Introduction to Data Science

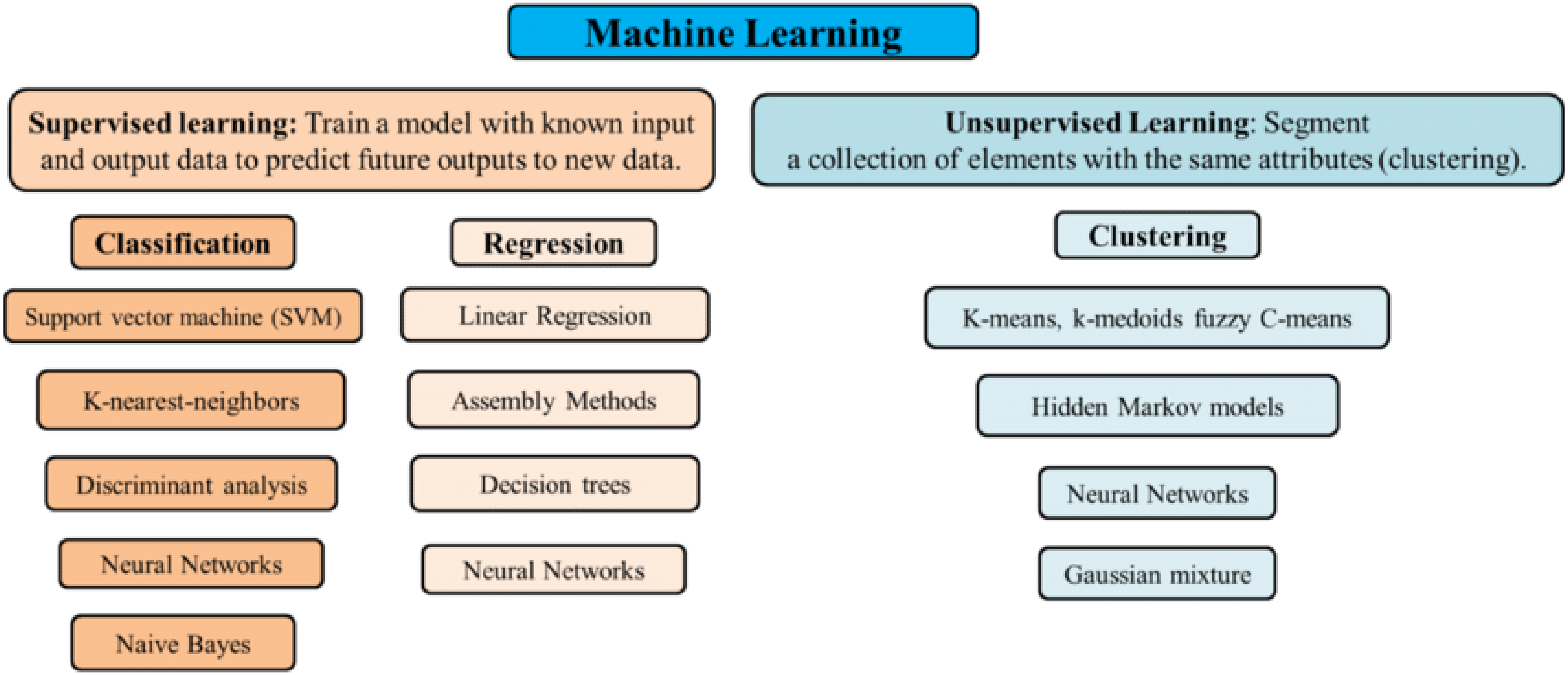
Dr. Irfan Yousuf
Department of Computer Science (New Campus)
UET, Lahore

(Lecture # 17; November 07, 2022)

Outline

- K-means

Machine Learning Algorithms



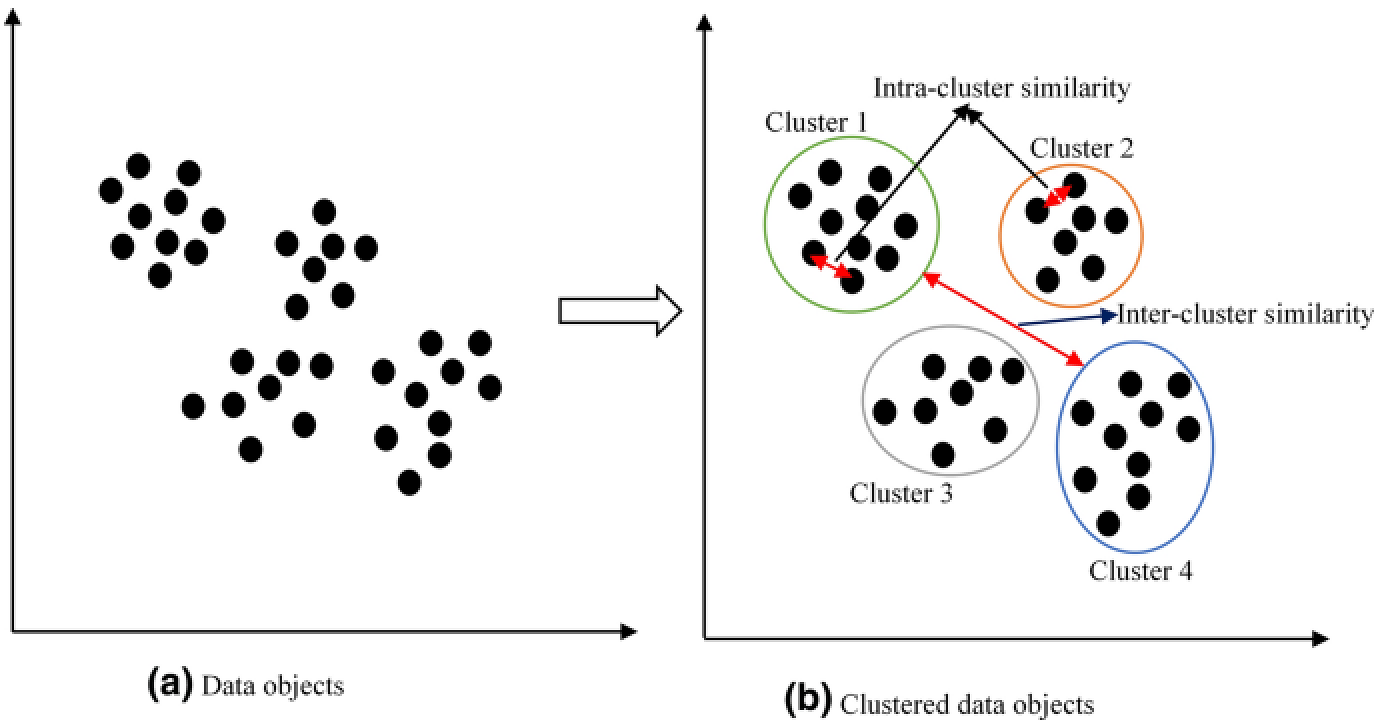
Clustering

- Clustering is one of the most common exploratory data analysis technique used to get an intuition about <u>the structure of the data.</u>
- It can be defined as the task of <u>identifying subgroups</u> in the data such that data points in the same subgroup (cluster) are very similar while data points in different clusters are very different.
- we try to find <u>homogeneous subgroups</u> within the data such that data points in each cluster are as similar as possible according to a similarity measure such as Euclidean-based distance or correlation-based distance.

Clustering

- Clustering is considered an **unsupervised learning** method since we don't have the ground truth to compare the output of the clustering algorithm to the true labels to evaluate its performance.

Clustering


Intra-cluster similarity
Cluster 1
Cluster 2
Inter-cluster similarity
Cluster 3
Cluster 4
(a) Data objects
(b) Clustered data objects

K-Means

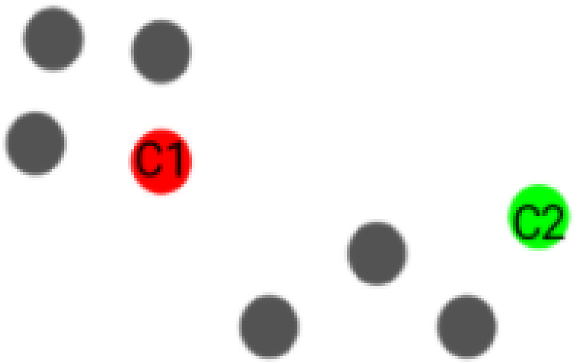
- K-means clustering is one of the simplest and popular unsupervised machine learning algorithms.
- Unsupervised algorithms make inferences from datasets using **only input vectors** without referring to known, or labelled, outcomes.
- The objective of K-means is to group similar data points together and discover underlying patterns. To achieve this objective, K-means looks for a fixed number (k) of clusters in a dataset.
    - A cluster refers to a collection of data points aggregated together because of certain similarities.

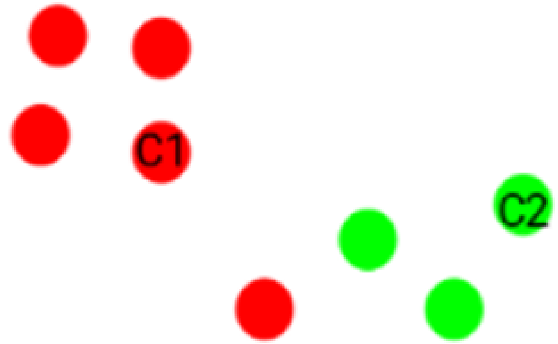
K-Means

- We define a target number k, which refers to **the number of centroids we need in the dataset**. A centroid is the imaginary or real location representing the center of the cluster.
- Every data point is allocated to each of the clusters through reducing the **within-cluster sum of squares**.
- K-means algorithm is an **iterative algorithm** that tries to partition the dataset into K pre-defined distinct non-overlapping subgroups (clusters) where each data point belongs to only one group

K-Means Algorithm

- The first step in k-means is to pick the number of clusters, K.
- Next, we randomly select the centroid for each cluster. Let's say we want to have 2 clusters, so k is equal to 2 here. We then randomly select the centroid.
- Once we have initialized the centroids, we assign each point to the closest cluster centroid:

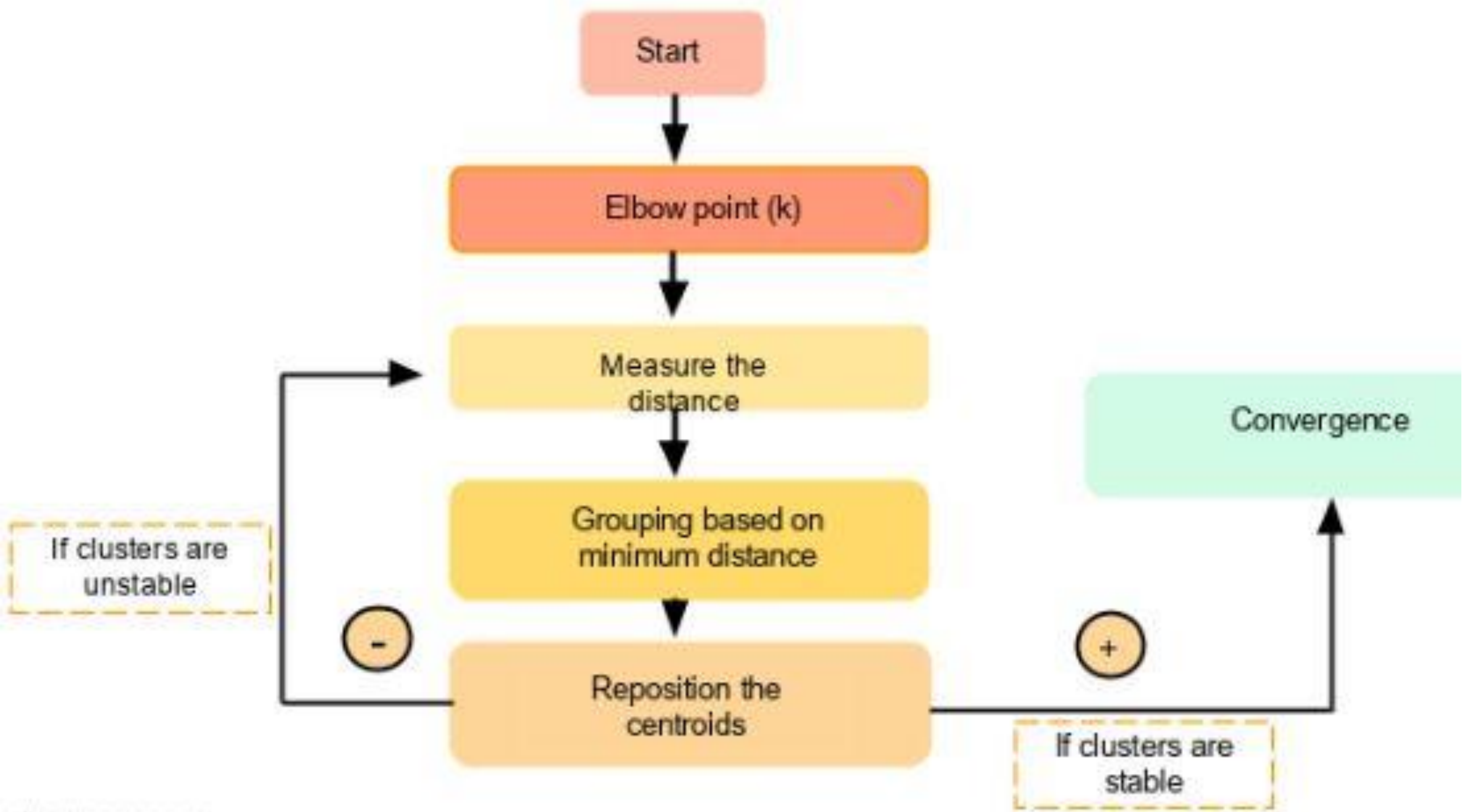




K-Means Algorithm

- Specify number of clusters K.
- Initialize centroids by first <u>shuffling the dataset </u>and then randomly selecting K data points for the centroids without replacement.
- Compute the sum of the squared distance between data points and all centroids.
- Assign each data point to the closest cluster (centroid).
- Compute the centroids for the clusters by taking the average of the all-data points that belong to each cluster.
- Keep iterating until there is no change to the centroids, i.e., assignment of data points to clusters isn’t changing.

K-Means Algorithm



Expectation Maximization

- The approach k-means follows to solve the problem is called **Expectation-Maximization**.
- The E-step is assigning the data points to the closest cluster. The M-step is computing the centroid of each cluster.

$$\underset{\text{objective function}}{J} = \sum_{j=1}^{\overset{\text{number of clusters}}{k}} \sum_{i=1}^{\overset{\text{number of cases}}{n}} \underbrace{\left\| \overset{\text{case } i}{x_i^{(j)}} - \overset{\text{centroid for cluster } j}{c_j} \right\|^2}_{\text{Distance function}}$$

K-Means Example

Suppose we want to group the visitors to a website using just their age (one-dimensional space) as follows:

$$n = 19$$

15,15,16,19,19,20,20,21,22,28,35,40,41,42,43,44,60,61,65

Initial clusters (random centroid or average):

$$k = 2$$

$$c_1 = 16$$

$$c_2 = 22$$

$$Distance\ 1 = |x_i - c_1|$$

$$Distance\ 2 = |x_i - c_2|$$

Source: https://www.saedsayad.com/clustering_kmeans.htm

K-Means Example

Iteration 1:

$$c_1 = 15.33$$
$$c_2 = 36.25$$

x_i	c_1	c_2	Distance 1	Distance 2	Nearest Cluster	New Centroid
15	16	22	1	7	1	**15.33**
15	16	22	1	7	1	
16	16	22	0	6	1	
19	16	22	3	3	2	**36.25**
19	16	22	3	3	2	
20	16	22	4	2	2	
20	16	22	4	2	2	
21	16	22	5	1	2	
22	16	22	6	0	2	
28	16	22	12	6	2	
35	16	22	19	13	2	
40	16	22	24	18	2	
41	16	22	25	19	2	
42	16	22	26	20	2	
43	16	22	27	21	2	
44	16	22	28	22	2	
60	16	22	44	38	2	
61	16	22	45	39	2	
65	16	22	49	43	2	

K-Means Example

Iteration 2:

$$c_1 = 18.56$$
$$c_2 = 45.90$$

x_i	c_1	c_2	Distance 1	Distance 2	Nearest Cluster	New Centroid
15	15.33	36.25	0.33	21.25	1	**18.56**
15	15.33	36.25	0.33	21.25	1	
16	15.33	36.25	0.67	20.25	1	
19	15.33	36.25	3.67	17.25	1	
19	15.33	36.25	3.67	17.25	1	
20	15.33	36.25	4.67	16.25	1	
20	15.33	36.25	4.67	16.25	1	
21	15.33	36.25	5.67	15.25	1	
22	15.33	36.25	6.67	14.25	1	
28	15.33	36.25	12.67	8.25	2	**45.9**
35	15.33	36.25	19.67	1.25	2	
40	15.33	36.25	24.67	3.75	2	
41	15.33	36.25	25.67	4.75	2	
42	15.33	36.25	26.67	5.75	2	
43	15.33	36.25	27.67	6.75	2	
44	15.33	36.25	28.67	7.75	2	
60	15.33	36.25	44.67	23.75	2	
61	15.33	36.25	45.67	24.75	2	
65	15.33	36.25	49.67	28.75	2	

K-Means Example

Iteration 3:

$$c_1 = 19.50$$

$$c_2 = 47.89$$

x_i	c_1	c_2	Distance 1	Distance 2	Nearest Cluster	New Centroid
15	18.56	45.9	3.56	30.9	1	**19.50**
15	18.56	45.9	3.56	30.9	1	
16	18.56	45.9	2.56	29.9	1	
19	18.56	45.9	0.44	26.9	1	
19	18.56	45.9	0.44	26.9	1	
20	18.56	45.9	1.44	25.9	1	
20	18.56	45.9	1.44	25.9	1	
21	18.56	45.9	2.44	24.9	1	
22	18.56	45.9	3.44	23.9	1	
28	18.56	45.9	9.44	17.9	1	
35	18.56	45.9	16.44	10.9	2	**47.89**
40	18.56	45.9	21.44	5.9	2	
41	18.56	45.9	22.44	4.9	2	
42	18.56	45.9	23.44	3.9	2	
43	18.56	45.9	24.44	2.9	2	
44	18.56	45.9	25.44	1.9	2	
60	18.56	45.9	41.44	14.1	2	
61	18.56	45.9	42.44	15.1	2	
65	18.56	45.9	46.44	19.1	2	

K-Means Example

Iteration 4:

$c_1 = 19.50$

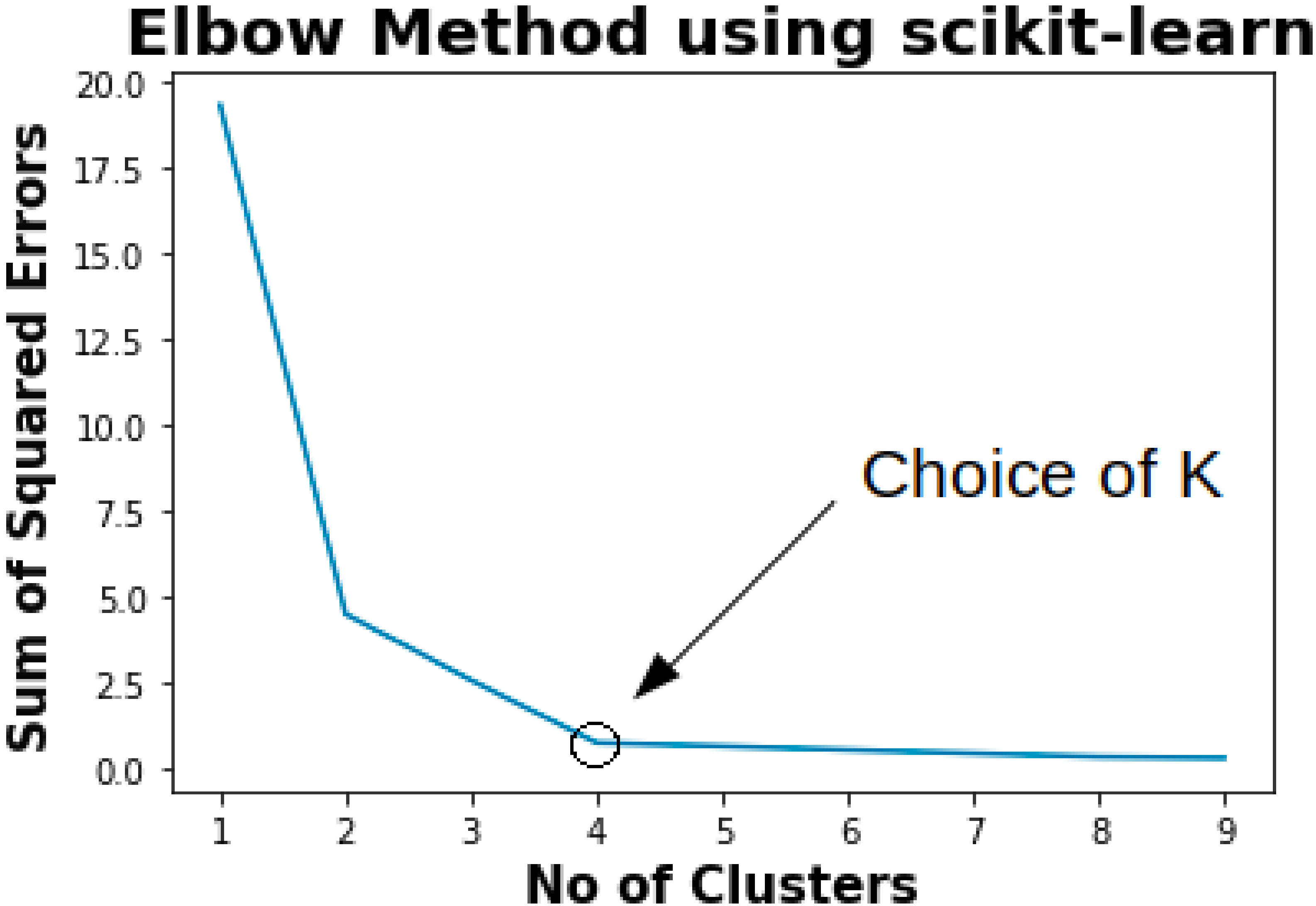
$c_2 = 47.89$

x_i	c_1	c_2	Distance 1	Distance 2	Nearest Cluster	New Centroid
15	19.5	47.89	4.50	32.89	1	19.50
15	19.5	47.89	4.50	32.89	1	
16	19.5	47.89	3.50	31.89	1	
19	19.5	47.89	0.50	28.89	1	
19	19.5	47.89	0.50	28.89	1	
20	19.5	47.89	0.50	27.89	1	
20	19.5	47.89	0.50	27.89	1	
21	19.5	47.89	1.50	26.89	1	
22	19.5	47.89	2.50	25.89	1	
28	19.5	47.89	8.50	19.89	1	
35	19.5	47.89	15.50	12.89	2	47.89
40	19.5	47.89	20.50	7.89	2	
41	19.5	47.89	21.50	6.89	2	
42	19.5	47.89	22.50	5.89	2	
43	19.5	47.89	23.50	4.89	2	
44	19.5	47.89	24.50	3.89	2	
60	19.5	47.89	40.50	12.11	2	
61	19.5	47.89	41.50	13.11	2	
65	19.5	47.89	45.50	17.11	2	

Elbow Method

- The elbow method runs k-means clustering on the dataset for a range of values for k (say from 1-10) and then for each value of k computes an average score for all clusters.
- We can compute Within-Cluster Sum of Squares (WCSS), the sum of square distances from each point to its assigned center.
- We then draw k vs. WCSS.

Elbow Method



K-Means Implementation

Implement k-Means Algorithm

Summary

- K-means