Naive Bayes

Naive Bayes is a probabilistic machine learning algorithm used primarily for solving classification problems. It is based on Bayes' theorem, which calculates the probability of an event occuring given evidence. Despite its simplicity, it's surprisingly effective in many real-world applications like text classification, spam detection, sentiment analysis and more.

Bayes Theorem: Bayes Theorem provides a way to update the probability of a hypothesis given new evidence. It is expressed as —

$$P(H|E) = \frac{P(E|H) * P(H)}{P(E)}$$

($P(H|E)$: posterior probability; $P(E|H)$: likelihood; $P(H)$: prior probability; $P(E)$: marginal likelihood.)

- $P(H|E)$ is the posterior probability : the probability of hypothesis H being true given the evidence E.
- $P(E|H)$ is the likelihood: the probability of the evidence E given that hypothesis H is true.
- $P(H)$ is the prior probability : the initial probability of the hypothesis H before seeing the evidence.
- $P(E)$ is the marginal likelihood: the probability of evidence (E) under all possible hypotheses.

The Naive Assumption: The "naive" part of Naive Bayes assumes that all features (or predictors) are **independent** of each other, given the class label. This is often unrealistic in real-world scenarios but simplifies the computation, making the model fast and efficient.

Working of Naive Bayes Classifier

Given the class variable 'y_k' and dependent feature vector $x_1, x_2, \dots, x_n$:

$$P(y_k|x_1, x_2, x_3 \dots, x_n) = \frac{P(x_1, x_2, x_3, \dots, x_n|y_k)P(y_k)}{P(x_1, x_2, x_3, \dots, x_n)}$$

Using our naive conditional independence assumption-

$$P(x_i | y_k, x_1, x_2, \dots, x_n) = P(x_i | y_k)$$

for all such i, the relationship is simplified to

$$P(y_k | x_1, x_2, x_3, \dots, x_n) = \frac{P(y_k) \prod_{i=1}^{n} P(x_i | y_k)}{P(x_1, x_2, \dots, x_n)}$$

Since $P(x_1, x_2, \dots, x_n)$ is constant given the input, we can use the following classification rule:

$$P(y_k | x_1, x_2, \dots, x_n) \propto P(y_k) \cdot \prod_{i=1}^{n} P(x_i | y_k)$$

$$\Rightarrow \boxed{\hat{y} = \underset{y_k}{\arg\max}\ P(y_k) \prod_{i=1}^{n} P(x_i | y_k)}$$

and we can use Maximum A Posteriori (MAP) estimation to estimate $P(y)$ and $P(x_i|y)$; the former is then the relative frequency of class y in the training set.

→ $P(y_k)$ is the prior probability of class y_k.

→ $P(x_i | y_k)$ is the likelihood of feature x_i given class y_k. We calculate this for each class y_k and choose the class with highest posterior probability.

Eg:1 Let's understand the working of Naive Bayes Classifier using an example-

Outlook	Temperature	Humidity	Wind	Play Tennis
Sunny	Hot	High	Weak	No
Sunny	Hot	High	Strong	No
Overcast	Hot	High	Weak	Yes
Rain	Mild	High	Weak	Yes
Rain	Cool	Normal	Weak	Yes
Rain	Cool	Normal	Strong	No
Overcast	Cool	Normal	Strong	Yes
Sunny	Mild	High	Weak	No
Sunny	Cool	Normal	Weak	Yes
Rain	Mild	Normal	Weak	Yes
Sunny	Mild	Normal	Weak	Yes
Overcast	Mild	Normal	Strong	Yes
Overcast	Hot	High	Strong	Yes
Rain	Mild	Normal	Weak	Yes
		High	Strong	No

In this example, based on the given dataset find the probability a person will be able to play tennis given outlook = Sunny, temperature = cool, humidity high, wind = strong.

Solution: We need to find

$$P(y=\text{Yes} \mid \text{Outlook=sunny, temperature=cool, humidity=high, wind=strong}) = [P(\text{Outlook=sunny}/\text{yes}) * P(\text{temp=cool}/\text{yes}) * P(\text{hum=high}/\text{yes}) * P(\text{wind=strong}/\text{yes})] * P(\text{yes})$$

Similarly,

$$P(y=\text{No} \mid \text{Out=sunny, temp=cool, humid=high, wind=strong}) = [P(\text{Out=sunny}\mid\text{No}) * P(\text{temp=cool}\mid\text{No}) * P(\text{humid=high}\mid\text{No}) * P(\text{wind=strong}\mid\text{No})] * P(\text{No})$$

Step 1: We calculate the prior probabilities, P(yes) & P(No)

$$P(\text{yes}) = \frac{9}{14}, \quad P(\text{No}) = \frac{5}{14}$$

Step 2: We calculate the likelihoods for each column one by one—

Outlook	Yes	No
Sunny	2/9	3/5
Overcast	4/9	0
Rain	3/9	2/5

Temperature	Yes	No
Hot	2/9	2/5
Mild	4/9	2/5
Cool	3/9	1/5

Humidity	Yes	No
High	3/9	4/5
Normal	6/9	1/5

Wind	Yes	No
Weak	6/9	2/5
Strong	3/9	3/5

$$P(Y=\text{yes}) = P(\text{outlook=sunny}\mid\text{Yes}) * P(\text{temp=cool}\mid\text{yes}) * P(\text{Humid=high}\mid\text{yes}) * P(\text{Wind=strong}\mid\text{yes}) * P(\text{Yes})$$

$$= \frac{2}{9} * \frac{3}{9} * \frac{3}{9} * \frac{3}{9} * \frac{9}{14} = 0.0053$$

$$P(y=\text{no}) = P(\text{Out=sunny}\mid\text{No}) * P(\text{temp=cool}\mid\text{No}) * P(\text{Humid=high}\mid\text{No}) * P(\text{wind=strong}\mid\text{No}) * P(\text{No})$$

$$= 3/5 * 1/5 * 4/5 * 3/5 * \frac{5}{14} = \frac{36}{1750} = 0.0205$$

Since P(No) > P(Yes), therefore we will predict the person will not be able to play tennis.

Eg:2 Given a dataset of cars being stolen or not, predict a car will get stolen given colour is Red, Type is SUV Origin is domestic.

Colour	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

Soln:

$P(Yes \mid Colour=Red, Type=SUV, Origin=Domestic) = P(Yes) * P(Colour=Red \mid Yes) * P(Type=SUV \mid Yes) * P(origin=dome \mid Yes)$

and, $P(No \mid col=Red, Ty=SUV, Origin=Domestic) = P(No) * P(Col=Red \mid No) * P(Ty=SUV \mid No) * P(origin=domestic \mid No)$

probability

Step1: We first calculate the prior and likelihoods —

$P(Yes) = 5/10$, $P(No) = 5/10$

Colour	Yes	No
Red	3/5	2/5
Yellow	2/5	3/5

Type	Yes	No
Sports	4/5	2/5
SUV	1/5	3/5

Origin	Yes	No
Domestic	2/5	3/5
Imported	3/5	2/5

$\therefore P(Yes \mid all\ conditions) = \frac{5}{10} * \frac{3}{5} * \frac{1}{5} * \frac{2}{5} = 0.024$

$P(No \mid all\ conditions) = \frac{5}{10} * \frac{2}{5} * \frac{3}{5} * \frac{3}{5} = 0.072$

$\because P(No) > P(Yes)$, we will say car will not be stolen.

Types of Naive Bayes Classifier : There are several types of Naive Bayes Classifier depending on how the likelihood $P(x_i/y_k)$ gets calculated.

Gaussian Naive Bayes: These are used when the features are numerical and continuous in nature, and assume a Gaussian (Normal) distribution. The likelihood is calculated using the probability density function of a normal distribution :

$$P(x_i|y_k) = \frac{1}{\sqrt{2\pi\sigma_k^2}} \exp\left(-\frac{(x_i - \mu_k)^2}{2\sigma_k^2}\right)$$

where μ_k and σ_k^2 are mean and standard deviation of feature x_i for class y_k.

Eg:3 Based on the given data, determine the gender of a person having height 6 ft, weight 130 pounds, and foot size 8 inches.

Height	Weight	Foot size	Gender
6.00	180	12	Male
5.92	190	11	Male
5.58	170	12	Male
5.92	165	10	Male
5.00	100	6	Female
5.50	150	8	Female
5.42	130	7	Female
5.75	150	9	Female

Sol^n: Step 1: We find the prior probabilities

$P(\text{Male}) = 4/8$, $P(\text{Female}) = 4/8$

Step 2: For each feature we need to calculate their likelihoods, so we will first calculate mean and variance for all features one by one for each class.

Male: Mean (Height) $= \frac{6+5.92+5.58+5.92}{4} = 5.855$

Variance (Height) $= \frac{\sum_i (x_i - \bar{x})^2}{n-1} = 0.035055$

Female: Mean (Height) $= \frac{5+5.5+5.42+5.75}{4} = 5.4175$

Variance (Height) $= \frac{\sum_i (x_i - \bar{x})^2}{n-1} = \frac{(5-5.4175)^2+(5.5-5.4175)^2+(5.42-5.4175)^2+(5.75-5.4175)^2}{4-1}$

Variance (Height) $= 0.097225$

Similarly for Weight —

Male: Mean (Weight) = 176.25
Variance (Weight) = 122.92

Female: Mean (Weight) = 132.5
Variance (Weight) = 558.33

Similarly for foot size —

Male: Mean (size) = 11.25
Variance (size) = 0.92

Female: Mean (size) = 7.5
Variance (size) = 1.67

Now, we have to calculate

$P(\text{Gender} = \text{Male} \mid \text{height} = 6\text{ft}, \text{weight} = 130\text{pounds}, \text{foot size} = 8\text{inches})$

and $P(\text{Gender} = \text{Female} \mid \text{height} = 6\text{ft}, w = 130\text{pounds}, \text{foot size} = 8\text{inches})$

$P(\text{Gender} = \text{Male}) = P(M) * P(H|M) * P(W|M) * P(S|M)$

$P(\text{Gender} = \text{Female}) = P(F) * P(H|F) * P(W|F) * P(S|F)$

Now, we need to calculate the likelihood for each numerical feature using the probability density function for a gaussian (normal) distribution:

$\therefore P(H|M) = \frac{1}{\sqrt{2\times 3.14 * 0.035}} * e^{\frac{-(6-5.855)^2}{2*0.035}} = 1.5789$

$P(W|M) = 5.9881\, e^{-6}$, $P(S|M) = 1.3112e^{-3}$

$M(H|F) = 2.2346e^{-1}$, $P(W|F) = 1.6789\, e^{-2}$, $P(S|F) = 2.8669e^{-1}$

$\therefore P(M) = 0.5 * 1.5789 * 5.9881e^{-6} * 1.3112e^{-3} = 6.1984e^{-9}$

$P(F) = 0.5 * 2.2346e^{-1} * 1.6789e^{-2} * 2.8669e^{-1} = 5.377e^{-4}$

Since $P(F) > P(M)$, so the new example will be classified as Female.

(b) **Multinomial Naive Bayes :** It is typically used for discrete data, such as word counts in text classification. The likelihood $P(X_i|C_K)$ is proportional to the frequency of feature X_i within class C_K.

(c) **Bernoulli Naive Bayes :** It is suitable for binary / boolean features or distribution which follows bernoulli's distribution. The likelihood $P(X_i|C_K)$ represents the probability of feature X_i being 1 (present) or 0 (absent) in class C_K.

* **Laplace Smoothing :** Laplace smoothing (additive smoothing) is a technique used to handle the zero probability problem. It involves adding a small constant (usually 1) to the count of each feature during likelihood estimation to ensure that no probability is ever zero.

For Eg. if we are estimating the likelihood of a word in text classification :

$$P(\text{word}|C_K) = \frac{\text{count of (word in class } C_K) + 1}{\text{total words in class } C_K + V}$$

where V is the total number of unique words (vocab size).

Eg:4 Given the dataset as shown below, predict the output value for new instance where outlook = overcast, temperature = 60, humidity = 62 and windy = false.

Outlook	Temp	Humidity	Wind	Play Tennis
Sunny	85	85	False	No
Sunny	80	90	True	No
Overcast	83	86	False	Yes
Rainy	70	96	False	Yes
Rainy	68	80	False	Yes
Rainy	65	70	True	No
Overcast	64	65	True	Yes
Sunny	72	95	False	No
Sunny	69	70	False	Yes
Rainy	75	80	False	Yes
Sunny	75	70	True	Yes
Overcast	72	90	True	Yes
Overcast	81	75	False	Yes
Rainy	71	91	True	No

Soln: We need to calculate the following —
P(Yes | Outlook = overcast, temp = 60, humid = 62, wind = false)
& P(No | out = overcast, temp = 60, humid = 62, wind = False).

Step1: We calculate the prior probabilities,
$P(Yes) = \frac{9}{14}$, $P(No) = \frac{5}{14}$

Step2: Now, we calculate the likelihood for each feature —

Outlook	Yes	No	Wind	Yes	No
Sunny	2/9	3/5	True	3/9	3/5
Overcast	4/9	0/5	False	6/9	2/5
Rainy	3/9	2/5			

From these we can get,
$P(Sunny|Yes) = \frac{2}{9}$, $P(Sunny|No) = \frac{3}{5}$, $P(Rainy|Y) = \frac{3}{9}$, $P(Rainy|N) = \frac{2}{5}$
$P(W_T|Y) = \frac{3}{9}$, $P(W_F|Y) = \frac{6}{9}$, $P(W_T|N) = \frac{3}{5}$, $P(W_F|N) = \frac{2}{5}$
But $P(Overcast|Yes) = \frac{4}{9}$ & $P(Overcast|No) = 0$.
Now, when $P(Overcast|No)$ becomes 0, whole $P(No | any condition) = 0$, to overcome this limitation we make use of the Laplace Smoothing and add a constant value 1.

$\therefore$ $P(Overcast|No) = \frac{0+1}{5+3} = 1/8$ → since there are 3 unique categories in outlook

$P(Overcast|Yes) = \frac{4+1}{9+3} = \frac{5}{12}$

Now, we need to calculate the prior probabilities for the numerical columns as well, so we first find the mean and ~~standard dev~~ variance of both the features.
Mean(Temp|Yes) = 73, Variance(Temp|Y) = 6.2
Mean(Temp|No) = 74.6, Variance(Temp|No) = 8.0
Mean(Hum|Yes) = 79.1, Variance(Hum|Yes) = 10.2
Mean(Hum|No) = 86.2, Variance(Hum|No) = 9.7

Step 3: We now need to calculate -

$$p(temp=60|yes) = \frac{1}{6.2\sqrt{2\pi}} e^{-\frac{(60-73)^2}{2*(6.2)^2}} = 0.071$$

$$p(temp=60|No) = \frac{1}{8\sqrt{2\pi}} e^{-\frac{(60-74.6)^2}{2*8^2}} = 0.0094$$

$$p(humid=62|yes) = \frac{1}{10.2\sqrt{2\pi}} e^{-\frac{(62-79.1)^2}{2*(10.2)^2}} = 0.0096$$

$$p(humid=62|No) = \frac{1}{9.7\sqrt{2\pi}} e^{-\frac{(62-86.2)^2}{2*(9.7)^2}} = 0.0018$$

Step 4: $P(yes|E) = P(out=overcast|yes) * P(temp=60|yes) * P(humid=62|Y)$
$* P(wind=false|yes) * P(yes)$

$$= \frac{5}{12} * 0.071 * 0.0096 * \frac{6}{9} * \frac{9}{14} = 1.22 * 10^{-5}$$

$P(No|E) = P(out=overcast|No) * P(temp=60|No) * P(humid=62|No)$
$* P(wind=False|No) * P(No)$

$$= \frac{1}{8} * 0.0094 * 0.0018 * \frac{2}{5} * \frac{5}{14} = 3.02 * 10^{-7}$$

Since $P(yes|E) > P(No|E)$, so we classify the new instance as Yes.

Advantages of Naive Bayes:

1. It is simple to implement and computationally inefficient, especially for large datasets.
2. Training and prediction are fast due to the independence assumption and closed-form solutions for likelihoods.
3. Despite the naive assumption, it often performs well even on small amount of data.

Disadvantages of Naive Bayes:

1. The assumption that features are independent given the class label is rarely true in real-world data. Violations of this assumption can degrade performance.
2. If a certain feature X_i never appears in the training data for a particular class, then $P(X_i|C_k)$ will be zero, making entire product zero. This is prevented by using laplace Smoothing.
3. For continuous data, the Gaussian assumption might not always hold.