Naive Bayes Classification Numerical Example

We use different classification algorithms to build classifiers in machine learning. The naive Bayes classification algorithm is one of the easiest classification algorithms to understand and implement. In this article, we will discuss the Bayes algorithm and the intuition of Naive Bayes classification. We will also discuss a numerical example of Naive Bayes classification to understand it in a better manner.

The Bayes' Theorem

Before discussing the Naive Bayes classification algorithm, we need to understand the Bayes theorem. We can state the formulae for the Bayes algorithm as shown below.

```
P(A/B)=  P(B/A)* P(A)/P(B)
```

Here,

- A is called the hypothesis.
- B is the evidence.
- P(A) is termed as prior probability. It is the probability of occurrence of the hypothesis.
- P(B) is termed marginal probability. It is the probability of occurrence of the evidence.
- P(B/A) is called likelihood. It is the probability of occurrence of B given that A has already occurred.
- P(A/B) is called posterior probability. It is the probability of occurrence of A given that B has already occurred.

The Bayes theorem is directly derived from the formulas of conditional probability. For instance, you might have studied the conditional probability formulae given below.

```
P(A/B)=P(A∩B)/P(B)
```

Here,

- P(B) is the probability of occurrence of event B.
- P(A∩B) is the probability of occurrence of events A and B together.
- P(A/B) is the probability of occurrence of event A given that B has already occurred.

In a similar manner, we can write the above formulae as shown below.

```
P(B/A)=P(A∩B)/P(A)
```

Here,

- P(A) is the probability of occurrence of event A.
- P(A∩ B) is the probability of occurrence of events A and B together.
- P(B/A) is the probability of occurrence of event B given that A has already occurred.

Now, if we extract the probability P(A∩B) from both formulas, we get the following.

```
P(A∩B)=P(B/A)*P(A)
P(A∩B)=P(A/B)*P(B)
```

When we equate both formulas, we get the following equation.

```
P(B/A)*P(A)=P(A/B)*P(B)
```

From the above equation, we can get the posterior probability P(A/B) as shown below.

```
P(A/B)=P(B/A)*P(A)/P(B)
```

Similarly, we can get the posterior probability P(B/A) as shown below.

```
P(B/A)=P(A/B)*P(B)/P(A)
```

The above two formulas represent the Bayes theorem in alternate forms.

Bayes Theorem Numerical Example

You are given a deck of cards. You have to find the probability of a card being king if you know that it is a face card.

We will approach this problem as follows.

- Let A be the event of a given card being a face card.
- Let B be the event of a card being a King.
- Now, if we need to find the probability of a card being king if you know that it is a face card, we need to find the probability P(B/A).

Using Bayes theorem,

```
P(B/A)=P(A/B)*P(B)/P(A)
```

To find P(B/A), we need to find the following probabilities.

- P(A) i.e. the probability of a card being a face card. As there are 12 face cards out of 52, P(A)=12/52.
- P(B) i.e. the probability of a card being a King. As there are 4 Kings, P(B)=4/52.
- P(A/B) i.e. the probability of a King being a face card. As all the kings are face cards, P(A/B)=1.

Now, using Bayes theorem, we can easily find the probability of a card being a King if it is a face card.

```
P(B/A)=P(A/B)*P(B)/P(A)
      =1*(4/52)/(12/52)
      =4/12
      =1/3
```

Hence, the probability of a card being a King, if it is a face card, is 1/3. I hope that you have understood the Bayes theorem at this point. Now, let us discuss the Naive Bayes classification algorithm.

What is The Naive Bayes Classification Algorithm?

The naive Bayes classification algorithm is a supervised machine learning algorithm based on the Bayes theorem. It is one of the simplest and most effective classification algorithms that help us build efficient classifiers with minimum training and computation costs.

In the Naive Bayes algorithm, we assume that the features in the input dataset are independent of each other. In other words, each feature in the input dataset independently decides the target variable or class label and is not affected by other features. While the assumption doesn't hold true for most of the real-world classification problems, Naive Bayes classification is still one of the goto algorithms for classification due to its simplicity.

Naive Bayes Classification Numerical example

To implement a Naive Bayes classifier, we perform three steps.

1. First, we calculate the probability of each class label in the training dataset.
2. Next, we calculate the conditional probability of each attribute of the training data for each class label given in the training data.
3. Finally, we use the Bayes theorem and the calculated probabilities to predict class labels for new data points. For this, we will calculate the probability of the new data point belonging to each class. The class with which we get the maximum probability is assigned to the new data point.

To understand the above steps using a naive Bayes classification numerical example, we will use the following dataset.

Sl. No.	Color	Legs	Height	Smelly	Species
1	White	3	Short	Yes	M
2	Green	2	Tall	No	M
3	Green	3	Short	Yes	M
4	White	3	Short	Yes	M
5	Green	2	Short	No	H
6	White	2	Tall	No	H
7	White	2	Tall	No	H
8	White	2	Short	Yes	H

Using the above data, we have to identify the species of an entity with the following attributes.

```
X={Color=Green, Legs=2, Height=Tall, Smelly=No}
```

To predict the class label for the above attribute set, we will first calculate the probability of the species being M or H in total.

```
P(Species=M)=4/8=0.5
P(Species=H)=4/8=0.5
```

Next, we will calculate the conditional probability of each attribute value for each class label.

```
P(Color=White/Species=M)=2/4=0.5
P(Color=White/Species=H)=¾=0.75
P(Color=Green/Species=M)=2/4=0.5
P(Color=Green/Species=H)=¼=0.25
```

```
P(Legs=2/Species=M)=1/4=0.25
P(Legs=2/Species=H)=4/4=1
P(Legs=3/Species=M)=3/4=0.75
P(Legs=3/Species=H)=0/4=0
P(Height=Tall/Species=M)=3/4=0.75
P(Height=Tall/Species=H)=2/4=0.5
P(Height=Short/Species=M)=1/4=0.25
P(Height=Short/Species=H)=2/4=0.5
P(Smelly=Yes/Species=M)=3/4=0.75
P(Smelly=Yes/Species=H)=1/4=0.25
P(Smelly=No/Species=M)=1/4=0.25
P(Smelly=No/Species=H)=3/4=0.75
```

We can tabulate the above calculations in the tables for better visualization.

The conditional probability table for the Color attribute is as follows.

Color	M	H
White	0.5	0.75
Green	0.5	0.25

The conditional probability table for the Legs attribute is as follows.

Legs	M	H
2	0.25	1
3	0.75	0

The conditional probability table for the Height attribute is as follows.

Height	M	H
Tall	0.75	0.5
Short	0.25	0.5

The conditional probability table for the Smelly attribute is as follows.

Smelly	M	H
Yes	0.75	0.25
No	0.25	0.75

Now that we have calculated the conditional probabilities, we will use them to calculate the probability of the new attribute set belonging to a single class.

Let us consider X= {Color=Green, Legs=2, Height=Tall, Smelly=No}.

Then, the probability of X belonging to Species M will be as follows.

```
P(M/X)=P(Species=M)*P(Color=Green/Species=M)*P(Legs=2/Species=M)*P(Height=Tall/
                Species=M)*P(Smelly=No/Species=M)
      =0.5*0.5*0.25*0.75*0.25
      =0.0117
```

Similarly, the probability of X belonging to Species H will be calculated as follows.

```
P(H/X)=P(Species=H)*P(Color=Green/Species=H)*P(Legs=2/Species=H)*P(Height=Tall/
                Species=H)*P(Smelly=No/Species=H)
      =0.5*0.25*1*0.5*0.75
      =0.0468
```

So, the probability of X belonging to Species M is 0.0117 and that to Species H is 0.0468. Hence, we will assign the entity X with attributes {Color=Green, Legs=2, Height=Tall, Smelly=No} to species H.

In this way, we can predict the class label for any number of new data points.

What are the Different Types of Naive Bayes Models?

Based on the use cases and features of input data, naive Bayes classifiers can be classified into the following types.

- **Gaussian Classifiers**: The Gaussian Naive Bayes classifier assumes that the attributes of a dataset have a normal distribution. Here, if the attributes have continuous values, the classification model assumes that the values are sampled from a Gaussian distribution.
- **Multinomial Naive Bayes Classifier**: When the input data is multinomially distributed, we use the multinomial naive Bayes classifier. This algorithm is primarily used for document classification problems like sentiment analysis.
- **Bernoulli Classifiers**: The Bernoulli Naive Bayes classification works in a similar manner to the multinomial classification. The difference is that the attributes of the dataset contain boolean values representing the presence or absence of a particular attribute in a data point.

Advantages and Disadvantages of Naive Bayes Classification Algorithm

Due to its simple implementation, the naive Bayes classifier has the following advantages.

- The naive Bayes classification algorithm is one of the fastest and easiest machine learning algorithms for classification.
- We can use the Naive Bayes classification algorithm for building binary as well as multi-class classification models.
- The Naive Bayes algorithm performs better than many classification algorithms while implementing multi-class classification models.

Apart from its advantages, the naive Bayes classification algorithm also has some drawbacks. The algorithm assumes that the attributes of the training dataset are independent of each other. This assumption is not always True. Hence, when there is a correlation between two attributes in a given training set, the naive Bayes algorithm will not perform well.

Applications of The Naive Bayes Classification Algorithm

The Naive Bayes classification algorithm is used in many classifiers.

- The most popular use of the Naive Bayes classification algorithm is in text classification. We often build spam filtering and sentiment analysis models using the naive Bayes algorithm.
- We can use the Naive Bayes classification algorithm to build applications to predict the credit score and loan worthiness of customers in a bank.
- The Naive Bayes classifier is an eager learner. Hence, we can use it for real-time predictions too.

- We can also use the Naive Bayes classification algorithm to implement models for detecting diseases based on the medical results of the patients.