Title
Robust Multi-Network Clustering Via Joint Cross-Domain Cluster Alignment

Authors
Liu, Rui
Cheng, Wei
Tong, Hanghang
et al.

Publication Date
2015-11-01

DOI
10.1109/icdm.2015.13

Peer reviewed

HHS Public Access

Author manuscript

Proc IEEE Int Conf Data Min. Author manuscript; available in PMC 2016 May 25.

Published in final edited form as:

Proc IEEE Int Conf Data Min. 2015 November ; 2015: 291–300. doi:10.1109/ICDM.2015.13.

Robust Multi-Network Clustering via Joint Cross-Domain Cluster Alignment

Rui Liu[*,¶], **Wei Cheng**[†,¶], **Hanghang Tong**[‡], **Wei Wang**[§], and **Xiang Zhang**[*]

Rui Liu: rui.liu4@case.edu; Wei Cheng: weicheng@cs.unc.edu; Hanghang Tong: htong6@asu.edu; Wei Wang: weiwang@cs.ucla.edu; Xiang Zhang: xiang.zhang@case.edu

[*]Department of Electrical Engineering and Computer Science, Case Western Reserve University, Cleveland, OH 44106

[†]Department of Computer Science, University of North Carolina at Chapel Hill, NC 27599

[‡]School of Computing, Informatics, Decision Systems Engineering, Arizona State University, Tempe, AZ 85281

[§]Department of Computer Science, University of California, Los Angeles, CA 90095

Abstract

Network clustering is an important problem that has recently drawn a lot of attentions. Most existing work focuses on clustering nodes within a single network. In many applications, however, there exist *multiple related* networks, in which each network may be constructed from a different domain and instances in one domain may be related to instances in other domains. In this paper, we propose a robust algorithm, MCA, for multi-network clustering that takes into account cross-domain relationships between instances. MCA has several advantages over the existing single network clustering methods. First, it is able to detect associations between clusters from different domains, which, however, is not addressed by any existing methods. Second, it achieves more consistent clustering results on multiple networks by leveraging the *duality* between clustering individual networks and inferring cross-network cluster alignment. Finally, it provides a multi-network clustering solution that is more robust to noise and errors. We perform extensive experiments on a variety of real and synthetic networks to demonstrate the effectiveness and efficiency of MCA.

I. Introduction

Networks (or graphs) are widely used in representing relationships between instances, in which each node corresponds to an instance and each edge depicts the relationship between a pair of instances. Network clustering (or graph clustering) [1]–[3] has become an effective means in discovering modules formed by closely related instances in such networks, which may in turn reveal functional structure of the networks. Recently, the attention has moved from clustering in a single homogeneous network (built on instances from one domain) to joint clustering on multiple heterogeneous networks (from different but related domains), due to obvious reasons: integrating information from different but related domains not only

[¶]These authors contributed equally to this work

may help to resolve ambiguity and inconsistency in clustering outcome, but also may discover and leverage strong associations between clusters from different domains. Consequently, these multi-view network clustering methods [3], [4] are able to substantially improve the clustering accuracy. For example, millions of genetic variants on human genome have been reported to be disease related, most of which are in the form of single nucleotide polymorphism (SNP). These SNPs do not function independently. Instead, a set of SNPs may play joint roles in a disease. Such interactions between SNPs can be modeled by a SNP interaction network. Fig. 1 shows an exemplar SNP interaction network $\mathscr{G}_1$ of 17 SNPs on the left, in which nodes are SNPs and weighted edges represent interactions between SNPs. Even though the underlying biological processes are complex and only partially solved, it is well established that SNPs may alter the expression levels of related genes which may in turn have a cascading effect to other genes, e.g., in the same biological pathways [5]. The interactions between genes can be measured by correlations of gene expressions and represented by a gene interaction network. Fig. 1 shows an exemplar gene interaction network of 20 genes on the right, in which nodes are genes and weighted edges represent interactions between genes. These two networks are heavily related because of the (complicated) relationships between SNPs and genes, as demonstrated in many expression quantitative trait loci (eQTL) studies. These cross-domain relationships are represented by dotted edges between SNPs and genes in Fig. 1. The strength of such relationship is coded by the edge weight. It is evident that a joint analysis becomes essential in these related domains.

Despite the success of previous approaches in network clustering, they still suffer from two common limitations. First, existing methods usually assume that information collected in different domains are for the same set of instances. Thus, the cross-domain instance relationships are strictly *one-to-one* correspondence. This assumption may not hold in many applications. More often than not, data instances (e.g., SNPs) in one domain may be related to multiple instances (e.g., genes) in another domain. Methods that can account for many-to-many cross-domain relationships are in need [6]. Second, existing approaches tend to focus on network clustering and ignore any associations that may be exhibited between clusters from different domains. However, “alignment” between clusters from multiple domains may provide a more comprehensive depiction of the whole system. For example, a cluster of SNPs may jointly regulate the expressions of a cluster of genes, which may be revealed by cluster level associations. Fig. 1 shows 2 SNP clusters: A (including SNPs {1, 2, 3, 4}) and B (including SNPs {12, 13, 14, 16}), and 3 gene clusters: C (including genes {*a, b, c, d*}), D (including genes {*p, q, r, s*}) and E (including genes {*i, j, k, m*}). As summarized in Table I, SNP cluster A is strongly associated with gene cluster C and SNP cluster B is strongly associated with gene cluster D. Gene cluster E is not strongly associated with any SNP cluster. Although we are given cross-domain associations at the instance level, it is still nontrivial to discover cross-domain associations at the cluster level, especially in the presence of noise. Our goal is to discover such strong associations between pairs of clusters from different domains simultaneously when we perform network clustering.

In this paper, we propose a robust approach, MCA (Multi-network Clustering via cluster Alignment), to detect network clusters in multiple domains and their cross-domain

associations. In addition to the advantages discussed above, the *duality* between clustering in individual networks and inferring cross-network cluster alignment enables mutual reinforcement when both tasks are performed simultaneously. As a result, MCA can effectively filter noise and resolve ambiguities in individual networks, and achieve much higher accuracy in detecting network clusters and their cross-domain associations. It also employs a sparsity regularizer on the cluster alignment to provide additional robustness to noise in the prior cross-domain (instance-level) relationships.

Our contributions are summarized as follows.

- To the best of our knowledge, little prior work has studied the problem of cross-domain cluster association detection. In this paper, we propose and investigate this novel problem under the multi-domain setting. The problem is essential to a wide range of applications.
- We develop a framework, MCA, based on nonnegative matrix tri-factorization to simultaneously cluster instances within each domain and reveal the associations between clusters from different domains. Clustering and cluster association discovery could mutually enhance each other. We provide rigorous theoretical analysis of MCA in terms of its correctness, convergence and complexity.
- We evaluate MCA by extensive experiments on both synthetic and real datasets. The experimental results demonstrate that MCA is superior to existing approaches in both clustering accuracy and cluster association accuracy.

II. Related Work

To the best of our knowledge, this is the first work to “align” clusters across multiple domains. Existing work on network clustering primarily focused on clustering in a single network [7], [9]. In [9], the authors pioneered a graph partitioning algorithm using normalized cut. Spectral clustering has gained popularity as an efficient clustering algorithms in recent years [11]. It is simple to implement since its computation burden is primarily on the computation of eigenvectors which has been studied in-depth in numerical analysis. In [7], the authors proposed a framework to detect communities in a network based on modularity. Some other multi-domain graph (network) clustering methods [3] focus on improving the clustering accuracy within each domain utilizing information from other domains. The cross-domain instance relationships are only used to enhance the clustering result within each domain. They do not capture associations between clusters across multiple domains. Multi-domain data are inherently heterogenous. Networks constructed from multiple related domains can be transformed into a heterogenous information network, on which clustering may be performed [12]. However, this pioneer work focused on ranking-based clustering which ranks clusters on a pre-specified target type (domain). This is different from our goal of performing clustering within and cross domains. In addition, some methods on co-clustering also make use of Nonnegative Matrix Tri-Factorization (NMTF) and graph regularizer. Co-clustering [13], DNMTF [14] and RCC [15] were originally designed to improve the clustering accuracy on documents by clustering rows and columns of a term-document matrix simultaneously [16]. They usually pay special attentions to the

duality of clustering of terms and clustering of documents. The cross-domain cluster associations are not explicitly considered by co-clustering methods, even though some of these methods may be adapted to derive information about cross-domain cluster associations. They are either incapable of handling multi-network setting or sensitive to noise, since they were not designed for network clustering.

III. Multi-Network Clustering via Cross-Domain Cluster Alignment

In this section, we discuss the problem definition and our proposed algorithm MCA.

A. Problem Definition

Suppose that we have N domains $\{\mathscr{D}_1, \mathscr{D}_2, \ldots, \mathscr{D}_N\}$. Instances and their relationships within each domain are represented by a network $\mathscr{G}_p$, ($1 \le p \le N$). Let $\mathbf{A}_p$ be affinity/adjacency matrix $\mathscr{G}_p$. It is possible that some instances in domain $\mathscr{D}_p$ $(1 \le p \le N)$ may be related to some instances in domain $\mathscr{D}_q$ $(1 \le q \le N, p \ne q)$. These cross-domain relationships between instances can be represented by a matrix $\mathbf{W}_{pq}$. Important notations are listed in Table II. More often than not, the matrix $\mathbf{W}_{pq}$ is derived from prior knowledge and may be incomplete and noisy. Our goal is to integrate these cross-network instance relationships into the task of multi-network clustering and infer cross-network associations between clusters. We formulate this problem as an optimization problem that generates clustering and cluster associations simultaneously. We now discuss them in detail.

For simplicity, we begin with 2 domains. For clustering in domain $\mathscr{D}_1$, We want to minimize the following objective function

$$\sum_{ij} \mathbf{A}_{1_{ij}} \|\mathbf{H}_{1_{i\cdot}} - \mathbf{H}_{1_{j\cdot}}\|_2^2, \quad s.t.\, \mathbf{H}_1^T \mathbf{H}_1 = \mathbf{I}, \mathbf{H}_1 \ge 0$$

where $\mathbf{H}_1$ is the cluster assignment matrix in domain $\mathscr{D}_1$ and $\mathbf{H}_{1_{i\cdot}}$ means the i-th row of matrix $\mathbf{H}_1$. $\mathbf{H}_{1_{i,k}}$ can be viewed as the probability that the i-th instance in domain $\mathscr{D}_1$ belongs to the k-th cluster of this domain. A similar objective function can be applied to clustering in domain $\mathscr{D}_2$.

In order to capture the cross-network cluster associations, we adopt the co-clustering strategy that minimizes the following objective function

$$\|\mathbf{W}_{12} - \mathbf{H}_1 \mathbf{S}_{12} \mathbf{H}_2^T\|_F^2 + \eta_1 \|\mathbf{S}_{12}\|_1,$$
$$s.t.\, \mathbf{H}_1^T \mathbf{H}_1 = \mathbf{I}, \mathbf{H}_2^T \mathbf{H}_2 = \mathbf{I}, \mathbf{H}_1 \ge 0, \mathbf{H}_2 \ge 0$$

This objective function is the Sparse Nonnegative Matrix Tri-Factorization. With orthogonality constraints on $\mathbf{H}_1$ and $\mathbf{H}_2$, it is equivalent to running K-means co-clustering on $\mathbf{W}_{12}$ [13]. $\mathbf{S}_{12}$ is the *cross-domain alignment matrix*, depicting the alignment of the clusters from the two domains. Because $\mathbf{W}_{12}$ may contain noise, we employ the ℓ_1-norm on $\mathbf{S}_{12}$ to suppress the influence of any inconsistencies in $\mathbf{W}_{12}$.

Combining together the above two parts, we have the following optimization problem

$$\begin{aligned} min\ &\|\mathbf{W}_{12}-\mathbf{H}_1\mathbf{S}_{12}\mathbf{H}_2^T\|_F^2+\alpha_1\sum_{ij}\mathbf{A}_{1_{ij}}\|\mathbf{H}_{1_{i\cdot}}-\mathbf{H}_{1_{j\cdot}}\|_2^2 \\ &+\alpha_2\sum_{ij}\mathrm{A}_{2_{ij}}\|\mathbf{H}_{2_{i\cdot}}-\mathbf{H}_{2_{j\cdot}}\|_2^2+\eta_1\|\mathbf{S}\|_1 \\ s.t.\ &\mathbf{H}_1^T\mathbf{H}_1=\mathbf{I},\mathbf{H}_2^T\mathbf{H}_2=\mathbf{I},\mathbf{H}_1\geq\mathbf{0},\mathbf{H}_2\geq\mathbf{0} \end{aligned} \tag{1}$$

where a_1, a_2 and η_1 are parameters that balance different between terms, whose values can be determined via cross validation.

By simplifying it, we obtain

$$\begin{aligned} \min_{\mathbf{H}_1\geq\mathbf{0},\mathbf{H}_2\geq\mathbf{0},\mathbf{S}\geq\mathbf{0}}&\|\mathbf{W}_{12}-\mathbf{H}_1\mathbf{S}_{12}\mathbf{H}_2^T\|_F^2+\alpha_1\mathrm{Tr}(\mathbf{H}_1^T\mathbf{\Theta}_1\mathbf{H}_1) \\ &+\alpha_2\mathrm{Tr}(\mathbf{H}_2^T\mathbf{\Theta}_2\mathbf{H}_2)+\eta_1\|\mathbf{S}\|_1 \\ &s.t.\ \mathbf{H}_1^T\mathbf{H}_1=\mathbf{I},\mathbf{H}_2^T\mathbf{H}_2=\mathbf{I} \end{aligned} \tag{2}$$

where $\mathbf{\Theta}_1$ and $\mathbf{\Theta}_2$ are the Laplacian Matrices of $\mathscr{G}_1$ and $\mathscr{G}_2$, respectively.

Now we come back to the multi-domain case. Let $\mathbf{W}_{pq}$ be the matrix defining instance level relationships between domain $\mathscr{D}_p$ and domain $\mathscr{D}_q$. Here we assume $\mathbf{W}_{pq}=\mathbf{W}_{qp}^T$. Similarly, we use $\mathbf{S}_{pq}$ to represent the cross-domain cluster alignment matrix between domain $\mathscr{D}_p$ and domain $\mathscr{D}_q$ and we have $\mathbf{S}_{pq}=\mathbf{S}_{qp}^T$. The optimization problem in Eq. (2) can be naturally extended to the following multi-domain case.

$$\begin{aligned} \min_{\mathbf{H}_p\geq\mathbf{0},\mathbf{S}_{pq}\geq\mathbf{0},\forall p\neq q}&\sum_{p\neq q}\|\mathbf{W}_{pq}-\mathbf{H}_p\mathbf{S}_{pq}\mathbf{H}_q^T\|_F^2+ \\ &\sum_p\alpha_p\mathrm{Tr}\left(\mathbf{H}_p^T\mathbf{\Theta}_p\mathbf{H}_p\right)+ \\ &\sum_{p\neq q}\eta_{pq}\|\mathbf{S}_{pq}\|_1,\ \ s.t.\ \forall p,\ \ \mathbf{H}_p^T\mathbf{H}_p=\mathbf{I} \end{aligned} \tag{3}$$

B. Learning Algorithm

In this section, we present the learning algorithm, MCA, to solve the optimization problem in Eq. (3). Since the objective function is not jointly convex with respect to all variables, we adopt an alternating optimization scheme. Specifically, each time we optimize the objective with respective to one variable while fixing others. The following two theorems set the foundation for our algorithm. Their correctness and convergence are guaranteed, which will be proven later.

Theorem 1—While fixing other variables, the objective function in Eq. (3) will monotonically decrease every time we update $\mathbf{S}_{pq}$ according to Eq. (4) until convergence.

$$\mathbf{S}_{pq}\leftarrow\mathbf{S}_{pq}\circ\sqrt{\frac{\left(\mathbf{H}_p^T\mathbf{W}_{pq}\mathbf{H}_q\right)}{\left(\mathbf{H}_p^T\mathbf{H}_p\mathbf{S}_{pq}\mathbf{H}_q^T\mathbf{H}_q\right)+\frac{1}{2}\eta_{pq}}} \tag{4}$$

Theorem 2—While fixing other variables, the objective function in Eq. (3) will monotonically decrease every time we update $\mathbf{H}_p$ according to Eq. (5) until convergence.

$$\mathbf{H}_p \leftarrow \mathbf{H}_p \circ \sqrt{\frac{\sum_{q\neq p}\left(\mathbf{W}_{pq}\mathbf{H}_q\mathbf{S}_{pq}^T\right)+\alpha_p\left(\mathbf{\Theta}_p^-\mathbf{H}_p\right)}{\mathbf{H}_p\mathbf{H}_p^T\left(\sum_{q\neq p}\left(\mathbf{W}_{pq}\mathbf{H}_q\mathbf{S}_{pq}^T\right)+\alpha_p\left(\mathbf{\Theta}_p^-\mathbf{H}_p\right)\right)}} \quad (5)$$

where $\mathbf{\Theta}_p^-$ is the negative part of $\mathbf{\Theta}_p$, i.e., $\mathbf{\Theta}_{p_{ij}}^-=(|\mathbf{\Theta}_{p_{ij}}|-\mathbf{\Theta}_{p_{ij}})/2$.

Note that $\circ$, $\frac{(\cdot)}{(\cdot)}$ and $\sqrt{(\cdot)}$ are element-wise multiplication, division and square root, respectively. Based on Theorems 1 and 2, we develop an iterative updating algorithm summarized in Algorithm 1.

C. Correctness Analysis

In this section, we give the correctness analysis of the updating rules in Theorem 1, according to the Karush-Kuhn-Tucker (KKT) condition. The proof of updating rules for Theorem 2 is similar and hence omitted here.

Define the Lagrangian function with respect to $\mathbf{S}_{pq}$ as

$$L(\mathbf{S}_{pq})=-2\,\mathrm{Tr}(\mathbf{H}_p^T\mathbf{W}_{pq}\mathbf{H}_q\mathbf{S}_{pq}^T)+\mathrm{Tr}(\mathbf{H}_p\mathbf{S}_{pq}\mathbf{H}_q^T\mathbf{H}_q\mathbf{S}_{pq}^T\mathbf{H}_p^T)+\eta_{pq}\mathbf{S}_{pq}-\mathrm{Tr}(\mathbf{\Lambda}_{pq}\mathbf{S}_{pq}^T) \quad (6)$$

where $\mathbf{\Lambda}_{pq}$ is a symmetric matrix whose entries are the lagrangian multipliers. Note that $\|\mathbf{S}_{pq}\|_1$ becomes $\mathbf{S}_{pq}$ in Eq. (6) because $\mathbf{S}_{pq}\geq 0$. We also omit all constant terms with respect to $\mathbf{S}_{pq}$ in Eq.(6). The same tricks are used in the following analysis.

Algorithm 1: The MCA Algorithm

input : $\mathbf{W}_{pq}$, $\mathbf{A}_p$ $(1\leq p,q\leq N, p\neq q)$
output: $\mathbf{H}_p$, $\mathbf{S}_{pq}$ $(1\leq p,q\leq N, p\neq q)$

begin
  Random initialization of $\mathbf{H}_p$ *and* $\mathbf{S}_{pq}$ $(1\leq p,q\leq N, p\neq q)$;
  $\mathbf{\Theta}_p=\mathbf{D}_{A_p}-\mathbf{A}_p, (1\leq p\leq N)$ *where* $\mathbf{D}_{A_p}$ *is the degree matrix of* $\mathbf{A}_p$;
  while *no convergence* **do**
    for $p=1$ **to** N **do**
      for $q=1$ **to** N **do**
        if $p\neq q$ **then**
          update $\mathbf{S}_{pq}\leftarrow$ $\mathbf{S}_{pq}\circ\sqrt{\frac{(\mathbf{H}_p^T\mathbf{W}_{pq}\mathbf{H}_q)}{(\mathbf{H}_p^T\mathbf{H}_p\mathbf{S}_{pq}\mathbf{H}_q^T\mathbf{H}_q)+\frac{1}{2}\eta_{pq}}}$;
    for $p=1$ **to** N **do**
      update $\mathbf{H}_p\leftarrow\mathbf{H}_p\circ$ $\sqrt{\frac{\sum_{q\neq p}(\mathbf{W}_{pq}\mathbf{H}_q\mathbf{S}_{pq}^T)+\alpha_p(\mathbf{\Theta}_p^-\mathbf{H}_p)}{(\mathbf{H}_p\mathbf{H}_p^T(\sum_{q\neq p}(\mathbf{W}_{pq}\mathbf{H}_q\mathbf{S}_{pq}^T)+\alpha_p(\mathbf{\Theta}_p^-\mathbf{H}_p)))}}$;

The partial derivative with respect to $\mathbf{S}_{ij}$ is

$$\frac{\partial L(\mathbf{S}_{pq})}{\partial \mathbf{S}_{pq}} = -2(\mathbf{H}_p^T \mathbf{W}_{pq} \mathbf{H}_q) + 2(\mathbf{H}_p^T \mathbf{H}_p \mathbf{S}_{pq} \mathbf{H}_q^T \mathbf{H}_q) + \eta_{pq} - \mathbf{\Lambda}_{pq} \quad (7)$$

From the optimality condition $\frac{\partial L(\mathbf{S}_{pq})}{\partial \mathbf{S}_{pq}} = 0$,

$$\mathbf{\Lambda}_{pq} = -2\left(\mathbf{H}_p^T \mathbf{W}_{pq} \mathbf{H}_q\right)_{ij} + 2\left(\mathbf{H}_p^T \mathbf{H}_p \mathbf{S}_{pq} \mathbf{H}_q^T \mathbf{H}_q\right) + \eta_{pq} \quad (8)$$

The KKT complementarity condition for the nonnegativity of $\mathbf{S}_{ij}$ is

$$\mathbf{\Lambda}_{pq} \mathbf{S}_{pq} = 0 \quad (9)$$

Combining with Eq. (8), the KKT complementarity condition becomes

$$\left(-2(\mathbf{H}_p^T \mathbf{W}_{pq} \mathbf{H}_q)_{ij} + 2(\mathbf{H}_p^T \mathbf{H}_p \mathbf{S}_{pq} \mathbf{H}_q^T \mathbf{H}_q) + \eta_{pq}\right) \circ \mathbf{S}_{pq} = 0 \quad (10)$$

According to Eq. (4), the update rule for **S** is

$$\mathbf{S}_{pq} \leftarrow \mathbf{S}_{pq} \circ \sqrt{\frac{\left(\mathbf{H}_p^T \mathbf{W}_{pq} \mathbf{H}_q\right)}{\left(\mathbf{H}_p^T \mathbf{H}_p \mathbf{S}_{pq} \mathbf{H}_q^T \mathbf{H}_q\right) + \frac{1}{2}\eta_{pq}}}$$

At convergence, $\mathbf{S}_{pq}$ at the left-hand side and right-hand side should be equal. Then, via simple derivation, we can verify that the update rule for $\mathbf{S}_{pq}$ in Eq. (4) satisfies the KKT complementarity condition in Eq. (10).

D. Convergence Analysis

In this subsection, we prove the guarantee of convergence using an auxiliary function [17].

Definition 1—[17] $Z(h, h')$ is an auxiliary function for $f(h)$ if the conditions

$$Z(h, h') \geq f(h), Z(h, h) = f(h) \quad (11)$$

are satisfied for any given h, h'.

Lemma 1—If Z is an auxiliary function for f, then f is non-increasing under the update [17]

$$h^{(t+1)} = \arg\min_h Z(h, h^{(t)}) \quad (12)$$

Proof—$f(h^{(t+1)}) \leq Z(h^{(t+1)}, h^{(t)}) \leq Z(h^{(t)}, h^{(t)}) = f(h^{(t)})$. ■

The auxiliary function with respect to **S** is

$$Z(S_{pq}, S'_{pq}) = -2\sum_{i,j}(\mathbf{H}_p^T\mathbf{W}\mathbf{H}_q)_{ij}S'_{pq_{ij}}(1+log\frac{\mathbf{S}_{pq_{ij}}}{S'_{pq_{ij}}}) + \sum_{i,j}\frac{(\mathbf{H}_p^T\mathbf{H}_pS'_{pq}\mathbf{H}_q^T\mathbf{H}_q)_{ij}\mathbf{S}^2_{pq_{ij}}}{S'_{pq_{ij}}} + \eta_{pq}\sum_{i,j}\frac{\mathbf{S}^2_{pq_{ij}}+\mathbf{S}'^2_{pq_{ij}}}{2\mathbf{S}'^2_{pq_{ij}}} \quad (13)$$

$$\frac{\partial Z}{\partial \mathbf{S}_{pq_{ij}}} = -2\left(\mathbf{H}_p^T\mathbf{W}_{pq}\mathbf{H}_q\right)_{ij}\frac{\mathbf{S}'_{pq_{ij}}}{\mathbf{S}_{pq_{ij}}} + 2\frac{\left(\mathbf{H}_p^T\mathbf{H}_p\mathbf{S}'_{pq}\mathbf{H}_q^T\mathbf{H}_q\right)_{ij}}{\mathbf{S}'_{pq_{ij}}}\mathbf{S}_{pq_{ij}} + \eta_{pq}\frac{\mathbf{S}_{pq_{ij}}}{\mathbf{S}'_{pq_{ij}}} \quad (14)$$

Letting $\frac{\partial Z}{\partial \mathbf{S}_{pq_{ij}}}=0$, we obtain

$$\mathbf{S}_{pq_{ij}} = \mathbf{S}'_{pq_{ij}}\sqrt{\frac{\left(\mathbf{H}_p^T\mathbf{W}_{pq}\mathbf{H}_q\right)ij}{\left(\mathbf{H}_p^T\mathbf{H}_p\mathbf{S}'_{pq}\mathbf{H}_q^T\mathbf{H}_q\right)_{ij}+\frac{1}{2}\eta_{pq}}} \quad (15)$$

Similarly, the auxiliary function with respect to $\mathbf{H}_p$ is

$$Z(\mathbf{H}_p, \mathbf{H}'_p) = -2\sum_{q\neq p}\sum_{i,j}(\mathbf{W}_{pq}\mathbf{H}_q\mathbf{S}_{pq}^T)_{ij}\mathbf{H}'_{p_{ij}}(1+log\frac{\mathbf{H}_{p_{ij}}}{\mathbf{H}'_{p_{ij}}}) + \sum_{q\neq p}\sum_{i,j}\frac{(\mathbf{H}'_p\mathbf{S}_{pq}\mathbf{H}_q^T\mathbf{H}_q\mathbf{S}_{pq}^T)_{ij}\mathbf{H}^2_{p_{ij}}}{\mathbf{H}'_{p_{ij}}} + \alpha_p\sum_{i,j}\frac{(\mathbf{\Theta}_p^+\mathbf{H}'_p)_{ij}\mathbf{H}^2_{p_{ij}}}{\mathbf{H}'_{p_{ij}}} - \alpha_p\sum_{i,j,k}\mathbf{\Theta}^-_{p_{ki}}\mathbf{H}'_{p_{kj}}\mathbf{H}'_{p_{ij}}(1+log\frac{\mathbf{H}_{p_{kj}}\mathbf{H}_{p_{ij}}}{\mathbf{H}'_{p_{kj}}\mathbf{H}'_{p_{ij}}}) + \sum_{i,j}\frac{(\mathbf{H}'_p\mathbf{\Lambda}_p)_{ij}\mathbf{H}^2_{p_{ij}}}{\mathbf{H}'_{p_{ij}}} - \mathrm{Tr}(\mathbf{\Lambda}_p) \quad (16)$$

$$\frac{\partial Z}{\partial \mathbf{H}_{p_{ij}}} = -2\sum_{q\neq p}\left(\mathbf{W}_{pq}\mathbf{H}_q\mathbf{S}_{pq}^T\right)_{ij}\frac{\mathbf{H}'_{p_{ij}}}{\mathbf{H}_{p_{ij}}} + 2\sum_{q\neq p}\frac{\left(\mathbf{H}'_p\mathbf{S}_{pq}\mathbf{H}_q^T\mathbf{H}_q\mathbf{S}_{pq}^T\right)_{ij}\mathbf{H}_{p_{ij}}}{\mathbf{H}'_{p_{ij}}} + 2\alpha_p\frac{\left(\mathbf{\Theta}_p^+\mathbf{H}'_p\right)_{ij}\mathbf{H}_{p_{ij}}}{\mathbf{H}'_{p_{ij}}} - 2\alpha_p\frac{\left(\mathbf{\Theta}_p^-\mathbf{H}'_p\right)_{ij}\mathbf{H}'_{p_{ij}}}{\mathbf{H}_{p_{ij}}} + 2\frac{\left(\mathbf{H}'_p\mathbf{\Lambda}_p\right)_{ij}\mathbf{H}_{p_{ij}}}{\mathbf{H}'_{p_{ij}}} \quad (17)$$

Letting $\frac{\partial Z}{\partial \mathbf{H}_{p_{ij}}}=0$, we obtain

$$\mathbf{H}_{p_{ij}} = \mathbf{H}'_{p_{ij}} \sqrt{\frac{\sum_{q\neq p}\left(\mathbf{W}_{pq}\mathbf{H}_q\mathbf{S}_{pq}^T\right)_{ij} + \alpha_p\left(\mathbf{\Theta}_p^- \mathbf{H}'_p\right)_{ij}}{\sum_{q\neq p}\left(\mathbf{H}'_p\mathbf{S}_{pq}\mathbf{H}_q^T\mathbf{H}_q\mathbf{S}_{pq}^T\right)_{ij} + \alpha_p\left(\mathbf{\Theta}_p^+ \mathbf{H}'_p\right)_{ij} + \left(\mathbf{H}'_p\mathbf{\Lambda}_p\right)_{ij}}} \quad (18)$$

In order to determine $\mathbf{\Lambda}_p$, we have

$$\frac{\partial L}{\partial \mathbf{H}_p} = -2\sum_{q\neq p}\mathbf{W}_{pq}\mathbf{H}_q\mathbf{S}_{pq}^T + 2\sum_{q\neq p}\mathbf{H}_p\mathbf{S}_{pq}\mathbf{H}_q^T\mathbf{H}_q\mathbf{S}_{pq}^T + 2\alpha_p\mathbf{\Theta}_p\mathbf{H}_p + 2\mathbf{H}_p\mathbf{\Lambda}_p \quad (19)$$

Letting $\frac{\partial L}{\partial \mathbf{H}_p} = 0$, we can solve $\mathbf{\Lambda}_p$.

After submitting $\mathbf{\Lambda}_p$, we obtain

$$\mathbf{H}_{p_{ij}} = \mathbf{H}'_{p_{ij}} \sqrt{\frac{\sum_{q\neq p}\left(\mathbf{W}_{pq}\mathbf{H}_q\mathbf{S}_{pq}^T\right)_{ij} + \alpha_p\left(\mathbf{\Theta}_p^- \mathbf{H}'_p\right)_{ij}}{\left(\mathbf{H}'_p\mathbf{H}_p'^T\left(\sum_{q\neq p}\mathbf{W}_{pq}\mathbf{H}_q\mathbf{S}_{pq}^T + \alpha_p\mathbf{\Theta}_p^-\mathbf{H}'_p\right)\right)_{ij}}} \quad (20)$$

E. Complexity Analysis

With proper order of multiplication, updating **S** and **H** once require $\mathcal{O}(\tilde{n}^2\tilde{k} + \tilde{n}\tilde{k}^2 + \tilde{k}^3)$ and $\mathcal{O}(N\tilde{n}\tilde{k}^2 + \tilde{n}^2\tilde{k})$, where $\tilde{n} = \max_p\{n_p\}$ is the largest number of instances among all domains and $k = \max_p\{k_p\}$ is the largest number of clusters among all domains. If the number of iterations is *Iter*, the overall time complexity is $O(Iter(\tilde{n}\tilde{k}^2 + \tilde{n}^2\tilde{k} + \tilde{k}^3 + N\tilde{n}\tilde{k}^2))$. In practice, the number of instances is much larger than the number of clusters in a domain, which leads to $\tilde{n} \gg \tilde{k}$. In this scenario, the overall time complexity of MCA can be simplified to $\mathcal{O}(Iter(\tilde{n}^2\tilde{k} + N\tilde{n}\tilde{k}^2))$.

IV. Experimental Results

In this section, we evaluate the performance of MCA on both synthetic and real datasets. To the best of our knowledge, there is no previous method that was specifically designed to discover cluster associations. Some co-clustering methods might be adapted to infer cluster associations. We compare our method MCA with three well-known co-clustering methods– Nonnegative Matrix Tri-Factorization proposed by Chris Ding (denoted as NMTF_Chris in our paper) [13], Graph Dual Regularization Non-negative Matrix Tri-Factorization (DNMTF) [14] and Robust Co-Clustering (RCC) [15]. The parameters of each algorithm are tuned using a 5-fold cross validation. Other co-clustering methods are not compared because they are either not suitable for network clustering or do not impose nonnegativity on association matrix **S**. The nonnegativity constraint on **S** is essential to ensure the result interpretability in this problem setting.

A. Evaluation Metrics

We evaluate our results in two perspectives: clustering accuracy within each domain and cluster association accuracy across domains.

1) Clustering Accuracy—We use the widely used normalized mutual information (MI) metric to evaluate the clustering accuracy in each domain. For any domain $\mathscr{D}$, assume that $\mathscr{C}=\left\{c_i, i=1,2,\cdots,\hat{k}\right\}$ is the clustering result where c_i is the i-th cluster. Let $\mathscr{T}=\{t_i, i=1,2,\cdots k\}$ be the ground truth where t_i is the i-th cluster. The normalized MI is defined as

$$\hat{MI}(\mathscr{C},\mathscr{T})=\frac{MI(\mathscr{C},\mathscr{T})}{max(H(\mathscr{C}),H(\mathscr{T}))} \quad (21)$$

where $H(\mathscr{C})$ and $H(\mathscr{T})$ are the entropies for clustering $\mathscr{C}$ and $\mathscr{T}$ and $MI(\mathscr{C},\mathscr{T})$ is the mutual information between $\mathscr{C}$ And $\mathscr{T}$.

$$MI(\mathscr{C},\mathscr{T})=\sum_{c_i\in\mathscr{C},t_j\in\mathscr{T}} p(c_i,t_j)\ln\frac{p(c_i,t_j)}{p(c_i)p(t_j)} \quad (22)$$

where $p(c_i)$ is the percentage of instances contained in c_i and $p(c_i, t_j)$ is the percentage of instances contained in the intersection of c_i and t_j.

2) Cluster Association Accuracy—To evaluate the cross-network cluster association accuracy, we propose a new metric, Clustering Association Metric (CAM). For simplicity, we consider the case where there are only two domains. Assume that we discover $\hat{h}$ pairs of cluster associations $\{c_j, c_j^{'}\}, 1\le j\le\hat{h}$, where c_j is a cluster in domain $\mathscr{D}_1$ and $c_j^{'}$ is its corresponding cluster in domain $\mathscr{D}_2$. Also assume that the ground-truth contains h pairs of cluster associations $\{t_i, t_i^{'}\}, 1\le i\le h$, where t_i is a cluster in domain $\mathscr{D}_1$ and $t_j^{'}$ is its corresponding cluster in domain $\mathscr{D}_2$. Then the CAM is defined by the following equation.

$$\mathrm{CAM}=\frac{1}{h}\sum_{i=1}^{h}\max_{1\le j\le\hat{h}}\frac{|(t_i\cup t_i^{'})\cap(c_j\cup c_j^{'})|}{|(t_i\cup t_i^{'})\cup(c_j\cup c_j^{'})|} \quad (23)$$

where $\cup$ is the set union and $\cap$ is the set intersection. Here, in order to measure the similarity between an inferred cluster association $\{c_j, c_j^{'}\}$ and the ground-truth $\{t_i, t_i^{'}\}$, we use $\frac{|(t_i\cup t_i^{'})\cap(c_j\cup c_j^{'})|}{|(t_i\cup t_i^{'})\cup(c_j\cup c_j^{'})|}$ to evaluate the degree of overlap between $t_i\cup t_i^{'}$ and $c_j\cup c_j^{'}$. For each ground-truth association pair $\{t_i, t_i^{'}\}$, we get the maximal value of $\frac{|(t_i\cup t_i^{'})\cap(c_j\cup c_j^{'})|}{|(t_i\cup t_i^{'})\cup(c_j\cup c_j^{'})|}, \forall j=1,\cdots,\hat{h}$. The CAM is the average of the maximal values for all ground-truth pairs.

B. Simulation Study

We constructed a simulation study on the two synthetic networks $\mathcal{G}_1$, $\mathcal{G}_2$ in Fig. 1. We compare our method MCA with DNMTF, NMTF_Chris and RCC with respect to robustness to varying levels of noise in the cross-domain instance-level relationship matrix $\mathbf{W}_{12}$. Fig. 2 shows that MCA achieves much higher clustering accuracy than all existing methods at all noise levels. Fig. 3 demonstrates the clear advantage of MCA over existing methods in capturing cross-domain cluster associations.

C. DBLP Dataset

We also evaluate our method MCA using a labeled DBLP dataset [18], [19]. The dataset consists of papers and authors from 4 research areas: Database (DB), Artificial Intelligence (AI), Data Mining (DM) and Information Retrieval (IR). It contains 20 conferences and 4057 authors. These conferences are listed by area in Table III and the author distribution by area is shown in Table IV. We use $\mathcal{D}_1$ to denote the author domain and $\mathcal{D}_2$ to denote the conference domain. In $\mathcal{D}_1$, the network $\mathcal{G}_1$ represents the co-authorship. Each entry $A_{1_{ij}}$ in the affinity matrix A_1 of $\mathcal{G}_1$ is the number of papers coauthored by the i-th and j-th authors. The affinity matrix A_2 of $\mathcal{G}_2$ represents the similarities between the topics covered by two conferences. To compute it, we first construct the term-conference matrix $\mathbf{F}$, in which each entry $\mathbf{F}_{ij}$ is the number of occurrences of the i-th term in the titles of papers published in the j-th conference. Thus each column $\mathbf{F}_j$ of the matrix can be viewed as a feature vector describing the j-th conference. The similarity score of two conferences j and j' can be computed as $\frac{\mathbf{F}_j \cdot \mathbf{F}_{j'}}{\|\mathbf{F}_j\|\|\mathbf{F}_{j'}\|}$. $\mathbf{W}_{12}$ represents the relationships between authors and conferences, in which each entry denotes the number of papers that an author published in a given conference. A snapshot of the DBLP network used in our experiment can be seen in Figure 6.

To compare the robustness of different methods, we introduce noise by randomly shuffling a certain percentage of the entries in $\mathbf{W}_{12}$. Fig. 4 shows that noise in the prior knowledge on cross-domain relationships does not affect the clustering accuracy of MCA in the conference domain at all, and only lowers the accuracy of MCA in the author domain modestly when $\mathbf{W}_{12}$ is dominated by noise. Fig. 5 shows that the accuracy of the inferred cross domain cluster associations also only drops modestly for MCA when the noise level is very high. In contrast, we observe that all other methods are far more sensitive to noise, among which NMTF_Chris performs noticeably better than ONMTF and RCC.

To better understand how these methods perform, we list the top 4 associations between conference clusters and author clusters returned by MCA and NMTF_Chris in Table V when the noise level is set to 30%. We do not list the results by DNMTF and RCC because they return only a single cluster that includes all conferences in the conference domain. This is obviously not what we desire. From Table V, we observe that MCA produces the correct clustering result in the conference domain. The conference cluster in each of the top 4 pairs corresponds to a distinct research area. However, NMTF_Chris makes many mistakes. It splits the conferences from the Database area into two clusters. The third and fourth conference clusters are mixtures of conferences from different areas. In the author domain,

for each author cluster, the percentage of authors from each of the 4 research areas is also shown in Table V. Each author cluster returned by MCA is primarily dominated by authors from one research area, as indicated by the largest percentage highlighted in bold in each column, which perfectly matches the area suggested by the associated conference cluster. For example, consider the 1st pair of conference cluster and author cluster returned by MCA. The conference cluster includes PODS, SIGMOD, VLDB, ICDE and EDBT, all of which are Database conferences. 94.2% of authors in the author cluster also come from the Database area. MCA correctly infers this association between the author cluster and conference cluster. We can make same observation on the remaining cluster pairs by MCA in Table V. It demonstrates that MCA can discover meaningful associations between clusters from different domains. From Table V, we also observe that some conference clusters and author clusters discovered by NMTF_Chris represent a mixture of multiple research areas. To further quantify this observation, for the 4 author clusters, we compute the KL-divergence between author's research area distributions of each pair of author clusters. A KL-divergence of 1 indicates that authors in the two clusters are from two distinct areas. A KL-divergence of 0 indicates that the two author clusters have identical research area distributions and thus are not distinguishable from each other. We use dark color for small KL-divergence and light color for large KL-divergence in Fig. 7. A diagonal entry depicts the KL-divergence of an author cluster to itself which is always 0. Off-diagonal entries correspond to KL-divergence of two author clusters — the larger the KL-divergence, the better the clustering result. We observe that the 4 clusters by MCA all have large KL-divergence to each other but the first two clusters by NMTF_Chris have small KL-divergence. This proves that MCA is more robust to random noise.

Duplicate Names—It is possible that some authors may have the same name in the DBLP database. Publications by these authors might be mistakenly associated with other authors with the same name. In order to evaluate the robustness of our method in this context, we first randomly pick a certain percentage of authors and then randomly pair them up. We "pretend" that the two authors in each pair use the same name and thus their publications are not distinguishable. We replace the two corresponding row vectors in $\mathbf{W}_{12}$ by their average vector. Since only a small percentage of authors may have this issue, we only test the robustness for up to 30% of duplicate names. Fig. 8 shows that MCA can successfully resolve the ambiguity and thus its clustering accuracies and cluster association accuracy are not impacted. NMTF_Chris is the second best, having comparable accuracies in cluster association and author clustering, but failing short on the conference clustering accuracy.

D. Yeast eQTL Dataset

Expression quantitative trait loci (eQTL) mapping is the process of identifying single nucleotide polymorphisms (SNPs) that play important role in the expression of genes. It has been widely used to dissect genetic basis of complex traits [5]. Traditionally, associations between individual expression traits and SNPs are assessed separately [20]. Since genes in the same biological pathways are often coregulated and may share a common genetic basis [21], it is crucial to understand how multiple modesetly-associated SNPs interact to influence the phenotypes [22]. To answer this question, several approaches have been proposed to study the joint effect of multiple SNPs by testing the association between a set

of SNPs and a gene expression trait [23]–[26]. Despite their success, these methods have two common limitations. First, only the association between a set of SNPs and a single expression trait is studied. Therefore, they overlook the joint effect of a set of SNPs on the activities of a set of genes, which may act and interact with each other to achieve certain biological function. Second, the SNP sets used in these methods are usually taken from known biological pathways, which are far from being complete. These methods cannot identify unknown associations between SNP sets or gene sets. To better elucidate the genetic basis of gene expression and understand the underlying biology pathways, it is highly desirable to develop methods that can automatically infer association between a group of SNPs and a group of genes. The process of identifying such associations is referred to as *group-wise* eQTL mapping, to distinguish it from the *individual* eQTL mapping [6] process that identifies associations between individual SNPs and genes. The MCA method proposed in this paper is suitable for *group-wise* eQTL mapping.

We compare MCA with NMTF_Chris, DNMTF and RCC on a yeast eQTL dataset [27]. This dataset originally includes expression profiles of 6229 genes and genotype profiles of 2956 SNPs. After preprocessing (e.g., removing missing values), the dataset is reduced to 1017 SNPs and 4474 genes expression profiles.

We denote the SNP domain as $\mathscr{D}_1$ and the gene domain as $\mathscr{D}_2$, respectively. The SNP interaction network $\mathscr{G}_1$ is generated as in [28]. The gene interaction network $\mathscr{G}_2$ is constructed by computing the Pearson's correlation of the expression levels of each pair of genes.

$$r_{XY}=\frac{\sum_i \left(X_i-\overline{X}\right)\left(Y_i-\overline{Y}\right)}{\sqrt{\sum_i\left(X_i-\overline{X}\right)^2}\sqrt{\sum_i\left(Y_i-\overline{Y}\right)^2}} \quad (24)$$

where $\mathbf{X}$ and $\mathbf{Y}$ are vectors representing the expression profiles of the two genes. $\mathbf{X}_i$ and $\mathbf{Y}_i$ are the ith components of $\mathbf{X}$ and $\mathbf{Y}$, respectively. From Eq. (24), the value of Pearson's correlation r ranges from −1 to 1, where 1 means that two genes are completely positively correlated and −1 means that they are completely negatively correlated. The edge between genes $\mathbf{X}$ and $\mathbf{Y}$ in the gene interaction network is weighted by $|r_{XY}|$. The association matrix $\mathbf{W}_{12}$ is given by the association tests between individual SNPs and individual genes using PLINK [29].

Gene Ontology Enrichment Analysis—Since there is no ground-truth in the Yeast eQTL dataset, we cannot measure the clustering accuracy and cluster association accuracy directly. Here, we evaluate the quality of our result by the Gene Ontology Enrichment Analysis (GOEA) [30]. For each inferred gene cluster c_i, we identify the most significantly enriched Gene Ontology categories [31]. The significance (p-value) is determined by the Fisher's exact test. The raw p-values are further calibrated to correct for the multiple testing problem [32]. To compute calibrated p-values for c_i, we perform a randomization test, wherein we apply the same test to 1000 randomly created gene sets that have the same number of genes as in c_i. In order to evaluate the clusters in the SNP domain, we first need to map the SNPs in a cluster to their nearest genes on the genome, and then apply the

standard procedure of GOEA on the set of genes to compute a p-value. In Fig. 9, clusters are arranged in ascending order of their p-values. We consider the clusters with p-value less than 0.05 to be significant. The numbers of significant gene and SNP clusters are listed in Table VI. Not surprisingly, MCA can identify more significant clusters in both gene and SNP domains than the competitors.

E. Gene Disease Dataset

We further evaluate our algorithm MCA on a gene disease network dataset [33]. The dataset contains 590 disease phenotypes in 20 disease classes and 7997 genes in 200 gene pathways. There are two domains: gene domain $\mathscr{D}_1$ and disease phenotype domain $\mathscr{D}_2$. $\mathscr{G}_1$ represents the "functional" relationships between genes which are measured by interactions between the proteins transcribed from the genes, because most genes "perform" their functions through their transcribed proteins. This protein-protein interaction network can be obtained from HPRD [34]. The relationships among phenotypes are represented by a phenotype similarity network $\mathscr{G}_2$, which is obtained from [35]. It is an undirected network with vertices representing OMIM [36] disease phenotypes and edges (with weights between 0 and 1) representing the similarities between phenotypes measured by their co-occurrences in clinical synopsis records. The associations between disease phenotypes and genes are also available in OMIM. We evaluate the clustering accuracy in each domain using the normalized MI discussed in Section IV-A. The first row in Table VII is the normalized MI in the phenotype domain and the second row is the normalized MI in the gene domain. As we can see, MCA is again the winner.

F. Performance Evaluation

In this section, we study the run-time performance of MCA, measured by the number of iterations before converging to a local optima. Table VIII summarizes the network size and the number of iterations upon convergence on difference data sets. We observe that MCA can converge within a reasonable number of iterations even for large networks. As expected, the number of iterations will increase as the network size increases. Usually, several hundreds of iterations are needed before convergence, but the actual running time is fast. Table IX shows the time used by different methods to convergence on the DBLP dataset. All methods except RCC run very fast. We can conclude that MCA can achieve much better accuracy without entailing more computation time.

V. CONCLUSION

In this paper, we propose a novel algorithm, MCA, for network clustering across multiple related domains. By leveraging the *duality* between single network clustering and inferring cross-network cluster alignment, MCA well incorporates any prior knowledge on cross-network instance relationships into multi-network clustering. The algorithm is robust to noise and is capable of detecting cross-domain associations between clusters, which, was never addressed in previous study. Extensive experiments on both synthetic and several real datasets demonstrate the effectiveness and efficiency of MCA and its advantages over existing methods.

Acknowledgments

This work was partially supported by the National Science Foundation grants IIS-1162374, IIS-1218036, IIS-1313606 and IIS-1017415, by the Army Research Laboratory under Cooperative Agreement Number W911NF-09-2-0053, by National Institutes of Health under the grant number R01LM011986 and U54GM114833-01, Region II University Transportation Center under the project number 49997-33 25.

References

1. MacQueen J, et al. Some methods for classification and analysis of multivariate observations. Proceedings of the fifth Berkeley symposium on mathematical statistics and probability. 1967; 1:281–297.
2. Kuang D, Park H, Ding CH. Symmetric nonnegative matrix factorization for graph clustering. SDM. 2012; 12:106–117.
3. Cheng W, Zhang X, Guo Z, Wu Y, Sullivan PF, Wang W. Flexible and robust co-regularized multi-domain graph clustering. KDD. 2013:320–328.
4. Chaudhuri K, Kakade SM, Livescu K, Sridharan K. Multi-view clustering via canonical correlation analysis. ICML. 2009:129–136.
5. Michaelson JJ, Loguercio S, Beyer A. Detection and interpretation of expression quantitative trait loci (eqtl). Methods. 2009; 48(3):265–276. [PubMed: 19303049]
6. Cheng W, Yu S, Zhang X, Wang W. Fast and robust group-wise eQTL mapping using sparse graphical models. BMC Bioinformatics. 2015; 16:2. [PubMed: 25593000]
7. Newman ME. Modularity and community structure in networks. PNAS. 2006; 103(23):8577–8582. [PubMed: 16723398]
8. Shi J, Malik J. Normalized cuts and image segmentation. TPAMI. 2000; 22(8):888–905.
9. Von Luxburg U. A tutorial on spectral clustering. Statistics and computing. 2007; 17(4):395–416.
10. Sun Y, Han J, Zhao P, Yin Z, Cheng H, Wu T. Rankclus: integrating clustering with ranking for heterogeneous information network analysis. EDBT. 2009:565–576. ACM.
11. Ding C, Li T, Peng W, Park H. Orthogonal nonnegative matrix t-factorizations for clustering. KDD. 2006:126–135.
12. Shang F, Jiao L, Wang F. Graph dual regularization non-negative matrix factorization for co-clustering. Pattern Recognition. 2012; 45(6):2237–2250.
13. Du L, Shen Y-D. Towards robust co-clustering. IJCAI. 2013:1317–1322.
14. Dhillon IS, Mallela S, Modha DS. Information-theoretic co-clustering. KDD. 2003:89–98.
15. Lee DD, Seung HS. Algorithms for non-negative matrix factorization. NIPS. 2000:556–562.
16. Ji, M.; Sun, Y.; Danilevsky, M.; Han, J.; Gao, J. Machine Learning and Knowledge Discovery in Databases. Springer; 2010. Graph regularized transductive classification on heterogeneous information networks; p. 570-586.
17. Gao J, Liang F, Fan W, Sun Y, Han J. Graph-based consensus maximization among multiple supervised and unsupervised models. NIPS. 2009:585–593.
18. Cheung VG, Spielman RS, Ewens KG, Weber TM, Morley M, Burdick JT. Mapping determinants of human gene expression by regional and genome-wide association. Nature. 2005; 437(7063): 1365–1369. [PubMed: 16251966]
19. Musani SK, Shriner D, Liu N, Feng R, Coffey CS, Yi N, Tiwari HK, Allison DB. Detection of gene x gene interactions in genome-wide association studies of human population data. Human heredity. 2007; 63(2):67–84. [PubMed: 17283436]
20. Lander ES. Initial impact of the sequencing of the human genome. Nature. 2011; 470(7333):187–197. [PubMed: 21307931]
21. Holden M, Deng S, Wojnowski L, Kulle B. Gsea-snp: applying gene set enrichment analysis to snp data from genome-wide association studies. Bioinformatics. 2008; 24(23):2784–2785. [PubMed: 18854360]

22. Wu MC, Lee S, Cai T, Li Y, Boehnke M, Lin X. Rare-variant association testing for sequencing data with the sequence kernel association test. The American Journal of Human Genetics. 2011; 89(1):82–93. [PubMed: 21737059]
23. Braun R, Buetow K. Pathways of distinction analysis: a new technique for multi–snp analysis of gwas data. PLoS Genetics. 2011; 7(6):e1002101. [PubMed: 21695280]
24. Listgarten J, Lippert C, Kang EY, Xiang J, Kadie CM, Heckerman D. A powerful and efficient set test for genetic markers that handles confounders. Bioinformatics. 2013; 29(12):1526–1533. [PubMed: 23599503]
25. Brem RB, Storey JD, Whittle J, Kruglyak L. Genetic interactions between polymorphisms that affect gene expression in yeast. Nature. 2005; 436(7051):701–703. [PubMed: 16079846]
26. Lee S, Xing EP. Leveraging input and output structures for joint mapping of epistatic and marginal eqtls. Bioinformatics. 2012; 28(12):i137–i146. [PubMed: 22689753]
27. Purcell S, Neale B, Todd-Brown K, Thomas L, Ferreira MA, Bender D, Maller J, Sklar P, De Bakker PI, Daly MJ, et al. Plink: a tool set for whole-genome association and population-based linkage analyses. The American Journal of Human Genetics. 2007; 81(3):559–575. [PubMed: 17701901]
28. Subramanian A, Tamayo P, Mootha VK, Mukherjee S, Ebert BL, Gillette MA, Paulovich A, Pomeroy SL, Golub TR, Lander ES, et al. Gene set enrichment analysis: a knowledge-based approach for interpreting genome-wide expression profiles. PNAS. 2005; 102(43):15545–15550. [PubMed: 16199517]
29. Cheng W, Zhang X, Wu Y, Yin X, Li J, Heckerman D, Wang W. Inferring novel associations between snp sets and gene sets in eqtl study using sparse graphical model. Proceedings of the ACM Conference on Bioinformatics, Computational Biology and Biomedicine. 2012:466–473. ACM.
30. Westfall, PH. Resampling-based multiple testing: Examples and methods for p-value adjustment. Vol. 279. John Wiley & Sons; 1993.
31. Hwang T, Atluri G, Xie M, Dey S, Hong C, Kumar V, Kuang R. Co-clustering phenome–genome for phenotype classification and disease gene discovery. Nucleic acids research. 2012; 40(19):e146–e146. [PubMed: 22735708]
32. Peri S, Navarro M, et al. Development of human protein reference database as an initial platform for approaching systems biology in humans. Genome research. 2003; 13(10):2363–2371. [PubMed: 14525934]
33. van Driel MA, Bruggeman J, Vriend G, Brunner HG, Leunissen JA. A text-mining analysis of the human phenome. European journal of human genetics. 2006; 14(5):535–542. [PubMed: 16493445]
34. Hamosh A, Scott AF, Amberger JS, Bocchini CA, McKusick VA. Online mendelian inheritance in man (omim), a knowledge-base of human genes and genetic disorders. Nucleic acids research. 2005; 33(suppl 1):D514–D517. [PubMed: 15608251]

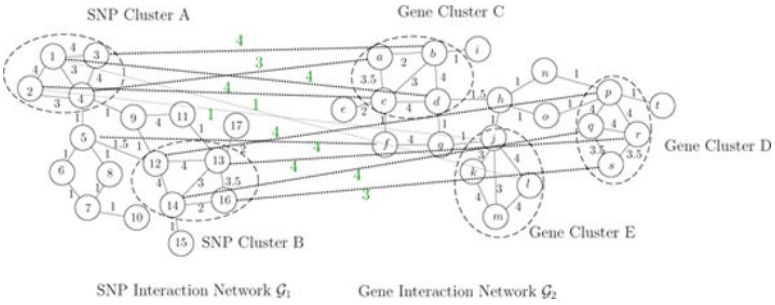


Fig. 1.

An exemplar SNP interaction network and gene interaction network in an eQTL study

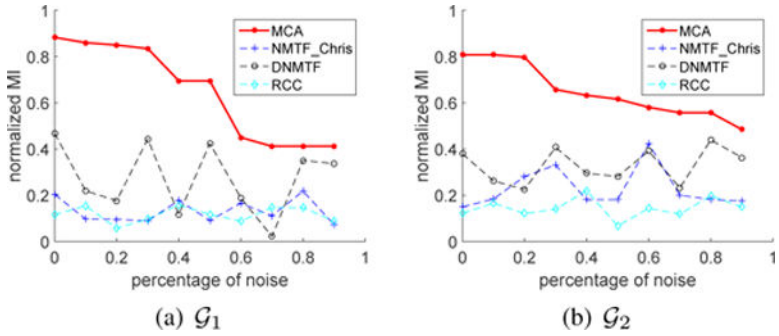


Fig. 2.

Clustering accuracy as a function of increasing percentage of noise in $\mathbf{W}_{12}$ on simulated data.

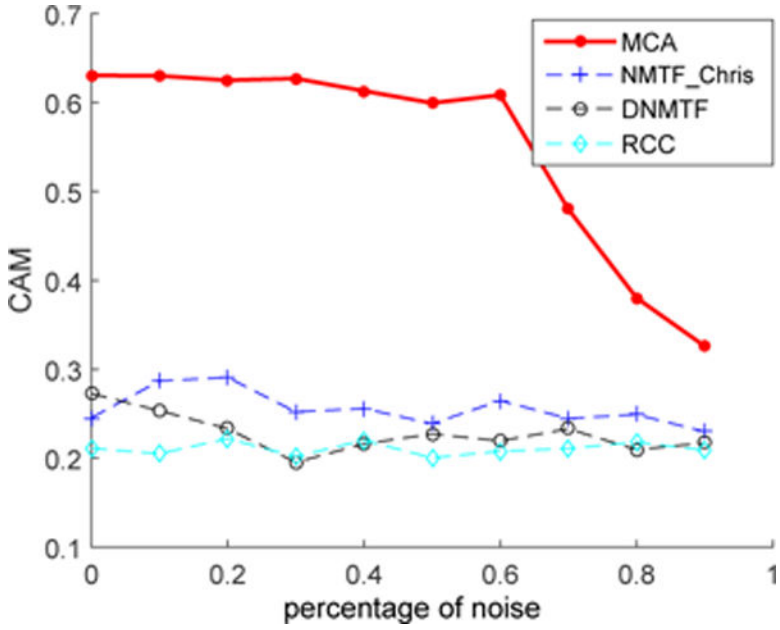


Fig. 3.

Cluster association accuracy as a function of increasing percentage of noise in $\mathbf{W}_{12}$ on simulated data.

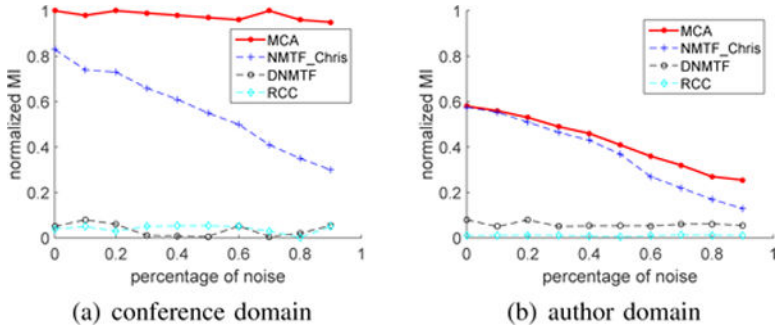


Fig. 4.

Normalized Mutual Information with respect to different noise levels on the DBLP dataset.

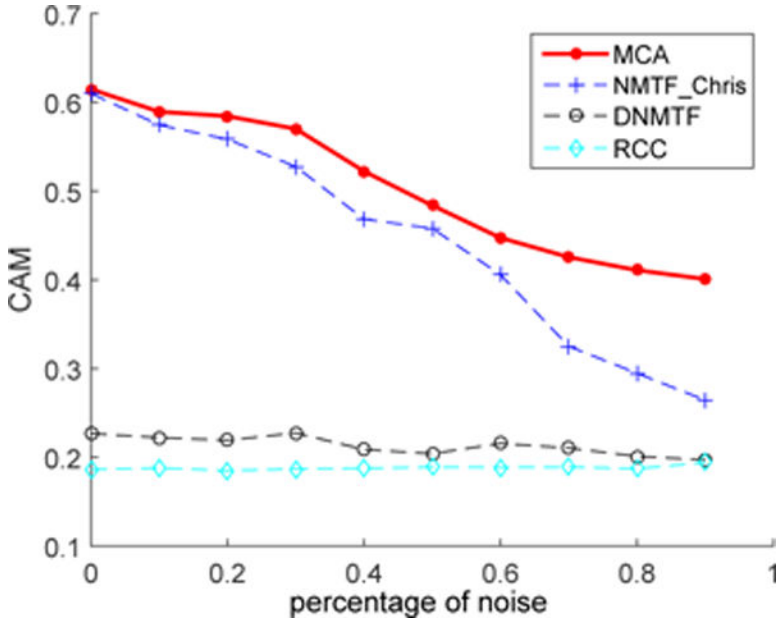


Fig. 5.
Cluster association accuracy with respect to different noise levels on the DBLP dataset.

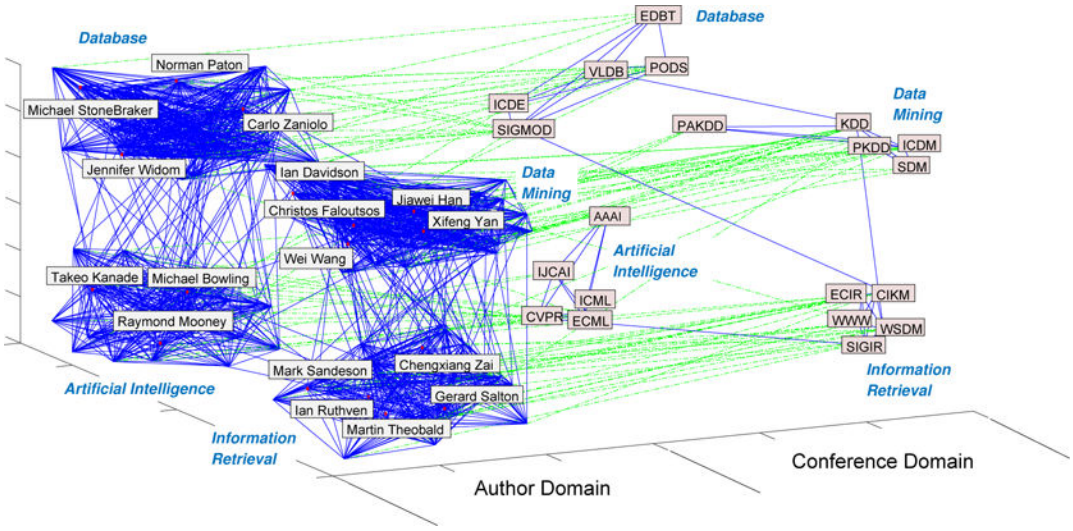


Fig. 6.

A snapshot of the real DBLP network. We only display edges whose weights are above some threshold.

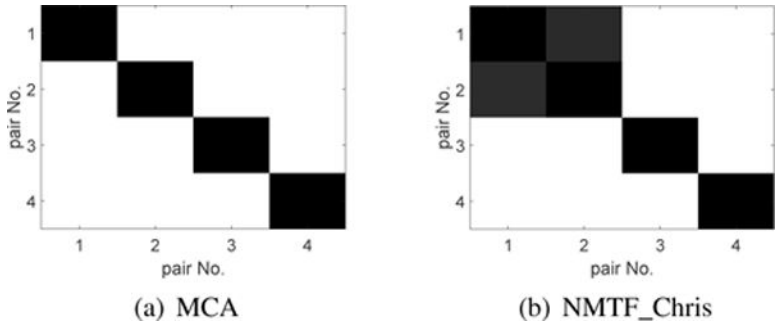
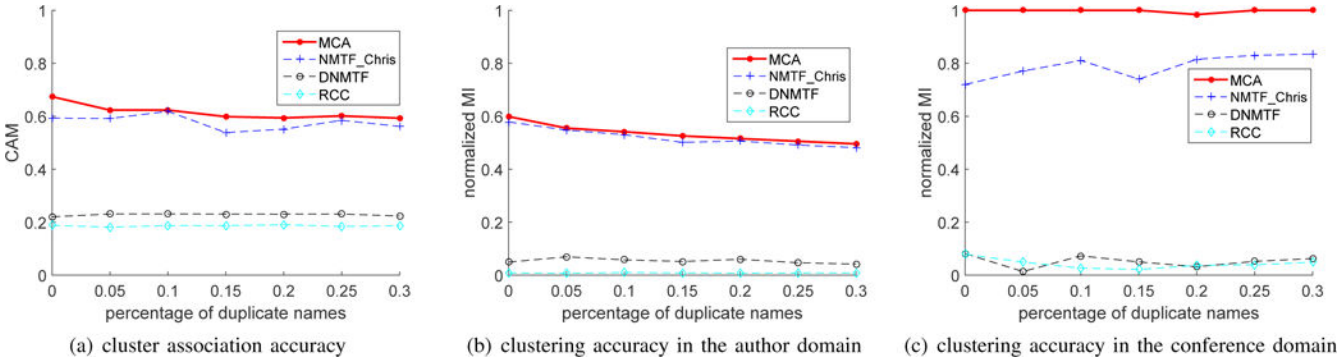


Fig. 7.

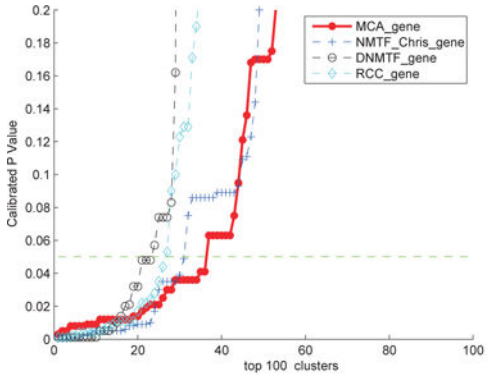
The pairwise KL-divergence between research area distributions of author clusters in Table V.



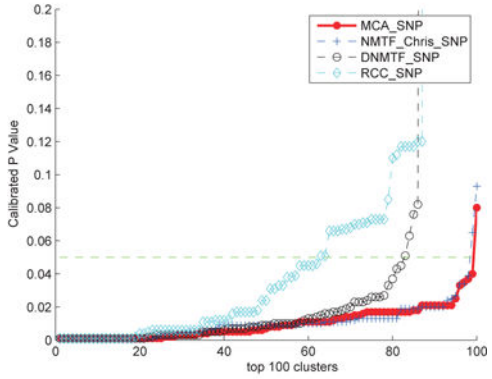
(a) cluster association accuracy (b) clustering accuracy in the author domain (c) clustering accuracy in the conference domain

Fig. 8.

Results on the DBLP dataset with duplicate names.



(a) gene clusters



(b) SNP clusters

Fig. 9.

Gene ontology enrichment analysis on the yeast eQTL data.

TABLE I

Cross-domain associations between SNP clusters and gene clusters in Fig. 1

association	1st cluster pair	2nd cluster pair
SNP interaction network $\mathcal{G}_1$	{12, 13, 14, 16}	{1, 2, 3, 4}
gene interaction network $\mathcal{G}_2$	{*p*, *q*, *r*, *s*}	{*a*, *b*, *c*, *d* }

TABLE II

Summary of symbols and their meanings

Symbol	Description
N	The number of domains
$\mathscr{D}_p$	The p-th domain
n_p	The number of instances in $\mathscr{D}_p$
k_p	The number of clusters in $\mathscr{D}_p$
$\mathscr{G}_p$	The network representing relationship among instances in $\mathscr{D}_p$
$\mathbf{A}_p$	The affinity/adjacency matrix of $\mathscr{G}_p$
$\mathbf{W}_{pq}$	The cross-domain relationship between instances from $\mathscr{D}_p$ and $\mathscr{D}_q$
$\mathbf{H}_p$	The cluster assignment matrix in $\mathscr{D}_p$
$\mathbf{S}_{pq}$	The cross-domain alignment matrix between $\mathscr{D}_p$ and $\mathscr{D}_q$

TABLE III

List of conferences from each research area in the DBLP dataset

DB	AI	DM	IR
PODS	AAAI	KDD	SIGIR
SIGMOD	ICML	ICDM	WWW
VLDB	IJCAI	SDM	WSDM
EDBT	CVPR	PKDD	ECIR
ICDE	ECML	PAKDD	CIKM

TABLE IV

Number of authors from each research area in the DBLP dataset

	DB	AI	DM	IR
number	1197	1109	745	1006
percentage	29.5%	27.3%	18.4%	24.8%

TABLE V

The results on DBLP by MCA and NMTF_Chris at 30% noise level

			pair No. 1	pair No. 2	pair No. 3	pair No. 4
MCA	conference		PODS SIGMOD VLDB ICDE EDBT	KDD ICDM SDM PKDD PAKDD	AAAI ICML IJCAI CVPR ECML	SIGIR WWW WSDM ECIR CIKM
	author	DB	**0.943**	0.134	0.038	0.293
		DM	0.036	**0.835**	0.151	0.088
		AI	0.007	0.000	**0.698**	0.00
		IR	0.014	0.031	0.113	**0.619**
NMTF_Chris	conference		VLDB ICDE EDBT	PODS SIGMOD	AAAI ICML IJCAI ECML PKDD PAKDD	SIGIR WWW WSDM ECIR CIKM KDD ICDM SDM CVPR
	author	DB	**0.736**	**0.868**	0.035	0.084
		DM	0.236	0.088	0.292	0.361
		AI	0.000	0.022	**0.602**	0.042
		IR	0.028	0.022	0.071	**0.513**

Proc IEEE Int Conf Data Min. Author manuscript; available in PMC 2016 May 25.

Author Manuscript

Author Manuscript

Author Manuscript

TABLE VI

The number of significantly enriched clusters measured by GOEA

	MCA	NMTF_Chris	DNMTF	RCC
gene	**36**	31	23	26
SNP	**99**	98	82	62

TABLE VII

Results on the gene disease network

method	MCA	NMTF_Chris	DNMTF	RCC
Normalized MI_pheno	**0.19**	0.13	0.14	0.15
Normalized MI_gene	**0.05**	0.02	**0.05**	0.04

TABLE VIII

Number of iterations to converge

dataset	size of network $\mathcal{G}_1$	size of network $\mathcal{G}_2$	number of iterations
synthetic	17	20	24
DBLP	441	20	80
eQTL	1017	4474	741
Gene-disease	3619	366	144

Author Manuscript

Author Manuscript

Author Manuscript

TABLE IX

Amount of time to converge on DBLP

method	MCA	NMTF_Chris	DNMTF	RCC
time cost (second)	0.8	0.4	0.4	11.5