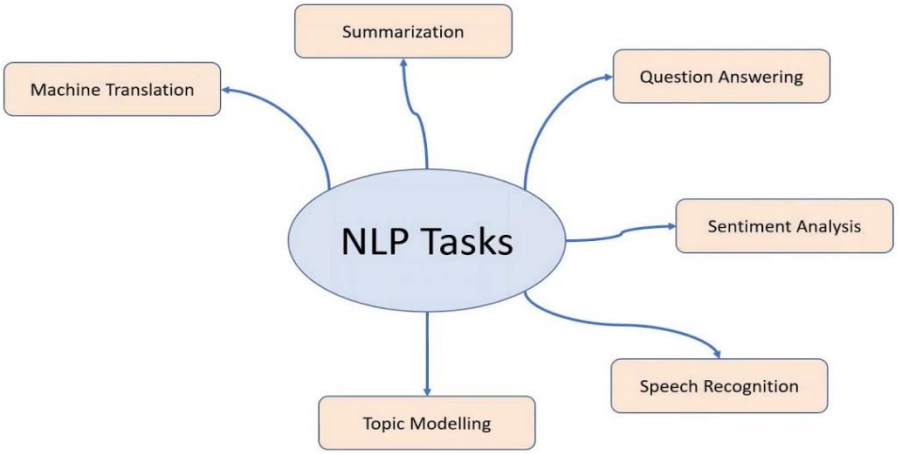
Module – 1 (Introduction to NLP)

NLP Tasks and Applications, Language-Building Blocks, Challenges of NLP, Machine Learning for NLP – Naïve Bayes Classifier, Logistic Regression, Support Vector Machines, Approaches to NLP-- Heuristics-Based NLP, Machine Learning-based NLP.

NLP Tasks and Applications

Natural language processing (NLP) refers to the branch of computer science—and more specifically, the branch of artificial intelligence or AI—concerned with giving computers the ability to understand text and spoken words in much the same way human beings can.



There are several NLP tasks that are commonly performed, including:

1. Text classification: This involves assigning categories or labels to text data based on their content. Examples of text classification tasks include sentiment analysis, topic classification, and spam detection.
2. Named Entity Recognition (NER): This task involves identifying and extracting entities such as names, locations, and organizations from text data.
3. Text summarization: This involves creating a shorter version of a text by identifying and extracting the most important information from the original text.
4. Sentiment Analysis: This involves analyzing the sentiment or emotion conveyed in a piece of text. The sentiment can be positive, negative or neutral.
5. Machine Translation: This involves translating text from one language to another.
6. Information Retrieval: This involves finding relevant documents or information based on a query or set of keywords.
7. Text Generation: This involves creating new text content based on a given prompt or set of input parameters.
8. Speech recognition: This involves converting spoken language into text.

9. Language modeling: This involves predicting the likelihood of a sequence of words in a language, which is used in applications like auto-complete or spell checking.
10. Question Answering: This involves automatically answering questions posed in natural language, such as those commonly encountered in search engines and customer support.

Applications of Natural Language Processing

1. Chatbots

Chatbots are a form of artificial intelligence that are programmed to interact with humans in such a way that they sound like humans themselves. Depending on the complexity of the chatbots, they can either just respond to specific keywords or they can even hold full conversations that make it tough to distinguish them from humans. Chatbots are created using Natural Language Processing and Machine Learning, which means that they understand the complexities of the English language and find the actual meaning of the sentence and they also learn from their conversations with humans and become better with time.

2. Autocomplete in Search Engines

Have you noticed that search engines tend to guess what you are typing and automatically complete your sentences? For example, On typing "game" in Google, you may get further suggestions for "game of thrones", "game of life" or if you are interested in maths then "game theory". All these suggestions are provided using autocomplete that uses Natural Language Processing to guess what you want to ask

3. Voice Assistants

These days voice assistants are all the rage! Whether its Siri, Alexa, or Google Assistant, almost everyone uses one of these to make calls, place reminders, schedule meetings, set alarms, surf the internet, etc. These voice assistants have made life much easier. But how do they work? They use a complex combination of speech recognition, natural language understanding, and natural language processing to understand what humans are saying and then act on it.

4. Language Translator

Want to translate a text from English to Hindi but don't know Hindi? Well, Google Translate is the tool for you! While it's not exactly 100% accurate, it is still a great tool to convert text from one language to another. Google Translate and other translation tools as well as use Sequence to sequence modeling that is a technique in Natural Language Processing. It allows the algorithm to convert a sequence of words from one language to another which is translation.

5. Sentiment Analysis

Almost all the world is on social media these days! And companies can use sentiment analysis to understand how a particular type of user feels about a particular topic, product, etc. They can use natural language processing, computational linguistics, text analysis, etc. to understand the general sentiment of the users for their products and services and find out if the sentiment is good, bad, or neutral. Companies

can use sentiment analysis in a lot of ways such as to find out the emotions of their target audience, to understand product reviews, to gauge their brand sentiment, etc. And not just private companies, even governments use sentiment analysis to find popular opinion and also catch out any threats to the security of the nation.

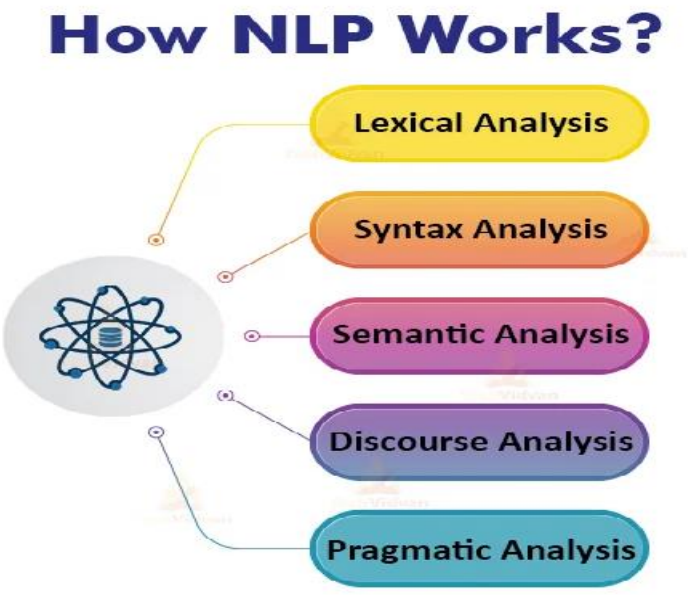
6. Grammar Checkers

Grammar and spelling is a very important factor while writing professional reports for your superiors even assignments for your lecturers. After all, having major errors may get you fired or failed! That's why grammar and spell checkers are a very important tool for any professional writer. They can not only correct grammar and check spellings but also suggest better synonyms and improve the overall readability of your content. And guess what, they utilize natural language processing to provide the best possible piece of writing!. Some of the most popular grammar checkers that use NLP include Grammarly, WhiteSmoke, ProWritingAid, etc.

7. Email Classification and Filtering

Emails are still the most important method for professional communication. However, all of us still get thousands of promotional Emails that we don't want to read. Thankfully, our emails are automatically divided into 3 sections namely, Primary, Social, and Promotions which means we never have to open the Promotional section! But how does this work? Email services use natural language processing to identify the contents of each Email with text classification so that it can be put in the correct section.

Language-Building blocks



Five main Component of Natural Language processing in AI are:

- Morphological and Lexical Analysis
- Syntactic Analysis
- Semantic Analysis
- Discourse Integration
- Pragmatic Analysis

Morphological and Lexical Analysis

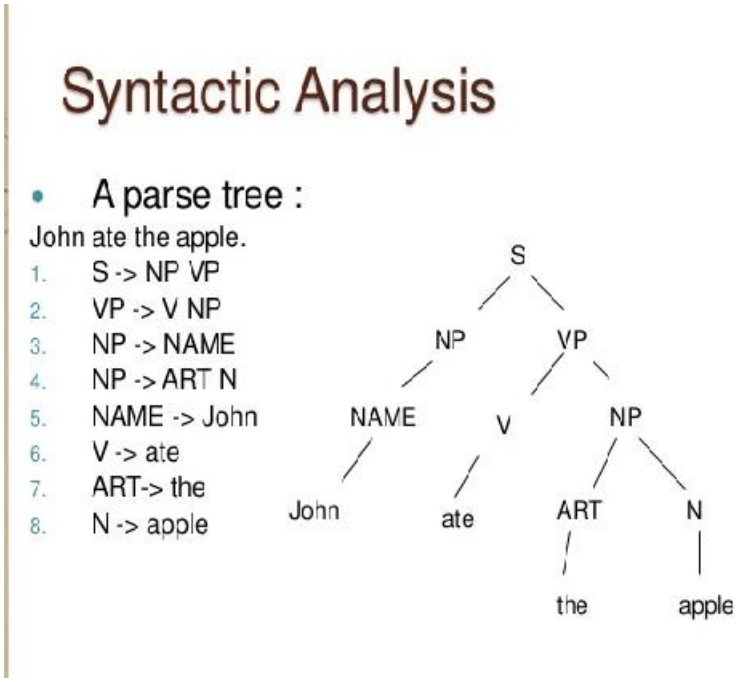
Lexical analysis is a vocabulary that includes its words and expressions. It depicts analyzing, identifying and description of the structure of words. It includes dividing a text into paragraphs, words and the sentences.

Individual words are analyzed into their components, and nonword tokens such as punctuations are separated from the words.

Syntax analysis

The words are commonly accepted as being the smallest units of syntax. The syntax refers to the principles and rules that govern the sentence structure of any individual languages.

Syntax focus about the proper ordering of words which can affect its meaning. This involves analysis of the words in a sentence by following the grammatical structure of the sentence. The words are transformed into the structure to show hows the word are related to each other.



Semantic Analysis

Semantic Analysis is a structure created by the syntactic analyzer which assigns meanings. This component transfers linear sequences of words into structures. It shows how the words are associated with each other.

Semantics focuses only on the literal meaning of words, phrases, and sentences. This only abstracts the dictionary meaning or the real meaning from the given context. The structures assigned by the syntactic analyzer always have assigned meaning

E.g.. “colorless green idea.” This would be rejected by the Symantec analysis as colorless Here; green doesn’t make any sense.

Discourse Integration

It means a sense of the context. The meaning of any single sentence which depends upon that sentences. It also considers the meaning of the following sentence.

For example, the word "that" in the sentence "He wanted that" depends upon the prior discourse context.

Next in this NLP tutorial, we will learn about NLP and writing systems.

Pragmatic Analysis

Pragmatic Analysis deals with the overall communicative and social content and its effect on interpretation. It means abstracting or deriving the meaningful use of language in situations. In this analysis, the main focus always on what was said in reinterpreted on what is meant.

Pragmatic analysis helps users to discover this intended effect by applying a set of rules that characterize cooperative dialogues.

E.g., "close the window?" should be interpreted as a request instead of an order.

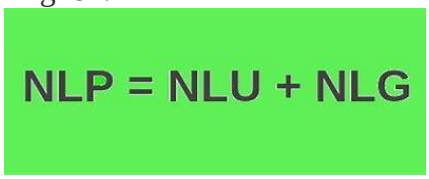
OVERVIEW

In NLP, language-building blocks are the basic components that help machines understand human language. These include:

1. Phonetics and phonology: Phonetics deals with the physical properties of speech sounds, while phonology focuses on the systematic organization of those sounds in a particular language. These building blocks help machines understand the sounds of human language.
2. Morphology: Morphology is the study of the internal structure of words and the rules that govern word formation. It helps machines understand the meaning of words by breaking them down into their constituent parts.
3. Syntax: Syntax deals with the arrangement of words and phrases to create meaningful sentences. It helps machines understand the relationships between words in a sentence and how they contribute to the overall meaning.
4. Semantics: Semantics is the study of the meaning of words and sentences. It helps machines understand the meaning of words in context and how they relate to each other to create meaningful sentences.
5. Pragmatics: Pragmatics is the study of how language is used in context. It helps machines understand the social and cultural aspects of language use, such as politeness, sarcasm, and humor.
6. Discourse analysis: Discourse analysis deals with the structure and meaning of extended texts, such as conversations, speeches, and written documents. It helps machines understand how ideas are connected and organized in longer texts.

Natural Language Understanding(NLU) is an area of artificial intelligence to process input data provided by the user in natural language say text data or speech data. It is a way that enables interaction between a computer and a human in a way like humans do using natural languages like English, French, Hindi etc.

Natural Language Generation(NLG) is a sub-component of Natural language processing that helps in generating the output in a natural language based on the input provided by the user. This component responds to the user in the same language in which the input was provided say the user asks something in English then the system will return the output in English.



Challenges of NLP

some of the most significant challenges of NLP:

1. Ambiguity: Natural language is often ambiguous, and it can be challenging for computers to understand the intended meaning of a sentence, especially when there are multiple possible interpretations.
2. Context: The meaning of a word or phrase can depend on the context in which it is used. For example, the word "bank" can refer to a financial institution or the edge of a river. Understanding the context is crucial for accurate NLP.
3. Idioms and colloquialisms: Natural language is full of idioms and colloquialisms that can be difficult for computers to understand. For example, the phrase "kick the bucket" means "to die," but a computer might interpret it literally.
4. Sarcasm and irony: Sarcasm and irony are often used in natural language, but they can be challenging for computers to detect and understand.
5. Data quality: NLP models require large amounts of high-quality training data to perform well. However, the data used to train these models is often biased, incomplete, or outdated, which can affect their performance.
6. Multilingualism: NLP models must be able to process text in multiple languages, which adds an additional layer of complexity.
7. Privacy and ethical concerns: NLP technology has the potential to extract personal and sensitive information from text data, raising concerns about privacy and ethical use.
8. Continual learning: NLP models need to be updated continuously to adapt to new languages, new words, and new ways of using language, which requires ongoing research and development.

Different ambiguities are

- Lexical Ambiguity - When words have more than one meaning .
- 2. Syntactic Ambiguity - When sequence of words or a sentence has more than one meaning.
- 3. Referential Ambiguity - When the subject is pointed more than once in a sentence.
- 4. Anaphoric Ambiguity - A phrase or word refers to something previously mentioned, but there is more than one possibility. ...

Naïve Bayes Classifier

Naïve Bayes classifier is a supervised machine learning algorithm, which is used for classification tasks, like text classification.

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

- **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.
- **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.

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.

Problem: If the weather is sunny, then the Player should play or not?

Solution: To solve this, first consider the below dataset:

	Outlook	Play
0	Rainy	Yes
1	Sunny	Yes
2	Overcast	Yes
3	Overcast	Yes
4	Sunny	No
5	Rainy	Yes
6	Sunny	Yes
7	Overcast	Yes
8	Rainy	No
9	Sunny	No
10	Sunny	Yes
11	Rainy	No
12	Overcast	Yes
13	Overcast	Yes

Frequency table for the Weather Conditions:

Weather	Yes	No
Overcast	5	0
Rainy	2	2
Sunny	3	2
Total	10	5

Likelihood table weather condition:

Weather	No	Yes	
Overcast	0	5	5/14= 0.35

Rainy	2	2	4/14=0.29
Sunny	2	3	5/14=0.35
All	4/14=0.29	10/14=0.71	

Applying Bayes'theorem:

P(Yes|Sunny)= P(Sunny|Yes)*P(Yes)/P(Sunny)

P(Sunny|Yes)= 3/10= 0.3
P(Sunny)= 0.35
P(Yes)=0.71
So P(Yes|Sunny) = 0.3*0.71/0.35= **0.60**

P(No|Sunny)= P(Sunny|No)*P(No)/P(Sunny)

P(Sunny|NO)= 2/4=0.5
P(No)= 0.29
P(Sunny)= 0.35
So P(No|Sunny)= 0.5*0.29/0.35 = **0.41**

So as we can see from the above calculation that **P(Yes|Sunny)>P(No|Sunny)**

Hence on a Sunny day, Player can play the game.

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

Applications of Naïve Bayes Classifier:

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

Logistic Regression

- 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.
- 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**.

- 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**.
- In Logistic regression, instead of fitting a regression line, we fit an "S" shaped logistic function, which predicts two maximum values (0 or 1).
- The curve from the logistic function indicates the likelihood of something such as whether the cells are cancerous or not, a mouse is obese or not based on its weight, etc.
- 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 Regression can be used to classify the observations using different types of data and can easily determine the most effective variables used for the classification. The below image is showing the logistic function:
- Logistic regression is a type of regression that predicts the probability of a **binary outcome**, such as yes or no, spam or not spam, or crack or no crack.

Logistic Function (Sigmoid Function):

- 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.
- 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.
- 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.

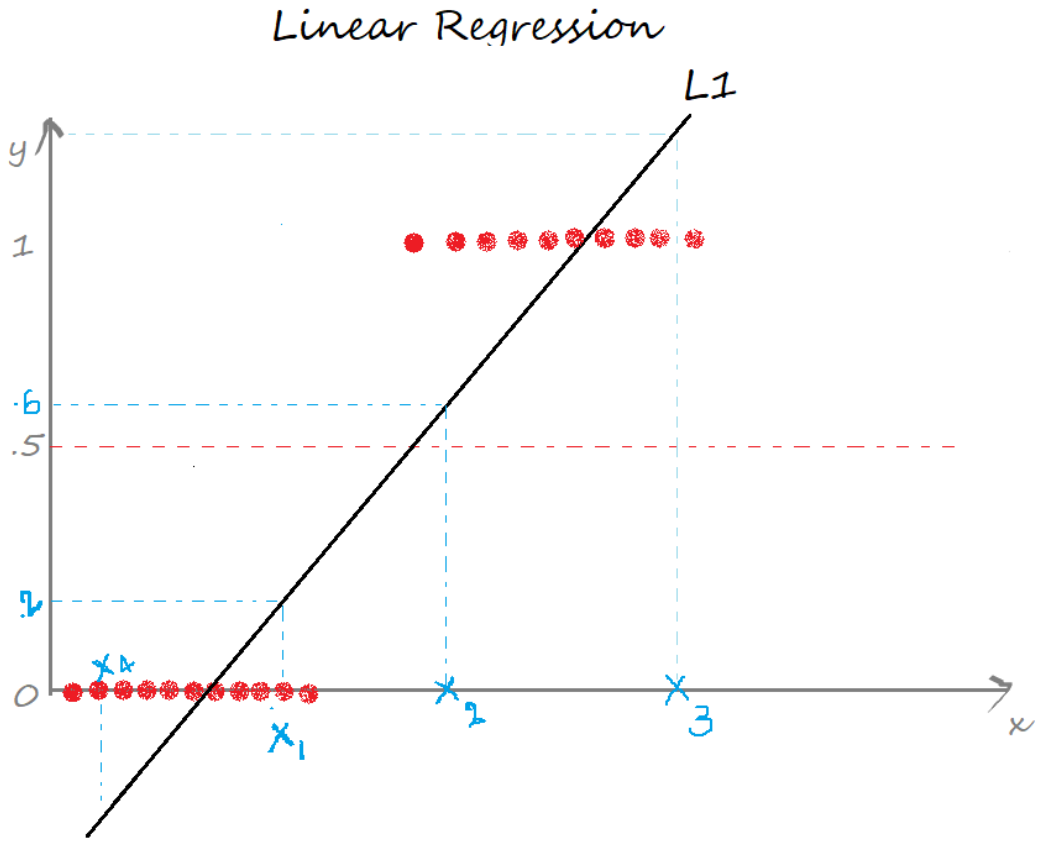
If we can use linear regression to solve a binary class classification problem. Assume we have a dataset that is linearly separable and has the output that is discrete in two classes (0, 1).

In Linear regression, we draw a straight line(the best fit line) L1 such that the sum of distances of all the data points to the line is minimal. The equation of the line L1 is y=mx+c, where m is the slope and c is the y-intercept.

We define a threshold T = 0.5, above which the output belongs to class 1 and class 0 otherwise.

$$y = mx + c, \text{ Threshold } T = 0.5$$

$$y = \begin{cases} 1, & mx+c \geq 0.5 \\ 0, & mx+c < 0.5 \end{cases}$$



Case 1: the predicted value for x1 is ≈0.2 which is less than the threshold, so x1 belongs to class 0.

Case 2: the predicted value for the point x2 is ≈0.6 which is greater than the threshold, so x2 belongs to class 1.

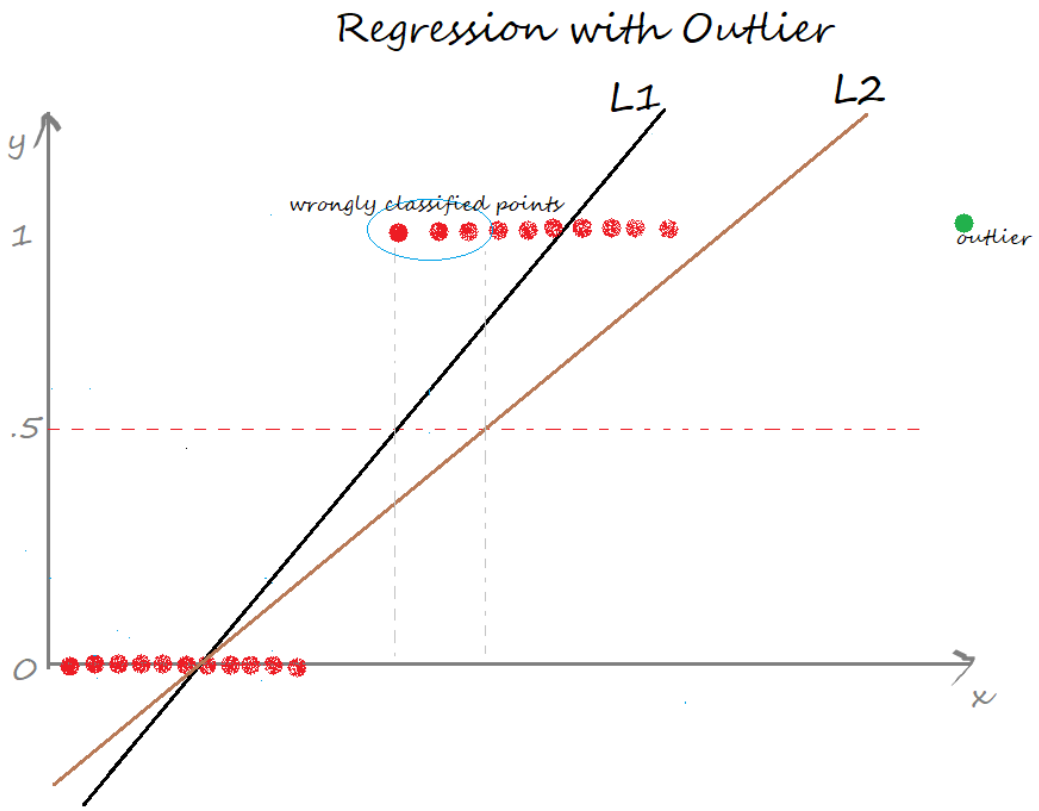
So far so good, yeah!

Case 3: the predicted value for the point x3 is beyond 1.

Case 4: the predicted value for the point x4 is below 0.

The predicted values for the points x3, x4 exceed the range (0,1) which doesn't make sense because the probability values always lie between 0 and 1. And our output can have only two values either 0 or 1. Hence, this is a problem with the linear regression model.

Now, introduce an outlier and see what happens. The regression line gets deviated to keep the distance of all the data points to the line to be minimal.



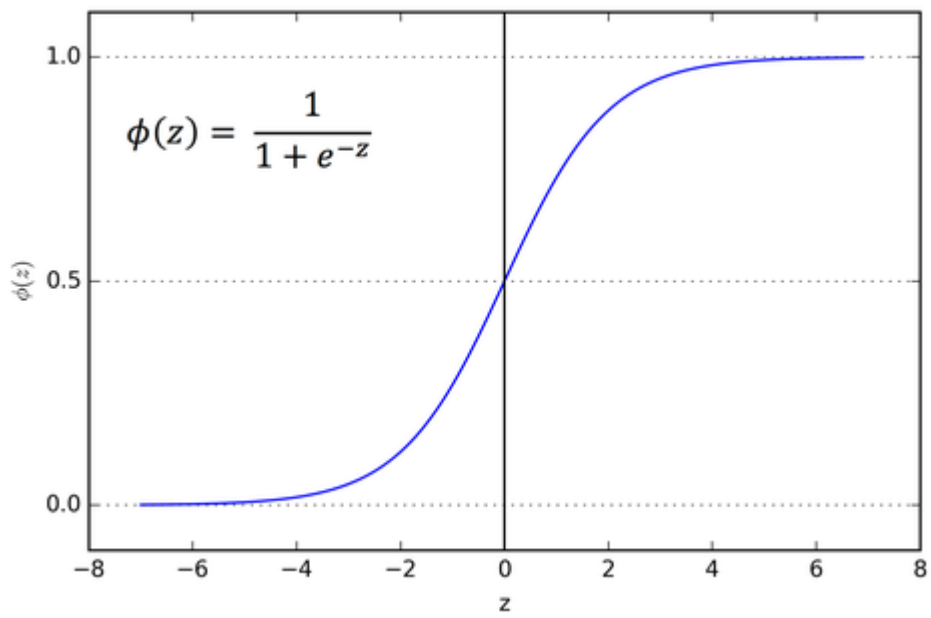
L2 is the new best-fit line after the addition of an outlier. Seems good till now. But the problem is, if we closely observe, some of the data points are wrongly cssified.

The two limitations of using a linear regression model for classification problems are:

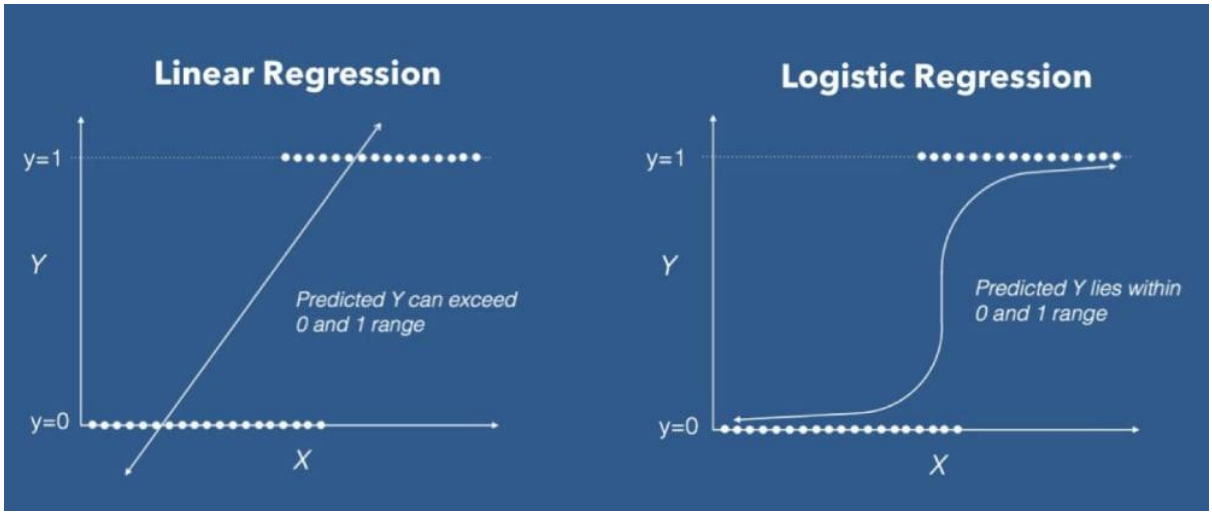
- the predicted value may exceed the range (0,1)
- error rate increases if the data has outliers
- The sigmoid function is useful to map any predicted values of probabilities into another value between 0 and 1.

Sigmoid function: $\sigma(z) = 1/(1+e^{-z})$

unlike linear regression, we get an 'S' shaped curve in logistic regression.



Produced values between 1 and 0



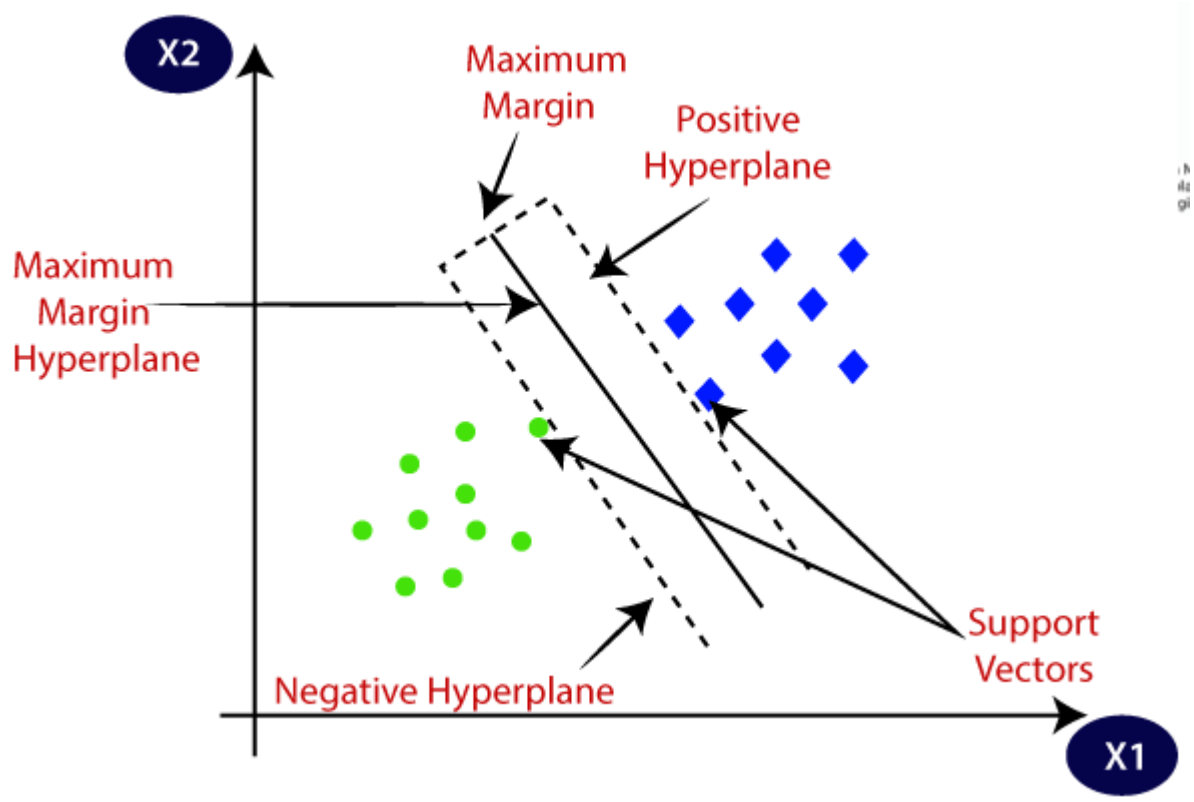
Will produce>1 values

Support Vector Machine(SVM)

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:



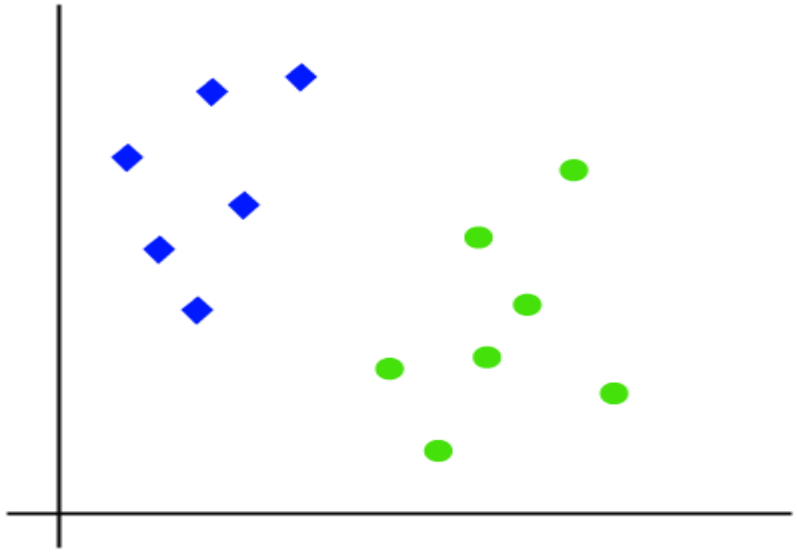
SVM can be of two types:

- **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.

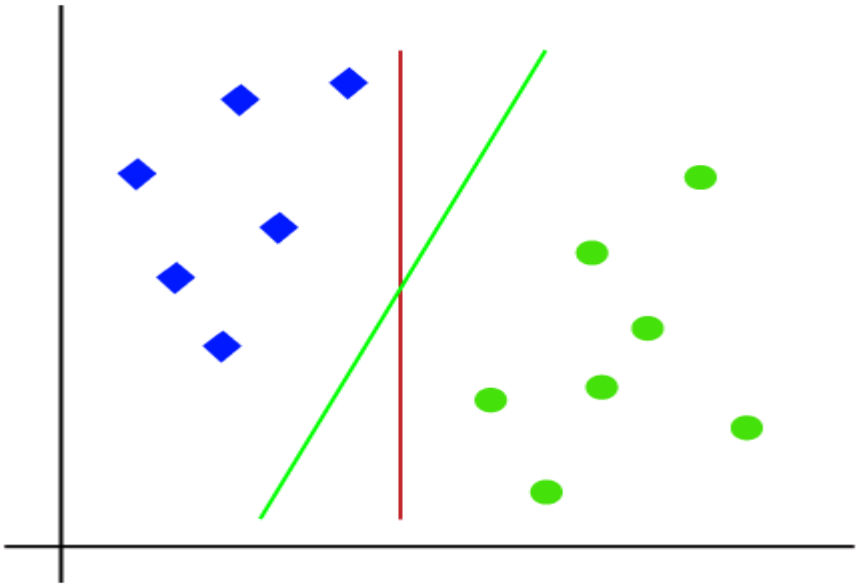
Hyperplane: There can be multiple lines/decision boundaries to segregate the classes in n-dimensional space, but we need to find out the best decision boundary that helps to classify the data points. This best boundary is known as the hyperplane of SVM.The dimensions of the hyperplane depend on the features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

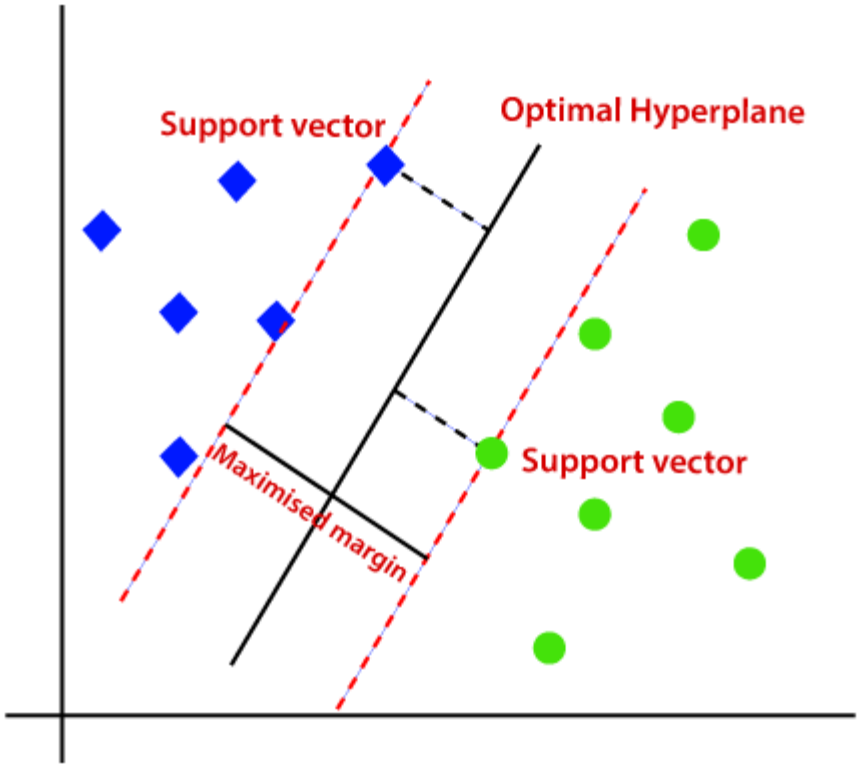
Support Vectors: The data points or vectors that are the closest to the hyperplane and which affect the position of the hyperplane are termed as Support Vector. Since these vectors support the hyperplane, hence called a Support vector.

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:

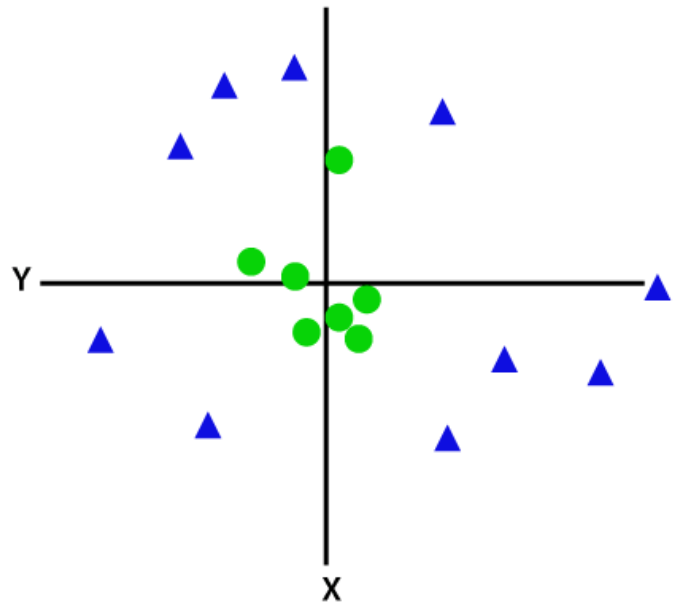


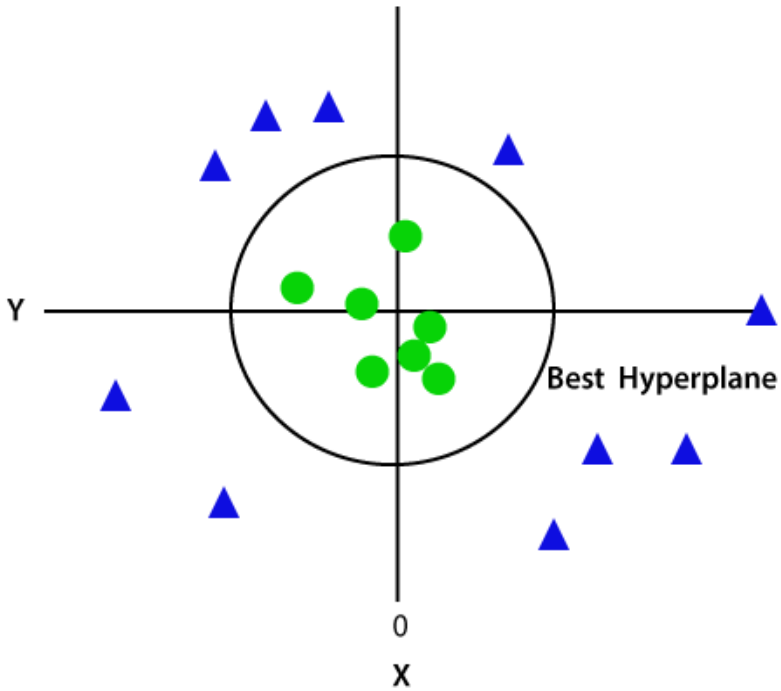


Hence, the SVM algorithm helps to find the best line or decision boundary; this best boundary or region is called as a **hyperplane**. SVM algorithm finds the closest point of the lines from both the classes. These points are called support vectors. The distance between the vectors and the hyperplane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

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.





Non-Linear SVM is another working principle on SVM that is used for data that cannot be separated linearly because of its high dimensions. Non-linear classification is carried out using the **kernel** concept. The kernel concept in the non-linear case plays a role in determining the classification limits used as a model.

Non-Linear SVM applies the function of the kernel concept to a space that has high dimensions. What is meant by high dimension is that the dataset has more than two features to classify. For example, non-linear classification cases, namely factors that affect human health, consist of age factors, dietary factors, exercise factors, heredity, disease history and stress levels.

The accuracy of the model generated by the process in the SVM algorithm is very dependent on the parameters and **kernel functions** used. In the use of kernel functions in non-linear SVM is something that needs to be considered because the performance of SVM depends on the choice of kernel function.

Non-linier SVM is implemented in practice using a kernel, so it can separate data with the kernel function it called kernel trick**.**

The Kernel Trick

SVM can work well in non-linear data cases using kernel trick. The function of the kernel trick is to map the low-dimensional input space and tranforms into a higher dimensional space.

- **Radial Basis Function Kernel (RBF)**

The RBF kernel is the most widely used kernel concept to solve the problem of classifying datasets that cannot be separated linearly. This kernel is known to have good performance with certain parameters, and the results of the training have a small error value compared to other kernels. The equation formula for the RBF kernel function is:

$$K(x,xi) = \exp(-gamma * sum((x - xi^2))$$

In the RBF kernel function equation, ‖xi-x ‖ is the Euclidean Distance between x1 and x2 in two different feature spaces and σ (sigma) is the RBF kernel parameter that determines the kernel weight.

Polynomial Kernel

A Polynomial Kernel is more generalized form of the linear kernel. In machine learning, the polynomial kernel is a kernel function suitable for use in support vector machines (SVM) and other kernelizations, where the kernel represents the similarity of the training sample vectors in a feature space. Polynomial kernels are also suitable for solving classification problems on normalized training datasets. The equation for the polynomial kernel function is:

$$K(x,xi) = 1 + sum(x * xi)^d$$

- **Sigmoid Kernel**

The concept of the sigmoid kernel is a development of an artificial neural network (ANN) with the equation for the kernel function is:

$$K(x,xi) = \tanh(\alpha xi.xj + \beta)$$

Linear Kernel

A linear kernel can be used as normal dot product any two given observations. The equation for the kernel function is:

$$K(x, xi) = sum(x * xi)$$