Recent advances and emerging challenges of feature selection in the context of big data

V. Bolón-Canedo*, N. Sánchez-Marono, A. Alonso-Betanzos

Department of Computer Science, Universidade da Coruña, 15071 A Coruña, Spain

Abstract

In an era of growing data complexity and volume and the advent of big data, feature selection has a key role to play in helping reduce high-dimensionality in machine learning problems. We discuss the origins and importance of feature selection and outline recent contributions in a range of applications, from DNA microarray analysis to face recognition. Recent years have witnessed the creation of vast datasets and it seems clear that these will only continue to grow in size and number. This new big data scenario offers both opportunities and challenges to feature selection researchers, as there is a growing need for scalable yet efficient feature selection methods, given that existing methods are likely to prove inadequate.

Keywords: Feature selection, Big data, High dimensionality

1. Introduction

The “big data” phenomenon is unfolding before our eyes and its transformational nature is unquestionable. Between the dawn of time up to 2003 humanity generated a total of 5 exabytes of data and by 2008 this figure had tripled to 14.7 exabytes. Nowadays 5 exabytes of data is produced every 2 days—and the pace of production continues to rise. Because the volume, velocity, variety and complexity of datasets is continuously increasing, machine learning techniques have become indispensable in order to extract useful information from huge amounts of otherwise meaningless data. One machine learning technique is feature selection (FS), whereby attributes that allow a problem to be clearly defined are selected, while irrelevant or redundant data are ignored. Feature selection methods have traditionally been categorized as filter methods, wrapper methods or embedded methods [1], although new approaches that combine existing methods or based on other machine learning techniques are continuously appearing to deal with the challenges of today’s datasets.

In the last few years, feature selection has been successfully applied in different scenarios involving huge volumes of data, such as DNA microarray analysis, image classification, face recognition, and text classification. However, the advent of big data has raised unprecedented challenges for researchers. This paper outlines hot spots in feature selection research, aimed at encouraging scientific community to seek and embrace new opportunities and challenges that have recently arisen.

The remainder of this paper is organized as follows. Section 2 explains why feature selection is so paramount nowadays, Section 3 briefly describes the history of feature selection and reviews state-of-the-art methods, Section 4 reviews recent applications, Section 5 describes the emerging challenges that feature selection researchers need to meet in the coming years and, finally, Section 6 concludes the paper.

2. The need for feature selection

In recent years, most enterprises and organizations have stored large amounts of data in a systematic way, but without a clear idea of its potential usefulness. In addition, the growing popularity of the Internet has generated data in many different formats (text, multimedia, etc.) and from many different sources (systems, sensors, mobile devices, etc.). To be able to extract useful information from all these data, we require new analysis and processing tools. Most of these data have been generated in the last few years—as we continue to generate quintillions of bytes daily [2]. Big data—large volumes and ultrahigh dimensionality—is now a recurring feature of various machine learning application fields, such as text mining and information retrieval [3]. Weinberger et al. [4], for instance, conducted a study of a collaborative email-spam filtering task with 16 trillion unique features, whereas the study by Tan et al. [3] was based on a wide range of synthetic and real-world datasets of tens of million data points with $O(10^{14})$ features. The growing size of datasets raises an interesting challenge for the
research community; to cite Donoho et al. [5] "our task is to find a needle in a haystack, teasing the relevant information out of a vast pile of glut".

Ultra-high dimensionality implies massive memory requirements and a high computational cost for training. Generalization capacities are also undermined by what is known as the "curse of dimensionality". According to Donoho et al. [5], Bellman coined this colorful term in 1957 to describe the difficulty of optimization by exhaustive enumeration on product spaces [6]. This term refers to various phenomena that arise when analyzing and organizing data in high-dimensional spaces (with hundreds or thousands of dimensions) that do not occur in low-dimensional settings. A dataset is usually represented by a matrix where the rows are the recorded instances (or samples) and the columns are the attributes (or features) that represent the problem at hand. In order to tackle the dimensionality problem, the dataset can be summarized by finding "narrower" matrices that in some sense are close to the original. Since these narrower matrices have a smaller number of samples and/or features, they can be used much more efficiently than the original matrix. The process of finding these narrow matrices is called dimensionality reduction.

The ultra-high dimensionality not only incurs into unbearable memory requirements and high computational cost in training, but also deteriorates the generalization ability because of the "curse of dimensionality" issue. According to [5], in 1957 Bellman [6] coined the colorful term curse of dimensionality, in connection with the difficulty of optimization by exhaustive enumeration on product spaces. This phenomenon arises when analyzing and organizing data in high-dimensional spaces that do not occur in low-dimensional settings. Any dataset is usually represented by a matrix where the rows are the instances or samples that have been recorded and the columns are the attributes/features that are required to represent the problem at hand. Therefore, to tackle the curse of dimensionality problem, the dataset can be summarized by finding "narrower" matrices that in some sense are close to the original. These narrow matrices have only a small number of samples and/or a small number of attributes, and therefore can be used much more efficiently than the original large matrix can. The process of finding these narrow matrices is called dimensionality reduction.

Feature extraction is a dimensionality reduction technique that addresses the problem of finding the most compact and informative set of features for a given problem so as to improve data storage and processing efficiency. Feature extraction is decomposed into the steps of construction and selection. Feature construction methods complement human expertise in converting "raw" data into a set of useful features using preprocessing transformation procedures such as standardization, normalization, discretization, signal enhancement, and local feature extraction. Some construction methods do not alter space dimensionality, while others enlarge it, reduce it or both. It is crucial not to lose information during the feature construction stage; Gruyau and Elissseff [7] recommend that it is always better to err on the side of being too inclusive rather than run the risk of discarding useful information. Adding features may seem reasonable but it comes at a price: the increase in the dimensionality of patterns incurs the risk of losing relevant information in a sea of possibly irrelevant, noisy or redundant features. The goal of feature selection methods is to reduce the number of initial features so as to select a subset that retains enough information to obtain satisfactory results.

In a society that needs to deal with vast quantities of data and features in all kinds of disciplines, there is an urgent need for solutions to the indispensable issue of feature selection. To understand the challenges that researchers face, the next section will briefly describe the origins of feature selection and recent contributions.

3. A brief history

Feature selection is defined as the process of detecting relevant features and discarding irrelevant and redundant features with the goal of obtaining a subset of features that accurately describe a given problem with a minimum degradation of performance [1]. Theoretically, having a large number of input features might seem desirable, but the curse of dimensionality is not only an intrinsic problem of high-dimensionality data, but more a joint problem of the data and the algorithm being applied. For this reason, researchers began to select feature in a pre-processing phase in an attempt to convert their data into a lower-dimensional form.

The first research into feature selection dates back to the 1960s [8]. Hughes [9] used a general parametric model to study the accuracy of a Bayesian classifier as a function of the number of features, concluding as follows: "Measurement selection, reduction and combination are not proposed as developed techniques. Rather, they are illustrative of a framework for further investigation". Since then, research into feature selection has posed many challenges, with some researchers highly skeptical of progress; in "Discussion of Dr. Miller's paper" [10], for example, RL Plackett stated: "If variable elimination has not been sorted out after two decades of work assisted by high-speed computing, then perhaps the time has come to move on to other problems". In the 1990s, notable advances were made in feature selection used to solve machine learning problems [11–13]. Nowadays, feature selection is acknowledged to play a crucial role in reducing the dimensionality of real problems, as evidenced by the growing number of publications on the issue [1,7,14,15].

The new feature selection methods developed in the last few decades—classified as filter, wrapper or embedded methods—are based upon the relationship between the feature selection algorithm and the inductive learning method used to infer the model [1]. Feature selection methods can also be classified according to individual evaluation and subset evaluation approaches [16]; the former—also known as feature ranking—assesses individual features by assigning them weights according to relevance, whereas the latter produces candidate feature subsets based on a specific search strategy which are subsequently evaluated by some measure.

Given its ability to enhance the performance of learning algorithms, feature selection has attracted increasing interest in the field of machine learning, in processes such as clustering [17,18], regression [19,20] and classification [12,21], whether supervised or unsupervised.

Of the numerous feature selection algorithms available, several have become particularly popular among researchers. Table 1 briefly lists the most widely used feature selection methods, indicating whether they are univariate or multivariate, whether they return a ranking or subset, the original publication reference and computational complexity (where \(n\) is the number of samples and \(m\) is the number of features).

These widely used methods are part of the state of the art in feature selection. Multivariate methods generally tend to obtain better results than univariate approaches, but at a greater computational cost. There is no one-size-fits all method, as each is more suitable for particular kinds of problems. In a previous work [32], we reviewed the performance of some of these state-of-the-art algorithms in an artificial controlled scenario, checking their efficiency in tackling problems such as redundancy between features, non-linearity, noise in inputs and in output and a higher number of features than samples (as happens with DNA microarray classification). Table 2 summarizes our conclusions (more asterisks means greater suitability for a given problem). Note that versions of SVM-RFE with a linear and with a non-linear kernel were tested, but the latter (SVM-RFE-nl) could
not be applied for computational reasons to a scenario with thousands of features.

ReliefF, which appears to a good option independently of the particularities of the problem, is known for being robust and capable of dealing with incomplete and noisy data. It can be applied in most situations, has low bias, includes interaction among features and can capture local dependencies that other methods might miss. SVM-RFE also performed well, although its computational complexity prevents its use with extremely high-dimensionality datasets, especially when a non-linear kernel is used. mRMR also performed acceptably except with datasets with a high number of features. Even though it was developed with the idea of removing redundancy, mRMR was not able to discard redundant features in experiments with an artificial DNA microarray dataset, where redundancy is an acknowledged problem. Finally, poor results for correlation and redundancy were obtained with CFS. Consistency, INTERACT and InfoGain evaluated with the popular CorrAL dataset, which has four binary values necessary to predict and the class, with an extra feature that is correlated to the class label by 75%. These four methods selected the correlated feature but discarded the four genuinely relevant features. However, they were quite effective in not selecting redundant features when confronted with scenarios with thousands of features [32].

As can be seen, existing feature selection methods have their merits and demerits. Note that computational time was not taken into account in our previous analysis [32]. Nowadays, however, this factor plays a crucial role in big data problems. In general, univariate methods have an important scalability advantage, but at the cost of ignoring feature dependencies and degrading classification performance. In contrast, multivariate techniques improve classification performance, but their computational burden often means that they cannot be applied to big data.

It is evident that feature selection researchers need to adapt existing methods or propose new ones in order to cope with the challenges posed by the explosion of big data (discussed in Section 5).

4. Recent contributions

New feature selection methods are constantly being developed so there is a wide suite available to researchers. Below we assess recent developments in solutions for high-dimensionality problems in areas such as clustering [33,34], regression [35–37] and classification [38,39].

The use of different feature types and combinations are becoming the norm in many todays real applications, leading to a veritable feature explosion given rapid advances in computing and information technologies [2]. Traditionally, and because of the necessity of dealing with extremely high-dimensionality data, most newer feature selection approaches are filter methods. Nonetheless, embedded methods have increased in popularity in the last few years, given that they allow simultaneous feature selection and classification [40–42]. As for wrapper methods, these have received less attention, due to the heavy computational burden and the high risk of overfitting when the number of samples is insufficient. There is also a tendency to combine algorithms, either in the form of hybrid methods [43–46] or ensemble methods [47–51].

Apart from our own review [32], commented in the previous section, other works have reviewed the most widely used feature selection methods of the last few years. Molina et al. [52] assessed the performance of fundamental feature selection algorithms in a controlled scenario, taking into account dataset relevance, irrelevance and redundancy. Saeys et al. [53] created a basic taxonomy of classical feature selection techniques, discussing their use in bioinformatics applications. Hua et al. [54] compared some basic feature selection methods in settings involving thousands of features, using both model-based synthetic data and real data. Brown et al. [55] presented a unifying framework for information theoretic feature selection, bringing almost two decades of research into heuristic filter criteria under a single theoretical umbrella. Finally, Garcia et al. [56] dedicated a chapter in their data preprocessing book to a discussion of feature selection and an analysis of its main aspects and methods.

Another perspective is obtained when focusing on a given problem, with researchers applying different feature selection techniques in an attempt to improve performance. In this case the methodologies are highly dependent on the problem at hand. The most representative applications are discussed below.

4.1. Applications

Feature selection methods are currently being applied to problems in very different fields. Below we describe some of the most popular applications promoting the use of these methods.
4.1.1. Microarray analysis

DNA microarrays are used to collect information on gene expression differences in tissue and cell samples that could be useful for disease diagnosis or for distinguishing specific types of tumors. The sample size is usually small (often less than 100 patients) but the raw data measuring the gene expression en masse may have from 6000 to 60000 features. In this scenario, feature selection inevitably became an indispensable preprocessing step.

The earliest work in this field, in the 2000s [53], was dominated by the univariate paradigm [57–59], which is fast and scalable but which ignores feature dependencies. However, some attempts were also made with multivariate methods, as these can model feature dependencies, although they are slower and less scalable than univariate techniques [32]. Multivariate filter methods were used [60–63] and also more complex techniques such as wrapper and embedded methods [64–67]. A complete review of the most up-to-date feature selection methods used for microarray data can be found in [68], which indicates that many contributions since 2008 fall into the filter category, mostly based on information theory (see Fig. 1). The wrapper approach has largely been avoided due to the heavy computational consumption of resources and the high risk of overfitting. Although the embedded approach did not receive much attention in the infancy of microarray data classification, several proposals have emerged in recent years. Finally, it is worth noting that the recent literature reveals a tendency to combine algorithms in hybrid or ensemble methods (represented as “Other” in Fig. 1).

4.1.2. Image classification

Image classification has become a popular research field given the demand for efficient ways to classify images into categories. The numerical properties of image features are usually analyzed to determine to which category they belong. With recent advances in image capture and storage and Internet technologies, a vast amount of image data has become available to the public, from smartphone photo collections to websites and even video databases. Since image processing usually requires a large amount of computer memory and power, feature selection can help reduce the number of features needed in order to be able to correctly classify the image.

Although the explosion of data has evidenced the adequacy of feature selection techniques to deal with millions of images, a need to know precisely which features to extract from each pixel arised decades ago. A common problem in this field is that the literature refers to many models for extracting textural features from a given image, such as Markov random fields and co-occurrence features.

However, as Ohanian and Dubes pointed out [69], there is no universally best subset of features. For this reason, the feature selection task has to be specific to each problem in order to decide which type of feature to use. Jain and Zongker [70] subsequently also tried to determine whether the classification error rate for synthetic aperture radar images could be reduced by applying feature selection to a set of 18 features derived from four different texture models for each pixel. More recently, several filters were applied to the features extracted with five different texture analysis techniques [71], although, in this case, the authors were not so much interested in finding out which texture features to use, but rather in reducing the computational time necessary to extract the features. When the number of features extracted and processed is reduced, the time required is also reduced in consonance, and this can usually be achieved with minimum performance degradation.

Feature selection is also applicable to automatic image annotation. Two weighted feature selection algorithms have been proposed [72,73] to help clustering algorithms deal with a large number of data dimensions and implement scaling to a large number of keywords. Gao et al. [74] and Jin and Yang [75] introduced a solution based on hierarchical feature selection algorithms to address the problems of automatic feature extraction and image classifier training and feature subset selection, using a multi-resolution grid-based framework and a boosting algorithm to scale up support vector machines in high-dimensional feature spaces, respectively. Lu et al. [76] later presented a genetic-algorithm-based wrapper method in order to choose between MPEG-7 feature descriptors. Meanwhile, Little and Ruger [77] proposed a non-parametric density estimation algorithm to evaluate subsets of features. More recently, Ma et al. [78] proposed a novel method, based on a sparsity-based model, that jointly selected the most relevant features from all data points while also uncovering the shared subspace of original features (beneficial for multi-label learning).

4.1.3. Face recognition

Identifying a human face is a complex visual recognition problem. In the last few decades, face recognition has become one of the most active research fields due to its numerous commercial and legal applications. A common application is to identify or verify a person from a digital image or a video-sourced frame by comparing selected facial features from the image with features in a facial database. An important issue in this field is to determine which image features are the most informative for recognition purposes. Unfortunately, this is no trivial task since great redundancy exists in object images; moreover, facial databases contain a large number of features but a reduced number of samples. Feature selection algorithms for face recognition have recently been suggested as a way of solving these issues.

The filter method of feature selection is a common choice, mainly due to its low computational cost compared to the wrapper or embedded methods. Yang et al. [79] presented a method based on the physical meaning of the generalized Fisher criterion in order to choose the most discriminative features for recognition. Lu et al. [76] proposed a novel method for choosing a subset of original features containing the most essential information; called principal feature analysis (PFA), it is similar to principal component analysis (PCA) methods. de S Matos et al. [80] introduced a face recognition method based on discrete cosine transform (DCT) coefficient selection. More recently, Lee et al. [81] introduced a new color face recognition method that uses sequential floating forward search (SFFS) to obtain a set of optimal color components for recognition purposes. It is also worth noting that several proposed methods based on evolutionary computation techniques have been demonstrated to be successful in this field [82–85].

![Fig. 1. Recent feature selection contributions to DNA microarray analysis according to the data collected in [68].](image-url)
4.1.4. Text classification

The goal of text classification is to categorize documents into a fixed number of predefined categories or labels. This problem has become particularly relevant to Internet applications for spam detection and shopping and auction websites. Each unique word in a document is considered a feature. However, since this implies far more input features than examples (usually by more than an order of magnitude), it is necessary to select a fraction of the vocabulary and so allow the learning algorithm to reduce computational, storage and/or bandwidth requirements.

A preprocessing stage is usually applied prior to feature selection to eliminate rare words and to merge word forms such as plurals and verb conjugations into the same term. There are several approaches to representing the feature values, for instance, a Boolean value to indicate if a word is present or absent or including the count of word occurrences. Even after this preprocessing step, the number of possible words in a document may still be high, so feature selection is paramount. A number of techniques have been developed and applied to this problem in recent years. Forman [86] proposed a novel feature selection metric, called bi-normal separation (BNS), which is a useful heuristic for increased scalability when used with wrapper techniques of text classification. Kim et al. [87] applied several novel feature selection methods to clustered data, while Dasgupta et al. [88] proposed an unsupervised feature selection strategy that theoretically guarantees the generalization power of the resulting classification function with respect to the classification function based on all the features. Forman [89] reviewed a series of filters applied to binary, multiclass and hierarchical text classification problems, focusing especially on scalability. Uguz [43] subsequently proposed a two-stage feature selection method for text categorization using InfoGain, PCA and genetic algorithms, obtaining high categorization effectiveness for two classical benchmark datasets. Shang et al. [90] recently proposed a novel metric called global information gain (GIG) that avoids redundancy naturally and also introduced an efficient feature selection method called maximizing global information gain (MGIG), which has proved to be effective for feature selection in the text domain. More recently, Baccianella et al. [91] presented six novel feature selection methods specifically devised for ordinal text classification.

As can be seen, most machine learning methods can take advantage of feature selection for preprocessing purposes, since it usually improves accuracy and reduces the computational cost of pattern recognition. Our brief review has covered the more popular applications for feature selection, but the literature describes many more application areas, including intrusion detection [92] and machinery fault diagnosis [93].

5. Hot spots: what next in feature selection?

As mentioned at the beginning of this paper, ongoing advances in computer-based technologies have enabled researchers and engineers to collect data at an increasingly fast pace. To address the challenge of analyzing these data, feature selection becomes an imperative preprocessing step that needs to be adapted and improved to be able to handle high-dimensional data. We have highlighted the need for feature selection and discussed recent contributions in several different application areas. However, in the new big data scenario, an important number of challenges are emerging, representing current hot spots in feature selection research.

5.1. Millions of dimensions

In the new era of big data, machine learning methods need to be able to deal with the unprecedented scale of data. Analogous to big data, the term “big dimensionality” has been coined to refer to the unprecedented number of features arriving at levels that are rendering existing machine learning methods inadequate [2].

The widely-used UCI Machine Learning Repository [94] indicates that, in the 1980s, the maximum dimensionality of data was only about 100. By the 1990s, this number had increased to more than 1500 and, by 2009, to more than 3 million. If we focus on the number of attributes of the UCI datasets, 13 have more than 5000 features and most have a samples/features ratio below 0—a level that potentially hinders any learning process. Illustratively, Fig. 2 shows the number of features of the highest dimensionality datasets included in the UCI Machine Learning Repository in the last seven years. In the popular LIBSVM Database [95] the maximum dimensionality of the data was about 62,000 in the 1990s, increasing to some 16 million in the 2000s and to more than 29 million in the 2010s; analogously, 20 of the existing 92 datasets have more than 5000 features and 11 datasets have many more features than samples. Seven of the datasets included in these two repositories in the last 9 years have dimensionality in the order of millions. Apart from these generic repositories, there are others with specific high dimensionality problems, such as the...
5.2. Scalability

Most existing learning algorithms were developed when data-set sizes were much smaller, but nowadays different solutions are required for the case of small-scale versus large-scale learning problems. Small-scale learning problems are subject to the usual approximation–estimation trade-off, but this trade-off is more complex in the case of large-scale learning problems, not only because of accuracy but also due to the computational complexity of the learning algorithm. Moreover, since most algorithms were designed under the assumption that the dataset would be represented as a single memory-resident table, these algorithms are useless when the entire dataset does not fit in the main memory. Dataset size is therefore one reason for scaling up machine learning algorithms. However, there are other settings where a researcher could find the scale of a machine learning task daunting [98], for instance:

- Model and algorithm complexity: A number of high-accuracy learning algorithms either rely on complex, non-linear models, or employ computationally expensive subroutines.
- Inference time constraints: Applications that involve sensing, such as robot navigation or speech recognition, require predictions to be made in real time.
- Prediction cascades: Applications that require sequential, interdependent predictions have a highly complex joint output space.
- Model selection and parameter sweeps: Tuning learning algorithm hyper-parameters and evaluating statistical significance require multiple learning executions.

For all these reasons, scaling up learning algorithms is a trend issue. Cases in point are the workshop “PASCAL Large Scale Learning Challenge” held at the 25th International Conference on Machine learning (ICML08) and the “Big Learning” workshop held at the 2011 conference of the Neural Information Processing Systems Foundation (NIPS2011). Scaling up is desirable because increasing the size of the training set often increases the accuracy of algorithms [99]. In scaling up learning algorithms, the issue is not so much one of speeding up a slow algorithm as one of turning an impracticable algorithm into a practical one. Today, there is a consensus in machine learning and data mining communities that data volume presents an immediate challenge pertaining to the scalability issue [2]. The crucial point is seldom how fast you can run on a particular problem, but rather how large a problem you can deal with [100].

Scalability is defined as the impact of an increase in the size of the training set on the computational performance of an algorithm in terms of accuracy, training time and allocated memory. Thus the challenge is to find a trade-off among these criteria—in other words, to obtain “good enough” solutions as “fast” as possible and as “efficiently” as possible. As explained before, this issue becomes critical in situations in which there are temporal or spatial constraints as happens with real-time applications dealing with large datasets, unapproachable computational problems requiring learning and initial prototyping requiring rapidly implemented solutions.

Similarly to instance selection, which aims at discarding superfluous, i.e., redundant or irrelevant, samples [101], feature selection can scale machine learning algorithms by reducing input dimensionality and therefore algorithm run-time. However, when dealing with a dataset containing a huge number of both features and samples, the scalability of the feature selection method also assumes crucial importance. Since most existing feature selection techniques were designed to process small-scale data, their efficiency is likely to be downgraded, if not reduced totally, with high-dimensional data. Fig. 3 shows run-time responses to modifications to the number of features and samples for four well-known feature selection ranker methods applied to the SD1 dataset, a synthetic dataset that simulates DNA microarray data [102].

In this scenario, feature selection researchers need to focus not only on the accuracy of the selection but also on other aspects. One such factor is stability, defined as the sensitivity of the results to training set variations. The other important factor, scalability, refers to feature selection response to an increasingly large training set. Few studies have been published regarding filter behavior in small training sets with a large number of features [55,103–105] and even fewer on the issue of scalability [106]. What studies do exist are mainly focused on scalability in particular applications [107], modifications of existing approaches [108], combinations of instance and feature selection strategies [109] and online [110] and parallel [111] approaches. A recent paper by Tan et al. [3] describes a new adaptive feature-scaling method—applied to several synthetic and real big datasets; based on group feature selection and multiple kernel learning, it enables scalability to big data scenarios.

Broadly speaking, although most classical univariate feature selection approaches (with each feature considered separately) have an important advantage in terms of scalability, they ignore feature dependencies and thus potentially perform less well than other feature selection techniques. Multivariate techniques, in contrast, may improve performance, but at the cost of reduced scalability [112]. The scalability of a feature selection method is thus crucial and deserves more attention from the scientific community. One of the solutions commonly adopted to deal with the scalability issue is to distribute the data into several processors, discussed in the following section.

5.3. Distributed feature selection

Traditionally, feature selection is applied in a centralized manner, i.e., a single learning model is used to solve a given problem.
However, since nowadays data may be distributed, feature selection can take advantage of processing multiple subsets in sequence or concurrently. There are several ways to distribute a feature selection task [113] (note: real-time processing will be discussed in subsection 5.4):

(i) **The data is together in one very large dataset.** The data can be distributed on several processors, an identical feature selection algorithm can be run on each and the results combined.

(ii) **The data may be in different datasets in different locations** (e.g., in different parts of a company or even in different cooperating organizations). As for the previous case, an identical feature selection algorithm can be run on each and the results combined.

(iii) **Large volumes of data may be arriving in a continuous infinite stream in real time.** If the data is all streaming into a single processor, different parts can be processed by different processors acting in parallel. If the data is streaming into different processors, they can be handled as above.

(iv) **The dataset is not particularly large but different feature selection methods need to be applied to learn unseen instances and combine results** (by some kind of voting system). The whole dataset may be in a single processor, accessed by identical or different feature selection methods that access all or part of the data.

This last approach, known as *ensemble learning*, has recently been receiving a great deal of attention [114]. The interest in this approach is due to the fact that, since high variance is a problem of feature selection methods, one possible solution is to use an ensemble approach based on combining methods [115,51].

The individual selectors in an ensemble are known as base selectors. If the base selectors are all of the same kind, the ensemble is termed homogeneous. Ensemble feature selection is accomplished in two steps. First, a set of different feature selectors are applied, on the principle that there is no universally optimal technique and that there may be more than one subset of features that discriminate data similarly. Second, each feature selector produces outputs that are subsequently aggregated via consensus feature ranking, choosing the most frequent features selected, etc. [116].

Fig. 4 shows different partitioned feature selection scenarios. **Fig. 4a** represents the situation described in (i): the original data is distributed between several processors and local results are combined in a final result. **Fig. 4b** represents the situation described in (iv): the data is replicated on different processors, local results are obtained as a consequence of applying different feature selection methods and, again, local results are combined into a global result.

As mentioned, most existing feature selection methods are not expected to scale efficiently when dealing with millions of features; indeed, they may even become inapplicable. A possible solution might be to distribute the data, run feature selection on each partition and then combine the results. The two main approaches to partitioned data distribution are by feature (vertically) or by sample (horizontally). Distributed learning has been used to scale up datasets that are too large for batch learning by samples [117–119]. While distributed learning is not common, there have been some developments regarding data distribution by features [120,121]. One proposal is a distributed method where data partitioning is both vertical and horizontal [122]. Another is a distributed parallel feature selection method that can read data in distributed form and perform parallel feature selection in symmetric multiprocessing mode via multithreading and massively parallel processing [111]. However, when dealing with big dimensionality datasets, researchers, of necessity, have to partition by features. In the case of DNA microarray data, the small sample size combined with big dimensionality prevent the use of horizontal partitioning. However, the previous mentioned vertical partitioning methods do not take into account some of the particularities of these datasets, such as the high redundancy among features, as is done in the methods described by Sharma et al. [123] and Bolón-Canedo et al. [124], the latter at a much lower computational cost.

Several paradigms for performing distributed learning have emerged in the last decade. MapReduce [125] is one such popular programming model with an associated implementation for processing and generating large data sets with a parallel, distributed algorithm on a cluster. Hadoop, developed by Cutting and Cafarella in 2005 [126], is a set of algorithms for distributed storage and distributed processing of very large datasets on computer clusters; it is built from commodity hardware and has a processing part based on MapReduce. Developed more recently, is Apache Spark [127], a fast, general engine for large-scale data processing, popular among machine learning researchers due to its suitability for iterative procedures. Developed within the Apache Spark
paradigm was MLlib [128], created as a scalable machine learning library containing algorithms. Although it already includes a number of learning algorithms such as SVM and naive Bayes classification and k-means clustering, as yet, it includes no feature selection algorithms. This poses a challenge for machine learning researchers, as well as offering an opportunity to initiate a new line of research.

Another open line of research is the use of graphics processing units (GPUs) to distribute and thus accelerate calculations made in feature selection algorithms. With many applications to physics simulations, signal processing, financial modelling, neural networks, and countless other fields, parallel algorithms running on GPUs often achieve up to 100X speedup over similar CPU algorithms. The challenge now is to take advantage of GPU capabilities to adapt existing state-of-the-art feature selection methods to be able to cope effectively and accurately with millions of features.

5.4. Real-time processing

Data is being collected at an unprecedented fast pace and, correspondingly, needs to be processed rapidly. Social media networks and portable devices dominate our day-to-day and we need sophisticated methods that are capable of dealing with vast amounts of data in real time, e.g., for spam detection and video/image detection [2].

Classical batch learning algorithms cannot deal with continuously flowing data streams, which require online approaches. Online learning [129], which is the process of continuously revising and refining a model by incorporating new data on-demand, has become a trending area in the last few years, because it solves important problems for processes occurring in time (e.g., a stock value given its history and other external factors). The mapping process is updated in real time and as more samples are obtained. Online learning can also be useful for extremely large-scale datasets, since a possible solution might be to learn data in a sequential fashion.

Online feature selection has not received the same attention as online learning [129]. Nonetheless, a few studies exist that describe attempts to select relevant features in a scenario in which both new samples and new features arise. Zhang et al. [130] proposed an incremental feature subset selection algorithm which, originating in the Boolean matrix technique, efficiently selects useful features for the given data objective. Nevertheless, the efficiency of the feature selection method was not tested with an incremental machine learning algorithm. Katakis et al. [131] proposed the idea of a dynamic feature space, whereby features selected from an initial collection of training documents are subsequently considered by the learner during system operation. However, features may vary over time and an initial training set is often not available in some applications. Katakis et al. [131] combined incremental feature selection with what they called a feature-based learning algorithm to deal with online learning in high-dimensional data streams. This same framework was applied to the special case of concept drift [132] inherent to textual data streams (i.e., the appearance of new predictive words over time). The problem with this approach is that features are assumed to have discrete values. Perkins et al. [133] described a novel and flexible approach, called grafting, which treats the selection of suitable features as an integral part of learning a predictor in a regularized learning framework. What makes grafting suitable for large problems is that it operates in an incremental iterative fashion, gradually building up a feature set while training a predictor model using gradient descent. Perkins and Theiler [134] tackled the problem of features arriving one at a time rather than being available from the outset; their approach, called online feature selection (OFS), assumes that, for whatever reason, it is not worthwhile waiting until all features have arrived before learning begins. They thus derived a “good enough” mapping function from inputs to outputs based on a subset of features seen so date. The potential of OFS in the image processing domain was demonstrated by applying it to the problem of edge detection [135]. A promising alternative method, called online streaming feature selection (OSFS), selects strongly relevant and non-redundant features [136]. In yet another approach, two novel online feature selection methods use relevance to select features on the fly; redundancy is only later taken into account, when these features come via streaming, but the number of training examples remains fixed [137]. Finally, the literature contains a number of studies referring to online feature selection and classification. One is an online learning algorithm for feature extraction and classification, implemented for impact acoustics signals to sort hazelnut kernels [138]. Another, by Levi and Ullman [139], proposed classifying images by ongoing feature selection, although their approach only uses a small subset of the training data at each stage. Yet another describes online feature selection performed based on the weights assigned to each classifier input [140].

As can be seen, online feature selection has been dealt with mostly on an individual basis, i.e., by pre-selecting features in a step independent of the online machine learning step, or by performing online feature selection without subsequent online classification. Therefore, achieving real-time analysis and prediction for high-dimensional datasets remains a challenge for computational intelligence on portable platforms. The question now is to find flexible feature selection methods capable of modifying the selected subset of features as new training samples arrive. It is also desirable for these methods to be executed in a dynamic feature space that would initially be empty but would add features as new information arrived (e.g., documents in their text categorization application).

5.5. Feature cost

As can be seen throughout this paper, although new feature selection methods are being developed, most focus more on removing irrelevant and redundant features rather than on the cost of obtaining input features. The cost associated with a feature is

---

**Fig. 5.** Three examples of features with a cost. (a) Computational cost has become a significant issue in big data scenarios. (b) In medical diagnosis, the cost of a scan is not the same as the cost of a blood test. (c) Zoometry in living animals has an added cost in the form of risk.
related to different concepts. For example, a pattern in medical diagnostics consists of observable symptoms (such as age and sex), which have no cost, along with the results of tests, which are associated with costs and risks; as one example, invasive exploratory surgery is much more expensive and risky than a blood test [141]. Another example of feature extraction risk is given by Bahamonde et al. [142], where zoometry on living animals is necessary to evaluate the merits of beef cattle. Another cost is that related to computational issues. In the medical imaging field, feature extraction from a medical image can be computationally costly; moreover, in the texture analysis technique known as co-occurrence features [143], the fact that the computational cost varies implies different computational times. In real-time applications, the space complexity is negligible, whereas the time complexity is crucial [144]. Fig. 5 shows some examples of feature cost.¹

As one may notice, features with an associated cost can be found in many real-life applications. However, this has not been the focus of much attention for machine learning researchers. Most of the works have only considered the mis-classification cost, which is the penalty that is received while deciding that an object belongs to a class that it is not the real one [145].

There have been some attempts to balance the contribution of features and their cost. For instance, in classification, Friedman [146] included a regularization term to the traditional linear discriminant analysis (LDA); the left side term of their cost function evaluates error and the right side term is a regularization parameter weighted with \( i \), providing a framework in which different regularized solutions depend on \( i \) value. Related to feature extraction, You et al. [147] proposed a criterion to select kernel parameters based on maximizing between-class scattering and minimizing within-class scattering. A general classification framework for application to face recognition was proposed by Wright et al. [148] to study feature extraction and robustness to occlusion by obtaining a sparse representation. This method, instead of measuring correlation between feature and class, evaluates the representation error.

Despite the previous attempts at classification and feature extraction, there are a smaller number of works that deal with this issue in feature selection. In the early 1990s, Feddema et al. [144] developed methodologies for the automatic selection of image features by a robot. For this selection process, they employed a weighted criterion that took into account the computational cost of features, i.e., the time and space complexities of the feature extraction process. Several years later, Yang and Honavar [141] proposed a genetic algorithm to perform feature subset selection, designing the fitness function on the basis of the two criteria of neural network accuracy in classification and classification cost (defined as the cost of measuring the value of a particular feature needed for classification, the risk involved, etc.). Huang and Wang [149] also used a genetic algorithm for feature selection and parameter optimization for a support vector machine, using classification accuracy, the number of selected features and the feature cost as criteria to design the fitness function.


In recent years, several dimensionality reduction techniques for data visualization and preprocessing have been developed. However, although the aim may be better visualization, most techniques have the limitation that the features being visualized are transformations of the original features [153–155]. Thus, when model interpretability is important, feature selection is the preferred technique for dimensionality reduction.

A model is only as good as its features, for which reason features have played and will continue to play a preponderant role in model interpretability. Users have a twofold need for interpretability and transparency in feature selection and model creation processes: (i) they need more interactive model visualizations where they can change input parameters to better interact with the model and visualize future scenarios and (ii) they need more interactive feature selection process where, using interactive visualizations, they are empowered to iterate through different feature subsets rather than be tied to a specific subset chosen by an algorithm.

Some recent works describe using feature selection to improve the interpretability of models obtained in different fields. One example is a method for the automatic and iterative refinement of a recommender system, in which the feature selection step selects the best characteristics of the initial model in order to automatically refine it [156]. Another is the use of feature selection to improve decision trees—representing agents simulating personnel in an organization so as to model sustainability behaviors—through an expert review of their theoretical consistency [157]. Yet another is a generative topographic mapping-based data visualization approach that estimates feature saliency simultaneously as the visualization model is trained [158]. Krause et al. [159] describe a tool in which visualization helps users develop a predictive model of their problem by allowing them to rank features (according to predefined scores), combine features and detect similarities between dimensions.

However, data is everywhere, continuously increasing, and heterogeneous. We are witnessing a form of Diogenes syndrome referring to data: organizations are collecting and storing tonnes of data, but most do not have the tools or the resources to access and generate strategic reports and insights from their data. Organizations need to gather data in a meaningful way, so as to evolve from a data-rich/knowledge-poor scenario to a data-rich/knowledge-rich scenario. The challenge is to enable user-friendly visualization of results so as to enhance interpretability. The complexity implied by big data applications also underscores the need to limit the growth in visualization complexity. Thus, even though feature selection and visualization have been dealt with in relative isolation from each other in most research
to date, the visualization of data features may have an important role to play in real-world high dimensionality scenarios. However, it is also important to bear in mind that, although visualization tools are increasingly used to interpret and make complex data understandable, the quality of associated decision making is often impaired due to the fact that the tools fail to address the role played by heuristics, biases, etc. in human–computer interactive settings. Therefore, interactive tools similar to that described by Krause et al. [159] are an interesting line of research.

6. Discussion and conclusions

Feature selection has been widely used as a preprocessing step that reduces the dimensions of a problem and improves classification accuracy. The need for this kind of technique has increased dramatically in recent years in order to cope with scenarios characterized by both a high number of input features and/or of samples. In other words, the big data explosion now has the added problem of big dimensionality.

This paper analyzed the paramount need for feature selection and briefly reviewed the most popular feature selection methods and some typical applications. Although feature selection may well be one of the better known preprocessing techniques, it is important not to overlook the factors affecting feature selection choices. For instance, it is important to choose an adequate discretization technique, given that some feature selection methods—especially those from the information theory field—were developed to work with discrete data. Indeed, it has been demonstrated that the choice of discretization method affects the results of the feature selection process [160,161].

The need for new preprocessing techniques not only affects the decision as to which feature selection method to use but also has impact on the other processing stages. Some methods return an ordered ranking of features according to some metric, in which case the goodness of the features needs to be evaluated and a decision made as to where to set the threshold. Classification algorithms are the preferred means of evaluating features, although this usually implies an additional computational burden. Moreover, the use of certain classifiers may obscure the effectiveness of the feature selection process. There is some evidence that good accuracy can be achieved by classifiers with embedded capacities even when the set of selected features is less than optimal [32]. Another issue that may affect the choice of a feature selection method or classifier is the intrinsic complexity of the data [162,163].

In certain applications, in fact, the medical domain, it is often desirable to be able to interpret the power of each feature. In this case, it is better to use a feature selection method that returns a score (e.g., ReliefF, InfoGain, Chi-Squared) instead of methods that return only a ranking or a subset of features, where the specific power of features is overlooked. Care needs to be taken when deciding which feature selection method to use, as this will depend on the problem, the type of data (numerical or discrete, complexity, etc.) and future use of the data.

In conclusion, the suitability of using feature selection has been demonstrated in a variety of applications that require the processing of huge amounts of data. However, recent years have witnessed the creation of datasets with features numbering in the order of millions; furthermore, it seems clear that this number will only continue to increase, given the rapid advances in computing and information technologies. This new scenario offers both opportunities and challenges to machine learning researchers. There is a growing need for scalable yet efficient feature selection methods, given that existing methods are likely to prove inadequate to cope with such an unprecedented number of features. Furthermore, new needs are arising in society, such as in the areas of distributed learning and real-time processing, where an important gap that needs to be filled is developing. Beyond a shadow of doubt, the explosion in the number of features points to a number of hot spots for feature selection researchers to launch new lines of research.

Acknowledgments

This research has been economically supported in part by the Ministerio de Economía y Competitividad of the Spanish Government through the research project TIN 2012-37954 (funded by European Union FEDER funds), and by the Consellería de Industria of the Xunta de Galicia through the research project GRC2014/035.

References


A. Amine, A. El Akadi, M. Rizza, D. Aboutajdine, Ga-vm and mutual information based frequency feature selection for face recognition, GSCM-IRIT, Faculty of Sciences, Mohammed V University, BP, 1014, 2009.


