Meta-Path Graphical Lasso for Learning Heterogeneous Connectivities

Yao Zhang * Yun Xiong * Xinyue Liu † Xiangnan Kong † Yangyong Zhu *

Abstract

Sparse inverse covariance estimation has attracted lots of research interests since it can recover the structure of the underlying Gaussian graphical model. This is a useful tool to demonstrate the connections among objects (nodes). Previous works on sparse inverse covariance estimation mainly focus on learning one single type of connections from the observed activities with a lasso, group lasso or tree-structure penalty. However, in many real-world applications, the observed activities on the nodes can be related to multiple types of connections. In this paper, we consider the problem of learning heterogeneous connectivities from the observed activities by incorporating meta paths extracted from a heterogeneous information network (HIN), an information network with multiple types of nodes and links, into the conventional graphical lasso framework. We aim at extracting the strongest type of relation between any pairs of entities and ignoring other minor relations. Specially, we introduce two novel kinds of constraints: meta path constraints and exclusive constraints, which ensure the unique type of relation among a pair of objects. This problem is highly challenging due to the non-convex optimization. We proposed a method based upon the alternating direction method of multipliers (ADMM) to efficiently solve the problem. The conducted experiments on both synthetic and real-world datasets illustrate the effectiveness of the proposed method.

1 Introduction

In recent years, undirected graphical models have gained prominence since they provide a natural way to model the complex interactions among a set of random variables. For example, a group of genes tends to work together if they perform the same biological functions, and there are some regulatory relationships among genes [5]. By treating genes as nodes and their regulatory relationships as the corresponding edges, the biological system can be represented as a graph. However, in many applications, the structure of the graph is unknown and must be inferred from the limited observations.

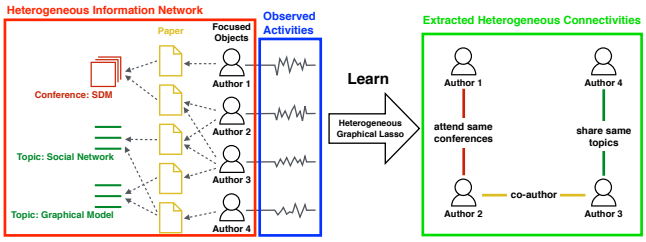


Figure 1: An illustration of learning heterogeneous connections from the observed activities of the focused objects (nodes) by incorporating a heterogeneous information network.

One of the widely used graphical models is the Gaussian Graphical Model (GGM) [21], which assumes the variables follow a multivariate Gaussian distribution. In the framework of GGM, the problem of learning the structure of a graphical model is equivalent to estimating the inverse of the covariance matrix, also referred to as the precision or concentration matrix, since the non-zero pattern of this precision matrix corresponds to the edges in the underlying graph structure [21]. It is natural to enforce the sparseness on the precision matrix to get an interpretable graph and maintain the low model complexity. For this purpose, some researchers considered the sparse inverse covariance matrix estimation problem [1,8,15,17,18,22], which is also known as Graphical Lasso (GLasso).

However, most learning algorithms suffer from a very high computational complexity and are impractical when the number of nodes exceeds tens of thousands. So some additional information should be considered to reduce the search space. M. Grechkin *et al.* [11] proposed a new method called Pathway Graphical Lasso (PathGLasso), inspired by the fact that in the biological system, a pair of genes will not be connected if they do not participate together in any of the cellular processes, typically referred to as pathways. By taking advantage of the domain knowledge, this method gains great acceleration and gives a more meaningful result.

Conventional approaches for graphical lasso mainly focus on learning one type of relation from the node activities. However, in many real-world application, the node activities can usually be explained by multiple

*Department of Computer Science, Fudan University
†Department of Computer Science, Worcester Polytechnic Institute

types of relations among the nodes. For example, in bibliographic networks, the publication activities can be influenced by multiple types of relationships among the authors. In Figure 1, the publication activities of Author 2 can be not only influenced by the "co-author" relation with Author 3, but also be in affected by the "attend-same-conference" relationship with Author 1.

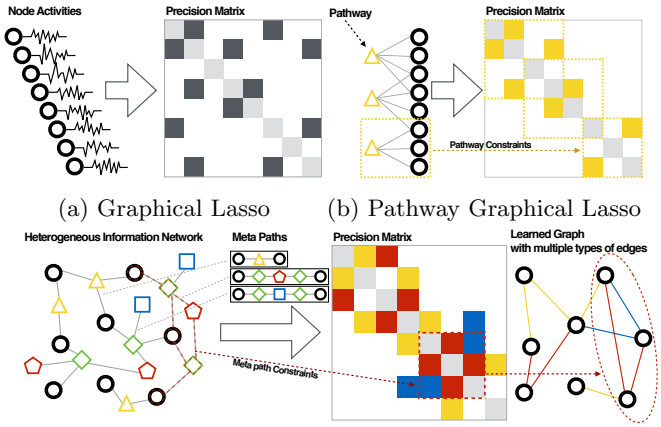
With the recent advance in data collection techniques, many real-world applications are facing large scale heterogeneous information networks (HIN) with multiple types of objects interconnected through multiple types links, which involves a significant amount of information. With the help of the additional information, we can infer the connections between the objects from different perspectives.

In this paper, we study the problem of learning heterogeneous connectivities between objects in a HIN from the observed activities, where the goal is to learn which pairs of objects have connections, and in the meantime identify the main sources of these connections in HIN. This problem is very important in various applications, *e.g.*, learning the main reason for gene regulatory relationships in biological networks and learning the main types of relationships between users in social networks. This is a challenging problem due to the following reasons:

• **Large Search Space**: There are tens of thousands of objects in a real-world network, and there are various types of possible relations among them. We need to prune the impossible links in advance to reduce the search space. For example, in Figure 1, if we only care about the three simple relations listed in the right part of figure, we can assert that Author 1 and 4 cannot be connected in the learned graph because they do not have any of the relations.

• **Exclusive Relations**: We aim at extracting the strongest type of relations between a pair of objects and ignoring other minor relations. For example, in Figure 1, Author 2 and Author 3 not only are co-authors but also have attended the same conferences and share the same topics. In that case, we only care about the main reason for their connection, *i.e.*, co-author relation, ignoring all other types of links. This means we need to ensure there is at most one type of relations among a pair of objects in the result. Maintaining the exclusive relations is a challenging task.

To address the above issues, we incorporate meta paths extracted from a HIN into the conventional graphical lasso framework. We present a novel method called Heterogeneous Graphical Lasso (HeteGLasso), and solve it using the non-convex alternating direction method of multipliers (ADMM) [2, 7]. HeteGLasso estimates multiple precision matrices simultaneously



(a) Graphical Lasso (b) Pathway Graphical Lasso

(c) Heterogeneous Graphical Lasso

Figure 2: Comparison of three different methods for graphical lasso. (a) Graphical Lasso [8] only uses the observed activities and outputs a sparse precision matrix. (b) Pathway Graphical Lasso [11] further considers the pathway-based prior. Pathways constrain the nonzero pattern in the precision matrix. (c) Heterogeneous Graphical Lasso incorporates a HIN to learn multiple types of connections simultaneously. Meta path-based constraints regulate at which position elements can be nonzero, while exclusive constraints ensure the unique type of the edge between a pair of objects.

by incorporating the objects' individual activities and the external heterogeneous information network. Each learned graph corresponds to a certain kind of relations. The learned precision matrices have property that there is at most one nonzero element among all matrices at each off-diagonal position (i, j). This means multiple precision matrices and corresponding graphs can be easily combined as a graph with multiple types of edges. We compare and contrast the inputs and outputs of 3 methods in Figure 2. As we can see, compared to the conventional GLasso, PathGLasso uses domain knowledge as a prior, which can be seen as a bipartite graph, while HeteGLasso uses a HIN as a prior.

The main contributions of our paper are as follows:

• We study the problem of learning heterogeneous connectivities in a HIN and formulate the problem as Heterogeneous Graphical Lasso. We use meta paths extracted from the HIN to define the path groups among objects and introduce meta path constraints in HeteGLasso. Further, meta path-based dissimilarity is used to infer the element-wise regularization matrices.

• To extract the strongest relations among objects, we introduce the exclusive constraints in HeteGLasso using cardinality function, which ensure the uniqueness of link between a pair of objects in the learned graph.

• We propose an efficient algorithm to solve HeteGLasso using non-convex ADMM. The experimental results

Table 1: Mathematical Notation.

Notation	Description
$\Re$	The space of real numbers
$\mathcal{S}_+^n$	The space of $n \times n$ symmetric positive semi-definite matrices
$\mathcal{S}_{++}^n$	The space of $n \times n$ symmetric positive definite matrices
$A \succ \mathbf{0}$	A is symmetric and positive definite
$\|A\|_{1,\Lambda}$	Element-wise ℓ_1 norm, *i.e.*, $\|A\|_{1,\Lambda} = \sum_{ij} \Lambda_{ij}\|A_{ij}\|$
$\|A\|_F$	Frobenius norm of matrix A, *i.e.*, $\|A\|_F = \sqrt{\sum_{ij} A_{ij}^2}$
$A_{ij}^{(\cdot)}$	A vector consisting of elements among all $A^{(k)}$ at position (i,j), *i.e.*, $\left(A_{ij}^{(1)}, A_{ij}^{(2)}, \ldots, A_{ij}^{(K)}\right)$
$\mathbf{card}(\mathbf{x})$	Cardinality function on a vector $\mathbf{x}$, which gives the number of nonzero elements in $\mathbf{x}$
$\mathbf{sgn}(c)$	Sign function on a scalar c, $\mathbf{sgn}(c) = 1$ for $c > 0$, $\mathbf{sgn}(c) = -1$ for $c < 0$ and $\mathbf{sgn}(c) = 0$ for $c = 0$
$\Pi_{\mathcal{D}}(\mathbf{x})$	The projection of a vector $\mathbf{x}$ on set $\mathcal{D}$

show the effectiveness of our method.

2 Problem Formulation

In this section, we briefly formulate the problem of learning heterogeneous connections and the concepts of heterogeneous information network. Mathematical notations used throughout the paper are summarized in Table 1.

2.1 Preliminaries

2.1.1 Graphical Lasso. Given a dataset consisting of samples drawn i.i.d. from an n-variate Gaussian distribution with zero mean and covariance matrix $\Sigma \in \mathcal{S}_{++}^n$:

$$x_i \sim \mathcal{N}_n(0, \Sigma), i = 1, 2, \ldots, m,$$

the task of graphical lasso is to estimate its precision matrix $\Theta = \Sigma^{-1} \in \mathcal{S}_{++}^n$ under the prior assumption that Θ is sparse. To achieve a sparse estimate, a number of works [1,8,15,17,18,22] have considered the following Graphical Lasso (GLasso) problem:

$$\min_{\Theta} \quad -l(S, \Theta) - \|\Theta\|_{1,\Lambda},$$

where $S = \frac{1}{m}\sum_{i=1}^m x_i x_i^T \in \mathcal{S}_+^n$ is the sample covariance matrix, Λ is an ℓ_1 regularization parameter matrix with $\Lambda_{ij} > 0$ for all off-diagonal elements and $\Lambda_{ij} \geqslant 0$ for all diagonal elements, and $l(S, \Theta) = \log\det\Theta - \mathrm{Tr}(S\Theta)$ is the log-likelihood function. Note that the constraint $\Theta \succ \mathbf{0}$ is implicit because of the convention that $\log\det(\Theta) = -\infty$ when $\Theta \not\succ \mathbf{0}$.

2.1.2 Heterogeneous Information Network.

Definition 1 (Heterogeneous Information Network [16, 19]). A heterogeneous information network is a special kind of information network with multiple types of nodes and multiple types of links. It can be represented as a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$. $\mathcal{V}$ denotes the set of nodes, which involves w types of objects: $\mathcal{V}^{(1)} = \{v_1^{(1)}, \ldots, v_{|\mathcal{V}^{(1)}|}^{(1)}\}, \ldots, \mathcal{V}^{(w)} = \{v_1^{(w)}, \ldots, v_{|\mathcal{V}^{(w)}|}^{(w)}\}$, where $v_p^{(i)}$ denotes the p-th object of type i. $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ denotes the links between the objects in $\mathcal{V}$, which includes multiple types of links. □

Each type of links starting from a source node of type s and ending at a target node of type t corresponds to a binary relation $R^{(st)}$, where $R_{pq}^{(st)}$ holds if $v_p^{(s)}$and $v_q^{(t)}$are linked by a link of type $R^{(st)}$. For example, in Figure 1, there are 4 types of nodes: *author*, *paper*, *conference*, *topic*, and 3 types of relations(links): *author* $\rightarrow$ *paper*, *paper* $\rightarrow$ *conference*, *paper* $\rightarrow$ *topic*.

Definition 2 (Meta Path [16,19]). A meta path $P = V^{(i_1)} \xrightarrow{R^{(i_1 i_2)}} \cdots \xrightarrow{R^{(i_{l-1} i_l)}} V^{(i_l)}$ is defined as a sequence of relations in the network schema. □

For simplicity, we can use object names to denote the meta path if there exist no multiple relations between the same pair of types: $P = \mathcal{V}^{(i_1)} \cdots \mathcal{V}^{(i_l)}$. We say $v_1 v_2 \cdots v_l$ is a path instance of meta path $\mathcal{V}^{(i_1)} \cdots \mathcal{V}^{(i_l)}$, if $v_p \in \mathcal{V}^{(i_q)}\ \forall\ q$.

2.2 Heterogeneous Graphical Lasso.

First, we give the definition of the path group:

Definition 3 (Path Group). Given a meta path $P = \mathcal{V}^{(s)} \cdots \mathcal{V}^{(t)}$ defined on $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where nodes set $\mathcal{V} = \bigcup_i \mathcal{V}^{(i)}$. A path group $g_j^{(t)}$ is a set of nodes with type $\mathcal{V}^{(s)}$, and we say $v_i^{(s)} \in g_j^{(t)}$ if there exists a path instance $v_i^{(s)} \cdots v_j^{(t)}$ of P. A meta path P can induce $|\mathcal{V}^{(t)}|$ (maybe overlapping) path groups $g_1^{(t)}, \ldots, g_{|\mathcal{V}^{(t)}|}^{(t)}$. □

For example, in the toy bibliographic network shown in Figure 1, meta path *author* $\rightarrow$ *paper* $\rightarrow$ *topic* induces 2 path groups of authors corresponding to 2 topics: $\{A_2, A_3, A_4\}, \{A_3, A_4\}$, and *author* $\rightarrow$ *paper* induces 5 path groups corresponding to 5 papers: $\{A_1\}, \{A_2, A_3\}, \{A_2, A_3\}, \{A_3\}, \{A_4\}$, where A_i denotes the i-th author.

Given a heterogeneous information network $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ consisting of different types of objects, *i.e.*,

$\mathcal{V} = \bigcup_i \mathcal{V}^{(i)}$, we now focus only on one certain type of objects $\mathcal{V}^{(s)}$ with size $|\mathcal{V}^{(s)}| = n$, where each object has its own observed activities, and therefore, the sample covariance matrix $S \in \mathcal{S}_+^n$ can be computed. We call this kind of objects as focused objects. Besides, we are also given K meta paths starting from $\mathcal{V}^{(s)}$, which induce the K sets of path groups. Though we can learn a precision matrix over $\mathcal{V}^{(s)}$ to determine which pairs of objects have direct connections, it is hard to tell what is the main reason for these connections. To address this problem, we can learn K precision matrices $\Theta^{(1)}, \ldots, \Theta^{(K)}$ simultaneously corresponding to K meta paths, with exclusive constraints: at the same off-diagonal position (i, j), there is at most one non-zero element among K precision matrices. We formulate the problem of Heterogeneous Graphical Lasso as follows:

$$\begin{aligned} \min_{\Theta^{(1)},\ldots,\Theta^{(K)}} \quad & \sum_{k=1}^{K} \left(-l(S, \Theta^{(k)}) + \|\Theta^{(k)}\|_{1,\Lambda^{(k)}} \right) \\ \text{s. t.} \quad & \Theta_{ij}^{(k)} = 0, \quad \forall (i,j) \notin \mathcal{P}^{(k)} \\ & \mathbf{card}(\Theta_{ij}^{(\cdot)}) \leqslant 1, \qquad \forall i \neq j, \end{aligned} \tag{2.1}$$

where $(i, j) \in \mathcal{P}^{(k)}$ if there exists a path group of k-th meta path, such that $v_i^{(s)}$ and $v_j^{(s)}$ are all in this group. The first group of constraints $\Theta_{ij}^{(k)} = 0, \quad \forall (i,j) \notin \mathcal{P}^{(k)}, k = 1, 2, \ldots, K$ means that a pair of variables can be connected to each other only if they co-occur in at least one path group. Note that $\mathbf{card}(\cdot)$ is a non-convex and non-smooth function, which leads the entire problem hard to solve. In the next section, we propose an algorithm to solve Problem 2.1 approximately and efficiently.

3 Proposed Method

In this section, we propose an ADMM based algorithm to solve Problem 2.1.

3.1 Nonconvex ADMM. Note that without $\mathbf{card}(\cdot)$ constraints, Problem 2.1 can be decomposed into K independent parts and each part is actually a Pathway Graphical Lasso problem [11]. Define functions $\phi_k : \mathcal{S}_{++}^n \to \Re \cup \{-\infty, \infty\}$ as follows

$$\phi_k(\Theta^{(k)}) = \begin{cases} -l(S, \Theta^{(k)}) + \|\Theta^{(k)}\|_{1,\Lambda^{(k)}} \\ \qquad \text{if } \Theta_{ij}^{(k)} = 0, \forall (i,j) \notin \mathcal{P}^{(k)}, \\ +\infty \qquad \text{otherwise.} \end{cases}$$

$\min \phi_k(\cdot)$ is convex since it is the minimization of a convex function with equality constraints. Now the

Algorithm 1 Algorithm for solving Problem 2.1

Require: $S, \Lambda, \mathcal{P}^{(k)}, \tau > 0$
1: Initialize $W^{(k)} = I, V^{(k)} = O$
2: **repeat**
3: Convex proximal step: Solve Problem 3.2 to update $\Theta^{(k)}$
4: Projection: $W_{ij}^{(\cdot)} \coloneqq \Pi_{\mathcal{D}}(\Theta_{ij}^{(\cdot)} + V_{ij}^{(\cdot)})$
5: Dual update: $V^{(k)} \coloneqq V^{(k)} + \Theta^{(k)} - W^{(k)}$
6: **until** convergence
7: **return** $\Theta^{(k)}$

Problem 2.1 is equvilent to

$$\begin{aligned} \min_{\Theta^{(1)},\ldots,\Theta^{(K)}} \quad & \sum_{k=1}^{K} \phi_k(\Theta^{(k)}) \\ \text{s. t.} \quad & \Theta_{ij}^{(\cdot)} \in \mathcal{D}, \forall i \neq j, \end{aligned} \tag{3.1}$$

where $\mathcal{D} = \{\mathbf{x} | \, \mathbf{card}(\mathbf{x}) \leqslant 1\}$.

Problem 3.1 consists of a convex objective function and non-convex constraints, which can be solved heuristically via non-convex ADMM [2, 7]. The algorithm is described in Algorithm 1.

The non-convex ADMM consists of 3 steps inside each iteration: The first convex proximal step involves solving the relaxed problem with additional Frobenius norm,

$$\min_{\Theta^{(1)},\ldots,\Theta^{(K)}} \sum_{k=1}^{K} \left[\phi_k(\Theta^{(k)}) + \frac{\tau}{2} \|\Theta^{(k)} - W^{(k)} + V^{(k)}\|_F^2 \right], \tag{3.2}$$

where $\tau > 0$ is the model parameter, $W^{(k)}$'s and $V^{(k)}$'s are auxiliary variables with initial value $W^{(k)} = I, V^{(k)} = O$. This problem can be solved separately with respect to k and is discussed in detail in next subsection.

The second step $W_{ij}^{(\cdot)} \coloneqq \Pi_{\mathcal{D}}(\Theta_{ij}^{(\cdot)} + V_{ij}^{(\cdot)})$ is to project the variables onto the non-convex set $\mathcal{D}$, which can be done easily in our setting: $W_{ij}^{(\cdot)}$ keeps the element of the largest magnitude in $(\Theta_{ij}^{(\cdot)} + V_{ij}^{(\cdot)})$ and zeroes out the rest. The final step is to update dual variables.

3.2 Inner Subproblem. To finish Algorithm 1, we need to optimize the separable convex optimization Problem 3.2. Consider the following subproblem(suppressing superscript k for simplicity):

$$\begin{aligned} \min_{\Theta} \quad & -l(S, \Theta) + \|\Theta\|_{1,\Lambda} + \frac{\tau}{2} \|\Theta - W + V\|_F^2 \\ \text{s. t.} \quad & \Theta_{ij} = 0, \qquad \forall (i,j) \notin \mathcal{P}, \end{aligned} \tag{3.3}$$

which is a graphical lasso problem with additional Frobenius norm and meta path constraints.

Now we take advantage of our prior knowledge: a pair of variables can be connected to each other only if they co-occur in at least one path group. Following the method proposed in [11], we iteratively update the parameters that correspond to one path group, with all of the other parameters held fixed. After re-arranging the variables, Θ takes the form:

$$\Theta = \begin{bmatrix} A & B & 0 \\ B^T & C & D \\ 0 & D^T & E \end{bmatrix},$$

where $\Theta_1 = \begin{bmatrix} A & B \\ B^T & C \end{bmatrix}$ contains the parameters in the current path group, $\Theta_2 = \begin{bmatrix} C & D \\ D^T & E \end{bmatrix}$ contains the parameters in the rest of the path groups, and C corresponds to the overlapping part.

To update A, B, C with D, E fixed, the Problem 3.3 is reduced to

$$\min \quad -l(S_1, \Omega) + \|\Omega + \Delta\|_{1,\Lambda_1} + \frac{\rho}{2}\|\Omega + \Delta - W_1 + V_1\|_F^2, \tag{3.4}$$

where $\Omega = \Theta_1 - \Delta$, $\Delta = \begin{bmatrix} 0 \\ D \end{bmatrix} \cdot E^{-1} \cdot \begin{bmatrix} 0, D^T \end{bmatrix}$, and S_1, W_1, V_1 are the corresponding part of S, W, V. We refer readers to [11] for more details about how to calculate E^{-1} efficiently.

Problem 3.4 is a graphical lasso with shift and extra Frobenius penalty, which means some classical algorithms for solving graphical lasso can be modified to support our formulation. In this paper, we use ADMM algorithm again because almost no extra efforts are needed compared to the conventional problem.

Introducing auxiliary variable Z, Problem 3.4 can be rewritten as

$$\begin{aligned} \min_{\Omega} \quad & -l(S_1, \Omega) + \|Z + \Delta\|_{1,\Lambda_1} + \frac{\tau}{2}\|\Omega + \Delta - T\|_F^2 \\ \text{s.t.} \quad & \Omega = Z, \end{aligned}$$

where $T = W_1 - V_1$ for simplicity.

The scaled augmented Lagrangian [2] for this problem is given by

$$\begin{aligned} L_\rho(\Omega, Z, U) = & -l(S_1, \Omega) + \|Z + \Delta\|_{1,\Lambda_1} \\ & + \frac{\tau}{2}\|\Omega + \Delta - T\|_F^2 \\ & + \frac{\rho}{2}\|\Omega - Z + U\|_F^2, \end{aligned}$$

where $\rho > 0$ is a model parameter. In each iteration, the ADMM has the form:

$$\begin{aligned} \Omega &:= \operatorname{argmin}_\Omega L_\rho(\Omega, Z, U) \\ Z &:= \operatorname{argmin}_Z L_\rho(\Omega, Z, U) \\ U &:= U + \Omega - Z \end{aligned}$$

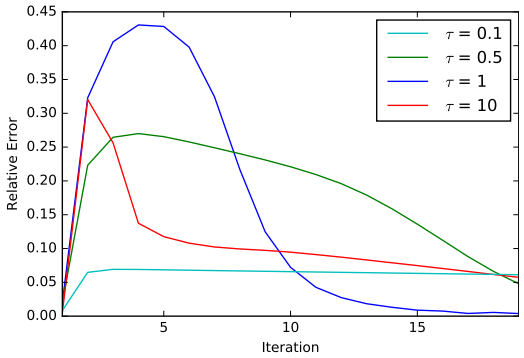


Figure 3: Relative Errors under different τ.

The analytic solution to update Ω is given by [23]:

- Compute the eigendecomposition

$$QDQ^T = \rho(Z - U) + \tau(T - \Delta) - S_1.$$

- Form diagonal matrix $\tilde{D}$ with

$$\tilde{D}_{ii} = \frac{D_{ii} + \sqrt{D_{ii}^2 - 4(\tau + \rho)}}{2(\tau + \rho)}.$$

- Let $\Omega = Q\tilde{D}Q^T$.

To update Z, we could apply soft-thresholding at each entry:

$$Z_{ij} = S_{\lambda_{ij}/\rho}(\Omega_{ij} + \Delta_{ij} + U_{ij}) - \Delta_{ij},$$

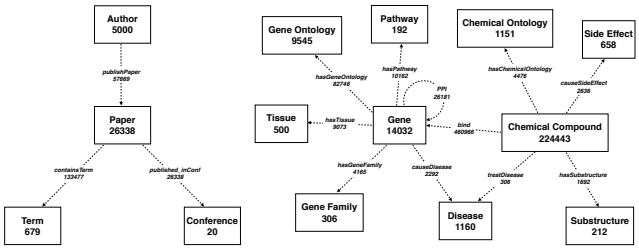
where $S_\lambda(t) = \mathbf{sgn}(t)\max\{|t| - \lambda, 0\}$.

Different from Problem 2.1, Problem 3.4 is a convex problem, which means ADMM is guaranteed to converge to the optimal value regardless of the choice of ρ, which only affects the speed of the convergence in the convex cases. However, the parameter τ of Problem 2.1 must be chosen carefully, since non-convex ADMM may not converge to the global minimum and whether it converge and the quality of approximate solution all depend on the the choice of τ [2, 7]. Figure 3 shows the relative errors under different τ's. We can observe that when τ is too large or too small, the algorithm cannot converge in a finite number of steps. In this paper, we let $\tau = 1$ in all experiments.

3.3 The Choice of $\mathbf{\Lambda}$. Though many methods for solving graphical lasso support element-wise regularization $\|\Theta\|_{1,\Lambda}$, it is still hard to determine Λ_{ij} for each pair of objects. But with the heterogeneous information, we may infer Λ_{ij} from the network, *e.g.*, any nonnegative pairwise dissimilarity metrics can be used as regularization weight. In this paper, we let

$$\Lambda_{ij}^{(k)} = \frac{\alpha}{(Sim_{ij}^{(k)})^{1/\beta}}, \tag{3.5}$$

where $k = 1, 2, \ldots, K$ and $Sim_{ij}^{(k)} \in [0, 1]$ is the PathSim [19] score between objects (i, j) under the k-th meta path. $\alpha > 0$ is a global parameter and $\beta \geqslant 1$ is used to smooth the similarity score. If two objects do not co-occur in any path group, their PathSim score will be 0, and thus $\Lambda_{ij}^{(k)} = +\infty$ forcing the $\Theta_{ij}^{(k)} = 0$, which is consistent with explicit meta path constraints. In Section 4.5, we show that the choice of β merely affects the performance of our method, so we can simply let $\beta = 1$. Now we only have one global regularization parameter α to deal with.



(a) DBLP (b) SLAP

Figure 4: The network schema

4 Experiments

4.1 Data Collection. We evaluate the proposed method on both synthetic datasets and real-world datasets. The generation method of synthetic data is as follows:

Given the number of objects n, the number of meta paths K, the number of path groups of each meta path p, and the load factor $\eta > 0$, we first generate K partial commuting matrices [19] $PC_{n\times p}$ by drawing each element $PC_{ij}^{(k)}$ from Poisson distribution $PC_{ij}^{(k)} \sim Poisson(\frac{\eta}{p})$. We call η load factor because it controls the degree of overlapping between path groups, and [11] shows that higher load factor leads to longer running time.

It is easy to see that nonzero elements in each column of partial commuting matrix belong to the same path group. Besides, we can get K similarity matrices [19] by letting commuting matrix $C^{(k)} = PC \cdot PC^T$, and then $Sim_{ij}^{(k)} = \frac{2C_{ij}^{(k)}}{C_{ii}^{(k)} + C_{ij}^{(k)}}$.

Now we generate a sparse positive definite matrix $\tilde{\Sigma}^{-1}$ as the unprocessed precision matrix, then let $\tilde{\Sigma}_{ij}^{-1} = 0$ if $Sim_{ij}^{(k)} = 0$ for all $1 \leqslant k \leqslant K$. Since zeroing out some elements may cause $\tilde{\Sigma}^{-1}$ losing definiteness, we let $\Sigma^{-1} = \tilde{\Sigma}^{-1} + \mu I$, such that the minimum eigenvalue of Σ^{-1} is 1. By sampling $x_i \sim \mathcal{N}_n(0, \Sigma), i = 1, \ldots, m$, we can get the sample covariance matrix S.

We also use 2 real-world datasets in the experiments:

• **DBLP**[1]: The first dataset is a subset of DBLP bibliographical network. We extracted 20 conferences and top-5000 authors among 4 areas[2] from 2006 to 2015. The target object is *author*. After removing the stop words in paper titles, we get 679 frequent terms as a new type of objects in the network. Meanwhile we use these terms as the vocabulary to generate bag-of-words representations as authors' activities. The schema of DBLP network is shown in Figure 4(a).

• **SLAP+AML**: The second dataset is a bioinformatic dataset SLAP [3], which is a heterogeneous network composed by over 290K nodes and 720K edges. As shown in Figure 4(b), the SLAP dataset contains integrated data related to chemical compounds, genes, diseases, side effects, pathways *etc.*. In this network, our target object is *gene* and their activities come from another two gene expression datasets of AML (acute myeloid leukemia) studies [9, 12] used in [11].

4.2 Compared Methods. In order to validate the effectiveness of our proposal, we test with following methods:

• Heterogeneous Graphical Lasso (HeteGLasso): We first test our proposed method, which takes sample covariance matrix S and the set of partial commuting matrices $PC^{(1)}, \ldots, PC^{(K)}$ as inputs, and returns K precision matrices $\Theta^{(1)} \ldots \Theta^{(K)}$. The $\Lambda^{(k)}$'s are inferred from meta paths using Equation 3.5 with the global regularization parameter α.

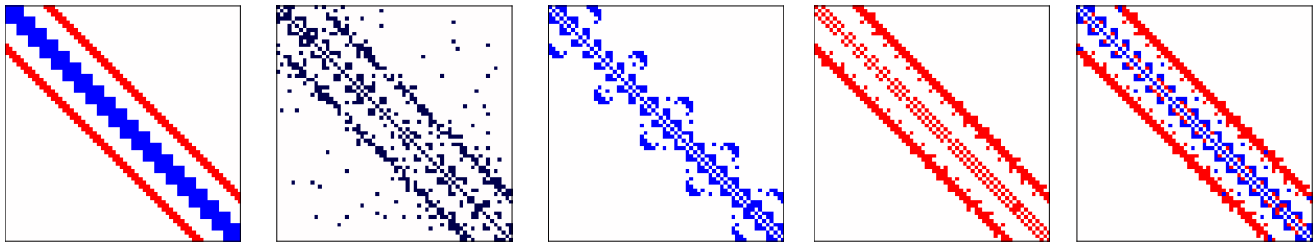
• Graphical Lasso(GLasso): This method only takes S as input, and returns a precision matrix Θ. Due to lack of meta paths information, we simply set $\Lambda_{ij} = \alpha$, where α is a global parameter.

• Pathway Graphical Lasso (PathGLasso): This method takes S and one partial commuting matrix $PC^{(k)}$ as input, and then returns $\Theta^{(k)}$ corresponding to the k-th meta path. The $\Lambda^{(k)}$ is inferred via Equation 3.5 with global parameter α. By running this method K times, we get K independent precision matrices $\Theta^{(1)} \ldots \Theta^{(K)}$.

4.3 Experiment Settings. First, we want to test whether these 3 methods can correctly recover the nonzero pattern. This can be seen as a binary classification problem if we merge all edges in $\Theta^{(1)} \ldots \Theta^{(K)}$ given by PathGLasso or HeteGLasso regardless their types. For this purpose, we follow [24] to define F1 score as an evaluation metric:

[1] http://dblp.uni-trier.de

[2] Data Mining: KDD, PKDD, ICDM, SDM, PAKDD; Database: SIGMOD Conference, VLDB, ICDE, PODS, EDBT; Information Retrieval: SIGIR, ECIR, ACL, WWW, CIKM; and Machine Learning: NIPS, ICML, ECML, AAAI, IJCAI.



(a) Ground Truth (b) GLasso (c) PathGLasso (d) HeteGLasso

Figure 5: The learned precision matrices of 3 methods. (a) Red and blue dots denote nonzero elements under different meta paths. (b) The precision matrix given by GLasso is noisy. (c) Two precision matrices given by PathGLasso correspond to different meta paths and cannot be combined due to overlap. (d) The combined precision matrix given by HeteGLasso recovers the origin matrix best.

$$\text{F1} = \frac{2 \times n_d^2}{n_a n_d + n_g n_d},$$

where n_d is the number of true edges detected by the algorithm, n_g is the number of edges in the true precision matrix, and n_a is the total number of detected edges. The larger the value, the better the performance.

Besides, we want to test whether PathGLasso and HeteGLasso can determine the edges types correctly, which can be treated this as a multi-label classification problem with $K+1$ labels by letting

$$label_{ij} = \begin{cases} \{0\} & \text{if } \Theta_{ij}^{(k)} = 0, \forall k, \\ \{k | \Theta_{ij}^{(k)} \neq 0\} & \text{otherwise.} \end{cases}$$

Similarly, we let the ground truth to be

$$label_{ij}^{\star} = \begin{cases} \{0\} & \text{if } \Sigma_{ij}^{-1} = 0, \\ \{\text{argmax}_k \, Sim_{ij}^{(k)}\} & \text{otherwise.} \end{cases}$$

Note that $label_{ij}$ given by HeteGLasso and the ground truth $label_{ij}^{\star}$ must contains only one element.

Considering not all elements facing misclassification, to avoid overestimation, we only care elements at $(i,j) \in \mathcal{X}$, where $\mathcal{X} = \{(i,j) | \, \textbf{card}(Sim_{ij}^{(\cdot)}) \geqslant 2\}$. We now have the following evaluation criteria to verify the multi-label classification performance:

• mirco-F1 [10]: is the harmonic mean of micro average of Precision and Recall.

$$\text{micro-F1} = \frac{2 \times \sum_{(i,j)\in\mathcal{X}} |label_{ij}^{\star} \cap label_{ij}|}{\sum_{(i,j)\in\mathcal{X}} |label_{ij}^{\star}| + \sum_{(i,j)\in\mathcal{X}} |label_{ij}|}.$$

The larger the value, the better the performance.

• Hamming loss [4]: evaluates the symmetric difference between true labels and predicted labels.

$$\text{HLoss} = \frac{1}{|\mathcal{X}|} \sum_{(i,j)\in\mathcal{X}} \frac{1}{K+1} |label_{ij}^{\star} \oplus label_{ij}|,$$

where $\oplus$ is the symmetric difference of two sets. The smaller the value, the better the performance.

4.4 Experiment Results. To illustrate the outputs of these 3 methods first, we manually generate a toy example with 2 meta paths and show the results in Figure 5. As we can see, GLasso gives a noisy result because of the lack of prior knowledge. PathGLasso gives two independent precision matrices but recovers more elements than we want. Meanwhile our method gives a much better result with benefit from **card**$(\cdot)$ constraints.

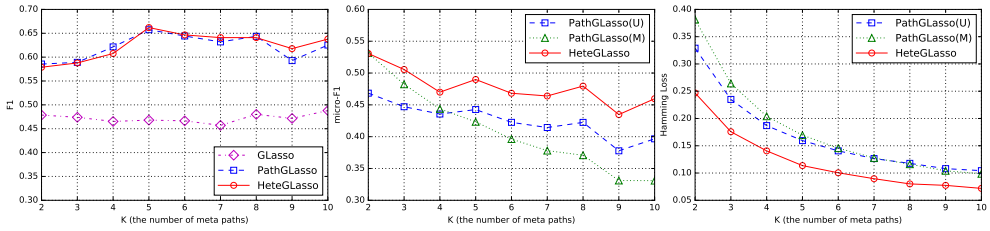
Our first experiment evaluates the ability of algorithms to recover the edges in the precision matrix. For a fair comparison, the parameter α is tuned so that the number of edges detected by algorithms is near the number of edges in the true precision matrix. The result is shown in Figure 6(a). We can observe that both PathGLasso and HeteGLasso have good and stable performance on this task, while lacking the prior knowledge results in poor performance of GLasso.

Then we study whether PathGLasso and HeteGLasso can classify the edge types correctly. Considering $label_{ij}$ given by PathGLasso may contains more than one elements, we tune α for PathGLasso in two different ways:

1. PathGLasso(Unified): The number of edges between a pair of objects (i,j) is $|label_{ij}|$. The α is tuned so that the sum of the number of edges $\sum_{ij} |label_{ij}|$ is near the number of edges in the true precision matrix times K.

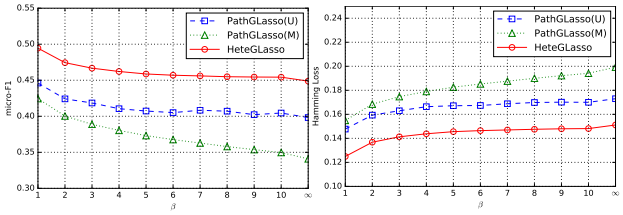
2. PathGLasso(Merged): The number of edges between a pair of objects (i,j) is 1 if $label_{ij} \neq \{0\}$, otherwise 0. The α is tuned so that the number of edges is near the number of edges in the true precision matrix.

As we can see from Figure 6(b), the micro-F1 scores of both unified and merged PathGLasso drop quickly as K increases, while HeteGLasso remains high score. As for Hamming loss in Figure 6(c), HeteGLasso is better than PathGLasso consistently.



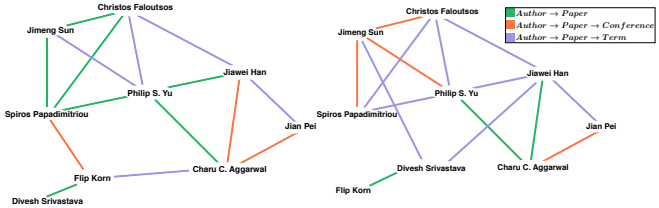
(a) F1 score ↑ (b) micro-F1 score ↑ (c) Hamming Loss ↓

Figure 6: Experimental Results($n = m = 500, \eta = 1, p = 10$). "U" and "M" in parentheses denote "Unified" and "Merged" respectively; ↑ indicates the larger the value the better the performance; ↓ indicates the smaller the value the better the performance.



(a) micro-F1 score ↑ (b) Hamming Loss ↓

Figure 7: The influence of $\beta(n = m = 500, \eta = 1, p = 10, K = 5)$.



(a) 2006 ∼ 2010 (b) 2011 ∼ 2015

Figure 8: A subgraph of DBLP identified by HeteGLasso.

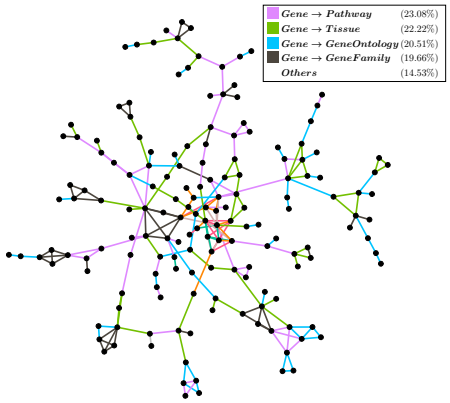


Figure 9: The largest connected component of SLAP identified by HeteGLasso

4.5 Parameter Study. We may worry that the good performance of our method on this synthetic dataset is gained via using ground truth implicitly (Equation 3.5), but Figure 7 tells us even using the same $\Lambda_{ij}^{(k)}$(obtained at $\beta = \infty$), HeteGLasso still maintains the good performance.

4.6 Case Study. We split the DBLP dataset into two periods: 2006∼2010 and 2011∼2015 and apply HeteGLasso to these two parts of dataset. Figure 8 shows a subgraph of the learned DBLP network. We can observe that the relations between authors had changed. For example, Jiawei Han and Charu C. Aggarwal were connected via $author \rightarrow paper \rightarrow conference$ because they had attended the same conferences, but recently they were connected via $author \rightarrow paper$ due to their more frequent cooperation. This means HeteGLasso successfully captured the relations between authors in different periods.

Figure 9 shows the largest connected component of the learned SLAP network, which consists of 166 nodes and 234 edges. Edges in different colors denote the different relations. It's obvious that HeteGLasso can capture the different relations among different pairs of genes.

5 Related Work

To obtain a sparse estimate of the precision matrix, numerous researchers have considered the minimum negative log-likelihood estimation using ℓ_1 regularization [1, 8, 15, 17, 18, 22], also referred to as graphical lasso. Most of these methods suffer from intensive computation (typically cubic time complexity with respect to the number of nodes). [18] and [22] proved that under certain conditions, a single graphical lasso problem can be decomposed into several smaller sized and independent problems corresponding to the non-overlapping diagonal blocks of the true precision matrix. Pathway Graphical Lasso (PathGLasso) [11] provides an efficient framework dealing with overlapping blocks if overlapping structure information is given in advance. If there is only one kind of meta path in our method, HeteGLasso degenerates into PathGLasso.

There are some prior works [6,13,14,20,24] on learning multiple precision matrices simultaneously from multiple different but related sets of observations. All these methods assume that the jointly learned precision

matrices (graphs) should share the similar structure. For example, [13] proposed a method to learn common substructures among multiple graphs. [6] used ADMM to estimate multiple precision matrices with pairwise fused lasso penalty and group lasso penalty. [24] using sequential fused penalty to encourage adjacent graphs to be similar. The biggest difference between these methods and our proposal is that we learn multiple precision matrices from only one set of observations but with meta path and exclusive constraints, which ensures that each learned graph corresponds to a certain kind of relation and is different from the others. All learned graphs can be combined into one meaningful graph with multiple types of links.

6 Conclusion

In this paper, we first incorporate heterogeneous information networks into the conventional graphical lasso framework and propose the Heterogeneous Graphical Lasso to learn multiple types of connections. We introduced the meta path constraints and cardinality constraints to ensure the unique type of relation among a pair of objects. The regularization parameter matrices in HeteGLasso are inferred from meta paths. Our proposal can be solved via non-convex ADMM, and experiments conducted on synthetic datasets demonstrated the effectiveness of HeteGLasso and the case studies showed that our method can output a meaningful result.

References

[1] O. Banerjee, L. E. Ghaoui, and A. d'Aspremont. Model selection through sparse maximum likelihood estimation for multivariate gaussian or binary data. *JMLR*, 9(Mar):485–516, 2008.

[2] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends in Machine Learning*, 3(1):1–122, 2011.

[3] B. Chen, Y. Ding, and D. J. Wild. Assessing drug target association using semantic linked data. *PLoS Comput Biol*, 8(7):e1002574, 2012.

[4] W. Cheng, E. Hüllermeier, and K. J. Dembczynski. Bayes optimal multilabel classification via probabilistic classifier chains. In *ICML*, pages 279–286, 2010.

[5] H. Chuang, E. Lee, Y. Liu, D. Lee, and T. Ideker. Network-based classification of breast cancer metastasis. *Mol Syst Biol*, 3(1):140, 2007.

[6] P. Danaher, P. Wang, and D. M. Witten. The joint graphical lasso for inverse covariance estimation across multiple classes. *J R Stat Soc Series B Stat Methodol*, 76(2):373–397, 2014.

[7] S. Diamond, R. Takapoui, and S. Boyd. A general system for heuristic solution of convex problems over nonconvex sets. *arXiv preprint arXiv:1601.07277*, 2016.

[8] J. Friedman, T. Hastie, and R. Tibshirani. Sparse inverse covariance estimation with the graphical lasso. *Biostatistics*, 9(3):432–441, 2008.

[9] A. J. Gentles, S. K. Plevritis, R. Majeti, and A. A. Alizadeh. Association of a leukemic stem cell gene expression signature with clinical outcomes in acute myeloid leukemia. *JAMA*, 304(24):2706–2715, 2010.

[10] N. Ghamrawi and A. McCallum. Collective multi-label classification. In *CIKM*, pages 195–200, 2005.

[11] M. Grechkin, M. Fazel, D. Witten, and S. Lee. Pathway graphical lasso. In *AAAI*, page 2617, 2015.

[12] T. Haferlach, A. Kohlmann, L. Wieczorek, G. Basso, G. Te Kronnie, M. Béné, J. De V, J. M. Hernández, W. Hofmann, K. I. Mills, et al. Clinical utility of microarray-based gene expression profiling in the diagnosis and subclassification of leukemia: report from the international microarray innovations in leukemia study group. *J Clin Oncol*, 28(15):2529–2537, 2010.

[13] S. Hara and T. Washio. Learning a common substructure of multiple graphical gaussian models. *Neural Networks*, 38:23–38, 2013.

[14] J. Honorio and D. Samaras. Multi-task learning of gaussian graphical models. In *ICML*, pages 447–454, 2010.

[15] C. Hsieh, M. A. Sustik, I. S. Dhillon, and P. D. Ravikumar. Quic: quadratic approximation for sparse inverse covariance estimation. *JMLR*, 15(1):2911–2947, 2014.

[16] X. Kong, P. S. Yu, Y. Ding, and D. J. Wild. Meta path-based collective classification in heterogeneous information networks. In *CIKM*, pages 1567–1571, 2012.

[17] L. Li and K. Toh. An inexact interior point method for l 1-regularized sparse covariance selection. *Math Program Comput*, 2(3-4):291–315, 2010.

[18] R. Mazumder and T. Hastie. Exact covariance thresholding into connected components for large-scale graphical lasso. *JMLR*, 13(1):781–794, 2012.

[19] Y. Sun, J. Han, X. Yan, P. S. Yu, and T. Wu. Pathsim: Meta path-based top-k similarity search in heterogeneous information networks. *PVLDB*, 4(11):992–1003, 2011.

[20] G. Varoquaux, A. Gramfort, J. Poline, and B. Thirion. Brain covariance selection: better individual functional connectivity models using population prior. In *NIPS*, pages 2334–2342, 2010.

[21] J Whittaker. *Graphical models in applied multivariate statistics*. Wiley Publishing, 2009.

[22] D. M. Witten, J. H. Friedman, and N. Simon. New insights and faster computations for the graphical lasso. *J Comput Graph Stat*, 20(4):892–900, 2011.

[23] D. M. Witten and R. Tibshirani. Covariance-regularized regression and classification for high dimensional problems. *J R Stat Soc Series B Stat Methodol*, 71(3):615–636, 2009.

[24] S. Yang, Z. Lu, X. Shen, P. Wonka, and J. Ye. Fused multiple graphical lasso. *SIAM J. Optim.*, 25(2):916–943, 2015.