Naïve Bayes Algorithm

D Sunitha · Follow

7 min read · Jul 3, 2023

What is Naïve Bayes ?

Naïve Bayes is a statistical classification technique based on Bayes Theorem. It is one of the simplest supervised learning algorithms. Naïve Bayes classifier is the fast, accurate and reliable algorithm. Naïve Bayes classifiers have high accuracy and speed on large datasets.

Naïve Bayes classifier assumes that the effect of a particular feature in a class is independent of other features. For example, a loan applicant is desirable or not depending on his/her income, previous loan and transaction history, age, and location. Even if these features are interdependent, these features are still considered independently. This assumption simplifies computation, and that's why it is considered as naïve. This assumption is called class conditional independence.

$$P(h|D) = \frac{P(D|h)P(h)}{P(D)}$$

- P(h): the probability of hypothesis h being true (regardless of the data). This is known as the prior probability of h.
- P(D): the probability of the data (regardless of the hypothesis). This is known as the prior probability.
- P(h|D): the probability of hypothesis h given the data D. This is known as posterior probability.

- P(D|h): the probability of data d given that the hypothesis h was true. This is known as posterior probability.

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.

How Naïve Bayes works:

For example you have 1000 fruits which could be either 'banana', 'orange' or 'other'. These are the 3 possible classes of the Y variable. We have data for the following X variables, all of which are binary (1 or 0).

- Long
- Sweet
- Yellow

The first few rows of the training dataset look like this:

For the sake of computing the probabilities, let's aggregate the training data to form a counts table like this.

Type	Long	Not Long	Sweet	Not Sweet	Yellow	Not Yellow	**Total**
Banana	400	100	350	150	450	50	500
Orange	0	300	150	150	300	0	300
Other	100	100	150	50	50	150	200
Total	500	500	650	350	800	200	1000

So the objective of the classifier is to predict if a given fruit is a 'Banana' or 'Orange' or 'Other' when only the 3 features (long, sweet and yellow) are known.

Let's say you are given a fruit that is: Long, Sweet and Yellow, can you predict what fruit it is?

This is the same of predicting the Y when only the X variables in testing data are known.

Let's solve it by hand using Naïve Bayes. The idea is to compute the 3 probabilities, that is the probability of the fruit being a banana, orange or other. Whichever fruit type gets the highest probability wins.

All the information to calculate these probabilities is present in the above tabulation.

Step 1: Compute the 'Prior' probabilities for each of the class of fruits. That is, the proportion of each fruit class out of all the fruits from the population.

You can provide the 'Priors' from prior information about the population. Otherwise, it can be computed from the training data. For this case, let's compute from the training data. Out of 1000 records in training data, you have 500 Bananas, 300 Oranges and 200 Others.

So the respective priors are 0.5, 0.3 and 0.2.

P(Y=Banana) = 500 / 1000 = 0.50

P(Y=Orange) = 300 / 1000 = 0.30

P(Y=Other) = 200 / 1000 = 0.20

Step 2: Compute the probability of evidence that goes in the denominator. This is nothing but the product of P of Xs for all X. This is an optional step because the denominator is the same for all the classes and so will not affect the probabilities.

P(x1=Long) = 500 / 1000 = 0.50

P(x2=Sweet) = 650 / 1000 = 0.65

P(x3=Yellow) = 800 / 1000 = 0.80

Step 3: Compute the probability of likelihood of evidences that goes in the numerator. It is the product of conditional probabilities of the 3 features. If you refer back to the formula, it says P(X1 |Y=k).

Here X1 is 'Long' and k is 'Banana'.

That means the probability the fruit is 'Long' given that it is a Banana. In the above table, you have 500 Bananas. Out of that 400 is long.

So, P(Long | Banana) = 400/500 = 0.8. Here, I have done it for Banana alone.

Probability of Likelihood for Banana

P(x1=Long | Y=Banana) = 400 / 500 = 0.80

P(x2=Sweet | Y=Banana) = 350 / 500 = 0.70

P(x3=Yellow | Y=Banana) = 450 / 500 = 0.90.

Step 4: Substitute all the 3 equations into the Naïve Bayes formula, to get the probability that it is a banana.

P(Banana | Long, Sweet and Yellow) = 0.252/P(Evidence)

Similarly, you can compute the probabilities for 'Orange' and 'Other fruit'. The denominator is the same for all 3 cases, so it's optional to compute.

P(Orange | Long, Sweet and Yellow) = 0, because P(Long | Orange) = 0
P(Other Fruit | Long, Sweet and Yellow) = 001875/ P(Evidence)

Clearly, Banana gets the highest probability, so that will be our predicted class.

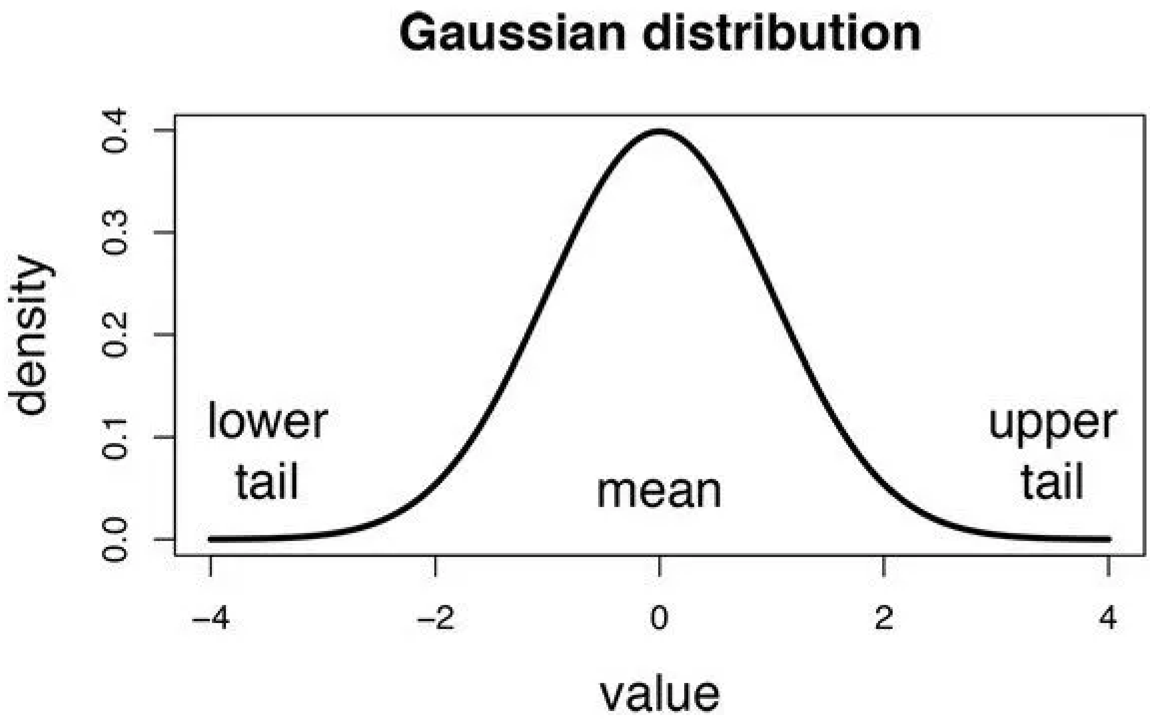
Flavors of Naïve Bayes Model:

There are three types of Naïve Bayes Model, which are given below:

- **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.

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

Probability Density function for Normal Distribution



- **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.

- **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.

Drawbacks of Naïve Bayes:

- **Zero probability problem :** When we encounter words in the test data for a particular class that are not present in the training data, we might end up with zero class probabilities. See the example below for more details: P(bumper |

Ham) is 0 since bumper does not occur in any ham (non-spam) documents in the training data.

- **Numerical instability:** For all the Naïve Bayes, numerical instability occurs when we have data with high dimensionality.

Solutions:

1. **Laplace Smoothing:**

The zero probability problem can be Remedied through the Regularization technique called **Laplace smoothing** where we add a small smoothing factor to the numerator and denominator of every probability to avoid zero even for new words.

For example, we have built a word finder model, and we have to find the word 'sport' in a sentence, but the model is not trained for that word. Then the probability of sport, i.e., P(word/sport), is zero, and after we multiply the possibilities, the product will be zero.

That is why we need Laplace smoothing; it ensures that the posterior probability is never zero. It increases the zero probability values to small positive values and simultaneously reduces other matters so that the final sum remains to be one.

We modify the posterior probability formula in the following way:

$$P(w'|positive) = \frac{\text{number of reviews with w' and y = positive} + \alpha}{\text{N} + \alpha * K}$$

In the above formula mentioned above,

α is the number of smoothing/hyper parameters [0,+∞]

K represents the number of parameters.

N represents the number of reviews considering y is positive.

2. Log Transformation:

Due to the large number of input features, we must multiply all conditional probabilities *(0<P<1)* in order to determine the final value. As a result, the outcome is a relatively low number that is close to 0.

We need to remember that multiplication operation becomes an addition in the logarithm space. So, taking the logarithm of the whole equation gives us:

logP(Ci|D) = log[P(Ci)P(D|Ci)]

logP(Ci|D) = logP(Ci)+logP(D|Ci)

Advantages:

- It is not only a simple approach but also a fast and accurate method for prediction.
- Naïve Bayes has a very low computation cost.
- It can efficiently work on a large dataset.
- It performs well in case of discrete response variable compared to the continuous variable.
- It can be used with multiple class prediction problems.
- It also performs well in the case of text analytics problems.
- When the assumption of independence holds, a Naive Bayes classifier performs better compared to other models like logistic regression.

Disadvantages:

- The assumption of independent features. In practice, it is almost impossible that model will get a set of predictors which are entirely independent.
- If there is no training tuple of a particular class, this causes zero posterior probability. In this case, the model is unable to make predictions. This problem is known as Zero Probability/Frequency Problem.

Use cases:

The Naïve Bayes Algorithm is used for various real-world problems like those below:

- **Text classification:** The Naïve Bayes Algorithm is used as a probabilistic learning technique for text classification. It is one of the best-known algorithms used for document classification of one or many classes.

- **Sentiment analysis:** The Naïve Bayes Algorithm is used to analyze sentiments or feelings, whether positive, neutral, or negative.
- **Recommendation system:** The Naïve Bayes Algorithm is a collection of collaborative filtering issued for building hybrid recommendation systems that assist you in predicting whether a user will receive any resource.
- **Spam filtering:** It is also similar to the text classification process. It is popular for helping you determine if the mail you receive is spam.
- **Medical diagnosis:** This algorithm is used in medical diagnosis and helps you to predict the patient's risk level for certain diseases.
- **Weather prediction:** You can use this algorithm to predict whether the weather will be favorable.
- **Face recognition:** This helps you identify faces.

Naive Bayes