An Evolutionary Multi-Objective Model and Instance Selection for Support Vector Machines with Pareto-based Ensembles

Alejandro Rosales-Pérez, Salvador García, Jesus A. Gonzalez, Carlos A. Coello Coello, *Fellow, IEEE,* Francisco Herrera, *Senior Member, IEEE*

***Abstract*—Support Vector Machines are among the most powerful learning algorithms for classification tasks. However, these algorithms require a high computational cost during the training phase, which can limit their application on large-scale datasets. Moreover, it is known that their effectiveness highly depends on the hyper-parameters used to train the model. With the intention of dealing with these, this paper introduces an Evolutionary Multi-Objective Model and Instance Selection approach for support vector machines with Pareto-based Ensemble, whose goals are, precisely, to optimize the size of the training set and the classification performance attained by the selection of the instances, which can be done using either a wrapper or a filter approach. Due to the nature of multi-objective evolutionary algorithms, several Pareto optimal solutions can be found. We study several ways of using such information to perform a classification task. To accomplish this, our proposal performs a processing over the Pareto solutions in order to combine them into a single ensemble. This is done in five different ways, which are based on a global Pareto ensemble, error reduction, a complementary error reduction, maximized margin distance and boosting. Through a comprehensive experimental study we evaluate the suitability of the proposed approach and the Pareto processing, and we show its advantages over a single-objective formulation, traditional instance selection techniques and learning algorithms.**

***Index Terms*—Instance Selection, Model Selection, Multi-Objective Optimization, Support Vector Machines.**

I. Introduction

Support Vector Machines (SVMs) [1] are kinds of supervised learning algorithms that can be used either for classification or regression tasks. The SVM is among the most powerful learning algorithms reported in the literature. The popularity of this method relies on its strong theoretical foundations and high performance. However, training an SVM requires solving a constrained quadratic programming optimization problem. Solving this optimization problem has a computational complexity of $\mathcal{O}\left(|\mathcal{T}|^3\right)$ [2], where $|\mathcal{T}|$ is the number of samples (or instances) in the training set. In spite of this, there are some approaches [3], [4] that reduce this computational cost to $\mathcal{O}\left(|\mathcal{T}|^2\right)$. This computational cost can still be computationally prohibitive on large-scale datasets, which can severely limit their applicability on such problems.

The goal of training an SVM is to find the hyperplane that maximizes the separation between two classes. The hyperplane is defined by a set of instances called *support vectors*, which are selected from the entire training set. The size of the support vectors set is usually smaller than the training set size. Therefore, selecting instances from the training set that have a high probability of becoming support vectors can help improve the efficiency of this method.

Besides the high computational cost in training an SVM, another major issue to be taken into consideration is the definition of the kernel function and its parameters (called hyper-parameters) which play an important role in the SVM's effectiveness; choosing the appropriate set of hyper-parameters is sometimes referred to as *model selection* (MS) [5]–[7]. Notwithstanding, most of the studies developed in this regard have focused on optimizing the parameters for a given kernel type and little attention has been paid to consider it alongside a pre-processing of the training set.

Data pre-processing is one of the major steps in the data mining process and can have a significant impact on the generalization performance of a machine learning algorithm [8]. Data, however, can have inconsistencies or irrelevant information that might make harder for learning algorithms find useful patterns. Furthermore, the growing amount of data has given raise the so-called big data, which demands more complex mechanisms, such as deep learning [9], to analyze it. Data pre-processing, on the other hand, can adapt the data to fulfill the requirements to be processed by a learning algorithm at the time that the performance of the learned models is enhanced [10].

This work was supported by the Spanish National Research Project TIN2014-57251-P and by CONACyT project 221551.

A. Rosales-Pérez is with the School of Engineering and Science, Tecnologico de Monterrey, Monterrey, N.L., 64849, Mexico

S. García and F. Herrera are with the Department of Computer Science and Artificial Intelligence, University of Granada, Granada, E-18071, Spain. F. Herrera is also with the Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah, 21859, Saudi Arabia.

J. A. Gonzalez is with the Computer Science Department, Instituto Nacional de Astrofísica, Óptica y Electrónica (INAOE), Puebla, 72840, Mexico

C. A. Coello Coello is with the Computer Science Department, Centro de Investigación y de Estudios Avanzados del IPN (CINVESTAV-IPN), Mexico City, 07360, Mexico

Outstanding applications in which data pre-processing plays an important role on a large-scale problem are described in [11], [12].

Data pre-processing includes data cleaning, data transformation, and data reduction, which encompasses the selection and extraction of both features and instances from a dataset. One of the most influential data pre-processing is the Instance Selection (IS) [10]. IS aims at selecting a relevant instance subset of the entire training set, while preserving the performance of the whole dataset. There are a number of IS techniques reported in the literature, which have been mainly focused on the k-nearest neighbor (k-NN) classifier [13]–[17]. In recent years, there has been a growing interest for exploring IS techniques for SVMs (e.g. [18]–[24]) as an alternative for handling the computational cost in the training, as well as reducing the complexity of the classifier and improving its performance by removing noise instances.

To the best of the authors' knowledge, the instance set selection problem for an SVM in combination with the selection of its hyper-parameters has not been previously studied. Taking the aforementioned into account, in this paper we propose to tackle both problems simultaneously. These two goals, the reduction in the training set and the SVM's effectiveness, can be approached in a natural fashion as a multi-objective optimization problem (MOP).

Evolutionary algorithms (EAs) have gained popularity for solving multi-objective problems; these are usually called multi-objective EAs (MOEAs). Unlike single-objective optimization, where the goal is to find a single solution, in multi-objective optimization several trade-off solutions are usually present, which are called non-dominated or Pareto optimal solutions. In recent years, MOEAs have been successfully applied in problems from the domains of data-mining and supervised learning [25]–[28], such as feature selection [29]–[31], association rules mining [32]–[34], clustering [35], [36], and classification [37]–[40]. Here, we investigate the use of a MOEA for addressing classification problems with an SVM as well as the different forms of handling the several trade-off solutions available, in order to construct only one classification model.

This paper introduces EMOMIS-PbE: Evolutionary Multi-Objective Model and Instance Selection with Pareto-based Ensemble, a novel multi-objective algorithm that aims at explicitly and simultaneously optimizing the size of the training set, by means of an instance selection, and the performance of an SVM, by means of an appropriate selection of hyper-parameters. EMOMIS-PbE is able to perform the IS either using a wrapper or a filter approach. Moreover, EMOMIS-PbE also performs a post-processing over the Pareto solutions for the sake of combining them into a single ensemble. This is achieved through five different strategies, which are a global Pareto ensemble, an error reduction ensemble approach, complementary error reduction, margin distance, and boosting. The contributions of this paper are the following:

- The hybridization of well-known IS techniques (ENN, FCNN, HMN-EI, DROP3, and RNGE) and MS for SVMs through a multi-objective formulation. To the best of our knowledge, this is the first attempt to use MOEAs to address both IS and MS for SVMs.
- The multi-objective formulation of the problem naturally allows generating a Pareto set with several and diverse SVM models. We study different Pareto-based ensemble strategies for using the information from these solutions to construct a single classification model.
- Through several experiments, we validate our proposal over an extensive number of benchmark datasets and with a comprehensive statistical analysis.

We assess EMOMIS-PbE over a suite of 43 well-known classification problems. We have developed the following studies. First, we perform a comparative study among the Pareto-based ensemble strategies. Second, we evaluate the suitability of performing MS against not doing so, and the advantages of performing IS against using the entire training set. Third, we evaluate the benefit of the Pareto-based ensemble post-processing against choosing a single solution. Fourth, we compare our proposed method with respect to those generated by traditional IS techniques (ENN, FCNN, HMN-EI, DROP3, and RNGE). Fifth, EMOMIS-PbE is contrasted against traditional learning algorithms. The experimental results, validated with statistical tests, show the feasibility of our proposal, being able to reach solutions with a good trade-off between accuracy performance and reduction rate.

The remainder of this paper is organized as follows. Section II describes the multi-objective optimization problem focusing on evolutionary algorithms. It also describes some previous related work, focusing mainly on the instance selection problem. Section III presents the multi-objective proposal to be adopted with the model and instance selection problems. Next, Section IV details the experimental settings and presents the results and statistical tests to validate our proposal. Finally, the main conclusions and future research directions are presented in Section V.

II. Preliminaries

This section describes some basic concepts related to multi-objective optimization and to instance selection.

A. Multi-Objective Evolutionary Algorithms

A general MOP can be stated as follows:

$$\begin{aligned} &\text{minimize } \mathbf{f}(\mathbf{x}) = [f_1(\mathbf{x}), \ldots, f_l(\mathbf{x})]^T \\ &\text{subject to } \mathbf{x} \in \mathcal{X} \end{aligned} \tag{1}$$

where $\mathbf{x} = [x_1, \ldots, x_n]^T \in \mathbb{R}^n$ is a decision variables vector, $f_i(\mathbf{x})$, $i = 1, \ldots, l$, are the l-objective functions, and $\mathcal{X}$ is the set of feasible solutions.

Generally, the objectives in a MOP are conflicting, and therefore a single best solution for all of them does not exist. In these cases, we use Pareto optimality. We say that a solution $\mathbf{x}^1$ dominates a solution $\mathbf{x}^2$ (denoted by $\mathbf{x}^1 \preceq \mathbf{x}^2$) if and only if $\mathbf{x}^1$ is better than $\mathbf{x}^2$ at least in one objective and is not worse in the rest, i.e.,

$$\forall i : f_i\left(\mathbf{x}^1\right) \leq f_i\left(\mathbf{x}^2\right) \wedge \exists i : f_i\left(\mathbf{x}^1\right) < f_i\left(\mathbf{x}^2\right) \tag{2}$$

A solution $\mathbf{x}^*$ is a Pareto optimal solution if another solution $\mathbf{x}' \in \mathcal{X}$ does not exist such that $\mathbf{x}' \preceq \mathbf{x}^*$. One should note that this definition does not produce a single solution, but a set of trade-off solutions; this set is called Pareto optimal set, and the image of this set in objective function space is referred to as the Pareto Front.

EAs are stochastic search techniques inspired by Darwin's evolutionary theory. These algorithms have gained popularity within multi-objective optimization due to the several advantages that they offer with respect to mathematical programming techniques. For instance, MOEAs can obtain several elements of the Pareto optimal set in a single run, instead of generating one at a time (as normally done by mathematical programming techniques). Moreover, MOEAs are less susceptible than mathematical programming techniques to the shape and continuity of the Pareto front [41], [42].

Since Schaffer's seminal work [43], a large number of MOEAs have been proposed. Among them, we can find the Strength Pareto Evolutionary Algorithm 2 (SPEA2) [44], the Pareto Archived Evolution Strategy (PAES) [45], the Non-dominated Sorting Genetic Algorithm II (NSGA-II) [46], and the Multi-Objective Evolutionary Algorithm based on Decomposition (MOEA/D) [47], etc. A comprehensive review of MOEAs can be found in [41], [42], [48].

In this study, we used MOEA/D, due to its high performance on a variety of difficult problems [49]. Additionally, MOEA/D has a lower computational complexity than other MOEAs (such as the NSGA-II), and is able to provide well-distributed solutions along the Pareto front [47]. The latter is important since it provides the user with solutions that represent different trade-offs among the objectives, which is very helpful for the decision making process.

MOEA/D is presented in Algorithm 1. The basic idea is to decompose a MOP into a number of scalar optimization problems, called subproblems, through a weighted aggregation of the objectives. A neighborhood relation based on the distance of the aggregation weights vectors is defined among the subproblems. The optimal solutions to two neighboring subproblems should be very similar. The best solution in the population is found in each subproblem and is optimized in MOEA/D by using information from its neighbors. A new solution $\mathbf{y}$ is generated by using a selection operator for choosing the parents, which are used by the crossover operator to produce the new solution. A mutation operator is used to modify the new solution. After that, the reference point $\mathbf{z}$ is updated, as well as the neighboring solutions. The external population EP stores all the non-dominated solutions found so far during the search.

Algorithm 1 MOEA/D [47]

Require: A stopping criterion,
N: number of subproblems considered in MOEA/D,
A uniform spread of N weight vectors: $\lambda^1, \ldots, \lambda^N$,
T: the number of weight vectors in the neighborhood of each weight vector

Ensure: EP: an external population

Initialize $EP \rightarrow \emptyset$
Compute the Euclidean distance between any two weight vectors and then work out the T closest weight vectors to each weight vector. For each $i = 1, \ldots, N$, set $B(i) = \{i_1, \ldots, i_T\}$, where $\lambda^{i_1}, \ldots, \lambda^{i_T}$ are the closest weight vectors to λ^i.
Generate an initial population $\mathbf{x}^1, \ldots, \mathbf{x}^N$
Initialize $\mathbf{z} = [z_1, \ldots, z_m]$ by setting $z_j = \min_{1 \leq i \leq N} f_j\left(\mathbf{x}^i\right)$
while stopping criterion is not satisfied **do**
  for $i = 1$ **to** N **do**
    Randomly select two indexes k, l from $B(i)$, and then generate a new solution $\mathbf{y}$ from $\mathbf{x}^k$ and $\mathbf{x}^l$ by using genetic operators.
    Apply a problem-specific repair/improvement heuristic on $\mathbf{y}$ to produce $\mathbf{y}'$.
    Update $\mathbf{z}$, for each $j = 1, \ldots, m$ if $z_j > f_j(\mathbf{y})$, then set $z_j = f_j(\mathbf{y})$
    Update of Neighboring Solutions: For each index $j \in B(i)$, if $g^{te}\left(\mathbf{y}'\lambda^j, \mathbf{z}\right) \leq g\left(\mathbf{x}^j\lambda^j, \mathbf{z}\right)$, then set $\mathbf{x}^j = \mathbf{y}'$, $FV^j = F(\mathbf{y}')$
    Update of EP: Add $F(\mathbf{y}')$ to EP if it is non-dominated with respect to the vectors stored in EP, and remove from EP the vectors dominated by $F(\mathbf{y}')$.
  end for
end while

A key issue in MOEA/D is to decompose the problem into subproblems; to this end, we used the Tchebycheff approach [50]. In the Tchebycheff approach, a MOP is decomposed into N scalar optimization subproblems as follows:

$$minimize\ g(\mathbf{x} \mid \lambda, \mathbf{z}^*) = max_{1 \leq i \leq m}\{\lambda_i \mid f_i(\mathbf{x}) - \mathbf{z}_i^* \mid\} \tag{3}$$

where $\lambda = [\lambda_1, \ldots, \lambda_m]$ is a weight vector, $\mathbf{z}^* = [z_1, \ldots, z_m]$ is a reference point, and m is the number of objectives in the problem.

B. Instance Selection

A review, categorization, and experimental comparison among different state of the art IS techniques is reported in [15]. There, IS techniques are classified according to different criteria, including direction search, the type of selection, and the evaluation search. Here, we focus on the type of selection and summarize the main conclusions of this paper.

The type of selection adopted distinguishes the IS techniques by means of whether they seek to retain border points, central points or some other set of points [15]. Hence, according to this criterion, IS techniques are classified in three main groups:

1) ***Edition***: This group seeks to remove points that are noisy or that do not agree with their nearest

neighbors. An example of this group is the Edited Nearest Neighbor (ENN) [51], which removes noisy points and those points close to the decision boundary following a simple but effective rule. This rule consists of removing an instance $\mathbf{x}$ if it does not agree with the majority of its k nearest neighbors. This yields to smoother decision boundaries. In the experimental evaluation performed in [15], the Relative Neighborhood Graph Editing (RNGE) [17] is one of the most effective edition methods. RNGE constructs a proximity graph and an instance $\mathbf{x}$ is removed if it is incorrectly classified by its neighbors in the graph.

2) ***Condensation***: This group tries to retain the points that are closer to decision boundaries. It is based on the idea that the internal points do not affect the decision boundaries as the border points do. In this group, the Fast Condensed Nearest Neighbor (FCNN) [15] stands out. FCNN starts with an empty set of instances, $\mathcal{S}$, and it runs over all instances in the training set. An instance is added to the set $\mathcal{S}$ if it is wrongly classified by its k nearest neighbors when $\mathcal{S}$ is used as the training set.
3) ***Hybrid***: In this group, both internal and border points are allowed to be removed. Two outstanding examples of this group are the Decremental Reduction Optimization Procedure (DROP) [52] and the Hit-Miss Network Edition Iterative (HMN-EI) [16] algorithm. In DROP, an instance $\mathbf{x}$ is removed if its associates can be correctly classified without such an instance $\mathbf{x}$. The authors proposed five variants of these algorithms and, according to their experimental evaluation, DROP3 and DROP5 were the best. On the other hand, HMN-EI constructs a directed graph called Hit-Miss network that accounts for the degree in which an instance $\mathbf{x}$ is the nearest neighbor of others belonging to the same class (hit) and those belonging to different classes (miss); the removal of instances is based on a set of rules.

The above mentioned techniques have been extensively studied for the k-NN classifier, but they can be considered as representative methods of each class [15] and are used in the present study. In the context of Support Vector Machines (SVMs), there are some approaches with the aim of reducing the training set size. For instance, in [23], a random sampling technique is applied for selecting training instances for an SVM. In [53], the authors propose the Multi-Class Instance Selection (MCIS) technique, which is used to reduce the training set size in a multi-class dataset. The multi-class problem with the SVM is approached by following a one-vs-all scheme. Another method designed for instance selection for an SVM is the Sparsifying Neural Gas SVM [24], in which the idea is to select instances that are very likely to become support vectors. In [21], the authors use a single-objective memetic algorithm to perform instance selection for an SVM. In [20], an ensemble margin-based instance selection is proposed. In this approach, the idea is to select the instances that have a low margin. Dornaika [18] proposes a more general approach, called decremental sparse modeling, which selects representative instances from the training set. The selected instances are used to train different classifiers, such as k-NN, Nearest Subspace, and SVMs. One should note that these algorithms are designed to improve the efficiency in the training phase of a support vector machine, but none of these have dealt with the problem of choosing the kernel and hyper-parameters simultaneously in order to improve the classification performance of the SVM.

We are aware that there are some proposals (e.g. [5]–[7], [54]) that combine the model selection task with some type of data pre-processing. In [6] the authors deal with the model selection problem for an SVM, in which the model also considers feature selection and data normalization/standardization. Sun et al. [7] and Thornton [54] consider different feature selection and data pre-processing techniques. However, instance selection is not taken into account; thus, our proposal is novel in this sense.

III. EMOMIS: Evolutionary Multi-Objective Model and Instance Selection

The proposed approach formulates the simultaneous instance and hyper-parameter selection problem as a multi-objective one. Figure 1 shows the diagram of the proposed approach. The process starts, as usually happens with evolutionary algorithms, with the creation of an initial population, in which each individual in the population encodes a possible solution to the problem at hand; i.e., a possible set of hyper-parameters for the SVM and a way of reducing the training set. For each individual, how well the set of hyper-parameters performs on the entire training set as well as to what extent it is possible to reduce the training set size is computed. After that, new individuals are created by applying evolutionary operators over the existing ones and the individuals that satisfy the best trade-off are kept in an external archive. This process is repeated iteratively until a stop criterion is met. The proposed approach is detailed in the following.

A. Representation

EAs work with a population of individuals, where each individual encodes a potential solution to the optimization problem; i.e., the set of hyper-parameters for an SVM and the method for reducing the training set size. The first step is to define how to encode a solution to the problem into an individual. In this paper, we explore two different encodings which are shown in Figure 2. Since one of the main goals is to reduce the training set size, this can be achieved by either following a filter (EMOMFIS) or a wrapper (EMOMWIS) approach for IS. For filter instance selection, we consider five well-known traditional techniques. These are ENN, FCNN, RNGE, DROP3, and HMN-EI. The first four filters have a parameter that indicates the number of nearest neighbors to be considered

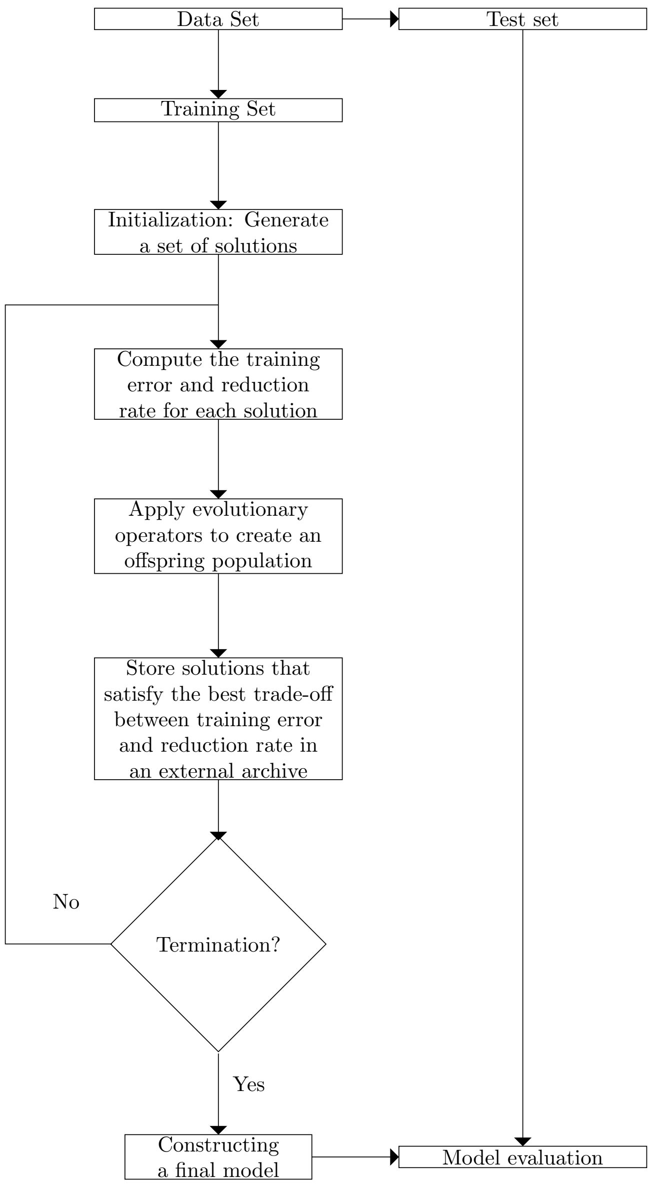


Fig. 1: General scheme for dealing with the problem of model and instance selection.

by the algorithm, which is encoded as an integer value. The latter has a real-valued parameter in the range from 0 to 1, which represents a threshold for a removing rule. In the case of the wrapper, the instances are encoded with a binary variable indicating whether the corresponding instance is used or not.

For both filter and wrapper instance selection, we perform the selection of the SVM's hyper-parameters. These hyper-parameters are the penalty parameter C, the type of kernel function (1-linear, 2-polynomial, and 3-RBF), and the kernel's parameters, such as γ in RBF, or the polynomial degree d. Since SVM is designed for binary classification problems, the parameter OVO/A is used to indicate if a multi-class problem is approached through either the OVO or the OVA decomposition. Figure 2 shows examples of the chromosome for each of these cases.

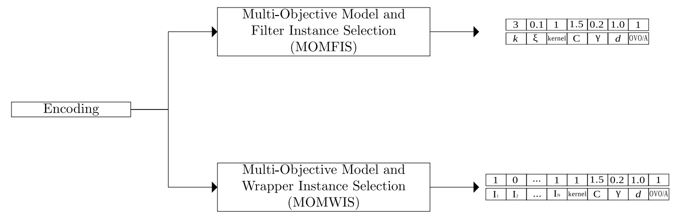


Fig. 2: The two different encodings adopted in this paper for the multi-objective model and instance selection problem.

B. Initialization and Evolutionary Operators

Once the encoding is defined, the next step is to create an initial population that properly represents the search space. The Latin hyper-cube technique [55] is used to this end. A Latin hyper-cube is constructed such that each of the optimization variables is divided into N equal levels, where N is the population size and there is only one point (individual) at each level. To determine the location of each point, the Latin hyper-cube sampling technique maximizes the minimum distance between pairs of points, with the aim of spreading them out as much as possible inside the search space. In this manner, we ensure that we have a representative initial population from the entire search space.

The initial population is evaluated with the fitness functions and they enter in an iterative process to be evolved by means of evolutionary operators. As evolutionary operators, we used uniform crossover for integer/binary representations and SBX for the real part. As a mutation operator, uniform mutation is used for the integer representation, bit-flip for the binary part, and polynomial mutation for the real part. In this way, we apply an appropriate operator according to the type of encoding.

C. Fitness Functions

The fitness functions are in charge of determining how well a solution performs. Our aim is to optimize both the error rate (f_1), and the reduction rate (f_2) attained on the training set, which are computed as follows:

$$\begin{aligned} f_1 &= \frac{1}{|\mathcal{T}|}\sum_{i=1}^{|\mathcal{T}|}\mathcal{L}\left(y_i, y_i^*\right) \\ f_2 &= \frac{|\mathcal{S}|}{|\mathcal{T}|} \end{aligned} \tag{4}$$

where $\mathcal{T}$ and $\mathcal{S}$ are, respectively, the training and reduced sets, y_i and y_i^* are the class label and the predicted class of an instance i, and $\mathcal{L}\left(y_i, y_i^*\right)$ is a function that measures the expected classification performance. In our case, we minimize a measure of the generalization bound for the SVM [56], which can be computed as:

$$\mathcal{L}\left(y_i, y_i^*\right) = e_{y_i, y_i^*} + \sqrt{\frac{\log\left(nsv\right) + log\left({}^{1}/_{\delta}\right)}{2\left|\mathcal{T}\right|}} \tag{5}$$

where e_{y_i,y_i^*} is the error rate in the training set, nsv is the number of support vectors and δ is the probability that the true error is greater than the estimated one.

This expression allows getting an insight of the generalization error without doing several experiments, such as cross validation, that can increase the computational cost.

Taking into account these two criteria, the proposed approach aims at exploring the space of hyper-parameters and IS techniques in order to find the solutions that satisfy the best trade-off. One should recall that the result of a multi-objective optimization is not usually a single solution, but a set of them. The next section explains how we approach the problem of constructing a final classification model from the resulting trade-off solutions.

D. Pareto-based Ensembles

The outcome of a MOEA is a set of non-dominated solutions that approximates the Pareto optimal set. All these solutions are equally acceptable for the problem at hand when there is no preference information available. However, in the problem that we face, the goal is to construct an SVM model with a reduced training set, which is used in the classification of unknown patterns. Thus, it is desirable to perform a post-processing step over the trade-off solutions in order to get a final classification model. In this section, we describe the strategies that we have studied to this end.

The solutions in the resulting non-dominated set would share some information about the training data. Moreover, each solution in the non-dominated set corresponds to an SVM-classifier trained with different hyper-parameter sets and different subsets of the original training set. An ensemble of classifiers allows combining the individual information given by each model and providing more information about the class label than a single classifier. Due to the nature of multi-objective optimization, ensemble learning can be approached in a straightforward fashion since it yields to a set of optimal and diverse models [57]. In this regard, we study five different ways of combining the information which are described in the following.

- ***Global Pareto Ensemble*** **(GPE)**: The basic idea here is to build an ensemble using all solutions in the resulting non-dominated set.
- ***Incremental Error Reduction Ensemble*** **(IERE)**: The idea of this approach is not to use all solutions in the non-dominated set, but a subset of these. The ensemble is constructed in an incremental fashion. First, the solution with the lowest error on the training dataset is included in the ensemble. The second solution included is the one that minimizes the error rate (on the training dataset) of the partial ensemble, and so on. Let $\mathcal{NC}$ be the set of classifiers in the resulting non-dominated set, the ensemble $\mathcal{E}$ is formed as follows:

$$\mathcal{E}_u = \underset{k}{\operatorname{argmin}} \frac{1}{|\mathcal{T}|} \sum_{i=1}^{|\mathcal{T}|} \mathcal{L}\left(\mathcal{E}_{u-1} \cup \mathcal{NC}_k\left(\mathbf{x}_i\right), y_i\right) \tag{6}$$

where the index k runs over all classifiers that are not already included in the ensemble and $\mathcal{E}_{u-1} \cup \mathcal{NC}_k\left(\mathbf{x}_i\right)$ is the predicted class when $\mathcal{NC}_k$ is inserted into the ensemble.
- ***Complementary Incremental Ensemble*** **(CIE)**: This is also an incremental approach for ensemble construction. The idea is to include, at each iteration, the classifier whose performance is the most complementary to that of the partial ensemble. As in error reduction, the first classifier included is the one that has the lowest error rate on the training dataset. Subsequent classifiers are incorporated by adding the one that has the lowest error rate on the samples that were missclassified by the partial ensemble, i.e.,

$$\begin{aligned}\mathcal{E}_u = &\underset{k}{\operatorname{argmin}} \frac{1}{|\mathcal{T}|} \sum_{i=1}^{|\mathcal{T}|} \mathcal{L}\left(\mathcal{E}_{u-1} \cup \mathcal{NC}_k\left(\mathbf{x}_i\right), y_i\right) \\ &\wedge \mathcal{L}\left(\mathcal{E}_{u-1}\left(\mathbf{x}_i\right), y_i\right) = 1\end{aligned} \tag{7}$$

Similar to error reduction, the index k runs over all classifiers that are not already included in the ensemble.
- ***Margin Distance based Ensemble*** **(MDE)**: Margin distance minimization is a method introduced in [58] for pruning bagging ensembles; here, we study it in the context of Pareto-based ensembles. In this approach, a signature vector $\mathbf{c}^{(t)}$ of a classifier k is defined as:

$$\mathbf{c}_i^{(k)} = 1 - 2 \times \mathcal{L}\left(\mathcal{NC}_k\left(\mathbf{x}_i\right), y_i\right), \forall i \in \{1, \ldots, |\mathcal{T}|\} \tag{8}$$

where $\mathbf{c}_i^{(k)}$ is equal to 1 if the k^{th} classifier correctly predicts the i^{th} instance or -1 otherwise. The average signature vector is defined as:

$$\overline{\mathbf{c}_i} = \frac{1}{|\mathcal{NC}|} \sum_{k=1}^{|\mathcal{NC}|} \mathbf{c}_i^{(k)}, \forall i \in \{1, \ldots, |\mathcal{T}|\} \tag{9}$$

An instance i is correctly classified by the ensemble if $\overline{\mathbf{c}_i} > 0$. The goal of this approach is to minimize the distance of the average signature vector to a positive reference point. Let $\mathbf{o}$ be the reference point, the classifier selected in the u iteration is determined by

$$\mathcal{E}_u = \underset{k}{\operatorname{argmin}} \left\| \mathbf{o} - \frac{1}{|\mathcal{NC}|} \left(\mathbf{c}^{(k)} + \sum_{t=1}^{u-1} \mathbf{c}^{(t)} \right) \right\| \tag{10}$$

where $\|d\|$ is the euclidean norm and $0 < o_i < 1, \forall i \in \{1, \ldots, |\mathcal{T}|\}$.
- ***Boosting***: This approach, proposed in [59], consists of selecting, at each iteration, the classifier that minimizes the weighted error rate. The weighting scheme used in the error computation follows the one given by Adaboost [60]. Here, instead of training a classifier with the weighted training set, the classifier with the lowest error rate on such a dataset is selected from

the pool of classifiers available in the resulting non-dominated set.

These schemes describe different ways of choosing the SVM classification model from the non-dominated set. The next step is to combine the information given by each model into a single final prediction. We face this issue in a straightforward fashion by taking a majority vote given by the individual prediction of the models when combining them into an ensemble model.

Finally, note that since the SVM models in the non-dominated front are learned during the optimization step, none of them requires any further training. Thus, the Pareto-based ensemble construction does not add a significantly extra computational cost to the one required for the evolutionary search.

IV. Experiments and Results

In this section, we describe the experiments performed as well as the results obtained by our proposal using different classification datasets. We also present statistical tests comparing our results with some classical instance selection methods.

A. Experimental Settings

For our experiments, we used a set of 43 datasets available in the KEEL repository [61], which were also used in the comparative study of IS techniques performed in [15]. Table I shows some characteristics of these datasets, such as the number of instances, the number of features, and the number of classes. These datasets[1] were previously partitioned into 10 training/test subsets by means of the k-fold cross validation technique. In 10-fold cross validation, a dataset is divided into 10 disjoint subsets, which are used for training and testing. In the i^{th} iteration, the i^{th} subset is used as a test set, while the rest are used as a training set. Therefore, this procedure is repeated 10 times and ensures that each subset has been used for testing the model. We apply our proposal to each fold independently.

We use two criteria for assessing the performance of our algorithm, the first one is related to the classification performance on the test set, which is computed as the average accuracy per class. The second one is the reduction rate attained in the instance selection[2]. Table II shows the parameter configuration used in our experiments. For the SVM, we used the implementation of LibSVM [62]. Experiments were carried out by a computer having the following features:

- Processor: Intel Core i7
- Clock speed: 2.8 GHz
- RAM: 24 GB
- Hard drive: 1 TB
- Operating system: Fedora release 20

[1]The datasets and the partitions are available at http://sci2s.ugr.es/keel/datasets.php

[2]The reduction rate is computed as one minus the ratio between the number of selected instances and the training set size. For instance, a reduction rate equal to 80% indicates that only 20% of the training samples are used during the learning of the classification model.

TABLE I: Description of the datasets used in our study. For each dataset, we show the number of instances, the number of attributes, and the number of classes.

ID	Dataset	Atts.	Insts.	Classes
1	Appendicitis	7	106	2
2	Australian	14	690	2
3	Automobile	25	150	6
4	Balance	4	625	3
5	Bands	19	365	2
6	Breast	9	277	2
7	Bupa	6	345	2
8	Car	6	1,728	4
9	Cleveland	13	297	5
10	Contraceptive	9	1,473	3
11	CRX	15	653	2
12	Dermatology	34	358	6
13	Ecoli	7	336	8
14	Flare solar	11	1,066	6
15	German	20	1,000	2
16	Glass	9	214	7
17	Haberman	3	306	2
18	Hayes-roth	4	160	3
19	Heart	13	270	2
20	Hepatitis	19	80	2
21	Housevotes	16	232	2
22	Ionosphere	33	351	2
23	Iris	4	150	3
24	Led7digit	7	500	10
25	Lymphography	18	148	4
26	Mammographic	5	830	2
27	Monk-2	6	432	2
28	Movement Libras	90	360	15
29	Newthyroid	5	215	3
30	Pima	8	768	2
31	Post-operative	8	87	3
32	Saheart	9	462	2
33	Sonar	60	208	2
34	Spectfheart	44	267	2
35	Tae	5	151	3
36	Tic-tac-toe	9	958	2
37	Vehicle	18	846	4
38	Vowel	13	990	11
39	Wdbc	30	569	2
40	Wine	13	178	3
41	Wisconsin	9	683	2
42	Yeast	8	1,484	10
43	Zoo	16	101	7

B. Experimental Results

This section presents the experimental evaluation of the proposed approach. Several experiments are carried out. The first one compares the performance of the Pareto-based Ensembles strategies. The second set of experiments assesses the suitability of performing either IS or model selection. The third one aims at evaluating the performance of the multi-objective formulation against a single objective formulation. The fourth experiment compares the performance over traditional IS selection techniques and, finally, the fifth experiment compares with standard learning algorithms.

*1) **Comparing Among the Pareto-based Ensemble Strategies:*** This section aims at comparing among the different Pareto-based ensemble strategies in order to determine if there is one of them that performs better. To this end, we present the results obtained by the different variants of the proposed EMOMIS-PbE: Evolutionary Multi-Objective Model and Instance Selection

TABLE II: Parameters configuration adopted for our experiments.

Method	Parameters
EMOMIS-PbE	Population size: 100
	Generations: 100
	Crossover rate: 0.9
	Mutation rate: 0.1
	Stopping criterion: A max number of generations or when the improvement in the non-dominated front is not greater or equal than 0.001 in the hypervolume value
SVM (LibSVM)	Kernel: RBF
	γ: 0.1
	C: 1
Kernel Ridge Reg. (KRR)	Kernel: RBF
	γ: 0.1
Random Forest (RF)	number of trees: 50
RF-Oblique	number of tress: 50
	No. features to sample: $\sqrt{no.\ feat.}$
	Regularization: Tikhonov
DROP3	k: 3
ENN	k: 3
FCNN	k: 3
RNGE	k: 3
HMN-EI	ϵ: 0.1

with Pareto-based Ensembles approach[3]. One should recall that the IS can be performed in two ways: either using a filter approach (EMOMFIS) or a wrapper approach (EMOMWIS). Moreover, the Pareto-based ensemble can be performed in five different manners: Global Pareto Ensemble (GPE), Incremental Error Reduction Ensemble (IERE), Complementary Incremental Ensemble (CIE), Margin Distance based Ensemble (MDE), and boosting. For instance, EMOMFIS-MDE represents our proposal, performing the IS with a filter approach and the decision in the classification of a test pattern is done following the margin distance ensemble.

Table III shows the obtained results by each of the Pareto-based ensembles in both EMOMFIS and EMOMWIS. The reported results are the average accuracy in the test set and the reduction rate in training set. These results are the mean and standard deviation over the 43 classification problems. For each case, the best result is shown in **boldface**.

Apart from the average results reported above, we carried out a statistical analysis on the results gathered with the variants of EMOMIS-PbE in the test set. The Wilcoxon signed rank test is used in order to conduct the pairwise comparisons among all variants. The considered level of significance in the statistical test is set to $\alpha = 0.05$. Table IV summarizes the results obtained by the statistical tests. For each method in the rows, the column represented by the symbol "+" indicates the number of variants of EMOMIS that were outperformed according to the Wilcoxon test. The column with the "±" symbol indicates the number of wins and ties obtained by the method in the row. The maximum value for each case is highlighted in **boldface**.

[3]We provide adequate information for the readers to reproduce the results at http://ccc.inaoep.mx/~arosales/resources/sources/emomis_pbe.zip

TABLE III: Average accuracy and reduction rate obtained by the different Pareto-based Ensembles strategies. We report the mean and the standard deviations over the 43 datasets. For each case, the best result is shown in **boldface**.

Method	Acc. Test	Reduction
EMOMFIS-GPE	73.10 ± 17.10	75.21 ± 05.24
EMOMFIS-IERE	75.79 ± 18.69	30.98 ± 21.44
EMOMFIS-CIE	73.17 ± 18.04	59.82 ± 18.06
EMOMFIS-MDE	64.22 ± 17.00	$\mathbf{93.49 \pm 80.36}$
EMOMFIS-Boosting	$\mathbf{76.34 \pm 18.35}$	78.37 ± 11.84
EMOMWIS-GPE	77.24 ± 17.75	63.15 ± 03.31
EMOMWIS-IERE	76.46 ± 18.11	58.17 ± 05.22
EMOMWIS-CIE	76.11 ± 17.60	66.14 ± 07.36
EMOMWIS-MDE	71.56 ± 17.38	$\mathbf{73.27 \pm 07.19}$
EMOMWIS-Boosting	$\mathbf{77.36 \pm 17.69}$	64.43 ± 05.64

TABLE IV: Wilcoxon test results for accuracy on the test set and reduction rate in the training set.

Method	Acc. Test	
	+	±
EMOMFIS-GPE	1	3
EMOMFIS-IERE	2	**4**
EMOMFIS-CIE	1	2
EMOMFIS-MDE	0	0
EMOMFIS-Boosting	**3**	**4**
EMOMWIS-GPE	**3**	**4**
EMOMWIS-IERE	1	2
EMOMWIS-CIE	1	2
EMOMWIS-MDE	0	0
EMOMWIS-Boosting	**3**	**4**

Another aspect to be taken into consideration is the running time required by EMOMFIS and EMOMWIS. Table V shows the times averaged over the 43 datasets. The information reported is the following:

- Training time, which indicates the total time required by EMOMFIS and EMOMWIS for performing the IS, hyper-parameters optimization, learning the SVM, and the ensemble construction in a single training partition;
- Fitness time, which indicates the total time spent in the fitness evaluations during the training stage;
- Testing time, which indicates the time required by EMOMFIS and EMOMWIS for predicting the test set; and
- The evolutionary operators rate (E.O. Rate), which indicates the ratio of time that is spent by the evolutionary operators with respect to the total time spent during the evolutionary search stage.

From Table V, note that for the different variants both EMOMFIS and EMOMWIS, the fitness time and E.O. rate are not affected by the ensemble construction. This is due to the fact that the Pareto-based Ensemble is performed as a post-processing step over the generated Pareto optimal set. Therefore, the ensembles do not significantly increase the computational cost of the proposal.

From the results and statistical tests presented in Tables III, IV and V, we can highlight the following:

TABLE V: Required time in seconds for both EMOFIS and EMOMWIS.

Method	Running time (s)			E.O. Rate
	Training	Fitness	Testing	
EMOMFIS-GPE	444.00	375.00	0.030	0.36
EMOMFIS-IERE	443.58	375.00	0.003	0.36
EMOMFIS-CIE	443.94	375.00	0.004	0.36
EMOMFIS-MDE	444.54	375.00	0.003	0.36
EMOMFIS-Boosting	443.93	375.00	0.038	0.36
EMOMWIS-GPE	509.58	472.12	0.022	0.41
EMOMWIS-IERE	509.17	472.12	0.004	0.41
EMOMWIS-CIE	509.53	472.12	0.004	0.41
EMOMWIS-MDE	510.14	472.12	0.001	0.41
EMOMWIS-Boosting	509.52	472.12	0.051	0.41

- The performance in the test set is very similar both in Filter and Wrapper IS, except for the MDE Pareto-based ensemble approach.
- Both in EMOMWIS and EMOMFIS, boosting and IERE are good alternatives for the Pareto-based ensemble.
- The reduction rate in the training set is usually higher following a filter IS approach than a following a wrapper IS approach.
- Boosting is the outstanding Pareto-based ensemble both for wrapper IS and filter IS.

Finally, in order to evaluate the scalability of EMOMWIS and EMOMFIS, we tested them on an artificial dataset (banana dataset) by sampling subsets with different size and measuring the running time required according to the number of samples. Figure 3 shows the behavior over different numbers of instances. From this, we can note that the time required by our proposal grows almost quadratically with respect to the number of instances.

Figure 4 shows examples of the generated Pareto fronts for each type of encoding. From this, we can note that, indeed, a trade-off exists between the two criteria, such that when increasing the number of instances for training the SVM, its classification performance can be improved, while reducing the size of the training set could lead to a trained model with a lower classification performance. Over these solutions, we apply the Pareto processing strategies that are studied in this paper for constructing a final reliable classification model, taking into account these two conflicting criteria.

Figure 5 shows the normalized frequency in which each of the filter IS techniques were chosen by EMOMFIS-Boosting. From this figure, we can observe that, on average, DROP3 and HMN-EI were the most preferred methods, while ENN and RNGE were the least preferred ones. This should not be so surprising, since ENN and RNGE are edition based IS methods that seek to remove border points, which are the ones that have a higher probability of becoming support vectors. For a given dataset, different IS techniques were employed with different SVM hyper-parameters, resulting in classifiers that were trained with different subsets of the dataset and with a different configuration. This leads to a diversity in the models,

TABLE VI: Reported results for wrapper/filter instance selection using the Pareto-based ensemble.

Method	Acc. Test	Reduction	Running Time (s)	
			Training	Testing
EMO-FIS-GPE	69.23 ± 18.11	60.45 ± 10.87	200.25	0.003
EMO-FIS-IERE	71.90 ± 17.10	50.42 ± 23.42	201.32	0.007
EMO-FIS-CIE	66.41 ± 17.97	74.29 ± 13.77	199.99	0.007
EMO-FIS-MDE	58.03 ± 18.28	93.43 ± 04.03	200.85	0.003
EMO-FIS-Boosting	70.16 ± 19.61	57.51 ± 15.87	200.32	0.090
EMO-WIS-GPE	65.65 ± 20.39	49.38 ± 02.24	189.93	0.004
EMO-WIS-IERE	66.07 ± 20.60	49.54 ± 03.45	190.21	0.004
EMO-WIS-CIE	65.64 ± 20.39	49.38 ± 02.24	190.34	0.006
EMO-WIS-MDE	65.71 ± 19.17	49.84 ± 02.22	191.24	0.004
EMO-WIS-Boosting	61.89 ± 25.89	49.98 ± 02.44	190.92	0.028
EMOMIS-NIS	70.79 ± 18.58	—	689.11	0.004
EMOMFIS-Boosting	76.34 ± 18.35	78.37 ± 11.84	509.52	0.051
EMOMWIS-Boosting	77.36 ± 17.69	64.43 ± 05.64	443.93	0.038

which can be exploited for an ensemble construction, as has been done in the present study.

In summary, Boosting can be highlighted as the best Pareto-based ensemble for wrapper IS and filter IS. It is able to achieve the highest performance in classifying the test set, while reducing the training set size in a significant manner. Therefore, Boosting Pareto-based Ensemble is used hereafter in the comparison with other approaches.

*2) **Comparing with a Non-Instance Selection and a Non-Model Selection Approaches:*** EMOMIS-PbE combines both IS and MS into a single approach. Thus, the goal of this section is twofold. On the one hand, we want to evaluate the advantages of performing IS against using the entire training set. To this end, our proposal has been modified so that it only performs the MS. This results in a variant that we call EMOMIS-NIS.

On the other hand, we also want to evaluate the suitability of performing MS against not doing so. For this, only IS is performed, while MS is omitted. This yields to two variants of EMOMIS-Pbe, which are called EMO-FIS and EMO-WIS. For each, the five Pareto-based Ensemble strategies can be applied.

Considering the aforementioned experimental study, Table VI shows the results obtained for EMOMIS-NIS, as well as for EMO-WIS and EMO-FIS for each of the Pareto-based ensemble strategies.

From the results in Table VI, we can note that the best results for both EMO-FIS and EMO-WIS are reached when IERE Pareto-based ensemble is used. Thus, these two versions are used for comparison with EMOMFIS-Boosting and EMOMWIS-Boosting.

In [63], [64], the authors recommend the use of nonparametric tests for a safe and robust statistical comparison. Here, The Wilcoxon signed rank test [63], [64] is used to statistically assess their performance. Table VII shows the results.

Based on the comparative study and on the statistical results, we summarize them as follows:

- An improvement of above 10% in accuracy in the test set is reached when model selection is performed for wrappers and around 6% for filter IS.
- EMOMFIS-Boosting and EMOMWIS-Boosting performs better when both IS and MS are carried out.

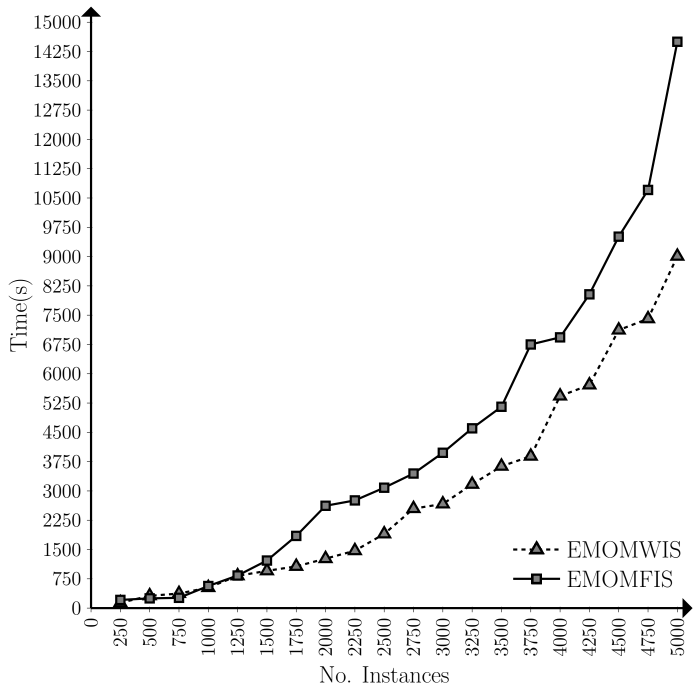


Fig. 3: Running time for EMOMWIS and EMOMFIS for different number of instances.

TABLE VII: Results obtained by the Wilcoxon signed rank test.

Methods	R^+	R^-	p-value
EMOMFIS-Boosting vs			
EMO-WIS-IERE	329	532	0.191
EMOMIS-NIS	62	884	< 0.05
EMOMWIS-Boosting vs			
EMO-FIS-IERE	88	858	< 0.05
EMOMIS-NIS	61	885	< 0.05

These can be seen when they are compared with EMOMIS-NIS and EMO-FIS-IERE and EMO-WIS-IERE, respectively.

- The reduction rate in the training set is better both in EMOMFIS-Boosting and EMOMWIS-Boosting.
- The Wilcoxon test revealed, with $p < 0.05$, that there is a statistically significant difference between EMOMWIS-Boosting and EMO-WIS-IERE. This test did not find a statistically significant difference between EMOMFIS-Boosting and EMO-FIS-IERE.
- EMOMIS-NIS is the worst with respect to training time. This is because in the fitness evaluation, the entire training set is used.
- The training time in EMOMFIS-Boosting is slightly higher than the other filter approaches and also in EMOMWIS-Boosting with respect to the wrapper methods. This is because EMO-FIS and EMO-WIS do not perform MS, which significantly reduces the search space and avoids exploring time-consuming model configurations.
- The time required for classifying instances (testing time) among the approaches is virtually the same.

Summarizing, the appropriate selection of the SVM's hyper-parameters together with the IS performed by EMOMWIS-Boosting and EOMFIS-Boosting allows the construction of more reliable classification models than when either only focusing on the selection of the instances or the model hyper-parameters. This shows the advantages of performing model selection as compared to not doing so.

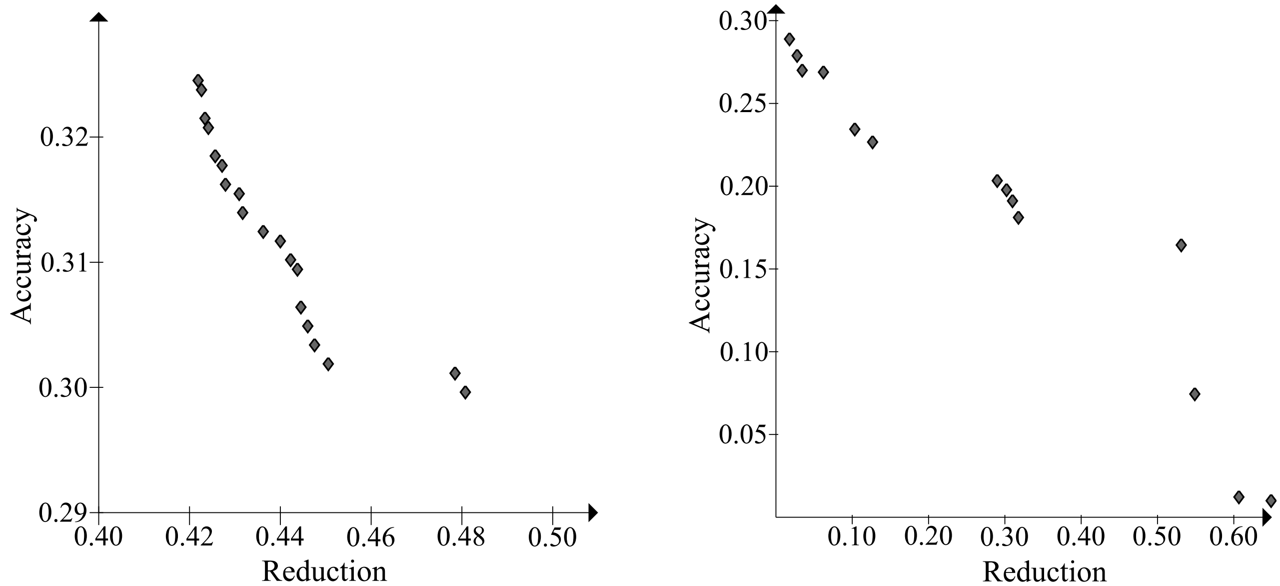
*3) **Comparing with a Single Objective Approach:*** The goal of this section is to determine the suitability of Pareto processing instead of choosing a single solution. To this aim, a single solution from the resulting non-dominated front is chosen. To allow a fair comparison, both approaches use the same computational budget, i.e., the same population size, number of generations, and they are run in the same hardware configuration. The solution chosen for the wrapper approach is called EMOMWIS-single and the one for the filter approach is called EMOMFIS-single. The chosen solution is the one that minimizes the following aggregation function:

$$f_{ag} = \frac{1}{2\,|\mathcal{T}|}\left(\sum_{i=1}^{|\mathcal{T}|} \mathcal{L}\left(y_i, y_i^*\right) + |\mathcal{S}|\right) \tag{11}$$

where $\mathcal{L}\left(y_i, y_i^*\right)$ and $|\mathcal{S}|$ are the optimization criteria used in the multi-objective formulation (see Section III-C).

Table VIII shows the results for both EMOMFIS-Boosting and EMOMWIS-Boosting when they are compared with the single objective versions.

Wilcoxon Signed Rank Test is used with the aim of statistically assessing the performance of the multi-objective version against the single objective one. The results of this test are shown in Table IX.



(a) EMOMWIS (b) EMOMFIS

Fig. 4: Non-dominated fronts generated for each representation in the Multi-Objective Model and Instance Selection approach.

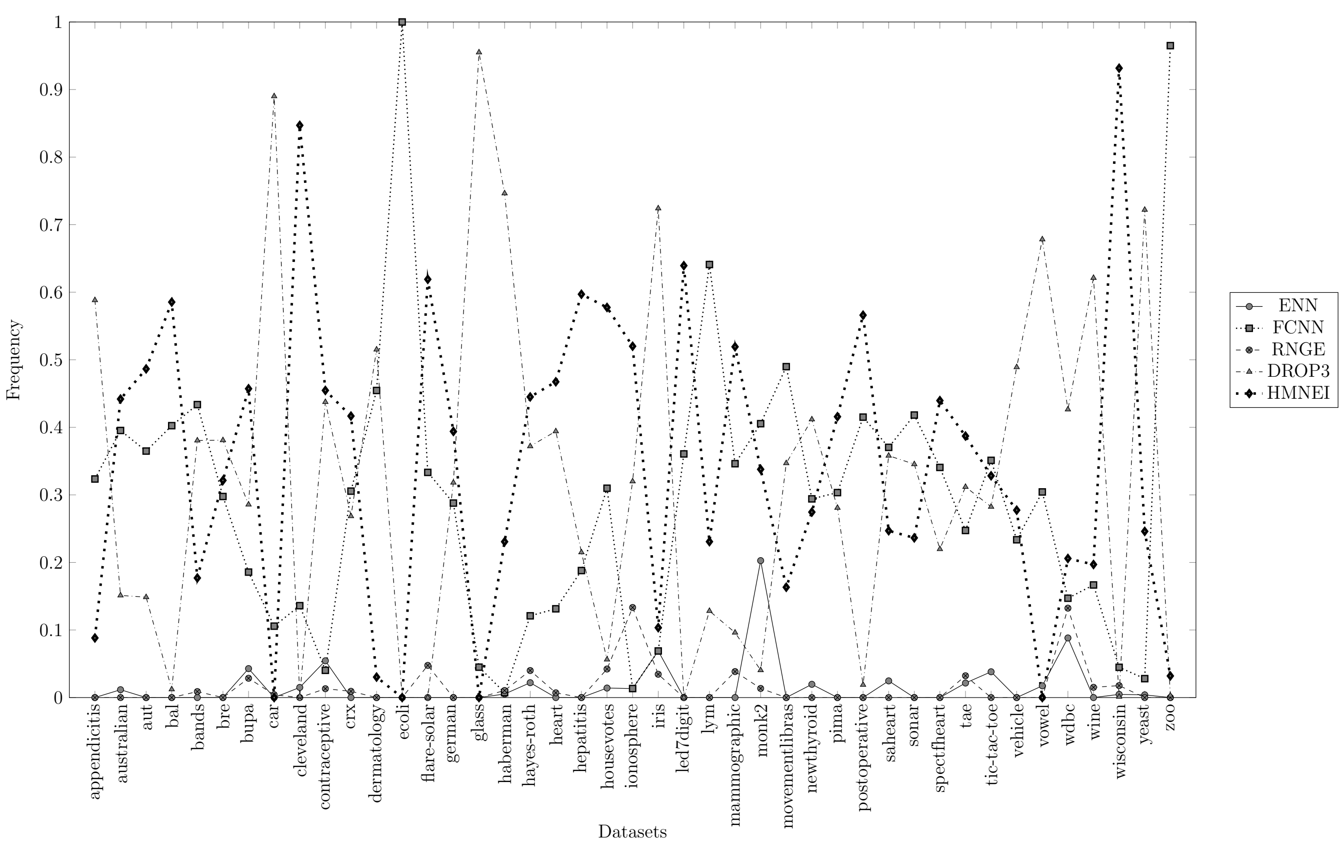


Fig. 5: Normalized frequency of the selection of the different filter instance selection techniques in EMOMFIS-Boosting.

From the results in Table VIII and Table IX, the following can be remarked:

- There is an improvement of around 2% on the classification performance on the test set when the Pareto-based ensemble is used.
- The difference in terms of the reduction rate on the training set is approximately 2%.
- The statistical test shows, with $p < 0.05$, that the difference between EMOMWIS-Boosting and EMOMWIS-single is statistically significant.

TABLE VIII: Reported results for the single and the multi-objective versions of EMOMIS-PbE.

Método	Acc. Test	Reduction	Runtime (s)	
			Training	Testing
EMOMFIS-Boosting	76.34 ± 18.35	78.37 ± 11.84	443.93	0.038
EMOMFIS-Single	74.54 ± 18.11	81.39 ± 12.59	439.25	0.003
EMOMWIS-Boosting	77.36 ± 17.69	64.43 ± 05.64	509.52	0.051
EMOMWIS-Single	75.95 ± 17.63	66.85 ± 04.65	504.84	0.004

TABLE IX: Obtained results for the Wilcoxon test.

Método	R^+	R^-	p-value
EMOMFIS-Boosting vs EMOMFIS-Single	176	770	< 0.05
EMOMWIS-Boosting vs EMOMWIS-Single	182	764	< 0.05

TABLE X: Comparison of the performance of EMOMFIS-Boosting and EMOMWIS-Boosting against traditional IS techniques.

Method	Acc. Test	Reduction
SVM	60.15 ± 22.30	–
DROP3	65.17 ± 18.71	62.62 ± 09.77
FCNN	66.83 ± 18.61	62.60 ± 17.08
HMN-EI	68.25 ± 17.94	56.43 ± 21.96
RNGE	68.95 ± 17.68	36.30 ± 22.18
ENN	69.83 ± 17.68	26.06 ± 13.90
EMOMFIS-Boosting	76.34 ± 18.35	78.37 ± 11.84
EMOMWIS-Boosting	77.36 ± 17.69	64.43 ± 05.64

- The statistical test also shows, with $p < 0.05$, that the difference between EMOMFIS-Boosting and EMOMFIS-single is statistically significant.
- Both training time and testing time between the single criterion approaches and the MO approach are almost the same.

Summarizing, the multi-objective approach enables the ensemble of models, which were shown to be beneficial with respect to the classification performance on the test set when compared with the single objective formulation. Moreover, the multi-objective formulation also allows introducing two sources of diversity; one of them is because of the diversity in the hyper-parameters for the SVM and the other is because of the use of different IS selection techniques, which means different subsets of training samples to be used in the training step. Therefore, EMOMWIS-Boosting and EMOMFIS-Boosting are two competent methods for addressing the IS and model selection problem for an SVM based on a MO approach.

4) ***Comparing with Traditional Instance Selection Techniques:*** In this section, we compare the performance of EMOMWIS-Boosting and EMOMFIS-Boosting with respect to a set of well-known filter IS techniques, which are described in [15].

Table X shows the results. The SVM is used as a baseline to compare the performance of the different methods.

For the statistical comparison, we used the Friedman Aligned test and Holm's procedure as the post-hoc test. Two types of analysis are performed: on the one hand, we contrast the performance of EMOMFIS-Boosting with the traditional IS techniques and, on the other hand, we contrast EMOMWIS-Boosting with the traditional IS techniques. The results of these tests are shown in Table XI.

TABLE XI: Results of the Friedman Aligned statistical test. We show the average ranking and the adjusted p values (APV) obtained with Holm's procedure.

Method	Rank	APV
SVM	195.70	< 0.05
DROP3	187.79	< 0.05
FCNN	159.02	< 0.05
RNGE	152.35	< 0.05
HMN-EI	148.55	< 0.05
ENN	135.62	< 0.05
EMOMFIS-Boosting	77.98	—
SVM	195.93	< 0.05
DROP3	188.95	< 0.05
FCNN	160.63	< 0.05
RNGE	153.12	< 0.05
HMN-EI	150.87	< 0.05
ENN	137.03	< 0.05
EMOMFIS-Boosting	70.47	—

From the results shown in these tables, we can highlight the following:

- Both EMOMWIS-Boosting and EMOMFIS-Boosting are able to improve the performance in classification when they are compared with traditional approaches.
- Traditional IS techniques outperform the standard SVM.
- Among the traditional IS techniques, ENN is the one with the highest classification rate in the testing set, but has the lowest reduction rate in the training set.
- The statistical test revealed that both EMOMWIS-Boosting and EMOMFIS-Boosting are able to statistically outperform the reference methods.

In summary, both EMOMWIS-Boosting and EMOMFIS-Boosting show to significantly improve traditional IS techniques in both criteria: the average classification accuracy in the test set and the reduction attained in the training set. Therefore, they are competitive methods for performing data reduction and hyper-parameter selection for an SVM and can be applied to a wide range of supervised learning problems.

5) ***Comparing with Standard Learning Algorithms:*** This section aims at comparing the performance of EMOMIS-PbE with respect to standard learning algorithms. As the best performance of EMOMIS-PbE is reached with EMOMWIS-Boosting, this is used. The learning algorithms adopted for the comparison are Kernel Ridge Regression (KRR) [65], Random Forest (RF) [66] and RF-Oblique [67]. Table XII shows the results obtained by each method.

The Friedman Aligned test and Holm procedure are used for statistically comparing EMOMWIS-Boosting against the standard learning algorithms. Since the goal is to compare the performance of EMOMWIS-Boosting against the reference methods, this is used as a control method. The rankings reported by the Friedman Aligned test and the adjusted p-values (APV) are shown in Ta-

TABLE XII: Comparison of the performance of EMOMWIS-Boosting against standard learning algorithms.

Method	Acc. Test	Running Time (s)	
		Training	Testing
KRR	68.78 ± 19.90	4.23	0.002
Random Forest	70.95 ± 19.53	5.20	0.172
RF-Oblique-Tikhonov	74.06 ± 17.42	10.44	0.168
EMOMWIS-Boosting	77.36 ± 17.69	509.53	0.051

TABLE XIII: Results from Friedman Aligned test and Holm procedure when comparing with standard learning algorithms.

Método	Rank	APV
KRR	113.23	< 0.01
Random Forest	99.60	< 0.01
RF-Oblique-Tikhonov	77.24	0.047
EMOMWIS-Boosting	55.92	—

ble XIII.

Based on the above, we can summarize the following:

- EMOMWIS-Boosting is able to statistically outperform KRR, RF, and RF-Oblique-Tikhonov at the considered significance level of $\alpha = 0.05$.
- EMOMWIS-Boosting is the most computationally expensive algorithm in the training stage. On the other hand, KRR is the most efficient in the training stage.
- RF and RF-Oblique are the slowest in the prediction of unknown patterns.

EMOMWIS-Boosting showed to be a highly effective classification algorithm when is compared to standard learning algorithms. RF-Oblique showed to be the second most effective methods. Both RF and RF-Oblique required more time to classify a single test pattern. This is because the test instance needs to be classified by a large number of trees. EMOMWIS-Boosting, on the other hand, is also able to construct an ensemble of classification models with high performance. Although EMOMWIS-Boosting requires a higher computational time in the training stage than the reference methods, this step can be done offline. Moreover, the Pareto-based ensemble allows pruning the Pareto solutions to become an ensemble member, producing ensembles with a lower number of SVM models. As a result, EMOMWIS-Boosting does not significantly increase the required time to predict an instance.

V. Conclusions

In this paper, we have presented EMOMIS-PbE, an Evolutionary Multi-Objective Model and Instance Selection approach with Pareto-based Ensembles. EMOMIS-PbE deals simultaneously with the problem of selecting instances and the hyper-parameters for training an SVM. It takes into account, as the criteria to be optimized, the reduction attained on the training set and the performance when such a reduction is used with a given set of an SVM's hyper-parameters. EMOMIS-PbE explores two types of encoding in order to perform the IS either using a wrapper or filter approach. Moreover, it also introduces five different strategies for processing the Pareto solutions by combining them into an ensemble. These are the global Pareto ensemble (GPE), incremental error reduction ensemble (IERE), complementary incremental ensemble (CIE), margin distance ensemble (MDE), and boosting. EMOMIS-PbE showed the following advantages:

- the reduction of the training set size can help to improve the performance of the SVM;
- the selection of the hyper-parameters allows for the construction of a tuned model for the specific problem at hand;
- the multi-objective formulation generates multiple Pareto solutions, which enables the user to work with different preferences for the objectives without having to perform a new search for each preference; and
- the multiple trade-off solutions also allow ensembles to be constructed in a straightforward fashion without adding any significant computational cost.

A comprehensive experimental evaluation indicates that the performance in terms of classification accuracy is similar for both, the wrapper and filter approaches. However, filter approaches have been shown to achieve higher reduction rates in the training set. The experimental evaluation also revealed that among the Pareto-based ensemble strategies, boosting is the one with the highest performance in both classification accuracy and reduction rate. Taking into account both the results of the comparative study and the well-known IS techniques, we can conclude that EMOMIS-PbE allows us to obtain solutions with a good trade-off between the size of the training set and the performance of the SVM.

Although the datasets used in our study cannot be considered as large-scale, they are a common benchmark for assessing the performance of learning algorithms. They have also been used in several IS studies, such as those performed in [15], [19], [20], [22]. Moreover, they give insights about the behavior of our proposal. Nevertheless, evaluating our proposal on large-scale datasets and working on the design of methods based on evolutionary multi-objective optimization for instance and model selection using big data technologies are interesting paths for future research. Comparing with RF with a higher number of trees (approximately 500 trees, which makes it more robust [68]), extending the ideas to tune the hyper-parameters for both KRR and RF, and incorporating ideas from surrogate-assisted optimization or meta-learning to reduce the search space are also research lines for future work.

References

[1] V. N. Vapnik, *The nature of statistical learning theory.* New York, NY, USA: Springer-Verlag New York, Inc., 1995.

[2] O. Chapelle, "Training a support vector machine in the primal," *Neural Comput.*, vol. 19, no. 5, pp. 1155–1178, 2007.

[3] R. Collobert and S. Bengio, "SVMTorch: Support Vector Machines for large-scale regression problems," *J. Mach. Learn. Res.*, vol. 1, pp. 143–160, 2001.

[4] J. C. Platt, "Fast training of support vector machines using sequential minimal optimization," in *Advances in Kernel Methods*, B. Schölkopf, C. J. C. Burges, and A. J. Smola, Eds. Cambridge, MA, USA: MIT Press, 1999, pp. 185–208.

[5] H. J. Escalante, M. Montes, and L. E. Sucar, "Particle swarm model selection," *J. Mach. Learn. Res.*, vol. 10, pp. 405–440, 2009.

[6] A. Rosales-Pérez, J. A. Gonzalez, C. A. Coello Coello, H. J. Escalante, and C. A. Reyes-Garcia, "Surrogate-assisted multi-objective model selection for support vector machines," *Neurocomputing*, vol. 150, Part A, pp. 163 – 172, 2015.

[7] Q. Sun, B. Pfahringer, and M. Mayo, "Full model selection in the space of data mining operators," in *Proceedings of the fourteenth international conference on Genetic and evolutionary computation conference companion*, ser. GECCO Companion '12. New York, NY, USA: ACM, 2012, pp. 1503–1504.

[8] S. García, J. Luengo, and F. Herrera, *Data Preprocessing in Data Mining*. Springer Publishing, 2014.

[9] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[10] S. García, J. Luengo, and F. Herrera, "Tutorial on practical tips of the most influential data preprocessing algorithms in data mining," *Knowl.-Based Syst.*, vol. 98, pp. 1–29, 2016.

[11] A. Arnaiz-González, J.-F. Díez-Pastor, J. J. Rodríguez, and C. García-Osorio, "Instance selection of linear complexity for big data," *Knowl.-Based Syst.*, vol. 107, pp. 83–95, 2016.

[12] I. Triguero, S. del Río, V. López, J. Bacardit, J. M. Benítez, and F. Herrera, "ROSEFW-RF: The winner algorithm for the ECBDL'14 big data competition: An extremely imbalanced big data bioinformatics problem," *Knowl.-Based Syst.*, vol. 87, pp. 69 – 79, 2015, computational Intelligence Applications for Data Science.

[13] J. Cano, F. Herrera, and M. Lozano, "Using evolutionary algorithms as instance selection for data reduction in KDD: an experimental study," *IEEE T. Evol. Comput.*, vol. 7, no. 6, pp. 561–575, 2003.

[14] S. García, J. R. Cano, and F. Herrera, "A memetic algorithm for evolutionary prototype selection: A scaling up approach," *Pattern Recogn.*, vol. 41, no. 8, pp. 2693 – 2709, 2008.

[15] S. García, J. Derrac, J. Cano, and F. Herrera, "Prototype selection for nearest neighbor classification: Taxonomy and empirical study," *IEEE T. Pattern Anal.*, vol. 34, no. 3, pp. 417–435, 2012.

[16] E. Marchiori, "Hit miss networks with applications to instance selection," *J. Mach. Learn. Res.*, vol. 9, pp. 997–1017, 2008.

[17] J. Sánchez, F. Pla, and F. Ferri, "Prototype selection for the nearest neighbour rule through proximity graphs," *Pattern Recogn. Lett.*, vol. 18, no. 6, pp. 507 – 513, 1997.

[18] F. Dornaika and I. K. Aldine, "Decremental sparse modeling representative selection for prototype selection," *Pattern Recogn.*, vol. 48, no. 11, pp. 3714 – 3727, 2015.

[19] N. Garcia-Pedrajas, "Constructing ensembles of classifiers by means of weighted instance selection," *IEEE T. Neural Networ.*, vol. 20, no. 2, pp. 258–277, 2009.

[20] L. Guo and S. Boukir, "Fast data selection for SVM training using ensemble margin," *Pattern Recogn. Lett.*, vol. 51, pp. 112 – 119, 2015.

[21] J. Nalepa and M. Kawulok, "A memetic algorithm to select training data for support vector machines," in *Proceedings of the 2014 Conference on Genetic and Evolutionary Computation*, ser. GECCO '14. New York, NY, USA: ACM, 2014, pp. 573–580.

[22] N. Verbiest, J. Derrac, C. Cornelis, S. García, and F. Herrera, "Evolutionary wrapper approaches for training set selection as preprocessing mechanism for support vector machines: Experimental evaluation and support vector analysis," *Appl. Soft Comp.*, vol. 38, pp. 10 – 22, 2016.

[23] H. Yu, J. Yang, and J. Han, "Classifying large data sets using SVMs with hierarchical clusters," in *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '03. ACM, 2003, pp. 306–315.

[24] M. Zechner and M. Granitzer, "A competitive learning approach to instance selection for support vector machines," in *Knowledge Science, Engineering and Management*, ser. Lecture Notes in Computer Science, D. Karagiannis and Z. Jin, Eds. Springer Berlin Heidelberg, 2009, vol. 5914, pp. 146–157.

[25] S. Bandyopadhyay, U. Maulik, C. A. Coello Coello, and W. Pedrycz, "Guest editorial: Special issue on advances in multiobjective evolutionary algorithms for data mining," *IEEE T. Evol. Comput.*, vol. 18, no. 1, pp. 1–3, 2014.

[26] Y. Jin and B. Sendhoff, "Pareto-based multiobjective machine learning: An overview and case studies," *IEEE T. Syst. Man Cy. C*, vol. 38, no. 3, pp. 397–415, May 2008.

[27] A. Mukhopadhyay, U. Maulik, S. Bandyopadhyay, and C. A. Coello Coello, "A survey of multiobjective evolutionary algorithms for data mining: Part I," *IEEE T. Evol. Comput.*, vol. 18, no. 1, pp. 4–19, 2014.

[28] ——, "Survey of multiobjective evolutionary algorithms for data mining: Part II," *IEEE T. Evol. Comput.*, vol. 18, no. 1, pp. 20–35, 2014.

[29] C.-M. Wang and Y.-F. Huang, "Evolutionary-based feature selection approaches with new criteria for data mining: A case study of credit approval data," *Expert Syst. Appl.*, vol. 36, no. 3, Part 2, pp. 5900 – 5908, 2009.

[30] Z. Wang, M. Li, and J. Li, "A multi-objective evolutionary algorithm for feature selection based on mutual information with a new redundancy measure," *Inform. Sciences*, vol. 307, pp. 73 – 88, 2015.

[31] B. Xue, M. Zhang, W. Browne, and X. Yao, "A survey on evolutionary computation approaches to feature selection," *IEEE T. Evol. Comput.*, vol. 20, no. 4, pp. 606–626, 2016.

[32] D. Martín, A. Rosete, J. Alcalá-Fdez, and F. Herrera, "A new multiobjective evolutionary algorithm for mining a reduced set of interesting positive and negative quantitative association rules," *IEEE T. Evol. Comput.*, vol. 18, no. 1, pp. 54–69, 2014.

[33] ——, "QAR-CIP-NSGA-II: A new multi-objective evolutionary algorithm to mine quantitative association rules," *Inform. Sciences*, vol. 258, pp. 1 – 28, 2014.

[34] B. Minaei-Bidgoli, R. Barmaki, and M. Nasiri, "Mining numerical association rules via multi-objective genetic algorithms," *Inform. Sciences*, vol. 233, pp. 15 – 24, 2013.

[35] J. Handl and J. Knowles, "An evolutionary approach to multiobjective clustering," *IEEE T. Evol. Comput.*, vol. 11, no. 1, pp. 56–76, 2007.

[36] A. Mukhopadhyay, U. Maulik, and S. Bandyopadhyay, "A survey of multiobjective evolutionary clustering," *ACM Comput. Surv.*, vol. 47, no. 4, pp. 61:1–61:46, 2015.

[37] M. Antonelli, P. Ducange, and F. Marcelloni, "A fast and efficient multi-objective evolutionary learning scheme for fuzzy rule-based classifiers," *Inform. Sciences*, vol. 283, pp. 36 – 54, 2014.

[38] R. C. Barros, M. P. Basgalupp, A. A. Freitas, and A. C. P. L. F. de Carvalho, "Evolutionary design of decision-tree algorithms tailored to microarray gene expression data sets," *IEEE T. Evol. Comput.*, vol. 18, no. 6, pp. 873–892, 2014.

[39] A. Rosales-Pérez, J. A. Gonzalez, C. A. Coello Coello, H. J. Escalante, and C. A. Reyes-Garcia, "Multi-objective model type selection," *Neurocomputing*, vol. 146, pp. 83 – 94, 2014.

[40] Y. J. Zheng, H. F. Ling, J. Y. Xue, and S. Y. Chen, "Population classification in fire evacuation: A multiobjective particle swarm optimization approach," *IEEE T. Evol. Comput.*, vol. 18, no. 1, pp. 70–81, 2014.

[41] C. A. Coello Coello, G. B. Lamont, and D. A. V. Veldhuizen, *Evolutionary Algorithms for Solving Multi-Objective Problems*, 2nd ed., ser. Genetic and Evolutionary Computation. Springer US, 2007.

[42] K. Deb, *Multi-Objective Optimization Using Evolutionary Algorithms*. Wiley, 2001.

[43] J. D. Schaffer, "Multiple objective optimization with vector evaluated genetic algorithms," in *Proceedings of the 1st International Conference on Genetic Algorithms*. Hillsdale, NJ, USA: L. Erlbaum Associates Inc., 1985, pp. 93–100.

[44] E. Zitzler, M. Laumanns, and L. Thiele, "SPEA2: Improving the strength pareto evolutionary algorithm for multiobjective optimization," in *Evolutionary Methods for Design Optimization and Control with Applications to Industrial Problems*, K. C. Giannakoglou, D. T. Tsahalis, J. Périaux, K. D. Papailiou, and T. Fogarty, Eds. International Center for Numerical Methods in Engineering, 2001, pp. 95–100.

[45] J. Knowles and D. Corne, "Approximating the nondominated front using the pareto archived evolution strategy," *Evol. Comput.*, vol. 8, no. 2, pp. 149–172, 2000.

[46] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE T. Evol. Comput.*, vol. 6, no. 2, pp. 182–197, 2002.

[47] Q. Zhang and H. Li, "MOEA/D: A multiobjective evolutionary algorithm based on decomposition," *IEEE T. Evol. Comput.*, vol. 11, no. 6, pp. 712–731, 2007.

[48] A. Zhou, B.-Y. Qu, H. Li, S.-Z. Zhao, P. N. Suganthan, and Q. Zhang, "Multiobjective evolutionary algorithms: A survey of the state of the art," *Swarm Evol. Comput.*, vol. 1, no. 1, pp. 32 – 49, 2011.

[49] H. Li and Q. Zhang, "Multiobjective optimization problems with complicated pareto sets, MOEA/D and NSGA-II," *IEEE T. Evol. Comput.*, vol. 13, no. 2, pp. 284–302, 2009.

[50] K. Miettinen, *Nonlinear Multiobjective Optimization*, ser. International Series in Operations Research and Management Science. Kluwer Academic Publishers, Dordrecht, 1999, vol. 12.

[51] D. L. Wilson, "Asymptotic properties of nearest neighbor rules using edited data," *IEEE T. Syst. Man Cyb.*, vol. SMC-2, no. 3, pp. 408–421, 1972.

[52] D. R. Wilson and T. R. Martinez, "Reduction techniques for instance-basedlearning algorithms," *Mach. Learn.*, vol. 38, no. 3, pp. 257–286, 2000.

[53] J. Chen, C. Zhang, X. Xue, and C.-L. Liu, "Fast instance selection for speeding up support vector machines," *knowl.-Based Syst.*, vol. 45, pp. 1 – 7, 2013.

[54] C. Thornton, F. Hutter, H. H. Hoos, and K. Leyton-Brown, "Auto-WEKA: Combined selection and hyperparameter optimization of classification algorithms," in *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '13. New York, NY, USA: ACM, 2013, pp. 847–855.

[55] M. D. Mckay, R. J. Beckman, and W. J. Conover, "A comparison of three methods for selecting values of input variables in the analysis of output from a computer code," *Technometrics*, vol. 42, no. 1, pp. 55–61, 2000.

[56] V. Cherkassky and F. M. Mülier, *Learning from data: concepts, theory, and methods*. Wiley. com, 2007.

[57] Y. Ren, L. Zhang, and P. N. Suganthan, "Ensemble classification and regression-recent developments, applications and future directions," *IEEE Comput. Intell. M.*, vol. 11, no. 1, pp. 41–53, 2016.

[58] G. Martínez-Muñoz and A. Suárez, "Aggregation ordering in bagging," in *Proc. of the IASTED International Conference on Artificial Intelligence and Applications*, 2004, pp. 258–263.

[59] Y. Zhang, S. Burer, and W. N. Street, "Ensemble pruning via semi-definite programming," *J. Mach. Learn. Res.*, vol. 7, pp. 1315–1338, Dec. 2006.

[60] Y. Freund and R. E. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," *J. Comput. Syst. Sci.*, vol. 55, no. 1, pp. 119–139, 1997.

[61] J. Alcalá, A. Fernández, J. Luengo, J. Derrac, S. García, L. Sánchez, and F. Herrera, "KEEL data-mining software tool: Data set repository, integration of algorithms and experimental analysis framework," *J. Mult.-Valued Log. S.*, vol. 17, no. 2-3, pp. 255–287, 2011.

[62] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for Support Vector Machines," *ACM T. Intel. Syst. Tec.*, vol. 2, no. 3, pp. 27:1–27:27, 2011.

[63] J. Derrac, S. García, D. Molina, and F. Herrera, "A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms," *Swarm Evol. Comput.*, vol. 1, no. 1, pp. 3 – 18, 2011.

[64] S. García, A. Fernández, J. Luengo, and F. Herrera, "A study of statistical techniques and performance measures for genetics-based machine learning: accuracy and interpretability," *Soft Comput.*, vol. 13, no. 10, pp. 959–977, 2009.

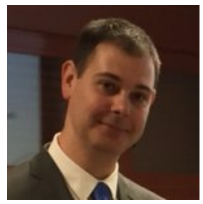
[65] N. Cristianini and J. Shawe-Taylor, *An Introduction to Support Vector Machines: And Other Kernel-based Learning Methods*. Cambridge University Press, 2000.

[66] L. Breiman, J. Friedman, C. J. Stone, and R. A. Olshen, *Classification and regression trees*. CRC press, 1984.

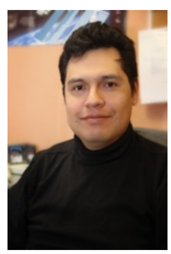
[67] L. Zhang and P. N. Suganthan, "Oblique decision tree ensemble via multisurface proximal support vector machine," *IEEE T. Cybernetics*, vol. 45, no. 10, pp. 2165–2176, 2015.

[68] M. Fernández-Delgado, E. Cernadas, S. Barro, and D. Amorim, "Do we need hundreds of classifiers to solve real world classification problems?" *J. Mach. Learn. Res.*, vol. 15, pp. 3133–3181, 2014.

Alejandro Rosales-Pérez received the B.S. degree in electronic engineering from the Instituto Tecnológico de Tuxtla Gutiérrez, Chiapas, Mexico in 2008, and the M.Sc. and Ph.D. degrees in computer science from the Instituto Nacional de Astrofísica, Óptica y Electrónica (INAOE), located at Puebla, Mexico in 2011 and 2016, respectively. He is currently with the Tecnologico de Monterrey. His doctoral thesis was awarded by the National Association of Education Institutions in Information Technology (ANIEI) in 2016 and also by the Mexican Society on Artificial Intelligence (SMIA) in 2016. He is member of the Mexican National System of Researchers (SNI). His current research interests mainly include evolutionary computation and machine learning.



Salvador García received his M.Sc. and Ph.D. in computer science from the University of Granada, Granada, Spain, in 2004 and 2008, respectively. He is currently an Associate Professor at the Department of Computer Science and Artificial Intelligence, University of Granada, Granada, Spain. He has had more than 50 papers published in international journals. He currently acts as co-editor in chief of the journal "Progress in Artificial Intelligence" (Springer) and serves as associate editor in the journals "Information Fusion", "AI Communications" and "Swarm and Evolutionary Computation". He has been selected in 2014, 2015 and 2016 as Thomson Reuters Highly Cited Researcher http://highlycited.com/ in the field of Computer Science. He is coauthor of the book entitled "Data Preprocessing in Data Mining", published by Springer. His research interests include data mining, data reduction, data complexity, imbalanced learning, semisupervised learning, statistical inference and evolutionary algorithms.



Jesus A. Gonzalez received his M.Sc. and Ph.D. degrees from The University of Texas at Arlington, Texas. He currently holds a position as a researcher at Instituto Nacional de Astrofísica, Óptica y Electrónica, Mexico. He is member of the Mexican National Researchers System (SNI). His main research areas include artificial intelligence, machine learning, data analysis, medical informatics, bioinformatics, and remote sensing.

Carlos A. Coello Coello (M'98–SM'04–F'11) received a Ph.D. in computer science from Tulane University, New Orleans, LA, USA, in 1996.

He is a Professor with Distinction (CINVESTAV-3F Researcher) at the Department of Computer Science of CINVESTAV-IPN, Mexico City, Mexico. He has authored and co-authored over 450 technical papers and book chapters. He has also co-authored the book "Evolutionary Algorithms for Solving Multi-Objective Problems" (Second Edition, Springer, 2007). His publications currently report over 33,700 citations in Google Scholar (his h-index is 73). His research interests include evolutionary multiobjective optimization and constraint-handling techniques for evolutionary algorithms.

Dr. Coello Coello has been a recipient of the 2007 National Research Award from the Mexican Academy of Sciences in the area of Exact Sciences, the 2013 IEEE Kiyo Tomiyasu Award and the 2012 National Medal of Science and Arts in the area of Physical, Mathematical and Natural Sciences. He is an Associate Editor of IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION and serves on the editorial board of 12 other international journals. He is a member of the Association for Computing Machinery, Sigma Xi, and the Mexican Academy of Science.

Francisco Herrera (SM'15) received his M.Sc. in Mathematics in 1988 and Ph.D. in Mathematics in 1991, both from the University of Granada, Spain. He is currently a Professor in the Department of Computer Science and Artificial Intelligence at the University of Granada.

He has been the supervisor of 40 Ph.D. students. He has published more than 300 journal papers that have received more than 50000 citations (Scholar Google, H-index 113). He is coauthor of the books "Data Preprocessing in Data Mining" (Springer, 2015) and "Multilabel Classification. Problem analysis, metrics and techniques" (Springer, 2016), among others.

He currently acts as Editor in Chief of the international journals "Information Fusion" (Elsevier) and "Progress in Artificial Intelligence" (Springer). He acts as editorial member of a dozen of journals.

He received the following honors and awards: ECCAI Fellow 2009, IFSA Fellow 2013, 2010 Spanish National Award on Computer Science ARITMEL to the "Spanish Engineer on Computer Science", International Cajastur "Mamdani" Prize for Soft Computing (Fourth Edition, 2010), IEEE TRANSACTIONS ON FUZZY SYSTEMS Outstanding 2008 and 2012 Paper Award (bestowed in 2011 and 2015 respectively), 2011 Lotfi A. Zadeh Prize Best paper Award (IFSA Association), 2013 AEPIA Award to a scientific career in Artificial Intelligence, and 2014 XV Andalucía Research Prize Maimónides and 2017 Andalucía Medal (by the regional government of Andalucía). He belongs to the list of Highly Cited Researchers http://highlycited.com/ (Engineering and Computer Sciences areas, Thomson Reuters).

His current research interests include among others, soft computing, information fusion, decision making, biometric, data preprocessing, data science and big data.