Probabilistic Graphical Models for Image Analysis - Lecture 7

Stefan Bauer

2nd November 2018

Max Planck ETH Center for Learning Systems

Overview

1. Factor Analysis
2. Principal Component Analysis
3. Connection with State Space Models

Motivation

- Often there are some unknown *underlying causes* of the data.
- Continuous factors which control the data we observe *data manifold* (or subspace).
- Training continuous latent variable models is often called *dimensionality reduction*, since there are typically many fewer latent dimensions.
- Examples (see reference) PCA, Factor Analysis, ICA
- Reason for choosing continuous representation is often motivated by efficiency.
- Mixture models uses discrete class variable: clustering
- Simplest case: *linear* subspace and underlying latent variable with a Gaussian distribution.

Recall - Principal Component Analysis

Limitations of PCA

- No probabilistic model for observed data
- Difficulty to deal with missing data
- Naive PCA uses a simplistic distance function to assess covariance.

Motivation for probabilistic PCA:

- address limitations
- allows to combine multiple PCA models as probabilistic mixtures

"... the definition of a likelihood measure enables a comparison with other probabilistic techniques, while facilitating statistical testing and permitting the application of Bayesian models..."*

*Tipping and Bishop: Probabilistic Principal Component Analysis, Royal Statistical Society: Series B, 1999

Factor Analysis

Factor Analysis Model

$z \in \mathbb{R}^k$ is a latent variable and y is the observed data:

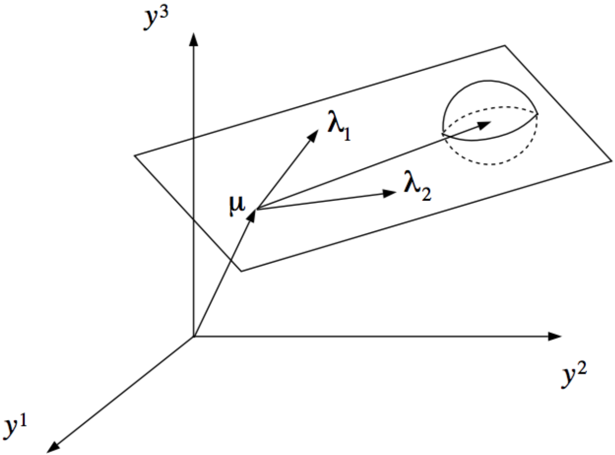
$$z \sim \mathcal{N}(0, 1)$$
$$x|z \sim \mathcal{N}(\mu + \Lambda z, \Psi)$$

Parameters of our model are thus:

- $\mu \in \mathbb{R}^n$
- $\Lambda \in \mathbb{R}^{n \times k}$
- Diagonal matrix $\Psi \in \mathbb{R}^{n \times n}$

Note: Dimensionality reduction since k is chosen smaller than n.

Illustration



where y are the observations.

Equivalent formulation

$$z \sim \mathcal{N}(0, 1)$$
$$\varepsilon \sim \mathcal{N}(0, \Psi)$$
$$x = \mu + \Lambda z + \varepsilon$$

where ε and z are independent.

Joint model:

$$\begin{bmatrix} z \\ x \end{bmatrix} \sim \mathcal{N}(\mu_{zx}, \Sigma)$$

Goal: Identify μ_{zx} and Σ.

Factor Analysis

Joint model:

$$\begin{bmatrix} z \\ x \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} 0 \\ \mu \end{bmatrix}, \begin{bmatrix} 1 & \Lambda^\intercal \\ \Lambda & \Lambda\Lambda^\intercal + \Psi \end{bmatrix} \right)$$

Marginal Distribution

$$x \sim \mathcal{N}(\mu, \Lambda\Lambda^\intercal + \Psi)$$

Log-Likelihood of parameters

$$l(\mu, \Lambda, \Psi) = \log \prod_{i=1}^{m} \frac{1}{(2\pi)^{\frac{n}{2}} |\Lambda\Lambda^\intercal + \Psi|^{\frac{1}{2}}} \exp\left(-\frac{1}{2} (x^{(i)} - \mu)^\intercal (\Lambda\Lambda^\intercal + \Psi)^{-1} (x^{(i)} - \mu) \right)$$

EM for Factor Analysis

Maximum likelihood learning using EM:

- **E-Step**: $q^{t+1} = p(z|x, \theta^t)$
- **M-Step**: $\theta^{t+1} = \arg\max_\theta \sum_n \int_z q^{t+1}(z|x) \log p(x, z|\theta) dz$

where $\theta = (\mu, \Lambda, \Psi)$. Results for both steps:

- **E-Step**
  $q^{t+1} = p(z|x, \theta^t) = \mathcal{N}(z|m^{(i)}, V^{(i)})$ where
  $V^{(i)} = (1 + \Lambda^\intercal \Psi^{-1} \Lambda)^{-1}$ and $m^{(i)} = V^{(i)} \Lambda^\intercal \Psi^{-1} (x - \mu)$.
- **M-Step** $\Lambda^{t+1} = \left(\sum_i x^{(i)} m^{(i)\intercal} \right) \left(\sum_i V^{(i)} \right)^{-1}$
  $\Psi^{t+1} = \frac{1}{n} \mathrm{diag} \left[\sum_i x^{(i)} x^{(i)\intercal} + \Lambda^{t+1} \sum_i m^{(i)} x^{(i)\intercal} \right]$

Principal Component Analysis

Probabilistic Principal Component Analysis (PPCA)

In Factor Analysis, we can write the marginal density as:

$$x \sim \mathcal{N}(\mu, \Lambda\Lambda^\intercal + \Psi)$$

where we assumed that Ψ was a diagonal matrix.

Now we make the further restriction that $\Psi = \sigma^2$ i.e.:

$$\begin{aligned} z &\sim \mathcal{N}(0, 1) \\ x|z &\sim \mathcal{N}(\mu + \Lambda z, \sigma^2 \mathbb{1}) \end{aligned}$$

where again μ is the mean vector, σ^2 the global sensor noise and Λ are the *principal components*.

Likelihood

For both FA and PCA, the data model is Gaussian:

$$
\begin{aligned}
\mathcal{L}(\theta, \mathcal{D}) &= -\frac{N}{2}\log\left|\Lambda\Lambda^\intercal + \Psi\right| - \frac{1}{2}\sum_n (x^n - \mu)^\intercal (\Lambda\Lambda^\intercal + \Psi)^{-1}(x^{(n)} - \mu) \\
&=: -\frac{N}{2}\log|V| - \frac{1}{2}\mathrm{trace}\left[V^{-1}\sum_n (x^{(n)} - \mu)(x^{(n)} - \mu)^\intercal\right] \\
&=:: -\frac{N}{2}\log|V| - \frac{1}{2}\mathrm{trace}\left[V^{-1}S\right]
\end{aligned}
$$

where V is the model covariance and S is the sample covariance.

EM for PCA

Recall from FA and setting $\Psi = \sigma^2 \mathbb{1}$:

- **E-Step**: $q^{t+1} = p(z|x, \theta^t)$
- **M-Step**: $\theta^{t+1} = \arg\max_\theta \sum_n \int_z q^{t+1}(z|x) \log p(x, z|\theta) dz$

where $\theta = (\mu, \Lambda, \sigma)$. Results for both steps:

- **E-Step**
  $q^{t+1} = p(z|x, \theta^t) = \mathcal{N}(z|m^{(i)}, V^{(i)})$ where
  $V^{(i)} = (1 + \sigma^{-2}\Lambda^\intercal\Lambda)^{-1}$ and $m^{(i)} = \sigma^{-2} V^{(i)} \Lambda^\intercal (x - \mu)$.
- **M-Step** $\Lambda^{t+1} = \left(\sum_i x^{(i)} m^{(i)\intercal}\right) \left(\sum_i V^{(i)}\right)^{-1}$
  ${\sigma^2}^{t+1} = \frac{1}{n} \mathrm{diag}\left[\sum_i x^{(i)} x^{(i)\intercal} + \Lambda^{t+1} \sum_i m^{(i)} x^{(i)\intercal}\right]$

Principal Component Analysis - Zero noise limit

- For $\sigma^2 \to 0$ we obtain the "classic" PCA.
- The maximum likelihood parameters are the same, the only difference is the sensor noise σ^2.
- In the "classic" setting, inference is easier since it corresponds to orthogonal projection:

$$\lim_{\sigma^2 \to 0} \Lambda^\intercal(\Lambda\Lambda^\intercal + \sigma^2\mathbb{1})^{-1} = \Lambda^\intercal(\Lambda\Lambda^\intercal)^{-1} \tag{1}$$

- Data compression:

$$\mu_{z|x} = \Lambda^\dagger(x - \mu) \tag{2}$$

where $\Lambda^\dagger$ is the pseudo-inverse.

Interpretation of Differences

PPCA

- PCA looks for directions of large variance i.e. will identifiy large noise directions
- For PCA the rotation is unimportant.

FA

- FA looks for directions of large correlation in data!
- Since Λ only appears in outer product $\Lambda\Lambda^\top$, the rotation of data is important!
- Scale of data is unimportant.

Latent Covariance

So far $z \sim \mathcal{N}(0, 1)$, now:

$$z \sim \mathcal{N}(0, P)$$
$$x|z \sim \mathcal{N}(\mu + \Lambda z, \Psi)$$

The marginal probability is

$$x \sim \mathcal{N}(\mu, \Lambda P \Lambda^\intercal + \Psi)$$

Decomposing $P = EDE^\intercal$ and setting $\Lambda = \Lambda ED^{\frac{1}{2}}$ leads to another identifiability issue between Λ and P. Thus:

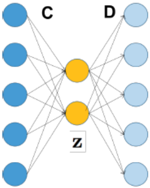
- Set covariance P equal to identity (FA)
- Force columns of Λ to be orthonormal (PCA)

Linear Autoencoder

Again: Given data points $x_i \in \mathbb{R}^n$, $i = 1, \cdots, N$

Goal: Find lower m-dimensional representation , $m < n$ by minimizing the reconstruction error $\sum_i^N ||x_i - DCx_i||^2$, where:

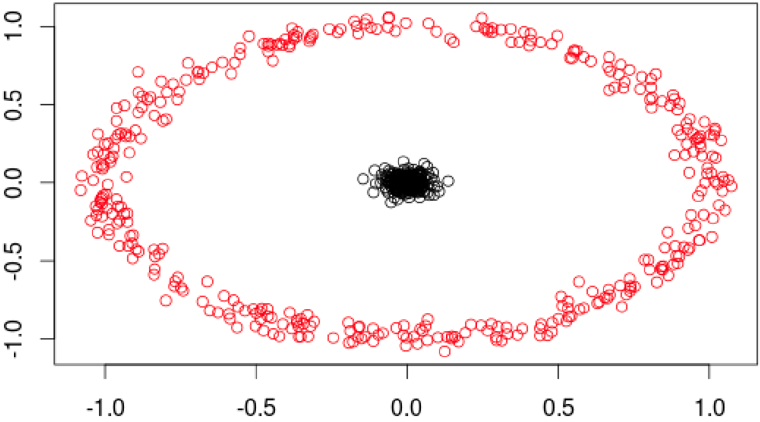
$$x \xrightarrow{C} z \xrightarrow{D} \widehat{x}$$



Problem: Find optimal C and D.

Solution: PCA!

Problem



Connection with State Space Models

Connection with State Space Models

State space models are dynamical generalizations of FA model.

$$x_t = Ax_{t-1} + Gw_t$$

whee $w_t = \mathcal{N}(0, Q)$

- at each point in time t, we use a FA model to represent the output
- State Space models are just sequential Factor Models
- C is the *loading* matrix, shared across all (x_t, y_t) pairs.
- We assume all data points lie in the same low-dimensional space.

Summary - so far

- Factor analysis implies latent variable is assumed to lie on low-dimensional linear subspace
- Similar to mixture model, now just continuous
- Dimensionality reduction technique
- PCA, ICA, sensible PCA, linear autoencoder can all be combined in one framework -> reference unifying review.
- For non-linear extensions, we use variational inference.

Application to Images - Eigenfaces

Motivation Represent face images efficiently and capture relevant information while removing nuisance factors like lighting conditions, facial expression, occlusion etc.

Idea

- Given training set of N images, use PCA to form a basis of K images, K«N.
- PCA for dimensionality reduction: Eigenface = eigenvector of covariance function
- Use lower dimensional features e.g. for face classification

Literature

Sirovich and Kirby, Low-dimensional procedure for the characterization of human face, 1987

Turk and Pentland, Eigenfaces for Recognition, Journal of Cognitive Neuroscience, 1991

Turk and Pentland, Face Recognition using Eigenfaces, CVPR 1991

Training*

*image from Turk and Pentland, Eigenfaces for Recognition, Journal of Cognitive Neuroscience, 1991

Eigenface*

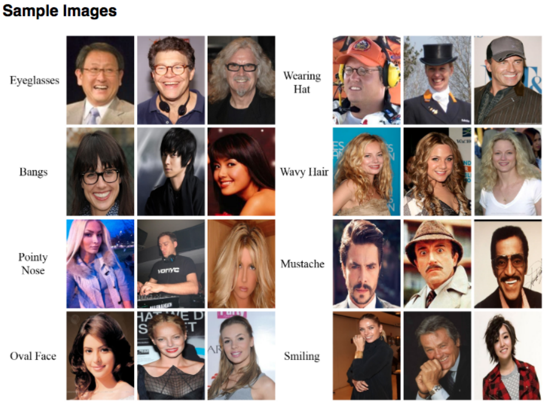
*image from Turk and Pentland, Eigenfaces for Recognition, Journal of Cognitive Neuroscience, 1991

Reconstruction*

*image from Turk and Pentland, Eigenfaces for Recognition, Journal of Cognitive Neuroscience, 1991

Exercise

Coding Exercise: Eigenfaces using CelebA dataset (> 200K celebrity images).



Report and discuss (mean, speed, rotation, scaling, etc.) using piazza.

Next week

So far:

- Latent variable models
- Maximum Likelihood Estimation to find parameters
- Variational Inference for non-tractable models

Alternative: Implicit Models, which do not require a tractable likelihood function.

Plan for next week:
Guest Lecture: Olivier Bachem - Generative Adversarial Networks

Questions?