Feature Selection Methods using Meta Heuristics

Manu Singla¹ Mrs. Poonam Dabas²
¹M.Tech. Student
²University Institute of Engineering and Technology-136119, Kurukshetra, Haryana, India

Abstract— The Feature Selection approaches can generally be divided into three groups: filter, wrapper, and hybrid approaches. The filter approach operates independently of any learning algorithm. These methods rank the features by some criteria and omit all features that do not achieve a sufficient score. Due to its computational efficiency, the filter methods are very popular to high-dimension data. Traditional search and optimization methods such as gradient-based methods are difficult to extend to the multi objective case because their basic design precludes the consideration of multiple solutions. In contrast, population-based methods such as evolutionary algorithms are well-suited for handling such situations. There are different approaches for solving multi objective optimization problems. For a large number of features, evaluating all states is computationally non-feasible requiring heuristic search methods. More recently, nature inspired metaheuristic algorithms have been used to select features, namely: particle swarm optimization (PSO), genetic algorithm (GA)-based attribute reduction, gravitational search algorithm (GSA). These methods attempt to achieve better solutions by application of knowledge from previous iterations. In this paper we survey feature selection and various Feature selection related to multi-objective optimization using meta heuristics.

Key words: Data Mining, Feature Selection, Multi-Objective Optimization, Meta Heuristics

I. DATA MINING

Data mining is used to mine useful data from a large amount of data in the same way as extraction of minerals from mine fields. Clustering is the classification of similar objects into different groups, or more precisely, the partitioning of a data set into subsets (clusters), so that the data in each subset share some common trait.

Data mining techniques can be implemented rapidly on existing software and hardware platforms to enhance the value of existing information resources, and can be integrated with new products and systems as they are brought on-line. When implemented on high performance client/server or parallel processing computers, data mining tools can analyze massive databases to deliver answers to questions such as, "Which clients are most likely to respond to my next promotional mailing, and why?"

Data mining is more than just conventional data analysis. It uses traditional analysis tools like statistics and graphics plus those associated with artificial intelligence such as rule induction and neural nets. It is all of these, but different. It is a distinctive approach or attitude to data analysis. The emphasis is not so much on extracting facts, but on generating hypotheses. The aim is more to yield questions rather than answers. Insights gained by data mining can then be verified by conventional analysis.

Knowledge Discovery in Data (KDD) is employed to describe the whole process of extraction of knowledge from data. In this context knowledge means relationships and patterns between data elements. The term 'data mining' should be used exclusively for the discovery stage of KDD process. So KDD is: ‘the non-trivial extraction of implicit, previously unknown and potentially useful knowledge from data’. Knowledge discovery consists of following types.

II. CLASSIFICATION

The problem of classification is basically one of partitioning the feature space into regions, one region for each category of inputs. Thus, it attempts to assign every data point in the entire feature space to one of the possible (say, K) classes. Classifiers are usually, but not always, designed with labeled data, in which case these problems are sometimes referred to as supervised classification (where the parameters of a classifier function are learned). Supervised classifiers assume that a set of training data is available. The training dataset consists of a set of instances that are properly labeled with the correct class labels. A learning algorithm then generates a model that attempts to minimize the prediction error on the training instances as much as possible, and also generalize as far as possible to new data.

The problem of supervised classification can formally be stated as follows. Given an unknown function g: X → Y (the ground truth) that maps input instances x ∈ X to output class labels y ∈ Y, and a training dataset D={(x₁,y₁),...,(xₙ,yₙ)} which is assumed to represent accurate examples of the mapping g, produce a function h: X → Y that approximates the correct mapping close as closely as possible. The learning algorithms help in identifying the class boundaries in the training set as correctly as possible by minimizing the training error. Various classification algorithms are available in the literature. Some common examples of the supervised pattern classification techniques are the nearest neighbor (NN) rule, the Bayes maximum likelihood classifier, support vector machines (SVM), and neural networks. A number of applications of evolutionary algorithms for classification purposes can also be found in the literature.

III. CLUSTERING

Clustering is an important unsupervised classification technique where a set of patterns, usually vectors in a multidimensional space, are grouped into clusters in such a way that patterns in the same cluster are similar in some sense and patterns in different clusters are dissimilar in the same sense.

Clustering in ad-dimensional Euclidean space Rⁿ is the process of partitioning a given set of n points into a number, say K, of groups (or clusters){C₁,C₂,...,Cₖ}based on some similarity/dissimilarity metric. The value of K may or may not be known a priori. The main objective of any clustering technique is to produce a K × n partition matrix U(X) of the given data set X consisting of n patterns,
X={x₁,x₂,...,xₙ}. The partition matrix may be represented as U = [uᵢⱼ], k = 1,...,K and j=1,...,n, where uᵢⱼ is the membership of pattern xᵢ to cluster Cⱼ. In the case of hard or crisp partitioning

\[ uᵢⱼ = \begin{cases} 1 & \text{if } Xⱼ \in Ck \\ 0 & \text{if } Xⱼ \notin Ck \end{cases} \]

On the other hand, for probabilistic fuzzy partitioning of the data, the following conditions hold on U (representing non degenerate clustering):

\[ \forall k \in \{1,2,...,K\} 0 \leq \sum_{j=1}^{n} uⱼk < n \]
\[ \forall j \in \{1,2,...,n\} \sum_{k=1}^{K} uⱼk = 1 \]

Several clustering methods are available in the literature. These can be broadly categorized into hierarchical (agglomerative and divisive), partitional (K-means, fuzzy C-means, etc.), and density-based (density-based spatial clustering of applications with noise, clustering for large applications, etc.) clustering. Evolutionary algorithms have also widely been used for clustering.

IV. ASSOCIATION RULE MINING

The principle of association rule mining (ARM) lies in the market basket or transaction data analysis. Association analysis is the discovery of rules showing attribute value associations that occur frequently. Let I= {i₁,i₂,..., iₙ} be a set of n items and X be an item set where X∈I. A k-item set is a set of k items. Let T={(t₁,X₁),(t₂,X₂),...,(tₘ,Xₘ)} be a set of m transactions, where t₁ and X₁, i=1, 2,...,m, are the transaction identifier and the associated item set, respectively. The cover of an item set X in T is defined as follows:

cover (X, T)=\{tᵢ|tᵢ,Xᵢ ∈ T, X ⊆ Xᵢ\}.

The support of an item set X in T is support(X, T)=|cover(X, T)| and the frequency of an item set is

Frequency (X, T) = \frac{support (X,T)}{|T|}

Thus, support of an item set X is the number of transactions where all the items in X appear in each transaction. The frequency of an item set is the probability of its occurrence in a transaction in T. An item set is called frequent if its support in T is greater than some threshold min_sup. The collection of frequent item sets with respect to a minimum support min_sup in T, denoted by F(T, min_sup), is defined as

F(T, min_sup)={X⊂I, support(X, T)> min_sup}.

The objective of ARM is to find all rules of the form X⇒Y, X∩Y =∅ with probability c%, indicating that if item set X occurs in a transaction, the item set Y also occurs with probability c%. X and Y are called the antecedent and consequent of the rule, respectively. Support of a rule denotes the percentage of transactions in T that contains both X and Y. This is taken to be the probability P(X∩Y). An association rule (AR) is called frequent if its support exceeds a minimum value min_sup.

The confidence of a rule X⇒Y in T denotes the percentage of the transactions in T containing X that also contains Y. It is taken to be the conditional probability P(X|Y). In other words

Confidence (X⇒Y, T) = \frac{support (X \cup Y, T)}{support (X,T)}

A rule is called confident if its confidence value exceeds a threshold min_conf. formally, the ARM problem can be defined as follows. Find the set of all rules R of the form X⇒Y such that

R= \{X⇒Y|X,Y ⊂ I, X \cap Y =∅, X \cup Y ∈ F(T, min_sup), min_conf ≤ \frac{P(X \cup Y)}{P(X)}\}

Generally, the ARM process consists of the following two steps.

1) Find all frequent item sets.
2) Generate strong ARs from the frequent item sets.

The number of item sets grows exponentially with the number of items |I|. A commonly used algorithm for generating frequent item sets is the a priori algorithm. This is based on the concept of downward closure property which states that if even one subset of an item set X is not frequent, then X cannot be frequent. It starts from all item sets of size one, and proceeds in a recursive fashion. If any item set X is not frequent, then that branch of the tree is pruned, since any possible superset of X can never be frequent.

V. FEATURE SELECTION

Feature selection problem deals with selection of an optimum relevant set of features or attributes that are necessary for the recognition process (classification or clustering). It helps reduce the dimensionality of the measurement space. The goal of feature selection is mainly threefold. First, it is practically and computationally difficult to work with all the features if the number of features is too large. Second, many of the given features may be noisy, redundant, and irrelevant to the classification or clustering task at hand. Finally, it is a problem when the number of features becomes much larger than the number of input data points. For such cases, reduction in dimensionality is required to permit meaningful data analysis. Feature selection facilitates the use of easily computable algorithms for efficient classification or clustering.

In general, the feature selection problem (Ω, P) can formally be defined as an optimization problem: determine the feature set F' for which

P(F') = min P(F,X)

F⊆Ω

Where Ω is the set of the possible feature subsets, F refers to a feature subset, and P: Ω × ψ → (R) denotes a criterion to measure the quality of a feature subset with respect to its utility in classifying/clustering the set of points X∈ψ. The elements of X, which are vectors in d-dimensional space, are projected into the subspace of dimension d=F=|F| defined by F. P is used to judge the quality of this subspace.

Feature selection can be either supervised or unsupervised. For the supervised case, the actual class labels of the data points are known. In filter approaches for supervised feature selection, features are selected based on their discriminatory power with regard to the target classes. In wrapper approaches for supervised feature selection, the utility of F is usually measured in terms of the performance of a classifier by comparing the class labels predicted by the classifier for feature space F with the actual class labels. For the unsupervised case, actual class labels are not available.
Hence, in filter approaches, features are selected based on the distribution of their values across the set of point vectors available. In wrapper-based unsupervised feature selection, the utility of a feature subset \( F \) is generally computed in terms of the performance of a clustering algorithm when applied to the input dataset in the feature space \( F \).

VI. DIMENSIONALITY REDUCTION THROUGH FEATURE SELECTION

Feature selection is the process of detecting the relevant features and discarding the irrelevant ones. A correct selection of the features can lead to an improvement of the inductive learner, either in terms of learning speed, generalization capacity or simplicity of the induced model. Moreover, there are some other benefits associated with a smaller number of features: a reduced measurement cost and hopefully a better understanding of the domain.

There are several situations that can hinder the process of feature selection, such as the presence of irrelevant and redundant features, noise in the data or interaction between attributes. Feature selection (FS), since it is an important activity in data preprocessing, has been widely studied in the past years by the machine learning researchers. This technique has found success in many different real-world applications like DNA microarray analysis, intrusion detection, text categorization or information retrieval, including image retrieval or music information retrieval.

There exist numerous papers and books proving the benefits of the feature selection process. However, most researchers agree that there is not a so-called “best method” and their efforts are focused on finding a good method for a specific problem setting. Therefore, new feature selection methods are constantly appearing using different strategies:

1) combining several feature selection methods, which could be done by using algorithms from the same approach, such as two filters, or coordinating algorithms from two different approaches, usually filters and wrappers;
2) combining FS approaches with other techniques, such as feature extraction or tree ensembles;
3) reinterpreting existing algorithms, sometimes to adapt them to specific problems;
4) Creating new methods to deal with still unresolved situations; and
5) Using an ensemble of feature selection techniques to ensure a better behavior.

Bearing in mind the large amount of FS methods available, it is easy to note that carrying out a comparative study is an arduous task. Another problem is to test the effectiveness of these FS methods when real datasets are employed, usually without knowing the relevant features.

In these cases the performance of the FS methods clearly relies on the performance of the learning method used afterward and it can vary notably from one method to another. Moreover, performance can be measured using many different metrics such as computer resources (memory and time), accuracy, ratio of features selected, etc. Besides, datasets may include a great number of challenges: multiple class output, noisy data, huge number of irrelevant features, redundant or repeated features, ratio number of samples/number of features very close to zero and so on. It can be noticed that a comparative study tackling all these considerations could be unapproachable, and therefore, most of the interesting comparative studies are focused on the problem to be solved. The majority of current real datasets (microarray, text retrieval, etc.) also present noisy data; however, no specific FS comparative studies dealing with this complex problem were found in the literature, although some interesting works have been proposed. From a theoretical perspective, in, a survey of feature selection methods was presented, providing some guidelines in selecting feature selection algorithms, paving the way to build an integrated system for intelligent feature selection.

More experimental work on feature selection algorithms for comparative purposes can be found in, some of which were performed over artificially generated data, like the widely used Parity, LED or Monks problems. Several authors choose to use artificial data since the desired output is known, therefore a feature selection algorithm can be evaluated with independence of the classifier used. Although the final goal of a feature selection method is to test its effectiveness over a real dataset, the first step should be on synthetic data. The reason for this is twofold:

1) Controlled experiments can be developed by systematically varying chosen experimental conditions, like adding more irrelevant features or noise in the input. This fact facilitates to draw more useful conclusions and to test the strengths and weaknesses of the existing algorithms.
2) The main advantage of artificial scenarios is the knowledge of the set of optimal features that must be selected; thus, the degree of closeness to any of these solutions can be assessed in a confident way.

Although works studying some of these problems can be found, up to the authors’ knowledge a complete study, such as the one described in here, has not been carried out. Besides, a very interesting problem, since it is very probable in very datasets, such as the alteration of the input variables, has not been addressed elsewhere.

VII. MULTI OBJECTIVE OPTIMIZATION

In this section, some basic concepts of MOO are first introduced. Then, an overview of available MOEAs is provided.

A. Concepts of Multi Objective Optimization

In many real-world situations, there may be several objectives that must be optimized simultaneously in order to solve a certain problem. This is in contrast to the problems tackled by conventional EAs, which involve optimization of just a single criterion. The main difficulty in considering multi objective optimization is that there is no accepted definition of optimum in this case, and therefore it is difficult to compare one solution with another one. In general, these problems admit multiple solutions, each of which is considered acceptable and equivalent when the relative importance of the objectives is unknown. The best solution is subjective and depends on the need of the designer or decision maker.

We are interested in the multi objective optimization problem (MOP), which can be stated as follows

Minimize \( \bar{F}(\bar{x}) := [f_1(\bar{x}), f_2(\bar{x}), \ldots, f_k(\bar{x})] \)
subject to  
\[ g_i(\bar{x}) \leq 1, 2, ..., m \]
\[ h_i(\bar{x}) = 0 \quad i = 1, 2, ..., p \]

Where \( \bar{x} = [x_1, x_2, ..., x_n] \) is the vector of decision variables, \( f_i : \mathbb{R}^n \rightarrow \mathbb{R}, i = 1, ..., k \) are the objective functions and \( g_i, h_i : \mathbb{R}^n \rightarrow \mathbb{R}, i = 1, ..., m, j = 1, ..., p \) are the constraint functions of the problem.

**B. Multi Objective Evolutionary Algorithms**

Traditional search and optimization methods such as gradient-based methods are difficult to extend to the multi objective case because their basic design precludes the consideration of multiple solutions. In contrast, population-based methods such as evolutionary algorithms are well-suited for handling such situations. There are different approaches for solving multi objective optimization problems.

MOEAs have evolved over several years, starting from traditional aggregating approaches to the elitist Pareto-based approaches and, more recently, to the indicator-based algorithms. In the aggregating approaches, multiple objective functions are combined into a single scalar value using weights, and the resulting single-objective function is then optimized using conventional evolutionary algorithms. In population based non-Pareto approaches such as the vector evaluated genetic algorithm, a special selection operator is used and a number of subpopulations are generated by applying proportional selection based on each objective function in turn. Among the Pareto-based approaches, multiple objectives GA, niched Pareto GA (NPGA), and no dominated sorting GA (NSGA) are the most representative nonelitist MOEAs. Although these techniques take into account the concept of Pareto optimality in their selection mechanism, they do not incorporate elitism and, therefore, they cannot guarantee that the nondominated solutions obtained during the search are preserved. In the late 1990s, a number of elitist models of Pareto-based multi objective evolutionary algorithms were proposed. The most representative elitist MOEAs include strength Pareto evolutionary algorithm (SPEA) and SPEA2, Pareto archived evolutionary strategy (PAES), Pareto envelope-based selection algorithm (PESA) and PESA-II, and nondominated sorting genetic algorithm-II (NSGA-II). Most of the recent applications of MOEAs for data mining problems have used one of these Pareto-based elitist approaches as their underlying optimization strategy. A more recent trend regarding the design of MOEAs is to adopt a selection mechanism based on some performance measure. For example, the indicator-based evolutionary algorithm is intended to be adapted to the user’s preferences by formalizing such preferences in terms of continuous generalizations of the dominance relation. Since then, other indicator-based approaches, such as the S metric selection evolutionary multi objective optimization algorithm (SMS-EMOA) (which is based on the hyper volume) have also been proposed. The main advantage of indicator based MOEAs such as SMS-EMOA is that they seem to scale better in the presence of many objectives (four or more). However, approaches based on the hyper volume are very computationally expensive. Since we do not review any application of an indicator-based MOEA in data mining, these approaches are not discussed further in this paper, and they are mentioned only for the sake of completeness.

The FS approaches can generally be divided into three groups: filter, wrapper, and hybrid approaches. The filter approach operates independently of any learning algorithm. These methods rank the features by some criteria and omit all features that do not achieve a sufficient score.

Due to its computational efficiency, the filter methods are very popular to high-dimension data. Some popular filter methods are F-score criterion, mutual information, information gain and correlation. The wrapper approach involves with the predetermined learning model, selects features on measuring the learning performance of the particular learning model. Although wrappers may produce better results, they are expensive to run and can break down with very large numbers of features. This is due to the use of learning algorithms in the evaluation of feature subsets every time. Filter and wrapper are two complementary approaches, then the hybrid approach attempts to take advantage of the filter and wrapper approaches by exploiting their complementary strengths.

For a large number of features, evaluating all states is computationally non-feasible requiring heuristic search methods. More recently, nature inspired metaheuristic algorithms have been used to select features, namely: particle swarm optimization (PSO), genetic algorithm (GA)-based attribute reduction, gravitational search algorithm (GSA). These methods attempt to achieve better solutions by application of knowledge from previous iterations.

Ant colony optimization (ACO) is another promising approach to solve the combinational optimization problems and has been widely employed in feature selection. It was initially used for solving Traveling Salesman Problem (TSP) and then has been successfully applied to a large number of NP-hard problems such as Quadratic Assignment Problem (QAP), vehicle routing, system fault detecting, scheduling, etc. In recent years, some ACO-based methods for feature selecting are reported. The hybrid of ACO and mutual information has been used for feature selection in the forecaster.

**VIII. RELATED WORK**

Pablo Bermejo, Jose A. Gámez et. al. (2011)[1] In this paper, feature subset selection is a key problem in the data mining classification task that helps to obtain more compact and understandable models without degrading (or even improving) their performance. In this work we focus on FSS in high-dimensional datasets, that is, with a very large number of predictive attributes. In this case, standard sophisticated wrapper algorithms cannot be applied because of their complexity, and computationally lighter filter-wraper algorithms have recently been proposed. In this work we propose a stochastic algorithm based on the GRASP meta-heuristic, with the main goal of speeding up the feature subset selection process, basically by reducing the number of wrapper evaluations to carry out. GRASP is a multi-start constructive method which constructs a solution in its first stage, and then runs an improving stage over that solution. Several instances of the proposed GRASP method are experimentally tested and compared with state-of-the-art algorithms over 12 high-dimensional datasets. The statistical analysis of the results shows that our proposal is comparable.
in accuracy and cardinality of the selected subset to previous algorithms, but requires significantly fewer evaluations.

Li-Yeh Chuang, Cheng-Hong Yang et. al. (2011) [2] In this paper, feature selection is a useful pre-processing technique for solving classification problems. The challenge of solving the feature selection problem lies in applying evolutionary algorithms capable of handling the huge number of features typically involved. Generally, given classification data may contain useless, redundant or misleading features. To increase classification accuracy, the primary objective is to remove irrelevant features in the feature space and to correctly identify relevant features. Binary particle swarm optimization (BPSO) has been applied successfully to solving feature selection problems. In this paper, two kinds of chaotic maps—so-called logistic maps and tent maps—are embedded in BPSO. The purpose of chaotic maps is to determine the inertia weight of the BPSO. We propose chaotic binary particle swarm optimization (CBPSO) to implement the feature selection, in which the K-nearest neighbor (K-NN) method with leave-one-out cross-validation (LOOCV) serves as a classifier for evaluating classification accuracies. The proposed feature selection method shows promising results with respect to the number of feature subsets. The classification accuracy is superior to other methods from the literature.

Li-Yeh Chuang, Sheng-Wei Tsai et. al. (2011) [3] In this paper, the feature selection process constitutes a commonly encountered problem of global combinatorial optimization. This process reduces the number of features by removing irrelevant, noisy, and redundant data, thus resulting in acceptable classification accuracy. Feature selection is a preprocessing technique with great importance in the fields of data analysis and information retrieval processing, pattern classification, and data mining applications. This paper presents a novel optimization algorithm called catfish binary particle swarm optimization (CatfishBPSO), in which the so-called catfish effect is applied to improve the performance of binary particle swarm optimization (BPSO). This effect is the result of the introduction of new particles into the search space (“catfish particles”), which replace particles with the worst fitness by the initialized at extreme points of the search space. The fitness of the global best particle has not improved for a number of consecutive iterations. In this study, the K-nearest neighbor (K-NN) method with leave-one-out cross-validation (LOOCV) was used to evaluate the quality of the solutions. CatfishBPSO was applied and compared to 10 classification problems taken from the literature. Experimental results show that CatfishBPSO simplifies the feature selection process effectively, and either obtains higher classification accuracy or uses fewer features than other feature selection methods.

MdMonirulKabir et. al. (2011) [4] In this paper presents a new hybrid genetic algorithm (HGA) for feature selection (FS), called as HGAFS. The vital aspect of this algorithm is the selection of salient feature subset within a reduced size. HGAFS incorporates a new local search operation that is devised and embedded in HGA to fine-tune the search in FS process. The local search technique works on basis of the distinct and informative nature of input features that is computed by their correlation information. The aim is to guide the search process so that the newly generated offsprings can be adjusted by the less correlated (distinct) features consisting of general and special characteristics of a given dataset. Thus, the proposed HGAFS receives the reduced redundancy of information among the selected features. On the other hand, HGAFS emphasizes on selecting a subset of salient features with reduced number using a subset size determination scheme. We have tested our HGAFS on 11 real-world classification datasets having dimensions varying from 8 to 7129. The performances of HGAFS have been compared with the results of other existing ten well-known FS algorithms. It is found that, HGAFS produces consistently better performances on selecting the subsets of salient features with resulting better classification accuracies.

Sungyoung, Lee, Young-Tack Park et. al. (2012) [5] In this paper, a novel feature selection method based on the normalization of the well-known mutual information measurement is presented. Our method is derived from an existing approach, the max-relevance and min-redundancy (mRMR) approach. We, however, propose to normalize the mutual information used in the method so that the domination of the relevance or of the redundancy can be eliminated. We borrow some commonly used recognition models including Support Vector Machine (SVM), k-Nearest-Neighbor (kNN), and Linear Discriminant Analysis (LDA) to compare our algorithm with the original (mRMR) and a recently improved version of the mRMR, the Normalized Mutual Information Feature Selection (NMIFS) algorithm. To avoid data-specific statements, we conduct our classification experiments using various datasets from the UCI machine learning repository. The results confirm that our feature selection method is more robust than the others with regard to classification accuracy.

M. Janaki Meena, K. R. Chandran et. al. (2012) [6] In this paper, feature selection is an indispensable preprocessing step for effective analysis of high dimensional data. It removes irrelevant features, improves the predictive accuracy and increases the comprehensibility of the model constructed by the classifiers sensitive to features. Finding an optimal feature subset for a problem in an outsized domain becomes intractable and many such feature selection problems have been shown to be NP-hard. Optimization algorithms are frequently designed for NP-hard problems to find nearly optimal solutions with a practical time complexity. This paper formulates the text feature selection problem as a combinatorial problem and proposes an Ant Colony Optimization (ACO) algorithm to find the nearly optimal solution for the same. It differs from the earlier algorithm by Aghdam et al. by including a heuristic function based on statistics and a local search. The algorithm aims at determining a solution that includes ‘n’ distinct features for each category. Optimization algorithms based on wrapper models show better results but the processes involved in them are time intensive. The availability of parallel architectures as a cluster of machines connected through fast Ethernet has increased the interest on parallelization of algorithms.

Esmat,Rashedi, Hossein Nezamabadi-Pour et. al. (2013) [7] In this paper, in content-based image retrieval (CBIR) applications, each database needs its corresponding parameter setting for feature extraction. However, most of the CBIR systems perform indexing by a set of fixed and...
show that our proposed algorithm can obtain higher processing speed as well as better classification accuracy using a smaller feature set than other existing methods.

Esmat, Rashedi, and Hossein Nezamabadi-pour et. al. (2014) [11] In this paper, feature selection is one of the important activities in various fields such as computer vision and pattern recognition. In this paper, an improved version of the binary gravitational search algorithm (BGSA) is proposed and used as a tool to select the best subset of features with the goal of improving classification accuracy. By enhancing the transfer function, we give BGSA the ability to overcome the stagnation situation. This allows the search algorithm to explore a larger group of possibilities and avoid stagnation. To evaluate the proposed improved BGSA (IBGSA), classification of some well-known datasets and improving the accuracy of CBIR systems are experienced. Results are compared with those of original BGSA, genetic algorithm (GA), binary particle swarm optimization (BPSO), and electromagnetic-like mechanism. Comparative results confirm the effectiveness of the proposed IBGSA in feature selection.

IX. Conclusion and Future Scope

As feature selection is one of the important activities in various fields such as computer vision and pattern recognition. Traditional search and optimization methods such as gradient-based methods are difficult to extend to the multi-objective case because their basic design precludes the consideration of multiple solutions. In this paper we survey feature selection and various Feature selection related to multi-objective optimization using meta heuristics. In contrast, population-based methods such as evolutionary algorithms are well-suited for handling such situations. Ant colony optimization (ACO) is another promising approach to solve the combinatorial optimization problems and has been widely employed in feature selection. It was initially used for solving Traveling Salesman Problem (TSP) and then has been successfully applied to a large number of NP-hard problems such as Quadratic Assignment Problem (QAP), vehicle routing, system fault detecting, scheduling, etc. In recent years, some ACO-based methods for feature selecting are reported. In future we will work on the hybrid of ACO and correlation information for feature selection.

REFERENCES


