IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received October 1, 2019, accepted October 22, 2019, date of publication October 29, 2019, date of current version November 22, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2950220

Metaheuristic-Based Dimensionality Reduction and Classification Analysis of PPG Signals for Interpreting Cardiovascular Disease

SUNIL KUMAR PRABHAKAR[1], HARIKUMAR RAJAGURU[2], AND SEONG-WHAN LEE[1,3]

[1]Department of Brain and Cognitive Engineering, Korea University, Seoul 02841, South Korea
[2]Department of Electronics and Communication Engineering, Bannari Amman Institute of Technology, Erode 638401, India
[3]Department of Artificial Intelligence, Korea University, Seoul 02841, South Korea

Corresponding author: Seong-Whan Lee (sw.lee@korea.ac.kr)

This work was supported by Defense Acquisition Program Administration (DAPA) and Agency for Defense Development (ADD) of Korea (06-201-305-001), A Study on Human Computer Interaction Technology for the Pilot Status Recognition.

ABSTRACT One of the most useful methodologies which provide a specific waveform showing the pulsating peripheral blood flow in a non-invasive manner is Photoplethysmography (PPG). The design, application and implementation of a PPG system are quite inexpensive and have a very easy maintenance. Without having direct contact with the surface of the skin, PPG can easily take the measurements. Therefore, PPG has a good medical competency and due to its widespread availability, it has a lot of advantages. A PPG signal can sometimes be substituted or complemented by an Electrocardiography (ECG) signal as it can provide Heart Rate Variability (HRV) analysis. In this work, an in-depth analysis of classification of Cardiovascular Disease (CVD) is done with the help of Capnobase dataset. Initially, metaheuristic optimization algorithms are utilized as dimensionality reduction techniques and then the dimensionally reduced values are classified with the help of different classifiers for the classification of CVD. The results show that for the PPG normal cases, a high classification accuracy of 99.48% is obtained when Chi square Probability Density Function (PDF) optimized values are classified with Artificial Neural Networks (ANN) and a second highest classification accuracy of 98.96% is obtained when Chicken swarm optimized values are classified with Naïve Bayesian Classifier (NBC). Similarly when the PPG abnormal cases or PPG with CVD cases are concerned, a high classification accuracy of 99.48% is obtained when Chi square PDF optimized values are classified with Logistic Regression and a second highest classification accuracy of 98.96% is obtained when Chi square CDF optimized values are classified with Gaussian Support Vector Machine (SVM) and when Chicken swarm optimized values are classified with NBC.

INDEX TERMS PPG, HRV, cardiovascular, classification, optimization.

I. INTRODUCTION

To detect the changes in the blood volume levels of the blood vessels by means of utilizing optical techniques, PPG serves as an efficient one as it is non-invasive and quite simple in nature [1]. The light is passed through the blood vessels by means of an infrared emitter. The light reflected from the vessels is detected by detector. Both the emitter and detector is placed on a transducer and then it is kept on a finger. Generally, the configuration used to keep the emitter and detector on a finger is by means of using a reflected type. To obtain PPG signals experimentally, the transducer used is Infrared plethysmography [2]. An Infrared (IR) photoelectric sensor is utilized by the transducer to trace the changes in the tissues of blood volume. The variation in the blood volume is reflected by the obtained PPG waveforms and is generally synchronized with the heartbeat.

Through various invasive and non-invasive methods, distinct diseases can be diagnosed. The most significant invasive techniques are endoscopy, angioplasty etc. They are quite expensive, time-consuming and require a high maintenance. The non-invasive techniques are quite reliable, easy to use and have good compatibility. To record the changes in the blood, various non-invasive plethysmography techniques

The associate editor coordinating the review of this manuscript and approving it for publication was Zijian Zhang.

are used such as strain gauge, photo impedance plethysmography [3]. PPG is a simple technique that allows for incessant, non-invasive determination of heart rate which is mandatory in the Intensive Care Unit (ICU). It is also used for the peripheral blood flow assessment and venous filing time. For measuring oxygen saturation, cardiac output, blood pressure, the PPG technique is used in a wide range of commercially available medical devices [4]. For assessing autonomic functions and for detecting peripheral vascular diseases, it is used. The recording equipment for a PPG is lighter and less bulky than an Electrocardiogram (ECG) device. A very few opto electronic components are required by PPG. A light is used to illuminate the tissue and to measure the minor variations in the light intensity associated with catchment volume changes. Despite being very simple, the origin of the various components of the PPG signals are not understood [5]. It is generally known that they can provide general information about the cardiovascular system. Often contaminated with noise and artifacts, pre-processing is often required to remove it and it includes procedures like normalization and artifact removal [6]. The main artifacts are powerline interference, motion artifacts, low amplitude PPG signal and Premature Ventricular Contraction (PVC). The general parameters which are extracted from the PPG signal analysis are the Heart Rate (HR), Pulse Transit Time (PTT), Pulse Wave Velocity (PWV) and Stiffness Index (SI). The types of features extracted from PPG are systolic amplitude, pulse width, pulse area, peak-to-peak interval, pulse interval, augmentation index and large artery stiffness index [7].

Some of the prominent works in the PPG signal analysis and its application to various biomedical fields are mentioned as below. For heart rate monitoring from PPG signals, a robust motion artifact reduction algorithm was designed by Baca *et al.* [8]. Independent Component Analysis (ICA) was used to reduce the motion artifact in PPG by Kim and Yoo [9]. Kalman smoother was used to eliminate the motion artifacts from PPG signals [10]. Wavelets were evaluated in detail for reducing the motion artifacts in PPG signals by Raghuram *et al.* [11]. An AS-LMS adoptive filter was utilized to reduce the artifacts in PPG signals by Ram *et al.* [12]. A wearable PPG sensor was used for the development of the irregular pulse detection techniques by Suzuki *et al.* [13]. The Heart Rate Turbulence (HRT) was analyzed by Gil *et al.* which was mainly based on PPG [14]. For automatic detection of Pre-ventricular Contraction (PVC), a PPG method was used by Solosenko *et al.* [15]. With the help of chaotic features and Higher Order Statistics (HOS), the automatic detection of PVC based on PPG was done by Yousefi *et al.* [16]. Cardiac arrhythmia classification was done for PPG signals by Polania *et al.* [17]. From PPG and ECG signals, the inter beat and inter pulse intervals was found by Nano *et al.* [18]. An automatic beat detection algorithm was devised for pressure signals by Aboy *et al.* [19]. In this work, an interesting approach of utilizing various metaheuristic algorithms for dimensionality reduction was done and then the dimensionally reduced values are classified using various post classifiers.

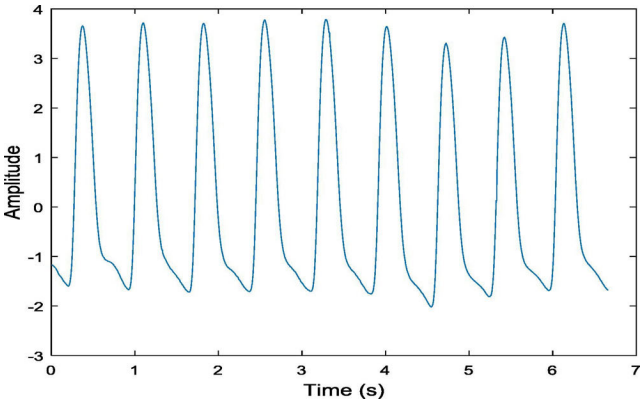

FIGURE 1. Normal Signal.

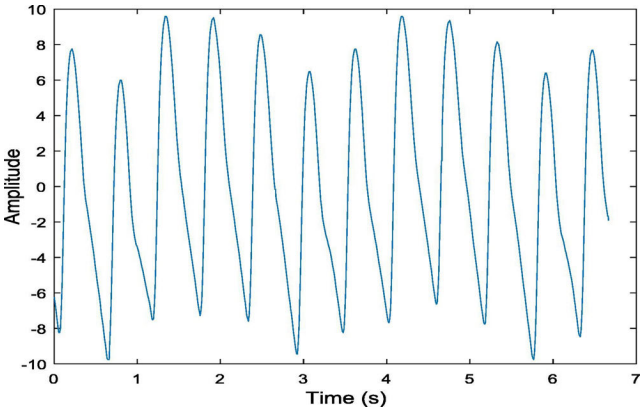

FIGURE 2. Abnormal Signal.

II. MATERIALS AND METHODS

This dimensionality reduction and classification methodology has been validated utilizing many PPG recordings with a different waveform of morphology from IEEE TMBE Pulse Oximetry dataset of the Capnobase database [20]. The dataset has raw PPG signal recordings of 8-minute duration. Annotated representation signals such as Inhaled and Exhaled CO_2 (Capnogram), pressure and respiratory flow is contained in the database. The Capnobase has consistent recordings of about 13 adults and 29 pediatrics. Observable events like rebreathing, cardiac oscillations, hyperventilation, hypoventilation, in adequate ventilation, apnea, respiratory rate changes are present. The entire IEEE benchmark dataset from Capnobase database (42 records) has been considered for the experiment, 14 normal and 28 for Cardiovascular detection. Fig. 1 and Fig. 2 shows the normal and abnormal versions of the PPG signal respectively.

In this study, the classification of PPG, whether it is normal, or it has CVD has been carried out. The PPG signal dataset has been sampled at a sampling rate of 100 Hz. The data sample length obtained is around 1,50,000 samples. The block diagram of the work is shown in Fig. 3. The PPG signals are pre-processed with the help of

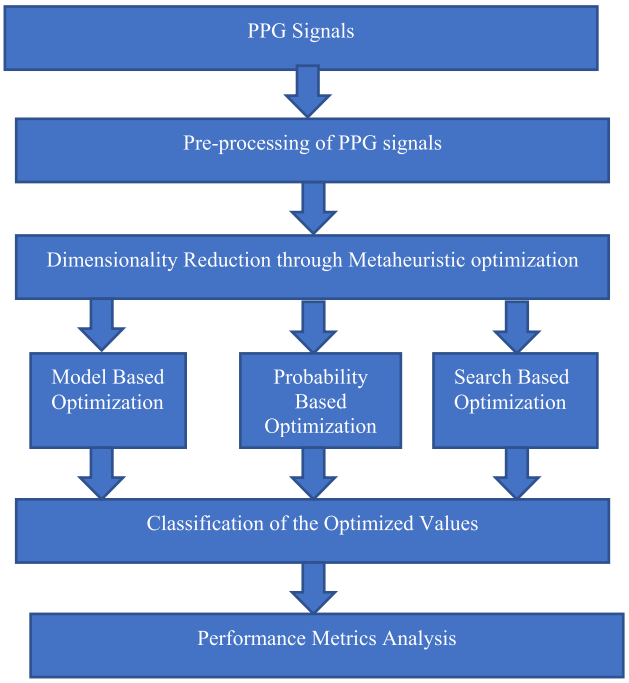

FIGURE 3. Block Diagram of the Work.

Independent Component Analysis (ICA) and then the dimensionality of it is reduced with the help of metaheuristic optimization and finally classified for the analysis of normal/CVD.

III. DIMENSIONALITY REDUCTION TECHNIQUES

The optimization techniques used for dimensionality reduction here are categorized under three different categories such as model based optimization, probability-based optimization and search based optimization techniques. Model based optimization includes Density Peaks optimization, probability based optimization includes Chi square Probability Density Function (PDF) optimization and Chi square Cumulative Density Function (CDF) optimization and search based optimization includes Harmonic search optimization, Elephant search optimization, Particle swarm optimization, Chicken swarm optimization and Cat swarm optimization techniques.

A. CHI SQUARE PDF OPTIMIZATION

The chi square statistic looks quite different from other statistical tests. The chi square statistic is the same for both the test of independence and goodness of fit test [21]. When the data is obtained from the sample, it is referred to as observed number of cases. These are represented as the frequency of occurrence for every category into which the data is segregated. In chi square tests, the null hypothesis makes a statement which concerns about the expected number of cases in every category if the hypothesis is correct. The chi square test is based on the difference between the observed and the expected values for every category. The chi square statistic is defined as:

$$\chi^2 = \sum_i \frac{(O_i - E_i)^2}{E_i} \tag{1}$$

where O_i denotes the observed number of cases in category i, E_i denotes the number of cases in category i

By computing the difference between the observed number of cases and the expected number of cases in every category, the chi square statistic is obtained. Then the squaring of the difference is done and divided by the expected number of cases in that category. Such values are added for all the categories and then the total value is referred as chi square value. Depending on the data distribution, the null hypothesis can be of great concern. For the chi square test, the null and alternative hypothesis is as follows:

$$\begin{aligned} H_0 &: O_i = E_i \\ H_1 &: O_i \neq E_i \end{aligned} \tag{2}$$

If the claim done in the null hypothesis is true, then the observed values and the expected values are pretty close to each other and $O_i - E_i$ is quite small for every category. When the observed data does not ally to what has been expected on the context of null hypothesis, then the difference between them is, $O_i - E_i$ is large. When the null hypothesis is not true, the chi square statistic is large and when the null hypothesis is true, the chi square statistic is small. Based on the number of categories which are utilized in the calculation of statistic, the degree of freedom is dependent on it. The chi square statistic and chi square distribution allows the researcher to determine whether the data is distributed as claimed or not. The chi square distribution is easily implemented by the chi square distribution class. As it has one constructor which assumes the degree of freedom as its only argument, extreme optimization is done easily.

B. DENSITY PEAKS OPTIMIZATION

Density peaks optimization through clustering are also a widely used technique in clustering analysis. Here the clusters are represented by a higher density than their respective neighbours and by a pretty large distance from points containing the higher densities [22]. Two quantities are utilized in density peak clustering, firstly the density γ_j of point y_j and the other is its respective distance ρ_j from points having the higher densities. The definition of the density γ_j of point y_j is expressed as:

$$\sigma_j = \sum_i \exp\left(-\frac{D\left(y_j, y_i\right)^2}{2\sigma^2}\right) \tag{3}$$

where $D\left(y_j, y_i\right)$ is the Euclidean distance between points y_j and y_i, σ indicates variance.

Pseudocode for the definition of σ is given in Pseudocode 1.

Pseudocode 1:

(1) Input: Dataset $Y = \{y_j, y_j \in \Re^Q, j = 1, 2, \ldots, m\}$, the total number of points m and a predefined parameter w
(2) Output: The value of σ
(3) Start
(4) The pairwise distance between the points is calculated and sorted in ascending order, that is $\{d_1, d_2, \ldots, d_{m(m-1)/2}\}$
(5) Threshold calculation $th = \lceil w * m(m-1)/2 \rceil$
(6) Calculate $\sigma = d_{th}$
(7) End

Between the points y_j and the other points with higher densities, the distance ρ_j is denoted as:

$$\rho_j = \begin{cases} \min\limits_{i:\gamma_j:\gamma_i} \left(d\left(y_j, y_i\right)\right), & if\ \gamma_j < \gamma_i \\ \max\limits_{i} \left(d\left(y_j, y_i\right)\right), & otherwise \end{cases} \tag{4}$$

For each point when the density γ_j and the distance ρ_j are calculated, the points with higher γ_j and ρ_j are considered as cluster centres.

C. HARMONIC SEARCH OPTIMIZATION

When working with Harmonic search optimization through clustering, an unconstrained optimization problem is expressed as follows $\min f(Z)$, such that:

$$Qz_i \le z_i \le Vz_i \tag{5}$$

where $f(Z)$ denotes the object function, Z indicates the set of every decision variable, z_i is the i^{th} decision variable, Qz_i is the lower bound of the i^{th} decision variable and Vz_i is the upper bound of the i^{th} decision variable. The following five steps should be taken to solve the standard Harmonic Search problem [23].

Step 1: The parameters of the problem algorithm are initialized. The problem parameters like m, Qz_i, Vz_i are initialized in this step. Four more algorithm parameters are initialized which includes Harmony Size (HMS), Pitch Adjusting Rate (PAR), Harmony Memory Considering Rate (HMCR) and the stopping criterion or the highest number of improvisation ($T_{\max}$).

Step 2: The harmony memory is initialized. This Harmonic Search is quite similar to Genetic Algorithms (GA). In GA, it is population dependent optimization algorithm, but the population in Harmonic search is referred as Harmony Memory (HM) which is constructed as solution vector. The representation of HM is as follows:

$$HM = \begin{bmatrix} z_1^1 & z_2^1 & \cdots & z_m^1 \\ z_1^2 & z_1^2 & \cdots & z_m^2 \\ \vdots & \vdots & \vdots & \vdots \\ z_1^{HMS} & z_2^{HMS} & \cdots & z_m^{HMS} \end{bmatrix} \tag{6}$$

where z_j^i is the j^{th} decision of the i^{th} solution vector. The HM is initialized randomly in a general manner. For z_j^i, with the help of following equation it can be generated as:

$$z_j^i = Qz_j + \left(Vz_j - Qz_j\right) \times rand\ () \tag{7}$$

where *rand* () expresses a random number between 0 and 1.

Step 3: A new harmony is improved. A novel harmonic vector $Z^{new} = \left(z_1^{new}, z_2^{new}, \ldots, z_m^{new}\right)$ is improved using the following three rules (1) pitch adjustment (2) harmony memory consideration (3) randomization. The probability of pitch adjustment is dependent on PAR and the probability of harmony consideration is dependent on HMCR. For example, the i^{th} variable z_i^{new} of the normal harmony vector is expressed as follows:

$$\begin{aligned} z_i^{new} &= z_i^{new} \in \left\{z_i^1, z_i^2, \ldots, z_i^{HMS}\right\} \\ z_i^{new} &= Qz_j + (Vz_i - Qz_i) \times rand\ () \end{aligned} \tag{8}$$

Before the generation of the variable, a random vector r_1 should be generated between 0 and 1, so that r_1 could be compared with HMCR, if $r_1 \le HMCR$, it can be utilized for variable improvisation. If the variable is obtained from the HM, then it should be adjusted with probability *PAR* as follows:

$$z_i^{new} = z_i^{new} \pm bw \times rand\ () \tag{9}$$

with Probability *PAR*. The arbitrary distance *rand* () between 0 and 1 and bandwidth is expressed as *bw*.

Step 4: The Harmony Memory is updated. In terms of objective function value, if the new improved harmony vector is better than the worst harmony vector, then the new harmony vector easily supersedes the worst harmony vector.

Step 5: The stopping criterion is checked. If there is a good satisfaction in the stopping criterion, then the iteration is terminated otherwise the steps 3 and 4 are repeated.

The pseudocode for the simple harmony search algorithm is given in Pseudocode 2 below;

Pseudocode 2:

Parameter and HM initializations
While $t < T_{\max}$ do
  For $i = 1$ to m do
    If *rand* () $\le$ *HMCR* then
      $z_i^{new} = z_i^j$, where $j \sim V(1, 2, \ldots,$ HMS)
      If *rand* () $\le$ *PAR* then
        $z_i^{new} = z_i^{new} \pm bw \times rand\ ()$
      End if
    Else
      $z_i^{new} = Qz_j + (Vz_i - Qz_i) \times rand\ ()$
      End if
  End for
  HM Updation
End while

D. CHI SQUARE CDF OPTIMIZATION

If $Q_1, \ldots, Q_k$ are independent, standard random normal variables, then the sum of their squares is expressed as:

$$H = \sum_{i=1}^{l} Q_i^2 \tag{10}$$

It is distributed according to the Chi square distribution with l degree of freedom. This is expressed as:

$$H \sim \chi^2(l) \text{ or } H \sim \chi_l^2 \tag{11}$$

Only one parameter is found in Chi square distribution: l, a positive integer that denotes the total number of degree of freedom [24]. The PDF of the chi square distribution is expressed as:

$$f(x; l) = \left\{ \begin{array}{cc} \frac{x^{\frac{l}{2}-1} e^{-\frac{x}{2}}}{2^{\frac{l}{2}} \Gamma\left(\frac{l}{2}\right)}, & x > 0 \\ 0, & otherwise \end{array} \right\} \tag{12}$$

where $\Gamma\left(\frac{l}{2}\right)$ expresses the gamma function, where the integer l has closed-form values.

For Chi square CDF;

$$F(x; l) = \frac{\gamma\left(\frac{l}{2}, \frac{x}{2}\right)}{\Gamma\left(\frac{l}{2}\right)} = P\left(\frac{l}{2}, \frac{x}{2}\right) \tag{13}$$

where $\gamma(w, t)$ denotes the lower incomplete gamma function and $P(w, t)$ denotes the regularized gamma function.

If $l = 2$ is assumed as a special case, then the function becomes as represented below:

$$F(x; 2) = 1 - e^{-\frac{x}{2}} \tag{14}$$

For other small even values of l, it gives easy way for its computation because of the integer recurrence in the gamma function. If $q = x/l$, then the Chernoff bounds for both lower and upper tails of the CDF is traced. For cases where $0 < q < 1$, then:

$$F(ql, l) \leq \left(qe^{1-q}\right)^{l/2} \tag{15}$$

When $q > 1$, the tail bound for such cases is expressed as:

$$1 - F(ql; l) \leq \left(qe^{1-q}\right)^{l/2} \tag{16}$$

When the Chernoff bounds for both lower and upper tails of the CDF is traced, then the optimization is achieved easily.

E. ELEPHANT SEARCH OPTIMIZATION

It is a famous heuristic search optimization algorithm. The characteristics and behavior of an elephant is mimicked by this algorithm and its implementation is dependent on dual search mechanism [25]. Generally, elephants thrive in groups and an elephant group is divided into several sub-groups under the dynamic leadership of the oldest elephant in the entire group. The characteristics of this elephant head is mimicked by Elephant Search Algorithm (ESA). Elephants generally have a different social structure, the female elephants prefer to live in a family group whereas the male elephants always prefer to live in isolation. The female elephants are responsible for spatial enhancement and the male elephants are responsible for exploration of the target areas. As an effective search optimization algorithm, it has 3 scenarios.

(i) To get the optimal solution, the solution is refined iteratively by the search process.

(ii) The local search at the place is done intensively by the chief female elephant so that there is a high probability of finding the best solution.

(iii) The male elephants have a primary duty to explore out of the local optima area. Thus the elephant's biological behavior is important to determine the features and characteristics but makes the inspiration process quite interesting. The ESA is explained as follows:

As the elephants live together under the dynamic leadership of the oldest elephant, assume the elephant clan is represented as *cla*. The elephant f in the clan *cla* is expressed according to the following equation:

$$Z_{new,cla,f} = Z_{cla,f} + q.\left(Z_{Best,cla} - Z_{cla,f}\right).r \tag{17}$$

where $Z_{new,cla,f}$ are the new updated and old position values for the elephant f in clan *cla* and $q \in [0, 1]$ is a factor which assesses the effect of clan *cla* on $Z_{cla,f}$. $Z_{Best,cla}$ indicates the clan *cla*, and $r \in [0, 1]$. When $Z_{cla,f} = Z_{Best,cla}$, the fittest elephant is expressed by the following equation as:

$$Z_{new,cla,f} = \alpha.Z_{centre,cla} \tag{18}$$

where $\alpha \in [0, 1]$ indicates the effect of the $Z_{centre,cla}$ on $Z_{new,cla,f}$. Then for the new individual $Z_{new,cla,f}$, the l^{th} dimension is updated as $Z_{center,cla,l} = \frac{1}{m_{cla}}.\sum_{f=1}^{m_{cla}} Z_{cla,f,l}$, where $1 \leq l \leq L$ represents the l^{th} dimension and L is its total dimension. m_{cla} is the total number of elephants in the clan *cla*, and $Z_{cla,f,l}$ is the l^{th} dimension of the elephant individual $Z_{cla,f}$. As the male elephants leave this group to stay in isolation, this situation can be simulated by means of separating operator to solve the complicated optimization problems. To enhance the search ability of this algorithm, the individual elephant with a poor fitness case shall implement the separating operator as follows:

$$Z_{worst,cla} = Z_{Min} + (Z_{Max} - Z_{Min} + 1).Rand. \tag{19}$$

where $Z_{\max}$ and $Z_{\min}$ represent the upper bound and lower bound of the elephants individual position. $Z_{worst,cla}$ indicate the worst individual elephant in class *cla*. *Rand* $\in [0, 1]$ is a stochastic distribution. Therefore, clan updation and separating operator implementation is very important in ESA algorithm as it aids in the optimization process.

F. PARTICLE SWARM OPTIMIZATION

The PSO uses some basic parameters and definition of the optimization process and then initiate it with an initial random population called particles [26]. For the main problem, each of these particles has a possible solution and is processed as a part in q-dimensional space. Here in our work, every particle has a specific position traced by

$z_j^k = \left(z_{j1}^k, z_{j2}^k, \ldots, z_{jq}^k\right)$ and a velocity traced by $v_j^k = \left(v_{j1}^k, v_{j2}^k, \ldots, v_{jq}^k\right)$.

The velocity of each particle is updated as:

$$v_j^{k+1} = w_j v_j^k + c_1 r_1 \left(pbest_j - z_j^k\right) + c_2 r_2 \left(gbest_j - z_j^k\right) \tag{20}$$

where r_1 and r_2 are the random selection in the range [0,1] and c_1 and c_2 are the acceleration coefficients that inspect the motion of particles.

The position of each particle is expressed as:

$$z_j^{k+1} = z_j^k + v_j^{k+1} \tag{21}$$

If a particle has a best position, then it is progressed to the next. The representation of the best position is expressed as *pbest* and the representation of the best position of all particles is expressed as *gbest*.

The weight function is expressed as:

$$w_j = w_{\max} - \frac{w_{\max} - w_{\min}}{k_{\max}} \times k \tag{22}$$

The pseudocode is given as follows.

Pseudocode 3:

Load objective function $f(z)$

Initial population and velocity generation of q particles

Global best finding at $k = 0$

While $\left(\left\| \min f(k+1) - \min f(k) \right\| \right) \leq tolerance\ or\ k >$ max *number of iterations*)

Calculate new velocity and position of each particle

Fitness evaluation

If $f\left(z_j(k+1)\right) < f\left(pbest_j(k)\right)$

$pbest_j(k+1) = z_j(k+1)$

Global best Updation

Weight Updation

$k = k + 1$

End

G. CHICKEN SWARM OPTIMIZATION

The behavior of the chicken swarm and its hierarchical order is mimicked by this Chicken Swarm Optimization Algorithm (CSOA) [27]. By various groups the chicken swarm can be explained. Every group consists of only one rooster and many hens and chicks. Between the various chickens in the swarm, there is a competition in a required hierarchical order. The social life of the chickens such as structure of the flock which includes hens, chicks, mother hens is determined by the hierarchical order. The chicken swarm behavior differs greatly with male and female. The food is searched positively by the head rooster and it fights with the neighboring chicks who are around the search space of the group. The chicken that looks out actively for food will be consistent and on par with the head rooster while the other chicken which are submissive in nature shall also search for food in the same location. Therefore, in this swarm, a heavy competition prevails between the chickens and therefore chicks seek their food around their mother. The mathematical model of CSOA for our work is expressed as follows. This swarm has several groups such as rooster, hens and chicks. The chickens which has the several best fitness values are assumed to be roosters, while the chickens with worst fitness values are assigned as chicks and the others can be hens which can easily choose the group to live in. Between the hens and chicks, this hierarchical order is updated under several time steps, say T. Therefore, to search the food, the chickens will follow their group mate rooster. The food search by the chicks is done around their hens. Assuming the ZR, ZH, ZC and ZM indicate the number of roosters, hens, chicks other hens respectively. The best ZR chickens are assumed to be roosters while the worst ZC ones will be indicated as chicks. Assume $\left(A_{f,g}^q, f \in [1, 2, \ldots, Z], g \in [1, 2, \ldots, Y]\right)$ indicate all the Z virtual chickens at time step q. The rooster which have the best fitness value shall access to food easily than others who have the worst fitness value. This can be mathematically modelled as:

$$A_{f,g}^{q+1} = A_{f,g}^q . \left(1 + rand\left(0, \alpha^2\right)\right) \tag{23}$$

$$\alpha^2 = \left\{ \begin{array}{ll} 1, & if\ b_f \leq b_c \\ \exp\left(\dfrac{(b_c - b_f)}{|b_f| + \varepsilon}\right), & c \in [1, Z], c \neq f \end{array} \right\} \tag{24}$$

where $\left(0, \alpha^2\right)$ is a standard Gaussian distribution where 0 explains the mean and α^2 indicates the standard deviation, ε explains a minor value which is not to be divided by zero. From the rooster groups, the random selection of the rooster is done and it is expressed by c which is nothing but the index of rooster. For each corresponding 'α', the fitness value is represented by 'b'. In this chicken swarm, the hens are led by the group mate roosters to search this food. This situation is modelled as:

$$A_{f,g}^{q+1} = A_{f,g}^q + T1.rand . \left(A_{r1,g}^q - A_{f,g}^q\right) + T2.rand . \left(A_{r2,g}^q - A_{f,g}^q\right) \tag{25}$$

$$T_1 = \exp \frac{(b_1 - b_{r1})}{abs\left(b_f\right)} + \varepsilon \tag{26}$$

$$T_2 = \exp\left(b_{r2} - b_f\right) \tag{27}$$

where *rand* denotes a random value in the interval [0, 1], $r1$ and $r2 \in [1, 2, \ldots, Z]$ are rooster index and chicken index respectively, $r1 \neq r2$. Therefore $T2 < 1 < T1$ is possible only if $b_f > b_{r1}, b_f > b_{r2}$. Assuming that T1=0, then the f^{th} hen shall not move around her head rooster and shall search for food by itself. If T2=0, then the f^{th} hen shall seek for food in its surrounding environment. Because of the heavy competition in the group, the value of T1 and T2 should be completely different. Then the movement of the chicks and their mother to search for food is modelled as:

$$A_{f,g}^{q+1} = A_{f,g}^q + KL . \left(A_{m,g}^q - A_f^q, g\right) \tag{28}$$

where $A_{m,g}^q$ indicates the position of the f^{th} chick's mother and $m \in [1, Z]$. KL is a parameter and it belongs to 0,

indicating that the chick follows its mother to obtain the food. Therefore the two important parameters in this algorithm are, initially the total number of iterations F and secondly the updation of the relationship interval G. F and G is set based on the clustering problem. If F is small, then finding a local optimum is easy by this algorithm aiding in the optimization and if F is very large, then it cannot converge to the global optimum quickly in a conducive manner.

H. CAT SWARM OPTIMIZATION

The behavior of the cats can be imitated by an optimization algorithm called cat swarm optimization [28]. To find the optimal solution to some applications it has been used. During the resting period of the cats, the seeking mode is implemented and during the alert period, the training mode is implemented which corresponds to the local search technique in order to obtain the optimal solution of a given problem.

1) MODE OF SEEKING

This behavior comprises of 4 factors such as i)finding the Memory Pool (MP) which aids to define the pool size of the seeking memory, ii)then finding the selected range dimensions (SRD) is done which helps to define the minima and maxima values of the range, iii) finding the total number of dimension counts and finding the supposed to change parameters which indicates the dimension number and finally iv) Self position consideration which is nothing but a Boolean variable. To make sure that the cats spend their valuable time in observation and resting periods, a small term called Mixture Ratio (MR) is used. The procedure is explained as follows:

(a) For finding cats, the MR is selected randomly as a fraction of population n_q.
(b) Implement the memory pool copies for the j^{th} cat
(c) The position of each copy as an approximation of the SRD fractions of the present value is done randomly and then replace it.
(d) The evaluation of the fitness value for all the copies is done.
(e) For all the copies, the probability of each candidate is calculated and then the best one is placed at the position of the j^{th} finding cat.
(f) To make sure that all the cats are involved, the step (b) of this procedure is repeated:

2) MODE OF TRACING

In the optimization process, the tracing mode is considered as an exploration technique. With a high energy, the cat can trace the intended target. By changing its position, the chase of the cat can be modelled in a mathematical form easily. Therefore, in the D-dimensional space, the position and velocity of the j^{th} cat is defined by:

$$Q_j = \left(Q_{j1}, Q_{j2}, \ldots, Q_{jD}\right) \text{ and} \tag{29}$$

$$W_j = \left(W_{j1}, W_{j2}, \ldots, W_{jD}\right) \text{ and } 1 \leq d \leq D \tag{30}$$

TABLE 1. Permutation entropy values.

Sl.No	Types of Optimized values	M=3		M=4	
		Normal	CVD	Normal	CVD
1	Chi square PDF	0.8451	0.8218	0.998	0.8404
2	Chi square CDF	0.9899	0.9592	1.2477	1.2107
3	Harmonic Search	1.8618	1.9038	2.5034	2.4921
4	Elephant Search	2.7143	2.7031	3.0843	3.0545
5	PSO	2.919	2.897	3.1678	3.1703
6	Chicken Swarm	2.9383	2.9484	3.1637	3.1698
7	Cat Swarm	2.3486	2.1224	2.8472	1.0973
8	Density Peak	1.7535	1.7135	2.4356	2.1441

For a cat swarm, the best position Z_{gbq} can be explained as follows:

$$Z_{gbq} = \left(Z_{gbq1}, Z_{gbq2}, \ldots, Z_{gbqD}\right) \tag{31}$$

The steps involved in the tracing mode are presented as follows:

(a) For the j^{th} cat, the new velocity is calculated by the following mathematical form as:

$$W_{jd} = jy.W_{jd} + ac.rn.\left(Z_{gbqd} - Z_{jd}\right) \tag{32}$$

where jy represents the inertia weight, the acceleration constant is expressed as ac and rn is a random number selected in the interval [0,1]. The global best Z_{gbq} is selected in a random fashion from the external archive.
(b) For the j^{th} cat, the updated position is evaluated by the following equation and is represented as $Z_{jd} = Z_{jd} + W_{jd}$
(c) The respective boundary value is modelled to be a new dimension.
(d) The fitness of each cat is evaluated.
(e) With the help of the position of the cats, the contents of the archive can be updated easily thereby achieving the dimensionally reduced values.

The complexity of a dynamic system can be quantified by a robust time series tool called Permutation Entropy [29]. The order relations between the time series values and the probability distribution extraction of the ordinal patterns is captured by the permutation entropy. The Permutation Entropy for all the dimensionality reduction values through optimization is computed in Table 1.

Normal value for Permutation Entropy is always greater than CVD value of Permutation Entropy. Here M is dimension and embedded time lag t=1 is assumed. A famous way of inferring information from cross-variance matrices is by means of using Canonical Correlation Analysis (CCA). If there are two vectors of random variables such as $A = (A_1, .., A_m)$ and $B = (B_1, \ldots, B_m)$ there is a significant correlation among the variables and therefore the linear combinations of A and B which has the maximum correlation with each other is found out by CCA. Thus, for investigating the relationship between two sets of variables, CCA is used.

TABLE 2. Canonical correlation analysis values for all the optimized values among Normal/CVD.

Sl.no	Chi square PDF	Chi square CDF	Harmonic Search	Elephant Search	PSO	Chicken Swarm	Cat Swarm	Density Peak
1	0.9012	0.9132	0.4526	0.4065	0.2605	0.2528	0.9516	0.9142
2	0.8518	0.8533	0.3953	0.2622	0.2243	0.2136	0.9212	0.8868
3	0.7160	0.7961	0.3875	0.2060	0.2025	0.1830	0.8751	0.8463
4	0.6564	0.7751	0.3809	0.1834	0.1842	0.1718	0.7291	0.7959
5	0.5744	0.7360	0.3186	0.1779	0.1660	0.1496	0.7151	0.7588
6	0.4859	0.6090	0.3180	0.1486	0.1332	0.1311	0.6835	0.6714
7	0.4162	0.4851	0.2983	0.1334	0.1213	0.1239	0.4743	0.6119
8	0.3606	0.3961	0.2744	0.1205	0.1106	0.1019	0.4274	0.5796
9	0.3219	0.3358	0.2362	0.0969	0.0892	0.0854	0.3740	0.5031
10	0.2540	0.3095	0.2054	0.0740	0.0729	0.0572	0.2821	0.4958
11	0.1728	0.2588	0.1737	0.0614	0.0586	0.0402	0.2017	0.3357
12	0.1351	0.1804	0.1221	0.0586	0.0487	0.0338	0.1722	0.2813
13	0.1247	0.1659	0.1011	0.0417	0.0276	0.0238	0.0787	0.2356
14	0.0592	0.0707	0.0456	0.0169	0.0179	0.0211	0.0293	0.1397
15	0.0143	0.0084	0.0168	0.0020	0.0018	0.0096	0.0165	0.0814
Average	**0.403**	**0.4596**	**0.2484**	**0.1327**	**0.1146**	**0.1066**	**0.4621**	**0.5425**

TABLE 3. Average canonical correlation analysis among Normal/CVD.

Sl.No	Types of Optimization	Average CCA
1	Chi square PDF	0.403
2	Chi square CDF	0.4596
3	Harmonic Search	0.2484
4	Elephant Search	0.1327
5	PSO	0.1146
6	Chicken Swarm	0.1066
7	Cat Swarm	0.4621
8	Density Peak	0.5425

The CCA for all the optimized values is expressed in Table 2. The Average CCA value for all the optimized values is given in Table 3.

For a single patient, 1,50,000 (1000 x 150) data is reduced to 15,000 only through the process of dimensionality reduction (1000 x 15). Table 2 shows the value of CCA with 15 column values which are transposed. The CCA value of 1 is always perfect whereas correlation zero implies that there is no correlation. CCA value of 0.4 to 0.5 leads to overlapping nonlinear clusters with lesser correlation among two classes. The histogram plot is nonlinear and is evident from Figures below. Fig. 4 shows the histogram plot of Density optimized values for CVD person. Fig. 5 shows the histogram plot of Density Peak optimized values for Normal person and it proves that the Histogram is normal and Gaussian in nature. Fig. 6 shows the Scatter plot for Chi square PDF normal and CVD patient. It is observed in the scatter plot that

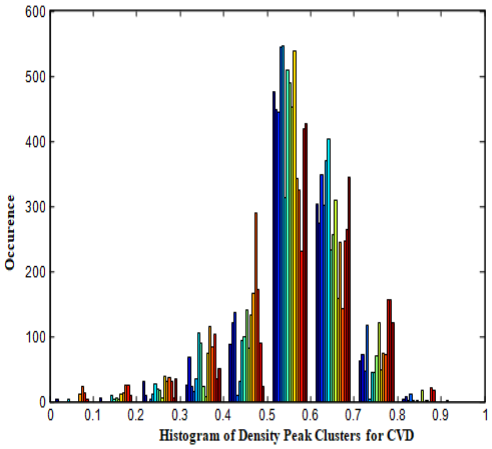

FIGURE 4. Histogram Analysis of Density Peak optimized values for CVD person.

both classes are overlapped at two distinct corners, which is evident in the close correlation of first three optimized values in Table 2. Fig. 7 elucidates the Scatter plot for Chi square CDF normal and CVD patient. Fig. 8 shows the Scatter plot for cat swarm optimized values and it indicates the lesser overlapping among the classes. Fig. 9 explains the Histogram of PSO optimized values for normal cases. Fig. 10 shows the Histogram of PSO optimized values for CVD. Both the Fig. 9 and Fig. 10 shows that the optimized values are centered or converged towards the centroid of the optimized values.

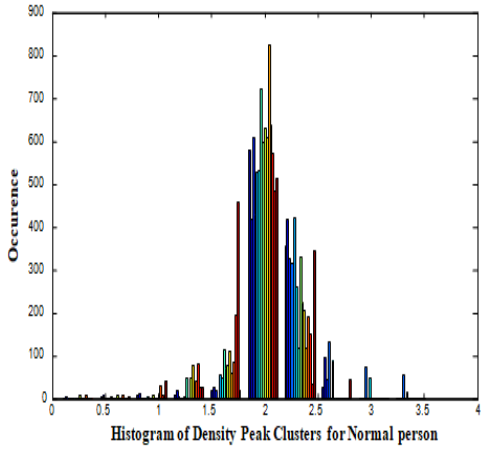

FIGURE 5. Histogram Analysis of Density Peak optimized values for Normal person.

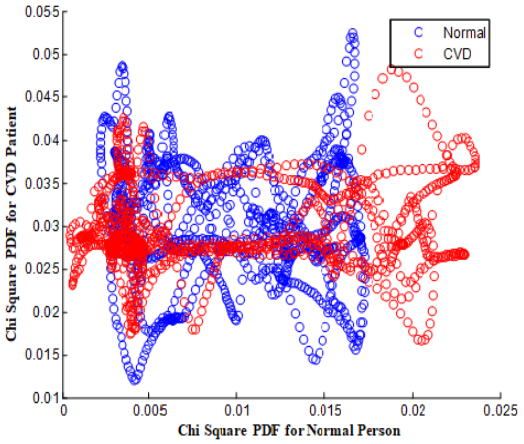

FIGURE 6. Scatter plot for Chi square PDF normal and CVD patient.

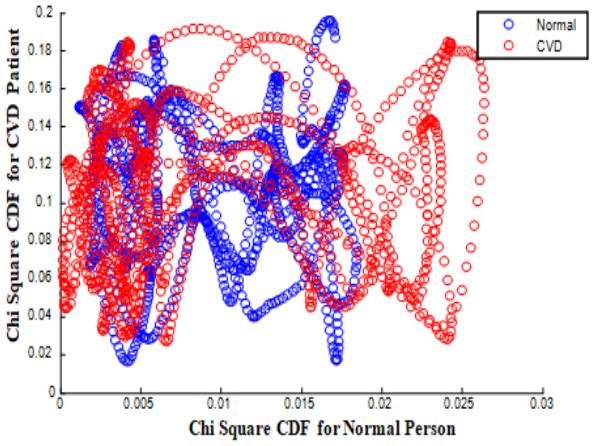

FIGURE 7. Scatter plot for Chi square CDF normal and CVD patient.

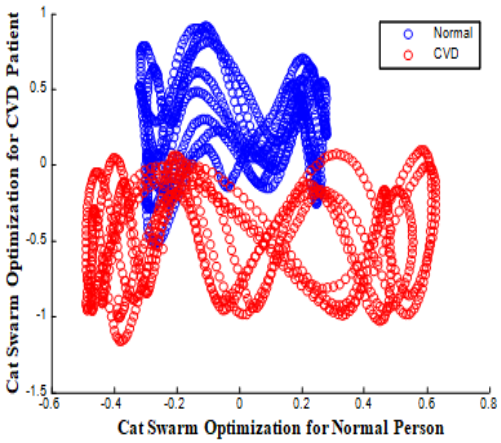

FIGURE 8. Scatter plot for Cat swarm optimized values scatter plot.

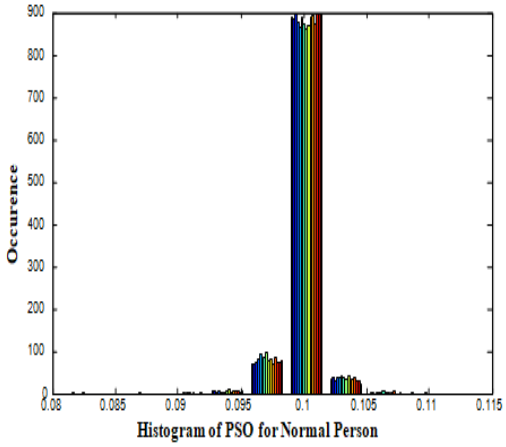

FIGURE 9. Histogram of PSO optimized values normal.

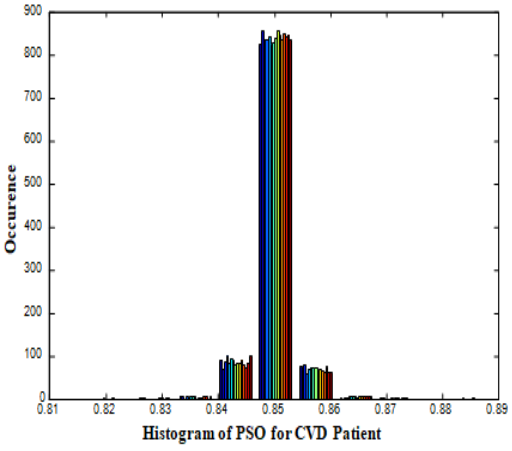

FIGURE 10. Histogram of PSO optimized values CVD.

IV. CLASSIFICATION TECHNIQUES

The dimensionally reduced values are then classified with the classifiers as discussed in the upcoming subsections.

A. LOGISTIC REGRESSION

One of the most commonly used classifiers is Logistic Regression. For a binary classification problem, to future estimate the value of a predictive variable z when $z \in [0, 1]$, 0 implies negative class and 1 implies positive class. To predict the value of z, it also utilizes multi classification when $z \in [0, 1, 2, 3]$.

To classify two classes 0 and 1, a hypothesis $h(\theta) = \theta^T W$ is designed. The classifier output threshold value of $h\theta(w)$ is 0.5. If the value of hypothesis $h\theta(w) \geq 0.5$, then it shows that

$z = 1$ implying that the person has Cardiovascular disease and if $h\theta(w) < 0.5$, then $z = 0$ implying that the person is healthy. Therefore, under the condition $0 \leq h\theta(w) \leq 1$, the prediction of Logistic Regression is done [30]. The Logistic Regression function is expressed as:

$$h\theta(w) = q\left(\theta^T W\right) \tag{33}$$

where $q(v) = \frac{1}{(1+w^{-v})}$ and $h\theta(w) = \frac{1}{(1+w^{-v})}$

The logistic regression cost function $J(\theta)$ is expressed as follows:

$$J(\theta) = \frac{1}{n}\sum_{j=1}^{n} \cos t\left(h\theta\left(w^{(j)}\right), z^{(j)}\right) \tag{34}$$

B. SUPPORT VECTOR MACHINES

One of the most commonly machine learning algorithm is SVM. A maximum margin strategy was used by SVM that helps to transform it into solving a complex quadratic programming issue [31]. Various applications are utilized by SVM. In a binary classification problem, a hyperplane $v^T y + c = 0$ is used to separate the instances, where v and c are the dimensional coefficient vectors which are normal to the hyperplane of the surface. The offset value from the origin is c and y denotes the data set values. By solving Lagrange on multipliers in the linear case, the SVM gets solved and results of v and $c.v$ are obtained. The support vectors are the cases which calls the data points on borders. The solution of v is expressed as $v = \sum_{j=1}^{m} \alpha_j z_j y_j$, where m denotes the number of support vectors and z_j are the target labels to y. The value of v and c are calculated and the linear discriminant function is expressed as follows:

$$h(y) = sgn\left(\sum_{j=1}^{m} \alpha_j z_j y_j^T y + c\right) \tag{35}$$

The non-linear scenario for the decision function and kernel trick is expressed as follows:

$$h(y) = sgn\left(\sum_{j=1}^{m} \alpha_j z_j K\left(y_j, y\right) + c\right) \tag{36}$$

Mercer's condition projected as kernel function is obeyed by positive semi-definite functions. Here in this work, Linear, Polynomial and Gaussian Kernel are utilized for classification.

C. K-NEAREST NEIGHBOUR

One of the supervised learning classification algorithms is KNN. The class label of a new input can be easily predicted by KNN algorithm [32]. In KNN, the similarity between the input sample in the training set and the new input is computed. If the samples in the training set are the same as the new input, then the KNN Classification performance is not good. Assume (a, b) be the training observation and the learning function $q : A \rightarrow B$, so that if an observation a is given, then $q(a)$ can determine the b value.

D. ARTIFICIAL NEURAL NETWORKS

It is again a supervised Machine Learning algorithm and a mathematical model which aids in integrating neurons so that messages are passed easily [33]. The three vital components present here such as input, output and transfer functions. The extraordinary values are taken by the input unit and it is modified during the training process. The calculation for the known class is computed by the output of the ANN. Between the output of a predicted and actual class, the error margin is assessed and thus the weights are recomputed. By the simple integration of neurons, ANN can be designed easily. The architecture of ANN is (256-64-2) with sigmoid training function and MSE as the stopping criteria.

E. Naïve Bayesian CLASSIFIER

It is a famously used classification algorithm. To assess the class of a novel feature vector, NBC is used which is actually based on conditional probability [34]. The training dataset is used by the Naïve Bayesian to trace the conditional probability value of vectors for a particular class. For each vector the probability condition value of each vector is computed and then based on its conditional probability theme, the new vector classes are computed.

F. DECISION TREE CLASSIFIER

It is again a supervised machine learning algorithm. The shape of a decision tree is like a tree where each node is a leaf node. Analysis in deciding for classification is quite easy with the help of decision trees [35]. The internal and external nodes of the decision trees are linked with each other. The internal nodes aid in the decision-making process and make the child node to progress to the next node. On the other hand, leaf node has no child node and is associated with a label.

G. GAUSSIAN MIXTURE MODEL

It is a famous unsupervised classification methodology for recognizing patterns which is actually dependent on assimilating the similar objects together [36]. Here the similar items are classified using clustering technique and then only the ratio of those items in the same category is used to compute and predict the unrated items. As clustering is divided into soft and hard clustering techniques, GMM comes under a soft clustering technique. There the assumption is that the GMM obeys the Gaussian mixture distribution. From various Gaussian Distributions, the data is assumed to be generated here. Every GMM comprises of k Gaussian distributions. Each distribution is termed as a component and all such components are added in a linear manner together for the computation of the probability density function of GMM. For a random vector a in n-dimensional sample space χ, if 'a'

obeys Gaussian distribution then the pdf is expressed as:

$$p(a) = \frac{1}{(2\pi)^{n/2}\,|\Sigma|^{1/2}} e^{-\left(1/2\right)(a-\mu)^T \Sigma^{-1}(a-\mu)} \quad (37)$$

where a n-dimensional mean vector is represented by μ and the covariance matrix of $n \times n$ is represented as Σ. Only with the covariance matrix Σ and the mean vector μ, the determination of the Gaussian distribution is done. There are totally k mixing components in Gaussian mixture distribution each of which corresponds to a gaussian distribution. The definition of a mixture distribution is done as follows:

$$p_Q(a) = \sum_{i=1}^{k} \alpha_j . p\left(a|\, \mu_j, \Sigma_j\right) \quad (38)$$

where μ_j and Σ_j are the parameters of the j^{th} Gaussian mixture and α_j represents the corresponding mixing coefficient.

H. EXTREME LEARNING MACHINE

A new learning technique for the simple hidden layer feed forward neural network was proposed by Huang *et. al.* and it is called as ELM [37]. To overcome overfitting problems and the slow training speed of the conventional neural networks, it was progressed. The general idea is as follows:

For the Q training samples, the output of a SLFN network with H hidden nodes is written as follows:

$$g_H(y_k) = \sum_{m}^{H} \beta_m l\left(w_m . y_k + b_m\right), \quad k = 1, 2, 3, \ldots, Q \quad (39)$$

It can be expressed as $g(y) = h(y)\beta$, where y_k represents the input training vector, w_m represents the input weight and b_m represent the biases to the hidden layer. The m^{th} hidden node to the output layer and $l(.)$ representing the hidden nodes activation function is linked by the output weights β_m. Obtaining a least-square solution by implementing the Moore-Penrose generalized inverse is used to training a SLFN and is represented as:

$$\widehat{\beta} = \left(H'H\right)^{-1} H'T \text{ or} \quad (40)$$

$$\widehat{\beta} = H'(HH')^{-1}T \quad (41)$$

where it depends on the similarity of $H'H$ or HH'.

When calculating the output weights β_m, if $H'H$ is not singular then the coefficient $1/\varepsilon$ is added to the diagonal of $H'H$.

Therefore, a better generalization performance with a good stable learning system is obtained. The output function of ELM is written as follows:

$$g(y) = h(y)H'\left(\frac{1}{\varepsilon} + HH'\right)^{-1} T \quad (42)$$

The mappings of the hidden layer features are not known to the users and in this ELM kernel implementation, Gaussian kernel was used. The value for the positive regularization coefficient (ε) is chosen as 1 and the parameters of Gaussian kernel was chosen to be 5 in this classification experiment. In ELM, the hidden neurons are 30 in number and output neuron is just one and the threshold function used here is Sigmoid / radial basis function.

V. RESULTS AND DISCUSSION

A. VALIDATION METHOD OF CLASSIFIERS

K-fold cross-validation technique was used in this paper. In this type, the dataset is split into 'k' equal size points initially. Then to train the classifiers, $k-1$ groups are used and for checking performance in every step, the remaining part is utilized. For k number of times, the validation is repeated. Based on k results, the computation of the classifier performance is done. Different values of k are selected and here $k = 10$. In the 10-fold CV method, 90% of the data was used for training and 10% for testing. For each fold of process, the procedure was repeated 10 times. Over the whole dataset, the random division of all the instances in the training and testing groups was done before selection of the new sets of testing and training for another cycle. At the end of 10-fold process, average of all performance metrics is computed. The performance metrics parameters are as follows.

TP: The predicted output as true positive which implies that the CVD subject is correctly classified and therefore the subjects have heart disease.

TN: The healthy subject is classified correctly and so the subject is healthy.

FP: The predicted output as false positive, healthy subject is incorrectly classified as they do not have the risk of CVD.

FN: False negative, CVD is incorrectly classified and that the subject does not have heart disease as the subject is healthy.

The classification accuracy which depicts the overall performance of the classification system is expressed as:

$$Classification\ Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \times 100 \quad (43)$$

The classification Error is nothing but the overall incorrect classification of the classification model and is calculated as:

$$Classification\ Error = \frac{FP + FN}{TP + TN + FP + FN} \times 100 \quad (44)$$

The sensitivity or the true positive rate is nothing but the implication of diagnostic test is positive and the subject has disease and is expressed as:

$$Sensitivity = \frac{TP}{TP + TN} \times 100 \quad (45)$$

The specificity is the implication that the diagnostic test is negative and the person is healthy and is expressed as:

$$Specificity = \frac{TN}{TN + FP} \times 100 \quad (46)$$

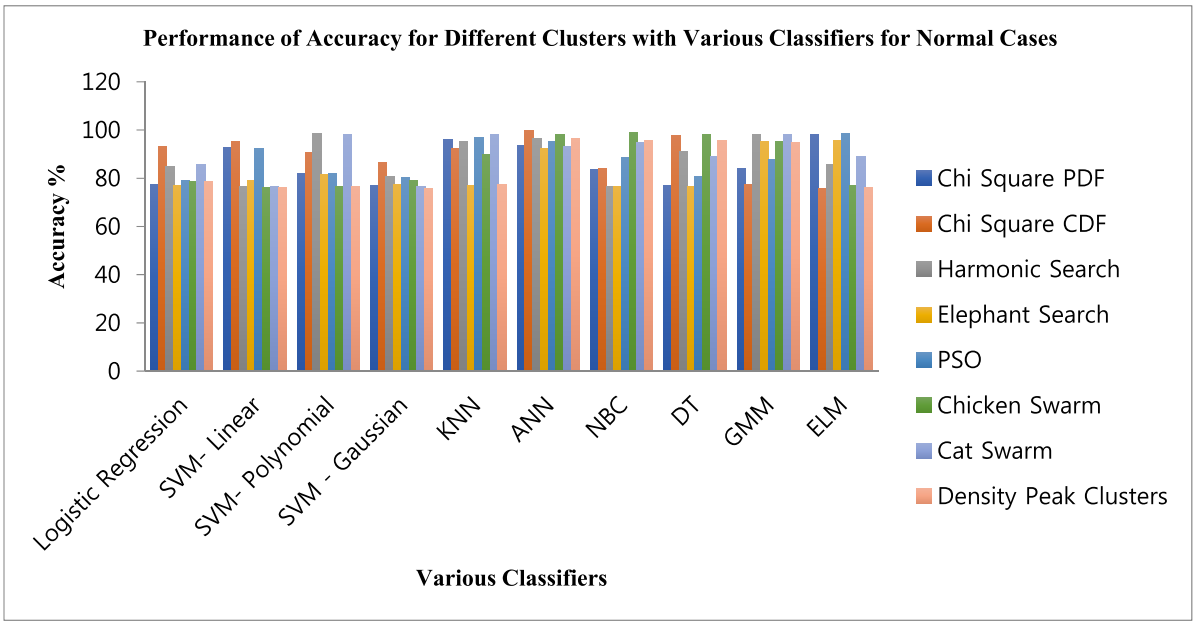

FIGURE 11. Performance of Accuracy for different optimization techniques in various classifiers for normal PPG signal.

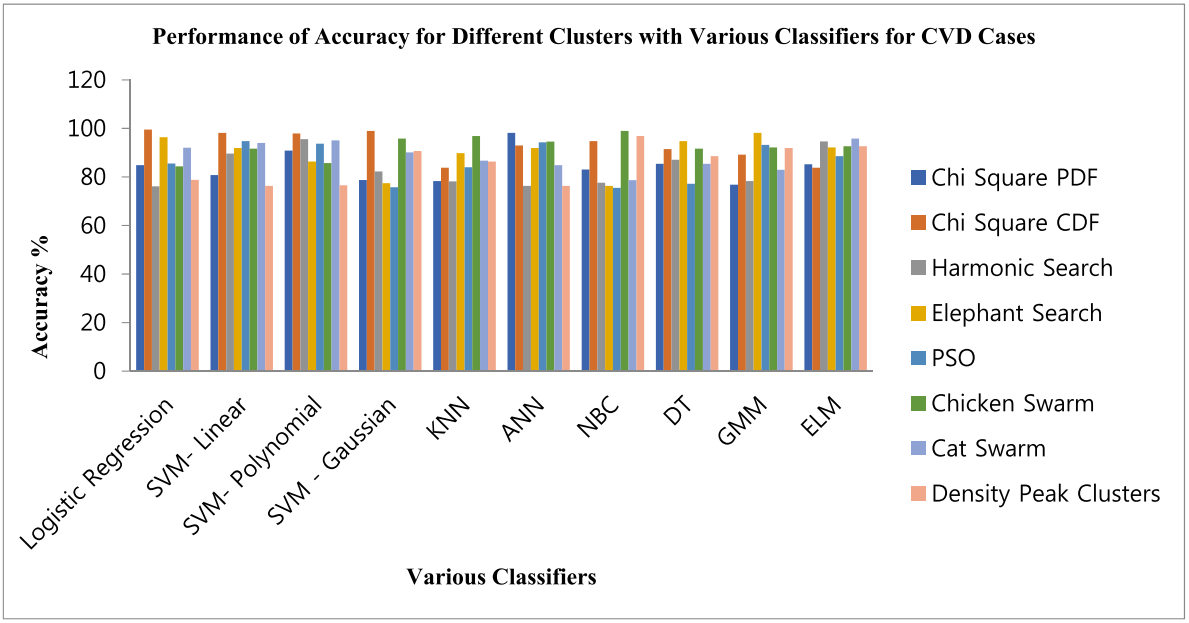

FIGURE 12. Performance of Accuracy for different optimized values in various classifiers for CVD PPG signal.

The Performance Index is expressed as follows:

$$PI = \left(\frac{(TP + TN) - FN - FP}{(TP + TN)} \right) \times 100 \tag{47}$$

Good Detection Rate (GDR): Being a vital criteria of a detector it signifies the successful detection and is mathematically expressed as:

$$GDR = \left(\frac{[(TP + TN) - FN]}{[(TP + TN) + FP]} \right) \times 100 \tag{48}$$

In Tables 4 to 19, the Perfect Classification (PC) is indicated as the sum of TP and TN. FN is represented by the Missed Classification (MC) and FP is represented by the False Alarm (FA).

The Mean Square Error (MSE) is mathematically expressed as follows:

$$MSE = \frac{1}{N} \sum_{i=1}^{N} (O_i - T_j)^2 \tag{49}$$

where O_i denotes the observed value at a specific time, T_j denotes the target value at model j; $j = 1$ to 15, and N

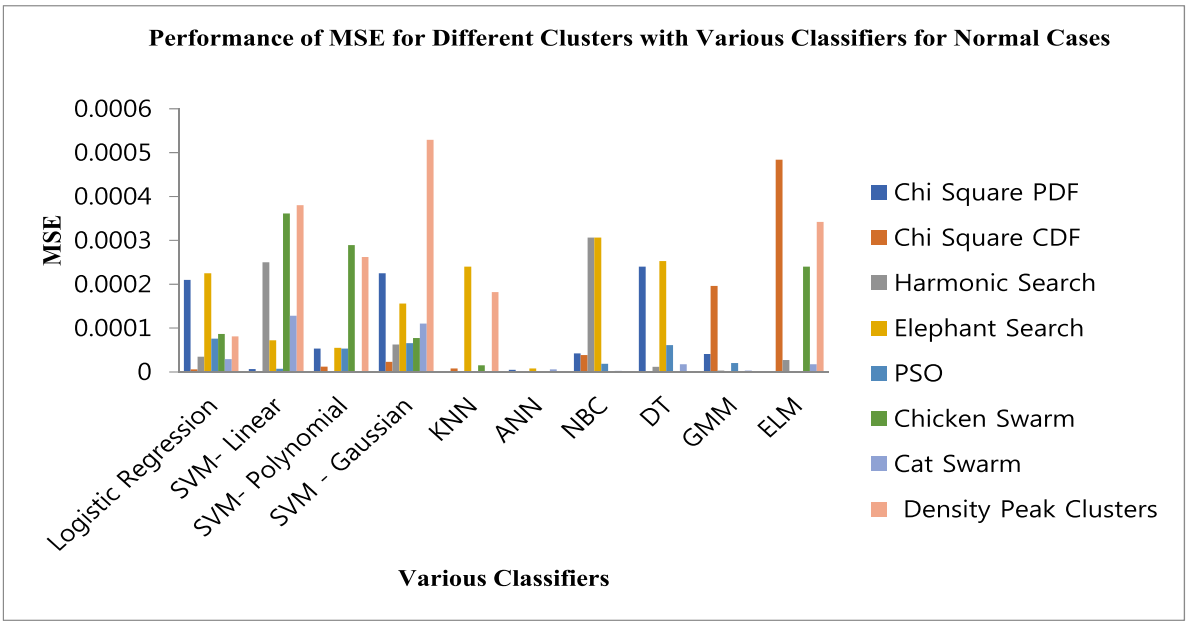

FIGURE 13. Performance of MSE for different optimized values in various classifiers for Normal PPG signal.

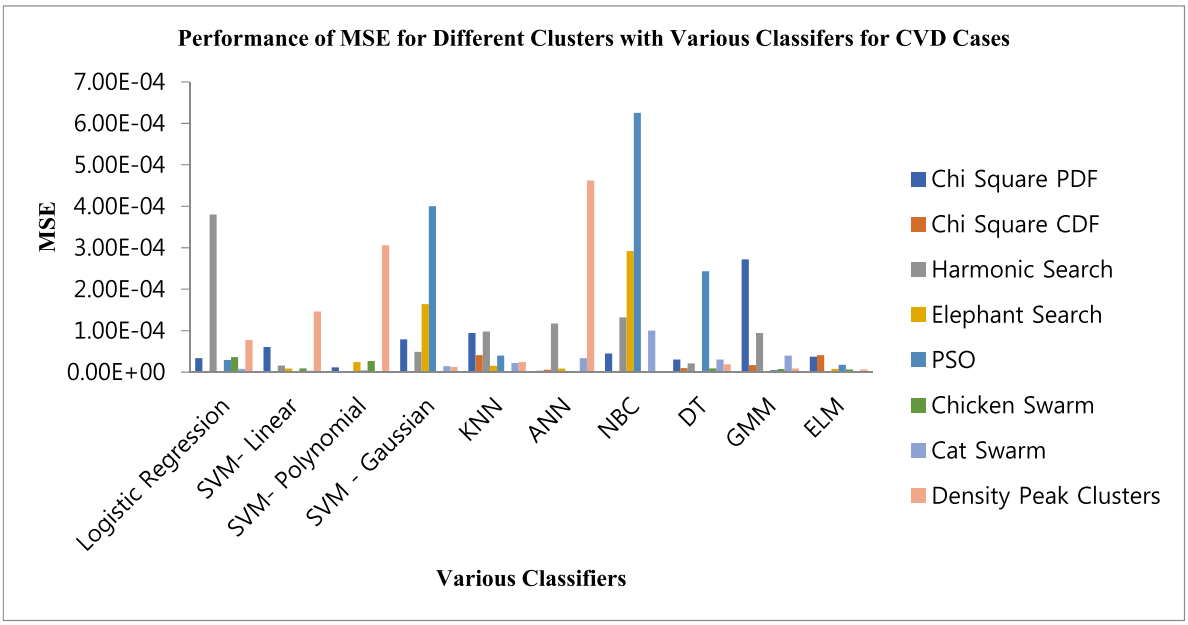

FIGURE 14. Performance of MSE for different optimized values in various classifiers for CVD PPG signal.

is the total number of observations per patient in our case, it is 1000. This research utilizes the dimensionally reduced values of EEG data both for training and testing classifiers. In a regressive manner, the training was progressed and the classifier MSE Values was mitigated to an utmost minimum level. With a zero-training error of MSE, all the classifiers were trained.

The predictability is expressed as:

$$Predictability = \left\{1 - \left(\frac{(Accuracy - GDR)}{2 \times (Accuracy + GDR)}\right)\right\} \times 100 \tag{50}$$

Table 4 shows the consolidated result analysis of Chi square PDF optimized values with the different classifiers for the normal PPG cases. Table 5 shows the consolidated result analysis of density peak optimized values with the different classifiers for the normal PPG cases. It is accessed from Table 4 and Table 5 there is no Missed classification among the classifiers. Table 6 shows the consolidated result analysis of harmonic search optimized values with the different classifiers for the normal PPG cases. As shown in the Table 6 the harmonic search is plugged in to more missed classification among the classifiers and hence the performance Index of the classifier is at the lower ebb

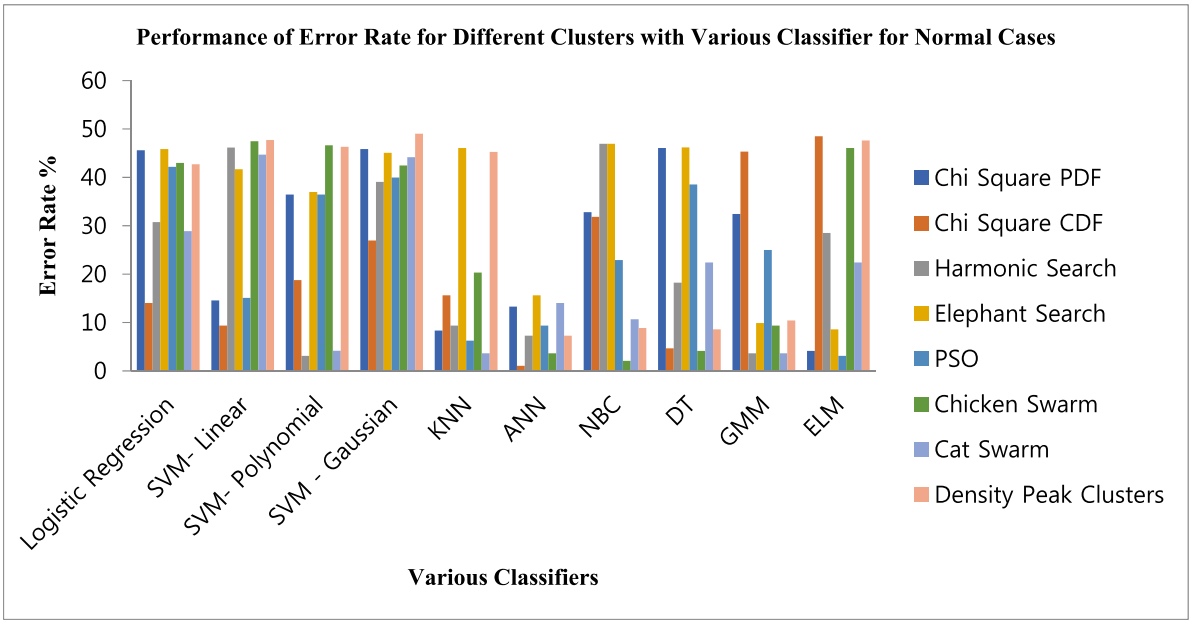

FIGURE 15. Performance of Error Rate for different optimized values in various classifiers for Normal PPG signal.

criteria. Table 7 explains the consolidated result analysis of Chi square CDF with the different classifiers for the normal PPG cases. Table 8 explains the consolidated result analysis of Elephant search optimized values with the different classifiers for the normal PPG cases. Table 9 explains the consolidated result analysis of Particle swarm optimized values with the different classifiers for the normal PPG cases. Table 10 explains the consolidated result analysis of chicken swarm optimized values with the different classifiers for the normal PPG cases. Table 11 explains the consolidated result analysis of cat swarm optimized values with the different classifiers for the normal PPG cases. Table 12 shows the consolidated result analysis of Chi square PDF optimized values with the different classifiers for the abnormal PPG cases. Table 13 shows the consolidated result analysis of density peak optimized values with the different classifiers for the abnormal PPG cases. Table 14 shows the consolidated result analysis of harmonic search optimized values with the different classifiers for the abnormal PPG cases. Table 15 explains the consolidated result analysis of Chi square CDF optimized values with the different classifiers for the abnormal PPG cases. Table 16 explains the consolidated result analysis of Elephant search optimized values with the different classifiers for the abnormal PPG cases. Table 17 explains the consolidated result analysis of Particle swarm optimized values with the different classifiers for the abnormal PPG cases. Table 18 explains the consolidated result analysis of chicken swarm optimized values with the different classifiers for the abnormal PPG cases. Table 19 explains the consolidated result analysis of cat swarm optimized values with the different classifiers for the abnormal PPG cases.

Fig. 11 shows the performance of accuracy for different optimized values in various classifiers for normal PPG signal. As expressed in Fig. 11, the ANN classifier scores better average accuracy of 95.52% than any other classifiers among the optimized values. SVM Gaussian Classifier stood at low level average accuracy of 79.02%. Fig. 12 shows the performance of accuracy for different optimized values in various classifiers for PPG CVD signal. ELM scores the best average accuracy of 90.71% among different optimized values for CVD cases and NBC attained low average accuracy value of 85.23% and this is due to more false alarms. Fig. 13 shows the performance of average MSE for different optimized values in various classifiers for normal PPG signal. The lowest MSE of 2.744E-06 is attained for the ANN classifier and SVM Gaussian classifier once again fails to reach the good MSE value, but settles at the MSE of 0.00156 among the optimized values for Normal Cases. Fig. 14 elucidates the performance of average MSE for different optimized values in various classifiers for CVD PPG signal. Once again ELM classifier shuttled to the lowest value of MSE 1.5E-05 and it is the turn of NBC to occupy the high MSE value of 1.51E-04 among the optimized values for CVD patients. Fig. 15 explains the performance of average Error Rate for different optimized values in various classifiers for normal PPG signals. For normal cases under ANN classifier, a 8.94% average error rate is attained. This is due to lower error rate in the optimized values of Chi square CDF optimization and Chicken swarm optimization. SVM Gaussian is placed in the poor performance category as it produced an average error rate of only 41.55%. Fig. 16 explains the performance of Error Rate for different optimized values in various classifiers for PPG CVD signals. The cat swarm optimization with error

1) NORMAL CASES - PPG

TABLE 4. Consolidated result analysis of Chi Square PDF optimized values with the different classifiers.

Classifiers	PC(%)	MC(%)	FA(%)	PI(%)	Sensitivity(%)	Specificity(%)	Accuracy (%)
Logistic Regression	54.42125	0	45.57875	16.2175	54.42125	100	77.21063
SVM- Linear	85.42	0	14.58	82.93	85.42	100	92.71
SVM- Polynomial	63.54	0	36.455	42.58	63.54	100	81.77
SVM - Gaussian	54.16	0	45.84	15.36	54.16	100	77.08
KNN	91.67	0	8.33	91.58	91.67	100	95.835
ANN	86.72	0	13.28	84.66125	86.72	100	93.36
NBC	67.188	0	32.82	51.163	67.188	100	83.594
DT	53.9575	0	46.0425	14.64094	53.9575	100	76.97875
GMM	67.5785	0	32.4275	52.00725	67.5785	100	83.78925
ELM	95.83	0	4.16	95.65	95.83	100	97.915

TABLE 5. Consolidated result analysis of density peak optimized values with the different classifiers.

Classifiers	PC(%)	MC(%)	FA(%)	PI(%)	Sensitivity(%)	Specificity(%)	Accuracy (%)
Logistic Regression	57.295	0	42.705	25.405	57.295	100	78.6475
SVM- Linear	52.27	0	47.73	8.64875	52.27	100	76.135
SVM- Polynomial	53.6875	0	46.3125	13.68219	53.6875	100	76.3375
SVM - Gaussian	51	0	49	3.92	51	100	75.5
KNN	54.74781	0	45.25219	17.28938	54.74781	100	77.37391
ANN	92.71	0	7.29	92.455	92.71	100	96.355
NBC	91.15	0	8.85	90.78	91.15	100	95.575
DT	91.41	0	8.59	91.18	91.41	100	95.705
GMM	89.59	0	10.41	88.38	89.59	100	94.795
ELM	52.405	0	47.595	9.128125	52.405	100	76.2025

TABLE 6. Consolidated result analysis of harmonic search optimized values with the different classifiers.

Classifiers	PC(%)	MC(%)	FA(%)	PI(%)	Sensitivity(%)	Specificity(%)	Accuracy (%)
Logistic Regression	69.2725	0	30.729	55.6125	69.2725	100	84.63625
SVM- Linear	53.83094	0	46.16906	14.19152	53.83094	100	76.3375
SVM- Polynomial	96.875	0	3.12	96.76	96.875	100	98.4375
SVM - Gaussian	60.94	0	39.06	35.72125	60.94	100	80.47
KNN	90.63	9.37	0	89.98	100	90.63	95.315
ANN	92.71	7.29	0	92.455	100	92.71	96.355
NBC	53.08	0	46.92	11.525	53.08	100	76.54
DT	81.7725	18.2275	0	77.6925	100	81.7725	90.88625
GMM	96.3525	0	3.64	96.205	96.3525	100	98.17625
ELM	71.48688	28.51538	0	60.105	100	71.48688	85.74344

TABLE 7. Consolidated result analysis of Chi Square CDF optimized values with the different classifiers.

Classifiers	PC(%)	MC(%)	FA(%)	PI(%)	Sensitivity(%)	Specificity(%)	Accuracy (%)
Logistic Regression	85.94	0	14.06	83.6225	85.94	100	92.97
SVM- Linear	90.63	9.37	0	89.98	100	90.63	95.315
SVM- Polynomial	81.25	18.75	0	76.92	100	81.25	90.625
SVM - Gaussian	73.04688	26.95313	0	63.04125	100	73.04688	86.52344
KNN	84.38	0	15.62	81.4325	84.38	100	92.19
ANN	98.96	0	1.04	98.935	98.96	100	99.48
NBC	68.16425	31.83875	0	53.27363	100	68.16425	84.08213
DT	95.31	4.6825	0	95.07	100	95.31	97.655
GMM	54.6825	0	45.3175	17.075	54.6825	100	77.34125
ELM	51.5	0	48.5	5.805	51.5	100	75.75

TABLE 8. Consolidated result analysis of elephant search optimized values with the different classifiers.

Classifiers	PC(%)	MC(%)	FA(%)	PI(%)	Sensitivity(%)	Specificity(%)	Accuracy (%)
Logistic Regression	54.16	45.84	0	15.36	100	54.16	77.08
SVM- Linear	58.34	0	41.66	28.59	58.34	100	79.17
SVM- Polynomial	63.02	0	36.9775	41.29	63.02	100	81.51
SVM - Gaussian	54.94375	0	45.05625	17.9325	54.94375	100	77.47188
KNN	53.9575	0	46.0425	14.64094	53.9575	100	76.97875
ANN	84.38	0	15.62	81.4325	84.38	100	92.19
NBC	53.08	0	46.92	11.525	53.08	100	76.54
DT	53.80563	0	46.19438	14.10164	53.80563	100	76.3375
GMM	90.11	0	9.89	89.18	90.11	100	95.055
ELM	91.41	0	8.59	91.18	91.41	100	95.705

TABLE 9. Consolidated result analysis of particle swarm optimized values with the different classifiers.

Classifiers	PC(%)	MC(%)	FA(%)	PI(%)	Sensitivity(%)	Specificity(%)	Accuracy (%)
Logistic Regression	57.8175	0	42.1825	26.9975	57.8175	100	78.90875
SVM- Linear	84.9	0	15.1	82.18125	84.9	100	92.45
SVM- Polynomial	63.54	0	36.455	42.58	63.54	100	81.77
SVM - Gaussian	60.03	0	39.97	33.22531	60.03	100	80.015
KNN	93.75	0	6.25	93.33	93.75	100	96.875
ANN	90.63	0	9.37	89.98	90.63	100	95.315
NBC	77.085	0	22.915	72.795	77.1	100	88.55
DT	61.46	0	38.54	37.1475	61.46	100	80.73
GMM	75	0	25	66.66	75	100	87.5
ELM	96.875	0	3.12	96.76	96.875	100	98.4375

TABLE 10. Consolidated result analysis of chicken swarm optimized values with the different classifiers.

Classifiers	PC(%)	MC(%)	FA(%)	PI(%)	Sensitivity(%)	Specificity(%)	Accuracy (%)
Logistic Regression	57.03375	0	42.96625	24.60875	57.03375	100	78.51688
SVM- Linear	52.54	0	47.46	9.6075	52.54	100	76.27
SVM- Polynomial	53.35	0	46.65	12.48375	53.35	100	76.675
SVM - Gaussian	57.55625	0	42.44375	26.20125	57.55625	100	78.77813
KNN	79.69	0	20.31	78.4275	79.7125	100	89.85625
ANN	96.3525	0	3.64	96.205	96.3525	100	98.17625
NBC	97.92	0	2.08	97.87	97.92	100	98.96
DT	95.83	0	4.16	95.65	95.83	100	97.915
GMM	90.63	0	9.37	89.98	90.63	100	95.315
ELM	53.9575	0	46.0425	14.64094	53.9575	100	76.97875

TABLE 11. Consolidated result analysis of cat swarm optimized values with the different classifiers.

Classifiers	PC(%)	MC(%)	FA(%)	PI(%)	Sensitivity(%)	Specificity(%)	Accuracy (%)
Logistic Regression	71.09875	0	28.90575	59.34	71.09875	100	85.54938
SVM- Linear	55.30297	0	44.69703	19.11156	55.30297	100	76.3375
SVM- Polynomial	95.83	4.166	0	95.65	100	95.83	97.915
SVM - Gaussian	55.85813	0	44.14188	20.93375	55.85813	100	76.3375
KNN	96.3525	3.64	0	96.205	100	96.3525	98.17625
ANN	85.94	0	14.06	83.6225	85.94	100	92.97
NBC	89.32875	10.67125	0	88.045	100	89.32875	94.66438
DT	77.60625	0	22.39375	74.32875	77.625	100	88.8125
GMM	96.3525	0	3.64	96.205	96.3525	100	98.17625
ELM	77.60625	22.39375	0	74.32875	100	77.625	88.8125

2) ABNORMAL CASES/CVD - PPG

TABLE 12. Consolidated result analysis of Chi Square PDF optimized values with the different classifiers.

Classifiers	PC(%)	MC(%)	FA(%)	PI(%)	Sensitivity(%)	Specificity(%)	Accuracy (%)
Logistic Regression	69.795	0	30.208	56.685	69.795	100	84.8975
SVM- Linear	61.46	38.54	0	37.1475	100	61.46	80.73
SVM- Polynomial	81.7725	18.2275	0	77.6925	100	81.7725	90.88625
SVM - Gaussian	57.42563	42.57438	0	25.80313	100	57.42563	78.71281
KNN	56.57656	0	43.42344	23.21531	56.57656	100	78.28828
ANN	96.3525	0	3.64	96.205	96.3525	100	98.17625
NBC	66.21	33.79125	0	48.91188	100	66.21	83.105
DT	70.84	0	29.166	58.83	70.84	100	85.42
GMM	53.62	0	46.38	13.4425	53.62	100	76.81
ELM	70.44813	0	29.55675	58.02563	70.44813	100	85.22406

TABLE 13. Consolidated result analysis of density peak optimized values with the different classifiers.

Classifiers	PC(%)	MC(%)	FA(%)	PI(%)	Sensitivity(%)	Specificity(%)	Accuracy (%)
Logistic Regression	57.55625	0	42.44375	26.20125	57.55625	100	78.77813
SVM- Linear	55.04172	0	44.95828	18.25406	55.04172	100	76.3375
SVM- Polynomial	53.08	0	46.92	11.525	53.08	100	76.54
SVM - Gaussian	81.25	18.75	0	76.92	100	81.25	90.625
KNN	72.65625	27.34375	0	62.3175	100	72.65625	86.32813
ANN	51.625	0	48.375	6.27625	51.625	100	76.3375
NBC	93.75	6.25	0	93.33	100	93.75	96.875
DT	77.085	22.915	0	72.795	100	77.1	88.55
GMM	83.86	0	16.14	80.72125	83.86	100	91.93
ELM	85.42	14.58	0	82.93	100	85.42	92.71

TABLE 14. Consolidated result analysis of harmonic search optimized values with the different classifiers.

Classifiers	PC(%)	MC(%)	FA(%)	PI(%)	Sensitivity(%)	Specificity(%)	Accuracy (%)
Logistic Regression	52.27	0	47.73	8.64875	52.27	100	76.135
SVM- Linear	79.17	0	20.83	78.93	79.2	100	89.6
SVM- Polynomial	91.15	0	8.85	90.78	91.15	100	95.575
SVM - Gaussian	64.58	0	35.41	45.16	64.58	100	82.29
KNN	56.38063	0	43.61938	22.61813	56.38063	100	78.19031
ANN	55.66219	0	44.33781	20.29063	55.66219	100	76.3375
NBC	55.205	0	44.795	18.79	55.205	100	77.6025
DT	74.21875	0	25.78125	65.2125	74.21875	100	87.10938
GMM	56.57656	0	43.42344	23.21531	56.57656	100	78.28828
ELM	89.32875	0	10.67125	88.045	89.32875	100	94.66438

TABLE 15. Consolidated result analysis of Chi Square CDF optimized values with the different classifiers.

Classifiers	PC(%)	MC(%)	FA(%)	PI(%)	Sensitivity(%)	Specificity(%)	Accuracy (%)
Logistic Regression	98.96	0	1.04	98.935	98.96	100	99.48
SVM- Linear	96.3525	0	3.64	96.205	96.3525	100	98.17625
SVM- Polynomial	95.83	0	4.16	95.65	95.83	100	97.915
SVM - Gaussian	97.92	0	2.08	97.87	97.92	100	98.96
KNN	67.5785	0	32.4275	52.00725	67.5785	100	83.78925
ANN	85.94	14.06	0	83.6225	100	85.94	92.97
NBC	89.59	0	10.41	88.38	89.59	100	94.795
DT	82.94813	0	17.05188	79.43063	82.94813	100	91.47406
GMM	78.38813	0	21.61188	76.62938	78.4125	100	89.20625
ELM	67.5785	32.4275	0	52.00725	100	67.5785	83.78925

TABLE 16. Consolidated result analysis of elephant search optimized values with the different classifiers.

Classifiers	PC(%)	MC(%)	FA(%)	PI(%)	Sensitivity(%)	Specificity(%)	Accuracy (%)
Logistic Regression	92.71	0	7.29	92.455	92.71	100	96.355
SVM- Linear	83.86	16.14	0	80.72125	100	83.86	91.93
SVM- Polynomial	72.65625	27.34375	0	62.3175	100	72.65625	86.32813
SVM - Gaussian	54.87027	45.12973	0	17.69133	100	54.87028	77.43514
KNN	79.69	20.31	0	78.4275	100	79.7125	89.85625
ANN	83.86	0	16.14	80.72125	83.86	100	91.93
NBC	53.2825	0	46.7175	12.24406	53.2825	100	76.3375
DT	89.59	10.41	0	88.38	100	89.59	94.795
GMM	96.3525	3.64	0	96.205	100	96.3525	98.17625
ELM	84.38	15.62	0	81.4325	100	84.38	92.19

TABLE 17. Consolidated result analysis of particle swarm optimized values with the different classifiers.

Classifiers	PC(%)	MC(%)	FA(%)	PI(%)	Sensitivity(%)	Specificity(%)	Accuracy (%)
Logistic Regression	71.09875	0	28.90575	59.34	71.09875	100	85.54938
SVM- Linear	89.59	10.41	0	88.38	100	89.59	94.795
SVM- Polynomial	87.5	12.5	0	85.7	100	87.5	93.75
SVM - Gaussian	51.5	0	48.5	5.805	51.5	100	75.75
KNN	67.969	0	32.035	52.8515	67.969	100	83.9845
ANN	88.545	0	11.455	87.04	88.545	100	94.2725
NBC	51.0625	0	48.9375	4.155625	51.0625	100	75.53125
DT	54.3975	0	45.6025	15.605	54.3975	100	77.19875
GMM	86.46	0	13.54	84.315	86.46	100	93.23
ELM	77.085	0	22.915	72.795	77.1	100	88.55

TABLE 18. Consolidated result analysis of chicken swarm optimized values with the different classifiers.

Classifiers	PC(%)	MC(%)	FA(%)	PI(%)	Sensitivity(%)	Specificity(%)	Accuracy (%)
Logistic Regression	68.75	0	31.25	54.54	68.75	100	84.375
SVM- Linear	83.34	0	16.66	80.01	83.34	100	91.67
SVM- Polynomial	71.3575	0	28.6455	59.85	71.3575	100	85.67875
SVM - Gaussian	91.67	8.33	0	91.58	100	91.67	95.835
KNN	93.75	0	6.25	93.33	93.75	100	96.875
ANN	89.0675	0	10.9325	87.71	89.0675	100	94.53375
NBC	97.92	2.08	0	97.87	100	97.92	98.96
DT	83.34	16.66	0	80.01	100	83.34	91.67
GMM	84.38	0	15.62	81.47	84.38	100	92.19
ELM	85.42	14.58	0	82.93	100	85.42	92.71

TABLE 19. Consolidated result analysis of cat swarm optimized values with the different classifiers.

Classifiers	PC(%)	MC(%)	FA(%)	PI(%)	Sensitivity(%)	Specificity(%)	Accuracy (%)
Logistic Regression	84.12	0	15.88	81.105	84.12	100	92.06
SVM- Linear	88.0225	0	11.9775	86.37	88.0225	100	94.01125
SVM- Polynomial	90.11	0	9.89	89.18	90.11	100	95.055
SVM - Gaussian	80.21	0	19.79	77.925	80.225	100	90.1125
KNN	73.4375	0	26.5625	63.765	73.4375	100	86.71875
ANN	69.795	0	30.208	56.685	69.795	100	84.8975
NBC	57.295	0	42.705	25.405	57.295	100	78.6475
DT	70.84	0	29.166	58.83	70.84	100	85.42
GMM	65.884	0	34.115	48.1615	65.884	100	82.942
ELM	91.67	0	8.33	91.58	91.67	100	95.835

rate of 8.33% makes the ELM classifier better suited for classifying CVD patients for average error rate of 18.58% and NBC average error rate is high with 29.46% among all the optimization techniques. Fig. 17 explains the performance

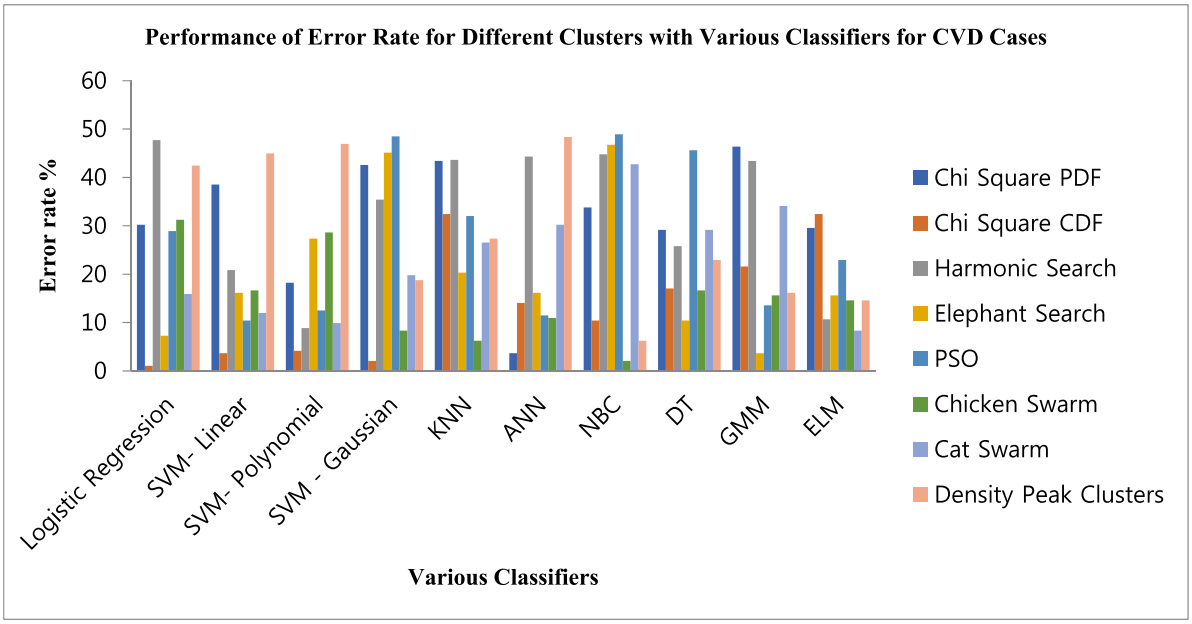

FIGURE 16. Performance of Error Rate for different optimized values in various classifiers for CVD PPG signal.

of PI for different optimized values in various classifiers for Normal PPG signal. Higher average PI of 89.96% is attained by ANN and the low average PI of 27.04% is attained in SVM Gaussian Classifier for Normal PPG signal. Fig. 18 explains the performance of PI for different optimized values in various classifiers for CVD PPG signal. The average PI value of 76.21% in ELM classifier is attained and the edged value of 48.63% is reached in NBC classifier for the CVD patients. Fig. 19 shows the average performance measures for different optimized values in various classifiers for normal PPG signals. The performance measures of all the classifiers among various optimized values in averaged form and the parametric values are obtained with Accuracy of 86.69%, GDR of 72.18%, PI of 57.29%, Error rate of 26.15, and predictability of 94.87% as in the case of normal person. Fig. 20 shows the average performance measures for different optimized values in various classifiers for CVD PPG signals. For the CVD patients the average performance measures of classifiers are averaged and depicted as accuracy of 87.89%, GDR of 73.49%, PI of 63.1%, Error rate of 24.14% and Predictability of 94.86%. Fig. 21 shows the performance of predictability vs accuracy for different optimized values in various classifiers for Normal PPG and CVD PPG signal. As indicated in the Fig. 21, among the classifiers the predictability and accuracy are closely following each other in the Normal and CVD classification. Fig. 22 shows the GDR vs accuracy for different optimized values in various classifiers for normal PPG and CVD PPG signal. As depicted in the Fig. 22, the GDR and accuracy parameter deviates 2.5% in the normal classifiers and in the cases of CVD the deviation among the GDR and Accuracy is ebbed to 1.35%. This is a good indicator for classifier performance. Fig. 23 explains the GDR vs predictability for different optimization techniques in various classifiers for Normal PPG and CVD PPG signal. GDR and predictability follows the same path as shown in the Fig. 22 with deviation of 2.96% in the normal cases and deviation of 1.62% in the CVD cases.

B. DISCUSSION FOR NORMAL PPG SIGNALS

The results show that when Chi square PDF optimized values are analyzed with different classifiers, a high classification accuracy of 97.91% is obtained and a high PI of 95.65% is obtained when classified with ELM. The lowest performance was produced when it was classified with Logistic Regression giving a classification accuracy of about 77.21% and a PI of 16.21%. There were only instances of FA here for all the classifiers.

When the Density peak optimized values are classified with different classifiers, a high classification accuracy of 96.35% along with a PI of 92.45% is obtained when classified with ANN. Similarly, a low performance of 75.5% was obtained if classified with Gaussian SVM.

When the Harmonic search optimized values are classified with different classifiers, a high classification accuracy of 98.43% along with a PI of 96.76% is obtained when classified with Polynomial SVM. Similarly, a low performance of 76.33% was obtained if classified with Linear SVM.

When the Chi square CDF optimized values are classified with different classifiers, a high classification accuracy of 99.48% along with a PI of 98.93% is obtained when classified with ANN. Similarly, a low performance of 75.75% was obtained if classified with ELM.

When the Elephant Search optimized values are classified with different classifiers, a high classification accuracy

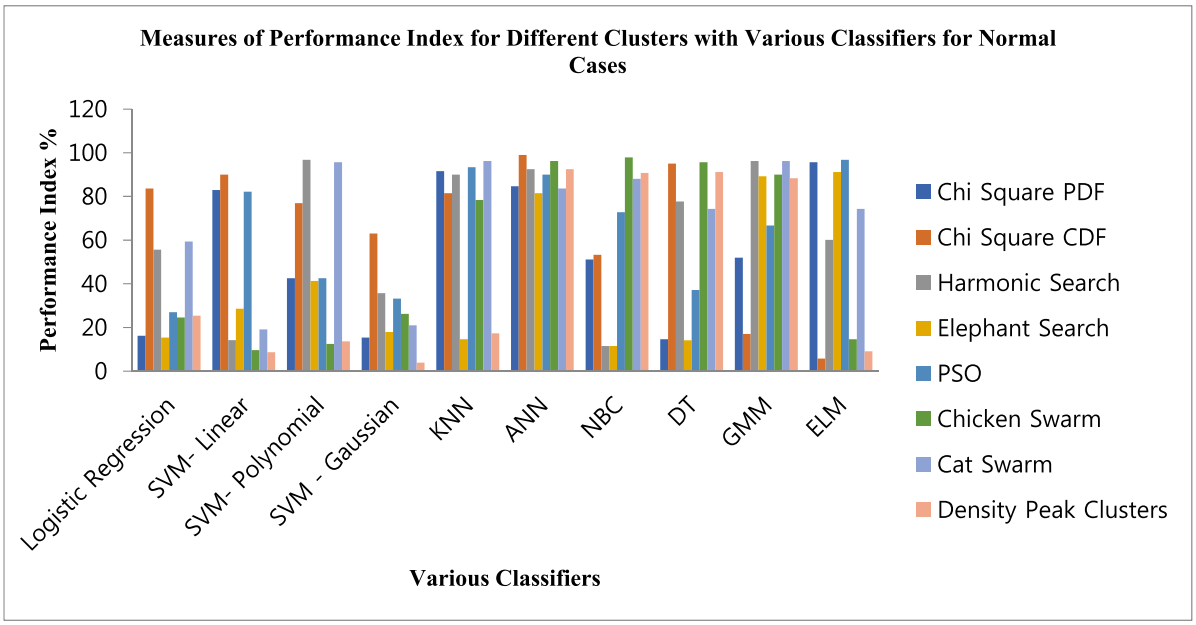

FIGURE 17. Performance of Performance Index for different optimized values in various classifiers for Normal PPG signal.

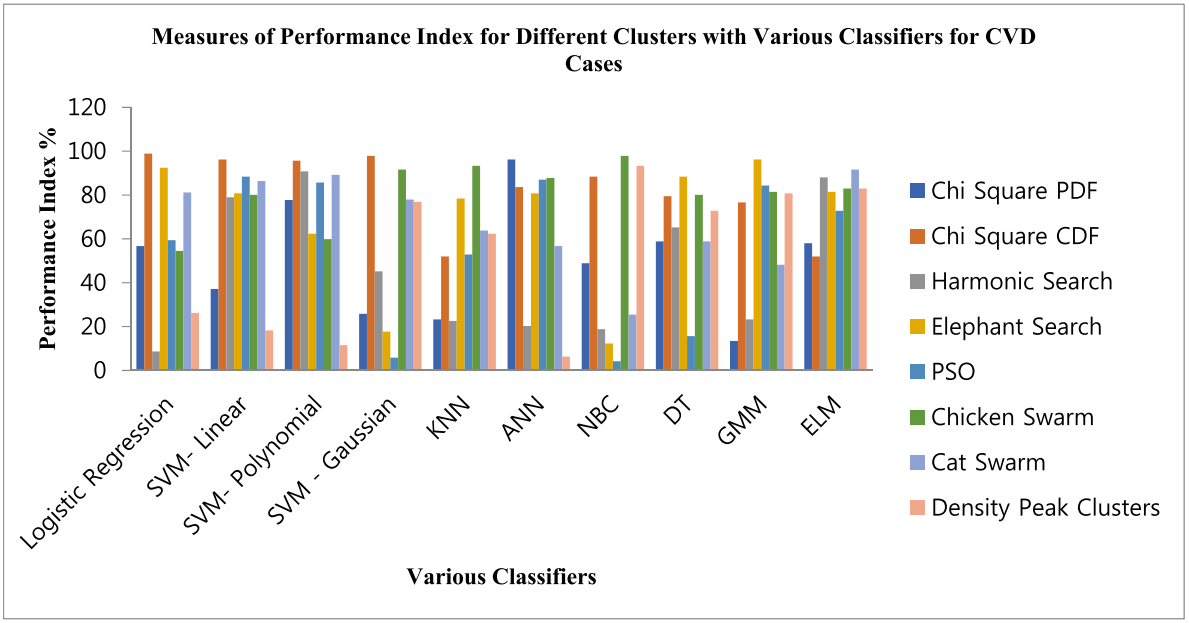

FIGURE 18. Performance of Performance Index for different optimized values in various classifiers for CVD PPG signal.

TABLE 20. Calculation of kappa values.

	Evaluator 2		
		Prediction of X value	Prediction of Y value
Evaluator 1	Prediction of X value	P_{XX}	P_{XY}
	Prediction of Y value	P_{YX}	P_{YY}

of 95.70% along with a PI of 91.18% is obtained when classified with ELM. Similarly, a low performance of 76.33% was obtained if classified with DT.

When the Particle swarm optimized values are classified with different classifiers, a high classification accuracy of 98.43% along with a PI of 96.76% is obtained when classified with ELM. Similarly, a low performance of 78.90% was obtained if classified with Logistic Regression.

When the chicken swarm optimized values are classified with different classifiers, a high classification accuracy of 98.96% along with a PI of 97.87% is obtained when classified with NBC. Similarly, a low performance of 76.27% was obtained if classified with Linear SVM.

When the cat swarm optimized values are classified with different classifiers, a high classification accuracy of 98.17%

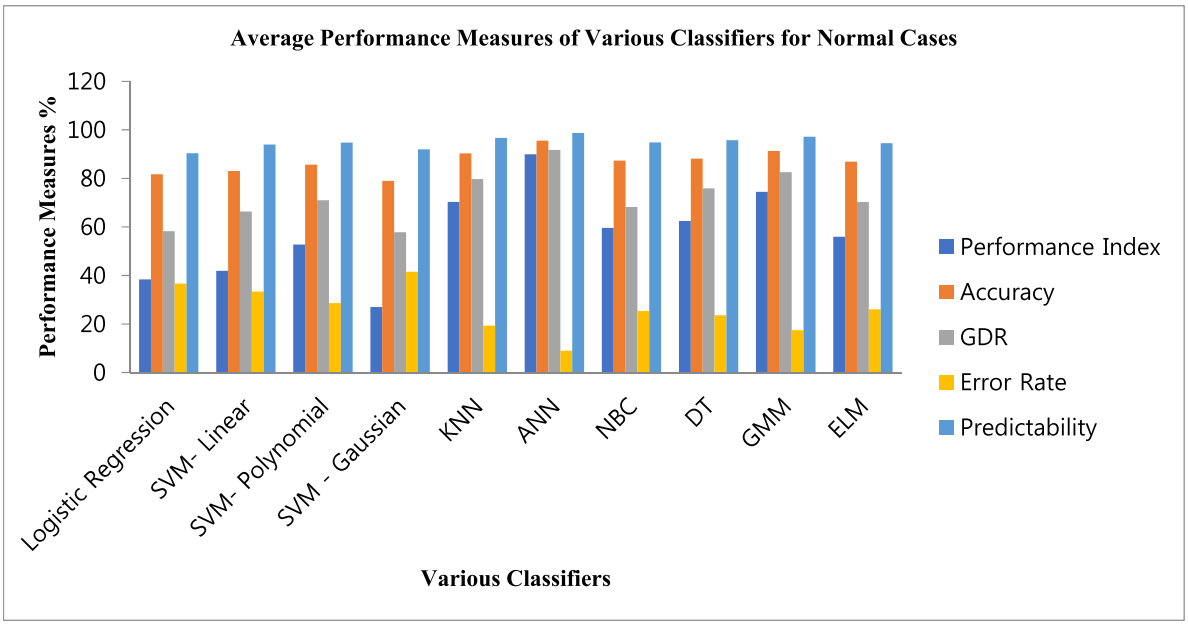

FIGURE 19. Performance Measures of different optimized values in various classifiers for Normal PPG signal.

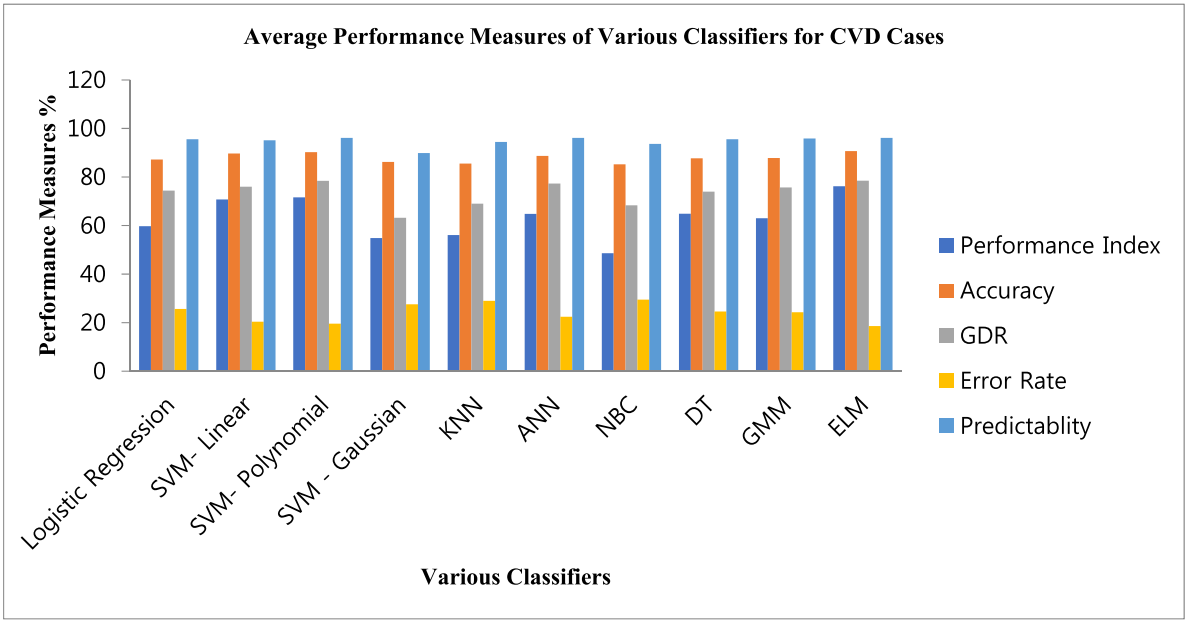

FIGURE 20. Performance Measures of different optimized values in various classifiers for CVD PPG signal.

along with a PI of 96.20% is obtained when classified with both GMM and KNN. Similarly, a low performance of 76.33% was obtained if classified with Linear SVM and Gaussian SVM.

C. DISCUSSION FOR CVD PPG SIGNALS

When the chi square PDF optimized values are classified with different classifiers, a high classification accuracy of 98.17 % along with a PI of 96.20% is obtained when classified with ANN. Similarly, a low performance of 76.81% was obtained if classified with GMM.

When the density peak optimized values are classified with different classifiers, a high classification accuracy of 96.87 % along with a PI of 93.33% is obtained when classified with NBC. Similarly, a low performance of 76.33% was obtained if classified with Linear SVM and ANN.

When the harmonic search optimized values are classified with different classifiers, a high classification accuracy of 95.57 % along with a PI of 90.78% is obtained when classified with Polynomial SVM. Similarly, a low performance of 76.13% was obtained if classified with Logistic Regression.

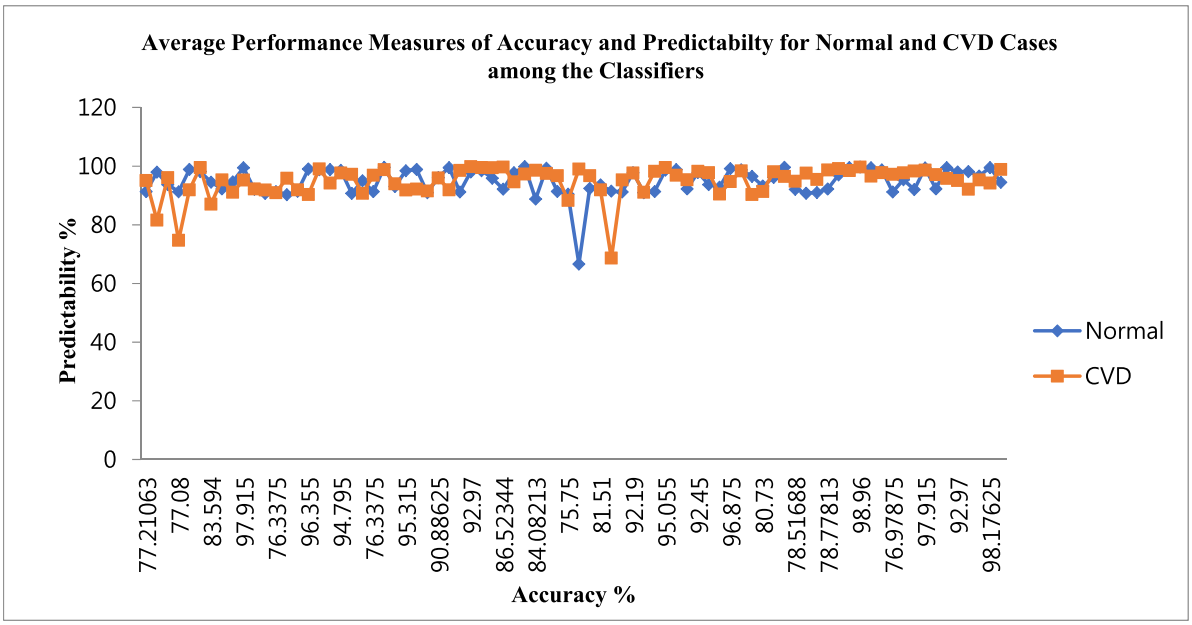

FIGURE 21. Predictability Vs Accuracy Chart for Normal and CVD patients.

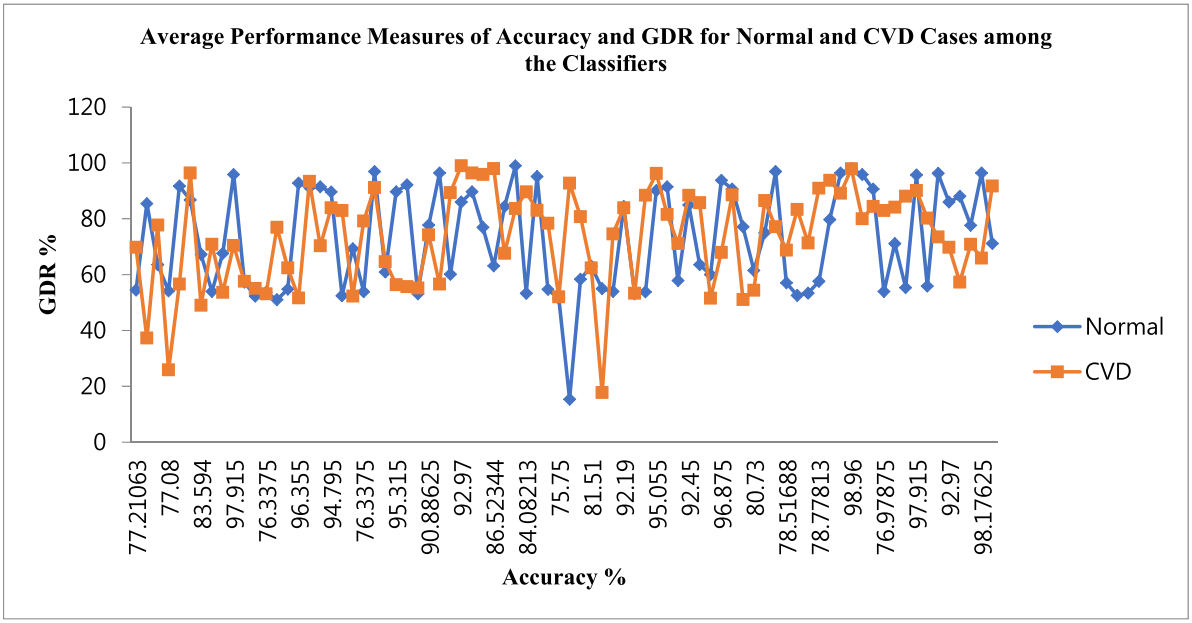

FIGURE 22. GDR Vs Accuracy for Normal and CVD patient.

When the Chi square CDF optimized values are classified with different classifiers, a high classification accuracy of 99.48% along with a PI of 98.93% is obtained when classified with Logistic Regression. Similarly, a low performance of 83.78% was obtained if classified with ELM and KNN.

When the elephant search optimized values are classified with different classifiers, a high classification accuracy of 98.17% along with a PI of 96.205% is obtained when classified with GMM. Similarly, a low performance of 76.33% was obtained if classified with NBC.

When the particle swarm optimized values are classified with different classifiers, a high classification accuracy of 94.79% along with a PI of 88.38% is obtained when classified with Linear SVM. Similarly, a low performance of 75.5% was obtained if classified with NBC.

When the chicken swarm optimized values are classified with different classifiers, a high classification accuracy of 98.96% along with a PI of 97.87% is obtained when classified with NBC. Similarly, a low performance of 84.37% was obtained if classified with Logistic Regression.

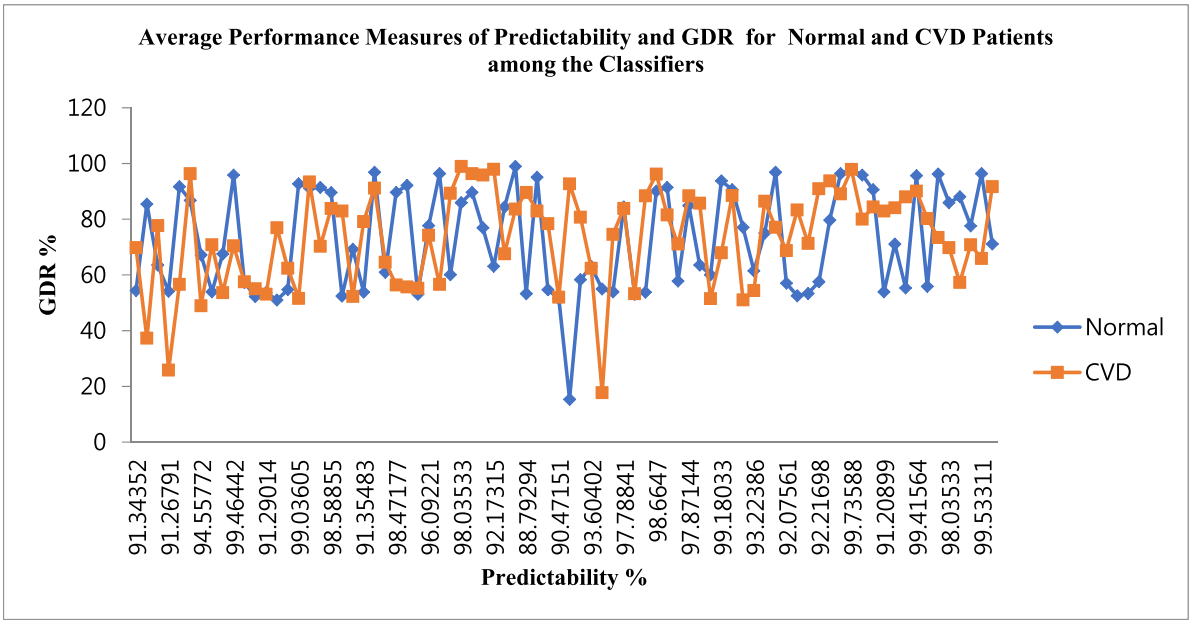

FIGURE 23. GDR Vs Predictability for Normal and CVD patient.

TABLE 21. Cohen's Kappa coefficients for all the classifiers among different clusters.

Classifiers	Clusters							
	Chi Square PDF	Chi Square CDF	Harmonic search	Elephant Search	PSO	Chicken Swarm	Cat Swarm	Density Peak clusters
Logistic Regression	0.821	0.943	0.784	0.974	0.835	0.84	0.917	0.716
SVM- Linear	0.81	0.872	0.926	0.958	0.961	0.904	0.957	0.725
SVM- Polynomial	0.916	0.985	0.947	0.853	0.946	0.861	0.961	0.775
SVM - Gaussian	0.762	0.992	0.831	0.762	0.741	0.964	0.873	0.895
KNN	0.795	0.852	0.773	0.886	0.821	0.955	0.842	0.841
ANN	0.979	0.915	0.743	0.905	0.952	0.951	0.836	0.741
NBC	0.825	0.947	0.762	0.741	0.748	0.986	0.762	0.958
DT	0.841	0.904	0.882	0.953	0.762	0.925	0.846	0.875
GMM	0.746	0.882	0.775	0.979	0.913	0.935	0.806	0.907
ELM	0.864	0.843	0.935	0.918	0.862	0.918	0.947	0.918

When the cat swarm optimized values are classified with different classifiers, a high classification accuracy of 95.83 % along with a PI of 91.58% is obtained when classified with ELM. Similarly, a low performance of 82.94% was obtained if classified with GMM.

D. KAPPA ANALYSIS MEASURES

Cohen's Kappa coefficient is a statistical method that measures both inter-rater reliability and intra-rater reliability between two evaluators. In our study, two raters were, a CVD and normal PPG. Kappa determines if there is an agreement between two evaluators by chance with a percentage. Table 20 shows cell probabilities for two evaluators. The predictions X and Y represent the CVD and normal PPG respectively. In other words, these predictions can be obtained from the confusion matrix.

To compute Kappa, we have to calculate the observed level of agreement (P_o) and expected agreement (P_e) which is defined as:

$$P_0 = \frac{(P_{XX} + P_{XY})}{P_T} \tag{51}$$

$$P_e = \left(\left[\left(\frac{P_{XX} + P_{XY}}{P_T}\right) \times \left(\frac{P_{XX} + P_{YX}}{P_T}\right)\right] + \left[\left(\frac{P_{XY} + P_{YY}}{P_T}\right) \times \left(\frac{P_{YX} + P_{YY}}{P_T}\right)\right]\right) \tag{52}$$

and

The total level of agreement P_T is expressed as:

$$P_T = P_{XX} + P_{XY} + P_{YX} + P_{YY} \tag{53}$$

The Kappa coefficient is calculated as

$$K = \frac{(P_o - P_e)}{(1 - P_e)} \tag{54}$$

The kappa value is less than or equal to 1. The Kappa coefficient of value 1 indicates perfect agreement among the evaluators. Any value less than 1 will be categorized as poor

TABLE 22. Computation complexity for all the classifiers among different clusters.

Classifiers	Clusters							
	Chi Square PDF	Chi Square CDF	Harmonic search	Elephant Search	PSO	Chicken Swarm	Cat Swarm	Density Peak clusters
Logistic Regression	$O(n^2)$	$O(n^2)$	$O(n\log n)$	$O(n\log n)$	$O(n\log n)$	$O(n\log n)$	$O(n\log n)$	$O(n^2)$
SVM-Linear	$O(n^2)$	$O(n^2)$	$O(n\log n)$	$O(n\log n)$	$O(n\log n)$	$O(n\log n)$	$O(n\log n)$	$O(n^2)$
SVM-Polynomial	$O(n^3)$	$O(n^3)$	$O(n^2\log n)$	$O(n^2\log n)$	$O(n^2\log n)$	$O(n^2\log n)$	$O(n^2\log n)$	$O(n^3)$
SVM - Gaussian	$O(n\log n)$	$O(n\log n)$	$O(n\log 2n)$	$O(n\log 2n)$	$O(n\log 2n)$	$O(n\log 2n)$	$O(n\log 2n)$	$O(n\log n)$
KNN	$O(n^3)$	$O(n^3)$	$O(n^2\log n)$	$O(n^2\log n)$	$O(n^2\log n)$	$O(n^2\log n)$	$O(n^2\log n)$	$O(n^3)$
ANN	$O(n^5)$	$O(n^5)$	$O(n^4\log n)$	$O(n^4\log n)$	$O(n^4\log n)$	$O(n^4\log n)$	$O(n^4\log n)$	$O(n^5)$
NBC	$O(n^2)$	$O(n^2)$	$O(n\log n)$	$O(n\log n)$	$O(n\log n)$	$O(n\log n)$	$O(n\log n)$	$O(n^2)$
DT	$O(n\log n)$	$O(n\log n)$	$O(n\log 2n)$	$O(n\log 2n)$	$O(n\log 2n)$	$O(n\log 2n)$	$O(n\log 2n)$	$O(n\log n)$
GMM	$O(n^2)$	$O(n^2)$	$O(n\log n)$	$O(n\log n)$	$O(n\log n)$	$O(n\log n)$	$O(n\log n)$	$O(n^2)$
ELM	$O(n^4)$	$O(n^4)$	$O(n^3\log n)$	$O(n^3\log n)$	$O(n^3\log n)$	$O(n^3\log n)$	$O(n^3\log n)$	$O(n^4)$

TABLE 23. Comparisons with previous works.

Year	Authors	Methodology	Result
2019	Divya *et. al.*[38]	Softmax Discriminant Classifier	97.88%
2019	Divya *et. al.* [38]	Gaussian Mixture Model	96.64%
2018	Hao *et. al.* [39]	Softmax Regression Model	94.44%
2018	*Miao and Miao* [40]	Deep Neural Networks	83.67%
2016	Shobitha *et. al.* [41]	ELM	82.5%
2015	Hosseini *et. al.* [42]	K-nearest neighbour	81.50%
Proposed Method	**Prabhakar *et. al.*** (Normal PPG)	**Chi square PDF + ANN**	**99.48%**
	Prabhakar *et. al.* (Normal PPG)	**Chicken swarm + NBC**	**98.96%**
	Prabhakar *et. al.* (CVD PPG)	**Chi square CDF + Logistic regression**	**99.48%**
	Prabhakar *et. al.* (CVD PPG)	**a) Chi square CDF + Gaussian SVM b) Chicken swarm + NBC**	**98.96%**

agreement (<0.2), fair agreement (0.2-0.4), Good agreement (0.6-0.8) and Very Good agreement (0.8-1). The Cohen's kappa coefficient for all the classifiers among the different clusters is computed and tabulated in the Table 21.

It is observed from the Table 21 that the Kappa coefficients obtained from all the classifiers among different clusters are belonging to either good or very good agreement.

E. COMPUTATIONAL COMPLEXITY OF THE METHODS

Next the classifiers are analyzed with respect to computational complexity. Finding the computational complexity is done according to the size of input (n). Computational complexity is very less if it is equal to $O(1)$. As the number of input increases then the computational complexity will be increased. In this case, the complexity does not depend on input size and this is one of the desired entities for any algorithm. If the computational complexity increases log(n) times with respect to increase in 'n', then it is denoted as $O(\log n)$. In this paper all the classifiers are hybrid in nature, and they are classifying the clustered outputs. Table 22 shows the Computation Complexity for all the classifiers among different clusters. From Table 22, the SVM Gaussian and DT classifiers attained the computational complexity of $O(n\log n)$ and $O(n\log 2n)$ among the eight cluster groups and this is the lowest computational complexity achieved in the paper. Logistic regression, SVM linear classifiers, NBC and GMM classifiers are at the mean computational complexity of $O(n^2)$ and $O(n\log n)$. The classifiers like SVM polynomial and KNN are at the level of moderate computational complexity as $O(n^3)$ and $O(n^2\log n)$. ELM classifier

provides the high computational complexity of $O\left(n^4\right)$ and $O\left(n^3 \log n\right)$. The ANN classifier is at the highest computational complexity of $O\left(n^5\right)$ and $O\left(n^4 \log n\right)$. The reason of high complexity in ELM and ANN classifiers are due to training of various layers of the classifiers.

F. COMPARISON WITH PREVIOUS WORKS

Table 23 exhibits the comparison of our current works with the previous works.

VI. CONCLUSION

The PPG is a signal which results from the local variations of blood volume in tissues. The peripheral pulse wave modulated by respiration and heart activity is reflected by PPG. In this work, the PPG signals obtained from Capnobase dataset are analyzed for classification of cardiovascular diseases using various classifiers. The results show that for the PPG normal cases, a high classification accuracy of 99.48% is obtained when Chi square Probability Density Function (PDF) optimized values are classified with Artificial Neural Networks (ANN) and a second highest classification accuracy of 98.96% is obtained when chicken swarm optimized values are classified with Naïve Bayesian Classifier (NBC). A third highest classification accuracy of 98.43% is obtained when Harmonic search optimized values are classified with Polynomial SVM and Particle swarm optimized values are classified with ELM. Similarly when the PPG abnormal cases or PPG with CVD cases are concerned, a high classification accuracy of 99.48% is obtained when Chi square PDF optimized values is classified with Logistic Regression and a second highest classification accuracy of 98.96% is obtained when Chi square CDF optimized values is classified with Gaussian Support Vector Machine (SVM) and when chicken swarm optimized values are classified with NBC. A third highest classification accuracy of 98.17% is obtained when Chi square PDF optimized values are classified with ANN, Chi square CDF optimized values are classified with Linear SVM and Elephant search optimized values are classified with GMM. Future works aim to work with other metaheuristic algorithms for dimensionality reduction and other machine learning algorithms like deep neural networks and Convolutional neural networks for classification of cardiovascular diseases from PPG signals.

REFERENCES

[1] C. F. Clarenbach, A. Stoewhas, A. J. R. Van Gestel, T. D. Latshang, C. M. L. Cascio, K. E. Bloch, and M. Kohler, "Comparison of photoplethysmographic and arterial tonometry-derived indices of arterial stiffness," *Hypertension Res.*, vol. 35, no. 2, pp. 228–233, 2012.

[2] J. Allen, "Photoplethysmography and its application in clinical physiological measurement," *Physiolog. Meas.*, vol. 28, no. 3, pp. R1–R39, Feb. 2007.

[3] A. A. R. Kamal, J. B. Harness, G. Irving, and A. J. Mearns, "Skin photoplethysmography—A review," *Comput. Methods Programs Biomed.*, vol. 28, no. 4, pp. 257–269, Apr. 1989.

[4] K. Pilt, K. Meigas, R. Ferenets, and J. Kaik, "Photoplethysmographic signal processing using adaptive sum comb filter for pulse delay measurement," *Estonian J. Eng.*, vol. 16, no. 1, pp. 78–94, Mar. 2010.

[5] K. Takazawa N. Tanaka, and M. Fujita, "Assessment of vasoactive agents and vascular aging by the second derivative of photoplethysmogram waveform," *Hypertension*, vol. 32, no. 2, pp. 365–370, 1998.

[6] U. Rubins, "Finger and ear photoplethysmogram waveform analysis by fitting with Gaussians," *Med. Biol. Eng. Comput.*, vol. 46, no. 12, pp. 1271–1276, Dec. 2008.

[7] T. Otsuka, T. Kawada, M. Katsumata, and C. Ibuki, "Utility of second derivative of the finger photoplethysmogram for the estimation of the risk of coronary heart disease in the general population," *Circulat. J.*, vol. 70, pp. 304–310, Mar. 2006.

[8] A. Bacà, G. Biagetti, M. Camilletti, P. Crippa, L. Falaschetti, S. Orcioni, L. Rossini, D. Tonelli, and C. Turchetti, "CARMA: A robust motion artifact reduction algorithm for heart rate monitoring from PPG signals," in *Proc. 23rd Eur. Signal Process. Conf. (EUSIPCO)*, Sep. 2015, pp. 2646–2650.

[9] B. S. Kim and S. K. Yoo, "Motion artifact reduction in photoplethysmography using independent component analysis," *IEEE Trans. Biomed. Eng.*, vol. 53, no. 3, pp. 566–568, Mar. 2006.

[10] B. Lee, J. Han, H. J. Baek, J. H. Shin, K. S. Park, and W. J. Yi, "Improved elimination of motion artifacts from a photoplethysmographic signal using a Kalman smoother with simultaneous accelerometry," *Physiol. Meas.*, vol. 31, no. 12, p. 1585, 2010.

[11] M. Raghuram, K. V. Madhav, E. H. Krishna, and K. A. Reddy, "Evaluation of wavelets for reduction of motion artifacts in photoplethysmographic signals," in *Proc. 10th Int. Conf. Inf. Sci. Signal Process. Their Appl. (ISSPA)*, May 2010, pp. 460–463.

[12] M. R. Ram, K. V. Madhav, E. H. Krishna, N. R. Komalla, and K. A. Reddy, "A novel approach for motion artifact reduction in PPG signals based on AS-LMS adaptive filter," *IEEE Trans. Instrum. Meas.*, vol. 61, no. 5, pp. 1445–1457, May 2012.

[13] T. Suzuki, K.-I. Kameyama, and T. Tamura, "Development of the irregular pulse detection method in daily life using wearable photoplethysmographic sensor," in *Proc. Annu. Int. Conf. IEEE Eng. Med. Biol. Soc.*, Sep. 2009, pp. 6080–6083.

[14] E. Gil, P. Laguna, J. P. Martínez, Ó. Barquero-Pérez, A. García-Alberola, and L. Sörnmo, "Heart rate turbulence analysis based on photoplethysmography," *IEEE Trans. Biomed. Eng.*, vol. 60, no. 11, pp. 3149–3155, Nov. 2013.

[15] A. Sološenko, A. Petrėnas, and V. Marozas, "Photoplethysmography-based method for automatic detection of premature ventricular contractions," *IEEE Trans. Biomed. Circuits Syst.*, vol. 9, no. 5, pp. 662–669, Oct. 2015.

[16] M. R. Yousefi, M. Khezri, R. Bagheri, and R. Jafari, "Automatic detection of premature ventricular contraction based on photoplethysmography using chaotic features and high order statistics," in *Proc. IEEE Int. Symp. Med. Meas. Appl. (MeMeA)*, Jun. 2018, pp. 1–5.

[17] L. F. Polania, L. K. Mestha, D. T. Huang, and J.-P. Couderc, "Method for classifying cardiac arrhythmias using photoplethysmography," in *Proc. 37th Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. (EMBC)*, Aug. 2015, pp. 6574–6577.

[18] M. Nano, G. Papini, and P. Fonseca, "Comparing inter beat and inter pulse intervals from ecg and ppg signals," in *Proc. 11th Biomedica Summit*, Eindhoven, The Netherlands, May 2017, pp. 1–7.

[19] M. Aboy, J. McNames, T. Thong, D. Tsunami, M. S. Ellenby, and B. Goldstein, "An automatic beat detection algorithm for pressure signals," *IEEE Trans. Biomed. Eng.*, vol. 52, no. 10, pp. 1662–1670, Oct. 2005.

[20] W. Karlen, M. Turner, E. Cook, G. A. Dumont, and J. M. Ansermino, "CapnoBase: Signal database and tools to collect, share and annotate respiratory signals," in *Proc. Annu. Meeting Soc. Technol. Anesthesia (STA) Conf.*, 2010, p. 25.

[21] J. F. Kenney and E. S. Keeping. *Mathematics of Statistics*, 2nd ed. Princeton, NJ, USA: David Van Nostrand, 1951.

[22] K. Jain and R. C. Dubes, *Algorithms for Clustering Data*. Upper Saddle River, NJ, USA: Prentice-Hall, 1988.

[23] Z. W. Geem, J. H. Kim, and G. V. Loganathan, "A new heuristic optimization algorithm: Harmony search," *J. Simul.*, vol. 76, no. 2, pp. 60–68, Feb. 2001.

[24] R. L. Plackett, "Karl Pearson and the Chi-Squared Test," *Int. Stat. Rev.*, vol. 51, no. 1, pp. 59–72, Apr. 1983.

[25] S. Deb, S. Fong, Z. Tian, R. K. Wong, S. Mohammed, and J. Fiaidhi, "Finding approximate solutions of NP-hard optimization and TSP problems using elephant search algorithm," *J. Supercomput.* New York, NY, USA: Springer, vol. 72, no. 10, pp. 3960–3992, Oct. 2016, doi: 10.1007/s11227-016-1739-2.

[26] P. R. Sujin, T. R. D. Prakash, and M. M. Linda, "Particle swarm optimization based reactive power optimization," *J. Comput.*, vol. 2, pp. 73–78, Jan. 2010.
[27] X. Meng, Y. Liu, X. Gao, and H. Zhang, "A new bio-inspired algorithm: Chicken swarm optimization," in *Advances in Swarm Intelligence*. Berlin, Germany: Springer, 2014, pp. 86–94.
[28] A. Bouzidi and M. E. Riffi, "Cat swarm optimization to solve job shop scheduling problem," *J. Theor. Appl. Inf. Technol.*, vol. 72, no. 2, pp. 1–6, Feb. 2015.
[29] C. Bandt and B. Pompe, "Permutation entropy: A natural complexity measure for time series," *Phys. Rev.Lett.*, vol. 88, Apr. 2002, Art. no. 174102.
[30] G. Li, "Application of finite mixture of logistic regression for heterogeneous merging behavior analysis," *J. Adv. Transp.*, vol. 2018, Nov. 2018, Art. no. 1436521, doi: 10.1155/2018/1436521.
[31] T. Joachims, *Text Categorization with Support Vector Machines: Learning With Many Relevant Features*. Berlin, Germany: Springer, 1998.
[32] S. Wu, Z. Yang, X. Zhu, and B. Yu, "Improved k-nn for short-term traffic forecasting using temporal and spatial information," *J. Transp. Eng.*, vol. 140, no. 7, Jul. 2014, Art. no. 04014026.
[33] A. M. ElNawasany, A. F. Ali, and M. E. Waheed, "A novel hybrid perceptron neural network algorithm for classifying breast MRI tumors," in *Proc. Int. Conf. Adv. Mach. Learn. Technol. Appl.*, Cairo, Egypt, Nov. 2014, pp. 357–366.
[34] A. A. Balamurugan, R. Rajaram, S. Pramala, S. Rajalakshmi, C. Jeyendran, and J. D. S. Prakash, "NB+: An improved Naïve Bayesian algorithm," *Knowl.-Based Syst.*, vol. 24, no. 5, pp. 563–569, 2011.
[35] J. S. Lee and E. S. Lee, "Exploring the usefulness of a decision tree in predicting people's locations," *Procedia-Social Behav. Sci.*, vol. 140, pp. 447–451, Aug. 2014.
[36] D. Chen and F. Kong, "Hybrid Gaussian pLSA model and item based collaborative filtering recommendation," *Comput. Eng. Appl.*, vol. 46, no. 23, pp. 209–211, 2010.
[37] G.-B. Huang, H. Zhou, X. Ding, and R. Zhang, "Extreme learning machine for regression and multiclass classification," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 42, no. 2, pp. 513–529, Apr. 2012.
[38] D. Ramachandran, V. P. Thangapandian, and H. Rajaguru, "Computerized approach for cardiovascular risk level detection using Photoplethysmography signals," *Measurement*, vol. 150, Jan. 2020, Art. no. 107048.
[39] L. Hao, S. H. Ling, and F. Jiang, "Classification of cardiovascular disease via a New SoftMax Model," in *Proc. 40th Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. (EMBC)*, Jul. 2018, pp. 486–489.
[40] K. H. Miao and J. H. Miao, "Coronary heart disease diagnosis using deep neural networks," *Int. J. Adv. Comput. Sci. Appl.*, vol. 9, no. 10, pp. 1–8, 2018.
[41] S. Shobitha, R. Sandhya, B. K. Niranjana, and M. A. M. Ali, "Recognizing cardiovascular risk from photoplethysmogram signals using ELM," in *Proc. 2nd Int. Conf. Cogn. Comput. Inf. Process. (CCIP)*, Aug. 2016, pp. 1–5.
[42] Z. S. Hosseini, E. Zahedi, A. H. Movahedian, H. Fakhrzadeh, and M. H. Parsafar, "Discrimination between different degrees of coronary artery disease using time-domain features of the finger photoplethysmogram in response to reactive hyperemia," *Biomed. Signal Process. Control*, vol. 18, no. 2, pp. 282–292, Apr. 2015.

SUNIL KUMAR PRABHAKAR received the B.E., M.E., and Ph.D. degrees from Anna University, Chennai, India, in 2012, 2014, and 2017, respectively. He is currently a Postdoctoral Research Fellow with the Department of Brain and Cognitive Engineering, Korea University, Seoul, South Korea. His research interests include signal processing, pattern recognition, and machine learning.

HARIKUMAR RAJAGURU received the B.E. degree in electronics and communication engineering from Bharathidasan University, Tiruchirappalli, India, in 1988, and the M.E. degree in applied electronics and the Ph.D. degree in information and communication engineering from Anna University, in 1990 and 2009, respectively. His research interests include soft computing, pattern recognition, machine learning, bio signal processing, and image processing.

SEONG-WHAN LEE received the B.S. degree in computer science and statistics from Seoul National University, Seoul, South Korea, in 1984, and the M.S. and Ph.D. degrees in computer science from the Korea Advanced Institute of Science and Technology, Seoul, in 1986 and 1989, respectively. He is currently the Hyundai-Kia Motor Chair Professor and the Head of the Department of Artificial Intelligence, Korea University. His research interests include pattern recognition, artificial intelligence and brain engineering. He is a Fellow of IAPR and the Korea Academy of Science and Technology.