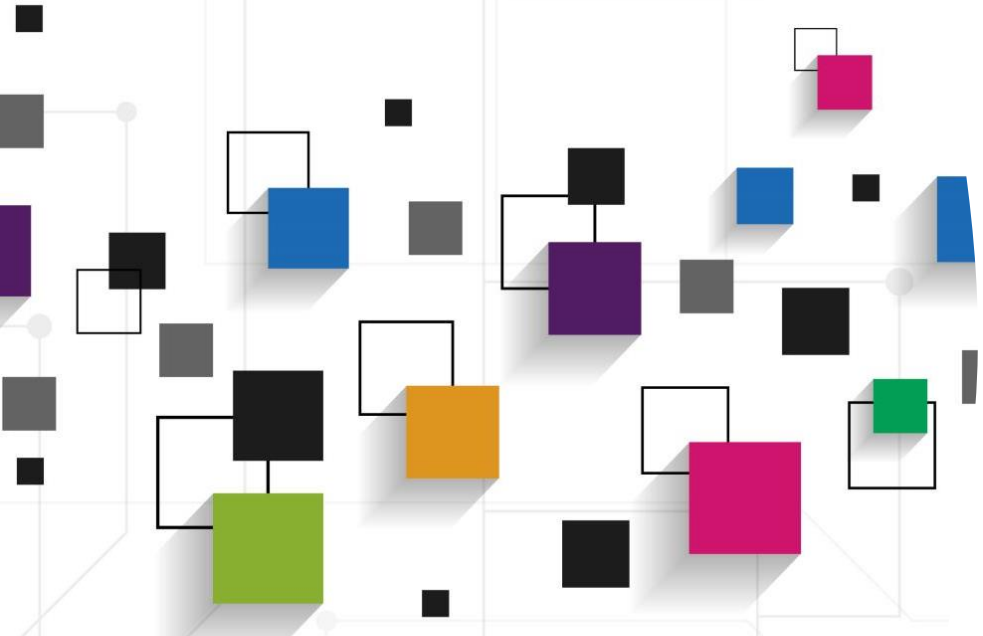
- **Naïve Bayes Classifier Algorithm**

Dr. Jagendra Singh

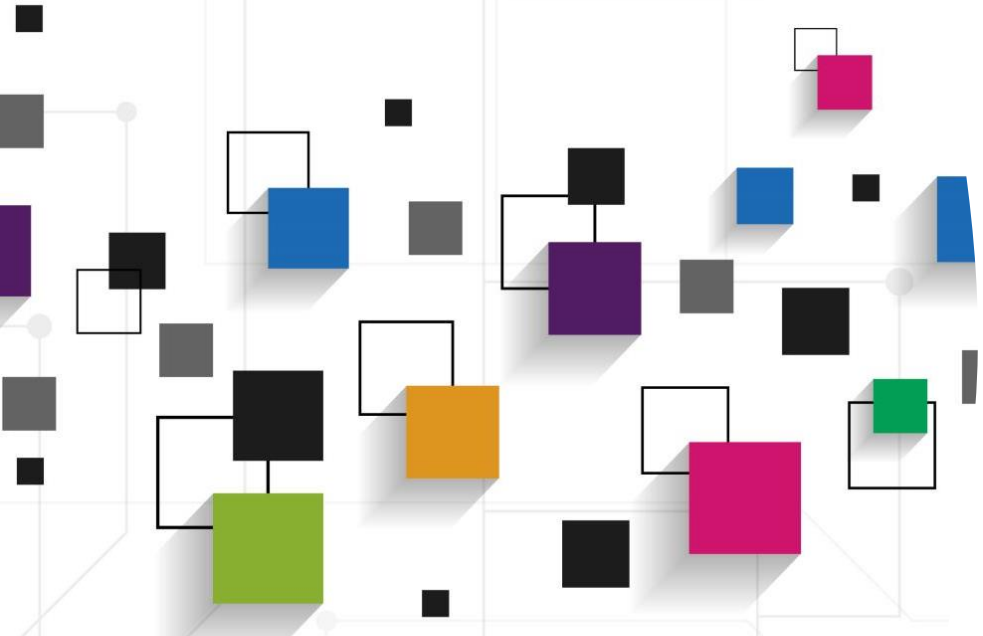
Machine Learning



NAÏVE BAYES CLASSIFIER

- Naïve 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.



NAÏVE BAYES CLASSIFIER

- 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.

NAÏVE BAYES CLASSIFIER

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 CALLED NAÏVE BAYES

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

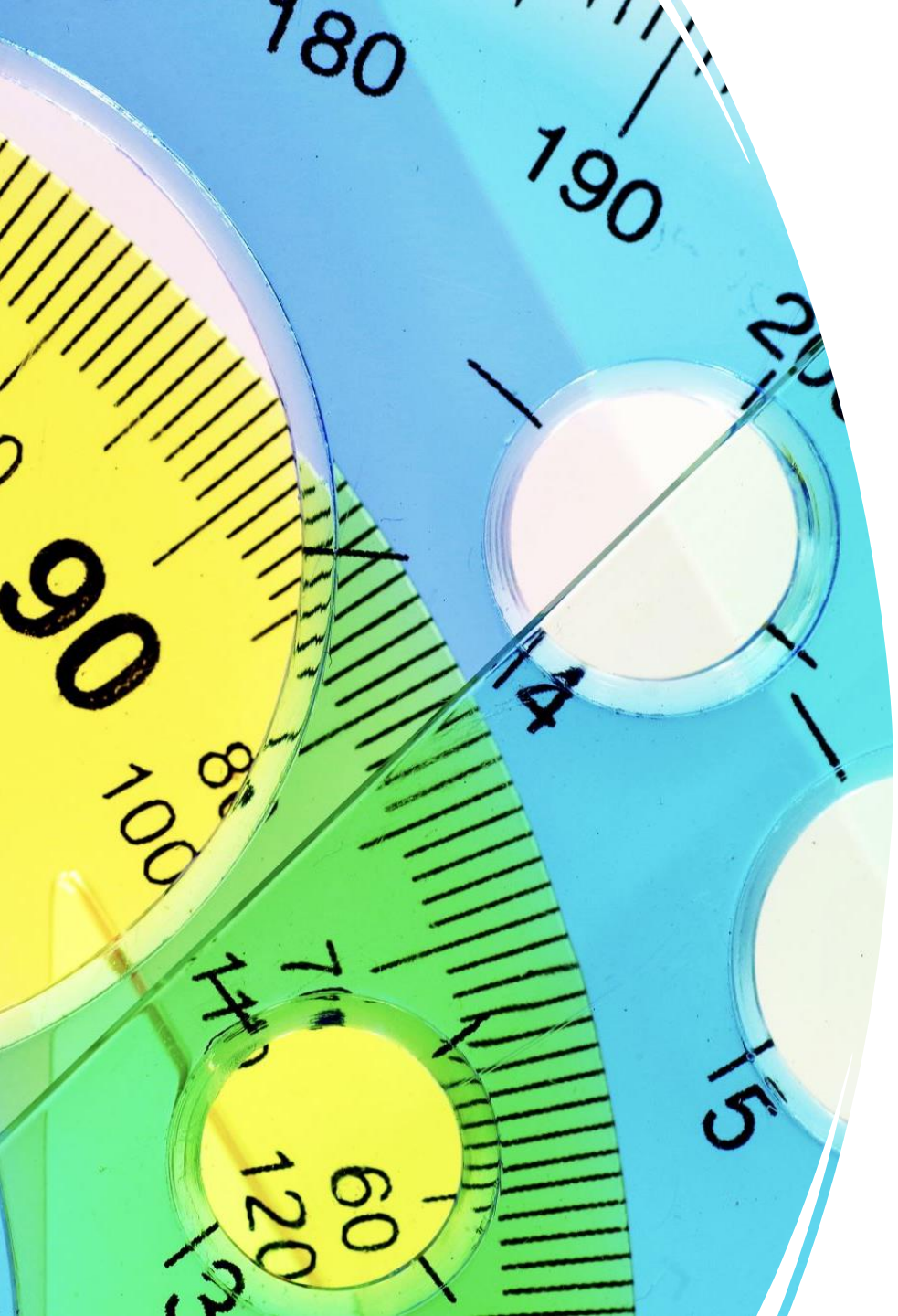
Where,

- **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.

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

NAÏVE BAYES' CLASSIFIER WORKING

- 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:



NAÏVE BAYES' CLASSIFIER WORKING

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.

NAÏVE BAYES' CLASSIFIER WORKING

- **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 BAYE'S 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**

APPLYING BAYE'S THEOREM

- **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.**

ADVANTAGES OF NB CLASSIFIER

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

ADVANTAGES OF NB CLASSIFIER

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

DISADVANTAGES OF NB CLASSIFIER

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

APPLICATIONS OF NB 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**.

TYPES OF NAÏVE BAYES MODEL

There are **three types** of Naive Bayes Model, which are given below: (Gaussian, Multinomial, Bernoulli)

- **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.

TYPES OF NAÏVE BAYES MODEL

- For example, suppose the training data contains a continuous attribute x.
- We first segment the data by the class, and then compute the mean and variance of x in each class.
- Let μi be the mean of the values and let σi be the variance of the values associated with the ith class.
- Suppose we have some observation value xi .
- Then, the probability distribution of xi given a class can be computed by the following equation –

$$p(x_i|y_j) = \frac{1}{\sqrt{2\pi\sigma_j^2}} e^{-\frac{(x_i-\mu_j)^2}{2\sigma_j^2}}$$

TYPES OF NAÏVE BAYES MODEL

- Formula:

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$$

$f(x)$ = probability density function

σ = standard deviation

μ = mean

TYPES OF NAÏVE BAYES MODEL

- **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.

TYPES OF NAÏVE BAYES MODEL

- **Multinomial**: The probability mass function of this multinomial distribution is:

$$f(x_1,\ldots,x_k;n,p_1,\ldots,p_k) = \Pr(X_1 = x_1 \text{ and } \ldots \text{ and } X_k = x_k)$$

$$= \begin{cases} \dfrac{n!}{x_1!\cdots x_k!} p_1^{x_1} \times \cdots \times p_k^{x_k}, & \text{when } \sum_{i=1}^{k} x_i = n \\ 0 & \text{otherwise,} \end{cases}$$

for non-negative integers $x_1, \ldots, x_k$.

The probability mass function can be expressed using the gamma function as:

$$f(x_1,\ldots,x_k;p_1,\ldots,p_k) = \frac{\Gamma(\sum_i x_i + 1)}{\prod_i \Gamma(x_i + 1)} \prod_{i=1}^{k} p_i^{x_i}.$$

This form shows its resemblance to the Dirichlet distribution, which is its conjugate prior.

TYPES OF NAÏVE BAYES MODEL

- **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.

TYPES OF NAÏVE BAYES MODEL

- **Bernoulli**: Formula

$$f(k;p) = pk + (1-p)(1-k)$$

p = probability

k = possible outcomes

f = probability mass function

IMPLEMENTATION OF NB 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.

IMPLEMENTATION OF NB ALGORITHM

- **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.

THANK YOU