Naive Bayes(Numerical Example)

Naive Bayes algorithm is a supervised machine learning algorithm which is based on **Bayes Theorem** used mainly for classification problem.

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 prediction.

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 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.

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

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.

$P(A|B) = (P(B|A) * P(A))/ P(B) == P(A \cap B) / P(B)$

|||ly $P(B|A) = (P(A|B) * P(B))/ P(A) == P(B \cap A) / P(A)$

if $P(A \cap B) == P(B \cap A)$.

then $P(B|A) * P(A) = P(A|B) * P(B)$

Types Of Naive Bayes:

There are three types of Naive Bayes model under the scikit-learn library:

- **Gaussian:** It is used in classification and it assumes that features follow a normal distribution.
- **Multinomial:** It is used for discrete counts. For example, let's say, we have a text classification problem. Here we can consider Bernoulli trials which is one step further and instead of "word occurring in the document", we have "count how often word occurs in the document", you can think of it as "number of times outcome number x_i is observed over the n trials".
- **Bernoulli:** The binomial model is useful if your feature vectors are binary (i.e. zeros and ones). One application would be text classification with 'bag of words' model where the 1s & 0s are "word occurs in the document" and "word does not occur in the document" respectively.

Numerical Example:

Consider the given Dataset ,Apply Naive Baye's Algorithm and Predict that if a fruit has the following properties then which type of the fruit it is
Fruit = {Yellow , Sweet ,long}

Frequency Table:

Fruit	Yellow	Sweet	Long	Total
Mango	350	450	0	650
Banana	400	300	350	400
Others	50	100	50	150
Total	800	850	400	1200

Solution:

P(A|B) = (P(B|A) * P(A))/ P(B)

1. Mango:

P(X | Mango) = P(Ye | Yellow) * P(Sweet | Mango) * P(Long | Mango)

1. a)P(Yellow | Mango) = (P(Mango | Yellow) * P(Yellow))/ P (Mango)

= ((350/800) * (800/1200)) / (650/1200)

P(Yellow | Mango)= 0.53 →1

1.b)P(Sweet | Mango) = (P(Sweet | Mango) * P(Sweet))/ P (Mango)

= ((450/850) * (850/1200)) / (650/1200)

P(Sweet | Mango)= 0.69 → 2

1. c)P(Long | Mango) = (P(Long | Mango) * P(Long))/ P (Mango)

= ((0/650) * (400/1200)) / (800/1200)

P(Long | Mango)= 0 → 3

On multiplying eq 1,2,3 ==> P(X | Mango) = 0.53 * 0.69 * 0

P(X | Mango) = 0

2. Banana:

P(X | Banana) = P(Yellow | Banana) * P(Sweet | Banana) * P(Long | Banana)

2.a) P(Yellow | Banana) = (P(Banana | Yellow) * P(Yellow))/ P (Banana)

= ((400/800) * (800/1200)) / (400/1200)

P(Yellow | Banana) = 1 → 4

2.b) P(Sweet | Banana) = (P(Banana | Sweet) * P(Sweet))/ P (Banana)

= ((300/850) * (850/1200)) / (400/1200)

P(Sweet | Banana) = .75→ 5

2.c)P(Long | Banana) = (P(Banana | Yellow) * P(Long))/ P (Banana)

= ((350/400) * (400/1200)) / (400/1200)

P(Yellow | Banana) = 0.875 → 6

On multiplying eq 4,5,6 ==> P(X | Banana) = 1 * .75 * 0.875

P(X | Banana) = 0.6562

3. Others:

P(X | Others) = P(Yellow | Others) * P(Sweet | Others) * P(Long | Others)

3.a) P(Yellow | Others) = (P(Others| Yellow) * P(Yellow))/ P (Others)

= ((50/800) * (800/1200)) / (150/1200)

P(Yellow | Others) = 0.34→ 7

3.b) P(Sweet | Others) = (P(Others| Sweet) * P(Sweet))/ P (Others)

= ((100/850) * (850/1200)) / (150/1200)

P(Sweet | Others) = 0.67 → 8

3.c) P(Long | Others) = (P(Others| Long) * P(Long))/ P (Others)

= ((50/400) * (400/1200)) / (150/1200)

P(Long | Others) = 0.34 → 9

On multiplying eq 7,8,9 ==> P(X | Others) = 0.34 * 0.67* 0.34

P(X | Others) = 0.07742

So finally from P(X | Mango) == 0 , P(X | Banana) == 0.65 and P(X| Others) == 0.07742.

We can conclude **Fruit{Yellow,Sweet,Long} is Banana.**

To implement a Naive Bayes classifier, we perform three steps.

1. First, we calculate the probability of each class label in the training dataset.
2. Next, we calculate the conditional probability of each attribute of the training data for each class label given in the training data.
3. Finally, we use the Bayes theorem and the calculated probabilities to predict class labels for new data points. For this, we will calculate the probability of the new data point belonging to each class. The class with which we get the maximum probability is assigned to the new data point.

To understand the above steps using a naive Bayes classification numerical example, we will use the following dataset.

Sl. No.	Color	Legs	Height	Smelly
1	White	3	Short	Yes
2	Green	2	Tall	No
3	Green	3	Short	Yes
4	White	3	Short	Yes
5	Green	2	Short	No
6	White	2	Tall	No
7	White	2	Tall	No
8	White	2	Short	Yes

Dataset For Naive Bayes Classification

Using the above data, we have to identify the species of an entity with the following attributes.

X={Color=Green, Legs=2, Height=Tall, Smelly=No}

To predict the class label for the above attribute set, we will first calculate the probability of the species being M or H in total.

P(Species=M)=4/8=0.5
P(Species=H)=4/8=0.5

Next, we will calculate the conditional probability of each attribute value for each class label.

P(Color=White/Species=M)=2/4=0.5
P(Color=White/Species=H)=¾=0.75
P(Color=Green/Species=M)=2/4=0.5
P(Color=Green/Species=H)=¼=0.25
P(Legs=2/Species=M)=1/4=0.25
P(Legs=2/Species=H)=4/4=1
P(Legs=3/Species=M)=3/4=0.75
P(Legs=3/Species=H)=0/4=0
P(Height=Tall/Species=M)=3/4=0.75
P(Height=Tall/Species=H)=2/4=0.5
P(Height=Short/Species=M)=1/4=0.25
P(Height=Short/Species=H)=2/4=0.5
P(Smelly=Yes/Species=M)=3/4=0.75
P(Smelly=Yes/Species=H)=1/4=0.25
P(Smelly=No/Species=M)=1/4=0.25
P(Smelly=No/Species=H)=3/4=0.75

We can tabulate the above calculations in the tables for better visualization.

The conditional probability table for the Color attribute is as follows.

Color	M	H
White	0.5	0.75
Green	0.5	0.25

Conditional Probabilities for Color Attribute

The conditional probability table for the Legs attribute is as follows.

Legs	M	H
2	0.25	1
3	0.75	0

Conditional Probabilities for Legs Attribute

The conditional probability table for the Height attribute is as follows.

Height	M	H
Tall	0.75	0.5

Short	0.25	0.5

Conditional Probabilities for Height Attribute

The conditional probability table for the Smelly attribute is as follows.

Smelly	M	H
Yes	0.75	0.25
No	0.25	0.75

Conditional Probabilities for Smelly Attribute

Now that we have calculated the conditional probabilities, we will use them to calculate the probability of the new attribute set belonging to a single class.

Let us consider X= {Color=Green, Legs=2, Height=Tall, Smelly=No}.

Then, the probability of X belonging to Species M will be as follows.

P(M/X)=P(Species=M)*P(Color=Green/Species=M)*P(Legs=2/Species=M)*P(Height=Tall/Species=M)*P(Smelly=No/Species=M)
=0.5*0.5*0.25*0.75*0.25
=0.0117

Similarly, the probability of X belonging to Species H will be calculated as follows.

P(H/X)=P(Species=H)*P(Color=Green/Species=H)*P(Legs=2/Species=H)*P(Height=Tall/Species=H)*P(Smelly=No/Species=H)
=0.5*0.25*1*0.5*0.75
=0.0468

So, the probability of X belonging to Species M is 0.0117 and that to Species H is 0.0468. Hence, we will assign the entity X with attributes {Color=Green, Legs=2, Height=Tall, Smelly=No} to species H.