Introduction to Machine Learning

Lecture 8 Representation and Clustering - Data Clustering and Typical Methods

Hongteng Xu

中国人民大學
RENMIN UNIVERSITY OF CHINA

高瓴人工智能学院
Gaoling School of Artificial Intelligence

Outline

Review

- Manifold learning (MDS, ISOMAP, LLE, ...)
- Kernel method (Kernel PCA)
- Large-scale manifold learning (t-SNE)
- Autoencoders (briefly)

Outline

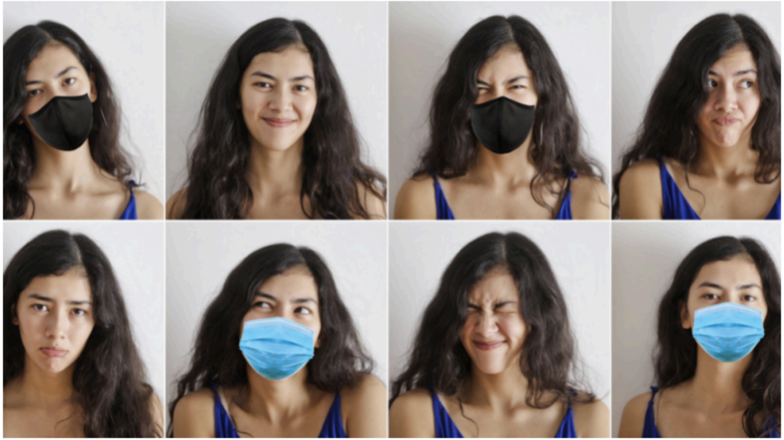
Review

- Manifold learning (MDS, ISOMAP, LLE, ...)
- Kernel method (Kernel PCA)
- Large-scale manifold learning (t-SNE)
- Autoencoders (briefly)

Today

- Data clustering (Motivations and Applications)
- Typical methods: K-means and Spectral Clustering
- Evaluation measurements

Data Clustering: Real-world Examples



Data Clustering: Real-world Examples

Could you enumerate more real-world clustering problems?

K-means: One of The Most Commonly-Used Clustering Methods

Motivation:

- A sample should be closer to the centroid of its cluster than to the centroid of other clusters.

K-means: One of The Most Commonly-Used Clustering Methods

Motivation:

- A sample should be closer to the centroid of its cluster than to the centroid of other clusters.

Principle:

- Find the centroids of the clusters in a heuristic way.

K-means: One of The Most Commonly-Used Clustering Methods

Motivation:

- A sample should be closer to the centroid of its cluster than to the centroid of other clusters.

Principle:

- Find the centroids of the clusters in a heuristic way.

1. Initialize K centroids randomly from the observed data, e.g., $\{\mathbf{c}_k\}_{k=1}^K$.

K-means: One of The Most Commonly-Used Clustering Methods

Motivation:

- A sample should be closer to the centroid of its cluster than to the centroid of other clusters.

Principle:

- Find the centroids of the clusters in a heuristic way.

1. Initialize K centroids randomly from the observed data, e.g., $\{\mathbf{c}_k\}_{k=1}^{K}$.
2. Repeat the following steps till converge
   a. Assign each data to the nearest cluster: $\forall\, \boldsymbol{x}_n$

$$\boldsymbol{x}_n \in \mathcal{C}_k, \quad \text{if } k = \arg\min_{k \in \{1,\dots,K\}} d(\boldsymbol{x}_n, \mathbf{c}_k).$$

K-means: One of The Most Commonly-Used Clustering Methods

Motivation:

- A sample should be closer to the centroid of its cluster than to the centroid of other clusters.

Principle:

- Find the centroids of the clusters in a heuristic way.

1. Initialize K centroids randomly from the observed data, e.g., $\{\mathbf{c}_k\}_{k=1}^{K}$.
2. Repeat the following steps till converge
   a. Assign each data to the nearest cluster: $\forall\, \boldsymbol{x}_n$

$$\boldsymbol{x}_n \in \mathcal{C}_k, \quad \text{if } k = \arg\min_{k\in\{1,\ldots,K\}} d(\boldsymbol{x}_n, \mathbf{c}_k). \tag{1}$$

   b. Update the centriods:

$$\mathbf{c}_k = \frac{1}{|\mathcal{C}_k|} \sum\nolimits_{x_n \in \mathcal{C}_k} \boldsymbol{x}_n. \tag{2}$$

Extensions of K-means

Classic K-means: Consider the samples and the centroids in the Euclidean space

- $d(\boldsymbol{x}_n, \mathbf{c}_k)$ is the Euclidean distance.
- The new centroids are updated via averaging samples.

Extensions of K-means

Classic K-means: Consider the samples and the centroids in the Euclidean space

- $d(\boldsymbol{x}_n, \mathbf{c}_k)$ is the Euclidean distance.
- The new centroids are updated via averaging samples.

K-means defined in other spaces:

- $d(\boldsymbol{x}_n, \mathbf{c}_k)$ can be other valid metrics.
- The new centroids correspond to the **barycenters** of clusters computed.

Extensions of K-means

Classic K-means: Consider the samples and the centroids in the Euclidean space

- $d(\boldsymbol{x}_n, \mathbf{c}_k)$ is the Euclidean distance.
- The new centroids are updated via averaging samples.

K-means defined in other spaces:

- $d(\boldsymbol{x}_n, \mathbf{c}_k)$ can be other valid metrics.
- The new centroids correspond to the **barycenters** of clusters computed.

(Let's revisit the Euclidean average as the Euclidean barycenter.)

Spectral Clustering

Motivations:

- Overcome the drawbacks of (classic) K-means
  - Not work well for linearly inseparable data
  - Curse of dimensionality

Spectral Clustering

Motivations:

- Overcome the drawbacks of (classic) K-means
  - Not work well for linearly inseparable data
  - Curse of dimensionality

Principle:

- Use the **spectrum** (eigenvalues) of the **similarity matrix** of the data to perform dimensionality reduction before clustering.

Spectral Clustering

Motivations:

- Overcome the drawbacks of (classic) K-means
  - Not work well for linearly inseparable data
  - Curse of dimensionality

Principle:

- Use the **spectrum** (eigenvalues) of the **similarity matrix** of the data to perform dimensionality reduction before clustering.

The spectrum of a symmetric matrix: For a symmetric $\boldsymbol{A} \in \mathbb{R}^{N \times N}$:

$$\boldsymbol{A} = \boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{U}^T = \sum\nolimits_{n=1}^{N} \lambda_n \boldsymbol{u}_n \boldsymbol{u}_n^T.$$

Spectral Clustering

Motivations:

- Overcome the drawbacks of (classic) K-means
  - Not work well for linearly inseparable data
  - Curse of dimensionality

Principle:

- Use the **spectrum** (eigenvalues) of the **similarity matrix** of the data to perform dimensionality reduction before clustering.

The spectrum of a symmetric matrix: For a symmetric $\boldsymbol{A} \in \mathbb{R}^{N \times N}$:

$$\boldsymbol{A} = \boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{U}^T = \sum_{n=1}^{N} \lambda_n \boldsymbol{u}_n \boldsymbol{u}_n^T. \tag{3}$$

Question:

- What is the difference between the eigen-decomposition and the SVD of $\boldsymbol{A}$?

Spectral Clustering

Motivations:

- Overcome the drawbacks of (classic) K-means
  - Not work well for linearly inseparable data
  - Curse of dimensionality

Principle:

- Use the **spectrum** (eigenvalues) of the **similarity matrix** of the data to perform dimensionality reduction before clustering.

The spectrum of a symmetric matrix: For a symmetric $\boldsymbol{A} \in \mathbb{R}^{N \times N}$:

$$\boldsymbol{A} = \boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{U}^T = \sum\nolimits_{n=1}^{N} \lambda_n \boldsymbol{u}_n \boldsymbol{u}_n^T. \tag{3}$$

Question:

- What is the difference between the eigen-decomposition and the SVD of $\boldsymbol{A}$?
- When are they same?

Spectral Clustering

Solution: Given a set of data $\{\boldsymbol{x}_n\}_{n=1}^{N}$

1 Define a similarity matrix $\boldsymbol{A} = [a(\boldsymbol{x}_n, \boldsymbol{x}_m)] \in \mathbb{R}^{N \times N}$.

Spectral Clustering

Solution: Given a set of data $\{\boldsymbol{x}_n\}_{n=1}^{N}$

1. Define a similarity matrix $\boldsymbol{A} = [a(\boldsymbol{x}_n, \boldsymbol{x}_m)] \in \mathbb{R}^{N \times N}$.
2. Define a **Laplacian** matrix $\boldsymbol{L} = \operatorname{diag}(\boldsymbol{A}\boldsymbol{1}_N) - \boldsymbol{A}$.

Spectral Clustering

Solution: Given a set of data $\{\boldsymbol{x}_n\}_{n=1}^N$

1. Define a similarity matrix $\boldsymbol{A} = [a(\boldsymbol{x}_n, \boldsymbol{x}_m)] \in \mathbb{R}^{N \times N}$.
2. Define a **Laplacian** matrix $\boldsymbol{L} = \mathrm{diag}(\boldsymbol{A}\mathbf{1}_N) - \boldsymbol{A}$.
3. Apply the eigenvalue decomposition of $\boldsymbol{L} = \boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{U}^T$, $0 = \lambda_1 \leq \cdots \leq \lambda_N$.

Spectral Clustering

Solution: Given a set of data $\{\boldsymbol{x}_n\}_{n=1}^N$

1. Define a similarity matrix $\boldsymbol{A} = [a(\boldsymbol{x}_n, \boldsymbol{x}_m)] \in \mathbb{R}^{N \times N}$.
2. Define a **Laplacian** matrix $\boldsymbol{L} = \mathrm{diag}(\boldsymbol{A}\mathbf{1}_N) - \boldsymbol{A}$.
3. Apply the eigenvalue decomposition of $\boldsymbol{L} = \boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{U}^T$, $0 = \lambda_1 \leq \cdots \leq \lambda_N$.
4. Consider the eigenvectors corresponding to the top-L smallest eigenvalues, e.g., $\boldsymbol{U}_L \in \mathbb{R}^{N \times L}$.

Spectral Clustering

Solution: Given a set of data $\{\boldsymbol{x}_n\}_{n=1}^N$

1. Define a similarity matrix $\boldsymbol{A} = [a(\boldsymbol{x}_n, \boldsymbol{x}_m)] \in \mathbb{R}^{N\times N}$.
2. Define a **Laplacian** matrix $\boldsymbol{L} = \mathrm{diag}(\boldsymbol{A}\boldsymbol{1}_N) - \boldsymbol{A}$.
3. Apply the eigenvalue decomposition of $\boldsymbol{L} = \boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{U}^T$, $0 = \lambda_1 \leq \cdots \leq \lambda_N$.
4. Consider the eigenvectors corresponding to the top-L smallest eigenvalues, e.g., $\boldsymbol{U}_L \in \mathbb{R}^{N\times L}$.
5. Applying K-means to the rows of $\boldsymbol{U}_L$.

Spectral Clustering

Solution: Given a set of data $\{\boldsymbol{x}_n\}_{n=1}^{N}$

1. Define a similarity matrix $\boldsymbol{A} = [a(\boldsymbol{x}_n, \boldsymbol{x}_m)] \in \mathbb{R}^{N\times N}$.
2. Define a **Laplacian** matrix $\boldsymbol{L} = \text{diag}(\boldsymbol{A}\mathbf{1}_N) - \boldsymbol{A}$.
3. Apply the eigenvalue decomposition of $\boldsymbol{L} = \boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{U}^T$, $0 = \lambda_1 \leq \cdots \leq \lambda_N$.
4. Consider the eigenvectors corresponding to the top-L smallest eigenvalues, e.g., $\boldsymbol{U}_L \in \mathbb{R}^{N\times L}$.
5. Applying K-means to the rows of $\boldsymbol{U}_L$.

Spectral Clustering = Manifold Learning + K-means, where the Manifold learning method is called **Laplacian Eigenmap**

Connect Spectral Clustering with Laplacian Eigenmap

- Recall that we have define a similarity matrix $\boldsymbol{A}$ for the samples, each element $a(\boldsymbol{x}_m, \boldsymbol{x}_n) = a_{mn} \in [0, 1]$ measures the similarity between $\boldsymbol{x}_m$ and $\boldsymbol{x}_n$.

Connect Spectral Clustering with Laplacian Eigenmap

- Recall that we have define a similarity matrix $\boldsymbol{A}$ for the samples, each element $a(\boldsymbol{x}_m, \boldsymbol{x}_n) = a_{mn} \in [0, 1]$ measures the similarity between $\boldsymbol{x}_m$ and $\boldsymbol{x}_n$.
- If we want to reduce the dimension of the data, i.e., obtaining low-dimensional representations $\{\boldsymbol{z}_n\}_{n=1}^{N}$, a reasonable criterion/objective is:

$$\min_{\boldsymbol{Z} \in \mathbb{R}^{N \times L}} \sum_{m,n=1}^{N} \|\boldsymbol{z}_m - \boldsymbol{z}_n\|^2 a(\boldsymbol{x}_m, \boldsymbol{x}_n) \tag{4}$$

(For the highly-similar paired samples, their representations should own a short distance)

Connect Spectral Clustering with Laplacian Eigenmap

- Equivalent formulation:

$$\sum_{m,n=1}^{N} \|\boldsymbol{z}_m - \boldsymbol{z}_n\|^2 a_{mn} = \sum_{m,n=1}^{N} (\boldsymbol{z}_m^T \boldsymbol{z}_m + \boldsymbol{z}_n^T \boldsymbol{z}_n - 2\boldsymbol{z}_m^T \boldsymbol{z}_n) a_{mn}$$

Connect Spectral Clustering with Laplacian Eigenmap

- Equivalent formulation:

$$\sum_{m,n=1}^{N} \|\mathbf{z}_m - \mathbf{z}_n\|^2 a_{mn} = \sum_{m,n=1}^{N} (\mathbf{z}_m^T \mathbf{z}_m + \mathbf{z}_n^T \mathbf{z}_n - 2\mathbf{z}_m^T \mathbf{z}_n) a_{mn}$$

$$= \sum_{m=1}^{N} \mathbf{z}_m^T \mathbf{z}_m \Big(\sum_{n=1}^{N} a_{mn} \Big) + \sum_{n=1}^{N} \mathbf{z}_n^T \mathbf{z}_n \Big(\sum_{m=1}^{N} a_{mn} \Big) - 2 \sum_{m,n=1}^{N} \mathbf{z}_m^T \mathbf{z}_n a_{mn}$$

Connect Spectral Clustering with Laplacian Eigenmap

- Equivalent formulation:

$$\sum_{m,n=1}^{N} \|\boldsymbol{z}_m - \boldsymbol{z}_n\|^2 a_{mn} = \sum_{m,n=1}^{N} (\boldsymbol{z}_m^T \boldsymbol{z}_m + \boldsymbol{z}_n^T \boldsymbol{z}_n - 2\boldsymbol{z}_m^T \boldsymbol{z}_n) a_{mn}$$

$$= \sum_{m=1}^{N} \boldsymbol{z}_m^T \boldsymbol{z}_m (\sum_{n=1}^{N} a_{mn}) + \sum_{n=1}^{N} \boldsymbol{z}_n^T \boldsymbol{z}_n (\sum_{m=1}^{N} a_{mn}) - 2 \sum_{m,n=1}^{N} \boldsymbol{z}_m^T \boldsymbol{z}_n a_{mn}$$

$$= 2\text{trace}(\boldsymbol{Z}^T \text{diag}(\boldsymbol{A}\boldsymbol{1}_N)\boldsymbol{Z}) - 2\text{trace}(\boldsymbol{Z}^T \boldsymbol{A}\boldsymbol{Z})$$

Connect Spectral Clustering with Laplacian Eigenmap

- Equivalent formulation:

$$\sum_{m,n=1}^{N} \|\boldsymbol{z}_m - \boldsymbol{z}_n\|^2 a_{mn} = \sum_{m,n=1}^{N} (\boldsymbol{z}_m^T \boldsymbol{z}_m + \boldsymbol{z}_n^T \boldsymbol{z}_n - 2\boldsymbol{z}_m^T \boldsymbol{z}_n) a_{mn}$$

$$= \sum_{m=1}^{N} \boldsymbol{z}_m^T \boldsymbol{z}_m \left(\sum_{n=1}^{N} a_{mn} \right) + \sum_{n=1}^{N} \boldsymbol{z}_n^T \boldsymbol{z}_n \left(\sum_{m=1}^{N} a_{mn} \right) - 2 \sum_{m,n=1}^{N} \boldsymbol{z}_m^T \boldsymbol{z}_n a_{mn}$$

$$= 2\text{trace}(\boldsymbol{Z}^T \text{diag}(\boldsymbol{A}\mathbf{1}_N)\boldsymbol{Z}) - 2\text{trace}(\boldsymbol{Z}^T \boldsymbol{A}\boldsymbol{Z})$$

$$= 2\text{trace}(\boldsymbol{Z}^T (\text{diag}(\boldsymbol{A}\mathbf{1}_N) - \boldsymbol{A})\boldsymbol{Z}) = 2\text{trace}(\boldsymbol{Z}^T \boldsymbol{L}\boldsymbol{Z})$$

Connect Spectral Clustering with Laplacian Eigenmap

- Equivalent formulation:

$$\begin{aligned}&\sum\nolimits_{m,n=1}^{N} \|\mathbf{z}_m - \mathbf{z}_n\|^2 a_{mn} = \sum\nolimits_{m,n=1}^{N} (\mathbf{z}_m^T\mathbf{z}_m + \mathbf{z}_n^T\mathbf{z}_n - 2\mathbf{z}_m^T\mathbf{z}_n)a_{mn} \\ &= \sum\nolimits_{m=1}^{N} \mathbf{z}_m^T\mathbf{z}_m(\sum\nolimits_{n=1}^{N} a_{mn}) + \sum\nolimits_{n=1}^{N} \mathbf{z}_n^T\mathbf{z}_n(\sum\nolimits_{m=1}^{N} a_{mn}) - 2\sum\nolimits_{m,n=1}^{N} \mathbf{z}_m^T\mathbf{z}_n a_{mn} \\ &= 2\mathrm{trace}(\boldsymbol{Z}^T\mathrm{diag}(\boldsymbol{A}\mathbf{1}_N)\boldsymbol{Z}) - 2\mathrm{trace}(\boldsymbol{Z}^T\boldsymbol{A}\boldsymbol{Z}) \\ &= 2\mathrm{trace}(\boldsymbol{Z}^T(\mathrm{diag}(\boldsymbol{A}\mathbf{1}_N) - \boldsymbol{A})\boldsymbol{Z}) = 2\mathrm{trace}(\boldsymbol{Z}^T\boldsymbol{L}\boldsymbol{Z})\end{aligned} \tag{5}$$

- We can find that
  - $\boldsymbol{L}$ is positive semidefinite
  - 0 is the smallest eigenvalue of $\boldsymbol{L}$ and the corresponding eigenvector is $\frac{1}{N}\mathbf{1}_N$.
- As a result, the Laplacian Eigenmap corresponds to

$$\min\nolimits_{Z^TZ=I_L} \mathrm{trace}(\boldsymbol{Z}^T\boldsymbol{L}\boldsymbol{Z}) \quad \Rightarrow \quad \boldsymbol{Z}^* = \boldsymbol{U}_L, \text{ where } \boldsymbol{L} = \boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{U}^T. \tag{6}$$

Connect Spectral Clustering with Kernel Methods

The construction of similarity matrix

- In general, we can apply the Gram matrix of kernel function as the similarity matrix, e.g., the RBF kernel

$$a(\boldsymbol{x}_m, \boldsymbol{x}_n) := K(\boldsymbol{x}_m, \boldsymbol{x}_n) = \exp(-\frac{\|\boldsymbol{x}_m - \boldsymbol{x}_n\|_2^2}{h}).$$

Connect Spectral Clustering with Kernel Methods

The construction of similarity matrix

- In general, we can apply the Gram matrix of kernel function as the similarity matrix, e.g., the RBF kernel

$$a(\boldsymbol{x}_m, \boldsymbol{x}_n) := K(\boldsymbol{x}_m, \boldsymbol{x}_n) = \exp(-\frac{\|\boldsymbol{x}_m - \boldsymbol{x}_n\|_2^2}{h}). \tag{7}$$

- **Question:** Suppose that $\boldsymbol{A}$ is a kernel, and define $\boldsymbol{D} = \mathrm{diag}(\boldsymbol{A}\mathbf{1}_N)$.

Connect Spectral Clustering with Kernel Methods

The construction of similarity matrix

- In general, we can apply the Gram matrix of kernel function as the similarity matrix, e.g., the RBF kernel

$$a(\boldsymbol{x}_m, \boldsymbol{x}_n) := K(\boldsymbol{x}_m, \boldsymbol{x}_n) = \exp(-\frac{\|\boldsymbol{x}_m - \boldsymbol{x}_n\|_2^2}{h}). \tag{7}$$

- **Question:** Suppose that $\boldsymbol{A}$ is a kernel, and define $\boldsymbol{D} = \mathrm{diag}(\boldsymbol{A}\boldsymbol{1}_N)$. A normalized Laplacian matrix is

$$\widehat{\boldsymbol{L}} = \boldsymbol{D}^{-1/2}\boldsymbol{L}\boldsymbol{D}^{-1/2} = \boldsymbol{I} - \boldsymbol{D}^{-1/2}\boldsymbol{A}\boldsymbol{D}^{-1/2} = \boldsymbol{I} - \widehat{\boldsymbol{A}}.$$

Connect Spectral Clustering with Kernel Methods

The construction of similarity matrix

- In general, we can apply the Gram matrix of kernel function as the similarity matrix, e.g., the RBF kernel

$$a(\boldsymbol{x}_m, \boldsymbol{x}_n) := K(\boldsymbol{x}_m, \boldsymbol{x}_n) = \exp(-\frac{\|\boldsymbol{x}_m - \boldsymbol{x}_n\|_2^2}{h}). \tag{7}$$

- **Question:** Suppose that $\boldsymbol{A}$ is a kernel, and define $\boldsymbol{D} = \mathrm{diag}(\boldsymbol{A}\boldsymbol{1}_N)$. A normalized Laplacian matrix is

$$\widehat{\boldsymbol{L}} = \boldsymbol{D}^{-1/2}\boldsymbol{L}\boldsymbol{D}^{-1/2} = \boldsymbol{I} - \boldsymbol{D}^{-1/2}\boldsymbol{A}\boldsymbol{D}^{-1/2} = \boldsymbol{I} - \widehat{\boldsymbol{A}}. \tag{8}$$

  Is the Laplacian Eigenmap defined on $\widehat{\boldsymbol{L}}$ equivalent to the Kernel PCA defined on $\widehat{\boldsymbol{A}}$?

How to evaluate your clustering results?

Evaluation: When Ground Truth Is Available

Purity

- $\Omega = \{w_1, ..., w_K\}$ is the set of K clusters, each w_k contains the indices of the samples in the k-th cluster.

Evaluation: When Ground Truth Is Available

Purity

- $\Omega = \{w_1, ..., w_K\}$ is the set of K clusters, each w_k contains the indices of the samples in the k-th cluster.
- $\mathcal{C} = \{c_1, ..., c_J\}$ is the set of J classes (ground truth), each c_j contains the indices of the samples in the j-th class.

Evaluation: When Ground Truth Is Available

Purity

- $\Omega = \{w_1, \dots, w_K\}$ is the set of K clusters, each w_k contains the indices of the samples in the k-th cluster.
- $\mathcal{C} = \{c_1, \dots, c_J\}$ is the set of J classes (ground truth), each c_j contains the indices of the samples in the j-th class.
- Purity: assign each cluster to the class which is most frequent in the cluster, and calculate the averaged accuracy of the assignment.

$$\text{Purity}(\Omega, \mathcal{C}) := \frac{1}{N} \sum_{k=1}^{K} \max_{j \in \{1,\dots,J\}} |w_k \cap c_j|, \tag{9}$$

where N is the number of samples.

Evaluation: When Ground Truth Is Available

Purity

- $\Omega = \{w_1, \dots, w_K\}$ is the set of K clusters, each w_k contains the indices of the samples in the k-th cluster.
- $\mathcal{C} = \{c_1, \dots, c_J\}$ is the set of J classes (ground truth), each c_j contains the indices of the samples in the j-th class.
- Purity: assign each cluster to the class which is most frequent in the cluster, and calculate the averaged accuracy of the assignment.

$$\text{Purity}(\Omega, \mathcal{C}) := \frac{1}{N} \sum_{k=1}^{K} \max_{j \in \{1, \dots, J\}} |w_k \cap c_j|, \tag{9}$$

  where N is the number of samples.
- What is its drawback?

Evaluation: When Ground Truth Is Available

Purity

- $\Omega = \{w_1, ..., w_K\}$ is the set of K clusters, each w_k contains the indices of the samples in the k-th cluster.
- $\mathcal{C} = \{c_1, ..., c_J\}$ is the set of J classes (ground truth), each c_j contains the indices of the samples in the j-th class.
- Purity: assign each cluster to the class which is most frequent in the cluster, and calculate the averaged accuracy of the assignment.

$$\text{Purity}(\Omega, \mathcal{C}) := \frac{1}{N} \sum_{k=1}^{K} \max_{j \in \{1,\ldots,J\}} |w_k \cap c_j|, \tag{9}$$

  where N is the number of samples.

- What is its drawback? (Consider the case with $K \geq J$)

Evaluation: When Ground Truth Is Available

Normalized Mutual Information (NMI)

- Overcome the drawback of purity: achieve a trade-off between the quality of the clustering and the number of clusters.

Evaluation: When Ground Truth Is Available

Normalized Mutual Information (NMI)

- Overcome the drawback of purity: achieve a trade-off between the quality of the clustering and the number of clusters.
- $P(w_k) = \frac{|w_k|}{N}$ is the probability of a sample belonging to the k-th cluster.
- $P(c_j) = \frac{|c_j|}{N}$ is the probability of a sample belonging to the j-th class.

Evaluation: When Ground Truth Is Available

Normalized Mutual Information (NMI)

- Overcome the drawback of purity: achieve a trade-off between the quality of the clustering and the number of clusters.
- $P(w_k) = \frac{|w_k|}{N}$ is the probability of a sample belonging to the k-th cluster.
- $P(c_j) = \frac{|c_j|}{N}$ is the probability of a sample belonging to the j-th class.
- $P(w_k \cap c_j) = \frac{|w_k \cap c_j|}{N}$ is the probability of a sample belonging to both the k-th cluster and the j-th class.

Evaluation: When Ground Truth Is Available

Normalized Mutual Information (NMI)

- Overcome the drawback of purity: achieve a trade-off between the quality of the clustering and the number of clusters.
- $P(w_k) = \frac{|w_k|}{N}$ is the probability of a sample belonging to the k-th cluster.
- $P(c_j) = \frac{|c_j|}{N}$ is the probability of a sample belonging to the j-th class.
- $P(w_k \cap c_j) = \frac{|w_k \cap c_j|}{N}$ is the probability of a sample belonging to both the k-th cluster and the j-th class.
- NMI:

$$\mathrm{NMI}(\Omega, \mathcal{C}) = \frac{2I(\Omega, \mathcal{C})}{H(\Omega) + H(\mathcal{C})}, \tag{10}$$

  - **Mutual Information:** $I(\Omega, \mathcal{C}) = \sum_{k,j} P(w_k \cap c_j) \log \frac{P(w_k \cap c_j)}{P(w_k)P(c_j)}$.
  - **Entropy:** $H(\Omega) = -\sum_k P(w_k) \log P(w_k)$

Evaluation: When Ground Truth Is Available

Rand Index (RI): The percentage of pairwise decision correctness.

Evaluation: When Ground Truth Is Available

Rand Index (RI): The percentage of pairwise decision correctness.

- Consider $N(N-1)/2$ pairs of the samples.
    - **True Positive (TP):** The percentage of the paired samples in the same class assigned to the same cluster
    - **True Negative (TN):** The percentage of the paired samples in different classes assigned to different clusters

Evaluation: When Ground Truth Is Available

Rand Index (RI): The percentage of pairwise decision correctness.

- Consider $N(N-1)/2$ pairs of the samples.
  - **True Positive (TP):** The percentage of the paired samples in the same class assigned to the same cluster
  - **True Negative (TN):** The percentage of the paired samples in different classes assigned to different clusters
  - **False Positive (FP):** The percentage of the paired samples in different classes assigned to the same cluster
  - **False Negative (FN):** The percentage of the paired samples in the same class assigned to different clusters

Evaluation: When Ground Truth Is Available

Rand Index (RI): The percentage of pairwise decision correctness.

- Consider $N(N-1)/2$ pairs of the samples.
    - **True Positive (TP):** The percentage of the paired samples in the same class assigned to the same cluster
    - **True Negative (TN):** The percentage of the paired samples in different classes assigned to different clusters
    - **False Positive (FP):** The percentage of the paired samples in different classes assigned to the same cluster
    - **False Negative (FN):** The percentage of the paired samples in the same class assigned to different clusters
- Rand Index:

$$\mathrm{RI} = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FN}} \tag{11}$$

Evaluation: When Ground Truth Is Available

Other measurements:

- Precision, recall, and F1 score:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}, \quad \text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}, \quad \text{F1} = \frac{2P \cdot R}{P + R}$$

Evaluation: When Ground Truth Is Available

Other measurements:

- Precision, recall, and F1 score:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}, \quad \text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}, \quad \text{F1} = \frac{2P \cdot R}{P + R} \tag{12}$$

- Jaccard index:

$$\text{JI} = \frac{\text{TP}}{\text{TP} + \text{FP} + \text{FN}}$$

Evaluation: When Ground Truth Is Available

Other measurements:

- Precision, recall, and F1 score:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}, \quad \text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}, \quad \text{F1} = \frac{2P \cdot R}{P + R} \tag{12}$$

- Jaccard index:

$$\text{JI} = \frac{\text{TP}}{\text{TP} + \text{FP} + \text{FN}} \tag{13}$$

- Dice index:

$$\text{DI} = \frac{2\text{TP}}{2\text{TP} + \text{FP} + \text{FN}}$$

Evaluation: When Ground Truth Is Available

Other measurements:

- Precision, recall, and F1 score:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}, \quad \text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}, \quad \text{F1} = \frac{2P \cdot R}{P + R} \tag{12}$$

- Jaccard index:

$$\text{JI} = \frac{\text{TP}}{\text{TP} + \text{FP} + \text{FN}} \tag{13}$$

- Dice index:

$$\text{DI} = \frac{2\text{TP}}{2\text{TP} + \text{FP} + \text{FN}} \tag{14}$$

- Fowlkes-Mallows Index:

$$\text{FMI} = \sqrt{\frac{\text{TP}}{\text{TP} + \text{FP}} \cdot \frac{\text{TP}}{\text{TP} + \text{FN}}} = \sqrt{P \cdot R} \tag{15}$$

Evaluation: When Ground Truth Is Unavailable

- The above measurements are called “external” evaluation methods because of using external ground truth information.
- The evaluation methods not relying on the ground truth are called “internal” measurements.

Evaluation: When Ground Truth Is Unavailable

- The above measurements are called “external” evaluation methods because of using external ground truth information.
- The evaluation methods not relying on the ground truth are called “internal” measurements.

Davies-Bouldin Index:

$$\mathrm{DBI} = \frac{1}{K}\sum\nolimits_{i=1}^{K} \max\nolimits_{j\neq i} \frac{\sigma_i + \sigma_j}{d(\mathbf{c}_i, \mathbf{c}_j)} \tag{16}$$

- $\mathbf{c}_i$ is the centroid of the i-th cluster
- σ_i is the average distance of all the samples in the i-th cluster to the centroid $\mathbf{c}_i$.

Evaluation: When Ground Truth Is Unavailable

- The above measurements are called “external” evaluation methods because of using external ground truth information.
- The evaluation methods not relying on the ground truth are called “internal” measurements.

Davies-Bouldin Index:

$$\mathrm{DBI} = \frac{1}{K} \sum_{i=1}^{K} \max_{j \neq i} \frac{\sigma_i + \sigma_j}{d(\mathbf{c}_i, \mathbf{c}_j)} \tag{16}$$

- $\mathbf{c}_i$ is the centroid of the i-th cluster
- σ_i is the average distance of all the samples in the i-th cluster to the centroid $\mathbf{c}_i$.

Principle:

- Encourage low intra-cluster distances and high inter-cluster distance.
- In general, the lower DBI is, the better clustering result we have.

Evaluation: When Ground Truth Is Unavailable

Dunn Index:

- Aim at identifying dense and well-separated clusters.
- The ratio between the minimal inter-cluster distance to maximal intra-cluster distance

$$\mathrm{DI} = \frac{\min_{1 \leq i < j \leq K} d(\mathbf{c}_i, \mathbf{c}_j)}{\max_{1 \leq i \leq K, \boldsymbol{x}_n \in c_i} d(\mathbf{c}_i, \boldsymbol{x}_n)} \tag{17}$$

Evaluation: When Ground Truth Is Unavailable

Silhouette:

- Given the i-th cluster $\mathcal{C}_i$, for the n-th sample in it, e.g., $n \in \mathcal{C}_i$. The averaged distance of the sample to other samples in the same cluster is

$$a(n) = \frac{1}{|\mathcal{C}_i - 1|} \sum\nolimits_{m \in \mathcal{C}_i, m \neq n} d(\boldsymbol{x}_m, \boldsymbol{x}_n)$$

Evaluation: When Ground Truth Is Unavailable

Silhouette:

- Given the i-th cluster $\mathcal{C}_i$, for the n-th sample in it, e.g., $n \in \mathcal{C}_i$. The averaged distance of the sample to other samples in the same cluster is

$$a(n) = \frac{1}{|\mathcal{C}_i - 1|} \sum\nolimits_{m \in \mathcal{C}_i, m \neq n} d(\boldsymbol{x}_m, \boldsymbol{x}_n) \tag{18}$$

- The smallest averaged distance to the samples in other clusters is

$$b(n) = \min_{j \in \{1,\ldots,K\},\ \text{and}\ j \neq i} \frac{1}{|\mathcal{C}_j|} \sum\nolimits_{m \in \mathcal{C}_j} d(\boldsymbol{x}_m, \boldsymbol{x}_n). \tag{19}$$

Evaluation: When Ground Truth Is Unavailable

Silhouette:

- The silhouette value of $\boldsymbol{x}_n$ is defined as

$$s(n) = \begin{cases} 1 - a(n)/b(n), & a(n) < b(n) \\ 0, & a(n) = b(n) \\ b(n)/a(n) - 1, & a(n) > b(n) \end{cases}$$

Evaluation: When Ground Truth Is Unavailable

Silhouette:

- The silhouette value of $\boldsymbol{x}_n$ is defined as

$$s(n) = \begin{cases} 1 - a(n)/b(n), & a(n) < b(n) \\ 0, & a(n) = b(n) \\ b(n)/a(n) - 1, & a(n) > b(n) \end{cases} \tag{20}$$

- Setting the number of clusters as k, the averaged silhouette value of all the data points measures the tightness of the clusters.

$$\bar{s}_k = \frac{1}{N} \sum_{n=1}^{N} s_k(n) \tag{21}$$

- Setting the number of clusters from 1 to K, the silhouette coefficient is defined as

$$\mathrm{SC} = \max_{k \in \{1,\ldots,K\}} \bar{s}_k \tag{22}$$

In Summary

- The motivations and applications of data clustering
- K-means and Spectral Clustering
- Evaluation methods and challenges

Next...

- Parametric Data Clustering: Gaussian Mixture Model
- EM algorithm
- Revisit K-means from a Statistical viewpoint

Homework 3, DDL: April 27, 2022

Python Programming

1. Lab # 5 (4 Pts, Done)
2. Lab # 6 (4 Pts)

Questions for Tech Report (6 Pts, $\leq$ 3 Pages)

1. **Alternating Optimization of RPCA.** When doing robust PCA (RPCA), if we assume $\boldsymbol{X} = \boldsymbol{U}\boldsymbol{V}^T$, where $\boldsymbol{U} \in \mathbb{R}^{N \times L}$ and $\boldsymbol{V} \in \mathbb{R}^{D \times L}$, the RPCA problem becomes

   $$\min_{U,V} \|\boldsymbol{X}_{noisy} - \boldsymbol{U}\boldsymbol{V}^T\|_1, \quad \text{where } \|\boldsymbol{A}\|_1 = \sum_{i,j} |a_{ij}|. \tag{23}$$

   Can we solve this problem via alternating optimization? e.g.,

   $$\boldsymbol{U}_t = \arg\min_U \|\boldsymbol{X}_{noisy} - \boldsymbol{U}\boldsymbol{V}_t^T\|_1, \quad \text{and} \quad \boldsymbol{V}_t = \arg\min_V \|\boldsymbol{X}_{noisy} - \boldsymbol{U}_t\boldsymbol{V}^T\|_1. \tag{24}$$

   If yes, derive the algorithm, otherwise, show your reason. (4 Pts)
2. Proof that ISOMAP, LLE, Kernel PCA, and Eigenmap lead to the same problem: $\min_{\boldsymbol{Z} \in \mathbb{R}^{N \times L}} \mathrm{tr}(\boldsymbol{Z}^T \boldsymbol{\Phi} \boldsymbol{Z}),\ s.t.\ \boldsymbol{Z}^T\boldsymbol{Z} = \boldsymbol{I}_d$, and derive the $\boldsymbol{\Phi}$'s for the methods. (2 Pts)