Expert Systems With Applications 114 (2018) 46–53

Contents lists available at ScienceDirect

Expert Systems With Applications

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

Unsupervised graph-based feature selection via subspace and pagerank centrality

K. Henni[a,b,*], N. Mezghani[a,b], C. Gouin-Vallerand[a]

[a] LICEF Research Center, TELUQ university, 5800 Rue St-Denis, Montréal, QC, Canada
[b] Laboratoire de recherche en imagerie et orthopédie (LIO), Centre de recherche du CHUM, 900 Rue St-Denis, Montréal, QC, Canada

ARTICLE INFO

Article history:
Received 18 February 2018
Revised 12 July 2018
Accepted 13 July 2018
Available online 25 July 2018

Keywords:
Unsupervised feature selection
Graph centrality measure
Pagerank
Subspace learning
Projected densities
K-Nearest neighbors

ABSTRACT

Feature selection has become an indispensable part of intelligent systems, especially with the proliferation of high dimensional data. It identifies the subset of discriminative features leading to better learning performances, i.e., higher learning accuracy, lower computational cost and significant model interpretability. This paper proposes a new efficient unsupervised feature selection method based on graph centrality and subspace learning called UGFS for '*Unsupervised Graph-based Feature Selection*'. The method maps features on an affinity graph where the relationships (edges) between feature nodes are defined by means of data points subspace preference. Feature importance score is then computed on the entire graph using a centrality measure. For this purpose, we investigated the Google's PageRank method originally introduced to rank web-pages. The proposed feature selection method has been evaluated using classification and redundancy rates measured on the selected feature subsets. Comparisons with the well-known unsupervised feature selection methods, on gene/expression benchmark datasets, demonstrate the validity and the efficiency of the proposed method.

1. Introduction

The explosive use of new information technologies and their various applications involves large amounts of high dimensional and complex data, which suffer from the curse of dimensionality (Duda, Hart, & Stork, 2001). The data complexity affects the efficiency of expert and intelligent systems and their decision-making performance. To overcome these limitations, a selection of relevant features from these high dimensional data is needed. The selection of the best features to be used in expert systems is a key issue in obtaining a satisfactory performance (Martinez-Gonzalez, Pardo, Echeverry-Correa, & San-Segundo, 2017). An efficient feature selection method identifies the subset of discriminative features leading to better learning performances, i.e., higher learning accuracy, lower computational cost and significant model interpretability. Hence, various methods have been proposed in the literature, such as (1) feature extraction, where a feature space is computed based on the combination of the original features (Abdi & Williams, 2010; Aliyari, Rudzicz, & Moghaddam, 2015; Choi & Choi, 2007), (2) feature selection, where a subset of relevant and less redundant features are selected or ranked respecting their relevance order (Bennasar, Hicks, & Setchi, 2015; Hall, 2000; Hu, Gao, Zhao, Zhang, & Wang, 2018) and (3) subspace and projected learning, where a subset of relevant features for a given clusters or data instance is selected/weighted and used in learning simultaneously (Elhamifar & Vidal, 2013; Parsons, Haque, & Liu, 2004; Vidal, 2011). Subspace learning is currently introduced in several data-mining techniques, especially in data stream analysis where computational time and costs reduction are crucial (Hassani, Kim, Choi, & Seidl, 2014).

Both feature extraction and feature selection are designed to improve learning performance as well as to decrease computational complexity and required storage. Feature extraction algorithms are very popular. However, they consist in transforming and compressing the original data which can affect data analysis efficiency. Therefore, feature selection methods, which select the most relevant features without any transformation, are considered as an alternative in processing high-dimensional data. Such methods become attractive in recent years.

Feature selection algorithms can be categorized into (1) supervised/unsupervised methods according on whether the data are labeled or not, (2) filter/wrapper/embedded methods according to

[*] Corresponding author at: LICEF Research Center, TELUQ university, 5800 Rue St-Denis, Montréal, QC, Canada.

E-mail addresses: khadidja.henni@licef.teluq.ca, khadidja.henni@licef.ca (K. Henni), neila.mezghani@teluq.ca (N. Mezghani), charles.gouin-vallerand@teluq.ca (C. Gouin-Vallerand).

https://doi.org/10.1016/j.eswa.2018.07.029
0957-4174/© 2018 Elsevier Ltd. All rights reserved.

the degree of learning involvement or (3) univariate/multivariate according to the consideration of the features interaction potential. The well-known unsupervised feature selection algorithms, such as Laplacian Score (He, Cai, & Niyogi, 2005), Spectral Feature Selection (SFS) (Zhao & Liu, 2007), Multi-Cluster Feature Selection (MCFS) (Cai, Zhang, & He, 2010), Minimum Redundancy Spectral Feature selection (MRSF) (Zheng, Lei, & Huan, 2010), characterize the manifold structure by graphs where nodes are the data instances. The Laplacian Score and SPFS use metrics to rank features, while MCFS and MRSF rank features based on a multi-output sparse regression. These methods rank features by capturing the manifold structure in a given graph. Thus, their efficiency depends strongly on the instances graph design. Unlike the previous graph-based methods, the supervised EigenVector Centrality for Feature Selection (ECFS), maps features into a graph and ranks them by the Eigenvector centrality measure (Roffo & Melzi, 2017). The graph design proposed by ECFS is based on pairwise relationships between features and some basic statistical metrics to define discriminative features (mutual information, Fisher score, and the standard deviation). Hence, it neglects the manifold structure preservation and does not exploit the features combination potential. Moradi and Rostami (2015) represented the set of features by a weighted graph, where features similarities, measured by Pearson product-moment correlation coefficient, are graph edges. Then, investigated the Louvain community detection algorithm to identify the feature clusters. Finally, a centrality measure is proposed to filter and rank features. This graph-based method demonstrated competitive results. Nevertheless, it is slow and addressed more feature redundancy than relevance. Despite the centrality measures popularity in graph theory and their efficiency in scoring and ranking nodes according to their topological importance and roles within the graph, the ranking still depends on the graph design.

In this research, we propose a new unsupervised feature selection method called '*Unsupervised Graph-based Feature Selection*' (UGFS), which outputs the features ranking vector. We investigated the Google's PageRank centrality measure (Gleich, 2015), to analysis feature graph structure and attribute to each feature an importance score. We also addressed the problem of defining the relationships between features, in order to establish the feature graph structure. This graph is designed by means of the 'subspace preference clusters' concept, which is driven from subspace learning and supports the PageRank to highly score the relevant features for classification problems.

This paper is organized as follows: Section 2 presents related works and Section 3 describes the mathematical framework. In Section 4, the details of the proposed unsupervised graph-based method are given. Experimental results are depicted in Section 5. Finally, Section 6 concludes the study and presents perspectives.

2. Related work

The high-dimensional data analysis methods attempt to reduce the number of treated features by (1) a preprocessing step in which relevant features are selected and/or highly scored and (2) adapting learning algorithms to consider feature subspaces in the learning task. This section overviews the unsupervised methods, both in the feature selection field and in subspace learning. Then, it presents the well-known graph centrality measures which are a key contribution of this study.

2.1. Unsupervised feature selection algorithms

The two families of unsupervised feature selection methods are filters and embedded. Filter methods are univariate as they scored features individually and neglected the features interaction potential (Somol, Baesens, Pudil, & Vanthienen, 2005). Features are evaluated according to filter criteria such as variances among features in MaxVar (Krzanowski, 1987) and Laplacian score (He et al., 2005). In contrast to univariate methods, multivariate methods have been proposed as spectral feature selection (SFS) (Zhao & Liu, 2007). Such algorithms preserve the manifold structure of data, but they do not investigate discriminative information.

Several levels of embedded methods have been proposed, which differ in terms of the used learning algorithm and in which step it is used. TraceRatio (Nie, Xiang, Jia, Zhang, & Yan, 2008) and Unsupervised Discriminative Feature Selection (UDFS) (Yang, Shen, Ma, Huang, & Zhou, 2011) are the simplest embedded algorithms. They capture the manifold structure of data by performing a fit learning to highly score the most discriminative features. Nevertheless, these algorithms present some limitations. Indeed, TraceRatio generates redundant features and UDFS uses restrictive constraints.

Algorithms based on clustering such: Multi-Class Feature Selection (MCFS) (Cai et al., 2010), Similarity Preserving Feature Selection (SPFS) (Zhao, Wang, Liu, & Ye, 2013) and Minimum Redundancy Spectral Feature Selection (MRFS) (Zheng et al., 2010), use cluster analysis to select features after a fit learning step. Others, like the Local Learning based Clustering Feature Selection (Zeng & Cheung, 2010) (LLCFS), uses clustering to learn adaptive data structure with selected features. It updated the Laplacian graph iteratively by means of the relevance of each feature. These algorithms gave relevant features subsets but they are slow and not scalable.

Data sparsity in high-dimensional spaces reduced the impact of the pairwise similarity between samples to discriminate classes. Thereby, the sparse representation studies (Zhang et al., 2017) emerged and where the of $\ell_{2,1}$-norm demonstrated high learning performances on those spaces. This norm has been implemented in recent embedded feature selection methods. The purpose of these later consists on the minimization of the $\ell_{2,1}$-norm based on regression learning, where (1) The Regularized Self-Representation method (RSR) minimizes of the error between the projected data and the target matrix (Zhu, Zuo, Zhang, Hu, & Shiu, 2015), (2) the Simultaneous Orthogonal basis Clustering Decomposition Feature Selection (SOFCS), decomposes the target matrix based on orthogonal constraints (Han & Kim, 2015) and (3) Robust Unsupervised Feature Selection via Matrix Factorization (RUFSM) combines the feature selection with matrix factorization and manifold regularization into unified framework (Du, Ma, Li, & Ma, 2017).

Table 1 summarizes a comparative study of the well-known feature selection algorithms based on their theoretical proprieties: (1) their categories, (2) the classes of the used filters (statistic, similarity, etc.), (3) the number of user-input parameters, (4) the level of sensibility to parameters values changes, (5) the scalability.

2.2. Subspace and projected algorithms

Subspace learning algorithms have been proposed to cope with the various curse of dimensionality aspects (Li, Hung, & Chung, 2011), such (1) the distances concentration problem, where geometrical distances gave insignificant differences between different pairs of samples and (2) the hubness phenomenon related to the distance concentration problem, which affects the distribution of k-occurrences (Flexer & Schnitzer, 2015)

Indeed, a pair (C, S) is selected, where C is a set of points composing a cluster and S is a set of the most characterizing features of the considered cluster. CLIQUE is a subspace clustering algorithm based on grid (Böhm, Kailing, Kriegel, & Kroger, 2004). Based on an Apriori-like method, it recursively searches the set of all possible subspaces. It used a density threshold to filter cells. Based on conclusions given by Flexer and Schnitzer (2015) which demonstrated that Euclidian distances are not efficient, Böhm et al. (2004) proposed a weighted Euclidian distance based on subspace concepts

Table 1
Comparison of various feature selection algorithms corresponding to their theoretical proprieties.

Methods	Category	Filter class	Parameter	Parameter sensibility	Scalability
MaxVar	Filter	Statistic	1	Low	High
Laplacian	Filter	Similarity	3	Low	Low
SFS	Filter	Similarity	2	High	Low
UDFS	Embedded	Sparse learning	4	High	Low
TraceRatio	Embedded	Similarity	3	Low	Low
MCFS	Embedded	Sparse learning	4	Low	High
SPFS	Embedded	similarity	3	Low	Low
MRFS	Embedded	Statistic	2	High	High
LLCFS	Embedded	Clustering	4	High	Low
RSR	Embedded	Sparse learning	2	High	High
SOFCS	Embedded	Sparse learning	3	Low	High
RURSM	Embedded	Clustering	5	High	Low

and the well-founded notion of density connected clusters. Authors proposed the use of subspace preference cluster concepts, based on the variance of the data neighborhood along features, then weighted the Euclidian distance by these variances (more details are given in Section 3).

2.3. Graph centrality measures

The growth of social networks and web services motivated the centrality measures researches. Several points of view have been proposed to evaluate node importance in a graph. In '*Degree Centrality*', node importance is the number of its directly connected edges. '*Closeness Centrality*' (Opsahl, Agneessens, & Skvoretz, 2010) used distances between nodes and lower values reflect information on the graph. The '*Betweenness Centrality*' (Opsahl et al., 2010) highly scored nodes communicated to others with few intermediaries. The '*Eigenvector centrality*' (Opsahl et al., 2010) reflected the number of connections with nodes strongly connected with other graph actors. Google has proposed an efficient measure based on Eigenvector centrality called '*PageRank*' to investigate web pages relevance (Gleich, 2015). This simple and fast measure is general and well-defined for any given graph structure to capture various relations among nodes. PageRank has been applied in biology and bioinformatics to find and rank genes '*GeneRank*' (Morrison, Breitling, Higham, & Gilbert, 2005), proteins '*ProteinRank*' (Wu, Zhang, & Wei, 2013) or even to match protein-protein interactions (*IsoRank*). It is also used in neuroscience, complex engineered systems (*MoniorRank*), in the Linux kernel, bibliometrics (*CiteRank, TimedPageRank, AuthorRank*), in social networks (*SuperedgeRank* (Ma & Liu, 2014), *BuddyRank* and *TwitterRank*) as well as in other contexts (Gleich, 2015).

3. Mathematical framework

This section first presents notations then recalls the bases of subspace learning and PageRank.

3.1. Notations

Let X be a set of n points and $d-$dimensional features $X = \{\mathbf{x_1}, \ldots, \mathbf{x_n}\}$. Each point $\mathbf{x_i}$ is a vector of d features $\mathbf{x_i} = (x_i^1, \ldots, x_i^d)$. $dist(\mathbf{x_p}, \mathbf{x_q})$ is the Euclidean distance between two data points $\mathbf{x_p}$, $\mathbf{x_q} \in X$ and the $dist_p : X \times X \rightarrow \mathbb{R}$ is a metric distance function between projected points.

Our aim is to develop a new feature selection algorithm which maps features into an undirected graph. Let $G =< V, E >$ a graph, where the vertices (nodes) V are the set of features $V = \{\mathbf{x^1}, \ldots, \mathbf{x^d}\}$, and E the edges linking the vertices. A is the adjacency matrix associated to the graph G, where each of its element $a_{i,j}$ represents a pairwise relationship between features $\mathbf{x^i}$ and $\mathbf{x^j}$. $a_{i,j}$ is associated to a potential function $\phi(\mathbf{x^i}, \mathbf{x^j})$:

$$a_{i,j} = \phi(\mathbf{x^i}, \mathbf{x^j}) \tag{1}$$

The function ϕ can be a binary function as it can weight nodes composing the graph via several metrics.

3.2. Subspace preference clusters

Several studies have demonstrated the capacity of subspace preference to deal with high-dimensional spaces (Böhm et al., 2004; Elhamifar & Vidal, 2013; Parsons et al., 2004; Vidal, 2011). In this study, we use the subspace preference clusters among features (Böhm et al., 2004) to define relationships between features.

Subspace preference cluster is a set of points belonging to the same dense regions called 'density connected points', which are associated to a set of features called 'subspace preference vector'. Subspace preference clusters are sets of points with small variance along one or more features, i.e. a variance smaller than a given threshold $\delta \in \mathbb{R}$.

Let $\mathbf{x_p} \in X$ a data point and $k \in \mathbb{N}$. The variance $var_i(NN_k(\mathbf{x_p}))$ along a feature $\mathbf{x^i}$ is defined as follows:

$$var_i(NN_k(\mathbf{x_p})) = \frac{\sum_{\mathbf{x_q} \in NN_k(\mathbf{x_p})} \left(dist_p(x_p^i, x_q^i)\right)^2}{|NN_k(\mathbf{x_p})|} \tag{2}$$

where $NN_k(\mathbf{x_p})$ define the set of $k-$nearest neighborhoods of an object $\mathbf{x_p} \in X$.

The feature subspace preference associated to the data point $\mathbf{x_p} \in X$ is the set of features with $var_i(NN_k(\mathbf{x_p})) \leq \delta$, $\delta \in \mathbb{R}, k \in \mathbb{N}$. This features set preserves the density in the neighborhood of the point $\mathbf{x_p}$. Therefore, if the selected set of features (subspace preference cluster) has low variance in the neighborhood of points, thus those features are relevant and preserve the density inside the cluster of the data point $\mathbf{x_p}$.

3.3. PageRank

The PageRank measure has been introduced originally by Google to rank web-pages. It simulated the behavior of users when browsing the Web to rank pages, where pages are graph nodes and hyperlinks are edges. PageRank denotes the 'importance' of nodes under the assumptions that the importance of a node is the expected sum of the importance of all connected nodes and the direction of edges. Its value corresponds to the probability distribution of nodes being accessed at random. In graph theory, PageRank computes recursively a normalized and propagated value for each node in a graph.

Let x and p two nodes in a graph G, the PageRank of x is given as follows:

$$PR(x) = (1-c) + c. \sum_{p \in Pnt_{in}(x)} \frac{PR(p)}{|Pnt_{out}(p)|} \tag{3}$$

where c is a damping factor which takes its value in [0,1] (typically 0.85), $Pnt_{in}(x)$ is the set of nodes pointing to x and $Pnt_{out}(p)$ the set of nodes pointed by p and $|Pnt_{out}(p)|$ is its cardinality. The PageRank operated on the directed graph and its value for a given node is computed iteratively based on PageRank of nodes pointing on it. In order to deal with undirected graphs, some variants of PageRank have been proposed (Avrachenkov, Kadavankandy, Prokhorenkova, & Raigorodskii, 2015; Zhang, Lofgren, & Goel, 2016). In our study, we used the basic version of the algorithm. The PageRank vector is a stationary distribution of special formed Markov process, more details about its convergence are given in Gleich (2015).

4. Usupervised graph-based feature selection method '*UGFS*'

The purpose of this work consists in investigating the importance of features in an undirected graph using PageRank. It highlights nodes (feature) having a lot of connections. The graph design is a crucial step because features must be connected with respect to PageRank proprieties. We use the subspace preference clusters in order to define the edges linking features. Features relationships are defined according to their abilities to preserve the neighborhood densities of data points, i.e., features minimizing variances among projected neighborhood data of each core data point are linked.

In order to define feature relationships, the proposed algorithm scans the whole dataset searching the neighborhood of each data point. Then, it computes the variances among these sets. Based on a given threshold and the computed variances (see Section 3.2), the algorithm selects subspace preference clusters for each data point. Features belonging to the same subspace preference clusters S_p associated to the neighborhood of the point $\mathbf{x_p}$ are linked. Otherwise, if the subspace S_p preserves the local densities into the projected neighborhoods of the data point $\mathbf{x_p}$, then features composing S_p are the most relevant for the cluster of $\mathbf{x_p}$. That is, the edges linking those features must be created. The potential function associated to the graph G is given by:

$$\phi(\mathbf{x^i}, \mathbf{x^j}) = \begin{cases} 1, & \text{if } var_{s_p}(NN_k(\mathbf{x_p})) \leq \delta \\ 0, & \text{otherwise} \end{cases} \tag{4}$$

where $\mathbf{x^i}$**,** $\mathbf{x^j} \in S_p$ and $\delta \in \mathbb{R}$ is a variance threshold.

More details about the definition of feature relationships and graph design are given in Algorithm 1[1].

Finally, UGFS applies the PageRank system as a centrality measure of graph G, then features are ranked according to their PageRank score.

5. Experimental results

5.1. Experimental setup

UGFS is implemented in the MATLAB R2017 software (The Mathworks Inc, Massachusetts, USA), under Windows Operating System. Experimental evaluation is done on a laptop i5 Intel dual processor 2.3 GHz/CPU and 8 GB DRAM.

Algorithm 1: Feature graph design.

```
Input: Observed data $X = \{\mathbf{x_1}, \ldots, \mathbf{x_n}\}$, $k$, $\delta$.
Output: $G$ undirected graph of features.
Compute $NN_k(\mathbf{x_i})$, with $i = 1, \ldots, n$.
Compute $Var_j(NN_k(\mathbf{x_i}))$, with $i = 1, \ldots, n$ and $j = 1, \ldots, d$ (see section 3.2; equation 2).
$A(i, j) = 0, i = 1, \ldots, d$ and$j = 1, \ldots, d$.
for $i = 1 : n$ do
    for $j = 1 : d$ do
        if $Var_j(NN_k(\mathbf{x_i})) \leq \delta$ then
            $Var_{Binarized}(i, j) = 1$
        else
            $Var_{Binarized}(i, j) = 0$
        end if
    end for
end for
for $i = 1 : n$ do
    $S_p = \{\}$
    for $j = 1 : d$ do
        if $Var_{Binarized}(i, j) == 1$ then
            $S_p = S_p \cup \{\mathbf{x^j}\}$
        end if
    end for
    for $l = 1 : size(S_p)$ do
        for $m = 1 : size(S_p)$ do
            $A(l, m) = 1$
        end for
    end for
end for
$G = (\{\mathbf{x^1}, \ldots, \mathbf{x^d}\}, A)$
```

Table 2
Datasets description.

Datasets	Number of features	Number of instances	Number of classes
Colon	2000	62	2
Leukemia	7129	72	2
Ovarian cancer	4000	216	2
CLL_SUB_111	11,340	111	3

The evaluation of the proposed algorithm are done by means of (1) the classification rate and its standard deviation corresponding to feature subsets of different sizes are computed by a cross-validation representing 40% of the whole dataset, (2) the minimum number of features corresponding to the best classification rates and (3) the redundancy rate of the selected feature subsets.

The redundancy rate of a given feature subset S, is given by:

$$RED(S) = \frac{1}{d(d-1)} \sum_{\mathbf{x^i}, \mathbf{x^j} \in S, i > j} corr(\mathbf{x^i}, \mathbf{x^j}) \tag{5}$$

where d is the size of the feature dataset and $\mathbf{x^i}$**,** $\mathbf{x^j} \in S$. Large values of $RED(s)$ means that features of the subset S are significantly correlated.

We use two classifiers: (1) the support vector machine (SVM) for supervised classification (Cortes & Vapnik, 1995), which is widely used both in feature selection algorithm design and/or evaluation and (2) the k−means for unsupervised learning (Celebi, Kingravi, & Vela, 2013), which is simple, fast and requires only the number of clusters as an input parameter. k−means initial centroids choice influences highly its accuracy, that is why we use k−means++ algorithm to choose the centroids initial values.

Best feature ranking is then demonstrated by minimization of the evaluation criteria, except the classification rate where higher

[1] The source code will be posted on line to provide the needed material for the use of UGFS.

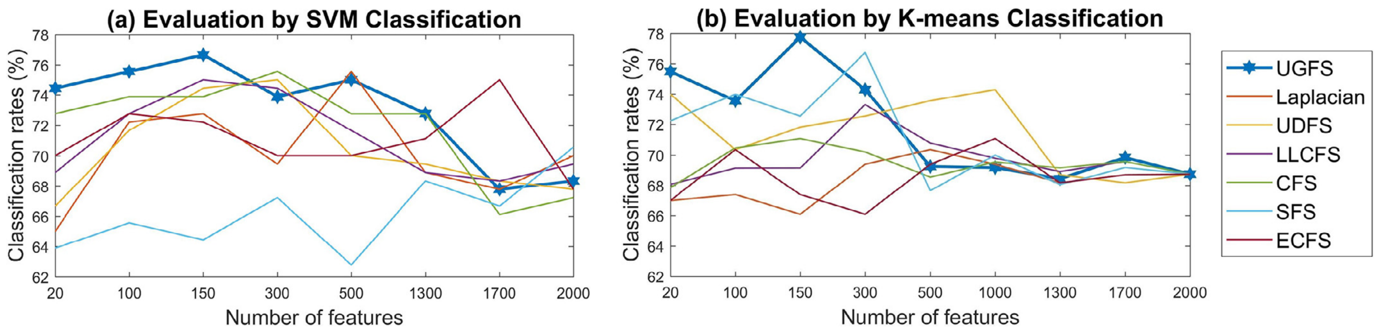

Fig. 1. Colon dataset: correct classification rate (%) of different feature selection algorithms, over a varied number of features.

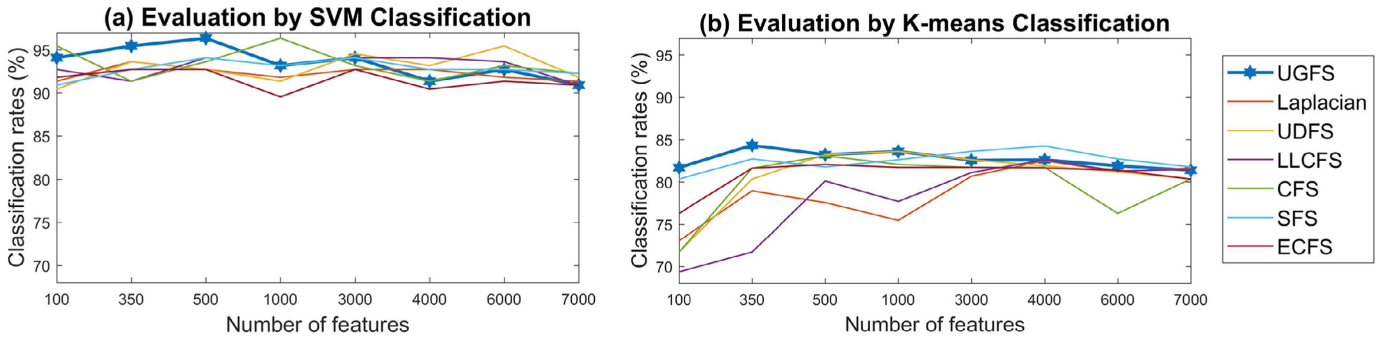

Fig. 2. leukemia dataset: correct classification rate (%) of different feature selection algorithms, over a varied number of features.

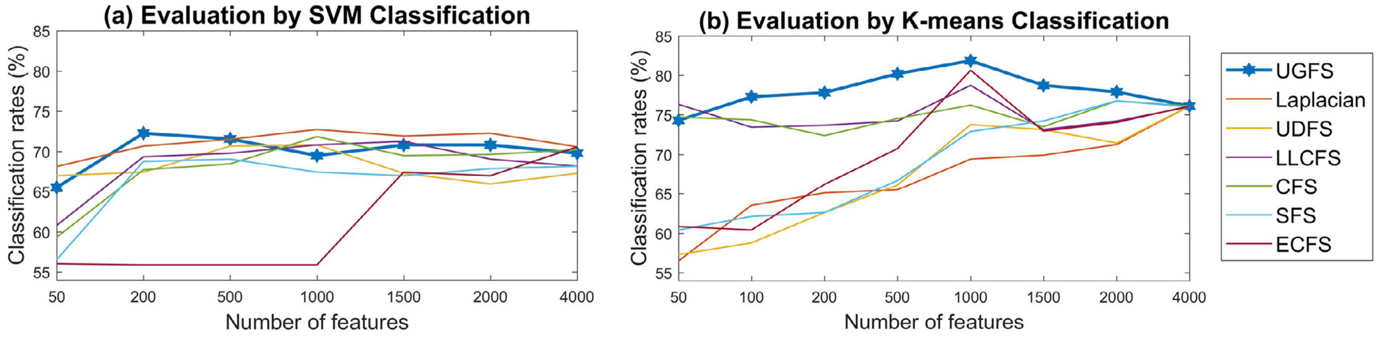

Fig. 3. Ovarian cancer dataset: correct classification rate (%) of different feature selection algorithms, over a varied number of features.

values indicated the features relevance and their ability to discriminate classes.

5.2. Comparison with feature selection methods

To validate the effectiveness of the proposed feature selection algorithm, we compare it with the following feature selection methods:

- Laplacian score: Selects features preserving the similarity of the original data (He et al., 2005).
- Unsupervised Discriminative Feature Selection (UDFS): Selects features by the local discriminative score and preserves manifold structure (Yang et al., 2011).
- Local Learning-Based Clustering Feature Selection (LLCFS): Selects features by incorporating the feature relevance evaluation into local learning-based clustering algorithm (Zeng & Cheung, 2010).
- Correlation-based Feature Selection (CFS): Selects features corresponding to the minimum pairwise correlation (Hall, 2000).
- Spectral Feature Selection (SFS): Selects features using the spectrum information of the Laplacian graph (Zhao & Liu, 2007).
- Eigenvector Centrality for Feature Selection (ECFS): Ranks features by measuring the eigenvector centrality of the pairwise features graph (Roffo & Melzi, 2017).

Note that, all these algorithms are unsupervised, except the ECFS, which analysis feature graph to rank them. We compare UGFS with ECFS in order to validate the proposed graph design.

5.3. Dataset

We are interested in data scenarios where the dimensionality of the input space is much larger than the data size, so-called High Dimension Low Sample Size (HDLSS) datasets (Zhang & Lin, 2013). Most of machine learning algorithms are less efficient when dealing with such data, which emerged these days, particularly in bioinformatics where gene/expression datasets are HDLSS. We used 4 open access datasets[2]: Colon, leukemia, ovarian cancer and CLL_SUB_111 described in Table 2.

[2] http://biogps.org/dataset/.

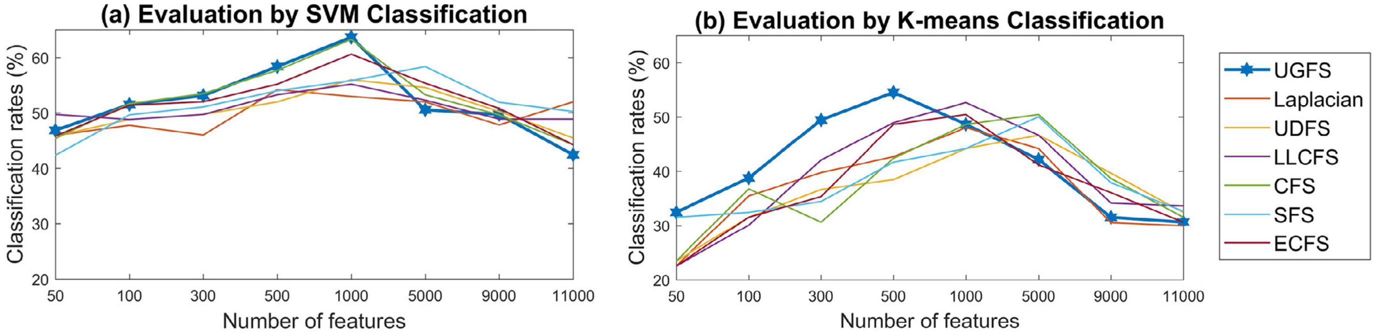

Fig. 4. CLL_sub_111 dataset: correct classification rate (%) of different feature selection algorithms, over a varied number of features.

Table 3
Comparison of different feature selection methods using SVM classifiers.

Datasets	Measures	UGFS	Laplacian	UDFS	LLCFS	CFS	SFS	ECFS
Colon	ACC(%) ± STD	**77.4 ± 3.4**	75.5 ± 6.9	75 ± 5.8	76.6 ± 8.7	76.6 ± 9.2	70.5 ± 5.4	**78.8 ± 3.8**
	RED	**0.21**	0.22	0.23	0.23	0.22	0.29	0.225
	# Features	**127**	338	270	236	320	1320	493
Leukemia	ACC(%) ± STD	**98.6 ± 4**	**98.6 ± 3.8**	97.7 ± 4.	97.3 ± 5	97.3 ± 5	96.8 ± 4.5	97.7 ± 4.5
	RED	**0.149**	0.155	0.155	**0.149**	0.155	0.155	0.155
	# Features	**137**	1284	1564	468	4987	2939	5185
Ovarian	ACC(%) ± STD	71.5 ± 2.9	72 ± 2.4	70.8 ± 7.3	71.5 ± 4.3	71.8 ± 3.7	70.4 ± 4	**78.2 ± 5.2**
	RED	0.179	0.139	0.161	0.155	0.139	0.154	0.116
	# Features	**207**	984	846	1300	785	1237	**10**
CLL_SUB	ACC(%) ± STD	**65.8 ± 4.5**	64.8 ± 4.8	65.6 ± 7.8	64.9 ± 6.4	64.6 ± 4	63.6 ± 5.4	**65.7 ± 5.8**
	RED	0.31	0.52	0.38	0.42	0.28	0.45	0.31
	# Features	**1713**	4841	1924	2631	1802	3820	2395

Table 4
Comparison of different feature selection methods using *k*-means classifiers.

Datasets	Measures	UGFS	Laplacian	UDFS	LLCFS	CFS	SFS	ECFS
Colon	ACC(%) ± STD	**80.4 ± 3**	78.7 ± 2.9	79.5 ± 2.8	78.7 ± 2.9	79.5 ± 2.9	79.7 ± 2.8	**82.35 ± 3.3**
	RED	**0.23**	0.24	0.225	0.23	0.2	0.247	0.24
	# Features	**213**	479	257	349	250	1341	575
Leukemia	ACC(%) ± STD	**91.3 ± 4.3**	88.2 ± 5.2	90.3 ± 4.2	87.7 ± 4.8	86.8 ± 4.6	79.1 ± 6.1	86.6 ± 5.4
	RED	**0.141**	0.145	0.148	0.152	0.153	0.15	0.143
	# Features	**338**	1400	486	5860	6591	5610	509
Ovarian	ACC(%) ± STD	**83.75 ± 5.2**	73.5 ± 5.7	80.7 ± 4.9	81.2 ± 4.7	79.2 ± 6.1	78.9 ± 5.9	**84.2 ± 5.9**
	RED	**0.164**	0.21	0.234	0.55	0.25	0.512	**0.175**
	# Features	**280**	548	659	1382	692	1245	**315**
CLL_SUB	ACC(%) ± STD	**51.3 ± 7.5**	55.6 ± 8.2	49.8 ± 8.8	**51.4 ± 8.4**	50.6 ± 7.1	45.6 ± 7.5	49.6 ± 8.1
	RED	**0.3**	0.49	0.375	0.4	0.25	0.47	0.34
	# Features	**1626**	4754	1894	2415	1756	3884	2045

5.4. Results and discussion

We compared the developed method (UGFS) to different feature selection methods (Laplacian Score, UDFS, LLCFS, CFS, SFS, and ECFS) using the 4 datasets. Figs. 1–4 represent the classification rate according to the number of selected features when we used an SVM (Figs. 1(a)–4(a) and a *k*−means algorithm (Figs. 1(b)–4(b)). We notice that on most of the cases the classification rate decreases as the number of feature increases. In other words, feature selection algorithms improve the accuracy of learning algorithms by using only relevant features and improve also the computational time.

The effectiveness of the UGFS method to highly score the relevant features is demonstrated for both SVM and *k*-means classifiers, where we obtain high classification rates for the firsts features (150 features for colon datasets and 500 for both leukemia and ovarian cancer). These results are confirmed in Tables 3 and 4, where we summarized the classification rate (ACC), the standard deviation (STD), the redundancy rate (RED) and the selected number of features (# features). We notice that considering the smallest number of features and the stability of the classification rate (via STD values), classifiers based on UGFS ranking obtain, generally, good classification rate and a low redundancy rate.

The ECFS algorithm, which is a supervised method, allows in most cases the best classification rate. However, it uses usually a large number of features. Therefore, it is not efficient in ranking relevant features. Table 3 depicts that ECFS allows an SVM classification rate of 78.17% using only 10 features. This result is perfect both in terms of classification rate and the number of features. However, in gene/expression data analysis, the use of only 10 genes from 4000 to describe 216 tissues is too restrictive. Therefore, comparisons of UGFS and ECFS show the efficiency of UGFS in dimensionality reducing while retaining the relevant features, which confirms the importance of the graph design in the feature ranking by centrality measure.

Note that, in this study, the considered datasets are real-world data characterized by low linear correlations between their features. This explains the small variations in the redundancy rates based essentially on linear pairwise feature correlations (Tables 3 and 4).

Table 5
Comparison of the feature selection and the classification runtime.

Algorithms	Original set	UGFS	Laplacian	UDFS	LLCFS	CFS	SFS	ECFS
Feature selection	–	1124,2 s	916,3 s	5523,5 s	4265,4 s	90,6 s	2124,5s	1515,7 s
SVM	286,5 s	14,9 s	66,2 s	192,1 s	97,7 s	50,3 s	137,4 s	248,2 s
k-means	197,4 s	10,6 s	49,8 s	135,4 s	65,3 s	37,8 s	98,1 s	171,6 s

In order to assess the differences between UGFS and other methods regarding the size of the retained features subsets (reported in Tables 3 and 4), a statistical analysis was performed by the paired samples Wilcoxon test. In all cases, the statistical analysis shows a significant difference between UGFS and other methods.

In order to investigate the implications of feature selection algorithms in terms of runtime, we have generated a big dataset in a high dimensional space (10 000 objects and 7000 features), and we have compared the runtime of the feature selection algorithms as well as the runtime of classification methods (SVM and *k*-means) for classifying the original dataset and the reduced dataset (only the features given the best classification rates are considered).

Table 5 summarizes the obtained runtime of all algorithms. First, we note that filter methods are the faster ones, for instance, the CFS algorithm, which ranks features based only on their pairwise correlation, have needed just 90,6 s to perform ranking. However, embedded methods such as UGFS and LLCFS, are slower but support classifiers (SVM and *k*-means) to speed up the classification runtime while obtaining better accuracies.

To summarize, UGFS is a graph-based method for an unsupervised feature selection, it needs only one parameter which can be estimated from data distribution. It is a multivariate method, which leads to a higher effectiveness in selecting discriminative features. However, this method is slower compared to the filter methods (univariate and less effective feature selection methods). Indeed, it is executed in an iterative processing.

6. Conclusion

This paper proposes a novel unsupervised feature selection method based on graph and subspace concepts. Features are mapped in an undirected graph using subspace learning, where data manifold structure is preserved. We used the prestigious Google's PageRank system as a centrality measure for ranking features by means of their importance and topological roles in the graph. Graph-based methods and centrality measures exploit the feature combination potential, although their effectiveness depends on the graph design. Therefore, we defined in this paper a novel feature relationships measure based on subspace learning, it linked the features which their interaction discriminated the classes. Then, PageRank assigned higher scores for the most relevant features and found the smallest feature subset guaranteeing the best precision.

Experimental results on real-world high dimension low sample size datasets demonstrate the effectiveness of our method (UGFS) against the existing unsupervised algorithms. The subsets selected by UGFS are almost the smallest, and they support classifiers to achieve higher classification rates in a lower runtime.

In the future, we plan to further investigate the following aspects of UGFS: 1) the graph direction will be considered and constraints will be added to avoid outliers and noisy data. 2) UGFS has one parameter to tune, therefore we plan to investigate the density threshold tuning and use a learning method such as 'association rules' to extract feature relationships. 3) This paper initiated the study of feature relationships in terms of their relevance, unlike traditional methods which considered the features redundancy. This allows the future consideration of advanced feature relationship measures. 4) For UGFS applications in intelligent systems, it can benefit from the domain knowledge, for instance, the use of ontologies knowledge in the graph design. 5) The use of the UGFS algorithm with state-of-the-art classifiers such as the convolutional neural network. These methods require a large database, however, this limitation could be overcome using transfer learning.

Acknowledgements

This research was supported in part by the program MITACS, accelerate FR20930 and by the Canada research chair on biomedical data mining (950–231214).

References

Abdi, H., & Williams, L. J. (2010). Principal component analysis. *Wiley Interdisciplinary Reviews: Computational Statistics, 2*(4), 433–459.

Aliyari, G. Y., Rudzicz, F., & Moghaddam, H. A. (2015). Fast incremental LDA feature extraction. *Pattern Recognition, 48*(6), 1999–2012.

Avrachenkov, K., Kadavankandy, A., Prokhorenkova, L. O., & Raigorodskii, A. (2015). Pagerank in undirected random graphs. In *Lecture notes in computer science (including subseries lecture notes in artificial intelligence & lecture notes in bioinformatics), 9479* (pp. 151–163).

Bennasar, M., Hicks, Y., & Setchi, R. (2015). Feature selection using joint mutual information maximisation. *Expert Systems with Applications, 42*(22), 8520–8532.

Böhm, C., Kailing, K., Kriegel, H. P., & Kroger, P. (2004). Density connected clustering with local subspace preferences. In *Proceedings of the fourth IEEE international conference on data mining, ICDM '04, washington, DC, USA* (pp. 27–34).

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 & data mining - KDD '10* (pp. 333–342).

Celebi, M. E., Kingravi, H. A., & Vela, P. A. (2013). A comparative study of efficient initialization methods for the k-means clustering algorithm. *Expert Systems with Applications, 40*(1), 200–210.

Choi, H., & Choi, S. (2007). Robust kernel isomap. *Pattern Recognition, 40*(3), 853–862.

Cortes, C., & Vapnik, V. (1995). Support vector machine. *Machine Learning, 20*(3), 1303–1308.

Du, S., Ma, Y., Li, S., & Ma, Y. (2017). Robust unsupervised feature selection via matrix factorization. *Neurocomputing, 241*, 115–127. (C)

Duda, R. O., Hart, P. E., & Stork, D. G. (2001). *Pattern classification* (2nd ed.). New York, USA.: Wiley-Interscience.

Elhamifar, E., & Vidal, R. (2013). Sparse subspace clustering: Algorithm, theory, & applications. *IEEE Transactions on Pattern Analysis & Machine Intelligence, 35*(11), 2765–2781.

Flexer, A., & Schnitzer, D. (2015). Choosing ℓ_p norms in high-dimensional spaces based on hub analysis. *Neurocomputing, 169*, 281–287.

Gleich, D. F. (2015). Pagerank beyond the web. *Society for Industrial & Applied Mathematics, 57*(3), 321–363.

Hall, M. A. (2000). Correlation-based feature selection for discrete & numeric class machine learning. In *Proceedings of the seventeenth international conference on machine learning, ICML '00, San Francisco, CA, USA* (pp. 359–366).

Han, D., & Kim, J. (2015). Unsupervised simultaneous orthogonal basis clustering feature selection. In *Proceedings of the IEEE computer society conference on computer vision & pattern recognition* (pp. 5016–5023).

Hassani, M., Kim, Y., Choi, S., & Seidl, T. (2014). Subspace clustering of data streams: New algorithms & effective evaluation measures. *Journal of Intelligent Information Systems, 45*(3), 319–335.

He, X., Cai, D., & Niyogi, P. (2005). Laplacian score for feature selection. In *Proceedings of the 18th international conference on neural information processing systems, NIPS'05, Vancouver, BC, Canada* (pp. 507–514).

Hu, L., Gao, W., Zhao, K., Zhang, P., & Wang, F. (2018). Feature selection considering two types of feature relevancy & feature interdependency. *Expert Systems with Applications, 93*, 423–434. (C)

Krzanowski, W. J. (1987). Selection of variables to preserve multivariate data structure using principal components. *Journal of the Royal Statistical Society. Series C (Applied Statistics), 36*(1), 22–33.

Li, Y., Hung, E., & Chung, K. (2011). A subspace decision cluster classifier for text classification. *Expert Systems with Applications, 38*(10), 12475–12482.

Ma, N., & Liu, Y. (2014). Superedgerank algorithm & its application in identifying opinion leader of online public opinion supernetwork. *Expert Systems with Applications, 41*, 1357–1368. (4, Part 1)

Martinez-Gonzalez, B., Pardo, J. M., Echeverry-Correa, J. D., & San-Segundo, R. (2017). Spatial features selection for unsupervised speaker segmentation and clustering. *Expert Systems with Applications, 73*, 27–42.

Moradi, P., & Rostami, M. (2015). A graph theoretic approach for unsupervised feature selection. *Engineering Applications of Artificial Intelligence, 44*, 33–45. (Supplement C)

Morrison, J. L., Breitling, R., Higham, D. J., & Gilbert, D. R. (2005). Generank: Using search engine technology for the analysis of microarray experiments. *BMC bioinformatics, 6*(1), 233.

Nie, F., Xiang, S., Jia, Y., Zhang, C., & Yan, S. (2008). Trace ratio criterion for feature selection. In *Proceeding of the twenty-third AAAI conference on artificial intelligence, AAAI'08, Chicago, Illinois, USA* (pp. 671–676).

Opsahl, T., Agneessens, F., & Skvoretz, J. (2010). Node centrality in weighted networks: Generalizing degree & shortest paths. *Social Networks, 32*(3), 245–251.

Parsons, L., Haque, E., & Liu, H. (2004). Subspace clustering for high dimensional data. *ACM SIGKDD Explorations Newsletter, 6*(1), 90–105.

Roffo, G., & Melzi, S. (2017). Ranking to learn: Feature ranking & selection via eigenvector centrality. In *New frontiers in mining complex patterns* (pp. 1–15).

Somol, P., Baesens, B., Pudil, P., & Vanthienen, J. (2005). Filter- versus wrapper-based feature selection for credit scoring. *International Journal of Intelligent Systems, 20*(10), 985–999.

Vidal, R. (2011). Subspace clustering. *IEEE Signal Processing Magazine, 28*(2), 52–68.

Wu, G., Zhang, Y., & Wei, Y. (2013). Accelerating the arnoldi-type algorithm for the pagerank problem & the proteinrank problem. *Journal of Scientific Computing, 57*(1), 74–104.

Yang, Y., Shen, H. T., Ma, Z., Huang, Z., & Zhou, X. (2011). $\ell_{2,1}$-Norm regularized discriminative feature selection for unsupervised learning. In *IJCAI international joint conference on artificial intelligence* (pp. 1589–1594).

Zeng, H., & Cheung, Y. M. (2010). Feature selection & kernel learning for local learning based clustering. *IEEE Transactions on Pattern Analysis & Machine Intelligence, 33*(8), 1532–1547.

Zhang, H., Li, F., Liu, P., Chen, Y., Ren, D., & Wang, K. (2017). How can a sparse representation be made applicable for very low-dimensional data? *Expert Systems with Applications, 77*, 66–70. (c)

Zhang, H., Lofgren, P., & Goel, A. (2016). Approximate personalized pagerank on dynamic graphs. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery & data mining, KDD '16* (pp. 1315–1324).

Zhang, L., & Lin, X. (2013). Some considerations of classification for high dimension low-sample size data. *Statistical Methods in Medical Research, 22*(5), 537–550.

Zhao, Z., & Liu, H. (2007). Spectral feature selection for supervised & unsupervised learning. In *Proceedings of the 24th international conference on machine learning - ICML '07* (pp. 1151–1157).

Zhao, Z., Wang, L., Liu, H., & Ye, J. (2013). On similarity preserving feature selection. *IEEE Transactions on Knowledge & Data Engineering, 25*(3), 619–632.

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).

Zhu, P., Zuo, W., Zhang, L., Hu, Q., & Shiu, S. C. (2015). Unsupervised feature selection by regularized self-representation. *Pattern Recognition, 48*(2), 438–446.