Supervised Learning: Classification with Naïve Bayes Classifier

Dr. Yuzana Win (Nagasaki University, Japan)
Lecturer
Department of Computer Engineering and Information Technology

Lecture Objectives

- To introduce
  - Classification Techniques or Methods
  - What is Naïve Bayes Classifier?
  - How does the Naïve Bayes Classifier works?
  - Pros and Cons

Classification Techniques or Methods

- Naïve Bayes Classifier *
- Support Vector Machines *
- Neural Networks (Deep Learning)
- Decision Tree based Methods
- Rule-based Methods
- K-Nearest Neighbors (kNN) *

Classification Techniques or Methods

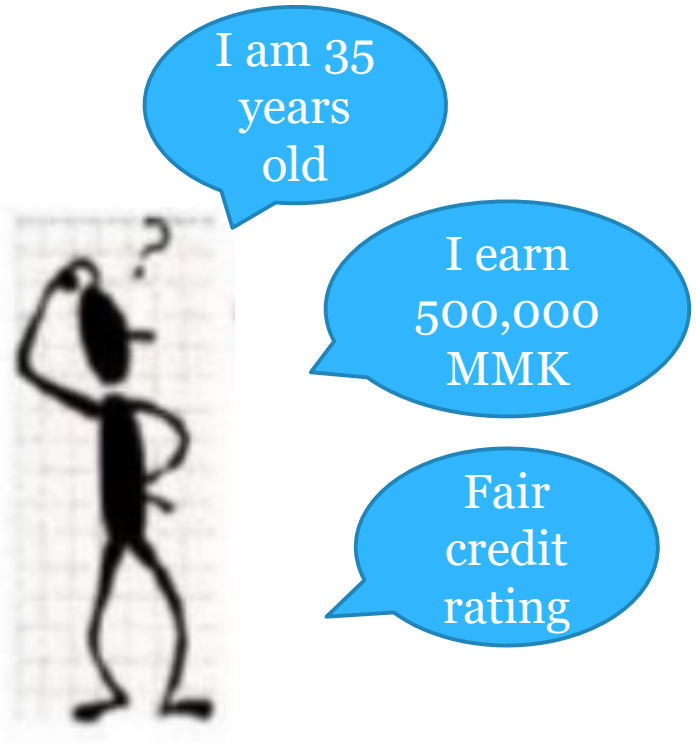
- Naïve Bayes Classifier *
- Support Vector Machines *
- Neural Networks (Deep Learning)
- Decision Tree based Methods
- Rule-based Methods
- K-Nearest Neighbors (KNN) *

Naïve Bayes Classifier

- Supervised Learning Method
- Statistical method for classification
- Probabilistic model under the Bayes theorem
- Solve problems involving both categorical and continuous valued attributes

How Naïve Bayes Classifier Works?

- X : 35 years old customer with an income of 500,000 MMK, and fair credit rating
- H: Hypothesis that the customer will buy a computer



Bayes Theorem

Probability that the customer is 35 years old, have fair credit rate and earns MMK 500000, given that he has bought computer (Posterior Probability of A)

Probability that the customer will buy a computer regardless of age, credit rating, income (Prior Probability of C)

$$P(C \mid A) = \frac{P(A \mid C)P(C)}{P(A)}$$

Probability that the customer will buy a computer given that his age, credit rating and income (Posterior Probability of C)

Probability that a person from 35 years old, have fair credit rate and earns MMK 500000 (Prior Probability of A)

How Naïve Bayes Classifier Works?

- Given:
  - A doctor knows that meningitis causes stiff neck 50% of the time
  - Prior probability of any patient having meningitis is 1/50,000
  - Prior probability of any patient having stiff neck is 1/20

- If a patient has stiff neck, what's the probability he/she has meningitis?

$$P(M \mid S) = \frac{P(S \mid M)P(M)}{P(S)} = \frac{0.5 \times 1/50000}{1/20} = 0.0002$$

Naïve Bayes Classifier

- A: Set of tuples
  - Each Tuple is an 'n' dimensional attribute vector
  - X: $(x_1, x_2, x_3, \ldots\ldots, x_n)$
    - Where x_i is the value of attribute A_i
- Let there are 'm' Classes:
  - $C_1, C_2, C_3, \ldots, C_m$
- Bayesian classifier predicts X belongs to Class Ci if
  - $P(C_i \mid X) > P(C_j \mid X)$ for $1 <= j <= m, j \neq i$
- Maximum Posterior Hypothesis
  - $P(C_i \mid X) = \dfrac{P(X \mid C_i)\, P(C_i)}{P(X)}$
  - Maximize $P(X \mid C_i)\, P(C_i)$ as $P(X)$ is constant

Naïve Bayes Classifier

- With many attributes, it is computationally expensive to Calculate $P(X|C_i)$

- Naïve Assumption of "class conditional independence"

$$P(X \mid C_i) = P(x_1, x_2, \ldots, x_n \mid C_i)$$

$$= P(x_1 \mid C_i) * P(x_2 \mid C_i) * \ldots\ldots\ldots * P(x_n \mid C_i)$$

$$= \prod_{k=1}^{n} P(x_k \mid C_i)$$

Naïve Bayes Classifier

- To calculate, $P(x_k|C_i)$
- A_k => categorical:

$$P(x_k|C_i) = \frac{\text{the number of tuples of class } C_i \text{ in } D \text{ having the value } x_k \text{ for } A_k}{\text{the number of tuples of class } C_i \text{ in } D.}$$

- A_k => contiuous:

A continuous-valued attribute is typically assumed to have a Gaussian distribution with a mean μ and standard deviation σ

$$g(x, \mu, \sigma) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}},$$

$$P(x_k|C_i) = g(x_k, \mu_{C_i}, \sigma_{C_i}).$$

How to Solve "ZERO" Problem?

- If there is a Class, C_i and X has an attribute value x_k, such that none of the samples in C_i has that attribute value, how to solve this problem?

- In that case, we solve the probability estimation equation as below

$$\text{Original} : P(A_i \mid C) = \frac{N_{ic}}{N_c}$$

$$\text{Laplace} : P(A_i \mid C) = \frac{N_{ic} + 1}{N_c + c}$$

$$\text{m-estimate} : P(A_i \mid C) = \frac{N_{ic} + mp}{N_c + m}$$

c: number of classes

p: prior probability

m: parameter

CEIT
Think smart, be different.

Example of Naïve Bayes Classifier I

Name	Give Birth	Can Fly	Live in Water	Have Legs	Class
human	yes	no	no	yes	mammals
python	no	no	no	no	non-mammals
salmon	no	no	yes	no	non-mammals
whale	yes	no	yes	no	mammals
frog	no	no	sometimes	yes	non-mammals
komodo	no	no	no	yes	non-mammals
bat	yes	yes	no	yes	mammals
pigeon	no	yes	no	yes	non-mammals
cat	yes	no	no	yes	mammals
leopard shark	yes	no	yes	no	non-mammals
turtle	no	no	sometimes	yes	non-mammals
penguin	no	no	sometimes	yes	non-mammals
porcupine	yes	no	no	yes	mammals
eel	no	no	yes	no	non-mammals
salamander	no	no	sometimes	yes	non-mammals
gila monster	no	no	no	yes	non-mammals
platypus	no	no	no	yes	mammals
owl	no	yes	no	yes	non-mammals
dolphin	yes	no	yes	no	mammals
eagle	no	yes	no	yes	non-mammals

A: attributes

M: mammals

N: non-mammals

$$P(A \mid M) = \frac{6}{7} \times \frac{6}{7} \times \frac{2}{7} \times \frac{2}{7} = 0.06$$

$$P(A \mid N) = \frac{1}{13} \times \frac{10}{13} \times \frac{3}{13} \times \frac{4}{13} = 0.0042$$

$$P(A \mid M)P(M) = 0.06 \times \frac{7}{20} = 0.021$$

$$P(A \mid N)P(N) = 0.004 \times \frac{13}{20} = 0.0027$$

Give Birth	Can Fly	Live in Water	Have Legs	Class
yes	no	yes	no	?

P(A|M)P(M) > P(A|N)P(N)

=> Mammals

CEIT
Think smart, be different.

Example of Naïve Bayes Classifier II

Outlook	Temp	Humidity	Windy	Play
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

The Evidence relates all attributes without Exceptions.

Outlook	Temp.	Humidity	Windy	Play
Sunny	Cool	High	True	?

Evidence E

$$\Pr[yes \mid E] = \Pr[Outlook = Sunny \mid yes]$$
$$\times \Pr[Temperature = Cool \mid yes$$
$$\times \Pr[Humidity = High \mid yes]$$
$$\times \Pr[Windy = True \mid yes]$$
$$\times \frac{\Pr[yes]}{\Pr[E]}$$
$$= \frac{\frac{2}{9} \times \frac{3}{9} \times \frac{3}{9} \times \frac{3}{9} \times \frac{9}{14}}{\Pr[E]}$$

Probability of class "yes"

Outlook			Temperature			Humidity			Windy			Play	
	Yes	*No*		*Yes*	*No*		*Yes*	*No*		*Yes*	*No*	*Yes*	*No*
Sunny	**2**	**3**	**Hot**	**2**	**2**	**High**	**3**	**4**	**False**	**6**	**2**	**9**	**5**
Overcast	**4**	**0**	**Mild**	**4**	**2**	**Normal**	**6**	**1**	**True**	**3**	**3**		
Rainy	**3**	**2**	**Cool**	**3**	**1**								

Sunny	2/9	3/5	Hot	2/9	2/5	High	3/9	4/5	False	6/9	2/5	9/14	5/14
Overcast	4/9	0/5	Mild	4/9	2/5	Normal	6/9	1/5	True	3/9	3/5		
Rainy	3/9	2/5	Cool	3/9	1/5								

Outlook	Temp	Humidity	Windy	Play
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

Sunny	2/9	3/5	Hot	2/9	2/5	High	3/9	4/5	False	6/9	2/5	9/14	5/14
Overcast	4/9	0/5	Mild	4/9	2/5	Normal	6/9	1/5	True	3/9	3/5		
Rainy	3/9	2/5	Cool	3/9	1/5								

For compute prediction for new day:

Outlook	Temp.	Humidity	Windy	Play
Sunny	Cool	High	True	?

Likelihood of the two classes

For "yes" = 2/9 × 3/9 × 3/9 × 3/9 × 9/14 = 0.0053

For "no" = 3/5 × 1/5 × 4/5 × 3/5 × 5/14 = 0.0206

Conversion into a probability by normalization:

P("yes") = 0.0053 / (0.0053 + 0.0206) = **0.205**

P("no") = 0.0206 / (0.0053 + 0.0206) = 0.795

Example of Naïve Bayes Classifier III

Training dataset

age	income	student	credit_rating	buys_computer
<=30	high	no	fair	no
<=30	high	no	excellent	no
30…40	high	no	fair	yes
>40	medium	no	fair	yes
>40	low	yes	fair	yes
>40	low	yes	excellent	no
31…40	low	yes	excellent	yes
<=30	medium	no	fair	no
<=30	low	yes	fair	yes
>40	medium	yes	fair	yes
<=30	medium	yes	excellent	yes
31…40	medium	no	excellent	yes
31…40	high	yes	fair	yes
>40	medium	no	excellent	no

P(buys_computer=„yes“)=9/14

P(buys_computer=„no“)=5/14

Class:
C1:buys_computer= ‘yes’
C2:buys_computer= ‘no’

Data sample
X =(age<=30,
Income=medium,
Student=yes
Credit_rating=
Fair)

Example of Naïve Bayes Classifier III

- Compute $P(X|C_i)$ for each class

P(age="<30" | buys_computer="yes") = 2/9=0.222
P(age="<30" | buys_computer="no") = 3/5 =0.6
P(income="medium" | buys_computer="yes")= 4/9 =0.444
P(income="medium" | buys_computer="no") = 2/5 = 0.4
P(student="yes" | buys_computer="yes)= 6/9 =0.667
P(student="yes" | buys_computer="no")= 1/5=0.2
P(credit_rating="fair" | buys_computer="yes")=6/9=0.667
P(credit_rating="fair" | buys_computer="no")=2/5=0.4

- X=(age<=30 ,income =medium, student=yes,credit_rating=fair)

$P(X|C_i)$: P(X|buys_computer="yes")= 0.222 x 0.444 x 0.667 x 0.0.667 =0.044
P(X|buys_computer="no")= 0.6 x 0.4 x 0.2 x 0.4 =0.019

$P(X|C_i)*P(C_i)$: P(X|buys_computer="yes") * P(buys_computer="yes")=**0.028**
P(X|buys_computer="no") * P(buys_computer="no")=**0.007**

- X belongs to class "buys_computer=yes"

Application Area of Naïve Bayes Classifier

- Text Classification
- Spam Filtering
- Hybrid Recommender System
- Online Application
  - Cyberbullying Detection System
  - Simple Emotion Modeling

Text Classification with Naïve Bayes Classifier

- Naïve Bayes classification method based on Bayes rule
- Relies on very simple representation of document
  - Bag of words

The bag of words representation

I love this movie! It's sweet, but with satirical humor. The dialogue is great and the adventure scenes are fun… It manages to be whimsical and romantic while laughing at the conventions of the fairy tale genre. I would recommend it to just about anyone. I've seen it several times, and I'm always happy to see it again whenever I have a friend who hasn't seen it yet.

CEIT
Think smart, be different.

The bag of words representation

I **love** this movie! It's **sweet**, but with **satirical** humor. The dialogue is **great** and the adventure scenes are **fun**… It manages to be **whimsical** and **romantic** while **laughing** at the conventions of the fairy tale genre. I would **recommend** it to just about anyone. I've seen it **several** times, and I'm always **happy** to see it **again** whenever I have a friend who hasn't seen it yet.

The bag of words representation using a subset of words

x **love** xxxxxxxxxxxxxxxx **sweet**
xxxxxxx **satirical** xxxxxxxxxx
xxxxxxxxxxxx **great** xxxxxxx
xxxxxxxxxxxxxxxxxxxx **fun** xxxx
xxxxxxxxxxxxxx **whimsical** xxxx
romantic xxxx **laughing**
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
xxxxxxxxxxxxxxx **recommend** xxxxx
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
xx **several** xxxxxxxxxxxxxxxxx
xxxxx **happy** xxxxxxxxx **again**
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
xxxxxxxxxxxxxxxxx

The bag of words representation

great	2
love	2
recommend	1
laugh	1
happy	1
...	...

Text Classification with Naïve Bayes Classifier

- **Naïve Bayes Text Classifier Algorithm** is used which is a probabilistic model.

$$c = \arg\max \Pr(C = c_j) \prod^{|A|} \Pr(A_i = a_i \mid C = c_j)$$

$$\Pr(A_i = a_i \mid C = c_j) = \frac{n_{ij} + \lambda}{n_j + \lambda m_i}$$

- Let A be the attributes of the training dataset and C be the class attribute
- n_{ij} be the number of examples that have both $A_i = a_i$ and $C = c_j$
- n_j be the total number of examples with $C=c_j$ in the training data set
- m_i is the number of values of attribute A_i
- λ is a multiplicative factor

Text Classification with Naïve Bayes Classifier

- Input:
  - a document *d* (Cyber-bullied terms document)
  - classes C = {c1, c2} => bully or non-bully
  - A training set of m hand-labeled documents (d1,c1),....,(dm, cm)
- Output:
  - a learned classifier γ:d → c

Doc	Text	Class
1.	['not', 'lazi', 'jobless', 'just', ‘f**k', 'poor']	-
2.	['come', 'speed', 'good', 'movi']	+
3.	['dumbass', 'inde']	-
4.	['nah', 'neither', 'good']	+
5.	['agre', 'movi', ‘s**t', 'act']	-

Unique words 17
<not, lazi, jobless, just, f**k, poor, come, speed, good, movi, dumbass, inde, nah, neither, agre, s**t, act>

Doc	not	lazi	jobless	just	f**k	poor	come	speed	good	movi	dumbass	inde	nah	neither	agre	s**t	act	Class
1.	1	1	1	1	1	1												-
2.							1	1	1	1								+
3.											1	1						-
4.									1				1	1				+
5.										1					1	1	1	-

The probabilities are calculated using **naïve bayes** algorithm

Doc	not	lazi	jobless	just	f**k	poor	come	speed	good	movi	dumbass	inde	nah	neither	agre	s**t	act	Class
2.							1	1	1	1								+
4.									1				1	1				+

P(+) = 2/5 = 0.4

P(not|+) = 0+1/ 7+17 = 0.0416
P(lazi+) = 0+1/ 7+17 = 0.0416
P(jobless|+) = 0+1/ 7+17 = 0.0416
P(just|+) = 0+1/ 7+17 = 0.0416
P(f**k|+) = 0+1/ 7+17 = 0.0416
P(poor|+) = 0+1/ 7+17 = 0.0416
P(come|+) = 1+1/ 7+17 = 0.0833
P(speed|+) = 1+1/ 7+17 = 0.0833
P(good|+) = 2+1/ 7+17 = 0.125
P(movi|+) = 1+1/ 7+17 = 0.0833

P(dumbass|+) = 0+1/ 7+17 = 0.0416
P(inde|+) = 0+1/ 7+17 = 0.0416
P(nah|+) = 1+1/ 7+17 = 0.0833
P(neither|+) = 1+1/ 7+17 = 0.0833
P(agre|+) = 0+1/ 7+17 = 0.0416
P(s**t|+) = 0+1/ 7+17 = 0.0416
P(act|+) = 0+1/ 7+17 = 0.0416

Doc	not	lazi	jobless	just	f**k	poor	come	speed	good	movi	dumbass	inde	nah	neither	agre	s**t	act	Class
1.	1	1	1	1	1	1												-
3.											1	1						-
5.										1					1	1	1	-

P(-) = 3/5 = 0.6

P(not|-) = 1+1/ 12+17 = 0.0689
P(lazi-) = 1+1/ 12+17 = 0.0689
P(jobless|-) = 1+1/ 12+17 = 0.0689
P(just|-) = 1+1/ 12+17 = 0.0689
P(f**k|-) = 1+1/ 12+17 = 0.0689
P(poor|-) = 1+1/ 12+17 = 0.0689
P(come|-) = 0+1/ 12+17 = 0.0344
P(speed|-) = 0+1/ 12+17 = 0.0344
P(good|-) = 0+1/ 12+17 = 0.0344
P(movi|-) = 1+1/ 12+17 = 0.0689

P(dumbass|-) = 1+1/ 12+17 = 0.0689
P(inde|-) = 1+1/ 12+17 = 0.0689
P(nah|-) = 0+1/ 12+17 = 0.0344
P(neither|-) = 0+1/ 12+17 = 0.0344
P(agre|-) = 1+1/ 12+17 = 0.0689
P(s**t|-) = 1+1/ 12+17 = 0.0689
P(act|-) = 1+1/ 12+17 = 0.0689

Let's classify a new sentence according to:

$$V_{NB} = \underset{v_j \in V}{\operatorname{argmax}} P(v_j) \prod_{w \in words} P(w|v_j)$$

Where V stands for "class"

Example Sentence : "dumbass inde"

V = +; P(+) P(dumbass|+) P(inde|+)

=(0.4) (0.0416)(0.0416)

=6.92224×10^{-4}

V= - ; P(-) P(dumbass|-) P(inde|-)

=(0.6)(0.0689) (0.0689)

=2.848326×10^{-3}

Greater

Therefore, the sentence is negative(contained bullied word).

Naïve Bayes Coding

```
train = [('not lazi jobless just fuck poor','neg'),
         ('come speed good movi','pos'),
         ('dumbass inde','neg'),
         ('nah neither good','pos'),
         ('agre movi shit act','neg'),
         ('yeah justin stop fuck forum troll','neg'),
         ('yeah like less blu ray version bought','pos'),
         ('yeah lot problem associ pretti much know elimin peopl think thatll never work','pos')]

from textblob.classifiers import NaiveBayesClassifier
cl = NaiveBayesClassifier(train)

a=cl.classify("come speed good movi")
print ("The input sentence has " + a + " sense")

b=cl.classify("jobless life give shit")
print ("The input sentence has " + b + " sense")

c=cl.classify("lazi peopl usually think good problem never work")
print ("The input sentence has " + c + " sense")
```

Output

```
======== RESTART: I:\My data\BE thesis\Step 2\Testing\Nb Testing\4.py ========
The input sentence has pos sense
The input sentence has neg sense
The input sentence has pos sense
>>>
```

Advantages and Disadvantages of Naïve Bayes Classifier

- Advantages :
  - Easy to implement.
  - Good results obtained in most of the cases
- Disadvantages
  - Class conditional independence assumption, therefore loss of accuracy
  - Practically, dependencies exist among variables
    - E.g., hospital patients' profile: age, family history etc.,
    - Symptoms: fever, cough etc., Disease: lung cancer, diabetes etc.,

Conclusion

- Handle missing values by ignoring the instance during probability estimate calculations
- Robust to irrelevant attributes
- Widely used in text classification and spam filtering

Next Week Lecture

- Supervised Learning: Classification with Decision Tree

Thank You