**1. Introduction**

Naive Bayes is a probabilistic machine learning algorithm that can be used in a wide variety of classification tasks.

Typical applications include

1. **Real-time Prediction:**As Naive Bayes is super fast, it can be used for making predictions in real time.
2. **Multi-class Prediction:**This algorithm can predict the posterior probability of multiple classes of the target variable.
3. **Text classification/ Spam Filtering/ Sentiment Analysis:** Naive Bayes classifiers are mostly used in text classification (due to their better results in multi-class problems and independence rule) have a higher success rate as compared to other algorithms. As a result, it is widely used in Spam filtering (identify spam e-mail) and Sentiment Analysis (in social media analysis, to identify positive and negative customer sentiments)
4. **Recommendation System:** Naive Bayes Classifier along with algorithms like Collaborative Filtering makes a Recommendation System that uses machine learning and data mining techniques to filter unseen information and predict whether a user would like a given resource or not.

* Filtering spam (emails/fake news in social media),
* Classifying documents(Sports,Politics,Entertaintment),
* Sentiment prediction (Movie, Hotel, Product) etc.

It is based on the works of Rev. Thomas Bayes (1702–61) and hence the name.

But why is it called ‘Naive’?

The name naive is used because it assumes the features that go into the model is independent of each other. That is changing the value of one feature, does not directly influence or change the value of any of the other features used in the algorithm.

Alright. By the sounds of it, Naive Bayes does seem to be a simple yet powerful algorithm. But why is it so popular?

That’s because there is a significant advantage with NB. Since it is a probabilistic model, the algorithm can be coded up easily and the predictions made real quick. Real-time quick. Because of this, it is easily scalable and is traditionally the algorithm of choice for real-world applications (apps) that are required to respond to user’s requests instantaneously.

But before you go into Naive Bayes, you need to understand what ‘Conditional Probability’ is and what is the ‘Bayes Rule’.

## 2. What is Conditional Probability?

Lets start from the basics by understanding conditional probability.

**Coin Toss and Fair Dice Example**

When you flip a fair coin, there is an equal chance of getting either heads or tails. So you can say the probability of getting heads is 50%.

Similarly what would be the probability of getting a 1 when you roll a dice with 6 faces? Assuming the dice is fair, the probability of 1/6 = 0.166.

Alright, one final example with playing cards.

**Playing Cards Example**

If you pick a card from the deck, can you guess the probability of getting a queen given the card is a spade?



Well, I have already set a condition that the card is a spade. So, the denominator (eligible population) is 13 and not 52. And since there is only one queen in spades, the probability it is a queen given the card is a spade is 1/13 = 0.077

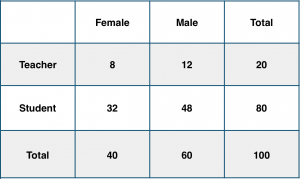
This is a classic example of conditional probability. So, when you say the conditional probability of A given B, it denotes the probability of A occurring given that B has already occurred.

Mathematically, Conditional probability of A given B can be computed as: P(A|B) = P(A AND B) / P(B)

**School Example**

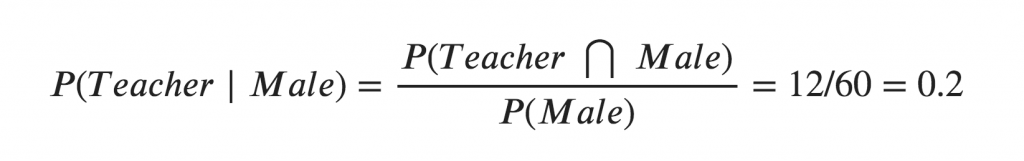
Let’s see a slightly complicated example. Consider a school with a total population of 100 persons. These 100 persons can be seen either as ‘Students’ and ‘Teachers’ or as a population of ‘Males’ and ‘Females’.

With below tabulation of the 100 people, what is the conditional probability that a certain member of the school is a ‘Teacher’ given that he is a ‘Man’?

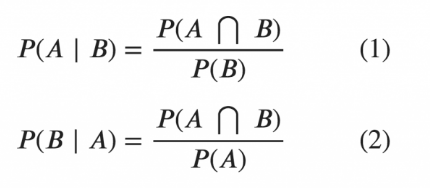
[](https://www.machinelearningplus.com/wp-content/uploads/2018/11/school_new.png)

To calculate this, you may intuitively filter the sub-population of 60 males and focus on the 12 (male) teachers.

So the required conditional probability P(Teacher | Male) = 12 / 60 = 0.2.

[](https://www.machinelearningplus.com/wp-content/uploads/2018/11/f2.png)

This can be represented as the intersection of Teacher (A) and Male (B) divided by Male (B). Likewise, the conditional probability of B given A can be computed. The Bayes Rule that we use for Naive Bayes, can be derived from these two notations.



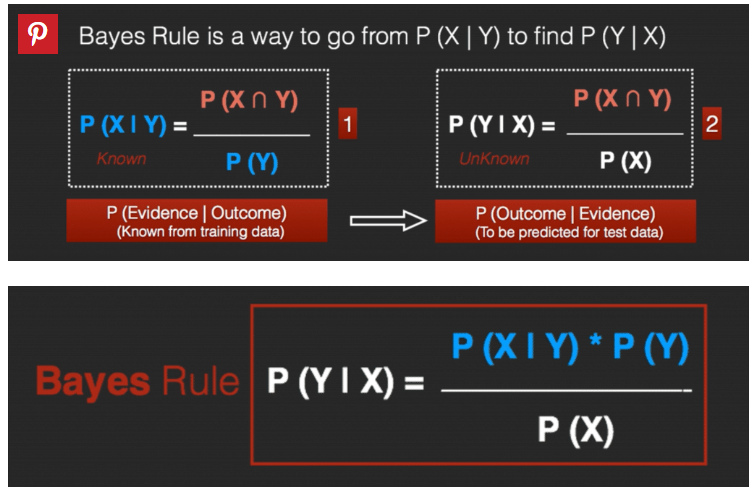
## 3. The Bayes Rule

The Bayes Rule is a way of going from P(X|Y), known from the training dataset, to find P(Y|X).

To do this, we replace A and B in the above formula, with the feature X and response Y.

For observations in test or scoring data, the X would be known while Y is unknown. And for each row of the test dataset, you want to compute the probability of Y given the X has already happened.

What happens if Y has more than 2 categories? we compute the probability of each class of Y and let the highest win.

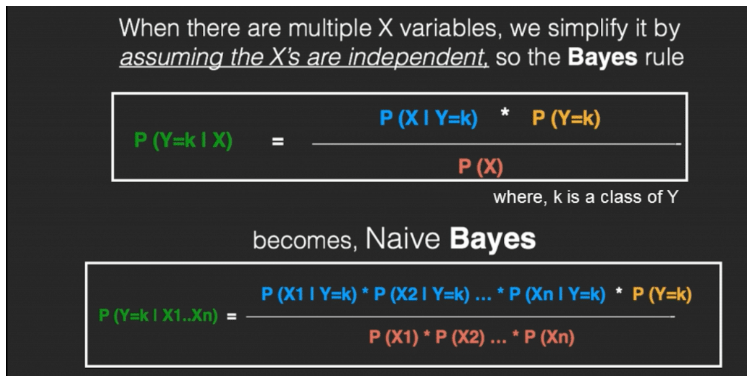


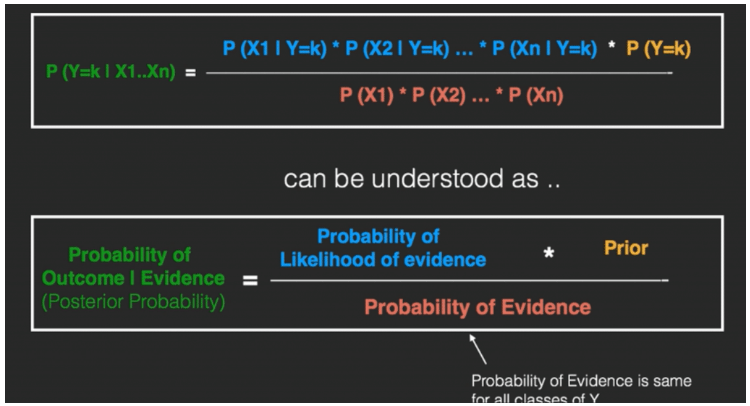
**4. The Naive Bayes**

The Bayes Rule provides the formula for the probability of Y given X. But, in real-world problems, you typically have multiple X variables.

When the features are independent, we can extend the Bayes Rule to what is called Naive Bayes.

It is called ‘Naive’ because of the naive assumption that the X’s are independent of each other. Regardless of its name, it’s a powerful formula.





In technical jargon, the left-hand-side (LHS) of the equation is understood as the posterior probability or simply the posterior

The RHS has 2 terms in the numerator.

The first term is called the **‘Likelihood of Evidence’**. It is nothing but the conditional probability of each X’s given Y is of particular class ‘c’.

Since all the X’s are assumed to be independent of each other, you can just multiply the ‘likelihoods’ of all the X’s and called it the ‘Probability of likelihood of evidence’. This is known from the training dataset by filtering records where Y=c.

The second term is called the prior which is the overall probability of Y=c, where c is a class of Y. In simpler terms, Prior = count(Y=c) / n\_Records.

[[1]](#footnote-1)

A real-time prediction is a synchronous call to Amazon Machine Learning (Amazon ML). The prediction is made when Amazon ML gets the request, and the response is returned immediately. Real-time predictions are commonly used to enable predictive capabilities within interactive web, mobile, or desktop applications. You can query an ML model created with Amazon ML for predictions in real time by using the low-latency Predict API. The Predict operation accepts a single input observation in the request payload, and returns the prediction synchronously in the response.

 Amazon ML responds to most real-time prediction requests within 100 milliseconds.

References

<https://docs.aws.amazon.com/machine-learning/latest/dg/requesting-real-time-predictions.html>

1. [↑](#footnote-ref-1)