BIOINFORMATICS

Vol. 00 no. 00 2005
Pages 1–9

Optimized data fusion for K-means Laplacian Clustering

Shi Yu [1]*, Xinhai Liu [1,3], Léon-Charles Tranchevent [1], Wolfgang Glänzel [2], Johan A. K. Suykens [1], Bart De Moor [1] and Yves Moreau [1]

[1]Department of Electrical Engineering, Katholieke Universiteit Leuven.
[2]Centre for R & D Monitoring and Depart. of Managerial Economics, Strategy and Innovation, Katholieke Universiteit Leuven.
[3]College of Information Science and Engineering & ERCMAMT, Wuhan University of Science and Technology.

Received on XXXXX; revised on XXXXX; accepted on XXXXX

Associate Editor: XXXXXXX

ABSTRACT

Motivation: We propose a novel algorithm to combine multiple kernels and Laplacians for clustering analysis. The new algorithm is formulated on a Rayleigh quotient objective function and is solved as a bi-level alternating minimization procedure. Using the proposed algorithm, the coefficients of kernels and Laplacians can be optimized automatically.

Results: Three variants of the algorithm are proposed. The performance is systematically validated on two real-life data fusion applications. The proposed OKLC (Optimized Kernel Laplacian Clustering) algorithms perform significantly better than other methods. Moreover, the coefficients of kernels and Laplacians optimized by OKLC show some correlation with the rank of performance of individual data source. Though in our evaluation the K values are predefined, in practical studies, the optimal cluster number can be consistently estimated from the eigenspectrum of the combined kernel Laplacian matrix.

Availability: The MATLAB code of algorithms implemented in this paper is downloadable from
http://homes.esat.kuleuven.be/∼sistawww/bioi/syu/oklc.html.

Contact: shiyu@uchicago.edu

1 INTRODUCTION

Clustering is a fundamental problem in unsupervised learning and a number of different algorithms and methods have emerged over the years. K-means and spectral clustering are two popular methods for clustering analysis. K-means (KM) is proposed to cluster attribute-based data into K numbers of clusters with the minimal distortion (Bishop (2006); Duda *et al*. (2001)). Another well known method, spectral clustering (SC) (Ng *et al*. (2001); Shi and Malik (2000)), is also widely adopted in many applications. Unlike KM, SC is specifically developed for graphs, where the data samples are represented as vertices connected by non-negatively weighted undirected edges. The problem of clustering on graphs belongs to another paradigm than the algorithms based on the distortion measure. The goal of graph clustering is to find partitions on the graph such that the edges between different groups have a very low weight (von Luxburg (2007)). To model this, different objective functions are adopted and the typical criteria include the RatioCut (Hagen and Kahng (1992)), the normalized cut (Shi and Malik (2000)), and many others. To solve these objectives, the discrete constraint of the clustering indicators is usually relaxed to real values; thus, the approximated solution of spectral clustering can be obtained from the eigenspectrum of the graph Laplacian matrix. Many investigations (e.g., Dhillon *et al*. (2004)) have shown the connection between KM and SC. Moreover, in practical applications, the weighted similarity matrix is often used interchangeably as the kernel matrix in KM or the adjacency matrix in SC.

Recently, a new algorithm, Kernel Laplacian clustering (KL), is proposed to combine a kernel and a Laplacian simultaneously in clustering analysis (Wang *et al*. (2009)). This method combines the objectives of KM and SC in a quotient trace maximization form and solves the problem by eigen-decomposition. KL is shown to empirically outperforming KM and SC on real data sets. This straightforward idea is useful to solve many practical problems, especially those pertaining to combine attribute-based data with interaction-based networks. For example, in web analysis and scientometrics, the combination of text mining and bibliometrics has become a standard approach in clustering science or technology fields towards the detection of emerging fields or hot topics (Liu *et al*. (2010)). In bioinformatics, protein-protein interaction network and expression data are two of the most important sources used to reveal the relevance of genes and proteins with complex diseases. Conventionally, the data is often transformed into similarity matrices or interaction graphs, then consequently clustered by KM or SC. In KL, the similarity based kernel matrix and the interaction based Laplacian matrix are combined, which provides a novel approach to combine heterogeneous data structures in clustering analysis.

Our preliminary experiments show that when using KL to combine a single kernel and a single Laplacian, its performance strongly depends on the quality of the kernel and the Laplacian, which results in a model selection problem to determine the optimal settings of the kernel and the Laplacian. To perform model selection on unlabeled data is non-trivial because it is difficult to evaluate the models. To tackle the new problem, we propose a novel algorithm to

*to whom correspondence should be addressed, currently affiliated to the Department of Medicine, the University of Chicago

© Oxford University Press 2005.

incorporate multiple kernels and Laplacians in KL clustering. Our recent work proposes a method to integrate multiple kernel matrices in kernel k-means clustering (Yu *et al.* "Optimized data fusion for kernel K-means clustering", submitted). The main contribution of the present work lies in the additive combination of multiple kernels and Laplacians, moreover, the coefficients assigned to the kernels and the Laplacians are optimized automatically. This paper presents the mathematical derivations of the additive integration form of kernels and Laplacians. The optimization of coefficients and clustering are achieved via a solution based on bi-level alternating minimization (Csiszar and Tusnady (1984)). We validate the proposed algorithm on heterogeneous data sets taken from two real applications, where the advantage and reliability of the proposed method are systematically compared and demonstrated.

2 METHODS

2.1 Combine kernel and Laplacian as generalized Ralyeigh quotient for clustering

We first briefly review the KL algorithm proposed by (Wang *et al*. (2009)). All the mathematical symbols used in the paper are consistent and their representations are listed in **Supplementary Material 1**. Let us denote X as an attribute data set and W as a graph affinity matrix, both of them are representations of the same sets of samples. The objective of the KL integration to combine X and W for clustering can be defined as

$$J_{KL} = \kappa J_{SC} + (1-\kappa) J_{KM}, \tag{1}$$

where J_{SC} and J_{KM} are respectively the objectives of SC and KM clustering, $\kappa \in [0,1]$ is a coefficient adjusting the effect of the two objectives. Let us denote $A \in \mathbb{R}^{N\times K}$ as the weighted scalar cluster membership matrix, given by

$$A_{ab} = \begin{cases} \frac{1}{\sqrt{n_b}} & \text{if } \vec{x}_a \in C_b \\ 0 & \text{if } \vec{x}_a \notin C_b, \end{cases} \tag{2}$$

where n_b is the number of data points belonging to cluster C_b and $A^T A = I_K$, where I_K denotes a $K \times K$ identity matrix. Let's denote D as the diagonal matrix whose (a,a) entry is the sum of the entries of row a in the affinity matrix W. The *normalized Laplacian matrix* (von Luxburg (2007)) is given by

$$\tilde{L} = I - D^{-\frac{1}{2}} W D^{-\frac{1}{2}}. \tag{3}$$

The objective of normalized cut based SC is formulated as

$$\underset{A}{\text{minimize}} \ \text{trace}\left(A^T \tilde{L} A\right). \tag{4}$$

As discussed in the literature (Duda *et al.* (2001); Bishop (2006); Hastie *et al*. (2009)), if the data X has zero sample means, the objective of the KM is given by

$$\underset{A}{\text{maximize}} \ \text{trace}(A^T X^T X A). \tag{5}$$

We further generalize (5) by applying the feature map $\phi(\cdot) : \mathbb{R} \to \mathcal{F}$ on X, then the centered data in $\mathcal{F}$ is denoted as X^Φ, given by

$$X^\Phi = [\phi(\vec{x}_1) - \vec{\mu}^\Phi, \phi(\vec{x}_2) - \vec{\mu}^\Phi, ..., \phi(\vec{x}_N) - \vec{\mu}^\Phi], \tag{6}$$

where $\phi(\vec{x}_i)$ is the feature map applied on the column vector of the a-th data point in $\mathcal{F}$, $\vec{\mu}^\Phi$ is the global mean in $\mathcal{F}$. The inner product $X^T X$ in (5) can be combined using the kernel trick $G(\vec{x}_u, \vec{x}_v) = \phi(\vec{x}_u)^T \phi(\vec{x}_v)$, where $G(\cdot,\cdot)$ is a Mercer kernel. We denote G_c as the centered kernel matrix as $G_c = PGP$, where P is the centering matrix $P = I_N - (1/N)\vec{1}_N^T$, G is the kernel matrix, I_N is the $N \times N$ identity matrix, $\vec{1}_N$ is a column vector of N ones. Without loss of generality, the K-means objective in (5) can be equivalently written as

$$\underset{A}{\text{maximize}} \ \text{trace}(A^T G_c A). \tag{7}$$

Then the objective of KL integration becomes

$$\begin{aligned} &\underset{A}{\text{minimize}} \ \text{trace}\left(A^T \tilde{L} A\right) - (1-\kappa)\,\text{trace}\left(A^T G_c A\right) \\ &\text{subject to} \ \ A^T A = I_K, \\ &\qquad 0 \le \kappa \le 1. \end{aligned} \tag{8}$$

To solve the optimization problem without tuning the ad hoc parameter κ, Wang *et al.* formulate it as a trace quotient of the two components (Wang *et al*. (2009)). The trace quotient is then further relaxed as a maximization of quotient trace, given by

$$\begin{aligned} &\underset{A}{\text{maximize}} \ \text{trace} \ (A^T \tilde{L} A)^{-1} (A^T G_c A) \\ &\text{subject to} \ \ A^T A = I_K. \end{aligned} \tag{9}$$

The problem in (9) is a generalized Rayleigh quotient and the optimal solution A^* is obtained in the generalized eigenvalue problem. To maximize this objective, A^* is approximated as the largest K eigenvectors of $\tilde{L}^+ G_c$, where $\tilde{L}^+$ is the pseudo inverse of $\tilde{L}$ (Wang *et al*. (2009)).

2.2 Combine kernel and Laplacian as additive models for clustering

As discussed, the original KL algorithm is proposed to optimize the generalized Rayleigh quotient objective. In ths paper, we propose an alternative integration method using a different notation of Laplacian (von Luxburg (2007)), $\hat{L}$, given by

$$\hat{L} = D^{-1/2} W D^{-1/2}, \tag{10}$$

where D and W are defined the same as in (3). The objective of spectral clustering is equivalent to maximizing the term as

$$\underset{A}{\text{maximize}} \ \text{trace}(A^T \hat{L} A). \tag{11}$$

Therefore, the objective of the KL integration can be rewritten in an additive form, given by

$$\begin{aligned} &\underset{A}{\text{maximize}} \ \text{trace}\left\{\kappa A^T \hat{L} A + (1-\kappa) A^T G_c A\right\} \\ &\text{subject to} \ \ A^T A = I_k, \\ &\qquad 0 \le \kappa \le 1, \end{aligned} \tag{12}$$

where A, G_c are defined the same as in (8), κ is the free parameter to adjust the effect of kernel and Laplacian in KL integration. If κ is pre-defined, (12) is a Rayleigh quotient problem and the optimal A^* can be obtained from eigenvalue decomposition, known as the spectral relaxation (Ding and He (2004)). Therefore, to maximize this objective, we denote $\mathbf{\Omega} = \kappa\tilde{L} + (1-\kappa)G_c$ thus A^* is solved as the dominant K eigenvectors of $\mathbf{\Omega}$.

In section 2.1 and section 2.2, two different methods are investigated to integrate a single Laplacian matrix with a single kernel matrix for clustering, where the main difference is to either optimize the cluster assignment affinity matrix A as a generalized Rayleigh quotient (ratio model) or as a Rayleigh quotient (additive model). The main advantage of the ratio based solution is to avoid tuning the parameter κ. However, since the main contribution of this paper is to optimize the combination of multiple kernels and Laplacians, the coefficients assigned on each kernel and Laplacian still need to be optimized. Moreover, the optimization of the additive integration model is computationally simpler than optimizing the ratio based model. Therefore, in the following sections we will focus on extending the additive KL integration to multiple sources.

2.3 Clustering by multiple kernels and Laplacians: an additive model solved with bi-level optimization

Let us denote a set of graphs as H_i, $i \in \{1, ..., r\}$, all having N vertices, and a set of Laplacians $\hat{L}_i$ constructed from H_i as (10). Let us also denote a set of centered kernel matrices as G_{cj}, $j \in \{1, ..., s\}$ with N samples. To extend (12) by incorporating multiple kernels and Laplacians for clustering, we propose a strategy to learn their optimal weighted convex linear combinations. The extended objective function is then given by

$$\boxed{\text{Q1:}} \quad \underset{A,\vec{\theta}}{\text{maximize}} \ \mathcal{J}_{Q1} = \text{trace}\left(A^T(\hat{\mathbf{L}} + \mathbf{G})A\right) \tag{13}$$

$$\text{subject to } \tilde{\mathbf{L}} = \sum_{i=1}^{r} \theta_i \tilde{L}_i,$$

$$\mathbf{G} = \sum_{j=1}^{s} \theta_{j+r} G_{cj},$$

$$\sum_{i=1}^{r} \theta_i^{\delta} = 1, \ \sum_{j=1}^{s} \theta_{j+r}^{\delta} = 1,$$

$$\theta_l \geq 0, \ l = 1, ..., (r+s),$$

$$A^T A = I_K,$$

where $\theta_1, ..., \theta_r$ and $\theta_{r+1}, ..., \theta_{r+s}$ are respectively the optimal coefficients assigned to the Laplacians and the kernels. $\mathbf{G}$ and $\tilde{\mathbf{L}}$ are respectively the combined kernel matrix and the combined Laplacian matrix. The κ parameter in (12) is replaced by the coefficients assigned on each individual data sources.

To solve Q1, in the first phase we maximize $\mathcal{J}_{Q1}$ with respect to A, keeping $\vec{\theta}$ fixed (initialized by random guess). In the second phase we maximize $\mathcal{J}_{Q1}$ with respect to $\vec{\theta}$, keeping A fixed. The two phases optimize the same objective and repeat until convergence locally. When $\vec{\theta}$ is fixed, denoting $\mathbf{\Omega} = \tilde{\mathbf{L}} + \tilde{\mathbf{G}}$, Q_1 is exactly a Rayleigh quotient problem and the optimal A^* can be solved as a eigenvalue problem of $\mathbf{\Omega}$. When A is fixed, the problem reduces to the optimization of the coefficients θ_l with given cluster memberships. In **Supplementary material 2** we show that when the A is given, Q1 can be formulated as Kernel Fisher Discriminant (KFD) in the high dimensional feature space $\mathcal{F}$. We introduce $W = [\vec{w}_1, ..., \vec{w}_K]$, a projection matrix determining the pairwise discriminating hyperplane. Since the discriminant analysis is invariant to the magnitude of $\vec{w}$, we assume that $W^T W = I_K$, thus Q1 can be equivalently formulated as

$$\boxed{\text{Q2:}} \ \underset{A,W,\vec{\theta}}{\text{maximize}} \ \mathcal{J}_{Q2} = \text{trace}\left(W^T A^T A W\right)^{-1}\left(W^T A^T (\mathbf{G} + \hat{\mathbf{L}}) A W\right), \tag{14}$$

$$\text{s.t. } A^T A = I_k,$$

$$W^T W = I_k,$$

$$\hat{\mathbf{L}} = \sum_{i=1}^{r} \theta_i \hat{L}_i,$$

$$\mathbf{G} = \sum_{j=1}^{s} \theta_{j+r} G_{cj},$$

$$\theta_l \geq 0, \ l = 1, ..., (r+s),$$

$$\sum_{i=1}^{r} \theta_i^{\delta} = 1, \ \sum_{j=1}^{s} \theta_{j+r}^{\delta} = 1.$$

The bi-level optimization to solve Q1 correspond to two steps to solve Q2. In the first step (clustering), we set $W = I_k$ and optimize A, which is exactly the additive kernel Laplacian integration as (12); In the second step (KFD), we fix A and optimize W and $\vec{\theta}$. Therefore, the two components optimize towards the same objective as a Rayleigh quotient in $\mathcal{F}$ so the iterative optimization converges to a local optimum. Moreover, in the second step, we are not interested in the separating hyperplane defined in W, instead, we only need the optimal coefficients θ_l assigned on the Laplacians and the kernels. It is known that Fisher Discriminant Analysis is related to the least squares approach (Duda *et al*. (2001)), and the KFD (Mika *et al*. (1999)) is related to and can be solved as a least squares support vector machine (LS-SVM), proposed by (Suykens *et al*. (2002)). The problem of optimizing multiple kernels for supervised learning (MKL) has been studied by (Lanckriet *et al*. (2004);Bach *et al*. (2004)). In our recent work (Yu *et al*. (2010b)), we derive the MKL extension for LSSVM and propose some efficient solutions to solve the problem. In this paper, the KFD problems are formulated as LSSVM MKL and solved by Semi-infinite programming (SIP) (Sonnenburg *et al*. (2006)). The concrete solutions and algorithms are presented in (Yu *et al*. (2010b)).

Algorithm 2.1: OKLC($G_{c1}, ..., G_{cs}, \hat{L}_1, ..., \hat{L}_r, K$)

comment: Obtain the $\mathbf{\Omega^{(0)}}$ using the initial guess of $\theta_1^{(0)}, ..., \theta_{r+s}^{(0)}$

$A^{(0)} \leftarrow$ EIGENVALUE DECOMPOSITION($\mathbf{\Omega^{(0)}}, K$)

$\gamma = 0$

while $(\Delta A > \epsilon)$

do
- $step1: F^{(\gamma)} \leftarrow A^{(\gamma)}$
- $step2: \theta_1^{(\gamma)}, ..., \theta_r^{(\gamma)} \leftarrow$ SIP-LSSVM-MKL($\hat{L}_1, ..., \hat{L}_r, F^{(\gamma)}$)
- $step3: \theta_{r+1}^{(\gamma)}, ..., \theta_{r+s}^{(\gamma)} \leftarrow$ SIP-LSSVM-MKL($G_{c1}, ..., G_{cs}, F^{(\gamma)}$)
- $step4: \mathbf{\Omega^{(r+1)}} \leftarrow \theta_1^{(\gamma)} \hat{L}_1^{(\gamma)} + ... + \theta_r^{(\gamma)} \hat{L}_r^{(\gamma)} + \theta_{r+1}^{(\gamma)} G_{c1}^{(\gamma)} + ... + \theta_{r+s}^{(\gamma)} G_{cs}^{(\gamma)}$
- $step5: A^{(\gamma+1)} \leftarrow$ EIGENVALUE DECOMPOSITION($\mathbf{\Omega}^{(\gamma+1)}, K$)
- $step6: \Delta A = ||A^{(\gamma+1)} - A^{(\gamma)}||^2 / ||A^{(\gamma+1)}||^2$
- $step7: \gamma := \gamma + 1$

return $(A^{(\gamma)}, \theta_1^{(\gamma)}, ..., \theta_r^{(\gamma)}, \theta_{r+1}^{(\gamma)}, ..., \theta_{r+s}^{(\gamma)})$

2.3.1 Optimize A with given θ When θ are given, the kernel-Laplacian combined matrix $\mathbf{\Omega}$ is also fixed, therefore, the optimal A can be found as the dominant K number of eigenvectors of $\mathbf{\Omega}$.

2.3.2 Optimize θ with given A When A is given, the optimal θ assigned on Laplacians can be solved via the following KFD problem

$$\boxed{\text{Q3:}} \ \underset{W,\vec{\theta}}{\text{maximize}} \ \mathcal{J}_{Q3} = \text{trace}\left(W^T A^T A W\right)^{-1}\left(W^T A^T \hat{\mathbf{L}} A W\right) \tag{15}$$

$$\text{s.t. } W^T W = I_k,$$

$$\hat{\mathbf{L}} = \sum_{i=1}^{r} \theta_i \hat{L}_i,$$

$$\theta_i \geq 0, \ i = 1, ..., r,$$

$$\sum_{i=1}^{r} \theta_i^{\delta} = 1.$$

In our recent work, we have found that the δ parameter controls the sparseness of source coefficients $\theta_1, ..., \theta_r$ (Yu *et al*. (2010b)). The issue of sparseness in MKL is also addressed by Kloft *et al.* (Kloft *et al*. (2009)) When δ is set to 1, the optimized solution is sparse, which assigns dominant values to only one or two Laplacians (kernels) and zero values to the others. The sparseness is useful to distinguish relevant sources from a large number of irrelevant data sources. However, in many applications, there are usually a small number of sources and most of these data sources are carefully selected

and preprocessed. They thus often are directly relevant to the problem. In these cases, a sparse solution may be too selective to thoroughly combine the complementary information in the data sources. We may thus expect a non-sparse integration method which smoothly distributes the coefficients on multiple kernels and Laplacians and, at the same time, leverages their effects in the objective optimization. We have proved that when δ is set to 2, the KFD step in (15) optimizes the L_2-norm of multiple kernels, which yields a non-sparse solution. If we set δ to 0, the cluster objective is simplified as to averagely combine multiple kernels and Laplacians. In this paper, we set δ to three different vales (0,1,2) to respectively optimize the sparse, average, and non-sparse coefficients on kernels and Laplacians.

When δ is set to 1, the KFD problem in Q3 is solved as LSSVM MKL (Yu *et al*. (2010b)), given by

$$\boxed{\text{Q4:}} \quad \underset{\vec{\beta},t}{\text{minimize}} \ \frac{1}{2}t + \frac{1}{2\lambda}\sum_{b=1}^{K}\vec{\beta}_b^T\vec{\beta}_b - \sum_{b=1}^{K}\vec{\beta}_b^T Y_b^{-1}\vec{1} \tag{16}$$

$$\text{s.t.} \ \sum_{a=1}^{N}\beta_{ab} = 0, \ \ b = 1, ..., K,$$

$$t \geq \sum_{b=1}^{K}\vec{\beta}_b^T\hat{L}_i\vec{\beta}_b, \ \ i = 1, ..., r, \ b = 1, ..., K,$$

where $\vec{\beta}$ is the vector of dual variables, t is a dummy variable in optimization, a is the index of data samples, b is the cluster label index of the discriminating problem in KFD, Y_b is the diagonal matrix representing the binary cluster assignment, the vector on the diagonal of Y_b is equivalent to the b-th column of a affinity matrix F_{ab} using $\{+1, -1\}$ to discriminate the cluster assignments, given by

$$F_{ab} = \begin{cases} +1 & \text{if } A_{ab} > 0, \ \ a = 1, ..., N, \ b = 1, ..., K \\ -1 & \text{if } A_{ab} = 0, \ \ a = 1, ..., N, \ b = 1, ..., K \ . \end{cases} \tag{17}$$

The problem presented in Q4 has an efficient solution based on Semi-infinite programming (SIP), which is presented in equation forty-one of (Yu *et al*. (2010b)). The optimal coefficients θ_i correspond to the dual variables bounded by the quadratic constraint $t \geq \sum_{b=1}^{K}\vec{\beta}_b^T\hat{L}_i\vec{\beta}_b$ in (16). When δ is set to 2, the solution to Q3 is given by

$$\boxed{\text{Q5:}} \quad \underset{\vec{\beta},t}{\text{minimize}} \ \frac{1}{2}t + \frac{1}{2\lambda}\sum_{j=1}^{K}\vec{\beta}_b^T\vec{\beta}_b - \sum_{b=1}^{K}\vec{\beta}_b^T Y_b^{-1}\vec{1} \tag{18}$$

$$\text{s.t.} \ \sum_{a=1}^{N}\beta_{ab} = 0, \ \ b = 1, ..., K,$$

$$t \geq ||\vec{s}||_2,$$

where $\vec{s} = \{\sum_{b=1}^{K}\vec{\beta}_b^T\hat{L}_1\vec{\beta}_b, ..., \sum_{b=1}^{K}\vec{\beta}_b^T\hat{L}_r\vec{\beta}_b\}^T$, other variables are defined the same as (16). The problem Q5 also has an efficient solution presented in equation forty-two in our recent work (Yu *et al*. (2010b)). The main difference between Q4 and Q5 is that Q4 optimizes the L_∞ norm of multiple kernels whereas Q5 optimizes the L_2 norm. The optimal coefficients solved by Q4 are more likely to be sparse, in contrast, the ones obtained by Q5 are nonsparse. The algorithm to solve Q4 and Q5 is concretely explained in Algorithm 0.2 in (Yu *et al*. (2010b)).

Analogously, the coefficients assigned on kernels can also be obtained in the similar formulation, given by

$$\boxed{\text{Q6:}} \quad \max_{W,\vec{\theta}} \mathcal{J}_{Q6} = \text{trace}\left(W^T A^T A W\right)^{-1}\left(W^T A^T \mathbf{G} A W\right) \tag{19}$$

$$\text{s.t.} \ W^T W = I_K,$$

$$\mathbf{G} = \sum_{j=1}^{s}\theta_{j+r}G_{cj},$$

$$\theta_{j+r} \geq 0, \ j = 1, ..., s,$$

$$\sum_{j=1}^{s}\theta_{j+r}^{\delta} = 1,$$

where most of the variables are defined in the similar way as Q3 in (15). The main difference is that the Laplacian matrices $\hat{\mathbf{L}}$ and $\hat{L}_i$ are replaced by the centered kernel matrices $\mathbf{G}$ and G_{cj}. The solution of Q6 is exactly the same as Q3, depending on the δ value, it can be solved either as Q4 or Q5.

2.3.3 Algorithm: Optimized kernel Laplacian clustering As discussed, the proposed algorithm optimizes A and θ iteratively to convergence. The coefficients assigned to the Laplacians and the kernels are optimized in parallel. Putting all the steps together, the pseudocode of the proposed optimized kernel Laplacian clustering (OKLC) is presented in Algorithm 2.1.

The iterations in Algorithm 2.1 terminate when the cluster membership matrix A stops changing. The tolerance value ϵ is a constant value as the stopping rule of OKLC and in our implementation it is set to 0.05. In our implementation, the final cluster assignment is obtained using K-means algorithm on $A^{(\gamma)}$. In Algorithm 2.1, we consider the δ as predefined values. When δ is set to 1 or 2, the SIP-LSSVM-MKL function optimizes the coefficients as the formulation in (16) or (18) respectively. It is also possible to combine Laplacians and kernels in an average manner. In this paper, we compare all these approaches and implement three different OKLC models. These three models are denoted as OKLC model 1, OKLC model 2, and OKLC model 3 which respectively correspond to the objective Q2 in (14) when $\delta = 1$, average combination, $\delta = 2$.

2.4 Data sets and experimental setup

The proposed OKLC models are validated in two real applications to combine heterogeneous data sets in clustering analysis. The data sets in the first experiment is taken from the work of multi-view text mining for disease gene identification (Yu *et al*. (2010a)). The data sets contain nine different gene-by-term text profiles indexed by nine controlled vocabularies. The original disease relevant gene data set contains 620 genes which are known to be relevant to 29 diseases. To avoid the effect of imbalanced clusters which may affect the evaluation, we only keep the diseases who have 11 to 40 relevant genes. This results in 14 genetic diseases and 278 genes. Because the present paper is focused on non-overlapping ("hard") clustering, we further remove 16 genes which are relevant to multiple diseases. The remaining 262 disease relevant genes are clustered into 14 clusters and evaluated biologically by their disease labels. For each vocabulary based gene-by-term data source, we create a kernel matrix using the linear kernel function and the *kernel normalisation* method proposed by (Chapter 5, Shawe-Taylor and Cristianini (2004)). An element in the kernel matrix is then equivalent to the value of cosine similarity of two vectors (Baeza-Yates and Ribeiro-Neto (1999)). This kernel is then regarded as the weighted adjacency matrix to create the Laplacian matrix. In total, nine kernels and nine Laplacian matrices are combined in clustering.

The data sets in the second experiment are taken from Web of Science (WOS) database provided by Thomson Scientific (Liu *et al*. (2010)). After preprocessing, the data set contains 8305 journals categorized in 22 scientific fields. To create a balanced benchmark data for evaluation, we select seven fields consisting 1421 journals. The titles, abstracts and keywords

Table 1. Performance on disease data set. All the comparing methods combine nine kernels and nine Laplacians. The mean values and the standard deviations are observed from 20 random repetitions. The best performance is shown in bold. The P-values are statistically evaluated with the best performance using paired t-test.

Algorithm	ARI	P-value	NMI	P-value
OKLC 1	**0.5859**±0.0390	-	**0.7451**±0.0194	-
OKLC 2	0.5369±0.0493	2.97E-04	0.7106±0.0283	9.85E-05
OKLC 3	0.5469±0.0485	1.10E-03	0.7268±0.0360	2.61E-02
CSPA	0.4367±0.0266	5.66E-11	0.6362±0.0222	4.23E-12
HGPA	0.5040±0.0363	8.47E-07	0.6872±0.0307	7.42E-07
MCLA	0.4731±0.0320	2.26E-10	0.6519±0.0210	5.26E-14
QMI	0.4656±0.0425	7.70E-11	0.6607±0.0255	8.49E-11
EACAL	0.4817±0.0263	2.50E-09	0.6686±0.0144	5.54E-12
AdacVote	0.1394±0.0649	1.47E-16	0.4093±0.0740	6.98E-14

Table 2. Performance on journal data set. All the comparing methods combine four kernels and four Laplacians. The mean values and the standard deviations are observed from 20 random repetitions. The best performance is shown in bold. The P-values are statistically evaluated with the best performance using paired t-test.

Algorithm	ARI	P-value	NMI	P-value
OKLC 1	0.7346±0.0584	0.3585	0.7688±0.0364	0.1472
OKLC 2	0.7235±0.0660	0.0944	0.7532±0.0358	0.0794
OKLC 3	**0.7336**±0.0499	-	**0.7758**±0.0362	-
CSPA	0.6703±0.0485	8.84E-05	0.7173±0.0291	1.25E-05
HGPA	0.6673±0.0419	4.74E-06	0.7141±0.0269	5.19E-06
MCLA	0.6571±0.0746	6.55E-05	0.7128±0.0463	2.31E-05
QMI	0.6592±0.0593	5.32E-06	0.7250±0.0326	1.30E-05
EACAL	0.5808±0.0178	3.85E-11	0.7003±0.0153	6.88E-09
AdacVote	0.5899±0.0556	1.02E-07	0.6785±0.0325	6.51E-09

of the journal publications are indexed by a text mining program using no controlled vocabulary. The weights of terms are calculated using four weighting schemes: TF-IDF, IDF, TF and binary. The citations among journals are also investigated from four different aspects: cross-citation, co-citation, bibliographic coupling and binary cross-citation. The lexical similarities are represented as normalized linear kernel matrices (using the same methods applied on the disease data) and the citation metrics are regarded as weighted adjacency matrices to create the Laplacians. Totally four kernels and four Laplacians are combined on journal data. The details about the two data sets are presented in **Supplementary material 3**.

The data sets used in our experiments are provided with labels, therefore the clustering performance is evaluated as comparing the automatic partitions with the labels using Adjusted Rand Index (ARI) (Hubert and Arabie (1985)) and Normalized Mutual Information (NMI) (Strehl and Ghosh (2002)). To evaluate the ARI and NMI performance, we set $K = 14$ on disease data and $K = 7$ on journal data. We also tune the OKLC model using different K values.

3 RESULTS

We implement the proposed OKLC models to integrate multiple kernels and Laplacians on disease data and journal set data. To compare the performance, we also apply six popular ensemble clustering methods mentioned in relevant work (Yu *et al*. (2010a)) to combine the partitions of individual kernels and Laplacians as a consolidate partition. These six methods are CSPA (Strehl and Ghosh (2002)), HGPA (Strehl and Ghosh (2002)), MCLA (Strehl and Ghosh (2002)), QMI (Topchy *et al*. (2005)), EACAL (Fred and Jain (2005)), and AdacVote (Ayad and Kamel (2008)). As shown in Table 1 and Table 2, the performance of OKLC algorithms is better than all the compared methods and the improvement is significant. On disease data, the best performance is obtained by OKLC model 1, which uses sparse coefficients to combine 9 text mining kernels and 9 Laplacians to identify disease relevant clusters (ARI: 0.5859, NMI: 0.7451). On journal data, all three OKLC models perform comparably well. The best one seems coming from OKLC model 3 (ARI: 0.7336, NMI: 0.7758), which optimizes the non-sparse coefficients on the 4 kernels and 4 Laplacians.

To evaluate whether the combination of kernel and Laplacian indeed improve the clustering performance, we first systematically compared the performance of all the individual data sources using KM and SC. As shown in **Supplementary material 4**, on disease data, the best KM performance (ARI 0.5441, NMI 0.7099) and SC (ARI 0.5199, NMI 0.6858) performance is obtained on LDDB text mining profile. Next, we enumerate all the paired combinations of a single kernel and a single Laplacian for clustering. The integration is based on equation (12) and the κ value is set to 0.5 so the objectives of KM and SC are combined averagely. The performance of all 45 paired combinations is presented in **Supplementary material 5**. As shown, the best KL clustering performance is obtained by integrating the LDDB kernel with KO Laplacian (ARI 0.5298, NMI 0.6949). Moreover, we also found that the integration performance varies significantly by the choice of kernel and Laplacian, which proves our previous point that the KL performance is highly dependent on the quality of kernel and Laplacian. Using the proposed OKLC algorithm, there is no need to enumerate all the possible paired combinations. OKLC combines all the kernels and Laplacians and optimizes their coefficients in parallel, yielding a comparable performance with the best paired combination of a single kernel and a single Laplacian.

In Figure 1 two confusion matrices of disease data for a single run are depicted. The values on the matrices are normalized according to $R_{ij} = C_j/T_i$, where T_i is the total number of genes belonging in disease i and C_j is the number of these T_i genes that were clustered to belong to class j. First, it is worth noting that OKLC reduces the number of mis-clustered genes on breast cancer (Nr.1), cardiomyopathy (Nr.2), and muscular dystrophy (Nr.11). Among the mis-clustered genes in LDDB, five genes (TSG101, DBC1, CTTN, SLC22A18, AR) in breast cancer, two genes in cardiomyopathy (COX15, CSRP3), and two genes in muscular dystrophy (SEPN1, COL6A3) are correctly clustered in OKLC model 1. Second, there are several diseases where consistent misclustering occurs in both methods, such as diabetes (Nr.6) and neuropathy (Nr.12). The intuitive confusion matrices correspond to the numerical evaluation results, as shown, the quality of clustering obtained by OKLC model 1 (ARI=0.5898, NMI=0.7429) is higher than LDDB.

The performance of individual data sources of journal data is shown in **Supplementary material 6**. The best KM (ARI 0.6482, NMI 0.7104) is obtained on the IDF kernel and the best SC (ARI 0.5667, NMI 0.6807) is obtained on the cross-citation Laplacian. To combine the 4 kernels with 4 Laplacians, we evaluate all the 10 paired combinations and show the performance in **Supplementary material 7**. The best performance is obtained by integrating the

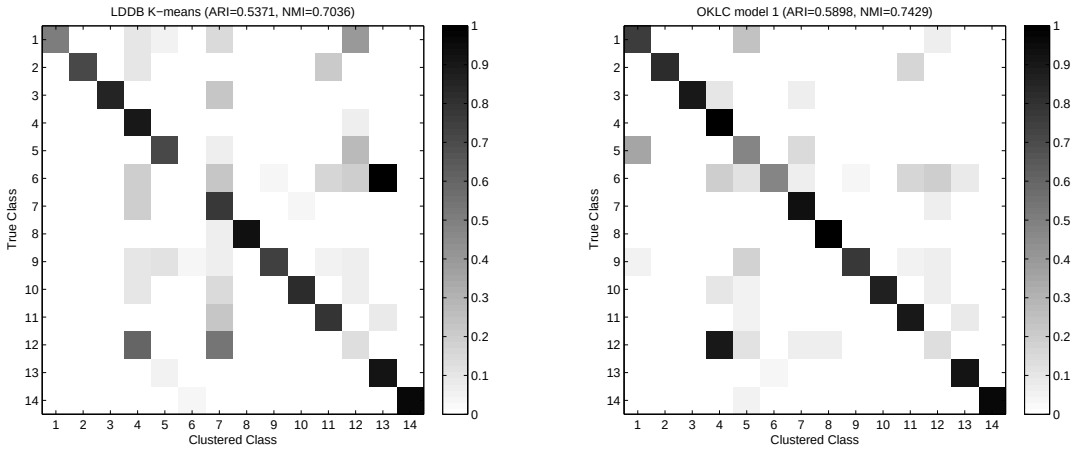


Fig. 1. Confusion matrices of disease data obtained by kernel K-means on LDDB (figure on the left) and OKLC model 1 integration (figure on the right). The numbers of cluster labels are consistent with the numbers of diseases presented in **Supplementary material 3**. In each row of the confusion matrix, the diagonal element represents the fraction of correctly clustered genes and the off-diagonal non-zero element represents the fraction of mis-clustered genes.

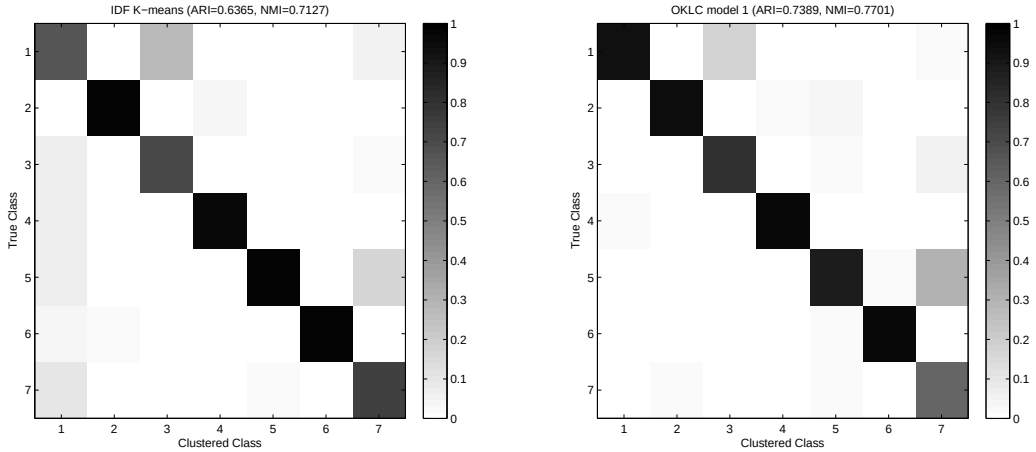


Fig. 2. Confusion matrices of journal data obtained by kernel K-means on IDF (figure on the left) and OKLC model 1 integration (figure on the right). The numbers of cluster labels are consistent with the numbers of ESI journal categories presented in **Supplementary material 3**. In each row, the diagonal element represents the fraction of correctly clustered journals and the off-diagonal non-zero element represents the fraction of mis-clustered journals.

IDF kernel with the cross-citation Laplacian (ARI 0.7566, NMI 0.7702). As shown, the integration of lexical similarity information and citation based Laplacian indeed improves the performance.

In Figure 2 the confusion matrices (also normalized) of journal data for a single run are illustrated. We compare the best individual data source (IDF with kernel K-means, figure on the left) with the OKLC model 1. In the confusion matrix of IDF K-means, 79 journals belonging to agriculture science (Nr.1) are mis-clustered to environment ecology (Nr.3), 9 journals are mis-clustered to pharmacology and toxicology (Nr.7). In OKLC, the number of agriculture journals mis-clustered to environment ecology is reduced to 45, and the number to pharmacology and toxicology is reduced to 5. On other journal clusters, the performance of the two models is almost equivalent.

We also investigate the performance of only combining multiple kernels or multiple Laplacians. On the disease data set, we respectively combined the 9 kernels and the 9 Laplacians for clustering, using all the compared methods in Table 1 and Table 2. On the journal data set, we combine the 4 text mining kernels and the 4 citation Laplacians. The proposed OKLC method is simplified as only optimizing coefficients on Laplacians (step 2 in Algorithm 2.1) or kernels (step 3). As shown in **Supplementary material 8**, the performance of OKLC is also comparable to the best performance obtained either by kernel combination or Laplacian combination. In particular, of all the methods we compared, the best performance is all obtained on OKLC models or its simplified forms.

It is interesting to observe that the average combination model (OKLC model 2) performs quite well on the journal data set but not on the disease data set. This is probably because most of the

sources in journal data set are relevant to the problem whereas in disease data set some data sources are noisy, thus the integration of disease data sources is a non-trivial task. We expect that the other two OKLC models (model 1 and model 3) optimize the coefficients assigned on the kernels and the Laplacians to leverage multiple sources in integration, at the same time, to increase the robustness of the combined model on combining relevant and irrelevant data sources. To evaluate whether the optimized weights assigned on individual sources have correlation with the performance, we compare the rank of coefficients with the rank of performance from Table 3 to Table 6. As shown, the largest coefficients correctly indicate the best individual data sources. It is worth noting that in multiple kernel learning, the rank of coefficients are only moderately correlated with the rank of individual performance. In our experiments, the MeSH kernel gets the 2nd largest weights though its performance in evaluation is low. In MKL, it is usual that the best individual kernel found by cross-validation may not lead to a large weight when used in combination (Ye *et al*. (2008)). Kernel fusion combines multiple sources at a refined granularity, where the "moderate" kernels containing weak and insignificant information could complement to other kernels to compose a "good" kernel containing strong and significant information. Though such complementary information cannot be incorporated when cross-validation is used to choose a single best kernel, these "moderate" kernels are still useful when combined with other kernels (Ye *et al*. (2008)). Based on the ranks presented in Table 5 and Table 6, we calculate the spearman correlations between the ranks of weights and the ranks of performance on both data sets. The correlations of disease kernels, disease Laplacians, journal kernels and journal Laplacians are respectively 0.5657, 0.6, 0.8, and 0.4. In some relevant work, the average spearman correlations are mostly around 0.4 (Lanckriet *et al*. (2004);Ye *et al*. (2008)). Therefore, the optimal weights obtained in our experiments are generally consistent with the rank of performance.

As a spectral clustering algorithm, the optimal cluster number of OKLC can be estimated by checking the plot of eigenvalues (von Luxburg (2007)). To demonstrate this, we investigate the dominant eigenvalues of the optimized combination of kernels and Laplacians. In Figure 3, we compare the difference of three OKLC models with the pre-defined K (set as equal to the number of class labels). In practical research, one can predict the optimal cluster number by checking the "elbow" of the eigenvalue plot. As shown in Figure 3, the "elbow" in disease data is quite obvious at the number of 14. In journal data, the "elbow" is more likely to range from 6 to 12. All the three OKLC models show a similar trend on the eigenvalue plot. Moreover, in **Supplementary material 9** we also compare the eigenvalue curves using different K values as input. As shown, the eigenvalue plot is quite stable w.r.t. the different inputs of K, which means the optimized kernel and Laplacian coefficients are quite independent with the K value. This advantage enables a reliable prediction about the optimal cluster number by integrating multiple data sources.

To investigate the computational time, we benchmark OKLC algorithms with other clustering methods on the two data sets. As shown in Table 7, when optimizing the coefficients, OKLC algorithm (model 1 and model 3) spends longer time than the other methods to optimize the coefficients on the Laplacians and the kernels. However, the proposed algorithm is still efficient. Considering the fact that the proposed algorithm yields much better performance and more enriched information (the ranking of the individual sources) than other methods, it is worth spending extra computational complexity on a promising algorithm.

Table 3. The average values of coefficients of kernels and Laplacians in disease data set optimized by OKLC model 1. The sources assigned with 0 coefficient are not presented. The performance is ranked by the average values of ARI and NMI evaluated on each individual sources (Supplementary Material 3).

Rank of θ	Source	θ value	Performance rank
1	LDDB kernel	0.6113	1
2	MESH kernel	0.3742	6
3	Uniprot kernel	0.0095	5
4	Omim kernel	0.0050	2
1	LDDB Laplacian	1	1

Table 4. The average values of coefficients of kernels and Laplacians in journal data set optimized by OKLC model 1. The sources assigned with 0 coefficient are not presented. The performance is ranked by the average values of ARI and NMI evaluated on each individual sources (Supplementary Material 5).

Rank of θ	Source	θ value	Performance rank
1	IDF kernel	0.7574	1
2	TF kernel	0.2011	3
3	Binary kernel	0.0255	2
4	TF-IDF kernel	0.0025	4
1	Bibliographic Laplacian	1	1

Table 5. The average values of coefficients of kernels and Laplacians in disease data set optimized by OKLC model 3.

Rank of θ	Source	θ value	Performance rank
1	LDDB kernel	0.4578	1
2	MESH kernel	0.3495	6
3	OMIM kernel	0.3376	2
4	SNOMED kernel	0.3309	7
5	MPO kernel	0.3178	3
6	GO kernel	0.3175	8
7	eVOC kernel	0.3180	4
8	Uniprot kernel	0.3089	5
9	KO kernel	0.2143	9
1	LDDB Laplacian	0.6861	1
2	MESH Laplacian	0.2799	4
3	OMIM Laplacian	0.2680	2
4	GO Laplacian	0.2645	7
5	eVOC Laplacian	0.2615	6
6	Uniprot Laplacian	0.2572	8
7	SNOMED Laplacian	0.2559	5
8	MPO Laplacian	0.2476	3
9	KO Laplacian	0.2163	9

4 CONCLUSION

In this paper we propose a new clustering approach, OKLC, to optimize the combination of multiple kernels and Laplacians in

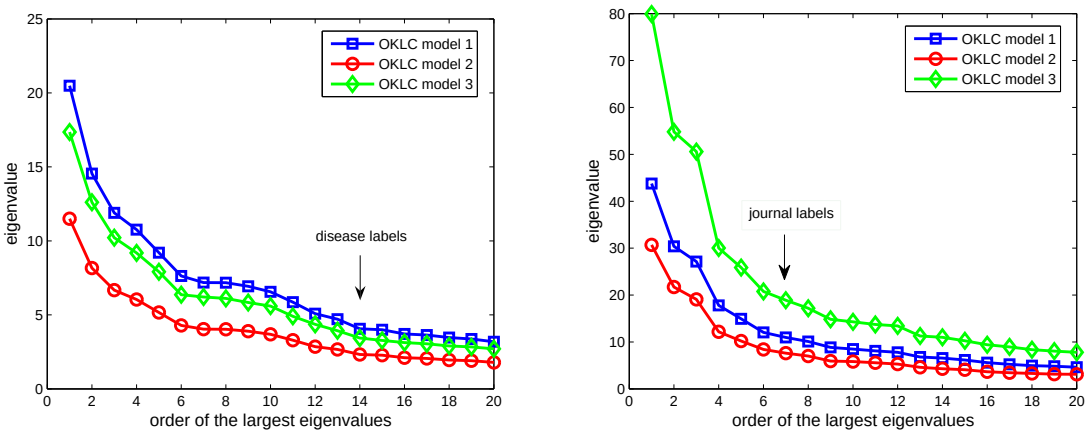


Fig. 3. The plot of eigenvalues (disease data on the left, journal data on the right) of the optimal kernel-Laplacian combination obtained by all OKLC models. The parameter K is set as equivalent as the reference label numbers.

Table 6. The average values of coefficients of kernels and Laplacians in journal data set optimized by OKLC model 3.

Rank of θ	Source	θ value	Performance rank
1	IDF kernel	0.5389	1
2	Binary kernel	0.4520	2
3	TF kernel	0.2876	4
4	TF-IDF kernel	0.2376	3
1	Bibliographic Laplacian	0.7106	1
2	Cocitation Laplacian	0.5134	4
3	Crosscitation Laplacian	0.4450	2
4	Binarycitation Laplacian	0.1819	3

Table 7. Comparison of CPU time of all algorithms. The reported values are averaged from 20 repetitions. The CPU time is evaluated on Matlab v7.6.0 + Windows XP2 installed on a Laptop computer with Intel Core 2 Duo 2.26G Hz and 2G memory.

Algorithm	disease data (seconds)	journal data (seconds)
OKLC model 1	42.39	1011.4
OKLC model 2	0.19	13.27
OKLC model 3	37.74	577.51
CSPA	9.49	177.22
HGPA	10.13	182.51
MCLA	9.95	320.93
QMI	9.36	186.25
EACAL	9.74	205.59
AdacVote	9.22	172.12

clustering analysis. The objective of OKLC is formulated as a Rayleigh quotient function and is solved iteratively as a bi-level optimization procedure. In the simplest interface, the proposed algorithm only requires one input parameter, the cluster number K, from the user. Moreover, depending on user's expectation to select the most relevant sources or to evenly combine all sources, the sparseness of coefficient vector θ can be controlled via the parameter δ. In our paper we propose three variants of the OKLC algorithm and validate them on two real applications. The performance of clustering is systematically compared with a variety of algorithms and different experimental settings. The proposed OKLC algorithms perform significantly better than other methods. Moreover, the coefficients of kernels and Laplacians optimized by OKLC show strong correlation with the rank of performance of individual data source. Though in our evaluation the K values are predefined, in practical studies, the optimal cluster number can be consistently estimated from the eigenspectrum of the combined kernel Laplacian matrix.

The proposed OKLC algorithm demonstrates the advantage of combining and leveraging information from heterogeneous data structures and sources. It is potentially useful in bioinformatics and many other application areas, where there is a surge of interest to integrate similarity based information and interaction based relationships in statistical analysis and machine learning.

ACKNOWLEDGEMENT

The work was supported by (1) Research Council KUL: ProMeta, GOA Ambiorics, GOA MaNet, CoE EF/05/006, PFV/10/016 SymBioSys, START 1, Optimization in Engineering(OPTEC), IOF-SCORES4CHEM, several PhD/postdoc & fellow grants; (2) FWO: G.0302.07(SVM/Kernel), G.0318.05 (subfunctionalization), G.0553.06 (VitamineD), research communities (ICCoS, ANMMM, MLDM); G.0733.09 (3UTR), G.082409 (EGFR); (3) IWT: PhD Grants, Eureka-Flite+, Silicos; SBO-BioFrame, SBO-MoKa, SBO LeCoPro, SBO Climaqs, SBO POM, TBM-IOTA3, O&O-Dsquare; (4) Belgian Federal Science Policy Office: IUAP P6/25 (BioMaGNet, Bioinformatics and Modeling: from Genomes to Networks, 2007-2011), IUAP P6/04 (DYSCO, Dynamical systems, control and optimization, 2007-2011); (5) FOD:Cancer plans; (6) Centre for R&D Monitoring of the Flemish Government; (7) EU-RTD: ERNSI: European Research Network on System

Identification; FP7-HEALTH CHeartED; FP7-HD-MPC (INFSO-ICT-223854), COST intelliCIS, FP7-EMBOCON (ICT-248940).

REFERENCES

Ayad,H.G. and Kamel,M.S. (2008) Cumulative Voting Consensus Method for Partitions with a Variable Number of Clusters. *IEEE Trans. PAMI*, **30**(1), 160-173.

Bach,F.R. *et al*. (2004) Multiple kernel learning, conic duality, and the SMO algorithm. In *21st International Conference on Machine Learning*. ACM, Banff, Alberta, pp.6-13.

Baeza-Yates,R. and Ribeiro-Neto,B. (1999) *Modern Information Retrieval*. ACM press.

Bishop,C.M. (2006) *Pattern recognition and machine learning*. Springer.

Csiszar,I. and Tusnady,G. (1984) Information geometry and alternating minimization procedures. *Statistics and Decisions*, Suppl. 1, 205-237.

Dhillon,L.S. *et al*. (2004) Kernel k-means, Spectral Clustering and Normalized Cuts. In *Proceedings of the 10th ACM KDD*, ACM, Seattle, WA, pp.551-556.

Ding,C. and He,X. (2004) K-means clustering via principal component analysis. In *21st International Conference on Machine Learning*, ACM, Banff, Alberta, pp.225-232.

Duda,R.O. *et al*. (2001) *Pattern Classification*. 2nd edn. John Wiley & Sons Inc.

Fred,A.L.N. and Jain,A.K. (2005) Combining Multiple Clusterings Using Evidence Accumulation. *IEEE Trans. PAMI*, **27**(6), 835-850.

Girolami,M. (2002) Mercer Kernel-Based Clustering in Feature Space. *IEEE Trans. Neural Network*, **13**, 780-784.

Hagen,L. and Kahng,A. (1992) New spectral methods for ratio cut partitioning and clustering. *IEEE Trans. Computer-Aided Design*, **11**, 1074-1085.

Hastie,T. *et al*. (2009) *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. 2nd edn. Springer-Verlag.

Hubert,L. and Arabie,P. (1985) Comparing Partition. *Journal of Classification*, **2**, 193-218.

Kloft, M. *et al*. (2009) Efficient and Accurate Lp-norm Multiple Kernel Learning. In *Advances in Neural Information Processing System 22*.

Lanckriet,G. *et al*. (2004) Learning the kernel matrix with semidefinite programming. *Journal of Machine Learning Research*, **5**, 27-72.

Liu,X. *et al*. (2010) Weighted Hybrid Clustering by Combining Text Mining and Bibliometrics on Large-Scale Journal Database. *Journal of the American Society for Information Science and Technology*, **61**, 1105-1119.

Mika,S. *et al*. (1999) Fisher discriminant analysis with kernels. *IEE N.N. for Singal Processing*, **9**, 41-48.

Ng,A.Y. (2001) On spectral clustering: Analysis and an algorithm. In *Advances in Neural Information Processing 14*, pp.849-856.

Shawe-Taylor,J. and Cristianini,N. (2004) *Kernel methods for Pattern Analysis*. Cambridge University Press.

Shi,J. and Malik,J. (2000) Normalized cuts and image segmentation. *IEEE Trans. PAMI*, **22**, 888-905.

Sonnenburg,S. *et al*. (2006) Large Scale Multiple Kernel Learning. *Journal of Machine Learning Research*, **7**, 1531-1565.

Strehl,A. and Ghosh,J. (2002) Cluster Ensembles: A Knowledge Reuse Framework for Combining Multiple Partitions. *Journal of Machine Learning Research*,**3**,583-617.

Suykens,J.A.K. *et al*. (2002) *Least Squares Support Vector Machines*. World Scientific Publishing, Singapore.

Topchy,A. *et al*. (2005) Clustering Ensembles: Models of Consensus and Weak Partitions. *IEEE Trans. PAMI*, **27**, 1866-1881.

Wang,F. *et al*. (2009) Integrated KL(K-means-Laplacian) clustering: a new clustering approach by combining attribute data and pairwise relations. in *Proccedings of SDM 09*, pp. 38-48.

Ye,J. *et al*. (2007) Nonlinear adaptive distance metric learning for clustering. In *Proccedings of the 13th ACM KDD*, ACM, San Jose, CA, pp. 123-132.

Ye,J. *et al*. (2008) Multi-class discriminant kernel learning via convex programming. *Journal of Machine Learning Research*, **9**, 719-758.

Yu,S. *et al*. (2010a) Gene prioritization and clustering by multi-view text mining. *BMC Bioinformatics*, **11**, 1-28.

Yu,S. *et al*., (2010b) L2-norm multiple kernel learning and its application to biomedical data fusion. *BMC Bioinformatics*, **11**,1-53.

von Luxburg,U. (2007) A tutorial on spectral clustering. *Statistics and Computing*, **17**, 395-416.