DTU

Signal Representations

Dictionary Learning and Source Separation

The Linear Factor Model

Model Definition

$$X = AZ$$

- X is the observed data matrix ($l \times n$),
- A is the loading matrix ($l \times m$),
- Z is the matrix of latent factors ($m \times n$).

Estimation Challenge

Given only the observed data X, the challenge lies in estimating the matrices A and Z. Additional constraints are required to achieve a unique and meaningful solution.

Singular Value Decomposition (SVD)

Given a matrix $X \in \mathbb{R}^{I \times N}$, SVD decomposes X as:

$$X = UDV^T$$

where U and V are orthogonal matrices, and D is a diagonal matrix of singular values.

Components:

- U is an $I \times r$ matrix (eigenvectors of XX^T)
- V is an $N \times r$ matrix (eigenvectors of X^TX)
- D contains the singular values $\sigma_i = \sqrt{\lambda_i}$, for $i = 1, 2, \ldots, r$

Principal Component Analysis (PCA) I

Principal Components and Data Transformation

For PCA, we transform X using only the top m principal components: U_m and D_m where U_m includes only the first m columns of U and D_m is the top $m \times m$ block of D. We define Z as:

$$Z = D_m V_m^T$$

Thus, the matrix X can be approximated by:

$$X \approx U_m Z$$

where $U_m Z$ represents the data projected onto the subspace defined by the first m principal axes, encapsulating the most significant variance of X.

Dictionary Learning: The k-SVD Algorithm I

The k-SVD Model

Consider a dataset represented by matrix $X \in \mathbb{R}^{l \times N}$, where each column x_n can be expressed as:

$$x = Az,$$

with $A \in \mathbb{R}^{l \times m}$ as the dictionary and $z \in \mathbb{R}^{m}$ as the sparse code. Here, $m > l$ indicating an overcomplete dictionary.

Dictionary Learning: The k-SVD Algorithm II

Optimization Problem

The goal is to find A and Z that minimize:

$$\min_{A,Z} \|X - AZ\|_F^2$$

subject to sparsity constraints $\|z_n\|_0 \leq T_0$ for each n from 1 to N.

Dictionary Learning: The k-SVD Algorithm III

Iterative Approach

Stage 1: Sparse Coding

- Fix A and solve for Z to obtain sparse representations z_n for each observation x_n.

Stage 2: Dictionary Update

- Fix Z and update A by optimizing each dictionary atom one at a time while also updating corresponding elements of Z.

Independent Component Analysis (ICA) I

Main Idea

Decompose a multivariate signal into additive subcomponents that are assumed to be statistically independent.

Mathematical Model

Given observed random variables $\mathbf{x}$ modeled as:

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

where A is an unknown mixing matrix and $\mathbf{s}$ are the independent components. The goal is to estimate both A and $\mathbf{s}$, where components of $\mathbf{s}$ are statistically independent.

Independent Component Analysis (ICA) II

Challenges with Gaussianity

ICA excels particularly where the components s are non-Gaussian. If s were Gaussian, then ICA would not provide more information than PCA because uncorrelated Gaussian variables are also independent, making A non-identifiable.

Estimation Process

- **Model Assumption:** $\hat{\mathbf{s}} := \mathbf{z} = W\mathbf{x}$, where W is an unmixing matrix.
- **Objective:** Make $\mathbf{z}$ as close to $\mathbf{s}$ as possible. For square A, $A = W^{-1}$, assuming invertibility.

ICA Based on Mutual Information I

Initialization and Update Rule

The ICA model starts with a randomly initialized unmixing matrix W, which estimates A^{-1} where the model is $X = AZ$. The update rule for W is given by:

$$W^{(i)} = W^{(i-1)} + \mu_i \left(I - E[\phi(\mathbf{z})\mathbf{z}^T] \right) W^{(i-1)}$$

where μ_i is the learning rate at iteration i.

ICA Based on Mutual Information II

Choice of Non-linear Function $\phi(\mathbf{z})$

The function $\phi(\mathbf{z})$ is chosen based on the Gaussianity of the components:

- For super-Gaussian components (heavy-tailed distributions):

$$\phi(\mathbf{z}) = 2\tanh(\mathbf{z})$$

- For sub-Gaussian components (light-tailed distributions):

$$\phi(\mathbf{z}) = z - \tanh(\mathbf{z})$$

ICA Based on Mutual Information III

Theoretical Background

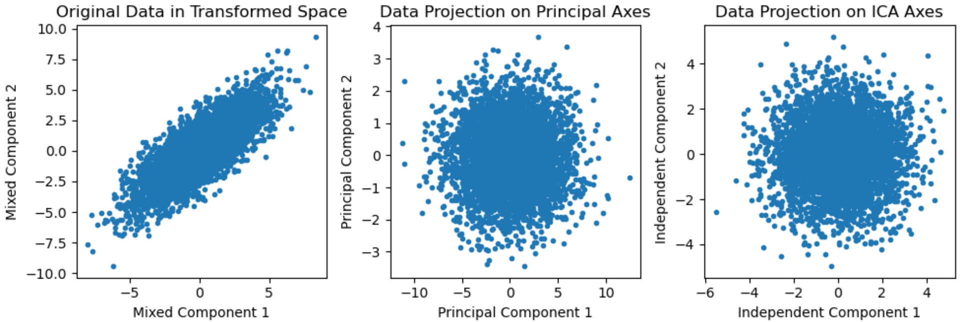
Mutual information $I(x; y)$ measures the amount of information one random variable contains about another. It is defined as:

- **Discrete Variables:**

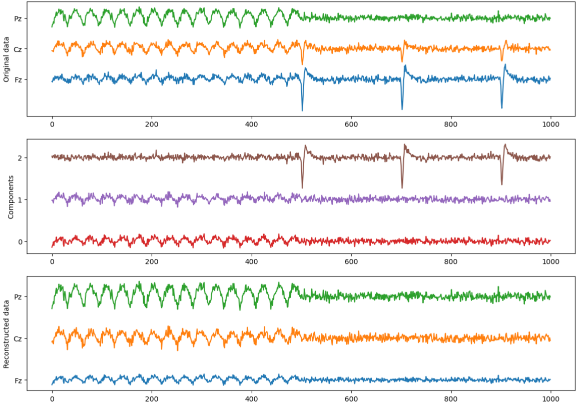
$$I(x; y) = \sum_{x \in X} \sum_{y \in Y} P(x, y) \log \frac{P(x, y)}{P(x)P(y)}$$

The update rule seeks to minimize this mutual information, promoting the independence of the estimated components z.

Exercise 8.3: Two Gaussian Components



Exercise 8.4: ICA for Removing Eye Blinks



8.1 ICA and Gaussian signals I

Given Relationships

Given an orthogonal matrix A with $A^T = A^{-1}$ and a random vector x, we consider the transformation $s = A^T x$.
The PDF of s for I Gaussian sources, $p_s(s)$, is defined as:

$$p_s(\mathbf{s}) = \left(\frac{1}{\sqrt{(2\pi)^{I/2}}}\right) \exp\left(-\frac{||\mathbf{s}||^2}{2}\right)$$

8.1 ICA and Gaussian signals II

PDF Transformation

Using the change of variables formula, the PDF of x, $p_x(x)$, is given by:

$$p_x(x) = \frac{p_s(s)}{|\det(J(x, s))|}$$

where $J(x, s)$ is the Jacobian of the transformation from x to s, which simplifies to $|\det(A)|$ due to the linear transformation.

8.1 ICA and Gaussian signals III

Simplification with Orthogonal Matrix

Since A is orthogonal, $\det(A^T) = \det(A^{-1}) = \pm 1$, hence $|\det(A^T)| = 1$. Moreover, because $||A^T x||^2 = x^T A A^T x = ||x||^2$, we have:

$$p_x(x) = \left(\frac{1}{\sqrt{(2\pi)^{I/2}}} \right) \exp\left(-\frac{||x||^2}{2} \right)$$

which shows that $p_x(x)$ retains the form of a multivariate Gaussian distribution, unchanged in variance.