Effective Global Approaches for Mutual Information Based Feature Selection

Nguyen Xuan Vinh Jeffrey Chan Simone Romano James Bailey
Department of Computing and Information Systems, The University of Melbourne, Australia
{ vinh.nguyen | jeffrey.chan | simone.romano | baileyj }@unimelb.edu.au

ABSTRACT

Most current mutual information (MI) based feature selection techniques are greedy in nature thus are prone to sub-optimal decisions. Potential performance improvements could be gained by systematically posing MI-based feature selection as a global optimization problem. A rare attempt at providing a global solution for the MI-based feature selection is the recently proposed Quadratic Programming Feature Selection (QPFS) approach. We point out that the QPFS formulation faces several non-trivial issues, in particular, how to properly treat feature 'self-redundancy' while ensuring the convexity of the objective function. In this paper, we take a systematic approach to the problem of global MI-based feature selection. We show how the resulting NP-hard global optimization problem could be efficiently approximately solved via spectral relaxation and semi-definite programming techniques. We experimentally demonstrate the efficiency and effectiveness of these novel feature selection frameworks.

Categories and Subject Descriptors

I.5.2 [**Pattern Recognition**]: Feature evaluation and selection

Keywords

Feature selection; mutual information; spectral relaxation; semi-definite programming; global optimization.

1. INTRODUCTION

Mutual information (MI) based approaches are an important feature selection paradigm in data mining. Over the years, these methods have gained increasing popularity, thanks especially to their ease of use, effectiveness and strong theoretical foundation rooted in information theory. Seventeen MI based feature selection approaches are listed in a recent comprehensive survey [3], summarizing nearly two decades of research in this area. The commonality of these methods lies in the fact that they all employ a greedy scheme to incrementally build the selected feature set, one at a time.

To gain some concreteness to our discussion, let us revisit a very popular mutual information-based feature selection family that is centred around the concepts of *redundancy* and *relevancy*. A particularly successful and well known instance of this family is the Minimum Redundancy Maximum Relevance (MRMR) framework [20]. Given a set of n features (which are often referred interchangeably to as variables, or attributes) $\mathbb{X} = \{X_1, \ldots, X_n\}$ and a target class variable C, the relevancy of X_i is measured by its mutual information (MI) with the class variable, i.e.,

$$Rel(X_i) \triangleq I(X_i; C) \triangleq \sum_{X_i, C} P(X_i, C) \log \frac{P(X_i, C)}{P(X_i)P(C)}$$

while its redundancy with respect to an already selected feature subset $\mathbb{S}$ is defined as

$$Red(X_i|\mathbb{S}) \triangleq \frac{1}{|\mathbb{S}|} \sum_{X_j \in \mathbb{S}} I(X_i; X_j)$$

Given these definitions of feature relevancy and redundancy, the MRMR framework [20] is a greedy scheme to select features one at a time, such that the i-th feature is selected maximizing the MRMR-objective:

$$\max_{X_i \in \mathbb{X} \setminus \mathbb{S}} \{Rel(X_i) - Red(X_i|\mathbb{S})\}$$

The generalized MRMR family is parameterized as

$$\mathcal{MRMR}: \max_{X_i \in \mathbb{X} \setminus \mathbb{S}} \left\{ I(X_i; C) - \alpha \sum_{X_j \in \mathbb{S}} I(X_i; X_j) \right\} \quad (1)$$

where α is a weighting factor that balances relevancy and redundancy, which is chosen to be $1/|\mathbb{S}|$ in the case of MRMR. The first member of this family, known as MIFS (Mutual Information Feature Selection) [2] with $\alpha = 1$, has been in fact introduced much earlier in the feature selection literature.

1.1 Global MI-based feature selection

Most current MI-based feature selection approaches are of an incremental nature, similar to the $\mathcal{MRMR}$ formulation. As such, these methods are prone to suboptimal decisions, as selected features cannot be deselected at a later stage. Potential performance improvement could be gained by systematically posing MI-based feature selection as a global optimization problem, and making a global decision considering the interaction between all features concurrently. The first attempt in this direction is the recently proposed

KDD'14, August 24–27, 2014, New York, NY, USA.

http://dx.doi.org/10.1145/2623330.2623611 .

Quadratic Programming Feature Selection (QPFS) approach [21]. QPFS reformulates the MRMR feature selection problem as the following quadratic program:

$$\mathcal{QPFS}: \min_{\mathbf{x}} \left\{ \alpha \mathbf{x}^T \mathbf{H} \mathbf{x} - \mathbf{x}^T \mathbf{f} \right\} \text{ s.t. } \sum_{i=1}^{n} x_i = 1, x_i \geq 0 \quad (2)$$

where $\mathbf{f}_{n\times 1} = [I(X_1;C), \ldots, I(X_n;C)]^T$ is the vector of feature relevancy, $\mathbf{H}_{n\times n} = [I(X_i;X_j)]_{i,j=1\ldots n}$ is the matrix of feature pairwise redundancy, and $\mathbf{x}_{n\times 1}$ represents relative feature weights. Note that $\mathbf{H}_{ii}$ *is set to feature self-redundancy*, i.e. entropy, $\mathbf{H}_{ii} = I(X_i;X_i) = H(X_i)$. The most attractive characteristic of the QPFS formulation in (2) is that if $\mathbf{H}$ is positive (semi)definite, then $\mathcal{QPFS}$ is a convex quadratic program which can be solved efficiently in polynomial time for the globally optimal solution. The output $\mathbf{x}$ of this program is used for global feature ranking.

1.2 Theoretical issues with the QPFS framework

The reformulation of the incremental $\mathcal{MRMR}$ as a global quadratic program $\mathcal{QPFS}$ as proposed in [21], although being very attractive, poses several non-trivial intriguing questions that we shall elaborate below.

• **Positive definiteness of H**: A pre-requisite for $\mathcal{QPFS}$ to be a convex quadratic program, thus admitting an efficient polynomial-time procedure to find the global minimum, is that the Hessian matrix $\mathbf{H}$ of pairwise feature mutual information be positive (semi)definite. In other words, the mutual information function on the space of features must be a proper kernel function. Our investigation into this problem shows that there is currently little understanding on whether the MI is a proper kernel. While we have not been able to theoretically prove nor disprove the positive definiteness of $\mathbf{H}$, our practical evaluation of QPFS using Matlab quadratic program solver sometimes numerically encounters indefinite $\mathbf{H}$ (for instance, the largest negative eigenvalue could be as large as -50 on a dataset of 2000 features), where Matlab solver declares the problem to be non-convex and aborts the operation.

In the original paper [21], this theoretical issue has been mostly neglected. For problems of a very large number of features, the authors proposed to approximate $\mathbf{H}$ using only its largest eigenvalues, so $\mathcal{QPFS}$ becomes convex. However, as there exist many small and medium size problems where an approximation might not be needed, establishing the positive definiteness of $\mathbf{H}$ is still required to ensure the theoretical soundness of the approach.

• **How to treat self-redundancy?** In $\mathcal{QPFS}$, note that the cost matrix $\mathbf{H}$ penalizes features for their redundancy with respect to other features. The 'self-redundancy' terms $\mathbf{H}_{ii} = I(X_i;X_i) = H(X_i)$, as designated in the original paper [21], in fact penalize features for their intrinsic entropy. The question of how to treat self-redundancy presents us with the following dilemmas:

– Arguably, features should not be penalized for self redundancy. Unfortunately, if we put $\mathbf{H}_{ii} = 0$, then the Hessian matrix $\mathbf{H}$ becomes indefinite, violating the pre-requisite for the $\mathcal{QPFS}$ formulation to be convex.

– If we put $\mathbf{H}_{ii} = H(X_i)$ as proposed in [21], then there will be selection bias in favor of features with low entropy. In general, discrete features may have higher entropy because of more uniform distribution across its categories, or having more categories. As we will theoretically and empirically show next in this paper, penalizing features for self-redundancy leads to undesired behaviors.

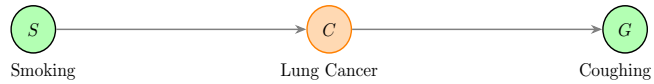
Example 1: We use a simple example here to show the counter-intuitive behavior of QPFS that penalizing features differently based on their entropy can lead to suboptimal decisions. Let us consider the following scenario, where a quaternary variable S (Smoking) takes 4 possible values ((1) none smoker; (2) 1 to 5 cigarettes per day; (3) 5 to 15 cigarettes per day; (4) more than 15 cigarettes per day). S causes the binary class variable C (0–none, 1–lung cancer) with joint probability distribution $P(S,C)$. C then in turn causes a binary characteristic feature G (Coughing, 0–occasionally, 1–frequently) with joint probability $P(C,G)$. The scenario is denoted by the Bayesian network and joint probability tables in Figure 1. The joint probability $P(S,G)$ can also be calculated as in Fig. 1. In this example, S can be used to perfectly classify C (using the rule $C = 0$ if $S \in \{1,2\}$ and $C = 1$ otherwise), while if G were used the minimal error rate achievable, i.e. Bayes error rate, will be 5%. Thus S–smoking should be clearly preferred over G–coughing as a predictive feature for C–lung cancer, albeit having a higher entropy, i.e. 2 bits vs. 1 bit. We can compute the following quantities:

$$\begin{aligned} I(G;C) &= 0.7136 \text{ bit} \\ I(S;C) &= 1 \text{ bit} \\ I(S;G) &= 0.7136 \text{ bit} \end{aligned}$$

The optimal solution to the $\mathcal{QPFS}$ formulation

$$\min_{x_i>0, \sum x_i = 1} \left\{ \mathbf{x}^T \begin{bmatrix} 2 & 0.7136 \\ 0.7136 & 1 \end{bmatrix} \mathbf{x} - \mathbf{x}^T \begin{bmatrix} 1 \\ 0.7136 \end{bmatrix} \right\} \quad (3)$$

is $\mathbf{x}^* = [0.42, 0.58]^T$, that is, the coughing G (weight 0.58) is ranked higher than smoking S (weight 0.42), which is incorrect.



P(S,C)	C=0	C=1
S=1	0.25	0
S=2	0.25	0
S=3	0	0.25
S=4	0	0.25

P(C,G)	C=0	C=1
G=0	0.475	0.025
G=1	0.025	0.475

P(S,G)	G=0	G=1
S=1	0.2375	0.0125
S=2	0.2375	0.0125
S=3	0.0125	0.2375
S=4	0.0125	0.2375

Figure 1: The three-variable example: QPFS gives preference to the feature with smaller entropy G, while S is a better explanatory variable for C albeit having a higher entropy.

1.3 Contribution

Motivated by the initial success as well as the theoretical gap within the QPFS framework, we set out to systematically investigate the problem of global MI-based feature selection. Our first contribution in this paper is to reconsider the QPFS formulation and resolve the theoretical issues associated with its current form, as discussed above. Our second, and principal contribution, is to propose a novel formulation for global MI-based feature selection that can be solved effectively via spectral relaxation and semi-definite programming techniques. Via extensive experiments on a

wide range of data sets, we establish the effectiveness and efficiency of our approach against other successful MI based feature selection techniques. We further show that for large data, low rank approximation can be applied to gain computational advantage to our global algorithm over its greedy counterpart.

2. PRELIMINARIES

It is worth noting that, in the original paper [21], the authors propose the quadratic formulation (2) without much explanatory detail. While that formulation is intuitively reasonable, let us take a more systematic, step-by-step derivation process implied behind QPFS, through which the inconsistency within QPFS itself will also be exposed. From a global optimization perspective, the incremental $\mathcal{MRMR}$ feature selection problem (1) can be reconsidered as a global *subset-selection* problem as follows:

$$\mathcal{SS}: \max_{\substack{\mathbb{S}\subset\mathbb{X}\\|\mathbb{S}|=k}} \left\{ \sum_{X_i\in\mathbb{S}} I(X_i;C) - \alpha \sum_{\substack{X_i,X_j\in\mathbb{S}\\ i\neq j}} I(X_i;X_j) \right\} \tag{4}$$

which can in turn be equivalently formulated as a quadratic integer programming problem as

$$\mathcal{QIP}: \max_{\mathbf{x}} \left\{ \mathbf{x}^T\mathbf{f} - \alpha\mathbf{x}^T\mathbf{H}\mathbf{x} \right\} \text{ s.t. } \mathbf{x}\in\{0,1\}^n, \sum_{i=1}^{n} x_i = k \tag{5}$$

Here, k is the desired size for the final feature set, $\mathbf{f}_{n\times 1} = [I(X_1;C),\ldots,I(X_n;C)]^T$ is still the vector of feature relevancy, and $\mathbf{H}_{n\times n} = [I(X_i;X_j)]_{i,j=1\ldots n}$ is still the matrix of feature pairwise redundancy, except $\mathbf{H}_{ii} = 0$, i.e. zero-valued 'self-redundancy' terms. Note that this is also the critical difference between our global formulation and $\mathcal{QPFS}$: clearly and naturally, there should be no penalty for feature self-redundancy, as evidenced in the $\mathcal{MRMR}$ and $\mathcal{SS}$ formulations.

Unfortunately, there is no known efficient solution for both $\mathcal{SS}$ and $\mathcal{QIP}$. $\mathcal{SS}$ is a hard combinatorial problem for which an exhaustive search will cost $O(n^k)$, i.e. exponential in the target set size, while similarly $\mathcal{QIP}$ is known to be an NP-hard problem [5]. Noting that relaxing the problem to the continuous domain might lead to a more computationally tractable problem, we drop the integral 0–1 constraint, resulting in

$$\max_{\mathbf{x}} \left\{ \mathbf{x}^T\mathbf{f} - \alpha\mathbf{x}^T\mathbf{H}\mathbf{x} \right\} \text{ s.t. } \sum_{i=1}^{n} x_i = k,\ x_i \geq 0 \tag{6}$$

With a change of variable $y_i = x_i/k$, we arrive at:

$$\min_{\mathbf{y}} \left\{ k\alpha\mathbf{y}^T\mathbf{H}\mathbf{y} - \mathbf{y}^T\mathbf{f} \right\} \text{ s.t. } \sum_{i=1}^{n} y_i = 1,\ y_i \geq 0 \tag{7}$$

Herein, $k\alpha$ plays the role of a dynamic balancing factor. The $\mathcal{QPFS}$ formulation (2) is essentially a simplified variant of (7), where one disregards k and fixes the balancing factor to the same constant, i.e. α.

2.1 The extended MRMR family

Before providing a systematic analysis on how convexity could be ensured and 'self-redundancy' should be treated in the QPFS framework, let us gain further insight into both MRMR and QPFS by considering an extended MRMR family that incorporates second-order dependancy. The material discussed in this section will also serve as building blocks for our new approach, presented in Section 3. We start by elaborating the theoretical underpinnings behind MRMR and other similar heuristics. The ultimate goal of mutual information (MI) based feature selection is to select a subset of features $\mathbb{S}$ that shares the highest MI with C, i.e. $\max_{\mathbb{S}\subset\mathbb{X}} I(\mathbb{S};C)$. As this is a hard combinatorial problem, a practical approach is to build the feature subset incrementally, so that the i-th feature is selected as:

$$\arg\max_{X_i\in\mathbb{X}\setminus\mathbb{S}} I(\mathbb{S}\cup X_i;C) \equiv \arg\max_{X_i\in\mathbb{X}\setminus\mathbb{S}} I(X_i;C|\mathbb{S}) \tag{8}$$

As the high-dimensional MI term $I(X_i;C|\mathbb{S})$ is still hard to estimate from limited samples, MRMR and many other MI-based heuristics approximate (8) using low-order MI terms as:

$$I(X_i;C|\mathbb{S}) \simeq I(X_i;C) - \frac{1}{|\mathbb{S}|}\sum_{X_j\in\mathbb{S}} I(X_i;X_j) \tag{9}$$

However, the following natural decomposition of $I(X_i;C|\mathbb{S})$

$$I(X_i;C|\mathbb{S}) = I(X_i;C) - [I(X_i;\mathbb{S}) - I(X_i;\mathbb{S}|C)] \tag{10}$$

suggests that redundancy is in fact composed of two parts: an unconditional redundancy term $I(X_i;\mathbb{S})$ and a class conditional part $I(X_i;\mathbb{S}|C)$ [6]. This insight gives rise to the following extended minimal redundancy maximal relevance (EMRMR) objective:

$$\mathcal{EMRMR}: \max_{X_i} \left\{ I(X_i;C) - \alpha \sum_{X_j\in\mathbb{S}} [I(X_i;X_j) - I(X_i;X_j|C)] \right\} \tag{11}$$

This variant of MRMR has been introduced in the literature and observed to be more effective [3, 13, 14]. Similar to MRMR, EMRMR can be cast as an extended quadratic programming feature selection (EQPFS) problem as:

$$\begin{aligned} \mathcal{EQPFS} \quad &: \quad \min_{\mathbf{x}} \left\{ \alpha\mathbf{x}^T[\mathbf{H}_1 - \mathbf{H}_2]\mathbf{x} - \mathbf{x}^T\mathbf{f} \right\} \\ &\text{s.t.} \qquad \sum_{i=1}^{n} x_i = 1, x_i \geq 0 \end{aligned} \tag{12}$$

Here, $\mathbf{H}_1 = [I(X_i;X_j)]_{n\times n}$ and $\mathbf{H}_2 = [I(X_i;X_j|C)]_{n\times n}$ together make up the 'total-redundancy' matrix $\mathbf{H} = \mathbf{H}_1 - \mathbf{H}_2$. Similar to the QPFS formulation, we are presented with different choices about how to treat the total-self-redundancy terms, i.e. the diagonal elements of $\mathbf{H}_1$ and $\mathbf{H}_2$.

- If we set $\mathbf{H}_{1ii} = \mathbf{H}_{2ii} = 0$, i.e. no penalty for self-total-redundancy, then $\mathbf{H}$ is indefinite, hence $\mathcal{EQPFS}$ is non-convex and a global solution cannot be efficiently located.

- If we set $\mathbf{H}_{1ii} = I(X_i;X_i) = H(X_i)$ as in the original QPFS formulation [21], then analogously, $\mathbf{H}_{2ii}$ should be set to $I(X_i;X_i|C) = H(X_i|C)$. Thus $\mathbf{H}_{ii} = \mathbf{H}_{1ii} - \mathbf{H}_{2ii} = H(X_i) - H(X_i|C) = I(X;C)$, i.e. features which share more information with C are penalized more, which is clearly counter-intuitive and undesirable. In the next section, we provide a systematic analysis on how self-redundancy in the QPFS and EQPFS frameworks should be treated.

2.2 How to properly treat self-redundancy?

We argue that the most proper approach for treating self redundancy is that, *there should be no penalty for self redundancy*, i.e. $\mathbf{H}_{ii} = 0$, as clearly evident in the original

$\mathcal{MRMR}$ formulation. This choice however leads us to a non-convex quadratic program. We shall point out here that assigning $\mathbf{H}_{ii} = H(X_i)$ as in [21] in fact provides a convex approximation to the originally non-convex quadratic program. However, setting $\mathbf{H}_{ii} = H(X_i)$ leads to some counter-intuitive observation about $\mathcal{QPFS}$ (higher penalty for features with higher entropy) and $\mathcal{EQPFS}$ (higher penalty for features which share higher MI with C) as we have pointed out.

We propose that $\mathcal{QPFS}$ and $\mathcal{EQPFS}$ could be convexified by setting the diagonal elements of $\mathbf{H}$ to the same value $\lambda > 0$ sufficiently large to ensure the positive (semi)definiteness of the Hessian matrix. Formally, the general convexified $\mathcal{EQPFS}$ is as follows:

$$\min_{\mathbf{x}, \sum_{i=1}^{n} x_i = 1, x_i \geq 0} \left\{ \alpha \mathbf{x}^T [\mathbf{H}_1 - \beta \mathbf{H}_2 + \lambda \mathbf{I}] \mathbf{x} - \mathbf{x}^T \mathbf{f} \right\} \quad (13)$$

where $\mathbf{I}$ is the identity matrix, both α and β play the role of balancing factors, and λ is a convexification parameter. β is employed to balance the unconditional redundancy (in $\mathbf{H}_1$) and the class conditional redundancy (in $\mathbf{H}_2$), as proposed in [14]. At $\beta = 0$, $\mathcal{EQPFS}$ reduces to the original $\mathcal{QPFS}$. With this approach, all features receive the same penalty for 'self-redundancy' λ, although the real purpose of λ is to convexify the problem, not to impose a penalty on self-redundancy. It is noted that different choices of $\{\alpha, \beta, \lambda\}$ can lead to different solutions corresponding to different feature rankings.

3. A NOVEL GLOBAL MI-BASED FEATURE SELECTION PARADIGM

In this section, we set out to design a novel, systematic global approach for MI-based feature selection. Our desiderata for such an ideal global framework is two-fold: (i) ability to handle second-order feature dependancy as in $\mathcal{EMRMR}$, (ii) strong theoretical foundation, with few or no ad-hoc parameters, such as the balancing parameters and convexification parameter as in the 'remedied' QPFS framework (13).

Our first ingredient for such new framework is the following nice theoretical result, which states that the relevancy, unconditional redundancy and class-conditional redundancy, can all be combined neatly into a single quantity, namely the conditional mutual information (CMI).

THEOREM 1. *We have:*

$$\begin{aligned} \sum_{X_j \in \mathbb{S}} I(X_i; C|X_j) &= |\mathbb{S}| I(X_i; C) \quad (14) \\ &- \sum_{X_j \in \mathbb{S}} [I(X_i; X_j) - I(X_i; X_j|C)] \end{aligned}$$

PROOF. The proof is straightforward using the following decomposition of the conditional MI:

$$I(X_i; C|X_j) = I(X_i; C) - I(X_i; X_j) + I(X_i; X_j|C) \quad (15)$$

□

In fact, now we can see a chain of relationship between the high-dimensional conditional relevancy term in (8), the CMI and the extended MRMR criteria:

$$\begin{aligned} I(X_i; C|\mathbb{S}) &\simeq \textstyle\sum_{X_j \in \mathbb{S}} I(X_i; C|X_j) \quad (16) \\ &\equiv |\mathbb{S}| I(X_i; C) - \textstyle\sum_{X_j \in \mathbb{S}} [I(X_i; X_j) + I(X_i; X_j|C)] \end{aligned}$$

In light of these connections, we propose a global subset selection problem based on the CMI as follows:

$$\mathcal{SS}_{\mathcal{CMJ}} : \max_{\substack{\mathbb{S} \subset \mathbb{X} \\ |\mathbb{S}| = k}} \left\{ \sum_{X_i \in \mathbb{S}} I(X_i; C) + \sum_{X_i, X_j \in \mathbb{S}} I(X_i; C|X_j) \right\} \quad (17)$$

which can be equivalently reformulated in the form of a quadratic integer programming problem:

$$\mathcal{QIP}_{\mathcal{CMJ}} : \max_{\mathbf{x}} \left\{ \mathbf{x}^T \mathbf{Q} \mathbf{x} \right\} \text{ s.t. } \mathbf{x} \in \{0, 1\}^n, \|\mathbf{x}\| = \sqrt{k} \quad (18)$$

where $\mathbf{Q}_{ii} = I(X_i; C)$ and $\mathbf{Q}_{ij} = I(X_i; C|X_j), i \neq j$. Note that for $\mathbf{x} \in \{0, 1\}^n$, we have $\sum_{i=1}^{n} x_i = k \Leftrightarrow \|\mathbf{x}\| = \sqrt{k}$. Here we use the norm constraint for set cardinality, as it results in more computationally tractable relaxations, as will be seen in the next sections. It is noted that $\mathbf{Q}$ is, in general, asymmetric. However, it could be replaced by the symmetric form $(\mathbf{Q} + \mathbf{Q}^T)/2$ without changing the objective value. Thus hereafter, $\mathbf{Q}$ refers to the matrix with $\mathbf{Q}_{ij} = \frac{1}{2} \{I(X_i; C|X_j) + I(X_j; C|X_i)\}, i \neq j$ and $\mathbf{Q}_{ii} = I(X_i; C)$. It can be seen that our Hessian matrix $\mathbf{Q}$ embodies both the notions of relevancy and total redundancy. With this novel formulation, we have resolved several issues associated with the self-redundancy terms, as well as eliminating the need of introducing (and thus, tuning) the balancing factors α, β and the convexification parameter λ, as in the general $\mathcal{EQPFS}$ formulation.

We now present an interesting geometrical interpretation for the CMI criterion as follows. Besides relevancy, the global subset selection formulations $\mathcal{SS}_{\mathcal{CMJ}}$ and $\mathcal{QIP}_{\mathcal{CMJ}}$ favor features having large total pairwise conditional relevance. It is interesting to note that the quantity $d_C(X_i, X_j) = I(X_i; C|X_j) + I(X_j; C|X_i)$ could be regarded as a *distance measure* in the feature space. Sotoca and Pla [22] further claimed that this distance measure, named the conditional mutual information distance, is a proper metric, that is, it satisfies the triangle inequality[1]. The interpretation of $d_C(X_i, X_j)$ as a distance measure brings about an interesting insight on $\mathcal{SS}_{\mathcal{CMJ}}$, which can be rewritten as $\max_{\substack{\mathbb{S} \subset \mathbb{X} \\ |\mathbb{S}| = k}} \left\{ \sum_{X_i \in \mathbb{S}} I(X_i; C) + \frac{1}{2} \sum_{X_i, X_j \in \mathbb{S}} d_C(X_i, X_j) \right\}$. This criterion selects k features such that their total relevance and total pairwise distance is maximized. In other words, the criterion aims to choose a set of highly relevant representative features that also provide good coverage over the feature space, i.e. far apart from each-other in CMI distance.

As $\mathcal{QIP}_{\mathcal{CMJ}}$ is NP-hard, in the next sections we investigate efficient approximation techniques for solving this problem.

3.1 Global MI-based feature selection via spectral relaxation

We propose an efficient yet simple spectral relaxation technique for solving $\mathcal{QIP}_{\mathcal{CMJ}}$. We shall relax $\mathcal{QIP}_{\mathcal{CMJ}}$ to the continuous domain, by dropping the integral 0–1 constraints which in fact cause NP-hardness, while keeping only the

[1]We recently pointed out that Sotoca and Pla's proof is flawed, and that the triangle inequality holds true under the Naïve Bayes assumption, i.e. all the features are independent given the class variable [24]. Whether the CMI distance is a proper metric in general is still an open problem.

norm constraint, resulting in

$$\max_{\mathbf{x}} \left\{ \mathbf{x}^T \mathbf{Q} \mathbf{x} \right\} \text{ s.t. } \|\mathbf{x}\| = \sqrt{k}, x_i \geq 0$$

$$\stackrel{\circ}{=} \mathcal{SPEC}_{\mathcal{CMI}} \quad : \quad \max_{\mathbf{x}} \left\{ \mathbf{x}^T \mathbf{Q} \mathbf{x} \right\} \text{ s.t. } \|\mathbf{x}\| = 1, x_i \geq 0 \quad (19)$$

where $\stackrel{\circ}{=}$ denotes equivalence in feature *ordering*, noting that replacing $\|\mathbf{x}\| = \sqrt{k}$ with $\|\mathbf{x}\| = 1$ only scales the solution by a multiplicative constant $1/\sqrt{k}$. The non-negativity constraints $x_i \geq 0$ ensure that the relaxed solution can be reasonably interpreted as feature 'weights'.

Without the non-negativity constraints $x_i \geq 0$, albeit being a non-convex problem in general, $\mathcal{SPEC}_{\mathcal{CMI}}$ admits a simple global solution which coincides with that maximizing the well-known Rayleigh quotient of the form $\frac{\mathbf{x}^T \mathbf{Q} \mathbf{x}}{\mathbf{x}^T \mathbf{x}}$. The solution to this problem is *any unit-norm eigenvector corresponding to the dominant eigenvalue of* $\mathbf{Q}$ [10]. At optimality, the dominant eigenvalue of $\mathbf{Q}$ is also the maximum objective value. When the entries in $\mathbf{Q}$ are all non-negative, as $I(X_i; C|X_j) \geq 0$, then we can prove the following result:

THEOREM 2. *If* $\mathbf{Q}_{ij} \geq 0 \ \forall i, j$ *then:*

(i) the optimal solution $\mathbf{x}^*$ *for* $\max_{\|\mathbf{x}\|=1} \left\{ \mathbf{x}^T \mathbf{Q} \mathbf{x} \right\}$ *must be sign-consistent, i.e. having all* x_i^**'s of the same sign.*

(ii) any dominant eigenvector of $\mathbf{Q}$ *must be sign-consistent.*

(iii) if there exists a dominant eigenvector $\mathbf{x}^*$ *having* $x_i^* > 0, \forall i$*, i.e. strictly positive, then its eigenvalue must be the unique dominant eigenvalue of* $\mathbf{Q}$*.*

PROOF. (i) Assume that $\mathbf{x}^*$ has mixed-sign components, as $\mathbf{Q}_{ij} \geq 0 \ \forall i, j$ the value of the quadratic form $\mathbf{x}^{*T} \mathbf{Q} \mathbf{x}^* = \sum_{i,j} x_i^* x_j^* \mathbf{Q}_{ij}$ can always be increased by assigning the same sign to all x_i^*'s (still satisfying $\|\mathbf{x}^*\| = 1$), contradicting the assumption that $\mathbf{x}^*$ is the globally optimal solution.

(ii) We shall note that the critical points and critical values of $\max_{\|\mathbf{x}\|=1} \left\{ \mathbf{x}^T \mathbf{Q} \mathbf{x} \right\}$ are respectively all the unit-norm eigenvectors of $\mathbf{Q}$ and their eigenvectors. In case $\mathbf{Q}$ has duplicate dominant eigenvalues, all their associated eigenvectors are globally optimal solution for $\max_{\|\mathbf{x}\|=1} \left\{ \mathbf{x}^T \mathbf{Q} \mathbf{x} \right\}$, and therefore must be sign-consistent, as per (i).

(iii) As $x_i^* > 0$, there cannot exist any other sign-consistent (dominant) eigenvector that is orthogonal to $\mathbf{x}^*$, thus its eigenvalue must be the unique dominant eigenvalue of $\mathbf{Q}$. □

In view of this result, we can use any unit-norm dominant eigenvector of $\mathbf{Q}$ with all non-negative entries as the solution to $\mathcal{SPEC}_{\mathcal{CMI}}$. As for the feature ranking purpose, features with higher weights x_i will appear higher in the ranking, i.e. more important features. It is noted that in the $\mathcal{SPEC}_{\mathcal{CMI}}$ formulation, the Hessian $\mathbf{Q}$ is not required to be positive semidefinite as in the $\mathcal{QPFS}$ formulation.

Example 1 revisited: we have

$$I(G, C|S) = I(G; C) - I(G; S) + I(G, S|C) = 0 \text{ bit}$$
$$I(S, C|G) = I(S; C) - I(G; S) + I(G, S|C) = 0.2864 \text{ bit}$$

The solution to the $\mathcal{SPEC}_{\mathcal{CMI}}$ formulation

$$\max_{\|\mathbf{x}\|=1, x_i \geq 0} \left\{ \mathbf{x}^T \begin{bmatrix} 1 & 0.2864/2 \\ 0.2864/2 & 0.7136 \end{bmatrix} \mathbf{x} \right\} \quad (20)$$

is $\mathbf{x}^* = [0.92, 0.38]^T$, that is, smoking S (weight 0.92) is ranked higher than coughing G (weight 0.38).

3.2 Global MI-based feature selection via semi-definite programming

In this section, we investigate another strategy for solving the integer quadratic programming problem $\mathcal{QIP}_{\mathcal{CMI}}$, via semi-definite programming. Semi-definite relaxation has recently gained increasing interest as an effective approximation tool for solving hard combinatorial problems. This significant interest was sparked by the seminal work of Goemans and Williamson [11] in approximating the NP-hard max-cut problem in graph theory,

$$\mathcal{MAXCUT} : \max_{\mathbf{x}} \{ \mathbf{x}^T \mathbf{L} \mathbf{x} \} \quad \text{s.t. } \mathbf{x} \in \{-1, +1\}^n \quad (21)$$

where $\mathbf{L}$ is the graph Laplacian matrix. As semidefinite programming (SDP) is known to generate tighter approximation for $\mathcal{MAXCUT}$ over spectral relaxation, here we are interested in seeing whether for the $\mathcal{QIP}_{\mathcal{CMI}}$ problem, SDP can significantly improve over spectral relaxation as presented above. In order to employ the semidefinite relaxation technique in [11], we first transform the binary 0–1 problem $\mathcal{QIP}_{\mathcal{CMI}}$ into a bipolar $-1/+1$ problem similar to $\mathcal{MAXCUT}$, via the transformation $x_i = \frac{y_i + 1}{2}$, resulting in

$$\max_{\mathbf{y}} \quad \left\{ \frac{1}{4} (\mathbf{y} + \mathbf{1})^T \mathbf{Q} (\mathbf{y} + \mathbf{1}) \right\} \quad (22)$$
$$\text{s.t.} \quad \mathbf{y} \in \{-1, 1\}^n, (\mathbf{y} + \mathbf{1})^T \mathbf{I} (\mathbf{y} + \mathbf{1}) = 4k$$

where $\mathbf{1}_{n \times 1}$ is the vector of all 1's. Note that we could rewrite the norm constraint $\|\mathbf{x}\| = \sqrt{k}$ as $\mathbf{x}^T \mathbf{I} \mathbf{x} = k$ where $\mathbf{I}_{n \times n}$ is the identity matrix, hence $(\mathbf{y} + \mathbf{1})^T \mathbf{I} (\mathbf{y} + \mathbf{1}) = 4k$. Since the problem (22) is not in a homogeneous quadratic form, we shall transform it back to an equivalent homogeneous form via simply introducing an additional dummy variable $y_0 = 1$ (the variable expansion trick [27]), i.e. $\mathbf{y} = \{y_0 \equiv 1, y_1, \ldots, y_n\}$, resulting in

$$\max_{\mathbf{y}} \quad \mathbf{y}^T \widehat{\mathbf{Q}} \mathbf{y} \quad (23)$$
$$\text{s.t.} \quad y_0 = 1, \mathbf{y} \in \{-1, 1\}^{n+1}, \mathbf{y}^T \widehat{\mathbf{I}} \mathbf{y} = 4k$$

where $\widehat{\mathbf{Q}}_{(n+1) \times (n+1)} = \begin{bmatrix} \mathbf{1}^T \mathbf{Q} \mathbf{1} & \mathbf{1}^T \mathbf{Q} \\ \mathbf{Q} \mathbf{1} & \mathbf{Q} \end{bmatrix}$ and $\widehat{\mathbf{I}}_{(n+1) \times (n+1)} = \begin{bmatrix} \mathbf{1}^T \mathbf{I} \mathbf{1} & \mathbf{1}^T \mathbf{I} \\ \mathbf{I} \mathbf{1} & \mathbf{I} \end{bmatrix}$. We further note that the constraint $y_0 = 1$ could also be relaxed to $y_0 \in \{-1, 1\}$. This is because homogeneous quadratic problems are symmetric in $\mathbf{y}$ and $-\mathbf{y}$, therefore if $\mathbf{y}^*$ is optimal, then $-\mathbf{y}^*$ will also be optimal, and we simply need to pick the solution with $y_0^* = 1$ as the final solution. Now, note that the quadratic form $\mathbf{y}^T \widehat{\mathbf{Q}} \mathbf{y}$ can also be rewritten as $\widehat{\mathbf{Q}} \bullet \mathbf{y}\mathbf{y}^T$, where $\mathbf{U} \bullet \mathbf{V} = \sum_{i,j} U_{ij} V_{ij}$, we arrive at:

$$\max_{\mathbf{y}} \quad \widehat{\mathbf{Q}} \bullet \mathbf{y}\mathbf{y}^T \quad (24)$$
$$\text{s.t.} \quad \mathbf{y} \in \{-1, 1\}^{n+1}, \widehat{\mathbf{I}} \bullet \mathbf{y}\mathbf{y}^T = 4k$$

We next substitute $\mathbf{Y} = \mathbf{y}\mathbf{y}^T$, noting that an arbitrary matrix could only be factorized as such iff $\mathbf{Y} \succeq 0$, i.e. $\mathbf{Y}$ is positive semidefinite, and $\text{rank}(\mathbf{Y}) = 1$. Also note that for $y_i \in \{-1, 1\}$ we have $y_i . y_i = 1 \Leftrightarrow \text{diag}(\mathbf{Y}) = \mathbf{1}$, we arrive at

$$\max_{\mathbf{Y}} \quad \widehat{\mathbf{Q}} \bullet \mathbf{Y} \quad (25)$$
$$\text{s.t.} \quad \widehat{\mathbf{I}} \bullet \mathbf{Y} = 4k, \text{diag}(\mathbf{Y}) = \mathbf{1}, \mathbf{Y} \succeq 0, \text{rank}(\mathbf{Y}) = 1$$

Until now we have not yet gained any computational advantage, as the problem (25) is still exactly equivalent to

the NP-hard $\mathcal{QIP}_{\mathcal{CMI}}$ problem. The specific constraint that causes NP-hardness in this case is the rank-1 constraint, since without it, the following problem can be solved to optimality in polynomial time via semidefinite programming [18]:

$$\mathcal{SDP}_{\mathcal{CMI}}: \max_{\mathbf{Y}} \quad \widehat{\mathbf{Q}} \bullet \mathbf{Y} \qquad (26)$$
$$\text{s.t.} \quad \widehat{\mathbf{I}} \bullet \mathbf{Y} = 4k, \text{diag}(\mathbf{Y}) = \mathbf{1}, \mathbf{Y} \succeq 0$$

After solving $\mathcal{SDP}_{\mathcal{CMI}}$ we need to recover the discrete $\{-1, +1\}$ solution to (23), a process known as rounding. Herein, we simply adapt the random projection rounding technique proposed in [11], with 100 random projections. In each projection, the top k features are selected as the ones with corresponding rows $\mathbf{Y}_{i\cdot}$ having largest cosine similarity to a randomly picked vector uniformly distributed on the unit hypersphere in $\mathbb{R}^{n+1}$. Finally, the random projection that results in largest value for the original $\mathcal{QIP}_{\mathcal{CMI}}$ problem is selected. Interested readers are referred to [11] for these details.

3.3 Complexity analysis

For all methods, generally there will be time needed for computing the similarity matrix and the time needed for ranking the features. The time for computing MI quantities, such as $I(X_i; X_j)$ and $I(X_i; C|X_j)$ comprises mainly $O(d)$ time for computing the joint probability table. Thus, the time for computing the similarity matrix is $O(n^2 d)$. The ranking time complexity for MRMR, $\mathcal{SPECC}_{\mathcal{CMI}}$ and $\mathcal{SDP}_{\mathcal{CMI}}$ is provided in Table 1. The dominant time component for MRMR and $\mathcal{SPECC}_{\mathcal{CMI}}$ is in fact, for computing the similarity matrix, rather than ranking. In terms of ranking time, $\mathcal{SPECC}_{\mathcal{CMI}}$ is significantly less expensive than $\mathcal{QPFS}$, while $\mathcal{SDP}_{\mathcal{CMI}}$ is the most expensive.

Table 1: Ranking complexity in the number of features n.

Method	MRMR	$\mathcal{SPECC}_{\mathcal{CMI}}$	$\mathcal{QPFS}$	$\mathcal{SDP}_{\mathcal{CMI}}$
Complexity	$O(n^2)$	$O(n^2)$	$O(n^3)$	$O(n^{4.5})$

It is noted that greedy algorithms, such as MRMR, fill the similarity matrix gradually and could be stopped at any point to produce a partial ranking. In data mining and knowledge discovery, it is also often desirable to produce a complete ranking of all features. Indeed, while the top ranking features are important for building accurate classifiers, features with low ranks are important for understanding the data generating process. Such knowledge could be used, for example, to improve the data collecting process, where the least important features could be omitted from data collection. Also, a domain expert may be interested in studying how a feature of interest is ranked compared to others, in such case a complete ranking of all features is required.

3.4 Global feature selection for large data

For large data, computing the kernel-like matrix $\mathbf{Q}$ itself becomes expensive. Herein, we investigate a strategy to reduce this cost via low-rank approximation for $\mathbf{Q}$, in particular via the Nyström method. Nyström based methods for large-scale data analysis have been successfully applied on numerous problems in the pattern recognition and machine learning literature [9, 15]. Without loss of generality, we can assume $\mathbf{Q}$ in the $\mathcal{SPECC}_{\mathcal{CMI}}$ formulation (19) to be positive semi-definite. Indeed

$$\mathcal{SPECC}_{\mathcal{CMI}}: \underset{\|\mathbf{x}\|=1, x_i \geq 0}{\arg\max} \left\{\mathbf{x}^T \mathbf{Q} \mathbf{x}\right\} \equiv \underset{\|\mathbf{x}\|=1, x_i \geq 0}{\arg\max} \left\{\mathbf{x}^T (\mathbf{Q} + \lambda \mathbf{I}) \mathbf{x}\right\}$$

where λ can be chosen as a sufficiently large positive constant without affecting the ranking. Nyström method approximates the positive semi-definite $\mathbf{Q}$ as

$$\widetilde{\mathbf{Q}} = \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{B}^T & \mathbf{B}^T \mathbf{A}^{-1} \mathbf{B} \end{bmatrix}$$

using only a subset of $p = \gamma n$ rows of $\mathbf{Q}$, namely those comprising $[\mathbf{A}_{p \times p} \ \mathbf{B}_{p \times (n-p)}]$, where the rows are usually randomly sampled without replacement and $0 < \gamma < 1$ is the Nyström sampling rate. A useful characteristic of the Nyström approximation is that the approximated solution to the $\mathcal{SPECC}_{\mathcal{CMI}}$ formulation, namely the dominant eigenvector of $\widetilde{\mathbf{Q}}$, could be computed exactly from submatrices of much smaller size, without explicitly evaluating the block $\mathbf{B}^T \mathbf{A}^{-1} \mathbf{B}$ and fill in $\widetilde{\mathbf{Q}}$. This could be very useful for situations where the number of features is large, such that merely storing $\widetilde{\mathbf{Q}}_{n \times n}$ could already be a problem. Let $\mathbf{A}^{\frac{1}{2}}$ denote the symmetric positive definite square root of $\mathbf{A}$, define $\widehat{\mathbf{A}}_{p \times p} = \mathbf{A} + \mathbf{A}^{-\frac{1}{2}} \mathbf{B} \mathbf{B}^T \mathbf{A}^{-\frac{1}{2}}$ then the dominant eigenvector of $\widetilde{\mathbf{Q}}$ is simply

$$\sigma^{-\frac{1}{2}} \begin{bmatrix} \mathbf{A} \\ \mathbf{B}^T \end{bmatrix} \mathbf{A}^{-\frac{1}{2}} \mathbf{u}$$

where σ and $\mathbf{u}$ are the dominant eigenvalue and its associated eigenvector of $\widehat{\mathbf{A}}$ [9]. The complexity of the Nyström approximated solution is $O(\gamma n^2 d)$ for computing the similarity matrix and $O(\gamma n^2 + \gamma^2 n^2)$ for ranking. One remaining detail left is that although $\mathbf{Q}$ is entry-wise positive, it is not guaranteed that this property carries over to its approximation $\widetilde{\mathbf{Q}}$. Thus, $\widetilde{\mathbf{Q}}$ can have negative elements and as a results, its dominant eigenvector can have negative entries. In such cases, we induce a global ranking as follows. First, the problem is converted from a binary 0–1 problem to an equivalent bi-polar +1/-1 problem as in (22). Then a dummy variable $x_0 \equiv 1$, which is supposed to be always chosen, is included as in (23). The dominant eigenvector of $\begin{bmatrix} \mathbf{1}^T \widetilde{\mathbf{Q}} \mathbf{1} & \mathbf{1}^T \widetilde{\mathbf{Q}} \\ \widetilde{\mathbf{Q}} \mathbf{1} & \widetilde{\mathbf{Q}} \end{bmatrix}$ with the first entry (corresponding to x_0) being positive is chosen for feature ranking where the weights are sorted in descending order.

In [21] Nyström approximation was also applied for approximating the $\mathcal{QPFS}$ formulation (2). For $\mathcal{QPFS}$, a second level of approximation was further proposed, where the quadratic programming problem is approximated with one at a lower dimension, using only the largest eigenvalues of (the Nyström approximated) $\mathbf{H}$. As opposed to $\mathcal{QPFS}$, for the proposed $\mathcal{SPECC}_{\mathcal{CMI}}$ formulation, only one level of approximation, i.e. approximating $\mathbf{Q}$, is necessary. In general, Nyström approximation quality improves with increasing p. With a fixed sampling rate, approximation is better when there exists more redundancy in $\mathbf{Q}$, i.e. there are similar features.

4. EXPERIMENTAL EVALUATION

We perform a series of experiments to evaluate the efficiency and effectiveness of the two novel MI-based feature selection frameworks, namely $\mathcal{SPECC}_{\mathcal{CMI}}$ and $\mathcal{SDP}_{\mathcal{CMI}}$. First,

Table 2: Average time (in seconds) required for solving $\mathcal{SDP}_{\mathcal{EMI}}$ and $\mathcal{SPECC}_{\mathcal{EMI}}$ at different problem sizes (given pre-computed similarity matrices Q and H).

Dataset	#Features n	$\mathcal{SDP}_{\mathcal{EMI}}$	$\mathcal{SPECC}_{\mathcal{EMI}}$
Waveform	21	0.78 ± 0.07	0.005 ± 0.001
Promoter	57	1.18 ± 0.44	0.005 ± 0.001
Optdigits	64	1.59 ± 1.27	0.005 ± 0.001
Musk	166	3.38 ± 0.15	0.005 ± 0.000
Arrhythmia	257	9.46 ± 0.20	0.006 ± 0.001
Lung cancer	325	16.92 ± 0.73	0.007 ± 0.005
	$n > 700$	N/A	See Sec. 4.2

we compare $\mathcal{SPECC}_{\mathcal{EMI}}$ and $\mathcal{SDP}_{\mathcal{EMI}}$ in terms of their capability to approximate $\mathcal{QIP}_{\mathcal{EMI}}$, and draw the conclusion that $\mathcal{SPECC}_{\mathcal{EMI}}$ is the preferred approach. Second, we test $\mathcal{SPECC}_{\mathcal{EMI}}$ against $\mathcal{QPFS}$ in terms of scalability, and draw the conclusion that $\mathcal{SPECC}_{\mathcal{EMI}}$ is much more computationally scalable. Lastly, we compare $\mathcal{SPECC}_{\mathcal{EMI}}$ with other MI-based feature selection techniques on an extensive set of 15 small and medium sized real life data sets and 4 large datasets. The experiments were carried out on an Intel Core-i7 2.9Ghz PC with 16Gb of main memory.

4.1 $\mathcal{SPECC}_{\mathcal{EMI}}$ vs. $\mathcal{SDP}_{\mathcal{EMI}}$: a test of approximation tightness and scalability

We select several small size data sets in Table 4, namely Waveform ($n = 21$), Promoter ($n = 57$), Optdigits ($n = 64$), Musk ($n = 166$), Arrythmia ($n = 257$) and Lung cancer ($n = 325$) for this experiment. To solve the $\mathcal{SDP}_{\mathcal{EMI}}$ formulation, we employ the CVX toolbox for convex optimization [12], with the underlying solver being SDPT3 [23]. We set the number of features to be selected k to the range $[1, \min(n, 100)]$, thus in total there are 442 test cases. The average runtime comparison for $\mathcal{SDP}_{\mathcal{EMI}}$ and $\mathcal{SPECC}_{\mathcal{EMI}}$ (for the ranking phase) is reported in Table 2. For these small problems, the time required for $\mathcal{SPECC}_{\mathcal{EMI}}$ is negligible, while $\mathcal{SDP}_{\mathcal{EMI}}$ is orders of magnitude slower, but still acceptable. While $\mathcal{SDP}_{\mathcal{EMI}}$ running time does not seem a problem, it exhibits a large memory footprint. In fact, for problems with $n \sim 700$, CVX returns an out-of-memory error[2] on our PC. Note that the number of variables in the relaxed space for semidefinite programming is $O(\frac{1}{2}n^2)$. For example, with $n = 500$, CVX reports problem size of 125,751 variables and employs additionally $\sim$ 8Gb of memory.

We next compare $\mathcal{SDP}_{\mathcal{EMI}}$ and $\mathcal{SPECC}_{\mathcal{EMI}}$ in terms of the objective value of the original 0–1 problem $\mathcal{QIP}_{\mathcal{EMI}}$ in (18). Of all the 442 test cases, $\mathcal{SDP}_{\mathcal{EMI}}$ and $\mathcal{SPECC}_{\mathcal{EMI}}$ return different results in only 63 cases ($\sim$ 14%), within which $\mathcal{SDP}_{\mathcal{EMI}}$ 'wins' over in 46 cases ($\sim$ 73%). Thus it can be seen that $\mathcal{SDP}_{\mathcal{EMI}}$ tends to outperforms $\mathcal{SPECC}_{\mathcal{EMI}}$. This observation conforms well with previous studies, that semidefinite relaxation provides tighter approximation than spectral relaxation, as in other hard combinatorial problems such as graph max-cut. Nevertheless, the difference herein observed, if any, is often minor. More specifically, we have 58/63 cases in which the absolute relative difference, computed as $|obj_{SDP} - obj_{SPEC}|/obj_{SPEC}$ (where obj_{SDP} and obj_{SPEC} are the objective values of the $\mathcal{SDP}_{\mathcal{EMI}}$ and $\mathcal{SPECC}_{\mathcal{EMI}}$ approximated solution respectively), is $< 0.5\%$. Furthermore, a closer inspection reveals that in all cases, the feature sets differ in at most 2 features. For the rest 379/442 cases (86%), $\mathcal{SDP}_{\mathcal{EMI}}$ and $\mathcal{SPECC}_{\mathcal{EMI}}$ return identical objective value and identical feature sets. For the smallest Waveform data set, we also compute the optimal objective value found by exhaustive enumeration. In this case, the maximum relative difference between the optimal objective and that of $\mathcal{SPECC}_{\mathcal{EMI}}$ is only 0.07%, confirming the effectiveness of the two approximation schemes. For the other larger data sets, exhaustive enumeration is unacceptably slow, even with $k = 5$, ruling out this brute-force approach as a practical solution.

From this set of experiment, we draw the conclusion that, while semidefinite programming tends to generate tighter approximation, the difference is negligible. More importantly, the two techniques most often generate identical feature sets. In view of the fact that $\mathcal{SPECC}_{\mathcal{EMI}}$ is significantly simpler and more computationally efficient, we therefore promote $\mathcal{SPECC}_{\mathcal{EMI}}$ as the method of choice. In the next sections, we establish the efficiency and effectiveness of $\mathcal{SPECC}_{\mathcal{EMI}}$ against other popular MI-based feature selection approaches.

4.2 $\mathcal{SPECC}_{\mathcal{EMI}}$ vs. $\mathcal{QPFS}$: a test of scalability

To fix a concrete idea about how scalable and speedy dominant eigenvalue computation is, compared to quadratic convex optimization, we generate 10 random positive definite $\mathbf{Q}$ matrices for each size ranging from 1,000 to 30,000 (and also random relevancy vectors $\mathbf{f}$), and solve the $\mathcal{QPFS}$ and $\mathcal{SPECC}_{\mathcal{EMI}}$ problems using popular off-the-shelf solvers, specifically those provided by Matlab with default options. The average wall-clock time to solve these problems is provided in Table 3. Note that at $n > 16,000$, Matlab solver (`quadprog`) returned an out-of-memory error for $\mathcal{QPFS}$. On average, we observe that $\mathcal{QPFS}$ running time is two or more orders of magnitude slower than $\mathcal{SPECC}_{\mathcal{EMI}}$. For comparison, the ranking time for the incremental MRMR approach on the same similarity matrices is also reported in Table 3. With a carefully tuned implementation[3], MRMR outpaces $\mathcal{SPECC}_{\mathcal{EMI}}$ in running time, but practically this difference should not be a major concern.

In terms of practical implementation, the solution to $\mathcal{SPECC}_{\mathcal{EMI}}$ amounts to finding the dominant eigenvector of the Hessian matrix $\mathbf{Q}$. Algorithmically, this can be done as simply as repeatedly applying $\mathbf{Q}$ to any nondegenerate initial solution (the power method). In practice, dominant eigenvalue finding is a basic and efficient operation, which is built-in at the core of most, if not all, numerical packages. On the other hand, $\mathcal{QPFS}$ requires the solution to a quadratic convex optimization problem with linear constraints, which is arguably not always readily available as eigenvector computation. A further advantage of the $\mathcal{SPECC}_{\mathcal{EMI}}$ formulation over $\mathcal{QPFS}$ is that its solution via eigenvector decomposition is much more amenable to parallel computation, and can be implemented straightforwardly, readily exploiting the benefit of currently popular multi-core PC systems. Parallel implementation for quadratic and semi-definite programming on the other hand is an advanced research topic [26].

[2]Technically, we could employ more virtual memory using hard disk to circumvent memory shortage. But this results in a huge running time due to the high latency of hard disks.

[3]Implementation details can have considerable effects on the actual run time of the algorithms. Here, we employ our own optimized C++ implementation for MRMR to ensure its competitiveness, given that the same code implemented in Matlab could be 40-100 times slower.

Table 3: Average time (in seconds) required for ranking the features at different problem sizes (given pre-computed similarity matrices Q and H).

#Features	$\mathcal{QPFS}$	$\mathcal{SPEC}_{\mathcal{CMJ}}$	MRMR
1,000	0.81 ± 0.11	0.03 ± 0.01	0.01 ± 0.00
5,000	55.94 ± 5.14	0.81 ± 0.03	0.17 ± 0.02
10,000	417.22 ± 25.23	2.91 ± 0.26	0.68 ± 0.04
13,000	1026.73 ± 85.99	5.12 ± 0.45	1.17 ± 0.06
16,000	2012.63 ± 157.15	7.66 ± 0.89	1.97 ± 0.41
20,000	N/A	10.64 ± 0.30	2.63 ± 0.11
30,000	N/A	25.03 ± 1.42	6.05 ± 0.25

Table 4: Dataset summary. n: #features, d: #samples, #C: #classes, Error: average cross validation error rate (%) using all features.

Data	n	d	#C	Error	Source
NCI60	9996	60	10	43.3	[21] [20]
SRBCT	2308	84	4	1.2	[21]
Lung	325	73	7	12.3	[20] [7]
Colon	2000	62	2	17.7	[7]
Leukemia	7129	73	2	1.4	[7]
Lymphoma	4026	96	9	3.1	[7]
Promoter	57	106	2	16.0	[1]
Spambase	57	4601	2	9.5	[1]
Musk2	166	6598	2	4.9	[1]
Arrhythmia	257	430	2	21.6	[1]
Multi-features	649	2000	10	1.6	[1]
Waveform	21	5000	3	13.0	[1]
Optdigits	64	3823	10	1.8	[1]
Gisette	5000	6000	2	50.0	[1]
Madelon	500	2000	2	34.7	[1]

4.3 Small and medium data sets

We compare the proposed $\mathcal{SPEC}_{\mathcal{CMJ}}$ method with state of the art MI-based feature selection approaches on an extensive set of 15 well-known public datasets used in previous research [3, 14, 20, 21], covering a wide range of number of features, samples and classes. Feature selection methods are compared in terms of the average cross validation (CV) classification error rate on the range of 10 to 100 features in step of 1 (or 10 to n if $n < 100$). We employ 10-fold CV for datasets with number of samples $d \geq 100$ and leave-one-out CV otherwise. Following [14,21], the based classifier for most data sets is chosen as linear SVM (with the regularization parameter set to 1), except for the Gisette and Madelon datasets, where a 3-NN classifier was used following [3]. Details of the datasets used are given in Table 4. Continuous features were discretized using Fayyad and Irani's minimum description length (MDL) method [8]. Feature selection was done on discretized data, while classification was performed on the original feature space.

Apart from the feature selection approaches mentioned herein, namely MRMR, EMRMR, $\mathcal{QPFS}$ and $\mathcal{SPEC}_{\mathcal{CMJ}}$, we also compare our approach with other well-known MI based methods, namely maximum relevance (maxRel), mutual information quotient (MIQ) [7] and conditional infomax feature extraction (CIFE) [16]. The connections between these methods are presented in great details in [3]. We note that [3] recommends the so-called joint mutual information (JMI), $\max_{X_i} \sum_{X_j \in \mathbb{S}} I(X_i, X_j; C)$, as the criterion of choice, for providing 'the best tradeoff in terms of accuracy, stability and flexibility with small samples'. We note that the JMI criterion is in fact exactly equivalent to the EMRMR criterion presented herein, and also the 'average-CMIM' criterion in [14]. For $\mathcal{QPFS}$ the balancing factor α was set as recommended in [21]. $\mathcal{SPEC}_{\mathcal{CMJ}}$ requires no parameter tuning. We implemented and optimized the codes for all the above methods in Malab/C++, which are made publicly available via our website.

The experimental results for all methods are presented in Table 5. In order to summarize the statistical significance of the findings, as in [14], we employ the one sided paired t-test at 5% significance level to test the hypothesis that $\mathcal{SPEC}_{\mathcal{CMJ}}$ or a compared method performs significantly better than the other. Overall, we found the proposed $\mathcal{SPEC}_{\mathcal{CMJ}}$ framework to perform strongly against other popular MI-based criteria for feature selection. In particular, $\mathcal{SPEC}_{\mathcal{CMJ}}$ consistently outperforms the alternative global formulation $\mathcal{QPFS}$. Of the incremental methods, $\mathcal{SPEC}_{\mathcal{CMJ}}$ strongly outperforms maxRel, MIQ and CIFE. On the other hand, MRMR and EMRMR are two leading local algorithms, being only narrowly behind $\mathcal{SPEC}_{\mathcal{CMJ}}$. The bold entries in Table 5 indicate the best performing algorithms (in terms of the average error rate and its standard deviation)—although the difference compared with other methods might not necessarily be statistically significant. The distribution of the 'bold entries' seems to suggest that no algorithm is universally dominant—a reminiscence of the no free lunch theorem for machine learning [25]. Nevertheless, from a practical viewpoint, for the supervised feature selection problem, one can, and should, try multiple feature selection strategies and use, e.g. the cross validation error rate as a guidance to choose the final set of features. From this perspective, we propose that $\mathcal{SPEC}_{\mathcal{CMJ}}$ is a valuable addition to the current literature on feature selection.

4.4 Large data

We employ four datasets from the handwritten Chinese character database [17] as detailed in Table 6. These data are characterized by a large number of training samples, and especially a very large number of classes, making classification a challenging task. Indeed, the SVM implementation we employed, namely LibSVM [4], does not scale very well with this application where it has to train a large number (~ 3700) of one-versus-all classifiers. We therefore resort to a much simpler and more computationally efficient nearest class mean (NCM) classifier [19]. We select the top two performing greedy algorithms, namely MRMR and EMRMR, from section 4.3 together with $\mathcal{QPFS}$ and $\mathcal{SPEC}_{\mathcal{CMJ}}$ for this test. In addition, we test the effectiveness of Nyström approximation with both $\mathcal{SPEC}_{\mathcal{CMJ}}$ and $\mathcal{QPFS}$. We train the classifier on the train data and test accuracy is estimated on the separate test data. Since the number of samples is large, we expect this performance indicator to be representative. On these data, the MDL algorithm [8] binarizes most features, i.e. discretizing to only 2 states. Observing that the very large number of samples can support a finer discretization, we therefore also discretize the data to 5 and 10 equal-frequency bins. While finding the optimal discretization strategy is beyond the scope of this paper, we summarize our finding as follows: at lower number of bins, i.e. 2 and 5, methods that are based on the MI such as MRMR and $\mathcal{QPFS}$ outperform methods based on conditional MI, such that $\mathcal{SPEC}_{\mathcal{CMJ}}$ and EMRMR. On the other hand,

Table 5: Cross validation error rate comparison of $\mathcal{SPEC}_{\mathcal{CMJ}}$ against other methods. W: win (+), T: tie (=), L: loss (−) for $\mathcal{SPEC}_{\mathcal{CMJ}}$ against the compared method according to the 1-sided paired t-test.

Data	maxRel	MIQ	CIFE	MRMR	EMRMR*	$\mathcal{QPFS}$	$\mathcal{SPEC}_{\mathcal{CMJ}}$
Lung	14.2 ± 5.7 (+)	12.0±2.7 (+)	16.0±2.2 (+)	9.8±3.4 (+)	9.7±3.4 (+)	10.4±2.8 (+)	**9.4±2.5**
NCI60	35.1 ± 8.6 (+)	40.1±14.1 (+)	64.9±3.6 (+)	30.5±10.7 (=)	30.2±8.7 (−)	**28.1±9.4** (−)	31.3±8.4
Colon	12.8 ± 1.5 (=)	12.8±1.6 (=)	14.4±2.7 (+)	12.8±1.4 (=)	**12.0±1.0** (−)	13.2±1.4 (+)	12.7±1.2
Leukemia	3.1 ± 1.0 (+)	**1.0±1.6** (−)	5.0±1.0 (+)	2.4±0.8 (−)	2.5±0.6 (−)	3.0±1.0 (=)	2.9±1.0
Lymphoma	4.3 ± 2.5 (=)	6.0±5.0 (+)	16.6±2.9 (+)	4.1±2.1 (=)	**4.0±2.4** (=)	5.5±3.7 (+)	4.3±3.9
SRBCT	0.8 ± 1.1 (−)	2.1±3.5 (+)	11.8±3.7 (+)	0.6±1.2 (−)	0.9±1.2 (=)	**0.1±0.3** (−)	0.9±1.3
Promoter	8.7 ± 2.9 (=)	8.9±3.4 (=)	12.6±2.4 (+)	9.4±3.4 (+)	8.3±3.2 (=)	8.7±3.3 (=)	**8.3±3.1**
Spambase	11.5 ± 1.4 (+)	12.3±3.2 (+)	17.9±4.1 (+)	**11.3±1.5** (=)	11.4±1.5 (+)	12.0±1.9 (+)	**11.3±1.5**
Musk2	7.8 ± 1.8 (+)	**7.1±1.6** (−)	7.4±1.1 (=)	7.4±1.6 (=)	7.4±1.0 (=)	7.4±1.9 (=)	7.4±0.9
Arryth.	22.2 ± 1.0 (+)	24.0±3.5 (+)	24.6±1.8 (+)	22.3±1.5 (+)	22.2±0.8 (+)	22.8±2.0 (+)	**21.7±0.7**
Multifeat.	2.0 ± 0.9 (+)	3.2±2.8 (+)	2.8±0.7 (+)	**1.8±0.4** (=)	1.8±0.6 (=)	2.4±1.3 (+)	1.9±0.6
Optdigits	3.3 ± 2.5 (+)	3.3±2.6 (+)	3.4±2.5 (+)	**3.0±2.0** (=)	3.0±2.2 (=)	4.1±4.0 (+)	3.1±2.2
Waveform	13.9 ± 1.5 (=)	14.2±1.3 (+)	16.3±2.1 (+)	**13.7±1.0** (=)	13.7±1.1 (=)	**13.7±1.0** (=)	**13.7±1.0**
Gisette	7.8 ± 2.7 (+)	9.2±4.2 (+)	6.7±1.8 (+)	**6.0±2.5** (=)	6.4±2.4 (+)	8.0±2.3 (+)	6.1±2.2
Madelon	18.7 ± 3.2 (+)	37.4±3.7 (+)	17.5±3.4 (+)	28.2±2.2 (+)	16.0±3.1 (=)	23.3±3.5 (+)	**15.9±3.2**
#W/T/L:	10/4/1	11/2/2	14/1/0	4/9/2	4/8/3	9/4/2	

*Also equivalent to the JMI and ave-CMIM criteria, see [3, 14]

Table 6: Large dataset summary. n: #features, d: #samples, #C: #classes, Error: test error rate (%) using all features with NCM classifier.

Data	n	d (train)	d (test)	#C	Error
HWDB1.0	512	1,246,991	309,684	3,740	21.93
HWDB1.1	512	897,758	223,991	3,755	26.42
OLHWDB1.0	512	1,256,009	314,042	3,740	15.99
OLHWDB1.1	512	898,573	224,559	3,755	17.18

at higher number of bins, EMRMR and $\mathcal{SPEC}_{\mathcal{CMJ}}$ performs slightly better than $\mathcal{QPFS}$ and MRMR. Our hypothesis is that a larger number of bins can leverage the large number of samples such that higher-dimensional mutual information quantities, such as the conditional MI, could be estimated at greater resolution. The test error rate on sets of up to 200 features on the 10-bin discretized data are reported in Figure 2(a-d). It is noted that $\mathcal{SPEC}_{\mathcal{CMJ}}$+Nyström at a sampling rate of $\gamma = 0.2$ perform remarkably well on the HWDB1.0 and HWDB1.1 datasets, in fact better than $\mathcal{SPEC}_{\mathcal{CMJ}}$—a somewhat intriguing observation, while being slightly better than $\mathcal{QPFS}$ on the OLHWDB1.0 and OLHWDB1.1 data. The effect of different sampling rate for $\mathcal{SPEC}_{\mathcal{CMJ}}$+Nyström on the OLHWDB1.1 is presented in Fig. 2(e). The wall clock execution time of all algorithms on each data set is presented in Fig. 2(f). It is observed that methods that make use of the conditional MI such as $\mathcal{SPEC}_{\mathcal{CMJ}}$ and EMRMR are more expensive than methods that make use of the MI such as $\mathcal{QPFS}$ and MRMR, mainly due to the fact that computing conditional MI is more time consuming. $\mathcal{SPEC}_{\mathcal{CMJ}}$ and EMRMR admit similar execution time, while $\mathcal{QPFS}$ and MRMR admit similar execution time. Nyström approximation significantly reduces the execution time for both $\mathcal{SPEC}_{\mathcal{CMJ}}$ and $\mathcal{QPFS}$.

5. CONCLUSION

In this paper, we have introduced a novel global optimization framework for the mutual information based feature selection problem. Our criterion for optimization is formulated based on the conditional mutual information, an information theoretic quantity which neatly captures feature relevancy, redundancy as well as class-conditional redundancy, leading to a neat homogeneous quadratic optimization criterion. We have demonstrated that this global formulation can be efficiently solved via spectral relaxation, admitting a very simple numerical solution. We also compared the spectral relaxation approach with the more sophisticated semi-definite relaxation, and establish that spectral relaxation returns mostly identical solution at a much cheaper computational cost. Compared to the local formulations MRMR and EMRMR, the global formulations can overcome the issue of local minima faced by local greedy schemes. Compared to the alternative global $\mathcal{QPFS}$ formulation, our new $\mathcal{SPEC}_{\mathcal{CMJ}}$ framework naturally resolves several theoretical issues associated with the previous global $\mathcal{QPFS}$ formulation. Moreover, $\mathcal{SPEC}_{\mathcal{CMJ}}$ admits a significantly simpler and much more efficient global solution, yet without any strict condition, such as positive definiteness, on the Hessian matrix.

Acknowledgments: This work is supported by the Australian Research Council via grant number FT110100112.

6. REFERENCES

[1] K. Bache and M. Lichman. UCI machine learning repository, 2013.
[2] R. Battiti. Using mutual information for selecting features in supervised neural net learning. *Neural Networks, IEEE Transactions on*, 5(4):537–550, 1994.
[3] G. Brown, A. Pocock, M.-J. Zhao, and M. Luján. Conditional likelihood maximisation: A unifying framework for information theoretic feature selection. *J. Mach. Learn. Res.*, 13:27–66, Mar. 2012.
[4] C.-C. Chang and C.-J. Lin. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27, 2011. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.
[5] A. W. Chaovalitwongse, I. P. Androulakis, and P. M. Pardalos. Quadratic integer programming: complexity and equivalent forms quadratic integer programming: Complexity and equivalent forms. In C. A. Floudas and P. M. Pardalos, editors, *Encyclopedia of Optimization*, pages 3153–3159. 2009.
[6] H. Cheng, Z. Qin, W. Qian, and W. Liu. Conditional mutual information based feature selection. In *Knowledge Acquisition and Modeling*, pages 103–107, 2008.
[7] C. Ding and H. Peng. Minimum redundancy feature selection from microarray gene expression data. In *Bioinformatics Conference, 2003*, pages 523–528, 2003.
[8] U. M. Fayyad and K. B. Irani. Multi-interval discretization of continuous-valued attributes for classification learning. In *IJCAI*, pages 1022–1029, 1993.
[9] C. Fowlkes, S. Belongie, and J. Malik. Efficient spatiotemporal grouping using the nystrom method. In *CVPR 2001*, volume 1, pages I–231–I–238 vol.1, 2001.

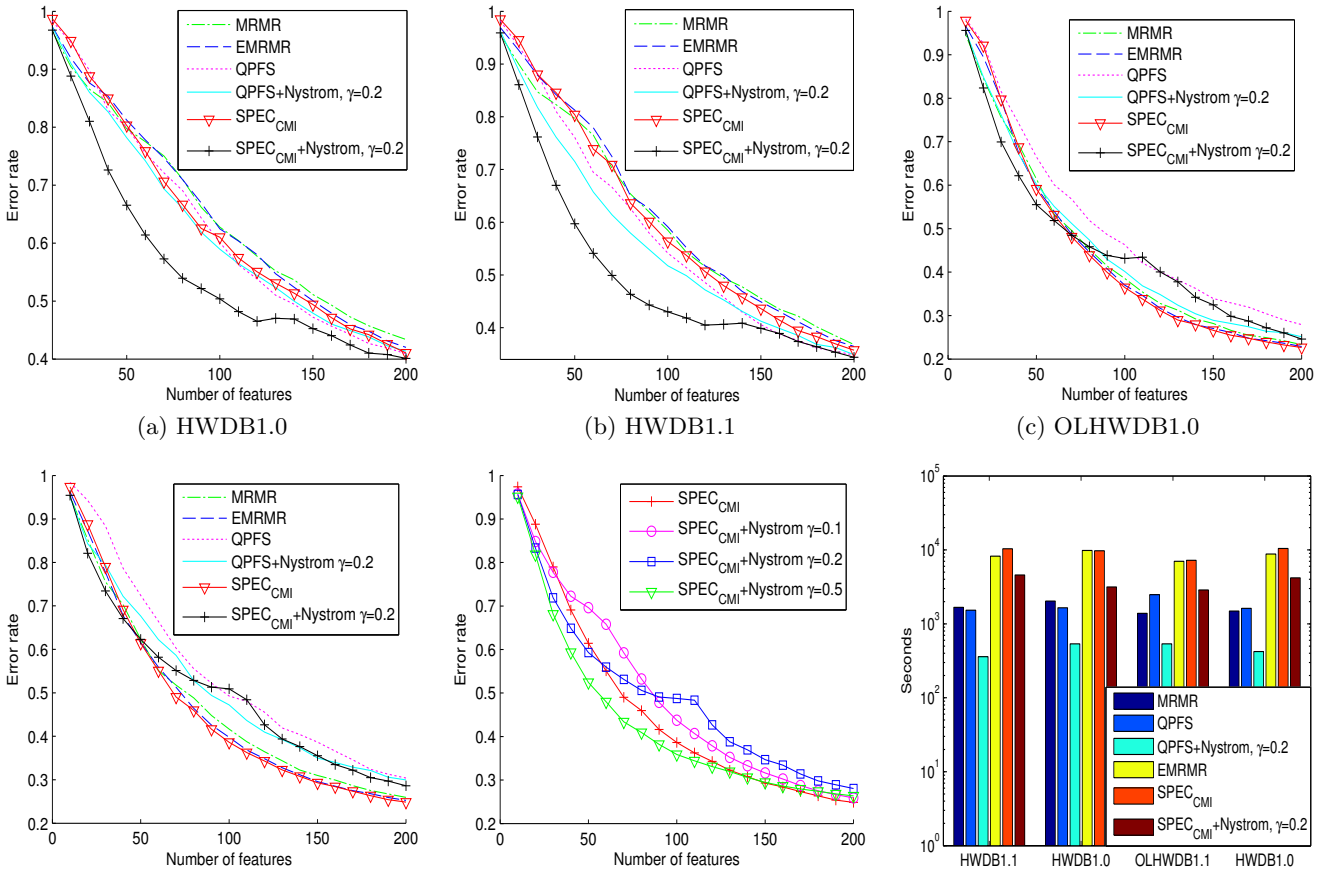


Figure 2: Test accuracy and execution time on the handwritten Chinese character database.

[10] J. Gallier. *Geometric Methods and Applications: For Computer Science and Engineering*. Texts in Applied Mathematics. Springer-Verlag GmbH, 2001.

[11] M. X. Goemans and D. P. Williamson. Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. *J. ACM*, 42(6):1115–1145, Nov. 1995.

[12] M. Grant and S. Boyd. CVX: Matlab software for disciplined convex programming, version 2.0 beta. http://cvxr.com/cvx, Sept. 2013.

[13] B. Guo and M. Nixon. Gait feature subset selection by mutual information. *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, 39(1):36–46, 2009.

[14] G. Herman, B. Zhang, Y. Wang, G. Ye, and F. Chen. Mutual information-based method for selecting informative feature sets. *Pattern Recognition*, 46(12):3315 – 3327, 2013.

[15] M. Li, J. T. Kwok, and B.-L. Lu. Making large-scale nyström approximation possible. In J. Fürnkranz and T. Joachims, editors, *ICML*, pages 631–638. Omnipress, 2010.

[16] D. Lin and X. Tang. Conditional infomax learning: an integrated framework for feature extraction and fusion. In *ECCV'06*.

[17] C.-L. Liu, F. Yin, D.-H. Wang, and Q.-F. Wang. Casia online and offline chinese handwriting databases. In *Document Analysis and Recognition (ICDAR), 2011 International Conference on*, pages 37–41, Sept 2011.

[18] Z.-Q. Luo, W.-K. Ma, A.-C. So, Y. Ye, and S. Zhang. Semidefinite relaxation of quadratic optimization problems. *Signal Processing Magazine, IEEE*, 27(3):20–34, 2010.

[19] T. Mensink, J. Verbeek, F. Perronnin, and G. Csurka. Distance-based image classification: Generalizing to new classes at near-zero cost. *IEEE TPAMI*, 35(11):2624–2637, Nov 2013.

[20] H. Peng, F. Long, and C. Ding. Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Trans. Pattern Anal. Mach. Intell.*, 27(8):1226–1238, 2005.

[21] I. Rodriguez-Lujan, R. Huerta, C. Elkan, and C. S. Cruz. Quadratic programming feature selection. *J. Mach. Learn. Res.*, 11:1491–1516, 2010.

[22] J. M. Sotoca and F. Pla. Supervised feature selection by clustering using conditional mutual information-based distances. *Pattern Recognition*, 43(6):2068–2081, June 2010.

[23] K. C. Toh, M. Todd, and R. H. Tütüncü. Sdpt3 – a matlab software package for semidefinite programming. *Optimization methods and software*, 11:545–581, 1999.

[24] N. Vinh and J. Bailey. Comments on supervised feature selection by clustering using conditional mutual information-based distances. *Pattern Recognition*, 46(4):1220 – 1225, 2013.

[25] D. H. Wolpert. The lack of a priori distinctions between learning algorithms. *Neural computation*, 8(7):1341–1390, 1996.

[26] M. Yamashita, K. Fujisawa, M. Fukuda, K. Nakata, and M. Nakata. Algorithm 925: Parallel solver for semidefinite programming problem having sparse schur complement matrix. *ACM Trans. Math. Softw.*, 39(1):6:1–6:22, Nov. 2012.

[27] Y. Zhang, S. Burer, and W. N. Street. Ensemble pruning via semi-definite programming. *J. Mach. Learn. Res.*, 7:1315–1338, Dec. 2006.