



Horseshoe-type priors for Independent Component Estimation

Bayesian Inference with Generative Maps

with Soham Ghosh and Nick Polson

Wisconsin-Madison and UChicago Booth School of Business

Key Idea

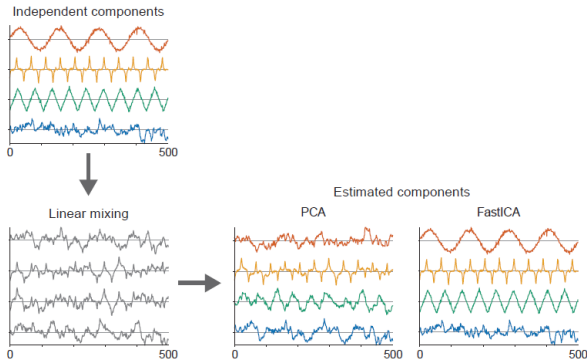
- We address the classic ICA problem from a full Bayesian perspective: jointly estimating latent sources and mixing matrix using generative latent variable models.
- Existing methods (e.g., MacKay, FastICA) rely on MAP or heuristics; full posterior inference remains underexplored.
- We use a latent variable hierarchy to derive:
 - Gibbs and EM algorithms
 - Envelope/proximal optimization updates
- Unifies multiple priors (e.g., Horseshoe, t , Lasso) via scale mixture representation.

Independent Component Analysis

- Independent component analysis (ICA) is a statistical and computational technique for revealing hidden factors that underlie sets of random variables, measurements, or signals.
- ICA defines a generative model for multivariate data, typically given as a large database of samples.
- Data are assumed to be linear or nonlinear mixtures of some unknown latent variables, and the mixing system is also unknown.
- The latent variables are assumed non-gaussian and mutually independent, and they are called the **independent components** of the observed data.
- These IC's, also called sources or factors, can be found by ICA.
- ICA can be seen as an extension to PCA and factor analysis.
- ICA is a much more powerful technique, however, capable of finding the underlying factors or sources when these classic methods fail completely.

PCA vs. ICA

Original signals with four visually distinctive shapes are linearly mixed, and both principal component analysis (PCA) and linear independent component analysis (ICA) are applied. While PCA fails to recover the original sources, ICA is shown to successfully separate them.



1.1.2 Benefit for Modeling – Identifiability

A remarkable property of tensors, first discovered by [Kruskal \(1977\)](#), is that nearly all low rank tensors can be uniquely written as a sum of rank-one tensors. This is to be contrasted with the fact that there are infinite many ways to write a low rank matrix as a sum of rank-one matrices. The implication of such identifiability can be demonstrated in the comparison between PCA and independent component analysis (ICA). The principal components (PC) can be identified as the singular vectors of the covariance matrix, and likewise, the independent components (IC) are associated with the “singular vectors” of the fourth order cumulant tensor. Suppose that we observe d independent random variables with mean zero, unit variance and nonzero excess kurtosis, up to an unknown rotation. We cannot reconstruct these variables via PCA because the variance is invariant to any rotation. We can however do so from its fourth order cumulant tensor which has rank d with each component corresponding to one of the independent variables.



Figure 2: ICA vs PCA: $\kappa_4(\mathbf{u}^\top \mathbf{X}) = \mathbb{E}(\mathbf{u}^\top \mathbf{X})^4 - 3$ and $\text{cov}(\mathbf{u}^\top \mathbf{X})$ as functions of \mathbf{u} over the unit circle.

As a more concrete example, consider the case of two independent random variables $\mathbf{X} = (X_1, X_2)^\top$ with $\text{var}(\mathbf{X}) = \mathbb{I}_d$. The left panel of Figure 2 shows the excess kurtosis as a function of $\mathbf{u} = (u_1, u_2)^\top$: $f(\mathbf{u}) := \kappa_4(u_1 X_1 + u_2 X_2)$ over the unit circle. There are only four maximizers corresponding to the independent components $\pm X_1$ and $\pm X_2$ respectively. In contrast, $\text{cov}(\mathbf{u}^\top \mathbf{X})$ remains constant over all \mathbf{u} on the unit circle, as shown in the right panel.

¹from [Auddy and Yuan \[2023\]](#), [Auddy et al. \[2024\]](#)

Blind source separation

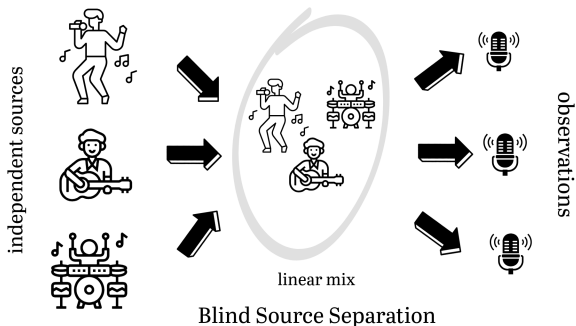
- [MacKay \[1996\]](#): Blind Source Separation (BSS) consists in recovering n source signals (\mathbf{s}) from the observations (\mathbf{x}) which are linear mixtures (with unknown coefficients \mathbf{A}) of the source signals.

$$\mathbf{x} = \mathbf{A}\mathbf{s}.$$

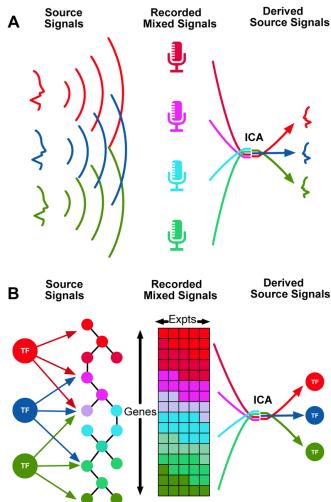
- The algorithms find inverse of \mathbf{A} (within a post-multiplicative factor) given only a set of examples $\{\mathbf{x}\}$.
- Diverse applications: signal processing, machine learning and medical imaging.
- Examples include speech separation (the ‘cocktail party problem’), processing of arrays of radar or sonar signals, and processing of multi-sensor biomedical recordings.
- BSS: ‘An array of d receivers picks up linear mixtures of n source signals.’

Cocktail Party Problem

Special case where $\# \text{ sensors} = \# \text{ sources}$.



Applications



(A): The 'cocktail party problem,' (B): Identifying source signals of transcriptional regulators from complex gene expression measurements. Figure from [Karczewski et al., 2014].

Formal statement

- Attempt to recover source signals \mathbf{s} from observations \mathbf{x} , where $\mathbf{x} = \mathbf{A}\mathbf{s} + \epsilon$, are linear mixtures with unknown weights \mathbf{A} .
- This is done by finding a square matrix \mathbf{W} which is the inverse of the mixing matrix \mathbf{A} , up to permutation and change of scale.
- Want to 'demix' observations \mathbf{x} into $\mathbf{y} = \mathbf{W}\mathbf{x}$, such that $\mathbf{y} \approx \mathbf{s}$, we need $\mathbf{W} \approx \mathbf{A}^{-1}$.
- To recover the independent sources, we need to estimate $\mathbf{W} \approx \mathbf{A}^{-1}$.
- The observations can be thought of as linear instantaneous mixture of time series, where, at each epoch, the observations are linear combinations of the sources at that epoch.
- The linear instantaneous model:

$$\mathbf{x}_t = \mathbf{A}\mathbf{s}_t + \epsilon_t, \quad t = 0, \dots, N - 1, \quad (1)$$

where, $\mathbf{x}_t = (x_{1t}, x_{2t}, \dots, x_{dt})^T$ are the observation vectors, $\mathbf{s}_t = (s_{1t}, s_{2t}, \dots, s_{nt})^T$ are the sources and $\epsilon_t = (\epsilon_{1t}, \epsilon_{2t}, \dots, \epsilon_{dt})$ are the additive noises.

- The goal is to estimate \mathbf{s} and \mathbf{A} .

Time versus Transfer Domain

- The linear instantaneous model in time domain: observations at time t are noisy combination of sources at t , and signals of length n :

$$\mathbf{x}_t = \mathbf{A}\mathbf{s}_t + \boldsymbol{\epsilon}_t, \quad t = 0, \dots, N - 1, \quad (2)$$

where, $\mathbf{x}_t = (x_{1t}, x_{2t}, \dots, x_{dt})^T$ are the observation vectors, $\mathbf{s}_t = (s_{1t}, s_{2t}, \dots, s_{nt})^T$ are the sources and $\boldsymbol{\epsilon}_t = (\epsilon_{1t}, \epsilon_{2t}, \dots, \epsilon_{dt})$ are the additive noises.

- The goal is to estimate \mathbf{s} and \mathbf{A} .
- We assume that there is a basis $n \times n$ matrix Φ such that sources have a sparse representation on it. The equivalent model on the transfer domain is:

$$\mathbf{x}^{(k)} = \mathbf{A}\mathbf{s}^{(k)} + \boldsymbol{\epsilon}^{(k)}, \quad k = 0, \dots, N - 1, \quad (3)$$

where, k is the coefficient of the basis in the decomposition.

- Separation using (2) and (3) are equivalent since Φ is a basis matrix.

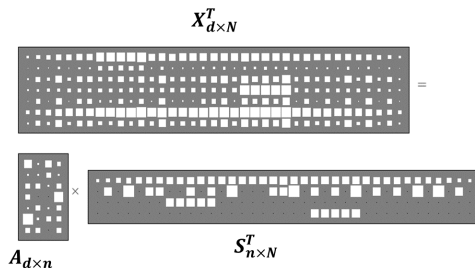
$$\mathbf{X}_{d \times N}^T = \mathbf{A}_{d \times n} \mathbf{S}_{n \times N}^T$$


Figure 1: $\mathbf{X}^T = \mathbf{A}\mathbf{S}^T(+\epsilon)$

- Assume that we have a matrix of gene expressions X where x_{tg} is the expression level of gene g in tissue (or subject) t .

Mackay's ICA genes ii

- How do we model such data using latent variable methods?
- We assume that there are latent variables $\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_d$ possibly affecting each gene expression, with their strength given by vectors \mathbf{s} . (MacKay calls them 'signatures')
- The gene expression values are then the signatures a_{th} weighted by the strengths s_{hg} , with some noise.

$$x_{tg} = \sum_h a_{th}s_{hg} + \epsilon_{tg}, \text{ or } \mathbf{X}^T = \mathbf{A}\mathbf{S}^T + \mathbb{E}.$$

- The prior on \mathbf{s} can be used to specify which latent factors are important, and, it can be largely sparse. We can also specify that each gene is influenced by only one latent factor – similar to 'clustering'.
- Use non-Gaussian, heavy-tailed priors.
- Without informative priors, linear ICA is unidentifiable – our Bayesian approach leverages global-local shrinkage priors to achieve separation.

- Bhadra et al. [2024]: Statistical view of high-dimensional deep learning.
- Output Y and input X connected via a statistical model:

$$P(\mathbf{Y} \mid \mathbf{W}, \mathbf{X}) = P(\mathbf{Y} \mid \mathbf{a}), \mathbf{a} = \mathbf{WX}.$$

- \mathbf{a} is linear factors to be extracted, connected via a hierarchical model with no error in second stage of hierarchy.
- How to solve inverse problem and find \mathbf{W} ? Unidentified if all Gaussian.
- Kruskal problem: use higher order moment matching.
- Other approach is to assume $\mathbf{x} \sim p(\mathbf{x})$ is stochastic and use Bayes.

Bayesian framework

Generative Models and Latent Variables [MacKay, 1996]

- Many statistical models are generative models that use latent variables to describe a probability distribution over observables (Everitt 1984).
- Examples of latent variable models include:
 - Mixture models: Model the observables as coming from a superposed mixture of simple probability distributions (Hanson et al. 1991). The latent variables are the unknown class labels of the examples.
 - Hidden Markov models (Rabiner and Juang 1986)
 - Factor analysis
 - Helmholtz machines (Hinton et al. 1995; Dayan et al. 1995)
 - Density networks (MacKay 1995; MacKay 1996)
- Latent variables often have a simple, separable distribution.
- Learning a latent variable model involves finding a description of the data in terms of independent components.
- Natural to expect that ICA should admit a generative latent variable representation.

MacKay [1996] formulation

- MacKay [1996] shows that the heuristic algorithms of Bell and Sejnowski [1995] can be viewed as a maximum likelihood algorithm.

$$\begin{aligned} p(\mathbf{x}^{(n)} \mid \mathbf{A}, \mathbf{H}) &= \int p(\mathbf{x}^{(n)} \mid A, \mathbf{s}^{(n)}) p(\mathbf{s}^{(n)}) d\mathbf{s}^{(n)} \\ &= \int \prod_j \delta(\mathbf{x}_j^{(n)} - A_{ji} \mathbf{s}_i^{(n)}) \prod_i p_i(\mathbf{s}_i^{(n)}) d\mathbf{s}^{(n)} \end{aligned}$$

$$p(\mathbf{x} \mid A) = \frac{1}{|A|} \prod_i p_i(A_{ij}^{-1} x_j) \Rightarrow \log p(\mathbf{x}^{(n)} \mid A, \mathbf{H}) = \log |W| + \log p_i(\sum A_{ij}^{-1} x_j).$$

- We need to assume \mathbf{x} is generated without noise to get the Bell and Sejnowski [1995] algorithm.
- Straightforward to replace $\delta(\mathbf{x}_j^{(n)} - A_{ji} \mathbf{s}_i^{(n)})$ with a probability distribution over $\mathbf{x}_j^{(n)}$ with mean $A_{ji} \mathbf{s}_i^{(n)}$.²

² $\int \delta(\mathbf{x} - A\mathbf{s}) f(\mathbf{s}) d\mathbf{s} = |A|^{-1} f(\mathbf{x}/A).$

Bayesian framework

- The joint likelihood on observables and hidden states is:

$$p(\mathbf{X}, \mathbf{S} \mid \mathbf{A}, \sigma^2) = \prod_{n=0}^{N-1} p(\mathbf{x}^{(n)} - \mathbf{A}\mathbf{s}^{(n)}) p(\mathbf{s}^{(n)}) = \prod_{n=0}^{N-1} \mathcal{N}(\mathbf{x}^{(n)} \mid \mathbf{A}\mathbf{s}^{(n)}, \sigma^2 I_d) \times p(\mathbf{s}^{(n)}).$$

- We can think of this as a Bayesian hierarchical model:

$$\mathbf{x}^{(n)} \sim \mathcal{N}(\mathbf{A}\mathbf{s}^{(n)}, \sigma^2 I_d) \text{ with } p(\mathbf{s}) = \prod_{i=1}^d p_i(s_i^{(n)}).$$

- The sources $\mathbf{s}^{(n)}$ have an independent components distribution and can be viewed as latent variables.
- We can introducing auxiliary variables λ where

$$p(\mathbf{s}^{(n)}) = \int p(\mathbf{s}^{(n)} \mid \lambda) p(\lambda) d\lambda,$$

allowing EM and MCMC algorithms using the joint posterior $p(\mathbf{s}^{(n)}, \lambda \mid \mathbf{x}^{(n)})$.

Related Work:

- **MacKay (1996)**: Interprets Bell-Sejnowski as MAP estimation with hyperbolic secant prior. We extend this to full Bayesian inference using Pólya-Gamma augmentation.
- **Fevotte et al. (2004), Fevotte and Godsill (2006)**: Bayesian ICA with Student's- t and Jeffreys priors, using Gibbs sampling.
- **Ablin et al. (2019)**: EM algorithm for ICA using Gaussian scale mixtures. Our work complements this by including full posterior sampling and EM with Horseshoe-type priors.
- **Horseshoe Priors**: Connect shrinkage-based learning with heavy-tailed ICA models via scale mixtures.

Our contributions:

- Unified Bayesian ICA model encompassing multiple priors.
- Derivation of Gibbs, EM, and envelope-based optimization strategies.
- Empirical comparisons of EM vs MacKay algorithms with realistic noise and scale settings.

General case

- In the general case, BSS consists of estimating n signals (the sources) from the sole observation of d mixtures of them (the observations).
- Over-determined ($d \geq n$) case easy: many efficient approaches, esp. within Independent Component Analysis.
- the general linear instantaneous case, with mixtures possibly noisy and under-determined ($d \leq n$) is challenging.
- Sparsity-Based BSS: Common approach for Blind Source Separation (BSS), especially for underdetermined mixtures.
- Exploits source sparsity assumptions: only a few expansion coefficients of sources are significantly different from zero.
- Existing approaches: Student's t or Jeffrey's prior on \mathbf{s} [Févotte et al., 2004, Févotte and Godsill, 2006].

MacKay's cosh prior

- We adopt a Bayesian shrinkage approach: $s \sim p(s)$ acts as a regularization prior and unifies several MAP estimators under a common latent variable model.
- MacKay [1996] shows that the Bell–Sejnowski algorithm corresponds to a source prior:

$$p(s) \propto \frac{1}{\cosh(s)} = \frac{1}{e^s + e^{-s}}.$$

- This distribution can be expressed as a normal scale mixture with a Pólya-Gamma mixing distribution:

$$s \mid \tau \sim \mathcal{N}\left(0, \frac{1}{4\tau}\right), \quad \tau \sim PG(1, 0),$$

using the identity:

$$\frac{1}{\cosh(s)} \propto \int_0^\infty \exp(-2\tau s^2) p(\tau) d\tau.$$

Generalization via gain parameter

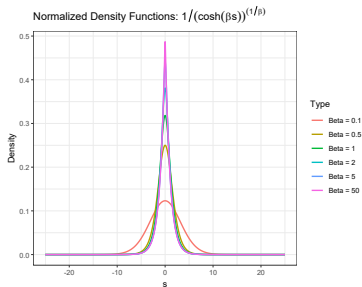
- Introduce a gain parameter β to obtain a family of heavy-tailed priors:

$$p(s \mid \beta) \propto \frac{1}{\cosh^{1/\beta}(\beta s)}$$

- Interpolates between:

$$p(s \mid \beta) \rightarrow \begin{cases} \exp(-|s|) & \beta \rightarrow \infty \text{ (Lasso)} \\ \mathcal{N}(0, 1/\beta) & \beta \rightarrow 0 \text{ (Ridge)} \end{cases}$$

- The hyperbolic secant family bridges sparse and dense priors.



Pólya-Gamma distribution

Definition

Suppose $b \geq 0$, and $z \geq 0$. The Pólya-Gamma distribution $PG(\beta)$ is defined as the density p_{PG} on \mathbb{R}^+ that has the following Laplace transform:

$$\cosh^{-\beta} \left(\sqrt{\frac{t}{2}} \right) = \int_0^{\infty} \exp(-tx) p_{PG}(x \mid \beta) dx.$$

- Reparametrize $t \mapsto 2\tau^2$, we get the normal scale mixture results.
- This is connected to the Jacobi theta distributions studied by [Biane et al. \[2001\]](#), [Devroye \[2009\]](#): J^* has a Jacobi distribution if,

$$J^* \stackrel{D}{=} \frac{2}{\pi^2} \sum_{k=1}^{\infty} \frac{e_k}{(k - \frac{1}{2})^2}, \quad e_k \sim \text{Exp}(1),$$

and the moment generating function is:

$$\mathbb{E}(e^{-tJ^*}) = \frac{1}{\cosh \sqrt{2t}}.$$

Connection with horseshoe priors

- The $\cosh(\cdot)$ priors have a connection with the popular horseshoe priors:

$$\beta_j \sim \mathcal{N}(0, \lambda_j^2 \tau^2 \sigma^2), \lambda_j \sim p(\lambda_j), \tau \sim p(\tau).$$

- The horseshoe prior places a standard half-Cauchy distribution over λ_j .
- This induces an induce an un-standardised unit hyperbolic secant distribution over $\xi_j = \log(\lambda_j)$

$$p_{HS}(\xi_j) = \frac{1}{\pi} \frac{1}{\cosh(\xi_j)}.$$

- Log-scale shrinkage priors can unify commonly used continuous shrinkage priors [Schmidt and Makalic, 2018].

Computation

- Three common approaches for ICA:
 1. Method of moments: Well established
 2. Iteratively nonlinear algorithms: Established. No general theory
 3. Bayesian framework: Little done.
- We develop (3) in more details.
- We also show that (2) can be viewed as EM/MM algorithms.

Févotte et al. [2004]: each s_i has a $t(\alpha_i, \lambda_i)$ prior, with d.f. α_i and scale λ_i , which can be written as a Normal scale mixture of Inverse Gamma distribution.

- Let $\Sigma_{s_j} = (\frac{1}{\sigma^2} \mathbf{A}^T \mathbf{A} + \text{diag}(v_j)^{-1})^{-1}$, $\mu_{s_j} = \frac{1}{\sigma^2} \Sigma_{s_j} \mathbf{A}^T \mathbf{x}_j$, then,

$$\mathbf{s} \mid \mathbf{A}, \sigma, \mathbf{v}, \alpha, \lambda \sim \prod_{j=1}^K \mathcal{N}(\mathbf{s}_j \mid \mu_{s_j}, \Sigma_{s_j}).$$

- Let $\Sigma_a = \sigma^2 (\sum_{j=1}^K S_j^T S_j)^{-1}$, $\mu_a = \frac{1}{\sigma^2} \Sigma_a \sum_j S_j^T \mathbf{x}_j$, then,

$$\mathbf{a} \mid s, \sigma, \mathbf{v}, \alpha, \lambda \sim \mathcal{N}(a \mid \mu_a, \Sigma_a).$$

- Let $\gamma_\sigma = mK/2$ and $\beta_\sigma = 2 / \sum_j \|\mathbf{x}_j - \mathbf{A} s_j\|_F^2$.

$$\sigma \sim \sqrt{\text{IG}(\gamma_\sigma, \beta_\sigma)}.$$

- Let $\gamma_{v_i} = (\alpha_i + 1)/2$, and $\beta_{v_i,t} = 2/(s_{it}^2 + \alpha_i \lambda_i^2)$.

$$\mathbf{v} \mid \sim \prod_{j=1}^K \prod_{i=1}^n \mathcal{IG}(v_{i,t} \mid \gamma_{v_i}, \beta_{v_i,t}).$$

- Finally, λ can be drawn from $\mathcal{G}(\gamma_{\lambda_i}, \beta_{\lambda_i})$, where $\gamma_{\lambda_i} = \alpha_i K/2$, and $\beta_{\lambda_i} = 2/(\alpha_i S_i)$,
- The conditional distribution of α is not available in a closed form but known up to a constant, one can use any approximation given that the individual values of the hyperparameter are not very critical for the algorithm.

Gibbs Sampler: MacKay Prior with PG Augmentation i

Given the generative model: $\mathbf{X} = \mathbf{A}\mathbf{S}^T + \mathbb{E}$, with

$$s_{ij} \mid \tau_{ij} \sim \mathcal{N}\left(0, \frac{1}{4\tau_{ij}}\right), \quad \tau_{ij} \sim PG(1, 0)$$

The Gibbs sampler proceeds as follows:

(1) **Sample** $s_{ij} \mid \mathbf{A}, \tau, \sigma^2, \mathbf{X}$:

$$s_{ij} \sim \mathcal{N}(\mu_{s,ij}, \Sigma_{s,ij})$$

where

$$\begin{aligned}\Sigma_{s,ij} &= (\text{diag}(\mathbf{A}^\top \mathbf{A})_{jj} / \sigma^2 + 4\tau_{ij})^{-1} \\ \mu_{s,ij} &= \Sigma_{s,ij} \cdot (\mathbf{X}^\top \mathbf{A} - \mathbf{S} \mathbf{A}^\top \mathbf{A} + \mathbf{S} \cdot \text{diag}(\mathbf{A}^\top \mathbf{A}))_{ij}\end{aligned}$$

(2) **Sample** $a_{kj} \mid S, \tau, \sigma^2, \mathbf{X}$:

$$a_{kj} \sim \mathcal{N}(\mu_{a,kj}, \Sigma_{a,kj})$$

where

$$\begin{aligned}\Sigma_{a,kj} &= (\text{diag}(S^T S)_{jj} + (\sigma/\sigma_2)^2)^{-1}, \\ \mu_{a,kj} &= \Sigma_{a,kj} \cdot (\mathbf{X}^T S - \mathbf{A} S^T S + \mathbf{A} \cdot \text{diag}(S^T S))_{kj}\end{aligned}$$

(3) **Sample** $\tau_{ij} \mid s_{ij}$:

$$p(\tau|s) \propto \exp(-2\tau s^2) p_{PG}(\tau|1, 0).$$

(4) **Sample** $\sigma^2 \mid \mathbf{A}, S, \mathbf{X}$:

$$\sigma^2 \sim \text{IG}(\alpha_\sigma, \beta_\sigma), \quad \text{where } \beta_\sigma = \sum_i \|\mathbf{x}_i - \mathbf{A} \mathbf{s}_i\|^2$$

Gibbs Sampler Comparison: t vs. cosh Priors

Fevotte et al. (2004)

Student's-t Prior

- $s_i \mid v_i \sim \mathcal{N}(0, v_i)$
- $v_i \sim \text{Inv-Gamma}$
- Heavy tails from mixing with inverse-gamma
- Gibbs steps involve:
 - Sample s, a, v, λ, σ

Both use Gaussian scale mixtures but with different mixing densities:

MacKay + PG Augmentation

Hyperbolic Secant Prior

- $s_i \mid \tau_i \sim \mathcal{N}(0, 1/(4\tau_i))$
- $\tau_i \sim PG(1, 0)$
- Heavy tails via Pólya-Gamma scale mixture
- Gibbs steps involve:
 - Sample s, a, τ, σ

Inv-Gamma vs. Pólya-Gamma

EM for PG-augmented cosh prior

We aim to maximize the marginal likelihood:

$$p(\mathbf{x} \mid \mathbf{A}) = \int p(\mathbf{x} \mid \mathbf{A}, \mathbf{s}) p(\mathbf{s} \mid \boldsymbol{\tau}) p(\boldsymbol{\tau}) d\mathbf{s} d\boldsymbol{\tau}$$

E-step: Compute the expected value of the latent variable:

$$\mathbb{E}[\tau_i \mid \mathbf{x}, \mathbf{A}] = \frac{1}{4A_{ij}^{-1}x_j} \tanh(|A_{ij}^{-1}x_j|).$$

since τ_i is drawn from a tilted Pólya-Gamma distribution.

M-step: Maximize expected complete-data log-likelihood:

$$\log P(\mathbf{x} \mid \mathbf{A}, \mathbb{E}[\boldsymbol{\tau}]) = C - \log |(\mathbf{A})| + \sum_{i=1}^n \log p(A_{ij}^{-1}x_j \mid \tau_i)$$

This leads to:

$$z_i = \frac{d \log p(a_i \mid \tau_i)}{da_i} = -4\tau_i a_i$$

which defines the update direction for each a_i (gradient-based).

Mixtures and Envelopes

- We have two approaches at our disposal: mixtures and envelopes.

$$p(s) = \int p(s, \tau) d\tau \quad (\textbf{mixture}), \quad \text{and} \quad p(s) = \sup_{\tau > 0} \{p(s, \tau)\} \quad (\textbf{envelope}).$$

- Here, τ is an auxiliary variable [Geman and Yang, 1995].
- (Derivation skipped; see supplementary slides for update rules.)
- Link with Proximal Algorithms/sub-differentials/EM if differentiable
- Speeds up ICA algorithms.
- Gradient based avoids matrix inversion and auxiliary variables dynamically adjusts step-sizes.
- Lets you perform large step sizes at the beginning. Fast convergence.
- Can add Nesterov acceleration.

- We fix $n = 500, d = 4, \epsilon \sim \mathcal{N}(0, \sigma^2), v_{ij} \sim \mathcal{N}(0, \sigma^2)$ with the following data generating process:

$$\begin{aligned} v &\sim \mathcal{N}(0, \sigma_2^2), \quad v \in \mathbb{R}^{d \times d}, \quad \tau \sim PG(b = 1, c = 0), \quad \tau \in \mathbb{R}^{n \times d} \\ s &\sim \mathcal{N}\left(0, \frac{1}{4\tau}\right), \quad s \in \mathbb{R}^{n \times d} \quad x \sim \mathcal{N}(0, \sigma^2) + s \cdot v^T, \quad x \in \mathbb{R}^{n \times d} \end{aligned} \tag{4}$$

- We set $\sigma = 0.01$ and $\sigma_2 = 1$ for the first experiment and compare Mackay's original method with the EM algorithm.
- Then, we scale the first τ by 100 and setting $\sigma = 0.1$ in the data generating process (4), and repeat the comparisons.

Gibbs vs. Truth

Posterior sampling via Gibbs recovers the true s as shown in Fig. 2.

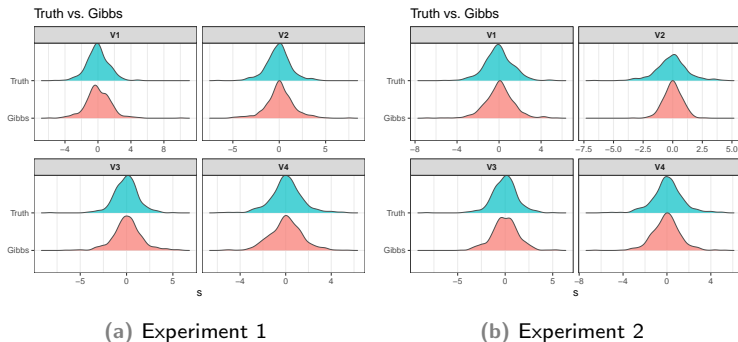


Figure 2: Comparison of the densities for \hat{s} and s

EM vs. MacKay vs. Truth

The densities of $\hat{\mathbf{s}}$ and \mathbf{s} for the candidate methods are shown in Fig. 3. EM and MacKay's algorithms seem to have similar performance in terms of recovering signals.

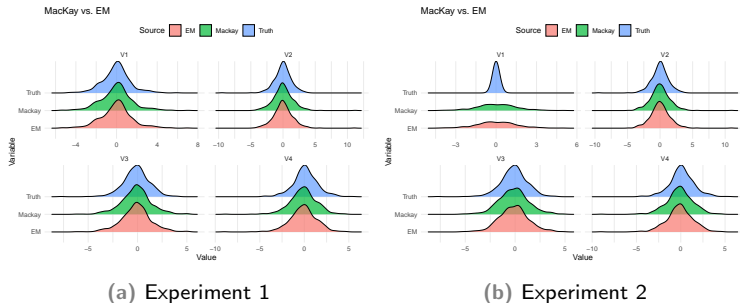
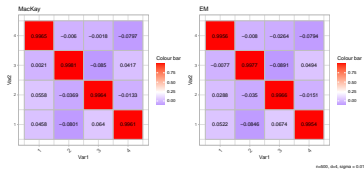


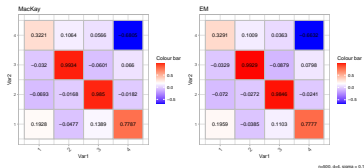
Figure 3: Comparison of the densities for $\hat{\mathbf{s}}$ and \mathbf{s}

Correlation between \hat{s} and s

The correlation map 4 suggests good recovery by the EM algorithm & MacKay.



(a) Experiment 1



(b) Experiment 2

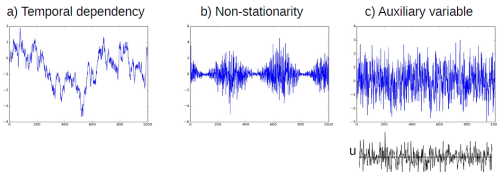
Figure 4: Correlations between \hat{s} and s

Future scope

We are currently extending these methods to nonlinear invertible networks and flow-based models, with layers using ICA-style priors.

Independent components are mixed through an arbitrary but usually smooth function: $f : \mathbb{R}^d \mapsto \mathbb{R}^d$, i.e., $\mathbf{x} = f(\mathbf{s})$, and the goal of nonlinear ICA is to learn a demixing function $g(\cdot)$, such that $\mathbf{z} \doteq g(\mathbf{x})$.

Nonlinear ICA is fundamentally non-identifiable without additional structure: multiple nonlinear functions can yield statistically independent outputs from the same data. ³



³Darmois' 1950s result showed that for any two independent variables ξ_1 and ξ_2 , one can construct infinitely many pairs (y_1, y_2) with $y_1 = f_1(\xi_1, \xi_2)$ and $y_2 = f_2(\xi_1, \xi_2)$ that are also independent.

Thank you!

This is an ongoing work with:



Soham Ghosh



Nicholas G. Polson

Enjoy the rest of your conference!



Preprint: [arXiv:2406.17058](https://arxiv.org/abs/2406.17058) – Horseshoe-type Priors for Independent Component Estimation

Supplementary Slides

Envelopes

- Let the convex conjugate of a function be defined as $\theta^*(\lambda) = \sup_x \{\lambda^T x - \theta(x)\}$ for a closed convex function $\theta(x) : \mathbb{R}^n \mapsto \bar{R}$,
- By the Fenchel–Moreau theorem, the dual relationship holds, i.e. $\theta(x) = \sup_{\lambda} \{\lambda^T x - \theta^*(\lambda)\}$ [Polson and Scott, 2016].

Theorem ([Polson and Scott, 2016])

Let $p(x) \propto \exp\{-\phi(x)\}$ is symmetric in x , and $\theta(x) = \phi(\sqrt{2x})$ for $x > 0$, with completely monotone derivative. Then $p(x)$ admits both normal scale mixture and envelope representation.

$$\exp\{-\phi(x)\} \propto \int_{\mathbb{R}^+} \mathcal{N}(x \mid 0, \lambda^{-1}) p_I(\lambda) d\lambda \propto \sup_{\lambda \geq 0} \{\mathcal{N}(x \mid 0, \lambda^{-1}) p_V(\lambda)\}, \quad (5)$$

where, $p_V(\lambda)$, the variational prior is $p_V(\lambda) \propto \lambda^{-\frac{1}{2}} \exp\{\theta^*(\lambda)\}$, and any optimal value of λ would either lie in the sub-differential of $\theta(x^2/2)$ or satisfy: $\hat{\lambda}(x) = \phi'(x)/x$.

- The envelope approach generalizes the [Ono and Miyabe, 2010] approach. This leads to an iterative algorithm from the recursions:

$$\begin{aligned}x^{(t+1)} &= \arg_x \min_{\tau} \left\{ (y - x)^2 + \tau^{(t)} x^2 \right\}, \\ \tau^{(t+1)} &= \phi'(x^{(t+1)})/x^{(t+1)}, \quad t = 1, 2, \dots\end{aligned}$$

- Recall the envelope representation: $p(x) = \sup_{\tau \geq 0} p(x, \tau)$ where τ is an auxiliary variable [Geman and Yang, 1995].
- The key result for $1/\cosh$ is given in Polson and Scott [2016]:

$$\frac{1}{\cosh(s)} \propto \int_0^\infty \mathcal{N}(s \mid 0, \tau^{-1}) \cdot p_I(\tau) d\tau = \sup_{\tau \geq 0} \left\{ \mathcal{N}(x \mid 0, \tau^{-1}) \cdot p_V(\tau) \right\} \quad (6)$$

where $p_I(\tau) \sim 4 \cdot PG(1, 0)$ is the density of Pólya-Gamma distribution.

- Maximize the likelihood $p(\mathbf{x}|\mathbf{V})$ after integrating out \mathbf{s} (or $\boldsymbol{\tau}$ in our model).
- The update in each step is

$$\Delta(v^{-1}) \propto v - \tanh(v^{-1}x) \cdot x$$

- Simulation performance: recover \mathbf{s} quite well in most settings (the recovered signals have correlations > 0.9 with the true ones.)
- Update rule is simply MM.

Factor Analysis vs. Independent Component Analysis

Both FA and ICA are latent variable models:

$$\text{Data model: } \mathbf{x} = \mathbf{A}\mathbf{s} + \epsilon$$

FA

- Latent factors \mathbf{s} are assumed *Gaussian*
- Identifiability only up to rotation
- Goal: explain *covariance structure*
- Maximum likelihood under Gaussian assumption
- Used in psychology, finance

ICA

- Latent components \mathbf{s} are *non-Gaussian and independent*
- Model is identifiable (up to scale/permutation)
- Goal: recover *independent sources*
- Often uses non-Gaussian priors (e.g., *sech*, *t*, Laplace)
- Used in signal processing, genomics, fMRI

Key distinction: ICA assumes independence and non-Gaussianity; FA does not.