



Hybrid artificial fish particle swarm optimizer and kernel extreme learning machine for type-II diabetes predictive model

N. Kanimozhi¹ · G. Singaravel²

Received: 15 June 2020 / Accepted: 3 February 2021 / Published online: 18 March 2021
© International Federation for Medical and Biological Engineering 2021

Abstract

The World Health Organization (WHO) estimated that in 2016, 1.6 million deaths caused were due to diabetes. Precise and on-time diagnosis of type-II diabetes is crucial to reduce the risk of various diseases such as heart disease, stroke, kidney disease, diabetic retinopathy, diabetic neuropathy, and macrovascular problems. The non-invasive methods like machine learning are reliable and efficient in classifying the people subjected to type-II diabetics risk and healthy people into two different categories. This present study aims to develop a stacking-based integrated kernel extreme learning machine (KELM) model for identifying the risk of type-II diabetic patients based on the follow-up time on the diabetes research center dataset. The Pima Indian Diabetic Dataset (PIDD) and a Diabetic Research Center dataset are used in this study. A min-max normalization is used to preprocess the noisy datasets. The Hybrid Particle Swarm Optimization-Artificial Fish Swarm Optimization (HAFPSO) algorithm used satisfies the multi-objective problem by increasing the Classification Accuracy (CA) and decreasing the kernel complexity of the optimal learners (NBC) selected. At last, the model is integrated by utilizing the KELM as a meta-classifier which combines the predictions of the twenty Base Learners as a whole. The proposed classification 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 proposed method is tested with different measures such as accuracy, sensitivity, specificity, Mathews Correlation Coefficient, and Kappa Statistics are calculated. The results obtained show that the KELM-HAFPSO approach is a promising new tool for identifying type-II diabetes.

Keywords Kernel extreme learning machine; · Particle swarm optimization; · Artificial fish swarm algorithm; · Meta-classifier; · Type-II diabetes classification

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 at 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 2 diabetes is also known as non-insulin-dependent diabetes mellitus (NIDDM) or adult-onset diabetes. 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 does not allow the body's cells to absorb the glucose content. The reasons for type-II diabetes is mainly due to overweight, lack of exercise, and other genetic factors. 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

✉ N. Kanimozhi
kanimozhi.na@gmail.com; kanimozhiraja3108@gmail.com

¹ Department of Computer Science and Engineering, GKM College of Engineering and Technology, Chennai, India

² Department of Information Technology, K S Rangasamy College of Engineering, Tiruchengode, India

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]. 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 $\geq 25 \text{ kg/m}^2$, 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 12 h of fasting, and the postprandial blood glucose (PPG) level can be tested after a meal. A 2-h 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 hemoglobin or hemoglobin A1c (HbA1c) [7] is used to evaluate the sugar level present in the red blood cells. An HbA1c value of $\geq 6.5\%$ indicates type-II diabetes present in the hemoglobin 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 [8–11]. A wide range of Machine Learning (ML) algorithms [12, 13] has been proposed for diagnosing type-II diabetes. Machine Learning leads to predictive modeling, an approach that develops a mathematical model to offer accurate predictions [14]. 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.

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. Delshi Howsalya Devi et al. [15] used the Farthest First Clustering (FFC), Support Vector Machine (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 hypertriglyceridemic 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 et 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 neighbors (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.

Bassam Farran et al. [23] 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. [24]) 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. [25] 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. [26] 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 are its small sample size and cross-sectional nature; however, they did not consider taking a larger sample size for detailed investigations. In recent years, evolutionary methods [27–31] have been achieved better performance in engineering and medical applications.

A novel approach based on transfer learning has been deployed for human gene regulatory network reconstruction (Paolo Mignone et al. [32]). The Gene Regulatory Network (GRN) is mainly reconstructed using the Gene Expression data to identify the regulatory schemes utilized in identifying different human diseases. The main problem considered here is to overcome the low availability of the labeled samples and make full use of the unlabelled samples. Emanuele Pio Barracchia et al. [33] presented a link prediction approach using hierarchical clustering to discover the ncRNA relationship in a heterogeneous network comprising different biological attributes. This approach offers increased prediction accuracy along with predictions at a different level of granularity. The classification of this methodology falls into two categories namely algorithm-based and similarity-based techniques. Table 1 provides a summary of different methods proposed for detecting type-II diabetes and its associated symptoms.

The present study proposed an automatic type-II diagnosis system utilizing KELM and a hybrid PSO-AFSO optimization algorithm which could improve the classification accuracy of Type-II diabetes prediction. 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 is used for training and 10% are used to test the model. In the second phase, a Hybrid Particle Swarm Optimization-Artificial Fish Swarm Optimization (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 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 the KELM-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, Matthews correlation coefficient, and Kappa statistics.

The remainder of this paper is structured accordingly. Section 2 introduces the problem statement and new models of the current prediction framework; Section 3 demonstrates the formulation of the HAFPSO algorithm; Section 4 describes the KELM methodology proposed; Section 5 explains our study's methodological setup; Section 6 describes the dataset and various important aspects used in this study for experimental evaluation; Section 7 demonstrates the experimental results and discussion; and Section 8 concludes this paper.

2 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 5 years 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 (A_{j1}), True-negative (A_{j2}), False-Positive (A_{j3}), and False-Negative (A_{j4}) are used to compute this objective function. The objective function CA is modeled using the four random reference variables $A_{j1} = I \{EL_j = PL_j = PC\}$, $A_{j2} = I \{EL_j = PL_j = NC\}$, $A_{j3} = I \{EL_j \neq PL_j = PC\}$, and $A_{j4} = I \{EL_j \neq PL_j = NC\}$. Here 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 as

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. [15]	SVM, FFC, and SMO	Empirically solves two Lagrange multipliers at once	Pima Indian Diabetic Dataset(PIDD)	99.4%
2020	Namrata Singh and Pradeep Singh [20]	stacking-based evolutionary ensemble learning	Diabetes mellitus (DM)	University of California repository	83.8%
2019	Hang Lai et al. [17]	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. [19]	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. [21]	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%
2018	Bassam Farran et al. [23]	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%
2018	A. Sheik Abdullah et al. [24]	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. [25]	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%
2016	Bum Ju Lee et al. [16]	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 et al.[18]	Decision Tree	The system randomization and classification results in the best accuracy	Egyptian National Research Center	84%
2014	Hamid R. Marateb et al. [26]	EBFMA classifier, PSO, M- LR	Golden Standard for detecting Microalbuminuria in Type-II diabetic patients.	Isfahan University of Medical Sciences	92%
2013	Bassam Farran et al. [22]	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%

$$\sum_{k=1}^4 A_{jk} = 1, \forall j; \text{ TruePositive} = \sum_{j=1}^N A_{j1}. \text{ Where, TrueNegative} = \sum_{j=1}^N A_{j2}, \text{ FalsePositive} = \sum_{j=1}^N A_{j3}, \text{ and FalseNegative} = \sum_{j=1}^N A_{j4}.$$

The proposed model uses a two-class classification problem on the PIDD and Diabetic Research Center dataset 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 expressions: $p = TP + FN = \sum_{j=1}^N A_{j1} + \sum_{j=1}^N A_{j4}$ and $n = TN + FP = \sum_{j=1}^N A_{j2} + \sum_{j=1}^N A_{j3}$.

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

$$\text{Increase } B_1 = CA = \frac{\sum_{j=1}^N A_{j1} + \sum_{j=1}^N A_{j2}}{\sum_{j=1}^N \sum_{k=1}^4 A_{jk}} \tag{1}$$

$$\text{Decrease } B_2 = NBC = \frac{N_{CS}}{T_{NC}} \tag{2}$$

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 maximize the CA value by identifying the optimum number of base classifiers required. NBC represents the kernel complexity for the optimum learners chosen. The optimum learners are mainly selected based on the opinion that different learners introduce diverse classification issues and to enhance the classification accuracy, these learners can be merged.

3 Formulation of HAFPSO

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

i. PSO for global search

The PSO algorithm [34–36] 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 swarm optimization algorithm, a particle and an artificial fish represent a solution found in the search space. 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 the velocity of the neighbor’s best solution.

The PSO swarm consists of a set of particles also known as the initial population where $P = \{p_1, p_2, \dots, p_n\}$. 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.

$$\vec{v}_j^{t+1} = w \vec{v}_j^t + \alpha_1 \vec{M}_1 \left(\vec{b}_j^t - \vec{a}_j^t \right) + \alpha_2 \vec{M}_2 \left(\vec{n}_j^t - \vec{a}_j^t \right) \tag{3}$$

$$\vec{a}_j^{t+1} = \vec{a}_j^t + \vec{v}_j^{t+1} \tag{4}$$

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:

$$f \left(\vec{n}_j^t \right) \leq f \left(\vec{b}_j^t \right) \quad \forall p_j \in N_j \tag{5}$$

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 (3)
    Update the position of the particle using equation (4)
  End for
 $t=t+1$ 
End Begin

```

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 the performance of the particle compared to its neighbors. It usually represents the group norm or standard that should be attained.

ii. AFSO for local search

The AFSO algorithm mathematically models the collective movement and social behavior of the fish [37].

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 as C_s , visual distance as V_d , and Visual Position as V_p . The A_f moves to its next state (N_{state}) if it finds the state at V_p better than its C_s . Let $C_s = (c_1, c_2, \dots, c_n)$ and $P_v = (p_1^v, p_2^v, \dots, p_n^v)$ then the random process is explained as follows:

$$p_i^v = p_i + V_d \cdot rand(), \quad i \in (0, n) \quad (6)$$

$$N_{state} = C_s + \frac{V_p - C_s}{\|V_p - C_s\|} \cdot Step.rand() \quad (7)$$

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 Vd increases the A_f finds its global extreme value rapidly and converges soon.

$$C_v = C_s + V_d \cdot rand() \tag{8}$$

$$C_s^{(t+1)} = C_s^{(t)} + \frac{C_v - C_s^{(t)}}{\|C_v - C_s^{(t)}\|} \cdot Step \cdot rand() \tag{9}$$

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

$$C_s^{(t+1)} = C_s^{(t)} + \frac{C_{centre} - C_s^{(t)}}{\|C_{centre} - C_s^{(t)}\|} \cdot Step \cdot rand() \tag{10}$$

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.

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

- c) *Af*-follow: The following behavior represents the swarm moving towards the single fish or multiple fishes finding food. If $F_v > F_s$ and $\frac{F_v}{n} < \chi$, this indicates the companion C_v has a major amount of food (higher fitness value) and less crowd which results in the *Af* 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 . AFSO algorithm is presented in algorithm 2.

iii. 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 [38]. 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 Fig. 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 (3). 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 (4). 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

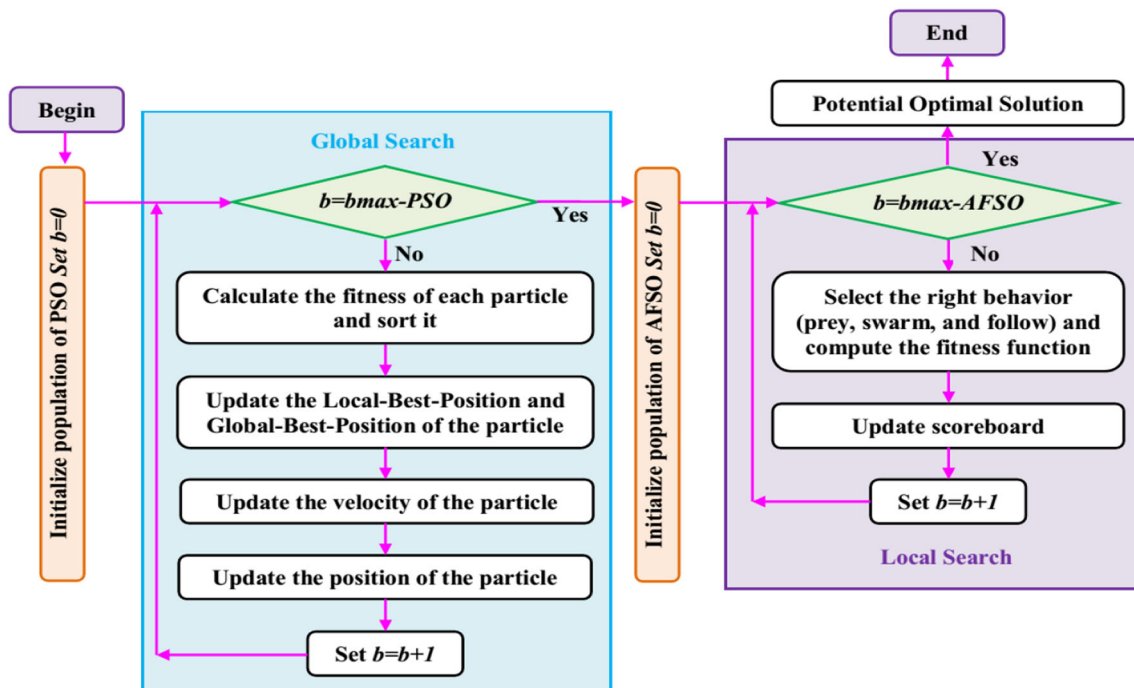


Fig. 1 Flowchart of HAFPSO

the maximum 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

4 ELM and KELM

4.1 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 [39]. 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 reduces the risk of overfitting, and lowers computational cost [40]. The ELM model first constructs a classification model from the input dataset $D = \{d_i \in R^f\}$, where $i = 1, 2, \dots, n$ 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:

$$c_i(l) = r_i^T q(l) \tag{11}$$

where the weight vector r_i is denoted as $r_i \in R^g, i \in \{1, 2, 3, \dots, o\}$, which connects the hidden neuron to the i^{th} output neuron. $q(l) \in R^g, i \in \{1, 2, \dots, o\}$ 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:

$$q(l) = [f(w_1^T d(l) + b_1), f(w_2^T d(l) + b_2), \dots, f(w_g^T d(l) + b_g)] \tag{12}$$

Here b_k ($k = 1, 2, \dots, g$) denotes the bias value of the k^{th} hidden neuron, $w_g \in R^f$ which is the weight representation of the g^{th} hidden neuron, and $f(\cdot)$ 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 $W = [r_1, r_2, \dots, r_o]$ is estimated by a Moore–Penrose pseudo inverse approach as follows:

$$W = (H_o \times H_o^T)^{-1} H_o \times P^T \tag{13}$$

where $P = [p(1), p(2), \dots, p(n)]$ 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:

$$L = \operatorname{argmax} \{Out_l^{l=1,2,\dots,o}\} \tag{14}$$

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

4.2 KELM

KELM is a kernel representation of the ELM which can be derived as follows. For S arbitrary distinct samples $\{(a_i, t_i) | a_i \in R^m, t_i \in R^n, i = 1, 2, \dots, S\}$, the output function of the ELM with X hidden neurons is derived as follows.

$$f_X(a) = \sum_{i=1}^X \beta_i h_i(a) = h(a) \beta \tag{15}$$

The output weight vector that lies between the hidden layer X is denoted as $\beta = \beta_1, \beta_2, \dots, \beta_X$ and the output vector of the same is represented as $h(x) = [h_1(a), h_2(a), \dots, h_x(a)]$ 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:

$$\text{Minimize} : \|A\beta - O\|, \|\beta\| \tag{16}$$

The least-square solution obtained in equation (16) is based on Karush-Kuhn-Tucker Theory [41] and it can be derived as shown in equation (17).

$$\beta = A^T \left(\frac{1}{C} + AA^T \right) O \tag{17}$$

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.

$$f(a) = h(a) A^T \left(\frac{1}{C} + AA^T \right)^{-1} O \tag{18}$$

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 [42] as shown as follows:

$$M = AA^T; m_{xy} = h(a_x)h(a_y); = k(a_x, a_y) \quad (19)$$

From the above equation, the output of the KELM can be derived as

$$f(a) = [k(a, a_1), \dots, k(a, a_N)] \left(\frac{1}{C} + M \right)^{-1} O \quad (20)$$

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 [43] 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.

$$k(a_i, a_j) = a_i^T \cdot a_j \quad (21)$$

- 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.

$$k(a_i, a_j) = (\gamma a_i^T a_j + C)^d \quad (22)$$

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

$$k(a_i, a_j) = \tanh(\gamma a_i^T a_j + C) \quad (23)$$

- 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 .

$$k(a_i, a_j) = \exp(-\gamma \|a_i - a_j\|^2) \quad \gamma > 0 \quad (24)$$

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 [20]. 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.

5 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.

5.1 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 BL constructed and the stack-based integration approach as shown in Fig. 2. The KELM learning model comprises of two core comprehensive modules. The first module is utilized for base-level learning which constructs the BL from 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.

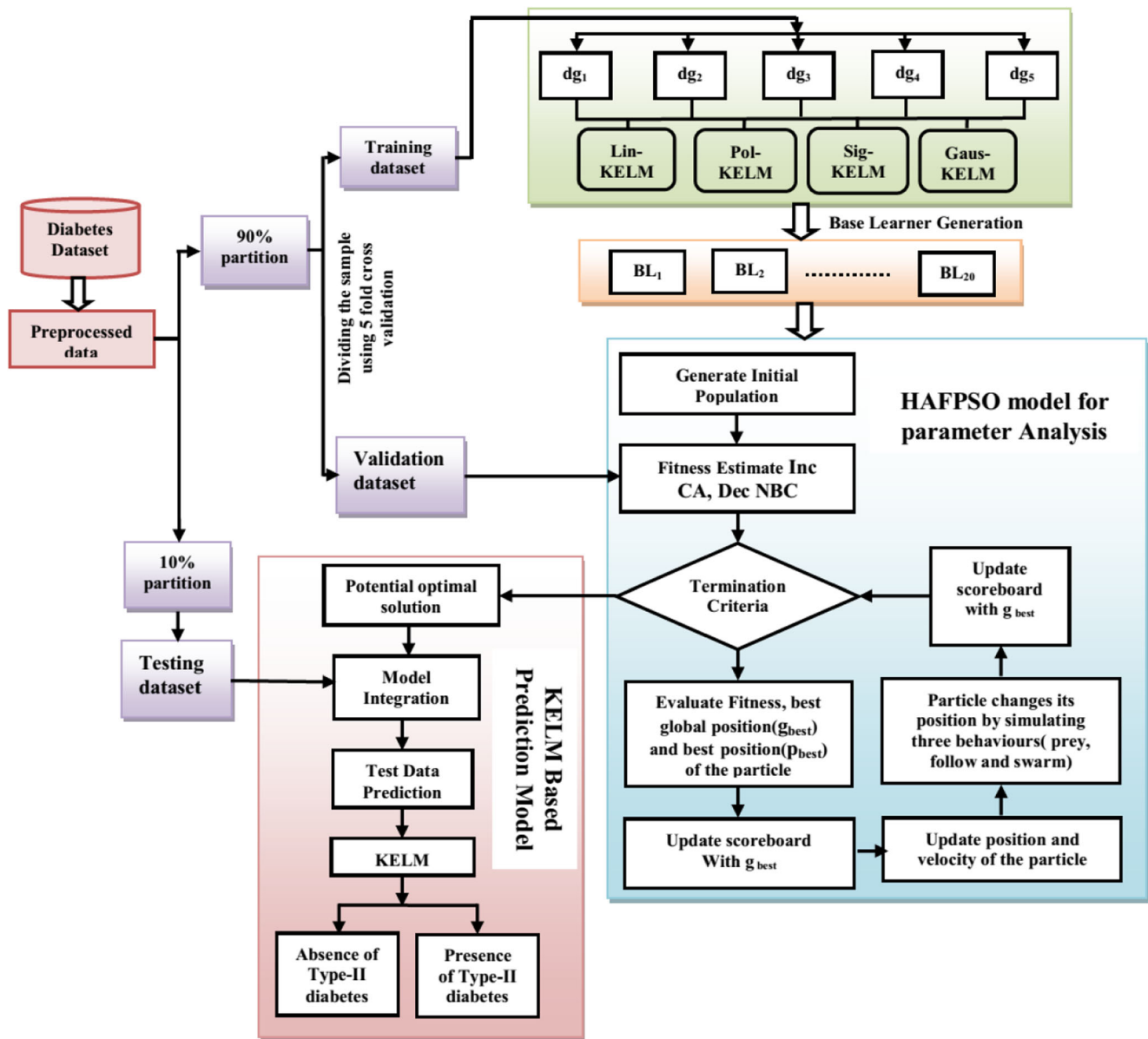


Fig. 2 Working of the proposed KELM-HAFPSO architecture

5.2 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 dataset are divided into two parts randomly in a ratio of 90 to 10. The training 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 are a result of fivefold cross-validation. To

this five training dataset created four learners are applied. These four BL are 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 BL should 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.

5.3 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 twofold 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 BL used 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 are adopted 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 AFPSO 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 AFPSO 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:

$$\forall j f_j(v) \leq f_j(u) \text{ and } \exists_i f_i(v) < f_i(u) \text{ when } j, i \in \{1, 2, 3, \dots, N\} \quad (25)$$

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 cumulative number of objectives used consequently $f_j(v)$ specifies the value for the j^{th} objective function based on u .

- ii. Particle representation: 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 the appropriate kernel function for classification with its corresponding data group. The particle size is equal to the five data group's sizes 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 Fig. 3, Lin-KELM, Sig-KELM, and Gaus-KELM classifiers with data-group dg_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 produced according to the BL selected and their combination is shown by highlighting it in blue color in Fig. 3.

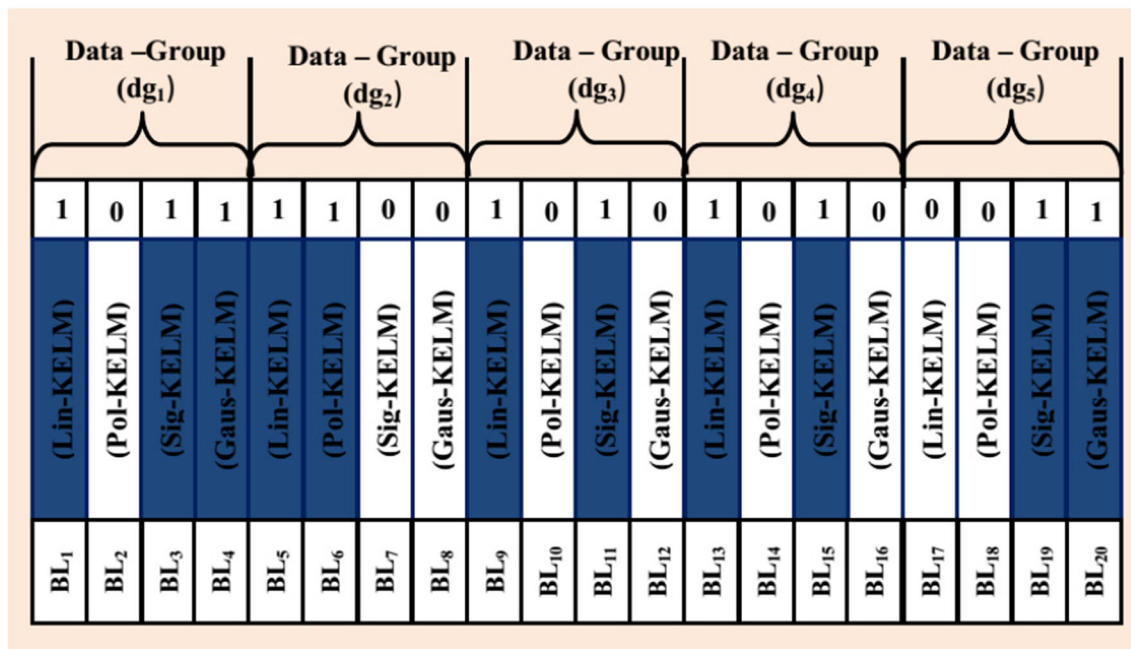


Fig. 3 Particle representation

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 AFPSO 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 AFPSO 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 tradeoff 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 classifiers can be enhanced by using the correct model integration method. For model integration, the proposed HAFPSO 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 B Land the KELM. KELM has gained wide popularity in various classification tasks [38, 44, 45] due to the promising results obtained. As the new attribute to the KELM classification function, the predicted output of the BL is 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.

6 Experimental evaluation

6.1 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 [46]. 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

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/MI
6	bmi	Body mass index (kg/m ²)	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

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 the remaining values are labeled as zero. Since our proposed work focuses on identifying type-II diabetes, the insulin measures and the number of times pregnant are 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 preprocessed data which is free from errors is taken by the disease prediction model for processing.

In our second dataset, the test values were obtained during the physical examination of different patients from a Diabetes Research Centre, Tamilnadu. From a total of 8700 samples, 5000 patients were not subjected to type-II diabetes and the remaining 3700 patients were affected with type-II diabetes. 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 Table 3 shown below. The type-II diabetic patient produces enough insulin but their body is not able to utilize it properly. Hence, the glucose remains in their blood and the blood glucose levels need to be checked to determine their insulin resistance capability. The blood glucose levels of these patients are measured via the FPG, PPG, HbA1c, and 2hpg tests done. From, these test values obtained, the insulin resistance is measured. The

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 year
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 that indicates the presence and absence of the disease	Positive, Negative

PPG test, FPG test, HbA1c, TC, and TGL, BMI, age, and 2hPG are the important factors present in the dataset to predict type-II diabetes in patients. The duration of diabetes mainly represents both the type-I and type-II diabetes present in the patient for a specific time duration. The type-II diabetes class label mainly represents the prediction of new type-II diabetes patients(incident diabetes).

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.

$$REplaced\ Value = \frac{Old-value(feature)-min(feature)}{max(feature)-Min(feature)} \quad (26)$$

$$\times (new_Max(feature)-new_Min(feature))$$

$$+ new_Max(feature)$$

6.2 Performance evaluation metrics

The efficiency of this proposed work is 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 accurately identified diabetic patients. The specificity indicates the probability of accurately identified non-diabetic patients.

$$Accuracy = \frac{A_{j1} + A_{j2}}{A_{j1} + A_{j2} + A_{j3} + A_{j4}} \quad (27)$$

$$Sensitivity = \frac{A_{j1}}{A_{j1} + A_{j4}} \quad (28)$$

$$Specificity = \frac{A_{j2}}{A_{j2} + A_{j3}} \quad (29)$$

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 and 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.

$$MCC = \frac{A_{j1} \times A_{j2} - A_{j3} \times A_{j4}}{\sqrt{(A_{j1} + A_{j3})(A_{j1} + A_{j2})(A_{j2} + A_{j3})(A_{j2} + A_{j4})}} \quad (30)$$

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

$$KS = \left[\frac{P(observed) - P(chance)}{1 - P(chance)} \right] \quad (31)$$

$$P(observed) = \left[\frac{A_{j1} + A_{j2}}{TN} \right] \quad (32)$$

$$P(chance) = \left[\frac{(A_{j1} + A_{j2}) * (A_{j1} + A_{j3}) * (A_{j2} + A_{j4})}{TN^2} \right] \quad (33)$$

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.

6.3 Parameter settings

This section gives a brief description of the 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 are taken as $C \in \{2^{-5}, 2^{-3}, \dots, 2^{15}\}$ and $\gamma \in \{2^{-15}, 2^{-13}, \dots, 2^3\}$. 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 $\chi=1$. 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.

Table 4 Parameters used for the model selection process

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

6.4 Experimental setup and test functions

The performance evaluation of this proposed model is compared with various conventional ML 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 [35], AFSO [37], GWO [47], *Non-dominated Sorting Genetic Algorithm- II* (NGSA-II) [48], and Pareto Archived Evolution Strategy (PAES) [49]. 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 three 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.

6.5 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:

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 A_j + \prod_{j=1}^m A_j $	30	[-10,10]	0
F ₃	$F_3(a) = \max_j \{ A_j , 1 \leq j \leq m\}$	30	[-100,100]	0
F ₄	$F_4(a) = \sum_{j=1}^m j a_j^4 + random[0, 1]$	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	$F_6(a) = \sum_{j=1}^m -a_j \sin\left(\sqrt{ a_j }\right)$	30	[-500,500]	-2095
F ₇	$F(a) = \left(\frac{1}{500} + \sum_{i=1}^{25} \frac{1}{i + \sum_{j=1}^2 (a_i - b_{ij})^6}\right)^{-1}$	2	[-65,65]	1
F ₈	$F_8(a) = \sum_{j=1}^{11} \left[x_j - \frac{a_1(y_j^2 + y_j a_2)}{y_j^2 + y_j a_3 + a_4} \right]^2$	4	[-5,5]	0.0029
F ₉	$F_9(a) = - \sum_{j=1}^6 c_j \exp\left(- \sum_{i=1}^6 x_{ij} (a_i - p_{ji})^2\right)$	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	$BV = \max_{j \leq N_i} F_j^*$
Standard Deviation(SD)	Measures the reliability of the proposed model	$SD = \sqrt{\frac{1}{N_i-1} \sum_{j=1}^{N_i} (F_j - \mu)^2}$

- 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 $O(TS*(n^3 + nf^2 + nf + NS^2))$. Complexity depends primarily on three factors: the number of basic learners, the KELM meta-learner, and the method of optimization.

7 Discussion

The KELM-HAFPSO model proposed has been assessed for Accuracy, Sensitivity, Specificity, MCC, and KS by means of 5-fold cross-validation based on two related datasets. Comparative Analysis was conducted between the proposed KELM-HAFPSO with the other seven competitive methods namely ELM-GA [50], Decision Tree C4.5-PSO [24], k-NN [23], MLP [21], LR [17], SVM [19], and NB [16].

8 Test functions to evaluate the performance of the 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 less 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 AFPSO algorithm's convergence speed and accuracy, respectively. In other words, the suggested AFPSO algorithm is statistically assumed to outperform the other five standard algorithms in terms of their applicability and practicality.

Figures 4, 5, and 6 show the different optimization curves obtained for 9 different benchmark functions by HAFPSO, PSO, AFPSO, 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 Figs. 5 and 6 respectively.

8.1 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 Fig. 7. The potential optimal solution is obtained from the input dataset on a single run of the HAFPSO algorithm. In Fig. 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 tradeoff between the potential optimal solutions. Each kernel is termed as a solution that ensures that the accuracy and kernel complexity is balanced. From Fig. 7, it is clear that solution 5 has the best accuracy obtained when the appropriate kernel is 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

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

the details of both the test accuracy and validation accuracy obtained for the proposed KELM-HAFPSO model in a single run. 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 BL which 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, 0.859, 0.908, and 0.962 respectively. For instance, in D-II the Accuracy, Sensitivity, Specificity, MCC, and KS values are 0.985, 0.987, 0.832, 0.918, and 0.977. 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-HAFPSO 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 Tables 9 and 10 prove 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.

Figures 8, 9, 10, 11, and 12 display 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

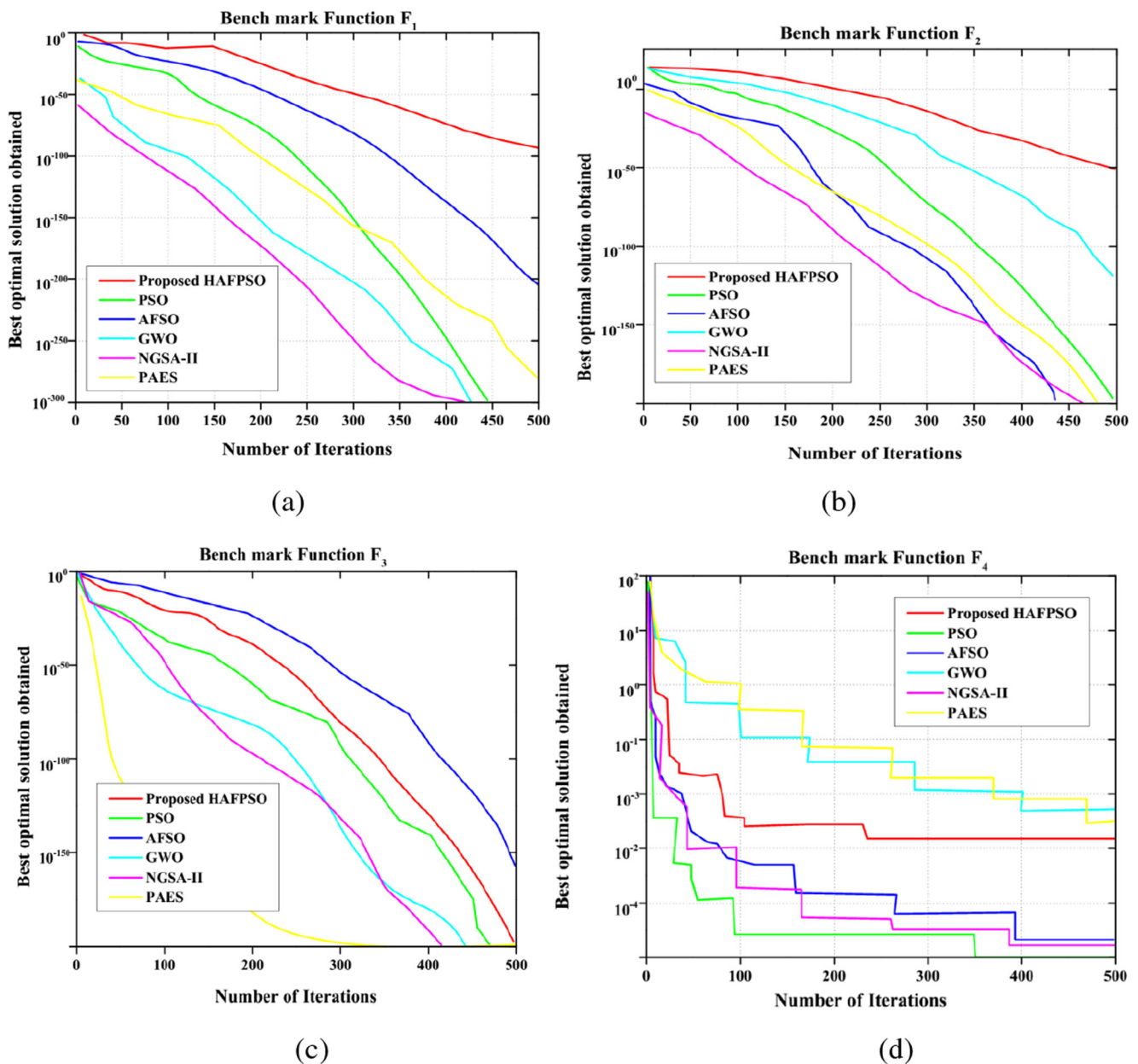


Fig. 4 Comparison of optimization results with Benchmark Functions. a–d: F_1 – F_4

approach is shown in Figure 8 for both datasets D-I and D-II when compared with other classifiers. The second highest accuracy value is obtained by C4.5-PSO. Figure 9 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 Fig. 10, our proposed model achieves higher specificity values when compared to other classifiers. The MCC scores obtained are shown in Fig. 11, where our proposed approach achieves a maximum MCC score(0.905) than other classifiers. In contrast, MLP and SVM achieve the lowest MCC values.

Figure 12 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

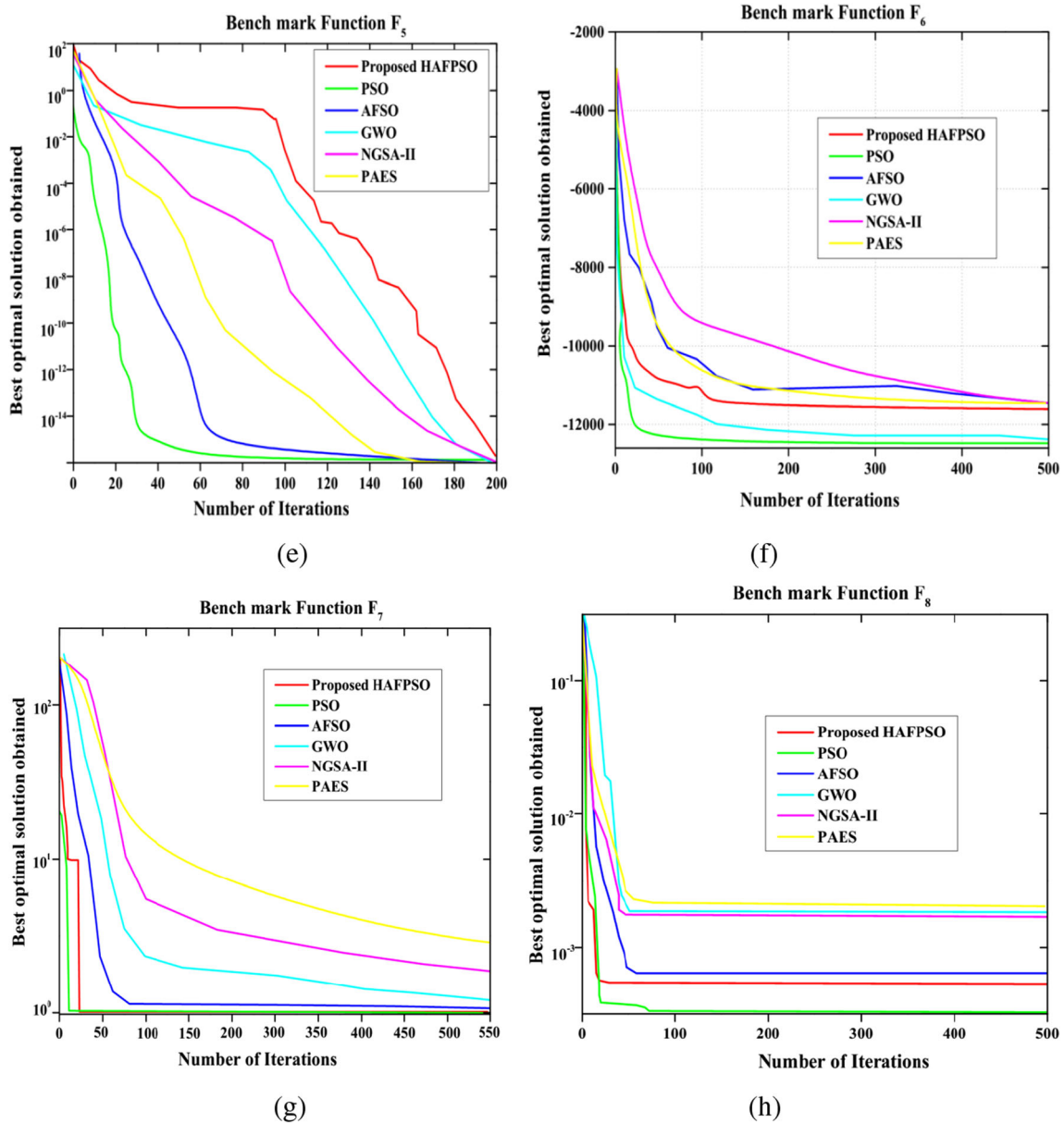


Fig. 5 Comparison of optimization results with Benchmark Functions. e–h: From F_5 – F_8

of 99.92% and 98.57% for Datasets D-I and D-II respectively.

8.2 Risk factor analysis

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

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

Such that

$$\|\alpha\|_0 = p_s \leq p, \alpha \in \{0, 1\}^f \tag{34}$$

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 $\|\cdot\|_0$ is a L_0 normalization function. α_i is a binary value that indicates whether the i^{th} feature in the subset is selected or not. The L_0 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.

$$\min_{\hat{\alpha}, \gamma} \frac{1}{N} \sum_{i=1}^N \left(T_i - f(\hat{\alpha}^T a_i; \gamma) \right)^2 + C_1 \|\hat{\alpha}\|_1 \tag{35}$$

C_1 is a regularized coefficient and $\|\cdot\|_1$ denotes the L_1 normalization. The value $\hat{\alpha}$ is not binary and it can take any real number values. If the i th entry $\hat{\alpha}$ is termed to be non-zero then

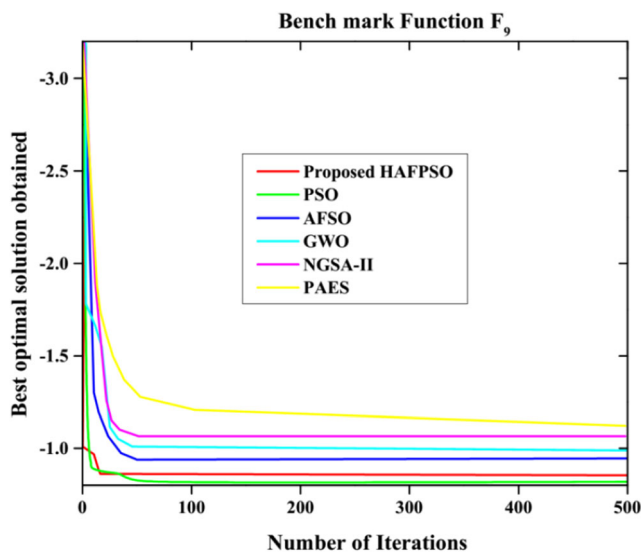
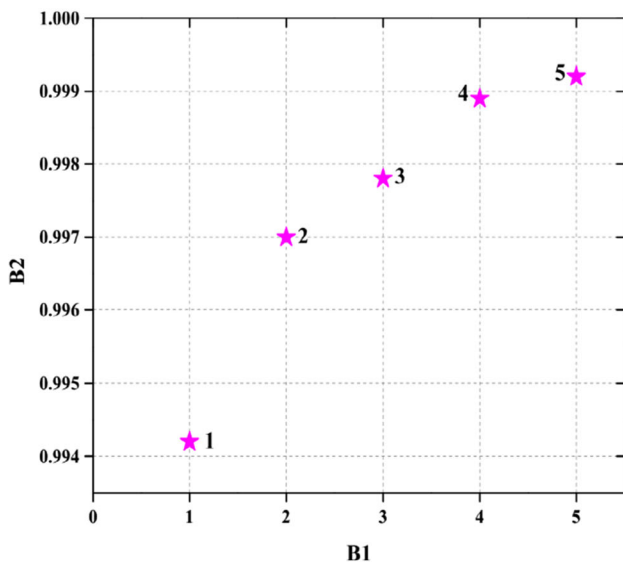


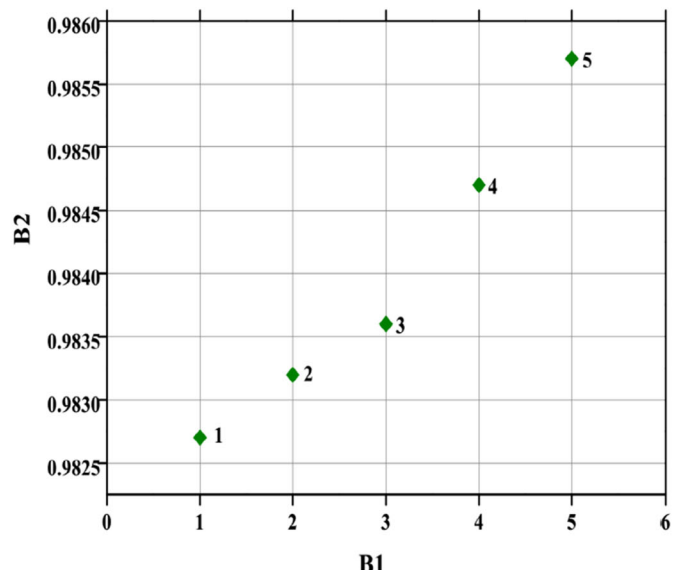
Fig. 6 Comparison of optimization results with Benchmark Functions F9

it is immediately selected as a feature. The feature can be extracted via a multivariate LR technique [51]. 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, \dots, a_n to form a linear function $\alpha_0 + \alpha_1 a_1 + \alpha_2 a_2 + \dots + \alpha_n a_n$. In the linear function formed α_0 is considered as the intercept value and $\alpha_1, \alpha_2, \dots, \alpha_n$ 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.

$$p = \frac{e^{(\alpha_0 + \alpha_1 a_1 + \alpha_2 a_2 + \dots + \alpha_n a_n)}}{1 + e^{(\alpha_0 + \alpha_1 a_1 + \alpha_2 a_2 + \dots + \alpha_n a_n)}} \tag{36}$$



(a)



(b)

Fig. 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 BL used (B2)	BL used	D-I		D-II	
			B1	Test accuracy	B1	Test accuracy
5	5	Lin-KELM,Gaus-KELM(2),Sig-KELM(2)	0.9992	0.9785	0.9857	0.9867
4	4	Lin-KELM, Gaus-KELM,Poly-KELM, and Sig-KELM	0.9989	0.9763	0.9847	0.9856
3	3	Gaus-KELM(2) ,Poly-KELM,	0.9978	0.9742	0.9836	0.9584
2	2	Lin-KELM, Sig-KELM	0.9970	0.9720	0.9832	0.9874
1	1	Gaus-KELM	0.9942	0.9680	0.9827	0.9845

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 Statistics	Accuracy	Sensitivity	Specificity	MCC	Kappa Statistics
1	4	0.98	0.98	0.82	0.89	0.94	0.97	0.97	0.82	0.89	0.95
2	5	0.97	0.97	0.79	0.91	0.96	0.96	0.98	0.79	0.89	0.97
3	10	0.96	0.97	0.79	0.92	0.96	0.99	0.98	0.79	0.89	0.97
4	11	0.96	0.98	0.84	0.87	0.98	0.99	0.98	0.84	0.90	0.96
5	6	0.94	0.98	0.84	0.89	0.98	0.97	0.96	0.84	0.90	0.96
6	7	0.98	0.92	0.84	0.89	0.98	0.99	0.94	0.84	0.90	0.97
7	8	0.99	0.97	0.86	0.89	0.97	0.96	0.94	0.86	0.90	0.94
8	5	0.98	0.97	0.82	0.93	0.96	0.94	0.97	0.82	0.91	0.98
9	7	0.97	0.98	0.82	0.94	0.95	0.97	0.98	0.82	0.92	0.98
10	8	0.97	0.97	0.81	0.96	0.94	0.97	0.99	0.81	0.94	0.97
11	8	0.99	0.97	0.78	0.89	0.97	0.99	0.99	0.78	0.90	0.96
12	9	0.98	0.99	0.86	0.89	0.96	0.98	0.98	0.86	0.90	0.98
13	7	0.99	0.99	0.85	0.92	0.94	0.98	0.97	0.85	0.89	0.94
14	8	0.97	0.98	0.91	0.92	0.97	0.98	0.98	0.91	0.91	0.95
15	8	0.96	0.98	0.86	0.91	0.96	0.97	0.98	0.86	0.92	0.95
16	5	0.89	0.98	0.85	0.91	0.96	0.95	0.97	0.85	0.92	0.97
17	4	0.94	0.98	0.82	0.95	0.94	0.94	0.95	0.82	0.94	0.97
18	7	0.94	0.97	0.91	0.89	0.97	0.96	0.98	0.91	0.94	0.98
19	7	0.96	0.97	0.92	0.91	0.97	0.93	0.97	0.92	0.92	0.98
20	6	0.98	0.96	0.87	0.92	0.98	0.98	0.99	0.87	0.89	0.97
Minimum	4	0.89	0.92	0.79	0.87	0.94	0.91	0.95	0.79	0.89	0.94
Median	7	0.98	0.98	0.85	0.90	0.96	0.98	0.98	0.83	0.91	0.97
Maximum	11	0.99	0.99	0.92	0.96	0.98	0.99	0.99	0.92	0.94	0.98
Average	7	0.98	0.98	0.85	0.90	0.96	0.98	0.98	0.85	0.91	0.97

The features present in both the datasets are applied to a logistic regression model and the results obtained are displayed in Tables 12 and 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

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	0.859	0.832	0.908	0.918	0.962	0.977
ELM-GA	0.901	0.895	0.914	0.902	0.794	0.785	0.874	0.854	0.876	0.869
C4.5-PSO	0.972	0.953	0.954	0.942	0.802	0.727	0.815	0.798	0.901	0.899
k-NN	0.834	0.821	0.832	0.814	0.820	0.810	0.854	0.849	0.894	0.875
MLP	0.875	0.874	0.847	0.844	0.817	0.795	0.794	0.784	0.874	0.869
LR	0.847	0.835	0.865	0.853	0.837	0.809	0.814	0.801	0.875	0.869
NB	0.958	0.934	0.917	0.904	0.826	0.815	0.823	0.815	0.901	0.895
SVM	0.787	0.774	0.847	0.836	0.814	0.801	0.748	0.735	0.862	0.853

Fig. 8 Accuracy of the proposed KELM-HAFPSO approach compared with other competitive classifiers

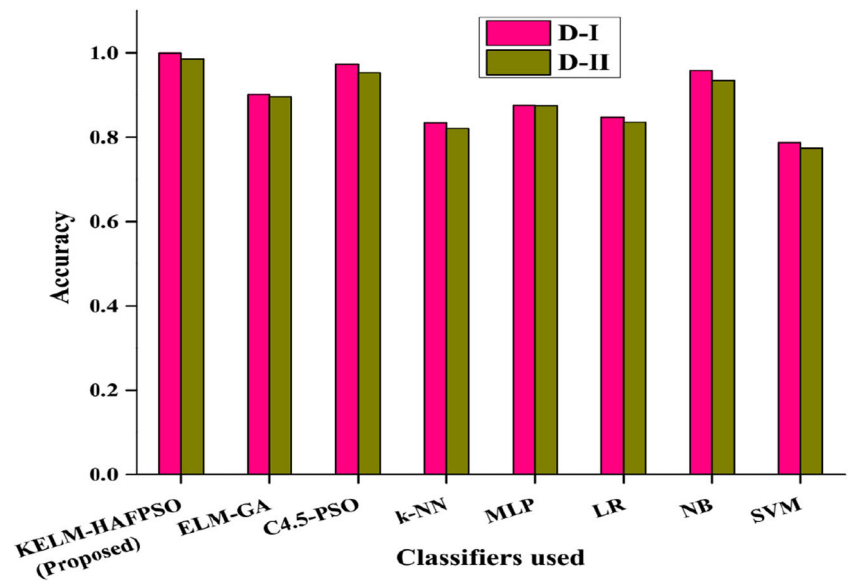


Fig. 9 Sensitivity of the proposed KELM-HAFPSO approach compared with other competitive classifiers

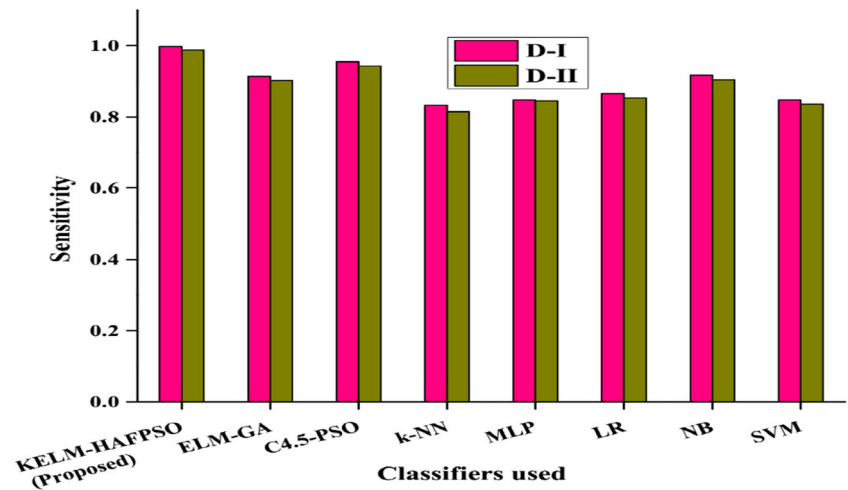


Fig. 10 Specificity of the proposed KELM-HAFPSO approach compared with other competitive classifiers

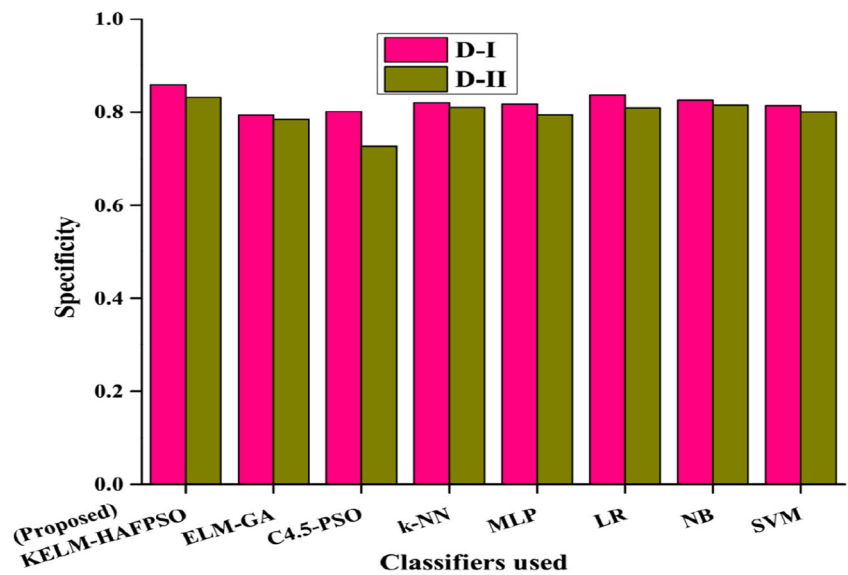


Fig. 11 MCC values of the proposed KELM-HAFPSO approach compared with other competitive classifiers

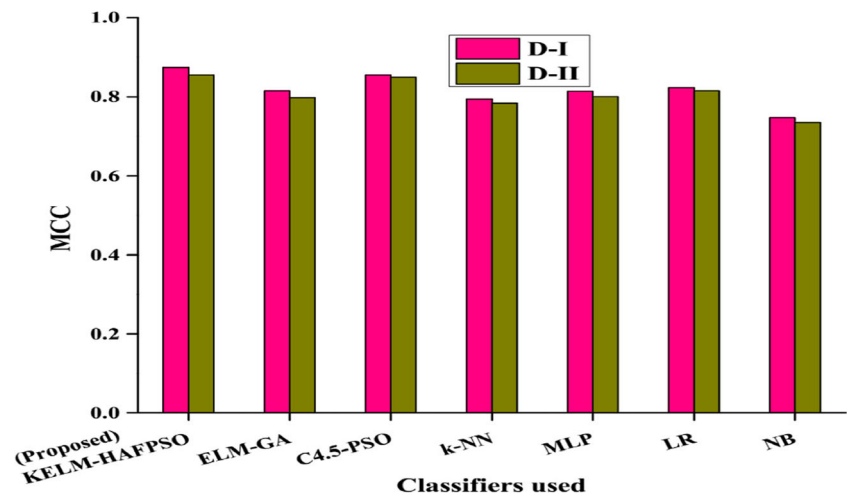


Fig. 12 Kappa Statistics values of the proposed KELM-HAFPSO approach compared with other competitive classifiers

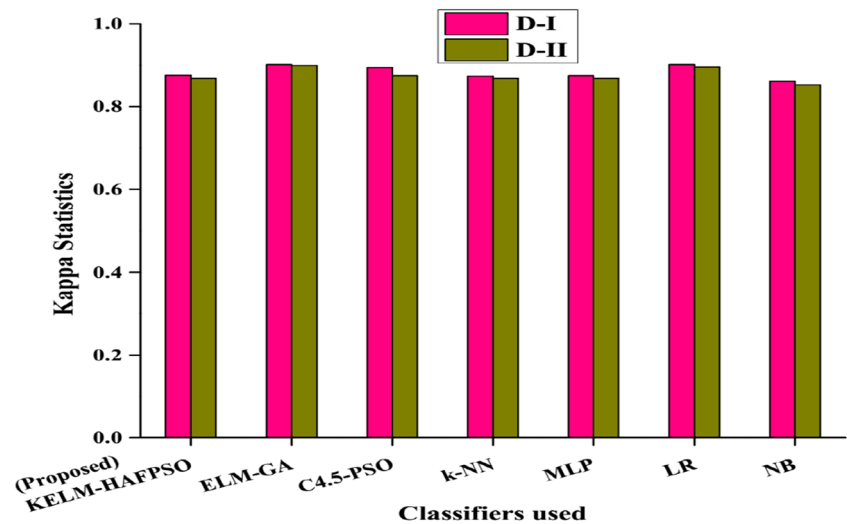


Table 11 Comparative analysis of classification accuracy of the proposed model with classifiers from literature

Sl.No	Reference number	Classifier used	Classifier accuracy
1	-	Proposed KELM-HAFPSO (D-I)	99.92%
		Proposed KELM-HAFPSO (D-II)	98.57%
2	[15]	SVM, FFC, and SMO	99.4%
3	[16]	BLR+NB	95%
4	[17]	LR+GBM	83.8%
5	[18]	Decision Tree	84%
6	[19]	SVM+XGBoost	74.7%
7	[21]	MLP	87%
8	[21]	AD, TRF, SVM, GTB	86%
9	[22]	LR,K-NN,MDR, and SVM	>85%
10	[23]	LR	74%
11	[23]	k-NN	83%
12	[24]	PSO and Decision Tree(C4.5)	98.60%
13	[25]	K-Means and LR	95.92%
14	[26]	EBFMA classifier, PSO, M-LR	92%
16	[20]	SVM, GA	83.80%

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	-0.912	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
dpg	1.099	0.210	3.532	0.000
age	0.017	0.009	1.299	0.179

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 endeavors 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, Tamilnadu incorporating a total of 8700 samples. These two datasets were preprocessed using the Min-Max Normalization approach. The accurate classification is yielded in this paper by using a stacking based kernel approach for the

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	0.912	-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

KELM classifier. The KELM uses 4 BL namely 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 Statistics on both the datasets applied. In the future, we plan to implement this approach in different high dimensional datasets that include data about proteomics, genetics, metabolomics, etc.

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

Human and animal rights This article does not contain any studies with human or animal subjects performed by any of the authors.

Informed consent Informed consent was obtained from all individual participants included in the study.

References

1. Cho N, Shaw JE, Karuranga S, Huang Y, da Rocha Fernandes JD, Ohlrogge AW, Malanda B (2018) IDF Diabetes Atlas: Global estimates of diabetes prevalence for 2017 and projections for 2045. *Diabetes Res Clin Pract* 138:271–281
2. Bogatyrev SN (2016) Physical activity and type 2 diabetes mellitus risk: population studies review. *Diabetes Mellitus* 19(6):486–493
3. Punthakee Z, Miller ME, Launer LJ, Williamson JD, Lazar RM, Cukierman-Yaffe T, Seaquist ER, Ismail-Beigi F, Sullivan MD, Lovato LC, Bergenstal RM (2012) Poor cognitive function and risk of severe hypoglycemia in type 2 diabetes: post hoc epidemiologic analysis of the ACCORD trial. *Diabetes Care* 35(4):787–793
4. Bonds JA, Hart PC, Minshall RD, Lazarov O, Haus JM, Bonini MG (2016) Type 2 Diabetes Mellitus as a Risk Factor for Alzheimer's Disease. In: *Genes, Environment and Alzheimer's Disease*. Academic Press, pp 387–413
5. American Diabetes Association (2020) 2. Classification and Diagnosis of Diabetes: Standards of Medical Care in Diabetes—2020. *Diabetes Care* 43(Supplement 1):S14–S31
6. World Health Organization (2011) . Use of glycated haemoglobin (HbA1c) in diagnosis of diabetes mellitus: abbreviated report of a WHO consultation. World Health Organization. <https://apps.who.int/iris/handle/10665/70523>
7. Abdul-Ghani MA, DeFronzo RA (2009) Plasma glucose concentration and prediction of future risk of type 2 diabetes. *Diabetes Care* 32(suppl 2):S194–S198

8. Nilashi M, Bin Ibrahim O, Ahmadi H, Shahmoradi L (2017) An analytical method for diseases prediction using machine learning techniques. *Comput Chem Eng* 106:212–223
9. Hassan BA (2020) CSCF: a chaotic sine cosine firefly algorithm for practical application problems. *Neural Comput & Applic*. <https://doi.org/10.1007/s00521-020-05474-6>
10. Hassan BA, Rashid TA (2020) Operational framework for recent advances in backtracking search optimisation algorithm: A systematic review and performance evaluation. *Appl Math Comput* 370:124919
11. Jose J, Gautam N, Tiwari M, Tiwari T, Suresh A, Sundararaj V, Rejeesh MR (2021) An image quality enhancement scheme employing adolescent identity search algorithm in the NSST domain for multimodal medical image fusion. *Biomed Signal Process Control* 66:102480
12. Yang S, Wei R, Guo J, Xu L (2017) Semantic inference on clinical documents: combining machine learning algorithms with an inference engine for effective clinical diagnosis and treatment. *IEEE Access* 5:3529–3546
13. Shalev-Shwartz S, Ben-David S (2014) *Understanding machine learning: From theory to algorithms*. Cambridge University Press, Cambridge
14. Kuhn M, Johnson K (2013) *Applied predictive modeling* (Vol. 26). Springer, New York
15. Devi RDH, Bai A, Nagarajan N (2020) A novel hybrid approach for diagnosing diabetes mellitus using farthest first and support vector machine algorithms. *Obes Med* 17:100152
16. Lee BJ, Kim JY (2015) Identification of type 2 diabetes risk factors using phenotypes consisting of anthropometry and triglycerides based on machine learning. *IEEE J Biomed Health Inform* 20(1):39–46
17. Lai H, Huang H, Keshavjee K, Guergachi A, Gao X (2019) Predictive models for diabetes mellitus using machine learning techniques. *BMC Endocr Disord* 19(1):1–9
18. Orabi KM, Kamal YM, Rabah TM (2016) Early predictive system for diabetes mellitus disease. In: *Industrial Conference on Data Mining*. Springer, Cham, pp 420–427
19. Singh N, Singh P, Bhagat D (2019) A rule extraction approach from support vector machines for diagnosing hypertension among diabetics. *Expert Syst Appl* 130:188–205
20. Singh N, Singh P (2020) Stacking-based multi-objective evolutionary ensemble framework for prediction of diabetes mellitus. *Biocybern Biomed Eng* 40(1):1–22
21. Xiong XL, Zhang RX, Bi Y, Zhou WH, Yu Y, Zhu DL (2019) Machine Learning Models in Type 2 Diabetes Risk Prediction: Results from a Cross-sectional Retrospective Study in Chinese Adults. *Curr Med Sci* 39(4):582–588
22. Farran B, Channanath AM, Behbehani K, Thanaraj TA (2013) 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* 3(5):e002457
23. Farran B, AlWotayan R, Alkandari H, Al-Abdulrazzaq D, Channanath A, Thangavel AT (2019) 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. *Front Endocrinol* 10:624
24. Abdullah AS, Selvakumar S (2019) 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 Comput* 23(20):9995–10017
25. Wu H, Yang S, Huang Z, He J, Wang X (2018) Type 2 diabetes mellitus prediction model based on data mining. *Inform Med Unlocked* 10:100–107
26. Marateb HR, Mansourian M, Faghihimani E, Amini M, Farina D (2014) A hybrid intelligent system for diagnosing microalbuminuria in type 2 diabetes patients without having to measure urinary albumin. *Comput Biol Med* 45:34–42
27. Sundararaj V (2016) An efficient threshold prediction scheme for wavelet based ECG signal noise reduction using variable step size firefly algorithm. *Int J Intell Eng Syst* 9(3):117–126
28. Vinu Sundararaj V (2019a) Optimal task assignment in mobile cloud computing by queue based Ant-Bee algorithm. *Wirel Pers Commun* 104(1):173–197
29. Sundararaj V (2019b) Optimised denoising scheme via opposition-based self-adaptive learning PSO algorithm for wavelet-based ECG signal noise reduction. *Int J Biomed Eng Technol* 31(4):325–345
30. Vinu S, Muthukumar S, Kumar RS (2018) An optimal cluster formation based energy efficient dynamic scheduling hybrid MAC protocol for heavy traffic load in wireless sensor networks. *Comput Secur* 77:277–288
31. Sundararaj V, Anoop V, Dixit P, Arjaria A, Chourasia U, Bhambri P, MR, R. and Sundararaj, R. (2020) CCGPA-MPPT: Cauchy preferential crossover-based global pollination algorithm for MPPT in photovoltaic system. *Prog Photovolt Res Appl* 28(11):1128–1145
32. Paolo M, Pio G, D'Elia D, Ceci M (2020) Exploiting transfer learning for the reconstruction of the human gene regulatory network. *Bioinformatics* 36(5):1553–1561
33. Barracchia EP, Pio G, D'Elia D, Ceci M (2020) Prediction of new associations between ncRNAs and diseases exploiting multi-type hierarchical clustering. *BMC Bioinform* 21(1):1–24
34. Eberhart, Shi Y (2001) Particle swarm optimization: developments, applications, and resources. In: *Proceedings of the 2001 Congress on Evolutionary Computation (IEEE Cat. No.01TH8546)*, Seoul, South Korea, vol 1, pp 81–86
35. Kennedy J, Eberhart RC, Shi Y (2001) *The Particle Swarm*, *Swarm Intelligence*, pp. 287–325.
36. Engelbrecht A (2012) Particle swarm optimization: Velocity initialization, 2012 IEEE Congress on Evolutionary Computation, Brisbane, QLD, pp. 1-8.
37. Neshat M, Sepidnam G, Sargolzaei M, Toosi AN (2014) Artificial fish swarm algorithm: a survey of the state-of-the-art, hybridization, combinatorial and indicative applications. *Artif Intell Rev* 42:965–997
38. Chen H, Wang S, Li J, Li Y (2007) A Hybrid of Artificial Fish Swarm Algorithm and Particle Swarm Optimization for Feedforward Neural Network Training, *Proceedings on Intelligent Systems and Knowledge Engineering (ISKE2007)*, 2007
39. Hoang N-D, Bui DT (2017) Slope Stability Evaluation Using Radial Basis Function Neural Network, Least Squares Support Vector Machines, and Extreme Learning Machine. In: *Handbook of Neural Computation*, pp 333–344 2017
40. Huang G, Huang G-B, Song S, You K (2015) Trends in extreme learning machines: A review. *Neural Netw* 61:32–48
41. Borwein JM, Lewis AS (2000) Karush-Kuhn-Tucker Theory, *Convex Analysis and Nonlinear Optimization*, pp. 153–177.
42. Mercer's Theorem, Feature Maps, and Smoothing. [Online]. Available: <http://people.cs.uchicago.edu/~niyogi/papersps/MinNiyYao06.pdf>. [Accessed: 27-Jan-2020].
43. Cawley GC, Talbot NLC (2007) Preventing over-fitting in model selection via Bayesian regularisation of the hyper-parameters. *J Mach Learn Res* 8:841–861
44. Liu T, Hu L, Ma C, Wang Z-Y, Chen H-L (2015) A fast approach for detection of erythemato-squamous diseases based on extreme learning machine with maximum relevance minimum redundancy feature selection. *Int J Syst Sci* 46(5):919–931
45. Zhao D, Huang C, Wei Y, Yu F, Wang M, Chen H (2016) An effective computational model for bankruptcy prediction using kernel extreme learning machine approach. *Comput Econ*:1–17
46. Smith JW, Everhart JE, Dickson WC, Knowler WC, Johannes RS (1988) Using the ADAP learning algorithm to forecast the onset of diabetes mellitus. In: *Proceedings of the Symposium*

- on Computer Applications and Medical Care. IEEE Computer Society Press, pp 261–265
47. Albina K, Lee SG (2019) Hybrid Stochastic Exploration Using Grey Wolf Optimizer and Coordinated Multi-Robot Exploration Algorithms. *IEEE Access* 7:14246–14255
 48. Deb K, Agrawal S, Pratap A, Meyarivan T (2000) A Fast Elitist Non-dominated Sorting Genetic Algorithm for Multi-objective Optimization: NSGA-II, Parallel Problem Solving from Nature PPSN VI Lecture Notes in Computer Science, pp. 849–858, 2000.
 49. Oltean M, Grosan C, Abraham A, Koppen M (2005) Multiobjective optimization using adaptive Pareto archived evolution strategy, 5th International Conference on Intelligent Systems Design and Applications (ISDA'05), Warsaw, 2005, pp. 558–563.
 50. Alharbi A, Alghahtani M (2018) Using Genetic Algorithm and ELM Neural Networks for Feature Extraction and Classification of Type 2-Diabetes Mellitus. *Applied Artificial Intelligence*, 1–18. <https://doi.org/10.1080/08839514.2018.1560545>
 51. Liu L (2018) Advanced Biostatistics and Epidemiology Applied in Heart Failure Study. In: *Heart Failure: Epidemiology and Research Methods*, pp 83–102

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Kanimozhi Natesan received her B.TECH in Information Technology from the Annai Mathammal Sheela Engineering College, Namakkal in 2008 and M.E. in Computer Science and Engineering at the Annai Mathammal Sheela Engineering College, Namakkal in 2010. She is pursuing her Ph.D. in Information and Communication in Anna University, Chennai. She is currently working as an Assistant Professor in the Department of Computer Science

and Engineering at GKM College of Engineering and Technology (Affiliated to Anna University), Chennai, Tamilnadu, India. She has 7 years of teaching experience and three years in research. Her

specializations include Network Security, Cloud Computing, Internet of Things, Deep Learning and Image Processing. Her current research interests are Deep learning and Machine Learning.



Dr. G. Singaravel received the D.C.T degree in Computer Technology in 1995 from Kongu Polytechnic College and the Undergraduate degree B.E. (Electronics and communication and Engineering) in 1999 in Kongu Engineering College from the Bharathiyar University. He completed his Master degree M.E. (Computer Science and Engineering) in 2002 at Arulmigu Kalasalingam College of Engineering from Madurai Kama raj University. He received

his Ph.D. degree in the area of Information and Communication from Anna University, Chennai in 2010. He has 21 years of teaching experience. He is a Professor and the Head of the Department of Information Technology, K.S.R College of Engineering (Autonomous), Tiruchengode. He is recognized as a Supervisor for Ph.D Programme (By Research) in Information Technology for Anna University, Chennai. He is in the editorial board member of International/National Journals. He has published 120 papers in international, national journals and conference proceedings and having Google Scholar citations and h-index. He has guided 25 Master Degree projects. He is guiding more than 14 Ph.D research scholars in various recent domains. Life Member of various Professional Bodies like ISTE, MCSI, ISRD etc., Funds received from AICTE, DST, SERB, and CSIR to organizing the Research Projects, Seminar and Faculty Development Program. His areas of research include Software Engineering, Agile Project Management, Green & Cloud Computing, Big Data Analytics, etc.,