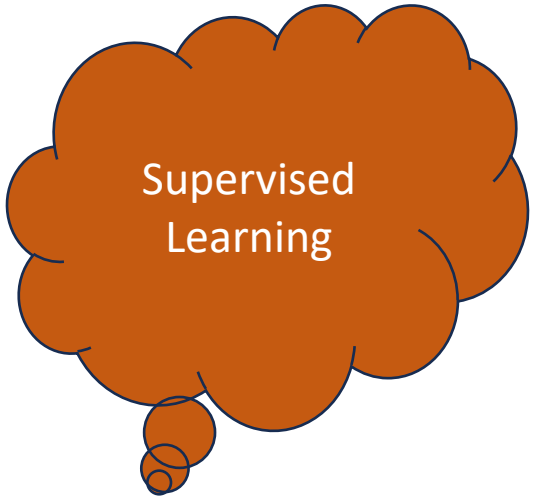
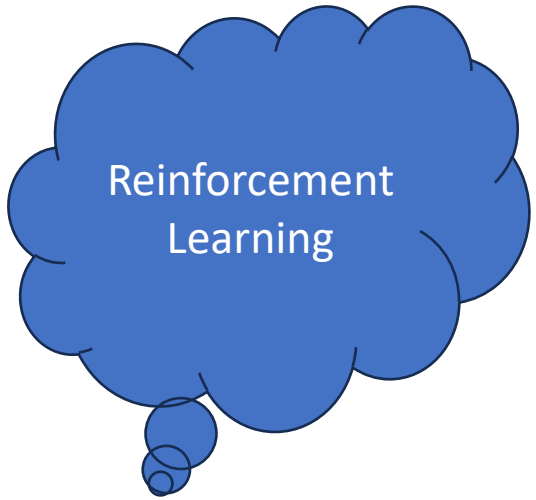
Cluster Analysis

Agenda:

1. Unsupervised learning
2. Cluster analysis
3. Application of cluster
4. K-Means cluster
5. Objective function
6. Elbow method

Intro to Machine Learning

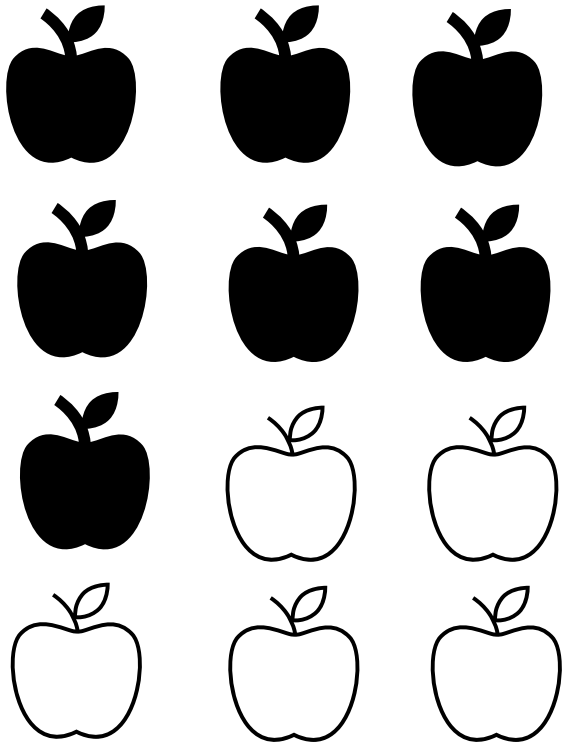




Statistical Machine Learning

Supervised Learning

1. Classification Tasks

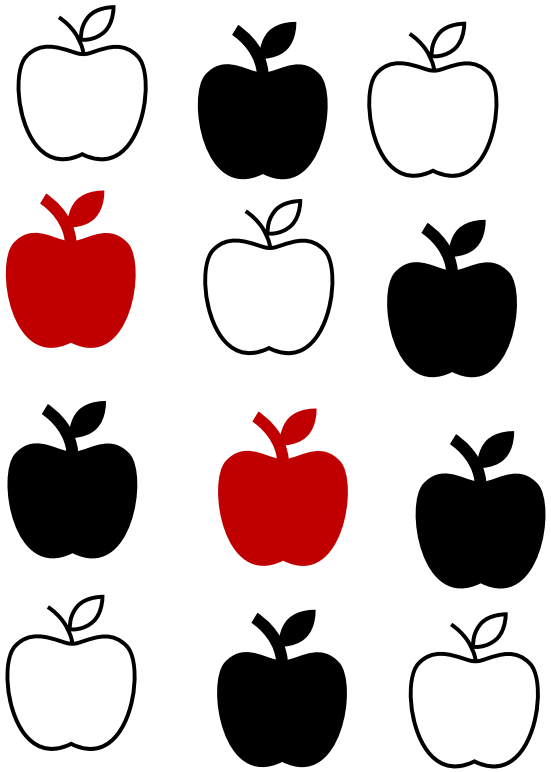


2. Regression Tasks

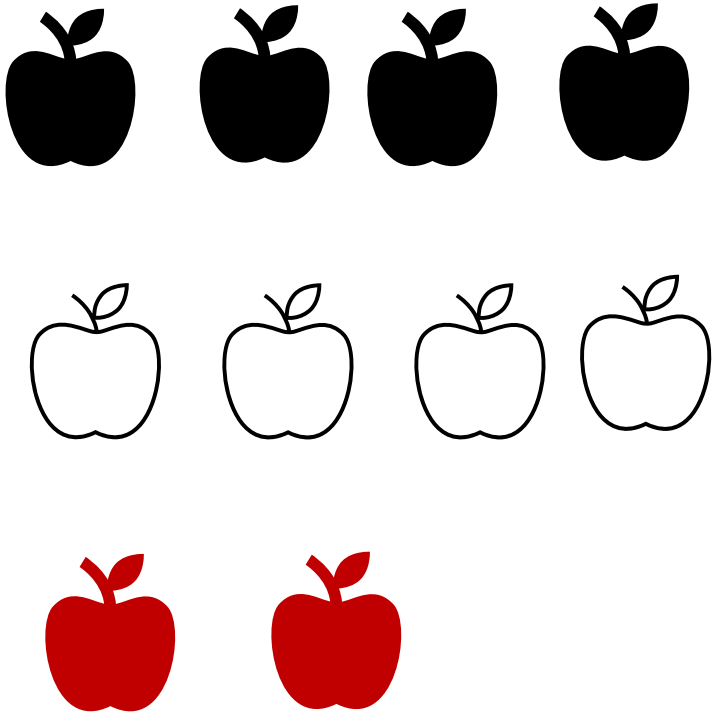
Statistical Machine Learning

Unsupervised Learning

Before Cluster



After Cluster



Cluster Analysis

- **Cluster:** A collection of data objects within a larger set that is:
  - Similar (or related) to one another within the same group and,
  - dissimilar (or unrelated) to the objects outside the group

- **Cluster analysis (or clustering, data segmentation, . . .):**
  - Define similarities among data based on the characteristics found in the data (input from the user!)
  - Group similar data objects into clusters.

- **Unsupervised learning:**
  - No predefined classes or specific labels.
  - learning by observation

- **Typical applications:**
  - As a stand-alone tool to get insight into data distribution.
  - As a preprocessing step for other algorithms.
  - Market basket analysis & customer segmentation.

Quality of Clustering

Major Qualities

- **A good clustering method will produce high-quality clusters.**
  - **High intra-class similarity:**
    - Cohesive within clusters.
  - **Low inter-class similarity:**
    - Distinctive between clusters.

- **The quality of a clustering method depends on:**
  - The similarity measure used by the method, its implementation, and its ability to discover some or all the hidden patterns.

Cluster Analysis Approach

Major Types

- **Partitioning approach:**
  - Construct various partitions and then evaluate them by some criterion.
  - Minimizing the sum of square errors.
  - **Typical methods:** K-means, Gaussian Mixture Model (GMM) k-medoids, CLARA, CLARANS.

- **Hierarchical approach:**
  - Create a hierarchical decomposition of the set of data (or objects) using some criterion.
  - **Typical methods:** AGNES, DIANA, BIRCH, CHAMELEON.

- **Density-based approach:**
  - Based on connectivity and density functions.
  - **Typical methods:** DBSCAN, OPTICS, DENCLUE.

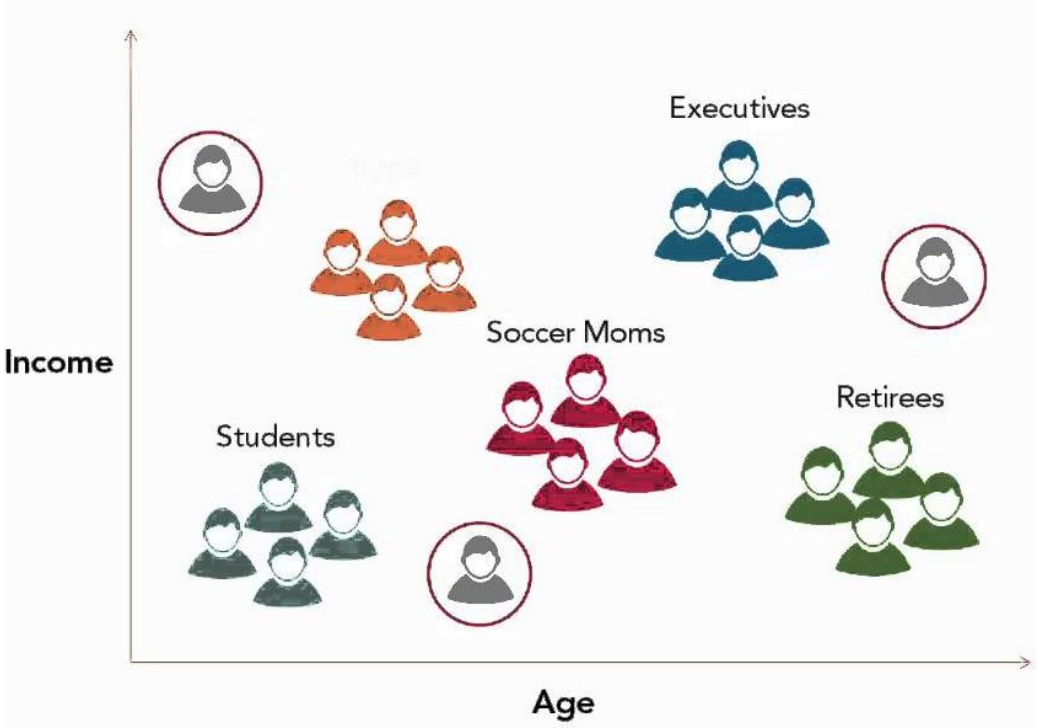
- **Grid-based approach:**
  - Based on a multiple-level granularity structure.
  - **Typical methods:** STING, WaveCluster, CLIQUE.

Customer Segmentation

Concepts

- **Customer segmentation** is the process of grouping customers based on similar characteristics for the purpose of targeted marketing and resource allocation.

- **K-means clustering** is a technique used for customer segmentation that groups similar data points together and is classified under unsupervised machine learning.



Note: When dealing with complex or non-spherical clusters and the presence of outliers, it is advisable to consider other clustering algorithms like DBSCAN and Gaussian Mixture Model (GMM).

Distance Formula in Clustering Algorithms

Common Distance Formula

There are several distance formulas or metrics that can be used in clustering algorithms, depending on the nature of the data and the specific problem at hand. Here are some commonly used distance metrics:

1. Euclidean distance: The most common distance metric used in K-means and many other clustering algorithms. It calculates the straight-line distance between two points in an n-dimensional space, as explained in the previous response.

2. Manhattan distance (L1 distance): It measures the distance between two points by summing the absolute differences of their coordinates along each dimension. In 2D space, it corresponds to the "city-block" distance.

3. Chebyshev distance (L∞ distance): It calculates the maximum absolute difference between the coordinates of two points along any dimension. It represents the maximum distance in any direction.

4. Minkowski distance: A generalization of Euclidean and Manhattan distances that introduces a parameter "p" to control the distance calculation. When p=2, it becomes the Euclidean distance, and when p=1, it becomes the Manhattan distance.

1. Initialization: Choose the number of clusters (K) you want to form and randomly initialize **K points called centroids.** These centroids will serve as the initial cluster centers.

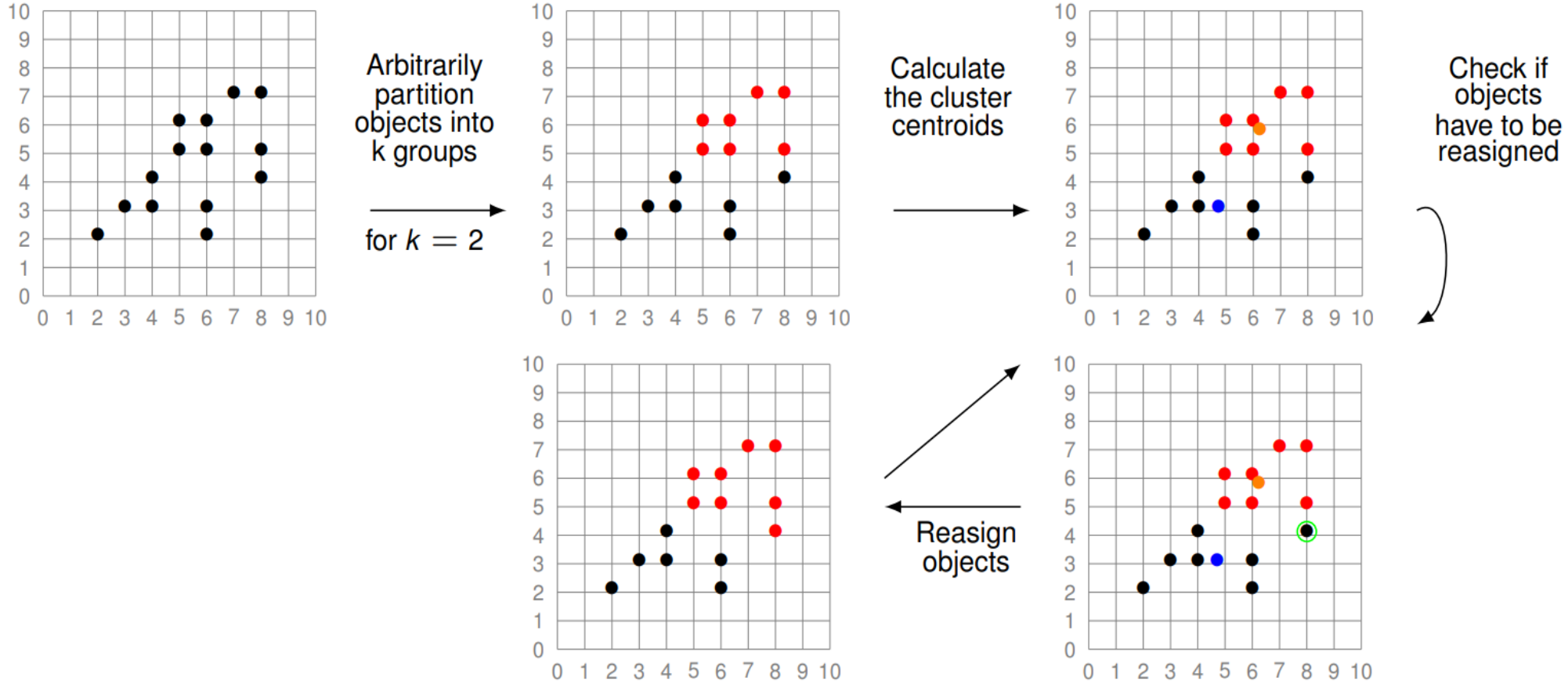
2. Assignment: For each data point in the dataset, calculate the distance (e.g., Euclidean distance, Manhattan distance) to each centroid. Assign the data point to the cluster whose centroid is the closest (i.e., the minimum distance). This step creates K clusters.

3. Update: After all data points are assigned to clusters, update the centroids of each cluster by calculating the mean of all data points assigned to that cluster. The new centroid will be the center of gravity of the points in that cluster.

4. Repeat: Repeat the assignment and update steps iteratively until convergence. Convergence is reached when the **centroids no longer change significantly between iterations** or when a maximum number of iterations is reached.

K-means Cluster

Visual Approach



picture credit: www.fau.de

How K-Means Work?

In-depth intuition

1. Euclidean Distance: $$d(\mathbf{p}, \mathbf{q}) = \sqrt{\sum_{i=1}^{n} (q_i - p_i)^2}$$

p, q = two points in Euclidean n-space

q_i, p_i = Euclidean vectors, starting from the origin of the space (initial point)

n = n-space

2. Manhattan Distance: The formula to calculate the Manhattan distance between two points (x1, y1) and (x2, y2) is:

- Manhattan distance = |x2 - x1| + |y2 - y1|

How K-Means Works?

Point ID	X	Y
P1	0.5	1.0
P2	0.0	4.0
P3	3.5	2.0
P4	0.5	-4.5
P5	-0.5	-3.5

Centroid	X	Y
M1	-2.5	0.5
M2	2.5	4.5

How K-Means Work?

In-depth intuition

Data, X = { (0.5 , 1.0), (0.0, 4.0), (3.5, 2.0), (0.5, -4.5) (-0.5, -3.5) }

Initial Centroid, M1 = (-2.5, 0.5); M2 = (2.5, 4.5), 2-Means Cluster for iterations=2.

$$d(\mathbf{p}, \mathbf{q}) = \sqrt{\sum_{i=1}^{n} (q_i - p_i)^2}$$

Centroids / Distance	(0.5 , 1.0)	(0.0, 4.0)	(3.5, 2.0)	(0.5, -4.5)	(-0.5, -3.5)
M1 = (-2.5, 0.5)	3.04	4.30	6.18	5.83	4.47
M2 = (2.5, 4.5)	4.03	2.54	2.69	9.21	8.54
Clusters =	M1	M2	M2	M1	M1

New, C1 = Mean(M1) = 0.5+0.5-0.5)/3 = 0.167 , 1.0-4.5-3.5/3 = -2.33 = (0.167 , -2.33)

New, C2 = Mean(M2) = (0+3.5)/2 = 1.75 , 4.0+2.0/2 = 3.0 = (1.75 , 3.0)

How K-Means Work?

In-depth intuition

New M1 = (0.167 , 2.33) & M2 = (1.75 , 3.0)

$$d(\mathbf{p}, \mathbf{q}) = \sqrt{\sum_{i=1}^{n} (q_i - p_i)^2}$$

New Centroids / Distance	(0.5 , 1.0)	(0.0, 4.0)	(3.5, 2.0)	(0.5, -4.5)	(-0.5, -3.5)
M1 = (0.167, -2.33)					
M2 = (1.75 , 3.0)					
Clusters =					

How K-Means Work?

In-depth intuition (Do it Yourself)

Consider the data set $X := \{x_1, ..., x_6\} := \left\{\begin{pmatrix}1\\1\end{pmatrix}, \begin{pmatrix}1.5\\0.5\end{pmatrix}, \begin{pmatrix}-2.5\\0\end{pmatrix}, \begin{pmatrix}1\\3\end{pmatrix}, \begin{pmatrix}0\\2\end{pmatrix}, \begin{pmatrix}-1\\-2\end{pmatrix}\right\}$, clustered in Clusters $C_1 := \{1,2,3\}$ and $C_2 := \{4,5,6\}$, with cluster means $m_1 = \begin{pmatrix}0\\0.5\end{pmatrix}$ and $m_2 = \begin{pmatrix}0\\1\end{pmatrix}$. Calculate three iterations of the k-means algorithm, starting with the given data.

Solution: Initial state: $C_1 := \{1,2,3\}$ and $C_2 := \{4,5,6\}$
The cluster means, as given in the exercise, are the arithmetic means:

$$m_1 := \frac{1}{|C_1|}\sum_{i\in C_1} x_i = \frac{1}{3}\left(\begin{pmatrix}1\\1\end{pmatrix} + \begin{pmatrix}1.5\\0.5\end{pmatrix} + \begin{pmatrix}-2.5\\0\end{pmatrix}\right) = \begin{pmatrix}0\\0.5\end{pmatrix}$$

and furthermore

$$m_2 := \frac{1}{|C_2|}\sum_{i\in C_2} x_i = \frac{1}{3}\left(\begin{pmatrix}1\\3\end{pmatrix} + \begin{pmatrix}0\\2\end{pmatrix} + \begin{pmatrix}-1\\-2\end{pmatrix}\right) = \begin{pmatrix}0\\1\end{pmatrix}.$$

First iteration:

Assignment step: We assign each data point to the cluster whose mean is the closest of all means to the data point, i.e., x is assigned to cluster number $\operatorname{argmin}_k \|x - m_k\|$.
Since the first component of m_1 and m_2 are equal, the argmin statement reduces to "is the second component of x closer to 0.5 or 1". Hence, we get

$$C_1 := \{2,3,6\} \text{ and } C_2 := \{1,4,5\}.$$

We update the cluster means, i.e.,

$$m_1 := \frac{1}{|C_1|}\sum_{i\in C_1} x_i = \frac{1}{3}\left(\begin{pmatrix}1.5\\0.5\end{pmatrix} + \begin{pmatrix}-2.5\\0\end{pmatrix} + \begin{pmatrix}-1\\-2\end{pmatrix}\right) = \begin{pmatrix}-\frac{2}{3}\\-\frac{1}{2}\end{pmatrix}$$

and furthermore

$$m_2 := \frac{1}{|C_2|}\sum_{i\in C_2} x_i = \frac{1}{3}\left(\begin{pmatrix}1\\1\end{pmatrix} + \begin{pmatrix}1\\3\end{pmatrix} + \begin{pmatrix}0\\2\end{pmatrix}\right) = \begin{pmatrix}\frac{2}{3}\\2\end{pmatrix}$$

First iteration is complete.

Second iteration: We again assign each data point to the cluster whose mean is closest to the data points amongst all means, i.e.,

$$C_1 := \{3,6\} \text{ and } C_2 := \{1,2,4,5\}.$$

Update step:

$$m_1 := \frac{1}{|C_1|}\sum_{i\in C_1} x_i = \frac{1}{2}\begin{pmatrix}-\frac{7}{2}\\-2\end{pmatrix} = \begin{pmatrix}-\frac{7}{4}\\-1\end{pmatrix}$$

and

$$m_2 := \frac{1}{|C_2|}\sum_{i\in C_2} x_i = \frac{1}{4}\begin{pmatrix}\frac{5}{2}\\\frac{5}{2}\end{pmatrix} = \frac{1}{8}\begin{pmatrix}7\\13\end{pmatrix}$$

Note that in the next assignment step, x gets assigned to cluster number $\operatorname{argmin}_k \|x - m_k\|$.

$$C_1 := \{3,6\} \text{ and } C_2 := \{1,2,4,5\}.$$

Hence this is the final iteration (since we got the same clusters than last time).

Objective Function for K-Means Cluster

Cost/WCSS

The objective function of k-means clustering is to minimize the within-cluster variance, which is achieved by minimizing the squared Euclidean distance between each data point and the centroid of the cluster to which it has been assigned. Mathematically, the objective function J for k-means clustering can be expressed as:

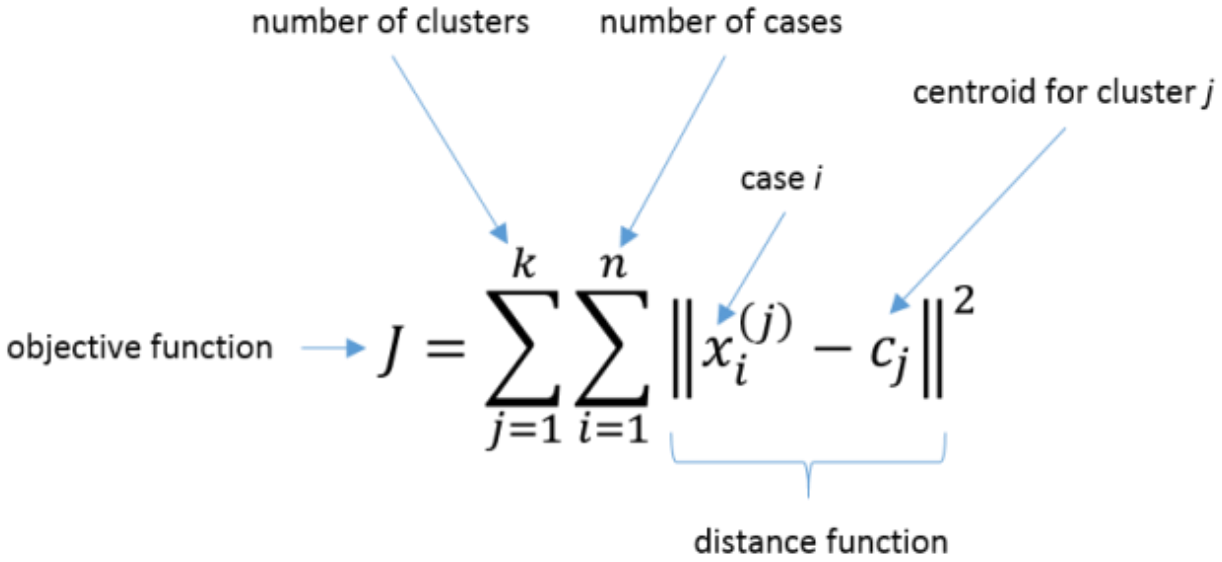
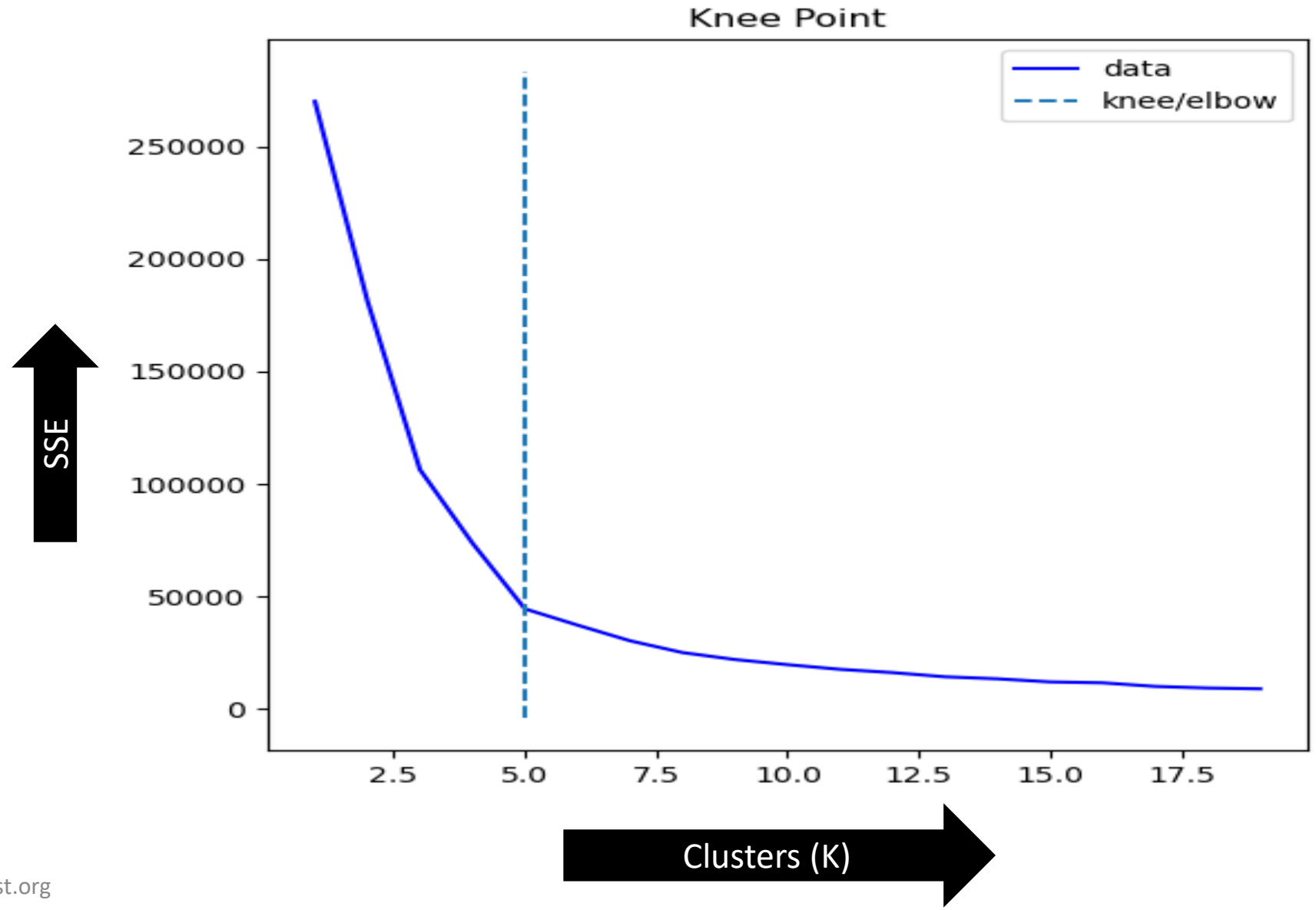


fig: Objective function of K-means cluster

How to Select the Optimal Number of Clusters?

Elbow Method



How to Select the Optimal Number of Clusters?

Elbow Method Algorithm

The Elbow Method is a technique used to determine the optimal number of clusters (k) in K-means clustering. It is a graphical method that helps to find the value of k where the **within-cluster sum of squares (WCSS)** starts to level off, creating an **"elbow"** shape in the plot. Here's a step-by-step explanation of the Elbow Method in K-means:

1. Initialize: Choose a range of values for k (the number of clusters) that you want to try. Usually, this range is based on prior knowledge or domain expertise.

2. K-means Clustering: Apply the K-means algorithm for each value of k in the chosen range. This involves running K-means multiple times with different values of k.

3. Calculate WCSS: For each k value, calculate the sum of squared distances between data points and their assigned cluster centroids. This is the Within-Cluster Sum of Squares (WCSS) for that specific value of k.

4. Plot the Elbow Curve: Create a line plot with the number of clusters (k) on the x-axis and the corresponding WCSS on the y-axis.

5. Identify the Elbow: Examine the plot to identify the point where the decrease in WCSS starts to slow down, forming an elbow-like bend. This point is considered the optimal number of clusters for your data.

6. Choose k: Once you have identified the "elbow" point, you can select the corresponding k value as the optimal number of clusters for your K-means clustering.

Let’s Implement it with Python!