}^{N}a_{ki}k(X, X_i) \quad (33)$$

The power of kernel methods is that they don't have to measure $\phi(x_i)$ explicitly. Directly construct the kernel matrix from the training data set $\{x_i\}$. Two commonly utilized kernels are the polynomial kernel

$$k(x, y) = (x^T y)^d \quad (34)$$

or

$$k(x, y) = (x^T y + c)^d, \quad (35)$$

Where c>0 is a constant, and the Gaussian kernel

$$k(x, y) = \exp(-\|x - y\|^2/2\sigma^2 \quad (36)$$

with parameter σ. Approximate PCA and ℓ_2 - Error Fitting. For a matrix A = $[a_{ij}]$, let $\|A\|_F^2 = \sum_{i,j}a_{ij}^2$ be its Frobenius norm, and let $\sigma_i(A)$ be the i-th singular value of A [28]. Let $A^{(t)}$ represent the matrix that comprises of the first t columns of A. Let L_X represent the linear subspace spanned by the columns of X. For a point p, let $\pi_L(p)$ be its projection onto subspace L and let $\pi_X(p)$ be shorthand for π_{LX} (p). For a point $p \in R^d$ and a subspace $L \subseteq R^d$, represent the squared distance between p and L by

$$d^2(p, L) := \min_{q\epsilon L}\|p - q\|_2^2 = \|p - \pi_L(p)\|_2^2 \quad (37)$$

Definition 1. The linear (or affine) r-Subspace k-Clustering on $P \in R^{n\times d}$ is

$$\min_{\mathcal{L}} d^2(P,\mathcal{L}) := \sum_{i=1}^{n}\min_{L\epsilon\mathcal{L}} d^2(p_i, L) \quad (38)$$

where P is an n × d matrix whose rows are $p_1, \dots, p_n$, and $\mathcal{L} = \{L_j\}_{j=1}^{k}$ is a set of k centers, each of which is an r-dimensional linear (or affine) subspace.

PCA is a special case when k = 1 and the center is an r-dimensional subspace. This optimal r-dimensional subspace is spanned by the top r right singular vectors of P, also called as the principal components, and it is recognized with the help of the singular value decomposition (SVD). Another special case of the above is k-means clustering when the centers are points (r = 0). Primarily be concerned with relative-error approximation algorithms, for which would like to output a set L' of k centers for which $d^2(P, L') \le (1 + \epsilon)\, min_{\mathcal{L}}\, d^2(P, \mathcal{L})$.

For approximate distributed PCA, the following protocol is implicit in [29]: each server i measures its top $O(r/\epsilon)$ principal components Y_i of P_i and broadcasts them to the coordinator. The coordinator stacks the $O(r/\epsilon) \times d$ matrices Y_i on top of each other, forming an $O(sr/\epsilon) \times d$ matrix Y, and measures the top r principal components of Y, and returns these to the servers. This gives a relative-error approximation to the PCA problem.

- **Improved Communication:** To enhance the communication cost for utilizing the distributed PCA for kmeans clustering and similar ℓ_2-fitting problems. Provided a data matrix P, if project the rows onto the space spanned by the top $O(k/\epsilon^2)$ principal components, and rectify the k-means problem in this subspace, obtain a $(1+)\epsilon$ −approximation. In the distributed setting, this would need first running Algorithm disPCA with parameter $r = O(k/\epsilon^2)$, and hence communication at least $O(skd/\epsilon^3)$ to compute the $O(k/\epsilon^2)$ global principal components. Then one can rectify a distributed k-means problem in this subspace, and an α-approximation in it translates to an overall $\alpha(1 + \epsilon)$ approximation.
- **Improved Computation**: A major drawback is measuring a singular value decomposition (SVD) of its point set P_i, which considers the $min(n_i d^2, n_i^2 d)$ time. Then change Algorithm disPCA to instead have each server first sample an oblivious subspace embedding (OSE) matrix H_i, and rather run the algorithm on the point set defined by the rows of H_iP_i.
- With the help of known OSEs, can select H_i to have only a single non-zero entry per column and thus H_iP_i can be measured in $nnz(P_i)$ time. Furthermore, the number of rows of H_i is $O(d^2/\epsilon^2$ which may be considerably less than the original n_i number of rows.
- This number of rows can be further minimized to $O(dlog^{o(1)}\, d/\epsilon^2)$, if willing to spend $O(nnz(P_i)\, log^{O(1)}\, d/\epsilon)$ time. Note that the number of non-zero entries of H_iP_i is no more than that of P_i.

Classification of Diabetes Disease using Support Vector Machine (SVM)

SVM is a global classification model that generates non-overlapping partitions and usually employs all attributes, as a classification method. A classification task includes the training and test sets which comprises of data instances each instance in the training set comprises of one target value (class label) and several attributes (features).

The target of a classifier generates a model able to predict target values of data instances in the testing set, for which only the attributes are known [30]. Without loss of generality, the classification problem is addressed as a two-class problem in which one's target is to separate the two classes by a function induced from available examples.

The aim is to generate a classifier that generalizes well, i.e. that works well on unseen examples.

But, only one increase the distance among itself and the nearest example of each class (i.e. the margin) and for that is known as the optimal separating hyperplane. It is intuitively expected that this classifier generalizes better when compared with the other options [4].

The basic concpet of SVM classifier makes use of this approach, i.e. to select the hyperplane that has the maximum margin.SVM simultaneously reduce the empirical classification error and increase the geometric margin. So SVM is termed as Maximum Margin Classifiers. SVMs can effectively perform non-linear classification using what is called the kernel trick, implicitly mapping their inputs into high-dimensional feature spaces.

The constructing the classifier without explicitly knowing the feature space is done by kernel trick.

An SVM model is an indication of the examples as points in space, mapped so that the examples of the individual categories are classified by a clear gap that is as wide as possible For example, provided a set of points belonging to either one of the two classes, an SVM recognizes a hyperplane having the huge possible fraction of points of the same class on the same plane. This separating hyperplane is also known as the optimal separating hyperplane (OSH) that increases the distance among the two parallel hyper planes and can reduce the risk of misclassifying examples of the test dataset. Given labeled training data as data points of the form.

$$M = \{(x_1, y_1), (x_2, y_2), \dots\dots, (x_n, y_n)\} \quad (39)$$

where $y_n = 1/-1$, a constant that denotes the class to which that point x_n belongs. Where, n=number of data sample. Each x_n is a p-dimensional real vector. The SVM classifier initially maps the input vectors into a decision value, and then executes the classification through an appropriate threshold value as shown in figure 2. To view the training data, divide the hyperplane, which can be described as:

$$Mapping: \quad w^T.x + b = 0 \quad (40)$$

where w is a p-dimensional weight vector and b is a scalar. The vector w points perpendicular to the separating hyper plane. The offset parameter b allows maximizing the margin. When the training data are linearly separable, we choose these hyper planes so that there are no points among them and then put an effort on maximizing the distance among the hyper plane. Here found out the distance among the hyper plane as 2/ |w|. To minimize |W|, need to ensure for all either

$$w.x_i - b \geq 1 \; or \; w.x_i - b \leq -1 \quad (41)$$

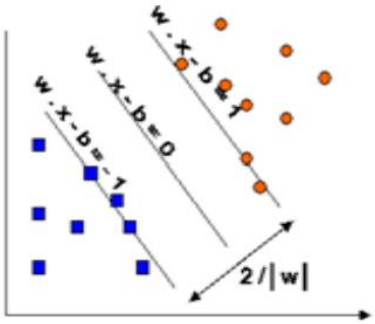

Figure 2: Maximum Margin Hyperplanes for SVM Trained with samples from Two Classes

Radial Basis Kernel Function

The Radial Basis Function (RBF) kernel of SVM helpsthe Classifier, as RBF kernel function can examine the higher dimensional data. The output of the kernel is dependent on the Euclidean distance of from (one of these will be the support vector and the other will be the testing data point). The support vector will be the center of the RBF and will define the area of influence this support vector has over the data space. RBF Kernel function can be defined as,

$$k(x_i, x_j) = \exp\left(-\gamma \left\| x_i - x_j \right\|^2\right) \quad (42)$$
$$\gamma > 0$$

where k is a kernel parameter and is the training vector. A larger value of it will produce a smoother decision surface and more regular decision boundary. This is because an RBF with huge value will allow a support vector to have a strong influence over a larger area. The best parameter set is enforced to the training dataset and the classifier is acquired. The designed classifier helps to categorize the testing dataset.

IV. RESULTS AND DISCUSSION

With the help of the MATLAB toolkit, it was suitable for us to examine the result of the experiment through a visualized interface. In general, the process of prediction comprises of four different results called true positive (TP), true negative (TN), false positive (FP), and false negative (FN). The results were computed with respect to precision, recall, sensitivity, specificity, f-measure, g-mean and accuracy.

The precision is caomputed by (27)

$$Precision(Pr) = \frac{TP}{TP + FP} \quad (43)$$

The recall is also called as the sensitivity, is measured by (28)

$$Recall(Re) = \frac{TP}{TP + FN} \quad (44)$$

Specificity (True negative rate) Specificity (SP) is computed as the number of correct negative predictions which is divided by the total number of negatives. It is also called true negative rate (TNR). It is calculated by (29)

$$\text{Specificity} = \frac{TN}{TN + FP} \quad (45)$$

F-measure is a measure of a test's accuracy. It assumed both the precision ‘pr’ and the recall ‘re’ of the test to measure the score: ‘pr’ is the number of correct positive results divided by the number of all positive results returned by the classifier, and ‘re’ is the number of correct positive results divided by the number of appropriate samples (all samples that should have been identified as positive). It is computed by (30),

$$\text{F} - \text{measure} = 2.\frac{Pr.Re}{Pr + Re} \quad (46)$$

G-mean is the geometric mean of sensitivity and precision. G-mean is calculated by (31)

$$G-mean = \sqrt{Sen * Pre} \quad (47)$$

Accuracy is the most intuitive performance measure and it is simply a ratio of correctly predicted observation to the total observations., calculated by (32)

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (48)$$

Table 2: Performance Comparison Metrics vs. Classifiers

Classifiers	Results (%)					
	Sensitivity	Specificity	Precision	F-measure	G-mean	Accuracy
IKM+LR	85.1852	84.7826	86.7925	85.9813	84.9837	85
IKM-ADWFA+LR	93.7500	88.9706	80	86.3309	91.3290	90.500
IKM-EIWBBA+SVM	94.15	89.16	87.54	88.57	92.87	91.87

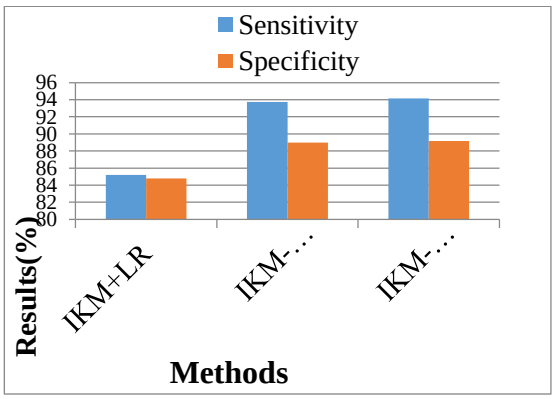

Figure 3: Performance Comparison Results of Sensitivity and Specificity vs. Methods

Figure 3 gives the performance comparison results of the current IKM+LR, IKM-ADWFA+LR and the proposed IKM-BA+SVM classifier. The proposed and existing results were computed among the sensitivity and specificity. From the results it confirmed that the proposed IKM-BA+SVM classifier gives superior sensitivity results of 93.75%, whereas existing IKM-ADWFA+LR gives 85.1852% and IKM+LR provides 94.15% only.

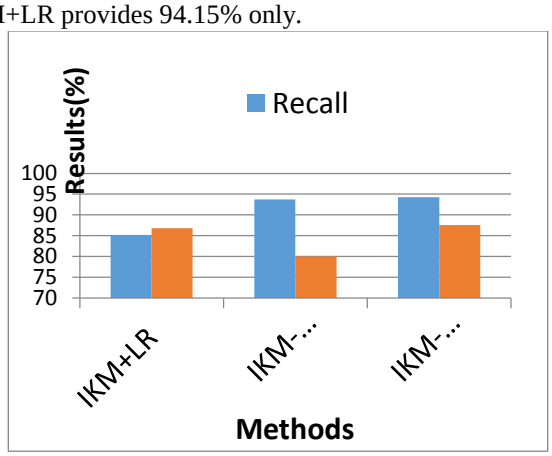

Figure 4: Performance Comparison Results of Precision and Recall vs. Methods

Figure 4 gives the precision and recall comparison results of the existing IKM+LR, IKM-ADWFA+LR and the proposed IKM-EIWBBA+SVM classifier. From the results it confirmed that the proposed IKM-EIWBBA+SVM classifier gives higher recall results of 94.27%, whereas existing IKM-ADWFA+LR provides 93.75% and IKM+LR gives 85.18% only.

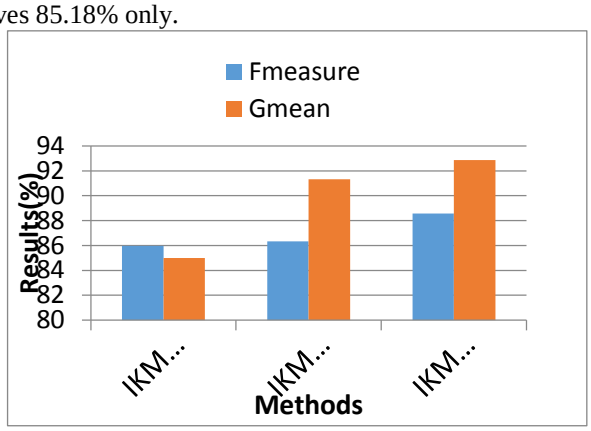

Figure 5: Performance Comparison Results of F-Measure and G-Mean vs. Methods

Figure 5 gives the f-measure and g-mean comparison results of the existing improved k means clustering with logistic regression (IKM+LR) model, IKM-ADWFA+LR and the proposed IKM-EIWBBA+SVM classifier. From the results it confirmed that the proposed IKM-EIWBBA+SVM classifier gives superior G-mean results of 92.87%, whereas existing algorithm IKM-ADWFA+LR provides 91.32% and IKM+LR gives 84.98% only.

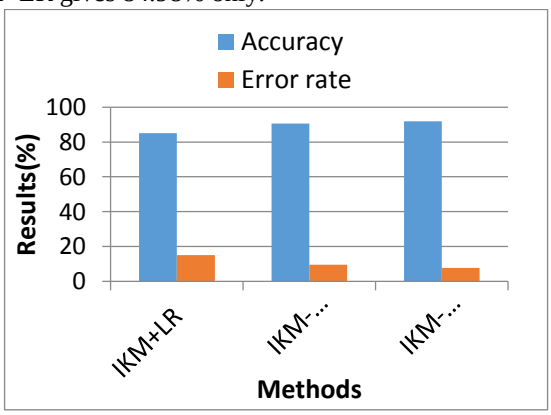

Figure 6: Performance Comparison Results of Accuracy vs. Methods

Figure 6 gives the accuracy and error rate comparison results of the existing improved k means clustering with logistic regression (IKM+LR) model, IKM-ADWFA+LR and the proposed IKM-EIWBBA+SVM classifier. From the results it confirmed that the proposed IKM-EIWBBA+SVM classifier gives superior accuracy results of 91.87%, whereas existing IKM-EIWBBA+SVM gives 90.5% and IKM+LR provides 85% only.

V. CONCLUSION

The detection of diabetes at its early stage is the significant real-world medical issue. Our work concentrates to bring-in a suitable prediction model for the high-risk T2DM group. According to a number of researchers' experiences, we bring-in a model, which comprises of double-level algorithms, i.e., the improved K-means clustering (IKM) and Enhanced Inertia Weight Binary Bat

Algorithm (EIWBBA) algorithms and it makes use of Adaptive Neuro Fuzzy Inference System (ANFIS), in order to acquire the missing value in dataset. Then for effective seed selection in Improved K-means algorithm, Enhanced Inertia Weight Binary Bat Algorithm (EIWBBA) is proposed, which results in high convergence speed. This research work proposed for feature selection using Improved Distributed Kernel based Principal Component analysis (IDKPCA) with less time, after reducing the whole feature space to the best features set. Then for classification of clustered samples, we bring-in the Finally Support Vector Machine (SVM). With the help of Pima Indians Diabetes Database the experiments were done and it defines the adequacy of the designed system with an accomplished accuracy of 91.87 % using the Support Vector Machine (SVM) classification algorithm. As a future work: the designed system with the used machine learning classification algorithms helps to estimate or diagnose other diseases.

The work can be extended and enhance the prediction of diabetes analysis along with some other machine learning algorithms.

REFERENCES

1. Joshi, R., & Alehegn, M. (2017). Analysis and prediction of diabetes diseases using machine learning algorithm: Ensemble approach. *International Research Journal of Engineering and Technology*, *4*(10).
2. Kaul, K., Tarr, J. M., Ahmad, S. I., Kohner, E. M., & Chibber, R. (2013). Introduction to diabetes mellitus. In *Diabetes* (pp. 1-11). Springer, New York, NY.
3. Lingaraj, H., Devadass, R., Gopi, V., & Palanisamy, K. (2015). Prediction of diabetes mellitus using data mining techniques: a review. *Journal of Bioinformatics & Cheminformatics*, *1*(1), 1-3.
4. Sengamuthu, R., Abirami, R., Karthik, D., (2018) various data mining techniques analysis to predict diabetes mellitus. In *International Research Journal of Engineering and Technology (IRJET),* 5(5) pp. 676-679.
5. Lakshmi, K. R., & Kumar, S. P. (2013). Utilization of data mining techniques for prediction of diabetes disease survivability. International Journal of Scientific & Engineering Research, 4(6), 933-940.
6. Nagarajan, S., Chandrasekaran, R. M., & Ramasubramanian, P. (2015). Data mining techniques for performance evaluation of diagnosis in gestational diabetes. *International journal of current research and academic review*, *2*(10), 91-98.
7. Daghistani, T., & Alshammari, R. (2016). Diagnosis of diabetes by applying data mining classification techniques. *International Journal of Advanced Computer Science and Applications (IJACSA)*, *7*(7), 329-332.
8. Sreedevi, E., & Padmavathamma, M. (2016). Design and Development of Hybrid Genetic Classifier Model for Prediction of Diabetes. *International Journal of Modern Trends in Engineering and Research*, *3*, 260-265.
9. Rajesh, K., & Sangeetha, V. (2012). Application of data mining methods and techniques for diabetes diagnosis. *International Journal of Engineering and Innovative Technology (IJEIT)*, *2*(3).
10. Perveen, S., Shahbaz, M., Guergachi, A., & Keshavjee, K. (2016). Performance analysis of data mining classification techniques to predict diabetes. *Procedia Computer Science*, 82, 115-121.
11. Han, J., Rodriguez, J. C., & Beheshti, M. (2008, December). Diabetes data analysis and prediction model discovery using rapidminer. *In 2008 Second International Conference on Future Generation Communication and Networking* (Vol. 3, pp. 96-99). IEEE.
12. Soliman, O. S., & AboElhamd, E. (2014). Classification of diabetes mellitus using modified particle swarm optimization and least squares support vector machine. arXiv preprint arXiv:1405.0549.
13. Sreedevi, E., & Padmavathamma, M. (2016). Design and Development of Hybrid Genetic Classifier Model for Prediction of Diabetes. *International Journal of Modern Trends in Engineering and Research*, 3, 260-265.
14. Priyadarshini, R., Dash, N., & Mishra, R. (2014, February). A Novel approach to predict diabetes mellitus using modified Extreme learning machine. *In 2014 International Conference on Electronics and Communication Systems (ICECS)* (pp. 1-5). IEEE.
15. Mercaldo, F., Nardone, V., & Santone, A. (2017). Diabetes mellitus affected patients classification and diagnosis through machine learning techniques. *Procedia computer science*, 112, 2519-2528.
16. Perveen, S., Shahbaz, M., Guergachi, A., & Keshavjee, K. (2016). Performance analysis of data mining classification techniques to predict diabetes. *Procedia Computer Science*, 82, 115-121.
17. Kannadasan, K., Edla, D. R., & Kuppili, V. (2018). Type 2 diabetes data classification using stacked autoencoders in deep neural networks. *Clinical Epidemiology and Global Health.*
18. Wu, H., Yang, S., Huang, Z., He, J., & Wang, X. (2018). Type 2 diabetes mellitus prediction model based on data mining. *Informatics in Medicine Unlocked*, 10, 100-107.
19. Sisodia, D., & Sisodia, D. S. (2018). Prediction of diabetes using classification algorithms. *Procedia computer science*, 132, 1578-1585.
20. Samant, P., & Agarwal, R. (2018). Machine learning techniques for medical diagnosis of diabetes using iris images. *Computer methods and programs in biomedicine*, 157, 121-128.
21. Yu, W., Liu, T., Valdez, R., Gwinn, M., & Khoury, M. J. (2010). Application of support vector machine modeling for prediction of common diseases: the case of diabetes and pre-diabetes. *BMC medical informatics and decision making*, 10(1), 16.
22. Orabi, K. M., Kamal, Y. M., & Rabah, T. M. (2016, July). Early predictive system for diabetes mellitus disease. *In Industrial Conference on Data Mining (pp. 420-427). Springer, Cham.*
23. Barakat, N., Bradley, A. P., & Barakat, M. N. H. (2010). Intelligible support vector machines for diagnosis of diabetes mellitus. *IEEE transactions on information technology in biomedicine*, 14(4), 1114-1120.
24. Wu, H., Yang, S., Huang, Z., He, J., & Wang, X. (2018). Type 2 diabetes mellitus prediction model based on data mining. Informatics in Medicine Unlocked, 10, 100-107.
25. M. Ashok Kumar., I. Laurence Aroquiaraj., (2019). Adaptive Divergence Weight Firefly Algorithm (ADWFA) with Improved K-Means Algorithm and Adaptive Neuro Fuzzy Inference System (ANFIS) for Type 2 Diabetes Mellitus Prediction. *Jour of Adv Research in Dynamical & Control Systems*, Vol. 11, 06-Special Issue.
26. Alby, S., & Shivakumar, B. L. (2016). A Prediction Model for Type 2 Diabetes Risk Among Indian Women. *ARPN Journal of Engineering and Applied Sciences*, 11(3), 2037-2043.

27. Huang, X., Zeng, X., & Han, R. (2017). Dynamic inertia weight binary bat algorithm with neighborhood search. *Computational intelligence and neuroscience*, 2017.
28. Liang, Y., Balcan, M. F. F., Kanchanapally, V., & Woodruff, D. (2014). Improved distributed principal component analysis. *In Advances in Neural Information Processing Systems* (pp. 3113-3121).
29. Feldman, D., Schmidt, M., & Sohler, C. (2013, January). Turning big data into tiny data: Constant-size coresets for k-means, pca and projective clustering. *In Proceedings of the twenty-fourth annual ACM-SIAM symposium on Discrete algorithms* (pp. 1434-1453). Society for Industrial and Applied Mathematics.
30. Tambade, S., Somvanshi, M., Chavan, P., & Shinde, S. (2017). SVM based Diabetic Classification and Hospital Recommendation. *International Journal of Computer Applications*, 167(1).