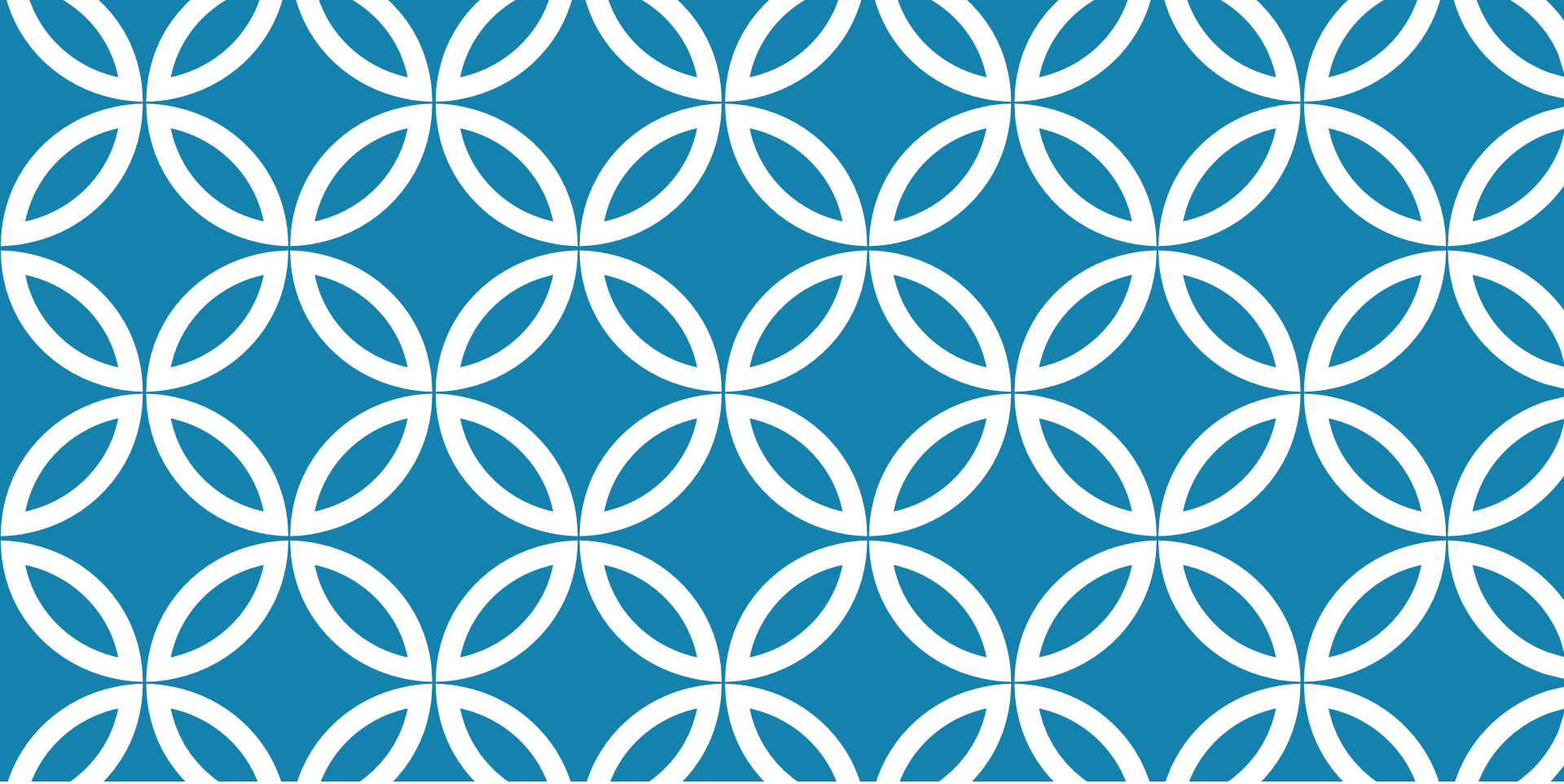

CLUSTERING

Mohit Raj Aryal

OVERVIEW

- Clustering is an unsupervised learning technique that groups similar data points together.
- It helps identify underlying patterns or structures in data.
- Common applications: customer segmentation, anomaly detection, image segmentation.
- Two popular algorithms: K-Means and DBSCAN.

K-MEANS CLUSTERING

- K-Means partitions data into K clusters based on feature similarity.
- Objective: minimize within-cluster variance.
- Works well for spherical clusters and large datasets.
- Sensitive to outliers and requires K to be predefined.

K-MEANS ALGORITHM STEPS

1. Choose number of clusters (K).
2. Initialize K centroids randomly.
3. Assign each point to the nearest centroid.
4. Recalculate centroids as the mean of points in each cluster.
5. Repeat steps 3–4 until convergence (centroids stabilize).

DBSCAN CLUSTERING

- DBSCAN: Density-Based Spatial Clustering of Applications with Noise.
- Groups points that are closely packed together (dense regions).
- Can identify outliers as noise.
- Does not require number of clusters (K).

DBSCAN ALGORITHM STEPS

1. Choose parameters ε (epsilon: radius) and MinPts (minimum points).

2. For each point, find neighbors within ε.

3. If the number of neighbors ≥ MinPts, mark as a core point.

4. Expand clusters from core points by connecting density-reachable points.

5. Label remaining points as noise.

K-MEANS VS DBSCAN

- K-Means
  - Requires predefined number of clusters.
  - Sensitive to noise and outliers.
  - Works best for spherical clusters.
  - Fast and efficient for large datasets.

- DBSCAN
  - No need to specify number of clusters.
  - Can find clusters of arbitrary shapes.
  - Handles noise/outliers effectively.
  - May struggle with varying densities.

SUMMARY

- Clustering helps discover structure in unlabeled data.
- K-Means is simple and efficient for well-separated data.
- DBSCAN excels in detecting irregular shapes and outliers.
- Algorithm choice depends on data distribution and clustering goals.