

## Enhancing the Performance of Classifier Using Particle Swarm Optimization (PSO) - based Dimensionality Reduction

D. Asir Antony Gnana Singh, E. Jebamalar Leavline, K. Valliyappan  
and M. Srinivasan

*University College of Engineering, Bharathidasan Institute of Technology  
Campus, Anna University, Tiruchirappalli- 620 024*

*asirantony@gmail.com, jebi.lee@gmail.com, valliyappank@gmail.com,  
srinivasancse1994@gmail.com*

### **Abstract**

*Nowadays, the massive growth of data makes the data classification a challenging task. The feature selection is a demanding area to take this challenge and produce the higher accuracy in data classification by reducing the dimensionality of the data. Particle Swarm Optimization (PSO) is a computational technique which is applied in the feature selection process to get an optimal solution. This paper proposes a PSO and F-Score based feature selection algorithm for selecting the significant features that contribute to improve the classification accuracy. The performance of the proposed method is evaluated with various classifiers such as support vector machine (SVM), Naive Bayes, KNN and Decision Tree. The experimental results show that the proposed method outperforms the other methods compared.*

**Keywords:** *Dimensionality reduction, PSO, Feature selection, Classification*

### **1. Introduction**

Feature selection is an important task in data classification. It is very significant for the reason that it improves the accuracy of the classification process. Hence the contribution of feature selection to the field of pattern classification is at a greater scale. It helps in reducing the inputs for processing and analysis to a manageable size. Also, it facilitates efficient analysis of the given dataset. The feature selection algorithm is classified into three types namely wrapper, filter, and embedded approach. In the wrapper approach, the unsupervised learning algorithm is used as performance criteria with a searching algorithm. The searching algorithm generates the feature subsets based on any one of the searching strategies and these feature subsets are selected based on the unsupervised learning algorithm with a criterion. The filter approach first selects significant feature subset before application of any classification algorithm and removes least significant features from the given dataset. The embedded approach uses a part of the learning algorithms for selecting the features. It is less computationally expensive than wrapper algorithm. The embedded approach is the combination of filter and wrapper approaches.

Basically there are two types of optimization techniques that can be employed in the feature selection. They are deterministic algorithms and stochastic algorithms. The deterministic algorithms include approaches such as breadth first search (BFS), depth first search (DFS), gradient method, and etc. The stochastic algorithm generates random variables. For stochastic optimization, the random variables appear in the formulation of optimization problem itself, which involves random objective functions or random constraints. Hence, the outputs of these kinds of algorithms are not always constant. These algorithms include particle swarm optimization (PSO), genetic algorithm (GA), and ant colony optimization (ACO).

The breadth-first search is a graphical data structure. It traverses along the breadth of the tree. It can be employed in applications for finding the shortest path in a graph. The depth first search is a graphical data structure. It traverses along the depth of the tree. It has its applications in the areas of artificial intelligence and web crawling. The genetic algorithm is an evolutionary algorithm based on natural selection. It is used in optimization and search problems. It includes the genetic functions of inheritance, mutation, selection and crossover. Genetic algorithm is initialized with a random population. The fitness of every individual in the population is evaluated by a fitness function. The ant colony optimization algorithm was initially proposed to find the shortest path in the graph of travelling sales person problem. This algorithm is also an evolutionary algorithm. It was proposed based on the behaviour of ants. The ants travel from their nest in search of food. They find the path of the food using the pheromone deposited by other ants. Pheromone is a substance released into the environment by ants.

The remainder of this paper is organized as follows: Section 2 reviews the literature related to the proposed work, Section 3 describes the proposed work, Section 4 discusses the results and Section 5 concludes the paper.

## 2. Related Works

This section explores the research works related to our proposed method. Yuanning et al. proposed a feature selection based on PSO in order to improve the classification accuracy. They used the F-Score method for feature selection. Further, they described about the multiple swarm particle swarm optimization (MSPSO) for feature selection [1] in comparison with genetic algorithm and grid search. Alejandro et al. described the use of binary particle swarm optimization for classification (BSO). They have used the concept of association rules and patterns for classification using PSO. It also deals with Monk's problem in classification [2]. Laskari et al. proposed the use of PSO for minimization and maximization problems. The results of PSO are compared with sequential quadratic programming approach. It explains the ability of PSO to solve minimax problems in the field of Mathematics [3].

The authors Rabab *et al.*, used particle swarm optimization for face recognition. The feature selection methods used in this work are discrete cosine transform (DCT) and discrete wavelet transforms (DWT). The results of this paper show that PSO with feature selection gives higher classification accuracy in the face recognition system [4]. E. K. Tang *et al.*, depicted the use of particle swarm optimization in gene selection. This paper uses PSO with least square support vector machines for finding the classification accuracy. As this paper classifies genes, it has used microarray data such as Leukemia, Carcinoma and Golima datasets [5]. Yamille del Valle *et al.*, discussed different variants of PSO. They also described the use of PSO in power systems [6]. Jagdeep Kaur *et al.*, used particle swarm optimization for classification of remote sensing images. They employed PSO with clustering and uses kappa statistic for classification [7]. Barnali Sahoo *et al.*, illustrated the use of particle swarm optimization for classification of Microarray cancer data. Support Vector Machines, k-nearest neighbour and probabilistic neural network are used as evaluators for the optimized feature subset [8]. Tiago Sousa et al. proposed the use of particle swarm for data mining tasks such as classification. It compares the results with different forms of PSO such as Discrete PSO and Continuous PSO [9]. F. Van presented an empirical analysis of particle swarm optimization. It also includes theoretical analysis of particle swarm optimization [10]. From the review of the literature, it is identified that the PSO provides better solution to solve the problem of feature selection in real world applications. The F-score feature selection

measure is considerably better to adopt in the feature selection process to calculate the weight of the features based on their significance.

### 3. Proposed Work

The proposed method is implemented in two forms: One is with PSO and the other is PSO with F-score. The details of dataset and the methodology are described below.

#### 3.1. Details of Dataset

The datasets (Table. 1) are collected from various domains for the conduction of the experiment. The Ionosphere and Lung cancer dataset are collected from the UCI repository [11]. The Imports-85 is collected from the MATLAB sample dataset repository.

**Table 1. Details of the Dataset**

S.No	Name	No. of Features	No. of Instances	No. of Classes
1	Ionosphere	34	351	2
2	Lung cancer	57	32	2
3	Imports-85	56	205	2

#### 3.2. F-Score Calculation

F-Score is a metric to find the importance of a feature. The score of the features in the data set is calculated using the following formula:

$$F_i = \frac{\sum_{j=1}^l (\bar{x}_i^{(j)} - \bar{x}_i)^2}{\sum_{j=1}^l \frac{1}{n_j - 1} \sum_{k=1}^{n_j} (x_{k,i}^{(j)} - x_i^{(j)})^2}$$

where

$\bar{x}_i^{(j)}$  - Average of  $i$ th feature in  $j^{\text{th}}$  data set

$\bar{x}_i$  - Average of  $i$ th feature in the whole data set

$n_j$  - No. of instances in  $j^{\text{th}}$  data set

$x_{k,i}^{(j)}$  -  $i^{\text{th}}$  feature of  $k^{\text{th}}$  instance in  $j^{\text{th}}$  data set

#### 3.3. PSO Algorithm

Particle swarm optimization is an optimization technique that finds a solution by several iterations. It works by a population consisting of a set of particles. Each particle is associated with a position and a velocity. The position and velocity of the particles are updated using simple mathematical formulae. It contains 2 best positions known as local best and global best. Local best is the best position of the current particle and global best is the position of the overall particles. It is a meta-heuristic approach. It means that it finds a lower level method to find the solution to

the given optimization problem. Meta-heuristic optimization follows the stochastic approach for optimization.

**Table 2. Parameters and their Initialized Values used in the Algorithm**

Parameter	Value
Population size	20-30 (Empirical Analysis)
No. of particles	No. of features in the data set
Inertia coefficient-w	0.8 (Empirical Analysis)
c <sub>1</sub>	2 (Empirical Analysis)
c <sub>2</sub>	2 (Empirical Analysis)
Maximum Number of Iteration	According to the problem
Position	Randomly generated from Population Size and No. of Particles
Velocity	Randomly generated from Population Size and No. of Particles
Xp	Fitness of current particle
pbestp	Local best position of particles-Initially zero
gbest	Global best position of the overall particles

### 3.4. Calculation of the Fitness Function

The work starts with particle swarm optimization and chooses the features using F-Score metric. Then the dataset with the selected subset of features is passed to different classifiers. Hence it can be noted that the proposed method is based on the filter algorithm of classification. The classification accuracy is found for a set of selected features. The fitness function in PSO is calculated using the following formulae:

$$fitness_i = \theta_a \times \frac{\text{positive}}{\text{ratio}} + \theta_b \times \left[ \frac{\sum_{j=1}^{N_b} F(FS(i))}{\sum_{k=1}^{N_b} F(k)} \right]$$

Where  $\theta_a$ - Weight for features based on F-Score. The features with greater F-Score gets a greater weight,

$\theta_b$ - Weight based on the instance of the class. If the class instance is positive it is 1 otherwise it is zero

Positive ratio - Ratio of no. of positive class instances in the data set to the total no. of instances in the data set

F(FS(i))- Function for calculating the score of the selected features

F(F(k))- Function for calculating the score of all the features

### 3.5. Proposed Algorithm:

**Initialize Population** // Population Size, No. of particles, position, velocity, local best, global best & maximum number of iterations

```
while(maximum iteration no. is not met)
    for each member in population
        for each particle
```

```
Evaluate the fitness function
  if fitness of Xp > fitness of pbestp
    Update pbestp=Xp
  end if
  if fitness of Xp > gbest
    Update gbest=Xp
  end if
  for each member in population
    for each particle
      Updated Velocity  $v = w * \text{old velocity} + c1 * \text{rand1} * (\text{pbestp} - \text{xpd}) + c2 * \text{rand2} * (\text{gbest} - \text{xpd})$ 
       $\text{xpd} = \text{xpd} + \text{vpd}$ 
      
$$\text{sol} = \frac{1}{1 + e^{-v}}$$

      if rand() < sol
        posn=1
      else
        posn=0
      end if
    end for
  end for
end while
```

## 4. Experimental Procedures

In this proposed method, two experimental procedures are carried out as explained in the following sections.

### 4.1. Experimental steps for PSO with F-Score

1. Collect the dataset.
2. Calculate the scores of each feature using F-Score formula.
3. Set the threshold value for F-Score as the mean value of F-Score of all the features.
4. Select the features that are above the threshold value.
5. Execute Particle Swarm Optimization algorithm with the selected set of features.
6. PSO algorithm further optimises the features and selects a subset of features.
7. The selected subset of features is evaluated using evaluators such as support vector machine, Naïve Bayes, K-nearest neighbor and decision tree.
8. The classification accuracy is found for each set of evaluators.

### 4.2. Experimental steps for PSO without F-Score

1. Collect the dataset.
2. Calculate the scores of each feature using F-Score formula.
3. Set the threshold value for F-Score as the mean value of F-Score of all the features.
4. Select the features that are above the threshold value.
5. Then PSO algorithm is again executed without selecting any features.
6. It optimises the features and a subset is selected.
7. This subset of features is also sent to the same set of evaluators.
8. The classification accuracy is calculated for each set of evaluators.

## 5. Results and Discussion

In order to evaluate the performance of the proposed method in terms of classification accuracy, the experiment is conducted using three classifiers namely support vector machine (SVM), Naive Bayes (NB), k-nearest neighbour (k-NN), and decision tree with the three datasets Ionosphere, Lung Cancer, and Imports and the results are obtained and shown in the Figure 1.

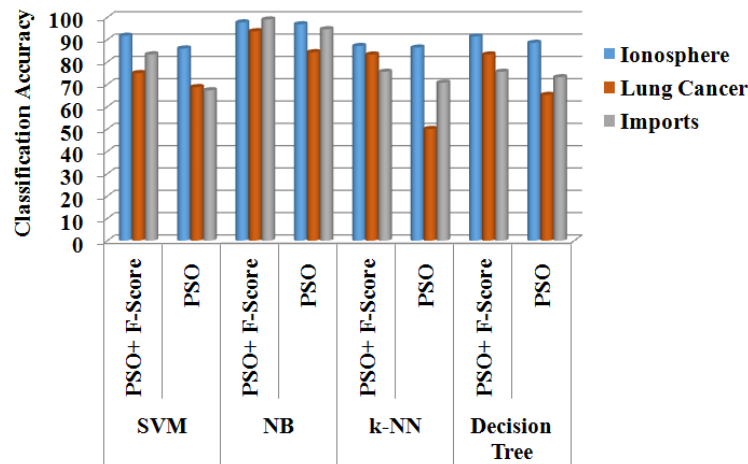


Figure 1. Performance Comparison of the Proposed Method

## 6. Conclusion

In this paper, particle swarm optimization-based feature selection is proposed for improving the performance of the classifiers in terms of the classification accuracy. This method improves the classification accuracy compared to other method. This proposed method employs the particle swarm optimization and the F-score feature selection metrics. This work can be extended with combining other optimization technique to improve the performance of the classifiers.

## References

- [1] Y. Liua, G. Wanga, H. Chena, H. Donga, X. Zhua and S. Wanga, "An improved particle swarm optimization for feature selection", *Journal of Bionic Engineering*, vol. 8, no. 2, (2011), pp. 191-200.
- [2] A. Cervantes, I. M. Galván and P. Isasi, "Binary particle swarm optimization in classification", (2005).
- [3] E. C. Laskari, K. E. Parsopoulos and M. N. Vrahatis, "Particle swarm optimization for minimax problems", *Proceedings of the IEEE Congress on Evolutionary Computation*, Honolulu, HI, (2002), pp. 1576-1581.
- [4] R. M. Ramadan and R. F. Abdel-Kader, "Face recognition using particle swarm optimization-based selected features", *International Journal of Signal Processing, Image Processing and Pattern Recognition*, vol. 2, no. 2, (2009), pp. 51-65.
- [5] E. Ke Tang, P. N. Suganthan and X. Yao, "Feature selection for microarray data using least squares svm and particle swarm optimization", *Proceedings of the IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology*, (2005) November 14-15, pp. 1-8.
- [6] Y. Del Valle, G. K. Venayagamoorthy, S. Mohagheghi, J. C. Hernandez and R. G. Harley, "Particle swarm optimization: basic concepts, variants and applications in power systems", *IEEE Transactions on Evolutionary Computation*, vol. 12, no. 2, (2008), pp. 171-195.
- [7] J. Kaur and K. Kaur, "Remote Image Classification Using Particle Swarm Optimization", *International Journal of Emerging Technology and Advanced Engineering*, vol. 2, no. 7, (2012), pp. 470-473.
- [8] B. Sahu and D. Mishra, "A novel feature selection algorithm using particle swarm optimization for cancer microarray data", *Procedia Engineering*, vol. 38, (2012), pp. 27-31.
- [9] T. Sousa, A. Silva and A. Neves, "Particle swarm based data mining algorithms for classification tasks", *Parallel Computing*, vol. 30, no. 5, (2004), pp. 767-783.
- [10] F. Van den Bergh and A. Petrus Engelbrecht, "A study of particle swarm optimization particle trajectories", *Information sciences*, vol. 176, no. 8, (2006), pp. 937-971.

- [11] M. Lichman, "UCI Machine Learning Repository", [<http://archive.ics.uci.edu/ml>]. Irvine, CA: University of California, School of Information and Computer Science, (2013).

## Authors



**D. Asir Antony Gnana Singh** is with Department of Computer Science and Engineering Bharathidasan Institute of Technology, Anna University, Tiruchirappalli, Tamil Nadu, India. He received his B.E and M.E degree in Computer Science and Engineering from Anna University, Chennai, Tamil Nadu, India in 2005 and 2008 respectively and currently pursuing Ph.D at Anna University, Chennai, India. He has published a good number of research papers in International Journals and conferences. His research interests are data mining, wireless networks, parallel computing, mobile computing, computer networks, image processing, software engineering, soft computing, teaching learning process and engineering education. He has participated and co-coordinated the faculty development programmes in India and abroad. He is a Review/Editorial committee member of several reputed journals and also he is a Member of ISTE, IET and IAENG.



**Dr. E. Jebamalar Leavline** is with Department of Electronics and Communication Engineering, Bharathidasan Institute of Technology, Anna University, Tiruchirappalli, Tamil Nadu, India. She received her B.E, Electronics and Communication Engineering in 2005 and M.E in Applies Electronics in 2008, and Ph.D in 2015 from Anna University, Chennai, Tamil Nadu, India. Also she is a M.B.A graduate in Education Management. She has published a good number of research papers in reputed International Journals and conferences. Her research interests includes image processing, signal processing, VLSI design, data mining, teaching learning process and engineering education. She is a Review committee member of several reputed journals and also he is a Member of ISTE and IAENG.

**K. Valliyappan and M. Srinivasan** are Under Graduate (B. E) students of Department of Computer Science and Engineering Bharathidasan Institute of Technology, Anna University, Tiruchirappalli, Tamil Nadu, India.

