Adaptive feature selection and classification by using optimization algorithm

A dissertation submitted to the University of Hyderabad in partial fulfilment of the requirements for the award of degree of

Master of Technology

in

Information Technology

by

Mandala Sookshma
16MCMB09

School of Computer and Information Sciences
University of Hyderabad
Hyderabad-500046

June 2018

CERTIFICATE

This is to certify that the dissertation entitled **Adaptive feature selection and classification by using optimization algorithm** submitted by **Mandala Sookshma**, bearing Reg. No. 16MCMB09, in partial fulfilment of the requirements for the award of Master of Technology in Information Technology is a bonafide work carried out by her under my supervision and guidance.

The dissertation has not been submitted previously in part or in full to this or any other University or Institution for the award of any degree or diploma.

Dr Naveen Nekuri
Asst Professor
Supervisor
School of CIS
University of Hyderabad

Dean
School of CIS
University of Hyderabad

Declaration

I, Mandala Sookshma, hereby declare that this dissertation entitled **Adaptive feature selection and classification by using optimization algorithm** submitted by me under the guidance and supervision of **Dr. Naveen Nekuri, Asst Prof.** is a bonafide work carried out by myself. I also declare that this dissertation has not been submitted previously in part or in full to this University or other University or Institution for the award of any degree or diploma. I hereby agree that my disseratation can be deposited in Shodganga/INFLIBNET.

A report of plagiarism statistics from the University Librarian is enclosed.

Date: (Mandala Sookshma)
Place: 16MCMB09

//Countersigned//

(Dr. Naveen Nekuri, Asst Prof)
Supervisor

This Thesis is dedicated to,
My family and supervisor

Acknowledgement

First and foremost, my utmost gratitude to **Dr. Naveen Nekuri, Asst Prof.**, my project supervisor, for his supervision, advice, and guidance from the very early stage of this research as well as giving me extraordinary experiences throughout the work. His vision, encouragement and support in various ways made this work possible. I feel blessed to have got an opportunity to work under his guidance.

I am extremely grateful to our dean, **Prof. Arun Agarwal**, and also **Prof. C.R.Rao** for providing excellent computing facilities and a nice atmosphere for doing my project.

My thanks and appreciations also goes to Srileka Panda and my fellow colleagues for their direct and indirect support in completing my dissertation work.

Last but not the least I would like to express my sincere regards to my family for their encouragement and support during my project.

With Sincere Regards,
Mandala Sookshma

Abstract

Classification will be done by building a model using neural networks, fuzzy, rough sets. These models which are developed may not be used as an early warning system. Hence, rule generation developed by other models like decision trees which can be used as an early warning system. Hence, we propose a model to generate rules by using optimization techniques called Particle Swarm Optimisation(PSO).

In problems related to classification, the dataset contains a vast number of features. Out of these, some are useful and some are not. In order to evaluate these useful features, Feature Selection(FS) is being implemented. Among the different subset of features, in order to find best among them in lesser time heuristic/optimization algorithms are being implemented. Optimal feature subsets obtained from these may contain redundancy as they do depend on the correlation of features, so in order to reduce this redundancy a correlation coefficient known as Pearson Correlation Coefficient(PCC) is being implemented.

In this proposed model, PSO along with PCC is used for finding optimal feature subset and these are used to implement the rule generation for the classification accuracy.

This project aims to implement "correlation coefficient with PSO for Feature Selection" and"PSO for Classification". These are being tested on benchmark datasets like Iris, Wine, Thyroid, Australian, German etc.

Contents

List of Tables

List of Algorithms

Chapter 1

Introduction

1.1 Introduction to Feature Selection and Classification

Since few years the size of the datasets has been increasing very rapidly resulting in the huge number of features and this huge dataset size may not be completely useful for the process of classification. Datasets which contain a large number of features, contains some features with sufficient information, some features with less information. Presence of these insufficient features is of no use in the classification, so in order to remove these features, Feature Selection(FS) is being implemented to increase classificaion accuracy. Feature selection aim is to provide finally a subset of features which are useful for classification thus by removing features with are of no use or it can be redundant features also.

So in order to get the optimal subset of features, all the possible subsets have to be evaluated which is a time-consuming process, thus leading to an NP-Hard problem. So, in order to reduce the time complexity and to get optimal subset heuristic, metaheuristic, random search strategies are being implemented. There are various feature selection methods that are being implemented.

1.2 Types of Feature Selection:

Feature Selection can be done in various ways like-

- **Filter Method:** This is based on the statistical formula and independent of any algorithm. This can also be considered as a preprocessing step.

  Examples include Pearson Correlation Coefficient(PCC), Linear Discriminant Analysis(LDA), etc.

- **Wrapper Method:** In this, we find the subset of features and these are used to build a model. Depending on the conclusions of the model, removal/addition of features occurs. This method includes feature forward selection, backward elimination, recursive approach etc.

  1:Feature forward selection In this, the model starts with no features and at each and every iteration features are being added. Adding of features stops when there is no better change in the classification accuracy.

  2:Feature Backward selection In this, the model starts with all features . At each and every iteration the features are removed which are of no use. This process continues until there is no change in the accuracy of the classifier by removal of features.

  3:Recursive selection It is used to find the optimal feature subsets from the available features. These subsets are being evaluated in each and every iteration and the optimal subset is being found.

- **Hybrid method:** Combination of both filter and wrapper methods.

- **Embedded method:** Feature Selection is being used as a part of building a model.

By implementing any of these above methods results in a optimal feature subset, whose size may be lesser or equal to the original dataset.

1.3 Motivation

In order to find the best feature subsets in lesser time, the feature selection uses heuristic methods, because of their better performance, we have chosen Particle Swarm Optimization(PSO)[14] because it has fewer parameters to adjust, computationally low cost according to the memory and also in speed point of view. As PSO focuses on obtaining better feature subsets rather than not depending on the correlation information, so we choosen PCC(Pearson Correlation Coefficient) along with PSO to find optimal feature subset. Upon available various classifiers, we have chosen to implement early warning system as it takes lesser time to train a model.

1.4 Problem statement

The main purpose of this project aims to implement an early warning system using rule generation. Here we generate rules by an optimization technique called PSO. Model include: "Correlation coefficient with PSO for Feature Selection" and "PSO for Classification".

1.5 Thesis Organization

The dissertation is as follows, Chapter 2 deals with the literature survey which includes optimization algorithms like PSO, ACO, GA, BPSO, PCC, study on various research papers, classification techniques etc. Chapter 3 deals with methodologies and implementation, wherein details about the proposed model. Chapter 4 deals with the PSO implementaion against various test objective functions, results of the proposed model and discussions, experimental configuration. Chapter 5 deals with the future work and the conclusion.

Chapter 2

Literature Survey

2.1 Feature selection for classification

Feature selection is like a pre-processing step for dataset [that contain the huge number of features] that is being used in classification. This helps in increasing the accuracy of the classifier, decreasing time, memory requirements.

Various meta-heuristic algorithms have been used for Feature-Selection, among them include Particle Swarm Optimization(PSO), Ant Colony Optimization(ACO), Genetic Algorithm(GA) etc. As the feature subsets that are being obtained from these algorithms may contain redundant features so in order to remove the redundancy, the correlation coefficient is being used. In this proposed model we have used a linear correlation coefficient known as Pearson Correlation Coefficient(PCC). Apart from PCC, there are other feature selection methods depending on different criteria as discussed in [19] An early warning system is being implemented by the rule generation as the classifier. The optimal feature subset that is being obtained by the feature selection is being evaluated by the classifier to find the accuracy. In literature, various metaheuristics algorithms like ACO, PSO, GA etc have been used for the feature selection and classification problems.

2.2 Optimization Algorithms:

2.2.1 Particle Swarm Optimization(PSO)

The PSO is proposed by Kennedy and Eberhart(1995). This algorithm is based on the swarm intelligence by finding the global optimum by updating the generations. In this, each solution is being termed as the particle and group of all those particles are called population. Every particle is associated with the position and velocity to move in the multi-dimensional search space. The algorithm is being run by a predetermined number of iterations. In the first iteration, the position and velocities are being uniformly distributed in the space by using a random number generator. In each and every other iterations, the particles is being updated by its previous best performance and that of the best of its neighbors. The process is being continued until the stopping criterion is being met.

In this, the position and velocities are being update as-
pbest: The best solution that a particle has been attained so far.
gbest: The best solution that is being attained by any particle in the population.
Position and velocity updation is done as below-

Velocity update :

$$\bar{v_{id}}(t+1) = w * v_{id}(t) + c_1 * r_1 * (x_i^{pbest} - x_{id}(t) * v_i(t) + c_2 * r_2 * (xgbest - x_{id}(t)) \tag{2.1}$$

where :
D-> Dimension,
w-> inertia weight $x_{id}(t)$ -> Position of the i_{th} particle at the d_{th} dimension,
$v_{id}(t)$ ->velocity of the i_{th} particle
c_1,c_2 -> learning factors.
r_1,r_2 -> random numbers ,range is [0,1],

Postion update:

$$x_{id}(t+1) = x_{id}(t) + v_{id}(t+1) \tag{2.2}$$

Algorithm 1: PSO algorithm

```
Input: $Population_{size}$
 output: $P_{gbest}$
 for $i := 1$ to $Population_{size}$ do
     $P_{position}$=Randomposition($Population_{size}$)
     $P_{velocity}$=Randomvelocity()
     pbest=$P_{position}$
     //evaluate fitness function
     if $fitness(pbest)$ is better than $fitness(gbest)$ then
         $gbest := pbest$ ;
while stopcondition() do
     For$P \in$ population
     $P_{velocity}$=UpdateVelocity();
     $P_{position}$=UpdatePosition();
     //evaluate fitness
     if $fitness(position)$ is better than $fitness(pbest)$ then
         $pbest := position$ ;
     if $fitness(pbest)$ is better than $fitness(gbest)$ then
         $gbest := pbest$ ;
return gbest;
```

PSO [1] is the most commonly used metaheuristic algorithm for the feature selection problems. The size of each particles is equal to the number of features of the given dataset problem. The final result of this algorithm results in providing the optimal feature subsets . These optimal feature subsets are being used to improve the classification accuracy.

2.2.2 Binary Particle Swarm Optimization(BPSO)

In this binary version of PSO(BPSO) [2]is being used for the binary problem spaces. In this, position of the particle is being represented as a binary vector, the size of this vector is equal to the number of the features. If the value of the feature in the binary vector is 1 then it means that the particular feature is selected, if 0 then that feature is not selected. It uses the concept of velocity as a probability that a bit position takes the value of 0 or 1. As in the PSO the position and velocity changes, the same happens in BPSO also. For updating position will be calculated as-
Position update:

$$S(v_{id}(t+1)) = \frac{1}{1+e^{-v_{id}}} \tag{2.3}$$

if(rand()<S(v_{id})) then x_{id}=1
else x_{id}=0

where S() denotes the sigmoidal function which is used for changing the velocity to a probability which is then used in position updation. rand() is the random number generator function([0.0,1.0]).

2.2.3 Ant Colony Optimization(ACO)

ACO[9] is based on the principle that, ants are able to find the shortest distance between the nest and food by means of path construction and updating of phermona. Path construction is done as the shortest path will have more phermone than compared to the longer paths as these are frequently visited paths because ants have the capability to smell phermone and thus prefer short paths. These two are the principles in the algorithm which is being applied in higher dimensional search space for the solution.

In paper [9], for the feature selection problems, ACO is being used to find an optimal subset of features. Over the iterations, based on the phermone the ants search for the features in the search space. In these feature selection problems, the size of the ants is equal to the number of features present and the features obtained are being evaluated by the classifier. They can be using feature selection methods like the filter, wrapper or hybrid methods in order to get optimal feature subsets for the better accuracy.

2.2.4 Artificial Bee Colony Optimization(ABCO)

ABCO [9], it is based on the behavior of the honey bee swarm. In this, there are three types of bee's -employed, onlookers, scouts. In this algorithm, the food sources can be termed as the possible solutions and the fitness correspond to the nectar amount in that food source. Initially, the food sources are distributed randomly in the search space, now the employed bee's searches for the food source where, the vector amounts greater than that of the previous one. After this process has been finished, now the onlooker bees get this information and depending on the nectars in those, they choose the food sources. The food sources that are being abandoned are being identified and new food sources are to be replaced with those by the scout bees. This complete process is being successfully completed by the social co-operation.

ABC algorithm is being combined with ACO in the paper[9] resulting in a hybrid algorithm in order to find the better feature selection and classification accuracy. The hybrid algorithm helps in the drawbacks of both the algorithms and helps them to improve the classification accuracy. This process can be explained as- in the available search space ants find the feature subsets and these are being passed as food sources to the ABC algorithm. In ABC algorithm these are being evaluated according to the accuracy, fitness function and the best feature subset is being obtained. Depending on this feature subset, the value of ABC algorithms depends. This process continues until a stopping criterion is being met and the best feature subset which has the highest accuracy is being obtained in the end.

2.2.5 Genetic Algorithm(GA)

It is one of the evolutionary algorithms which depends on the concept of the natural selection.

The steps in the Genetic algorithm include:

1)Selection

2)Crossover

3)Mutation

1)Selection:

The process involved in this is, in each and every iteration from a randomly generated population, the two best among them is being selected as parents depending on the fitness function.

2)Crossover:

From the parents selected, new children are being produced by using different crossover techniques like -

a)Uniform crossover

b)one-one

c)multipoint etc

3)Mutation:

This method is being used to apply very slight changes to the children.

By performing the above methods untill the stopping criterion, finally we get the optimized result.

Applications: Genetic algorithms has a variety of applications which include- feature selection, trafficking & routing, robotics,engineering design etc.

2.3 Feature Selection and Feature Extraction:

With reference to [17], overview of different feature selection , feature extraction algorithms are being presented . Inorder to remove not so useful features, dimensionality reduction is being used as the pre-pocessing step. This can be achieved by feature selection and also feature extraction.

- **Feature Selection:** This is used to select features that are relevant.
- **Feature Extraction:** In this method, transformation takes place from higher dimensional space to the lower dimensional space.

L. Ladha et al in [16], signified the advantages of the feature selection as -

- Dimensionality reduction
- Keeps only the relevant features, removes other irrelevant features.
- Running time of the proposed algorithms are being the speedup
- Quality of data is being improved
- Classification accuracy is also being improved.

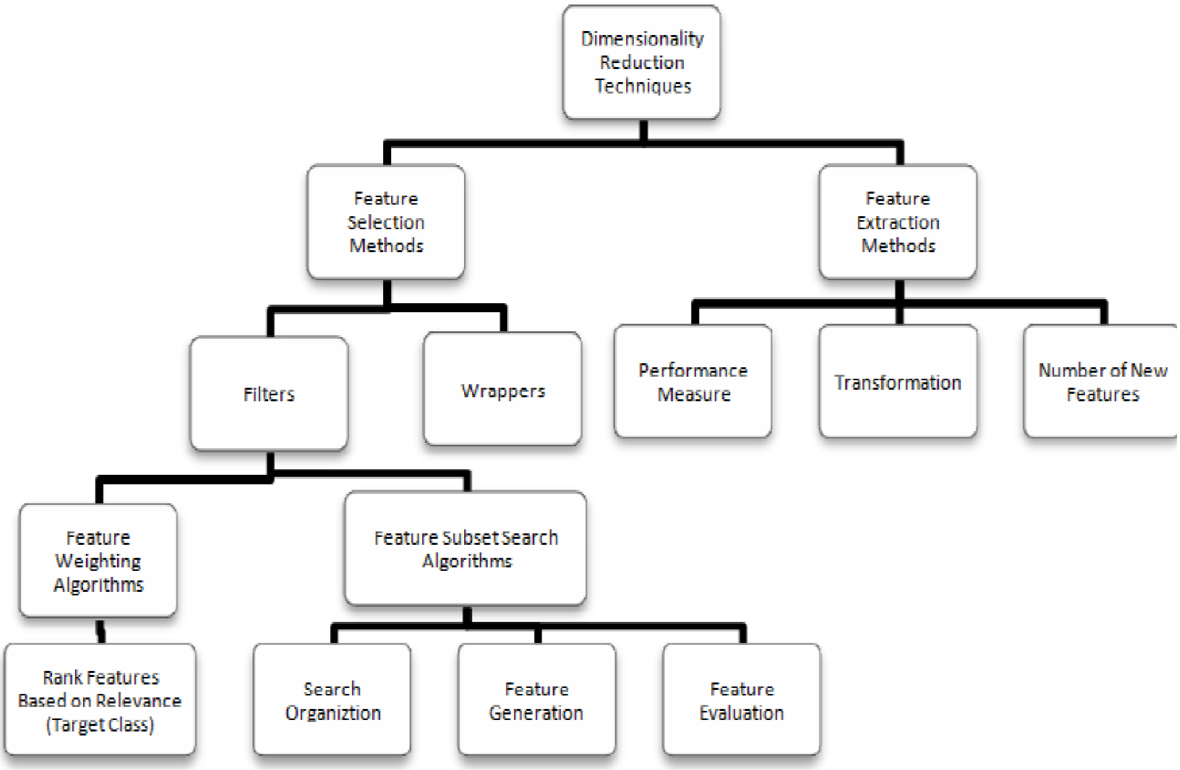
Dimensionality Reduction Techniques
Feature Selection Methods
Feature Extraction Methods
Filters
Wrappers
Performance Measure
Transformation
Number of New Features
Feature Weighting Algorithms
Feature Subset Search Algorithms
Rank Features Based on Relevance (Target Class)
Search Organiztion
Feature Generation
Feature Evaluation

Figure 2.1: Different types of FS and Feature Extraction methods

2.3.1 Algorithms

2.3.1.1 Algorithms related to feature selection

A few algorithms related for feature selection are [17]-

- chi-square
- Information gain
- Correlation based algorithms like PCC [25]

2.3.1.2 Algorithms related to feature Extraction

few algorithms related for feature extraction are [17]-

- Independant component analysis(ICA)
- PrincipaleComponent Analysis etc.

2.4 Binary Bat Algorithm and cuckoo search algorithm

In [22], it explains about two new algorithms called binary bat algorithm, cuckoo search algorithm. Parkinson's is a disease which effects humans, it leads to several problems like the decrease in the movement, effect on muscles etc. This paper[22] presents the classification model of the normal patients to that of the disease affected persons by using feature selection.

2.4.1 Binary Bat Algorithm

This algorithm is based on the behavior of bats. General principle where bats can find its prey, obstacles by use of the echo. Every bat has initially the position, velocity, frequency, loudness ,wavelength. If the bat gets its food, then the loudness is being decreased and pulse emission increases. For the further iterations, it updates its position, velocity, frequency, loudness. After finding the best fitness values, a bat in that neighborhood searches for the optimal solution.

2.4.2 cuckoo search algorithm

This algorithm relies on the concept of the cuckoo bats, where they lay eggs on other bird's nests. Initially birds lay eggs in randomly chosen nets, this can be considered as an n-dimensional space. The eggs that have the better solutions are carried to the next generation. This algorithm can be explained as - In the first iteration, the eggs are laid in the randomly selected nests and the best fitness values are be observed, according to these values the eggs in the nests are being sorted and the top eggs are done crossover to get better solution eggs. This process is continued untill the stop condition is being achieved.

In this paper[22] it is observed that the results for this problem are better when done by bat algorithm, cuckoo algorithm compared to that of the PSO, Genetic algorithm.

2.5 Chaotic maps in BPSO

In [20], they have explained the chaotic maps in BPSO for the feature selection. Chaotic maps used in this paper are - logistic maps, tent maps and the classifier used is k-nn. In feature selection problems related to BPSO, 'w' is the inertia weight which is used to differentiate between the global and local exploration. By using these chaotic maps in BPSO, the value of 'w' is being determined in each and every iteration i.e., 'w' value is not constant for all iterations, it changes every time. 'w' can be calculated as-

$$w(t+1) = 4.0 * w(t) * (1 - w(t)) \tag{2.4}$$

where w ϵ(0,1)
normally 'w' value is set to 0.48[25]

The value of 'w' for tent map is calculated as -

if$(w(t) < 0.7)$
$w(t+1) = \frac{w(t)}{0.7}$
else
$w(t+1) = \frac{10}{3*w(t)*(1-w(t))}$

Effect of value 'w':when the value is nearer to '1', it determines the global exploration, if nearer to '0' then local search.
The paper proposed a model known as CBPSO-KNN , it is described as-
step1: Initial population is being generated
step2: Calculate the fitness for those particles
step3: Find the value of p_{best}, g_{best}. For the next iterations these values are updated according to equations-(2.1),(2.3),(2.4)
step4: 'w' is being calculated acc to equations-(2.5), (2.6)
step5: Go to step 2 until termination condition is not met.
step6: Return final solution.

2.6 Pearson Redundancy Based Filter

In this [21], a Pearson χ^2 test is been proposed and tested for feature selection problems. Generally, features are being ranked and sorted, these top most features have to be evaluated by a wrapper method. It may be expensive, so they have proposed a method a filter method that can perform feature ranking and also remove these redundant features.

2.6.0.1 Pearson's Redundancy Based Filters

χ^2 test is used to find the difference of the probability distributions of two variables. If a feature is redundant to the already selected feature then it's distribution should have the high probability.

The formula is -

$$\chi^2(X,X^{'}) = \sum_{i=1}^{k} \frac{(F_i - F_i^{'})^2}{F_i^{'}} \tag{2.5}$$

where f_i,$f_i^{'}$ are frequency of occurences of features, F_i, $F_i^{'}$ are probability distributions.

Algorithm for the PRBF is as follows-

Here, X contains the feature set, C is for classification.

Relevance Analysis

1. S ∪(X,C) is being calculated and S ordered list is being made .

Redundancy analysis

2.Take first feature X from the list S.

3.Perform X^2 test and remove all features for which X is redundant.

4.Now, select other feature in S and repeat step3 for all other features which are present in S.

2.7 HPSO-LS

In this paper[1], here PSO algorithm uses the local search to select distinct features by using the correlation coefficient. In this paper, they have choosen PSO as it gives results better when compared to other alogithms. The classifier used in this k-nearest neighbour[4] classfier to evaluate the classification accuracy. Proposed model which is explained in the below steps-

- **Step 1:**In this step, size of the subsets that are to be included in feature subset selection are being determined.

- **Step 2:** Feature grouping happens in this step. Grouping of features means forming the similar features into a particular groups, for this PCC is being used. If for any two features the value of PCC is high then it means that those are similar features if value is low then vice-versa.

Now, the correlation value for a_{th} feature is being calculated as-

$$cor_a = \frac{\sum_{b=1}^{f} |c_{ab}|}{b-1} \tag{2.6}$$

Here 'f' is the total features, c_{ab} is the PCC value for features a,b . Now all the features are being arranged in a sorted order from low correlation value to high. Till mid all the features have less correlation values are grouped into a dissimilar group(D), the other half into a similar group(S).

- **Step 3:**Particles initialisation takes place in this. As it is feature selection problem BPSO[2] is being implemented.
- **Step 4:** Positions and velocities are being updated according to equations -(2.1), (2.3), (2.4).
- **Step 5: Local search stratergy:** Main steps include feature segmentation, particle movement. The feature segmentation can be explained as - the similar features are being deleted and dissimilar features are added. Addition of the similar features takes place by 'Add' operator, deletion by 'Delete' operator. Groups D, S that are being obtained in step 2 are being used here. This can be explained better by the following example-
  Example: Let us consider a particle as 110101, now from this particle it is inferred that let A=(f1,f2,f4,f6) are being selected. The features in A are being compared to that of the groups of D, S and they are being divided into two groups like- A_d that contains non-similar features, A_s for similar features.

  Next is the particle movement, where the features are being added or deleted according to the minimum size of the features in both the groups as in paper.
- **Step 6:** This step invloves fitness calculation and gbest values are being updated . Steps 4 to 6 are being repeated till the stipping condition is

being achieved. This proposed model[1] uses k-nn classifier to evaluate these feature subsets.

In this [1] paper, inorder to find the global search it has introduced the concept of local search ability and PCC[5] is being used to remove the dis-similar features. The results obtained from this proposed model is being compared with other algorithms, and it is being concluded that this proposed model has achieved better results.

2.8 Three-stage Feature selection

In this paper[5], the feature selection problem to remove the redundant features to improve the accuracy of the classifier is being done in three-stages-

- Feature Ranking(FR).
- Correlation Analysis (CA)
- Chaotic Binary Particle Swarm Optimisation.

2.8.1 Feature Ranking

In this stage, the features are being ranked and the topmost features are being considered for classification. It includes- Dataset is being divided into training and test sets, a classifier SVM with nfold cross validation is used. Now the training set which contains the features is being divided into nfolds to train a model. Now the performance of the classification is being obtained by applying it on the test dataset. According to the classification accuracies, finally features are being ranked. Now the top ranked features are being selected for classification.

2.8.2 Correlation Analysis(CA) and Pearson Correlation Coefficient(PCC)

For large datasets there may be possibility of redundancy from the obtained features during the first stage. So, in order to remove this redundancy a correlation co-efficient known as Pearson Correlation Coefficient(PCC) is being used.

Pearson Correlation Coefficient(PCC) As there is a possibility in redundancy of the features, Correlation Analysis(CA), [5] is being used to measure the redundancy among the features. If two features have highest correlation values, then they are said to be redundant features .
The values of the PCC ranges between -1 to +1, where -1 means there is a negative correlation , +1 indicates that there is a positive correlation and 0 indicates that they are independent of each other.
PCC can be calculated as-

$$\rho = \frac{\sum (x - \bar{x})(y - \bar{y})}{\sqrt{\sum (x - \bar{x})^2} \sqrt{\sum (y - \bar{y})^2}} \tag{2.7}$$

where : x,$\bar{x}$ is the value and mean of the feature 'x',
y,$\bar{y}$ is the value and mean of the feature 'y',

The value of the Pearson correlation coefficent for the features is being represented in the matrix format as below-

$$R = \begin{bmatrix} 1 & f_{12} & f_{13} & \cdots & f_{1n} \\ 0 & 1 & f_{23} & \cdots & f_{2n} \\ 0 & 0 & 1 & \text{f}_{ij} & . \\ 0 & 0 & 0 & 1 & f_{(n-1)n} \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

If the value of f_{xy} is greater than the threshold limit, it indicates that the features x,y are redundant features
If the value is zero , then there is no redundancy between the features of x and y.

2.8.3 CBPSO

Even though after stage1, stage2 still there may be a possibility of redundant features, irrelevant features because of huge datasets. So, in order to remove these CBPSO[20] is being used. In this, the solution that has high classification accuracy, few number of features has the highest probability to be selected. The fitness function is being calculated as -

$$Fitness = \frac{FP + FN}{TP + TN + FP + FN} + \alpha * Features \tag{2.8}$$

where-

- FP means False Positives
- FN - False Negatives
- TP - True Positives
- TN - True Negatives
- Features are the selected number of features
- α denotes the importance of those selected features.

The main steps involved in this CBPSO is :

- **step 1:**From the second stage of the proposed model a feature subset is being generated.
- **step 2:**Particles positions are being initialized randomly.
- **step 3:**Fitness of the particles are being evaluated using SVM nfold cross-validation.
- **step 4:**Now by using chaotic maps inertia weight is being calculated as according to equation 2.5

- **step 5:**Now the values of p_{best}, g_{best} are being evaluated according to the objective/fitness function. Velocities, positions are being updated according to equations- (2.1), (2.3), (2.4)

- **step 6:**If the termination condition is being met then the optimal subset of features is being obtained. Else again go to **step 3**.

In this paper [5], they have proposed a model on hybrid feature selections and introduced a three-stage method for removal of redundant features so as to improve the classification accuracy. They have evaluated this model on benchmark datasets like wine, australian, vehicle, wbcd, sonar, movement and achieve better results.

2.9 Single Feature Ranking and BPSO based Feature Subset Ranking

The paper [8], has proposed a model which includes two algorithms they are -

- 1. Single feature ranking algorithm

- 2. Using BPSO[2], feature subset ranking algorithm.

2.9.1 Single Feature Ranking Algorithm

Proves that including the topmost features for classification gives accuracy than compared to using all the given features. In this method, the given dataset is partitioned into training and test sets. K-nearest neighbour with n-fold cross validation is being used to find the accuracy. Now, the features are being ranked according to their accuracies.

2.9.2 BPSO based Feature Subset Ranking

From above topmost ranked features there may be a possibility of redundancy among them, inorder to overcome this step is being introduced. This method can be explained as- now dataset is divided into training, test sets into n-folds.'p' feature subset has 'p' features. If a dataset contains 'S' features then 'S' feature subsets will be formed and for this purpose 'S' steps. Each and every step include selecting most useful features. The 'p' step of the BPSO it involves finding those 'p' relevant features that improves the accuracy. For example, if more than 'p' features are selected then those extra features are being deleted randomly, if lesser than added. These subsets are being evaluated by the classifier on the test data. The classifier used is K-nearest neighbor with n-fold cross validation.

2.10 Feature selection by using Genetic Algorithm

In this proposed model [11], feature selection is based on bi-objective GA. Objective functions that are being used are - rough set theory[12], Mutual information[7]. These are being implemented parallel for optimal feature subset. Final feature subset is the aggregation of the feature subsets.

The objective function is being used to select the most relevant features. By using rough set[12] used for providing the approximations[lower, upper] of the given dataset. It also provides minimal sets of the given data.

Ojective function: These objective functions are maximised by assuming two chromosomes, for dis-similar features τ_a(D) (similarity for the 'D' dataset) is high then objective function is also high. So, τ_a(D) is maintained to 0, now only γ_a[12] is the only term.

For example, chromosomes c_1, c_2 are (1 0 1 1 1) and (1 1 1 0 0) respectively. For a decision system, D=U, A, C, D here the conditional C is given as - (a_1, a_2, a_3, a_4, a_5) then chromosomes now become ch_1=(a_1, a_3, a_4, a_5) and ch_2 =(a_1, a_2, a_3). After calculation of τ_a, γ_a for the chromosomes then-

Chromosomes	τ_a	γ_a
ch_1	τ_1	γ_1
ch_2	τ_2	γ_2

Table 2.1: Objective functions of related chromosomes

Possibility	condition	Chromosome selected
1	$\gamma_1 > \gamma_2 \wedge \tau_1 > \tau_2$	ch_1
2	$\gamma_1 > \gamma_2 \wedge \tau_1 < \tau_2$	ch_1
3	$\gamma_1 > \gamma_2 \wedge \tau_1 = \tau_2$	ch_1
4	$\gamma_1 < \gamma_2 \wedge \tau_1 > \tau_2$	ch_1 or ch_2
5	$\gamma_1 < \gamma_2 \wedge \tau_1 < \tau_2$	ch_2
6	$\gamma_1 < \gamma_2 \wedge \tau_1 = \tau_2$	ch_2
7	$\gamma_1 = \gamma_2 \wedge \tau_1 > \tau_2$	ch_1
8	$\gamma_1 = \gamma_2 \wedge \tau_1 < \tau_2$	ch_2
9	$\gamma_1 = \gamma_2 \wedge \tau_1 = \tau_2$	ch_1 or ch_2

Table 2.2: Selection of Chromosomes

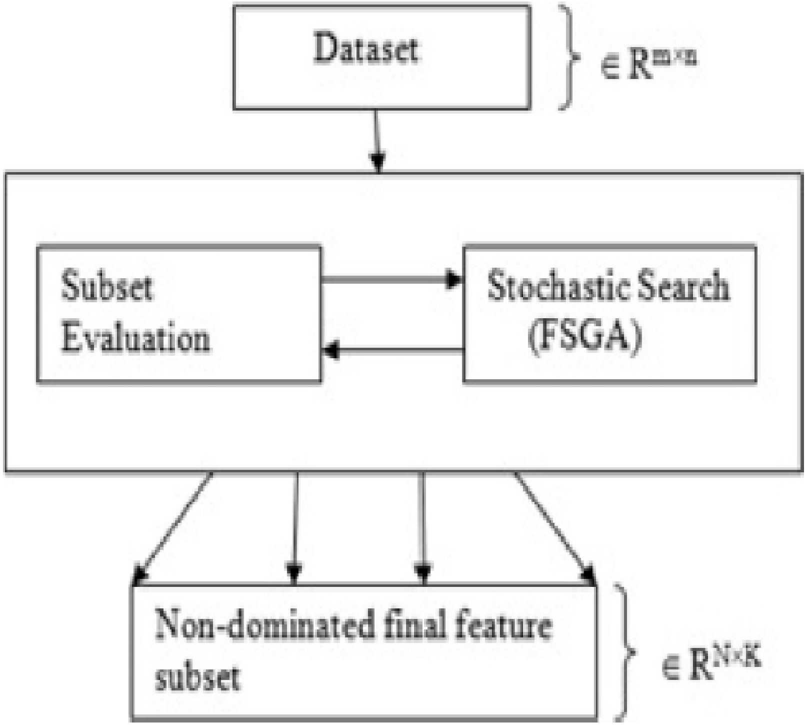

Figure 2.2: Proposed model

In this model, there is equal probability for every population to be a parent, single point crossover[15], is being used. There will be a mutation pool of different mutation techniques any one of them is being selected dynamically. Mutation pool in the algorithm is the increase the quality of members.

The proposed model can be explained as -

- **Step 1:** Initialise population of size POP of size 'A'
- **Step 2:** Calculate fitness, global best values
- **Step 3:** For each POP, select one as parent and the next as also a parent.
- **Step 4:** Produce offspring by using single point cross-over [15]
- **Step 5:** From mutation pool, select any techniques and apply to the generated offspring
- **Step 6:** Now calculate fitness and compare with the previous fitness and update best among them, update parents too
- **Step 7:** Repeat **Step 3-7** until stoping condition is not met
- **Step 8:** Return reduced subset.

2.11 Classification by using PSO

In paper[18], focuses on improving the tree classification rules by using Adaptive Particle Swarm Optimisation(APSO) [13]. IN this the fitness function that is being used is the classification accuracy.

2.11.1 Adaptive Particle Swarm Optimisation(APSO):

Over the years since PSO[14], has proved many real-time applications, but it has drawbacks like getting trapped easily in local optima. In order to overcome this, many versions and modifications of PSO are being introduced. Like to avoid local optima Comprehensive Learning PSO, time variant adjust of the parameters, adapting to parameters by self by performing the algorithm etc., out of these one among is the adaptive PSO [13]. In APSO, in each and every iteration the population distribution varies. Depending on these, four state estimations are being calculated, they are -

- Exploration(s1)
- Exploitation(s2)
- Convergence(s3)
- Jumping out(s4)

The states categorization is dependent on the value 'f', evolutionary factor. This factor is being calculated as-

- **step1:** After the population is distributed, the distance from every particle to other particles is being calculated.
- **step2:** From all those distances, the best distance d_g, minimum distance d_{mn}, maximum distance d_{mx} are saved. Now 'f' value is calculated as -

$$f = \frac{d_g - d_{mx}}{d_{mx} - d_{mn}} \tag{2.9}$$

Parameters tuning 'w' value : The parameter 'w' is now being calculated as -

$$w = \frac{1}{1 + 1.5e^{-2.6f}} \tag{2.10}$$

c1 and c2 parameters: Initially c1, c2 parameters are being set to the value of 2.0, after according to the value of 'f' these parameters are being tuned as -

- If 's1' then increase c1, decrease c2.
- If 's3' then slightly increase c1, slightly decrease c2.
- If 's3' then slightly increase c1, slightly increase c2.
- If 's4' then decrease c1, increase c2.

Description of the proposed model[18]: This method describes a method for threshold values needed to split variables in the decision tree by using APSO13. The steps involved in this proposed model are -

- Decision tree construction
- Optimizing the tree
- Simplification of rules

2.11.2 Decision Tree Construction

Decision tree is being constructed by using CART, the steps involved for tree construction are -

- 1. Tree growth is carried on until there is a single node for every class
- 2. If the time-complexity is high then the pruning takes place, then the best tree is being selected as the preliminary tree

2.11.3 Optimizing the tree

- **step 1:** Using APSO, the position and velocities are being initialized. Threshold values for the tree are the gbest value of these APSO algorithm.
- **step 2:** According to the fitness function, the particles are being evaluated.
- **step 3:** Updation of pbest, gbest are evaluated.
- **step 4:** Parameters are updated adaptively
- **step 5:** Change velocities, positions.
- **step 6:** If stopping condition is satisfied then stop the process, else repeat from step 2.

2.11.4 Simplification of rules:

- **step 1**Each cell is being represented by the binary digits, now all these are being sorted in the ascending order.
- **step 2** Cells that are having the same class are being combined.
- **step 3**If no chance of combining of cells, then it is stopped, else repeat from step 1.

2.12 Supervised and Unsupervised Learning

Algorithms are broadly classified into supervised, unsupervised.

- **Supervised Learning:** In this, the variables are being represented as an input-output pairs, i.e., there is a relation between those input's and output.

  For example, for 'n' training samples the pairs are being represented as- ((a_1,b_1), (a_2,b_2), ... , (a_n,b_n)), for i_{th} sample a_i is feature vector

and $b_i$is the class label, a function is being inferred as $F : A \to B$. Using this learning algorithm is being designed and evaluated on the unknown input.
Some of the algorithms of these include-

  - 1. Support Vector Machine (SVM)
  - 2. k-nn classifier
  - 3. Decision tree etc.

- **Unsupervised learning:** In unsupervised learning, unlabeled vectors are being present. There are few methods of these like -
  - 1. Cluster analysis
  - 2. Principal component analysis
  - 3. k-means clustering etc.

2.13 Classification

Task of assigning objects to one of the several known categories. Examples of the classification include:

- Loan repayment(good/bad customers)
- Medical diagnosis(disease detection)

2.13.1 Classification Techniques:

A few classification techniques include-

- Support Vector Machines(SVM)
- KNN-Classifier

2.13.1.1 Support Vector Machines(SVM)

It is a supervised machine learning algorithm used in classification and regression. For an 'n' number of features in an 'n' dimensional space, a point indicates each data item with the value referring to that particular feature coordinates. In this, the classification is being performed by selecting the suitable hyper-plane to differentiate the classes.
The data-points are also being termed as support vectors. In this the classification is being done in the following way-

- Finding the right hyper-plane which differntiates two classes better.
- Hyper-plane which has the highest margin(distance between hyper-plane to nearest vector).

Applications: SVM has applications like-
1.Used in image segmentation
2.Recogntion of hand written characters.
3.used in text categorization.

2.13.1.2 KNN-Classifier

This[4] is used for the both classification and regression problems.

The training data are being taken as the data points in the multidimensional space, each consisting of the known class label.

For the classification, a constant 'k' is being chosen and the unlabeled data is being classified by assigning the class label that is of the majority in the k-nearest neighbors.

In order to find the nearest neighbors we can use distance metric measures like Euclidean distance, Hamming distance .

Drawbacks:

Knn-classifier depends on the factor of the majority voting in those of 'k' nearest points. The main drawback is that majority of the k-class labels dominate the class label of the new unlabeled data point.

In order to avoid this the weight factor is being introduced, i.e., measure the distance from the unlabeled data point to all the nearest neighbors.

2.14 Performance of a classifier

Evaluating the performance of a classifier can be done in below ways[23]-

- Holdout method
- Random sampling
- Cross validation

2.14.1 Holdout method:

In this method [23], the given dataset is being divided into training, test datasets. A model is built by using training set and then it is evaluated on the test data. This division varies according to the work undertaken, it can be like 50-50, or like $\frac{2}{3}$ of the data for the training, remaining for test.
There is a disadvantage with method also as if any class has higher instances in one partition there may be a chance that only a few instances may be present in other partition. The size of the training set effects the performance of the classifier, it should not be too large or too small.

2.14.2 Random subsampling

If the previous method is run for a few times then it is being termed as random subsampling[23]. The final accuracy is being calculated as-

$$\sum_{i=1}^{k} \frac{acc_i}{k} \tag{2.11}$$

where acc_i is the accuracy at ith time.

2.14.3 Cross validation

In this method, each data is being used many times for the training and one time for testing. If the given data is divided into two equal partitions, one of these partitions is for training and other for test data. Next iteration these are being interchanged, this method is called 2-fold cross-validation. If this being done for 'K' times, it is termed as 'K-fold cross validation'.

2.15 Sensitivity and Specificity

For a two-class problem the sensitivity and specificity is being calculated as[24]-

Sensitivity:

It is the measure of actual positives that are exactly identified.
Sensitivity is being calculated as -

$$sensitivity = \frac{TP}{P} = \frac{TP}{TP + FN} \tag{2.12}$$

where-
TP is True Positive rate
P is Total number of positives
FN is False Negatives

- The other names include recall, true positive rate.

Specificity:

It is the measure of actual negatives that are exactly identified. Specificityis being calculated as -

$$sensitivity = \frac{TN}{N} = \frac{TN}{TN + FP} \tag{2.13}$$

where-
TN is True Negative rate
N is Total number of negatives
FP is False Positives
The other names include true negative rate.

2.16 Early warning system:

Normally classification is being done as-
1)Model construction from the given data.
2)Applying this model on the unseen instances.

But in the early warning systems like the decision tree induction and the rule-based learning, a model is being constructed just after the training set is given. They are also termed as eager learners, as they are eager to learn the model very early after providing the training data.

2.16.1 Rule generation

To implement early warning system, rule generation is being used. Here, we are going to propose a model for rule generation using an optimization technique called Particle Swarm Optimization(PSO). Each rule can be generated as-

$$r_i : (condition_i) \rightarrow y_i \tag{2.14}$$

The left-hand is: **rule-antecedent**
and right side : **rule-consequent** containing predicted class y_i

Example: Let take an example of a sample feature particle of an iris data as 1010

In this sample, we can notice that the feature f1, f3 are 1 , f2, f4 are 0 .It means that f1, f3 are being selected and f2, f4 are not selected. And the boundaries for these respective features are - L1, U1, L2, U2, L3, U3, L4, U4.

The rule generation for this can be written as-

$L1 \leq f1 \leq U1 and L3 \leq f3 \leq U3 \rightarrow$ Classlabel(y).

Chapter 3

Methodologies and Implementations

3.1 Proposed model

3.2 Description

Proposed model includes PSO is used for feature selection along with PCC and Binary-PSO is being used as a classifier for the rule generation on the obtained feature subsets. This model can be described in 3steps-
1:Initialization for outer and inner PSO
2: Feature Selection using PCC.
3: Classification by using rule generation.

Algorithm 2: Algorithm for Proposed Model

```
Input: Initializations of Outer and Inner PSO rom
       $Pop_{size}$:population size
       Count: size of training data
       $Max_{count}$: 15percent of the training data
       $Stop_{count}$: Stop if value doesnt change for 100 iterations)
       $S_{xtrain}$:size of the training data
Output: Final Feature subset and accuracy
//Begin algorithm
 Initialisation for outer PSO
 for $i := 1$ to $Pop_{size}$ do
     $X_{pos}(i, j)$=Randomposition(0,1)
     $V_{vel}$=Randomvelocity()
// End initialisation

 //Initialization for inner PSO
 for $j := 1$ to all features do
     //Find lower and upper limits of datasets
      //Initilise positions in this ranges
while $S_{xtrain} > Max_{count}$ and $Stop_{count} <= 100$ do
     //Outer PSO
      //Inner PSO
      //Update size of traindata, $stop_{count}$ values
Return $X_{gbest}$,max(acc);
 Return accuracy
 //end algorithm
```

3.2.1 Initialization of outer and inner PSO

Algorithm 2, is the initialization process of the proposed model. It contains the initialisation of outer and inner PSO like-

3.2.1.1 Initialization of outer PSO

This initialization step occurs only once in the proposed model. Positions of the PSO are being taken as the real numbers, which are present in the range of $[1, 2^n - 1]$, where 'n' is the number of features in the dataset. velocities ranging from [v_{min},v_{max}] used for velocity minimum and maximum limits, for

Algorithm 3: Algorithm for Outer PSO

```
Input:
        Feature=$f_1$,$f_2$.$f_3$,....,$f_n$
        $S_f$:Stop condition for Feature Selection
        $Pop_{size}$:Population size
        k: 'k' number of features to be in final Feature Subset
        Selection
Output: Final Feature subset
//Begin algorithm

 if $s_f$ <=10000 then
for $j := 1$ to $Pop_{size}$ do
    //Calculate Fitness function(PCC)
     //Features > k value
//update positions, velcocities
 //update $x_{gbest}$,$x_{pbest}$
 end if
Return $X_{gbest}$;

//end algorithm
```

any iterations if the velocity exceeds the maximum limit then it is limited to v_{max} .velocities are being randomly initialised in this range.

3.2.1.2 Initialization of inner PSO

In this initialization of inner PSO, lower and upper limit of the given dataset for every feature is being taken as [lower, upper]. Every feature in a particle has its lower and upper limit values which are been obtained from [lower, upper] values. These ranges are being used in the rule generation. The velocities for this PSO are also randomly initialized in the range of [lower, upper].

Algorithm 4: Algorithm for $PSO_{Classifier}$

```
Input:
        $X_{pos}$:Position values
        $S_{xtrain}$:size of the training data
        $S_f$:stopping condition
Output: accuracy
//Begin algorithm
 if $s_f <= 10000 then$
 $//increments_f$
 for $j := 1$ to $X_{pos}$ do
   // Rule-generation
//delete samples which are classified
and update $S_{xtrain}$
Extract best rules
 end if
 //Apply on test-data
 //Find accuracy
 Return accuracy
 //end algorithm
```

3.2.2 Feature Selection using PCC:

In algorithm 3, as these features may have redundancy, a correlation is being used to measure those relevant features. Here, in this proposed model, we are using a correlation coefficient known as PCC .

The correlation values for every pair of two features is greater than the threshold limit, then that pair of features are being redundant to each other. In order to reduce redundancy, either of the features is to be included by deleting inferior/posterior feature. In this proposed model we are going to delete the posterior feature. The objective/fitness function for this is being taken as PCC. Then the velocity and position updations are being done accordingly to the equations of PSO algorithm.

For example, if the feature (2,4) in a 4feature dataset after finding PCC, if the correlation value is greater than the threshold limit, then the posterior feature '4' is being deleted and only the feature '2' is being included.

3.2.3 Classification by rule-generation:

In algorithm 4, PSO as a classifier is being implemented by using rule-generation. Here, Binary PSO is being implemented which signifies the respective features to be selected (or) not. The feature that has to be selected/not is being decided by the feature subset selection done in the outer PSO. If 'pop' is the population size, 'n' is the number of classes, then pop/n partitions are being made, the first pop/n particles are used for rule generation for the first class, next pop/n particles for the next class till the last population size. An incremental count value is being maintained for each particle which is used for the rule generation–equation on the corresponding training data.

For example
IRIS dataset contains- 150 samples
4 features
3 classes each of 50samples (Iris-Setosa, Versicolor, Verginica)
let population size is : 30
According to rule generation, number of partitions=pop/n=30/3=10
. First 10[1...10] particles are going to classifiy class setosa, next [11...20] for class versiclor, next[21...30] for class verginica.
Objective/fitness function is the count for the respective particles that are going to classify on the training dataset. The best rules among the individual classes are taken as the best rules and saved. The training samples that are being classified are removed from the training dataset and the iterations for the outer, inner PSO are being performed until a stopping criterion of 15percent of the remaining training dataset is remainS. The best rules that are being obtained finally are used to classify on the test data and the accuracy is being calculated.

Chapter 4

Results and Discussion

4.1 Experimental Configuration

The experiment configurations are-

- MATLAB 2015b
- 64-bit windows system
- Intel i3, 2.40GHz processor
- 4 GB RAM
- Partition of size 30GB

4.2 Results

4.2.1 Test Objective Functions

PSO has been implemented and tested for a few test objective functions[26] like below-

- Rastrigin Function
- Matyas Function
- Square Function

- Rosenbrock Function

Parameters that are being used for these test objective are- w=0.5, [v_{min},v_{max}]=[-2,2],[c1,c2]=[2,2] maximum number of iteration is 50,

4.2.1.1 Rastrigin Function

Formula:

$$f(x) = A_n + \sum_{i=1}^{n} [x_i^2 - Acos(2\pi x_i)] whereA = 10 \tag{4.1}$$

Global minimum: f(0,0,...0)=0
Search Domain: $-5.12 \leq x_i \leq 5.12$

4.2.1.2 Matyas Function

Formula:

$$f(x, y) = 0.26(x^2 + y^2) - 0.48 * x * y \tag{4.2}$$

Global minimum: f(0,0)=0
Search Domain: $-10 \leq x \leq y \leq 10$

4.2.1.3 Square Function

Formula:

$$f(x) = \sum_{i}^{n} x_i^2 \tag{4.3}$$

Global minimum: $f(x_1, x_2, ..., x_n) = f(0, 0, ..., 0) = 0$
Search Domain: $-\infty \leq x_i \leq \infty$, $1 \leq i \leq n$

4.2.1.4 Rosenbrock Function

Formula:

$$f(x) = \sum_{i=1}^{n-1} [100(x_{i+1} - x_i^2)^2 + (1 - x_i)^2] \tag{4.4}$$

Global minimum:
Min(f(0,0)=0 for n=2 ,
f(0,0,0)=0 for n=3 ,
f(0,0,...,0)=0 for n$\geq$ 3)
Search Domain: $-\infty \leq x_i \leq \infty$, $1 \leq i \leq n$

Datasets	Features	instances	Classes	Training set	Test data
Iris	4	150	3	120	30
wine	13	178	3	125	53
Thyroid	5	215	3	151	64
Australian	14	690	2	483	207
German	24	1000	2	700	300
WBCD	30	569	2	399	170
Vehicle	18	846	4	593	253

Table 4.1: Datasets used in experiment.

4.2.2 Datasets

Datasets we used to evaluate the proposed model is taken from UCI-Repository. We used six-datasets in this experiments, the data is being divided into 70:30 ratio, where 70percent for the training and remaining for the test data.

Table 4.1 shows the description of the datasets that are being used in the proposed model.

4.2.3 Parameters used

The parameters used in this proposed model include- Population size is 30, value of ’w’ is set to 0.5. For **outer PSO**, the parameters included are - velocities are being taken as $[v_{min}, v_{max}]$ like [0,2], [0,4], [1,4], if the particle velocities exceeds the maximum velocity then it is limited to its v_{max} value and parameters c_1, c_2 values are taken from the different combinations from 1 to 4.

Inner PSO parameters include : population size,’w’ value is same as the outer PSO. The vmin, vmax for the inner PSO is being taken as the upper and lower bounds of the given dataset, between these ranges the velocities are being generated. For datasets like iris, wine, thyroid the maximum limit for while performing is at least 50percent of the features has to be present, for remaining datasets there is no minimum feature size limitation.

(v1,v2)	**threshold**	**Features**		**Accuracy**		
		min	**max**	**Min**	**Max**	**Avg**
(0,2)	0.5	1	1	37.77	**95.55**	64.57
	0.65	2	1	33.33	**95.55**	60.27
	0.78	1	0	24.44	**95.55**	57.59
	0.82	2	1	31.11	**95.55**	60.27
	0.9	1	2	17.77	**95.55**	52.46
(0,4)	0.5	2	0	40	88.88	64.5
	0.65	2	2	17.77	82.22	54.4
	0.78	2	1	17.7	82.22	54.4
	0.82	2	0	37.77	86.66	57.49
	0.9	2	1	8.88	86.6	57.49
(1,4)	0.5	2	0	0	93.33	63.87
	0.65	0	2	33.33	86.66	63.87
	0.78	1	1	40	91.11	60.51
	0.82	2	1	37.77	95.5	64.14
	0.9	2	1	31.11	86.66	55.55

Table 4.2: Results of IRIS dataset.

4.2.4 Analysis

4.2.4.1 IRIS dataset:

Description of dataset: Iris[6] belongs to the category of the benchmark datasets. Number of classes in iris dataset are 3-Iris Setosa, versicolor, verginica. 150 instances of the iris dataset are being divided equally i.e., 50instances each class. It has four attributes namely -sepal-length, sepal-width, petal width, petal-length.

From the table 4.2, it has the highest accuracy of **95.55** for all the categories related when (v1,v2)=(0,2). The number of features related to the highest classification accuracy is -(4, 4, 0, 1, (2,3))

(v1,v2)	**threshold**	**Features**		**Accuracy**		
		min	**max**	**Min**	**Max**	**Avg**
(0,2)	0.5	2	5	1.88	45.28	11.55
	0.65	2	7	1.8	43.39	13.12
	0.78	3	3	1.8	47.16	12.25
	0.82	2	3	1.8	30.18	8.36
	0.9	2	2	1.8	41.5	9.66
(0,4)	0.5	3	4	1.88	45.28	14.61
	0.65	2	3	1.88	30.18	10.01
	0.78	2	5	1.8	28.3	6.36
	0.82	4	4	1.8	32.07	7.65
	0.9	2	6	1.8	26.4	7.77
(1,4)	0.5	2	2	1.8	33.96	9.21
	0.65	2	3	1.8	9.43	4.11
	0.78	4	5	1.8	26.41	7.03
	0.82	4	4	1.8	50.94	10.72
	0.9	2	9	1.8	**56.6**	15.08

Table 4.3: Results of WINE dataset.

4.2.4.2 WINE dataset:

Description of dataset: Wine[6] belongs to the category of the benchmark datasets. It has three class dataset consisting of classes 1, 2, 3. The total instances of 178 are being divided as- class 1 contains 59 instances, class 2 of 71 and remaining i.e., 48 of class 3.

From the table 4.3, it has the highest accuracy of **56.66** when (v1,v2)=(1,4), threshold limit is 0.9. The number of features related to the highest classification accuracy is -(6, 7, 8, 12).

(v1,v2)	**threshold**	**Features**		**Accuracy**		
		min	**max**	**Min**	**Max**	**Avg**
(0,2)	0.5	2	3	1.56	50.0	10.93
	0.65	0	1	1.56	71.87	25.48
	0.78	2	2	1.56	43.75	12.00
	0.82	3	3	3.12	71.87	17.08
	0.9	3	0	1.56	79.68	24.02
(0,4)	0.5	4	3	1.56	76.56	23.92
	0.65	1	3	1.56	68.75	17.57
	0.78	1	3	1.56	**90.62**	17.08
	0.82	1	0	1.56	70.31	18.74
	0.9	4	5	1.8	31.25	11.77
(1,4)	0.5	2	2	1.56	59.37	13.66
	0.65	0	2	1.56	79.68	25.77
	0.78	1	1	1.56	70.31	19.13
	0.82	1	0	1.56	87.5	14.93
	0.9	1	3	1.56	62.5	19.23

Table 4.4: Results of THYROID dataset.

4.2.4.3 THYROID dataset:

Description: Thyroid[6] belongs to the category of the benchmark datasets. Number of classes of the thyroid database include 3classes namely- class1, class2, class3 , also called as normal class, hyper class, hypo class respectively. Normal class containes 150instances, while hyper of 35, hypo has 30 instances.

From the table 4.4, it has the highest accuracy of **90.62** when (v1,v2)=(0,4), threshold 0.78, (c1,c2)=(2,1), w=0.5. The number of features related to the highest classification accuracy is - (2, 3, 4, 5).

(v1,v2)	**threshold**	**Features**		**Accuracy**		
		min	**max**	**Min**	**Max**	**Avg**
(0,2)	0.5	7	9	0.33	17.66	6.76
	0.6	8	7	0.33	23.33	6.34
	0.7	7	9	0.33	13	6.88
(0,4)	0.5	9	12	0.33	**29.66**	8.65
	0.6	9	11	0.3	19.33	5.64
	0.7	8	7	0.33	17.33	5.99
(1,4)	0.5	6	12	0.33	19	6.08
	0.6	8	11	0.33	27.3	6.96
	0.7	8	11	0.33	20	4.61

Table 4.5: Results of GERMAN dataset.

4.2.4.4 GERMAN dataset:

Description: German[6] belongs to the category of the benchmark datasets. From the table 4.5, it has the highest accuracy of **29.66** when (v1,v2)=(0,4), threshold 0.5, w=0.68, (c1,c2)= (1,1). The number of features related to the highest classification accuracy is - (7, 8, 9, 10, 12, 13, 14, 18, 20, 21, 22, 24).

(v1,v2)	**threshold**	**Features**		**Accuracy**		
		min	**max**	**Min**	**Max**	**Avg**
(0,2)	0.5	6	6	2.41	33.33	16.07
(0,4)	0.5	2	6	6.28	34.78	21.91
(1,4)	0.5	6	6	0.48	**44.44**	19.98

Table 4.6: Results of AUSTRALIAN dataset.

4.2.4.5 AUSTRALIAN dataset:

Description: Australian[6] belongs to the category of the benchmark datasets. This is a binary class dataset. The total instances of 690 are being divided as - 383 for class1, 307 for class2.

From the table 4.6, it has the highest accuracy of **44.44** when (v1,v2)=(1,4), threshold =0.5, w=0.68, (c1,c2)=(3,2). The number of features related to the highest classification accuracy is -(5, 6, 11, 12, 13, 14).

(v1,v2)	**threshold**	**Features**		**Accuracy**		
		min	**max**	**Min**	**Max**	**Avg**
(0,2)	0.5	2	7	0.58	36.47	18.01
	0.65	4	6	0.58	37.64	8.0
	0.78	5	7	0.58	1.17	0.65
	0.82	7	10	0.58	12.35	1.31
	0.9	8	16	0.58	14.11	1.42
(0,4)	0.5	3	4	0.58	**60.58**	16.61
	0.65	2	7	0.58	31.17	6.46
	0.78	6	5	0.58	4.7	0.911
	0.82	6	12	0.58	1.76	0.80
	0.9	7	12	0.58	0.5	0.5
(1,4)	0.5	1	3	0.58	58.82	20.71
	0.65	4	6	0.58	27.64	5.20
	0.78	6	10	0.58	12	1.43
	0.82	6	10	0.58	4.7	0.91
	0.9	8	11	0.58	1.17	0.65

Table 4.7: Results of WBCD dataset.

4.2.4.6 WBCD dataset:

Description: WBCD[6] is termed as breast cancer dataset. It is a two-class datasets which containes class names as Malignant, Benign. Class distribution is 212 are Malignant class, remaining 357 are of benign .

From the table 4.7, it has the highest accuracy of **60.58** when (v1,v2)=(0,4), threshold 0.5, w=0.68, (c1,c2)=(1,1). The number of features related to the highest classification accuracy is (5, 14, 15, 17).

c **(v1,v2)**	**threshold**	**Features**		**Accuracy**		
		min	**max**	**Min**	**Max**	**Avg**
(0,2)	0.5	2	5	2.76	35.96	17.0
	0.65	5	5	0.39	44.66	13.56
	0.78	2	4	0.39	20.94	6.46
	0.82	4	5	0.39	11.06	3.13
	0.9	3	7	0.39	22.13	3.70
(0,4)	0.5	5	5	1.18	50.59	22.25
	0.65	2	5	3.16	46.24	17.73
	0.78	3	2	0.39	26.87	7.98
	0.82	3	5	0.39	13.43	5.23
	0.9	4	6	0.39	15.09	6.17
(1,4)	0.5	7	2	1.58	38.73	20.40
	0.65	4	4	0.09	**56.12**	18.68
	0.78	5	3	0.39	22.92	8.61
	0.82	3	4	0.39	10.67	3.91
	0.9	4	4	0.39	16.2	5.22

Table 4.8: Results of VEHICLE dataset.

4.2.4.7 VEHICLE dataset:

Description: Vehicle[6] dataset contains four-classes, where depending on the attributes the type of vehicle is being determined. classes and their distribution are- 240 for opel class, 240 for saab, 240 for bus class, remaining van class consists of 226.

From the table 4.8, it has the highest accuracy of **56.12** when (v1,v2)=(1,4), threshold =0.5, (c1,c2)=(2,3), w=0.68. The number of features related to the highest classification accuracy is -(9, 14, 15, 16).

Datasets	Features	Classes	Accuracy	Senisitivity	Specificity
Iris	4	3	95.5	-	-
wine	13	3	56.6	-	-
Thyroid	5	3	90.62	-	-
Australian	14	2	44.44	0.67	0.55
German	24	2	29.66	0.26	0.69
Vehicle	18	4	56.12	-	-
WBCD	30	2	60.58	0.67	0.55

Table 4.9: Sensitivity Specificity results.

4.2.5 Sensitivity Specificity results

Table 4.9 gives the results of sensitivity, specivicity for the binary class datasets.

Chapter 5

Conclusion and Future Scope

5.1 Conclusion

A study has been done on various techniques available for feature selection and classification. Among the available techniques, we have chosen PSO along with PCC for feature selection. Early warning system has been implemented by using rule generation in PSO-as-a-classifier. This proposed model is being validated on benchmark datasets that are taken from the UCI repository. The results are better for datasets like iris, thyroid and for other datasets results are comparatively less than the existing methodologies. PSO is also been implemented for various optimization techniques.

5.2 Future Scope

We have chosen UCI-repository datasets for the evaluation of the proposed model. In future, we will try to evaluate on other datasets related to banking and financial datasets that contain the huge number of records. In future, we also try to work on to improve the accuracy and also on the missing value datasets.

References

[1] Parham Moradi and, Mozhgan Gholampour, *“A hybrid particle swarm optimization for feature subset selection by integrating a novel local search strategy” Appl. Soft Comput.* Vol. 43, No.3,pp.117–130, 2016 .

[2] James Kennedy and, Russell C.Eberhart, *“A Discrete Binary Version of Particle Swarm Algorithm” In:Systems,Man and Cybernetcis,1997.Computational Cyber-netics and SImulation.*, .

[3] https://archive.ics.uci.edu/ml/index.php

[4] https://en.wikipedia.org/wiki/K-nearest$_{n}eighbors_{a}lgorithm$

[5] Fei W,Yi Y,Xianchao L,Jiao X and Lian L(2014) *Feature selection using Feature Ranking,Correlation Analysis and Chaotic Binary Particle Swarm Optimization.* pp:305-309, oct, 2014,Proceedings of the IEEE International Conference on Software Engineering and Service Sciences, ICSESS,.

[6] https://archive.ics.uci.edu/ml/index.php

[7] C.E. Shannon , A mathematical theory of communication, ACM SIGMOBILE Mob.Comput. Commun. Rev. 5 (1) (2001) 355 .

[8] Bing Xue and Mengjie Zhang and Will N. Browne *Single Feature Ranking and Binary Particle Swarm Optimisation Based Feature Subset Ranking for Feature Selection.* Thirty-Fifth Australasian Computer Science Conference, ACSC 2012, Melbourne, Australia, January 2012, 2012, pp. 27–36.

[9] Parham Moradi and, Mozhgan Gholampour, *“A hybrid algorithm using ant and bee colony optimization for feature selection and classification (AC-ABC Hybrid” Swarm and Evolutionary Computation.* Vol. 36,pp.27–36, 2017 .

[10] Zhi-hui Zhan and Jun Zhang and Yun Li and Henry Shu-Hung Chung, *“Adaptive Particle Swarm Optimization” IEEE Trans. Systems, Man, and Cybernetics, Part B* Vol. 39,pp.1362–1381, 2009 .

[11] Asit Kumar Das and Sunanda Das and Arka Ghosh, *"Ensemble feature selection using bi-objective genetic algorithm" Knowl.-Based Syst.* Vol. 123,pp.116–127, 2017 .

[12] Z. Pawlak , *"Rough set approach to knowledge-based decision support" Eur. J. Oper. Res. 99 (1)*pp.48–57, 1997 .

[13] Zhi-hui Zhan and Jun Zhang and Yun Li and Henry Shu-Hung Chung, *"Adaptive Particle Swarm Optimization" IEEE Trans. Systems, Man, and Cybernetics, Part B* Vol. 39,pp.1362–1381, 2009 .

[14] J. Kennedy and R. C. Eberhart, *"AParticle swarm optimization" Proc. IEEE Int. Conf. Neural Netw., Perth, Australia* Vol. ,pp.19421948, 1995 .

[15] David E. Goldberg and John H. Holland , *"Genetic Algorithms and Machine Learning" Machine Learning* Vol. 3 ,pp.95–99, 1988 .

[16] L. Ladla and T. Deepa , *"Feature Selection Methods And Algorithms" International Journal on Computer Science and Engineering (IJCSE)* Vol. 3(5) ,pp.1787-1797, 2011 .

[17] S. Khalid and T. Khalil and S. Nasreen, *"A survey of feature selection and feature extraction techniques in machine learning" 2014 Science and Information Conference* pp.372-378.

[18] Chi-Hyuck Jun and Yun-Ju Cho and Hyeseon Lee, *"Improving Tree-Based Classification Rules Using a Particle Swarm Optimization" Advances in Production Management Systems. Competitive Manufacturing for Innovative Products and Services - IFIP WG 5.7 International Conference, APMS 2012, Rhodes, Greece, September 24-26, 2012, Revised Selected Papers, Part II* ,pp.9–16, 20012 .

[19] Razieh Sheikhpour and Mehdi Agha Sarram and Sajjad Gharaghani and Mohammad Ali Zare Chahooki *"A Survey on semi-supervised feature selection methods" Pattern Recognition* Vol. 63,pp.141–158, 2017 .

[20] Cheng-San Yang, Li-Yeh Chuang, Jung-Chike and Cheng-Hong Yang *"Chaotic maps in Binary Particle Swarm Optimisation" Pattern Recognition* Vol. 63,pp.141–158, 2017 .

[21] Jacek Biesiada, and Wlodzislaw Duch *"Feature Selection for High-Dimensional Data: A Pearson Redundancy Based Filter" Computer Recognition Systems 2* ,pp.242–249, 2008 .

[22] Prashant Shrivastava,Anupam Shukla,Praneeth Vepakomma,Neera Bhansali,Kshitij Verma, *“A survey of nature-inspired algorithms for feature selection to identify Parkinson’s disease” Computer Methods and Programs in Biomedicine* ,Feb 2017 .

[23] https://www-users.cs.umn.edu/ kumar001/dmbook/ch4.pdf

[24] https://en.wikipedia.org/wiki/Sensitivity$_and_specificity$

[25] J. Chuanwen and E. Bompard *“A hybrid method of chaotic particle swarm optimization and linear interior for reactive power optimisation”Mathematics and Computers in Simulation*, vol. 68, pp. 57-65, 2005.

[26] https://en.wikipedia.org/wiki/Test$_functions_for_optimization$