

# N Bayes Algorithm

D1	3	4	5	6	7	8	9
4	5	6	7	8	9	10	
5	6	7	8	9	10	11	
6	7	8	9	10	11	12	

✓ What is the probability that  $D1 = 2$  given that  $D1 + D2 \leq 5$ ?

Table 3 shows that for 3 of these 10 outcomes,  $D1 = 2$ .

Thus, the conditional probability  $P(D1=2 \mid D1+D2 \leq 5) = \frac{3}{10} = 0.3$ .

English

conditional-prob

given

Conditioned-on

$P(D1=2 \mid D1+D2 \leq 5)$

$= \frac{3}{10}$

		D2					
+		1	2	3	4	5	6
D1	1	2	3	4	5	6	7
	2	3	4	5	6	7	8
	3	4	5	6	7	8	9
	4	5	6	7	8	9	10
	5	6	7	8	9	10	11
	6	7	8	9	10	11	12

What is the probability that  $D1 = 2$  given that  $D1 + D2 \leq 5$ ?

Table 3 shows that for 3 of these 10 outcomes,  $D1 = 2$ .

Thus, the conditional probability  $P(D1=2 \mid D1+D2 \leq 5) = \frac{3}{10} = 0.3$ .

Table 3

		D2					
+		1	2	3	4	5	6
D1	1	2	3	4	5	6	7
	2	3	4	5	6	7	8
	3	4	5	6	7	8	9
	4	5	6	7	8	9	10
	5	6	7	8	9	10	11
	6	7	8	9	10	11	12

Here in the earlier notation for the definition of conditional probability, the conditioning event  $B$  is that  $D1 + D2 \leq 5$ , and the event  $A$  is  $D1 = 2$ . We have  $P(A|B) = \frac{P(A \cap B)}{P(B)} = \frac{3/36}{10/36} = \frac{3}{10}$ , as seen in the table.

$D1 = 2 : A$

$D1 + D2 \leq 5 : B$

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

def

if

$\left( \frac{3}{36} \right) / \left( \frac{10}{36} \right) = \frac{3}{10}$

A intersection B means what is probability of event a and b both occurs .

Conditional prob:

$$P(A|B) = P_r(A=a \mid B=b)$$

$\downarrow$  value  
 $\uparrow$  value

$$\begin{cases} A: r.v \\ B: r.v \end{cases}$$

def:  $P(A|B) = \frac{P(A \cap B)}{P(B)}$  ;  $P(B) \neq 0$

Independent Events & Mutually Exclusive Events

A, B are said to be independent

Def:  $\begin{cases} P(A|B) = P(A) \\ P(B|A) = P(B) \end{cases}$

$\begin{cases} P(D_1=6 \mid D_2=3) \\ P(D_2=3 \mid D_1=6) \end{cases} = \begin{cases} P(D_1=6) \\ P(D_2=3) \end{cases}$

④ ③  
 1 2  
 A: getting value of 6 in die 1 throw  
 ((D<sub>1</sub>=6))  
 B: getting a value of 3 in die 2's throw  
 (D<sub>2</sub>=3)

Now see two dice d1 and d2 . See above image p of a given b = p of a

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

There is no relation between both dice so prob of both of them will different means not depend on each other that is called Independent events .

Mutually Exclusive :

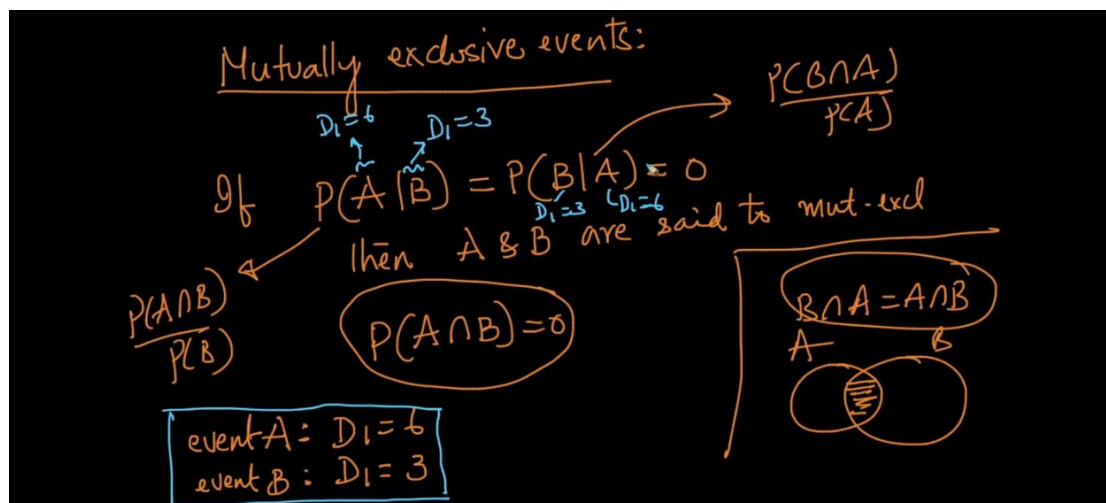
Here see below image A is dice 1 roll p will 6  
 B = same dice roll probability 3 .

Now in this case dice is only one so probability is dependant now

$P(A|B) = 0$  (What is prob of A if B is given . )

Means here already we get  $P(B) = 3$  so getting  $P(A)$  is zero

This condition called Mutually exclusive .



**Bayes therm :**

$P(A|B)$  means prob of a and b \* prob a / Prob b.

Means probability of A if b is given .

# Bayes' Theorem: (1700's)

Thm:

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)} \text{ if } P(B) \neq 0 \checkmark$$

(likelihood)  $P(B|A)$  (prior)  $P(A)$  (posterior)  $P(A|B)$  (evidence)  $P(B)$   
 (wiki)

Proof:

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

$A \cap B$  (A & B)  
 $P(B)$  (defn)

$A \cap B = B \cap A \rightarrow$  set theory

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

$$P(A) P(B|A) = \frac{P(B \cap A)}{P(A)} \rightarrow \text{defn.}$$

output defective

denote the event that a randomly chosen item is defective. Then, we are given the following information:

$P(A_1) = 0.2, P(A_2) = 0.3, P(A_3) = 0.5.$

If the item was made by the first machine, then the probability that it is defective is 0.05; that is,  $P(B|A_1) = 0.05$ . Overall, we have

$P(B|A_1) = 0.05, P(B|A_2) = 0.03, P(B|A_3) = 0.01.$

To answer the original question, we first find  $P(B)$ . That can be done in the following way:

$P(B) = \sum_i P(B|A_i) P(A_i) = (0.05)(0.2) + (0.03)(0.3) + (0.01)(0.5) = 0.024.$

Hence 2.4% of the total output of the factory is defective.

We are given that  $B$  has occurred, and we want to calculate the conditional probability of  $A_3$ . By Bayes' theorem,

$P(A_3|B) = P(B|A_3) P(A_3) / P(B) = (0.01)(0.5) / (0.024) = 5/24.$

Given that the item is defective, the probability that it was made by the third machine is only 5/24. Although machine 3 produces half of the total output, it produces a much smaller fraction of the defective items. Hence the knowledge that the item selected was defective enables us to replace the prior probability  $P(A_3) = 1/2$  by the smaller posterior probability  $P(A_3|B) = 5/24$ .

Interpretations

$P(A_3|B) = \frac{P(B|A_3)P(A_3)}{P(B)}$

## N Bayes algorithm :

$x \rightarrow \text{class}$

$p(C_1|x)$   
 $p(C_2|x)$   
 $p(C_3|x)$  Largest  
 $\vdots$   
 $p(C_k|x)$

The problem with the above formulation is that if the number of features  $n$  is large or if a feature can take on a large number of values, then basing such a model on probability tables is infeasible. We therefore reformulate the model to make it more tractable. Using Bayes' theorem, the conditional probability can be decomposed as

$$p(C_k | \mathbf{x}) = \frac{p(C_k) p(\mathbf{x} | C_k)}{p(\mathbf{x})} \rightarrow p(C_k) p(\mathbf{x} | C_k) = p(\mathbf{x} \cap C_k)$$

In plain English, using Bayesian probability terminology, the above equation can be written as

$$\text{posterior} = \frac{\text{prior} \times \text{likelihood}}{\text{evidence}}$$

In practice, there is interest only in the numerator of that fraction, because the denominator does not depend on  $C$  and the values of the features  $x_i$  are given, so that the denominator is effectively constant. The numerator is equivalent to the joint probability model

$$p(C_k, x_1, \dots, x_n)$$

which can be rewritten as follows, using the chain rule for repeated applications of the definition of conditional probability:

$$\begin{aligned}
 p(C_k, x_1, \dots, x_n) &= p(x_1, \dots, x_n, C_k) \\
 &= p(x_1 | x_2, \dots, x_n, C_k) p(x_2, \dots, x_n, C_k) \\
 &= p(x_1 | x_2, \dots, x_n, C_k) p(x_2 | x_3, \dots, x_n, C_k) p(x_3, \dots, x_n, C_k) \\
 &= \dots \\
 &= p(x_1 | x_2, \dots, x_n, C_k) p(x_2 | x_3, \dots, x_n, C_k) \dots p(x_{n-1} | x_n, C_k) p(x_n, C_k)
 \end{aligned}$$

See above Bayes formula .  $C \times$  means class from  $c_1 \dots c_n$  .

So here we are trying to find prob of  $C \times$  given point  $x$  .

As  $p(X)$  is same for every term we will not consider it here . because we just change class  $c_1 \dots c_n$  each time and we will pick that class whose  $p$  is very high .

So now  $P(C|x_1 \dots x_n)$  which can be written using chain rule by using definition of conditional probability .

Def condition prob =  $P(A|B) = P(B|A) * P(B)$

$$p(C_k, x_1, \dots, x_n)$$

which can be rewritten as follows, using the chain rule for repeated applications of the definition of conditional probability:

$$\begin{aligned} p(C_k, x_1, \dots, x_n) &= p(x_1, \dots, x_n, C_k) \checkmark \\ &= p(x_1 | x_2, \dots, x_n, C_k) p(x_2, \dots, x_n, C_k) \checkmark \rightarrow \text{defn. of cond. prob} \\ &= p(x_1 | x_2, \dots, x_n, C_k) p(x_2 | x_3, \dots, x_n, C_k) p(x_3, \dots, x_n, C_k) \\ &= \dots \\ &= p(x_1 | x_2, \dots, x_n, C_k) p(x_2 | x_3, \dots, x_n, C_k) \dots p(x_{n-1} | x_n, C_k) p(x_n | C_k) p(C_k) \end{aligned}$$

Now computing this probability is really very hard in big train data .

values of the features  $x_i$  are given

probability model

$$p(C_k, x_1, \dots, x_n)$$

which can be rewritten as follows

$$\begin{aligned} p(C_k, x_1, \dots, x_n) &= p(C_k) \\ &= p(C_k | x_1, \dots, x_n) p(x_1, \dots, x_n) \\ &= \dots \\ &= p(C_k | x_1, \dots, x_n) \end{aligned}$$

Now the "naive" conditional independence assumption is made, that each feature is independent of every other feature given the class.

Thus, the joint model can be expressed as:

$$p(C_k | x_1, \dots, x_n) \propto p(C_k) \prod_{i=1}^n p(x_i | C_k)$$

This means that under the above

*Handwritten notes:*

- obese fit lean
- $p(x_1 | x_2, x_3, \dots, x_n, C_k)$
- $p(w=180 | hc=bl, hl=5cm, ec=br, C_k)$
- Train
- Examples

180	bl	5	br	obese
170	br	10	br	lean
160	bl	15	br	fit
150	br	20	br	obese
140	bl	25	br	lean
130	br	30	br	fit
120	bl	35	br	obese
110	br	40	br	lean
100	bl	45	br	fit
90	br	50	br	obese
80	bl	55	br	lean
70	br	60	br	fit
60	bl	65	br	obese
50	br	70	br	lean
40	bl	75	br	fit
30	br	80	br	obese
20	bl	85	br	lean
10	br	90	br	fit
0	bl	95	br	obese

Lets see above image I want to predict  
 $P(\text{weight} = 180 \mid \text{hair color} = \text{blue}, \text{length} = 5\text{cm} \dots, \text{class} = \text{fat})$



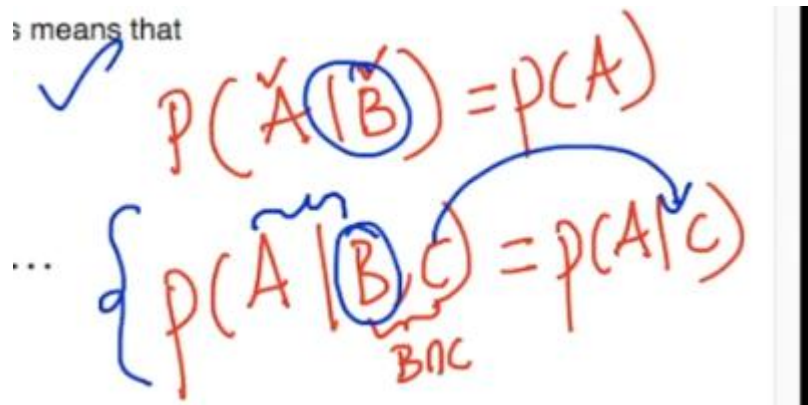
So finding this exact data is very very low prob .

So as per N Bayes algorithm :

means that

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

...

$$\{ P(A | \underbrace{B, C}_{B \cap C}) = P(A | C) \}$$


See first a and B are simple independent because there is no rel between them .

In second we can a and B are conditionally independent . ..

So in N algorithm we Conditional ind concept .

So above image complex equation we make simple :

See below image now we are telling here  $x_i$  is conditionally independent from  $x_{i+1} \dots$

$$\begin{aligned}
p(C_k, x_1, \dots, x_n) &= p(x_1, \dots, x_n, C_k) \\
&= p(x_1 | x_2, \dots, x_n, C_k) p(x_2, \dots, x_n, C_k) \\
&= p(x_1 | x_2, \dots, x_n, C_k) p(x_2 | x_3, \dots, x_n, C_k) p(x_3, \dots, x_n, C_k) \\
&= \dots \\
&= p(x_1 | x_2, \dots, x_n, C_k) p(x_2 | x_3, \dots, x_n, C_k) \dots p(x_{n-1} | x_n, C_k) p(x_n | C_k) p(C_k)
\end{aligned}$$

Now the "naive" conditional independence assumptions come into play: assume that each feature  $x_i$  is conditionally independent of every other feature  $x_j$  for  $j \neq i$ , given the category  $C$ . This means that

$$p(x_i | x_{i+1}, \dots, x_n, C_k) = p(x_i | C_k) \Rightarrow x_1 \text{ is indep of } x_{i+1}, x_{i+2}, \dots, x_n \text{ given } C_k$$

Thus, the joint model can be expressed as

$$p(x_i | x_{i+1}, \dots, x_n, C_k) = p(x_i | C_k).$$

Thus, the joint model can be expressed as

$$\begin{aligned}
p(C_k | x_1, \dots, x_n) &\propto p(C_k, x_1, \dots, x_n) \\
&\propto p(C_k) p(x_1 | C_k) p(x_2 | C_k) p(x_3 | C_k) \dots \\
&\propto p(C_k) \prod_{i=1}^n p(x_i | C_k).
\end{aligned}$$

This means that under the above independence assumptions, the conditional distribution over the class variable  $C$  is:

$$p(C_k | x_1, \dots, x_n) = \frac{1}{Z} p(C_k) \prod_{i=1}^n p(x_i | C_k)$$

where the evidence  $Z = p(\mathbf{x}) = \sum_k p(C_k) p(\mathbf{x} | C_k)$  is a scaling factor dependent only on  $x_1, \dots, x_n$ , that is, a constant

if the values of the feature variables are known.

Pi is used to denote multiplication operation .

Z = p(X) which we neglect on top .

Contents - Google Docs x ShatterLine Blog » Not-so-Naive x Naive Bayes classifier - Wiki x Chekur Srikant...

shatterline.com/blog/2013/09/12/not-so-naive-classification-with-the-naive-bayes-classifier/

### The Learning Phase

In the learning phase, we compute the table of likelihoods (probabilities) from the training data. They are:

$P(\text{Outlook}=o | \text{Class}_{\text{play}}=b)$ , where  $o \in [\text{Sunny, Overcast, Rainy}]$  and  $b \in [\text{yes, no}]$

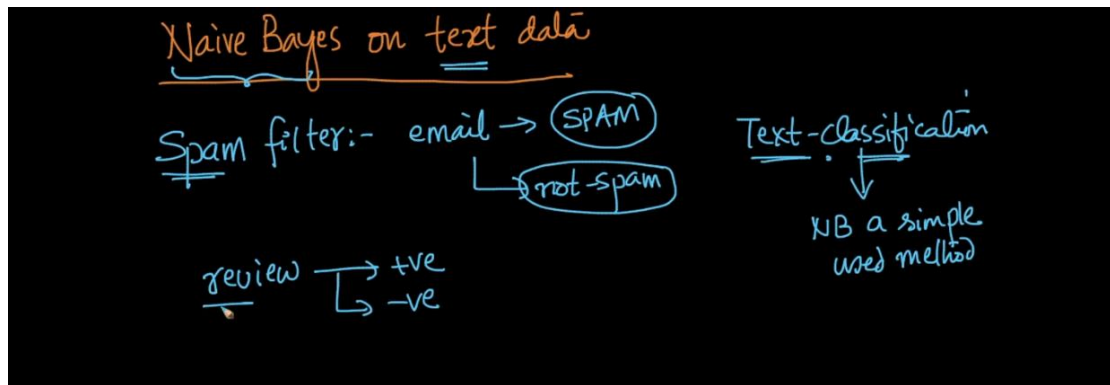
$P(\text{Temperature}=t | \text{Class}_{\text{play}}=b)$ , where  $t \in [\text{Hot, Mild, Cool}]$  and  $b \in [\text{yes, no}]$ ,

$P(\text{Humidity}=h | \text{Class}_{\text{play}}=b)$ , where  $h \in [\text{High, Normal}]$  and  $b \in [\text{yes, no}]$ ,

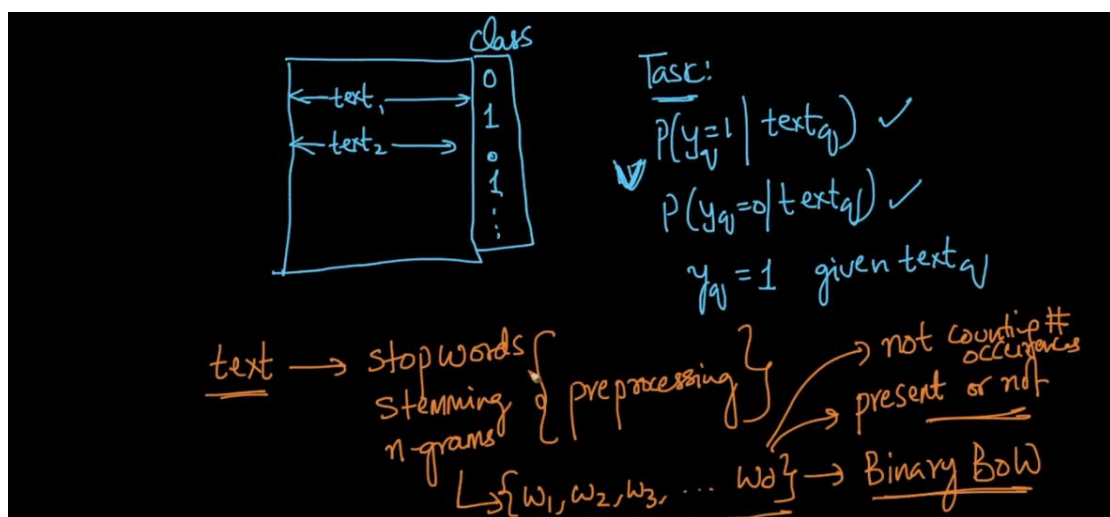
$P(\text{Wind}=w | \text{Class}_{\text{play}}=b)$ , where  $w \in [\text{Weak, Strong}]$  and  $b \in [\text{yes, no}]$ .

P(Outlook=o   Class <sub>play</sub> =b)	Frequency		Probability in Class	
Outlook =	Play=Yes	Play=No	Play=Yes	Play=No
Sunny	2	3	2/9	3/5
Overcast	4	0	4/9	0/5
Rain	3	2	3/9	2/5
	total= 9	total=5		





Now our task is text classification here we have text data with 2 class label 0 and 1 we need to predict new text class .



Using N Bayes we can write :

$$P(y=1 | \text{Text}) = p(y=1 | w_1, w_2, \dots, w_n)$$

Simply we can use N Bayes formula . :

text  $\xrightarrow{\text{preproc}}$   $\{w_1, w_2, \dots, w_d\}$

$$p(y=1 | \text{text}) = p(y=1 | \underbrace{w_1, w_2, \dots, w_d}_{\text{features}})$$

class prior  $\propto p(y=1) * p(w_1 | y=1) * p(w_2 | y=1) \dots p(w_d | y=1)$

likelihood  $\propto p(y=1) * \prod_{i=1}^d p(w_i | y=1)$

$$p(y=0 | \text{text}) \propto p(y=0) * \prod_{i=1}^d p(w_i | y=0)$$


---

$p(y=1) = \frac{\# \text{ Train pts with } y=1}{\text{Total } \# \text{ Train pts}}$   
 $p(y=0) = \frac{\# \text{ Train pts with } y=0}{\text{Total } \# \text{ Train pts}}$

$\leftarrow \text{text}_i \rightarrow$   

0
1
0
0
1

  
 Train

$P(Y=1)$  and  $P(Y=0)$  we can easily cal but main task is likelihood prob find .

How can we do this ?

Simple we split train data half part like all class label 1 point separate from 0 points .

Simply then we can find probability for each word in text .

See below image in this way we can find P for all words in a text and we can tell whether new text belongs to class 1 or 0 .

Text classification performance By N algorithm is very very simple and very good also .

$$P(\underline{w_i} | \underline{y=1}) = \frac{\# \text{ train data ples with contain } w_i \text{ \&\& } y=1}{\# \text{ train data ples with } y=1}$$

✓ Text-classification problems 15:02 / 15:06

{ Spam-detection  
polarity of a review } → NB is a very good baseline

benchmark

✓ (1) NB → acc (98%)  
✓ (2) LR →  
✓ (3) GBDT →  
✓ (4) DL → 98.5%

✓ DL → 98.5%  
NB → 95%

## Laplace Smoothing :

The image shows handwritten notes on a black background. At the top, 'Laplace Smoothing:' is written in orange and underlined. Below it, 'Training:-' is written in white. To the right, a large curly bracket groups two columns of probability expressions. The first column contains  $P(y=1)$ ,  $p(w_1|y=1)$ ,  $p(w_2|y=1)$ , a vertical ellipsis, and  $p(w_m|y=1)$ . The second column contains  $P(y=0)$ ,  $p(w_1|y=0)$ ,  $p(w_2|y=0)$ , a vertical ellipsis, and  $p(w_m|y=0)$ . An arrow points from the text 'class priors' to the  $P(y=1)$  and  $P(y=0)$  terms. Another arrow points from the text 'likelihoods' to the conditional probability terms in the second column.

Laplace Smoothing:

Training:-

$P(y=1)$  ;  $P(y=0)$  ← class priors

$\left\{ \begin{array}{ll} p(w_1|y=1) & p(w_1|y=0) \\ p(w_2|y=1) & p(w_2|y=0) \\ \vdots & \vdots \\ p(w_m|y=1) & p(w_m|y=0) \end{array} \right\}$  likelihoods

While training all P are computed for all words .

Now while testing new word come and its not available in train then how we will solve this problem .

So we cant directly ignore that word we are ignoring any word means indirectly we are telling that new word belong to class 1 . this is wrong .

See below image

$$\begin{aligned}
 P(w' | y=1) &= \frac{P(w', y=1)}{P(y=1)} \\
 &= \frac{\# \text{pts st } w' \text{ occurs in } y=1}{n_1} \\
 &= \frac{0}{n_1} = 0
 \end{aligned}$$

See instead of 0 we add alpha see below image  
 and k means distinct possibility in this case only 2  
 hence  $k=2$  . means word is present or not  
 present only 2 possibility .

Laplace Smoothing (not Laplacian Smoothing)  
Additive Smoothing

$$P(w' | y=1) = \frac{0 + \alpha}{n_1 + \alpha k}$$

$1 \rightarrow \text{present}$   
 $0 \rightarrow \text{not present}$

$k = \# \text{ distinct values}$   
 $k = 2$  (since  $w'$  can take 2 values)

How solve that 0 problem in image 1 . simple lets  
 say  $n = 100$  here and  $\alpha = 1$  if we put this value  
 we will get solution from that zero problem .

$$p(\underline{w'} | y=1) = \frac{0 + \alpha}{100 + 2\alpha}$$

Let  $n_1 = 100$

$\alpha = 0.001$

$k=2$  because  $w = 0$  or  $1$

Case 1:-  $\alpha = 1 = \frac{1}{102} \neq 0$

$0 \neq p(y=1 | \text{test}_0) \leftarrow p(w' | y=1) \neq 0$

Lets take 2<sup>nd</sup> case alpha = 10k

So here we tell that  $p(w' | y=1) = p(w' | y=0)$  is equally same .

Ans this is too good because we don't know anything about that new word .

Case 2:-  $\alpha = 10000$

$$p(\underline{w'} | y=1) = \frac{0 + 10000}{100 + 20000} = \frac{10000}{20100} \approx \frac{1}{2}$$

$n_1 = 100$

$\checkmark w' = 0 \rightarrow$  two possibilities

$\checkmark w' = 1$

$$p(w' | y=1) = p(w' | y=0) = \frac{1}{2}$$

In this way Laplace will help to solve the problem of unknown word .



In real what we can do we take Laplace for all points means even word is present or not we add Laplace term see below images .

Laplace Smoothing

for all words

$$P(w_i | y=1) = \frac{(\# \text{ data pts with } w_i \text{ s.t. } y=1) + \alpha}{(\# \text{ data pts } y=1) + \alpha k}$$

present in my training data

often times  $\alpha = 1$   
add one smoothing

Default value is alpha =1 as we increase alpha we move towards uniform dist .

Lets see below image say we have very less data , means word  $w_1$  occurs only 2 times and P of  $y = 1$  only 50 points are in my data set .

$$P(w_1 | y=1) = \frac{2 + \alpha}{50 + \alpha k} = \frac{2}{50}$$

$\alpha = 1 \Rightarrow \frac{2+1}{50+1} = \frac{3}{51}$

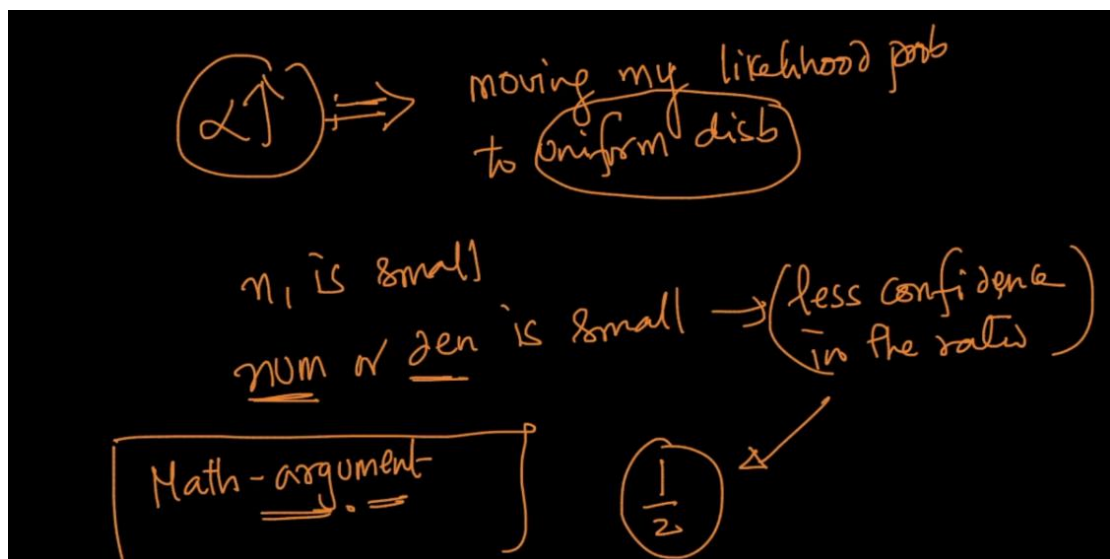
$\alpha = 10 \Rightarrow \frac{2+10}{50+20} = \frac{12}{70}$

$\alpha = 100 \Rightarrow \frac{2+100}{50+200} = \frac{102}{250}$

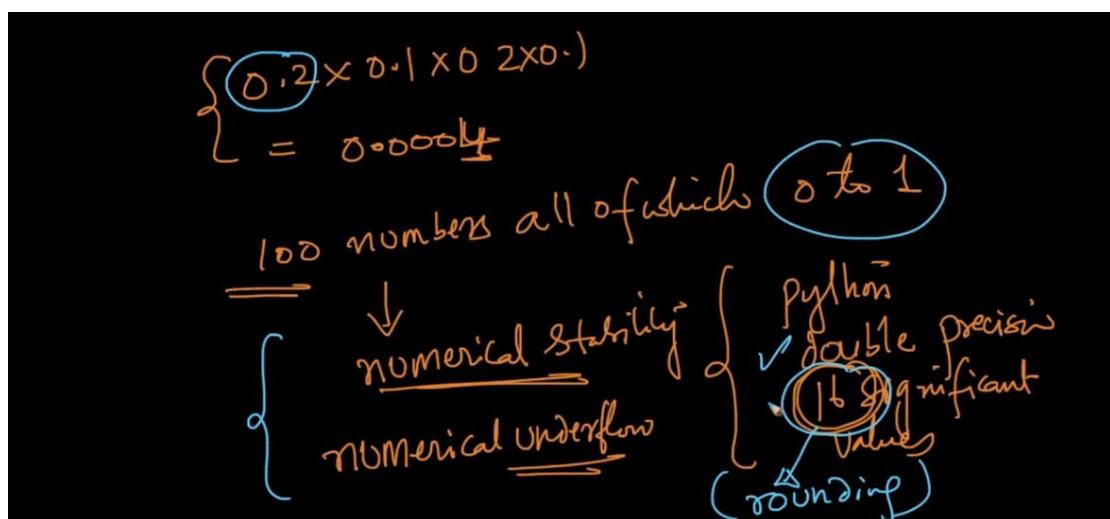
$\alpha = 1000 \Rightarrow \frac{2+1000}{50+2000} = \frac{1002}{2050} = \frac{1}{2}$

See as alpha increase we are moving toward uniform dist . because at the end we get almost  $1/2$  value .

Ans this is really good because if we see above example confidence on data is very low so its better to predict 50% (instead of say 1 or 0) ..



Log Probabilities :



We know that P values always between 0 to 1 so its very difficult to maintain number stability .

Lets  $p = 0.0004$  .

Also multiplying small numbers is really very hard so we convert values of P in log ..

Log has very beautiful properties :

1. Convert multiplication into sum .
2. Convert exp into multiplication .

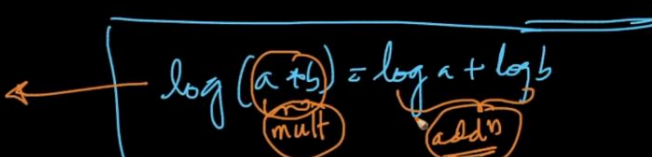
Log-probabilities:

$$\log(P(y=1 | w_1, w_2, \dots, w_d)) = \log(P(y=1) \cdot \prod_{i=1}^d p(w_i | y=1))$$
$$\log(P(y=0 | w_1, w_2, \dots, w_d)) = \log(P(y=0) \cdot \prod_{i=1}^d p(w_i | y=0))$$

$x \uparrow ; \log x \uparrow$   
 $\hookrightarrow$  monotonic fn

Above N Bayes formula also convert to sum .

$$\log(P(y=1|w_1, w_2, \dots, w_d))$$

$$= \log(P(y=1)) + \sum_{i=1}^d \log(P(w_i|y=1))$$


So if values are very small then log prob will be big negative number so sum of it is really very easy and time saving also .

### Bias Variance Trade Off N Bayes Laplace :

High bias = Under fir

High Variance = Over fit.

Here this is depend on value of alpha . same like KNN when  $k = 1$  very small we saw over fit model and if  $K = n$  then under fit model .

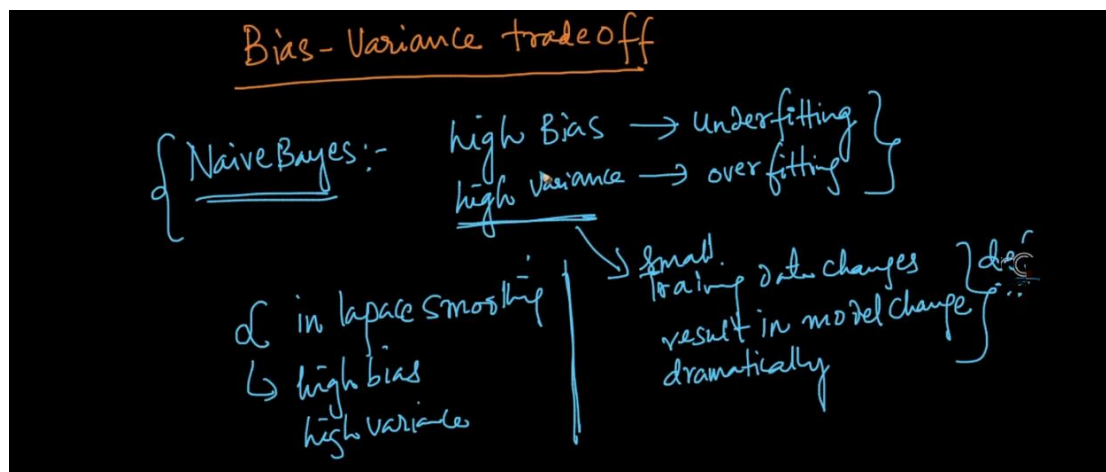
Now lets take Case 1:

Alpha = 0 .

Now lets say we have data set of 2000 values out of that 1000 are positive and 1000 are neg points.

Now new word come and we want to predict whether it belong to + or - .

We saw in our data that word occurs very very low :  
 $2/1000$  2 times only .



Case 1:-  $\alpha = 0$

$$p(w_i | y=1) = \frac{\# \text{ Train data p's } w_i \text{ occurs \& } y=1}{\# \text{ Train p's with } y=1}$$

$(n = 2000 \text{ p's } \rightarrow \begin{array}{l} \rightarrow 1000 \text{ +ve} \\ \rightarrow 1000 \text{ -ve} \end{array})$

$= \frac{2}{1000} \rightarrow \text{only 2 times} \Rightarrow \underline{\text{rare}}$   
 $1000 \rightarrow \text{+ve data p's}$

{ Words that are rare  $\uparrow p(w_i | y=1)$  }  $\rightarrow$  overfitting

So as that word occurs only 2 times if I made very small change in data means I just remove that word then I will get  $0/1000$  means direct 0 prob .

See how drastic change here occurs and this is called high variance over fit problem .

Since  $w_i$  occurs only in (2) out of 2000 cases  
 1000 true 1000 -ve

Small change in my D-train  
 → remove the 2 texts ( $x_i$ 's) that contain  $w_i$

$p(w_i | y=1) = \frac{2}{1000}$  to  $\frac{0}{1000}$

Case 1:  $\alpha = 0 \Rightarrow$  Small change in D-train results in large change in the model  $\Rightarrow$  high var  $\Downarrow$  overfitting

Case 2:  $\alpha$  is v. large: - ( $\alpha = 10000$ )

$p(w_i | y=1) = \frac{2 + 10000}{1000 + 20000} \approx \frac{1}{2}$   
 $\downarrow$   
 $0 \approx 1$  ( $K=2$ )

$\alpha = \text{v-large:}$

Compare  $p(y=1 | \overbrace{w_1, w_2, \dots, w_n}^{xw}) = \underbrace{p(y=1)}_{n_1/n} \cdot \underbrace{\prod_{i=1}^d p(w_i | y=1)}_{\approx 1/2}$

$p(y=0 | w_1, w_2, \dots, w_n) = \underbrace{p(y=0)}_{n_2/n} \cdot \underbrace{\prod_{i=1}^d p(w_i | y=0)}_{\approx 1/2}$

$\alpha_q \xrightarrow{\text{NB}} y_q = 1$   $n \rightarrow \begin{cases} n_1 & +ve \\ n_2 & -ve \end{cases}$  ( $n_1 > n_2$ )



Now if alpha is very large then It will behave same like KNN whatever class has high probability then it will blindly declare new point belongs to that class only .

This is high bias problem .

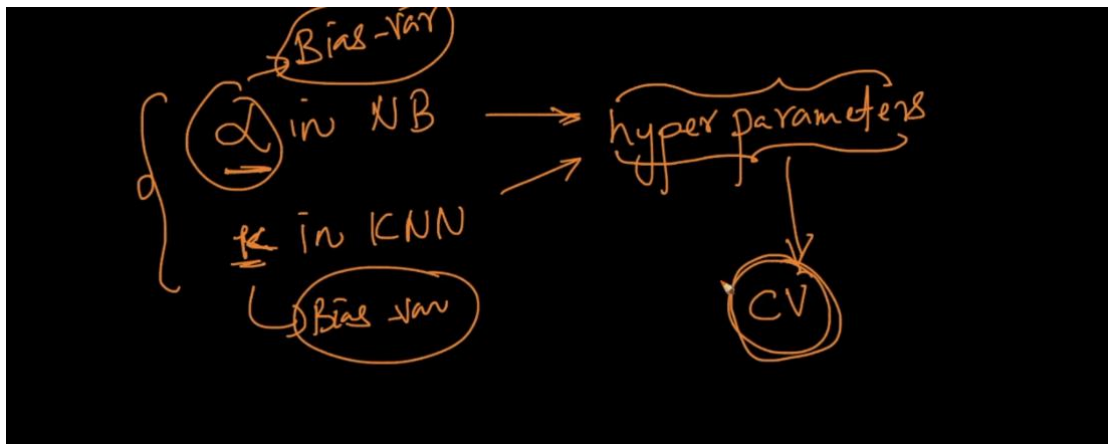
Case 2:  $\alpha \rightarrow \text{very large}$   
Underfitting  $\rightarrow$  high bias

Case 1:  $\alpha = 0$   
overfitting  $\rightarrow$  high -var

(Q) How to find the right  $\alpha$

$\rightarrow$  KNN ; right  $K \rightarrow$  using Simple CV  
or 10 fold CV

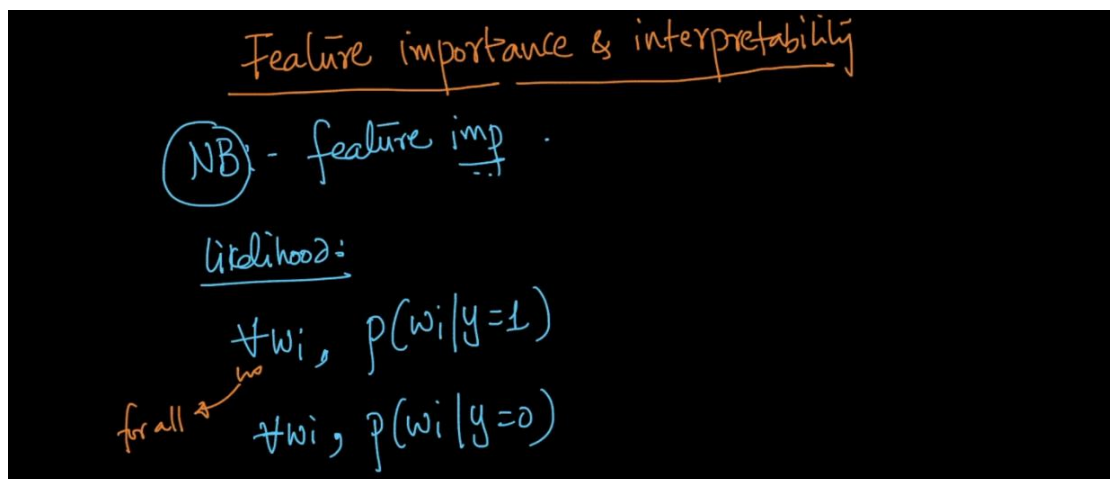
$\rightarrow$  right  $\alpha$  :  $\rightarrow$  Simple CV  
or 10 fold CV

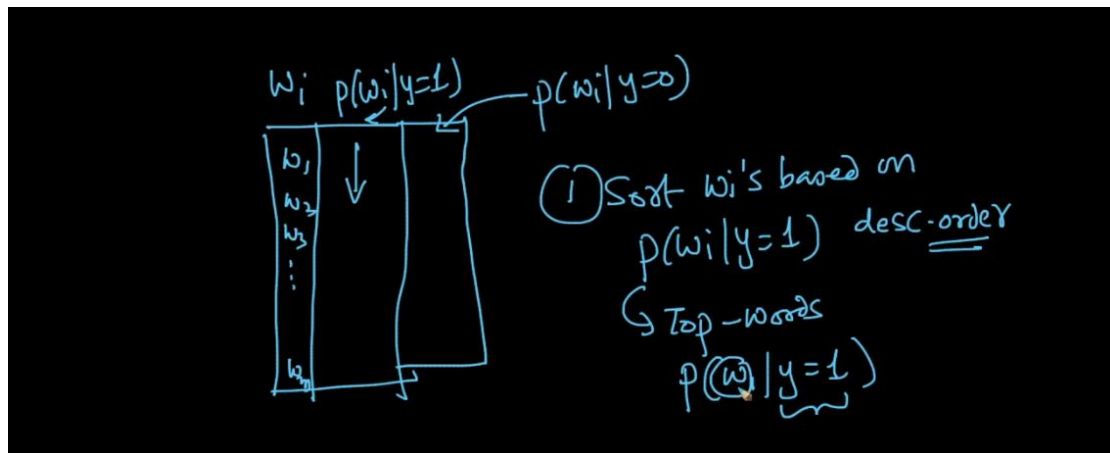


Alpha and K are hyper parameter and they cal by Cross validation method .

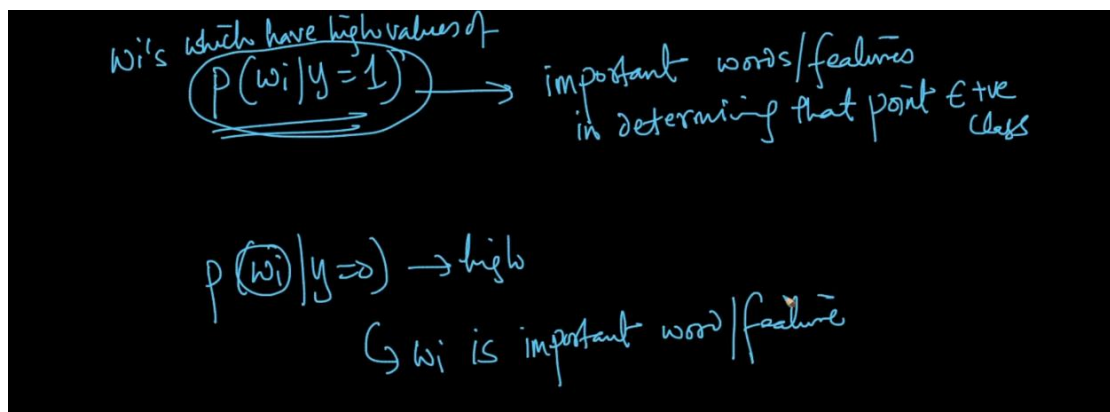
Feature Importance :

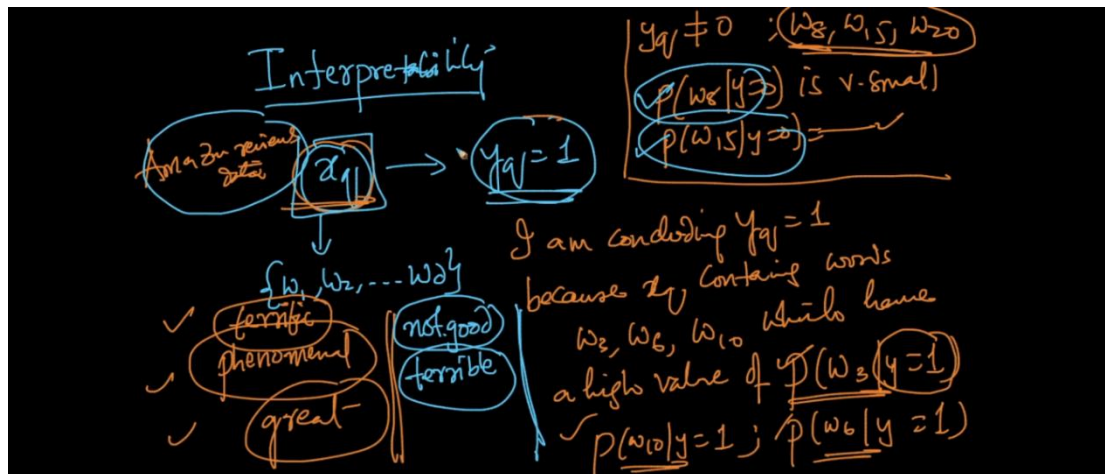
Here while training model we have P of all words .  
So just by sorting them we can tell which feature is important .





See below image lets say we have  $w_1$  feature if its likelihood  $P$  is very high then we can say its important feature belong to class 1 or 0 .



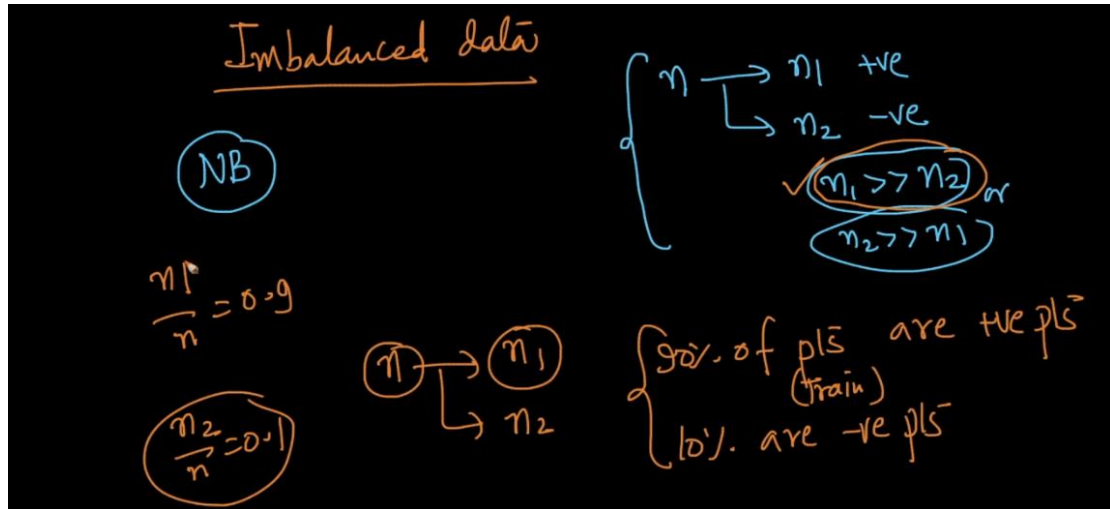


Now how interpret N Bayes is ?

Its really very Interpret lets say new review come and we want to predict review is + or - and also why ?

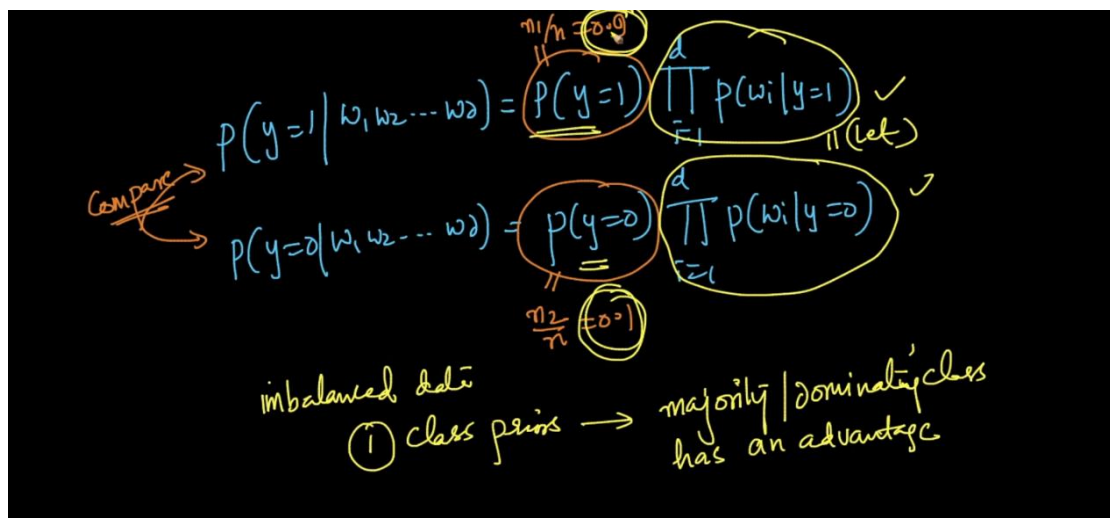
Simple our model will break review into words and we already cal prob of all train data words so it will check that prob for each word in new sent and as per that prob it will say I found  $w_1, w_3, w_4$  in new review has very high + prob and for  $w_2, w_7 \dots$  low probability (-) so this review is + ..

Imbalance Data :



See above image data is totally imbalanced  $n_1$  too much greater than  $n_2$  .

At the we have to compare all the prob that is our obj .



See above image 1<sup>st</sup> value is 9 times greater than second so it will affect too much on model output .

Because for any single word we get high probability in 1<sup>st</sup> case than 2<sup>nd</sup> .

Soln: (1) upsampling (or) downsampling  
 $\hookrightarrow n_1 \approx n_2$   $\boxed{p(y=1) = p(y=0) = \frac{1}{2}}$

(2) drop  $p(y=1)$  &  $p(y=0)$   
 $p(y=1) = p(y=0) = 1$

We use above we can make  $n_1 = n_2$  or  $n_2 = n_1$  ..

Another problem with not balanced data :

Lets say we are using Laplace then impact of alpha on minority class will be very high and very low on majority class .

imbalanced data

majority (+ve)  $n_1 = 900 \rightarrow p(w_i | y=1) = \frac{[ ]}{900} \rightarrow 0.690$

minority (-ve)  $n_2 = 100 \rightarrow p(w_i | y=0) = \frac{[ ]}{100} \rightarrow 0.610$

$\alpha = 10$

Laplace Smoothing

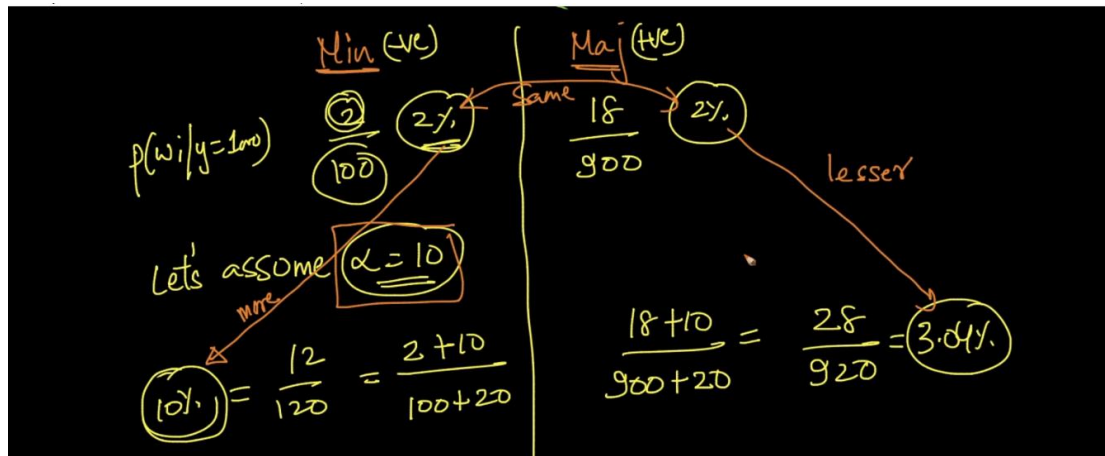
Same  $\alpha$  for +ve & -ve

minority  $\alpha$  impacts more

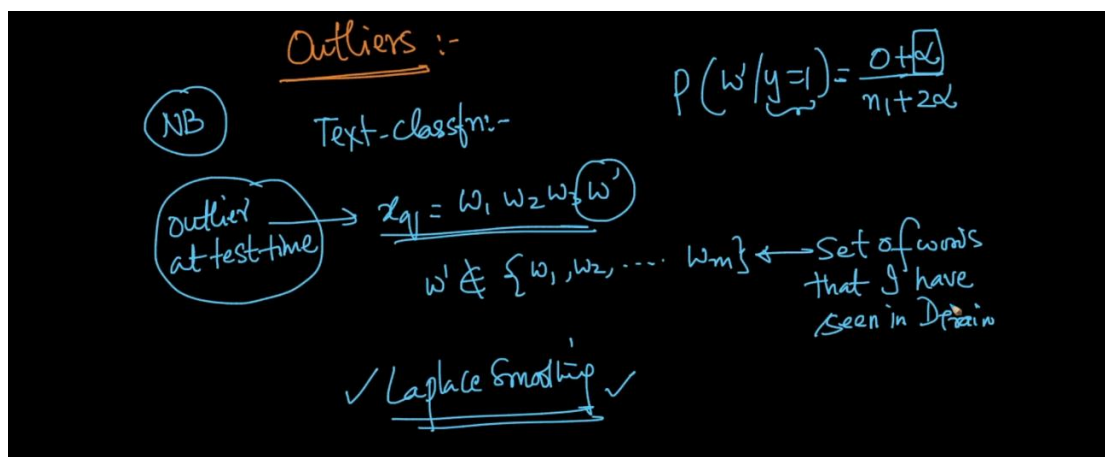
majority  $\alpha$  impacts less



See below image we start for both class with 2 % data and at the end we get very diff result minority class with high prob as compare to majority class.



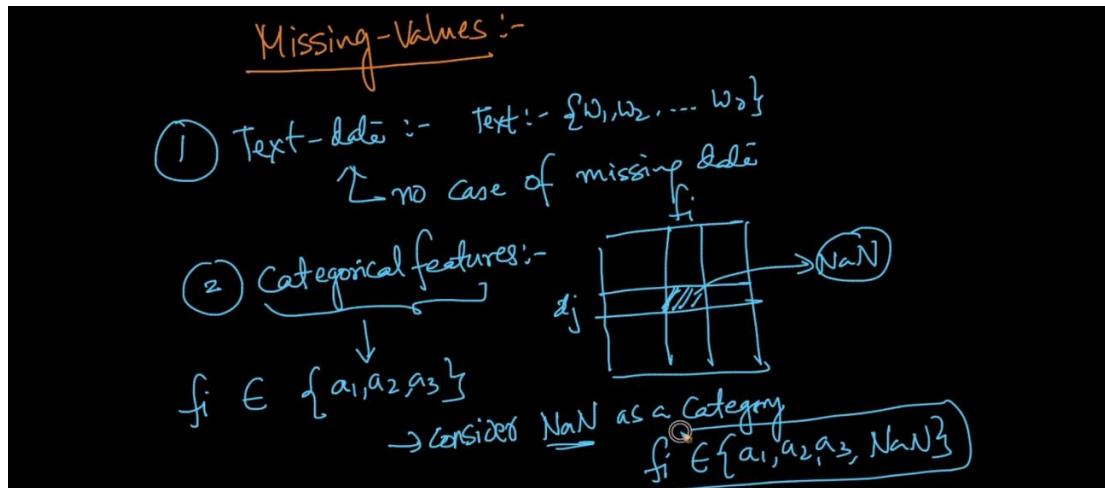
Out Li :



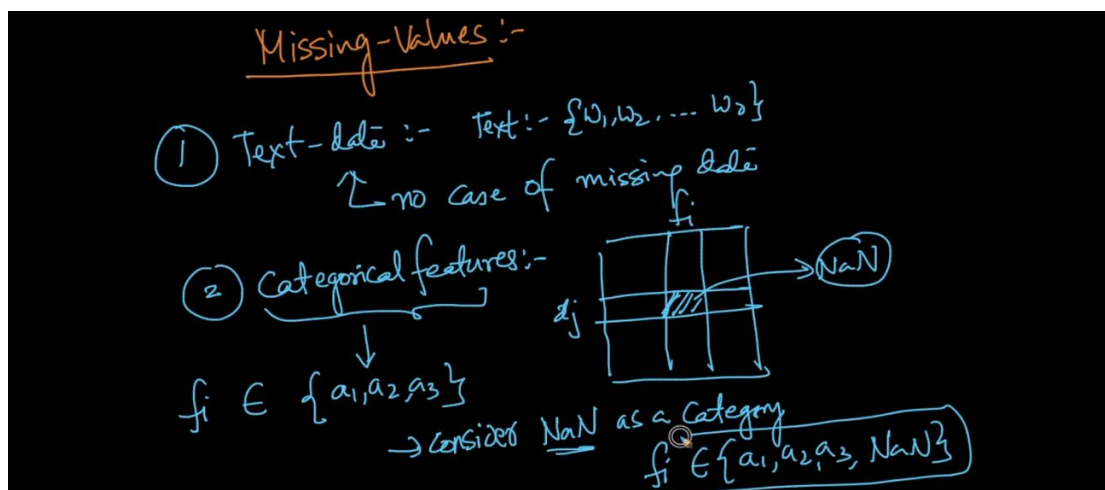
If any word occurs very very less time on both class in then it is out l.

We can avoid simply by telling if any word lets say occurs less than 10 then just ignore that word.

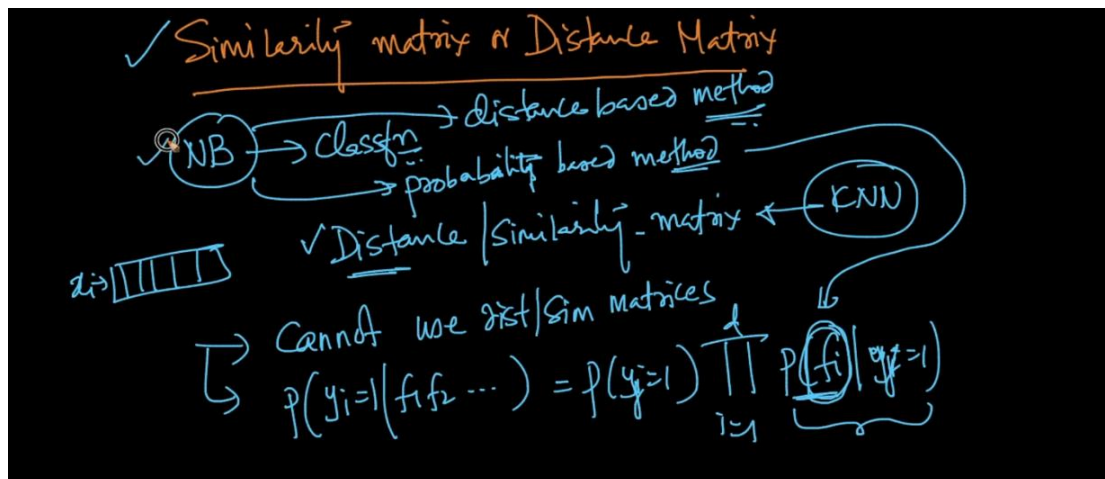
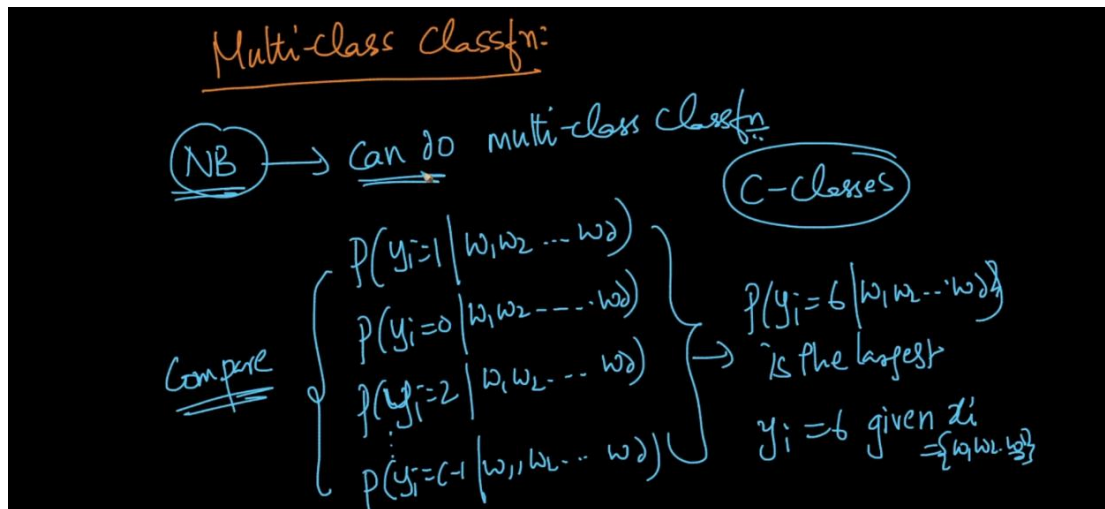
Another method is simply use Laplace Smoothie with good alpha .



For text data no missing values because text is group of word  $w_1, w_2, w_3 \dots$  so no missing .

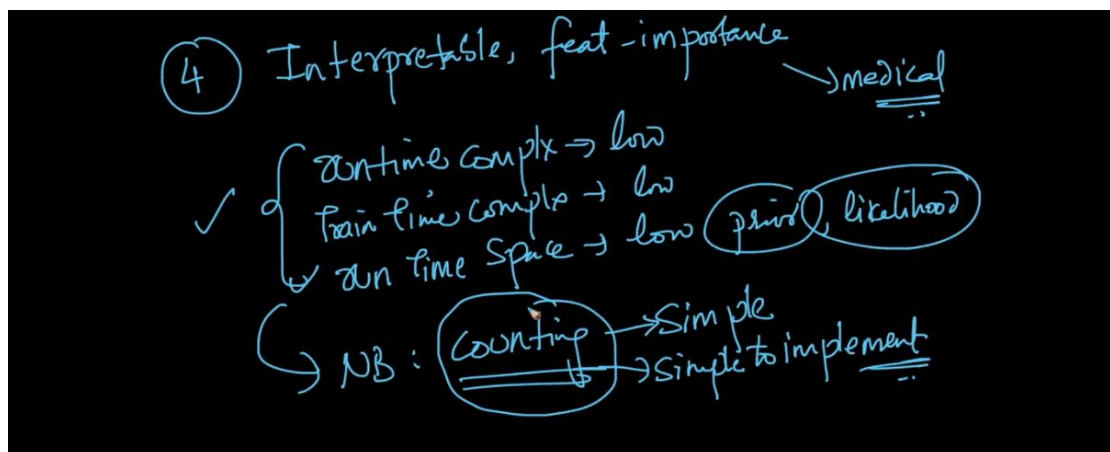
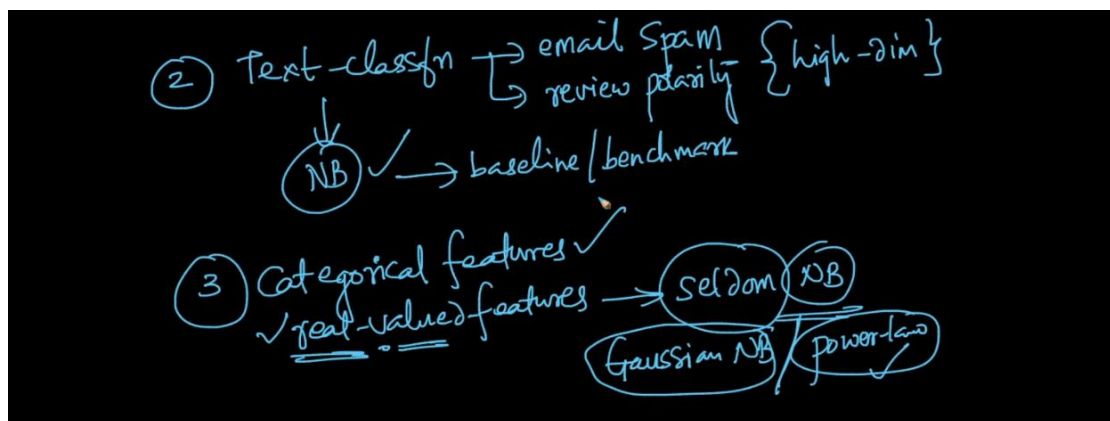
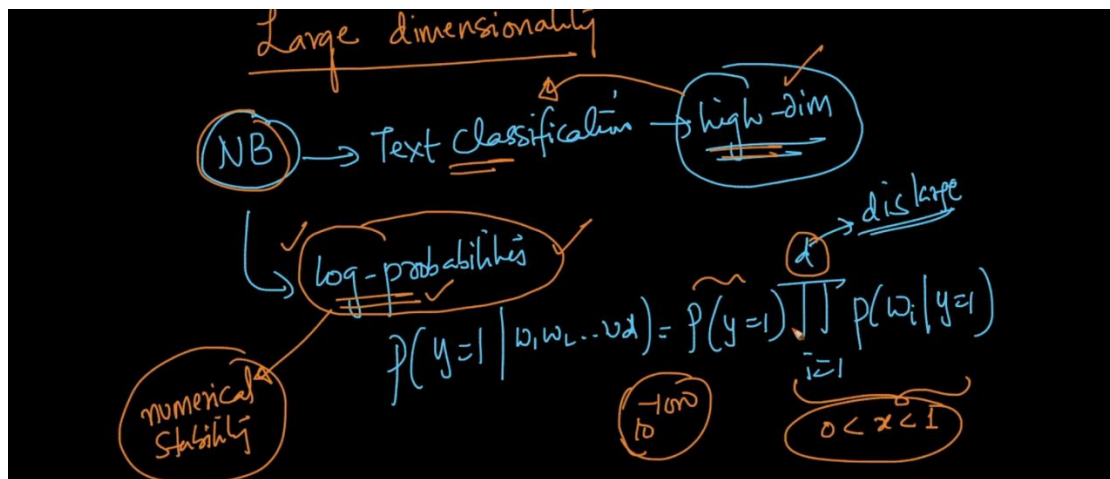


For 2<sup>nd</sup> type whatever missing with Nan values we will consider it self as new category .



Because N Bayes is not distance based method it is probabilistic method.

N Bayes can easily used for very large dimension data just make sure use log of prob .



⑤ easily overfit if you don't do Laplace Smoothing

$\alpha : CV$

Learn code from SKLEARN library .