Clustering: Anti-Clustering, Impossibility Theorem, and Evaluation Methods

Class Notes

10th–13th March 2025

1 Introduction

Clustering is a fundamental technique in data analysis, aiming to group similar objects into clusters. These notes explore advanced topics in clustering, including ***anti-clustering***, Jon Kleinberg's ***impossibility theorem***, and ***evaluation methods***. They are designed for clarity and depth, with mathematical rigor and practical insights, making them valuable for final exams.

2 Anti-Clustering

Anti-clustering reverses the goal of traditional clustering. Instead of minimizing variance within clusters and maximizing it between clusters, anti-clustering seeks to *maximize* intra-cluster variance and *minimize* inter-cluster variance.

2.1 Mathematical Formulation

Consider a dataset of points $\{x_1, x_2, \ldots, x_n\}$ with overall mean μ. The total variance is:

$$\sigma = \sum_x \|x - \mu\|^2$$

In clustering, we partition the data into clusters $c_1, c_2, \ldots, c_k$, each with mean u_i. The variance can be expressed over clusters:

Total variance measures dispersion from the overall mean.

$$\sigma = \sum_{c_i} \sum_{x \in c_i} \|x - \mu\|^2$$

Decompose $x - \mu$ as:

$$x - \mu = (x - u_i) + (u_i - \mu)$$

Define: - $a = x - u_i$ (distance from a point to its cluster mean), - $b = u_i - \mu$ (distance from the cluster mean to the overall mean).

Then:

$$\|x - \mu\|^2 = \|a + b\|^2 = a^2 + 2a \cdot b + b^2$$

Summing over all points in cluster c_i:

$$\sum_{x \in c_i} \|x - \mu\|^2 = \sum_{x \in c_i} \|x - u_i\|^2 + 2 \sum_{x \in c_i} (x - u_i) \cdot (u_i - \mu) + \sum_{x \in c_i} \|u_i - \mu\|^2$$

- $a^2 = \|x - u_i\|^2$: The ***Sum of Squared Errors (SSE)*** for cluster c_i, i.e., $\text{SSE}_{c_i} = \sum_{x \in c_i} \|x - u_i\|^2$. - $2a \cdot b$: This term sums to zero because:

$$\sum_{x \in c_i} (x - u_i) = 0 \quad \text{(since } u_i \text{ is the mean of } c_i\text{)},$$

$$\Rightarrow 2(u_i - \mu) \cdot \sum_{x \in c_i} (x - u_i) = 2(u_i - \mu) \cdot 0 = 0.$$

- $b^2 = \|u_i - \mu\|^2$: Constant for all points in c_i, so $\sum_{x \in c_i} \|u_i - \mu\|^2 = |c_i| \cdot \|u_i - \mu\|^2$, where $|c_i|$ is the size of cluster c_i.

Thus, the total variance is:

$$\sigma = \sum_{c_i} \left(\text{SSE}_{c_i} + |c_i| \cdot \|u_i - \mu\|^2 \right)$$

In ***anti-clustering***: - Maximize SSE_{c_i} (intra-cluster variance), - Minimize $|c_i| \cdot \|u_i - \mu\|^2$ (inter-cluster variance).

2.2 Updating Clusters

When updating clusters incrementally (e.g., moving a point from one cluster to another), recomputing u_i and $x - u_i$ for all points is inefficient. Instead:

- Subtract the removed point's value from the cluster sum and adjust the mean. - Add the new point's value to the target cluster's sum and update its mean.

This allows efficient computation of $x - u_i$ for each point affected by the update.[1]

3 Jon Kleinberg's Impossibility Theorem

Jon Kleinberg's impossibility theorem highlights a fundamental limitation in clustering: no single algorithm can satisfy three intuitive properties simultaneously.

3.1 Basic Clustering Methods

Clustering can be performed in several ways:

1.**K-Clustering via Minimum Spanning Tree (MST):** - Construct an MST of the data points based on pairwise distances. - Remove the $k - 1$ longest edges to form k clusters.

2.**R-Distance Clustering:** - Set a threshold distance r. - Connect points with distances $< r$, forming clusters as connected components. In an MST, this corresponds to edges $< r$.

3.**P* Distance Clustering:** - Define a threshold on the maximum distance between any two points in a cluster (cluster diameter).

3.2 Properties of Clustering

Kleinberg defined three desirable properties:

1.**Scale Invariance:** Clustering remains unchanged if all distances are scaled by a constant $\alpha > 0$ (e.g., converting meters to kilometers).

2.**Richness:** The algorithm can produce any possible partition of the data by adjusting its parameters.

3.**Consistency:** If intra-cluster distances decrease and inter-cluster distances increase, the clustering remains the same.

[1]This method is particularly useful in iterative algorithms like k-means variants adapted for anti-clustering.

3.3 The Impossibility Theorem

Theorem 1. Impossibility Theorem: No clustering algorithm can simultaneously satisfy scale invariance, richness, and consistency.

3.4 Properties Satisfied by Basic Methods

Method	Scale Invariance	Richness	Consistency
K-Clustering (MST)	Yes	No	Yes
R-Distance	No	Yes	Yes
P* Distance	Yes	Yes	No

Table 1: Properties satisfied by basic clustering algorithms.

-**K-Clustering:** Scale-invariant (edge rankings in MST are preserved), consistent (distance changes don't alter the MST structure), but not rich (limited to k clusters). -**R-Distance:** Rich (any partition possible by tuning r), consistent, but not scale-invariant (fixed r breaks under scaling). -**P* Distance:** Scale-invariant, rich, but not consistent (reducing intra-cluster distances may split clusters).

3.5 Proof by Contradiction

Assume a clustering algorithm A satisfies all three properties. Consider a dataset with points $\{p_1, p_2, p_3\}$ and distances $d(p_1, p_2) = 1$, $d(p_2, p_3) = 1$, $d(p_1, p_3) = 2$.

- By richness, A can produce partition $\{\{p_1, p_2\}, \{p_3\}\}$. - Scale distances by $\alpha = 2$: $d(p_1, p_2) = 2$, $d(p_2, p_3) = 2$, $d(p_1, p_3) = 4$. By scale invariance, the partition remains $\{\{p_1, p_2\}, \{p_3\}\}$. - Now adjust distances: $d'(p_1, p_2) = 1$ (decreased), $d'(p_2, p_3) = 3$ (increased), $d'(p_1, p_3) = 4$. By consistency, the partition should still be $\{\{p_1, p_2\}, \{p_3\}\}$. - But by richness, A can also produce $\{\{p_1\}, \{p_2, p_3\}\}$ for some parameter. Scale invariance and consistency force A to maintain one partition, contradicting richness's flexibility.

Thus, no such A exists.[2]

[2]See Kleinberg's 2002 paper for the full proof.

4 Evaluation Methods

Evaluating clustering quality is essential. Methods are divided into external (using true labels) and internal (using data alone), with techniques to combine multiple algorithms.

4.1 External Evaluation Methods

Assuming true labels are known, we compare clusters to ground truth.

4.1.1 Confusion Matrix Terms

For pairs of points: -**TP (True Positive):** Same cluster in both true labels and result. -**FP (False Positive):** Same cluster in result, different in true labels. -**TN (True Negative):** Different clusters in both. -**FN (False Negative):** Different in result, same in true labels.

4.1.2 Rand Index

$$\text{Rand Index} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{TN} + \text{FN}}$$

Measures overall agreement but includes TN, which may inflate scores in sparse clustering.

4.1.3 Jaccard Coefficient

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

Excludes TN, focusing on positive agreements, addressing Rand's bias.

4.1.4 Precision and Recall

-**Precision:** $\frac{\text{TP}}{\text{TP+FP}}$ (accuracy of positive predictions). -**Recall:** $\frac{\text{TP}}{\text{TP+FN}}$ (coverage of true positives).

These can be exploited separately (e.g., high precision, low recall), so combined metrics are preferred.

4.1.5 F1 Score

$$\text{F1} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

Harmonic mean, balancing precision and recall.

4.1.6 Fowlkes-Mallows Index

$$\text{FM} = \sqrt{\text{Precision} \cdot \text{Recall}}$$

Geometric mean, sensitive to both metrics.

No method is universally best; choose based on context.

4.2 Internal Evaluation Methods

No true labels are needed.

4.2.1 Average Intra-Cluster Distance

$$\text{Avg Intra} = \frac{1}{|c_i|} \sum_{x,y \in c_i} d(x, y)$$

Lower values indicate tighter clusters.

4.2.2 Average Distance to Nearest Cluster

$$\text{Avg Nearest} = \frac{1}{|c_i|} \sum_{x \in c_i} \min_{y \in c_j, j \neq i} d(x, y)$$

Higher values indicate better separation.

4.2.3 Dunn Index

$$\text{Dunn} = \frac{\min_{i \neq j} d(c_i, c_j)}{\max_l \text{diam}(c_l)}$$

- $d(c_i, c_j)$: Minimum distance between clusters c_i and c_j. - $\text{diam}(c_l)$: Maximum distance within cluster c_l. Higher values signify compact, well-separated clusters.

4.3 Combining Multiple Clustering Algorithms

Run multiple algorithms, creating a consensus: - Each algorithm produces a binary matrix (1 if points are clustered together, 0 otherwise). - Average these matrices to form a consensus matrix. - Apply a final clustering (e.g., hierarchical clustering) to the consensus matrix.