Multi-Objective Evolutionary Optimization Algorithms for Machine Learning: A Recent Survey

Stamatios-Aggelos N. Alexandropoulos, Christos K. Aridas, Sotiris B. Kotsiantis, and Michael N. Vrahatis

Abstract The machine learning algorithms exploit a given dataset in order to build an efficient predictive or descriptive model. Multi-objective evolutionary optimization assists machine learning algorithms to optimize their hyper-parameters, usually under conflicting performance objectives and selects the best model for a given task. In this paper, recent multi-objective evolutionary approaches for four major data mining and machine learning tasks, namely: (a) data preprocessing, (b) classification, (c) clustering, and (d) association rules, are surveyed.

1 Introduction

For a given optimization task, in general, optimization consists of the following main issues [35]:

(a) *Objective function*: the quantity to be optimized (maximized or minimized).
(b) *Variables*: the inputs to the objective function.
(c) *Constraints*: the restrictions assigned to the inputs of the objective function.

Therefore, the purpose of an optimizer is to determine properly the values to the inputs of the objective function, in such a way to attain the optimal solution for the given function and fulfilling all the required constraints.

Various real-world optimization tasks often suffer from the following difficulties [112]:

1. In many cases, it is difficult to discern the global optimal minimizers from the local optimal ones.

S.-A. N. Alexandropoulos · C. K. Aridas · S. B. Kotsiantis · M. N. Vrahatis (✉)
Computational Intelligence Laboratory – CILab, Department of Mathematics, University of Patras, Patras, Greece
e-mail: alekst@math.upatras.gr; char@upatras.gr; sotos@math.upatras.gr; vrahatis@math.upatras.gr

© Springer Nature Switzerland AG 2019

I. C. Demetriou, P. M. Pardalos (eds.), *Approximation and Optimization*, Springer Optimization and Its Applications 145,
https://doi.org/10.1007/978-3-030-12767-1_4

2. The evaluation of the solutions may be difficult in the presence of noise.
3. The search space may be large, so the dimensionality of the problem grows similarly. This causes the so-called curse of dimensionality problem.
4. Difficulties associated with the given limitations assigned to the inputs of the objective function.
5. Necessity for problem-specific optimization techniques.

In the case where the quantity to be optimized is expressed by only one objective function, the problem is referred to as a *uni-objective* or *single-objective* problem. While, a *multi-objective* problem identifies more than one individual targets (sub-objectives) that should be optimized at the same time.

Various applications [44, 62, 85, 119] of machine learning and other types of problems [55, 63, 91] have been handled by techniques [6, 47, 108, 113] that belong to the field of machine learning, require the fulfillment of various conditions. The simultaneous fulfillment of such conditions, as well as the optimization of the parameters incorporated in machine learning methods, constitutes a difficult optimization problem. Indeed, the majority of these algorithms [20, 21, 132] require the optimization of multiple objectives, so that the outcome result to be reliable and competitive. For instance, in feature selection task, the desired feature set has to be the minimum set that maximizes the performance of the classifier. Therefore, two conditions are required:

(a) A minimum subset of features, and
(b) These features should maximize the performance of the algorithm.

Hence, the majority of learning problems are multi-objective in nature and thus, it is evident to consider learning problems as multi-objective ones. Freitas [37] presented the above-described purpose, that should be simultaneously optimized by certain conditions, so that the performance of the building model to be eventually high. As for the most part, the optimization of a number of parameters is required, in order for the accuracy of the model to be maximized. To achieve this goal, there are three different approaches, namely:

1. The conversion of the initial *multi-objective* problem into *single-objective* one by using properly a *weighted approach*.
2. The *lexicographical approach*, where the objectives are prioritized.
3. The well-known and widely used *Pareto approach*, which gives a whole set of non-dominated solutions.

An important issue provided by a multi-objective optimization algorithm [92, 118, 130] is that, instead of one solution, to return a set of good "candidate" solutions, the so-called *non-dominated solutions*, which the user can compare with one another in order to select the most appropriate one for the given purpose [22]. The set of solutions that the algorithm returns represents the best possible trade-offs among the objectives. However, the *Pareto principle* [90] indicates that the relationship between inputs and outputs is not balanced. Thus, the *Pareto rule* (or 80/20 rule) [82] "obeys" to a distribution, known as *Pareto distribution*, which reveals that 20% of the invested inputs are responsible for 80% of the obtained results.

In many cases, the decision of an expert, the so-called *decision maker* [56], plays a key role. Thus, the opinion of the decision maker may be requested initially before the beginning of the solution process. According to the information provided by the expert, the most appropriate solution is required to meet the conditions that have been set. However, the opinion of an expert can be requested after finding a number of appropriate solutions. In this case, the expert selects the best among them. In addition, the expert's opinion may take place during the process. Specifically, the model iteratively asks for the expert's opinion in order to improve the solutions and eventually returns the required optimal set of solutions.

Despite the fact that most of the real-world problems require the optimization of multiple objectives [15], there is always an effort to reduce the number of objectives to a minimum [17]. This occurs due to the fact that the more objectives are required to be optimized, the more solutions will be. Consequently, the dimensionality and the complexity of the problem are increased and therefore the problem becomes more difficult to be solved. For an analysis of the different types of multi-objective techniques, the reader can reach more details in [58].

In this paper, the references are focused on recent published refereed journals, books, and conference proceedings. In addition, various references regarding the original work that has emanated for tackling the particular line of research under discussion are incorporated. According to this purpose, the first major attempts that have surveyed a various multi-objective evolutionary approach are appeared in [77, 78].

As it is expected, it is not possible for a single work to cover extensively all the aspects of the *multi-objective evolutionary optimization* algorithms. However, we hope through this recent review work and the most recent references that the reader will be informed on the latest interests of the scientific community on these subjects.

The following section provides the necessary background material and basic concepts of multi-objective optimization. Section 3 covers a very important aspect of machine learning, which is the data preprocessing. To this end, various multi-objective evolutionary algorithms that tackled the most common and widely used data cleaning steps are presented. The classification task and the basic models used to handle this in accordance with multi-objective optimization algorithms are described in Sect. 4. Next, in Sect. 5, cluster analysis and association rules are presented. In Sect. 6 a few of the most recent applications regarding the multi-objective evolutionary optimization algorithms are given. The paper ends in Sect. 7 with a synopsis and a short discussion.

2 Basic Concepts of Multi-Objective Optimization

The *multi-objective optimization* (MO) also named *multiple criteria optimization* handles problems where different objectives must be optimized simultaneously. For this kind of problems, *Pareto optimality* replaces the optimality notion of single-objective optimization and each Pareto optimal solution represents a trade-off of the objective functions. Hence, two solutions may obtain the same fitness value and it is desirable to obtain the largest possible count of solutions with different inherent properties.

Suppose that $\mathscr{S} \subset \mathbb{R}^n$ is an n-dimensional search space and assume that

$$f_i(x) : \mathscr{S} \to \mathbb{R}, \quad i = 1, 2, \dots, k,$$

are k *objective functions* defined over $\mathscr{S}$. Let us assume that,

$$g_j(x) \leqslant 0, \quad j = 1, 2, \dots, m,$$

are m inequality *constraints*, then the MO problem can be stated as follows: Detect the point:

$$x^* = (x_1^*, x_2^*, \dots, x_n^*) \in \mathscr{S},$$

that fulfills the constraints and optimizes the following function:

$$F_{nk}(x) = \big(f_1(x), f_2(x), \dots, f_k(x)\big) : \mathbb{R}^n \to \mathbb{R}^k.$$

The objective functions may be conflicting with each other, thus, it is usually impossible to find the global minimum for all the objectives at the same point. The aim of MO is to provide a set of *Pareto optimal solutions* (points) to the above-mentioned problem.

Specifically, assume that $u = (u_1, u_2, \dots, u_k)$ and $v = (v_1, v_2, \dots, v_k)$ are two vectors. Then, *u dominates v* if and only if $u_i \leqslant v_i$, for $i = 1, 2, \dots, k$, and $u_i < v_i$ for at least one component. This condition is known as *Pareto dominance* and it is used to determine the Pareto optimal solutions. Therefore, a solution x of the MO problem is called *Pareto optimal* if and only if there is not another solution y, such that $F_{nk}(y)$ dominates $F_{nk}(x)$.

The set of all Pareto optimal solutions of an MO problem, denoted by $\mathscr{P}^*$, is called *Pareto optimal set* while the set:

$$\mathscr{P}\mathscr{F}^* = \big\{\big(f_1(x), f_1(x), \dots, f_k(x)\big) \mid x \in \mathscr{P}^*\big\},$$

is called *Pareto front*. A Pareto front $\mathscr{P}\mathscr{F}^*$ is said to be *convex* if and only if there exists a $w \in \mathscr{P}\mathscr{F}^*$, such that:

$$\lambda\|u\| + (1 - \lambda)\|v\| \geq \|w\|, \quad \forall\, u, v \in \mathscr{P}\mathscr{F}^*,\ \forall\, \lambda \in (0, 1),$$

while it is called *concave* if and only if there exists a $w \in \mathscr{P}\mathscr{F}^*$, such that:

$$\lambda\|u\| + (1 - \lambda)\|v\| \leqslant \|w\|, \quad \forall\, u, v \in \mathscr{P}\mathscr{F}^*,\ \forall\, \lambda \in (0, 1).$$

A Pareto front can be convex, concave, or partially convex and/or concave and/or discontinuous. The last three cases exhibit the greatest difficulty for the majority of MO techniques.

Using the MO approach is desirable to detect all the Pareto optimal solutions. On the other hand, the Pareto optimal set may be infinite and since the computation is usually restricted within strict time and space limitations, the main aim of MO is the detection of the largest possible number of Pareto optimal solutions, with the smallest possible deviation from the Pareto front and suitable spread [93].

The *evolutionary algorithms* have the ability to evolve multiple Pareto optimal solutions simultaneously and thus, they are particularly efficient and effective in tackling MO problems. The detected Pareto optimal solutions are stored in memory structures, called *external archives* which, in general, increase the performance of MO approaches. A plethora of well-known and widely applied MO evolutionary approaches have been proposed that are based on different approaches including *niching fitness sharing* and *elitism*, among others [23, 30, 36, 52, 93, 114, 131].

3 Data Preprocessing

The *machine learning (ML) algorithms* aim to automate the process of knowledge extraction from formats that can be easily processed by computer systems. In general, the "quality of the data" could decrease the performance of a learning algorithm. Thus, data preprocessing [40] is an important task in the machine learning pipeline that usually is executed by removing objects and features that contain extraneous and irrelevant information.

Feature Selection The task of detecting and eliminating irrelevant and redundant *features* also known as *attributes* is called *feature selection* (FS) [110]. This task tries to compact the cardinality of the data attributes and to assist the learning algorithms in order to function faster and more efficiently. In general, features can be distinguished as follows:

(a) *Relevant*: Features that contribute an important role for the class and they cannot be assumed by the remaining ones.
(b) *Irrelevant*: Features that do not have any influence on the target class.
(c) *Redundant*: Features that can be replaced by other features.

By eliminating the irrelevant and redundant features, the FS process could assist towards decreasing the training time as well as to simplify the learned models and/or to improve the performance measure of the problem. In general, FS could be considered as a multi-objective task. The main objectives are two: the first one is the maximization of the model's performance while the second one is the minimization of the number of features that will be fed in the learning algorithm. The aforementioned objectives are conflicted and the optimal choice has to be made by considering a balance between the two objectives. Multi-objective FS can acquire a set of non-dominated feature splits in order to meet diverse requirements in real-world applications.

Next, we briefly present various approaches for FS. The efficient and effective *particle swarm optimization* (PSO) method [59, 93] is considered as a metaheuristic approach that attempts to solve an optimization task by maintaining a population of candidate solutions (which are called particles). The members of the swarm are moving around the search space according to a mathematical model that tackles two parameters: the particles' position and velocity. Xue et al. [125] conducted a study on different types of *multi-objective PSO* for FS. The main objective of their work was to create a *PSO-based multi-objective FS scheme* in order to tackle classification problems. Their approach tries to achieve a Pareto front of non-dominated solutions, which will contain a subset of the initial feature space, by simultaneously achieving a more accurate classification performance without using all the available attributes.

Han and Ren [48] proposed a multi-objective technique to improve the performance of FS. They believe that their method could meet different requirements as well as to achieve a trade-off between different conflicting objectives.

Paul and Das [96] proposed an FS and the weighting method supported by an evolutionary multi-objective algorithm on decomposition. The instance attributes are selected and weighted, or scaled and at the same time the data points are displayed to a specific hyper-space. Furthermore, the distances between the data points of the non-identical classes are increased in such a way to facilitate their classification.

Wang et al. [121] presented an algorithm called MECY-SF by using class-dependent redundancy for the FS procedure. Their algorithm exploits genetic search and multi-objective optimization to overcome the limitations of greedy FS algorithms. Furthermore, the fast and elitist multi-objective genetic algorithm named NSGA-II [23] was adopted to solve the multi-objective feature selection problem. Xue et al. [126] gave recently an up-to-date review of the most promising works on evolutionary computation for FS, which provides the contributions of these algorithms.

Cano et al. [11] through their new multi-objective method have succeeded feature extraction and data visualization. Their algorithm is based on Pareto optimal set and is combined with genetic programming. Various classification and visualization measures were assumed as objectives to be optimized by their algorithm.

Das and Das [19] formulated the FS as a *bi-objective optimization problem* of some real-valued weights that correspond to each attribute in the feature space. Therefore, a subset of the weighted attributes is selected as the best subset for subsequent classification of the dataset. The *relevancy* and *redundancy* measures were selected for creating the objective functions.

The FS problem was handled by Hancer et al. [49] through a new *multi-objective artificial bee colony (ABC) algorithm*. Specifically, the authors developed an FS method that will search for a Pareto optimal set of features. They proposed two versions, namely the *binary multi-objective artificial bee colony* named Bin-MOABC version and the corresponding continuous version Num-MOABC. Their algorithm approaches the multi-objective problem through the minimum selection of features that provides the lower classification error in accordance to the original set of features. They tested the proposed algorithm in twelve benchmark datasets and their experimental results show that the Bin-MOABC algorithm exhibits a better classification accuracy and outperforms the other considered methods regarding the dimensionality reduction.

Last but not least, Zheng and Wang [129] proposed an FS method that combines the *joint maximal information entropy* (JMIE), as a measurement metric of a feature subset, and a *binary particle swarm optimization* (BPSO) algorithm for searching the optimal set of features. The authors conducted experiments on five UCI datasets and their experimental results show that the provided technique exhibits a better performance in FS with multiple classes. In addition, their method is more consistent and achieves a better time-efficiency than the BPSO-SVM algorithm.

Instance Selection The *instance selection* or *prototype selection* [25] can be considered as an optimization problem since required the maintenance of mining quality, at first, and secondary the minimization of the sample size. The complexity of the induced solution tends to be increased by the number of the training examples. On the other hand, this may decrease the interpretability of the results. Thus, instance selection is highly recommended in the case of big datasets. Fernández et al. [33] used a multi-objective evolutionary algorithm for searching to obtain the best joint set of both features and instances.

Acampora et al. [1] proposed a multi-objective optimization scheme for the *training set selection (TSS) problem*. The main difference between the provided technique and the evolutionary approaches that had already been developed is the *multi-objective a priori technique*. This means that their method maintains two objectives, namely the *classification accuracy* and the *rate reduction*, unlike all the other evolutionary methods for TSS problem in *support vector machines* (SVM). The authors tested their method using the UCI datasets and the conducted experiments show that the provided algorithm exhibits a better performance on well-known TSS techniques and reinforces the efficiency of SVMs.

Missing Data Imputation The *incomplete* or *corrupted* data values is a common problem [69] in many of the real-life databases. There are many considerations that have to be keep in view in accordance with processing unknown attributes. Determining the origin of the *unknownness* is a major issue. Thus, we lead to the following reasons:

1. The feature is omitted because somehow it was forgotten or for some reason it got lost.
2. For a given object a specific feature value is not applicable.
3. The training dataset collector may not interested in a specific feature value for a given instance.

Lobato et al. [70] presented a multi-objective *genetic algorithm* for data imputation, based on the fast and elitist multi-objective genetic algorithm called NSGA-II [23], which is suitable for mixed (categorical and continuous) attribute datasets and it considers information from incomplete instances and the modelling task. In order to compute the objective function, the following two most common evaluation measures were chosen: (a) the *root mean square error* and (b) the *classification accuracy*.

Discretization The *discretization* assists to the transformation of the space of real-valued attributes to a fixed number of distinct discrete values. A large number of possible feature values could lead to time-consuming machine learning learners. The choice of the number of bins in the discretization process remains an open problem.

Tahan and Asadi [116] proposed an evolutionary approach for the discretization process by using two objectives. The first objective function minimizes the classification error, while the second one minimizes the number of cut points.

Imbalanced Datasets The ideal situation for a *supervised predictor* is to generalize over unknown objects of any class with the same accuracy. In real-life tasks, learners deal with *imbalanced datasets* [13]. This phenomenon leads the learner to be "subjective" towards one class. This can happen when one class is greatly under-represented in the training set in relation to the others. It is associated with training of learning algorithms.

Algorithms in the *inductive machine learning* usually are designed to minimize a predefined metric over a training dataset. Moreover, if any class contains a small amount of examples, in the most of the cases, it can be ignored by the learning algorithms. This is because the cost of performing well on the over-represented class outweighs the cost of doing poorly on the smaller class. Recently, a *convex-hull-based multi-objective genetic programming algorithm* was proposed [128]. This algorithm was applied to binary classification cases and achieved to maximize the convex hull area by minimizing the false positive rate and maximizing the true positive rate simultaneously. The area under the *receiver operating characteristic* (ROC) curve was used as a performance assessment and for the guidance of the search.

Zhao et al. [128] in their attempt to improve the 2D ROC space incorporated the complexity to the objectives. This led to the creation of a 3D objective space (in contrast with the previous 2D ROC space). Li et al. [67] applied swarm optimization on two aspects for re-balancing the imbalanced datasets. One aspect is the search for the appropriate amount of majority instances, while the other one is the estimation of the best control parameters, namely the intensity and the distance of the neighbors of the minority samples in order to be synthesized.

4 Supervised Learning

In machine learning, the *classification* [61] is the paradigm where an algorithm is trained using a training set of correctly identified instances in such a way to produce a model that will be able to correctly identify unseen objects.

Decision Trees The *decision trees* [99, 102] classify examples starting from the root node and afterwards they sort them based on their feature values. In a decision tree each *node* represents a feature of an instance to be classified, while each *branch* represents a value that the node can have.

Zhao [127] proposed a multi-objective genetic programming approach in order to develop a *Pareto optimal decision tree*. This implementation allows the user to select priorities for the conflicting objectives, such as *false negative* versus *false positive*, *sensitivity* versus *specificity*, and *recall* versus *precision*.

Fieldsend [34] used the *particle swarm optimization* (PSO) method [59, 93] in order to train near optimal decision tree using the multi-objective formulation for trading off error rates in each class.

Basgalupp et al. [7] proposed a genetic algorithm for inducing decision trees called LEGAL-Tree. Specifically, they proposed a *lexicographic* approach, where multiple objectives are evaluated in the order of their priority.

Barros et al. [6] provided a taxonomy which groups works that evolve decision trees using evolutionary algorithms. Chikalov et al. [14] created bi-criteria optimization problems for decision trees. The authors considered different cost functions such as *number of nodes*, *depth*, and *average depth*. They design algorithms that are able to determine Pareto optimal points for a given decision table.

Rule Learners The *classification rules* [38, 123] represent each class by the *disjunctive normal form*. The aim is to find the smallest rule-set that is consistent with the training set. Many produced rules are usually a sign that the learning algorithm over-fits the training data.

Dehuri et al. [24] gave an *elitist multi-objective genetic algorithm* (EMOGA) for producing classification rules. They proposed a multi-objective genetic algorithm with a hybrid crossover operator for simultaneously optimizing the objectives of the *comprehensibility*, the *accuracy*, and the *interestingness of rules*.

Pappa and Freitas [87] also successfully produced accurate as well as compact rule models using a *multi-objective grammar-based genetic programming* algorithm.

Srinivasan and Ramakrishnan [115] tackled the problem of discovering rules as a multi-objective optimization problem. They used an approach with three objectives to be optimized. These were metrics such as *accuracy*, *comprehensibility*, and *novelty*.

Rudzinski [104] presented a multi-objective genetic approach in order to produce *interpretability-oriented fuzzy rules* from data. Their proposed approach allows the user to obtain systems with various levels of compromise between *accuracy* and *interpretability*.

Bayesian Classifiers A *Bayesian network* (BN) [51, 124] is a graphical model for probabilistic relationships among the variables. The structure S of a BN is a *directed acyclic graph*. The nodes in S are in one-to-one correspondence with the variables and the arcs represent casual influences among the variables. The lack of possible arcs in S represents conditional independency, while a node (variable) is conditionally independent from its non-descendants given its parents.

Rodriguez and Lozano [101] introduced a structural learning approach of a multi-dimensional Bayesian learner based on the fast and elitist multi-objective genetic algorithm NSGA-II [23].

Panda [86] used the so-called ENORA algorithm which is an FS multi-objective evolutionary algorithm for multi-class classification problems. Specifically, the author estimated the averaged 1-dependence estimators of naive Bayes, through the aforesaid algorithm. The proposed scheme was tested on twenty one real-world datasets and the experimental results show that the implementation of the method is promising in terms of time and accuracy.

Support Vector Machines The *support vector machine* (SVM) [50, 109] is a classification model that is based on the *structured risk minimization theory*. Selecting C, kernel, and γ parameters of SVM is crucial for producing an efficient SVM model. The parameter C of the radial basis function (RBF) kernel SVM compromises misclassification of the training examples contrary to simplicity of the decision surface. A low value of the parameter C causes the decision surface smooth, while a high value of C attempts at classifying all the training examples correctly by providing the model freedom to select more samples as support vectors. The γ parameters can be considered as the inverse of the radius of influence of samples selected by the model as support vectors.

Aydin et al. [5] used a *multi-objective artificial immune algorithm* in order to optimize the kernel as well as the parameters of SVM. Miranda et al. [76] proposed a *hybrid multi-objective architecture* which combines *meta-learning* with *multi-objective particle swarm optimization algorithms* in order to tackle the SVM parameter selection problem.

Gu et al. [45] proposed a *bi-parameter space partition algorithm* for SVMs, which is able to fit all the solutions for every parameter pair. Based on the bi-parameter space partition, they proposed a K-fold cross-validation algorithm for computing the global optimum parameter pairs.

Rosales-Perez et al. [103] used an *evolutionary multi-objective model* and instance selection for SVMs for producing Pareto-based ensemble. Their aims were to minimize the size of the training data and maximize the classification accuracy by the selecting instances.

Neural Networks It is well known that the *perceptrons* [106, 107] are only able to classify linearly separable sets of instances. If the instances are not linearly separable, learning will never find a hyperplane for which all examples are correctly classified. To this end, the *multilayered perceptrons* (artificial neural networks) have been proposed in order to tackle this problem.

Tan et al. [117] used a *modified micro-genetic algorithm optimizer* for twofold, i.e., to select a small number of input features for classification and to improve the accuracy of the neural network model.

Ojha et al. [84] proposed a *multi-objective genetic program* (MOGP) in order to create a *heterogeneous flexible neural tree*, which is a tree-like flexible feed-forward neural network model.

Lazy Learners The *K-nearest neighbor* (k-NN) [18, 72] is based on the principle that the instances within a dataset will generally share similar properties. The k-NN finds the k nearest instances to the testing instance and predicts its class by identifying the most frequent class. Prototype generation is the generation of a small set of instances to replace the initial data, in order to be used by k-NN for classification. The main aspects to consider when implementing a prototype generation method are:

(a) the accuracy of a k-NN classifier using the prototypes and
(b) the percentage of dataset reduction.

Both factors are in conflict and thus this problem can be naturally handled with multi-objective optimization techniques.

Escalante et al. [31] proposed a *multi-objective evolutionary algorithm for prototype generation*, named MOPG. In addition, Hu and Tan [54] presented a prototype generation using a *multi-objective particle swarm optimization for k-NN classifier*.

Ensembles The selection of a single algorithm in order to produce a reliable classification model is not an easy task. A simple approach could be the estimation of the accuracy of the candidate algorithms on a problem and then the selection of the best performer. The idea of combining classifiers [26, 27] is proposed as a direction for increasing the classification accuracy in real-world problems. In this case, the objective is to use the strengths of one model to complement the weaknesses of the other. In general, the multi-objective evolutionary algorithms for the construction of classifier ensembles is an interesting area of study and research.

Chandra and Yao [12] presented an *ensemble learning algorithm*, which is named DIVACE (DIVerse and ACcurate Ensemble learning algorithm). This algorithm tries to find a trade-off between diversity and accuracy by treating these two objectives explicitly separately. Three other approaches for the *Pareto-based multi-objective ensemble generation* approach are compared and discussed in [57].

Bhowan et al. [9] proposed a *multi-objective genetic programming* method in order to evolve accurate and diverse classifiers with acceptable accuracy both on the minority and majority of class. Furthermore, Bhowan et al. [10] presented another similar approach in order to evolve ensembles by using genetic programming for imbalanced data.

Nguyen et al. [83] used a genetic algorithm approach that focuses on the following three objectives:

1. The count of correct classified instances,
2. The count of selected attributes, and
3. The count of selected classifiers.

Gu et al. [46] presented a survey on multi-objective ensemble generation methods, including the diversity measures, member generation, as well as the selection and integration techniques.

Nag and Pal [80] presented an integrated algorithm for simultaneous attribute selection and inducing diverse learners using a steady state multi-objective genetic programming, which minimizes the following three objectives:

(a) False positives predictions,
(b) False negatives predictions, and
(c) The count of leaf nodes in the decision tree.

Albukhanajer et al. [3] propose classifier ensembles that use multiple Pareto image features for invariant image identification.

Last but not least, very recently, Pourtaheri et al. [98] developed two multi-objective heuristic ensemble classifiers by combining the *multi-objective inclined planes optimization* algorithm and the *multi-objective particle swarm optimization* (MOPSO) algorithm.

5 Unsupervised Learning

Clustering The *cluster analysis* [88] is a process that is very useful for the exploration of a collection of data. As it is implied by the term "cluster," through this process, elements or features with an inter-relationship are detected, which can lead to the homogeneous clustering of data. It can be either *supervised* or *unsupervised* and the major difference between this process and the classification process is that the first one does not use labels to assist in the categorization of the data in order to create a cluster structure. Furthermore, whether an item belongs to a particular cluster is determined through an intra-connectivity measurement. If this measure is high, it means that the clusters are "compact" and the data of the same group are highly dependent on each other. On the other hand, the inter-connectivity measurement is a criterion that declares the independence between the clusters. Thus, if the inter-connectivity is low, it means that the individual clusters are largely independent to each other. More details about *multi-objective evolutionary clustering algorithms*, the reader can reach in [29, 79].

The *mathematical programming* [32] has an important contribution to the issue of cluster analysis. The direct connection of the two areas can be easily understood, since it is required the minimum number of clusters to which the original dataset can be grouped. Thus, this approach can be considered as an optimization problem, with specific features and constraints. An important issue is also the appropriate selection of solutions, since from a set of feasible, "good" solutions, the best solutions are those of interest [89].

Luo et al. [71] proposed a method for modelling spectral clustering and through specific operators they selected a set of good individuals at the optimization process. Furthermore, the authors through the *ratio cut criterion* selected a trade-off solution from the Pareto set. Finally, the various problems that have been analyzed for supervised and unsupervised classification tasks contributed to the creation of semi-supervised clustering techniques. With a small amount of labelled data and the data distribution, Alok et al. [4] proposed a *semi-supervised clustering method* by using the multi-objective optimization framework.

Wang et al. [122], recently, through an *evolutionary multi-objective* (EMO) algorithm tackled a very difficult and timeless challenge for a clustering method problem, namely the *"determination of the number of clusters."* To this end, the authors proposed a scheme that uses an EMO algorithm, specifically the rapid elitist multi-objective genetic algorithm named NSGA-II, in order to select the non-dominated solutions. The process that follows includes a validity index for selecting the optimal clustering result. The authors tested their model on three datasets and their experimental results show that the *EMO-k-clustering method* is effective and by executing only a single run it is able to obtain all the clustering results for different values of the parameter k.

Last but not least, very recently, Nayak et al. [81] proposed the *elitism-based multi-objective differential evolution* (EMODE) algorithm for automatic clustering. Their work handles complex datasets using three objectives. The authors conducted experiments on ten datasets and the results show that their approach provides an alternative solution for data clustering in many different areas.

Association Rules The *association rule mining* (ARM) [111] has as a primary goal the discovery of associations rules between data of a given database. The first goal of this procedure is to come up with the data that have the greatest appearance in the database. Then, the appropriate association rules are created for the whole dataset by using the feature values that their appearance exceed a certain predetermined threshold.

Minaei-Bidgoli et al. [75] proposed a *multi-objective genetic algorithm* for mining association rules from numerical variables. It is known that well-known and widely used models that handle the association rule mining process cannot be applied to datasets which consist of numerical data. For this reason, it is necessary the preprocessing of the data and in particular the discretization process. Minaei-Bidgoli et al. [75], using three measures, namely: *confidence*, *interestingness*, and *comprehensibility*, defined three different objective functions for their approach and extracted the best association rules through Pareto optimality.

Beiranvand et al. [8] proposed a *multi-objective particle swarm optimization model* named MOPAR for mining numerical association rules in only one single step without a priori discretization. The authors conducted experiments and the results show that their approach extracts reliable, comprehensible, and interesting numerical association rules.

Martin et al. [73] proposed a *multi-objective evolutionary model* named QAR-CIP-NSGA-II which extents the well-known elitist multi-objective genetic algorithm named NSGA-II [23]. Their method performs an evolutionary learning and a selection condition for each association rule. Furthermore, their approach maximizes two of the three objective functions that Minaei-Bidgoli et al. considered. In addition, their approach maximizes the performance of the objective functions for mining a set of quantitative association rules with enough interpretability as well as accurate results.

6 A Few of the Most Recent Applications

The various applications that have been provided over the last years show the importance of the *multi-objective evolutionary optimization algorithms* (MOEOA). A few of the most recent and very interesting applications regarding MOEOA are the following ones.

Mason et al. [74] developed an artificial neural network that has been trained through a *differential evolution* (DE) algorithm. Their proposed neural network has the ability to handle multi-objective optimization problems using properly an approximation function. Specifically, the proposed approach uses a single objective global optimizer (the DE algorithm) in order to evolve the neural network. In other words, the so-called MONNDE algorithm is capable to provide further *Pareto fronts* without any further optimization effort. The authors applied the MONNDE algorithm to the well-known *dynamic economic emission dispatch* problem and through the experiments that they conducted, they show that the performance of their algorithm is equally optimal in comparison with other well-known and widely used algorithms. Furthermore, they show that it is more efficient to optimize the topology of the neural network dynamically with an online way, instead of to optimize the weights of the neural network.

Rao et al. [100] proposed an alternative *classifier for disease diagnosis*. Specifically, the proposed scheme includes a sequential minimal optimization, the SVM classifier, and three evolutionary algorithms for the evolution of the parameters. Moreover, the authors presented a new technique, which is named *cuboids elephant herding optimization* (CEHO). Their approach is applied to seventeen medical datasets and the experimental results show that the proposed technique exhibits a very good performance for all the tested datasets.

Sabar et al. [105] considered the configuration of a SVM as a bi-objective optimization problem. The accuracy of the model was the first objective while the other one was the complexity of the model. The authors proposed a novel hyper-heuristic framework for the optimization of the above-mentioned two conflicting objectives. The developed approach tested on two *cyber security problems* and the experimental results show that their proposed scheme is very effective and efficient.

7 Synopsis and Discussion

In general, the subject of the *multi-objective evolutionary optimization algorithms* (MOEOA) is related to an interesting concept with many different aspects and a crucial role, not only in machine learning, but also in many other scientific fields. This is evident, since in nowadays, the necessity of handling conflicting performance objectives appears in many scientific fields. The plethora of papers written regarding MOEOA show in an emphatic way that the scientific community has a great concern about this subject.

The very first evolutionary approaches to solve multi-objective optimization problems and especially the particle swarm optimization and differential evolution algorithms appeared very promising [91–95]. It is worth mentioning that the vector evaluated particle swarm optimization and the vector evaluated differential evolution [91, 95] remain the basis of current research in *multi-objective optimization*, *many-objective optimization*, and *dynamic multi-objective optimization*. Furthermore, the multi-objective optimization has led to better performing machine learning models in contrast to the traditional single objective ones.

The importance of multi-objective evolutionary algorithms is apparent not only from the plethora of papers that have been presented by the scientific community, but also from a huge amount of various applications that have been presented over the last decades such as engineering [43], industry [64], economy [16, 66], and many others [2, 28, 39, 41, 42, 53, 65, 68, 97]. The reader could also reach more details about the variety of the problems and the amount of applications in [60] and [120].

This paper describes how multi-objective evolutionary optimization algorithms have been used in the field of machine learning, in relative detail. It should be noted that the list of the references in the current work does not provide a complete list of the papers corresponding to this subject. The main aim was to provide a survey of the basic ideas, rather than a simple list, of all the research papers that have been discussed or have been used these ideas. Nevertheless, it is hoped that the mentioned references will cover the most important theoretical issues and will give guidelines to the main branches of literature regarding such techniques and schemes, guiding the interested reader to up-to-date research directions.

Acknowledgements S.-A. N. Alexandropoulos is supported by Greece and the European Union (European Social Fund-ESF) through the Operational Programme "Human Resources Development, Education and Lifelong Learning" in the context of the project "Strengthening Human Resources Research Potential via Doctorate Research" (MIS-5000432), implemented by the State Scholarships Foundation (IKY).

References

1. Acampora, G., Herrera, F., Tortora, G., Vitiello, A.: A multi-objective evolutionary approach to training set selection for support vector machine. Knowl. Based Syst. **147**, 94–108 (2018)
2. Ahmed, F., Deb, K., Jindal, A.: Multi-objective optimization and decision making approaches to cricket team selection. Appl. Soft Comput. **13**(1), 402–414 (2013)
3. Albukhanajer, W.A., Jin, Y., Briffa, J.A.: Classifier ensembles for image identification using multi-objective Pareto features. Neurocomputing **238**, 316–327 (2017). https://doi.org/10.1016/j.neucom.2017.01.067
4. Alok, A.K., Saha, S., Ekbal, A.: A new semi-supervised clustering technique using multi-objective optimization. Appl. Intell. **43**(3), 633–661 (2015). https://doi.org/10.1007/s10489-015-0656-z
5. Aydin, I., Karakose, M., Akin, E.: A multi-objective artificial immune algorithm for parameter optimization in support vector machine. Appl. Soft Comput. **11**(1), 120–129 (2011). https://doi.org/10.1016/j.asoc.2009.11.003

6. Barros, R.C., Basgalupp, M.P., De Carvalho, A.C., Freitas, A.A.: A survey of evolutionary algorithms for decision-tree induction. IEEE Trans. Syst. Man Cybern. Part C Appl. Rev. **42**(3), 291–312 (2012)
7. Basgalupp, M.P., Carvalho, A.C.D., Barros, R.C., Ruiz, D.D., Freitas, A.A.: Lexicographic multi-objective evolutionary induction of decision trees. Int. J. Bio-Inspired Comput. **1**(1/2), 105 (2009). https://doi.org/10.1504/ijbic.2009.022779
8. Beiranvand, V., Mobasher-Kashani, M., Bakar, A.A.: Multi-objective PSO algorithm for mining numerical association rules without a priori discretization. Expert Syst. Appl. **41**(9), 4259–4273 (2014). https://doi.org/10.1016/j.eswa.2013.12.043
9. Bhowan, U., Johnston, M., Zhang, M., Yao, X.: Evolving diverse ensembles using genetic programming for classification with unbalanced data. IEEE Trans. Evol. Comput. **17**(3), 368–386 (2013). https://doi.org/10.1109/tevc.2012.2199119
10. Bhowan, U., Johnston, M., Zhang, M., Yao, X.: Reusing genetic programming for ensemble selection in classification of unbalanced data. IEEE Trans. Evol. Comput. **18**(6), 893–908 (2014). https://doi.org/10.1109/tevc.2013.2293393
11. Cano, A., Ventura, S., Cios, K.J.: Multi-objective genetic programming for feature extraction and data visualization. Soft. Comput. **21**(8), 2069–2089 (2017). https://doi.org/10.1007/s00500-015-1907-y
12. Chandra, A., Yao, X.: Ensemble learning using multi-objective evolutionary algorithms. J. Math. Model. Algorithms **5**(4), 417–445 (2006). https://doi.org/10.1007/s10852-005-9020-3
13. Chawla, N.V.: Data mining for imbalanced datasets: an overview. In: Data Mining and Knowledge Discovery Handbook, pp. 875–886. Springer, Boston (2009)
14. Chikalov, I., Hussain, S., Moshkov, M.: Bi-criteria optimization of decision trees with applications to data analysis. Eur. J. Oper. Res. **266**(2), 689–701 (2018). https://doi.org/10.1016/j.ejor.2017.10.021
15. Coello Coello, C.A.: A comprehensive survey of evolutionary-based multiobjective optimization techniques. Knowl. Inf. Syst. **1**(3), 269–308 (1999)
16. Coello Coello, C.A.: Evolutionary multi-objective optimization and its use in finance. In: Rennard, J.-P. (ed.) Handbook of Research on Nature Inspired Computing for Economy and Management. Idea Group Publishing, Hershey (2006)
17. Costa, L., Oliveira, P.: Dimension reduction in multiobjective optimization. Proc. Appl. Math. Mech. **7**(1), 2060047–2060048 (2007)
18. Cunningham, P., Delany, S.J.: *k*-nearest neighbour classifiers. Mult. Classif. Syst. **34**, 1–17 (2007)
19. Das, A., Das, S.: Feature weighting and selection with a Pareto-optimal trade-off between relevancy and redundancy. Pattern Recogn. Lett. **88**, 12–19 (2017). https://doi.org/10.1016/j.patrec.2017.01.004
20. Deb, K.: Multi-Objective Optimization using Evolutionary Algorithms. Wiley, Chichester (2001)
21. Deb, K.: Multi-objective optimization using evolutionary algorithms: an introduction. In: Wang, L.-H., Ng, A.H.C., Deb, K. (eds.) Multi-Objective Evolutionary Optimisation for Product Design and Manufacturing, pp. 401–430. Springer, London (2011)
22. Deb, K.: Multi-objective optimization. In: Burke, E.K., Kendall, G. (eds.) Search Methodologies, pp. 403–449. Springer, Boston (2014)
23. Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Trans. Evol. Comput. **6**(2), 182–197 (2002)
24. Dehuri, S., Patnaik, S., Ghosh, A., Mall, R.: Application of elitist multi-objective genetic algorithm for classification rule generation. Appl. Soft Comput. **8**(1), 477–487 (2008). https://doi.org/10.1016/j.asoc.2007.02.009
25. Derrac, J., García, S., Herrera, F.: A survey on evolutionary instance selection and generation. Int. J. Appl. Metaheuristic Comput. **1**, 60–92 (2010)
26. Dietterich, T.G.: Ensemble methods in machine learning. In: Multiple Classifier Systems, MCS 2000. Lecture Notes in Computer Science, vol. 1857, pp. 1–15. Springer, Berlin (2000)

27. Díez-Pastor, J.F., Rodríguez, J.J., García-Osorio, C.I., Kuncheva, L.I.: Diversity techniques improve the performance of the best imbalance learning ensembles. Inf. Sci. **325**, 98–117 (2015)
28. Domingo-Perez, F., Lazaro-Galilea, J.L., Wieser, A., Martin-Gorostiza, E., Salido-Monzu, D., de la Llana, A.: Sensor placement determination for range-difference positioning using evolutionary multi-objective optimization. Expert. Syst. Appl. **47**, 95–105 (2016)
29. Dutta, D., Dutta, P., Sil, J.: Simultaneous feature selection and clustering with mixed features by multi objective genetic algorithm. Int. J. Hybrid Intell. Syst. **11**(1), 41–54 (2014). https://doi.org/10.3233/HIS-130182
30. Erickson, M., Mayer, A., Horn, J.: The niched Pareto genetic algorithm 2 applied to the design of groundwater remediation systems. Lect. Notes Comput. Sci. **1993**, 681–695 (2001)
31. Escalante, H.J., Marin-Castro, M., Morales-Reyes, A., Graff, M., Rosales-Pérez, A., y Gómez, M.M., Reyes, C.A., Gonzalez, J.A.: MOPG: a multi-objective evolutionary algorithm for prototype generation. Pattern. Anal. Applic. **20**(1), 33–47 (2015). https://doi.org/10.1007/s10044--015-0454--6
32. Fan, Y.J., Iyigun, C., Chaovalitwongse, W.A.: Recent advances in mathematical programming for classification and cluster analysis. In: CRM Proceedings & Lecture Notes, vol. 45, pp. 67–93 (2008)
33. Fernández, A., Carmona, C.J., del Jesus, M.J., Herrera, F.: A Pareto-based ensemble with feature and instance selection for learning from multi-class imbalanced datasets. Int. J. Neural Syst. **27**(06), 1750028 (2017). https://doi.org/10.1142/s0129065717500289
34. Fieldsend, J.E.: Optimizing decision trees using multi-objective particle swarm optimization. In: Studies in Computational Intelligence, pp. 93–114. Springer, Berlin (2009). https://doi.org/10.1007/978-3-642-03625-5_5
35. Floudas, C.A., Pardalos, P.M.: Encyclopedia of Optimization, vol. 2. Springer, Berlin (2001)
36. Fonseca, C.M., Fleming, P.J.: Genetic algorithms for multiobjective optimization: formulation, discussion and generalization. In: Proceedings of the Fifth International Conference on Genetic Algorithms (ICGA 1993), Urbana-Champaign, IL, USA, pp. 416–423 (1993)
37. Freitas, A.A.: A critical review of multi-objective optimization in data mining. ACM SIGKDD Explor. Newsl. **6**(2), 77 (2004). https://doi.org/10.1145/1046456.1046467
38. Fürnkranz, J.: Separate-and-conquer rule learning. Artif. Intell. Rev. **13**(1), 3–54 (1999)
39. Ganesan, T., Elamvazuthi, I., Vasant, P.: Multiobjective design optimization of a nano-CMOS voltage-controlled oscillator using game theoretic-differential evolution. Appl. Soft Comput. **32**, 293–299 (2015)
40. García, S., Luengo, J., Herrera, F.: Data Preprocessing in Data Mining. Springer, Cham (2015)
41. Giannopoulos, N., Nearchou, A.C.: Bi-criteria scheduling against restrictive common due dates using a multi-objective differential evolution algorithm. IMA J. Manag. Math. **29**(1), 119–136 (2018)
42. Giannopoulos, N., Moulianitis, V.C., Nearchou, A.C.: Multi-objective optimization with fuzzy measures and its application to flow-shop scheduling. Eng. Appl. Artif. Intell. **25**(7), 1381–1394 (2012)
43. Gong, W., Cai, Z., Zhu, L.: An efficient multiobjective differential evolution algorithm for engineering design. Struct. Multidiscip. Optim. **38**(2), 137–157 (2009)
44. Graves, A., Mohamed, A.R., Hinton, G.: Speech recognition with deep recurrent neural networks. In: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2013, pp. 6645–6649. IEEE, Piscataway (2013)
45. Gu, B., Sheng, V.S., Li, S.: Bi-parameter space partition for cost-sensitive SVM. In: Proceedings of the 24th International Conference on Artificial Intelligence, IJCAI'15, pp. 3532–3539. AAAI Press, Washington (2015). http://dl.acm.org/citation.cfm?id=2832581.2832741
46. Gu, S., Cheng, R., Jin, Y.: Multi-objective ensemble generation. Wiley Interdiscip. Rev. Data Min. Knowl. Disc. **5**(5), 234–245 (2015). https://doi.org/10.1002/widm.1158
47. Hahsler, M., Grün, B., Hornik, K.: A computational environment for mining association rules and frequent item sets. J. Stat. Softw. **14**, 1–25 (2005)

48. Han, M., Ren, W.: Global mutual information-based feature selection approach using single-objective and multi-objective optimization. Neurocomputing **168**, 47–54 (2015). https://doi.org/10.1016/j.neucom.2015.06.016
49. Hancer, E., Xue, B., Zhang, M., Karaboga, D., Akay, B.: Pareto front feature selection based on artificial bee colony optimization. Inf. Sci. **422**, 462–479 (2018)
50. Hearst, M.A., Dumais, S.T., Osuna, E., Platt, J., Scholkopf, B.: Support vector machines. IEEE Intell. Syst. Appl. **13**(4), 18–28 (1998)
51. Heckerman, D., Geiger, D., Chickering, D.M.: Learning Bayesian networks: the combination of knowledge and statistical data. Mach. Learn. **20**(3), 197–243 (1995)
52. Horn, J., Nafpliotis, N., Goldberg, D.E.: A niched Pareto genetic algorithm for multiobjective optimization. In: Proceedings of the first IEEE International Conference on Evolutionary Computation (ICEC 1994), Orlando, FL, USA, pp. 82–87 (1994)
53. Hruschka, E.R., Campello, R.J., Freitas, A.A., de Carvalho, A.: A survey of evolutionary algorithms for clustering. IEEE Trans. Syst. Man Cybern. Part C Appl. Rev. **39**(2), 133–155 (2009)
54. Hu, W., Tan, Y.: Prototype generation using multiobjective particle swarm optimization for nearest neighbor classification. IEEE Trans. Cybern. **46**(12), 2719–2731 (2016). https;//doi.org/10.1109/tcyb.2015.2487318
55. Huang, G.B., Zhou, H., Ding, X., Zhang, R.: Extreme learning machine for regression and multiclass classification. IEEE Trans. Syst. Man Cybern. B Cybern. **42**(2), 513–529 (2012)
56. Hwang, C.L., Masud, A.S.M.: Multiple Objective Decision Making Methods and Applications: A State-of-the-Art Survey, vol. 164. Springer, Berlin (2012)
57. Jin, Y., Sendhoff, B.: Pareto-based multiobjective machine learning: an overview and case studies. IEEE Trans. Syst. Man Cybern. Part C Appl. Rev. **38**(3), 397–415 (2008). https://doi.org/10.1109/tsmcc.2008.919172
58. Jones, D.F., Mirrazavi, S.K., Tamiz, M.: Multi-objective meta-heuristics: an overview of the current state-of-the-art. Eur. J. Oper. Res. **137**(1), 1–9 (2002)
59. Kennedy, J., Eberhart, R.: Particle swarm optimization. In: Proceedings of the IEEE International Conference on Neural Networks, vol. IV, pp. 1942–1948. IEEE, Piscataway (1995)
60. Khor, E.F., Tan, K.C., Lee, T.H., Goh, C.K.: A study on distribution preservation mechanism in evolutionary multi-objective optimization. Artif. Intell. Rev. **23**(1), 31–33 (2005)
61. Kodratoff, Y.: Introduction to Machine Learning. Morgan Kaufmann, Burlington (2014)
62. Kolari, P., Java, A., Finin, T., Oates, T., Joshi, A.: Detecting spam blogs: a machine learning approach. In: AAAI, vol. 6, pp. 1351–1356 (2006)
63. Kotsiantis, S.B., Zaharakis, I.D., Pintelas, P.E.: Machine learning: a review of classification and combining techniques. Artif. Intell. Rev. **26**(3), 159–190 (2006)
64. Lei, D.: Multi-objective production scheduling: a survey. Int. J. Adv. Manuf. Technol. **43**(9–10), 926–938 (2009)
65. Li, X.: A real-coded predator-prey genetic algorithm for multiobjective optimization. In: International Conference on Evolutionary Multi-Criterion Optimization, pp. 207–221. Springer, Berlin (2003)
66. Li, J., Taiwo, S.: Enhancing financial decision making using multi-objective financial genetic programming. In: IEEE Congress on Evolutionary Computation, 2006. CEC 2006, pp. 2171–2178. IEEE, Piscataway (2006)
67. Li, J., Fong, S., Wong, R.K., Chu, V.W.: Adaptive multi-objective swarm fusion for imbalanced data classification. Inf. Fusion **39**, 1–24 (2018). https://doi.org/10.1016/j.inffus.2017.03.007
68. Lim, D., Ong, Y.S., Jin, Y., Sendhoff, B., Lee, B.S.: Inverse multi-objective robust evolutionary design. Genet. Program Evolvable Mach. **7**(4), 383–404 (2006)
69. Liu, Z.g., Pan, Q., Dezert, J., Martin, A.: Adaptive imputation of missing values for incomplete pattern classification. Pattern Recogn. **52**, 85–95 (2016)
70. Lobato, F., Sales, C., Araujo, I., Tadaiesky, V., Dias, L., Ramos, L., Santana, A.: Multi-objective genetic algorithm for missing data imputation. Pattern Recogn. Lett. **68**, 126–131 (2015). https://doi.org/10.1016/j.patrec.2015.08.023

71. Luo, J., Jiao, L., Lozano, J.A.: A sparse spectral clustering framework via multiobjective evolutionary algorithm. IEEE Trans. Evol. Comput. **20**(3), 418–433 (2016). https://doi.org/10.1109/tevc.2015.2476359
72. Malhotra, S., Bali, V., Paliwal, K.: Genetic programming and k-nearest neighbour classifier based intrusion detection model. In: 7th International Conference on Cloud Computing, Data Science and Engineering-Confluence, 2017, pp. 42–46. IEEE, Piscataway (2017)
73. Martín, D., Rosete, A., Alcalá-Fdez, J., Herrera, F.: QAR-CIP-NSGA-II: A new multi-objective evolutionary algorithm to mine quantitative association rules. Inf. Sci. **258**, 1–28 (2014). https://doi.org/10.1016/j.ins.2013.09.009
74. Mason, K., Duggan, J., Howley, E.: A multi-objective neural network trained with differential evolution for dynamic economic emission dispatch. Int. J. Electr. Power Energy Syst. **100**, 201–221 (2018)
75. Minaei-Bidgoli, B., Barmaki, R., Nasiri, M.: Mining numerical association rules via multi-objective genetic algorithms. Inf. Sci. **233**, 15–24 (2013). https://doi.org/10.1016/j.ins.2013.01.028
76. Miranda, P.B., Prudêncio, R.B., de Carvalho, A.P., Soares, C.: A hybrid meta-learning architecture for multi-objective optimization of SVM parameters. Neurocomputing **143**, 27–43 (2014). https://doi.org/10.1016/j.neucom.2014.06.026
77. Mukhopadhyay, A., Maulik, U., Bandyopadhyay, S., Coello Coello, C.A.: A survey of multiobjective evolutionary algorithms for data mining: Part I. IEEE Trans. Evol. Comput. **18**(1), 4–19 (2014). https://doi.org/10.1109/tevc.2013.2290086
78. Mukhopadhyay, A., Maulik, U., Bandyopadhyay, S., Coello Coello, C.A.: Survey of multi-objective evolutionary algorithms for data mining: Part II. IEEE Trans. Evol. Comput. **18**(1), 20–35 (2014). https://doi.org/10.1109/tevc.2013.2290082
79. Mukhopadhyay, A., Maulik, U., Bandyopadhyay, S.: A survey of multiobjective evolutionary clustering. ACM Comput. Surv. **47**(4), 1–46 (2015). https://doi.org/10.1145/2742642
80. Nag, K., Pal, N.R.: A multiobjective genetic programming-based ensemble for simultaneous feature selection and classification. IEEE Trans. Cybern. **46**(2), 499–510 (2016). https://doi.org/10.1109/tcyb.2015.2404806
81. Nayak, S.K., Rout, P.K., Jagadev, A.K.: Automatic clustering by elitism-based multi-objective differential evolution. Int. J. Manag. Decis. Mak. **17**(1), 50–74 (2018)
82. Newman, M.E.: Power laws, Pareto distributions and Zipf's law. Contemp. Phys. **46**(5), 323–351 (2005)
83. Nguyen, T.T., Liew, A.W.C., Pham, X.C., Nguyen, M.P.: Optimization of ensemble classifier system based on multiple objectives genetic algorithm. In: 2014 International Conference on Machine Learning and Cybernetics. IEEE, Piscataway (2014). https://doi.org/10.1109/icmlc.2014.7009090
84. Ojha, V.K., Abraham, A., Snášel, V.: Ensemble of heterogeneous flexible neural trees using multiobjective genetic programming. Appl. Soft Comput. **52**, 909–924 (2017). https://doi.org/10.1016/j.asoc.2016.09.035
85. Oza, N.C., Tumer, K.: Classifier ensembles: select real-world applications. Inf. Fusion **9**(1), 4–20 (2008)
86. Panda, M.: Combining multi-objective evolutionary algorithm with averaged one-dependence estimators for big data analytics. Int. J. Comput. Intell. Stud. **7**(1), 1–18 (2018)
87. Pappa, G.L., Freitas, A.A.: Evolving rule induction algorithms with multi-objective grammar-based genetic programming. Knowl. Inf. Syst. **19**(3), 283–309 (2008). https://doi.org/10.1007/s10115--008-0171--1
88. Pardalos, P.M., Hansen, P.: Data Mining and Mathematical Programming, vol. 45. American Mathematical Society, Providence (2008)
89. Pardalos, P.M., Žilinskas, A., Žilinskas, J.: Multi-objective branch and bound. In: Non-convex Multi-Objective Optimization, pp. 45–56. Springer, Cham (2017)
90. Pareto, V.: Manuale di economica politica, societa editrice libraria. In: Manual of Political Economy, vol. 1971. A.M. Kelley, New York (1906)

91. Parsopoulos, K.E., Vrahatis, M.N.: Particle swarm optimization method in multiobjective problems. In: Proceedings of the ACM Symposium on Applied Computing (SAC 2002), pp. 603–607. ACM, New York (2002)
92. Parsopoulos, K.E., Vrahatis, M.N.: Multi-objective particle swarm optimization approaches. In: Bui, L.T., Alam, S. (eds.) Multi-Objective Optimization in Computational Intelligence: Theory and Practice, Chapter 2, pp. 20–42. IGI Global, Hershey (2008)
93. Parsopoulos, K.E., Vrahatis, M.N.: Particle Swarm Optimization and Intelligence: Advances and Applications. Information Science Publishing (IGI Global), Hershey (2010)
94. Parsopoulos, K.E., Tasoulis, D.K., Vrahatis, M.N.: Multiobjective optimization using parallel vector evaluated particle swarm optimization. In: Hamza, M.H. (ed.) Artificial Intelligence and Applications, vol. 2, pp.823–828. ACTA Press, Anaheim (2004)
95. Parsopoulos, K.E., Tasoulis, D.K., Pavlidis, N.G., Plagianakos, V.P., Vrahatis, M.N.: Vector evaluated differential evolution for multiobjective optimization. In: Proceedings of the IEEE Congress on Evolutionary Computation (CEC 2004), vol. 1, pp. 204–211. IEEE, Piscataway (2004)
96. Paul, S., Das, S.: Simultaneous feature selection and weighting – an evolutionary multi-objective optimization approach. Pattern Recogn. Lett. **65**, 51–59 (2015). https://doi.org/10.1016/j.patrec.2015.07.007
97. Pena-Reyes, C.A., Sipper, M.: Evolutionary computation in medicine: an overview. Artif. Intell. Med. **19**(1), 1–23 (2000)
98. Pourtaheri, Z.K., Zahiri, S.H., Razavi, S.M.: Stability investigation of multi-objective heuristic ensemble classifiers. Int. J. Mach. Learn. Cybern. 1–13 (2018). https://doi.org/10.1007/s13042-018-0789-6
99. Quinlan, J.R.: Induction of decision trees. Mach. Learn. **1**(1), 81–106 (1986)
100. Rao, N.M., Kannan, K., Gao, X.-Z., Roy, D.S.: Novel classifiers for intelligent disease diagnosis with multi-objective parameter evolution. Comput. Electr. Eng. **67**, 483–496. https://doi.org/10.1016/j.compeleceng.2018.01.039. Available online 13 February 2018 (2018)
101. Rodríguez, J.D., Lozano, J.A.: Multi-objective learning of multi-dimensional Bayesian classifiers. In: 2008 Eighth International Conference on Hybrid Intelligent Systems. IEEE, Piscataway (2008). https://doi.org/10.1109/his.2008.143
102. Rokach, L., Maimon, O.: Data Mining with Decision Trees: Theory and Applications. World Scientific, Singapore (2008)
103. Rosales-Perez, A., Garcia, S., Gonzalez, J.A., Coello Coello, C.A., Herrera, F.: An evolutionary multiobjective model and instance selection for support vector machines with Pareto-based ensembles. IEEE Trans. Evol. Comput. **21**(6), 863–877 (2017). https://doi.org/10.1109/tevc.2017.2688863
104. Rudziński, F.: A multi-objective genetic optimization of interpretability-oriented fuzzy rule-based classifiers. Appl. Soft Comput. **38**, 118–133 (2016). https://doi.org/10.1016/j.asoc.2015.09.038
105. Sabar, N.R., Yi, X., Song, A.: A bi-objective hyper-heuristic support vector machines for big data cyber-security. IEEE Access **6**, 10421–10431 (2018)
106. Samarasinghe, S.: Neural Networks for Applied Sciences and Engineering: From Fundamentals to Complex Pattern Recognition. CRC Press, Boca Raton (2016)
107. Schalkoff, R.J.: Artificial neural networks, vol. 1. McGraw-Hill, New York (1997)
108. Schmidhuber, J.: Deep learning in neural networks: an overview. Neural Netw. **61**, 85–117 (2015)
109. Shen, C., Wang, X., Yu, D.: Feature weighting of support vector machines based on derivative saliency analysis and its application to financial data mining. Int. J. Adv. Comput. Technol. **4**(1), 199–206 (2012)
110. Shu, W., Shen, H.: Multi-criteria feature selection on cost-sensitive data with missing values. Pattern Recogn. **51**, 268–280 (2016)
111. Soysal, Ö.M.: Association rule mining with mostly associated sequential patterns. Expert Syst. Appl. **42**(5), 2582–2592 (2015)

112. Spall, J.C.: Introduction to Stochastic Search and Optimization: Estimation, Simulation, and Control, vol. 65. Wiley, Hoboken (2005)
113. Sra, S., Nowozin, S., Wright, S.J.: Optimization for Machine Learning. MIT Press, Cambridge (2012)
114. Srinivas, N., Deb, K.: Multiobjective optimization using nondominated sorting in genetic algorithms. Evol. Comput. **2**(3), 221–248 (1994). https://doi.org/10.1162/evco.1994.2.3.221
115. Srinivasan, S., Ramakrishnan, S.: Evolutionary multi objective optimization for rule mining: a review. Artif. Intell. Rev. **36**(3), 205–248 (2011). https://doi.org/10.1007/s10462-011-9212-3
116. Tahan, M.H., Asadi, S.: MEMOD: a novel multivariate evolutionary multi-objective discretization. Soft Comput. **22**, 301–323 (2017). https://doi.org/10.1007/s00500-016-2475-5
117. Tan, C.J., Lim, C.P., Cheah, Y.: A multi-objective evolutionary algorithm-based ensemble optimizer for feature selection and classification with neural network models. Neurocomputing **125**, 217–228 (2014). https://doi.org/10.1016/j.neucom.2012.12.057
118. Veldhuizen, D.A.V., Lamont, G.B.: Multiobjective evolutionary algorithms: analyzing the state-of-the-art. Evol. Comput. **8**(2), 125–147 (2000)
119. Viola, P., Jones, M.J.: Robust real-time face detection. Int. J. Comput. Vis. **57**(2), 137–154 (2004)
120. von Lücken, C., Barán, B., Brizuela, C.: A survey on multi-objective evolutionary algorithms for many-objective problems. Comput. Optim. Appl. **58**(3), 707–756 (2014)
121. Wang, Z., Li, M., Li, J.: A multi-objective evolutionary algorithm for feature selection based on mutual information with a new redundancy measure. Inf. Sci. **307**, 73–88 (2015). https://doi.org/10.1016/j.ins.2015.02.031
122. Wang, R., Lai, S., Wu, G., Xing, L., Wang, L., Ishibuchi, H.: Multi-clustering via evolutionary multi-objective optimization. Inf. Sci. **450**, 128–410 (2018)
123. Witten, I.H., Frank, E., Hall, M.A., Pal, C.J.: Data Mining: Practical Machine Learning Tools and Techniques. Morgan Kaufmann, Burlington (2016)
124. Wong, T.T.: A hybrid discretization method for naïve Bayesian classifiers. Pattern Recogn. **45**(6), 2321–2325 (2012)
125. Xue, B., Zhang, M., Browne, W.N.: Particle swarm optimization for feature selection in classification: A multi-objective approach. IEEE Trans. Cybern. **43**(6), 1656–1671 (2013). https://doi.org/10.1109/tsmcb.2012.2227469
126. Xue, B., Zhang, M., Browne, W.N., Yao, X.: A survey on evolutionary computation approaches to feature selection. IEEE Trans. Evol. Comput. **20**(4), 606–626 (2016). https://doi.org/10.1109/tevc.2015.2504420
127. Zhao, H.: A multi-objective genetic programming approach to developing Pareto optimal decision trees. Decis. Support. Syst. **43**(3), 809–826 (2007). https://doi.org/10.1016/j.dss.2006.12.011
128. Zhao, J., Fernandes, V.B., Jiao, L., Yevseyeva, I., Maulana, A., Li, R., Back, T., Tang, K., Emmerich, M.T.: Multiobjective optimization of classifiers by means of 3D convex-hull-based evolutionary algorithms. Inf. Sci. **367-368**, 80–104 (2016). https://doi.org/10.1016/j.ins.2016.05.026
129. Zheng, K., Wang, X.: Feature selection method with joint maximal information entropy between features and class. Pattern Recogn. **77**, 20–29 (2018)
130. Zhou, A., Qu, B.Y., Li, H., Zhao, S.Z., Suganthan, P.N., Zhang, Q.: Multiobjective evolutionary algorithms: a survey of the state of the art. Swarm Evol. Comput. **1**(1), 32–49 (2011). https://doi.org/10.1016/j.swevo.2011.03.001
131. Zitzler, E., Thiele, L.: Multiobjective evolutionary algorithms: a comparative case study and the strength Pareto approach. IEEE Trans. Evol. Comput. **3**(4), 257–271 (1999) https://doi.org/10.1109/4235.797969
132. Zitzler, E., Deb, K., Thiele, L.: Comparison of multiobjective evolutionary algorithms: empirical results. Evol. Comput. **8**(2), 173–195 (2000)