Information Sciences 228 (2013) 150–174

Contents lists available at SciVerse ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

# A scalable approach to simultaneous evolutionary instance and feature selection

Nicolás García-Pedrajas *, Aida de Haro-García, Javier Pérez-Rodríguez

*Department of Computing and Numerical Analysis, Edificio Einstein, 3ª Planta, University of Córdoba, Campus de Rabanales, 14071 Córdoba, Spain*

ARTICLE INFO

*Article history:*
Received 23 March 2012
Received in revised form 25 August 2012
Accepted 14 October 2012
Available online 25 October 2012

*Keywords:*
Simultaneous instance and feature selection
Instance selection
Feature selection
Instance-based learning
Very large problems

ABSTRACT

An enormous amount of information is continually being produced in current research, which poses a challenge for data mining algorithms. Many of the problems in extremely active research areas, such as bioinformatics, security and intrusion detection and text mining, involve large or enormous datasets. These datasets pose serious problems for many data mining algorithms.

One method to address very large datasets is data reduction. Among the most useful data reduction methods is simultaneous instance and feature selection. This method achieves a considerable reduction in the training data while maintaining, or even improving, the performance of the data-mining algorithm. However, it suffers from a high degree of scalability problems, even for medium-sized datasets. In this paper, we propose a new evolutionary simultaneous instance and feature selection algorithm that is scalable to millions of instances and thousands of features.

This proposal is based on the divide-and-conquer principle combined with bookkeeping. The divide-and-conquer principle allows the execution of the algorithm in linear time. Furthermore, the proposed method is easy to implement using a parallel environment and can work without loading the entire dataset into memory.

Using 50 medium-sized datasets, we will demonstrate our method's ability to match the results of state-of-the-art instance and feature selection methods while significantly reducing the time requirements. Using 13 very large datasets, we will demonstrate the scalability of our proposal to millions of instances and thousands of features.



## 1. Introduction

The overwhelming amount of data that is currently available in any field of research poses new problems for data mining and knowledge discovery methods. This enormous amount of data makes most of the existing algorithms inapplicable to many real-world problems. Two approaches have been utilized to face this problem: scaling up data mining algorithms [53] and data reduction. Nevertheless, scaling up a certain algorithm is not always feasible. Data reduction consists of removing missing, redundant, information-poor data and/or erroneous data from the dataset to obtain a tractable problem size. Data reduction techniques use different approaches; among the most common are feature selection [44], feature-value discretization [34] and instance selection [5].

Instance selection [45] consists of choosing a subset of the total available data to achieve the original purpose of the data-mining application as though all the data were being used. Different variants on instance selection exist. We can distinguish two main models [10]: instance selection as a method for prototype selection for algorithms based on prototypes (such as

* Corresponding author.
*E-mail addresses:* npedrajas@uco.es (N. García-Pedrajas), adeharo@uco.es (A. de Haro-García), javier.perez@uco.es (J. Pérez-Rodríguez).

0020-0255/$ - see front matter © 2012 Elsevier Inc. All rights reserved.
http://dx.doi.org/10.1016/j.ins.2012.10.006

*k*-Nearest Neighbors) and instance selection for obtaining the training set for a learning algorithm that uses this training set (such as classification trees or neural networks).

The problem of instance selection for instance-based learning can be defined as [7] "the isolation of the smallest set of instances that enable us to predict the class of a query instance with the same (or higher) accuracy than the original set".

It has been shown that different groups of learning algorithms need different instance selectors in order to suit their learning/search biases [9]. This may render many instance selection algorithms useless if their philosophy of design is not suitable for the problem at hand. Wrapper approaches do not assume any structure of the data or behavior of the classifier, adapting the instance selection to the performance of the classifier. Therefore, they are usually the best-performing methods. However, wrapper approaches are most computationally expensive and may over-fit.

Brighton and Mellish [7] argued that the structure of the classes formed by the instances can be very different and therefore, an instance selection algorithm can have a good performance in one problem and be very inefficient in another. They stated that the instance selection algorithm must gain some insight into the structure of the classes to perform an efficient instance selection. However, this insight is usually unavailable or very difficult to acquire, especially in real-world problems with many variables and complex boundaries between classes. In such a situation, an approach based on evolutionary computation (EC) may be helpful. Approaches based on EC do not assume any particular form of the space, the classes or the boundaries between the classes; they are only guided by each solution's ability to solve the task. In this way, the algorithm learns the relevant instances from the data without imposing any constraint in the form of classes or boundaries between classes.

Evolutionary computation has been shown [10,25] to be the most efficient method of instance selection. However, it suffers from scalability problems. The computational cost, even for moderately large datasets [27], is a serious handicap for this approach. EC-based methods are not applicable to very large datasets [17].

Feature selection is one of the most important and frequently used techniques in data preprocessing for data mining [46,14]. In contrast to other dimensionality reduction techniques, feature selection preserves the original semantics of the variables, offering the advantage of interpretability by a domain expert [58].

Feature selection has been a fertile field of research and development since the 1970s in statistical pattern recognition [50,48], machine learning [5,36] and data mining [15]. It has been widely applied to many fields, such as text categorization [42], image retrieval [56] customer relationship management [52], intrusion detection [41] and genomic analysis [64].

Feature selection can be defined as the selection of a subset of $M'$ features from a set of $M$ features, $M' < M$ such that the value of a criterion function is optimized over all the subsets of size $M'$ [51]. The objectives of feature selection are manifold. The most important objectives include the following [58]:

- To avoid over-fitting and to improve model performance, i.e., prediction performance in the case of supervised classification and better cluster detection in the case of clustering.
- To provide faster and more cost-effective models.
- To gain a deeper insight into the underlying processes that generated the data.

As shown above, many algorithms have been developed to tackle either instance or feature selection. However, few papers have aimed at simultaneous instance and feature selection [61]. One of the problems with combining both approaches is the lack of theoretical basis for developing algorithms with strong theoretical backgrounds. In the absence of this theory, we must resort to heuristics methods. An initial straightforward approach is to perform one process after the other. However, this approach does not benefit from the combination of both processes.

Although most proposed methods for both instance and feature selection address one of the problems or the other, but not both, feature and instance selection are closely related. Depending on the subset of instances considered, the relevant features might change. Conversely, different subsets of features might yield different subsets of relevant instances. Fig. 1

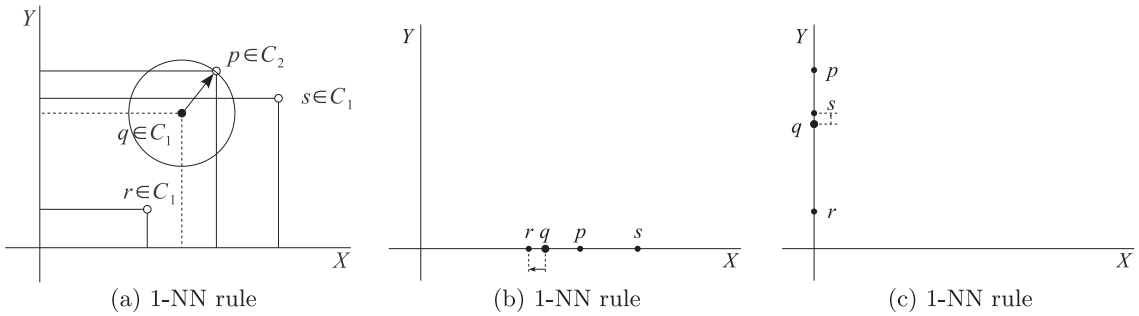


**Fig. 1.** Relationship between instance and feature selection. We have a test instance, $q$, and three training instances, $p, s$ and $r$, belonging to classes 2, 1 and 1, respectively. Using a 1-NN rule, to classify $q$ correctly, we need to select different instances depending on the features selected. If we select feature $x$, we need to select instance $r$, and if we select feature $y$, we need to select instance $s$. If both features are selected, then $q$ is misclassified.