# Feature Selection from Higher Order Correlations

**Zhihong Zhang**

**A Thesis Submitted for the Degree of Doctor of Philosophy**

**Departments of Computer Science**

**University of York**

**Deramore Lane**

**York Y010 5GH**

**October 2012**

**Abstract**

This thesis addresses the problems in feature selection, particularly focusing on selecting features from higher order correlations. To this end, we present two supervised feature selection approaches named *Graph based Information-theoretic Feature Selection* and *Hypergraph based Information-theoretic Feature Selection* respectively, which are capable of considering third or even higher order dependencies between the relevant features and capturing the optimal size of relevant feature subset. Furthermore, we develop two unsupervised feature selection methods which can evaluate features jointly rather than individually. In this case, larger feature combinations are considered. The reason for this is that although an individual feature may have limited relevance to a particular class, when taken in combination with other features it may be strongly relevant to the class.

In Chapter 2, we thoroughly review the relevant literature of the classifier independent (filter-based) feature selection methods. One dominant direction of research in this area is exemplified by the so-called information theoretic feature selection criteria, which is measuring the mutual dependence of two variables. Another influential direction is the graph-based feature selection methods, which are to select the features that best preserve the data similarity or a manifold structure derived from the entire feature set. We notice that most existing feature selection methods evaluate features individually or just simply consider pairwise feature interaction, and hence cannot handle redundant features. Another shortcoming of existing feature selection methods is that most of them select features in a greedy way and do not provide a direct measure to judge whether to add additional features or not. To deal with this problem, they require a user to supply the number of selected features in advance. However, in real applications, it is hard to estimate the number of useful features before the feature selection process. This thesis

addresses these weaknesses, and fills a gap in the literature of selecting features from higher order correlations.

In Chapter 3 we propose a graph based information-theoretic approach to feature selection. There are three novel ingredients. First, by incorporating mutual information (MI) for pairwise feature similarity measure, we establish a novel feature graph framework which is used for characterizing the informativeness between the pair of features. Secondly, we locate the relevant feature subset (RFS) from the feature graph by maximizing features' average pairwise relevance. The RFS is expected to have little redundancy and very strong discriminating power. This strategy reduces the optimal search space from the original feature set to the relatively smaller relevant feature subset, and thus enable an efficient computation. Finally, based on RFS, we evaluate the importance of unselected features by using a new information theoretic criterion referred to as the multi-dimensional interaction information (MII). The advantage of MII is that it can go beyond pairwise interaction and consider third or higher order feature interactions. As a result, we can evaluate features jointly, and thus avoid the redundancies arising in individual feature combinations. Experimental results demonstrate the effectiveness of our feature selection method on a number of standard data-sets.

In Chapter 4, we find that in some situations the graph representation for relational patterns can lead to substantial loss of information. This is because in real-world problems objects and their features tend to exhibit multiple relationships rather than simple pairwise ones. This motive us to establish a feature hypergraph (rather than feature graph) to characterize the multiple relationships among features. We draw on recent work on hypergraph clustering to select the most informative feature subset (mIFS) from a set of objects using high-order (rather than pairwise) similarities. There are two novel ingredients. First, we use MII to measure the significance of different feature combinations with respect to the class labels. Secondly, we use hypergraph clustering to select the most informative feature subset (mIFS), which has both low redundancy and strong discriminating power.

The advantage of MII is that it incorporates third or higher order feature interactions. Hypergraph clustering, which extracts the most informative features. The size of the most informative feature subset (mIFS) is determined automatically. Experimental results demonstrate the effectiveness of our feature selection method on a number of standard data-sets.

In addition to the supervised feature selection methods, we present two novel unsupervised feature selection methods in Chapter $5$ and Chapter $6$. Specifically, we propose a new two-step spectral regression technique for unsupervised feature selection in Chapter $5$. In the first step, we use kernel entropy component analysis (kECA) to transform the data into a lower-dimensional space so as to improve class separation. Second, we use $\ell_1$-norm regularization to select the features that best align with the data embedding resulting from kECA. The advantage of kECA is that dimensionality reducing data transformation maximally preserves entropy estimates for the input data whilst also best preserving the cluster structure of the data. Using $\ell_1$-norm regularization, we cast feature discriminant analysis into a regression framework which accommodates the correlations among features. As a result, we can evaluate joint feature combinations, rather than being confined to consider them individually. Experimental results demonstrate the effectiveness of our feature selection method on a number of standard face data-sets.

In Chapter $6$, by incorporating MII for higher order similarities measure, we establish a novel hypergraph framework which is used for characterizing the multiple relationships within a set of samples (e.g. face samples under varying illumination conditions). Thus, the structural information latent in the data can be more effectively modeled. We then explore a strategy to select the discriminating feature subset on the basis of the hypergraph representation. The strategy is based on an unsupervised method which derive the hypergraph embedding view of feature selection. We develop the strategy based on a number of standard image datasets, and the results demonstrate the effectiveness of our feature selection method.

We summarize the contributions of this thesis in Chapter 7, and analyze the developed methods. Finally, we give some suggestions to the future work in feature selection.

# Contents

# List of Figures

# List of Tables

# Acknowledgements

First, I would like to thank my supervisor, Prof. Edwin Hancock, for his helpful advice and continued support during my research and writing up. His broad knowledge in the field and his down-to-earth attitude have been of great help to my PhD study. I also thank my assessor Dr. William Smith for his constructive feedback on my various reports and presentations.

My thanks go to all the rest of the people in the computer vision group for all the discussions, friendship, and entertainment. Most notably, I thank Dr. Peng Ren and Lichi Zhang for their endless help in study.

Finally, to my parents, I will be forever grateful for their love, encouragement, and unconditional support in every aspect of my life.

# Declaration

I hereby declare that all the work in this thesis is solely my own, except where attributed and cited to another author. Most of the material in this thesis has been previously published by the author. A complete list of publications can be found on page xi.

# List of Publications

## Journal Papers

1. Zhihong Zhang and Edwin R. Hancock. Hypergraph Based Information-theoretic Feature Selection. *Pattern Recognition Letters*, 33: 1991-1999, 2012.

2. Zhihong Zhang and Edwin R. Hancock. Kernel Entropy Based Unsupervised Spectral Feature Selection. *International Journal of Pattern Recognition and Artificial Intelligence* , 26(5), 2012.

## Conference Papers

1. Zhihong Zhang, Edwin R. Hancock and Peng Ren. Unsupervised Feature Selection Via Hypergraph Embedding. In *Proceedings of The 23rd British Machine Vision Conference* (BMVC), 2012.

2. Zhihong Zhang, Edwin R. Hancock and Xiao Bai. Hypergraph Spectra for Semi-supervised Feature Selection. In *Proceedings of The European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases* (ECML-PKDD), 2012.

3. Zhihong Zhang and Edwin R. Hancock. Face Recognition Using Semi-supervised Spectral Feature Selection. In *Proceedings of 21st International Conference on Pattern Recognition* (ICPR), 2012.

4. Zhihong Zhang, Peng Ren and Edwin R. Hancock. Hypergraph Based Semi-supervised Learning for Gender Classification. In *Proceedings of 21st International Conference on Pattern Recognition* (ICPR), 2012.

5. Zhihong Zhang and Edwin R. Hancock. Kernel Entropy Based Unsupervised Spectral Feature Selection. In *Proceedings of 21st International Conference on Pattern Recognition* (ICPR), 2012.

6. Zhihong Zhang and Edwin R. Hancock. Hypergraph Spectra for Unsupervised Feature Selection. In *Proceedings of Joint IAPR International Workshops on Structural and Syntactic Pattern Recognition and Statistical Techniques in Pattern Recognition* (S+SSPR), 2012.

7. Zhihong Zhang and Edwin R. Hancock. Localized Graph-Based Feature Selection for Clustering. In *Proceedings of International Conference on Image Analysis and Recognition* (ICIAR), 2012.

8. Zhihong Zhang, Jing Wu and Edwin R. Hancock. An Information Theoretic Approach to Feature Selection. In *Proceedings of IEEE Workshop on Information Theory in Computer Vision and Pattern Recognition* (ICCV workshop), 2011.

9. Zhihong Zhang and Edwin R. Hancock. Mutual Information Criteria for Feature Selection. In *Proceedings of International Workshop on SIMBAD* (SIMBAD), 2011.

10. Zhihong Zhang and Edwin R. Hancock. A Hypergraphs-Based Approach to Feature Selection. In *Proceedings of 14th International Conference on Computer Analysis of Images and Patterns* (CAIP), 2011.

11. Zhihong Zhang and Edwin R. Hancock. A Graph-based Approach to Feature Selection. In *Proceedings of 8th IAPR-TC-15 International Workshop on Graph-Based Representations in Pattern Recognition* (GbR PR), 2011.

12. Zhihong Zhang and Edwin R. Hancock. Feature Selection for Gender Classification. In *Proceedings of 5th Iberian Conference on Pattern Recognition and Image Analysis* (IbPRIA), 2011.

# Chapter 1

# Introduction

In this chapter we provide an introduction and motivation for the research work presented in this thesis, explaining why we are interested in selecting features from higher order correlations. We commence by introducing the problems encountered in existing classifier independent (filter-based) feature selection methods. Then we briefly describe the possible alternative approaches to these problems, following by our research goals and contributions. Finally, an outline of the thesis is provided at the end of this chapter.

## 1.1 The problems

In many data analysis tasks, one is often confronted with the problem of selecting features from very high dimensional data. In order to render the analysis of high-dimensional data tractable, it is crucial to identify a smaller subset of features that are informative for classification and clustering [57]. Dimensionality reduction aims to reduce the number of variables under consideration, and the process can be divided into feature extraction and feature selection. Feature extraction usually projects the features onto a low-dimensional and distinct feature space, e.g., Locally Linear Embedding (LLE) [63], kernel PCA [10], Locality preserving Projection (LPP) [74], Neighborhood Preserving Embedding (NPE) [75] and Laplacian eigenmap [44]. Unlike feature extraction, feature selection identi-

fies the optimal feature subset in the original feature space. By maintaining the original features, feature selection improves the interpretability of the data, which is preferred in many real world applications, such as face recognition and text mining [40].

The feature selection problem is essentially a combinatorial optimization problem which is computationally expensive. To overcome this problem, most existing filter-based feature selection methods focus on ranking individual features based on a utility criterion, and select the optimal feature set in a greedy manner. However, the feature combinations found in this way has four limitations which cannot give optimal classification performance [67] [66].

The first is that they evaluate features individually and hence cannot lead to redundant features being evaluated. Redundant features increase the dimensionality unnecessarily [48], and degrade learning performance when faced with a shortage of data. It is also shown empirically that removing redundant features can result in significant performance improvement [37].

The second weakness is that they assume that each individual relevant feature should be dependent with the target class. This means that if a single feature is considered to be relevant it should be correlated with the target class, otherwise the feature is irrelevant [28]. So only a small set of relevant features is selected, and larger feature combinations are not considered. The reason for this is that although an individual feature may have limited relevance to a particular class, when taken in combination with other features it may be strongly relevant to the class.

The third weakness is that most of existing methods select features in a greedy way and do not provide a direct measure to judge whether to add additional features or not. To deal with this problem, they require a user to supply the number of selected features in advance. However, in real applications, it is hard to estimate the number of useful features before the feature selection process.

Finally, most of the methods simply consider pairwise feature dependencies, and do

not check for third or higher order dependencies between the candidate features and the existing features. Thus, optimal feature subset cannot be located.

## 1.2 Our Goals

The overall goal of this thesis is to develop novel classifier independent (filter-based) feature selection methods addressing the problems shown above. Specifically,

i) We aim to develop a feature selection criterion which is able to measure the significance of different feature combinations. In particular, it can go beyond pairwise feature interaction and consider third or even higher order dependencies between the relevant features. Hence, we can evaluate features jointly rather than individually. Thus we are able to handle feature redundancy.

ii) We aim to select an optimal feature subset, where the size of the feature subset can be automatically determined.

iii) We aim to develop a novel framework for unsupervised feature selection which considers the correlations among features. In this case, it is very possible that the combination of several "week" features can better differentiate different clusters, although they are not very informative in differentiating different clusters if evaluated independently.

## 1.3 Contributions

To achieve the research goals described in Section 1.2, we make the following specific contributions:

### 1.3.1 Graph based Information-theoretic Feature Selection

In Chapter 3, we propose a new information theoretic criterion referred to as the multidimensional interaction information (MII) to measure the significance of different feature

combinations. The advantage of MII is that it is sensitive to the relations between feature combinations. As a result it can be used to seek third or even higher order dependencies between the relevant features. Hence, we can evaluate features jointly rather than individually [82]. Thus we are able to handle feature redundancy. However, MII involves evaluating all possible interactions among the selected features which has two problems: 1) it requires an exhaustive "combinatorial" search over the feature space, and 2) it demands large training sample sizes to estimate the higher order joint probability distribution in MII with a high dimensional kernel [49].

To reduce the search space in using MII, we propose a graph-based feature selection algorithm consisting of three steps, namely, i) by incorporating mutual information (MI) for pairwise feature similarity measure, we first establish a novel feature graph framework which is used for characterizing the informativeness between the pair of features, ii) we then extract the relevant feature subset (RFS) from the feature graph by maximizing features' average relevance. The main property of RFS is that the overall relevance among the internal features is greater than that between the external feature and the internal features, iii) based on RFS, we evaluate the importance of unselected features by using MII. In this feature selection scheme, we commence by extracting the relevant feature subset (RFS) from the initial features as a pre-processing step for ranking features. This strategy reduces the optimal search space from the original feature set to the relatively smaller relevant feature subset, and thus enable an efficient computation. In addition, the size of the relevant feature subset is determined automatically.

However, in some situations the graph representation for relational patterns can lead to substantial loss of information. This is because in real-world problems objects and their features tend to exhibit multiple relationships rather than simple pairwise ones. This motivates our work in Chapter 4.

### 1.3.2 Hypergraph based Information-theoretic Feature Selection

A natural way of remedying the information loss described in Chapter 3 is to represent the features as hypergraph instead of a graph. In Chapter 4, we propose to use a hypergraph-based feature selection algorithm consisting of two steps [83]. Firstly, we construct a hypergraph in which each node corresponds to a feature, and each edge has a weight corresponding to the MII among features connected by that edge. Secondly, we apply hypergraph clustering to the hypergraph in order to locate the most informative feature subset (mIFS), which has both low redundancy and strong discriminating power. In contrast with existing feature selection methods, our proposed methods is able to determine the number of relevant features automatically.

The proposed feature selection methods in Chapter 3 and Chapter 4 are supervised feature selection methods. While the labeled data required by supervised feature selection can be scarce, there is usually no shortage of unlabeled data. Hence, there are obvious attractions in developing unsupervised feature selection algorithms which can utilize this data. Therefore, in the following two chapters (Chapter 5 and Chapter 6), we extend our attention to unsupervised feature selection methods.

### 1.3.3 Kernel Entropy Analysis for Unsupervised Feature Selection

Feature selection for unsupervised learning is difficult because, without class labels, it is hard to assess the relevance of a feature or a subset of features. In Chapter 5, we develop a novel regularization based unsupervised feature selection method for feature subset selection. The idea underpinning our proposed method is to select the features which best preserve the cluster structure derived from the entire feature set. Specifically, we propose a new two-step spectral regression technique for unsupervised feature selection. In the first step, we use kernel entropy component analysis (kECA) to transform the data into a lower-dimensional space so as to improve class separation. Second, we use $\ell_1$-norm

regularization to select the features that best align with the data embedding resulting from kECA. The advantage of kECA is that dimensionality reducing data transformation maximally preserves entropy estimates for the input data whilst also best preserving the cluster structure of the data. Using $\ell_1$-norm regularization, we cast feature discriminant analysis into a regression framework which accommodates the correlations among features. As a result, we can evaluate joint feature combinations, rather than being confined to consider them individually.

### 1.3.4 Hypergraph Spectral Analysis for Unsupervised Feature Selection

In Chapter 6, by incorporating MII for higher order similarities measure, we establish a novel hypergraph framework which is used for characterizing the multiple relationships within a set of samples (e.g. face samples under varying illumination conditions). Thus, the structural information latent in the data can be more effectively modeled. Then an unsupervised method is proposed to find the discriminating feature subset on the basis of hypergraph representation. For the unsupervised learning, we derive a hypergraph embedding view of feature selection, where the projection matrix is constrained to be a selection matrix designed to select the optimal feature subset. Experimental results demonstrate the effectiveness of our feature selection methods on a number of standard image datasets.

## 1.4 Thesis Outline

The rest of the thesis is organized as follows: In Chapter 2, we give a thorough review of the relevant literature. We commence by discussing the information-theoretic based approaches for feature selection. Then, we extend our attention to graph based feature

selection methods. In Chapter 3, we present our first attempt to select the discriminating features by a new criterion referred to as MII; Furthermore, to reduce the search space in using MII, we introduces a graph based information theoretic to feature selection; Chapter 4 describes a hypergraph based information theoretic to feature selection, which can be more effective in representing multiple relationships among features; Chapter 5 presents an unsupervised regularization based feature selection method using kernel entropy analysis; Chapter 6 introduces hypergraph spectral analysis for unsupervised feature selection; Finally, in Chapter 7, we summarize the contributions of this thesis, discuss the weaknesses in the work, and suggest avenues for future work.

# Chapter 2

# Literature Review

Feature selection can be divided into two categories, i) filter methods [30] where the feature selector is independent of classifiers, ii) wrapper methods [58] [80] utilize the classifier (i.e. support vector machine recursive feature selection referred as SVM-RFE [32]) to evaluate each possible feature subset by the estimated accuracy. Although wrapper approaches usually have good performance, their computational cost is very expensive when the number of features is large. This is because a learning algorithm is employed to evaluate each and every set of features considered, wrappers are prohibitively expensive to run, and can be intractable for large databases containing many features. In this thesis, we focus on exploring filter approaches to feature selection, which are model-independent criteria that provide a ranking of the features. We commence in Section 2.1 with a review of the mutual information (MI) based feature selection method, which is the most popular filter approach to feature selection. Additionally, we also describe the shortcomings of existing MI-based methods which motivate us to select features from higher order correlations. We then review the graph-based feature selection methods in Section 2.2, followed by an overview of hypergraph representation for structural pattern recognition in Section 2.3. Finally, Section 2.4 concludes the chapter.

## 2.1 Information theoretic based Feature Selection Methods

High-dimensional data pose a significant challenge for pattern recognition [51]. The most popular methods for reducing dimensionality are variance based subspace methods such as PCA [31]. However, the extracted PCA feature vectors only capture sets of features with a significant combined variance, and this renders them relatively ineffective for classification tasks. Hence it is crucial to identify a smaller subset of features that are informative for classification and clustering. Recently, mutual information has been shown to provide a principled way of measuring the mutual dependence of two variables, and has been used by a number of researchers to develop information theoretic feature selection criteria. We present a selection of the most well-known criteria as below:

### 2.1.1 MIFS

**Battiti** [55] has developed the Mutual Information-Based Feature Selection (MIFS) criterion,

$$J_{mifs} = I(f_i; C) - \beta \sum_{f_s \in S} I(f_s; f_i) \ . \tag{2.1}$$

It is used to select the most relevant $m$ features from an initial set of $d$ features and the features are selected in a greedy manner. Given a set of existing selected features $S$, at each step it locates the candidate feature $f_i$ that maximize the relevance to the class $I(f_i; C)$ without considering the joint MI between the selected feature set and the output class $C$. The selection is regulated by a proportional term $\beta I(f_i; S)$ that measures the overlap information between the candidate feature and existing features. The parameter $\beta$ may significantly affect the features selected, and its control remains an open problem. It will overestimate the redundancy between features in the case where $\beta$ is too large.

It can be seen that the MIFS algorithm only consider those features that have maximum MI with the output classes, and are less dependent. However, these feature combinations cannot produce an optimal feature subset, since they are possibly discarding "redundant" features which have much information about the output class and selecting irrelevant features.

### 2.1.2 MIFS-U

**Kwak and Choi** [49] [50] improve MIFS by developing MIFS-U under the assumption of a uniform distribution of information for selected features $S$. This can be defined as,

$$J_{mifs-u} = I(f_i; C) - \beta \sum_{f_s \in S} \frac{I(f_s; C)}{H(f_s)} I(f_i; f_s) \ . \tag{2.2}$$

where $H(f_s) = -\sum_{f_s \in S} P(f_s) \log P(f_s)$ is the entropy. The uniform probability distribution assumption can make sure conditioning by the class $C$ does not change the ratio of the entropy of $f_s$ and the mutual information between $f_s$ and $f_i$.

This criterion makes better estimation than MIFS which considering the conditional MI $I(C; f_i|S)$ between output class $C$ and the candidate feature $f_i$ for a given selected features $S$. However, instead of calculating $I(C; f_i|S)$ directly, only $I(S; f_i)$ and $I(C; f_i)$ are computed, where the conditional MI $I(C; f_i|S)$ can be approximated as

$$I(C; f_i|S) = I(C; f_i) - \{I(S; f_i) - I(S; f_i|C)\} \ . \tag{2.3}$$

MIFS-U makes a better estimation of the MI criterion than MIFS, but it also needs to carefully choose the parameter $\beta$. With an unproper value of $\beta$, the algorithm may produce bad results.

### 2.1.3 MRMR

**Peng et al.** [27] on the other hand, propose a parameter-free method (referred to as Maximum-Relevance Minimum-Redundancy criterion (MRMR)), which is equivalent to MIFS with $\beta = \frac{1}{|S|}$. It is defined as,

$$J_{mrmr} = I(f_i; C) - \frac{1}{|S|} \sum_{f_s \in S} I(f_i; f_s) \ . \tag{2.4}$$

where $|S|$ is the cardinality of the selected feature set $S$. It takes the average of the redundancy term, which is used to eliminate the difficulty of parameter $\beta$ selection with MIFS and MIFS-U approaches. The unproper value of $\beta$ in MIFS and MIFS-U will make the relevance term (first term in Equation (2.1) and Equation (2.2)) and the redundancy term (second term in Equation (2.1) and Equation (2.2)) in the substraction unbalance. This is due to the fact that the redundancy term (second term in Equation (2.1) and Equation (2.2)) is a cumulative sum, it will grow in magnitude with respect the the relevance term (first term), as the cardinality of the subset of selected features increases. When the relevance term (first term) becomes negligible with respect to the redundancy term (second term), the feature selection algorithm tend to select features based on minimum redundancy. This may cause the selection of irrelevant features. Although the MRMR algorithm solve the unbalance problem on some degree by averaging the feature-feature mutual information in the second term of the substraction, it also omits the conditional MI $I(C; f_i|S)$ between output class $C$ and the candidate feature $f_i$ for a given selected features $S$. The MRMR is a first-order incremental feature selection method, which assuming that each feature independently influences the output class $C$. The MRMR also can be effectively combined with wrapper schemes into a two-stage selection algorithm. In the first stage, the MRMR method is used to locate a candidate feature set. In the second stage, the backward and forward selections are used to search a compact feature subset from the candidate feature set that minimizes the classification error. However, as the first-order assumption, MRMR presents similar limitations as MIFS and MIFS-U in

the presence of many irrelevant and redundant features.

### 2.1.4 NMIFS

**Estevez et al.** [53] develop an improved version of MRMR, called NMIFS, which is dividing the normalized feature-feature mutual information to achieve a balance between the relevance and the redundancy term as below,

$$J_{nmifs} = I(f_i; C) - \frac{1}{|S|} \sum_{f_s \in S} \hat{I}(f_i; f_s) \ . \tag{2.5}$$

where $\hat{I}$, the normalized mutual information, is defined as,

$$\hat{I}(f_i; f_s) = \frac{I(f_i; f_s)}{\min(H(f_s), H(f_i))} \ . \tag{2.6}$$

It can be seen that NMIFS used normalized mutual information to overcome the unbalance problem between relevance and redundancy term in MIFS, MIFS-U and MRMR algorithm.

### 2.1.5 JMI

**Yang and Moody's** [29] Joint Mutual Information (JMI) criterion is based on conditional MI,

$$\begin{aligned} J_{jmi} &= \sum_{f_s \in S} I(f_i f_s; C) \\ &= I(f_i, C) - \frac{1}{|S|} \sum_{f_s \in S} [I(f_i, f_s) - I(f_i; f_s | C)] \ . \end{aligned} \tag{2.7}$$

It selects features by checking whether they bring additional information to an existing feature set. This method effectively rejects redundant features. The JMI criterion is MRMR criterion plus $\frac{1}{|S|} \sum_{f_s \in S} I(f_i; f_s|C)$. The JMI criterion, like MRMR, has a strong belief in the pairwise independence assumptions as the selected feature set $S$ grows.

## 2.2 Graph based Feature Selection Methods

Recently, graph-based methods, such as spectral embedding [44], spectral clustering [33], and semi-supervised learning [8] [21], have played an important role in machine learning due to their ability to encode the similarity relationships among data. Various applications of graph-based methods can be found in clustering [33] [70], data mining [56], manifold learning [76] [45], subspace learning [73] and speech recognition [23]. A preliminary step for all these graph-based methods is to establish a graph over the training data. Data samples are represented as vertices of the graph and the edges represent the pairwise similarity relationships between them. In feature selection, the attractive feature of graph representations is that they provide a universal and flexible framework that reflects the underlying manifold structure and the relationships between feature vectors. A frequently used criterion in graph-based feature selection methods is to select the features which best preserve the data similarity or a manifold structure derived from the entire feature set. The best known methods are the Laplacian score [73], SPEC [81], Fisher score [12] and Trace ratio [24].

### 2.2.1 Laplacian Score

Laplacian score [73] uses a k-nearest neighbor graph to model the local geometric structure of the data and selects the features most consistent with the graph structure. Consider a dataset $\mathbf{X} = x_1, \ldots, x_N$. In order to approximate the manifold structure of dataset, a $k$-nearest neighbor graph is built, which contains an edge with weight $\mathbf{W}_{ij}$ between $x_i$ and $x_j$ if $x_i$ is among the $k$ nearest neighbors of $x_j$ or conversely. There are different similarity based methods that can be used to determine the edge weights. In general, the Euclidean distance [15] is widely used as similarity measure. Therefore, the weight matrix $\mathbf{W}$ can

be defined as below,

$$\mathbf{W}_{ij} = \begin{cases} e^{-\frac{\|x_i - x_j\|^2}{t}}, & \text{if } x_i \text{ and } x_j \text{ are neighbors} \\ 0, & \text{otherwise.} \end{cases} \tag{2.8}$$

where $t$ is a suitable constant. A feature that is consistent with the graph structure can be thought of as the one on which two data points are close to each other if and only if there is an edge between these two points. Let $f_{ri}$ denote the $i$-th sample of the $r$-th feature and $f_r = (f_{r1}, \ldots, f_{rN})^T$. To select a good feature, we need to minimize the following objective function:

$$SC_{Ls} = \frac{\sum_{ij}(f_{ri} - f_{rj})^2 \mathbf{W}_{ij}}{Var(f_r)} \ . \tag{2.9}$$

where $Var(f_r)$ is the estimated variance of the $r$-th feature. Features with larger variance are preferred, as they are expected to have more representative power. Given $\mathbf{W}$, its corresponding degree matrix $\boldsymbol{D}_{ii} = \sum_j \mathbf{W}_{ij}$ and Laplacian matrix $\boldsymbol{L} = \boldsymbol{D} - \mathbf{W}$, the variance of weight data can be calculated based on $\boldsymbol{D}$ which modeling the importance of the data points.

$$Var(f_r) = \tilde{f}_r^T \boldsymbol{D} \tilde{f}_r \ , \tag{2.10}$$

where

$$\tilde{f}_r = f_r - \frac{f_r^T \boldsymbol{D} \mathbf{1}}{\mathbf{1}^T \boldsymbol{D} \mathbf{1}} \mathbf{1} \ , \tag{2.11}$$

Here, we remove the mean of each feature $f_r$ by Equation (2.11). This is done to prevent a non-zero constant vector such as $\mathbf{1}$ to be assigned a zero Laplacian score as such a feature obviously does not contain any information.

For a good feature, the bigger $W_{ij}$, the smaller $(f_{ri} - f_{rj})$, and thus it is easy to see that,

$$\sum_{ij}(f_{ri} - f_{rj})^2 \mathbf{W}_{ij} = 2 f_r^T \boldsymbol{L} f_r = 2 \tilde{f}_r^T \boldsymbol{L} \tilde{f}_r \ , \tag{2.12}$$

Finally, the Laplacian score of the $r$-th feature is reduced to

$$SC_{Ls}(f_r) = \frac{\tilde{f}_r^T \boldsymbol{L} \tilde{f}_r}{\tilde{f}_r^T \boldsymbol{D} \tilde{f}_r} \,, \tag{2.13}$$

### 2.2.2 SPEC

The SPEC [81] algorithm is an extension for Laplacian score to make it more robust to noise. In SPEC, given the affinity matrix $\boldsymbol{A}$, the degree matrix $\boldsymbol{D}$, and the normalized Laplacian matrix $\boldsymbol{L}$, three evaluation criteria are proposed for measuring feature relevance in the following ways:

$$SC_{Spec,1}(f_i) = \hat{f_i}^T \gamma(\boldsymbol{L}) \hat{f_i} = \sum_{j=1}^{N} \alpha_j^2 \gamma(\lambda_j) \,, \tag{2.14}$$

$$SC_{Spec,2}(f_i) = \frac{\hat{f_i}^T \gamma(\boldsymbol{L}) \hat{f_i}}{1 - (\hat{f_i}^T \xi_j)^2} = \frac{\sum_{j=1}^{N} \alpha_j^2 \gamma(\lambda_j)}{\sum_{j=1}^{N} \alpha_j^2} \,, \tag{2.15}$$

$$SC_{Spec,3}(f_i) = \sum_{j=1}^{k} (\gamma(2) - \gamma(\lambda_j)) \alpha_j^2 \,. \tag{2.16}$$

In the above equations, $\hat{f_i} = (\boldsymbol{D}^{\frac{1}{2}} f_i) \cdot \|(\boldsymbol{D}^{\frac{1}{2}} f_i)\|^{-1}$; $(\lambda_j, \xi_j)$ is the $j$-th Eigen-pair of $\boldsymbol{L}$; $\alpha_j = \cos\theta_j$, where $\theta_j$ is the angle between $\hat{f_i}$ and $\xi_j$; and $\gamma(\cdot)$ is an increasing function which is used to re-scale the eigenvalues of $\boldsymbol{L}$ for de-noising. The top eigenvectors of $\boldsymbol{L}$ are the optimal soft cluster indicators of the data [69]. By comparing with these eigenvectors, SPEC selects features that assign similar values to instances that are similar according to $\mathbf{W}$. In [81], it is shown that Laplacian score is a special case of the second criterion, $SC_{Spec,2}(\cdot)$, defined is SPEC. Note that SPEC also evaluates feature independently.

### 2.2.3 Fisher Score

In contract to Laplacian score and SPEC, Fisher score is supervised with class label and it seeks feature subsets which preserve the discriminative ability. Given class labels, Fisher

score [12] selects features that assign similar values to data points from the same class and different values to data points from different classes. Let $\mu_{i,j}$ and $\sigma^2_{i,j}$ be the mean and variance of feature $f_i$ on class $j$, $j = 1, \ldots, C$, respectively. $\mu_i$ is the mean of the feature $f_i$ and $n_j$ is the number of samples in class $j$. Then the Fisher score of the $i$-th feature can be formulated as

$$SC_{Fs}(f_i) = \frac{\sum_{j=1}^{C} n_j(\mu_{i,j} - \mu_i)^2}{\sum_{j=1}^{C} n_j \sigma^2_{i,j}} , \tag{2.17}$$

After computing the Fisher score for each feature, it selects the top $m$ ranked features with large scores. Because the score of each feature is computed independently, the features selected by the heuristic algorithm is suboptimal. More importantly, the heuristic algorithm fails to select those features which have relatively low individual scores but a very high score when they are combined together as a whole. In addition, it cannot handle redundant features. In [73], it is shown that Fisher score is a special case of Laplacian score, when the similarity matrix is defined as

$$\mathbf{S}_{ij} = \begin{cases} \frac{1}{n_l}, & y_i = y_j = l \\ 0, & \text{otherwise.} \end{cases} \tag{2.18}$$

where $n_l$ is the number of instances in the $l$-th class.

### 2.2.4 Trace Ratio

The trace ratio criterion [24] is proposed to locate a feature subset for which the within class pairwise affinities are large, while the between class separation is large. In order to discover both geometrical and discriminant structure of the data manifold, it constructs two weighted graphs to capture the similarity structure of the data. The first is the intra-class or within class similarity graph $G_w(X, \mathbf{W}_w)$, while the second is the inter or between class similarity graph $G_b(X, \mathbf{W}_b)$. The within-class similarity graph $G_w$ is characterized by the weight matrix $\mathbf{W}_w$ and reflects the interclass compactness of the data, while $G_b$

can be regarded as a between class penalty graph, characterized by the weight matrix $\mathbf{W}_b$ which reflects the intraclass separability. The two weight matrices $(\mathbf{W}_w)_{ij}$ and $(\mathbf{W}_b)_{ij}$ are respectively determined by the within class and between class pairwise similarity of instances. When $(\mathbf{W}_w)_{ij}$ is large, this implies that data $x_i$ and data $x_j$ belong to same class and a small value indicates they belong to different classes. Similarly, since $(\mathbf{W}_b)_{ij}$ represents the global between class affinity relationships in the data, it provides a heavy penalty if data $x_i$ and $x_j$ belong to different classes. These features can be captured if the weight matrices $\mathbf{W}_w$ and $\mathbf{W}_b$ are defined as follows

$$(\mathbf{W}_w)_{ij} = \begin{cases} \frac{1}{N_{c(i)}}, & \text{if } c(i) = c(j); \\ 0, & \text{if } c(i) \neq c(j). \end{cases} \tag{2.19}$$

$$(\mathbf{W}_b)_{ij} = \begin{cases} \frac{1}{N} - \frac{1}{N_{c(i)}}, & \text{if } c(i) = c(j); \\ \frac{1}{N}, & \text{if } c(i) \neq c(j). \end{cases} \tag{2.20}$$

where $c(i)$ represents class label of data point $x_i$, and $N_{c(i)}$ dentes the number of data in class $i$.

The trace ratio criterion works with the Laplacian matrices for the graphs $G_w$ and $G_b$. To this end let $\boldsymbol{D}_b$ and $\boldsymbol{D}_w$ denote the diagonal matrices of $\mathbf{W}_b$ and $\mathbf{W}_w$, where $(\boldsymbol{D}_b)_{ii} = \sum_{k=1}^{N}(\mathbf{W}_b)_{ik}$ and $(\boldsymbol{D}_w)_{ii} = \sum_{k=1}^{N}(\mathbf{W}_w)_{ik}$. The weighted within-class degree of node $i$, i.e. $(\boldsymbol{D}_w)_{ii}$ provides a natural measure of the density of data in the proximity of the data point $x_i$. Since the more data points that are close to $x_i$, the larger the weighted degree $(\boldsymbol{D}_w)_{ii}$, the more important the point $x_i$. From the weight matrices $\mathbf{W}_b$ and $\mathbf{W}_w$, and the degree matrices $\boldsymbol{D}_b$ and $\boldsymbol{D}_w$, the corresponding between class and within class Laplacian matrices are $\boldsymbol{L}_b = \boldsymbol{D}_b - \mathbf{W}_b$ and $\boldsymbol{L}_w = \boldsymbol{D}_w - \mathbf{W}_w$ respectively. The optimal feature subsets should be the those for which the within class pairwise affinities are large, while the between class separation is large. These features are captured by selecting the set of features that minimize $\sum_{ij} \|x_i - x_j\|^2(\mathbf{W}_w)_{ij}$ and while maximizing $\sum_{ij} \|x_i - x_j\|^2(\mathbf{W}_b)_{ij}$. To achieve the above tow objective functions, the trace ratio criterion seeks

the best selection matrix $\Phi$ by maximizing the following criterion:

$$\Phi^* = \arg\max \frac{\sum_{i \neq j} \parallel \Phi^T(x_i - x_j) \parallel^2 (\mathbf{W}_b)_{ij}}{\sum_{i \neq j} \parallel \Phi^T(x_i - x_j) \parallel^2 (\mathbf{W}_w)_{ij}} = \max \frac{tr(\Phi^T X \boldsymbol{L}_b X^T \Phi)}{tr(\Phi^T X \boldsymbol{L}_w X^T \Phi)} \,. \tag{2.21}$$

For the sake of simplicity, we denote $B = X\boldsymbol{L}_b X^T$ and $E = X\boldsymbol{L}_w X^T$. Suppose the subset-level score in Equation (2.21) reaches the global maximum $\zeta^*$ if $\Phi = \Phi^*$, that is to say,

$$\frac{tr(\Phi^{*T} B \Phi^*)}{tr(\Phi^{*T} E \Phi^*)} = \zeta^* \,. \tag{2.22}$$

and

$$\frac{tr(\Phi^T B \Phi)}{tr(\Phi^T E \Phi)} \leq \zeta^* \,. \tag{2.23}$$

From Equation (2.23), we can derive that

$$\max_{\Phi} tr(\Phi^T (B - \zeta^* E)\Phi) \leq 0 \,. \tag{2.24}$$

Note that $tr(\Phi^{*T}(B - \zeta^* E)\Phi^*) = 0$ and let

$$f(\zeta) = \max tr(\Phi^T (B - \zeta E)\Phi) \,. \tag{2.25}$$

then we have $f(\zeta^*) = 0$. As $f(\zeta)$ is a monotonically decreasing function, finding the global optimal $\zeta^*$ can be converted into the problem of locating the single root of equation $f(\zeta) = 0$. Here, we define score of the $i$-th feature as

$$SC_{Tr}(f_i) = \Phi_i^T (B - \zeta E)\Phi_i \,. \tag{2.26}$$

The function $f(\zeta)$ can be rewritten as

$$f(\zeta) = \max \sum_{i=1}^{m} \Phi_i^T (B - \zeta E)\Phi_i \,. \tag{2.27}$$

Thus $f(\zeta)$ equals to the sum of the first $m$ largest scores. The task of subset-level based feature selection is to seek the feature subset with the maximum score according to Equation (2.22). The root can be located using an iterative procedure to update $\zeta$ and thus find the root of equation $f(\zeta) = 0$.

Although the trace ratio criterion evaluates a set of features jointly, it does not take feature redundancy into account and is prone to selecting redundant or even duplicated features.

## 2.3 Hypergraph Representation for Pattern Recognition

In many situations the graph representation for relational patterns can lead to substantial loss of information. This is because in real-world problems objects and their features tend to exhibit multiple relationships rather than simple pairwise ones. For example, consider the problem of classifying faces which are under different lighting conditions. See Fig. 2.1 for an illustration. It is well known that images of the same objects may look drastically different under different lighting conditions [78] [17]. In this scenario, pairwise similarity measures for images of the same person may exhibit great variety. This misleading result is due to the fact that the set of images of a Lambertian surface under arbitrary lighting lies on a 3D liner subspace in the image space [54] where multiple relationships exist, and the higher order relations cannot be suitably characterized by pairwise similarity measures.

A natural way of remedying the information loss described above is to represent the data set as a hypergraph instead of a graph. Hypergraph representations allow vertices to be multiply connected by hyperedges and can hence capture multiple or higher order relationships between features. Due to their effectiveness in representing multiple relationships, hypergraph based methods have been applied to various practical problems, such as partitioning circuit netlists [38], clustering [62, 19], clustering categorial data [18], and image segmentation [61]. For multi-label classification, Sun et al. [41] construct a

Figure 2.1: Shown above are images of five persons under varying illumination conditions. Is it possible to group them into clusters based on pairwise similarity measure?

hypergraph to exploit the correlation information contained in different labels. In this hypergraph, instances correspond to the vertices and each hyperedge includes all instances annotated with a common label. With this hypergraph representation, the higher-order relations among multiple instances sharing the same label can be explored. Following the theory of spectral graph embedding [21], they transform the data into a lower-dimensional space through a linear transformation, which preserves the instance-label relations captured by the hypergraph. The projection is guided by the label information encoded in the hypergraph and a linear Support Vector Machine (SVM) is used to handle the multi-label classification problem. Huang et al. [79] used a hypergraph cut algorithm [19] to solve the unsupervised image categorization problem, where a hypergraph is used to represent the complex relationship among unlabeled images based on shape and appearance features. Specifically, they first extract the region of interest (ROI) of each image, and then construct hyperedges among images based on shape and appearance features in their ROIs. Hyperedges are defined as either a) a group formed by each vertex (image) or b)

its k-nearest neighbors (based on shape or appearance descriptors). The weight of each hyperedge is computed as the sum of the pairwise affinities within the hyperedge. In this way, the task of image categorization is transferred into a hypergraph partition problem which can be solved using the hypergraph cut algorithm [59].

One common feature of these existing hypergraph representations is that they exploit domain specific and goal directed representations. Specifically, most of them are confined to uniform hypergraphs and do not lend themselves to generalization. The reason for this lies in the difficulty in formulating a nonuniform hypergraph in a mathematically neat way for computation. These has yet to be a widely accepted and consistent way for representing and characterizing nonuniform hypergraphs, and this remains an open problem when exploiting hypergraphs for feature selection. Moreover, to be easily manipulated, hypergraphs must be represented in a mathematically consistent form, using structures such as matrices or vectors.

Since Chung's [22] definition of the Laplacian matrix for K-uniform hypergraphs, there have been several attempts to develop matrix representations of hypergraphs. To establish the adjacency matrix and Laplacian matrix for a hypergraph, an equivalent graph representation is often required. Once the graph approximation is to hand, its graph representation matrices are often referred to as the corresponding hypergraph representation matrices. Based on these approximate matrix representations, subsequent hypergraph processing (e.g., hypergraph embedding) is performed. In machine learning, Agarwal et al. [62] have compared a number of alternative graph representations [47, 71, 34, 19] for hypergraphs and also explained their interrelationships. One common feature for these methods, as well as the method in [61], is that a weight is assumed to be associated with each hyperedge. The available graph representations for a hypergraph can be classified into two categories: a) the clique expansion [61, 47, 34], b) the star expansion [71, 19].

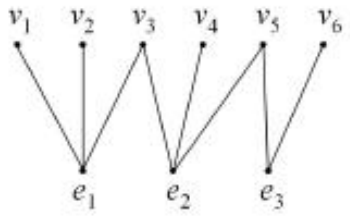



Figure 2.2: Bipartite graph

### 2.3.1 Star Expansion

The star expansion represents a hypergraph by introducing a new vertex for every hyperedge, and then constructing a graph with those existing vertices within a hyperedge connected to the newly introduced vertex. Specifically, for a hypergraph $G = (V, E)$, the star expansion constructs a graph $G^* = (V^*, E^*)$ from the original hypergraph, where $V^* = V \cup E$ and $E^* = (u, e) : u \in e, e \in E$. Thus each hyperedge in $G$ is expanded into a star in $G^*$, which is bipartite graph, see Fig. 2.2. The weight $\mathbf{W}^*(u, e)$ of an edge $(u, e)$ in $G^*$ is given by

$$\mathbf{W}^*(u, e) = \frac{\mathbf{W}(e)}{\delta(e)} \, . \tag{2.28}$$

where $\mathbf{W}(e)$ is the weight associated with the hyperedge $e \in E$ and $\delta(e)$ is the degree of hyperedge $e$ which is the number of vertices in $e$.

### 2.3.2 Clique Expansion

The clique expansion assumes that the hyperedge weight and edge weights are equal to each other. In the clique expansion, each hyperedge is expanded into a clique. Denote by $G_c = (V_c, E_c)$ the 2-graph expanded from hypergraph $G = (V, E)$ using the clique expansion. The relationship between a hyperedge and the edge weights in the clique is given by

$$\mathbf{W}_c(v_i, v_j) = \frac{1}{\mu(n, k)} \sum_{v_i, v_j \in e} \mathbf{W}(e) \, . \tag{2.29}$$

where $\mu(n,k) = \binom{n-2}{k-2}$ is the number of hyperedges that contain a particular pair of vertices and $k$ is the size of the hyperedge.

In the above two strategies, each edge in each individual graph representation is weighted in a manner determined by the corresponding hyperedge weight in a task-specific way. Moreover, these graph-based representations for hypergraphs are just approximations, and hence give rise to information loss. This deficiency may result in ambiguities when approximation methods are used to distinguish structures with different relational orders.

To address these shortcomings, an effective matrix representation for hypergraphs is needed, such that the low-pass information loss in the process of averaging hypergraph weights can be overcome. To this end, we use clique averaging [61] to approximate a hypergraph. This is closely related to the clique expansion. However it is able to precisely preserve information contained in the original hypergraph. According to this scheme, the relationship between a hyperedge weight and its related simple graph weights is determined by a particular generative model $F$. How well the graph $G_c$ captures the structure of hypergraph $G$ is now a function of $F$, i.e. the hyperedge weight is given by

$$\mathbf{W}(e) = F(\mathbf{W}_c(v_1, v_2), \ldots, \mathbf{W}_c(v_i, v_j), \ldots, \mathbf{W}_c(v_{k-1}, v_k)) \; . \tag{2.30}$$

where the function $F$ should satisfy three conditions, namely, i) positivity, ii) symmetry and iii) monotonicity. We can now write Equation (2.30) as

$$\mathbf{W}(e) = \begin{pmatrix} k \\ 2 \end{pmatrix} \sum_{v_i, v_j \in e, i<j} \mathbf{W}_c(v_i, v_j) \; . \tag{2.31}$$

The above equation states that the $L_1$ norm for the clique weights is proportional to the hyperedge weight. Without loss of generality we will assume that the set of hyperedges has been ordered in a lexicographic order based on the vertices incident on each hyperedge. A similar ordering is done on the set of graph edges too. We can now define the zero-one incidence matrix $\boldsymbol{H}$, that represents the incidence relationship between a hyperedge in a

hypergraph and the edge in the related simple graph.

$$\boldsymbol{H}_{i,j} = \begin{cases} 1 & \text{if edge j is incident on hyperedge } i \\ 0 & \text{otherwise.} \end{cases} \tag{2.32}$$

Denote by $\mathbf{W}_2$ the vector of graph edge weights of length $\binom{n}{2}$ and, denote by $\mathbf{W}_k$ the vector of hyperedge weights. Then Equation (2.31) can be written in matrix form as

$$\begin{pmatrix} k \\ 2 \end{pmatrix} \boldsymbol{H}\mathbf{W}_2 = \mathbf{W}_k \ . \tag{2.33}$$

This equation assumes that $\mathbf{W}_2 \geq 0$, i.e., each element of the vector $\mathbf{W}_2$ is non-negative. If we enforce an upper bound $\mathbf{W}_2 \leq 1$ also, the graph approximation of hypergraph is given by the edge weight vector $\mathbf{W}_2$ that satisfies the following constrained minimization problem:

$$\min_{\mathbf{W}_2} \| \begin{pmatrix} k \\ 2 \end{pmatrix} \boldsymbol{H}\mathbf{W}_2 - \mathbf{W}_k \|_F^2, 0 \leq \mathbf{W}_2 \leq 1 \ . \tag{2.34}$$

This method is closely related to the clique expansion. Denote by $\mathbf{W}_2^e$ the vector of approximating graph edge weights, then we can derive the following equation from the solution of Equation (2.29):

$$\mu(n,k)\boldsymbol{H}\mathbf{W}_2^e = \boldsymbol{H}\boldsymbol{H}^T\mathbf{W}_k \ . \tag{2.35}$$

Neglecting the constants in Equation (2.33) and (2.35), which differ only in the right hand side by a pre-multiplication by the matrix $\boldsymbol{H}\boldsymbol{H}^T$. This is a symmetric matrix, the effect of multiplying this matrix by $\mathbf{W}_k$ is equivalent to a convolution of the hyperedge weights by a quadratically decreasing kernel [61]. Thus $\boldsymbol{H}\boldsymbol{H}^T\mathbf{W}_k$ is a low passed version of $\mathbf{W}_k$. This implies that the **clique expansion** solves the same approximation problem as **clique averaging**. However, instead of operating on the original hypergraph it operates on a low passed version of it.

## 2.4 Conclusion

We have reviewed two dominant directions of research in filter-based feature selection. We have analyzed the deficiencies of the existing feature selection methods and pointed out our possible solutions for overcoming these shortcomings. This chapter can be summarized as follows.

There is a substantial body of research on MI-based feature selection methods. As we discuss above, there are four limitations for the existing MI-based feature selection methods. Firstly, these methods do not provide a direct measure to judge whether to add additional features or not, so the number of selected features need to be specified in advance. In real applications, it is hard to estimate the number of useful features before the feature selection process. The second weakness is that they assume that each individual relevant feature should be dependent with the target class. This means that if a single feature is considered to be relevant it should be correlated with the target class, otherwise the feature is irrelevant [28]. So only a small set of relevant features is selected, and larger feature combinations are not considered. The third weakness is that most of these methods focus on ranking features based on an information criterion and select the best $m$ features in a greedy way. Here, commencing from an empty feature pool, features are added into the pool one by one until the user-defined number is reached. However, several authors find that the optimal feature combinations do not give the best classification performance [67] [66]. Finally, most of the methods simply consider pairwise feature dependencies, and do not check for third or higher order dependencies between the candidate features and the existing features. For example, there are four features $f_1$,$f_2$,$f_3$,$f_4$, the existing selected feature subset is $\{f_1,f_4\}$. Assume $I(f_2, C) = I(f_3, C)$, $I(f_2, f_1|C) = I(f_3, f_1|C)$, $I(f_2, f_4|C) = I(f_3, f_4|C)$, $I(f_1, f_4, f_2) \gg I(f_1, f_4, f_3)$ and $I(f_1, f_4, f_2) \gg I(f_1, f_2) + I(f_4, f_2)$. This indicates that $f_2$ has strong affinity with the joint subset $\{f_1,f_4\}$, although it has smaller individual affinity to each of them. So in this situation, $f_2$ may be discarded, and $f_3$ is selected, although the combination $\{f_1,f_4,f_2\}$

can produce a better cluster than $\{f_1, f_4, f_3\}$ [36].

The research literature on graph-based feature selection methods. In feature selection, the attractive feature of graph representations is that they provide a universal and flexible framework that reflects the underlying manifold structure and the relationships between feature vectors. The idea underpinning graph-based feature selection methods is to select the features which best preserve the data similarity or a manifold structure derived from the entire feature set. However, there are two limitations to the above graph-based spectral feature selection methods. Firstly, they evaluate features individually, and hence cannot handle redundant features. Redundant features increase the dimensionality unnecessarily, and worsen learning performance when faced with a shortage of data. It is also shown empirically that removing redundant features can result in significant performance improvement. The second weakness is that in many situations the graph representation for relational patterns can lead to substantial loss of information. This is because in real-world problems objects and their features tend to exhibit multiple relationships rather than simple pairwise ones.

Research on hypergraph based learning algorithms is generally confined to tensor factorization, which is a higher order extension of its pairwise counterpart. When hypergraphs are used for representing higher order structured data in structural pattern recognition, they are often approximated by a graph representation. Trivial graph approximations may give rise to certain information loss and result in ambiguities in distinguishing different relational orders. To address these shortcomings in the existing hypergraph based methods, we will employ more effective matrix for hypergraph representation.

Above all, the work in this thesis addresses the shortcomings in the research literature. We will compare our proposed methods with the state of the art methods and discuss in detail our contributions to the research literature in the subsequent chapters.

# Chapter 3

# Graph based Information-theoretic Feature Selection

In many data analysis tasks, one is often confronted with very high dimensional data. As shown in Chapter 2, the feature selection problem is essentially a combinatorial optimization problem which is computationally expensive. To overcome this problem it is frequently assumed either that features independently influence the class variable or do so only involving pairwise feature interaction. However, several authors find that the optimal feature combinations do not give the best classification performance [66][67]. The reason for this is that although individual features may have limited relevance to a particular class, when taken in combination with other features it can be strongly relevant to the class. To tackle this problem, in this chapter, we propose a graph based information-theoretic approach to feature selection. There are three novel ingredients. First, by incorporating mutual information (MI) for pairwise feature similarity measure, we establish a novel feature graph framework which is used for characterizing the relevance between the pair of features. Secondly, we locate the relevant feature subset (RFS) from the feature graph by maximizing features' average pairwise relevance. The RFS is expected to has little redundancy and very strong discriminating power. This strategy reduces the optimal search space from the original feature set to the relatively smaller relevant feature subset, and thus enable an efficient computation. Finally, based on RFS, we evaluate the impor-

tance of unselected features by using a new information theoretic criterion referred to as the multidimensional interaction information (MII). The advantage of MII is that it can go beyond pairwise feature interaction and consider third or higher order feature interactions. As a result, we can evaluate features jointly, and thus avoid the redundancies arising in individual feature combinations. Experimental results demonstrate the effectiveness of our feature selection method on a number of standard data-sets.

### Contribution

In summary, there are three main contributions in this chapter. First we develop a graph representation based on the attributes of feature vectors, i.e. a feature graph. Each edge in the graph has a weight corresponding to the mutual information (MI) between features connected by that edge. With this representation, the informativeness latent in the features can be more effectively modeled. Second we use a new information theoretic criterion referred to as MII to measure the significance of different feature combinations. The advantage of MII is that it is sensitive to the relations between feature combinations. As a result it can be used to seek third or even higher order dependencies between the relevant features. Hence, we can evaluate features jointly rather than individually. Thus we are able to handle feature redundancy. Third we extract the relevant feature subset (RFS) from the initial features as a preprocessing step for ranking features. In doing so we can limit the search space for higher order interactions.

### Chapter outline

The outline of this chapter is as follows. Section 3.1 commences by reviewing the fundamental knowledge related to MII and describes how to apply this new criterion to discriminating feature selection. Section 3.2 describes how to reduce the search space by locating the relevant feature subset (RFS) as a preprocessing step and how to use MII criterion for further feature selection. In Section 3.3, a detailed description of the feature evaluation

indices is given. Experimental results on a number of standard data-sets are presented in Section 3.4. Finally, conclusions and future work are presented in Section 3.5.

## 3.1 Feature Selection Criteria Based on Mutual Information

This section describes how to develop a new feature selection criteria based on the concepts about Mutual Information (MI). Instead of finding some feature low-order interactions [6], our proposed new criterion can go beyond pairwise feature interaction and consider third or higher order feature interactions. We commence by reviewing the relevant information theory. We then derive multidimensional interaction information (MII) for feature selection and describe how to estimate it in practical computation.

### 3.1.1 Definition of Mutual Information

In accordance with Shannon's information theory [13], the uncertainty of a random variable $C$ can be measured by the entropy $H(C)$. For two variables $F$ and $C$, the conditional entropy $H(C|F)$ measures the remaining uncertainty about $C$ when $F$ is known. The mutual information (MI) represented by $I(F;C)$ quantifies the information gain about $C$ provided by variable $F$. The relationship between $H(C)$, $H(C|F)$ and $I(F;C)$ can be given by

$$I(F;C) = H(C) - H(C|F)\ . \tag{3.1}$$

For training a classifier, we prefer features which can minimize the uncertainty on the output class set $C$. If $I(F;C)$ is large, this implies that feature vector $F$ and output class set $C$ are closely related. When $F$ and $C$ are independent, the MI of $F$ and $C$ goes to zero, and this means that the feature $F$ is irrelevant to class $C$. As defined by Shannon, the initial uncertainty in the output class $C$ is expressed as:

$$H(C) = -\sum_{c \in C} P(c) \log P(c) \ . \tag{3.2}$$

where $P(c)$ is the prior probability over the set of class $C$. The remaining uncertainty in the class set $C$ if the feature vector $F$ is known is defined by the conditional entropy $H(C|F)$

$$H(C|F) = -\int_f p(f) \{ \sum_{c \in C} p(c|f) \log p(c|f) \} df \ . \tag{3.3}$$

where $p(c|f)$ denotes the posterior probability for class $c$ given the input feature vector $f$. After observing the feature vector $f$, the amount of additional information gain is given by the mutual information (MI)

$$I(F;C) = H(C) - H(C|F) = \sum_{c \in C} \int_f p(c,f) \log \frac{p(c,f)}{p(c)p(f)} df \ . \tag{3.4}$$

### 3.1.2 Conditional Mutual Information

Assume that $S$ is the set of existing selected features, $\vec{F}$ is the set of candidate features, $S \cap \vec{F} = \emptyset$, and $C$ is the output class set. The next feature in $\vec{F}$ to be selected is the one that maximizes $I(C; f_i|S)$, i.e. the conditional mutual information (CMI) which can be represented as

$$I(C; f_i|S) = H(C|S) - H(C|f_i, S) \ . \tag{3.5}$$

where $C$ is the output class set, $S$ is the selected feature subset, $\vec{F}$ is the candidate feature subset, and $f_i \in \vec{F}$. From information theory, the conditional mutual information is the expected value of the mutual information between the candidate feature $f_i$ and class set $C$ when the existing selected feature set $S$ is known. It can be also rewritten as

$$I(C; f_i|S) = \sum_S \sum_{c \in C} \int_{f_i \in \vec{F}} P(f_i, S, c) \log \frac{P(S)P(f_i, S, c)}{P(f_i, S)P(S, c)} \ . \tag{3.6}$$

### 3.1.3 Multidimensional Interaction Information for Feature Selection

The conditioning on a third random variable may either increase or decrease the original mutual information. That is, the difference $I(X;Y|Z) - I(X;Y)$, referred to as the interaction information and represented by $I(X;Y;Z)$, can measure the difference between the original mutual information $I(X;Y)$ when a third random variable is taken into account or not. The difference may be positive, negative, or zero, but it is always true that $I(X;Y|Z) \geqslant 0$ [72].

Given the existing selected feature set $S$, the interaction information between the output class set and the next candidate feature $f_i$ can be defined as

$$I(C;f_i;S) = I(C;f_i|S) - I(C;f_i)\ . \tag{3.7}$$

From Equation (3.7), the interaction information measures the influence of the existing selected feature set $S$ on the amount of information shared between the candidate feature $f_i$ and the output class set $C$, i.e. $I(C,f_i)$. A zero value of $I(C;f_i;S)$ means that the information contained in the observation $f_i$ is not useful for determining the output class set $C$, even when combined with the existing selected feature set $S$. A positive value of $I(C;f_i;S)$ means that the observation $f_i$ is independent of the output class set $C$, so $I(C,f_i)$ will be zero. However, once $f_i$ is combined with the existing selected feature set $S$, then the observation $f_i$ immediately becomes relevant to the output class set $C$. As a result $I(C;f_i|S)$ will be positive. As a result the positive interaction information implies synergy between the existing selected feature set $S$ and new feature $f_i$, meaning that they yield more information together than what could be expected from they individual interactions with the label. Thus, it is capable of solving $XOR$-gate type classification problems. A negative value of $I(C;f_i;S)$ indicates redundancy between the existing selected feature set $S$ and new feature $f_i$, meaning that $S$ can account for or explain the correlation between $I(C;f_i)$. As a result the shared information between $I(C;f_i)$ is decreased due

to the additional knowledge of the existing selected feature set $S$. Hence, negative interactions offer opportunity for eliminating redundant feature, even if the feature is relevant on its own.

According to the above definition, we propose the following multidimensional interaction information for feature selection. Assume that $S$ is the set of existing selected feature sets, $\vec{F}$ is the set of candidate features, $S \cap \vec{F} = \emptyset$, and $C$ is the output class set. The objective of selecting the next feature is to maximize $I(C; f_i|S)$, defined by introducing the multidimensional interaction information:

$$I(C; f_i|S) = I(C; f_i) + I(\{f_i, S, C\}) \, . \tag{3.8}$$

where

$$I(\{f_i, S, C\}) = I(f_i, s_1, \ldots, s_{m-1}; C) = \sum_{s_1,\ldots,s_{m-1}} \sum_{c \in C} P(f_i, s_1, \ldots, s_{m-1}; c) \times \log \frac{P(f_i, s_1, \ldots, s_{m-1}; c)}{P(f_i, s_1, \ldots, s_{m-1}) P(c)} \, . \tag{3.9}$$

Consider the joint distribution $P(f_i, S) = P(f_i, s_1, \ldots, s_{m-1})$. By the chain rule of probability, we expand $P(f_i, S)$, $P(f_i, S; C)$ as

$$P(f_i, S) = P(s_1)P(s_2|s_1) \times P(s_4|s_3, s_2, s_1) \cdots P(x_i|s_1, s_2 \ldots s_{m-1}) \, , \tag{3.10}$$

$$\begin{aligned} P(f_i, S; C) &= P(C)p(s_1|C)P(s_2|s_1, C)P(s_3|s_1, s_2, C) \\ &\quad \times P(s_4|s_1, s_2, s_3, C) \cdots P(f_i|s_1, \ldots, s_{m-1}, C) \, . \end{aligned} \tag{3.11}$$

There are two key properties of our proposed definition in Equation (3.8). The first is that the interaction information term $I(\{f_i, S, C\})$ which can be zero, negative and positive. It can deal with a variety of cluster classification problems including the $XOR$-gate when the value is positive. When it taken on a negative value, it can help to eliminate redundant features and thus select optimal feature sets. The second benefit is its multidimensional

form, compared to most existing MI methods which only check for pairwise feature interactions. Our definition can be used to check for third and higher order dependencies among features.

However, in practice and as noted in Chapter 2, locating a feature subset that maximizes $I(\{f_i, S, C\})$ presents two problems: 1) it requires an exhaustive "combinatorial" search over the feature space, and 2) it demands large training sample sizes to estimate the higher order joint probability distribution in $I(\{f_i, S, C\})$ with a high dimensional kernel [49]. Bearing these obstacles in mind, most of the existing related papers approximate $I(\{f_i, S, C\})$ based on the assumption of lower-order dependencies between features. For example, the first-order class dependence assumption includes only first-order interactions. That is it assumes that each feature independently influences the class variable, so as to select the $m$-th feature, $f_i$, $P(f_i|s_1 \dots s_{m-1}, C) = P(f_i|C)$. A second-order feature dependence assumption is proposed by Guo and Nixon [7] to approximate $I(\{f_i, S, C\})$, and this is arguably the most simple yet effective evaluation criterion for selecting features. The approximation is given as

$$I(\{f_i, S, C\}) \approx \widehat{I}(\{f_i, S, C\}) = \sum_i I(f_i; C) - \sum_i \sum_{s_j \in S} I(f_i; s_j) + \sum_i \sum_{s_j \in S} I(f_i; s_j | C) \; . \tag{3.12}$$

By using $\widehat{I}(\{f_i, S, C\})$ instead of $I(\{f_i, S, C\})$, it is possible to locate a subset of informative features by implementing a greedy "pick-one-feature-at-a-time" selection procedure. Given $d$ features, out of which $m$ are to be selected $(m < d)$, this involves two steps: 1) select the first feature $f'_{max}$ that maximizes $I(f'; C)$, and 2) select the $m-1$ subsequent features that maximize the criterion in Equation (3.12) , i.e., select the second feature $f''_{max}$ that maximizes $I(f''; C) - I(f''; f'_{max}) + I(f''; f'_{max}|C)$, select the third feature $f'''_{max}$ that maximizes $I(f'''; C) - I(f'''; f'_{max}) - I(f'''; f''_{max}) + I(f'''; f'_{max}|C) + I(f'''; f''_{max}|C)$ and so on.

Although an MII based on the second-order feature dependence assumption can se-

lect features that maximize class-separability and simultaneously minimize dependencies between feature pairs, there is no reason to assume that the final optimal feature subset is formed by pairwise interactions between features. In fact, it neglects the fact that third or higher order dependencies can be lead to an optimal feature subset.

The primary reason for using the approximation $\widehat{I}(\{f_i, S, C\})$ for feature selection instead of directly using multidimensional interaction information $I(\{f_i, S, C\})$ is that $I(\{f_i, S, C\})$ requires an exhaustive "combinatorial" search over the feature space.

To tackle the above problems, we establish a novel graph-based information-theoretic framework for characterizing the feature correlations, where we employ mutual information (MI) for measuring features relevance. We commence by extracting the relevant feature subset (RFS) from the initial features, as a pre-processing step for ranking features. This strategy reduces the optimal search space from the original feature set to the relatively smaller relevant feature subset, and thus enable an efficient computation. Therefore, we do not need to use the approximation $\widehat{I}(F; C)$. Instead, we can use the multidimensional interaction information $I(F; C)$ criterion directly for feature selection.

### 3.1.4 Estimation of MII

In the above definition of the MII measure, the class label takes on discrete values while the input feature vectors are usually continuous random variables. In this case, one solution to the high dimension of the feature vectors is to incorporate data discretization as a preprocessing step. For some applications where it is unclear how to properly discretize the continuous data, an alternative solution is to use the density estimation method (e.g., Parzen windows) to estimate the class conditional probability density function for the feature vectors. Given $N$ $d$-dimension samples $\mathbf{X} = \{x_1, x_2, \ldots, x_N\}$, the probability density estimation $p(x)$ of a $d$-dimension continuous random vectors $x$ is given by

$$p(x) = \frac{1}{N}\phi(\frac{x - x_i}{h}), \tag{3.13}$$

where $x_i$ is $i$-th $d$-dimensional sample, $N$ is the number of samples, $\phi(\frac{x-x_i}{h})$ is the window function and $h$ is the window width. Here, we use a Gaussian as the window function, so

$$\phi(\frac{x-x_i}{h}) = \frac{1}{(2\pi)^{\frac{l}{2}} h^l |\Sigma|^{\frac{1}{2}}} \exp(\frac{(x-x_i^T)\Sigma^{-1}(x-x_i)}{-2h^2}) \ . \tag{3.14}$$

where $\Sigma$ is the covariance matrix for the feature vector and $l$ is the length of vector $x$. When $l = 1$, $p(x)$ estimates the marginal density and when $l = 3$, $p(x)$ estimates the joint density of the feature vectors. We use the Parzen window method to estimate the conditional entropy in Equation (3.3). From Bayesian a posterior rule, the class probability $p(c|x)$ can be written as

$$p(c|x) = \frac{p(x|c)p(c)}{p(x)} \ . \tag{3.15}$$

If there are $C$ classes, then, we obtain the class conditional probability density $p(x|c)$ of each class using the Parzen window estimation

$$p(x|c) = \frac{1}{N_c} \sum_{i=1}^{N_c} \phi(\frac{x-x_i}{h}) \ . \tag{3.16}$$

where $c = 1, 2, \cdots, C$ and $N_c$ is the number of the training examples belonging to class $c$. Because the conditional probability normalized, i.e., $\sum_{c=1}^{C} p(c|x) = 1$, the conditional probability $p(c|x)$ is

$$p(c|x) = \frac{p(c|x)}{\sum_{c=1}^{C} p(c|x)} = \frac{p(c)p(x|c)}{\sum_{c=1}^{C} p(t)p(x|c)} \ . \tag{3.17}$$

Using Equation (3.16), the estimate of the a posterior probability becomes

$$p(c|x) = \frac{\sum_{i=1}^{N_c} \phi(\frac{x-x_i}{h_c})}{\sum_{c=1}^{C} \sum_{i=1}^{N_c} \phi(\frac{x-x_i}{h_c})} \ . \tag{3.18}$$

where $h_c$ is the class specific window width parameters.

## 3.2 The Proposed Feature Selection Scheme

To avoid the exhaustive "combinatorial" search over the feature space in using MII, our proposed method works in two phases. In the first phase, we extract the relevant feature

subset (RFS) from the initial features as a preprocessing step and in doing so we can limit the search space for higher order interactions. We commence by constructing a feature relevance matrix $\boldsymbol{R} = (R_{ij})$ to characterize the relevance of features. Then, we employ a coherence function on $\boldsymbol{R}$ for the purpose of identifying RFS. Second phase describes how to rank the features based on MII criteria. Next we discuss these phases in detail.

### 3.2.1 Relevant Feature Subset Extraction

According to the properties of interaction information described in 3.1.3, straightforwardly identifying a feature subset that maximizes $I(\{f_i, S, C\})$ in (3.9) requires an exhaustive "combinatorial" search over the feature space. Furthermore, a high dimensional kernel should be computed through estimate the higher order joint probability distribution in $I(\{f_i, S, C\})$ [49]. To address these obstacles, we aim to reduce the search space from the initial feature set to a smaller relevant feature subset (RFS), which is expected to have little redundancy and very strong discriminating power.

From Section 3.1, we can see that mutual information quantifies the information which is shared by two variables $X$ and $Y$. When $I(X;Y)$ is large, this implies that variable $X$ and variable $Y$ are closely related. Otherwise, when $I(X;Y)$ is equal to 0, this means that two variables are totally unrelated. Therefore, in our method, the relevance of pairs of feature vectors is captured using mutual information. For a feature pair $\{f_i, f_j\}$, the relevance degree between the feature vectors can be defined as

$$R_{i,j} = \frac{1}{2} I(f_i; C) + \frac{1}{2} I(f_j; C) - [I(f_i; f_j) - I(f_i; f_j|C)] \,. \tag{3.19}$$

The above degree relevance definition consists of three terms. The first term $I(f_i; C)$ referred to as *relevancy* indicates individual feature's relevance with the class set $C$. The third term of the form $I(f_i; f_j)$ is used to measure the redundancy between features. The fourth term $I(f_i; f_j|C)$ measures the influence of the features combination on the class

set $C$. We refer to this as the *class-conditional redundancy*. Therefore, a large value of $R_{(f_i,f_j)}$ means that both $I(f_i;C)$ and $I(f_i;f_j|C)$ are large (indicating features $\{f_i, f_j\}$ are relevant with respect to the class set $C$) and $I(f_i;f_j)$ is small (indicating features $\{f_i, f_j\}$ are less redundant).

Supposed the cardinality of the initial feature set is $d$. Given a $d$-dimensional indicator vector $\boldsymbol{a}$ with $a_i$ representing the $i$-th element, we employ a coherence function as the objective function for the purpose of identifying the most homogenous subset of the initial feature set

$$\max f(\boldsymbol{a}) = \sum_{i=1}^{d}\sum_{j=1}^{d} a_i a_j R_{i,j} \; . \tag{3.20}$$

subject to $\boldsymbol{a} \in \triangle$, where the multidimensional solution vector $\boldsymbol{a}$ fall on the simplex $\triangle = \{\boldsymbol{a} \in \mathbb{R}^d : \boldsymbol{a} \geq 0 \text{ and } \sum_{i=1}^{d} a_i = 1\}$ and $R_{ii} = 0$, i.e., all diagonal entries of $\boldsymbol{R}$ are set to zero. Our idea is motivated by the the graph-based clustering method which group the most dominant vertices into cluster. On the other hand, in our work, the feature subset $\{f_i|1 \leq i \leq d, a_i > 0\}$ is the most coherent subset of the initial feature set, with maximum internal homogeneity of the feature relevance (3.19). According to the value of $\boldsymbol{a}$, all features $F$ fall into two disjoint subsets, $S_1(\boldsymbol{a}) = \{f_i|a_i = 0\}$ and $S_2(\boldsymbol{a}) = \{f_i|a_i > 0\}$. We refer to the set of nonzero variables $S_2(\boldsymbol{a})$ as the relevant feature subset (RFS), because the objective function (3.20) selects RFS by maximizing features' average pairwise relevance.

In fact, the main property of RFS is that the overall relevance among the internal features is greater than that between the external features and the internal features. From graph theory, RFS turns out to be equivalent to maximal cliques [42]. The definition of RFS simultaneously emphasizes internal homogeneity together with external inhomogeneity. Thus it is can be used as a general definition of a "cluster". To provide an example, assume there are $N$ training samples, each having $5$ feature vectors. In order to capture the RFS from these 5 features (represented as $F_1, \ldots, F_5$), we construct a graph

$G = (V, E)$ with node-set $V$, edge-set $E \subseteq V \times V$ and edge weight matrix $\mathbf{W}$ whose elements are in the interval $[0, 1]$. Each vertex represents a feature and the edge between two features represents their pairwise relationship. The weight on the edge reflects the degree of relevance between two features. Therefore, we represent the graph $G$ with the corresponding edge-weight or weighted relevance matrix. Let $S \subseteq V$ be a non-empty subset of vertices and $i \in S$. The average weighted degree of $i$ w.r.t $S$ is defined as

$$awdeg_S(i) = \frac{1}{|S|} \sum_{j \in S} R_{i,j} \ . \tag{3.21}$$

if $j \notin S$, we have the following definition: $\phi_S(i, j) = R_{i,j} - awdeg_S(i)$ which measures the similarity between nodes $j$ and $i$, with respect to the average similarity between node $i$ and its neighbors $S$. The weight of $i$ w.r.t. (S) is

$$W_S(i) = \begin{cases} 1, & \text{if } |S| = 1 \\ \sum_{j \in S \setminus \{i\}} \phi_{S \setminus \{i\}}(j, i) W_{S \setminus \{i\}}(j), & \text{otherwise.} \end{cases} \tag{3.22}$$

Moreover, the total weight of $S$ is defined to be $W(S) = \sum_{i \in S} W_S(i)$. Inspired from the recent work on graph partition [43], for the constructed feature graph, a feature subset $S$ is said to be RFS if: 1) $W_S(i) > 0$, for all $i \in S$, 2) $W_{S \cup \{i\}}(i) < 0$, for all $i \notin S$. In our example, in Fig. 3.1, features $\{F_1, F_2, F_3\}$ form the RFS, since the edge weights "internal" to that set (0.6, 0.7 and 0.9) are larger than the sum of those between the internal and external features (which is between 0.05 and 0.25).

The objective function (3.20) is typical quadratic program, and here we apply discrete-time first-order replicator equation [35] to approximating the solution for the RFS.

$$a_i^{new} = \frac{a_i \sum_{j=1}^{d} a_j R_{i,j}}{\sum_{i=1}^{d} \sum_{j=1}^{d} a_i a_j R_{i,j}} \ . \tag{3.23}$$

where $a_i^{new}$ corresponded to the $i$-th feature vector after the update process. The complexity of finding the RFS is $O(t|E|)$, where $|E|$ is the number of edges of the feature graph constructed above and $t$ is the average number of iteration needed to converge.

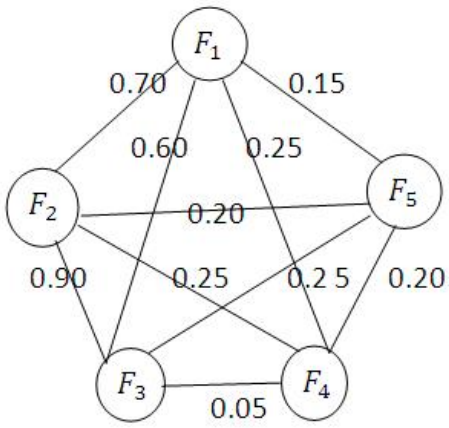



Figure 3.1: The subset of features $\{F_1, F_2, F_3\}$ is RFS

### 3.2.2 Feature Ranking using MII criterion

The features inside RFS have both little redundancy and very strong discriminating power. Based on the RFS, we can evaluate the importance of features contained in the unselected feature set $S_1(\boldsymbol{a}) = \{f_i | a_i = 0\}$ using MII criterion. Consequently, we can obtain a complete feature ranking list.

The multidimensional interaction information between feature vector $F = \{f_1, \ldots, f_m\}$ and class variable $C$ is:

$$I(F;C) = I(f_1, \ldots, f_m; C) = \sum_{f_1,\ldots,f_m} \sum_{c \in C} P(f_1, \ldots, f_m; c) \times \log \frac{P(f_1, \ldots, f_m; c)}{P(f_1, \ldots, f_m)P(c)} \,. \tag{3.24}$$

For the unselected features in $S_1(\boldsymbol{a}) = \{f_i | a_i = 0\}$, we use MII to rank the features and record the incremental gain (IG) score for each feature. Assume the existing selected relevant feature subset is $S_{RFS}$, for the unselected feature $j$, we define the IG score for the feature as

$$IG(j) = I(f_j; C) - [I(S_{RFS}; f_j) - I(S_{RFS}; C | f_j)] \,. \tag{3.25}$$

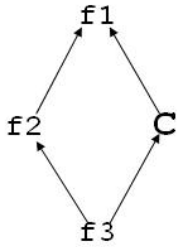



Figure 3.2: Illustration the IG score

The above definition of the IG score consists of three terms. The first term $I(f_j; C)$ is referred to as relevance. The second term of the form $I(S_{RFS}; f_j)$ is referred to as redundancy. The third term $I(S_{RFS}; C|f_j)$ is referred to as conditional redundancy. We sort the features in descending order according to their IG scores. Equation (3.25) has an intuitive geometric explanation as shown in Figure 3.2. This example contains 4 variables in a diamond configuration and $C$ is our target. We generate a very simple dependence in which $f_1$, $f_2$, $f_3$ and $C$ are normally distributed variables. Our target $C$, is a noisy observation of $f_1$ and $f_2$ is correlated with $f_1$. $f_3$ is correlated with the target $C$ and $f_2$. We rank the features $f_1$, $f_2$, $f_3$ using the MII criterion and record the incremental gain (IG) score for each feature in Table 3.1.

| Rank | IG Score | Relevance | Redundancy | Conditional redundancy |
|---|---|---|---|---|
| $f_1$ | 0.1794 | 0.1794 | 0 | 0 |
| $f_3$ | 0.0697 | 0.1617 | 0.1416 | 0.0495 |
| $f_2$ | 0.0006 | 0.0879 | 0.2662 | 0.1789 |

Table 3.1: The IG score for each feature

As Table 3.1 shows, since $f_3$ depends on $f_2$ as well, $f_1$ receives a higher score. Clearly, due to the common dependency, $f_2$ bears some mutual information on the target $C$. Note, however, how this is outweighed by the interaction term (redundancy - conditional redundancy): In fact, once we know $f_1$, $f_2$ cannot provide any additional information about the

target $C$.

In order to evaluate the effectiveness in finding interacting features by using MII, we ran experiments on three synthetic data sets with known feature interactions. The first data set is Corral [26], which contains six boolean features $A_0$, $A_1$, $B_0$, $B_1$, irrelevant feature $I$ and redundant feature $R$. The target concept $C$ is defined by $C = (A_0 \bigwedge A_1) \bigvee (B_0 \bigwedge B_1)$ and feature $A_0$, $A_1$, $B_0$, $B_1$ are independent of each other. The irrelevant feature $I$ is uniformly random and the redundant feature $R$ matches the class label $75\%$ of the time (for specific instances). This is an example of data sets in which if a redundant feature like $R$ is removed, a more accurate result will be obtained. The other two data sets are taken from MONK's problem [65]. They have six features. Their target concepts are defined by three features: (1) MONK1, $(A_1 = A_2)$ or $(A_5 = 1)$; Here $A_1$ and $A_2$ are two interacting features. Consider individually, the correlation between $A_1$ and the target class $C$ (similarly for $A_2$ and $C$) is zero, measured by mutual information. Hence, $A_1$ or $A_2$ is irrelevant when each is individually evaluated. However, if we combine $A_1$ and $A_2$, they are strongly relevant in defining the target concept. (2) MONK3, $(A_5 = 3$ and $A_4 = 1)$ or $(A_5 \neq 4$ and $A_2 \neq 3)$ ($5\%$ class noise added to the training data ). We apply three alternative MI-based criterion methods to the synthetic data sets for comparison. These methods are the MRMR algorithm [27], the MIFS algorithm [55] and the JMI algorithm [29].

Table 3.2 shows the comparative feature ranking results of MII criterion with other three alternative MI-based feature selection algorithms. For Corral, because the redundant feature $R$ is highly correlated with the class label, MRMR, MIFS and JMI pick it as the best one. Our proposed method MII, on the other hand, discover that the feature $R$ is hurting performance after the evaluation of higher order feature interactions and thus avoid selecting it. Features $A_0$, $A_1$ and $B_0$, $B_1$ interact with each other to determine the class label of an instance. For the two MONKs data sets, only two features out the three relevant ones are selected by MRMR and MIFS. Both of them missed $A_2$ for MONK1

| | MII | MRMR | MIFS | JMI |
|---|---|---|---|---|
| Corral | $A_0, A_1, B_0,$ $B_1, R, I$ | $R, A_0, A_1,$ $B_0, B_1, I$ | $R, A_0, A_1,$ $B_0, B_1, I$ | $R, A_0, A_1,$ $B_0, B_1, I$ |
| Monk1 | $A_5, A_1, A_2,$ $A_4, A_6, A_3$ | $A_5, A_1, A_4,$ $A_3, A_6, A_2$ | $A_5, A_1, A_3,$ $A_4, A_6, A_2$ | $A_5, A_1, A_2,$ $A_4, A_6, A_3$ |
| Monk3 | $A_2, A_5, A_4,$ $A_1, A_6, A_3$ | $A_2, A_5, A_6,$ $A_1, A_3, A_4$ | $A_2, A_5, A_6,$ $A_3, A_1, A_4$ | $A_2, A_5, A_4,$ $A_1, A_3, A_6$ |

Table 3.2: Feature ranked by different algorithms on synthetic data

and $A_4$ for MONK3 respectively. As seen in Table 3.2, MII and JMI perform similarly for the MONKs data sets. However, as an exhaustive search algorithm, JMI is impractical because finding moderately high-order interactions can be too expensive.

The sequence of steps shown in Algorithm 1 illustrates our method in detail.

**Algorithm 1:** A graph based information-theoretic framework for feature selection

**Input**: Dataset $\mathbf{X}$ with all features $d$

**Output**: The selected relevant feature subset (RFS) according to the non-zero elements of $\boldsymbol{a}$ and unselected features are ranked by IG score

1) Compute the feature mutual relevance using Equation (3.19) ;

2) Extract the relevant feature subset (RFS) by using Equation (3.20) and Equation (3.23) ;

3) Using MII criterion (see Equation (3.24)) to rank the unselected features based on RFS and record the IG score for each feature according to Equation (3.25);

4) rank features according to their IG scores.

## 3.3 Feature Evaluation Indices

Our proposed feature selection method (referred to as the RFS+MII method) (which utilizes the multidimensional interaction information criterion and relevant feature subset for feature selection) involves extracting the relevant feature subset (RFS) form the initial features as a pre-processing step and using MII for further feature selection. In order to examine the performance of our proposed method RFS+MII, we need to assess the quality of the relevant feature subset obtained together with its useful information content. In view of this, we would like to measure the performance of our proposed algorithm using three different indices, namely, (1) **Relevant Feature Subset Evaluation**, (2) **Classification Accuracy** and (3) **Redundancy Rate**. Assume $S$ is the set of selected features, the redundancy rate can be defined as follow:

$$RED(S) = \frac{1}{m(m-1)} \sum_{f_i, f_j \in S, i>j} \rho_{i,j} \ . \tag{3.26}$$

where $\rho_{i,j}$ returns the Pearson correlation between two features $f_i$ and $f_j$. The measurement assesses the averaged correlation among all feature pairs, and a large value indicates that many selected features are strongly correlated and thus redundancy is expected to exist in $S$.

## 3.4 Experiments and Comparisons

The data sets used to test the performance of our proposed algorithm are benchmark data sets from the UCI Machine Learning Repository. Table. 3.3 summarizes the extents and properties of the six data-sets.

| Data-set | Training data | Testing data | Features | Classes |
|---|---|---|---|---|
| Wine | 100 | 78 | 13 | 3 |
| Pendigits | 7494 | 3498 | 16 | 10 |
| Vowel | 528 | 462 | 10 | 11 |
| Letter | 15000 | 5000 | 16 | 26 |
| Satimage | 4435 | 2000 | 36 | 6 |
| Dna | 2000 | 1186 | 180 | 3 |

Table 3.3: Summary of UCI benchmark data sets

### 3.4.1 Relevant Feature Subset Evaluation

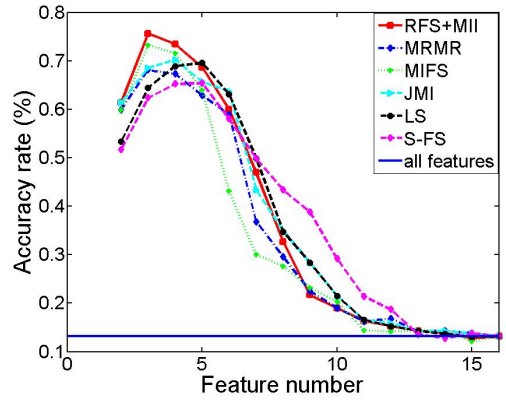
We compare the classification result from the features captured by the relevant feature subset (RFS) with those obtained using both alternative MI-based criterion methods and graph-based methods. These methods are the MRMR algorithm [27], the MIFS algorithm [55], the JMI algorithm [29], the Laplacian score (LS) [73] and the subset-level based Fisher score (S-FS) [24]. We use 5-fold cross-validation for the SVM classifier on the feature subsets obtained by the feature selection algorithms to verify their classification performance. Here we use the linear SVM with LIBSVM [14].

We summarize the classification accuracy rate of different methods in Table. 3.4. In the last row, the classification accuracy for the features from the relevant feature subset which referred as RFS and the automatically determined size of RFS are reported. Suppose that the determined size of RFS is $k$. To make a fair comparison, for each alternative method, we measure the classification accuracy for $k-1$, $k$ and $k+1$ features, and take the best result as the baseline performance. As shown by the results in Table. 3.4, our extracted RFS consistently outperforms other feature subsets obtained by the alternative methods on all six multi class data sets. The results verify that our proposed method is effective to locate the relevant feature subset. There are two reasons for this improvement in

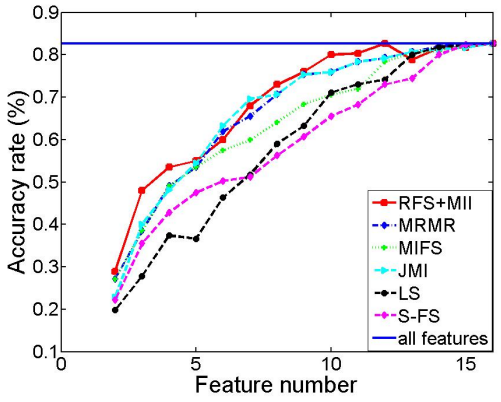
| Dataset | Pendigits | Letter | Wine | Vowel | Satimage | Dna |
|---|---|---|---|---|---|---|
| MRMR | 68.1% | 78.3% | 98.3% | 49.6% | 85% | 92.7% |
| MIFS | 73.2% | 71.9% | 97.2% | 49.6% | 85.2% | 90.6% |
| JMI | 70.2% | 78.3% | 97.8% | 51.3% | 85% | 94.1% |
| LS | 69% | 73% | 97.8% | 49.6% | 84.9% | 90.6% |
| S-FS | 65.2% | 68% | 96.1% | 52.2% | 84.9% | 92.4% |
| RFS | $\mathbf{75.6\%(3)}$ | $\mathbf{80\%(10)}$ | $\mathbf{98.9\%(4)}$ | $\mathbf{53.6\%(3)}$ | $\mathbf{86\%(15)}$ | $\mathbf{95.6\%(16)}$ |

Table 3.4: Performance comparison of accuracy rate around the size of features in RFS selected by different methods on the multi class data sets

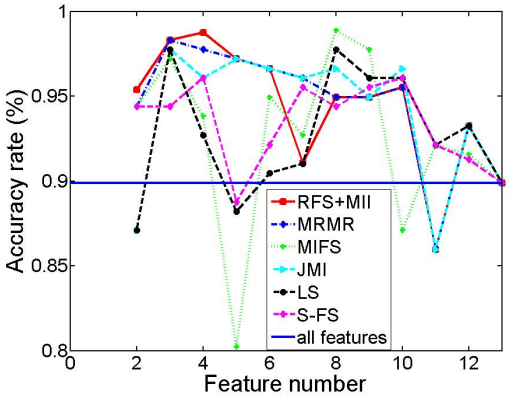
performance. First, the mutual information is applied to measure the features relevance, and this can capture the effects of pairwise dependencies between the features and the class. Second, the extraction of the relevant feature subset simultaneously considers the information-contribution for each feature together with the correlation between features. Thus structural information latent in the data can be effectively identified. As a result the optimal feature combinations can be located so as to group the greatest number of relevant features into homogenous subset. On the other hand, other graph-based methods (i.e. LS and S-FS) fail to locate the most discriminative features. This may be explained by our observation that both methods employ distance based methods for similarity measurement. Their selected features are proximity only in values. There are positively correlated, negatively correlated, and interdependent segments which are not accounted for. As a result, they are not able to discover meaningful feature structure latent in the graph.



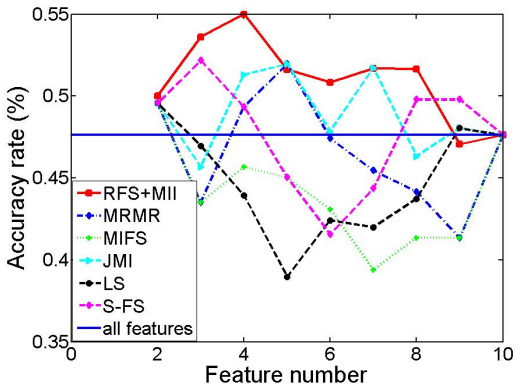

(a) Pendigits



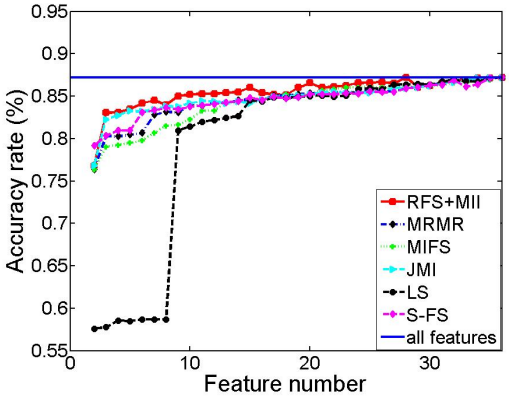

(b) Letter



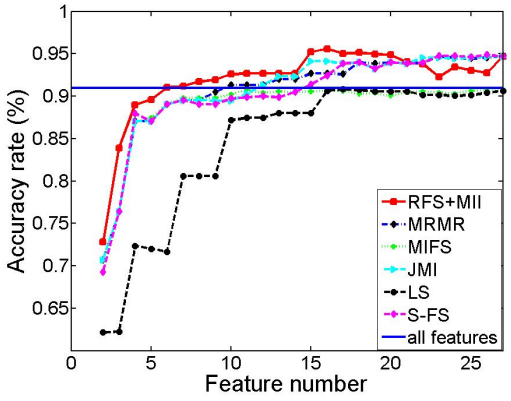

(c) Wine




(d) Vowel




(e) Satimage




(f) Dna

Figure 3.3: Accuracy rate vs. the number of selected features on multi class data sets.

### 3.4.2 Classification Accuracy

The classification accuracies obtained with different feature subsets are shown in Fig. 3.3. As shown by the results, it is clear that our proposed method RFS+MII is, by and large, superior to the alternative feature selection methods. Specifically, it selects a both smaller and better performing (in terms of classification accuracy) set of discriminative features on all the six multi class data sets. Moreover, RFS+MII rapidly converges to the best results, with typically smaller size of features. Each of the alternative methods, usually require more features achieve a comparable result. This may be explained by the fact that the MII criterion is sensitive to the relations among features combinations, and as a result can be used to seek third or even higher order dependencies among the relevant features. As a result the optimal feature combinations can be located so as to remove redundant features.

Our RFS+MII algorithm consistently outperforms the alternative MI-based feature selection algorithms (e.g. MRMR, MIFS and JMI) in all cases. Our algorithm performs especially well when the number of selected features is small (see Fig. 3.3(b, d, f)). The reason for this is that the alternative MI-based methods (i.e. MRMR and MIFS) select features in a greedy way. Commencing from an empty feature pool, features are added into the pool one by one until the user-defined number is reached. As a result, they may neglect the possible correlation between different features (referred to the conditional redundancy term $I(S; C|f_i)$). As a result, there is a tendency to overestimate the redundancy between features. Thus some important features can be discarded, which in turn leads to information loss.

The best results for each method together with their corresponding size of selected feature subset cardinality are shown in Table. 3.5. In the table, the classification accuracy is shown followed by the optimal number of features selected in brackets. From Table. 3.5, it is clear that RFS+MII outperforms the alternative methods. However, in the Letter and Satimage data sets, although the alternative methods can obtain the best per-

| Dataset | Pendigits | Letter | Wine | Vowel | Satimage | Dna |
|---|---|---|---|---|---|---|
| MRMR | 68.1%(3) | **82.7%(16)** | 98.3%(3) | 52%(5) | **87.2%(36)** | 94.8%(27) |
| MIFS | 73.2%(3) | **82.7%(16)** | 97.2%(3) | 49.6%(2) | **87.2%(36)** | 90.8%(26) |
| JMI | 70.2%(4) | **82.7%(16)** | 97.8%(3) | 52%(5) | **87.2%(36)** | 94.7%(27) |
| LS | 69.6%(5) | **82.7%(16)** | 97.8%(3) | 49.6%(2) | **87.2%(36)** | 90.8%(17) |
| S-FS | 65.4%(5) | **82.7%(16)** | 96.1%(4) | 52.2%(3) | **87.2%(36)** | 94.9%(26) |
| RFS+MII | **75.6%(3)** | **82.7%(12)** | **98.9%(4)** | **55%(4)** | **87.2%(28)** | **95.6%(16)** |

Table 3.5: The best result of all methods and their corresponding size of selected feature subset on the multi class data sets

formance using the entire feature-set, our proposed method RFS+MII achieves the same classification accuracy with much smaller number of features, (i.e., only 12 features for the Letter data set and 28 features for the Satimage data set). This implies that the discriminative information exists in a small set of features, which can be effectively selected by RFS+MII and then those features can be used to construct classifiers effectively.

### 3.4.3 Redundancy Rate

Table. 3.6 shows the comparative results of our proposed method with the alternative feature selection methods using the top $t$ features. In the table, the boldfaced values are the lowest redundancy rates. The subset obtained by our proposed scheme has the least redundant. This further verifier that our propose algorithm is able to remove redundant features.

The results from the accuracy rate in Table. 3.5 and redundancy rate in Table. 3.6 together indicate that RFS+MII both contains the least redundancy, and result in highest accuracy. They also underline necessity of removing redundant features for improving learning performance. It should also be observed that the MRMR algorithm also produces

| Dataset | Pendigits | Letter | Wine | Vowel | Satimage | Dna |
|---|---|---|---|---|---|---|
| MRMR | 0.5420 | 0.2549 | **0.2398** | 0.1820 | 0.2099 | 0.1467 |
| MIFS | 0.4441 | 0.2774 | 0.2463 | 0.1847 | 0.2151 | 0.1537 |
| JMI | 0.4597 | 0.2549 | 0.2428 | 0.1820 | 0.2175 | 0.1490 |
| LS | 0.6145 | 0.3075 | 0.2512 | 0.1872 | 0.2261 | 0.1730 |
| S-FS | 0.7681 | 0.3199 | 0.2482 | 0.1861 | 0.2261 | 0.1510 |
| RFS+MII | **0.3365** | **0.2320** | **0.2398** | **0.1792** | **0.2044** | **0.1450** |

Table 3.6: Averaged redundancy rate of subsets selected using different algorithms

low redundancy rates. However, it does not perform as well in the terms of classification accuracy. This can be explained by the observation that: in MRMR, feature contributions to classification is considered individually by evaluating the correlation between each feature and the class label. However, the class label may be jointly determined by a set of features. This interaction among features is not considered by MRMR.

## 3.5 Conclusion

In this chapter, we have presented a new graph based information theoretic approach to feature selection. The proposed feature selection method offers three major advantages. First, by incorporating mutual information for pairwise feature similarity measure, we establish a novel feature graph. With this representation, the informativeness latent in the features can be more effectively modeled. Second, we have reduced the optimal search space from the original feature set to a relatively smaller relevant feature subset subject to the cohesiveness in feature mutual information. Thirdly, we have preserved features associated with the greatest amount of joint information through refining the relevant feature subset based on the multidimensional interaction information (MII). All these

advantages enable an effective performance of our framework in feature selection.

There are a number of directions in which the research described in this chapter can be extended. In the following chapter, we will use hypergraphs to represent higher order feature relationships. This will provide a natural way of measuring the representation power of the RFS extracting process.

# Chapter 4

# Hypergraph based Information-theoretic Feature Selection

In many situations the graph representation for relational patterns can lead to substantial loss of information. This is because in real-world problems objects and their features tend to exhibit multiple relationships rather than simple pairwise ones. A natural way of remedying the information loss described above is to represent the data set as a hypergraph instead of a graph. Hypergraph representations allow vertices to be multiply connected by hyperedges and can hence capture multiple or higher order relationships among features. Due to their effectiveness in representing multiple relationships, in this chapter, we draw on recent work on hypergraph clustering to select the most informative feature subset (mIFS) from a set of objects using high-order (rather than pairwise) similarities. There are two novel ingredients. First, we use a new information theoretic criterion referred to as the multidimensional interaction information (MII) to measure the significance of different feature combinations with respect to the class labels. Secondly, we use hypergraph clustering to select the most informative feature subset (mIFS), which has both low redundancy and strong discriminating power. The advantage of MII is that it incorporates third or higher order feature interactions. Hypergraph clustering, which extracts the most informative features. The size of the most informative feature subset (mIFS) is determined automatically. Experimental results demonstrate the effectiveness of our feature selection

method on a number of standard data-sets.

### Contributions

In summary, there are three main contributions in this chapter. The first is that we develop a hypergraph representation based on the attributes of feature vectors, i.e. a feature hypergraph. With this representation, the structural information latent in the data can be more effectively modeled. The second is that unlike most existing graph or hypergraph methods, which use distance metrics (i.e. Euclidean distance or Pearson's correlation coefficient) to represent the weight of edge or hyperedge, here we determine the weight of the hyperedges using an information measure referred to as multidimensional interaction information (MII). There are two advantages of MII. First, it effectively reflects functional similarity, such as the positive or negative correlation and interdependency among features. Second, it is sensitive to the relations between feature combinations, and as a result can be used to seek third or even higher order dependencies between the relevant features. Thirdly, we can use the method to locate the most informative feature subset (mIFS) by hypergraph cluster analysis. In contrast with existing feature selection methods, our proposed methods is able to determine the number of relevant features automatically.

### Chapter outline

The remainder of this chapter is organized as follows. Section 4.1 describes the relevant background on hypergraph. Section 4.2 describes how to combine MII criterion and hypergraph cluster analysis to locate most informative feature subset (mIFS). The classification methods are presented in Section 4.3. In Section 4.4, we first give a description of the real-world benchmark data sets. We then examine the performance of our proposed hypergraph based information-theoretic feature selection method, and compare the classification results with those obtained by alternative feature selection methods. Finally, conclusions are presented in Section 4.5.

## 4.1 Hypergraph Fundamentals

A hypergraph is defined as a triplet $H = (V, E, \mathbf{W})$, where $V = \{1, \ldots, n\}$ is the node-set, $E$ is a set of non-empty subsets of $V$ or hyperedges and $\mathbf{W}$ is a weight function which associates a real value with each edge. A hypergraph is a generalization of a graph. Unlike graph edges which consist of pairs of vertices, hyperedges can be arbitrarily sized sets of vertices. Examples of a hypergraph are shown in Fig. 4.1. For the hypergraph, the vertex set is $V = \{v_1, v_2, v_3, v_4, v_5\}$, where each vertex represents a feature, and the hyperedge set is $E = \{e_1 = \{v_1, v_3\}, e_2 = \{v_1, v_2\}, e_3 = \{v_2, v_4, v_5\}, e_4 = \{v_3, v_4, v_5\}\}$. The number of vertices constituting each hyperedge represent the order of the relationship between features.

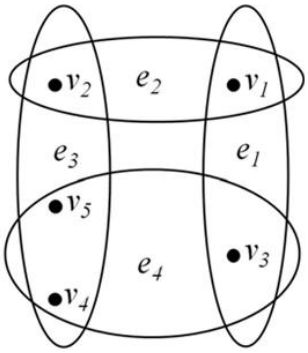



Figure 4.1: Hypergraph example

Hypergraph clustering is an extension of graph-based clustering to the hypergraph domain. In graph-based clustering, the aim is to find sets of nodes that exhibit strong within cluster inter-connectivity and weak between cluster connectivity. This is usually expressed in terms of a clustering criterion defined over the edges of the graph representing pairwise relationships between the objects to be clustered. Examples of such criteria include the normalized cut [33] and the dominant set [43]. In hypergraphs, the relationships between objects are higher order, and the concept of the cluster must be extended to reflect this. In a hypergraph, an edge represents a set of nodes participating in a mutual

high order relation. Thus hypergraph clustering, must be expressed in terms of a criterion reflecting overlapping sets of nodes participating in different hyperedges.

## 4.2 Feature Selection Using Hypergraph Cluster Analysis

In this section we aim to utilize the hypergraph cluster analysis to perform feature selection. Using a hypergraph representation of the features, there are two steps to the algorithm, namely a) computing the weight matrix $\mathbf{W}$ based on the multidimensional interaction information (MII) among feature vectors, b) hypergraph cluster analysis to select the most informative feature subset (mIFS). In the remainder of this chapter we describe these elements of our feature selection algorithm in more detail.

### 4.2.1 Computing Weight Matrix

Instead of using the Euclidean distance or Pearson's correlation coefficient, our similarity measure employs a new information measure referred to as multidimensional interaction information criterion to evaluate the interdependence of features to reflect functional similarity such as positive and negative correlation and interdependency among features. The use of this information measure allows hypergraph cluster analysis to locate the most informative feature subset (mIFS) using higher-order interaction information reflecting similarity.

According to the definition of MII in Chapter 3, we propose to use the following multidimensional interaction information to measure the high-order relevance of features. For a set of features $F = \{f_1, \ldots, f_k\}$, the relevance degree among these feature vectors

can be defined as

$$\mathbf{W}(f_{1:k}; C) = \sum_{i=1}^{k} I(f_i; C) - \sum_{f \subseteq F} I(\{f\}) + \sum_{f \subseteq F} I(\{f\}|C) \, . \tag{4.1}$$

The above relevance degree definition consists of three terms. The first $\sum_{i=1}^{k} I(f_i; C)$ referred to as *relevancy* is the sum of each individual feature's relevance with the class set $C$. The second term of the form $\sum_{f \subseteq F} I(\{f\})$ referred to as *redundancy* is used to measure the redundant among features. The third term $\sum_{f \subseteq F} I(\{f\}|C)$ measures the influence of the features combination on the class set $C$. We refer to this as the *class-conditional correlations*. Therefore, a large value of $\mathbf{W}(f_{1:k}; C)$ means that both $\sum_{i=1}^{k} I(f_i; C)$ and $\sum_{f \subseteq F} I(\{f\}|C)$ are large (indicating features $F = \{f_1, \ldots, f_k\}$ are informative with respect to the class set $C$) and $\sum_{f \subseteq F} I(\{f\})$ is small (indicating features $F = \{f_1, \ldots, f_k\}$ are less redundant).

### 4.2.2 Most Informative Feature Subset Selection

Let $H = (V, E, \mathbf{W})$ be a hypergraph. We can locate the most informative feature subset (mIFS) by finding the solutions of the following non-linear optimization problem that maximizes the functional [64]

$$f(\boldsymbol{a}) = \sum_{e \in E} \mathbf{W}(e) \prod_{i \in e} a_i \, . \tag{4.2}$$

subject to $\boldsymbol{a} \in \triangle$, where the multidimensional solution vector $\boldsymbol{a}$ fall on the simplex $\triangle = \{\boldsymbol{a} \in \mathbb{R}^d : \boldsymbol{a} \geq 0, \sum_{i=1}^{d} a_i = 1\}$. The set $E$ represents the set of hyperedges, so each $e \in E$ is a hyperedge, i.e. a set of $k$ vertices. The solution vector $\boldsymbol{a}$ is $d$-dimensional where $d$ is the number of vertices (features), representing a probability distribution over the set of vertices (features). In other words, $\boldsymbol{a}$ is an $d$-dimensional indicator vector such that $a_i > 0$ if $i$-th feature belongs to the dominant cluster which corresponds to the most informative feature subset (mIFS). A feature $f_i$ is selected if and only if the $i$-th component of $\boldsymbol{a}$

is positive, i.e. $a_i > 0$. Consequently, the number of selected informative features $m$ ($m < d$) is determined by the number of positive components of $\boldsymbol{a}$. The function $\mathbf{W}$ is a weight function which associates a real value of weight with each hyperedge. The local maximum of $f(\boldsymbol{a})$ can be solved using the Baum-Eagon inequality equation [39] and leads to the iterative update:

$$z_i = \frac{a_i \partial_i f(\boldsymbol{a})}{\sum_{j=1}^{d} a_j \partial_j f(\boldsymbol{a})}, i = 1, \ldots, d \,. \tag{4.3}$$

where $f(\boldsymbol{a})$ is a homogeneous polynomial in the variables $a_i$ and $z = \mathcal{M}(\boldsymbol{a})$ is a growth transformation of $\boldsymbol{a}$. The Baum-Eagon inequality $f(\mathcal{M}(\boldsymbol{a})) > f(\boldsymbol{a})$ provides an effective iterative means for maximizing polynomial functions are probability domains. By taking the support of $\boldsymbol{a}$, we can locate the mIFS under our framework. The complexity of finding mIFS is thus $O(t|E|)$, where $|E|$ is the number of hyperedges of the hypergraph and $t$ is the average number of iteration needed to converge. Note that $t$ never exceeded 100 in our experiments.

The Baum-Eagon inequality only can be used on polynomials with non-negative coefficients, whereas our weight function (see Equation (4.1)) could lead to negative weights. In order to solve this problem, we convert the polynomial with negative coefficients into a polynomial with nonnegative coefficients by adding a term $\beta * (\sum_{j=1}^{d} a_j)^k$ with arbitrarily large positive $\beta$. This does not affect the maximizers, and since $\sum_{j=1}^{d} a_j = 1$ it is equivalent to adding a constant to the objective function. According to the value of $\boldsymbol{a}$, all features $F$ fall into two disjoint subsets, $S_1(\boldsymbol{a}) = \{f_i | a_i = 0\}$ and $S_2(\boldsymbol{a}) = \{f_i | a_i > 0\}$. The set of nonzero variables $S_2(\boldsymbol{a})$ is our selected most informative feature subset (mIFS). The sequence of steps shown in Algorithm 2 illustrates our algorithm in detail.

### 4.2.3 Complete Feature Ranking

According to the value of $\boldsymbol{a}$, all features $F$ fall into two disjoint subsets, $S_1(\boldsymbol{a}) = \{f_i | a_i = 0\}$ and $S_2(\boldsymbol{a}) = \{f_i | a_i > 0\}$. The set of positive variables $S_2(\boldsymbol{a})$ is our selected most in-

**Algorithm 2:** Hypergraph based information-theoretic feature selection algorithm

**Input**: Dataset $\mathbf{X}$ with all features $d$

**Output**: The selected feature subset corresponding to the non-zero elements of $d$-dimensional indicator vector $\boldsymbol{a}$

1) Construct a hypergraph in which each node corresponds to a feature, the weight matrix $\mathbf{W}$ for the hyperedges is computed using Equation (4.1) ;

2) Compute $\boldsymbol{a}$ based on $\mathbf{W}$ by Equation (4.2) and iteratively update $\boldsymbol{a}$ by Equation (4.3) ;

3) Select the most informative feature subset (mIFS) from the non-zero elements of $d$-dimensional indicator vector $\boldsymbol{a}$.

formative feature subset (mIFS). We rank the features contained in the unselected feature set $S_1(\boldsymbol{a}) = \{f_i | a_i = 0\}$ using feature selection algorithm MRMR [27]. Consequently, we can obtain a complete feature ranking list, which starts from the size of most informative feature subset (mIFS) and ends at any user-specified fixed number.

## 4.3 Classification Strategy

After finding the discriminating features, we then run classification experiments on them by two classifiers. For multi class data set, we use the linear SVM with LIBSVM [14]. However, for binary class data set, we apply the variational EM (VBEM) algorithm [11] to fit a mixture of Gaussians model to the selected feature subset. After learning the mixture model, we use the a class posteriori probabilities, see Equation (4.4), to classify one of testing sample data. Given a sample, we first compute its selected feature vector $b$ through feature selection. Then we compute its a posteriori probabilities $r_c$, the mean vectors $\hat{b}_c$, and the precision matrices $\Lambda_c$, where $c \in c_1, \ldots, c_l$ and $l$ is the number of class for the data. For example, in binary classification, if $r_{c_1} > r_{c_2}$ then the sample is classified as

class $c_1$. Otherwise, the sample is classified as $c_2$. The posterior probabilities are given by

$$r_{nk} \propto \pi_k |\Lambda_k|^{\frac{-1}{2}} \exp\{\frac{1}{2}(x_n - \mu_k)^T \Lambda_k (x_n - \mu_k)\} \; . \tag{4.4}$$

where $k = 1, \ldots, C$ are the mixture components and $n = 1, \ldots, N$ denotes the data index. The model parameters $\pi_k$, $\mu_k$ and $\Lambda_k$ are respectively the a priori probability, the mean of selected feature vectors and the precision matrices of the $k$-th component. In the variational Bayesian EM (VBEM) algorithm, all of these model parameters are characterized by distributions, each of which has hyper-parameters, which take into account the uncertainty in the parameter estimation. The parameters $r_{nk}$ represent the responsibility the $k$-th component takes in explaining the $n$-th observation. The posteriori probability can be arranged into a matrix $R = (r_{nk})$ and where it had to satisfy the condition: $0 \leq r_{nk} \leq 1$.

## 4.4 Experiments and Comparisons

The data sets used to test the performance of our proposed algorithm are benchmark data sets from the UCI Machine Learning Repository and Statlog. Table. 4.1 summarizes the extents and properties of the ten data-sets.

Our proposed feature selection method (referred to as the MII+HG) utilizes the multidimensional interaction information criterion and hypergraph cluster analysis for feature selection. It involves applying the MII criterion as the weight measure and then using hypergraph cluster analysis to locate the most informative feature subset (mIFS). We compare the classification results from our proposed method MII+HG with those obtained using both alternative MI-based criterion methods and graph-based methods. These methods are the MRMR algorithm [27], the MIFS algorithm [55], the JMI algorithm [29], the Laplacian score (LS) [73] and the subset-level based Fisher score (S-FS) [24]. The evaluation scheme is depicted in Fig. 4.2. We first explore the discriminating

| Data-set | From | Training data | Testing data | Features | Classes |
|---|---|---|---|---|---|
| Ion | UCI | 200 | 151 | 32 | 2 |
| Breast cancer | UCI | 399 | 300 | 10 | 2 |
| Sonar | UCI | 108 | 100 | 60 | 2 |
| Pima | UCI | 468 | 300 | 8 | 2 |
| Wine | UCI | 100 | 78 | 13 | 3 |
| Pendigits | UCI | 7494 | 3498 | 16 | 10 |
| Vowel | UCI | 528 | 462 | 10 | 11 |
| Letter | Statlog | 15000 | 5000 | 16 | 26 |
| Satimage | Statlog | 4435 | 2000 | 36 | 6 |
| Dna | Statlog | 2000 | 1186 | 180 | 3 |

Table 4.1: Summary of UCI and Statlog benchmark data sets

features using the different methods on binary classification and clustering problems (i.e. Ion, Sonar, Pima and Breast cancer dataset). For performance comparison, we apply the VBEM algorithm to the selected feature subset for the purpose of classification. Next, we extend our attention from the binary to the multi class case and compare the classification performance for the different methods using an SVM classifier with the features selected by these methods. Here we use the linear SVM with LIBSVM [14].

Using the feature selection algorithms outlined above, we first examine the classification performance on binary classification and clustering problems (i.e. Ion, Sonar, Pima and Breast cancer dataset). The classification accuracies obtained on different feature subsets are shown in Fig. 4.3. From the figure, it is clear that using the most informative feature subset (mIFS), MII+HG achieves the best classification accuracy. This is higher than that obtained using alternatively sized feature subsets. Adding some highly ranked features from outside the mIFS results in a deterioration of accuracy. Moreover, MII+HG

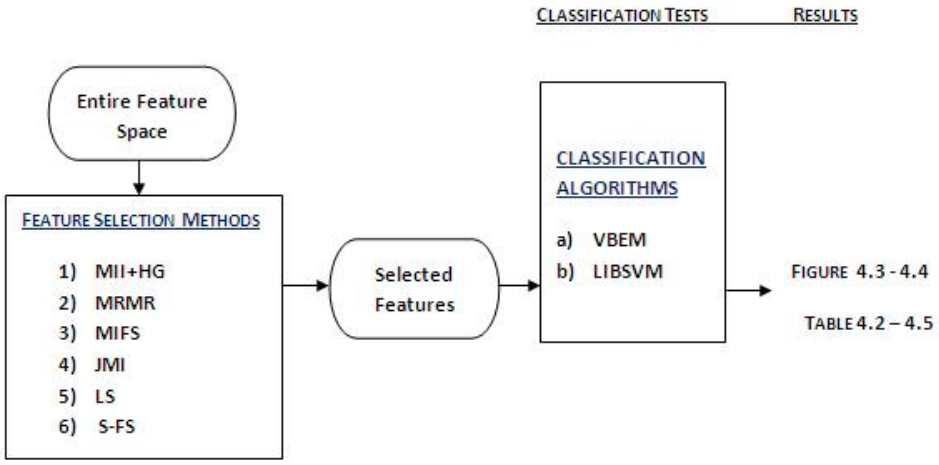
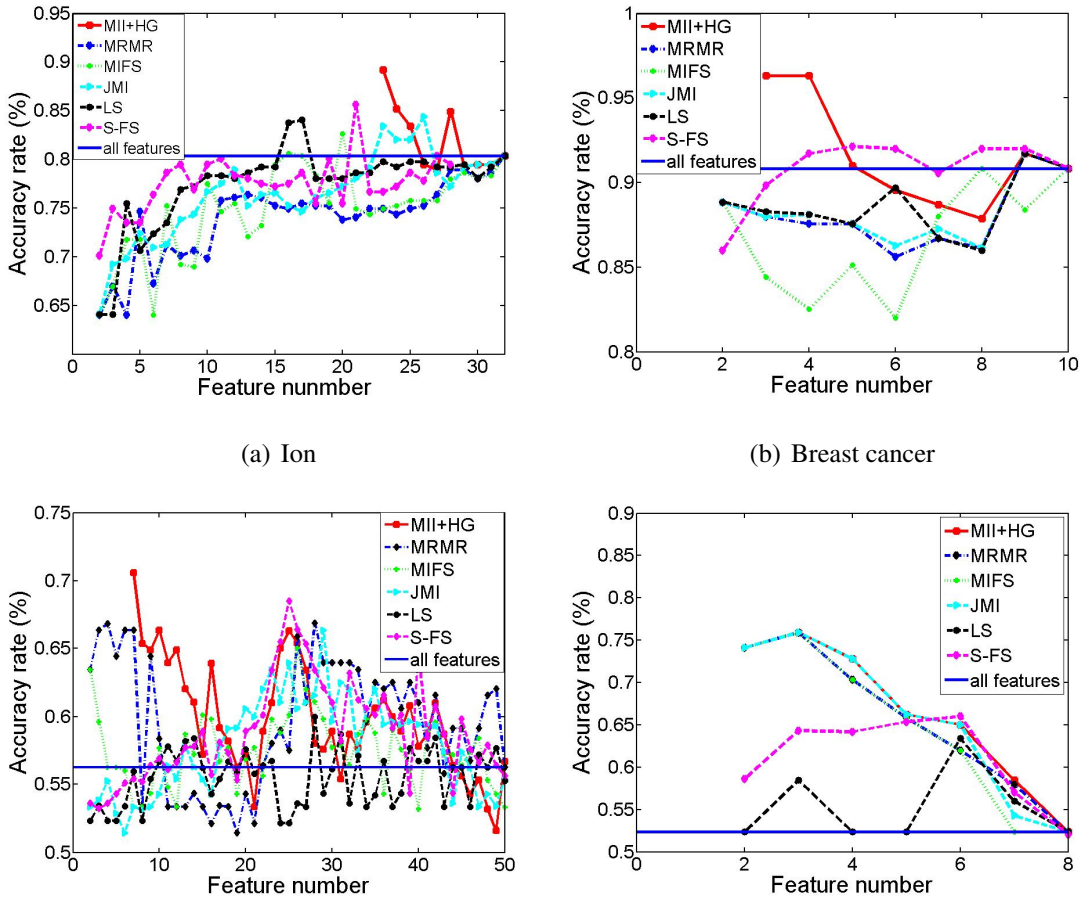



Figure 4.2: The scheme for evaluating the classificatory effectiveness of selected features

is, by and large, superior to the alternative feature selection methods. This implies that our proposed method is able to locate both the optimal size of feature subset and performs accurate classification of samples based on a very few of the most important features. However, for the alternative feature selection methods, user input is required to supply the number of features to be selected in advance. This is because they focus on ranking features based on their scores. In other word, they select the best $m$ features in a greedy way. Commencing from an empty feature pool, features are added into the pool one by one until the user-defined number is reached. As a result, they may neglect the possible correlation between different features and thus can not produce an optimal feature subset.

To make a detailed comparison, we summarize the classification accuracy on binary class data sets for the different methods in Table. 4.2. In the last row, the classification accuracy for MII+HG and the automatically determined size of most informative feature subset (mIFS) are reported. To make a fair comparison, suppose that the determined size of most informative feature subset (mIFS) is $m$. For each alternative method, we measure the classification accuracy for $m-1$, $m$ and $m+1$ features, and take the best result as the baseline performance. As shown by the results in Table. 4.2, at small dimensionality




(a) Ion

(b) Breast cancer

(c) Sonar

(d) Pima

Figure 4.3: Accuracy rate vs. the number of selected features on binary class data sets.

(i.e. Pima data set) there is little difference between our proposed method and the alternative MI-based methods. However, at higher dimensionality, the features subset located by MII+HG clearly has a higher discriminability power than the features selected by the alternative feature selection methods. This may be explained by the fact that the traditional feature selection methods may work well on low dimensional binary classification and clustering problem. However, they are very likely to fail in high dimensional binary or multi classification and clustering problems. The reason for this is that the alternative MI-based methods (i.e. MRMR and MIFS) select features in a greedy way and neglect the

| Dataset | Ion | Breast cancer | Sonar | Pima |
|---|---|---|---|---|
| MRMR | 74.93% | 88.84% | 66.35% | 75.91% |
| MIFS | 75.21% | 88.84% | 56% | 75.91% |
| JMI | 83.4% | 88.84% | 53.37% | 75.91% |
| LS | 79.77% | 88.84% | 55.98% | 58.46% |
| S-FS | 85.63% | 91.7% | 56.9% | 64.32% |
| MII+HG | 89.19%(23) | 96.3%(3) | 70.56%(7) | 75.91%(3) |

Table 4.2: The Performance of VBEM at the given number of features selected by different methods on the binary class data sets

conditional redundancy term $I(x_i, S|C)$. As a result, there is a tendency to overestimate the redundancy between features. Thus some important features can be discarded, which in turn leads to information loss. In order to further verify this assertion, in the following experiments, we extend our attention from the binary to the multi class case and compare the performance of different methods.

The best result for each method together with their corresponding size of selected feature subset cardinality are shown in Table. 4.3. In the table, the classification accuracy is shown first and the optimal number of features selected is reported in brackets. Overall, MII+HG achieves the highest degree of dimensionality reduction, i.e. it selects a smaller feature subset compared with those obtained by the alternative methods. For example, in the Sonar data set, the best result obtained by the alternative feature selection methods is 68.5% on the S-FS algorithm with 24 features. However, our proposed method MII+HG gives a better accuracy 70.56% when only 7 features are used. The results further verify that our feature selection method can guarantee the optimal size of feature subset, as it not only achieves a higher degree of dimensionality reduction but it also gives a better discriminability power. We also observe that in most cases (i.e. the Ion, Breast cancer and

| Dataset | Ion | Breast cancer | Sonar | Pima |
|---|---|---|---|---|
| MRMR | 80.34%(32) | 91.7%(9) | 66.87%(27) | 75.91%(3) |
| MIFS | 82.62%(20) | 90.84%(8) | 65%(25) | 75.91%(3) |
| JMI | 84.33%(26) | 91.7%(9) | 66.34%(29) | 75.91%(3) |
| LS | 84.05%(17) | 91.7%(9) | 60%(28) | 63.41%(6) |
| S-FS | 85.63%(21) | 92.13%(4) | 68.5%(24) | 66.01%(6) |
| MII+HG | 89.19%(23) | 96.3%(3) | 70.56%(7) | 75.91%(3) |

Table 4.3: The best result of all methods and their corresponding size of selected feature subset on on the binary class data sets

Sonar data set), S-FS (subset-level Fisher score) gives a better result than the alternative methods. The reason is that unlike traditional methods which treat each feature individually and hence are suboptimal, the S-FS method directly optimizes the score over the entire selected feature subset. As a result, a better feature subset can be obtained. In the following experiments, we extend our study from the binary to the multi class case and compare the performance of the alternative methods.

Fig. 4.4 shows the plots of the classification accuracy versus the number of selected features on multi class data sets. Our proposed MII+HG algorithm consistently outperforms the alternative methods on all six multi class data sets by selecting a smaller set of discriminative features than the alternatives. This is reflected by the classification results. There are two reasons for this improvement in performance. First, the multidimensional interaction information (MII) criterion is applied to measure the hyperedge weight, and this can capture the effects of multiple or higher order dependencies between the features and the class. Second, hypergraph clustering analysis simultaneously considers the information-contribution for each feature together with the correlation between features. Thus structural information latent in the data can be effectively identified. As a result the

optimal feature combinations can be located so as to guarantee an optimal feature subset. On the other hand, the graph-based methods (i.e. LS and S-FS) fail to locate the most discriminative features, as shown in Fig. 4.4(a, b, c, d). This may be explained by our observation that both methods employ distance based methods for similarity measurement. Their selected features are proximity only in values. There are positively correlated, negatively correlated, and interdependent segments which are not accounted for. As a result, they are not able to discover meaningful feature structure latent in the graph.

| Dataset | Pendigits | Letter | Wine | Vowel | Satimage | Dna |
|---|---|---|---|---|---|---|
| MRMR | 68.1% | 78.3% | 98.3% | 49.6% | 85% | 92.7% |
| MIFS | 73.2% | 71.9% | 97.2% | 49.6% | 85.2% | 90.6% |
| JMI | 70.2% | 78.3% | 97.8% | 51.3% | 85% | 94.1% |
| LS | 69% | 73% | 97.8% | 49.6% | 84.9% | 90.6% |
| S-FS | 65.2% | 68% | 96.1% | 52.2% | 84.9% | 92.4% |
| MII+HG | 76.2%(3) | 82.7%(10) | 98.9%(4) | 56.7%(3) | 87.2%(17) | 96.3%(15) |

Table 4.4: The Performance of LIBSVM at the given number of features selected by different methods on the multi class data sets

We summarize the classification accuracy on multi class data sets for the different methods in Table. 4.4. Again, MII+HG is, by and large, superior to the alternative methods, giving the optimal size of feature subset. The best results for each method together with their corresponding size of selected feature subset cardinality are shown in Table. 4.5. In the table, the classification accuracy is shown followed by the optimal number of features selected in brackets. From Table. 4.5, it is clear that MII+HG outperforms the alternative methods. However, in the Letter and Satimage data sets, although the alternative methods can obtain the best performance using the entire feature-set, our proposed method MII+HG achieves the same classification accuracy with much smaller number

| Dataset | Pendigits | Letter | Wine | Vowel | Satimage | Dna |
|---|---|---|---|---|---|---|
| MRMR | 68.1%(3) | 82.7%(16) | 98.3%(3) | 52%(5) | 87.2%(36) | 94.8%(27) |
| MIFS | 73.2%(3) | 82.7%(16) | 97.2%(3) | 49.6%(2) | 87.2%(36) | 90.8%(26) |
| JMI | 70.2%(4) | 82.7%(16) | 97.8%(3) | 52%(5) | 87.2%(36) | 94.7%(27) |
| LS | 69.6%(5) | 82.7%(16) | 97.8%(3) | 49.6%(2) | 87.2%(36) | 90.8%(17) |
| S-FS | 65.4%(5) | 82.7%(16) | 96.1%(4) | 52.2%(3) | 87.2%(36) | 94.9%(26) |
| MII+HG | 76.2%(3) | 82.7%(10) | 98.9%(4) | 56.7%(3) | 87.2%(17) | 96.3%(15) |

Table 4.5: The best result of all methods and their corresponding size of selected feature subset on the multi class data sets

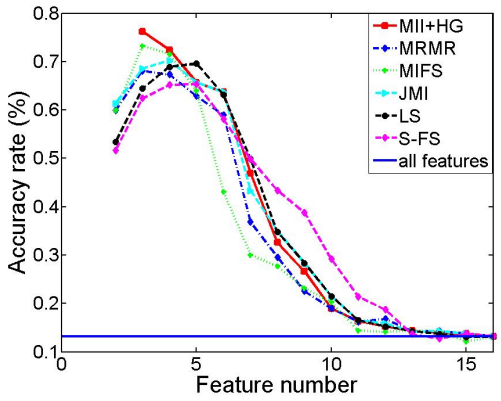
of features, (i.e., only 10 features for the Letter data set and 17 features for the Satimage data set). This implies that the discriminative information exists in a small set of features, which can be effectively selected by MII+HG and then those features can be used to construct classifiers effectively. The results from the accuracy rate in Table. 3.4 and Table. 4.4 together reveal the significance of feature hypergraph representation over feature graph representation for relevant feature subset extraction. As shown by the results, the feature hypergraph based method is, by and large, superior to the feature graph based method. Specifically, it selects a both a smaller and better performing (in terms of classification accuracy) set of relevant feature subset on most of the data sets.

## 4.5 Conclusions

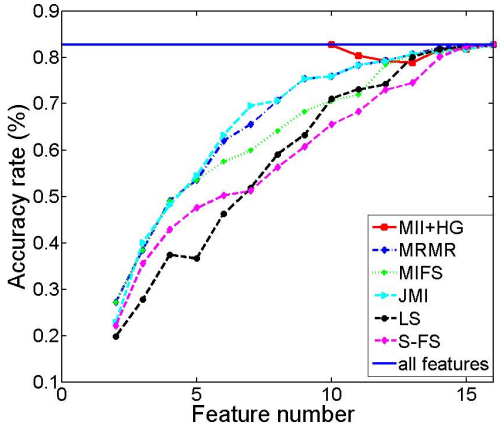
In this chapter, we have presented a new hypergraph based information theoretic approach to feature selection. The proposed feature selection method offers three major advantages. First, the MII criteria is applied to measure the weight of hyperedges, which takes into account high-order feature interactions, overcoming the problem of overselected feature

redundancy. As a result the features associated with the greatest amount of joint information can be preserved. Second, hypergraph clustering analysis is used to locate the most informative feature subset (mIFS), therefore, the optimal size of feature subset can be automatically determined. Third, the variational EM (VBEM) algorithm and a Gaussian mixture model are applied to the selected feature subset. This improves the overall classification accuracies by automatically determining the number of clusters present in the data during the learning process.

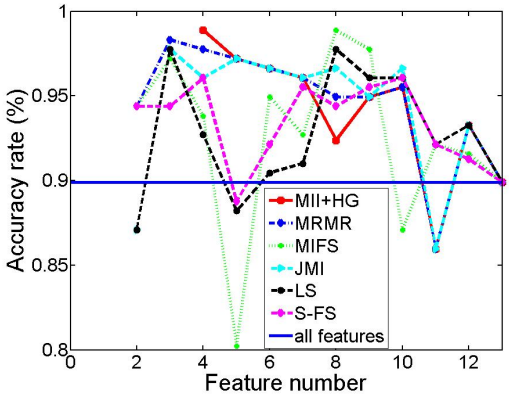
The proposed feature selection methods above are supervised feature selection methods. While the labeled data required by supervised feature selection can be scarce, there is usually no shortage of unlabeled data. Hence, there are obvious attractions in developing unsupervised feature selection algorithms which can utilize this data. Therefore, in the following two chapters (Chapter 5 and Chapter 6), we extend our attention to unsupervised feature selection methods.



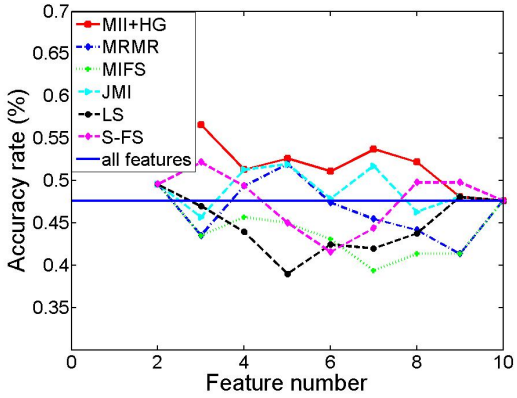

(a) Pendigits



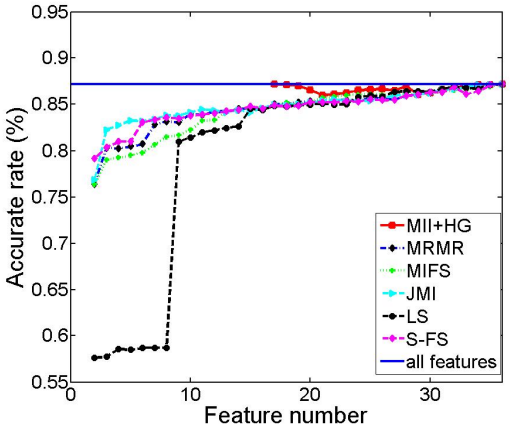

(b) Letter



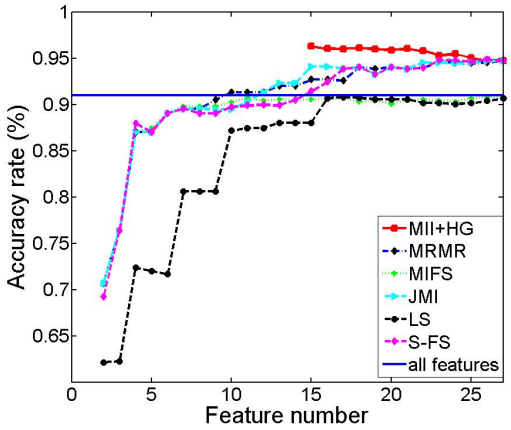

(c) Wine




(d) Vowel




(e) Satimage




(f) Dna

Figure 4.4: Accuracy rate vs. the number of selected features on multi class data sets.

# Chapter 5

# Kernel Entropy Analysis for Unsupervised Feature Selection

Most existing feature selection methods focus on ranking individual features based on a utility criterion, and select the optimal feature set in a greedy manner. However, the feature combinations found in this way do not give optimal classification performance, since they neglect the correlations among features. In an attempt to overcome this problem, in this chapter, we develop a novel feature selection technique using the spectral data transformation and by using $\ell_1$-norm regularized models for subset selection. In this chapter, we propose a new two-step spectral regression technique for unsupervised feature selection. In the first step, we use kernel entropy component analysis (kECA) to transform the data into a lower-dimensional space so as to improve classes separation. Second, we use $\ell_1$-norm regularization to select the features that best align with the data embedding resulting from kECA. The advantage of kECA is that dimensionality reducing data transformation maximally preserves entropy estimates for the input data whilst also best preserving the cluster structure of the data. Using $\ell_1$-norm regularization, we cast feature discriminant analysis into a regression framework which accommodates the correlations among features. As a result, we can evaluate joint feature combinations, rather than being confined to consider them individually. Experimental results demonstrate the effectiveness of our feature selection method on a number of standard face data-sets.

**Chapter outline**

The outline of this chapter is as follows. In Section 5.1 and Section 5.2, we respectively describe the two different strategies (kernel PCA and kernel ECA) used to transform the data into a lower-dimensional space. Section 5.3 presents the robust feature selection based on $\ell_1$-norms. A detailed description of the feature evaluation indices is given in Section 5.4. In Section 5.5, we first give a description of the real-world benchmark face data sets. We then examine the performance of our proposed method, and compare the classification results with those obtained by alternative feature selection methods. Finally, Section 5.6 concludes this chapter.

## 5.1 Kernel PCA

One of the best known non-linear data transformation methods for similarity data is the kernel principal components analysis technique of $Schölkopf$ et al. [10]. Suppose the original high-dimensional data is represented by $\mathbf{X} = [x_1, x_2, \ldots, x_N]$, where $x_i \in R^d, i = 1, \ldots, N$ and the number of features (dimensions) of the data set is $d$. The basic idea underpinning kernel PCA is that by using a nonlinear mapping $\Phi$, we implicitly perform PCA in a possibly high-dimensional space $\mathcal{F}$ which is related to the input space in a non-linear way.

The non-linear map from input space to feature space is given by $\Phi : R^d \rightarrow F$ such that $x_t \rightarrow \Phi(x_t), t = 1, \ldots, N$. Let $\Phi = [\Phi(x_1), \Phi(x_2), \ldots, \Phi(x_N)]$. To perform PCA in $\mathcal{F}$, we need to find an expression for the projection $P_{\alpha_i}\Phi$ of $\Phi$ onto a feature space principal axes $\alpha_i$, or onto a subspace $E_l$ spanned by the leading $l$ eigenvectors. This projection is achieved implicitly via the kernel function.

The estimated covariance matrix of the mapped data $\Phi(x_i)$ in kernel PCA is defined

as

$$C_{\Phi(x)} = \frac{1}{N}\sum_{i=1}^{N}\Phi(x_i)\cdot\Phi(x_i)^T \ . \tag{5.1}$$

By analyzing with PCA, we solve the eigenvector problem:

$$\begin{aligned}\lambda w_\Phi &= C_{\Phi(x)}w_\Phi = (\frac{1}{N}\sum_{i=1}^{N}\Phi(x_i)\cdot\Phi(x_i)^T)w_\Phi \\ &= \frac{1}{N}\sum_{i=1}^{N}(\Phi(x_i)^T\cdot w_\Phi)\Phi(x_i) \ . \end{aligned} \tag{5.2}$$

From Equation (5.2), we can see that all solutions $w_\Phi$ with $\lambda \neq 0$ lie in the span of $\Phi(x_1), \Phi(x_2), \ldots, \Phi(x_N)$, i.e. the coefficients $\alpha_i (i = 1, \ldots, N)$ exist such that

$$w_\Phi = \sum_{i=1}^{N}\alpha_i\Phi(x_i) \ . \tag{5.3}$$

By multiplying Equation (5.2) with $\Phi(x_t)^T$ from the left and substituting from Equation (5.3), we obtain

$$\lambda\sum_{i=1}^{N}\alpha_i(\Phi(x_t)^T\cdot\Phi(x_i)) = \frac{1}{N}\sum_{i=1}^{N}\alpha_i\{\Phi(x_i)\cdot\Phi(x_i)^T\sum_{j=1}^{N}\Phi(x_t)^T\Phi(x_j)\} \ . \tag{5.4}$$

where $t \in [1, N]$. Defining the $N \times N$ kernel matrix $\mathbf{K}$ by $K_{ij} = k(x_i, x_j) = \Phi(x_i)^T \cdot \Phi(x_j)$, the above equation turns reduces to an eigenvalue problem,

$$N\lambda\alpha = \mathbf{K}\alpha \ . \tag{5.5}$$

For the non-zero eigenvalues $\lambda_i$ and eigenvectors $\alpha_i = (\alpha_1, \ldots, \alpha_N)'$ subject to the normalization condition $N\lambda\alpha^T\alpha = 1$.

Note that kernel PCA performs dimensionality reduction by selecting the top $\alpha_l$ eigenvectors solely associated with the $l$ largest eigenvalues. From an information theoretic perspective, the resulting transformation may be based on uninformative eigenvectors.

## 5.2 Kernel Entropy Component Analysis

Although kernel Entropy Component Analysis, provides a means of overcoming this problem and is similar to kernel PCA in terms of solving an eigenvector equation, the underlying data transformation is based on the information content of the eigenvectors and eigenvalues. As a result it is not necessarily the leading eigenvalues and eigenvectors of the kernel matrix that are selected.

The Renyi quadratic entropy is given by [3]

$$H(p) = -\log \int p^2(x)dx \ . \tag{5.6}$$

where $p(x)$ is the probability density function generating the data set, or sample, $\mathbf{X} = x_1, x_2, \ldots, x_N$. Because of the monotonic nature of the logarithmic function, one can work instead with the quantity

$$V(p) = \int p^2(x)dx \ . \tag{5.7}$$

The Parzen estimate of $V(p)$ can be effected using the window density estimator with either a Gaussian or Radial Basis Function (RBF) as suggested in [20], and is given by

$$\hat{p}(x) = \frac{1}{N} \sum_{x_t \in S} k_\sigma(x, x_t) \ . \tag{5.8}$$

where $k_\sigma(x, x_t)$ is the kernel centered at $x_t$ with width governed by the parameter $\sigma$. Hence,

$$\begin{aligned} \hat{V}(p) &= \frac{1}{N} \sum_{x_t \in S} \hat{p}(x_t) = \frac{1}{N} \sum_{x_t \in S} \frac{1}{N} \sum_{x_{t'} \in S} k_\sigma(x_t, x_{t'}) \\ &= \frac{1}{N^2} \mathbf{1}^T \mathbf{K} \mathbf{1} \ . \end{aligned} \tag{5.9}$$

where, the element $(t, t^{'})$ of the $N \times N$ kernel matrix $\mathbf{K}$ is $k_\sigma(x_t, x_{t'})$ and $\mathbf{1}$ is an $N \times 1$ vector containing all ones.

Hence, the Renyi entropy estimator may be expressed in terms of the eigenvalues and eigenvectors of the kernel matrix, which may be decomposed as $\mathbf{K} = \mathbf{E}\mathbf{D}\mathbf{E}^T$, where $\mathbf{D}$ is

the diagonal matrix containing the eigenvalues $\lambda_1, \lambda_2, \ldots, \lambda_N$ and $\mathbf{E}$ is a matrix with the corresponding eigenvectors $\alpha_1, \alpha_2, \ldots, \alpha_N$ as columns. Rewriting the above, we have

$$\hat{V}(p) = \frac{1}{N^2} \sum_{i=1}^{N} (\sqrt{\lambda_i} \alpha_i^T \mathbf{1})^2 \, . \tag{5.10}$$

From the above expression indicates, kernel ECA is the projection of $\Phi$ onto those feature-space principal axes that maximally preserve the entropy estimate for the input data set (contributions $\sqrt{\lambda_i}\alpha_i^T\mathbf{1}$). These axes will, in general, not necessarily correspond to the leading eigenvalues $\lambda_i$ since $\alpha_i^T\mathbf{1}$ also contributes to the entropy estimate. Hence, kernel ECA is defined as an $m$-dimensional data transformation technique which projects $\Phi$ onto a subspace $E_m$ spanned by those $m$ kernel PCA axes contributing most significantly to the Renyi entropy estimate for the data. Hence $E_m$ is composed of a subset of kernel PCA axes but not necessarily those corresponding to the leading $m$ eigenvalues. Hence in kernel ECA, the selected transformation features best preserve the cluster structure of the data.

## 5.3 Robust Feature Selection Based on L1-Norms

Suppose the original high-dimensional data $\mathbf{X} \in R^{N \times d}$, that is, the number of samples is $N$ and the number of features (dimensions) of the data is $d$. The task of kernel ECA is to find a low-dimensional embedding space $\mathbf{Y}$ such that the clusters are more "obvious" or separate. Specifically, $\mathbf{Y} = [y_1, \ldots, y_C]$, $y_1$ is the embedding eigenvector which contributes most to Equation (5.10), correspondingly, the projection onto the kernel PCA axes that maximally preserve the entropy estimate for the input space data set. The intrinsic dimensionality of the data is $C$ and each vector $y_k$ reflects the data distribution along the corresponding dimension [44]. When one tries to perform cluster analysis of the data, each $y_k$ can reflect the data distribution on the corresponding cluster. Thus, the $C$ is set to be equal to the number of clusters [2]. For feature selection, we can evaluate features

jointly that align well along each intrinsic dimension (each column of $\mathbf{Y}$), correspondingly, the contribution of each feature for differentiating each cluster. That is, given $y_k$, a column of $\mathbf{Y}$, we propose to find a set of $m$ features, such that their linear span is close to $y_k$. This idea can be formulated as the minimization problem:

$$\min_{\Phi_{\mathcal{M},k},\mathcal{M}} \|y_k - \mathbf{X}_{\mathcal{M}}\Phi_{\mathcal{M},k}\|^2 . \tag{5.11}$$

where $\mathcal{M} = \{i_1, \ldots, i_m\} \subseteq \{1, \ldots, d\}$, $\mathbf{X}_{\mathcal{M}} = (f_{i_1}, \ldots, f_{i_m}) \in R^{N\times m}$ and $\Phi_{\mathcal{M},k}$ is corresponding to a transformation vector that measures the importance of different features in approximating $y_k$. When all $y_k$ are considered, their joint optimization can be formulated as:

$$\underset{\Phi_{\mathcal{M},k},\mathcal{M}}{\operatorname{argmin}} \sum_{k=1}^{C} \|y_k - \mathbf{X}_{\mathcal{M}}\Phi_{\mathcal{M},k}\|^2 = \|\mathbf{Y} - \mathbf{X}_{\mathcal{M}}\Phi_{\mathcal{M}}\|^2 . \tag{5.12}$$

In the above equation, $\Phi_{\mathcal{M}} = [\Phi_{\mathcal{M},1}, \cdots, \Phi_{\mathcal{M},k}, \cdots, \Phi_{\mathcal{M},C}]$. Note, when $\Phi_{\mathcal{M}}$ contains only one feature, the formulation reduces to searching for features that maximize the Equation (5.12).

Given $\mathbf{Y}$ and $\mathbf{X}_{\mathcal{M}}$, $\Phi_{\mathcal{M}}$ can be obtained in a closed form. However, feature selection needs to locate a optimal subset of features $f_{\mathcal{M}}$ that are close to $\mathbf{Y}$. This is a combinatorial problem which is NP-hard [60]. We approximate the problem, as the minimization

$$\begin{aligned} &\min_{\Phi_k,\gamma} \|y_k - \mathbf{X}\Phi_k\|^2 \\ &s.t. |\Phi_k| \leq \gamma \end{aligned} \tag{5.13}$$

where $|\Phi_k|$ is the $\ell_1$-norm which is defined in the following way:

$$|\Phi_k| = \sum_{i=1}^{d} |\Phi_{i,k}|. \tag{5.14}$$

where $\Phi_k$ is a $d$ dimensional vector that contains the combination coefficients required to compute for different features in approximating $y_k$. When applied in regression, the $\ell_1$-norm constraint is equivalent to applying a Laplace prior [46] on $\Phi_k$. This tends to force

many rows of $\Phi$ to be zero, resulting in a sparse solution. Therefore, the representation is generated by using only a small set of selected features.

There are three advantages of the formulation presented in Equation (5.13). First, it can find a set of features that jointly preserve the cluster structure resulting from kECA. Second, it can handle redundant features. Traditional feature selection methods always treat each feature individually which may repeatedly select highly correlated features in the selection process and thus being unable to handle feature redundancy. In our algorithm, by jointly evaluating a set of features, we tend to select non-redundant features. As a result combinations of several "irrelevant" features may create stronger discriminating power. This can improve the global optimality of feature selection. Thirdly and finally, our algorithm is tractable. Given a value for $\gamma$, the solution of Equation (5.13) can be found by applying a general solver [1]. Given the number of selected features $m$, an appropriate $\gamma$ value, which results in the selection of about $m$ features, can be found. This is done by applying either a) a grid search or b) a binary search based on the observation that, a smaller $\gamma$ value usually results in selecting fewer features. However, for a given $m$, this approach may require us to run a solver many times to locate the best value of $\gamma$ value, and this is computationally inefficient.

In order to efficiently solve the optimization problem in Equation (5.13), we use the Least Angle Regression (LARs) algorithm [9]. Instead of setting the parameter $\gamma$, LARs allows to control the sparseness of $\Phi_k$. This is done by specifying the cardinality (the number of non-zero entries) of $\Phi_k$, which is particularly convenient for feature selection.

We consider selecting $m$ features from the $d$ feature candidates. For a data set containing $C$ clusters, we can compute $C$ selection vectors $\{\Phi_k\}_{k=1}^{C} \in R^d$. The cardinality of each $\Phi_k$ is $m$ and each entry in $\Phi_k$ corresponds to a feature. Here, we use the following computationally effective method for selecting exactly $m$ features based on the $C$ selection vectors.

For every feature $j$, we define the KECAR score for the feature as

$$KECAR(j) = \max_{k} |\Phi_{j,k}| \, . \tag{5.15}$$

where $\Phi_{j,k}$ is the $j$-th element of vector $\Phi_k$. We then sort the features in descending order according to their KECAR scores, and then select the top $m$ features. The sequence of steps shown in Algorithm 3 illustrates our method in detail.

**Algorithm 3:** Regression-based Feature Selection Framework Under Kernel ECA

**Input**: Dataset $\mathbf{X}_{N\times d}$

**Output**: $m$ selected features

1: Using kernel entropy component analysis (kernel ECA) for data transformation and dimensionality reduction, we get the top $C$ embedding space $\mathbf{Y} = [y_1, \ldots, y_C]$ with respect to the most entropy estimate, see Equation (5.10);

2: Solve the regression problem in Equation (5.13) using the LARs algorithm with the cardinality constraint set to $m$. We obtain $C$ sparse coefficient vectors $\{\Phi_k\}_{k=1}^{C} \in R^d$;

3: Compute the KECAR score for each feature according to Equation (5.15);

4: Return the top $m$ features according to their KECAR scores.

## 5.4 Feature Evaluation Indices

Our proposed unsupervised feature selection method (referred to as kECA+LARs) utilizes kernel entropy component analysis (kECA) and the Least Angle Regression (LARs) algorithm for unsupervised feature selection. It involves applying kECA to embed the data into another space and then uses LARs to select features that align well to the embedded data resulting from kECA. In order to examine the performance of our proposed

method kECA+LARs, we need to assess the data transformation obtained and its useful information content it. In view of this, we would like to measure the performance of our proposed algorithm using three different indices, namely, (1) **data transformation**, (2) **classification accuracy** and (3) **redundancy rate**. Assume $S$ is the set of selected features, the redundancy rate can be defined as follow:

$$RED(S) = \frac{1}{m(m-1)} \sum_{f_i, f_j \in S, i>j} \rho_{i,j} \ . \tag{5.16}$$

where $\rho_{i,j}$ returns the Pearson correlation between two features $f_i$ and $f_j$. The measurement assesses the averaged correlation among all feature pairs, and a large value indicates that many selected features are strongly correlated and thus redundancy is expected to exist in $S$.

## 5.5 Experiments and Comparisons

### 5.5.1 Data sets

The data sets used to test the performance of our proposed algorithm are publicly available face-recognition benchmarks. Table. 5.1 summarizes the coverage and properties of the three data-sets. In all of experiments performed, face location preprocessing was applied. The original images were normalized (in scale and orientation) such that the two eyes were aligned at the same position. Then, the facial areas were cropped to give images for matching. The size of each cropped image is 32×32 pixels, with 256 grey levels per pixel. Thus, each image is represented by a 1024-dimensional vector. In Fig. 5.1, we show the closely cropped images and these all contain facial structure.

The Yale face dataset contains 165 images of 15 individuals that include variations in facial expression and lighting conditions, together with subjects both with and without glasses. A random subset with 7 images per individual (hence, 105 images in total) was

| Data-set | Examples | Features | Classes |
|---|---|---|---|
| Yale | 165 | 1024 | 15 |
| ORL | 400 | 1024 | 40 |
| PIE | 1428 | 1024 | 68 |

Table 5.1: Summary of benchmark face data sets

taken to form the training set. The remainder of the dataset was used as the test-set. The training samples were used to learn the relevant feature subset. The test samples were then represented by the relevant extracted features.

The ORL face dataset contains 40 distinct individuals with ten images per person. The images are taken at different time instances, and include variations in facial expression and facial detail (glasses/no glasses). A random subset with 6 images per individual (hence, 240 images in total) was used as the training set. The remainder of the dataset was used as the test set.

The PIE is a multiview face dataset, consisting of 41,368 images of 68 people. The views cover a wide range of poses from profile to frontal views, varying illumination and expression. In this experiment, we fixed the pose and expression. Thus, for each person, we have 21 images obtained under different lighting conditions, We got 14 of these images for training and the remaining 7 for testing.

### 5.5.2 Data Transformation

we compare the data transformation performance of our proposed method using kECA with alternative methods, including kernel PCA [10], the Laplacian eigenmap [44] and LPP [74]. In order to visualize the results, we have used five randomly selected subjects from each dataset, and these are shown in Fig. 5.2, Fig. 5.3 and Fig. 5.4. In each figure, we have shown the projections onto the leading two most significant eigenmodes from

(a) Yale dataset

(b) ORL dataset

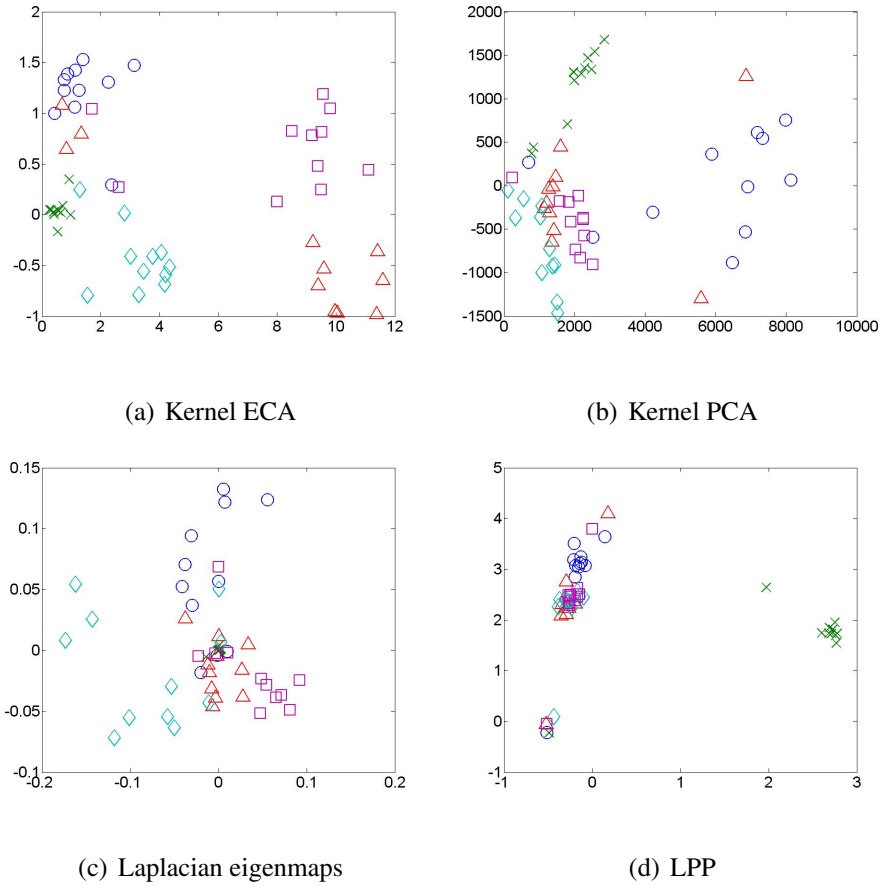
(c) PIE dataset

Figure 5.1: The sample cropped face images of two individual form three face dataset.

the kernel PCA, Laplacian eigenmaps and LPP respectively, ordered according to their eigenvalues. This provides a low-dimensional representation for the images. We also have shown the projections onto the leading two principal components extracted using kernel ECA. From the above figures, it is clear that the kernel ECA based clustering of the face samples represents a significant improvement over that obtained using the alternative data transformation methods. This implies that the entropy based principal component vectors selection is more appropriate than selecting principal component vectors based only on magnitude of eigenvalues.

### 5.5.3 Classification Accuracy

In order to explore the discriminative capabilities of the information captured by our method, we use the selected features for further classification. We compare the classification results from our proposed method kECA+LARs with six representative feature



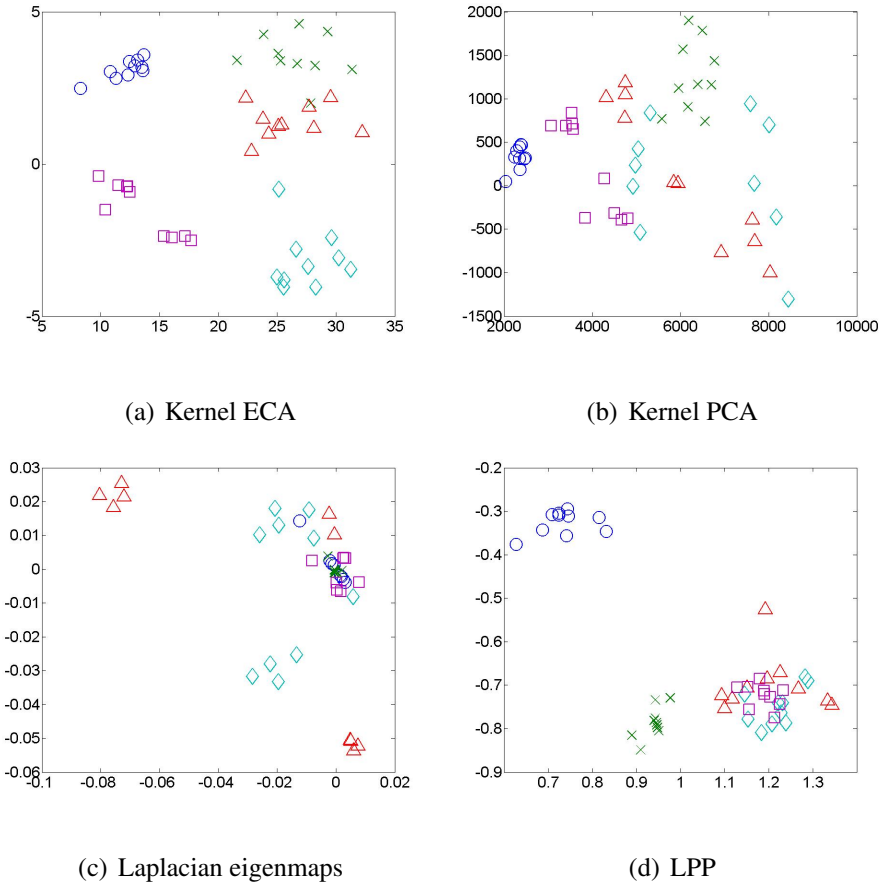
(a) Kernel ECA (b) Kernel PCA

(c) Laplacian eigenmaps (d) LPP

Figure 5.2: Distribution of samples of five subjects in Yale dataset.

selection algorithms. For unsupervised learning, four alternative feature selection algorithms are selected as baselines. These methods are the Laplacian score [73], SPEC [81], MCFS [16] and UDFS [77]. We also compare our obtained results with two supervised feature selection methods, namely a) the Fisher score [24] and b) the MRMR algorithm [27]. We use 5-fold cross-validation for the SVM classifier on the feature subsets obtained by the feature selection algorithms to verify their classification performance. Here we use the linear SVM with LIBSVM [14].

The classification accuracies obtained with different feature subsets are shown in Fig. 5.5. From the figure, it is clear that our proposed method kECA+LARs is, by and




(a) Kernel ECA

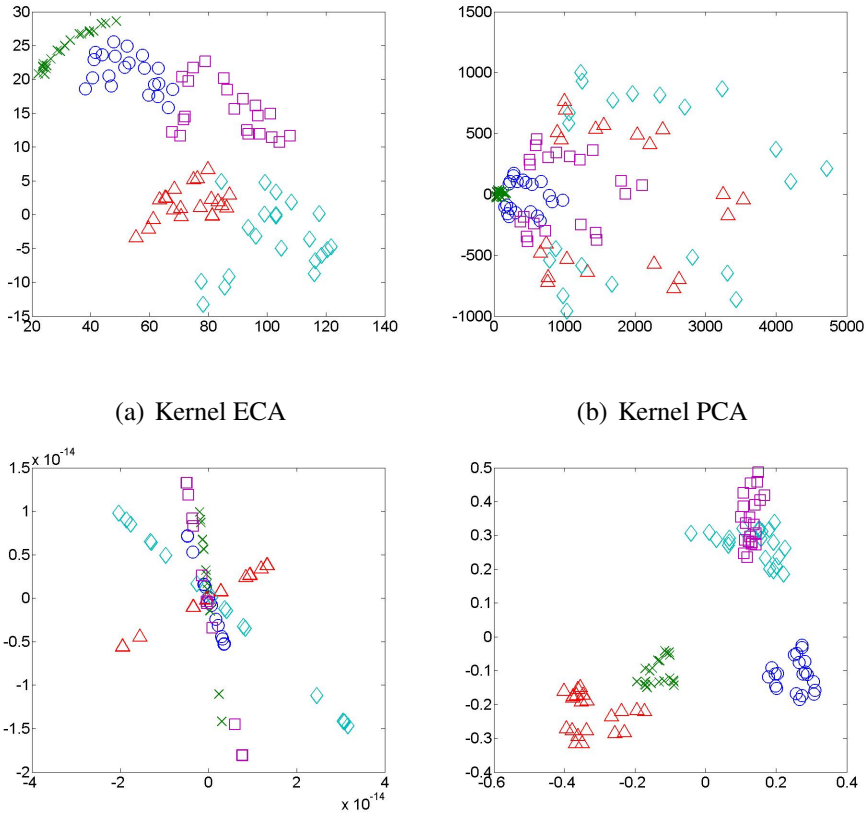
(b) Kernel PCA

(c) Laplacian eigenmaps

(d) LPP

Figure 5.3: Distribution of samples of five subjects in ORL dataset.

large, superior to the alternative unsupervised feature selection methods. Specifically, it selects a both a smaller and better performing (in terms of classification accuracy) set of discriminative features on all the three data sets. Moreover, kECA+LARs rapidly converges to the best results, with typically around 80 features. Each of the alternative unsupervised methods, usually require more than 100 features to achieve a comparable result. There are two reasons for this improvement in performance. First, the kernel-based methodology is integrated together with entropy-based analysis to select the best principal component vectors. Thus both structural and the entropy (complexity-based) information latent in the data can be effectively preserved. Second, the LARs algorithm is applied to



(a) Kernel ECA (b) Kernel PCA

(c) Laplacian eigenmaps (d) LPP

Figure 5.4: Distribution of samples of five subjects in PIE dataset.

select features that align well to the embedded data resulting from kECA. As a result the optimal feature combinations can be located so as to remove redundant features.

Compared with the supervised feature selection algorithms, kECA+LARs outperforms the Fisher score algorithm in all cases. On the Yale dataset, even though MRMR gives the best classification performance of 86.67% with 100 features, kECA+LARs achieves a comparable result with a much smaller number of features, i.e., only 86 features. This implies that our proposed method is able to locate both the optimal size of the feature subset and perform accurate classification of the samples based on just a few of the most important features.

| Dataset | Yale | ORL | PIE |
|---|---|---|---|
| MRMR | $\mathbf{86.67\%(100)}$ | 83.5%(95) | 99.15%(99) |
| Fisher score | 75.15%(92) | 80%(99) | 99.37%(97) |
| Laplacian score | 65.45%(100) | 65.25%(99) | 71.43%(99) |
| SPEC | 70.91%(100) | 64.5%(95) | 89.64%(100) |
| UDFS | 66.06%(99) | 76.5%(99) | 96%(98) |
| MCFS | 83.64%(86) | 87.75%(88) | 99.85%(98) |
| kECA+LARs | 84.24%(86) | $\mathbf{93\%(74)}$ | $\mathbf{100\%(84)}$ |

Table 5.2: The best result of all methods and their corresponding size of selected feature subset on the three face datasets

It is interesting to note that on the ORL dataset, the classification results obtained by the unsupervised spectral regression based methods (kECA+LARs and MCFS) are even better than those obtained using the supervised feature selection methods (Fisher score and MRMR). Our proposed kECA+LARs algorithm achieves the best classification accuracy, which is higher than that obtained using the alternative supervised and unsupervised feature selection algorithms. This implies that spectral regression based methods are a better way to analyze data features jointly. Moreover, kECA+LARs performs better than the MCFS algorithm, although they are both belong to the spectral regression framework. The reason is that the MCFS algorithm uses the Lapalacian eigenmap [44] to reveal the data manifold structure. On one hand, the Laplacian eigenmap performs dimensionality reduction by selecting $l$ eigenvalues (spectrum) and eigenvectors solely based on the magnitude of the eigenvalues, and the resulting transformation may be based on uninformative eigenvectors from an entropy perspective which can lead to substantial loss of information. On the other hand, the Laplacian eigenmap only provides an approximation solution of the ratio cut clustering [52], which cannot guarantee have the clear cluster

structure, hence usually further clustering algorithm such as K-means need to be applied to obtain the final clustering result [25]. Our proposed kECA+LARs algorithm is fundamentally different from other spectral methods, where the kernel-based methodology is combined with entropy analysis for data transformation. Thus it is capable of revealing more "obvious" or accurate data structure.

The best result for each method together with the corresponding size of the selected feature subset are shown in Table. 5.2. In the table, the classification accuracy is shown first and the optimal number of features selected is reported in brackets. Overall, our proposed method kECA+LARs achieves the highest degree of dimensionality reduction, i.e. it selects a smaller feature subset compared with those obtained by the alternative methods. For example, in the ORL data set, the best result obtained by the alternative feature selection methods is 87.75% with the MCFS algorithm and 88 features. However, our proposed method kECA+LARs gives a better accuracy 93% when only 74 features are used. The results further verify that our feature selection method can guarantee the optimal size of feature subset, as it not only achieves a higher degree of dimensionality reduction but it also gives better discriminability. We also observe that the UDFS algorithm gives a better result than the alternative unsupervised methods (i.e. the Laplacian score and the SPEC). The reason for this is that unlike traditional methods which treat each feature individually and which hence are suboptimal, the UDFS method directly optimizes the score over the entire selected feature subset. As a result, a better feature subset can be obtained. On the other hand, the alternative unsupervised feature selection method, i.e. the Laplacian score and SPEC, fail to locate the most discriminative features. This may be explained by our observation that they are unable to handle feature redundancy. For instance, they may repeatedly select highly correlated features in the selection process. And It has been known that redundant features can adversely affect the performance of classification and clustering.

| Dataset | Yale | ORL | PIE |
|---|---|---|---|
| MRMR | **0.58** | 1.47 | 0.51 |
| Fisher score | 0.62 | 1.72 | 0.53 |
| Laplacian score | 0.73 | 1.68 | 0.67 |
| SPEC | 0.72 | 1.65 | 0.66 |
| UDFS | 0.71 | 1.62 | 0.63 |
| MCFS | 0.66 | 1.52 | 0.64 |
| kECA+LARs | 0.65 | **1.37** | **0.49** |

Table 5.3: Averaged Redundancy Rate of Subsets Selected using Different Algorithms

### 5.5.4 Redundancy Rate

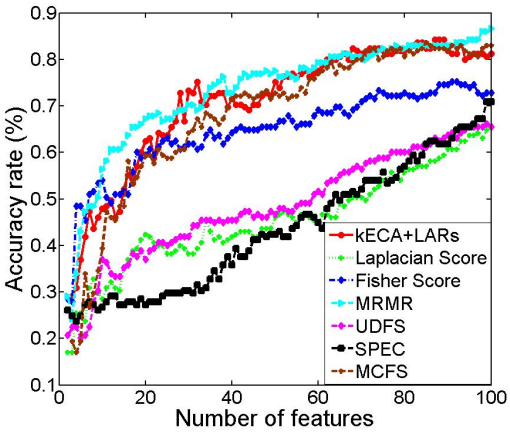
Table. 5.3 shows the comparative results of our proposed method with the alternative feature selection methods using the top $n$ features, where $n$ is the instance number of the training data. We chose $n$, since when the number of selected features is larger than $n$, any feature can be expressed by a linear combination of the remaining ones, which will introduce unnecessary redundancy in the evaluation stage. In the table, the boldfaced values are the lowest redundancy rates. The subset obtained by our proposed scheme has the least redundancy in two of the three datasets. This further verifier that our propose algorithm is able to remove redundant features.

The results from the accuracy rate in Table. 5.2 and redundancy rate in Table. 5.3 together indicate that kECA+LARs both contains the least redundancy, and result in highest accuracy. They also underline necessity of removing redundant features for improving learning performance. It should also be observed that the MRMR algorithm also produces low redundancy rates. However, it does not perform as well in the terms of classification accuracy. This can be explained by the observation that: in MRMR, feature contributions to classification is considered individually by evaluating the correlation between each fea-

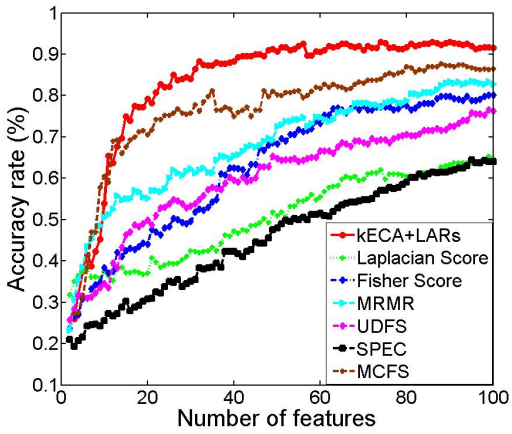
ture and the class label. However, the class label may be jointly determined by a set of features. This interaction among features is not considered by MRMR.

## 5.6 Conclusion

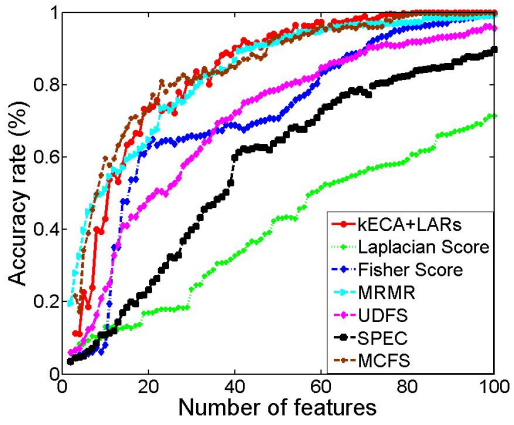
In this chapter, we have presented a new spectral regression technique for unsupervised feature selection. The proposed feature selection method offers two major advantages. First, a kernel-based methodology is combined with entropy analysis for data transformation, which reflects the cluster structure of the data. It is fundamentally different from most existing spectral methods, which are only based on the magnitude of eigenvalues of specially constructed data matrices. Second, using $\ell_1$-norm regularization, casts the feature discriminant analysis into a regression framework which considers the correlations among features. Therefore, we can evaluate features jointly rather than individually. Thus the method is able to handle redundant features.




(a) Yale dataset




(b) ORL dataset




(c) PIE dataset

Figure 5.5: Accuracy rate vs. the number of selected features on three face dataset.

# Chapter 6

# Hypergraph Spectral Analysis for Unsupervised Feature Selection

In this chapter, by incorporating multidimensional interaction information (MII) for higher order similarities measure, we establish a novel hypergraph framework which is used for characterizing the multiple relationships within a set of samples (e.g. face samples under varying illumination conditions). Thus, the structural information latent in the data can be more effectively modeled. Then an unsupervised method is proposed to find the discriminating feature subset on the basis of hypergraph representation. For the unsupervised learning, we derive a hypergraph embedding view of feature selection, where the projection matrix is constrained to be a selection matrix designed to select the optimal feature subset. Experimental results demonstrate the effectiveness of our feature selection methods on a number of standard image datasets.

### Contribution

We establish a novel hypergraph framework which can be more effective capture the high-order relations among samples rather than approximating them in terms of pairwise interactions can lead to substantia loss of information. Specifically, we construct a hypergraph in which each node corresponds to a sample, and each hyperedge has a weight corresponding to the multidimensional interaction information (MII) among samples connected by

that hyperedge. The advantage of MII is that it is sensitive to the relations between sample combinations, and as a result can be used to seek third or even higher order dependencies between the relevant samples. Thus, the structural information latent in the data can be more effectively modeled. Different from representing the hypergraph by the clique expansion or the star expansion in traditional hypergraph based learning methods, we employ a more effective matrix representation for hypergraphs. With this representation the low-pass information loss in the process of averaging hypergraph weights can be overcome.

On the basis of hypergraph representation, we explore the discriminating features in an unsupervised way. Specifically, we describe a new feature selection strategy through hypergraph embedding, which casts the feature discriminant analysis into a regression framework that considers the correlations among features. As a result, we can evaluate joint feature combinations, rather than being confined to consider them individually, thus it is able to handle feature redundancy. Experimental results demonstrate the effectiveness of our unsupervised feature selection method on a number of standard image data-sets.

**Chapter outline**

The remainder of this chapter is organized as follows. Section 6.1 describes how to construct the hypergraph. Section 6.2 presents how to approximate the hypergraph. The unsupervised feature selection method and its experimental results on a number of standard image data-sets are respectively presented in Section 6.3 and Section 6.4. Finally, conclusions are presented in Section 6.5.

## 6.1 Hypergraph Construction

In this section, we establish a novel hypergraph framework which is used for characterizing the multiple relationships within a set of samples. To this end, we commence by

applying a new method for measuring higher order similarities among samples based on multidimensional interaction information. The generalization of Interaction Information to $K$ variables is defined recursively as follow

$$I(\{X_1, \cdots, X_K\}) = I(\{X_2, \cdots, X_K\}|X_1) - I(\{X_2, \cdots, X_K\}) \ . \tag{6.1}$$

Based on the higher order similarity measure, we establish a hypergraph framework for characterizing a set of high dimensional samples. A hypergraph is defined as a triplet $H = (V, E, \mathbf{W})$. Here $V$ denotes the vertex set, $E$ denotes the hyperedge set in which each hyperedge $e \in E$ represents a subset of $V$, and $\mathbf{W}$ is a weight function which assigns a real value $\mathbf{W}(e)$ to each hyperedge $e \in E$. We only consider $K$-uniform hypergraphs (i.e. those for which the hyperedges have identical cardinality $K$) in our work. Given a set of high dimensional samples $\mathbf{X} = [x_1, \cdots x_N]^T$ where $x_i \in \mathbb{R}^d$, we establish a $K$-uniform hypergraph, with each hypergraph vertex representing an individual sample and each hyperedge representing the $K$-th order relations among a $K$-tuple of participating samples. A $K$-uniform hypergraph can be represented in terms of $K$-th order matrix, i.e. a tensor $\mathcal{W}$ of order $K$, whose element $\mathbf{W}_{i_1,\cdots,i_K}$ is the hyperedge weight associated with the $K$-tuple of participating vertices $\{v_{i_1}, \cdots, v_{i_K}\}$. In our work, the hyperedge weight associating with $\{x_{i_1}, x_{i_2}, \cdots, x_{i_K}\}$ is computed as follows

$$\mathbf{W}_{i_1,\cdots,i_K} = K \frac{I(x_{i_1}, x_{i_2}, \cdots, x_{i_K})}{H(x_{i_1}) + H(x_{i_2}) + \cdots H(x_{i_K})} \ . \tag{6.2}$$

It is clear that $\mathbf{W}_{i_1,\cdots,i_K}$ is a normalized version of $K$-th order Interaction Information. The greater the value of $\mathbf{W}_{i_1,\cdots,i_K}$ is, the more relevant the $K$ samples are. On the other hand, if $\mathbf{W}_{i_1,\cdots,i_K} = 0$, the $K$ samples are totally unrelated.

## 6.2 Hypergraph Representation

Unlike matrix eigen-decomposition, there has not yet been a widely accepted method for spanning a rational eigen-space for a tensor [68]. Therefore, it is hard to directly embed

a hypergraph into a feature space spanned by its tensor representation through eigen-decomposition. In our work, we consider the transformation of a $K$-uniform hypergraph into a graph. Accordingly, the associated hypergraph tensor $\mathcal{W}$ is transformed to a graph adjacency matrix $\boldsymbol{A}$, and the higher order information exhibited in the original hypergraph can be encoded in an embedding space spanned by the related matrix representation. In this scenario, one straightforward way for the transformation is marginalization which computes the arithmetical average over all the hyperedge weights $\mathbf{W}_{i_1,\cdots,i_{K-2},i,j}$ associated with the edge weight $A_{i,j}$

$$\tilde{A}_{i,j} = \sum_{i_1=1}^{|V|} \cdots \sum_{i_{K-2}=1}^{|V|} \mathbf{W}_{i_1,\cdots,i_{K-2},i,j} \tag{6.3}$$

The edge weight $\tilde{A}_{i,j}$ for edge $ij$ is generated by a uniformly weighted sum of hyperedge weights $\mathbf{W}_{i_1,\cdots,i_{K-2},i,j}$. However, the form appearing in (6.3) behaves as a low pass filter, and thus results in information loss through marginalization.

To make the process of marginalization more comprehensive, we use marginalization to constrain the sum of edge weights and then estimate their values through solving an over-constrained system of linear equations. Our idea is motivated by the so called *clique average* introduced in the higher order clustering literature [61]. We characterize the relationships between $\boldsymbol{A}$ and $\mathcal{W}$ as follows

$$\mathbf{W}_{i_1,\cdots,i_K} = \sum_{\{i,j\}\subseteq\{i_1,\cdots,i_K\}} A_{i,j} \tag{6.4}$$

Fig. 6.1 is an example illustrating the relationship between a 3-uniform hypergraph and its graph representation resulted from (6.4). The cube on the right illustrates the tensor $\mathcal{W}$ for the 3-uniform hypergraph, and the square on the left illustrates the adjacency matrix $\boldsymbol{A}$ for the graph representation. Here $i_1$, $i_2$ and $i_3$ denote the indices of boldly selected entries of $\mathcal{W}$ and $\boldsymbol{A}$. It is clear that the hyperedge weight $\mathbf{W}_{i_1,i_2,i_3}$ is the sum of the involved graph edge weights $A_{i_1,i_2}$, $A_{i_1,i_3}$ and $A_{i_2,i_3}$.

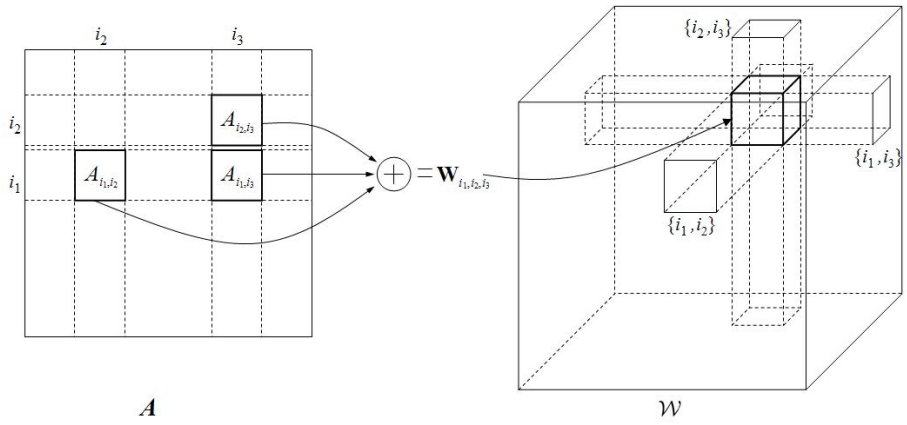



Figure 6.1: An example for hypergraph representation.

There are $\binom{|V|}{2}$ variables and $\binom{|V|}{K}$ equations in the system of equations described in (6.3). When $K > 2$, the linear system (6.3) is over-determined and cannot be solved analytically. We thus approximate the solution to (6.3) by minimizing the least squares error

$$\widehat{\boldsymbol{A}} = \underset{\boldsymbol{A}}{\operatorname{argmin}} \sum_{i_1,\cdots,i_K} \left( \sum_{\{i,j\}\subseteq\{i_1,\cdots,i_K\}} A_{i,j} - \mathbf{W}_{i_1,\cdots,i_K} \right)^2 \tag{6.5}$$

In practical computation, we normalize the compatibility tensor $\mathcal{W}$ by using the extended Sinkhorn normalization scheme [4], and constrain the element of $\boldsymbol{A}$ to be in the interval $[0, 1]$ to avoid unexpected infinities. The effective iterative numerical method based on Gram-Schmidt decompositions is used to compute the approximated solutions [5].

The adjacency matrix $\boldsymbol{A}$ computed through (6.5) is one effective representation for a $K$-uniform hypergraph, because it naturally avoids the operation of arithmetic average and thus to a certain degree overcomes the low pass information loss arising in (6.3). Furthermore, the Laplacian matrix $\boldsymbol{L}$ for a hypergraph can be defined as $\boldsymbol{L} = \boldsymbol{D} - \boldsymbol{A}$, where $\boldsymbol{D}$ is the diagonal matrix with its $i$-th diagonal element being $A_{ii} = \sum_j A_{ij}$. In

this context, a hypergraph can be easily embedded into a feature space spanned by its Laplacian matrix, which will be explained in detail in the next section of this chapter.

## 6.3 Unsupervised Feature Selection through Hypergraph Embedding

In this section, we formulate the procedure of feature extraction on a basis of hypergraph spectral embedding. One goal of spectral embedding is to represent the high dimensional data $\mathbf{X} \in \mathbb{R}^{N\times d}$ by a low dimensional representation $\mathbf{Y} \in \mathbb{R}^{N\times C}$ $(C \ll d)$ in the low dimensional feature space such that the structural characteristics of the high dimensional data are well preserved or even emphasized. Here we use the representations $\mathbf{X} = [x_1, \cdots x_N]^T$ and $\mathbf{Y} = [y_1, \cdots, y_k, \cdots, y_C]$, where $y_k$ is a $N$-dimensional vector and its $N$ elements represent the $N$ samples $x_1, \cdots x_N$ separately in the $k$-th dimension of the low dimensional feature space [69].

Based on the hypergraph transformation described in Section 6.2 and using Laplacian eigen-decomposition [44], the hypergraph spectral embedding can be formulated as

$$\boldsymbol{D}^{-1}\boldsymbol{L}\mathbf{Y} = \lambda\mathbf{Y} \ . \tag{6.6}$$

where $\boldsymbol{D}$ is diagonal matrix with its entries are column (or row, since $\boldsymbol{A}$ is symmetric) sums of $\boldsymbol{A}$. Let $y_0, y_1, \ldots, y_C$ be the eigenvector solutions of Equation (6.6), ordered according to their eigenvalues ($0 = \lambda_0 \leq \lambda_1 \leq \cdots \leq \lambda_C$). We leave out the eigenvector $y_0$ corresponding to eigenvalue 0 and use the next $C$ eigenvectors for embedding in $C$-dimensional Euclidean space. For example, for the data point $x_i$, its embedding space can be represented as follow

$$x_i \longrightarrow (y_1(i), \ldots, y_C(i)) \ . \tag{6.7}$$

The hypergraph embedding procedure can be viewed as feature extraction, and can be expressed as $\mathbf{Y} = \mathbf{X}\Phi$ where $\Phi \in \mathbb{R}^{d \times C}$ is a column-full-rank projection matrix. However, unlike feature extraction, feature selection attempts to select the optimal feature subset in the original feature space. Therefore, for the task of feature selection, the projection matrix $\Phi = [\Phi_1, \ldots \Phi_C]$ can be constrained to be a selection matrix which contains the combination coefficients for different features in approximating $\mathbf{Y} = [y_1, \ldots, y_C]$. That is, given the $k$-th column of $\mathbf{Y}$, i.e $y_k$, we aim to find a subset of features, such that their linear span is close to $y_k$. This idea can be formulated as the minimization problem

$$\widehat{\Phi} = \underset{\Phi}{\operatorname{argmin}} \sum_{k=1}^{C} \|y_k - \mathbf{X}\Phi_k\|^2 . \tag{6.8}$$

where $\Phi = [\Phi_1, \cdots, \Phi_k, \cdots, \Phi_C]$ and $\Phi_k$ is a $d$-dimensional vector that containing the combination coefficients required to compute for different features in approximating $y_k$. However, feature selection requires us to locate an optimal subset of features that are close to $y_k$. This is a combinatorial problem which is NP-hard. Thus we approximate the problem in (6.8) subject to the constraint

$$|\Phi_k| \leq \gamma \tag{6.9}$$

where $|\Phi_k|$ is the $\ell_1$-norm and $|\Phi_k| = \sum_{j=1}^{d} |\Phi_{j,k}|$. When applied in regression, the $\ell_1$-norm constraint is equivalent to applying a Laplace prior [46] on $\Phi_k$. This tends to force some entries in $\Phi_k$ to be zero, resulting in a sparse solution. Therefore, the representation $\mathbf{Y}$ is generated by using only a small set of selected features in $\mathbf{X}$.

In order to efficiently solve the optimization problem in Equations (6.8) and (6.9), we use the Least Angle Regression (LARs) algorithm [9]. Instead of setting the parameter $\gamma$, LARs allows us to control the sparseness of $\Phi_k$. This is done by specifying the cardinality of the number of nonzero subsets of $\Phi_k$, which is particularly convenient for feature selection.

We consider selecting $m$ features from the $d$ feature candidates. For a dataset containing $C$ clusters, we can compute $C$ selection vectors $\{\Phi_k\}_{k=1}^{C} \in R^d$. The cardinality

of each $\Phi_k$ is $m$ and each entry in $\Phi_k$ corresponds to a feature. Here, we use the following computationally effective method for selecting exactly $m$ features based on the $C$ selection vectors. For every feature $j$, we define the $HG$ score for the feature as

$$HG(j) = \max_k |\Phi_{j,k}| \,. \tag{6.10}$$

where $\Phi_{j,k}$ is the $j$-th element of vector $\Phi_k$. We then sort the features in descending order according to their $HG$ scores, and then select the top $m$ features. The sequence of steps shown in Algorithm 4 illustrates our method in detail.

**Algorithm 4:** Unsupervised Feature Selection through Hypergraph Embedding

**Input**: Dataset $\mathbf{X}_{N\times d}$

**Output**: $m$ selected features

1: Constructing a hypergraph representation of the data where weights are computed by Equation (6.2), that takes into higher order interactions as opposed to only pairwise ones, measuring how related multiple samples are.

2. Converting the hypergraph representation into an adjacency matrix $\boldsymbol{A}$ through Equation (6.5)

3: Using hypergraph spectral learning for data transformation and dimensionality reduction, we get the top $C$ embedding space $\mathbf{Y} = [y_1, \ldots, y_C]$ with respect to the eigenvectors of Equation (6.6) ;

4: Solve the regression problem in Equation (6.9) using the LARs algorithm with the cardinality constraint set to $m$. We obtain $C$ sparse coefficient vectors $\{\Phi_k\}_{k=1}^{C} \in R^d$;

5: Compute the $HG$ score for each feature according to Equation (6.10) ;

6: Return the top $m$ features according to their $HG$ scores.

## 6.4 Experiments and Comparisons

### 6.4.1 Data sets

(a) ORL dataset

(b) CMU PIE dataset

(c) MPEG-7 dataset

(d) USPS dataset

(e) MNIST dataset

Figure 6.2: The sample of cropped face images and other three benchmark image datasets.

We test the performance of our proposed algorithm on two publicly available face database (ORL, CMU PIE), one shape image database (MPEG-7), and two handwritten

digit database (USPS, MNIST). Table. 6.1 summarizes the coverage and properties of the five benchmark data-sets. For the face dataset, face location preprocessing was applied. The original images were normalized (in scale and orientation) such that the two eyes were aligned at the same position. Then, the facial areas were cropped to give images for matching. The size of each cropped image is $32 \times 32$ pixels, with 256 grey levels per pixel. Thus, each image is represented by a 1024-dimensional vector. In Fig. 6.2, we show the closely cropped face images and other three benchmark image samples.

| Data-set | Examples | Features | Classes |
|---|---|---|---|
| ORL | 400 | 1024 | 40 |
| CMU PIE | 1428 | 1024 | 68 |
| MPEG-7 | 1400 | 6000 | 70 |
| USPS | 9298 | 256 | 10 |
| MNIST | 4000 | 784 | 10 |

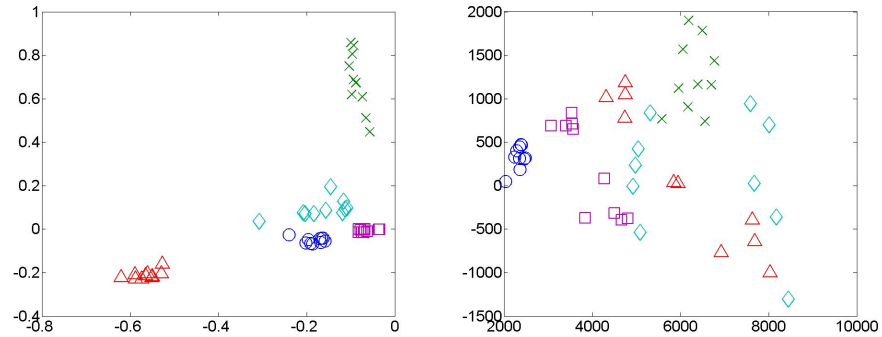
Table 6.1: Summary of benchmark data sets

**ORL dataset:** it contains 40 distinct individuals with ten images per person. The images are taken at different time instances, and include variations in facial expression and facial detail (glasses/no glasses).

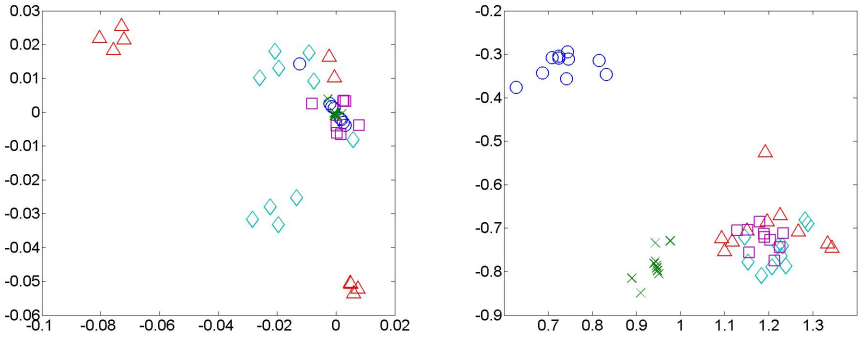
**CMU PIE dataset:** it is a multiview face dataset, consisting of 41,368 images of 68 people. The views cover a wide range of poses from profile to frontal views, varying illumination and expression. In this experiment, we fixed the pose and expression. Thus, for each person, we have 21 images obtained under different lighting conditions.

**MPEG-7 dataset:** it consists of 1,400 silhouette images grouped into 70 classes. Each class has 20 different shapes.

**USPS dataset:** this handwritten digits database contains 9,298 images of handwritten digits. The digits 0 to 9 have 1553, 1269, 929, 824, 852, 716, 834, 792, 708, and 821




(a) Hypergraph Embedding (b) Kernel PCA

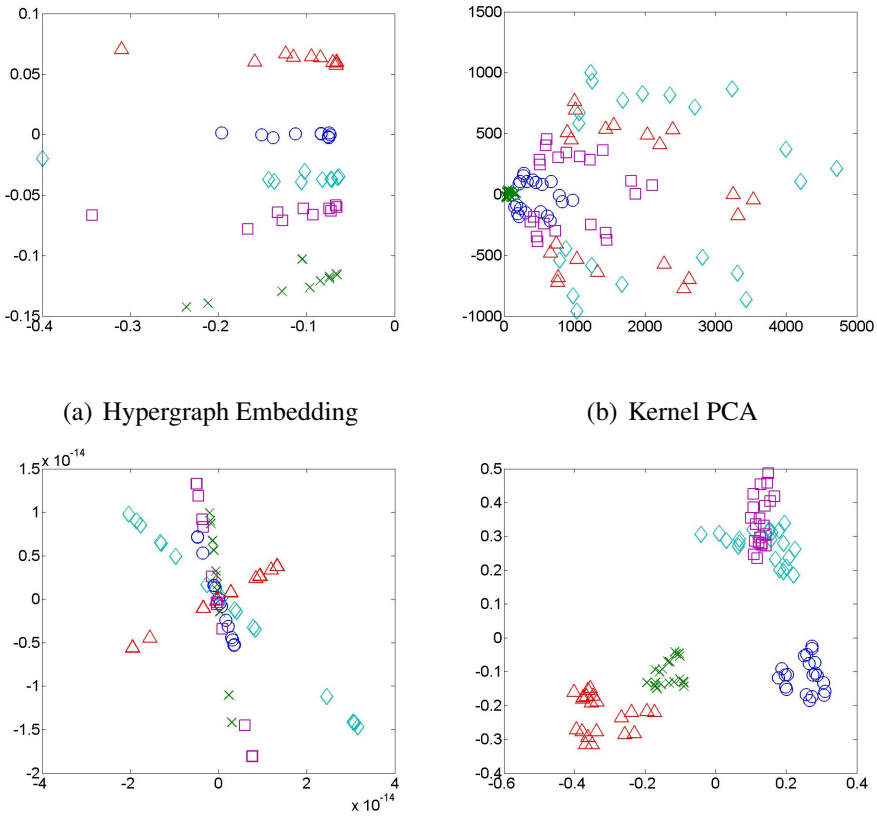



(c) Laplacian eigenmaps (d) LPP

Figure 6.3: Distribution of samples of five subjects in ORL dataset.

samples respectively. The USPS digits data were gathered at the Center of Excellence in Document Analysis and Recognition (CEDAR) at SUNY Buffalo, as part of a project sponsored by the US postal Service. The size of each image is 16×16 pixels, with 256 grey levels per pixel. Thus, each image is represented by a 256-dimensional vector.

**MNIST dataset:** this handwritten digit database has a training set of 60,000 samples (denoted as set A) and a test set of 10,000 samples (denoted as set B). In our experiment, we take the first 2,000 samples from set A as our training set and the first 2,000 samples from set B as our test set. Each digit image is of size 28×28, there are around 200 samples of each digit in both the training and test sets.




(a) Hypergraph Embedding

(b) Kernel PCA

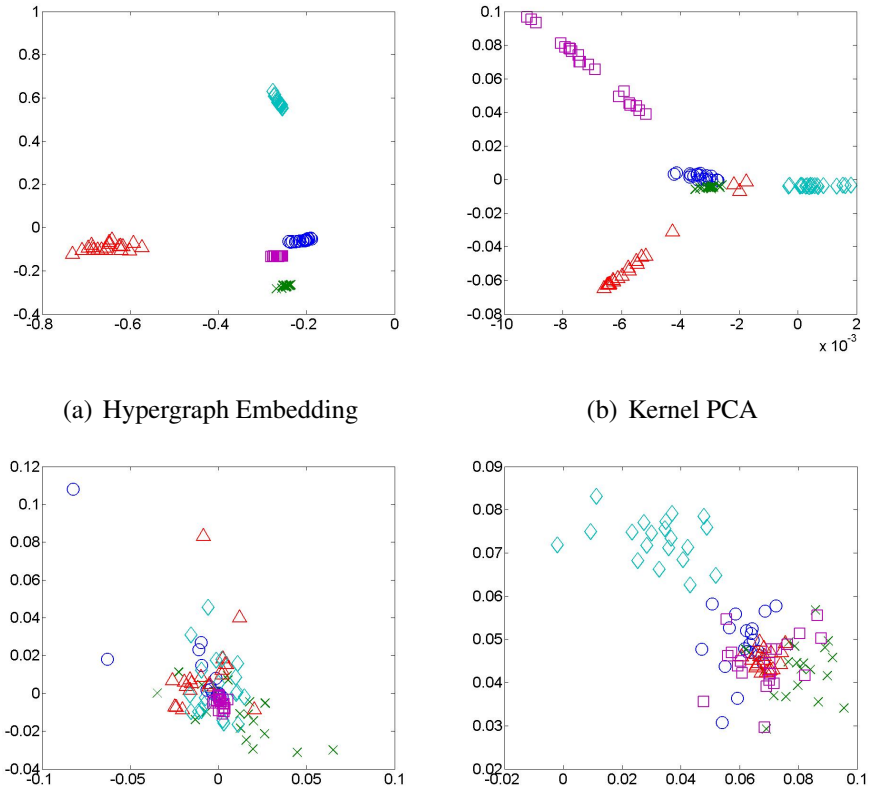
(c) Laplacian eigenmaps

(d) LPP

Figure 6.4: Distribution of samples of five subjects in CMU PIE dataset.

### 6.4.2 Data Transformation

We compare the data transformation performance of our proposed method using hypergraph spectral learning with alternative methods, including kernel PCA [10], the Laplacian eigenmap [44] and LPP [74]. In order to visualize the results, we have used five randomly selected subjects from each dataset, and these are shown from Fig. 6.3 to Fig. 6.7. In each figure, we have shown the projections onto the leading two most significant eigenmodes from different spectral embedding methods, ordered according to their eigenvalues. This provides a low-dimensional representation for the images. From the above figures, it is clear that our hypergraph spectral embedding method demonstrates much



(a) Hypergraph Embedding

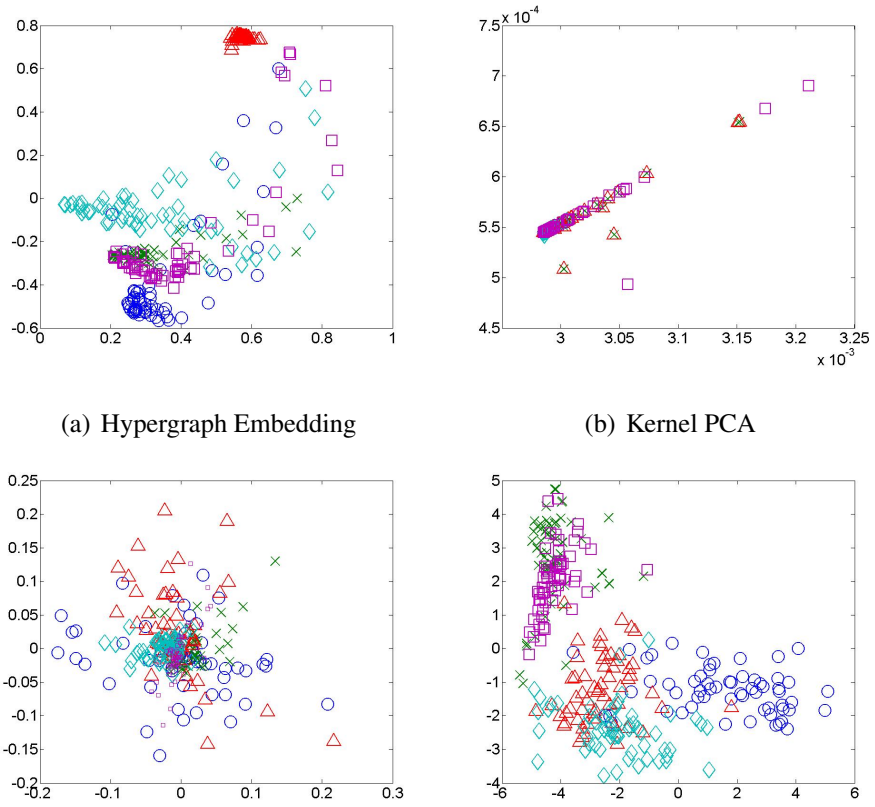
(b) Kernel PCA

(c) Laplacian eigenmaps

(d) LPP

Figure 6.5: Distribution of samples of five subjects in MPEG-7 dataset.

clearer cluster structure than that by traditional spectral clustering method. Therefore, our method can perform clustering task directly by using the embedded result, while traditional spectral clustering (e.g., Laplacian eigenmap) need additional clustering algorithm (e.g., k-means) on the embedded result to obtain the final clustering result. This implies that the hypergraph representation is more appropriated and completeness in describing feature relations and structures.



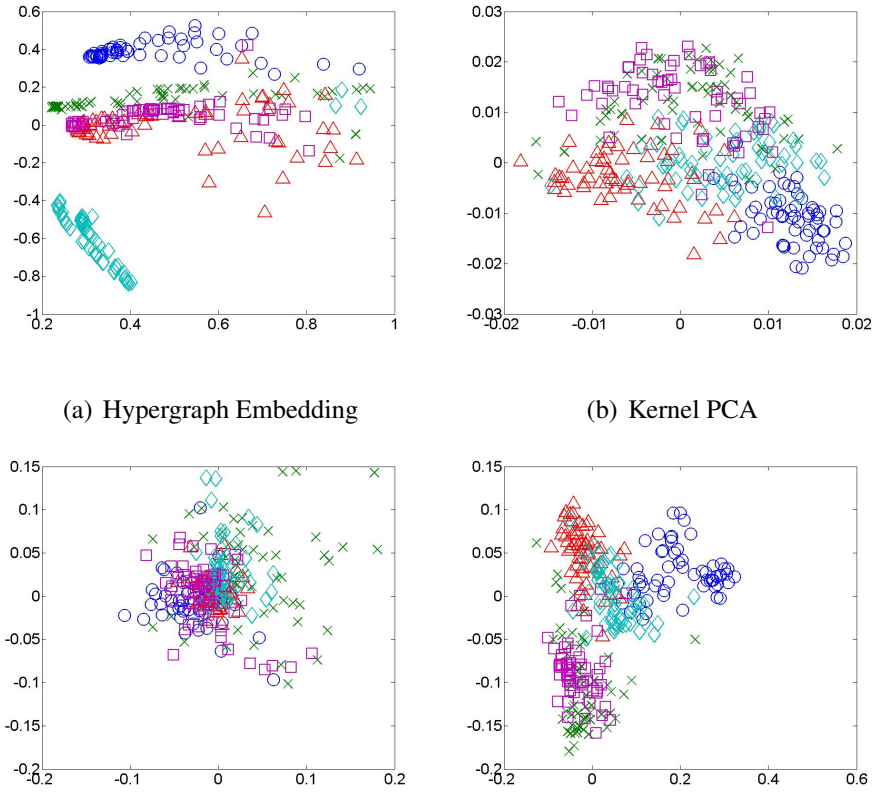
(a) Hypergraph Embedding (b) Kernel PCA

(c) Laplacian eigenmaps (d) LPP

Figure 6.6: Distribution of samples of five subjects in USPS dataset.

### 6.4.3 Classification Accuracy

In order to explore the discriminative capabilities of the information captured by our method, we use the selected features for further classification. We compare the classification results from our proposed method HG+LARs with five representative feature selection algorithms. For unsupervised learning, three alternative feature selection algorithms are selected as baselines. These methods are the Laplacian score [73], SPEC [81] and UDFS [77]. We also compare our obtained results with two state-of-art supervised feature selection methods, namely a) the Fisher score [24] and b) the MRMR algorithm [27]. We use 5-fold cross-validation for the SVM classifier on the feature subsets obtained



(a) Hypergraph Embedding (b) Kernel PCA

(c) Laplacian eigenmaps (d) LPP

Figure 6.7: Distribution of samples of five subjects in MNIST dataset.

by the feature selection algorithms to verify their classification performance. Here we use the linear SVM with LIBSVM [14].

The classification accuracies obtained with different feature subsets are shown in Fig. 6.8. From the figure, it is clear that our proposed method HG+LARs is, by and large, superior to the alternative feature selection methods. Specifically, it selects a both a smaller and better performing (in terms of classification accuracy) set of discriminative features on all the five data sets. Moreover, HG+LARs rapidly converges to the good results, with typically around 30 features (see Fig. 6.8(a),(b),(d) and (e)). Each of the alternative unsupervised methods, usually require more than 100 features to achieve a comparable result. There are two reasons for this improvement in performance. First,

| Dataset | ORL | CMU PIE | MPEG-7 | USPS | MNIST |
|---|---|---|---|---|---|
| MRMR | 83.5%(95) | 99.15%(99) | 80.83%(194) | 95.75%(143) | 82.5%(284) |
| Fisher Score | 80%(99) | 99.37%(100) | 77.83%(200) | 95.8%(103) | 81.25%(293) |
| Laplacian Score | 65.25%(99) | 71.43%(99) | 76.5%(198) | 94.05%(165) | 82.05%(291) |
| SPEC | 64.5%(95) | 89.64%(100) | 63.67%(200) | 94.2%(198) | 82.1%(292) |
| UDFS | 76.5%(99) | 96%(98) | 75.17%(190) | 95.65%(161) | 81.3%(293) |
| HG+LARs | **91%(75)** | **100%(70)** | **82.33%(151)** | **98.8%(59)** | **84.33%(90)** |

Table 6.2: The best result of all methods and their corresponding size of selected feature subset on five benchmark image datasets.

the hypergraph representation is effective in capturing the high-order relations among samples rather than approximating them in terms of pairwise interactions can lead to substantial loss of information. Thus the structural information latent in the data can be effectively preserved. Second, the LARs algorithm is applied to select features that align well to the embedded data resulting from hypergraph spectral embedding. As a result the optimal feature combinations can be located so as to remove redundant features.

Compared with two the-state-of-art supervised feature selection algorithms, our proposed unsupervised method HG+LARs outperforms the MRMR algorithm and Fisher score in all cases. On the USPS dataset (see Fig. 6.8(d)), even though MRMR and Fisher score can give good classification performance when more than 100 features are selected, HG+LARs achieves a better result with a much smaller number of features, i.e., less than 60 features. This implies that our proposed method is able to locate both the optimal size of the feature subset and perform accurate classification of the samples based on just a few of the most important features.

From Fig. 6.8(a)-(c), it is interesting to note that that the UDFS algorithm gives a better result than the alternative unsupervised methods (i.e. the Laplacian score and the

SPEC) when the number of selected features is small. The reason for this is that unlike traditional methods which treat each feature individually and which hence are suboptimal, the UDFS method directly optimizes the score over the entire selected feature subset. As a result, a better feature subset can be obtained. On the other hand, the alternative unsupervised feature selection method, i.e. the Laplacian score and SPEC, fail to locate the most discriminative features. This may be explained by our observation that they are unable to handle feature redundancy. For instance, they may repeatedly select highly correlated features in the selection process. And It has been known that redundant features can adversely affect the performance of classification and clustering.

The best result for each method together with the corresponding size of the selected feature subset are shown in Table. 6.2. In the table, the classification accuracy is shown first and the optimal number of features selected is reported in brackets. Overall, HG+LARs achieves the highest degree of dimensionality reduction, i.e. it selects a smaller feature subset compared with those obtained by the alternative methods. For example, in the MNIST data set, the best result obtained by the alternative feature selection methods is 82.5% with the MRMR algorithm and 284 features. However, our proposed method HG+LARs gives a better accuracy 84.33% when only 90 features are used. The results further verify that our feature selection method can guarantee the optimal size of feature subset, as it not only achieves a higher degree of dimensionality reduction but it also gives better discriminability.

From the accuracy rate in Table. 5.2 and Table. 6.2, we also notice that the hypergraph representation based method HG+LARs does not outperform the graph representation based method kECA+LARs on ORL dataset. The reason is that, although hypergraph representations allow vertices to be multiply connected by hyperedges and hence capture multiple or higher order relationships, our method is confined to uniform hypergraph and does not lend itself to generalization. The reason for this lies in the difficulty in formulation a nonuniform hypergraph in a mathematically neat way for computation. These

have yet to be a widely accepted and consistent way for representing and characterizing nonuniform hypergraph, and this remains an open problem when exploiting hypergraph for feature selection.

| Dataset | ORL | CMU PIE | MPEG-7 | USPS | MNIST |
|---|---|---|---|---|---|
| MRMR | 1.47 | 0.51 | **0.0839** | 1.1830 | 0.2112 |
| Fisher Score | 1.72 | 0.53 | 0.1508 | 1.0920 | 0.2304 |
| Laplacian Score | 1.68 | 0.67 | 0.1221 | 1.3540 | 0.2587 |
| SPEC | 1.65 | 0.66 | 0.2310 | 1.5600 | 0.2431 |
| UDFS | 1.62 | 0.63 | 0.0920 | 1.4200 | 0.3123 |
| HG+LARs | **1.37** | **0.47** | 0.0906 | **0.9825** | **0.1373** |

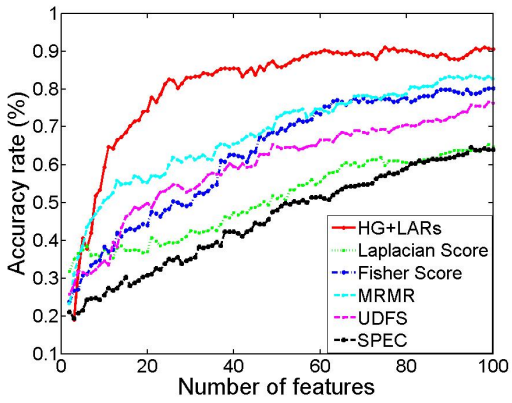
Table 6.3: Averaged Redundancy rate of Subsets Selected Using Different Algorithms.

### 6.4.4 Redundancy Rate

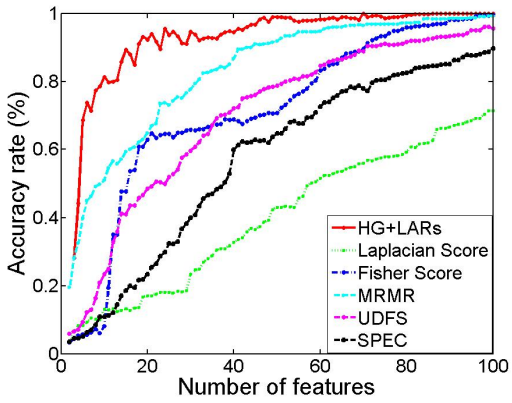
Table. 6.3 shows the comparative results of our proposed method with the alternative feature selection methods using the top $n$ features, where $n$ is the instance number of the training data. We chose $n$, since when the number of selected features is larger than $n$, any feature can be expressed by a linear combination of the remaining ones, which will introduce unnecessary redundancy in the evaluation stage. In the table, the boldfaced values are the lowest redundancy rates. The subset obtained by our proposed scheme has the least redundant. This further verifier that our propose algorithm is able to remove redundant features.

## 6.5 Conclusion

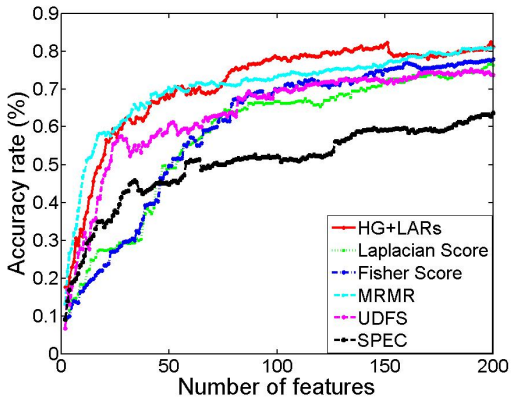
In this chapter, we have presented an unsupervised feature selection method based on a novel hypergraph representation framework. There are two main advantages can be drawn from this work. The first is that by incorporating MII for higher order similarities measure, we establish a novel hypergraph framework which is used for characterizing the multiple relationships within a set of samples. Thus, the structural information latent in the data can be more effectively modeled. Secondly, we derive a hypergraph embedding view of feature selection which casting the feature discriminant analysis into a regression framework that considers the correlations among features. As a result, we can evaluate joint feature combinations, rather than being confined to consider them individually. These properties enable our method to be able to handle feature redundancy effectively.



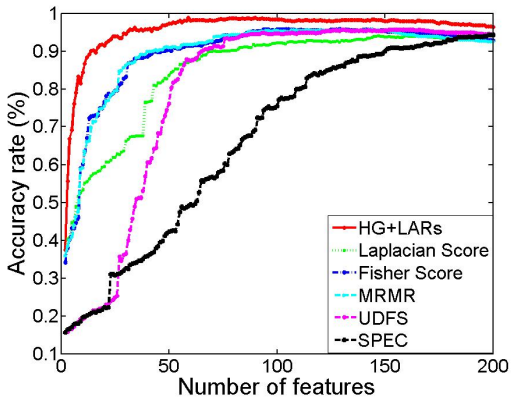

(a) ORL dataset



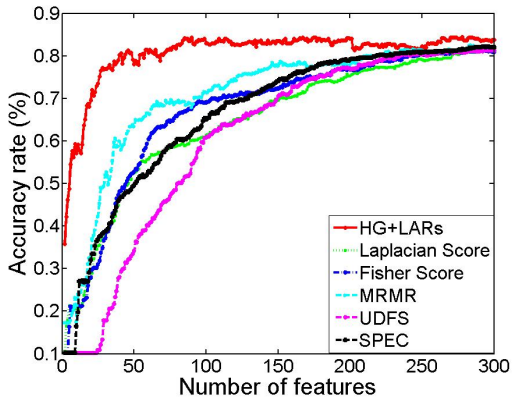

(b) CMU PIE dataset




(c) MPEG-7 dataset




(d) USPS dataset




(e) MNIST dataset

Figure 6.8: Accuracy rate vs. the number of selected features on five benchmark image datasets.

# Chapter 7

# Conclusions and Future Work

In this chapter, we first summarize the main contributions of this thesis, and then analyze the limitations of the developed methods. Following the analysis, we discuss some possible solutions and give suggestions for future feature selection.

## 7.1 Summary of Contributions

To select features from higher order correlations, we have firstly proposed a new information theoretic criterion referred to as the multidimensional interaction information (MII) to measure the significance of different feature combinations. The advantage of MII is that it is sensitive to the relations between feature combinations. As a result it can be used to seek third or even higher order dependencies between the relevant features. Hence, we can evaluate features jointly rather than individually. Thus we are able to handle feature redundancy. However, MII involves evaluating all possible interactions among the selected features which has two problems. The first is that it requires an exhaustive "combinatorial" search over the feature space. The second is that it demands large training sample sizes to estimate the higher order joint probability distribution in MII with a high dimensional kernel. In Chapter 3, we take into account above problems, and develop a filter-based approaches named *Graph based Information-theoretic Feature Selection*,

which is capable of reducing the search space for higher order interactions. Specifically, by incorporating mutual information for pairwise feature similarity measure, we first establish a novel feature graph framework which is used for characterizing the informativeness between the pair of features. We then locate the relevant feature subset (RFS) from the feature graph by maximizing features' average pairwise relevance. The RFS is expected to has little redundancy and very strong discriminating power. In doing so we can limit the search space in using MII for further feature selection.

However, in some situations the graph representation for relational patterns can lead to substantial loss of information. Therefore, in Chapter 4, we construct a feature hypergraph in which each node corresponds to a feature, and each edge has a weight corresponding to the MII among features connected by that edge. Then, we apply hypergraph clustering to the hypergraph in order to locate the most informative feature subset (mIFS), which has both low redundancy and strong discriminating power. In contrast with existing feature selection methods, our proposed methods is able to determine the number of relevant features automatically.

Furthermore, we develop two regularization based unsupervised feature selection methods, which on one hand can utilize the unlabeled data, on the other hand can evaluate features jointly rather than individually. In this case, larger feature combinations are considered. The reason for this is that although an individual feature may have limited relevance to a particular class, when taken in combination with other features it may be strongly relevant to the class. The idea underpinning these two methods is to select the features which best preserve the manifold structure derived from the entire feature set. Specifically, in Chapter 5, we propose a new two-step spectral regression technique for unsupervised feature selection. In the first step, we use kernel entropy component analysis (kECA) to transform the data into a lower-dimensional space so as to improve class separation. Second, we use $\ell_1$-norm regularization to select the features that best align with the data embedding resulting from kECA. The advantage of kECA is that dimensionality

reducing data transformation maximally preserves entropy estimates for the input data whilst also best preserving the cluster structure of the data. Using $\ell_1$-norm regularization, we cast feature discriminant analysis into a regression framework which accommodates the correlations among features. As a result, we can evaluate joint feature combinations, rather than being confined to consider them individually. In Chapter 6, by incorporating MII for higher order similarities measure, we establish a novel hypergraph framework which is used for characterizing the multiple relationships within a set of samples (e.g. face samples under varying illumination conditions). Thus, the structural information latent in the data can be more effectively modeled. Then an unsupervised method is proposed to find the discriminating feature subset on the basis of hypergraph representation. For the unsupervised learning, we derive a hypergraph embedding view of feature selection, where the projection matrix is constrained to be a selection matrix designed to select the optimal feature subset.

## 7.2 Limitations

Although the methods described in this thesis outperform the state of the art methods, there are still some limitations to be noted. Some of these weaknesses could be addressed in future work.

#### Limitations of Selecting Global Feature Subset

The greatest limitation of our proposed methods is their attempt to select a global feature subset for all the clusters present in the data. However, in doing so we neglect the fact that different clusters may exist in different feature subset which is referred to as local features. Fig. 7.1 shows an intuitive example. Traditional feature selection methods may select a global relevant feature subset $\{X_1, X_2, X_3\}$, which is obviously unable to work well, as different clusters exists in different subspaces. As shown in Fig. 7.1a, $C_1$ and $C_2$

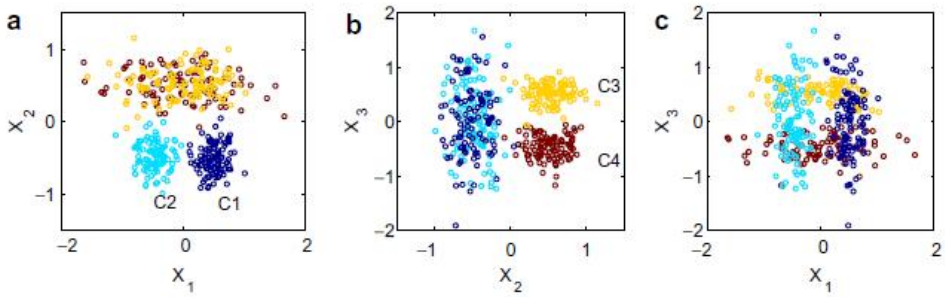



Figure 7.1: a-c show the projections of four clusters on the plane of two joint features, respectively. (a) in $X_1$ and $X_2$, (b) in $X_2$ and $X_3$, (c) in $X_1$ and $X_3$

can only be separated in $\{X_1, X_2\}$ after removing $X_3$ while shown in Fig. 7.1b $C_3$ and $C_4$ can only be separated in $\{X_2, X_3\}$ after removing $X_1$.

**Limitations of Hypergraph Construction**

Although we have described how to construct a hypergraph to abstract higher order feature relations, the problem of how to mathematically represent the underpinning hypergraph remains an open problem. In this thesis, we only consider those hypergraphs whose hyperedges have the same number of vertices, which are referred to as uniform hypergraphs. It would be more interesting to generate the problem to non-uniform hypergraphs (i.e. hypergraphs with varying hyperedge cardinalities). In addition, the similarity measure for hyperedge plays an important role in establishing the compatibility tensor and has a great influence on the subsequent hypergraph representation. In this thesis, we have use MII as a higher order similarity measure for point tuples. Although this measure has already been used in algorithms for various pattern recognition problems, there is still no theoretical evidence to prove it to be optimal options. Therefore, the choice of similarity measures in this work is heuristic, and we need to carry out a further investigation on how to define a reasonable similarity measure that is capable of reflecting structural features more convincingly.

#### Restrictions on Separating Data Structure Learning and Feature Selection

In Chapter 6, we apply hypergraph embedding and lasso penalized regression for feature discriminant analysis. We have shown both theoretically and experimentally that this method outperform the alternative feature selection methods. However, the performance of feature selection is largely determined by the effectiveness of data transformation obtained by hypergraph embedding. The reason for this limitation is that the process of hypergraph embedding is independent with feature selection. Once the hypergraph is determined to characterize data structure, it is fixed in the subsequent feature selection or regression steps. It would be more interesting the hypergraph embedding and feature selection could be performed in an integrated fashion. That is to say that if the hypergraph embedding can adaptively change w.r.t. the subsequent feature selection or regression procedures, i.e., the hypergraph not only can characterize data structure, but also indicate the requirements of regression, this method would perform better.

## 7.3 Future Work

To address the shortcomings described in the preceding section, we suggest some possible approaches to overcome them in future work.

#### Localized Feature Selection

In order to select the local feature subset, we would like to associate different classes with different feature subsets. One possible solution is to develop a localized graph-based feature selection algorithm consisting of two steps, namely, i) based on the label information, we first construct a graph for each class of dataset in which each node corresponds to a feature, and each edge has a weight corresponding to the mutual information between features connected by that edge, ii) we then perform dominant set clustering analysis for

the graphs to locate the informative feature subset for each class.

### Non-uniform Hypergraph

In this thesis, we only consider the case where all the hyperedges have the same number of vertices, which is referred to uniform hypergraph. It would be more interesting to develop the non-uniform hypergraph (i.e. the hyperedge cardinality varies). Furthermore, we will investigate how the new hypergraph models can be used to encode more complex multiple relationships so that more effective feature selection strategies can be developed. Since the method of similarity measure for hyperedge plays important part in determining the representational power of hypergraph construction, it might be interesting for us to adopt some more sophisticated strategies (e.g. Mahalanobis distance matric) to identify the similarity among data. Moreover, we may introduce more feature descriptors (such as texture information, shape information) into our frameworks to construct more hyperedges to further improve the expressive power of hypergraph based models. We also plan to introduce prior information into the hypergraph framework for real-world problems including video segmentation and information retrieval.

# Glossary of Notation

| | |
|---|---|
| $G(V, E)$ | Graph with vertex set $V$ and edge set $E$ |
| $H(V, E)$ | Hypergraph with vertex set $V$ and edge set $E$ |
| $\boldsymbol{A}$ | Adjacency matrix |
| $\boldsymbol{L}$ | Laplacian matrix |
| $\boldsymbol{H}$ | Incidence matrix |
| $\boldsymbol{D}$ | Degree matrix |
| $\mathbf{W}$ | Weight matrix |
| $\mathbf{Y}$ | Low dimensional embedding matrix |
| $\Phi$ | Projection matrix |
| $\mathbf{K}$ | Kernel matrix |
| $\boldsymbol{a}$ | Indicator vector |
| $\mathbf{X}$ | Dataset |
| $N$ | The number of samples |
| $d$ | The dimension of dataset |
| $C$ | Class labels |
| $S$ | Selected feature subset |
| $f_i$ | The $i$-th feature |
| $\alpha$ | Eigenvector |
| $\lambda$ | Eigenvalue |

# Bibliography


[1] A. Argyriou, T. Evgeniou and M. Pontil. Convex multi-task feature learning. *Machine Learning*, 73(3): 243-272, 2008.

[2] A. Ng, M. Jordan and Y. Weiss. On spectral clustering: Analysis and an algorithm. *Advances in Neural Information Processing Systems*, 2: 849-856, 2001.

[3] A. Renyi. On measures of entropy and information. *Fourth Berkeley Symposium on Mathematical Statistics and Probability*, pp. 547-561, 1961.

[4] A. Shashua, R. Zass and T. Hazan. Multi-way clustering using super-symmetric non-negative tensor factorization. In *Proceedings of European Conference on Computer Vision*, 4: 595-608, 2006.

[5] A. Björck. Numberical methods for least squares problems. SIAM Press, Philadelphia, PA, 1996.

[6] A. Jakulin and I. Bratko. Analyzing attribute dependencies. In *Proceedings of Principles of Knowledge Discovery in Data*, 2838: 229-240, 2003.

[7] B. Guo and M. Nixon. Gait feature subset selection by mutual information. *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans*, 39(1): 36-46, 2008.

[8] B. Kulis, S. Basu, I. Dhillon and R. Mooney. Semi-supervised graph clustering: A kernel approach. In *Proceedings of The 22nd International Conference on Machine Learning*, 74(1): 457-464, 2005.

[9] B. Efron, T. Hastie, I. Johnstone and R. Tibshirani. Least angle regression. *The Annals of Statistics*, 32(2): 407-499, 2004.

[10] B. Scholkopf, A. Smola and K. R. Muller. Nonlinear component analysis as a kernel eigenvalue problem. *Neural Computation*, 10(5): 1299-1319, 1998.

[11] C. Bishop. Pattern Recognition and Machine Learning. Springer New York, 2006.

[12] C. Bishop. Neural networks for pattern recognition. Oxford University Press, 1995.

[13] C. Shannon. A mathematical theory of communication. *ACM SIGMOBILE Mobile Computing and Communications Review*, 5(1): 3-55, 2001.

[14] C. Chang and C. Lin. LIBSVM: a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2: 1-27, 2011.

[15] D. Jiang, C. Tang and A. Zhang. Cluster analysis for gene expression data: A survey. *IEEE Transactions on Knowledge and Data Engineering*, 16(11): 1370-1386, 2004.

[16] D. Cai, C. Zhang and X. He. Unsupervised feature selection for multi-cluster data. In *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge discovery and data mining*, 6: 333-342, 2010.

[17] D. W. Jacobs, P. N. Belhumeur and R. Basri. Comparing images under variable illumination. In *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 610-617, 1998.

[18] D. Gibson, J. Kleinberg and P. Raghavan. Clustering categorical data: An approach based on dynamical systems. *The International Journal on Very Large Data Bases*, 8(3-4): 222-236, 2000.

[19] D. Zhou, J. Huang and B. Scholkopf. Learning with hypergraphs: Clustering, classification, and embedding. *Advances in Neural Information Processing Systems*, 19: 1601-1608, 2006.

[20] E. Parzen. On estimation of a probability density function and mode. *The Annals of Mathematical Statistics*, 33(3): 1065-1076, 1962.

[21] F. Chung. Spectral Graph Theory. *American Mathematical Society*, 1992.

[22] F. Chung. The Laplacian of a hypergraph. *AMS DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, pp. 21-36, 1993.

[23] F. Bach and M. Jordan. Learning spectral clustering, with application to speech separation. *The Journal of Machine Learning Research*, 7: 1963-2001, 2006.

[24] F. Nie, S. Xiang, Y. Jia, C. Zhang and S. Yan. Trace ratio criterion for feature selection. In *Proceedings of 23rd AAAI Conf. Artif. Intell*, 2: 671-676, 2008.

[25] F. Nie, H. Wang, H. Huang and C. Ding. Unsupervised and semi-supervised learning via L1-norm graph. In *Proceedings of IEEE International Conference on Computer Vision*, pp. 2268-2273, 2011.

[26] G. H. John, R. Kohavi and K. Pfleger. Irrelevant features and the subset selection problem. In *Proceedings of the Eleventh International Conference on Machine Learning*, pp. 121-129, 1994.

[27] H. Peng, F. Long and C. Ding. Feature selection based on mutual information: Criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(8): 1226-1238, 2005.

[28] H. Cheng, Z. Qin, W. Qian and W. Liu. Conditional mutual information based feature selection. In *Proceedings of IEEE International Symposium on Knowledge Acquisition and Modeling*, pp. 103-107, 2008.

[29] H. Yang and J. Moody. Feature selection based on joint mutual information. In *Proceedings of International ICSC Symposium on Advances in Intelligent Data Analysis*, pp. 22-25, 1999.

[30] H. Almuallim and T. G. Dietterich. Learning with many irrelevant features. In *Proceedings of the Ninth National Conference on Artificial Intelligence*, pp. 547-552, 1991.

[31] I. Jolliffe. Principal Component Analysis. Springer Series in Statistics, Berlin: Springer, 1986.

[32] I. Guyon, J. Weston, S. Barnhill and V. Vapnik. Gene selection for cancer classification using support vector machines. *Machine Learning*, 46(1): 389-422, 2002.

[33] J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8): 888-905, 2000.

[34] J. A. Rodri. On the Laplacian eigenvalues and metric parameters of hypergraphs. *Linear and Multilinear Algebra*, 50(1): 1-14, 2002.

[35] J. W. Weibull. Evolutionary Game Theory. *MIT Press*, 1995.

[36] K. Balagani, V. Phoha, S. Iyengar and N. Balakrishnan. On Guo and Nixon's criterion for feature subset selection: Assumptions, implications, and alternative options. *IEEE TSMC-A: Systems and Humans*, pp. 651-655, 2010.

[37] L. Zhou, L. Wang and C. Shen. Feature selection with redundancy-constrained class separability. *IEEE Transactions on Neural Networks*, 21(5): 853-858, 2010.

[38] L. Hagen and A. B. Kahng. New spectral methods for ratio cut partitioning and clustering. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 11(9): 1074-1085, 1992.

[39] L. E. Baum and J. A. Eagon. An inequality with applications to statistical estimation for probabilistic functions of Markov processes and to a model for ecology. Bull. Amer. Math. Soc, 73(3): 360-363, 1967.

[40] L. P. Jing, H. K. Huang and H. B. Shi. Improved feature selection approach TFIDF in text mining. In *Proceedings of International Conference on Machine Learning and Cybernetics*, 2: 944-946, 2002.

[41] L. Sun, S. Ji and J. Ye. Hypergraph spectral learning for multi-label classification. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 668-676, 2008.

[42] M. Pavan and M. Pelillo. A new graph-theoretic approach to clustering and segmentation. In *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 145-152, 2003.

[43] M. Pavan and M. Pelillo. Dominant sets and pairwise clustering. *IEEE Transactions Pattern Analysis and Machine Intelligence*, 29(1): 167-172, 2007.

[44] M. Belkin and P. Niyogi. Laplacian eigenmaps and spectral techniques for embedding and clustering. *Advances in Neural Information Processing Systems*, 1: 585-592, 2002.

[45] M. Belkin and P. Niyogi. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Computation*, 15(6): 1373-1396, 2003.

[46] M. W. Seeger. Bayesian inference and optimal design for the sparse linear model. *The Journal of Machine Learning Research*, 9: 759-813, 2008.

[47] M. Bolla. Spectra, Euclidean representations and clusterings of hypergraphs. *Discrete Mathematics*, 117(1-3): 19-39, 1993.

[48] M. A. Hall. Correlation-based feature selection for machine learning. Ph.D. Thesis, The University of Waikato, 1999.

[49] N. Kwak and C. H. Choi. Input feature selection by mutual information based on parzen window. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(12): 1667-1671, 2002.

[50] N. Kwak and C. H. Choi. Input feature selection for classification problems. *IEEE Transactions on Neural Networks*, 13(1): 143-159, 2002.

[51] P. Devijver and J. Kittler. Pattern Recognition: A statistical approach. Prentice-Hall London, 1982.

[52] P. K. Chan, M.D. F. Schlag and J. Y. Zien. Spectral k-way ratio-cut partitioning and clustering. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 13(9): 1088-1096, 1994.

[53] P. A. Estévez, M. Tesmer, C. A. Perez and J. M. Zurada. Normalized mutual information feature selection. *IEEE Transactions on Neural Networks*, 20(2): 189-201, 2009.

[54] P. N. Belhumeur and D. J. Kriegman. What is the set of images of an object under all possible illumination conditions?. *International Journal of Computer Vision*, 28(3): 245-260, 1998.

[55] R. Battiti. Using mutual information for selecting features in supervised neural net learning. *IEEE Transactions on Neural Networks*, 5(4): 537-550, 2002.

[56] R. Jin, C. Ding and F. Kang. A probabilistic approach for optimizing spectral clustering. *Advances in Neural Information Processing systems*, 18, MIT Press, Cambridge, MA, 2005.

[57] R. O. Duda, P. E. Hart and D. G. Stork. Pattern classification. Wiley New York, 2001.

[58] R. Kohavi and G. H. John. Wrappers for feature subset selection. *Artificial Intelligence*, 97(1-2): 273-324, 1997.

[59] R. Klimmek and F. Wagner. A simple hypergraph min cut algorithm. Technical Report B 96-02, University Berlin, Germany, March 1996.

[60] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, 58(1): 267-288, 1996.

[61] S. Agarwal, J. Lim, L. Zelnik-Manor, P. Perona, D. Kriegman and S. Belongie. Beyond pairwise clustering. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2: 838-845, 2005.

[62] S. Agarwal, K. Branson and S. Belongie. Higher order learning with graphs. In *Proceedings of the 23rd International Conference on Machine Learning*, pp. 17-24, 2006.

[63] S. T. Roweis and L. K. Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290: 2323-2326, 2000.

[64] S. R. Bulò and M. Pelillo. A game-theoretic approach to hypergraph clustering. *Advances in Neural Information Processing Systems*, pp. 1571-1579, 2009.

[65] S. B. Thrun and et al. The monk's problems a performance comparison of different learning algorithms. *Carnegie Mellon University, Tech. Rep. CMU-CS-91-197*, 1991.

[66] T. M. Cover, J. A. AThomas and J. Wiley. Elements of information theory. Wiley Online Library, 1991.

[67] T. M. Cover. The best two independent measurements are not the two best. *IEEE Transactions on Systems, Man and Cybernetics*, 4(1): 116-117, 1974.

[68] T. G. Kolda and B. W. Bader. Tensor decompositions and applications. *SIAM Review*, 51(3): 455-500, 2009.

[69] U. Von Luxburg. A tutorial on spectral clustering. *Statistics and Computing*, 17(4): 395-416, 2007.

[70] V. Jain and H. Zhang. A spectral approach to shape-based retrieval of articulated 3D models. *Computer-Aided Design*, 39(5): 398-407, 2007.

[71] W. C. W. Li and P. Solé. Spectra of regular graphs and hypergraphs and orthogonal polynomials. *European Journal of Combinatorics*, 17(5): 461-477, 1996.

[72] W. McGill. Multivariate information transmission. *IRE Professional Group on Information Theory*, 4(4): 93-111, 1954.

[73] X. He, D. Cai and P. Niyogi. Laplacian score for feature selection. *Advances in Neural Information Processing Systems*, 18: 507-514, 2005.

[74] X. He and P. Niyogi. Locality preserving projections (LPP). *Advances in Neural Information Processing Systems*, 16: 153-160, 2004.

[75] X. He, D. Cai, S. Yan and H. J. Zhang. Neighborhood preserving embedding. In *Proceedings of Tenth IEEE International Conference on Computer Vision*, 2: 1208-1213, 2005.

[76] X. Zhu, Z. Ghahramani and J. Lafferty. Semi-supervised learning using gaussian fields and harmonic functions. In *Proceedings of the Twentieth International Conference on Machine Learning*, 20(2): 912-919, 2003.

[77] Y. Yang, H. T. Shen, Z. Ma, Z. Huang and X. Zhou. L21-norm regularized discriminative feature selection for unsupervised learning. In *Proceedings of International Joint Conferences on Artificial Intelligence*, pp. 1589-1594, 2011.

[78] Y. Adini, Y. Moses and S. Ullman. Face recognition: The problem of compensating for changes in illumination direction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7): 721-732, 1997.

[79] Y. Huang, Q. Liu, F. Lv, Y. Gong and D. N. Metaxas. Unsupervised image categorization by hypergraph partition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(6): 1266-1273, 2011.

[80] Y. Freund and R. R. Schapire. Experiments with a new boosting algorithm. In *Proceedings of the Thirteenth International Conference on Machine Learning*, pp. 148-156, 1996.

[81] Z. Zhao and H. Liu. Spectral feature selection for supervised and unsupervised learning. In *Proceedings of the 24th International Conference on Machine Learning*, pp. 1151-1157, 2007.

[82] Z. Zhang and E. R. Hancock. Feature selection for gender classification. In *Proceedings of the 5th Iberian Conference on Pattern Recognition and Image Analysis*, pp. 76-83, 2011.

[83] Z. Zhang and E. R. Hancock. Hypergraph based Information-theoretic Feature Selection. *Pattern Recognition Letters*, 33: 1991-1999, 2012.