

Clustering Analysis with Documents

CS4422/7263 Information Retrieval

Lecture 09

Jiho Noh

Department of Computer Science
Kennesaw State University

CS4422/7263 Summer 2025



- 1 **Clustering Methods**
 - Basic Concepts
 - Partitioning Methods
 - Hierarchical Methods
 - Density-based Methods

- 2 **Evaluation of Clustering**
 - Internal Evaluation
 - External Evaluation

What is Clustering?

- Clustering is the process of grouping a set of documents into clusters of similar items.
 - ▶ Items within a cluster should be similar.
 - ▶ Items from different clusters should be dissimilar.
- Clustering is the most representative form of **Unsupervised Learning**.
 - ▶ Unsupervised = *“There are no labeled or annotated data.”*

Clustering for Data Analysis and Applications

- Grouping similar texts or documents together and discovering **patterns**
- Identifying recurring support issues and discovering new content to drive SEO practices
- Detecting topic trends in social media
- Discovering duplicate content
- Allows for creativity in finding new applications
- Can be used as a quick method for **exploratory data analysis**

Goals of Clustering

- **General goal:**

- ▶ Put related items in the same cluster
- ▶ Put unrelated items in different clusters

- **Secondary goals:**

- ▶ Avoid very small and very large clusters
- ▶ Define clusters that are easy to explain to the user

- **Number of Clusters**

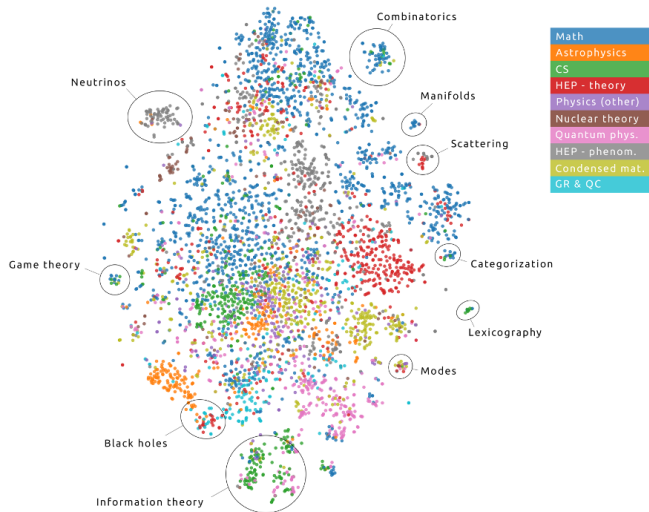
- ▶ The number of clusters should be appropriate for the data set we are clustering.
- ▶ Initially, we will assume the number of clusters k is given.
- ▶ Later, Semi-automatic methods for determining k .

Summary

- Questions?
- Discussion?

Document Clustering

- arXiv abstracts
- on 2d using t-SNE



How to Measure the Quality of Clustering

- Similarity is expressed in terms of a distance function.
- **Distance functions** differ for different types of variables:
 - ▶ Interval-scaled
 - ▶ Boolean
 - ▶ Categorical
 - ▶ Ordinal ratio
- Quality of clustering:
 - ▶ A separate “quality” function should measure the “goodness” of a cluster.
 - ▶ Defining “goodness” of a cluster is subjective.

Considerations for Cluster Analysis

- **Partitioning method**
 - ▶ Single level (e.g., k-means), hierarchical, density-based, etc.
- **Separation of clusters (hard vs. soft clustering)**
 - ▶ Can an item belong to only one cluster or multiple clusters?
- **Similarity measure**
 - ▶ Distance-based (Euclidean, Manhattan distance, cosine similarity) or Connectivity-based (density or contiguity)
- **Number of clusters**
- **Initialization methods**
- ...

Clustering Algorithms

- **Partitioning**

- ▶ K-means, K-medoids, PAM, CLARA, CLARANS

- **Hierarchy**

- ▶ BIRCH, CURE, ROCK, Chameleon

- **Density**

- ▶ DBSCAN, OPTICS, Mean-shift

- **(Distribution) Model**

- ▶ COBWEB, GMM, SOM, ART, DBCLASD

- **Graph theory**

- ▶ Louvain, Affinity propagation, Spectral clustering, InfoMap, Density peaks

- **Grid-based**

- ▶ STING, WaveCluster, CLIQUE

- **Fractal theory**

- 1 **Clustering Methods**
 - Basic Concepts
 - Partitioning Methods
 - Hierarchical Methods
 - Density-based Methods
- 2 **Evaluation of Clustering**
 - Internal Evaluation
 - External Evaluation

Partitioning Algorithms

Partitioning a dataset D into a set of k clusters, such that the sum of squared distances is minimized

$$E = \sum_{i=1}^k \sum_{p \in C_i} (p - c_i)^2$$

where c_i is the centroid of cluster C_i .

- **k-means clustering**

- ▶ Each cluster is represented by the center of the cluster.
- ▶ Vector Quantization; we use the vector space model.
- ▶ Relatedness between vectors is measured by Euclidean distance.
- ▶ **Euclidean distance vs. cosine similarity?**

Lloyd's Algorithm

Specify the number k of clusters to assign.

Randomly initialize k centroids.

while *The centroid positions change* **do**

expectation: Assign each point to its closest centroid.

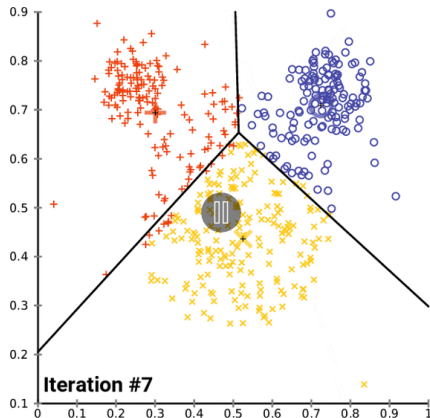
maximization: Compute the new centroid of each cluster.

end

Algorithm 1: k-means algorithm

- The Expectation-Maximization (EM) algorithm
 - ▶ **E-step:** Computes the expected value given the observed data.
 - ▶ **M-step:** Maximizing the expectation computed in E-step.

k-means Clustering Example



k-means clustering iterations

Does it converge?

- ① Residual sum of squares (RSS) decreases during each reassignment step because each vector is moved to a closer centroid

$$RSS = \sum_{k=1}^k \sum_{x \in C_k} |x - \mu_k|^2$$

- ② There is only a finite number of clusters.
- ③ Thus, we must reach a fixed point.
- ④ A finite set & monotonically decreasing evaluation function implies convergence.

Initialization of k-means

- **Random seed selection** is just one of many ways K-means can be initialized.
 - ▶ Random seed selection is not very robust; Cluster assignment converges, but it can be sub-optimal.
- We need better ways of computing initial centroids:

Methods of Initializing K-means

- **K-mean++**: selects initial cluster centroids using sampling based on an empirical probability distribution of the points' contribution to the overall inertia.
- **RP**: Randomly selected point.
- **RGC**: The data points are partitioned randomly.
- **SIMFP**: Farthest points (simple selection); the first centroid is selected as a random case. The second centroid is selected as the case maximally distant from the first. Continues
- **Hierarchical Clustering Initialization**
- **Multiple Random Initialization**

K-means++ Initialization

- 1 Choose one center uniformly at random among the data points.
- 2 For each data point x not chosen yet, compute $D(x)$, the distance between x and the nearest center that has already been chosen.
- 3 Choose one new data point at random as a new center, using a weighted probability distribution where a point x is chosen with probability proportional to $D(x)^2$.
- 4 Repeat Steps 2 and 3 until k centers have been chosen.

Hierarchical Clustering Initialization

- First, perform hierarchical clustering.
- The K clusters with the largest dissimilarity between them are selected as the initial centroids.
- Effective with complex structure.

Multiple Random Initialization

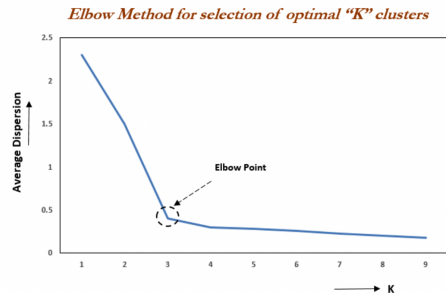
- Run K-means multiple times with different random initialization
- Select the clustering with the lowest RSS (Residual Sum of Squares)

How to find K? The Elbow Method

- Most well-known method
- Calculate the **Within-Cluster-Sum of Squared Errors (WSS)** for different values of K

$$\arg \min_{\mathbf{S}} \sum_{i=1}^k \sum_{\mathbf{x} \in S_i} \|\mathbf{x} - \mu_i\|^2 = \arg \min_{\mathbf{S}} \sum_{i=1}^k |S_i| \text{Var } S_i$$

where S_k is the set of observations in the k -th cluster.



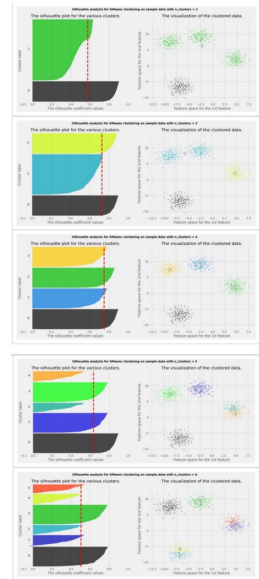
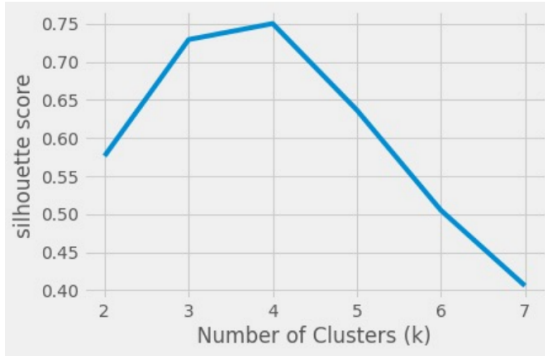
How to find K? The Silhouette Method

- The silhouette value measures how similar a point is to its own cluster (cohesion) compared to other clusters (separation).
- The range of the Silhouette value is between +0 and -1 (a high value is desirable).

Algorithm:

- $a(i)$: The average distance of that point with all other points in the same cluster.
- $b(i)$: The average distance of that point with all the points in the closest cluster to its cluster.
- $s(i)$: The silhouette value.

$$s(i) = \frac{b(i) - a(i)}{\max(b(i), a(i))}$$



- 1 **Clustering Methods**
 - Basic Concepts
 - Partitioning Methods
 - **Hierarchical Methods**
 - Density-based Methods

- 2 **Evaluation of Clustering**
 - Internal Evaluation
 - External Evaluation

Hierarchical Clustering

- Use distance matrix as clustering criteria.
- This method does not require the number of clusters k as an input, but needs a termination condition.

Agglomerative (bottom-up)

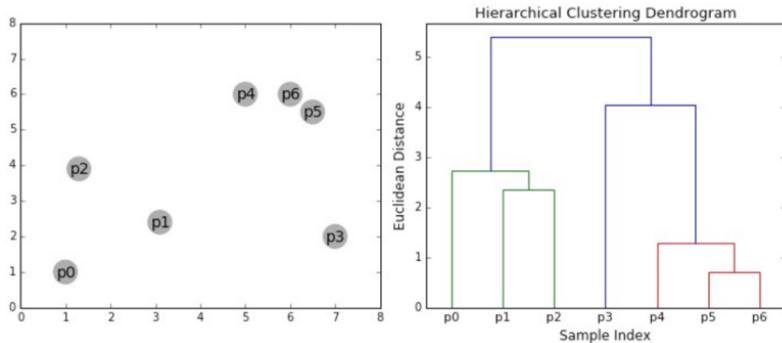
- Assign each data point as one cluster.
- Iteratively combine sub-clusters.
- Eventually, all data points is a part of 1 cluster.

Divisive (top-down)

- Assign all data points to the same cluster.
- Iteratively divide into smaller groups.
- Eventually each data point forms its own cluster.

AGNES (Agglomerative Nesting)

- 1 Assign each data point to its own cluster
- 2 Compute similarity between clusters
- 3 Merge two most similar clusters to form one cluster



Cluster Similarity

- How do we compute similar clusters?
 - ▶ Distance between two points in the clusters?
 - ▶ Distance from means of two clusters?
 - ▶ Distance between two closest points in the clusters?
- Different similarity metric could produce different types of cluster
- Common similarity metric used
 - ▶ Single linkage
 - ▶ Complete linkage
 - ▶ Average group linkage

Similarity between Clusters

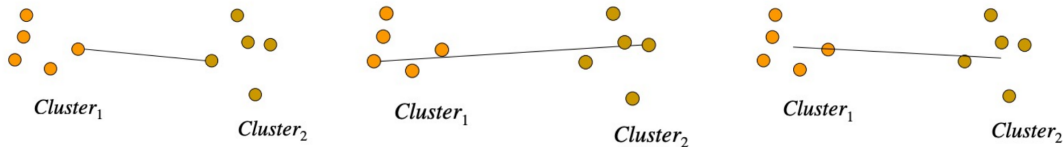


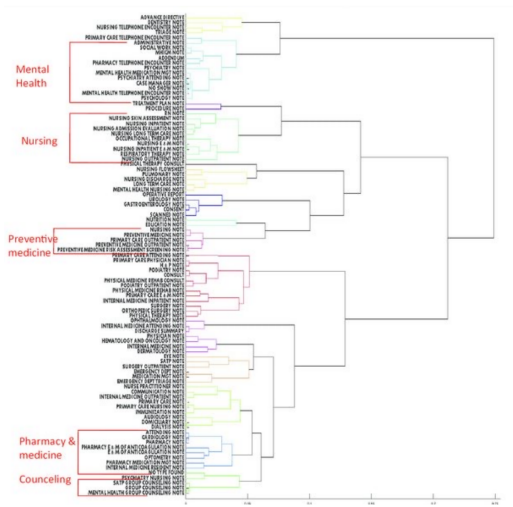
Figure: Single- (left), Complete- (middle), and Average-Linkage (right)

DIANA (DIvisive ANAlysis)

- Introduced by Kaufman and Rousseeuw in 1990
- The algorithm starts with all data points in one cluster
- Clusters are recursively divided into smaller sub-clusters
- Division continues until each data point is in its own cluster
- Based on a chosen dissimilarity measure

Dendrogram

- A tree diagram that is used to represent hierarchical relationships between data points or objects.
- It shows the similarities and differences between groups.
- A clustering of the data objects is obtained by cutting the dendrogram at the desired level, then each connected component forms a cluster.



- 1 **Clustering Methods**
 - Basic Concepts
 - Partitioning Methods
 - Hierarchical Methods
 - Density-based Methods
- 2 **Evaluation of Clustering**
 - Internal Evaluation
 - External Evaluation

Density-Based Clustering Methods

- Clustering is based on the density (local cluster criterion) of data points in the feature space.
- *How do you measure the density?*
 - ▶ By looking at the number of data points within a certain radius.
- Features
 - ▶ Discover clusters of arbitrary shape
 - ▶ Handle noise and outliers
 - ▶ Do not require the number of clusters to be specified in advance
 - ▶ Need density parameters as termination condition
- Methods:
 - ▶ DBSCAN, OPTICS, DENCLUE, CLIQUE

Density-based Clustering: Concepts

Two important parameters: ϵ (eps) and MinPts

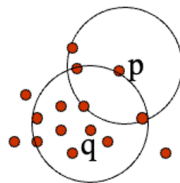
- ϵ (eps): The maximum distance between two data points to be considered in the same neighborhood
- MinPts: The minimum number of data points required to form a dense region (core point)

$$N_{\text{Eps}}(p) : \{q \in D \mid \text{dist}(p, q) \leq \text{Eps}\}$$

Core Point: A data point with at least MinPts data points in its ϵ -neighborhood

$$|N_{\text{Eps}}(q)| \geq \text{MinPts}$$

Border Point: A data point that is not a core point but is within the ϵ -neighborhood of a core point



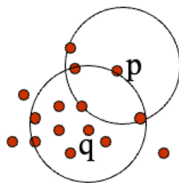
MinPts = 5

Eps = 1 cm

Density-based Clustering: Concepts (Cont.)

Directly Density-Reachability: A data point p is density-reachable from another data point q if

- $p \in N_\epsilon(q)$
- core point condition: $|N_\epsilon(q)| \geq \text{MinPts}$

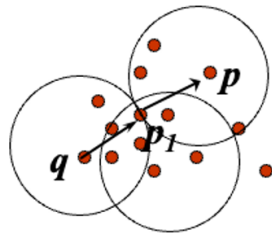


$\text{MinPts} = 5$

$\text{Eps} = 1 \text{ cm}$

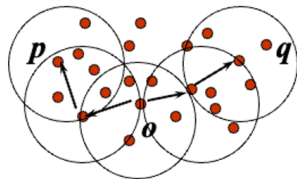
Density-based Clustering: Concepts (Cont.)

Density-Reachable: A data point p is density-reachable from another data point q if there is a chain of core points directly density-reachable among them.



Density-based Clustering: Concepts (Cont.)

Density-Connected: A data point p is density-reachable from another data point q if there is a point o such that both, p and q are density-reachable from o w.r.t. Eps and $MinPts$



DBSCAN Algorithm

- DBSCAN (Density-Based Spatial Clustering of Applications with Noise) is a popular density-based clustering algorithm.
- It groups data points together that are closely packed and marks outliers as noise.

The algorithm works by:

- 1 Finding core points by identifying data points with at least MinPts data points in their ϵ -neighborhood
- 2 Expanding the clusters by finding density-reachable points from the core points
- 3 Assigning border points to the clusters
- 4 Continue the process until all of the points have been processed
- 5 Marking noise points as outliers

- 1 **Clustering Methods**
 - Basic Concepts
 - Partitioning Methods
 - Hierarchical Methods
 - Density-based Methods

- 2 **Evaluation of Clustering**
 - Internal Evaluation
 - External Evaluation

Measuring Clustering Quality — Intrinsic

- An external reference is not needed
- Unsupervised
- Methods
 - ▶ **Silhouette Score**
 - ▶ **Davies-Bouldin Index**
 - ▶ **Dunn Index**

Measuring Clustering Quality — Extrinsic

- Compare a clustering against the ground truth
- Supervised
- Methods
 - ▶ **Adjusted Rand Index**
 - ▶ **Fowlkes-Mallows Index**
 - ▶ **Jaccard Index**

Evaluation and Assessment

- **Internal evaluation**

- ▶ The clustering is summarized to a single quality score. (e.g., Silhouette coefficient)
- ▶ (the evaluation measures themselves can be seen as a clustering objectives.)

- **External evaluation**

- ▶ The clustering is compared to an existing “ground truth” classification. (e.g., Rand index, F-measure)
- ▶ (If we have such “ground truth”, we would not need to cluster. It becomes a classification task.)

- **Manual evaluation**

- ▶ A human expert evaluates the quality of clustering.
- ▶ (Human evaluation is subjective.)

- 1 **Clustering Methods**
 - Basic Concepts
 - Partitioning Methods
 - Hierarchical Methods
 - Density-based Methods
- 2 **Evaluation of Clustering**
 - Internal Evaluation
 - External Evaluation

Internal Evaluation methods

- These methods usually assign the best score to the algorithm that produces clusters with
 - ▶ high similarity within a cluster and
 - ▶ low similarity between clusters.
- Best suited to get some insight into situations where one algorithm performs better than another.

The Silhouette Method

- The silhouette value measures how similar a point is to its own cluster (cohesion) compared to other clusters (separation)
- The range of the Silhouette value is between -1 and +1
- High Silhouette value indicates that **the object** is well matched to its own cluster and poorly matched to neighboring clusters.

The Silhouette Method (Cont.)

- Definition:

- ▶ $a(i)$: The average distance of that point with all other points in the same cluster

$$a(i) = \frac{1}{|C_I| - 1} \sum_{j \in C_I, i \neq j} d(i, j)$$

- ▶ $b(i)$: The average distance of that point with all the points in the closest cluster to its cluster

$$b(i) = \min_{J \neq I} \frac{1}{|C_J|} \sum_{j \in C_J} d(i, j)$$

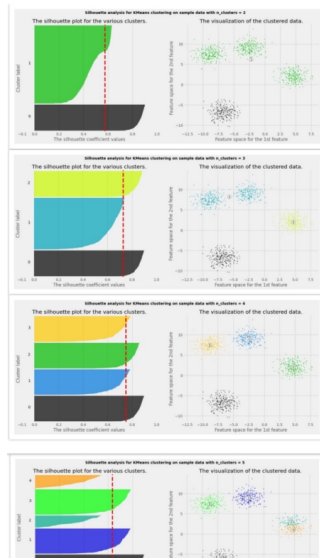
- ▶ $s(i)$: The silhouette value

$$s(i) = \frac{b(i) - a(i)}{\max(a(i), b(i))}$$

Silhouette Coefficient

- The maximum value of the mean $s(i)$ over all data of the entire dataset

$$SC = \max_k \tilde{s}(k)$$



Dunn Index

- The Dunn index is defined as the ratio between the minimal **inter-cluster distance** to **maximal intra-cluster distance**.
- For each cluster partition, we calculate

$$D = \frac{\min_{1 \leq i < j \leq n} \delta(i, j)}{\max_{1 \leq k \leq n} \Delta(k)},$$

where $\delta(i, j)$ is the inter-cluster distance between two clusters, and $\Delta(i, j)$ is the intra-cluster distance such as the maximal distance between any pair of data points in the same cluster.

- Algorithms that produce clusters with high Dunn index are desirable..

Davies-Bouldin Index

- The Davies-Bouldin index can be calculated by

$$DB = \frac{1}{n} \sum_{i=1}^n \max_{j \neq i} \left(\frac{\sigma_i + \sigma_j}{d(c_i, c_j)} \right)$$

- ▶ n is the number of clusters
 - ▶ σ_i is average distance of all data points in the cluster i to centroid c_i
 - ▶ $d(c_i, c_j)$ is the distance between the cluster centroids c_i and c_j .
- Algorithms producing clusters with the **smallest** DB index are desired.

- 1 **Clustering Methods**
 - Basic Concepts
 - Partitioning Methods
 - Hierarchical Methods
 - Density-based Methods
- 2 **Evaluation of Clustering**
 - Internal Evaluation
 - External Evaluation

External Evaluation

- In external evaluation, clustering results are evaluated based on data that was not used for clustering, such as known ground truth (GT) labels and external benchmarks.

Concerns:

- The GT can contain internal structure, which may be different from the one of clusters.
- From a **knowledge discovery** point of view, the reproduction of known knowledge may not necessarily be the intended results.

Rand Index (RI)

- Measures how similar the clusters are to the benchmark classifications.

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

- Note, the confusion matrix in the Rand Index is different from that of *accuracy*.

TP: same class & same cluster

FP: diff. class & same cluster

FN: same class & diff. cluster

TN: diff. class & diff. cluster

Adjusted Rand Index (ARI)

- One issue with the RI is that FP and FN are equally weighted to TP and TN.
- Why is this a problem?
 - ▶ When the number of clusters is large, the chance of items in different clusters become higher.
 - ▶ Scaling is necessary.

$$\begin{aligned}ARI &= \frac{RI - ExpectedRI}{Max(RI) - ExpectedRI} \\&= \frac{\sum_{ij} \binom{n_{ij}}{2} - \left[\sum_i \binom{a_i}{2} \sum_j \binom{b_j}{2} \right] / \binom{n}{2}}{\frac{1}{2} \left[\sum_i \binom{a_i}{2} \sum_j \binom{b_j}{2} \right] - \left[\sum_i \binom{a_i}{2} \sum_j \binom{b_j}{2} \right] / \binom{n}{2}}\end{aligned}$$

How to test the accuracy of clustering algorithm?

- Generally to speak, accuracy is not a good measure for clustering.
- If the ground truth labels are given and you want to get an insight according to the labels...
 - ▶ Find an optimal mapping between 'true labels' and 'predicted labels' based on the total number of matching items.

[numpy example]

```
def find_mapping(truLabels, kLabels):  
    mapp = {k: k for k in numpy.unique(kLabels)}  
    for k in numpy.unique(kLabels):  
        k_mapping = numpy.argmax(numpy.bincount(kLabels[truLabels==k]))  
        mapp[k] = k_mapping  
    return mapp
```

Jaccard Index

- The Jaccard Index quantifies the similarity between two sets.

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|} = \frac{TP}{TP + FP + FN}$$

- This measure considers the ratio of correct predictions over true positive plus all false predictions.

Normalized Mutual Information (NMI)

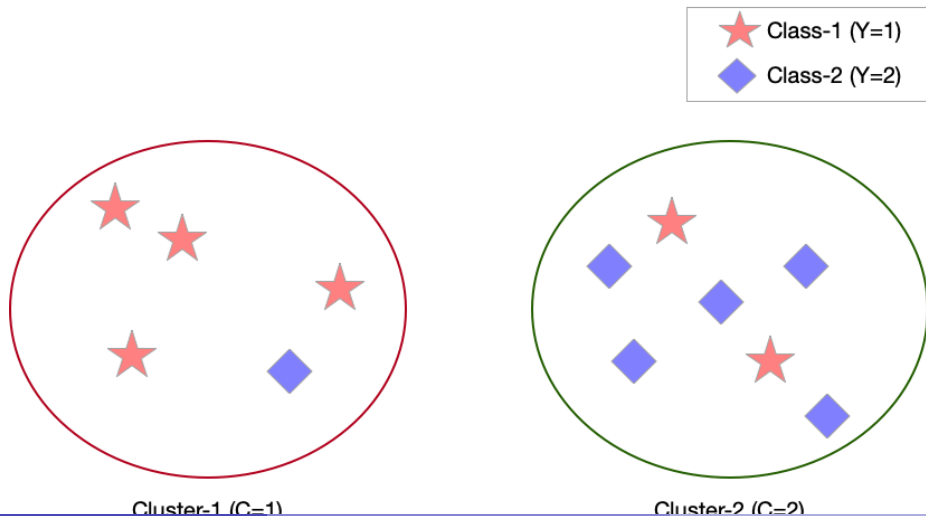
- Measures how much information is shared between a clustering and a ground-truth classification.
- NMI ranges between 0 (independent variables) and 1 (perfect correlation)

$$NMI(Y, C) = \frac{2 \times I(Y; C)}{[H(Y) + H(C)]}$$

- ▶ where Y = class labels
- ▶ C = cluster labels
- ▶ $H(\cdot)$ = Entropy
- ▶ $I(Y; C)$ = Mutual Information between Y and C

Calculating NMI for Clustering

- Assume that we have $m = 2$ classes and $k = 2$ clusters.



$H(Y)$ = Entropy of Class Labels

$$H(Y) = \mathbb{E} [-\log p(Y)] = - \sum_{y \in Y} p(y) \log p(y)$$

- $P(Y = 1) = 6/12 = 1/2$
- $P(Y = 2) = 6/12 = 1/2$
- $H(Y) = -\frac{1}{2} \log(\frac{1}{2}) - \frac{1}{2} \log(\frac{1}{2}) = 0.3010$

This prior probabilities can be pre-calculated, as it will not change depending on the clustering output

H(C) = Entropy of Cluster Labels

- $P(C = 1) = 5/12$
- $P(C = 2) = 7/12$
- $H(C) = -\frac{5}{12} \log(\frac{5}{12}) - \frac{7}{12} \log(\frac{7}{12}) = 0.2949$

This should be calculated for every clustering outputs.

$I(Y;C)$ = Mutual Information

- Mutual information measures the reduction of uncertainty after an observation.
- $I(Y;C) = H(Y) - H(Y|C)$
- We know $H(Y)$, but not $H(Y|C)$. How do we calculate the conditional entropy?

Conditional Entropy

$H(Y|C)$, Conditional entropy of class labels Y for clustering C

$$\begin{aligned} H(Y|X = x) &= p(X = x)H(Y|X = x) \\ &= -p(X = x) \sum_y p(Y = y|X = x) \log p(Y = y|X = x) \end{aligned}$$

Consider Cluster-1

- $P(Y = 1|C = 1) = 4/5$
- $P(Y = 2|C = 1) = 1/5$

$$\begin{aligned} H(Y|C = 1) &= -p(C = 1) \sum_{y=\{1,2\}} p(y|C = 1) \log p(y|C = 1) \\ &= -\frac{5}{12} \times \left[\frac{4}{5} \log\left(\frac{4}{5}\right) + \frac{1}{5} \log\left(\frac{1}{5}\right) \right] = 0.0906 \end{aligned}$$

Conditional Entropy

$$\begin{aligned} H(Y|X = x) &= p(X = x)H(Y|X = x) \\ &= -p(X = x) \sum_y p(Y = y|X = x) \log p(Y = y|X = x) \end{aligned}$$

Consider Cluster-2

- $P(Y = 1|C = 2) = 2/7$
- $P(Y = 2|C = 2) = 5/7$

$$\begin{aligned} H(Y|C = 1) &= -p(C = 2) \sum_{y=\{1,2\}} p(y|C = 2) \log p(y|C = 2) \\ &= -\frac{7}{12} \times \left[\frac{2}{7} \log\left(\frac{2}{7}\right) + \frac{5}{7} \log\left(\frac{5}{7}\right) \right] = 0.1516 \end{aligned}$$

$I(Y;C)$ = Mutual Information

$$\begin{aligned} I(Y;C) &= H(Y) - H(Y|C) \\ &= 0.3010 - (0.0906 + 0.1516) = 0.0588 \end{aligned}$$

The NMI is therefore,

$$NMI(Y,C) = \frac{2 \times I(Y;C)}{[H(Y) + H(C)]}$$

$$NMI(Y,C) = \frac{2 \times 0.0588}{[0.3010 + 0.2949]} \approx \mathbf{0.197}$$

Scikit-learn NMI

```
>>> from sklearn.metrics.cluster \
      import normalized_mutual_info_score as nmi
>>> nmi([1, 1, 1, 1, 2, 1, 2, 2, 2, 2, 1, 2],
        [1, 1, 1, 1, 1, 2, 2, 2, 2, 2, 2, 2])
0.19769959815999483
```

(Data element order: per each cluster from top-left to bottom-right)