K-means clustering

Prashant Shekhar

shekharp@erau.edu
Department of Mathematics
Embry-Riddle Aeronautical University

April 14, 2023

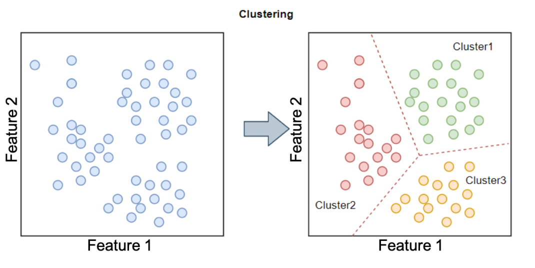
Lecture Outline

Unsupervised Learning

According to Mathworks

Unsupervised learning is a type of machine learning algorithm used to draw inferences from datasets consisting of input data without labeled responses.

The most common unsupervised learning method is **cluster analysis**, which is used for exploratory data analysis to find hidden patterns or grouping in data. The clusters are modeled using a measure of similarity which is defined upon metrics such as Euclidean or probabilistic distance.



Few applications of clustering

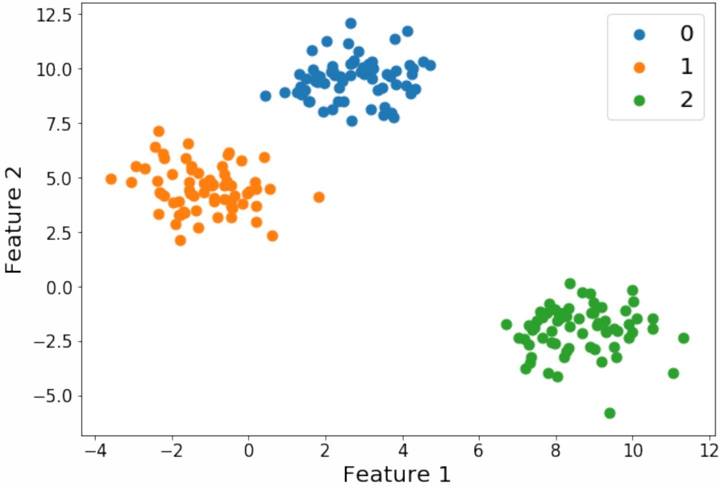
- Clustering can help marketers discover distinct groups in their customer base. And they can characterize their customer groups based on the purchasing patterns.
- Clustering also helps in classifying documents on the web for information discovery
- Netflix uses clustering to identify Viewer groups
- Clustering is also used in outlier detection applications such as detection of credit card fraud.

In this lecture, we will discuss **K-means** algorithm to come up with such clusters in unlabeled datasets.

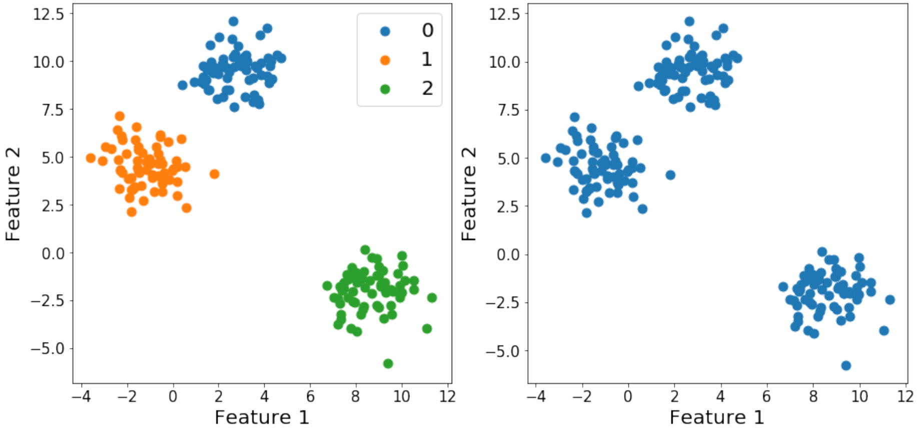
K-means clustering algorithm

1. **STEP 1:** Specify the number of clusters K
2. **STEP 2:** Initialize K centroids (either from the datapoints or some other points in the feature space). If centroids are selected from the datapoints, it should be without replacement.
3. **STEP 3:** Keep iterating until there is no change to the centroids. i.e assignment of data points to clusters isn't changing.
    1. **Step 3.1:** Compute the distance between the datapoints and all centroids.
    2. **Step 3.2:** Assign each datapoint to the closest centroid (cluster)
    3. **Step 3.3:** Compute new centroids for all clusters (averaging all the coordinates)

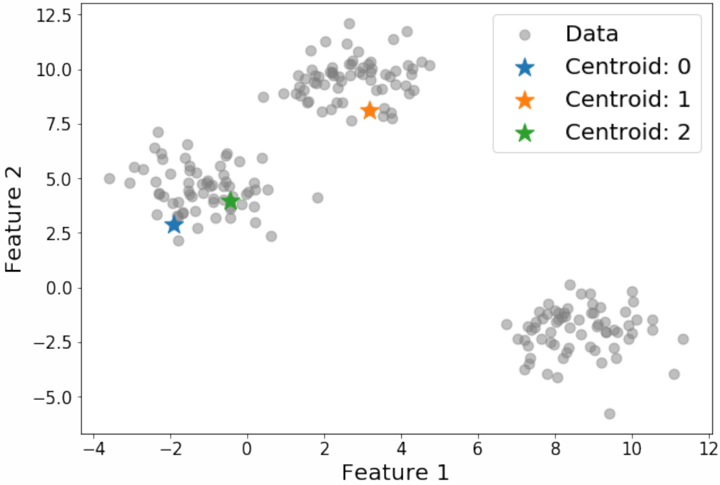
A visual example: randomly generated data



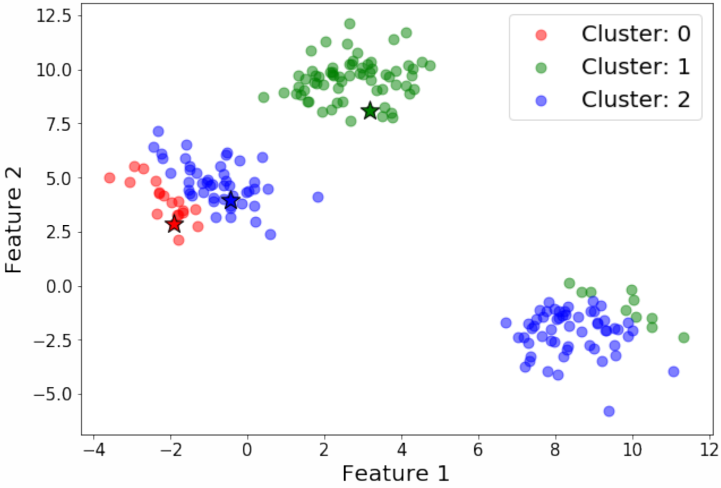
A visual example: truth vs what algorithm sees



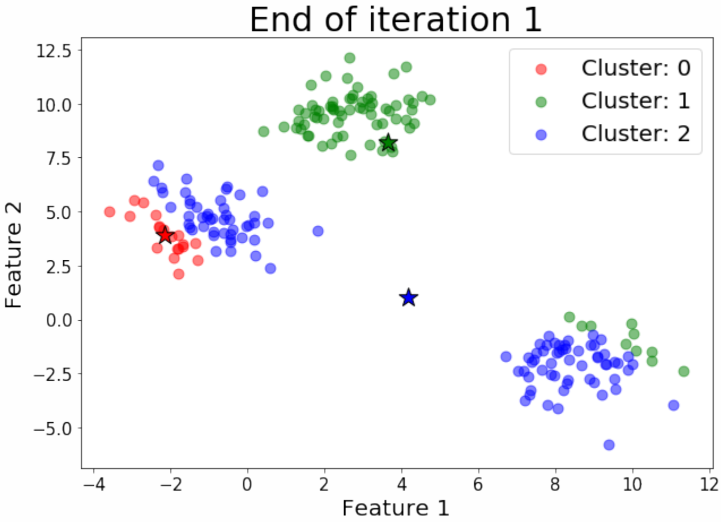
Step 1 and 2: assuming K = 3



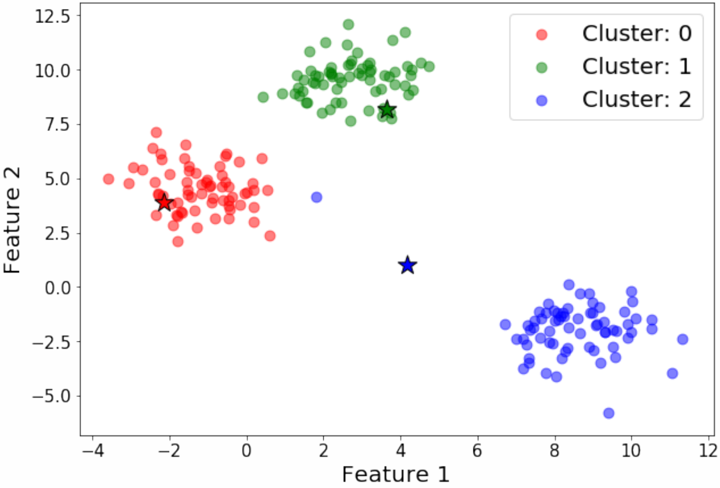
Step 3.1 and 3.2: cluster based on distance



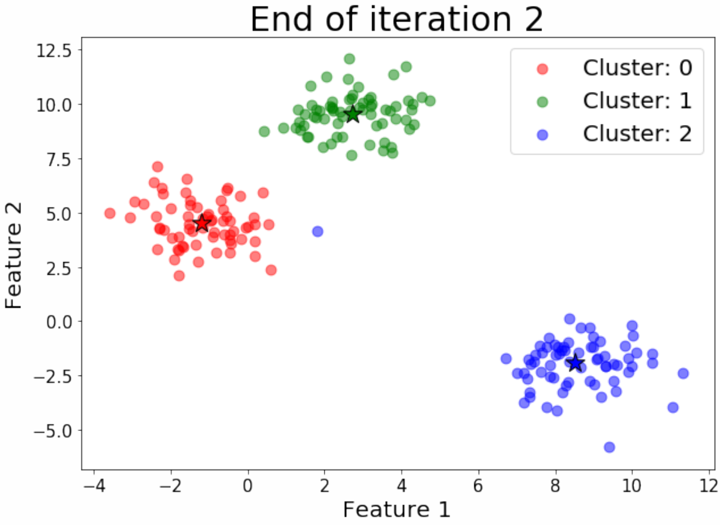
Step 3.3: recompute centroid



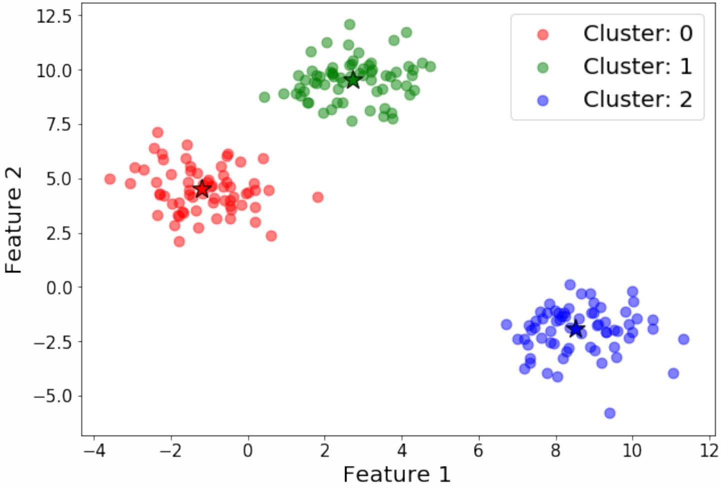
Step 3.1 and 3.2: reassign cluster



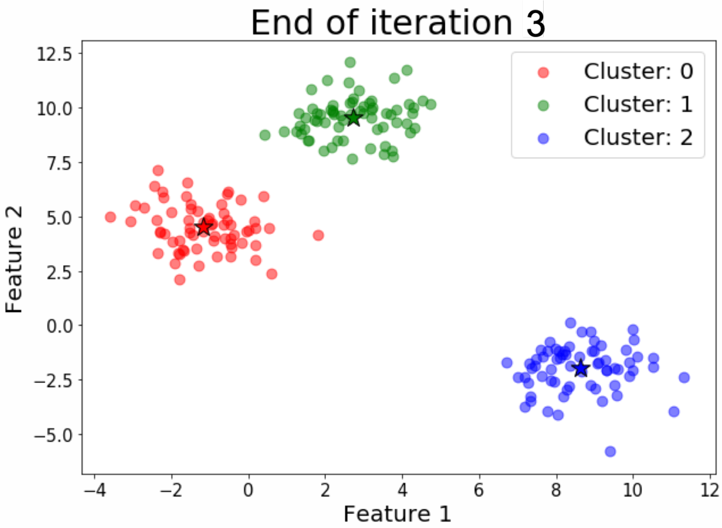
Step 3.3: recompute centroid



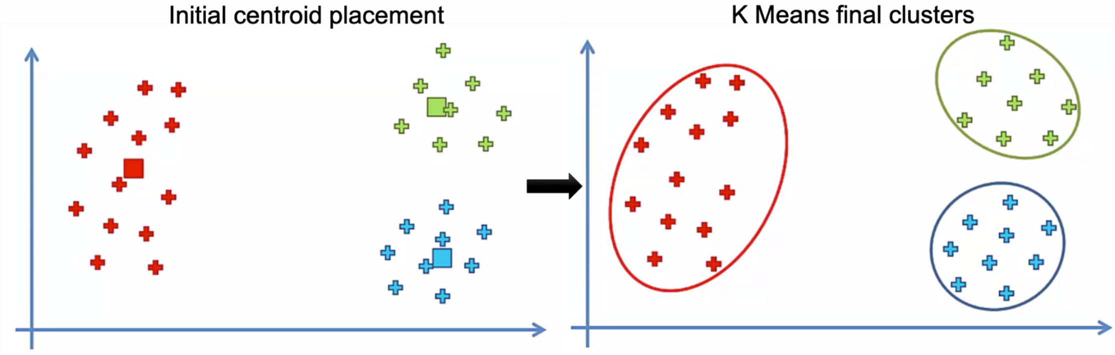
Step 3.1 and 3.2: reassign cluster



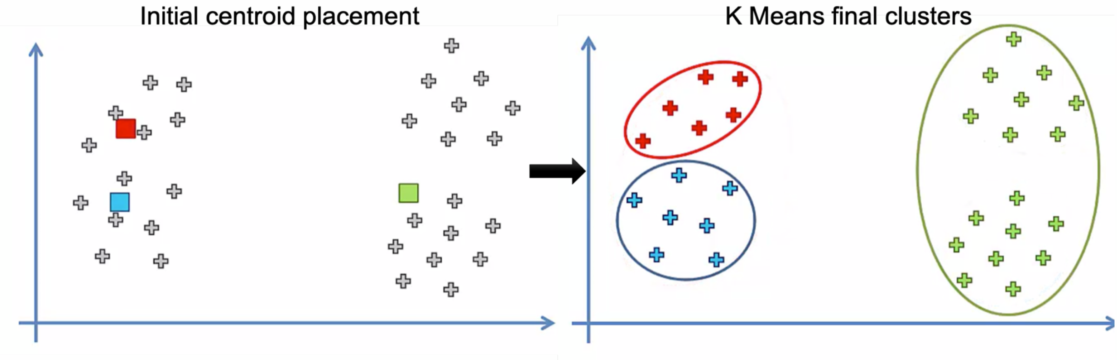
Step 3.3: recompute centroid



Problems with random initialization



Problems with random initialization



Problems with random initialization

- Hence, both number and location of initial centroids can affect the final clusters obtained by the KMeans algorithm.
- Once number of clusters are chosen, K Means++ is an additional algorithm which can help to determine the suitable initial location of the centroids
- To choose number of clusters (or centroids - K) we use a metric called Within Cluster Sum of Squares (WCSS)

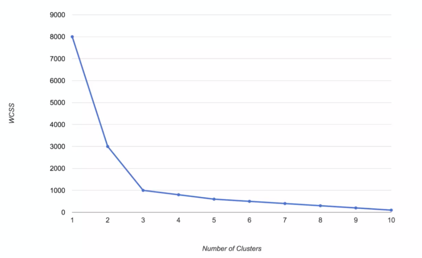
Within Cluster Sum of Squares (WCSS)

Assuming we are considering 3 clusters and $D(\cdot,\cdot)$ represents distance between two points. Then,

$$WCSS(3) = \sum_{P_i \in cluster1} D(P_i, centroid_1)^2 + \sum_{P_i \in cluster2} D(P_i, centroid_2)^2 + \sum_{P_i \in cluster3} D(P_i, centroid_3)^2$$

Elbow method for deciding number of clusters

- Compute WCSS by considering all possible number of clusters from 1 to number of data points.
- Plot the results with number of clusters on the X axis and WCSS metric on the Y-axis.
- Find the number of clusters after which the drop in WCSS is not very high (judgment call !!). Kind of like looking for an elbow in the plot below



- Please note, this is a very subjective way of choosing number of clusters. Infact, this is a very active area of research right now !!