Machine Learning
Probabilistic Learning

DR. BHARGAVI R

SCOPE

VIT CHENNAI

Bayes Theorem

- Given the Hypotheses space and Data, we want to find the **Best Hypothesis** .
- Bayes theorem provides a way of calculating the probability of a hypothesis based on its prior probability, probabilities of observing various data given the hypothesis, and the observed data.
- Let P(h) be the initial probability that the hypothesis h holds before the data is observed(training data).
- P(h) is called the prior probability of h.
- If no such prior knowledge is available then equal prior probability to all candidate hypotheses.
- Let P(D) be the prior probability that the training data D will be observed given no knowledge about the which hypothesis holds.

Bayes Theorem (cont…)

- Let P(D|h) be the probability of observing the data D given that h holds.
- p(h|D) is the **Posterior probability of h** – indicates probability that h holds given that the data D is observed.
- Bayes Theorem $p(h|D) = \frac{p(D|h)p(h)}{p(D)}$
- In ML, we have some set of candidate hypotheses H and our interest is in finding the most probable hypothesis $h \in H$ given the observed data D.
- A maximally probable hypothesis is called a maximum a posteriori (MAP) hypothesis (denoted as h_{MAP}).

Maximum A Posteriori Hypothesis

- h_{MAP} is obtained using Bayes Theorem as :

$$= \underset{h \in H}{\operatorname{argmax}} \frac{p(D|h)p(h)}{p(D)}$$

$$h_{MAP} = \underset{h \in H}{\operatorname{argmax}} p(D|h)p(h)$$

- P(D) is dropped since it is constant and independent of h.
- In some cases, we will assume that every hypothesis in H is equally probable.

So the equation further simplified to have just the term p(D|h)

Maximum A Posteriori Hypothesis (cont…)

- P(D|h) is called the likelihood of the data D given h, and any hypothesis that maximizes P(D|h) is called a maximum likelihood (ML) hypothesis, h_{ML}
- $$h_{ML} = \underset{h \in H}{\operatorname{argmax}}\, p(D \mid h)$$

Example

Consider a medical diagnosis problem in which there are two alternative hypotheses: (1) that the patient has a particular form of cancer. and (2) that the patient does not. The available data is from a particular laboratory test with two possible outcomes: + (positive) and - (negative). We have prior knowledge that over the entire population of people only 0.008 have this disease.

Furthermore, the lab test is only an imperfect indicator of the disease. The test returns a correct positive result in only 98% of the cases in which the disease is actually present and a correct negative result in only 97% of the cases in which the disease is not present. In other cases, the test returns the opposite result.

Now, suppose a new patient for whom the lab test returns a positive result. What is the probability of that patient having cancer or no cancer?

Example (cont…)

$p(cancer) = 0.008$	$p(\neg cancer) = 0.992$
$p(+\|cancer) = 0.98$	$p(-\|cancer) = 0.02$
$p(+\|\neg cancer) = 0.03$	$p(-\|\neg cancer) = 0.97$

Solution : Using Bayes Theorem

$p(cancer|+) = p(+|cancer) * p(cancer)$ = 0.98 * 0.008 = 0.0078

$p(\neg cancer|+) = p(+|\neg cancer) * p(\neg cancer)$ = 0.03 * 0.992 = 0.0298

Therefore, Maximum a posteriori hypothesis $h_{MAP} = \neg cancer$

Naive Bayesian Classifier

- Naïve Bayes is a probabilistic classifier based on Bayes theorem.
- Suitable for high-dimensional data.
- Performs well in many real-world situations, especially in text classification tasks like spam filtering, sentiment analysis, and document categorization.
- Bayes' Theorem provides a way to calculate the posterior probability P(C|X), which is the probability of a class C given the observed data X.
- It is expressed as: $p(C|X) = \frac{p(X|C)p(C)}{p(X)}$

p(C/X): Posterior probability of class C given the data X.

p(X/C) : Likelihood of data X given the class C.

p(C) : Prior probability of class C.

P(X) : Evidence or total probability of the data X.

Naive Bayesian Classifier (cont…)

- Let D be a training set of tuples and their associated class labels, and each tuple is represented by an m dimensional attribute vector **x** = (x_1, x_2, …, x_m)
- Let there be k classes C_1, C_2, …, C_k.
- Classification is to derive the maximum posteriori, i.e., the maximal P(C_i | **x**).
- Naïve Bayes Assumption: Attributes are conditionally independent (i.e., no dependence relation between attributes) given the class label.

$$p(\boldsymbol{x}|c_i) = \prod_{j=1}^{m} p(x_j|c_i) = p(x_1|c_i) \times p(x_2|c_i) \times \cdots \ldots \times p(x_m|c_i)$$

Here c_i represents the i^{th} class.

- To classify a new instance, the classifier computes the posterior probability for each class and predicts the class with the highest probability.

$$C_{Pred} = \underset{c}{\text{argmax}}\ p(C \mid \mathbf{x})$$

Example: Predicting Potential Computer Buyer

ID	Age	Income	Student	Credit Rating	Buy_Status
1	Youth	High	No	Fair	No
2	Youth	High	No	Excellent	No
3	Middle Aged	High	No	Fair	Yes
4	Senior	Medium	No	Fair	Yes
5	Senior	Low	Yes	Fair	Yes
6	Senior	Low	Yes	Excellent	No
7	Middle Aged	Low	Yes	Excellent	Yes
8	Youth	Medium	No	Fair	No
9	Youth	Low	Yes	Fair	Yes
10	Senior	Medium	Yes	Fair	Yes
11	Youth	Medium	Yes	Excellent	Yes
12	Middle Aged	Medium	No	Excellent	Yes
13	Middle Aged	High	Yes	Fair	Yes
14	Senior	Medium	No	Excellent	No

Example (cont…)

- Consider the new test instance $\mathbf{x}_t$ = (age = youth, Income = medium, Student = yes, Credit_rating = Fair).
- Does this person buys the computer?

Solution

Let C_1:buys_computer = yes

C_2:buys_computer = no

$P(C_1)$ = P(buys_computer = yes) = 9/14 = 0.643

$P(C_2)$ = P(buys_computer = no) = 5/14= 0.357

Need to compute $P(C_i|\mathbf{x}_t)$ for the given test instance.

Example (cont…)

$P(\mathbf{x}|C_i)$:

$P(\mathbf{x}|$buys_computer = "yes"$)$ = P(age = "youth" | buys_computer = "yes")
x P(income = "medium" | buys_computer = "yes")
x P(student = "yes" | buys_computer = "yes")
x P(credit_rating = "fair" | buys_computer = "yes")

$P(\mathbf{x}|$buys_computer = "No"$)$ = P(age = "youth" | buys_computer = "No")
x P(income = "medium" | buys_computer = "No")
x P(student = "yes" | buys_computer = "No")
x P(credit_rating = "fair" | buys_computer = "No")

$P(\mathbf{x}|C_i)*P(C_i)$: P(X|buys_computer = "yes") * P(buys_computer = "yes")
P(X|buys_computer = "no") * P(buys_computer = "no")

Example (cont…)

Prior Probabilities :

p(No) = 5/14 p(Yes) = 9/14

Likelihood Computations:

Age	Yes(9)	No(5)
Youth(5)	2/9	3/5
Middle(4)	4/9	0/5
Senior(5)	3/9	2/5

Income	Yes(9)	No(5)
Low(4)	3/9	1/5
Medium(6)	4/9	2/5
High (4)	2/9	2/5

Student	Yes(9)	No(5)
Yes(7)	6/9	1/5
No(7)	3/9	4/5

Credit	Yes(9)	No(5)
Excellent(6)	3/9	3/5
Fair(8)	6/9	2/5

Example (cont…)

P(age = “youth” | buys_computer = “yes”) = 2/9 = 0.222

P(age = “youth” | 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

$P(\mathbf{x}|C_i)$: P(**x**|buys_computer = “yes”) = 0.222 x 0.444 x 0.667 x 0.667 = 0.044

P(**x**|buys_computer = “no”) = 0.6 x 0.4 x 0.2 x 0.4 = 0.019

$P(\mathbf{x}|C_i)*P(C_i)$: P(X|buys_computer = “yes”) * P(buys_computer = “yes”) = 0.028

P(X|buys_computer = “no”) * P(buys_computer = “no”) = 0.007

Therefore, $\mathbf{x}_{test}$ belongs to class (“buys_computer = yes”)

Avoiding the 0-Probability Problem

- Naïve Bayesian prediction requires each conditional prob. be non-zero. Otherwise, the predicted prob. will be zero

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

- **Laplacian correction :** Add dummy records 1 to each value of the attribute whose probability is 0 , for each of the class.
- The "corrected" prob. estimates are close to their "uncorrected" counterparts
- $p_{Lap,k}(x|y) = \frac{C(x,y)+k}{C(y)+k|x|}$
- Here k is the smoothing parameter and |x| represents the cardinality of x, i.e number of values that the attribute takes

Laplacian correction - Example

Age	Student	Credit Rating	Buys Computer
Youth	Yes	Excellent	No
Youth	Yes	Fair	No
Middle Age	No	Fair	Yes
Middle Age	Yes	Excellent	No
Senior	Yes	Excellent	No
Senior	Yes	Fair	Yes
Middle Age	No	Fair	Yes
Youth	No	Excellent	No
Youth	No	Excellent	No

Likelihood Computations

Prior Probabilities :

P(No) = 6/9 p(Yes) = 3/9

Likelihood Computations:

Age	Yes(3)	No(6)
Youth(4)	0/3	4/6
Middle(3)	2/3	1/6
Senior(2)	1/3	1/6

Student	Yes(3)	No(6)
Yes(5)	1/3	4/6
No(4)	2/3	2/6

Credit	Yes(3)	No(6)
Excellent(5)	0/3	5/6
Fair(4)	3/3	1/6

Zero Probability

Test instance (**x**) = (Age = Middle Age, Student = No, Credit = Excellent)

P(No|**x**) = p(**x**|No) x p(No) = p(Middle|No) x p(No|No) x p(Excellent|No) x p(No)

= 1/6 x 2/6 x 5/6 x 6/9 = 0.0309

P(Yes|**x**) = p(**x**|Yes) x p(Yes) = p(Middle| Yes) x p(No| Yes) x p(Excellent| Yes) x p(Yes)

= 2/3 x 2/3 x 0/3 x 3/9 = 0

Here p(No) > p(Yes)

Therefore **x** is classified as Buy_Computer = No. (This needs to be revisited)

Zero Probability (cont …)

Test instance (X) = (Age = Middle Age, Student = No)

P(No/X) = p(X/No) x p(No) = p(Middle/No) x p(No/No) x p(No)

= 1/6 x 2/6 x 6/9 = 0.0370

P(Yes/X) = p(X/Yes) x p(Yes) = p(Middle/ Yes) x p(No/ Yes) x P(Yes)

= 2/3 x 2/3 x 3/9 = 0.1481

Here p(No) < p(Yes)

Problem : Zero probability

Solution : Laplace correction/Smoothing

Laplace smoothing

P(Excellent/Yes) = (0+1)/(3+2) = 1/5

P(Excellent/No) = (5+1)/(6+2) = 6/8

With this correction recompute p(X/No) and p(X/Yes) and then use the values in p(No/X) and p(Yes/x)

P(No/x) = 1/6 x 2/6 x 6/8 x 6/9 = 0.0277

P(Yes/X) = 2/3 x 2/3 x 1/5 x 3/9 = 0.0296

So p(Yes) > p(No)

Hence x belongs to the class Buys_computer = Yes

Gaussian NB – Continuous Valued attributes

Continuous valued attribute A_k is typically assumed to have a Gaussian distribution with a mean μ and standard deviation σ, given by

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

Therefore $P\left({}^{x_k}/_{C_i}\right) = g\left(x_k, \mu_{C_i}, \sigma_{C_i}\right)$ where

μ_{C_i} - mean of the attribute A_k for training tuples of class C_i.

σ_{C_i} - standard deviation of the attribute A_k for training tuples of class C_i.

Example

Consider the following dataset. We want to classify a person if he/she has illness based on Gender and Income.

Test - Find if a Female with an income of 100000 has illness or not

Gender	Income	Illness
Male	40367	No
Female	41524	Yes
Male	46373	Yes
Male	98096	No
Female	102089	No
Female	100662	No
Male	117263	Yes
Male	56645	No

Example (cont…)

From the given data

P(Illness = Yes) = 3/8, P(Illness = No) = 5/8

P(Female/Yes) = 1/3, P(Female/No) = 2/5

$\mu_{Yes} = 68386.66$; $\sigma_{Yes} = 42397.52$

$\mu_{No} = 79571.8$; $\sigma_{No} = 28972.49$

$$g(100000,68386.66,42397.52) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} = 0.00000712$$

$$g(100000,79571.8,28972.49) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} = 0.00001074$$

P(Yes/**x**) = P(Yes) x P(**x**/Yes) = (3/8) x (1/3) x 0.00000712 = 0.00000089

P(No/**x**) = P(No) x P(**x**/No) = (5/8) x (2/5) x 0.00001074 = 0.00000268

Since P(No/**x**) > P(Yes/**x**) the test data <Female, 100000> does not have illness

Multinomial NB
Text Document Classification

- In text document classification, the goal is to find the *best* class for the document.
- The best class in NB classification is the *maximum a posteriori* (MAP) class:

$$c_{map} = \underset{c \in C}{\arg max}\ \hat{p}(c) \prod_{1 \le k \le n_d} \hat{p}\left(t_k / c\right)$$

  - $\hat{p}(c)$ is the prior probability of a document occurring in class c.
  - $\langle t_1, t_2, \ldots, t_{nd} \rangle$ are the tokens in d that are part of the vocabulary we use for classification
  - n_d is the number of such tokens in d.
  - For example: Consider the document content : Beijing and Taipei join the WTO
  - Here $\langle t_1, t_2, \ldots, t_{nd} \rangle$ is ⟨Beijing, Taipei, join, WTO⟩
  - $n_d = 4$,
  - The terms and, the are stop words.

Document Classification (cont…)

Here the prior and Conditional Probabilities are calculated as follows

$$\hat{p}(c) = \frac{N_c}{N}$$

N_c is the number of documents in class c

N is the total number of documents.

$$\hat{p}(t/c) = \frac{T_{ct}}{\sum_{t` \in V} T_{ct`}}$$

T_{ct} is the number of occurrences of t in training documents from class c, including multiple occurrences of a term in a document. (*positional independence assumption).*

Laplacian Correction:

$$\hat{p}(t/c) = \frac{T_{ct}+1}{\sum_{t` \in V} T_{ct`}+B`}$$

B is the number of terms in the vocabulary.

Example

Training Documents	ID	Document	In china?
	1	Chinese Beijing Chinese	Yes
	2	Chinese Chinese Shanghai	Yes
	3	Chinese Macao	Yes
	4	Tokyo Japan Chinese	No
Test Doc	5	Chinese Chinese Chinese Tokyo Japan	?

Example (cont…)

Prior probabilities of class

$$\hat{p}(c) = \frac{3}{4} \ , \ \hat{p}(\bar{c}) = \frac{1}{4}$$

Conditional Probabilities

$$\hat{p}(^{Chinese}/_{c}) = \frac{5+1}{8+6} = \frac{6}{14} = \frac{3}{7}$$

$$\hat{p}(^{Tokyo}/_{c}) = \frac{0+1}{8+6} = \frac{1}{14}$$

$$\hat{p}(^{Japan}/_{c}) = \frac{0+1}{8+6} = \frac{1}{14}$$

$$\hat{p}(^{Chinese}/_{\bar{c}}) = \frac{1+1}{3+6} = \frac{2}{9}$$

$$\hat{p}(^{Tokyo}/_{\bar{c}}) = \hat{p}(^{Japan}/_{\bar{c}}) = \frac{1+1}{3+6} = \frac{2}{9}$$

B = 6 since the vocabulary has 6 words, Number of terms in c = 8 & Number of terms in $\bar{c}$ = 3

Example (cont…)

$$\hat{p}(^{c}/_{d5}) = \frac{3}{4} * \left(\frac{3}{7}\right)^{3} * \frac{1}{14} * \frac{1}{14} \approx 0.0003$$

$$\hat{p}(^{\bar{c}}/_{d5}) = \frac{1}{4} * \left(\frac{2}{9}\right)^{3} * \frac{2}{9} * \frac{2}{9} \approx 0.0001$$

Therefore, the classifier assigns the test document to c = *China*.

Bernoulli model

- ***Multinomial model*** generates ***one term from the vocabulary in each position*** of the document.
- ***Multivariate Bernoulli*** model or ***Bernoulli*** model generates an indicator for ***each term of the vocabulary, either 1 indicating presence of the term in the document or 0 indicating absence***.
- Bernoulli model estimates $\hat{p}(t/c)$ as the *fraction of documents* of class *c* that contain term *t*

Bernoulli model (cont…)

Prior probabilities of class

$$\hat{p}(c) = \frac{3}{4} \ , \ \hat{p}(\bar{c}) = \frac{1}{4}$$

Conditional Probabilities

$$\hat{p}(^{Chinese}/_{c}) = \frac{3+1}{3+2} = \frac{4}{5}$$

$$\hat{p}(^{Tokyo}/_{c}) = \hat{p}(^{Japan}/_{c}) = \frac{0+1}{3+2} = \frac{1}{5}$$

$$\hat{p}(^{Beijing}/_{c}) = \hat{p}(^{Macao}/_{c}) = \hat{p}(^{Shanghai}/_{c}) = \frac{1+1}{3+2} = \frac{2}{5}$$

$$\hat{p}(^{Chinese}/_{\bar{c}}) = \frac{1+1}{1+2} = \frac{2}{3}$$

$$\hat{p}(^{Tokyo}/_{\bar{c}}) = \hat{p}(^{Japan}/_{\bar{c}}) = \frac{1+1}{1+2} = \frac{2}{3}$$

Bernoulli model (cont…)

$$\hat{p}\left({}^{Beijing}/_{\bar{c}}\right) = \hat{p}\left({}^{Macao}/_{\bar{c}}\right) = \hat{p}\left({}^{Shanghai}/_{\bar{c}}\right) = \frac{0+1}{1+2} = \frac{1}{3}$$

The denominators are (3 + 2) and (1 + 2) because there are 3 documents in c and 1 document in $\bar{c}$.

Since there are two cases to consider for each term, occurrence and non-occurrence.

$$\hat{p}\left({}^{c}/_{d5}\right) \approx \hat{p}(c).\hat{p}\left({}^{Chinese}/_{c}\right).\hat{p}\left({}^{Japan}/_{c}\right).\hat{p}\left({}^{Tokyo}/_{c}\right).\left(1-\hat{p}\left({}^{Beijing}/_{c}\right)\right).\left(1-\hat{p}\left({}^{Shanghai}/_{c}\right)\right).\left(1-\hat{p}\left({}^{Macao}/_{c}\right)\right)$$

$$\hat{p}\left({}^{c}/_{d5}\right) \approx \frac{3}{4}.\frac{4}{5}.\frac{1}{5}.\frac{1}{5}.\left(1-\frac{2}{5}\right).\left(1-\frac{2}{5}\right).\left(1-\frac{2}{5}\right) \approx 0.005$$

$$\hat{p}\left({}^{\bar{c}}/_{d5}\right) \approx 1/4 \cdot 2/3 \cdot 2/3 \cdot 2/3 \cdot (1-1/3) \cdot (1-1/3) \cdot (1-1/3) \approx 0.022$$

Therefore the classifier assigns the document d5 to $\bar{c}$ (i.e. not in China)