MATH466/MATH766
Math of machine learning
02/26 Lecture 14 Clustering

References:

•

Todays contents:

- k means
- Review of linear algebra

Important concepts:

- nearest neighbor

Recommend reading:

•

Unsupervised Learning

data : $\{x_i\}_{i=1}^n$ unlabeled

common tasks : ① clustering – separate objects into groups

(≠ classification)

② dimension reduction, feature extraction

representation learning, manifold learning

(can be used for pre-processing of supervised learning and other tasks)

③ generative model

common challenges :

intrinsic (high dimension data, high volume of data) computation

scarcity of data

Plan for this module

clustering : K-means

spectral clustering ← closely related

Expectation Maximization (if time permits)

dim reduction & manifold learning :

Principal Component Analysis (PCA)

Multidimensional Scaling (MDS)

graph → Isomap

spectral embedding (Laplace eigenmap) ← closely related

↓ Page Rank

diffusion map (HW)

t-distribution Stochastic Neighbor Embedding (t-SNE)

Clustering

Data: $\{x_i\}_{i=1}^n$

Goal: find an assignment map π s.t.

(hard clustering)

(soft membership)

K-means (hard clustering)

Find k cluster centers $\{\mu_1, \cdots, \mu_k\}$

and the assignment map π

s.t. $\min_{\pi, \mu} \sum_{i=1}^{n} \| x_i - \mu_{\pi(x_i)} \|^2$

- Generally NP-hard
- A heuristic algorithm: Lloyd's Algorithm (1957)

> randomly initialize μ, loop until μ converges
>
> | Fix center μ, update π
> |
> | Fix assignment π, update μ

The algorithm usually converges

But ① different initialization → different converged solutions

② difficult case → easier to get different results

- Strategies

  ① run the algorithm multiple times and pick the best solution

  ② k-mean ++ initialization
  (spread out the k initial cluster centers)

  choose μ_1 uniformly randomly from $\{x_i\}_{i=1}^{n}$

  for $l = 2, \cdots, k$

  for x_i not chosen, compute $D(x_i)$: the dist between x_i and its nearest center

  choose μ_l randomly from remaining x_i with a weight proportional to $D^2(x_i)$.

- Demo : difficult case for k-means

  ① unknown value of k ←

  ② unbalanced clusters ←

  ③ anisotropically distributed blobs

  ④ unequal variance

  ⑤ nonconvex clusters

Eigen - Decomposition

1. $A \in \mathbb{R}^{n \times n}$

eigenvalue λ, eigenvector x $\quad Ax = \lambda x$,

$\downarrow$ n linearly independent eigenvectors

diagonalizable $\quad A = P \Lambda P^{-1}$

$$P = [x_1 \; x_2 \; \cdots \; x_n]$$

$$\Lambda = \begin{bmatrix} \lambda_1 & & & \\ & \lambda_2 & & \\ & & \ddots & \\ & & & \lambda_n \end{bmatrix}$$

① Not all matrices are diagonalizable

② The diagonalizing matrix P is not unique

③

---

2. $A \in \mathbb{R}^{n \times n}$, $\quad A = A^T$

(Spectral Thm) A can be factored into $A = Q \Lambda Q^T$

where $Q = [x_1, x_2, \cdots, x_n]$ is orthogonal

① Always diagonalizable

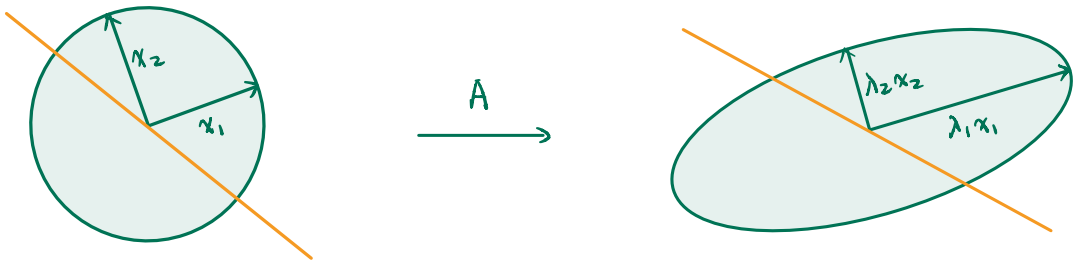
② eig-vals and eig-vecs are real

③

WLOG, assume $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$

Define Rayleigh quotient $R(x) := \dfrac{x^T A x}{x^T x}$

✱ Courant - Fischer - Weyl min-max principle



$$\lambda_1 = \max_x R(x), \qquad \lambda_n = \min_x R(x)$$

$$x_1 = \operatorname{argmax}_x R(x) \qquad x_n = \operatorname{argmin}_x R(x)$$

$$\lambda_j = \max_{S_j} \left[\min_{x \in S_j} R(x) \right] \quad \text{"max"} \quad \text{when} \quad S_j = \operatorname{span}\{x_1, x_2, \cdots, x_j\}$$

$$= \min_{S_{n-j+1}} \left[\max_{x \in S_{n-j+1}} R(x) \right] \quad \text{"min"} \quad \text{when} \quad S_{n-j+1} = \operatorname{span}\{x_j, x_{j+1}, \cdots, x_n\}$$

---

3. $A \in \mathbb{R}^{n \times n}$, $A = A^T$, A p.s.d.

Def. (Frobenius norm) $\|A\|_F^2 = \sum_{i,j=1}^{n} A_{ij}^2$

Low-rank approximation:

$$\min_{\tilde{A}} \|\tilde{A} - A\|_F \quad \text{s.t.} \quad \operatorname{rank}(\tilde{A}) \le j$$

Solution: