DATA, MSML, BIOI 602 Principles of Data Science

Unsupervised Learning

[Slides from Fardina Fathmiul Alam]

We have covered so far

Supervised Machine Learning algorithms and techniques to develop models where the data had **labels previously known**.

In other words, our data had some target variables with specific values that we used to train our models.

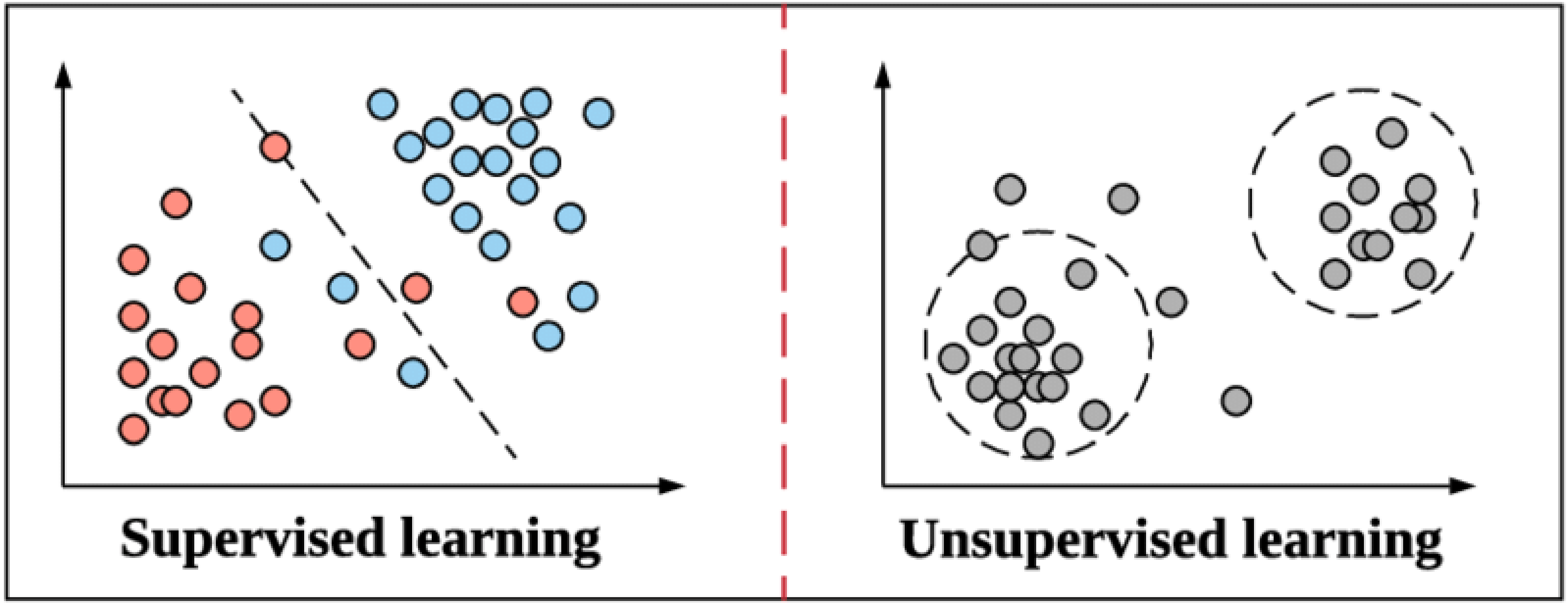
However, When Dealing with Real-World Problems

Most of the time, data **does not come** with **predefined labels**!!!

What can we do?

We can develop machine learning models that can correctly classify **unlabeled data** by autonomously **identifying commonalities** in the features. These commonalities are then used to predict classes for new data.

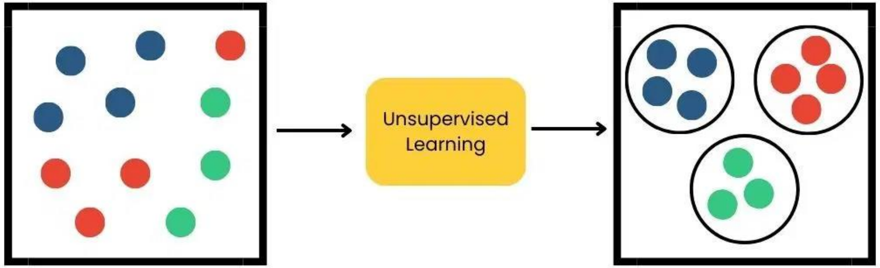
Unsupervised Learning


Supervised learning
Unsupervised learning

Definition: Unsupervised Learning

Unsupervised learning, also known as unsupervised machine learning, uses machine learning (ML) algorithms to analyze and **cluster unlabeled data sets**.

- These algorithms discover hidden patterns or data groupings without the need for human intervention. **E.g. Clustering**.



The main goal of these types of algorithms is to **study the intrinsic and hidden structure of the data** in order to *get meaningful insights*, segment the datasets in **similar groups or to simplify them**

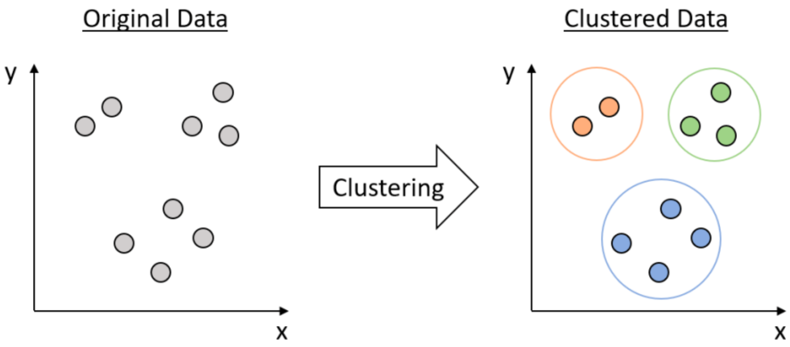
What can we Cluster in Practice?

- News articles or web pages by topic
- Protein sequences by function, or genes according to expression profile
- Users of social networks by interest
- Customers according to purchase history

Clustering

Cluster analysis is a powerful unsupervised learning technique used to identify natural groupings or clusters within datasets.

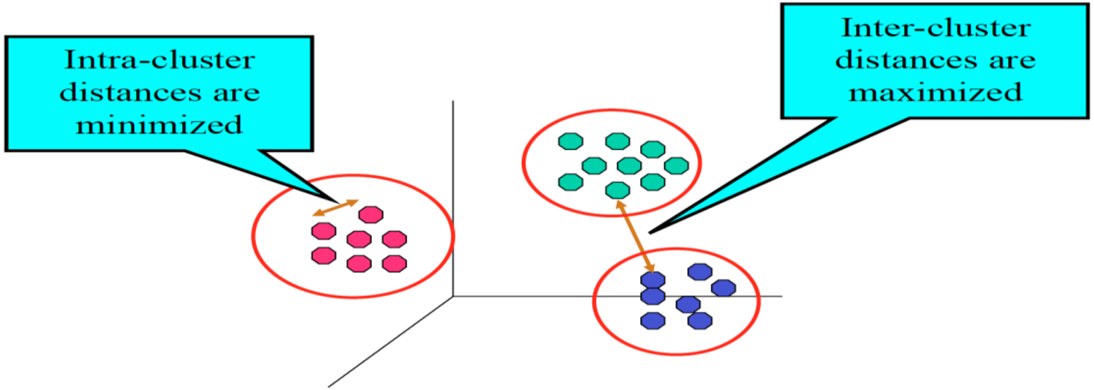
- **Group similar data points** or objects into clusters or categories.
- By grouping data points with similar characteristics, cluster analysis helps **reveal hidden patterns, trends, and relationships**.



Objectives of “Clustering”

Finding groups of objects such that the objects within the <u>same group are similar</u> or related to one another, while being different from or unrelated to the objects in <u>other groups</u>.

- ❑ **Intra-Cluster Distance are Minimized:** Group objects (observations) that are very similar or homogeneous together (in same cluster).
- ❑ **Inter-Cluster Distance are Maximized:** Observations belonging to different clusters are different or heterogeneous
- ❑ Facilitates interpretation.



Most Common Clustering Algorithms

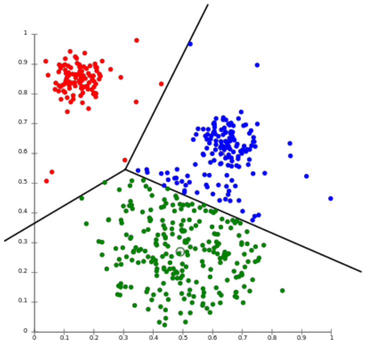
- **K-Means**
- Hierarchical Clustering
- Spectral Clustering

K-Means

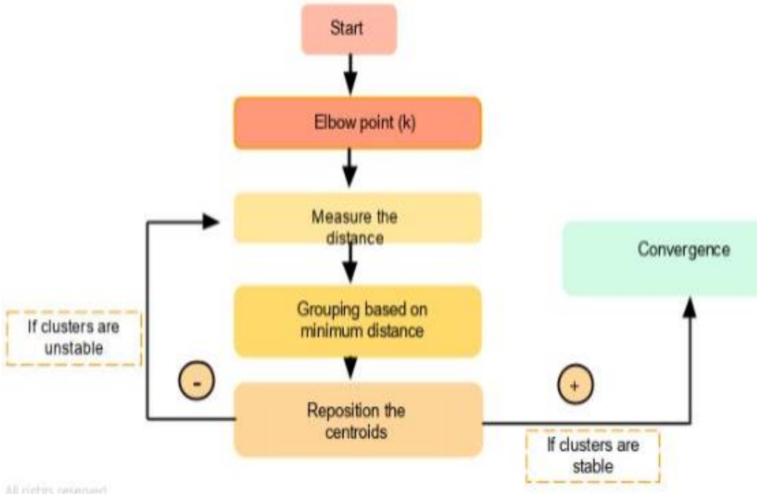
- As **unsupervised learning** algorithm
- An algorithm designed to assign clusters to each of your datapoints
- You begin with *k*, how many clusters you expect
- The algorithm randomly starts, then converges

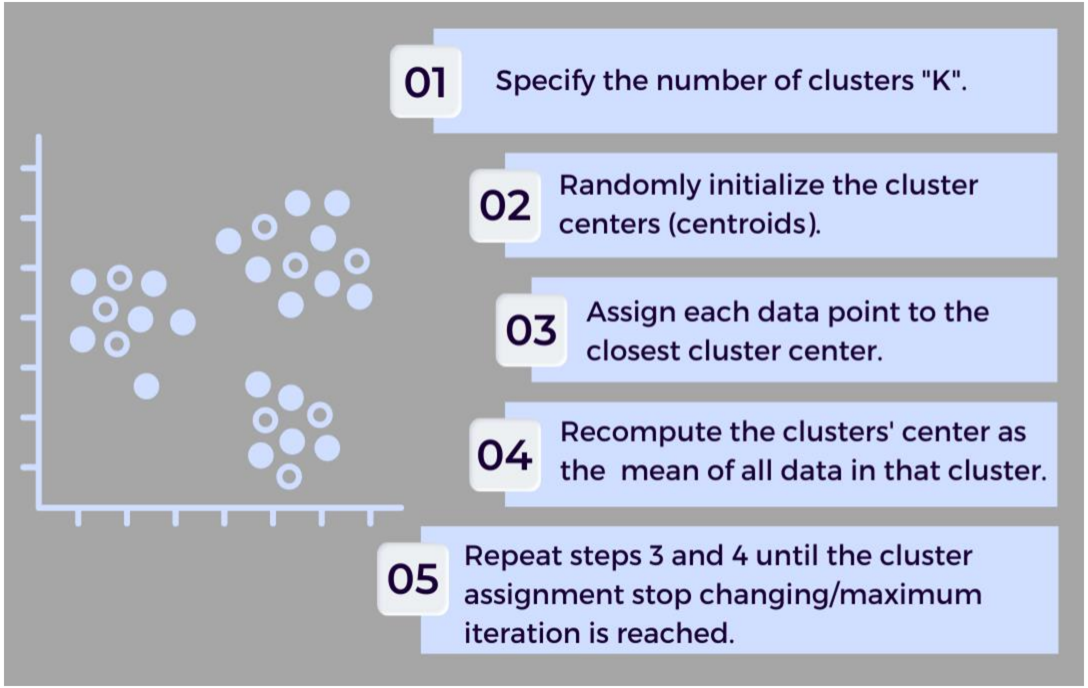
In Clustering, Quick Recap

- Input
  - a set S of n points in feature space
  - a distance measure specifying distance $d(x_i, x_j)$ between pairs (x_i, x_j)
- Output
  - A partition $\{S_1, S_2, \ldots, S_k\}$ of S



K-Means Clustering Algorithm: steps

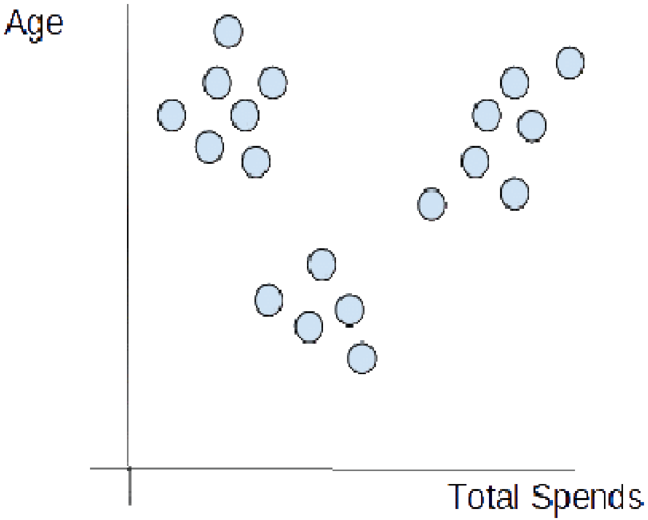




Algorithm 1 k-means algorithm

Specify the number k of clusters to assign.
Randomly initialize k centroids.
repeat
    expectation: Assign each point to its closest centroid.
    maximization: Compute the new centroid (mean) of each cluster.
until The centroid positions do not change.

How K-Means Works

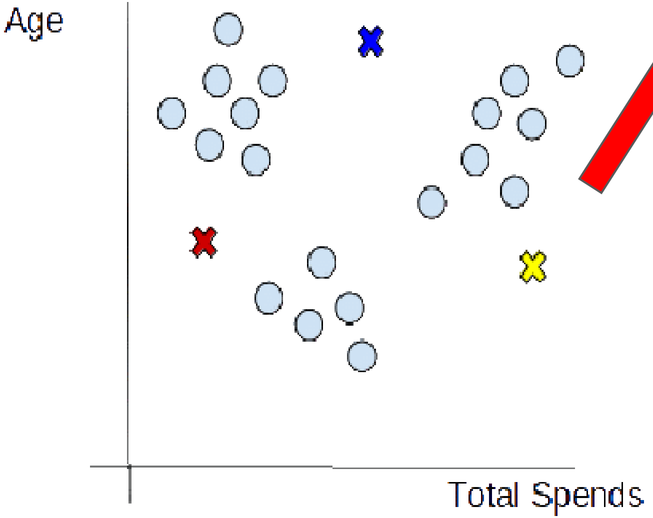


This is my Training Data

How K-Means Works

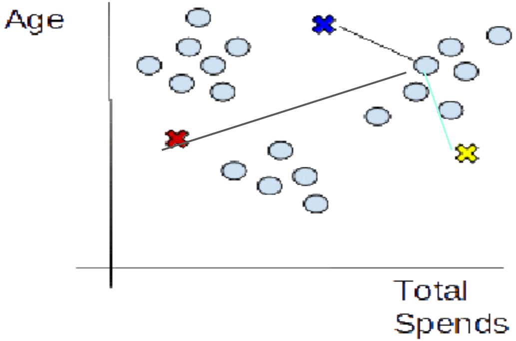
1 Let's Number of **K=3**

2 **Initialize** 3 Centroids randomly



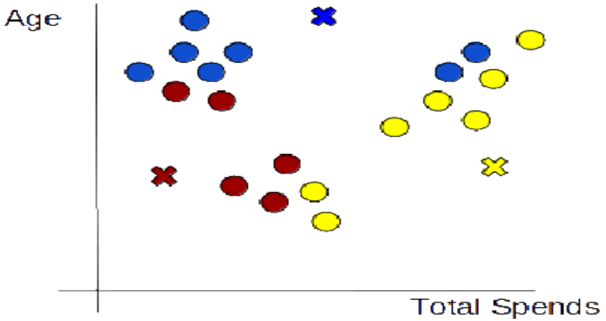
3 Assign Data Points to the Nearest Cluster (**2 STEPS**)

Step 3.1 Calculate the distance between each data point X and centroids



$$d(x, y) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}$$

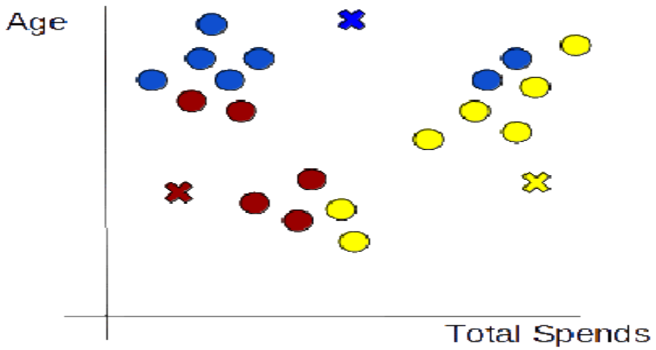
Step 3.2 Each point joins the **closest/nearest** cluster (based on its minimum distance to the centroid).



$$\underset{c_i \in C}{\arg\min}\ dist(c_i, x)^2$$

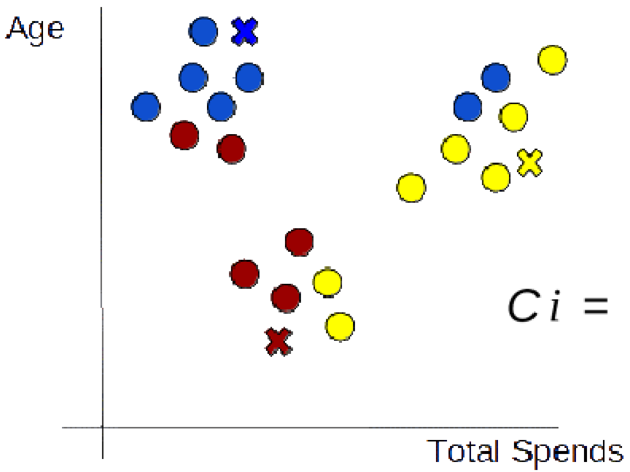
3 Assign Data Points to the Nearest Cluster (**2 STEPS**)

Calculate the distance between each data point X and centroids and Each point joins the **closest/nearest** cluster (based on its minimum distance to the centroid).



"PREVIOUS STEP FROM LAST SLIDE"

4 **Re-initialize the centroids** by calculating the average of all data points of that cluster.



$$C_i = \frac{1}{|N_i|} \sum x_i$$

'N_i' represents the number of data points X_i in ith cluster C_i.

In figure, for Red cluster,

C_{re}: (x',y')= ((x1+x2+...+x5)/5 , (y1+y2+...+y5)/5)

where N_i =5

How does K-means work?

5 **Repeat steps 3 and 4 until convergence**

Repeat **Steps 3 and 4** until optimal centroids and the assignments of data points to **correct clusters are not changing anymore.**

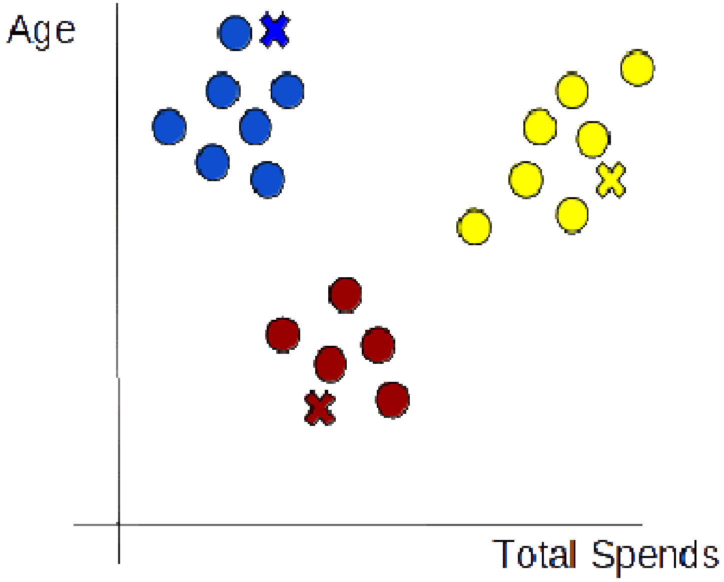


Figure: Repeat Step 3 and 4 until Convergence

The K-Means Algorithm

Training Data

K: number of clusters to discover

Algorithm 4 $\textsc{K-Means}(\mathbf{D}, K)$

for $k = 1$ **to** K **do**
    $\mu_k \leftarrow$ some random location // randomly initialize mean for kth cluster
end for
repeat
    for $n = 1$ **to** N **do**
        $z_n \leftarrow \operatorname{argmin}_k \|\boldsymbol{\mu}_k - \boldsymbol{x}_n\|^2$ // assign example n to closest center
    end for
    for $k = 1$ **to** K **do**
        $\mathbf{X}_k \leftarrow \{\, \boldsymbol{x}_n : z_n = k \,\}$ // points assigned to cluster k
        $\boldsymbol{\mu}_k \leftarrow \textsc{mean}(\mathbf{X}_k)$ // re-estimate mean of cluster k
    end for
until μs stop changing
return z // return cluster assignments

Convergence/ Stopping Criteria for K-Means Clustering

- ❑ Centroids of newly formed clusters do not change → not learning any new pattern.
- ❑ Points remain in the same cluster.
- ❑ Maximum number of iterations are reached → how many times you want to run.

Important Question to Consider

How to **determine** K ?

How to choose K?

- **If K is very large** → a cluster for every data point.
  - May defeats the purpose of clustering (assign cluster for every data point)
- **If K = 1**
  - One big cluster for the entire data set.

To select an ideal **K**, ensure that the identified clusters are distinct from each other.

- The **distance** *from each point to its cluster's center* is **much smaller** than the **distance** *between the centers of different clusters*.

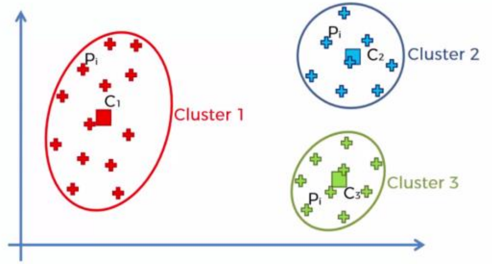
The Objective of K-Means Clustering

*"Minimize total **intra-cluster** variance, or **Within-Cluster Sum of Square (WCSS)***

Where WCSS= sum of squares of the distances of each data point in all clusters to their respective centroids (WCSS).

number of clusters   number of cases   centroid for cluster *j*   case *i*

$$\underbrace{J}_{\text{objective function}} = \sum_{j=1}^{k} \sum_{i=1}^{n} \underbrace{\left\| x_i^{(j)} - c_j \right\|^2}_{\text{Distance function}}$$



$$\text{WCSS} = \sum_{P_i \text{ in Cluster 1}} \text{distance}(P_i, C_1)^2 + \sum_{P_i \text{ in Cluster 2}} \text{distance}(P_i, C_2)^2 + \sum_{P_i \text{ in Cluster 3}} \text{distance}(P_i, C_3)^2$$

$$\text{WCSS} = \sum_{C_k}^{C_n} \left(\sum_{d_i \text{ in } C_i}^{d_m} distance(d_i, C_k)^2 \right)$$

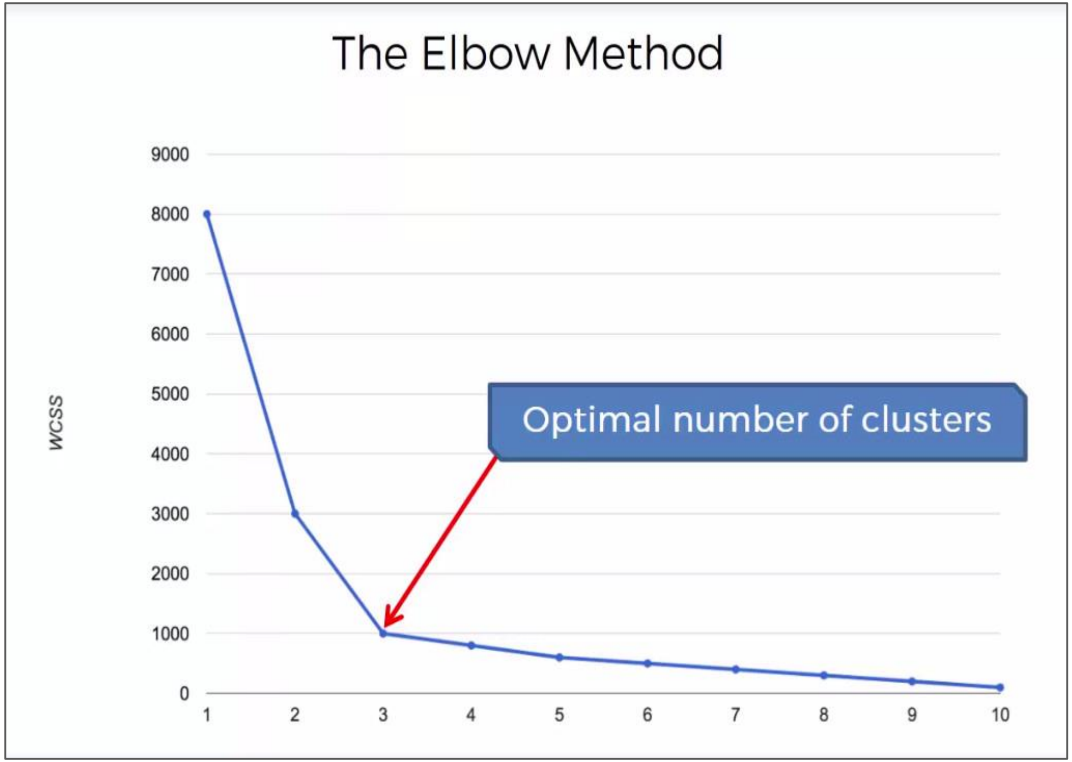
Where,
C is the cluster centroids and d is the data point in each Cluster.

Assumptions:

- Each observation belongs to at least one of the K clusters
- The clusters do not overlap
- Variation within each cluster is minimized.

Finding the optimal number of clusters K - **Elbow Method**

- X axis: vary the number of clusters K
- Y axis -> for each value of K, calculate WCSS (Within-Cluster Sum of Square).



Observe:

- As the value of K increases, the sum of square of distances decreases for every step.
- But this decreases is very fast for the initial values of K, and then it slows down. This defines the right value of K.

K-Means Properties

- Time complexity: O(KNL) where
  - K is the number of clusters
  - N is number of examples
  - L is the number of iterations
- K is a hyperparameter
  - Needs to be set in advance (or learned on dev set)
- Different initializations yield different results!
  - Doesn’t necessarily converge to best partition
- “Global” view of data: revisits all examples at every iteration