Bayes Classifier

Fundamental

- A **naïve classifier** is a basic classification algorithm that makes predictions based on simple assumptions or heuristics. It does not consider any underlying relationship between the features of the dataset. The term "naïve" reflects its simplicity and lack of sophistication.
- One of the most popular naïve classifiers is the **Naïve Bayes classifier**, which assumes:
- All features are **independent** of each other.
- Each feature contributes equally to the prediction.

Contd.

- **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**.

Why is it called Naïve 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.

Introduction

- The naive Bayes classifier is probably among the most effective algorithms for learning tasks to classify text documents.
- The naive Bayes technique is extremely helpful in case of huge datasets.
- For example, Google employs naive Bayes classifier to correct the spelling mistakes in the text typed in by users.
- it gives a meaningful perspective to the comprehension of various learning algorithms that do not explicitly manipulate probabilities.
- Bayes theorem is the cornerstone of Bayesian learning methods.

Bayes' Law

Bayes' Theorem:

- Bayes' theorem is also known as **Bayes' Rule** or **Bayes' law**, which is used to determine the probability of a hypothesis with prior knowledge. It depends on the conditional probability.
- The formula for Bayes' theorem is given as:

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

Where,

P(A|B) is Posterior probability: Probability of hypothesis A on the observed event B.

P(B|A) is Likelihood probability: Probability of the evidence given that the probability of a hypothesis is true.

- **P(A) is Prior Probability**: Probability of hypothesis before observing the evidence.
- **P(B) is Marginal Probability**: Probability of Evidence.

Bayes theorem

- Bayes theorem offers a method of calculating the probability of a hypothesis on the basis of its prior probability, the probabilities of observing different data given the hypothesis, and the observed data itself.
- The distribution of all possible values of discrete random variable y is expressed as probability distribution.

$$P(y) = \langle P(y_1), \ldots, P(y_M) \rangle$$

$$P(y_1) + \cdots + P(y_M) = 1$$

- We assume that there is some a priori probability (or simply prior) $P(y_q)$ that the next feature vector belongs to the class q.

Bayes theorem

- The continuous attributes are binned and converted to categorical variables.
- Therefore, each attribute x_j is assumed to have value set that are countable.
- Bayes theorem provides a way to calculate posterior $P(y_k \mid x);\ k \in \{1, \ldots, M\}$ from the known priors $P(y_q)$, together with known conditional probabilities $P(x \mid yq);\ q = 1, \ldots, M$.

$$P(y_k \mid \mathbf{x}) = \frac{P(y_k)\, P(\mathbf{x} \mid y_k)}{P(\mathbf{x})}$$

Directly, difficult to calculate

Using this relation, easier

$$P(\mathbf{x}) = \sum_{q=1}^{M} P(\mathbf{x} \mid y_q)\, P(y_q)$$

- $P(x)$ expresses variability of the observed data, independent of the class.

Naive Rule

- As per this rule, the record gets classified as a member of the majority class.
- Assume that there are six attributes in the data table
  - x1: Day of Week, x2: Departure Time, x3: Origin, x4: Destination,
  - x5: Carrier, x6: Weather
  - and output y gives class labels (Delayed, On Time).
- Say 82% of the entries in y column record ‘On Time’.
- A naive rule for classifying a flight into two classes, ignoring information on x1, x2, …, x6 is to classify all flights as being ‘On Time’.
- The naive rule is used as a baseline for evaluating the performance of more complicated classifiers.
- Clearly, a classifier that uses attribute information should outperform the naive rule.

Naive Bayes Classifier

- Takes into account the features as equally important and independent of each other, considering the class.
  - Not the scenario in real-life data.
- Each of the $P(y_q)$ may be estimated simply by counting the frequency with which class y_q occurs in the training data:

$$P(y_q) = \frac{\text{Number of data with class } y_q}{\text{Total number } (N) \text{ of data}}$$

- If the decision must be made with **so little information**, it seems logical to use the following rule: **(Just like Naive rule)**

$$\text{Decide } y_k \text{ if } P(y_k) > P(y_l);\ k \neq l$$ **For balanced data, it will not work**

Very much greater ⟹ **decision will be right**

Naive Bayes Classifier

- In most other circumstances, we need to estimate class-conditional probabilities P(x|yq) as well

$$P(\mathbf{x}|y_q) = \frac{\text{Number of times pattern } \mathbf{x} \text{ appears with } y_q \text{ class}}{\text{Number of times } y_q \text{ appears in the data}}$$

- According to the assumption (attribute values are conditionally independent, given the class), given the class of the pattern, the probability of observing the conjunction x1, x2, …, xn is just the product of the probabilities for the individual attributes:

$$P(x_1, x_2, \ldots, x_n|y_q) = \prod_i P(x_j|y_q)$$

Naive Bayes Classifier

$$y_{\mathrm{NB}} = arg \max_{q} P(y_q) \prod_{j} P(x_j | y_q)$$

- where y_{NB} denotes the class output by the naive Bayes classifier.
- The number of distinct $P(x_j \mid y_q)$ terms that must be estimated from the training data is just the number of distinct attributes (n) times the number of distinct classes (M).

Summary

The Bayes Naive classifier selects the most likely classification V_{nb} given the attribute values $a_1, a_2, \ldots a_n$. This results in:

$$V_{nb} = \text{argmax}_{v_j \in V} P(v_j) \prod P(a_i|v_j) \tag{1}$$

We generally estimate $P(a_i|v_j)$ using m-estimates:

$$P(a_i|v_j) = \frac{n_c + mp}{n + m} \tag{2}$$

where:

$n =$ the number of training examples for which $v = v_j$
$n_c =$ number of examples for which $v = v_j$ and $a = a_i$
$p =$ a priori estimate for $P(a_i|v_j)$
$m =$ the equivalent sample size

Example:

y for x : {M, 1.95 m} ?

- y1 corresponds to the class ‘short’,
- y2 corresponds to the class ‘medium’, and
- y3 corresponds to the class ‘tall’.

Table 3.1 Dataset for Example 3.1

	Gender x_1	Height x_2	Class	y
$s^{(1)}$	F	1.6 m	Short	y_1
$s^{(2)}$	M	2 m	Tall	y_3
$s^{(3)}$	F	1.9 m	Medium	y_2
$s^{(4)}$	F	1.88 m	Medium	y_2
$s^{(5)}$	F	1.7 m	Short	y_1
$s^{(6)}$	M	1.85 m	Medium	y_2
$s^{(7)}$	F	1.6 m	Short	y_1
$s^{(8)}$	M	1.7 m	Short	y_1
$s^{(9)}$	M	2.2 m	Tall	y_3
$s^{(10)}$	M	2.1 m	Tall	y_3
$s^{(11)}$	F	1.8 m	Medium	y_2
$s^{(12)}$	M	1.95 m	Medium	y_2
$s^{(13)}$	F	1.9 m	Medium	y_2
$s^{(14)}$	F	1.8 m	Medium	y_2
$s^{(15)}$	F	1.75 m	Medium	y_2

Example:

y for x : {M, 1.95 m} ?

N_1= no. of y_1=4; N_2= no. of y_2=8; N_3= no. of y_3=3;

$$M = 3, N = 15.$$

$$P(y_1) = \frac{N_1}{N} = \frac{4}{15} = 0.267; \quad P(y_2) = \frac{N_2}{N} = \frac{8}{15} = 0.533$$

$$P(y_3) = \frac{N_3}{N} = \frac{3}{15} = 0.2$$

$$V_{x_1} : \{M, F\} = \{v_{1x_1}, v_{2x_1}\}; d_1 = 2$$

$$V_{x_2} = \{v_{1x_2}, v_{2x_2}, v_{3x_2}, v_{4x_2}, v_{5x_2}, v_{6x_2}\}; d_2 = 6$$

$$= \text{bins } \{(0, 1.6], (1.6, 1.7], (1.7, 1.8], (1.8, 1.9], (1.9, 2.0], (2.0, \infty)\}$$

Table 3.1 Dataset for Example 3.1

	Gender x_1	Height x_2	Class	y
$s^{(1)}$	F	1.6 m	Short	y_1
$s^{(2)}$	M	2 m	Tall	y_3
$s^{(3)}$	F	1.9 m	Medium	y_2
$s^{(4)}$	F	1.88 m	Medium	y_2
$s^{(5)}$	F	1.7 m	Short	y_1
$s^{(6)}$	M	1.85 m	Medium	y_2
$s^{(7)}$	F	1.6 m	Short	y_1
$s^{(8)}$	M	1.7 m	Short	y_1
$s^{(9)}$	M	2.2 m	Tall	y_3
$s^{(10)}$	M	2.1 m	Tall	y_3
$s^{(11)}$	F	1.8 m	Medium	y_2
$s^{(12)}$	M	1.95 m	Medium	y_2
$s^{(13)}$	F	1.9 m	Medium	y_2
$s^{(14)}$	F	1.8 m	Medium	y_2
$s^{(15)}$	F	1.75 m	Medium	y_2

Example:

y for x : {M, 1.95 m} ?

N_1= no. of y_1=4; N_2= no. of y_2=8; N_3= no. of y_3=3;

$$M = 3, N = 15.$$

$$P(y_1) = \frac{N_1}{N} = \frac{4}{15} = 0.267; \quad P(y_2) = \frac{N_2}{N} = \frac{8}{15} = 0.533$$

$$P(y_3) = \frac{N_3}{N} = \frac{3}{15} = 0.2$$

$$V_{x_1} : \{M, F\} = \{v_{1x_1}, v_{2x_1}\}; d_1 = 2$$

$$V_{x_2} = \{v_{1x_2}, v_{2x_2}, v_{3x_2}, v_{4x_2}, v_{5x_2}, v_{6x_2}\}; d_2 = 6$$

$$= \text{bins } \{(0, 1.6], (1.6, 1.7], (1.7, 1.8], (1.8, 1.9], (1.9, 2.0], (2.0, \infty)\}$$

Sorted w.r.t x2:

Gender x1	Height x2 (m)	Class	y
F	1.6	Short	y1
F	1.6	Short	y1
F	1.7	Short	y1
M	1.7	Short	y1
F	1.75	Medium	y2
F	1.8	Medium	y2
F	1.8	Medium	y2
M	1.85	Medium	y2
F	1.88	Medium	y2
F	1.9	Medium	y2
F	1.9	Medium	y2
M	1.95	Medium	y2
M	2	Tall	y3
M	2.1	Tall	y3
M	2.2	Tall	y3

Example:

y for x : {M, 1.95 m} ?

N_1= no. of y_1=4; N_2= no. of y_2=8; N_3= no. of y_3=3;

$$M = 3, N = 15.$$

$$P(y_1) = \frac{N_1}{N} = \frac{4}{15} = 0.267; \quad P(y_2) = \frac{N_2}{N} = \frac{8}{15} = 0.533$$

$$P(y_3) = \frac{N_3}{N} = \frac{3}{15} = 0.2$$

$$V_{x_1} : \{M, F\} = \{v_{1x_1}, v_{2x_1}\}; d_1 = 2$$

$$V_{x_2} = \{v_{1x_2}, v_{2x_2}, v_{3x_2}, v_{4x_2}, v_{5x_2}, v_{6x_2}\}; d_2 = 6$$

$$= \text{bins } \{(0, 1.6], (1.6, 1.7], (1.7, 1.8], (1.8, 1.9], (1.9, 2.0], (2.0, \infty)\}$$

The count table generated from data is given in Table 3.2.

Table 3.2 Number of training samples, $N_{qv_{Ix_j}}$, of class q having value v_{Ix_j}

Value v_{Ix_j}	Count $N_{qv_{Ix_j}}$		
	Short $q = 1$	**Medium** $q = 2$	**Tall** $q = 3$
v_{1x_1}: M	1	2	3
v_{2x_1}: F	3	6	0
v_{1x_2}: (0, 1.6] bin	2	0	0
v_{2x_2}: (1.6, 1.7] bin	2	0	0
v_{3x_2}: (1.7, 1.8] bin	0	3	0
v_{4x_2}: (1.8, 1.9] bin	0	4	0
v_{5x_2}: (1.9, 2.0] bin	0	1	1
v_{6x_2}: (2.0, ∞] bin	0	0	2

Example:

y for x : {M, 1.95 m} ?

N_1= no. of y_1=4; N_2= no. of y_2=8; N_3= no. of y_3=3;

In the discretized domain, '*M*' corresponds to v_{1x_1} and '1.95 m' corresponds to v_{5x_2}.

$$P(x_1|y_1) = N_{1v_{1x_1}}/N_1 = 1/4$$

$$P(x_1|y_2) = N_{2v_{1x_1}}/N_2 = 2/8$$

$$P(x_1|y_3) = N_{3v_{1x_1}}/N_3 = 3/3$$

$$P(x_2|y_1) = N_{1v_{5x_2}}/N_1 = 0/4$$

$$P(x_2|y_2) = N_{2v_{5x_2}}/N_2 = 1/8$$

$$P(x_2|y_3) = N_{3v_{5x_2}}/N_3 = 1/3$$

$$P(\mathbf{x}|y_1) = P(x_1|y_1) \times P(x_2|y_1) = \frac{1}{4} \times 0 = 0$$

The count table generated from data is given in Table 3.2.

Table 3.2 Number of training samples, $N_{qv_{Ix_j}}$, of class q having value v_{Ix_j}

Value v_{Ix_j}	Count $N_{qv_{Ix_j}}$		
	Short $q=1$	**Medium** $q=2$	**Tall** $q=3$
v_{1x_1}: M	1	2	3
v_{2x_1}: F	3	6	0
v_{1x_2}: (0, 1.6] bin	2	0	0
v_{2x_2}: (1.6, 1.7] bin	2	0	0
v_{3x_2}: (1.7, 1.8] bin	0	3	0
v_{4x_2}: (1.8, 1.9] bin	0	4	0
v_{5x_2}: (1.9, 2.0] bin	0	1	1
v_{6x_2}: (2.0, ∞] bin	0	0	2

Example:

y for x : {M, 1.95 m} ?

N_1= no. of y_1=4; N_2= no. of y_2=8; N_3= no. of y_3=3;

In the discretized domain, '*M*' corresponds to v_{1x_1} and '1.95 m' corresponds to v_{5x_2}.

$$P(x_1|y_1) = N_{1v_{1x_1}}/N_1 = 1/4$$

$$P(x_1|y_2) = N_{2v_{1x_1}}/N_2 = 2/8$$

$$P(x_1|y_3) = N_{3v_{1x_1}}/N_3 = 3/3$$

$$P(x_2|y_1) = N_{1v_{5x_2}}/N_1 = 0/4$$

$$P(x_2|y_2) = N_{2v_{5x_2}}/N_2 = 1/8$$

$$P(x_2|y_3) = N_{3v_{5x_2}}/N_3 = 1/3$$

$$P(\mathbf{x}|y_1) = P(x_1|y_1) \times P(x_2|y_1) = \frac{1}{4} \times 0 = 0$$

$$P(\mathbf{x}|y_2) = P(x_1|y_2) \times P(x_2|y_2) = \frac{2}{8} \times \frac{1}{8} = \frac{1}{32}$$

$$P(\mathbf{x}|y_3) = P(x_1|y_3) \times P(x_2|y_3) = \frac{3}{3} \times \frac{1}{3} = \frac{1}{3}$$

$$P(\mathbf{x}|y_1)\,P(y_1) = 0 \times 0.267 = 0$$

$$P(\mathbf{x}|y_2)\,P(y_2) = \frac{1}{32} \times 0.533 = 0.0166$$

$$P(\mathbf{x}|y_3)\,P(y_3) = \frac{1}{3} \times 0.2 = 0.066$$

$$y_{\mathrm{NB}} = \arg\max_q P(\mathbf{x}|y_q) \times P(y_q)$$

- This gives q = 3.
- Therefore, for the pattern x = {M 1.95m}, the predicted class is 'tall'.

- The true class in the data table is 'medium'.
- Use of naive Bayes algorithm on real-life datasets will bring out the power of naive Bayes classifier when N is large.

Example 2:

Let us say, we want to classify a Red Domestic SUV, as stolen or not

Attributes are Color , Type , Origin, and the subject, stolen can be either yes or no.

data set

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

Example 2:

Let us say, we want to classify a Red Domestic SUV, as stolen or not

Attributes are Color , Type , Origin, and the subject, stolen can be either yes or no.

data set

```
Yes:
    Red:
        n = 5
        n_c= 3
        p = .5
        m = 3
    SUV:
        n = 5
        n_c = 1
        p = .5
        m = 3
    Domestic:
        n = 5
        n_c = 2
        p = .5
        m = 3
```

```
No:
    Red:
        n = 5
        n_c = 2
        p = .5
        m = 3
    SUV:
        n = 5
        n_c = 3
        p = .5
        m = 3
    Domestic:
        n = 5
        n_c = 3
        p = .5
        m =3
```

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

Example 2:

- We need to calculate the probabilities P(Red|Yes), P(SUV|Yes), P(Domestic|Yes) , P(Red|No) , P(SUV|No), and P(Domestic|No)
- and multiply them by P(Yes) and P(No) respectively.
- Then we can estimate these values using equation for Y_{NB}
- Looking at P(Red|Y es), we have 5 cases where vj = Yes , and in 3 of those cases ai = Red.
- So for P(RedjY es), n = 5 and nc = 3.
- Note that all attribute are binary (two possible values).
- We are assuming no other information so, p = 1 / (number-of-attribute-values) = 0.5 for all of our attributes.
- Our m value is arbitrary, (We will use m = 3) but consistent for all attributes.
- Now we simply apply eqauation (3) using the precomputed values of n , nc, p, and m.

Example 2:

$$P(Red|Yes) = \frac{3+3*.5}{5+3} = .56 \qquad P(Red|No) = \frac{2+3*.5}{5+3} = .43$$

$$P(SUV|Yes) = \frac{1+3*.5}{5+3} = .31 \qquad P(SUV|No) = \frac{3+3*.5}{5+3} = .56$$

$$P(Domestic|Yes) = \frac{2+3*.5}{5+3} = .43 \qquad P(Domestic|No) = \frac{3+3*.5}{5+3} = .56$$

We have $P(Yes) = .5$ and $P(No) = .5$, so we can apply equation (2). For $v = Yes$, we have

```
P(Yes) * P(Red | Yes) * P(SUV | Yes) * P(Domestic|Yes)

    =  .5 * .56 * .31 * .43   = .037
```

and for $v = No$, we have

```
P(No) * P(Red | No) * P(SUV | No) * P (Domestic | No)

    = .5 * .43 * .56 * .56 = .069
```

Since $0.069 > 0.037$, our example gets classified as 'NO'