**Computing manuscript No.**
(will be inserted by the editor)

# Feature selection based on community detection in feature correlation networks

**Miloš Savić · Vladimir Kurbalija · Zoran Bosnić · Mirjana Ivanović**

Received: date / Accepted: date

**Abstract** Feature selection is an important data preprocessing step in data mining and machine learning tasks, especially in the case of high dimensional data. In this paper, we propose a novel feature selection method based on feature correlation networks, i.e. complex weighted networks describing the strongest correlations among features in a dataset. The method utilizes community detection techniques to identify cohesive groups of features in feature correlation networks. A subset of features exhibiting a strong association with the class variable is selected according to the identified community structure taking into account the size of feature communities and connections within them. The proposed method is experimentally evaluated on a high dimensional dataset containing signaling protein features related to the diagnosis of Alzheimer's disease. We compared the performance of seven commonly used classifiers that were trained without feature selection, after feature selection by four variants of our method determined by different community detection techniques, and after feature selection by four widely used state-of-the-art feature selection methods available in the WEKA machine learning library. The results of the experimental evaluation indicate that our method improves the classification accuracy of several classification models while greatly reducing the dimensionality of the dataset. Additionally, our method tends to outperform traditional feature selection methods provided by the WEKA library.

**Keywords** feature selection · feature correlation networks · community detection · Alzheimer's disease

Miloš Savić (corresponding author) · Vladimir Kurbalija · Mirjana Ivanović
University of Novi Sad, Faculty of Sciences, Department of Mathematics and Informatics
Trg Dositeja Obradovića 4, 21000 Novi Sad, Serbia
E-mail: {svc, kurba, mira}@dmi.uns.ac.rs

Zoran Bosnić
Univeristy of Ljubljana, Faculty of Computer and Information Science
Večna pot 113, SI-1000 Ljubljana, Slovenia
E-mail: zoran.bosnic@fri.uni-lj.si

This paper is an extended version of our paper [30] presented at the 7th International Conference on Model and Data Engineering (MEDI 2017).

## 1 Introduction

The feature selection problem has been studied by data mining and machine learning researchers for many years. The main aim of feature selection is to reduce the dimensionality of a dataset such that the most significant aspects of the data are represented by selected features. Consequently, feature selection has become an important data preprocessing step in data mining and machine learning tasks due to the rise of high dimensional data in many application domains. Feature selection usually leads to better machine learning models in terms of prediction accuracy, lower training time and model comprehensibility [36]. The two most dominant types of feature selection approaches are filter and wrapper methods [9,16]. Wrapper methods rely on the performance of a prespecified classification model to evaluate the quality of selected features. In contrast to wrapper methods, filter methods do not utilize classification learning algorithms' performance to select features. Those methods are usually based on some efficiently computable measure for scoring features considering their redundancy, dependency and discriminative power.

In this paper we present a novel graph-based approach to feature selection. Our feature selection approach belongs to the class of filter-based methods. The main idea of the proposed approach is to select relevant features considering the community structure of *feature correlation networks*. A feature correlation network is a weighted graph where nodes correspond to features and links represent the strongest correlations among them. Feature correlation networks used in our feature selection method are conceptually similar to weighted correlation networks used in the analysis of genomic datasets [11,17] with one important difference: a class variable (a special feature denoting the class of a data instance) is not represented by a node in the corresponding feature correlation network, but to each node in the feature feature correlation network is associated a number which specifies the strength of association between the corresponding feature and the class variable.

A *community* (cluster, module or cohesive group) of a weighted network is a subset of nodes such that links within the community tend to be significantly stronger than links connecting nodes from the community with the rest of the network [20]. We say that a network has a community structure if the set of nodes can be partitioned into communities. The existence of communities is a typical feature of complex networks from various domains [2,19,28]. Their automatic identification is enabled by various community detection techniques proposed in the literature [7,29]. The identification of communities in a complex network enables us to study its structure on a higher level of abstraction by constructing and analyzing its coarse-grained description (the networks of communities). Our feature selection approach relies on community detection techniques to identify communities of features such that correlations within a feature community are stronger than correlations between features belonging to different communities. Then, one or more features strongly associated to the class variable are selected to represent each identified community taking into account the number of nodes and connections within feature communities.

The rest of the paper is structured as follows. Related work is presented in Section 2. The proposed method for feature selection is described in Section 3.

The evaluation of the method is given in Section 4. The last section concludes the paper and gives directions for possible future work.

## 2 Related Work

Feature selection is a common data mining preprocessing step, which aims at reducing the dimensionality of the original dataset. Adequate selection of features has numerous advantages [27] such as simplification of learning models, improving the performance of learning algorithms, data reduction (avoidance of the curse of dimensionality), improved generalization by reducing overfitting, and so on.

Wrapper-based feature selection methods estimate usefulness of features using the selected learning algorithm. These methods usually give better results than filter methods since they are adapting their result to a chosen learning algorithm. However, since a learning algorithm is employed to evaluate each subset of features, wrapper methods are very time consuming and almost unusable for high dimensional data. Furthermore, since the feature selection process is tightly interconnected with a learning algorithm, wrappers are less general than filters and have the increased risk of overfitting. On the other hand, filter methods do not utilize the learning algorithm in the feature selection process. They are commonly based on scoring metrics such as the correlation with the variable to predict. These methods are generally many times faster than wrappers and robust to overfitting [10]. Recently, some embedded methods are introduced [15] which try combine the positive characteristics of both previous methods.

Relying on the characteristics of data, filter models evaluate features without utilizing any classification algorithms. Usually, a filter algorithm has two steps: it ranks features based on certain criteria and then it selects the features with highest rankings [6]. To address the first step, a number of performance criteria have been proposed for filter-based feature selection. Correlation based Feature Selection (CFS) is a simple filter algorithm that ranks features according to a feature-class correlation [10]. The fast correlated-based filter (FCBF) method [36] is based on symmetrical uncertainty, which is defined as the ratio between the information gain and the entropy of two features. The INTERACT algorithm [38] uses the same goodness measure as FCBF filter, and it also includes the consistency contribution as an indicator about how significantly the elimination of particular feature will affect accuracy. The original RELIEF [12] and its variants [25, 31] algorithms estimate the quality of attributes according to how well their values distinguish between instances that are near to each other but belong to different classes.

Recently, several approaches proposed feature clustering in order to avoid selection of redundant features [3, 14, 33]. The authors in [32] proposed Fast clustering-bAsed feature Selection algoriThm (FAST). Here, the features are divided into clusters by using graph-theoretic clustering methods and the final subset of features is selected by choosing the most representative feature that is strongly related to target classes from each cluster. Similarly, the approach in [37] proposed hyper-graph clustering to extract maximally coherent feature groups from a set of objects. Furthermore, this approach neglects the assumption that the optimal feature subset is formed by features that only exhibit pairwise interactions. Instead

of that, they use multidimensional interaction information which includes third or higher order dependencies feature combinations in final selection.

Compared to existing graph-based and clustering-based feature selection methods, our approach leans on community detection techniques to cluster graphs that describe the strongest correlations among features. Additionally, the approach takes into account the size of identified communities. In contrast to traditional graph partitioning and data clustering techniques, a majority of community detection techniques are not computationally demanding and they do not require to specify the number of clusters in advance [7].

## 3 FSFCN: Feature Selection based on Feature Correlation Networks

The feature selection method proposed in this paper, denoted by FSFCN, is based on the notion of feature correlation networks. A feature correlation network describes correlations between features in a dataset that are equal to or higher than a specified threshold. To formally define feature correlation networks, we will assume that a dataset is composed of data instances having numeric features and a categorical class variable. However, the below given definition of feature correlation networks can be adapted in a straightforward manner for other types of datasets (categorical features, a mix of categorical and numeric features, continuous class variable) by taking appropriate correlation measures.

**Definition 1 (Feature Correlation Network)** Let $D$ be a dataset composed of data instances described by $k$ real-valued features $f_1, f_2, \ldots, f_k \in \mathbb{R}$ and a categorical class variable $c$. Let $C_f : \mathbb{R} \times \mathbb{R} \rightarrow [-1, 1]$ denote a correlation measure applicable to features (e.g, the Pearson or Spearman correlation coefficient) and let $C_c$ be a correlation measure applicable to a feature and the class variable (e.g., the mutual information, the Goodman-Kruskal index, and so on). The feature correlation network corresponding to $D$ is an undirected, weighted, attributed graph $G = (V, E)$ with the following properties:

- The set of **nodes** $V$ corresponds to the set of features ($f_i \in V$ for each $i$ in $[1 \,..\, k]$).
- Two features $f_i$ and $f_j$, $i \neq j$, are connected by an **edge** $e_{i,j}$ in $G$, $e_{i,j} \in E$, if $|C_f(f_i, f_j)| \geq T$, where $T$ is a previously specified threshold indicating a significant correlation between two features. The weight of $e_{i,j}$ is equal to $|C_f(f_i, f_j)|$.
- Each node in the network has a real-valued **attribute** reflecting its association with the class variable measured by $C_c$.

The features in $D$ can be ranked according to the measure $C_c$ and the top ranked features can be considered as the most relevant for training a classifier.

**Definition 2 (Subset of Relevant Features)** A subset $F_r$ of the set of features $F$ is called *relevant* if $(\forall f \in F_r) C_c(f) \geq R$, where $R$ is a feature relevance threshold indicating a significant association between a feature and the class variable.

**Definition 3 (Pruned Feature Correlation Network)** A pruned feature correlation network is a feature correlation network constructed from a subset of relevant features.

Our implementation of the FSFCN method[1] for datasets with real-valued features and categorical class variables uses pruned feature correlation networks that are constructed without explicitly stating the threshold indicating a significant correlation between two features ($T$). This means that the FSFCN algorithm has only one parameter – the feature relevance threshold $R$ separating relevant from irrelevant features. Additionally, the FSFCN method instruments the Spearman correlation coefficient to determine correlations among relevant features (the $C_f$ measure), while correlations between relevant features and the class variable are quantified by their mutual information (the $C_c$ measure). The mutual information between a real-valued feature $f$ and the categorical class variable $c$, denoted by $I(f, c)$, can be approximated by

$$I(f,c) \approx \sum_{y \in c} \sum_{x \in f'} p(x,y) \log \left( \frac{p(x,y)}{p(x)p(y)} \right),$$

where $f'$ is the set of discrete values obtained by a discretization of $f$, $p(x, y)$ is the joint probability distribution function of $f'$ and $c$, and $p(x)$ and $p(y)$ are the marginal probability distribution functions of $f'$ and $c$, respectively. $I(f, c)$ equal to 0 means that $f$ and $c$ are totally unrelated. A higher value of $I(f, c)$ implies a stronger association between $f$ and $c$. The JavaMI library[2] is used in our implementation of the FSFCN method to discretize continuous features and compute the mutual information between features and the class variable. For the default value of $R$ we use 0.05 which means that features having $I(f, c)$ lower than 0.05 are by default treated as irrelevant.

The algorithm for constructing pruned feature correlation networks consists of the following steps (see Algorithm 1):

1. The subset of relevant features $F_r$ is determined using the mutual information measure. Then, the nodes of the network are created such that each node corresponds to one feature from $F_r$.
2. In the second step, the algorithm forms a list $L$ containing tuples in the form $(f_i, f_j, S_{ij})$ for each pair of relevant features $f_i$ and $f_j$, where $S_{ij}$ denotes the value of the Spearman correlation coefficient between features $f_i$ and $f_j$.
3. $L$ is sorted by the third component of its elements ($S_{ij}$) in the decreasing order, i.e. the first element of the sorted list is the pair of features exhibiting the highest correlation, while the last element is the pair of features with the lowest correlation.
4. In the last step, the algorithm forms the links of the network by iterating through the sorted list $L$ beginning from the first element. In each iteration, for the currently processed element $e_k = (f_i, f_j, S_{ij})$ the algorithm creates a link $l_{ij}$ between $f_i$ and $f_j$ with weight $S_{ij}$. If the addition of $l_{ij}$ into the network results in a connected graph (i.e., a graph that has exactly one connected component or, equivalently, a graph in which there is a path between each pair of nodes) then the algorithm stops, otherwise it goes to the next element in the sorted list and repeats the previous link creation step. In other words, the algorithm iteratively builds the network by connecting features having the highest correlation until the network becomes a connected graph. Consequently, the weight of the last added link determines the value of the threshold $T$.

[1] The source code of FSFCN can be downloaded from https://github.com/milsav/FSFCN

[2] https://github.com/Craigacp/JavaMI

**Algorithm 1: Construction of pruned feature correlation networks**

**input :** $D$, $R$
$D$ – a dataset with real-valued features $F = \{f_1, f_2, \ldots, f_k\}$ and a categorical class variable $c$
$R$ – the feature relevance threshold separating relevant from irrelevant features

**output:** $G = (V, E)$, $T$
$G$ – the pruned feature correlation network of $D$
$T$ – the threshold indicating a significant correlation between features

// determine relevant features and form nodes in $G$
$F_r :=$ an empty set of relevant features
**foreach** $f \in F$ **do**
  $m :=$ the value of the mutual information of $f$ and $c$
  **if** $m \geq R$ **then**
    $F_r := F_r \cup \{f\}$
  **end**
**end**
$V := F_r$

// compute the Spearman correlation for each pair of relevant features
$L :=$ an empty list of tuples $(f_i, f_j, s)$
**foreach** $(f_i, f_j) \in F_r \times F_r, i \neq j$ **do**
  $s :=$ the value of the Spearman correlation for $f_i$ and $f_j$
  add $(f_i, f_j, s)$ to $L$
**end**

sort $L$ in the decreasing order of the Spearman correlation between paired features

// form links in the pruned feature correlation network
$i := 1$
*cont* := true
**while** *cont* **do**
  $s :=$ the first component of $L[i]$
  $d :=$ the second component of $L[i]$
  $w :=$ the third component of $L[i]$
  $l :=$ create a link between $s$ and $d$ with weight $w$
  $E := E \cup \{l\}$
  $i := i + 1$
  $T := w$
  *cont* := $G$ is not a connected graph
**end**

**return** $G$, $T$

The basic idea of the FSFCN method is to cluster a pruned feature correlation network in order to obtain cohesive groups of relevant features such that correlations among features within a group are stronger than correlations among features belonging to different groups. The FSFCN method leans on community detection techniques to identify clusters in feature correlation networks. The development of community detection techniques started with Newman and Girvan [21] who introduced a measure called modularity to estimate the quality of a partition of a network into cohesive node groups. The main idea behind the modularity measure is that a subgraph can be considered as a community if the actual number of links connecting nodes within the subgraph is significantly higher than the expected number of links in the same subgraph according to some null random graph model. In the case of weighted networks, the modularity measure accumulates differences

between the total weight of links within a community and the mathematical expectation of the previous quantity with respect to a random network having the same distributions of node degrees and link weights [20].

**Definition 4 (Modularity)** The modularity of a partitioned weighted network, denoted by $Q$, is defined as

$$Q = \sum_{c=1}^{n_c} \left[ \frac{W_c}{W} - \left( \frac{S_c}{2W} \right)^2 \right],$$

where $n_c$ is the number of communities in the network, $W_c$ is the sum of weights of intra-community links in $c$, $S_c$ is the total weight of links incident with nodes in $c$, and $W$ is the total weight of links in the network.

Four different community detection algorithms provided by the iGraph library [5] are used in our implementation of the FSFCN method to detect non-overlapping communities in feature correlation networks:

1. The Greedy Modularity Optimization (GMO) algorithm [4]. This algorithm uses a greedy hierarchical agglomeration strategy to maximize modularity. The algorithm starts with the partition in which each node is assigned to a singleton cluster. In each iteration of the algorithm, the variation in modularity obtained by merging any two communities is computed. The merge operation that maximally increases (or minimally decreases) modularity is chosen and the merge of corresponding clusters is performed.
2. The Louvain algorithm [1]. This method is an improvement of GMO. The Louvain algorithm is based on a greedy multi-resolution strategy to maximize modularity starting from the partition in which all nodes are put in different communities. When modularity is optimized locally by moving nodes to neighboring clusters, the algorithm creates a network of communities and then repeats the same procedure on that network until a maximum of modularity is obtained.
3. The Walktrap algorithm [22]. This algorithm relies on a node distance measure reflecting probability that a random walker moves from one node to another node in exactly $k$ steps ($k$ is the only parameter of the algorithm with the default value $k = 4$). The clustering dendrogram is constructed by the Ward's agglomerative clustering technique and the partition which maximizes modularity is taken as the output of the algorithm.
4. The Infomap algorithm [26]. This method reveals communities by optimally compressing descriptions of information flows on the network. The algorithm uses a greedy strategy to minimize the map equation which reflects the expected description length of a random walk on a partitioned network.

Each of the previously mentioned community detection algorithms defines one concrete implementation instance (i.e. one variant) of the FSFCN method.

The final step in the FSFCN method is the selection of features according to the obtained community partition of the pruned feature correlation network. The main idea is to select one or more features within each community such that:

1. selected features have a strong association with the class variable, and
2. any two selected features belonging to the same community are not directly connected in the pruned feature correlation network.

**Algorithm 2: The FSFCN algorithm**

**input :** $D$, $R$, CDA
$D$ – a dataset with real-valued features $F = \{f_1, f_2, \ldots, f_k\}$ and a categorical class variable $c$
$R$ – the feature relevance threshold separating relevant from irrelevant features
CDA – a community detection algorithm

**output:** $S$ – the set of selected features

// form the pruned feature correlation network corresponding to $D$
$G, T :=$ Algorithm1$(D, R)$

$C :=$ set of clusters in $G$ obtained by CDA

$S :=$ an empty set
**foreach** $c \in C$ **do**
  $(V_q, E_q) :=$ the subgraph of $G$ induced by nodes in $c$
  **while** $V_q \neq$ is not empty **do**
    // determine feature having the highest mutual information
    // with the class variable
    $f := \text{argmax}_{x \in V_q}\ C_c(x)$

    // remove $f$ and its neighbors from $(V_q, E_q)$
    $V_r := \{a \in V_q\ :\ \{f, a\} \in E_q\} \cup \{f\}$
    $E_r := \{\{a, b\} \in E_q\ : a \in V_r \vee b \in V_r\}$
    $V_q := V_q \setminus V_r$
    $E_q := E_q \setminus E_r$

    // add $f$ to the set of selected features
    $S := S \cup \{f\}$
  **end**
**end**

The pseudo-code describing the FSFCN feature selection is shown in Algorithm 2. After the pruned correlation network is constructed and clustered, the FSFCN method forms subgraphs of the network corresponding to identified communities where one subgraph is induced by nodes belonging to one community. The following operations are performed for each community subgraph:

1. A feature having the highest association with the class variable is identified and put in the set of selected features. Then, it is removed from the community subgraph together with its neighbors.
2. The previous step is repeated until the community subgraph becomes empty.

In other words, for each feature community the FSFCN method selects one or more features that represent the whole community. The method also takes into account the size of communities – a higher number of features is selected for larger feature communities. When a feature is added to the set of selected features its neighbors are removed from the community subgraph which means that the set of selected features will not contain features having a high mutual correlation (such two features are directly connected in the community subgraph).

## 4 Experiments and Results

The experimental evaluation of the FSFCN feature selection method was performed on a dataset with 120 plasma signaling protein features related to the diagnosis of the Alzheimer's disease [24]. The class variable indicates whether a patient was diagnosed with Alzheimer's or not. The total number of instances in the dataset is equal to 176 where 64 data instances correspond to patients diagnosed with Alzheimer's.

We performed feature selection using 4 variants of the FSFCN method. Each of those variants relies on a different community detection technique to cluster feature correlation networks. The variants of the FSFCN method are denoted by:

1. FG – the FSFCN method with the Fast Greedy Modularity Optimization (GMO) community detection algorithm,
2. LV – the FSFCN method with the Louvain algorithm,
3. WT – the FSFCN method with the Walktrap algorithm, and
4. IM – the FSFCN method with the Infomap algorithm.

The effectiveness of the FSFCN method is investigated by analyzing performance of seven classifiers trained without feature selection and after feature selection by different FSFCN variants and four other methods implemented in the WEKA machine learning library [35,8]. More specifically, the FSFCN method is compared with one feature subset selection and three feature ranking methods:

1. CFS – the correlation-based feature subset selection method proposed by Hall et al. [10],
2. GAINR – the feature ranking method based on the gain ratio measure,
3. INFOG – the feature ranking method based on the information gain measure, and
4. RFF – the ReliefF feature ranking method [12,13].

The WEKA machine learning library is also exploited to train and evaluate classifiers. The classification models used in the experimental evaluation are denoted by:

1. RF – the random forest classifier,
2. J48 – the C4.5 decision tree classifier,
3. LMT – the logistic model tree classifier,
4. JRIP – the RIPPER rule induction classifier,
5. LOGR – the logistic regression classifier,
6. SMO – the Support Vector Machine classifier, and
7. NB – the Naive Bayes classifier.

The default WEKA values are used for parameters of previously mentioned classification learning and feature selection methods. The performance of classifiers is compared using the classification accuracy measure (the fraction of correctly classified data instances).

### 4.1 Community structure in feature correlation networks

We firstly examined whether pruned feature correlation networks of our experimental dataset obtained at different values of the feature relevance threshold

parameter (denoted by $R$) exhibit a significant community structure since this the main assumption of the FSFCN feature selection method. We use $M$ to denote the maximum value of $R$ for which the corresponding pruned feature correlation network contains at least one link. For our experimental dataset we have obtained $M = 0.14$.

We applied the community detection algorithms used in different variants of the FSFCN method (Walktrap, GMO, Louvain and Infomap) to a sequence of pruned feature correlation networks formed by varying parameter $R$ from 0 (no feature pruning) to $M$ with a step size of 0.01. The basic characteristics of the pruned feature correlation networks (the number of nodes and links) and identified community partitions (the number of communities) are summarized in Table 1. It can be observed that Walktrap, GMO and Louvain identified community partitions containing more than one community for $R < 0.13$. The number of identified communities varies from 2 to 10 in the case of Walktrap and from 2 to 5 in the case of GMO and Louvain. On the other hand, the Infomap algorithm for a majority of $R$ values ($R \in [0.03, 0.08]$ and $R > 0.11$) identified exactly one community containing all the features present in the corresponding pruned feature correlation network.

**Table 1** Characteristics of community structure in the feature correlation networks identified by four community detection algorithms. $R$ – the feature relevance threshold, $N$ and $L$ – the number of nodes and links in the pruned feature correlation network, $C$ – the number of identified communities, $p$ – the probability that a randomly selected intra-community link has a higher weight than a randomly selected inter-community link. The bullet mark (•) indicates that intra-community links tend to have significantly higher weights according to the Mann-Whitney $U$ test.

| | | | Walktrap | | GMO | | Louvain | | Infomap | |
|---|---|---|---|---|---|---|---|---|---|---|
| $R$ | $N$ | $L$ | $C$ | $p$ | $C$ | $p$ | $C$ | $p$ | $C$ | $p$ |
| 0 | 120 | 1290 | 9 | 0.61• | 5 | 0.59• | 5 | 0.61• | 5 | 0.65• |
| 0.01 | 112 | 1109 | 9 | 0.61• | 4 | 0.59• | 4 | 0.63• | 6 | 0.63• |
| 0.02 | 90 | 786 | 10 | 0.61• | 5 | 0.60• | 4 | 0.65• | 5 | 0.68• |
| 0.03 | 56 | 535 | 6 | 0.65• | 4 | 0.65• | 3 | 0.64• | 1 | |
| 0.04 | 43 | 245 | 2 | 0.62• | 3 | 0.65• | 4 | 0.65• | 1 | |
| 0.05 | 35 | 161 | 7 | 0.65• | 4 | 0.64• | 4 | 0.64• | 1 | |
| 0.06 | 27 | 106 | 6 | 0.66• | 4 | 0.65• | 5 | 0.63• | 1 | |
| 0.07 | 21 | 102 | 3 | 0.58 | 2 | 0.66• | 2 | 0.66• | 1 | |
| 0.08 | 15 | 48 | 3 | 0.83• | 2 | 0.79• | 2 | 0.79• | 1 | |
| 0.09 | 10 | 15 | 3 | 0.40 | 3 | 0.40 | 3 | 0.40 | 2 | 1 |
| 0.10 | 9 | 13 | 2 | 1.00 | 3 | 0.40 | 3 | 0.40 | 2 | 1 |
| 0.11 | 8 | 10 | 2 | 1.00 | 3 | 0.25 | 3 | 0.25 | 2 | 1 |
| 0.12 | 7 | 7 | 2 | 1.00 | 2 | 1.00 | 2 | 1.00 | 1 | |
| 0.13 | 5 | 5 | 1 | | 2 | 0.00 | 2 | 0.00 | 1 | |
| 0.14 | 3 | 2 | 1 | | 1 | | 1 | | 1 | |

After performing community detection, the links in a pruned feature correlation network can be divided into two groups:

1. intra-community links – links connecting features belonging to the same community, and
2. inter-community links – links connecting features that are in different communities.

A good partition of a weighted network into communities should exhibit a significant value of the weighted modularity measure ($Q \gg 0$). Secondly, the weights of the intra-community links should be significantly higher than the weights of the inter-community links. Figure 1 shows the value of weighted modularity for identified community partitions of pruned feature correlation networks constructed at different feature relevance thresholds. It can be seen that Walktrap, GMO and Louvain detected community partitions with a significant weighted modularity ($Q > 0.2$) in pruned feature correlation networks obtained at $R \leq 0.6$.

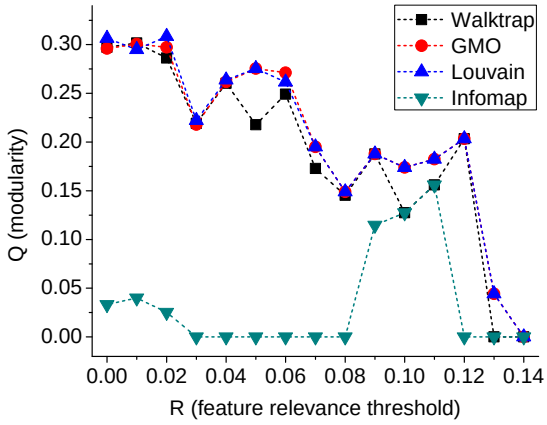



**Fig. 1** The value of the weighted modularity measure for community partitions identified by four community detection algorithms (Walktrap, GMO, Louvain and Infomap) in pruned feature correlation networks formed at different feature relevance thresholds.

We used the Mann-Whitney $U$ (MWU) test [18] to examine whether intra-community links tend to have higher weights than inter-community links. Let $W^{\text{intra}}$ and $W^{\text{inter}}$ denote the weight of a randomly selected intra-community and inter-community link, respectively. The MWU test can be instrumented to test the null hypothesis that

$$P(W^{\text{intra}} > W^{\text{inter}}) = P(W^{\text{inter}} > W^{\text{intra}})$$

against the alternative hypothesis that

$$P(W^{\text{intra}} > W^{\text{inter}}) > P(W^{\text{inter}} > W^{\text{intra}}),$$

where

- $P(W^{\text{intra}} > W^{\text{inter}})$ is the probability of superiority of intra-community links over inter-community links with respect to link weights, i.e. the probability that a randomly selected intra-community link has a higher weight than a randomly selected inter-community links, and
- $P(W^{\text{inter}} > W^{\text{intra}})$ is the opposite probability of superiority.

The obtained values of $P(W^{\text{intra}} > W^{\text{inter}})$ are also shown in Table 1. The probability of superiority of intra-community links is higher than 0.5 for all non-trivial community partitions (partitions containing more than one community) identified in pruned feature correlation networks corresponding to $R \leq 0.08$ which implies that

$$P(W^{\text{intra}} > W^{\text{inter}}) > P(W^{\text{inter}} > W^{\text{intra}}).$$

The application of the MWU test revealed that the null hypothesis of equal probabilities of superiority can be rejected in favor of the alternative hypothesis for all non-trivial community partitions in pruned feature correlation networks obtained at $R \leq 0.08$, except for the community partition identified by the Walktrap algorithm in the pruned feature correlation network obtained at $R = 0.07$. The null hypothesis of the equal probabilities of superiority is also rejected for community partitions identified by Infomap in pruned feature correlation networks obtained at $R < 0.03$. However, those community partitions exhibit an extremely low value of the weighted modularity measure ($Q < 0.05$) suggesting that they contain one giant and several small-size communities. Indeed, the fraction of nodes in the largest community identified by Infomap in the pruned feature correlation networks obtained at $R < 0.03$ varies from 0.85 to 0.89 implying the presence of a giant community encompassing the vast majority of relevant features. On the other hand, the fraction of nodes in the largest community identified by Walktrap, GMO and Louvain in pruned feature correlation network obtained at $R \leq 0.06$ varies from 0.25 to 0.56 implying that those three community detection algorithms identified relatively balanced community partitions. Summarizing all findings, it can be concluded that pruned feature correlation networks of our experimental dataset obtained at $R \leq 0.6$ exhibit a moderately strong community structure characterized by (1) a significant modularity, (2) the tendency of intra-community links to be more stronger than inter-community links, and (3) relatively balanced communities in terms of their size.

The pruned feature correlation network of our experimental dataset obtained at the default value of the feature relevance threshold ($R = 0.05$) contains 35 nodes which means that 35 out of 120 features exhibit a significant association with the class variable in terms of mutual information. Those 35 nodes representing relevant features are connected by 161 links which implies that a randomly selected relevant feature has a significant correlation with 9.2 other relevant features on average. The maximal and minimal absolute value of link weights are 0.72 and 0.32, respectively, which means that there are moderate to strong Spearman correlations among relevant features. The results of community detection on the pruned feature correlation network obtained at the default value of the $R$ parameter are summarized in Table 2. To compare obtained community partitions we computed the Rand index [23] for each pair of them. It can be noticed that GMO and Louvain identified the same number of communities with the same value of the weighted modularity measure and the same distribution of community sizes. Actually, those two methods identified exactly the same communities – the Rand index for the community partitions obtained by GMO and Louvain is equal to 1. The Walktrap method identified a partition with a higher number of communities and a lower value of weighted modularity compared to GMO/Louvain. The Rand index between partitions obtained by Walktrap and GMO/Louvain is equal to 0.79 indicating that those two community partitions are highly similar. Finally, it

can be seen that Infomap failed to identify communities in the network, i.e. this method identified one community encompassing all nodes in the network.

**Table 2** Results of the community detection for the pruned feature correlation network obtained at the default value of the feature relevance threshold ($R = 0.05$). $C$ – the number of identified communities, $Q$ – the value of the weighted modularity measure, $S$ – the vector giving the size of identified communities.

| Algorithm | $C$ | $Q$ | $S$ |
|---|---|---|---|
| GMO | 4 | 0.275 | (14, 8, 7, 6) |
| Louvain | 4 | 0.275 | (14, 8, 7, 6) |
| Walktrap | 7 | 0.218 | (13, 8, 5, 5, 2, 1, 1) |
| Infomap | 1 | 0 | (35) |

The features selected by different variants of the FSFCN method from the full experimental dataset at the default value of the feature relevance threshold parameter are shown in Table 3. FG and LV selected the same features since community partitions obtained by the corresponding community detection algorithms are identical. It can be observed that each FSFCN variant greatly reduced the dimensionality of the dataset – the number of selected features varies from 7 to 12. On the other hand, the CFS method implemented in WEKA applied to the full experimental dataset selected a higher number of features (25 features).

**Table 3** The features selected by four different variants of the FSFCN method from the full dataset at the default value of the feature relevance threshold ($R = 0.05$). Feature ranks are determined according to the mutual information with the class variable.

| FG/LV | | WT | | IM | |
|---|---|---|---|---|---|
| | Rank | | Rank | | Rank |
| IL-1a | 1 | IL-1a | 1 | IL-1a | 1 |
| IL-8 | 2 | TNF-a | 3 | PDGF-BB | 7 |
| TNF-a | 3 | GCSF | 6 | sTNF RI | 12 |
| PDGF-BB | 7 | PDGF-BB | 7 | Eotaxin | 15 |
| sTNF RI | 12 | sTNF RI | 12 | MCP-2 | 17 |
| VEGF-B | 14 | Eotaxin | 15 | IGFBP-2 | 23 |
| Eotaxin | 15 | SCF | 16 | TPO | 31 |
| MIP-1d | 19 | MIP-1d | 19 | | |
| IGFBP-2 | 23 | CTACK | 22 | | |
| | | IGFBP-2 | 23 | | |
| | | BTC | 30 | | |
| | | TPO | 31 | | |

We formed the reduced datasets containing those features selected by different variants of the FSFCN and WEKA CFS feature selection algorithms applied to the full experimental dataset. Then, we trained and evaluated the considered classifiers on the full and reduced datasets using the stratified 10-fold cross-validation procedure implemented in the WEKA library. The obtained classification accuracies are shown in Table 4. It can be observed that the classifiers trained on

the full dataset tend to exhibit the lowest classification accuracy. The classifiers trained on the dataset containing features selected by the WEKA CFS method constantly performed better than the classifiers trained on the full dataset. On the other hand, the classifiers trained on the datasets containing features selected by FG/LV and WT exhibited a better classification performance compared to the classifiers trained on the full dataset except in one case. Namely, the accuracy of LMT without feature selection is equal to the accuracy of the same classifier trained on the reduced datasets containing features selected by FG/LV and WT. Consequently, we can say that the feature selection based on properly clustered feature correlation networks does not decrease the performance of all considered classifiers while notably reducing the dimensionality of the dataset.

**Table 4** The accuracy of classifiers trained on the reduced datasets containing features selected by different FSFCN variants applied to the full dataset. The column FULL corresponds to classifiers trained on the full dataset, while the column WEKA-CFS corresponds to classifiers trained on the dataset containing features selected by the CFS feature selection method from WEKA (also applied to the full dataset). One star indicates the lowest performance, while two stars indicate the highest performance.

| | FULL | WEKA-CFS | FG/LV | WT | IM |
|---|---|---|---|---|---|
| RF | 0.82 | 0.85** | 0.82 | 0.85** | 0.79* |
| J48 | 0.74* | 0.77 | 0.77 | 0.81** | 0.74* |
| LMT | 0.84 | 0.85** | 0.84 | 0.84 | 0.83* |
| JRIP | 0.72* | 0.81** | 0.79 | 0.78 | 0.75 |
| LOGR | 0.73* | 0.81 | 0.85** | 0.85** | 0.84 |
| SMO | 0.82* | 0.83 | 0.84 | 0.86** | 0.85 |
| NB | 0.78* | 0.84 | 0.88** | 0.88** | 0.84 |

The next important result that can be observed in Table 4 is that the IM variant of the FSFCN method exhibits the worst classification performance compared to other three FSFCN variants. The IM variant in this case is actually equivalent to the FSFCN method without the clustering step since IM identified exactly one community encompassing all features in the pruned feature correlation network corresponding to $R = 0.05$. Consequently, it can be concluded that clustering of pruned feature correlation networks enables a better selection of relevant features.

The best performing classifier trained without feature selection is LMT, achieving accuracy of 0.84. The best classifiers trained on the reduced dataset containing features selected by WEKA CFS are RF and LMT, achieving accuracy of 0.85. On the other hand, the classifier with the highest accuracy is NB trained on the reduced dataset containing features selected by three different variants of the FSFCN method. Finally, the classifiers trained on the reduced dataset containing features selected by the WT variant of the FSFCN method tend to exhibit the best overall performance.

### 4.2 Effectiveness of the FSFCN method

To evaluate the effectiveness of the FSFCN feature selection method, we determined the average number of selected features and the accuracy of classifiers

trained after FSFCN feature selection variants for different values of the feature relevance threshold parameter. In contrast to the experiments described in Section 4.1, the FSFCN feature selection was not conducted once on the full experimental dataset, but several times on parts of the dataset. More specifically, for each value of the feature relevance threshold in the range $[0, 0.14]$ with a step size of 0.01, a stratified 10-fold cross-validation procedure was used to determine the accuracy of classifiers after FSFCN feature selection performed over 9 folds that are used to train classifiers. The experimental dataset was divided into stratified folds using the WEKA library. Then, the accuracy of classifiers trained after FSFCN feature selection was compared to the accuracy of baseline classifiers trained without feature selection.

Figure 2 shows the average number of selected features by different FSFCN variants at different feature relevance thresholds. As expected, the number of selected features decreases with $R$ due to smaller pruned feature correlation networks. We can see that the average number of selected features decreases from maximally 27.3 for $R = 0$ (no feature pruning) to maximally 2.3 for $R = 0.14$, which means that all four variants of the FSFCN feature selection method greatly reduce the dimensionality of the experimental dataset (the number of features in the dataset is equal to 120).

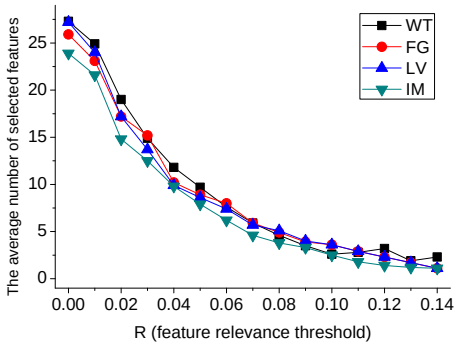
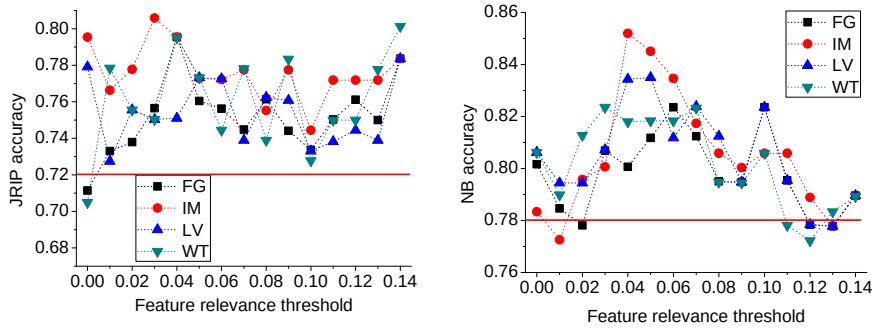



**Fig. 2** The average number of selected features by different FSFCN variants at different feature relevance thresholds.

The classification accuracy of JRIP and NB after the FSFCN feature selection variants is shown in Figure 3. It can be observed that all FSFCN variants for $R > 0$ improve the accuracy of JRIP compared to the baseline JRIP classifier trained without feature selection. For $R = 0$, JRIP classifiers trained after FG and WT exhibit a lower accuracy than the baseline JRIP classifier. On the other hand, JRIP classifiers trained after LW and IM at $R = 0$ have a better accuracy compared to the baseline JRIP classifier. The maximum accuracy of JRIP is achieved after feature selection by the IM FSFCN variant at $R = 0.03$. It is equal to 0.81 which is 9 percentage points higher than the accuracy of the JRIP baseline classifier (0.72). Similarly as for JRIP, all FSFCN feature selection variants improve the accuracy of NB, except in a small number of cases corresponding to extremely small and extremely high values of the feature relevance threshold. The highest accuracy of NB is achieved after feature selection by IM at $R = 0.04$ and it is equal to

0.85, which is an increase of 13 percents compared to the NB baseline classifier trained without feature selection. It is also interesting to observe that JRIP and NB after the IM FSFCN variant tend to have a better classification accuracy compared to the same classifiers trained after feature selection by other three FSFCN variants. This suggests that the Infomap community detection algorithm performs drastically better on reduced datasets (9 folds) than on the full dataset where it failed to identify feature communities.




(a) (b)

**Fig. 3** The accuracy of JRIP (a) and NB (b) trained after feature selection by 4 variants of the FSFCN method. The solid horizontal line denotes the accuracy of the classifiers trained without feature selection.

Figure 4 shows the accuracy of J48, RF and SMO classification models after FSFCN feature selection. It can be seen that all FSFCN variants for $R$ in the interval $[0.04, 0.07]$ improve the accuracy of J48 compared to the baseline J48 classifier. However, J48 classifiers trained after all FSFCN variants for large values of the feature relevance threshold ($R > 0.08$) exhibit a lower accuracy than the baseline J48 classifier. For small feature relevance thresholds ($R < 0.04$) the situation is mixed:

1. IM improves the accuracy of J48,
2. J48 classifiers trained after LV exhibit either a higher or equal accuracy compared to the J48 baseline classifier, and
3. the J48 baseline classifier has a higher accuracy than J48 classifiers trained after FG and WT.

A mixed situation can be also observed regarding the accuracy of SMO classifiers:

1. LV improves the accuracy of SMO for $R$ in the range $[0.04, 0.08]$ and for $R$ equal to 0 and 0.02,
2. IM improves the accuracy of SMO for $R$ in the range $[0.04, 0.06]$ and for $R$ equal to 0 and 0.08,
3. FG and WT improve the accuracy of SMO in only 4 (out of 15) cases of different $R$ values, and
4. the SMO classifiers trained after FSFCN variants at $R \geq 0.09$ exhibit a lower accuracy than the baseline SMO classifier trained without feature selection.

The FSFCN feature selection method is not very successful in combination with the RF classification model on our experimental dataset. The baseline RF classifier has a higher accuracy than RF classifiers trained after all FSFCN feature selection variants for $R > 0.07$. Feature selection by FG, LV and LT in a very small number of cases (less than 3) slightly increase the accuracy of RF. On the other hand, RF classifiers trained after IM for $R \leq 0.07$ exhibit a better or equal accuracy compared to the baseline RF classifier.

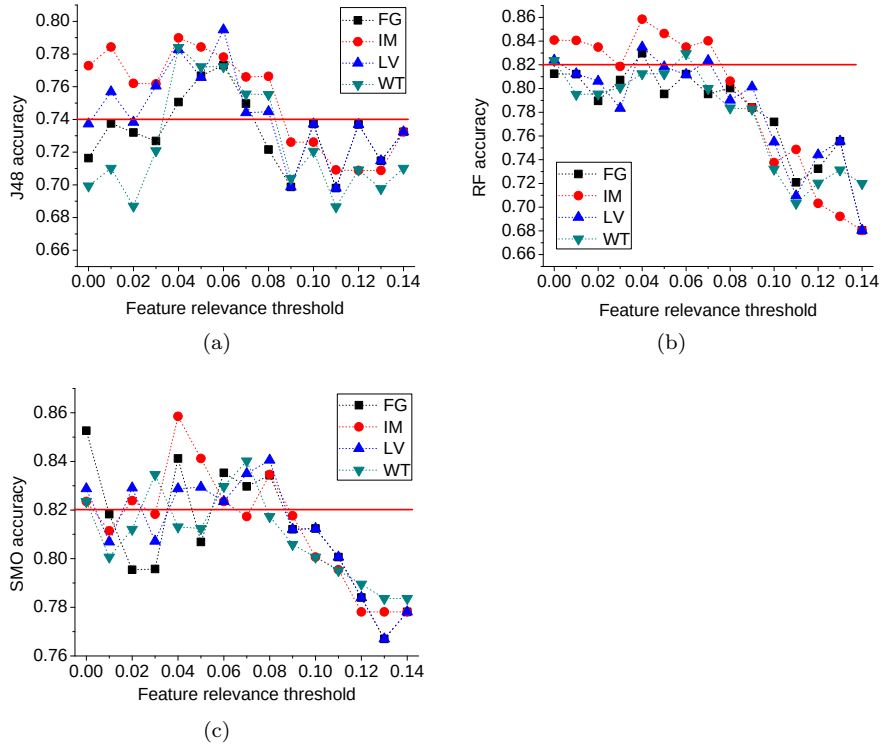
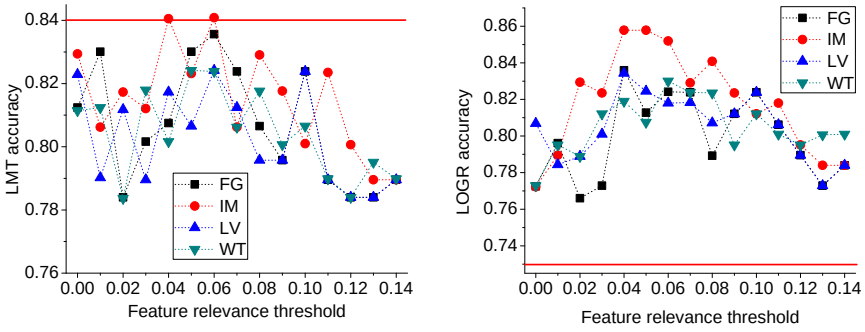



**Fig. 4** The accuracy of J48 (a), RF (b) and SMO (c) trained after feature selection by 4 variants of the FSFCN method. The solid horizontal line denotes the accuracy of classifiers trained without feature selection.

The impact of the FSFCN feature selection on the accuracy of LMT and LOGR is shown in Figure 5. It can be observed that FSFCN feature selection in opposite ways affects the accuracy of LMT and LOGR:

– The accuracy of LOGR after feature selection by all FSCFN variants through the whole range of $R$ values is higher than the accuracy of the LOGR baseline classifier trained without feature selection. The highest accuracy of LOGR equals 0.84 and it is achieved after IM feature selection at $R = 0.04$. On the other hand, the accuracy of the LOGR baseline classifier is 0.73, which is 11 percentage points lower than the highest observed accuracy.

– The accuracy of LMT after feature selection by all FSFCN variants through the whole range of $R$ values is lower than the accuracy of the LMT baseline classifier, except in two cases (IM at $R = 0.04$ and $R = 0.06$) where the accuracy of LMT after FSFCN feature selection is equal to the accuracy of the LMT baseline classifier.




(a) (b)

**Fig. 5** The accuracy of LMT (a) and LOGR (b) trained after feature selection by 4 variants of the FSFCN method. The solid horizontal line denotes the accuracy of classifiers trained without feature selection.

Summarizing the findings presented in this section, it can be concluded that all variants of the FSFCN method determined by different community detection techniques greatly reduce the dimensionality of the dataset and at the same time notably improve the accuracy of LOGR, JRIP and NB classifiers. Secondly, the classification accuracy of all considered classification models, except LMT, can be increased after feature selection by one or more variants of the FSFCN method.

### 4.3 Comparative analysis of FSFCN with other feature selection methods

We also compared the variants of the FSFCN method to four other widely used feature selection methods (CFS, GAINR, INFOG and RFF to which we refer as the baseline feature selection methods) by analyzing the performance of seven classifiers trained after feature selection. The experimental procedure used to compare FSFCN with baseline feature selection methods is described in Algorithm 3.

The accuracy of each considered classifier in combination with each examined feature selection method was determined by 10 runs of a stratified 10-fold cross-validation process. This means that 100 instances of each classifier were trained for each feature selection method. As in the previous experiment, feature selection and classification training were performed over 9 folds in each cross-validation step, while the accuracy of classifiers was computed relying on the remaining fold. The WEKA library was used to form 10 different divisions of the experimental dataset into stratified folds, select features by baseline methods with the default WEKA values for their parameters, train classifiers and evaluate their performance. The Wilcoxon-signed rank test [34] was utilized to examine whether there

**Algorithm 3: Experimental procedure to compare FSFCN variants to baseline feature selection methods**

```
Classifiers := {J48, JRIP, LMT, LOGR, NB, RF, SMO}
FSFCNVariants := {FG, IM, LV, WT}
BaselineMethods := {FULL, CFS, GAINR, INFOG, RFF}
// FULL means that a classifier will be trained without feature selection

D := an empty array of dataset divisions into stratified folds
for i := 1 to 10 do
    D[i] := divide the experimental dataset into 10 stratified folds
end

foreach (F, B, C) ∈ FSFCNVariants × BaselineMethods × Classifiers do
    A_f, A_b := empty arrays of real numbers
    k := 1
    for i := 1 to 10 do
        for j := 1 to 10 do
            testSet := the j-th fold of D[i]
            trainingSet := all folds of D[i] excluding the j-th fold

            f := apply F to trainingSet
            n := the number of features in f
            C_f := train(C, f, trainingSet)
            A_f[k] := accuracy(C_f, f, testSet)

            b := an empty set of features
            if B = FULL then
                b := all features
            else if B is a feature ranking method then
                b := top n ranked features by B in trainingSet
            else
                b := apply B to trainingSet
            end
            C_b := train(C, b, trainingSet)
            A_b[k] := accuracy(C_b, b, testSet)

            k := k + 1
        end
    end
    F^acc := the average value of A_f
    B^acc := the average value of A_b
    p := apply the Wilcoxon-signed rank test to A_f and A_b
    if p < 0.05 then
        if F^acc > B^acc then
            C trained after FSFCN variant F has a significantly higher accuracy than
            C trained after B
        else
            C trained after FSCFN variant F has a significantly lower accuracy than
            C trained after B
        end
    end
end
```

is a statistically significant difference in the accuracy of a classifier trained after FSFCN variants and after baseline feature selection methods. The default value of the feature relevance threshold parameter ($R = 0.05$) was used in all FSFCN feature selection variants. Finally, classifiers trained after baseline feature ranking methods (GAINR, INFOG and REF) were trained considering the top $k$ ranked features, where $k$ is the number of features selected by FSFCN.

Table 5 shows the results of comparison between the FG variant of FSFCN and the baseline feature selection methods including also classifiers trained without the feature selection. It can be observed that FG significantly improves the accuracy of NB and LOGR classifiers trained without feature selection. Secondly, NB and LOGR trained after FG significantly outperform the same classifiers trained after feature selection by CFS and RFF. Additionally, NB after FG exhibits a significantly better accuracy than NB trained after all baseline feature selection methods. The only case in which a baseline feature selection method gives a significantly better selection of features for a particular classifier is CFS in combination with RF. Finally, it can be observed that FG outperforms all baseline feature selection methods in terms of the total number of classification models which perform better after FG (the FG win-loss scores are shown at the bottom of Table 5).

**Table 5** The comparison of the FG FSFCN variant with the baseline feature selection methods. The down arrow (↓) denotes that a classifiers trained without feature selection (the column FULL) or trained after a baseline feature selection method has a significantly worse accuracy than the same classifier trained after FG (according to the Wilcoxon-signed rank test), while the up arrow (↑) corresponds to the opposite case.

| | FG | FULL | CFS | GAINR | INFOG | RFF | Best |
|---|---|---|---|---|---|---|---|
| J48 | 0.756 | 0.736 | 0.749 | 0.739 | 0.739 | 0.750 | FG |
| JRIP | 0.766 | 0.759 | 0.769 | 0.766 | 0.771 | 0.752 | INFOG |
| LMT | 0.830 | 0.836 | 0.827 | 0.817 | 0.822 | 0.819 | FULL |
| LOGR | 0.836 | 0.698 ↓ | 0.808 ↓ | 0.832 | 0.830 | 0.810 ↓ | FG |
| NB | 0.826 | 0.777 ↓ | 0.799 ↓ | 0.801 ↓ | 0.790 ↓ | 0.792 ↓ | FG |
| RF | 0.819 | 0.823 | 0.833 ↑ | 0.823 | 0.815 | 0.817 | CFS |
| SMO | 0.826 | 0.832 | 0.819 | 0.828 | 0.830 | 0.810 | FULL |
| FG wins | | 4 | 5 | 4 | 5 | 7 | |
| FG losses | | 3 | 2 | 2 | 2 | 0 | |

The results of the comparison between the IM FSCFN variant and the baseline feature selection methods are summarized in Table 6. It can be seen that J48, LOGR and NB after IM have significantly higher accuracies than the corresponding baseline classifiers trained without feature selection. IM significantly outperforms CFS for NB and SMO, GAINR and INFOG in the case of J48 and NB, and RFF for all classifiers except J48. It can be also noticed that all considered classifiers trained after IM do not never exhibit a significantly lower accuracy compared to classifiers trained after baseline feature selection methods. To the contrary, classifiers trained after IM strongly tend to have a higher accuracy than classifiers obtained after the baseline feature selection methods.

The results of the comparative analysis between the LV FSFCN variant and the baseline feature selection methods are given in Table 7. The J48 classifier trained after LV shows a significantly higher accuracy compared to the J48 classifier trained on the full dataset and J48 classifiers trained after all baseline feature selection methods except RFF. Additionally, LV significantly improves the accuracy of NB compared to all baseline feature selection methods. LV also significantly outperforms CFS in the case of the LOGR classifier. The only case when a baseline method leads to a significantly better selection of features for a particular classifier is RF trained after CFS.

**Table 6** The comparison of the IM FSFCN variant with the baseline feature selection methods.

| | IM | FULL | CFS | GAINR | INFOG | RFF | Best |
|---|---|---|---|---|---|---|---|
| J48 | 0.773 | 0.736 ↓ | 0.749 | 0.746 ↓ | 0.737 ↓ | 0.757 | IM |
| JRIP | 0.775 | 0.759 | 0.769 | 0.774 | 0.770 | 0.750 ↓ | IM |
| LMT | 0.830 | 0.836 | 0.827 | 0.818 | 0.829 | 0.808 ↓ | FULL |
| LOGR | 0.829 | 0.698 ↓ | 0.808 | 0.829 | 0.832 | 0.808 ↓ | IM/GAINR |
| NB | 0.850 | 0.777 ↓ | 0.799 ↓ | 0.808 ↓ | 0.795 ↓ | 0.798 ↓ | IM |
| RF | 0.830 | 0.823 | 0.833 | 0.819 | 0.813 | 0.814 ↓ | CFS |
| SMO | 0.834 | 0.832 | 0.819 ↓ | 0.833 | 0.825 | 0.804 ↓ | IM |
| IM wins | | 6 | 6 | 6 | 6 | 7 | |
| IM losses | | 1 | 1 | 0 | 1 | 0 | |

**Table 7** The comparison of the LV variant of FSFCN with baseline feature selection methods.

| | LV | FULL | CFS | GAINR | INFOG | RFF | Best |
|---|---|---|---|---|---|---|---|
| J48 | 0.772 | 0.736 ↓ | 0.749 ↓ | 0.743 ↓ | 0.736 ↓ | 0.753 | LV |
| JRIP | 0.771 | 0.759 | 0.769 | 0.778 | 0.777 | 0.751 ↓ | GAINR |
| LMT | 0.834 | 0.836 | 0.827 | 0.822 | 0.823 | 0.820 | FULL |
| LOGR | 0.840 | 0.698 ↓ | 0.808 ↓ | 0.829 | 0.829 | 0.818 ↓ | LV |
| NB | 0.826 | 0.777 ↓ | 0.799 ↓ | 0.802 ↓ | 0.799 ↓ | 0.797 ↓ | LV |
| RF | 0.821 | 0.823 | 0.833 ↑ | 0.821 | 0.820 | 0.817 | CFS |
| SMO | 0.833 | 0.832 | 0.819 | 0.823 | 0.830 | 0.812 ↓ | LV |
| LV wins | | 5 | 6 | 5 | 6 | 7 | |
| LV looses | | 2 | 1 | 1 | 1 | 0 | |

The comparison of the WT FSFCN variant with the baseline feature selection methods is summarized in Table 8. As for all previous FSFCN variants, WT outperforms baseline feature selection methods in terms of the number of classifiers that have a higher accuracy after WT. It can be also noticed that WT, similarly to IM and LV variants of the FSFCN method, significantly improves the accuracy of J48, LOGR and NB compared to the baseline classifiers trained without feature selection. Statistically significant differences in accuracies of classifiers trained after WT and baseline feature selection methods are present in the following cases:

1. J48 and NB trained after WT possess a significantly higher accuracy than J48 and NB trained after all baseline feature selection methods.
2. WT significantly outperforms CFS in the case of three classifiers (J48, LOGR and NB). However, RF trained after CFS exhibits a significantly higher accuracy than RF trained after WT.
3. The accuracy of all classification models trained after WT is never significantly lower compared to the same models trained after the baseline feature ranking methods. To the contrary, WT significantly outperforms GAINR for three classification models, INFOG for two classification models and RFF for five classification models.

**Table 8** The comparison of the WT variant of FSFCN with baseline feature selection methods.

| | WT | FULL | CFS | GAINR | INFOG | RFF | Best |
|---|---|---|---|---|---|---|---|
| J48 | 0.774 | 0.736 ↓ | 0.749 ↓ | 0.740 ↓ | 0.741 ↓ | 0.749 ↓ | WT |
| JRIP | 0.775 | 0.759 | 0.769 | 0.769 | 0.772 | 0.754 ↓ | WT |
| LMT | 0.830 | 0.836 | 0.827 | 0.815 ↓ | 0.823 | 0.816 ↓ | FULL |
| LOGR | 0.831 | 0.698 ↓ | 0.808 ↓ | 0.831 | 0.832 | 0.813 | INFOG |
| NB | 0.827 | 0.777 ↓ | 0.799 ↓ | 0.796 ↓ | 0.799 ↓ | 0.794 ↓ | WT |
| RF | 0.820 | 0.823 | 0.833 ↑ | 0.819 | 0.818 | 0.816 | CFS |
| SMO | 0.832 | 0.832 | 0.819 | 0.824 | 0.833 | 0.809 ↓ | INFOG |
| WT wins | | 4 | 6 | 6 | 5 | 7 | |
| WT looses | | 2 | 1 | 0 | 2 | 0 | |

## 5 Conclusion and Future Work

In this paper we presented the FSFCN method for feature selection based on community detection techniques applied to feature correlation networks. Feature correlation networks are weighted graphs showing the strongest correlations among features present in a dataset. The first step of FSFCN is the construction of a pruned feature correlation network for a given value of the feature relevance threshold parameter. The feature relevance threshold separates relevant from irrelevant features, i.e. features having a strong and weak association with the class variable, respectively, in terms of the mutual information measure. The pruned feature correlation network is formed incrementally in a greedy manner: links between features are created in the decreasing order of feature correlations until the network becomes a connected graph. The crucial idea of FSFCN is to cluster the pruned feature correlation network using a community detection technique in order to identify groups of features such that correlations between features within a group tend to be stronger than correlations between features belonging to different groups. Then, one or more features representing each group of features are selected taking into account correlations of features with the class variable, the size of feature communities and connections within them.

The experimental evaluation of four variants of the FSFCN method, where each variant employs a different community detection technique, was conducted on a highly dimensional dataset (120 features) related to the diagnosis of Alzheimer's disease. We firstly demonstrated that pruned feature correlation networks of the dataset obtained at different feature relevance thresholds exhibit a significant community structure. Then, we performed feature selection by the FSFCN variants and the WEKA CFS feature selection method for default values on their parameters on the full dataset, created reduced datasets containing selected features and trained seven classifiers on the full dataset and reduced datasets. The evaluation of classification performance by the 10-fold cross-validation method revealed that the classifiers trained on the reduced dataset formed by the FSFCN variant employing the Walktrap community detection algorithm exhibit the best overall performance. Additionally, the obtained results indicate that clustering of feature correlation networks yields to a better selection of features for classification purposes.

In the next experiment, we investigated the effectiveness of the FSFCN feature selection variants at different values of the feature relevance threshold parameter.

In contrast to the previous experiment, FSFCN feature selection was performed on a part of the experimental dataset used to train classifiers that were evaluated using the 10-fold cross-validation method. The obtained results show that all FSFCN variants significantly reduce the dimensionality of the experimental dataset. Secondly, all FSFCN variants are able to improve the classification accuracy of all considered classification models excluding LMT, being very successful for three classification models (LOGR, JRIP and NB) for a wide range of feature relevance threshold values.

Finally, we compared the FSFCN variants to four widely used feature selection methods implemented in the WEKA library. Ten runs of the stratified 10-fold cross-validation procedure were performed in order to make statistically sound comparison of classifiers trained after FSFCN and classifiers trained after reference feature selection methods. In each cross-validation step, feature selection and classification training were conducted on 9 folds of the experimental dataset with default values for parameters of feature selection and classification algorithms, while the classification accuracy was computed relying on the remaining fold. The experimental results revealed that all FSFCN variants outperform reference feature selection methods in terms of the number of classifiers having a higher accuracy after FSFCN feature selection. Additionally, J48, LOGR and NB classifiers trained after FSFCN variants tend to have a significantly higher accuracy than classifiers trained after reference feature selection methods.

The main task in our future work will be to perform a more comprehensive evaluation of our approach considering high dimensional datasets from various domains. It is also possible to experiment with additional variants of the method taking into account other correlation measures and community detection algorithms (including also community detection techniques which identify overlapping communities). Finally, in this paper we have focused on feature selection in the context of data classification. In our future work we will also focus on adaptations of the FSFCN method for data clustering. Currently, the selection of features representing communities of features is guided by the mutual information between a feature and the class variable. We plan to examine different network centrality measures instead of the mutual information in order to be able to apply the method on datasets containing uncategorized data instances and investigate its performance in this setting.

**Acknowledgements** This work is supported by the bilateral project "Intelligent computer techniques for improving medical detection, analysis and explanation of human cognition and behavior disorders" between the Ministry of Education, Science and Technological Development of the Republic of Serbia and the Slovenian Research Agency. M. Savić, V. Kurbalija and M. Ivanović also thank the Ministry of Education, Science and Technological Development of the Republic of Serbia for additional support through project no. OI174023, "Intelligent techniques and their integration into wide-spectrum decision support.".

## References

1. Blondel, V.D., Guillaume, J.L., Lambiotte, R., Lefebvre, E.: Fast unfolding of communities in large networks. Journal of Statistical Mechanics: Theory and Experiment **2008**(10), P10008 (2008). DOI 10.1088/1742-5468/2008/10/P10008

2. Boccaletti, S., Latora, V., Moreno, Y., Chavez, M., Hwang, D.U.: Complex networks: Structure and dynamics. Physics Reports **424**(4-5), 175–308 (2006). DOI 10.1016/j.physrep.2005.10.009
3. Butterworth, R., Piatetsky-Shapiro, G., Simovici, D.A.: On feature selection through clustering. In: Proceedings of the Fifth IEEE International Conference on Data Mining, ICDM '05, pp. 581–584. IEEE Computer Society, Washington, DC, USA (2005). DOI 10.1109/ICDM.2005.106
4. Clauset, A., Newman, M.E.J., Moore, C.: Finding community structure in very large networks. Physical Review E **70**, 066111 (2004). DOI 10.1103/PhysRevE.70.066111
5. Csardi, G., Nepusz, T.: The igraph software package for complex network research. InterJournal Complex Systems p. 1695 (2006)
6. Duch, W.: Filter Methods, pp. 89–117. Springer Berlin Heidelberg, Berlin, Heidelberg (2006). DOI 10.1007/978-3-540-35488-8_4
7. Fortunato, S.: Community detection in graphs. Physics Reports **486**(35), 75 – 174 (2010). DOI 10.1016/j.physrep.2009.11.002
8. Frank, E., Hall, M., Holmes, G., Kirkby, R., Pfahringer, B., Witten, I.H., Trigg, L.: Weka - A Machine Learning Workbench for Data Mining, pp. 1269–1277. Springer US, Boston, MA (2010)
9. Guyon, I., Elisseeff, A.: An introduction to variable and feature selection. Journal of Machine Learning Research **3**, 1157–1182 (2003)
10. Hall, M.A.: Correlation-based feature subset selection for machine learning. Ph.D. thesis, University of Waikato, Hamilton, New Zealand (1998)
11. Horvath, S.: Correlation and Gene Co-Expression Networks, pp. 91–121. Springer New York, New York, NY (2011). DOI 10.1007/978-1-4419-8819-5_5
12. Kononenko, I.: Estimating attributes: Analysis and extensions of RELIEF, pp. 171–182. Springer Berlin Heidelberg, Berlin, Heidelberg (1994). DOI 10.1007/3-540-57868-4_57
13. Kononenko, I., Šimec, E., Robnik-Šikonja, M.: Overcoming the myopia of inductive learning algorithms with RELIEFF. Applied Intelligence **7**(1), 39–55 (1997). DOI 10.1023/A:1008280620621
14. Krier, C., Franois, D., Rossi, F., Verleysen, M.: Feature clustering and mutual information for the selection of variables in spectral data. In: Proc European Symposium on Artificial Neural Networks Advances in Computational Intelligence and Learning, pp. 157–162 (2007)
15. Lal, T.N., Chapelle, O., Weston, J., Elisseeff, A.: Embedded Methods, pp. 137–165. Springer Berlin Heidelberg, Berlin, Heidelberg (2006). DOI 10.1007/978-3-540-35488-8_6
16. Li, J., Cheng, K., Wang, S., Morstatter, F., Trevino, R.P., Tang, J., Liu, H.: Feature selection: A data perspective. arXiv preprint arXiv:1601.07996 (2016)
17. Li, Y., Liu, W., Jia, Y., Dong, H.: A weighted mutual information biclustering algorithm for gene expression data. Computer Science and Information Systtems **14**(3), 643–660 (2017). DOI 10.2298/CSIS170301021Y
18. Mann, H.B., Whitney, D.R.: On a test of whether one of two random variables is stochastically larger than the other. The Annals of Mathematical Statistics **18**(1), 50–60 (1947). DOI 10.2307/2236101
19. Newman, M.E.J.: The structure and function of complex networks. SIAM Review **45**(2), 167–256 (2003). DOI 10.1137/S003614450342480
20. Newman, M.E.J.: Analysis of weighted networks. Physical Review E **70**, 056131 (2004). DOI 10.1103/PhysRevE.70.056131
21. Newman, M.E.J., Girvan, M.: Finding and evaluating community structure in networks. Physical Review E **69**, 026113 (2004). DOI 10.1103/PhysRevE.69.026113
22. Pons, P., Latapy, M.: Computing communities in large networks using random walks. Journal of Graph Algorithms and Applications **10**(2), 191–218 (2006). DOI 10.1007/11569596_31
23. Rand, W.M.: Objective Criteria for the Evaluation of Clustering Methods. Journal of the American Statistical Association **66**(336), 846–850 (1971). DOI 10.2307/2284239
24. Ray, S., Britschgi, M., Herbert, C., Takeda-Uchimura, Y., Boxer, A., Blennow, K., Friedman, L., Galasko, D., Jutel, M., Karydas, A., Kaye, J., Leszek, J., Miller, B., Minthon, L., Quinn, J., Rabinovici, G., Robinson, W., Sabbagh, M., So, Y., Sparks, D., Tabaton, M., Tinklenberg, J., Yesavage, J., Tibshirani, R., Wyss-Coray, T.: Classification and prediction of clinical Alzheimer's diagnosis based on plasma signaling proteins. Nature Medicine **13**(11), 1359–1362 (2007). DOI 10.1038/nm1653

25. Robnik-Šikonja, M., Kononenko, I.: Theoretical and empirical analysis of ReliefF and RReliefF. Machine Learning **53**(1), 23–69 (2003). DOI 10.1023/A:1025667309714
26. Rosvall, M., Bergstrom, C.T.: Maps of information flow reveal community structure in complex networks. Proceedings of the National Academy of Sciences of the United States of America **105**(4), 1118–1123 (2007). DOI 10.1073/pnas.0706851105
27. Sánchez-Maroño, N., Alonso-Betanzos, A., Tombilla-Sanromán, M.: Filter Methods for Feature Selection – A Comparative Study, pp. 178–187. Springer Berlin Heidelberg, Berlin, Heidelberg (2007). DOI 10.1007/978-3-540-77226-2_19
28. Savić, M., Ivanović, M., Radovanović, M., Ognjanović, Z., Pejović, A., Jakšić Krüger, T.: Exploratory analysis of communities in co-authorship networks: A case study. In: A.M. Bogdanova, D. Gjorgjevikj (eds.) ICT Innovations 2014, pp. 55–64. Springer International Publishing, Cham (2015). DOI 10.1007/978-3-319-09879-1_6
29. Savić, M., Ivanović, M., Surla, B.D.: A community detection technique for research collaboration networks based on frequent collaborators cores. In: Proceedings of the 31st Annual ACM Symposium on Applied Computing, SAC '16, pp. 1090–1095. ACM, New York, NY, USA (2016). DOI 10.1145/2851613.2851809
30. Savić, M., Kurbalija, V., Ivanović, M., Bosnić, Z.: A feature selection method based on feature correlation networks. In: Y. Ouhammou, M. Ivanovic, A. Abelló, L. Bellatreche (eds.) Model and Data Engineering, pp. 248–261. Springer International Publishing, Cham (2017). DOI 10.1007/978-3-319-66854-3_19
31. Slavkov, I., Karcheska, J., Kocev, D., Dzeroski, S.: HMC-ReliefF: Feature ranking for hierarchical multi-label classification. Computer Science and Information Systems **15**(1), 187–209 (2018). DOI 10.2298/CSIS170115043S
32. Song, Q., Ni, J., Wang, G.: A fast clustering-based feature subset selection algorithm for high-dimensional data. IEEE Transactions on Knowledge and Data Engineering **25**(1), 1–14 (2013). DOI 10.1109/TKDE.2011.181
33. Van Dijck, G., Van Hulle, M.M.: Speeding Up the Wrapper Feature Subset Selection in Regression by Mutual Information Relevance and Redundancy Analysis, pp. 31–40. Springer Berlin Heidelberg, Berlin, Heidelberg (2006). DOI 10.1007/11840817_4
34. Wilcoxon, F.: Individual Comparisons by Ranking Methods. Biometrics Bulletin **1**(6), 80–83 (1945). DOI 10.2307/3001968
35. Witten, I.H., Frank, E.: Data Mining: Practical Machine Learning Tools and Techniques, Second Edition (Morgan Kaufmann Series in Data Management Systems). Morgan Kaufmann Publishers Inc., San Francisco, CA, USA (2005)
36. Yu, L., Liu, H.: Feature selection for high-dimensional data: A fast correlation-based filter solution. In: T. Fawcett, N. Mishra (eds.) Proceedings of the 20th International Conference on Machine Learning (ICML-03), pp. 856–863 (2003)
37. Zhang, Z., Hancock, E.R.: A Graph-Based Approach to Feature Selection, pp. 205–214. Springer Berlin Heidelberg, Berlin, Heidelberg (2011). DOI 10.1007/978-3-642-20844-7_21
38. Zhao, Z., Liu, H.: Searching for interacting features. In: Proceedings of the 20th International Joint Conference on Artifical Intelligence, IJCAI'07, pp. 1156–1161. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA (2007)