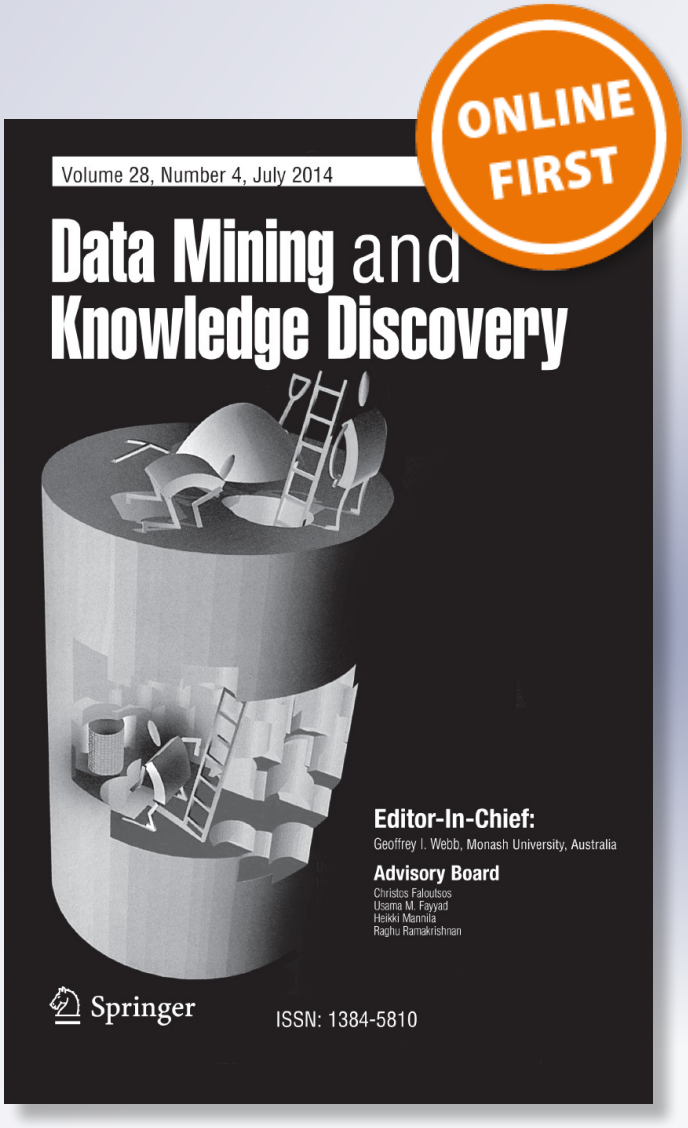
Evolutionary soft co-clustering: formulations, algorithms, and applications

Wenlu Zhang, Rongjian Li, Daming Feng, Andrey Chernikov, Nikos Chrisochoides, Christopher Osgood & Shuiwang Ji

Data Mining and Knowledge Discovery

ISSN 1384-5810

Data Min Knowl Disc
DOI 10.1007/s10618-014-0375-9



Springer

Data Min Knowl Disc
DOI 10.1007/s10618-014-0375-9

Evolutionary soft co-clustering: formulations, algorithms, and applications

Wenlu Zhang · Rongjian Li · Daming Feng · Andrey Chernikov · Nikos Chrisochoides · Christopher Osgood · Shuiwang Ji

Received: 14 November 2013 / Accepted: 22 July 2014

Abstract We consider the co-clustering of time-varying data using evolutionary co-clustering methods. Existing approaches are based on the spectral learning framework, thus lacking a probabilistic interpretation. We overcome this limitation by developing a probabilistic model in this paper. The proposed model assumes that the observed data are generated via a two-step process that depends on the historic co-clusters. This allows us to capture the temporal smoothness in a probabilistically principled manner. To perform maximum likelihood parameter estimation, we present an EM-based algorithm. We also establish the convergence of the proposed EM algorithm. An appealing feature of the proposed model is that it leads to soft co-clustering assignments naturally. We evaluate the proposed method on both synthetic and real-world data sets.

Responsible editor: Eamonn Keogh.

W. Zhang · R. Li · D. Feng · A. Chernikov · N. Chrisochoides · S. Ji (✉)
Department of Computer Science, Old Dominion University, Norfolk, VA 23529, USA
e-mail: sji@cs.odu.edu

W. Zhang
e-mail: wzhang@cs.odu.edu

R. Li
e-mail: rli@cs.odu.edu

D. Feng
e-mail: dfeng@cs.odu.edu

A. Chernikov
e-mail: achernik@cs.odu.edu

N. Chrisochoides
e-mail: nikos@cs.odu.edu

C. Osgood
Department of Biological Sciences, Old Dominion University, Norfolk, VA 23529, USA
e-mail: cosgood@odu.edu

Published online: 08 August 2014

Experimental results show that our method consistently outperforms prior approaches based on spectral method. To fully exploit the real-world impact of our methods, we further perform a systematic application study on the analysis of *Drosophila* gene expression pattern images. We encode the spatial gene expression information at a particular developmental time point into a data matrix using a mesh-generation pipeline. We then co-cluster the embryonic domains and the genes simultaneously for multiple time points using our evolutionary co-clustering method. Results show that the co-clusters of gene and embryonic domains reflect the underlying biology.

Keywords Evolutionary co-clustering · Expectation maximization · Biological image computing · Bioinformatics

1 Introduction

The data generated by many real-world processes are dynamically changing over time. For example, in literature mining, the author-conference co-occurrence matrix evolves dynamically over time, since authors may shift their research interests smoothly. In biology, gene expression controls are deployed sequentially in many biological processes. This generates the expression data matrices that are evolving over time. Temporal data mining aims at discovering knowledge from time-varying data and is now receiving increasing attention in many domains, including graph and network analysis (Leskovec et al. 2007; Asur et al. 2007; Sun et al. 2007), information retrieval (Tong et al. 2008; Saha and Sindhwani 2012), text mining (Mei and Zhai 2005), clustering analysis (Aggarwal et al. 2003; Chakrabarti et al. 2006; Chi et al. 2009; Lin et al. 2009), and matrix factorization (Wang et al. 2008, 2011a,b). Since the data are evolving smoothly over time, the patterns embedded into the data are also expected to change smoothly. Therefore, one of the key challenges in temporal data mining is how to incorporate temporal smoothness into the patterns identified from adjacent time points.

In traditional clustering analysis, the sample and feature dimensions are treated asymmetrically (Jain et al. 1999). In contrast, co-clustering aims at clustering both the samples and the features simultaneously to identify hidden block structures embedded into the data matrix (Hartigan 1972; Cheng and Church 2000; Dhillon et al. 2003; Long et al. 2005; Deodhar and Ghosh 2010). Co-clustering is closely related to matrix and tensor factorization (Tao et al. 2007; Li and Tao 2013a,b). Currently, co-clustering has been widely applied in many domains, including biological data analysis (Madeira and Oliveira 2004; Kluger et al. 2003), text mining (Dhillon et al. 2003; Dhillon 2001), and social studies (Giannakidou et al. 2008). However, most existing studies on co-clustering assume that the data are static.

In this paper, we consider the co-clustering of data matrices that evolve dynamically over time. A simple approach is to apply co-clustering methods to each data matrix separately. This approach, however, ignores the smoothness between adjacent matrices. Existing methods are based on the spectral learning framework and do not require the co-cluster indicator matrices to be nonnegative, hindering a probabilistic interpretation of the results (Green et al. 2011). We overcome this limitation by developing a

probabilistic model for this problem. The proposed probabilistic model assumes that the observed data matrices are generated via a two-step process that depends on the historic co-clusters. This allows us to capture the temporal smoothness in a probabilistically principled manner. To enable maximum likelihood parameter estimation, we develop an EM algorithm for the probabilistic model. We further establish the convergence of the EM algorithm. An appealing feature of the proposed method is that it leads to soft co-clustering assignments naturally. We evaluate the proposed methods on both synthetic and real data sets. Experimental results show that the proposed model consistently outperforms prior methods based on spectral learning.

We perform a systematic application study on the analysis of *Drosophila* gene expression pattern images. In this application, we use a geometric domain tessellation pipeline to convert gene expression pattern images to an algebraic representation, which is a data matrix for each of the developmental time point. We then apply our evolutionary co-clustering algorithm to cluster the genes and the mesh elements simultaneously across multiple time points. Experimental results show that the co-expressed embryonic domains and the associated genes reflect the underlying biology of *Drosophila* embryogenesis.

A preliminary version of the evolutionary soft co-clustering formulation described in the current paper appeared at the 2013 SIAM International Conference on Data Mining (Zhang et al. 2013). We extend the conference paper by establishing the convergence of the proposed EM algorithm in this paper. In addition, complete details on the probability distributions underlying the proposed model are provided. We also perform a systematic application study on the analysis of *Drosophila* gene expression pattern images. The new study involves a mesh generation pipeline that converts genome-wide expression pattern images into the same coordinate space in which the evolutionary co-clustering method is applied to identify co-expressed genes and embryonic domains. The background and discussions have also been substantially expanded to provide more insights.

The rest of this paper is organized as follows: We begin by introducing some background in Sect. 2 and related work and extensions in Sect. 3. In Sect. 4, the probabilistic model is presented. The experimental studies on synthetic and publication data sets are reported in Sect. 5. The in-depth application study is described in Sect. 6, and this paper concludes with discussions in Sect. 7.

Notations We use $\mathrm{Tr}(W)$ to represent the trace of matrix W where $\mathrm{Tr}(W) = \sum_{i=1}^{n} w_{ii}$ for any matrix $W \in \mathbb{R}^{n\times n}$. The squared Frobenius norm of a matrix W is defined as $\|W\|_F^2 = \sum_{i,j} w_{i,j}^2 = \mathrm{Tr}(W^T W)$. We use $A \in \mathbb{R}^{m\times n}$ to denote the data matrix for a problem with k co-clusters, the co-clustering results can be encoded into a co-cluster indicator matrix $R \in \mathbb{R}^{(m+n)\times k}$. Let $R^T = [R_1^T, R_2^T]$, where $R_1 \in \mathbb{R}^{m\times k}$ and $R_2 \in \mathbb{R}^{n\times k}$. The indicator matrix R is defined as follows: $(R_1)_{ij} = 1$ if the ith row belongs to the jth co-cluster, and zero otherwise; $(R_2)_{ij} = 1$ if the ith column belongs to the jth co-cluster, and zero otherwise. We further define $\tilde{R} \in \mathbb{R}^{(m+n)\times k}$, where each column of $\tilde{R}$ is the corresponding column in R divided by the square root of the number of ones in that column.

2 Background

Cluster analysis aims at grouping a set of data points into clusters so that the data points in the same cluster are similar, while those in different clusters are dissimilar. Given a data matrix $A = [a_1, a_2, \ldots, a_n] \in \mathbb{R}^{m \times n}$ consisting of n data points $\{a_i\}_{i=1}^n \in \mathbb{R}^m$. Let $\Pi = \{\pi_j\}_{j=1}^k$ denote a partition of the data into k clusters; that is, $\pi_j = \{v | a_v \text{in cluster } j\}$ and $\pi_i \bigcap \pi_j = \emptyset$ for $i \neq j$. The partition can also be encoded equivalently into an $n \times k$ cluster indicator matrix $Y = [y_1, y_2, \ldots, y_k]$, where $Y_{pq} = 1$ if the pth data point belongs to the qth cluster, and 0 otherwise. We further define a normalized cluster indicator matrix $\tilde{Y} = [\tilde{y}_1, \tilde{y}_2, \ldots, \tilde{y}_k]$, where $\tilde{y}_i = y_i / \sqrt{|\pi_i|}$ and $|\pi_i|$ denotes the number of data points in the ith cluster. It can be verified that the columns of $\tilde{y}$ are orthonormal, i.e., $\tilde{y}^T \tilde{y} = I_k$.

2.1 Spectral clustering

In spectral clustering (Shi and Malik 2000; Luxburg 2007; Ng et al. 2001; Dhillon et al. 2004), the data set is represented by a weighted graph $G = (V, E)$ in which the vertices in V correspond to data points, and the edges in E characterize the similarities between data points. The weights of the edges are usually encoded into the adjacency matrix W. Several constructions of similarity graph are regularly used, such as the ϵ-neighborhood graph and the k-nearest neighbor graph (Luxburg 2007).

Spectral clustering is based on the idea of graph cuts, and different graph cut measures have been defined. Two popular approaches are to maximize the average association and to minimize the normalized cut (Shi and Malik 2000). For two subsets, $\pi_p, \pi_q \in \Pi$, the cut between π_p and π_q is defined as $cut(\pi_p, \pi_q) = \sum_{i \in \pi_p, j \in \pi_q} W(i, j)$. Then the k-way average association (AA) and the k-way normalized cut (NC) can be written as

$$\text{AA} = \sum_{l=1}^{k} \frac{cut(\pi_l, \pi_l)}{|\pi_l|}, \quad \text{NC} = \sum_{l=1}^{k} \frac{cut(\pi_l, \Pi \setminus \pi_l)}{cut(\pi_l, \Pi)}, \tag{1}$$

where $\setminus$ denotes the set minus operation. In Chi et al. (2009), the negated average association is defined as $\text{NA} = \text{Tr}(W) - AA$. Note that the average association characterizes the within cluster association, while the normalized cut captures the between cluster separation. Furthermore, maximizing the average association is equivalent to minimizing the negated average association. Hence, the negated average association will be used throughout this paper.

It has been shown (Shi and Malik 2000) that exact minimization of common graph cut measures, such as the normalized cut and the negated average association, is an intractable problem. Hence, a two-step procedure is commonly employed in spectral clustering. In the first step, the graph cut problems are relaxed to a trace optimization problem, whose solution typically can be obtained by computing the eigendecomposition of the graph Laplacian matrices (Luxburg 2007; Chung 1997). Then in the second step, the final clustering results are generated by clustering the solution

of the relaxed problem. The focus of this paper is on how to incorporate smoothness constraints into the first step, the second step is outside the scope of this paper.

2.2 Spectral co-clustering

In Dhillon (2001), Zha et al. (2001), the spectral clustering formalism is extended to solve co-clustering problems. Given a data matrix $A \in \mathbb{R}^{m \times n}$, such as the word-by-document matrix, a bipartite graph is constructed, where the two sets of vertices correspond to the rows and the columns, respectively. Then the co-clustering problem is reduced to perform graph cuts on this bipartite graph. Formally, the similarity matrix of the bipartite graph can be written as

$$W = \begin{bmatrix} 0 & A \\ A^T & 0 \end{bmatrix}. \tag{2}$$

A variety of graph cut criteria can then be applied to partition the bipartite graph. For example, when the normalized cut is used, the Laplacian matrix and the degree matrix for this bipartite graph can be written as

$$L = \begin{bmatrix} D_1 & -A \\ -A^T & D_2 \end{bmatrix}, \quad D = \begin{bmatrix} D_1 & 0 \\ 0 & D_2 \end{bmatrix}, \tag{3}$$

where D_1 and D_2 are diagonal matrices whose diagonal elements are defined as

$$D_1(ii) = \sum_j A_{ij}, \quad D_2(jj) = \sum_i A_{ij}.$$

Then the normalized cut criterion can be relaxed, and the solution for the relaxed problem can be obtained by solving the following eigenvalue problem:

$$\begin{bmatrix} D_1 & -A \\ -A^T & D_2 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \lambda \begin{bmatrix} D_1 & 0 \\ 0 & D_2 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}, \tag{4}$$

where $x \in \mathbb{R}^m$ and $y \in \mathbb{R}^n$ are the relaxed row and column cluster indicator matrices, respectively.

2.3 Evolutionary clustering

When the data matrices evolve along the temporal dimension, it is desirable to capture the temporal smoothness in clustering analysis. Recently, several evolutionary clustering methods have been developed to cluster time-varying data by incorporating temporal smoothness constraints directly into the clustering framework (Chakrabarti et al. 2006; Chi et al. 2009; Lin et al. 2009; Tianbing et al. 2012).

In Chi et al. (2009), two main frameworks, known as preserving cluster quality (PCQ) and preserving cluster membership (PCM), are proposed to incorporate temporal smoothness. In these two formulations, the cost functions contain two terms, known as the snapshot cost (CS) and the temporal cost (CT) as $\text{Cost} = \alpha \cdot \text{CS} + (1-\alpha)\text{CT}$, where $0 \leq \alpha \leq 1$ is a tunable parameter. In this formulation, the snapshot cost captures the clustering quality on the current data matrix, while the temporal cost encourages the temporal smoothness with respect to either historic data or historic clustering results. The main difference between PCQ and PCM lies in the definitions of the temporal costs. Specifically, the temporal cost in PCQ is devised to encode the consistency between current clustering results with historic data, while that in PCM is used to encourage temporal smoothness between current and historic clustering results.

Let Y_t denotes the cluster indicator matrix for time t, then the objective function for PCQ can be expressed as $\text{Cost}_{\text{PCQ}} = \alpha \cdot \text{Cost}_t|_{Y_t} + (1-\alpha) \cdot \text{Cost}_{t-1}|_{Y_t}$, where $\text{Cost}_t|_{Y_t}$ and $\text{Cost}_{t-1}|_{Y_t}$ denote the costs of applying the clustering results in Y_t to the data at time points t and $t-1$, respectively. In contrast, the temporal cost in PCM is expressed as the difference between the current and the historic clustering results, leading to the following overall objective function $\text{Cost}_{\text{PCM}} = \alpha \cdot \text{Cost}_t|_{Y_t} + (1-\alpha) \cdot \text{dist}(Y_t, Y_{t-1})$, where $\text{dist}(\cdot, \cdot)$ denotes certain distance measure.

Following the soft clustering framework proposed in Yu et al. (2006), an evolutionary clustering method based on nonnegative matrix factorization (NMF) has been developed in Lin et al. (2009). Let W_t be the similarity matrix for time point t, the objective function for evolutionary clustering in Lin et al. (2009) can be expressed as

$$\begin{aligned} \text{Cost}_{\text{NMF}} = \alpha \cdot D\left(W_t \| X_t \Lambda_t X_t^T\right) \\ + (1-\alpha) \cdot D(X_{t-1}\Lambda_{t-1} \| X_t \Lambda_t), \end{aligned} \tag{5}$$

where $D(\cdot\|\cdot)$ is the KL-divergence, X_t is the soft clustering indicator matrix, and Λ_t is a diagonal matrix. An iterative procedure is devised to compute the solution. It is also shown in Lin et al. (2009) that the proposed method can be interpreted from the perspective of probabilistic generative models.

3 Related work and extensions

Following the evolutionary spectral clustering framework in Chi et al. (2009), two spectral methods for evolutionary co-clustering have been proposed in Green et al. (2011). In this section, we systematically extend the spectral methods in Green et al. (2011) using two different graph cut criteria, leading to four different methods for capturing the temporal smoothness. Our experimental results in Sect. 5 show that the probabilistic model proposed in this paper consistently outperforms the spectral methods.

3.1 Preserving co-cluster quality

In preserving co-cluster quality (PCCQ), the temporal cost measures the quality of current co-clustering results when applied to historic data. In the following, we describe

the PCCQ formalism using both the negated average association and the normalized cut criteria.

3.1.1 Negated average association

Given a data matrix $A \in \mathbb{R}^{m \times n}$, the negated average association objective function in co-clustering can be written as

$$NA = \mathrm{Tr}(W) - \mathrm{Tr}\left(\tilde{R}^T W \tilde{R}\right), \tag{6}$$

where $\tilde{R} \in \mathbb{R}^{(m+n)\times k}$ is the normalized co-cluster indicator matrix, W is defined in Eq. (2) and denotes the similarity matrix associated with the bipartite graph. Writing $\tilde{R} = [P^T, Q^T]^T$, where $P \in \mathbb{R}^{m\times k}$ and $Q \in \mathbb{R}^{n\times k}$ are the row and column cluster indicator matrices, respectively, and substituting W into Eq. (6), we obtain

$$NA = -\mathrm{Tr}\left(P^T A^T Q + P^T A Q\right) = -2\mathrm{Tr}\left(P^T A Q\right). \tag{7}$$

We propose to employ the following cost function for the PCCQ evolutionary co-clustering formalism based on negated average association:

$$\begin{aligned} \mathrm{NA}_{\mathrm{PCCQ}} &= \alpha \cdot NA_t|_{\tilde{R}_t} + (1-\alpha)\cdot NA_{t-1}|_{\tilde{R}_t} \\ &= -\mathrm{Tr}\left(P_t^T \left(\alpha A_t + (1-\alpha)A_{t-1}\right) Q_t\right), \end{aligned}$$

where A_t, P_t, and Q_t denote the corresponding matrices for time point t. Since solving the above problem exactly is intractable, we propose to relax the constraints on the entries in P_t and Q_t while keeping the orthonormality constraints. It follows from the spectral co-clustering formalism (Dhillon 2001) that columns of the optimal P_t^* and Q_t^* that minimize the relaxed problem are given by the k principal left and right, respectively, singular vectors of the matrix $\alpha A_t + (1-\alpha)A_{t-1}$.

3.1.2 Normalized cut

It follows from Proposition 1 in Bach and Jordan (2006) that the normalized cut criterion can be expressed equivalently as

$$NC = k - \mathrm{Tr}\left(S^T (D^{-\frac{1}{2}} W D^{-\frac{1}{2}}) S\right), \tag{8}$$

where

$$D = \begin{bmatrix} D_1 & 0 \\ 0 & D_2 \end{bmatrix}, \qquad W = \begin{bmatrix} 0 & A \\ A^T & 0 \end{bmatrix}, \tag{9}$$

and $S \in \mathbb{R}^{(m+n)\times k}$ satisfies two conditions: (a) the columns of $D^{-1/2}S$ are piecewise constant with respect to R, and (b) $S^T S = I$. Let $S = [E^T, F^T]^T$, where $E \in \mathbb{R}^{m\times k}$

and $F \in \mathbb{R}^{n\times k}$, then the normalized cut criterion in Eq. (8) can be written as $NC = k - 2\mathrm{Tr}\left(E^T(D_1^{-1/2}AD_2^{-1/2})F\right)$.

We propose to employ the following cost function in PCCQ under the normalized cut criterion:

$$\begin{aligned}\mathrm{NC}_{\mathrm{PCCQ}} &= \alpha \cdot NC_t|_{S_t} + (1-\alpha)\cdot NC_{t-1}|_{S_t} \\ &= k - 2\mathrm{Tr}\left(E_t^T(\alpha D_{1,t}^{-1/2}A_t D_{2,t}^{-1/2} \right. \\ &\quad \left. +\ (1-\alpha)D_{1,t-1}^{-1/2}A_{t-1}D_{2,t-1}^{-1/2})F_t\right),\end{aligned}$$

where $D_{1,t}$ and $D_{2,t}$ are the diagonal matrices at time t. Similar to the case of negated average association, we relax the constraints on the entries of E_t and F_t while keep the orthonormality constraints. It can be verified that columns of the optimal E_t^* and F_t^* that minimize the relaxed problem consist of the principal left and right, respectively, singular vectors of the matrix $\alpha D_{1,t}^{-1/2}A_t D_{2,t}^{-1/2} + (1-\alpha)D_{1,t-1}^{-1/2}A_{t-1}D_{2,t-1}^{-1/2}$. Then the rows of the matrix $\begin{bmatrix} D_{1,t}^{-1/2}E_t^* \\ D_{2,t}^{-1/2}F_t^* \end{bmatrix}$ are clustered to identify co-clusters.

3.2 Preserving co-cluster membership

In preserving co-cluster membership (PCCM), the temporal cost measures the consistency between temporally adjacent co-clustering results. Let U_t and V_t denote the solutions of the relaxed problems at time point t as described in Sect. 3.1. Note that columns of U_t and V_t are the left and right singular vectors, respectively, of certain matrix. Since the singular vectors of a matrix may not be unique (Golub and van Loan 1996), we cannot require U_t and U_{t-1} to be similar and V_t and V_{t-1} to be similar. However, it is known that $U_t V_t^T$ is unique in all cases. Hence, we propose to employ the following temporal cost in PCCM:

$$\mathrm{CT}_{\mathrm{PCCM}} = \|U_t V_t^T - U_{t-1}V_{t-1}^T\|_F^2. \tag{10}$$

3.2.1 Negated average association

By using the temporal cost in Eq. (10) to quantify the smoothness, we propose the following overall cost function for PCCM under the negated average association criterion:

$$\begin{aligned}\mathrm{NA}_{\mathrm{PCCM}} &= \alpha\cdot \mathrm{CS}_{\mathrm{NA}} + (1-\alpha)\cdot \mathrm{CT}_{\mathrm{PCCM}} \\ &= 2(1-\alpha)k \\ &\quad - 2\mathrm{Tr}\left(U_t^T\left(\alpha A_t + (1-\alpha)U_{t-1}V_{t-1}^T\right)V_t\right).\end{aligned}$$

Minimizing NA_{PCCM} is equivalent to maximizing $\text{Tr}\left(U_t^T \left(\alpha A_t + (1-\alpha)U_{t-1}V_{t-1}^T\right) V_t\right)$. Hence, columns of the optimal U_t^* and V_t^* consist of the principal left and right singular vectors, respectively, of the matrix $\alpha A_t + (1-\alpha)U_{t-1}V_{t-1}^T$.

3.2.2 *Normalized cut*

When the temporal cost in Eq. (10) is used along with the normalized cut criterion, we obtain the following problem:

$$\begin{aligned}\text{NC}_{\text{PCCM}} = {} & (2-\alpha)k \\ & - 2\text{Tr}\left(U_t^T \left(\alpha D_{1,t}^{-1/2} A_t D_{2,t}^{-1/2} + (1-\alpha)U_{t-1}V_{t-1}^T\right) V_t\right).\end{aligned}$$

Minimizing NC_{PCCM} is equivalent to maximizing

$$\text{Tr}\left(U_t^T \left(\alpha D_{1,t}^{-1/2} A_t D_{2,t}^{-1/2} + (1-\alpha)U_{t-1}V_{t-1}^T\right) V_t\right).$$

Hence, columns of the optimal U_t^* and V_t^* consist of the principal left and right singular vectors, respectively, of the matrix $\alpha D_{1,t}^{-1/2} A_t D_{2,t}^{-1/2} + (1-\alpha)U_{t-1}V_{t-1}^T$. The final co-clusters are obtained by clustering the rows of the matrix $\begin{bmatrix} D_{1,t}^{-1/2}U_t^* \\ D_{2,t}^{-1/2}V_t^* \end{bmatrix}$.

4 Evolutionary soft co-clustering

Although both co-clustering and evolutionary clustering have been intensively studied, the field of evolutionary co-clustering remains largely unexplored (Green et al. 2011). In addition, prior method (discussed in Sect. 3) employs singular value decomposition (SVD) in computing the solutions of relaxed problems. In many applications, such as image and text analysis, the original data matrices are nonnegative. A factorization such as SVD produces factors containing negative entries. This leads to complex cancelations between positive and negative numbers, and the results are usually difficult to interpret (Lee and Seung 1999). We address this challenge by proposing a probabilistic model for evolutionary co-clustering in this section. This model results in nonnegative factors, thereby overcoming the limitation of spectral methods. In addition, the probabilities can be interpreted to produce soft co-clusters.

4.1 The proposed model

In the proposed model, we assume that the similarity matrix W_t of the bipartite graph can be factorized as

$$W_t = H_t \tilde{H}_t, \tag{11}$$

where

$$W_t = \begin{bmatrix} 0 & A_t \\ {A_t}^T & 0 \end{bmatrix}, \tag{12}$$

$A_t \in \mathbb{R}^{m \times n}$ is the data matrix,

$$H_t = \begin{bmatrix} H_{1,t} & 0 \\ 0 & H_{2,t} \end{bmatrix}, \quad \tilde{H}_t = \begin{bmatrix} 0 & H_{2,t}^T \\ H_{1,t}^T & 0 \end{bmatrix}, \tag{13}$$

where $H_t \in \mathbb{R}^{(m+n)\times(2k)}$, $\tilde{H}_t \in \mathbb{R}^{(2k)\times(m+n)}$, $H_{1,t} \in \mathbb{R}^{m\times k}$ denotes the row cluster indicator matrix, and $H_{2,t} \in \mathbb{R}^{n\times k}$ denotes the column cluster indicator matrix. It follows that

$$H_t \tilde{H}_t = \begin{bmatrix} 0 & H_{1,t} H_{2,t}^T \\ \left(H_{1,t} H_{2,t}^T\right)^T & 0 \end{bmatrix}, \tag{14}$$

which matches the structure of W_t in Eq. (12).

In the proposed probabilistic model, the similarity matrix W_t is generated via a two-step process. In the first step, $H_t \tilde{H}_t$ is generated based on the co-clustering results $H_{t-1}\tilde{H}_{t-1}$ at time point $t-1$ using $P(H_t\tilde{H}_t|H_{t-1}\tilde{H}_{t-1})$. In the second step, the observed similarity matrix W_t is generated based on $H_t\tilde{H}_t$ using $P(W_t|H_t\tilde{H}_t)$. Following Lin et al. (2009), we employ the Dirichlet and multinomial distributions in the first and second steps, respectively.

Specifically, we model $\log P(W_t|H_t\tilde{H}_t)$ by a multinomial distribution with parameter $\omega_{t,ij} = (H_t\tilde{H}_t)_{ij}$ as

$$\begin{aligned} \log P(W_t|H_t\tilde{H}_t) &= \log \frac{\Gamma\left(1+\sum_{ij}(W_t)_{ij}\right)}{\prod_{ij}\Gamma\left(1+(W_t)_{ij}\right)} \prod_{ij} \omega_{t,ij}^{(W_t)_{ij}} \\ &\propto \sum_{ij} (W_t)_{ij} \log\left(H_t\tilde{H}_t\right)_{ij} \\ &= 2\sum_{ij}(A_t)_{ij} \log\left(H_{1,t}H_{2,t}^T\right)_{ij}, \end{aligned}$$

where the last step follows from the symmetry of matrices W_t and $H_t\tilde{H}_t$.

Since the conjugate prior for the multinomial distribution is the Dirichlet distribution, it is natural to use a Dirichlet distribution to model $P(H_t\tilde{H}_t|H_{t-1}\tilde{H}_{t-1})$. Specifically, we assume that $H_t\tilde{H}_t$ follows a Dirichlet distribution with parameters $\psi_t = \nu \cdot \mathrm{vec}(H_{t-1}\tilde{H}_{t-1}) + \mathbf{1}$, where $\mathbf{1}$ is a vector of ones of appropriate length. Through the parameters ψ_t, the distribution $P(H_t\tilde{H}_t)$ at time t is determined by $H_{t-1}\tilde{H}_{t-1}$ at time $t-1$. Under this model, we have

$$\log P\left(H_t\tilde{H}_t|H_{t-1}\tilde{H}_{t-1}\right) = \log\frac{\Gamma(\sum_k \psi_{t,k})}{\prod_k \Gamma(\psi_{t,k})}\prod_{ij}\left(H_t\tilde{H}_t\right)_{ij}^{\nu\cdot\left(H_{t-1}\tilde{H}_{t-1}\right)_{ij}}$$
$$\propto 2\nu\sum_{ij}\left(H_{1,t-1}H_{2,t-1}^T\right)_{ij}\log\left(H_{1,t}H_{2,t}^T\right)_{ij}.$$

This gives rise to the following log likelihood function of observing the current weight matrix W_t:

$$\begin{aligned} L &= \log P\left(W_t|H_t\tilde{H}_t\right) + \nu\log P\left(H_t\tilde{H}_t|H_{t-1}\tilde{H}_{t-1}\right) \\ &= 2\sum_{ij}(A_t)_{ij}\log\left(H_{1,t}H_{2,t}^T\right)_{ij} \\ &\quad + 2\nu\sum_{ij}\left(H_{1,t-1}H_{2,t-1}^T\right)_{ij}\log\left(H_{1,t}H_{2,t}^T\right)_{ij}, \end{aligned} \tag{15}$$

where parameter ν controls the temporal smoothness.

4.2 An EM algorithm

To maximize the log likelihood in Eq. (15), we derive an EM algorithm in the following. To simplify notation, we omit the subscript t when the time information is clear from context. We use variables with hat (e.g., $\hat{h}_{1;ik}$ and $\hat{H}_1$) to denote the values obtained from the previous iteration.

In the E-step, we compute the expectation as

$$\phi_{ijk} = \frac{\hat{h}_{1;ik}\hat{h}_{2;jk}}{\left(\hat{H}_1\hat{H}_2^T\right)_{ij}}, \tag{16}$$

where $\sum_k \phi_{ijk} = 1$, $\hat{h}_{1;ik}$ and $\hat{h}_{2;jk}$ denote the ikth and the jkth entries, respectively, of H_1 and H_2 computed from the previous iteration.

In the M-step, we maximize the expectation of log likelihood with respect to $\Phi = (\Phi)_{ijk}$

$$E_\Phi[L] = 2\times\sum_{ijk}\phi_{ijk}a_{ij}^t\log\left(h_{1;ik}^t h_{2;jk}^t\right) + 2\times\nu\sum_{ijk}h_{1;ik}^{t-1}h_{2;jk}^{t-1}\log\left(h_{1;ik}^t h_{2;jk}^t\right), \tag{17}$$

where the superscripts t and $t-1$ are used to denote variables at the corresponding time points. To facilitate a probabilistic interpretation of the co-clustering results, we impose the following normalization constraints:

$$\sum_i h_{1;ik}^t = 1, \quad \sum_j h_{2;jk}^t = 1.$$

By using Lagrange multipliers for these constraints, it can be shown that the following update rules will monotonically increase the expected log likelihood defined in Eq. (17), thereby leading to convergence to an locally optimal solution (Yu et al. 2006):

$$h_{1;ik} \leftarrow 2 \times \sum_j \frac{\hat{h}_{1;ik}\hat{h}_{2;jk}a^t_{ij}}{\left(\hat{H}_1\hat{H}_2^T\right)_{ij}} + 2 \times \nu \sum_j \left(h^{t-1}_{1;ik}h^{t-1}_{2;jk}\right),$$

$$h_{2;jk} \leftarrow 2 \times \sum_i \frac{\hat{h}_{1;ik}\hat{h}_{2;jk}a^t_{ij}}{\left(\hat{H}_1\hat{H}_2^T\right)_{ij}} + 2 \times \nu \sum_i \left(h^{t-1}_{1;ik}h^{t-1}_{2;jk}\right).$$

The results are then normalized such that $\sum_i h^t_{1;ik} = 1$ and $\sum_j h^t_{2;jk} = 1$, $\forall k$.

The E-step and M-step are repeated until a locally optimal solution is obtained. Then the matrices $H_{1,t}$ and $H_{2,t}$ can be used as row and column co-cluster indicator matrices, respectively, to obtain soft co-clustering results. Our experimental results show that this probabilistic model achieves superior performance on both synthetic and real data sets.

4.3 Convergence of the EM algorithm

In this section, we establish the convergence of the proposed EM algorithm. To this end, we need to show that the cost function in Eq. (15) is non-decreasing under the update rules proposed in Sect. 4.2. To simplify the notation, we use H_t to denote $\{H_{1,t}, H_{2,t}\}$. Given H_{t-1}, the objective in Eq. (15) is a function of H_t as

$$L(H_t) = 2\sum_{ij} a^t_{ij} \log\left(H_{1,t}H_{2,t}^T\right)_{ij} + 2\nu \sum_{ij}\left(H_{1,t-1}H_{2,t-1}^T\right)_{ij} \log\left(H_{1,t}H_{2,t}^T\right)_{ij}.$$

To show that the above objective is non-decreasing, we need to construct an auxiliary function $f(\cdot,\cdot)$ so that

$$f(H_t, H_t) = L(H_t),$$
$$f(H_t, H_t^*) \leq L(H_t),$$

for any H_t^*. Then, we need to show that the proposed update rule satisfies

$$H^{p+1} = \arg\max_H f(H, H^p),$$

where the superscript p denotes the value at the p-th iteration. This would allow us to show that the objective function does not decrease, since

$$L(H^{p+1}) \geq f(H^{p+1}, H^p) \geq f(H^p, H^p) = L(H^p).$$

To this end, we define

$$f(H_t, H_t^*) = \sum_{ijk} 2a_{ij}^t \frac{h_{1;ik}^{t,*} h_{2;jk}^{t,*}}{\sum_l h_{1;il}^{t,*} h_{2;jl}^{t,*}} \left[\log\left(h_{1;ik}^t h_{2;jk}^t\right) - \log \frac{h_{1;ik}^{t,*} h_{2;jk}^{t,*}}{\sum_l h_{1;il}^{t,*} h_{2;jl}^{t,*}} \right] + 2\nu \sum_{ijk} h_{1;ik}^{t-1} h_{2;jk}^{t-1} \log\left(h_{1;ik}^t h_{2;jk}^t\right),$$

where we use the superscripts t and $t-1$ to denote variables at the corresponding time points. The following derivations verify that $f(H_t, H_t) = L(H_t)$:

$$\begin{aligned} f(H_t, H_t) &= \sum_{ijk} 2a_{ij}^t \frac{h_{1;ik}^t h_{2;jk}^t}{\sum_l h_{1;il}^t h_{2;jl}^t} \left[\log\left(h_{1;ik}^t h_{2;jk}^t\right) - log \frac{h_{1;ik}^t h_{2;jk}^t}{\sum_l h_{1;il}^t h_{2;jl}^t} \right] \\ &\quad + 2\nu \sum_{ijk} h_{1;ik}^{t-1} h_{2;jk}^{t-1} log\left(h_{1;ik}^t h_{2;jk}^t\right) \\ &= \sum_{ijk} 2a_{ij}^t \frac{h_{1;ik}^t h_{2;jk}^t}{\sum_l h_{1;il}^t h_{2;jl}^t} log \sum_l h_{1;il}^t h_{2;jl}^t \\ &\quad + 2\nu \sum_{ijk} h_{1;ik}^{t-1} h_{2;jk}^{t-1} log\left(h_{1;ik}^t h_{2;jk}^t\right) \\ &= \sum_{ij} 2a_{ij}^t \log \sum_l h_{1;il}^t h_{2;jl}^t + 2\nu \sum_{ijk} h_{1;ik}^{t-1} h_{2;jk}^{t-1} \log\left(h_{1;ik}^t h_{2;jk}^t\right) \\ &= L(H_t). \end{aligned}$$

To prove the inequality $f(H_t, H_t^*) \leq L(H_t)$, we need to show that the following inequality is satisfied for all $i,\ j$:

$$\sum_k \frac{h_{1;ik}^{t,*} h_{2;jk}^{t,*}}{\sum_l h_{1;il}^{t,*} h_{2;jl}^{t,*}} \left[\log\left(h_{1;ik}^t h_{2;jk}^t\right) - \log \frac{h_{1;ik}^{t,*} h_{2;jk}^{t,*}}{\sum_l h_{1;il}^{t,*} h_{2;jl}^{t,*}} \right] \leq \log \sum_l h_{1;il}^t h_{2;jl}^t, \forall i,\ j.$$

We denote $\beta_k = \frac{h_{1;ik}^{t,*} h_{2;jk}^{t,*}}{\sum_l h_{1;il}^{t,*} h_{2;jl}^{t,*}}$ and $X_k = h_{1;ik}^t h_{2;jk}^t$. Then, the above inequality can be written as

$$\sum_k \beta_k \left[log\left(\frac{X_k}{\beta_k}\right) \right] \leq \log\left(\sum_k \left(\beta_k \cdot \frac{X_k}{\beta_k} \right) \right) = \log \sum_k X_k. \tag{18}$$

The inequality in Eq. (18) follows from the concavity of log function. This proves that $f(H_t, H_t^*) \leq L(H_t)$. Finally, the update rules can be obtained by setting the derivative of f with respect to $h_{1;ik}^t$ and $h_{2;jk}^t$ to zero respectively.

4.4 Co-cluster evolution

An unique property of the proposed probabilistic model is that the identified co-clusters can be related across time points, giving rise to co-cluster evolution. Figure 1 shows how co-clusters evolve for a 5×4 example data matrix, where r_1 to r_5 correspond to the five rows, c_1 to c_4 correspond to the four columns, and R_1 to R_4 denote the co-clusters. In panel (a), the matrix is co-clustered into 3 co-clusters as indicated by the dashed ovals. At time t in panel (b), the data is clustered into 4 co-clusters. The row and column co-clusters across time points can be related naturally by considering the sharing of rows and columns between co-clusters. This is illustrated in panels (c) and (d), which depict how the row and column co-clusters, respectively, evolves from time points $t-1$ to t. Note that the co-cluster evolution is a direct product of the soft co-cluster assignment proposed in this paper. This demonstrates that the soft co-cluster assignment formalism captures additional temporal dynamics, which have been ignored by prior methods. More importantly, we show in Sect. 5 that our evolutionary soft co-clustering formulation outperforms prior methods consistently.

5 Experimental evaluation

5.1 Synthetic data # 1

We generate a synthetic data set with 7 time-steps and 5 co-clusters, each containing 200 instances and 10 features. At $t = 0$, the entries corresponding to rows and columns

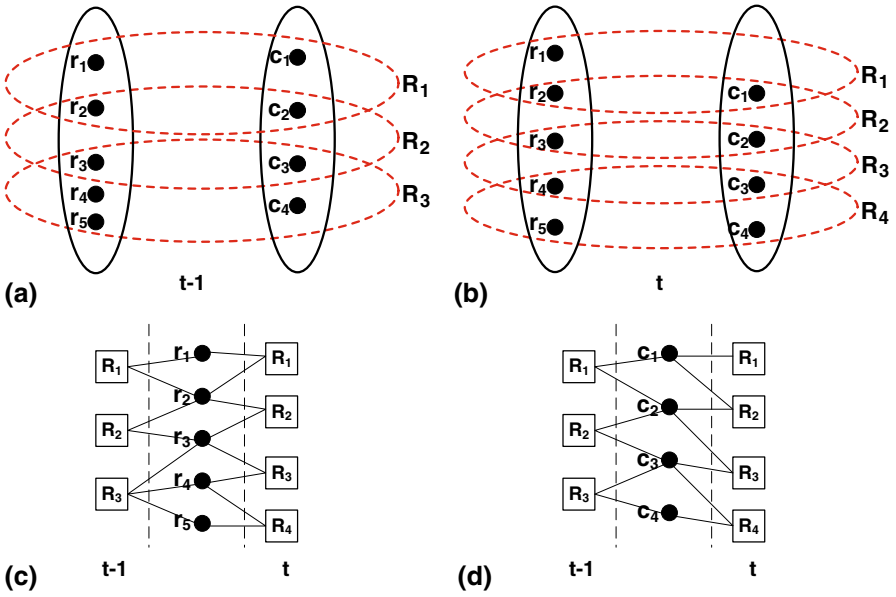


Fig. 1 Illustration of co-cluster evolution. **a** and **b** show the co-clustering results at time points $t-1$ and t, respectively. **c** and **d** show the row and column co-cluster evolution, respectively, between time points $t-1$ and t. See text for detailed explanations

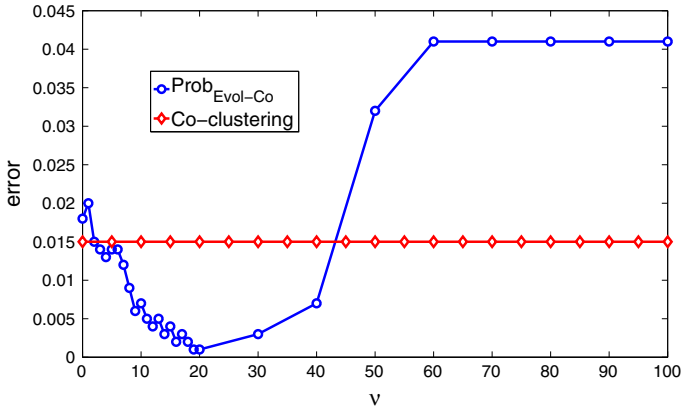


Fig. 2 Performance comparison between the proposed probabilistic model ($\text{Prob}_{\text{Evol-Co}}$) with that of the co-clustering method when ν varies from 0 to 100

in the same co-cluster are set to nonzero with a high probability p while other entries are set to nonzero with a low probability q which satisfies $p = 4q$ and $p + 4q = 1$. The data at $t = 1$ are generated by adding a Gaussian noise to each entry of the data at $t = 0$. To simulate the evolving nature of the data, 20 % of the instances in co-cluster I are set to be weakly correlated to features in co-cluster III at $t = 2$. The level of correlation by the same set of instances is increased at $t = 3$ so that they are equally correlated to features in co-clusters I and III. At $t = 4$, this set of instances are no longer correlated to features in co-cluster I, and their correlations with features in co-cluster III are further increased. At $t = 5$, a sudden change occurs and the data matrix at $t = 1$ is restored. At $t = 6$, the size of the data matrix is changed by adding some extra instances to co-cluster I.

To demonstrate the effectiveness of the temporal cost, we compare our formulation with co-clustering method without the temporal cost. We use an error rate as the performance measure, since the co-cluster memberships are known for synthetic data. The performance of the proposed model along with that of the co-clustering method (equivalent to $\nu = 0$) is reported in Fig. 2. It can be observed that when ν is increased from 0 to 20, the error rate drops gradually. When ν is increased beyond 20, the error rate increases gradually. When ν lies in the interval (Chakrabarti et al. 2006; Shewchuk 1996), the proposed method outperforms the co-clustering method significantly. This shows that the evolutionary co-clustering formulation yields improved performance for a large range of ν.

5.2 Synthetic data # 2

The second synthetic data set is generated to evaluate the performance of the proposed model in comparison to prior methods based on spectral learning. This data set contains 50 time-steps, each with 4 co-clusters, and each co-cluster contains 100 instances and 10 features. At $t = 0$, the data set is generated by following the same strategy as

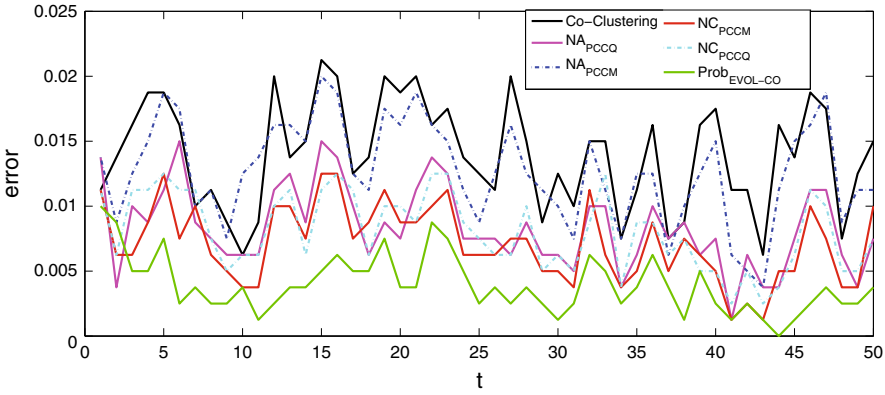


Fig. 3 Performance of the probabilistic model with four methods based on spectral learning and the co-clustering method on synthetic data # 2

the first synthetic data set when $t = 0$. In each of the 0 to 49 time-steps, we add Gaussian noise to the data from previous time-step. We optimize the α and ν values on the synthetic data separately. This set of experiments, including data generation, are repeated 40 times and the average results are reported in Fig. 3 for all time-steps.

We can observe from Fig. 3 that the proposed probabilistic model ($\text{Prob}_{\text{EVOL-CO}}$) consistently outperforms prior methods (i.e., NA_{PCCQ}, NC_{PCCQ}, NA_{PCCM}, and NC_{PCCM}). This demonstrates that the proposed model is very effective in improving performance by requiring the factors to be nonnegative. Similar to the observation in Sect. 5.1, all evolutionary co-clustering approaches outperform co-clustering method consistently across most time-steps. This demonstrates that the temporal cost is effective in improving performance.

5.3 DBLP data

We conduct experiments on the DBLP data to evaluate the proposed methods. The DBLP data (Tong et al. 2008; Wang et al. 2011b) contain the author-conference information for 418,236 authors and 3,571 conferences during 1959–2007. For each year, the author-conference matrix captures how many papers are published by an author in a conference. The author-conference data matrices are very sparse, and we sample 252 conferences spanning 12 main research areas (Internet Computing, Data Mining, Machine Learning, AI, Programming Language, Data Base, Multimedia, Distributed System, Security, Network, Social Network, Operating System) in our experiments. We also remove authors with too few papers, resulting in 4,147 authors from the 252 conferences. We choose the data for ten years (1998–2007) and add the data for two consecutive years, leading to data of five time points.

We apply the probabilistic model to the DBLP data in order to discover the author-conference co-occurrence relationship and their temporal evolution. We set the number of co-clusters to be 12 in the experiments, and this results in 5 major co-clusters and 7 minor co-clusters as shown in Fig. 4. The 5 major co-clusters can be easily identified

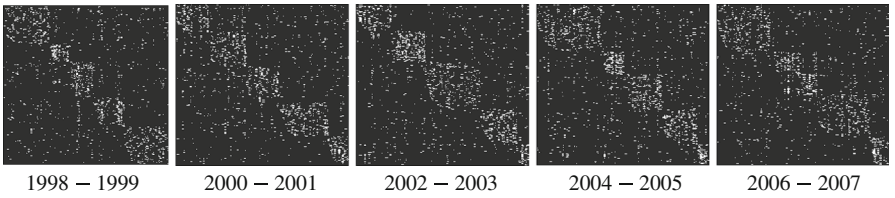


Fig. 4 The block structures identified by the proposed probabilistic model on the DBLP data

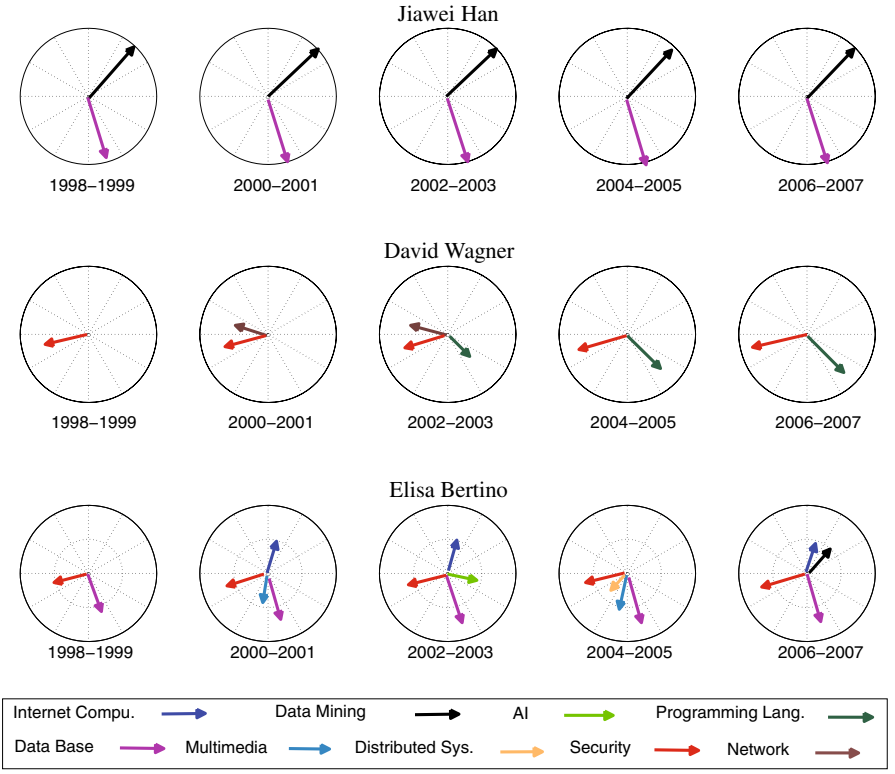


Fig. 5 The evolution patterns of three authors identified by the proposed probabilistic model

from our co-clustering results, and their evolutions are temporally smooth. A close examination of the results shows that related conferences are clustered into the same co-cluster consistently across all time points. For example, the co-cluster for Data Mining always contains KDD, ICDM, SDM etc., and the co-cluster for Data Base always contains SIGMOD, ICDE, VLDB, etc.

We also investigate how the authors' research interests change dynamically over time. In Fig. 5, we plot the results for three authors: Jiawei Han, David Wagner, and Elisa Bertino. For each author and each time point, we distribute the 12 conference categories evenly around a circle, and each category occupies a sector. We then use an arrow pointing to a particular sector to indicate the author's participation in the

conferences in this category, where the level of participation is indicated by the length of the arrow.

It can be observed from Fig. 5 that Jiawei Han was actively participating Data Mining and Data Base conferences across all five time points, and this pattern remains very stable across years. On the other hand, David Wagner showed some change of research interests. He is actively participating Security conferences across all years. During 2000–2001, he developed interests in Network, and this is maintained through 2002–2003 before he smoothly switched to Programming Language. Elisa Bertino showed very dynamic change of research interests during this 10-year period. She is actively participating Data Base and Security conferences across all years. During some period of time, she also participated Internet Computing, Distributed Systems, AI, and Data Mining conferences. These results demonstrate that the proposed methods can identify smooth evolution of author's research interests over years.

6 Application study

To fully exploit the real-world impact of our methods, we further perform a systematic application study on the analysis of *Drosophila* gene expression pattern images.

6.1 Background

Genes are the fundamental elements for regulating many biological activities from cell division to protein composition. Currently, the protein-coding genes of many organisms have been largely identified. However, how these sequences are orchestrated by the regulatory sequences to transform a single cell into a functional organism during development remains largely unknown. Investigations into the spatial and temporal gene expression dynamics are essential for understanding the regulatory biology governing development. Recently, genome-wide spatial gene expression patterns in the model organism fruit fly *Drosophila melanogaster* have been generated using high-throughput RNA in situ hybridization (Tomancak et al. 2002; Lécuyer et al. 2007. These data provide useful information to study the temporal and spatial gene expression patterns and the underlying developmental regulatory networks (Tomancak et al. 2007; Kumar et al. 2002; Frise et al. 2010; Lécuyer and Tomancak 2008).

In order to better understand the spatial and temporal gene regulation during development, we use a geometry-based, standardized mesh-generation method to convert *Drosophila* gene expression patterns into matrix representations. We build a fully automated mesh generation pipeline to map every gene expression pattern into the same geometric space. We then organize the gene expression pattern images at a particular time point as a data matrix in which one dimension represents the genes and the other dimension represents the mesh elements. To identify the co-expressed embryonic domains and the associated genes over different temporal stages, we apply the proposed evolutionary co-clustering model to study the gene expression images in the FlyExpress database (Kumar et al. 2011). Our results show that the co-clusters of mesh elements and genes are correlated with the key events of embryogenesis.

6.2 Intensity-based mapping of geometry to algebraic representation

We employ a mesh generation pipeline to map all the *in situ* hybridization (ISH) images into the same coordinate space with the goal of eliminating the effect of the shape variations (Zhang et al. 2013). To this end, we compute the best-fit ellipse for the boundary of each image using the least squares criterion. This way, a generic ellipse for each time point can be obtained by averaging the fitted ellipses associated with all images in that particular time point. We then generate a mesh on the generic ellipse to obtain a discretized representation. This is achieved by first subdividing the boundary iteratively using linear interpolation so that all the segmented pieces of the boundary are approximately the same. The Delaunay mesh generator software (Shewchuk 1996) is subsequently used to tessellate the interior of the generic ellipse. Once the generic ellipse is meshed, we deform it to each of the individual images at the same time point. More details behind the rationale for the use of triangular meshes can be found in Frise et al. (2010).

We focus on *Drosophila* gene expression pattern images from stage 4 to stage 16, which have been divided into five stage ranges; namely stage 4–6, 7–8, 9–10, 11–12, and 13–16. We collect the images for genes that appear in all five stage ranges. This generates a data set with 2,675 images capturing the expression patterns of 1,878 genes with clearly defined expression boundaries. We preprocess the images using a similar procedure as in Frise et al. (2010) and then apply our tessellation pipeline to obtain triangulated images. Following Frise et al. (2010), we extract the median of gray-level intensities from each mesh element. This converts each triangulated image into an n-dimensional vector. Hence, the images for m genes at a particular time point can be encoded into a data matrix A, in which each row corresponds to a gene, and each column corresponds to a mesh element. We apply the evolutionary co-clustering algorithm on the five data matrices corresponding to five stage ranges to identify the gene and mesh co-clusters simultaneously for multiple temporal time points.

6.3 Evolutionary clustering of mesh elements

We apply our methods to all of the *Drosophila* gene expression pattern images from stages 4–6 to 13–16 to gain insight on the developmental gene co-expression dynamics. Evolutionary co-clustering with different numbers of co-clusters is applied to the five data matrices simultaneously. The results are mapped to the average ellipsoid and color-coded to visualize the co-clusters. In order to make sure that the generated clusters are not the results of data processing artifacts, we randomize the data sets at multiple points of the pipeline. Our results show that the co-expressed domains established via our evolutionary co-clustering algorithm are consistent with many actual embryonic structures. Moreover, we show that the co-clusters of mesh elements and genes have strong correlation with the key events of *Drosophila* embryogenesis.

In Fig. 6, we show the co-clustering results of mesh elements when the number of clusters is varied from 20 to 40 on stage 4–6 data. A number of existing

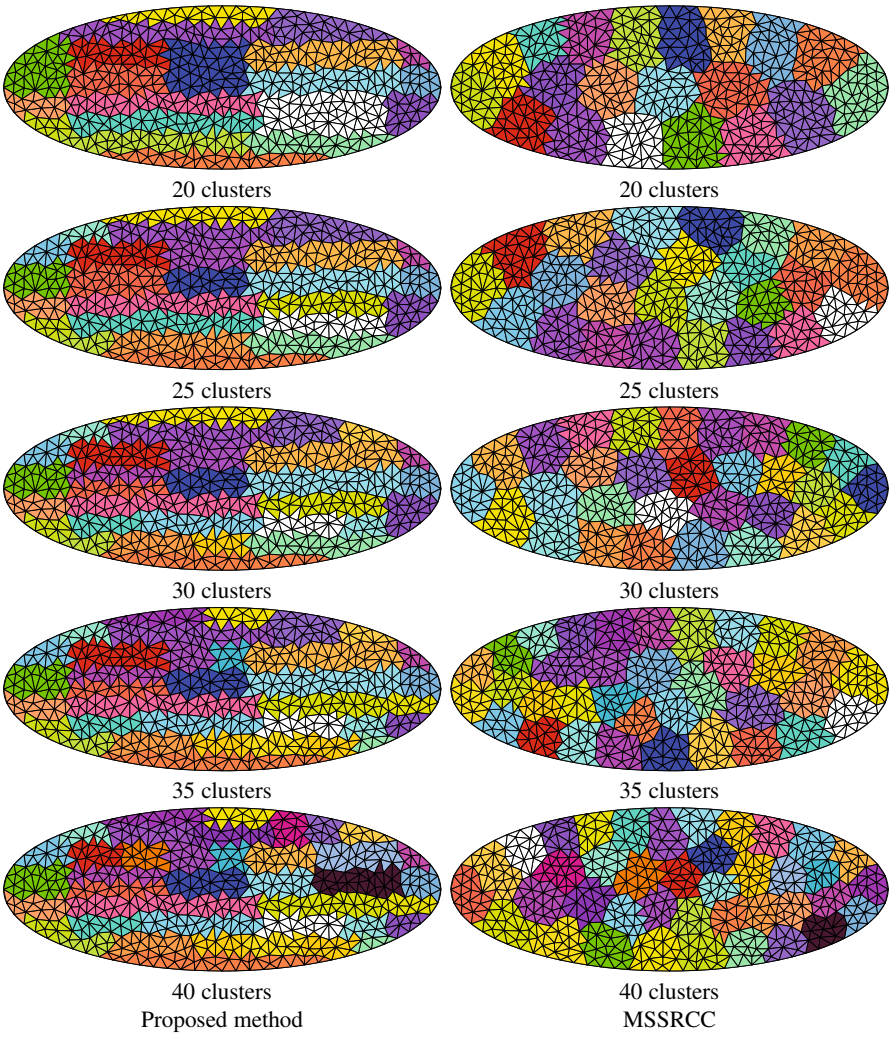


Fig. 6 Clusters of mesh elements when the number of clusters is varied from 20 to 40 with a step size of 5 (*top to bottom*) on stage 4–6 expression patterns. The *left column* shows the results of the proposed method and the *right column* shows the results of NBIN + RI + MSSRCC + LS

co-clustering techniques also aim to identifying the block structures. In particular, we compare our evolutionary co-clustering method with a variant of the minimum sum-squared residue co-clustering (MSSRCC) method (Cho and Dhillon 2008); namely NBIN + RI + MSSRCC + LS, which denotes MSSRCC with random initialization, local search, and data binormalization (Livne and Golub 2004), since different variants of MSSRCC generate similar results. We can observe that the co-clustering boundaries of the proposed method are mostly parallel to the anterior/posterior (A/P) and dorsal/ventral (D/V) axes of the embryo. This is consistent with the underlying biology of *Drosophila* embryonic patterning, which is achieved by two sets of systems

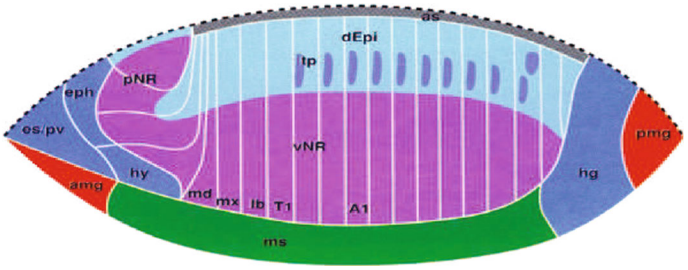


Fig. 7 The fate map of *Drosophila* blastoderm (Hartenstein 1995)

along the horizontal and vertical axis independently (Hartenstein 1995; Fig. 7). Furthermore, as the number of co-clusters is increased, the shape of rectangular cluster generated by our method is continuously preserved (the left column of Fig. 6); namely, new clusters are generated by subdividing existing clusters, and all other clusters are preserved. In comparison, the cluster boundaries generated by MSSRCC do not align with the horizontal or vertical axes. Additionally, the cluster boundaries generated by MSSRCC are mostly not preserved when the number of clusters varies.

In Fig. 8, we show the clustering results generated by our evolutionary co-clustering method and by NBIN + RI + MSSRCC + LS for the five stage range data (i.e., stages 4–6 to 13–16) when the number of clusters is fixed to 35. We can again observe that the clusters generated by our method usually have rectangular shapes whose sides are approximately aligned with the horizontal or vertical axes. In comparison, the results generated by MSSRCC do not have a rectangular shape. More importantly, our evolutionary co-clustering is able to produce smoothly varying clustering boundaries across time points, while MSSRCC is not able to achieve such effect. Note that, theoretically, the EM algorithm might converge to different optimal points when it is initialized to different values. However, we find in experiments that the clustering results are the same when the EM algorithm is randomly initialized multiple times. This empirical evidence shows that the clustering results are not sensitive to the initial values.

6.4 Evolutionary co-clustering of genes and mesh elements

We evaluate the co-clustering of mesh elements and genes and show how they are correlated with developmental events of *Drosophila* embryogenesis. We apply our mesh generation and evolutionary co-clustering methods to the data set of 2675 images of gene expression in stage 4–6. Following Frise et al. (2010), we set the number of co-clusters to 39. We compute the enriched Gene Ontology terms (biological process) (Ashburner et al. 2000) and evaluate the terms with p value < 0.001. We subsequently apply the one-sided significance test and retain the enriched terms with $\geq 90\,\%$ significance. Among the 39 clusters, 22 of them have at least one enriched term. The enriched terms in the 22 clusters are shown in Fig. 9, and the corresponding mesh clusters are given in Fig. 10.

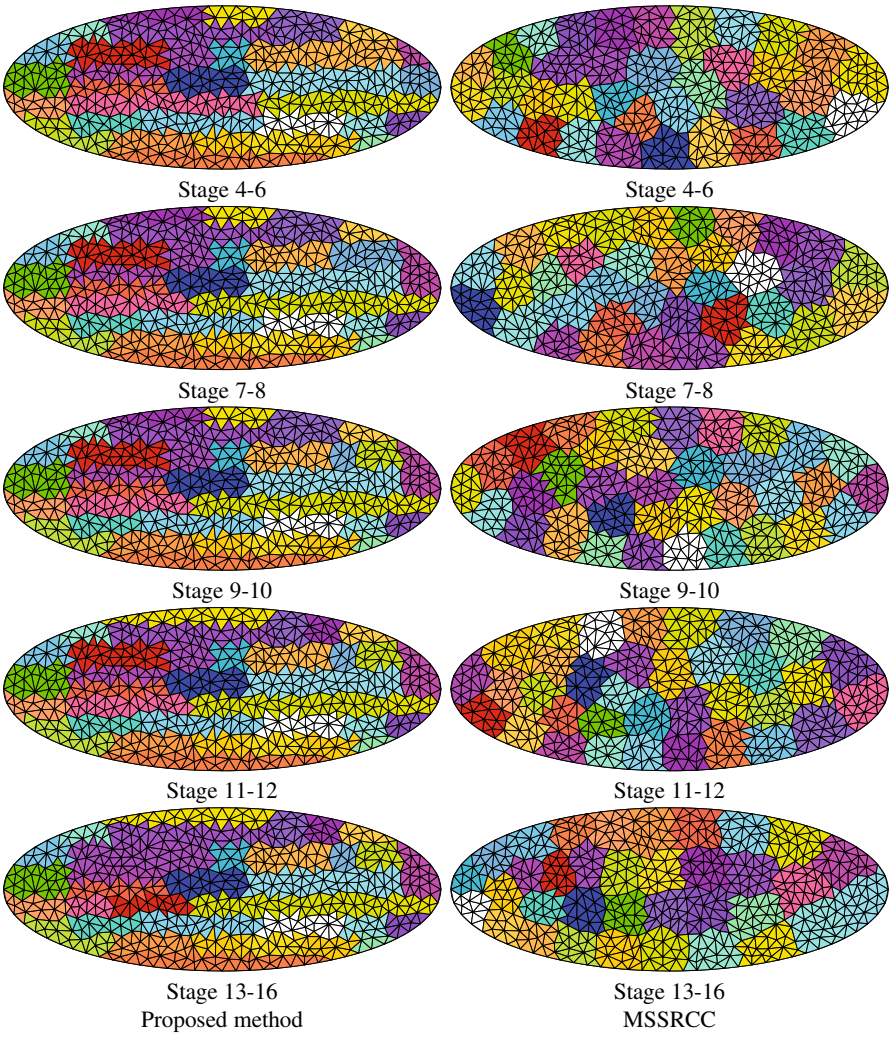


Fig. 8 Clusters of mesh elements when the number of clusters is fixed to 35, and the time points are changed from stages 4–6 to 13–16 (*top to bottom*, stages 4–6, 7–8, 9–10, 11–12, and 13–16). The *left column* shows the results of the proposed method and the *right column* shows the results of NBIN + RI + MSSRCC + LS

We can see that terms such as gene regulation, pattern formation and embryo development appear in the enriched term list. Note that stage 4–6 is the cellularization and gastrulation stage, and thus the enrichment of these terms makes biological sense. With the fixed stage 4–6, we can map the enriched GO terms back into the mesh cluster visualization (Fig. 10). We can see that similar terms are located in spatially adjacent clusters. We also find a subset of well known genes that are activated in the ventral region of the embryo during stage 4–6 containing *twist*, *snail*, *Mes2*, *brinker*, and *tinman*. Our findings are consistent with the biological results reported in Stathopoulos and Levine (2005); Sandmann et al. (2007).

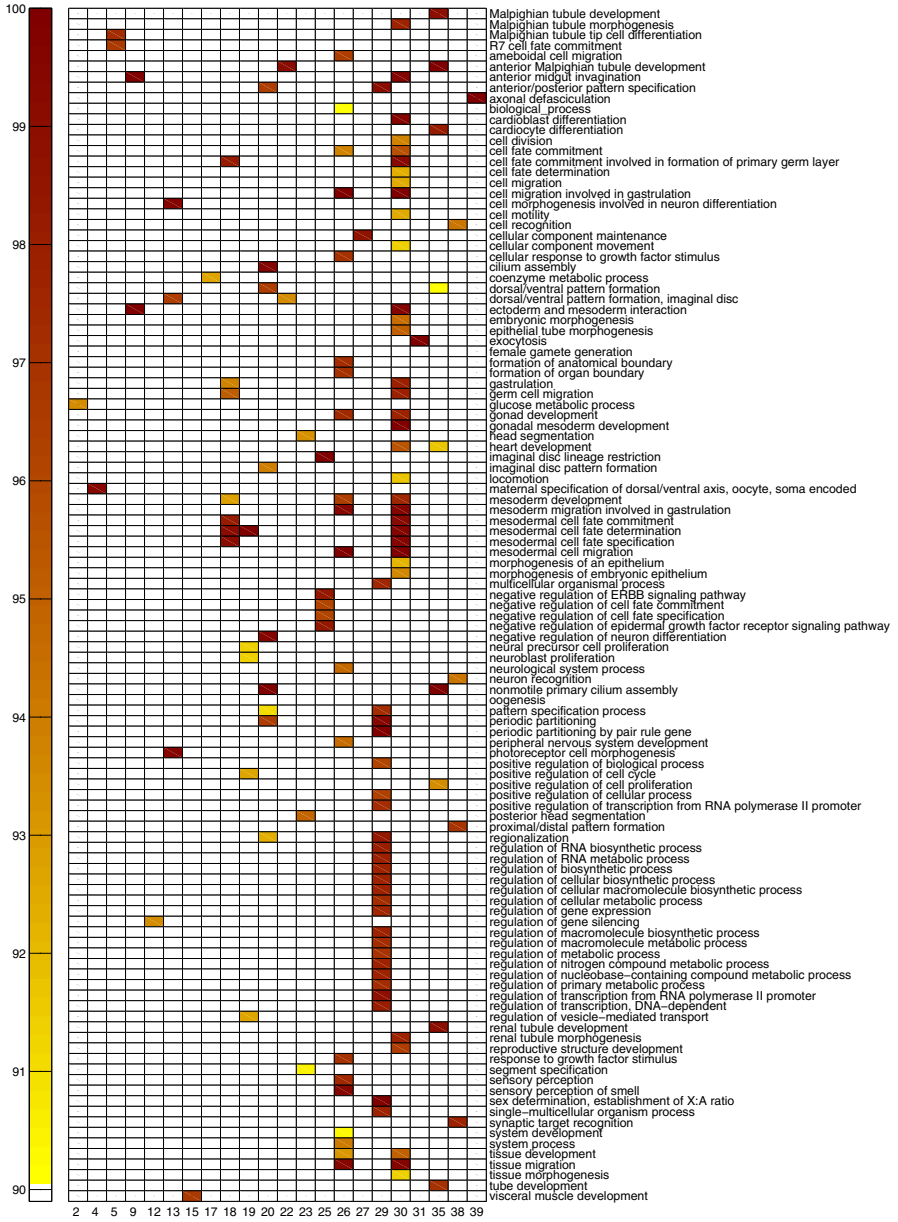


Fig. 9 The clusters with enriched terms and the corresponding terms. We use a p value threshold of 0.001 to obtain the enriched GO terms (biological process) and then apply the one-sided significance test to retain the enriched terms with $\geq 90\,\%$ significance. Figure 10 shows the corresponding mesh clusters

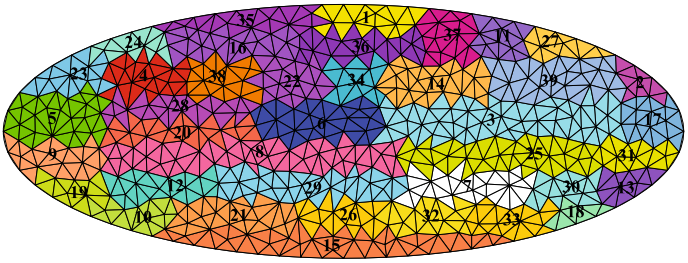


Fig. 10 Mesh clusters when the number of clusters is set to 39. Each mesh cluster element is labeled with the cluster number

7 Conclusions and discussions

This paper studies the evolutionary co-clustering of time-varying data for the identification of smooth block structures. To overcome the limitation of existing methods and enable a probabilistic interpretation of the results, we propose a probabilistic model for evolutionary co-clustering. We propose an EM algorithm to perform maximum likelihood parameter estimation and establish the convergence of this algorithm. The proposed methods are evaluated on both synthetic and real data sets. Results show that the proposed method consistently outperforms prior methods.

We also perform an application study of *Drosophila* gene expression pattern image analysis to demonstrate the impact of our method. We use a new mesh generation pipeline that can more accurately map the expression patterns of many genes into the same coordinate space. We then apply the evolutionary co-clustering algorithm to identify the co-expressed mesh elements and genes across multiple developmental time points. Experimental results indicate that the co-clusters of genes and mesh elements have strong correlation with major embryogenesis events.

In this work, we describe a method for unsupervised learning from bipartite graphs. In many applications, the relational data are more conveniently captured by k-partite graphs (Long et al. 2006). We will extend our methods for unsupervised mining of dynamic k-partite graphs. In addition, in our current work, we assume that the number of co-clusters across all time points is the same. We will extend our method to this more general setting in the future.

Acknowledgments We thank Hanghang Tong and Fei Wang for providing the DBLP data, Yun Chi and Yu-Ru Lin for many insightful discussions. This research was supported in part by NSF Grants DBI-1147134, DBI-1356621, CCF-1139864, CCF-1136538, and CSI-1136536, and by Old Dominion University Office of Research.

References

Aggarwal CC, Han J, Wang J, Yu PS (2003) A framework for clustering evolving data streams. In: Proceedings of the 29th international conference on very large data bases, pp 81–92

Ashburner M, Ball CA, Blake JA, Botstein D, Butler H, Cherry JM, Davis AP, Dolinski K, Dwight SS, Eppig JT, Harris MA, Hill DP, Issel-Tarver L, Kasarskis A, Lewis S, Matese JC, Richardson JE, Ringwald M, Rubin GM, Sherlock G (2000) Gene Ontology: tool for the unification of biology. Nat Genet 25:25–29

Asur S, Parthasarathy S, Ucar D (2007) An event-based framework for characterizing the evolutionary behavior of interaction graphs. In: Proceedings of the 13th ACM SIGKDD international conference on knowledge discovery and data mining, pp 913–921

Bach FR, Jordan MI (2006) Learning spectral clustering, with application to speech separation. J Mach Learn Res 7:1963–2001

Chakrabarti D, Kumar R, Tomkins A (2006) Evolutionary clustering. In: Proceedings of the 12th ACM SIGKDD international conference on knowledge discovery and data mining, pp 554–560

Cheng Y, Church GM (2000) Biclustering of expression data. In: Proceedings of the eighth international conference on intelligent systems for molecular biology, pp 93–103

Chi Y, Song X, Zhou D, Hino K, Tseng BL (2009) On evolutionary spectral clustering. ACM Trans Knowl Discov Data 3:17:1–17:30

Cho H, Dhillon IS (2008) Coclustering of human cancer microarrays using minimum sum-squared residue coclustering. IEEE/ACM Trans Comput Biol Bioinform 5:385–400

Chung FRK (1997) Spectral graph theory, vol 92. American Mathematical Society.

Deodhar M, Ghosh J (2010) SCOAL: a framework for simultaneous co-clustering and learning from complex data. ACM Trans Knowl Discov Data 4(3):11:1–11:31

Dhillon IS, Guan Y, Kulis B (2004) Kernel k-means: spectral clustering and normalized cuts. In: Proceedings of the tenth ACM SIGKDD international conference on knowledge discovery and data mining, pp 551–556

Dhillon IS, Mallela S, Modha DS (2003) Information-theoretic co-clustering. In: Proceedings of the ninth ACM SIGKDD international conference on knowledge discovery and data mining, pp 89–98

Dhillon IS (2001) Co-clustering documents and words using bipartite spectral graph partitioning. In: Proceedings of the seventh ACM SIGKDD international conference on knowledge discovery and data mining, pp 269–274

Frise E, Hammonds AS, Celniker SE (2010) Systematic image-driven analysis of the spatial Drosophila embryonic expression landscape. Mol Syst Biol 6:345

Giannakidou E, Koutsonikola V, Vakali A, Kompatsiaris Y (2008) Co-clustering tags and social data sources. In: Proceedings of the 2008 the ninth international conference on web-age information management, pp 317–324

Golub GH, van Loan CF (1996) Matrix computations, 3rd edn. Johns Hopkins University Press, Baltimore, MD

Green N, Rege M, Liu X, Bailey R (2011) Evolutionary spectral co-clustering. In: The 2011 international joint conference on neural networks, pp 1074–1081

Hartigan JA (1972) Direct clustering of a data matrix. J Am Stat Assoc 67(337):123–129

Jain AK, Murty MN, Flynn PJ (1999) Data clustering: a review. ACM Comput Surv 31:264–323

Kluger Y, Basri R, Chang JT, Gerstein M (2003) Spectral biclustering of microarray data: coclustering genes and conditions. Genome Res 13(4):703–716

Kumar S, Jayaraman K, Panchanathan S, Gurunathan R, Marti-Subirana A, Newfeld SJ (2002) BEST: a novel computational approach for comparing gene expression patterns from early stages of Drosophila melanogaster develeopment. Genetics 169:2037–2047

Kumar S, Konikoff C, Van Emden B, Busick C, Davis KT, Ji S, Lin-Wei W, Ramos H, Brody T, Panchanathan S, Ye J, Karr TL, Gerold K, McCutchan M, Newfeld SJ (2011) Flyexpress: visual mining of spatiotemporal patterns for genes and publications in drosophila embryogenesis. Bioinformatics 27(23):3319–3320

Lécuyer E, Yoshida H, Parthasarathy N, Alm C, Babak T, Cerovina T, Hughes TR, Tomancak P, Krause HM (2007) Global analysis of mRNA localization reveals a prominent role in organizing cellular architecture and function. Cell 131:174–187

Lécuyer E, Tomancak P (2008) Mapping the gene expression universe. Curr Opin Genet Dev 18(6):506–512

Lee DD, Seung HS (1999) Learning the parts of objects by non-negative matrix factorization. Nature 401:788–791

Leskovec J, Kleinberg J, Faloutsos C (2007) Graph evolution: densification and shrinking diameters. ACM Trans Knowl Discov Data 1(1):2

Li J, Tao D (2013) Simple exponential family PCA. IEEE Trans Neural Netw Learn Syst 24(3):485–497

Lin Y-R, Chi Y, Zhu S, Sundaram H, Tseng BL (2009) Analyzing communities and their evolutions in dynamic social networks. ACM Trans Knowl Discov Data 3:8:1–8:31

Li J, Tao D (2013) A Bayesian factorised covariance model for image analysis. In: Proceedings of the international joint conferences on artificial intelligence

Livne OE, Golub GH (2004) Scaling by binormalization. Numer Algorithms 35:97–120

Long B, Wu X, Zhang ZM, Yu PS (2006) Unsupervised learning on k-partite graphs. In: Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining, pp 317–326

Long B, Zhang ZM, Yu PS (2005) Co-clustering by block value decomposition. In: Proceedings of the eleventh ACM SIGKDD international conference on knowledge discovery in data mining. ACM, pp 635–640

Luxburg U (2007) A tutorial on spectral clustering. Stat Comput 17:395–416

Madeira SC, Oliveira AL (2004) Biclustering algorithms for biological data analysis: a survey. IEEE/ACM Trans Comput Biol Bioinform 1:24–45

Mei Q, Zhai CX (2005) Discovering evolutionary theme patterns from text: an exploration of temporal text mining. In: Proceedings of the eleventh ACM SIGKDD international conference on knowledge discovery and data mining, pp 198–207

Ng AY, Jordan MI, Weiss Y (2001) On spectral clustering: analysis and an algorithm. Adv Neural Inf Process Syst 14:849–856

Saha A, Sindhwani V (2012) Learning evolving and emerging topics in social media: a dynamic NMF approach with temporal regularization. In: Proceedings of the fifth ACM international conference on web search and data mining, pp 693–702

Sandmann T, Girardot C, Brehme M, Tongprasit W, Stolc V, Furlong EEM (2007) A core transcriptional network for early mesoderm development in Drosophila melanogaster. Genes Dev 21(4):436–449

Shewchuk JR (1996) Triangle: engineering a 2D quality mesh generator and delaunay triangulator. In: Lin MC, Manocha D (eds) Applied computational geometry: towards geometric engineering, volume 1148 of lecture notes in computer science. Springer, Berlin, pp 203–222. From the First ACM Workshop on Applied Computational Geometry

Shi J, Malik J (2000) Normalized cuts and image segmentation. IEEE Trans Pattern Anal Mach Intell 22(8):888–905

Stathopoulos A, Levine M (2005) Genomic regulatory networks and animal development. Dev Cell 9(4):449–462

Sun J, Faloutsos C, Papadimitriou S, Yu PS (2007) GraphScope: parameter-free mining of large time-evolving graphs. In: Proceedings of the 13th ACM SIGKDD international conference on knowledge discovery and data mining, pp 687–696

Tao D, Li X, Wu X, Maybank SJ (2007) General tensor discriminant analysis and gabor features for gait recognition. IEEE Trans Pattern Anal Mach Intell 29(10):1700–1715

Tianbing X, Zhang Z, Yu PS, Long B (2012) Generative models for evolutionary clustering. ACM Trans Knowl Discov Data 6(2):7

Tomancak P, Berman B, Beaton A, Weiszmann R, Kwan E, Hartenstein V, Celniker S, Rubin G (2007) Global analysis of patterns of gene expression during Drosophila embryogenesis. Genome Biol 8(7):R145

Tomancak P, Beaton A, Weiszmann R, Kwan E, Shu S, Lewis SE, Richards S, Ashburner M, Hartenstein V, Celniker SE, Rubin GM (2002) Systematic determination of patterns of gene expression during Drosophila embryogenesis. Genome Biol 3(12):0081–0088

Tong H, Papadimitriou S, Philip SY, Faloutsos C (2008) Proximity tracking on time-evolving bipartite graphs. In: Proceedings of the SIAM international conference on data mining, pp 704–715

Volker Hartenstein (1995) Atlas of Drosophila development. Cold Spring Harbor Laboratory Press, New York

Wang F, Li P, König AC (2011a) Efficient document clustering via online nonnegative matrix factorizations. In: Proceedings of the SIAM international conference on data mining. SIAM, pp 908–919

Wang F, Li T, Zhang C (2008) Semi-supervised clustering via matrix factorization. In: Proceedings of the SIAM international conference on data mining. SIAM, pp 1–12

Wang F, Tong H, Lin C-Y (2011b) Towards evolutionary nonnegative matrix factorization. In: Proceedings of the twenty-fifth AAAI conference on artificial intelligence

Yu K, Yu S, Tresp V (2006) Soft clustering on graphs. In: Weiss Y, Schölkopf B, Platt J (eds) Advances in neural information processing systems, vol 18. MIT Press, Cambridge, MA, pp 1553–1560

Zha H, He X, Ding C, Simon H, Gu M (2001) Bipartite graph partitioning and data clustering. In: Proceedings of the tenth international conference on information and knowledge management, pp 25–32

Zhang W, Feng D, Li R, Chernikov A, Chrisochoides N, Osgood C, Konikoff C, Newfeld S, Kumar S, Ji S (2013) A mesh generation and machine learning framework for Drosophila gene expression pattern image analysis. BMC Bioinform 14:372

Zhang W, Ji S, Zhang R (2013) Evolutionary soft co-clustering. In: Proceedings of the 2013 SIAM international conference on data mining. Society for Industrial and Applied Mathematics, Philadelphia, PA, pp 121–129