Applied Machine Learning

Dr. Harikrishnan N B
Computer Science and Information Systems

BITS Pilani
Pilani Campus

innovate achieve lead

SE ZG568 / SS ZG568,
Applied Machine Learning
Lecture No. 9 [16- March-2025]

Topic of Discussion

Naive Bayes

Spam or Not Spam

ID	Message	Label
1	Buy cheap concert tickets now	Spam
2	Got cheap movie tickets today	Not Spam
3	Win win win a lottery ticket today	Spam
4	Our team will win concert tickets today	Not Spam

Test data

Now, use Naive Bayes to classify a new message, eg: win ticket today

The **Naïve Bayes classifier** is a probabilistic algorithm based on **Bayes' Theorem**, assuming that features (words in text classification) are **conditionally independent given the class**.

Multinomial Naive Bayes

Bag of Words

ID	Message	Label
1	Buy cheap concert tickets now	Spam
2	Got cheap movie tickets today	Not Spam
3	Win win win a lottery ticket today	Spam
4	Our team will win concert tickets today	Not Spam

The **Bag of Words model** represents text as a vector of word counts.

Vocabulary (Unique Words)

First, extract unique words from all messages:

buy, cheap, concert, tickets, now, got, movie, win, will, our, a, lottery, ticket, today, team

Bag of Words

ID	Message	Label
1	Buy cheap concert tickets now	Spam
2	Got cheap movie tickets today	Not Spam
3	Win win win a lottery ticket today	Spam
4	Our team will win concert tickets today	Not Spam

ID	buy	cheap	concert	tickets	now	got	movie	win	will	our	a	lottery	ticket	today	team	Label
1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	Spam
2	0	1	0	1	0	1	1	0	0	0	0	0	0	1	0	Not Spam
3	0	0	0	0	0	0	0	3	0	0	1	1	1	1	0	Spam
4	0	0	1	1	0	0	0	1	1	1	0	0	0	1	1	Not Spam

Test data

Now, use Naive Bayes to classify a new message, eg: win ticket today

The **Naïve Bayes classifier** is a probabilistic algorithm based on **Bayes' Theorem**, assuming that features (words in text classification) are **conditionally independent given the class**.

Bayes Theorem

Bayes' theorem states:

$$P(C|X) = \frac{P(X|C)P(C)}{P(X)}$$

where:

- $P(C|X)$ = Probability of class C given features X (posterior probability).
- $P(X|C)$ = Probability of features X given class C (likelihood).
- $P(C)$ = Prior probability of class C.
- $P(X)$ = Probability of features X (evidence, a normalizing constant).

Since we only care about comparing probabilities, we can ignore $P(X)$ because it's the same for all classes.

$$P(C|X) \propto P(X|C)P(C)$$

Bayes Theorem Example

- There are 30 fruits in total:
  - 18 Apples
  - 12 Oranges
- **Apples are smooth** 80% of the time. = 0.8
- **Oranges are smooth** 30% of the time. = 0.3

$P(Apple \mid Smooth) = \frac{P(Smooth \mid Apple) \times P(Apple)}{P(Smooth)}$

$P(Apple) = \frac{18}{30}$, $P(Smooth \mid Apple) = 0.8$

Now, you **pick a fruit and feel that it is smooth**. What is the probability that the fruit is an **apple**?

$P(Smooth) = P(A \cap Smooth) + P(O \cap Smooth)$

$P(Smooth) = P(S \mid A) \times P(A) + P(S \mid O) P(O)$

- $P(Apple) = \frac{18}{30} = 0.6$ (Prior probability of picking an apple)
- $P(Orange) = \frac{12}{30} = 0.4$ (Prior probability of picking an orange)
- $P(Smooth|Apple) = 0.8$ (Likelihood of smooth texture given apple)
- $P(Smooth|Orange) = 0.3$ (Likelihood of smooth texture given orange)

$$P(Apple|Smooth) = \frac{P(Smooth|Apple)P(Apple)}{P(Smooth)}$$

$$P(Smooth) = P(Smooth|Apple)P(Apple) + P(Smooth|Orange)P(Orange)$$

Independence Concept

Imagine you roll **two fair dice**, one red and one blue. The outcome of the red die **does not affect** the outcome of the blue die.

- **A = "Red die shows a 3"**
- **B = "Blue die shows a 5"**

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

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

Since the dice rolls do not influence each other, we say that **A and B are independent**.

A & B are independent

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

$$P(A \cap B) = P(A) \cdot P(B)$$

Theory of Independence

Two events **A** and **B** are **independent** if knowing that **one happened does NOT change the probability of the other happening**. Mathematically, this means:

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

which leads to:

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

This means that the probability of **both events happening together** is simply the **product of their individual probabilities**.

Conditional Independence

- A = "Student solves a hard math problem correctly"
- B = "Student solves an easy math problem correctly"
- C = "Student is generally good at math"

$P(A|B) = P(A)$ $\quad P(A|B,C) = P(A|C)$

Before knowing C, solving the easy problem (B) might give us a hint about solving the hard problem (A).

But **once we already know the student is good at math (C), knowing they solved an easy problem (B) does not change our belief about them solving a hard problem.**

So,

$$P(A|B,C) = P(A|C)$$

$P(A|B,C) = P(A|C)$ $\quad P(B|A,C) = P(B|C)$

This means: **Given that the student is good at math, solving the easy problem does not give extra information about solving the hard one.**

Summary

- $P(A|B,C) = P(A|C)$ means that **A and B are conditionally independent given C.**
- **Before knowing C**, A and B may be dependent.
- **After knowing C**, B does not affect A anymore.

Naive Bayes

The **Naïve Bayes classifier** is a probabilistic algorithm based on **Bayes' Theorem**, assuming that features (words in text classification) are **conditionally independent given the class**.

Bayes' theorem states:

$$P(C|X) = \frac{P(X|C)P(C)}{P(X)}$$

where:

- $P(C|X)$ = Probability of class C given features X (posterior probability).
- $P(X|C)$ = Probability of features X given class C (likelihood).
- $P(C)$ = Prior probability of class C.
- $P(X)$ = Probability of features X (evidence, a normalizing constant).

$X = (X_1, X_2, X_3)$

$P(X|C)$

$P(X_1, X_2, X_3 | C)$

$= P(X_1|C) \times P(X_2|C) \cdot P(X_3|C)$

$C = \nearrow$ Spam / Not Spam

$P(C|X) = \frac{P(X|C) \cdot P(C)}{P(X)}$ (X: win ticket today)

In Naïve Bayes, we assume that each feature (word) is **independent** given the class.

$$P(X|C) = P(x_1, x_2, ..., x_n|C) = P(x_1|C)P(x_2|C)...P(x_n|C)$$

For text classification, features x_i represent words in the message, and their probabilities are calculated using word frequencies.

$P(\text{Not Spam}|\text{win ticket today}) = \frac{P(\text{win ticket today}|\text{Not Spam}) \times P(\text{Not Spam})}{P(\text{win ticket today})}$

$$P(C|X) \propto P(C)\prod_{i=1}^{n} P(x_i|C)$$

$P(\text{Spam}|\text{win ticket today}) = \frac{P(\text{win ticket today}|\text{Spam}) \cdot P(\text{Spam})}{P(\text{win ticket today})}$

$X = X_1 X_2 X_3 X_4 X_5$

Conditional Independence: Given the class label the features X_i, X_j are conditionally independent

$$P(X_1 | X_2) =$$

$$P(X_1 | X_2, C) = P(X_1 | C)$$

$$P(A | B, C) = P(A | C)$$

$$P(X_1 | X_2, X_3, X_4, C) = P(X_1 | C)$$

$P(X_2 | X_1 X_3 X_4 C) = P(X_2 | C)$

$P(X_1 X_2 \ldots X_5)$

$P(X_1, A | C) =$

Spam
Not spam

$P(C | X_1 X_2 X_3 X_4 X_5) = \dfrac{P(X_1, X_2, X_3, X_4, X_5, A | C) P(C)}{P(X_1 X_2 \ldots X_5)}$

$P(A|C) = \dfrac{P(A \cap C)}{P(C)}$

$\propto P(X_1, A | C) \times P(C)$

B

$\propto \dfrac{P(X_1 \cap A \cap C)}{P(C)} \times P(C)$

$P(B|C) = \dfrac{P(B \cap C)}{P(C)}$

$B = A \cap X_1$

$\dfrac{P(A \cap X_1 \cap C)}{P(C)}$

$P(X_1 \cap B) = P(X_1 | B) \times P(B)$

$= P(X_1 | X_2 X_3 X_4 X_5 C) \times P(X_2 X_3 X_4 X_5 C)$

$= P(X_1 | X_2 X_3 X_4 X_5 C) \times P(X_2 | X_3 X_4 X_5 C) \times P(X_3 X_4 X_5 C)$

$P(X_3 | X_4 X_5 C) \times P(X_4 X_5 C)$

$= P(x_1 | x_2 x_3 x_4 x_5 C) \times P(x_2 | x_3 x_4 x_5 C) \times P(x_3 | x_4 x_5 C) \times P(x_4 | x_5 C) \times P(x_5 | C) \times P(C)$

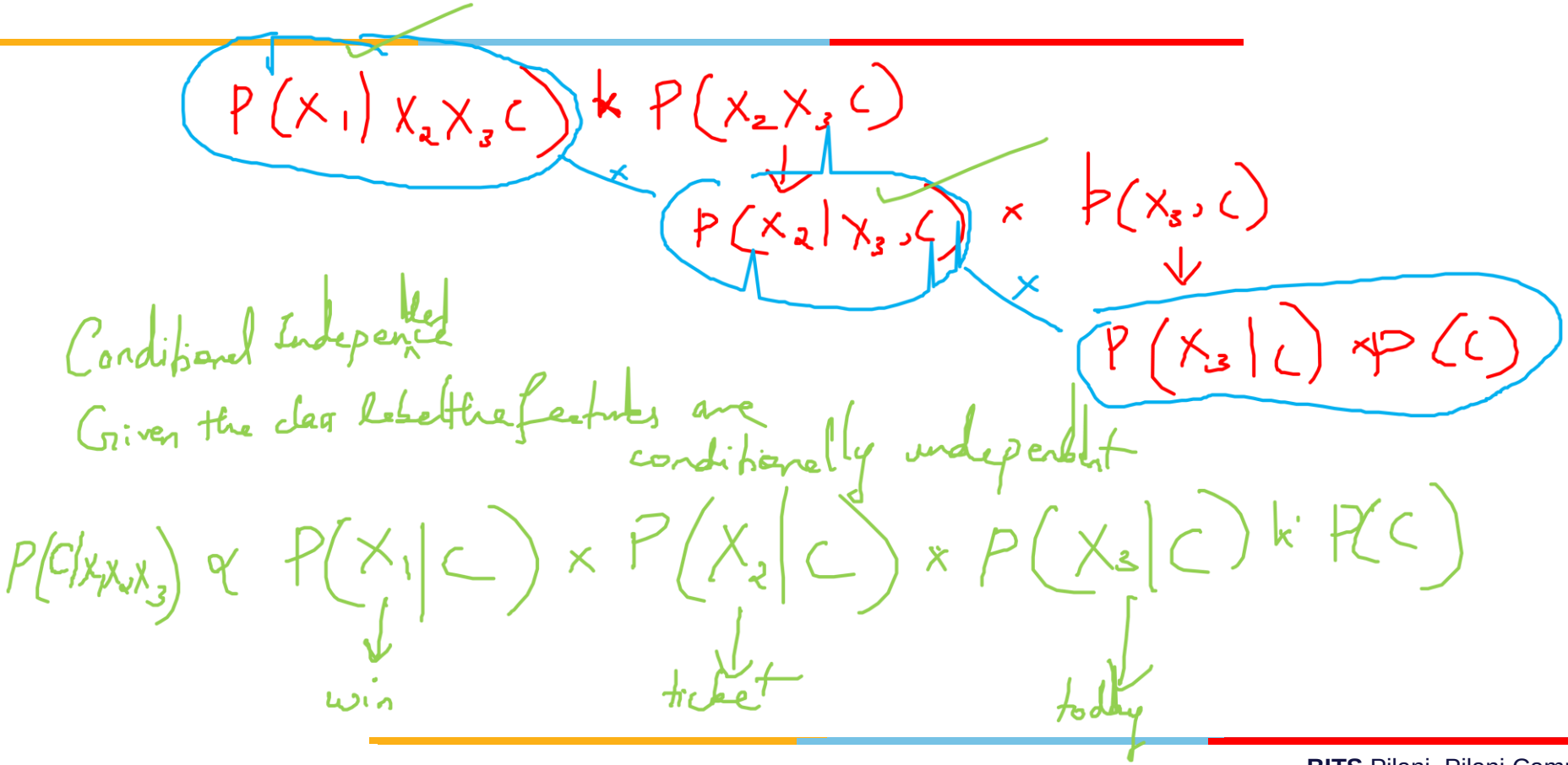
$P(C | x_1 x_2 x_3) = \dfrac{P(x_1 x_2 x_3 | C) P(C)}{P(x_1 x_2 x_3)}$

$P(x_1 \cap x_2 \cap x_3 | C) P(C)$

$P(A|C) P(C) = P(A \cap C)$

$P(x_1 \cap x_2 \cap x_3 \cap C)$

$P(x_1 \cap B) = P(x_1 | B) P(B)$


P(X1 | X2 X3 C)
P(X2 X3 C)
P(X2 | X3 , C)
P(X3 , C)
P(X3 | C) * P(C)
Conditional Independence
Given the class label the features are conditionally independent
P(C | X1 X2 X3) ∝ P(X1 | C) * P(X2 | C) * P(X3 | C) * P(C)
win
ticket
today

$$\hat{y} = \arg\max_{C} P(C) \prod_{i=1}^{n} P(x_i|C)$$

where:

- $\hat{y}$ is the predicted class.
- C represents the possible classes (e.g., Spam, Not Spam).
- $P(C)$ is the prior probability of class C.
- $P(x_i|C)$ is the probability of word x_i given class C.

Log formulation

Applying Logarithm

Since multiplying many small probabilities can lead to **numerical underflow**, we take the **logarithm**:

$$\log P(C|x_1, x_2, \ldots, x_n) \propto \log P(C) + \sum_{i=1}^{n} \log P(x_i|C)$$

Final Decision Rule

To classify a new observation, we compute this log probability for each class and choose the one with the highest value:

$$\hat{C} = \arg\max_C \left(\log P(C) + \sum_{i=1}^{n} \log P(x_i|C) \right)$$

where:

- $P(C)$ is the **prior probability** of class C,
- $P(x_i|C)$ is the **likelihood** of feature x_i given class C.

Test data

Now, use Naive Bayes to classify a new message, eg: win ticket today

The **Naïve Bayes classifier** is a probabilistic algorithm based on **Bayes' Theorem**, assuming that features (words in text classification) are **conditionally independent given the class**.

Train Dataset

ID	Message	Label
1	Buy cheap concert tickets now	Spam
2	Got cheap movie tickets today	Not Spam
3	Win win win a lottery ticket today	Spam
4	Our team will win concert tickets today	Not Spam

$P(Spam) = \frac{2}{4} = 0.5$

ID	buy	cheap	concert	tickets	now	got	movie	win	will	our	a	lottery	ticket	today	team	Label
1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	Spam
2	0	1	0	1	0	1	1	0	0	0	0	0	0	1	0	Not Spam
3	0	0	0	0	0	0	0	3	0	0	1	1	1	1	0	Spam
4	0	0	1	1	0	0	0	1	1	1	0	0	0	1	1	Not Spam

Compute Prior Probability

The prior probability of each class is calculated as:

$$P(\text{Spam}) = \frac{\text{Number of Spam messages}}{\text{Total messages}} = \frac{2}{4} = 0.5$$

$$P(\text{Not Spam}) = \frac{\text{Number of Not Spam messages}}{\text{Total messages}} = \frac{2}{4} = 0.5$$

Now, use Naive Bayes to classify a new message, eg: **win ticket today**

$$P(\text{Spam}|\text{"win ticket today"}) \propto P(\text{Spam}) \times P(\text{win}|\text{Spam}) \times P(\text{ticket}|\text{Spam}) \times P(\text{today}|\text{Spam})$$

$$P(\text{Not Spam}|\text{"win ticket today"}) \propto P(\text{Not Spam}) \times P(\text{win}|\text{Not Spam}) \times P(\text{ticket}|\text{Not Spam}) \times P(\text{today}|\text{Not Spam})$$

Compute P(word|class)?

Let's say class = spam, word = win

P(win|spam) ?

Let the total number of words in the class spam be *n*. [What is *n*?]

Let n_k be the total number of occurrences of word "win" in the class spam. [What is n_k?]

ID	buy	cheap	concert	tickets	now	got	movie	win	will	our	a	lottery	ticket	today	team	Label
1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	Spam
2	0	1	0	1	0	1	1	0	0	0	0	0	0	1	0	Not Spam
3	0	0	0	0	0	0	0	3	0	0	1	1	1	1	0	Spam
4	0	0	1	1	0	0	0	1	1	1	0	0	0	1	1	Not Spam

Compute P(word|class)?

$P(win|spam) = n_k/n$

Applying **Laplace Smoothing**. Why?

$P(win|spam) = (n_k+1) / (n+|Vocabulary|)$

What is |Vocabulary|?

ID	buy	cheap	concert	tickets	now	got	movie	win	will	our	a	lottery	ticket	today	team	Label
1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	Spam
2	0	1	0	1	0	1	1	0	0	0	0	0	0	1	0	Not Spam
3	0	0	0	0	0	0	0	3	0	0	1	1	1	1	0	Spam
4	0	0	1	1	0	0	0	1	1	1	0	0	0	1	1	Not Spam

Compute the likelihoods

$$P(\text{Spam}|\text{"win ticket today"}) \propto P(\text{Spam}) \times P(\text{win}|\text{Spam}) \times P(\text{ticket}|\text{Spam}) \times P(\text{today}|\text{Spam})$$

ID	buy	cheap	concert	tickets	now	got	movie	win	will	our	a	lottery	ticket	today	team	Label
1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	Spam
2	0	1	0	1	0	1	1	0	0	0	0	0	0	1	0	Not Spam
3	0	0	0	0	0	0	0	3	0	0	1	1	1	1	0	Spam
4	0	0	1	1	0	0	0	1	1	1	0	0	0	1	1	Not Spam

***P(word|spam)* = $(n_{word}+1)$ / $(n+|\text{Vocabulary}|)$**

P(spam)	P(win|spam)	P(ticket|spam)	P(today|Spam)
1/2	4/27	2/27	2/27

= 0.29

P(Spam| win ticket today)

Using Laplace Smoothing:

$$P(w|\text{Spam}) = \frac{\text{count of } w \text{ in Spam} + 1}{\text{total words in Spam} + V}$$

- Total words in Spam messages = 12
- Vocabulary size $V = 15$
- New denominator: $12 + 15 = 27$

Updated Probabilities:

1. $P(\text{win}|\text{Spam}) = \frac{3+1}{27} = \frac{4}{27} = 0.1481$
2. $P(\text{ticket}|\text{Spam}) = \frac{1+1}{27} = \frac{2}{27} = 0.0741$
3. $P(\text{today}|\text{Spam}) = \frac{1+1}{27} = \frac{2}{27} = 0.0741$

Compute the likelihoods

$$P(\text{Not Spam}|\text{"win ticket today"}) \propto P(\text{Not Spam}) \times P(\text{win}|\text{Not Spam}) \times P(\text{ticket}|\text{Not Spam}) \times P(\text{today}|\text{Not Spam})$$

ID	buy	cheap	concert	tickets	now	got	movie	win	will	our	a	lottery	ticket	today	team	Label
1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	Spam
2	0	1	0	1	0	1	1	0	0	0	0	0	0	1	0	Not Spam
3	0	0	0	0	0	0	0	3	0	0	1	1	1	1	0	Spam
4	0	0	1	1	0	0	0	1	1	1	0	0	0	1	1	Not Spam

P(word|Not Spam) =

(n_{word}+1) / (n+|Vocabulary|)

P(Not Spam)	P(win|Not Spam)	P(ticket|Not Spam)	P(today|Not Spam)
1/2	2/27	1/27	2/27

P(Not Spam| win ticket today)

Since $V = 15$ and total words in Not Spam = **12**, our new denominator is:

$$12 + 15 = 27$$

Now, we compute for each word:

1. $P(\text{win}|\text{Not Spam})$
   - "win" appears **1 time** in Not Spam.

$$P(\text{win}|\text{Not Spam}) = \frac{1+1}{27} = \frac{2}{27} = 0.0741$$

2. $P(\text{ticket}|\text{Not Spam})$
   - "ticket" appears **1 time** in Not Spam.

$$P(\text{ticket}|\text{Not Spam}) = \frac{1+1}{27} = \frac{2}{27} = 0.0741$$

3. $P(\text{today}|\text{Not Spam})$
   - "today" appears **2 times** in Not Spam.

$$P(\text{today}|\text{Not Spam}) = \frac{2+1}{27} = \frac{3}{27} = 0.1111$$

Calculations

$$P(\text{"win ticket today"}|\text{Spam})P(\text{Spam}) = P(\text{win}|\text{Spam}) \times P(\text{ticket}|\text{Spam}) \times P(\text{today}|\text{Spam}) \times P(\text{Spam})$$

Using Laplace-smoothed probabilities:

4×10^{-4} mail

C = Spam

$$P(\text{"win ticket today"}|\text{Not Spam})P(\text{Not Spam}) = P(\text{win}|\text{Not Spam}) \times P(\text{ticket}|\text{Not Spam}) \times P(\text{today}|\text{Not Spam}) \times P(\text{Not Spam})$$

$= 5.6 \times 10^{-6}$

Calculations

$$P(\text{"win ticket today"}|\text{Spam})P(\text{Spam}) = P(\text{win}|\text{Spam}) \times P(\text{ticket}|\text{Spam}) \times P(\text{today}|\text{Spam}) \times P(\text{Spam})$$

Using Laplace-smoothed probabilities:

$$= 0.1481 \times 0.0741 \times 0.0741 \times 0.5$$

$$= 0.000407 \rightarrow a$$

$$P(\text{"win ticket today"}|\text{Not Spam})P(\text{Not Spam}) = P(\text{win}|\text{Not Spam}) \times P(\text{ticket}|\text{Not Spam}) \times P(\text{today}|\text{Not Spam}) \times P(\text{Not Spam})$$

$$= 0.0741 \times 0.0741 \times 0.1111 \times 0.5$$

$$= 0.000305 \rightarrow b$$

After Normalization

We normalize:

$$P(\text{Spam}|\text{"win ticket today"}) = \frac{0.000407}{0.000407 + 0.000305} = \frac{0.000407}{0.000712} = 0.5718$$

$$P(\text{Not Spam}|\text{"win ticket today"}) = \frac{0.000305}{0.000712} = 0.4282$$

After Normalization

We normalize:

$$P(\text{Spam}|\text{"win ticket today"}) = \frac{0.000407}{0.000407 + 0.000305} = \frac{0.000407}{0.000712} = 0.5718$$

$$P(\text{Not Spam}|\text{"win ticket today"}) = \frac{0.000305}{0.000712} = 0.4282$$

Example 2

Play Tennis or Not

ID	Outlook	Temperature	Humidity	Wind	Play Tennis
1	Sunny	Hot	High	Weak	No
2	Sunny	Hot	High	Strong	No
3	Overcast	Hot	High	Weak	Yes
4	Rain	Mild	High	Weak	Yes
5	Rain	Cool	Normal	Weak	Yes
6	Rain	Cool	Normal	Strong	No
7	Overcast	Cool	Normal	Strong	Yes
8	Sunny	Mild	High	Weak	No
9	Sunny	Cool	Normal	Weak	Yes
10	Rain	Mild	Normal	Weak	Yes
11	Sunny	Mild	Normal	Strong	Yes
12	Overcast	Mild	High	Strong	Yes
13	Overcast	Hot	Normal	Weak	Yes
14	Rain	Mild	High	Strong	No

Predict for

(Outlook = Sunny, Temperature = Cool, Humidity = High, Wind = Strong)

Play Tennis or Not

ID	Outlook	Temperature	Humidity	Wind	Play Tennis
1	Sunny	Hot	High	Weak	No
2	Sunny	Hot	High	Strong	No
3	Overcast	Hot	High	Weak	Yes
4	Rain	Mild	High	Weak	Yes
5	Rain	Cool	Normal	Weak	Yes
6	Rain	Cool	Normal	Strong	No
7	Overcast	Cool	Normal	Strong	Yes
8	Sunny	Mild	High	Weak	No
9	Sunny	Cool	Normal	Weak	Yes
10	Rain	Mild	Normal	Weak	Yes
11	Sunny	Mild	Normal	Strong	Yes
12	Overcast	Mild	High	Strong	Yes
13	Overcast	Hot	Normal	Weak	Yes
14	Rain	Mild	High	Strong	No

$$P(\text{Yes}) = \frac{\text{Total Yes}}{\text{Total Samples}} = \frac{9}{14} = 0.6429$$

$$P(\text{No}) = \frac{\text{Total No}}{\text{Total Samples}} = \frac{5}{14} = 0.3571$$

Play Tennis or Not

ID	Outlook	Temperature	Humidity	Wind	Play Tennis
1	Sunny	Hot	High	Weak	No
2	Sunny	Hot	High	Strong	No
3	Overcast	Hot	High	Weak	Yes
4	Rain	Mild	High	Weak	Yes
5	Rain	Cool	Normal	Weak	Yes
6	Rain	Cool	Normal	Strong	No
7	Overcast	Cool	Normal	Strong	Yes
8	Sunny	Mild	High	Weak	No
9	Sunny	Cool	Normal	Weak	Yes
10	Rain	Mild	Normal	Weak	Yes
11	Sunny	Mild	Normal	Strong	Yes
12	Overcast	Mild	High	Strong	Yes
13	Overcast	Hot	Normal	Weak	Yes
14	Rain	Mild	High	Strong	No

(Outlook = Sunny, Temperature = Cool, Humidity = High, Wind = Strong)

Vocabulary Size (V) for Each Feature

- **Outlook:** {Sunny, Overcast, Rain} → $V = 3$
- **Temperature:** {Hot, Mild, Cool} → $V = 3$
- **Humidity:** {High, Normal} → $V = 2$
- **Wind:** {Weak, Strong} → $V = 2$

For Laplace Smoothing, we add **1** to the numerator and V to the denominator:

$$P(\text{Feature}|\text{Class}) = \frac{\text{Count of Feature in Class} + 1}{\text{Total Count in Class} + V}$$

$$P(\text{No}|\text{Sunny, Cool, High, Strong}) \propto P(\text{Sunny}|\text{No}) \times P(\text{Cool}|\text{No}) \times P(\text{High}|\text{No}) \times P(\text{Strong}|\text{No}) \times P(\text{No})$$

$$P(\text{Yes}|\text{Sunny, Cool, High, Strong}) \propto P(\text{Sunny}|\text{Yes}) \times P(\text{Cool}|\text{Yes}) \times P(\text{High}|\text{Yes}) \times P(\text{Strong}|\text{Yes}) \times P(\text{Yes})$$

$$P(\text{No}|\text{Sunny, Cool, High, Strong}) \propto P(\text{Sunny}|\text{No}) \times P(\text{Cool}|\text{No}) \times P(\text{High}|\text{No}) \times P(\text{Strong}|\text{No}) \times P(\text{No})$$

- $P(\text{Sunny}|\text{No}) = \frac{3+1}{5+3} = \frac{4}{8} = 0.5$
- $P(\text{Cool}|\text{No}) = \frac{1+1}{5+3} = \frac{2}{8} = 0.25$
- $P(\text{High}|\text{No}) = \frac{4+1}{5+2} = \frac{5}{7} = 0.7143$
- $P(\text{Strong}|\text{No}) = \frac{3+1}{5+2} = \frac{4}{7} = 0.5714$
- $P(\text{No}) = \frac{5}{14} = 0.3571$

$$P(\text{No}|\text{Sunny, Cool, High, Strong}) \propto 0.0182$$

ID	Outlook	Temperature	Humidity	Wind	Play Tennis
1	Sunny	Hot	High	Weak	No
2	Sunny	Hot	High	Strong	No
3	Overcast	Hot	High	Weak	Yes
4	Rain	Mild	High	Weak	Yes
5	Rain	Cool	Normal	Weak	Yes
6	Rain	Cool	Normal	Strong	No
7	Overcast	Cool	Normal	Strong	Yes
8	Sunny	Mild	High	Weak	No
9	Sunny	Cool	Normal	Weak	Yes
10	Rain	Mild	Normal	Weak	Yes
11	Sunny	Mild	Normal	Strong	Yes
12	Overcast	Mild	High	Strong	Yes
13	Overcast	Hot	Normal	Weak	Yes
14	Rain	Mild	High	Strong	No

P(Yes| Sunny, cool, High, Strong)

$$P(\text{Yes}|\text{Sunny, Cool, High, Strong}) \propto P(\text{Sunny}|\text{Yes}) \times P(\text{Cool}|\text{Yes}) \times P(\text{High}|\text{Yes}) \times P(\text{Strong}|\text{Yes}) \times P(\text{Yes})$$

- $P(\text{Sunny}|\text{Yes}) =$
- $P(\text{Cool}|\text{Yes}) =$
- $P(\text{High}|\text{Yes}) =$
- $P(\text{Strong}|\text{Yes})$

$$\frac{2+1}{9+3}$$

$$P(w|\text{Class}) = \frac{\text{count}(w, \text{Class}) + 1}{\text{total words in Class} + V}$$

ID	Outlook	Temperature	Humidity	Wind	Play Tennis
1	Sunny	Hot	High	Weak	No
2	Sunny	Hot	High	Strong	No
3	Overcast	Hot	High	Weak	Yes
4	Rain	Mild	High	Weak	Yes
5	Rain	Cool	Normal	Weak	Yes
6	Rain	Cool	Normal	Strong	No
7	Overcast	Cool	Normal	Strong	Yes
8	Sunny	Mild	High	Weak	No
9	Sunny	Cool	Normal	Weak	Yes
10	Rain	Mild	Normal	Weak	Yes
11	Sunny	Mild	Normal	Strong	Yes
12	Overcast	Mild	High	Strong	Yes
13	Overcast	Hot	Normal	Weak	Yes
14	Rain	Mild	High	Strong	No

Gaussian Naive Bayes

We have a small dataset of students with two features:

- **Study Hours**: Number of hours a student studied.
- **Previous Scores**: Their last exam percentage.
- **Target (Pass/Fail)**: Whether they passed or failed this exam.

Study Hours	Previous Scores	Pass (1) / Fail (0)
2	50	0
3	55	0
5	65	1
7	70	1
8	80	1
1	45	0
4	60	1
6	75	1

Now, given a new student with **Study Hours = 4.5** and **Previous Scores = 67**, we will use **Gaussian Naïve Bayes** to predict if they will pass or fail.

Assumption

Naïve Bayes is based on **Bayes' Theorem**:

$$P(Y|X) = \frac{P(X|Y)P(Y)}{P(X)}$$

mean, Variance

where:

- $P(Y|X)$ = Posterior probability (Pass or Fail given study hours & scores)
- $P(X|Y)$ = Likelihood (Probability of observing features given the class)
- $P(Y)$ = Prior probability (Overall probability of each class)
- $P(X)$ = Evidence (Normalization factor, same for both classes)

Since we assume features are **independent and Gaussian-distributed**, the **likelihood** is given by:

$$P(X_i|Y) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(\frac{-(X_i - \mu_i)^2}{2\sigma^2}\right)$$

where μ (mean) and σ^2 (variance) are computed separately for each feature in each class.

We have a small dataset of students with two features:

- **Study Hours**: Number of hours a student studied.
- **Previous Scores**: Their last exam percentage.
- **Target (Pass/Fail)**: Whether they passed or failed this exam.

Study Hours	Previous Scores	Pass (1) / Fail (0)
2	50	0
3	55	0
5	65	1
7	70	1
8	80	1
1	45	0
4	60	1
6	75	1

$$P(\text{Pass}) = \frac{\text{Number of Pass cases}}{\text{Total samples}} = \frac{5}{8}$$

$$P(\text{Fail}) = \frac{\text{Number of Fail cases}}{\text{Total samples}} \quad \frac{3}{8}$$

Now, given a new student with **Study Hours = 4.5** and **Previous Scores = 67**, we will use **Gaussian Naïve Bayes** to predict if they will pass or fail.

Compute Class Priors

We have a small dataset of students with two features:

- **Study Hours**: Number of hours a student studied.
- **Previous Scores**: Their last exam percentage.
- **Target (Pass/Fail)**: Whether they passed or failed this exam.

Study Hours	Previous Scores	Pass (1) / Fail (0)
2	50	0
3	55	0
5	65	1
7	70	1
8	80	1
1	45	0
4	60	1
6	75	1

$$P(\text{Pass}) = \frac{\text{Number of Pass cases}}{\text{Total samples}} = \frac{5}{8} = 0.625$$

$$P(\text{Fail}) = \frac{\text{Number of Fail cases}}{\text{Total samples}} = \frac{3}{8} = 0.375$$

Now, given a new student with **Study Hours = 4.5** and **Previous Scores = 67**, we will use **Gaussian Naïve Bayes** to predict if they will pass or fail.

Compute Mean and Variance of Each Feature

We have a small dataset of students with two features:

- **Study Hours**: Number of hours a student studied.
- **Previous Scores**: Their last exam percentage.
- **Target (Pass/Fail)**: Whether they passed or failed this exam.

Study Hours	Previous Scores	Pass (1) / Fail (0)
2	50	0
3	55	0
5	65	1
7	70	1
8	80	1
1	45	0
4	60	1
6	75	1

Now, given a new student with **Study Hours = 4.5** and **Previous Scores = 67**, we will use **Gaussian Naïve Bayes** to predict if they will pass or fail.

Compute Mean and Variance of Each Feature

We have a small dataset of students with two features:

- **Study Hours**: Number of hours a student studied.
- **Previous Scores**: Their last exam percentage.
- **Target (Pass/Fail)**: Whether they passed or failed this exam.

Study Hours	Previous Scores	Pass (1) / Fail (0)
2	50	0
3	55	0
5	65	1
7	70	1
8	80	1
1	45	0
4	60	1
6	75	1

Now, given a new student with **Study Hours = 4.5** and **Previous Scores = 67**, we will use **Gaussian Naïve Bayes** to predict if they will pass or fail.

For Pass (Y=1)

- Study Hours Mean:
  $\mu_{\text{pass}} = \frac{5+7+8+4+6}{5} = 6$
- Variance:

$$\sigma^2_{\text{pass}} = \frac{(5-6)^2 + (7-6)^2 + (8-6)^2 + (4-6)^2 + (6-6)^2}{5-1}$$
$$= \frac{1+1+4+4+0}{4} = \frac{10}{4} = 2.5$$

- Previous Scores Mean:
  $\mu_{\text{pass}} = \frac{65+70+80+60+75}{5} = 70$
- Variance:

$$\sigma^2_{\text{pass}} = \frac{(65-70)^2 + (70-70)^2 + (80-70)^2 + (60-70)^2 + (75-70)^2}{5-1}$$
$$= \frac{25+0+100+100+25}{4} = \frac{250}{4} = 62.5$$

For Fail (Y=0)

- Study Hours Mean:
  $\mu_{\text{fail}} = \frac{2+3+1}{3} = 2$
- Variance:

$$\sigma^2_{\text{fail}} = \frac{(2-2)^2 + (3-2)^2 + (1-2)^2}{3-1}$$
$$= \frac{0+1+1}{2} = \frac{2}{2} = 1$$

- Previous Scores Mean:
  $\mu_{\text{fail}} = \frac{50+55+45}{3} = 50$
- Variance:

$$\sigma^2_{\text{fail}} = \frac{(50-50)^2 + (55-50)^2 + (45-50)^2}{3-1}$$
$$= \frac{0+25+25}{2} = \frac{50}{2} = 25$$

New student: Study Hours = 4.5, Previous Score = 67

$P(X_1 = 4.5|Pass) = ?$
$P(X_2 = 67| Pass) = ?$
$P(Pass) = ?$

$P(X_1 = 4.5|Fail) = ?$
$P(X_2 = 67| Fail) = ?$
$P(Fail) = ?$

New student: Study Hours = 4.5, Previous Score = 67

For a given feature X_i, the **Gaussian probability density function** is:

$$P(X_i|Y) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(\frac{-(X_i - \mu)^2}{2\sigma^2}\right)$$

$$P(\text{Pass}) = \frac{\text{Number of Pass cases}}{\text{Total samples}} = \frac{5}{8} = 0.625$$

$$P(\text{Fail}) = \frac{\text{Number of Fail cases}}{\text{Total samples}} = \frac{3}{8} = 0.375$$

For Pass ($Y = 1$)

$$P(4.5|\text{Pass}) = \frac{1}{\sqrt{2\pi(2.5)}} \exp\left(\frac{-(4.5 - 6)^2}{2(2.5)}\right)$$

$$P(67|\text{Pass}) = \frac{1}{\sqrt{2\pi(62.5)}} \exp\left(\frac{-(67 - 70)^2}{2(62.5)}\right)$$

For Fail ($Y = 0$)

$$P(4.5|\text{Fail}) = \frac{1}{\sqrt{2\pi(1)}} \exp\left(\frac{-(4.5 - 2)^2}{2(1)}\right)$$

$$P(67|\text{Fail}) = \frac{1}{\sqrt{2\pi(25)}} \exp\left(\frac{-(67 - 50)^2}{2(25)}\right)$$

Decision Rule

$$P(\text{Pass}|X) = P(X_1|\text{Pass})P(X_2|\text{Pass})P(\text{Pass})$$

$$P(\text{Fail}|X) = P(X_1|\text{Fail})P(X_2|\text{Fail})P(\text{Fail})$$

Final Decision Rule:

$$\text{Predict Pass if } P(\text{Pass}|X) > P(\text{Fail}|X), \text{ otherwise predict Fail.}$$

Final Answer

Final Computed Values:

$$P(\text{Pass}|X) = 0.00472, \quad P(\text{Fail}|X) = 1.62 \times 10^{-6}$$