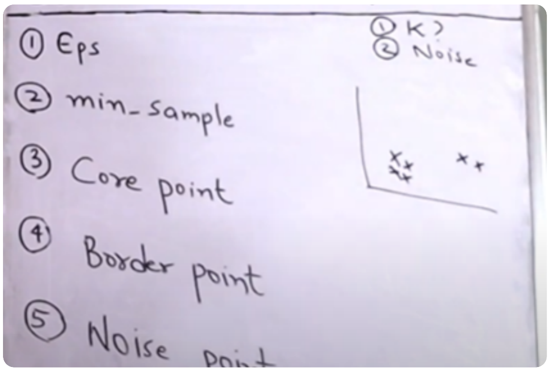

① Eps

② min_sample

③ Core point

④ Border point

⑤ Noise Point

① K?

② Noise

EXPLAIN Nicely DBSCAN CLUSTERING, including these topics with an Real time example ? Snapshot Attached...IN WHICH field this type of Clustering is used?

DBSCAN Clustering: Explanation with Real-Time Example

DBSCAN (Density-Based Spatial Clustering of Applications with Noise) is a popular clustering algorithm used to identify groups of closely packed points in a dataset, where the number of clusters is not predefined. Unlike other clustering algorithms (like K-means), DBSCAN doesn't require you to specify the number of clusters beforehand. Instead, it uses density to determine clusters and can identify noise points.

Here are the key concepts behind **DBSCAN**:

1. **Eps (Epsilon)**

- **Eps** is the maximum distance between two points for them to be considered neighbors. It defines the radius of a neighborhood around a point.
- In other words, if the distance between two points is less than or equal to **Eps**, they are considered to be part of the same neighborhood.

Example: If **Eps = 0.5**, then any points within a distance of 0.5 units from a point are considered neighbors.

2. **min_samples**

- **min_samples** is the minimum number of points required to form a dense region (i.e., a cluster).
- A point is considered **core** if it has at least **min_samples** points within its **Eps** neighborhood (including the point itself).

Example: If **min_samples = 4**, then a point needs to have at least 4 neighboring points within its **Eps** distance to be considered a **core point**.

3. **Core Point**

- A **core point** is a point that has at least **min_samples** within a distance of **Eps**. These points form the foundation of a cluster.
- Core points are at the center of dense regions.

Example: If you have a group of points closely packed together, the points with more than 4 neighbors within a 0.5 radius would be **core points**.

4. **Border Point**

- A **border point** is a point that has fewer than **min_samples** neighbors within **Eps** distance but is still in the neighborhood of a **core point**.
- These points are part of a cluster but do not have enough density to be considered **core points**.

Example: A point that has only 3 neighbors but is within **Eps** distance from a **core point** would be a **border point**.

5. **Noise Point**

- A **noise point** (also known as an **outlier**) is a point that does not meet the criteria to be a **core point** or a **border point**. These points lie outside the dense regions and do not belong to any cluster.

Example: If there are points that do not have enough neighboring points (less than **min_samples**) and are not close enough to any **core points**, they are considered noise points.

Real-Time Example of DBSCAN in Customer Segmentation

Imagine you have data from an online store and you want to segment customers based on their **annual spending** and **annual visits** to the website.

Dataset:

Customer ID	Annual Spending (k)	Annual Visits
1	5	50
2	6	55
3	6	53
4	15	150
5	16	160
6	2	20
7	30	300

Applying DBSCAN:

1. **Set Parameters**:
   - **Eps = 10**: If the distance between two customers' spending is less than or equal to 10k, they are considered close.
   - **min_samples = 3**: At least 3 points (including the point itself) should be within the **Eps** distance to form a cluster.
2. **Step 1: Identifying Core Points**:
   - Customer 1, Customer 2, and Customer 3 are close to each other (within **Eps = 10**), and they have at least 3 points in their neighborhood, so they form a **core** group.
3. **Step 2: Identifying Border Points**:
   - Customer 4 and Customer 5 are neighbors to Customer 1, 2, and 3 but have fewer than 3 neighbors in their own **Eps** distance. These are **border points**.
4. **Step 3: Identifying Noise Points**:
   - Customer 6 and Customer 7 do not belong to any dense region and do not have enough neighbors within **Eps**. These are **noise points**.

Resulting Clusters:

- **Cluster 1**: Customers 1, 2, and 3 (Core points).
- **Cluster 2**: Customers 4 and 5 (Border points).
- **Noise**: Customers 6 and 7 (Noise points).

Where is DBSCAN used?

DBSCAN is widely used in various fields, especially when:

- **The number of clusters is not known beforehand**.
- **Data contains noise or outliers**.
- **Clusters have irregular shapes** (unlike K-means, which assumes spherical clusters).

Applications of DBSCAN:

1. **Geospatial Data**: DBSCAN is used in geographic data analysis, for example, identifying clusters of earthquakes or analyzing spatial data for cities or infrastructure.
2. **Anomaly Detection**: It can be used in fraud detection or identifying anomalous patterns in large datasets, such as credit card transactions or network security.
3. **Image Segmentation**: DBSCAN helps to identify regions of interest in images where the pixel density forms natural clusters.
4. **Social Network Analysis**: In detecting communities or groups in social networks based on interactions or friendships.
5. **Market Segmentation**: In customer segmentation for targeting marketing strategies, as shown in the customer example.

Conclusion

DBSCAN is a powerful clustering algorithm that focuses on **density** rather than distances between points. It is particularly useful in identifying clusters of arbitrary shapes and handling noise in real-world data.