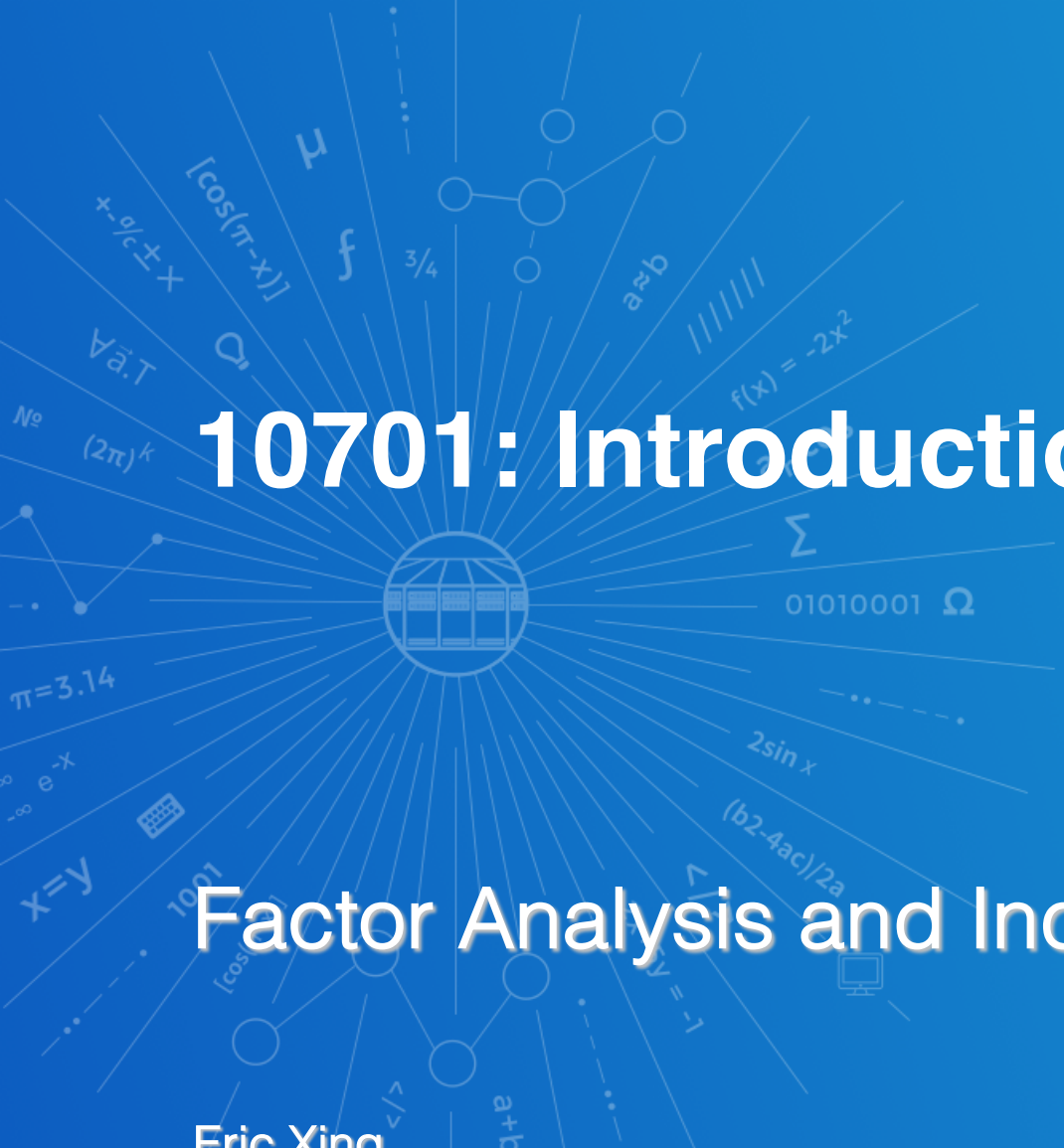


Carnegie Mellon University
School of Computer Science

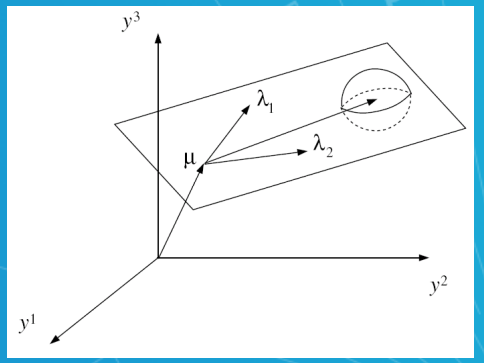
10701: Introduction to Machine Learning

Factor Analysis and Independent Component Analysis

Eric Xing

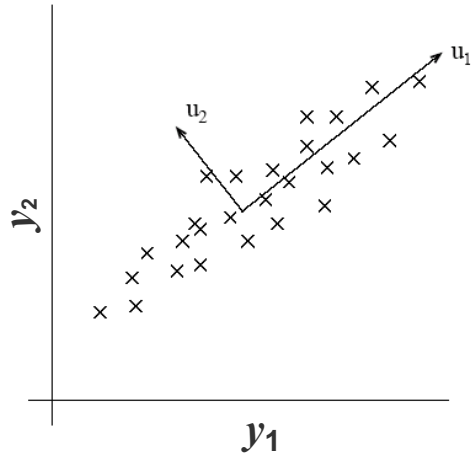
Lecture 16, October 28, 2020

Reading: 12.3-4, C.B book



Recap of PCA

- Popular dimensionality reduction technique
- Project data onto directions of greatest variation



$$u = \arg\max \frac{1}{m}\sum_{i=1}^{m}\left(\vec{y}_i^T u\right)^2$$
$$= \arg\max u^T\left(\frac{1}{m}\sum_{i=1}^{m}\vec{y}_i\vec{y}_i^T\right)u$$
$$= \arg\max\left(u^T Cov(y)u\right)$$

$$\vec{x}_i = \begin{bmatrix} u_1^T\vec{y}_i \\ u_2^T\vec{y}_i \\ \vdots \\ u_q^T\vec{y}_i \end{bmatrix} = U_q^T\vec{y}_i \in \mathrm{R}^q$$

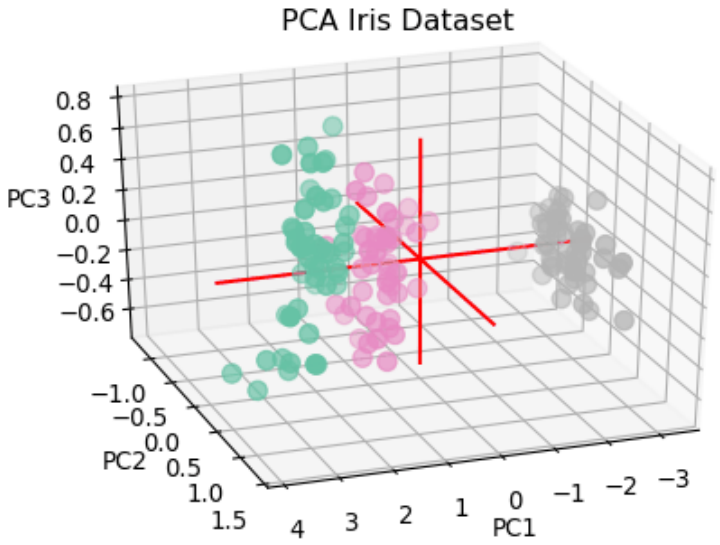
$$\vec{y}_i = U\vec{x}_i$$

- Consequence:
  - x_i are uncorrelated such that the covariance matrix $\frac{1}{m}\sum_{i=1}^{m}\vec{x}_i\vec{x}_i^T$ is $\begin{bmatrix}\gamma_1 & & \\ & \ddots & \\ & & \gamma_q\end{bmatrix}$
  - Truncation error

$$\Sigma_y = \sum_{k=1}^{K}\gamma_k\left(u_k u_k^T\right) \approx \sum_{k=1}^{q}\gamma_k\left(u_k u_k^T\right) = \Sigma_x$$

Recap of PCA

- Popular dimensionality reduction technique
- Project data onto directions of greatest variation



Useful tool for visualising patterns and clusters within the data set, but …

Need centering

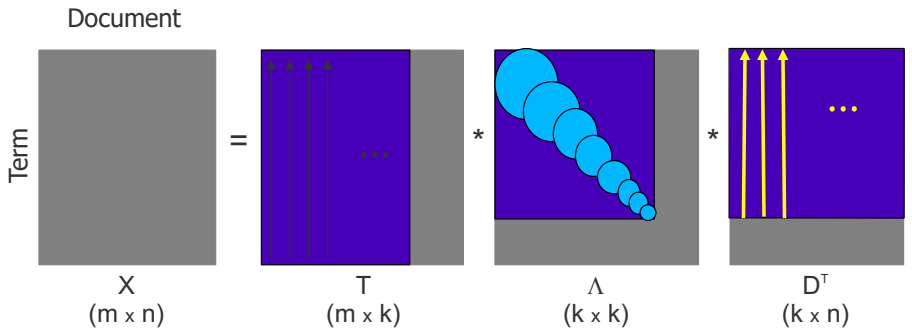
Does not explicitly model data noise

PCA , SVD, and Eigenvalue Decomposition

- The first root is called the prinicipal eigenvalue which has an associated orthonormal ($\mathbf{u}^T\mathbf{u} = 1$) *eigenvector* $\mathbf{u}$
- Subsequent roots are ordered such that $\lambda_1 > \lambda_2 > \ldots > \lambda_M$ with rank($\mathbf{D}$) non-zero values.
- Eigenvectors form an orthonormal basis i.e. $\mathbf{u}_i^T\mathbf{u}_j = \delta_{ij}$
- The eigenvalue decomposition of $\mathbf{X}\mathbf{X}^T = \mathbf{U}\boldsymbol{\Sigma}\mathbf{U}^T$,

  where $\mathbf{U} = [\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_M]$ and $\boldsymbol{\Sigma} = \text{diag}[\lambda_1, \lambda_2, \ldots, \lambda_M]$
- Similarly the eigenvalue decomposition of $\mathbf{X}^T\mathbf{X} = \mathbf{V}\boldsymbol{\Sigma}\mathbf{V}^T$

- The SVD is closely related to the above $\mathbf{X}=\mathbf{U}\,\boldsymbol{\Sigma}^{1/2}\,\mathbf{V}^T$
- The left eigenvectors $\mathbf{U}$, right eigenvectors $\mathbf{V}$,
- singular values = square root of eigenvalues.



Probabilistic PCA

- PCA can be cast as a probabilistic model

$$y_n = \Lambda x_n + \mu + \varepsilon_n \qquad \varepsilon_n \sim \mathsf{N}(0, \sigma^2 I)$$

with q - dimensional latent variables $x_n \sim \mathsf{N}(0, I)$

- The resulting data distribution is

$$y_n \sim \mathsf{N}(\mu, \Lambda\Lambda^T + \sigma^2 I)$$

- Maximum likelihood solution is equivalent to PCA

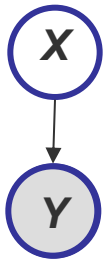
$$\mu^{ML} = \frac{1}{N}\sum_n y_n \qquad \Lambda^{ML} = U_q(\Gamma_q - \sigma^2 I)^{1/2}$$

Diagonal Γ_q contains the top q sample covariance eigen-values and U_q contains associated eigenvectors

Tipping and Bishop, *J. Royal Stat. Soc.* 6, 611 (1999).

Factor analysis

- An unsupervised linear regression model

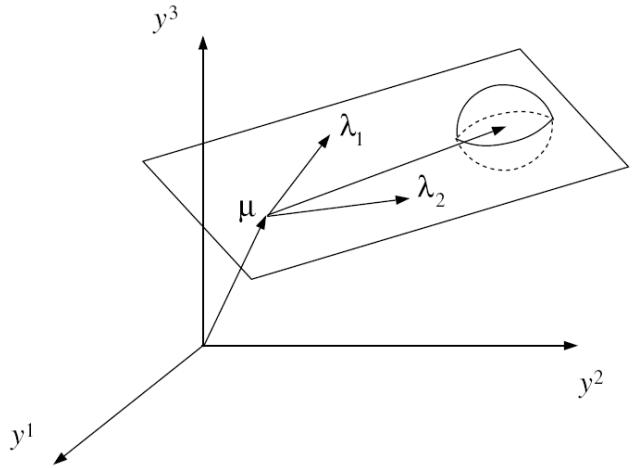


$$p(\mathbf{x}) = \mathcal{N}(\mathbf{x}; 0, I)$$

$$p(\mathbf{y}|\mathbf{x}) = \mathcal{N}(\mathbf{y}; \mu + \Lambda\mathbf{x}, \Psi)$$

where Λ is called a factor loading matrix, and Ψ is diagonal.

- Geometric interpretation



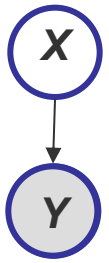
  - To generate data, first generate a point within the manifold then add noise. Coordinates of point are components of latent variable.

Relationship between PCA and FA

- Probabilistic PCA is equivalent to factor analysis with equal noise for every dimension, i.e., $\varepsilon_n \sim$ isotropic Gaussian $\mathsf{N}(0, \sigma^2 I)$
- In factor analysis $\varepsilon_n \sim \mathsf{N}(0, \Psi)$ for a diagonal covariance matrix Ψ
- An iterative algorithm (eg. EM) is required to find parameters if precisions are not known in advance

Factor analysis

- An unsupervised linear regression model

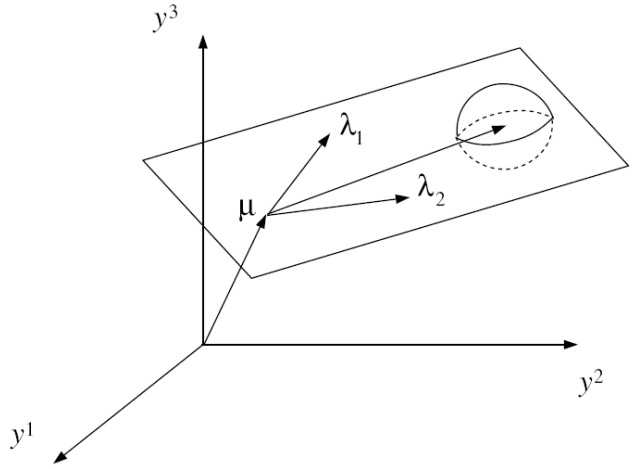


$$p(\mathbf{x}) = \mathcal{N}(\mathbf{x}; 0, I)$$

$$p(\mathbf{y}|\mathbf{x}) = \mathcal{N}(\mathbf{y}; \mu + \Lambda\mathbf{x}, \Psi)$$

where Λ is called a factor loading matrix, and Ψ is diagonal.

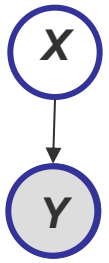
- Geometric interpretation



  - To generate data, first generate a point within the manifold then add noise. Coordinates of point are components of latent variable.

Marginal data distribution

- A marginal Gaussian (e.g., $p(\mathbf{x})$) times a conditional Gaussian (e.g., $p(\mathbf{y}|\mathbf{x})$) is a joint Gaussian
- Any marginal (e.g., $p(\mathbf{y})$ of a joint Gaussian (e.g., $p(\mathbf{x},\mathbf{y})$) is also a Gaussian
  - Since the marginal is Gaussian, we can determine it by just computing its mean and variance. (Assume noise uncorrelated with data.)



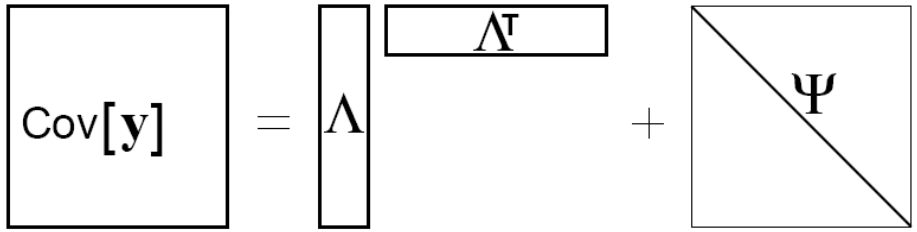
$$
\begin{aligned}
E[\mathbf{Y}] &= E[\mu + \Lambda\mathbf{X} + \mathbf{W}] \qquad \text{where } \mathbf{W} \sim \mathcal{N}(0, \Psi) \\
&= \mu + \Lambda E[\mathbf{X}] + E[\mathbf{W}] \\
&= \mu + 0 + 0 = \mu \\
Var[\mathbf{Y}] &= E\left[(\mathbf{Y} - \mu)(\mathbf{Y} - \mu)^T\right] \\
&= E\left[(\mu + \Lambda\mathbf{X} + \mathbf{W} - \mu)(\mu + \Lambda\mathbf{X} + \mathbf{W} - \mu)^T\right] \\
&= E\left[(\Lambda\mathbf{X} + \mathbf{W})(\Lambda\mathbf{X} + \mathbf{W})^T\right] \\
&= \Lambda E\left[\mathbf{X}\mathbf{X}^T\right]\Lambda^T + E\left[\mathbf{W}\mathbf{W}^T\right] \\
&= \Lambda\Lambda^T + \Psi
\end{aligned}
$$

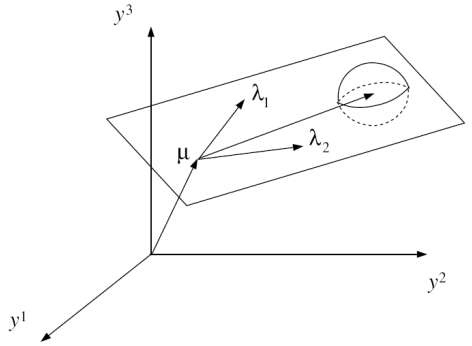
FA = Constrained-Covariance Gaussian

- Marginal density for factor analysis ($\mathbf{y}$ is p-dim, $\mathbf{x}$ is k-dim):

$$p(\mathbf{y} \mid \theta) = \mathcal{N}(\mathbf{y}; \mu, \Lambda\Lambda^T + \Psi)$$

- So the effective covariance is the low-rank outer product of two long skinny matrices plus a diagonal matrix:



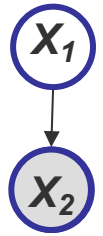


- In other words, factor analysis is just a constrained Gaussian model. (If were not diagonal then we could model any Gaussian and it would be pointless.)

Review:
A primer to multivariate Gaussian

- Multivariate Gaussian density:

$$p(\mathbf{x} \mid \mu, \Sigma) = \frac{1}{(2\pi)^{n/2}|\Sigma|^{1/2}} \exp\left\{-\tfrac{1}{2}(\mathbf{x}-\mu)^T \Sigma^{-1}(\mathbf{x}-\mu)\right\}$$



- A joint Gaussian:

$$p\left(\begin{bmatrix}\mathbf{x}_1\\ \mathbf{x}_2\end{bmatrix} \mid \mu, \Sigma\right) = \mathcal{N}\left(\begin{bmatrix}\mathbf{x}_1\\ \mathbf{x}_2\end{bmatrix} \middle| \begin{bmatrix}\mu_1\\ \mu_2\end{bmatrix}, \begin{bmatrix}\Sigma_{11} & \Sigma_{12}\\ \Sigma_{21} & \Sigma_{22}\end{bmatrix}\right)$$

- How to write down $p(\mathbf{x}_1)$, $p(\mathbf{x}_1|\mathbf{x}_2)$ or $p(\mathbf{x}_2|\mathbf{x}_1)$ using the block elements in μ and Σ?
  - Formulas to remember:

$$p(\mathbf{x}_2) = \mathcal{N}(\mathbf{x}_2 \mid \mathbf{m}_2^m, \mathbf{V}_2^m)$$

$$\mathbf{m}_2^m = \mu_2$$

$$\mathbf{V}_2^m = \Sigma_{22}$$

$$p(\mathbf{x}_1|\mathbf{x}_2) = \mathcal{N}(\mathbf{x}_1 \mid \mathbf{m}_{1|2}, \mathbf{V}_{1|2})$$

$$\mathbf{m}_{1|2} = \mu_1 + \Sigma_{12}\Sigma_{22}^{-1}(\mathbf{x}_2 - \mu_2)$$

$$\mathbf{V}_{1|2} = \Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{21}$$

Review:
Some matrix algebra

- Trace and derivatives
  - Cyclical permutations

$$\mathrm{tr}[A] \overset{\text{def}}{=} \sum_i a_{ii}$$

$$\mathrm{tr}[ABC] = \mathrm{tr}[CAB] = \mathrm{tr}[BCA]$$

  - Derivatives

$$\frac{\partial}{\partial A}\mathrm{tr}[BA] = B^T$$

$$\frac{\partial}{\partial A}\mathrm{tr}[x^T Ax] = \frac{\partial}{\partial A}\mathrm{tr}[xx^T A] = xx^T$$

- Determinants and derivatives

$$\frac{\partial}{\partial A}\log|A| = A^{-T}$$

FA joint distribution

- Model

$$p(\mathbf{x}) = \mathcal{N}(\mathbf{x}; 0, I)$$
$$p(\mathbf{y}|\mathbf{x}) = \mathcal{N}(\mathbf{y}; \mu + \Lambda\mathbf{x}, \Psi)$$

- Covariance between $\mathbf{x}$ and $\mathbf{y}$

$$\begin{aligned} Cov[\mathbf{X},\mathbf{Y}] &= E\left[(\mathbf{X}-0)(\mathbf{Y}-\mu)^T\right] = E\left[\mathbf{X}(\mu+\Lambda\mathbf{X}+\mathbf{W}-\mu)^T\right] \\ &= E\left[\mathbf{X}\mathbf{X}^T\Lambda^T + \mathbf{X}\mathbf{W}^T\right] \\ &= \Lambda^T \end{aligned}$$

- Hence the joint distribution of $\mathbf{x}$ and $\mathbf{y}$:

$$p\left(\begin{bmatrix}\mathbf{x}\\ \mathbf{y}\end{bmatrix}\right) = \mathcal{N}\left(\begin{bmatrix}\mathbf{x}\\ \mathbf{y}\end{bmatrix} \middle| \begin{bmatrix}0\\ \mu\end{bmatrix}, \begin{bmatrix} I & \Lambda^T \\ \Lambda & \Lambda\Lambda^T + \Psi \end{bmatrix}\right)$$

- Assume noise is uncorrelated with data or latent variables.

Inference in Factor Analysis

- Apply the Gaussian conditioning formulas to the joint distribution we derived above, where

$$\Sigma_{11} = I$$
$$\Sigma_{12} = \Sigma_{12}^{T} = \Lambda^{T}$$
$$\Sigma_{22} = \left(\Lambda\Lambda^{T} + \Psi\right)$$

we can now derive the posterior of the latent variable $\mathbf{x}$ given observation $\mathbf{y}$, $p(\mathbf{x}|\mathbf{y}) = \mathcal{N}(\mathbf{x} \mid \mathbf{m}_{1|2}, \mathbf{V}_{1|2})$, where

$$\mathbf{m}_{1|2} = \mu_1 + \Sigma_{12}\Sigma_{22}^{-1}(\mathbf{y} - \mu_2) = \Lambda^{T}\left(\Lambda\Lambda^{T} + \Psi\right)^{-1}(\mathbf{y} - \mu)$$

$$\mathbf{V}_{1|2} = \Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{21} = I - \Lambda^{T}\left(\Lambda\Lambda^{T} + \Psi\right)^{-1}\Lambda$$

Applying the matrix inversion lemma $(E - FH^{-1}G)^{-1} = E^{-1} + E^{-1}F\left(H - GE^{-1}F\right)^{-1}GE^{-1}$

$$\Rightarrow \quad \mathbf{V}_{1|2} = \left(I + \Lambda^{T}\Psi^{-1}\Lambda\right)^{-1} \qquad \mathbf{m}_{1|2} = \mathbf{V}_{1|2}\Lambda^{T}\Psi^{-1}(\mathbf{y} - \mu)$$

- Here we only need to invert a matrix of size $|\mathbf{x}| \times |\mathbf{x}|$, instead of $|\mathbf{y}| \times |\mathbf{y}|$.

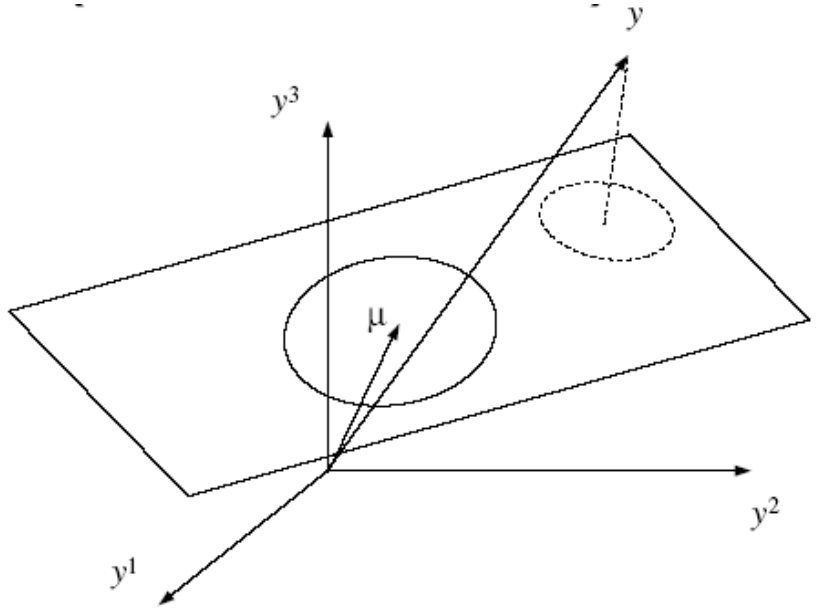
Geometric interpretation: inference is linear projection

- The posterior is:

$$p(\mathbf{x}|\mathbf{y}) = \mathcal{N}(\mathbf{x}; \mathbf{m}_{1|2}, \mathbf{V}_{1|2})$$

$$\mathbf{V}_{1|2} = \left(I + \Lambda^T \Psi^{-1} \Lambda\right)^{-1} \qquad \mathbf{m}_{1|2} = \mathbf{V}_{1|2} \Lambda^T \Psi^{-1} (\mathbf{y} - \mu)$$

- Posterior covariance does not depend on observed data $\mathbf{y}$!
- Computing the posterior mean is just a linear operation:



EM for Factor Analysis

- Incomplete data log likelihood function (marginal density of y)

$$\ell(\theta, D) = -\frac{N}{2}\log\left|\Lambda\Lambda^T + \Psi\right| - \frac{1}{2}\sum_n (y_n - \mu)^T \left(\Lambda\Lambda^T + \Psi\right)^{-1}(y_n - \mu)$$

$$= -\frac{N}{2}\log\left|\Lambda\Lambda^T + \Psi\right| - \frac{1}{2}\mathrm{tr}\left[\left(\Lambda\Lambda^T + \Psi\right)^{-1}\mathbf{S}\right], \qquad \text{where } \mathbf{S} = \sum_n (y_n - \mu)(y_n - \mu)^T$$

  - Estimating m is trivial: $\hat{\mu}^{ML} = \frac{1}{N}\sum_n y_n$
  - Parameters Λ and Ψ are coupled nonlinearly in log-likelihood

- Complete log likelihood

$$\ell_c(\theta, D) = \sum_n \log p(x_n, y_n) = \sum_n \log p(x_n) + \log p(y_n \mid x_n)$$

$$= -\frac{N}{2}\log|I| - \frac{1}{2}\sum_n x_n^T x_n - \frac{N}{2}\log|\Psi| - \frac{1}{2}\sum_n (y_n - \Lambda x_n)^T \Psi^{-1}(y_n - \Lambda x_n)$$

$$= -\frac{N}{2}\log|\Psi| - \frac{1}{2}\sum_n \mathrm{tr}\left[x_n x_n^T\right] - \frac{N}{2}\mathrm{tr}\left[\mathbf{S}\Psi^{-1}\right], \qquad \text{where } \mathbf{S} = \frac{1}{N}\sum_n (y_n - \Lambda x_n)(y_n - \Lambda x_n)^T$$

E-step for Factor Analysis

- Compute $\left\langle \ell_c(\theta, D) \right\rangle_{p(x|y)}$

$$\left\langle \ell_c(\theta, D) \right\rangle = -\frac{N}{2}\log|\Psi| - \frac{1}{2}\sum_n \mathrm{tr}\left[\left\langle X_n X_n^T \right\rangle\right] - \frac{N}{2}\mathrm{tr}\left[\langle \mathbf{S} \rangle \Psi^{-1}\right]$$

$$\langle \mathbf{S} \rangle = \frac{1}{N}\sum_n (y_n y_n^T - y_n \left\langle X_n^T \right\rangle \Lambda^T - \Lambda \left\langle X_n^T \right\rangle y_n^T + \Lambda \left\langle X_n X_n^T \right\rangle \Lambda^T)$$

$$\langle X_n \rangle = E[X_n \mid y_n]$$

$$\left\langle X_n X_n^T \right\rangle = Var[X_n \mid y_n] + E[X_n \mid y_n] E[X_n \mid y_n]^T$$

  - Recall that we have derived:

$$\mathbf{V}_{1|2} = \left(I + \Lambda^T \Psi^{-1} \Lambda\right)^{-1} \qquad \mathbf{m}_{1|2} = \mathbf{V}_{1|2} \Lambda^T \Psi^{-1} (\mathbf{y} - \mu)$$

$$\Rightarrow \quad \langle X_n \rangle = \mathbf{m}_{x_n|y_n} = \mathbf{V}_{1|2} \Lambda^T \Psi^{-1} (y_n - \mu) \quad \text{and} \quad \left\langle X_n X_n^T \right\rangle = \mathbf{V}_{1|2} + \mathbf{m}_{x_n|y_n} \mathbf{m}_{x_n|y_n}^T$$

M-step for Factor Analysis

- Take the derivates of the expected complete log likelihood wrt. parameters.
  - Using the trace and determinant derivative rules:

$$\frac{\partial}{\partial \Psi^{-1}}\langle \ell_c \rangle = \frac{\partial}{\partial \Psi^{-1}}\left(-\frac{N}{2}\log|\Psi| - \frac{1}{2}\sum_n \mathrm{tr}\left[\langle X_n X_n^T\rangle\right] - \frac{N}{2}\mathrm{tr}\left[\langle \mathbf{S}\rangle \Psi^{-1}\right]\right)$$

$$= \frac{N}{2}\Psi - \frac{N}{2}\langle \mathbf{S}\rangle \qquad \Rightarrow \qquad \Psi^{t+1} = \langle \mathbf{S}\rangle$$

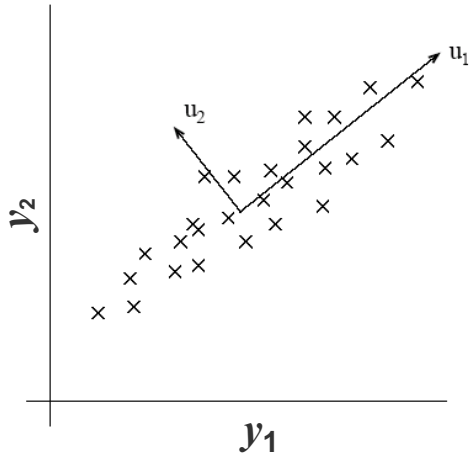
$$\frac{\partial}{\partial \Lambda}\langle \ell_c \rangle = \frac{\partial}{\partial \Lambda}\left(-\frac{N}{2}\log|\Psi| - \frac{1}{2}\sum_n \mathrm{tr}\left[\langle X_n X_n^T\rangle\right] - \frac{N}{2}\mathrm{tr}\left[\langle \mathbf{S}\rangle \Psi^{-1}\right]\right) = -\frac{N}{2}\Psi^{-1}\frac{\partial}{\partial \Lambda}\langle \mathbf{S}\rangle$$

$$= -\frac{N}{2}\Psi^{-1}\frac{\partial}{\partial \Lambda}\left(\frac{1}{N}\sum_n (y_n y_n^T - y_n\langle X_n^T\rangle \Lambda^T - \Lambda\langle X_n^T\rangle y_n^T + \Lambda\langle X_n X_n^T\rangle \Lambda^T)\right)$$

$$= \Psi^{-1}\sum_n y_n\langle X_n^T\rangle - \Psi^{-1}\Lambda\sum_n \langle X_n X_n^T\rangle \qquad \Rightarrow \qquad \Lambda^{t+1} = \left(\sum_n y_n\langle X_n^T\rangle\right)\left(\sum_n \langle X_n X_n^T\rangle\right)^{-1}$$

Comparison of PCA and FA

- PCA

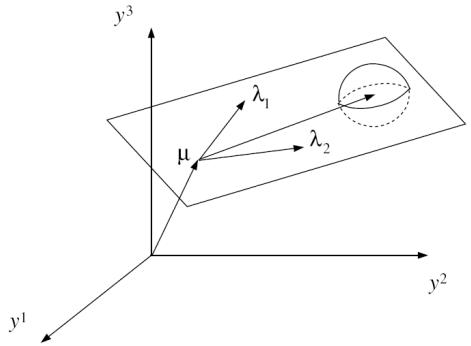


$$y_n = Ux_n$$

$$u = \arg\max\left(u^T Cov(y)u\right)$$

$$\vec{x}_i = \begin{bmatrix} u_1^T \vec{y}_i \\ u_2^T \vec{y}_i \\ \vdots \\ u_k^T \vec{y}_i \end{bmatrix} = U_q^T \vec{y}_i \in \mathrm{R}^q$$

- FA



$$y_n = \Lambda x_n + \mu + \varepsilon_n$$

$$\varepsilon_n \sim \mathsf{N}(0, \Psi)$$

$$\langle X_n \rangle = \mathbf{m}_{x_n|y_n} = \mathbf{V}_{1|2}\Lambda^T \Psi^{-1}(y_n - \mu)$$

and $$\langle X_n X_n^T \rangle = \mathbf{V}_{1|2} + \mathbf{m}_{x_n|y_n}\mathbf{m}_{x_n|y_n}^T$$

$$\Lambda^{t+1} = \left(\sum_n y_n \langle X_n^T \rangle\right)\left(\sum_n \langle X_n X_n^T \rangle\right)^{-1}$$

Comparison of PCA and FA

- PCA

$$u = \arg\max\left(u^T Cov(y) u\right)$$

$$\vec{x}_i = \begin{bmatrix} u_1^T \vec{y}_i \\ u_2^T \vec{y}_i \\ \vdots \\ u_k^T \vec{y}_i \end{bmatrix} = U_q^T \vec{y}_i \in \mathrm{R}^q$$

  - SVD on a $K'K$ matrix
  - Covariant under rotation: Ay
  - Principle axis can be found incrementally

- FA

$$\langle X_n \rangle = \mathbf{m}_{x_n|y_n} = \mathbf{V}_{1|2} \Lambda^T \Psi^{-1} (y_n - \mu)$$

and $$\left\langle X_n X_n^T \right\rangle = \mathbf{V}_{1|2} + \mathbf{m}_{x_n|y_n} \mathbf{m}_{x_n|y_n}^T$$

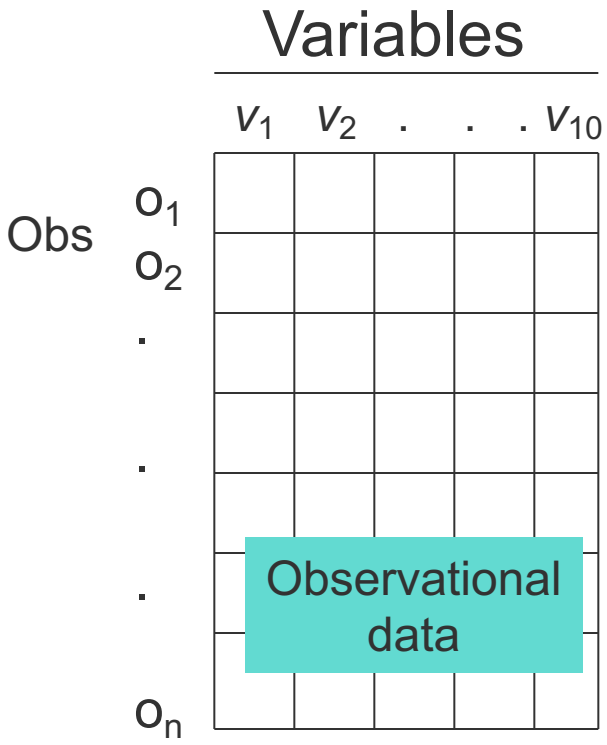
$$\Lambda^{t+1} = \left(\sum_n y_n \left\langle X_n^T \right\rangle \right) \left(\sum_n \left\langle X_n X_n^T \right\rangle \right)^{-1}$$

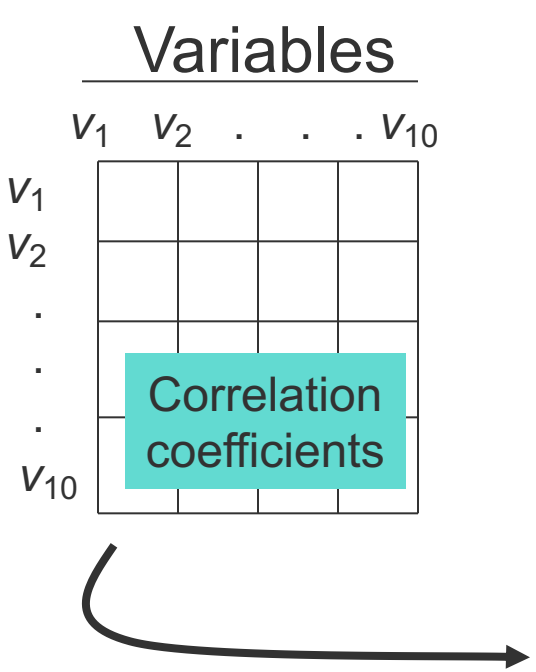
$$\Psi^{t+1} = \langle \mathbf{S} \rangle$$

  - Invert a $q'q$ matrix
  - Covariant under rescaling: $diag(\alpha)y$
  - Neither of the factors found by a two-factor model is necessarily the same as that found by a single factor model, and …

Example:

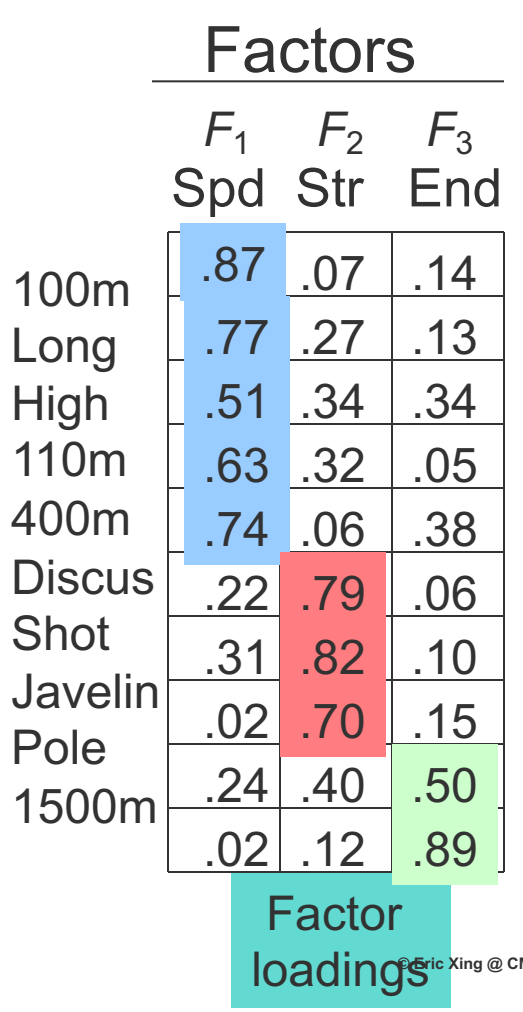
Original data matrix → Correlation matrix → Factor matrix





Decisions

- Factoring method
- # of factors to retain
- Factor rotation

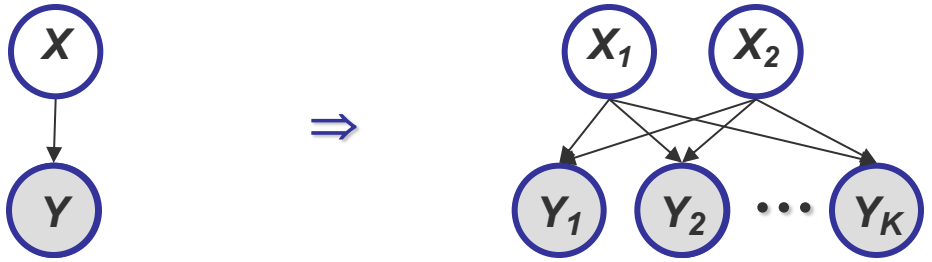


Model Invariance and Identifiability

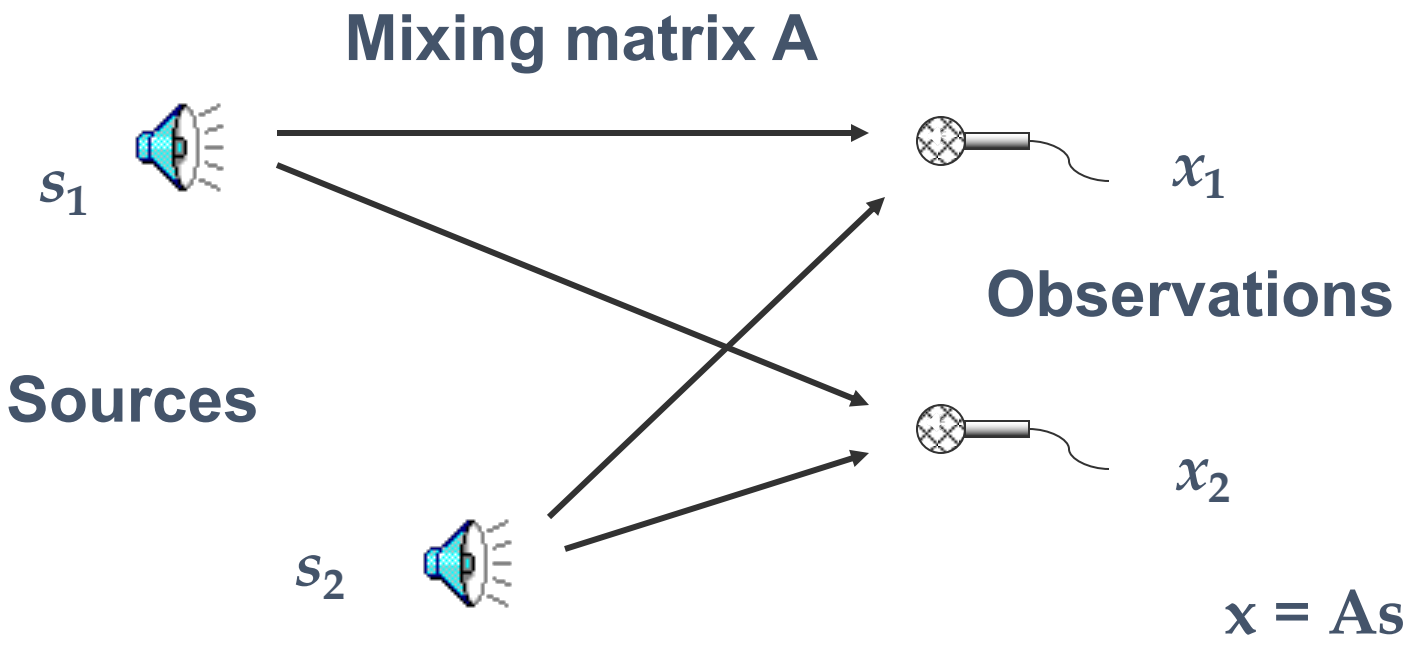
- There is *degeneracy* in the FA model.
- Since Λ only appears as outer product $\Lambda\Lambda^T$, the model is invariant to rotation and axis flips of the latent space.
- We can replace Λ with ΛQ for any orthonormal matrix Q and the model remains the same: $(\Lambda Q)(\Lambda Q)^T=\Lambda(QQ^T)\Lambda^T=\Lambda\Lambda^T$.
- This means that there is no "one best" setting of the parameters. An infinite number of parameters all give the ML score!
- Such models are called un-identifiable since two people both fitting ML parameters to the identical data will not be guaranteed to identify the same parameters.

Independent Components Analysis (ICA)

- ICA is similar to FA, except it assumes the latent source has non-Gaussian density.
- Hence ICA can extract higher order moments (not just second order).
- It is commonly used to solve blind source separation (cocktail party problem).

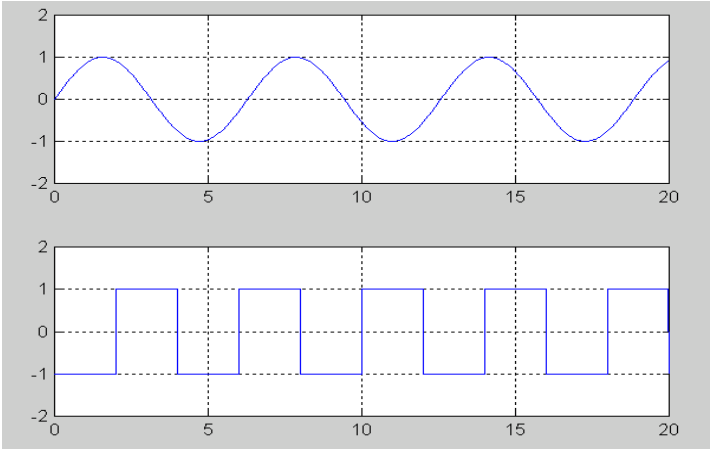


The simple “Cocktail Party” Problem

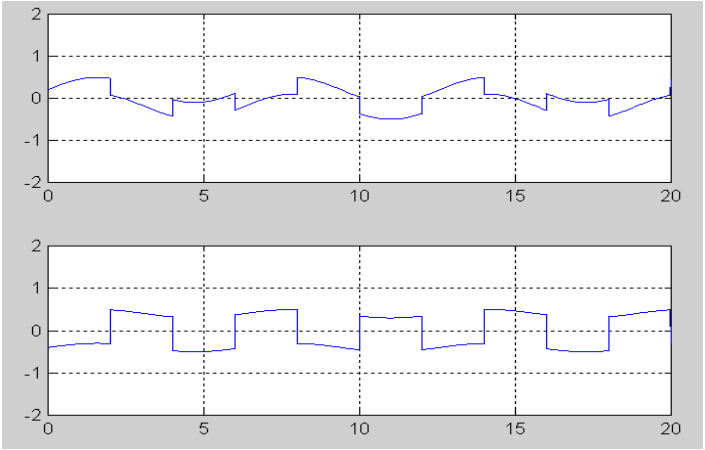


n sources, m=*n* observations

Motivation



Two Independent Sources

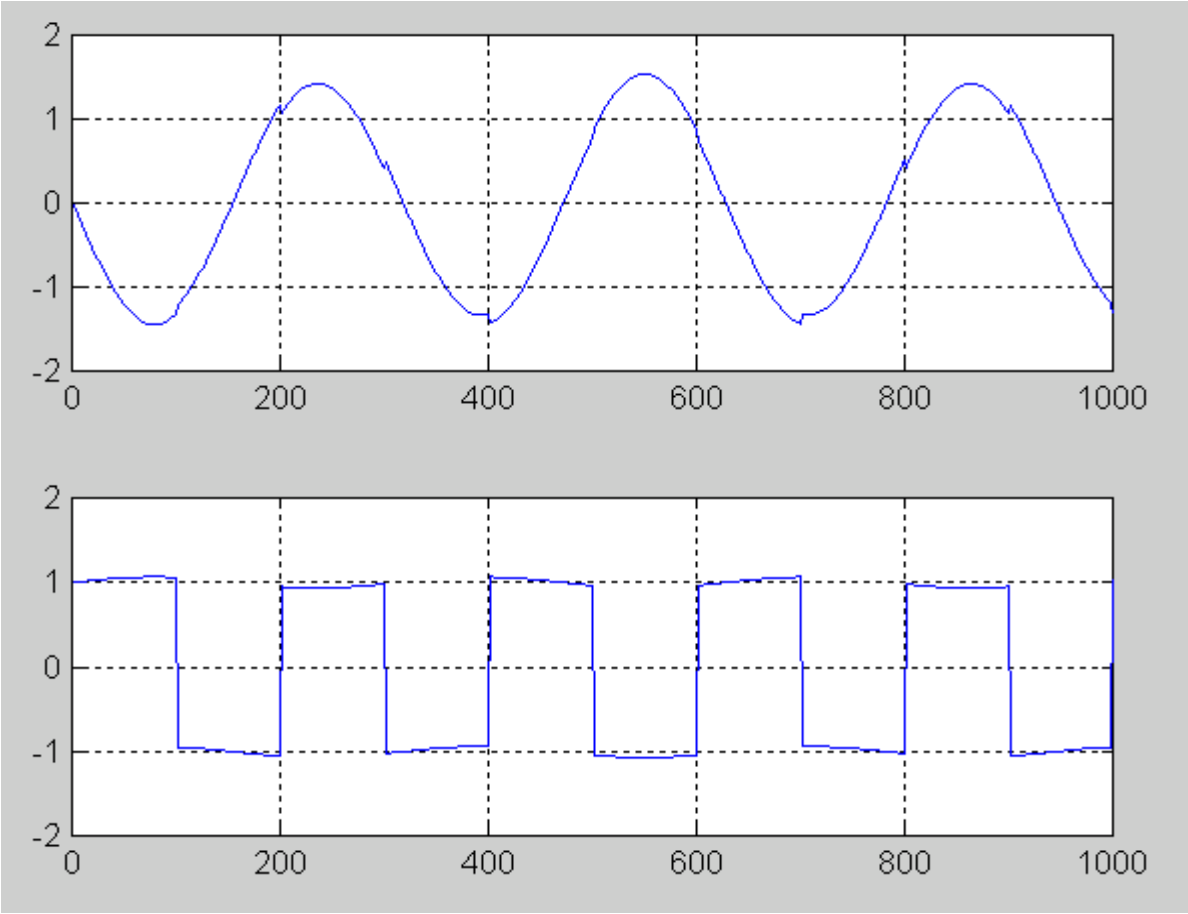


Mixture at two Mics

$$x_1(t) = a_{11}s_1(t) + a_{12}s_2(t)$$
$$x_2(t) = a_{21}s_1(t) + a_{22}s_2(t)$$

a_{ij} ... Depend on the distances of the microphones from the speakers

Motivation



Get the Independent Signals out of the Mixture

Blind Source Separation

- Suppose that there are k unknown independent sources

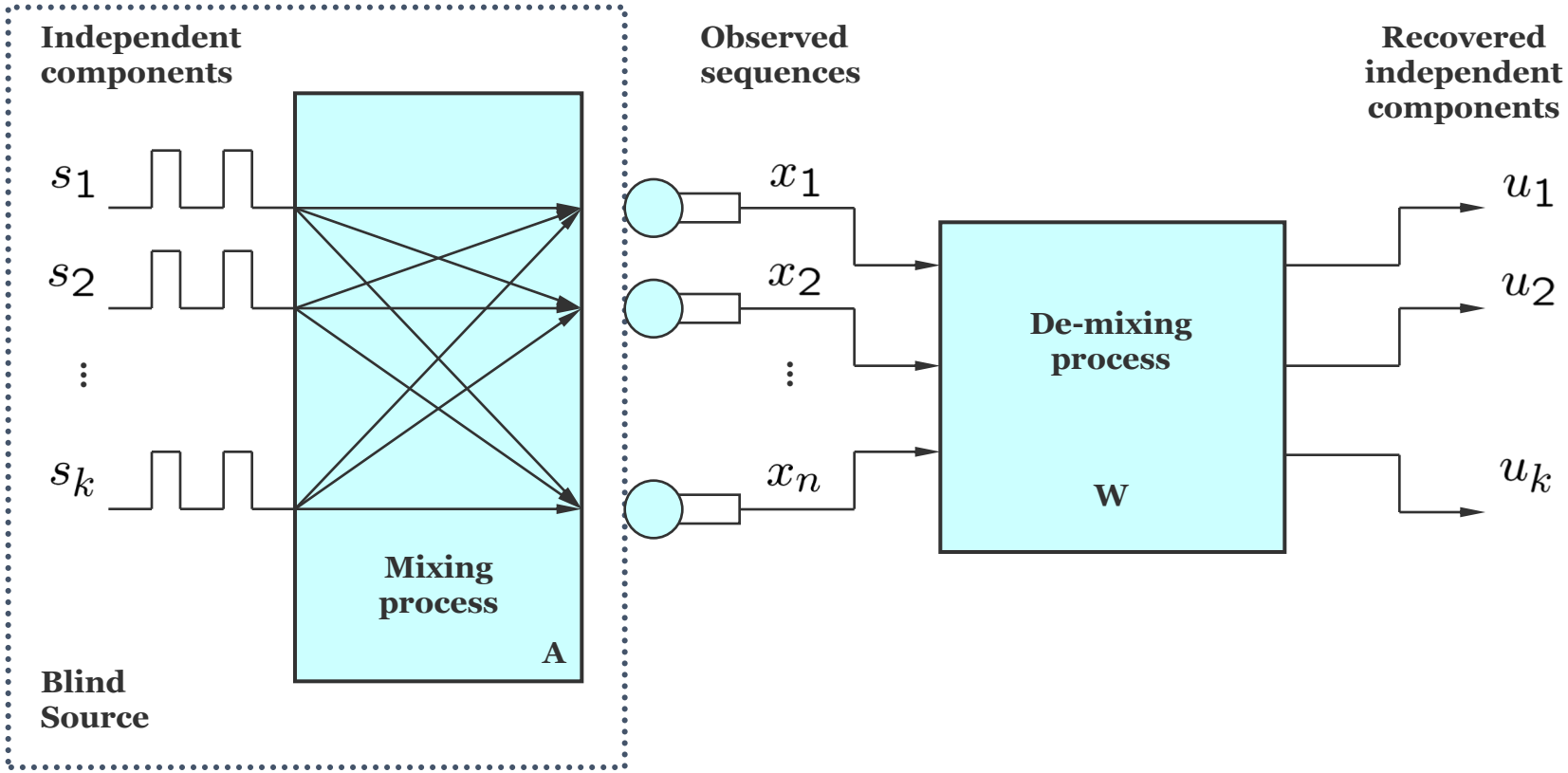
$$\mathbf{s}(t) = \left[s_1(t), \ldots, s_k(t)\right]^T \qquad \text{with} \qquad E\left[\mathbf{s}(t)\right] = \mathbf{1}$$

- A data vector x(t) is observed at each time point t, such that

$$\mathbf{x}(t) = \mathbf{A}\mathbf{s}(t)$$

where $\mathbf{A}$ is a $n \times k$ full rank scalar matrix

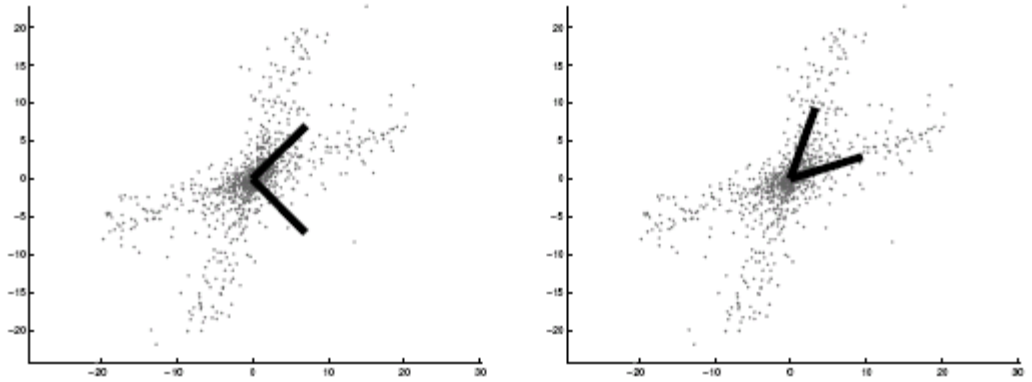
Blind source separation



ICA versus PCA (and FA)

- Similarity
  - Feature extraction
  - Dimension reduction

- Difference
  - PCA uses up to second order moment of the data to produce uncorrelated components
  - ICA strives to generate components as independent as possible



Probability 101: Pearson's correlation

- Normalized covariance

$$\rho(X,Y) = \frac{\mathrm{Cov}(X,Y)}{\sqrt{\mathrm{Var}(X)}\sqrt{\mathrm{Var}(Y)}}$$

- Captures linear dependency
  - Linear regression from X to Y gives $\beta = \frac{\mathrm{Cov}(X,Y)}{\mathrm{Var}(X)}$

- Important properties:
  - X ⫫ Y implies ρ(X,Y) = 0 (Why?)
  - ρ(X,Y) = 0 does not imply X ⫫ Y (Counterexamples?)

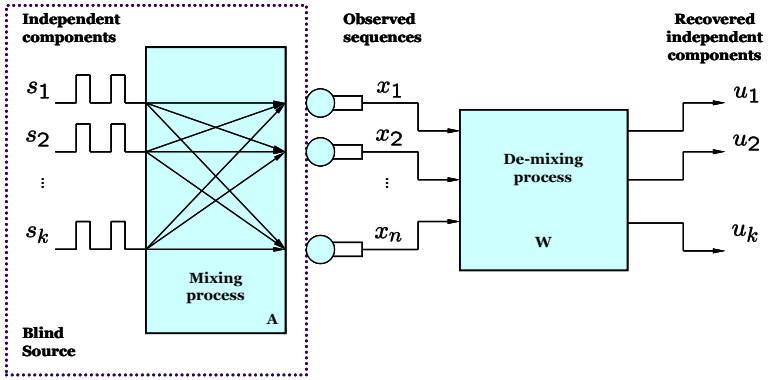
- Q1: Is there any measure that implies independence?
- Q2: What kind of dependency should they consider?

Problem formulation

- The goal of ICA is to find a linear mapping $\mathbf{W}$ such that the unmixed sequences $\mathbf{u}$

$$\mathbf{u}(t) = \mathbf{W}\mathbf{x}(t) = \mathbf{W}\mathbf{A}\mathbf{s}(t)$$

are maximally statistically independent



- Find some

$$\mathbf{V} = \mathbf{W}\mathbf{A} = \mathbf{P}\mathbf{C}$$

where $\mathbf{C}$ is a diagonal matrix and $\mathbf{P}$ is a permutation matrix.

Principle of ICA: Nongaussianity

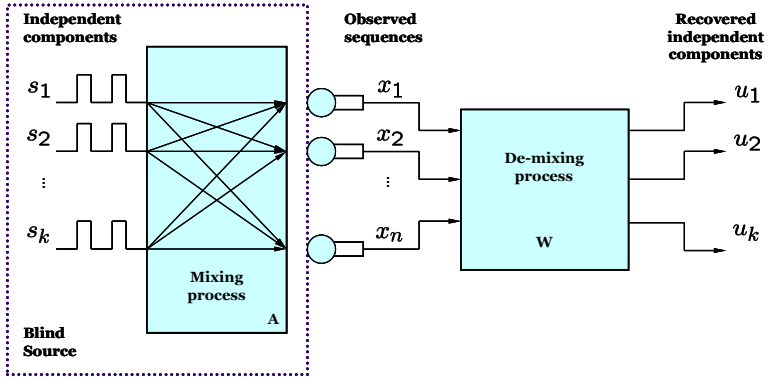
- The fundamental restriction in ICA is that the independent components must be nongaussian for ICA to be possible.

- This is because gaussianity is invariant under orthogonal transformation and hence make the matrix $\mathbf{A}$ not identifiable for gaussian independent components.

Measures of nongaussianity (1)

- Kurtosis
  - $$\text{Kurt}(y) = E\left\{y^4\right\} - 3\left(E\left\{y^2\right\}\right)^2$$
  - Kurtosis can be very sensitive to outliers, when its value has to be estimate from a measured sample.
- Mutual information
- Negative Entropy

FastICA — Preprocessing



- Centering:
  - Make the x-s mean 0 variables
- Whitening
  - Transform the observed vector x linearly so that it has unit variance:

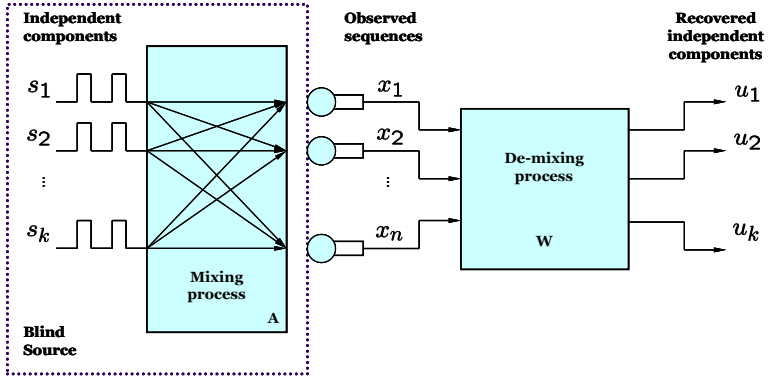
$$\mathbf{E}\left\{\tilde{\mathbf{x}}\tilde{\mathbf{x}}^T\right\} = \mathbf{I}$$

  - One can show that:

$$\tilde{\mathbf{x}} = \mathbf{E}\mathbf{D}^{-1/2}\mathbf{E}^T\mathbf{x} = \widetilde{\mathbf{A}}\mathbf{s}$$

where $\mathbf{E}\left\{\mathbf{x}\mathbf{x}^T\right\} = \mathbf{E}\mathbf{D}\mathbf{E}^T$

FastICA algorithm



- A fixed-point iteration scheme, that maximizes a measure of non-Gaussianity of the rotated components. Non-Gaussianity serves as a proxy for statistical independence

- To measure non-Gaussianity, FastICA relies on a nonquadratic nonlinear function f(u), its first derivative g(u), and its second derivative g'(u).

$$f(u) = \log\cosh(u),$$
$$f(u) = -e^{-u^2/2}$$

Algorithm FastICA

Input: C Number of desired components

Input: $\mathbf{X} \in \mathbb{R}^{N \times M}$ Prewhitened matrix, where each column represents an N-dimensional sample, where $C <= N$

Output: $\mathbf{W} \in \mathbb{R}^{N \times C}$ Un-mixing matrix where each column projects $\mathbf{X}$ onto independent component.

Output: $\mathbf{S} \in \mathbb{R}^{C \times M}$ Independent components matrix, with M columns representing a sample with C dimensions.

for p **in** 1 to C:

$\mathbf{w_p} \leftarrow$ *Random vector of length N*

while $\mathbf{w_p}$ changes

$$\mathbf{w_p} \leftarrow \frac{1}{M}\mathbf{X}g(\mathbf{w_p}^T\mathbf{X})^T - \frac{1}{M}g'(\mathbf{w_p}^T\mathbf{X})\mathbf{1_M}\mathbf{w_p}$$

$$\mathbf{w_p} \leftarrow \mathbf{w_p} - \sum_{j=1}^{p-1}(\mathbf{w_p}^T\mathbf{w_j})\mathbf{w_j}$$

$$\mathbf{w_p} \leftarrow \frac{\mathbf{w_p}}{\|\mathbf{w_p}\|}$$

output $\mathbf{W} \leftarrow [\mathbf{w_1}, \ldots, \mathbf{w_C}]$

output $\mathbf{S} \leftarrow \mathbf{W}^T\mathbf{X}$

Summary

- There has been a wide discussion about the application of Independence Component Analysis (ICA) in Signal Processing, Neural Computation and Finance.
- First introduced as a novel tool to separate blind sources in a mixed signal.
- The Basic idea of ICA is to reconstruct from observation sequences the hypothesized independent original sequences.