Artificial Intelligence Review (2020) 53:907–948
https://doi.org/10.1007/s10462-019-09682-y

A review of unsupervised feature selection methods

Saúl Solorio-Fernández[1] · **J. Ariel Carrasco-Ochoa**[1] · **José Fco. Martínez-Trinidad**[1]

Published online: 29 January 2019

Abstract
In recent years, unsupervised feature selection methods have raised considerable interest in many research areas; this is mainly due to their ability to identify and select relevant features without needing class label information. In this paper, we provide a comprehensive and structured review of the most relevant and recent unsupervised feature selection methods reported in the literature. We present a taxonomy of these methods and describe the main characteristics and the fundamental ideas they are based on. Additionally, we summarized the advantages and disadvantages of the general lines in which we have categorized the methods analyzed in this review. Moreover, an experimental comparison among the most representative methods of each approach is also presented. Finally, we discuss some important open challenges in this research area.

Keywords Unsupervised learning · Dimensionality reduction · Unsupervised feature selection · Feature selection for clustering

1 Introduction

Feature selection (Liu and Motoda 1998, 2007; Guyon et al. 2003) (also known as attribute selection) appears in different areas such as pattern recognition (Tou and González 1974; Theodoridis and Koutroumbas 2008a), machine learning (Kotsiantis 2011; Hall 1999), data mining (García et al. 2015; Chakrabarti et al. 2008) and statistical analysis (Webb 2003; Friedman et al. 2001). In all these areas, often the objects[1] under study include in their description irrelevant and redundant features (Ritter 2015), which can significantly affect the analysis of the data, resulting in biases or even incorrect models (Zhao and Liu 2011). Feature selection is the process of selecting the most useful features for building models in

[1] Also called instances, observations or samples; commonly represented as vectors.

✉ Saúl Solorio-Fernández
sausolofer@inaoep.mx

J. Ariel Carrasco-Ochoa
ariel@inaoep.mx

José Fco. Martínez-Trinidad
fmartine@inaoep.mx

[1] Computer Sciences Department, Instituto Nacional de Atrofísica, Óptica y Electrónica, Luis Enrique Erro # 1, Tonantzintla, 72840 Puebla, Mexico

tasks like classification, regression or clustering. Moreover, feature selection not only reduces the dimensionality of the data facilitating their visualization and understanding; but also it commonly leads to more compact models with better generalization ability (Pal and Mitra 2004). All these characteristics make feature selection an interesting research area, wherein the last decades, numerous feature selection methods have been introduced.

According to the information available in the datasets, feature selection methods can be classified as supervised (Kotsiantis 2011; Tang et al. 2014), semi-supervised (Sheikhpour et al. 2017) and unsupervised (Alelyani et al. 2013). The former require a set of labeled data (supervised dataset) in order to identify and select relevant features; this label, assigned to each object in the dataset, can be a category, an ordered value or a real value (depending on the specific task). Semi-supervised methods only require that some objects be labeled. On the other hand, Unsupervised Feature Selection (UFS) methods (Dy and Brodley 2004; Alelyani et al. 2013; Fowlkes et al. 1988) do not require a supervised dataset.

Over the last decades, many feature selection methods have been proposed, most of them developed for supervised classification tasks (Tang et al. 2014). However, due to the technological development raised in the last years, as well as the vast amount of unlabeled data generated in different applications such as text mining (Feldman and Sanger 2006; Bharti and kumar Singh 2014; Forman 2003), bioinformatics (Saeys et al. 2007), image retrieval (Yasmin et al. 2014; Swets and Weng 1995), social media (Zafarani et al. 2014; Tang and Liu 2014) and intrusion detection (Ahmed et al. 2016; Lee et al. 2000; Agrawal and Agrawal 2015; Ambusaidi et al. 2015), to name a few; UFS methods have gained significant interest in the scientific community. Moreover, according to (Guyon et al. 2003; Niijima and Okuno 2009; Devakumari and Thangavel 2010), UFS methods have two important advantages. (1) they are unbiased and perform well when prior knowledge is not available, and (2) they can reduce the risk of data overfitting in contrast to supervised feature selection methods that may be unable to deal with a new class of data.

In the same way as in supervised and semi-supervised feature selection, according to the strategy used for selecting features, Unsupervised Feature Selection methods can be divided into three main approaches (Alelyani et al. 2013; Dong and Liu 2018):

- Filter methods select the most relevant features through the data itself, i.e., features are evaluated based on intrinsic properties of the data, without using any clustering algorithm that could guide the search of relevant features. The main characteristic of filter methods is their speed and scalability.
- Wrapper methods evaluate feature subsets using the results of a specific clustering algorithm. Methods developed under this approach are characterized by finding features subsets that contribute to improving the quality of the results of the clustering algorithm used for the selection. However, the main disadvantage of wrapper methods is that they usually have a high computational cost, and they are limited to be used in conjunction with a particular clustering algorithm.
- Hybrid methods try to exploit the qualities of both approaches, filter, and wrapper, trying to have a good compromise between efficiency (computational effort) and effectiveness (quality in the associated objective task when using the selected features).

Currently, in the literature we can find some reviews about feature selection (Cai et al. 2018; Sheikhpour et al. 2017; Miao and Niu 2016; Li et al. 2016; Ang et al. 2016; Chandrashekar and Sahin 2014; Vergara and Estévez 2014; Kotsiantis 2011; Liu et al. 2005; Yu 2005). Nevertheless, all of them are focused either on supervised/semi-supervised feature selection, or feature selection in general; while some reviews concentrate on describing feature selection applied to specific domains (Lee et al. 2017; Bharti and kumar Singh 2014; Lazar et al. 2012;

Mugunthadevi et al. 2011; Saeys et al. 2007). As far we know, the most similar work to our review is presented in Alelyani et al. (2013), where feature selection for clustering is reviewed. However, in Alelyani et al. (2013) only a few relevant methods of the state-of-the-art are mentioned; and mainly focusing on feature selection methods designed exclusively for specific domains, such as text data, streaming data, and link data. In our paper, we focus on Unsupervised Feature Selection (UFS). We intend to provide a big picture over UFS methods throughout a comprehensive and structured review of the most relevant (most referenced) and recent works of the state-of-the-art; describing their main characteristics and the fundamental ideas these methods are based on. Furthermore, in our review, we present a taxonomy of reviewed UFS methods; classifying them according to their approach, type, and subtype, and pointing out the major advantages and disadvantages of these general lines. Additionally, we perform an experimental comparison, on standard public datasets among the most representative methods of each approach and conclude our review highlighting some open challenges in Unsupervised Feature Selection. To the best of our knowledge, this is the first comprehensive review in Unsupervised Feature Selection that provides a general perspective to the audience, practitioners and academics, about the most relevant and recent feature selection methods in this field of research.

The structure of this paper is as follows: in Sect. 2, the main Unsupervised Feature Selection methods proposed in the literature are reviewed. An analysis and discussion of the UFS methods is presented in Sect. 3. In this section, the advantages, disadvantages, feature selection criteria, analysis of the performance evaluation, and the experimental comparison of the reviewed UFS methods are provided. Finally, in Sect. 4, our conclusions are exposed; pointing out some open challenges and research directions in Unsupervised Feature Selection.

2 Unsupervised feature selection methods

As we have commented in the previous section, Unsupervised Feature Selection (UFS) methods can be categorized according to the strategy used for selecting features as filter, wrapper, and hybrid methods. In this section, first, we organize the UFS methods reported in the literature into the taxonomy shown in Fig. 1. Then, we describe each one of these methods by focusing on their main characteristics and the ideas they are based on.

2.1 Filter approach

According to Alelyani et al. (2013), UFS methods based on the filter approach can be categorized as univariate and multivariate. The former, also known as ranking based UFS methods use some criteria to evaluate each feature in order to get an ordered list (ranking) of features, where the final feature subset is selected according to this order. Such methods can effectively identify and remove irrelevant features, but they are unable to remove redundant ones since they do not take into account possible dependencies among features. On the other hand, multivariate filter methods evaluate the relevance of the features jointly rather than individually. Multivariate methods can handle redundant and irrelevant features; thus, in many cases, the accuracy reached by learning algorithms using the subset of features selected by multivariate methods is better than the one achieved by using univariate methods (Tabakhi et al. 2015).

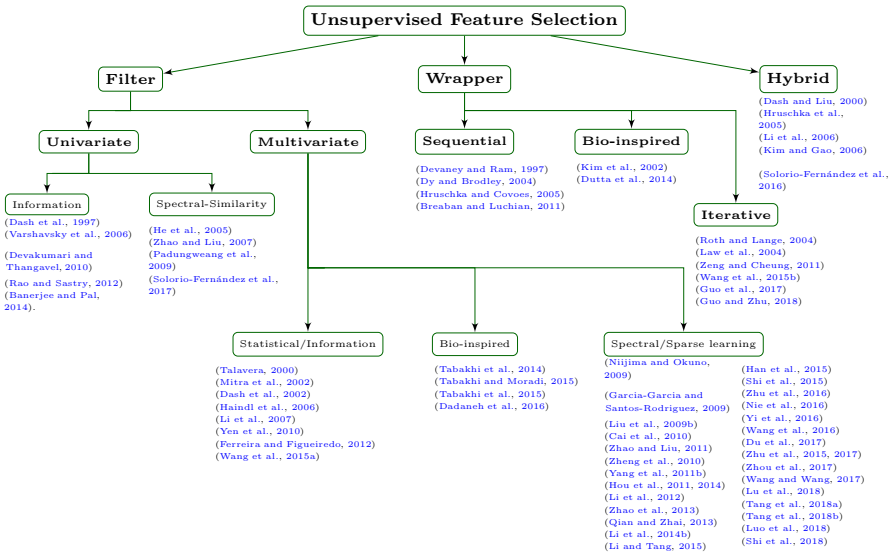

Fig. 1 Taxonomy of unsupervised feature selection (UFS) methods

2.1.1 Univariate filter methods

Within the univariate filter methods, two main groups can be highlighted: methods that assess the relevancy of each feature based on Information Theory (Cover and Thomas 2006), and those methods that evaluate features based on Spectral Analysis (manifold learning) (Chung 1997; Luxburg 2007) using the similarities among objects. The former follow the idea of assessing the degree of dispersion of the data through measures such as entropy, divergence, mutual information, among others, to identify cluster structures in the data. On the other hand, methods based on Spectral Analysis—Similarity, also known as Spectral Feature Selection methods (Zhao and Liu 2011), follow the idea of modeling or identifying the local or global data structure using the eigen-system of Laplacian or normalized Laplacian matrices (Luxburg 2007) derived from an object similarity matrix.

Information based methods One of the first methods developed in this category was introduced in Dash et al. (1997), where the authors presented a new filter unsupervised feature selection method called SUD (Sequential backward selection method for Unsupervised Data). This filter method weighs features using a measure of entropy of similarities based on distance, which is defined as the total entropy induced from a similarity matrix W, where the elements of this matrix contain the similarity between pairs of objects in the dataset. The idea is to measure the entropy of the data based on the fact that when every pair of objects is very close or very far, the entropy is low, and it is high if most of the distances between pairs of objects are close to the average distance. Therefore, if the data has low entropy, there are well-defined cluster structures, while there are not when the entropy is high. The relevance of each feature is quantified using a leave-one-out sequential backward strategy jointly with the entropy measure above mentioned. The final result is a feature ranking ordered from the most to the least relevant feature.

In Varshavsky et al. (2006) another information-based UFS method called SVD-Entropy was proposed. The basic idea is to select those features that best represent the data, measuring the entropy of the original data matrix through its singular values (Alter and Alter 2000). This entropy varies between 0 and 1, in such a way that when the entropy is low (close to zero), well-formed clusters are generated, since the spectrum of the data matrix is not uniformly distributed; by contrast, when the entropy is high, the spectrum[2] is uniformly distributed, and the cluster structure is not well-defined. Through a leave-one-out comparison, the contribution of each feature to the entropy (CE) is evaluated, and the features are sorted according to their respective CE values. In this work, three different ways of selecting a final feature subset were presented: simple ranking, forward selection, and backward elimination. The first strategy consists in selecting the first d features from the ranking. The forward selection, on the other hand, starts choosing the first feature according to the highest CE, then the CE values of the remaining set of features are recalculated and the second feature according to the highest CE value is selected, this procedure continues until selecting d features. The backward elimination is similar to the forward selection, with the difference that it begins with the whole set of features and removes that feature with lowest CE value in each iteration until reaching the pre-specified number of features. Two more recent works based on this same idea were introduced in Devakumari and Thangavel (2010) and Banerjee and Pal (2014), where the authors propose to solve some drawbacks of SVD-Entropy. In Devakumari and Thangavel (2010) an Adaptive Floating Search which alleviates the weaknesses of forward/backward selection searches used in SVD-Entropy was proposed. Meanwhile, in Banerjee and Pal (2014), the inability of SVD-Entropy to distinguish features with a constant value was addressed. Furthermore, in the last, an extension to the supervised case was also proposed.

Another unsupervised univariate filter method that ranks features using information theory was proposed in Rao and Sastry (2012). In this method, the aim is weighting each feature using the concept of Representation Entropy (Devijver and Kittler 1982). Representation Entropy is a measure of information compression in a dataset, and it is computed from the entropy of eigenvalues of the covariance matrix of the data. Representation Entropy ranges from 0 to 1, where 1 represents the maximum compression, and 0 is the minimum one. In Rao and Sastry (2012), as in the previous methods, features are scored using a leave-one-out strategy, i.e., the importance of a particular feature in the dataset will depend on the increase in the value of the Entropy (CE value) of the dataset calculated without that particular feature. In this way, it is possible to obtain a feature ranking sorted from the most relevant feature (that one with the highest CE value) to the least one.

Spectral-similarity based methods One of the most referenced and relevant univariate filter UFS methods based on Spectral Feature Selection is Laplacian Score (LS) (He et al. 2005). In Laplacian Score, the importance of a feature is evaluated by its variance and its power of locality preserving (He and Niyogi 2004). This method assigns high weights to those features that most preserve the predefined graph structure (manifold structure) represented by the Laplacian matrix. This idea comes from the observation that two objects are probably related to the same cluster if they are close to each other; in such a way that those features that take similar values for close objects, and dissimilar values for the far away ones are the most relevant. An extension of the Laplacian Score called Laplacian++ was proposed in Padungweang et al. (2009), where the idea is to evaluate the features based on the global topology instead of the local topology.

[2] The set composed by the square of the singular values of the data matrix.

Another univariate filter method in this category is SPEC (SPECtrum decomposition) (Zhao and Liu 2007). SPEC evaluates the relevance of a feature by its consistency with the structure of the graph induced from the similarities among objects. This method consists of three steps: (1) building the object similarity matrix W as well as its graph representation. (2) evaluating features using the eigensystem of the graph by measuring the consistency between each feature and those nontrivial eigenvectors of the Laplacian matrix. And (3), ranking features in descending order in term of their feature relevance (consistency). According to Zhao and Liu (2011), SPEC is a generalization of Laplacian Score.

Finally, a recent univariate unsupervised spectral feature selection method developed to be applied over mixed data (De Leon and Chough 2013) called USFSM (Unsupervised Spectral Feature Selection Method for mixed data) was introduced in Solorio-Fernández et al. (2017). USFSM assess features by analyzing the changes in the spectrum distribution (spectral gaps) of the first non-trivial eigenvalues of the Normalized Laplacian matrix when each feature is excluded from the whole set of features separately. Features are sorted in descending order according to their respective spectral gaps values.

2.1.2 Multivariate filter methods

Multivariate filter methods can be divided into three main groups: Statistical/Information, Bio-inspired, and Spectral/Sparse Learning based methods. The former, as its name suggests, includes UFS methods that perform the selection using statistical and/or information theory measures such as variance-covariance, linear correlation, entropy, mutual information, among others. The second group, on the other hand, includes UFS methods that use stochastic search strategies based on the swarm intelligence paradigm (Beni and Wang 1993; Dorigo and Gambardella 1997) for finding a good subset of features, which satisfies some criterion of quality. Finally, the third group includes those UFS methods based on Spectral Analysis (Zhao and Liu 2011) or on a combination of Spectral Analysis and Sparse Learning (El Ghaoui et al. 2011). It is noteworthy that some authors (Chandrashekar and Sahin 2014; Ang et al. 2016) often call these last methods as *embedded* because feature selection is achieved as part of the learning process, commonly through the optimization of a constrained regression model. However, in this study, we prefer to categorize them as filter multivariate, since in addition to jointly evaluate features, the primary objective is to perform feature selection (or ranking) rather than finding the cluster labels. Moreover, we think that embedded methods could be considered as a sub-category inside the main approaches (i.e., filter, wrapper, and hybrid), not hindering the possibility of having embedded methods in the three approaches.

Statistical/information based methods One of the most representative and referenced works in this category is FSFS (Feature Selection using Feature Similarity) (Mitra et al. 2002). In this work, the authors introduced a statistical measure of dependency/similarity to reduce feature redundancy; this measure called Maximal Information Compression Index (MICI) is based on the variance-covariance between features. The idea of this method is partitioning the original set of features into clusters, such that those features in the same cluster are highly similar, while those in different clusters are dissimilar. Feature clustering is done iteratively based on the kNN principle as follows: In each iteration, FSFS computes the k-nearest features of each feature (using MICI). Then, the feature with the most compact subset of k-nearest features (determined by the distance to its farthest feature among the k-nearest) is selected, and its k nearest features are discarded. This procedure is repeated for the remaining features until all of them are either picked or discarded. Following a similar idea, in Li et al. (2007) a hierarchical method called Mitra's + AIF that removes both redundant

and irrelevant features was proposed. This method uses the algorithm developed in Mitra et al. (2002) to remove redundant features. Subsequently, an exponential entropy measure is used to sort the features according to their relevance. Afterward, from the feature ranking obtained in the previous step, a relevant-non-redundant feature subset is selected using the fuzzy evaluation index FFEI (Pal et al. 2000) in combination with a forward selection search.

Other two multivariate filter methods based on statistical measures were proposed in Haindl et al. (2006) and Ferreira and Figueiredo (2012) respectively. In Haindl et al. (2006), the idea is to evaluate all mutual correlations for all feature pairs. Then, the feature with the largest average mutual correlation with all other features is removed, and the process is repeated for the remaining features until a number of features, previously specified by the user, is reached. Meanwhile, in Ferreira and Figueiredo (2012), a filter supervised/unsupervised feature selection method called RRFS (Relevance Redundancy Feature Selection), which selects features in two steps was proposed. In this method, first, the features are sorted according to a relevance measure (variance for the unsupervised version and the Fisher's Ratio or mutual information for the supervised one). Then, in the second step, following the order generated in the previous step, the features are evaluated using a feature similarity measure to quantify the redundancy between them. Afterward, the first p features with the lowest redundancy are selected.

Following the idea of using statistical measures for feature selection, in Talavera (2000) a multivariate filter method based on a dependency measure was introduced. This method, unlike the previous ones, proposes that in the absence of classes, the relevant features are those that are highly correlated with others; and those features having low correlation with other features are not likely to play an important role in the clustering process (irrelevant features). This conjecture is based on the observation that cohesive and distinct clusters tend to capture feature inter-correlations (Fisher 1987). Therefore, the idea is to evaluate each individual feature f_i through the dependency measure above mentioned. Afterward, the p features with the highest dependency are selected.

Another multivariate statistical-based filter method was introduced in Yen et al. (2010). In this work, the objective is to remove redundant features using the concept of minimization of the feature dependency. The idea is to find independent features (relevant) by choosing a set of coefficients such that the linear dependency of features (expressed by the error vector E) could be close to zero. At each iteration, the feature with the largest absolute coefficients (that one with the smallest $||E||^2$) is removed, and the effect of its removal is updated. This process is iterated until all the remaining error vectors E are smaller than a threshold fixed by the user. Another statistical-based method with a similar idea called MPMR (feature selection based on Maximum Projection and Minimum Redundancy) was proposed in Wang et al. (2015a). In this work, a new criterion called maximum projection and minimum redundancy feature selection was introduced. The idea is to select a feature subset such that all original features are projected into a feature subspace (applying a linear transformation) with minimum reconstruction error. Moreover, in this work, with the aim of maintaining low redundancy, a term for quantifying the redundancy among features (redundancy rate using the Pearson correlation coefficient) was added.

Finally in Dash et al. (2002) a multivariate information-based method similar to Dash et al. (1997) was introduced. In this method, as in Dash et al. (1997), the basic idea is to select features using a measure of entropy of similarities based on distance. The main difference between (Dash et al. 1997) and (Dash et al. 2002) is that in Dash et al. (2002) some weighing parameters for the entropy measure were added, and the entropy measure was reformulated as an exponential function instead of a logarithmic function. Additionally, the authors select a subset of features using a forward selection search.

Bio-inspired methods Recently, several bio-inspired unsupervised feature selection methods based on the swarm intelligence paradigm (Beni and Wang 1993; Dorigo and Gambardella 1997) have been proposed. In Tabakhi et al. (2014), one of the first methods based on this idea called UFSACO (Unsupervised Feature Selection based on Ant Colony Optimization) was introduced. The main objective is to select feature subsets with low similarity among features (low redundancy). In this work, the search space is represented as a complete undirected graph; where the nodes represent the features and the weights of the edges represent the similarities between features. This similarity is computed using the cosine similarity function. The authors follow the idea that if two features are similar, then these features are redundant. Each node in the graph has a *desirability* value called pheromone, which is updated by agents (ants) in function of its current value, a pre-specified decay rate, and the number of times that a given feature has been selected by an agent. The agents traverse the graph iteratively preferring high pheromone values and low similarities until a pre-specified stop criterion (number of iterations) is reached. Finally, those features with the highest pheromone value are selected. Thus, it is expected to pick feature subsets with low redundancy. Other later methods based on the same idea are MGSACO (Microarray Gene Selection based on Ant Colony Optimization) (Tabakhi et al. 2015), RR-FSACO (Relevance-Redundancy Feature Selection based on ACO) (Tabakhi and Moradi 2015), and UPFS (Unsupervised Probabilistic Feature Selection using ant colony optimization) (Dadaneh et al. 2016). In both MGSACO and RR-FSACO, in addition to quantifying the feature redundancy as in the previous method, they also measure the relevance of each feature through its variance (Theodoridis and Koutroumbas 2008b). Therefore, the main objective of all these methods is to select features that minimize redundancy and at the same time maximize relevance. Meanwhile, UPFS, the idea is to pick non-redundant features, but using the Pearson's correlation instead of the cosine similarity.

Spectral/sparse learning methods Some multivariate methods based on Spectral Analysis derived from the SPEC and the Laplacian Score were introduced in Garcia-Garcia and Santos-Rodriguez (2009), Liu et al. (2009b), Niijima and Okuno (2009). In Garcia-Garcia and Santos-Rodriguez (2009), a feature selection method called mR-SP (minimum-Redundancy SPectral feature selection) that combines the SPEC ranking and the minimum redundancy optimality criterion (Peng et al. 2005) was proposed. The basic idea of this method is to add a way for controlling the feature redundancy in SPEC, by introducing an evaluation measure for quantifying the similarity of each pair of features through a modified cosine similarity function. While in Liu et al. (2009b) a method that combines the Laplacian Score with the distance entropy introduced in Dash et al. (2002) was developed. This method selects a feature subset (using the entropy measure) based on the ranking produced by the Laplacian Score. Likewise, in Niijima and Okuno (2009) a method called LLDA-RFE (Laplacian Linear Discriminant Analysis-based Recursive Feature Elimination) was proposed. This method extends the Linear Discriminant Analysis (LDA) (Fukunaga 1990) to the unsupervised case using the similarities among objects; this extension is called LLDA. The idea is to recursively remove features with the smallest absolute values of the discriminant vectors of the LLDA to identify features that potentially reveals clusters in the samples. According to the authors, LLDA-RFE is closely related to Laplacian Score; the main difference is that LLDA-RFE is a multivariate method, which allows selecting features that in combination contribute to discriminate.

Other multivariate feature selection methods that have received attention in the last years, due to their good performance and interpretability (Li et al. 2016), are those based on Spectral Analysis combined with Sparse Learning (El Ghaoui et al. 2011). Sparse Learning refers to those methods that seek a trade-off between some goodness-of-fit measure and the sparsity

(El Ghaoui et al. 2011) of the results. Examples of earlier methods based on this idea are: MCFS (Cai et al. 2010), MRSF (Zheng et al. 2010), UDFS (Yang et al. 2011b) NDFS (Li et al. 2012), JELSR (Hou et al. 2011, 2014), SPFS (Zhao et al. 2013), CGSSL (Li et al. 2014b), RUFS (Qian and Zhai 2013), and RSFS (Shi et al. 2015).

MCFS (Cai et al. 2010) and MRSF (Zheng et al. 2010) were among the earliest unsupervised multivariate spectral/sparse learning feature selection methods. MCFS (Multi-Cluster Feature Selection) consists of three steps: (1) spectral analysis, (2) sparse coefficient learning, and (3) feature selection. In the first step, spectral analysis (Luxburg 2007) is applied on the dataset to detect the cluster structure of the data. Then, in the second step, since the embedding clustering structure of the data is known, through of the first k eigenvectors of the Laplacian matrix, MCFS measures the importance of the features by a regression model with a l_1-norm regularization (Donoho and Tsaig 2008). Finally, in the third step, after solving the regression problem, MCFS selects d features based on the highest absolute values of the coefficients obtained through the regression problem. On the other hand, MRSF (Minimize the feature Redundancy for Spectral Feature selection) evaluates the features all together in order to eliminate redundant features. The idea is to formulate the feature selection problem as a multi-output regression problem (Friedman et al. 2001), and the selection is performed by enforcing the sparsity applying the norm $l_{2,1}$ (Argyriou et al. 2008) instead of the l_1-norm. Moreover, in this work, an efficient algorithm based on the Nesterov's method (Liu et al. 2009a) for solving the regression problem was also proposed. The final feature subset is selected based on the values of a weighted W matrix.

Following a similar idea to MRFS, UDFS (Yang et al. 2011b) (Unsupervised Discriminative Feature Selection algorithm) performs feature selection by simultaneously exploiting discriminative information contained in the scatter matrices and feature correlations. This method proposes to address the feature selection problem taking into account the trace criterion (Fukunaga 1990) into the regression problem. Furthermore, UDFS adds some additional constraints to the regression problem and proposes an efficient algorithm to optimize it. UDFS ranks each feature according to the corresponding weight value in descending order, and the top-ranked features are selected. Another method that shares many common features with MRSF is JELSR (Joint Embedding Learning and Sparse Regression) (Hou et al. 2011). JELSR works with the same objective function as MRSF, and it only differs in the construction of the Laplacian graph, since in this work, locally linear approximation weight (Roweis and Saul 2000) is used to measure local similarity for building the Laplacian graph. A later generalization of JELSR was introduced in Hou et al. (2014), where instead of using the Laplacian graph to characterize the structure of high dimensional data and then apply regression, a unify embedding learning and sparse regression framework was proposed. Furthermore, in this work, a unified perspective for understanding and comparing many popular unsupervised feature selection methods was presented. A recent work related to JELSR is USFS (Wang et al. 2016) (Unsupervised Spectral Feature Selection with l_1-norm graph), where the idea is to use spectral clustering and a l_1-norm graph to select discriminative features. The main difference between USFS and JELSR is the way of building the Laplacian graph; JELSR uses locally linear approximation weights to construct the graph, while USFS adopts a new l_1-norm graph.

Another method related to the works described above is NDFS (Nonnegative Discriminative Feature Selection) (Li et al. 2012). NDFS like UDFS and MRFS, performs feature selection exploiting the discriminative information and feature correlations in a unified framework. First, NDFS uses Spectral Analysis to learn pseudo class labels (defined as non-negative real values). Then, a regression model with $l_{2,1}$-norm regularization (Argyriou et al. 2008) is formulated and optimized through a special solver also proposed in this work. According to

the authors, the main difference between NDFS and UDFS is that NDFS adds a non-negativity constraint to the regression problem, since removing this constraint NDFS becomes UDFS. The same authors proposed a later modification of NDFS in Li and Tang (2015), where a method called NSCR (Nonnegative Spectral analysis with Constrained Redundancy) was introduced. The main difference regarding NDFS is that NSCR adds a mechanism to explicitly control the redundancy. Following the idea of NDFS in Han et al. (2015), a method called FSLR (Feature subset with Sparsity and Low Redundancy) was proposed. FSLR employs Spectral Analysis to represent the data in a lower dimension and introduces a novel regularization term into the objective function with a non-negative constraint. Additionally, an iterative multiplicative algorithm to efficiently solve the constrained optimization problem was proposed. Another UFS method called CDL-FS (Couple Dictionary Learning Feature Selection) which uses a coupled analysis/synthesis dictionary instead of Spectral Analysis to learn pseudo class labels was proposed in Zhu et al. (2016). The general idea is to use a dictionary learning (Gu et al. 2014) in order to model the cluster structure of the data. Feature selection is achieved by imposing an $l_{2,p}$-norm ($0 < p \leq 1$) regularization of the feature weight matrix on the dictionary learning model.

In Nie et al. (2016) a sparse learning based method called SOGFS (Structured Optimal Graph Feature Selection) which simultaneously performs feature selection and local structure learning, was proposed. SOGFS adaptively learns local manifold structure by introducing a similarity matrix in a sparse optimization model based on $l_{2,1}$-norm minimization on both loss function and regularization (Nie et al. 2010). Features are selected according to the corresponding weights once the proposed model has been optimized. Another sparse learning feature selection method named SPFS (Similarity Preserving Feature Selection) was introduced in Zhao et al. (2013). In this method, the idea is to select the d features that best preserve the similarity of the objects using multiple-output regression (Friedman et al. 2001) with an $l_{2,1}$-norm constraint. Additionally, in this work, the authors show the relationship between the proposed method and many other supervised and unsupervised feature selection methods of the state-of-the-art. The authors show that many existing feature evaluation criteria can be unified under a common formulation, where the relevance of features is quantified by measuring their capability in preserving the pairwise sample similarity specified by a predefined similarity matrix. Likewise, in Li et al. (2014b) another method called CGSSL (Clustering-Guided Sparse Structural Learning) was proposed. This work presents a general method for feature selection which jointly exploits nonnegative spectral analysis and structural learning with sparsity. The idea is to use the cluster indicators (learned with nonnegative spectral clustering) in a linear model to provide label information for the structural learning. Moreover, similar to the previous method, in this work, the authors show the relationships between the introduced method and several feature selection methods, including SPFS, MCFS, UDFS, and NDFS.

In order to address the problem of outliers or noise present in many datasets, in Qian and Zhai (2013) a filter method named RUFS (Robust Unsupervised Feature Selection) was proposed. The objective is to achieve both robust clustering and robust feature selection. Unlike the unsupervised feature selection methods above mentioned such as MCFS, UDFS, and NDFS, RUFS learns the pseudo cluster labels via local learning regularized robust nonnegative matrix factorization (Kong et al. 2011). The idea is to learn the labels while feature selection is performed by means of a robust joint $l_{2,1}$ norms minimization. In this work, the authors also proposed an iterative limited-memory BFGS (Liu and Nocedal 1989) algorithm for solving the optimization problem efficiently, and to make RUFS applicable on real-world applications. Following a similar idea to RUFS, in Du et al. (2017) a method called RUFSM (Robust Unsupervised Feature Selection via Matrix Factorization) was proposed. RUFSM

selects features by performing discriminative feature selection and robust clustering simultaneously using the $l_{2,1}$-norm. The main difference between RUFS and RUFSM is that the latter uses the cluster centers as objective concept rather than the pseudo labels of the data. Another method that addresses the problem of noisy features and outliers is RSFS (Robust Spectral learning framework for unsupervised Feature Selection) (Shi et al. 2015). RSFS selects features by applying a graph embedding step (using kernel regression) to efficiently learn the cluster structure, and sparse spectral regression to handle noise and outliers. The idea is to build the Laplacian graph taking into account a weight assigned to each object by local kernel regression and develop an efficient iterative algorithm in order to solve the optimization problem proposed.

In recent years, some works developed under Sparse Learning/Spectral analysis category but under a new perspective called *self-representation* of features, have been proposed. The assumption behind these methods is that each feature can be well approximated by a linear combination of relevant features and a coefficient matrix with sparsity constraints (which can be used as feature weights). RSR (Zhu et al. 2015) (Regularized Self-Representation model for unsupervised feature selection) was the first one on exploiting this idea. In this work, the authors argue that if a feature is important, then it will participate in the representation of most of the other features. The feature selection is done by the minimization of the self-representation error using the $l_{2,1}$-norm for the characterization of residuals, and the most representative features (those with high feature weights) are selected. In Zhu et al. (2017) an extended version of RSR was proposed, where the authors use the $l_{2,p}$-norm regularization instead of $l_{2,1}$-norm to select features with emphasis on small values for p ($0 \leq p < 1$). Another method related to RSR is GRNSR (Graph Regularized Non-negative Self Representation) (Yi et al. 2016). Like RSR, GRNSR exploits the self-representation capability of the features, but with the difference that GRNSR also takes into account the geometrical structure of the data using a neighborhood weighted graph (low-rank representation graph). In GRNSR each feature is first represented by all other features through a non-negative linear combination. Then, a similarity matrix is constructed to uncover the local structure information of the objects and a Nonnegative Least Squares (NNLS) problem is formulated and considered as a new term in the final $l_{2,1}$-norm nonnegative constraint regression problem. Afterward, once the model (regression problem) has been optimized, the top d ranked features with the highest weights are selected.

Other more recent methods also developed under *self-representation* perspective are SPNFSR (Zhou et al. 2017), LRSL (Wang and Wang 2017), DSRMR (Tang et al. 2018a), $l_{2,1}$-UFS (Tang et al. 2018b) and the proposed introduced in Lu et al. (2018). SPNFSR (Structure-Preserving Non-negative Feature Self-Representation), $l_{2,1}$-UFS ($l_{2,1}$ based graph regularized UFS method) and DSRMR (Dual Self-Representation and Manifold Regularization) take into account both the self-representation and the structure-preserving ability of features by optimizing a model based on the $l_{2,1}$-norm. The general idea of these methods is to optimize a model (objective function) take into account three aspects: (1) the self-representation of features using the $l_{2,1}$ norm. (2) the local manifold geometrical structure of the original data using a graph-based norm regularization term. And 3) a regularization term W to reflect the importance of each feature. The optimization problem is solved through an efficient iterative algorithm. At the final stage, each feature is sorted according to the corresponding W values in descending order and the top p ranked features are selected. For its part LRSL (Low-rank approximation and structure learning for unsupervised feature selection), unlike the previous methods, uses the Frobenius norm instead of $l_{2,1}$-norm. Finally, the method introduced in Lu et al. (2018) proposes an objective function for modeling the feature selection problem through a linear combination of all the features in the original feature space and considering

the local manifold structure of the data using an object similarity matrix. Then, once the model has converged, features are ordered according to the corresponding weights and the top p ranked features are selected.

Recently, some works that use Locally Linear Embedding (LLE) and non-convex sparse regularizers functions in sparse learning models have been proposed. In Luo et al. (2018), a novel unsupervised feature selection method that uses LLE (Roweis and Saul 2000) to model the manifold structure of the data was proposed. The idea is to characterize the intrinsic local geometric through an LLE graph-based instead of the typical pairwise similarity matrix jointly with a structure regularization term. For each feature, a feature-level reconstruction score based on the LLE graph is defined, and the final feature subset is selected according to this score. On the other hand, in Shi et al. (2018) a non-convex sparse learning model was proposed. The idea is to perform feature selection through an orthogonal-nonnegative constraint sparse regularized model using a new norm named $\ell_{2,1-2}$-norm defined as the difference of the $l_{2,1}$ and the Frobenius norm. To solve the model efficiently, an iterative algorithm based on the Alternating Direction Method of Multipliers (ADMM) (Boyd et al. 2011) was also proposed.

2.2 Wrapper approach

UFS methods based on the wrapper approach can be divided into three broad categories according to the feature search strategy: sequential, bio-inspired, and iterative. In the former, features are added or removed sequentially. Methods based on sequential search are easy to implement and fast. On the other hand, bio-inspired methods try to incorporate randomness into the search process, aiming to escape from local optima. Finally, iterative methods address the unsupervised feature selection problem by casting it as an estimation problem and thus avoiding a combinatorial search.

2.2.1 Sequential methods

One of the most outstanding methods in this category was introduced in Dy and Brodley (2004). In this work, two feature selection criteria were evaluated: the criterion of Maximum Likelihood (ML) and the scatter separability criterion (trace criterion TR) (Fukunaga 1990). This method searches through the space of feature subsets, evaluating each candidate subset as follows: First, Expectation Maximization (EM) (Dempster et al. 1977) or k-means (MacQueen 1967) clustering algorithms are applied on the data described by each candidate subset. Then, the obtained clusters are evaluated with the ML or TR criteria. The method uses a forward selection search for generating subsets of features that will be evaluated as described above. The method ends when the change in the value of the used criterion is smaller than a given threshold.

In Breaban and Luchian (2011), a method that uses a new optimization criterion for, respectively, minimizing and maximizing the intra-cluster and inter-cluster inertias was proposed. The authors propose a function, unbiased w.r.t. the number of clusters and features, based minimization-maximization of the variance of scatter matrices obtained from the clusters built by the k-means clustering algorithm. This function assigns a ranking score to each partition that may be defined in the search space of all possible subsets of features and number of clusters. The criterion proposed in this method provides both a ranking of relevant features and an optimal partition.

A UFS method that uses a conceptual clustering algorithm for feature selection was proposed in Devaney and Ram (1997). In this work, the authors developed an unsupervised feature selection method based on a measure called *category utility*, which is used to measure the quality of the clusters found by the COBWEB hierarchical clustering algorithm (Fisher 1987). This method generates subsets of features with two search strategies: forward selection and backward elimination. Feature selection is performed running the COBWEB algorithm using the subset of features generated by the search strategy and evaluating the *category utility* for this feature subset. The process ends when no higher category utility score can be obtained in the backward or forward selection.

Finally, in Hruschka and Covoes (2005), a method for feature selection called SS-SFS (Simplified Silhouette Sequential Forward Selection) was proposed. This method selects a feature subset that provides the best quality according to the simplified silhouette criterion. In this method, a forward selection search is used for generating subsets of features. Each feature subset is used to cluster the data using the k-means clustering algorithm, and the quality of the feature subset is evaluated through the quality of the clusters measured with the simplified silhouette criterion. The feature subset that produces the best value of this criterion in the forward selection is selected.

2.2.2 Bio-inspired methods

A representative UFS method in this category was introduced in Kim et al. (2002), where an evolutionary local selection algorithm (ELSA) was proposed to search feature subsets as well as the number of clusters based on the k-means and Gaussian Mixture clustering algorithms. Each solution provided by the clustering algorithms is associated with a vector whose elements represent the quality of the evaluation criteria, which are based on the cohesion of the clusters, inter-class separation, and maximum likelihood. Those features that optimize the objective functions in the evaluation stage are selected.

Another method, also based on an evolutionary algorithm, was introduced in Dutta et al. (2014). In this work, feature selection is performed while the data are clustered using a multi-objective genetic algorithm (MOGA). This method proposes a multi-objective fitness function that minimizes the intra-cluster distance (uniformity) and maximizes the inter-cluster distance (separation). Each chromosome represents a solution, which is composed by a set of k cluster centroids (cluster center for continuous features and cluster mode for categorical features) described by a subset of features. The number of features used for each centroid in each chromosome is randomly generated, and the cluster centers and cluster modes of chromosomes in the initial population are created by generating random numbers, and feature values from the same feature domain, respectively. Then, for reassigning cluster centroids, MOGA uses the k-prototypes clustering algorithm (Huang 1997, 1998) which obtains its inputs from the initial population generated in the previous step. Afterward, the crossover, mutation, and substitution operators are applied, and the process is repeated until a pre-specified stop criterion is met. In the final stage, this method returns the feature subset that optimizes the fitness function jointly with the clusters that they produced.

2.2.3 Iterative

An outstanding method in this category was proposed in Law et al. (2004). The method proposes a strategy to cluster data and to perform feature selection simultaneously using the EM (Dempster et al. 1977) clustering algorithm. The idea is to estimate a set of weights (real

values in $[0-1]$) called "*feature saliences*" (one for each feature) to quantify the relevance of each feature. This estimation is carried out by a modified EM algorithm derived for the task. The method returns the parameters of the density functions that model the components (clusters), as well as the set of *feature saliences* values. Then, the user can consider those *feature saliencies* that best discriminate between different components (those with the highest values). Similar to the previous method, in Roth and Lange (2004) the authors perform feature selection and clustering simultaneously using a Gaussian mixture model (Figueiredo and Jain 2002). In this method, the idea is to optimize the Gaussian mixture model via the EM clustering algorithm, where the Maximization-step of this algorithm was reformulated as a l_1-constraint LASSO problem (Tibshirani 1996; Osborne et al. 2000). The method returns the clusters as well as the coefficients of the model; the coefficients indicate the relevance of each feature.

In more recent years, wrapper methods that use clustering algorithms for initialization or optimization of Sparse Learning models have been proposed, such is the case of the methods introduced in Zeng and Cheung (2011), Wang et al. (2015b), Guo et al. (2017), and Guo and Zhu (2018). In Zeng and Cheung (2011) a wrapper method called LLC-fs (Local Learning-based Clustering algorithm with feature selection) was proposed. In this method, it is assumed that the cluster indicator value at each point should be estimated by a ridge regression model. The authors propose to use the Local Learning-Based Clustering (LLC) framework (Wu and Schölkopf 2007) to formulate the final ridge regression model. Feature selection is done by introducing a binary feature selection vector τ to the local discriminant function of the model. At the end, after the convergence, the output is the vector τ along with a discretized cluster indicator matrix. In Wang et al. (2015b) a method called EUFS (Embedded Unsupervised Feature Selection), which directly embeds the feature selection in the clustering algorithm via Sparse Learning was proposed. In this work, a not convex sparse regression model using a loss function based on $l_{2,1}$-norm is introduced and optimized through an Alternating Direction Method of Multipliers (Boyd et al. 2011). EUFS uses the k-means clustering algorithm to initialize a pseudo cluster indicator matrix U and a latent feature matrix V (used for indicating feature weights) in the final model. Once the model has converged, the output is a feature ranking sorted according to the final values of the latent feature matrix along with the pseudo clusters indicators. A more recent work based on the same idea as the previous work was in introduced in Guo et al. (2017). This method proposes the same objective function as EUFS and only differs in that the loss function of the final model uses the Frobenius-norm instead of $l_{2,1}$-norm, and the update of U and V is performed iteratively by the k-means clustering algorithm until convergence of the model. Moreover, in Guo and Zhu (2018), the first author of the last work proposed another wrapper method called DGUFS (Dependence Guided Unsupervised Feature Selection), which simultaneously performs feature selection and clustering[3] using a constraint model based on $l_{2,0}$-norm. The model is optimized using a modified algorithm based on the iterative Alternating Direction Method of Multipliers (Boyd et al. 2011).

2.3 Hybrids

In order to take advantage of the filter and wrapper approaches, hybrid methods, in a filter stage, the features are ranked or selected applying a measure based on intrinsic properties of the data. While, in a wrapper stage, certain feature subsets are evaluated for finding the

[3] Clustering can be made using the Constrained Boolean Matrix Factorization (CBMF) algorithm proposed by Li et al. (2014a) or employing eigendecomposition and exhaustive search.

best one, through a specific clustering algorithm. We can distinguish two types of hybrid methods: methods based on ranking and methods non-based on ranking of features. In this section, we described some methods of both types belonging to this approach.

In Dash and Liu (2000) one of the first based on ranking unsupervised hybrid feature selection methods was introduced. This method is based on the entropy measure proposed in Dash et al. (1997) (filter stage), jointly with the internal scatter separability criterion (Dy and Brodley 2004) (wrapper stage). In the filter stage, each feature, one by one, is removed from the whole set of features, and the entropy generated in the dataset after the elimination of the feature is computed. This produces a sorted list of features according to the degree of disorder that each feature generates when it was removed from the whole set of features. Once all features have been sorted, in the wrapper stage, a forward selection search is applied jointly with the k-means clustering algorithm in order to build clusters which are evaluated using the scatter separability criterion. This method selects the feature subset that reaches the highest value for the separability criterion.

Another hybrid method also based on feature ranking was proposed in Li et al. (2006). In this method, the authors combine an exponential entropy measure with the fuzzy evaluation index FFEI (Pal et al. 2000) for feature ranking and feature subset selection, respectively. The method employs sequential search considering subsets of features based on the generated ranking and using the fuzzy evaluation index as quality measure. In the wrapper stage, with the purpose of selecting even a smaller feature subset, the fuzzy-c-means algorithm and the scatter separability criterion (Dy and Brodley 2004) are used to select what the authors called a "compact" subset of features.

A more recent hybrid based on ranking unsupervised feature selection method was proposed in Solorio-Fernández et al. (2016). In this method, the authors combine spectral feature selection and the Calinski-Harabasz index (Calinski and Harabasz 1974) for selecting a relevant feature subset. The feature selection is divided into two stages: (1) Feature ranking and, (2) feature subset selection. In the first stage, the idea is to identify those features that preserve the data structure computing for each feature the Laplacian Score (He et al. 2005); this produces a feature ranking. After, in the second stage, taking advantage of the ranking generated in the previous stage and using forward or backward selection search, feature subsets are evaluated through a modified internal evaluation index called WNCH (Weighted Normalized Calinski-Harabasz index). The feature subset with the highest WNCH value is selected.

On the other hand, in Hruschka et al. (2005) a hybrid UFS method non-based on ranking called BFK that combines k-means and a Bayesian filter was introduced. This method, unlike all the above mentioned hybrid methods, begins with the wrapper stage, by running the k-means clustering algorithm on the dataset with a range of clusters specified by the user. The clusters are evaluated with the simplified silhouette criterion and the one with the highest value is selected. Subsequently, in the filter stage, using the concept of Markov blanket, a feature subset is selected through a Bayesian network, where each cluster represents a class, the nodes represent features, and the edges represent relationships between features.

Another hybrid method non-based on ranking that removes both irrelevant and redundant features was introduced in Kim and Gao (2006). This method performs feature selection in two steps: In the first step, a subset of features is founded by applying the least-square estimation (LSE)-based evaluation (Mao 2005). The second step works only with those features identified in the first step, and by using a Sequential Forward Selection search the best feature subset that maximizes the clustering performance (using a modified version of the EM clustering algorithm) is found.

Finally, It is worth noting that, in the literature, some hybrid unsupervised feature selection methods like (Jashki et al. 2009; Hu et al. 2009; Yang et al. 2011a; Yu 2011) designed specifically for handling data in specific domains also have been proposed. Likewise, there are other works such as those proposed in Hruschka et al. (2007), Luo and Xiong (2009) and Dash and Ong (2011), which solve the problem from another different perspective; performing feature selection assuming that a set of clusters can be modeled as being a set of different classes, where they can apply traditional supervised feature selection methods on data.

3 Analysis and discussion

In the previous section, Unsupervised Feature Selection methods were categorized and reviewed according to their approach, type, and subtype. In this section, some overall aspects, advantages, and disadvantages of the UFS methods described in Sect. 2 are discussed. Furthermore, in this section, an experimental evaluation of the most relevant and recent UFS methods of each category is carried out.

In Table 1, we summarize the general advantages and disadvantages of UFS methods belonging to the filter, wrapper, hybrid approaches, and in Table 2, we show the advantages and disadvantages of the described UFS methods regarding their type, subtype, and approach; in concordance to the taxonomy shown in Fig. 1. Moreover, in order to give more details about the UFS methods analyzed in this review, in Table 3, we show a summary of these methods. In this table, the reference, approach, type of method, as well as the datasets,[4] classifiers/clustering algorithms, and the validation measures used to assess the quality of the selection, are shown.

As we can see in Tables 1 and 2, in general, there is not a better UFS approach or method for all kind of data and domain, every approach has its own pros and cons. Nevertheless, from our literature study and from Tables 1, 2 and 3 we can highlight some important general characteristics of the different methods belonging to the different approaches and types.

In Table 3, we can see that there are only a few wrapper methods for Unsupervised Feature Selection, in contrast to filter methods. This is mainly because wrappers become less useful for high dimensionality problems, which makes them seldom used in practice. On the other hand, hybrid methods are preferred to wrapper ones, given their compromise between efficiency and quality of the selected feature subsets. However, there are also few hybrid methods for unsupervised feature selection reported in the literature. Conversely, the filter approach has received more attention. This is understandable given the technological advancement in the last years, and the vast amount of unlabeled data generated across many scientific disciplines, such as text mining, genomic analysis, social media, and intrusion detection, to name a few, where fast and scalable methods are needed. Unsupervised feature selection methods under the filter approach rely on general characteristics of data and evaluate features without involving any clustering algorithm; therefore, they do not have a bias to specific learning models. Besides, filter methods are easy to design, easy to be understood by other researchers, and they are usually very fast (Zhao 2010), which makes them attractive for high-dimensional data. Moreover, as we can see in the taxonomy of Fig. 1, there is an inclination to the development of filter methods based on Spectral Feature Selection and Sparse Learning. This last is mainly because these methods besides being fast, obtain good results in terms of the quality of the selected features.

[4] The number in parentheses denotes the number of datasets used for validation.

Table 1 General advantages and disadvantages of UFS methods regarding their approach

Approach	Advantages	Disadvantages
Filter	Fast	Ignores interaction with clustering algorithms
	Scalable	
	Independent of the clustering algorithm	
	Parallelizable	
Wrapper	Interact with the clustering algorithm to be used	Risk of overfitting
	Can model feature dependencies	High computational cost
		The selection is specific for the used clustering algorithm
Hybrid	Interact with the clustering algorithm to be used	The selection is specific for the used clustering algorithm
	Less time consuming than wrappers	
	Can model feature dependencies	

Table 2 Advantages and disadvantages of UFS methods regarding the type

Type/subtype of method	Approach	Advantages	Disadvantages
Univariate-information based	Filter	Solid theoretical background	Ignore correlation among features
		Information based measures can model linear and non-linear relationships	
		Information based measures are unbiased regarding the dimensionality of the data	
Univariate-spectral/similarity based	Filter	Solid theoretical background	Ignore correlation among features
		Provide a powerful framework for unsupervised feature selection	
Multivariate-statistical/information based	Filter	Can model feature dependencies	Slower than univariate methods
		Less time consuming than wrapper methods	Less scalable than univariate methods
Multivariate-bio-inspired	Filter	Less prone to local optima	Slower than univariate methods
		Model feature dependencies	High memory requirements
Multivariate-spectral/sparse learning based	Filter	Solid theoretical background	Slower than univariate methods
		Handling redundant features	Less scalable than univariate methods
Sequential	Wrapper	Simple to implement	Risk of overfitting
			Prone to local optima
Bio-inspired	Wrapper	Less prone to local optima	Higher risk of overfitting than sequential based-methods
		Can model feature dependencies	High memory requirements
Iterative	Wrapper	Can model feature dependencies	Risk of overfitting
		Feature selection and clustering are made simultaneously	
Based on ranking	Hybrid	Individual relevant features can be more easily identified and selected from the feature ranking	The filter and wrapper approaches can not be truly integrated with each other, which may lead to lower quality performance
Non-based on ranking	Hybrid	Can exploit other ideas that ranking-based methods cannot, for example, modeling feature dependency in the filter stage	Individual relevant features cannot be easily identified because there is no a ranking of features
			The filter and wrapper approaches cannot be truly integrated with each other, which may lead to lower quality performance

3.1 Criteria for determining relevant features

Unlike supervised and semi-supervised feature selection, Unsupervised Feature Selection is considered a much harder problem due to the difficulty of defining feature relevancy (Dy and Brodley 2004).[5] In this regard, from all UFS methods analyzed in our review, we have been able to identify three main criteria commonly used to determine relevant features. The first one consists in choosing those features that can best preserve the manifold structure of the original data; we can find examples of methods using this criterion in the Univariate and Multivariate Spectral/Sparse Learning-based methods belonging to the filter approach. The second criterion consists in seeking cluster indicators (considered as pseudo labels) through clustering algorithms and then transform the unsupervised feature selection into a supervised context; some examples of this kind of methods can be found in Multivariate Spectral/Sparse Learning-based methods of the filter, wrapper, and hybrid approaches. Finally, there is another criterion based on the analysis of correlation among features (feature dependency), where the objective consists in selecting a feature subset with the highest or lowest correlation among features. Some examples of this last criterion can be found in the Multivariate-Statistical based methods in the filter approach.

3.2 Criteria for determining redundant features

Feature correlation, besides to be used as a criterion for selecting relevant features, it is also used for defining feature redundancy. In general, in the literature of Unsupervised Feature Selection, we have identified two main approaches for quantifying redundancy of a particular subset of features: (1) quantifying redundancy without considering an objective concept, and (2) quantifying redundancy considering an objective concept. In the first case, the objective consists in measuring the degree of dependence, similarity, association or correlation (commonly by pairs) among the features by using statistical or information based measures. Some examples of methods under this approach are Mitra et al. (2002), Haindl et al. (2006) , Garcia-Garcia and Santos-Rodriguez (2009), Yen et al. (2010), Zhao et al. (2013), Tabakhi et al. (2014), Tabakhi and Moradi (2015), Tabakhi et al. (2015), Han et al. (2015) and Li and Tang (2015). Meanwhile, in the second case, the aim is to quantify the relationship among features; considering further a specific task or objective concept for which these features could be considered redundant. This is commonly achieved by evaluating features jointly and using sparsity regularization in a constrained regression optimization model. Some examples of UFS methods using this last approach are Zheng et al. (2010), Cai et al. (2010), Zhao and Liu (2011), Hou et al. (2011) and Zhu et al. (2016).

3.3 Performance evaluation and datasets used for assessing UFS methods

Table 3 help us to appreciate that performance evaluation of Unsupervised Feature Selection methods has been done in different ways. Nevertheless, from the analysis made in this review, we can identify three main ways for evaluating the results of the UFS methods:

- Evaluation in terms of the quality of the selected features for a specific supervised/unsupervised classifier. This evaluation is the most widely used, and it has become

[5] Unlike supervised feature selection, which has class labels to guide the search for discriminative features, in UFS, we must define feature relevancy in the form of objective concepts.

the most accepted way for assessing Unsupervised Feature Selection methods. Within this type of evaluation, two standard ways are distinguished.

1. Evaluation using the *classification accuracy* or *error rate* of supervised classifiers such as *k*NN (Fix and Hodges 1951), SVM (Cortes and Vapnik 1995), and Naive Bayes (NB) (Maron 1961; John and Langley 1995), among others. From Table 3, we can see that this evaluation is commonly used by Spectral Feature Selection, Statistic-based, and Bio-inspired methods.
2. Evaluation using the results of clustering algorithms such as *k*-means (MacQueen 1967), EM (Dempster et al. 1977), and COBWEB (Fisher 1987). For assessing the clustering quality, measures like *Normalized Mutual Information (NMI) and Clustering Accuracy (ACC)* are commonly used. Wrapper and hybrid UFS methods, as well as multivariate filter methods based on Sparse Learning and Spectral Feature Selection commonly use clustering algorithms to assess the quality of the selected features.

- Evaluation in terms of the redundancy of the selected features. This evaluation is used by those methods that consider the elimination of redundant features (Mitra et al. 2002; Li et al. 2007; Haindl et al. 2006; Yen et al. 2010; Wang et al. 2015a; Tabakhi et al. 2014; Garcia-Garcia and Santos-Rodriguez 2009; Li et al. 2012; Li and Tang 2015). For this evaluation, the *redundancy rate* (Zheng et al. 2010) and *Representation Entropy* (Devijver and Kittler 1982) are the most used redundancy measures.
- Evaluation in terms of the correctness of the selected features. This evaluation consists in quantifying with a specific measure such as *precision*, *recall* or *F-measure* the amount of relevant features selected by an unsupervised feature selection method. Of course, this is commonly done using synthetic datasets, where the actual relevant features are known a priori, which usually is not possible for real-world datasets.

Regarding the datasets used for evaluation of UFS methods, from Table 3, it can be seen that at least half of the reviewed works use data from the well-known UCI machine learning repository[6] (Lichman 2013), which contains many kinds of datasets with different sizes in both, number of objects and features (including numeric, non-numeric and mixed). The other half of the reviewed works, especially those based on Spectral Analysis and Sparse Learning, mostly use datasets of high dimensionality, such as text, biological data, and images, among others. Likewise, we can observe in Table 3 that the number of datasets used to validate UFS methods ranges from 1 to 42, being seven the average. This indicates, from our point of view, that a more extensive empirical study using a large number of datasets is required to evaluate the actual performance of the UFS methods proposed in the literature.

3.4 Experimental comparison

In order to make a comparison of the performance of the different approaches and categories of the UFS methods reviewed in this paper, we selected 15 of the most relevant an recent UFS methods (taking into account each approach and category) and we evaluated them on 15 datasets from the UCI machine learning repository (detailed information about the selected datasets is summarized in Table 4). The aim is to carry out an empirical comparison about the performance of these methods, regarding the quality of selected features and runtime, over different kind of data (numerical, non-numerical, and mixed data) and perform a further

[6] https://archive.ics.uci.edu/ml/index.php.

Table 3 Summary of unsupervised feature selection methods

Literature	Approach	Type of method	Datasets used for validation	Classifier/clustering algorithm used for validation	Validation measure
Dash et al. (1997)	Filter	Univariate-entropy based	UCI machine learning repository (17)	C4.5 classifier	Error rate using tenfold cross-validation
Varshavsky et al. (2006)	Filter	Univariate-entropy based	Public available datasets (3)	QC	Averages of the Jaccard score of 100 runs of a clustering algorithm
				K-means	
Devakumari and Thangavel (2010)	Filter	Univariate-entropy based	UCI machine learning repository (5)	K-means	Objective function computed by K-means
Rao and Sastry (2012)	Filter	Univariate-entropy based	UCI machine learning repository (4)	C4.5 decision tree classifier	Error in classification and clustering using tenfold cross-validation
				K-means	
Banerjee and Pal (2014)	Filter	Univariate-entropy based	UCI machine learning repository (14)	Fuzzy C-means	Sammon's error
				1NN (nearest neighbor) classifier	Cluster Preserving Index (CPI)
					Error rate
Zhao and Liu (2007)	Filter	Univariate-spectral-similarity	Public available text and face image datasets (3)	1NN classifier	Average of classification accuracy of 10 trials
He et al. (2005)	Filter	Univariate-spectral-similarity	Iris	K-means	NMI and ACC
			PIE face data		
Padungweang et al. (2009)	Filter	Univariate-spectral-similarity	UCI machine learning repository (6)	5NN (nearest neighbor) classifier	Average of classification accuracy using tenfold cross validation
Solorio-Fernández et al. (2017)	Filter	Univariate-spectral-similarity	UCI machine learning repository (20)	K-prototypes clustering algorithm	NMI and ACC
				Naive bayes, 3NN and SVM classifiers	Classification accuracy using tenfold cross validation

Table 3 continued

Literature	Approach	Type of method	Datasets used for validation	Classifier/clustering algorithm used for validation	Validation measure
Mitra et al. (2002)	Filter	Multivariate-statistical based	UCI machine leaning repository (9)	KNN (value of K not specified) and Bayes classifiers	Representation entropy, entropy, fuzzy evaluation index, and class separability
					Classification accuracy
Li et al. (2007)	Filter	Multivariate-statistical/information-based	UCI machine learning repository (3)	3NN classifier	FFEI values
			Synthetic datasets		Classification accuracy using tenfold cross validation
Haindl et al. (2006)	Filter	Multivariate-statistical based	UCI machine learning repository (2)	Naive Bayes classifier	Classification error using tenfold cross validation
			Emotional speech data collections (2)		
Talavera (2000)	Filter	Multivariate-statistical based	UCI machine learning repository (8)	COBWEB clustering algorithm	Average error rate of 5 replications using twofold cross validation
Ferreira and Figueiredo (2012)	Filter	Multivariate-statistical based	UCI machine learning repository (12)	SVM, Naive Bayes and 3NN classifiers	Average error rate for 10 runs with random train/test partitions of the datasets
			NIPS2003 challenge (5)		
			Microarray gene expression datasets (11)		
Yen et al. (2010)	Filter	Multivariate-statistical based	UCI machine learning repository (3)	None	Area under receiver operating characteristic (ROC) curve (AUC) versus the percent of features removed

Table 3 continued

Literature	Approach	Type of method	Datasets used for validation	Classifier/clustering algorithm used for validation	Validation measure
Wang et al. (2015a)	Filter	Multivariate-statistical based	ASU feature selection repository (6)	Not specified	NMI and ACC
					Redundancy rate
Dash et al. (2002)	Filter	Multivariate-entropy based	Iris dataset	None	Correctness of the selected features
			Synthetic datasets		
Tabakhi et al. (2014)	Filter	Multivariate-bio-inspired	UCI machine learning repository (7)	SVM, decision trees, and Naive Bayes classifiers	Average error rate over 5 independent runs with random train/test partitions
			NIPS2003 challenge (2)		
Tabakhi and Moradi (2015)	Filter	Multivariate-bio-inspired	UCI machine learning repository (9)	SVM, decision trees, and Naive Bayes classifiers	Average error rate over 5 independent runs with random train/test partitions
			Bioinformatics Research Group of Universidad Pablo de Olavide (1)		
			NIPS2003 challenge (2)		
Tabakhi et al. (2015)	Filter	Multivariate-bio-inspired	Microarray datasets from Universidad Pablo de Olavide (2)	SVM, Naive Bayes, and decision trees classifiers	Average error rate over 5 independent runs with random train/test partitions
			Gene expression model selector from Vanderbilt University (3)		
Dadaneh et al. (2016)	Filter	Multivariate-bio-inspired	UCI machine learning repository (10)	SVM, Naive Bayes, and 1NN classifiers	Classification accuracy using tenfold cross validation
Niijima and Okuno (2009)	Filter	Multivariate-spectral	Public datasets of cancer microarrays (7)	Nearest mean classifier (NMC)	Average error rate over the 100 runs

Table 3 continued

Literature	Approach	Type of method	Datasets used for validation	Classifier/clustering algorithm used for validation	Validation measure
Garcia-Garcia and Santos-Rodriguez (2009)	Filter	Multivariate-spectral	Gene expression Profiles in human cancers challenge dataset (1)	Spectral clustering method	Clustering error
Liu et al. (2009b)	Filter	Multivariate-spectral	UCI machine learning repository (6)	K-means	Average clustering accuracy from 10 trials
Cai et al. (2010)	Filter	Multivariate-spectral-sparse learning	UCI machine learning repository (4)	K-means	NMI
				1NN classifier	Error rate using leave-one-out cross validation
Zheng et al. (2010)	Filter	Multivariate-spectral-sparse learning	Public high dimensional datasets (6)	SVM classifier	Classification accuracy, Jaccard score
					Redundancy rate
Yang et al. (2011b)	Filter	Multivariate-spectral-sparse learning	Public benchmark datasets (6)	K-means	NMI and ACC
Hou et al. (2011), Hou et al. (2014)	Filter	Multivariate-spectral-sparse learning	Several public datasets, including images, voice and biological data	K-means	NMI and ACC
				KNN classifier (value of K not specified)	Classification accuracy
Wang et al. (2016)	Filter	Multivariate-Spectral-Sparse learning	Public available datasets (5)	K-means	ACC
Li et al. (2012)	Filter	Multivariate-spectral-sparse learning	Public available datasets (8)	K-means	NMI and ACC

Table 3 continued

Literature	Approach	Type of method	Datasets used for validation	Classifier/clustering algorithm used for validation	Validation measure
Li and Tang (2015)	Filter	Multivariate-spectral-sparse learning	Public available image datasets (9)	K-means	NMI and ACC
					Redundancy rate
Han et al. (2015)	Filter	Multivariate-spectral-sparse learning	Public available datasets (5)	K-means	NMI
					Purity
					F1-score
Zhu et al. (2016)	Filter	Multivariate-sparse learning	Public available datasets (6)	K-means	NMI and ACC
Nie et al. (2016)	Filter	Multivariate-sparse learning	Public available datasets (8)	K-means	ACC
Zhao et al. (2013)	Filter	Multivariate-spectral-sparse learning	Public high dimensional datasets (8)	Linear SVM classifier	Classification accuracy
					Jaccard score
					Redundancy rate
Li et al. (2014b)	Filter	Multivariate-spectral-sparse learning	Public benchmark datasets (12)	K-means	NMI and ACC
Qian and Zhai (2013)	Filter	Multivariate-sparse learning	Benchmark real world datasets (6)	K-means	NMI and ACC
Du et al. (2017)	Filter	Multivariate-sparse learning	Public available datasets (6)	K-means	NMI and ACC
Zhu et al. (2015, 2017)	Filter	Multivariate-sparse learning	Synthetic and real-world datasets	K-means	NMI and ACC

Table 3 continued

Literature	Approach	Type of method	Datasets used for validation	Classifier/clustering algorithm used for validation	Validation measure
Shi et al. (2015)	Filter	Multivariate-sparse learning	Public available datasets (6)	K-means	NMI and ACC
Yi et al. (2016)	Filter	Multivariate-sparse learning	Standard face datasets (3)	5NN classifier	Classification accuracy
Zhou et al. (2017)	Filter	Multivariate-sparse learning	Public available benchmark datasets (6)	K-means	NMI and ACC
Wang and Wang (2017)	Filter	Multivariate-sparse learning	Public available datasets (12)	K-means	NMI and ACC
Tang et al. (2018a)	Filter	Multivariate-sparse learning	Public available datasets (10)	K-means	NMI and ACC
Tang et al. (2018b)	Filter	Multivariate-sparse learning	Public available datasets (10)	K-means	NMI and ACC
Lu et al. (2018)	Filter	Multivariate-sparse learning	Public available datasets (6)	K-means	NMI and ACC
Luo et al. (2018)	Filter	Multivariate-sparse learning	Public available datasets (8)	K-means	NMI and ACC
Shi et al. (2018)	Filter	Multivariate-sparse learning	Public available datasets (6)	K-means	NMI and ACC
Dy and Brodley (2004)	Wrapper	Sequential	Synthetic datasets (5)	EM and K-means	Class error rate
			UCI machine learning repository (3)		Bayes error
					Precision and recall
Breaban and Luchian (2011)	Wrapper	Sequential	synthetic datasets (40)	K-means	Adjusted rand index
			UCI machine learning repository (2)		Precision, recall, and F-measure

Table 3 continued

Literature	Approach	Type of method	Datasets used for validation	Classifier/clustering algorithm used for validation	Validation measure
Devaney and Ram (1997)	Wrapper	Sequential	UCI machine learning repository (2)	COBWEB	Accuracy of predicting the class label of the previously unseen testing objects
Hruschka and Covoes (2005)	Wrapper	Sequential	Synthetic datasets (1)	K-means	Class error
			Bioinformatics datasets (6)		
Law et al. (2004)	Wrapper	Iterative	Synthetic datasets (2)	EM	Error rates (using the ground truth labels) on the test data repeated 20 times
			UCI machine learning repository (4)		
Roth and Lange (2004)	Wrapper	Iterative	USPS handwritten digits dataset (1)	EM	Stability of data partitions
			Stirling faces database (1)		
Zeng and Cheung (2011)	Wrapper	Iterative	UCI machine learning repository (4)	None	ACC
			Public available benchmark datasets (6)		
Wang et al. (2015b)	Wrapper	Iterative	Public available benchmark datasets (6)	K-means	ACC and NMI
Guo et al. (2017)	Wrapper	Iterative	Public available benchmark datasets (6)	K-means	ACC
Guo and Zhu (2018)	Wrapper	Iterative	Public available benchmark datasets (6)	K-means	ACC and NMI
Kim et al. (2002)	Wrapper	Bio-inspired	Real datasets (1)	K-means and EM	F-accuracy
			Synthetic datasets (1)		F-within and F-between

Table 3 continued

Literature	Approach	Type of method	Datasets used for validation	Classifier/clustering algorithm used for validation	Validation measure
Dutta et al. (2014)	Wrapper	Bio-inspired	UCI machine learning repository (4)	K-Prototypes	Davies–Bouldin index, C index, and Dunn index
					ACC
Dash and Liu (2000)	Hybrid	Based on ranking	Synthetic datasets (4)	K-means	Feature ranking
			UCI machine learning repository (5)		Impurity
Li et al. (2006)	Hybrid	Based on ranking	Synthetic and real-world datasets from the UCI machine learning repository	3NN	Feature ranking
					Classification accuracy using cross validation
Solorio-Fernández et al. (2016)	Hybrid	Based on ranking	Synthetic datasets (20)	K-means	Jaccard index and global silhouette
			UCI machine learning repository (22)		Retention and Run-time
Hruschka et al. (2005)	Hybrid	Non-based on ranking	Synthetic datasets (1)	K-means	Class error
			UCI machine learning repository (4)		
Kim and Gao (2006)	Hybrid	Non-based on ranking	Synthetic datasets (2)	EM	Classification error using tenfold cross validation
			UCI machine learning repository (2)		

analysis based on the experimental results. Specifically, in our experiments, we compared the following UFS methods:

- Filter
  - *Univariate*: SVD-Entropy (Varshavsky et al. 2006), Laplacian Score (LS) (He et al. 2005), SPEC (Zhao and Liu 2007), and USFSM (Solorio-Fernández et al. 2017).
  - *Multivariate*: FSFS (Mitra et al. 2002), RRFS (Ferreira and Figueiredo 2012), UDFS (Yang et al. 2011b), NDFS (Li et al. 2012), UFSACO (Tabakhi et al. 2014), MGSACO (Tabakhi et al. 2015), and DSRMR (Tang et al. 2018a).
- Wrapper
  LLC-fs (Zeng and Cheung 2011) and DGUFS (Guo and Zhu 2018).
- Hybrid
  Li et al. (2006) and WNCH-BE (Solorio-Fernández et al. 2016).
- All original features are adopted as the baseline in our experiments.

Following the standard ways to assess UFS methods, we evaluate the UFS methods in terms of clustering and classification performance. For evaluating the clustering results, the commonly used clustering performance metrics ACC (Clustering Accuracy) and the NMI (Normalized Mutual Information) were applied over the partitions produced by the k-means clustering algorithm[7] on the selected features by each UFS method on each dataset. On the other hand, for evaluating the UFS methods in terms of classification performance, we used the well-known and broadly used SVM (Cortes and Vapnik 1995) classifier. For the evaluation, we applied stratified fivefold cross-validation, and the final classification performance is reported as the average accuracy over the five folds. For each fold, each UFS method is first applied on the training set (ignoring the class labels) to obtain a feature subset. Then, after training the classifier using the selected features, the respective test sets are used for assessing the classifier through its accuracy. Additionally, we evaluate the runtime spent by each UFS method for performing feature selection.

The SVM classifier and the k-means clustering algorithm used in our experiments were taken from the Weka data mining software tool (Hall et al. 2009), for SVM we used its default parameter values while the parameter k for k-means was set as the number of classes declared for each dataset. Likewise, for the different UFS methods analyzed in our experiments, we used the author's implementation, and the parameter values were fixed according to the recommendation of their respective authors. All experiments were run in Matlab® R2018a with Java 1.8, using a computer with an Intel Core i7-2600 3.40 GHz $\times$ 8 processor with 32 GB DDR4 RAM, running 64-bit Ubuntu 16.04 LTS (GNU/Linux 4.13.0-38 generic) operating system.

In our experiments, for those UFS methods that provide a feature ranking as output, or those that need as an input parameter the number of features to select, we set 40%, 50% and 60% of the ranked features for the first ones, and the same percentage of the whole set of features for the second ones, respectively. The best classification and clustering results in the different percents were reported as the final result for all the feature selection methods. Furthermore, in our experiments, the Friedman test (Friedman 1937) was used to make and evaluate the ranking of all the evaluated methods over all the datasets. It is important to mention that for all datasets, class labels were removed for feature selection and clustering, and for those UFS methods that can only process numerical features, the non-numerical

[7] In order to get more reliable results, we repeat the k-means algorithm ten times with different initial points and report the average clustering quality results.

Table 4 Description of the used datasets taken from the UCI machine learning repository

#	Dataset	No. of objects	No. of features	No. of classes
1	Automovile	205	25	6
2	Breast-cancer	286	9	2
3	Heart-c	303	13	2
4	Heart-statlog	270	13	2
5	Hepatitis	155	19	2
6	Ionosphere	351	34	2
7	Liver-disorders	345	6	2
8	Lung cancer	32	56	3
9	Lymphography	148	18	4
10	Monks-problems-2-train	169	6	2
11	Sonar	208	60	2
12	Soybean	683	35	19
13	Wdbc	569	30	2
14	Wine	178	13	3
15	Zoo	101	17	7

features were transformed into numerical ones by mapping each categorical value into an integer value in the order of appearance of the dataset.

Tables 5, 6, 7 and 8 show the final results regarding classification (see Table 5), clustering (see Tables 6 and 7), and runtime (see Table 8) performance. In Tables 5, 6 and 7 the best method on average for each dataset appears in "bold", and the last row of each table shows the average rank over all tested datasets.

Regarding the evaluation of the UFS methods in terms of supervised classification performance, from Table 5, it can be seen that UFS methods allow obtaining competitive or in some cases better classification performance than using all the features, but with fewer features. In this table, we can see that USFSM and NDFS obtained the best average ranking among those UFS methods in the filter approach; LLC-fs was the best in the wrapper approach, and the method proposed by Li et al. (2006) was the best in the hybrid approach.

On the other hand, regarding the evaluation of the UFS methods in terms of clustering performance, in Tables 6 and 7 we can see that among univariate methods, for both quality measures NMI and ACC, into the filter approach, the best results were obtained by SVD-entropy and LS methods among UFS univariate methods; meanwhile UDFS, NDFS, DSRMR, and UFSACO got the best results among the multivariate ones. Notice that most of above mentioned univariate and multivariate methods got even better results than those obtained when using all the features. The worst results in the filter approach were obtained by the multivariate statistical methods. In this case, in general, the methods in the wrapper and hybrid approaches obtained the worst results.

Regarding the runtime, from Table 8, we can see that the fastest UFS methods were those in the filter approach; LS and SPEC among univariate UFS methods, and FSFS, RRFS among multivariate UFS methods. While LLC-fs and LS-WNCH-BE were the fastest methods in the wrapper and hybrid approaches respectively. It also can be noted that the slowest methods were DSRMR, USFSM, and the hybrid method proposed in Li et al. (2006).

Finally, from the results shown in Tables 5, 6, 7 and 8, we can conclude the following:

Table 5 Classification accuracy of the evaluated UFS methods using SVM

Dataset	Filter											Wrapper		Hybrid		Original
	Univariate				Multivariate											
	SVD-entropy	LS	SPEC	USFSM	FSFS	RRFS	UDFS	NDFS	UFSACO	MGSACO	DSRMR	LLC-fs	DGUFS	Li et al.	LS-WNCH-BE	
Automovile	0.498	0.478	0.478	0.595	0.659	0.668	0.507	0.566	0.678	0.668	0.561	0.517	0.683	**0.693**	0.673	**0.693**
Breast-cancer	0.710	**0.724**	0.703	0.696	0.717	0.692	0.706	0.710	0.713	0.696	0.699	0.703	0.713	0.678	0.685	0.685
Heart-c	0.822	0.818	0.835	**0.845**	0.792	0.802	0.815	0.812	0.785	0.818	0.809	0.802	0.759	0.819	0.815	0.832
Heart-statlog	0.830	0.778	0.826	0.763	0.778	0.826	0.778	0.822	0.833	0.796	0.830	0.804	0.807	0.793	0.726	**0.837**
Hepatitis	0.852	0.852	0.858	0.832	0.839	0.826	0.858	**0.871**	0.839	0.839	0.845	0.845	0.819	0.852	0.845	0.845
Ionosphere	0.866	0.832	0.789	0.889	0.852	0.866	0.869	0.880	0.889	0.866	0.886	0.866	**0.889**	0.874	0.852	0.883
Liver-disorders	**0.580**	**0.580**	**0.580**	**0.580**	**0.580**	**0.580**	**0.580**	**0.580**	**0.580**	**0.580**	**0.580**	**0.580**	**0.580**	**0.580**	**0.580**	**0.580**
Lung-cancer	0.529	0.476	0.467	0.533	0.471	0.505	0.533	0.476	0.471	0.529	0.443	**0.538**	0.433	0.510	0.476	0.500
Lymphography	0.784	0.777	0.831	**0.845**	0.764	0.689	0.804	0.837	0.817	0.845	0.804	**0.845**	0.797	0.838	0.616	0.838
Monks-problems-2_train	**0.621**	**0.621**	0.598	**0.621**	0.610	0.598	**0.621**	0.610	**0.621**	0.598	0.610	0.610	**0.621**	**0.621**	**0.621**	0.604
Sonar	**0.798**	0.789	0.755	0.759	0.760	0.746	0.774	0.774	0.779	0.784	0.765	0.765	0.779	0.702	0.755	0.755
Soybean	0.871	0.880	0.892	0.898	0.898	0.862	0.873	0.868	0.889	0.921	0.896	0.925	0.902	0.912	**0.937**	0.928
Wdbc	0.944	0.952	0.965	0.952	0.958	0.963	0.951	0.967	0.965	0.961	0.956	0.975	0.972	0.949	0.935	**0.977**
Wine	0.955	0.955	0.955	0.949	0.933	0.921	0.926	0.966	0.938	0.944	0.938	0.972	0.938	0.910	0.961	**0.989**
Zoo	0.941	0.940	0.891	0.950	0.871	0.831	0.921	0.921	0.881	0.921	0.920	0.930	0.921	**0.960**	0.960	0.960
Average rank	**7.533**	**8.633**	**9.466**	**7.466**	**10.833**	**11.833**	**8.866**	**7.500**	**7.866**	**8.033**	**9.466**	**7.033**	**8.166**	**8.133**	**9.400**	**5.766**

Table 6 ACC results of the evaluated UFS methods using k-means

Dataset	Filter											Wrapper		Hybrid		Original
	Univariate				Multivariate											
	SVD-entropy	LS	SPEC	USFSM	FSFS	RRFS	UDFS	NDFS	UFSACO	MGSACO	DSRMR	LLC-fs	DGUFS	Li et al.	LS-WNCH-BE	
Automovile	0.398	0.374	0.382	0.399	0.382	0.340	**0.403**	0.365	0.392	0.380	0.366	0.389	0.389	0.401	0.401	0.401
Breast-cancer	0.629	0.595	0.616	0.653	0.603	0.654	0.639	0.564	0.594	**0.655**	0.563	0.605	0.628	0.550	0.609	0.609
Heart-c	0.785	0.799	0.786	0.714	0.683	0.795	0.791	0.779	0.779	0.782	0.736	0.782	0.735	0.697	0.716	**0.809**
Heart-statlog	0.789	0.768	0.753	0.773	0.727	**0.796**	0.739	0.789	0.670	0.727	0.763	0.589	0.681	0.753	0.619	0.796
Hepatitis	0.746	0.725	0.768	0.733	0.726	0.701	0.645	0.645	0.667	0.672	0.761	0.726	0.644	**0.790**	0.675	0.675
Ionosphere	0.707	0.707	0.698	0.698	**0.726**	0.676	0.712	0.712	0.694	**0.726**	0.716	0.715	0.698	0.693	0.707	0.711
Liver-disorders	0.559	0.553	0.559	0.559	0.553	0.556	0.547	0.559	**0.578**	0.545	0.559	0.548	0.551	0.551	0.548	0.548
Lung-cancer	0.563	0.563	0.506	**0.600**	0.556	0.500	0.569	0.531	0.563	0.581	0.575	0.544	0.538	0.550	0.488	0.594
Lymphography	**0.546**	0.462	0.492	0.509	0.486	0.380	0.497	0.482	0.518	0.462	0.466	0.470	0.476	0.462	0.418	0.458
Monks-problems-2_train	0.559	**0.598**	0.546	0.538	0.538	0.568	0.541	0.579	**0.598**	0.579	**0.598**	0.559	0.544	0.530	0.521	0.543
Sonar	0.554	0.575	0.539	0.552	0.587	0.591	0.544	0.544	**0.625**	0.559	0.545	0.550	0.587	0.510	0.549	0.549
Soybean	0.597	0.601	0.616	0.589	0.383	0.414	0.620	**0.669**	0.542	0.529	0.558	0.467	0.378	0.493	0.479	0.472
Wdbc	0.928	0.930	0.940	0.936	0.953	**0.958**	0.924	0.927	0.938	0.924	0.903	0.935	0.914	0.921	0.921	0.928
Wine	0.957	0.957	0.936	0.912	0.916	0.935	0.957	**0.973**	0.943	0.908	0.942	0.933	0.927	0.892	0.946	0.949
Zoo	**0.893**	0.885	0.697	0.667	0.721	0.513	0.895	0.715	0.806	0.776	0.816	0.719	0.703	0.727	0.707	0.707
Average rank	**5.233**	**6.766**	**8.066**	**7.566**	**9.033**	**8.966**	**7.066**	**8.400**	**7.166**	**8.566**	**8.000**	**9.400**	**11.166**	**11.233**	**11.433**	**7.933**

Table 7 NMI results of the evaluated UFS methods using *k*-means

Dataset	Filter											Wrapper		Hybrid		Original
	Univariate				Multivariate											
	SVD-entropy	LS	SPEC	USFSM	FSFS	RRFS	UDFS	NDFS	UFSACO	MGSACO	DSRMR	LLC-fs	DGUFS	Li et al.	LS-WNCH-BE	
Automovile	0.188	0.166	0.181	0.264	0.249	0.183	0.257	0.189	0.245	0.192	0.162	0.180	**0.266**	0.244	0.213	0.244
Breast-cancer	0.024	0.010	0.028	0.015	0.004	0.007	**0.031**	0.006	0.008	0.022	0.004	0.009	0.013	0.003	0.021	0.021
Heart-c	0.246	0.272	0.256	0.176	0.117	0.268	0.279	0.255	0.240	0.247	0.202	0.247	0.175	0.156	0.147	**0.292**
Heart-statlog	0.254	0.220	0.190	0.230	0.172	**0.270**	0.190	0.254	0.097	0.168	0.205	0.024	0.103	0.190	0.266	0.270
Hepatitis	0.126	0.130	0.192	0.157	0.106	0.107	0.072	0.071	0.056	0.100	0.177	0.077	0.094	**0.195**	0.107	0.107
Ionosphere	0.126	0.126	0.108	0.112	**0.183**	0.082	0.131	0.131	0.102	0.141	0.130	0.129	0.095	0.092	0.126	0.128
Liver-disorders	0.001	0.001	0.003	0.000	0.001	0.002	0.001	0.003	**0.010**	0.002	0.001	0.001	0.000	0.000	0.001	0.000
Lung-cancer	**0.282**	0.271	0.176	0.243	0.179	0.147	0.274	0.217	0.214	0.236	0.224	0.203	0.180	0.241	0.221	0.279
Lymphography	0.144	0.122	0.138	0.111	0.148	0.055	0.125	0.117	**0.185**	0.149	0.119	0.131	0.103	0.137	0.042	0.132
Monks-problems-2_train	0.002	**0.017**	0.008	0.003	0.004	0.008	0.012	0.007	**0.017**	0.007	**0.017**	0.003	0.012	0.001	0.001	0.006
Sonar	0.007	0.018	0.010	0.013	0.019	0.023	0.007	0.008	**0.042**	0.012	0.012	0.005	0.022	0.002	0.007	0.007
Soybean	0.711	0.695	0.724	0.700	0.495	0.481	0.707	**0.731**	0.612	0.625	0.663	0.577	0.487	0.607	0.645	0.593
Wdbc	0.625	0.619	0.660	0.641	0.719	**0.753**	0.611	0.605	0.686	0.611	0.538	0.672	0.579	0.581	0.584	0.611
Wine	0.846	0.846	0.783	0.743	0.697	0.800	0.835	**0.889**	0.798	0.706	0.815	0.787	0.760	0.704	0.808	0.835
Zoo	**0.882**	0.881	0.667	0.645	0.748	0.425	0.901	0.738	0.803	0.759	0.817	0.703	0.658	0.773	0.752	0.752
Average rank	**6.500**	**6.500**	**7.533**	**8.800**	**9.366**	**9.233**	**6.066**	**8.000**	**7.533**	**7.733**	**8.533**	**10.800**	**11.133**	**11.100**	**9.800**	**7.366**

Table 8 Runtime of the evaluated UFS methods

Dataset	Filter											Wrapper		Hybrid	
	Univariate				Multivariate										
	SVD-entropy	LS	SPEC	USFSM	FSFS	RRFS	UDFS	NDFS	UFSACO	MGSACO	DSRMR	LLC-fs	DGUFS	Li et al.	LS-WNCH-BE
Automovile	0.042	0.045	0.031	0.604	0.038	0.045	0.072	0.327	0.081	0.070	2.854	0.134	0.892	0.664	0.422
Breast-cancer	0.008	0.007	0.017	0.555	0.003	0.040	0.088	0.298	0.042	0.038	12.925	0.150	0.733	0.557	0.076
Heart-c	0.007	0.006	0.015	0.752	0.005	0.037	0.082	0.236	0.041	0.067	13.884	0.156	0.598	0.744	0.141
Heart-statlog	0.005	0.008	0.016	0.611	0.004	0.032	0.084	0.132	0.044	0.036	9.823	0.127	0.628	0.412	0.202
Hepatitis	0.010	0.005	0.006	0.173	0.006	0.027	0.050	0.129	0.034	0.039	4.639	0.062	0.290	0.342	0.084
Ionosphere	0.035	0.009	0.019	2.828	0.016	0.043	0.121	0.182	0.115	0.113	25.374	0.251	0.808	2.762	1.817
Liver-disorders	0.002	0.007	0.018	0.600	0.002	0.036	0.116	0.239	0.039	0.038	20.834	0.229	0.865	0.195	0.193
Lung-cancer	0.038	0.005	0.003	0.024	0.047	0.047	0.103	0.194	0.371	0.348	0.118	0.070	0.001	0.750	0.124
Lymphography	0.011	0.004	0.005	0.169	0.007	0.025	0.057	0.198	0.046	0.039	3.666	0.079	0.296	0.530	0.122
Monks-problems-2_train	0.002	0.007	0.008	0.087	0.002	0.021	0.047	0.174	0.031	0.026	5.018	0.078	0.350	0.160	0.323
Sonar	0.125	0.007	0.011	1.191	0.039	0.037	0.117	0.118	0.385	0.410	7.297	0.088	0.415	1.611	0.108
Soybean	0.048	0.019	0.120	20.191	0.021	0.092	0.403	0.468	0.165	0.167	451.395	0.716	2.117	5.483	0.857
Wdbc	0.061	0.015	0.049	10.462	0.015	0.059	0.223	0.238	0.101	0.112	220.122	0.456	1.702	2.920	0.415
Wine	0.006	0.005	0.006	0.180	0.004	0.019	0.050	0.087	0.023	0.026	14.511	0.057	0.320	0.452	0.352
Zoo	0.013	0.004	0.003	0.057	0.004	0.038	0.055	0.107	0.043	0.049	2.336	0.056	0.236	0.341	0.107
Average	**0.028**	**0.010**	**0.022**	**2.566**	**0.014**	**0.040**	**0.111**	**0.208**	**0.104**	**0.105**	**52.986**	**0.180**	**0.683**	**1.195**	**0.356**

1. The quality of the features selected by each UFS method depends to a large extent on the learning algorithm and the validation measure used. For example, we can observe that a useful feature subset for SVM might not be as good for k-means and vice versa.
2. The best results in both classification and clustering tasks were obtained by filter multivariate Spectral/Sparse Learning based methods. Conversely, the multivariate statistical based methods generally got the worst results in both classification and clustering tasks. Especially those methods that eliminate redundant features without first considering the elimination of irrelevant ones.
3. The quality of the results of clustering algorithms is better when feature selection is applied, while in tasks of supervised classification it is worse.
4. Filter methods are the fastest, specifically, statistical based methods. However, these filter methods usually provide the worst results in terms of quality.

4 Concluding remarks

Unsupervised Feature Selection methods have drawn interest in various research areas due to their ability to select features in unlabeled data (unsupervised datasets). This paper provides a review of the most relevant and recent UFS methods of the state-of-the-art. Additionally, we have introduced a taxonomy of UFS methods, and we have summarized the advantages and disadvantages of the general lines in which we have categorized the methods analyzed in this review. Moreover, an experimental comparison among the most representative methods of each approach was also presented.

In general, we observe that many researchers have devoted huge and fruitful efforts in developing methods under the filter approach. This because, commonly, filter methods have lower computational cost than wrappers and hybrids, which makes them suitable for high dimensionality datasets. Moreover, recent developments indicate that filter methods based on Spectral Feature Selection (Zhao and Liu 2011) and Sparse Learning (El Ghaoui et al. 2011) have increasingly been developed, particularly for their application on image, text, and biological data.

Regarding the main challenges and open problems in Unsupervised Feature Selection, we can mention the following:

- Based on the literature review, it was observed that most of the unsupervised feature selection methods (filter, wrapper or hybrids) require the specification of hyper-parameters such as the number of features, number of clusters or other parameters inherent to the feature selection technique used by each method. However, there is no such knowledge in practice, and most of the time it is impossible to know the best parameters values for each dataset. Therefore, the automatic selection of the best parameter values is an open problem.
- Scalability is another important challenge in feature selection, since many applications involve very large collections of objects and/or features. In the last few years, datasets with millions of features have been produced, and according to Bolón-Canedo et al. (2015) there is evidence that this number will increase, given the rapid advancements in computing and information technologies. Therefore, scalable methods are needed, since existing ones can not deal with a huge number of features.
- Stability of feature selection methods is the sensitivity of the selection toward data perturbation (Alelyani et al. 2011). According to Li et al. (2016), studying stability for Unsupervised Feature Selection is much more difficult than supervised methods because,

in Unsupervised Feature Selection, we do not have enough prior knowledge about the cluster structure of the data. Although some recent efforts for analyzing the stability of feature selection methods in the unsupervised contexts have been done (Alelyani 2013), there is a lot of work to do in this direction.

– Another important challenge in Unsupervised Feature Selection is regarding how to select relevant features in problems where data are described simultaneously by both numerical an non-numerical features (mixed data). Mixed data is very common, and it appears in many real-world problems. For example, in biomedical and health-care applications (Daniels and Normand 2005), socioeconomics and business (De Leon and Chough 2013), software cost estimations (Liu et al. 2013), etc. However, as we have seen in this review, most of the current methods (except those proposed in Solorio-Fernández et al. (2017) and Dutta et al. (2014)) have been designed only for numerical data. Therefore, there is a room for developing new Unsupervised Feature Selection methods for mixed data.

Acknowledgements The first author gratefully acknowledges to the National Council of Science and Technology of Mexico (CONACyT) for his Ph.D. fellowship, through the scholarship 224490.

References

Agrawal S, Agrawal J (2015) Survey on anomaly detection using data mining techniques. Procedia Comput Sci 60(1):708–713. https://doi.org/10.1016/j.procs.2015.08.220

Ahmed M, Mahmood AN, Islam MR (2016) A survey of anomaly detection techniques in financial domain. Future Genera Comput Syst 55:278–288. https://doi.org/10.1016/j.future.2015.01.001

Alelyani S (2013) On feature selection stability: a data perspective. Arizona State University, Tempe

Alelyani S, Liu H, Wang L (2011) The effect of the characteristics of the dataset on the selection stability. In: Proceedings—international conference on tools with artificial intelligence, ICTAI, pp 970–977. https://doi.org/10.1109/ICTAI.2011.167

Alelyani S, Tang J, Liu H (2013) Feature selection for clustering: a review. Data Cluster Algorithms Appl 29:110–121

Alter O, Alter O (2000) Singular value decomposition for genome-wide expression data processing and modeling. Proc Natl Acad Sci USA 97(18):10101–10106

Ambusaidi MA, He X, Nanda P (2015) Unsupervised feature selection method for intrusion detection system. In: Trustcom/BigDataSE/ISPA, 2015 IEEE, vol 1, pp 295–301. https://doi.org/10.1109/Trustcom.2015.387

Ang JC, Mirzal A, Haron H, Hamed HNA (2016) Supervised, unsupervised, and semi-supervised feature selection: a review on gene selection. IEEE/ACM Trans Comput Biol Bioinform 13(5):971–989. https://doi.org/10.1109/TCBB.2015.2478454

Argyriou A, Evgeniou T, Pontil M (2008) Convex multi-task feature learning. Mach Learn 73(3):243–272

Banerjee M, Pal NR (2014) Feature selection with SVD entropy: some modification and extension. Inf Sci 264:118–134. https://doi.org/10.1016/j.ins.2013.12.029

Beni G, Wang J (1993) Swarm intelligence in cellular robotic systems. In: Dario P, Sandini G, Aebischer P (eds) Robots and biological systems: towards a new bionics?. Springer, Berlin, pp 703–712. https://doi.org/10.1007/978-3-642-58069-7_38

Bharti KK, kumar Singh P (2014) A survey on filter techniques for feature selection in text mining. In: Proceedings of the second international conference on soft computing for problem solving (SocProS 2012), December 28–30, 2012. Springer, pp 1545–1559

Bolón-Canedo V, Sánchez-Maroño N, Alonso-Betanzos A (2015) Feature selection for high-dimensional data. https://doi.org/10.1007/978-3-319-21858-8

Boyd S, Parikh N, Chu E, Peleato B, Eckstein J et al (2011) Distributed optimization and statistical learning via the alternating direction method of multipliers. Found Trends Mach® Learn 3(1):1–122

Breaban M, Luchian H (2011) A unifying criterion for unsupervised clustering and feature selection. Pattern Recognit 44(4):854–865. https://doi.org/10.1016/j.patcog.2010.10.006

Cai D, Zhang C, He X (2010) Unsupervised feature selection for multi-cluster data. In: Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, pp 333–342

Cai J, Luo J, Wang S, Yang S (2018) Feature selection in machine learning: a new perspective. Neurocomputing 0:1–10. https://doi.org/10.1016/j.neucom.2017.11.077

Calinski T, Harabasz J (1974) A dendrite method for cluster analysis. Commun Stat Theory Methods 3(1):1–27. https://doi.org/10.1080/03610927408827101, http://www.tandfonline.com/doi/abs/10.1080/03610927408827101?journalCode=lsta19#preview

Chakrabarti S, Frank E, Güting RH, Han J, Jiang X, Kamber M, Lightstone SS, Nadeau TP, Neapolitan RE et al (2008) Data mining: know it all. Elsevier Science. https://books.google.com.mx/books?id=WRqZ0QsdxKkC

Chandrashekar G, Sahin F (2014) A survey on feature selection methods. Comput Electr Eng 40(1):16–28. https://doi.org/10.1016/j.compeleceng.2013.11.024

Chung FRK (1997) Spectral graph theory, vol 92. American Mathematical Society, Providence

Cortes C, Vapnik V (1995) Support-vector networks. Mach Learn 20(3):273–297

Cover TM, Thomas JA (2006) Elements of information theory, 2nd edn. Wiley, New York

Dadaneh BZ, Markid HY, Zakerolhosseini A (2016) Unsupervised probabilistic feature selection using ant colony optimization. Expert Syst Appl 53:27–42. https://doi.org/10.1016/j.eswa.2016.01.021

Daniels MJ, Normand SLT (2005) Longitudinal profiling of health care units based on continuous and discrete patient outcomes. Biostatistics 7(1):1–15

Dash M, Liu H (2000) Feature selection for Clustering. In: Terano T, Liu H, Chen ALP (eds) Knowledge discovery and data mining. Current issues and new applications, vol 1805, pp 110–121. https://doi.org/10.1007/3-540-45571-X_13

Dash M, Ong YS (2011) RELIEF-C: efficient feature selection for clustering over noisy data. In: 2011 23rd IEEE international conference on tools with artificial intelligence (ICTAI). IEEE, pp 869–872

Dash M, Liu H, Yao J (1997) Dimensionality reduction of unsupervised data. In: Proceedings Ninth IEEE international conference on tools with artificial intelligence. IEEE Computer Society, pp 532–539. https://doi.org/10.1109/TAI.1997.632300, http://ieeexplore.ieee.org/document/632300/

Dash M, Choi K, Scheuermann P, Liu HLH (2002) Feature selection for clustering—a filter solution. In: 2002 Proceedings 2002 IEEE international conference on data mining. pp 115–122. https://doi.org/10.1109/ICDM.2002.1183893

De Leon AR, Chough KC (2013) Analysis of mixed data: methods and applications. CRC Press, London

Dempster AP, Laird NM, Rubin DB (1977) Maximum Likelihood from Incomplete Data via the EM-Alogrithm, vol 39. https://doi.org/10.2307/2984875, arXiv:0710.5696v2

Devakumari D, Thangavel K (2010) Unsupervised adaptive floating search feature selection based on Contribution Entropy. In: 2010 International conference on communication and computational intelligence (INCOCCI). IEEE, pp 623–627

Devaney M, Ram A (1997) Efficient feature selection in conceptual clustering. In: ICML '97 Proceedings of the fourteenth international conference on machine learning. pp 92–97. Morgan Kaufmann Publishers Inc, San Francisco, CA. http://dl.acm.org/citation.cfm?id=645526.657124

Devijver PA, Kittler J (1982) Pattern recognition: a statistical approach. Pattern recognition: a statistical approach. http://www.scopus.com/inward/record.url?eid=2-s2.0-0019926397&partnerID=40

Dong G, Liu H (2018) Feature engineering for machine learning and data analytics. CRC Press. https://books.google.com.au/books?hl=en&lr=&id=QmNRDwAAQBAJ&oi=fnd&pg=PT15&ots=4FR0a_rfAH&sig=xMBalldd_vLcQdcnDWy9q7c_z7c#v=onepage&q&f=false

Donoho DL, Tsaig Y (2008) Fast solution of-norm minimization problems when the solution may be sparse. IEEE Trans Inf Theory 54(11):4789–4812

Dorigo M, Gambardella LM (1997) Ant colony system: a cooperative learning approach to the traveling salesman problem. IEEE Trans Evolut Comput 1(1):53–66

Du S, Ma Y, Li S, Ma Y (2017) Robust unsupervised feature selection via matrix factorization. Neurocomputing 241:115–127. https://doi.org/10.1016/j.neucom.2017.02.034

Dutta D, Dutta P, Sil J (2014) Simultaneous feature selection and clustering with mixed features by multi objective genetic algorithm. Int J Hybrid Intell Syst 11(1):41–54

Dy JG, Brodley CE (2004) Feature selection for unsupervised learning. J Mach Learn Res 5:845–889. https://doi.org/10.1016/j.patrec.2014.11.006

El Ghaoui L, Li GC, Duong VA, Pham V, Srivastava AN, Bhaduri K (2011) Sparse machine learning methods for understanding large text corpora. In: CIDU, pp 159–173

Feldman R, Sanger J (2006) The text mining handbook. Cambridge university press. https://doi.org/10.1017/CBO9780511546914, https://www.cambridge.org/core/product/identifier/9780511546914/type/book, arXiv:1011.1669v3

Ferreira AJ, Figueiredo MA (2012) An unsupervised approach to feature discretization and selection. Pattern Recognit 45(9):3048–3060. https://doi.org/10.1016/j.patcog.2011.12.008

Figueiredo MAT, Jain AK (2002) Unsupervised learning of finite mixture models. IEEE Trans Pattern Anal Mach Intell 24(3):381–396. https://doi.org/10.1109/34.990138
Fisher DH (1987) Knowledge acquisition via incremental conceptual clustering. Mach Learn 2(2):139–172. https://doi.org/10.1023/A:1022852608280
Fix E, Hodges Jr JL (1951) Discriminatory analysis-nonparametric discrimination: consistency properties. Technical report. California University Berkeley
Forman G (2003) An extensive empirical study of feature selection metrics for text classification. J Mach Learn Res 3:1289–1305
Fowlkes EB, Gnanadesikan R, Kettenring JR (1988) Variable selection in clustering. J Classif 5(2):205–228. https://doi.org/10.1007/BF01897164
Friedman M (1937) The use of ranks to avoid the assumption of normality implicit in the analysis of variance. J Am Stat Assoc 32(200):675–701. https://doi.org/10.1080/01621459.1937.10503522
Friedman J, Hastie T, Tibshirani R (2001) The elements of statistical learning, 1st edn. Springer series in statistics. Springer, New York
Fukunaga K (1990) Introduction to statistical pattern recognition, vol 22. https://doi.org/10.1016/0098-3004(96)00017-9, http://books.google.com/books?id=BIJZTGjTxBgC&pgis=1, arXiv:1011.1669v3
García S, Luengo J, Herrera F (2015) Data preprocessing in data mining, 72nd edn. Springer, New York. https://doi.org/10.1007/978-3-319-10247-4
Garcia-Garcia D, Santos-Rodriguez R (2009) Spectral clustering and feature selection for microarray data. In: International conference on machine learning and applications, 2009 ICMLA '09 pp 425–428. https://doi.org/10.1109/ICMLA.2009.86
Gu S, Zhang L, Zuo W, Feng X (2014) Projective dictionary pair learning for pattern classification. In: Advances in neural information processing systems, pp 793–801
Guo J, Zhu W (2018) Dependence guided unsupervised feature selection. In: Aaai, pp 2232–2239
Guo J, Guo Y, Kong X, He R (2017) Unsupervised feature selection with ordinal locality school of information and communication engineering. Dalian University of Technology National, Laboratory of Pattern Recognition, CASIA Center for Excellence in Brain Science and Intelligence Technology, Dalian
Guyon I, Elisseeff A, De AM (2003) An introduction to variable and feature selection. J Mach Learn Res 3:1157–1182. https://doi.org/10.1016/j.aca.2011.07.027, arXiv:1111.6189v1
Haindl M, Somol P, Ververidis D, Kotropoulos C (2006) Feature selection based on mutual correlation. In: Progress in pattern recognition, image analysis and applications, pp 569–577
Hall MA (1999) Correlation-based feature selection for machine learning. Ph.D. thesis, University of Waikato Hamilton
Hall M, Frank E, Holmes G, Pfahringer B, Reutemann P, Witten IH (2009) The WEKA data mining software: an update. SIGKDD Explor Newsl 11(1):10–18. https://doi.org/10.1145/1656274.1656278
Han J, Sun Z, Hao H (2015) Selecting feature subset with sparsity and low redundancy for unsupervised learning. Knowl Based Syst 86:210–223. https://doi.org/10.1016/j.knosys.2015.06.008
He X, Niyogi P (2004) Locality preserving projections. In: Advances in neural information processing systems, pp 153–160
He X, Cai D, Niyogi P (2005) Laplacian score for feature selection. In: Advances in neural information processing systems 18, vol 186, pp 507–514
Hou C, Nie F, Yi D, Wu Y (2011) Feature selection via joint embedding learning and sparse regression. In: IJCAI Proceedings-international joint conference on artificial intelligence, Citeseer, vol 22. pp 1324
Hou C, Nie F, Li X, Yi D, Wu Y (2014) Joint embedding learning and sparse regression: a framework for unsupervised feature selection. IEEE Trans Cybern 44(6):793–804
Hruschka ER, Covoes TF (2005) Feature selection for cluster analysis: an approach based on the simplified Silhouette criterion. In: 2005 and international conference on intelligent agents, web technologies and internet commerce, international conference on computational intelligence for modelling, control and automation, vol 1. IEEE, pp 32–38
Hruschka ER, Hruschka ER, Covoes TF, Ebecken NFF (2005) Feature selection for clustering problems: a hybrid algorithm that iterates between k-means and a Bayesian filter. In: Fifth international conference on hybrid intelligent systems, 2005. HIS '05. IEEE. https://doi.org/10.1109/ICHIS.2005.42
Hruschka ER, Covoes TF, Hruschka JER, Ebecken NFF (2007) Adapting supervised feature selection methods for clustering tasks. In: Methods for clustering tasks in managing worldwide operations and communications with information technology (IRMA 2007 proceedings), information resources management association (IRMA) international conference vancouver 2007 99-102 Hershey: Idea Group Publishing. https://doi.org/10.4018/978-1-59904-929-8.ch024
Hu J, Xiong C, Shu J, Zhou X, Zhu J (2009) An improved text clustering method based on hybrid model. Int J Modern Educ Comput Sci 1(1):35

Huang Z (1997) Clustering large data sets with mixed numeric and categorical values. In: Proceedings of the 1st Pacific-Asia conference on knowledge discovery and data mining,(PAKDD), Singapore. pp 21–34
Huang Z (1998) Extensions to the k-means algorithm for clustering large data sets with categorical values. Data Min Knowl Discov 2(3):283–304
Jashki A, Makki M, Bagheri E, Ghorbani AA (2009) An iterative hybrid filter-wrapper approach to feature selection for document clustering. In: Proceedings of the 22nd Canadian conference on artificial intelligence (AI'09) 2009
John GH, Langley P (1995) Estimating continuous distributions in Bayesian classifiers. In: Proceedings of the eleventh conference on uncertainty in artificial intelligence. Morgan Kaufmann Publishers Inc., pp 338–345
Kim Y, Gao J (2006) Unsupervised gene selection for high dimensional data. In: Sixth IEEE symposium on bioinformatics and bioengineering (BIBE'06), pp 227–234. https://doi.org/10.1109/BIBE.2006.253339, http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4019664
Kim Y, Street WN, Menczer F (2002) Evolutionary model selection in unsupervised learning. Intell Data Anal 6(6):531–556
Kong D, Ding C, Huang H (2011) Robust nonnegative matrix factorization using l21-norm. In: Proceedings of the 20th ACM international conference on Information and knowledge management (CIKM), pp 673–682. https://doi.org/10.1145/2063576.2063676, http://dl.acm.org/citation.cfm?id=2063676
Kotsiantis SB (2011) Feature selection for machine learning classification problems: a recent overview. Artifi Intell Rev 42:157–176. https://doi.org/10.1007/s10462-011-9230-1
Law MHC, Figueiredo MAT, Jain AK (2004) Simultaneous feature selection and clustering using mixture models. IEEE Trans Pattern Anal Mach Intell 26(9):1154–1166
Lazar C, Taminau J, Meganck S, Steenhoff D, Coletta A, Molter C, De Schaetzen V, Duque R, Bersini H, Nowé A (2012) A survey on filter techniques for feature selection in gene expression microarray analysis. IEEE/ACM Trans Comput Biol Bioinform 9(4):1106–1119. https://doi.org/10.1109/TCBB.2012.33
Lee W, Stolfo SJ, Mok KW (2000) Adaptive intrusion detection: a data mining approach. Artif Intell Rev 14(6):533–567
Lee PY, Loh WP, Chin JF (2017) Feature selection in multimedia: the state-of-the-art review. Image Vis Comput 67:29–42. https://doi.org/10.1016/j.imavis.2017.09.004
Li Z, Tang J (2015) Unsupervised feature selection via nonnegative spectral analysis and redundancy control. IEEE Trans Image Process 24(12):5343–5355. https://doi.org/10.1109/TIP.2015.2479560, http://ieeexplore.ieee.org/document/7271072/
Li Y, Lu BL, Wu ZF (2006) A hybrid method of unsupervised feature selection based on ranking. In: 18th international conference on pattern recognition (ICPR'06), Hong Kong, China, pp 687–690. https://doi.org/10.1109/ICPR.2006.84, http://dl.acm.org/citation.cfm?id=1172253
Li Y, Lu BL, Wu ZF (2007) Hierarchical fuzzy filter method for unsupervised feature selection. J Intell Fuzzy Syst 18(2):157–169. http://dl.acm.org/citation.cfm?id=1368376.1368381
Li Z, Yang Y, Liu J, Zhou X, Lu H (2012) Unsupervised feature selection using nonnegative spectral analysis. In: AAAI
Li Z, Cheong LF, Zhou SZ (2014a) SCAMS: Simultaneous clustering and model selection. In: Proceedings of the IEEE computer society conference on computer vision and pattern recognition, pp 264–271. https://doi.org/10.1109/CVPR.2014.41
Li Z, Liu J, Yang Y, Zhou X, Lu H (2014b) Clustering-guided sparse structural learning for unsupervised feature selection. IEEE Trans Knowl Data Eng 26(9):2138–2150
Li J, Cheng K, Wang S, Morstatter F, Trevino RP, Tang J, Liu H (2016) Feature selection: a data perspective. J Mach Learn Res 1–73. arXiv:1601.07996
Lichman M (2013) UCI Machine learning repository. http://archive.ics.uci.edu/ml
Liu H, Motoda H (1998) Feature selection for knowledge discovery and data mining. https://doi.org/10.1007/978-1-4615-5689-3, arXiv:1011.1669v3
Liu H, Motoda H (2007) Computational methods of feature selection. CRC Press, London
Liu DC, Nocedal J (1989) On the limited memory BFGS method for large scale optimization. Math Program 45(1–3):503–528. https://doi.org/10.1007/BF01589116, arXiv:1011.1669v3
Liu H, Yu L, Member SS, Yu L, Member SS (2005) Toward integrating feature selection algorithms for classification and clustering. IEEE Trans Knowl Data Eng 17(4):491–502. https://doi.org/10.1109/TKDE.2005.66
Liu J, Ji S, Ye J (2009a) Multi-task feature learning via efficient l 2, 1-norm minimization. In: Proceedings of the twenty-fifth conference on uncertainty in artificial intelligence. AUAI Press, pp 339–348
Liu R, Yang N, Ding X, Ma L (2009b) An unsupervised feature selection algorithm: Laplacian score combined with distance-based entropy measure. In: 3rd international symposium on intelligent information technology application, IITA 2009, vol 3, pp 65–68. https://doi.org/10.1109/IITA.2009.390

Liu H, Wei R, Jiang G (2013) A hybrid feature selection scheme for mixed attributes data. Comput Appl Math 32(1):145–161
Lu Q, Li X, Dong Y (2018) Structure preserving unsupervised feature selection. Neurocomputing 301:36–45. https://doi.org/10.1016/j.neucom.2018.04.001
Luo Y, Xiong S (2009) Clustering ensemble for unsupervised feature selection. In: Fourth international conference on fuzzy systems and knowledge discovery. IEEE Computer Society, Los Alamitos, vol 1, pp 445–448. https://doi.org/10.1109/FSKD.2009.449
Luo M, Nie F, Chang X, Yang Y, Hauptmann AG, Zheng Q (2018) Adaptive unsupervised feature selection with structure regularization. IEEE Trans Neural Netw Learn Syst 29(4):944–956. https://doi.org/10.1109/TNNLS.2017.2650978, http://www.contrib.andrew.cmu.edu/~uqxchan1/papers/TNNLS2017_ANFS.pdf
Luxburg U (2007) A tutorial on spectral clustering. Stat Comput 17(4):395–416. https://doi.org/10.1007/s11222-007-9033-z, http://dl.acm.org/citation.cfm?id=1288832
MacQueen JB (1967) Some methods for classification and analysis of multivariate observations. In: Proceedings of 5-th Berkeley symposium on mathematical statistics and probability, vol 1, pp 281–297. http://projecteuclid.org/euclid.bsmsp/1200512992
Mao K (2005) Identifying critical variables of principal components for unsupervised feature selection. Syst Man Cybern Part B Cybern 35(2):339–44. https://doi.org/10.1109/TSMCB.2004.843269
Maron ME (1961) Automatic indexing: an experimental inquiry. J ACM 8(3):404–417. https://doi.org/10.1145/321075.321084, http://portal.acm.org/citation.cfm?doid=321075.321084
Miao J, Niu L (2016) A survey on feature selection. Procedia Comput Sci 91(Itqm):919–926. https://doi.org/10.1016/j.procs.2016.07.111
Mitra PFSUFS, Ca M, Pal SK (2002) Unsupervised feature selection using feature similarity. IEEE Trans Pattern Anal Mach Intelligence 24(3):301–312. https://doi.org/10.1109/34.990133
Mugunthadevi K, Punitha SC, Punithavalli M (2011) Survey on feature selection in document clustering. Int J Comput Sci Eng 3(3):1240–1244. http://www.enggjournals.com/ijcse/doc/IJCSE11-03-03-077.pdf
Nie F, Huang H, Cai X, Ding CH (2010) Efficient and robust feature selection via joint 2, 1-norms minimization. In: Advances in neural information processing systems, pp 1813–1821
Nie F, Zhu W, Li X (2016) Unsupervised feature selection with structured graph optimization. In: Proceedings of the 30th conference on artificial intelligence (AAAI 2016), vol 13, No. 9, pp 1302–1308
Niijima S, Okuno Y (2009) Laplacian linear discriminant analysis approach to unsupervised feature selection. IEEE ACM Trans Comput Biol Bioinform 6(4):605–614. https://doi.org/10.1109/TCBB.2007.70257
Osborne MR, Presnell B, Turlach BA (2000) On the lasso and its dual. J Comput Graph Stat 9(2):319–337
Padungweang P, Lursinsap C, Sunat K (2009) Univariate filter technique for unsupervised feature selection using a new Laplacian score based local nearest neighbors. In: Asia-Pacific conference on information processing, 2009. APCIP 2009, vol 2. IEEE, pp 196–200
Pal SK, Mitra P (2004) Pattern Recognit Algorithms Data Min, 1st edn. Chapman and Hall/CRC, London
Pal SK, De RK, Basak J (2000) Unsupervised feature evaluation: a neuro-fuzzy approach. IEEE Trans Neural Netw 11(2):366–376
Peng H, Long F, Ding C (2005) Feature selection based on mutual information: criteria of max-dependency, max-relevance, and min-redundancy. IEEE Trans Pattern Anal Mach Intell 27(8):1226–1238. https://doi.org/10.1109/TPAMI.2005.159
Qian M, Zhai C (2013) Robust unsupervised feature selection. In: Proceedings of the twenty-third international joint conference on artificial intelligence, pp 1621–1627. http://dl.acm.org/citation.cfm?id=2540361
Rao VM, Sastry VN (2012) Unsupervised feature ranking based on representation entropy. In: 2012 1st international conference on recent advances in information technology, RAIT-2012, pp 421–425. https://doi.org/10.1109/RAIT.2012.6194631
Ritter G (2015) Robust cluster analysis and variable selection, vol 137. CRC Press, London
Roth V, Lange T (2004) Feature selection in clustering problems. Adv Neural Inf Process Syst 16:473–480
Roweis ST, Saul LK (2000) Nonlinear dimensionality reduction by locally linear embedding. Science (New York, NY) 290(5500):2323–2326. https://doi.org/10.1126/science.290.5500.2323
Saeys Y, Inza I, Larrañaga P (2007) A review of feature selection techniques in bioinformatics. Bioinformatics 23(19):2507–2517. https://doi.org/10.1093/bioinformatics/btm344
Sheikhpour R, Sarram MA, Gharaghani S, Chahooki MAZ (2017) A survey on semi-supervised feature selection methods. Pattern Recognit 64(2016):141–158. https://doi.org/10.1016/j.patcog.2016.11.003
Shi L, Du L, Shen YD (2015) Robust spectral learning for unsupervised feature selection. In: Proceedings—IEEE international conference on data mining, ICDM 2015-Janua, pp 977–982. https://doi.org/10.1109/ICDM.2014.58

Shi Y, Miao J, Wang Z, Zhang P, Niu L (2018) Feature Selection With L2,1–2 Regularization. IEEE Trans Neural Netw Learn Syst 29(10):4967–4982. https://doi.org/10.1109/TNNLS.2017.2785403, https://ieeexplore.ieee.org/document/8259312/

Solorio-Fernández S, Carrasco-Ochoa J, Martínez-Trinidad J (2016) A new hybrid filterwrapper feature selection method for clustering based on ranking. Neurocomputing 214, https://doi.org/10.1016/j.neucom.2016.07.026

Solorio-Fernández S, Martínez-Trinidad JF, Carrasco-Ochoa JA (2017) A new unsupervised spectral feature selection method for mixed data: a filter approach. Pattern Recognit 72:314–326. https://doi.org/10.1016/j.patcog.2017.07.020

Swets D, Weng J (1995) Efficient content-based image retrieval using automatic feature selection. Proceedings, international symposium on computer vision, 1995. pp 85–90, http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=476982

Tabakhi S, Moradi P (2015) Relevance-redundancy feature selection based on ant colony optimization. Pattern Recognit 48(9):2798–2811. https://doi.org/10.1016/j.patcog.2015.03.020

Tabakhi S, Moradi P, Akhlaghian F (2014) An unsupervised feature selection algorithm based on ant colony optimization. Eng Appl Artif Intell 32:112–123. https://doi.org/10.1016/j.engappai.2014.03.007

Tabakhi S, Najafi A, Ranjbar R, Moradi P (2015) Gene selection for microarray data classification using a novel ant colony optimization. Neurocomputing 168:1024–1036. https://doi.org/10.1016/j.neucom.2015.05.022

Talavera L (2000) Dependency-based feature selection for clustering symbolic data. Intell Data Anal 4:19–28

Tang J, Liu H (2014) An unsupervised feature selection framework for social media data. IEEE Trans Knowl Data Eng 26(12):2914–2927

Tang J, Alelyani S, Liu H (2014) Feature selection for classification: a review. In: Data Classification, CRC Press, pp 37–64. https://doi.org/10.1201/b17320

Tang C, Liu X, Li M, Wang P, Chen J, Wang L, Li W (2018a) Robust unsupervised feature selection via dual self-representation and manifold regularization. Knowl Based Syst 145:109–120. https://doi.org/10.1016/j.knosys.2018.01.009

Tang C, Zhu X, Chen J, Wang P, Liu X, Tian J (2018b) Robust graph regularized unsupervised feature selection. Expert Syst Appl 96:64–76. https://doi.org/10.1016/j.eswa.2017.11.053

Theodoridis S, Koutroumbas K (2008a) Pattern recognition. Elsevier Science. https://books.google.com.mx/books?id=QgD-3Tcj8DkC

Theodoridis S, Koutroumbas K (2008b) Pattern recognition, 4th edn. Academic Press, New York

Tibshirani R (1996) Regression shrinkage and selection via the lasso. J R Stat Soc Ser B (Methodological) 58:267–288

Tou JT, González RC (1974) Pattern recognition principles. Addison-Wesley Pub. Co. https://books.google.com/books?id=VWQoAQAAIAAJ

Varshavsky R, Gottlieb A, Linial M, Horn D (2006) Novel unsupervised feature filtering of biological data. Bioinformatics 22(14):e507–e513. https://doi.org/10.1093/bioinformatics/btl214, http://bioinformatics.oxfordjournals.org/content/22/14/e507.abstract

Vergara JR, Estévez PA (2014) A review of feature selection methods based on mutual information. Neural Comput Appl 24(1):175–186, https://doi.org/10.1007/s00521-013-1368-0, arXiv:1509.07577

Wang S, Wang H (2017) Unsupervised feature selection via low-rank approximation and structure learning. Knowl Based Syst 124:70–79. https://doi.org/10.1016/j.knosys.2017.03.002

Wang S, Pedrycz W, Zhu Q, Zhu W (2015a) Unsupervised feature selection via maximum projection and minimum redundancy. Knowl Based Syst 75:19–29. https://doi.org/10.1016/j.knosys.2014.11.008

Wang S, Tang J, Liu H (2015b) Embedded unsupervised feature selection. In: Twenty-ninth AAAI conference on artificial intelligence, p 7

Wang X, Zhang X, Zeng Z, Wu Q, Zhang J (2016) Unsupervised spectral feature selection with l1-norm graph. Neurocomputing 200:47–54. https://doi.org/10.1016/j.neucom.2016.03.017

Webb AR (2003) Statistical pattern recognition, vol 35, 2nd edn. Wliey, New York. https://doi.org/10.1137/1035031

Wu M, Schölkopf B (2007) A local learning approach for clustering. In: Advances in neural information processing systems, pp 1529–1536

Yang Y, Liao Y, Meng G, Lee J (2011a) A hybrid feature selection scheme for unsupervised learning and its application in bearing fault diagnosis. Expert Syst Appl 38(9):11311–11320. http://dblp.uni-trier.de/db/journals/eswa/eswa38.html#YangLML11

Yang Y, Shen HT, Ma Z, Huang Z, Zhou X (2011b) L2,1-Norm regularized discriminative feature selection for unsupervised learning. In: IJCAI international joint conference on artificial intelligence, pp 1589–1594. https://doi.org/10.5591/978-1-57735-516-8/IJCAI11-267

Yasmin M, Mohsin S, Sharif M (2014) Intelligent image retrieval techniques: a survey. J Appl Res Technology 12(1):87–103

Yen CC, Chen LC, Lin SD (2010) Unsupervised feature selection: minimize information redundancy of features. In: Proceedings—international conference on technologies and applications of artificial intelligence, TAAI 2010. pp 247–254. https://doi.org/10.1109/TAAI.2010.49

Yi Y, Zhou W, Cao Y, Liu Q, Wang J (2016) Unsupervised feature selection with graph regularized nonnegative self-representation. In: You Z, Zhou J, Wang Y, Sun Z, Shan S, Zheng W, Feng J, Zhao Q (eds) Biometric recognition: 11th Chinese conference, CCBR 2016, Chengdu, China, October 14–16, 2016, Proceedings. Springer International Publishing, Cham, pp 591–599. https://doi.org/10.1007/978-3-319-46654-5_65

Yu L (2005) Toward integrating feature selection algorithms for classification and clustering. IEEE Trans Knowl Data Eng 17(4):491–502

Yu J (2011) A hybrid feature selection scheme and self-organizing map model for machine health assessment. Appl Soft Comput 11(5):4041–4054

Zafarani R, Abbasi MA, Liu H (2014) Social media mining: an introduction. Cambridge University Press, Cambridge

Zeng H, Cheung YM (2011) Feature selection and kernel learning for local learning-based clustering. IEEE Trans Pattern Anal Mach Intell 33(8):1532–1547. https://doi.org/10.1109/TPAMI.2010.215

Zhao Z (2010) Spectral feature selection for mining ultrahigh dimensional data. Ph.d thesis, Tempe

Zhao Z, Liu H (2007) Spectral feature selection for supervised and unsupervised learning. In: Proceedings of the 24th international conference on machine learning. ACM, pp 1151–1157

Zhao Z, Liu H (2011) Spectral feature selection for data mining. CRC Press. pp 1–216. https://www.taylorfrancis.com/books/9781439862100

Zhao Z, Wang L, Liu H, Ye J (2013) On similarity preserving feature selection. IEEE Trans Knowl Data Eng 25(3):619–632. https://doi.org/10.1109/TKDE.2011.222, http://ieeexplore.ieee.org.proxy.lib.umich.edu/ielx5/69/6419729/06051436.pdf?tp=&arnumber=6051436&isnumber=6419729

Zheng Z, Lei W, Huan L (2010) Efficient spectral feature selection with minimum redundancy. In: Twenty-fourth AAAI conference on artificial intelligence, pp 1–6

Zhou W, Wu C, Yi Y, Luo G (2017) Structure preserving non-negative feature self-representation for unsupervised feature selection. IEEE Access 5:8792–8803. https://doi.org/10.1109/ACCESS.2017.2699741

Zhu P, Zuo W, Zhang L, Hu Q, Shiu SCK (2015) Unsupervised feature selection by regularized self-representation. Pattern Recognit 48(2):438–446

Zhu P, Hu Q, Zhang C, Zuo W (2016) Coupled dictionary learning for unsupervised feature selection. In: AAAI, pp 2422–2428

Zhu P, Zhu W, Wang W, Zuo W, Hu Q (2017) Non-convex regularized self-representation for unsupervised feature selection. Image Vis Comput 60:22–29. https://doi.org/10.1016/j.imavis.2016.11.014