MULTI-VIEW INDEPENDENT COMPONENT ANALYSIS WITH SHARED AND INDIVIDUAL SOURCES

Anonymous authors
Paper under double-blind review

ABSTRACT

Independent component analysis (ICA) is a blind source separation method for linear disentanglement of independent latent sources from observed data. We investigate the special setting of noisy linear ICA where the observations are split among different views, each receiving a mixture of shared and individual sources. We prove that the corresponding linear structure is identifiable, and the shared sources can be recovered, provided that sufficiently many diverse views and data points are available. To computationally estimate the sources, we optimize a constrained form of the joint log-likelihood of the observed data among all views. We show empirically that our objective recovers the sources in high dimensional settings, also in the case when the measurements are corrupted by noise. Finally, we apply the proposed model in a challenging real-life application, where the estimated shared sources from two large transcriptome datasets (observed data) provided by two different labs (two different views) lead to a more plausible representation of the underlying graph structure than existing baselines.

1 INTRODUCTION

Consider that the organizers of a musical festival want to promote it by releasing recordings of the key performing bands. To do this, they place microphones in different areas of the festival. Unfortunately, the microphones next to the main stage also record signals from nearby stages. That raises the question of how they can recover high-quality recordings from the main stage.

In this particular case, we collect data from different views (the microphones), each of which captures a mixture of shared sources (the main stage) and individual sources (the local stages). Moreover, we do not access any prior knowledge about the mixing structure and the latent sources. That is a typical example of a noisy blind source separation (BSS) problem where latent sources (bands' recordings) need to be recovered from observed data signals. However, standard BSS methods such as ICA would fail at recovering the model in this particular setting, as we will see later. We will discuss how to utilize prior knowledge about the data model to retrieve a better representation of the quantities of interest.

This kind of task is not only relevant for music festival organizers; it is prevalent in many areas of sciences and real-world applications. A prominent application of BSS (or ICA) is neuroscience, where often the goal is to recover spatially independent sources representing brain activity regions from magnetoencephalography (MEG) data (Vigário et al., 1997) or functional MRI (fMRI) data (McKeown & Sejnowski, 1998). This field's growing tendency shifts the focus from group-level inference to extracting individual-specific signals. For instance, a typical application will be collecting fMRI data from participants in a natural stimuli experiment where we want to investigate the individual (individual-specific brain functions) and shared (behavioral phenotypes) patterns in individuals' brain activity (Seghier & Price, 2018; Bartolomeo et al., 2017; Dubois & Adolphs, 2016). Solving this particular problem would benefit other biological domains, e.g., by assisting the scientist in accumulating information from heterogeneous experimental datasets for further downstream tasks, e.g., developing better tools for diagnostics (Planell et al., 2021; Karczewski & Snyder, 2018; El-Manzalawy, 2018).

All these applications have in common that the observed datasets come from diverse views and have both view-specific and shared biological signals. Due to the datasets' nature, we can also assume that the mixing is linear.

Related Work. Extracting signals in a multi-view BSS setting requires an additional modeling strategy. The existing body of work, inspired by the linear ICA literature, considers mostly shared response model applications (i.e., no individual sources), some of them adopting a maximum likelihood approach (Guo & Pagnoni, 2008; Richard et al., 2020; 2021) to model the noisy views of the proposed models. Many of these approaches, such as Group ICA (Calhoun et al., 2001), shared response ICA (SR-ICA) (Zhang et al., 2016), and MultiViewICA, incorporate a dimensionality reduction step for every view (CCA (Varoquaux et al., 2009; Richard et al., 2021) or PCA) to extract the mutual signal between the multiple objects before applying an ICA procedure on the reduced data. However, there are no guarantees that the pre-processing procedure will entirely remove the influence of the object-specific sources on the transformed data. Moreover, independent vector analysis (IVA) relaxes the assumption about the shared sources across views by allowing them to be dependent (Lee et al., 2008; Anderson et al., 2011; 2014; Engberg et al., 2016; Vía et al., 2011). However, as the previously discussed models, they fix the number of latent sources to be equal across views. To the best of our knowledge, two existing ICA-based methods exist for extracting shared and individual sources from data. Maneshi et al. (2016) proposes a heuristic way of using FastICA for the given task without discussing the identifiability of the results. The other exploits temporal correlations (Lukic et al., 2002) rather than the non-Gaussianity of the sources and thus is not applicable in the context we are considering.

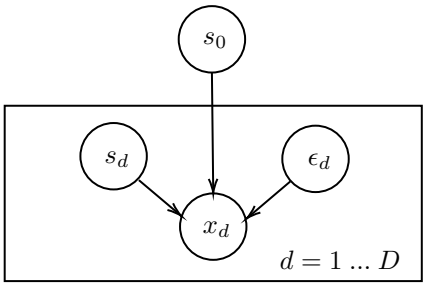


Figure 1: A graphical representation of 1

A common tool for analyzing multi-view data is canonical correlation analysis (CCA), initially proposed by Hotelling (1936). It finds two datasets' projections that maximize the correlation between the projected variables. Gaussian-CCA (Bach & Jordan, 2005), its kernelized version (Bach & Jordan, 2002) and deep learning (Andrew et al., 2013) formulations of the classical CCA problem aim to recover shared latent sources of variations from the multiple views. There are extensions of CCA that model the observed variables as a linear combination of group-specific and dataset-specific latent variables. (Klami et al., 2013) estimated with Bayesian inference methods or exponential families with MCMC inference (Virtanen, 2010). However, most of them assume that the latent sources are Gaussian or non-linearly related to the observed data (Wang et al., 2016) and thus lack identifiability results.

Recent advances in representation learning techniques are reported to be most suited to transfer learning or data integration problems since they infer latent sources from observed data that are subsequently mapped to fundamental biological processes (Uhler & Shivashankar, 2022). However, many of those methods (Tian et al., 2020; Federici et al., 2020) cannot recover both shared and individual signals across multiple measurements, and do not assure the identifiability of the proposed generative models.

Model Summary. To tackle the abovementioned issues, we formalize the stated noisy BSS model as a linear generative model for a multi-view data regime, assuming that the mixing matrix and number of individual sources are view-specific. By requiring that the sources are non-Gaussian and mutually independent, and the linear mixing matrices are with full column rank, we address the lack of identifiability problem that most related methods have. We adopt a maximum likelihood approach for the joint log-likelihood of the observed views, which we use to estimate the mixing matrices. Furthermore, we suggest a novel strategy for data integration of transcriptome datasets by performing a stochastic gradient descent for the proposed objective function. We show empirically that our method works well compared to the baseline methods (one of them is a state-of-the-art method for omics data integration) when the estimated components are used for a graph inference task.

Contributions. Our contribution can be summarized as follows:

1. We provide theoretical guarantees for the identifiability of the recovered linear structure and the sources and noise distributions.

2. We propose an optimization procedure based on MLE for estimating the mixing matrices.

2 PROBLEM FORMALIZATION

Consider the following D-view multivariate linear BSS model

$$\boxed{x_d = A_d(\tilde{s}_d + \epsilon_d) = A_{d0}s_0 + A_{d1}s_d + A_d\epsilon_d, \qquad d \in \{1, \dots, D\},} \tag{1}$$

where we assume that for $d = 1, \dots, D$

- $x_d \in \mathbb{R}^{k_d}$ is a random vector with $\mathbb{E}[x_d] = 0$,
- $\tilde{s}_d = (s_0^\top, s_d^\top)^\top$ are latent random sources with and $s_0 \in \mathbb{R}^c$ and $s_d \in \mathbb{R}^{k_d - c}$ being the shared and individual sources and $\mathbb{E}[\tilde{s}_d] = 0$ and $\mathrm{Var}[\tilde{s}_d] = \mathbb{I}_{k_d}$,
- $A_d \in \mathbb{R}^{k_d \times k_d}$ is a mixing matrix with full column rank, A_{d0} and A_{d1} are the columns corresponding to the shared and individual sources,
- $\epsilon_d \sim \mathcal{N}(0, \sigma^2 \mathbb{I}_{k_d})$ is Gaussian noise on the sources as in (Richard et al., 2020; 2021).

Additionally, we assume that the variables $s_{01}, \dots, s_{0c}, s_{11}, \dots, s_{1(k_1-c)}, \dots, s_{D1}, \dots, s_{D(k_D-c)}$, $\epsilon_{11}, \dots, \epsilon_{1k_1}, \dots, \epsilon_{D1}, \dots, \epsilon_{Dk_D}$ are mutually independent.

3 IDENTIFIABILITY RESULTS

In unsupervised machine learning methods, like ICA, the reliability of the algorithm cannot be directly verified outside of simulations due to the non-existence of labels. For this reason, theoretical guarantees are necessary to trust that the algorithm estimates the quantities of interest. For an ICA approach, we want the sources and mixing matrices to be (up to certain equivalence relations) unambiguously determined (or *identifiable*) by the data, at least in the large sample limit.

Identifiability results for noiseless single-view ICA are proved by (Comon, 1994). It turns out that if at most one of the latent sources is normal and the mixing matrix is invertible, then both the mixing matrix and sources can be recovered almost surely up to permutation and scaling. However, this result does not hold in the general additive noise setting. (Davies, 2004) shows that if the mixing matrix has a full column rank, then the structure is identifiable, but not the latent sources.

In the following we extend Comon (1994); Davies (2004) results for the noisy setting inspired from our model (see 1). Note that all identifiability theoretical results provided in this work can be seen as a corollary from the work by Kagan et al. (1973) and are proven with similar techniques (see Appendix A).

Theorem 3.1. *Let $x_1, \dots, x_D$ for $D \geq 2$ be random vectors with the following two representations:*

$$A_d^{(1)}\Big(\begin{bmatrix} s_0^{(1)} \\ s_d^{(1)} \end{bmatrix} + \epsilon_d^{(1)}\Big) = x_d = A_d^{(2)}\Big(\begin{bmatrix} s_0^{(2)} \\ s_d^{(2)} \end{bmatrix} + \epsilon_d^{(2)}\Big), \qquad d \in \{1, \dots, D\},$$

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

1. *$A_d^{(i)} \in \mathbb{R}^{p_d \times k_d^{(i)}}$ is a (non-random) matrix with full column rank, i.e. $\mathrm{rank}(A_d^{(i)}) = k_d^{(i)}$,*
2. *$\epsilon_d^{(i)} \in \mathbb{R}^{k_d^{(i)}}$ and $\epsilon_d^{(i)} \sim \mathcal{N}(0, \sigma_d^{(i)2}\mathbb{I}_{k_d^{(i)}})$ is a $k_d^{(i)}$-variate normal random variable,*
3. *$\tilde{s}_d^{(i)} = (s_0^{(i)\top}, s_d^{(i)\top})^\top$ with $s_0^{(i)} \in \mathbb{R}^{c^{(i)}}$ and $s_d^{(i)} \in \mathbb{R}^{k_d^{(i)} - c^{(i)}}$ is a random vector such that:*
   - *(a) the components of $\tilde{s}_d^{(i)}$ are mutually independent and each of them is a.s. a non-constant random variable,*
   - *(b) $\tilde{s}_d^{(i)}$ is non-normal with 0 mean and unit variance.*
4. *$\epsilon_d^{(i)}$ is independent from $s_0^{(i)}$ and $s_d^{(i)}$: $\epsilon_d^{(i)} \perp\!\!\!\perp s_0^{(i)}$ and $\epsilon_d^{(i)} \perp\!\!\!\perp s_d^{(i)}$.*

Then, $c^{(1)} = c^{(2)} =: c$ *and for all* $d = 1, \dots, D$ *we get that* $k_d^{(1)} = k_d^{(2)} =: k_d$*, and there exist a sign matrix* Γ_d *and a permutation matrix* $P_d \in \mathbb{R}^{k_d \times k_d}$ *such that:*

$$A_d^{(2)} = A_d^{(1)} P_d \Gamma_d,$$

and furthermore the sources and noise distributions are identifiable, i.e.

$$\begin{bmatrix} s_0^{(2)} \\ s_d^{(2)} \end{bmatrix} \sim \Gamma_d^{-1} P_d^{-1} \begin{bmatrix} s_0^{(1)} \\ s_d^{(1)} \end{bmatrix}, \qquad \sigma_d^{(2)} = \sigma_d^{(1)}.$$

Theorem 3.1 assures the identifiability of the mixing matrices and *sources and noise distributions* up to sign and permutation for a multi-view ($D \geq 2$) noisy ICA model. This is a more general case than 1 since here the noise distribution can be view-specific and the mixing matrices can be non-square. We also provide identifiability results for the single-view setting (see Theorem A.2). In the single-view case compared to the multiview one, we get similar but "weaker" results. The identifiability relation is up to sign, permutation, and *scale* only if the latent sources do not have normal components. This means that for any of the latent sources variables j in both equivalent representations in Theorem A.2, if we have $\tilde{s}_j^{(i)} \sim v + w$ with $v \perp\!\!\!\perp w$, then v and w are non-normal. That is a stronger assumption than the non-Gaussian assumption 3 (b) of Theorem 3.1.

Recent works by Richard et al. (2020; 2021); Anderson et al. (2014) provide identifiability results for a multi-set scenario by imposing different assumptions to the one in Theorem 3.1. For example, Richard et al. (2021) proposes a model with no individual sources, where the shared sources have unit variance. Under additional assumptions about the noise covariance matrices, the requirement about the non-Gaussianity of the sources in this particular model can be relaxed (Richard et al., 2021). Similar results about the shared sources are provided in Theorem A.4 by imposing additional requirements about the sources variance that are not covered by our model assumptions.

4 Optimization

Here, we derive the joint log-likelihood of the observed views which we use for estimating the mixing matrices. Let $z_d := W_d x_d = \tilde{s}_d + \epsilon_d$, and $z_d^{(1)} := s_0 + \epsilon_{d0} \in \mathbb{R}^c$ and $z_d^{(2)} := s_d + \epsilon_{d1} \in \mathbb{R}^{k_d - c}$, i.e. $z_d = (z_d^{(1)\top}, z_d^{(2)\top})^\top$. Furthermore, let $p_{Z_d^{(2)}}$ be the probability distribution of $z_d^{(2)}$ and $|W_d| = |\det W_d|$. Thus, by applying similar derivations as in Richard et al. (2020), we can show that the data log-likelihood is given by

$$\begin{aligned}\mathcal{L}(W_1, \dots, W_D) = &\sum_{i=1}^{N} \log f(\bar{s}_0^i) + \sum_{i=1}^{N} \sum_{d=1}^{D} \log p_{Z_d^{(2)}}(z_d^{(2)i}) + N \sum_{d=1}^{D} \log |W_d| \\ &- \frac{1}{2\sigma^2} \Big(\sum_{d=1}^{D} \operatorname{trace}(Z_d^{(1)} Z_d^{(1)\top}) - \frac{1}{D} \sum_{d=1}^{D} \sum_{l=1}^{D} \operatorname{trace}(Z_d^{(1)} Z_l^{(1)\top}) \Big) + C\end{aligned}$$

where $Z_d^{(1)} \in \mathbb{R}^{c \times N}$ for $d = 1, \dots D$ is the data matrix that stores N observations of $z_d^{(1)}$ and $\bar{s}_0^i = \sum_{d=1}^{D} z_d^{(1)i} / D$ and $f(\bar{s}_0) = \int \exp\Big(- \frac{D \|s_0 - \bar{s}_0\|^2}{2\sigma^2} \Big) p_{S_0}(s_0) ds_0$.

We further simplify the loss function by assuming that the data matrices $X_1 \in \mathbb{R}^{k_1 \times N}, \dots, X_D \in \mathbb{R}^{k_D \times N}$ are whitened. That consists of centering and linearly transforming the random variables' realizations x_d such that the resulting variable $\tilde{x}_d = K_d x_d$ has unit variance, $\mathbb{E}[\tilde{x}_d \tilde{x}_d^\top] = \mathbb{I}_{k_d}$, where K_d is the whitening matrix. Thus, from the last equation we get that $\mathbb{I}_{k_d} = \mathbb{E}[\tilde{x}_d \tilde{x}_d^\top] = (1 + \sigma^2) K_d A_d A_d^\top K_d^\top$. It follows that the matrix $(1 + \sigma^2)^{\frac{1}{2}} K_d A_d$ is orthogonal, which we estimate by the matrix W_d. After training we set $\hat{A}_d = K_d^{-1} W_d$ which differs from the true one by $(1 + \sigma^2)^{\frac{1}{2}}$. Due to the orthogonal constraints the objective function becomes

$$\mathcal{L}(W_1, \dots, W_D) \propto \sum_{i=1}^{N} \log f_\sigma(\bar{s}_0^i) + \sum_{i=1}^{N} \sum_{d=1}^{D} \log p_{Z_d^{(2)}}(z_d^{(2)i}) + \frac{1 + \sigma^2}{2D\sigma^2} \sum_{d=1}^{D} \sum_{l=1}^{D} \operatorname{trace}(Z_d^{(1)} Z_l^{(1)\top}) \tag{2}$$

where here $f_\sigma(\bar{s}_0) = \int \exp\Big(-\frac{D\|s_0 - (1+\sigma^2)^{\frac{1}{2}}\bar{s}_0\|^2}{2\sigma^2}\Big)p_{S_0}(s_0)ds_0$. This results from the fact that after whitening we have $\mathrm{trace}(Z_d^{(1)} Z_d^{(1)\top}) = c$ and $|W_d| = 1$. All proofs can be found in Appendix B. Note that in our optimization procedure, both $f_\sigma(\bar{s}_0)$ and $p_{Z_d^{(2)}}$, we approximate by the negative of a nonlinear function $g(s)$, e.g. $g(s) = \log\cosh(s)$ for super-Gaussian or $g(s) = -e^{-s^2/2}$ for sub-Gaussian sources. Moreover, since we do not estimate σ^2 we treat it as a Lagrange multiplier via the relation $\lambda = \frac{1+\sigma^2}{\sigma^2}$.

For the parameter estimation of the orthogonal unmixing matrices, we use the transformation framework proposed in (Lezcano-Casado, 2019; Lezcano-Casado & Martınez-Rubio, 2019)). The framework developed in (Lezcano-Casado, 2019) allows us to transform manifold optimization problems to unconstrained (Euclidean) optimization problems. To accomplish the transformation, the scheme uses trivializations, $\phi : \mathcal{V} \to \mathcal{M}$, which are smooth, surjective mappings between Euclidean spaces (e.g. $\mathcal{V}$) to the manifold (denoted by $\mathcal{M}$) (Lezcano-Casado, 2019). Thus, the optimization problem defined in 2 $\max_{W_1,\ldots,W_d \in \mathcal{M}} \mathcal{L}(W_1, \ldots, W_D)$ becomes equivalent to $\max_{y_1,\ldots,y_D \in \mathcal{V}} \mathcal{L}(\phi(y_1), \ldots, \phi(y_D))$ for $\mathcal{M}$ being the Stiefel manifold and ϕ a trivialization as the one described above. Subsequently, we can apply L-BFGS or (stochastic) gradient descent to compute approximate minimizers to our parameter estimation problem. Note that an alternative optimization approach is Riemannian optimization (Sato, 2021; Boumal et al., 2014). Moreover, there are other L-BFGS based methods, such as Ablin et al. (2018) that develop optimization solver for ICA under orthogonal constraints which relies on the Lie exponential map parametrization.

5 Experiments

Preprocessing. Before running any of the ICA-based methods (our or the baselines) we whiten each single view by performing PCA to speed up computation.

Model Implementation. We used the python library `pytorch` (Paszke et al., 2017) to implement our method. We model each view with a separate unmixing matrix. To impose orthogonality constraints on the unmixing matrices, we made use of the `geotorch` library, which is an the extension of `pytorch` (Lezcano-Casado, 2019). The stochastic gradient based method applied for training is L-BFGS.

Initialization. We estimate the mixing matrix up to scale (due to the whitening) and permutation (see Sections 3 and 4). To force the algorithm to output the shared sources in the same order across all views we initialize the unmixing matrices by means of CCA. This follows from the fact that the CCA weights are orthogonal matrices due to whitening, and the transformed views' components are paired and ordered across views such that the first pair correspond to the largest singular value from the singular value decomposition of the cross correlation matrix.

Training. For all conducted experiments we fixed the parameter λ from equation 2 to 1. In the simulated data experiments we conducted each experiment 50 times and based on that we provided error bars in all figures where applicable. For the computational specifics of the real-life data experiment please refer to Section 5.2.

Baselines Implementation. We compare our method to the standard single-view ICA method Infomax (`picard` library (Ablin et al., 2018)). To adopt it to the multi-view setting, we run Infomax on each view separately and then we apply the Hungarian algorithm (Kuhn & Yaw, 1955) to match components from different views based on their cross-correlation.

For the shared response model settings, we compare it to related methods such as MultiViewICA Richard et al. (2020), ShiCA, ShICA-ML Richard et al. (2021), and GroupICA as proposed by Richard et al. (2020). The latter involves a two-step pre-processing procedure, first whitening the data in the single views and then dimensionality reduc-

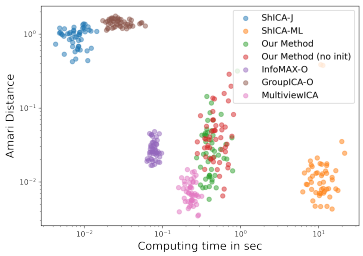
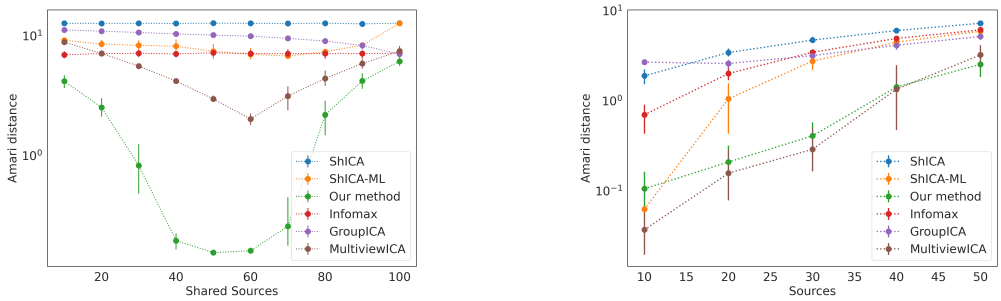


Figure 3: Comparison of the computing time (x-axis) and Amari distance (y-axis). The data is generated according to a shared response model with 5 views, 5 sources and 1000 samples and noise with the variance $\sigma = 1$. The addition "O" to the model's name refers to learning with orthogonal constraints.



(a) Noiseless views

(b) Noisy views

Figure 2: We compare the performance of our method, ShICA, Infomax, GroupICA, MultiViewICA and ShICA-ML Amari distance (the lower the better) for different number of shared sources (x-axis). The error bars correspond to 95% confidence intervals based on 50 independent runs of the experiment. Figure 2a: The datasets come from two different views with total number of sources 100 and sample size 1000. We vary the number of shared sources from 10 to 100. Figure 2b: The data is generated according to a model, where no individual sources are present and the noise per view is uniformly sampled from the interval $[1, 2]$. The number of views is set to 10 and the sample size is 1000. We vary the number of sources from 10 to 50 (y-axis).

tion on the joint views. The code for the latter models is based on `https://github.com/hugorichard/multiviewica` (Richard et al., 2020) and for ShICA-ML `https://github.com/hugorichard/ShICA` (Richard et al., 2021).

For the data integration experiment we use a method based on partial least squares estimation, closely related to CCA, that extracts between-views correlated components and view-specific ones. This method is provided by the `OmicsPLS` R package Bouhaddani et al. (2018) and is especially developed for data integration task for omics data. We refer to this method as PLS. We also make use of IVA-L-SOS (Bhinge et al., 2019) as a representative of the independent vector analysis methods which aligns well with this task. This method assumes a linear noiseless model $x_d = A_d s_d, d \in \{1, \dots, D\}$ with Laplacian independent sources $s_d \in \mathbb{R}^k$ per view where the following between view dependence holds: $(s_{1i}, \dots, s_{Di}) \sim \text{Laplace}(0, \Sigma)$ with s_{di} being the $i-$th component of s_d. The implementation of this method is provided by the `independent-vector-analysis` package provided by (Lehmann et al., 2022).

5.1 Synthetic Experiments

Data Simulation. We simulated the data using the Laplace distribution $\exp(-\frac{1}{2}|x|)$, and the mixing matrices are sampled with normally distributed entries with mean 1 and 0.1 standard deviation. The realizations of the observed views are obtained according to the proposed model. In the different scenarios described below we vary the noise distribution.

Evaluation. The quality of the mixing matrix estimation is measured with the Amari distance (Amari et al., 1995), which cancels if the estimated matrix differs from the ground truth one up to scale and permutation. More experiments than the one stated below are provided in Appendix D.2

Noiseless views. In Figure 2a, we consider a noiseless view setting, where we fixed the dimension to be 100 and we vary the number of shared sources from 10 to 100 in a two view setting. We can see that as soon as the ratio of shared sources to individual sources gets around 1:1 we can recover almost all shared and individual sources compared to the baseline methods which cannot perform well in higher dimensions. The reason of the performance drop of our method is insufficient number of samples for the learning task. See Appendix D.2 for additional experiments in the noisy case.

Robustness to model misspecification in a shared response model application. Here we apply our method to a shared response setting, i.e. no individual sources are available. For this experiment the views have view-specific variance uniformly sampled from $[1, 2]$. Our method shows similar performance to the competitor model MultiViewICA in this special (see Figure 2b) .

Table 1: Shared Sources Recovery Performance

Views	$\sigma = 0.1$	$\sigma = 0.5$	$\sigma = 1$	$\sigma = 2$
2	0.975 ± 0.002	0.940 ± 0.003	0.812 ± 0.003	0.281 ± 0.01
5	$\mathbf{0.981 \pm 0.001}$	0.967 ± 0.001	0.922 ± 0.003	0.5 ± 0.04
10	0.980 ± 0.002	$\mathbf{0.972 \pm 0.002}$	$\mathbf{0.95 \pm 0.003}$	$\mathbf{0.741 \pm 0.08}$

Computing time. Figure 3 the computing times of response models trained on the same datasets. The experiment was carried out on a local machine using 8 CPU cores in parallel. It seems that MultiViewICA has a similar performance as ShICA-ML but much faster. Our method is faster than ShICA-ML and shows similar mean perfromance as the Infomax method.

Mean cross correlation (MCC) of shared sources. We explore how well our method recovers shared sources. We estimate them by taking their average across views and compute the mean cross correlation (MCC) between the estimates and the ground truth. That involves matching estimated with the ground truth components by using the Hungarian algorithm and then computing the mean over all correlations between the aligned pairs. In this experiment, we fixed the total number of sources to be 60 and the shared to be 30. We investigated four cases corresponding to a different noise standard deviation $\sigma = 0.1, 0.5, 1, 2.$ and reported the mean MMC and its standard deviation from 50 runs in Table 1. As expected by increasing the number of views we get better estimates of the shared sources, i.e. the MMC score increases.

5.2 Data Fusion of Transcriptome Data

Background. Transcriptome datasets are relevant for the field of genomics. After preprocessing they have the form of random data matrices, where each row correspond to a gene and each column refers to an experiment. Based on these datasets, scientists try to infer gene-gene interactions in the genome.

Co-regulation Inference. Since the transcriptome datasets are in the high-dimension-low-sample-size regime (number of genes>number of samples), usually graphical lasso (Friedman et al., 2007) is well-suited for inferring graphical structure from the observed data. More precisely in this application, we want to estimate an undirected graph with nodes referring to the genes and with edges connecting genes with a common regulator.

Data Integration Task. To boost the graphical lasso performance, we would like to combine as many experiments as possible. Since the datasets usually come from different labs there is non-biological noise present. Therefore, just pooling the two datasets together for performing downstream tasks can lead to suboptimal results. The goal of the data integration task is to "denoise" the datasets, such that the transformed data can be used as samples for the graph inference task. In this kind of application usually a dimensionality reduction step is required. This is done by a dimensionality reduction step followed by a feature extracting algorithm, such as IVA-L-SOS, or combined dimensionality reduction method and a feature extractor such as PLS. We follow the latter approach for both, our method and Infomax, and directly optimize for unmixing matrices W_d of dimension $c_d \times k_d$ with $c_d \leq k_d$ for $d = 1, \ldots, D$. That means that by minimizing the loss function with respect to possibly non-square $W_1, \ldots, W_D$ we directly extract c_d features, which are then used for the graph inference task.

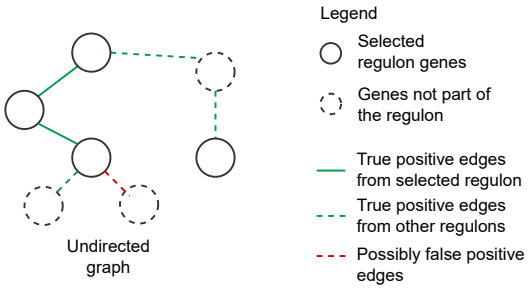
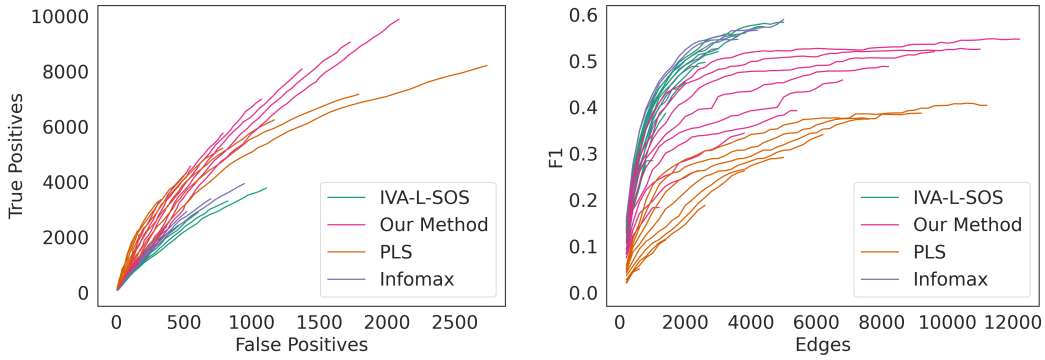


Figure 4: Induced subgraph with vertices from a selected regulon (in the main text denoted by R) (not dotted nodes) and its neighbors (dotted nodes). The green edges belong to the ground truth and the red one are possibly false positives.



(a) True Positive vs False Positive pairs

(b) F1 score

Figure 5: After the proposed hyperparameter tuning, we compare the top ten EBIC glasso models with our model, PLS, Infomax, and IVA-L-SOS. We order the edges from the selected networks according to their strength. In Figure 5a we count the true positives (y-axis) and possibly false positive edges in the first 100, 200,... edges (x-axis). It seems that our method and PLS outperform the other two methods, and for our method, the true positive/false positive rate increases faster than for PLS. In Figure 5b, we compute the average regulon F1 score (y-axis) from the selected edges (x-axis). Our method seems to perform better than PLS but worse than IVA-L-SOS and Infomax. That might result from the proposed F1 score prioritizing sparse graphs.

Data Assumptions. We do a one-to-one translation to our proposed model by assuming that each experiment is a noisy linear combination of independent gene pathways and that some pathways are active for both datasets (shared sources).

Datasets. In this example, we consider the bacterium *B. subtilis*, for which a very rich collection of the discovered gene-gene interactions are publicly available, which we use as our ground truth model. For this data integration task we use two vast publicly available datasets (Arrieta-Ortiz et al., 2015; Nicolas et al., 2012). Each of the datasets contain gene expression levels of about 4000 genes measured across more than 250 experimental outcomes. For detailed description of the datasets we refer to Appendix C.

Experiment. As in most real-life applications, the number of latent sources per dataset is unknown. We treat it as a hyperparameter for each model, i.e., we perform grid search on $\{50, 60, 70, \ldots, 200\}^2$, for the total number of sources for both datasets. Note that for IVA-L-SOS the number of sources in both datasets should coincide. The number of shared sources for our method and PLS varies between $10, 20, 30$, and 40. We fit 30 graphical lasso models for different penalization parameters on the estimated components. We select the top 10 models by employing a statistical goodness-of-fit measure, called EBIC, for each combination of hyperparameters (see Appendix C for more details). Then, the hyperparameter setting is selected, yielding the best true positive/false positive ratio curves (as the ones shown in Figure 5a). The resulting hyperparameter settings are IVA-L-SOS (130 latent sources), Our Method (50 for dataset 1, 60 for dataset 2, 40 shared sources), PLS (180 for dataset 1, 80 for dataset 2, 10 shared sources), and Infomax (200 for dataset 1, 50 for dataset 2).

True Positives vs False Positives. The below-described evaluation is used for our hyperparameter selection. The output graph from the graphical lasso for each pre-processing method is compared to the ground truth one. The evaluation strategy is as follows. For each estimated graph, we order the edges according to their strength. Then we count the true positive and false positive edges in the first $100, 200, \ldots$ edges. Then for each method separately, we select the hyperparameter combination for which the graphical lasso has the best true positive/false positive ratio curves. The results are depicted in Figure 5, where the best models for each method are compared. We can conclude that our model boosts the graphical lasso's performance compared to the others. We also run the graphical lasso on the pooled data without any pre-processing. Surprisingly, the method outputs an empty graph, i.e., the goodness-of-fit measure we use evaluates the empty graph as the best model describing the data.

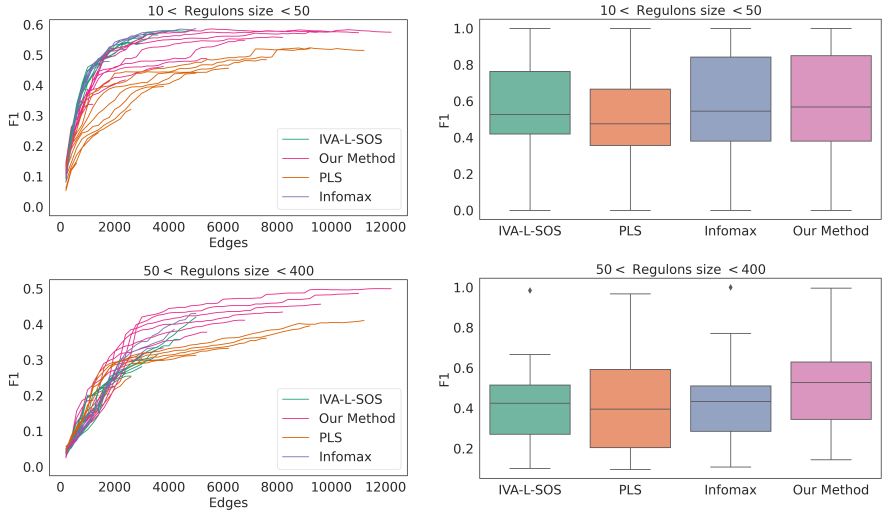


Figure 6: F1 score curves per model (left) and best model F1 regulon performance (right) for two groups of regulons depending on their size (between 10 and 50, and between 50 and 400). In the first group all methods show comperable results. In the second case (regulon size > 50) it seems that our method shows better F1 score for the large regulons.

F1 score. We are also interested if the output graph resembles the known regulon structure. A regulon is a set of genes that are controlled as a unit by the same regulator. The species of interest has 225 known regulons, which form cliques in our ground truth graph. We compute the precision and recall per regulon from the graphical lasso output in the following way. First, we select a subgraph induced by the regulon genes denoted by R and their neighbors as depicted in Figure 4. The precision is the ratio of the true positive edges and the total number of edges in the selected subgraph. The recall is the ratio of the number of nodes in the biggest connected component with vertices in R and the cardinality of R. Note that the recall score does not incorporate any information about the connectivity of the selected subgraph. In Figure 5b we order the edges according to their strength and from the first $100, 200, \ldots$ edges we the F1 score per regulon from the corresponding precision and recall. It seems that IVA-L-SOS and Infomax outperform the other methods. The reason is that the output graphs are sparser in both cases and recover smaller regulons better (most of the regulons have size < 10). Due to the overlapping structure of regulons, i.e. larger regulons encompass smaller ones, we want to compare the aggregated F1 scores for the larger regulons. From Figure 6, we can see that our method can recover very large regulons with higher F1 scores than the other methods. In medium size regulon group Infomax, IVA-L-SOS and our method show comparable results.

6 Discussion

We proposed a novel noisy linear ICA approach that utilizes the prior knowledge that the different views share information to infer both shared and view-specific sources. Compared to other models from related fields, our model does not assume that all views have the same dimensionality. We provided theoretical guarantees for the identifiability of the model's linear structure and the latent sources in distribution. We adopted a maximum likelihood strategy for estimating the unmixing matrices by maximizing the joint log-likelihood of the observed views. Our empirical results showed that our model performs well in high-dimensional data regimes that often resemble real-life applications. We also suggested a novel strategy for combining transcriptome data and empirically showed that our model improves the performance of a graphical inference model chosen for the particular task. In future work, we would like to address some possible extensions to our model, such as the case when the noise has an arbitrary covariance matrix. Moreover, to state the identifiability results of our model, we assumed that the individual sources from the different views are mutually independent. That might not be the case in real-life applications like the one considered above.

7 ETHICS STATEMENT

Our model can be used in many scientific domains such as neuroscience, genomics, physics, medicine, causal discovery, etc., where the proposed model assumptions hold. Therefore, its societal impact is tied to the ethical concerns of the domains it is applied to.

8 REPRODUCIBILITY STATEMENT

The code and transcriptome data and ground truth model are in process of releasing. We provided detailed proofs of the theorem and other theoretical results stated in the main paper. Furthermore, all synthetic experiments are run multiple times with different random seeds and the reported results are in terms of mean and 95% confidence interval. Additional information regarding the data integration experiment, such as pseudo code of algorithm and explanation of the main steps are provided in the appendix.

REFERENCES

Pierre Ablin, Jean-François Cardoso, and Alexandre Gramfort. Faster ica under orthogonal constraint. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 4464–4468. IEEE, 2018.

Shun-ichi Amari, Andrzej Cichocki, and Howard Yang. A new learning algorithm for blind signal separation. *Advances in neural information processing systems*, 8, 1995.

Matthew Anderson, Tuelay Adali, and Xi-Lin Li. Joint blind source separation with multivariate gaussian model: Algorithms and performance analysis. *IEEE Transactions on Signal Processing*, 60(4):1672–1683, 2011.

Matthew Anderson, Geng-Shen Fu, Ronald Phlypo, and Tülay Adalı. Independent vector analysis: Identification conditions and performance bounds. *IEEE Transactions on Signal Processing*, 62 (17):4399–4410, 2014.

Galen Andrew, Raman Arora, Jeff Bilmes, and Karen Livescu. Deep canonical correlation analysis. In *International conference on machine learning*, pp. 1247–1255. PMLR, 2013.

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.

Francis R Bach and Michael I Jordan. Kernel independent component analysis. *Journal of machine learning research*, 3(Jul):1–48, 2002.

Francis R Bach and Michael I Jordan. A probabilistic interpretation of canonical correlation analysis. 2005.

Paolo Bartolomeo, Tal Seidel Malkinson, and Stefania De Vito. Botallo's error, or the quandaries of the universality assumption. *Cortex*, 86:176–185, 2017.

Suchita Bhinge, Rami Mowakeaa, Vince D Calhoun, and Tülay Adalı. Extraction of time-varying spatiotemporal networks using parameter-tuned constrained iva. *IEEE transactions on medical imaging*, 38(7):1715–1725, 2019.

Said el Bouhaddani, Hae-Won Uh, Geurt Jongbloed, Caroline Hayward, Lucija Klarić, Szymon M Kiełbasa, and Jeanine Houwing-Duistermaat. Integrating omics datasets with the omicspls package. *BMC bioinformatics*, 19(1):1–9, 2018.

Nicolas Boumal, Bamdev Mishra, P-A Absil, and Rodolphe Sepulchre. Manopt, a matlab toolbox for optimization on manifolds. *The Journal of Machine Learning Research*, 15(1):1455–1459, 2014.

Vince D Calhoun, Tulay Adali, Godfrey D Pearlson, and James J Pekar. A method for making group inferences from functional mri data using independent component analysis. *Human brain mapping*, 14(3):140–151, 2001.

Jiahua Chen and Zehua Chen. Extended bayesian information criteria for model selection with large model spaces. *Biometrika*, 95(3):759–771, 2008.

Pierre Comon. Independent component analysis, a new concept? *Signal processing*, 36(3):287–314, 1994.

Mike Davies. Identifiability issues in noisy ica. *IEEE Signal processing letters*, 11(5):470–473, 2004.

Julien Dubois and Ralph Adolphs. Building a science of individual differences from fmri. *Trends in cognitive sciences*, 20(6):425–443, 2016.

Yasser El-Manzalawy. Cca based multi-view feature selection for multi-omics data integration. In *2018 IEEE Conference on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB)*, pp. 1–8. IEEE, 2018.

Astrid ME Engberg, Kasper W Andersen, Morten Mørup, and Kristoffer H Madsen. Independent vector analysis for capturing common components in fmri group analysis. In *2016 international workshop on pattern recognition in neuroimaging (prni)*, pp. 1–4. IEEE, 2016.

Marco Federici, Anjan Dutta, Patrick Forré, Nate Kushman, and Zeynep Akata. Learning robust representations via multi-view information bottleneck. *ICLR*, 2020.

Rina Foygel and Mathias Drton. Extended bayesian information criteria for gaussian graphical models. *arXiv preprint arXiv:1011.6640*, 2010.

Jerome Friedman, Trevor Hastie, and Robert Tibshirani. Sparse inverse covariance estimation with the graphical lasso. *Biostatistics*, 9(3):432–441, 12 2007. ISSN 1465-4644. doi: 10.1093/biostatistics/kxm045. URL `https://doi.org/10.1093/biostatistics/kxm045`.

Ying Guo and Giuseppe Pagnoni. A unified framework for group independent component analysis for multi-subject fmri data. *NeuroImage*, 42(3):1078–1093, 2008.

Harold Hotelling. Relations between two sets of variates. In *Breakthroughs in statistics*, pp. 162–190. Springer, 1936.

Maxim (https://math.stackexchange.com/users/491644/maxim). Multivariate transformation formula correct? Mathematics Stack Exchange, 2019. URL `https://math.stackexchange.com/q/3325459`. URL:https://math.stackexchange.com/q/3325459 (version: 2019-08-17).

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

Konrad J Karczewski and Michael P Snyder. Integrative omics for health and disease. *Nature Reviews Genetics*, 19(5):299–310, 2018.

Arto Klami, Seppo Virtanen, and Samuel Kaski. Bayesian canonical correlation analysis. *Journal of Machine Learning Research*, 14(4), 2013.

H. W. Kuhn and Bryn Yaw. The hungarian method for the assignment problem. *Naval Res. Logist. Quart*, pp. 83–97, 1955.

Jong-Hwan Lee, Te-Won Lee, Ferenc A Jolesz, and Seung-Schik Yoo. Independent vector analysis (iva): multivariate approach for fmri group study. *Neuroimage*, 40(1):86–109, 2008.

Isabell Lehmann, Evrim Acar, Tanuj Hasija, M.A.B.S. Akhonda, Vince D. Calhoun, Peter J. Schreier, and Tulay Adali. Multi-task fmri data fusion using iva and parafac2. 2022.

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

Mario Lezcano-Casado and David Martınez-Rubio. Cheap orthogonal constraints in neural networks: A simple parametrization of the orthogonal and unitary group. In *International Conference on Machine Learning*, pp. 3794–3803. PMLR, 2019.

Ana S Lukic, Miles N Wernick, Lars Kai Hansen, and Stephen C Strother. An ica algorithm for analyzing multiple data sets. In *Proceedings. International Conference on Image Processing*, volume 2, pp. II–II. IEEE, 2002.

Paul Lévy. Propriétés asymptotiques des sommes de variables aléatoires indépendantes ou enchaînées. *The Annals of Probability*, 1(1):9–18, 1935. ISSN 00911798. URL `http://www.jstor.org/stable/2959343`.

Mona Maneshi, Shahabeddin Vahdat, Jean Gotman, and Christophe Grova. Validation of shared and specific independent component analysis (ssica) for between-group comparisons in fmri. *Frontiers in neuroscience*, 10:417, 2016.

Martin J McKeown and Terrence J Sejnowski. Independent component analysis of fmri data: examining the assumptions. *Human brain mapping*, 6(5-6):368–372, 1998.

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.

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.

Nuria Planell, Vincenzo Lagani, Patricia Sebastian-Leon, Frans van der Kloet, Ewoud Ewing, Nestoras Karathanasis, Arantxa Urdangarin, Imanol Arozarena, Maja Jagodic, Ioannis Tsamardinos, et al. Stategra: multi-omics data integration–a conceptual scheme with a bioinformatics pipeline. *Frontiers in genetics*, 12:143, 2021.

Hugo Richard, Luigi Gresele, Aapo Hyvarinen, Bertrand Thirion, Alexandre Gramfort, and Pierre Ablin. Modeling shared responses in neuroimaging studies through multiview ica. *Advances in Neural Information Processing Systems*, 33:19149–19162, 2020.

Hugo Richard, Pierre Ablin, Bertrand Thirion, Alexandre Gramfort, and Aapo Hyvarinen. Shared independent component analysis for multi-subject neuroimaging. *Advances in Neural Information Processing Systems*, 34:29962–29971, 2021.

Hiroyuki Sato. *Riemannian Optimization and Its Applications*. Springer, 2021.

Mohamed L Seghier and Cathy J Price. Interpreting and utilising intersubject variability in brain function. *Trends in cognitive sciences*, 22(6):517–530, 2018.

Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive multiview coding. In *European conference on computer vision*, pp. 776–794. Springer, 2020.

Caroline Uhler and GV Shivashankar. Machine learning approaches to single-cell data integration and translation. *Proceedings of the IEEE*, 2022.

Gaël Varoquaux, Sepideh Sadaghiani, Jean Baptiste Poline, and Bertrand Thirion. Canica: Model-based extraction of reproducible group-level ica patterns from fmri time series. *arXiv preprint arXiv:0911.4650*, 2009.

Javier Vía, Matthew Anderson, Xi-Lin Li, and Tülay Adalı. A maximum likelihood approach for independent vector analysis of gaussian data sets. In *2011 IEEE International Workshop on Machine Learning for Signal Processing*, pp. 1–6. IEEE, 2011.

Ricardo Vigário, Veikko Jousmäki, Matti Hämäläinen, Riitta Hari, and Erkki Oja. Independent component analysis for identification of artifacts in magnetoencephalographic recordings. *Advances in neural information processing systems*, 10, 1997.

Seppo Virtanen. *Bayesian exponential family projections*. PhD thesis, Aalto University, 2010.

Weiran Wang, Xinchen Yan, Honglak Lee, and Karen Livescu. Deep variational canonical correlation analysis. *arXiv preprint arXiv:1610.03454*, 2016.

Hejia Zhang, Po-Hsuan Chen, Janice Chen, Xia Zhu, Javier S Turek, Theodore L Willke, Uri Hasson, and Peter J Ramadge. A searchlight factor model approach for locating shared information in multi-subject fmri analysis. *arXiv preprint arXiv:1609.09432*, 2016.

A IDENTIFIABILITY RESULTS

Theorem A.1 ((Kagan et al., 1973, Theorem 10.3.1))**.** *Let $x = Af$ and $x = Bg$ be two representations of a p-dimensional random vector x, where A and B are (non-random) matrices of orders $p \times r$ and $p \times s$ respectively, and $f = (f_1, \dots, f_r)$ and $g = (g_1, \dots, g_s)$ are random vectors with non-degenerate (i.e. non-constant) independent components. Then the following assertions hold.*

1. *If the i-th column of A is not proportional to any column of B, then f_i is normal.*
2. *If the i-th column of A is proportional to the j-th column of B, then the logarithms of the characteristic functions (c.f.) of f_i and g_j differ by a polynomial in a neighborhood of the origin.*

Theorem A.2 (Identifiability of the single view ICA model 1)**.** *Let $x \in \mathbb{R}^p$ be a random variable. Assume that we have the following two representations of x:*

$$A^{(1)}(y^{(1)} + \epsilon^{(1)}) + b^{(1)} = x = A^{(2)}(y^{(2)} + \epsilon^{(2)}) + b^{(2)}, \tag{3}$$

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

1. *$A^{(i)} \in \mathbb{R}^{p \times k^{(i)}}$ is a (non-random) matrix with full column rank, i.e. $\operatorname{rank}(A^{(i)}) = k^{(i)} \leq p$,*
2. *$b^{(i)} \in \mathbb{R}^p$ a (non-random) column vector,*
3. *$\epsilon^{(i)} \in \mathbb{R}^{k^{(i)}}$ is an uncorrelated k-variate normal random variable: $\epsilon^{(i)} \sim \mathcal{N}(\mu^{(i)}, \Sigma^{(i)})$, with mean $\mu^{(i)} \in \mathbb{R}^{k^{(i)}}$ and a positive-definite diagonal covariance matrix $\Sigma^{(i)} \in \mathbb{R}^{k^{(i)} \times k^{(i)}}$,*
4. *$y^{(i)} \in \mathbb{R}^{k^{(i)}}$ is a random variable such that:*
   (a) *its $k^{(i)}$-components $\{y_1^{(i)}, \dots, y_{k^{(i)}}^{(i)}\}$ are mutually independent,*
   (b) *each of its component $y_j^{(i)}$ is a non-constant random variable (a.s.), $j = 1, \dots, k^{(i)}$,*
   (c) *$y^{(i)}$ has no normal components, i.e. if we can write: $y^{(i)} \sim \tilde{y}^{(i)} + \hat{y}^{(i)}$ with $\tilde{y}^{(i)} \perp\!\!\!\perp \hat{y}^{(i)}$, then $\tilde{y}^{(i)}$ and $\hat{y}^{(i)}$ are non-normal,*
5. *$\epsilon^{(i)}$ is independent from $y^{(i)}$: $\epsilon^{(i)} \perp\!\!\!\perp y^{(i)}$.*

Then $k^{(1)} = k^{(2)} =: k$ and there exist a permutation matrix $P \in \mathbb{R}^{k \times k}$, an invertible diagonal matrix $\Lambda \in \mathbb{R}^{k \times k}$ and a column vector $c \in \mathbb{R}^k$ such that:

$$A^{(2)} = A^{(1)} P \Lambda,$$

and such that the corresponding random variables have the same distributions:

$$P\Lambda y^{(2)} + c \sim y^{(1)}, \qquad P\Lambda(\epsilon^{(2)} - \mu^{(2)}) \sim \epsilon^{(1)} - \mu^{(1)}, \qquad P\Lambda\Sigma^{(2)}\Lambda^\top P^\top = \Sigma^{(1)}.$$

Proof. 1. In the first part of our proof we show that $k^{(1)} = k^{(2)} =: k$ and $A^{(2)} = A^{(1)} P \Lambda$ for some permutation matrix $P \in \mathbb{R}^{k \times k}$, an invertible diagonal matrix $\Lambda \in \mathbb{R}^{k \times k}$.

First, for $i = 1, 2$ we state an equivalent formulation of the linear representation of x given in 3. According to (Kagan et al., 1973, Lemma 10.2.3), there exist a constant column vector $c^{(2)} \in \mathbb{R}^{k^{(2)}}$ such that $b^{(2)} - b^{(1)} = A^{(2)} c^{(2)}$. It follows that $\tilde{x} = x - b^{(1)} = A^{(1)}(y^{(1)} + \epsilon^{(1)}) = A^{(2)}(y^{(2)} + \epsilon^{(2)} + c^{(2)})$.

Furthermore, note that if $y^{(i)}$ is non-normal, then the random variables $g^{(1)} = y^{(1)} + \epsilon^{(1)}$ and $g^{(2)} = y^{(2)} + \epsilon^{(2)} + c^{(2)}$ are also non-normal. This follows from the fact that if $g^{(i)}$ is normal then both $y^{(i)}$ and $\epsilon^{(i)}$ would be normal according to the Lévy-Cramér theorem (Lévy, 1935).

Thus, we can apply Theorem A.1 for the two representations of $\tilde{x}$, $\tilde{x} = A^{(1)} g^{(1)}$ and $\tilde{x} = A^{(2)} g^{(2)}$. Since every component of $g^{(i)}$ is non-normal, it follows that every column of $A^{(1)}$ is proportional to a column of $A^{(2)}$ and vice versa.

Now assume w.l.o.g that $k^{(1)} > k^{(2)}$. Then, there exist two columns of $A^{(1)}$ that are proportional to a column of $A^{(2)}$. However, this is a contradiction to assumption 1. that the matrix $A^{(1)}$ has full column rank.

Thus, it follows that $k^{(1)} = k^{(2)} =: k$ and $A^{(2)} = A^{(1)} P \Lambda$ for some permutation matrix $P \in \mathbb{R}^{k \times k}$, an invertible diagonal matrix $\Lambda \in \mathbb{R}^{k \times k}$. Moreover,

$$A^{(1)}(y^{(1)} + \epsilon^{(1)}) = A^{(1)} P \Lambda (y^{(2)} + \epsilon^{(2)} + c^{(2)}).$$

Multiplying with $(A^{(1),\top} A^{(1)})^{-1} A^{(1),\top}$, which gives:

$$y^{(1)} + \epsilon^{(1)} = P \Lambda (y^{(2)} + \epsilon^{(2)} + c^{(2)}).$$

2. In the remaining we show that there exist a column vector c such that $y^{(1)} \sim P\Lambda(y^{(2)} + c^{(2)}) + c$ and $\epsilon^{(1)} - \mu^{(1)} \sim P\Lambda(\epsilon^{(2)} - \mu^{(2)})$ (or equivalently $\Sigma^{(1)} = P\Lambda\Sigma^{(2)}\Lambda^\top P^\top$). Now, define $\tilde{y}^{(2)} = P\Lambda y^{(2)}$, $\tilde{c}^{(2)} = P\Lambda c^{(2)}$ and $\tilde{\epsilon}^{(2)} = P\Lambda\epsilon^{(2)}$ which is normally distributed with mean $\tilde{\mu^{(2)}} = P\Lambda\mu^{(2)}$ and a diagonal covariance matrix $\tilde{\Sigma}^{(2)} = P\Lambda\Sigma^{(2)}\Lambda^\top P^\top$.

Define the characteristic functions of $y^{(1)}, \tilde{y}^{(2)}, \epsilon^{(1)}, \tilde{\epsilon}^{(2)}$ as $\phi_{y^{(1)}}(\cdot), \phi_{\tilde{y}^{(2)}}(\cdot), \phi_{\epsilon^{(1)}}(\cdot), \phi_{\tilde{\epsilon}^{(2)}}(\cdot) : \mathbb{R}^k \to \mathbb{R}$, from assumption 5. it follows that

$$\phi_{\epsilon^{(1)}}(t)\phi_{y^{(1)}}(t) = e^{it^\top \tilde{c}^{(2)}} \phi_{\tilde{\epsilon}^{(2)}}(t)\phi_{\tilde{y}^{(2)}}(t)$$
$$\phi_{\epsilon^{(1)}}(t)\prod_{i=1}^{k}\phi_{y_i^{(1)}}(t_i) = e^{it^\top \tilde{c}^{(2)}} \phi_{\tilde{\epsilon}^{(2)}} \prod_{i=1}^{k}\phi_{\tilde{y}_i^{(2)}}(t_i)$$

The last equation follows from assumption 4a. Now set $t_i = 0$ for all $i \neq 1$. We get for all t_1

$$\exp(it_1\mu_1^{(1)} - \Sigma_{11}^{(1)} t_1^2)\phi_{y_1^{(1)}}(t_1) = \exp(it_1\tilde{c}_1^{(2)})\exp(it_1\tilde{\mu}_1^{(2)} - \tilde{\Sigma}_{11}^{(2)} t_1^2)\phi_{\tilde{y}_1^{(2)}}(t_1).$$

W.l.o.g. we assume $0 < \Sigma_{11}^{(1)} < \tilde{\Sigma}_{11}^{(2)}$. Thus, the characteristic function given by $\exp(-(\tilde{\Sigma}_{11}^{(2)} - \Sigma_{11}^{(1)})t_1^2)$ is a well defined characteristic function of a normally distributed random variable with mean 0 and variance $\tilde{\Sigma}_{11}^{(2)} - \Sigma_{11}^{(1)}$. Then, the characteristic function of $y_1^{(1)}$ is proportional to a product of the characteristic functions of $\tilde{y}_1^{(2)}$ and a Gaussian random variable. This is a contradiction to the assumption that $y_1^{(1)}$ does not have a normal component (assumption 4c). It follows that, $\Sigma_{11}^{(1)} = \tilde{\Sigma}_{11}^{(2)}$ and for all $t_1 \in \mathbb{R}$ $\phi_{y_1^{(1)}}(t_1) = \exp it_1(\tilde{c}_1^{(2)} + \tilde{\mu}_1^{(2)} - \mu_1^{(1)})\phi_{\tilde{y}_1^{(2)}}(t_1)$, i.e. $\tilde{y}_1^{(2)} + c_1 \sim y_1^{(1)}$ where $c_1 = \tilde{c}_1^{(2)} + \tilde{\mu}_1^{(2)} - \mu_1^{(1)}$. The remaining statements can be proven analogously.

□

Theorem A.3. *Let $x_1, \ldots, x_D$ for $D \geq 2$ be random vectors with the following two representations:*

$$A_d^{(1)}\Big(\begin{bmatrix} s_0^{(1)} \\ s_d^{(1)} \end{bmatrix} + \epsilon_d^{(1)}\Big) = x_d = A_d^{(2)}\Big(\begin{bmatrix} s_0^{(2)} \\ s_d^{(2)} \end{bmatrix} + \epsilon_d^{(2)}\Big), \qquad d \in \{1, \ldots, D\},$$

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

1. *$A_d^{(i)} \in \mathbb{R}^{p_d \times k_d^{(i)}}$ is a (non-random) matrix with full column rank, i.e. $\operatorname{rank}(A_d^{(i)}) = k_d^{(i)}$,*
2. *$\epsilon_d^{(i)} \in \mathbb{R}^{k_d^{(i)}}$ and $\epsilon_d^{(i)} \sim \mathcal{N}(0, \sigma_d^{(i)2}\mathbb{I}_{k_d^{(i)}})$ is a $k_d^{(i)}$-variate normal random variable,*
3. *$\tilde{s}_d^{(i)} = (s_0^{(i)\top}, s_d^{(i)\top})^\top$ with $s_0^{(i)} \in \mathbb{R}^{c^{(i)}}$ and $s_d^{(i)} \in \mathbb{R}^{k_d^{(i)} - c^{(i)}}$ is a random vector such that:*

(a) the components of $\tilde{s}_d^{(i)}$ are mutually independent and each of them is a.s. a non-constant random variable,

(b) $\tilde{s}_d^{(i)}$ is non-normal with 0 mean and unit variance.

4. $\epsilon_d^{(i)}$ is independent from $s_0^{(i)}$ and $s_d^{(i)}$: $\epsilon_d^{(i)} \perp\!\!\!\perp s_0^{(i)}$ and $\epsilon_d^{(i)} \perp\!\!\!\perp s_d^{(i)}$.

Then, $c^{(1)} = c^{(2)} =: c$ and for all $d = 1, \dots, D$ we get that $k_d^{(1)} = k_d^{(2)} =: k_d$, and there exist a sign matrix Γ_d and a permutation matrix $P_d \in \mathbb{R}^{k_d \times k_d}$ such that:

$$A_d^{(2)} = A_d^{(1)} P_d \Gamma_d,$$

and furthermore the sources and noise distributions are identifiable, i.e.

$$\begin{bmatrix} s_0^{(2)} \\ s_d^{(2)} \end{bmatrix} \sim \Gamma_d^{-1} P_d^{-1} \begin{bmatrix} s_0^{(1)} \\ s_d^{(1)} \end{bmatrix}, \qquad \sigma_d^{(2)} = \sigma_d^{(1)}.$$

Proof. First, we can directly apply Theorem A.2 to each single view $d, d \in \{1, \dots, D\}$ which ensures the identifiability of the mixing matrices up to permutation and scaling, i.e. there exist a permutation matrix P_d and an invertible diagonal matrix Λ_d such that $A_d^{(2)} = A_d^{(1)} P_d \Lambda_d$ and $\operatorname{rank}(A_d^{(2)}) = \operatorname{rank}(A_d^{(1)}) = k_d$.

W.l.o.g., let $c^{(1)} > c^{(2)}$. That means that the shared sources in representation (1) are more that the ones in representation (2). It follows according to Theorem A.1, that there exist a component of the shared sources from (1) and an individual component from (2) in every view such that they are both proportional. More precisely, for any $d \in \{1, \dots, D\}$ there exist $k, l \in \{1, \dots, k_d\}$ such that $s_{0k}^{(1)}$ is a component of the shared sources $s_0^{(1)}$ and $s_{dl}^{(2)}$ is a component from the individual sources $s_d^{(2)}$ such that $s_{0k}^{(1)} + \epsilon_{d0k}^{(1)} = (\Lambda_d)_{ll}(s_{dl}^{(2)} + \epsilon_{d1l}^{(2)})$. Let $r \neq d$ be another view such that there exist $m \in \{1, \dots, k_r\}$ with $s_{mr}^{(2)}$ being an individual component and $s_{0k}^{(1)} + \epsilon_{r0k}^{(1)} = (\Lambda_d)_{mm}(s_{rm}^{(2)} + \epsilon_{r1m}^{(2)})$. This is contradiction to the assumption that $s_{rm}^{(2)} \perp\!\!\!\perp s_{dl}^{(2)}$. It follows that $c^{(1)} = c^{(2)}$.

Furthermore, $\operatorname{Var}(x_d) = \sigma_d^{(1)2} A_d^{(1)} A_d^{(1),\top} = \sigma_d^{(2)2} A_d^{(2)} A_d^{(2),\top} = \sigma_d^{(2)2} A_d^{(1)} P_d \Lambda_d^2 P_d^\top A_d^{(1),\top}$. Multiplying with $A_d^{(1),\dagger} = (A_d^{(1)\top} A_d^{(1)})^{-1} A_d^{(1)\top}$ from left and $A_d^{(1),\dagger,\top} = A_d^{(1)} (A_d^{(1)\top} A_d^{(1)})^{-1}$ from right yields $\sigma_d^{(1)2} \mathbb{I}_{k_d} = \sigma_d^{(2)2} P_d \Lambda_d^2 P_d^\top$. It follows that $\frac{\sigma_d^{(2)2}}{\sigma_d^{(1)2}} \Lambda_d^2 = \mathbb{I}_{k_d}$. Computing the covariance between two different views $d, l \in \{1, \dots, D\}$ gives

$$\operatorname{Cov}(x_d, x_l) = A_{d0}^{(1)} A_{l0}^{(1),\top} = A_{d0}^{(2)} A_{l0}^{(2),\top} = A_{d0}^{(1)} \Lambda_d[c,c] \Lambda_l[c,c] A_{l0}^{(1),\top}$$

where $\Lambda_d[c,c]$ is an invertible diagonal matrix composed by the first c columns and rows of the matrix Λ_d. By multiplying with the left-inverse of $A_{d0}^{(1)}$ from the left and right-inverse of $A_d^{(1),\top}$ from the right, we get for any d and l $\Lambda_d[c,c]\Lambda_l[c,c] = \mathbb{I}_c$. It follows that all entries of Λ_d equal 1 or -1 and therefore $\frac{\sigma_d^{(2)2}}{\sigma_d^{(1)2}} = 1$ for every d.

In the remaining, we will show that the distribution of the sources is identifiable even in the cases when they have normal components. Let $s_i^{(1)}$ be component from $\tilde{s}_i^{(1)}$. Furthermore, there exist $j \in \{1, \dots, k_d\}$ such that $s_i^{(1)} + \epsilon_i^{(1)} = s_j^{(2)} + \epsilon_j^{(2)}$. Taking the characteristic functions from both sides yields

$$\phi_{s_i^{(1)}}(t) \phi_{\epsilon_i^{(1)}}(t) = \phi_{s_j^{(2)}}(t) \phi_{\epsilon_j^{(2)}}(t)$$

Since $\sigma_d^{(1)2} = \sigma_d^{(2)2}$ and the noise and sources are with 0 mean, the above equation simplifies to $\phi_{s_i^{(1)}}(t) = \phi_{s_j^{(2)}}(t)$, i.e. $\phi_{s_i^{(1)}}(t) \sim \phi_{s_j^{(2)}}(t)$. □

Theorem A.4. *Let $x_1, \dots, x_D$ for $D \geq 3$ be random vectors which are generated according to the model defined in 1. Furthermore, we assume that we have the following two representations of $x_1, \dots, x_D$ according to 1:*

$$A_{d0}^{(1)} s_0^{(1)} + A_{d1}^{(1)} s_d^{(1)} + A_d^{(1)} \epsilon_d^{(1)} = x_d = A_{d0}^{(2)} s_0^{(2)} + A_{d1}^{(2)} s_d^{(2)} + A_d^{(2)} \epsilon_d^{(2)}, \qquad d \in \{1, \dots, D\},$$

Additionally, to the assumptions of 1 it holds that

1. *each of the components* $s_{dj}^{(i)}$ *of* $s_d^{(i)}$ *for* $j = 1, \dots, k_d^{(i)} - c^{(i)}$ *is non-Gaussian.*

2. $s_0^{(i)}$ *can have Gaussian components. Furthermore, if the number of Gaussian components exceeds 2, for all* $k, l \in \{1, \dots, c\}$ *with* $k \neq l$ *it holds that* $\gamma_k^{(i)} \neq \gamma_l^{(i)}$*, where* $\gamma_k^{(i)}$ *and* $\gamma_l^{(i)}$ *are the variances of the components* $s_{0k}^{(i)}$ *and* $s_{0l}^{(i)}$

Then, for fixed number of shared sources c *and for all* $d = 1, \dots, D$ $k_d^{(1)} = k_d^{(2)} = k_d$*, and there exist a permutation matrix* $P_d \in \mathbb{R}^{k_d \times k_d}$ *and an ivertible diagonal matrix* $\Lambda_d \in \mathbb{R}^{k_d \times k_d}$ *such that*

$$A_d^{(2)} = A_d^{(1)} P_d \Lambda_d$$

Proof. Theorem A.1 yields that if the individual components are not normal, then for each column of $a_j^{(1)}$ of $A_{d1}^{(1)}$ there is a column $a_i^{(2)}$ of $A_{d1}^{(1)}$ such that there exist $\lambda \neq 0$ with $a_j^{(2)} = \lambda a_j^{(1)}$. Since all mixing matrices have full column rank it follows that there is one-to-one correspondence between the columns of $A_{d1}^{(1)}$ and the columns of $A_{d1}^{(2)}$, and thus $k_d^{(1)} = k_d^{(2)}$

If at most one of the shared components is normal please refer to Comon (1994). Now consider the case when at least two components are normal. First the number of normal components in both representation is the same since c is fixed and the number of non-normal components is identifiable with the same arguments as above.

Computing the covariance between two different views $d, l \in \{1, \dots, D\}$ yields

$$\mathrm{Cov}(x_d, x_l) = A_{d0}^{(1)} \Gamma^{(1)} A_{l0}^{(1),\top} = A_{d0}^{(2)} \Gamma^{(2)} A_{l0}^{(2),\top}$$

where $\Gamma^{(i)}$ is the covariance matrix of $s_0^{(i)}$ for $i = 1, 2$. We define $A_{d0}^{\gamma,(i)} = A_{d0}^{(i)} \Gamma^{(i)\frac{1}{2}}$ for any $d \in \{1, \dots, D\}$. Let $P_d = (A_{d0}^{\gamma,(1),\top} A_{d0}^{\gamma,(1)})^{-1} A_{d0}^{\gamma,(1),\top} A_{d0}^{\gamma,(2)}$. Following the proof of Theorem 1 (Richard et al., 2021) we get that $P_d P_l^\top = \mathbb{I}_c = P_d P_k^\top = P_k P_l^\top$ for any $d, k, l \in \{1, \dots, D\}$. Thus, $P_l = P_d = P_k = P$ and they are orthogonal. Moreover, for all $d = 1, \dots, D$ it holds $\tilde{s}_0^{(1)} + \tilde{\epsilon}_d^{(1)} = P(\tilde{s}_0^{(2)} + \tilde{\epsilon}_d^{(2)})$ where $\tilde{\epsilon}_d^{(i)} \sim \mathcal{N}(0, \sigma_d^{(i)2} \Gamma^{(i)-1})$ and $\tilde{s}_0^{(i)} = \Gamma^{(i)-\frac{1}{2}} s_0^{(i)}$. From the last equation it follows that $\sigma_d^{(1)2} \Gamma^{(1)-1} = P(\sigma_d^{(2)2} \Gamma^{(2)-1}) P^\top$. Lemma 2 (Richard et al., 2021) implies that P is a sign and permutation matrix. □

B OPTIMIZATION

Lemma B.1. *Let $W \in \mathbb{R}^{c\times k}$ such that $WW^\top = \mathbb{I}_c$ and $x^1, \dots, x^N \in \mathbb{R}^k$ such that for every $j = 1, \dots, k$, we have $\sum_{i=1}^N (x_j^i)^2 = 1$ and for every $j \neq k$, we have $\sum_{i=1}^N x_j^i x_k^i = 0$. Then for every $j = 1, \dots, c$, it also holds that $\sum_{i=1}^N ((Wx^i)_j)^2 = 1$.*

Proof. Let W_j be the $j-$th row of W. Then

$$\sum_{i=1}^N ((Wx^i)_j)^2 = \sum_{i=1}^N (\sum_{l=1}^k W_{jl} x_l^i)^2 = \sum_{i=1}^N \sum_{l=1}^k \sum_{r=1}^k W_{jl} x_l^i W_{jr} x_r^i$$
$$= \sum_{l=1}^k \sum_{r=1}^k W_{jl} W_{jr} \sum_{i=1}^N x_l^i x_r^i = \sum_{l=1}^k \sum_{r=1}^k W_{jl} W_{jr} \delta_{lr} = \sum_{r=1}^k W_{jr}^2 = 1$$

where $\delta_{lr} = 1$ if $l = r$ and 0 otherwise. For the fourth equation we used that $\sum_{i=1}^N (x_j^i)^2 = 1$ and $\sum_{i=1}^N x_j^i x_k^i = 0$ for all $j \neq k$; and for the last one we used $WW^\top = \mathbb{I}_c$. □

Under the generative model assumptions and optimization constraints stated in 2 it holds

$$\mathcal{L}(W_1, \dots, W_D) = \sum_{i=1}^N \log f(\bar{s}_0^i) + \sum_{i=1}^N \sum_{d=1}^D \log p_{Z_d^{(2)}}(z_d^{(2)i}) + N \sum_{d=1}^D \log |W_d| \tag{4}$$
$$- \frac{1}{2\sigma^2} \Big(\sum_{d=1}^D \mathrm{trace}(Z_d^{(1)} Z_d^{(1)\top}) - \frac{1}{D} \sum_{d=1}^D \sum_{l=1}^D \mathrm{trace}(Z_d^{(1)} Z_l^{(1)\top}) \Big) \tag{5}$$

Proof. Let $\mathbf{x} = (x_1^\top, x_2^\top, \dots, x_D^\top)^\top \in \mathbb{R}^{K_D}$, $\tilde{\mathbf{s}} = (\tilde{s}_1^\top, \tilde{s}_2^\top, \dots, \tilde{s}_D^\top)^\top \in \mathbb{R}^{K_D}$, $\epsilon = (\epsilon_1^\top, \epsilon_2^\top, \dots, \epsilon_D^\top)^\top \in \mathbb{R}^{K_D}$, where $K_D = \sum_{d=1}^D k_d$ and for $W_d = A_d^{-1}$ define

$$\mathbf{W} = \begin{pmatrix} W_1 & 0 & \dots & 0 & 0 \\ 0 & W_2 & \dots & 0 & 0 \\ & & \ddots & & \\ 0 & 0 & \dots & W_{D-1} & 0 \\ 0 & 0 & \dots & 0 & W_D \end{pmatrix}, \ \mathbf{A} = \begin{pmatrix} A_1 & 0 & \dots & 0 & 0 \\ 0 & A_2 & \dots & 0 & 0 \\ & & \ddots & & \\ 0 & 0 & \dots & A_{D-1} & 0 \\ 0 & 0 & \dots & 0 & A_D \end{pmatrix}.$$

Furthermore, let $z_d := W_d x_d = \tilde{s}_d + \epsilon_d$, and $z_d^{(1)} := s_0 + \epsilon_{d0} \in \mathbb{R}^c$ and $z_d^{(2)} := s_d + \epsilon_{d1} \in \mathbb{R}^{k_d - c}$, i.e. $z_d = (z_d^{(1)}, z_d^{(2)})^\top$. Let $p_{\mathbf{X}}$ be the joint distribution of $x_1, \dots, x_D$, $p_{\mathbf{Z}}$ the joint distribution of $z_1, \dots, z_D$, $p_{\mathbf{Z}^{(1)}}$ the joint distribution of $z_1^{(1)}, \dots, z_D^{(1)}$, $p_{\mathbf{Z}^{(2)}}$ the joint distribution of $z_1^{(2)}, \dots, z_D^{(2)}$ and $p_{Z_d^{(2)}}$ the probability distribution of $z_d^{(2)}$.

Note that the model in 1 is equivalent to $\mathbf{x} = \mathbf{A}\mathbf{z}$. By multiplying with the inverse of $\mathbf{A}$ (i.e. $\mathbf{W}$) from the left we get $\mathbf{W}\mathbf{x} = \mathbf{z}$. Then for the joint likelihood of $x_1, \dots, x_D$ we get

$$\begin{aligned} p_{\mathbf{X}}(\mathbf{x}) &= p_{\mathbf{Z}}(\mathbf{z}) |\mathbf{W}| \\ &= p_{\mathbf{z}}(\mathbf{z}) \prod_{d=1}^D |W_d| \\ &= p_{\mathbf{Z}^{(1)}}(z_1^{(1)}, \dots, z_D^{(1)}) p_{\mathbf{Z}^{(2)}}(z_1^{(2)}, \dots, z_D^{(2)}) \prod_{d=1}^D |W_d| \\ &= p_{\mathbf{Z}^{(1)}}(z_1^{(1)}, \dots, z_D^{(1)}) \prod_{d=1}^D p_{Z_d^{(2)}}(z_d^{(2)}) \prod_{d=1}^D |W_d|. \end{aligned}$$

1. Second equation: $\mathbf{W}$ is a block diagonal matrix and for all $d = 1, \dots, D$, and $W_d \in \mathbb{R}^{k_d \times k_d}$.

2. Third equation: $z_1^{(1)}, \dots, z_D^{(1)} \perp\!\!\!\perp z_1^{(2)}, \dots, z_D^{(2)}$.

3. Fourth equation follows from the fact that $z_1^{(2)}, \dots, z_D^{(2)}$ are mutually independent since $\{s_{1i}\}_{i=1}^{k_1-c}, \dots \{s_{Di}\}_{i=1}^{k_D-c}, \{\epsilon_{1i}\}_{i=1}^{k_1}, \dots, \{\epsilon_{Di}\}_{i=1}^{k_D}$ are mutually independent.

It follows that

$$
\begin{aligned}
p_{\mathbf{Z}^{(1)}}(z_1^{(1)}, \dots, z_D^{(1)}) &= \int p_{\mathbf{Z}^{(1)}|S_0}(z_1^{(1)}, \dots, z_D^{(1)}|s_0) p_{S_0}(s_0) ds_0 \\
&= \int \Big(\prod_{d=1}^{D} \mathcal{N}(z_d^{(1)}; s_0, \sigma^2 \mathbb{I}_c) \Big) p_{S_0}(s_0) ds_0 \\
&\propto \int \exp\Big(- \sum_{d=1}^{D} \frac{\|z_d^{(1)} - s_0\|^2}{2\sigma^2} \Big) p_{S_0}(s_0) ds_0 \\
&= \int \exp\Big(- \frac{D\|s_0 - \bar{s}_0\|^2 + \sum_{d=1}^{D} \|z_d^{(1)} - \bar{s}_0\|^2}{2\sigma^2} \Big) p_{S_0}(s_0) ds_0 \\
&= \exp\Big(- \frac{\sum_{d=1}^{D} \|z_d^{(1)} - \bar{s}_0\|^2}{2\sigma^2} \Big) \int \exp\Big(- \frac{D\|s_0 - \bar{s}_0\|^2}{2\sigma^2} \Big) p_{S_0}(s_0) ds_0
\end{aligned}
$$

where $\bar{s}_0 = \frac{1}{D} \sum_{d=1}^{D} z_d^{(1)}$.

- For the second and third equation recall that $z_d^{(1)} = s_0 + \epsilon_{d0} \in \mathbb{R}^c$, where $\epsilon_{d0} \sim \mathcal{N}(0, \sigma^2 \mathbb{I}_c)$ and $s_0 \perp\!\!\!\perp \epsilon_{d0}$. This means that $z_d^{(1)}|s_0 \sim \mathcal{N}(s_0, \sigma^2 \mathbb{I}_c)$. From the following equations follow

$$
\begin{aligned}
p_{\mathbf{Z}^{(1)}|S_0}(z_1^{(1)}, \dots, z_D^{(1)}|s_0) &= \prod_{d=1}^{D} p_{Z_d^{(1)}|s_0}(z_d^{(1)}|S_0) \\
&= \prod_{d=1}^{D} \mathcal{N}(z_d^{(1)}; s_0, \sigma^2 \mathbb{I}_c)
\end{aligned}
$$

- The fourth equation results from

$$
\begin{aligned}
\sum_{d=1}^{D} \|z_d^{(1)} - s_0\|^2 = \sum_{d=1}^{D} \|z_d^{(1)} - \bar{s}_0 + \bar{s}_0 - s_0\|^2 &= \sum_{d=1}^{D} \Big(\|z_d^{(1)} - \bar{s}_0\|^2 + 2\langle z_d^{(1)} - \bar{s}_0, \bar{s}_0 - s_0 \rangle + \|\bar{s}_0 - s_0\|^2 \Big) \\
&= \sum_{d=1}^{D} \|z_d^{(1)} - \bar{s}_0\|^2 + 2 \sum_{d=1}^{D} \langle z_d^{(1)} - \bar{s}_0, \bar{s}_0 - s_0 \rangle + D\|\bar{s}_0 - s_0\|^2 \\
&= \sum_{d=1}^{D} \|z_d^{(1)} - \bar{s}_0\|^2 + 2 \Big\langle \sum_{d=1}^{D} z_d^{(1)} - D \cdot \frac{1}{D} \sum_{d=1}^{D} z_d^{(1)}, \bar{s}_0 - s_0 \Big\rangle + D\|\bar{s}_0 - s_0\|^2 \\
&= \sum_{d=1}^{D} \|z_d^{(1)} - \bar{s}_0\|^2 + D\|\bar{s}_0 - s_0\|^2.
\end{aligned}
$$

We define $f(\bar{s}_0) = \int \exp\Big(- \frac{D\|s_0 - \bar{s}_0\|^2}{2\sigma^2} \Big) p_{S_0}(s_0) ds_0$.

Note that

$$
\|z_d^{(1)} - \bar{s}_0\|^2 = \|z_d^{(1)}\|^2 - \frac{2}{D} \sum_{l=1}^{D} \langle z_d^{(1)}, z_l^{(1)} \rangle + \frac{1}{D^2} \sum_{l=1}^{D} \sum_{r=1}^{D} \langle z_r^{(1)}, z_l^{(1)} \rangle.
$$

Thus, it follows that

$$\begin{aligned}\sum_{d=1}^{D}\|z_d^{(1)}-\bar{s}_0\|^2 &= \sum_{d=1}^{D}\Big(\|z_d^{(1)}\|^2-\frac{2}{D}\sum_{l=1}^{D}\langle z_d^{(1)},z_l^{(1)}\rangle+\frac{1}{D^2}\sum_{l=1}^{D}\sum_{r=1}^{D}\langle z_r^{(1)},z_l^{(1)}\rangle\Big)\\ &= \sum_{d=1}^{D}\|z_d^{(1)}\|^2-\frac{2}{D}\sum_{d=1}^{D}\sum_{l=1}^{D}\langle z_d^{(1)},z_l^{(1)}\rangle+D\frac{1}{D^2}\sum_{l=1}^{D}\sum_{r=1}^{D}\langle z_r^{(1)},z_l^{(1)}\rangle\\ &= \sum_{d=1}^{D}\|z_d^{(1)}\|^2-\frac{1}{D}\sum_{d=1}^{D}\sum_{l=1}^{D}\langle z_d^{(1)},z_l^{(1)}\rangle\end{aligned}$$

Collecting all terms together we get

$$p_{\mathbf{X}}(\mathbf{x})=\exp\Big(-\frac{\sum_{d=1}^{D}\|z_d^{(1)}\|^2-\frac{1}{D}\sum_{d=1}^{D}\sum_{l=1}^{D}\langle z_d^{(1)},z_l^{(1)}\rangle}{2\sigma^2}\Big)f(\bar{s}_0)\prod_{d=1}^{D}p_{Z_d^{(2)}}(z_d^{(2)})\prod_{d=1}^{D}|W_d|$$

The data log-likelihood can be expressed as

$$\begin{aligned}\sum_{i=1}^{N}\log p_{\mathbf{X}}(x_1^i,\dots,x_D^i) &= \sum_{i=1}^{N}\Big(-\frac{\sum_{d=1}^{D}\|z_d^{(1)i}\|^2-\frac{1}{D}\sum_{d=1}^{D}\sum_{l=1}^{D}\langle z_d^{(1)i},z_l^{(1)i}\rangle}{2\sigma^2}\\ &\quad+\log f(\bar{s}_0^i)+\sum_{d=1}^{D}\log p_{Z_d^{(2)}}(z_d^{(2)i})+\sum_{d=1}^{D}\log|W_d|\Big)\\ &= \sum_{i=1}^{N}\log f(\bar{s}_0^i)+\sum_{i=1}^{N}\sum_{d=1}^{D}\log p_{Z_d^{(2)}}(z_d^{(2)i})+N\sum_{d=1}^{D}\log|W_d|\\ &\quad-\frac{1}{2\sigma^2}\Big(\sum_{i=1}^{N}\sum_{d=1}^{D}\|z_d^{(1)i}\|^2-\frac{1}{D}\sum_{i=1}^{N}\sum_{d=1}^{D}\sum_{l=1}^{D}\langle z_d^{(1)i},z_l^{(1)i}\rangle\Big)\\ &= \sum_{i=1}^{N}\log f(\bar{s}_0^i)+\sum_{i=1}^{N}\sum_{d=1}^{D}\log p_{Z_d^{(2)}}(z_d^{(2)i})+N\sum_{d=1}^{D}\log|W_d|\\ &\quad-\frac{1}{2\sigma^2}\Big(\sum_{d=1}^{D}\mathrm{trace}(Z_d^{(1)}Z_d^{(1)\top})-\frac{1}{D}\sum_{d=1}^{D}\sum_{l=1}^{D}\mathrm{trace}(Z_d^{(1)}Z_l^{(1)\top})\Big)\end{aligned}$$

In the case when the data is pre-whitened, it holds that the unknown unmixing matrices are orthogonal, i.e. $W_dW_d^\top=W_d^\top W_d=\mathbb{I}_{k_d}$ and $|\det W_d|=1$, and x_d and z_d are uncorrelated. Making similar observations as before we get for the joint probability of the multiple views:

$$p_{\mathbf{X}}(\mathbf{x})=p_{\mathbf{Z}^{(1)}}(z_1^{(1)},\dots,z_D^{(1)})\prod_{d=1}^{D}p_{Z_d^{(2)}}(z_d^{(2)})$$

Note that after whitening $z_d^{(1)}=\alpha(\sigma)(s_0+\epsilon_{d0})$ with $\alpha(\sigma)=(1+\sigma^2)^{-\frac{1}{2}}$. With similar observations as above we get

$$\begin{aligned}p_{\mathbf{Z}^{(1)}|s_0}(z_1^{(1)},\dots,z_D^{(1)}|s_0) &= p_{\mathbf{Z}^{(1)}|s_0}(\alpha(\sigma)(s_0+\epsilon_{10}),\dots,\alpha(\sigma)(s_0+\epsilon_{D0})|s_0)=\prod_{d=1}^{D}p_{Z_d^{(1)}|S_0}(\alpha(\sigma)(s_0+\epsilon_{d0})|s_0)\\ &= \prod_{d=1}^{D}\mathcal{N}(\alpha(\sigma)(s_0+\epsilon_{d0});s_0,\sigma^2\mathbb{I}_c)=\prod_{d=1}^{D}\mathcal{N}(z_d^{(1)};\alpha(\sigma)s_0,\alpha(\sigma)^2\sigma^2\mathbb{I}_c)\end{aligned}$$

It follows that

$$
\begin{aligned}
p_{\mathbf{Z}^{(1)}}(z_1^{(1)},\dots,z_D^{(1)}) &= \int p_{\mathbf{Z}^{(1)}|s_0}(z_1^{(1)},\dots,z_D^{(1)}|s_0)p_{S_0}(s_0)ds_0 \\
&= \int \Big(\prod_{d=1}^{D}\mathcal{N}(z_d^{(1)};\alpha(\sigma)s_0,\alpha(\sigma)^2\sigma^2\mathbb{I}_c)\Big)p_{S_0}(s_0)ds_0 \\
&\propto \int \exp\Big(-\sum_{d=1}^{D}\frac{\|z_d^{(1)}-\alpha(\sigma)s_0\|^2}{2\alpha(\sigma)^2\sigma^2}\Big)p_{S_0}(s_0)ds_0 \\
&= \int \exp\Big(-\frac{D\|\alpha(\sigma)s_0-\bar{s}_0\|^2+\sum_{d=1}^{D}\|z_d^{(1)}-\bar{s}_0\|^2}{2\alpha(\sigma)^2\sigma^2}\Big)p_{S_0}(s_0)ds_0 \\
&= \exp\Big(-\frac{\sum_{d=1}^{D}\|z_d^{(1)}-\bar{s}_0\|^2}{2\alpha(\sigma)^2\sigma^2}\Big)\int \exp\Big(-\frac{D\|\alpha(\sigma)s_0-\bar{s}_0\|^2}{2\alpha(\sigma)^2\sigma^2}\Big)p_{S_0}(s_0)ds_0
\end{aligned}
$$

where $\bar{s}_0 = \frac{1}{D}\sum_{d=1}^{D} z_d^{(1)}$. We define $f_\sigma(\bar{s}_0) = \int \exp\Big(-\frac{D\|\alpha(\sigma)s_0-\bar{s}_0\|^2}{2\alpha(\sigma)^2\sigma^2}\Big)p_{S_0}(s_0)ds_0 = \int \exp\Big(-\frac{D\|s_0-(1+\sigma^2)^{\frac{1}{2}}\bar{s}_0\|^2}{2\sigma^2}\Big)p_{S_0}(s_0)ds_0$. For the data log-likelihood we get

$$
\begin{aligned}
\sum_{i=1}^{N}\log p_{\mathbf{x}}(x_1^i,\dots,x_D^i) &= \sum_{i=1}^{N}\log f_\sigma(\bar{s}_0^i)+\sum_{i=1}^{N}\sum_{d=1}^{D}\log p_{Z_d^{(2)}}(z_d^{(2)i})-N\cdot D\cdot 1 \\
&\quad -\frac{D\cdot c}{2\alpha(\sigma)\sigma^2}+\frac{1}{2D\alpha(\sigma)^2\sigma^2}\sum_{d=1}^{D}\sum_{l=1}^{D}\mathrm{trace}(Z_d^{(1)}Z_l^{(1)\top})
\end{aligned}
$$

It be easily derived from 4 by making the following observations resulting from whitening

- $N\sum_{d=1}^{D}\log|W_d| = ND$ since $\forall d\ W_d$ is orthogonal
- $\mathrm{trace}(Z_d^{(1)}Z_d^{(1)\top}) = c$ due to Lemma B.1

□

Remark. Let $D = 1$ and we have the following simple BSS model for $k < p$:

$$x = As \text{ with } x \in \mathbb{R}^p, s \in \mathbb{R}^k, A \in \mathbb{R}^{p\times k}$$

It follows that $s = (A^\top A)^{-1}A^\top x$. Define $W = (A^\top A)^{-1}A^\top$. The density function of $p_s(s)$ is given by (, https://math.stackexchange.com/users/491644/maxim)

$$p_s(s) = \det(WW^\top)^{-\frac{1}{2}}\int_{Wx=s} p_x(x)dS(x),$$

where we integrate over a $p-k$ dimensional surface. Thus, in that case we use this representation for our optimization.

C REAL DATA EXPERIMENT

C.1 DATA ACQUISITION AND PREPROCESSING

Our analysis is primarily based on two large gene expression data sets, denoted by (in our code) Dataset1[1] (Arrieta-Ortiz et al., 2015) with 265 transcriptome datasets obtained from 38 unique experimental designs and Dataset2 (Nicolas et al., 2012)[2] containing 262 samples from 104 different experimental conditions.

We removed genes with missing values from Dataset 1 and we selected 3994 genes that are present in both datasets. To evaluate our results, we collect a ground truth network from the online database *Subti*Wiki [3] which consists of 5,952 pairs of regulator and regulated gene. Since our method predicts pairs of co-regulated genes, we transform the ground truth network into an undirected graph that links genes with a common regulator. Thus, the ground truth network is stored in the form of an adjacency matrix with entries 1 if the genes are co-regulated and 0 otherwise.

C.2 GENE-GENE INTERACTION PIPELINE

The main steps of our method are presented in Algorithm 1. We infer latent components from the data as described in Appendix C.2.1. Afterward, we learn a sparse undirected graph from the estimated independent components (see Appendix C.2.2).

C.2.1 DATA INTEGRATION

Let $X \in \mathbb{R}^{n \times p}$ be a transcriptome data matrix with n samples (or experimental outcomes) and p genes. We assume that the transcriptome matrix follows a linear latent model, i.e. there exist a matrix $A \in \mathbb{R}^{n \times k}$ and a matrix $S \in \mathbb{R}^{k \times p}$ such that $X = AS$. The k components can be represent gene expression. If a group of genes is either over or under-expressed in a specific component they are usually assumed to share a functional property in the genome. Additionally, if the components are independent (i.e. a BSS model) we assume that the components represent independent gene pathways, i.e. the components' groups of over/under-expressed genes act independently from each other given the experimental conditions.

OmicsPLS This baseline is not a BSS model, i.e. the estimated components are not necessarily independent. We make an additional assumption that the view-specific sources are orthogonal to the other views. The model is defined by

$$\begin{aligned} X_1 &= A_1 Y_1 + B_1 Z_1 + E_1 \\ X_2 &= A_2 Y_2 + B_2 Z_2 + E_2, \end{aligned}$$

where $Y_1 \in \mathbb{R}^{c \times n}$ $Y_2 \in \mathbb{R}^{c \times n}$ are the latent variables that are responsible for the joint variation between X_1 and X_2, i.e. Y_1 and Y_2 are obtained by solving a CCA problem, and $Z_i \in \mathbb{R}^{k_i - c \times n}$ represent the components that are orthogonal to X_j with $j \neq i$, and E_i is the noise (or residuals). In our application we define $S_i = (Y_i, Z_i)$ for the downstream task of interest.

C.2.2 GRAPHICAL LASSO

Graphical lasso (glasso) is a maximum likelihood estimator for inferring graph structure in a high-dimensional setting (Friedman et al., 2007). This method uses l_1 regularization to estimate the precision matrix (or inverse covariance) of a set of random variables from which a graph structure can be determined. The optimization problem which glasso solves can be formalized as follows

$$\min_{\Theta \succ 0} - \log\det(\Theta) + \mathrm{tr}(\hat{\Sigma}\Theta) + \lambda \|\Theta\|_1, \tag{6}$$

[1]The dataset is available at `https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE67023`

[2]The dataset can be found at `https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE27219`

[3]See `http://www.subtiwiki.uni-goettingen.de/v4/exports`

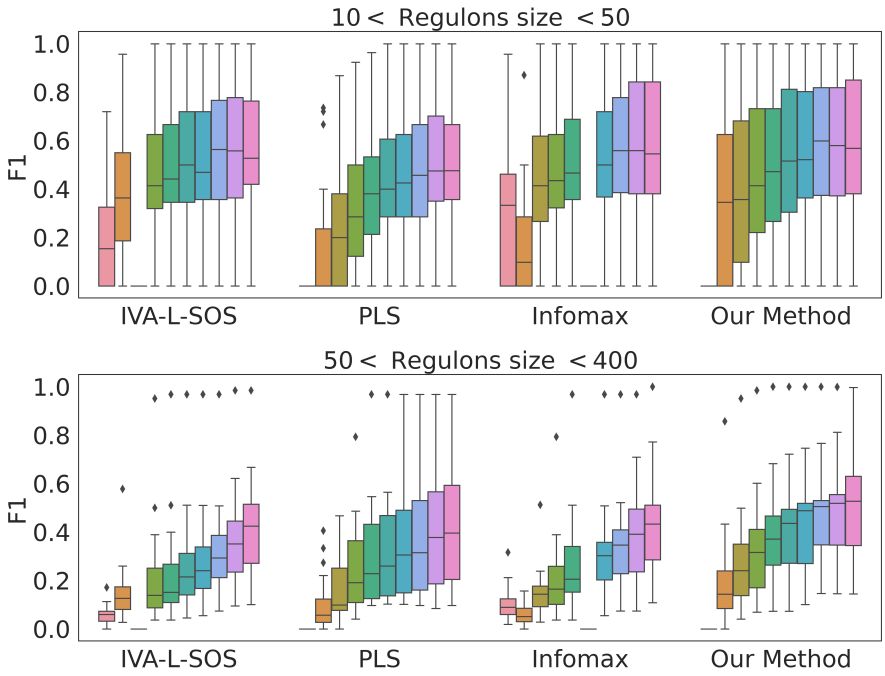


Figure 7: Boxplots of the regulons F1 score for two groups of regulons depending on their size (between 10 and 50, and between 50 and 400).

where $\hat{\Sigma}$ is the empirical covariance or correlation matrix and $\Theta := \Sigma^{-1}$ denotes the precision matrix. In our setting, the input for the glasso is the Pearson's correlation matrix of the gene representations retrieved with ICA at the preceding step. We can read graph structure from the estimated matrix $\hat{\Theta}$ as follows: if the ij entry of $\hat{\Theta}$ is not 0 (i.e. $\hat{\Theta}_{ij} \neq 0$) there is an edge between the genes i and j, i.e. the genes might be co-regulated. We used the `huge`[4] R package for the implementation of graphical lasso.

C.2.3 Extended EBIC

There are various criteria for model selection and hyperparameter tuning of glasso models. Chen & Chen (2008) propose an information criterion for Gaussian graphical models called extended BIC (EBIC) that takes the form

$$-\log\det(\Theta(E)) + \mathrm{tr}(\hat{\Sigma}\Theta(E)) + |E|\log n + 4|E|\gamma\log p, \tag{7}$$

where E is the edge set of a candidate graph and $\gamma \in [0, 1]$. Models that yield low EBIC scores are preferred. Note that positive values for γ lead to sparser graphs. Foygel & Drton (2010) suggest that $\gamma = 0.5$ is a good choice when no prior knowledge is available. In our experiments, we select the λ that minimizes the EBIC score with $\gamma = 0.5$.

C.2.4 Precision and Recall

To evaluate the proposed method, we use two different evaluation strategies. First, we count the true positive and false positive (or unknown) edges from the output undirected graph. Edges are annotated as true positive if they connect pairs of co-regulated genes. In the second part of our evaluation, we are interested in the regulon prediction power of our method. For each known regulon, we compute precision and recall score in the following way:

[4]See `https://CRAN.R-project.org/package=huge`.

$$\text{Precision}(R) = \frac{\sum_{g \in R} |N(g) \cap N^{gt}(g)|}{\sum_{g \in R} |N(g)|},$$
$$\text{Recall}(R) = \frac{\max_{C \in \mathcal{C}(R)} |C|}{|R|}$$

where R denotes the set of regulon genes, $N(g)$ and $N^{gt}(g)$ are the sets of all neighbours of gene g in the output network and ground truth network, respectively, and $\mathcal{C}(R)$ the set of all connected components in the induced graph with vertices in R.

C.2.5 METHOD

All steps described above are summarized in the following pseudo code.

Algorithm 1 Algorithmic description of the data integration task.

1: **Input:**
   $X_1, \in \mathbb{R}^{n_1 \times p}, X_2 \in \mathbb{R}^{n_2 \times p}$ is a data matrix with n_1 and n_2 samples and p genes
   Λ is a set of regularization parameters
   γ EBIC selection parameter (7)
2: Perform a data integration method to obtain $S_1, \in \mathbb{R}^{k_1 \times p}, S_2 \in \mathbb{R}^{k_2 \times p}$
3: Concatenate $S = (S_1, S_2)^\top \in \mathbb{R}^{k_1 + k_2 \times p}$
4: Compute the Pearson correlation matrix $\hat{\Sigma} \in \mathbb{R}^{p \times p}$ of S.
5: Estimate the precision matrices $\{\hat{\Theta}^\lambda\}_{\lambda \in \Lambda}$ which solves 6 for each λ from the set Λ
6: Select the final $\hat{\Theta}^{out} \in \{\hat{\Theta}^\lambda\}_{\lambda \in \Lambda}$ according to EBIC(γ) (see 7)
7: **Output:**
   the selected $\hat{\Theta}^{out}$

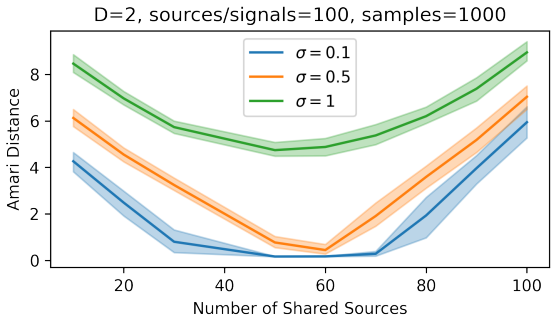


Figure 8: We have the two view case again with number of total sources and observed signals 100 and number of samples 1000. We consider three cases of noise standard deviation: $\sigma = 0.1, 0.5, 1$. As soon as enough shared sources are present (around 60) our method lower value of Amari distance (the lower the better) in all cases. In the the first two cases ($\sigma = 0.1$ or 0.5) the Amari distance gets closer to 0 when the shared sources are 60. The error bars correspond to 95% confidence intervals based on 50 independent runs of the experiment.

(a) (b)

Figure 9: Comparison of MultiViewICA and our method on a two-view shared response model setting. In Figure 9a we fix the sample size and measure the Amari distance for sources $60, 70, \ldots 110$. In Figure 9b the number of sources is set to 100 and we conduct the experiments for different sample sizes (x-axis). It seems that our method outperforms MultiViewICA in both scenarios.

D Synthetic Experiments

D.1 Amari distance

The Amari distance (Amari et al., 1995) between two invertible matrices $A, B \in \mathbb{R}^{n \times n}$ is defined by

$$\text{amari}(A, B) := \sum_{i=1}^{n} \Big(\sum_{j=1}^{n} \frac{|c_{ij}|}{\max_k |c_{ik}|} - 1 \Big) + \sum_{j=1}^{n} \Big(\sum_{i=1}^{n} \frac{|c_{ij}|}{\max_k |c_{kj}|} - 1 \Big), \qquad C := A^{-1}B.$$

D.2 Additional Experiments on Synthetic Data

Noisy high-dimensional views. First, we investigate the effect of noise on the Amari distance in the two-view experiment. In Section 5.1, we empirically showed in the noiseless case that if we have enough shared sources present in the two views, we can recover the individual ones up to permutation and scaling with high accuracy. We repeat this experiment in a noisy setting. We consider three cases when the noise's standard variation is $\sigma = 0.1, 0.5, 1$. The results are depicted in Figure 8. In the first two cases the results are close to the one discussed in Section 5.1. As expected, by adding noise with high variance ($\sigma = 1$) our method does not converge and affects the quality of the estimated mixing matrices measured with the Amari distance. The whole procedure is repeated 50 times, and the error bars are the 95% confidence intervals based on the independent runs.

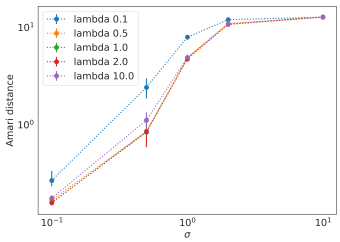


Figure 10: Choice of Hyperparameter λ. The data comes from a two-view model with 50 shared and 50 individual sources per view. The x-axis is represents the noise standard deviation and the y-axis the Amari distance.

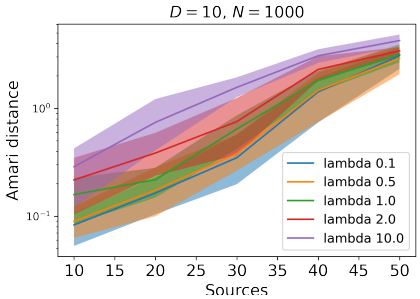


Figure 11: The data is generated according to a model, where no individual sources are present and the noise per view is uniformly samples from the interval [1, 2]. The number of views is set to 10 and the sample size is 1000 for all experiments. We vary the number of sources from 10 to 50. The experiment is repeated 50 times. We fit the model with 5 different hyperparameters. Our method shows better results in the case when the hyperparameter is 0.1 and 0.5.

Objective function motivation. In the following experiment, we compare MultiViewICA and our method when the observed data is high-dimensional on a two-view shared response model applications, i.e. no individual sources. The experimental setup allows for comparing standard MLE (MultiViewICA) and MLE after whitening (Our Method). Figure 2 summarizes the results. Figure 9a compares the two methods for fixed sample size 1000. In Figure 9b we fixed the number of sources to be 100 and vary the sample size. For all experiments the noise standard deviation is 0.01. It seems that our method performs better in the case of insufficient data. This could be empirical evidence that the trace has stronger regularization properties than the MMSE term in the MultiViewICA objective.

Choice of λ For this experiment we used data generated from 2 views with 50 individual and 50 shared sources with varying noise standard deviation $\sigma \in \{0.1, 0.5, 1, 2, 10\}$ (x-axis). Each of the lines in Figure 10 correspond to a fixed hyperparameter $\lambda \in \{0.1, 0.5, 1, 2, 10\}$. It can be deduced that for this particular experiment for $\lambda \geq 0.5$ there is no significant difference in the model performance.

Additional experiments on the impact of the hyperparameter λ. In this subsection, we perform the same experiment as in Figure 2b for different values of the hyperparameter lambda $0.1, 0.5, 1, 2$ and 10. Figure 11 suggests that in that case choosing lower values for the parameter yields better results.

D.3 Baselines Implementation

The code for GroupICA, ShICA, MultViewICA is distributed with BSD 3-Clause License. The `OmicsPLS` R library has a GPL-3 license, the `scikit-learn` library is distributed with BSD 2-Clause License.