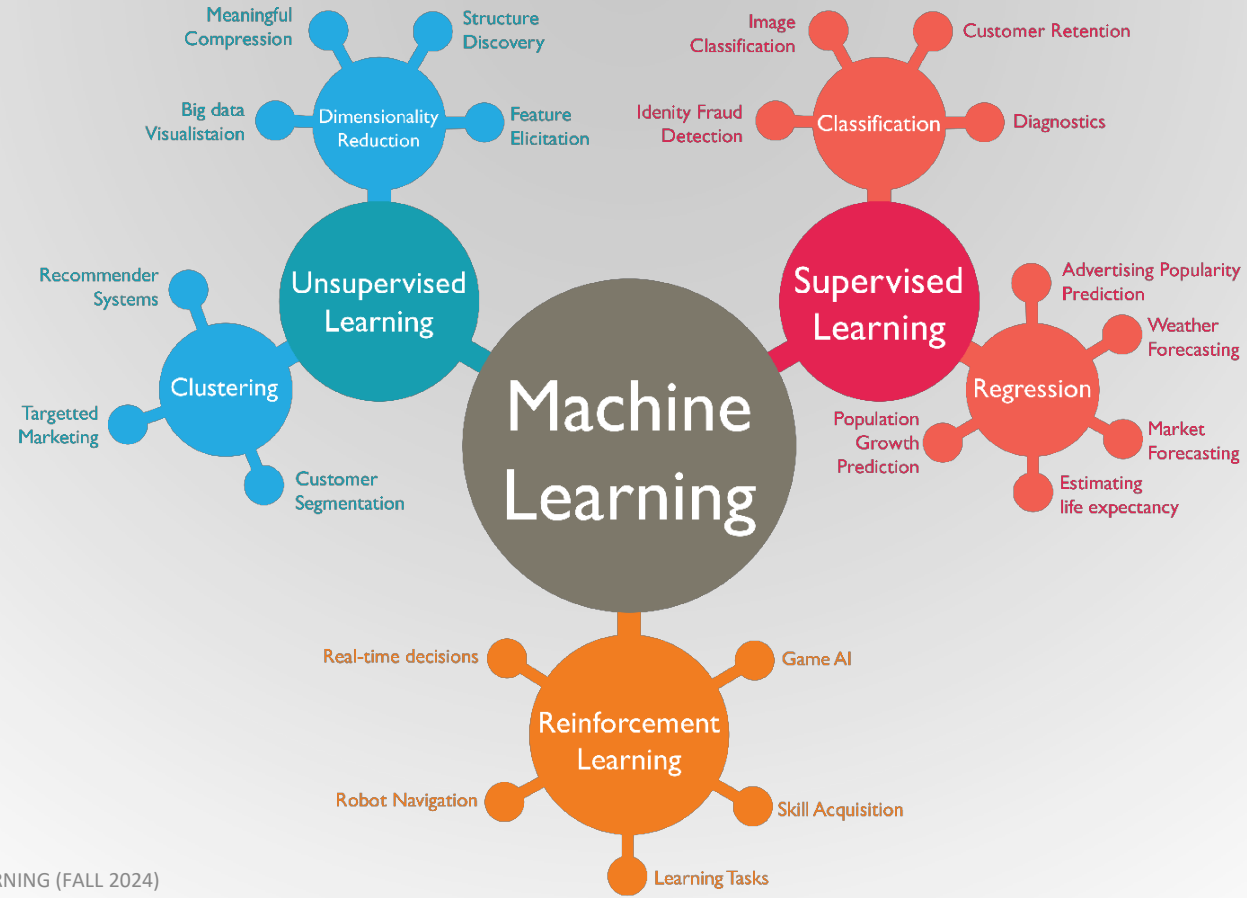
9.2 Clustering

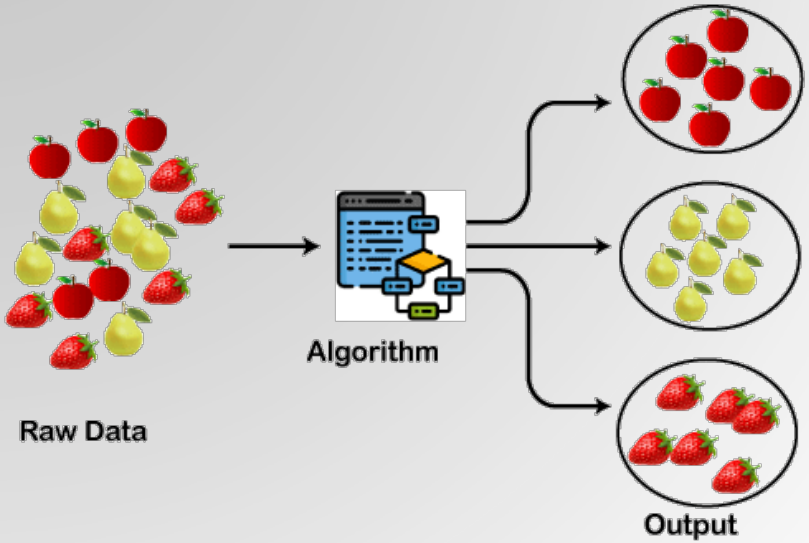
Dr. Sultan Alfarhood

Machine Learning Approaches

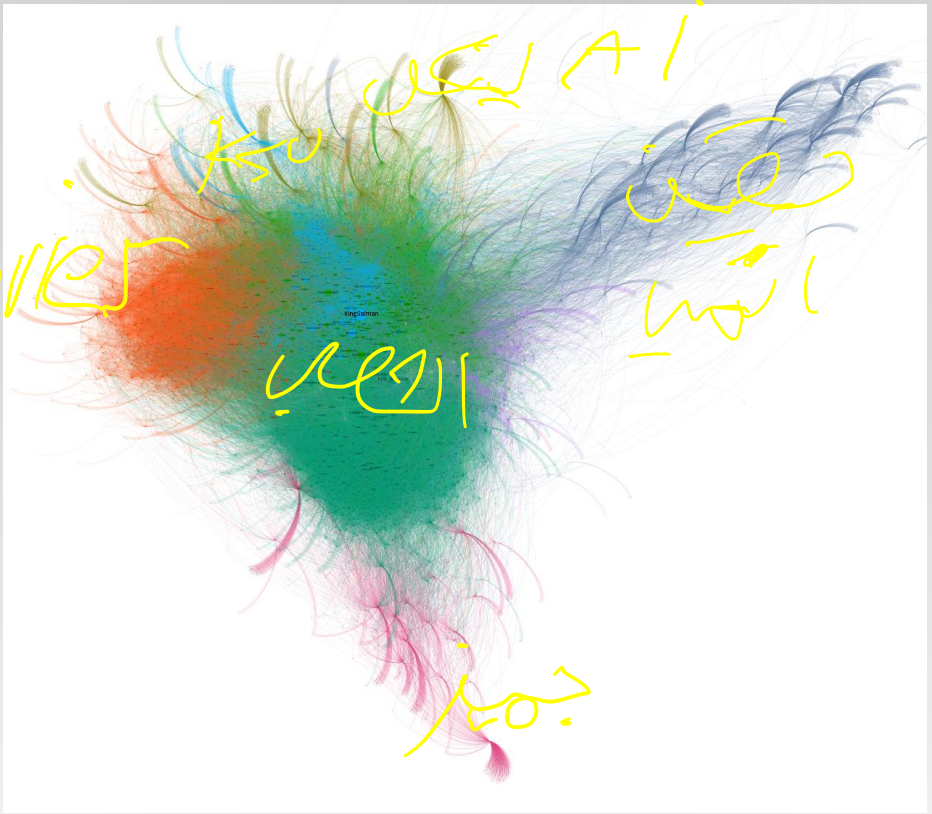


Clustering

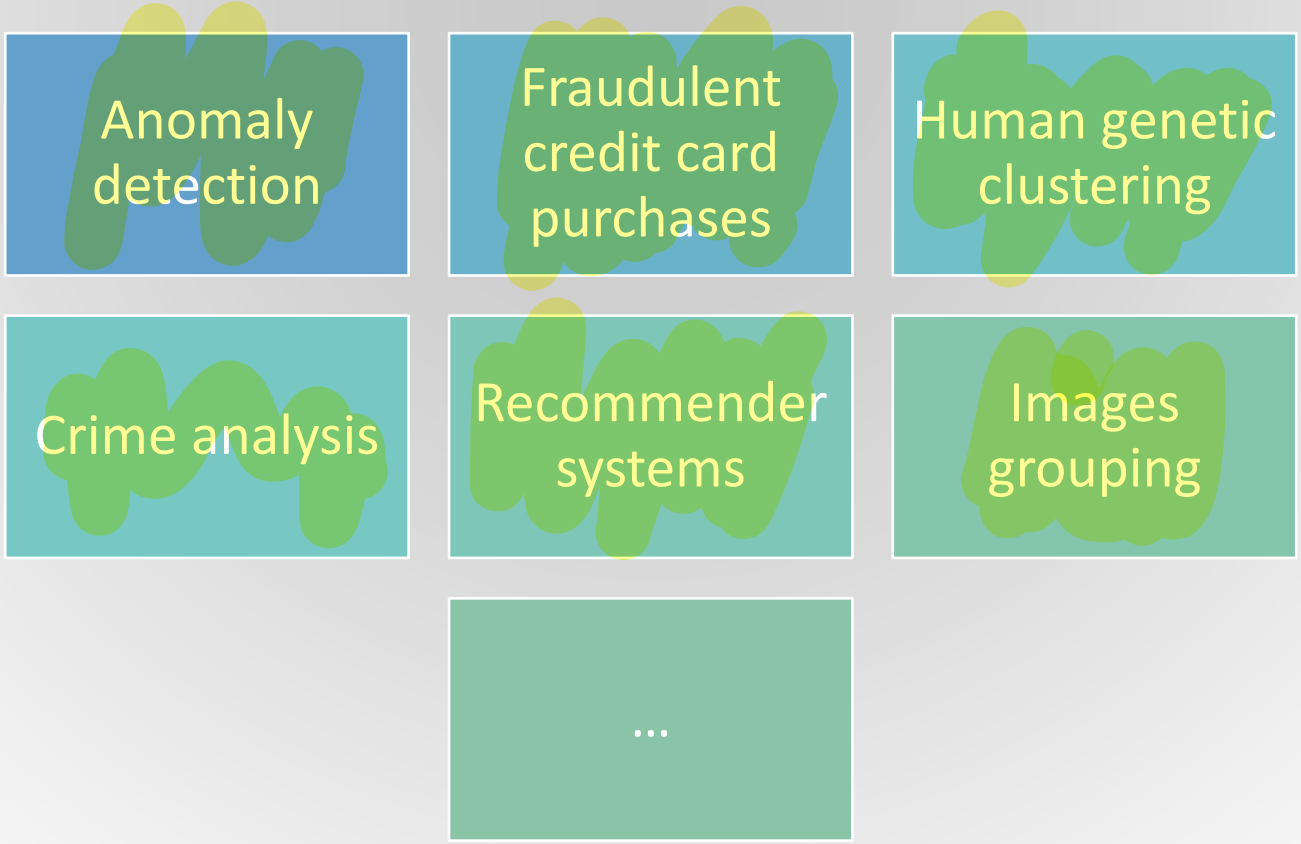
- A clustering algorithm discovers the built-in groupings in the data
  - e.g., grouping customers by purchasing behavior
- The goal of clustering is to create groups of data points
  - Points in different clusters are dissimilar
  - Points within a cluster are similar



Clustering Example of Social Media Accounts

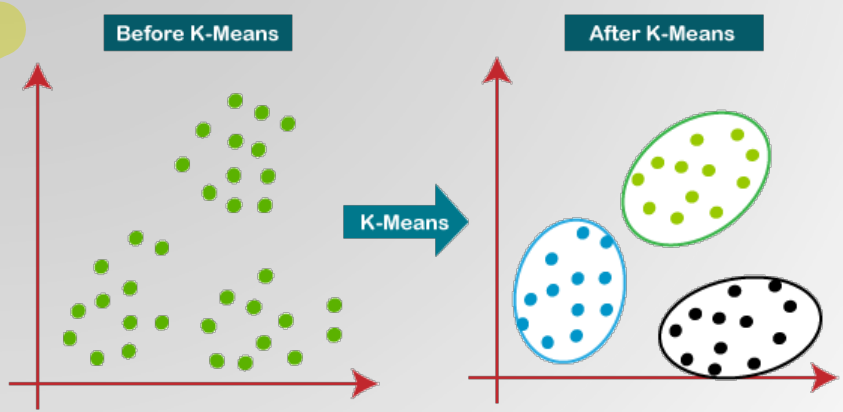


Applications



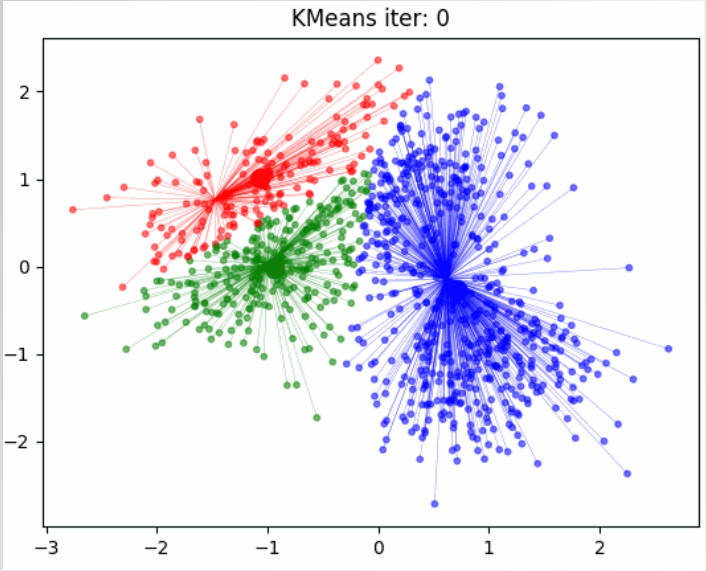
9.2.1 K-Means

- K-means is an example of a partitional clustering algorithm.
- Once the algorithm has been run and the groups are defined, any new data can be easily assigned to the existing groups.
- K-means is an extremely popular clustering algorithm, widely used in tasks like
  - Behavioral segmentation
  - Inventory categorization
  - Sorting sensor measurements
  - Detecting bots or anomalies

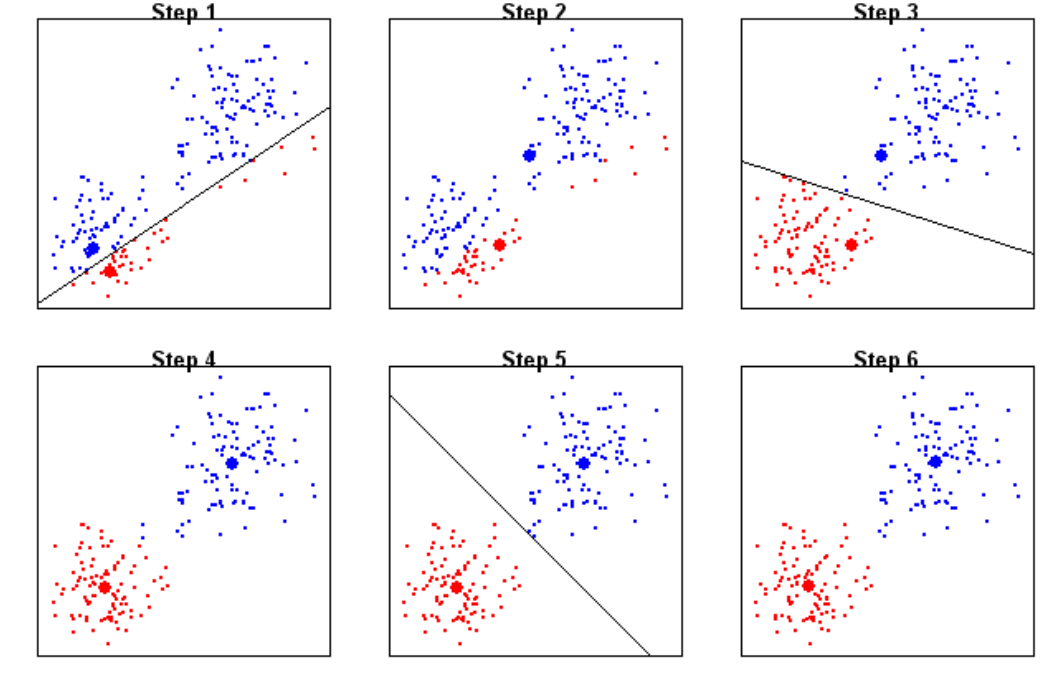


How K-Means Works?

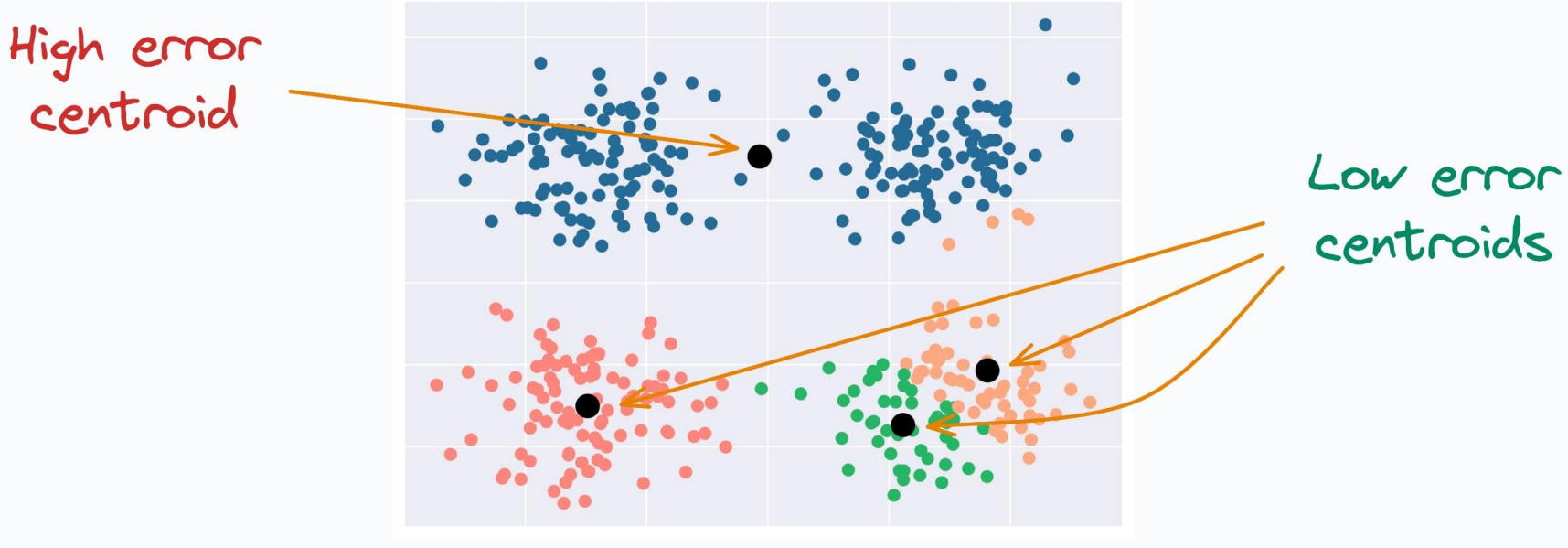
1. The algorithm randomly chooses a **centroid** for each cluster.
   - For example, if we choose a “k” of 3, the algorithm randomly picks 3 centroids.
2. K-means assigns every data point in the dataset to **the nearest centroid**,
   - Meaning that a data point is in a particular cluster if it is closer to that cluster’s centroid than any other centroid.
3. For every cluster, the algorithm **recomputes the centroid** by taking the average of all points in the cluster
   - Reducing the total intra-cluster variance in relation to the previous step.
   - Since the centroids change, the algorithm re-assigns the points to the closest centroid.
4. The algorithm repeats the calculation of centroids and assignment of points until
   - The sum of distances between the data points and their corresponding centroid is minimized
   - Or, a maximum number of iterations is reached
   - Or, no changes in centroids value are produced



K-Means Example

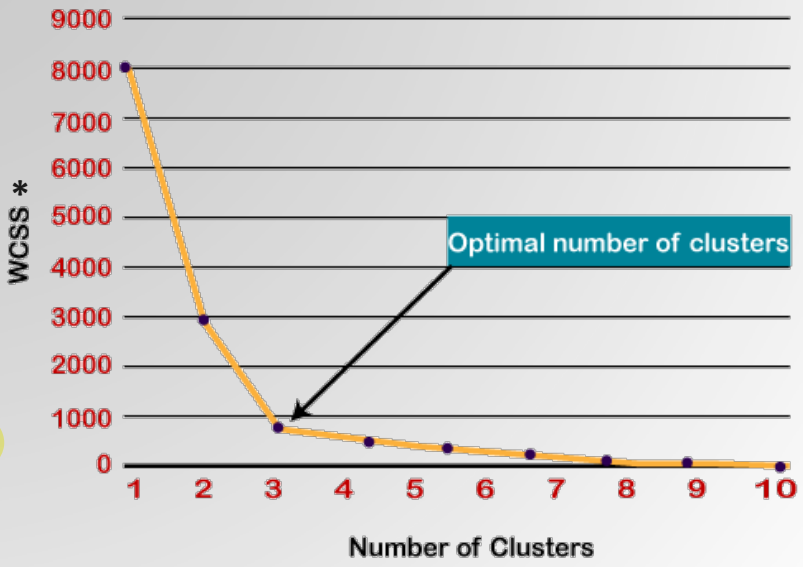


Centroids with high and low error



9.2.3 Determining the Number of Clusters

- One popular approach is testing different numbers of clusters and measuring the resulting Sum of Squared Errors
  - Choosing the "k" value at which an increase will cause a very small decrease in the error sum, while a decrease will sharply increase the error sum.
- This point that defines the optimal number of clusters is known as the **elbow method**.



* WCSS (Within-Cluster Sum of Square)

Advantages of K-Means

It is very simple to implement

It is faster to large datasets

it adapts the new examples very frequently

Generalization of clusters for different shapes and sizes

Disadvantages of K-Means

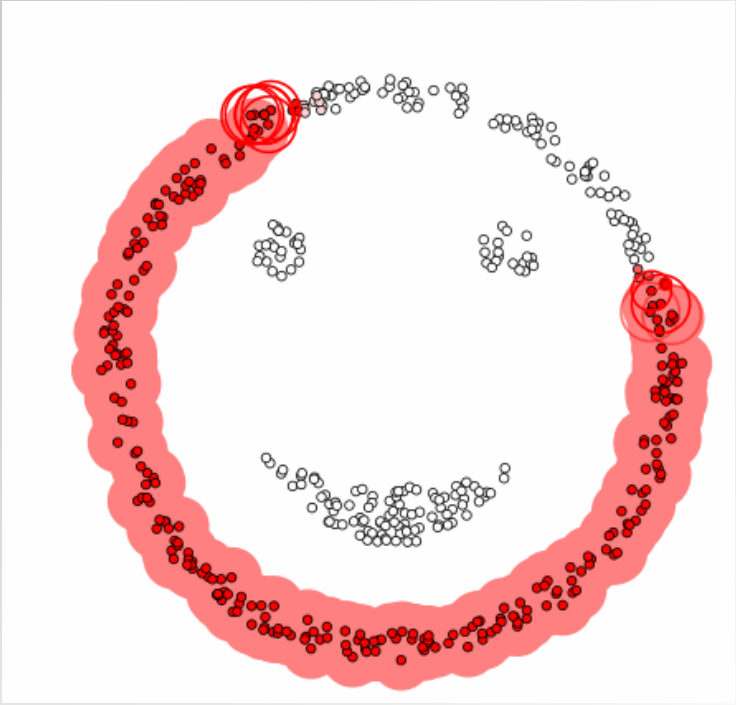
It is sensitive to the outliers

Because of distance

Choosing the k values manually is a tough job

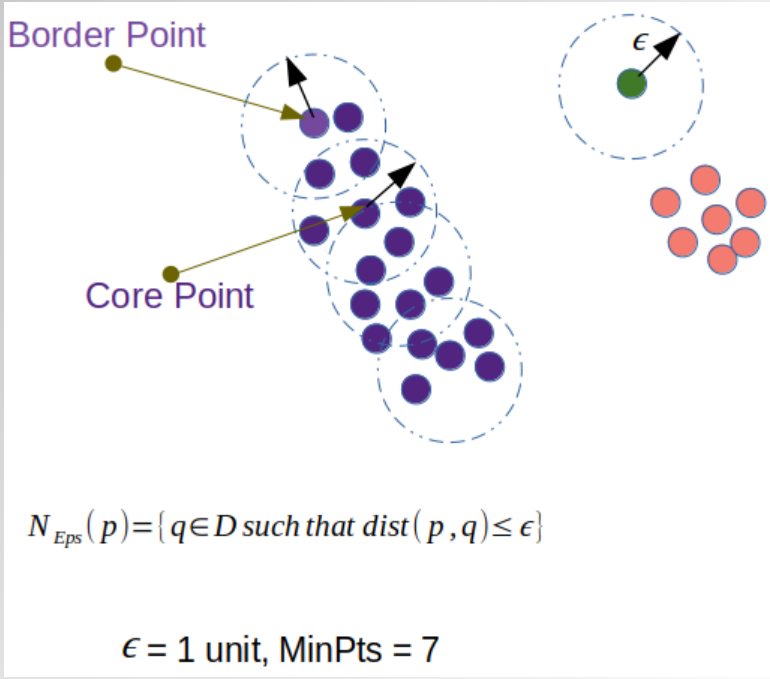
9.2.2 DBSCAN

- Density-based spatial clustering of applications with noise (DBSCAN)
- Density-based algorithms work by identifying dense regions in space (i.e., populated with many data points) separated by less dense regions.



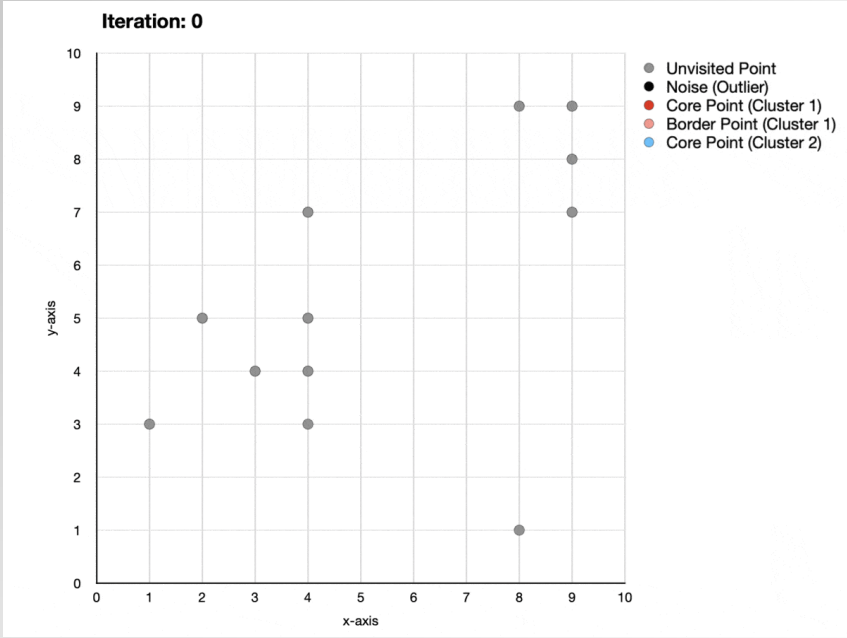
How Does DBSCAN work?

- To enable the algorithm to find these dense regions, two hyperparameters must be chosen:
  - **Epsilon ϵ** (sklearn: eps)
    - The radius of the area around the point defining the maximum distance between such point and any other points for one to be considered in the neighborhood of the other.
    - In simple terms, it is the radius of the circle you can see in the below gif image.
  - **Minimum points** (sklearn: min_samples)
    - The minimum number of points present in the neighborhood required to form a cluster.

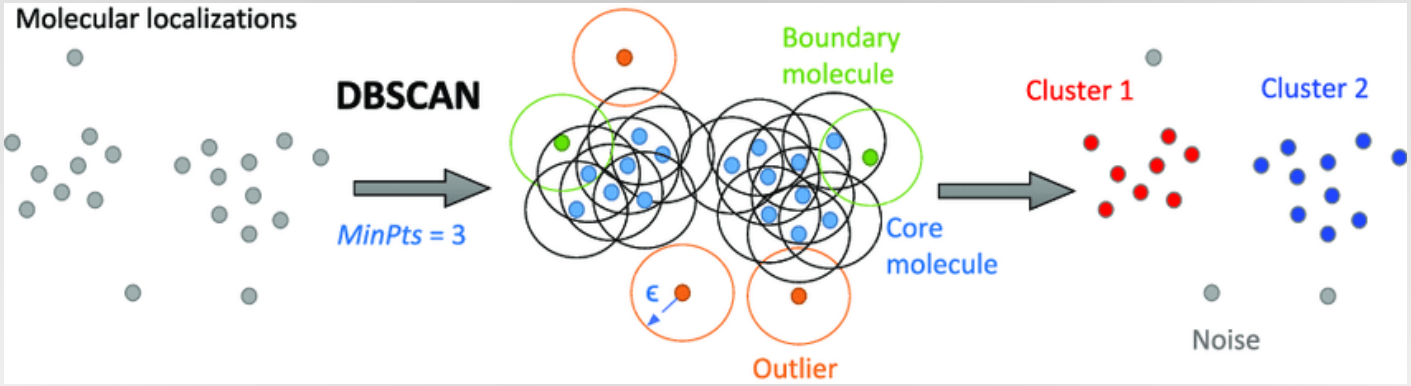


How Does DBSCAN work?

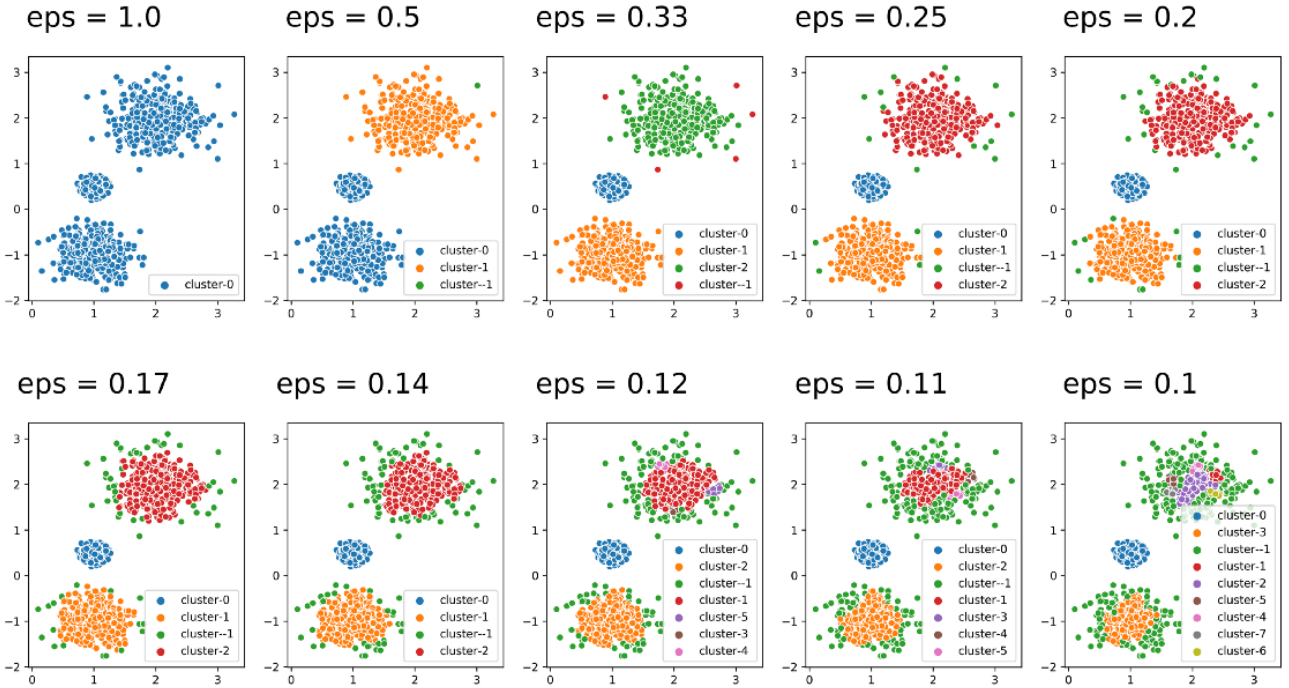
- The algorithm proceeds by arbitrarily picking up a point in the dataset (until all points have been visited).
- If there are at least ‘minPoint’ points within a radius of ‘ε’ to the point then we consider all these points to be part of the same cluster.
- The clusters are then expanded by recursively repeating the neighborhood calculation for each neighboring point.



DBSCAN Example



Example: Choosing different values for Epsilon ϵ



Advantages of DBSCAN

Works well for noisy datasets

Can identity Outliers easily

Clusters can take any irregular shape unlike K-Means where clusters are more or less spherical

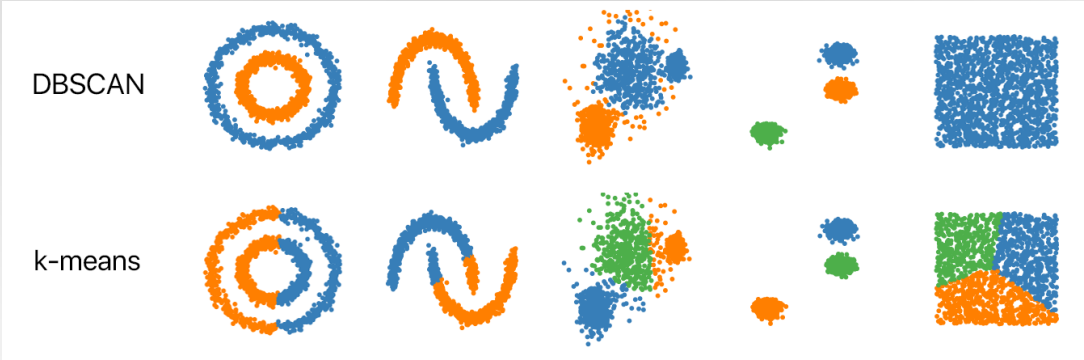
Disadvantages of DBSCAN

Does not work very well for sparse datasets or datasets with varying density

Sensitive to eps (ε) and *minPts* parameters

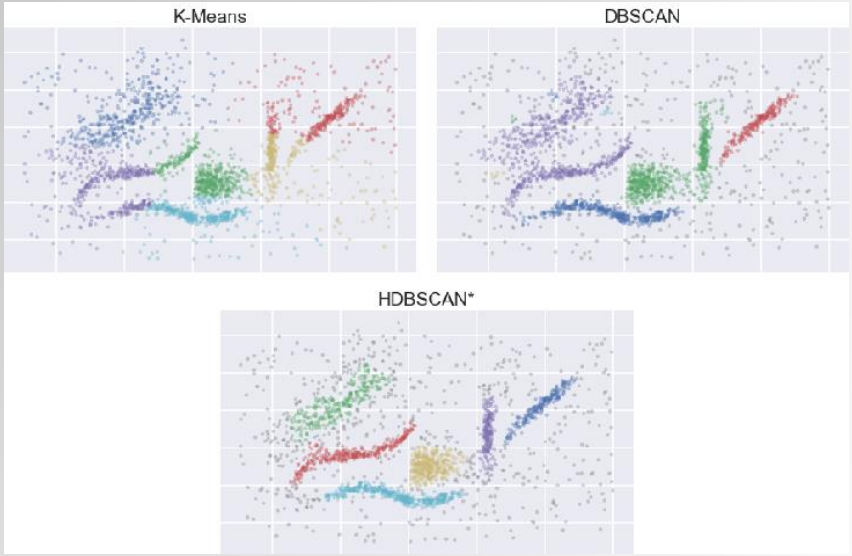
DBSCAN vs K-Means

K-Means	DBSCAN
Clusters formed are more or less **spherical or convex** in shape and must have same feature size.	Clusters formed are **arbitrary in shape** and may not have same feature size.
Sensitive to the number of clusters specified.	Number of clusters need not be specified.
More efficient for large datasets.	Can not efficiently handle high dimensional datasets.
Does not work well with outliers and noisy datasets.	Efficiently handles outliers and noisy datasets.
It requires **one parameter** : Number of clusters (K)	It requires **two parameters** : Radius (ε) and Minimum Points (M)



HDBSCAN

- HDBSCAN (Hierarchical DBSCAN) is the clustering algorithm that keeps the advantages of DBSCAN, by removing the need to decide on the value of ϵ.
- The algorithm is capable of building clusters of varying density.
- HDBSCAN only has one important hyperparameter: **n, the minimum number of examples to put in a cluster**.
  - This hyperparameter is relatively simple to choose by intuition.
- HDBSCAN has very fast implementations
  - It can deal with millions of examples effectively.



Clustering Challenges

- What does each cluster **mean**?
  - e.g., if your model indicates that the user is in the blue cluster, you'll have to determine what the blue cluster represents
- Sometimes, it hard to determine **what action** to take based on the cluster
  - You can try to assign a meaning to a cluster, but this can be tricky
    - The model might not group by criteria that people find intuitive
- One alternative approach is to **label some items before** you cluster, and then try to propagate those labels across the entire cluster
  - e.g., if all items with label X end up in one cluster, maybe you can spread label X to other examples

K-Means, DBSCAN, HDBSCAN Examples

- https://colab.research.google.com/drive/1fEZqC8NRT5Eub2trUOGC9WkaimvBFoiA?usp=sharing