Analytics in a Big Data World: The Essential Guide to Data Science and Its Applications - Book by Bart Baesens, Wiley Publications, 2004

Module 2 - Predictive and Descriptive Analytics

Target Definition, Linear Regression, Logistic Regression, Decision Trees, Neural Networks, Support Vector Machines, Ensemble Methods, Multiclass Classification Techniques, Evaluating Predictive Models; Association Rules, Sequence Rules, Segmentation.

- In predictive analytics, the aim is to build an analytical model predicting a target measure of interest.
- Predictive analytics is a branch of advanced analytics that makes predictions about future outcomes using historical data combined with statistical modelling, data mining techniques and machine learning.
- Predictive analytics is often associated with big data and data science.
- To gain insights from big data, data scientists use deep learning and machine learning algorithms to find patterns and make predictions about future events. These include linear and nonlinear regression, neural networks, support vector machines and decision trees.
- Learnings obtained through predictive analytics can then be used further within prescriptive analytics to drive actions based on predictive insights.
- Predictive analytics is used in insurance, banking, marketing, financial services, telecommunications, retail, travel, healthcare, pharmaceuticals, oil and gas and other industries.
- **Ex: Fraud Detection; Medical diagnosis; Recommender Systems; Customer churn analysis; Market Basket Analysis; Weather forecasting etc.**

Predictive Analytics Definition:

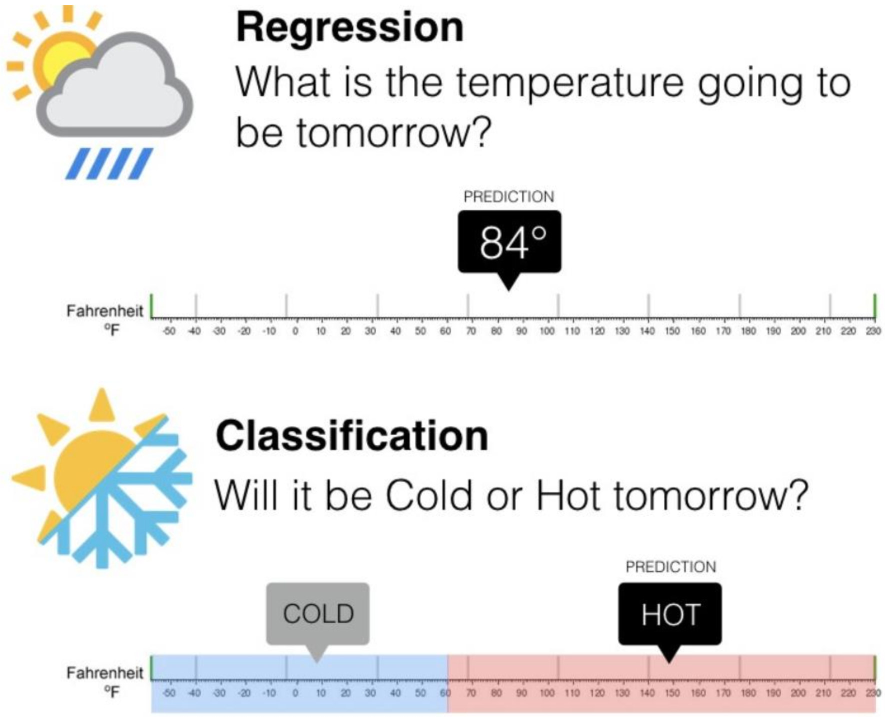
The term predictive analytics refers to the use of statistics and modelling techniques to make predictions about future outcomes and performance.
Predictive analytics looks at current and historical data patterns to determine if those patterns are likely to emerge again. This allows businesses and investors to adjust where they use their resources to take advantage of possible future events.
Predictive analytics can also be used to improve operational efficiencies and reduce risk.

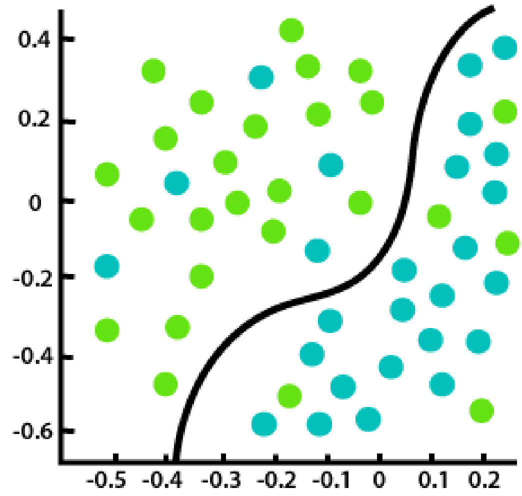
- Two types of predictive analytics can be distinguished: **Regression and Classification.**
- In regression, the target variable is continuous. Popular examples are predicting stock prices, loss given default (LGD) in banks and customer lifetime value (CLV) in general.
- In classification, the target variable is categorical. It can be binary (e.g., fraud, credit risk) or multiclass (e.g., predicting credit ratings).

Regression vs Classification:

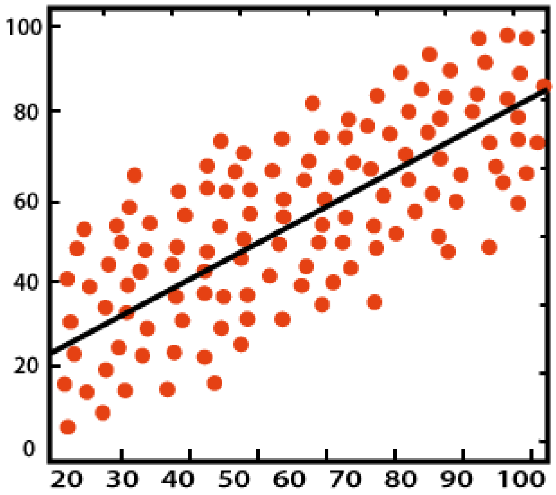
- Regression and Classification are types of Supervised Learning algorithms.
- Both the methods are used for prediction and work with the labelled datasets.
- Regression is a process of finding the correlations between dependent and independent variables. It helps in predicting the continuous variables such as prediction of **Market Trends**, **prediction of prices**, etc.

- Classification is a process of finding a function which helps in dividing the dataset into classes based on different parameters.
- The difference between both is how they are used for different problems.
- The main difference between Regression and Classification is that Regression is used to **predict the continuous** values such as price, salary, age, duration of time etc. and Classification is used to **predict/Classify the discrete values** such as Male or Female, True or False, Spam or Not Spam, etc.
- **Example:** Let's consider a dataset that contains student information of a particular university. A regression algorithm can be used in this case to predict the height of any student based on their weight, gender, diet, or subject major. We use regression in this case because height is a continuous quantity. There is an infinite number of possible values for a person's height.
- On the contrary, classification can be used to analyse whether an email is a spam or not spam. The algorithm checks the keywords in an email and the sender's address is to find out the probability of the email being spam. Similarly, while a regression model can be used to predict temperature for the next day, we can use a classification algorithm to determine whether it will be cold or hot according to the given temperature values.
-





Classification



Regression

Regression	Classification
In Regression, the output variable must be of continuous nature or real value.	In Classification, the output variable must be a discrete value.
The task of the regression algorithm is to map the input value (x) with the continuous output variable(y).	The task of the classification algorithm is to map the input value(x) with the discrete output variable(y).
Regression Algorithms are used with continuous data.	Classification Algorithms are used with discrete data.
In Regression, we try to find the best fit line, which can predict the output more accurately.	In Classification, we try to find the decision boundary, which can divide the dataset into different classes.
Regression algorithms can be used to solve the regression problems such as Weather Prediction, House price prediction, etc.	Classification Algorithms can be used to solve classification problems such as Identification of spam emails, Speech Recognition, Identification of cancer cells, etc.
The regression Algorithm can be further divided into Linear and Non-linear Regression.	The Classification algorithms can be divided into Binary Classifier and Multi-class Classifier.
Ex: **Predict the continuous** values such as price, salary, age, duration of time etc.	Ex: **Predict/Classify the discrete values** such as Male or Female, True or False, Spam or Not Spam, etc.
Simple Linear Regression, Multiple Linear Regression, Polynomial Regression, **Support Vector Regression**, **Decision Tree Regression**, Random Forest Regression	**Logistic Regression**, K-Nearest Neighbours, **Support Vector Machines**, Kernel SVM, Naïve Bayes, **Decision Tree Classification**, Random Forest Classification

Target Definition:

- The target variable plays an important role in the learning process.
- As it is of key importance in analytics process, it needs to be appropriately defined.
- **Example:** *In a customer attrition setting*, ***churn*** can be defined in various ways. **Active churn** implies that the customer stops the relationship with the firm. In a contractual setting (Say, post-paid connection) this can be easily detected when the customer cancels the contract.

In a non-contractual setting (Say, Super market) this is less obvious and needs to be operationalized in a specific way. For example, a customer churns if he or she has not purchased any products during the previous three months.
Passive churn occurs when a customer decreases the intensity of the relationship with the firm, for example, by decreasing product or service usage.
Forced churn implies that the company stops the relationship with the customer because he or she has been engaged in fraudulent activities. (Say Bank Loan)
Expected churn occurs when the customer no longer needs the product or service. (Say Baby Products).

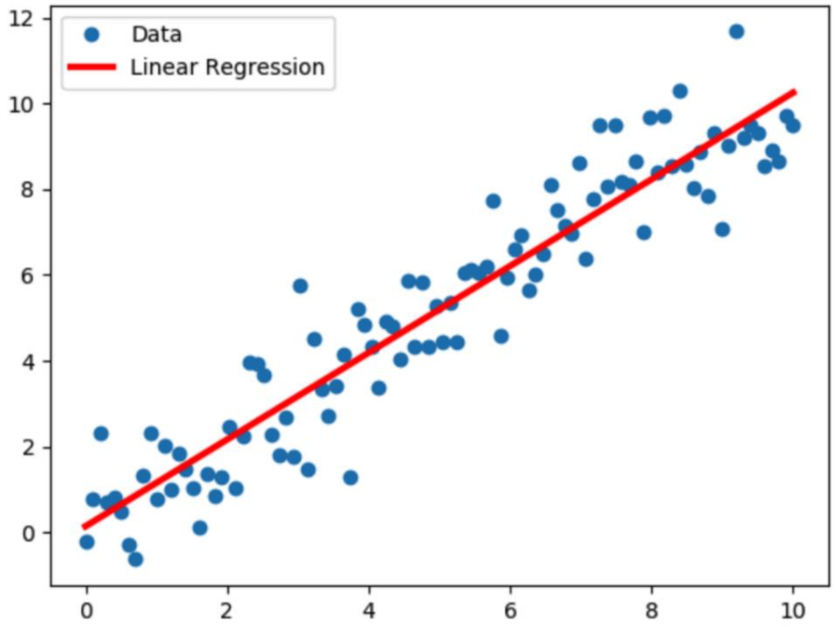
Example: *In fraud detection*, the target fraud indicator is usually hard to determine because one can never be fully sure that a certain transaction (e.g., credit card) or claim (e.g., insurance) is fraudulent. Typically, the decision is then made based on a legal judgment or a high suspicion by a business expert.

Before starting the analytical step, it is really important to check the robustness and stability of the target definition.

Linear Regression: (Regression) – Continuous Data

- Linear regression analysis is used to predict the value of a variable based on the value of another variable. The variable you want to predict is called the dependent variable. The variable you are using to predict the other variable's value is called the independent variable.
- Linear regression is a baseline modelling technique to model a continuous target/dependent variable, which assumes a linear connection between a continuous dependent variable (Y) and an independent variable (X).
- This form of analysis estimates the coefficients of the linear equation, involving one or more independent variables that best predict the value of the dependent variable.
- Linear regression fits a straight line that minimizes the discrepancies between predicted and actual output values.
- There are simple linear regression calculators that use a “least squares” method to discover the best-fit line for a set of paired data.
- It employs a regression line, also known as a best-fit line.
- The linear connection is defined as **Y = (m*X + c) + e**, where ‘c’ denotes the intercept, ‘m’ denotes the slope of the line, and ‘e’ is the error term.
- The linear regression model can be simple (with only one dependent and one independent variable) or complex (with numerous dependent and independent variables) (with one dependent variable and more than one independent variable).
- In linear regression, the model specification is that the dependent variable, y_i is a linear combination of the parameters. For example, in simple linear regression for modelling ***n*** data points, there is one independent variable x_i and 2 parameters β_0 and β_1.

$$y_i = \beta_0 + \beta_1 x_i + \varepsilon_i, \quad i = 1, \ldots, n.$$

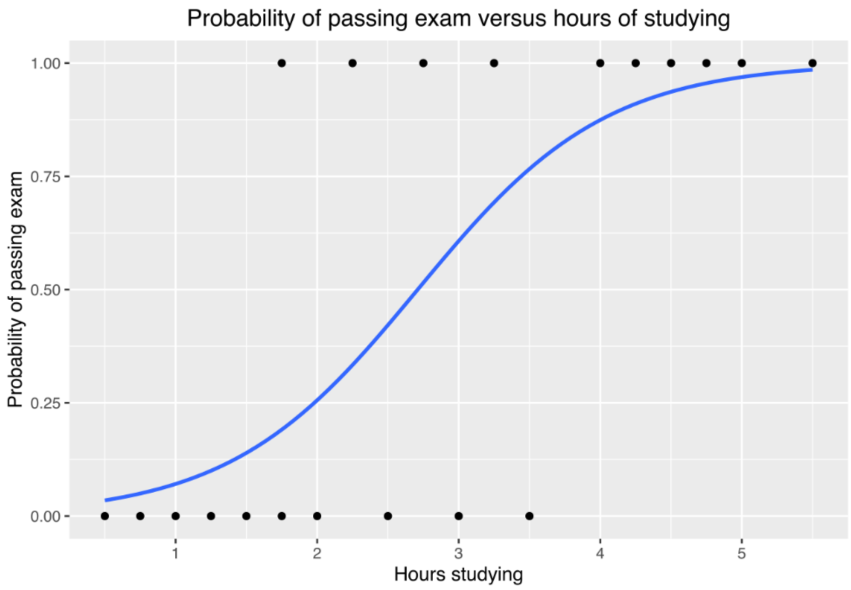


- Linear regression models are relatively simple and provide an easy-to-interpret mathematical formula that can generate predictions.
- Linear regression is used in everything from biological, behavioural, environmental and social sciences to business.
- Linear regression models have become a proven way to scientifically and reliably predict the future.

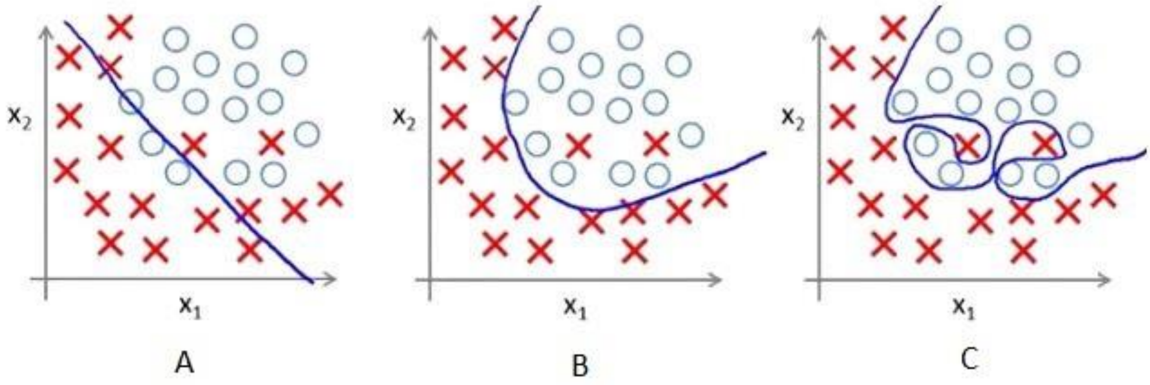
Logistic Regression(Classification) – Discrete Data

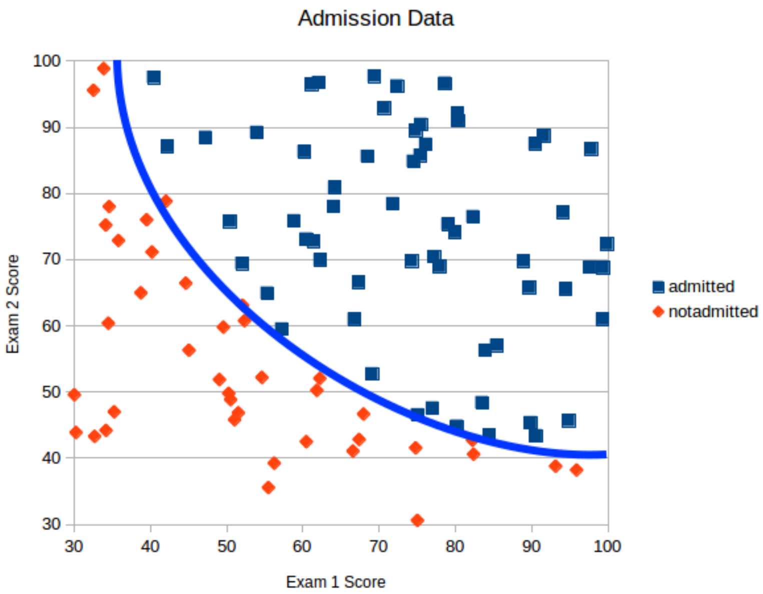
- Logistic regression is one of the most popular Machine Learning algorithms, used in the Supervised Machine Learning technique. It is used for predicting the categorical dependent variable, using a given set of independent variables. 2. It predicts the output of a categorical variable, which is discrete in nature.
- Logistic regression is a statistical analysis method to predict a binary outcome, such as yes or no, based on prior observations of a data set.
- A logistic regression model predicts a dependent data variable by analysing the relationship between one or more existing independent variables.
- Logistic regression is about fitting a curve to the data.
- Logistic regression estimates the probability of an event occurring, based on a given dataset of independent variables. Since the outcome is a probability, the dependent variable is bounded between 0 and 1.
- Linear regression provides a continuous output but Logistic regression provides discrete output.
- Logistic Regression is used when the dependent variable(target) is categorical. For example, to predict whether an email is spam (1) or (0); whether the tumour is malignant (1) or not (0).
- In logistic regression, a transformation is applied on the odds—that is, the probability of success divided by the probability of failure. This is also commonly known as the log odds and this logistic function is represented by the following formula:

$$f(x) = \frac{1}{1 + \mathrm{e}^{-(x)}}$$

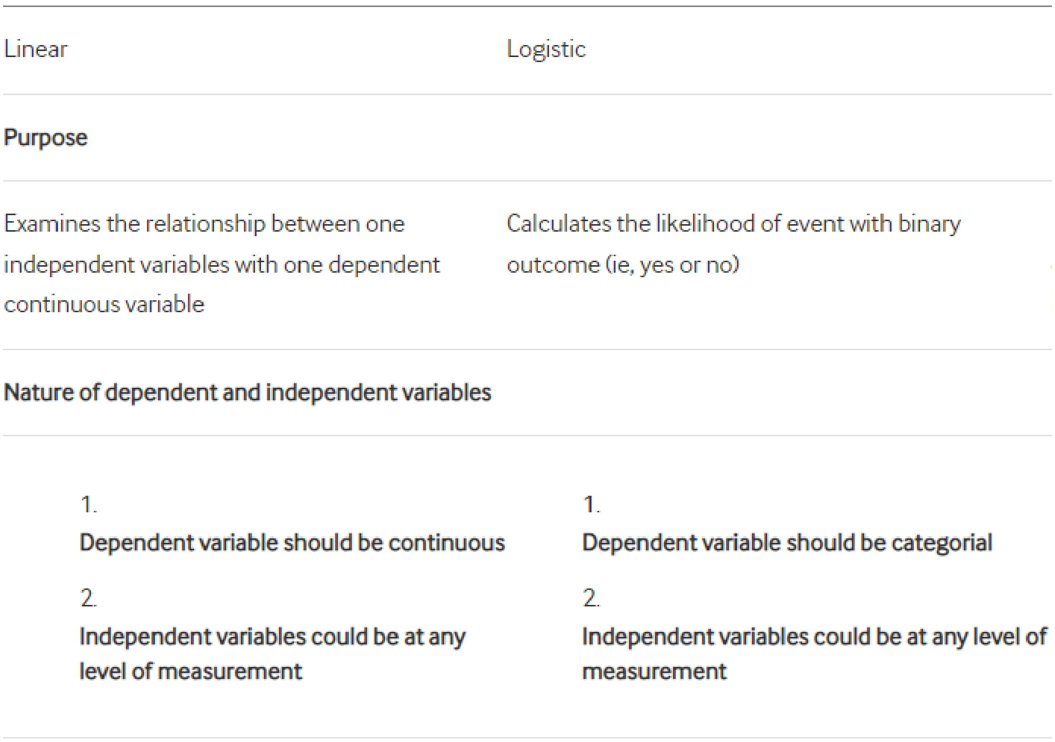


Decision boundary of logistic regression can be plotted as below:





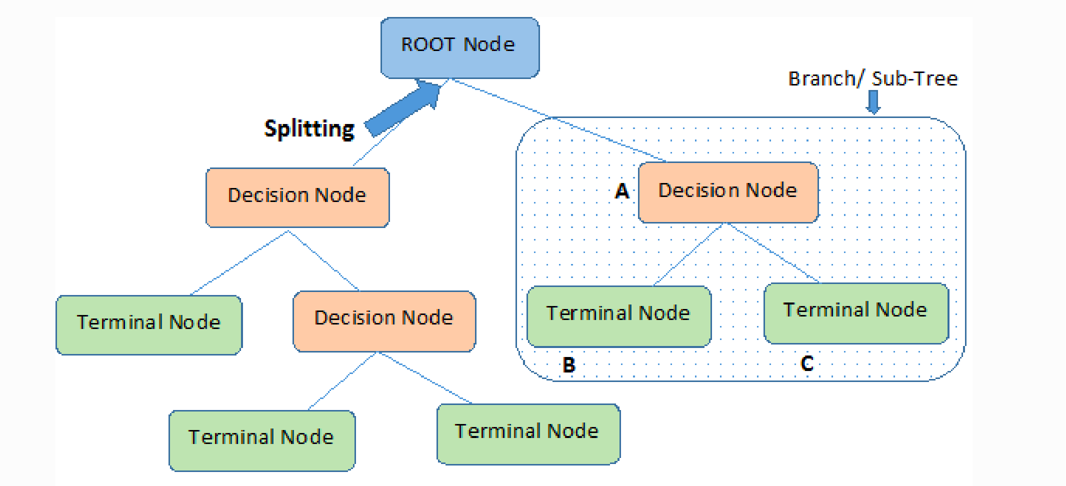
Comparison:



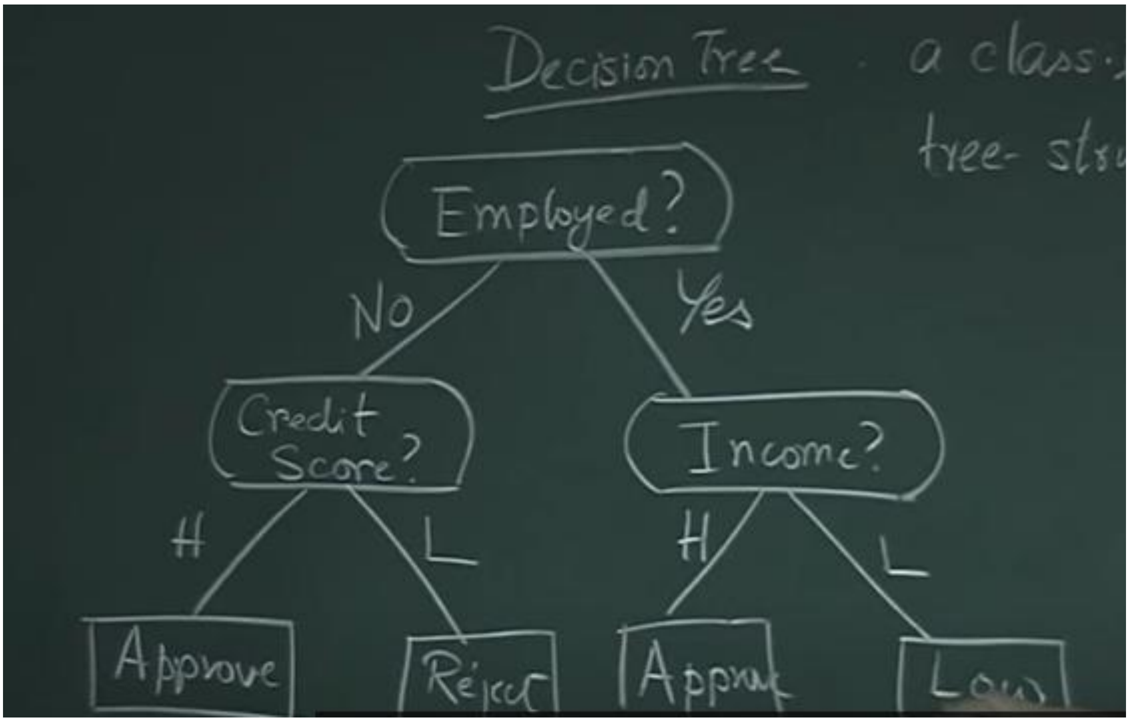
Linear	Logistic
Purpose	
Examines the relationship between one independent variables with one dependent continuous variable	Calculates the likelihood of event with binary outcome (ie, yes or no)
Nature of dependent and independent variables	
1. Dependent variable should be continuous 2. Independent variables could be at any level of measurement	1. Dependent variable should be categorial 2. Independent variables could be at any level of measurement

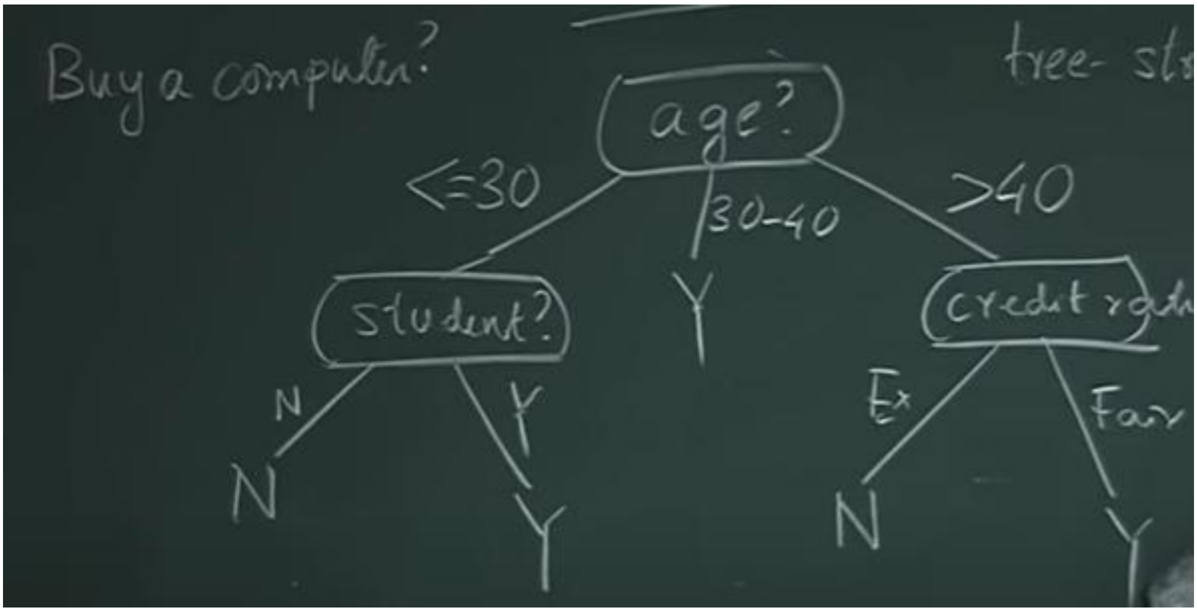
Decision Trees: (Regression and Classification)

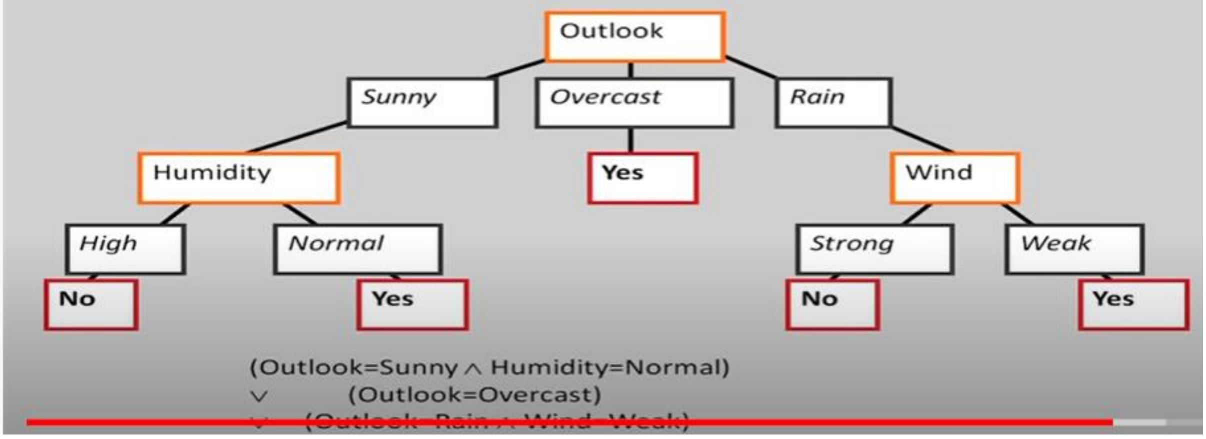
- **Decision Trees (DTs)** are a non-parametric supervised learning methods used for regression and classification.
- A Decision tree is a flowchart-like tree structure, where **each internal node denotes a test on an attribute, each branch represents an outcome of the test, and each leaf node (terminal node) holds a class label.**
- The top node is the root node specifying a testing condition of which the outcome corresponds to a branch leading up to an internal node.
- The terminal nodes of the tree assign the classifications and are also referred to as the leaf nodes.



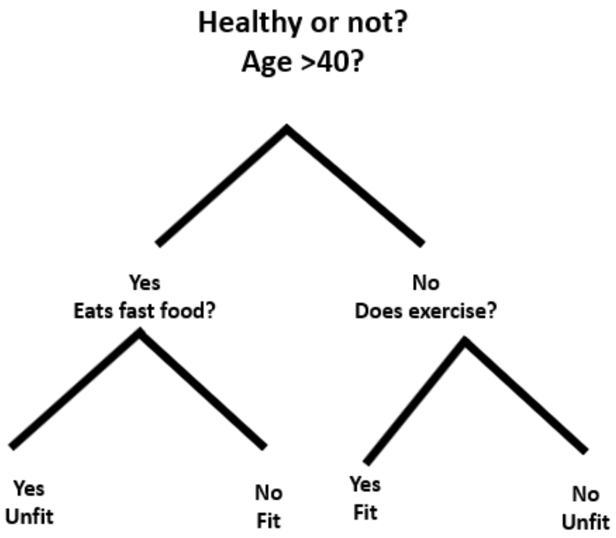
Note:- A is parent node of B and C.

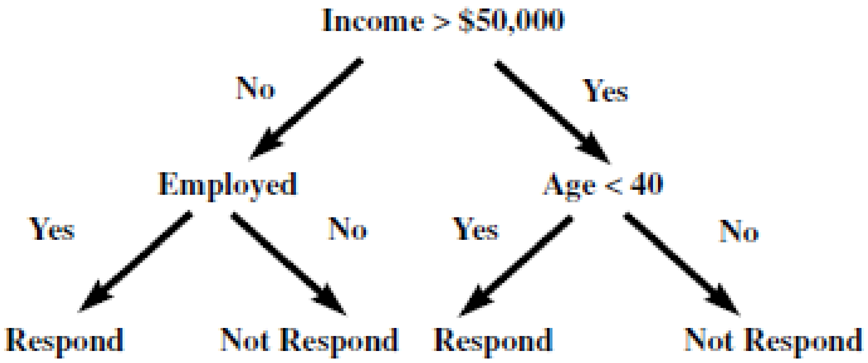






(Outlook=Sunny ∧ Humidity=Normal)
∨ (Outlook=Overcast)
∨ (Outlook=Rain ∧ Wind=Weak)





- The goal is to create a model that predicts the value of a target variable by learning simple decision rules inferred from the data features.
- Decision trees are **recursive partitioning algorithms** (RPAs) that come up with a tree-like structure representing patterns in an underlying data set.
- Many algorithms have been suggested to construct decision trees. Amongst them the most popular are: **ID3, C4.5, CART, CHAID.**
- These algorithms differ in their way of answering the key decisions to build a tree, which are:
  - ***Splitting decision***: Which variable to split at what value
  - ***Stopping decision***: When to stop growing a tree?
  - ***Assignment decision***: What class (e.g., good or bad customer) to assign to a leaf node?

- In order to answer the splitting decision, one needs to define the concept of impurity or chaos.
- A node having multiple classes is impure whereas a node having only one class is pure.

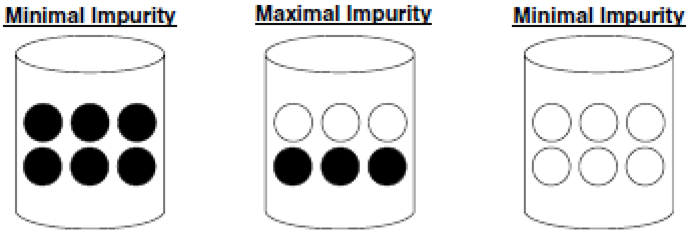


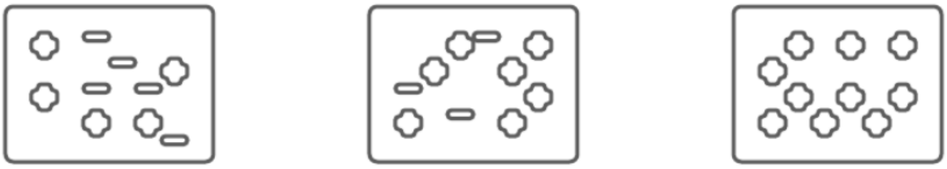
Figure 3.5 Example Data Sets for Calculating Impurity

- In Decision Tree the major challenge is to identification of the attribute for the root node in each level. This process is known as attribute selection.

  We have two popular attribute selection measures/tree splitting criterion:

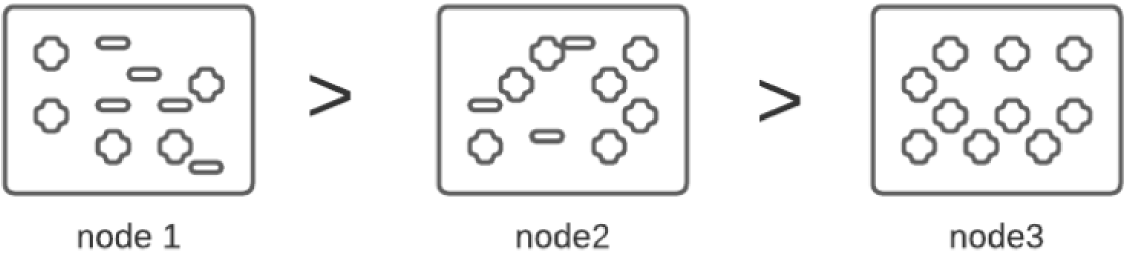
  1. **Information Gain**
  2. **Gini Index**

- **Entropy** - Measures the impurity or uncertainty in a group of observations. It determines how a decision tree chooses to split data.
- **Information Gain -** To perform a right split of the nodes in case of large variable holding data set information gain comes into the picture. It is the amount of information gained/improved/required in the nodes before splitting them for making further decisions.



In node3 we don't need to make any decision because 100% of the instances are representing the direction of the decision in favour of Class1, 70% of instances are representing Class1 in node2 wherein in node1 there are 50% chances to decide the direction of both classes.

We can say that in node1, we require more information than the other nodes to describe a decision. By the above, we can say the information gain in node 1 is higher.



By the above, we can say the balanced nodes or most impure nodes require more information to describe.

To measure the information gain we use the entropy.
Information Gain = 1 – Entropy

Gini Index: Gini Index is a metric to measure how often a randomly chosen element would be incorrectly identified.

It means an attribute with lower Gini index should be preferred.

The Formula for the calculation of the of the Gini Index is given below.

$$GiniIndex = 1 - \sum_j p_j^2$$

Advantages of decision trees are:

- Simple to understand and to interpret, as Trees can be visualized.
- Requires little data preparation.
- The cost of using the tree (i.e., predicting data) is logarithmic in the number of data points used to train the tree.
- Able to handle both numerical and categorical data.
- Able to handle multi-output problems.
- Possible to validate a model using statistical tests making it reliable model.
- Decision trees can handle high-dimensional data.
- Decision tree classifier has good accuracy.

The disadvantages of decision trees include:

- Decision-tree learners can create over-complex trees that do not generalize the data well.
- Decision trees can be unstable because small variations in the data might result in a completely different tree being generated.
- They are not good at extrapolation.
- Decision tree learners create biased trees if some classes dominate.

Construction of Decision Tree:

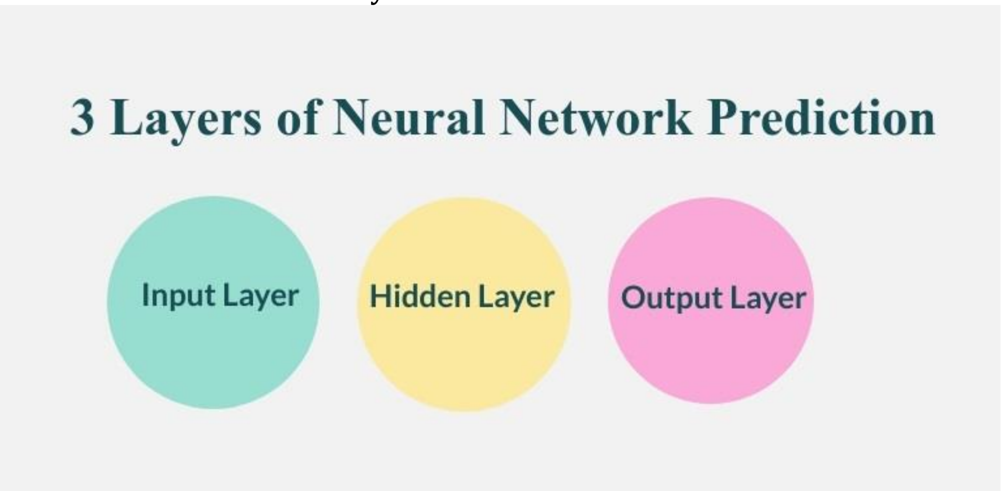
- Split the source set into subsets based on an attribute value test. This process is repeated on each derived subset in a recursive manner called recursive partitioning.
- The recursion is completed when the subset at a node all has the same value of the target variable, or when splitting no longer adds value to the predictions.

Neural Networks: (Regression and Classification):

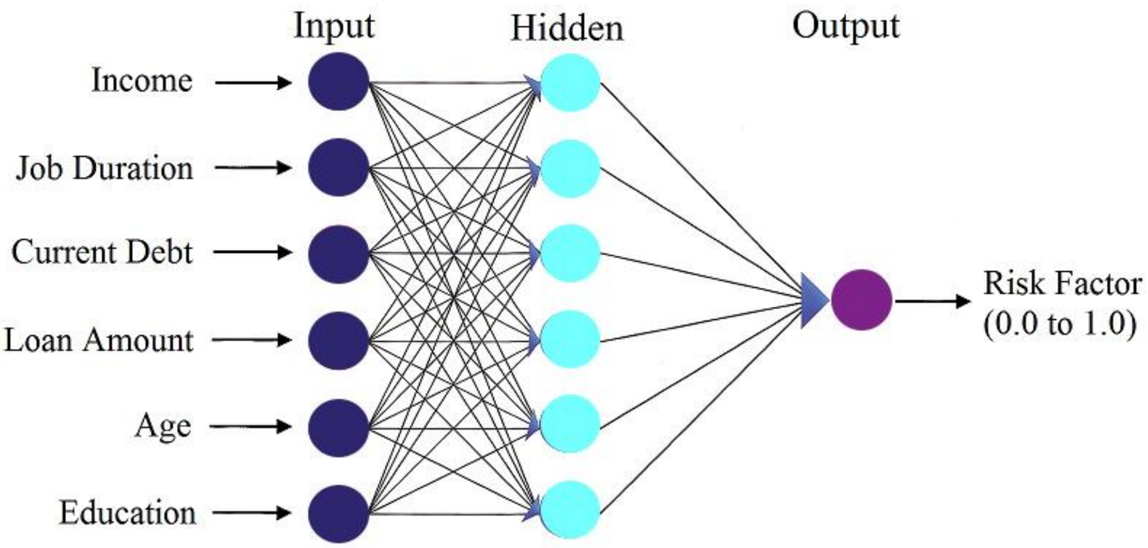
- Neural networks consist of simple input/output units called **neurons** (inspired by neurons of the human brain).
- Neural networks are mathematical representations inspired by the functioning of the human brain.
- On the other hand, Neural networks can be visualised as generalizations of existing statistical models.
- **For Ex:** Logistic regression is a neural network with one neuron. Similarly, Linear regression is also a one neuron neural network.
- Neural networks can model **very complex patterns** and decision boundaries in the data and, as such, are very powerful.
- Neural networks are flexible and can be used for both **classification and regression.**
- It creates an **adaptive system** that computers use to learn from their mistakes and improve continuously.

- There are three layers to the structure of a neural-network algorithm:
    - **The input layer**: This enters past data values into the next layer.
    - **The hidden layer**: This is a key component of a neural network. It has complex functions that create predictors. A set of nodes in the hidden layer called neurons represents math functions that modify the input data.
    - **The output layer**: Here, the predictions made in the hidden layer are collected to produce the final layer – which is the model's prediction.

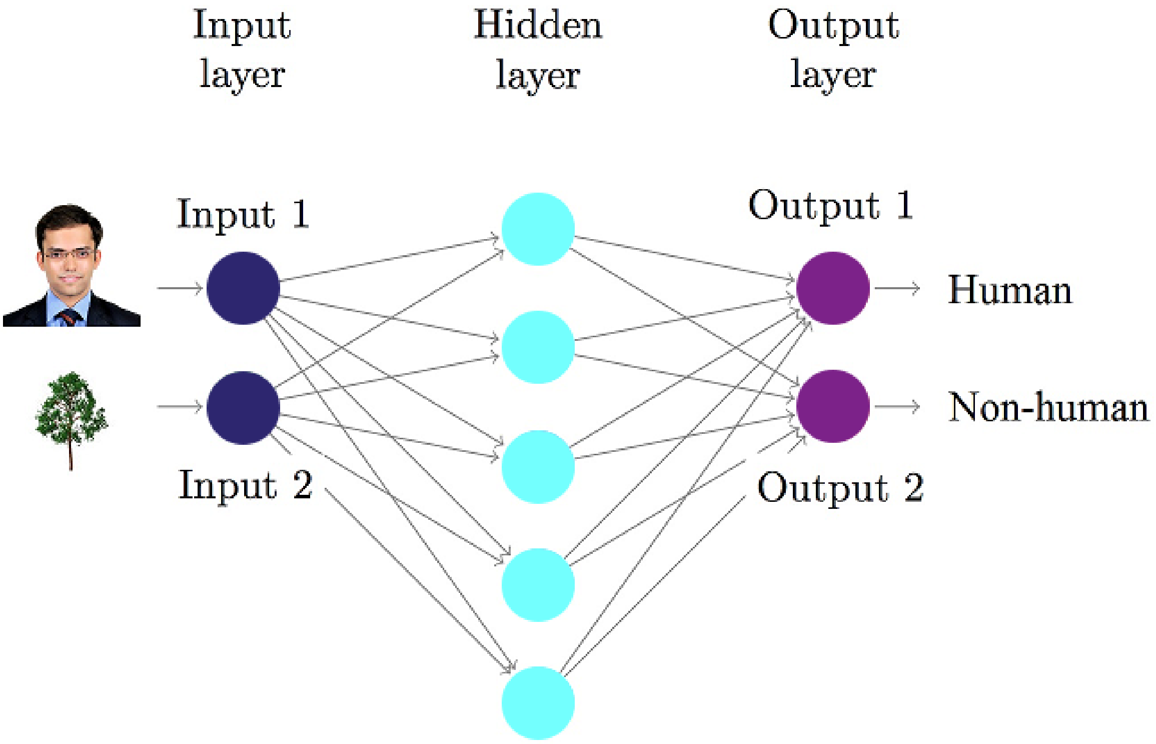
  In addition to the above 3 layers, Neural Network also has an **activation function** to decide the value of hidden layer neurons.



- **Forward Propagation** is the way to move from the Input layer (left) to the Output layer (right) in the neural network. The process of moving from the right to left i.e backward from the Output to the Input layer is called the **Backward Propagation.**



Model a neural network for banking system that predicts debtor risk



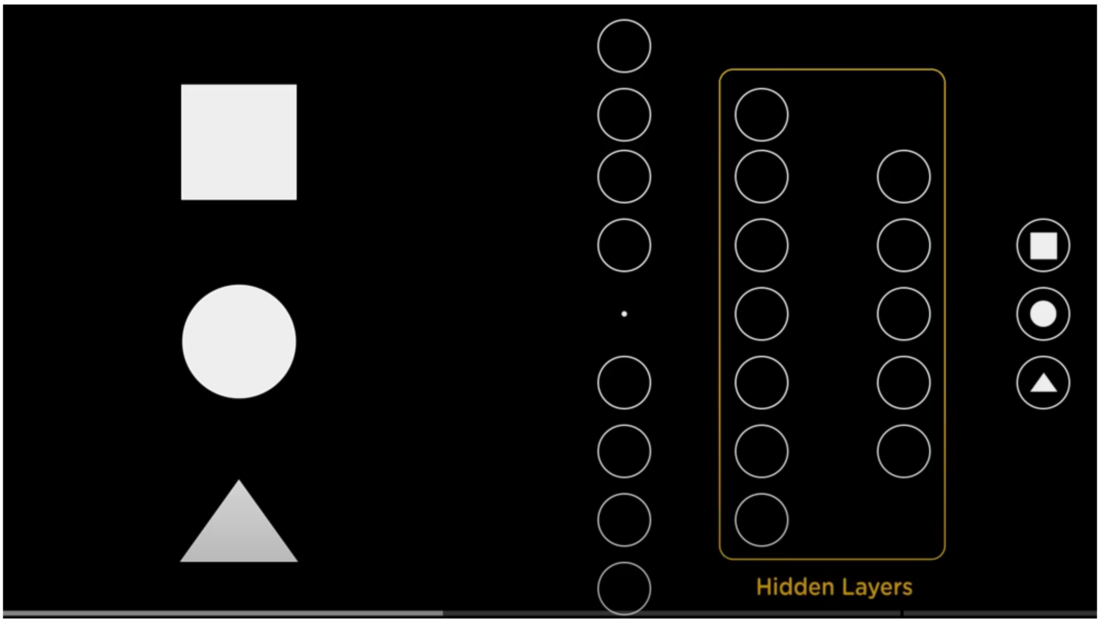
Model for Neural network Image recognition for human and non-human

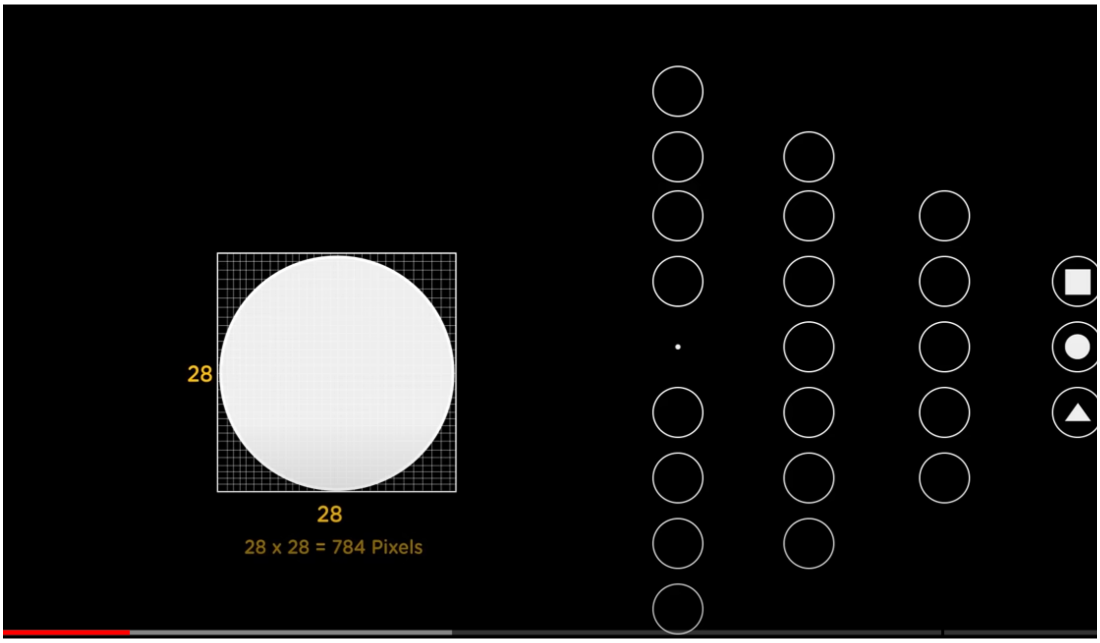
- Neural networks have **several use cases** across many industries, such as the following:
    - **Medical diagnosis** by medical image classification
    - Targeted **marketing** by social network filtering and behavioural data analysis
    - **Financial** predictions by processing historical data of financial instruments
    - Electrical **load and energy demand** forecasting
    - **Process and quality** control
    - Chemical compound identification

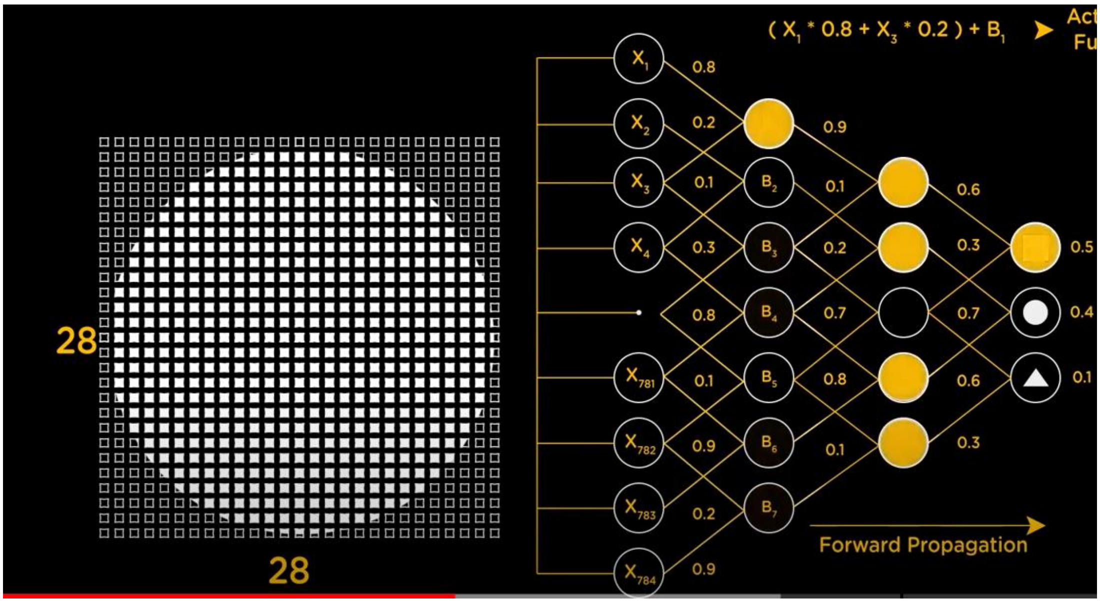
- Important **applications** of neural networks include **Computer Vision, Speech Recognition, Natural Language Processing, Recommender Systems** etc.

- Three important **types** of neural networks are **Artificial Neural Networks(ANN), Convolutional Neural Networks(CNN) and Recurrent Neural Networks(RNN).**

- How Neural Network Works: Ex: Identification of Circle from square, circle, triangle.


Hidden Layers


28
28
28 x 28 = 784 Pixels


(X1 * 0.8 + X3 * 0.2) + B1
Act
Fu
28
28
X1
X2
X3
X4
X781
X782
X783
X784
0.8
0.2
0.1
0.3
0.8
0.1
0.9
0.2
0.9
B2
B3
B4
B5
B6
B7
0.9
0.1
0.2
0.7
0.8
0.1
0.6
0.3
0.7
0.6
0.3
0.5
0.4
0.1
Forward Propagation

Advantages of Neural Networks:

Efficiency
Store information on the entire network
Good fault tolerance
The ability to work with insufficient knowledge
Distributed memory
The ability of parallel processing/ Multitasking
Wide Applications

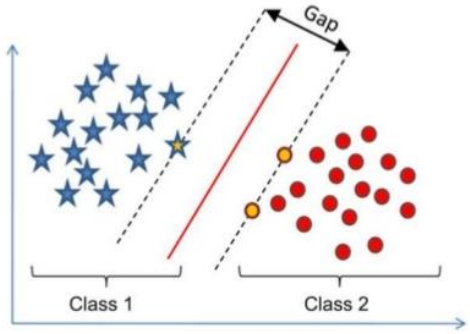
Disadvantages of Neural Networks:

Hardware dependent
Complex Algorithms are required
Black Box Nature
Data-dependency
Approximate Results

Support Vector Machines: (Regression and Classification)

- Support Vector Machine (SVM) is a supervised machine learning algorithm used for both **classification and regression**.
- The idea of SVM is simple: **The algorithm creates a line or a hyperplane which separates the data into classes.**
- Hyper plane is an (n-1) dimensional space for an n-dimensional space. For a 2-dimension space, its hyperplane will be 1-dimension, which is just a line. For a 3-dimension space, its hyperplane will be 2-dimension, which is a plane that slice the cube.
- **Support vectors** are data points that are **closer to the hyperplane** and influence the position and orientation of the hyperplane. Using these support vectors, we maximize the margin of the classifier.
- Hyperplanes are decision boundaries that help classify the data points. Data points falling on either side of the hyperplane can be attributed to different classes.
- The **dimension of the hyperplane depends upon the number of features**. If the number of input features is two, then the hyperplane is just a line. If the number of input features is three, then the hyperplane becomes a 2-D plane. It becomes difficult to imagine when the number of features exceeds three.
- To separate the two classes of data points, there are many possible hyperplanes that could be chosen. The objective is to find a plane that has the maximum margin, i.e., to pull both the classes as far apart as possible.
- The objective of the support vector machine algorithm is to find a hyperplane in an N-dimensional space (N — the number of features) that distinctly classifies the data points.

Basic concept of SVM



SVM algorithm is all about finding a linear decision surface (Hyper plane) that can separate classes and that hyper plane has the largest distance (gap or margin) between border line elements (support vectors)

Concept: At first approximation what SVMs do is to find a separating line (or hyperplane) between data of two classes. SVM is an algorithm that takes the data as an input and outputs a line that separates those classes if possible.

Lets begin with a problem. Suppose you have a dataset as shown below and you need to classify the blue rectangles from the green circles. So your task is to find an ideal line that separates this dataset in two classes (say blue and green).

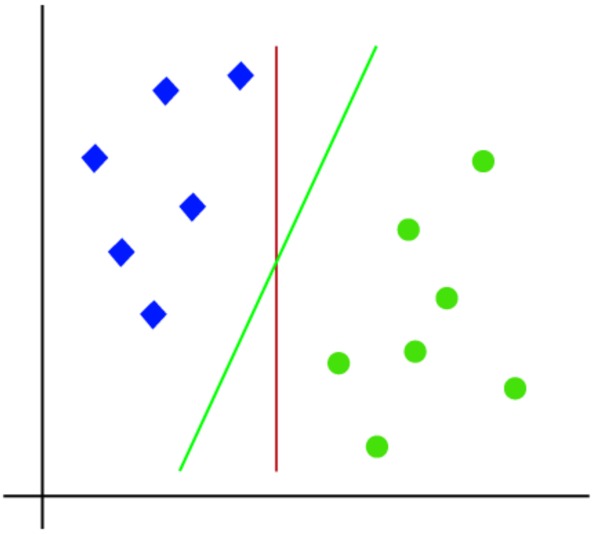
As we notice there are an infinite lines that can separate these two classes. So how does SVM find the ideal one???

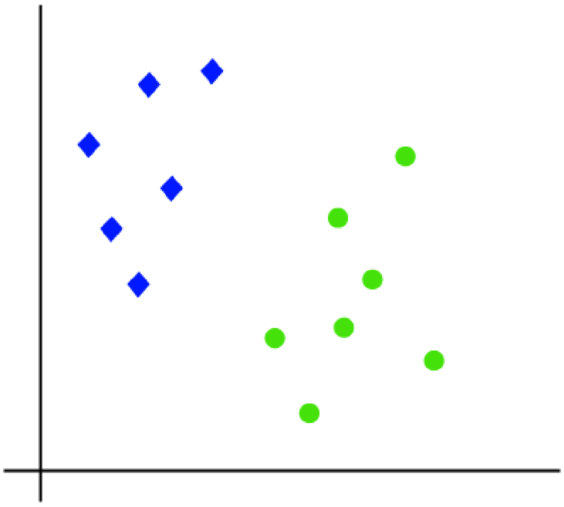
We have two candidates here, the red coloured line and the green coloured line. Which line according to you best separates the data?

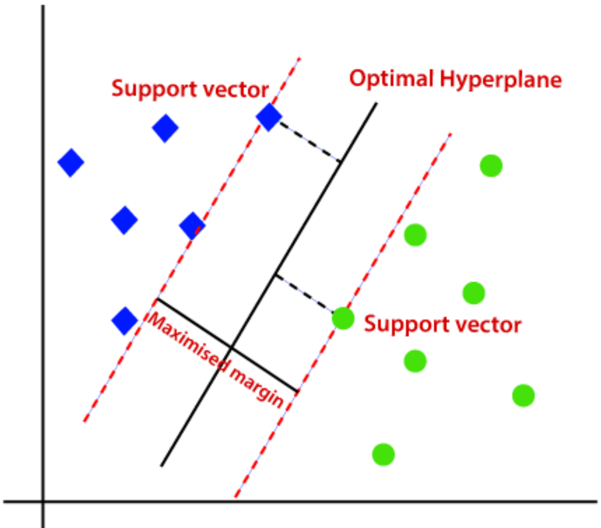
The red line in the image is quite close to the blue class. Though it classifies the current datasets, it is not a generalized line and our goal is to get a more generalized separator.

Next we have the green line and this line looks like solution from SVM because green line classifies better.

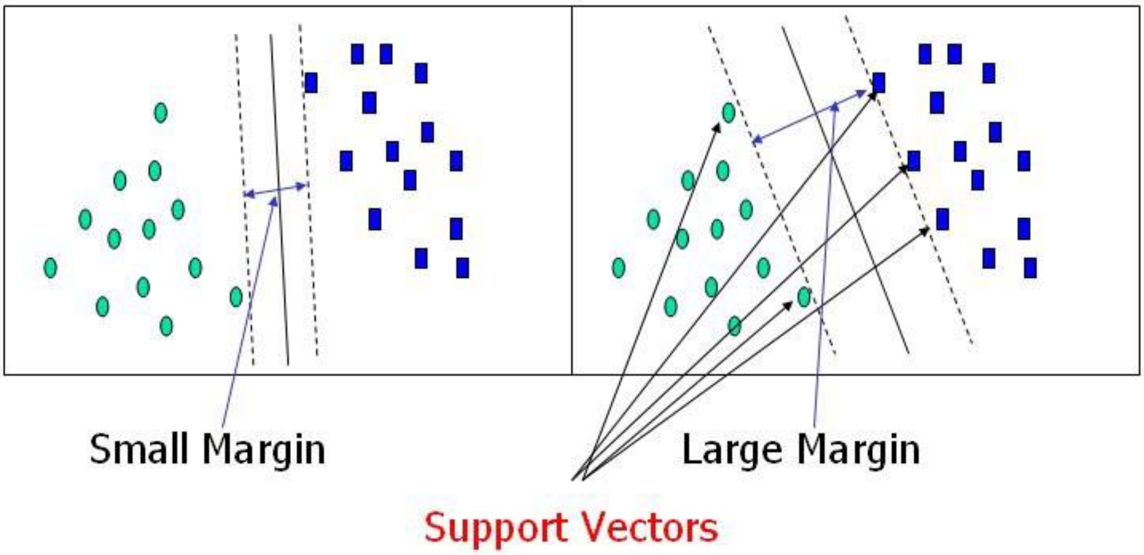
According to the SVM algorithm, we find the points closest to the line from both the classes. These points are called **support vectors.** Now, we compute the distance between the line and the support vectors. This distance is called the **margin**. Our goal is to maximize the margin. The hyperplane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







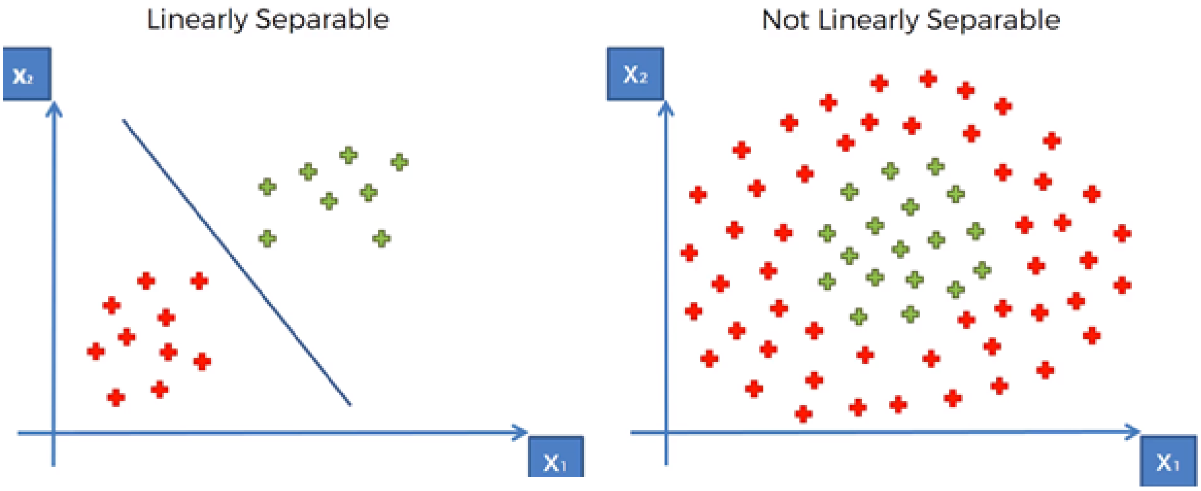
Thus SVM tries to make a decision boundary in such a way that the separation between the two classes is as wide as possible.

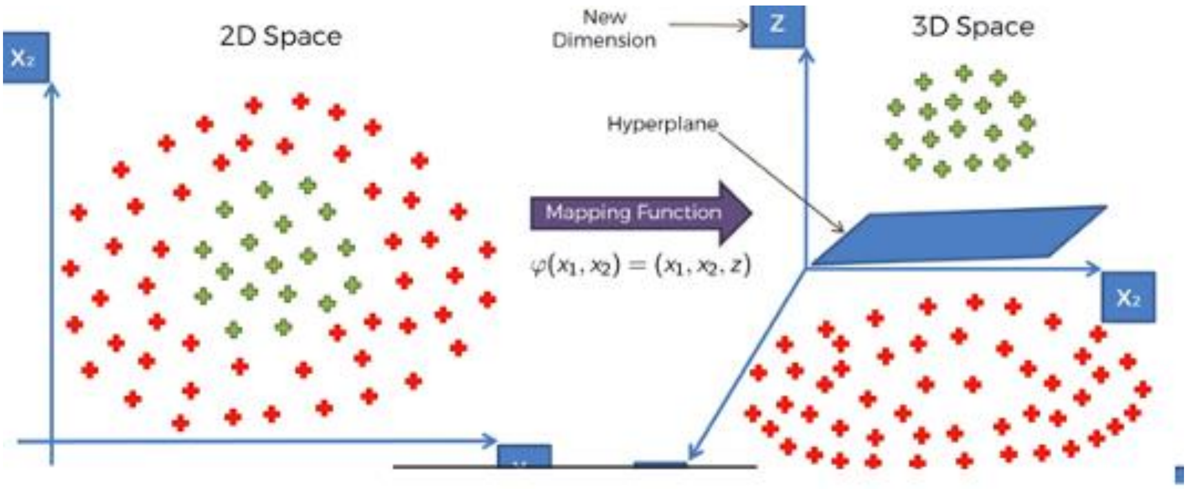


Types of SVM:

Linear SVM:

Non-Linear SVM: When unable to classify - Increase the dimension; Kernel Trick (RBF, Linear, Polynomial etc.)





Applications of Support Vector Machine(SVM):

1. **Face observation**
2. **Bioinformatics**
3. **Handwriting recognition**

Advantages of SVM:

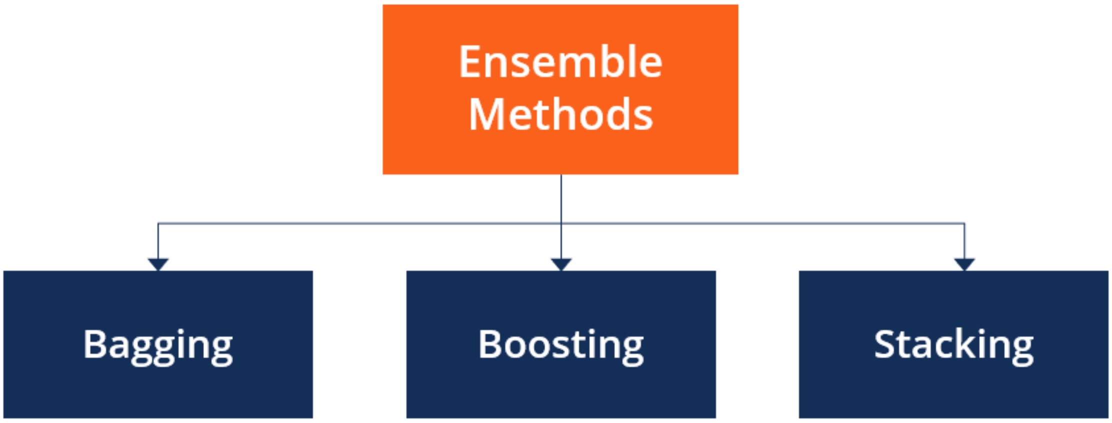
- Effective in high dimensional cases
- Its memory efficient as it uses a subset of training points in the decision function called support vectors

Disadvantages of SVM:

- Support vector machine algorithm is not acceptable for large data sets.
- It does not execute very well when the target classes are overlapping.
- In cases where the number of properties for each data point outstrips the number of training data specimens, the support vector machine will underperform.
- As the support vector classifier works by placing data points, above and below the classifying hyperplane there is no probabilistic clarification for the classification.

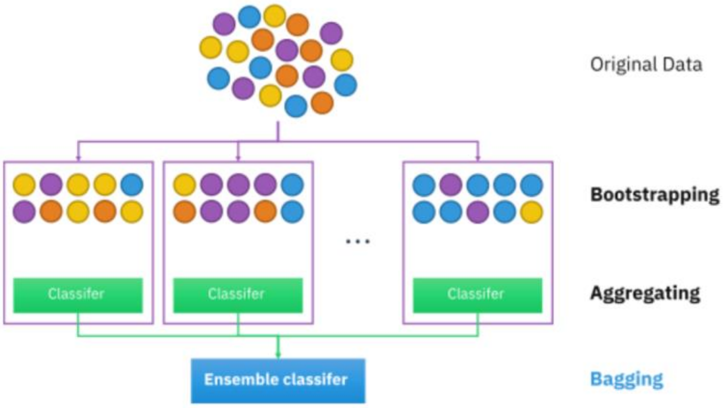
Ensemble Methods: (Classification):

- **Ensembling is a method of combining multiple individual models to create a master model for efficient and accurate prediction.**
- **Ensemble learning** is a general approach that seeks better predictive performance by combining the predictions from multiple models.
- Ensemble methods combine several base models in order to produce one optimal predictive model.
- Ensemble methods aim at estimating multiple analytical models instead of using only one.
- The idea here is that multiple models can cover different parts of the data input space and, as such, complement each other's deficiencies.
- The two main classes of ensemble learning methods are **bagging, boosting,stacking.**

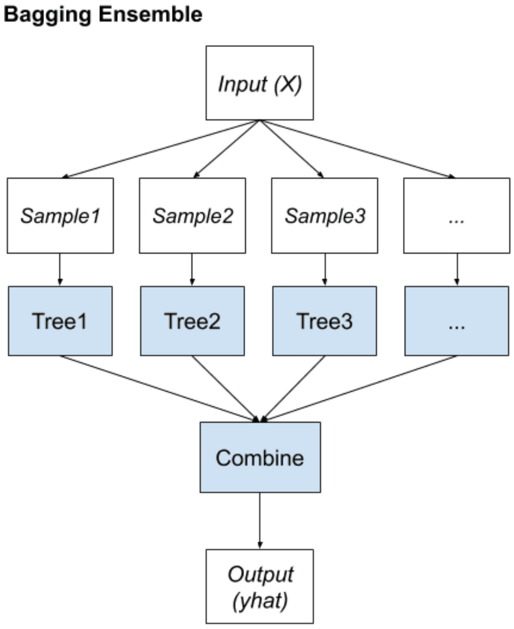


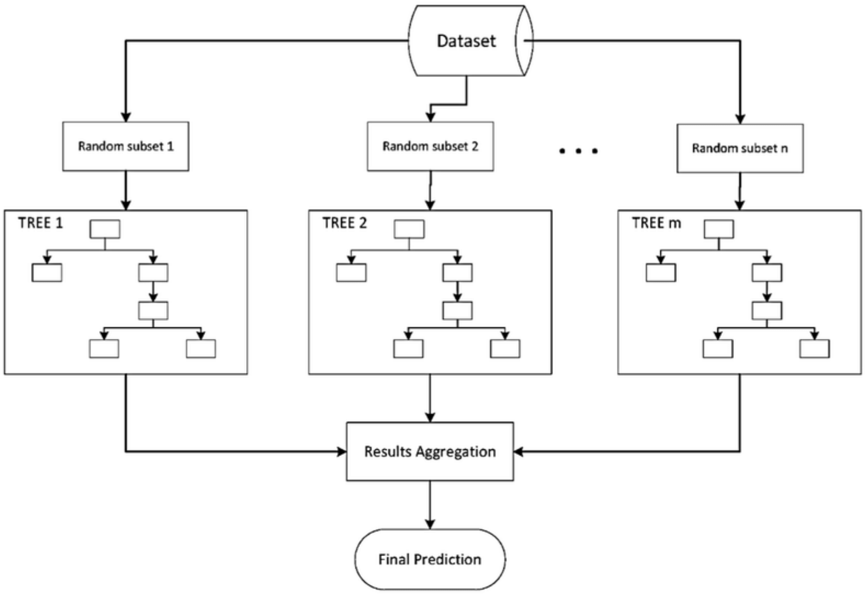
Bagging: (**B**ootstrap **agg**regat**ing**)

- Bagging gets its name because it combines ***Boo***tstrapping and ***Agg***regation to form one ensemble model.
- Bagging is a way to decrease the variance in the prediction by generating additional data for training from dataset using combinations with repetitions to produce multi-sets of the original data.
- This model learns from each other independently in parallel and combines them for determining the model average.
- Given a sample of data, multiple bootstrapped subsamples are pulled. The idea is then to build a classifier (e.g., decision tree) for every bootstrap. (Bootstrapping means randomly drawn observations from the original data, with replacement).



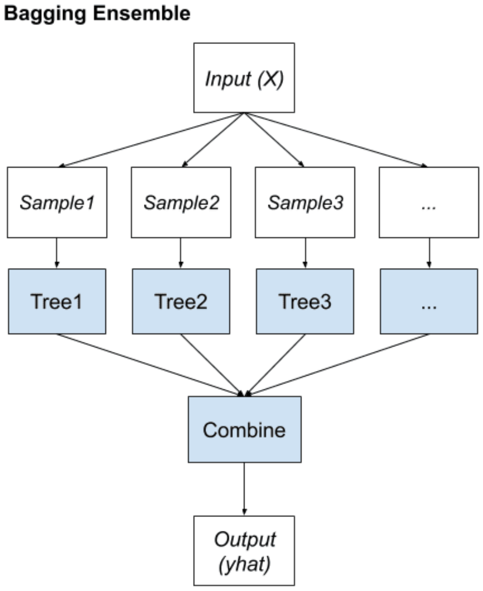
- A Decision Tree is formed on each of the bootstrapped subsamples. After each subsample Decision Tree has been formed, an algorithm is used to aggregate over the Decision Trees to form the most efficient predictor.
- The image below will help explain:





Bagging Steps:

- Suppose there are N observations and M features in training data set. A sample from training data set is taken randomly **with replacement**.
- A subset of M features are selected randomly and whichever feature gives the best split is used to split the node iteratively.
- The tree is grown to the largest.
- Above steps are repeated n times and prediction is given based on the aggregation of predictions from n number of trees.



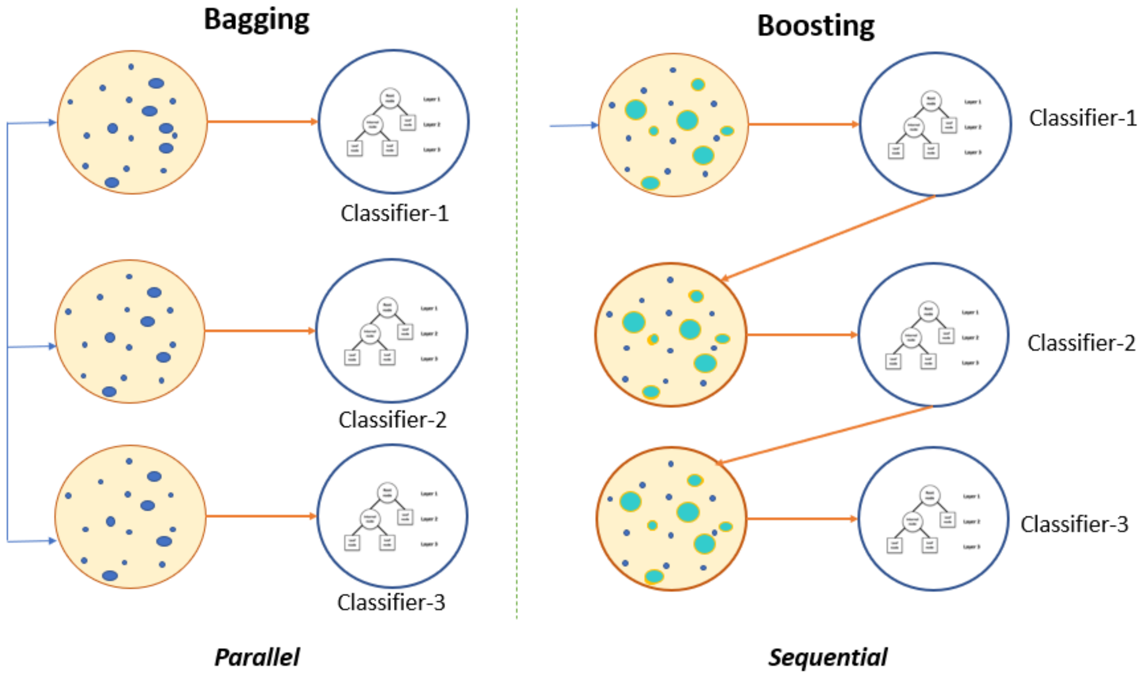
Boosting:

- Boosting is an ensemble modelling technique that attempts to build a strong classifier from the number of weak classifiers.
- Boosting is an iterative technique which adjusts the weight of an observation based on the last classification.
- Boosting is used to create a collection of predictors.
- In this technique, learners learn sequentially with early learners fitting simple models to the data and then analysing data for errors. Consecutive trees (random sample) are fit and at every step, the goal is to improve the accuracy from the prior tree.
- Bagging is parallel while boosting is sequential.

Boosting Steps:

- Draw a random subset of training samples d1 **without replacement** from the training set D to train a weak learner C1.
- Draw second random training subset d2 without replacement from the training set and add 50 percent of the samples that were previously falsely classified/misclassified to train a weak learner C2.
- Find the training samples d3 in the training set D on which C1 and C2 disagree to train a third weak learner C3.
- Combine all the weak learners via majority voting.

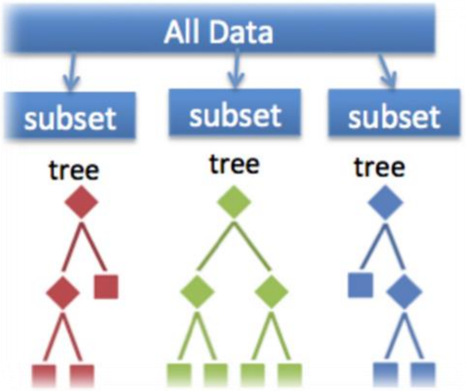
Differences Between Bagging and Boosting:



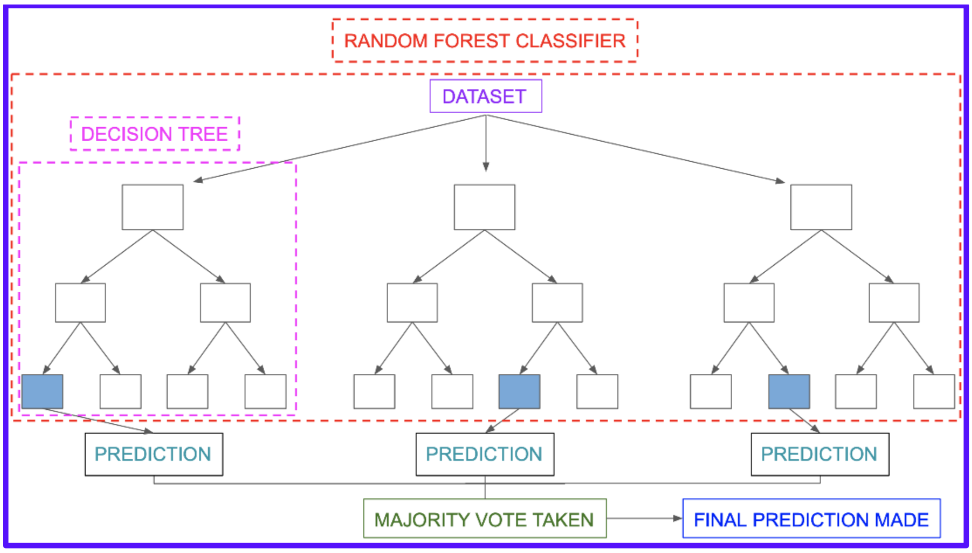
S.NO	Bagging	Boosting
1.	The simplest way of combining predictions that belong to the same type.	A way of combining predictions that belong to the different types.
2.	Each model receives equal weight.	Models are weighted according to their performance.
3.	Each model is built independently.	New models are influenced by the performance of previously built models.
4.	Different training data subsets are selected using row sampling with replacement and random sampling methods from the entire training dataset.	Every new subset contains the elements that were misclassified by previous models.
5.	If the classifier is unstable (high variance), then apply bagging.	If the classifier is stable and simple (high bias) the apply boosting.
6.	In bagging base classifiers are trained parallelly.	In boosting base classifiers are trained sequentially.
7.	Aims at decreasing variance (Overfitting leads to high variance – Overfitting means that the model performs well on the training data but does not perform accurately in the evaluation set.)	Aims at decreasing bias (Bias is a phenomenon that skews the result of an algorithm either in favour or against an idea).
8.	**Example:** The **Random forest** model uses Bagging.	**Example:** The **AdaBoost** algorithm uses Boosting.

Random Forests:

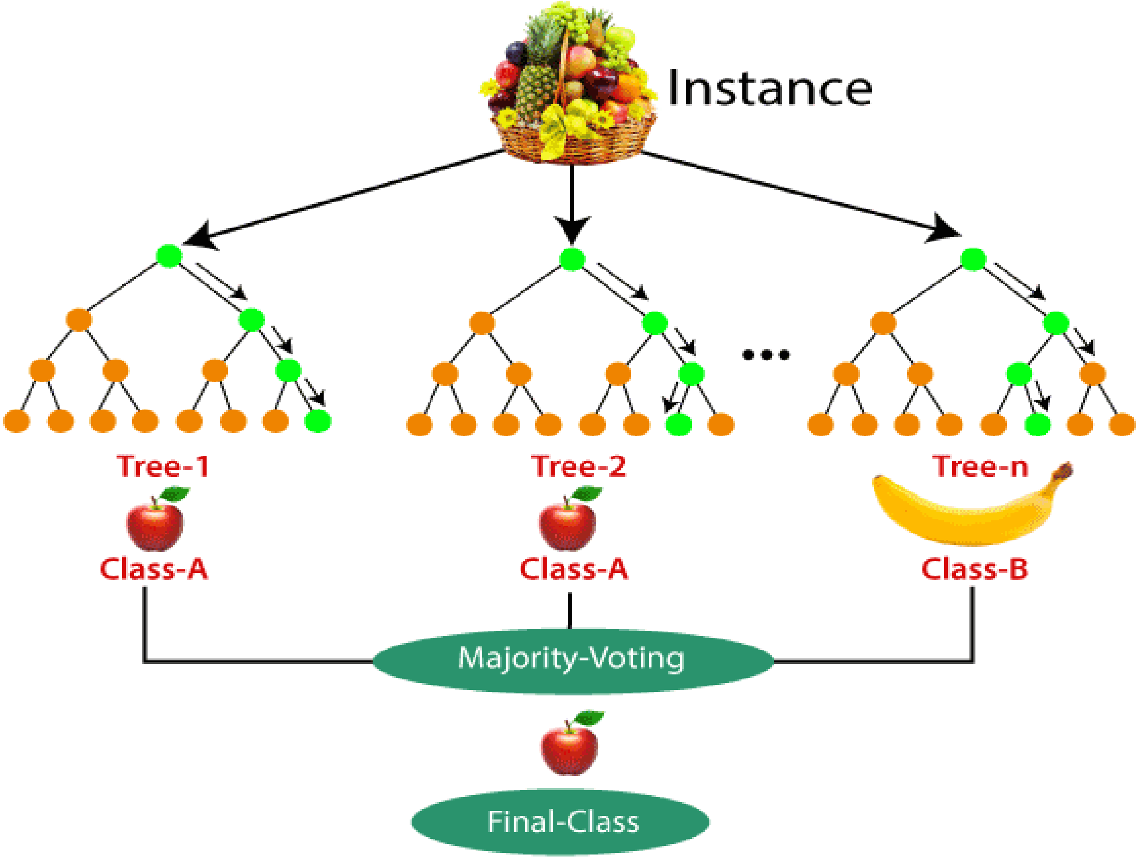
- Random forest is an example for Bagging.
- Random forest, like its name implies, consists of a large number of individual decision trees that operate as an ensemble.
- Each individual tree in the random forest splits out a class prediction and the class with the most votes becomes our model's prediction.
- The random forest is an ensemble learning method for classification/regression consisting of many decisions trees.
- Random forests are based on two key concepts namely **bagging and feature selection.**
- The key principle underlying the random forest approach comprises the construction of many "simple" decision trees in the training stage and the majority vote (mode) across them in the classification stage.
- In the training stage, random forests apply the general technique known as bagging to individual trees in the ensemble.
- Random Forest models decide where to split based on a random selection of features.
- Ex: Consider below image. In Random Forest, bootstrapped subsamples are pulled from a larger dataset, like in bagging. A decision tree is formed on each subsample. However, the decision tree is split on different features (in this diagram the features are represented by shapes).



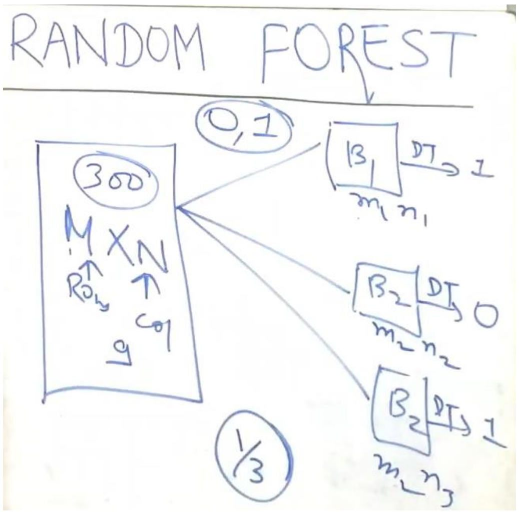
- Rather than splitting at similar features at each node throughout, Random Forest models implement a level of differentiation because each tree will split based on different features.
- This level of differentiation provides a greater ensemble to aggregate over, producing a more accurate predictor. At the end, the final prediction is given based on the concept of voting.
- For classification majority voting is considered for final prediction while for regression either mean or average of the voting is considered.



RANDOM FOREST CLASSIFIER
DATASET
DECISION TREE
PREDICTION
PREDICTION
PREDICTION
MAJORITY VOTE TAKEN
FINAL PREDICTION MADE



Instance
Tree-1
Tree-2
Tree-n
Class-A
Class-A
Class-B
Majority-Voting
Final-Class



RANDOM FOREST
0,1
300
M X N
Rows
Col
9
B1
DT
1
m1 n1
B2
DT
0
m2 n2
B2
DT
1
m2 n3
1/3

Miscellaneous Details:

Students may use the following data sets for illustration purposes.

Age	Salary	Travel Distance	Leave or Not
25	5 LPA	5	No
30	8 LPA	8	No
35	10 LPA	10	No
40	10 LPA	5	Yes
45	15 LPA	15	Yes
50	22 LPA	10	No

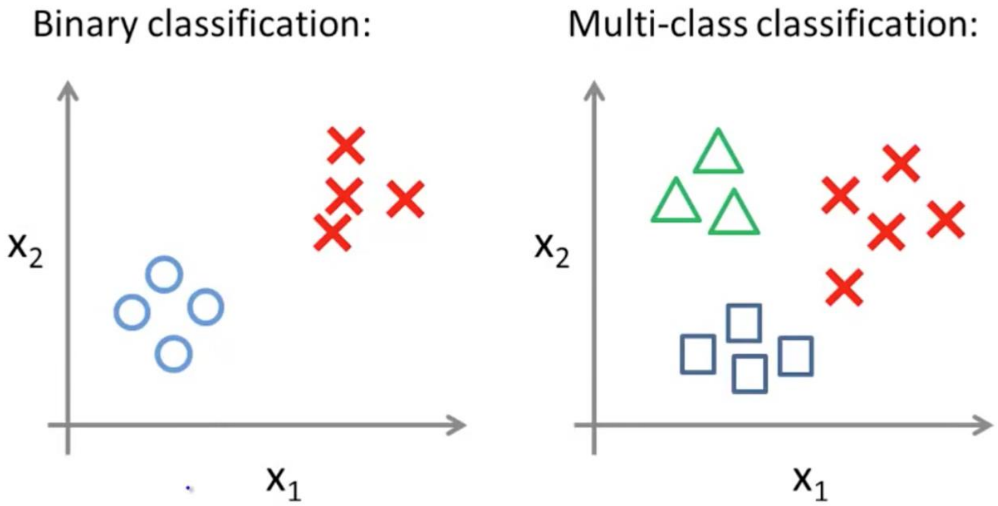
Customer Id	Loan Amount	Loan Status
1	100	Good
2	150	Good
3	500	Bad
4	200	Good
5	300	Bad

USN	CGPA	Placement
1	8.0	Good
2	7.0	Good
3	6.5	Bad
4	7.5	Good

Age	BMI	Gender
25	24	F
41	31	F
56	28	M
78	26	F
62	30	M

Multiclass Classification Techniques:

- In a binary classification, something like ***predicting 1 or 0***; ***the patient is diabetic or not diabetic***; the ***person is fraud or no-fraud***; etc. are determined meaning predicting two classes as target classes.
- In a multiclass classification problem, we will have multiple features and multiple classes.
- For example, in the case of identification of different types of vehicles, “Load”, “Length”, “Wheels”, “Type” can be features, while “Car”, “Bus”, “Truck”, “Boat” can be different class labels.
- Multiclass classification is a popular problem in supervised machine learning.
- If target classes or target variables or outputs are more than 2 in number, then we say such problem is a multi-class problem.



- Before applying multiclass models for analysis, one should know whether the target variables are nominal (relative – height of a person, blood group of a person, eating habits of a person) or ordinal (absolute – rank in a class, position in a queue, credit rating).
- A classification task with more than two classes; e.g., classify a set of images of vehicles which may be Car, Bus, Truck or Boat is a multiclass classification problem.
- The following are some of the multiclass classification techniques.
    - **Multiclass Logistic Regression**
    - **Multiclass Decision Trees**
    - **Multiclass Neural Networks**
    - **Multiclass Support Vector Machines**
- Multi-class classification makes the assumption that each sample is assigned to one and only one label: a vehicle can be either a car or a bus but not both at the same time.
- A common practise to estimate a multiclass classification technique is to map the multiclass classification problem to a set of binary classification problems. Also, a multiclass classification problem can be solved using one-versus-one or one-versus-all strategies.

Ex:

- **In multi-class classification, we have more than two classes. Say, we have different features and characteristics of cars, trucks, buses and boats as input features. Our job is to predict the label (car, truck, bus or boat).**
- We will treat each class as a binary classification problem the way we solved a fraud or no-fraud problem. This approach is called the **one-versus-all** method.
- In the **one-versus-all** method, when we work with a class, that class is denoted by 1 and the rest of the classes becomes 0.
- For our four classes of vehicles: cars, trucks, buses and boats; when we work on the car, we will use the car as 1 and the rest of the classes as zeros. Again, when we work on the truck, the element of the truck will be one, and the rest of the classes will be zeros.

Car	Truck	Bus	Boat
1	0	0	0
0	1	0	0
0	0	1	0
0	0	0	1

One-Versus-All:

Car vs (Truck, Bus, Boat)

Truck vs (Car, Bus, Boat)

Bus vs (Car, Truck, Boat)

Boat vs (Car, Truck, Bus)

One vs One:

Car vs Truck

Car vs Bus

Car vs Boat

Truck vs Bus

Truck vs Boat

Bus vs Boat

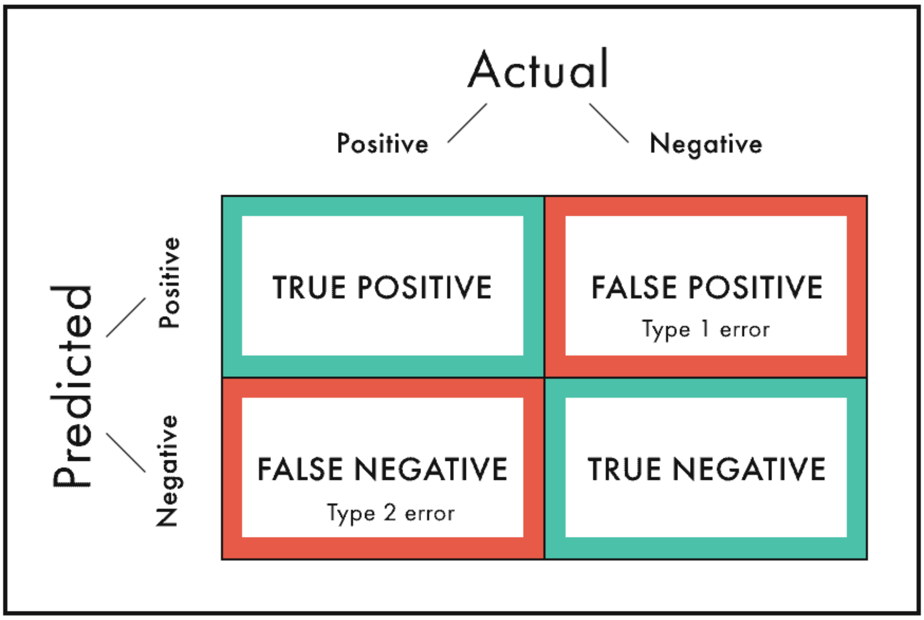
Note: Students to write related diagrams based on SVM, Neural Networks, Decision Tress in the similar way of binary classification diagrams but considering multi class problems.

Evaluating Predictive Models:

1. Performance measures for classification models:

- There are so many performance evaluation measures when it comes to selecting a classification model.
- They are
  - **Confusion Matrix**
  - **Accuracy**
  - **Precision**
  - **Recall/ Sensitivity**
  - **Specificity**
  - **F1-Score**
  - **AUC & ROC Curve**

- **Confusion Matrix:** The Confusion matrix is one of the most intuitive and easiest metrics used for finding the correctness and accuracy of the model. It is used for Classification problem where the output can be of two or more types of classes.
- Terms used in defining a confusion matrix are TP, TN, FP, and FN.



		Actual	
		Positive	Negative
Predicted	Positive	TRUE POSITIVE	FALSE POSITIVE Type 1 error
	Negative	FALSE NEGATIVE Type 2 error	TRUE NEGATIVE

Ex: Let's take an example of a patient who has gone to a doctor with certain symptoms. Since it's the season of Covid, let's assume that he went with fever, cough, throat ache, and cold. These are symptoms that can occur during any seasonal changes too. Hence, it is tricky for the doctor to do the right diagnosis.

True Positive (TP): Model says patient is having Covid when patient is actually having Covid.

Let's say the patient was actually suffering from Covid and on doing the required assessment, the doctor classified him as a Covid patient. This is called TP or True Positive. This is because the case is positive in real and at the same time the case was classified correctly. Now, the patient can be given appropriate treatment which means, the decision made by the doctor will have a positive effect on the patient and society.

True Negative (TN): Model says patient is not having Covid when patient is actually not having Covid.

Let's say the patient was not suffering from Covid and the doctor also gave him a clean chit. This is called TN or True Negative. This is because the case was actually negative and was also classified as negative which is the right thing to do. Now the patient will get treatment for his actual illness instead of taking Covid treatment.

False Positive (FP): Model says patient is having Covid when patient is actually is not Covid.

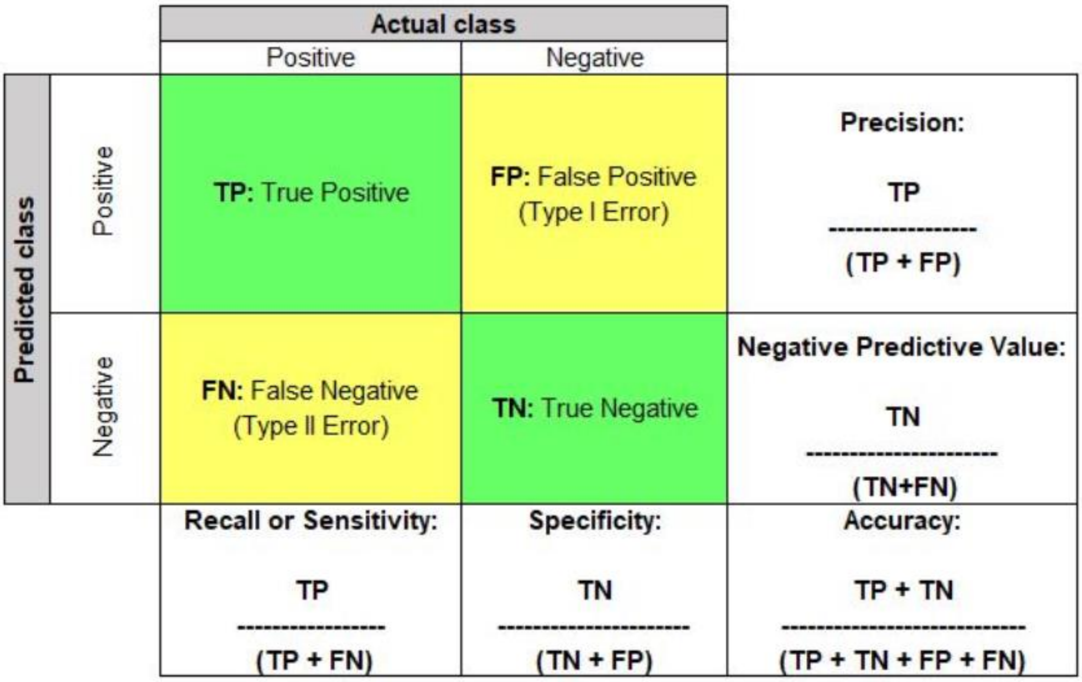
Let's say the patient was not suffering from Covid and he was only showing symptoms of seasonal flu but the doctor diagnosed him with Covid. This is called FP or False Positive. This is because the case was actually negative but was falsely classified as positive. Now, the patient will end up getting admitted to the hospital or home and will be given treatment for Covid. This is an unnecessary inconvenience for him and others as he will get unwanted treatment and quarantine. This is called Type I Error.

False Negative (FN): Model says patient is not having Covid when patient is actually having Covid.

Let's say the patient was suffering from Covid and the doctor did not diagnose him with Covid. This is called FN or False Negative as the case was actually positive but was falsely classified as negative. Now the patient will not get the right treatment and also he will spread the disease to others. This is a highly dangerous situation in this example. This is also called Type II Error.

Summary: In this particular example, **both FN and FP are dangerous** and the classification model which has the **lowest FN and FP values needs to be chosen** for implementation.

But in case there is a tie between few models which score very similar when it comes to FP and FN, in this scenario the **model with the least FN needs to be chosen**. This is because we simply cannot afford to have FNs! The goal of the hospital would be to not let even one patient go undiagnosed (no FNs) even if some patients get diagnosed wrongly (FPs) and asked to go under quarantine and special care.



		Actual class		
		Positive	Negative	
Predicted class	Positive	**TP:** True Positive	**FP:** False Positive (Type I Error)	**Precision:** **TP** ---------------- **(TP + FP)**
	Negative	**FN:** False Negative (Type II Error)	**TN:** True Negative	**Negative Predictive Value:** **TN** ---------------------- **(TN+FN)**
		Recall or Sensitivity: **TP** ---------------- **(TP + FN)**	**Specificity:** **TN** ---------------------- **(TN + FP)**	**Accuracy:** **TP + TN** ---------------------------- **(TP + TN + FP + FN)**

Confusion Matrix - Covid		Actual		**Total**
		Yes	No	
Predicted	Yes	100 (TP)	10 (FP)	**110**
	No	5 (FN)	50 (TN)	**55**
Total		**105**	**60**	**165**

- **Accuracy:** Accuracy is a good measure when the target variable classes in the data are nearly balanced.

Accuracy = Total Correctly Predicted/Total

=(TP + TN) / Total

= (100+50)/165=0.91

This term tells us how many right classifications were made out of all the classifications. In other words, how many TPs and TNs were done out of TP + TN + FP + FNs. It tells the ratio of “True”s to the sum of “True”s and “False”s.

Ex: Out of all the patients who visited the doctor, how many were correctly diagnosed as Covid positive and Covid negative.

- **Precision:**

Precision = Correctly Predicted/Total correct Predictions

= TP / (TP + FP)

= 100/(100+10)=0.91

Out of all that were marked as positive, how many are actually truly positive.

Ex: Let's take another example of a classification algorithm that marks emails as spam or not. Here, if emails that are of importance get marked as positive, then useful emails will end up going to the "Spam" folder, which is dangerous. Hence, the classification model which has the least FP value needs to be selected. In other words, a model that has the highest precision needs to be selected among all the models.

- **Recall or Sensitivity:**

Recall = Correctly Predicted / Total Actuals

=TP/ (TP + FN)

=100/(100+5)=0.95

Out of all the actual real positive cases, how many were identified as positive.

Ex: Out of all the actual Covid patients who visited the doctor, how many were actually diagnosed as Covid positive. Hence, the classification model which has the least FN value needs to be selected. In other words, a model that has the highest recall value needs to be selected among all the models.

- **Specificity:**

Specificity = TN/ (TN + FP)

= 50/(50+10)=0.83

Out of all the real negative cases, how many were identified as negative.

Use case: Out of all the non-Covid patients who visited the doctor, how many were diagnosed as non-Covid.

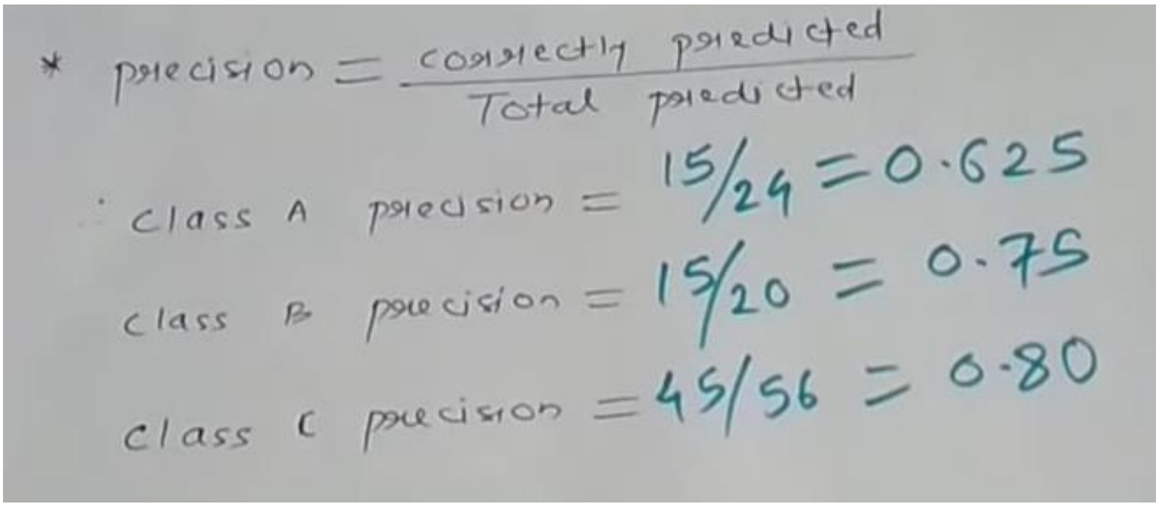
- **F1-Score:**

F1 score = 2* (Precision * Recall) / (Precision + Recall)

As we saw above, sometimes we need to give weightage to FP and sometimes to FN. F1 score is a weighted average of Precision and Recall, which means there is equal importance given to FP and FN. This is a very useful metric compared to "Accuracy". The problem with using accuracy is that if we have a highly imbalanced dataset for training (for example, a training dataset with 95% positive class and 5% negative class), the model will end up learning how to predict the positive class properly and will not learn how to identify the negative class. But the model will still have very high accuracy in the test dataset too as it will know how to identify the positives really well.

We have seen various evaluation methods for classification models. Based on the problem, we will use the appropriate metric. While accuracy is one of the widely used scores, in the case of imbalanced datasets, we will go for F1-score.

Confusion Matrix	Actual			Total
Predicted	15	7	2	24
	2	15	3	20
	3	8	45	56
Total	20	30	50	**100**

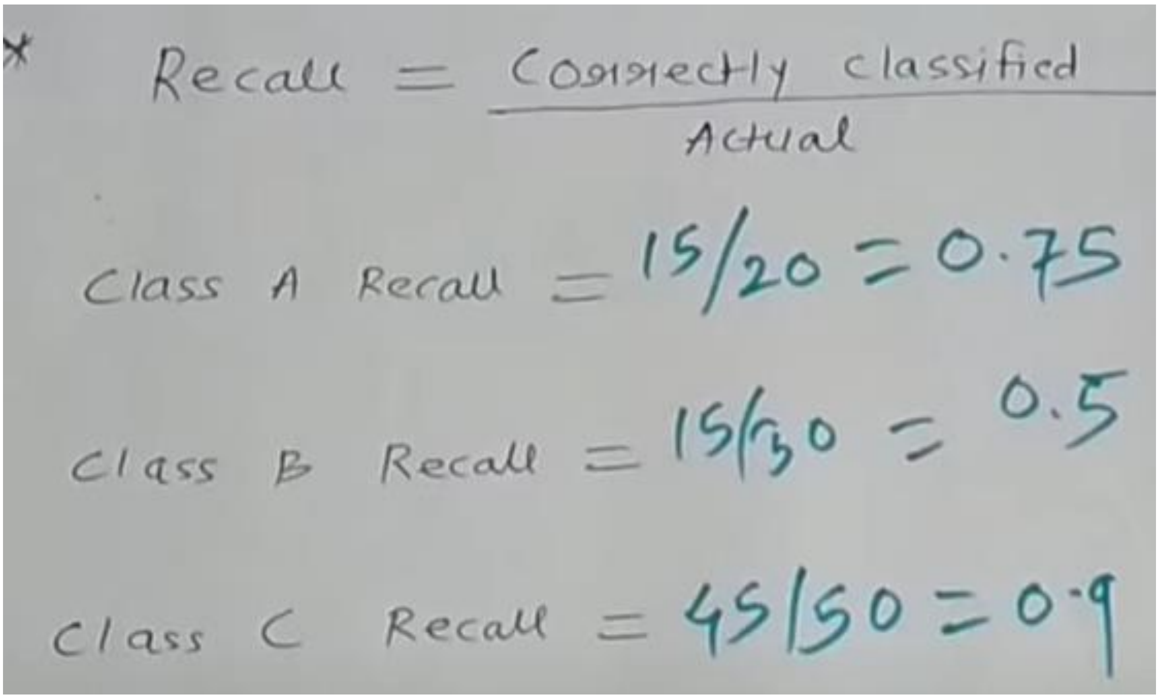


$$* \quad \text{precision} = \frac{\text{correctly predicted}}{\text{Total predicted}}$$

∴ Class A precision = $15/24 = 0.625$

Class B precision = $15/20 = 0.75$

Class C precision = $45/56 = 0.80$

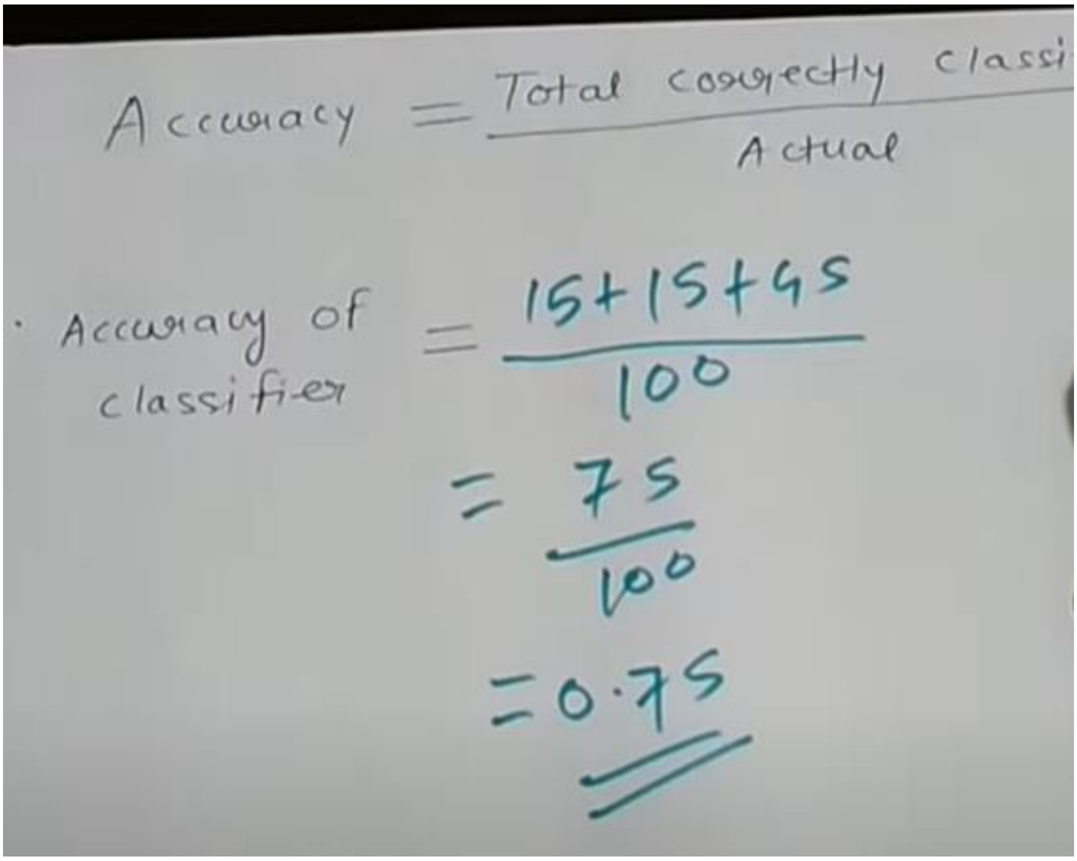


$$* \quad \text{Recall} = \frac{\text{Correctly classified}}{\text{Actual}}$$

Class A Recall = $15/20 = 0.75$

Class B Recall = $15/30 = 0.5$

Class C Recall = $45/50 = 0.9$

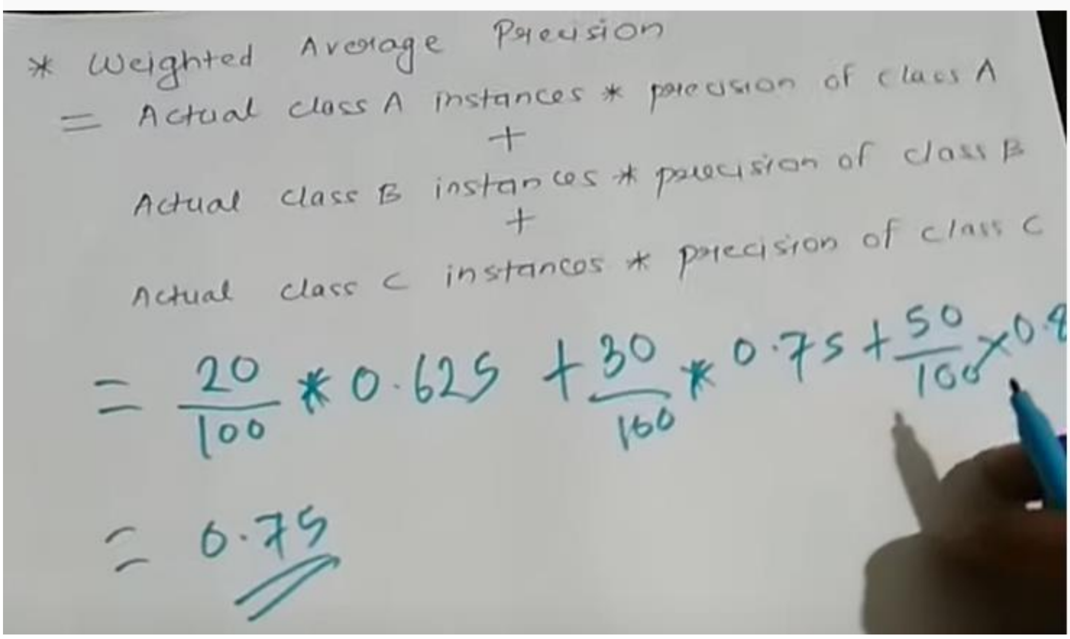


$$\text{Accuracy} = \frac{\text{Total correctly classi}}{\text{Actual}}$$

- Accuracy of classifier $= \dfrac{15+15+45}{100}$

$$= \frac{75}{100}$$

$$= \underline{\underline{0.75}}$$



* Weighted Average Precision

= Actual class A instances * precision of class A

+

Actual class B instances * precision of class B

+

Actual class C instances * precision of class C

$$= \frac{20}{100} * 0.625 + \frac{30}{100} * 0.75 + \frac{50}{100} \times 0.8$$

$$= \underline{\underline{0.75}}$$

2. Performance measures for regression models:

- Regression predictive modelling is the task of approximating a mapping function (f) from input variables (X) to a continuous output variable (y).

- Regression is different from classification, where classification involves in predicting a category or class label.

- Unlike classification, you cannot use classification accuracy to evaluate the predictions made by a regression model. Thus, it is the error component that describes if the regression model is correct or not.

- There are three error metrics that are commonly used for evaluating and reporting the performance of a regression model.

- They are:

  - **Mean Squared Error (MSE).**
  - **Root Mean Squared Error (RMSE)**
  - **Mean Absolute Error (MAE)**

Mean Squared Error (MSE):

- MSE is a most used and very simple metric with a little bit of change in mean absolute error. Mean squared error states that finding the squared difference between actual and predicted value.
- What actually the MSE represents? It represents the squared distance between actual and predicted values. we perform squared to avoid the cancellation of negative terms and it is the benefit of MSE.

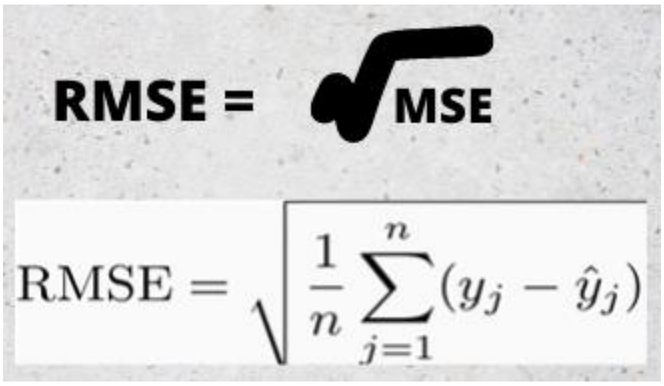
$$MSE = \frac{1}{n}\Sigma\underbrace{\left(y - \widehat{y}\right)^2}_{\text{The square of the difference between actual and predicted}}$$

- Advantages of MSE

  The graph of MSE is differentiable, so you can easily use it as a loss function.

Root Mean Squared Error(RMSE)

As RMSE is clear by the name itself, that it is a simple square root of mean squared error.



$$\text{RMSE} = \sqrt{\text{MSE}}$$

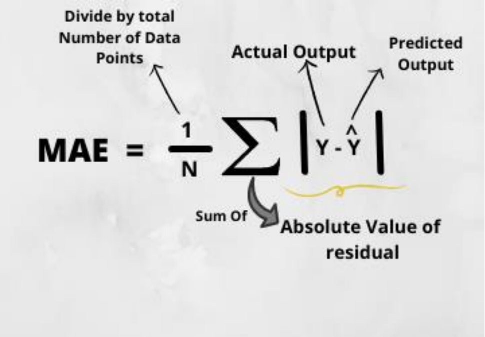
$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{j=1}^{n}(y_j - \hat{y}_j)}$$

- Advantages of RMSE:

  The output value you get is in the same unit as the required output variable which makes interpretation of loss easy.

Mean Absolute Error (MAE) is a very simple metric which calculates the absolute difference between actual and predicted values.

- Now you have to find the MAE of your model which is basically a mistake made by the model known as an error. Now find the difference between the actual value and predicted value that is an absolute error but we have to find the mean absolute of the complete dataset.



$$\text{MAE} = \frac{1}{N}\sum\left|Y - \hat{Y}\right|$$

Divide by total Number of Data Points
Actual Output
Predicted Output
Sum Of
Absolute Value of residual

- So, sum all the errors and divide them by a total number of observations and this is MAE. And we aim to get a minimum MAE because this is a loss.

- **Advantages of MAE:**
  - The MAE you get is in the same unit as the output variable.
  - It is most Robust to outliers.

Descriptive Analytics:

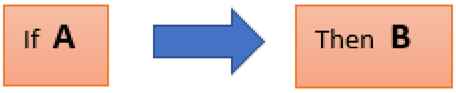
- Descriptive analytics is the process of using current and historical data to identify trends and relationships.
- Descriptive Analytics tells you what happened in the past. Diagnostic Analytics helps you understand why something happened in the past.
- It's sometimes called the simplest form of data analysis because it describes trends and relationships but doesn't dig deeper.
- In descriptive analytics, the aim is to describe patterns of customer behaviour.
- Contrary to Predictive analytics, there is no real target variable.
- Descriptive analytics is often referred to as unsupervised learning because there is no target variable to steer the learning process.
- **Ex Use Cases: Market Basket Analysis; Recommender Systems; Web Analytics; Traffic Analysis; Social media analysis; financial analysis; Customer behaviour analysis; Sales Analysis etc.**
- The three most common types of descriptive analytics are
    - **Association Rules**
    - **Sequence Rules**
    - **Segmentation**

Type of Descriptive Analytics	**Explanation**	**Example**
Association rules	Detect frequently occurring patterns between items	Detecting what products are frequently purchased together in a supermarket context. (Market Basket Analysis) Detecting what words frequently co-occur in a text document. (Co-occurrence Analysis) Detecting what elective courses are frequently chosen together in a university setting. (Course Selection Analysis)
Sequence rules	Detect sequences of events	Detecting sequences of purchase behaviour in a supermarket context (Customer Behaviour) Detecting sequences of web page visits in a web mining context (Web Analytics) Detecting sequences of words in a text document
Clustering/ Segmentation	Detect homogeneous segments of observations	Differentiate between brands in a marketing portfolio Segment customer population for targeted marketing

Association Rules:

- Association rules are used to find correlations and co-occurrences between data sets.
- They are ideally **used to explain patterns** in data from seemingly independent information repositories, such as relational databases and transactional databases.
- It is employed in Market Basket analysis, Web usage mining etc.
- Association rules typically start from a database of transactions.
- Each transaction consists of a transaction identifier and a set of items.
- Association rules are usually represented in the form X → Y, where X (also called rule Antecedent) and Y (also called Consequent) are disjoint item sets (i.e., disjoint conjunctions of features).
- An antecedent is an item (or item set) found in the data.
- A consequent is an item (or item set) that is found in combination with the antecedent.
- ***Ex: If a customer buys bread, he's 70% likely of buying milk."***
  In the above association rule, bread is the antecedent and milk is the consequent.

- Association rule shows how frequently a item set occurs in a transaction.
- Association rules are created by analysing data for frequent if/then patterns and using the criteria **support and confidence** to identify the most important relationships.
- Association rule learning works on the concept of *If and Else* Statement, such as ***if A then B.***



- Here the If element is called **antecedent**, and then statement is called as **Consequent**.
- Association rules are stochastic in nature which means they should not be interpreted as a universal truth. (Highly random). (Examples include the growth of a bacterial population, purchase behaviour of a customer.)

TID	Items
1	Bread, Milk
2	Bread, Apple, Chocolate, Biscuit
3	Milk, Apple, Chocolate, Coke
4	Bread, Milk, Apple, Chocolate
5	Bread, Milk, Apple, Coke

Support Count – Frequency of occurrence of an item set.

Consider ({Milk, Bread, Apple})=2

Association Rule – An implication expression of the form X -> Y, where X and Y are any 2 item sets.

Example: {Milk, Apple}->{Chocolate}

- To measure the associations between thousands of data items, there are several metrics. These metrics are given below:
  - **Support**
  - **Confidence**
  - **Lift**

Support:

Support describes how frequently if/then relationship appears in the dataset.

It is defined as the fraction of the transactions, T, that contains the item set X. If there are X datasets, then for transactions T, it can be written as:

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

S[{Milk, Apple}->{Chocolate}] = 2/5 = 0.4

Confidence:

Confidence indicates how often (number of times) 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.

It is the ratio of the transaction that contains X and Y to the number of records that contain X.

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

C[{Milk, Apple}->{Chocolate}] = 2/3=0.67

Lift:

It is the strength of any rule, which can be defined as below formula:

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

L[{Milk, Apple}->{Chocolate}] = (2/5)/[(3/5)*(3/5)]=0.4/(0.6*0.6)=1.11

Association Rule Learning/Mining:

Mining association rules from data is essentially a 2 – step process as shown below:

1. Identification of all item sets having support above minsup (i.e., frequent item sets)
2. Discovery of all derived association rules having confidence above minconf.

The step1 is performed by Apriori algorithm.

Apriori Algorithm:

This algorithm uses frequent datasets to generate association rules. It is designed to work on the databases that contain transactions. This algorithm uses a breadth-first search and Hash Tree to calculate the item set efficiently.

Apriori algorithm is a sequence of steps to be followed to find the most frequent itemset in the given database.

It is mainly used for market basket analysis and helps to understand the products that can be bought together. It can also be used in the healthcare field to find drug reactions for patients.

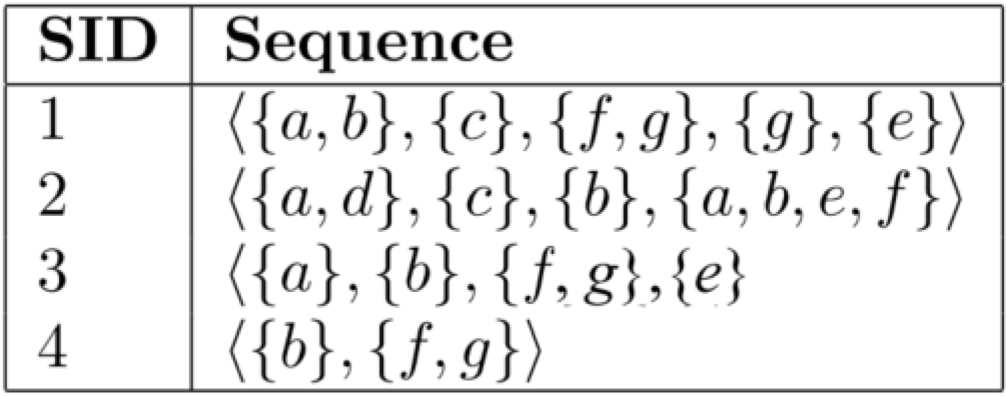
Applications of Association Rule Learning

It has various applications in machine learning and data mining. Below are some popular applications of association rule learning:

- **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.
- **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.
- **Protein Sequence:** The association rules help in determining the synthesis of artificial Proteins.

Sequence Rules:

- Association rule mining does not consider the order of transactions. However, in many applications such orderings are significant. For example, in market basket analysis, it is interesting to know whether people buy some items in sequence.
  Ex1: Buying bread first and then buying milk sometime later.
  Ex2: In natural language processing or text mining, considering the ordering of words in a sentence is vital in finding language or linguistic patterns.
- For such applications, as cited above, association rules are no longer appropriate. Sequential patterns are needed to process such scenarios/patterns.
- A **sequence** is an ordered list of sets of items. (Doesn't matter when they occur)
- A **sequential rule** is a rule of the form **X -> Y** where X and Y are sets of items (item sets).
- A rule X ->Y is interpreted as if items in X occurs (in any order), then it will be followed by the items in Y (in any order).
- An example could be a sequence of web page visits in a web analytics setting, say Amazon, as follows:
  Home page ⇒ Electronics ⇒ Mobile Phones ⇒ Samsung ⇒ 8GB RAM and above ⇒ Shopping cart ⇒ Order confirmation ⇒ Return to shopping
- **Ex:** Given a database D of customer transactions, the problem of mining sequential rules is to find the maximal sequences among all sequences that have certain user specified minimum support and confidence.



SID	Sequence
1	$\langle\{a,b\},\{c\},\{f,g\},\{g\},\{e\}\rangle$
2	$\langle\{a,d\},\{c\},\{b\},\{a,b,e,f\}\rangle$
3	$\langle\{a\},\{b\},\{f,g\},\{e\}$
4	$\langle\{b\},\{f,g\}\rangle$

- This database contains four sequences named Seq1, Seq2, Seq3 and Seq4. In our example, consider that the symbols **"a", "b", "c", d", "e", "f", "g" and "h"** respectively represent some **items** sold in a supermarket. For example, "a" could represent an "apple", "b" could be some "bread", etc.
- In this example, we will assume that each sequence represents what a customer has bought in the supermarket over time. For example, consider the second sequence "Seq2". This sequence indicates that the second customer bought items "a" and "d" together, then bought item "c", then bought "b", and then bought "a", "b", "e" and "f" together.
- Sequences are a very common type of data structures that can be found in many domains such as bioinformatics (DNA sequence), sequences of clicks on websites, the behaviour of learners in e-learning, sequences of what customers buy in retail stores, sentences of words in a text, etc.

Discovering sequential rules in sequences:

- A sequential rule is a rule of the form X -> Y where X and Y are sets of items (item sets). A rule X ->Y is interpreted as *if items in X occurs* (in any order), *then it will be followed by the items in Y* (in any order).
- For example, consider the rule {a} -> {e,f}. It means that if a customer buy item “a”, then the customer will later buy the items “e” and “f”. But the order among items in {e,f} is not important. This means that a customer may buy “e” before “f” or “f” before “e”.
- To find sequential rules, two measures are generally used: **the support and the confidence.** The support of a rule X -> Y is how many sequences contain the items from X followed by the items from Y. For example, the support of the rule {a} -> {e,f} is 3 sequences because {a} appears before the items from {e,f} in three sequences (Seq1, Seq2 and Seq3).
- The confidence of a rule X -> Y is the support of the rule divided by the number of sequences containing the items from X. For example, the confidence of the rule {a} -> {e,f} is 3/3=1, because every time that a customer buy item “a”, he then buy “e” and “f” in the example database. Another example is the rule {a} -> {b}. This rule has a support of 2 and a confidence of 2/3=0.67.

{a,b,c} -> {e} support = 2 sequences confidence = 2/2
{a} -> {c,e,f} support = 2 sequences confidence = 2/3
{a,b} -> {e,f} support = 3 sequences confidence = 3/3
{b} -> {e,f} support = 3 sequences confidence = 3/4
{a} -> {e,f} support = 3 sequences confidence = 3/3
{c} -> {e,f} support = 2 sequences confidence = 2/2
{a} -> {b} support = 2 sequences confidence = 2/3

- It is important to note that a transaction time or sequence field is included in the analysis.
- **Whereas association rules are concerned about what items appear together at the same time (intra-transaction patterns), sequence rules are concerned about what items appear at different times (inter-transaction patterns).**
- Consider the following example of a transactions data set in a web analytics setting. The letters A, B, C, ⋯ refer to web pages.

Session ID	Web Page	Sequence
1	A	1
1	B	2
1	C	3
2	B	1
2	C	2
3	A	1
3	C	2
3	D	3
4	A	1
4	B	2
4	D	3
5	D	1

5	C	2
5	A	3

A sequential version can then be obtained as follows:

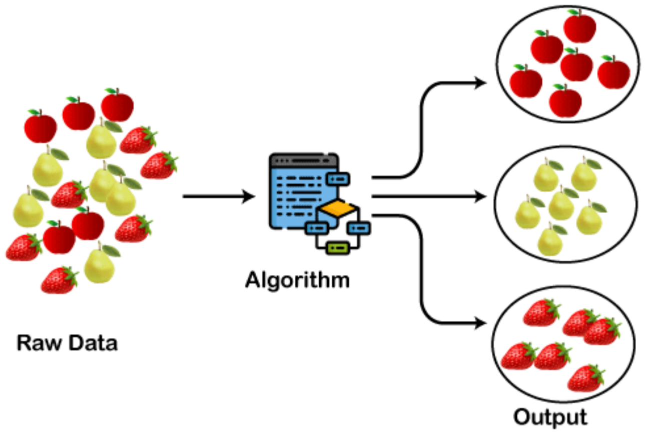
Session 1:	A, B, C
Session 2:	B, C
Session 3:	A, C, D
Session 4:	A, B, D
Session 5:	D, C, A

Applications of sequential rule mining:

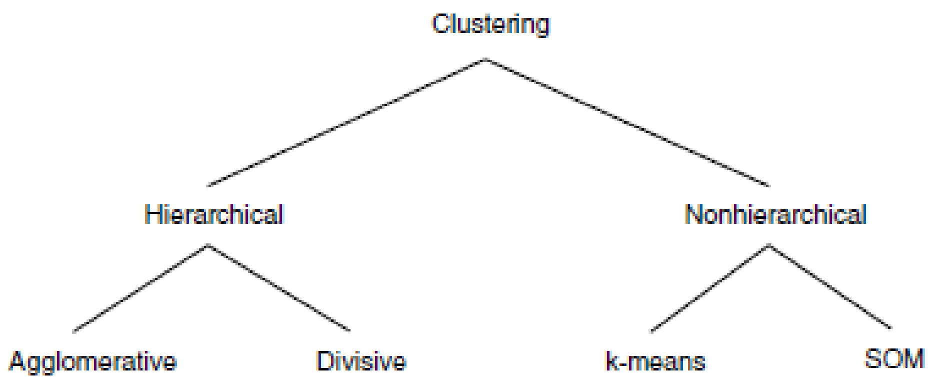
E-learning
Quality control
Web page prefetching
Food Order sequence in Restaurant
Fluctuation of the stock market
DNA sequence

Segmentation/Clustering:

- Segmentation is the technique of splitting customers into separate groups depending on their attributes or behaviour.
- The aim of segmentation is to split up a set of customer observations into segments such that the homogeneity within a segment is maximized (cohesive) and the heterogeneity between segments is maximized (separated).
- Segmenting is the process of putting customers into **groups based on similarities** and clustering is the process of **finding similarities** in customers so that they can be grouped, and therefore segmented.
- *Clustering is a process of grouping the data points into different clusters, consisting of similar data points. The objects with the possible similarities remain in a group that has less or no similarities with another group.*
- It does it by finding some similar patterns in the unlabelled dataset such as shape, size, colour, behaviour, etc., and divides them as per the presence and absence of those similar patterns.
- The clustering technique can be widely used in various tasks. Some most common uses of this technique are:
    - Market Segmentation
    - Statistical data analysis
    - Social network analysis
    - Image segmentation
    - Anomaly detection, etc.



- Clustering techniques can be categorized as either hierarchical or non-hierarchical.



- Hierarchical clustering, also known as hierarchical cluster analysis (HCA), is an algorithm that groups similar objects into groups called clusters.
- Hierarchical clustering begins by treating every data point as a separate cluster.
- Hierarchical clustering involves creating clusters that have a predetermined ordering, top-down or bottom-up. For example, all files and folders on the hard disk are organized in a hierarchy.
- There are two types of hierarchical clustering, **divisive (top-down)** and **agglomerative (bottom-up).**

Agglomerative hierarchical clustering is a Bottom-up algorithm that treat each data as a singleton cluster and then successively agglomerates pairs of clusters until all clusters have been merged into a single cluster that contains all data.

The algorithm for **Agglomerative Hierarchical Clustering** is:

- Calculate the similarity of one cluster with all the other clusters (calculate proximity matrix).
- Consider every data point as an individual cluster.
- Merge the clusters which are highly similar or close to each other.
- Recalculate the proximity matrix for each cluster.
- Repeat Steps 3 and 4 until only a single cluster remains.

Let's say we have six data points A, B, C, D, E, and F.

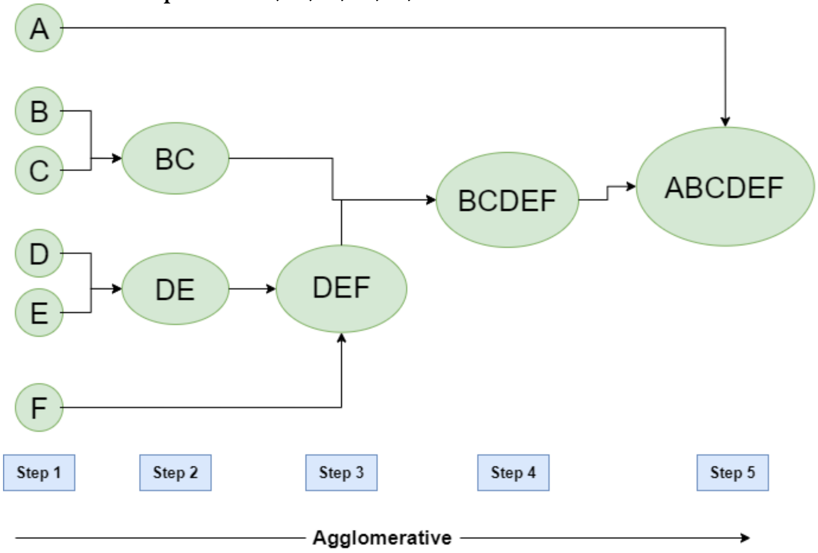


Figure – Agglomerative Hierarchical clustering

Divisive Hierarchical clustering is precisely the opposite of the Agglomerative Hierarchical clustering.

- In Divisive Hierarchical clustering, we take into account all of the data points as a single cluster and in every iteration, we separate the data points from the clusters which aren't comparable. In the end, we are left with N clusters.
- Top-down clustering requires a method for splitting a cluster that contains the whole data and proceeds by splitting clusters recursively until individual data have been split into singleton clusters.

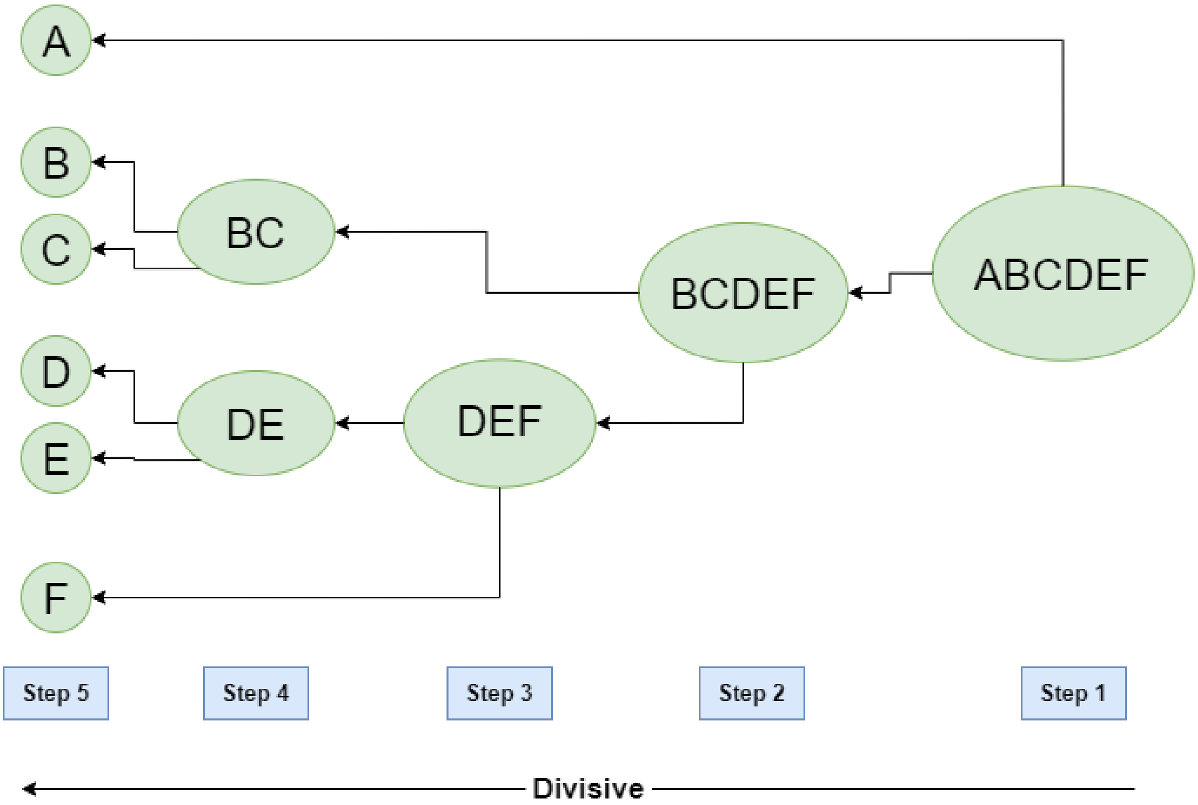
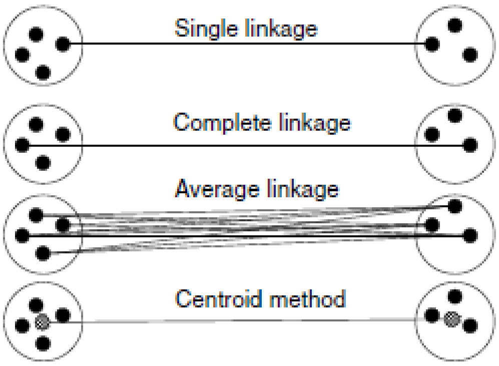


Figure – Divisive Hierarchical clustering

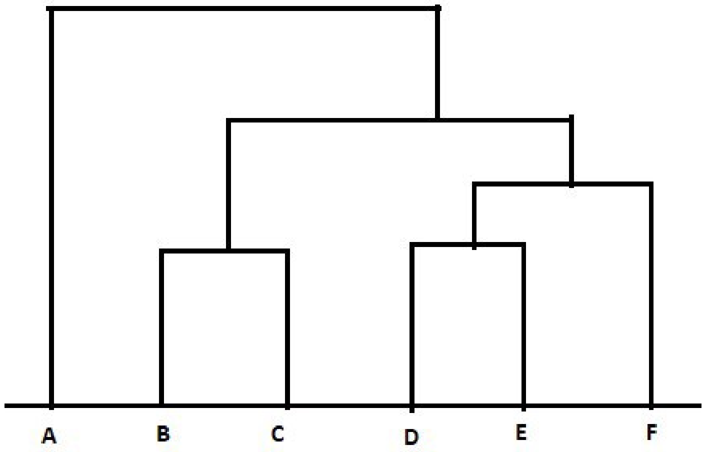
Scheme for Merger or Splitting:

- It is required to know the distance between 2 clusters to decide on split or merger.
- The schemes that are used for calculating distance include single linkage, complete linkage, centroid method etc.



Calculation of optimal number of clusters:

In order to decide on the optimal number clusters that can be used in hierarchical clustering, we can use dendrogram. (A tree like diagram that records sequences of merges).



Advantages of Hierarchical clustering

- It is simple to implement.
- It is easy and results in a hierarchy, a structure that contains more information.
- It does not need us to pre-specify the number of clusters.

Disadvantages of hierarchical clustering

- It breaks the large clusters.
- It is difficult to handle different sized clusters.
- It is sensitive to noise and outliers.

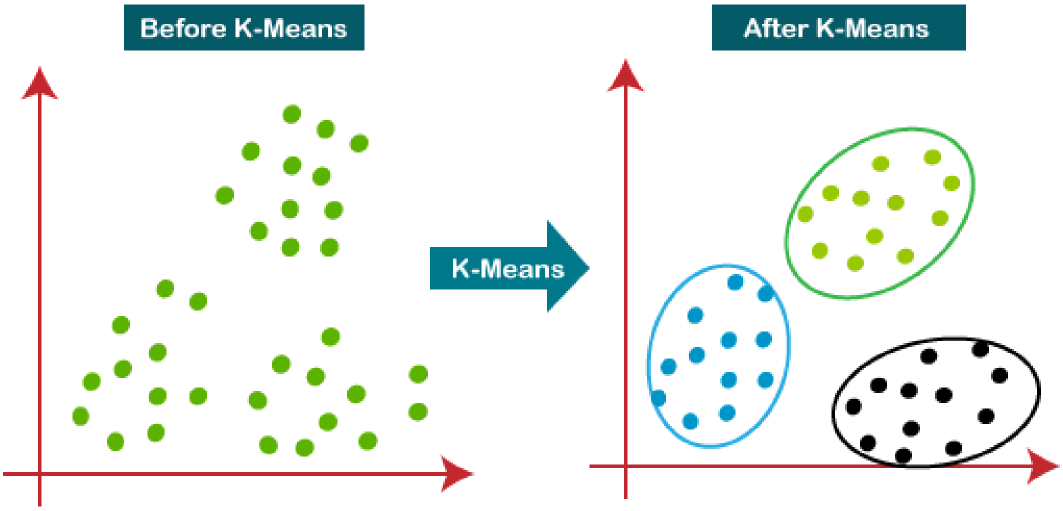
K-Means Clustering Algorithm:

- K-Means Clustering is non-hierarchical clustering algorithm that is used to solve the clustering problems in machine learning or data science.
- K-Means Clustering groups the unlabelled datasets into different clusters. Here K defines the number of pre-defined clusters that need to be created in the process. Ex: If K=2, there will be two clusters and if K=3, there will be three clusters, and so on.
- K-Means Clustering is an iterative algorithm that divides the unlabelled dataset into K different clusters in such a way that each dataset belongs to only one group that has similar properties.
- The algorithm takes the unlabelled datasets as input, divides the dataset into K-number of clusters, and repeats the process until it does not find the best clusters. The value of k should be predetermined in this algorithm.

The working of the K-Means algorithm is as below:

Step-1: Select the number K to decide the number of clusters.
Step-2: Select random K points or centroids. (It can be other from the input dataset).
Step-3: Assign each data point to their closest centroid, which will form the predefined K cluster

Step-4: Calculate the variance and place a new centroid of each cluster.
Step-5: Repeat the third steps, which means reassign each datapoint to the new closest centroid of each cluster.
Step-6: If any reassignment occurs, then go to step-4 else go to FINISH.
Step-7: The model is ready.



Applications of Clustering:

Below are some commonly known applications of clustering technique in Machine Learning:

- **In Identification of Cancer Cells:** The clustering algorithms are widely used for the identification of cancerous cells. It divides the cancerous and non-cancerous data sets into different groups.
- **In Search Engines:** Search engines also work on the clustering technique. The search result appears based on the closest object to the search query. It does it by grouping similar data objects in one group that is far from the other dissimilar objects. The accurate result of a query depends on the quality of the clustering algorithm used.
- **Customer Segmentation:** It is used in market research to segment the customers based on their choice and preferences.
- **In Biology:** It is used in the biology stream to classify different species of plants and animals using the image recognition technique.
- **In Land Use:** The clustering technique is used in identifying the area of similar lands use in the GIS database. This can be very useful to find that for what purpose the particular land should be used, that means for which purpose it is more suitable.

Difference between Hierarchical Clustering and Non Hierarchical Clustering:

Hierarchical Clustering:	Non Hierarchical Clustering:
Hierarchical Clustering involves creating clusters in a predefined order from top to bottom or bottom to top.	Non Hierarchical Clustering involves formation of new clusters by merging or splitting the clusters instead of following a hierarchical order.
It is considered less reliable than Non Hierarchical Clustering.	It is comparatively more reliable than Hierarchical Clustering.
It is considered slower than Non Hierarchical Clustering.	It is comparatively more faster than Hierarchical Clustering.
It is very problematic to apply this technique when we have data with high level of error.	It can work better then Hierarchical clustering even when error is there.
It is comparatively easier to read and understand.	The clusters are difficult to read and understand as compared to Hierarchical clustering.
It is relatively unstable than Non Hierarchical clustering.	It is a relatively stable technique.
Ex: Agglomerative, Divisive	Ex: K-Means, SOM