Proceedings of Machine Learning Research vol 275:490–513, 2025 4th Conference on Causal Learning and Reasoning

Robust Multi-view Co-expression Network Inference

Teodora Pandeva T.P.PANDEVA@GMAIL.COM
Informatics Institute, Swammerdam Institute for Life Sciences, University of Amsterdam

Martijs Johannes Jonker M.J.JONKER@UVA.NL
Swammerdam Institute for Life Sciences, University of Amsterdam

Leendert Hamoen L.W.HAMOEN@UVA.NL
Swammerdam Institute for Life Sciences, University of Amsterdam

Joris M. Mooij J.M.MOOIJ@UVA.NL
Korteweg-De Vries Institute, University of Amsterdam

Patrick Forré P.D.FORRE@UVA.NL
Informatics Institute, University of Amsterdam

Editors: Biwei Huang and Mathias Drton

Abstract

Unraveling the co-expression of genes across studies enhances the understanding of cellular processes. Inferring gene co-expression networks from transcriptome data presents many challenges, including the high-dimensionality of the data relative to the number of samples, sample correlations, and batch effects. To address these complexities, we introduce a robust method for high-dimensional graph inference from multiple independent studies. We base our approach on the premise that each dataset is essentially a noisy linear mixture of gene loadings that follow a multivariate t-distribution with a sparse precision matrix, which is shared across studies. This allows us to show that we can identify the co-expression matrix up to a scaling factor among other model parameters. Our method employs an Expectation-Maximization procedure for parameter estimation. Empirical evaluation on synthetic and gene expression data demonstrates our method's improved ability to learn the underlying graph structure compared to baseline methods.

Keywords: co-expression network inference, high-dimensional statistics, multi-view linear independent component analysis

1. Introduction

Over the past decades, advances in DNA sequencing technologies have led to significant advances in gene regulation research. These developments have provided deep insights into biological functions and disease processes. One notable example, which we will revisit later, is the comprehensive study of the bacterium *Bacillus subtilis*. This Gram-positive bacterium serves as a model organism for studying bacterial chromosome replication and cell differentiation. A substantial research endeavor has led to a continuous manual collection of biological findings about *Bacillus subtilis* regulation and gene functionality on the online platform *SubtiWiki* (Pedreira et al., 2021), providing a clearer and more precise understanding of its cellular processes. This underscores the importance of developing methods that facilitate this process by robustly identifying such gene-gene interactions in a vast collection of experimental data from multiple sources, such as different technologies and laboratories.

Biologically relevant gene-gene interactions are often represented by a gene co-expression network (GCN), which is an undirected graph where each node corresponds to a gene. Genes that are connected or positioned closely within the GCN belong to the same functional modules, indicating that they work together to perform coordinated cellular activities. Therefore, constructing a GCN facilitates the understanding of gene regulation mechanisms. In this work, we aim to construct a GCN that closely resembles a gene regulatory network, considering only links that connect genes within the same regulatory network, such as regulator-regulated gene pairs or co-regulated genes (see Figure 1).

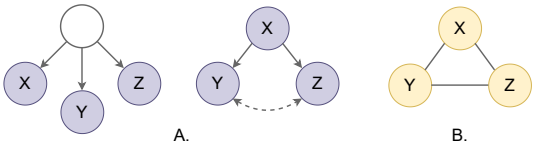

Figure 1: Two variations of the gene regulation of genes X, Y, Z (A) colored in purple and their corresponding co-expression network illustrated in (B) in yellow. In (A) (left), genes X, Y, Z are regulated by a common latent factor, such as another gene. The example in (A) (right) shows that gene X regulates both Y and Z. In addition, a bi-directional dashed line indicates potential confounding between genes Y and Z.

The analysis of transcriptome data presents several challenges. First, the number of genes typically far exceeds the number of samples ($p \gg n$), making it a high-dimensional problem. Second, experiments often have a limited number of replicates per condition, sometimes as few as two, limiting the effectiveness of causal discovery algorithms to infer gene regulatory relationships without additional prior knowledge. Third, correlations arise not only between genes but also between samples, due to overlapping experimental designs and batch effects from non-biological factors such as variations in technology or laboratory equipment. Together, these challenges complicate the inference of GCNs, not to mention the even more challenging task of inferring regulatory networks. In response, current research in gene co-expression analysis often makes specific assumptions about the data generation model to deal with this complexity. This is typically represented by a noisy decomposition model: $\mathbf{X} = \mathbf{S}A + \mathbf{E}$, where $\mathbf{X} \in \mathbb{R}^{p \times n}$ is a gene expression matrix describing the activity of p genes across n different samples (experiments, patients, tissues, etc.), $\mathbf{S} \in \mathbb{R}^{p \times k}$ is the *gene loading matrix*, A is the *sample loading matrix*, and $\mathbf{E}$ is the additive noise. A common assumption is that GCNs can be reconstructed from the gene loadings, where gene clusters are identified from each latent vector, a column in the gene loadings matrix $\mathbf{S}$ (e.g. (Moran et al., 2021; Hochreiter et al., 2010; Sastry et al., 2019)).

This paper presents a novel probabilistic method for inferring complex network structures from high-dimensional data across multiple views. Unlike traditional approaches that rely primarily on clustering techniques or Gaussian models (e.g. (Moran et al., 2021; Hochreiter et al., 2010; Gao et al., 2016; Kim and Park, 2007; Danaher et al., 2014; Guo et al., 2011)), our method employs a matrix-variate t-distribution framework that extends TLASSO by Finegold and Drton (2011). As pointed out by Finegold and Drton (2011) the t-distribution provides a more robust modelling approach under the assumption that the data is contaminated, what we often observe in practice. We refer to our model as MVTLASSO, which captures the covariance at both the sample and variable levels in the multi-view setting. Key contributions of this work, besides the proposed model, include the formulation of identifiability guarantees for the model parameters, such as the sparse precision matrix, which we can identify up to a scalar multiple (see Section 2.2). For model estimation, we implement an Expectation-Maximization (EM) procedure, which is described in Section 3. We apply MVTLASSO to both synthetic datasets and real-world gene expression data to validate its

effectiveness. Our empirical results in Section 4 show that MVTLASSO consistently demonstrates improved accuracy in reconstructing the underlying graph structures compared to baseline methods.

2. Robust Co-Expression Inference from non-i.i.d Samples

In this section, we introduce and justify our chosen generative model, which we will refer to as **MVTLASSO**, placing it within the broader context of known GCN inference methods. In Section 2.2, we present theoretical guarantees for recovering the true model parameters.

Our approach can be seen as an instance of ICA, where the latent components, or gene loadings, are divided into two categories: those used to construct the GCN, denoted by $\mathbf{S}$, and those considered noise, denoted by $\mathbf{Z}$, which do not contribute to the GCN inference. We infer the GCN from the sparse precision matrix Θ estimated from *all* "useful" gene loadings $\mathbf{S}$ across datasets (or *views*) that follow a multivariate t-distribution similar to (Finegold and Drton, 2011). More specifically, we make the following assumptions regarding the data generation process:

Definition 1 *Consider the scenario where we are given D different data sets $\mathbf{X}_d \in \mathbb{R}^{p\times n_d}$, which may come from different sources and follow the representation:*

$$\mathbf{X}_d = \mathbf{S}_d A_d + \mathbf{Z}_d B_d,$$

where for each $d = 1, \dots, D$, it holds:

1. *$(A_d^\top | B_d^\top)^\top \in \mathbb{R}^{(k_d+r_d)\times n_d}$ have full row rank with*

$$\operatorname{rank}(A_d) = k_d \quad \textit{and} \quad \operatorname{rank}(B_d) = n_d - k_d =: r_d,$$

2. *the columns of $\mathbf{S}_d \in \mathbb{R}^{p\times k_d}$ are mutually independent and follow a multivariate t-distribution, i.e. $\mathbf{S}_{d_{:,i}} \sim t_p(\nu, \mu_d, \Sigma)$ with $\nu > 2$ degrees of freedom and a sparse inverse dispersion matrix $\Theta := (\Sigma)^{-1}$ that has a prior distribution $p_\lambda(\Theta)$ with $\lambda > 0$ defined as*

$$p_\lambda(\Theta) \propto \exp\left(-\lambda\|\Theta\|_1\right) \quad \textit{with} \quad \|\Theta\|_1 = \sum_{i,j} |\Theta_{ij}|,$$

3. *the columns of $\mathbf{Z}_d \in \mathbb{R}^{p\times r_d}$ are noise random variables and are i.i.d multivariate t-distributed $t_p(\nu, 0, \sigma_d^2 \mathbb{I}_p)$, such that there is no $\lambda \in \mathbb{R}$ with $\sigma_d^2 \mathbb{I}_p = \lambda\Sigma$,*

4. *the latents $\mathbf{S}_d$ and noise matrix $\mathbf{Z}_d$ are conditionally independent given Θ.*

This perspective on Θ as a representation of the GCN closely aligns our work with that proposed by Stegle et al. (2011) for the single view case. Compared to (Stegle et al., 2011), we shift from a multivariate normal distribution to a multivariate t-distribution with sparse Θ. Although this moves away from the theoretical guarantees of conditional independence to a more relaxed condition of conditional uncorrelation, as outlined by Finegold and Drton (2011) and Section 2.1, this approach provides more robust inference for unknown parameters, in this case, A_d, B_d, μ_d, Θ. This robustness is particularly beneficial in the presence of data contamination, a common challenge in the analysis of transcriptome data. Additionally, compared to TLASSO by Finegold and Drton (2011), three major differences are that MVTLASSO models the correlation between the samples via the mixing

matrix A_d, it is designed to simultaneously model multiple views and assumes that some gene loadings represent noise and therefore do not contribute to the estimation of the GCN matrix.

Finally, our model is reminiscent of dimensionality reduction methods, similar to the application of principle component analysis (PCA), aiming to identify k_d components per dataset (or view) that capture the most significant signals from the data. The remaining components are considered as i.i.d. noise, following a multivariate t-distribution, which does not play a role in estimating the network structure, represented by Θ. A similar decomposition is proposed by Parsana et al. (2019), where the authors show that removing the noise components after applying PCA improves the GCN inference of several algorithms.

2.1. Dependence Relationship between the Genes in the GCN

The GCN is inferred from Θ as follows. Consider a graph $G = (V, E)$, where $V = \{1, \ldots, p\}$ represents the set of observed genes and E is a collection of edges between pair of nodes (or genes) i and j for which the corresponding entry Θ_{ij} is non-zero.

An interesting aspect is understanding the types of (in)dependencies encoded by this graph structure. For context, in Gaussian models, the absence of an edge between two nodes i and j implies conditional independence between them, given the remaining nodes. However, this direct implication does not translate to multivariate t-distributions. Instead, a weaker concept of dependence, conditional uncorrelation, applies, as discussed in (Finegold and Drton, 2011):

Theorem 1 ((Finegold and Drton, 2011)) *Let $S \sim t_p(\nu, \mu, \Sigma)$, where Σ is a positive definite matrix with $(\Sigma^{-1})_{ij} = 0$ for indices $i \neq j$ corresponding to non-edges in the graph G. If two nodes i and j are separated by a set of nodes C in G, then S_i and S_j are conditionally uncorrelated given S_C.*

While Theorem 1 shows that conditional uncorrelation can be derived from the graph structure, it leaves open the question of whether multivariate t-distributions can be factorized according to any Bayesian network. The following result addresses this issue by showing that the only Bayesian network compatible with the multivariate t-distribution is a fully connected DAG:

Lemma 1 *Let $G = (V, E)$ be a DAG with vertices $V = \{1, \ldots, p\}$. Furthermore, the joint distribution of the corresponding variables $S_1, \ldots, S_p$ is multivariate t-distribution $t_p(\nu, \mu, \Sigma)$ with $0 < \nu < \infty$. Let $\mathrm{pa}(k) \subseteq V \setminus \{k\}$ denote the set of parents of node k. Then, the following holds $P(S_1, \ldots, S_p) = \prod_{k=1}^{p} P(S_k \mid S_{\mathrm{pa}(k)})$ iff there exists an ordering $S_{\tau(1)}, \ldots, S_{\tau(p)}$ such that $\mathrm{pa}(\tau(k)) = \{\tau(1), \ldots, \tau(k-1)\}$, i.e. the graph is fully connected.*

Remark 1

(a) Lemma 1 suggests that from the estimated Θ, we can infer only conditional uncorrelation between the genes, not conditional independence. However, this result does not contradict the GCN notion used in this work, as detailed in Section 1, which is based on correlation rather than statistical independence.

(b) According to Theorem 1, the reconstructed GCN should exclude edges between genes that are conditionally uncorrelated given the rest of the genes. This implies that co-regulated genes will not be connected in the GCN, as they become conditionally uncorrelated when conditioned

on their regulators. However, in practice, this assumption is often violated due to specific limitations in the data, e.g. lack of examples with temporal information.

Temporal information plays a crucial role in inferring gene regulatory relationships from observed data. Consider an example with three genes: X, Y and Z. At time t the gene X becomes active and we observe its expression as X_t. At the subsequent time point $t+1$, changes in the expression of genes Y and Z are detected and recorded as Y_{t+1} and Z_{t+1} respectively. This sequential pattern suggests a regulatory relationship, with X_t potentially affecting Y_{t+1} and Z_{t+1}. In other words, the gene X probably regulates Y and Z.

However, in many experimental datasets there is limited availability of time series data, which limits the ability of machine learning algorithms to clearly infer sequential dependencies. For example, without observing gene profiles at time t but only at $t+1$, Y_{t+1} and Z_{t+1} appear to be confounded by the hidden variable X_t. Thus, Y and Z may appear to be directly related in GCN, even though X may be their regulator.

Yin et al. (2021) empirically demonstrated that without accounting for temporal information, regulators like X may appear uncorrelated with their targets, causing regulatory connections to be absent from the GCN.

2.2. Identifiability Guarantees

Next, we will present our theoretical guarantees for identifying model parameters from Definition 1, i.e $\{A_d, B_d, \mu_d, \sigma_d^2\}$, $d = 1, \ldots, D$, and Σ. We will show that the location μ_d and dispersion matrix Σ of the gene loadings, as well as the sample loadings A_d are identifiable up to the same constant across all views:

Proposition 1 *Let $\mathbf{X}_1, \ldots, \mathbf{X}_D$ with $\mathbf{X}_d \in \mathbb{R}^{p \times n_d}$ be random matrices with the following two representations:*

$$\mathbf{S}_d^{(1)} A_d^{(1)} + \mathbf{Z}_d^{(1)} B_d^{(1)} = \mathbf{X}_d = \mathbf{S}_d^{(2)} A_d^{(2)} + \mathbf{Z}_d^{(2)} B_d^{(2)},$$

where for $d = 1, \ldots, D$, both representations, $j \in \{1, 2\}$, satisfy:

$$A_d^{(j)} \in \mathbb{R}^{k_d^{(j)} \times n_d}, \qquad B_d^{(j)} \in \mathbb{R}^{(n_d - k_d^{(j)}) \times n_d}, \qquad \mathbf{S}_d^{(j)} \in \mathbb{R}^{p \times k_d^{(j)}}, \qquad \mathbf{Z}_d^{(j)} \in \mathbb{R}^{p \times (n_d - k_d^{(j)})},$$

and the properties of Definition 1. Then, for $d = 1, \ldots, D$, we have $k_d^{(1)} = k_d^{(2)} =: k_d$. Furthermore, there exist permutation matrices $P_{A_1}, \ldots, P_{A_D}, P_{B_1}, \ldots, P_{B_D}$ and constants $c, c_1, \ldots, c_D > 0$ such that:

$$A_d^{(2)} = c P_{A_d} A_d^{(1)}, \qquad \Sigma_S^{(2)} = \frac{\Sigma_S^{(1)}}{c^2}, \qquad \mu_{S_d}^{(2)} = \frac{\mu_{S_d}^{(1)}}{c},$$

$$B_d^{(2)} = c_d P_{B_d} B_d^{(1)}, \qquad \Sigma_{Z_d}^{(2)} = \frac{\Sigma_{Z_d}^{(1)}}{c_d^2}, \qquad \mu_{Z_d}^{(2)} = \frac{\mu_{Z_d}^{(1)}}{c_d}.$$

In contrast to well-established results in the ICA literature (Comon, 1994; Kagan et al., 1973), which provide identifiability for the univariate case, we extend these results to multivariate elliptic distributions, as shown in Corollary 1. Proposition 1 is a special case and a direct consequence of

our more general results. Unlike the single-view case, the multi-view setting allows us to achieve identifiability of the sample matrices A_d across views, up to a common scaling factor c. Furthermore, assumption (3) in Definition 1 is essential for distinguishing between noise and gene loadings; if no noise components are present, this assumption becomes unnecessary.

3. Parameter Estimation

We begin by deriving the data likelihood, drawing inspiration from the ICA literature, e.g. the works of (Hyvarinen, 1999; Amari et al., 1995). Instead of making derivations with respect to A_d and B_d, we proceed in terms of the inverse of the concatenated matrix, denoted as $W_d = (A_d \mid B_d)^{-1}$. Consequently, the "unmixed" signal $\mathbf{Y}_d := \mathbf{X}_d W_d$ represents the estimates for the latent vectors $\mathbf{S}_{d_{:,i}}$ for $i = 1, \ldots, k_d$, and $\mathbf{Z}_{d_{:,i}}$ for $i = 1, \ldots, n_d - k_d$, up to some scaling and permutation as described in Proposition 1[1]. These signals follow a multivariate t-distribution. Thus, the likelihood for all views $\mathbf{X}_1, \ldots, \mathbf{X}_D$ is:

$$
\begin{aligned}
p(\mathbf{X}_1, \ldots, \mathbf{X}_D \mid \{ W_d, \mu_d, \sigma_d \}_{d=1}^D, \Sigma) &= \prod_{d=1}^{D} p(\mathbf{X}_d) = \prod_{d=1}^{D} |\det W_d| \, p(\mathbf{X}_d \cdot W_d) \\
&= \prod_{d=1}^{D} |\det W_d| \prod_{i=1}^{n_d} t_p \left(\mathbf{Y}_{d_{:,i}} \mid \nu, \rho_{d_i}, \Phi_{d_i} \right),
\end{aligned} \tag{1}
$$

where $\Phi_{d_i} = \mathbb{1}_{\{i \le k_d\}} \Sigma + \mathbb{1}_{\{i > k_d\}} \sigma_d^2 \mathbb{I}_p$ and $\rho_{d_i} = \mathbb{1}_{\{i \le k_d\}} \mu_d$. Thus, the data likelihood is proportional to the product of the probabilities of $\sum_{d=1}^{D} n_d$ independent multivariate t-distributed vectors.

3.1. The Expectation-Maximization Procedure

Unfortunately, directly estimating the unknown parameters from (1) is infeasible. However, we can leverage the alternative representation of the multivariate t-distribution described in Theorem 4, which is central to the EM procedure proposed by Liu and Rubin (1995); Finegold and Drton (2011). For each random vector $\mathbf{Y}_{d_{:,i}}$, the generative process can equivalently be represented as:

$$
\tau_{d_i} \sim \Gamma\left(\frac{\nu}{2}, \frac{\nu}{2}\right), \qquad\qquad \mathbf{Y}_{d_{:,i}} \sim \mathcal{N}(\rho_{d_i}, \Phi_{d_i} / \tau_{d_i}),
$$

where the variables τ_{d_i} are unobserved. Thus, the complete data log-likelihood with unknown parameters $\boldsymbol{\gamma} := \{W_d, \mu_d, \sigma_d\}_{d=1}^D \cup \{\Sigma\}$ and random variables $\mathbf{X}_1, \ldots, \mathbf{X}_D$ and $\tau_1, \ldots \tau_D$ with $\tau_d := (\tau_{d_1}, \ldots, \tau_{d_{n_d}})$ is (up to additive constants) given by:

$$
\begin{aligned}
l\left(\boldsymbol{\gamma};\ \{\mathbf{X}_d, \tau_d\}_{d=1}^D\right) \overset{+}{\propto} \sum_d \Bigg\{ & \ln |\det W_d| + \sum_{i=1}^{n_d} \frac{1}{2} \ln \det \Phi_{d_i}^{-1} - \frac{\tau_{d_i}}{2} \operatorname{tr}\left(\Phi_{d_i}^{-1} \mathbf{Y}_{d_{:,i}} \mathbf{Y}_{d_{:,i}}^\top \right) \\
& + \tau_{d_i} \rho_{d_i}^\top \Phi_{d_i}^{-1} \mathbf{Y}_{d_{:,i}} - \frac{\tau_{d_i}}{2} \rho_{d_i}^\top \Phi_{d_i}^{-1} \rho_{d_i} \Bigg\},
\end{aligned} \tag{2}
$$

1. Specifically, the first k_d columns of $\mathbf{Y}_d$ correspond to the estimates of $\mathbf{S}_d$, while the remaining $n - k_d$ columns correspond to the estimates of $\mathbf{Z}_d$.

where $\Phi_{d_i}^{-1} = \mathbb{1}_{\{i \le k_d\}}\Theta + \mathbb{1}_{\{i > k_d\}}\frac{1}{\sigma_d^2}\mathbb{I}_p$ with $\Theta = (\Sigma)^{-1}$. The right side of (2) is linear in the latent variables τ_{d_i}. Thus, for the E-step it suffices to compute $\mathbb{E}[\tau_{d_i} \mid \mathbf{X}_d]$ for every $d = 1, \ldots, D$ and $i = 1, \ldots, n_d$. This can be derived directly by observing that the conditional distribution $p(\tau_{d_i} \mid \mathbf{X}_d) = p(\tau_{d_i} \mid \mathbf{Y}_{d_{:,i}})$ is given by

$$\tau_{d_i} \mid \mathbf{Y}_{d_{:,i}} \sim \Gamma\left(\frac{\nu + p}{2}, \frac{\nu + \delta(\mathbf{Y}_{d_{:,i}}, \rho_{d_i}, \Phi_{d_i})}{2}\right)$$

with

$$\delta(\mathbf{Y}_{d_{:,i}}, \rho_{d_i}, \Phi_{d_i}) = (\mathbf{Y}_{d_{:,i}} - \rho_{d_i})^\top \Phi_{d_i}^{-1}(\mathbf{Y}_{d_{:,i}} - \rho_{d_i}).$$

Consequently, for the conditional expectation we get: $\mathbb{E}[\tau_{d_i} \mid \mathbf{Y}_{d_{:,i}}] = \dfrac{\nu + p}{\nu + \delta(\mathbf{Y}_{d_{:,i}}, \rho_{d_i}, \Phi_{d_i})}$.

Hence, the EM procedure iterates through two main steps for each view d: 1) the estimation of τ_{d_i} while keeping ρ_{d_i}, Φ_{d_i}, and W_d fixed; and 2) the estimation of ρ_{d_i}, Φ_{d_i}, W_d, and $\Theta := (\Sigma)^{-1}$, where Θ is determined by solving the graphical lasso (GLASSO) problem as described by Friedman et al. (2008). This method is designed to estimate sparse precision matrices in a multi-view setting. The EM procedure at step $t \ge 1$ is performed as follows:

E-step: For fixed estimated $\mu_d^{(t-1)}, \Sigma^{(t-1)}, \sigma_d^{(t-1)}$ and $W_d^{(t-1)}$ compute $\mathbb{E}[\tau_{d_i} \mid \mathbf{X}_d]$, i.e.

$$\mathbf{Y}_d^{(t-1)} = \mathbf{X}_d W_d^{(t-1)}, \qquad \tau_{d_i}^{(t)} = \frac{\nu + p}{\nu + \delta(\mathbf{Y}_{d_{:,i}}^{(t-1)}, \rho_{d_i}^{(t-1)}, \Phi_{d_i}^{(t-1)})}.$$

M-step: Solve the optimization problem:

$$\boldsymbol{\gamma}^{(t)} \in \arg\max_{\boldsymbol{\gamma}} l\left(\boldsymbol{\gamma};\ \{\mathbf{X}_d, \tau_d^{(t)}\}_{d=1}^D\right),$$

with $\boldsymbol{\gamma}^{(t)} = \{W_d^{(t)}, \mu_d^{(t)}, \sigma_d^{(t)}\}_{d=1}^D \cup \{\Sigma^{(t)}\}$ that leads to the following steps for all $d = 1, \ldots, D$:

1. Calculate $\mu_d^{(t)}, \Sigma^{(t)}$ and $\sigma_d^{(t)}$ for fixed $\tau_{d_i}^{(t)}$ and $\mathbf{Y}_d^{(t)}$

$$\mu_d^{(t)} = \frac{\sum_{i=1}^{k_d} \tau_{d_i}^{(t)} \mathbf{Y}_{d_{:,i}}^{(t-1)}}{\sum_{i=1}^{k_d} \tau_{d_i}^{(t)}}, \quad \Sigma^{(t)} = \frac{1}{\sum_d k_d} \sum_d \sum_{i=1}^{k_d} \tau_{d_i}^{(t)} \left(\mathbf{Y}_{d_{:,i}}^{(t-1)} - \mu_d^{(t)}\right)\left(\mathbf{Y}_{d_{:,i}}^{(t-1)} - \mu_d^{(t)}\right)^\top,$$

$$\sigma_d^{(t)} = \sqrt{\frac{1}{p(n - k_d)} \sum_{i=k_d+1}^{n} \sum_{l=1}^{p} \tau_{d_i}^{(t)} (\mathbf{Y}_{d_{l,i}}^{(t-1)})^2}$$

2. Estimate Θ via solving the GLASSO optimization problem for $\Sigma^{(t)}$ with penalty parameter $\lambda > 0$ given by:

$$\Theta^{(t)} \in \arg\min_{\Theta \succ 0} -\ln\det(\Theta) + \operatorname{tr}(\Sigma^{(t)}\Theta) + \lambda\|\Theta\|_1 \tag{3}$$

3. Estimate $W_d^{(t)}$ for fixed $\mu_d^{(t)}, \Sigma^{(t)}, \sigma_d^{(t)}$ and $\tau_{d_i}^{(t)}$:

$$W_d^{(t)} \in \arg\min_W \Big\{ \operatorname{tr}\left(\left(\mathbf{X}_d W - \boldsymbol{\mu}_d^{(t)} \right)^\top \Theta^{(t)} \left(\mathbf{X}_d W - \boldsymbol{\mu}_d^{(t)} \right) \mathcal{T}_1^{(t)} \right) + \frac{1}{(\sigma_d^{(t)})^2} \operatorname{tr}\left(W^\top \mathbf{X}_d^\top \mathbf{X}_d W \mathcal{T}_2^{(t)} \right) - \ln|\det W| \Big\}, \tag{4}$$

where $\boldsymbol{\mu}_d^{(t)} := (\underbrace{\mu_d^{(t)}, \ldots, \mu_d^{(t)}}_{k_d}, 0 \ldots, 0) \in \mathbb{R}^{p \times n_d}$, and $\mathcal{T}_1^{(t)}, \mathcal{T}_2^{(t)} \in \mathbb{R}^{n_d \times n_d}$ are diagonal matrices defined as $\mathcal{T}_1^{(t)} = \operatorname{diag}(\tau_{d_1}^{(t)}, \ldots, \tau_{d_{k_d}}^{(t)}, 0, \ldots, 0)$ and $\mathcal{T}_2^{(t)} = \operatorname{diag}(0, \ldots, 0, \tau_{d_{k_d+1}}^{(t)}, \ldots, \tau_{d_{n_d}}^{(t)})$

Details on the implementation of the EM procedure can be found in Appendix E.1.

4. Results

4.1. Simulated Data

We benchmark our method, **MVTLASSO**, against **GLASSO** (Friedman et al., 2008) and **TLASSO** (Finegold and Drton, 2011) using a series of synthetic experiments. For each method, we run 100 independent experiments across various sparsity parameters, λ. By computing the average true positive and false positive rates across these experiments, we evaluate how well each method reconstructs the ground truth precision matrix, Θ, used in generating the synthetic data.

The simulated data follows the generative model defined in Definition 1 and mirrors the setup proposed by Finegold and Drton (2011). In particular, the sparse precision matrix Θ is constructed as follows: 1) off-diagonal entries Θ_{ij} with $i \neq j$ are sampled from $\{-1, 0, 1\}$ with probabilities $\{0.01, 0.98, 0.01\}$ 2) the diagonal entries are set to 1 plus the number of edges connected with the node, i.e. $\Theta_{ii} = 1 + \sum_j \mathbb{1}_{\{\Theta_{ij} \neq 0\}}$. Additionally, we set $\mu = 0$ and $\sigma = 1$ in all experiments. The sample loading matrices A and B have entries sampled according to standard normal distribution. The dimension of Θ is fixed at 200×200 for all experiments.

Figure 2 presents the results from three major experiments, each displayed in a separate panel corresponding to different total numbers of sources per view (i.e., the sum of gene loadings and noise sources): 20 (A), 40 (B), and 60 (C). Within each panel, nine figures are organized by the number of gene loadings k and the number of views D: each row represents a different value for $D \in \{2, 5, 10\}$ and each column corresponds to a specific k chosen to represent 50%, 75%, and 100% of the total sources. For example, in panel A, where the total number of sources is 20, the columns represent $k = 10, 15$ or 20.

From the results in Figure 2, we observe that: 1) **MVTLASSO**'s performance significantly improves with an increasing number of views, while the other two methods show only moderate improvements. 2) All methods perform better when the number of gene loadings is higher relative to the number of noise sources. 3) Most of the time, **MVTLASSO** achieves a more favorable true positive to false positive ratio compared to the other baselines.

4.2. Gene Co-Expression Inference

We revisit the motivational example of the GCN inference from *B. Subtilis* gene expression data. For this purpose, we use two well-controlled transcriptome data compendia. These datasets were

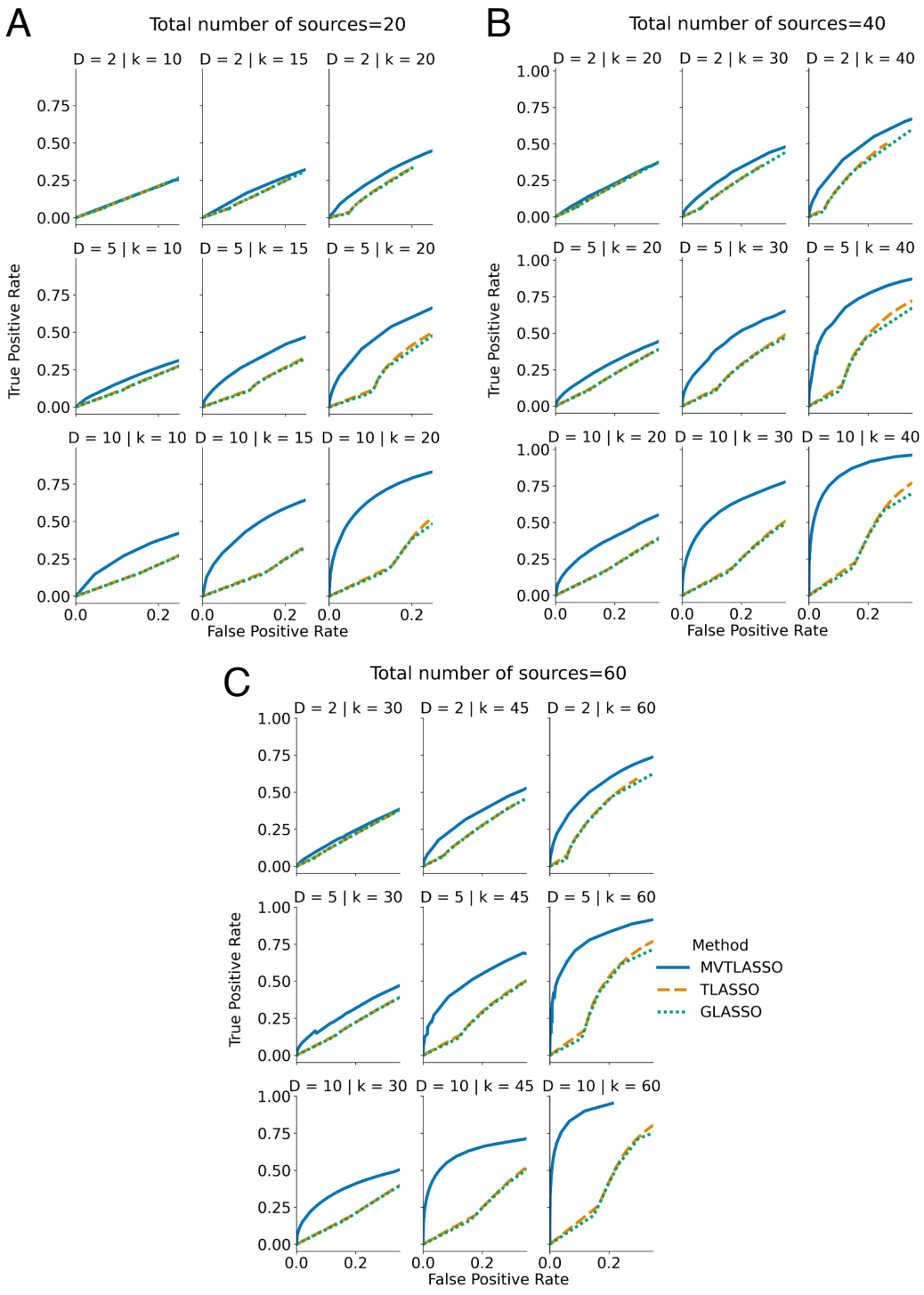

Figure 2: True positive rate vs false positive rate for total number of sources A) 20, B) 40, C) 60. Columns indicate the number of gene loadings k and rows the number of views D. In all cases MVTLASSO manages to retrieve the ground truth precision matrix better if not equally good as the baseline methods.

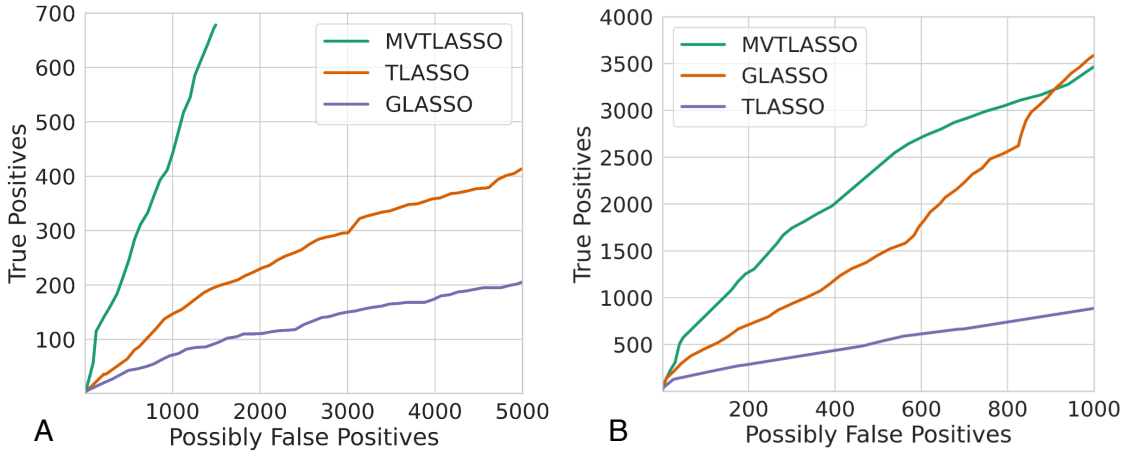

Figure 3: True positive vs. possibly false positive edges obtained via stability selection for *S. aureus* (A) and *B. subtilis* (B). The results suggest that MVTLASSO tends to infer more true positive edges across different settings.

collected using the *B. subtilis* strain BSB1, which contains 269 samples from 104 different experimental conditions (Nicolas et al., 2012), and the closely related strain PY79, which contains data from 38 unique experimental designs (Arrieta-Ortiz et al., 2015). The *B. Subtilis* genome contains approximately 4100 genes, and for every transcriptome experiment, gene expressions from 3994 genes were obtained. Both datasets include a wide range of conditions, including growth in different media, competence, biofilm formation, swarming, different stress conditions, sporulation, and knockout experiments. The data were preprocessed as outlined in Appendix E.2. We further split each dataset into three approximately equal subsets of samples, ensuring they are as distinct as possible in their experimental design.

We then used these six views to benchmark **MVTLASSO** against two other methods: **TLASSO** and **GLASSO+Standardization**, as described in Appendix E.1. Since the correct number of gene loadings k_d remains unknown, we set $k_d = 1$ in this study. The development of a more sophisticated method for determining k_d is left for future research.

The fitting process for all methods incorporates stability selection, as outlined by Meinshausen and Bühlmann (2010) and detailed in Appendix E.1. In this approach, for each penalty parameter λ, "stable" edges are selected from 100 precision matrices, each estimated from bootstrapped samples containing 90% of the data. An edge is considered stable if its stability score exceeds 0.5, meaning that the edge appears in more than 50% of the 100 precision matrices. We then aggregate the stable edges across all sparsity parameters by summing their stability scores across all experiments.

To evaluate the model's performance, we rank the edges of the weighted undirected graph according to their weights. We then count the number of true positive edges, as verified against the ground truth data from *SubtiWiki*, as well as potential false positives in the top 100, 200, 300, ... edges. The counts of true positive versus false positive edges for each method are shown in Figure 3(B). These results indicate that **MVTLASSO** consistently identifies more true positive edges across most penalty parameters compared to the other two methods.

We apply this process to gene expression data from the bacterium *Staphylococcus aureus*, a species commonly found on the skin and in the nose that can cause a range of infections. This dataset

contains gene expression profiles for $p = 2810$ genes measured under 53 different conditions, yielding a total of $n = 160$ samples. The gene expression data, provided by the DREAM challenge (Marbach et al., 2012), is accompanied by a ground truth network comprising $18,208$ edges. In this analysis, we treat the data as a single-view case ($D = 1$) due to the limited background information regarding the experimental design.

We estimate the gene loadings k_1 following the approach in (Parsana et al., 2019) prior to fitting the models. For this dataset, we use a **GLASSO** method based on the partial correlation matrix (similar to (Carter et al., 2024)), which enhances the performance of all methods. We then compare the performance of **MVTLASSO** with the same baselines as in the previous case with respect to the potential true positive and false positive edges. As before, we apply stability selection to estimate the stable edges from 100 independent runs for each sparsity parameter and subsequently aggregate all stable edges into a single weighted undirected graph. Figure 3(A) illustrates that for sparse gene co-expression networks (GCNs), **MVTLASSO** identifies more true positive edges compared to the baselines.

5. Discussion

We introduced MVTLASSO, a robust method for inferring gene co-expression networks from high-dimensional gene expression data across multiple independent studies. Our approach effectively addresses the inherent complexity of gene expression data, including gene and sample correlations as well as batch effects, by modeling each dataset as a noisy linear mixture of gene loadings governed by a multivariate t-distribution with a sparse precision matrix. We employ an EM procedure for parameter estimation, supported by theoretical guarantees that ensure the identifiability of the model parameters. Empirical evaluations on both synthetic and real gene expression data have demonstrated the superior performance of MVTLASSO compared to baseline methods. Our method consistently shows improved accuracy in learning the underlying graph structures, underscoring its robustness and reliability.

Thus, our model can be interpreted as an instance of independent component analysis, where the sources follow a multivariate t-distribution with an identifiable sparse precision matrix. Although our model is restricted to inferring only the conditionally uncorrelated relationships, we believe that it, along with the baseline **TLASSO**, more accurately captures the true data distribution. In contrast, the **GLASSO** method, which is based on the Gaussian model and is theoretically designed to infer conditional independence between genes, often faces practical challenges due to confounding factors present in the data (see (Parsana et al., 2019)). Consequently, in practice, GLASSO is primarily employed for inferring gene co-expression networks rather than direct gene regulation, e.g. see (Petralia et al., 2018; Lyu et al., 2018; Seal et al., 2023).

A promising direction for future work is to develop a more efficient and reliable hyperparameter selection procedure. The selection of sample dimensions and noise loadings can be challenging and time-consuming due to the implemented EM procedure. In addition, incorporating available experimental metadata into the modeling process could provide further refinement and improve the overall performance of MVTLASSO.

References

Shun-ichi Amari, Andrzej Cichocki, and Howard Yang. A New Learning Algorithm for Blind Signal Separation. *Advances in Neural Information Processing Systems*, 8, 1995.

Reinaldo B Arellano-Valle and Heleno Bolfarine. On some characterizations of the t-distribution. *Statistics & Probability Letters*, 25(1):79–85, 1995. doi: 10.1016/0167-7152(94)00208-P.

Mario L Arrieta-Ortiz, Christoph Hafemeister, Ashley Rose Bate, Timothy Chu, Alex Greenfield, Bentley Shuster, Samantha N Barry, Matthew Gallitto, Brian Liu, Thadeous Kacmarczyk, et al. An experimentally supported model of the *Bacillus subtilis* global transcriptional regulatory network. *Molecular systems biology*, 11(11):839, 2015. doi: 10.15252/msb.20156236.

Jack Storror Carter, David Rossell, and Jim Q Smith. Partial correlation graphical lasso. *Scandinavian Journal of Statistics*, 51(1):32–63, 2024. doi: 10.1111/sjos.12675.

Pierre Comon. Independent component analysis, a new concept? *Signal processing*, 36(3):287–314, 1994. doi: 10.1016/0165-1684(94)90029-9.

Patrick Danaher, Pei Wang, and Daniela M Witten. The joint graphical lasso for inverse covariance estimation across multiple classes. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 76(2):373–397, 2014. doi: 10.1111/rssb.12033.

Kai Wang Fang, Samuel Kotz, and Kai Wang Ng. *Symmetric Multivariate and Related Distributions*. Chapman and Hall/CRC, 2018. doi: 10.1201/9781351077040.

Michael Finegold and Mathias Drton. Robust graphical modeling of gene networks using classical and alternative t-distributions. *The Annals of Applied Statistics*, 5(2A):1057–1080, 2011. doi: 10.1214/10-AOAS410.

Jerome Friedman, Trevor Hastie, and Robert Tibshirani. Sparse inverse covariance estimation with the graphical lasso. *Biostatistics*, 9(3):432–441, 2008. doi: 10.1093/biostatistics/kxm045.

Jerome Friedman, Trevor Hastie, and Rob Tibshirani. *glasso: Graphical Lasso: Estimation of Gaussian Graphical Models*, 2019. URL https://CRAN.R-project.org/package=glasso. R package version 1.11.

Chuan Gao, Ian C McDowell, Shiwen Zhao, Christopher D Brown, and Barbara E Engelhardt. Context Specific and Differential Gene Co-expression Networks via Bayesian Biclustering. *PLoS computational biology*, 12(7):e1004791, 2016. doi: 10.1371/journal.pcbi.1004791.

Jian Guo, Elizaveta Levina, George Michailidis, and Ji Zhu. Joint estimation of multiple graphical models. *Biometrika*, 98(1):1–15, 2011. doi: 10.1093/biomet/asq060.

Sepp Hochreiter, Ulrich Bodenhofer, Martin Heusel, Andreas Mayr, Andreas Mitterecker, Adetayo Kasim, Tatsiana Khamiakova, Suzy Van Sanden, Dan Lin, Willem Talloen, et al. FABIA: factor analysis for bicluster acquisition. *Bioinformatics*, 26(12):1520–1527, 2010. doi: 10.1093/bioinformatics/btq227.

Aapo Hyvarinen. Fast and robust fixed-point algorithms for independent component analysis. *IEEE transactions on Neural Networks*, 10(3):626–634, 1999. doi: 10.1109/72.761722.

Abram Meerovich Kagan, Yurii Vladimirovich Linnik, Calyampudi Radhakrishna Rao, et al. *Characterization problems in mathematical statistics*. Wiley-Interscience, 1973. doi: 10.2307/2345228.

Hyunsoo Kim and Haesun Park. Sparse non-negative matrix factorizations via alternating non-negativity-constrained least squares for microarray data analysis. *Bioinformatics*, 23(12):1495–1502, 2007. doi: 10.1093/bioinformatics/btm134.

SungHwan Kim, Dongwan Kang, Zhiguang Huo, Yongseok Park, and George C Tseng. Meta-analytic principal component analysis in integrative omics application. *Bioinformatics*, 34(8):1321–1328, 11 2017. ISSN 1367-4803. doi: 10.1093/bioinformatics/btx765.

Kim-Anh Lê Cao, Debra Rossouw, Christele Robert-Granié, and Philippe Besse. A sparse PLS for variable selection when integrating omics data. *Statistical applications in genetics and molecular biology*, 7(1), 2008. doi: 10.2202/1544-6115.1390.

Mario Lezcano-Casado. Trivializations for gradient-based optimization on manifolds. In *Advances in Neural Information Processing Systems, NeurIPS*, pages 9154–9164, 2019.

Chuanhai Liu and Donald B Rubin. ML estimation of the t distribution using EM and its extensions, ECM and ECME. *Statistica Sinica*, pages 19–39, 1995.

Yafei Lyu, Lingzhou Xue, Feipeng Zhang, Hillary Koch, Laura Saba, Katerina Kechris, and Qunhua Li. Condition-adaptive fused graphical lasso (cfgl): An adaptive procedure for inferring condition-specific gene co-expression network. *PLoS computational biology*, 14(9):e1006436, 2018. doi: 10.1371/journal.pcbi.1006436.

Daniel Marbach, James C Costello, Robert Küffner, Nicole M Vega, Robert J Prill, Diogo M Camacho, Kyle R Allison, Manolis Kellis, James J Collins, et al. Wisdom of crowds for robust gene network inference. *Nature methods*, 9(8):796–804, 2012. doi: 10.1038/nmeth.2016.

Nicolai Meinshausen and Peter Bühlmann. Stability Selection. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 72(4):417–473, 2010. doi: 10.1111/j.1467-9868.2010.00740.x.

Gemma E. Moran, Veronika Ročková, and Edward I. George. Spike-and-slab Lasso biclustering. *The Annals of Applied Statistics*, 15(1):148 – 173, 2021. doi: 10.1214/20-AOAS1385.

Pierre Nicolas, Ulrike Mäder, Etienne Dervyn, Tatiana Rochat, Aurélie Leduc, Nathalie Pigeonneau, Elena Bidnenko, Elodie Marchadier, Mark Hoebeke, Stéphane Aymerich, et al. Condition-dependent transcriptome reveals high-level regulatory architecture in Bacillus subtilis. *Science*, 335(6072):1103–1106, 2012. doi: 10.1126/science.1206848.

Teodora Pandeva and Patrick Forré. Multi-View Independent Component Analysis for Omics Data Integration. In *2023 ICLR First Workshop on Machine Learning & Global Health*, 2023.

Princy Parsana, Claire Ruberman, Andrew E Jaffe, Michael C Schatz, Alexis Battle, and Jeffrey T Leek. Addressing confounding artifacts in reconstruction of gene co-expression networks. *Genome biology*, 20(1):1–6, 2019.

Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in PyTorch. 2017.

Tiago Pedreira, Christoph Elfmann, and Jörg Stülke. The current state of SubtiWiki, the database for the model organism Bacillus subtilis. *Nucleic Acids Research*, 50(D1):D875–D882, 10 2021. ISSN 0305-1048. doi: 10.1093/nar/gkab943.

Francesca Petralia, Li Wang, Jie Peng, Arthur Yan, Jun Zhu, and Pei Wang. A new method for constructing tumor specific gene co-expression networks based on samples with tumor purity heterogeneity. *Bioinformatics*, 34(13):i528–i536, 2018. doi: 10.1093/bioinformatics/bty280.

Kevin Rychel, Anand V Sastry, and Bernhard O Palsson. Machine learning uncovers independently regulated modules in the Bacillus subtilis transcriptome. *Nature Communications*, 11(1):1–10, 2020. doi: 10.1038/s41467-020-20153-9.

Wouter Saelens, Robrecht Cannoodt, and Yvan Saeys. A comprehensive evaluation of module detection methods for gene expression data. *Nature Communications*, 9(1):1–12, 2018. doi: 10.1038/s41467-018-03424-4.

Anand V Sastry, Ye Gao, Richard Szubin, Ying Hefner, Sibei Xu, Donghyuk Kim, Kumari Sonal Choudhary, Laurence Yang, Zachary A King, and Bernhard O Palsson. The *Escherichia coli* transcriptome mostly consists of independently regulated modules. *Nature Communications*, 10 (1):1–14, 2019. doi: 10.1038/s41467-019-13483-w.

Souvik Seal, Qunhua Li, Elle Butler Basner, Laura M Saba, and Katerina Kechris. Rcfgl: Rapid condition adaptive fused graphical lasso and application to modeling brain region co-expression networks. *PLoS computational biology*, 19(1):e1010758, 2023. doi: 10.1371/journal.pcbi.1010758.

Age K Smilde, Ingrid Måge, Tormod Naes, Thomas Hankemeier, Mirjam Anne Lips, Henk AL Kiers, Ervim Acar, and Rasmus Bro. Common and distinct components in data fusion. *Journal of Chemometrics*, 31(7):e2900, 2017. doi: 10.1002/cem.2900.

Oliver Stegle, Christoph Lippert, Joris M Mooij, Neil Lawrence, and Karsten Borgwardt. Efficient inference in matrix-variate Gaussian models with iid observation noise. *Advances in Neural Information Processing Systems*, 24, 2011.

Wencheng Yin, Luis Mendoza, Jimena Monzon-Sandoval, Araxi O Urrutia, and Humberto Gutierrez. Emergence of co-expression in gene regulatory networks. *PloS one*, 16(4):e0247671, 2021. doi: 10.1371/journal.pone.0247671.

Appendix A. Related Work

Inferring GCNs from data can be very challenging, mainly due to hidden confounders and batch effects associated with the different data sources. In response, current research in gene co-expression analysis often makes specific assumptions about the data generation model to deal with this complexity. This is typically represented by a noisy decomposition model: $\mathbf{X} = \mathbf{S}A + \mathbf{E}$, where $\mathbf{X} \in \mathbb{R}^{p \times n}$ is a gene expression matrix describing the activity of p genes across n different samples (experiments, patients, tissues, etc.), $\mathbf{S} \in \mathbb{R}^{p \times k}$ is the *gene loading matrix*, A is the *sample loading matrix*, and $\mathbf{E}$ is the additive noise. These approaches can be broadly categorized into decomposition methods and their refinements, biclustering algorithms.

Decomposition methods, including Independent Component Analysis (ICA), Principal Component Analysis (PCA), and other variations of factor analysis, have shown remarkable effectiveness in identifying clusters of genes connected in the GCN. These methods are used to analyze single data sets (Saelens et al., 2018; Rychel et al., 2020) as well as to integrate data from multiple studies (Lê Cao et al., 2008; Smilde et al., 2017; Kim et al., 2017; Pandeva and Forré, 2023). A common assumption is that GCNs can be reconstructed from the gene loadings, where gene clusters are identified from each latent vector, a column in the gene loadings matrix $\mathbf{S}$, usually by thresholding. Often, these clusters are assumed to represent sets of genes connected within the GCN and mapped to gene modules with a common function.

Biclustering algorithms aim to cluster genes and samples simultaneously by applying sparsity constraints to both gene and sample loadings, e.g., (Moran et al., 2021; Hochreiter et al., 2010; Gao et al., 2016; Kim and Park, 2007), providing a principled approach for a two-fold clustering. This approach assumes that the sample loading matrix A will have a sparse pattern, i.e., only a small group of genes will deviate within a small subset of samples. These methods are particularly useful for subgroup analyses, such as classifying patients into different subtypes based on gene expression levels.

Despite their ability to cluster, all these methods do not model the relationships between clusters and thus do not provide a comprehensive strategy for inferring gene co-expression graphs. One exception is the Kronecker graphical LASSO approach by (Stegle et al., 2011), which constructs a sparse graph structure while modeling sample covariance. However, this method has not been extended to handle multiple datasets collected from different labs and may lack robustness against data contamination. On the other hand, existing methods that use the graphical LASSO to infer GCNs from various data sources (Danaher et al., 2014; Guo et al., 2011) do not address the confounding variables in the experiments and assume that the data are independent and identically distributed.

Appendix B. Identifiability

In our analysis, we will make use of the multivariate elliptical distributions, denoted by $E_p(\mu, \Sigma)$, whose density $f(x; \mu, \Sigma)$ is proportional to $f(x; \mu, \Sigma) \propto g((x-\mu)^\top \Sigma (x-\mu))$ for some measurable function g and a positive semi-definite dispersion matrix Σ and median μ. An example of such elliptical distributions is the Gaussian and multivariate t-distribution. First, we show that the sample loadings are identifiable up to scaling and permutation, provided that none of the gene loadings have Gaussian marginals. This result is an extension of Theorem 10.3 in (Kagan et al., 1973) for the multivariate case:

Lemma 2 *Let $\mathbf{X} \in \mathbb{R}^{p \times n}$ be a random matrix. Assume the following two representations of $\mathbf{X}$*

$$\mathbf{S}^{(1)} A^{(1)} = \mathbf{X} = \mathbf{S}^{(2)} A^{(2)},$$

with the following properties for $i = 1, 2$:

1. *$A^{(i)} \in \mathbb{R}^{k^{(i)} \times n}$ is a (non-random) matrix with a full row rank*
2. *$\mathbf{S}^{(i)} \in \mathbb{R}^{p \times k^{(i)}}$ is a random matrix such that the columns of $\mathbf{S}^{(i)}$ are mutually independent.*

If the i-th row of $A^{(1)}$ is not proportional to any row of $A^{(2)}$ then the i-th column of $\mathbf{S}^{(1)}$ has Gaussian distributed marginals. Additionally, if the i-th column of $\mathbf{S}^{(1)}$ follows an elliptical distribution, then it is multivariate Gaussian.

Proof of Lemma 2 **Proof** W.l.o.g. let $i = 1$. According to (Kagan et al., 1973, Lemma 10.2.2) there exists a $n \times 2$ matrix H such that the matrices $C_1 = A^{(1)} H$ and $C_2 = A^{(2)} H$ of orders $k^{(1)} \times 2$ and $k^{(2)} \times 2$ respectively have the following property; the first row of C_1 is not proportional to any of the other rows of C_1 or to any of the rows of C_2.

Now consider the following algebraic relationship for $\mathbf{Y} = \mathbf{X}H$:

$$\mathbf{S}^{(1)} C_1 = \mathbf{Y} = \mathbf{S}^{(2)} C_2,$$

where $\mathbf{Y} \in \mathbb{R}^{p \times 2}$. For each row $r = 1, \dots p$ of $\mathbf{Y}$ we have the two equivalent representations

$$\mathbf{S}^{(1)}_{r,:} C_1 = \mathbf{Y}_{r,:} = \mathbf{S}^{(2)}_{r,:} C_2.$$

Thus, by (Kagan et al., 1973, Lemma 10.2.4), it follows that $\mathbf{S}^{(1)}_{r,1}$ is Gaussian distributed because the first row of C_1 is not proportional to any of the other rows of C_1 or to the one of C_2. Consequently, this implies that the marginal distributions of $\mathbf{S}^{(1)}_{:,1}$ are Gaussians since it is elliptically distributed. Given that $\mathbf{S}^{(1)}_{:,1}$ is elliptical with the previous argument it follows that it is Gaussian (Fang et al., 2018). ■

Theorem 2 *Let $\mathbf{X} \in \mathbb{R}^{p \times n}$ be a random matrix. Assume the following two representations of $\mathbf{X}$*

$$\mathbf{S}^{(1)} A^{(1)} = \mathbf{X} = \mathbf{S}^{(2)} A^{(2)}$$

with the following properties for $i = 1, 2$:

1. *$A^{(i)} \in \mathbb{R}^{k^{(i)} \times n}$ is a (non-random) matrix with full row rank, i.e. $\operatorname{rank}(A^{(i)}) = k^{(i)}$*
2. *$\mathbf{S}^{(i)} \in \mathbb{R}^{p \times k^{(i)}}$ is a random matrix such that*
   (a) *The columns of $\mathbf{S}^{(i)}$ are mutually independent,*
   (b) *For $k = 1, \dots, k^{(i)}$ the vectors $\mathbf{S}^{(i)}_{:,k}$ are distributed according to a non-Gaussian elliptical distribution $E_p(\mu^{(i)}, \Sigma^{(i)})$ with mean $\mu^{(i)}$ and a dispersion matrix $\Sigma^{(i)}$.*
   (c) *Additionally, $\mathbf{S}^{(i)}_{:,k}$ the random vectors do not have Gaussian components.*

Then $k^{(1)} = k^{(2)} = k$ and there exist a permutation matrix $P = P(\rho) \in \mathbb{R}^{k\times k}$ given by $Pe_j = e_{\rho(j)}$ and a constant $c > 0$ such that:

$$A^{(2)} = cPA^{(1)}, \qquad \Sigma^{(2)} = \frac{\Sigma^{(1)}}{c^2} \qquad \mu^{(2)} = \frac{\mu^{(1)}}{c}.$$

Proof Lemma 2 establishes that each row of matrix $A^{(1)}$ is proportional to a row of $A^{(2)}$. Now if we assume that $k^{(1)} > k^{(2)}$ then there must be at least two distinct rows in $A^{(1)}$ that are proportional to the same row of $A^{(2)}$. This contradicts the assumption that both $A^{(1)}$ and $A^{(2)}$ have full row rank. Thus, it follows that $k^{(1)} = k^{(2)} =: k$ and there exist a permutation matrix $P \in \mathbb{R}^{k\times k}$ and an invertible diagonal matrix $\Lambda = \mathrm{diag}(\lambda_1, \ldots, \lambda_k) \in \mathbb{R}^{k\times k}$ such that $A^{(2)} = \Lambda P A^{(1)}$.

Note that for the characteristic function of a matrix $\mathbf{S}$ that fulfills 2a) to c) for some mean μ and dispersion matrix Σ holds

$$\begin{aligned}\chi_{\mathbf{S}}(\mathbf{t}) &= \mathbb{E}\left[\exp(i\,\mathrm{tr}(\mathbf{t}^\top \mathbf{S}))\right] = \mathbb{E}\left[\exp\left(i\sum_j \mathbf{t}_{:,j}^\top \mathbf{S}_{:,j}\right)\right] \\ &= \prod_j \chi_{\mathbf{S}_{:,j}}\left(\mathbf{t}_{:,j}\right) = \prod_j \chi_{\mathbf{S}_{:,1}}\left(\mathbf{t}_{:,j}\right) \\ &= \prod_j \exp\left(i\mathbf{t}_{:,j}^\top \mu\right)\psi\left(\mathbf{t}_{:,j}^\top \Sigma \mathbf{t}_{:,j}\right),\end{aligned}$$

where ψ is the characteristic generator and $\mathbf{t} \in \mathbb{R}^{p\times k}$.

Let $\tilde{\mathbf{S}}^{(2)} = \mathbf{S}^{(2)}P$. Then we get for the characteristic functions of $\tilde{\mathbf{S}}^{(2)}$ and $\mathbf{S}^{(1)}$ for all $\mathbf{t} \in \mathbb{R}^{p\times k}$ $\chi_{\mathbf{S}^{(1)}}(\mathbf{t}) = \chi_{\tilde{\mathbf{S}}^{(2)}\Lambda}(\mathbf{t})$

$$\begin{aligned}&\prod_j \exp\left(i\mathbf{t}_{:,j}^\top \mu^{(1)}\right)\psi_1\left(\mathbf{t}_{:,j}^\top \Sigma^{(1)} \mathbf{t}_{:,j}\right) \\ &\quad = \prod_j \exp\left(i\lambda_j \mathbf{t}_{:,j}^\top \mu^{(2)}\right)\psi_2\left(\lambda_j^2 \mathbf{t}_{:,j}^\top \Sigma^{(2)} \mathbf{t}_{:,j}\right),\end{aligned}$$

where ψ_i is the characteristic generator corresponding to the $i-$the representation. Consequently, for each j with $\mathbf{t}_{:,j} = t \in \mathbb{R}^p$ and otherwise $\mathbf{t}_{:,r} = 0$ for all $r \neq j$ we get

$$\begin{aligned}&\exp\left(it^\top \mu^{(1)}\right)\psi_1\left(t^\top \Sigma^{(1)} t\right) \\ &\quad = \exp\left(i\lambda_j t^\top \mu^{(2)}\right)\psi_2\left(\lambda_j^2 t^\top \Sigma^{(2)} t\right)\end{aligned}$$

It follows that $\lambda_1 = \ldots = \lambda_k = c$ and $\mu^{(1)} = c\mu^{(2)}$, and $\Sigma^{(1)} = c^2\Sigma^{(2)}$. ■

Next, we show that by imposing additional constraints on the gene loadings - in particular, requiring that they come from the same elliptic non-Gaussian multivariate distribution - it becomes possible to determine that the sample matrix, along with its locations and dispersion matrix, are identifiable up to a scalar:

Theorem 3 *Let $\mathbf{X} \in \mathbb{R}^{p\times n}$ be a random matrix. Assume the following two representations of $\mathbf{X}$*

$$\mathbf{S}^{(1)}A^{(1)} + \mathbf{Z}^{(1)}B^{(1)} = \mathbf{X} = \mathbf{S}^{(2)}A^{(2)} + \mathbf{Z}^{(2)}B^{(2)}$$

with the following properties for $i = 1, 2$:

1. $(A^{(i)\top}|B^{(i)\top})^\top \in \mathbb{R}^{(k^{(i)}+l^{(i)})\times n}$ *is a (non-random) matrix with full row rank with* $\operatorname{rank}(A^{(i)}) = k^{(i)}$ *and* $\operatorname{rank}(B^{(i)}) = l^{(i)} \leq n - k^{(i)}$

2. $\mathbf{S}^{(i)} \in \mathbb{R}^{p\times k^{(i)}}$ *and* $\mathbf{Z}^{(i)} \in \mathbb{R}^{p\times l^{(i)}}$ *are random matrices such that for* $i = 1, 2$ *and* $\mathbf{V}^{(i)} \in \{\mathbf{S}^{(i)}, \mathbf{Z}^{(i)}\}$

   (a) *The columns of* $\mathbf{V}^{(i)}$ *are mutually independent,*

   (b) *The column vectors* $\mathbf{V}^{(i)}_{:,k}$ *are distributed according to a non-Gaussian elliptical distribution* $E_p(\mu_V^{(i)}, \Sigma_V^{(i)})$ *with location* $\mu_V^{(i)}$ *and a dispersion matrix* $\Sigma_V^{(i)}$.

   (c) *Additionally, the random column vectors of* $\mathbf{S}_d$ *and* $\mathbf{Z}_d$ *do not have Gaussian components.*

3. *the latents* $\mathbf{S}^{(i)}$ *and noise matrix* $\mathbf{Z}^{(i)}$ *are independent and there exist no* $\lambda \in \mathbb{R}$ *such that* $\mu_Z^{(i)} = \lambda\mu_S^{(i)}, \Sigma_Z^{(i)} = \lambda^2\Sigma_S^{(i)}$.

Then $k^{(1)} = k^{(2)} = k$ *and* $l^{(1)} = l^{(2)} = l$ *and exist permutation matrices* $P_A, P_B \in \mathbb{R}^{k\times k}$ *and constants* $c_A, c_B > 0$ *such that:*

$$A^{(2)} = c_A P_A A^{(1)}, \qquad B^{(2)} = c_B P_B B^{(1)},$$
$$\Sigma_S^{(2)} = \frac{\Sigma_S^{(1)}}{c_A^2}, \qquad \mu_S^{(2)} = \frac{\mu_S^{(1)}}{c_A},$$
$$\Sigma_Z^{(2)} = \frac{\Sigma_Z^{(1)}}{c_B^2}, \qquad \mu_Z^{(2)} = \frac{\mu_Z^{(1)}}{c_B}.$$

Proof of Theorem 3 **Proof** According to Lemma 2 each row of $(A^{(1)\top}|B^{(1)\top})^\top$ is proportional to a row of $(A^{(2)\top}|B^{(2)\top})^\top$. With similar arguments as above it holds that $k^{(1)} + l^{(1)} = k^{(2)} + l^{(2)}$.

Suppose that the $j-$th row of $A^{(1)}$ is proportional to the $r-$th row of $B^{(1)}$. It follows that that there exist a constant λ such that for all $t \in \mathbb{R}^p$:

$$\exp\left(it^\top\mu_S^{(1)}\right)\psi_1\left(t^\top\Sigma_S^{(1)}t\right) = \exp\left(i\lambda t^\top\mu_Z^{(2)}\right)\psi_2\left(\lambda^2 t^\top\Sigma_Z^{(2)}t\right),$$

i.e., $\mu_S^{(1)} = \lambda\mu_Z^{(2)}$ and $\Sigma_S^{(1)} = \lambda^2\Sigma_Z^{(2)}$. Thus, $k^{(1)} = k^{(2)}$ and $l^{(1)} = l^{(2)}$. The rest follows from Theorem 2. ■

Corollary 1 *Let* $\mathbf{X}_1, \ldots, \mathbf{X}_D$ *with* $\mathbf{X}_d \in \mathbb{R}^{p\times n_d}$ *be random matrices with the following two representations*

$$\mathbf{S}_d^{(1)}A_d^{(1)} + \mathbf{Z}_d^{(1)}B_d^{(1)} = \mathbf{X}_d = \mathbf{S}_d^{(2)}A_d^{(2)} + \mathbf{Z}_d^{(2)}B_d^{(2)}$$

with the following properties for $i = 1, 2$ *and* $d = 1, \ldots, D$:

1. $(A_d^{(i)\top}|B_d^{(i)\top})^\top \in \mathbb{R}^{(k_d^{(i)}+l_d^{(i)})\times n_d}$ *is a (non-random) matrix with full row rank:*

$$\operatorname{rank}(A_d^{(i)}) = k_d^{(i)}, \qquad \operatorname{rank}(B_d^{(i)}) = l_d^{(i)} \leq n_d - k_d^{(i)},$$

2. *the columns of* $\mathbf{S}_d^{(i)}$ *are independent and are distributed according to a non-Gaussian elliptical distribution* $E_p(\mu_{S_d}^{(i)}, \Sigma_{S_d}^{(i)})$ *with location* $\mu_{S_d}^{(i)}$ *and a dispersion matrix* $\Sigma_S^{(i)} := \Sigma_{S_1}^{(i)} = \ldots = \Sigma_{S_D}^{(i)}$.

3. *the columns of* $\mathbf{Z}_d^{(i)}$ *are noise random variables and are i.i.d non-Gaussian elliptical distributed* $E_p(\mu_{Z_d}^{(i)}, \Sigma_{Z_d}^{(i)})$ *with location* $\mu_{Z_d}^{(i)}$ *and a dispersion matrix* $\Sigma_{Z_d}^{(i)}$. *Furthermore, for each* d *there exist no* $\lambda \in \mathbb{R}$ *such that* $\mu_{Z_d}^{(i)} = \lambda\mu_S^{(i)}, \Sigma_{Z_d}^{(i)} = \lambda^2\Sigma_S^{(i)}$.

4. *the latents* $\mathbf{S}_d^{(i)}$ *and noise matrix* $\mathbf{Z}_d^{(i)}$ *are mutually independent.*

5. *Additionally, the random column vectors of* $\mathbf{S}_d$ *and* $\mathbf{Z}_d$ *do not have Gaussian components.*

Then, for $d = 1, \ldots, D$, $k_d^{(1)} = k_d^{(2)} = k_d$ *and* $l_d^{(1)} = l_d^{(2)} = l_d$. *Furthermore, there exist permutation matrices* $P_{A_1}, \ldots, P_{A_D}, P_{B_1}, \ldots, P_{B_D}$ *and constants* $c_A, c_{B_1}, \ldots, c_{B_D} > 0$ *such that:*

$$A_d^{(2)} = c_A P_{A_d} A_d^{(1)}, \qquad B_d^{(2)} = c_{B_d} P_{B_d} B_d^{(1)},$$
$$\Sigma_S^{(2)} = \frac{\Sigma_S^{(1)}}{c_A^2}, \qquad \mu_{S_d}^{(2)} = \frac{\mu_{S_d}^{(1)}}{c_A},$$
$$\Sigma_{Z_d}^{(2)} = \frac{\Sigma_{Z_d}^{(1)}}{c_{B_d}^2}, \qquad \mu_{Z_d}^{(2)} = \frac{\mu_{Z_d}^{(1)}}{c_{B_d}}.$$

Proof of Corollary 1 Theorem 3 guarantees the identifiability results for each view separately, i.e. for each $d = 1, \ldots, D$ there exist permutation matrices P_{A_d}, P_{B_d} and constants $c_{A_d}, c_{B_d} > 0$ such that:

$$A_d^{(2)} = c_{A_d} P_{A_d} A_d^{(1)}, \qquad B_d^{(2)} = c_{B_d} P_{B_d} B_d^{(1)},$$
$$\Sigma_S^{(2)} = \frac{\Sigma_S^{(1)}}{c_{A_d}^2}, \qquad \mu_{S_d}^{(2)} = \frac{\mu_{S_d}^{(1)}}{c_{A_d}},$$
$$\Sigma_{Z_d}^{(2)} = \frac{\Sigma_{Z_d}^{(1)}}{c_{B_d}^2}, \qquad \mu_{Z_d}^{(2)} = \frac{\mu_{Z_d}^{(1)}}{c_{B_d}}.$$

It follows that for all $d = 1, \ldots, D$:

$$\Sigma_S^{(2)} = \frac{\Sigma_S^{(1)}}{c_{A_d}^2}.$$

Thus, $c_A := c_{A_1} = \ldots = c_{A_D}$.

Appendix C. Dependence Structure and Properties of the Multivariate t-Distribution

C.1. Alternative Generative Model for the Multivariate t-Distribution

The probability density function of the multivariate t-distribution with ν degrees of freedom, mean vector μ, and scale matrix $\mathbf{\Sigma}$ in p dimensions is given by:

$$f(\mathbf{x}) = \frac{\Gamma\left(\frac{\nu+p}{2}\right)}{\Gamma\left(\frac{\nu}{2}\right)(\nu\pi)^{p/2}|\mathbf{\Sigma}|^{1/2}} \left(1 + \frac{1}{\nu}(\mathbf{x}-\mu)^T\mathbf{\Sigma}^{-1}(\mathbf{x}-\mu)\right)^{-\frac{\nu+p}{2}}$$

where:

- $\mathbf{x}$ is the variable vector,
- μ is the mean vector,
- $\mathbf{\Sigma}$ is the scale matrix,
- ν is the degrees of freedom,
- Γ is the gamma function.

The following result is central to the EM procedure and is shows that the multivariate t-distribution can be expressed by means of the multivariate normal distributed random variable and Gamma distributed random variable:

Theorem 4 ((Arellano-Valle and Bolfarine, 1995)) *Let $S \sim t_p(\nu, \mu, \Sigma)$ for some mean μ and positive semi-definite matrix Σ. Then, there exist random variables τ and N that follow Gamma distribution $\Gamma(\frac{\nu}{2}, \frac{\nu}{2})$ and a Gaussian distribution $\mathcal{N}(0, \Sigma)$, respectively, such that $S \sim \mu + N/\sqrt{\tau}$.*

Proof of Lemma 1 " $\Leftarrow$ " This direction follows directly from the chain rule of probabilities.

" $\Rightarrow$ " Assume that the DAG is not fully connected, i.e. there exist sets $A, B, C \subset V$, $A \neq \emptyset, B \neq \emptyset$ such that the random variables S_A and S_B are d-separated given S_C ($S_A \perp_G S_B | S_C$). Thus, it follows that $S_A \perp\!\!\!\perp S_B \mid S_C$ which implies that $p(S_A \mid S_B, S_C) = p(S_A | S_C)$.

According to (Arellano-Valle and Bolfarine, 1995) the joint distribution of S_A, S_B, S_C, their conditionals and marginals follow a multivariate t-distribution. More precisely, let $d = |A| + |B| + |C|$, $\mu_d = (\mu_A^\top, \mu_B^\top, \mu_C^\top)^\top$, $\Sigma = \Sigma_{(A,B,C),(A,B,C)}$, then $S_d = (S_A, S_B, S_C) \sim t_d(\nu, \mu_d, \Sigma)$. Furthermore, for the conditional distributions we have

$$S_A \mid S_B, S_C \sim t_{|A|}\left(\nu + |B| + |C|, \mu_{A|B,C}, \frac{\nu + d_{B,C}}{\nu + |B| + |C|}\Sigma_{A|B,C}\right) \tag{5}$$
$$S_A \mid S_C \sim t_{|A|}\left(\nu + |C|, \mu_{A|C}, \frac{\nu + d_C}{\nu + |C|}\Sigma_{A|C}\right)$$

Then, it follows that $\nu + |B| + |C| = \nu + |B| + |C|$ which implies that $|B| = 0$.

Appendix D. Parameter Inference: Background

D.1. Graphical LASSO

The Graphical lasso (GLASSO) is a maximum likelihood estimator for inferring graph structure within high-dimensional multivariate normal distributed data through estimating a sparse precision matrix (Friedman et al., 2008). More precisely, let $\mathbf{X} = (\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_n)$ be a collection of n i.i.d. samples distributed according to the multivariate normal distribution $\mathcal{N}(0, \Theta^{-1})$, where $\Theta^{-1} \in \mathbb{R}^{p \times p}$ is the covariance matrix and its inverse Θ known as the precision matrix. The underlying undirected graph structure among the variables can be inferred directly from the precision matrix: a non-zero entry Θ_{ij} indicates an undirected edge between the i-th and j-the variables in the multivariate vector. GLASSO estimates Θ by maximizing the posterior distribution of $\mathbf{X}$ given $\Theta := \Sigma^{-1}$ which is proportional to

$$p(\mathbf{X}, \Theta) = p_\lambda(\Theta) \prod_{i=1}^{n} \mathcal{N}(\mathbf{X}_i | \mu, \Theta^{-1}) \quad \text{where} \quad \Theta \succ 0.$$

The prior $p_\lambda(\Theta)$ on the positive-definite matrices Θ parametrized by $\lambda > 0$ is defined as

$$p_\lambda(\Theta) \propto \exp\left(-\lambda \|\Theta\|_1\right) \quad \text{with} \quad \|\Theta\|_1 = \sum_{i,j} |\Theta_{ij}|.$$

Thus, the MLE problem that GLASSO solves can be formalized as follows

$$\max_{\Theta \succ 0} \ln p(\mathbf{X}, \Theta) \equiv \min_{\Theta \succ 0} -\ln \det(\Theta) + \operatorname{tr}(\hat{\Sigma}\Theta) + \lambda \|\Theta\|_1, \tag{6}$$

where S is the empirical covariance matrix, $\hat{\Sigma} = \frac{1}{n}\sum_{i=1}^{n}(\mathbf{X}_i - \bar{\mathbf{X}})(\mathbf{X}_i - \bar{\mathbf{X}})^\top$, and $\bar{\mathbf{X}}$ is the empirical mean. Intuitively, the parameter λ controls the sparsity level of the precision matrix Θ. Specifically, selecting a higher value for λ leads to sparser precision matrix estimates.

D.2. Student's t-Lasso

The accuracy of graph inference can be significantly compromised by deviations from the normal distribution assumption. To address this robustness issue, (Finegold and Drton, 2011) propose an alternative to GLASSO for inferring graph structure of multivariate Student's t-distribution which we call TGLASSO. Consider the setting from above, where we are given a collection of n i.i.d samples $\mathbf{X} = (\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_n)$. Then the joint distribution of the data $\mathbf{X}$ and precision matrix Θ is given by

$$p(\mathbf{X}, \Theta) = p_\lambda(\Theta) \prod_{i=1}^{n} t_{\nu,p}(\mathbf{X}_i | \mu, \Theta^{-1}) \quad \text{where} \quad \Theta \succ 0,$$

where the density function of the Student's t-distribution $t_{\nu,p}(\mu, \Theta^{-1})$ is given by

$$\frac{\Gamma\left((\nu+p)/2\right) \det \Theta^{1/2}}{(\pi\nu)^{p/2}\Gamma(\nu/2)\left(1 + \delta\left(\mathbf{x}; \mu, \Theta\right)/\nu\right)^{(\nu+p)/2}}$$

with

$$\delta\left(\mathbf{x}; \mu, \Theta\right) = (\mathbf{x} - \mu)^\top \Theta (\mathbf{x} - \mu), \quad \mathbf{x} \in \mathbb{R}^p.$$

Estimating the precision matrix in this setting is not tractable, and (Finegold and Drton, 2011) propose an Expectation-Maximization procedure for estimating Θ by exploiting the following generative model with latent variables $\mathbf{Z}_i$ and τ_i for each sample $\mathbf{X}_i$

$$\begin{aligned} \mathbf{Z}_i &\sim \mathcal{N}(0, \Theta^{-1}) \\ \tau_i &\sim \Gamma(\nu/2, \nu/2) \\ \mathbf{X}_i := \mu + \mathbf{Z}_i/\sqrt{\tau}_i &\sim t_{\nu,p}(\mu, \Theta^{-1}). \end{aligned}$$

The proposed EM procedure operates under the assumption that τ_i's are latent variables and that $\mathbf{X}_i|\tau_i \sim \mathcal{N}(\mu, (\tau_i\Theta)^{-1})$. This process iterates through two main steps: 1) Estimating the τ_i for fixed μ and Θ^{-1} and 2) Estimating μ and Θ^{-1}, where Θ is a solution to the GLASSO problem in Equation (6) for an empirical covariance matrix of the estimated $\mathbf{Z}$. More precisely, at step $t \geq 0$ the EM procedure becomes

E-step: For fixed estimated $\mu^{(t-1)}$ and $\Theta^{(t-1)}$ compute

$$\tau_i^{(t)} = \frac{\nu + p}{\nu + \delta\left(\mathbf{X}_i; \mu^{(t-1)}, \Theta^{(t-1)}\right)}$$

M-step: Calculate $\mu^{(t)}$ and $\Sigma^{(t)}$

$$\mu^{(t)} = \frac{\sum_{i=1}^n \tau_i^{(t)} \mathbf{X}_i}{\sum_{i=1}^n \tau_i^{(t)}} \qquad \Sigma^{(t)} = \frac{1}{n}\sum_{i=1}^n \tau_i^{(t)} \left(\mathbf{X}_i - \mu^{(t)}\right)\left(\mathbf{X}_i - \mu^{(t)}\right)^\top \tag{7}$$

Estimate $\Theta^{(t)}$ via solving the GLASSO optimization problem

$$\Theta^{(t)} \in \underset{\Theta \succ 0}{\arg\min} - \ln\det(\Theta) + \operatorname{tr}(\Sigma^{(t)}\Theta) + \lambda\|\Theta\|_1$$

Appendix E. Experiments

E.1. Implementation

Implementation of MVTLASSO Here are the key points of the training and implementation. The implementation follows the steps explained in Section 3 with the following differences:

1. The optimization problem in step 3 is convex when $\{W_d\}_{d=1}^D$ are positive semi-definite matrices. In the general case, we only require $\{W_d\}_{d=1}^D$ to be invertible which makes solving equation 4 more challenging. However, by treating the datasets $\mathbf{X}_1, \ldots, \mathbf{X}_D$ as instances of ICA we can transform the original problem equation 4 into finding orthogonal matrices W_d. This process incorporates data whitening—a preprocessing step that ensures features are uncorrelated and have uniform variance, typically achieved through eigen-decomposition—commonly employed prior to applying ICA. We utilize the Python library `pytorch` (Paszke et al., 2017) for the implementation and employ the `geotorch` library (Lezcano-Casado, 2019) to enforce orthogonality constraints on each view's unmixing matrices. The optimization leverages the L-BFGS algorithm, a stochastic gradient-based method.

2. We initialize the parameter W_d with the estimated of FastICA, and the parameters μ_d, σ_d and Θ using a few iterations (not necessarily to convergence) of TLASSO by Finegold and Drton (2011).

3. We found that estimating the precision matrix through the partial correlation matrix instead of the covariance improves the MVTLASSO performance in some cases as for *S. aureus* (the first iteration of (Carter et al., 2024)). This means the input matrix to the graphical lasso has entries $\tilde{\Sigma}(i,j) = \frac{\Sigma_{ij}}{\sqrt{\Sigma(ii)\Sigma(jj)}}$. After that the estimate of the glasso $\tilde{\Theta}$ is mapped back to an estimate of the precision matrix via: $\Theta(ij) = \frac{\tilde{\Theta}(ij)}{\sqrt{\Sigma(ii)\Sigma(jj)}}$.

4. The steps in the M-step are interdependent, i.e., steps 1 and 2 are based on the inverse sample matrix W_d. Therefore, it is possible to iterate steps 1 to 3 multiple times. In our implementation, however, we perform only a single iteration.

5. Our model relies on several hyperparameters, including the number of multivariate t-distributed vectors k_d used for graph inference and the penalty parameter λ. Ideally, these parameters could be determined by cross-validation. However, our EM procedure involves a GLASSO step in each iteration, which is computationally intensive. Therefore, we preselect the number of components prior to the parameter estimation process as described by Parsana et al. (2019). This resulted in 107 noise components and 53 gene loadings for *S.aureus*.

Implementation of baselines For GLASSO, we used the implementation available in the R package (Friedman et al., 2019). The variants GLASSO+Standardization includes preliminary steps where all samples are subjected to standardization, before GLASSO is applied for precision matrix estimation. The TLASSO is implemented according to (Finegold and Drton, 2011).

Stability Selection The fitting procedure for all GLASSO-based methods makes use of stability selection by Meinshausen and Bühlmann (2010) with a predefined range of penalty parameters. The steps of the procedure are outlined as follows:

1. The data is repeatedly subsampled by selecting 90% of all samples per view $N = 100$ times. For each subsample, the selected GCN inference method is applied using the predefined set of penalty parameters, Λ.

2. The outcomes for each penalty parameter are gathered in the selection probability matrix Π_λ, where $(\Pi_\lambda)_{ij}$ represents the proportion of the N precision matrices $\hat{\Theta}^{(1)}, \ldots, \hat{\Theta}^{(N)}$ indicating a nonzero edge between nodes i and j, i.e. $(\Pi_\lambda)_{ij} = \frac{\sum_l \mathbb{1}_{\{\hat{\Theta}_{ij}^{(l)} \neq 0\}}}{N}$.

3. We select the edges whose selection probability exceeds 50% for each penalty parameter.

4. The final graph can be constructed by collecting all edges inferred from the range of penalty parameters with weights $w_{ij} = \sum_\lambda (\Pi_\lambda)_{ij}$

The primary benefit of stability selection, as outlined by (Meinshausen and Bühlmann, 2010), is that it can reduce the risk of false positives, i.e., incorrectly identifying edges in the network. By requiring that an edge be consistently identified across many subsamples of the data, stability selection ensures that the edges selected are robust and not the result of random variations in the data.

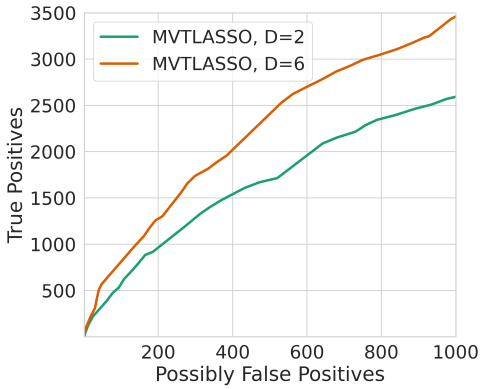

Figure 4: Comparison between six and two views for *B. Subtilis*.

Additional Experiment for *B. Subtilis* We compare two scenarios for *B. Subtilis*: one using the original setup with six views, and another using only two views. For the last experiment, we randomly selected 130 samples from each dataset and applied the same procedure as before. The final undirected graph was constructed using stability selection. As shown in Figure 4, including more views benefits the MVTLASSO method, as it results in the discovery of significantly more true positive edges.

E.2. Data Preprocessing

The dataset BSB1 is preprocessed following the method suggested by Rychel et al. (2020). Specifically, three samples (S3_3, G+S_1, and Mt0_2) were removed to ensure that the Pearson correlation between biological replicates was at least 0.9. Furthermore, we centered the data by subtracting the mean gene values in the M9 exponential growth condition. We used the preprocessed PY79 dataset by Arrieta-Ortiz et al. (2015). BSB1 and PY79 samples are then centered and rescaled before applying any graph inference procedures. We selected genes that are present in both datasets. In addition, we have split both datasets into three subsets of samples with experimental designs that are as different as possible to simulate six views instead of two.