



## Learn Naive Bayes Algorithm For Machine Learning [With Examples]

Blog Author Last updated: Views Read Time  
Pavan Vadapalli 24th Jun, 2023 6,522 15 Mins

### In this article

1. Introduction
2. Basics of Probability
3. Bayes Theorem
4. Naive Bayes Algorithm

[View All](#)

^



**Learn Naive Bayes**  
Algorithm with Examples

## Introduction

In mathematics and [programming](#), some of the simplest and most powerful ones. The naïve Bayes Algorithm is a prime example of this statement. Even with the rapid advancement and development in the field of [Machine Learning](#), this Naïve Bayes Algorithm still stands strong as one of the most widely used

Fast Forward Your Career with online courses, starting ₹10,000 per month     Let us know...

[Talk to Career Expert](#)

applications in a variety of problems including Classification tasks and

[Natural Language Processing \(NLP\)](#) problems.

## Best Machine Learning and AI Courses Online

<b>Master of Science in Machine Learning &amp; AI from LJMU</b>	<b>Executive Post Graduate Programme in Machine Learning &amp; AI from IIITB</b>
Advanced Certificate Programme in Machine Learning & NLP from IIITB	Advanced Certificate Programme in Machine Learning & Deep Learning from IIITB
<b>To Explore all our courses, visit our page below.</b>	
<a href="#"><u>Machine Learning Courses</u></a>	

The mathematical hypothesis of the Bayes Theorem serves as the fundamental concept behind this Naïve Bayes Algorithm. In this article, we shall go through the basics of Bayes Theorem, the Naïve Bayes Algorithm along with its implementation in Python with a real-time example problem. Along with these, we shall also look at some advantages and disadvantages of the [Naïve Bayes](#) Algorithm in comparison with its competitors.

## Basics of Probability

Before we venture out on understanding the Bayes Theorem and Naïve Bayes Algorithm, let us brush up our existing knowledge upon the fundamentals of Probability.

Talk to Career Expert

**Job openings for ML**  
Engineers have grown by **344%\***  
Build your career in the industry of the future.

**Executive PG Program in  
Machine Learning & AI  
from IIITB**

Online | 12 Months

\*AnalyticsIndiaMag.com

## In-demand Machine Learning Skills

Artificial Intelligence Courses	Tableau Courses
NLP Courses	Deep Learning Courses

Get [Machine Learning Certification](#) from the World's top Universities.

Earn Masters, Executive PGP, or Advanced Certificate Programs to fast-track your career.

As we all know by definition, given an event A, the probability of that event occurring is given by  $P(A)$ . In probability, two events A and B are termed as independent events if the occurrence of event A does not alter the probability of occurrence of event B and vice versa. On the other hand, if one's occurrence changes the probability of the other, then they are termed as Dependent events.

Let us get introduced to a new term called **Conditional Probability**. In mathematics, Conditional Probability for two events A and B given by  $P(A|B)$  is defined as the probability of the occurrence of event A given that event B has already occurred. Depending upon the relationship between the two events A and B as to whether they are dependent or independent, Conditional Probability is calculated in two ways.

Talk to Career Expert

given by  $P(A|B) = P(A \text{ and } B) / P(B)$

- The expression for the conditional probability of two independent events A and B is given by,  $P(A|B) = P(A)$

Knowing the math behind Probability and Conditional Probabilities, let us now move on towards the Bayes Theorem.

## Bayes Theorem

In statistics and probability theory, the Bayes' Theorem also known as the Bayes' rule is used to determine the conditional probability of events. In other words, the [Bayes' theorem](#) describes the probability of an event based on prior knowledge of the conditions that might be relevant to the event.

To understand it in a simpler way, consider that we need to know the probability of the price of a house is very high. If we know about the other parameters such as the presence of schools, medical shops and hospitals nearby, then we can make a more accurate assessment of the same. This is exactly what the Bayes Theorem performs.

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

Such that,

- $P(A|B)$  – the conditional probability of event A occurring, given event B has occurred also known as Posterior Probability.
- $P(B|A)$  – the conditional probability of event B occurring, given event A has occurred also known as Likelihood Probability.
- $P(A)$  – the probability of event A occurring also known as Prior Probability.
- $P(B)$  – the probability of event B occurring also known as Marginal Probability.

Talk to Career Expert

independent variables and the dependent variable which is the output is a Boolean value (True or False). Suppose the independent attributes are categorical in nature let us consider 2 categories for this example. Hence, with these data, we need to calculate the value of the Likelihood Probability,  $P(B|A)$ .

Hence, on observing the above we find that we need to calculate  $2^*(2^n - 1)$  parameters in order to learn this Machine Learning model.

Similarly, if we have 30 Boolean independent attributes, then the total number of parameters to be calculated will be close to 3 billion which is extremely high in computational cost.

This difficulty in building a Machine Learning model with the Bayes Theorem led to the birth and development of the Naïve Bayes Algorithm.

## Naïve Bayes Algorithm

In order to be practical, the above-mentioned complexity of the Bayes Theorem needs to be reduced. This is exactly achieved in the Naïve Bayes Algorithm by making few assumptions. The assumptions made are that each feature makes an **independent** and **equal** contribution to the outcome.

The naïve Bayes Algorithm is a supervised learning algorithm and it is based on the Bayes theorem which is primarily used in solving classification problems. It is one of the simplest and most accurate Classifiers which build Machine Learning models to make quick predictions. Mathematically, it is a probabilistic classifier as it makes predictions using the probability function of the events.

## Example Problem

 In order to understand the logic behind the assumptions, let us go through a simple dataset to get a better intuition.

Talk to Career Expert

Colour	Type	Origin	Theft?
Black	Sedan	Imported	Yes
Black	SUV	Imported	No
Black	Sedan	Domestic	Yes
Black	Sedan	Imported	No
Brown	SUV	Domestic	Yes
Brown	SUV	Domestic	No
Brown	Sedan	Imported	No
Brown	SUV	Imported	Yes
Brown	Sedan	Domestic	No

From the above-given dataset, we can derive the concepts of the two assumptions that we defined for the Naïve Bayes Algorithm above.

- The first assumption is that all the features are independent of each other. Here, we see that each attribute is independent such as the colour “Red” is **independent** of the Type and Origin of the car.
- Next, each feature is to be given equal importance. Similarly, only having knowledge about the Type and Origin of the Car isn’t sufficient to predict the output of the problem. Hence, none of the variables is irrelevant and hence they all make an **equal** contribution to the outcome.

To sum it up, A and B are conditionally independent given C if and only if, given the knowledge that C occurs, knowledge of whether A occurs provides no information on the likelihood of B occurring, and knowledge of whether B occurs provides no information on the likelihood of A occurring. These assumptions make the Bayes algorithm – **Naïve**. Hence the name, Naïve Bayes Algorithm.

Talk to Career Expert

rewritten as –

$$P(y|X) = \frac{P(X|y)P(y)}{P(X)}$$

Such that,

- The independent feature vector,  $X = (x_1, x_2, x_3, \dots, x_n)$  representing the features such as Colour, Type and Origin of the Car.
- The output variable,  $y$  has only two outcomes Yes or No.

Hence, by substituting the above values, we obtain the Naïve Bayes Formula as,

$$P(y|x_1, \dots, x_n) = \frac{P(x_1|y)P(x_2|y)\dots P(x_n|y)P(y)}{P(x_1)P(x_2)\dots P(x_n)}$$

In order to calculate the posterior probability  $P(y|X)$ , we have to create a Frequency Table for each attribute against the output. Then converting the frequency tables to Likelihood Tables after which we finally use the Naïve Bayesian equation to calculate the posterior probability for each class. The class with the highest posterior probability is chosen as the outcome of the prediction. Below are the Frequency and likelihood tables for all three predictors.

Frequency Table of Colour

Likelihood Table of Colour

		Stolen?	
		Yes	No
Colour	Black	2	2
	Brown	2	3

Talk to Career Expert

		P(Yes)	P(No)
Colour	Black	2/4	2/4
	Brown	2/5	3/5

Frequency Table of Type

Likelihood Table of Type

		Stolen?	
		Yes	No
Type	Sedan	2	3
	SUV	2	2

		Stolen?	
		P(Yes)	P(No)
Type	Sedan	2/5	3/5
	SUV	2/4	2/4

Frequency Table of Origin

Likelihood Table of Origin

		Stolen?	
		Yes	No
Origin	Domestic	2	2
	Imported	2	3

		Stolen?	
		P(Yes)	P(No)
Origin	Domestic	2/4	2/4
	Imported	2/5	3/5



Consider the case where we need to calculate the posterior probabilities for the below-given conditions –

Talk to Career Expert

Colour	Type	Origin
Brown	SUV	Imported

Thus, from the above given formula, we can calculate the Posterior Probabilities as shown below—

$$P(\text{Yes} | X) = P(\text{Brown} | \text{Yes}) * P(\text{SUV} | \text{Yes}) * P(\text{Imported} | \text{Yes}) * P(\text{Yes})$$

$$= 2/5 * 2/4 * 2/5 * 1$$

$$= 0.08$$

$$P(\text{No} | X) = P(\text{Brown} | \text{No}) * P(\text{SUV} | \text{No}) * P(\text{Imported} | \text{No}) * P(\text{No})$$

$$= 3/5 * 2/4 * 3/5 * 1$$

$$= 0.18$$

From the above-calculated values, as the Posterior Probabilities for No is Greater than Yes ( $0.18 > 0.08$ ), then it can be inferred that a car with Brown Colour, SUV Type of an Imported Origin is classified as “No”. Hence, the car is not stolen.

## Implementation in Python

Now that we have understood the math behind the Naïve Bayes algorithm and also visualized it with an example, let us go through its Machine Learning code in Python language.

**Related:** [Naive Bayes Classifier](#)

## Problem Analysis

Talk to Career Expert

Machine Learning using Python, we will be using the very famous ‘Iris Flower Dataset’. The Iris flower data set or Fisher’s Iris data set is a multivariate data set introduced by the British statistician, eugenicist, and biologist Ronald Fisher in 1998. This is a very small and basic dataset that consists of very less numeric data containing information about 3 classes of flowers belonging to the Iris species which are –

- Iris Setosa
- Iris Versicolour
- Iris Virginica

There are 50 samples of each of the three species amounting to a total dataset of 150 rows. The 4 attributes (or) independent variables that are used in this dataset are –

- sepal length in cm
- sepal width in cm
- petal length in cm
- petal width in cm

The dependant variable is the “species” of the flower that is identified by the above given four attributes.

## Step 1 – Importing the Libraries

As always, the primary step in building any Machine Learning model will be to import the relevant libraries. For this, we shall load the NumPy, Matplotlib and the Pandas libraries for pre-processing the data.

```
import numpy as np  
import matplotlib.pyplot as plt  
import pandas as pd
```

Talk to Career Expert

The Iris flower dataset to be used for training the Naïve Bayes Classifier shall be loaded into a Pandas DataFrame. The 4 independent variables shall be assigned to the variable X and the final output species variable is assigned to y.

```
dataset = pd.read_csv('https://raw.githubusercontent.com/mkgurucharan/Classification/master/IrisDataset.csv')X =  
dataset.iloc[:, :4].values  
y = dataset['species'].values  
dataset.head(5)>>  
sepal_length sepal_width petal_length petal_width species  
5.1      3.5      1.4      0.2    setosa  
4.9      3.0      1.4      0.2    setosa  
4.7      3.2      1.3      0.2    setosa  
4.6      3.1      1.5      0.2    setosa  
5.0      3.6      1.4      0.2    setosa
```

### Step 3 – Splitting the dataset into the Training set and Test set

After loading the dataset and the variables, the next step is to prepare the variables that will undergo the training process. In this step, we have to split the X and y variables to training and the test datasets. For this, we shall assign 80% of the data randomly to the training set which will be used for training purposes and the remaining 20% of the data as the test set on which the trained Naïve Bayes Classifier shall be tested for accuracy.

```
from sklearn.model_selection import train_test_split  
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.2)
```

### Step 4 – Feature Scaling

Though this is an additional process to this small dataset, I am adding this for you to use it in a larger dataset. In this, the data in the training and test sets are scaled down to a range of values between 0 and 1. This reduces the computational cost.

---

Talk to Career Expert

```
sc = StandardScaler()  
X_train = sc.fit_transform(X_train)  
X_test = sc.transform(X_test)
```

## Step 5 – Training the Naïve Bayes Classification model on the Training Set

It is in this step that we import the Naïve Bayes class from sklearn library. For this model, we use the Gaussian model, there are several other models such as Bernoulli, Categorical and Multinomial. Thus, the X\_train and y\_train are fitted to the classifier variable for training purpose.

```
from sklearn.naive_bayes import GaussianNB  
classifier = GaussianNB()  
classifier.fit(X_train, y_train)
```

## Step 6 – Predicting the Test set results –

We predict the class of the species for the Test set using the model trained and compare it with the Real Values of the species class.

```
y_pred = classifier.predict(X_test)
```

```
df = pd.DataFrame({'Real Values':y_test, 'Predicted Values':y_pred})  
df>>  
Real Values Predicted Values  
setosa      setosa  
setosa      setosa  
virginica   virginica  
versicolor  versicolor  
setosa      setosa  
setosa      setosa  
... ... ... ...  
virginica   versicolor  
virginica   virginica
```

Talk to Career Expert

```
versicolor  versicolor  
versicolor  versicolor
```

In the above comparison, we see that there is one incorrect prediction that has predicted Versicolor instead of virginica.

## Step 7 – Confusion Matrix and Accuracy

As we are dealing with Classification, the best way to evaluate our classifier model is to print the Confusion Matrix along with its accuracy on the test set.

```
from sklearn.metrics import confusion_matrix  
cm = confusion_matrix(y_test, y_pred)  
from sklearn.metrics import  
accuracy_score  
  
print ("Accuracy : ", accuracy_score(y_test, y_pred))  
  
cm>>Accuracy : 0.9666666666666667
```

```
array([[14,  0,  0],  
       [ 0,  7,  0],  
       [ 0,  1,  8]])
```

## Types of Naive Bayes Classifiers

The different types of **Naive Bayes classifiers** in machine learning are as follows:

- **Multinomial Naive Bayes Classifier**

A **multinomial** distribution generates certain events whose frequencies are represented by feature vectors. This event model is particularly useful for document classification.

Talk to Career Expert

The Gaussian or normal distribution method is assumed to distribute values associated with different features. When plotted, it provides a bell-shaped curve symmetric to the mean of the feature values.

- **Bernoulli Naive Bayes Classifier**

Features in a multivariate Bernoulli event model are independent binary variables or booleans describing inputs. Similar to the multinomial model, it is popular for performing document classification tasks. During these tasks, binary term occurrence features are leveraged instead of term frequencies.

- **Optimal Naive Bayes Classifier**

The optimal Naive Bayes classifiers choose the class with the greatest posterior probability of happenings. Even though the name suggests it is optimal, it will go through multiple possibilities. Therefore, the optimal Naive Bayes classifier is slow and time-consuming.

## How to Construct a Naive Bayes Classifier

You will have to combine different preprocessing techniques and build a dictionary of words and every word's count in training data.

1. Calculate the probability for each word in a text and sort out the words with a probability lower than the threshold. Any word with a lower probability than the threshold is irrelevant.
2. Next, you will have to create a probability for every word in the

Talk to Career Expert

determining the conditional probability is crucial for using the Naive Bayes algorithm.

3. The last step is making predictions with the help of conditional probabilities.

## Benefits of the Naive Bayes Classifier

The different benefits of the **Naive Bayes algorithm** are as follows:

- Easy and simple to implement
- Can handle discrete as well as continuous data
- Does not need a large amount of training data
- Extremely scalable with multiple data points and predictors
- Fast and useful for real-time predictions
- Does not support irrelevant features

## Applications of the Naive Bayes Algorithm



A few applications of the **Naive Bayes algorithm in machine learning** are as follows:

- **Spam Filtering:** It can help determine whether the emails and messages you are receiving are spam.
- **Text Classification:** It is a popular probabilistic learning

Talk to Career Expert

more classes.

- **Weather Prediction:** The algorithm is extremely useful for predicting if the weather will be good.
- **Recommendation System:** The Naive Bayes algorithm is used for developing hybrid recommendation systems with the help of collaborative filtering. The recommendation systems are useful for predicting whether a particular user will be receiving any resource.
- **Medical Diagnosis:** The Naive Bayes algorithm is extremely beneficial for medical diagnosis. It helps predict the risk level of patients for particular diseases.
- **Sentiment Analysis:** The Naive Bayes algorithm can help analyze whether a particular feeling or sentiment is positive, negative, or neutral.
- **Face Recognition:** The algorithm is also quite popular for identifying faces.

## Popular AI and ML Blogs & Free Courses

IoT: History, Present & Future	Machine Learning Tutorial: Learn ML	What is Algorithm? Simple & Easy
Robotics Engineer Salary in India : All Roles	A Day in the Life of a Machine Learning Engineer: What do they do?	What is IoT (Internet of Things)
Permutation vs Combination: Difference between Permutation and Combination	Top 7 Trends in Artificial Intelligence & Machine Learning	Machine Learning with R: Everything You Need to Know
 <b>AI &amp; ML Free Courses</b>		

Talk to Career Expert

<a href="#">Introduction to NLP</a>	<a href="#">Fundamentals of Deep Learning of Neural Networks</a>	<a href="#">Linear Regression: Step by Step Guide</a>
<a href="#">Artificial Intelligence in the Real World</a>	<a href="#">Introduction to Tableau</a>	<a href="#">Case Study using Python, SQL and Tableau</a>

## Conclusion

Thus, in this article, we have gone through the basics of the Naïve Bayes Algorithm, understood the math behind the Classification along with a hand-solved example. Finally, we implemented a Machine Learning code to solve a popular dataset using the Naïve Bayes Classification algorithm.

If you're interested to learn more about AI, machine learning, check out IIIT-B & upGrad's [PG Diploma in Machine Learning & AI](#) which is designed for working professionals and offers 450+ hours of rigorous training, 30+ case studies & assignments, IIIT-B Alumni status, 5+ practical hands-on capstone projects & job assistance with top firms.

**Pavan Vadapalli**

Blog Author

Director of Engineering @ upGrad. Motivated to leverage technology to solve problems. Seasoned leader for startups and fast moving orgs. Working on solving problems of scale and long term technology strategy.

## Frequently Asked Questions (FAQs)

### 1. How is probability helpful in Machine Learning?

We may have to make decisions based on partial or incomplete information in real-world scenarios. Probability helps us quantify the uncertainties in such systems and manage the risk.

Talk to Career Expert

actions, but there is always some scope of uncertainty in any prediction model. This uncertainty can come from many parameters from the input data, such as Noise in data. Also, bayesian views from probability theorems can help pattern recognition from the input data. For this, probability uses the maximum likelihood estimation concept and hence is helpful to produce relevant results.

## 2. What is the use of the Confusion Matrix?

The confusion matrix is a 2x2 matrix used to interpret the performance of the classification model. The true values for the input data must be known for this to work, so it cannot be represented for unlabeled data. It consists of the number of false positives (FP), true positives (TP), false negatives (FN), and true negatives (TN). The predictions are classified into these classes using the count from the training set and the test set. It helps us visualize useful parameters such as accuracy, precision, recall, and specificity. It is relatively easy to understand and gives you a clear idea about the algorithm.

## 3. What are the different types of Naive Bayes model?

All types are primarily based on Bayes Theorem. The Naive Bayes model generally has three types: Gaussian, Bernoulli, and Multinomial. The Gaussian Naive Bayes assists with continuous values from the input parameters, and it has the assumption that all the classes of input data are uniformly distributed. Bernoulli's naive Bayes is an event-based model where the data features are independent and present in boolean values. Multinomial Naive Bayes is also based on an event-based model. It has the data features in vector form, which represents relevant frequencies based on the occurrence of the events.

## Explore Free Courses



Talk to Career Expert