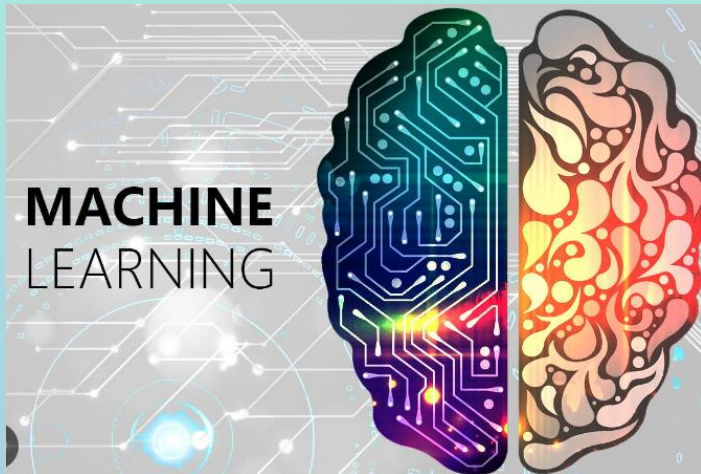


Artificial Intelligence and Machine Learning :
ACCA6011
Unsupervised learning

Dr. Seena Joseph

Unsupervised learning

Machine learning



Supervised learning



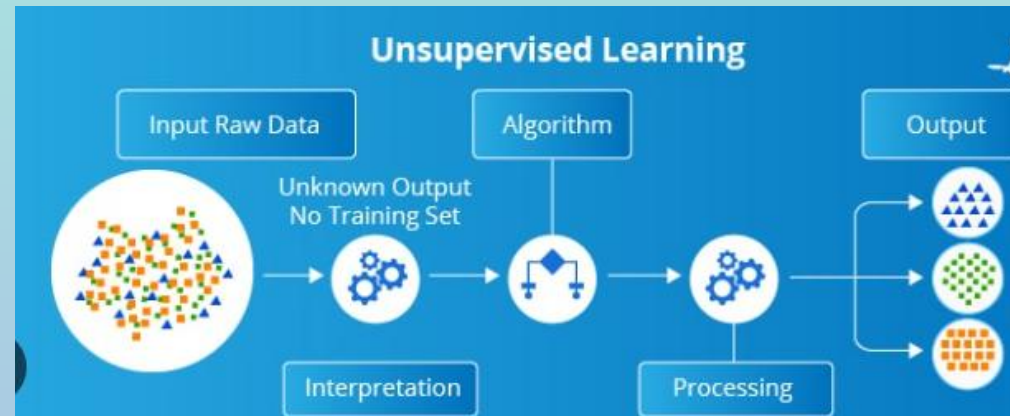
Unsupervised learning



Reinforcement
learning

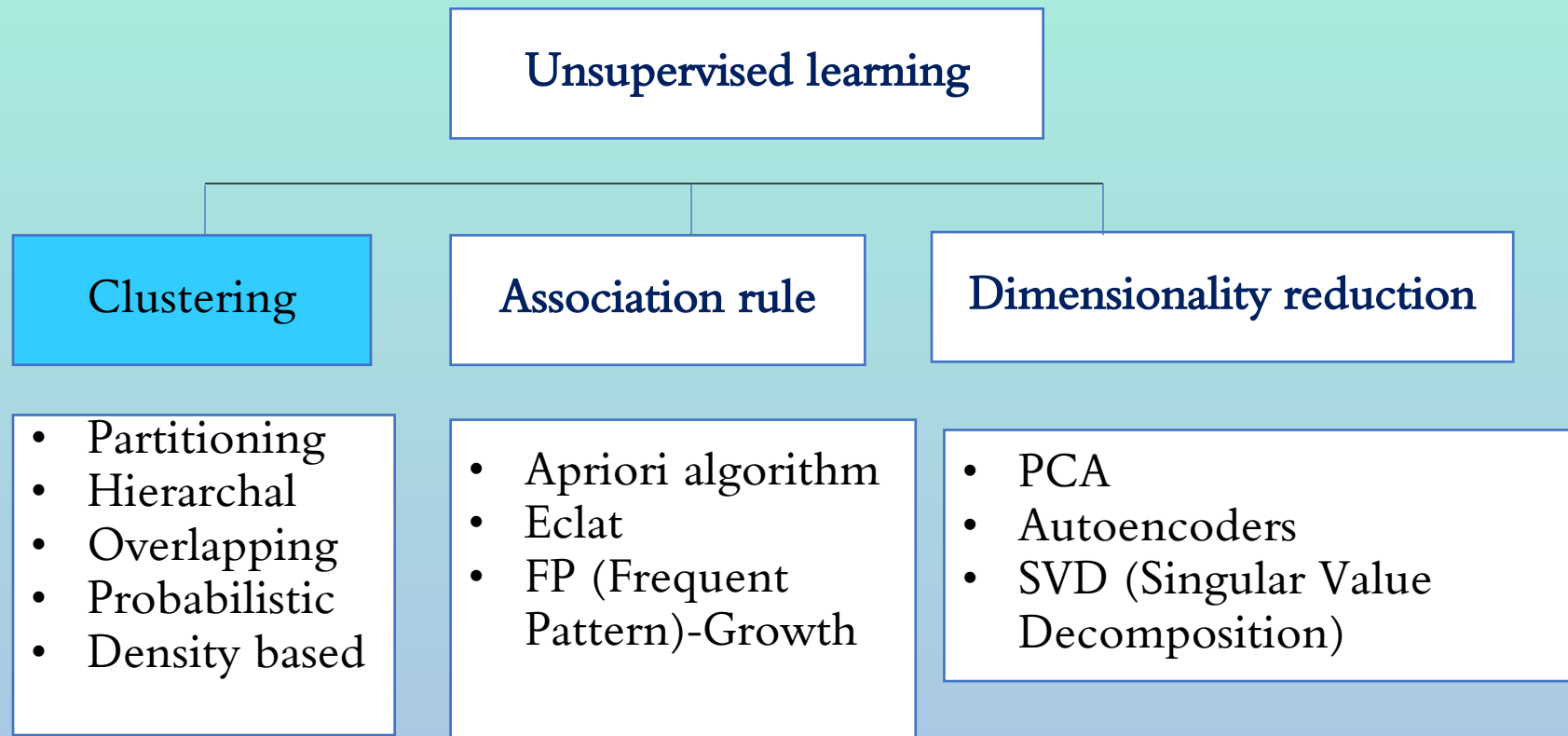
Unsupervised learning

- Analysis of unlabelled data
- Discover hidden structures in data
- Extract useful information using input data



Unsupervised learning

Types of unsupervised learning



Unsupervised learning

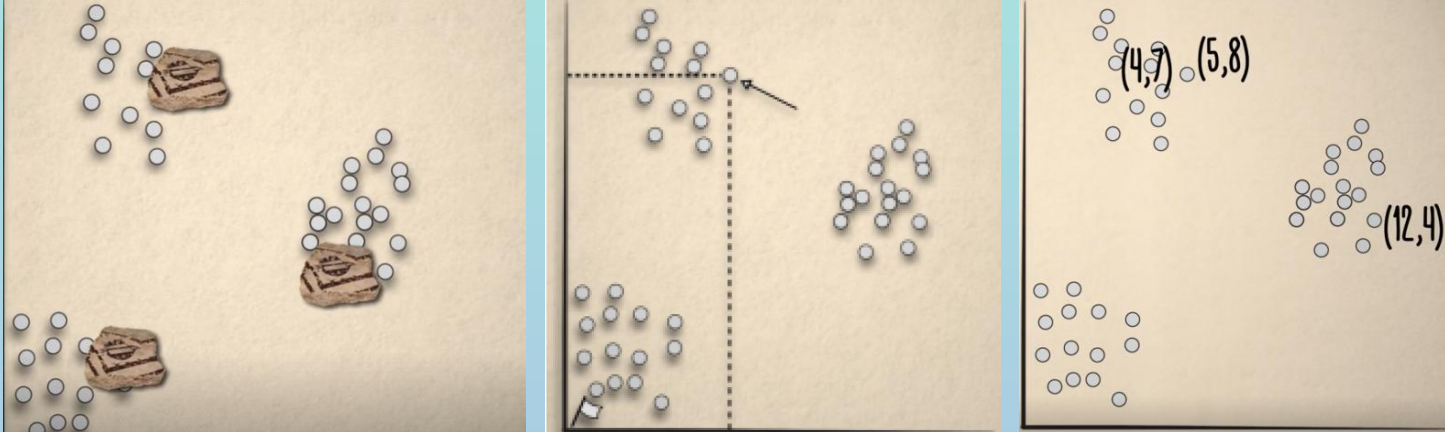
- Clustering
 - Automatically discovering natural groupings in the data.
 - It groups data points according to the notion of similarity.
 - Similarity
 - Find the similarity between the data points based on distance metric between them
 - Example of distance metric
 - Euclidean distance, Manhattan distance
 - Assigning each data point to exactly one group.
- Clustering methods simply try to group similar patterns into clusters whose members are more similar to each other (according to some **distance measure**) than to members of other clusters.

Unsupervised learning

Clustering

Goal: **Group** the **data** according to **Similarity**

Data: Imagine that you are leading an archaeological dig. And every time someone on your team unearths an artifact, record the position of the artifact.



Data point: Location for one artifact

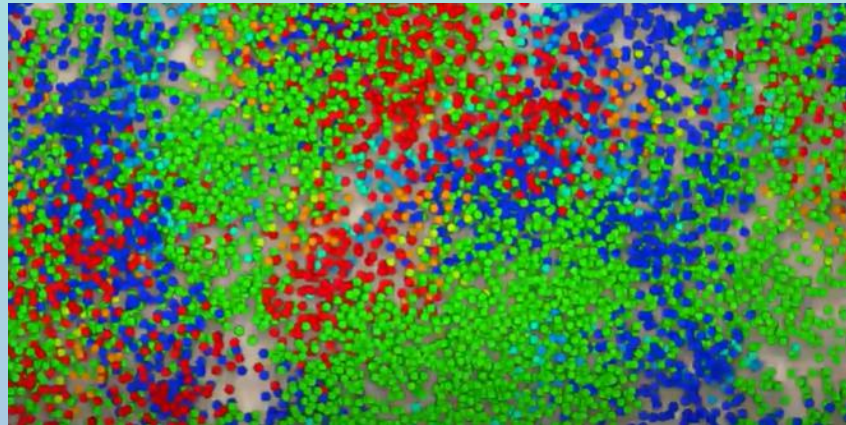
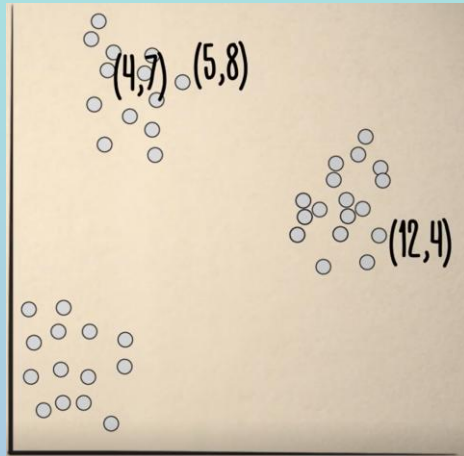
Dataset: a collection of all of the locations of the different data artifact

Similarity: Close in terms of physical distance

Groups: Three clusters

Unsupervised learning

- Higher dimensional data
- Data type : words or pictures or genomes

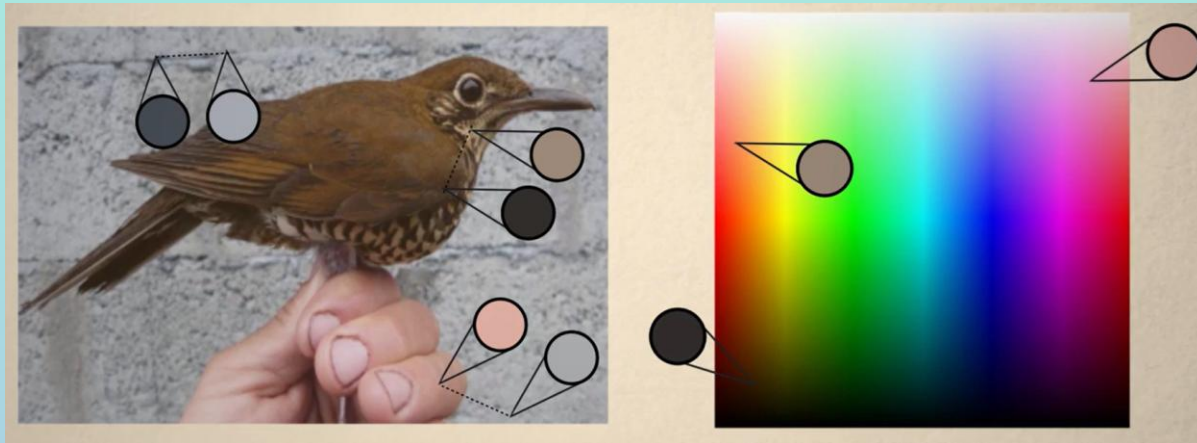


ut, placerat ac, adipiscing vitae, felis. Curabit
in arcu libero, nonummy eget, consectetur id, v
cula augue eu neque. Pellentesque habitant morbi
malesuada fames ac turpis egestas. Mauris ut
cus sem. Nulla et lectus vestibulum urna fringilla
amet tortor gravida placerat. Integer sapien est,
a ac, nunc. Praesent eget sem vel leo ultrices l
morbi dolor nulla, malesuada eu, pulvinar at, mo

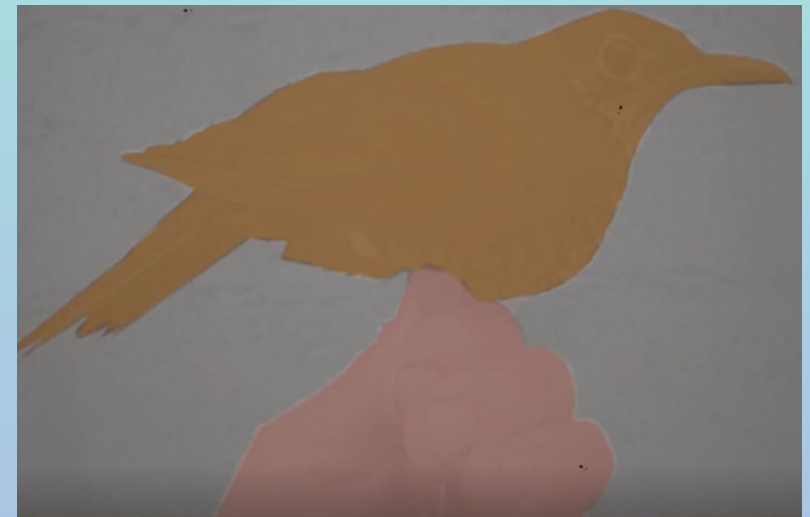


Unsupervised learning

- Data : Image

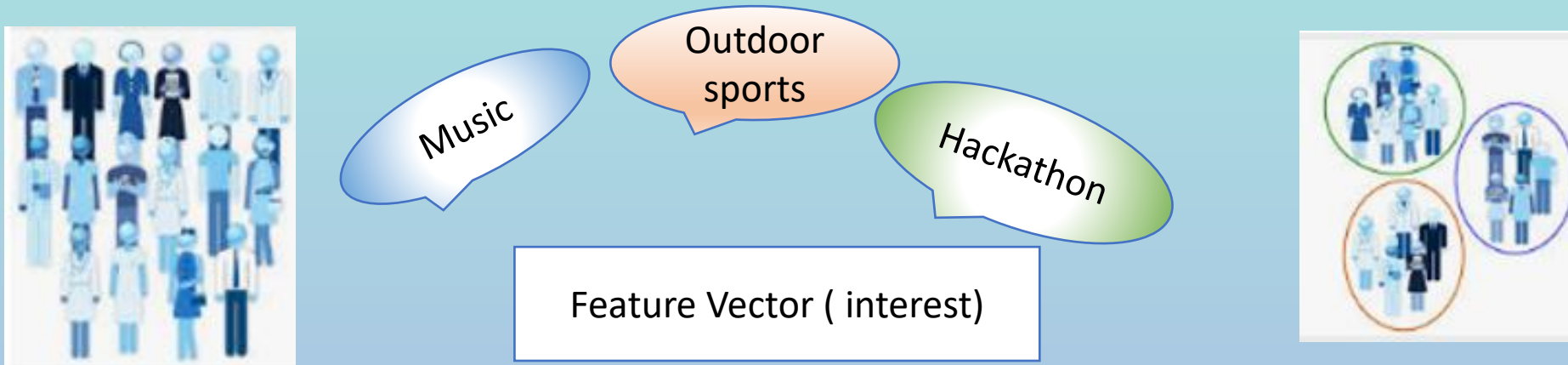


DATA POINT = COLOR OF PIXEL
AND
LOCATION OF PIXEL



Unsupervised learning

- Clustering – Example
- Users on meetup.com
- Similarity notion: Number of common interest



Unsupervised learning - summary

Clustering

Group data based on similarity

- Why and When to use clustering
- Why
 - Data analysis in Retail, Finance, and marketing, computer vision
 - Helps decision-making in this domain to gain further customer insight
- When
 - When you are starting out with a large and unlabeled dataset
 - When you do not know how many or which classes your data is divided
 - When annotating (labeling) your data can be very expensive



Unsupervised learning

K-Means Clustering

K-Means is one of the popular and simple clustering algorithm

K-means is a centroid-based clustering algorithm, where we calculate the distance between each data point and a centroid to assign it to a cluster.

The goal is to identify the K number of groups in the dataset.

It is an iterative process of assigning each data point to the groups and slowly data points get clustered based on similar features.

The objective is to minimize the sum of distances between the data points and the cluster centroid, to identify the correct group each data point should belong to.

Unsupervised learning

K-Means Clustering

K-Means Algorithm

1. Initialization

- Choose the number of clusters, k

- Random initialization of k centroids.

2. Assignment

- Assign each datapoint to the nearest centroid based on a distance metric (commonly, Euclidean distance)

3. Re-initialize centroids

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

4. Repeat steps 2 and 3

- Repeat steps 2 and 3 until the centroids stabilize(no significant change) or a maximum number of iterations reached.

Unsupervised learning

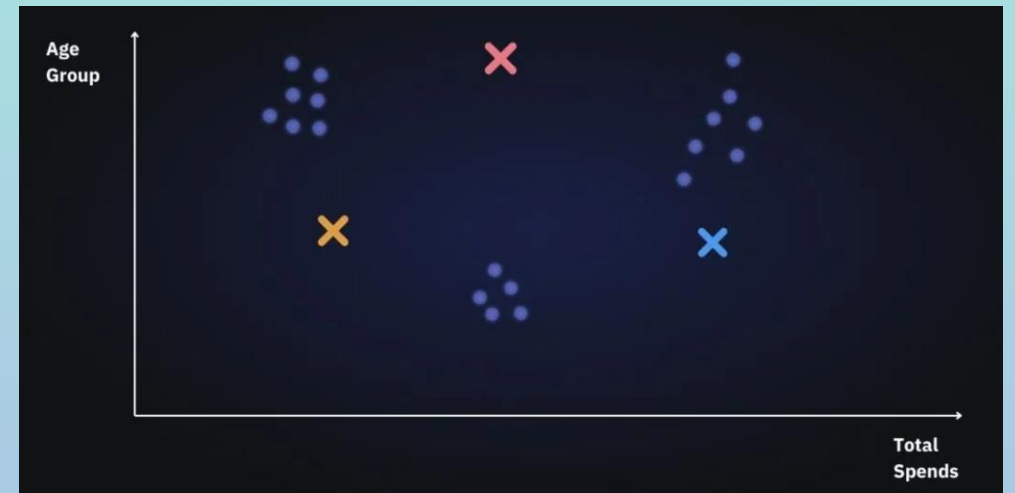
K-Means Algorithm

Example

A customer dataset with age and purchase history. Purpose is to segment the customers to better understand them and improve the marketing strategies.

Step 1 : Initialization

$k=3$ and find initialize 3 random centroids C_k



Unsupervised learning

K-Means Algorithm

Step 2 : Assignment

In this step, datapoints are allocated to the nearest centroid, C_k .

This is accomplished by calculating the distance between the datapoints and the centroid.

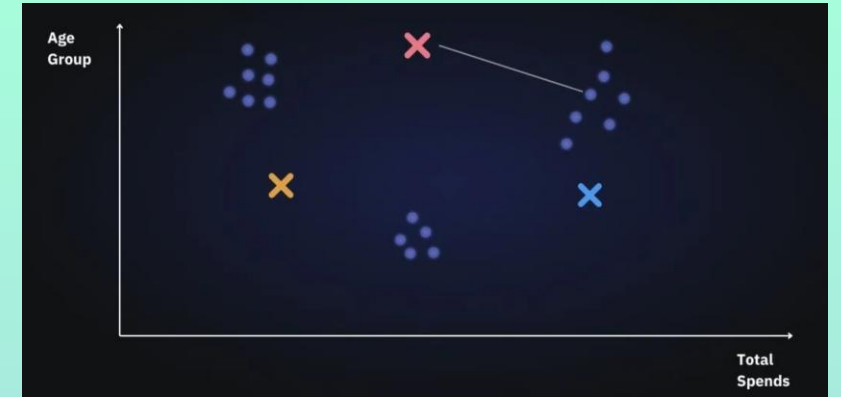
One of the popular distance metric used in K-Means algorithm is Euclidean distance.

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

Example

For two points $P=(2,3)$ and $Q=(5,7)$

$$d(P, Q) = \sqrt{(5-2)^2 + (7-3)^2} = \sqrt{3^2 + 4^2} = \sqrt{9+16} = \sqrt{25} = 5$$

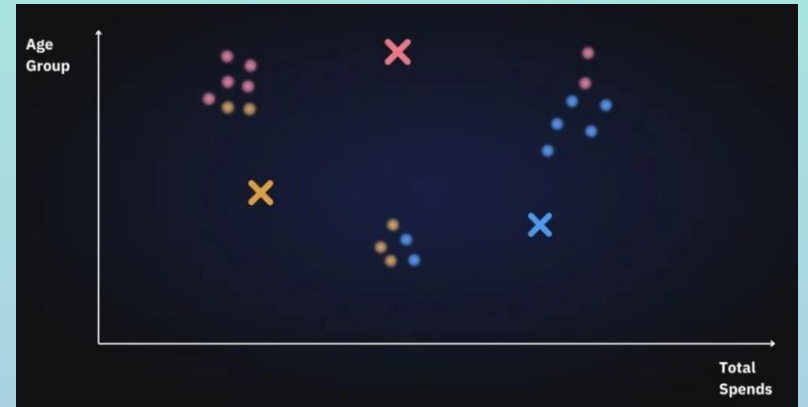


Unsupervised learning

K-Means Algorithm

Step 2 : Assignment

After calculating the distance between the datapoint and the centroid, and allocate the datapoints to the cluster where the distance between the datapoint and the cluster is minimum.



Unsupervised learning

K-Means Algorithm

3. Re-initialise centroids

In this steps the centroids are recalculated by averaging all datapoints in that cluster.

$$C = \frac{1}{N} \sum_{i=1}^n x_i$$

Example

Suppose a cluster has three data points in 2D space: $x_1=(2,3)$, $x_2=(4,5)$, $x_3=(6,7)$.

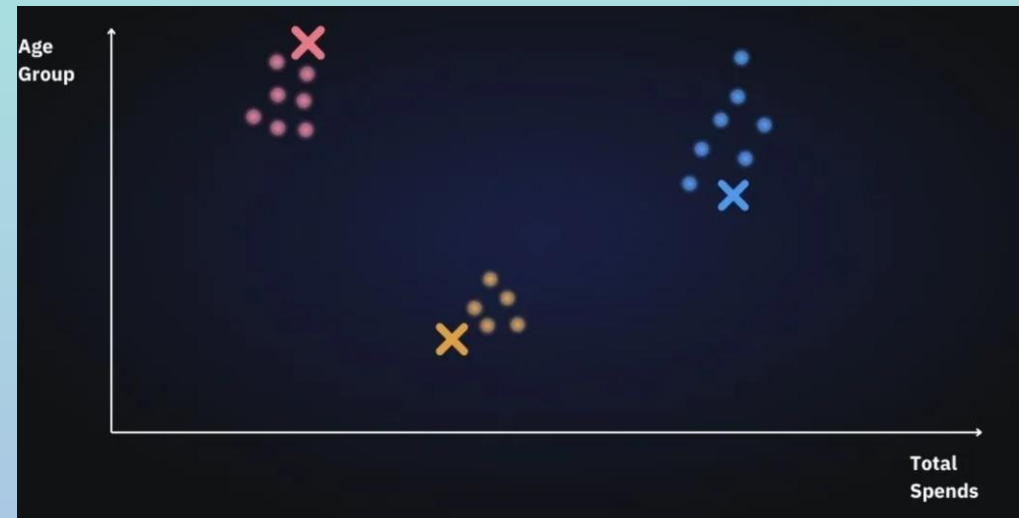
The centroid is: $C = \left(\frac{2+4+6}{3}, \frac{3+5+7}{3} \right) = \left(\frac{12}{3}, \frac{15}{3} \right) = (4,5)$

Unsupervised learning

K-Means Algorithm

Repeat steps 2 and 3

Repeat steps 2 and 3 until the centroids stabilize(no significant change) or a maximum number of iterations reached.



Unsupervised learning

K-Means

Demo in Python

Unsupervised learning

K-Means Clustering

Summary : K Means aim to create K clusters and assigns data points to them to ensure high intra-cluster similarity and low inter-cluster similarity. This is achieved by minimizing within cluster sum of square (WCSS) or inertia. Inertia is represented by

$$\sum_{i=1}^n \min_{\mu_j \in C} \|x_i - \mu_j\|^2$$

The algorithm begins by selecting random points (centroid) and then refine the initial centroid by perming multiple iterations. In each iteration, for each data point, the algorithm find the closest centroid and assign the datapoints to that cluster. Once all the points are assigned to the clusters, a new centroid is calculated by computing the mean of all the points (across all the dimensions). This process continues until there is no change of cluster centers or cluster assignments, or after a fixed number of iterations are over.

Unsupervised learning

K-Means Clustering – challenges

1. Centroid initialization :Due to the random centroid initialization, a good *convergence might not happen always.

*Convergence occurs when the centroids no longer change significantly or when a specified number of iterations is reached.

K-Means ++ is the solution do this issue. K-Means++ spread out the initial centroids.

1. Randomly select the first centroid from the data points.
2. For each data point compute its distance from the nearest, previously chosen centroid.
3. Select the next centroid from the data points such that the probability of choosing a point as centroid is directly proportional to its distance from the nearest, previously chosen centroid. (i.e. the point having maximum distance from the nearest centroid is most likely to be selected next as a centroid)
4. Repeat steps 2 and 3 until k centroids have been sampled

Unsupervised learning

K-Means Clustering – challenges

2. Number of clusters : Selecting an optimal number of clusters is a crucial step in K-Means algorithm.

The Elbow Method is a popular technique for selecting optimal k clusters.

The elbow point – the optimal k-value

1. Calculate WCSS for different K values: Iterate through a range of possible k values (number of clusters). For each k, perform k-means clustering and calculate the WCSS. This represents the total variance within each cluster.
2. Plot the WCSS vs. K: Plot the WCSS values on the y-axis and the number of clusters (k) on the x-axis.
3. Identify the elbow point: The ideal k value is identified as the “elbow” of the curve. This is the point where the WCSS starts to decrease at a much slower rate. In essence, you’re looking for the point where adding more clusters isn’t significantly improving the fit within the clusters.

Extra resources

[KMeans — scikit-learn 1.5.2 documentation](#)

[What is k-means clustering? | IBM](#)

[K-Means Clustering Explained](#)

References

- Carrasquilla, J., 2020. Machine learning for quantum matter. *Advances in Physics: X*, 5(1), p.1797528.
- Jordan, M.I. and Mitchell, T.M., 2015. Machine learning: Trends, perspectives, and prospects. *Science*, 349(6245), pp.255-260.