# Naive Bayes

Supervised ML Algorithm

Sahasra Ranjan

April 2020

Naive Bayes methods are a set of supervised learning algorithms based on applying Bayes' theorem with the ”naive” assumption of conditional independence between every pair of features given the value of class variable.

What actually Bayes' theorem is?

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)} \tag{1}$$

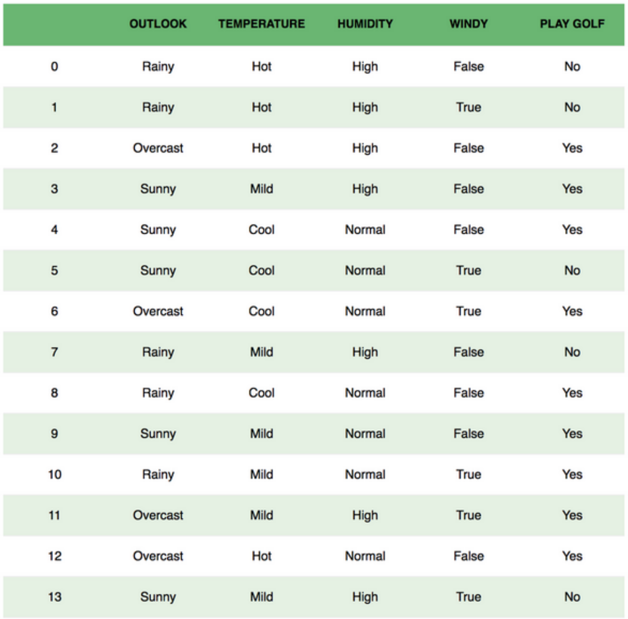
$P(A|B)$ is the probability of **A** happening, given that **B** has occured.

Why ”Naive”? Because the presence of one particular feature does not affect the other.

Without going to deep, let's see an example:

## The Golf Match Problem

Consider the problem of playing golf, dataset for the same:



| | OUTLOOK | TEMPERATURE | HUMIDITY | WINDY | PLAY GOLF |
|---|---|---|---|---|---|
| 0 | Rainy | Hot | High | False | No |
| 1 | Rainy | Hot | High | True | No |
| 2 | Overcast | Hot | High | False | Yes |
| 3 | Sunny | Mild | High | False | Yes |
| 4 | Sunny | Cool | Normal | False | Yes |
| 5 | Sunny | Cool | Normal | True | No |
| 6 | Overcast | Cool | Normal | True | Yes |
| 7 | Rainy | Mild | High | False | No |
| 8 | Rainy | Cool | Normal | False | Yes |
| 9 | Sunny | Mild | Normal | False | Yes |
| 10 | Rainy | Mild | Normal | True | Yes |
| 11 | Overcast | Mild | High | True | Yes |
| 12 | Overcast | Hot | Normal | False | Yes |
| 13 | Sunny | Mild | High | True | No |

Figure 1: Dataset for possiblity of a golf match

Bayes theorem for this example can be rewritten as:

$$P(y|X) = \frac{P(X|y)P(y)}{P(X)} \tag{2}$$

The variable **y** is the class variable (play golf), which represents if it is suitable to play golf or not given the conditions. Variable **X** is a matrix representing the parameters/features.

$\mathbf{X} = (x_1, x_2, x_3, ..., x_n)$ $x_1, x_2, ...x_n$ represent the features[1].

$$P(y|x_1, ..., x_n) = \frac{P(x_1|y)P(x_2|y)...P(x_n|y)P(y)}{P(x_1)P(x_2)...P(x_n)} \tag{3}$$

These values can be obtained by looking at the dataset and substituting them into equation will give us the result.

In our case, the the class variable(**y**) has only two outcomes, yes or no. Therefore, we need to find class **y** with maximum probability.

$$y = argmax_y P(y) \prod_{i=1}^{n} P(x_i|y) \tag{4}$$

## Types of Naive Bayes classifier:

### Miltinomial 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.

[1] temperature, humidity and windy (here)

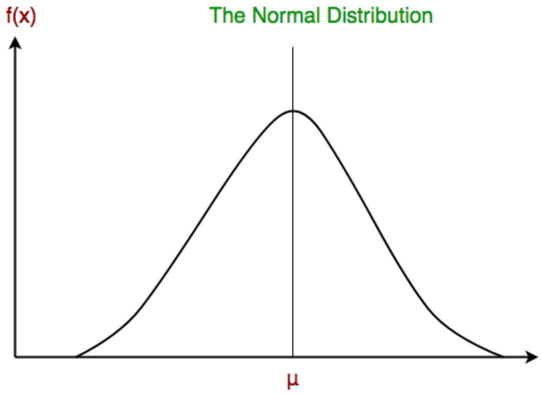


Figure 2: Gaussian Distribution(Normal Distribution)

The formula for conditional probability changes to:

$$P(x_i|y) = \frac{1}{\sqrt{2\pi\sigma_y^2}} e^{-\frac{(x_i-\mu_y)^2}{2\sigma_y^2}} \tag{5}$$

## Laplace Smoothing

It is problematic when a frequency-based probability is zero, because it will wipe out all the information in the other probabilities.

A solution would be **Laplace smoothing** , which is a technique for smoothing categorical data. A small-sample correction, or **pseudo-count**, will be incorporated in every probability estimate. Consequently, no probability will be zero. this is a way of regularizing Naive Bayes, and when the pseudo-count is zero, it is called Laplace smoothing. While in the general case it is often called **Lidstone smoothing.**

$$P_{i,\alpha-smoothed} = \frac{x_i + \alpha}{N + \alpha d} \tag{6}$$

where, $\alpha > 0$ the "pseudocount" is a smoothing parameter. And, $1/d$ is the Uniform Probability.

### Note

In practice, we use logs to represent probabilities:

$$\log\left(P(x_1|y)P(x_2|y)...P(x_n|y)\right) = \log P(x_1|y) + \log P(x_2|y) + ... + \log P(x_n|y) \quad (7)$$

## Applications

Naive Bayes algorithms are mostly used in sentiment analysis, spam filtering, recommendation systems etc. They are fast and easy to implement but their biggest disadvantage is that the requirement of predictors to be independent. In most of the real life cases, the predictors are dependent, this hinders the performance of the classifier.