Naive Bayes Classifier

Table of Contents:

1. What is Naive Bayes
2. Understanding Naive Bayes and Machine Learning
3. Where is Naive Bayes Used
4. Understanding Naive Bayes Classifier
5. Gaussian Naive Bayes algorithm
6. Multinomial Naive Bayes algorithm
7. Advantages of Naive Bayes Classifier
8. Reference

Overview:

Have you ever wondered how your email provider implements spam filtering Or how online news channels perform news text classification Or how companies perform sentiment analysis of their audience on social media?

With a machine learning algorithm called a Naive Bayes classifier, you can do all of these things.

Naive Bayes Classifier

Let's start with a basic introduction to the Bayes theorem, named after Thomas Bayes from the 1700s. The Naive Bayes classifier works on the principle of conditional probability, as given by the Bayes theorem.

Let us go through some of the simple concepts of probability that we will use. 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,

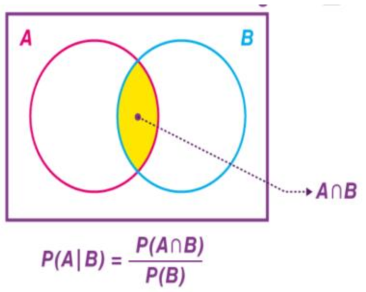
S :{HH, HT, TH, TT}

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

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.

According to Bayes theorem:



Let P(A) be the probability of occurrence of event A, P(B) be the probability of occurrence of event B and P(A∩B) be the probability of happening of both A and B.

P(A|B) formula is given by,

$P(A|B) = P(A \cap B)/P(B)$

$P(B|A) = P(A \cap B)/P(A)$

Naive Bayes Classifier

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.

Like with any of our other machine learning tools, it's important to understand where the Naive Bayes fits in the hierarchy.

Understanding Naive Bayes and Machine Learning

Machine learning falls into two categories:

- Supervised learning
- Unsupervised learning

Supervised learning falls into two categories:

- Classification
- Regression

Naive Bayes algorithm falls under classification.

Naive Bayes Classifier

Where is Naive Bayes Used?

You can use Naive Bayes for the following things:

- **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.

As the Naive Bayes Classifier has so many applications, it's worth learning more about how it works.

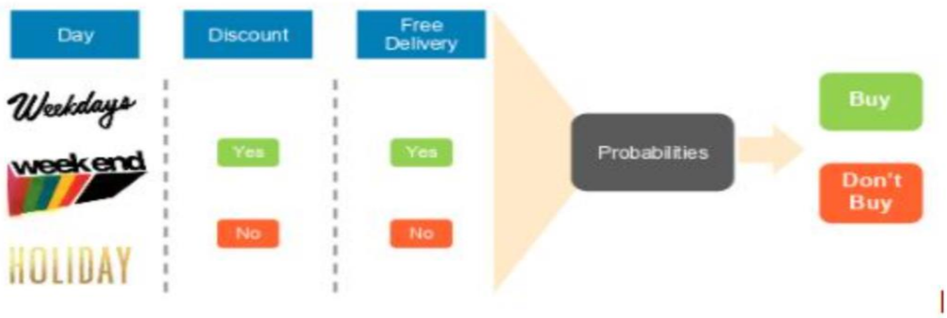
Naive Bayes Classifier

Based on the Bayes theorem, the Naive Bayes Classifier gives the conditional probability of an event A given event B.

Let us use the following demo to understand the concept of a Naive Bayes classifier:

Case Study: Shopping Example

Problem statement: To predict whether a person will purchase a product on a specific combination of day, discount, and free delivery using a Naive Bayes classifier.



Under the day, look for variables, like weekday, weekend, and holiday. For any given day, check if there are a discount and free delivery. Based on this information, we can predict if a customer would buy the product or not.

See a small sample data set of 30 rows, with 15 of them, as shown below:

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

customer_id	first_name	last_name	birth_date	phone	address	city	state	points
1	Babara	MacCaffrey	1986-03-28	781-932-9754	0 Sage Terrace	Waltham	MA	2273
2	Ines	Brushfield	1986-04-13	804-427-9456	14187 Commercial Trail	Hampton	VA	947
3	Freddi	Boagey	1985-02-07	719-724-7869	251 Springs Junction	Colorado Springs	CO	2967
4	Ambur	Roseburgh	1974-04-14	407-231-8017	30 Arapahoe Terrace	Orlando	FL	457
5	Clemmie	Betchley	1973-11-07	NULL	5 Spohn Circle	Arlington	TX	3675
6	Elka	Twiddell	1991-09-04	312-480-8498	7 Manley Drive	Chicago	IL	3073
7	Ilene	Dowson	1964-08-30	615-641-4759	50 Lillian Crossing	Nashville	TN	1672

Based on the dataset containing the three input types—day, discount, and free delivery— the frequency table for each attribute is populated.

For Bayes theorem, let the event 'buy' be A and the independent variables (discount, free delivery, day) be B.

Frequency Table		Buy	
		Yes	No
Discount	Yes	19	1
	No	5	5

Frequency Table		Buy	
		Yes	No
Free Delivery	Yes	21	2
	No	3	4

Frequency Table		Buy	
		Yes	No
Day	Weekday	9	2
	Weekend	7	1
	Holiday	8	3

We get a total of:
11 weekdays
8 weekends
11 holidays
The total number of days adds up to 30 days.

There are nine out of 24 purchases on weekdays

There are seven out of 24 purchases on weekends

There are eight out of 24 purchases on holidays

Based on the above likelihood table, let us calculate some conditional probabilities:

P(B) = P(Weekday)
= 11/30
= 0.37

P(A) = P(No Buy)
= 6/30
= 0.2

P(B | A)
= P(Weekday | No Buy)
= 2/6
= 0.33

P(A | B)
= P(No Buy | Weekday)
= P(Weekday| No Buy) * P(No Buy) / P(Weekday)
= (0.33 * 0.2) / 0.37
= 0.18

The probability of purchasing on the weekday = 11/30 or 0.37

It means out of the 30 people who came into the store throughout the weekend, weekday, and holiday, 11 of those purchases were made on weekdays.

The probability of not making a purchase = 6/30 or 0.2. There's a 20 percent chance that they're not going to make a purchase, no matter what day of the week it is.

Finally, we look at the probability of B (i.e., weekdays) when no purchase occurs.

The probability of the weekday without a purchase = 0.18 or 18 percent. As the probability of (No | Weekday) is less than 0.5, the customer will most likely buy the product on a weekday. Next, let's see how the table and conditional probabilities work in the Naive Bayes Classifier.

Frequency Table		Buy	
		Yes	No
Day	Weekday	3	7
	Weekend	8	2
	Holiday	9	1

Likelihood Table		Buy		
		Yes	No	
Day	Weekday	9/24	2/6	11/30
	Weekend	7/24	1/6	8/30
	Holiday	8/24	3/6	11/30
		24/30	6/30	

Frequency Table		Buy	
		Yes	No
Discount	Yes	19	1
	No	5	5

Likelihood Table		Buy		
		Yes	No	
Discount	Yes	19/24	1/6	20/30
	No	5/24	5/6	10/30
		24/30	6/30	

Frequency Table		Buy	
		Yes	No
Free Delivery	Yes	21	2
	No	3	4

Likelihood Table		Buy		
		Yes	No	
Free Delivery	Yes	21/24	2/6	23/30
	No	3/24	4/6	7/30
		24/30	6/30	

The likelihood tables can be used to calculate whether a customer will purchase a product on a specific combination of the day when there is a discount and whether there is free delivery. Consider a combination of the following factors where B equals:

- Day = Holiday
- Discount = Yes
- Free Delivery = Yes

Let us find the probability of them not purchasing based on the conditions above.

A = No Purchase

Applying Bayes Theorem, we get P(A | B) as shown:

$$P(A|B) = P(\text{No Buy} \mid \text{Discount} = \text{Yes}, \text{Free Delivery} = \text{Yes}, \text{Day} = \text{Holiday})$$

$$= \frac{P(\text{Discount} = \text{Yes} \mid \text{No}) * P(\text{Free Delivery} = \text{Yes} \mid \text{No}) * P(\text{Day} = \text{Holiday} \mid \text{No}) * P(\text{No Buy})}{P(\text{Discount=Yes}) * P(\text{Free Delivery=Yes}) * P(\text{Day=Holiday})}$$

$$= \frac{(1/6) * (2/6) * (3/6) * (6/30)}{(20/30) * (23/30) * (11/30)}$$

$$= 0.178$$

Similarly, let us find the probability of them purchasing a product under the conditions above.

Here, A = Buy

Applying Bayes Theorem, we get P(A | B) as shown:

$$P(A|B) = P(\text{Yes Buy} \mid \text{Discount} = \text{Yes}, \text{Free Delivery} = \text{Yes}, \text{Day} = \text{Holiday})$$

$$= \frac{P(\text{Discount} = \text{Yes} \mid \text{Yes}) * P(\text{Free Delivery} = \text{Yes} \mid \text{Yes}) * P(\text{Day} = \text{Holiday} \mid \text{Yes}) * P(\text{Yes Buy})}{P(\text{Discount=Yes}) * P(\text{Free Delivery=Yes}) * P(\text{Day=Holiday})}$$

$$= \frac{(19/24) * (21/24) * (8/24) * (24/30)}{(20/30) * (23/30) * (11/30)}$$

$$= 0.986$$

From the two calculations above, we find that:

Probability of purchase = 0.986

Probability of no purchase = 0.178

Finally, we have a conditional probability of purchase on this day.

Next, normalize these probabilities to get the likelihood of the events:

Sum of probabilities = 0.986 + 0.178 = 1.164
Likelihood of purchase = 0.986 / 1.164 = 84.71 percent
Likelihood of no purchase = 0.178 / 1.164 = 15.29 percent

Result: As 84.71 percent is greater than 15.29 percent, we can conclude that an average customer will buy on holiday with a discount and free delivery.

Naive Bayes Classifier

Gaussian Naive Bayes¶

`GaussianNB` implements the Gaussian Naive Bayes algorithm for classification. The likelihood of the features is assumed to be Gaussian:

$$P(x_i \mid y) = \frac{1}{\sqrt{2\pi\sigma_y^2}} \exp\left(-\frac{(x_i - \mu_y)^2}{2\sigma_y^2}\right)$$

The parameters σ_y and μ_y are estimated using maximum likelihood.

Features of Gaussian Naive Bayes:

- Based on Bayes Theorem.
- Assumes independence between features.
- Uses Gaussian distribution for each feature.
- Fast and efficient for large datasets.
- Mainly used for binary and multiclass classification problems.

Advantages:

- Simple and easy to implement.
- Fast training and prediction times.
- Works well with large datasets.
- Can handle a mix of continuous and categorical features.
- Can perform well when the independence assumption holds true.

Disadvantages:

- Strong independence assumption between features may not always hold true.
- Sensitive to irrelevant or noisy features.
- Limited to linear decision boundaries.
- All features must be numerical.

Naive Bayes Classifier

Multinomial Naive Bayes¶

Multinomial Naive Bayes is a variation of the Naive Bayes algorithm that is specifically designed for text classification tasks. Unlike Gaussian Naive Bayes, which assumes a Gaussian distribution for each feature, Multinomial Naive Bayes models the features as occurrences of terms in the documents. The algorithm works by counting the number of occurrences of each word (or term) in each document and using these counts to estimate the probability of each class. This makes it well-suited for tasks such as document classification, sentiment analysis, and spam filtering, where the features are typically represented as word counts or frequencies.

Features of Multinomial Naive Bayes:

- Specifically designed for text classification tasks.
- Models features as term occurrences in documents.
- Uses word counts or frequencies as input features.
- Handles both continuous and discrete features.
- Fast and efficient for large datasets.

Advantages:

- Simple and easy to implement.
- Fast training and prediction times.
- Scales well to large datasets.
- Can handle both continuous and discrete features.
- Good performance on text classification tasks.

Disadvantages:

- Assumes independence between features, which may not hold true.
- Sensitive to irrelevant or noisy features.
- May not perform well on tasks with a high-dimensional feature space.
- Not well-suited for tasks with continuous features.

Advantages of Naive Bayes Classifier

The following are some of the benefits of the Naive Bayes classifier:

- It is simple and easy to implement.
- It doesn't require as much training data.
- It handles both continuous and discrete data.
- It is highly scalable with the number of predictors and data points.
- It is fast and can be used to make real-time predictions.
- It is not sensitive to irrelevant features.

Disadvantages of Naive Bayes Classifier

- If your test data set has a categorical variable of a category that wasn't present in the training data set, the Naive Bayes model will assign it zero probability and won't be able to make any predictions in this regard. This phenomenon is called 'Zero Frequency,' and you'll have to use a smoothing technique to solve this problem.

- This algorithm is also notorious as a lousy estimator. So, you shouldn't take the probability outputs of 'predict_proba' too seriously.

- It assumes that all the features are independent. While it might sound great in theory, in real life, you'll hardly find a set of independent features.

Reference

- Simplilearn
- Analytical Vidhya
- Greeks for greeks
- sklearn