Naive Bayes Classification

Wahia Tasnim
Lecturer
Green University Of Bangladesh

Preliminary Knowledge

- **Conditional probability**: a measure of the probability of event A occurring given that another event has occurred. For example, “what is the probability that it will rain given that it is cloudy?” is an example of conditional probability.
- **Joint Probability**: a measure that calculates the likelihood of two or more events occurring at the same time.

Preliminary Knowledge

Consider the following example of tossing two coins. If we toss two coins and look at all the different possibilities, we have the sample space as:{HH, HT, TH, TT}

While calculating the math on probability, we usually denote probability as P. Some of the probabilities in this event would be as follows:

- The probability of getting two heads = 1/4
- The probability of at least one tail = 3/4
- The probability of the second coin being head given the first coin is tail = 1/2
- The probability of getting two heads given the first coin is a head = 1/2

Preliminary Knowledge

Bayes Theorem:

The Bayes theorem gives us the conditional probability of event A, given that event B has occurred. In this case, the first coin toss will be B and the second coin toss A. This could be confusing because we've reversed the order of them and go from B to A instead of A to B.

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

where:

$P(A|B)$ = Conditional Probability of A given B
$P(B|A)$ = Conditional Probability of A given B
$P(A)$ = Probability of event A
$P(B)$ = Probability of event A

Preliminary Knowledge

Let us apply Bayes theorem to our coin example. Here, we have two coins, and the first two probabilities of getting two heads and at least one tail are computed directly from the sample space.

Now in this sample space, let A be the event that the second coin is head, and B be the event that the first coin is tails. Again, we reversed it because we want to know what the second event is going to be.

We're going to focus on A, and we write that out as a probability of A given B:

Probability = P(A|B) = [P(B|A) * P(A)] / P(B)

= [P(First coin being tail given the second coin is the head) * P(Second coin being head)] / P(First coin being tail) = [(1/2) * (1/2)] / (1/2) = 1/2 = 0.5

Bayes theorem calculates the conditional probability of the occurrence of an event based on prior knowledge of conditions that might be related to the event.

What is Naive Bayes?

- Naïve Bayes algorithm is a supervised learning algorithm, which is based on **Bayes theorem** and used for solving classification problems.
- It is mainly used in *text classification* that includes a high-dimensional training dataset.
- Naïve Bayes Classifier is one of the simple and most effective Classification algorithms which helps in building the fast machine learning models that can make quick predictions.
- **It is a probabilistic classifier, which means it predicts on the basis of the probability of an object**.
- Some popular examples of Naïve Bayes Algorithm are **spam filtration, Sentimental analysis, and classifying articles**

What it is called Naive Bayes?

The Naïve Bayes algorithm is comprised of two words Naïve and Bayes, Which can be described as:

- **Naïve**: It is called Naïve because it assumes that the occurrence of a certain feature is independent of the occurrence of other features. Such as if the fruit is identified on the bases of color, shape, and taste, then red, spherical, and sweet fruit is recognized as an apple. Hence each feature individually contributes to identify that it is an apple without depending on each other.
- **Bayes**: It is called Bayes because it depends on the principle of Bayes' Theorem.

Understanding the Naive Bayes and ML

Machine learning falls into two categories:

- Supervised learning
- Unsupervised learning

Supervised learning falls into two categories:

- Classification
- Regression

Naive Bayes algorithm falls under classification. This means that Naive Bayes is used when the output variable is discrete.

How Naive Bayes Works

The Naive Bayes Classifier is inspired by Bayes Theorem which states the following equation:

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

This equation can be rewritten using X (input variables) and y (output variable) to make it easier to understand. In plain English, this equation is solving for the probability of y given input features X.

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

Because of the naive assumption (hence the name) that variables are independent given the class, we can rewrite P(X|y) as follows:

$$P(X|y) = P(x_1|y) * P(x_2|y) * \dots * P(x_n|y)$$

How Naive Bayes Works

Also, since we are solving for y, P(X) is a constant which means that we can remove it from the equation and introduce a proportionality. This leads us to the following equation:

$$P(y|X) \propto P(X|y) * P(y)$$

-or-

$$P(y|X) \propto P(y) * \prod_{i=1}^{n} P(x_i|y)$$

Example of Naive Bayes

Outlook	Temperature	Humidity	Windy	Play?
sunny	hot	high	FALSE	no
sunny	hot	high	TRUE	no
overcast	hot	high	FALSE	yes
rainy	mild	high	FALSE	yes
rainy	cool	normal	FALSE	yes
rainy	cool	normal	TRUE	no
overcast	cool	normal	TRUE	yes
sunny	mild	high	FALSE	no
sunny	cool	normal	FALSE	yes
rainy	mild	normal	FALSE	yes
sunny	mild	normal	TRUE	yes
overcast	mild	high	TRUE	yes
overcast	hot	normal	FALSE	yes
rainy	mild	high	TRUE	no

Example of Naive Bayes

First, we need to convert this into a frequency table, so that we can get the values of P(X|y) and P(X). *Recall that we are solving for P(y|X)*:

$$P(y|X) \propto P(X|y) * P(y)$$

outlook			temperature			humidity			windy			play?	
	yes	no		yes	no		yes	no		yes	no	yes	no
sunny	2	3	hot	2	2	high	3	4	FALSE	6	2	9	5
overcast	4	0	mild	4	2	normal	6	1	TRUE	3	3		
rainy	3	2	cool	3	1								

Second, we want to convert the frequencies into ratios or conditional probabilities:

outlook			temperature			humidity			windy			play?	
	yes	no		yes	no		yes	no		yes	no	yes	no
sunny	2/9	3/5	hot	2/9	2/5	high	3/9	4/5	FALSE	6/9	2/5	9/14	5/14
overcast	4/9	0/5	mild	4/9	2/5	normal	6/9	1/5	TRUE	3/9	3/5		
rainy	3/9	2/5	cool	3/9	1/5								

Example of Naive Bayes

Finally, we can use the proportionality equation to predict y, given X.

Imagine that X = {outlook: sunny, temperature: mild, humidity: normal, windy: false}.

First, we'll calculate the probability that you will play golf given X, P(yes|X) followed by the probability that you won't play golf given X, P(no|X).

Using the chart above, we can get the following information:

$$P(yes) = 9/14$$

$$P(outlook = sunny|yes) = 2/9$$

$$P(temperature = mild|yes) = 4/9$$

$$P(humidity = normal|yes) = 6/9$$

$$P(windy = false|yes) = 6/9$$

Example of Naive Bayes

Now we can simply input this information into the following formula:

$$P(yes|X) \propto P(X|y) * P(y)$$

$$P(yes|X) \propto P(x_1|y) * P(x_2|y) * P(x_3|y) * P(x_4|y) * P(y)$$

$$P(yes|X) \propto P(sunny|yes) * P(mild|yes) * P(normal|yes) * P(false|yes) * P(yes)$$

$$P(yes|X) \propto \frac{2}{9} * \frac{4}{9} * \frac{6}{9} * \frac{6}{9} * \frac{9}{14}$$

$$\boldsymbol{P(yes|X) \propto 0.0282}$$

Similarly, you would complete the same sequence of steps for P(no|X).

$$\boldsymbol{P(no|X) \propto 0.0069}$$

Pros and Cons of Naive Bayes

Pros

- As shown above, it is quite intuitive once you understand the concept
- It’s easy to implement and performs well in multiclass prediction
- It works well with categorical input variables

Cons

- You can encounter the **zero-frequency problem** when there’s a category in the test set that’s not in the training set (although there are workarounds for this)
- The probability estimates are not the most trustworthy from this algorithm
- Naive Bayes holds strong assumptions, as discussed above.

Applications of Naive Bayes

- **Real-time prediction**: Because Naive Bayes is fast and it's based on Bayesian statistics, it works well at making predictions in real-time. In fact, a lot of popular real-time models or **online** models are based on Bayesian statistics.
- **Multiclass prediction**: As previously stated, Naive Bayes works well when there are more than two classes for the output variable.
- **Text classification**: Text classification also includes sub-applications like spam filtering and sentiment analysis. Since Naive Bayes works best with discrete variables, it tends to work well in these applications.
- **Recommendation systems**: Naive Bayes is commonly used alongside other algorithms like Collaborative Filtering to build recommendations systems like Netflix's recommended for you section, or Amazon's recommended products, or Spotify's recommended songs.

Applications of Naive Bayes

Face Recognition

As a classifier, it is used to identify the faces or its other features, like nose, mouth, eyes, etc.

Weather Prediction

It can be used to predict if the weather will be good or bad.

Medical Diagnosis

Doctors can diagnose patients by using the information that the classifier provides. Healthcare professionals can use Naive Bayes to indicate if a patient is at high risk for certain diseases and conditions, such as heart disease, cancer, and other ailments.

News Classification

With the help of a Naive Bayes classifier, Google News recognizes whether the news is political, world news, and so on.