Dimensionality reduction

Feature selection

Milos Hauskrecht
milos@cs.pitt.edu
5329 Sennott Square

Dimensionality reduction

• Is there a lower dimensional representation of the data that captures well its characteristics?

• Assume:
  – We have an data set \{x_1, x_2, \ldots, x_N\} such that
    \[ x_i = (x_{i1}, x_{i2}, \ldots, x_{id}) \]
  – Assume the dimension \(d\) of the data point \(x\) is very large
  – We want to analyze \(x\)

• Methods of analysis are sensitive to the dimensionality \(d\)

• Our goal:
  – Find a lower dimensional representation of data of dimension \(d' < d\)
Principal component analysis (PCA)

- **Objective**: We want to replace a high dimensional input with a small set of features (obtained by combining inputs)
  - Different from the feature subset selection !!!
- **PCA**:
  - A linear transformation of \( d \) dimensional input \( x \) to \( M \) dimensional feature vector \( z \) such that \( M < d \) under which the retained variance is maximal.
  - Equivalently it is the linear projection for which the sum of squares reconstruction cost is minimized.
Principal component analysis (PCA)

• **PCA:**
  – linear transformation of a $d$ dimensional input $x$ to $M$ dimensional vector $z$ such that $M < d$ under which the retained variance is maximal.
  – Task independent

• **Fact:**
  – A vector $x$ can be represented using a set of orthonormal vectors $u$
    $$x = \sum_{i=1}^{d} z_i u_i$$
  – Leads to transformation of coordinates (from $x$ to $z$ using $u$’s)
    $$z_i = u_i^T x$$
PCA

- **Idea:** replace \( d \) coordinates with \( M \) of \( z_i \) coordinates to represent \( x \). We want to find the subset \( M \) of basis vectors.

\[
\tilde{x} = \sum_{i=1}^{M} z_i u_i + \sum_{i=M+1}^{d} b_i u_i
\]

\( b_i \) - constant and fixed

- **How to choose the best set of basis vectors?**
  - We want the subset that gives the best approximation of data \( x \) in the dataset on average (we use least squares fit)
  
  Error for data entry \( \mathbf{x}^n \quad \mathbf{x}^n - \tilde{\mathbf{x}}^n = \sum_{i=M+1}^{d} (z_i^n - b_i) \mathbf{u}_i \)

  Reconstruction error
  
  \[
  E_M = \frac{1}{2} \sum_{n=1}^{N} \| \mathbf{x}^n - \tilde{\mathbf{x}}^n \| = \frac{1}{2} \sum_{n=1}^{N} \sum_{i=M+1}^{d} (z_i^n - b_i)^2
  \]

  Differentiate the error function with regard to all \( b_i \) and set equal to 0 we get:

  \[
b_i = \frac{1}{N} \sum_{n=1}^{N} z_i^n = (\mathbf{u}_i^T \tilde{\mathbf{x}}) \quad \quad \quad \tilde{\mathbf{x}} = \frac{1}{N} \sum_{n=1}^{N} \mathbf{x}^n
  \]

  Then we can rewrite:

  \[
  E_M = \frac{1}{2} \sum_{i=M+1}^{d} \mathbf{u}_i^T \Sigma \mathbf{u}_i \quad \Sigma = \sum_{n=1}^{N} (\mathbf{x}^n - \tilde{\mathbf{x}})(\mathbf{x}^n - \tilde{\mathbf{x}})^T
  \]

  The error function is optimized when basis vectors satisfy:

  \[
  \Sigma \mathbf{u}_i = \lambda_i \mathbf{u}_i \quad \quad \quad E_M = \frac{1}{2} \sum_{i=M+1}^{d} \lambda_i
  \]

  **The best \( M \) basis vectors:** discard vectors with \( d-M \) smallest eigenvalues (or keep vectors with \( M \) largest eigenvalues)

  Eigenvector \( \mathbf{u}_i \) - is called a **principal component**
PCA

- Once eigenvectors $u_i$ with largest eigenvalues are identified, they are used to transform the original $d$-dimensional data to $M$ dimensions

- To find the “true” dimensionality of the data $d’$ we can just look at eigenvalues that contribute the most (small eigenvalues are disregarded)

- **Problem:** PCA is a linear method. The “true” dimensionality can be overestimated. There can be non-linear correlations.

- ** Modifications for nonlinearities:** kernel PCA

---

Dimensionality reduction with neural nets

- **PCA** is limited to linear dimensionality reduction
- To do non-linear reductions we can use neural nets
- **Auto-associative (or auto-encoder) network:** a neural network with the same inputs and outputs ($x$)

- The middle layer corresponds to the reduced dimensions
Dimensionality reduction with neural nets

- **Error criterion:**
  \[ E = \frac{1}{2} \sum_{n=1}^{N} \sum_{i=1}^{d} (y_{i}^{n} - x_{i}^{n})^2 \]

- Error measure tries to recover the original data through limited number of dimensions in the middle layer
- **Non-linearities** modeled through intermediate layers between the middle layer and input/output
- If no intermediate layers are used the model replicates PCA optimization through learning

Multidimensional scaling

- Find a lower dimensional space projection such that the distances among data points are preserved
- Used in visualization – d-dimentional data transformed to 3D or 2D
- **Dissimilarities before projection** \( \delta_{i,j} = \| x_i - x_j \| \)
- **Objective:** Optimize points and their coordinates by fitting the dissimilarities afterwards

\[
\min_{(x_1, x_2, \ldots, x_n)} \sum_{i<j} (\| x'_i - x'_j \| - \delta_{ij})^2
\]
Latent variable models

Latent variables (s): Dimensionality k

Observed variables x: real valued vars
Dimensionality d

Cooperative vector quantizer

Model:
Latent var s_i:
~ Bernoulli distribution parameter: \( \pi_i \)
\[ P(s_i | \pi_i) = \pi_i^{s_i} (1 - \pi_i)^{1-s_i} \]

Observable variables x:
~ Normal distribution parameters: W, \( \Sigma \)
\[ P(x | s) = N(Ws, \Sigma) \]
We assume \( \Sigma = \sigma I \)

Joint for one instance of x and s:
\[ P(x,s | \Theta) = \left(2\pi\right)^{-d/2} \sigma^{-d/2} \exp\left\{-\frac{1}{2\sigma^2} (x - Ws)^T (x - Ws)\right\} \prod_{i=1}^{k} \pi_i^{s_i} (1 - \pi_i)^{1-s_i} \]
Other unsupervised methods

- **Factor analysis (a latent variable model)**
  - Decompose signal into multiple Gaussian sources

  \[ x = As \quad X \text{ is a linear combination of values for sources} \]
  \[ s = Wx = A^{-1}x \]

- **Independent component analysis:**
  - Identify independent components/signals/sources in the original data
  - Non-Gaussian signals

  \[ x = As \]