**DBSCAN: Density-Based Spatial Clustering of Applications with Noise**

* **Core Idea:** DBSCAN groups together data points that are closely packed in high-density regions, while marking points in low-density regions as outliers or noise. It doesn't require you to specify the number of clusters beforehand.
* **Key Parameters:**
  + **eps (ϵ):** Defines the maximum distance between two points for one to be considered as in the neighborhood of the other.
  + **minPts:** The minimum number of data points required to form a dense region (i.e., points within the eps radius of a point).
* **How it Works & Point Types:**
  + The algorithm picks an arbitrary unvisited point.
  + It finds all neighbor points within the eps distance.
  + If a point has at least minPts neighbors (including itself), it's marked as a **Core Point**, and a new cluster is started.
  + All reachable points from the core point (within eps distance) are added to the cluster. If any of these neighbors are also core points, their neighbors are also added recursively (density-connected).
  + If a point has fewer than minPts neighbors but is within the eps distance of a core point, it's marked as a **Border Point**. Border points belong to a cluster but aren't used to expand it further.
  + If a point is neither a core point nor a border point, it's marked as **Noise**.
  + The process continues until all points have been visited.
* **Advantages:**
  + Doesn't require specifying the number of clusters (k) in advance.
  + Can find arbitrarily shaped clusters (unlike K-Means which assumes spherical clusters).
  + Robust to outliers and can identify them as noise.
* **Disadvantages:**
  + Can be sensitive to the choice of eps and minPts parameters; tuning them can be challenging.
  + Struggles with datasets where clusters have significantly varying densities, as a single (eps, minPts) combination might not work well for all clusters.
  + Performance can degrade on high-dimensional data due to the "curse of dimensionality" affecting distance measurements.
* **Common Applications:**
  + Anomaly detection (e.g., fraud detection).
  + Spatial data analysis (e.g., identifying geographic points of interest).
  + Image segmentation.
  + Recommendation systems (grouping users with similar behavior).





