Abstract—Cardiovascular or heart diseases consist a global major health concern. Cardiovascular diseases have the highest mortality rate worldwide, and the death rate increases with age, but an accurate prognosis at an early stage may increase the chances of surviving. In this paper, a combined approach, based on Machine Learning (ML) with an optimization method for the prediction of heart diseases is proposed. For this, the Improved Auto Categorical Particle Swarm Optimization (IACPSO) method was utilized to pick an optimum set of features, while ML methods were used for data categorization. Three heart disease datasets were taken from the UCI ML library for testing: Cleveland, Statlog, and Hungarian. The proposed model was assessed for different performance parameters. The results indicated that, with 98% accuracy, Logistic Regression (LR) and Support Vector Machine by Grid Search (SVMGS) performed better for the Statlog, SVMGS outperformed on the Cleveland, while the LR, Random Forest (RF), Support Vector Machine (SVM), and SVMGS performed better with 97% accuracy on the Hungarian dataset. The outcomes were improved by 3 to 33% in terms of performance parameters when ML was applied with IACPSO.

Keywords—SVMGS; IACPSO; KNN; LR

I. INTRODUCTION

Globally, many people suffer from heart diseases [1]. In 1990, there were 24 million fatalities related to heart disease in the United States, by 2010 that number had risen to 38 million, a 59% increase [2]. According to current forecasts, India will have the world’s highest prevalence of cardiovascular diseases and will soon overtake the rest of the world [3]. Heart disorders are responsible for almost 4 million fatalities in Europe and 1.9 million deaths in the EU [4]. In Africa, heart diseases are the leading cause of death among persons over the age of 35 [5]. Massive quantities of data on heart illnesses are collected from hospitals all around the world, which can be used manually to quantify disease rates. However, the data so far have not been efficiently translated to correlate with disease risk and symptoms [6]. Cardiovascular disorder is accompanied by common symptoms of chest tightness, loss of body strength, and swollen legs [7]. Health history examination, clinical test reports, and associated symptoms are usually used by the doctors for diagnosis. But the obtained results by this method are not always accurate, whereas they are costly and difficult to computationally analyze [8]. Researchers have tried to come up with an efficient technique to detect heart diseases since the current diagnostic approaches for heart disease are not very effective in identifying the early stages [9]. It has been reported from different methodological approaches that a combination of ML and optimization methods may be effective in predicting early stages of heart diseases [10]. Appropriate data are needed for training and testing in ML predictive models. We can increase the performance of ML algorithms by optimal dataset balancing for training and testing [11].

Feature selection is capable of reducing dimensionality, increasing efficiency, and enhancing classification accuracy [12-14]. The data comprising several dimensions may create trouble in feature selection. According to [15-18], classification and clustering methods of ML were proven to be more effective in terms of accuracy rates. Several feature selection evaluation metrics were investigated in [19], to improve the computational efficiency of ML algorithms as well as to discuss the unexpected problems of feature selection. Various data mining and combinations of data mining with optimization algorithms have been proposed as a means of detecting heart diseases. Ant Colony Optimization (ACO) was applied to select an effective subset from a large training set with improved accuracy in [20]. In [21], a combination of optimization and data mining was proposed, based on Glowworm Swarm Optimization (GSO) with k-means to improve the accuracy of image classification. For omics data
classification, a Particle Swarm Optimization (PSO)-based model was developed in [22] and claimed good accuracy. In [23], a hybrid structure, based on PSO and grid search was proposed to predict heart diseases with 95.95% accuracy. Authors in [24], suggested a combined approach of PSO with Support Vector Machine (SVM) and Convolutional Neural Networks (CNNs). The highest accuracy, 98%, was achieved by PSO with CNN. In [25], the performance of swarm optimization algorithms Artificial Bee Colony (ABC), PSO, and ACO to an Artificial Neural Network (ANN) was studied, and PSO was determined to be the most effective. A Hybrid Genetic Algorithm (HGA) with k-means was implemented in [26] for classifying and predicting heart diseases with 94.06% accuracy. In [27], ANNs with PCA and PSO were used to predict cardiac diseases, with an accuracy of 98%. In [28], PSO algorithm was proposed for dimension reduction, and several classification methods were used to diagnose heart diseases. Utilizing the dataset from the UCI library, Naïve Bayes (NB), Decision Tree (DT), and k-Nearest Neighbour (KNN) models were applied in [29] to predict heart diseases. In [30], PSO with feedforward backpropagation ANN were implemented to diagnose heart diseases, and achieved 91.94% accuracy. NB with a Genetic Algorithm (GA) were used to eliminate unnecessary information in [31] achieving high accuracy. To deal with the issues related to overfitting and underfitting, as well as selecting attributes, a deep ANN was implemented in [32], and the achieved precision was 93.33%. Similarly, Random Search Algorithm (RSA) for feature selection and Random Forest (RF) for classification were used in [33]. Several DT-based methods along with PSO were used in [34], to identify heart disease occurrence and the highest precision was obtained using a bagged tree with PSO. SVM with a multiclass approach of ML was applied in [35], to detect apple fruit diseases. In [36], crow search with deep learning method were used for the prediction of Parkinson’s disease with 96% accuracy. For the detection of various plant diseases, ML algorithms were used in [37]. Various unsupervised learning and optimization methods were used in [38-40] for the prediction of heart diseases.

Most researchers used supervised and unsupervised ML methods and swarm intelligence-based optimization methods such as ACO, GSO, PSO, etc., in conjunction with ML. But their approaches were not stable in handling real-time situations. Hence, there is a need for an automated approach, which can generate the optimal solution based on the current situation. The present research work proposes a combined approach, including ML algorithms with optimization to predict heart diseases. ML methods, such as Logistic Regression (LR), DT, SVM, SVM by Grid Search (SVMGS), RF, KNN, and NB, are used, and the Improved Auto Categorical PSO (IACPSO) method is applied for selecting an optimized set of features. The major objectives of this research are:

- To evaluate the combined impact of ML algorithms with optimization for heart disease prediction.

II. MATERIALS AND METHODS

Three heart disease datasets were taken from the UCI ML library for testing: Cleveland, Statlog, and Hungarian [41]. The datasets consist of a total of 76 attributes, but only 14 relevant attributes including the attributes preferred in most published experiments [42]. Predicted trait values were represented by A and P indicating the absence and presence of heart disease respectively. ML algorithms such as LR [43, 44], DT [45, 46], RF [47], KNN [48], SVM [49, 50], and NB [45, 48], were used for prediction and analysis. IACPSO was used to select an optimal set of features.

A. IACPSO

PSO is a search-based stochastic optimization technique based on population. The particles, which are potential solutions in PSO, follow the current optimum particles through the problem space [51]. The performance of PSO depends on three control parameters which are the inertial weight (w), the acceleration coefficients (C1 and C2), and random numbers (R1 and R2) [52]. The inertial weight is used for maintaining the effect of convergence and diversity. A large value of w indicates better global exploration, while a small value works on exploitation. Unbalanced values containing the parameters can hurt the results, such that if we take low values for C1, it tends to acquire a smooth particle trajectory and abrupt movements. Similarly, if C2 is much greater than C1, it tends to excessive wander and cause premature convergence [53]. Large inertial weight tends to global search ability and small inertial weight leads to increase in local search power. By dynamic changing w, the acceleration coefficients, efficiently explore the search space [52]. Therefore, in the proposed IACPSO, the control parameters are updated automatically based on the number of particles as well as balancing them at each iteration. The Steps included in IACPSO are given below:

Step I: Initialize the Particle size \( P_0, P_1, P_N, \ldots, P_n \) in the D-dimensional space.

Step II: Initialize the velocity \( V^d_x(0) \), where \( x \in \{ P_0, P_1, P_N, \ldots, P_n \} \), \( d \in \{0, 1, \ldots, D\} \). Calculate the velocity of a particle at the mth iteration by using (1):

\[
V^d_x(m) = wV^d_x(m-1) + C_1(L_x(m) - P^d_x(m-1)) + C_2(G(m) - P^d_x(m-1))
\]

where w is the inertial weight, \( C_1 = R_1 \) (local accelerations), \( C_2 = R_2 \) (global accelerations), \( C_1 \) and \( C_2 \) are the acceleration coefficient, and \( R_1 \) and \( R_2 \) are random numbers. \( P^d_x \) is the position of the particles, \( L_x(m) \) and \( G(m) \) are the local and global best positions. For each iteration, control parameter values have been chosen automatically on the basis of the number of swarm particles (n), which are given in (2.5). For the value of \( C_1 \) and \( C_2 \), generate the n/8 random number between 0 to 2 for the selection of \( C_1 \) and \( C_2 \).

\[
C_1 = \{ C_1^1, C_2^2, C_1^3, \ldots, C_1^n \} \text{ when } 0 \leq C_1 \leq 2
\]
\[ C_2 = \{ C_1^2, C_2^2, C_3^2, \ldots, C_n^2 \} \text{ where } 0 \leq C_2 \leq 2 \]  
(3)

\[ \emptyset_1 = \sum_{n=1}^{p_n} C_1 \cdot R_1 \]  
(4)

\[ \emptyset_2 = \sum_{n=1}^{p_n} C_2 \cdot R_2 \]  
(5)

Then, as per (6), divide the value of \( \emptyset_1 \) and \( \emptyset_2 \) into three categories: Low (L), Mid (M), and High (H).

\[ \emptyset_1, \emptyset_2 = \begin{cases} 
L & \text{if } (0 \leq C_1 \leq 0.8) \\
M & \text{if } (0.9 \leq C_1 \leq 1.2) \\
H & \text{else } (1.3 \leq C_1 \leq 2) 
\end{cases} \]  
(6)

Similarly, for the value of \( w \), select three values between 0.4 and 0.9 and categorize those into Low (L), Mid (M), and High (H), where L is close to 0.4, M is nearer to the mean of the other two values, and H is close to 0.9. For each iteration select the value of \( w \) as per the following conditions:

(i) If (\( \emptyset_1 \) \& \( \emptyset_2 \) \in L) then consider \( w = H \) (value)

(ii) If (\( \emptyset_1 \) \& \( \emptyset_2 \) \in M) then consider \( w = M \) (value)

(iii) If (\( \emptyset_1 \) \& \( \emptyset_2 \) \in H) then consider \( w = L \) (value)

So, the updated velocity at each iteration is:

\[ V_x^d(m) = w(L, W, H)V_x^d(m - 1) + \sum_{n=1}^{p_n} C_1 \cdot R_1(L_x(m) - P_x^d(m - 1)) + \sum_{n=1}^{p_n} C_2 \cdot R_2(G(m) - P_x^d(m - 1)) \]  
(7)

**Step III:** Calculate position of the particles, given in (8):

\[ P_x^d(m) = P_x^d(m - 1) + V_x^d(m) \]  
(8)

By using (8), we get n/8 number of positions for each particle, then we proceed to step IV.

**Step IV:** Calculate the current fitness function \( Y(P) \). Based on the n/8 position of particles, calculate the fitness function, given in (9) and (10), and choose the best one based on Minimization or Maximization.

\[ L_x(m) = P_x(m)_L : Y(P_x(m)_L) = \min_{\max_{\text{max}} \text{max}} Y(P_x(k)) \]  
(9)

\[ G(m) = L_x(m)_L : Y(L_x(m)_L) = \min_{\max_{\text{max}} \text{max}} Y(L_x(m)) \]  
(10)

**Step V:** On the basis of the fitness value, update \( L_x(m) \) and \( G(m) \).

### III. EXPERIMENTAL RESULTS AND ANALYSIS

Figure 1 shows the flow chart of the suggested methodology. Cleveland, Statlog, and Hungarian datasets of heart diseases were considered for the evaluation of the proposed approach. Experiments were run using an x64-based processor with Windows 10 OS and an Intel (R) Core (TM) i5-7200 CPU @ 2.50GHz. The analysis and visual presentation were done using Python and Java. The experimental results and analysis on heart diseases datasets, based on ML algorithms like LR, SVM, DT, SVMGS, RF, NB, and KNN and the IACPSO optimization method are reported.

**A. Result Analysis Based on ML Algorithms**

We assessed the effectiveness of the ML models by using parameters such as accuracy (AC), precision (PR), Matthews Correlation Coefficient (MCC), sensitivity (SV), and F-score (FS) [11]. Table 1 presents the AC, PR, SV, FS, and MCC values, where PR, SV, and FS were either A or P. Here, 25% of the data were used for testing and the rest for training. For KNN, the considered values of \( k \) were 11, 17, and 15 for Cleveland, Statlog, and Hungarian datasets respectively. According to the comparative analysis shown in Figure 2, SVMGS outperformed the other methods in all aspects and achieved accuracy of 89% for Cleveland and Hungarian datasets, while for Statlog dataset, NB and LR showed better accuracy of 91%.

**B. Result Analysis Based on IACPSO with ML Methods**

IACPSO was used for feature selection. The selection of features depend upon their ranks, their values are either 0 or 1, with 0 representing the rejection of a feature and 1 representing its selection. The number of features represents the solution size for each data set. For the optimization process, the selected features along with the performance of the classifier were taken into account. The fitness function was

\[ \alpha \cdot C(E) + \beta \cdot (|SF|/(|TF|)) \]  
(11)

where \( C(E) \) is the misclassification rate, \(|SF|\) shows the number of the selected features, \(|TF|\) represents the total features in the data set, and \( \alpha \) belongs to \([1,0]\), \( \beta = (1 - \alpha) \). The value of \( \alpha \) and \( \beta \) were taken from [54, 55]. For experimentation, the parameters values are: population size=12, number of iterations=100, dimension=7, \( \alpha = 0.4 \) to 0.9 and \( C1 \) and \( C2 \) = 0 to 2.
The percentage of data based on misclassification rates (shown in Figure 3) were considered for the experiment. Based on the IACPSO, only 7 of the 14 features were chosen for testing, which were thalassemia, chest pain, number of major vessels, ST anxiety exercise-induced relative to rest, exercise-induced angina, maximal heart rate achieved, and exercise-induced angina. Table II presents the AC, PR, SV, FS, and MCC values, based on the combined performance of ML methods and IACPSO. With 98% accuracy, SVMGS outperformed the other methods in the Cleveland dataset. SVMGS and LR with 98% accuracy outscored RF, DT, KNN, NB, and SVM on the Statlog dataset. LR, RF, SVM, and SVMGS with a 97% accuracy outperformed KNN, NB and DT on the Hungarian dataset. For Cleveland, with an optimal set of features, the highest MCC was achieved by SVMGS, SVM, and RF, while LR did better in terms of MCC in Statlog and Hungarian. Figure 4 compares the ML algorithms with IACPSO results based on AC, average PR, and SV.
Cleveland dataset, LR, RF, SVM, and SVMGS performed better in terms of average PR, RF and SVM in terms of average SV. LR and SVMGS achieved the highest average PR of 99%, whereas the highest average SV was achieved by RF, SVM, and SVMGS for the Statlog dataset. Regarding the Hungarian dataset, LR, RF, and SVM, with 98% of average PR and SV, performed better than DT, SVMGS, KNN, and NB. With 97%, LR, RF, SVM, and SVMGS outperformed in terms of AC.

Fig. 3. Comparison of the misclassification rates of ML algorithms.

### IV. DISCUSSION

The performance of ML algorithms with and without optimization was examined in this paper. An optimization algorithm (IACPSO) was used for the selection of features to improve accuracy in less time, using the optimum value of the acceleration coefficients in each iteration. In addition, these values were also linked to the graded inertial load, to balance exploration and exploitation. Table III compares the results, based on ML techniques with and without optimization. Figure 5 shows the improvement rates of performance parameters when combined approach (ML algorithms + IACPSO) were used. Outcomes were improved by 3 to 33% in terms of performance parameters when ML was applied with IACPSO. As per Table III and Figure 5, the major findings were as follows:

- When LR was applied separately, the achieved AC was 87% (Cleveland), 91% (Statlog), and 88% (Hungarian) but in the combined methodology of LR with IACPSO, the AC increased by 9% for Cleveland and Hungarian and 7% for Statlog.

- When the DT was applied separately, AC was 82% for Cleveland and Hungarian, and 83% for Statlog, but in the combined methodology of DT with the IACPSO, the AC increased by 10% for Cleveland and Statlog and 11% for Hungarian.

- In the case of RF with IACPSO, AC was 10%, 8%, and 11% higher for Cleveland, Statlog, and Hungarian than for separately applied RF.

- For SVM, the obtained AC with the IACPSO in Cleveland, Statlog and Hungarian was 97%, which was 10% higher than when using only SVM. There was an increase in AC of 1% in Cleveland and Statlog when IACPSO was used with SVMGS.

- For KNN with IACPSO, an increase of 16% in AC occurred for Cleveland and Statlog. A large difference was found in the improvement percentage in the value of MCC, being 33% in Cleveland, 31% in Statlog, and 27% in Hungarian.

- For NB, improvement after IACPSO was 3 to 5% in terms of AC, PR, SV, and FS.

For all the aspects of performance used in the current research, ML methods with IACPSO were proven to be superior. ML algorithms were used for the classification and IACPSO method was applied for the selection of effective features. This type of combined approach gave better models for the early prediction of cardiovascular diseases.

![Comparison of ML methods with and without IACPSO](image)

### TABLE III. RESULT COMPARISON OF ML METHODS WITH AND WITHOUT IACPSO

<table>
<thead>
<tr>
<th>Datasets</th>
<th>Cleveland</th>
<th>Statlog</th>
<th>Hungarian</th>
</tr>
</thead>
<tbody>
<tr>
<td>ML Methods</td>
<td>AC</td>
<td>PR</td>
<td>SV</td>
</tr>
<tr>
<td>LR</td>
<td>87</td>
<td>87</td>
<td>87</td>
</tr>
<tr>
<td>LR+IACPSO</td>
<td>96</td>
<td>97</td>
<td>97</td>
</tr>
<tr>
<td>DT</td>
<td>82</td>
<td>83</td>
<td>82</td>
</tr>
<tr>
<td>DT+IACPSO</td>
<td>92</td>
<td>93</td>
<td>94</td>
</tr>
<tr>
<td>RF</td>
<td>87</td>
<td>87</td>
<td>87</td>
</tr>
<tr>
<td>RF+IACPSO</td>
<td>97</td>
<td>97</td>
<td>97</td>
</tr>
<tr>
<td>SVM</td>
<td>87</td>
<td>87</td>
<td>87</td>
</tr>
<tr>
<td>SVM+IACPSO</td>
<td>97</td>
<td>97</td>
<td>98</td>
</tr>
<tr>
<td>SVMGS</td>
<td>89</td>
<td>89</td>
<td>89</td>
</tr>
<tr>
<td>SVMGS+IACPSO</td>
<td>98</td>
<td>97</td>
<td>96</td>
</tr>
<tr>
<td>KNN</td>
<td>76</td>
<td>76</td>
<td>75</td>
</tr>
<tr>
<td>KNN+IACPSO</td>
<td>92</td>
<td>92</td>
<td>91</td>
</tr>
<tr>
<td>NB</td>
<td>87</td>
<td>87</td>
<td>87</td>
</tr>
<tr>
<td>NB+IACPSO</td>
<td>92</td>
<td>93</td>
<td>92</td>
</tr>
</tbody>
</table>
local and global best, according to the objective function. The detecting different diseases.

more parameters, with real or primary datasets and various helped in predicting cardiovascular diseases and yielded better algorithms, such as DT, LR, SVM, SVMGS, NB, KNN, and proposed ML approach with an optimized set of features which helped in proper exploration and exploitation. In each iteration, the best solution was used for the next iteration. The proposed approach was investigated on Cleveland, Statlog, and Hungarian datasets and the evaluation was performed based on AC, PR, SV, FS, and MCC. The ML algorithms were compared and assessed with and without optimization, and it was found that the former were superior in all parameters. The proposed ML approach for preprocessing the data set for predicting fertility success rate, International Journal of Control Theory and Applications, vol. 9, no. 27, pp. 255–260, Jan. 2016.


L. Demidova and I. Klyueva, “Data classification based on the hybrid versions of the particle swarm optimization algorithm,” in 7th Mediterranean Conference on Embedded Computing, Badva,