CS 456 Data Mining

Central Washington University

COMPUTER SCIENCE
CENTRAL WASHINGTON UNIVERSITY

Today …

- Naïve Bayes
- Multinomial Naïve Bayes

Naïve Bayes

- Naïve Bayes uses the Bayes' Theorem and assumes that all predictors are independent.

- In other words, this classifier assumes that the presence of one particular feature in a class doesn't affect the presence of another one.

Naïve Bayes Example

Example:

- You’d consider fruit to be orange if it is round, orange, and is of around 3.5 inches in diameter.
- Now, even if these features require each other to exist, they all contribute independently to your assumption that this particular fruit is orange.
- That’s why this algorithm has ‘Naive’ in its name.

Naïve Bayes Equation

P (c|x) = P(x|c) * P(c) / P(x)

P(c|x) = P(x1 | c) * P(x2 | c) * … P(xn | c) * P(c)

- Here, P(c|x) is the posterior probability according to the predictor (x) for the class(c).
- P(c) is the prior probability of the class, P(x) is the prior probability of the predictor, and P(x|c) is the probability of the predictor for the particular class(c).

How does Naive Bayes Work?

Suppose we want to find stolen cars and have the following dataset:

Serial No.	Color	Type	Origin	Was it Stolen?
1	Red	Sports	Domestic	Yes
2	Red	Sports	Domestic	No
3	Red	Sports	Domestic	Yes
4	Yellow	Sports	Domestic	No
5	Yellow	Sports	Imported	Yes
6	Yellow	SUV	Imported	No
7	Yellow	SUV	Imported	Yes
8	Yellow	SUV	Domestic	No
9	Red	SUV	Imported	No
10	Red	Sports	Imported	Yes

How does Naive Bayes Work?

According to our dataset, we can understand that our algorithm makes the following assumptions:

•It assumes that every feature is independent. For example, the color 'Yellow' of a car has nothing to do with its Origin or Type.

•It gives every feature the same level of importance. For example, knowing only the Color and Origin would predict the outcome correctly. That's why every feature is equally important and contributes equally to the result.

•Now, with our dataset, we have to classify if thieves steal a car according to its features. Each row has individual entries, and the columns represent the features of every car. In the first row, we have a stolen Red Sports Car with Domestic Origin. We'll find out if thieves would steal a Red Domestic SUV or not (our dataset doesn't have an entry for a Red Domestic SUV).

How does Naive Bayes Work?

We can rewrite the Bayes Theorem for our example as:

P(y | X) = [P(X | y)* P(y)] / P(X)
Here, y stands for the class variable (Was it Stolen?) to show if the thieves stole the car not according to the conditions. X stands for the features.

X = x1, x2, x3, …., xn)

Here, x1, x2,…, xn stand for the features. We can map them to be Type, Origin, and Color. Now, we'll replace X and expand the chain rule to get the following:

P(y | x1, …, xn) = [P(x1 | y) P(x2 | y) … P(xn | y) P(y)]/[P(x1) P (x2) … P(xn)]

How does Naive Bayes Work?

You can get the values for each by using the dataset and putting their values in the equation. The denominator will remain static for every entry in the dataset to remove it and inject proportionality.

$P(y \mid x1, \ldots, xn) \propto P(y)\ i = 1nP(xi \mid y)$

In our example, y only has two outcomes, yes or no.

$y = argmaxyP(y)\ i = 1nP(xi \mid y)$

We can create a Frequency Table to calculate the posterior probability $P(y|x)$ for every feature. Then, we'll mold the frequency tables to Likelihood Tables and use the Naive Bayesian equation to find every class's posterior probability. The result of our prediction would be the class that has the highest posterior probability.

Likelihood and Frequency Tables:

Frequency Table of Color:

Color	Was it Stolen (Yes)	Was it Stolen (No)
Red	3	2
Yellow	2	3

Likelihood Table of Color:

Color	Was it Stolen [P(Yes)]	Was it Stolen [P(No)]
Red	3/5	2/5
Yellow	2/5	3/5

Likelihood and Frequency Tables:

Frequency Table of Type:

Type	Was it Stolen (Yes)	Was it Stolen (No)
Sports	4	2
SUV	1	3

Likelihood Table of Type:

Type	Was it Stolen [P(Yes)]	Was it Stolen [P(No)]
Sports	4/5	2/5
SUV	1/5	3/5

Likelihood and Frequency Tables:

Frequency Table of Origin:

Origin	Was it Stolen (Yes)	Was it Stolen (No)
Domestic	2	3
Imported	3	2

Likelihood Table of Origin :

Origin	Was it Stolen [P(Yes)]	Was it Stolen [P(No)]
Domestic	2/5	3/5
Imported	3/5	2/5

Naïve Bayes Prediction

Our problem has 3 predictors for X, so according to the equations we saw previously, the posterior probability P(Yes | X) would be as follows:

P(Yes | X) = P(Red | Yes) * P(SUV | Yes) * P(Domestic | Yes) * P(Yes)

$= \frac{3}{5} \times 1/5 \times \frac{2}{5} \times 0.5$

= 0.024

P(No | X) would be:

P(No | X) = P(Red | No) * P(SUV | No) * P(Domestic | No) * P(No)

$= \frac{2}{5} \times 3/5 \times \frac{3}{5} \times 0.5$

= 0.072

So, as the posterior probability P(No | X) is higher than the posterior probability P(Yes | X), our Red Domestic SUV will have ‘No’ in the ‘Was it stolen?’ section.

What is Multinomial Naïve Bayes?

- Multinomial Naive Bayes algorithm is a probabilistic learning method that is mostly used in Natural Language Processing (NLP).
- The algorithm is based on the Bayes theorem and predicts the tag of a text such as a piece of email or newspaper article.
- It calculates the probability of each tag for a given sample and then gives the tag with the highest probability as output.

How Multinomial Naive Bayes works?

Bayes theorem calculates the probability of an event occurring based on the prior knowledge of conditions related to an event.

Formula:

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

Where we are calculating the probability of class A when predictor B is already provided.

P(B) = prior probability of B

P(A) = prior probability of class A

P(B|A) = occurrence of predictor B given class A probability

This formula helps in calculating the probability of the tags in the text.

Multinomial Naive Bayes Example

Training Data Set

Weather	Sunny	Overcast	Rainy	Sunny	Sunny	Overcast	Rainy	Rainy
Play	No	Yes	Yes	Yes	Yes	Yes	No	No

Multinomial Naive Bayes Example

This can be easily calculated by following the below given steps:
Create a frequency table of the training data set given in the above problem statement. List the count of all the weather conditions against the respective weather condition.

Weather	Yes	No
Sunny	3	2
Overcast	4	0
Rainy	2	3
Total	9	5

Multinomial Naive Bayes Example

Find the probabilities of each weather condition and create a likelihood table.

Weather	Yes	No	
Sunny	3	2	=5/14(0.36)
Overcast	4	0	=4/14(0.29)
Rainy	2	3	=5/14(0.36)
Total	9	5	
	=9/14 (0.64)	=5/14 (0.36)	

(3+2) / (9+5)

Multinomial Naive Bayes Example

Calculate the posterior probability for each weather condition using the Naive Bayes theorem. The weather condition with the highest probability will be the outcome of whether the players are going to play or not.

Use the following equation to calculate the posterior probability of all the weather conditions:

$$P(A|B) = P(A) * P(B|A)/P(B)$$

Multinomial Naive Bayes Example

After replacing variables in the above formula, we get:

P(Yes|Sunny) = P(Yes) * P(Sunny|Yes) / P(Sunny)

Take the values from the above likelihood table and put it in the above formula.

P(Sunny|Yes) = 3/9 = 0.33, P(Yes) = 0.64 and P(Sunny) = 0.36

Hence, P(Yes|Sunny) = (0.64*0.33)/0.36 = 0.60

Weather	**Yes**	**No**	
Sunny	3	2	=5/14(0.36)
Overcast	4	0	=4/14(0.29)
Rainy	2	3	=5/14(0.36)
Total	9	5	
	=9/14 (0.64)	=5/14 (0.36)	

Multinomial Naive Bayes Example

P(No|Sunny) = P(No) * P(Sunny|No) / P(Sunny)

Take the values from the above likelihood table and put it in the above formula.

P(Sunny|No) = 2/5 = 0.40, P(No) = 0.36 and P(Sunny) = 0.36
P(No|Sunny) = (0.36*0.40)/0.36 = 0.6 = 0.40

The probability of playing in sunny weather conditions is higher. Hence, the player will play if the weather is sunny.

Similarly, we can calculate the posterior probability of rainy and overcast conditions, and based on the highest probability; we can predict whether the player will play.