UNIT-4
SUPERVISED MACHINE LEARNING MODELS

Prepared by : Ritesh Prajapati

INTRODUCTION TO MACHINE LEARNING(4350702)

SEMESTER-5 DIPLOMA COMPUTER ENGINEERING

Topics

4.1.1 Introduction of Supervised Learning

4.1.2 Example of Supervised Learning

4.2.1 Types of Supervised Learning

- Classification (concept and example)

  K-Nearest Neighbour,

  Decision Trees,

  Support Vector Machines,

  Logistic Regression

- Regression (concept and example):

  Linear Regression

Prepared by : Ritesh Prajapati

Supervised Machine Learning

- Idea of learning from past information.
- Past information= ‘EXPERIENCE’

- Supervised learning is the types of machine learning in which machines are trained using **well "labelled" training data, and on basis of that data, machines predict the output.**

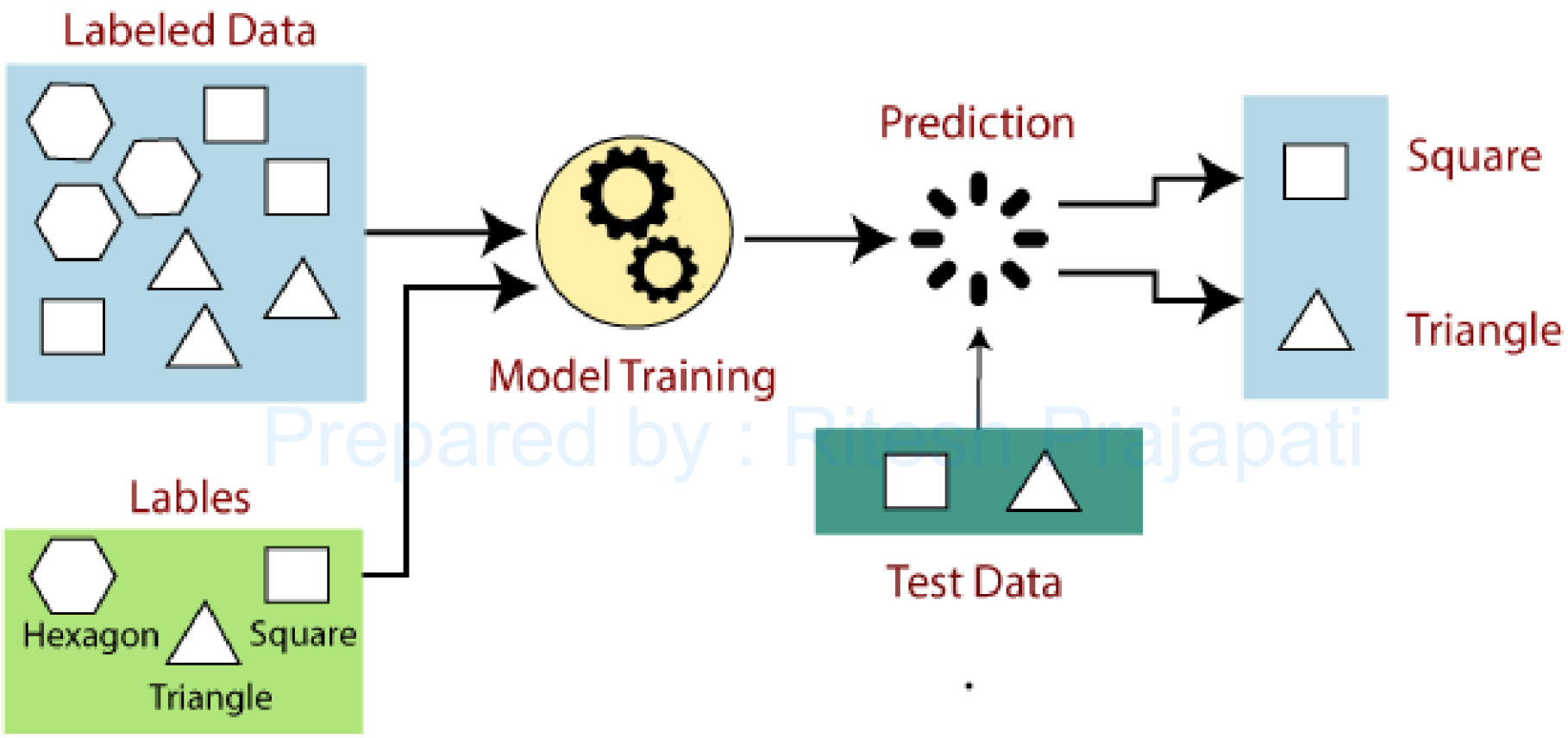
Examples of supervised learning tasks:

Prepared by : Ritesh Prajapati

- Predicting game results
- Predicting tumour malignancy
- Predicting the price of domains like real estate or stocks
- Fraud Detection
- Spam filtering
- Risk Assessment

Prepared by : Ritesh Prajapati

Supervised Machine Learning -Example



Prepared by : Ritesh Prajapati

Prepared by : Ritesh Prajapati

Supervised Machine Learning -Example

- Suppose we have a dataset of different types of shapes which includes square, rectangle, triangle, and Polygon. Now the first step is that we need to train the model for each shape.
- If the given shape has four sides, and all the sides are equal, then it will be labelled as a **Square**.
- If the given shape has three sides, then it will be labelled as a **triangle**.
- If the given shape has six equal sides then it will be labelled as **hexagon**.
- Now, after training, we test our model using the test set, and the task of the model is to identify the shape.
- The machine is already trained on all types of shapes, and when it finds a new shape, it classifies the shape on the bases of a number of sides, and predicts the output.

Prepared by : Ritesh Prajapati

Prepared by : Ritesh Prajapati

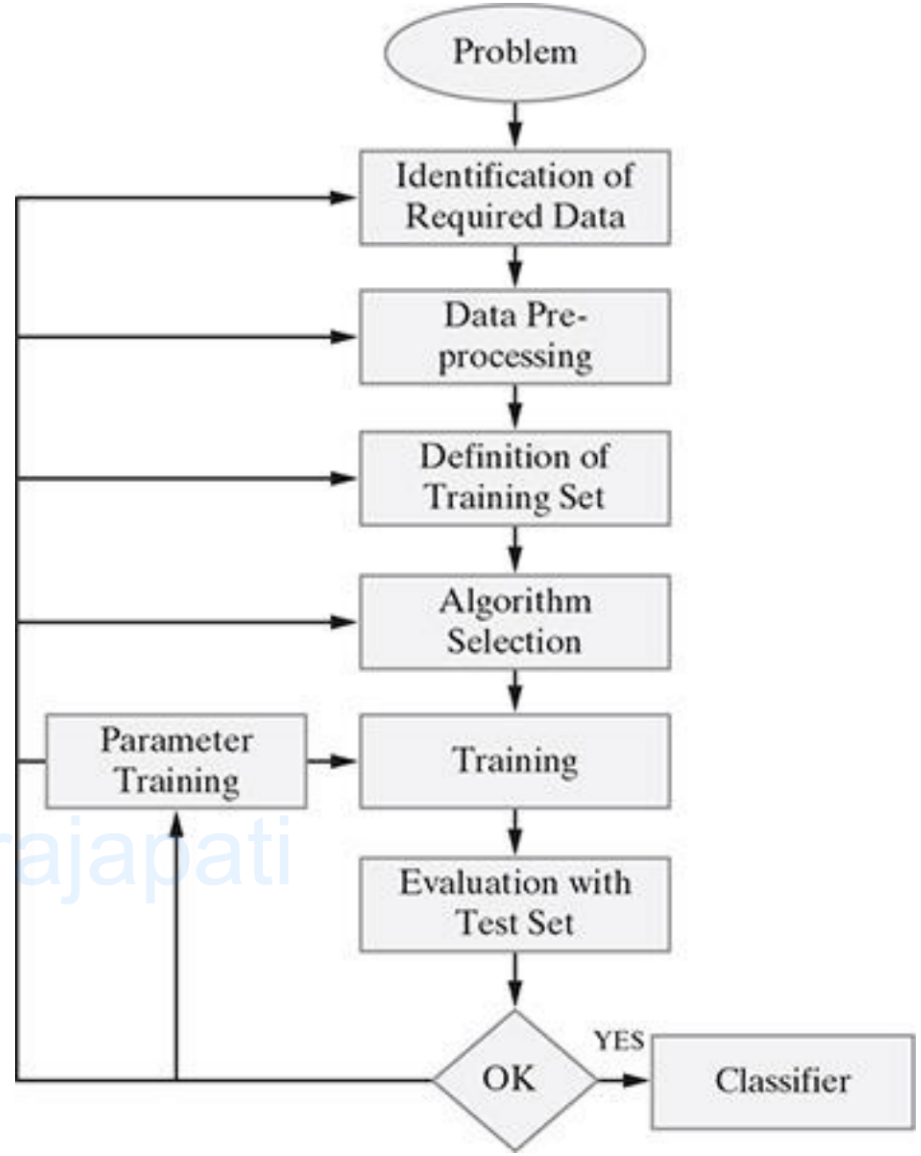
Classification

- Used when the output variable is categorical, which means there are two classes such as Yes-No, Male-Female, True-false, etc.
- The algorithm learns to classify input data into predefined classes.
- Examples
  - Email spam detection
  - Medical diagnosis
  - Handwriting detection
- Algorithms
  - k-Nearest Neighbours
  - Decision Tree
  - Support vector Machines
  - Logistic Regression

Prepared by : Ritesh Prajapati

Classification -Steps

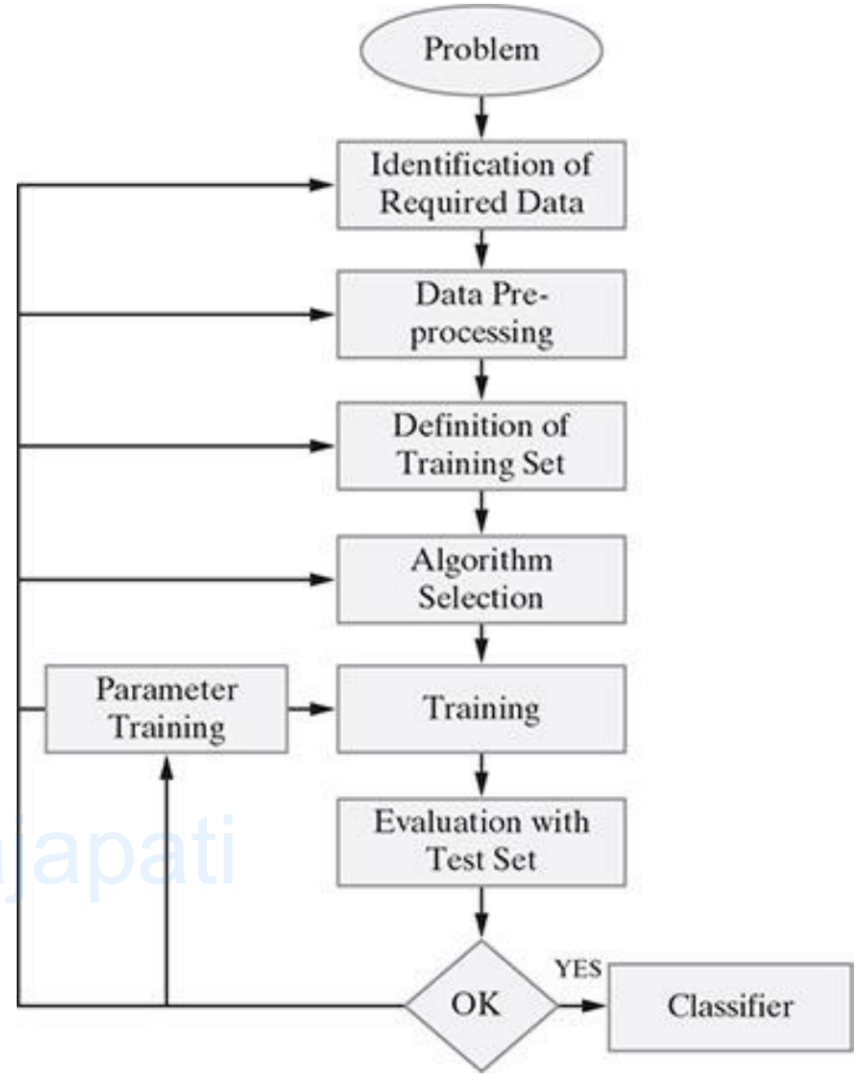
- **Problem Identification:** Identifying the problem is the first step in the supervised learning model.
- **Identification of Required Data:** On the basis of the problem identified, the required data set that represents the identified problem needs to be identified/evaluated.
- **Data Pre-processing:** cleaning/transforming the data set. This ensures that all the unnecessary/irrelevant data elements are removed.
- **Definition of Training Data Set:** decide what kind of data set is to be used as a training set.



Prepared by : Ritesh Prajapati

Classification -Steps

- **Algorithm Selection:** On the basis of various parameters, the best algorithm for a given problem is chosen.
- **Training:** The algorithm runs on the gathered training set for further fine tuning. The parameters (inputs given to algorithm) may also be adjusted by optimizing performance on a subset (called as validation set) of the training set.
- **Evaluation with the Test Data Set:** Training data is run on the algorithm, and its performance is measured here. If a suitable result is not obtained, further training of parameters may be required.



Prepared by : Ritesh Prajapati

k-Nearest Neighbours

- Concept of Nearest Neighbours
  - People having similar background or mindset tend to stay close to each other.
  - neighbours in a locality have a similar background
- It assumes the similarity between the new case/data and available cases and put the new case/data into the category that is most similar to the available categories.
- **KNN algorithm stores all the available data and classifies a new data point based on the similarity.**
- It is a **non-parametric algorithm**
  - It does not make any assumption on underlying data.
- It is called a **lazy learner algorithm**
  - 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.

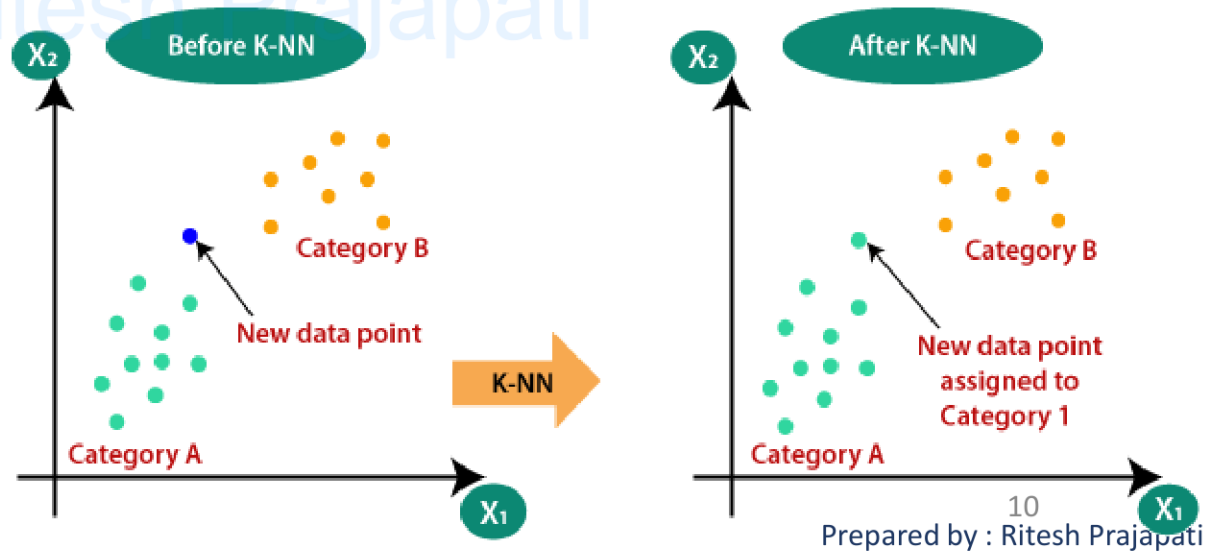
Prepared by : Ritesh Prajapati

k-Nearest Neighbours Example1

- **Predicting Cat or Dog**
- 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.
- 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.

Prepared by : Ritesh Prajapati

- Suppose there are two categories, A and B,
- we have a new data point x1, so this data point will lie in which of these categories.
- Using KNN, we can identify the category or class of a particular dataset



k-Nearest Neighbours Steps

- **Step-1:** Select the number K of the neighbors
- **Step-2:** Calculate the Euclidean distance of **K number of neighbors**

  Euclidean distance for points p(p1,p2) and q (q1,q2) is given by

$$d(p,q) = \sqrt{(q_1 - p_1)^2 + (q_2 - p_2)^2}.$$

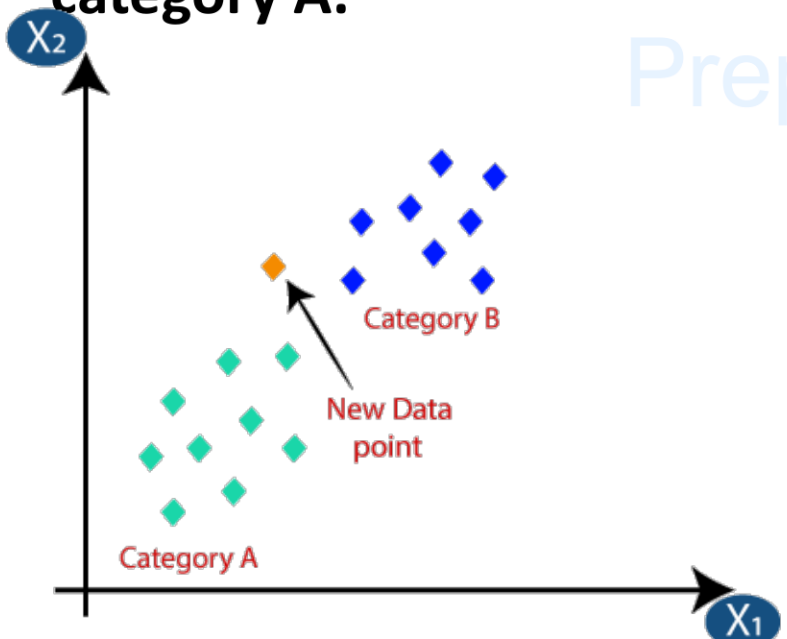
- **Step-3:** Take the K nearest neighbors as per the calculated Euclidean distance.
- **Step-4:** Among these k neighbors, count the number of the data points in each category.
- **Step-5:** Assign the new data points to that category for which the number of the neighbor is maximum.
- **Step-6:** Our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

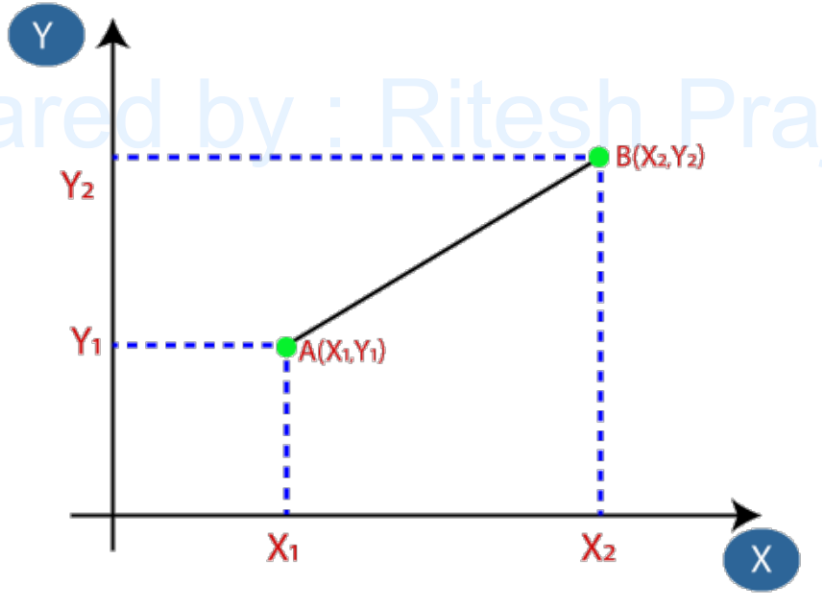
Prepared by : Ritesh Prajapati

k-Nearest Neighbours Example1

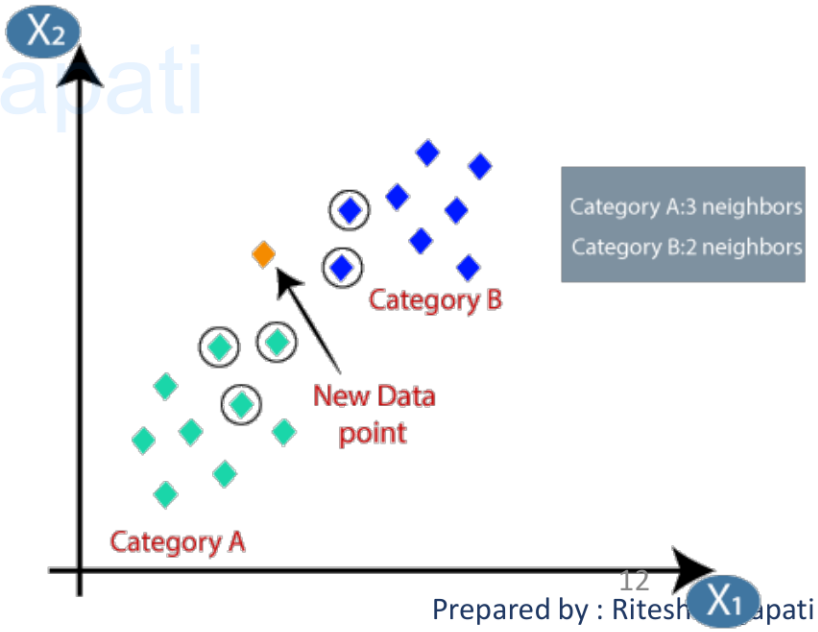
- We have a new data point and we need to put it in the required category
- Choose the number of neighbors, so we will choose the **k=5.**
- Calculate the **Euclidean distance** between the data points.
- By calculating the Euclidean distance ,Get the nearest neighbors, as **3 nearest neighbors in category A and 2 nearest neighbors in category B**
- As the 3 nearest neighbors are from category A, hence this new data point must **belong to category A.**

Prepared by : Ritesh Prajapati





Euclidean Distance between A_1 and B_2 = $\sqrt{(X_2-X_1)^2+(Y_2-Y_1)^2}$



k-Nearest Neighbours Example2

- Student data set of 15 students studying in a class.
- Each students has a score on a scale of 10 on two performance parameters – 'Aptitude' and 'Communication'.
- Also, a class value is assigned to each student :
- Good communication skills as well as a good level of aptitude as 'Leader'
- Good communication skills but not so good level of aptitude as 'Speaker'
- Not good communication skill but a good level of aptitude as 'Intel'

Name	Aptitude	Communication	Class
Karuna	2	5	Speaker
Bhuvna	2	6	Speaker
Gaurav	7	6	Leader
Parul	7	2.5	Intel
Dinesh	8	6	Leader
Jani	4	7	Speaker
Bobby	5	3	Intel
Parimal	3	5.5	Speaker
Govind	8	3	Intel
Susant	6	5.5	Leader
Gouri	6	4	Intel
Bharat	6	7	Leader
Ravi	6	2	Intel
Pradeep	9	7	Leader
Josh	5	4.5	Intel

Prepared by : Ritesh Prajapati

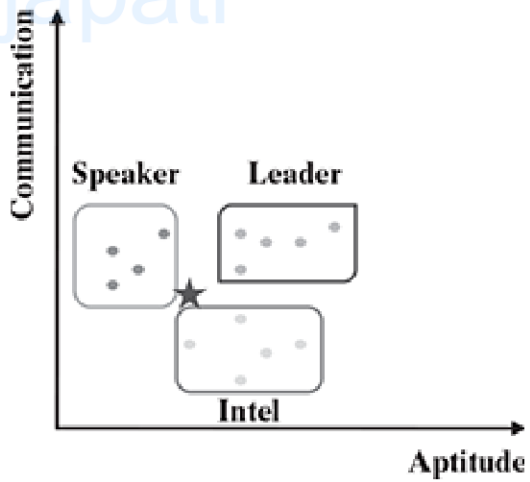
k-Nearest Neighbours Example2

- Dividing data into training and testing
- The record of the student named Josh is assumed to be the test data.
- the training data points of the Student data set considering only the features 'Aptitude' and 'Communication' can be represented as dots in 2D.
- Neighbouring data elements needs to found.
- Euclidean distance of the different dots need to be calculated from the asterisk.

$$d(p,q) = \sqrt{(q_1 - p_1)^2 + (q_2 - p_2)^2}.$$

	Name	Aptitude	Communication	Class
Training Data	Karuna	2	5	Speaker
	Bhuvna	2	6	Speaker
	Gaurav	7	6	Leader
	Parul	7	2.5	Intel
	Dinesh	8	6	Leader
	Jani	4	7	Speaker
	Bobby	5	3	Intel
	Parimal	3	5.5	Speaker
	Govind	8	3	Intel
	Susant	6	5.5	Leader
	Gouri	6	4	Intel
	Bharat	6	7	Leader
	Ravi	6	2	Intel
	Pradeep	9	7	Leader
Test Data	Josh	5	4.5	Intel

Prepared by : Ritesh Prajapati



k-Nearest Neighbours Example2

- In the kNN algorithm, the value of ‘k’ is the number of neighbours that need to be considered.
- If k = 3, 3 nearest neighbours closest to the test data element are considered and the class which is predominant is assigned to test element.
- Taking k=1, data of *Gouri* comes closest, and it has class Intel thus data *Josh* will assigned **Intel** class.
- Taking k=3,data of *Gouri,Bobby* and *Sushant* comes closest. Majority comes of Intel class thus data *Josh* will assigned **Intel** class.
- Taking k=2,data of *Gouri,* and *Sushant* comes closest. Both class having equal instances(1 intel and 1 leader),So its difficult to decide.

Name	Aptitude	Communication	Class	Distance	$k = 1$	$k = 2$	$k = 3$
Karuna	2	5	Speaker	3.041			
Bhuvna	2	6	Speaker	3.354			
Parimal	3	5.5	Speaker	2.236			
Jani	4	7	Speaker	2.693			
Bobby	5	3	Intel	1.500			1.500
Ravi	6	2	Intel	2.693			
Gouri	6	4	Intel	1.118	1.118	1.118	1.118
Parul	7	2.5	Intel	2.828			
Govind	8	3	Intel	3.354			
Susant	6	5.5	Leader	1.414			
Bharat	6	7	Leader	2.693			
Gaurav	7	6	Leader	2.500			
Dinesh	8	6	Leader	3.354			
Pradeep	9	7	Leader	4.717			
Josh	5	4.5	???				

Prepared by : Ritesh Prajapati

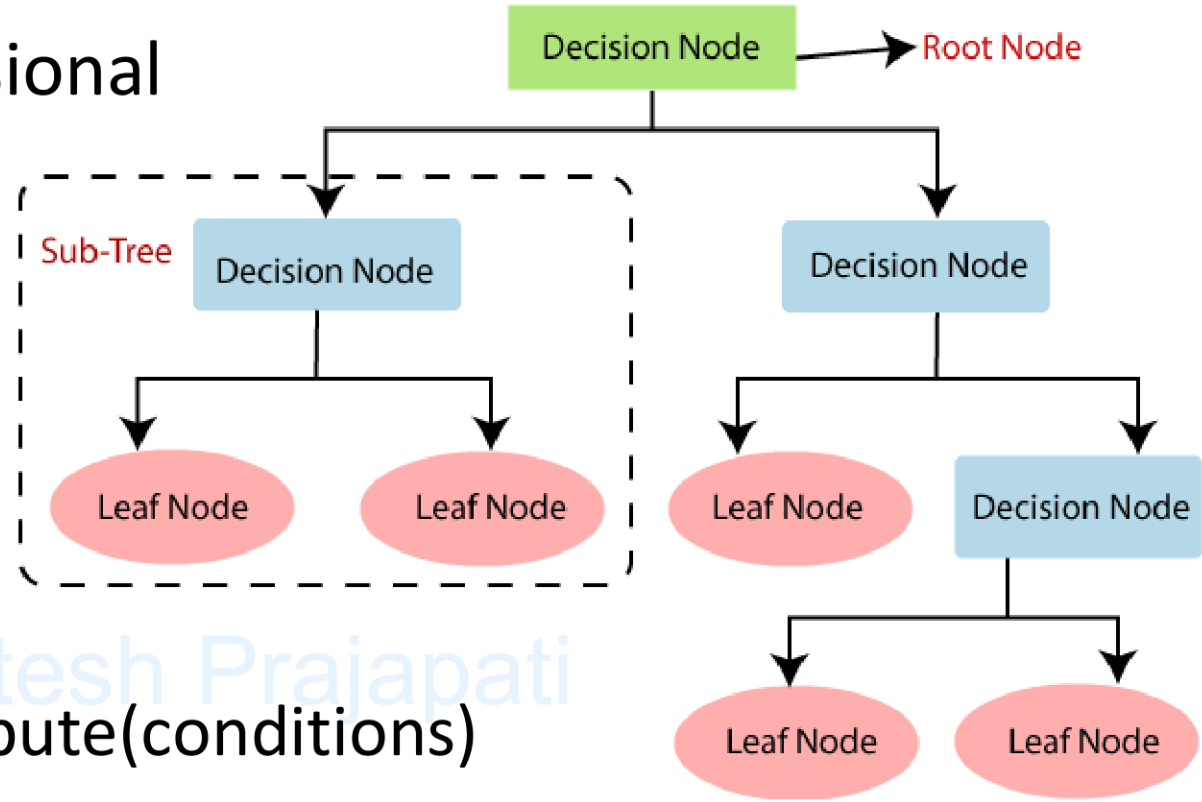
k-Nearest Neighbours

- **How to select the value of K in the K-NN Algorithm**
- There is no particular way to determine the best value for "K"
- Need to try some values to find the best out of them.
- The most preferred value for K is 5.
- Low value for K such as K=1 or K=2, can be noisy
- **Solution**
- Set k equal to the square root of the number of training records.
- Test several k values on a variety of test data sets and choose the one that delivers the best performance.

Prepared by : Ritesh Prajapati

Decision Trees

- A decision tree is used for multi-dimensional analysis with multiple classes.
- It builds a model in the form of a tree structure (starts with the root node, and expands on further branches)
- A decision tree simply asks a question, and based on the answer (Yes/No), it further split the tree into subtrees.
- This question is based on the feature attribute(conditions)
- Decision tree consists of three types of nodes
  - Root Node : It is the first node from where the decision tree starts (Green color).
  - Branch Node : Branches from the root node for which more nodes are created (Blue color).
  - Leaf Node : Branch nodes which are the final output node from which no other nodes are created.(Pink color)



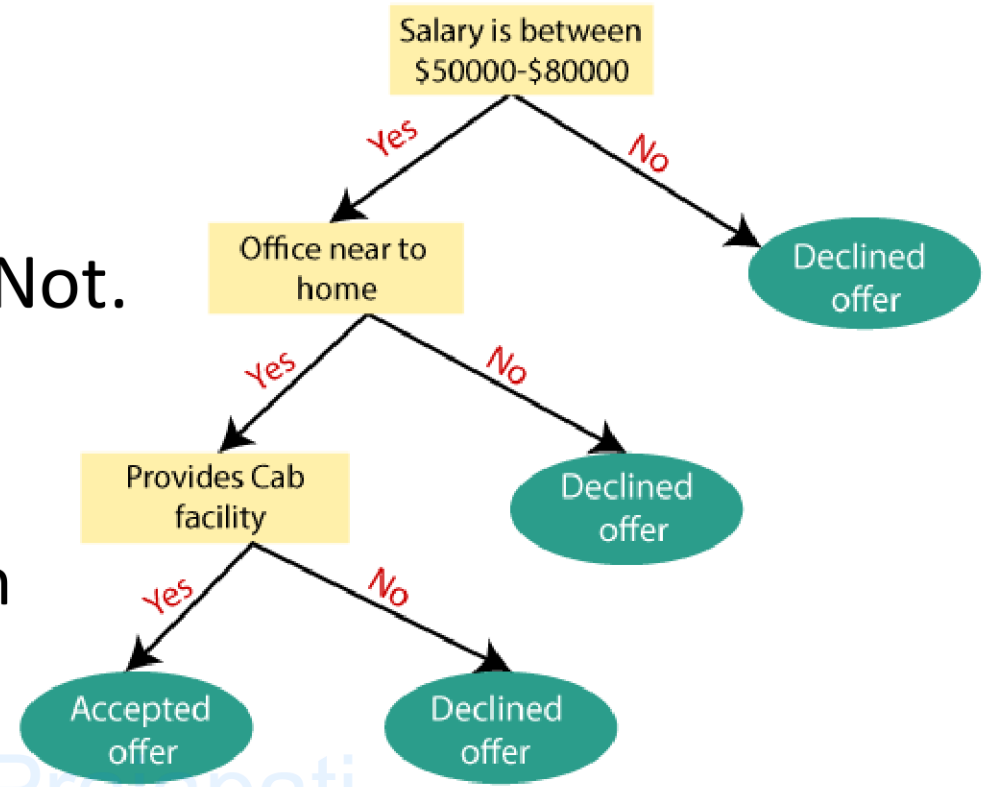
Prepared by : Ritesh Prajapati

Decision Trees Steps

- **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.
- **Step-6:** Continue this process until a stage is reached where you cannot further classify the nodes and called the final node as a leaf node.

Prepared by : Ritesh Prajapati

Decision Trees Example1



- A candidate who has a job offer and wants to decide whether he should accept the offer or Not.
- The decision tree starts with the root node (Salary attribute by ASM).
- The root node splits further into the next decision node (distance from the office) and one leaf node (Declined offer)
- The next decision node further gets split into one decision node (Cab facility) and one leaf node (Declined offer)
- Finally, the decision node splits into two leaf nodes (Accepted offers and Declined offer).
- *A decision tree can contain categorical data (YES/NO) as well as numeric data*.

Prepared by : Ritesh Prajapati

Decision Trees Example2

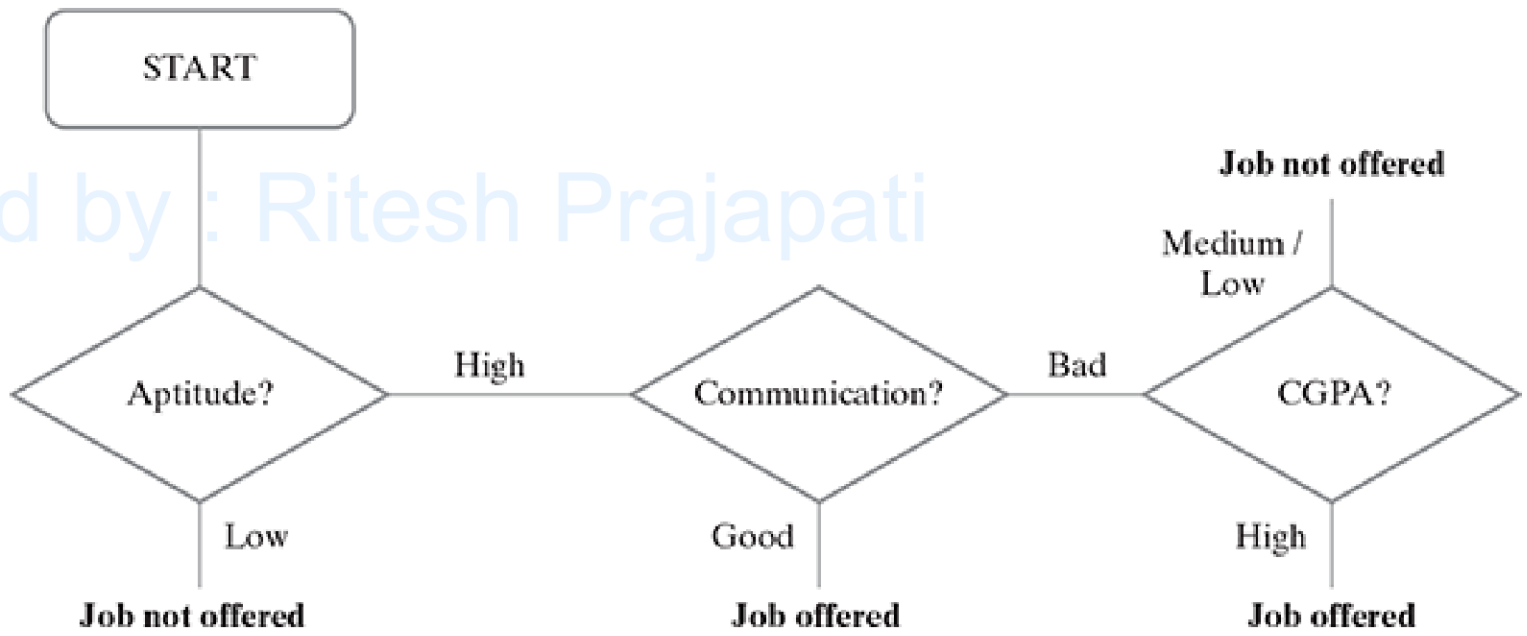
- Data for campus recruitment shortlisted 18 students for final interview.
- On the basis of different evaluation parameters –
  - CGPA,
  - Communication skill,
  - Aptitude,
  - Programming skills
- Predicting whether student s will get a job offer, by using the decision tree model.

CGPA	Communication	Aptitude	Programming Skill	Job offered?
High	Good	High	Good	Yes
Medium	Good	High	Good	Yes
Low	Bad	Low	Good	No
Low	Good	Low	Bad	No
High	Good	High	Bad	Yes
High	Good	High	Good	Yes
Medium	Bad	Low	Bad	No
Medium	Bad	Low	Good	No
High	Bad	High	Good	Yes
Medium	Good	High	Good	Yes
Low	Bad	High	Bad	No
Low	Bad	High	Bad	No
Medium	Good	High	Bad	Yes
Low	Good	Low	Good	No
High	Bad	Low	Bad	No
Medium	Bad	High	Good	No
High	Bad	Low	Bad	No
Medium	Good	High	Bad	Yes

Prepared by : Ritesh Prajapati

Decision Trees Example2

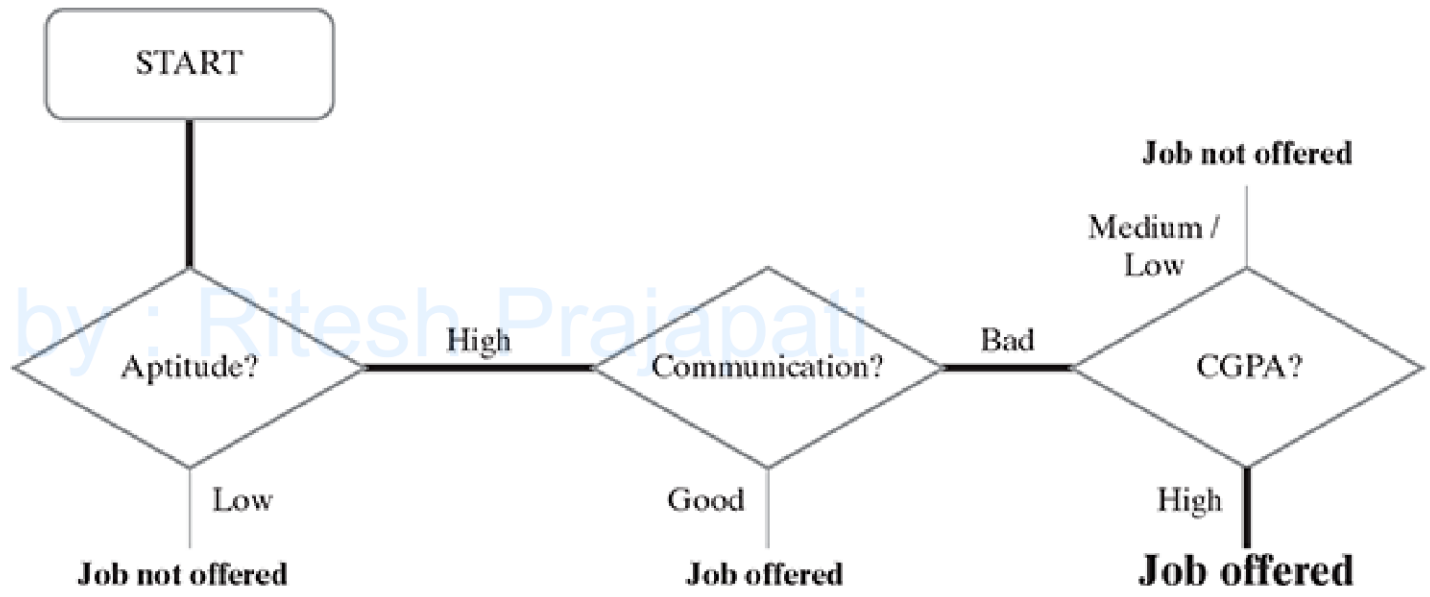
- First, we draw the decision tree corresponding to the training data given
- Job offer condition is FALSE for all the cases where **Aptitude = Low.**
- So feature **Aptitude** can be taken up as **the first node.**
- For Aptitude = High, job offer is TRUE for cases where Communication = Good.
- For cases where Communication = Bad, job offer is TRUE for cases where CGPA = High.



Prepared by : Ritesh Prajapati

Decision Trees Example2

- Prediction whether a student s might get a job offer for the given parameter values:
- *CGPA = High, Communication = Bad, Aptitude = High, Programming skills = Bad.*
- Figure depicts a sample path (thick line) for the conditions
- Student with such parameter get a job offer



Prepared by : Ritesh Prajapati

Decision Trees

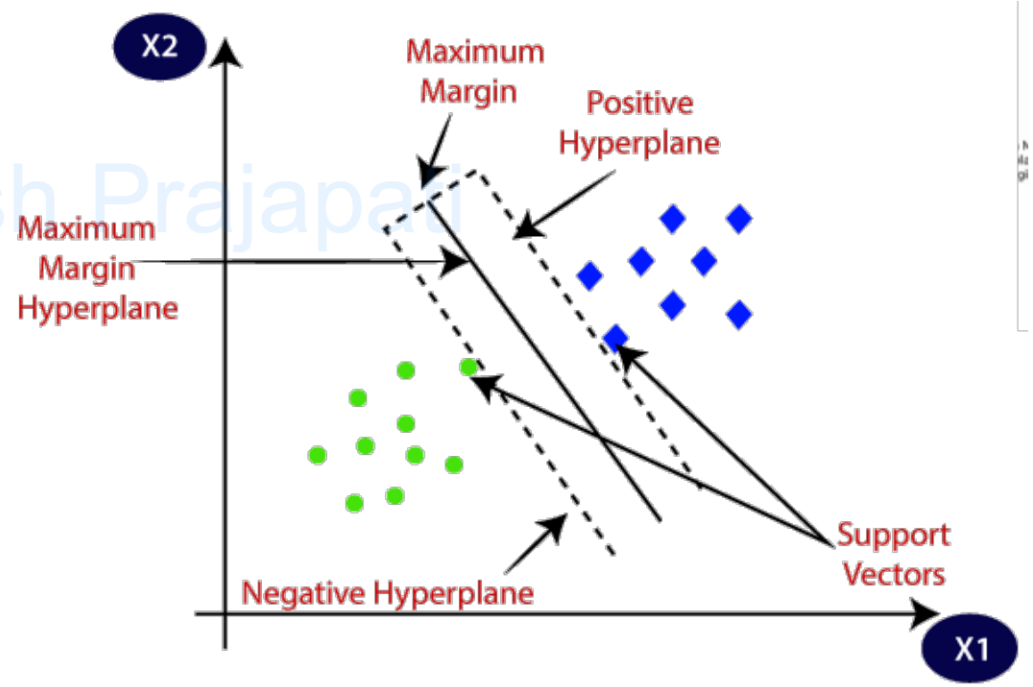
- **Attribute Selection Measures**
- Main issue how to select the best attribute for root node and for sub-nodes.
- We use a technique called as **Attribute selection measure or ASM**
- The data should be split in such a way that the partitions created by the split should contain examples belonging to a single class.
  - If that happens, the partitions are considered to be pure.
- Entropy is a **measure of impurity of an attribute**. It specifies randomness in data.
- Information gain for a particular feature A is calculated by **the difference in entropy before a split (or S_{bs}) with the entropy after the split (S_{as}).**

  Information Gain (S, A) = Entropy (S_{bs}) – Entropy (S_{as})
- Need to find an **attribute that returns the highest information gain.**

Prepared by : Ritesh Prajapati

Support vector machines

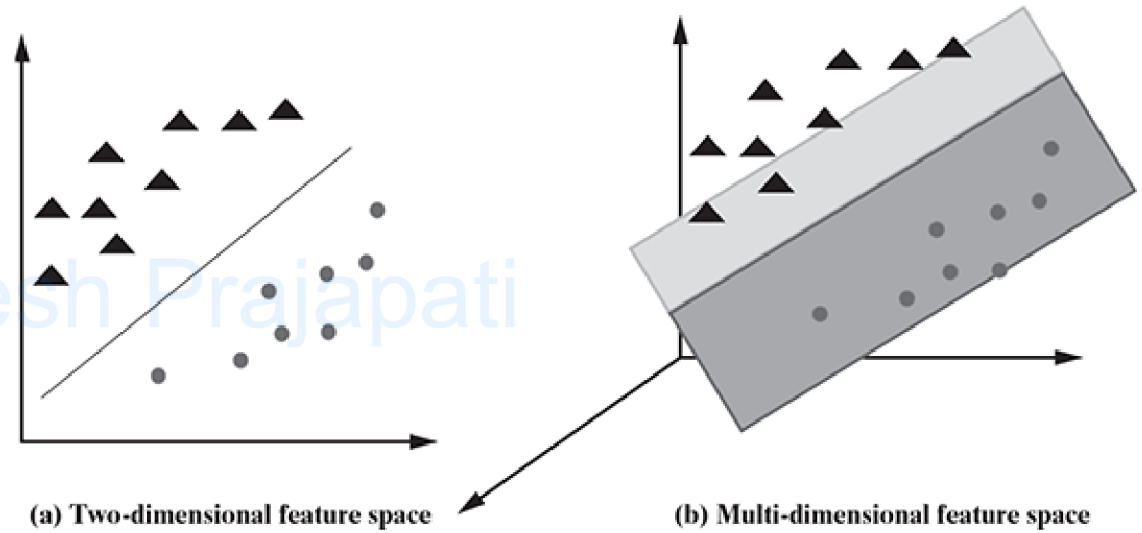
- SVM is based on the concept of a surface, called a **hyperplane**, which draws a **boundary between data instances** plotted in the multi-dimensional feature space.
- The SVM algorithm builds an N-dimensional hyperplane model that **assigns future instances into one of the two possible output classes.**
- SVM chooses the extreme points/vectors that help in creating the hyperplane. These extreme cases are called as **support vectors,** and hence algorithm is termed as Support Vector Machine.



Prepared by : Ritesh Prajapati

Support vector machines

- **Support Vectors:** Support vectors are the **data points(representing classes),**which are near the identified set of lines (hyperplane).
- **Hyperplane :** For an N-dimensional feature space, hyperplane is a **flat subspace of dimension** (N−1) that separates and **classifies a set of data**.
  - a 2-D feature space (data set having two features and a class variable), a hyperplane will be a one-dimensional subspace or a straight line.
  - a 3-D feature space (data set having three features and a class variable), hyperplane is a two-dimensional subspace or a simple plane.

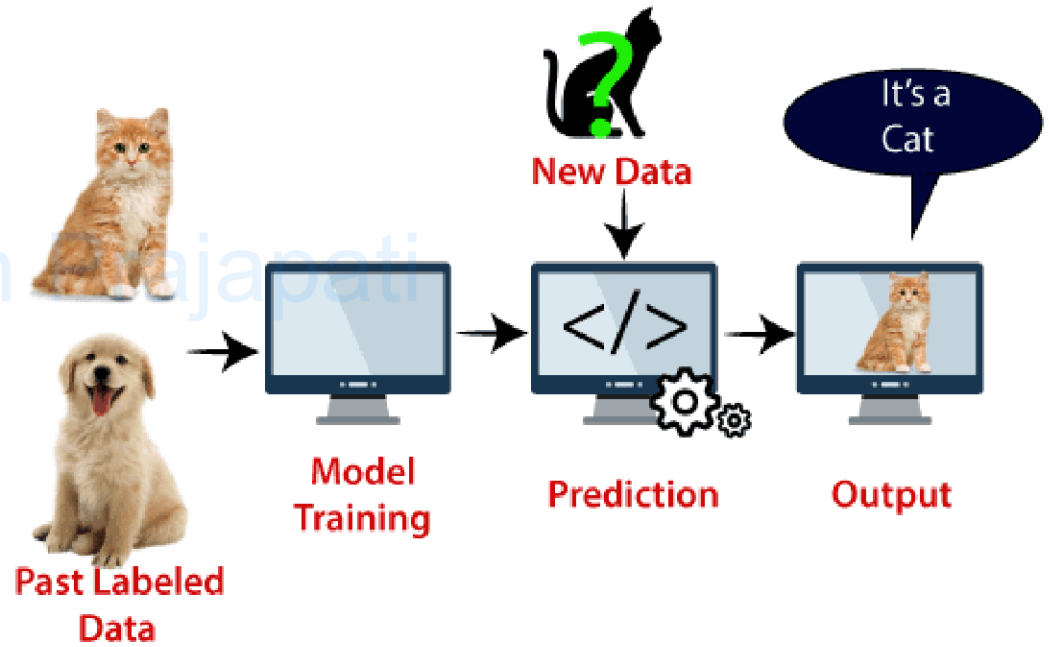


Prepared by : Ritesh Prajapati

- **Margin:** The distance between hyperplane and data points is known as margin.
  - focus is to identify a hyperplane which maximizes the margin.

Support vector machines Example

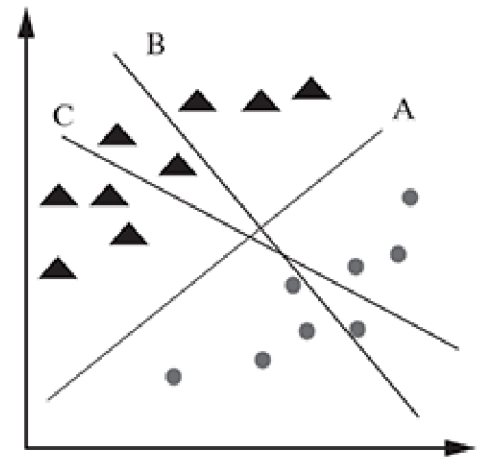
- Suppose we see a cat that also has some features of dogs
- If we want a model that can accurately identify whether it is a cat or dog, then such a model can be created by using the SVM algorithm.
- We will first train our model with lots of images of cats and dogs so that it can learn about different features of cats and dogs,
- And then we test it with new data creature.
- So as support vector creates a decision boundary between these two data (cat and dog) and choose extreme cases (support vectors), it will see the extreme case of cat and dog.
- On the basis of the support vectors, it will classify it as a cat.



Prepared by : Ritesh Prajapati

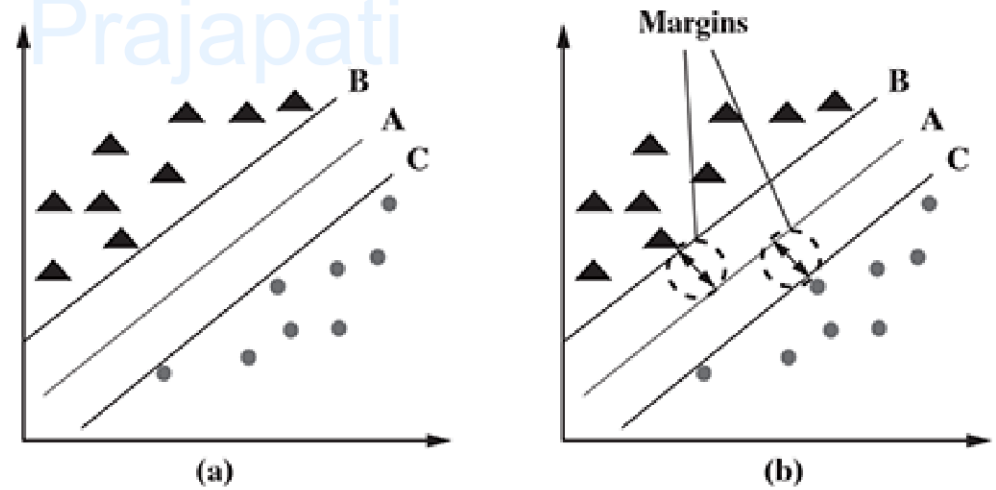
Support vector machines: Identifying correct hyperplane

- **Scenario1**
- we have three hyperplanes: A, B, and C.
- we need to identify the correct hyperplane which better segregates the two classes represented by the triangles and circles.
- hyperplane 'A' seems to do it well.



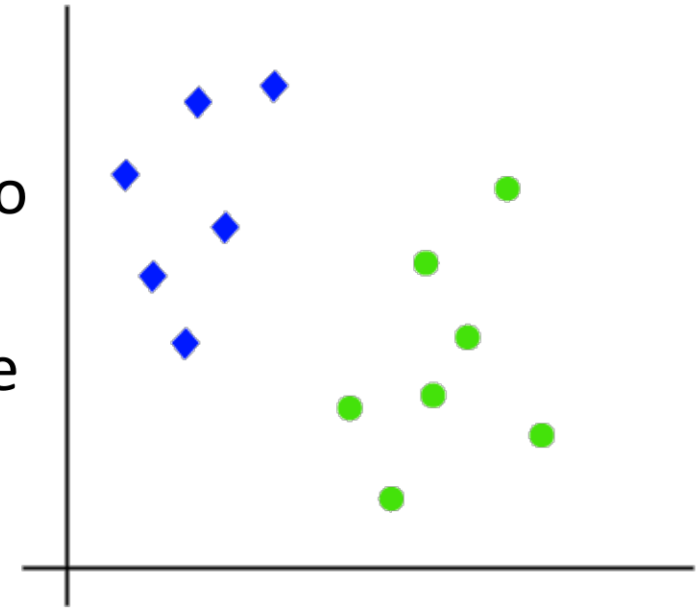
- **Scenario2**
- margin for hyperplane A is high as compared to those for both B and C
- If we select a hyperplane having a lower margin (distance), then there is a high probability of misclassification.

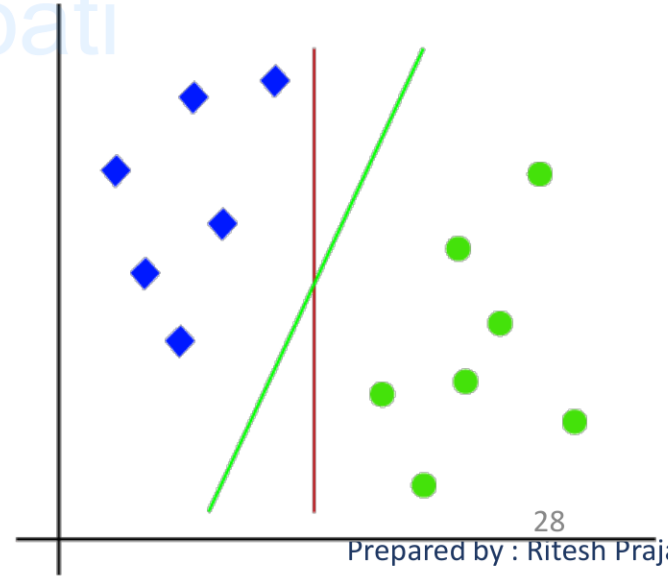
Prepared by : Ritesh Prajapati



Support vector machines: Types of SVM

- **Linear SVM**
- Used for linearly separated data (dataset can be classified into two classes by straight line).
- we have a dataset that has two tags (green and blue), and the dataset has two features x1 and x2.
- We want to classify the pair(x1, x2) of coordinates in either green or blue.
- As it is 2-d space, we can easily separate these two classes by just using a straight line,
- But there can be multiple lines that can separate these classes.
- We select the GREEN LINE as hyperplane with green line has maximum margin (Also called the optimal hyperplane).





Prepared by : Ritesh Prajapati

Support vector machines: Types of SVM

- **Non Linear SVM**
- Used for non linearly separated data (dataset can not be classified by straight line).
- Consider the image in Figure1
- To separate these data points, we need to add one more dimension, we will add a third dimension z as z=x2 +y2
- By adding the third dimension, the sample space will become as Figure2

Prepared by : Ritesh Prajapati

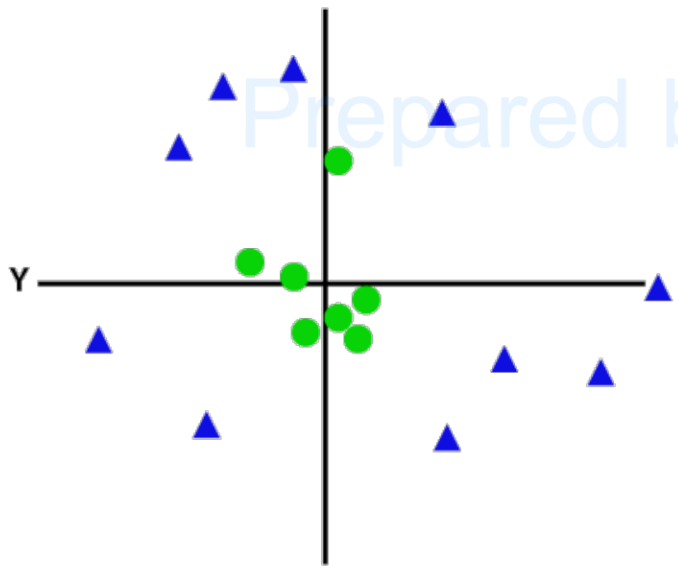


Figure1

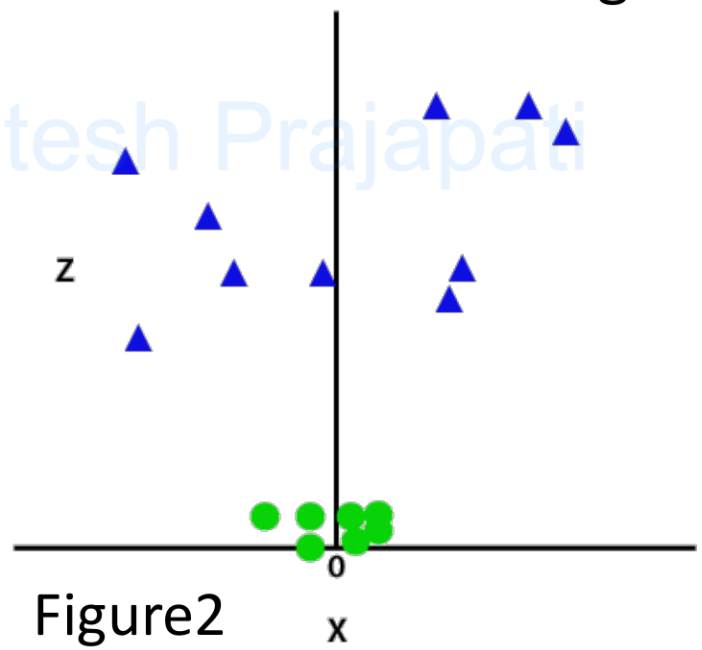


Figure2

Support vector machines: Types of SVM

- **Non Linear SVM**
- So now, SVM will divide the datasets into classes as Figure3.
- As in 3-d Space, it is looking like a plane parallel to the x-axis, we convert it in 2d space with z=1, then it will become as Figure4.
- we get a circumference of radius 1 in case of non-linear data.

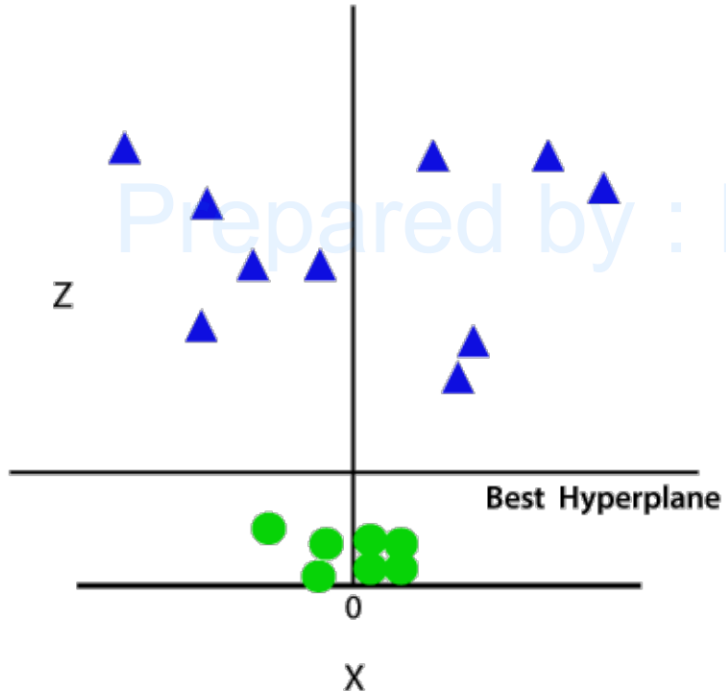


Figure3

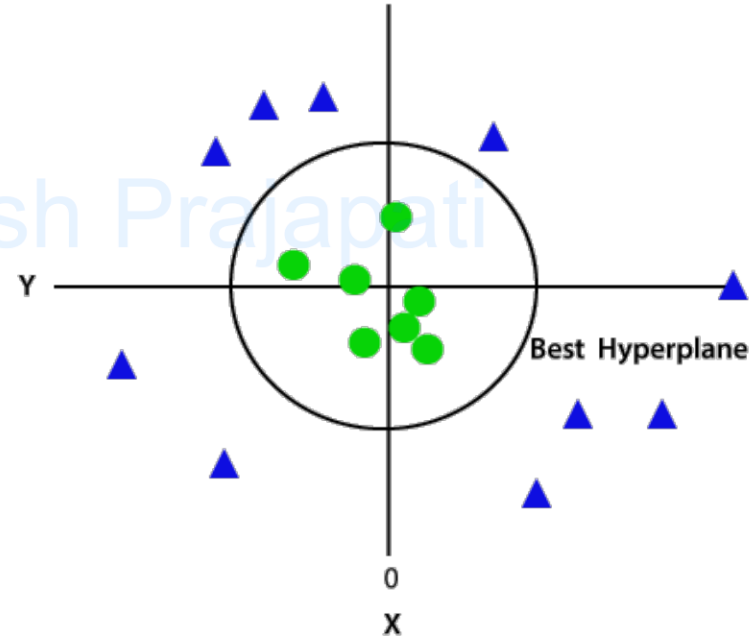


Figure4

Prepared by : Ritesh Prajapati

Regression

- Used for the prediction of continuous variables
- In regression tasks, the output is a continuous numerical value. The algorithm learns to map input features to a continuous target variable.
- Examples
  - Weather forecasting
  - Housing prices, stock prices prediction
- Algorithms
  - Linear Regression
  - Regression Trees
  - Non-Linear Regression
  - Bayesian Linear Regression
  - Polynomial Regression

Prepared by : Ritesh Prajapati

Prepared by : Ritesh Prajapati

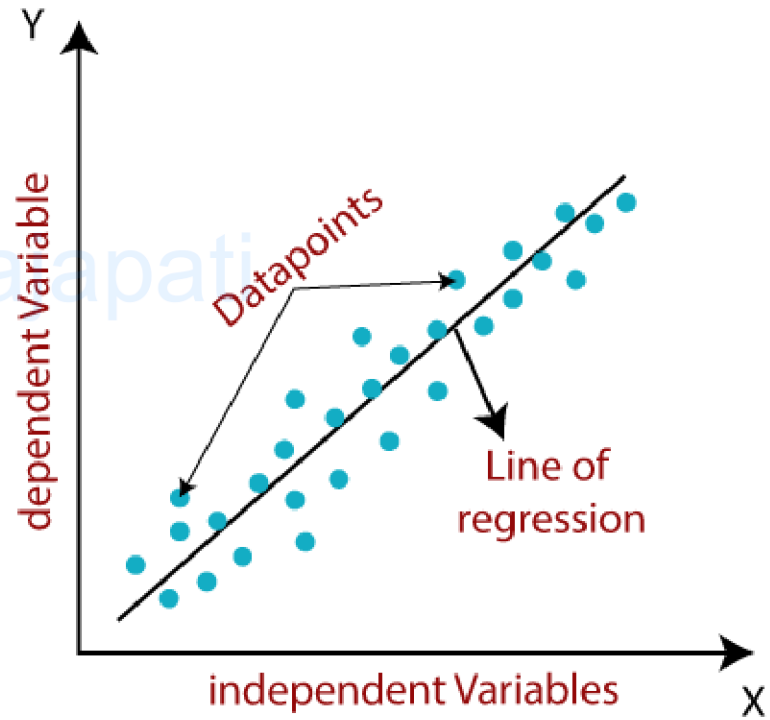
Regression

- Regression helps in finding the correlation between variables and enables us to predict the continuous output variable based on the one or more predictor variables.
- **Dependent Variable:** The main factor in Regression which we want to predict is called the dependent variable. It is also called **target variable**.
- **Independent Variable:** The factors which affect the dependent variables or which are used to predict the values of the dependent variables are called independent variable, also called as a **predictor**.
- **Outliers:** Outlier is an observation which contains either very low value or very high value in comparison to other observed values. It should be avoided.
- **Overfitting:** If algorithm works well with the training dataset but not well with test dataset, then such problem is called **Overfitting**.
- **Underfitting :** if algorithm does not perform well even with training dataset, then such problem is called **underfitting**.

Prepared by : Ritesh Prajapati

Linear Regression

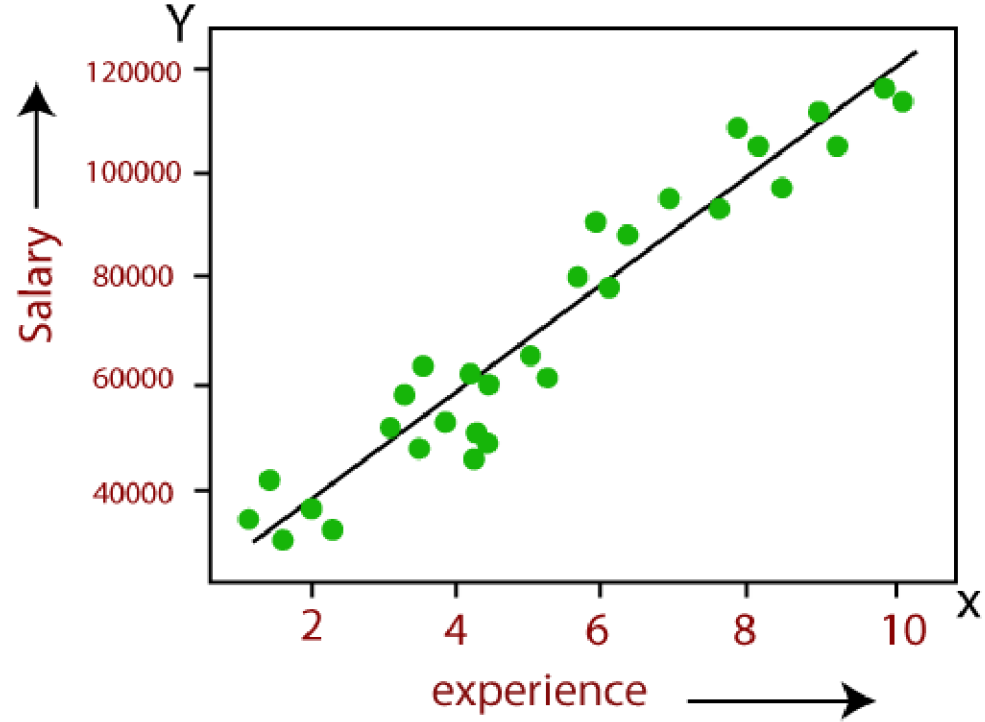
- Linear regression is a statistical method used for predictive analysis.
- It shows the relationship between the continuous variables.
- It shows the **linear relationship between the independent variable (X-axis) and the dependent variable (Y-axis)**, hence called linear regression.
- The linear regression model provides a sloped straight line representing the relationship between the variables.
- The goal is to estimate value of **dependent variable Y based** on value of **independent variable X**



Prepared by : Ritesh Prajapati

Linear Regression

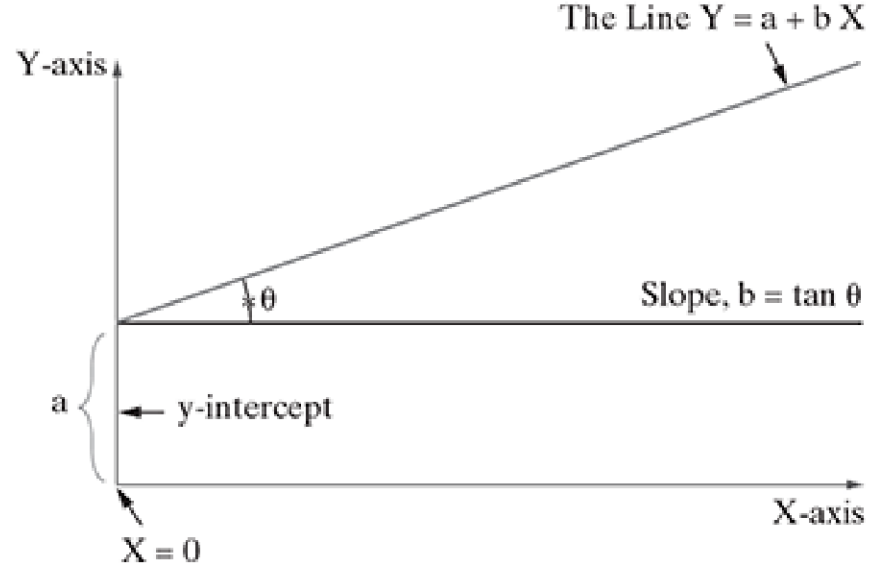
- **Example**
- predicting the salary of an employee on the basis of **the year of experience**.



- **Types of Linear Regression:**
- **Simple Linear Regression:**
  If a single independent variable is used to predict the value of a dependent variable, then it is called Simple Linear Regression.
- **Multiple Linear regression:**
  If more than one independent variable is used to predict the value of a dependent variable, then it is called Multiple Linear Regression.

Prepared by : Ritesh Prajapati

Simple Linear Regression



- Expression for simple linear regression is:
- **$Y = a + bX + \varepsilon$**
  - Y= Dependent Variable (Target Variable)
  - X= Independent Variable (predictor Variable)
  - a= intercept of the line (Gives value of Y when X=0)
  - b = slope(linear regression coefficient)(change in Y for one unit change in X).
  - ε = random error
- **Slope (b): $b = \Sigma[(X - \bar{X})(Y - \bar{Y})] / \Sigma(X - \bar{X})^2$**

  $\bar{X}$ = mean (average) of the independent variable X.

  $\bar{Y}$ = mean (average) of the dependent variable Y.

  Σ = summation, means calculate these values for each data point and sum them up.
- **Intercept (a): $a = \bar{Y} - b\bar{X}$**

  $\bar{X}$ = mean (average) of the independent variable X.

  $\bar{Y}$ = mean (average) of the dependent variable Y.

  b = slope

Prepared by : Ritesh Prajapati

Multiple Linear Regression

- Expression for multiple linear regression is:
- **$Y = a + b_1X_1 + b_2X_2 + \ldots + b_kX_k + \varepsilon$**
  - Y= Dependent Variable (Target Variable)
  - $X_1, X_2, \ldots, X_k$ = the k Independent Variables (predictor Variable)
  - a= intercept of the line (Gives value of Y when all X=0)
  - $b_1, b_2, \ldots, b_k$ = slope(regression coefficient)(change in each Y for one unit change in corresponding X).
  - ε = random error
- **Slope (b):**

$$b_1 = \Sigma[(X_1 - \bar{X}_1)(Y - \bar{Y})] / \Sigma(X_1 - \bar{X}_1)^2$$
$$b_2 = \Sigma[(X_2 - \bar{X}_2)(Y - \bar{Y})] / \Sigma(X_2 - \bar{X}_2)^2$$
$$\ldots$$
$$b_k = \Sigma[(X_k - \bar{X}_k)(Y - \bar{Y})] / \Sigma(X_k - \bar{X}_k)^2$$

$X_1, X_2, \ldots, X_k$ = the k Independent Variables
$\bar{X}_1, \bar{X}_2, \ldots, \bar{X}_k$ = mean (average) of k independent variables X
$\bar{Y}$ = mean (average) of the dependent variable Y.
Σ = summation

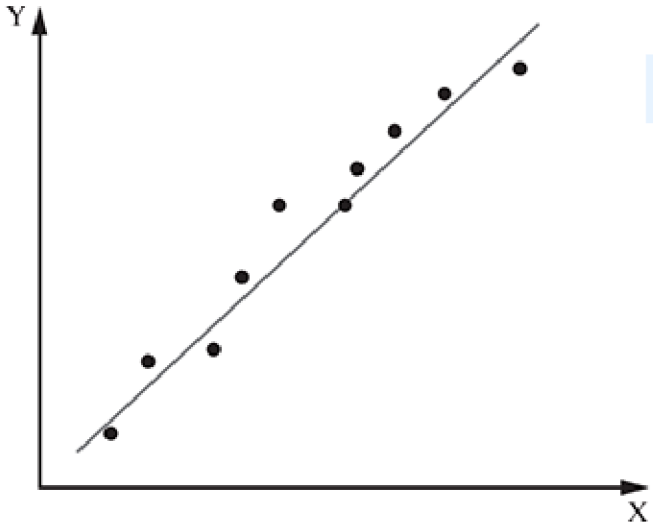
Prepared by : Ritesh Prajapati

- **Intercept (a): $a = \bar{Y} - b_1\bar{X}_1 - b_2\bar{X}_2 - \ldots - b_k\bar{X}_k$**

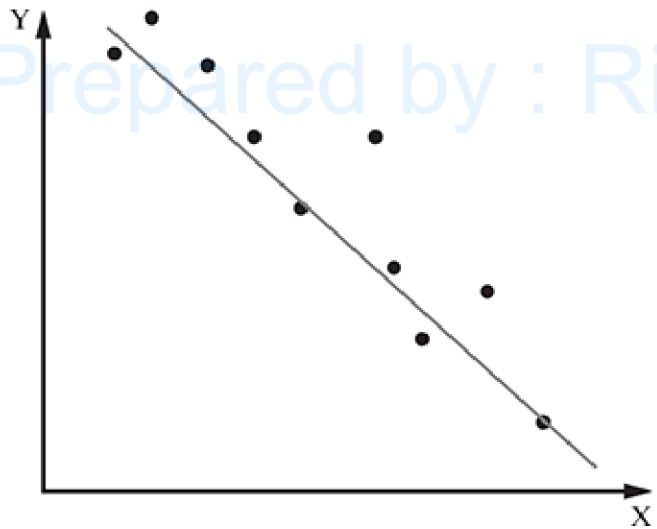
$\bar{X}_1, \bar{X}_2, \ldots, \bar{X}_k$ = mean (average) of k independent variables X
$\bar{Y}$ = mean (average) of the dependent variable Y.
$b_1, b_2, \ldots, b_k$ = slopes associated with each independent variable.

Linear Regression Line

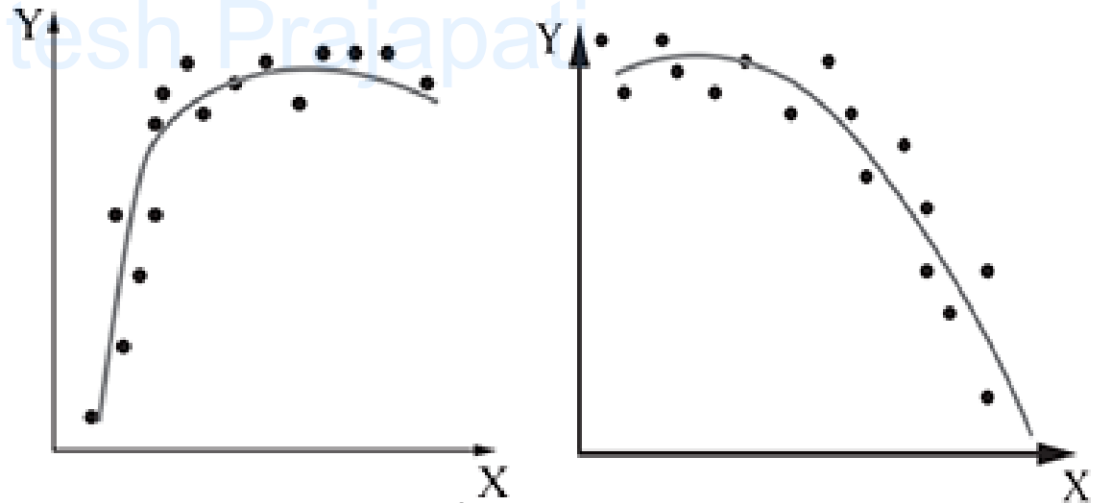
- A linear line showing the relationship between the dependent and independent variables is called a **regression line**.
- A **positive** slope always moves upward on a graph from left to right
- A **negative** slope always moves downward on a graph from left to right.
- Curve slope is non linear also known as Polynomial Regression Line.



Positive



Negative



Polynomial

Prepared by : Ritesh Prajapati

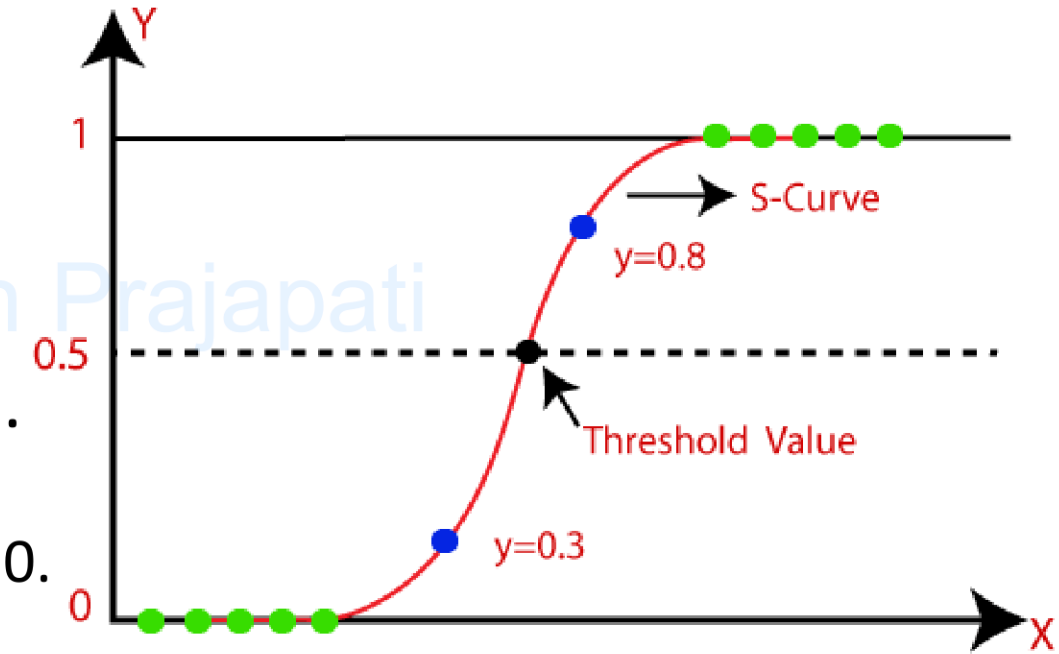
Logistic Regression

- It can be used for Classification as well as for Regression problems, but mainly used for **Classification problems.**
- Logistic regression is used to predict the **categorical dependent variable** with the help of independent variables.
- For classification problems, the outcome must be a categorical value. It can be either Yes or No, 0 or 1, true or False, etc.
- But in Logistic Regression**, instead** of giving the exact value as 0 and 1, **it gives the probabilistic values which lie between 0 and 1**.
- The dependent variable must be categorical in nature.
- Logistic regression is based on the concept of Maximum Likelihood estimation.
- In Logistic regression, instead of fitting a regression line, we fit an **"S" shaped logistic function(Sigmoid)**, which predicts two maximum values (0 or 1).

Prepared by : Ritesh Prajapati

Logistic Regression – Sigmoid Function

- The sigmoid function is a mathematical function used to map the predicted values to probabilities.
- We pass the weighted sum of inputs through a function that **can map values in between 0 and 1, which cannot go beyond this limit,** so it forms a curve like the **"S" form.**
- This function is known as **sigmoid function** and the curve obtained is called **S-curve**.
- We use the concept of the **threshold value**, which defines the probability of either 0 or 1.
  - values above the threshold value tends to 1, and a value below the threshold values tends to 0.



Prepared by : Ritesh Prajapati

Logistic Regression

- Expression for logistic regression is:
- **P(Y=1) = 1 / (1 + e^-(Z))**
- **$Z = a + b_1X_1 + b_2X_2 + \ldots + b_kX_k$**
  - P(Y=1) = probability that the dependent variable Y equals 1.
  - Y= Dependent Variable (Target Variable)
  - $X_1, X_2, \ldots, X_k$ = the k Independent Variables (predictor Variable)
  - a= intercept of the line (Gives value of Y when all X=0)
  - $b_1, b_2, \ldots, b_k$ = slope(regression coefficient)(change in each Y for one unit change in corresponding X).

Prepared by : Ritesh Prajapati

- **Type of Logistic Regression:**
- **Binomial:** only **two** possible types of the dependent variables, 0 or 1, Pass or Fail
- **Multinomial:** there can be **3 or more** possible **unordered** types of the dependent variable, such as "cat", "dogs", or "sheep"
- **Ordinal:** there can be **3 or more** possible **ordered** types of dependent variables, such as "low", "Medium", or "High".

Linear Regression	Logistic Regression
Used to predict the continuous dependent variable using a given set of independent variables.	Used to predict the categorical dependent variable using a given set of independent variables.
Used for solving Regression problem.	Used for solving Classification problems.
We predict the value of continuous variables.	We predict the values of categorical variables.
We find the best fit line, by which we can easily predict the output.	We find the S-curve by which we can classify the samples.
Least square estimation method is used for estimation of accuracy.	Maximum likelihood estimation method is used for estimation of accuracy.
The output must be a continuous value, such as price, age, etc.	The output must be a Categorical value such as 0 or 1, Yes or No, etc.
It is required that relationship between dependent variable and independent variable must be linear.	It is not required to have the linear relationship between the dependent and independent variable.

Prepared by : Ritesh Prajapati

THANK YOU

Prepared by : Ritesh Prajapati