Artificial Intelligence

Lab 7

K-Means
KNN

Agenda

- What is Machine Learning ?
- Supervised vs. Unsupervised Learning.
- Clustering.
- K-Means.
- KNN.
- Hands on.

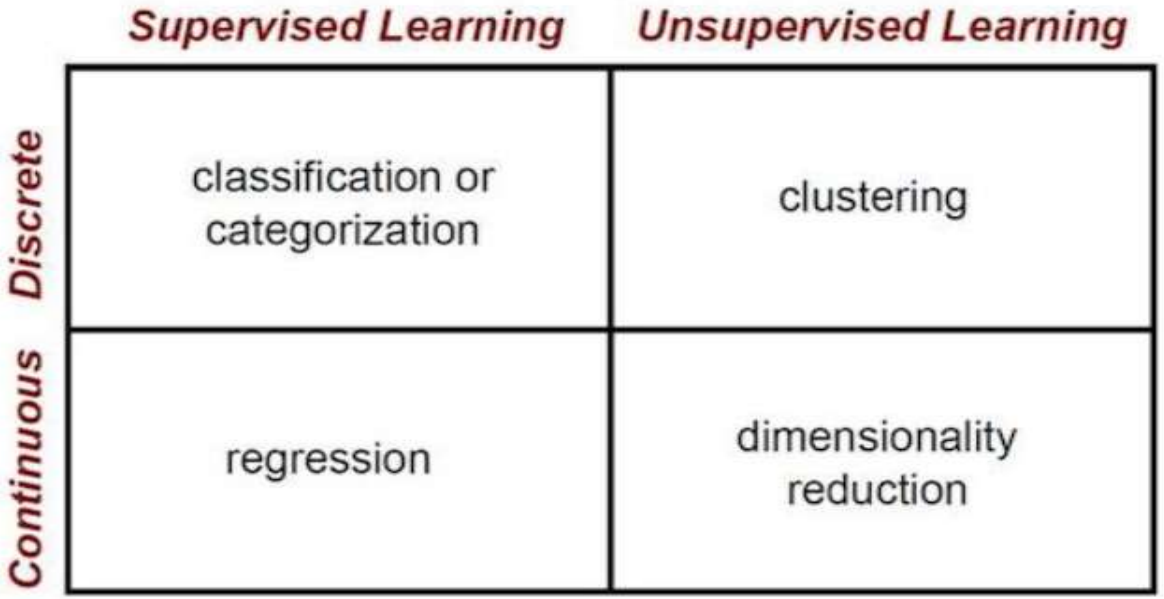
Machine Learning

- **Machine learning** is an application of artificial intelligence (AI) that provides systems the ability to automatically learn and improve from experience without being explicitly programmed.

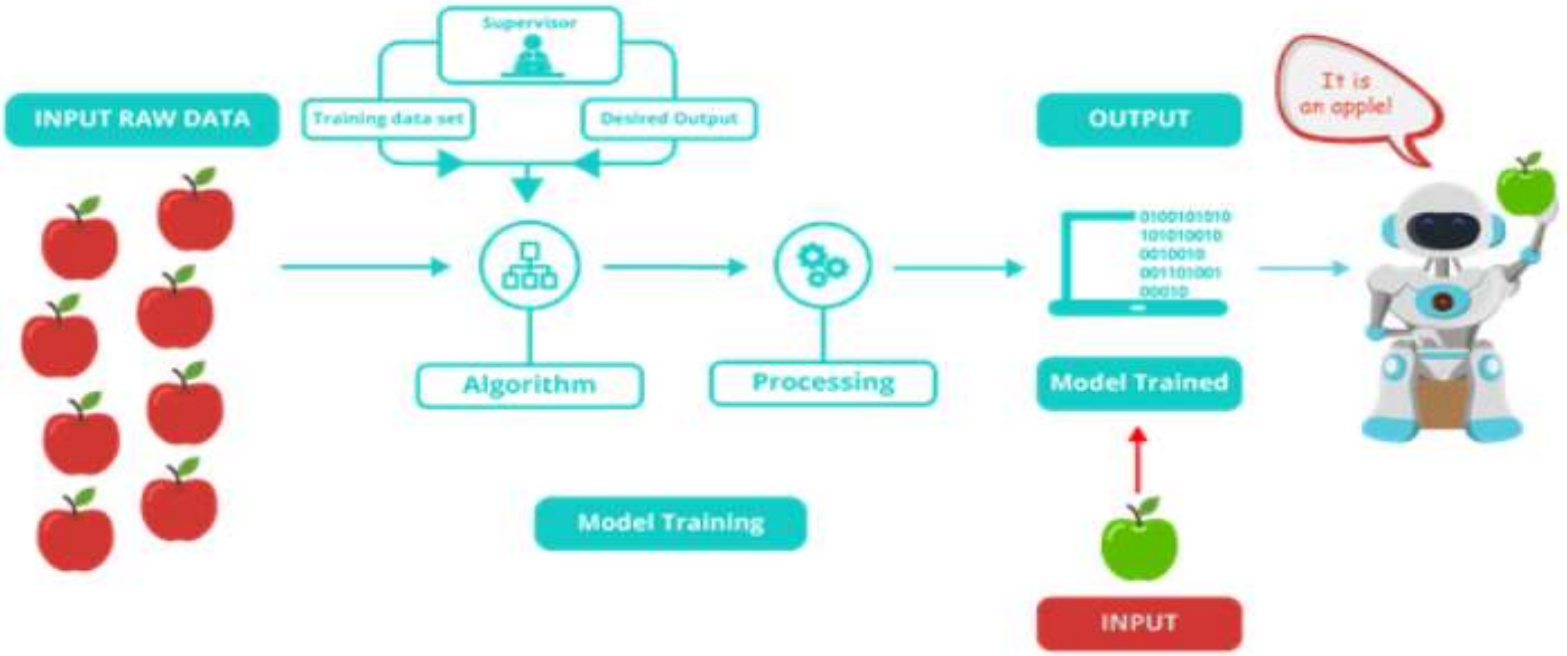
- **Machine learning** focuses on the development of computer programs that can access data and use it learn for themselves.

- Data scientists use many different kinds of machine learning algorithms to discover patterns in big data that lead to actionable insights

Machine Learning



Supervised learning



Supervised Learning

Supervised Learning

- Supervised learning requires that the algorithm's possible outputs are already known and that the data used to train the algorithm is already labeled with correct answers.

- It is called supervised learning because the process of an algorithm learning from the training dataset can be thought of as a teacher supervising the learning process.

- We know the correct answers, the algorithm iteratively makes predictions on the training data and is corrected by the teacher. Learning stops when the algorithm achieves an acceptable level of performance.

Supervised Learning

- Supervised learning is where you have input variables (x) and an output variable (Y) and you use an algorithm to learn the mapping function from the input to the output.

$$Y = f(X)$$

- The goal is to approximate the mapping function so well that when you have new input data (x) that you can predict the output variables (Y) for that data.

Supervised Learning

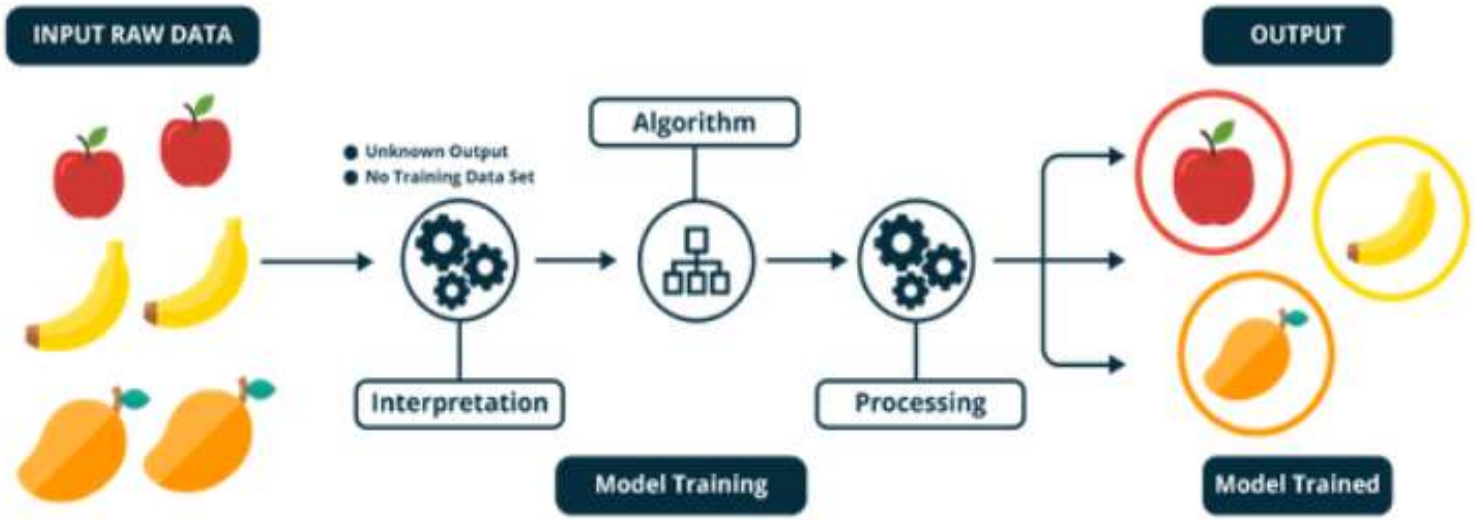
	Supervised Learning	Unsupervised Learning
Discrete	classification or categorization	clustering
Continuous	regression	dimensionality reduction

Classification problem	Regression problem
Output is a discrete value (category)	Output is a real value
Classify class	Estimate value
ex: we might be trying to predict whether someone likes pineapple (1) on their pizza or not (0) based on their age (the predictor).	ex: we could use the data in the table below to estimate someone's weight given their height.

Age	Likes Pinapple on Pizza
42	1
65	1
50	1
76	1
96	1
50	1
91	0

Height(Inches)	Weight(Pounds)
65.78	112.99
71.52	136.49
69.40	153.03

Unsupervised learning



Unsupervised Learning

Unsupervised Learning

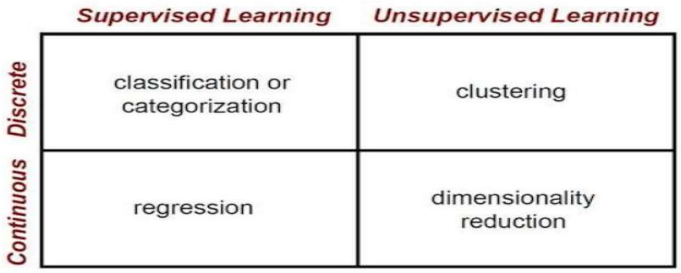
- Unsupervised machine learning is that a computer can learn to identify complex patterns without a human to provide guidance along the way.

- These are called unsupervised learning because unlike supervised learning above there is no correct answers and there is no teacher. Algorithms are left to their own devises to discover and present the interesting structure in the data.

Unsupervised Learning

- Unsupervised learning is where you only have input data (X) and no corresponding output variables.
- The goal for unsupervised learning is to model the underlying structure or distribution in the data in order to learn more about the data.

Unsupervised Learning



- **Clustering:** A clustering problem is where you want to discover the inherent groupings in the data, such as grouping customers by purchasing behavior.

- **Association**: An association rule learning problem is where you want to discover rules that describe large portions of your data, such as people that buy X also tend to buy Y.

- **Dimensionality Reduction:** is the process of reducing the number of random variables under consideration by obtaining a set of principal variables. It can be divided into feature selection and feature extraction.

Supervised vs Unsupervised

Supervised	Unsupervised
All data is labeled and the algorithms learn to predict the output from the input data.	All data is unlabeled and the algorithms learn to inherent structure from the input data
The goal is to approximate the mapping function so well that when you have new input data (x) that you can predict the output variables (Y) for that data.	The goal for unsupervised learning is to model the underlying structure or distribution in the data in order to learn more about the data.

Clustering

- **Clustering** is the task of dividing the population or data points into a number of groups such that data points in the same groups are more similar to other data points in the same group than those in other groups.

- A cluster refers to a collection of data points aggregated together because of certain similarities.

- Clustering is unsupervised learning approach.

Types of Clustering

- **Connectivity models:**
  - These models are based on the notion that the data points closer in data space exhibit more similarity to each other than the data points lying farther away.
  - Algorithms: hierarchical clustering algorithm.
- **Centroid models:**
  - These are iterative clustering algorithms in which the notion of similarity is derived by the closeness of a data point to the centroid of the clusters.
  - Algorithms: k-means algorithm
- **Density models:**
  - These models search the data space for areas of varied density of data points in the data space. It isolates various different density regions and assign the data points within these regions in the same cluster.
  - Algorithms: DBSCAN and OPTICS algorithm

K-Means

- You'll define a target number k, which refers to the number of centroids you 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 in-cluster sum of squares.

K-Means

- The **'K'** => identifies k number of centroids, and then allocates every data point to the nearest cluster, while keeping the centroids as small as possible.

- The **'means'** => in the K-means refers to averaging of the data; that is, finding the centroid.

K-Means

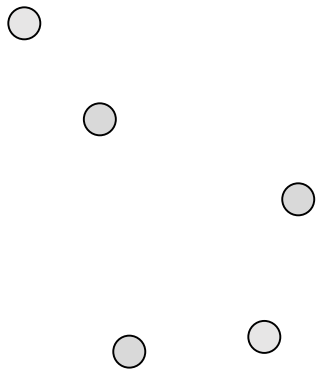
- The K-means algorithm in data mining starts with a first group of randomly selected centroids, which are used as the beginning points for every cluster, and then performs iterative (repetitive) calculations to optimize the positions of the centroids.

- It halts creating and optimizing clusters when either:
  - The centroids have stabilized—there is no change in their values because the clustering has been successful.
  - The defined number of iterations has been achieved.

K-Means

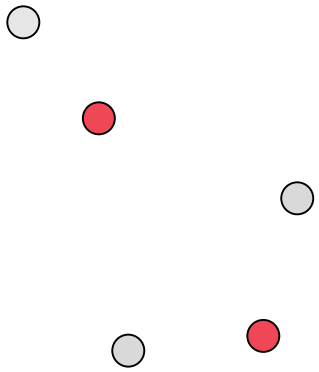
- The K means is an iterative clustering algorithm that aims to find local maxima in each iteration. This algorithm works in these 5 steps :

1. Specify the desired number of clusters K : Let us choose k=2 for these 5 data points in 2-D space.



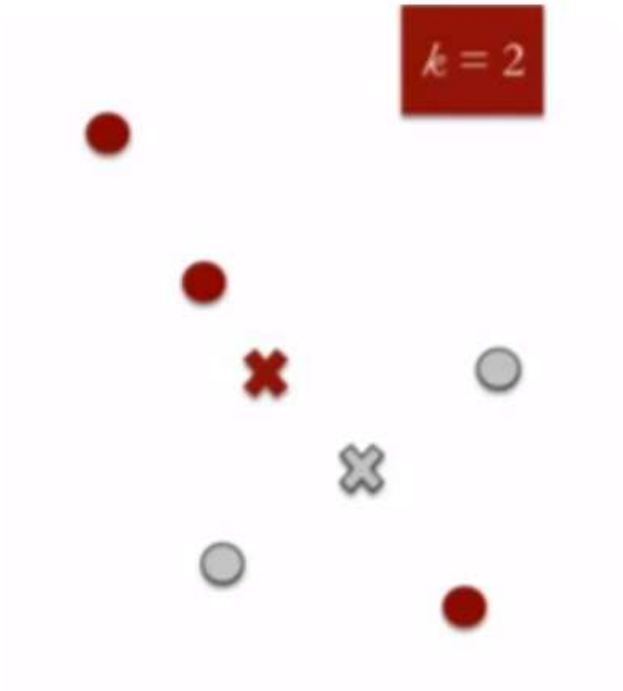
K-Means

2. Randomly choose k objects from the data points to be initial cluster center.



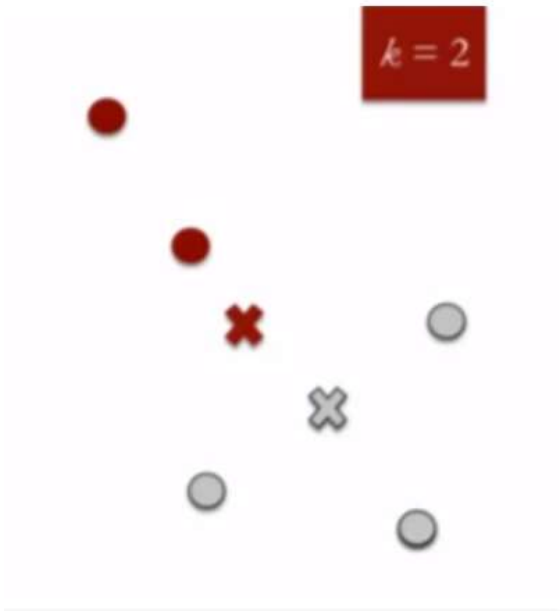
K-Means

3. Compute cluster centroids : The centroid of data points in the red cluster is shown using red cross and those in grey cluster using grey cross.



K-Means

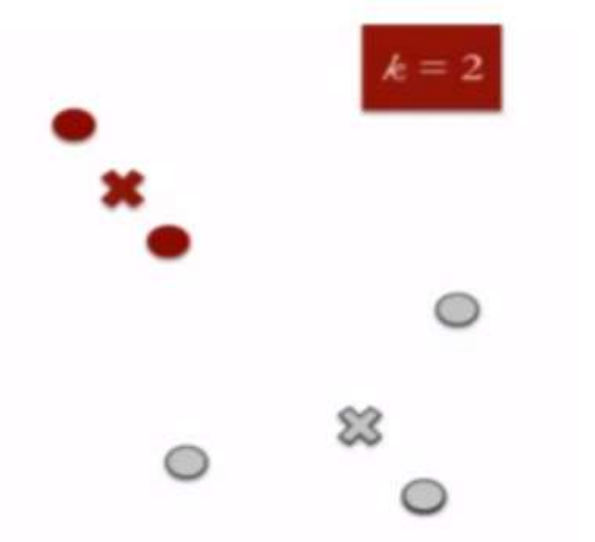
4. Re-assign each point to the closest cluster centroid : Note that only the data point at the bottom is assigned to the red cluster even though its closer to the centroid of grey cluster. Thus, we assign that data point into grey cluster.



K-Means

5. Re-compute cluster centroids : Now, re-computing the centroids for both the clusters.

6. Repeat steps 4 and 5 until no improvements are possible. Similarly, we'll repeat the 4th and 5th steps until we'll reach global optima. When there will be no further switching of data points between two clusters for two successive repeats. It will mark the termination of the algorithm if not explicitly mentioned.



K-Means

Input: k (the number of clusters),
D (a set of lift ratios)
Output: a set of k clusters
Method:
Arbitrarily choose k objects from D as the initial cluster centers;
Repeat:
1. (re)assign each object to the cluster to which the object is the most similar, based on the mean value of the objects in the cluster;
2. Update the cluster means, i.e., calculate the mean value of the objects for each cluster

Until no change;

KNN

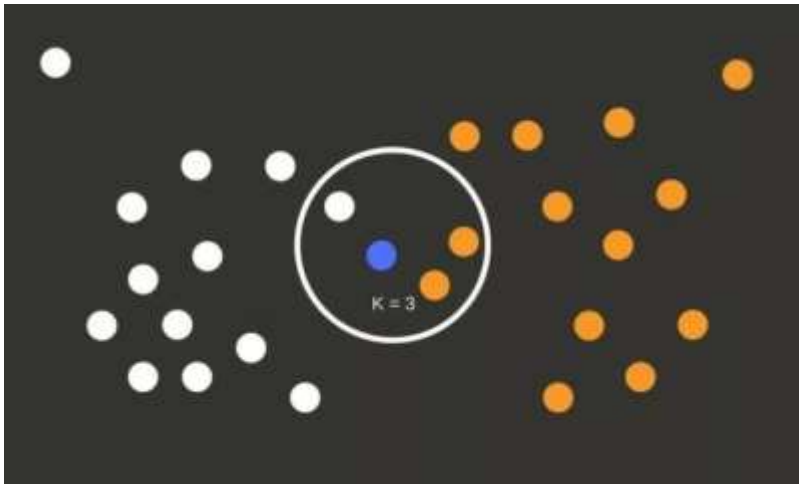
- The k-nearest neighbors (KNN) algorithm is a simple, easy-to-implement supervised machine learning algorithm that can be used to solve both classification and regression problems.

- The KNN algorithm assumes that similar things are near to each other.

- The simple version of the KNN classifier algorithms is to predict the target label by finding the nearest neighbor class. The closest class will be identified using the distance measures like Euclidean distance.

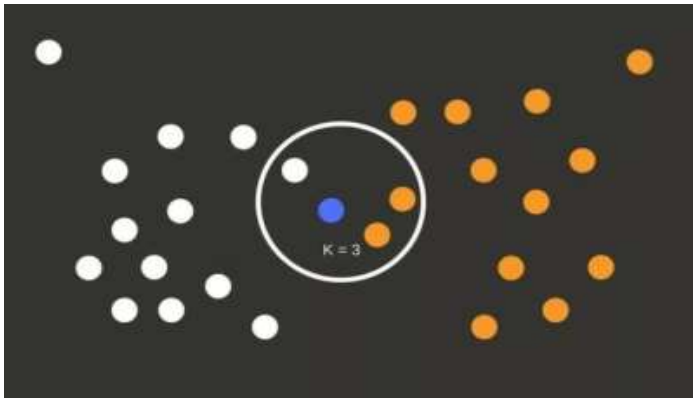
KNN

- We have two different target classes white & orange circles.
- We have total 26 training samples.
- We would like to predict the target class for the blue circle.
- Considering k value as three, we need to calculate the similarity distance using similarity measures like Euclidean distance.



KNN

- The first step is to calculate the distance(Euclidean) between the new data point and all the training data points.
- Arrange all the distances in ascending order.
- Now, we have K top distances. Let ki denotes no. of points belonging to the ith class among k points. If ki >kj ∀ i ≠ j then put x in class i.
- Nearest neighbor is a special case of k-nearest neighbor class.
  Where k value is 1 (k = 1).
  In this case, new data point target class will be assigned to the 1st closest neighbor.



KNN

1. Load the training and test data
2. Choose the value of K
3. For each point in test data:
   - find the Euclidean distance to all training data points
   - store the Euclidean distances in a list and sort it
   - choose the first k points
   - assign a class to the test point based on the majority of classes present in the chosen points
4. End

Hands on

- Install sklearn package to load iris dataset.

```
Iris plants dataset
--------------------

**Data Set Characteristics:**

    :Number of Instances: 150 (50 in each of three classes)
    :Number of Attributes: 4 numeric, predictive attributes and the class
    :Attribute Information:
        - sepal length in cm
        - sepal width in cm
        - petal length in cm
        - petal width in cm
        - class:
                - Iris-Setosa
                - Iris-Versicolour
                - Iris-Virginica
```

Hands on

```
from sklearn.datasets import load_iris

dataset = load_iris()

print(dataset.keys()) #dict_keys(['data',
                                 'target',
                                 'target_names',
                                 'DESCR',
                                 'feature_names',
                                 'filename'])


print(dataset['data'])   #[[5.1 3.5 1.4 0.2]
                         # [4.9 3.  1.4 0.2]
                         # [4.7 3.2 1.3 0.2] ....]
```

Hands on

- Input:
  - Iris Dataset
  - Num of Clusters = 3
  - Num of Iterations = 100
- Output:
  - Data points in each cluster.
  - Centroid of each cluster.

Questions?