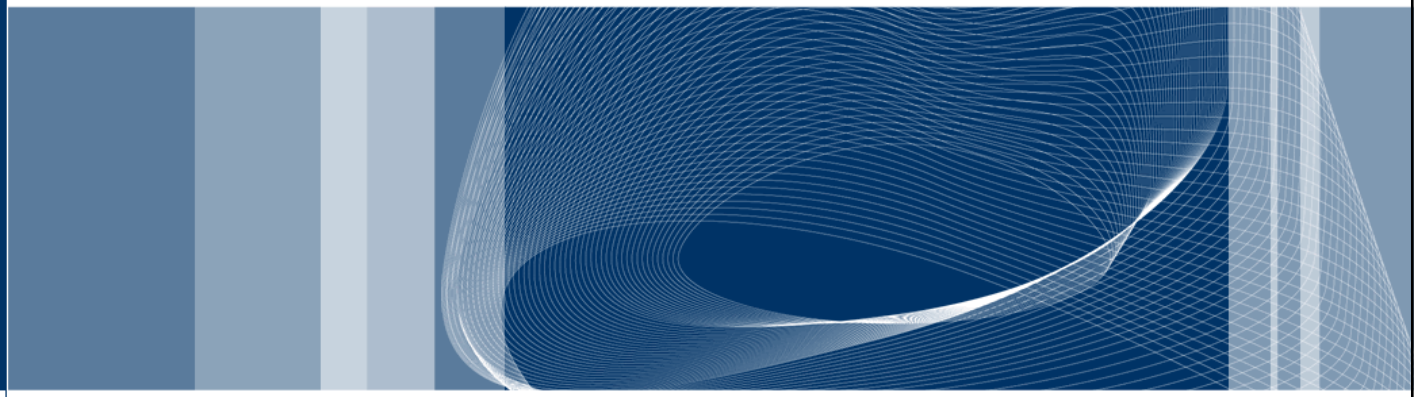
POLITECNICO DI MILANO



Clustering Validation

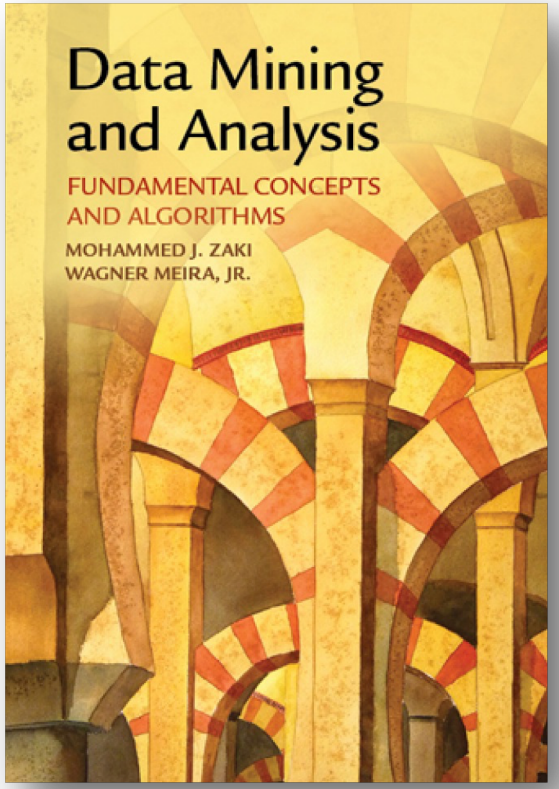
Data Science for Mobility

Prof. Pier Luca Lanzi

Readings

- "Data Mining and Analysis" by Zaki & Meira
  - Chapter 17

- http://www.dataminingbook.info



Cluster Validation and Assessment

Clustering Evaluation

assess the goodness or quality of the clustering

Clustering Stability

sensitivity of the clustering result to various algorithmic parameters

Clustering Tendency

suitability of applying clustering in the first place,
does the data have any inherent grouping structure?

Validity Measures

- External Validation Measures
  - Employ criteria that are not inherent to the dataset
  - E.g. prior or expert-specified knowledge about the clusters, for example, class labels for each point.
- Internal Validation Measures
  - Employ criteria that are derived from the data itself
  - For instance, intracluster and intercluster distances to measure cluster compactness (e.g., how similar are the points in the same cluster?) and separation (e.g., how far apart are the points in different clusters?).
- Relative Validation Measures
  - Aim to directly compare different clusterings, usually those obtained via different parameter settings for the same algorithm.

External Measures

(the correct or ground-truth clustering is known a priori)

Given a clustering partition C and
the ground truth partitioning T,
we redefine TP, TN, FP, FN
in the context of clustering

True Positives, True Negatives, False Positives, and False Negatives

- True Positives
  - x_i and x_j are a true positive pair if they belong to the same partition in T, and they are also in the same cluster in C
  - TP is defined as the number of true positive pairs
- False Negatives
  - x_i and x_j are a false negative pair if they belong to the same partition in T, but they do not belong to the same cluster in C.
  - FN is defined as the number of false negative pairs

True Positives, True Negatives, False Positives, and False Negatives

- False Positives
  - x_i and x_j are a false positive pair if the do not belong to the same partition in T, but belong to the same cluster in C
  - FP is the number of false positive pairs
- True Negatives
  - x_i and x_j are a false negative pair if they do not belong to the same partition in T, nor to the same cluster in C
  - TN is the number of true negative pairs

Given the number of pairs N

$$N=TP+FP+FN+TN$$

Jaccard Coefficient

- Measures the fraction of true positive point pairs, but after ignoring the true negatives as,

$$Jaccard = \frac{TP}{TP + FP + FN}$$

- For a perfect clustering C, the coefficient is one, that is, there are no false positives nor false negatives.

- Note that the Jaccard coefficient is asymmetric in that it ignores the true negatives

Rand Statistic

- Measures the fraction of true positives and true negatives over all pairs as

$$Rand = \frac{TP + TN}{N}$$

- The Rand statistic measures the fraction of point pairs where both the clustering C and the ground truth T agree
- A perfect clustering has a value of 1 for the statistic.
- The adjusted rand index is the extension of the rand statistic corrected for chance.

Fowlkes-Mallows Measure

- Define precision and recall analogously to what done for classification,

$$prec = \frac{TP}{TP + FP} \qquad recall = \frac{TP}{TP + FN}$$

- The Fowlkes–Mallows (FM) measure is defined as the geometric mean of the pairwise precision and recall

$$FM = \sqrt{precision \cdot recall}$$

- FM is also asymmetric in terms of the true positives and negatives because it ignores the true negatives. Its highest value is also 1, achieved when there are no false positives or negatives.

Mutual Information Based Scores

Mutual Information

- Mutual information tries to quantify the amount of shared information between the clustering C and ground truth partitioning T,

$$I(\mathcal{C},\mathcal{T}) = \sum_{i=1}^{r}\sum_{j=1}^{k} p_{ij} \log\left(\frac{p_{ij}}{p_{C_i} \cdot p_{T_j}}\right)$$

- Where
  - p_{ij} is the probability that a point in cluster i also belongs to partition j
  - p_{ci} is the probability of cluster Ci
  - p_{tj} is the probability of cluster Tj

Normalized Mutual Information

- The normalized mutual information (NMI) is defined as

$$NMI(\mathcal{C},\mathcal{T}) = \sqrt{\frac{I(\mathcal{C},\mathcal{T})}{H(\mathcal{C})} \cdot \frac{I(\mathcal{C},\mathcal{T})}{H(\mathcal{T})}} = \frac{I(\mathcal{C},\mathcal{T})}{\sqrt{H(\mathcal{C}) \cdot H(\mathcal{T})}}$$

- Where,

$$H(\mathcal{C}) = -\sum_{i=1}^{r} p_{C_i} \log p_{C_i}$$

$$H(\mathcal{T}) = -\sum_{j=1}^{k} p_{T_j} \log p_{T_j}$$

- Values close to zero indicate two label assignments that are largely independent, while values close to one indicate significant agreement.

Homogeneity, Completeness and V-measure

Homogeneity, Completeness, and V-measure

- Homogeneity
  - Each cluster contains only members of a single class.
- Completeness
  - All members of a given class are assigned to the same cluster
- V-measure
  - Harmonic mean of homogeneity and completeness
- The three measures are bounded between 0 and 1
- The higher the value the better

Internal Validation Measures

(criteria that are derived from the data itself)

Internal Validation Measures

- Based on the notions of intracluster similarity or compactness contrasted with the notions of intercluster separation
- They typically propose a trade-off to maximizing these two competing measures
- They are computed from the distance (or proximity) matrix
- The internal measures are based on various functions over the intracluster and intercluster weights.

Some Important Statistics

- Sum over all the intracluster weights over all the clusters

$$W_{in} = \frac{1}{2}\sum_{i=1}^{k} W(C_i, C_i)$$

- Sum of all intercluster weights

$$W_{out} = \frac{1}{2}\sum_{i=1}^{k} W(C_i, \overline{C_i}) = \sum_{i=1}^{k-1}\sum_{j>i} W(C_i, C_j)$$

- Number of distinct intracluster edges N_{in} and intercluster edges, N_{out}

$$N_{in} = \sum_{i=1}^{k} \binom{n_i}{2} = \frac{1}{2}\sum_{i=1}^{k} n_i(n_i - 1)$$

$$N_{out} = \sum_{i=1}^{k-1}\sum_{j=i+1}^{k} n_i \cdot n_j = \frac{1}{2}\sum_{i=1}^{k}\sum_{\substack{j=1 \\ j\neq i}}^{k} n_i \cdot n_j$$

BetaCV

- BetaCV is computed as the ratio of the mean intracluster distance to the mean intercluster distance

$$BetaCV = \frac{W_{in}/N_{in}}{W_{out}/N_{out}} = \frac{N_{out}}{N_{in}} \cdot \frac{W_{in}}{W_{out}} = \frac{N_{out}}{N_{in}} \frac{\sum_{i=1}^{k} W(C_i, C_i)}{\sum_{i=1}^{k} W(C_i, \overline{C_i})}$$

- The smaller the BetaCV ratio, the better the clustering, as it indicates that intracluster distances are on average smaller than intercluster distances

C-Index

- Let $W_{min}(N_{in})$ be the sum of the smallest N_{in} distances in the proximity matrix W, where N_{in} is the total number of intracluster edges, or point pairs
- Let $W_{max}(N_{in})$ be the sum of the largest N_{in} distances in W
- The C-index measures to what extent the clustering puts together the N_{in} points that are the closest across the k clusters.
- It is defined as,

$$Cindex = \frac{W_{in} - W_{\min}(N_{in})}{W_{\max}(N_{in}) - W_{\min}(N_{in})}$$

- The smaller the C-index, the better the clustering, as it indicates more compact clusters with relatively smaller distances within clusters rather than between clusters.

Dunn Index

- Defined as the ratio between the minimum distance between point pairs from different clusters and the maximum distance between point pairs from the same cluster

$$Dunn = \frac{W_{out}^{\min}}{W_{in}^{\max}}$$

- Where, the minimum intercluster distance is computed as,

$$W_{out}^{\min} = \min_{i,j>i} \left\{ w_{ab} | \mathbf{x}_a \in C_i, \mathbf{x}_b \in C_j \right\}$$

- And the maximum intracluster distance is computed as,

$$W_{in}^{\max} = \max_{i} \left\{ w_{ab} | \mathbf{x}_a, \mathbf{x}_b \in C_i \right\}$$

- The larger the Dunn index the better the clustering because it means even the closest distance between points in different clusters is much larger than the farthest distance between points in the same cluster. However, the Dunn index may be insensitive because the minimum intercluster and maximum intracluster distances do not capture all the information about a clustering.

Davies–Bouldin Index

- Let μ_i denote the cluster mean and $\sigma_{\mu i}$ denote the dispersion or spread of the points around the cluster mean,

$$\sigma_{\mu_i} = \sqrt{\frac{\sum_{\mathbf{x}_j \in C_i} \delta(\mathbf{x}_j, \mu_i)^2}{n_i}} = \sqrt{var(C_i)}$$

  where var(C_i) is the total variance of cluster C_i

- The Davies–Bouldin measure for a pair of clusters C_i and C_j is defined as the ratio

$$DB_{ij} = \frac{\sigma_{\mu_i} + \sigma_{\mu_j}}{\delta(\mu_i, \mu_j)}$$

- DB_{ij} measures how compact the clusters are compared to the distance between the cluster means.

Davies–Bouldin Index

- The Davies–Bouldin index is then defined as

$$DB = \frac{1}{k}\sum_{i=1}^{k}\max_{j\neq i}\{DB_{ij}\}$$

- For each cluster C_i, we pick the cluster C_j that yields the largest DB_{ij} ratio.

- The smaller the DB value the better the clustering, as it means that the clusters are well separated (i.e., the distance between cluster means is large), and each cluster is well represented by its mean (i.e., has a small spread).

Silhouette Coefficient

- Measure of both cohesion and separation of clusters, and is based on the difference between the average distance to points in the closest cluster and to points in the same cluster.
- For each point x_i we calculate its silhouette coefficient s_i as

$$s_i = \frac{\mu_{out}^{\min}(\mathbf{x}_i) - \mu_{in}(\mathbf{x}_i)}{\max\left\{\mu_{out}^{\min}(\mathbf{x}_i), \mu_{in}(\mathbf{x}_i)\right\}}$$

- Where $\mu_{in}(x_i)$ is the mean distance from x_i to points in its own cluster y_i

$$\mu_{in}(\mathbf{x}_i) = \frac{\sum_{\mathbf{x}_j \in C_{\hat{y}_i}, j \neq i} \delta(\mathbf{x}_i, \mathbf{x}_j)}{n_{\hat{y}_i} - 1}$$

Silhouette Coefficient

- And the mean of the distances from x_i to points in the closest cluster is computed as,

$$\mu_{out}^{\min}(\mathbf{x}_i) = \min_{j \neq \hat{y}_i} \left\{ \frac{\sum_{\mathbf{y} \in C_j} \delta(\mathbf{x}_i, \mathbf{y})}{n_j} \right\}$$

- The s_i value of a point lies in the interval $[-1,+1]$.
  - A value close to $+1$ indicates that x_i is much closer to points in its own cluster and is far from other clusters.
  - A value close to zero indicates that x_i is close to the boundary between two clusters.
  - A value close to -1 indicates that x_i is much closer to another cluster than its own cluster, and therefore, the point may be mis-clustered.

Silhouette Coefficient

- The silhouette coefficient is defined as the mean s_i value across all the points

$$SC = \frac{1}{n} \sum_{i=1}^{n} s_i$$

- A value close to +1 indicates a good clustering.

- Drawbacks
  - The Silhouette Coefficient is generally higher for convex clusters than other concepts of clusters, such as density based clusters like those obtained through DBSCAN

Calinski-Harabaz Index

- Given k clusters, the Calinski-Harabaz score s is given by the ratio of the between-cluster dispersion mean and the within-cluster dispersion,

$$C = \frac{BSS(C)/(k-1)}{WSS(C)/(N-k)}$$

- That is,

$$C = \frac{N-k}{k-1}\frac{BSS(C)}{WSS(C)}$$

- The score is higher when clusters are dense and well separated, which relates to a standard concept of a cluster
- The index is generally higher for convex clusters than other concepts of clusters, such as density based clusters like those obtained through DBSCAN.

Relative Measures

(compare different clusterings obtained by varying different parameters for the same algorithm, e.g., the number of clusters k)

Within/Between Clusters Sum of Squares

- Within-cluster sum of squares

$$\mathrm{WSS}(C) = \sum_{i=1}^{k} \sum_{x_j \in C_i} ||x_j - \mu_i||^2$$

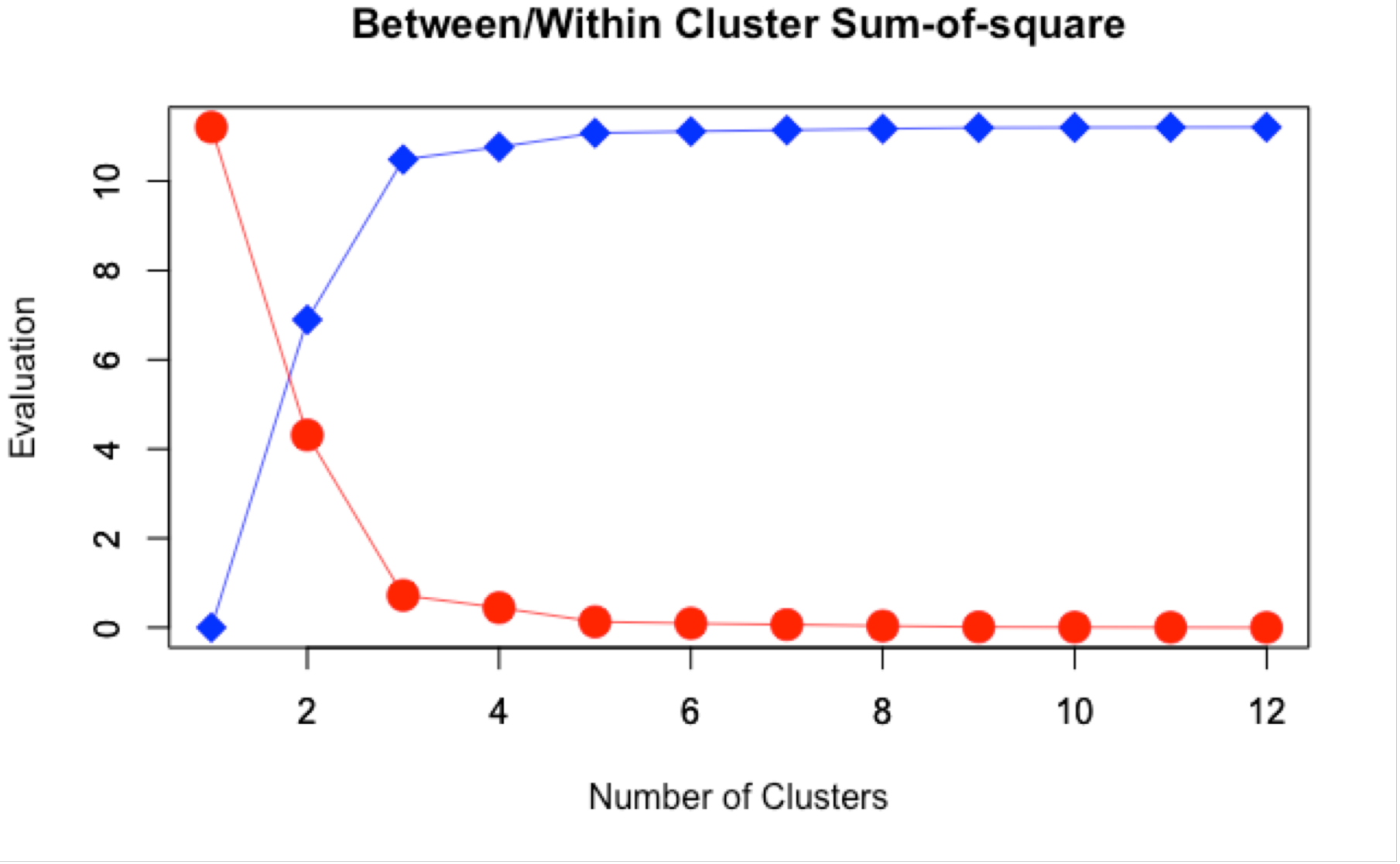
where μ_i is the centroid of cluster C_i (in case of Euclidean spaces)

- Between-cluster sum of squares

$$\mathrm{BSS}(C) = \sum_{i=1}^{k} |C_i| \cdot ||\mu - \mu_i||^2$$

where μ is the centroid of the whole dataset

Knee/Elbow Analysis of Clustering



Calinski-Harabaz Index

- We can use the Calinski-Harabaz index to select k
- In a good clustering, we expect the within-cluster scatter to be smaller relative to the between-cluster scatter, which should result in a higher value of the index
- Thus, we can either select the k corresponding to the higher index value or we can perform a knee analysis and look for a significant increase followed by much smaller differences
- For instance, we can choose the value k > 3 that minimizes,

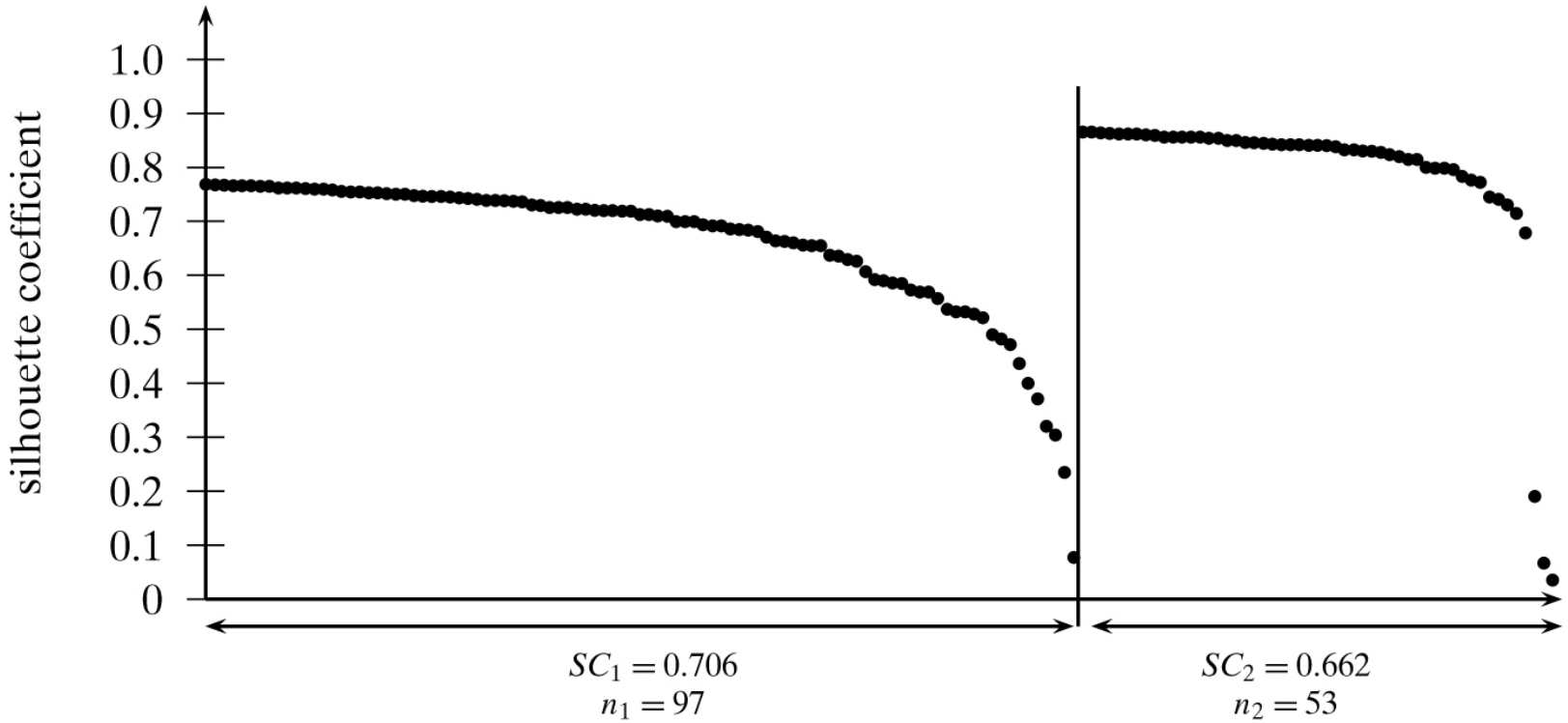
$$\Delta(k) = \Big(CH(k+1) - CH(k)\Big) - \Big(CH(k) - CH(k-1)\Big)$$

Silhouette Coefficient

- We can use the silhouette coefficient s_j of each point x_j and the average SC value to estimate the number of clusters in the data
- For each cluster, plot the s_j values in descending order
- Check the overall SC value for a particular value of k, as well as SC_i values for each cluster i

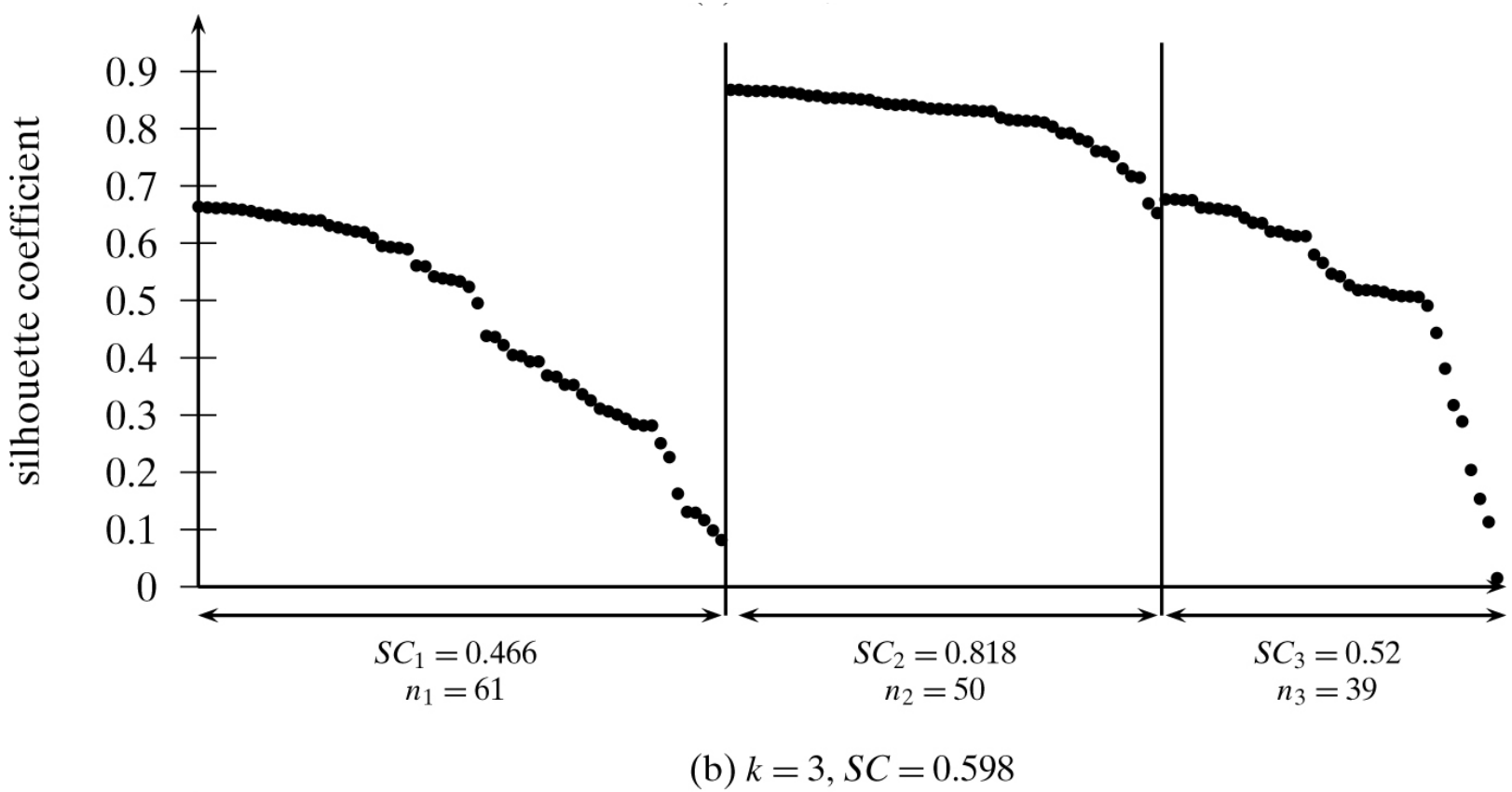
$$SC_i = \frac{1}{n_i} \sum_{\mathbf{x}_j \in C_i} s_j$$

- Pick the value of k that yields the best clustering, with many points having high s_j values within each cluster, as well as high values for SC and SC_i ($1 \leq i \leq k$).



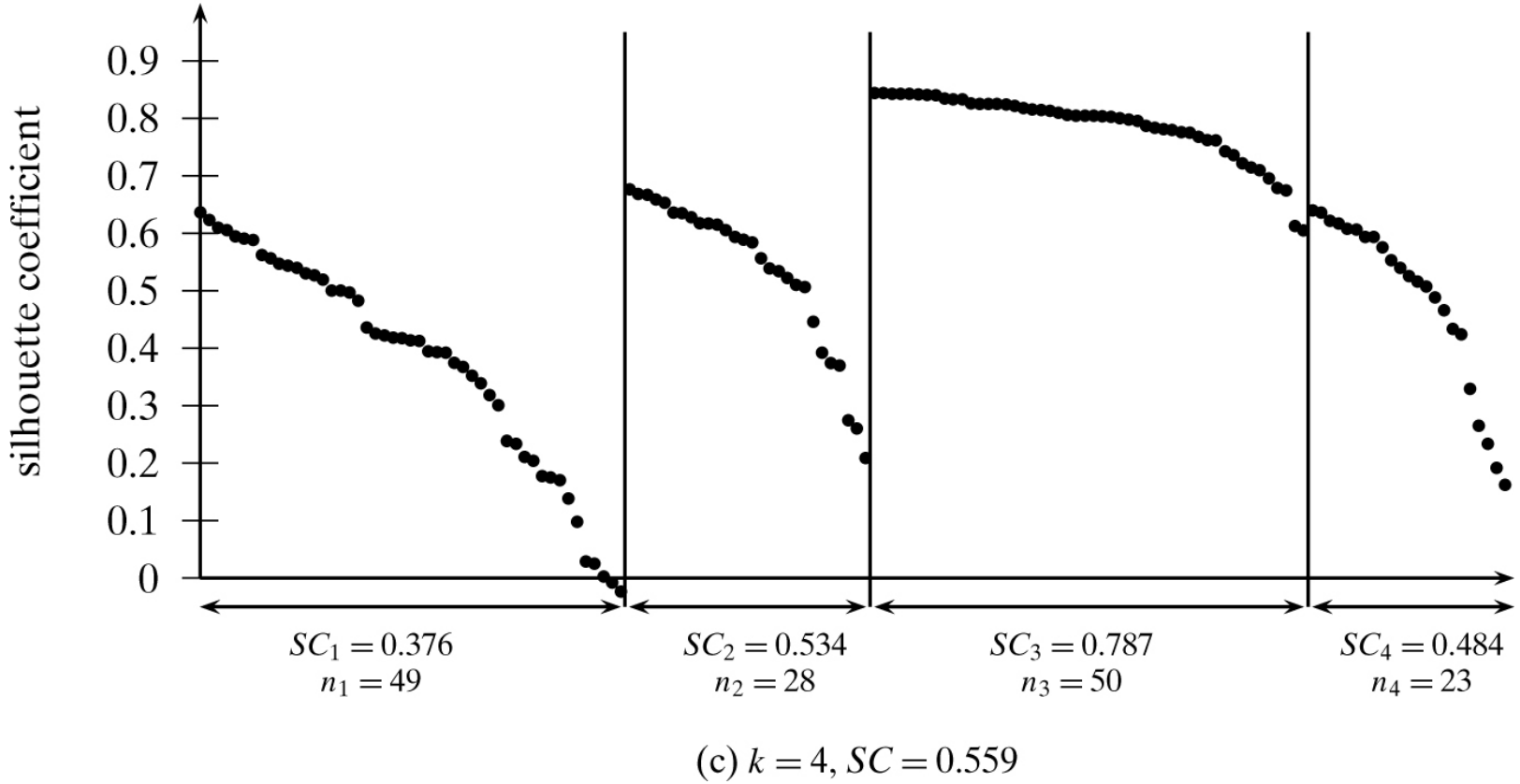
(a) $k = 2, SC = 0.706$

Silhouette coefficients for the Iris dataset computed using a k-means algorithm with k=2



(b) $k = 3, SC = 0.598$

Silhouette coefficients for the Iris dataset computed using a k-means algorithm with k=2



(c) $k = 4, SC = 0.559$

Silhouette coefficients for the Iris dataset computed using a k-means algorithm with k=4

Cluster Stability

the clusterings obtained from several datasets sampled from the same distribution should be similar or "stable."

ALGORITHM 17.1. Clustering Stability Algorithm for Choosing k

CLUSTERINGSTABILITY $(A, t, k^{\max}, \mathbf{D})$:

$n \leftarrow |\mathbf{D}|$

// Generate t samples

for $i = 1, 2, \ldots, t$ **do**

  $\mathbf{D}_i \leftarrow$ sample n points from $\mathbf{D}$ with replacement

// Generate clusterings for different values of k

for $i = 1, 2, \ldots, t$ **do**

  for $k = 2, 3, \ldots, k^{\max}$ **do**

    $\mathcal{C}_k(\mathbf{D}_i) \leftarrow$ cluster $\mathbf{D}_i$ into k clusters using algorithm A

// Compute mean difference between clusterings for each k

foreach *pair* $\mathbf{D}_i, \mathbf{D}_j$ *with* $j > i$ **do**

  $\mathbf{D}_{ij} \leftarrow \mathbf{D}_i \cap \mathbf{D}_j$ // create common dataset using Eq. (17.30)

  for $k = 2, 3, \ldots, k^{\max}$ **do**

    $d_{ij}(k) \leftarrow d\big(\mathcal{C}_k(\mathbf{D}_i), \mathcal{C}_k(\mathbf{D}_j), \mathbf{D}_{ij}\big)$ // distance between clusterings

for $k = 2, 3, \ldots, k^{\max}$ **do**

  $\mu_d(k) \leftarrow \frac{2}{t(t-1)} \sum_{i=1}^{t} \sum_{j>i} d_{ij}(k)$ // expected pairwise distance

// Choose best k

$k^* \leftarrow \arg\min_k \{\mu_d(k)\}$

Algorithm to choose k as the number of clusters that exhibits the least deviation between the clusterings. From Zaki's textbook © Cambridge University Press 2014

Clustering Tendency

Clustering Tendency

- Aims to determine whether the dataset has any meaningful groups to begin with
- Difficult task typically tackled by comparing the data distribution with samples randomly generated from the same data space
- Existing approaches include,
  - Spatial Histogram
  - Distance Distribution
  - Hopkins Statistic
  - …

Run the python notebooks and the KNIME workflows for this lecture