ACSIJ Advances in Computer Science: an International Journal, Vol. 2, Issue 3, No.4, July 2013
ISSN : 2322-5157
www.ACSIJ.org

ACSIJ
WWW.ACSIJ.ORG

A Hybrid Approach for Optimal Feature Selection based on Evolutionary Algorithms and Classic Approaches

Hassan Abedi[1], Habib Rostami[2] ,Shiva Rahimi[3]

[1] Electronic engineering Department, Bushehr Branch Islamic Azad University, Bushehr, Iran
Hassanabedi_2007@yahoo.com

[2] Computer engineering Department, Persian Gulf University of Bushehr, Bushehr, Iran
habib@pgu.ac.ir

[3] Electronic engineering Department, Bushehr Branch Islamic Azad University, Bushehr, Iran
shivarahimi10@yahoo.com

Abstract

Feature selection (FS) is a fundamental problem in the field of pattern recognition, which aims to find a minimal feature subset from the original feature space while retaining a suitably high accuracy in representing the original features. FS is used to improve the efficiency of learning algorithm especially for large scale datasets, by finding a minimal subset of features that has maximum efficacy on classifier

In this paper, we proposed a new hybrid approach based on Evolutionary Algorithms and Heuristic methods for effective feature selection. In other words, the proposed approach has a hybrid heuristic/random strategy for search optimal solution. We compare the obtained simulation results with other algorithms separately, like evolutionary algorithms (with the same situation like iteration, population and cost function) consist on genetic algorithm (GA), ant colony optimization (ACO) and particle swarm optimization (PSO), and also with Heuristic Methods consist on sequential forward selection (SFS) and sequential backward elimination (SBE). Obtained results demonstrate that the proposed hybrid algorithm is effective and efficient for effective feature selection.

Keywords: *Feature selection, SFS, SBE, optimization.*

1. Introduction

In many fields such as data mining, machine learning, pattern recognition and signal Processing, datasets containing huge numbers of features are often involved. In such cases, feature selection will be necessary [4, 5]. Feature selection is the process of choosing a subset of features from the original set of features forming patterns in a given dataset. The subset should be necessary and sufficient to describe target concepts, retaining a suitably high accuracy in representing the original features. The importance of feature selection is to reduce the problem size and resulting search space for learning algorithms. In the design of pattern classifiers it can improve the quality and speed of classification [1]. Due to the abundance of noisy, irrelevant or misleading features, the ability to handle imprecise and inconsistent information in real world problems has become one of the most important requirements for feature selection. Rough sets [10, 11, and 5] can handle uncertainty and vagueness, discovering patterns in inconsistent data. Rough sets have been a useful feature selection method in pattern recognition [14]. The rough set approach to feature selection is to select a subset of features (or attributes), which can predict the decision concepts as well as the original feature set. The optimal criterion for rough set feature selection is to find shortest or minimal reducts while obtaining high quality classifiers based on the selected features [1, 3].

There are many rough set algorithms for feature selection. The most basic solution to finding minimal reducts is to generate all possible reducts and choose any with minimal cardinality, which can be done by constructing a kind of discernibility function from the dataset and simplifying it [2, 6] Starzyk uses strong equivalence to simplify discernibility functions [1-5]. Obviously, this is an expensive solution to the problem and is only practical for very simple datasets. It has been shown that finding minimal reducts or all reducts are both NP-hard problems [10]. Therefore, heuristic approaches have to be considered. In general, there are two kinds of rough set methods for feature selection, hill-climbing (or greedy) methods and stochastic methods [11-13]. The hill-climbing approaches usually employ rough set attribute significance as heuristic knowledge. They start off with an empty set or attribute core and then adopt forward selection or backward elimination. Forward selection adds in turn, one at a time, the most significant attribute from the candidate set, until the selected set is a reduct. Backward elimination is the reverse, starting with

the full attribute set and removing attributes incrementally. X. Hu gives a reduction algorithm using the positive region-based attribute significance as the guiding heuristic [8]. Wang develops a conditional information entropy-based reduction algorithm, using conditional entropy-based attribute significance [7-9]. K. Hu computes the significance of an attribute making use of heuristic ideas from discernibility matrices and proposes a heuristic reduction algorithm [5-6]. The positive region and conditional entropy-based methods choose a minimal feature subset that fully describes all concepts in a given dataset. The discernibility matrix-based method is to select a feature subset with high iscriminatory power, which guarantees the maximal between-class separability for the reduced data sets. These methods consider the best candidate attribute, trying to find a minimal reduct. However, hill-climbing methods do not guarantee to find an optimal or minimal reduct. As no perfect heuristic exists, there can be no guarantee of optimality. Using attribute significance to discriminate between candidates may lead the search down a non-minimal path. It is impossible to predict which combinations of attributes will lead to an optimal reduct with the addition or deletion of single attributes. Some researchers use stochastic methods for rough set feature selection [17]. Wróblewski uses genetic algorithms to find minimal reducts [14]. He combines a genetic algorithm with a greedy algorithm to generate short reducts. However, it uses highly time-consuming operations and cannot assure that the resulting subset is really a reduct. Bjorvand applies genetic algorithms to compute approximate reducts [5-8]. He takes Wróblewski's work as a foundation, but makes several variations and practical improvements both in speed and the quality of approximation. To obtain a good initial population for the GA, Bjorvand includes the attribute core in all candidates. In addition to this, he uses domain knowledge to get the average size of actual reducts and lets the number of features in the candidates be similar to the number in the reducts. Also, he allows the user to assign a relative weight to each
Attribute when creating the initial population. To avoid wasting much processing power in a wrong search direction, he adopts a dynamic mutation rate that is proportional to the redundancy in the population, preventing all individuals from becoming equal. Zhai proposes an integrated feature extraction approach based on rough set theory and genetic algorithms [18]. Rough sets are used to perform consistency checks, concept formation and approximation. By calculating the lower and upper approximations, training data is split into certain training data and possible training data. Then, a GA discovers best rules from the data sets. The fitness function is defined as the classification quality of the extracted rules. Ultimately, the features or attributes within rules with highest indices are selected. Jensen finds minimal rough set reducts using another stochastic strategy, Ant Colony Optimization (ACO)[12-14]. Hill-climbing methods are more efficient when dealing with little noise and a small number of interacting features, but are not assured of optimality. Stochastic methods can provide a more robust solution at the expense of increased computational effort [14]. For systems where the optimal or minimal subset is required (perhaps due to the cost of feature measurement), stochastic feature selection must be used. In this article we propose a new feature selection mechanism, investigating how particle swarm optimization (PSO) can be applied to find optimal feature subsets or rough set reducts. PSO is a new evolutionary computation technique proposed by Kennedy and Eberhart [10]. The particle swarm concept was motivated from the simulation of social behavior. The original intent was to graphically simulate the graceful but unpredictable movement of bird flocking. The PSO algorithm mimics the behavior of flying birds and their means of information exchange to solve optimization problems. Each potential solution is seen as a particle with a certain velocity, and "flies" through the problem space. Each particle adjusts its flight according to its own flying experience and its companions' flying experience. The particle swarms find optimal regions of complex search spaces through the interaction of individuals in a population of particles. PSO has been successfully applied to a large number of difficult combinatorial optimization problems; studies show that it often outperforms Genetic Algorithms [14-16].

2. Brief introduction to Heuristic search Methods

In the methods with heuristic searching, feature space has been searching with some special strategies. We have 3 main strategies which are:

- Continual remove: in this method feature groups have all of the features and then the worst feature eliminate form the groups till lead to best result.
- Forward selection: In this method at first feature groups are zero then add best features till lead to best result.
- Mutual selection: In this method we can valued feature groups zero, fit or random. And then simultaneously start to add best features and eliminate worst ones till we lead to best result.

2.1 sequential forward selection (SFS)

In sequential forward selection algorithm that starts with a null group, all of the subgroups of feature with length of 1 accessorized. The best feature is selected and then is added to subgroup. In the next step all reminded features one by one is added to subgroups. These new subgroups are accessorized again and the best feature is selected. In this step the subgroup has 2 features. This procedure is continued till new recuperation has not generated.

2.2 sequential backward elimination (SBE)

Sequential backward elimination is in the opposite of the SFS method. This algorithm starts from a group of all features. At first all of the subgroups with the length of N-1 tested. The worst feature eliminated and the total group in this step has N-1 feature. In the next step all of the subgroups with the length of N-2 are generated. These subgroups test again then the worst feature eliminate and so on till demolition started. After introduction of basic SBE& SFS, floating SBE, SFS introduced that have eliminated the problems of the basic methods.

3. GA based Heuristic algorithm [18]:

For feature selection with the genetic algorithm, chromosomes constructed based on binary arrays that 1 determined that feature presence and 0 determined absence of the feature. Any of the arrays are named chromosome. Decency value of the chromosomes is tested with a fitness function. With this algorithm at first the selected parents and with compilation new children generated and then fitness function for this generation measured. Then the best children's as next generation are selected. This continued till the iteration of the algorithm is finished. At first the heuristic (SBE or SFS) on data base accomplished and at the end a gradation of the features generated. And this selected as the first parents and the others selected from the best chromosomes of the later generation.

3.1 Encoding:

In this paper, binary encoding method is selected. In this way every chromosome has an array of 0 and 1 bits (figure1).

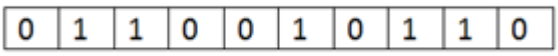

Fig1. Binary encoding

3.2 Evaluation chromosome intrinsic:

After generation of the all chromosomes, the intrinsic of the generated subgroups are evaluated. This evaluation is with this way that when the veracity of taxonomy of chromosome is high and the features of the chromosome is low then that chromosome is best. The error of the method with the I chromosome calculated from 1-1, 1-2:

$$(Cost)\_i = ((Error)\_i)^{\varphi 1} * ((Num)\_i)^{\varphi 2} \quad (1\text{-}1)$$

$$(Error)\_i = 1 - (Accuracy)\_i \quad (1\text{-}2)$$

In 1-1 Cost_i is the I chromosome error. And the Error_i is amount of classification and Num_i is the number of the solution of I chromosome.

3.3 Selection operator:

In this step the best chromosomes selected as the parents of the next generation. The usual way to parents selection is Roulette wheel model. In this model the probability of any chromosome selection has proportion to value of intrinsic. And the other model to select best chromosome is flavorzation. In this model any member quota has the ratio with the power of intrinsic of the chromosome. When the power is more than 1 better chromosomes selected. In this paper second model has been selected.

3.4 Crossover operator:

The most important operator in genetic algorithm is crossover or confection operator. The crossover operation operate on chromosome with Pc probability. In this paper every gen of the child chromosome individually and randomly selected from parents.

3.5 Mutation operator:

In this operation some gens of the chromosome altered randomly. The probability of the mutation is Pm that operation on every gen. in this paper we select binary mutation (figure 2).

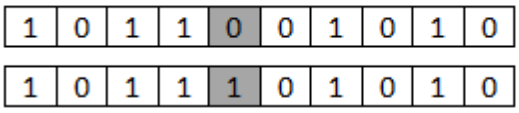

Fig 2. Binary mutation

4. Ant colony and Heuristic algorithm [19]:

When the solution generated from Heuristic algorithm this solution as historic data added to features selection function of ant colony algorithm (fig 3).

4.1 Subgroups selection of feature by ants

In common ways for solving a problem by ant colony algorithm, feature selection problem is simulated in a graph with N node (N is the all of the features) that any node has a feature.

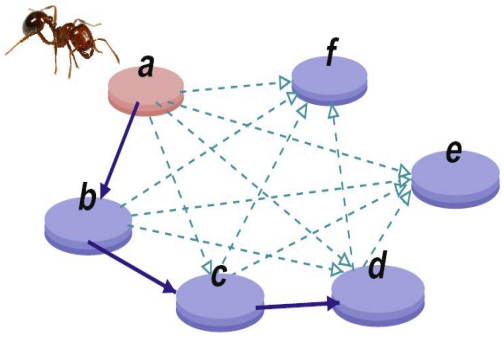

Fig 3. Ant colony based optimization

In picture (Fig 4) every way from one node to another means a feature, which 0 means absence of that feature for ant and 1 is the presence of the feature in the way of ant. For a solution select some features and not some others. In this approach we specialized Pheromone to feature.

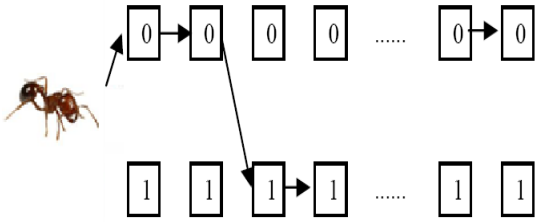

Fig 4. Subgroups selection of feature by ants

Selection of I feature by k ant is by the 1-3,1-4 :

$$P_i^k = \frac{\tau_i^\alpha * \eta_i^\beta}{\sum_l \tau_l^\alpha * \eta_l^\beta} \quad (1\text{-}3)$$

$$\eta_i = 1 - \gamma \frac{Rank_i}{N} \quad (1\text{-}4)$$

In 1-3 τ_i is pheromone η_i is historic data f I feature. γ is between 0 to 1. α, ß is coefficient Between 0 to 1. In this paper ß is zero and α can select alternatively: a=1 the problem is homogeny if α<1 features chance which is weak is high and if a>1 feature chance which is strong is high to be selected. When the ant migrates and generate the solutions competence calculated and the error for i,th ant can be calculated with (1-1,1-2).

Total updating of Pheromone: after testing the competence generated solutions Pheromones should be updated by *-*:

$$\tau_i(t+1) = (1-\lambda) * \tau_i(t) + \begin{cases} Q & \text{if edge}(i) \text{ belongs to the best ant} \\ 0 & \text{otherwise} \end{cases} \quad (1\text{-}5)$$

5. Feature selection based on Particle swarm optimization (PSO) and Heuristic algorithm:

Because PSO algorithm is basically continues so for generate primary population devotion array values for any swarm is random between 0 to 1. Every swarm is described in N dimension space that N is the all of the features. Heuristic algorithm solution with a random coefficient added to velocity of the PSO algorithm. In every iteration, in evaluation step if the dimension of i,th swarm that is bigger than 0.5 means that i,th feature selected and if is lower than 0.5 the i,th feature is not selected. After finishing the evaluation of all swarms, swarm that has best competence selected as gbest. Also quality of current swarm comprised with pbest of the swarm and if improvement has seen in its competence that values stored as pbest otherwise have no change. The error calculated with (1-1, 1-2).

6. Simulation

All of the simulations done with a personal computer(2.53GHz Intel processor, 4GB Ram and Windows 7 operating system and Matlab R2011). For simulations 8 data set downloaded from UCI site that information for every data set shown in table1.

Table 1 UCI site data sets which are used for this paper.

Database	No. Instance	No. Feature	No. Class
Breast Cancer	699	9	2
Pima	768	8	2
Hepatit	155	19	2
Iris	150	4	3
Wine	178	13	3
Chess	3196	36	2
Dermatology	366	34	6
Ionosphere	351	34	2

For every algorithm we have same qualifications and fitness function. Population and iteration for every algorithm are 30, 200 respectively. For Heuristic algorithm population and iterations are 20, 50 respectively.

7. Results

For first step we have no limitation on features numbers and the results shown in table 2, 3. The information's for these tables (2, 3) are for the last 10 iteration.

In the next step we limit the feature numbers and do simulations for 30, 60% of all features.

Table2 mean feature number extraction by using different algorithm

Dataset	Original Features	SFS	SBE	GA	ACO	PSO	ICA
Breast Cancer	9	4	5	3.8	4.4	3.7	3.4
Pima	8	5	7	5.2	4.7	5.1	5
Hepatit	19	8	13	12	11.5	10.3	8.6
Iris	4	4	2	3.3	3.1	3.2	3.7
Wine	13	10	7	8.8	8.4	8.1	8.3
Chess	36	22	19	19.3	21.3	19.2	17.2
Dermatology	34	18	12	18.8	18.6	22.4	15.1

GA&SFS	GA&SBE	ACO&SFS	ACO&SBE	PSO&SFS	PSO&SBE	ICA&SFS	ICA&SBE
3.8	5.2	3.1	3.5	3.8	4.7	3.3	4.6
6.4	4.7	4.6	6.2	4.8	5.2	5.2	5.5
13.1	12	10.5	7.5	11.9	13.4	10.2	7
3.1	3.1	3.8	4.2	3.7	3.8	4.3	3.5
8.4	8.1	8.6	8.3	9.3	9.6	8.3	8.7
16.6	19.5	17.6	18.6	20.5	18.5	15.6	16.4
17.1	16.9	21.4	16.3	18.3	19.2	17.4	16.3
11.6	15.8	14.3	16.4	13.3	17.5	15.4	17.3
Ionosphere		34	10	7	12.3	17.8	21.3	15.

Table3 mean feature number extraction by using different combination algorithm

The mean classification accuracy (%) for differe algorithms for last 10 iteration shown in table 4, 5.

Table 4 the mean classification accuracy (%) for different algorithms for last 10 iteration

Dataset	Original Accuracy	SFS	SBE	GA	ACO	PSO	ICA
Breast Cancer	94.44	95.07	94.96	96.67	96.53	94.23	96.62
Pima	72.31	72.96	73.64	75.22	73.23	71.56	76.77
Hepatit	84.12	80.86	81.67	89.44	86.20	87.22	88.49
Iris	82.29	81.44	87.79	94.89	91.22	88.95	93.44
Wine	98.86	98.12	95.31	100	98.66	99.23	100
Chess	94.26	95.28	94.20	96.23	97.12	94.56	96.19
Dermatology	98.09	100	99.09	99.36	98.55	89.38	99.32
Ionosphere	89.71	91.14	90.73	92.77	91.87	93.11	93.77

Table 5 The mean classification accuracy (%) for different combinational algorithms for last 10 iteration

GA&SFS	GA&SBE	ACO&SFS	ACO&SBE	PSO&SFS	PSO&SBE	ICA&SFS	ICA&SBE
96.47	96.43	96.66	96.23	96.67	96.87	97.12	96.82
77.81	72.60	76.34	79.23	76.30	74.78	80.43	78.29
89.32	87.39	88.43	88.12	89.17	87.40	87.97	88.92
95.84	93.33	88.58	94.18	95.52	93.76	95.78	94.26
100	99.41	99.23	100	99.65	100	100	100
97.13	96.59	95.39	96.10	96.77	95.90	96.82	97.27
99.61	99.29	99.20	99.32	98.78	98.37	99.75	99.51
92.45	90.43	95.46	95.71	93.40	92.63	94.28	94.88

Table6 the mean classification accuracy (%) adding limitation (30% of all features in simple and combinational algorithm).

Dataset	Original Accuracy	SFS	SBE	GA	ACO	PSO	ICA
Breast Cancer	94.44	94.71	94.96	95.20	94.23	93.72	95.46
Pima	72.31	74.27	75.58	71.56	72.92	72.16	74.19
Hepatit	84.12	87.10	88.32	92.36	91.28	89.38	92.72
Iris	82.29	68.89	71.34	72.55	74.27	74.12	73.17
Wine	98.86	94.55	92.36	97.16	98.30	97.52	98.11

GA&SFS	GA&SBE	ACO&SFS	ACO&SBE	PSO&SFS	PSO&SBE	ICA&SFS	ICA&SBE
96.66	96.19	97.72	96.55	95.20	94.99	96.77	95.88
77.87	75..54	82.91	80.66	75.47	77.01	79.70	83.07
93.77	92.63	94.20	93.89	93.29	92.94	95.04	95.55
76.99	79.72	80.08	81.82	76.89	78.91	79.10	81.92
98.55	100	98.88	100	98.48	97.52	99.39	100
94.35	96.87	96.98	96.30	95.89	96.62	95.88	96.58
99.77	97.76	96.77	98.88	99.68	98.18	98.39	99.32
94.29	94.20	91.33	92.78	93.99	93.22	94.19	94.88
Chess	94.26	94.92	95.31	94.29	95.47	95.19	96.00
Dermatology	98.09	98.63	99.06	99.32	89.76	89.23	99.32
Ionosphere	89.71	91.87	91.57	92.12	91.65	87.18	92.88

Table 7 the mean classification accuracy (%) adding limitation (60% of all features in simple and combinational algorithm)

Dataset	Original Accuracy	SFS	SBE	GA	ACO	PSO	ICA
Breast Cancer	94.44	96.34	95.88	96.15	95.56	96.10	96.32
Pima	72.31	74.73	74.97	72.44	75.29	74.19	77.32
Hepatic	84.12	86.33	85.82	93.21	90.19	91.91	91.57
Iris	82.29	83.36	81.89	92.12	90.10	87.52	91.19
Wine	98.86	98.33	97.28	99.12	98.29	98.91	99.33
Chess	94.26	96.12	96.05	95.33	92.19	90.44	96.88
Dermatology	98.09	99.32	99.08	99.23	98.29	97.66	99.46
Ionosphere	89.71	92.57	87.71	89.27	92.88	88.39	93.75

GA&SFS	GA&SBE	ACO&SFS	ACO&SBE	PSO&SFS	PSO&SBE	ICA&SFS	ICA&SBE
95.7	95.1	96.26	96.18	95.50	94.93	96.29	95.59
74.18	71.91	77.28	75.28	74.27	76.28	79.18	76.70
92.82	93.29	94.19	91.89	93.29	92.40	94.20	92.67
76.44	78.22	83.23	78.23	75.20	76.73	81.36	75.68
96.54	97.89	96.92	97.12	98.18	98.37	98.84	98.50
95.30	94.68	96.03	95.45	95.90	94.89	96.39	96.12
99.45	99.64	99.49	98.65	97.89	95.42	98.88	98.49
93.39	92.10	94.19	91.78	93.33	92.83	95.18	93.29

8. Conclusion

In this paper, we proposed a new hybrid approach based on Evolutionary Algorithms and Heuristic methods for effective feature selection. In other words, the proposed approach has a hybrid heuristic/random strategy for search optimal solution. We compare the obtained simulation results with other algorithms separately, like evolutionary algorithms (with the same situation like iteration, population and cost function) consist on genetic algorithm (GA), ant colony optimization (ACO) and particle swarm optimization (PSO), and also with Heuristic Methods consist on sequential forward selection (SFS) and sequential backward elimination (SBE). Obtained results demonstrate that the proposed hybrid algorithm is effective and efficient for effective feature selection.

References

[1] Hsu, C. W., & Lin, C. J. (2010). A simple decomposition method for support vector machine. Machine Learning, 46(1–3), 219–314.
[2] Joachims, T. (1998). Text categorization with support vector machines. In Proceedings of European conference on machine learning (ECML) (pp.137–142). Chemintz, DE.
[3] John, G., Kohavi, R., & Peger, K. (1994). Irrelevant features and the subsetselection problem. Proceedings of the 11th international conference on machine learning, San Mateo, CA pp. 121–129.
[4] Kecman, V. (2011). Learning and soft computing. Cambridge, MA: The MIT Press. Kohavi, R., & John, G. (1997). Wrappers for feature subset selection. Artificial Intelligence, 97(1–2), 273–324.
[5] LaValle, S. M., & Branicky, M. S. (2002). On the relationship between classical grid search and probabilistic roadmaps. International Journal of Robotics Research, 23(7–8), 673–692.
[6] Lin, H. T., & Lin, C. J. (2010). A study on sigmoid kernels for SVM and the training of non-PSD kernels by SMO-type methods. Technical report,
Department of Computer Science and Information Engineering, National Taiwan University. Available at: http://www.csie.ntu.edu.tw/~cjlin/papers/tanh.pdf.
[7] Mao, K. Z. (2012). Feature subset selection for support vector machines through discriminative function pruning analysis. IEEE Transactions on Systems, Man, and Cybernetics, 34(1), 60–67.
[8] Pontil, M., & Verri, A. (1998). Support vector machines for 3D object recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence, 20(6), 637–646.
[9] Raymer, M. L., Punch, W. F., Goodman, E. D., Kuhn, L. A., & Jain, A. K. (2004). Dimensionality reduction using genetic algorithms. IEEE Transactions on Evolutionary Computation, 4(2), 164–171.
[10] Salcedo-Sanz, S., Prado-Cumplido, M., Pe´rez-Cruz, F., & Bouson˜o-Calzo´n, C.(2012). Feature selection via genetic optimization Proceedings of the ICANN international conference on artificial neural networks, Madrid, Span pp. 547–552.
[11] Salzberg, S. L. (1997). On comparing classifiers: Pitfalls to avoid and a recommended approach. Data Mining and Knowledge Discovery, 1, 317–327.
[12] Scho˝lkopf, B., & Smola, A. J. (2000). Statistical learning and kernel methods. Cambridge, MA: MIT Press.
[13] Vapnik, V. N. (1995). The nature of statistical learning theory. New York: Springer.
[14] Weston, J., Mukherjee, S., Chapelle, O., Pontil, M., Poggio, T., & Vapnik, V.(2001). Feature selection for SVM. In S. A. Solla, T. K. Leen, & K.-R.
[15] Muller, Advances in neural information processing systems (Vol. 13) (pp.668–674). Cambridge, MA: MIT Press.
[16] Woods, K., & Bowyer, K. W. (1997). Generating ROC curves for artificial neural networks. IEEE Transactions on Medical Imaging, 16(3), 329–337.
[17] Yang, J., & Honavar, V. (1998). Feature subset selection using a genetic algorithm. IEEE Intelligent Systems, 13(2), 44–49.
[18] Yu, G. X., Ostrouchov, G., Geist, A., & Samatova, N. F. (2003). An SVM-based algorithm for identification of photosynthesis-specific genome features. Second IEEE computer society bioinformatics conference, CA, USA pp. 235–243.
[19] Sivagaminathan, R.K., Ramakrishnan, S., "A hybrid approach for feature subset selection using neural networks and ant colony optimization," Expert systems with applications, vol. 33, pp. 49-60, 2007.