Concrete Machine Learning

Deep User : 2020 Summer Program

Naïve Bayes Classifier

- Clustering algorithm
- Supervised/Unsupersvised learning
- Strong & Simple

- Labeling process
- Bayesian probability theory base

$$\underbrace{P(c \mid x)}_{\text{Posterior Probability}} = \frac{\overbrace{P(x \mid c)}^{\text{Likelihood}}\,\overbrace{P(c)}^{\text{Class Prior Probability}}}{\underbrace{P(x)}_{\text{Predictor Prior Probability}}}$$

A | Naïve Bayes Classifier

Main Purpose

Proceed with classification of objects as the group with the greatest **posterior probability**

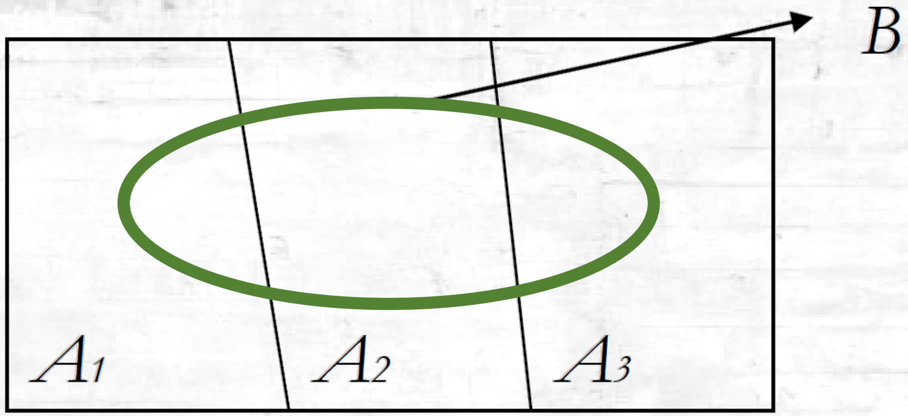
$$\text{posterior} = \frac{\text{prior} \times \text{likelihood}}{\text{evidence}}.$$

* posterior : 사후 확률, prior : 사전확률, likelihood : 우도, evidence : 관찰값

$$\hat{y} = \underset{k\in\{1,\dots,K\}}{\operatorname{argmax}}\ p(C_k)\prod_{i=1}^{n} p(x_i|C_k).$$

Assign classes with the highest post probability for k classes

A | Bayes Rule

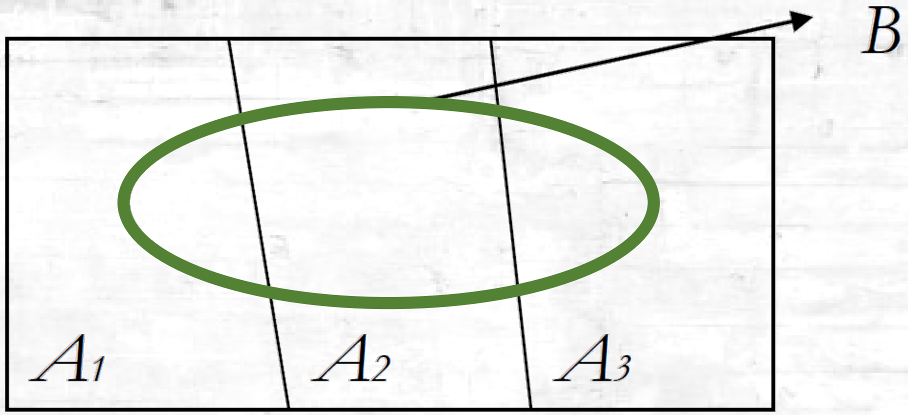


$$P(B) = P(A_1 \cap B) + P(A_2 \cap B) + P(A_3 \cap B)$$

$$P(B) = P(A_1)P(B|A_1) + P(A_2)P(B|A_2) + P(A_3)P(B|A_3) = \sum_{i=1}^{3} P(A_i)P(B|A_i)$$

Law of Total Probability

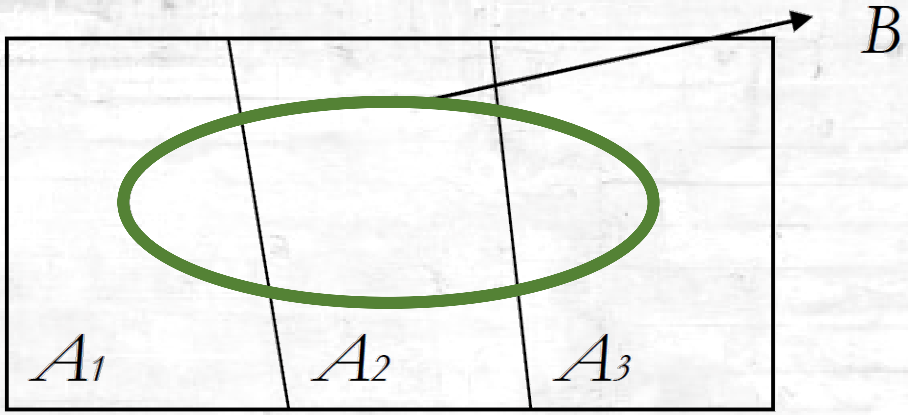
A | Bayes Rule



$P(A_1), P(A_2), P(A_3)$ 사전확률(prior probability)

$P(B|A1), P(B|A2), P(B|A3)$ 우도(likelihood probability)

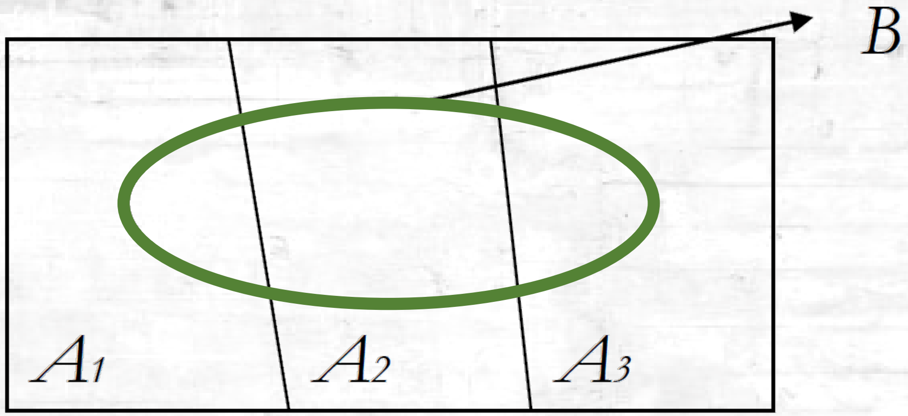
A | Bayes Rule



$$P(A_1|B) = \frac{P(A_1)P(B|A_1)}{P(B)}$$

$$= \frac{P(A_1)P(B|A_1)}{P(A_1)P(B|A_1) + P(A_2)P(B|A_2) + P(A_3)P(B|A_3)}$$

A | Bayes Rule



$P(A_1|B)$ 사후확률(posterior probability)

Posterior probability is an updated version of prior probability

A | Bayes Rule

: 쿠키가 들어 있는 그릇 두 개가 있다

첫번째 그릇에는 바닐라 쿠키 30개와 초콜렛 쿠키 10개가 들어있고, 두번째 그릇에는 두 가지 쿠키가 종류별로 20개씩 들어 있을 때

어떤 그릇인지 보지 않고 한 그릇에서 임의로 쿠키를 집었는데 바닐라 쿠키라면 이 때
'이 바닐라 쿠키가 그릇 1에서 나왔을 가능성'은 얼마일까요?

A | Bayes Rule

: 쿠키가 들어 있는 그릇 두 개가 있다

첫번째 그릇에는 바닐라 쿠키 30개와 초콜렛 쿠키 10개가 들어있고, 두번째 그릇에는 두 가지 쿠키가 종류별로 20개씩 들어 있을 때

어떤 그릇인지 보지 않고 한 그릇에서 임의로 쿠키를 집었는데 바닐라 쿠키라면 이 때 '이 바닐라 쿠키가 그릇 1에서 나왔을 가능성'은 얼마일까요?

$$P(H|D) = \frac{P(H)P(D|H)}{P(D)} = \frac{\frac{1}{2} \times \frac{3}{4}}{\frac{5}{8}} = \frac{3}{5}$$

A | # Bayes Rule

: 쿠키가 들어 있는 그릇 두 개가 있다

첫번째 그릇에는 바닐라 쿠키 30개와 초콜렛 쿠키 10개가 들어있고, 두번째 그릇에는 두 가지 쿠키가 종류별로 20개씩 들어 있을 때

어떤 그릇인지 보지 않고 한 그릇에서 임의로 쿠키를 집었는데 바닐라 쿠키라면
이 때 '이 바닐라 쿠키가 그릇 1에서 나왔을 가능성'은 얼마인가?

항목	가설1(그릇1)	가설2(그릇2)
사전확률 $P(H)$	1/2	1/2
우도 $P(D\|H)$	3/4	1/2
사전확률 × 우도	3/8	1/4
한정상수 $P(D)$	5/8	5/8
사후확률 $P(H\|D)$	3/5	2/5

Document Classifier

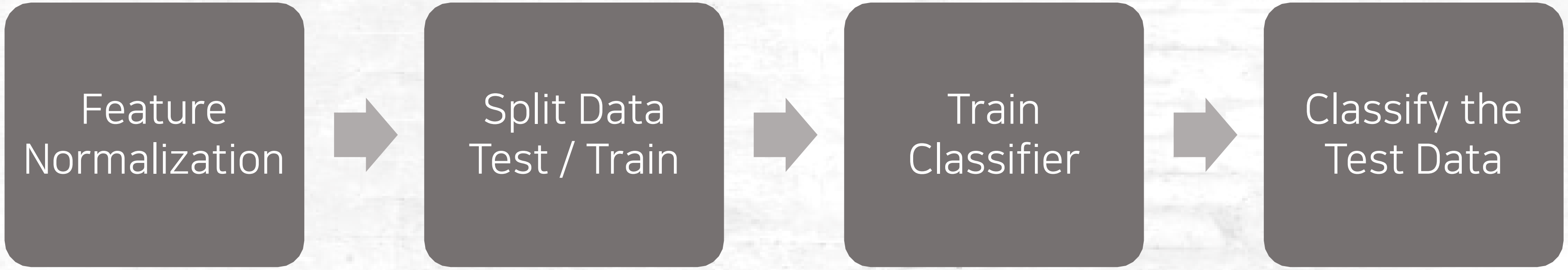
$$P(c_1|d) = \frac{P(c_1, d)}{P(d)} = \frac{\frac{P(c_1,d)}{P(c_1)} \cdot P(c_1)}{P(d)} = \frac{P(d|c_1)P(c_1)}{P(d)}$$

$$P(c_2|d) = \frac{P(d|c_2)P(c_2)}{P(d)}$$

$$P(c_i|d) \propto P(d|c_i)P(c_i)$$

$$\begin{aligned} P(c_i|d) &= P(c_i|w_1, w_2) \\ &\propto P(w_1, w_2|c_i)P(c_i) \\ &\propto P(w_1, w_2|c_i) \end{aligned}$$

A | Naïve Bayes Classifier



A | Naïve Bayes Classifier

Normalization Formula

$$X_{normalized} = \frac{(X - X_{minimum})}{(X_{maximum} - X_{minimum})}$$

Numpy.sum

Numpy.mean

Numpy.std

A | Naïve Bayes Classifier

Calculate the mean and standard deviation of train data each for label 1, label 2, label 3, feature 1, feature2, feature 3, feature4

the example of mean matrix of train data

	Feature1	**Feature2**	**Feature3**	**Feature4**
Label1	Mean	Mean	Mean	Mean
Label2	Mean	Mean	Mean	Mean
Label3	Mean	Mean	Mean	mean

A | Naïve Bayes Classifier

- Estimate the probability of feature vector each for class 1, class 2, class3
- Use Naive Bayesian theorem
  - $p(C_k|x) = \frac{p(C_k)p(x|C_k)}{p(x)}$
  - $p(x|C_k)$ = k_{th} - feature (observation)
  - $p(x)$ = normalization factor ($\sum_{k=1}^{class\_num} p(C_k)p(x|C_k)$) (Never mind)
  - $p(C_k)$ = initial probability of class
- Use chain rule
  - $p(C_k)p(x_{1\ and}\ x_{2\ and}\ x_3|C_k) = p(C_k)p(x_1|C_k)p(x_2|C_k)p(x_3|C_k)$
- Use log scale
  - $\ln(p(C_k)p(x_1|C_k)p(x_2|C_k)p(x_3|C_k)) = \ln(p(C_k)) + \ln(p(x_1|C_k)) + \ln(p(x_2|C_k)) + \ln(p(x_3|C_k))$

A | Naïve Bayes Classifier

ESTIMATION OF PROBABILITY DISTRIBUTION

	Probability of class1	Probability of class2	Probability of class3
Feature vector1	$\ln(p(C_1)p(x1\|C_1)p(x2\|C_1)p(x3\|C_1)p(x4\|C_1))$	$\ln(p(C_2)p(x1\|C_2)p(x2\|C_2)p(x3\|C_2)p(x4\|C_2))$	$\ln(p(C_3)p(x1\|C_3)p(x2\|C_3)p(x3\|C_3)p(x4\|C_3))$
Feature vector2	$\ln(p(C_1)p(x1\|C_1)p(x2\|C_1)p(x3\|C_1)p(x4\|C_1))$	$\ln(p(C_2)p(x1\|C_2)p(x2\|C_2)p(x3\|C_2)p(x4\|C_2))$	$\ln(p(C_3)p(x1\|C_3)p(x2\|C_3)p(x3\|C_3)p(x4\|C_3))$
Feature vector3	$\ln(p(C_1)p(x1\|C_1)p(x2\|C_1)p(x3\|C_1)p(x4\|C_1))$	$\ln(p(C_2)p(x1\|C_2)p(x2\|C_2)p(x3\|C_2)p(x4\|C_2))$	$\ln(p(C_3)p(x1\|C_3)p(x2\|C_3)p(x3\|C_3)p(x4\|C_3))$
Feature vector4	$\ln(p(C_1)p(x1\|C_1)p(x2\|C_1)p(x3\|C_1)p(x4\|C_1))$	$\ln(p(C_2)p(x1\|C_2)p(x2\|C_2)p(x3\|C_2)p(x4\|C_2))$	$\ln(p(C_3)p(x1\|C_3)p(x2\|C_3)p(x3\|C_3)p(x4\|C_3))$

A | Naïve Bayes Classifier

ESTIMATION OF PROBABILITY DISTRIBUTION

	Probability of class1	Probability of class2	Probability of class3
Feature vector1	$\ln(p(C_1)p(x1\|C_1)p(x2\|C_1)p(x3\|C_1)p(x4\|C_1))$	$\ln(p(C_2)p(x1\|C_2)p(x2\|C_2)p(x3\|C_2)p(x4\|C_2))$	$\ln(p(C_3)p(x1\|C_3)p(x2\|C_3)p(x3\|C_3)p(x4\|C_3))$
Feature vector2	$\ln(p(C_1)p(x1\|C_1)p(x2\|C_1)p(x3\|C_1)p(x4\|C_1))$	$\ln(p(C_2)p(x1\|C_2)p(x2\|C_2)p(x3\|C_2)p(x4\|C_2))$	$\ln(p(C_3)p(x1\|C_3)p(x2\|C_3)p(x3\|C_3)p(x4\|C_3))$
Feature vector3	$\ln(p(C_1)p(x1\|C_1)p(x2\|C_1)p(x3\|C_1)p(x4\|C_1))$	$\ln(p(C_2)p(x1\|C_2)p(x2\|C_2)p(x3\|C_2)p(x4\|C_2))$	$\ln(p(C_3)p(x1\|C_3)p(x2\|C_3)p(x3\|C_3)p(x4\|C_3))$
Feature vector4	$\ln(p(C_1)p(x1\|C_1)p(x2\|C_1)p(x3\|C_1)p(x4\|C_1))$	$\ln(p(C_2)p(x1\|C_2)p(x2\|C_2)p(x3\|C_2)p(x4\|C_2))$	$\ln(p(C_3)p(x1\|C_3)p(x2\|C_3)p(x3\|C_3)p(x4\|C_3))$

	Estimation class
Feature vector1	1
Feature vector2	2
Feature vector3	3
Feature vector4	1

A | Naïve Bayes Classifier

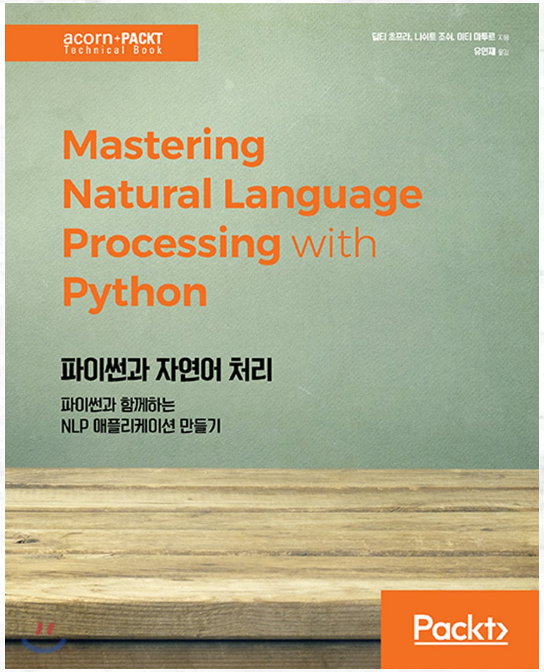
After build all of function, you can see below result from python console when you compile “main_app.py” (or yielded 90% accuracy due to shuffling data)

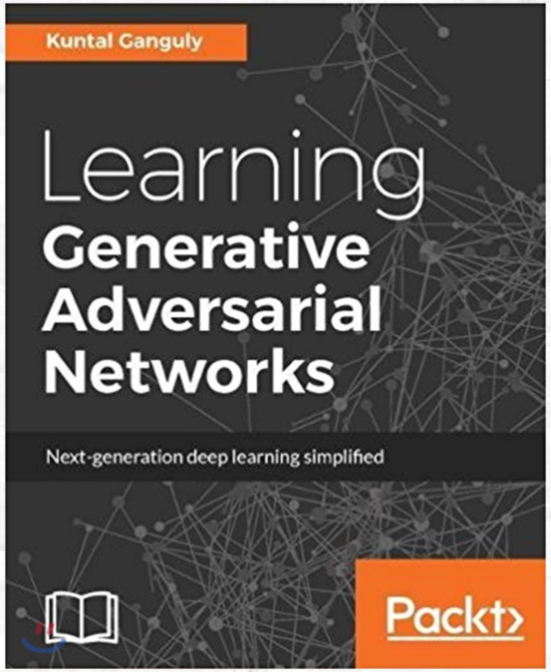
```
accuracy is 97.95918367346938% ! !
the number of correct data is 48 of 49 ! !

In [22]:
```

A | Books


코딩셰프의
3분 딥러닝
케라스맛
김성진 지음
Keras 코드로 맛보는
ANN, DNN, CNN,
RNN, AE, GAN, UNET
한빛미디어


텐서플로로 살펴보는
Q 러닝, MDP, DQN, A3C
강화학습 알고리즘
강화학습
첫걸음
한빛미디어


acorn+PACKT
Technical Book
Mastering
Natural Language
Processing with
Python
파이썬과 자연어 처리
파이썬과 함께하는
NLP 애플리케이션 만들기
Packt>


Kuntal Ganguly
Learning
Generative
Adversarial
Networks
Next-generation deep learning simplified
Packt>

DeepUser

Naïve Bayes Classifier

THANKS