Naive Bayes Classifier

Naive Bayes Classifier

Naive Bayes

- 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.

Why it is called as Naive Bayes?

Why is it called Naïve Bayes?

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

- **Naive**: 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.

Why it is called as Naive Bayes?

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.

Types of Naive Bayes?

Multinomial Naive Bayes:

This is mostly used for document classification problem, i.e. whether a document belongs to the category of sports, politics, technology etc.
The features/predictors used by the classifier are the frequency of the words present in the document.

Bernoulli Naive Bayes:

This is similar to the multinomial naive bayes but the predictors are Boolean variables.
The parameters that we use to predict the class variable take up only values yes or no, for example if a word occurs in the text or not.

Gaussian Naive Bayes:

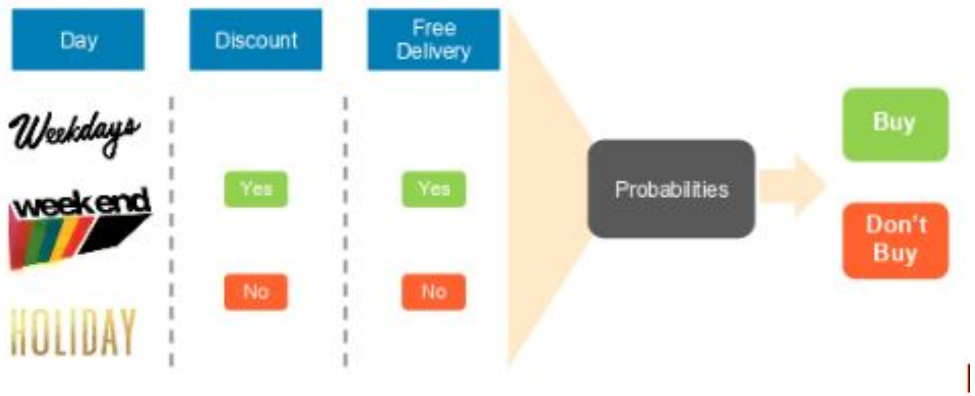
When the predictors take up a continuous value and are not discrete, we assume that these values are sampled from a gaussian distribution.

Working of Naive Bayes

Shopping Example

Problem statement: To predict whether a person will purchase a product on a specific combination of day, discount, and free delivery using a Naive Bayes classifier.

Day	Discount	Free Delivery	Buy
Weekend	No	No	No
Holiday	Yes	Yes	Yes
Weekday	Yes	Yes	Yes
Weekday	Yes	Yes	Yes
Holiday	Yes	Yes	Yes
Weekday	Yes	Yes	Yes
Weekday	Yes	Yes	Yes
Weekday	No	Yes	No
Weekend	Yes	Yes	Yes
Holiday	Yes	Yes	Yes
Holiday	No	No	No
Weekend	Yes	Yes	Yes
Holiday	No	No	No
Weekend	Yes	Yes	Yes
Weekday	Yes	Yes	Yes
Weekend	Yes	Yes	Yes
Weekday	Yes	Yes	Yes
Weekday	Yes	Yes	Yes
Holiday	No	No	Yes
Holiday	No	No	No
Weekday	Yes	Yes	No
Holiday	No	Yes	Yes
Weekend	Yes	Yes	Yes
Holiday	No	Yes	Yes
Weekend	Yes	Yes	Yes
Weekday	Yes	Yes	Yes
Holiday	No	No	Yes
Weekday	Yes	Yes	Yes
Weekend	Yes	Yes	Yes
Holiday	No	No	Yes



For any given day, check if there are a discount and free delivery.

Based on this information, can we predict if a customer would buy the product or not?

Working of Naive Bayes

Steps involved in naive Bayes :

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.

Based on the dataset containing the three input types—day, discount, and free delivery— the frequency table and likelihood table for each attribute is populated.

Frequency Table		Buy	
		Yes	No
Discount	Yes	19	1
	No	5	5

Frequency Table		Buy	
		Yes	No
Free Delivery	Yes	21	2
	No	3	4

Frequency Table		Buy	
		Yes	No
Day	Weekday	9	2
	Weekend	7	1
	Holiday	8	3

Likelihood Table		Buy		
		Yes	No	
Day	Weekday	9/24	2/6	11/30
	Weekend	7/24	1/6	8/30
	Holiday	8/24	3/6	11/30
		24/30	6/30	

Frequency Table		Buy		
		Yes	No	
Discount	Yes	19/24	1/6	20/30
	No	5/24	5/6	10/30
		24/30	6/30	

Frequency Table		Buy		
		Yes	No	
Free Delivery	Yes	21/24	2/6	23/30
	No	3/24	4/6	7/30
		24/30	6/30	

Working of Naive Bayes

The likelihood tables can be used to calculate whether a customer will purchase a product on a specific combination of the day when there is a discount and whether there is free delivery. Consider a combination of the following factors where B equals:

Day = Holiday & Discount = Yes & Free Delivery = Yes

Let us find the probability of them not purchasing based on the conditions above.

A = No Purchase

Applying Bayes Theorem, we get P(A | B) as shown:

$$P(A|B) = P(\text{No Buy} \mid \text{Discount} = \text{Yes}, \text{Free Delivery} = \text{Yes}, \text{Day} = \text{Holiday})$$

$$= \frac{P(\text{Discount} = \text{Yes} \mid \text{No}) * P(\text{Free Delivery} = \text{Yes} \mid \text{No}) * P(\text{Day} = \text{Holiday} \mid \text{No}) * P(\text{No Buy})}{P(\text{Discount=Yes}) * P(\text{Free Delivery=Yes}) * P(\text{Day=Holiday})}$$

$$= \frac{(1/6) * (2/6) * (3/6) * (6/30)}{(20/30) * (23/30) * (11/30)}$$

$$= 0.178$$

A = Purchase

Applying Bayes Theorem, we get P(A | B) as shown:

$$P(A|B) = P(\text{Yes Buy} \mid \text{Discount} = \text{Yes}, \text{Free Delivery} = \text{Yes}, \text{Day} = \text{Holiday})$$

$$= \frac{P(\text{Discount} = \text{Yes} \mid \text{Yes}) * P(\text{Free Delivery} = \text{Yes} \mid \text{Yes}) * P(\text{Day} = \text{Holiday} \mid \text{Yes}) * P(\text{Yes Buy})}{P(\text{Discount=Yes}) * P(\text{Free Delivery=Yes}) * P(\text{Day=Holiday})}$$

$$= \frac{(19/24) * (21/24) * (8/24) * (24/30)}{(20/30) * (23/30) * (11/30)}$$

$$= 0.986$$

Likelihood Table		Buy		
		Yes	No	
Day	Weekday	9/24	2/6	11/30
	Weekend	7/24	1/6	8/30
	Holiday	8/24	3/6	11/30
		24/30	6/30	

Frequency Table		Buy		
		Yes	No	
Discount	Yes	19/24	1/6	20/30
	No	5/24	5/6	10/30
		24/30	6/30	

Frequency Table		Buy		
		Yes	No	
Free Delivery	Yes	21/24	2/6	23/30
	No	3/24	4/6	7/30
		24/30	6/30	

Normalization of Probabilities

From the two calculations above, we find that:

Probability of purchase = 0.986
Probability of no purchase = 0.178

Finally, we have a conditional probability of purchase on this day.

Next, we will normalize these probabilities to get the likelihood of the events:

Sum of probabilities = 0.986 + 0.178 = 1.164

Likelihood of purchase = 0.986 / 1.164 = 0.8471
Likelihood of no purchase = 0.178 / 1.164 = 0.1529

Result: As 0.8471 is greater than 0.1529 , we can conclude that an average customer will buy on holiday with a discount and free delivery.

Laplace Smoothing

In the shopping example above what if a new category "Festival" comes in the test dataset?

In this case the posterior probability for this category will be 0 and ultimately the probability for the entire observation will be 0.
To solve this problem we use **Laplace Smoothing.**

Laplace Smoothing

It is introduced to solve the problem of zero probability i.e. **when a query point contains a new observation, which is not yet seen in training data while calculating probabilities**.

The idea behind Laplace Smoothing: To ensure that our posterior probabilities are never zero, we add 1 to the numerator, and we add k to the denominator.
So, in the case that we don't have a particular ingredient in our training set, the posterior probability comes out to 1 / N + k instead of zero.
Plugging this value into the product doesn't kill our ability to make a prediction as plugging in a zero does.

Using Laplace smoothing, we can represent P(x'|positive) as,

$$P(x'/positive) = \frac{(\text{number of reviews with } x' \text{ and target\_outcome=positive} + \alpha)}{(N+ \alpha * k)}$$

Here, **alpha(α)** represents the smoothing parameter,
K represents the dimensions(no of features) in the data,
N represents the number of observations with target_outcome=positive

Gaussian Naive Bayes

Usual naive bayes works with those predictors which are categorical but **what if the predictors are continuous?**
If the predictors are continuous than we use **Gaussian Naive Bayes.**

Gaussian Naive Bayes classifier

In Gaussian Naive Bayes, continuous values associated with **each feature are assumed to be distributed according to a Gaussian distribution**. A Gaussian distribution is also called Normal distribution.
The likelihood of the features is assumed to be Gaussian, hence, conditional probability is given by:

$$P(x_i \mid y) = \frac{1}{\sqrt{2\pi\sigma_y^2}} \exp\left(-\frac{(x_i - \mu_y)^2}{2\sigma_y^2}\right)$$

This probability is calculated using the distribution of feature with respect to target class. This distribution will be normal distribution.