Unit 5 — 12 hrs.

Machine Learning

5.1 Concept of Learning

5.2 Learning by Analogy, Inductive Learning, Explanation based Learning

5.3 Supervised Learning (Classification/Regression):
K-Nearest Neighbors (KNN), Naïve Bays, Logistic Regression, Support Vector Machine (SVM), Neural Networks

5.4 Unsupervised Learning:

5.4.1 Clustering, K-Means

5.4.2 Dimensionality Reduction [Principal Component Analysis (PCA), Linear Discriminant Analysis]

5.5 Reinforcement Learning

5.6 Fuzzy Learning

5.7 Boltzmann Machines

5.8 Deep Learning

5.1 Concept of Learning

- The *art* and *science* of:
  - Giving computers the ability to learn to make decisions from data
  - … without being explicitly programmed!
- Examples:
  - Learning to predict whether an email is spam or not
  - Clustering Wikipedia entries into different categories
- **Supervised learning:** Uses labeled data
- **Unsupervised learning:** Uses unlabeled data

Simple Definition I - Branch of Artificial Intelligence that gives computers to learn without being explicitly programmed.

Simple Definition II - Branch of Artificial Intelligence, about to construct a system that learn from data.

Actual Definition - A computer program is said to learn from **Experience** E with respect to some **Task** T and some **Performance** measure P, if its performance on T, as measured by P, improves with experience E.

Some more examples

1. Classification of e-mail into **SPAM** or **NON-SPAM**

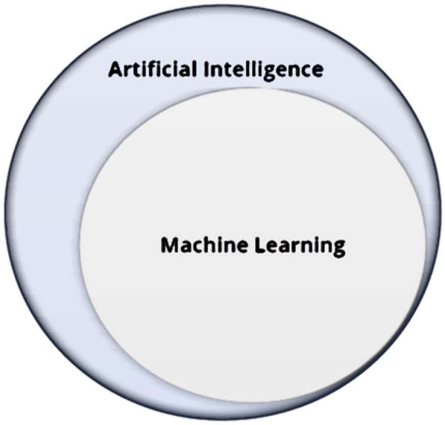
2. Classification of transaction as **FRAUD** or **GENUINE**

3. Patients diagnosed as **DIABETIC** or **NON-DIABETIC**

4. Gene Classification into **CODING** or **NON-CODING**.

…… Many more.

Machine Learning is an approach or subset of Artificial Intelligence that is based on the idea that machines can be given access to data along with the ability to learn from it.



The capability of Artificial Intelligence systems to learn by extracting patterns from data is known as Machine Learning.

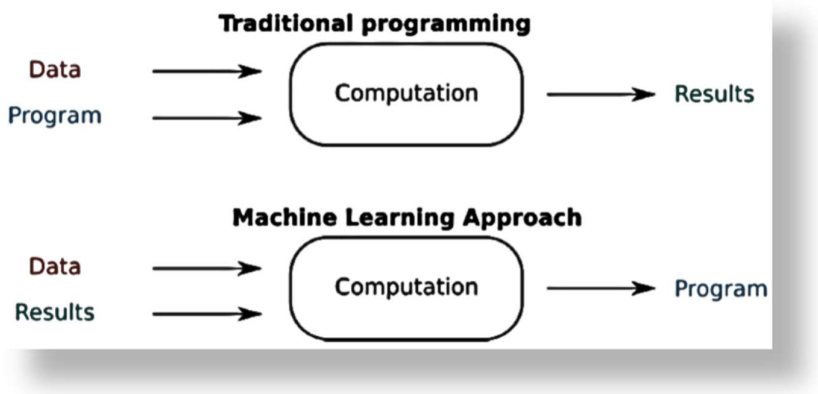
Machine learning is the science of getting computers to act without being explicitly programmed. Many researchers also think it is the best way to make progress towards human-level AI.

Machine learning is the category of algorithm that allows software applications to become more accurate in predicting outcomes without being explicitly programmed.

Machine learning is a sub set of Artificial Intelligence in the field of computer science that often uses statistical technique to give computers the ability to "learn" with data, without being explicitly programmed.

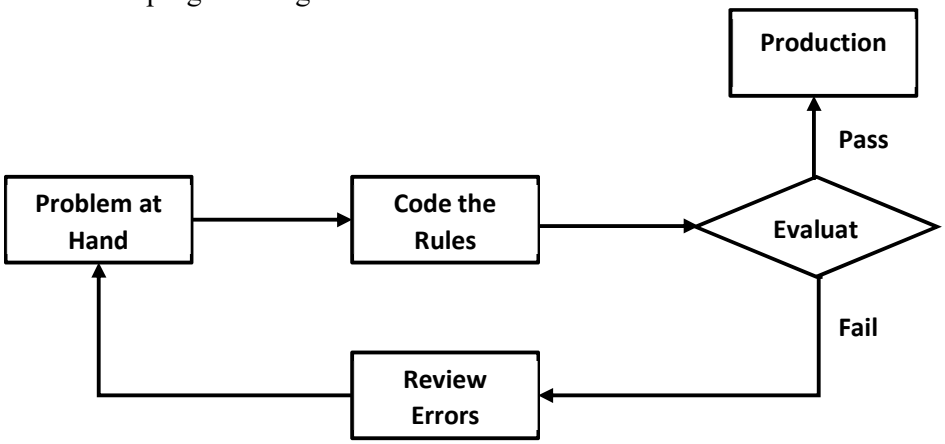
The name Machine learning was coined in 1959 by **-Arthur Samuel.**

Traditional Programming vs. Machine Learning Approach



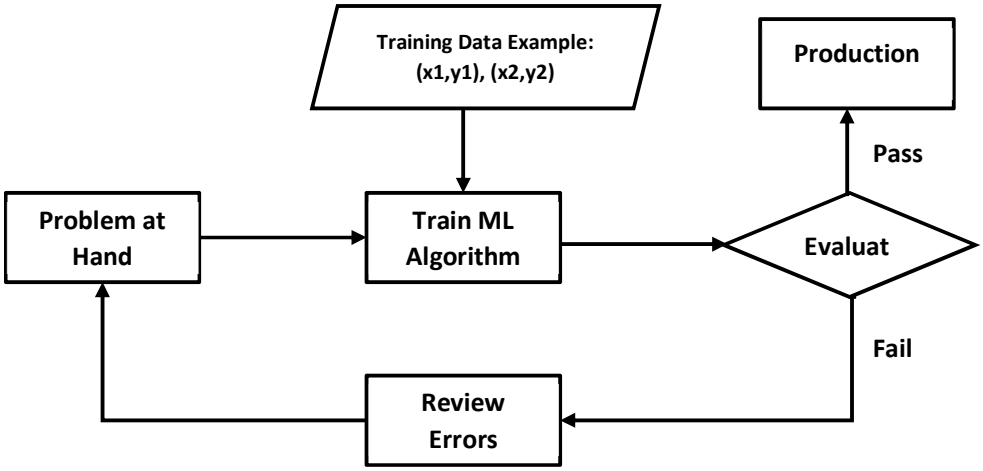
Traditional Approach

Traditional programming relies on hard-coded rules.



Machine Learning Approach

Machine Learning relies on learning patterns based on sample data.



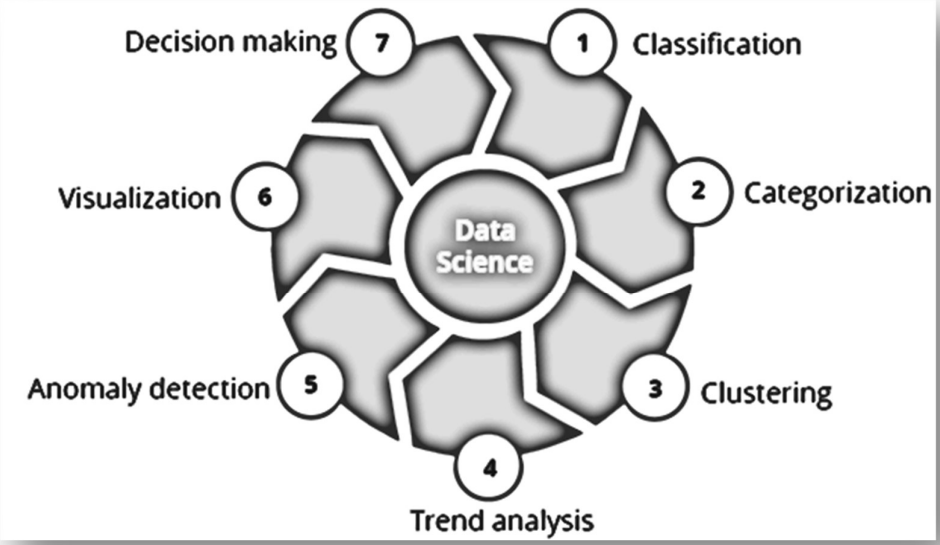
Relationship between Data Science and Machine Learning

Data Science and Machine Learning go hand in hand. Data Science helps evaluate data for Machine Learning algorithms.

- Data science is the use of statistical methods to find patterns in the data.
- Statistical machine learning uses the same math and techniques as data science.
- These techniques are integrated into algorithms that learn and improve on their own.
- Machine Learning facilitates Artificial Intelligence as it enables machines to learn from the patterns in data.

Machine Learning Techniques

Machine Learning uses a number of theories and techniques from Data Science:



Machine Learning Algorithms

Machine Learning can learn from **labelled data** (known as supervised learning) or unlabeled data (known as unsupervised learning).

Machine Learning algorithms involving **unlabeled data**, or unsupervised learning, are more complicated than those with the labelled data or supervised learning.

Machine Learning algorithms can be used to make decisions in subjective areas as well.

Examples:

- Logistic Regression can be used to predict which party will win at the ballots.
- Naïve Bayes algorithm can separate valid emails from spam.

Applications of Machine Learning

Applications of Machine Learning

Artificial intelligence and Machine learning are being increasingly used in various functions such as:

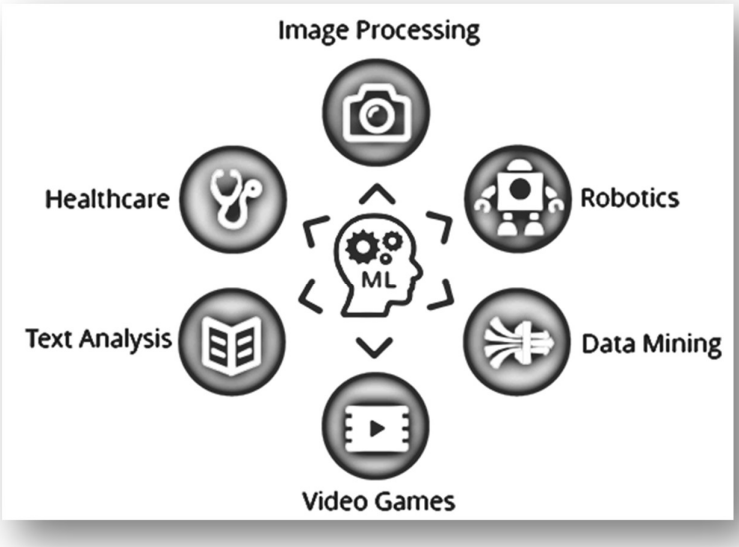
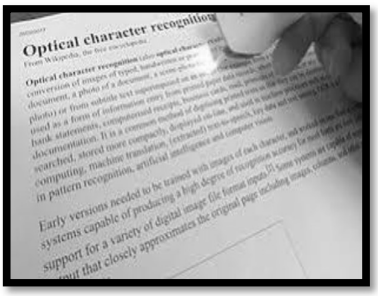
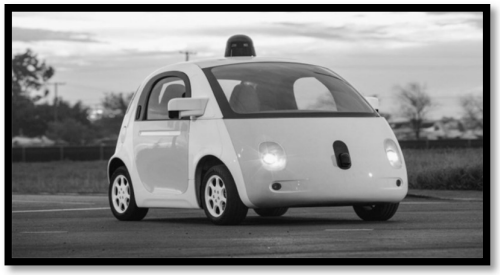


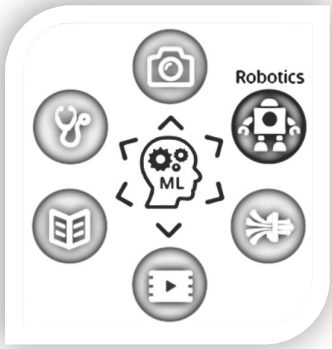
Image tagging and recognition



Optical Character Recognition (OCR)



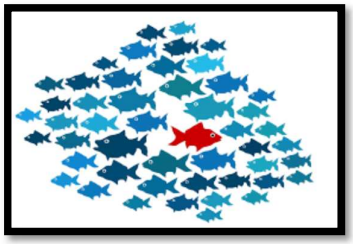
Self-driving cars



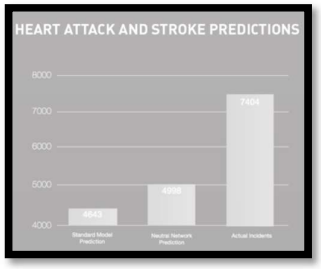
Human simulation

Robert

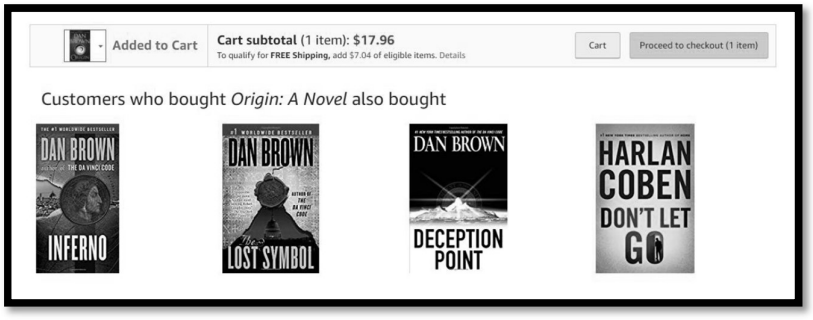
Industrial robotics



Anomaly detection

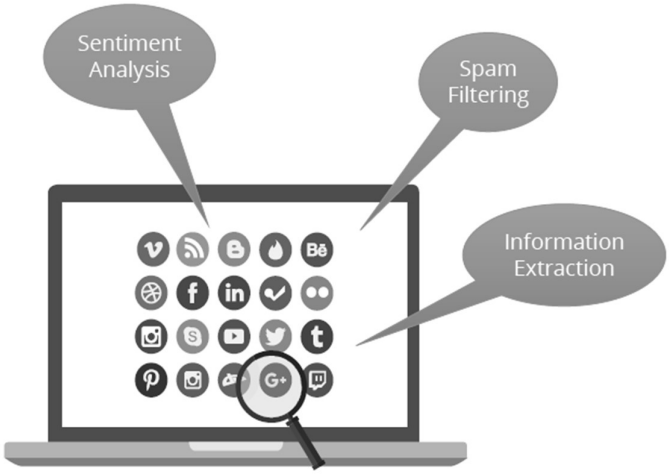


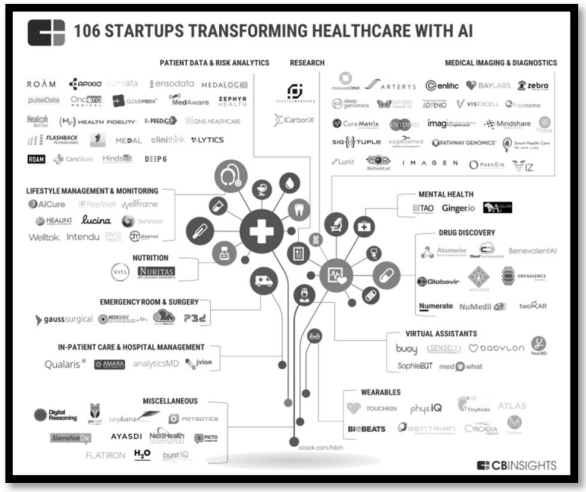
Grouping and Predictions



Association rules

Some games implement reinforcement learning





5.2 Learning by Analogy, Inductive Learning, Explanation based Learning

Learning:

Learning is one of those everyday terms which is broadly and vaguely used in English language.

- Learning is making useful changes in our minds.
- Learning is constructing or modifying representations of what is being experienced.
- Learning is the phenomenon of knowledge acquisition in the absence of explicit programming.

Herbert Simon, 1983

Learning denotes changes in the system that are adaptive in the sense that they enable the system to do the system to do the same task or tasks drawn from the same population more efficiently and more effectively next time.

Learning involves the recognition of patterns in data or experience and then using that information to improve performance on another task.

Learning involves 3 factors:

Changes: Learning changes the learner:

For machine learning the problem is determining the nature of these changes and how to best represent them.

Generalization: Learning leads to generalization:

Performance must improve not only on the same task but on similar tasks.

Improvement: Learning leads to improvement:

Machine learning must address the possibility that changes may degrade performance and find ways to prevent it.

Learning Methods:

There are two different kinds of information processing which must be considered in a machine learning system.

a. Inductive Learning
b. Deductive Learning

Inductive Learning:

Inductive learning is concerned with determining general patterns, organizational schemes, rules, and laws from raw data, experience or examples.

Deductive Learning:

Deductive learning is concerned with determining specific facts using general rules or that determination of new general rules from old general rules.

Learning by Analogy:

- Analogical reasoning is a powerful mechanism for exploiting past experience in planning and problem solving.
- An analogical transformation process is developed to extract knowledge from past successful problem-solving situations that bear a strong similarity to the current problem.
- Learning by analogy generally involves abstracting details from a particular set of problems and resolving structural similarities between previously distinct problems.
- Analogical learning refers to this process of recognition and then applying the solution from the known problem to the new problem.
- Such a technique is often identified as case-based reasoning (CBR).
- Analogical learning generally involves developing a set of mappings between features of two instances.

Inductive Learning:

Inductive Learning Algorithm (ILA) is an iterative and inductive machine learning algorithm which is used for generating a set of a classification rule, which produces rules of the form "IF-THEN", for a set of examples, producing rules at each iteration and appending to the set of rules.

This involves the process of learning by example -- where a system tries to induce a general rule from a set of observed instances.

Explanation Based Learning:

Explanation-Based Learning (EBL) is a principled method for exploiting available domain knowledge to improve supervised learning. Improvement can be in speed of learning, confidence of learning, accuracy of the learned concept, or a combination of these. In modern EBL the domain theory represents an expert's approximate knowledge of complex systematic world behavior. It may be imperfect and incomplete. Inference over the domain knowledge provides analytic evidence that compliments the empirical evidence of the training data. By contrast, in original EBL the domain theory is required to be much stronger; inferred properties are guaranteed. Another important aspect of modern EBL is the interaction between domain knowledge and labeled training examples afforded by explanations.

An EBL accepts 4 kinds of input:

A training examples
-- what the learning sees in the world.

A goal concepts
-- a high level description of what the program is supposed to learn.

An operational criterion
-- a description of which concepts are usable.

A domain theory
-- a set of rules that describe relationships between objects and actions in a domain.

From this EBL computes a generalization of the training example that is sufficient not only to describe the goal concept but also satisfies the operational criterion.

This has two steps:

Explanation
-- the domain theory is used to prune away all unimportant aspects of the training example with respect to the goal concept.

Generalization
-- the explanation is generalized as far possible while still describing the goal concept.

5.3 Supervised Learning (Classification/Regression)

What is Supervised Learning?

Supervised Learning is a type of Machine Learning used to learn models from labeled training data. It allows us to predict output for future or unseen data.

In Supervised learning, you train the machine using data which is well "labeled." It means some data is already tagged with the correct answer. It can be compared to learning which takes place in the presence of a supervisor or a teacher.

A supervised learning algorithm learns from labeled training data, helps you to predict outcomes for unforeseen data. Successfully building, scaling, and deploying accurate supervised machine learning Data science model takes time and technical expertise from a team of highly skilled data scientists. Moreover, Data scientist must rebuild models to make sure the insights given remains true until its data changes.

Why Supervised Learning?

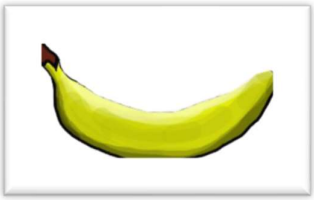
- Supervised learning allows you to collect data or produce a data output from the previous experience.
- Helps you to optimize performance criteria using experience.
- Supervised machine learning helps you to solve various types of real-world computation problems.

Supervised learning as the name indicates the presence of a supervisor as a teacher. Basically, supervised learning is a learning in which we teach or train the machine using data which is well labeled that means some data is already tagged with the correct answer. After that, the machine is provided with a new set of examples (data) so that supervised learning algorithm analyses the training data (set of training examples) and produces a correct outcome from labeled data.

For instance, suppose you are given a basket filled with different kinds of fruits. Now the first step is to train the machine with all different fruits one by one like this:

- If shape of object is rounded and depression at top having color Red then it will be labelled as –**Apple**.
- If shape of object is long curving cylinder having color Green-Yellow then it will be labelled as –**Banana**.

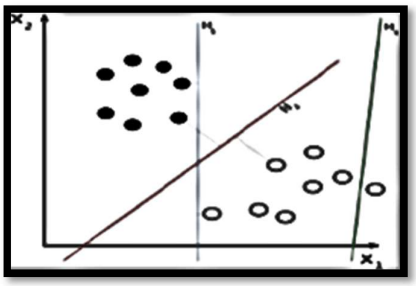
Now suppose after training the data, you have given a new separate fruit say Banana from basket and asked to identify it.



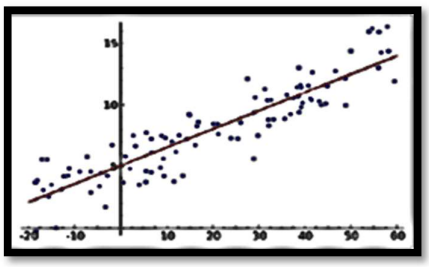
Since the machine has already learned the things from previous data and this time have to use it wisely. It will first classify the fruit with its shape and color and would confirm the fruit name as BANANA and put it in Banana category. Thus, the machine learns the things from training data (basket containing fruits) and then apply the knowledge to test data (new fruit).

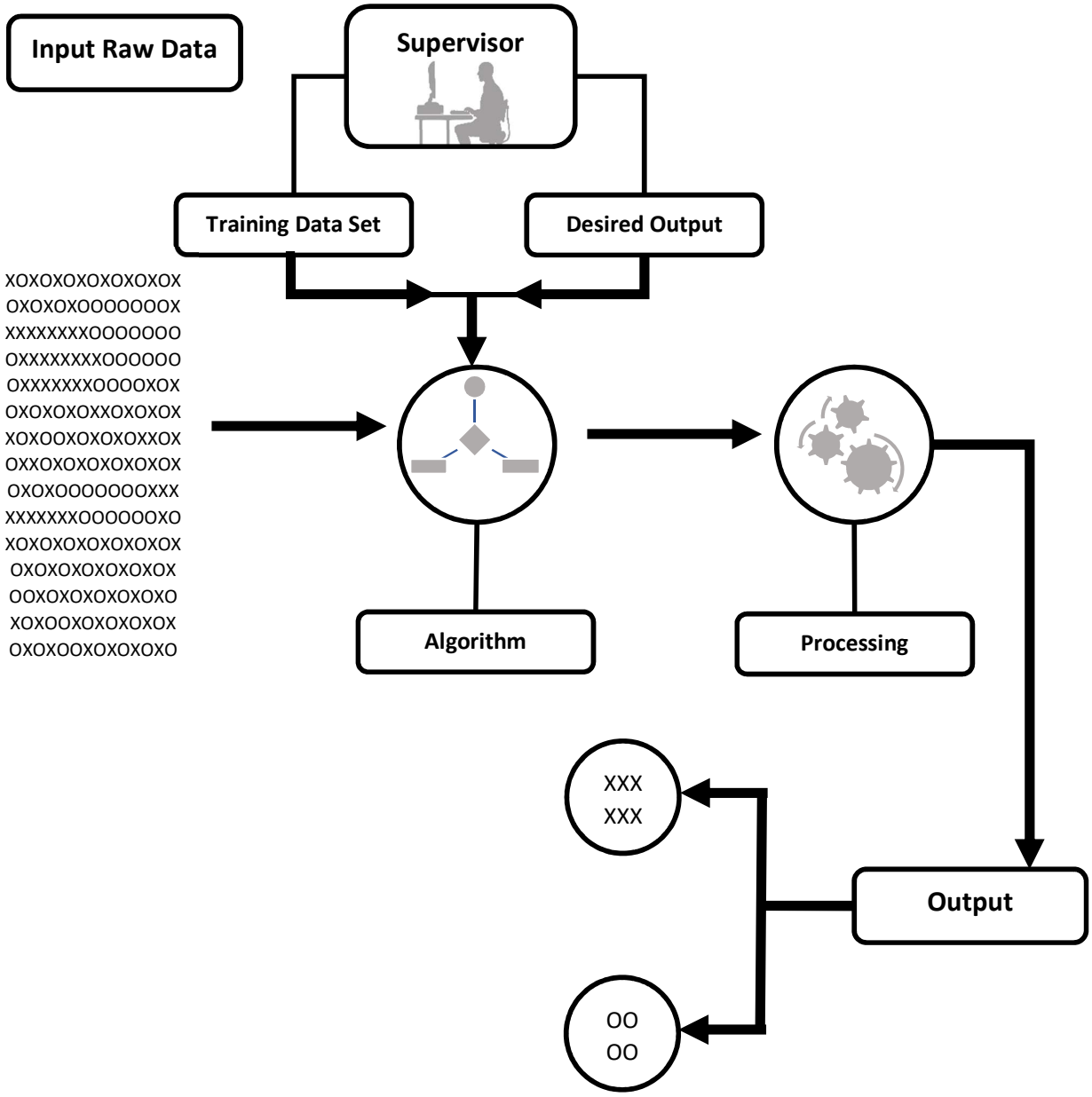
Supervised learning classified into two categories of algorithms:

- **Classification**: A classification problem is when the output variable is a category, such as "Red" or "Blue" or "Disease" and "No disease" or "Pass" and "Fail", "Spam" and "Not spam".
    - Answers **"What class?"**
    - Applied when the output has finite and discreet values



- **Regression**: A regression problem is when the output variable is a real value, such as "dollars" or "weight".
    - Answers **"How much?"**
    - Applied when the output is a continuous number
    - A simple regression algorithm: y = wx + b.





Amazon uses supervised learning algorithms to predict what items the user may like based on the purchase history of similar classes of users.

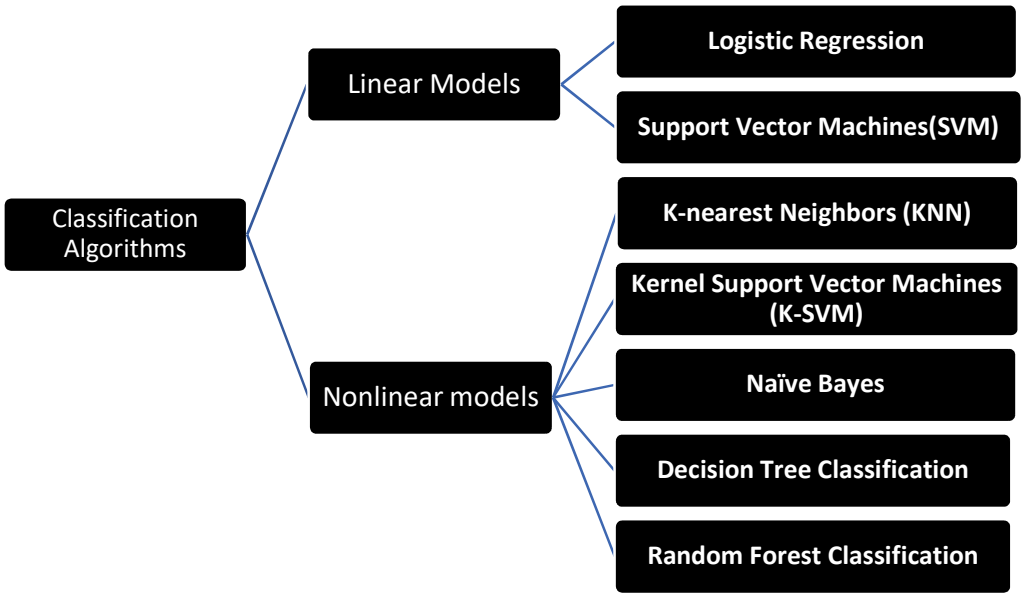
Classification

Classification problems categorize all the variables that form the output. Examples of these categories formed through classification would include demographic data such as marital status, sex, or age. The most common model used for this type of service status is the support vector machine. The support vector machines set forth to define the linear decision boundaries.

- Classification is a type of supervised learning.
- It specifies the class to which data elements belong to.
- It is best used when the output has finite and discreet values.
- It predicts a class for an input variable.

There are 2 types of classification, binomial and multi-class.

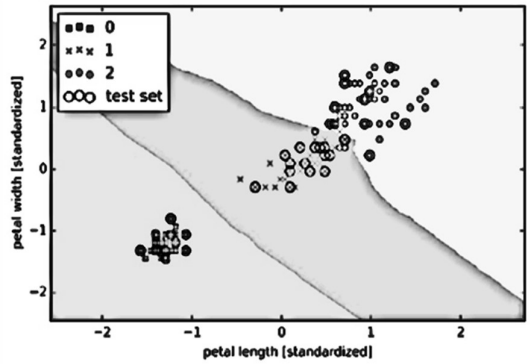
Types of Classification Algorithms



1. K-Nearest Neighbors (KNN):

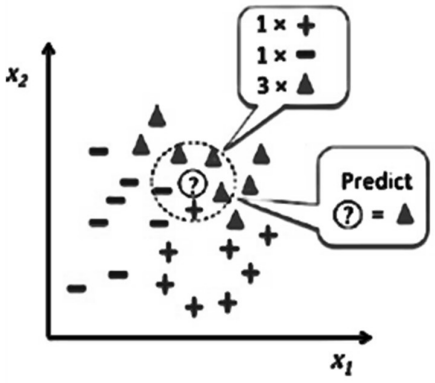
- K-nearest Neighbors algorithm is used to assign a data point to clusters based on similarity measurement.
- It uses supervised method for classification.

The steps to write a k-means algorithm are as given below:



- Choose the number of k and a distance metric. (k = 5 is common)
- Find k-nearest neighbors of the sample that you want to classify.
- Assign the class label by majority vote.

KNN Classification



A new input point is classified in the category such that it has the greatest number of neighbors from that category.

For example:

- Classify a patient as high risk or low risk.
- Mark email as spam or ham.

KNN can be used for both classification and regression predictive problems. However, it is more widely used in classification problems in the industry. To evaluate any technique, we generally look at 3 important aspects:

1. Ease to interpret output

2. Calculation time

3. Predictive Power

We can implement a KNN model by following the below steps:

1. Load the data
2. Initialize the value of k
3. For getting the predicted class, iterate from 1 to total number of training data points.
4. Calculate the distance between test data and each row of training data. Here we will use **Euclidean distance** as our distance metric since it's the most popular method.
5. Sort the calculated distances in ascending order based on distance values.
6. Get top k rows from the sorted array.
7. Get the most frequent class of these rows.
8. Return the predicted class

Python:

```
import pandas as pd          // pandas to read the datasets

from sklearn.cross_validation import train_test_split    // Older Version {Split into train | test}

from sklearn.model_selection import train_test_split    // Newer Version

df=pd.read_csv("iris_dataset.csv")    // Load the Dataset

X_train,X_test,Y_train,Y_test=train_test_split(X,Y,test_size=0.20)    {Data is splited into 80:20}

from sklearn.neighbors import KNeighborsClassifier      //KNN package

knn = KNeighborsClassifier(n_neighbors = 5)          //K=5

knn.fit(X_train, Y_train)                    //We always fit train data to predicate the new data

train_score= knn.score(X_train, Y_train)  //Score is the value to describe the algorithm for the given
test_score= knn.score(X_test, Y_test)        split the data sets data set ranging score from
                                            (>0 and <=1). This will generate the result on the basis
                                          of datasets and how you

print("The accuracy of the Knn classifier on training data is {:f}".format(train_score))

print("The accuracy of the Knn classifier on test data is {:f}".format(test_score))
```

2. Naïve Bays:

- According to Bayes model, the conditional probability P(Y|X) can be calculated as: P(Y|X) = P(X|Y)*P(Y) / P(X)
- This means you have to estimate a very large number of P(X|Y) probabilities for a relatively small vector space X.
- For example, for a Boolean Y and 30 possible Boolean attributes in the X vector, you will have to estimate 3 billion probabilities P(X|Y).
- To make it practical, a Naïve Bayes classifier is used, which assumes conditional independence of P(X) to each other, with a given value of Y.
- This reduces the number of probability estimates to 2*30=60 in the above example.

How Naive Bayes algorithm works?

Let's understand it using an example. Below I have a training data set of weather and corresponding target variable 'Play' (suggesting possibilities of playing). Now, we need to classify whether players will play or not based on weather condition. Let's follow the below steps to perform it.

Step 1: Convert the data set into a frequency table

Step 2: Create Likelihood table by finding the probabilities like Overcast probability = 0.29 and probability of playing is 0.64.

Weather	Play
Sunny	No
Overcast	Yes
Rainy	Yes
Sunny	Yes
Sunny	Yes
Overcast	Yes
Rainy	No
Rainy	No
Sunny	Yes
Rainy	Yes
Sunny	No
Overcast	Yes
Overcast	Yes
Rainy	No

Frequency Table		
Weather	**No**	**Yes**
Overcast	-	4
Rainy	3	2
Sunny	2	3
Grand Total	**5**	**9**

Likelihood Table				
Weather	**No**	**Yes**		
Overcast	-	4	=4/14	0.29
Rainy	3	2	=5/14	0.36
Sunny	2	3	=5/14	0.36
Grand Total	**5**	**9**		
	=5/14	=9/14		
	0.36	0.64		

Step 3: Now, use Naive Bayesian equation to calculate the posterior probability for each class. The class with the highest posterior probability is the outcome of prediction.

Problem: Players will play if weather is sunny. Is this statement is correct?

We can solve it using above discussed method of posterior probability.

P(Yes | Sunny) = P(Sunny | Yes) * P(Yes) / P (Sunny)

Here we have,

P (Sunny |Yes) = 3/9 = 0.33,
P(Sunny) = 5/14 = 0.36,
P(Yes)= 9/14 = 0.64

Now, P (Yes | Sunny) = 0.33 * 0.64 / 0.36 = 0.60, which has higher probability.

Naive Bayes uses a similar method to predict the probability of different class based on various attributes. This algorithm is mostly used in text classification and with problems having multiple classes.

Python:

```
import pandas as pd          // pandas to read the datasets
from sklearn.cross_validation import train_test_split    // Older Version {Split into train | test}
from sklearn.model_selection import train_test_split    // Newer Version
df=pd.read_csv("iris_dataset.csv")    // Load the Dataset
X_train,X_test,Y_train,Y_test=train_test_split(X,Y,test_size=0.20)    {Data is splited into 80:20}
from sklearn.naive_bayes import GaussianNB   //package for naïve bays
nb = GaussianNB()
nb.fit(X_train, Y_train)
train_score= nb.score(X_train, Y_train)
test_score= nb.score(X_test, Y_test)
print("The accuracy of the Gaussian NB on training data is {:f}".format(train_score))
print("The accuracy of the Gaussian NB on test data is {:f}".format(test_score))
```

What are the Pros and Cons of Naive Bayes?

Pros:

- It is easy and fast to predict class of test data set. It also performs well in multi class prediction
- When assumption of independence holds, a Naive Bayes classifier performs better compare to other models like logistic regression and you need less training data.
- It performs well in case of categorical input variables compared to numerical variable(s). For numerical variable, normal distribution is assumed (bell curve, which is a strong assumption).

Cons:

- If categorical variable has a category (in test data set), which was not observed in training data set, then model will assign a 0 (zero) probability and will be unable to make a prediction. This is often known as "Zero Frequency". To solve this, we can use the smoothing technique. One of the simplest smoothing techniques is called Laplace estimation.
- On the other side naive Bayes is also known as a bad estimator, so the probability outputs from predict_proba are not to be taken too seriously.
- Another limitation of Naive Bayes is the assumption of independent predictors. In real life, it is almost impossible that we get a set of predictors which are completely independent.

Applications of Naive Bayes Algorithms

- Real Time Prediction
- Multi Class Prediction
- Text Classification/ Spam Filtering/ Sentiment Analysis
- Recommendation System

3. Logistic Regression:

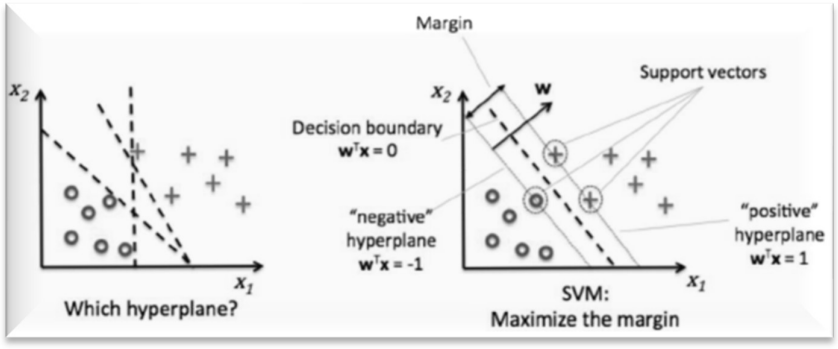
- This refers to a regression model that is used for classification.
- Logistic regression is a supervised learning algorithm because it uses true labels for training. Supervised learning algorithm should have **input variables (X)** and a **target variable (Y)** when you train the model.
- Logistic regression is a classification algorithm, don't confuse with the name regression.
- This method is widely used for binary classification problems. It can also be extended to multi-class classification problems.
- Here, the dependent variable is categorical: **y ϵ {0, 1}**
- A binary dependent variable can have only two values, like 0 or 1, win or lose, pass or fail, healthy or sick, etc.
- In this case, you model the probability distribution of output y as 1 or 0. This is called as sigmoid probability (σ).
- The probability in the logistic regression is often represented by the Sigmoid function (also called the logistic function or the S-curve):

$$S(t) = \frac{1}{1 + e^{-t}}$$

Python:

```
import pandas as pd          // pandas to read the datasets
from sklearn.cross_validation import train_test_split    // Older Version {Split into train | test}
from sklearn.model_selection import train_test_split    // Newer Version
df=pd.read_csv("iris_dataset.csv")    // Load the Dataset
X_train,X_test,Y_train,Y_test=train_test_split(X,Y,test_size=0.20)     {Data is splited into 80:20}

from sklearn.linear_model import LogisticRegression
lr = LogisticRegression()
lr.fit(X_train, Y_train)
print("The accuracy of the Logistic Regression on training data is {:f}".format(lr.score(X_train, Y_train)))
print("The accuracy of the Logistic Regression on test data is {:f}".format(lr.score(X_test, Y_test)))
```

4. Support Vector Machine (SVM):



- SVMs are classification algorithms used to assign data to various classes. They involve detecting hyperplanes which segregate data into classes.
- SVMs are very versatile and are also capable of performing linear or nonlinear classification, regression, and outlier detection.
- Once ideal hyperplanes are discovered, new data points can be easily classified.
- The optimization objective is to find "maximum margin hyperplane" that is farthest from the closest points in the two classes (these points are called support vectors).
- In the given figure, the middle line represents the hyperplane.

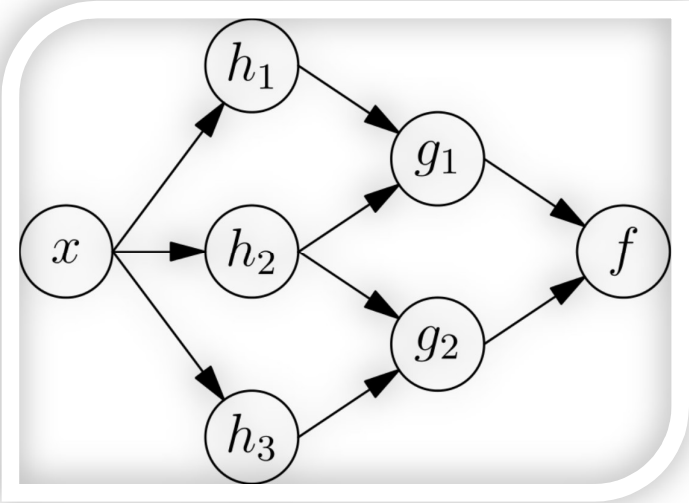
Python:

```
import pandas as pd          // pandas to read the datasets
from sklearn.cross_validation import train_test_split    // Older Version {Split into train | test}
from sklearn.model_selection import train_test_split    // Newer Version
df=pd.read_csv("iris_dataset.csv")    // Load the Dataset
X_train,X_test,Y_train,Y_test=train_test_split(X,Y,test_size=0.20)    {Data is splited into 80:20}
from sklearn.svm import SVC
svm = SVC()
svm.fit(X_train, Y_train)
print("The accuracy of the SVM classifier on training data is {:f}".format(svm.score(X_train, Y_train)))
print("The accuracy of the SVM classifier on test data is {:f}".format(svm.score(X_test, Y_test)))
```

5. Neural Networks:

- Neural networks are one of those cool words that are often used to lend credence to research.
- Neural nets take inspiration from the learning process occurring in human brains.
- They consist of an artificial network of functions, called parameters, which allows the computer to learn, and to fine tune itself, by analyzing new data.
- Each parameter, sometimes also referred to as neurons, is a function which produces an output, after receiving one or multiple inputs.
- Those outputs are then passed to the next layer of neurons, which use them as inputs of their own function, and produce further outputs.
- Those outputs are then passed on to the next layer of neurons, and so it continues until every layer of neurons have been considered, and the terminal neurons have received their input.
- Those terminal neurons then output the final result for the model.

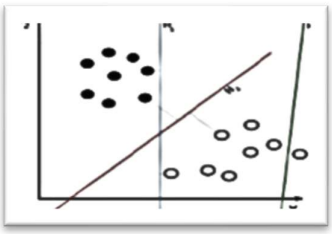
Below figure shows a visual representation of such a network. The initial input is x, which is then passed to the first layer of neurons (the h bubbles in Figure), where three functions consider the input that they receive, and generate an output. That output is then passed to the second layer (the g bubbles in Figure). There further output is calculated, based on the output from the first layer. That secondary output is then combined to yield a final output of the model.

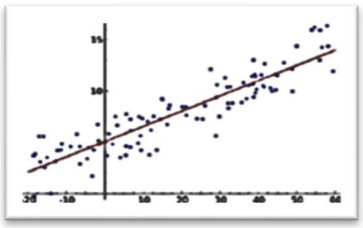


Regression

- A regression problem is when the output variable is a real value, such as “dollars” or “weight”.
- Regression technique predicts a single output value using training data.
- Example: You can use regression to predict the house price from training data. The input variables will be locality, size of a house, etc.

Classification	Regression
Classification means to group the output into a class.	Regression means to predict the output value using training data.
Discrete/Categorial values.	Real/Continuous values.
Method of calculation by measuring accuracy.	Method of calculation by measurement of root mean square error.
It predicts a label.	It predicts a quantity.
Find Decision boundary.	Find Best fit line.
Predicting student pass or fail	Predicting student Marks percentage.
Classification problem: Predicting the type of tumor i.e. harmful or not harmful.	Regression problem: Predicting the house rent price.



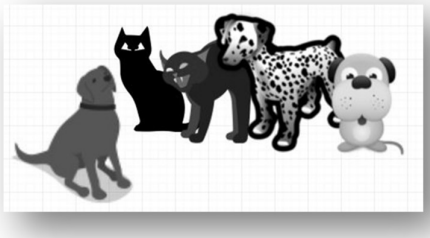


5.4 Unsupervised Learning:

Unsupervised learning is the training of machine using information that is neither classified nor labeled and allowing the algorithm to act on that information without guidance. Here the task of machine is to group unsorted information according to similarities, patterns and differences without any prior training of data.

Unlike supervised learning, no teacher is provided that means no training will be given to the machine. Therefore, machine is restricted to find the hidden structure in unlabeled data by our-self.

For instance, suppose it is given an image having both dogs and cats which have not seen ever.



Thus, the machine has no idea about the features of dogs and cat so we can't categorize it in dogs and cats. But it can categorize them according to their similarities, patterns, and differences i.e., we can easily categorize the above picture into two parts. First may contain all pics having dogs in it and second part may contain all pics having cats in it. Here you didn't learn anything before, means no training data or examples.

Unsupervised learning classified into two categories of algorithms:

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

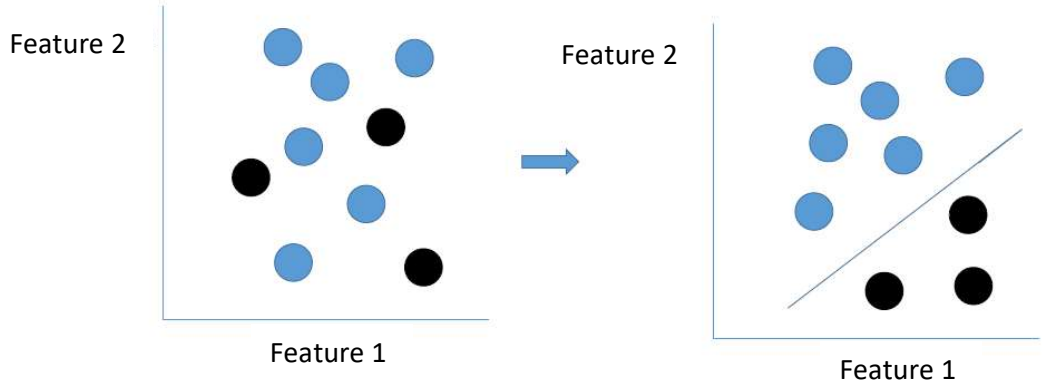
Association: An association rule learning problem is where you want to discover rules that describe large portions of your data, such as people that buy X also tend to buy Y.

Types of Unsupervised Learning

NASA uses unsupervised learning to create clusters of heavenly bodies, with each cluster containing objects of a similar nature.

1. CLUSTERING

The most common unsupervised learning method is cluster analysis. It is used to find data clusters so that each cluster has the most closely matched data.



Example:

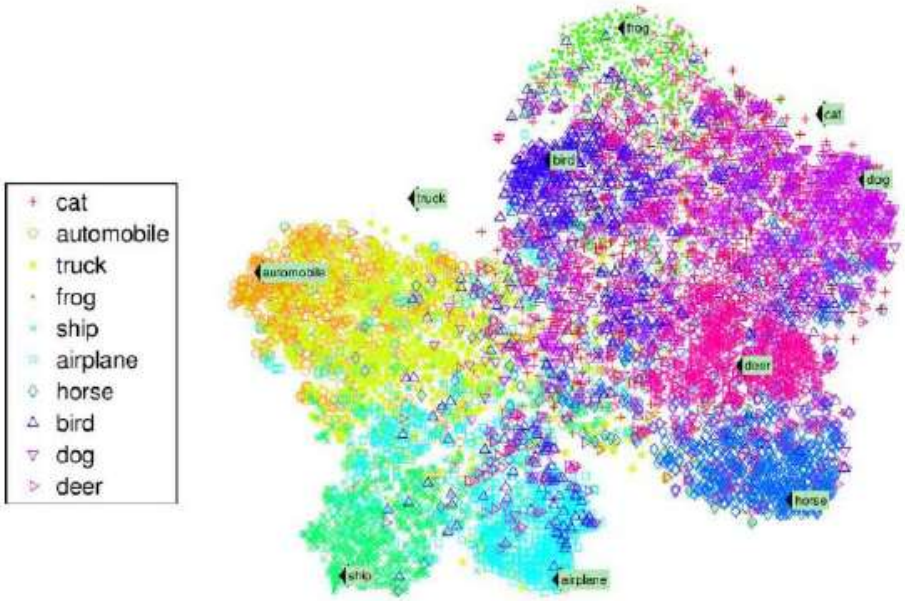
An online news portal segments articles into various categories like Business, Technology, Sports, etc.

2. VISUALIZATION ALGORITHMS

Visualization algorithms are unsupervised learning algorithms that accept unlabeled data and display this data in an intuitive 2D or 3D format. The data is separated into somewhat clear clusters to aid understanding.

Example:

In the figure, the animals are rather well separated from vehicles. Horses are close to deer but far from birds, and so on.

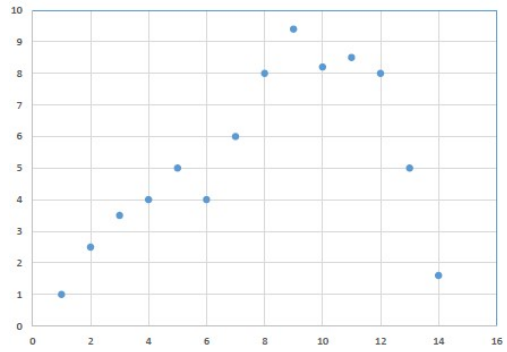


3. ANOMALY DETECTION

This algorithm detects anomalies in data without any prior training.

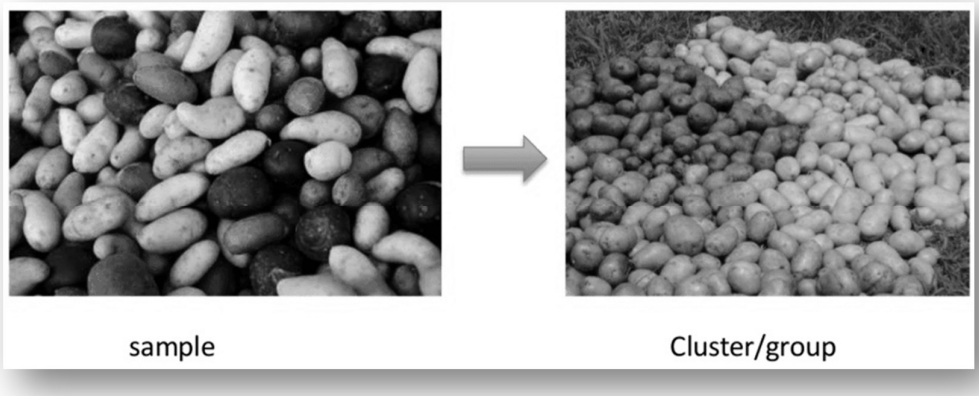
Example:

It can detect suspicious credit card transactions and differentiate a criminal from a set of people.



5.4.1 Clustering, K-Means

Clustering:



Clustering is an important concept when it comes to unsupervised learning. It mainly deals with finding a structure or pattern in a collection of uncategorized data. Clustering algorithms will process your data and find natural clusters(groups) if they exist in the data. You can also modify how many clusters your algorithms should identify. It allows you to adjust the granularity of these groups.
Cluster analysis or clustering is the most commonly used technique of unsupervised learning. It is used to find data clusters such that each cluster has the most closely matched data.

In simple, A group of objects having similar property is called clustering.

Why use Clustering?

Grouping similar entities together help profile the attributes of different groups. In other words, this will give us insight into underlying patterns of different groups. There are many applications of grouping unlabeled data, for example, you can identify different groups segments of customers and market each group in a different way to maximize the revenue.

Another example is grouping documents together which belongs to the similar topics etc.

Clustering is also used to reduces the dimensionality of the data when you are dealing with copious *[abundant in quantity (प्रचुर)]* number of variables.

Euclidian Distance: The Euclidean distance or Euclidean metric is the ordinary straight-line distance between two points in Euclidean space. With this distance, Euclidean space becomes a metric space.

The Euclidean distance between points p and q is the length of the lie segment connecting them.

$$\boldsymbol{P(x_1 y_1)}, \qquad \boldsymbol{Q(x_2 y_2)}$$

$$\boldsymbol{d = \sqrt{(x_n - x_m)^2 + (y_n - y_m)^2}}$$

K-Means

K-means Clustering: Example

PROBLEM STATEMENT

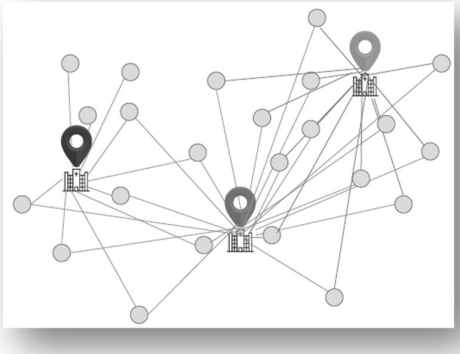
Let's say, in Sudur-Pacchim Pardesh (Province No.7) the local government tries to identify high density clusters to build hospitals (no other ground truth or features are provided apart from the population data). How can the clusters be identified?

STEP 1: RANDOMLY PICK K CENTROIDS Start by picking k centroids. Assume, k = 3

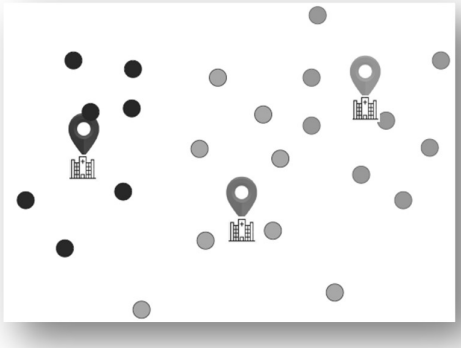
Finding the number of clusters: Use Elbow Method (to be reviewed later)

STEP 2: ASSIGN EACH POINT TO THE NEAREST CENTROID μ(j), j ∈ {1,…,k}.

The points are assigned such that the Euclidean distance of each point from the respective centroid is minimized.



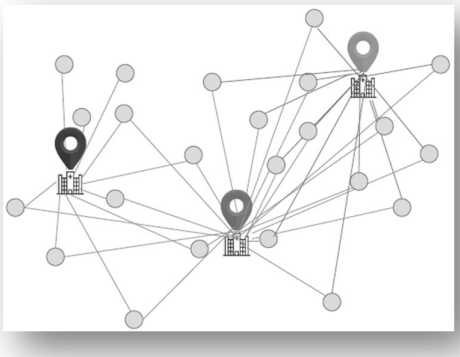
STEP 3: MOVE EACH CENTROID TO THE CENTRE OF THE RESPECTIVE CLUSTER



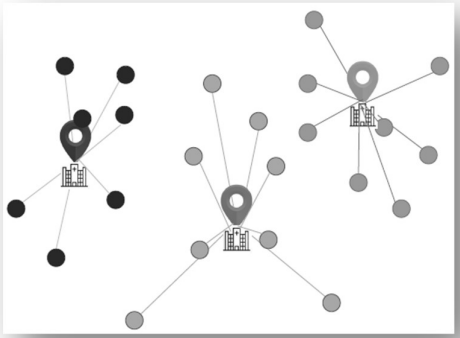
STEP 4: CALCULATE DISTANCE OF THE CENTROIDS FROM EACH POINT AGAIN

Calculate the Euclidean distance between each point and its centroid.

$$\boldsymbol{d = \sqrt{(x_n - x_m)^2 + (y_n - y_m)^2}}$$

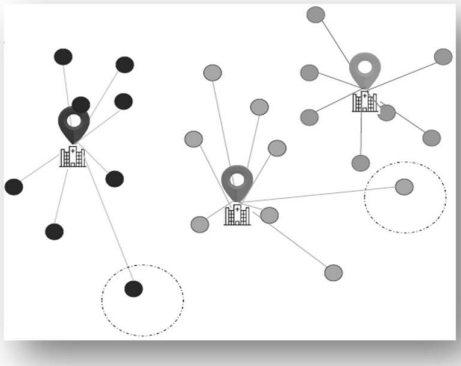


STEP 5: MOVE POINTS ACROSS CLUSTERS AND RE-CALCULATE THE DISTANCE FROM THE CENTROID



STEP 6: KEEP MOVING THE POINTS ACROSS CLUSTERS UNTIL THE EUCLIDEAN DISTANCE IS MINIMIZED

Repeat the steps until within-cluster Euclidean distance is minimized for each cluster (or a user-defined limit on number of iterations is reached)



Python:

In [1]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.cluster import KMeans
```

In [2]:
```
df=pd.read_csv("Mall_Customers.csv")
df.head()
```

Out[2]:

	CustomerID	Genre	Age	Annual Income (k$)	Spending Score (1-100)
0	1	Male	19	15	39
1	2	Male	21	15	81
2	3	Female	20	16	6
3	4	Female	23	16	77
4	5	Female	31	17	40

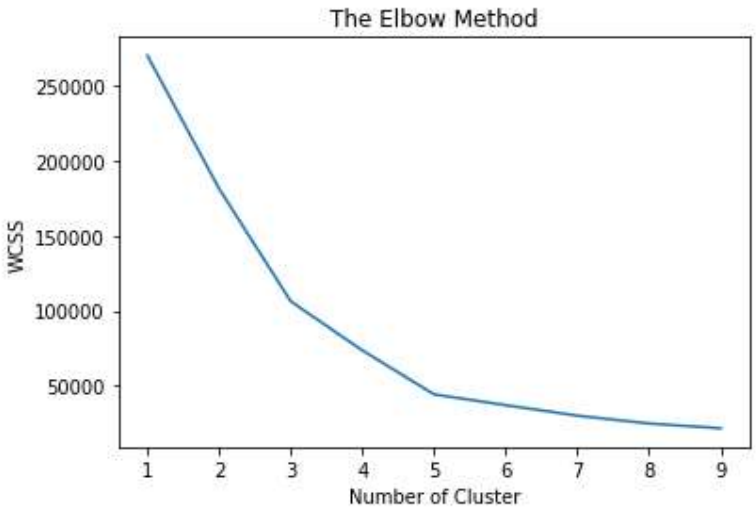
In [3]:
```
df.shape
```

Out[3]:
```
(200, 5)
```

In [4]:
```
X=df.iloc[:,[3,4]].values
```

Using the Elbow method to find the optimal number of Clusters

In [7]:
```
#WCSS=> Within Clusters Sum of Squares
wcss=[]
for i in range(1,10):
    kmeans=KMeans(n_clusters=i,init='k-means++',random_state=42)
    kmeans.fit(X)
    wcss.append(kmeans.inertia_)
plt.plot(range(1,10),wcss)
plt.title('The Elbow Method')
plt.xlabel('Number of Cluster')
plt.ylabel('WCSS')
plt.show()
```



Fitting K-Means to the Dataset

```
In [9]: kmeans=KMeans(n_clusters=5,init='k-means++', random_state=42)
        Y_kmeans=kmeans.fit_predict(X)
```

```
In [11]: Y_kmeans
```

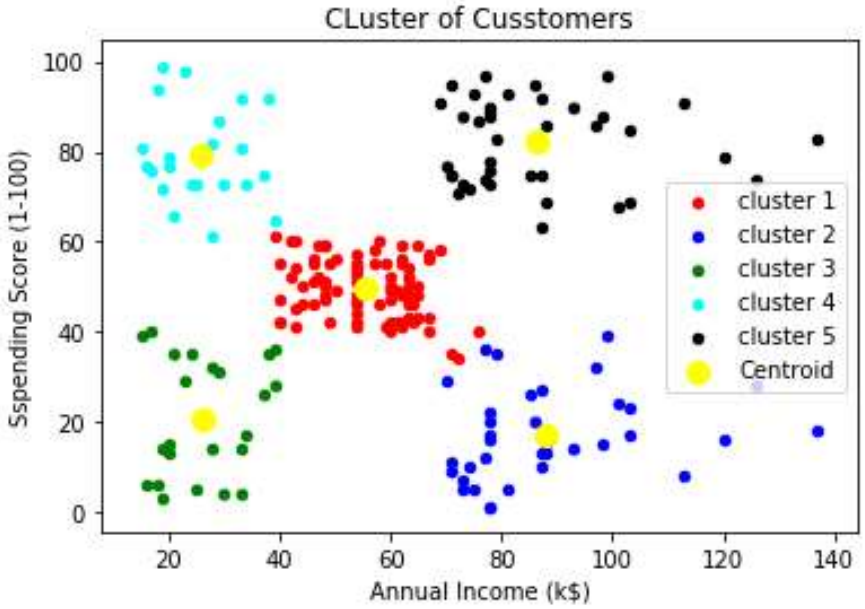
```
Out[11]: array([2, 3, 2, 3, 2, 3, 2, 3, 2, 3, 2, 3, 2, 3, 2, 3, 2, 3, 2, 3,
                2, 3, 2, 3, 2, 3, 2, 3, 2, 3, 2, 3, 2, 3, 2, 3, 2, 3, 2, 0,
                2, 3, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
                0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
                0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
                0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 4, 1, 4, 0, 4, 1, 4, 1, 4,
                0, 4, 1, 4, 1, 4, 1, 4, 1, 4, 0, 4, 1, 4, 1, 4, 1, 4, 1, 4, 1, 4,
                1, 4, 1, 4, 1, 4, 1, 4, 1, 4, 1, 4, 1, 4, 1, 4, 1, 4, 1, 4, 1, 4,
                1, 4, 1, 4, 1, 4, 1, 4, 1, 4, 1, 4, 1, 4, 1, 4, 1, 4, 1, 4, 1, 4,
                1, 4])
```

```
In [14]: len=list(Y_kmeans)
         len.count(4)
```

```
Out[14]: 39
```

Visualizing the Clusters

```
In [16]: plt.scatter(X[Y_kmeans==0,0], X[Y_kmeans==0,1],s=20,c='red',label='cluster 1')
         plt.scatter(X[Y_kmeans==1,0], X[Y_kmeans==1,1],s=20,c='blue',label='cluster 2')
         plt.scatter(X[Y_kmeans==2,0], X[Y_kmeans==2,1],s=20,c='green',label='cluster 3')
         plt.scatter(X[Y_kmeans==3,0], X[Y_kmeans==3,1],s=20,c='cyan',label='cluster 4')
         plt.scatter(X[Y_kmeans==4,0], X[Y_kmeans==4,1],s=20,c='black',label='cluster 5')

         plt.scatter(kmeans.cluster_centers_[:,0],kmeans.cluster_centers_[:,1],s=100,c='yellow',label='Centroid')

         plt.title('CLuster of Cusstomers')
         plt.xlabel('Annual Income (k$)')
         plt.ylabel('Sspending Score (1-100)')
         plt.legend()
         plt.show()
```



Other Examples of K-means Clustering

- Grouping articles (example: Google news).
- Grouping products (example: Amazon, Flipkart etc.)
- Grouping customers who share similar interests, example: analyzing customers who like contemporary fashion vs. those who prefer traditional clothing.
- Classifying high risk and low risk patients from a patient pool.
- Segregating criminals from normal crowd in a security control process.

Supervised Vs Un-Supervised Learning

Parameters	Supervised Learning	Unsupervised Learning
Process	In a supervised learning model, input and output variables will be given.	In unsupervised learning model, only input data will be given
Input Data	Algorithms are trained using labeled data.	Algorithms are used against data which is not labeled
Algorithms Used	Support vector machine, Neural network, Linear and logistics regression, random forest, and Classification trees.	Unsupervised algorithms can be divided into different categories: like Cluster algorithms, K-means, Hierarchical clustering, etc.
Computational Complexity	Supervised learning is a simpler method.	Unsupervised learning is computationally complex
Use of Data	Supervised learning model uses training data to learn a link between the input and the outputs.	Unsupervised learning does not use output data.
Accuracy of Results	Highly accurate and trustworthy method.	Less accurate and trustworthy method.
Real Time Learning	Learning method takes place offline.	Learning method takes place in real time.
Number of Classes	Number of classes is known.	Number of classes is not known.
Main Drawback	Classifying big data can be a real challenge in Supervised Learning.	You cannot get precise information regarding data sorting, and the output as data used in unsupervised learning is labeled and not known.

5.4.2 Dimensionality Reduction [Principal Component Analysis (PCA), Linear Discriminant Analysis]

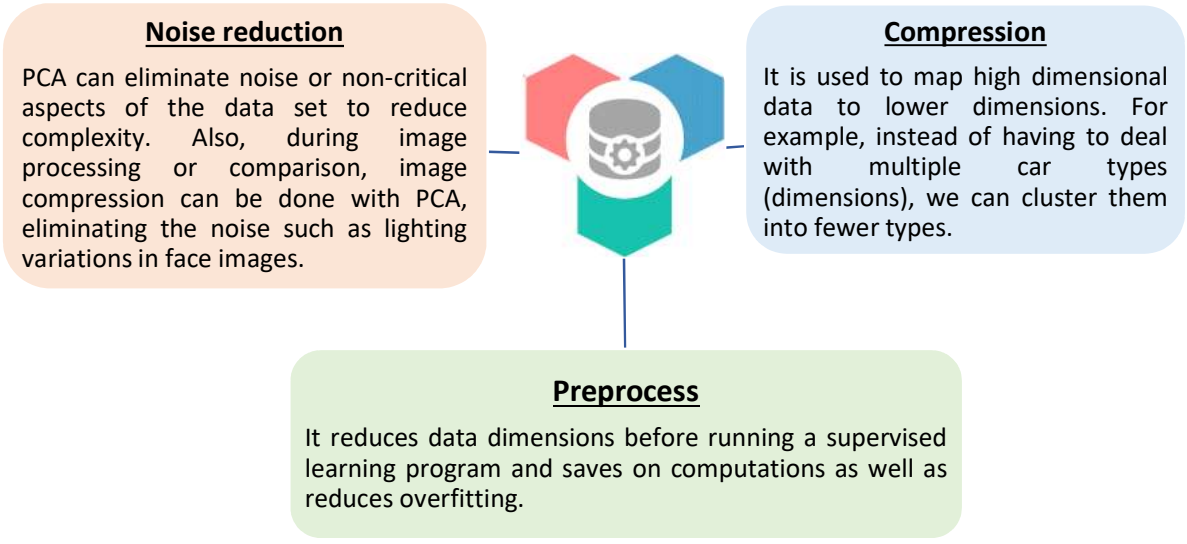
Dimensionality Reduction:

- Dimensionality reduction involves transformation of data to new dimensions in a way that facilitates discarding of some dimensions without losing any key information.
- Large scale problems bring about several dimensions that can become very difficult to visualize.
- Some of such dimensions can be easily dropped for a better visualization.
- Example: Car attributes might contain maximum speed in both units, kilometer per hour, and miles per hour. One of these can be safely discarded in order to reduce the dimensions and simplify the data.

Dimensionality Reduction with Principal Component Analysis (PCA)

- Principal component analysis (PCA) is a technique for dimensionality reduction that helps in arriving at better visualization models.

Applications of PCA

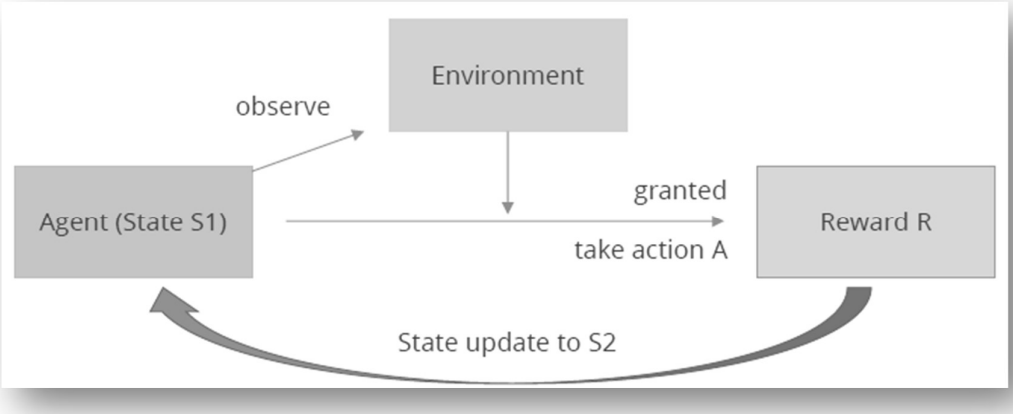


5.1 Reinforcement Learning

Reinforcement Learning is a type of Machine Learning that allows the learning system to observe the environment and learn the ideal behavior based on trying to maximize some notion of cumulative reward.

Features of Reinforcement Learning

- The learning system (agent) observes the environment, selects and takes certain actions, and gets rewards in return (or penalties in certain cases).
- The agent learns the strategy or policy (choice of actions) that maximizes its rewards over time.



Example of Reinforcement Learning

In a manufacturing unit, a robot uses deep reinforcement learning to identify a device from one box and put it in a container.

The robot learns this by means of a rewards-based learning system, which incentivizes it for the right action.

5.6 Fuzzy Learning

What is Fuzzy Logic?

Fuzzy logic is an approach to computing based on "degree of truth" rather than the usual "True or False" (1 or 0) Boolean logic on which the modern computer is based.

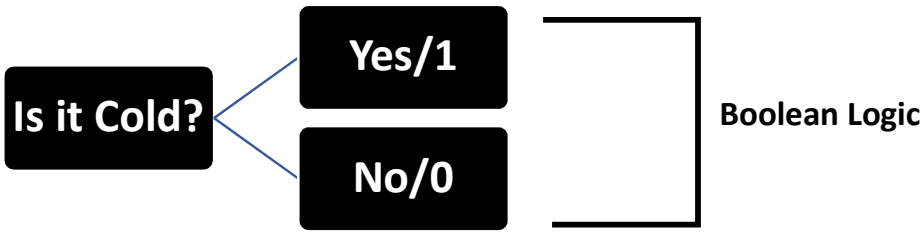
The idea of fuzzy logic was first announced by Dr. Lotfi Zadeh of the university of California at Berkeley in 1960s.

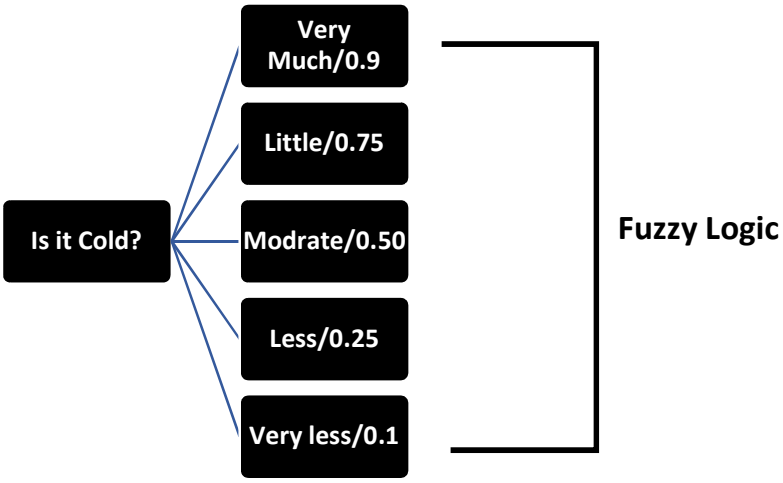
Fuzzy Logic (FL) is a method of reasoning that resembles human reasoning. This approach is similar to how humans perform decision making. And it involves all intermediate possibilities between YES and NO.

The conventional logic block that a computer can understand takes precise input and produces a definite output TRUE or FALSE, which is equivalent to humans YES or NO.

Fuzzy logic can be implemented in systems with various sizes and capabilities ranging from small micro-controllers to large, networked, workstation-based control systems.

Fuzzy logic can be implemented in hardware, software and or a combination of both.



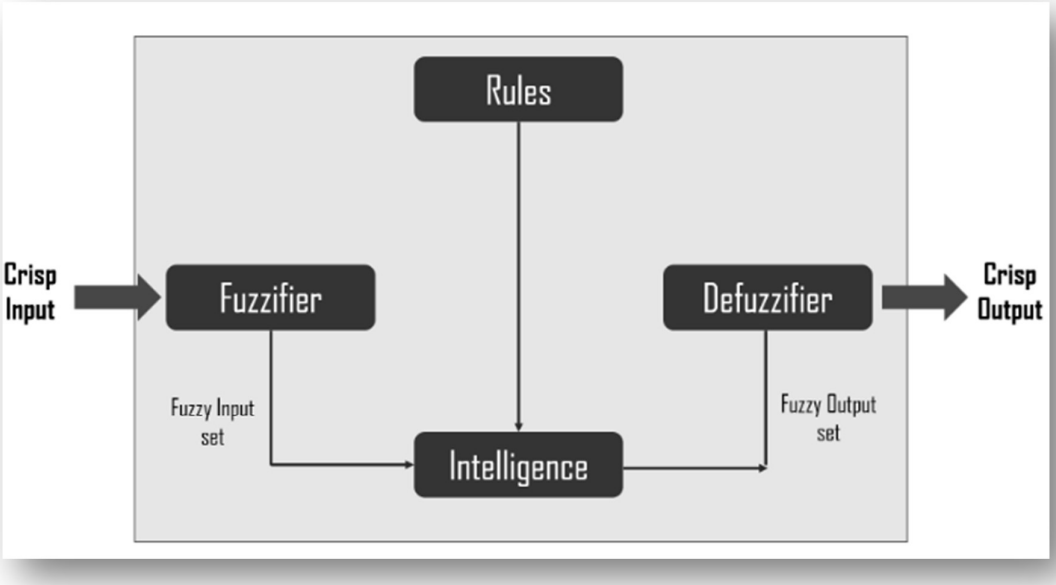


Applications of Fuzzy Logic

- The Fuzzy logic is used in various fields such as automotive systems, domestic goods, environment control, etc. Some of the common applications are:
- It is used in the aerospace field for altitude control of spacecraft and satellite.
- This controls the speed and traffic in the automotive systems.
- It is used for decision making support systems and personal evaluation in the large company business.
- It also controls the pH, drying, chemical distillation process in the chemical industry.
- Fuzzy logic is used in Natural language processing and various intensive applications in Artificial Intelligence.
- It is extensively used in modern control systems such as expert systems.
- Fuzzy Logic mimics how a person would make decisions, only much faster. Thus, you can use it with Neural Networks.

Fuzzy Logic Architecture

The fuzzy logic architecture consists of four main parts:



- **Rules** – It contains all the rules and the if-then conditions offered by the experts to control the decision-making system. The recent update in the fuzzy theory provides different effective methods for the design and tuning of **fuzzy controllers**. Usually, these developments reduce the number of fuzzy rules.
- **Fuzzification** – This step converts inputs or the crisp numbers into fuzzy sets. You can measure the crisp inputs by sensors and pass them into the **control system** for further processing. It splits the input signal into five steps such as-

LP	X is Large Positive
MP	X is Medium Positive
S	Small
MN	X is Medium Negative
LN	X is Large Negative

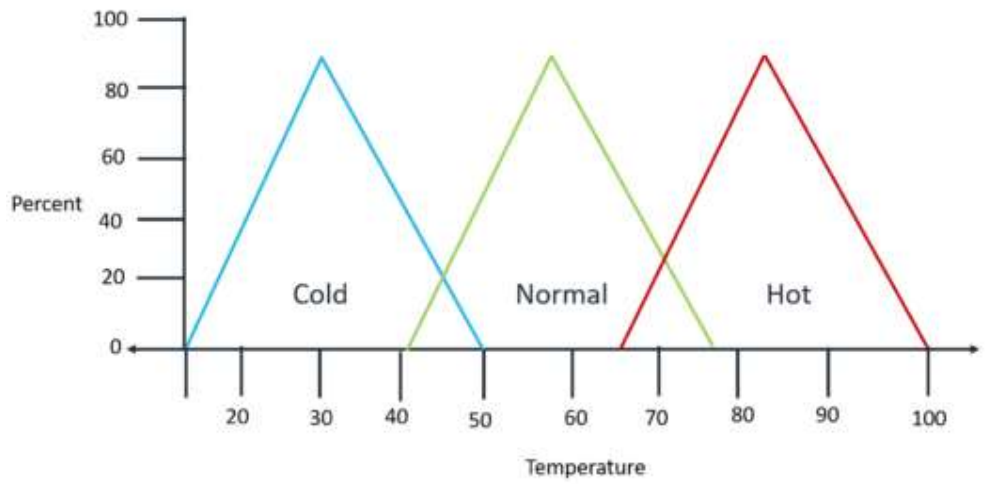
- **Inference Engine** – It determines the degree of match between fuzzy input and the rules. According to the input field, it will decide the rules that are to be fired. Combining the fired rules, form the control actions.
- **Defuzzification** – The Defuzzification process converts the fuzzy sets into a crisp value. There are different types of techniques available, and you need to select the best-suited one with an expert system.

Fuzzy Logic in AI: Example

The design of a fuzzy logic system starts with a set of membership functions for each input and a set for each output. A set of rules is then applied to the membership functions to yield a crisp output value. Let's take an example of process control and understand fuzzy logic.

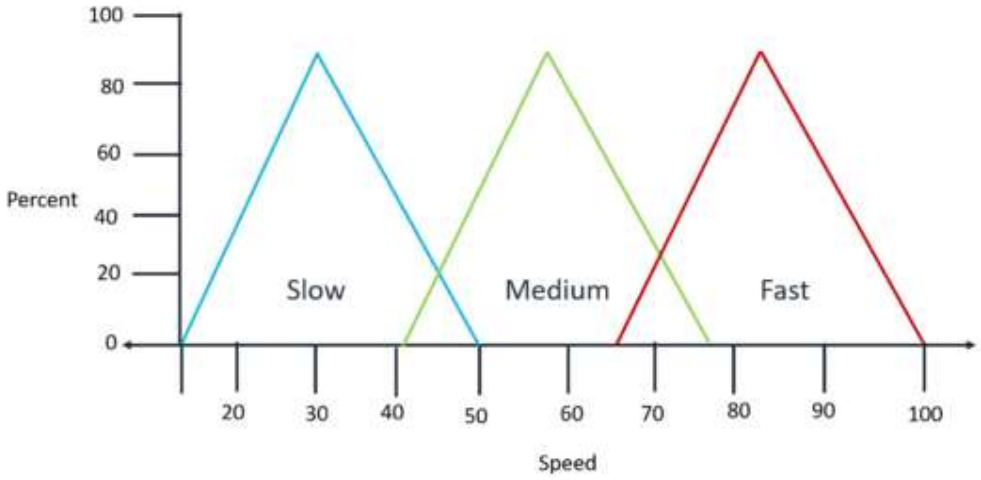
Step 1

Here, **Temperature** is the input and **Fan Speed** is the output. You have to create a set of membership functions for each input. A membership function is simply a graphical representation of the fuzzy variable sets. For this example, we will use three fuzzy sets, **Cold, Warm** and **Hot**. We will then create a membership function for each of three sets of temperature:



Step 2

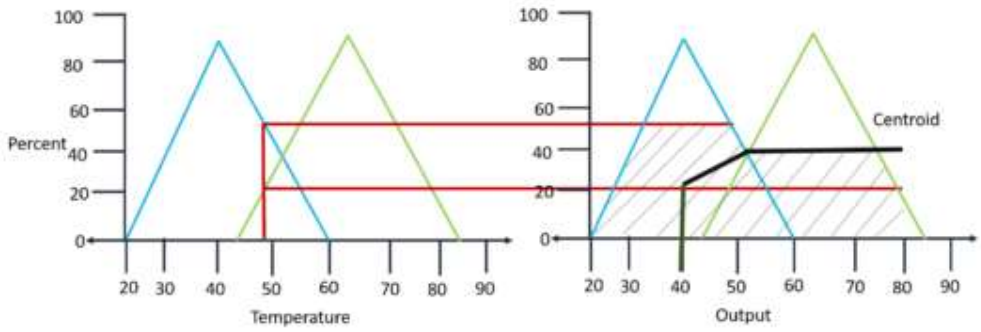
In the next step, we will use three fuzzy sets for the output, **Slow, Medium** and **Fast**. A set of functions is created for each output set just as for the input sets.



Step 3

Now that we have our membership functions defined, we can create the rules that will define how the membership functions will be applied to the final system. We will create three rules for this system.

- **If Hot then Fast**
- **If Warm then Medium**
- **And, if Cold then Slow**



These rules apply to the membership functions to produce the crisp output value to drive the system. Thus, for an input value of **52 degrees**, we intersect the membership functions. Here, we are applying two rules as the intersection occurs on both functions. You can extend the intersection points to the output functions to produce an intersecting point. You can then truncate the output functions at the height of the intersecting points.

Advantages of Fuzzy Logic

Fuzzy logic provides simple reasoning similar to human reasoning. There are more such **advantages** of using this logic, such as:

- The structure of Fuzzy Logic Systems is **easy and understandable.**
- Fuzzy logic is widely used for **commercial** and **practical purposes.**
- It helps you to **control machines** and consumer products.
- It helps you to deal with the **uncertainty in engineering.**
- Mostly **robust** as no precise inputs required.
- If the feedback sensor stops working, you can **program it** into the situation.
- You can **easily modify** it to improve or alter system performance.
- **Inexpensive sensors** can be used which helps you to keep the overall system cost and complexity low.

Disadvantages of Fuzzy Logic

Fuzzy logic is **not always accurate**. So, the results are perceived based on assumptions and may not be widely accepted

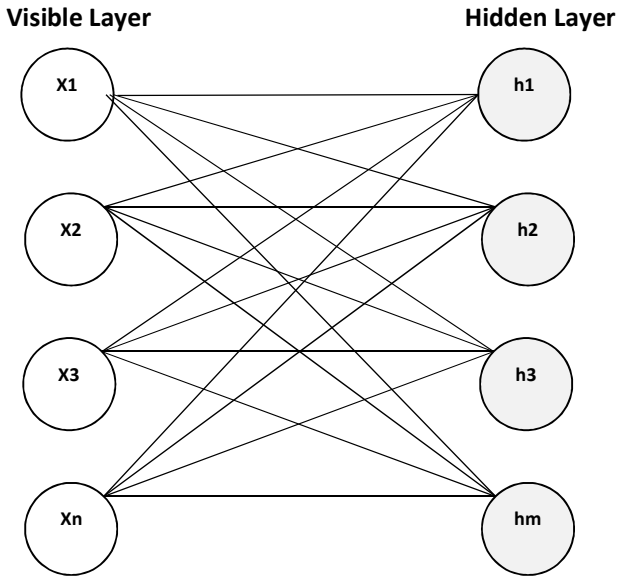
- It **cannot recognize** machine learning as-well-as neural network type patterns.
- **Validation and Verification** of a fuzzy knowledge-based system needs **extensive testing** with hardware.
- Setting exact, fuzzy rules and, membership functions is a **difficult task.**
- At times, the fuzzy logic is **confused** with **probability theory.**

5.7 Boltzmann Machines

A Hopfield Net consisting of Binary Stochastic Neuron with hidden units is called Boltzmann Machine.

A Boltzmann Machine is a network of symmetrically connected, neuron like units that make stochastic decisions about whether to be ON or OFF.

Structure of Boltzmann Machine



The stochastic neurons of Boltzmann machine are in two groups: Visible and Hidden.

Visible neurons provide an interface between the net and its environment.

During the training phase, the visible neurons are clamped; the hidden neurons always operate freely, they are used to explain underlying constraints in the environmental input vectors.

Boltzmann machines are stochastic and generative neural networks capable of learning internal representations and are able to represent and (given sufficient time) solve difficult combinatoric problems.

They are named after the Boltzmann distribution (**also known as Gibbs Distribution**) which is an integral part of Statistical Mechanics and helps us to understand the impact of parameters like Entropy and Temperature on the Quantum States in Thermodynamics.

That's why they are called Energy-Based Models (EBM). They were invented in 1985 by Geoffrey Hinton, then a Professor at Carnegie Mellon University, and Terry Sejnowski, then a Professor at Johns Hopkins University.

Some key features Boltzmann Machine:

- Each node is connected to all other nodes.
- There is no single output in the machine, only hidden units and visible units, where the visible units represent the data. This model tries to understand the distribution of the data and recreate the data based on that distribution.

Application of Boltzmann Machine:

- Stock market trend prediction
- Character recognition
- Face recognition
- Internet application
- Cancer detection
- Loan application
- Decision making

5.8 Deep Learning

What is Deep Learning?

Deep Learning is a new area of Machine Learning research, which has been introduced with the objective of moving Machine Learning closer to one of its original goals: Artificial Intelligence.

Deep Learning is a subfield of machine learning concerned with algorithms inspired by the structure and function of the brain called artificial neural networks.

Deep Learning is a specialized form of Machine Learning that uses supervised, unsupervised, or semi-supervised learning to learn data representations. It is similar to the structure and function of the human nervous system.

Why Deep Learning?

- Experts have discovered multi-layered learning networks that can be leveraged for deep learning as they learn in layers.
- The vast availability of Big Data enables machines to be trained.
- Scientists have figured out that high-performing graphics processing units (GPU) can be used for deep learning.
- GPUs allow parallel computing, unlike conventional Central Processing Units (CPU) that use sequential processing.

Difference: Machine Learning and Deep Learning

Machine Learning allows one to learn algorithms that represent the relationship between input and output.	Deep Learning is a more specialized form of Machine Learning that allows one to learn the representations of data itself, which leads to more complex decision-making, such as object detection, speech processing, or text recognition.

Applications of Deep Learning:

1. Automated Driving:

Automated driving is becoming one of the most emerging topics nowadays. Various companies are applying deep learning technique to create an automated vehicle which doesn't requires human supervision to function. They are automatically detecting objects such as stop signs and traffic lights. In addition, deep learning is used to detect pedestrians, which helps decrease accidents.

2. Medical Research:

Various researchers are working on deep learning so that they can create a machine which are automatically detected the defect or disease in living body by just observing its patterns. Cancer researchers are using deep learning to automatically detect cancer cells.

3. Aerospace and Defense:

Deep learning is used in various defense sectors as to identify an unauthorized or can be used to locate areas of interest by using given environmental aspects. Using this information, we can give commands or order to troops that whether it is safe or unsafe to work in environment.

THE END

PU Examination Questions with Solution

K-Means Algorithm

We know the K-Means Clustering Algorithm

Step 1: Select initial centroids at random of each cluster (from datasets of n objects).

Step 2: Assign each object to the cluster with the nearest centroid.

Step 3: Compute each centroid as the mean of the objects assigned to it. (Get the mean value of each cluster) & make them as a new centroid.

Step 4: Repeat **Step 2** and **Step 3** until no change in mean value.

Step 5: Stop when no more new assignment.

Euclidean Distance:

$$d = \sqrt{(x-a)^2 + (y-b)^2}$$

x & y => Observed value from the given dataset.

a & b => Centroid value or Mean value

[2016 Fall 4.b]

When is it useful to use clustering? The following data set contains the scores of two variables on each of seven individuals. Taking 1 and 4 as initial cluster center, applying K-means clustering to group the given data:

Subject	A	B
1	1.0	1.0
2	1.5	2.0
3	3.0	4.0
4	5.0	7.0
5	3.5	5.0
6	4.5	5.0
7	3.5	4.5

Ans: Clustering is used to find structure in unlabeled data. It's the most common form of unsupervised learning. Given a dataset we don't know anything about, a clustering algorithm can discover groups of objects where the average distances between the members of each cluster are closer than to members in other clusters.

Clustering → is grouping similar data → together → and this can be used for prediction or just clustering.

Or even dimension reduction → depending on how we use this.

Why do we cluster?

- **Summarizing data**
  - Look at large amount of data
  - Represent a large continuous vector with the cluster number
- **Counting**
  - Computing feature histograms
- **Prediction**
  - Data points in the same cluster may have the same labels

Soln: Given, 1 and 4 as initial cluster center.

Subject	A	B
1	**1.0**	**1.0**
2	1.5	2.0
3	3.0	4.0
4	**5.0**	**7.0**
5	3.5	5.0
6	4.5	5.0
7	3.5	4.5

Cluster	Individual	Mean Value / Centroid
K1 (Group 1)	1	(1.0, 1.0)
K2 (Group 2)	4	(5.0, 7.0)

The remaining individual are now examined in sequence and allocated to the cluster to which they are closest in terms of Euclidean distance to the cluster mean. The mean value (Centroid) is re-calculated each time a new member is added.

We know, Euclidean distance

$$\boldsymbol{d = \sqrt{(x_n - x_m)^2 + (y_n - y_m)^2}}$$

A	B	K1 = (1.0, 1.0) $\sqrt{(A-1.0)^2 + (B-1.0)^2}$	K2 = (5.0, 7.0) $\sqrt{(A-5.0)^2 + (B-7.0)^2}$	Cluster Assignment
1.0	1.0	**0**	7.211	K1
1.5	2.0	**1.118**	6.10	K1
3.0	4.0	**3.60**	3.60	K1
5.0	7.0	7.211	**0**	K2
3.5	5.0	4.716	**2.5**	K2
4.5	5.0	5.315	**2.061**	K2
3.5	4.5	4.301	**2.91**	K2

Here how we are picking cluster is that look the minimum value by comparing K1 and K2,

if (K1<K2) then
assign K1
else
assign K2

K1 = {(1.0, 1.0), (1.5, 2.0), (3.0, 4.0)}

K2 = {(5.0, 7.0), (3.5, 5.0), (4.5, 5.0), (3.5, 4.5)}

Now, Calculate Mean value (Centroid)

$$K1 = (A, B) = \left(\frac{A_1+A_2+A_3}{3}\right), \left(\frac{B_1+B_2+B_3}{3}\right)$$

$$= \left(\frac{1.0+1.5+3.0}{3}\right), \left(\frac{1.0+2.0+4.0}{3}\right)$$

$$= \left(\frac{5.5}{3}\right), \left(\frac{7}{3}\right)$$

K1 = 1.83, 2.33

$$K2 = (A, B) = \left(\frac{A_1+A_2+A_3+A_4}{4}\right), \left(\frac{B_1+B_2+B_3+B_4}{4}\right)$$

$$= \left(\frac{5.0+3.5+4.5+3.5}{4}\right), \left(\frac{7.0+5.0+5.0+4.5}{4}\right)$$

$$= \left(\frac{16.5}{4}\right), \left(\frac{21.5}{4}\right)$$

K2 = 4.125, 5.375

Now, Update the cluster centroid and calculate Euclidean distance:

A	B	K1 = (1.8, 2.3) $\sqrt{(A-1.8)^2+(B-2.3)^2}$	K2 = (4.1, 5.4) $\sqrt{(A-4.1)^2+(B-5.4)^2}$	Cluster Assignment
1.0	1.0	**1.526**	5.38	K1
1.5	2.0	**0.424**	4.28	K1
3.0	4.0	2.080	**1.78**	K2
5.0	7.0	5.685	**1.83**	K2
3.5	5.0	3.190	**0.72**	K2
4.5	5.0	3.818	**0.56**	K2
3.5	4.5	2.780	**1.08**	K2

Here, how we are picking cluster is that look the minimum value by comparing K1 and K2,

if (K1<K2) then
assign K1
else
assign K2

K1 = {(1.0, 1.0), (1.5, 2.0)}

K2 = {(3.0, 4.0), (5.0, 7.0), (3.5, 5.0), (4.5, 5.0), (3.5, 4.5)}

Now, Calculate Mean value (Centroid)

$$K1 = (A, B) = \left(\frac{A_1+A_2}{2}\right), \left(\frac{B_1+B_2}{2}\right)$$

$$= \left(\frac{1.0+1.5}{2}\right), \left(\frac{1.0+2.0}{2}\right)$$

$$= \left(\frac{2.5}{2}\right), \left(\frac{3}{2}\right)$$

K1 = 1.25=1.3, 1.5

$$K2 = (A, B) = \left(\frac{A_1+A_2+A_3+A_4+A_5}{5}\right), \left(\frac{B_1+B_2+B_3+B_4+B_5}{5}\right)$$

$$= \left(\frac{3.0+5.0+3.5+4.5+3.5}{5}\right), \left(\frac{4.0+7.0+5.0+5.0+4.5}{5}\right)$$

$$= \left(\frac{19.5}{5}\right), \left(\frac{25.5}{5}\right)$$

K2 = 3.9, 5.1

Now, Update the cluster centroid and calculate Euclidean distance:

A	B	K1 = (1.3, 1.5) $\sqrt{(A-1.3)^2+(B-1.5)^2}$	K2 = (3.9, 5.1) $\sqrt{(A-3.9)^2+(B-5.1)^2}$	Cluster Assignment
1.0	1.0	**0.583**	5.021	K1
1.5	2.0	**0.538**	3.92	K1
3.0	4.0	3.023	**1.421**	K2
5.0	7.0	6.628	**2.19**	K2
3.5	5.0	4.134	**0.41**	K2
4.5	5.0	4.74	**0.60**	K2
3.5	4.5	4.72	**0.72**	K2

Here, how we are picking cluster is that look the minimum value by comparing K1 and K2,

if (K1<K2) then
assign K1
else
assign K2

K1 = {(1.0, 1.0), (1.5, 2.0)}

K2 = {(3.0, 4.0), (5.0, 7.0), (3.5, 5.0), (4.5, 5.0), (3.5, 4.5)}

Now, Calculate Mean value (Centroid)

$$K1 = (A, B) = \left(\frac{A_1+A_2}{2}\right), \left(\frac{B_1+B_2}{2}\right)$$

$$= \left(\frac{1.0+1.5}{2}\right), \left(\frac{1.0+2.0}{2}\right)$$

$$= \left(\frac{2.5}{2}\right), \left(\frac{3}{2}\right)$$

K1 = 1.25=1.3, 1.5

$$K2 = (A, B) = \left(\frac{A_1+A_2+A_3+A_4+A_5}{5}\right), \left(\frac{B_1+B_2+B_3+B_4+B_5}{5}\right)$$

$$= \left(\frac{3.0+5.0+3.5+4.5+3.5}{5}\right), \left(\frac{4.0+7.0+5.0+5.0+4.5}{5}\right)$$

$$= \left(\frac{19.5}{5}\right), \left(\frac{25.5}{5}\right)$$

K2 = 3.9, 5.1

Since, mean value (Centroid) of K1 and K2 has no any change so, Stop no more new assignment.

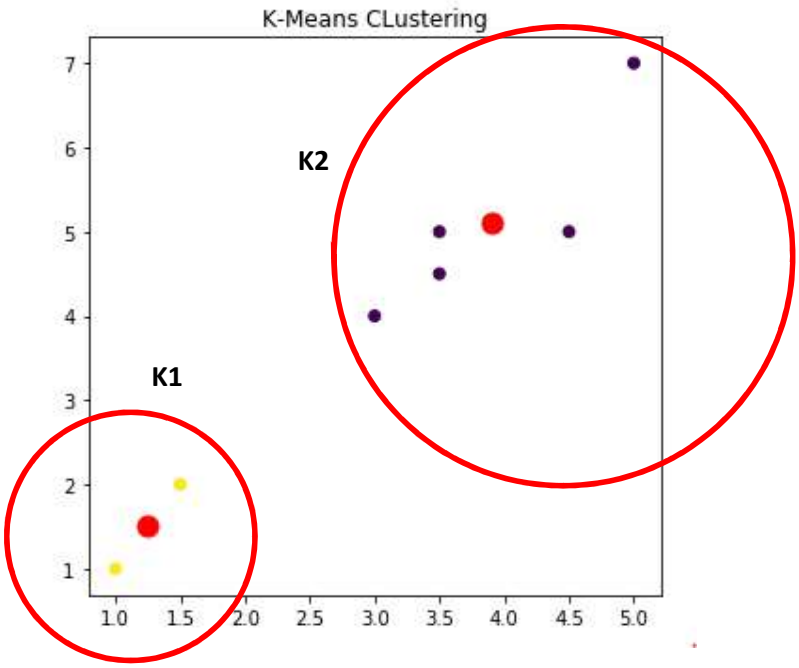
So final partition table would be,

Cluster	Individual	Mean Value / Centroid
K1 (Group 1)	1, 2	(1.3, 1.5)
K2 (Group 2)	3, 4, 5, 6, 7	(3.9, 5.1)

----------X----------X----------

Python:

```
In [1]: import numpy as np
        from sklearn.cluster import KMeans

In [2]: X2 = np.array([[1.0, 1.0], [1.5, 2.0], [3.0, 4.0],[5.0, 7.0], [3.5, 5.0], [4.5, 5.0], [3.5, 4.5]])

In [3]: X2_kmeans = KMeans(n_clusters=2).fit(X2)

In [4]: X2_kmeans.labels_
Out[4]: array([1, 1, 0, 0, 0, 0, 0])

In [5]: len=list(X2_kmeans.labels_)

In [6]: print("Total No. of Cluster K1= ",len.count(1))
        print("Total No. of Cluster K2= ",len.count(0))

        Total No. of Cluster K1=  2
        Total No. of Cluster K2=  5

In [7]: X2_kmeans.cluster_centers_
Out[7]: array([[3.9 , 5.1 ],
               [1.25, 1.5 ]])

In [9]: import matplotlib.pyplot as plt

        plt.figure(figsize=(12, 12))

        X2_pred = KMeans(n_clusters=2).fit_predict(X2)

        plt.subplot(221)
        plt.scatter(X2[:, 0], X2[:, 1], c=X2_pred)
        plt.scatter(X2_kmeans.cluster_centers_[:,0],X2_kmeans.cluster_centers_[:,1],s=120,c='red')
        plt.title("K-Means CLustering")
        plt.show()
```



[2018 Fall 3.b]

Define Bayes rule for probabilistic problem. If probability of symptoms on patients is 1/20 and probability of disease known by doctor is 1/45000, Also the probability of symptoms condition to disease is 50%. Now, calculate the probability of disease condition to symptoms using Bayes theorem. **[7]**

Ans: Bayes' Theorem is a way of finding a probability when we know certain other probabilities. Bayes' theorem is a way to figure out conditional probability. Conditional probability is the probability of an event happening, given that it has some relationship to one or more other events.

The formula is:

$$\boldsymbol{P(A|B) = \frac{P(A) * P(B|A)}{P(B)}}$$

Which tells us: how often A happens given that B happens, written P(A|B),

When we know: how often B happens given that A happens, written P(B|A)

and how likely A is on its own, written P(A)

and how likely B is on its own, written P(B)

Sol^{n}: Given,

Probability of **Symptoms** on patients P(S) = 1/20 = 0.50

Probability of **Diseases** known by doctor P(D) = 1/45000 = 0.0000222

Probability of **Symptoms** condition to **Diseases** P(S|D) = 50% = 0.50

Here we have to, calculate the probability of **Disease** condition to **Symptoms** P(D|S) using Bayes theorem.

According to Bayes Theorem, $\boldsymbol{P(D|S) = \frac{P(D) * P(S|D)}{P(S)}}$

$$= \frac{\mathbf{0.50 * 1/45000}}{\mathbf{1/20}}$$

$$= \mathbf{0.000222} \quad \boldsymbol{Ans.}$$

[2019 Fall 4.a]

What is Deep Learning? Explain the types of stochastic neurons of Boltzmann Machine?

Ans: Deep Learning is a subfield of machine learning concerned with algorithms inspired by the structure and function of the brain called artificial neural networks.

The field of artificial intelligence is essentially when machines can do tasks that typically require human intelligence. It encompasses machine learning, where machines can learn by experience and acquire skills without human involvement.

Examples of Deep Learning

1. Virtual assistants

Whether it's Alexa or Siri or Cortana, the virtual assistants of online service providers use deep learning to help understand your speech and the language humans use when they interact with them.

2. Translations

In a similar way, deep learning algorithms can automatically translate between languages. This can be powerful for travelers, business people and those in government.

3. Vision for driverless delivery trucks, drones and autonomous cars

The way an autonomous vehicle understands the realities of the road and how to respond to them whether it's a stop sign, a ball in the street or another vehicle is through deep learning algorithms. The more data the algorithms receive, the better they are able to act human-like in their information processing—knowing a stop sign covered with snow is still a stop sign.

4. Chatbots and service bots

Chatbots and service bots that provide customer service for a lot of companies are able to respond in an intelligent and helpful way to an increasing amount of auditory and text questions thanks to deep learning.

5. Image colorization

Transforming black-and-white images into color was formerly a task done meticulously by human hand. Today, deep learning algorithms are able to use the context and objects in the images to color them to basically recreate the black-and-white image in color. The results are impressive and accurate.

6. Facial recognition

Deep learning is being used for facial recognition not only for security purposes but for tagging people on Facebook posts and we might be able to pay for items in a store just by using our faces in the near future. The challenges for deep-learning algorithms for facial recognition is knowing it's the same person even when they have changed hairstyles, grown or shaved off a beard or if the image taken is poor due to bad lighting or an obstruction.

7. Medicine and pharmaceuticals

From disease and tumor diagnoses to personalized medicines created specifically for an individual's genome, deep learning in the medical field has the attention of many of the largest pharmaceutical and medical companies.

8. Personalized shopping and entertainment

Ever wonder how Netflix comes up with suggestions for what you should watch next? Or where Amazon comes up with ideas for what you should buy next and those suggestions are exactly what you need but just never knew it before? Yep, it's deep-learning algorithms at work.

The types of stochastic neurons of Boltzmann Machine are:

1. Restricted Boltzmann machine
2. Deep Boltzmann machine
3. Spike-and-slab RBMs

1. Restricted Boltzmann machine

Although learning is impractical in general Boltzmann machines, it can be made quite efficient in a restricted Boltzmann machine (RBM) which does not allow intralayer connections between hidden units. After training one RBM, the activities of its hidden units can be treated as data for training a higher-level RBM. This method of stacking RBMs makes it possible to train many layers of hidden units efficiently and is one of the most common deep learning strategies. As each new layer is added the generative model improves.

An extension to the restricted Boltzmann machine allows using real valued data rather than binary data.

One example of a practical RBM application is in speech recognition.

2. Deep Boltzmann machine

Deep Boltzmann machine (DBM) is a type of binary pairwise Markov random field (undirected probabilistic graphical model) with multiple layers of hidden random variables. It is a network of symmetrically coupled stochastic binary units.

DBMs can learn complex and abstract internal representations of the input in tasks such as object or speech recognition, using limited, labeled data to fine-tune the representations built using a large set of unlabeled sensory input data. However, unlike DBNs and deep convolutional neural networks, they pursue the inference and training procedure in both directions, bottom-up and top-down, which allow the DBM to better unveil the representations of the input structures.

3. Spike-and-slab RBMs

The need for deep learning with real-valued inputs, as in Gaussian RBMs, led to the spike-and-slab RBM (ssRBM), which models continuous-valued inputs with binary latent variables. Similar to basic RBMs and its variants, a spike-and-slab RBM is a bipartite graph, while like GRBMs, the visible units (input) are real-valued. The difference is in the hidden layer, where each hidden unit has a binary spike variable and a real-valued slab variable. A spike is a discrete probability mass at zero, while a slab is a density over continuous domain; their mixture forms a prior.

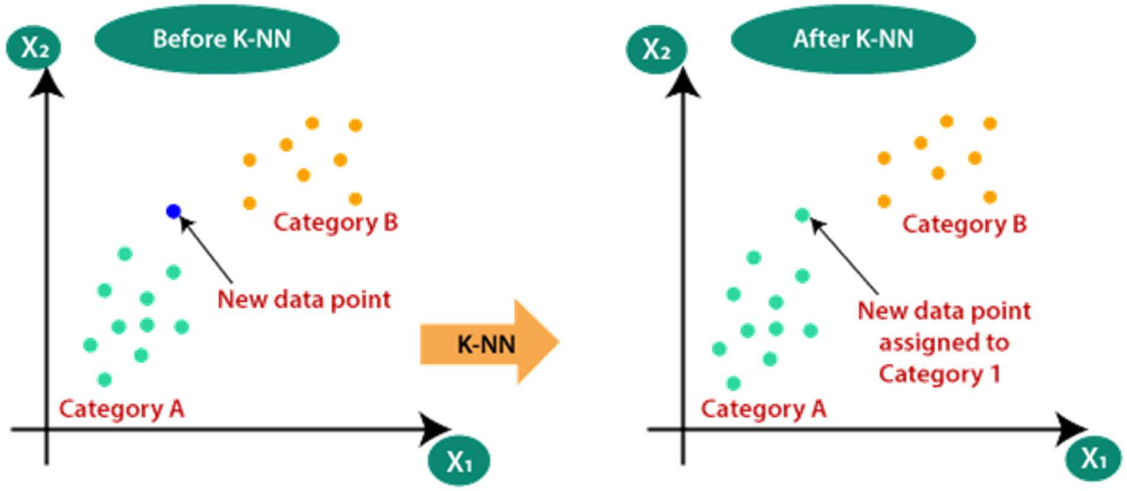
[2019 Fall 4.b]

Why K-Nearest Neighbors Algorithm is known as Non-Parametric Lazy algorithm? Explain with example.

Ans:

- K-Nearest Neighbor is one of the simplest Machine Learning algorithms based on Supervised Learning technique.
- K-NN algorithm assumes the similarity between the new case/data and available cases and put the new case into the category that is most similar to the available categories.
- K-NN algorithm stores all the available data and classifies a new data point based on the similarity. This means when new data appears then it can be easily classified into a well suite category by using K- NN algorithm.
- K-NN algorithm can be used for Regression as well as for Classification but mostly it is used for the Classification problems.

- **K-NN is a non-parametric algorithm**, which means it does not make any assumption on underlying data.
- It is also called a lazy learner algorithm because it does not learn from the training set immediately instead it stores the dataset and at the time of classification, it performs an action on the dataset.
- KNN algorithm at the training phase just stores the dataset and when it gets new data, then it classifies that data into a category that is much similar to the new data.
- Example: Suppose, we have an image of a creature that looks similar to cat and dog, but we want to know either it is a cat or dog. So, for this identification, we can use the KNN algorithm, as it works on a similarity measure. Our KNN model will find the similar features of the new data set to the cats and dogs images and based on the most similar features it will put it in either cat or dog category.



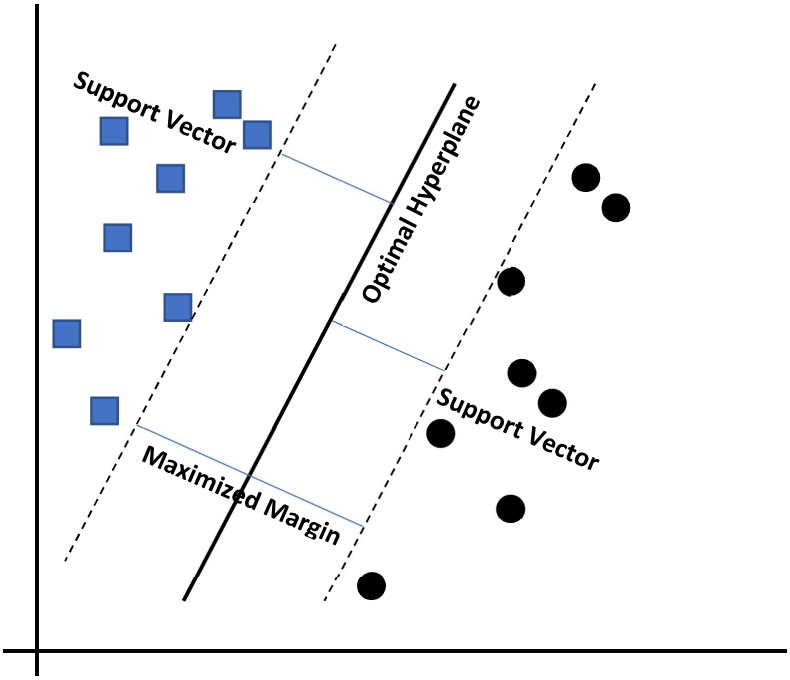
[2019 Fall 5.a]

How can we find the right hyperplane in SVM? Using K-Means clustering, cluster the following data into two clusters and show each step.

{2, 4, 10, 12, 3, 20, 30, 11, 25}

Ans: A hyperplane in an n-dimensional Euclidean space is a flat, n-1 Dimensional subset of that space that divides the space into two disconnected parts.

According to the SVM algorithm we find the points closest to the line from both the classes. These points are called support vectors. Now, we compute the distance between the line and the support vectors. This distance is called the margin. Our goal is to maximize the margin. The hyperplane for which the margin is maximum is the optimal hyperplane.



Sol^{n}: Given data set K = {2, 4, 10, 12, 3, 20, 30, 11, 25}, where k = 2

First sort the given dataset for easier distribution between two cluster.

K = {2, 3, 4, 10, 11, 12, 20, 25, 30, 25, 30}

How can you calculate Euclidean distance for one vector, this is?

$$\sqrt{(x-a)^2}$$

Let us assume Mean value (centroid)

M1 = 4 K1 = {2, 3, 4}	**M2 = 12** K2 = {10, 11, 12, 20, 25, 30}

So,

K	M1 = 4	M2 = 12	Cluster Assignment
2	**|4 – 2| = 2**	|12 – 2| = 10	K1
3	**|4 – 3| = 1**	|12 – 3| = 9	K1
4	**|4 – 4| = 0**	|12 – 4| = 8	K1
10	|4 – 10| = 6	**|12 – 10| = 2**	K2
11	|4 – 11| = 7	**|12 – 11| = 1**	K2
12	|4 – 12| = 8	**|12 – 12| = 0**	K2
20	|4 – 20| = 16	**|12 – 20| = 6**	K2
25	|4 – 25| = 21	**|12 – 25| = 8**	K2
26	|4 – 30| = 26	**|12 – 30| = 18**	K2

Now, Calculate Mean value (Centroid)

M1 = $\frac{2+3+4}{3} = \frac{9}{3} = \mathbf{3}$	**M2** = $\frac{10+11+12+20+25+30}{6} = \frac{108}{6} = \mathbf{18}$

Now, Update the cluster centroid and calculate Euclidean distance:

K	M1 = 3	M2 = 18	Cluster Assignment
2	**|3 – 2| = 1**	|18 – 2| = 16	K1
3	**|3 – 3| = 0**	|18 – 3| = 15	K1
4	**|3 – 4| = 1**	|18 – 4| = 14	K1
10	**|3 – 10| = 7**	|18 – 10| = 8	K1
11	|3 – 11| = 8	**|18 – 11| = 7**	K2
12	|3 – 12| = 9	**|18 – 12| = 6**	K2
20	|3 – 20| = 17	**|18 – 20| = 2**	K2
25	|3 – 25| = 22	**|18 – 25| = 7**	K2
26	|3 – 30| = 27	**|18 – 30| = 12**	K2

M1 = 3 K1 = {2, 3, 4, 10}	**M2 = 18** K2 = {11, 12, 20, 25, 30}

Now, Calculate Mean value (Centroid)

M1 = $\frac{2+3+4+10}{4} = \frac{19}{4} = \mathbf{4.75}$	**M2** = $\frac{11+12+20+25+30}{5} = \frac{98}{5} = \mathbf{19.6}$

Now, Update the cluster centroid and calculate Euclidean distance:

K	M1 = 4.75 = 5	M2 = 19.6 = 20	Cluster Assignment
2	**|5 – 2| = 3**	|20 – 2| = 18	K1
3	**|5 – 3| = 2**	|20 – 3| = 17	K1
4	**|5 – 4| = 1**	|20 – 4| = 16	K1
10	**|5 – 10| = 5**	|20 – 10| = 10	K1
11	**|5 – 11| = 6**	|20 – 11| = 9	K1
12	**|5 – 12| = 7**	|20 – 12| = 8	K1
20	|5 – 20| = 15	**|20 – 20| = 0**	K2
25	|5 – 25| = 20	**|20 – 25| = 5**	K2
26	|5 – 30| = 25	**|20 – 30| = 10**	K2

M1 = 5 K1 = {2, 3, 4, 10, 11, 12}	**M2 = 20** K2 = {20, 25, 30}

Now, Calculate Mean value (Centroid)

M1 = $\frac{2+3+4+10+11+12}{6} = \frac{42}{6} = \mathbf{7}$	**M2** = $\frac{20+25+30}{3} = \frac{75}{3} = \mathbf{25}$

Now, Update the cluster centroid and calculate Euclidean distance:

K	M1 = 7	M2 = 25	Cluster Assignment
2	**|7 – 2| = 5**	|25 – 2| = 23	K1
3	**|7 – 3| = 4**	|25 – 3| = 22	K1
4	**|7 – 4| = 3**	|25 – 4| = 21	K1
10	**|7 – 10| = 3**	|25 – 10| = 15	K1
11	**|7 – 11| = 4**	|25 – 11| = 14	K1
12	**|7 – 12| = 5**	|25 – 12| = 13	K1
20	|7 – 20| = 13	**|25 – 20| = 5**	K2
25	|7 – 25| = 18	**|25 – 25| = 0**	K2
26	|7 – 30| = 23	**|25 – 30| = 5**	K2

M1 = 5 K1 = {2, 3, 4, 10, 11, 12}	**M2 = 20** K2 = {20, 25, 30}

Now, Calculate Mean value (Centroid)

M1 = $\frac{2+3+4+10+11+12}{6} = \frac{42}{6} = \mathbf{7}$	**M2** = $\frac{20+25+30}{3} = \frac{75}{3} = \mathbf{25}$

Thus, we are getting the same mean value (Centroid). So we have to stop.

Therefore, new cluster is:

K1 = {2, 3, 4, 10, 11, 12}

K2 = {20, 25, 30}

----------X----------X----------

[2019 Spring 5.a]

What is Machine Learning and Why is Machine Learning important? Discuss K-Means Clustering for following dataset with K=2.

Staff ID	Age	No. of Leaves
1	21	3
2	39	7
3	43	6
4	58	12
5	32	15

Ans: Machine Learning is a field of computer science which gives the computer the ability to learn without being explicitly programmed. Machine Learnings main focus is to provide algorithms which can be trained to perform a task.

It is closely related to the field of computational statistics as well as mathematical optimization. It contains multiple methods like Supervised Learning, Unsupervised Learning, Semi-supervised Learning, and Reinforcement Learning which each has their own use cases and algorithms.

Machine learning (ML) is a method of data analysis that automates analytical model building. It is a branch of artificial intelligence based on the idea that systems can learn from data, identify patterns and make decisions with minimal human intervention.

Machine Algorithm types:

Machine learning algorithms can be organized based on the desired outcome of the algorithm or the type of input available during training the machine.

1. Supervised learning algorithms are trained on labeled examples, i.e., input where the desired output is known.

2. Unsupervised learning algorithms operate on unlabeled examples, i.e., input where the desired output is unknown.

3. Semi-supervised learning combines both labeled and unlabeled examples to generate an appropriate function or classifier.

4. Reinforcement learning is concerned with how intelligent agents ought to act in an environment to maximize some notion of reward from sequence of actions.

Machine Learning is important because of following points:

- Some tasks cannot be defined well, except by examples (e.g. recognizing people)
- Relationship and correlations can be hidden within large amounts of data. Machine learning may be able to find these relationships.
- Human designers often produce machines that do not work as well as desired in the environments in which they are used.
- The amount of knowledge available about certain tasks might be too large for explicit encoding by humans (e.g. Medical diagnostic).
- Environment changes over time.
- New knowledge about tasks is constantly being discovered by humans. It may be difficult to continuously re-design system "by hand".

Sol^n: Given that, K=2

Let, Age=X and Nos of Leaves = Y

Assume that: Initial mean value (centroid) from the given data sets be,

Staff ID	Age (X)	No. of Leaves (Y)
1	21	3
2	39	7
3	43	6
4	58	12
5	32	15

Cluster	Individual	Mean Value / Centroid
K1 (Group 1)	1	(21, 3)
K2 (Group 2)	2	(39, 7)

The remaining individual are now examined in sequence and allocated to the cluster to which they are closest in terms of Euclidean distance to the cluster mean. The mean value (Centroid) is re-calculated each time a new member is added.

We know, Euclidean distance

$$d = \sqrt{(x_n - x_m)^2 + (y_n - y_m)^2}$$

X	Y	K1 = (21, 3) $\sqrt{(X-21)^2+(Y-3)^2}$	K2 = (39, 7) $\sqrt{(X-39)^2+(Y-7)^2}$	Cluster Assignment
21	3	**0**	18.4	K1
39	7	18.4	**0**	K2
43	6	22.2	**4.1**	K2
58	12	38.1	**19.6**	K2
32	15	16.3	**10.6**	K2

Here how we are picking cluster is that look the minimum value by comparing K1 and K2,

if (K1<K2) then
    assign K1
else
    assign K2

K1 = {(21, 3)}

K2 = {(39, 7), (43, 6), (58, 12), (32, 15)}

Now, Calculate Mean value (Centroid)

$K1 = (X, Y) = \left(\frac{X_1}{1}\right), \left(\frac{Y_1}{1}\right)$

$= \left(\frac{21}{1}\right), \left(\frac{3}{1}\right)$

K1 = 21, 3

$K2 = (X, Y) = \left(\frac{X_1+X_2+X_3+X_4}{4}\right), \left(\frac{Y_1+Y_2+Y_3+Y_4}{4}\right)$

$= \left(\frac{39+43+58+3}{4}\right), \left(\frac{7+6+12+15}{4}\right)$

$= \left(\frac{172}{4}\right), \left(\frac{40}{4}\right)$

K2 = 43, 10

Now, Update the cluster centroid and calculate Euclidean distance:

X	Y	K1 = (21, 3) $\sqrt{(X-21)^2+(Y-3)^2}$	K2 = (43, 10) $\sqrt{(X-43)^2+(Y-10)^2}$	Cluster Assignment
21	3	**0**	23.1	K1
39	7	18.4	**5**	K2
43	6	22.2	**4**	K2
58	12	38.1	**15.1**	K2
32	15	16.3	**12.1**	K2

Here how we are picking cluster is that look the minimum value by comparing K1 and K2,

if (K1<K2) then
    assign K1
else
    assign K2

K1 = {(21, 3)}

K2 = {(39, 7), (43, 6), (58, 12), (32, 15)}

Now, Calculate Mean value (Centroid)

$$K1 = (X, Y) = \left(\frac{X_1}{1}\right), \left(\frac{Y_1}{1}\right)$$

$$= \left(\frac{21}{1}\right), \left(\frac{3}{1}\right)$$

K1 = 21, 3

$$K2 = (A, B) = \left(\frac{X_1+X_2+X_3+X_4}{4}\right), \left(\frac{Y_1+Y_2+Y_3+Y_4}{4}\right)$$

$$= \left(\frac{39+43+58+32}{4}\right), \left(\frac{7+6+12+15}{4}\right)$$

$$= \left(\frac{172}{4}\right), \left(\frac{40}{4}\right)$$

K2 = 43, 10

Thus, The Final Cluster is:

K1 = {(21, 3)}

K2 = {(39, 7), (43, 6), (58, 12), (32, 15)}

Model Question 1:

Apply K-means clustering for the following data sets for two clusters. Tabulate all the arguments.

Sample No.	X	Y
1	185	72
2	170	56
3	168	60
4	179	68
5	182	72
6	188	77

Solⁿ: Given, 1 and 4 as initial cluster center.

Sample No.	X	Y
1	185	72
2	170	56
3	168	60
4	179	68
5	182	72
6	188	77

Cluster	Individual	Mean Value / Centroid
K1 (Group 1)	1	(185, 72)
K2 (Group 2)	2	(170, 56)

The remaining individual are now examined in sequence and allocated to the cluster to which they are closest in terms of Euclidean distance to the cluster mean. The mean value (Centroid) is re-calculated each time a new member is added.

We know, Euclidean distance

$$d = \sqrt{(x_n - x_m)^2 + (y_n - y_m)^2}$$

X	Y	K1 = (185, 72) $\sqrt{(X-185)^2+(Y-72)^2}$	K2 = (170, 56) $\sqrt{(X-170)^2+(Y-56)^2}$	Cluster Assignment
185	72	**0**	21.93	K1
170	56	21.93	**0**	K2
168	60	20.80	**4.47**	K2
179	68	**7.21**	15	K1
182	72	**3**	20	K1
188	77	**5.83**	27.65	K1

Here how we are picking cluster is that look the minimum value by comparing K1 and K2,

if (K1<K2) then
    assign K1
else
    assign K2

K1 = {(185, 72), (179, 68), (182, 72), (188, 77)}

K2 = {(170, 56), (168, 60)}

Now, Calculate Mean value (Centroid)

$$K1 = (X, Y) = \left(\frac{X_1+X_2+X_3+X_4}{4}\right), \left(\frac{Y_1+Y_2+Y_3+Y_4}{4}\right)$$

$$= \left(\frac{185+179182+188}{4}\right), \left(\frac{72+68+72+77}{4}\right)$$

$$= \left(\frac{734}{4}\right), \left(\frac{289}{4}\right)$$

K1 = 183.5, 72.25

$$K2 = (X, Y) = \left(\frac{X_1+X_2}{2}\right), \left(\frac{Y_1+Y_2}{2}\right)$$

$$= \left(\frac{170+1}{2}\right), \left(\frac{56+6}{2}\right)$$

$$= \left(\frac{338}{2}\right), \left(\frac{116}{2}\right)$$

K2 = 169, 58

Now, Update the cluster centroid and calculate Euclidean distance:

X	Y	K1 = (183.5, 72.25) $\sqrt{(X-183.5)^2+(Y-72.25)^2}$	K2 = (169, 58) $\sqrt{(X-169)^2+(Y-58)^2}$	Cluster Assignment
185	72	**1.52**	21.26	K1
170	56	21.12	**2.23**	K2
168	60	19.75	**2.23**	K2
179	68	**6.189**	14.14	K1
182	72	**1.52**	19.10	K1
188	77	**6.54**	26.87	K1

K1 = {(185, 72), (179, 68), (182, 72), (188, 77)}

K2 = {(170, 56), (168, 60)}

Now, Calculate Mean value (Centroid)

$$K1 = (X, Y) = \left(\frac{X_1+X_2+X_3+X_4}{4}\right), \left(\frac{Y_1+Y_2+Y_3+Y_4}{4}\right)$$

$$= \left(\frac{185+179182}{4}\right), \left(\frac{72+68+72+77}{4}\right)$$

$$= \left(\frac{734}{4}\right), \left(\frac{289}{4}\right)$$

K1 = 183.5, 72.25

$$K2 = (X, Y) = \left(\frac{X_1+X_2}{2}\right), \left(\frac{Y_1+Y_2}{2}\right)$$

$$= \left(\frac{170+168}{2}\right), \left(\frac{56+60}{2}\right)$$

$$= \left(\frac{338}{2}\right), \left(\frac{116}{2}\right)$$

K2 = 169, 58

Thus, we are getting the same mean value (Centroid). So, we have to stop.

Therefore, new cluster is:

K1 = {(185, 72), (179, 68), (182, 72), (188, 77)}

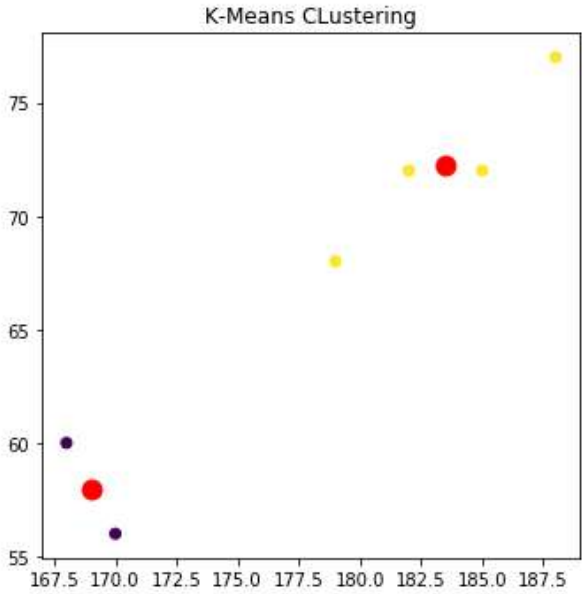
K2 = {(170, 56), (168, 60)}

----------X----------X----------

Python:

```
In [1]: import numpy as np
        from sklearn.cluster import KMeans

In [2]: X = np.array([[185, 72], [170, 56], [168, 60],[179, 68], [182, 72], [188, 77]])

In [3]: X_kmeans = KMeans(n_clusters=2).fit(X)

In [4]: X_kmeans.labels_
Out[4]: array([0, 1, 1, 0, 0, 0])

In [52]: #X_kmeans.predict([[150, 120], [12, 3]])

In [5]: X_kmeans.cluster_centers_
Out[5]: array([[183.5 ,  72.25],
               [169.  ,  58.  ]])
```

```
In [7]: import matplotlib.pyplot as plt

        plt.figure(figsize=(12, 12))

        X_pred = KMeans(n_clusters=2).fit_predict(X)

        plt.subplot(221)
        plt.scatter(X[:, 0], X[:, 1], c=X_pred)
        plt.scatter(X_kmeans.cluster_centers_[:,0],X_kmeans.cluster_centers_[:,1],s=120,c='red')
        plt.title("K-Means CLustering")
        plt.show()
```



Try Yourself

(2, 3), (5, 6), (8, 7), (1, 4), (2, 2), (6, 7), (3, 4), (8, 6) where k=2

Soln:

Hint:

Assume the mean value (centroid) from the given data sets,

K1 = (2, 3)

K2 = (5, 6)

THE END