Overview: Distance Matrix Learning, Independent Component Analysis (Blind Source Separation), Matrix Factorization and Manifold Embedding

1 Review for Last Lecture

Linear Dimensionality Reduction:
1. RP
2. PCA
3. LDA (supervised technique)
   “maximizing” the distance between class means
   “minimizing” the inter-cluster variance
4. MDS
   Given: \( \text{dist}(O_i, O_j) = \delta_{ij}, x_i, x_j \in \mathbb{R}^D \) s.t. \( ||x_i - x_j||_2 \equiv \delta_{ij} \)
   Goal: \( \min_S(x_1, ..., x_n) = \sum_{i<j} (D_{ij} - \delta_{ij})^2 \)
   Question: If new data comes, do we need to do the optimization again or there is a simple way?
   Answer: This is a question related to “out of simple” extension.

2 Distance Metric Learning

Given: \( x_1, ..., x_n \in \mathbb{R}^D \) \( \rho(x_i, x_j) = ||x_i - x_j||_2 = \left[ \sum_{d=1}^{D} (x_{id} - x_{jd})^2 \right]^{1/2} = \left[ (x_i - x_j)^T I (x_i - x_j) \right]^{1/2} \)
Output: Best Matrix \( L \in \mathbb{R}^{K \times D} \) for representing the data (improve the classification)
One observation:
\[
\rho_L(x_i, x_j) = ||Lx_i - Lx_j||_2 = \left[ (x_i - x_j)^T L^T L (x_i - x_j) \right]^{1/2}
\]
Define \( M = L^T L \)
“Supervision”: \( x_1, ..., x_n \in \mathbb{R}^D; y_1, ..., y_n \in \{0, 1\} \)
Ideas:
1. “similar set” \( S = \{(x_i, x_j)\} \) s.t. \( y_i = y_j \) 2. “different set” \( D = \{(x_i, x_j)\} \) s.t. \( y_i \neq y_j \)
Professor came up an objective function:
\[
\min \Psi(M) = \sum_{(x_i, x_j) \in S} \frac{\rho^2_M(x_i, x_j)}{|S|} - \lambda \sum_{(x_i, x_j) \in D} \frac{\rho^2_M(x_i, x_j)}{|D|}
\]
The first term can be called “pull term”, the second “push term”, $\lambda$ is a hyper-parameter. The classic approach is:

$$\max \sum_{(x_i, x_j) \in D} \rho_M^2(x_i, x_j)$$

s.t.

$$\sum_{(x_i, x_j) \in S} \rho_M^2(x_i, x_j) \leq 1$$

$$M \geq 0 \quad [M \in PSD]$$

$$\text{rank}(M) \leq k \quad (\text{"non-convex"})$$

Note1: $M \geq 0$ is “conic constrain”, it can be solved by “semi-definite program”, the basic idea is pick up negative eigenvalue and make it to be 0. Figure 1 shows some basic idea about how to deal with it.

![Figure 1](image)

Note2: Rank constraints are $L_0 - type$ and it is non-convex, the nearest convex constraints are $L_1 - type$ i.e. trace constraints($tr(M)$). Therefore, you can replace $\text{rank}(M) \leq k$ by $tr(M) \leq k$. However, if rank of $L$ is critical, you have to work with $\text{rank}(L)$, making this a $QP_2$ problem.

3 Independent Component Analysis

Idea: “Maximize the non-gaussian of each dimension”

Example: Try to separate the conversation in a cocktail party using microphone.

Define D: number of microphone; K: number of conversation; T: sound dimension

Let $X = M \times S$, where $X \in R^{D \times T}$ is what you get from all the microphones, $M \in R^{D \times K}$ is the
conversation gained by the microphone. \( S \in \mathbb{R}^{K \times T} \) is sound signal from K conversations.

Assumption: The assumption is based on CLT, i.e., linear combination of independent random variables is going to be gaussian like. Therefore, \( X \) is more gaussian than \( S \) (\( S \) is independent from each other and \( X \) will be more dependent).

Goal: Find \( WX = S \) which is less gaussian like.

Question: How to measure gaussian like?


### 3.1 Kurtosis Method

Define kurtosis for a distribution \( y \),

\[
\text{kurtosis}(y) := E[y^4] - 3(E[y^2])^2.
\]

Fact: \( g \sim N(0, 1) \quad E(g^4) = 3 \)

\( \text{kurtosis}(y) = 0 \iff \text{gaussian} \)

\( \text{kurtosis}(y) < 0 \iff \text{subgaussian} \)

\( \text{kurtosis}(y) > 0 \iff \text{supgaussian} \)

The objective function:

\[
\max (\text{kurtosis}(WX))^2
\]

s.t.

\[
\text{var}(WX) = 1
\]

Drawback: Not robust to outliers!

### 3.2 Negative Entropy Method

Reminder: Entropy \( H(y) := -\sum_p P[Y = y] \log P[Y = y] = -\int x p \log p \, dx \)

Observation: Guassian distribution has least information, i.e., has most entropy of all distribution with the same variance.

The objective function:

\[
\max -H(WTX)
\]

s.t.

\[
\text{Var}(WTX) = 1
\]

### 3.3 Minimize Mutual Information Method

Goal:

\[
\min \sum_{i<j} I(W_iTX; W_jTX)
\]

### 4 Matrix Factorization

Example: Netflix Problem

Description: Suppose we have \( m \) users and \( n \) movies, each user rates the movies which he has seen. Let \( r_{ij} \) be the rating assigned by user \( i \) to movies \( j \). Since each user can only rate few movies, the matrix would be super-sparse.
Idea: we assume there are $k$ factors which have vital influence on users and movies, these factors maybe include horror, romance, science, etc.
Define $u_i \in \mathbb{R}^k, m_j \in \mathbb{R}^k$, then $U \in \mathbb{R}^{m \times k}, M \in \mathbb{R}^{k \times n}$
Objective function:
$$\min_{U,M} \sum_{r_{ij} \in \text{observed}} (r_{ij} - u_i m_j)^2$$
Another way:
$$\min_{U,M} ||R - UM||^2_F$$

5 Manifold Embedding

Definitions:
1. n-dim manifolds: An object $\subseteq \mathbb{R}^D$ which locally looks like(homeomorphic) $\mathbb{R}^n$
2. Homeomorphic: continual $f$ and $f^{-1} := \text{homeomorphic}$
3. Diffeomorphic: differentiable $f$ and $f^{-1} := \text{diffeomorphic}$

Manifold hypothesis: $X \subseteq \mathbb{R}^D$ measurement are non-linear smoothly related. $X$ is sampled from an underlying(low-dimensional) manifold(perhaps with some noise).

Explain: There are few underlying factors(n independent) which “control” your observations and you make $D \gg n$ different measurement s.t. $x_i \in \mathbb{R}^D$.

Figure 2 gives some intuition from $\mathbb{R}^2$ to $\mathbb{R}^3$.

Goal of manifold embedding: find $f^{-1}$ or at least find $f^{-1}(x_i) \forall x_i \in X$

Figure 3 gives some intuition from $\mathbb{R}^2$ to $\mathbb{R}^1$. 
Approach: Isometric mapping
1. Create K-NN graph to approximate geodesic distance.

\[ \rho(x_i, x_j) = \text{geo}(x_i, x_j) \]

2. Run MDS on the geodesic distance.

\[ \min S(y_1, \ldots, y_n) = \sum_{i<j} (D(y_i, y_j) - \delta_{ij})^2 \]

Note: Other approaches such as t-SNE, LLE, Max var unfolding will be discussed in the next few lectures.