Statistics and Probability: Naïve Bayes

Introduction no Naïve Bayes

- Naive Bayes is a classification algorithm based on Bayes' theorem.
- It assumes that features are conditionally independent of each other given the class label.
- It builds a probability model by estimating the probabilities of features given each class label.
- Naive Bayes is primarily used for classification tasks and predicts the class label with the highest probability.
- It can handle both categorical and numerical features.
- Laplace smoothing is often applied to avoid zero probabilities.
- Naive Bayes is computationally efficient and requires a small amount of training data.
- It is commonly used in text classification tasks such as spam filtering and sentiment analysis.
- There are different variants of Naive Bayes, such as Multinomial Naive Bayes and Gaussian Naive Bayes.
- Naive Bayes can still perform well even if the independence assumption is violated to some extent.

Experimental Probability

Experimental Probability is found by repeating an experiment and observing the outcomes.

$$P(\text{event}) = \frac{\text{number of times event occurs}}{\text{total number of trials}}$$

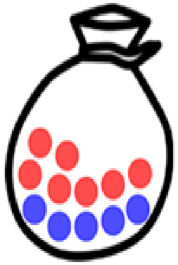
Example:

A coin is tossed 10 times.
A head is recorded 7 times
and a tail 3 times.

$$P(\text{head}) = \frac{7}{10} \qquad P(\text{tail}) = \frac{3}{10}$$

Simple Probability

$$\text{Probabilty} = \frac{\text{Favorable outcomes}}{\text{Total outcomes}}$$



Example:

$$P(red) = \frac{7}{12}$$

Number of red marbles

Total number of marbles (sample space)

$$P(blue) = \frac{5}{12}$$

Number of blue marbles

Total number of marbles (sample space)

Bayesian Classifier

Notation

$\boldsymbol{x} \in \mathbb{R}^d$: d-dimensional feature vector

y : class number (usually $y \in \{0, 1\}$ or $y \in \{-1, +1\}$)

$p(y)$: prior probability of pattern class y

$p(\boldsymbol{x})$: evidence (distribution of features in d-dimensional feature space)

$p(\boldsymbol{x}, y)$: joint probability density function (pdf)

$p(\boldsymbol{x}|y)$: class conditional density

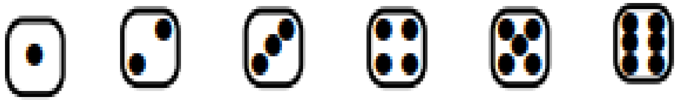
$p(y|\boldsymbol{x})$: posterior probability

Bayesian classifiers have been used in a wide range of applications, including email spam filtering, medical diagnosis, image recognition, and natural language processing.

Bayesian Classifier

Application

For the following observed counts of rolling either dice 1 (D1) or dice 2 (D1) multiple times:



	1	2	3	4	5	6
D1	9	12	7	11	7	11
D2	7	8	9	7	8	12

where $y \in \{D1, D2\}$ and $x \in \{1, 2, 3, 4, 5, 6\}$

(a) Estimate $P(x = 4)$. (11+7)/108

(b) Estimate $P(y = D2)$. 51/108

(c) Estimate $P(x = 4, y = D1)$. 11/108

(d) Estimate $P(x = 4 | y = D2)$. 7/51

Bayes' rule, also known as Bayes' theorem or Bayes' law, is a fundamental concept in probability theory. It describes how to update or revise the probability of an event based on new evidence or information. Mathematically, Bayes' rule is represented as:

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

In this equation:

- $P(A|B)$ is the posterior probability of event A given evidence B. It represents the probability of event A occurring given that evidence B is true.
- $P(B|A)$ is the conditional probability of evidence B given event A. It represents the probability of observing evidence B when event A is true.
- $P(A)$ is the prior probability of event A. It represents the initial or prior probability of event A before considering any evidence.
- $P(B)$ is the probability of evidence B. It represents the overall probability of observing evidence B, regardless of event A.

Bayesian Classifier

$$\underbrace{p(\mathbf{x}, y)}_{joint\ pdf} = \underbrace{p(y)}_{prior} \cdot \underbrace{p(\mathbf{x}|y)}_{class\ conditional\ pdf}$$

$$= \underbrace{p(\mathbf{x})}_{evidence} \cdot \underbrace{p(y|\mathbf{x})}_{posterior}$$

Now we get the posterior as follows:

$$p(y|\mathbf{x}) = \frac{p(y) \cdot p(\mathbf{x}|y)}{p(\mathbf{x})}$$

Now let us summarize the Bayesian decision rule:

We decide for the class y^* according to the decision rule

$$
\begin{aligned}
y^* &= \operatorname*{argmax}_y p(y|\boldsymbol{x}) \\
&= \operatorname*{argmax}_y \frac{p(y) \cdot p(\boldsymbol{x}|y)}{p(\boldsymbol{x})} \\
&= \operatorname*{argmax}_y p(y) \cdot p(\boldsymbol{x}|y) \\
&= \operatorname*{argmax}_y \{\log p(y) + \log p(\boldsymbol{x}|y)\}
\end{aligned}
$$

Using the $(0,1)$-loss function, the class decision is based on:

$$
\begin{aligned}
y^* &= \underset{y}{\operatorname{argmin}}\, \mathrm{AL}(\boldsymbol{x}, y) \\
&= \underset{y}{\operatorname{argmin}} \sum_{y'} l(y, y') \cdot p(y' | \boldsymbol{x}) \\
&= \underset{y}{\operatorname{argmax}}\, p(y | \boldsymbol{x})
\end{aligned}
$$

Naïve Bayes: Problems

Problems

Example No.	Color	Type	Origin	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

Problems

Dataset

Color	Type	Origin	Stolen?
Red	Sports	Domestic	Yes
Red	Sports	Domestic	No
Red	Sports	Domestic	Yes
Yellow	Sports	Domestic	No
Yellow	Sports	Imported	Yes
Yellow	SUV	Imported	No
Yellow	SUV	Imported	Yes
Yellow	SUV	Domestic	No
Red	SUV	Imported	No
Red	Sports	Imported	Yes

Frequency Table

		Stolen?	
		Yes	No
Color	Red	3	2
	Yellow	2	3

Likelihood Table

		Stolen?	
		P(Yes)	P(No)
Color	Red	3/5	2/5
	Yellow	2/5	3/5

Problems

Color	Type	Origin	Stolen?
Red	Sports	Domestic	Yes
Red	Sports	Domestic	No
Red	Sports	Domestic	Yes
Yellow	Sports	Domestic	No
Yellow	Sports	Imported	Yes
Yellow	SUV	Imported	No
Yellow	SUV	Imported	Yes
Yellow	SUV	Domestic	No
Red	SUV	Imported	No
Red	Sports	Imported	Yes

Color

Frequency Table

		Stolen?	
		Yes	No
Color	Red	3	2
	Yellow	2	3

Likelihood Table

		Stolen?	
		P(Yes)	P(No)
Color	Red	3/5	2/5
	Yellow	2/5	3/5

Problems

Color	Type	Origin	Stolen?
Red	Sports	Domestic	Yes
Red	Sports	Domestic	No
Red	Sports	Domestic	Yes
Yellow	Sports	Domestic	No
Yellow	Sports	Imported	Yes
Yellow	SUV	Imported	No
Yellow	SUV	Imported	Yes
Yellow	SUV	Domestic	No
Red	SUV	Imported	No
Red	Sports	Imported	Yes

Type

Frequency Table

		Stolen?	
		Yes	No
Type	Sports	4	2
	SUV	1	3

Likelihood Table

		Stolen?	
		P(Yes)	P(No)
Type	Sports	4/5	2/5
	SUV	1/5	3/5

Problems

Color	Type	Origin	Stolen?
Red	Sports	Domestic	Yes
Red	Sports	Domestic	No
Red	Sports	Domestic	Yes
Yellow	Sports	Domestic	No
Yellow	Sports	Imported	Yes
Yellow	SUV	Imported	No
Yellow	SUV	Imported	Yes
Yellow	SUV	Domestic	No
Red	SUV	Imported	No
Red	Sports	Imported	Yes

Origin

Frequency Table

		Stolen?	
		Yes	No
Origin	Domestic	2	3
	Imported	3	2

Likelihood Table

		Stolen?	
		P(Yes)	P(No)
Origin	Domestic	2/5	3/5
	Imported	3/5	2/5

What’s the Predicted Result?

Color	Type	Origin	Stolen
Red	SUV	Domestic	?

What’s the Predicted Result?

Now, The Question is -

Color	Type	Origin	Stolen
Red	SUV	Domestic	?

As per the equations discussed above, we can calculate the posterior probability P(Yes | X) as :

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

= 3/5 * 1/5 * 2/5 * 0.5
= 0.0092

What’s the Predicted Result?

Now, The Question is -

Color	Type	Origin	Stolen
Red	SUV	Domestic	?

P(No) >> P(Yes)
Output: No

As per the equations discussed above, we can calculate the posterior probability P(Yes | X) as :

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

= 3/5 * 1/5 * 2/5 * 0.5
= 0.0092

and P(No | X):

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

= 2/5 * 3/5 * 3/5 * 0.5
= 0.072

Online Learning Vs. Batch Learning

Online Learning Vs. Batch Learning

Online learning and batch learning are two different approaches to training machine learning models.

1. Batch Learning:

- Batch learning refers to training a machine learning model on the entire dataset all at once.
- The model is trained on a fixed set of data, and the parameters of the model are updated only after processing the entire dataset.
- It requires a lot of memory as the entire dataset needs to be loaded into memory.
- Computationally, it can be more expensive, especially for very large datasets.
- It tends to be slower and less flexible because the model has to be retrained from scratch whenever new data is available.
- Batch learning is suitable for scenarios where the data doesn't change much over time and you have the resources to process the entire dataset at once.

Online Learning Vs. Batch Learning

2. Online Learning:

- Online learning, also known as incremental learning or streaming learning, involves training a model on small batches of data sequentially, one at a time.
- The model updates its parameters after each batch, allowing it to quickly adapt to new information.
- It requires less memory since only a small batch of data needs to be stored at any given time.
- It can be computationally less expensive, especially when dealing with large datasets.
- It is more adaptive and flexible, as new data can continuously update the model.
- Online learning is suitable for scenarios where the data is constantly changing and you want the model to adapt quickly to new information.

Online Learning Vs. Batch Learning

When to Use Each?

- Use batch learning: when you have a relatively stable dataset, and you can afford the computational resources needed to process the entire dataset at once. This is common in scenarios where the data doesn’t change frequently, and you want to perform periodic model updates.

- Use online learning: when you have a dynamic and changing dataset, and you need your model to adapt quickly to new information. This is common in scenarios like real-time prediction systems, and recommendation systems, or in situations where the data is too large to process in one go.

In practice, a hybrid approach is often used, where models are initially trained using batch learning and then fine-tuned using online learning to adapt to changing conditions. This combines the benefits of both approaches

Let’s do it with Python -