

Unsupervised Graph-based Feature Selection via Subspace and PageRank centrality

K. Henni^{1,2}, N. Mezghani^{1,2}, C. Gouin-Vallerand¹

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.

Keywords: Unsupervised Feature Selection, Graph Centrality Measure, PageRank, Subspace Learning, Projected Densities, K-nearest neighbors.

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 et al., 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 et al., 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,

Email addresses: khadidja.henni@liceftelug.ca (K. Henni), neila.mezghani@telug.ca (N. Mezghani), charles.gouin-vallerand@telug.ca (C. Gouin-Vallerand)

¹LICEF Research Center, TELUQ university, 5800 Rue St-Denis, Montréal, Qc, Canada.

²Laboratoire de recherche en imagerie et orthopédie (LIO), Centre de recherche du CHUM, 900 Rue St-Denis, Montréal, QC, Canada.

15 various methods have been proposed in the literature, such as (1) feature extraction, where a feature space is computed
16 based on the combination of the original features (Abdi & Williams, 2010; Aliyari et al., 2015; Choi & Choi, 2007),
17 (2) feature selection, where a subset of relevant and less redundant features are selected or ranked respecting their
18 relevance order (Bennasar et al., 2015; Hall, 2000; Hu et al., 2018) and (3) subspace and projected learning, where a
19 subset of relevant features for a given clusters or data instance is selected/weighted and used in learning simultaneously
20 (Elhamifar & Vidal, 2013; Parsons et al., 2004; Vidal, 2011). Subspace learning is currently introduced in several
21 data-mining techniques, especially in data stream analysis where computational time and costs reduction are crucial
22 (Hassani et al., 2014).

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

28 Feature selection algorithms can be categorized into (1) supervised/unsupervised methods according on whether
29 the data are labeled or not, (2) filter/wrapper/embedded methods according to the degree of learning involvement
30 or (3) univariate/multivariate according to the consideration of the features interaction potential. The well-known
31 unsupervised feature selection algorithms, such as Laplacian Score (He et al., 2005), Spectral Feature Selection (SFS)
32 (Zhao & Liu, 2007), Multi-Cluster Feature Selection (MCFS) (Cai et al., 2010), Minimum Redundancy Spectral
33 Feature selection (MRSF) (Zheng et al., 2010), characterize the manifold structure by graphs where nodes are the data
34 instances. The Laplacian Score and SPFS use metrics to rank features, while MCFS and MRSF rank features based
35 on a multi-output sparse regression. These methods rank features by capturing the manifold structure in a given graph.
36 Thus, their efficiency depends strongly on the instances graph design. Unlike the previous graph-based methods, the
37 supervised EigenVector Centrality for Feature Selection (ECFS), maps features into a graph and ranks them by the
38 Eigenvector centrality measure (Roffo & Melzi., 2017). The graph design proposed by ECFS is based on pairwise
39 relationships between features and some basic statistical metrics to define discriminative features (mutual information,
40 Fisher score, and the standard deviation). Hence, it neglects the manifold structure preservation and does not exploit
41 the features combination potential. Moradi & Rostami. (2015) represented the set of features by a weighted graph,
42 where features similarities, measured by Pearson product-moment correlation coefficient, are graph edges. Then,
43 investigated the Louvain community detection algorithm to identify the feature clusters. Finally, a centrality measure
44 is proposed to filter and rank features. This graph-based method demonstrated competitive results. Nevertheless, it
45 is slow and addressed more feature redundancy than relevance. Despite the centrality measures popularity in graph
46 theory and their efficiency in scoring and ranking nodes according to their topological importance and roles within the
47 graph, the ranking still depends on the graph design.

48 In this research, we propose a new unsupervised feature selection method called '*Unsupervised Graph-based Fea-*
49 *ture Selection*' (UGFS), which outputs the features ranking vector. We investigated the Google's PageRank centrality

50 measure (Gleich, 2015), to analysis feature graph structure and attribute to each feature an importance score. We also
51 addressed the problem of defining the relationships between features, in order to establish the feature graph structure.
52 This graph is designed by means of the ‘subspace preference clusters’ concept, which is driven from subspace learning
53 and supports the PageRank to highly score the relevant features for classification problems.

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

57 2. Related Work

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

63 2.1. Unsupervised feature selection algorithms

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

70 Several levels of embedded methods have been proposed, which differ in terms of the used learning algorithm and
71 in which step it is used. TraceRatio (Nie et al., 2008) and Unsupervised Discriminative Feature Selection (UDFS)
72 (Yang et al., 2011) are the simplest embedded algorithms. They capture the manifold structure of data by performing a
73 fit learning to highly score the most discriminative features. Nevertheless, these algorithms present some limitations.
74 Indeed, TraceRatio generates redundant features and UDFS uses restrictive constraints.

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

81 Data sparsity in high-dimensional spaces reduced the impact of the pairwise similarity between samples to dis-
82 criminate classes. Thereby, the sparse representation studies (Zhang et al., 2017) emerged and where the of $\ell_{2,1}$ -norm

83 demonstrated high learning performances on those spaces. This norm has been implemented in recent embedded fea-
 84 ture selection methods. The purpose of these later consists on the minimization of the $\ell_{2,1}$ -norm based on regression
 85 learning, where (1) The Regularized Self-Representation method (RSR) minimizes of the error between the projected
 86 data and the target matrix (Zhu et al., 2015), (2) the Simultaneous Orthogonal basis Clustering Decomposition Feature
 87 Selection (SOFCS), decomposes the target matrix based on orthogonal constraints (Han & Kim, 2015) and (3) Ro-
 88 bust Unsupervised Feature Selection via Matrix Factorization (RUFMS) combines the feature selection with matrix
 89 factorization and manifold regularization into unified framework (Du et al., 2017).

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

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

93 2.2. Subspace and projected algorithms

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

98 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 char-
 99 acterizing features of the considered cluster. CLIQUE is a subspace clustering algorithm based on grid (Böhm et al.,
 100 2004). Based on an Apriori-like method, it recursively searches the set of all possible subspaces. It used a density

101 threshold to filter cells. Based on conclusions given by Flexer & Schnitzer (2015) which demonstrated that Euclidian
 102 distances are not efficient, Böhm et al. (2004) proposed a weighted Euclidian distance based on subspace concepts
 103 and the well-founded notion of density connected clusters. Authors proposed the use of subspace preference cluster
 104 concepts, based on the variance of the data neighborhood along features, then weighted the Euclidian distance by
 105 these variances (more details are given in Section 3).

106 2.3. Graph centrality measures

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

120 3. Mathematical Framework

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

122 3.1. Notations

123 Let X be a set of n points and d -dimensional features $X = \{\mathbf{x}_1, \dots, \mathbf{x}_n\}$. Each point \mathbf{x}_i is a vector of d features
 124 $\mathbf{x}_i = (x_i^1, \dots, x_i^d)$. $dist(\mathbf{x}_p, \mathbf{x}_q)$ is the Euclidean distance between two data points $\mathbf{x}_p, \mathbf{x}_q \in X$ and the $dist_p : X \times X \rightarrow \mathbb{R}$
 125 is a metric distance function between projected points.

126 Our aim is to develop a new feature selection algorithm which maps features into an undirected graph. Let
 127 $G = \langle V, E \rangle$ a graph, where the vertices (nodes) V are the set of features $V = \{\mathbf{x}^1, \dots, \mathbf{x}^d\}$, and E the edges linking
 128 the vertices. A is the adjacency matrix associated to the graph G , where each of its element $a_{i,j}$ represents a pairwise
 129 relationship between features \mathbf{x}^i and \mathbf{x}^j . $a_{i,j}$ is associated to a potential function $\phi(\mathbf{x}^i, \mathbf{x}^j)$:

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

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

131 3.2. Subspace preference clusters

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

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

138 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:

$$var_i(NN_k(\mathbf{x}_p)) = \frac{\sum_{\mathbf{x}_q \in NN_k(\mathbf{x}_p)} (dist_p(x_p^i, x_q^i))^2}{|NN_k(\mathbf{x}_p)|} \quad (2)$$

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

140 The feature subspace preference associated to the data point $\mathbf{x}_p \in X$ is the set of features with $var_i(NN_k(\mathbf{x}_p)) \leq \delta$,
141 $\delta \in \mathbb{R}, k \in \mathbb{N}$. This features set preserves the density in the neighborhood of the point \mathbf{x}_p . Therefore, if the selected
142 set of features (subspace preference cluster) has low variance in the neighborhood of points, thus those features are
143 relevant and preserve the density inside the cluster of the data point \mathbf{x}_p .

144 3.3. PageRank

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

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

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

152 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
153 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
154 graph and its value for a given node is computed iteratively based on PageRank of nodes pointing on it. In order to
155 deal with undirected graphs, some variants of PageRank have been proposed (Avrachenkov et al., 2015; Zhang et al.,
156 2016). In our study, we used the basic version of the algorithm. The PageRank vector is a stationary distribution of
157 special formed Markov process, more details about its convergence are given in (Gleich, 2015).

158 4. Unsupervised Graph-based Feature Selection method ‘UGFS’

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

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

$$\phi(\mathbf{x}^i, \mathbf{x}^j) = \begin{cases} 1, & \text{if } \text{var}_{S_p}(NN_k(\mathbf{x}_p)) \leq \delta \\ 0, & \text{otherwise} \end{cases} \quad (4)$$

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

173 More details about the definition of feature relationships and graph design are given in algorithm 1³.

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

176 5. Experimental Results

177 5.1. Experimental Setup

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

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

³The source code will be posted on line to provide the needed material for the use of UGFS.

Algorithm 1 : Feature Graph design

Input: Observed data $X = \{\mathbf{x}_1, \dots, \mathbf{x}_n\}$, k , δ .

Output: G undirected graph of features.

```
1: Compute  $NN_k(\mathbf{x}_i)$ , with  $i = 1, \dots, n$ .
2: Compute  $Var_j(NN_k(\mathbf{x}_i))$ , with  $i = 1, \dots, n$  and  $j = 1, \dots, d$  (see section 3.2; equation 2).
3:  $A(i, j) = 0$ ,  $i = 1, \dots, d$  and  $j = 1, \dots, d$ .
4: for  $i = 1 : n$  do
5:   for  $j = 1 : d$  do
6:     if  $Var_j(NN_k(\mathbf{x}_i)) \leq \delta$  then
7:        $Var_{Binarized}(i, j) = 1$ 
8:     else
9:        $Var_{Binarized}(i, j) = 0$ 
10:    end if
11:  end for
12: end for
13: for  $i = 1 : n$  do
14:    $S_p = \{\}$ 
15:   for  $j = 1 : d$  do
16:     if  $Var_{Binarized}(i, j) == 1$  then
17:        $S_p = S_p \cup \{\mathbf{x}^j\}$ 
18:     end if
19:   end for
20:   for  $l = 1 : size(S_p)$  do
21:     for  $m = 1 : size(S_p)$  do
22:        $A(l, m) = 1$ 
23:     end for
24:   end for
25: end for
26:  $G = ((\mathbf{x}^1, \dots, \mathbf{x}^d), A)$ 
```

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

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

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

188 We use two classifiers: (1) the support vector machine (SVM) for supervised classification (Cortes & Vapnik,
189 1995), which is widely used both in feature selection algorithm design and/or evaluation and (2) the k -means for
190 unsupervised learning (Celebi et al., 2013), which is simple, fast and requires only the number of clusters as input
191 parameter. k -means initial centroids choice influences highly its accuracy, that is why we use k -means++ algorithm
192 to choose the centroids initial values.

193 Best feature ranking is then demonstrated by minimization of the evaluation criteria, except the classification rate
194 where higher values indicated the features relevance and their ability to discriminate classes.

195 5.2. Comparison with feature selection methods

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

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

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

211 5.3. Dataset

212 We are interested in data scenarios where the dimensionality of the input space is much larger than the data size,
213 so-called High Dimension Low Sample Size (HDLSS) datasets (Zhang & Lin, 2013). Most of machine learning
214 algorithms are less efficient when dealing with such data, which emerged these days, particularly in bioinformatics
215 where gene/expression datasets are HDLSS. We used 4 open access datasets⁴: Colon, leukemia, ovarian cancer and
216 CLL_SUB_111 described in Table 2.

⁴<http://biogps.org/dataset/>

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	11340	111	3

217 5.4. Results and discussion

218 We compared the developed method (UGFS) to different feature selection methods (Laplacian Score, UDFS,
 219 LLCFS, CFS, SFS, and ECFS) using the 4 datasets. Figure 1, 2, 3 and 4 represent the classification rate according to
 220 the number of selected features, when we used an SVM (Figure 1.(a), 2.(a), 3.(a) and 4.(a) and a k -means algorithm
 221 (Figure 1.(b), 2.(b), 3.(b) and 4.(b)). We notice that on most of cases the classification rate decreases as the number of
 222 feature increases. In other words, feature selection algorithms improve the accuracy of learning algorithms by using
 223 only relevant features and improve also the computational time.

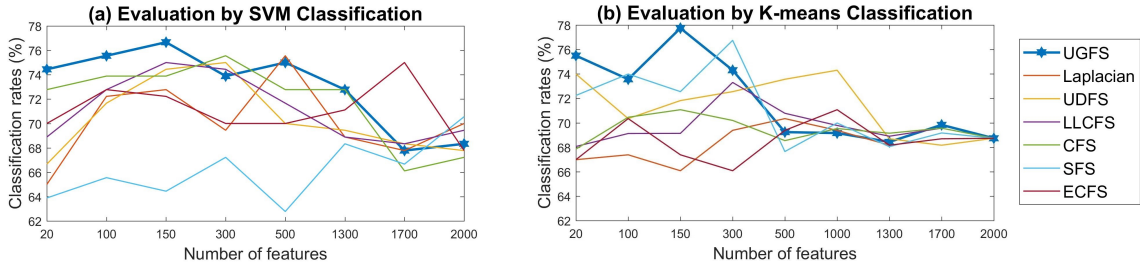


Figure 1: Colon dataset: correct classification rate (%) of different feature selection algorithms, over a varied number of features.

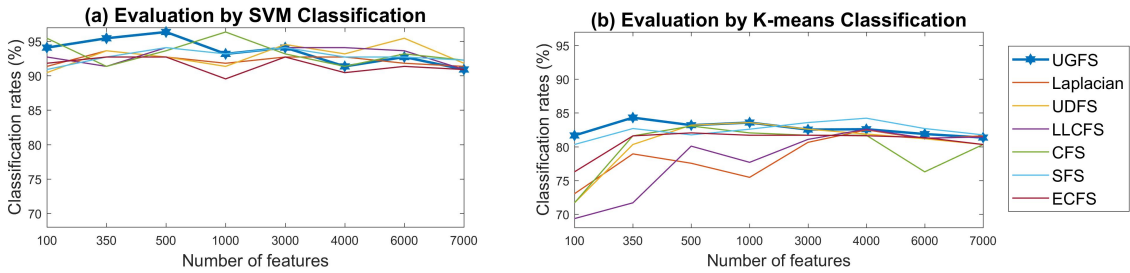


Figure 2: leukemia dataset: correct classification rate (%) of different feature selection algorithms, over a varied number of features.

224 The effectiveness of the UGFS method to highly score the relevant features is demonstrated for both SVM and
 225 k -means classifiers, where we obtain high classification rates for the firsts features (150 features for colon datasets and
 226 500 for both leukemia and ovarian cancer). These results are confirmed in Table 3 and 4, where we summarized the
 227 classification rate (ACC), the standard deviation (STD), the redundancy rate (RED) and the selected number of features

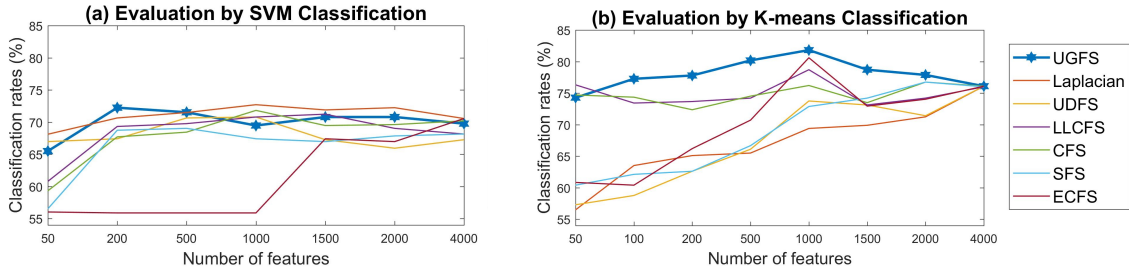


Figure 3: Ovarian cancer dataset: correct classification rate (%) of different feature selection algorithms, over a varied number of features.

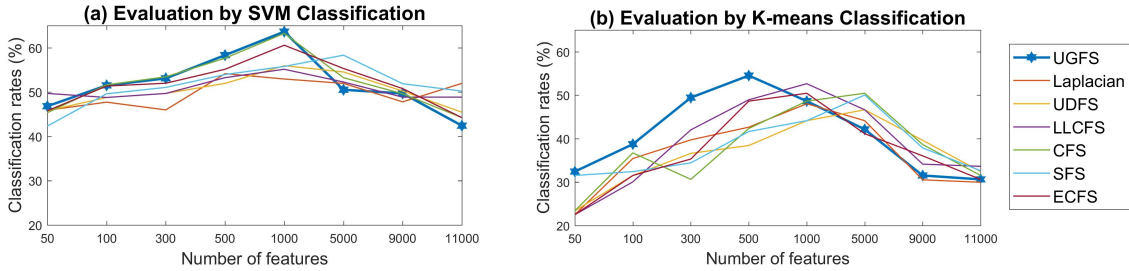


Figure 4: CLL_sub_111 dataset: correct classification rate (%) of different feature selection algorithms, over a varied number of features.

228 (# features). We notice that considering the smallest number of features and the stability of the classification rate (via
 229 STD values), classifiers based on UGFS ranking obtain, generally, good classification rate and a low redundancy rate.

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

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

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

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

243 In order to investigate the implications of feature selection algorithms in terms of runtime, we have generated a
244 big dataset in a high dimensional space (10000 objects and 7000 features), and we have compared the runtime of the
245 feature selection algorithms as well as the runtime of classification methods (SVM and k -means) for classifying the
246 original dataset and the reduced dataset (only the features given the best classification rates are considered).

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

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

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

255 6. Conclusion

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

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

267 In the future, we plan to further investigate the following aspects of UGFS: 1) the graph direction will be consid-
 268 ered and constraints will be added to avoid outliers and noisy data. 2) UGFS has one parameter to tune, therefore we
 269 plan to investigate the density threshold tuning and use a learning method such as ‘association rules’ to extract feature
 270 relationships. 3) This paper initiated the study of feature relationships in terms of their relevance, unlike traditional
 271 methods which considered the features redundancy. This allows the future consideration of advanced feature rela-
 272 tionship measures. 4) For UGFS applications in intelligent systems, it can benefit from the domain knowledge, for
 273 instance, the use of ontologies knowledge in the graph design. 5) The use of the UGFS algorithm with state-of-the-art

274 classifiers such as the convolutional neural network. These methods require a large database, however, this limitation
275 could be overcome using transfer learning.

276 7. Acknowledgements

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

279 References

- 280 Abdi, H. & Williams, L. J. (2010). Principal component analysis. *Wiley Interdisciplinary Reviews: Computational Statistics*, 2(4), 433–459.
- 281 Aliyari, G. Y., Rudzicz, F., & Moghaddam, H. A. (2015). Fast incremental LDA feature extraction. *Pattern Recognition*, 48(6), 1999–2012.
- 282 Avrachenkov, K., Kadavankandy, A., Prokhorenkova, L. O., & Raigorodskii, A. (2015). Pagerank in undirected random graphs. In *Lecture Notes
283 in Computer Science (including subseries Lecture Notes in Artificial Intelligence & Lecture Notes in Bioinformatics)*, 9479, 151–163.
- 284 Bannasar, M., Hicks, Y., & Setchi, R. (2015). Feature selection using joint mutual information maximisation. *Expert Systems with Applications*,
285 42(22), 8520–8532.
- 286 Böhm, C., Kailing, K., Kriegel, H.-P., & Kroger, P. (2004). Density connected clustering with local subspace preferences. In *Proceedings of the
287 Fourth IEEE International Conference on Data Mining, ICDM '04, Washington, DC, USA* (pp. 27–34).
- 288 Cai, D., Zhang, C., & He, X. (2010). Unsupervised feature selection for multi-cluster data. In *Proceedings of the 16th ACM SIGKDD international
289 conference on Knowledge discovery & data mining - KDD '10* (pp. 333–342).
- 290 Celebi, M. E., Kingravi, H. A., & Vela, P. A. (2013). A comparative study of efficient initialization methods for the k-means clustering algorithm.
291 *Expert Systems with Applications*, 40(1), 200–210.
- 292 Choi, H. & Choi, S. (2007). Robust kernel Isomap. *Pattern Recognition*, 40(3), 853–862.
- 293 Cortes, C., & Vapnik, V. (1995). Support vector machine. *Machine learning*, 20(3), 1303–1308.
- 294 Du, S., Ma, Y., Li, S., & Ma, Y. (2017). Robust unsupervised feature selection via matrix factorization. *Neurocomputing*, 241(C), 115–127.
- 295 Duda, R. O., Hart, P. E., & Stork, D. G. (2001). Pattern classification. 2nd. Edition. *Wiley-Interscience*, New York, USA.
- 296 Elhamifar, E., & Vidal, R. (2013). Sparse Subspace Clustering: Algorithm, Theory, & Applications. *IEEE Transactions on Pattern Analysis &
297 Machine Intelligence*, 35(11), 2765–2781.
- 298 Flexer, A., & Schnitzer, D. (2015). Choosing ℓ_p norms in high-dimensional spaces based on hub analysis. *Neurocomputing*, 169, 281–287.
- 299 Gleich, D. F. (2015). PageRank beyond the Web. *Society for Industrial & Applied Mathematics*, 57(3), 321–363.
- 300 Hall, M. A. (2000). Correlation-based feature selection for discrete & numeric class machine learning. In *Proceedings of the Seventeenth Interna-
301 tional Conference on Machine Learning, ICML '00, San Francisco, CA, USA* (pp. 359–366).
- 302 Han, D. & Kim, J. (2015). Unsupervised Simultaneous Orthogonal basis Clustering Feature Selection. In *Proceedings of the IEEE Computer
303 Society Conference on Computer Vision & Pattern Recognition* (pp. 5016–5023).
- 304 Hassani, M., Kim, Y., Choi, S., & Seidl, T. (2014). Subspace clustering of data streams: new algorithms & effective evaluation measures. *Journal
305 of Intelligent Information Systems*, 45(3), 319–335.
- 306 He, X., Cai, D., & Niyogi, P. (2005). Laplacian Score for Feature Selection. In *Proceedings of the 18th International Conference on Neural
307 Information Processing Systems, NIPS'05, Vancouver, BC, Canada* (pp. 507–514).
- 308 Hu, L., Gao, W., Zhao, K., Zhang, P., & Wang, F. (2018). Feature selection considering two types of feature relevancy & feature interdependency.
309 *Expert Systems with Applications*, 93(C), 423–434.
- 310 Krzanowski, W. J. (1987). Selection of variables to preserve multivariate data structure using principal components. *Journal of the Royal Statistical
311 Society. Series C (Applied Statistics)*, 36(1), 22–33.

312 Li, Y., Hung, E., & Chung, K. (2011). A subspace decision cluster classifier for text classification. *Expert Systems with Applications*, 38(10),
313 12475–12482.

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

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

318 Moradi, P. & Rostami, M. (2015). A graph theoretic approach for unsupervised feature selection. *Engineering Applications of Artificial Intelligence*,
319 44(Supplement C), 33–45.

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

322 Nie, F., Xiang, S., Jia, Y., Zhang, C., & Yan, S. (2008). Trace Ratio Criterion for Feature Selection. In *Proceeding of the Twenty-Third AAAI*
323 *Conference on Artificial Intelligence, AAAI'08, Chicago, Illinois, USA* (pp. 671–676).

324 Opsahl, T., Agneessens, F., & Skvoretz, J. (2010). Node centrality in weighted networks: Generalizing degree & shortest paths. *Social Networks*,
325 32(3), 245–251.

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

327 Roffo, G. & Melzi, S. (2017). Ranking to Learn: Feature Ranking & Selection via Eigenvector Centrality. In *New Frontiers in Mining Complex*
328 *Patterns* (pp. 1–15).

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

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

332 Wu, G., Zhang, Y., & Wei, Y. (2013). Accelerating the Arnoldi-type algorithm for the PageRank problem & the ProteinRank problem. *Journal of*
333 *Scientific Computing*, 57(1), 74–104.

334 Yang, Y., Shen, H. T., Ma, Z., Huang, Z., & Zhou, X. (2011). $\ell_{2,1}$ -Norm regularized discriminative feature selection for unsupervised learning. In
335 *IJCAI International Joint Conference on Artificial Intelligence* (pp. 1589–1594).

336 Zeng, H. & Cheung, Y.-M. (2010). Feature Selection & Kernel Learning for Local Learning Based Clustering. *IEEE transactions on pattern*
337 *analysis & machine intelligence*, 33(8), 1532–1547.

338 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
339 data?. *Expert Systems with Applications*, 77(c), 66–70.

340 Zhang, H., Lofgren, P., & Goel, A. (2016). Approximate Personalized PageRank on Dynamic Graphs. In *Proceedings of the 22nd ACM SIGKDD*
341 *International Conference on Knowledge Discovery & Data Mining, KDD '16* (pp. 1315–1324).

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

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

346 Zhao, Z., Wang, L., Liu, H., & Ye, J. (2013). On similarity preserving feature selection. *IEEE Transactions on Knowledge & Data Engineering*,
347 25(3), 619–632.

348 Zheng, Z., Lei, W., & Huan, L. (2010). Efficient Spectral Feature Selection with Minimum Redundancy. In *Twenty-Fourth AAAI Conference on*
349 *Artificial Intelligence* (pp. 1–6).

350 Zhu, P., Zuo, W., Zhang, L., Hu, Q., & Shiu, S. C. (2015). Unsupervised feature selection by regularized self-representation. *Pattern Recognition*,
351 48(2), 438–446.