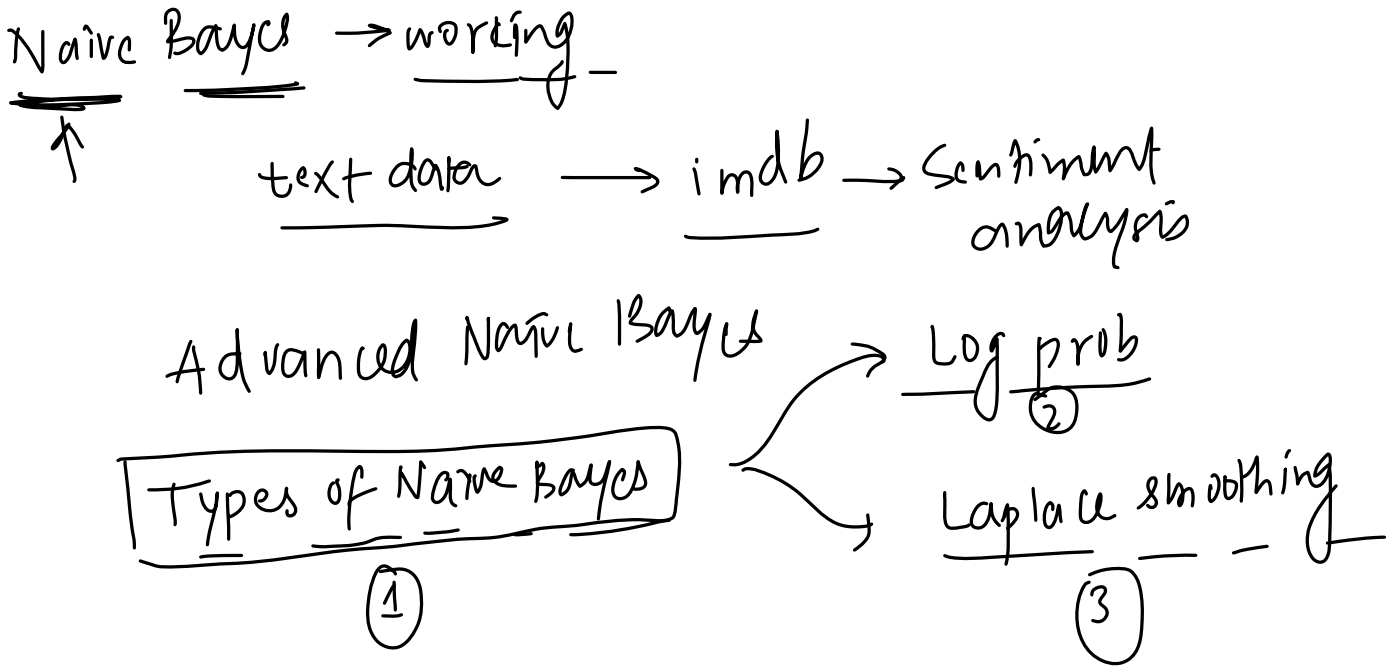
Recap

24 June 2023 09:32



Numerical Stability

23 June 2023 15:06

Underflow → represent decimals in memory

floating point

binary 0 and 1

0.0000001 X3

→ 0 cells

0.0000001 < 0.00003

0 = 0

2500

100 cols

cgpa | iq | placement

{8.1, 81} → Y | N

$P(Y \mid 8.1, 81) = P(Y)\, P(8.1 \mid Y)\, P(81 \mid Y)$

$0 < x < 1$

→ 2500 prob

$P(N \mid 8.1, 81) = 0$

$\log(PY) + \log(P(8.1 \mid Y)) + \log(P(81 \mid Y)) + 2500 \cdots$

$0.1 \times 0.3 \times 0.4 \times 0.6$

0.0000

= log-prob

log probabilities

$\log(P(a)\,P(b)\,P(c)\,P\ldots)$

$\log(ab) = \log a + \log b$

log (0.5)

−ve number → y = −153

N = −135

N class ✓

log (0.3 × 0.5 × 0.7)

log a + log b

$\log(0.5) + \log(0.5) + \log(0.7)$

$-1.2 \quad -0.7 \quad -0.3 = \boxed{-2.2}$

N

What is Underflow in Computing

24 June 2023 07:47

Underflow is a condition that can occur in computing when a number nears zero and the computer can no longer store it accurately in memory using floating-point representation. It happens when a calculated result is a smaller absolute value than the computer can actually represent.

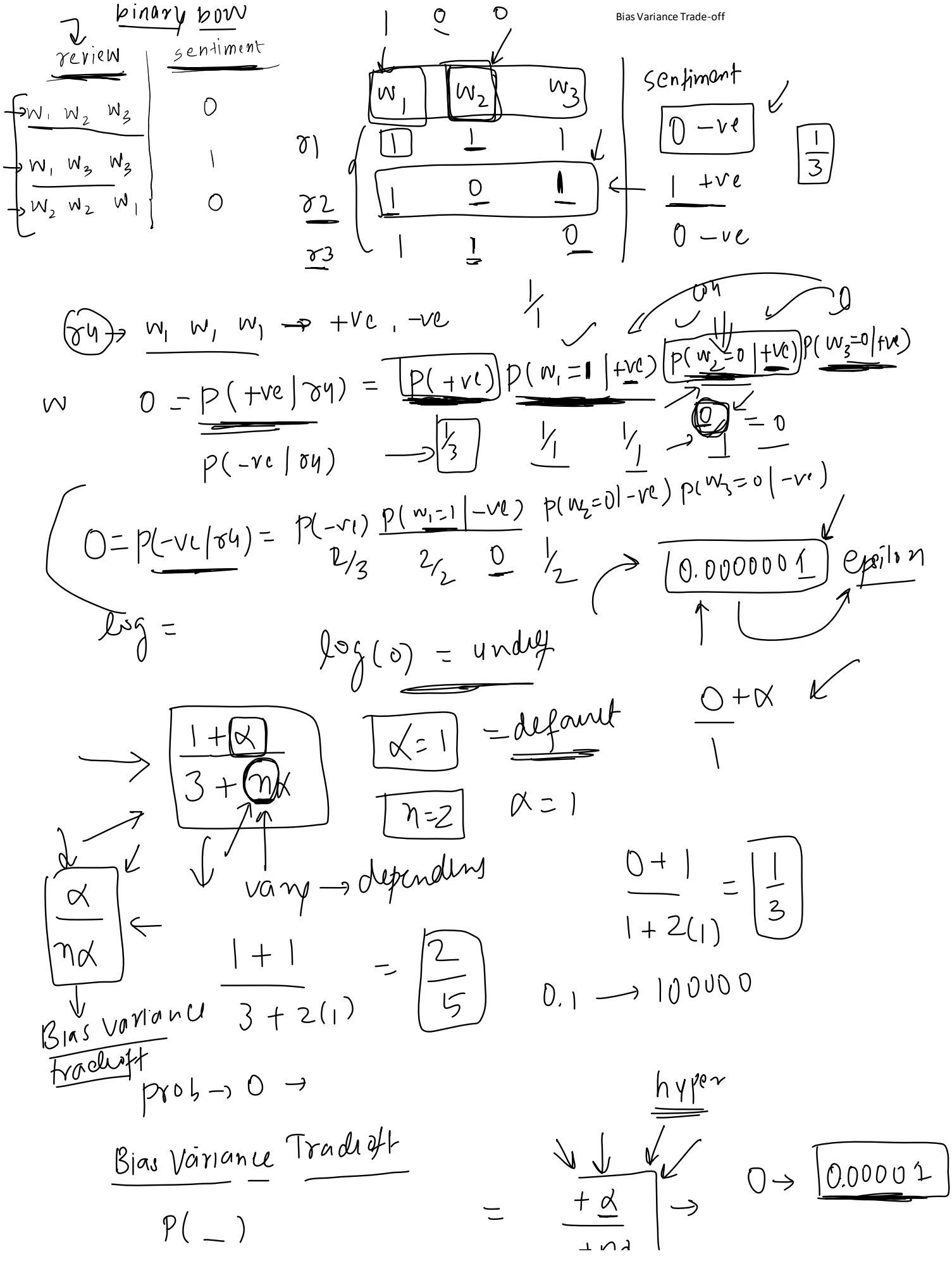
Most computers use a form of representation called floating-point to represent real numbers. This representation has a certain precision limit, and it can only represent numbers between a certain minimum and maximum value. If a number is too close to zero (but not zero), it might be smaller than the smallest representable positive number in the machine's floating-point representation. When an operation on such small numbers is performed, the machine might round the result to zero, leading to a loss of precision.

Underflow can be a problem in certain domains, such as machine learning, where calculations often involve probabilities. Probabilities are positive numbers that can be very close to zero. When multiplying many small probabilities together, the result can underflow. One common way to avoid underflow in such scenarios is to perform calculations in the log domain, where addition and subtraction are used instead of multiplication and division, thereby maintaining higher numerical precision.

Laplace Additive Smoothing

23 June 2023 15:13

Bias Variance Trade-off



$P(\_)$ → Laplace smo

$= \frac{+\alpha}{+n\alpha}$ →

↓

flexibility → control bias and variance

α { high bias
$\alpha \to$ low bias
high variance $\alpha \to$ low variance

α = small → $\alpha = 0$ min min = 0 0.063 0.02

100 Y = | f_1 | f_2 | f_n ... | Y | −500 ← sample −500

500 rows N = 100

$P(y|x) = P(y)\,P(f_1|y)\,P(f_2|y)\,P(f_3|y)\ldots$

$P(y|x)$ = 0

small $\frac{3}{500}$, $\frac{12}{500}$ $P(N|x)$

$\frac{0}{500}$ $\frac{1}{500}$

high variance

$\alpha = 0, 0.01, 0.007$

$n = 2$ $n\alpha$ overfitting ← high variance

α = very high = 1000

$\frac{1+1000}{500+2000} = \frac{1001}{2500} = \frac{10}{25} = \frac{2}{5}$

= 10000

$\frac{1+10000}{500+20000} = \frac{10000}{20500} \simeq \frac{1}{2}$

α = 100000

$\frac{100001}{200500} = \frac{1}{2}$ $\alpha \uparrow$ $\frac{1}{2}$ $\frac{+\alpha}{+n\alpha} = \frac{1}{2}$

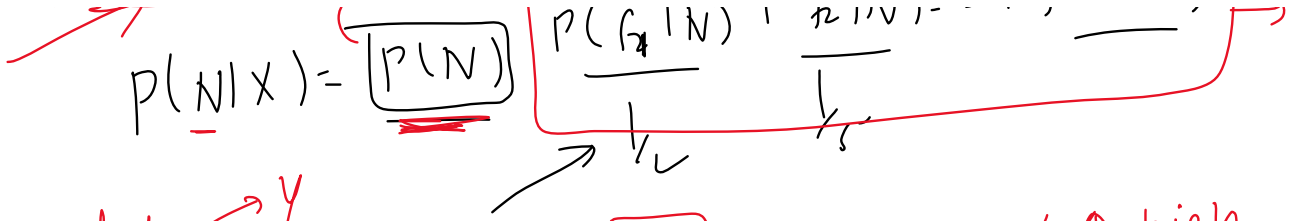
$\alpha = 1$

Y

$P(y|x) = P(y)\,P(f_1|y)\,P(f_2|y)\ldots P(f_n|y)$ ½ ½

$P(N|X) = P(N)\,P(f_1|N)\,P(f_2|N)\ldots P(f_n|N)$ 1

same



data → Y
→ N
underfitting

X → query

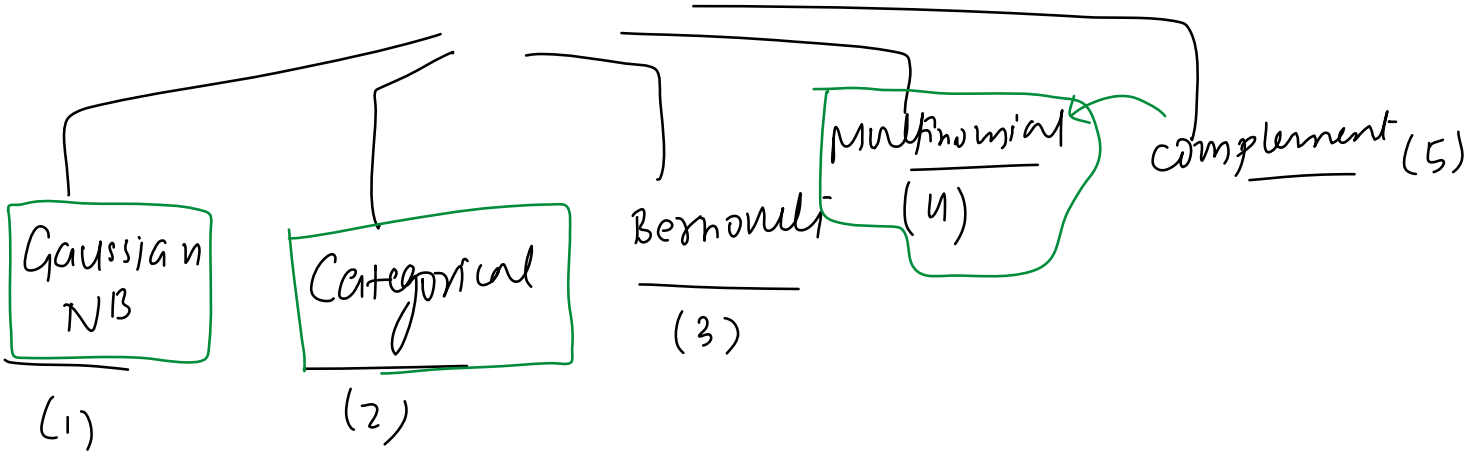
α ↑ high
lead to high bias
or
underfitting

α = hyperparameter
tune

α ↓ low = 0
high variance
or
overfitting

Types of Naïve Bayes

24 June 2023 07:48



Categorical Naïve Bayes

23 June 2023 15:12

Categorical Naive Bayes is a variant of the Naive Bayes algorithm designed specifically to handle categorical data.

Data – all features are categorical

X_q { Sunny, Hot, High, False } → Y/N

Outlook	Temperature	Humidity	Windy	PlayTennis
Sunny	Hot	High	False	No
Sunny	Hot	High	True	No
Overcast	Hot	High	False	Yes
Rainy	Mild	High	False	Yes
Rainy	Cool	Normal	False	Yes
Rainy	Cool	Normal	True	~~No~~
Overcast	Cool	Normal	True	Yes
Sunny	Mild	High	False	No
Sunny	Cool	Normal	False	Yes
Rainy	Mild	Normal	False	Yes
Sunny	Mild	Normal	True	Yes
Overcast	Mild	High	True	Yes
Overcast	Hot	Normal	False	Yes
Rainy	Mild	High	True	~~No~~

$$P(Y|X_q) = \frac{P(Y)}{9/14} \; P(Out=Sunny|Y)\; P(temp=Hot|Y)$$

likelihood

$$P(N|X_q) = \frac{P(N)}{5/14} \; P(Out=Sunny|N)\; P(temp=Hot|N)$$

Laplace additive smoothing

$$P(Out=Sunny|N) = \frac{3}{5} \xrightarrow{+\alpha} \frac{3+\alpha}{5+\alpha n_j} = \frac{3+1}{5+1(3)}$$

(N_{ic}: 3, N_c: 5)

$$P(temp=Hot|N) = \frac{2+1}{5+1(3)} = \frac{3}{8}$$

Question on Categorical NB

24 June 2023 14:40

Let's say I have 4 features in my dataset 2 of them are categorical like gender and is married and 2 are numerical like age and height. Can I apply Categorical Naïve Bayes?

1. Transform numerical features into categorical ones: You can discretize numerical features by binning them into different categories. For example, you could create an "age group" feature that bins age into categories like "0-18", "19-35", "36-50", "51+". This allows you to treat the numerical feature as categorical, so you can use Categorical Naive Bayes. However, you should be aware that this may lead to loss of information, as binning reduces the granularity of the data.

2. Use a mixed Naive Bayes model: These models can handle both numerical and categorical data by making different assumptions for different types of features. For instance, numerical features could be modelled using a Gaussian distribution while categorical features could be modelled using a multinomial or categorical distribution. You might need to look for a different library or implement it yourself.

3. Use another type of model: Some machine learning models can naturally handle mixed data types. Decision trees and their ensemble variants (like random forests and gradient boosted trees) are capable of handling both numerical and categorical features without requiring any explicit feature transformation.

Bernoulli Naïve Bayes

23 June 2023 15:12

BernoulliNB implements the naive Bayes training and classification algorithms for data that is distributed according to multivariate Bernoulli distributions; i.e., there may be multiple features but each one is assumed to be a binary-valued (Bernoulli, Boolean) variable.

Therefore, this class requires samples to be represented as binary-valued feature vectors; if handed any other kind of data, a BernoulliNB instance may binarize its input (depending on the binarize parameter).

Table 13.1: Data for parameter estimation examples.

	docID	words in document	in $c = \text{China}$?
training set	1	Chinese Beijing Chinese	yes
	2	Chinese Chinese Shanghai	yes
	3	Chinese Macao	yes
	4	Tokyo Japan Chinese	no
test set	5	Chinese Chinese Chinese Tokyo Japan	?

Gaussian Naïve Bayes

23 June 2023 18:04

Data → all features are numerical ←

cgpa | iq | placement — 1000 students — 500 Y, 500 N

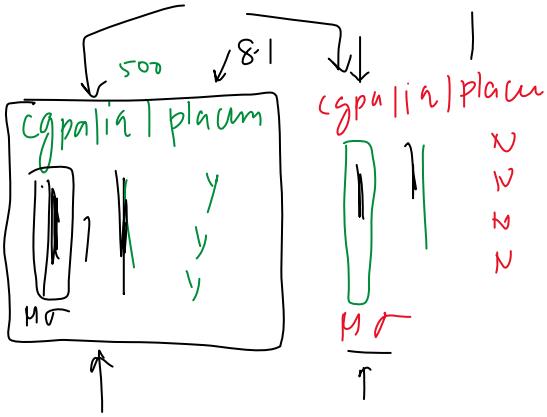
cgpa: num, iq: num

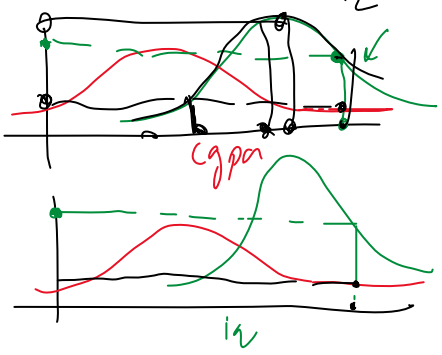
{8.1, 81} → Y / N ?

$P(Y \mid cgpa=8.1, iq=81) = P(Y)\, P(cgpa=8.1 \mid Y)\, P(iq=81 \mid Y)$ — P(Y): prior = 1/2; $P(cgpa=8.1 \mid Y) \to 0$; $P(iq=81 \mid Y) \to 0$

$P(N \mid cgpa=8.1, iq=81) = P(N)\, P(cgpa=8.1 \mid N)\, P(iq=81 \mid N)$ — P(N) = 1/2; $P(cgpa=8.1 \mid N) = 1/2$; $P(iq=81 \mid N) = 1/2$

cgpa = 3.00012





Question on Gaussian NB

24 June 2023 16:17

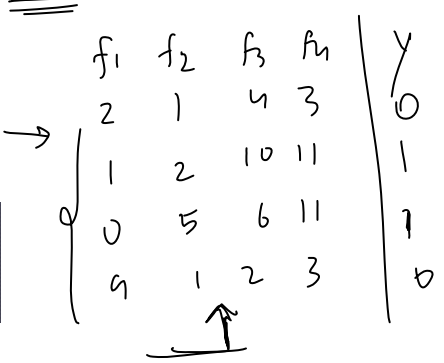
Why Laplace Additive Smoothing not applied on Gaussian Naïve Bayes?

Multinomial Naïve Bayes

23 June 2023 15:08

textual data — Multinomial naive bayes → discrete

Multinomial Naive Bayes is a variant of the Naive Bayes algorithm that is particularly suited for classification tasks involving discrete features, such as text classification where features correspond to word counts or frequencies within the documents.



`P(w|c) = (T_{c,w} + 1) / (text_c + B)`

Here, `T_{c,w}` is the count of word `w` in class `c`, `text_c` is the total count of words in class `c`, and `B` is the size of the vocabulary.

Table 13.1: Data for parameter estimation examples.

	docID	words in document	in c = China?
training set	1	Chinese Beijing Chinese	yes
	2	Chinese Chinese Shanghai	yes
	3	Chinese Macao	yes
	4	Tokyo Japan Chinese	no
test set	5	Chinese Chinese Chinese Tokyo Japan	?

BOW → binary / non-binar (count)

movies review | sentiment

→ great movie | 1

fun — | 0

— — | 1

→ BOW — discrete

great fairy epic

→ 2(1) 0 0

0 3(1) 2(1)

1 1 1

count BOW → multinomial Naive Bayes → fractions → Tf-idf

B = nα

α

`P(w|c) = (T_{c,w} + 1) / (text_c + B)`

Here, `T_{c,w}` is the count of word `w` in class `c`, `text_c` is the total count of words in class `c`, and `B` is the size of the vocabulary.

B = nα α = 1

B = n

Table 13.1: Data for parameter estimation examples.

	docID	words in document	in c = China?
training set	1	Chinese Beijing Chinese	yes
	2	Chinese Chinese Shanghai	yes
	3	Chinese Macao	yes
	4	Tokyo Japan Chinese	no
test set	5	Chinese Chinese Chinese Tokyo Japan	?

Y, N — textual BOW — BOW

	1 Chinese	2 Beijing	3 Shanghai	4 Macao	5 Tokyo	6 Japan	
d1	2	1	0	0	0	0	Y
d2	2	0	1	0	0	0	Y
							Y

3/4 1/8 1/8

P(Chinese | Y) 1/8

d1	2	0	1	0	0	0	Y
d2	2	0	1	0	0	0	Y
d3	1	0	0	1	0	0	N
d4	1	0	0	0	1	1	
d5	3	0	0	0	1	1	?

$P(\text{chinese}|Y) = \frac{5}{8} \rightarrow \frac{0}{8} \rightarrow \frac{0}{8}$

$\frac{5+1}{8+6}$

$P(\text{beijing}|Y)^0 = 1$

$\frac{0+\alpha}{8+n\alpha}$ ($\alpha = 1$; n = size of the vocab)

$\frac{0+1}{8+6\times1} = \frac{1}{14}$

$P(Y \mid \text{chinese}=3, \text{bei}=0, \text{sha}=0, \text{mao}=0, \text{tok}=1, \text{jap}=1) =$

$P(\text{chinese}|Y)^3 \; P(\text{beijing}|Y)^0$

$P(N \mid \text{chinese}=3, \text{bei}=0, \text{sha}=0, \text{mao}=0, \text{tok}=1, \text{jap}=1)$

$\Rightarrow 3/4 \quad P(Y) \; P(\text{chinese}|Y)^3 \; P(\text{beijing}|Y)^0 \cdots P(\text{Japan}|Y)^1$ ($P(\text{Japan}|Y)^0$)

$P(\text{chinese}|N) = \frac{1+1}{3+6(1)} = \frac{2}{9}$

$P(N) = \frac{1}{4}$

{ multinomial distribution } ↘

Complement Naïve Bayes

23 June 2023 15:12

Out of Core Naïve Bayes

23 June 2023 15:13