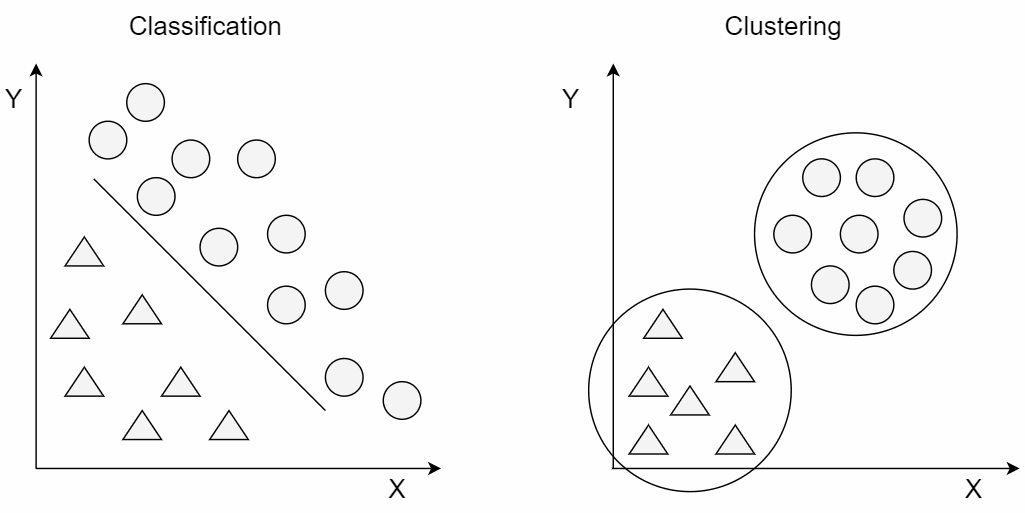
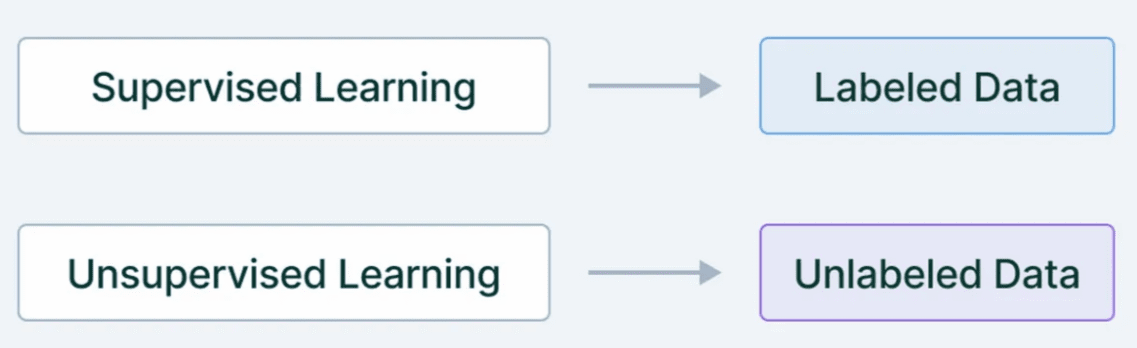
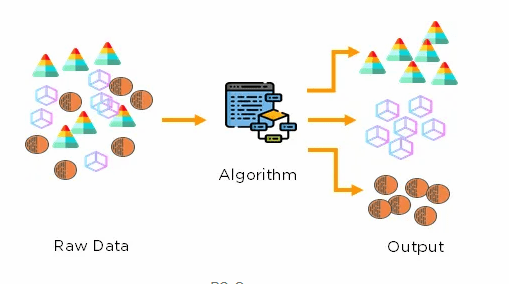
# Unsupervised Learning

## Overview of Unsupervised Learning

* Unsupervised learning involves models that learn patterns from untagged data.
* The primary goal is to discover underlying structures within datasets without pre-existing labels.
* It contrasts with supervised learning, where models are trained on labeled data.







## Key Concepts

* Features
* Training and Optimization
* Evaluation

## K-Means Clustering algorithm

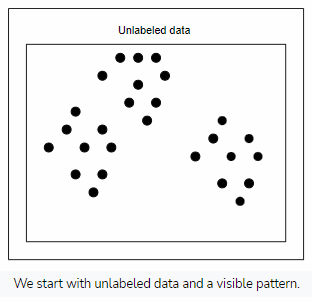
* Partitions data to k clusters
* Goal is to minimize cluster variance aka inertia.

## Assumption

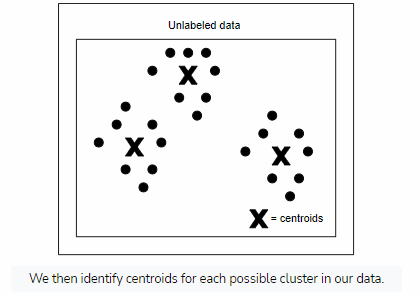
* clusters to be spherical, isotropic, and of equal size

## Process:

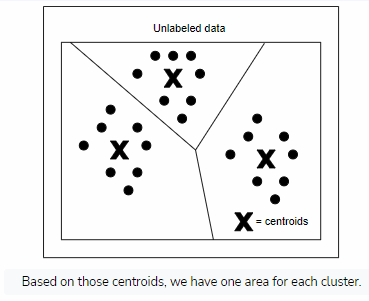
**Step1:**



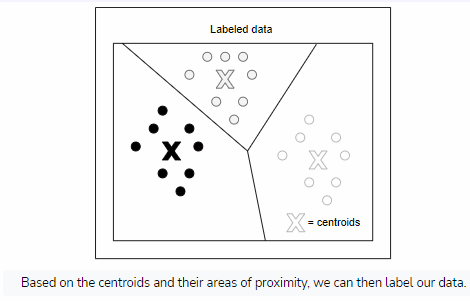
**Step2:**



**Step3:**

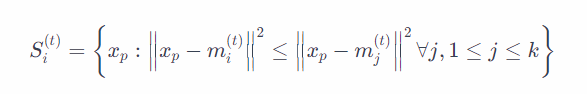


**Step4:**

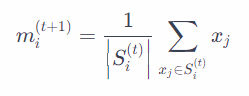


**Assignment Map Formula**

* Assign each observation to the cluster with the nearest mean, in most cases, it’s least squared Euclidean distance. Where each *xp* (data point) is assigned to one *St*.



* *Si*(*t*): This represents the cluster *i* at iteration *t*. It's a set that contains all the data points assigned to cluster *i* at a particular iteration.
* *xp*: This represents a data point in the dataset.
* *mi*(*t*): This represents the centroid of cluster *i* at iteration *t*. It's the mean (average) position of all the data points currently assigned to cluster *i*.
* ∥*xp*−*mi*(*t*)∥2: This is the Euclidean distance between the data point *xp* and the centroid *mi*(*t*) of cluster *i* at iteration *t*. It measures the straight-line distance between the data point and the centroid.
* ∀*j*,1≤*j*≤*k*: This means "for all *j* from 1 to *k*", where *k* is the total number of clusters. It indicates that we're considering each cluster centroid in the calculation.
* Recalculate means (cluster centroids) for data points assigned to each cluster.



## Example Scenario:

Suppose we have the following set of 2-dimensional data points:  
{(1,2),(2,3),(5,6),(6,7),(8,9),(10,11)}

And let's initialize the algorithm with 2 clusters and random initial centroids:  
*μ*1(1)=(1,2)  
*μ*2(1)=(5,6)

Step 1: Initialization

We randomly choose the initial centroids for the clusters.

Step 2: Assignment Step (E-step)

We calculate the distance between each data point and each centroid using the Euclidean distance formula. Then, we assign each data point to the nearest centroid.

Let's calculate the distances:

For data point (1, 2):

* Distance to centroid *μ*1(1): (1−1)2+(2−2)2=0
* Distance to centroid *μ*2(1): (1−5)2+(2−6)2=16+16=32≈5.66

So, (1, 2) is closer to *μ*1(1), hence it belongs to cluster 1.

Similarly, we calculate distances for other data points:

* (2, 3): Cluster 1
* (5, 6): Cluster 2
* (6, 7): Cluster 2
* (8, 9): Cluster 2
* (10, 11): Cluster 2

Step 3: Update Step (M-step)

We recalculate the centroids based on the data points assigned to each cluster.

For Cluster 1:

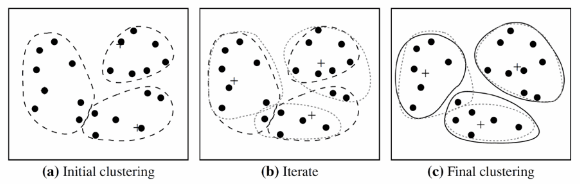
* New centroid: *μ*1(2)=2(1,2)+(2,3)=(21+2,22+3)=(1.5,2.5)

For Cluster 2:

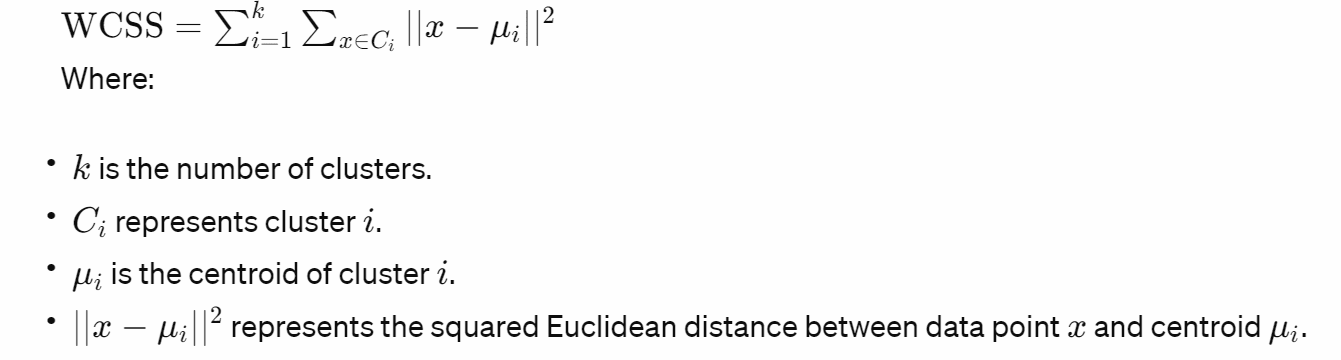
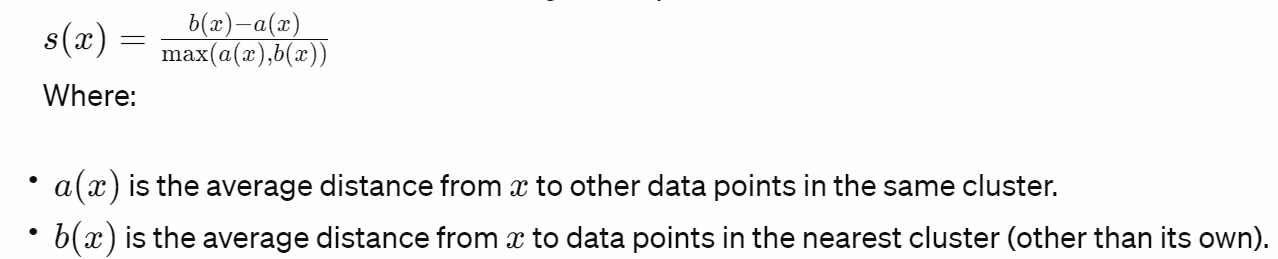
* New centroid: *μ*2(2)=4(5,6)+(6,7)+(8,9)+(10,11)=(45+6+8+10,46+7+9+11)=(7.25,8.25)

Step 4: Convergence Check

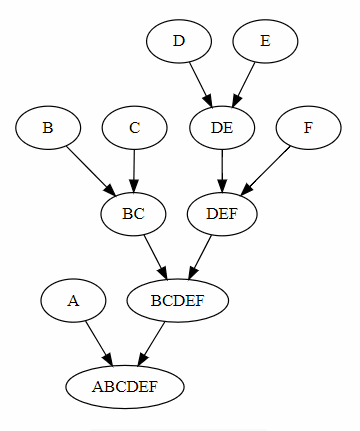
We compare the new centroids with the previous ones. If they are the same, the algorithm has converged. Otherwise, we repeat steps 2 and 3.



## Cluster Selection

* Elbow Method
  + The elbow method involves plotting the within-cluster sum of squares (WCSS) against the number of clusters (*k*). The WCSS is the sum of squared distances between each data point and its assigned centroid
  + 
* Silhoutte score
  + The silhouette score measures how similar an object is to its own cluster (cohesion) compared to other clusters (separation). It ranges from -1 to 1,
  + 

## Hierarchical clustering - Agglomerative

\

Suppose we have four data points A, B, C, and D, and their coordinates are as follows:

* A: (1, 1)
* B: (2, 2)
* C: (5, 5)
* D: (6, 6)

And the pairwise distance matrix using Euclidean distance is:

