Machine Learning Notes

- Krishna Shinde

Data Science: If is a process of extracting meaningful information/knowledge, insight from data by using scientific methods/resources.

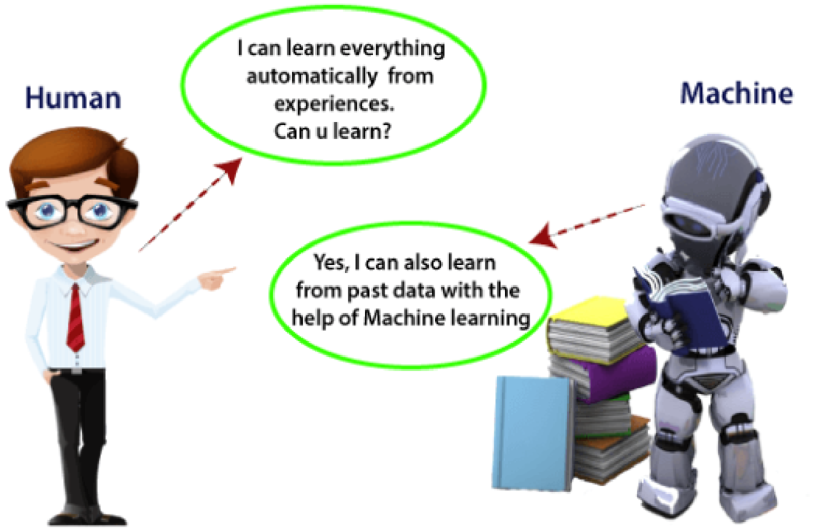
Scientific Methods/Resources:

1. Machine Learning(Data sets in the form of CSV, Excel, MongoDB, SQLite etc)
2. Deep Learning(Data sets in the form of images)
3. Natural Language Processing(NLP)(Data sets in the form of text)
4. Statistics
5. Data Visualization(Seaborn, Matplotlib, pandas, Autovis, plotly)

Types of Data:

1. **Structured data**: Data stored in Excel, CSV etc formats.
2. **Semi-structured data**: Data stored in JSON, HTML etc formats.
3. **Unstructured Data**: Data stored in image, videos, text, audio's, pdf etc formats.

Machine Learning: In the real world, we are surrounded by humans who can learn everything from their experiences with their learning capability, and we have computers or machines which work on our instructions. But can a machine also learn from experiences or past data like a human does? So here comes the role of Machine Learning.


Human
I can learn everything automatically from experiences. Can u learn?
Machine
Yes, I can also learn from past data with the help of Machine learning

Definition: Machine learning is a subset of Artificial intelligence(AI)/branch of computer science which deals with system programming in order to automate machine to learn from their past data/experiences and improve performance and make predictions.

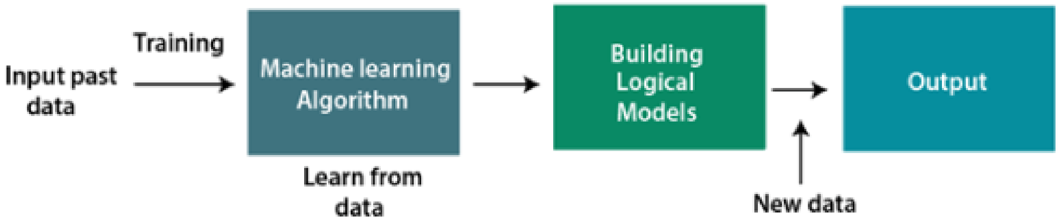
How does machine learning work?

A Machine Learning system learns from historical data, builds the prediction models, and whenever it receives new data, predicts the output for it. The accuracy of predicted output depends upon the amount of data, as the huge amount of data helps to build a better model which predicts the output more accurately.

Suppose we have a complex problem, where we need to perform some predictions, so instead of writing a code for it, we just need to feed the data to generic algorithms, and with the help of these algorithms, machine builds the logic as per the data and predict the output. Machine learning has changed our way of thinking about the problem. The below block diagram explains the working of Machine Learning algorithm:

Applications of Machine learning

We are using machine learning in our daily life even without knowing it



such as Google Maps, Google assistant, Alexa, etc. Below are some most trending real-world applications of Machine Learning:

1. Image Recognition:

It is used to identify objects, persons, places, digital images, etc. The popular use case of image recognition and face detection is, Automatic friend tagging suggestion:

Facebook provides us a feature of auto friend tagging suggestion. Whenever we upload a photo with our Facebook friends, then we automatically get a tagging suggestion with name, and the technology behind this is machine learning's face detection and recognition algorithm.

It is based on the Facebook project named "Deep Face," which is responsible for face recognition and person identification in the picture.

2. Speech Recognition:

While using Google, we get an option of "Search by voice," it comes under speech recognition, and it's a popular application of machine learning.

Speech recognition is a process of converting voice instructions into text, and it is also known as "Speech to text", or "Computer speech recognition." At present, machine learning algorithms are widely used by various applications of speech recognition. Google assistant, Siri, Cortana, and Alexa are using speech recognition technology to follow the voice instructions.

3. Traffic prediction:

If we want to visit a new place, we take help of Google Maps, which shows us the correct path with the shortest route and predicts the traffic conditions.

It predicts the traffic conditions such as whether traffic is cleared, slow-moving, or heavily congested with the help of two ways:

Real Time location of the vehicle form Google Map app and sensors

Average time has taken on past days at the same time. Everyone who is using Google Map is helping this app to make it better. It takes information from the user and sends back to its database to improve the performance.

4. Product recommendations:

Machine learning is widely used by various e-commerce and entertainment companies such as Amazon, Netflix, etc., for product recommendation to the user. Whenever we search for some product on Amazon, then we started getting an advertisement for the same product while internet surfing on the same browser and this is because of machine learning.

Google understands the user interest using various machine learning algorithms and suggests the product as per customer interest.

As similar, when we use Netflix, we find some recommendations for entertainment series, movies, etc., and this is also done with the help of machine learning.

5. Self-driving cars:

Machine learning plays a significant role in self-driving cars. Tesla, the most popular car manufacturing company is working on self-driving car. It is using unsupervised learning method to train the car models to detect people and objects while driving.

6. Email Spam and Malware Filtering:

Whenever we receive a new email, it is filtered automatically as important, normal, and spam. We always receive an important mail in our inbox with the important symbol and spam emails in our spam box, and the technology behind this is Machine learning. Below are some spam filters used by Gmail:

Content Filter
Header filter
General blacklists filter
Rules-based filters
Permission filters

Some machine learning algorithms such as Multi-Layer Perceptron, Decision tree, and Naïve Bayes classifier are used for email spam filtering and malware detection.

7. Virtual Personal Assistant:

We have various virtual personal assistants such as Google assistant, Alexa, Cortana, Siri. As the name suggests, they help us in finding the information using our voice instruction. These assistants can help us in various ways just by our voice instructions such as Play music, call someone, Open an email, Scheduling an appointment, etc.

These assistant record our voice instructions, send it over the server on a cloud, and decode it using ML algorithms and act accordingly.

8. Online Fraud Detection:

Machine learning is making our online transaction safe and secure by detecting fraud transaction. Whenever we perform some online transaction, there may be various ways that a fraudulent transaction can take place such as fake accounts, fake ids, and steal money in the middle of a transaction. So to detect this, Feed Forward Neural network helps us by checking whether it is a genuine transaction or a fraud transaction.

For each genuine transaction, the output is converted into some hash values, and these values become the input for the next round. For each genuine transaction, there is a specific pattern which gets change for the fraud transaction hence, it detects it and makes our online transactions more secure.

9. Stock Market trading:

In the stock market, there is always a risk of up and downs in shares, so for this machine learning's long short term memory neural network is used for the prediction of stock market trends.

10. Medical Diagnosis:

Medical technology is growing very fast and able to build 3D models that can predict the exact position of lesions in the brain.

It helps in finding brain tumors and other brain-related diseases easily.

11. Automatic Language Translation:

Nowadays, if we visit a new place and we are not aware of the language then it is not a problem at all, as for this also machine learning helps us by converting the text into our known languages. Google's GNMT (Google Neural Machine Translation) provide this feature, which is a Neural Machine Learning that translates the text into our familiar language, and it called as automatic translation.

The technology behind the automatic translation is a sequence to sequence learning algorithm, which is used with image recognition and translates the text from one language to another language.

12. Sentimental Analysis
13. House price prediction

Types of Machine Learning:

1. Supervised Machine Learning
2. Unsupervised Machine Learning
3. Reinforcement Machine Learning

1. Supervised Machine Learning:

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. The labelled data means some input data is already tagged with the correct output.

In supervised learning, the training data provided to the machines work as the supervisor that teaches the machines to predict the output correctly. It applies the same concept as a student learns in the supervision of the teacher.

Supervised learning is a process of providing input data as well as correct output data to the machine learning model. The aim of a supervised learning algorithm is to find a mapping function to map the input variable(x) with the output variable(y).

In the real-world, supervised learning can be used for Risk Assessment, Image classification, Fraud Detection, spam filtering, etc.

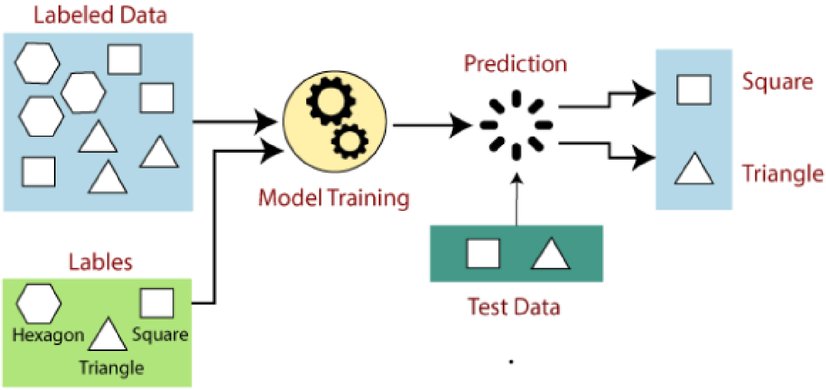
How Supervised Learning Works?

In supervised learning, models are trained using labelled dataset, where the model learns about each type of data. Once the training process is completed, the model is tested on the basis of test data (a subset of the training set), and then it predicts the output.

The working of Supervised learning can be easily understood by the below example and diagram:

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.

Steps Involved in Supervised Learning:

1. First Determine the type of training dataset.
2. Collect/Gather the labelled training data.
3. Split the training dataset into training dataset, test dataset, and validation dataset.
4. Determine the input features of the training dataset, which should have enough knowledge so that the model
5. can accurately predict the output.
6. Determine the suitable algorithm for the model, such as support vector machine, decision tree, etc.
7. Execute the algorithm on the training dataset. Sometimes we need validation sets as the control parameters, which are the subset of training datasets.

8. Evaluate the accuracy of the model by providing the test set. If the model predicts the correct output, which means our model is accurate.

Types of supervised Machine learning Algorithms:

Supervised learning can be further divided into two types of problems:

1. Regression
2. Classification

1. Regression:

Regression algorithms are used if there is a relationship between the input variable and the output variable. It is used for the prediction of continuous variables, such as Weather forecasting, Market Trends, etc. Below are some popular Regression algorithms which come under supervised learning:

1. Linear Regression
2. Regression Trees
3. Non-Linear Regression
4. Bayesian Linear Regression
5. Polynomial Regression

2. Classification:

Classification algorithms are used when the output variable is categorical, which means there are two classes such as Yes-No, Male-Female, True-false, etc.

1. Random Forest
2. Decision Trees
3. Logistic Regression
4. Support vector Machines

Advantages of Supervised learning:

1. With the help of supervised learning, the model can predict the output on the basis of prior experiences.
2. In supervised learning, we can have an exact idea about the classes of objects.
3. Supervised learning model helps us to solve various real-world problems such as fraud detection, spam filtering, etc.

Disadvantages of supervised learning:

1. Supervised learning models are not suitable for handling the complex tasks.
2. Supervised learning cannot predict the correct output if the test data is different from the training dataset.
3. Training required lots of computation times.
4. In supervised learning, we need enough knowledge about the classes of object.

2. Unsupervised Machine Learning:

In the previous topic, we learned supervised machine learning in which models are trained using labeled data under the supervision of training data. But there may be many cases in which we do not have labeled data and need to find the hidden patterns from the given dataset. So, to solve such types of cases in machine learning, we need unsupervised learning techniques.

What is Unsupervised Learning?

As the name suggests, unsupervised learning is a machine learning technique in which models are not supervised using training dataset. Instead, models itself find the hidden patterns and insights from the given data. It can be compared to learning which takes place in the human brain while learning new things. It can be defined as:

Unsupervised learning is a type of machine learning in which models are trained using unlabeled dataset and are allowed to act on that data without any supervision"

Unsupervised learning cannot be directly applied to a regression or classification problem because unlike supervised learning, we have the input data but no corresponding output data. The goal of unsupervised learning is to find the underlying structure of dataset, group that data according to similarities, and represent that dataset in a compressed format.

Example: Suppose the unsupervised learning algorithm is given an input dataset containing images of different types of cats and dogs. The algorithm is never trained upon the given dataset, which means it does not have any idea about the features of the dataset. The task of the unsupervised learning algorithm is to identify the image features on their own. Unsupervised learning algorithm will perform this task by clustering the image dataset into the groups according to similarities between images.

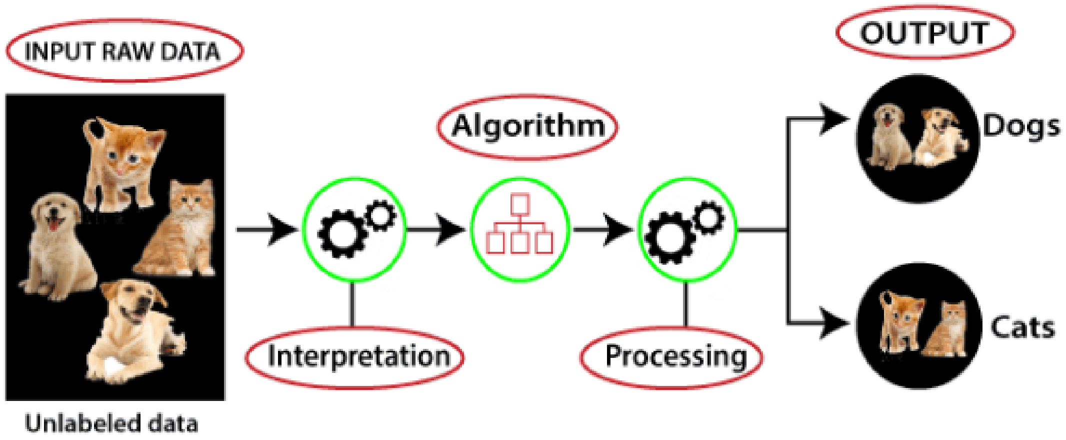
Why use Unsupervised Learning?
Below are some main reasons which describe the importance of Unsupervised Learning:

1. Unsupervised learning is helpful for finding useful insights from the data.
2. Unsupervised learning is much similar as a human learns to think by their own experiences, which makes it closer to the real AI.
3. Unsupervised learning works on unlabeled and uncategorized data which make unsupervised learning more important.
4. In real-world, we do not always have input data with the corresponding output so to solve such cases, we need unsupervised learning.

Working of Unsupervised Learning
Working of unsupervised learning can be understood by the below diagram:

Types of Unsupervised Learning Algorithm:
The unsupervised learning algorithm can be further categorized into two types of problems:

1. **Clustering:** Clustering is a method of grouping the objects into clusters such that objects with most similarities remains into a group and



has less or no similarities with the objects of another group. Cluster analysis finds the commonalities between the data objects and categorizes them as per the presence and absence of those commonalities.

2. **Association:** An association rule is an unsupervised learning method which is used for finding the relationships between variables in the large database. It determines the set of items that occurs together in the dataset. Association rule makes marketing strategy more effective. Such as people who buy X item (suppose a bread) are also tend to purchase Y (Butter/Jam) item. A typical example of Association rule is Market Basket Analysis.

Unsupervised Learning algorithms:
Below is the list of some popular unsupervised learning algorithms:

1. K-means clustering
2. KNN (k-nearest neighbors)
3. Hierarchal clustering
4. Anomaly detection
5. Neural Networks
6. Principle Component Analysis
7. Independent Component Analysis
8. Apriori algorithm

9. Singular value decomposition

Advantages of Unsupervised Learning

1. Unsupervised learning is used for more complex tasks as compared to supervised learning because, in unsupervised learning, we don't have labeled input data.
2. Unsupervised learning is preferable as it is easy to get unlabeled data in comparison to labeled data.

Disadvantages of Unsupervised Learning

1. Unsupervised learning is intrinsically more difficult than supervised learning as it does not have corresponding output.
2. The result of the unsupervised learning algorithm might be less accurate as input data is not labeled, and algorithms do not know the exact output in advance.

3. Rainforcement Machine Learning:

Reinforcement Learning is a feedback-based Machine learning technique in which an agent learns to behave in an environment by performing the actions and seeing the results of actions. For each good action, the agent gets positive feedback, and for each bad action, the agent gets negative feedback or penalty.

Data Science Project Steps:

1. Problem statement/Business statement
2. Data gathering
3. Exploratory data analysis(EDA)
4. Feature engineering
    (a) Imputation
    (b) Normalization

   (c) Standardization(Scaling)
   (d) Handling outliers
   (e) Encoding
   (f) Transformation

5. Feature selection

6. Model training(Building)

7. Model Evaluation
   Regression:

   (a) Mean squared error(MSE)
   (b) Root mean squared error(RMSE)
   (c) Mean absolute error(MAE)
   (d) R_2 score

   Classification:

   (a) Confusion matrix
   (b) Classification Report
   (c) Precision
   (d) Recall
   (e) Accuracy score
   (f) AUC-ROC curves

8. Model testing/Optimization/Improvement

   (a) Hyperparametric tunning

9. Web Development Framework

10. Project Development

Linear Regression:

It is a predictive model used to find linear relation ship between a dependent variable and one or more independent variables and predict continuous values.

Types of Linear Regression

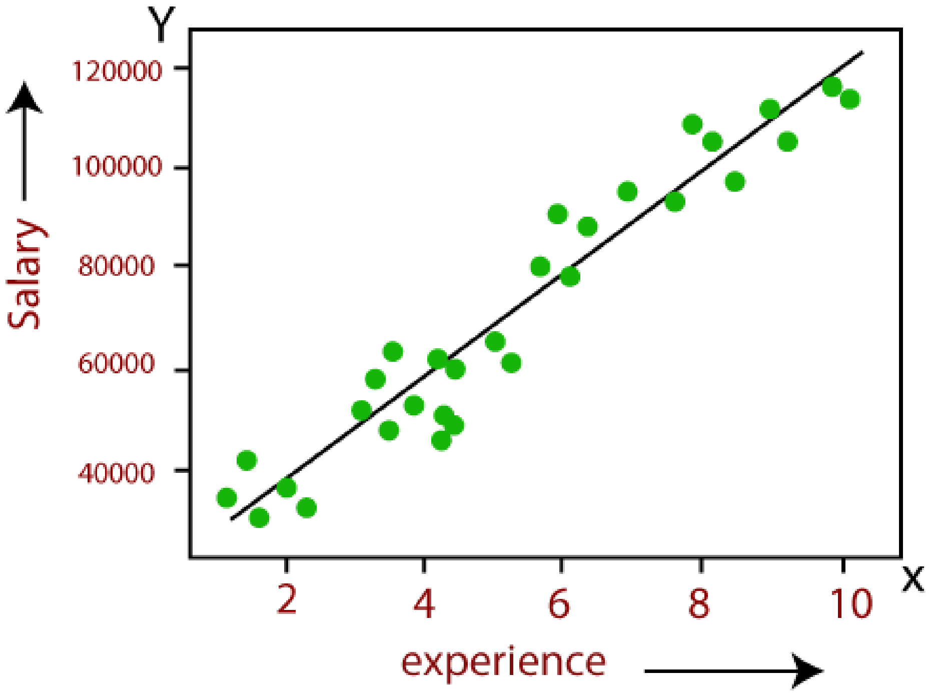
1. Simple linear regression
2. Multiple linear regression

If there is only one independent variable (x), then such linear regression is called simple **linear regression**. And if there are more than one independent variables, then such linear regression is called **multiple linear regression**.

The main target is to find a best fit line(BFL) which is also called as regression line.

The aim of linear regression is to predict values of Y based on independent variable.

The relationship between variables in the linear regression model can be explained using the below image. Here we are predicting the salary of an employee on the basis of the year of experience.



Assumptions of linear regression:

1. **Linear relationship between the features and target:** Linear regression assumes the linear relationship between the dependent and independent variables.

2. **Small or no multicollinearity between the features:** Multicollinearity means high-correlation between the independent variables. Due to multicollinearity, it may difficult to find the true relationship between the predictors and target variables. Or we can say, it is difficult to determine which predictor variable is affecting the target variable and which is not. So, the model assumes either little or no multicollinearity between the features or independent variables.

3. **Homoscedasticity:** The variance of residual is the same for any value of X.

4. **Normal distribution of error terms:** Linear regression assumes that the error term should follow the normal distribution pattern. If error terms are not normally distributed, then confidence intervals will become either too wide or too narrow, which may cause difficulties in finding coefficients. It can be checked using the q-q plot. If the plot shows a straight line without any deviation, which means the error is normally distributed.

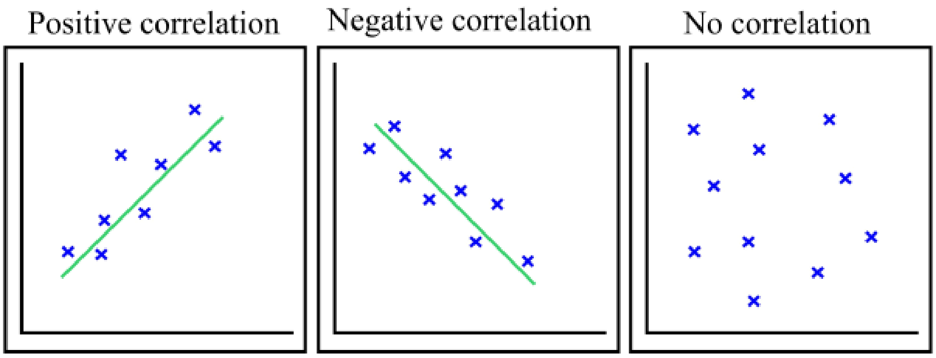
5. **No autocorrelations:** The linear regression model assumes no autocorrelation in error terms. If there will be any correlation in the error term, then it will drastically reduce the accuracy of the model. Autocorrelation usually occurs if there is a dependency between residual errors.

Correlation:

It indicates how closely the relation between two variables are. By default we use Pearson's correlation coefficient. It ranges from -1 to 1.

When X increase, Y also increase then there is positive relation and when X increase, and Y decrease then there is negative correlation. If correlation is close to -1 or $+1$ then it is called as good predictor(or there is linear relation between two variable).

When there is no linear relation between two variable or predict multiple values for Y corresponding to single value of X then it is called as bad predictor.



Coefficient of correlation($r-$value):
It is the indication how strong a relationship with variables. If the r value close to 1 or -1 (if r value lies between 0.7 to 1 or -0.7 to -1) then it is called there is good linear relation between two variables. If value of r is close to 0(if r value lies between -0.3 to 0 or 0 to 0.3) then there is no linear relation between two variables.
The formula to find the Pearson correlation coefficient is given below:

$$r = \frac{\sum\left(x_i-\bar{x}\right)\left(y_i-\bar{y}\right)}{\sqrt{\sum\left(x_i-\bar{x}\right)^2}\sqrt{\sum\left(y_i-\bar{y}\right)^2}}$$

Variance: Variance measures the variability of data points from central tendency(mean value). The formula to find the variance is given below:

$$\sigma^2 = \frac{1}{n-1}\sum\left(x_i-\bar{x}\right)^2$$

There are two types of variances stochastic and deterministic. The reason of occurrence of stochastic noise is unknown and the deterministic noise is occurred due to some reasons.
Central Tendancy: The statistical measure that identifies single value as representation of entire distribution. It aims to provide an accurate description of entire data. Examples: mean, median, mode.
Covarience: Covariance between two variables measure the variance between two variables. In other word it measures how two features varying together/how they influence each other. The formula to find the covariance is given below:

$$\mathrm{Cov}(x,y) = \frac{\sum\left(x_i-\bar{x}\right)\left(y_i-\bar{y}\right)}{n}$$

Standard deviation: The formula to find the standard deviation is given below:

$$\sigma = \sqrt{\frac{1}{n-1}\sum (x_i - \bar{x})^2}$$

Gradient Descent Algorithm

Gradient Descent is known as one of the most commonly used optimization algorithms to train machine learning models by means of minimizing errors between actual and expected results. Further, gradient descent is also used to train Neural Networks.

In mathematical terminology, Optimization algorithm refers to the task of minimizing/maximizing an objective function f(x) parameterized by x. Similarly, in machine learning, optimization is the task of minimizing the cost function parameterized by the model's parameters. The main objective of gradient descent is to minimize the convex function using iteration of parameter updates. Once these machine learning models are optimized, these models can be used as powerful tools for Artificial Intelligence and various computer science applications.

Gradient Descent is defined as one of the most commonly used iterative optimization algorithms of machine learning to train the machine learning and deep learning models. It helps in finding the local minimum of a function.

The best way to define the local minimum or local maximum of a function using gradient descent is as follows:

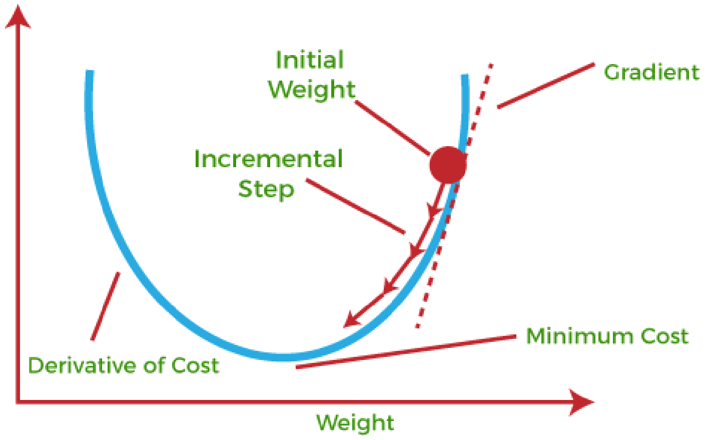
1. If we move towards a negative gradient or away from the gradient of the function at the current point, it will give the local minimum of that function.

2. Whenever we move towards a positive gradient or towards the gradient of the function at the current point, we will get the local maximum of that function.

   This entire procedure is known as Gradient Descent, which is also known as steepest descent. The main objective of using a gradient descent algorithm is to minimize the cost function using iteration. To achieve this goal, it performs two steps iteratively:

(a) Calculates the first-order derivative of the function to compute the gradient or slope of that function.

(b) Move away from the direction of the gradient, which means slope increased from the current point by alpha times, where Alpha is defined as Learning Rate. It is a tuning parameter in the optimization process which helps to decide the length of the steps.

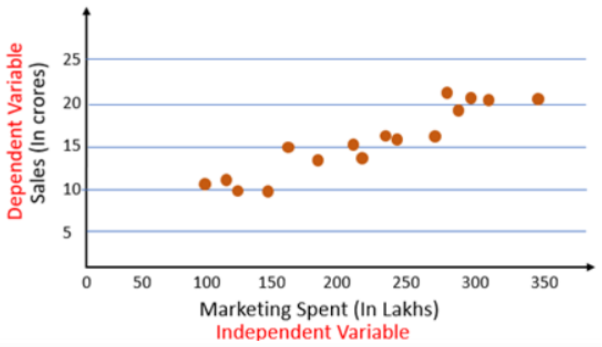


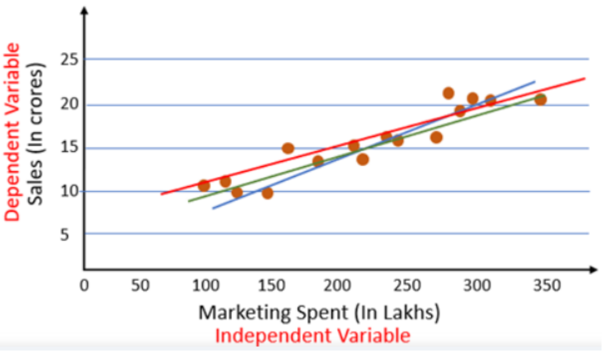
Gradient Descent Algorithm: A linear regression model attempts to explain the relationship between a dependent (output variables) variable and one or more independent (predictor variable) variables using a straight line.This straight line is represented using the following formula: $y = mx + c$, where m represents slope of the line and c represents y-intercept.

The first step in finding a linear regression equation is to determine if there is a relationship between the two variables. We can do this by using the Correlation coefficient and scatter plot. When a correlation coefficient shows that data is likely to be able to predict future outcomes and a scatter plot of the data appears to form a straight line, we can use simple linear regression to find a predictive function. Let us consider an example.

From the scatter plot we can see there is a linear relationship between Sales and marketing spent. The next step is to find a straight line between Sales and Marketing that explain the relationship between them. But there can be multiple lines that can pass through these points.

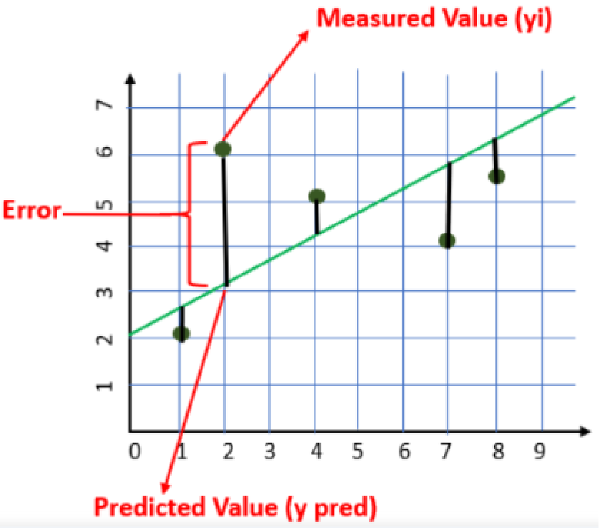
So how do we know which of these lines is the best fit line? That's the





problem that we will solve in this article. For this, we will first look at the cost function.

Cost-function: The cost function is defined as the measurement of difference or error between actual values and expected values at the current position and present in the form of a single real number.



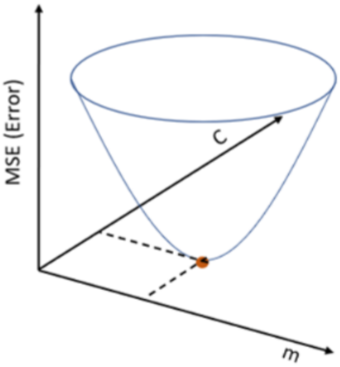
The cost function (MSE) is given by,

$$\text{MSE} = \frac{1}{n}\sum_{i=0}^{n}(y_i - y_{i\ pred})^2$$

Substituting $y_{i\ pred} = mx_i + c$ we get,

$$\text{MSE} = \frac{1}{n}\sum_{i=0}^{n}(y_i - (mx_i + c))^2$$

Our goal is to minimize the cost as much as possible in order to find the best fit line. We are not going to try all the permutation and combination of m and c (inefficient way) to find the best-fit line. For that, we will use Gradient Descent Algorithm. Grediant Descent Algorithm: Gradient Descent is an algorithm that finds the best-fit line for a given training dataset in a smaller number of iterations. If we plot m and c against MSE, it will acquire a bowl shape (As shown in the diagram below)



For some combination of m and c, we will get the least Error (MSE). That combination of m and c will give us our best fit line. The algorithm starts with some value of m and c. Then we reduce the value of m and c by some amount (Learning Step). We will notice a decrease in MSE (cost). We will continue doing the same until our loss function is a very small value or ideally 0 (which means 0 error or 100 percent accuracy).

Steps in Grediant Descent Algorithm:

1. Let L be our learning rate. It could be a small value like 0.01 for good accuracy. Learning rate gives the rate of speed where the gradient moves during gradient descent. Setting it too high would make your path instable, too low would make convergence slow. Put it to zero means your model isn't learning anything from the gradients.

2. Calculate the partial derivative of the cost function with respect to m. Let partial derivative of the cost function with respect to m be D_m (With little change in m how much cost function changes).

   Similarly, find the partial derivative with respect to c. Let partial derivative of the cost function with respect to c be D_c (With little change in c how much cost function changes).

3. Now update the current values of m and c using the following equation: $m = m - LD_m$ and $c = c - LD_c$.

4. We will repeat this process until our cost function is very small (ideally 0).

Types of Errors: (i) Residual error(SSE/RSS) is difference between actual and predicted values. It is also called as unexplained error.
(ii) Regression error(SSR) is a difference between predicted value and mean value. It is also called as explained error.
The total error is given by, SST = SSE + SSR, which is difference between actual value and mean value.
Coefficient of determination(r^2): Coefficient of Determination also popularly known as r square value is a regression error metric to evaluate the accuracy and efficiency of a model on the data values that it would be applied to.

r square value describes the performance of the model. It describes the variation in the response or target variable which is predicted by the independent variables of the data model.

Thus, in simple words we can say that, the r square value helps determine how well the model is blend and how well the output value is explained by the determining(independent) variables of the dataset.
The value of r square ranges between [0,1]. The formula to find the r square values is given below:

$$r^2 = 1 - \frac{\text{SSE}}{\text{SST}}$$

where SSE represents the sum of squares of the residual errors of the data model and SST represents the total sum of the errors.

Higher is the r square value, better is the model and the results.
• It is used to check goodness of BFL.
• r square is not considered as very good metric for evaluating the model because it is impacted by bad predictors.
• We make used of adjusted r square which will increase only when we add good predictors.
• Adjusted r square is always less than r square.
• The formula to find adjusted r square is given by,

$$Adjusted\ r^2 = 1 - \frac{(1-r^2)(1-n)}{n-k-1}$$

where n is sample size and k is number of features. **Variance Inflation Factor(VIF):** Variable Inflation Factors is used to detect multicollinearity. VIF determines the strength of the correlation between the independent variables. It is predicted by taking a variable and regressing it against every other variable. It is calculated by using following formula:

$$\text{VIF} = \frac{1}{1-r^2}$$

Stochastic Gradient Descent: When we have large number of observations and features the Gradient descent become slower. So in this case we can use stochastic gradient descent. It picks randomly one observation per instance hence it is also called as mini-batch Gradient descent. Stochastic Gradient descent work faster on huge dataset.

Advantages and Disadvantages of linear regression

Advantages:

1. Linear regression performs exceptionally well for linearly separable data.

2. Easier to implement, interpret and efficient to train.

3. It handles overfitting pretty well using dimensionally reduction techniques, regularization, and cross-validation.

4. One more advantage is the extrapolation beyond a specific data set.

Disadvantages:

1. The assumption of linearity between dependent and independent variables.

2. It is often quite prone to noise and overfitting.

3. Linear regression is quite sensitive to outliers.

4. It is prone to multicollinearity.

Encoding: The performance of a machine learning model not only depends on the model and the hyperparameters but also on how we process and feed different types of variables to the model. Since most machine learning models only accept numerical variables, preprocessing the categorical variables becomes a necessary step. We need to convert these categorical variables to numbers such that the model is able to understand and extract valuable information. Categorical variables are usually represented as 'strings' or 'categories' and are finite in number.
Ordinal Data: The categories have an inherent order. e.g low-medium-high, 0-1, child-father-grandfather etc.

Nominal Data: The categories do not have an inherent order. e.g. city names, departments(Finance, IT, HR) etc.
Types of encoding:

1. Label Encoding or Ordinal Encoding

2. One hot Encoding

3. Dummy Encoding

4. Binary Encoding

5. Target Encoding

6. Effect Encoding

7. BaseN Encoding

8. Hash Encoding

Label encoding/Ordinal encoding: We use this categorical data encoding technique when the categorical feature is ordinal. In this case, retaining the order is important. Hence encoding should reflect the sequence. In Label encoding, each label is converted into an integer value.
One hot encoding: We use this categorical data encoding technique when the features are nominal(do not have any order). In one hot encoding, for each level of a categorical feature, we create a new variable. Each category is mapped with a binary variable containing either 0 or 1. Here, 0 represents the absence, and 1 represents the presence of that category. These newly created binary features are known as dummy variables.

- If unique values are more than 10 try not to use label encoding.
- If unique values are more than 30 try not to use get dummies.
- If unique values are more than 50 try not to use that feature.

Logistic Regression:

1. Logistic regression is one of the most popular Machine Learning algorithms, which comes under the Supervised Learning technique. It is used for predicting the categorical dependent variable using a given set of independent variables.

2. Logistic regression predicts the output of a categorical dependent variable. Therefore the outcome must be a categorical or discrete value. It can be either Yes or No, 0 or 1, true or False, etc. but instead of giving the exact value as 0 and 1, it gives the probabilistic values which lie between 0 and 1.

3. Logistic Regression is much similar to the Linear Regression except that how they are used. Linear Regression is used for solving Regression problems, whereas Logistic regression is used for solving the classification problems.

4. In Logistic regression, instead of fitting a regression line, we fit an "S" shaped logistic function, which predicts two maximum values (0 or 1).

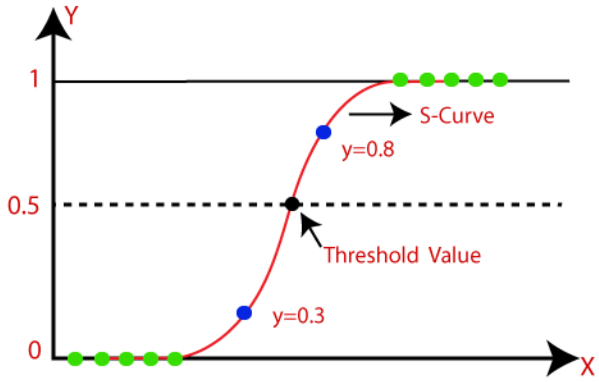
5. Logistic Regression is a significant machine learning algorithm because it has the ability to provide probabilities and classify new data using continuous and discrete datasets.

Logistic Function (Sigmoid Function):

1. The sigmoid function is a mathematical function used to map the predicted values to probabilities. It maps any real value into another value within a range of 0 and 1.

2. The value of the logistic regression must be between 0 and 1, which cannot go beyond this limit, so it forms a curve like the S form. The S-form curve is called the Sigmoid function or the logistic function.

3. In logistic regression, we use the concept of the threshold value, which defines the probability of either 0 or 1. Such as values above the threshold value tends to 1, and a value below the threshold values tends to 0.



Types of Logistic Regression:

1. Binary classification: Target feature consist of two values such as 0 or 1, Pass or Fail, etc.

2. Multiclass classification: Target feature consist of more than two values such as cat, dogs, or sheep.

Advantages and Disadvantages of logistic regression

Advantages:

1. Makes no assumptions about distribution of class.

2. We can easily use binary and multiclass classification.

3. Quick to train the model.

4. Very fast at classifying unknown records.

5. Good accuracy on many datasets.

6. Resisting to overfitting.

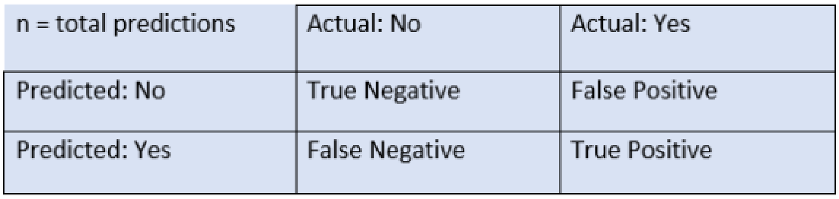
Disadvantages:

1. Sensitive to outliers.

2. Construct linear boundaries.
3. Required large datasets.

Evaluation metrics-Classification: Confusion matrix: The confusion matrix is a matrix used to determine the performance of the classification models for a given set of test data. It can only be determined if the true values for test data are known. The matrix itself can be easily understood, but the related terminologies may be confusing. Since it shows the errors in the model performance in the form of a matrix, hence also known as an error matrix.

The matrix is divided into two dimensions, that are predicted values and actual values along with the total number of predictions. Predicted values are those values, which are predicted by the model, and actual values are the true values for the given observations. It looks like the below table: The above table has the following cases:



n = total predictions	Actual: No	Actual: Yes
Predicted: No	True Negative	False Positive
Predicted: Yes	False Negative	True Positive

1. **True Negative:** Model has given prediction No, and the real or actual value was also No.
2. **True Positive:** The model has predicted yes, and the actual value was also true.
3. **False Negative:** The model has predicted no, but the actual value was Yes, it is also called as **Type-II error**.

4. **False Positive:** The model has predicted Yes, but the actual value was No. It is also called a **Type-I error**.

Calculations using Confusion Matrix:

1. **Classification Accuracy:** It is one of the important parameters to determine the accuracy of the classification problems. It defines how often the model predicts the correct output. It can be calculated as the ratio of the number of correct predictions made by the classifier to all number of predictions made by the classifiers. The formula is given below:

   $\text{Accuracy} = \frac{\text{TP+TN}}{\text{TP+TN+FP+FN}}$

2. **Misclassification rate:** It is also termed as Error rate, and it defines how often the model gives the wrong predictions. The value of error rate can be calculated as the number of incorrect predictions to all number of the predictions made by the classifier. The formula is given below:

   $\text{Error rate} = \frac{\text{FP+FN}}{\text{TP+TN+FP+FN}}$

3. **Precision:** It can be defined as the number of correct outputs provided by the model or out of all positive classes that have predicted correctly by the model, how many of them were actually true. If FP is important then precision is appropriate metric. e.g Finance, e-commerce etc. It can be calculated using the below formula:

   $\text{Precision} = \frac{\text{TP}}{\text{TP+FP}}$

4. **Recall:** It is defined as the out of actual positive classes, how many are positively predicted. The recall must be as high as possible. If FN is important then recall is appropriate metric. e.g. Medical, Pharma etc.
   Recall = $\frac{\text{TP}}{\text{TP+FN}}$

5. **F_1-score:** If two models have low precision and high recall or vice versa, it is difficult to compare these models. So, for this purpose, we can use F_1-score. This score helps us to evaluate the recall and precision at the same time. The F_1-score is maximum if the recall is equal to the precision. It can be calculated using the below formula:

   F_1-score = $\frac{2\times\text{Recall}\times\text{Precision}}{\text{Recall + Precision}}$

6. **AUC/ROC Curve:** The ROC is a graph displaying a classifier's performance for all possible thresholds. The graph is plotted between the true positive rate (on the Y-axis) and the false Positive rate (on the x-axis).

Regularization: Regularization is one of the most important concepts of machine learning. It is a technique to prevent the model from overfitting by adding extra information to it.

Sometimes the machine learning model performs well with the training data but does not perform well with the test data. It means the model is not able to predict the output when deals with unseen data by introducing noise in the output, and hence the model is called overfitted. This problem can be deal with the help of a regularization technique.

This technique can be used in such a way that it will allow to maintain all variables or features in the model by reducing the magnitude of the variables. Hence, it maintains accuracy as well as a generalization of the model.

It mainly regularizes or reduces the coefficient of features toward zero. In simple words, In regularization technique, we reduce the magnitude of the features by keeping the same number of features. **Working of Regularization:** Regularization works by adding a penalty or complexity term to the complex model. Let's consider the simple linear regression equation:

$$y = \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_n x_n + b$$

In the above equation, y represents the value to be predicted, $x_1, x_2, .., x_n$ are the features for y, $\beta_1, \beta_2, ..., \beta_n$ are the weights or magnitude attached to the features, respectively. Here represents the bias of the model, and b represents the intercept.

Linear regression models try to optimize the values of β_j and b to minimize the cost function. Then we will add a loss function and optimize parameter to make the model that can predict the accurate value of Y. The loss function for the linear regression is called as RSS or Residual sum of squares.

Techniques of Regularization: There are mainly two types of regularization techniques, which are given below:

1. Ridge Regression
2. Lasso Regression

1. Ridge regularization/regression: Ridge regression is one of the types of linear regression in which a small amount of bias is introduced so that we can get better long-term predictions.

Ridge regression is a regularization technique, which is used to reduce the complexity of the model. It is also called as L2 regularization.

In this technique, the cost function is altered by adding the penalty term to it. The amount of bias added to the model is called Ridge Regression penalty. We can calculate it by multiplying with the lambda to the squared weight of each individual feature.

The equation for the cost function in ridge regression will be:

$$\text{Cost} = \text{Loss} + \lambda \sum_{j=1}^{n} \beta_j^2$$

In the above equation, the penalty term regularizes the coefficients of the model, and hence ridge regression reduces the amplitudes of the coefficients that decreases the complexity of the model.

As we can see from the above equation, if the values of λ tend to zero, the equation becomes the cost function of the linear regression model. Hence, for the minimum value of λ, the model will resemble the linear regression model.

A general linear or polynomial regression will fail if there is high collinearity between the independent variables, so to solve such problems, Ridge regression can be used. It helps to solve the problems if we have more parameters than samples. **2. Lasso regularization/regression:** Lasso regression is another regularization technique to reduce the complexity of the model. It stands for Least Absolute and Selection Operator.

It is similar to the Ridge Regression except that the penalty term contains only the absolute weights instead of a square of weights.

Since it takes absolute values, hence, it can shrink the slope to 0, whereas Ridge Regression can only shrink it near to 0.

It is also called as L1 regularization. The equation for the cost function of Lasso regression will be:

$$\text{Cost} = \text{Loss} + \lambda \sum_{j=1}^{n} \beta_j$$

Some of the features in this technique are completely neglected for model evaluation. Hence, the Lasso regression can help us to reduce the overfitting in the model as well as the feature selection.

Difference between Lasso and Ridge regularization:

Ridge regression is mostly used to reduce the overfitting in the model, and it includes all the features present in the model. It reduces the complexity of the model by shrinking the coefficients.

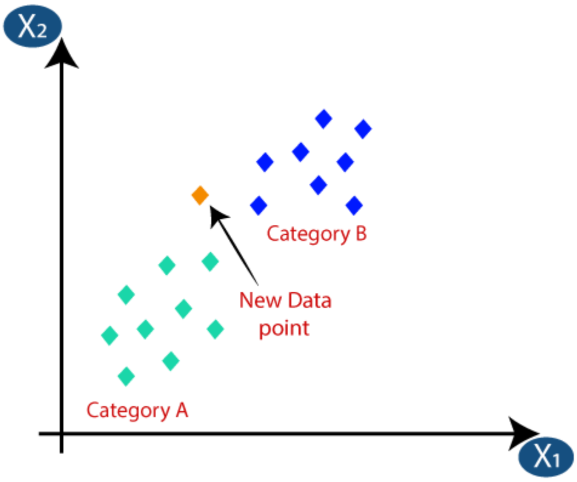
Lasso regression helps to reduce the overfitting in the model as well as feature selection.

K-Nearest Neighbour(KNN):

1. K-Nearest Neighbour is one of the simplest Machine Learning algorithms based on Supervised Learning technique.
2. K-NN algorithm assumes the similarity between the new case/data and available cases and put the new case into the category that is most similar to the available categories.
3. K-NN algorithm stores all the available data and classifies a new data point based on the similarity. This means when new data appears then it can be easily classified into a well suite category by using K- NN algorithm.
4. K-NN algorithm can be used for Regression as well as for Classification but mostly it is used for the Classification problems.
5. K-NN is a non-parametric algorithm, which means it does not make any assumption on underlying data.
6. It is also called a lazy learner algorithm because it does not learn from the training set immediately instead it stores the dataset and at the time of classification, it performs an action on the dataset.
7. KNN algorithm at the training phase just stores the dataset and when it gets new data, then it classifies that data into a category that is much similar to the new data.

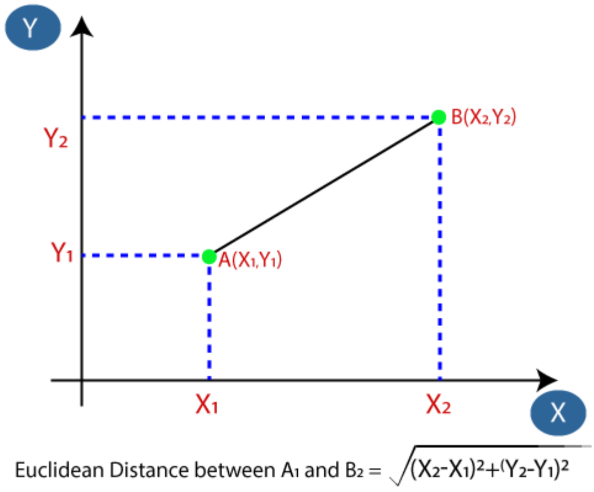
Example: Suppose, we have an image of a creature that looks similar to cat and dog, but we want to know either it is a cat or dog. So for this identification, we can use the KNN algorithm, as it works on a similarity measure. Our KNN model will find the similar features of the new data set to the cats and dogs images and based on the most similar features it will put it in either cat or dog category.

Steps in KNN algorithm:
The K-NN working can be explained on the basis of the below algorithm:
Step-1: Select the number K of the neighbours.
Step-2: Calculate the Euclidean distance of K number of neighbours.
Step-3: Take the K nearest neighbours as per the calculated Euclidean distance.
Step-4: Among these k neighbours, 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 neighbour is maximum.
Step-6: Our model is ready.
Suppose we have a new data point and we need to put it in the required category. Consider the below image:

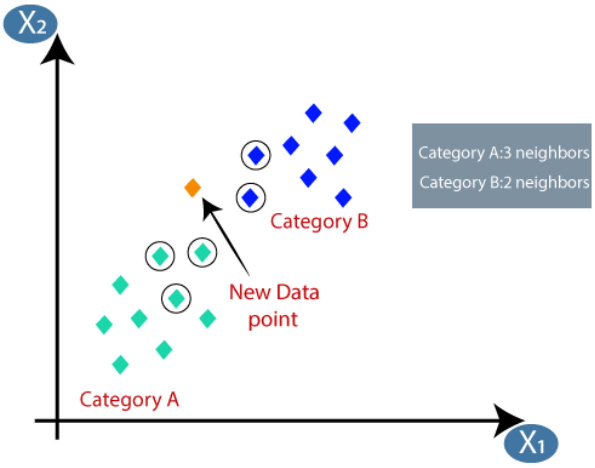


Firstly, we will choose the number of neighbours, so we will choose the k=5.

Next, we will calculate the Euclidean distance between the data points. The Euclidean distance is the distance between two points, which we have already studied in geometry. It can be calculated as:



By calculating the Euclidean distance we got the nearest neighbours, as three nearest neighbours in category A and two nearest neighbours in category B. Consider the below image:



As we can see the 3 nearest neighbours are from category A, hence this new data point must belong to category A.

Advantages and Disadvantages of KNN

Advantages:

1. It is simple to implement.
2. It is robust to the noisy training data.
3. It can be more effective if the training data is large.

Disadvantages:

1. Always needs to determine the value of K which may be complex some time.
2. The computation cost is high because of calculating the distance between the data points for all the training samples.

Features scaling: It is a technique/method used to normalise/standardize the independent features of data in a range.

Types of feature scaling:

1. Min Max Scaler (Normalization)
2. Standard Scaler (Standardization)
3. Robust Scaler (Effective on Outliers)
4. Max Absolute
5. Unit Vector Scaler
6. Power Transform
7. Quantile Transform
8. Mean Normalization

Note:

1. **Distance Based Algorithms**(Scaling is Required)
   (a) KNN
   (b) K-Means Clustering

   (c) SVM(Support Vector Machine)
   (d) PCA(Principal Component Analysis)

2. **Gradient Descent Based Algorithms**(Scaling is optional)
   (a) Linear Regression
   (b) Logistic Regression
   (c) Neural Network
   (d) Lasso and Ridge

3. **Tree Based Algorithms**(Scaling is not Required)
   (a) Decision tree
   (b) Random Forest
   (c) AdaBoost
   (d) Gradient Boost
   (e) XGBoost
   (f) CatBoost

Normalization:

1. This technique uses minimum and max values for scaling of model.
2. It is helpful when features are of different scales.
3. Scales values ranges between [0, 1] or [-1, 1].
4. It got affected by outliers.
5. Scikit-Learn provides a transformer called MinMaxScaler for Normalization.
6. It is also called Scaling normalization.
7. It is useful when feature distribution is unknown.

8. Normalisation can be expressed as: $x_{norm} = \dfrac{x_i - x_{min}}{x_{max} - x_{min}}$

Standardization:

1. This technique uses mean and standard deviation for scaling of model.
2. It is helpful when the mean of a variable is set to 0 and the standard deviation is set to 1.
3. Scale values are not restricted to a specific range.
4. It is comparatively less affected by outliers.
5. Scikit-Learn provides a transformer called StandardScaler for Normalization.
6. It is known as Z-score normalization.
7. It is useful when feature distribution is normal.
8. Standardization can be expressed as: $x_{std} = \dfrac{x_i - \mu}{\sigma}$

Outliers: Data points which area far away from observed values or out of the box values.

How outliers are introduced in the datasets?

1. Data Entry Error : Human Error
2. Measurement Error : Machine Error/Instrumental Error
3. Intentional Error : Dummy Dataset
4. Sampling Error : Mixing of Data from Wrong sources
5. Natural Errors : Most of actual Data belongs to this Category

Techniques to detect outliers:

1. Z-Score
2. IQR

3. Boxplot
4. Scatterplot
5. Hypothesis Testing

How to handle outliers?

1. Delete Observations/Trimming (Not preferable)
2. Imputation/Capping:
   (a) Mean
   (b) Median
   (c) zero
   (d) minimum
   (e) maximum
   (f) upper limit
   (g) lower tail
   (h) any static value
3. Transformations: (Used to reduce the impact of outliers)
   (a) Log Transformations
   (b) Cuberoot
   (c) Sqrt
   (d) Reciprocal
   (e) Standadization
   (f) Robust Scaling

Impact of Outliers:

1. Reduce the power of Statistical Analysis

2. High impact on mean and std values
3. Algorithm do not perform well in presence of outliers
4. Impact of basic assumptions of regression (normality, homoscedasticity)

Sensitive and non-sensitive algorithms to outliers:

1. **Sensitive:**
   (a) Linear Regression
   (b) Logistic Regression
   (c) K-Nearest Neighbor
   (d) Support Vector Machine
   (e) KMeans Clustering
2. **Non-sensitive:**
   (a) Decision Tree
   (b) Random Forest
   (c) AdaBoost
   (d) Gradient Boost
   (e) XGBoost
   (f) Naive Bayes

Decision Tree Classification Algorithm

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

2. In a Decision tree, there are two nodes, which are the Decision Node and Leaf Node. Decision nodes are used to make any decision and have multiple branches, whereas Leaf nodes are the output of those decisions and do not contain any further branches.

3. The decisions or the test are performed on the basis of features of the given dataset.

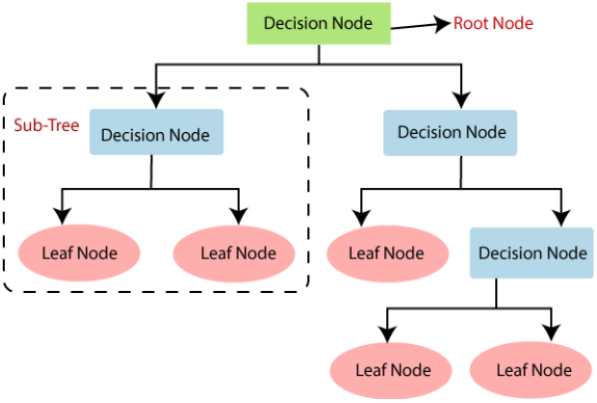
4. It is a graphical representation for getting all the possible solutions to a problem/decision based on given conditions.

5. It is called a decision tree because, similar to a tree, it starts with the root node, which expands on further branches and constructs a tree-like structure.

6. In order to build a tree, we use the CART algorithm, which stands for Classification and Regression Tree algorithm.

7. A decision tree simply asks a question, and based on the answer (Yes/No), it further split the tree into subtrees.

8. Below diagram explains the general structure of a decision tree:



Why use Decision Trees?

There are various algorithms in Machine learning, so choosing the best algorithm for the given dataset and problem is the main point to remember while creating a machine learning model. Below are the two reasons for using the Decision tree:

1. Decision Trees usually mimic human thinking ability while making a decision, so it is easy to understand.
2. The logic behind the decision tree can be easily understood because it shows a tree-like structure.

Decision Tree Terminologies

1. **Root Node:** Root node is from where the decision tree starts. It represents the entire dataset, which further gets divided into two or more homogeneous sets.
2. **Leaf Node:** Leaf nodes are the final output node, and the tree cannot be segregated further after getting a leaf node.
3. **Splitting:** Splitting is the process of dividing the decision node/root node into sub-nodes according to the given conditions.
4. **Branch/Sub Tree:** A tree formed by splitting the tree.
5. **Pruning:** Pruning is the process of removing the unwanted branches from the tree.
6. **Parent/Child node:** The root node of the tree is called the parent node, and other nodes are called the child nodes.

How does the Decision Tree algorithm Work?

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

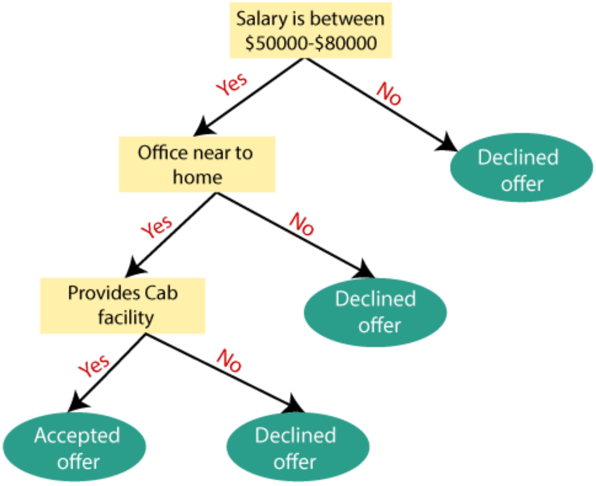
2. **Step-2:** Find the best attribute in the dataset using Attribute Selection Measure (ASM).

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

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

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

Example: Suppose there is a candidate who has a job offer and wants to decide whether he should accept the offer or Not. So, to solve this problem, 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 based on the corresponding labels. The next decision node further gets split into one decision node (Cab facility) and one leaf node. Finally, the decision node splits into two leaf nodes (Accepted offers and Declined offer). Consider the below diagram:



Attribute Selection Measures:

While implementing a Decision tree, the main issue arises that how to select the best attribute for the root node and for sub-nodes. So, to solve such problems there is a technique which is called as Attribute selection measure or ASM. By this measurement, we can easily select the best attribute for the nodes of the tree. There are two popular techniques for ASM, which are:

1. Information Gain
2. Gini Index

1. Information Gain:

1. Information gain is the measurement of changes in entropy after the segmentation of a dataset based on an attribute.
2. It calculates how much information a feature provides us about a class.
3. According to the value of information gain, we split the node and build the decision tree.
4. A decision tree algorithm always tries to maximize the value of information gain, and a node/attribute having the highest information gain is split first. It can be calculated using the below formula:

   IG = Entropy(S)- [(Weighted Avg) *Entropy (each feature)]

Entropy: Entropy is a metric to measure the impurity in a given attribute. It specifies randomness in data. Entropy can be calculated as:

$$\text{Entropy(s)}= -p(\text{yes})\log_2[p(\text{yes})] - p(\text{no})\log_2[p(\text{no})]$$

2. Gini Index:

1. Gini index is a measure of impurity or purity used while creating a decision tree in the CART(Classification and Regression Tree) algorithm.
2. An attribute with the low Gini index should be preferred as compared to the high Gini index.

3. It only creates binary splits, and the CART algorithm uses the Gini index to create binary splits.

4. Gini index can be calculated using the below formula:

$$\text{Gini Index} = 1 - \sum_j p_j^2$$

Pruning: Pruning is a process of deleting the unnecessary nodes from a tree in order to get the optimal decision tree.

Advantages and Disadvantages of Decision Tree

Advantages:

1. It is simple to understand as it follows the same process which a human follow while making any decision in real-life.

2. It can be very useful for solving decision-related problems.

3. It helps to think about all the possible outcomes for a problem.

4. There is less requirement of data cleaning compared to other algorithms.

Disadvantages:

1. The decision tree contains lots of layers, which makes it complex.

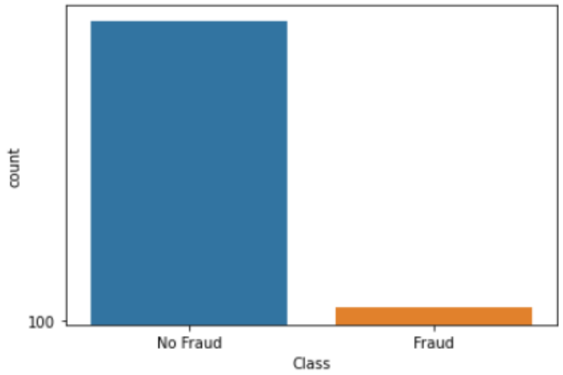
2. It may have an overfitting issue, which can be resolved using the Random Forest algorithm.

3. For more class labels, the computational complexity of the decision tree may increase.

Imbalanced Data:

When observation in one class is higher than the observation in other classes then there exists a class imbalance.

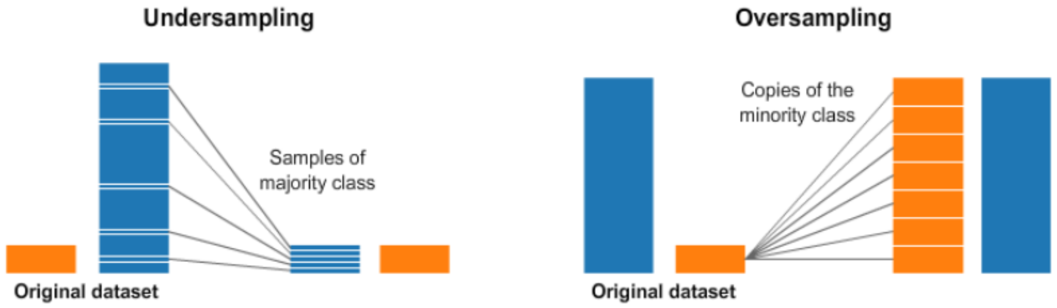
Example: To detect fraudulent credit card transactions. As you can see in the below graph fraudulent transaction is around 400 when compared with non-fraudulent transaction around 90000.



Techniques to handle imbalanced data:

1. **Undersampling:** It is defined as removing some observations of the majority class. This is done until the majority and minority class is balanced out. Which can cause loss of information.

2. **Oversampling:** It is defined as adding more copies to the minority class. Oversampling can be a good choice when you don't have a ton of data to work with. Which can cause overfitting.



Undersampling:

1. Random Undersampling

2. NearMiss

3. Condensed Nearest Neighbor(KNN)

Oversampling:

1. RandomOverSampler
2. SMOTE(Synthetic Minority Oversampling Technique)
3. ADASYN (Adaptive Synthetic Sampling)
4. SmoteTomek

Ensemble Techniques: Ensemble methods combine different decision trees to deliver better predictive results, afterward utilizing a single decision tree. The primary principle behind the ensemble model is that a group of weak learners come together to form an active learner. There are two techniques given below that are used to perform ensemble decision tree.

1. Bagging
2. Boosting

1. Bagging:

Bagging is used when our objective is to reduce the variance of a decision tree. Here the concept is to create a few subsets of data from the training sample, which is chosen randomly with replacement. Now each collection of subset data is used to prepare their decision trees thus, we end up with an ensemble of various models. The average of all the assumptions from numerous tress is used, which is more powerful than a single decision tree.

1. Various training data subsets are randomly drawn with replacement from the whole training dataset.
2. Bagging attempts to tackle the over-fitting issue.
3. If the classifier is unstable (high variance), then we need to apply bagging.
4. Every model receives an equal weight.

5. Objective to decrease variance, not bias.
6. It is the easiest way of connecting predictions that belong to the same type.
7. Every model is constructed independently.

Random Forest is an expansion over bagging. It takes one additional step to predict a random subset of data. It also makes the random selection of features rather than using all features to develop trees. When we have numerous random trees, it is called the Random Forest.
These are the following steps which are taken to implement a Random forest:

1. Let us consider X observations Y features in the training data set. First, a model from the training data set is taken randomly with substitution.
2. The tree is developed to the largest.
3. The given steps are repeated, and prediction is given, which is based on the collection of predictions from n number of trees.

Advantages and Disadvantages of Random Forest

Advantages:

1. It is used for classification as well as regression
2. It reduces overfitting of decision trees and helps to improve performance
3. No feature scaling is required
4. Non sensitive to outliers
5. It is Non-Parametric algorithm
6. Random Forest is more stable than Decision Tree

Disadvantages:

1. It manages a higher dimension data set very well.

2. It manages missing quantities and keeps accuracy for missing data.

2. Boosting:
Boosting is another ensemble procedure to make a collection of predictors. In other words, we fit consecutive trees, usually random samples, and at each step, the objective is to solve net error from the prior trees.

If a given input is misclassified by theory, then its weight is increased so that the upcoming hypothesis is more likely to classify it correctly by consolidating the entire set at last converts weak learners into better performing models.

Gradient Boosting is an expansion of the boosting procedure.
Gradient Boosting = Gradient Descent + Boosting

1. Each new subset contains the components that were misclassified by previous models.
2. Boosting tries to reduce bias.
3. If the classifier is steady and straightforward (high bias), then we need to apply boosting.
4. Models are weighted by their performance.
5. Objective to decrease bias, not variance.
6. It is a way of connecting predictions that belong to the different types.
7. New models are affected by the performance of the previously developed model.

Steps in Boosting techniques:

1. Consider a dataset having different data points and initialize it.
2. Now, give equal weight to each of the data points.
3. Assume this weight as an input for the model.

4. Identify the data points that are incorrectly classified.
5. Increase the weight for data points in step 4.
6. If you get appropriate output then terminate this process else follow steps 2 and 3 again.

Boosting Techniques:

1. Adaptive Boosting(AdaBoost)
2. Gradient Boosting Machine(GBM)
3. Extreme Gradient Boosting Machine(XGBM)
4. Light GBM
5. CatBoost

1. Adaptive Boosting:

The most common algorithm used with AdaBoost is decision trees with one level that means with Decision trees with only 1 split. These trees are also called Decision Stumps.

It builds a model and gives equal weights to all the data points. It then assigns higher weights to points that are wrongly classified. Now all the points which have higher weights are given more importance in the next model. It will keep training models until and unless a lower error is received.

Advantages and Disadvantages of AdaBoost

Advantages:

1. Advantages:
2. Because of stage wise estimation it tries to reduce the errors
3. It is used to improve accuracy of weak learners
4. Used for image and text classification

Disadvantages:

1. Only one spliiting (stumps)
2. Adaboost is slower than XGboost

K-Means Clustering Algorithm

K-Means Clustering is an unsupervised learning algorithm that is used to solve the clustering problems in machine learning or data science.

What is K-Means Algorithm?

K-Means Clustering is an Unsupervised Learning algorithm, which groups the unlabeled dataset into different clusters. Here K defines the number of pre-defined clusters that need to be created in the process, as if $K = 2$, there will be two clusters, and for $K = 3$, there will be three clusters, and so on.

It is an iterative algorithm that divides the unlabelled dataset into k different clusters in such a way that each dataset belongs only one group that has similar properties.

It allows us to cluster the data into different groups and a convenient way to discover the categories of groups in the unlabeled dataset on its own without the need for any training.

It is a centroid-based algorithm, where each cluster is associated with a centroid. The main aim of this algorithm is to minimize the sum of distances between the data point and their corresponding clusters.

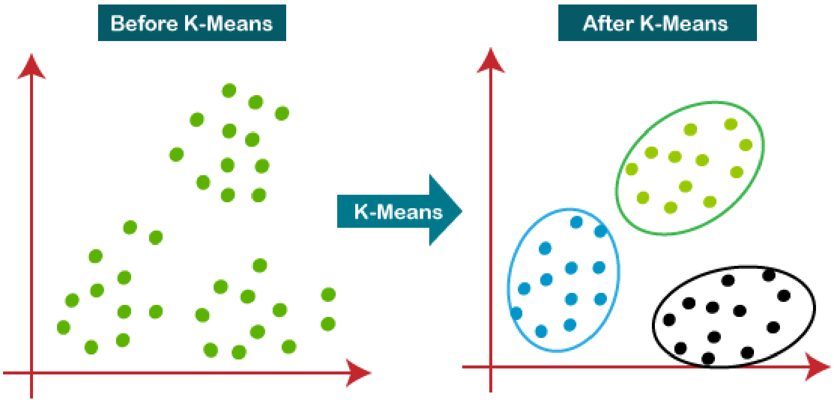
The algorithm takes the unlabeled dataset as input, divides the dataset into k-number of clusters, and repeats the process until it does not find the best clusters. The value of k should be predetermined in this algorithm.

The k-means clustering algorithm mainly performs two tasks:

1. Determines the best value for K center points or centroids by an iterative process.
2. Assigns each data point to its closest k-center. Those data points which are near to the particular k-center, create a cluster.

Hence each cluster has datapoints with some commonalities, and it is away from other clusters.

The below diagram explains the working of the K-means Clustering Algorithm:



How does the K-Means Algorithm Work?

The working of the K-Means algorithm is explained in the below steps:

1. **Step-1:** Select the number K to decide the number of clusters.

2. **Step-2:** Select random K points or centroids. (It can be other from the input dataset).

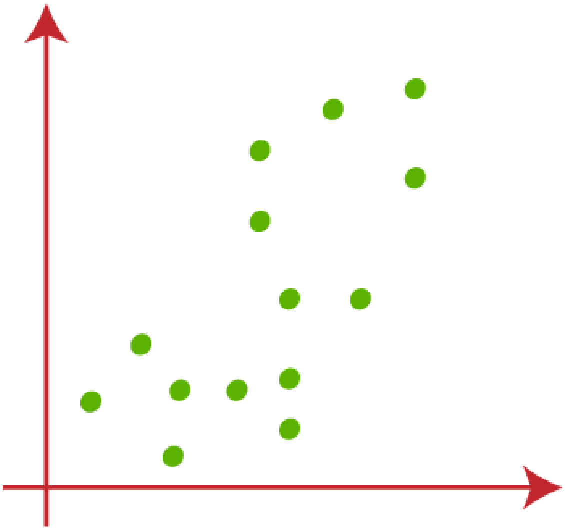
3. **Step-3:** Assign each data point to their closest centroid, which will form the predefined K clusters.

4. **Step-4:** Calculate the variance and place a new centroid of each cluster.

5. **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.

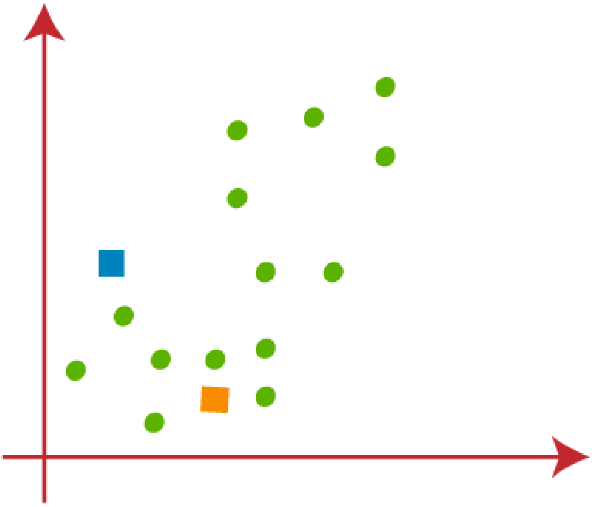
6. **Step-7:** The model is ready.

Suppose we have two variables M_1 and M_2. The $X - Y$ axis scatter plot of these two variables is given below:

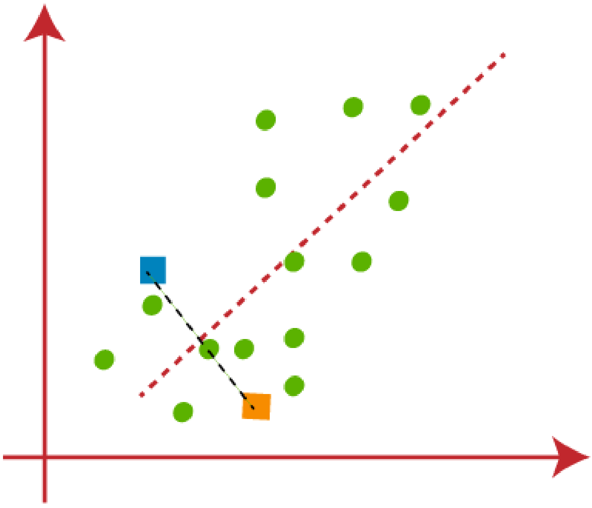


Let's take number K of clusters, i.e., $K = 2$, to identify the dataset and to put them into different clusters. It means here we will try to group these datasets into two different clusters.

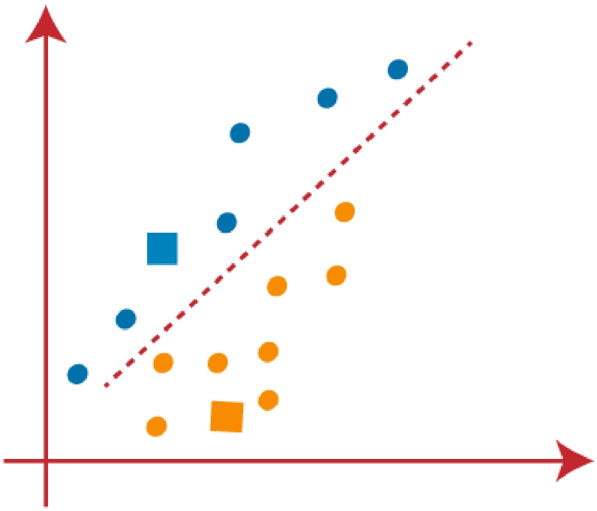
We need to choose some random k points or centroid to form the cluster. These points can be either the points from the dataset or any other point. So, here we are selecting the below two points as k points, which are not the part of our dataset. Consider the below image:



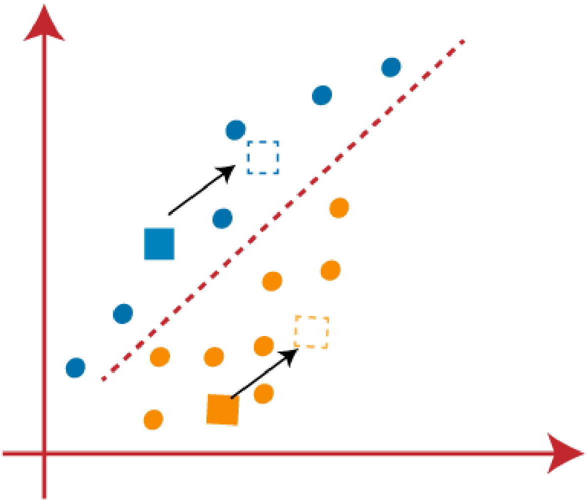
Now we will assign each data point of the scatter plot to its closest K-point or centroid. We will compute it by applying some mathematics that we have studied to calculate the distance between two points. So, we will draw a median between both the centroids. Consider the below image:



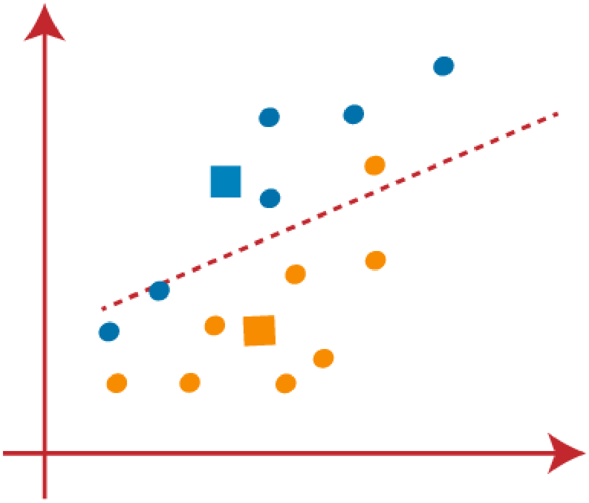
From the above image, it is clear that points left side of the line is near to the K_1 or blue centroid, and points to the right of the line are close to the yellow centroid. Let's color them as blue and yellow for clear visualization.



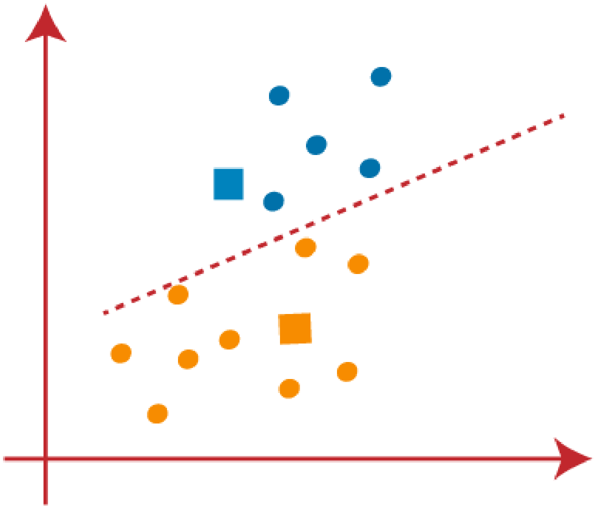
As we need to find the closest cluster, so we will repeat the process by choosing a new centroid. To choose the new centroids, we will compute the centre of gravity of these centroids, and will find new centroids as below:



Next, we will reassign each datapoint to the new centroid. For this, we will repeat the same process of finding a median line. The median will be like below image:

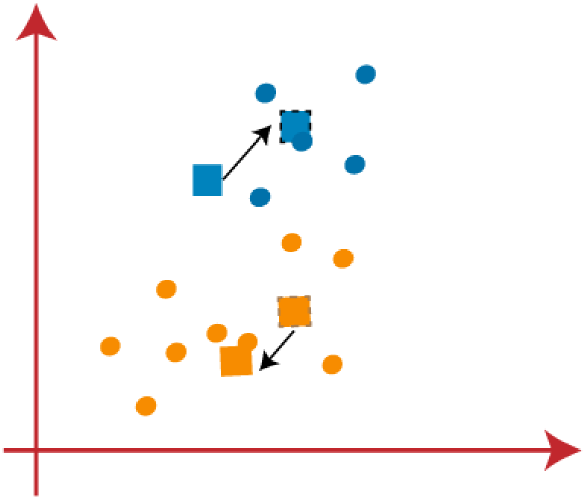


From the above image, we can see, one yellow point is on the left side of the line, and two blue points are right to the line. So, these three points will be assigned to new centroids.

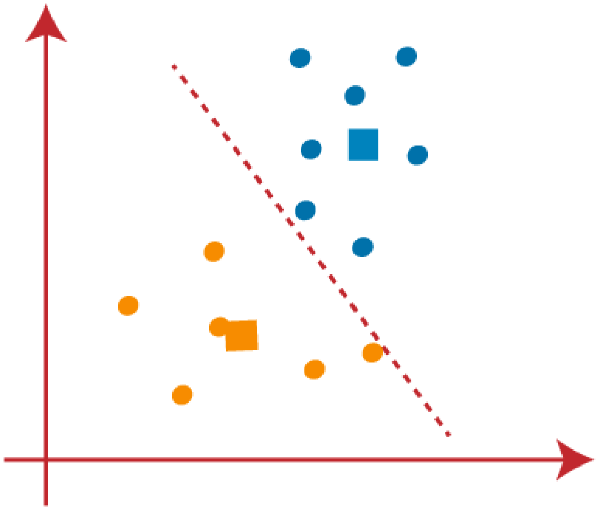


As reassignment has taken place, so we will again go to the step-4, which is finding new centroids or K-points.

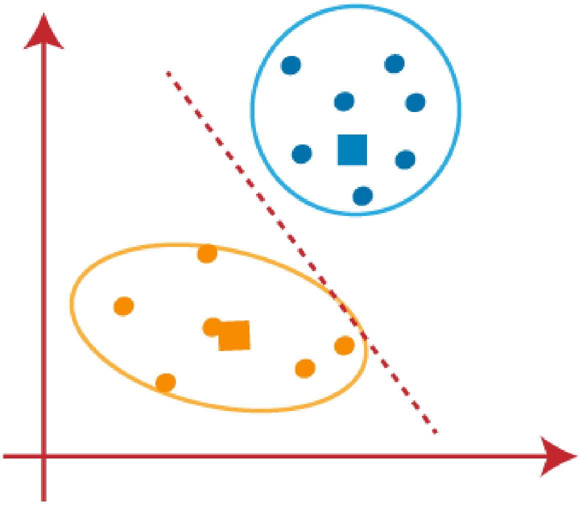
We will repeat the process by finding the center of gravity of centroids, so the new centroids will be as shown in the below image:



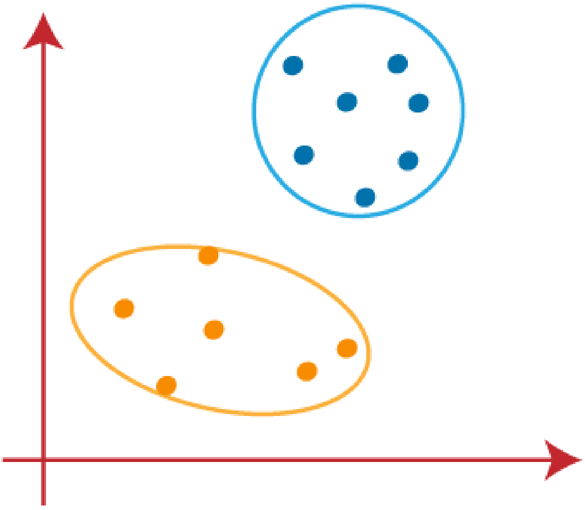
As we got the new centroids so again will draw the median line and reassign the data points. So, the image will be:



We can see in the above image; there are no dissimilar data points on either side of the line, which means our model is formed. Consider the below image:



As our model is ready, so we can now remove the assumed centroids, and the two final clusters will be as shown in the below image:

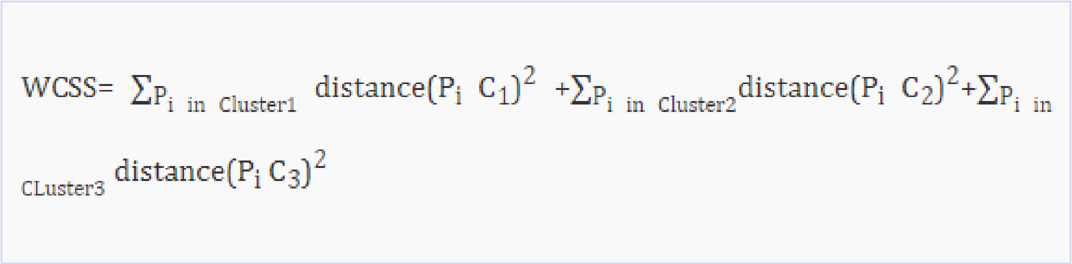


How to choose the value of "K number of clusters" in K-means Clustering?

The performance of the K-means clustering algorithm depends upon highly efficient clusters that it forms. But choosing the optimal number of clusters is a big task. There are some different ways to find the optimal number of clusters, but here we are discussing the most appropriate method to find the number of clusters or value of K. The method is given below:

Elbow Method:

The Elbow method is one of the most popular ways to find the optimal number of clusters. This method uses the concept of WCSS value. WCSS stands for Within Cluster Sum of Squares, which defines the total variations within a cluster. The formula to calculate the value of WCSS (for 3 clusters) is given below:



$$\text{WCSS}= \sum_{P_i \text{ in Cluster1}} \text{distance}(P_i\ C_1)^2 + \sum_{P_i \text{ in Cluster2}} \text{distance}(P_i\ C_2)^2 + \sum_{P_i \text{ in CLuster3}} \text{distance}(P_i\ C_3)^2$$

In the above formula of WCSS,

$\sum P_{\text{i in Cluster1}}\text{distance}(P_i, C_1)^2$ is the sum of the square of the distances between each data point and its centroid within a cluster1 and the same for the other two terms.

To find the optimal value of clusters, the elbow method follows the below steps:

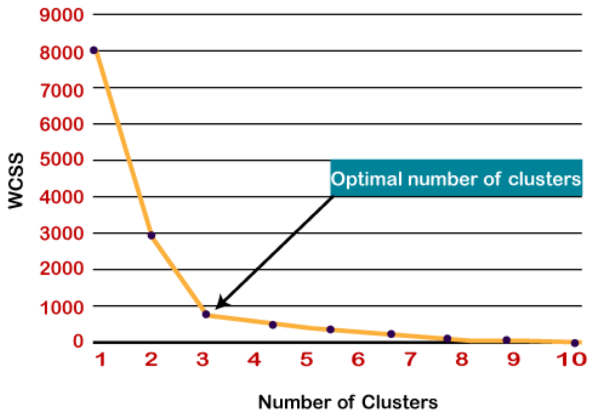
1. It executes the K-means clustering on a given dataset for different K values (ranges from 1-10).

2. For each value of K, calculates the WCSS value.

3. Plots a curve between calculated WCSS values and the number of clusters K.

4. The sharp point of bend or a point of the plot looks like an arm, then that point is considered as the best value of K.

Since the graph shows the sharp bend, which looks like an elbow, hence it is known as the elbow method. The graph for the elbow method looks like the below image:



Naïve Bayes Classifier Algorithm

1. Naïve Bayes algorithm is a supervised learning algorithm, which is based on Bayes theorem and used for solving classification problems.
2. It is mainly used in text classification that includes a high-dimensional training dataset.
3. Naïve Bayes Classifier is one of the simple and most effective Classification algorithms which helps in building the fast machine learning models that can make quick predictions.
4. It is a probabilistic classifier, which means it predicts on the basis of the probability of an object. Some popular examples of Naïve Bayes Algorithm are spam filtration, Sentimental analysis, and classifying articles.

Why is it called Naïve Bayes?
The Naïve Bayes algorithm is comprised of two words Naïve and Bayes, Which can be described as:

1. Naïve: It is called Naïve because it assumes that the occurrence of a certain feature is independent of the occurrence of other features. Such as if the fruit is identified on the bases of color, shape, and taste, then red, spherical, and sweet fruit is recognized as an apple. Hence each feature individually contributes to identify that it is an apple without depending on each other.
2. Bayes: It is called Bayes because it depends on the principle of Bayes' Theorem.

Bayes' Theorem:
Bayes' theorem is also known as Bayes' Rule or Bayes' law, which is used to determine the probability of a hypothesis with prior knowledge. It depends on the conditional probability.

The formula for Bayes' theorem is given as:

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

where,
$P(A|B)$ **is Posterior probability:** Probability of hypothesis A on the observed event B.
$P(B|A)$ **is Likelihood probability:** Probability of the evidence given that the probability of a hypothesis is true.
$P(A)$ **is Prior Probability:** Probability of hypothesis before observing the evidence.
P(B) is Marginal Probability: Probability of Evidence.

Working of Naïve Bayes' Classifier:
Working of Naïve Bayes' Classifier can be understood with the help of the below example:

Suppose we have a dataset of weather conditions and corresponding target variable ”Play”. So using this dataset we need to decide that whether we should play or not on a particular day according to the weather conditions. So to solve this problem, we need to follow the below steps:

1. Convert the given dataset into frequency tables.
2. Generate Likelihood table by finding the probabilities of given features.
3. Now, use Bayes theorem to calculate the posterior probability.

Advantages and Disadvantages of Naïve Bayes

Advantages:

1. Naïve Bayes is one of the fast and easy ML algorithms to predict a class of datasets.
2. It can be used for Binary as well as Multi-class Classifications.

3. It performs well in Multi-class predictions as compared to the other Algorithms.
4. It is the most popular choice for text classification problems.

Disadvantages:

Naive Bayes assumes that all features are independent or unrelated, so it cannot learn the relationship between features.

Applications of Naïve Bayes Classifier:

1. It is used for Credit Scoring.
2. It is used in medical data classification.
3. It can be used in real-time predictions because Naïve Bayes Classifier is an eager learner.
4. It is used in Text classification such as Spam filtering and Sentiment analysis.

Types of Naïve Bayes Model:

1. **Gaussian:** The Gaussian model assumes that features follow a normal distribution. This means if predictors take continuous values instead of discrete, then the model assumes that these values are sampled from the Gaussian distribution.
2. **Multinomial:** The Multinomial Naïve Bayes classifier is used when the data is multinomial distributed. It is primarily used for document classification problems, it means a particular document belongs to which category such as Sports, Politics, education, etc. The classifier uses the frequency of words for the predictors.
3. **Bernoulli:** The Bernoulli classifier works similar to the Multinomial classifier, but the predictor variables are the independent Booleans variables. Such as if a particular word is present

or not in a document. This model is also famous for document classification tasks.

Hypothesis Testing

Null Hypothesis H_0: A null hypothesis is a type of statistical hypothesis which tells that there is no statistically significant effect exists in the given set of observations. It is also known as conjecture and is used in quantitative analysis to test theories about markets, investment, and finance to decide whether an idea is true or false.

Alternative Hypothesis H_1: An alternative hypothesis is a direct contradiction of the null hypothesis, which means if one of the two hypotheses is true, then the other must be false. In other words, an alternative hypothesis is a type of statistical hypothesis which tells that there is some significant effect that exists in the given set of observations.

p-value: The p-value in statistics is defined as the evidence against a null hypothesis. In other words, P-value is the probability that a random chance generated the data or something else that is equal or rarer under the null hypothesis condition.

If the p-value is smaller, the evidence will be stronger, and vice-versa which means the null hypothesis can be rejected in testing. It is always represented in a decimal form, such as 0.035.

Significance level: The significance level is the primary thing that must be set before starting an experiment. It is useful to define the tolerance of error and the level at which effect can be considered significantly. During the testing process in an experiment, a 95% significance level is accepted, and the remaining 5% can be neglected. The significance level also tells the critical or threshold value.
For e.g., in an experiment, if the significance level is set to 98%, then the

critical value is 0.02%.

The decision about your model:

A p-value measures the strength of evidence in support of a null hypothesis. If the p-value is less than the significance level, we reject the null hypothesis.

If the p-value $< \alpha$, then we have statistically significant evidence against the null hypothesis, so we reject the null hypothesis and accept the alternate hypothesis.

If the p-value $> \alpha$, then we do not have statistically significant evidence against the null hypothesis, so we fail to reject the null hypothesis.

Feature Selection Techniques

Feature Selection is a way of selecting a subset of most relevent features from original features, also we try to remove less relevent features, noisy features, redundant features.

Need of Feature Selection:

1. To drop less important features.
2. To drop noisy features.

Benefits of Feature Selection:

1. It improves accuracy.
2. It takes less computational time(it reduces training and testing time).
3. It helps to reduce overfitting.
4. It helps in avoiding curse of dimensionality.

Feature selection techniques:

1. **Filter Method:**(It is used before training)
    (a) **Correlation:**
        i. Pearson correlation coefficient(continuous vs continuous)
        ii. Spearmans rank correlation coefficient(continuous vs continuous) :It works better when we don't have linear relationship.
        iii. Kendall(tau) correlation coefficient(categorical vs categorical): (1) It's a non-parametric test that is used to measure the degree of association between two variables.
        (2) It is best suited for discrete(categorical).
        (3) Kendall is proffered than Spearman because of more robustness(smaller gross error sensitivity(GES) and more efficient(smaller asymptotic variance)(AV)
    (b) Mutual Information(information gain)
    (c) Fishers Score(categorical)
    (d) Chi-Square Test(categorical)
    (e) Missing Value Ratio
    (f) Variance Threshold Method
    (g) Annova Test
    (h) Mean Absolute Difference
    (i) Variance Inflation Factor
2. **Wrapper Method:**(Used with training) it uses machine learning algorithms to find best subset of features.

   (a) Forward Feature Selection
   (b) Backward Feature Selection
   (c) Recursive Feature Elimination
   (d) Exhaustive Feature Selection

3. **Embedded Method:**(Used after training)
   (a) Regularization(Lasso Regression)
   (b) Random Forest Feature Importance
   (c) AdaBoost Feature Importance
   (d) Tree Based model Feature Importance

Note:

1. **Pearson:**
   - Variables should be continuous
   - Should be normally distributed
   - Relation should be linear
   - Outliers needs to handled
2. **Spearman:**
   - Works well on ordinal data
   - Variables should be continuous
   - Works well on monotonic and non-linear
   - Determines strength and direction of monotonic relation
   - Little bit of outliers can be cope up /over viewed
3. **Kendall:**
   - When data doesn't meet Pearson's requirement

- It’s non parametric, so no need of normally distributed data
- Does not require continuous Data
- Suited for categorical variable

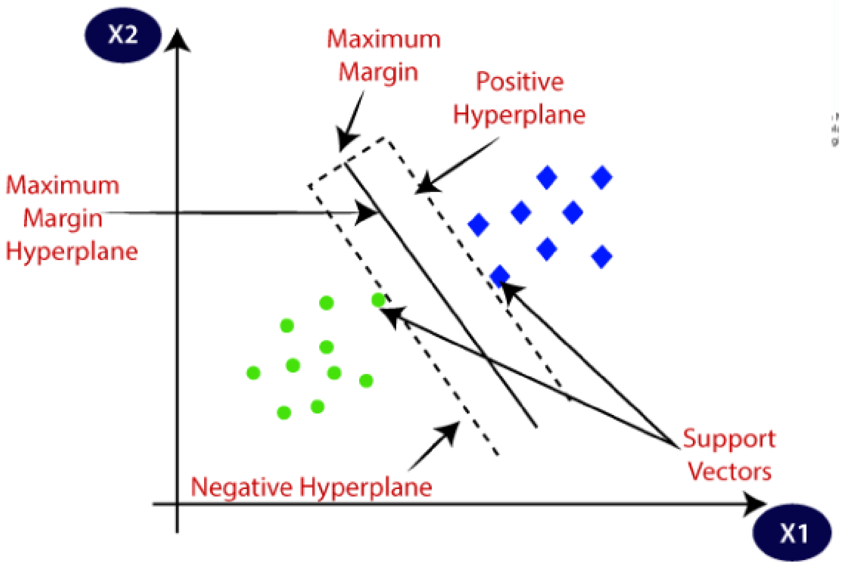
Support Vector Machine Algorithm

Support Vector Machine or SVM is one of the most popular Supervised Learning algorithms, which is used for Classification as well as Regression problems. However, primarily, it is used for Classification problems in Machine Learning.

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. Consider the below diagram in which there are two different categories that are classified using a decision boundary or hyperplane:

Types of SVM:



SVM can be of two types:

Linear SVM: Linear SVM is used for linearly separable data, which means if a dataset can be classified into two classes by using a single straight line, then such data is termed as linearly separable data, and classifier is used called as Linear SVM classifier.

Non-linear SVM: Non-Linear SVM is used for non-linearly separated data, which means if a dataset cannot be classified by using a straight line, then such data is termed as non-linear data and classifier used is called as Non-linear SVM classifier.

Hyperplane and Support Vectors in the SVM algorithm:
Hyperplane: There can be multiple lines/decision boundaries to segregate the classes in n-dimensional space, but we need to find out the best decision boundary that helps to classify the data points. This best boundary is known as the hyperplane of SVM.

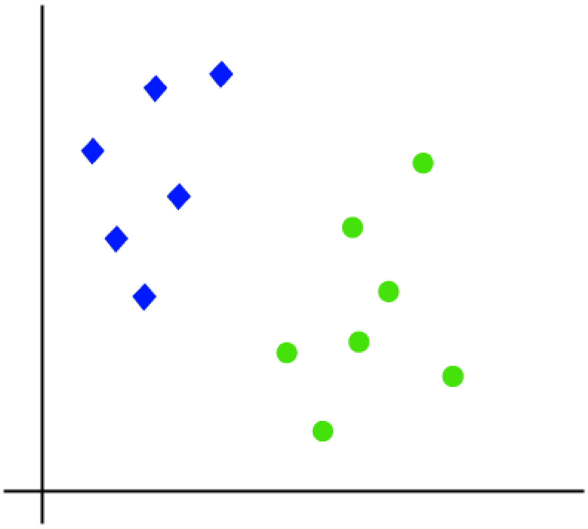
The dimensions of the hyperplane depend on the features present in the dataset, which means if there are 2 features (as shown in image), then hyperplane will be a straight line. And if there are 3 features, then hyperplane will be a 2-dimension plane.

We always create a hyperplane that has a maximum margin, which means the maximum distance between the data points.

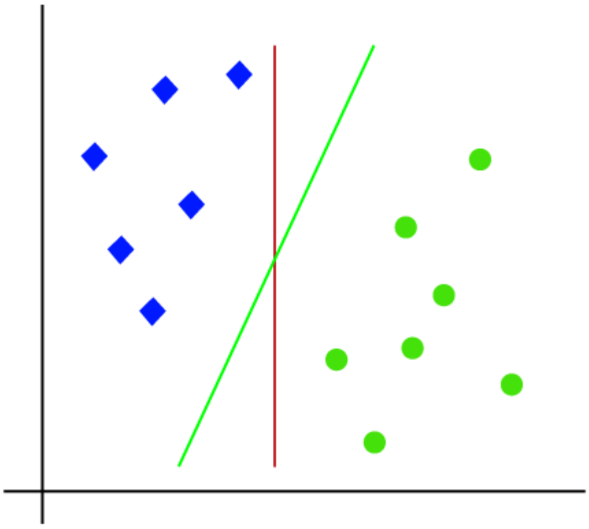
Support Vectors: The data points or vectors that are the closest to the hyperplane and which affect the position of the hyperplane are termed as Support Vector. Since these vectors support the hyperplane, hence called a Support vector.

Working of linear SVM:

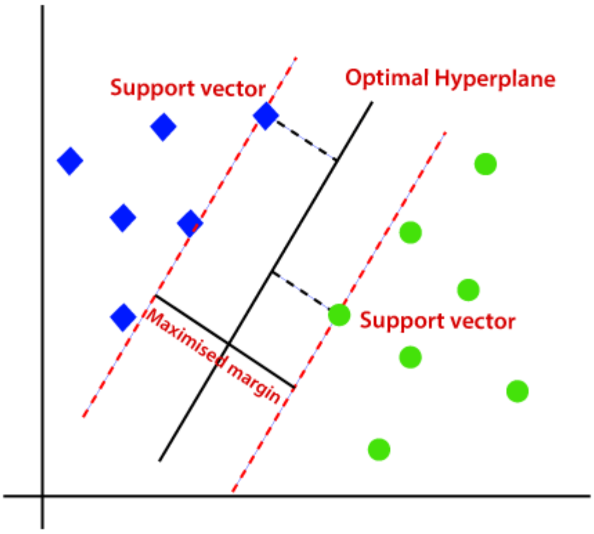
The working of the SVM algorithm can be understood by using an example. Suppose we have a dataset that has two tags (green and blue), and the dataset has two features x1 and x2. We want a classifier that can classify the pair(x1, x2) of coordinates in either green or blue. Consider the below image:



So as it is 2-d space so by just using a straight line, we can easily separate these two classes. But there can be multiple lines that can separate these classes. Consider the below image:

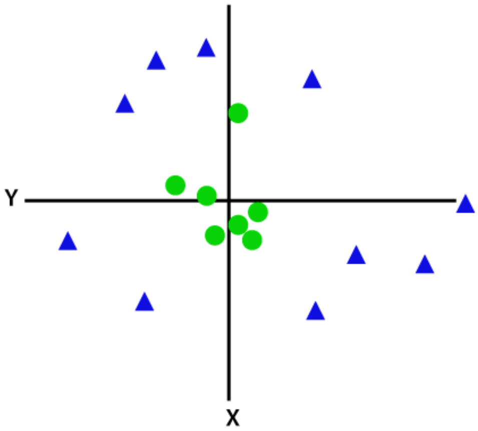


Hence, the SVM algorithm helps to find the best line or decision boundary; this best boundary or region is called as a hyperplane. SVM algorithm finds the closest point of the lines from both the classes. These points are called support vectors. The distance between the vectors and the hyperplane is called as margin. And the goal of SVM is to maximize this margin. The hyperplane with maximum margin is called the optimal hyperplane.



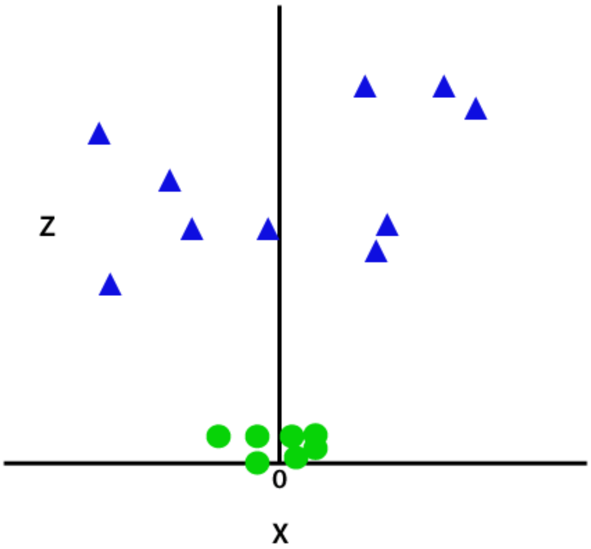
Working of non-linear SVM:

If data is linearly arranged, then we can separate it by using a straight line, but for non-linear data, we cannot draw a single straight line. Consider the below image:

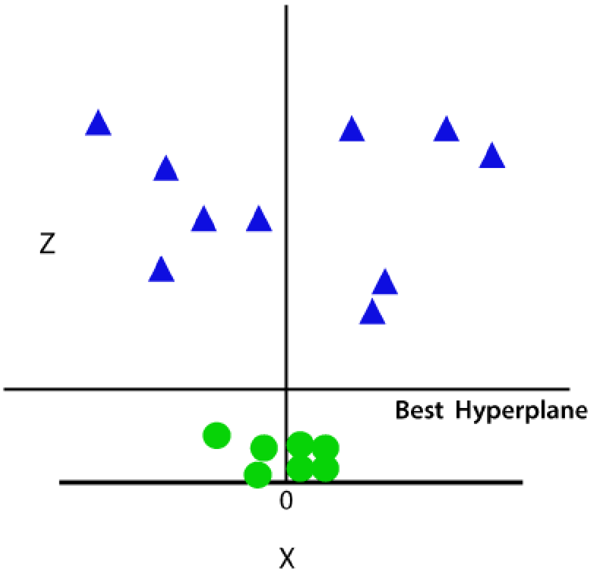


So to separate these data points, we need to add one more dimension. For linear data, we have used two dimensions x and y, so for non-linear data, we will add a third dimension z. It can be calculated as: $z = x^2 + y^2$

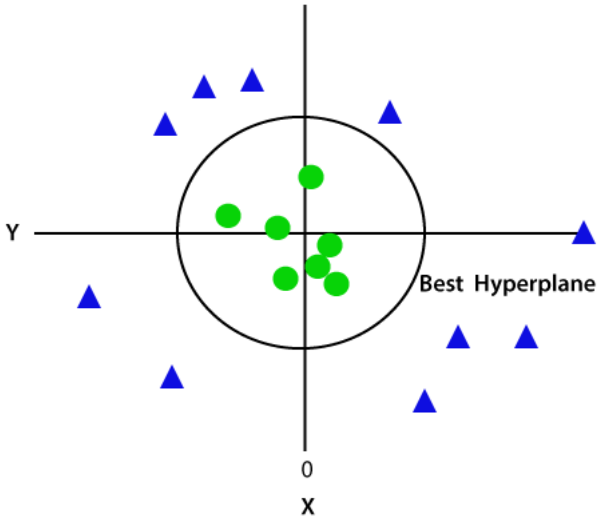
By adding the third dimension, the sample space will become as below image:



So now, SVM will divide the datasets into classes in the following way. Consider the below image:



Since we are in 3-d Space, hence it is looking like a plane parallel to the x-axis. If we convert it in 2d space with $z = 1$, then it will become as:



Hence we get a circumference of radius 1 in case of non-linear data.