Regression Analysis:

Regression analysis is a simple supervised learning technique used to find the best trendline to describe a dataset. Regression Analysis is predictive modelling technique which investigates relationship between two or more variables.

Cause -------> Effect

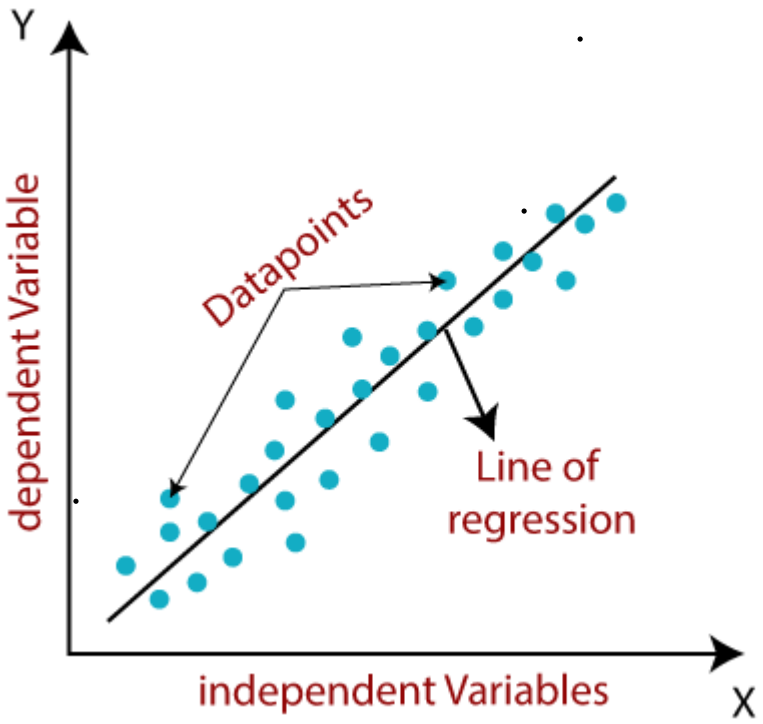
Linear Regression:

Linear regression is a predictive model used for finding the causal relationship between dependent variable and one or more independent variables

Process:

- Get sample Data
- Design a model
- Make a Prediction about the whole population

Geometric Representaion of Linear Regression Model



OLS -Ordinary Least Square is the most common method to estimate the linear regression. Least squares determine minimum Sum of Squared Errors -Lower the errors better the regression line.

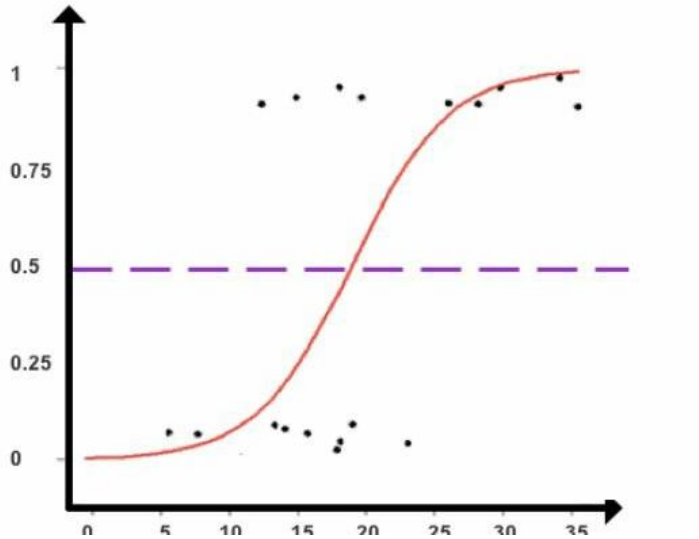
R-Squared values is a statistical measure to check how close the data are to the fitted regression line

Logistic Regression:

Logistic Regression produces results in a binary format which is used to predict the outcome of a categorical dependent variable.

The sigmoid function produces an S-shaped curve that can convert any number and map it into a numerical value between 0 and 1, but it does so without ever reaching those exact limits

A common application of the sigmoid function is found in logistic regression. Logistic regression adopts the sigmoid function to analyse data and predict discrete classes that exist in a dataset. Although logistic regression shares a visual resemblance to linear regression, it is technically a **classification technique**



Applications:

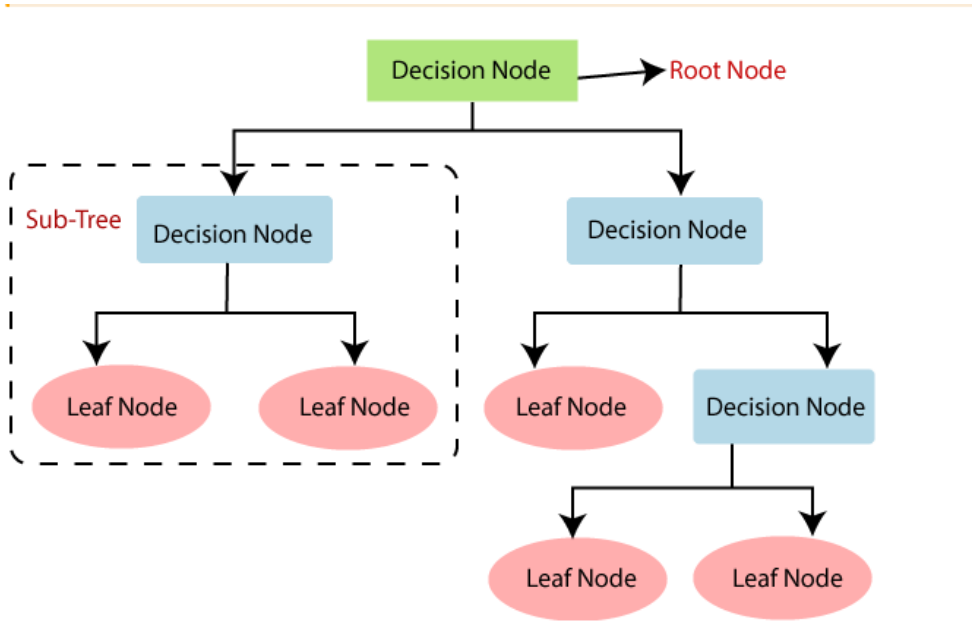
- fraud detection
- disease diagnosis
- emergency detection
- loan default detection
- spam and non-spam

Logistic regression with more than two outcome values is known as multinominal logistic regression

Decision Tree:

A decision tree is a graphical representation of all the possible solutions to a decision based on certain conditions. As a supervised learning technique, decision trees are used primarily for solving classification problems, but they can be applied to solve regression problems too.

Decision trees start with a root node, which acts as a starting point (at the top) and is followed by splits that produce branches. The statistical/mathematical term for these branches is edges. The branches then link to leaves, known also as nodes, which form decision points. A final categorization is produced when a leaf does not generate any new branches and results in what is known as a terminal node. Decision trees thus not only break down and explain how classification or regression is formulated, but they also produce a neat visual flowchart you can show to others. The ease of interpretation is a strong advantage of using decision trees, and they can be applied to a wide range of use cases.



Applications:

- picking a scholarship recipient
- assessing an applicant for a home loan
- predicting e-commerce sales
- selecting the right job applicant

Decision trees are built by first splitting data into two groups. This binary splitting process is then repeated at each branch (layer). The aim is to select a binary question that best splits the data into two homogenous groups at each branch of the tree, such that it minimizes the level of data entropy at the next.

Entropy is a mathematical term that explains the measure of variance in the data among different classes.

Information Gain measures reduction in entropy and decides which attribute is selected as decision node

Algorithms:

ID3, as an "Iterative Dichotomizer," is for binary classification only

ID3 applies entropy to determine which binary question to ask at each layer of the decision tree. At each layer, ID3 identifies a variable (converted into a binary question) that will produce the least entropy at the next layer

ID3, selects its splits based on Information Gain, which is the reduction in entropy between the parent node and (weighted sum of) children nodes

CART, or "Classification And Regression Trees," is a family of algorithms (including, but not limited to, binary classification tree learning).

CART, when used for classification, selects its splits to achieve the subsets that minimize Gini impurity

Random Forest:

It is an alternative technique to decision trees, by constructing multiple trees and combine their predictions to select an optimal path of classification or prediction. This involves a randomized selection of binary questions to grow multiple different decision trees, known as random forests. This is also known as “bootstrap aggregating” or “bagging.”

Bagging is the ensemble learning method that is commonly used to reduce variance within a noisy dataset. In bagging, a random sample of data in a training set is selected with replacement—meaning that the individual data points can be chosen more than once. After several data samples are generated, these weak models are then trained independently, and depending on the type of task—regression or classification, for example—the average or majority of those predictions yield a more accurate estimate.

Boosting Another variant of multiple decision trees is the popular technique of boosting, which are a family of algorithms that convert “weak learners” to “strong learners.” The underlying principle of boosting is to add weights to iterations that were misclassified in earlier rounds.

A popular boosting algorithm is gradient boosting. Rather than selecting combinations of binary questions at random (like random forests), gradient boosting selects binary questions that improve prediction accuracy for each new tree. Decision trees are therefore grown sequentially, as each tree is created using information derived from the previous decision tree.

One drawback of using random forests and gradient boosting is that we return to a black-box technique and sacrifice the visual simplicity and ease of interpretation that comes with a single decision tree

Applications:

- IT: network intrusion detection systems
- Healthcare: predict the onset of diabetes based on various risk predictors.
- Environment: Remote sensing, used to map the types of wetlands within a coastal landscape.

- Finance: automating critical tasks, including fraud detection, credit risk evaluations, and option pricing problem

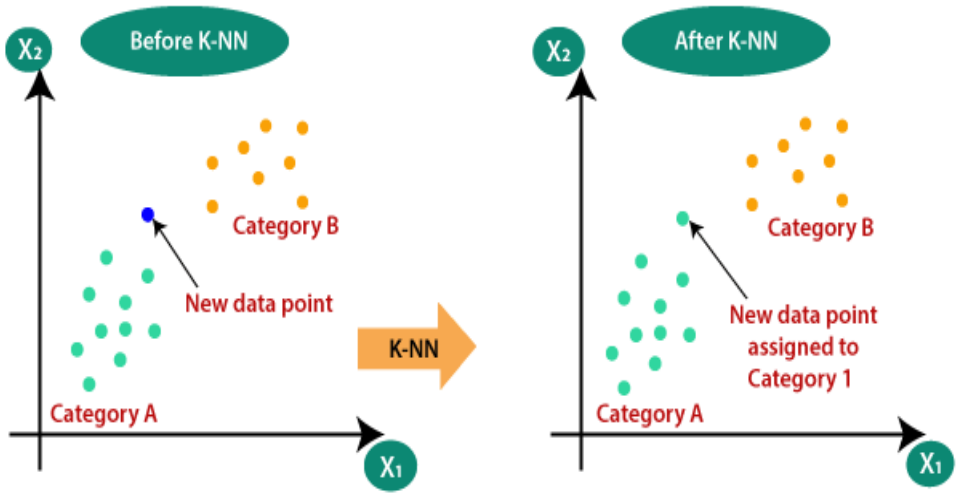
KNN (K- Nearest Neighbours):

K Nearest Neighbour is an instance-based Supervised Machine learning algorithm

- K-Nearest Neighbors is one of the simplest supervised machine learning algorithms used for classification. It classifies a data point based on its neighbors' classifications. It stores all available cases and classifies new cases based on similar features.

The KNN algorithm is used in the following scenarios:

Data is labelled, Data is noise-free, Dataset is small, as KNN is a lazy learner



Pros and Cons of Using KNN

Pros: Since the KNN algorithm requires no training before making predictions, new data can be added seamlessly, which will not impact the accuracy of the algorithm. KNN is very easy to implement. There are only two parameters required to implement KNN—the value of K and the distance function (e.g., Euclidean, Manhattan, etc.)

Cons: The KNN algorithm does not work well with large datasets. The cost of calculating the distance between the new point and each existing point is huge, which degrades performance. Feature scaling (standardization and normalization) is required before applying the KNN algorithm to any dataset. Otherwise, KNN may generate wrong predictions.

Applications:

In the real world, the KNN algorithm has applications for both classification and regression problems.

Healthcare: Healthcare companies use the KNN algorithm to determine if a patient is susceptible to certain diseases and conditions.

Financial services: Financial institutions predict credit card ratings or qualify loan applications and the likelihood of default with the help of the KNN algorithm.

Modelling legal reasoning and for guiding searches in complex manufacturing and transportation planning problems.

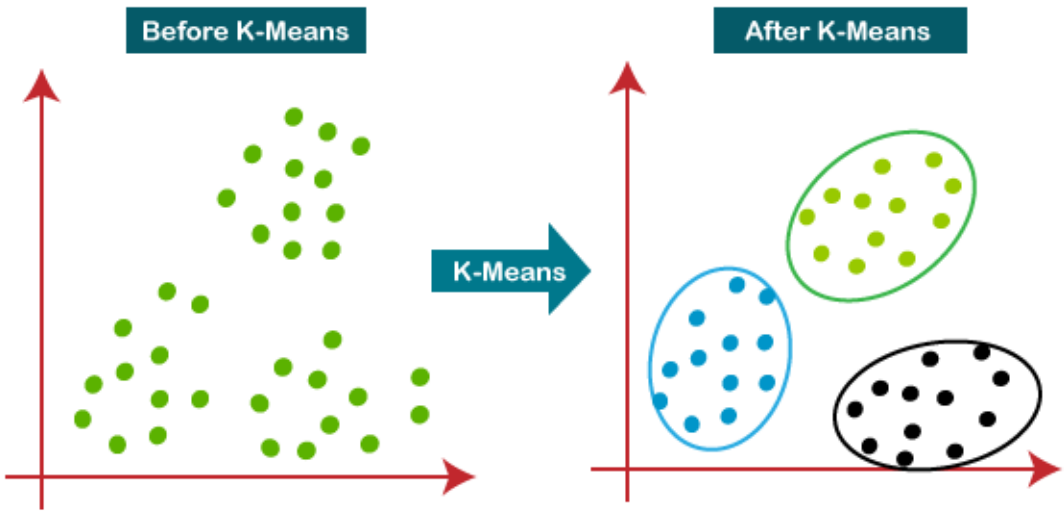
K-Means:

K – means is unsupervised machine learning algorithm

K-means algorithm is an iterative algorithm that tries to partition the dataset into K-pre-defined distinct non-overlapping subgroups (clusters) where each data point belongs to only one group. It tries to make the intra-cluster data points as similar as possible while also keeping the clusters as different (far) as possible. It assigns data points to a cluster such that the sum of the squared distance between the data points and the cluster's centroid (arithmetic mean of all the data points that belong to that cluster) is at the minimum. The less variation we have within clusters, the more homogeneous (similar) the data points are within the same cluster.

The way k-means algorithm works is as follows:

- Specify number of clusters K.
- Initialize centroids by first shuffling the dataset and then randomly selecting K data points for the centroids without replacement.
- Keep iterating until there is no change to the centroids. i.e., assignment of data points to clusters isn't changing.
- Compute the sum of the squared distance between data points and all centroids.
- Assign each data point to the closest cluster (centroid).
- Compute the centroids for the clusters by taking the average of all data points that belong to each cluster.



Applications:

- Market segmentation
- Document clustering
- Image segmentation and Image Compression
- Academic performances
- Search Engines

Naïve Bayes:

Naive Bayes is a supervised learning algorithm that is based on applying Bayes' theorem with the "naive" assumption. The Bayes Rule gives the formula for the probability of Y given X. It is called Naive because of the naive assumption that the X's are independent of each other.

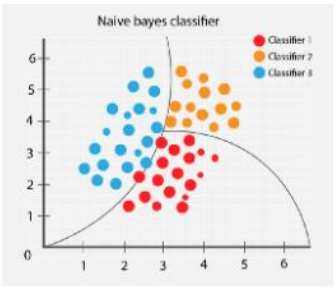
Types of Naïve Bayes Classifier:

- Multinominal Naïve Bayes
- Bernoulli Naïve Bayes
- Gaussian Naïve Bayes

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

using Bayesian probability terminology, the above equation can be written as

$$\text{Posterior} = \frac{\text{prior x likelihood}}{\text{evidence}}$$



Source: https://thatware.co/naive-bayes/

Applications:

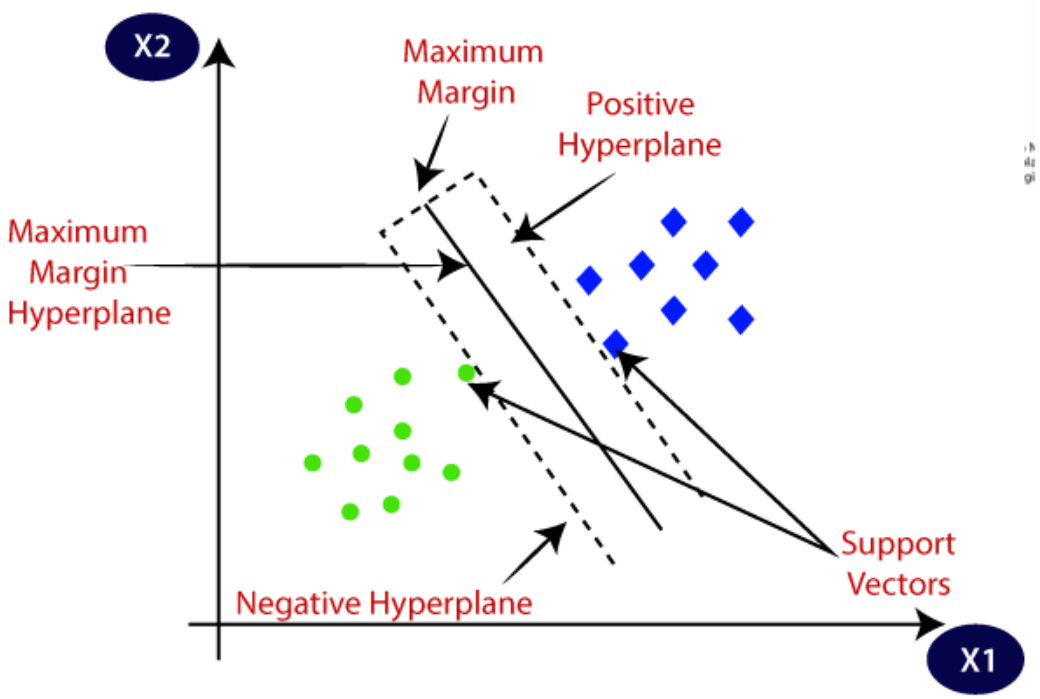
- Sentiment Analysis
- Spam Filtering
- Recommendation Systems

SVM (Support Vector Machine):

Support Vector Machines are powerful supervised learning algorithms for both classification and regression. It is a discriminative classifier that is formally defined by a separating hyperplane. So given labelled training data, the algorithm outputs an optimal hyperplane that categorizes new examples

The goal of the SVM algorithm is to create the best line or decision boundary that can segregate n-dimensional space into classes so that we can easily put the new data point in the correct category in the future. This best decision boundary is called a hyperplane. 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.



Applications:

- Face detection
- Image classification
- Text categorization