Contents

[Naïve Bayes 1](#_Toc17612343)

[How Naive Bayes algorithm works? 2](#_Toc17612344)

[What are the Pros and Cons of Naive Bayes? 3](#_Toc17612345)

[Pros: 3](#_Toc17612346)

[Cons: 3](#_Toc17612347)

[Usecase 3](#_Toc17612348)

[Naive Bayes Theorem 3](#_Toc17612349)

# Naïve Bayes

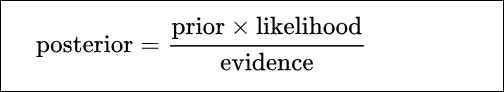
It is a classification technique based on [Bayes’ Theorem](https://en.wikipedia.org/wiki/Bayes%27_theorem) with an assumption of independence among predictors. In simple terms, a Naive Bayes classifier assumes that the presence of a particular feature in a class is unrelated to the presence of any other feature

. For example, a fruit may be considered to be an apple if it is red, round, and about 3 inches in diameter. Even if these features depend on each other or upon the existence of the other features, all of these properties independently contribute to the probability that this fruit is an apple and that is why it is known as ‘Naive’.

Along with simplicity, Naive Bayes is known to outperform even highly sophisticated classification methods.

Bayes theorem provides a way of calculating posterior probability P(c|x) from P(c), P(x) and P(x|c). Look at the equation below:

[](https://www.analyticsvidhya.com/wp-content/uploads/2015/09/Bayes_rule-300x172.png)



Above,

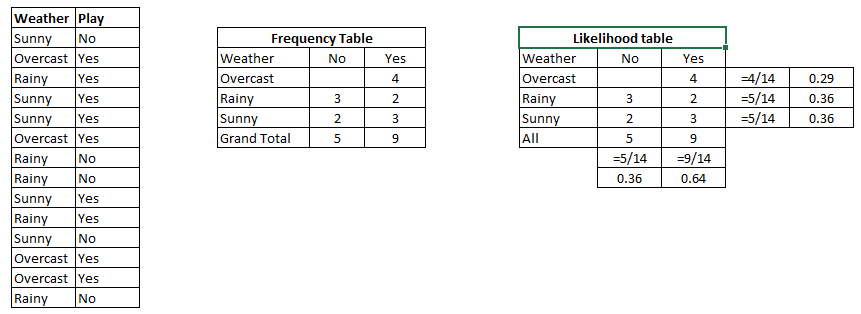
* *P*(*c|x*) is the posterior probability of *class* (c, *target*) given *predictor* (x, *attributes*).
* *P*(*c*) is the prior probability of *class*.
* *P*(*x|c*) is the likelihood which is the probability of *predictor* given *class*.
* *P*(*x*) is the prior probability of *predictor*.

# How Naive Bayes algorithm works?

Let’s understand it using an example. Below I have a training data set of weather and corresponding target variable ‘Play’ (suggesting possibilities of playing). Now, we need to classify whether players will play or not based on weather condition. Let’s follow the below steps to perform it.

Step 1: Convert the data set into a frequency table

Step 2: Create Likelihood table by finding the probabilities like Overcast probability = 0.29 and probability of playing is 0.64.



Step 3: Now, use Naive Bayesian equation to calculate the posterior probability for each class. The class with the highest posterior probability is the outcome of prediction.

We can solve it using above discussed method of posterior probability.

P(Yes | Sunny) = P( Sunny | Yes) \* P(Yes) / P (Sunny)

Here we have P (Sunny |Yes) = 3/9 = 0.33, P(Sunny) = 5/14 = 0.36, P( Yes)= 9/14 = 0.64

Now, P (Yes | Sunny) = 0.33 \* 0.64 / 0.36 = 0.60, which has higher probability.

Naive Bayes uses a similar method to predict the probability of different class based on various attributes. This algorithm is mostly used in text classification and with problems having multiple classes.

# What are the Pros and Cons of Naive Bayes?

### **Pros:**

* It is easy and fast to predict class of test data set. It also perform well in multi class prediction
* When assumption of independence holds, a Naive Bayes classifier performs better compare to other models like logistic regression and you need less training data.
* It perform well in case of categorical input variables compared to numerical variable(s). For numerical variable, normal distribution is assumed (bell curve, which is a strong assumption).

### **Cons:**

* If categorical variable has a category (in test data set), which was not observed in training data set, then model will assign a 0 (zero) probability and will be unable to make a prediction. This is often known as “Zero Frequency”.

To solve this, we can use the smoothing technique. One of the simplest smoothing techniques is called Laplace estimation.

* On the other side naive Bayes is also known as a bad estimator, so the probability outputs from predict\_proba are not to be taken too seriously.
* Another limitation of Naive Bayes is the assumption of independent predictors. In real life, it is almost impossible that we get a set of predictors which are completely independent.

# Usecase

Naive Bayes Theorem

Naive Bayes is a powerful supervised learning algorithm that is used for classification. The Naive Bayes classifier is an extension of the above discussed standard Bayes Theorem. In a Naive Bayes, we calculate the probability contributed by every factor. Most we use it in textual classification operations like spam filtering. Let us understand how Naive Bayes calculates the probability contributed by all the factors.

Suppose that, as a data scientist, you are tasked with developing a spam filter. You are provided with a list of spam keywords such as

* Free
* Discount
* Full Refund
* Urgent
* Weight Loss

However, the company you are working with is a product finance company. Therefore, some of the vocabulary occurring in the spam mails is used in the mails of your company. Some of these words are –

* Important
* Free
* Urgent
* Stocks
* Customers

You also have the probability of word usages in spam messages and company emails.

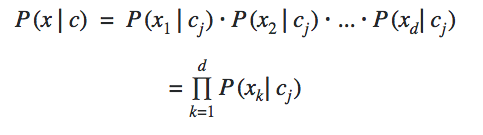
|  |  |
| --- | --- |
| **Spam Email** | **Company Email** |
| Free (0.3) | Important (0.5) |
| Discount (0.15) | Free (0.25) |
| Full Refund (0.1) | Urgent (0.1) |
| Urgent (0.2) | Stocks (0.5) |
| Weight Loss (0.25) | Customers (0.1) |

Suppose you obtain have a message “Free trials for weight loss program. Become members at a discount.” Is this message spam or a company email? Calculating the probability of the components occurring in the sentence – **Free (0.4) + Weight Loss (0.25) + Discount (0.15) = 0.8 or 80%**

Whereas, calculating the probability of it being an email from your company = F**ree (0.25) = 0.25 or 25%.**

Therefore, the probability of the mail being spam is much higher than a company email.

A Naive Bayes Classifier selects the outcome of the highest probability, which in the above case was the feature of spam. The Naive Bayes is referred to as ‘naive’ because it assumes the features to be independent of each other. The features in our example were the input words that are present in the sentence.

[](https://d2h0cx97tjks2p.cloudfront.net/blogs/wp-content/uploads/sites/2/2019/05/conditional-independence.png)

The conditional independence among all the features gives us the formula above. The frequency of the occurrence of features from x1 to xdis calculated based on their relation to the class cj. Along with the prior probability and the probability of the occurrence of an event, we calculate the posterior probability through which we are able to find the probability of the object belonging to a particular class.

Along with this Bayes’ Theorem, Data Scientists use various different tools. You must check the different [**tools used by a Data Scientist**](https://data-flair.training/blogs/data-science-tools/).

# Interview Question

## ****Why is naive Bayes so ‘naive’ ?****

**Answer:** naive Bayes is so ‘naive’ because it assumes that all of the features in a data set are equally important and independent. As we know, these assumption are rarely true in real world scenario.

## ****Q6. Explain prior probability, likelihood and marginal likelihood in context of naiveBayes algorithm?****

**Answer:** Prior probability is nothing but, the proportion of dependent (binary) variable in the data set. It is the closest guess you can make about a class, without any further information. For example: In a data set, the dependent variable is binary (1 and 0). The proportion of 1 (spam) is 70% and 0 (not spam) is 30%. Hence, we can estimate that there are 70% chances that any new email would  be classified as spam.

Likelihood is the probability of classifying a given observation as 1 in presence of some other variable. For example: The probability that the word ‘FREE’ is used in previous spam message is likelihood.

Marginal likelihood is, the probability that the word ‘FREE’ is used in any message.