NAIVE BAYES CLASSIFICATION ALGORITHM

- Naive Bayes algorithm is a supervised learning algorithm, which is based on Bayes theorem and used for solving classification problems.
- It is mainly used in *text classification* that includes a high-dimensional training dataset.
- Naive 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.
- It is a probabilistic classifier, which means it predicts on the basis of the probability of an object.
- Some popular examples of Naive Bayes Algorithm are spam filtration, Sentimental analysis, and classifying articles.

BAYES THEORM

- 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.

Applications of Naive Bayes Classifier:

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

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.

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	3
Total	10	5

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.

PYTHON IMPLEMENTATION OF THE NAÏVE BAYES ALGORITHM:

Now we will implement a Naive Bayes Algorithm using Python. So, for this we will use the "**user_data**" **dataset**, which we have used in our other classification model. Therefore, we can easily compare the Naive Bayes model with the other models.

Steps to implement:

- Data Pre-processing step
- Fitting Naive Bayes to the Training set
- Predicting the test result
- Test accuracy of the result (Creation of Confusion matrix)
- Visualizing the test set result.