This Lecture: Advanced Machine Learning

- Regression Approaches
  - Linear Regression
  - Regularization (Ridge, Lasso)
  - Kernels (Kernel Ridge Regression)
  - Gaussian Processes
- Bayesian Estimation & Bayesian Non-Parametrics
  - Prob. Distributions, Approx. Inference
  - Mixture Models & EM
  - Dirichlet Processes
  - Latent Factor Models
  - Beta Processes
- SVMs and Structured Output Learning
  - SV Regression, SVDD
  - Large-margin Learning

Topics of This Lecture

- Latent Factor Models
  - Motivation
  - Example: PCA
  - Applications of PCA
  - Probabilistic PCA
  - Maximum Likelihood for PCA
  - Other Latent Factor Models: FA, ICA
- Towards Infinite Latent Factor Models
  - General formulation
  - Sparse latent factor models
  - Priors on binary matrices
  - Finite latent feature model

Mixture Models vs. Latent Factor Models

- Mixture Models
  - Assume that each observation was generated by exactly one of \( K \) components.
  - The uncertainty is just about which component is responsible.
- Latent Factor Models
  - Weaken this assumption.
  - Each observation is influenced by each of \( K \) components (factors or features) in a different way.
  - Sparse factor models: only a small subset of factors is active for each observation.

Latent Factor/Feature Models

- Most popular examples
  - Principal Component Analysis (PCA)
  - Factor Analysis (FA)
  - Independent Component Analysis (ICA)
- Properties
  - All of those assume that the number of factors \( K \) is known.
  - Usually, \( K \) is smaller than the dimensionality of the data:
    \( K < D \)
  - Models provide dimensionality reduction.
- Let’s look at PCA and see how it fits into this framework...

Principal Component Analysis

- Goal
  - Given a data set \( X = \{ x_n \} \) in \( D \) dimensions, find the \( K \)-dimensional projection \( (K < D) \) that maximizes the variance of the projected data.
  - Intuition: preserve as much variance as possible.
- One-dimensional example
  - Project each data point \( x_n \) onto the unit vector \( u_1 \):
    \[ y_n = u_1^T x_n \]
  - What is the vector \( u_1 \) that maximizes the variance of the projected data?
Principal Component Analysis

• One-dimensional example (cont’d)
  • Mean of the projected data
    \[ \bar{y} = u_1^T \bar{x} \quad \bar{x} = \frac{1}{N} \sum_{n=1}^{N} x_n \]
  • Variance of the projected data
    \[ \frac{1}{N} \sum_{n=1}^{N} (u_1^T x_n - u_1^T \bar{x})^2 = u_1^T S u_1 \]
    \[ S = \frac{1}{N} \sum_{n=1}^{N} (x_n - \bar{x})(x_n - \bar{x})^T \]
  where \( S \) is the data covariance matrix.

Principal Component Analysis

• Optimization problem
  • Maximize the projected variance \( u_1^T S u_1 \) w.r.t. \( u_1 \).
  • Problem: trivial solution is \( u_1 = k \).
  \[ \Rightarrow \text{Need to enforce the normalization condition } u_1^T u_1 = 1. \]
  • Formulation with Lagrange multiplier
    \[ \arg \max_{u_1} u_1^T S u_1 + \lambda_1 (1 - u_1^T u_1) \]
    \[ \Rightarrow \text{Eigenvalue problem: } u_1 \text{ must be eigenvector of } S. \]
    \[ u_1^T S u_1 = \lambda_1 u_1 \]
    \[ \Rightarrow \text{Maximal variance if } \lambda_1 \text{ is the largest eigenvalue of } S. \]

Principal Component Analysis

• General case
  • Inductively, we can show that the optimal linear projection into a \( K \)-dimensional space is given by the first \( K \) eigenvectors \( u_1, \ldots, u_K \) of \( S \).
    \[ y_n = U_{1..K} x_n \]
  • Graphical interpretation

Uses of PCA

• Dimensionality reduction
  • Work in a subspace that contains only the \( K \) most important dimensions.
  • Advantages: faster processing, reduced memory footprint, robustness to noise.

Example: Eigenfaces

[Turk & Pentland, 1993]

Uses of PCA

• Dimensionality reduction
  • Work in a subspace that contains only the \( K \) most important dimensions.
  • Advantages: faster processing, reduced memory footprint, robustness to noise.

• Data Preprocessing
  • Remove correlations between different dimensions of the data and bring them to a common scale.
  • Many classification or regression algorithms work better when the data is standardized, i.e., when each variable has zero mean and unit variance.
  • Using PCA, we can make a more substantial normalization of the data to give it zero mean and unit covariance. This is known as whitening.
PCA for Whitening

- **Whitening procedure**
  - Rewrite the eigenvector equation in matrix form
    
    \[ SU_1 = \lambda_1 U_1 \Rightarrow SU = UL \]
    
    where \( L = \text{diag}(\lambda_i), \) \( U = [u_1, \ldots, u_p] \).
  - Define for each data point the transformed value as
    \[ y_n = L^{-1/2}U^T(x_n - \bar{x}) \]
    
    \( \Rightarrow \) The transformed set \( \{y_n\} \) has zero mean and unit covariance.

- **Whitening result**
  - Correlations are removed.
  - Distances are normalized to same value range.

Probabilistic PCA

- **Discussion**
  - The formulation of PCA we have just seen was based on a linear projection of data into a lower-dim. subspace.
  - We now show that PCA can also be expressed as the ML solution of a probabilistic latent variable model.

- **Advantages of Probabilistic PCA**
  - We can derive an EM algorithm that is efficient in situations where only few leading eigenvectors are required.
  - Probabilistic model + EM makes it possible to deal with missing data values.
  - Basis for a Bayesian treatment of PCA in which the dimensionality of the principal subspace can be found automatically.

Probabilistic PCA

- **Graphical Model**
  - Introduce an explicit latent variable \( z \) corresponding to the principal component subspace.
  - Define a Gaussian prior distribution
    \[ p(z) = \mathcal{N}(z; 0, I) \]
  - Conditional distribution also Gaussian
    \[ p(x|z) = \mathcal{N}(x|Wz + \mu, \sigma^2I) \]
    
    \( \Rightarrow \) Example of a Linear Gaussian framework: all of the marginal and conditional distributions are Gaussian.
  - As we will see, the columns of \( W \) span an \( K \)-dimensional linear subspace within the data space that corresponds to the principal subspace.

Topics of This Lecture

- **Latent Factor Models**
  - Motivation
  - Example: PCA
  - Applications of PCA
  - Probabilistic PCA
  - Maximum Likelihood for PCA
  - Other Latent Factor Models: FA, ICA

- **Towards Infinite Latent Factor Models**
  - General formulation
  - Sparse latent factor models
  - Priors on binary matrices
  - Finite latent feature model

Probabilistic PCA

- **Generative interpretation**
  - \( D \)-Dimensional observed variable \( x \) is defined by a linear transformation of the \( K \)-dimensional latent variable \( z \) plus some added (isotropic Gaussian) noise.
    
    \[ x = Wz + \mu + \epsilon \]
  - Marginal distribution
    \[ p(x) = \int p(x|z)p(z)dz \]
  - Because of the linear-Gaussian model, this will again be Gaussian
    \[ p(x) = \mathcal{N}(x|\mu, C) \]
    
    where
    \[ C = WW^T + \sigma^2I \]
Probabilistic PCA

- Properties
  - There is a rotational ambiguity in the parametrization.
  - Consider a rotation of the latent parameter space with orthonormal matrix $R$ (orthogonality property: $RR^T = I$).

  $$W = WR$$

  $$WW^T = WRR^TW = WW^T$$

  Thus, the covariance matrix $C$ is independent of $R$.

  - Efficiency trick: instead of evaluating $C^{-1}$ directly, use the following equivalence ($\mathcal{O}(D^3) \rightarrow \mathcal{O}(K^3)$).

  $$C^{-1} = \sigma^2I - \sigma^{-2}MW^{-1}W^T$$

  with

  $$M = W^TW + \sigma^2I$$

Posterior distribution

- Can again be derived from properties of linear Gaussian models

$$p(z|x) = \mathcal{N}(z|\mu^{-1}W^T(x - \mu), \sigma^2\mu^{-1})$$

Topics of This Lecture

- Latent Factor Models
  - Motivation
  - Example: PCA
  - Applications of PCA
  - Probabilistic PCA
  - Maximum Likelihood for PCA
  - Other Latent Factor Models: FA, ICA

- Towards Infinite Latent Factor Models
  - General formulation
  - Sparse latent factor models
  - Priors on binary matrices
  - Finite latent feature model

Maximum Likelihood for PCA

- Maximum Likelihood estimate
  - Log-likelihood function

$$\log p(X|\mu, \sigma^2) = \sum_{n=1}^{N} \log p(x_n|\mu, \sigma^2)$$

$$= -\frac{1}{2} \sum_{n=1}^{N} (x_n - \mu)^T C^{-1} (x_n - \mu)$$

$$- \frac{ND}{2} \log(2\pi) - \frac{N}{2} \log |C|$$

- Optimizing the parameters

$$\frac{\partial}{\partial \mu} p(X|\mu, \sigma^2) \perp 0 \Rightarrow \mu = \bar{x}$$

- Maximum Likelihood estimate (cont’d)
  - Maximizing w.r.t. $\sigma^2$

$$\sigma^2_{ML} = \frac{1}{D-K} \sum_{i=K+1}^{D} \lambda_i$$

$$\Rightarrow \sigma^2_{ML}$$ is the average variance associated with the discarded dimensions.
Interpretation of Probabilistic PCA

• Putting all those results together...
  - Consider again the covariance matrix
    \[ C = WW^T + \sigma^2 I \]
    where
    \[ W_{ML} = U_K(L_K - \sigma^2 I)^{1/2} R \]
    \[ \sigma^2_{ML} = \frac{1}{D-K} \sum_{i=K+1}^{D} \lambda_i \]
  - The model correctly captures the variance of the data along the principal axes and approximates the variance in all remaining directions by \( \sigma^2 \), the average of the discarded eigenvalues.
  - To construct \( C \), we simply set \( R = I \) and compute the principal eigenvalues and eigenvectors of the data covariance matrix \( S \).
  - If \( C \) is obtained in a different way, \( R \) may still be arbitrary.

Discussion: PCA vs. Probabilistic PCA

• Comparison with standard PCA:
  - PCA is generally formulated as a projection of points from the \( D \)-dimensional space onto a \( K \)-dimensional linear subspace.
  - Probabilistic PCA is more naturally expressed as a mapping from the latent space into the data space via
    \[ x = Wz + \mu + \epsilon \]
    - For applications such as visualization or data compression, we can reverse this mapping using Bayes’ theorem.
    \[ E[z|x] = M^{-1}W_{ML}(x - \bar{x}) \]
    - If \( \sigma \to 0 \), this reduces to the standard PCA model.

Topics of This Lecture

• Latent Factor Models
  - Motivation
  - Example: PCA
  - Applications of PCA
  - Probabilistic PCA
  - Maximum Likelihood for PCA
  - Other Latent Factor Models: FA, ICA

Other Latent Factor Models

• Factor Analysis (FA)
  - Linear-Gaussian latent variable model, closely related to Probabilistic PCA.
  - Probabilistic PCA uses an isotropic covariance \( p(x|z) = N(x|Wz + \mu, \sigma^2 I) \)
  - Factor Analysis instead assumes a diagonal covariance \( p(x|z) = N(x|Wz + \mu, \Psi) \)
  - The FA model explains the observed covariance structure of the data by representing the independent variables associated with each coordinate by the matrix \( \Psi \) and capturing the covariance between variables in the matrix \( W \).
  - In the literature, the columns of \( W \) are called factor loadings, the diagonal elements \( \psi_i \) are called uniquenesses.

Other Latent Factor Models (2)

• Independent Component Analysis (ICA)
  - Model for which the observed variables are related linearly to the latent variables, but for which the latent distribution is non-Gaussian.
  - Consider a distribution over latent variables that factorizes
    \[ p(z) = \prod_{i=1}^{K} p(z_i) \]
    i.e., the components \( z_i \) are independent.
  - This definition requires that the latent variables have a non-Gaussian distribution (as Gaussian models always have the rotational ambiguity \( I \) in latent space).
  - There is a large variety of ICA models and corresponding algorithms, differing mainly in the choice of latent-variable distribution.

Next Steps from Here...

• Discussion
  - We have now derived that the PCA result can be obtained as the ML estimate of the corresponding probabilistic model.
  - This result can directly be used to incorporate priors and derive a Bayesian extension of the model.
  - We can do similar things for FA and ICA...
  - In the following, we will go into a different direction
    - What happens when we let \( K \to \infty \)?
    - Can we automatically determine \( K \)?
In order to keep inference tractable, however, we have to restrict the model somehow.

Each group of observations is associated with a subset of the factors. Back there, we also had binary matrices due to 1-of-$K$ coding.

Latent Factor Models

- General formulation
  - Assume that the data are generated by noisy weighted combination of latent factors
    \[ x_n = F y_k + \epsilon \]
  - E.g., in Factor Analysis, $F$ would be a $D \times K$ factor loading matrix expressing how latent factor $k$ influences observation dimension $d$. $y_k$ would be a $K$-dimensional vector expressing the activity of each factor.

- Advantages of latent feature modeling
  - Each group of observations is associated with a subset of the possible latent features/factors.
  - Factorial power: There are $2^K$ combinations of $K$ features, while accurate mixture modeling may require many more clusters.

Towards Infinite Latent Factor Models

- General formulation
  - Incorporate sparsity
    - Decompose $F$ into the product of two components: $F = Z \otimes W$, where $\otimes$ is the Hadamard product (element-wise product).
    - $z_{nk}$ is a binary mask variable indicating whether factor $k$ is “on”.
    - $w_{nk}$ is a continuous weight variable.
  - Enforce sparsity by restricting the non-zero entries in $Z$.

Sparse Latent Factor Models

- Goal: Infinite models
  - We would like to work with infinite-dimensional models ($K \to \infty$).
  - In order to do keep inference tractable, however, we have to restrict the model somehow.

- Mixture Models: DPs enforce that the main part of the probability mass is concentrated on few cluster components.
- Latent Factor Models: enforce that each object is represented by a sparse subset of an unbounded number of features.

- Incorporating sparsity
  - Decompose $F$ into the product of two components: $F = Z \otimes W$, where $\otimes$ is the Hadamard product (element-wise product).
  - $z_{nk}$ is a binary mask variable indicating whether factor $k$ is “on”.
  - $w_{nk}$ is a continuous weight variable.
  - Enforce sparsity by restricting the non-zero entries in $Z$.

Towards a Full Bayesian Treatment

- Inference in Latent Feature Models
  - Goal: Infer the latent factors, mask variables, and weights.
  - Classical approaches (PCA, FA, ICA) fit point estimates of the parameters through ML estimation.

- Bayesian approach
  - Specify a prior over latent features/factors $p(F)$ and a distribution over observed property distributions $p(X|F)$.
  - Compute the posterior $p(F, Z, W|X)$.
  - Our focus will be on $p(F) = p(Z)p(W)$, showing how such a prior can be defined without placing an upper bound on the number of features/factors.

Priors on Binary Matrices

- Let’s first go back to DPs/CRPs
  - Back there, we also had binary matrices due to 1-of-$K$ coding.
  - What is different here?

- Binary matrices for clustering
  - We can think of CRPs as priors on infinite binary matrices, where...
    - ...each data point is assigned to one and only one cluster (class).
    - ...the rows sum to one.
Priors on Binary Matrices

- Let’s first go back to DPs/CRPs
  - Back there, we also had binary matrices due to 1-of-K coding.
  - What is different here?

- More general binary matrices
  - Each data point can now have multiple factors/features.
  - The rows sum to more than one.
  - What is the corresponding prior on infinite binary matrices?

Priors on Latent Factor Models

- Defining suitable priors
  - We will focus on defining a prior on \( Z \), since the effective dimensionality of the latent feature model is determined by \( Z \).
  - Assuming that \( Z \) is sparse, we can define a prior for infinite latent feature models by defining a distribution over infinite binary matrices.

- Desiderata for such a distribution
  - Objects should be exchangeable.
  - Inference should be tractable.

- Procedure
  - Start with a model that assumes a finite number of features and consider the limit as this number approaches infinity.
  - Next lecture...

References and Further Reading

- More information on latent factor models and particularly PCA can be found in Chapter 12 of

  Christopher M. Bishop
  *Pattern Recognition and Machine Learning*
  Springer, 2006

- Tutorial papers for infinite latent factor models
  - A good introduction to the topic