EAI Endorsed Transactions
on Scalable Information Systems

Research Article EAI.EU

Risk Assessment of Type 2 Diabetes Mellitus Prediction using an Improved Combination of NELM-PSO

Shiva Shankar Reddy[1,*], Gadiraju Mahesh[1]

[1]Department of Computer Science and Engineering, SRKR Engineering College, Bhimavaram, AndhraPradesh, INDIA.

Abstract

Risk Assessment of Diabetes Type-II is crucial in preventing it and reducing the risk of various comorbidities. There are many existing machine learning models for predicting Type-II diabetics in short term future or in unspecified future. But obtaining a model having optimal performance and predicting diabetes risk in long term future are the main problems. These problems are handled in this work by proposing a stacking based integrated KELM model to predict the risk of diabetes Type-II for a person within five years after assessment. The Pima Indian Diabetic Dataset (PIDD) and a Diabetic Research Center dataset are used in this study. A Min-Max normalization is used to pre-process the noisy datasets. The HAFPSO algorithm used in this work explores the best combination of Base learners by increasing the Classification Accuracy (CA) and decreasing the kernel complexity of the optimal learners. Finally, the model is integrated by utilizing the KELM as a meta-classifier that combines the predictions of the twenty Base Learners. The proposed method is assessed with different measures such as accuracy, sensitivity, specificity, Mathews Correlation Coefficient, and Kappa Statistics. The proposed KELM-HAFPSO approach has got better values of the considered metrics confirming its effectiveness in identifying type-II diabetes. The proposed method helps the clinicians to predict the patients who are at a high risk of Type-II diabetes in the future with the highest accuracy of 98.5%. The results obtained show that the KELM-HAFPSO approach is a promising new tool for identifying type-II diabetes.

Keywords: Kernel Extreme Learning Machine (KELM), Particle Swarm Optimization (PSO), Artificial fish swarm algorithm (AFSO), Meta-classifier, Type-II diabetes Classification, Next benchmark value (NBC)and hybrid PSO-AFSO (HAFPSO)

Received on 18 February 2021, accepted on 28 April 2021, published on 03 May 2021

doi: 10.4108/eai.3-5-2021.169579

1. Introduction

Diabetes is a chronic disease affecting the world's general population. The Diabetic Patient has elongated blood glucose levels, and they are also under risk of developing life-threatening medical conditions in the future. Research [1] has shown that 693 million people will be affected by diabetes by 2045. Diabetes is graded under Type-I, Type-II, Gestational Diabetes, Variation Diabetes, Maturity Onset Diabetes of the Young (MODY), etc. Although a variety of variants have been identified, the most common are Type-I and Type-II. Type-I diabetes occurs when their immune system invades the beta cells that produce insulin. It occurs in one's early stages of life and also affects the later stages. Type-II diabetes normally affects humans who are in their middle age of life and makes their bodies insulin resistant. Insulin Resistance is a condition that doesn't allow the body's cells to absorb the glucose content. People who follow a poor diet regime and are physically inactive [2],[3] tend to experience Type-II diabetes more than the one who follows a healthy diet regime and exercise routine. The complications of Type-II diabetes include renal failure, blindness, bleeding disorder, hypertension, impaired wound healing, heart disease, stroke, and neurodegeneration [4].

[*]Corresponding author. Email: shiva.shankar591@gmail.com

Earlier prediction of people with type-II diabetes is essential for reducing the risk and delaying chronic complications throughout their lifetime. The assessment of risk factors for type-II diabetes assists in the monitoring of undiagnosed patients at risk. The identified factors that are linked to increased risk of Type II diabetes [5] are adults with BMI value ≥25kg/m^2, overweight women planning for pregnancy, individuals with a family history of diabetes, signs of insulin resistance, physical inactivity, etc. Glucose concentration in the blood plays a significant role in predicting diabetes. Diabetes develops when the Fasting Plasma Glucose (FPG) of the individual exceeds an average of 126 mg/dl. The FPG level in the blood is tested after eight to twelve hours of fasting, and the Postprandial Blood Glucose (PPG) level can be tested after a meal. A two-hour plasma glucose level less than 140mg/dl indicates that patients will have greater potential for Type II diabetes [6]. A PPG value (≥160 mg/dl) indicates the possibility of Type II diabetes and also cardiovascular disease. FPG has a high degree of specificity when compared to PPG in predicting Diabetes. Glycosylated Haemoglobin or Haemoglobin A1C (HbA1c) [7] is used to evaluate the sugar level present in the Red Blood Cells. An HbA1c value of≥6.5% indicates Type-II diabetes present in the haemoglobin proteins.

In recent years the possibility of identifying Type-II diabetes in an earlier stage and increasing classification accuracy has been addressed by machine-learning algorithms [8]. A wide range of Machine Learning (ML) algorithms [9], [10] has been proposed for diagnosing Type-II diabetes. Machine Learning leads to predictive modelling, an approach that develops a mathematical model to offer accurate predictions [11]. Due to the vast amount of healthcare data generated daily worldwide, this process can be achievable. The clinical data analysis offers better healthcare solutions to the patients and also aids in financial and operational improvements. Integrating ML algorithms and data systems yields in predicting Type-II diabetes earlier.

M.R. Islam et al [12] observed the fact that Social networks are the repositories of communication between interested people and repository for their communication attributes like images / videos transferred and feeling / sentiments shared. They have exploited the repository for investigating the moods of people and their attitude and thereby for detection of depression problem in people. They have collected facebook data and applied ML techniques including Decision Trees. Finally, after evaluation of the considered algorithms, Decision tree was found to be effective and accurate than the other Machine learning algorithms for the detection of depression in people.

S.S. Reddy et al. in [13] discussed about an important problem of diabetic patients i.e. Diabetic retinopathy. The main thrust of their paper is to develop an effective model for diabetic retinopathy prediction. Different features were considered from the selected UCI data set. Then they have developed different ML models along with Support Vector Machine (SVM) with Gaussian kernel and applied on the data set. The proposed SVM algorithm was evaluated along with the other considered algorithms using novel performance measures and found that their proposed SVM algorithm performed better than other approaches.

R.Sarki et al.[14] emphasized the fact that Diabetes is a major risk for the people and the diabetes patients have a high probability of getting different eye complications. They have highlighted different research related works on detection of diabetes correlated eye ailments. The works they have studied include datasets resources and different detection methods. The detection methods considered for their study include image processing and machine learning. Their work also includes study on suitable performance metrics for the diabetes related eye ailment detection.

The remainder of this paper is structured accordingly. Section 2 has related research work Section 3 introduces the problem statement and new models of the current prediction framework; Section 4 demonstrates the formulation of HAFPSO algorithm; Section 5 describes the KELM methodology proposed; Section 6 explains our study's methodological setup; Section 6 describes the dataset and various important aspects used in this study for experimental evaluation, Section 8 demonstrates the experimental results and discussion and Section 9 concludes this paper.

2. Related work

In recent years, a wide range of clinical trials has been focused on predicting type-II diabetes, in order to evaluate the associated risk factors effectively. R. DelshiHowsalya Devi et al. [15] used the Farthest First Clustering (FFC), SVM, and Sequential Minimal Optimization (SMO) algorithms to present a hybrid approach for Type II diabetes prediction. Bum Ju Lee et al., [16] argued that the hyper triglyceridemic waist (HW) and Waist circumference (WC) is the high associated risk factor for type II diabetes. Triglyceride (TG) is not considered a strong factor when compared to these. Hang Lai et al [17] developed an interactive computer program to help doctors predict the risk of diabetes in their patients and provide preventive measures. They suggested that their proposed model performs well in detecting Type-II Diabetes than the models using Random Forest and Decision Tree. Karim M. Orabi at al. [18] designed a predictive system for Type-II diabetes, which can predict the person’s age in which they are prone to be diabetic by using the regression technique and random code mechanisms. Researchers did not consider the crucial risk factors associated with Type-II diabetes, however, and therefore the model suffered from low precision values. Namrata Singh et al., [19] developed a hybrid model utilizing an ensemble-based approach (XGBoost) to extract the rules from SVM to diagnose hypertension among diabetic patients. However, they did not handle the class imbalance problem present in the dataset.

Researchers have integrated several ML techniques to predict several diseases related to Type-II diabetes with other diseases in healthcare [20]. Xiao-lu XIONG et al. [21] adopted a Cross-sectional Retrospective Study in Chinese Adults by using different ML algorithms (AdaBoost (AD), Multilayer Perceptron (MLP), SVM, Trees Random Forest

(TRF), and Gradient Tree Boosting (GTB**)** to choose the most appropriate technique for the prediction of critical risk factors present in Type-II diabetic patients. This study evaluated eleven risk factors that are closely related to this disease. In literature**,** Bassam Farran et al. [22] combined four different ML techniques (Multifactor Dimensionality Reduction (MDR), k-nearest neighbours (k-NN), Conventional Logistic Regression (LR), and SVM to create predictive models for evaluating the likelihood of hypertension, type-II diabetes, and comorbidity using the Kuwaiti national health dataset. This work builds a predictive model using non-intrusive data to identify diabetic and hypertension patients at high risk. Here the risk factors associated with Type-II diabetes are BMI, ethnicity, and family history of diabetes.

Shiva Shankar et al. in [23] highlighted the importance of diabetes prediction. They have considered the problem of doctors/hospital authorities to predict probability of a diabetic patient's early readmission into hospital with diabetic problems and complications. For attaining this aim, they have considered appropriate ML methods. They have taken dataset of over hundred hospitals obtained from UCI repository for the purpose. Applied the ML methods on the data set considering 46 features. Then evaluated them based on different evaluation metrics. Gradient Boosting method was found to be most effective among all the algorithms considered by them for the prediction.

Reddy Shiva Shankar et al. [24] in their work concentrated on detecting of different kinds of diabetes including type 1 and type 2. They proposed a novel method for the detection using Datamining algorithms. Applied different mining methods including Naïve Bayes (NB) and SVM on medical data of hospitalized patients. An expert voting was also used to select the best algorithm for each record. The scheme proposed in this work was compared with their counterparts using tenfold cross-validation. It was found that the proposed scheme obtained good accuracy in detecting a particular diabetes for the considered dataset.

R Shiva Shankar et al [25] discussed about different side effects of diabetes especially diabetic nephropathy. The main aim of this work is to predict diabetic retinopathy with an effective deep learning method. Implemented different deep learning methods by considering dataset consisting of 84 records. Applied the deep learning methods on the dataset with 21 features and evaluated the results. Then compared and contrasted the algorithms and finally concluded that deep belief network is most effective among algorithms considered for better prediction of diabetic nephropathy.

Bassam Farran et al. [26] proposed a prognostic model to predict future risk for Type II diabetes (within 3,5 and 7 years) using LR, k-NN, and SVM from Kuwait health network data. They demonstrated that this model could identify subjects at higher risk for Type-II diabetes and prognosis at an early stage.We have performed an advanced study A. Sheik Abdullah et al. [27] using an enhanced combination of PSO and Decision Trees to examine the risk factors correlated with Type II diabetes. This study evaluated the risk factors associated with diabetes using a mathematical model named Fishers Linear Discriminant Analysis (FLDA) for the discovered attributes. Han Wu et al. [28] used K-Means and LR algorithms to implement a data mining technique for the prediction of Type 2 diabetes mellitus. This model suffers from high time complexity that occurs in the pre-processing phase. Hamid R. Marateb et al. [29] applied an Expert-Based Fuzzy Micro Albuminuria (EBFMA) Classifier, PSO, and Multiple LR techniques to identify MA in type-II diabetes patients without measuring urinary albumin. The limitations found in this study is its small sample size and cross-sectional nature; however, they did not consider taking a larger sample size for detailed investigations. Table 1 provides a summary of different methods proposed for detecting Type-II diabetes and its associated symptoms.

Reddy Shiva Shankar et al [30] in this paper reiterated the fact that disease prediction in particular diabetes prediction using Datamining and ML is a topic of research interest. They have considered the problem of predicting early readmission of patients suffering with diabetes to the hospitals due to its problems. The problem is handled in their paper with the prediction of the readmission time by the use of Deep belief networks (DBN). Applied different algorithm including DBN on the Pima Indian Diabetes Dataset using R. Then after evaluation and comparison of the considered algorithms found DBN as optimal method for the patient's hospital readmission prediction.

The present study proposed an automatic Type-II diagnosis system utilizing KELM and a hybrid PS0-AFSO optimization algorithm which could reveal the significant factors associated with Type-II diabetes. In the first phase, the input training dataset is divided into five sub-samples by a fivefold cross-validation approach where 90% of the input are used for training and 10% are used to test the model. In the second phase, a hybrid PSO-AFSO (HAFPSO) optimization algorithm is applied for improving the accuracy and base learner optimization. In the final phase, the proposed KELM framework is deployed for automatically selecting the appropriate classifier based on the derived features. The classifier chosen determines if a person is at or not at risk of developing Type II diabetes.The experiments were conducted on Pima Indian Diabetes Dataset (PIDD) and a Diabetes Research Centre dataset.

The table1 illustrated the fact that most of the algorithms have got less than 90% accuracy. But some of the works based on the Particle Swarm Optimization have got more than 90%. This fact is motivation of this work for proposal of a technique based on the PSO that could obtain better results.

The key goals of this research work are summed up as follows:

i. Utilizing two datasets (PIDD and Diabetes Research Centre) to classify important risk factors associated with Type II diabetes.

ii. Formation of theKELM-HAFPSO stacking method to assess Type II diabetes accurately.

iii. The solution for the multi-objective selection problem is analyzed using the HAFPSO algorithm by concurrently optimizing the two benchmark values (Classification Accuracy and Number of Classifiers used).

iv. Nine standard benchmark functions (F1-F9) are used to test the efficiency of the proposed HAFPSO algorithm.

v. The performance of the KELM-HAFPSO classifier has been compared with various state-of-the-art classifiers in terms of accuracy, specificity, sensitivity, Matthew's correlation coefficient, and Kappa statistics.

Table 1. A summary of Type-II diabetes prediction Models used in this literature

Year	Study	Classifiers Used	Findings	Dataset Obtained from	Classification Accuracy
2020	R.D.H Devi et al. (2020)	SVM, FFC, and SMO	Empirically solves two Lagrange multipliers at once	Pima Indian Diabetic Dataset (PIDD)	99.4%
2019	Hang Lai et al. (2019)	LR and Gradient Boosting Machine (GBM)	The results obtained from this model provided high sensitivity and specificity.	Canadian Primary Care Sentinel Surveillance Network (CPCSSN)	83.8%
2019	Namrata Singh et al. (2019)	SVM, XGBoost	Provides economical and beneficial evaluation rules to detect Hypertension.	Pt. JNM, Medical College, Raipur, India	74.7%
2019	Xiao-lu XIONG et al. (2019)	MLP, AD, TRF, SVM, GTB	The prediction and estimation model used can identify effective variables and relationships.	Nanjing Drum Tower Hospital	MLP- 87%; AD-86%; TRF-86%; SVM-86%; GTB-86%
2019	Bassam Farranet al. (2019)	LR, k-NN, SVM	The k-NN algorithm provided a high discriminatory power when compared to LR.	Kuwait Health Network (KHN)	Average Estimate for (3-7 years)-LR:72%; k-NN:81.3%; SVM-70.6%
2019	A. Sheik Abdullah et al. (2019	PSO and Decision Tree (C4.5)	It identifies four important risk factors with a strong correlation accuracy.	Medical Examination Dataset	98.60%
2018	Han Wu et al. (2018)	K-Means and LR	The cluster and class method used increased the prediction accuracy	PIDD, Dr. Schorling, Online Questionnaire (OQ)	PIDD- 95.42% Dr. Schorling-90.7% OQ-93.5%
2015	Bum Ju Lee et al. (2015)	Binary logistic regression (BLR), and Naïve Bayes	The first study to classify phenotypes ' predictive power	Korean Health and Genome Epidemiology Study database	95%
2016	Karim M. Orabi at al. (2016)	Decision Tree	The system randomization and classification results in best accuracy	Egyptian National Research Center	84%
2014	Hamid R. Marateb et al. (2014)	EBFMA classifier, PSO, M- LR	Golden Standard for detecting Microalbuminuria in Type-II diabetic patients.	Isfahan University of Medical Sciences	92%
2013	Bassam Farranet al. (2013)	LR, k-NN, MDR, SVM	In the public health setting, it can be adopted as a primary step to identify the patients and send them to treatment	Real World dataset obtained from the Kuwait Health Network (KHN)	>85%

3. Problem Statement

Type-II diabetes prediction is a bipartite classification problem that splits the patients involved in this study into two categories namely positive class (Diagnosed with type-II diabetes) and negative class (not Diagnosed with Type-II diabetes). This study proposes a KELM based HAFPSO stacking approach to identify whether a subject is likely to develop type-II diabetes within a five-year period based on the several risk factors as predictors. The HAFPSO algorithm is used as a classifier for the two datasets obtained for multi-objective optimization. In the medical industry, Classification Accuracy (CA) is crucial in predicting the disease at an early stage. Four indicators namely True-Positive (Aj1), True-negative (Aj2), False-Positive (Aj3), and False-Negative (Aj4) are used to compute this objective function. The objective function CA is modelled using the four random reference variables Aj1, Aj2, Aj3, and Aj4.

$$A_{j1} = I\{EL_j = PL_j = PC\} \quad (1)$$

$$A_{j2} = I\{EL_j = PL_j = NC\} \quad (2)$$

$$A_{j3} = I\{EL_j \neq PL_j = PC\} \quad (3)$$

$$A_{j4} = I\{EL_j \neq PL_j = NC\} \quad (4)$$

Where*PC* and *NC* denote the actual positive and negative classes respectively. The exact label value which indicates the actual presence of the disease is indicated by EL_j and the predicted label of the proposed model is indicated by PL_j. At any moment, the sum of the four random reference variables is equal to one which is indicated by the following equation.

$$\sum_{k=1}^{4} A_{jk} = 1, \forall_{jj}; \quad TruePositive = \sum_{j=1}^{N} A_{j1} \quad (5)$$

$$TrueNegative = \sum_{j=1}^{N} A_{j2} \quad (6)$$

$$FalsePositive = \sum_{j=1}^{N} A_{j3} \quad (7)$$

$$FalseNegative = \sum_{j=1}^{N} A_{j4} \quad (8)$$

The proposed model uses a two-class classification problem on the PIDD and Diabetic Research Centerdataset and *N* is known to be the total number of samples present in the dataset.The *N* samples are classified into positive (*p*) and negative (*n*) using the following expression.

$$\begin{aligned} p &= TrueNegative + FalsePositive \\ &= \textstyle\sum_{j=1}^{N} A_{j1} + \sum_{j=1}^{N} A_{j4} \end{aligned} \quad (9)$$

$$\begin{aligned} n &= True\ Negative + False\ Positive \\ &= \textstyle\sum_{j=1}^{N} A_{j2} + \sum_{j=1}^{N} A_{j3} \end{aligned} \quad (10)$$

This analysis uses two benchmark values which are specified in the following equations shown below:

$$Increase\ \ B_1 = CA = \frac{\sum_{j=1}^{N} A_{j1} + \sum_{j=1}^{N} A_{j2}}{\sum_{j=1}^{N}\sum_{k=1}^{4} A_{jk}} \quad (11)$$

$$Decrease B_2 = NBC = \frac{N_{CS}}{T_{NC}} \quad (12)$$

Where N_{CS} is the number of classifiers selected by the HAFPSO stacking approach and T_{NC}denotes the number of classifiers present in total. The main aim of this paper is to find the optimum number of base classifiers required to obtain the maximum *CA* value. *NBC* represents the kernel complexity for the optimum learners chosen.

4. Formulation of HAFPSO

Swarm Intelligence is a collection of algorithms that are inspired by nature's way of solving the problems. The emergent complexity of the diabetes prediction model can be solved by using swarm intelligence.

4.1. PSO for Global search

The PSO algorithm [20], [31], [32] is inspired by the social behavior of birds flocking, fish schooling and swarm theory for solving continuous optimization problems. It, in addition, provides fast convergence which leads to an optimal solution. The particles are often anything that will fly through the multi-dimensional search space to seek out an optimal solution for the given problem**.** In the initialization phase, they are allotted a random initial position and velocity. The position of the particle indicates the solution based on the value obtained by the objective function. The particles memorize the position of the best search space they found. Velocity is a weighted sum of three components namely the old velocity, the velocity of the previous best solution, and velocity of the neighbor's best solution.

The PSO swarm consists of a set of particles also known as the initial population where $P = \{p_1, p_2, \ldots\ldots, p_n\}$. The position of the particle which represents the candidate solution is represented by the fitness function *f*. For any time step *t*, p_j it has its associated position ($\vec{a}_j^{\,t}$) and velocity ($\vec{v}_j^{\,t}$) value. The global optimum solution (best solution) with respect to the fitness function $\vec{b}_j^{\,t}$ is estimated as for a given time step*t*. The particle p_j receives its information from its neighbor $N_j \in P$. The PSO algorithm is initialized by generating a random position for the population at the starting point of the Region $R' \subseteq R$. The velocity values are usually initialized R', but sometimes they are also set to zero or minimal values to prevent the particles from exiting the search space in the first iteration. When the algorithm enters its main loop, the velocity and the position of the particles are iteratively updated until convergence. The rules used to update the velocity and position are described below.

$$\vec{v}_j^{t+1} = w\vec{v}_j^t + \alpha_1 \vec{M}_1^t(\vec{b}_j^t - \vec{a}_j^t) + \alpha_2 \vec{M}_2^t(\vec{n}_j^t - \vec{a}_j^t) \tag{13}$$

$$\vec{a}_j^{t+1} = \vec{a}_j^t + \vec{v}_j^{t+1} \tag{14}$$

Where w represents the inertial weight, α_1 *and* α_2 are the acceleration coefficients, $\vec{M}_1$ *and* $\vec{M}_2$ are the two non-diagonal matrices with random numbers as the main diagonal entries. The random number falls under the interval [0, 1] and both matrices are regenerated at the end of the iteration. Vector $\vec{n}_j^t$ indicates the best solution found in the neighborhood by any particle p_j which can be identified as follows:

$$f(\vec{n}_j^t) \le f(\vec{b}_j^t) \qquad \forall p_j \in N_j \tag{15}$$

The values of w, α_1 *and* α_2 should be allocated with appropriate values to prevent the velocity from entering infinity. The value of the acceleration constant is usually the numbers between zeros to four. The PSO algorithm pseudocode is shown in Algorithm 1. The velocity update rule follows three characteristics to generate the local behavior of the particles which is listed as follows:

- **Inertia:** It helps the particle keep track of the previous flight direction and helps to prevent it from rigorously changing direction (velocity).
- **Cognitive Component:** It helps the particle to return to its previous best solution
- **Social Component:** It can access theperformance of the particle compared to its neighbors. It usually represents the group norm or standard that should be attained.

Algorithm 1: Pseudocode for the PSO algorithm

```
Begin
    For each particle present in the swarm
    Randomly initialize the position and velocity
    End For
Iteration t=1
Do
    For each particle present in the swarm
        Evaluate its Fitness-value
            If the Fitness-value is better than the
                Local-Best-Fit
            Then set the current value as Local-Best-
                Position
            End If
    End For
        Choose the particle with the best fitness value of
            all as Global-Best-Position
        For each particle
    Compute the velocity of the particle using
            equation (13)
    Update the position of the particle using
            equation (14)
        End for
    t=t+1
    End Begin
```

4.2. AFSO for local search

The AFSO algorithm mathematically models the collective movement and social behavior of the fish [33].The algorithm is highly convergent, fast, versatile and accurate. The AFSO algorithm imitates the behavior of the fish (preying, swarming, and local search) to reach the global optimum value. The environment where the Artificial fish *(A_f)* lives is the solution space and it also contains the states of every A_f's present. The next behavior of A_f relies on its current state as well as its local environment state.First, the algorithm generates potential solutions randomly and then it performs the search to find the optimal solution. The $A_f$analyzes the environment around it by using its vision. The current state of the $A_f$is represented asC_s, visual distance as V_d, and Visual Position as P_v. The A_f moves to its next state (N_{state}) if it finds the state at V_p better than its C_s. Let $C_s = (c_1, c_2, ..., c_n)$ and $P_v = (p_1^v, p_2^v, ..., p_n^v)$ then the random process is explained as follows:

$$p_i^v = p_i + V_d \bullet rand(), \quad i \in (0,n) \tag{16}$$

$$N_{state} = C_s + \frac{V_p - C_s}{\left\| V_p - C_s \right\|} \bullet Step.rand() \tag{17}$$

Where *rand ()* generates a random number 0 or 1, *n* is the number of variables used and *step* is the step length. The AFSO algorithm includes three behaviors namely *A_f-prey*, *A_f-swarm*, and *A_f-follow*.

a) *A_f-prey:* This function is based on the biological behavior of the $A_f$chasing its prey(food). The state-based on its random visual distance is represented as C_v, and F represents the prey concentration value (objective function/fitness value). When the value of V_d increases the A_f finds its global extreme value rapidly and converges soon.

$$C_v = C_s + V_d \bullet rand() \tag{18}$$

$$C_s^{(t+1)} = C_s^{(t)} + \frac{C_v - C_s^{(t)}}{\left\| C_v - C_s^{(t)} \right\|} \bullet Step.rand() \tag{19}$$

b) *A_f-swarm:*TheA_f always moves in swarms(groups) to exist in its colony and avoid potential threats.

$$C_s^{(t+1)} = C_s^{(t)} + \frac{C_{centre} - C_s^{(t)}}{\left\| C_{centre} - C_s^{(t)} \right\|} \bullet Step.rand() \tag{20}$$

Let C_{centre} be the center position, n_c be the number of companions, χ be the crowd factor, and n be the number of A_f's. If $F_{centre}>F_s$ and $\frac{n_c}{n}<\chi$, this indicates the companion center has a major amount of food (higher fitness value) and less crowd which results in the A_f moves near the companion center.

c) *A_f-follow:*The following behavior represents the swarm moving towards the single fish or multiple fishes finding food. If $F_v>F_s$ and $\frac{n_c}{n}<\chi$, this indicates the companion C_v has a major amount of food (higher fitness value) and less crowd which results in the A_f moves

Here F_v denotes the fitness value of C_v, F_{centre} denotes the fitness value of $C_{centre}$and F_s denotes the fitness value of C_s.The following pseudo-code is provided for the AFSO algorithm.

Algorithm 2: Pseudocode for the AFSO algorithm

```
Begin
Initialize the parameters of the AFSO algorithm such as population size N, Visual Distance, and iteration time t.
Generate N solutions randomly and calculate the fitness function respectively
Iteration t=1
Update the population
        For each individual $C_s$
If $A_f$-prey ($C_s^t$)>$A_f$-follow ($C_s^t$), then perform $A_f$-prey
                Else perform $A_f$-follow
If $A_f$-prey ($C_s^t$)>$A_f$-swarm ($C_s^t$), then perform $A_f$-prey
                Else perform $A_f$-swarm
Compare $C_s$ with the Global Optimal Solution (GOS), if the value of $C_s$ is greater than the GOS, then replace the GOS with $C_s$
Record the value of the GOS in the scoreboard
If the ending condition is met terminate the algorithm
t=t+1
End Begin
```

4.3. HAFPSO optimization

For training the KELM neural network, a HAFPSO optimization algorithm is used and makes full use of both the algorithms. This algorithm combines the behavior of the artificial fish in the swarm and the particle information of the PSO into one [34]. Here, first, the PSO algorithm is applied to the Type-II diabetes prediction problem to initiate the global search.

When the PSO algorithm is terminated, the AFSO algorithm is started; this obtains the final best population from PSO as its initial population and then conducts the local search. The AFSO performs the three functions of the artificial fish to obtain the final fitness value (best solution). The summary of the HAFPSO algorithm is narrated as follows and its flowchart is demonstrated in Figure 1.

Step 1: Initialize the population of PSO and set its iteration value $b=0$

Step 2: Check the convergence condition of PSO: If $b=b_{max-PSO}$, Then goto Step 9 or else go to step 3. Where $b_{max-PSO}$ indicates the maximum iteration number of PSO.

Step 3: Calculate the fitness value of each particle in the population and sort each one of them based on their fitness value.

Step 4: Update the Local-Best -Position, and Global-Best-Position experienced by the j^{th} particle.

Step 5: Update the velocity of each particle using equation (13). This process prevents the particle in the search space from entering in the wrong direction.

Step 6: Update the position of each particle using equation (14). The position indicates the current best solution and fitness value.

Step 7: Repeat step-4 to step-7 until the maximum fitness value ($b_{max-PSO}$) is attained.

Step 8: Set $b=b+1$ and go to step 2.

Step 9: Initialize the parameters of AFSO and select the elite population of the PSO as the initial population of AFSO. The scoreboard is updated with the best particle from the elite population. Initially, the iteration b is set as 0.

Step 10: Check if the convergence condition of AFSO is met or not. If $b=b_{max-AFSO}$ the algorithm converges and the result is updated to the score-board. Here, $b_{max-AFSO}$ is considered as themaximum number of iteration present in AFSO or else go to step 11.

Step 11: *Function selection*- Each function represents the different behaviors (*A_f-prey, A_f-swarm*and *A_f-follow*) exhibited by artificial fish. The artificial fish finds its food (best fitness value) by simulating these three behaviors respectively. If the best fitness value is found select the best behavior to perform if none is available to select the A_f-prey function.

Step 12: *Update the score-board*-Compare the best fitness value between each artificial fish and the score-board. If the fitness value of the artificial fish is found to be superior to the fitness value present in the score-board then update the score-board.

Step 13: Set b=b+1 and go to step 10

5. ELM and KELM

5.1. Extreme Learning Machine (ELM)

ELM is a novel technique for classifying patterns and approximating functions. ELM is a single, feed-forward neural network with one hidden node layer [35]. Weight is assigned randomly between the inputs and hidden nodes and they remained constant throughout the entire training and predicting phases. The weights which directly link the hidden node to the output can be trained quite quickly. ELM improves the prediction accuracy, gives better generalization performance, and reduce the risk of overfitting and lowers computational cost [36]. The ELM model first constructs a classification model from the input dataset D={ $d_l \in R^f$ }, where $l=1,2....,n$ with n samples and f input features. The ELM network consists of i input units, h hidden neurons, and o outputs, where its model output can be formulated as follows:

$$c_i(l)=r_i^{\,T}q(l) \tag{21}$$

Where the weight vector r_i is denoted as $r_i \in R^g, i \in \{1,2,3,........,o\}$, which connects the hidden neuron to the i^{th}output neuron. $q(l) \in R^g, i \in \{1,2,........,o\}$ is a vector representation of output hidden neurons for its corresponding input text $d(l) \in R^f$. Then $q(l)$ is derived as follows:

$$q(l) =[\ f(w_1^T d(l)+b_1),\ f(w_2^T d(l)+b_2)..,\ f(w_g^T d(l)+b_g)] \tag{22}$$

Here $b_k\ (k=1,2,...,g)$ denotes the bias value of the kth hidden neuron, $w_g \in R^f$ which is the weight representation of the g^{th} hidden neuron, and f(.) defines the sigmoid activation function. Both the weight (w_g) and bias vectors (b_k) are generated randomly from a Gaussian distribution. By taking the weight and bias vector as input, the next step generates a matrix that provides the hidden layer output H_O. An account of this, the weight matrix W=[$r_1, r_2,.....,r_o$] is estimated by a Moore–Penrose pseudo inverse approach as follows:

$$W=(H_o \times H_o{}^T)^{-1} H_o \times P^T \tag{23}$$

Where P=[$p(1), p(2),.....,p(n)$] denotes a $o \times n$ matrix, where the actual target vector is the l^{th} column present and $p(l) \in R^o$. After the ELM network's parameters are identified, the class label for the Type-II diabetes prediction is measured as follows:

$$L=\underset{l=1,2,.,0}{\arg\max}\{Out_l\} \tag{24}$$

Here, the predicted class label is represented as L in the above equation.

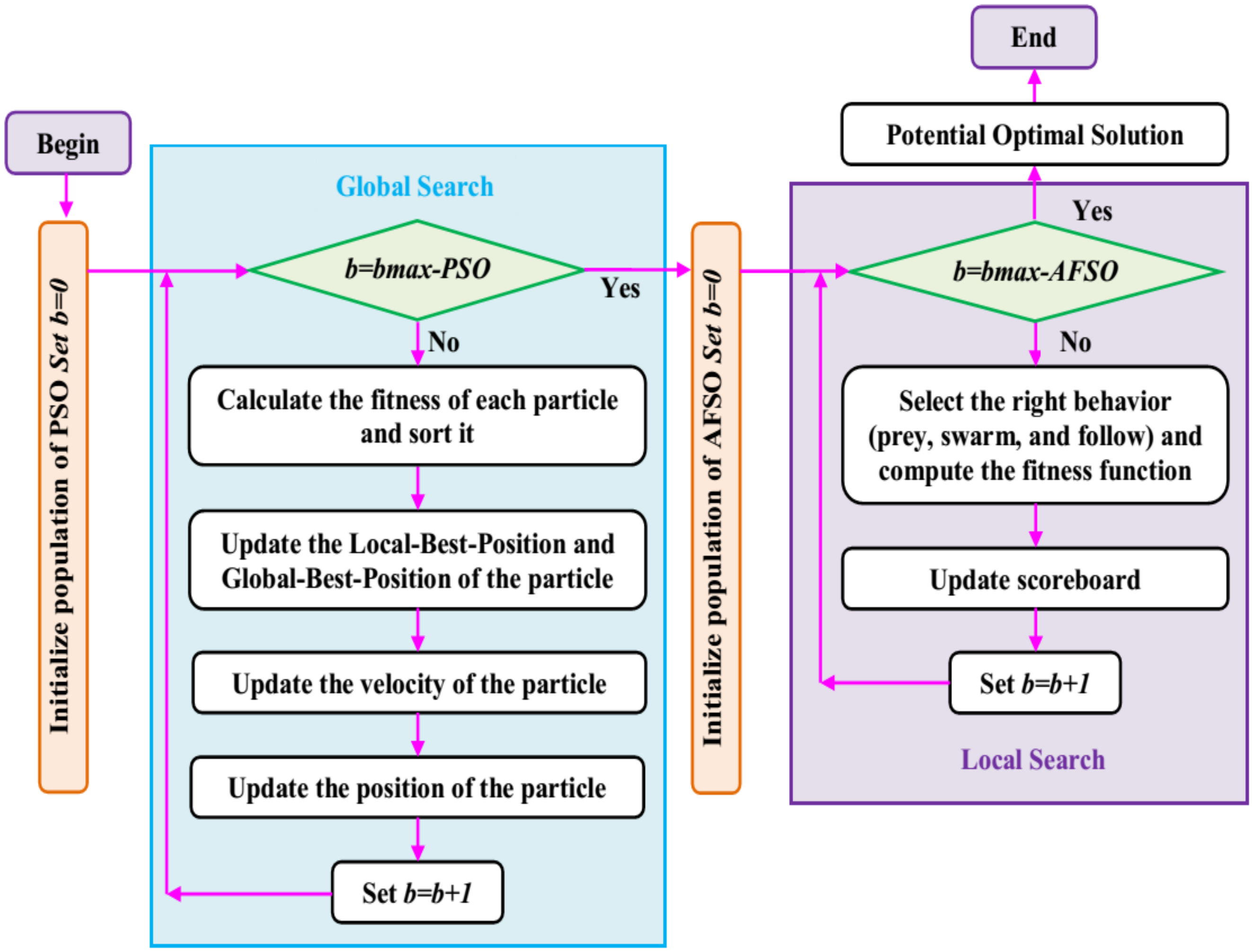

Begin
Initialize population of PSO Set b=0
Global Search
b=bmax-PSO
Yes
No
Calculate the fitness of each particle and sort it
Update the Local-Best-Position and Global-Best-Position of the particle
Update the velocity of the particle
Update the position of the particle
Set b=b+1
Initialize population of AFSO Set b=0
b=bmax-AFSO
Yes
No
Select the right behavior (prey, swarm, and follow) and compute the fitness function
Update scoreboard
Set b=b+1
Local Search
Potential Optimal Solution
End

Figure 1. Flowchart of HAFPSO

5.2. KELM

KELM is a kernel representation of the ELM which can be derived as follows.

For S arbitrary distinct samples

$\{ \{(a_i,t_i)|\ a_i \in R^m, t_i \in R^n, i=1,2,..,S\} \}$, the output function of the ELM with X hidden neurons is derived as follows.

$$f_X(a)=\sum_{i=1}^{X}\beta_i h_i(a) = h(a)\beta \tag{25}$$

The output weight vector that lies between the hidden layer X is denoted as $\beta = \beta_1, \beta_2, \ldots\ldots\ldots, \beta_X$ and the output vector of the same is represented as $h(x) = [\ h_1(a), h_2(a), \ldots\ldots, h_x(a)]$ with respect to the input a, whose sole purpose is to map the data from the input space to the KELM's feature space. When the training error and output weights are reduced it automatically improves the generalization performance of the KELM i.e., derived as follows:

$$Minimize : \|A\beta - O\|, \|\beta\| \tag{26}$$

The least-square solution obtained in equation (26) is based on Karush-Kuhn-Tucker Theory [37] and it can be derived as shown in equation (27).

$$\beta = A^T\left(\frac{1}{C} + AA^T\right)O \tag{27}$$

Here A indicates the hidden layer output matrix, O is the expected sample output matrix and C the regulation coefficient. The ELM learning algorithm's output function is derived as follows.

$$f(a) = h(a)\,A^T\left(\frac{1}{C} + AA^T\right)^{-1} O \tag{28}$$

In the above equation for feature mapping, initially, the value of h(a) remains unknown. In ELM, the unknown value of the kernel matrix can be identified by the Mercers condition (2020) as shown as follows:

$$M = AA^T ; m_{xy} = h(a_x)h(a_y);=k(a_x,a_y) \tag{29}$$

From the above equation, the output of the KELM can be derivedas

$$f(a)=[k(a,a_1),...............,k(a,a_N)]\left(\frac{1}{C}+M\right)^{-1}O \tag{30}$$

Where $M=AA^T$ and *k(a,b)* is the kernel function of the hidden neurons present in the KELM. The feature selection approach utilized by KELM follows a Leave One Out Error (LOOE) scheme [38] for faster convergence and to find the optimal feature in the subsets.

This paper uses four kernel functions as four Base Learners (BL) such as Linear-KELM (Lin-KELM), Polynomial KELM(Pol-KELM), Sigmoid-KELM(Sig-KELM), and Gaussian KELM(Gaus-KELM).

i. *Lin-KELM:*It offers the best performance for larger datasets by generating the best solution for the optimization problem and increasing the predictive performance simultaneously. It generates the results in a lesser amount of time.

$$k(a_i,a_j)=a_i{}^T.a_j \tag{31}$$

ii. *Pol-KELM:*The P-KELM finds the similarities and features from the input text. The exponent value of the polynomial kernel is always greater than one for various cases when its value is lesser than one it is said to be a fractional polynomial. Here *d* indicates the polynomial degree.

$$k(a_i,a_j)=(\gamma a_i{}^T a_j + C)^d \tag{32}$$

iii. *Sig-KELM:*The S-KELM is similar to the sigmoid function used in Logistic Regression.

$$k(a_i,a_j)=\tanh(\gamma a_i{}^T a_j + C) \tag{33}$$

iv. *Gaus-KELM:*In G-KELM**,** the input samples are mapped into the higher dimensional space in a non-linear fashion. There is no prior knowledge used in determining the parameter γ.

$$k(a_i,a_j)=\exp(-\gamma \| a_i - a_j \|^2) \quad \gamma \succ 0 \tag{34}$$

γ and C are the kernel parameters used. C is a constant used to tradeoff higher and lower-order features present in the input dataset. In this paper, the KELM and cross-validation method is combined in the training phase to yield higher prediction accuracy and reduce the overtraining problem. A fivefold cross-validation scheme is constructed initially to find the fitting parameters from the training dataset. The cross-validation based model selection used here automizes the four kernel functions used and reduce the Overfitting problem [39]. The automated process can sometimes degrade the performance of the system when it does not consider the whole process of fitting the model. A HAFPSO optimization algorithm is used to boost system performance.

6. Proposed Approach

This work suggests the KELM HAFPSO stacking method to construct a Type-II diabetes prediction model. The method is clearly identified in subparagraphs below.

6.1. The KELM stacking approach

This segment prescribes the new KELM method for the diagnosis of Type II diabetes from different data samples. The whole PIDD and the physical examination dataset are split into three parts-training, testing and validation. The training data set is used to get the model learners and the prediction error is calculated by the data set used for validation.This study utilizes a learning process with the HAFPSO algorithm for model selection by the twenty BLconstructed and the stack-based integration approach as shown in Figure2. The KELM learning model comprises of two core comprehensive modules. The first module is utilized for base-level learning which constructs the BLfrom the training dataset. The second module is the multi-objective generative module which generates an optimal solution by enhancing base learner count and CA count. The HAFPSO algorithm uses the validation set to perform the model selection procedure. The selected models are integrated via a stacking method. The preceding paragraphs elaborate on the concepts for the two modules used.

6.2. Base-Level Learning Module

This module constructs the base learner from the training dataset. To construct a KELM learning model with higher prediction accuracy four kernel functions are employed as the base learners. The whole PIDD dataset and the physical examination datasetare divided into two parts randomly in a ratio of 90 to 10. Thetraining and validation dataset is constructed from 90% of both datasets obtained. This dataset can be simultaneously used for both the model construction and estimation of the prediction error. The testing set is obtained from the remaining 10% data which is utilized to evaluate the generalization error. The five samples of distinct training datasets obtained is a result of fivefold cross-validation. To this five training dataset created four learners are applied. These four BLare Lin-KELM, Pol-KELM, Sig-KELM, and Gaus-KELM. The validation set estimates the fitness of each solution and identifies the optimal solution. The testing dataset evaluates the performance of the KELM model. The diversity present in the four kernel functions used serves as an essential component for building an efficient KELM model. The trained BLshould be diverse and complementary at the same time to obtain the maximum information from the metadata used for prediction.The crucial factor for the learning aspect at the base level is producing a fair amount of diverse KELM's. The five training samples generated by the fivefold cross-validation gives rise to distinct base learners. Consequently, the diversity is obtained by using the bias value of the four base learners, base learner selection is considered as a significant factor in measuring the proposed model for performance evaluation.

6.3. Multi-Objective Generative Module

The integrated approach of the model shows better productivity than their individual counterparts. Parameter Analysis and model construction is considered as an important aspect of the proposed model. The multi-objective generative module is the HAFPSO stacking approach that incorporates both the selection techniques and the combination of models. The training data is used to provide the candidate solutions, and the validation data is used to determine the candidate solutions ' fitness value for each iteration that is a generative process.

*i. HAFPSO for Parameter Analysis:*This section discusses the function of the HAFPSO to maximize the number of base classifiers for the classification of both datasets and how to achieve a higher CA value. The selection of the model is known as a two-fold issue for maximizing the optimal solution and giving a higher accuracy by using the minimum number of base classifiers. In order to find the right combination of base learners, this optimization process is most necessary. A binary coding scheme is employed to find a solution to the Type-II diabetes problem. The model's selection and rejection are indicated by the value ‘0’ and ‘1’ respectively. To find whether a model is selected or not the bit values ‘0’ and ‘1’ are used. The value ‘0’ indicates that the corresponding model is not selected and the value ‘1’ indicates that the corresponding model is selected. Therefore, the number of BLused matches the length of the solution. After the model selection is completed, the first benchmark value (*CA*) is increased and the next benchmark value(*NBC*) is reduced.

The evolutionary algorithms based on swarm intelligence areadopted to find a solution to multi-objective optimization problems and hence they are also known by the name multi-objective generalization algorithms.By investigating the candidate solution generated it selects the potential optimal solution by performing both global and local search. The HAFPSO algorithm combines the benefits of both algorithms to find the solution which results in improved accuracy, faster convergence, and enhanced global searching ability. The AFSO algorithm is acquainted with the PSO during the iteration process. It not only avoids the pre-mature phase in PSO but also improves the exploring and developing phases in the AFSO algorithm by increasing the diversity of the swarm optimization.This study uses a multi-objective prediction of type-II diabetes problems that must be simultaneously optimized.Since these goals are inherently contradictory, progress towards one goal can only be made to the detriment of at least one other. In order to achieve this goal, we typically aim for the best balance between competing objectives. The HAFPSO algorithm in this analysis is used to optimize the accuracy and number of base classifiers. This process can be explained by the upcoming step based on the concept of influence between two decision vectors present in the objective space. The solution is generated on one condition which the decision vector *v* influences another decision vector *u(*indicated by *v<u)*, if and only if:

$$\begin{aligned} &\forall j \;\; f_j(v) \le f_j(u) \;\; and \;\; \exists_i \; f_i(u) < f_i(u) \\ &when \;\; j,i \in \{1,2,3,..,N\} \end{aligned} \tag{35}$$

The above equation states that the objective of *v* is not aggravated than *u* and it also states that there is one objective present in *v* which is surely greater than the objective present in *u*.Here *N*reflects the cumulativenumber of objectives used consequently $f_j(v)$ specifies the value for the j^{th} objective function based on *u*.

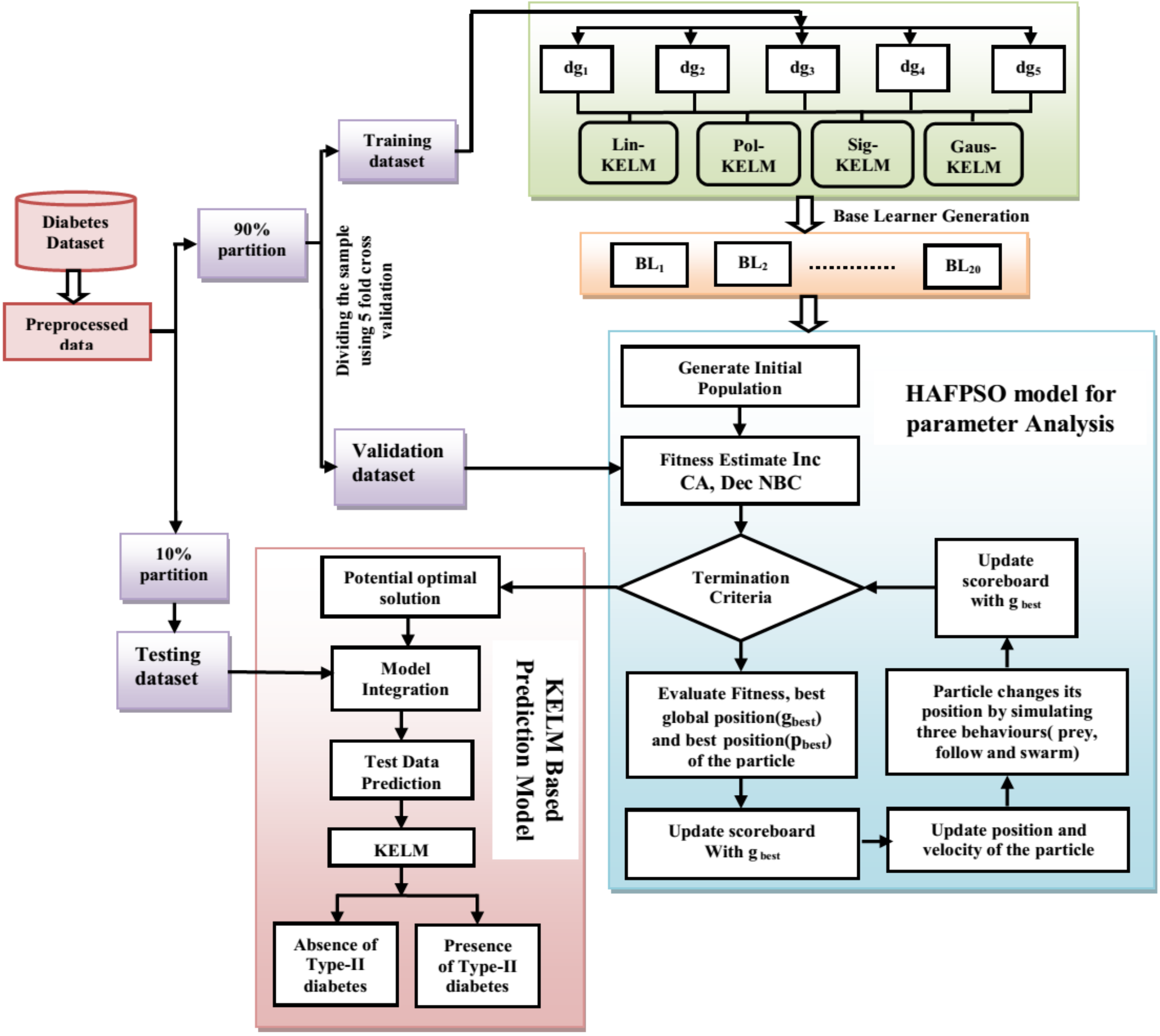

Diabetes Dataset
Preprocessed data
90% partition
10% partition
Training dataset
Dividing the sample using 5 fold cross validation
Validation dataset
Testing dataset
dg1
dg2
dg3
dg4
dg5
Lin-KELM
Pol-KELM
Sig-KELM
Gaus-KELM
Base Learner Generation
BL1
BL2
....
BL20
HAFPSO model for parameter Analysis
Generate Initial Population
Fitness Estimate Inc CA, Dec NBC
Termination Criteria
Update scoreboard with g best
Evaluate Fitness, best global position(gbest) and best position(pbest) of the particle
Particle changes its position by simulating three behaviours(prey, follow and swarm)
Update scoreboard With g best
Update position and velocity of the particle
KELM Based Prediction Model
Potential optimal solution
Model Integration
Test Data Prediction
KELM
Absence of Type-II diabetes
Presence of Type-II diabetes

Figure 2. Working of the proposed KELM-HAFPSO Architecture

*ii. Particle Representation:*In the swarm optimization algorithm a particle and an artificial fish represent a solution found in the search space. The optimal solution is found from the 20 BL's involved by utilizing it through a HAFPSO algorithm.By encoding the object, specifying the fitness function and using the internal velocity and memory to explore the search space, the problem can be formulated.In this study, the particle is encoded in a bit string format for the five training data groups. Each group has four bits associated with it, and each bit represents the classifier used. If the bit value is one, then it represents the use of appropriate kernel function for classification with its corresponding data group.The particle size is equal to the five data group's size created and each particle has four dimensions for its corresponding four classifiers. The whole particle length is twenty bits and it is a binary encoded string. The dimension of the particle reveals its equivalent data group-classifier pair and the bits encoded indicate its existence and nonexistence in the model. Figure 3 demonstrates a random particle with 11 ones and each one indicates its 11 corresponding data group and classifier combination. This process generated eleven BL(i.e., *BL-1,BL-2,BL-4,BL-5,BL-7,BL-9, BL-12,BL-14,BL-15,BL-17,and BL-20)* and gives a base-level model as output. As shown in figure 3, Lin-KELM, Sig-KELM, and Gaus-KELM classifiers with data-groupdg_1 results in the generation of the base-level model as output. Similarly, by training each classifier on certain data-groups base-level models are obtained. Thus the kernels are producedaccording to theBL selected and their combination is shown by highlighting it in blue color in figure 3.

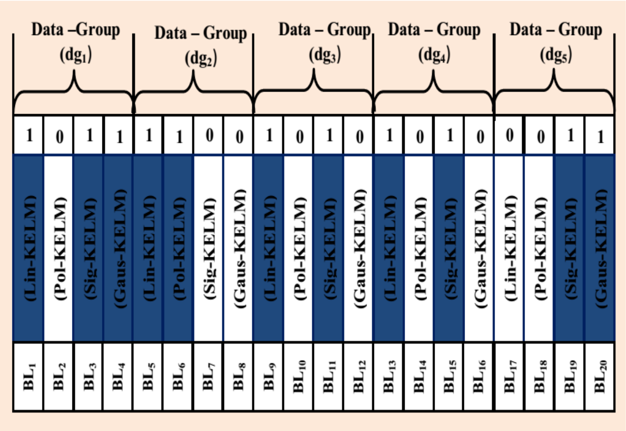


Figure 3. Particle Structure

Since exploration and exploitation are based on two different objectives a single algorithm cannot handle both simultaneously, hence the HAFPSO algorithm is deployed. In the HAFPSO algorithm, the local search phase(exploitation) is done by AFSO and global search space(exploration) is done by PSO. The PSO improves the solution by searching in a small region of the solution until the particle meets the same peak of the objective function. The AFSO algorithm leaves the current peak and searches for the best solution. In the initial phase, the algorithm has large velocities that focus more on exploration and in the later phase when the velocity converges to zero, it goes to exploitation. The balance between the two algorithms leads to a potential optimal solution. Once the HAFPSO algorithm is completed, a potential optimal solution will be obtained. This solution leads to a kernel that indicates the trade-off between the complexity of learning with kernels and prediction accuracy. To build a precise KELM model, the final particle derived is considered as the potential optimal solution with high accuracy. The final kernel was therefore built by the selected learners at the basic level of the encoded particle with a bit of value one.

*iii. Stacking Based Model Integration:*To develop a KELM model, model integration plays a crucial role. The predictive behaviors of the classifierscan be enhanced by using the correct model integration method. For model integration, the proposed hybrid PSO-AFSO uses the stacked generalization approach. The meta-data is combined by using KELM. Experimental analysis was conducted on the KELM model to explore the combination of the selected BLand the KELM. KELM has gained wide popularity in various classification tasks due to the promising results obtained. As the new attribute to the KELM classification function, the predicted output of the BLis used. The meta-data from the training set will be loaded into the KELM meta-learner. The qualified KELM classifier is utilized by the test set to obtain the final predictions. The output of the KELM is the subjects diagnosed with Type-II diabetes; otherwise, the output of the KELM is the subjects not diagnosed with Type-II diabetes.

7. Experimental Evaluation

7.1. Dataset Description and Preprocessing

This Study uses two datasets National Institute of Diabetes and Digestive and Kidney Diseases is where the PIDD dataset originated. The dataset was originally created with an objective to predict whether a patient is subjected to diabetes or not based on the diagnostic measurements obtained from the dataset. The study was conducted on the Pima Indian Women population near Phoenix, Arizona for a period of five years. This database is familiar for researchers to identify the onset of diabetes based on eight features from the 768 samples as shown in Table 2. The eight features tend to be the significant risk factors when predicting Type-II diabetes. From Table 2, the ninth feature describes a class label that identifies whether a patient is subjected to Type-II diabetes or not. Diabetes pedigree function is a likelihood value derived from the patient's family history of diabetes. From the 768 samples obtained, 268 patients were actually diagnosed with diabetes within a one year period. The actually identified 268 samples are annotated with the value one while remaining values are labelled as zero. Since our proposed work focuses on identifying Type-II diabetes, the insulin measures and the number of times pregnant is not considered as a very significant risk factor.

From the 768 samples obtained, 376 samples lacked experimental value because few attributes were considered missing. Due to the errors and deregulation present in the dataset, the missing value occurs. If the missing values are not replaced it leads to inaccuracy in the results. The pgc, tst, dbp, si, and bmi values cannot be termed as zero if it is then the real value is missing. The zero values are interchanged with the mean value of the corresponding attribute present in the training data to replace the missing values. The pre-processed data which is free from errors is taken by the disease prediction model for processing.

In our second dataset, Physical Examination Data were collected from a total of 8700 samples, where 5000 were not affected with Type-II diabetes and 3700 were affected with Type-II diabetes from a Diabetes Research Centre, Tamilnadu. There were a total of 230 indicators present in the physical examination dataset and some of them had no significant relationship with Type-II diabetes. For our study, we selected some indicators manually which are associated with Type-II diabetes of some sort and they are presented in the Table 3 shown below.

In both datasets obtained some values were missing and some of the samples had more than one feature missing. The missing value is interchanged with the mean value here. The mean value of the samples affected by Type-II diabetes and samples not affected with Type-II diabetes is calculated separately. Since each feature has a different interval replacing it with its mean value affects its prediction accuracy. This study uses a Min-Max Normalization to make sure that every feature at least has a value between zero and one.

$$ReplacedValue = \frac{Old - value(feature) - Min(feature)}{Max(feature) - Min(feature)} \quad (36)$$

$$\text{X}$$

$$(new\_Max(feature) - new\_Min(feature)) + new\_\text{Max}\ (feature)$$

Table 2. Features present in PIDD dataset

Sl.No	Attribute information	Attribute Description	Value Range
1	Np	Number of times pregnant	0-17
2	Pgc	Plasma glucose concentration after 2 h in an oral glucose tolerance test (mg/dl)	44-199 mg/dl
3	Dbp	Diastolic blood pressure (mmHg)	24-122 mmHg
4	Tst	Triceps skinfold thickness (mm)	7-63 mm
5	Si	2-h serum insulin (mU/ml)	14-510 mU/Ml
6	bmi	Body mass index (kg/m2)	18.2–59.4 Kg/m^2
7	dpf	Diabetes pedigree function	0.078–1.731
8	age	Age	21-81 years
9	class	Class label for prediction	0- Absence of Type-II diabetes 1- Presence of Type-II diabetes

Table 3. Features present in the Diabetes Research Centre dataset

S.No	Attribute Name	Description	Range
1	Age	Age in years	24-98
2	Sex	Sex of the patient	Male, Female
3	Duration of Diabetes	The time period of the patient affected by Diabetes	1 Month-25 years
4	BMI	Body Mass Index	20.2-40
5	FPG	Diabetes test took before the calorie intake for at least eight hours	100-125 mg/dL
6	PPG	Plasma Glucose Concentration tested after a meal	70-110 mg/dL
7	HbA1c	Test to check the sugar level present in the Red Blood Cells	5.5-15.3%
8	2hPG	A two hour glucose tolerance test	140-200 mg/dL
9	Total Cholesterol (TC)	Measuring the cholesterol levels present in the body	89-277 mg/dL
10	Triglyceride Level (TGL)	Measures the fat level present in the blood	35-500 mg/dL
11	Type-II Diabetes Class	A class label which indicates the presence and absence of the disease	Positive, Negative

7.2. Performance Evaluation Metrics

The efficiency of this proposed workis measured using different evaluation metrics such as Accuracy, Sensitivity, Specificity, and Mathews Correlation Coefficient (MCC) has been used.ROC (Receiver Operating Features) curve provides a graphical interface that offers an estimation of the predictive performance of the model proposed. The curve shows the true positive as well as the false-positive rates. Accuracy indicates the percentage of correctly identified samples from the PIDD dataset. The sensitivity indicates the probability of accuratelyidentified diabetic patients. The specificity indicates the probability of accurately identified non-diabetic patients.

$$Accuracy = \frac{A_{j1} + A_{j2}}{A_{j1} + A_{j2} + A_{j3} + A_{j4}} \quad (37)$$

$$Sensitivity = \frac{A_{j1}}{A_{j1} + A_{j4}} \tag{38}$$

$$Specificity = \frac{A_{j2}}{A_{j2} + A_{j3}} \tag{39}$$

The binary classification in the Type-II diabetes prediction problem can be measured by using MCC. The MCC value lies in a range between -1 to 1. The value -1 indicates that the model's prediction is completely inaccurate in terms of prediction and observation. An accurate prediction value is represented by 1 and 0 indicates that the model's prediction is not better than a random prediction.

$$MCC = \frac{A_{j1\times}A_{j2} - A_{j3} \times A_{j4}}{\sqrt{(A_{j1+}A_{j3})(A_{j1+}A_{j2})(A_{j2+}A_{j3})(A_{j2+}A_{j4})}} \tag{40}$$

The Kappa Statistics (KS) is a critical factor used to check the stability of our proposed approach. It is a comparison technique that compares the result of our proposed model with a result randomly generated by another classifier. The *KS*value ranges between 0 and 1. The range value close to 1 indicates that the model performs well and a range value close to 0 indicates that the model's performance is worse. The *KS*equation is derived as shown below

$$KS = \left[\frac{P(observed) - P(chance)}{1 - P(chance)}\right] \tag{41}$$

$$P(observed) = \left[\frac{A_{j1} + A_{j2}}{TN}\right] \tag{42}$$

$$P(chance) = \left[\frac{(A_{j1} + A_{j2})*(A_{j1} + A_{j3})*(A_{j2} + A_{j4})}{TN^2}\right] \tag{43}$$

Here, *TN* represents the total number of observations found, *P(observed)* indicates the actually observed agreement, *P(chance)* indicates the chance value, A_{j1} indicates the True-positive, A_{j2} indicates the True-negative, A_{j3} indicates the False-Positive, and A_{j4} indicates the False-Negative values.

Table 4. Parameters used in the model selection process

Parameter	Value
C	$\{2^{-5}, 2^{-3,\ldots,}2^{15}\}$
Γ	$\{2^{-15}, 2^{-13,}\ldots, 2^{3}\}$
Hidden Neurons	45
D	20
Population Size	50
Rand()	[0,1]
V_d	3.5
The Try-Number	10
Crowded Factor	χ =1
inertial weight *w* to and	0.7
Acceleration Coefficients	$\alpha_1 = \alpha_2 = 2$

7.3. Parameter Settings

This section gives a brief description ofthe various parameters used in this study.The regularization parameter *C* and the kernel function parameter γ is considered as the most important aspect in KELM and proper care should be given to them while tuning. The values of *C* and γ is taken as $C \in \{2^{-5}, 2^{-3,\ldots,}2^{15}\}$ and $\gamma \in \{2^{-15}, 2^{-13,}\ldots, 2^{3}\}$. This study utilized 150 combinations of both these parameters and the KELM model used 45 hidden neurons and a sigmoid activation function.The dimension size *D* of the problem is taken as 20 and the population size is initialized to 50. The maximum number of iteration *M* is set to 1000 to increase the quality of the potential optimal solution. *Rand()* is a random number value that lies between [0,1], and the step size is set to 0.3. The value of the visual field V_d is set to 3.5, try-number is set to 10, and the crowded factor is χ =1. The values of inertial weight *w* are set to 0.7 and $\alpha_1 = \alpha_2 = 2$ for the PSO algorithm. For the Gaus-KELM the parameter γ is tuned to 0.01, and in the Sig-KELM the parameter γ and *C* is both tuned to 0.5 and 0.01. In Pol-KELM the values for *C*, γ, and *d*are tuned to 0.5, 0.25, and 1 respectively. The parameter used for the model selection process in this work is depicted in Table 4.

7.4. Experimental Setup and Test Functions

The objective of this paper is to improve the classification accuracy of the proposed model. The performance evaluation of this proposed model is compared with various conventionalML techniques that exist in the literature to diagnose Type-II diabetes. The Pseudocode of the evolutionary algorithms is coded in Matlab 2018(a) environment and the experiments were conducted on an Intel Core i9-9980HK Processor, 5.00 GHz maximum turbo frequency, and windows 10-64 bit OS. In this study, nine statistical optimization problems are solved to verify the optimization performance of the HAFPSO algorithm. Nine conventional benchmark functions(F1-F9) are used to verify the effectiveness of the proposed algorithm by comparing it and testing it with PSO [31], AFSO [33], GWO [40], *Non-dominated Sorting Genetic Algorithm-* II (NGSA-II) [41] , and Pareto Archived Evolution Strategy(PAES) [42]. The functions from F1-F3 are termed as single peak function which is used to measure the algorithm's exploitation capacity, functions from F4-F6 are called the multi-peak functions and they are used to evaluate the algorithm's exploration capacity, and the functions F7-F9 are called the fixed multi-peak functions and they are used to evaluate the algorithm's ability to escape from the local minima. These 9 functions have their own expressions and variable ranges as shown in Table 5.

In Table 5, the number of variables used is represented as *v*, the range of values used is represented as *Range*, and the optimal value is indicated as$f_{optimal}$. The effectiveness of the proposed approach can be measured using four evaluation metrics namely Average-Fitness, the Best Value, and Standard Deviation.

Table 6 shows the three-evaluation metrics used to evaluate the proposed methodology with the other five techniques. Here $N_i$indicates the number of iterations used, F_j indicates the fitness value, and μ indicates the mean value of the population.

Table 5. Unimodal Benchmark functions

Benchmark Functions	Mathematical Expression	*v*	Range	$f_{optimal}$
F_1	$F_1(a)=\sum_{j=1}^{m} A_j^{2}$	30	[-100,100]	0
F_2	$F_2(a)=\sum_{j=1}^{m}\lvert A_j\rvert+\prod_{j=1}^{m}\lvert A_j\rvert$	30	[-10,10]	0
F_3	$F_3(a)=\max_j\{\lvert A_j\rvert,1\le j\le m\}$	30	[-100,100]	0
F_4	$F_4(a)=\sum_{j=1}^{m} ja_j^4+random[0,1]$	30	[-1.28,1.28]	0
F_5	$F_5(a)=\frac{1}{4000}\sum_{j=1}^{m}a^2-\prod_{j=1}^{m}\cos\left(\frac{a_j}{\sqrt{j}}\right)+1$	30	[-600,600]	0
F_6	$F_6(a)=\sum_{j=1}^{m}-a_j\sin(\sqrt{\lvert a_j\rvert})$	30	[-500,500]	-2095
F_7	$F_7(a)=\left(\frac{1}{500}+\sum_{i=1}^{25}\frac{1}{i+\sum_{j=1}^{2}(a_i-b_{ij})^6}\right)^{-1}$	2	[-65,65]	1
F_8	$F_8(a)=\sum_{j=1}^{11}\left[x_j-\frac{a_1(y_j^2+y_ja_2)}{y_j^2+y_ja_3+a_4}\right]^2$	4	[-5,5]	0.0029
F_9	$F_9(a)=-\sum_{j=1}^{6}c_j\exp(-\sum_{i=1}^{6}x_{ij}(a_i-p_{ji})^2)$	6	[0,1]	-3.29

Table 6. Evaluation Metric used for Fitness Estimation

Name	Description	Formula
Average-Fitness (AF)	Average value obtained for the best fitness function	$Average-Fitness=\frac{1}{N_i}\sum_{j=1}^{N_i}F_j\ *$
Best-Value (BV)	Minimum value calculated for the overall fitness	$BV=\max_j\le j\le N_i\ F_j\ *$
Standard Deviation (SD)	Measures the reliability of the proposed model	$SD=\sqrt{\frac{1}{N_i-1}\sum_{j=1}^{N_i}(F_j-\mu)^2}$

7.5. Computational Complexity of Kernels

This section discusses the computational complexity of the proposed approach. The worst-case computational complexity encountered in the KELM stacking approach is given as follows:

i. The training complexities of the four classifiers used are listed as follows: Lin-KELM has a complexity of *O(nf)*, and Pol-KELM, Sig-KELM, and Gaus-KELM have a complexity of *O(n^3)*. Where *n* represents the sample size and *f* represents the number of features present in the sample. The computational complexity of iterations 1-6 of the HAFPSO algorithm is *O(TS*(nf+2n^3+nf^2))*. Here *TS* is the number of training samples used.

ii. The computational Complexity of the KELM model for both accuracy and kernel complexity of the multiobjective optimization problem is $O(NS^2)$. Here, N denotes the number of objectives and S denotes the size of the population. This complexity is calculated for iteration number 7 to 36.

iii. The predictional time complexity is represented as $O(t)$ and it is termed as constant. For iteration Number 36-43, the complexity is computed as $O(nMt)$. Here, n is the number of the training samples used and M is the number of models selected.

The computational complexity of the solution proposed is derived from $O(TS*(^{n3}+nf^2+nf+NS^2))$. Complexity depends primarily on three factors: the number of basic learners, the KELM meta-learner and the method of optimization.

Table 7. Comparative Analysis for Optimization Results

Optimization Algorithms		**Benchmark functions**								
		F_1	**F_2**	**F_3**	**F_4**	**F_5**	**F_6**	**F_7**	**F_8**	**F_9**
HAFPSO	**AF**	0	0	0	0.00009	0	-10254.6	0.963	0.00037	-2.967
	BV	0	0	0	0.00009	0	0	0	0.00009	0.0009
	SD	0	0	0	0	0	-10254.6	0.999	0.00021	-2.967
PSO	**AF**	0	0	0	0.00027	0	-23574.6	1.3	0.00102	-4.5021
	BV	0	0	0	0.00019	0	0	0	0.00027	0.0039
	SD	0	0	0	0	0	-23574.6	1.4	0.00102	-4.5021
AFSO	**AF**	2.30E-76	1.39E-47	4.5320	0.00019	0	-7936.44	1.67	0.00065	-2.997
	BV	5.73E-76	2.00E-49	0.2764	0.00063	0	2.74E-39	1	0.00032	0.00027
	SD	1.27E-74	4.33E-48	2.0094	0.00079	0	-10333.4	0.94	0.00027	-2
GWO	**AF**	1.80E+40	0	0	0.00037	0.00278	-609.3	1	0.00046	-2.765
	BV	2.40E+20	0	0	0.00048	0.00398	29.0012	1.704	0.00039	0.100
	SD	2.90E-08	0	0	0	0.00107	-810.3	1.102	0.00027	-2.687
NGSA-II	**AF**	2.36E+29	1.710E-32	0.00374	0.00374	0.00074	-2143.2	0.937	0.00094	-2.9470
	BV	3.11E+72	2.12E-12	0.00598	0.00598	0.00093	5.2784	1.236	0.00004	0.059
	SD	3.79E+19	3.79E+12	0.00762	0.00762	0.00084	-7624.4	0.96	0.00027	-2.984
PAES	**AF**	1.74E+39	0	0.00026	0.00026	0.00240	-549.3	0.987	0.00034	-2.417
	BV	2.27E+10	0	0.00037	0.00037	0.00287	21.0012	1.541	0.00027	0.091
	SD	2.90E-04	0	0	0	0.00112	-810.3	1	0.00012	-2.451

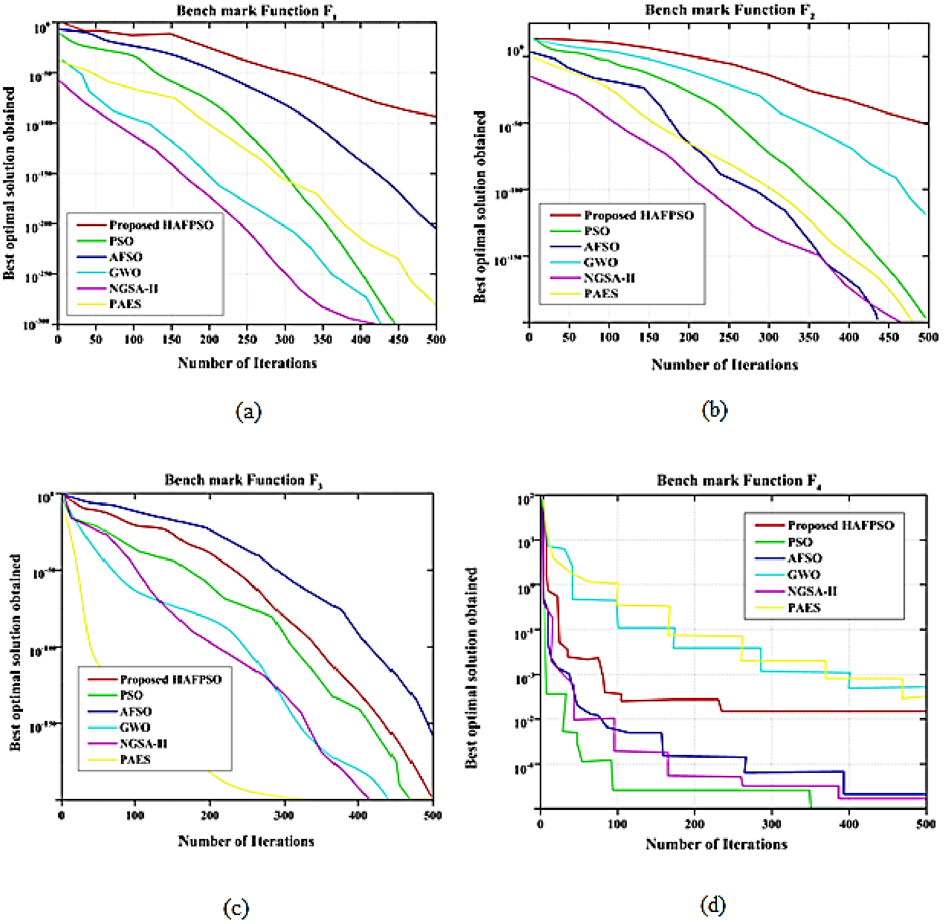
8. Discussion

The purpose of this study is to identify individuals who are at risk for Type-II diabetes and who are potentially vulnerable to this disease. The KELM-HAFPSO model proposed has been assessed for Accuracy, Sensitivity, Specificity, MCC, and KS by means of 5-fold cross-validation on the basis of two related datasets. Comparative Analysis was conducted between the proposed KELM-HAFSO with the other five competitive methods namely ELM-GA [43], Decision Tree C4.5-PSO [27], k-NN [22], MLP [21], LR [17], SVM [38] and NB [16].

8.1. Test functions to evaluate the performance of HAFPSO algorithm

The accuracy of the novel hybrid algorithm has been verified by evaluating them with nine different benchmark functions. Table 7 shows the test results of the six algorithms compared to evaluate their optimization results. The test results shown in Table 7 prove that the proposed hybrid HAFPSO algorithm has the best fitness capability for both the local and global search and it also prevents it from premature convergence. Additionally, based on the convergence accuracy, HAFPSO converges faster and in a smaller number of iterations than other algorithms. In view

of the above study, it is obvious that the proposed HAFPSO algorithm will effectively boost the PSO and AFSO algorithm's convergence speed and accuracy, respectively. In other words, the suggested AFSO algorithm is statistically assumed to outperform the other five standard algorithms in terms of their applicability and practicality.
Figure 4, 5, and 6 shows the different optimization curves obtained for 9 different benchmark functions by HAFPSO, PSO, AFSO, GWO, NGSA-II, and PAES algorithms to compare the convergence rate. Figure 4 consists of 4 representations for 4 benchmark functions(F$_1$-F$_4$) evaluated by the six algorithms respectively. The remaining representations for the other five benchmark functions(F$_5$-F$_9$) is shown in Figure 5 and 6 respectively.



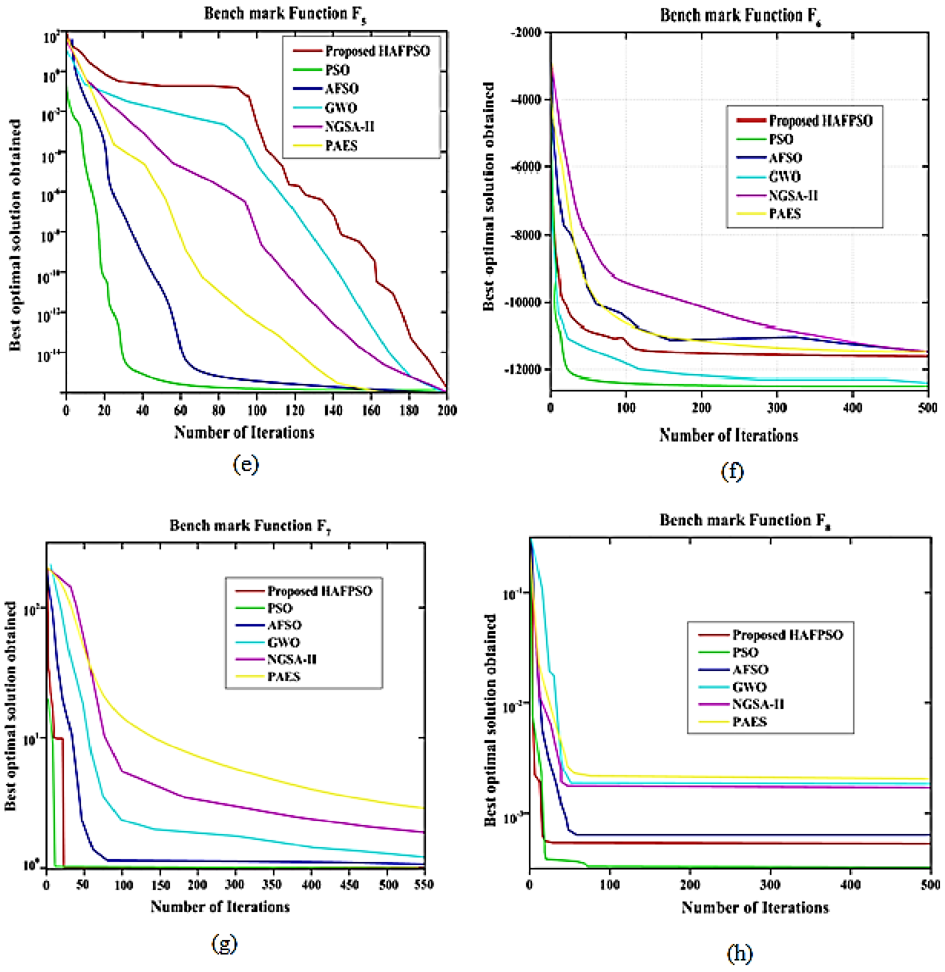
(a)

(b)

(c)

(d)

Figure 4. Comparison of Optimization results with Benchmark Functions. (a)-(d): F1 – F4



(e)

(f)

(g)

(h)

Figure 5. Comparison of Optimization results with Benchmark Functions. (e)-(h): From F5-F8

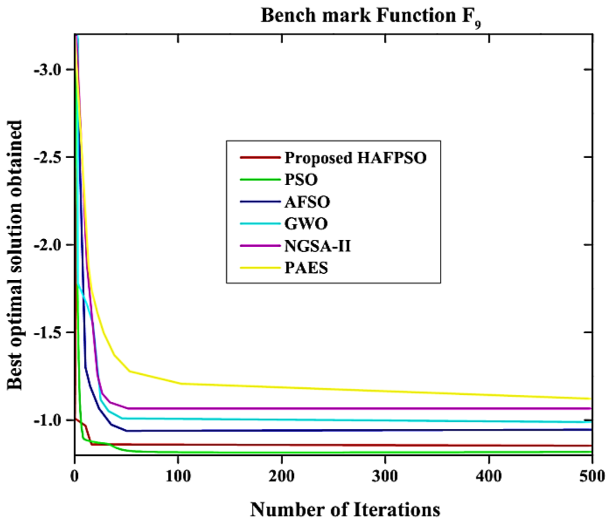


Figure 6. Comparison of Optimization results with Benchmark Functions F9

8.2. Performance and Comparative Analysis of the proposed KELM-HAFPSO Approach

The output of kernel generalization is evaluated by the test data.Here the Dataset-I (PIDD) and Dataset-II(Physical Examination Data) are indicated as D-I and D-II. The potential optimal solution obtained from the two input data sets on a single run as shown in figure 7. The potential optimal solution is obtained from the input dataset on a single run of the HAFPSO algorithm. In figure 7, *B1* represents the classification Accuracy *CA* and *B2* represents the number of base classifiers used *NBC* and it also shows the accuracy and kernel complexity trade-off between the potential optimal solutions. Each kernel is termed as a solution that ensures that the accuracy and kernel complexity is balanced. From Figure 7 it is clear that solution 5 has the best accuracy obtained when the appropriate kernel used. Figure 7 shows the best kernel for validation and test data with 5 BL's, and a B1 value of 0.9992(D-I) and 0.9857(D-II). Table 8 displays the details of both the test accuracy and validation accuracy obtained for the proposed KELM-HAFPSO model in a single run. The table 8 shows that during kernel formation Gaus-KELM and Sig-KELM is the frequently selected base learner to obtain the potential optimal solution and it helps to gain increased prediction accuracy.

The swarm optimization algorithms generate a random combination of BLwhich leads to diverse classification. The detailed classification result of the proposed KELM-HAFPSO classification model is shown in Table 9. For the classification of Type-II Diabetes in D-I, the Accuracy, Sensitivity, Specificity, MCC, and KS values are 0.999, 0.997, 0.869, 0.938, and 0.962 respectively. For instance, in D-II the Accuracy, Sensitivity, Specificity, MCC, and KS values are 0.985, 0.987, 0.832, 0.938, and 0.967. The performance of the proposed approach is superior to the other classifiers compared. The minimum and maximum kernel complexity associated with the potential optimal solution is 4 and 11 out of the twenty rounds of the algorithm. The average selection number of BL is 7 i.e., required to form a kernel.

As shown in Table 10, the proposed KELM-HAFSO model achieves the highest performance among all other competitive classifiers used with an average accuracy of 98.5%, Sensitivity of 98.2%, Specificity of 84.2%, MCC of 90.5%, and Kappa Statistics of 96.5%. In this study, we assessed the other seven competitive models for analysis on the same two datasets through 5-fold Cross-Validation. From Table 10, the accuracy of SVM is much low when compared to other models. Table 10 lists the detailed classification list of ELM-GA, C4.5-PSO, k-NN, MLP, LR, NB, and SVM models. The average classification accuracy of C4.5-PSO and NB is slightly lesser than our proposed model by 0.098 and 0.046. As seen from the Table, the classification accuracy of SVM in D-I and D-II is 78.7% and 77.4% which is very much lower than that of our proposed model by 21.2% and 21.1%. The simulation results of the proposed model obtained from Table 9 and 10proves the efficiency of our proposed approach. The competitive performance of the proposed model over the conventional classifiers shows promising results and superior performance of the model. The proposed KELM-HAFPSO method has therefore been concluded to achieve greater accuracy of predictions than those of the seven competitive classifiers.In contrast ELM-GA, k-NN, MLP, and LR provide slightly less accuracy, sensitivity, specificity, MCC and Kappa Statistics value than the proposed model.

Figure8displays the bar graph of the five performance measures namely accuracy, sensitivity, specificity, MCC, and Kappa Statistics obtained from different classifiers, and the proposed KELM-HAFPSO. The higher accuracy of the proposed KELM-HAFPSO approach is shown in Figure 8for both datasets D-I and D-II when compared with other classifiers.The second highest accuracy value is obtained by C4.5-PSO.Figure 8 demonstrates the significant sensitivity values obtained by our proposed approach. The classifiers such as k-NN, MLP, LR, and SVM yields lower sensitivity value than others. In figure 8, our proposed model achieves higher specificity values when compared to other classifiers. The MCC scores obtained are shown in figure 8, where our proposed approach achieves maximum MCC score(0.935) than other classifiers. In contrast, MLP and SVM achieve the most lowest MCC values. Figure 8 demonstrates the Kappa Statistics comparison of the proposed model with other classifiers. The kappa statistics value of the proposed approach is relatively high when compared to others. The Kappa statistics value of k-NN is slightly low when compared to others. Lastly based on the seven competitive classifiers, the performance of the KELM-HAFPSO on the datasets D-I and D-II tendS to be the highest.

The proposed algorithm is compared with ten classifiers from the literature as shown in Table 11. Table 11 indicates that our proposed model is a promising tool to classify high-risk diabetic patients with higher accuracy of 99.92% and 98.57% for Datasets D-I and D-II respectively.

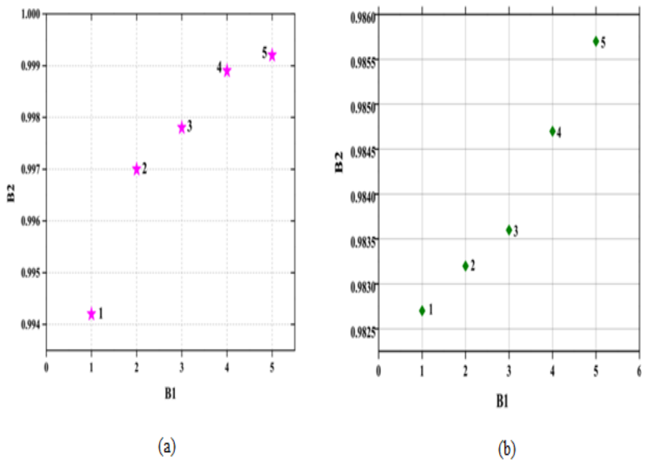


Figure 7. Accuracy (B1) versus kernel complexity (B2) of the potential solution obtained in a single run (a) D-I (b) D-II

Table 8. Validation (B1) and Test Accuracy obtained for the potential optimal solution in a single run for datasets D-I and D-II

Solution Number	Number of BLused (B2)	BLused	D-I		D-II	
			B1	Test Accuracy	B1	Test Accuracy
5	5	Lin-KELM,Gaus-KELM(2),Sig-KELM(2)	0.9992	0.9874	0.9857	0.9667
4	4	Lin-KELM, Gaus-KELM,Poly-KELM, and Sig-KELM	0.9989	0.9863	0.9847	0.9956
3	3	Gaus-KELM(2), Poly-KELM	0.9978	0.9842	0.9836	0.9684
2	2	Lin-KELM, Sig-KELM	0.9970	0.9887	0.9832	0.9674
1	1	Gaus-KELM	0.9942	0.9880	0.9827	0.9645

Table 9. Experimental Results of proposed KELM-HAFPSO with different runs on two different datasets

Number of runs	Kernel Complexity	Dataset-I					Dataset-II				
		Accuracy	Sensitivity	Specificity	MCC	Kappa Statistic	Accuracy	Sensitivity	Specificity	MCC	Kappa Statistic
1	4	0.98	0.98	0.82	0.89	0.94	0.96	0.96	0.82	0.89	0.95
2	5	0.96	0.96	0.82	0.93	0.96	0.96	0.98	0.82	0.89	0.96
3	10	0.96	0.96	0.82	0.93	0.96	0.99	0.98	0.82	0.89	0.96
4	11	0.96	0.98	0.84	0.87	0.98	0.99	0.98	0.84	0.93	0.96
5	6	0.94	0.98	0.84	0.89	0.98	0.96	0.96	0.84	0.93	0.96
6	7	0.98	0.93	0.84	0.89	0.98	0.99	0.94	0.84	0.93	0.96
7	8	0.99	0.96	0.86	0.89	0.96	0.96	0.94	0.86	0.93	0.94
8	5	0.98	0.96	0.82	0.93	0.96	0.94	0.96	0.82	0.93	0.98
9	7	0.96	0.98	0.82	0.94	0.95	0.96	0.98	0.82	0.93	0.98
10	8	0.96	0.96	0.83	0.96	0.94	0.96	0.99	0.83	0.94	0.96
11	8	0.99	0.96	0.78	0.89	0.96	0.99	0.99	0.78	0.93	0.96
12	9	0.98	0.99	0.86	0.89	0.96	0.98	0.98	0.86	0.93	0.98
13	7	0.99	0.99	0.86	0.93	0.94	0.98	0.96	0.86	0.89	0.94
14	8	0.96	0.98	0.93	0.93	0.96	0.98	0.98	0.93	0.93	0.95
15	8	0.96	0.98	0.86	0.93	0.96	0.96	0.98	0.86	0.93	0.95
16	5	0.89	0.98	0.86	0.93	0.96	0.95	0.96	0.86	0.93	0.96

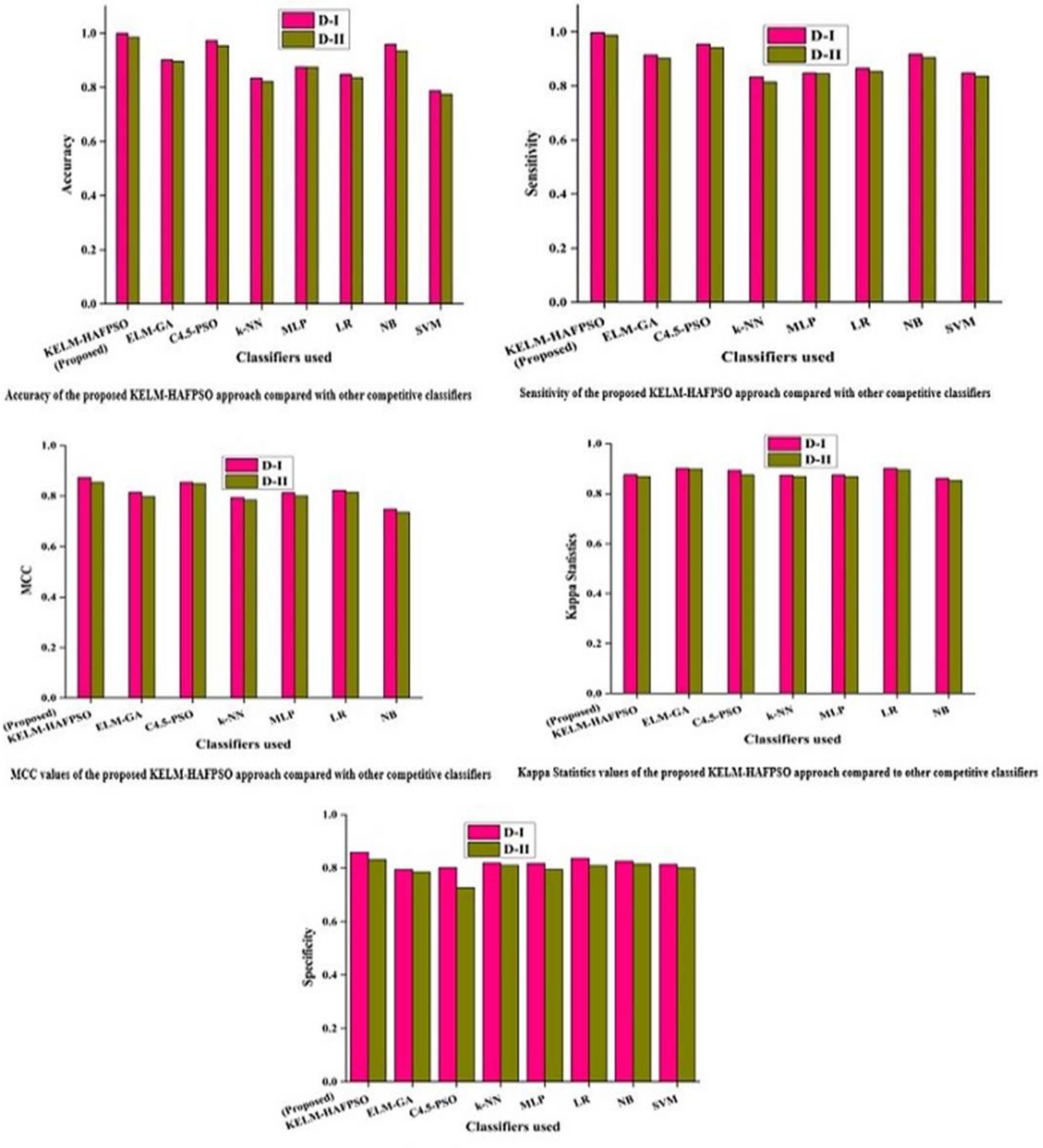
17	4	0.94	0.98	0.82	0.95	0.94	0.94	0.95	0.82	0.94	0.96
18	7	0.94	0.96	0.93	0.89	0.96	0.96	0.98	0.93	0.94	0.98
19	7	0.96	0.96	0.93	0.93	0.96	0.93	0.96	0.93	0.93	0.98
20	6	0.98	0.96	0.87	0.93	0.98	0.98	0.99	0.87	0.89	0.96
Minimum	4	0.89	0.93	0.82	0.87	0.94	0.93	0.95	0.82	0.89	0.94
Median	7	0.98	0.98	0.86	0.93	0.96	0.98	0.98	0.83	0.93	0.96
Maximum	11	0.99	0.99	0.93	0.96	0.98	0.99	0.99	0.93	0.94	0.98
Average	7	0.98	0.98	0.86	0.93	0.96	0.98	0.98	0.86	0.93	0.96

Table 10. Comparative Analysis of proposed Approach with individual Classifiers

Classifiers used	**Accuracy**		**Sensitivity**		**Specificity**		**MCC**		**Kappa Statistics**	
	D-I	**D-II**	**D-I**	**D-II**	**D-I**	**D-II**	**D-I**	**D-II**	**D-I**	**D-II**
Proposed KELM-HAFPSO	0.999	0.985	0.997	0.987	0.869	0.832	0.938	0.938	0.962	0.967
ELM-GA	0.934	0.895	0.934	0.932	0.824	0.785	0.874	0.864	0.876	0.869
C4.5-PSO	0.962	0.953	0.954	0.942	0.822	0.727	0.835	0.828	0.934	0.899
K-NN	0.834	0.821	0.832	0.834	0.820	0.830	0.864	0.849	0.894	0.875
MLP	0.875	0.874	0.847	0.844	0.837	0.825	0.824	0.784	0.874	0.869
LR	0.847	0.835	0.865	0.863	0.837	0.829	0.834	0.821	0.875	0.869
NB	0.958	0.934	0.937	0.934	0.826	0.835	0.823	0.835	0.934	0.895
SVM	0.787	0.774	0.847	0.836	0.834	0.821	0.748	0.735	0.862	0.863

Table 11. Comparative Analysis of Classification Accuracy of the proposed model with classifiers from literature

Sl.No	**Reference Number**	**Classifier used**	**Classifier Accuracy**
1	-	Proposed KELM-HAFPSO (D-I)	99.92%
		Proposed KELM-HAFPSO (D-II)	98.57%
2	[16]	LR+GBM	83.8%
3	[18]	SVM+XGBoost	74.7%
4	[19]	MLP	87%
5	[19]	AD, TRF, SVM, GTB	86%
6	[21]	LR	74%
7	[21]	k-NN	83%
8	[15]	BLR+NB	95%
9	[17]	Decision Tree	84%
10	[27]	EBFMA classifier, PSO, M-LR	92%
11	[35]	SVM, GA	83.80%



Accuracy of the proposed KELM-HAFPSO approach compared with other competitive classifiers

Sensitivity of the proposed KELM-HAFPSO approach compared with other competitive classifiers

MCC values of the proposed KELM-HAFPSO approach compared with other competitive classifiers

Kappa Statistics values of the proposed KELM-HAFPSO approach compared to other competitive classifiers

Specificity of the proposed KELM-HAFPSO approach compared with other competitive classifiers

Figure 8. Performance measures for the proposed KELM-HAFPSO approach compared with other competitive classifiers

8.3. Risk Factor Analysis

For a given training dataset {U, T}, the risk factor identification problem is represented as

$$\min_{\alpha,\gamma} \frac{1}{N}\sum_{i=1}^{N}(T_i - f(\alpha^T a_i ;\gamma))^2 \tag{44}$$

$$\text{Such that } \|\alpha\|_o = p_s \le p, \alpha \in \{0,1\}^f$$

Here *f* is the mapping function of the KELM network, γ is its output weight,$p_s$is the size of the feature subset, and $\|\bullet\|_0$ is a *Lo* normalization function. α_i is a binary value that indicates whether the i^{th} feature in the subset is selected or not. The *Lo* normalization is non-continuous, so there arises a complexity in the optimization function to find the fitness value which can be solved by using a relaxed version of L_1 normalization.

$$\min_{\hat{\alpha},\gamma} \frac{1}{N}\sum_{i=1}^{N}\left(T_i - f(\hat{\alpha}^T a_i;\gamma)\right)^2 + C_1\|\hat{\alpha}\|_1 \tag{45}$$

C_1 is a regularized coefficient and ||.||1 denotes the L1 normalization. The value $\hat{\alpha}$ is not binary and it can take any real number values. If the ith entry $\hat{\alpha}$ is termed to be non-zero then it is immediately selected as a feature. The feature can be extracted via a multivariate LRtechnique [43]. The individual risk factor associated with Type-II diabetes can be identified by substituting the features derived from the KELM algorithm directly into the LR equations. The prediction class is denoted by a variable *P* which takes the following values such as positive-class (value-1), negative–class(value-0) and the predictor variables $a_1, a_2, ..., a_n$ to form a linear function $\alpha_0 + \alpha_1 a_1 + \alpha_2 a_2 + ... + \alpha_n a_n$. In the linear function formed α_0 is considered as the intercept value and $\alpha_1, \alpha_2, ..., \alpha_n$ is the regression coefficient. The probability value of predicting Type-II diabetes lies in positive class value '1', and then the proposed model uses the following equation to gain the probability estimate.

$$p = \frac{e^{(\alpha_0+\alpha_1 a_1+\alpha_2 a_2+......+\alpha_n a_n)}}{1+e^{(\alpha_0+\alpha_1 a_1+\alpha_2 a_2+......+\alpha_n a_n)}} \tag{46}$$

Table 12. Experimental Results for Dataset-1 by fitting it by LR

Name of the feature	Feature Estimate	Standard Error	t-value	P >|t|
Intercept	-9.374	0.835	-12.104	0.000
np	0.109	0.027	3.682	0.000
pgc	0.028	0.002	9.741	0.000
dbp	-0.005	0.004	-0.932	0.392
tst	0.004	0.012	0.399	0.691
si	0.000	0.002	-0.297	0.780
bmi	0.089	0.017	5.014	0.000
dpf	1.099	0.210	3.532	0.000
age	0.017	0.009	1.299	0.179

Table 13. Experimental Results for Dataset-2 by fitting it by LR

Name of the feature	Feature Estimate	Standard Error	t-value	p >|t|
Intercept	-10.784	0.932	-14.237	0.000
Age	0.124	0.037	3.548	0.345
Sex	0.020	0.010	1.384	0.178
Duration of Diabetes	0.000	0.008	-0.310	0.654
BMI	0.148	0.047	3.821	0.000
FPG	0.041	0.002	9.843	0.000
PPG	0.098	0.017	5.014	0.000
HbA1c	0.157	0.053	2.997	0.000
2hPG	0.051	0.006	8.514	0.000
TC	0.101	0.037	6.845	0.000
TGL	0.087	0.035	2.999	0.000

The features present in both the datasets are applied to a logistic regression model and the results obtained are displayed in Table 12 and Table 13. The value *t* indicates the wald statistic which is obtained by taking the ratio of the feature estimate with its corresponding standard error for each feature present in the dataset. The risk factors that are highly associated with Type-II diabetes can be identified by the p>|t| column. In Dataset-I the risk factors are hp, pg, bmi, and dpf and the risk factors in dataset-II are MBI, FPG, PPG, HbA1c,2Hpg,TC, and TGL. From both the tables it is clear that dataset-II has more risk factors that are closely related to Type-II diabetes when compared to dataset-I.

9. Conclusion

This paper aims to develop an automatic Type-II diabetes prediction system based on the novel KELM-HAFPSO approach. Previously, a number of different ML classifiers have been used to develop a Type-II diabetes prediction model. However, due to the complexity of diverse features present in the dataset, it remains difficult to achieve an accurate predictive model that classifies the crucial risk factors associated with this disease. To overcome this obstacle this paper endeavours to devise a multiobjective KELM-HAFPSO approach for selecting the risk factors present in the dataset. Two datasets are used in this paper where one is the PIDD dataset and another one is obtained from the Diabetes Research Centre, Tamilnaduincorporatinga total of 8700 samples. These two datasets were pre-processed using the Min-Max Normalization approach. The accurate classification is yielded in this paper by using a stacking-based kernel approach for the KELM classifier. The KELM uses 4 BLnamely Lin-KELM, Pol-KELM, Sig-KELM, and Gaus-KELM to train the model via fivefold cross-validation to yield 20 trained BL's.The efficiency of this proposed model is evaluated by increasing the classification accuracy and decreasing the number of base classifiers used. The KELM model serves as a meta-classifier that accurately classifies the test samples linked with a higher risk of type-II diabetes based on the risk factors. The comparison results of the novel Hybrid HAFPSO algorithm with six competitive algorithms evaluated using nine different benchmark functions shows the superiority of our proposed algorithm in terms of accuracy and optimalism. The experimental results reveal that the proposed approach outperforms the other seven competitive classifiers in terms of accuracy, sensitivity, specificity, MCC and Kappa Statisticson both the datasets applied. In future side effects of diabetes could be explored.

References

[1] American Diabetes Association. Classification and diagnosis of diabetes: Standards of Medical Care in Diabetes-2020. Diabetes care. 2020 Jan 1; 43(Supplement 1):S14-31.

[2] Liu L. Advanced biostatistics and epidemiology applied in heart failure study. Heart Failure. 2018:83-102.

[3] Abdul-Ghani MA, DeFronzo RA. Plasma glucose concentration and prediction of future risk of type 2 diabetes. Diabetes care. 2009 Nov 1;32(Supplement 2):S194-8.

[4] Bogatyrev SN. Physical activity and type 2 diabetes mellitus risk: population studies review. Diabetes mellitus. 2016 Dec 2;19(6):486-93.

[5] Bonds JA, Hart PC, Minshall RD, Lazarov O, Haus JM, Bonini MG. Type 2 Diabetes Mellitus as a Risk Factor for Alzheimer’s disease. Genes, Environment and Alzheimer's Disease. 2016 Jan 1:387-413.

[6] Cho N, Shaw JE, Karuranga S, Huang Y, da Rocha Fernandes JD, Ohlrogge AW, Malanda B. IDF Diabetes Atlas: Global estimates of diabetes prevalence for 2017 and projections for 2045. Diabetes research and clinical practice. 2018 Apr 1;138:271-81.

[7] World Health Organization. Use of glycated haemoglobin (HbA1c) in diagnosis of diabetes mellitus: abbreviated report of a WHO consultation. World Health Organization; 2011.https://apps.who.int /iris/handle/10665/70523

[8] Nilashi M, bin Ibrahim O, Ahmadi H, Shahmoradi L. An analytical method for diseases prediction using machine learning techniques. Computers & Chemical Engineering. 2017 Nov 2;106:212-23.

[9] Yang S, Wei R, Guo J, Xu L. Semantic inference on clinical documents: combining machine learning algorithms with an inference engine for effective clinical diagnosis and treatment. IEEE Access. 2017 Mar 1;5:3529-46.

[10] Shalev-Shwartz S, Ben-David S. Understanding machine learning: From theory to algorithms. Cambridge university press; 2014 May 19.

[11] Cawley GC, Talbot NL. Preventing Over-Fitting during Model Selection via Bayesian Regularisation of the Hyper-Parameters. Journal of Machine Learning Research. 2007 Apr 1;8(4).

[12] Islam MR, Kabir MA, Ahmed A, Kamal AR, Wang H, Ulhaq A. Depression detection from social network data using machine learning techniques. Health information science and systems. 2018 Dec;6(1):1-2.

[13] Reddy SS, Sethi N, Rajender R. Discovering Optimal Algorithm to Predict Diabetic Retinopathy using Novel Assessment Methods. EAI Endorsed Transactions on Scalable Information Systems. 2021;8(29):e1.

[14] Sarki R, Ahmed K, Wang H, Zhang Y. Automated detection of mild and multi-class diabetic eye diseases using deep learning. Health Information Science and Systems. 2020 Dec;8(1):1-9.

[15] Devi RD, Bai A, Nagarajan N. A novel hybrid approach for diagnosing diabetes mellitus using farthest first and support vector machine algorithms. Obesity Medicine. 2020 Mar 1;17:100152.

[16] Lee BJ, Kim JY. Identification of type 2 diabetes risk factors using phenotypes consisting of anthropometry and triglycerides based on machine learning. IEEE journal of biomedical and health informatics. 2015 Feb 6;20(1):39-46.

[17] Lai H, Huang H, Keshavjee K, Guergachi A, Gao X. Predictive models for diabetes mellitus using machine learning techniques. BMC endocrine disorders. 2019 Dec;19(1):1-9.

[18] Orabi KM, Kamal YM, Rabah TM. Early predictive system for diabetes mellitus disease. In Industrial Conference on Data Mining;2016 Jul 18; Springer, Cham. pp. 420-427.

[19] Singh N, Singh P, Bhagat D. A rule extraction approach from support vector machines for diagnosing hypertension among diabetics. Expert Systems with Applications. 2019 Sep 15;130:188-205.

[20] Poli R, Kennedy J, Blackwell T. Particle swarm optimization. Swarm intelligence. 2007 Jun 1; 1(1):33-57.

[21] Xiong XL, Zhang RX, Bi Y, Zhou WH, Yu Y, Zhu DL. Machine learning models in type 2 diabetes risk prediction: results from a cross-sectional retrospective study in Chinese adults. Current medical science. 2019 Aug;39(4):582-588.

[22] Farran B, AlWotayan R, Alkandari H, Al-Abdulrazzaq D, Channanath A, Thanaraj TA. Use of non-invasive parameters and machine-learning algorithms for predicting future risk of type 2 diabetes: a retrospective cohort study of health data from kuwait. Frontiers in endocrinology. 2019 Sep 11;10:624.

[23] Reddy SS, Sethi N, Rajender R. A Comprehensive Analysis of Machine Learning Techniques for Incessant Prediction of Diabetes Mellitus. International Journal of Grid and Distributed Computing. 2020;13(1):1-22.

[24] Reddy SS, Sethi N, Rajender R. A Review of Data Mining Schemes for Prediction of Diabetes Mellitus

and Correlated Ailments. In2019 5th International Conference On Computing, Communication, Control And Automation (ICCUBEA); 2019 Sep 19; IEEE. pp. 1-5.

[25] Reddy SS, Sethi N, Rajender R. Diabetes Correlated Renal Fault Prediction through Deep Learning. EAI Endorsed Transactions on Pervasive Health and Technology. 2020;6(24):e4.

[26] Farran B, Channanath AM, Behbehani K, Thanaraj TA. Predictive models to assess risk of type 2 diabetes, hypertension and comorbidity: machine-learning algorithms and validation using national health data from Kuwait—a cohort study. BMJ open. 2013 Jan 1;3(5).

[27] Abdullah AS, Selvakumar S. Assessment of the risk factors for type II diabetes using an improved combination of particle swarm optimization and decision trees by evaluation with Fisher's linear discriminant analysis. Soft Computing. 2019 Oct;23(20):9995-10017.

[28] Wu H, Yang S, Huang Z, He J, Wang X. Type 2 diabetes mellitus prediction model based on data mining. Informatics in Medicine Unlocked. 2018 Jan 1;10:100-7.

[29] Marateb HR, Mansourian M, Faghihimani E, Amini M, Farina D. A hybrid intelligent system for diagnosing microalbuminuria in type 2 diabetes patients without having to measure urinary albumin. Computers in biology and medicine. 2014 Feb 1;45:34-42.

[30] Reddy SS, Sethi N, Rajender R. Evaluation of Deep Belief Network to Predict Hospital Readmission of Diabetic Patients. In2020 Second International Conference on Inventive Research in Computing Applications (ICIRCA) 2020 Jul 15; IEEE.pp. 5-9.

[31] Shi Y. Particle swarm optimization: developments, applications and resources. InProceedings of the 2001 congress on evolutionary computation (IEEE Cat. No. 01TH8546);2001 May 27; IEEE. pp. 81-86.

[32] Engelbrecht A. Particle swarm optimization: Velocity initialization. In2012 IEEE congress on evolutionary computation; 2012 Jun 10;IEEE. pp. 1-8.

[33] Neshat M, Sepidnam G, Sargolzaei M, Toosi AN. Artificial fish swarm algorithm: a survey of the state-of-the-art, hybridization, combinatorial and indicative applications. Artificial intelligence review. 2014 Dec 1;42(4):965-97.

[34] Chen H, Wang S, Li J, Li Y. A hybrid of artificial fish swarm algorithm and particle swarm optimization for feed forward neural network training. In International Conference on Intelligent Systems and Knowledge Engineering; 2007 Oct; Atlantis Press 2007.

[35] Huang G, Huang GB, Song S, You K. Trends in extreme learning machines: A review. Neural Networks. 2015 Jan 1;61:32-48.

[36] Liu T, Hu L, Ma C, Wang ZY, Chen HL. A fast approach for detection of erythemato-squamous diseases based on extreme learning machine with maximum relevance minimum redundancy feature selection. International Journal of Systems Science. 2015 Apr 4;46(5):919-31.

[37] Borwein J, Lewis AS. Convex analysis and nonlinear optimization: theory and examples. Springer Science & Business Media; 2010 May 5.

[38] Zhao D, Huang C, Wei Y, Yu F, Wang M, Chen H. An effective computational model for bankruptcy prediction using kernel extreme learning machine approach. Computational Economics. 2017 Feb 1;49(2):325-41.

[39] Hoang ND, Bui DT. Slope stability evaluation using radial basis function neural network, least squares support vector machines, and extreme learning machine. InHandbook of Neural Computation 2017 Jan 1 (pp. 333-344). Academic Press.

[40] Albina K, Lee SG. Hybrid stochastic exploration using grey wolf optimizer and coordinated multi-robot exploration algorithms. IEEE Access. 2019 Jan 31;7:14246-55.

[41] Deb K, Agrawal S, Pratap A, Meyarivan T. A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II. InInternational conference on parallel problem solving from nature; 2000 Sep 18;Springer, Berlin, Heidelberg: 849-858.

[42] Oltean M, Grosan C, Abraham A, Koppen M. Multiobjective optimization using adaptive pareto archived evolution strategy. In5th International Conference on Intelligent Systems Design and Applications (ISDA'05) 2005 Sep 8 (pp. 558-563). IEEE.

[43] Alharbi A, Alghahtani M. Using genetic algorithm and ELM neural networks for feature extraction and classification of type 2-diabetes mellitus. Applied Artificial Intelligence. 2019 Mar 21;33(4):311-28.