ISSN 2229-5518

Classification using Modified PSO Incorporating Probabilistic MH for Improved Convergence

M. Balasaraswathi, B. Kalpana

ABSTRACT
Classification has its roots deep inside the area of prediction and analytics. Performing effective classification can provide huge advantages and improvements in several area. The huge downsides are the unavailability of enhanced techniques and the huge time consumption associated with several machine learning techniques. This paper presents a metaheuristic method based on a modified form of PSO that tends to provide better accuracies within acceptable time limits.

Keywords
Particle Swarm Optimization (PSO); Classification; Simulated Annealing; Metaheuristics

1. INTRODUCTION

Particle Swarm Optimization (PSO) [4,5] is a metaheuristic optimization technique that helps solve optimization problems to provide near optimal results in specified time. This method tries to iteratively improve candidate solutions with respect to a give measure that defines the quality. A group of candidate solutions are considered and the system tries to find the best possible solution from them. The advantage of this approach is that it makes very few assumptions regarding the search space, hence is computationally less intensive [6,7].

Classification [8, 9] is the process of identifying the class to which a specific entity belongs to, given a set of classes. This process being an NP Complete problem, has its own specific time complexities associated with it. Hence it becomes mandatory to utilize an algorithm that is time constrained, provided the near optimal solutions can be compromised. Though PSO is an excellent candidate for Classification, very few research reports have been published in this area [3, 15]. A comparison of PSO, genetic algorithm and tree induction algorithm for classification has been reported in [10], while another similar comparison with K-means algorithm can be found in [13]. Apart from text, PSO has also been used for image classification [11]. Most of the used cases of PSO are usually limited to hybrid approaches, in which PSO is combined with several other techniques to perform the process. One of the mostly used hybrid technique is the usage of PSO for predicting weights in a neural network [12]. Several medical applications also use PSO for its minimal requirements and effective performance. Some of them include [16-19], in which PSO is used to aid thoracic surgery.

- *Balasaraswathi M is currently pursuing PhD Degree program and working as Associate Professor in SNR SONS College, Department of Information Technology, India, PH-0919842160797, E-mail:baladars@yahoo.co.in.*
- *Dr.Kalpana B is currently working as Professor in Department of Computer Science in Avinashilingam Institute of Home Science, India, PH-0919486447430. E-mail:kalpanabsekar@gmail.com*

Due to its dynamically adaptive nature of PSO, it is also used in several fuzzy dynamic systems [20-22]. This paper provides such a solution by using a modified form of PSO that provides better and more accurate convergence levels.

2. MODIFIED PSO INCORPORATING PROBABILISTIC MH: WORKING ARCHITECTURE

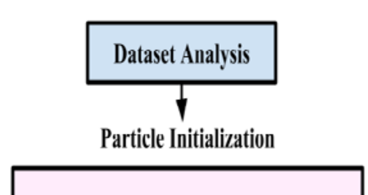
Particle swarm optimization algorithms usually operate on continuous space, but the problem of classification is discrete, hence the PSO proposed in this paper is modified to operate on discrete domain [14]. Figure 1 shows the proposed architecture for PSO based classification. The particles are initialized and they are set to acceleration. Every iteration provides the users with a local best solution. The global best solution is calculated after obtaining all the local best solutions using the metaheuristic technique of Simulated Annealing. This process is continued until termination condition is reached and the final accuracy of the method is identified.

3. CLASSIFICATION USING MODIFIED PSO INCORPORATING PROBABILISTIC MH FOR IMPROVED CONVERGENCE

Particle Swarm Optimization (PSO) is a metaheuristic technique that helps to solve optimization problems. Classification, being one such issue, can be solved effectively by PSO. This paper presents a modified form of PSO that modifies the selection of the global best value, hence providing effective convergence.

3.1. Data Analysis and Cleaning

The first phase of the classification process is data analysis, cleaning and pruning. Raw data is examined and converted to appropriate formats for use by the application. In general, raw data is prone to inconsistencies and missing values. The data is analyzed and appropriate replacements and eliminations are carried out to provide clean data to the PSO algorithm.



3.2. Particle Initialization

After the search space is built, the first process is to initialize the position of the particles for processing. This initialization is carried out on random basis. This contribution uses uniform distribution to disperse the particles in the search space. The particle best (pbest) of all the particles are set to this initial value. Further, it is also mandatory to calculate the initial velocity for accelerating the particle. This is also performed on random basis.

$$V_i \sim U(-|b_{up} - b_{lo}|, |b_{up} - b_{lo}|)$$

The default global best (gbest) is set to null to make sure that this value does not create an impact on the system until a legitimate global best value is obtained.

3.3. Particle Movement Triggering

The particle's initial location and the initial velocity are used to trigger the particle's movement. PSO intrinsically works in the continuous domain. Since our problem of classification require the domain to be discretized, the particle's current location is used to identify the node closest to the current location and the particle is shifted to the new location that constitutes a node.

$$P' = \min\left(\sum_{j=1}^{n}\left(\sum_{k=1}^{d}\sqrt{\left(P_{ik} - N_{jk}\right)^2}\right)\forall i = 1\ top\right)$$

Where Pik refers to the particle i's current location corresponding to dimension k, Njk refers to the k th dimension of node Ni .

This process is repeated for all the particles and this marks the end of a single transition in the search space. After the initial transition, the global best value is identified using Simulated Annealing, a metaheuristic method. The global best value is selected from the set of available particle best values. After the identification of a global best value, the process of calculating the velocity is calculated by

$$V_{i,d} \leftarrow \omega V_{i,d} + \varphi_p r_p\left(P_{i,d} - X_{i,d}\right) + \varphi_p r_p\left(g_d - X_{i,d}\right)$$

Where rp and rg are the random numbers, Pi,d and gd are the parameter best and the global best values, xi,d is the value current particle position, and the parameters ω, φp, and φg are selected by the practitioner.

This process is continued until time elapse or until specified number of iterations is completed without any change to the optimal value.

3.4. Global Best Identification

The process of identifying the global best value is usually carried out as every particle completes its movement and reaches a standard node. This contribution alters that scheme by identifying the global best only after all the particles complete their first hop movement. This provides a set of particle best solutions from which the best solution can be selected to be the global best. The process of selecting the global best is performed using Simulated Annealing [1].

3.5. Convergence Improvisation Using Simulated Annealing

Simulated Annealing is a probabilistic method that provides effective optimization by obtaining the best solution from a large search space. It has its roots in the area of metallurgy and works on the basis of thermodynamic flow. The major advantage of Simulated Annealing is that it has very low probability of getting struck in the local optima.

International Journal of Scientific & Engineering Research, Volume 6, Issue 8, August-2015 1208
ISSN 2229-5518
The process begins with a single randomly selected solution. It moves in a random manner for a defined number of steps or until a solution with the best result is obtained. This constrains both the time and also makes it certain that the operations are carried out in a random that helps us avoid getting struck in the local minima.

4. RESULTS AND DISCUSSION

The technique of embedded Particle Swarm Optimization (PSO) was implemented using C# and tested using six datasets from KEEL repository [2]. Details about the dataset are provided in Table 1. The application was executed and results obtained from several iterations were used for calculating the TPR, FPR, Precision and Recall values. These calculated values were used to plot the graphs.

Table I
Dataset Details

Name	Attributes	Instances	Classes	Imbalance Level
Iris	4	150	4	2
Bupa	6	345	2	-
Heart	13	257	2	-
Sonar	60	208	2	-
Libras-movement	90	360	15	-
Shuttle	9	58000	7	13.8

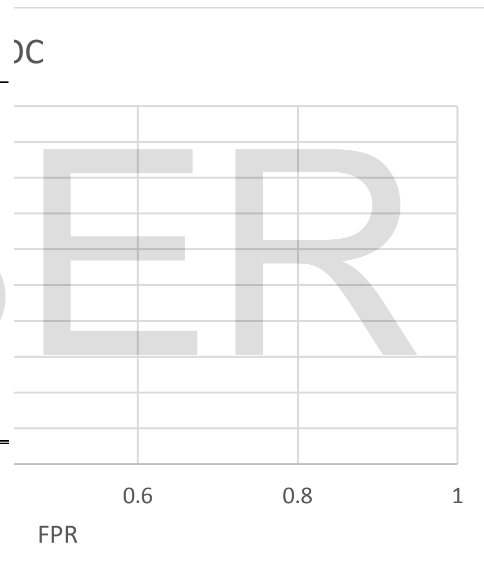


Fig 2: Libras Movement : ROC Curve

Figures 2,3 shows the ROC and PR plots of the Libras Movement prediction dataset. It could be observed from the graphs that perfect 100% accuracy is obtained in both the ROC and the PR curves. It could be observed that the ROC plot shows coordinate of (0,1) in which all the results are collected, while the PR curve shows the result of (1,1), where all its points are collected. The points (0,1) and (1,1) represent the best points for the ROC and PR plots respectively. Similar accuracy results can be observed in the Iris dataset.

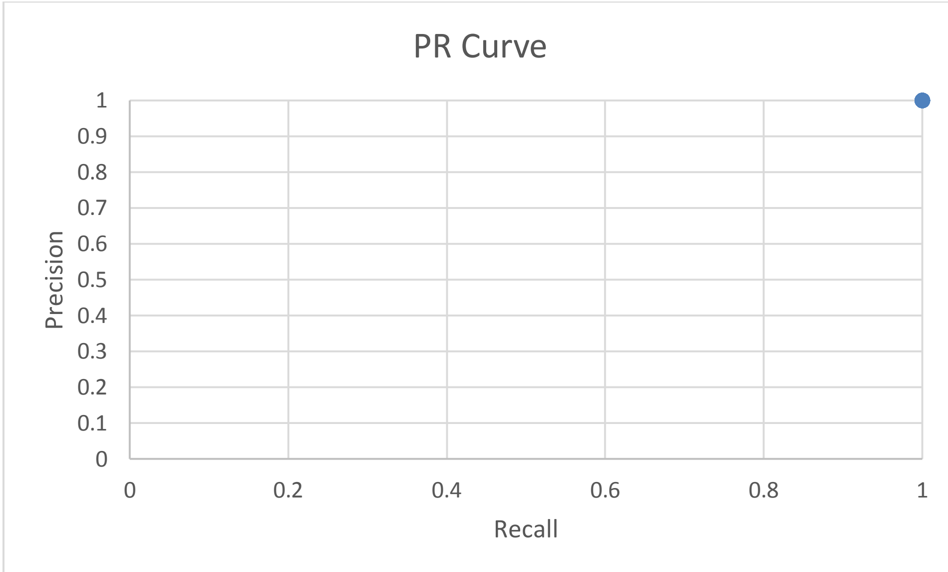


Fig 3: Libras Movement : PR Curve

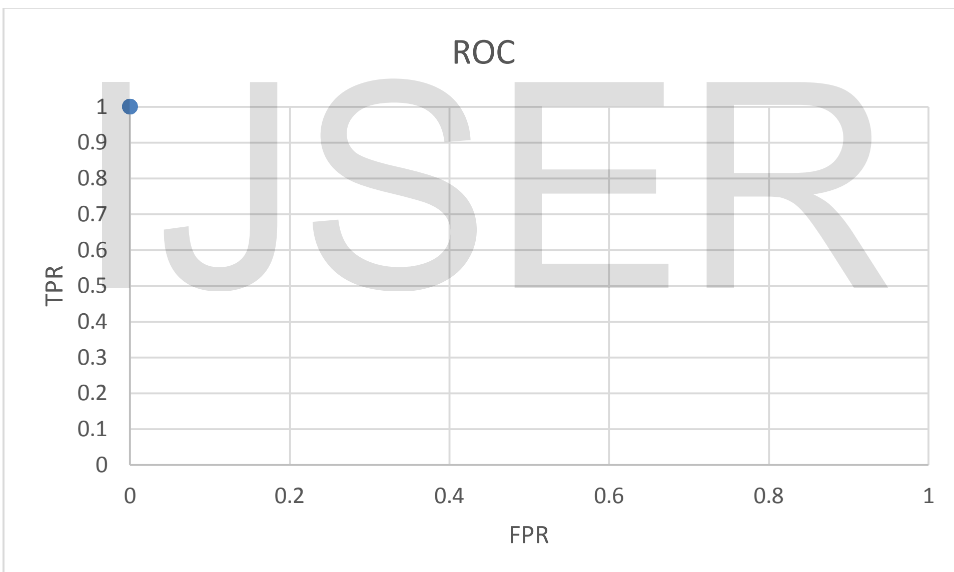


Fig 4: Iris : ROC Curve

The Figures 6,7 show ROC and PR values for the Bupa dataset. It could be observed from the ROC curve that the points are totally clustered in coordinates (1,0) and (1,1). From the data generated, it was observed that 90% of the points were grouped in the coordinate (1,1). Hence it could be implied that our method, though it provides accurate results, it is also prone to provide false positives.

ISSN 2229-5518

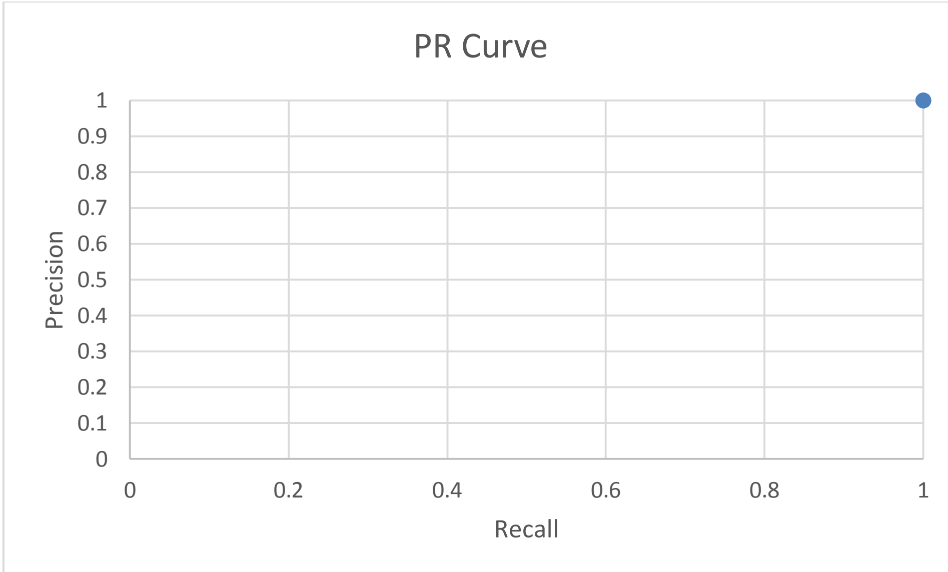


Fig 5: Iris: PR Curve

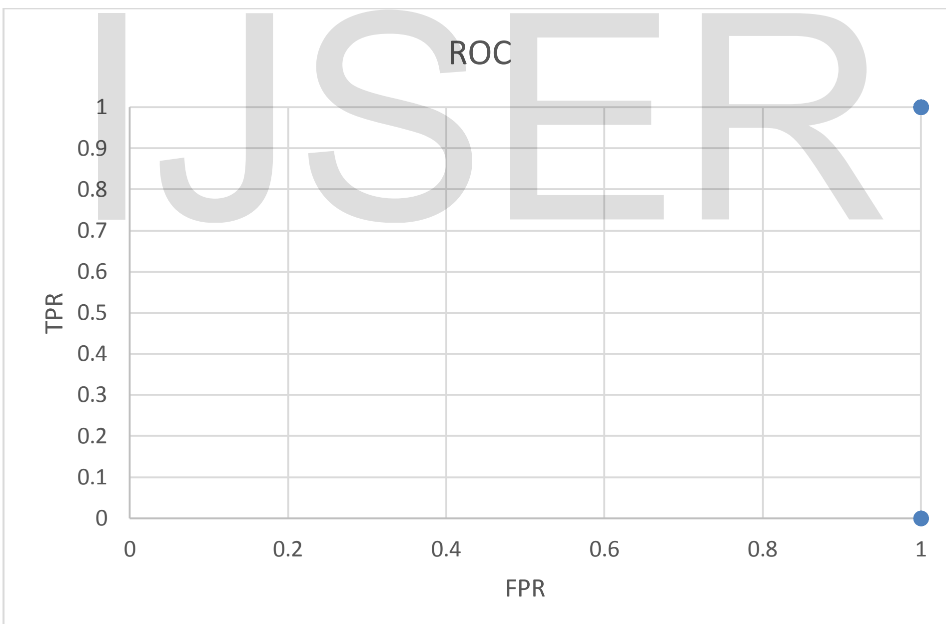


Fig 6: Bupa : ROC Curve

From the PR curve, the points could be observed to move towards the (1,1) co-ordinate, which shows that the accuracy of the prediction system is high and still has areas for improvisation

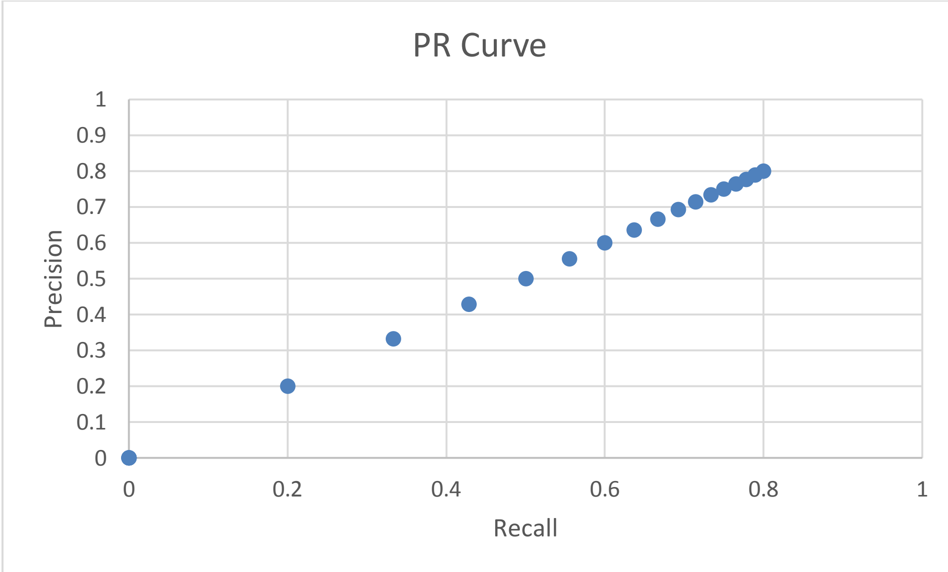


Fig 7: Bupa: PR Curve

Figures 8, 9 show the ROC and PR curves of the shuttle dataset. A similar scenario as observed in the Bupa data set can also be observed here.

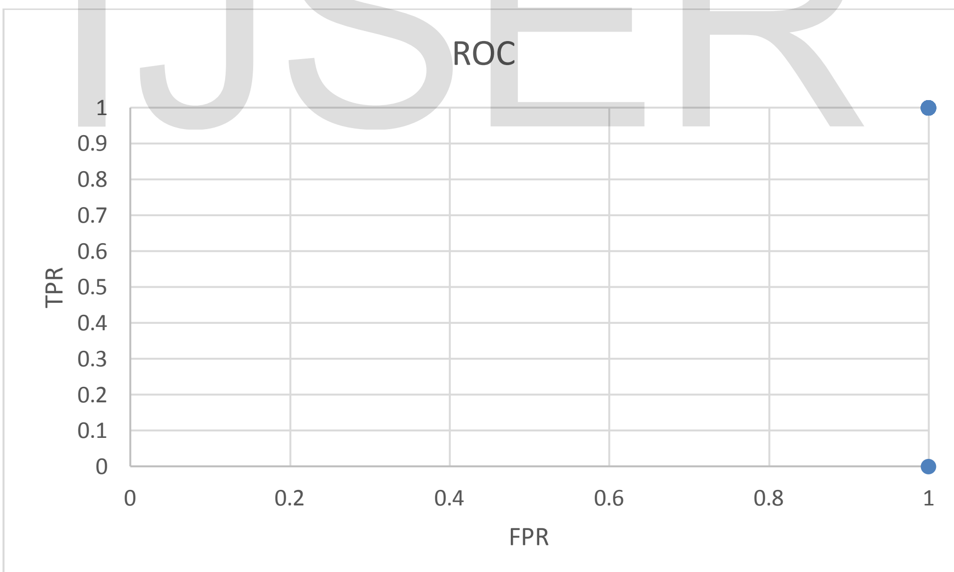


Fig 8: Shuttle : ROC Curve

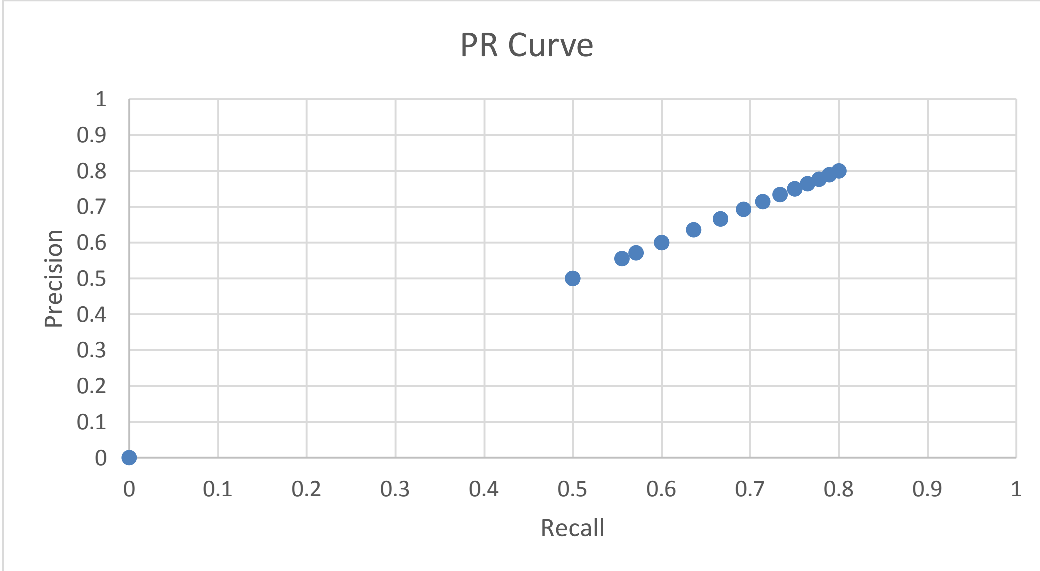


Fig 9: Shuttle: PR Curve

Figures 10, 11 show the ROC and PR plots for heart dataset. It could be observed that although they provide higher accuracies, the false positive levels are also higher. The PR curve also shows scope for improvement.

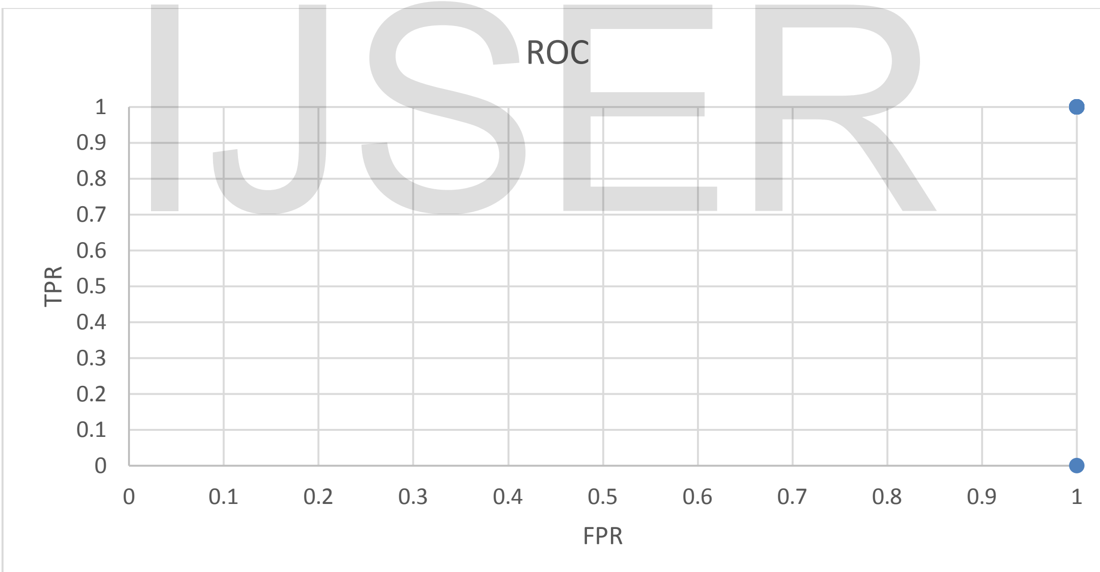


Fig 10: Heart : ROC Curve

ISSN 2229-5518

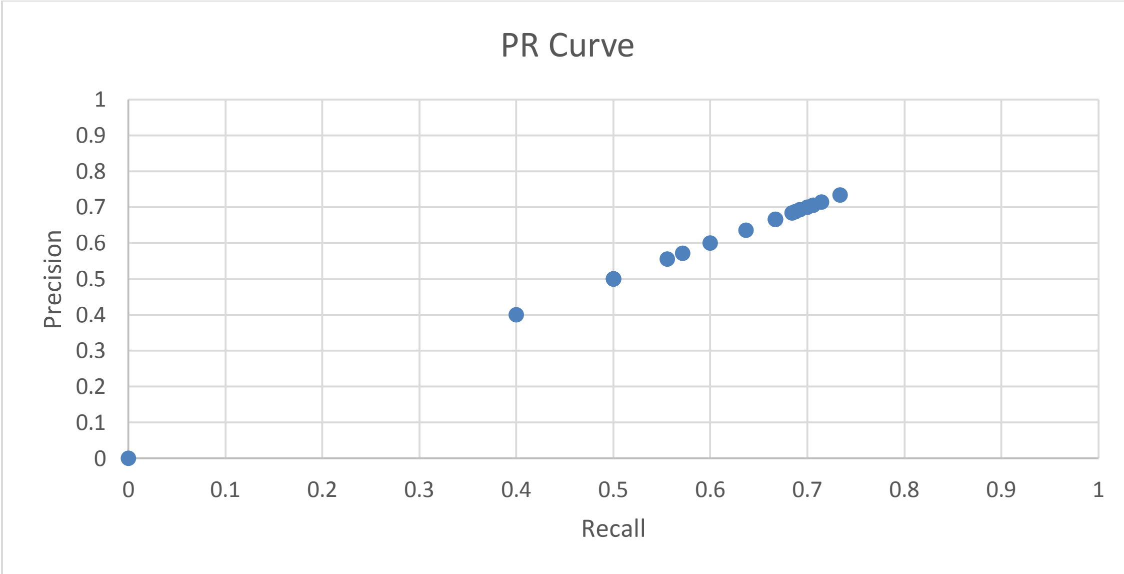


Fig 11: Heart: PR Curve

Figures 12, 13 show the ROC and PR plots of the Sonar dataset. It could be observed from the graphs that the points are collected to the coordinates (0,1) in the ROC plot and (1,1) in the PR plot, depicting good accuracies.

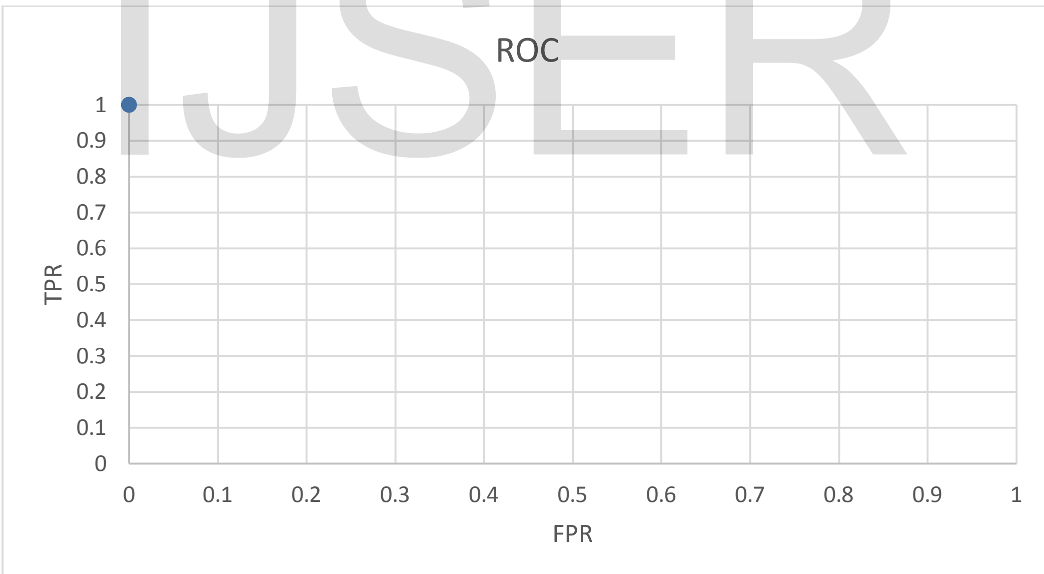


Fig 12: Sonar : ROC Curve

ISSN 2229-5518

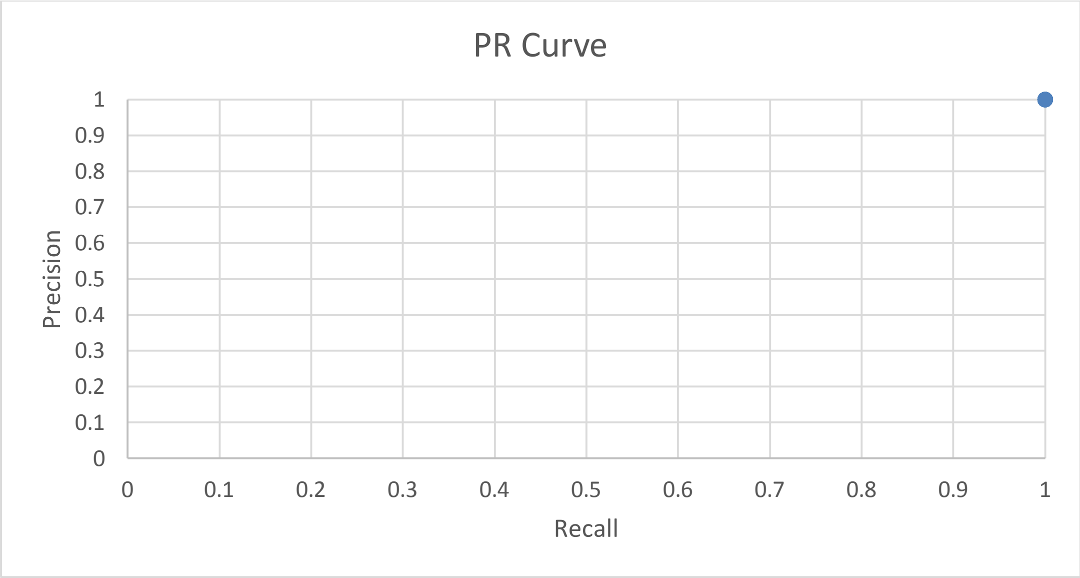


Fig 13: Sonar: PR Curve

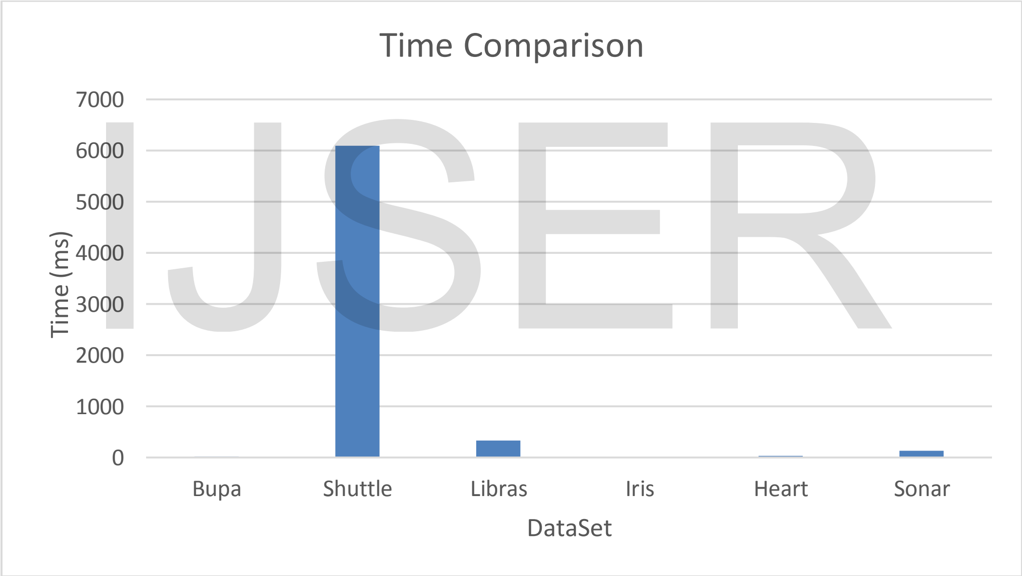


Fig 14: Time Comparison

Figure 14 shows the time taken for each of the datasets to complete their operation. It could be observed that the time taken for the dataset of maximum size (shuttle with 58000 instances and 9 attributes) takes a maximum of 6 seconds. All other datasets, which are considerably smaller takes less than a second.

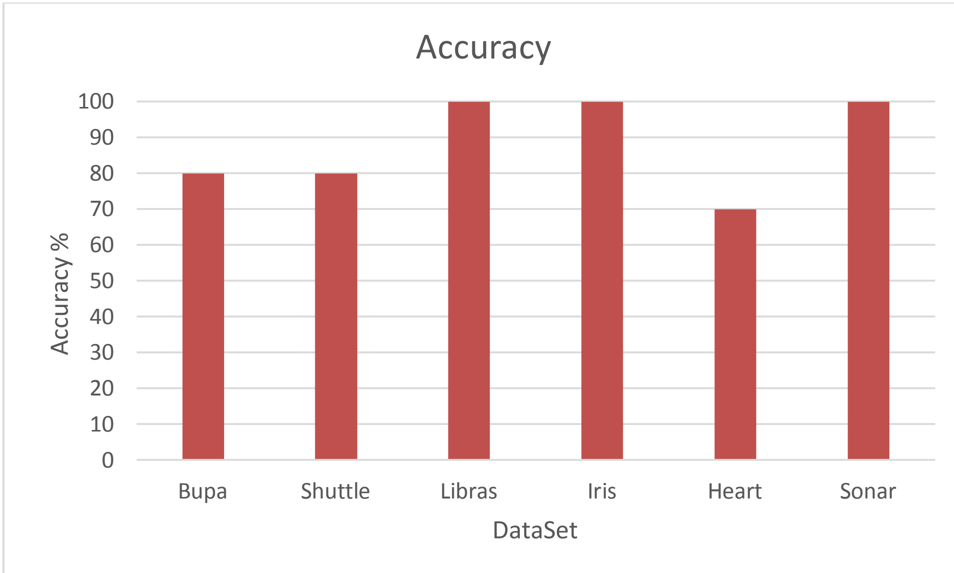


Fig 15: Accuracy

Figure 15 shows the accuracy levels obtained from each of the datasets. The accuracy levels range from 70% (heart) to a maximum of 100% (Libras, iris and sonar).

5. CONCLUSION

This paper presents an effective solution for Classification problem using a modified form of Particle Swarm Optimization incorporated with Simulated Annealing to eliminate the problem of getting struck in the local minima. The major advantage of this method includes faster processing and acceptable accuracies even for large datasets. Another major advantage of this approach is that it works effectively for both binary and multiclass classifiers. Slight accuracy deficits can be observed in some datasets, which is attributed to the imbalance nature of these datasets. Our future contributions will be based on countering the imbalance to provide effective results in acceptable time.

REFERENCES

[1] Kirkpatrick, S., Gelatt Jr, C. and Vecchi, M. P. 1983, Optimization by Simulated Annealing, Science 220(4598):671–680.

[2] http://sci2s.ugr.es/keel/category.php?cat=clas#sub2

[3] De Falco, I., Della Cioppa , A. and Tarantino, E. 2007. "Facing classification problems with Particle Swarm Optimization", *Applied Soft Computing*, Volume 7, Issue 3,Pages 652–658.

[4] Kennedy, J. and Eberhart, R. 1995. "Particle Swarm Optimization", *Proceedings of IEEE International Conference on Neural Networks IV*, pp. 1942–1948, doi:10.1109/ICNN.1995.488968.

[5] Shi, Y. and Eberhart, R.C. 1998. "A modified particle swarm optimizer", *Proceedings of IEEE International Conference on Evolutionary Computation*, pp. 69–73.

[6] Eberhart, R.C. and Kennedy, J. 1995. "A new optimizer using particle swarm theory", *in: Proceedings of the Sixth International Symposium on Micro Machine and Human Science*, IEEE Press, Piscataway, NJ, 1995, pp. 39–43.

[7] Eberhart, R.C., Shi, Y. 2003. Computational Intelligence: Concepts to Implementations, Morgan Kaufmann.

[8] Han, J. and Kamber, M. 2001. Data Mining: Concept and Techniques, Morgan Kaufmann.

[9] Hand, D.J., Mannila, H. and Smyth, P. 2001, Principles of Data Mining, The MIT Press.

[10] Sousa, T., Silva, A. and Neves, A. 2004. "Particle swarm based data mining algorithms for classification tasks", *Parallel Comput*. 30 (5/6)767–783.

[11] Omran, M., Salman, A. and Engelbrecht, A.P. 2002. "Image classification using Particle Swarm Optimization", *in: Proceedings of the Fourth Asia–Pacific Conference on Simulated Evolution and Learning, Singapore*, pp. 370–374.

[12] Eberhart, R.C. and Shi, Y. 1998. "Evolving artificial neural networks", *in: Proceedings of the 1998 International Conference on Neural Networks and Brain*, Beijing, China, pp. 5–13.

[13] Van der Merwe, D.W. and Engelbrecht, A.P. 2003. "Data clustering using Particle Swarm Optimisation", *Proceedings of the IEEE Congress on Evolutionary Computation*, IEEE Press, Piscataway, NJ.

[14] Kennedy, J. and Eberhart, R.C. 2001. Swarm Intelligence, Morgan Kauffman.

[15] Tiago Sousa ,Arlindo Silva and Ana Neves. 2004. "Particle Swarm based Data Mining Algorithms for classification tasks", *Parallel Computing*, Volume 30, Issues 5–6, Pages 767–783.

[16] Cervantes, A., Galvan, I. and Isasi, P. 2005. "A comparison between the Pittsburgh and Michigan approaches for the binary PSO algorithm", *IEEE Congress on Evolutionary Computation*, 1, 290–297.

[17] Clerc, M. and Kennedy, J. 2002. "The particle swarm-explosion, stability, and convergence in a multidimensional complex space", *IEEE Transactions on Evolutionary Computation*, 6(1), 58–73.

[18] De Falco, I., Cioppa, A. D. and Tarontino, E. 2006. "Evaluation of particle swarm optimization effectiveness in classification", LNAI (Vol. 3849). Berlin/Heidelberg: Springer-Verlag (pp. 164–171).

[19] Patricia Melin, Frumen Olivas, Oscar Castillo, Fevrier Valdez, Jose Soria and Mario Valdez. 2013, "Optimal design of fuzzy classification systems using PSO with dynamic parameter adaptation through fuzzy logic", *Expert Systems with*

Applications, Volume 40, Issue 8, Pages 3196–3206.

[20] Abdelbar, A., Abdelshahid, S. and Wunsch, D. 2005, "Fuzzy PSO: A generalization of particle swarm optimization", *In Proceedings 2005 IEEE international joint conference on neural networks* (Vol. 2, pp. 1086–1091).

[21] Bingül, Z. and Karahan, O. 2011. "A fuzzy logic controller tuned with PSO for 2 DOF robot trajectory control", *Expert Systems with Applications*, 38(1), 1017–1031.