Feature Subset Selection Approach by Gray-Wolf Optimization

E. Emary[1,4], Hossam M. Zawbaa[2,3,4], Crina Grosan[2], and Abul Ella Hassenian[1,4]

[1] Faculty of Computers and Information, Cairo University, Egypt
[2] Faculty of Mathematics and Computer Science, Babes-Bolyai University, Romania
[3] Faculty of Computers and Information, Beni-Suef University, Egypt
[4] Scientific Research Group in Egypt (SRGE), Egypt
http://www.egyptscience.net

Abstract. Feature selection algorithm explores the data to eliminate noisy, irrelevant, redundant data, and simultaneously optimize the classification performance. In this paper, a classification accuracy-based fitness function is proposed by gray-wolf optimizer to find optimal feature subset. Gray-wolf optimizer is a new evolutionary computation technique which mimics the leadership hierarchy and hunting mechanism of gray wolves in nature. The aim of the gray wolf optimization is find optimal regions of the complex search space through the interaction of individuals in the population. Compared with particle swarm optimization (PSP) and Genetic Algorithms (GA) over a set of UCI machine learning data repository, the proposed approach proves better performance in both classification accuracy and feature size reduction. Moreover, the gray wolf optimization approach proves much robustness against initialization in comparison with PSO and GA optimizers.

Keywords: Gray-wolf Optimization, feature selection, evolutionary computation.

1 Introduction

Feature selection algorithm explores the data to eliminate noisy, irrelevant, redundant data, and simultaneously optimize the classification performance. Feature selection is one of the most important stage in data mining, multimedia information retrieval, pattern classification, and machine learning applications, which can influence the classification accuracy rate [1],[2].

The main purpose of feature selection is to determine a minimal feature subset from a problem domain while retaining a suitably high accuracy in representing the original features [3]. In real world problems, feature selection is a must due to the abundance of noisy, misleading or irrelevant features [4]. By removing these factors, learning from data techniques can useful greatly. The motivation of feature selection in data mining, machine learning and pattern recognition is to reduce the dimensionality of feature space, improve the predictive accuracy of a classification algorithm, and develop the visualization and the comprehensibility of the induced concepts [5].

Genetic Algorithm (GA), Ant Colony Optimization (ACO), and Particle Swarm Optimization (PSO) are popular meta-heuristic optimization techniques. The Grey Wolf

A. Abraham, P. Krömer, & V. Snášel (eds.), *Afro-European Conf. for Ind. Advancement,*
Advances in Intelligent Systems and Computing 334, DOI: 10.1007/978-3-319-13572-4_1

Optimizer (GWO) is a new optimization algorithm which simulate the grey wolves leadership and hunting manner in nature. These techniques have been inspired by simple concepts.The inspirations are typically related to physical phenomena, animals behaviors, or evolutionary concepts [6]. In recent years, a lot of feature selection methods have been proposed. There are two key issues in structure a feature selection method: search strategies and evaluating measures. With respect to search strategies, complete , heuristic [7] , random [8] [9] strategies were proposed. And with respect to evaluating measures, these methods can be nearly divided into two classes: classification [10],[11],[12] and classification independent [13],[14],[15]. The previous employs a learning algorithm to evaluate the quality of selected features based on the classification accuracies or contribution to the classification boundary, such as the so-called wrapper method [10] and weight based algorithms [16],[17]. While the latter constructs a classifier independent measure to evaluate the importance of features, such as inter-class distance[13] mutual information[18], dependence measure[14] and consistency measure [15].

In recent years, a lot of feature selection methods have been proposed. There are two key issues in structure a feature selection method: search strategies and evaluating measures. With respect to search strategies, complete , heuristic [7] , random [8] [9] strategies were proposed. And with respect to evaluating measures, these methods can be nearly divided into two classes: classification [10],[11],[12] and classification independent [13],[14],[15]. The previous employs a learning algorithm to evaluate the quality of selected features based on the classification accuracies or contribution to the classification boundary, such as the so-called wrapper method [10] and weight based algorithms [16],[17] . While the latter constructs a classifier independent measure to evaluate the importance of features, such as inter-class distance[13] mutual information[18], dependence measure[14] and consistency measure [15].

In [22], the population of particles split into a set of interacting swarms. These interacting swarms applied the simple competition method. The winner is the swarm which has a best fitness value. The loser is eject and re-initialized in the search space, otherwise the winner remains. In [23], the swarm population divided into sub-populations species based on their similarity. Then, the repeated particles are removed when particles are identified as having the same fitness. After destroying the repeated ones, the new particles are added randomly until its size is resumed to its initial size.

In [24], the Bat Algorithm (BA) based on type of the sonar, which named echolocation behavior. The micro-bats have the capability of echolocation which attracting these bats can find their prey and discriminate different types of insects even in complete darkness.

In this paper, a classification accuracy-based fitness function is proposed by gray-wolf optimizer to find optimal feature subset. We compare Grey Wolf Optimizer (GWO) algorithm against Particle Swarm Optimization (PSO) and Genetic Algorithm (GA) algorithms for feature selection by applying three different initialization methods and eight different datasets. The results reveal that the GWO resulted in a higher accuracy compared to the other two optimization algorithms.

The rest of this paper is organized as follows: Section 2 presents basics of the gray wolf optimization. Section IV presents the details of the proposed system. In section V, there are experimental results and result analysis. Finally in Section VI, conclusions and future work are presented.

2 Preliminaries

2.1 Gray Wolf Optimization

Gray wolf optimization is presented in the following subsections based on the work in [25].

Inspiration. Grey wolves are considered as apex predators, meaning that they are at the top of the food chain. Grey wolves mostly prefer to live in a pack. The group size is 5-12 on average. They have a very strict social dominant hierarchy. The leaders are a male and a female, called *alpha*. The alpha is mostly responsible for making decisions about hunting, sleeping place, time to wake, and so on. The *alpha* decisions are dictated to the pack. The second level in the hierarchy of grey wolves is beta. The betas are subordinate wolves that help the alpha in decision-making or other pack activities. The beta wolf can be either male or female, and he/she is probably the best candidate to be the alpha in case one of the alpha wolves passes away or becomes very old. The lowest ranking grey wolf is omega. The omega plays the role of scapegoat. Omega wolves always have to submit to all the other dominant wolves. They are the last wolves that are allowed to eat. The fourth class is called subordinate (or delta in some references). Delta wolves have to submit to alphas and betas, but they dominate the omega. *Scouts*, *sentinels*, *elders*, *hunters*, and *caretakers* belong to this category. *Scouts* are responsible for watching the boundaries of the territory and warning the pack in case of any danger. *Sentinels* protect and guarantee the safety of the pack. *Elders* are the experienced wolves who used to be alpha or beta. *Hunters* help the alphas and betas when hunting prey and providing food for the pack. Finally, the *caretakers* are responsible for caring for the weak, ill, and wounded wolves in the pack.

Mathematical Model. In the mathematical model for the GWO the fittest solution is called the alpha (α). The second and third best solutions are named beta (β) and delta (δ) respectively. The rest of the candidate solutions are assumed to be omega (ω). The hunting is guided by α, β, and δ and the ω follow these three candidates. In order for the pack to hunt a prey they first encircling it. In order to mathematically model encircling behavior the following equations are used 1:

$$\overrightarrow{X}(t+1) = \overrightarrow{X}_p(t) + \overrightarrow{A}.\overrightarrow{D} \tag{1}$$

where $\overrightarrow{D}$ is as defined in 2 and t is the iteration number, $\overrightarrow{A}$, $\overrightarrow{C}$ are coefficient vectors, $\overrightarrow{X}_p$ is the prey position and $\overrightarrow{X}$ is the gray wolf position.

$$\overrightarrow{D} = |\overrightarrow{C}.\overrightarrow{X}_p(t) - \overrightarrow{X}(t)| \tag{2}$$

The $\overrightarrow{A}$, $\overrightarrow{C}$ vectors are calculated as in equations 3 and 4

$$\overrightarrow{A} = 2\overrightarrow{A}.\overrightarrow{r_1} - \overrightarrow{a} \tag{3}$$

$$\overrightarrow{C} = 2\overrightarrow{r_2} \tag{4}$$

where components of $\overrightarrow{a}$ are linearly decreased from 2 to 0 over the course of iterations and r_1, r_2 are random vectors in [0, 1]. The hunt is usually guided by the alpha. The beta and delta might also participate in hunting occasionally. In order to mathematically simulate the hunting behavior of grey wolves, the alpha (best candidate solution) beta, and delta are assumed to have better knowledge about the potential location of prey. The first three best solutions obtained so far and oblige the other search agents (including the omegas) to update their positions according to the position of the best search agents. So, the updating for the wolves positions is as in equations 5,6,7.

$$\overrightarrow{D_\alpha} = |\overrightarrow{C_1}.\overrightarrow{X_\alpha} - \overrightarrow{X}|, \overrightarrow{D_\beta} = |\overrightarrow{C_2}.\overrightarrow{X_\beta} - \overrightarrow{X}|, \overrightarrow{D_\delta} = |\overrightarrow{C_3}.\overrightarrow{X_\delta} - \overrightarrow{X}| \tag{5}$$

$$\overrightarrow{X_1} = |\overrightarrow{X_\alpha} - \overrightarrow{A_1}.\overrightarrow{D_\alpha}|, \overrightarrow{X_2} = |\overrightarrow{X_\beta} - \overrightarrow{A_2}.\overrightarrow{D_\beta}|, \overrightarrow{X_3} = |\overrightarrow{X_\delta} - \overrightarrow{A_3}.\overrightarrow{D_\delta}| \tag{6}$$

$$\overrightarrow{X}(t+1) = \frac{\overrightarrow{X_1} + \overrightarrow{X_2} + \overrightarrow{X_3}}{3} \tag{7}$$

A final note about the GWO is the updating of the parameter $\overrightarrow{a}$ that controls the tradeoff between exploation and exploitation. The parameter $\overrightarrow{a}$ is linearly updated in each iteration to range from 2 to 0 according to the equation 8:

$$\overrightarrow{a} = 2 - t.\frac{2}{Max_iter} \tag{8}$$

where t is the iteration number and Max_iter is the total number of iteration allowed for the optimization.

3 The Proposed Algorithm

In this section, we present the proposed GWO optimizer based on K-nearest neighbor for feature selection; see figure 1. We used the principles of gray wolf optimization for the optimal feature selection problem. Each feature subset can be seen as a position in such a space. If there are N total features, then there will be 2^N different feature subset, different from each other in the length and features included in each subset. The optimal position is the subset with *least length* and *highest classification accuracy*. We used gray wolf optimization for selecting the optimal feature set. Eventually, they should converge on good, possibly optimal, positions. The GWO makes iterations of exploration of new regions in the feature space and exploiting solution until reaching near-optimal solution.

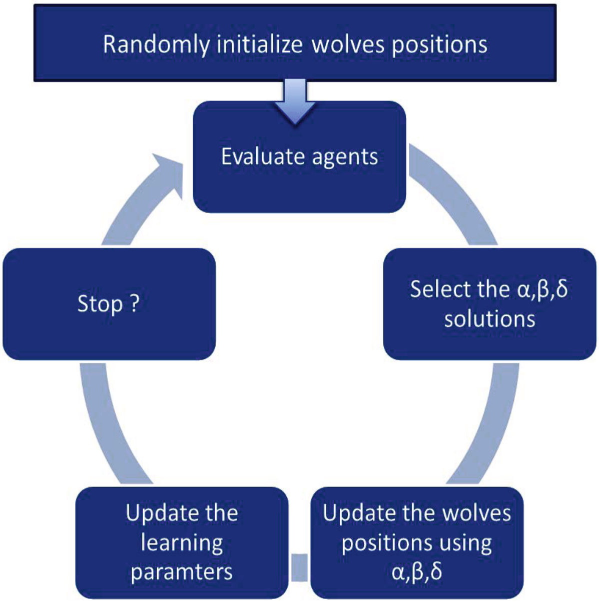
Randomly initialize wolves positions
Evaluate agents
Select the α,β,δ solutions
Update the wolves positions using α,β,δ
Update the learning paramters
Stop ?

Fig. 1. The overall feature selection algorithm

The solution space in here represents all possible selections of features and hence bat positions represents binary selection of feature sets. Each feature is considered as an individual dimension ranging from -2 to 2. To decide if a feature will be selected or not its position value will be threshold with a constant threshold. The used fitness function is the classification accuracy for k-nearest neighbors (KNN) classifier on the validation set. Each individual data set is divided into three equal subsets namely training, validation and test portions. The training set and validation set are used inside the fitness function to evaluate the selection classification accuracy while the test set is used in the end of optimization to evaluate the final selection classification performance.

We made use of two fitness functions in gray-wolf optimization (GWO) for feature selection, which are $KNN, andKNN_size$ resembling the well-known forward selection. Forward selection starts with an empty feature set (no features) and searches for a feature subset(s) with one feature by selecting the feature that achieves the highest classification performance. Then the algorithm selects another feature from the candidate features to add to S. Feature i is selected if adding i to S achieves the largest improvement in classification accuracy. While, backward selection starts with all the available features, then candidate features are sequentially removed from the feature subset until the further removal of any feature does not increase the classification performance. Small initialization resembles forward selection, large initialization motivated by backward selection and mixed initialization aiming to take the advantages of forward and backward selection to avoid their disadvantages.

4 Experimental Results and Discussion

4.1 Data Sets and Parameters Setting

Table 1 summarizes the 8 used data set for further experiments. The data set are drawn from the UCI data repository [27]. The data is divided into 3 equal parts one for *training*, the second part is for *validation* and the third part is for *testing*. We implement the GWO feature selection algorithms in MatLab R2009a. The computer used to get results is Intel (R), 2.1 GHz CPU; 2 MB RAM and the system is Windows 7 Professional. The parameter setting for the GWO algorithm is outlined in table 2. Same number of agents and same number of iterations are used for GA and PSO.

Table 1. Description of the data sets used in experiments

Dataset	No. of features	No. of samples
Lymphography	18	148
Zoo	16	101
Vote	16	300
Breastcancer	9	699
M-of-N	13	1000
Exactly	13	1000
Exactly2	13	1000
Tic-tac-toe	9	958

Table 2. Parameter setting for gray-wolf optimization

parameter	value(s)
No of wolves	5
No of iterations	100
problem dimension	same as number of features in any given database
Search domain	[0 1]

4.2 Results and Discussion

Four scenarios has been considered when we evaluate the proposed approach. They are: (1) **Scenario 1:** GWO, GA, and PSO features selection techniques using *normal* initialization, (2) **Scenario 2:** GWO, GA, and PSO features selection techniques using *large* initialization, (3) **Scenario 3:** GWO, GA, and PSO features selection techniques using *mixed* initialization, and (4) **Scenario 4:** GWO, GA, and PSO features selection techniques using *small* initialization.

Tables 3, 4, 5, and 6 are showing the performance of GWO, GA, PSO algorithms on the different eight data sets. Every algorithm is applied for 5 times on every data set to be sure about the algorithm robustness and we display the average result of all solutions. Gray wolf optimization (GWO) algorithm achieves high accuracy with the different data sets and initialization methods as showing in figures 2, 3, 4, and 5.

This demonstrates that GWO shows a good balance between exploration and exploitation that results in high local optima avoidance. This superior capability is due to the adaptive value of A. As mentioned above, half of the iterations are devoted to exploration ($|A| \geq 1$) and the rest to exploitation ($|A| < 1$). This mechanism assists GWO to provide very good exploration, local minima avoidance, and exploitation simultaneously.

Table 3. Normal initialization results for different datasets

Dataset No.	GWO	GA	PSO
1	0.980952	0.976190	0.985714
2	0.733333	0.970000	1.000000
3	0.773333	0.766667	0.740000
4	0.863636	0.886364	0.886364
5	1.000000	0.943333	0.946667
6	0.797909	0.797909	0.808362
7	0.977778	0.966667	0.944444
8	1.000000	0.933333	1.000000

Table 4. Large initialization results for different datasets

Dataset No.	GWO	GA	PSO
1	0.976190	0.990476	0.985714
2	1.000000	0.760000	0.723333
3	0.773333	0.756667	0.780000
4	0.886364	0.863636	0.795455
5	1.000000	0.920000	0.883333
6	0.839721	0.843206	0.832753
7	0.966667	0.966667	0.966667
8	1.000000	0.966667	0.933333

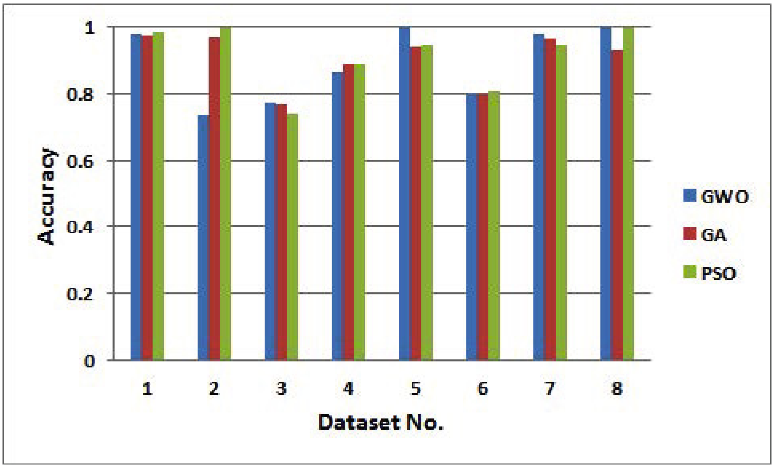

Fig. 2. Comparison of normal initialization results for GWO, GA, PSO with different datasets

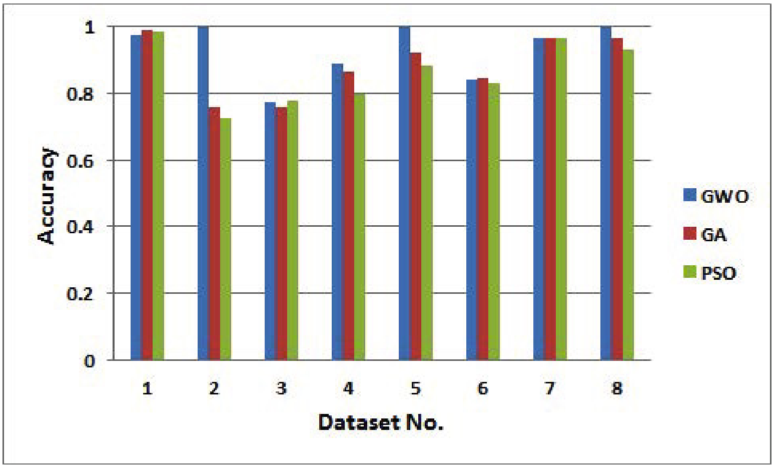

Fig. 3. Comparison of large initialization results for GWO, GA, PSO with different datasets

Figures 6 (a), (b), (c), and (d) present the standard deviation for the obtained fitness functions after running the each optimizer for 5 runs. The obtained standard deviation for GWO is much less than the obtained standard deviation for the PSO and GA which can be considered a prove for algorithm robustness regardless of the initialization method. SO, GWO always converge to the optimal solution or near optimal one regardless of its initialization method. GWO has abrupt changes in the movement of search agents over the initial steps of optimization. This assists a meta-heuristic to explore

Table 5. Mixed initialization results for different datasets

Dataset No.	GWO	GA	PSO
1	0.980952	0.985714	0.976190
2	1.000000	0.733333	0.773333
3	0.756667	0.736667	0.740000
4	0.818182	0.772727	0.772727
5	1.000000	0.966667	0.956667
6	0.832753	0.846690	0.850174
7	0.966667	0.944444	0.933333
8	0.966667	0.966667	0.900000

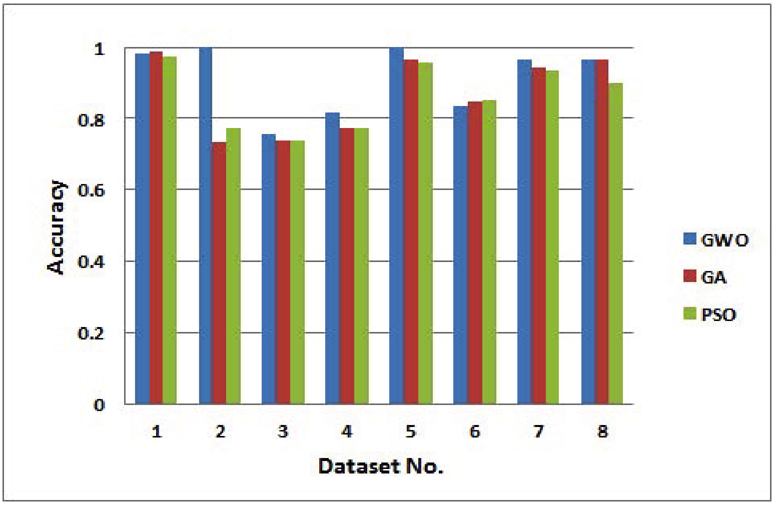

Fig. 4. Comparison curve of mixed initialization results for GWO, GA, PSO with different datasets

the search space extensively. Then, these changes should be reduced to emphasize exploitation at the end of optimization. In order to observe the convergence behavior of the GWO algorithm. The exploration power of GWO allow it to tolerate for initial solution as it quickly explore many regions and select the promising ones for further search.

Table 6. Small initialization results for different datasets

Dataset No.	GWO	GA	PSO
1	0.980952	0.966667	0.985714
2	1	0.86	0.753333
3	0.743333	0.773333	0.726667
4	0.863636	0.840909	0.863636
5	1	0.986667	0.94
6	0.850174	0.860627	0.84669
7	0.977778	0.955556	0.944444
8	0.966667	0.933333	0.966667

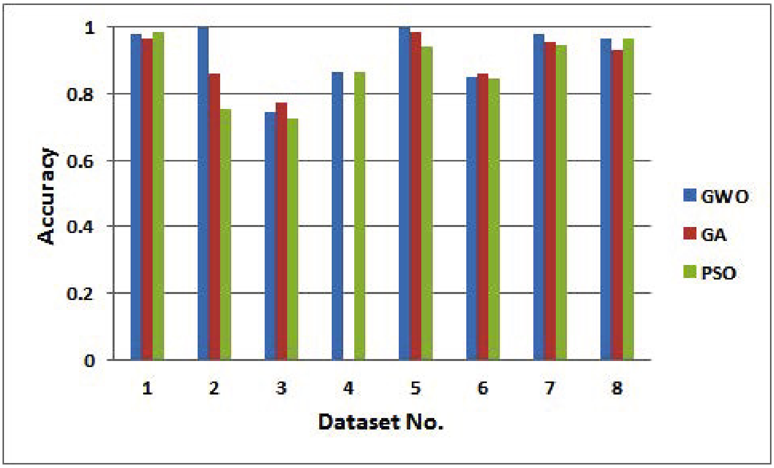

Fig. 5. Comparison curve of small initialization results for GWO, GA, PSO with different datasets

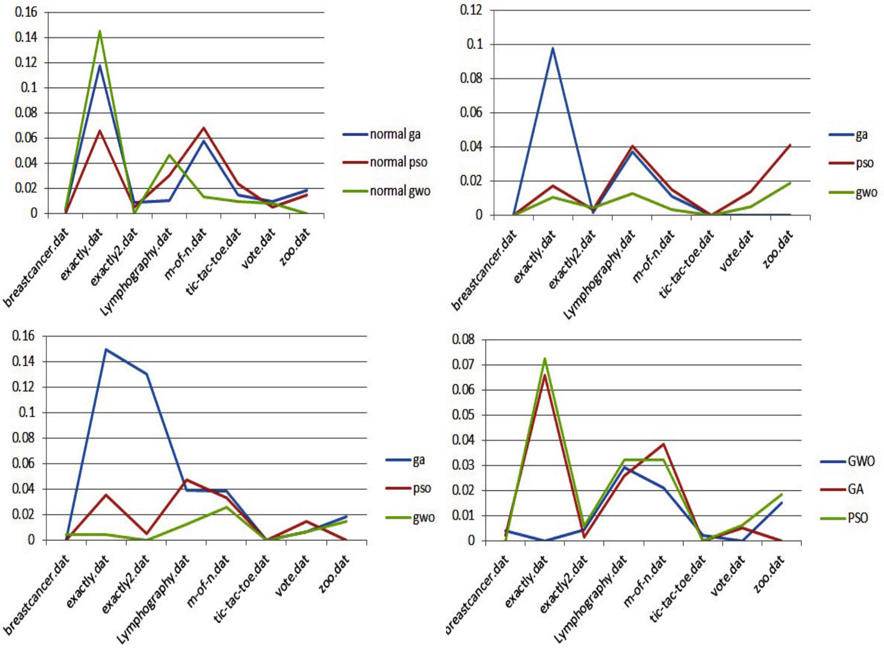

Fig. 6. (a) Standard deviation obtained for running the different methods on the different data sets [initialization normal], (b) Standard deviation obtained for running the different methods on the different data sets [initialization large] (c) Standard deviation obtained for running the different methods on the different data sets [initialization mixed] and (d) Standard deviation obtained for running the different methods on the different data sets [initialization small]

5 Conclusions and Future Work

In this paper, a system for feature selection based on intelligent search of gray wolf optimization has be proposed. Compared with PSO and GA over a set of UCI machine learning data repository, GWO proves much better performance as well as its proves much robustness and fast convergence speed. Moreover, the gray wolf optimization approach proves much robustness against initialization in comparison with PSO and GA optimizers. Other improvement to our work may involve applying some other feature selection algorithms and different fitness functions in the future which are expected to further enhance the results.

Acknowledgment. This work was partially supported by the IPROCOM Marie Curie initial training network, funded through the People Programme (Marie Curie Actions) of the European Union's Seventh Framework Programme FP7/2007-2013/ under REA grant agreement No. 316555. This fund only apply for two co-authors (Hossam M. Zawbaa and Crina Grosan).

References

1. Yang, C.-H., Tu, C.-J., Chang, J.-Y., Liu, H.-H., Ko, P.-C.: Dimensionality Reduction using GA-PSO. In: Proceedings of the Joint Conference on Information Sciences (JCIS), October 8-11. Atlantis Press, Kaohsiung (2006)
2. Cannas, L.M.: A framework for feature selection in high-dimensional domains. Ph.D. Thesis, University of Cagliari (2012)
3. Dash, M., Liu, H.: Feature selection for Classification. Intelligent Data Analysis 1(3), 131–156 (1997)
4. Jensen, R.: Combining rough and fuzzy sets for feature selection. Ph.D. Thesis, University of Edinburgh (2005)
5. Liu, H., Motoda, H.: Feature Selection for Knowledge Discovery and Data Mining. Kluwer, Boston (1998)
6. Mirjalili, S., Mirjalili, S.M., Lewis, A.: Grey Wolf Optimizer. Adv. Eng. Softw. 69, 46–61 (2014)
7. Zhong, N., Dong, J.Z.: Using rough sets with heuristics for feature selection. J. Intell. Inform. Systems 16, 199–214 (2001)
8. Raymer, M.L., Punch, W.E., Goodman, E.D., et al.: Dimensionality reduction using genetic algorithms. IEEE Trans. Evol. Comput. 4(2), 164–171 (2000)
9. Lai, C., Reinders, M.J.T., Wessels, L.: Random subspace method for multivariate feature selection. Pattern Recognition Lett. 27, 1067–1076 (2006)
10. Kohavi, R.: Feature subset selection using the wrapper method, Overfitting and dynamic search space topology. In: Proc. AAAI Fall Symposium on Relevance, pp. 109–113 (1994)
11. Gasca, E., Sanchez, J.S., Alonso, R.: Eliminating redundancy and irrelevance using a new MLP-based feature selection method. Pattern Recognition 39(2), 313–315 (2006)
12. Neumann, J., Schnorr, C., Steidl, G.: Combined SVM-based feature selection and classification. Machine Learning 61, 129–150 (2005)
13. Kira, K., Rendell, L.A.: The feature selection problem: Traditional methods and a new algorithm. In: Proc. AAAI 1992, San Jose, CA, pp. 129–134 (1992)
14. Modrzejewski, M.: Feature selection using rough sets theory. In: Proceedings of the European Conference on Machine Learning, Vienna, Austria, pp. 213–226 (1993)
15. Dash, M., Liu, H.: Consistency-based search in feature selection. Artif. Intell. 151, 155–176 (2003)
16. Guyon, I., Weston, J., Barnhill, S., Vapnik, V.: Gene selection for cancer classification using support vector machines. Machine Learning 46, 389–422 (2002)
17. Xie, Z.-X., Hu, Q.-H., Yu, D.-R.: Improved feature selection algorithm based on SVM and correlation. In: Wang, J., Yi, Z., Żurada, J.M., Lu, B.-L., Yin, H. (eds.) ISNN 2006. LNCS, vol. 3971, pp. 1373–1380. Springer, Heidelberg (2006)
18. Yao, Y.Y.: Information-theoretic measures for knowledge discovery and data mining. In: Karmeshu (ed.) Entropy Measures, Maximum Entropy and Emerging Applications, pp. 115–136. Springer, Berlin (2003)
19. Deogun, J.S., Raghavan, V.V., Sever, H.: Rough set based classication methods and extended decision tables. In: Proc. of the Int. Workshop on Rough Sets and Soft Computing, pp. 302–309 (1994)
20. Zhang, M., Yao, J.T.: A rough sets based approach to feature selection. In: IEEE Annual Meeting of the Fuzzy Information, Processing NAFIPS 2004, June 27-30, vol. 1, pp. 434–439 (2004)
21. Hu, X.: Knowledge discovery in databases: an attribute-oriented rough set approach. PhD thesis, University of Regina, Canada (1995)

22. Blackwell, T., Branke, J.: Multiswarms, “exclusion, and anti-convergence in dynamic environments”. IEEE Transactions on Evolutionary Computation 10, 459–472 (2006)
23. Parrott, D., Li, X.D.: Locating and tracking multiple dynamic optima by a particle swarm model using speciation. IEEE Transactions on Evolutionary Computation 10, 440–458 (2006)
24. Yang, X.-S.: A New Metaheuristic Bat-Inspired Algorithm. In: González, J.R., Pelta, D.A., Cruz, C., Terrazas, G., Krasnogor, N. (eds.) NICSO 2010. SCI, vol. 284, pp. 65–74. Springer, Heidelberg (2010)
25. Mirjalili, S., Mirjalili, S.M., Lewis, A.: Grey Wolf Optimizer. Advances in Engineering Software 69, 46–61 (2014)
26. Xue, B., Zhang, M., Browne, W.N.: Particle swarm optimisation for feature selection in classification:Novel initialisation and updating mechanisms. Applied Soft Computing 18, 261–276 (2014)
27. Bache, K., Lichman, M.: UCI Machine Learning Repository. University of California, School of Information and Computer Science, Irvine (2013), `http://archive.ics.uci.edu/ml`