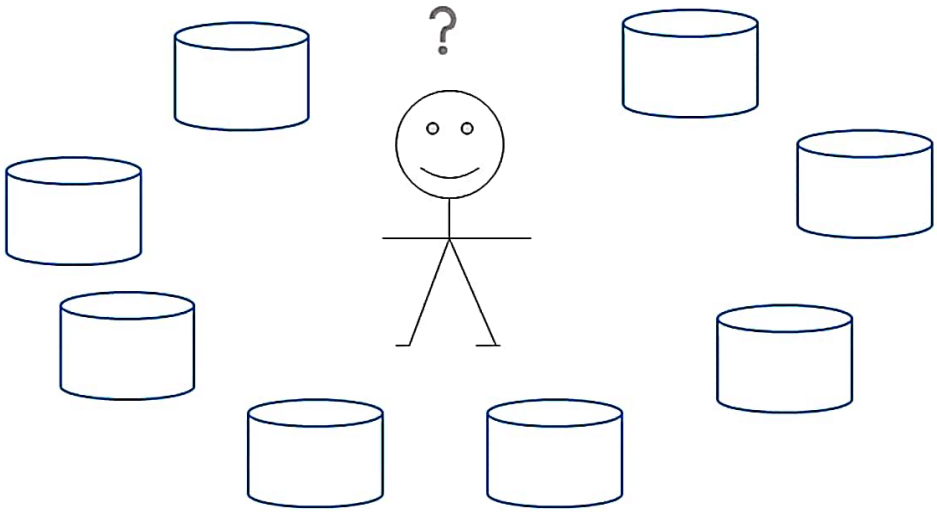
Data Mining

- Data mining is the process to extracting or mining relevant knowledge from large amounts of data. For example, mining of gold from rock, and mining diamond from coal.



Knowledge Discovery from Data (KDD)

- The knowledge discovery process have following steps:
  1. Data cleaning: To remove noise and inconsistent data.
  2. Data integration: Multiple data sources may be combined.
  3. Data selection: Data relevant to the analysis task are retrieved from the database.
  4. Data transformation: Data are transformed and consolidated into forms appropriate for mining by performing summary or aggregation operations.
  5. Data mining: An essential process where intelligent methods are applied to extract data patterns.

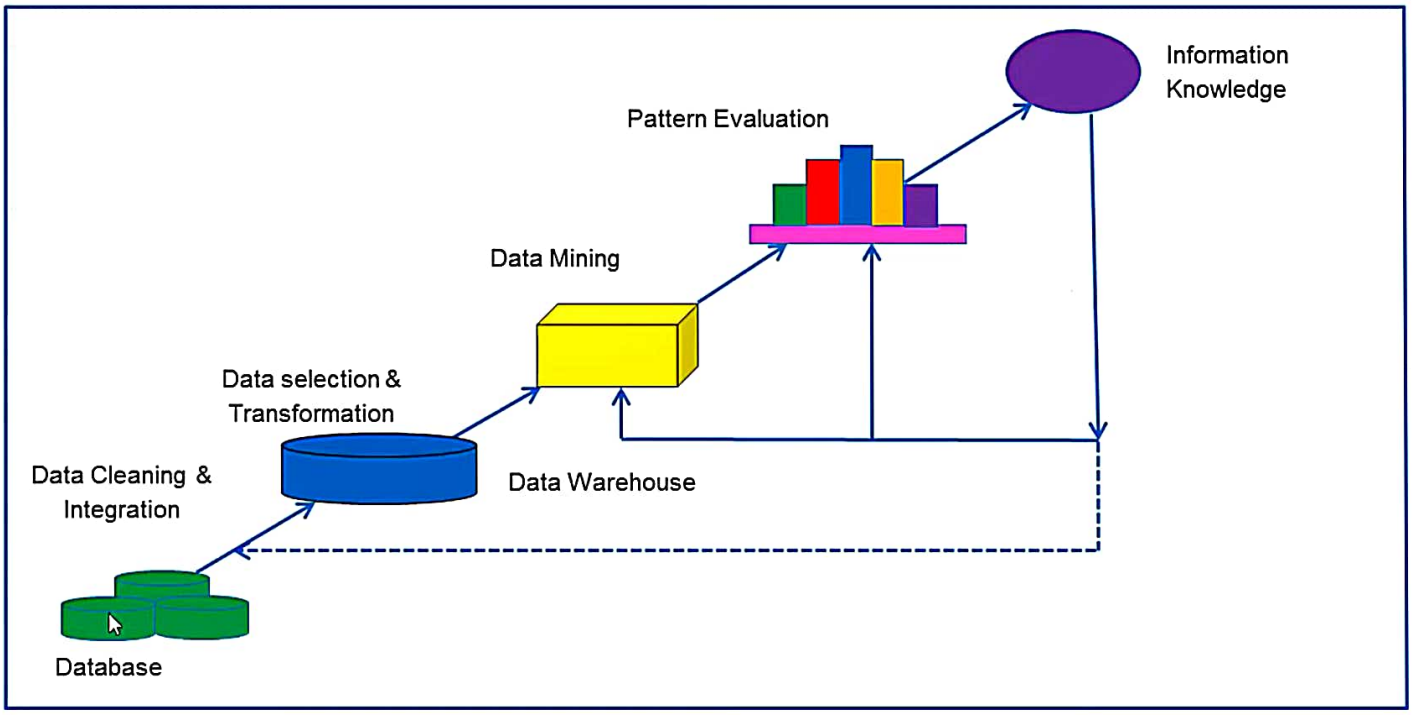
Knowledge Discovery from Data (KDD)

- The knowledge discovery process have following steps:
  1. Data cleaning: To remove noise and inconsistent data.
  2. Data integration: Multiple data sources may be combined.
  3. Data selection: Data relevant to the analysis task are retrieved from the database.
  4. Data transformation: Data are transformed and consolidated into forms appropriate for mining by performing summary or aggregation operations.
  5. Data mining: An essential process where intelligent methods are applied to extract data patterns.

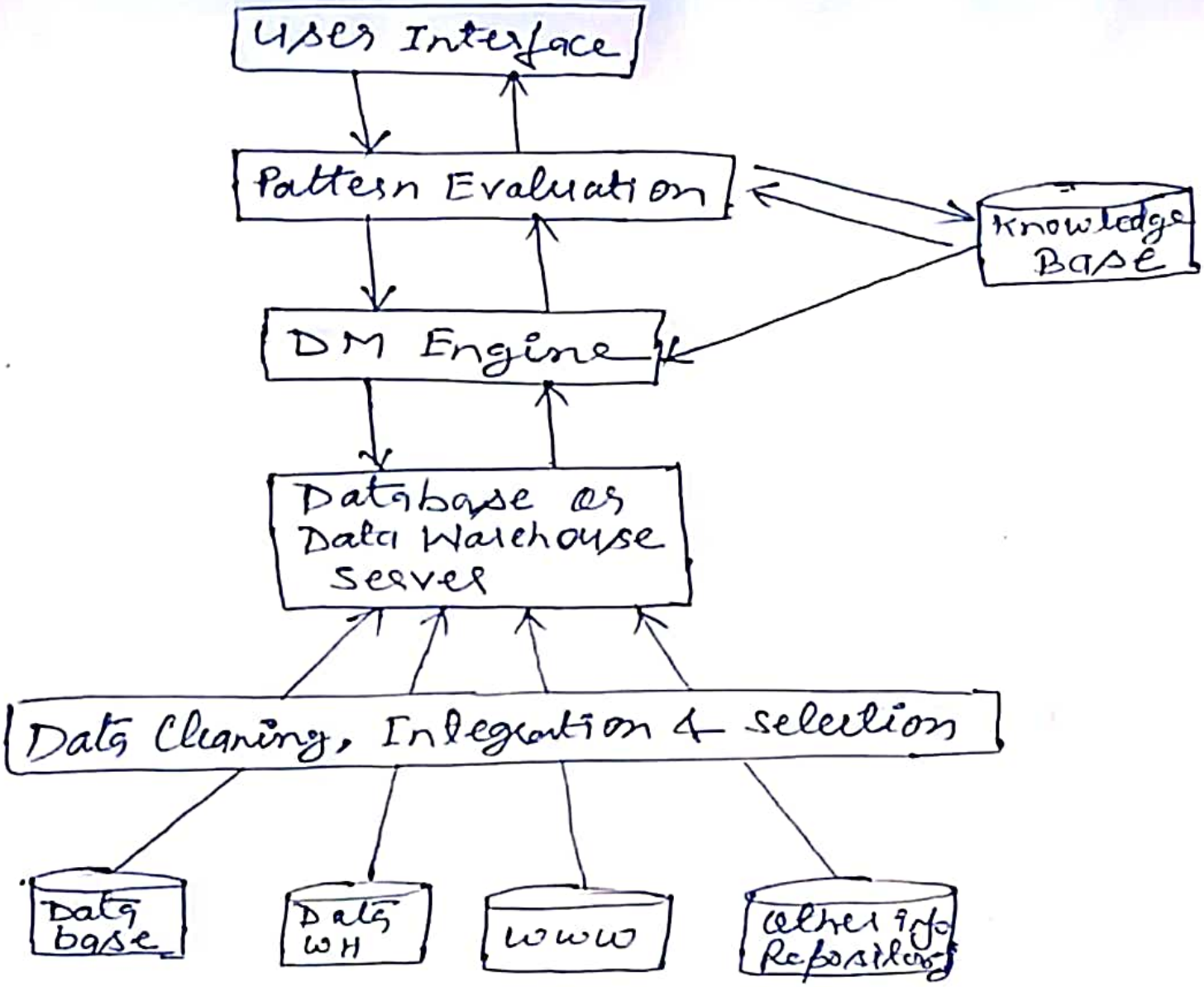
KDD

6. Pattern evaluation: To identify the truly interesting patterns representing knowledge based on interestingness measures.
7. Knowledge presentation: Visualization and knowledge representation techniques are used to present mined knowledge to users.

Data mining steps to discover knowledge from data



Data mining system architecture



Data Mining Architecture

(1) Data Sources:

Database, World Wide Web(WWW) and data warehouse are parts of data sources. The data in these sources may be in the form of plain text, spreadsheets or in other forms of media like photos or videos. WWW is one of the biggest sources of data.

(2) Database Server:

The database server contains the actual data ready to be processed. It performs the task of handling data retrieval as per the request of the user.

(3) Data Mining Engine:

It is one of the core components of the data mining architecture that performs all kinds of data mining techniques like association, classification, characterization, clustering, prediction, etc.

Data Mining Architecture

Please move this window away from

(4)Pattern Evaluation Modules:

They are responsible for finding interesting patterns in the data and sometimes they also interact with the database servers for producing the result of the user requests.

(5)Graphic User Interface:

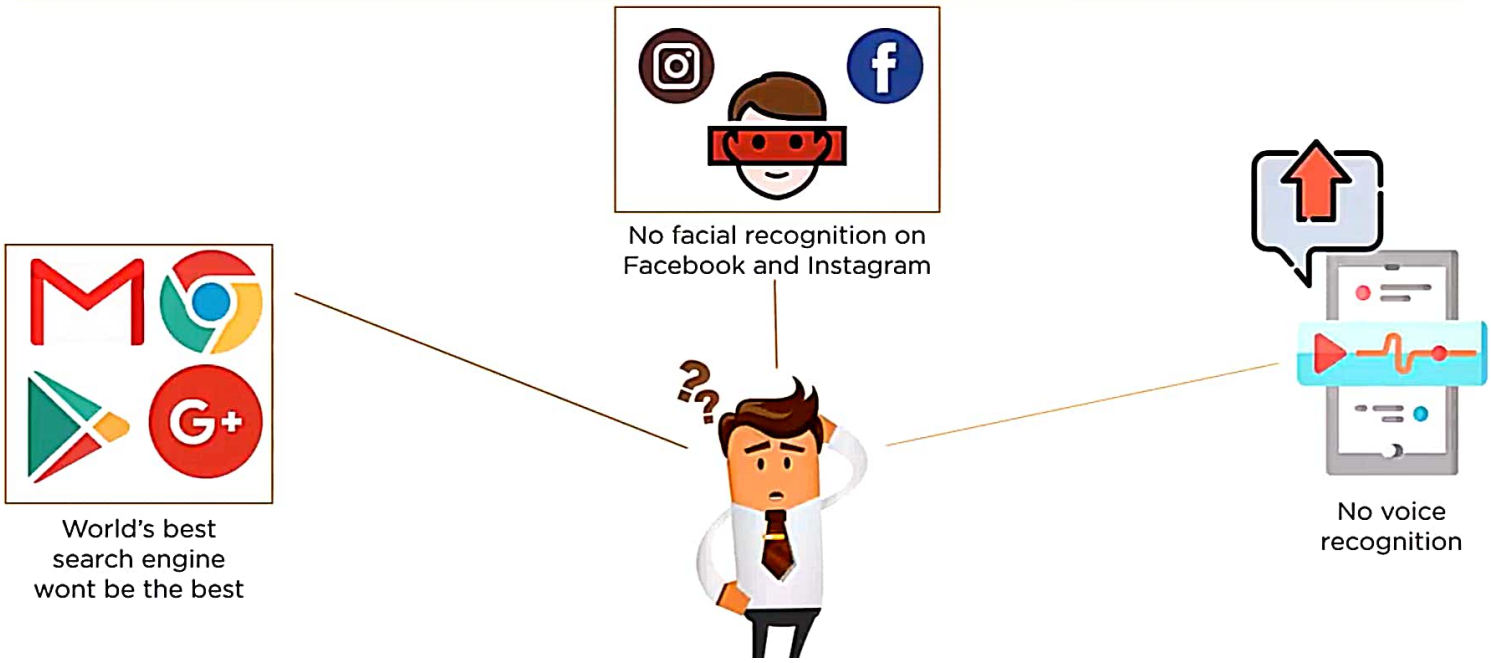
Since the user cannot fully understand the complexity of the data mining process so graphical user interface helps the user to communicate effectively with the data mining system.

Data Mining Architecture

(6) Knowledge Base:

Knowledge Base is an important part of the data mining engine that is quite beneficial in guiding the search for the result patterns. Data mining engine may also sometimes get inputs from the knowledge base. This knowledge base may contain data from user experiences. The objective of the knowledge base is to make the result more accurate and reliable.

Life without Machine Learning



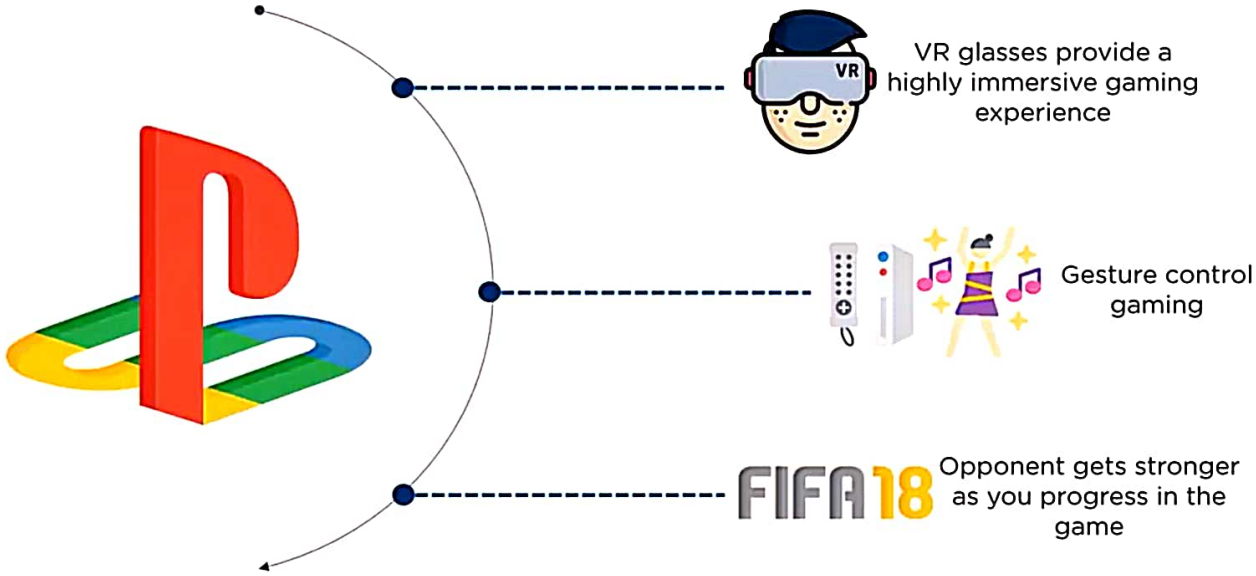
Life with Machine Learning

Shopping made easy

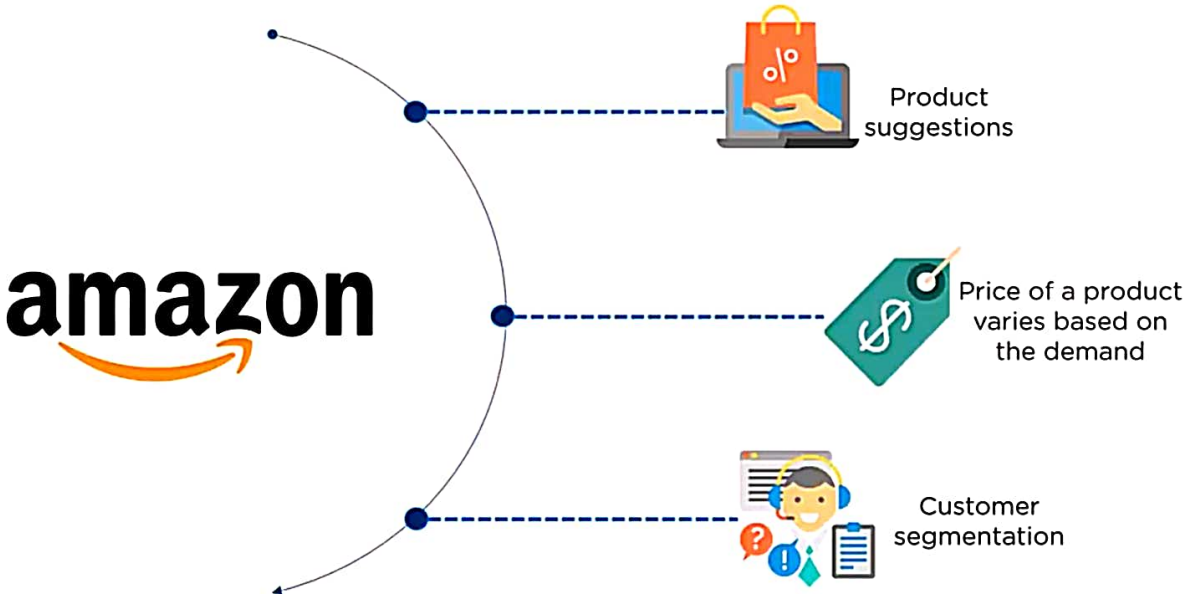
Connecting with old friends

Get the best movie suggestions

Life with Machine Learning

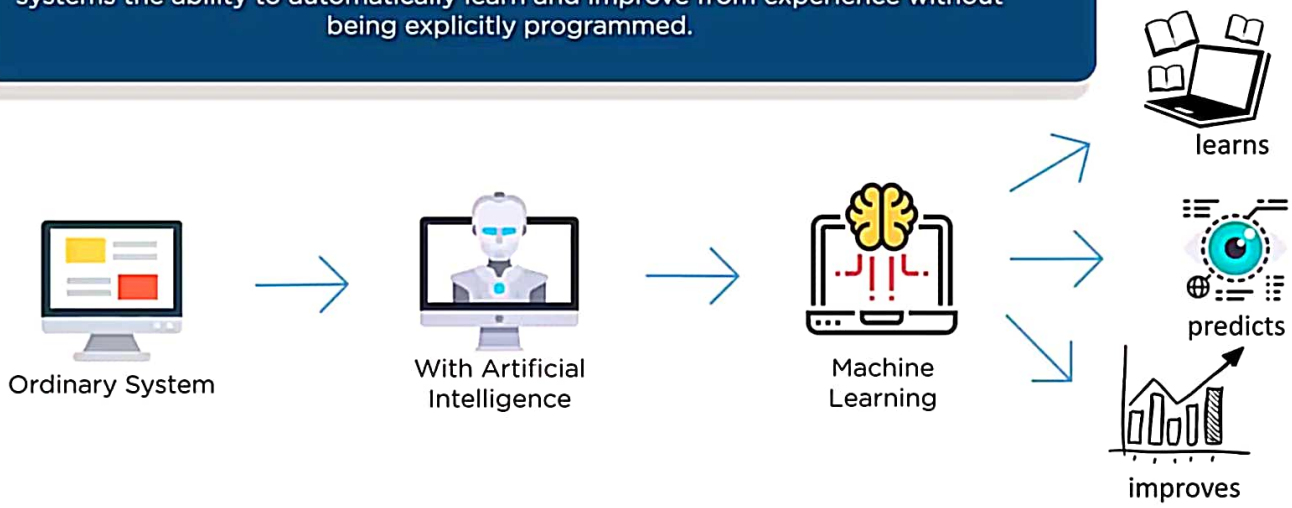


Life with Machine Learning

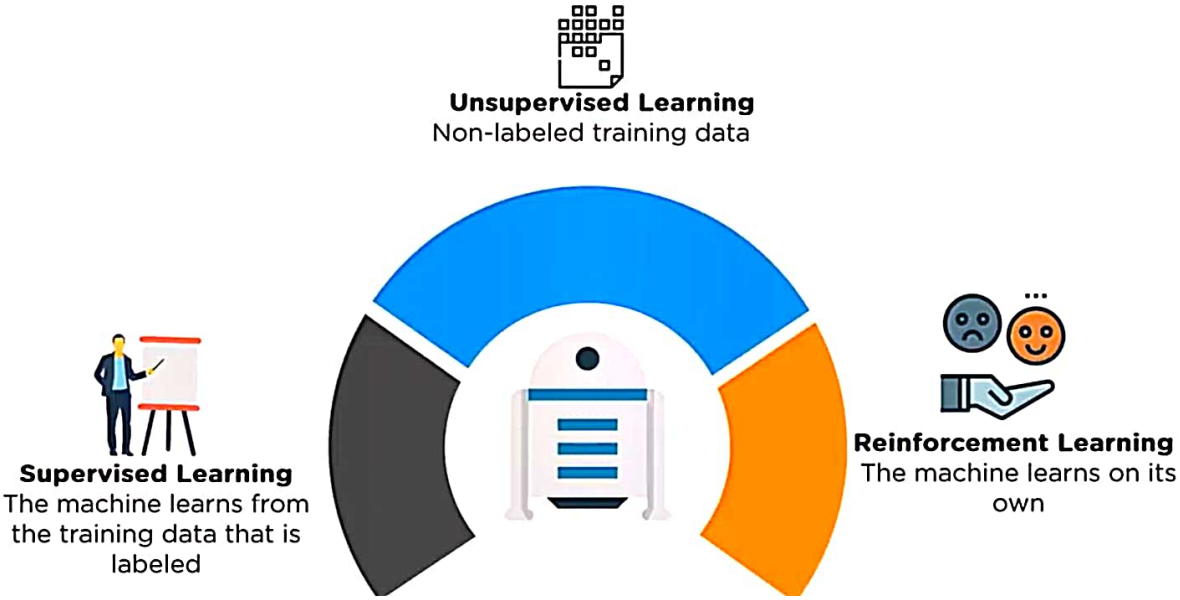


What is Machine Learning?

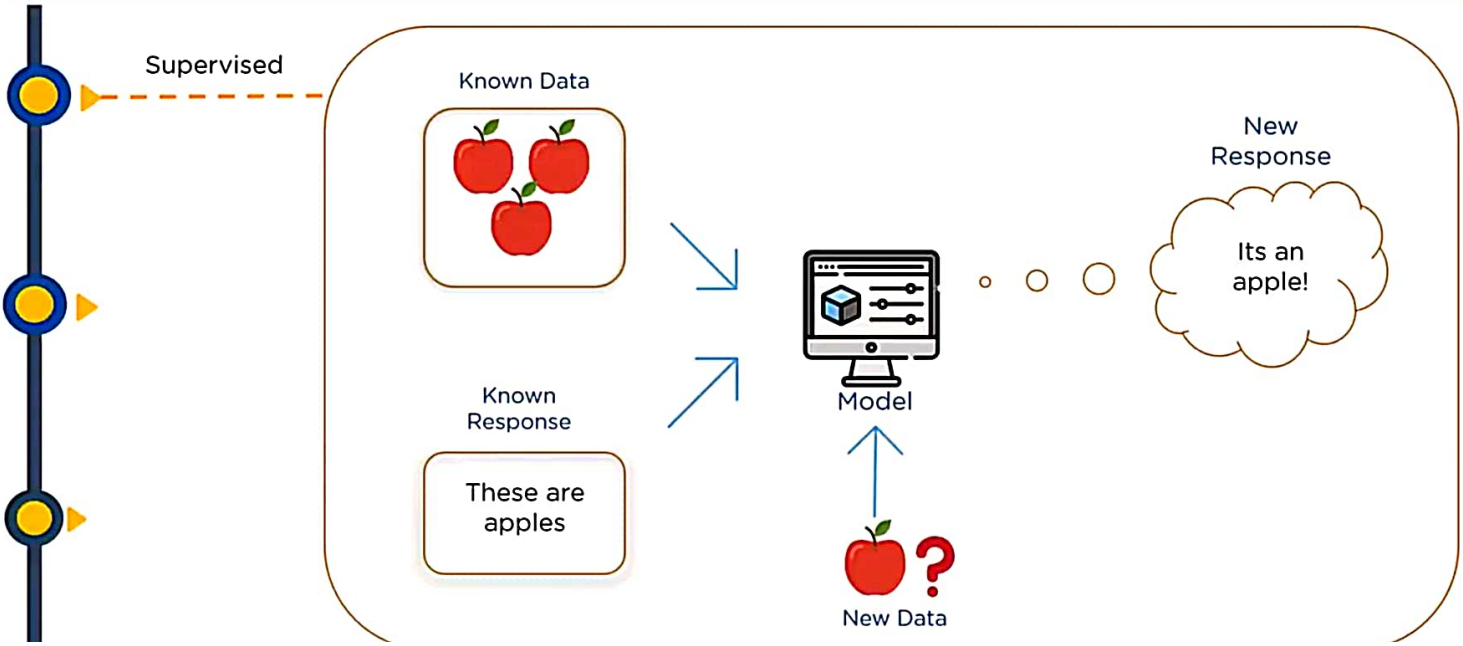
Machine Learning is an application of Artificial intelligence (AI) that provides systems the ability to automatically learn and improve from experience without being explicitly programmed.



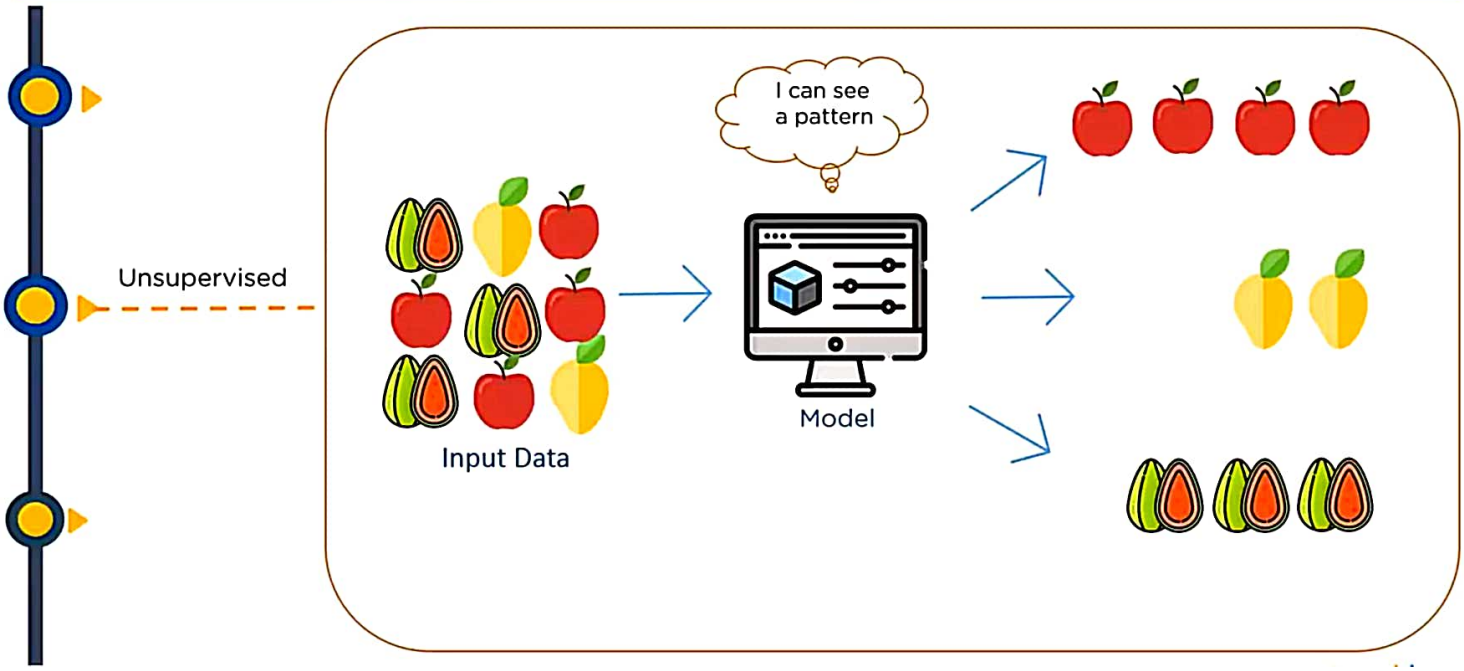
Types of Machine Learning



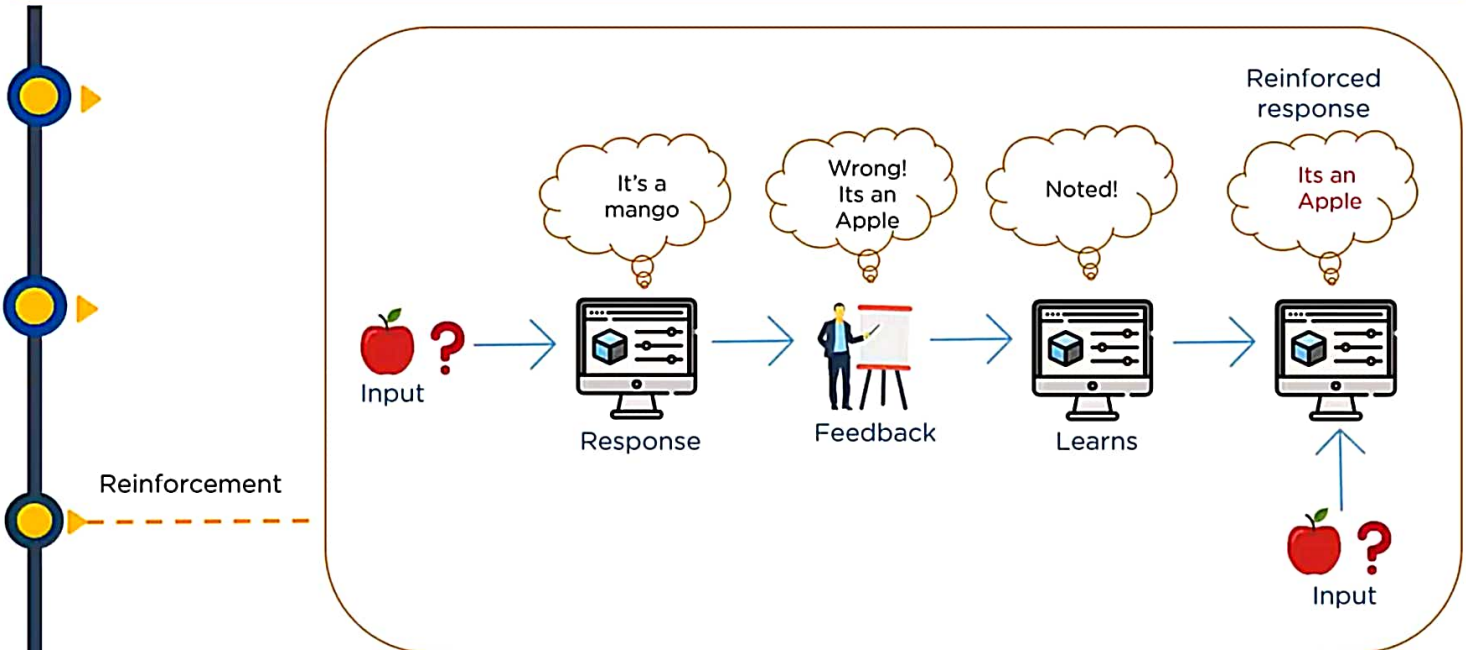
Types of Machine Learning



Types of Machine Learning



Types of Machine Learning



What is Classification?

- Classification is a form of data analysis that extracts models(classifiers) describing important data classes.

- It classifies the data into two or more categories based on class labels.
  (Yes/No, Male/Female, Positive/Negative, etc.)

- Example – Bank loan approval to customers based on the customers' age

Answer

•Classification is the process of building a classifier model based on known data (i.e. class labels).

•Prediction is the process to use the classifier model to know the outcome or value of the class label of unknown data.

Process of Classification

Classification is a two step process –

1. **Training the classifier / Model construction**
2. **Testing the classifier / Model usage**

Step 1: Training

•The model has to be trained by describing a set of predetermined classes in the training dataset.

•Each tuple/sample from the training dataset is assumed to belong to a predefined class, as determined by the class label attribute.

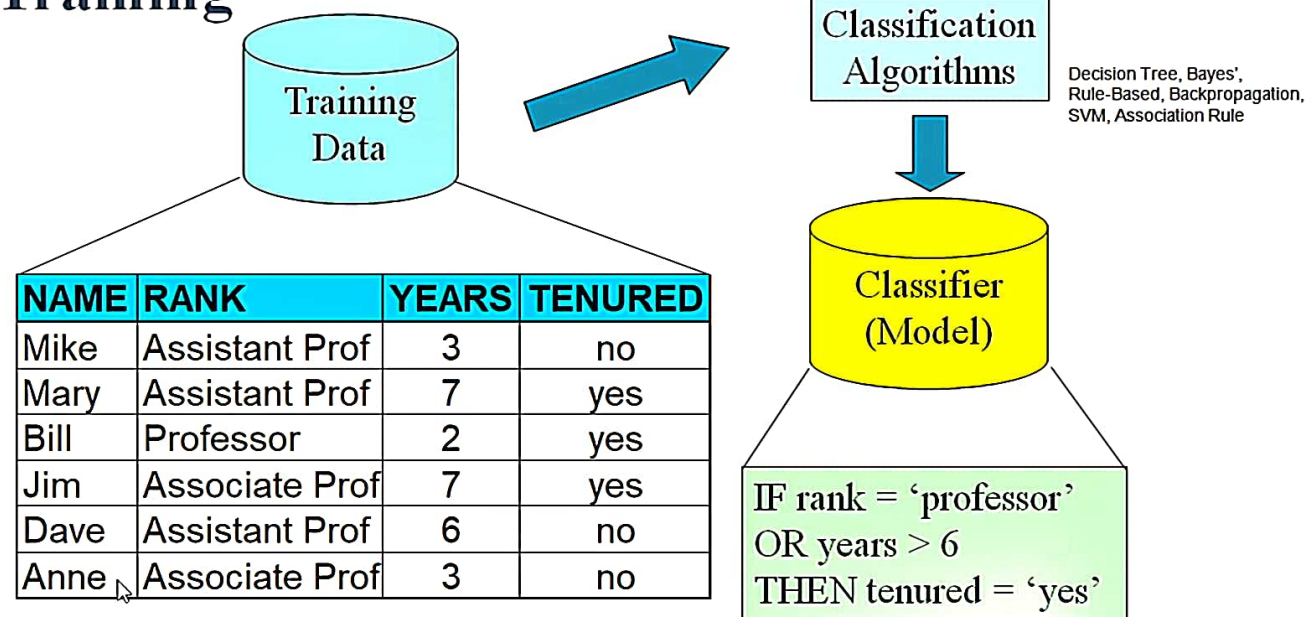
•The set of tuples used for model construction is training set.

•The model is represented as classification rules, decision trees, or mathematical formulae.

Step 2: Testing

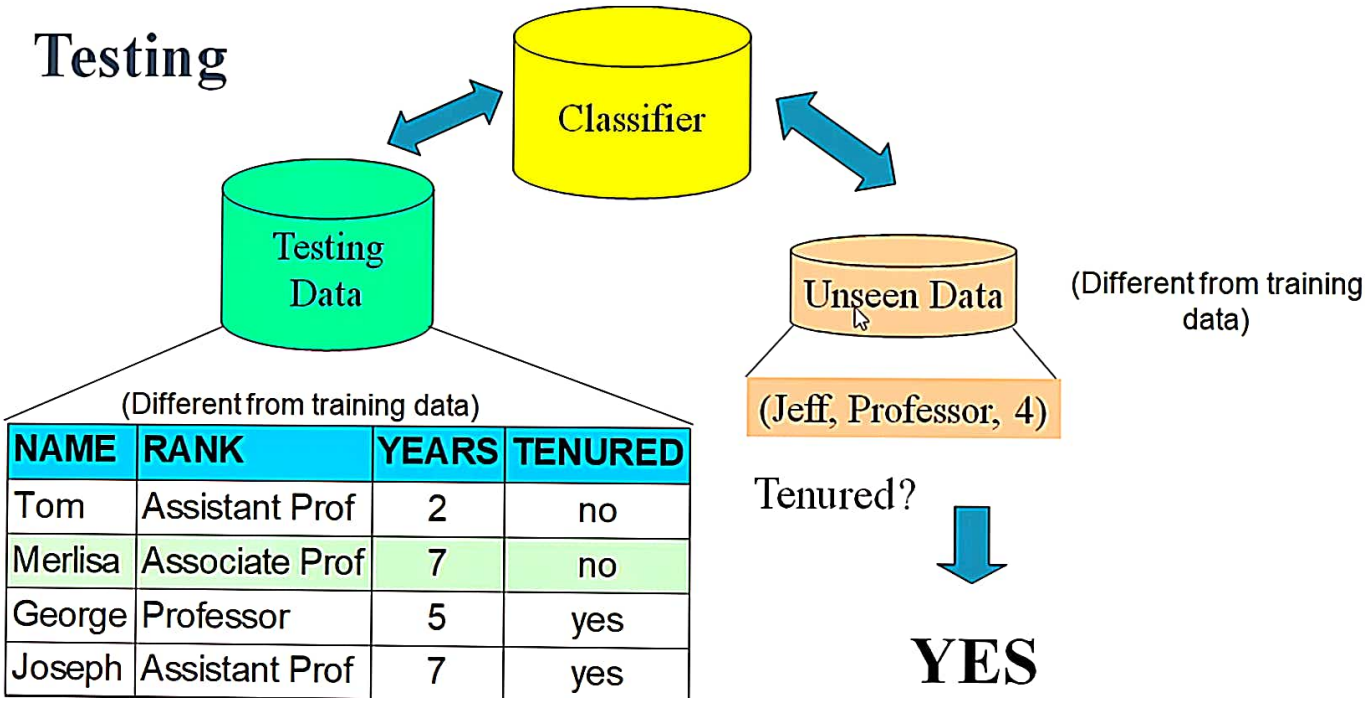
- Testing is done to predict class categories of data tuples which are unknown to the model.
- The known label of test sample is compared with the classified result from the model.
- Accuracy rate is the percentage of test set samples that are correctly classified by the model.
- The test dataset is independent of training set to judge the model's accuracy.
- If the accuracy is acceptable, use the model to classify new data.

Training

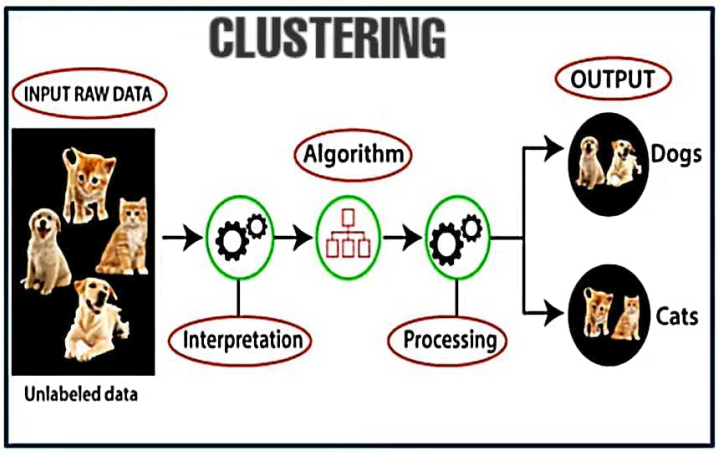


NAME	RANK	YEARS	TENURED
Mike	Assistant Prof	3	no
Mary	Assistant Prof	7	yes
Bill	Professor	2	yes
Jim	Associate Prof	7	yes
Dave	Assistant Prof	6	no
Anne	Associate Prof	3	no

Testing



NAME	RANK	YEARS	TENURED
Tom	Assistant Prof	2	no
Merlisa	Associate Prof	7	no
George	Professor	5	yes
Joseph	Assistant Prof	7	yes



- clustering is framed in **unsupervised learning**; that is, for this type of algorithm we only have one set of input data (not labelled), about which we must obtain information, without previously knowing what the output will be.

- Algorithms take data only containing inputs and then **add structure to the data in the form of clustering or grouping.** Cluster analysis uses unsupervised learning to sort through giant lakes of raw data to group certain data points together.
- **Unsupervised learning infers from unlabelled data, a function that describes hidden structures in data.**

Types of Regression Models

- Regression modeling is a process of determining a relationship between one or more independent variables and one dependent or output variable.
- **Examples:**
  1. Predicting the height of a person given the age of the person.
  2. Predicting the price of the car given the car model, year of manufacturing, mileage, engine capacity, etc.

Types of Regression Models

- Based on the type of functions used to represent the relationship between the dependent or output variable and independent variables, the regression models are categorized into four types. The regression models are,

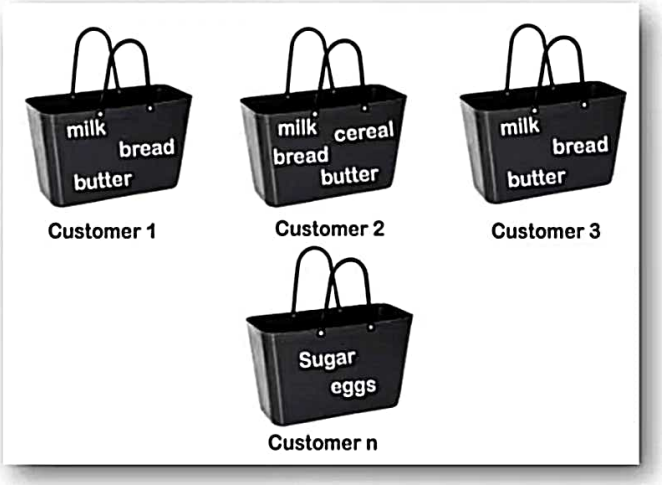
  1. Simple Linear Regression
  2. Multiple Regression
  3. Polynomial Regression
  4. Logistic Regression

Association Rule Learning

- Association rule learning is a type of unsupervised learning technique.
- It checks the dependency of one data item on another data item and maps accordingly so that it can be more profitable.
- It tries to find some interesting relations or associations among the variables of dataset.

Example: Market Basket Analysis

- In supermarket, all products that are purchased together are put together.
- If a customer buys bread, he most likely can also buy butter, eggs, or milk, so these products are stored within a shelf or mostly nearby.



How does Association Rule Learning work?

- Association rule learning works on the concept of <u>If and Else Statement, such as if A then B.</u>

- Here the <u>**If** element is called antecedent, and **then** statement is called as Consequent</u>.
- These types of relationships where we can <u>find out some association or relation between two items</u> is known as <u>single cardinality</u>

Association Rules Evaluation

1. Support:

- Support is the frequency of A or **how frequently an item appears** in the dataset.
- It is defined as the fraction of the transaction T that contains the itemset X.

$$\text{Supp(X)} = \frac{Freq(X)}{T}$$

$$\text{Rule: } X \Rightarrow Y$$

$$Support = \frac{frq(X,Y)}{N}$$

$$Confidence = \frac{frq(X,Y)}{frq(X)}$$

$$Lift = \frac{Support}{Supp(X) \times Supp(Y)}$$

2. Confidence:

- Confidence indicates how often the rule has been found to be true.
- Or how often the items **X and Y occur together** in the dataset when the occurrence of X is already given.

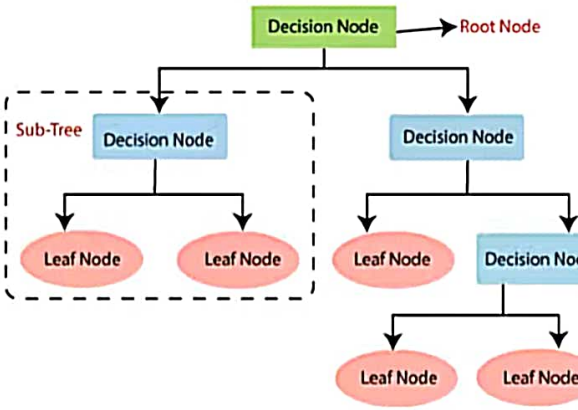
$$\text{Confidence} = \frac{Freq(X,Y)}{Freq(X)}$$

Introduction of Decision Tree

- Decision Tree is a **Supervised learning technique** that can be used for both classification and Regression problems, but mostly it is preferred for solving Classification problems.
- It is a tree-structured classifier, where **internal nodes represent the features of a dataset, branches represent the decision rules** and **external nodes represents the outcome.**

Decision Tree Terminologies

- **Root Node:** Root node is from where the decision tree starts. It represents the entire dataset, which further gets divided into two or more homogeneous sets.
- **Leaf Node:** Leaf nodes are the final output node, and the tree cannot be segregated further after getting a leaf node.
- **Decision nodes** are used to make any decision and have multiple branches.
- **Branch/Sub Tree:** A tree formed by splitting the tree.
- **Pruning:** Pruning is the process of removing the unwanted branches from the tree.
- **Parent/Child node:** The root node of the tree is called the parent node, and other nodes are called the child nodes.



Algorithm of Decision Tree

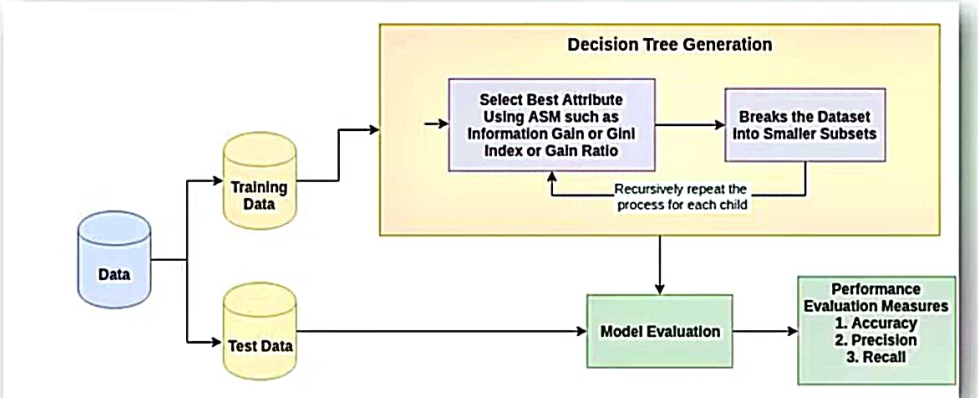
Step-1: Begin the tree with the root node, says S, which contains the complete dataset.

Step-2: Find the best attribute in the dataset using **Attribute Selection Measure (ASM).**

Step-3: Divide the S into subsets that contains possible values for the best attributes.

Step-4: Generate the decision tree node, which contains the best attribute.

Step-5: Recursively make new decision trees using the subsets of the dataset created in step -3. Continue this process until a stage is reached where you cannot further classify the nodes and called the final node as a leaf node.



What is a Sequence

- Sequences is set of Ordered Events
- Example : S = {e1,e2,e3,e4,e5}
- Here event e1 happened first and then e2 then e3 …. In an order

ATM Machine

- Insert Card
- Enter Pin
- Enter Withdrawal amount
- Collect Cash
- Remove Card

Sequences in market basket analysis

CID	TID	Transactions
1	100	a,b,c,d
3	111	a,f,d,c
1	122	d,e,p
3	133	b,f,s,a
1	144	b,c,d,e
3	155	a,f,d,c
1	166	a,e,p

We can represent Customer 1 all transactions in a transaction list called Sequence i.e

S = {(a,b,c,d),(d,e,p),(b,c,d,e),(a,e,p)}

Sequences in market basket analysis

CID	TID	Transactions
1	100	a,b,c,d
3	111	a,f,d,c
1	122	d,e,p
3	133	b,f,s,a
1	144	b,c,d,e
3	155	a,f,d,c
1	166	a,e,p

CID	Sequences
1	(a,b,c,d)(d,e,p)(b,c,d,e)(a,e,p)
3	(a,f,d,c)(b,f,s,a)(a,f,d,c)

Sequential Pattern Mining

- Finding Interesting(Frequent) Sub Sequences as patterns
- We find Frequent sequence's, Frequent Subsequeces
- Abnormalities in Sequences
- …..

Data Mining Apriori Algorithm in Machine Learning

- The Apriori algorithm uses frequent itemsets to generate association rules and it is designed to work on the databases that contain transactions.
- With the help of these association rule, it determines how strongly or how weakly two objects are connected.
- This algorithm was given by the **R. Agrawal** and **Srikant** in the year **1994**.
- It is mainly used for *market basket analysis* and helps to find those products that can be bought together.

What is Frequent Itemset?

- Frequent itemsets are those items whose support is greater than the threshold value or user-specified minimum support.
- Suppose there are the two transactions: A= {1,2,3,4,5}, and B= {2,3,7}, in these two transactions, 2 and 3 are the frequent itemsets.

Apriori Algorithm Working Example

Aim:

- Suppose we have the following dataset that has various transactions.
- From this dataset, we need to find the frequent itemsets and generate the association rules using the Apriori algorithm:

TID	ITEMSETS
T1	A, B
T2	B, D
T3	B, C
T4	A, B, D
T5	A, C
T6	B, C
T7	A, C
T8	A, B, C, E
T9	A, B, C

Given: Minimum Support= 2, Minimum Confidence= 50%

Example

TID	ITEMSETS
T1	A, B
T2	B, D
T3	B, C
T4	A, B, D
T5	A, C
T6	B, C
T7	A, C
T8	A, B, C, E
T9	A, B, C

Given: Minimum Support= 2, Minimum Confidence= 50%

Given Dataset

Solution:

➢ **Step-1:** Calculating **C1 and L1**

- We will create a table that contains support count (The frequency of each itemset individually in the dataset) of each itemset in the given dataset.
- This **table** is called the **Candidate set or C1.**

Itemset	Support_Count
A	6
B	7
C	5
D	2
E	1

C1

- Now, we will take out all the itemsets that have the greater support count that the Minimum Support (2).
- It will give us the **table** for the **frequent itemset L1.**
- Since all the itemsets have greater or equal support count than the minimum support, except the E, so E itemset will be removed.

Itemset	Support_Count
A	6
B	7
C	5
D	2

L1

Example

C2

Itemset	Support_Count
{A, B}	4
{A,C}	4
{A, D}	1
{B, C}	4
{B, D}	2
{C, D}	0

➢ **Step-2:** Calculating **C2 and L2**

- In this step, we will generate **C2** with the help of L1.
- In C2, we will create the pair of the itemsets of L1 in the form of subsets.
- After creating the subsets, we will again find the support count from the main transaction table of datasets, i.e., how many times these pairs have occurred together in the given dataset.

- Again, we need to compare the C2 Support count with the minimum support count, and after comparing, the itemset with less support count will be eliminated from the table C2.
- Now, Generate **L2 Table**.

Itemset	Support_Count
A	6
B	7
C	5
D	2

Step 1: L1 Table

TID	ITEMSETS
T1	A, B
T2	B, D
T3	B, C
T4	A, B, D
T5	A, C
T6	B, C
T7	A, C
T8	A, B, C, E
T9	A, B, C

Given: Minimum Support= 2, Minimum Confidence= 50%

Given Dataset

Itemset	Support_Count
{A, B}	4
{A, C}	4
{B, C}	4
{B, D}	2

A, B, C, D

L2

Example

➢ **Step-3:** Calculating **C3 and L3**

- For C3, we will repeat the same two processes, but now we will form the C3 table with subsets of three itemsets together, and will calculate the support count from the dataset.

C3

Itemset	Support_Count
{A, B, C}	2
{B, C, D}	1
{A, C, D}	0
{A, B, D}	0

- Now we will create the L3 table.
- As we can see from the above C3 table, there is **only one combination of itemset that has support count equal to the minimum support count.**
- So, the L3 will have only one combination, i.e., {A, B, C}.

Itemset	Support_Count
{A, B, C}	2

L3

Itemset	Support_Count
{A, B}	4
{A, C}	4
{B, C}	4
{B, D}	2

A, B, C, D

Step 2: L2 Table

TID	ITEMSETS
T1	A, B
T2	B, D
T3	B, C
T4	A, B, D
T5	A, C
T6	B, C
T7	A, C
T8	A, B, C, E
T9	A, B, C

Given: Minimum Support= 2, Minimum Confidence= 50%

Given Dataset

Example

➢ **Step-4**: Finding the **association rules** for the subsets:

- To generate the association rules, first, we will create a new table with the possible rules from the occurred combination **{A, B, C}. (L3 Table)**
- For all the rules, we will **calculate the Confidence** using formula **sup(A ^B)/A.** (Rules/LHS)
- After calculating the confidence value for all rules, we will exclude the rules that have less confidence than the minimum threshold (50%) (Given).

Example

Itemset	Support_Count
A	6
B	7
C	5
D	2

L1

Itemset	Support_Count
{A, B}	4
{A, C}	4
{B, C}	4
{B, D}	2

A, B, C, D

L2

Itemset	Support_Count
{A, B, C}	2

L3

<u>Confidence = Sup(A ^B)/A.</u> (Rules/LHS)

Consider the below table:

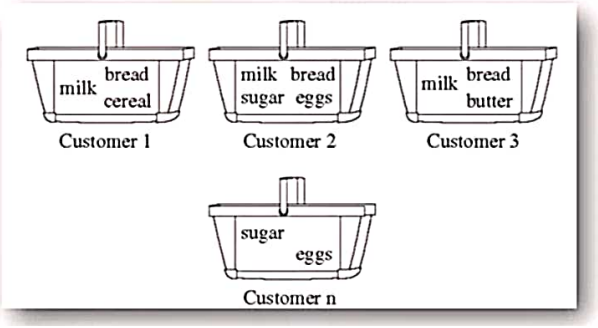
Rules	Support	Confidence
A ^B → C	2	Sup{(A ^B) ^C}/sup(A ^B)= 2/4=0.5=50%
B^C → A	2	Sup{(B^C) ^A}/sup(B ^C)= 2/4=0.5=50%
A^C → B	2	Sup{(A ^C) ^B}/sup(A ^C)= 2/4=0.5=50%
C→ A ^B	2	Sup{(C^(A ^B)}/sup(C)= 2/5=0.4=40%
A→ B^C	2	Sup{(A^(B ^C)}/sup(A)= 2/6=0.33=33.33%
B→ B^C	2	Sup{(B^(B ^C)}/sup(B)= 2/7=0.28=28%

- As the given dataset <u>minimum confidence is 50%.</u>
- so the **first three rules <u>A ^B → C, B^C → A, and A^C → B</u>** can be considered as **<u>the strong association rules</u>** <u>for the given problem</u>.

Applications of Association Rule Learning

1. **Market Basket Analysis:** It is one of the popular examples and applications of association rule mining. This technique is commonly used by big retailers to determine the association between items.

2. **Medical Diagnosis:** With the help of association rules, patients can be cured easily, as it helps in identifying the probability of illness for a particular disease.



Advantages & Disadvantages

- **Advantages of Apriori Algorithm:**

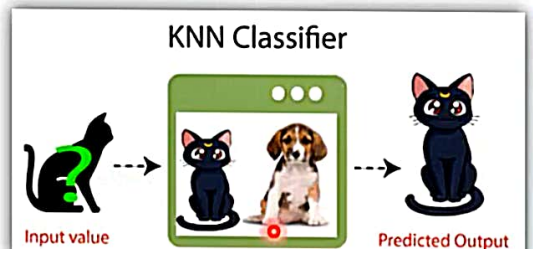
1. This is easy to understand algorithm
2. The join and prune steps of the algorithm can be easily implemented on large datasets.

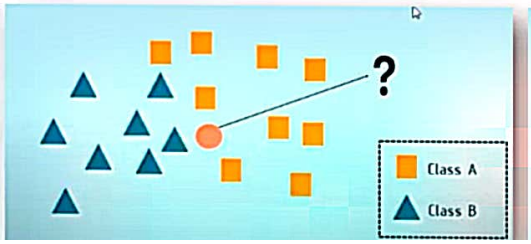
- **Disadvantages of Apriori Algorithm:**

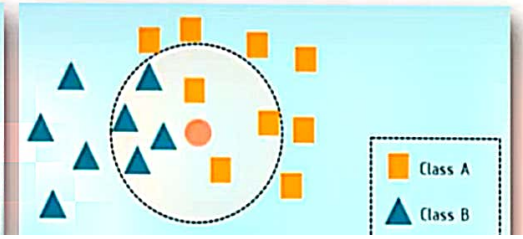
1. The apriori algorithm works slow compared to other algorithms.
2. The overall performance can be reduced as it scans the database for multiple times.
3. The time complexity and space complexity high some times.

KNN Introduction

- K-Nearest Neighbour 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.
- 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.

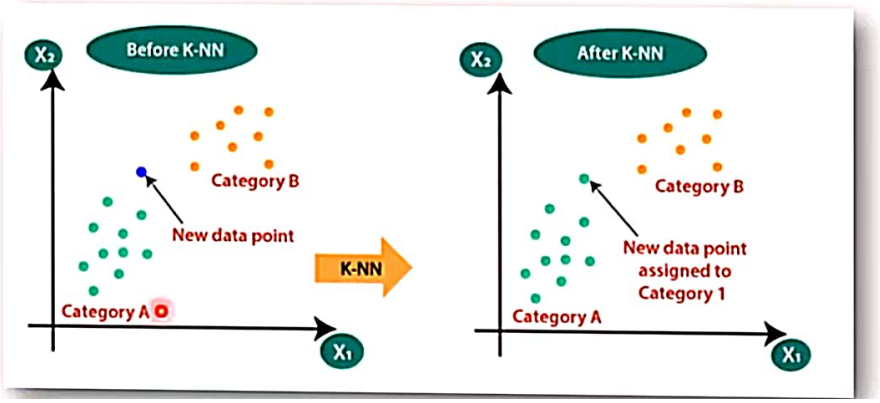






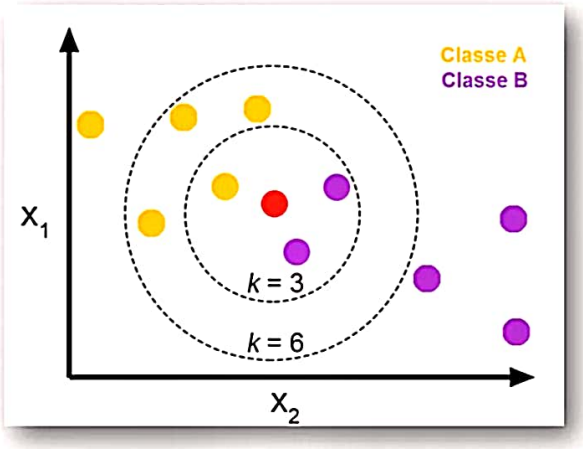
Why do we need a K-NN Algorithm?

- Suppose there are two categories, i.e., **Category A and Category B**, and we have a **new data point x1**, so this data point will lie in which of these categories.
- To solve this type of problem, <u>we need a K-NN algorithm</u>.
- With the help of K-NN, we can <u>easily identify the category or class of a particular dataset</u>.



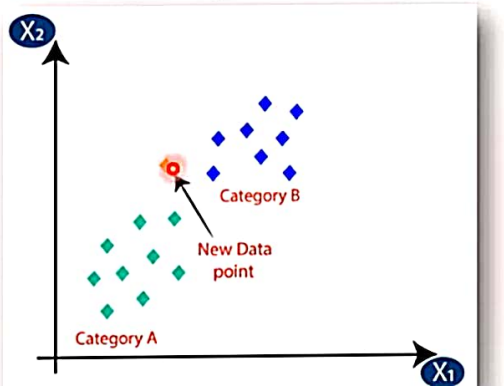
KNN Algorithm Steps

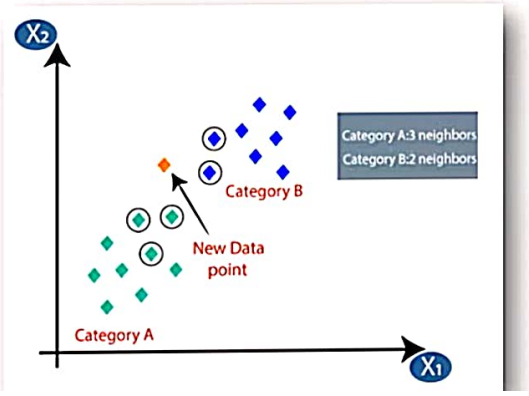
- **Step-1:** Select the number K of the neighbors
- **Step-2:** Calculate the Euclidean distance of **K number of neighbors**
- **Step-3:** Take the K nearest neighbors as per the calculated Euclidean distance.
- **Step-4:** Among these k neighbors, count the number of the data points in each category.
- **Step-5:** Assign the new data points to that category for which the number of the neighbor is maximum.
- **Step-6:** Our model is ready.



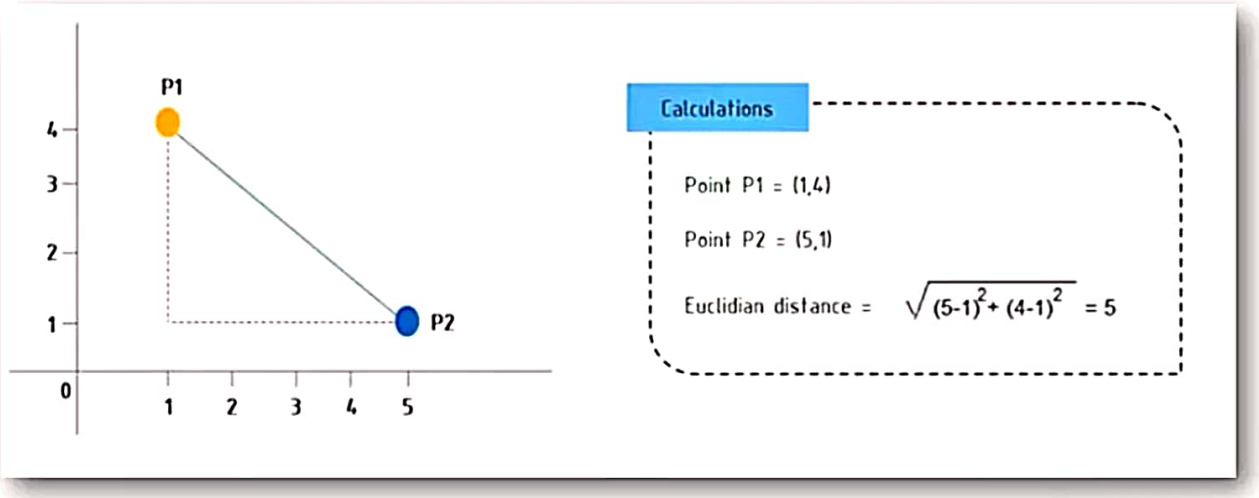
How to select the value of K in the K-NN Algorithm?

- There is no particular way to determine the best value for "K", so we need to try some values to find the best out of them.
- The most preferred value for K is 5.
- A very low value for K such as K=1 or K=2, can be noisy and lead to the effects of outliers in the model.
- Large values for K are good, but it may find some difficulties.





Revise: Euclidean Distance Formula



Example

Given Query: x=(Maths=6, CS=8) (X2, Y2)

Step 1: Select K neighbors. K=3

Sr. No.	Maths	CS	Result
1	4	3	Fail
2	6	7	Pass
3	7	8	Pass
4	5	5	Fail
5	8	8	Pass

Euclidean Distance Formula:

$$d = \sqrt{[(x_2 - x_1)^2 + (y_2 - y_1)^2]}$$

- X_2, y_2 = Observe Valúes
- X1. y1 = Actual Valúes
- d = distance between (x11, y11) and (x22, y22)

Step 2: Calculate the Euclidean distance of **K number of neighbors.**

1. $\sqrt{[(6-4)^2 + (8-3)^2]} = \sqrt{29} = 5.38$
2. $\sqrt{[(6-6)^2 + (8-7)^2]} = 1$
3. $\sqrt{[(6-7)^2 + (8-8)^2]} = 1$
4. $\sqrt{[(6-5)^2 + (8-5)^2]} = \sqrt{10} = 3.16$
5. $\sqrt{[(6-8)^2 + (8-8)^2]} = 2$

Step 3: As per Result, **2. Pass 3. Pass 5. Pass**

We have, **3 Pass & 0 Fail** So, **3>0**

Probability of Pass is more.

Hence, In Given Query x=(Maths=6, CS=8) is Pass.