Chapter 4

Bayesian Network

The basic Graphical Model can be broadly divided into two categories: the Bayesian network and the Markov random field. The main difference is that different types of graphs are used to express the relationship between variables: Bayesian network uses directed acyclic graph to express causality, and Markov random field uses undirected graph (undirected graph) to express the interaction between variables. This structural difference leads to a series of subtle differences in their modeling and inference. This chapter focuses on Bayesian network.

4.1 Introduction

Bayesian Network is a graphic model for describing the connection probabilities among variables. It provides a natural representation of casual relationship and is often used to explore potential relationships among data. In the network, nodes represent variables and directed links represent dependent relationships between variables. With firm mathematical foundation, Bayesian Theory offers a method for brief function calculation and describes the coincidence of brief and evidence, and it possesses the incremental learning property that brief varies along with the variation of evidence. In data mining, Bayesian networks can deal with incomplete or noisy data sets. It describes correlations among data with probabilistic measurement, thereby solving the problem of data inconsistency. It describes correlations among data with the graphical method, which has clear semantic meaning and understandable representation. It also makes predictions and analyses with casual relationships among data. Bayesian network is becoming one of the most remarkable data mining methods due to its unique properties, including unique knowledge representation of uncertain information, capability for handling probability, and incremental learning with prior knowledge.

4.1.1 *History of Bayesian Theory*

The foundational work of the Bayesian School is credited to Reverend Thomas Bayes' (1702–1761) *An Essay toward Solving a Problem in the Doctrine of Chances*. Maybe he felt the work was not perfect enough as this work was published not in his lifetime but posthumously by his friend. As the famous mathematician P. S. Laplace deduced Law of Succession based on Bayesian method, Bayesian method and theory began to be recognized. In the 19th century, because the problem of motivating and constructing prior probabilities was not adequately answered, Bayesian theory was not well accepted. Early in 20th century, B. de Finetti and H. Jeffreys made significant contribution to Bayesian theory. After World War II, Wald A. proposed the statistical decision theory. In this theory, Bayesian method played an important role. Besides, the development of information science also contributed to the reincarnation of Bayesian theory. In 1958, Bayes' paper was republished by *Biometrika*, the most historical statistical magazine in Britain. In the 1950s, H. Robbins suggested to combine empirical Bayesian approach and conventional statistical method. This novel approach captured the attention of the statistical research field and soon showed its merits and thus became an active research direction.

With the development of artificial intelligence, especially after the rise of machine learning and data mining, Bayesian theory gained increasingly more developments and applications. Its connotation has also varied greatly from its origin. In the 1980s, Bayesian networks were used for knowledge representation in expert systems. In the 1990s, Bayesian networks were applied to data mining and machine learning. Recently, more and more papers concerning Bayesian theory are being published, which cover most fields of artificial intelligence, including casual reasoning, uncertain knowledge representation, pattern recognition, clustering analysis, and so on. There is even an organization and a journal, *ISBA*, which focus especially on the progress on Bayesian theory.

4.1.2 *Basic Concepts of the Bayesian Method*

In Bayesian theory, all kinds of uncertainties are represented with probabilities. Learning and reasoning are implemented via probabilistic rules. The Bayesian learning results are distributions of random variables, which show the briefs to various possible results. The foundations of the Bayesian School are Bayesian theorem and Bayesian assumption. Bayesian theorem connects prior probabilities of events with their posterior probabilities. Assume that the joint probability density of random vectors $\mathbf{x}$ and $\boldsymbol{\theta}$ is $p(\mathbf{x}, \boldsymbol{\theta})$, and $p(\mathbf{x})$ and $p(\boldsymbol{\theta})$ give the marginal densities

of $\mathbf{x}$ and $\boldsymbol{\theta}$, respectively. In common cases, $\mathbf{x}$ is an observation vector and $\boldsymbol{\theta}$ is an unknown parameter vector. The estimation of parameter $\boldsymbol{\theta}$ can be obtained with the observation vector via Bayesian theorem. The Bayesian theorem is denoted as follows:

$$p(\theta|x) = \frac{\pi(\theta)p(x|\theta)}{p(x)} = \frac{\pi(\theta)p(x|\theta)}{\int \pi(\theta)p(x|\theta)d\theta} \quad (\pi(\theta) \text{ is the prior of } \theta). \tag{4.1}$$

From this formula, we see that in the Bayesian method the estimation of a parameter needs prior information of the parameter and information from evidence. In contrast, traditional statistical method, e.g., maximum likelihood, only utilizes the information from evidence. The general process to estimate a parameter vector via Bayesian method is as follows:

(1) Regard unknown parameters as random vectors. This is the fundamental difference between Bayesian method and traditional statistical approach.
(2) Define the prior $\pi(\theta)$ based on previous knowledge of the parameter θ. This step is a controversial step and is debated by conventional statistical scientists.
(3) Calculate posterior density and make estimation of parameters according to the posterior distribution.

In the second step, if there is no previous knowledge to determine the prior $\boldsymbol{\pi}(\boldsymbol{\theta})$ of a parameter, Bayes suggested to assume uniform distribution to be its distribution. This is called Bayesian assumption. Intuitionally, Bayesian assumption is well accepted. Yet, it encounters a problem when no information about prior distribution is available, especially when the parameter is infinite. Empirical Bayes (EB) estimator combines a conventional statistical method and Bayesian method, so that it applies the conventional method to gain the marginal density $\boldsymbol{p}(\boldsymbol{x})$, and then ascertains prior $\boldsymbol{\pi}(\boldsymbol{\theta})$ with the following formula:

$$p(x) = \int_{-\infty}^{+\infty} \pi(\theta)p(x|\theta)d\theta.$$

4.1.3 *Applications of Bayesian Network in Data Mining*

1. Bayesian method in classification and regression analysis

Classification is to classify an object based on its feature vector and some constrains. In data mining, we mainly study how to learn classification rules from data or experiences. For classification, sometimes each feature vector corresponds

to one class label (determinate classification); sometimes different classes can overlap, where samples from different classes are very similar and we can only tell the probabilities of a sample in all classes and choose a class for the sample according to the probabilities. Bayesian School provides two methods to handle this situation: one is selecting the class with maximum posterior probability; the other is selecting the class with maximum utility function or minimum lost function. Let the feature vector be $\boldsymbol{X} = (x_1, x_2, \ldots, x_m)$, and class vector be $\mathbf{C} = (c_1, c_2, \ldots, c_l)$. Classification then assigns a class c_i ($i \in (1, \ldots, l)$) to a feature vector $\boldsymbol{X}$.

In the first method, the class c_i with maximum posterior probability will be selected, viz. $P(c_i|x) \geq P(c_j|x) \quad j \in (1, \ldots, l)$. In this case, the decision function is $r_i(x) = p(c_i|x)$. It has been proved that in this method the minimum classification error can be guaranteed.

The second method is often used in decision theory. It utilizes average benefit to evaluate decision risk, which has a close relationship with degrees of uncertainty. Let $L_{ij}(\boldsymbol{X})$ be the loss of misclassifying a feature vector X of class c_i to class c_j. The class with minimum loss of $\boldsymbol{X}$ is Minimize$_i$ $\left\{\sum_{j=1}^{l} L_{ij}(x) \cdot P(c_j|x)\right\}$. In this case, the decision function is $r_i(x) = \sum_{j=1}^{l} L_{ij}(x) \cdot P(c_j|x)$. If the diagonal elements of $L_{ij}(\boldsymbol{X})$ are all 0 and non-diagonal elements of $L_{ij}(\boldsymbol{X})$ are all 1, viz. correct classification makes no loss and misclassification has same loss, the first method and the second method are equal.

In data mining, the research on Bayesian classification mainly focuses on how to learn the distribution of feature vectors and the correlation among feature vectors from data so as to find the best P($c_i|x$) and $L_{ij}(\boldsymbol{X})$). By now successful models have been proposed, including Naïve Bayesian, Bayesian Network, and Bayesian Neural Network. The Bayesian classification method has been successfully applied to many fields, such as text classification, alphabet recognition, and economic prediction.

2. Bayesian method in casual reasoning and uncertain knowledge representation

The Bayesian network is a graph that describes the probabilistic relations of random variables. Currently, Bayesian network has been the primary method of uncertain knowledge representation in an expert system. Many algorithms have been proposed to learn Bayesian network from data. These techniques have gained reasonable success in data modeling, uncertainty reasoning, and so on.

Compared with other knowledge representation methods in data mining, such as rule representation, decision tree, and artificial neural networks, Bayesian

network possesses the following merits in knowledge representation (Cooper *et al.* 1992):

(1) Bayesian network can conveniently handle incomplete data. For example, when we face a classification or regression problem with multiple correlative variables, the correlation among variables is not the key element for standard supervised learning algorithms. As a result, missing values will cause large predictive bias. Yet, Bayesian network can handle incomplete data with the probabilistic correlation of variables.
(2) Bayesian network can learn the casual relation of variables. Casual relation is a very important pattern in data mining, mainly because in data analysis, it is helpful for field knowledge understanding; it can also easily lead to precise prediction even under much interference. For example, some sale analyzers wonder whether increasing their advertisements will cause sales to increase. To get the answer, the analyzer must know whether the increase in advertisement is the cause of increase in sales with a Bayesian network. This question can be easily answered even without experimental data because the causal relation has been encoded in this network.
(3) The combination of Bayesian network and Bayesian statistics can take full advantage of field knowledge and information from data. Everyone with modeling experience knows that prior information or field knowledge is very important to modeling, especially when sample data are sparse or difficult to obtain. Some commercial expert systems, which are constructed purely based on field expert knowledge, are a perfect example. Bayesian network, which expresses a dependent relation with directed edge and uses probabilistic distribution to describe the strength of dependence, can integrate prior knowledge and sample information well.
(4) The combination of Bayesian network and other models can effectively avoid the problem of over-fitting.

3. Bayesian method in clustering and pattern discovery

Generally, clustering is a special case of model selection. Each clustering pattern can be viewed as a model. The task of clustering is to find a pattern, which best fits the nature of data, from many models based on analysis and some other strategies. Bayesian method integrates prior knowledge and characteristics of the current data to select the best model.

With Bayesian analysis, Vaithyanathan and Dom (1998) proposed a model-based hierarchical clustering method. By partitioning the feature set, they organized

data in to a hierarchical structure. The features either have a unique distribution in different classes or have the same distribution in some classes. They also give the method to determine the model structure with marginal likelihood, including how to automatically determine the number of classes, depth of the model tree, and the feature subset of each class.

AutoClass is a typical system that implements clustering with the Bayesian method. This system automatically determines the number of classes and the complexity of model by searching all possible classifications in the model space. It allows for features in certain classes to have correlation and successive relations existing among classes (in the hierarchical structure of classes, some classes can share some model parameters). Detailed information about AutoClass can be found on the website http://ic-www.arc.nasa.gov/ic/projects/bayes-group/autoclass.

Here, we have only listed some typical applications of Bayesian method. The applications of Bayesian method in data mining are far more than just these. Bayesian neural network, which combines Bayesian method and neural network, and Bayes Point Machine, which combines Bayesian method and statistical learning, are all interesting examples of applications of Bayesian method. Interested readers can find more in the book by Amari (1985).

4.2 Foundation of Bayesian Probability

4.2.1 *Foundation of Probability Theory*

Probability is a branch of mathematics which focuses on the regularity of random phenomena. Random phenomena are phenomena for which different results appear under the same conditions. Random phenomena include individual random phenomena and substantive random phenomena. The regularity from the observation of substantive random phenomena is called statistical regularity.

Statistically, we conventionally call an observation, a registration, or an experiment about phenomenon a trial. A random trial is an observation on a random phenomenon. Under the same condition, random trials may lead to different results. But the sphere of all the possible results is estimable. The result of a random trial is both uncertain and predictable. Statistically, the result of a random trial is called a random event, in short an event.

A random event is the result that will appear or not appear in a random trial. In a random phenomenon, the frequency of a mark is the total number of times the mark appears in all trials.

Table 4.1. The result of sampling of product quality

Number of products examined	5	10	50	100	300	600	1,000	5,000	10,000
Number of qualified products	5	8	44	91	272	542	899	4,510	8,999
Frequency of qualification	1	0.8	0.88	0.91	0.907	0.892	0.899	0.902	0.8997

Example 4.1. To study the product quality of a factory, we make some random samplings. In each sampling, the number of samples is different. The result of sampling is recoded and presented in Table 4.1.

In the table, the number of products examined is the total number of products examined in one sample. The number of qualified products is the total number of qualified products in the examination. The frequency of qualification is the proportion of qualified products in all the products examined in one sample. From the table, we can easily see the relation between the number of a mark and the frequency of a mark. We can also find a statistical regularity. That is, as the number of products examined increases, the frequency of qualification inclines to 0.9 stably, or the frequency of qualification wavers around a fixed number $p = 0.9$. So, p is the statistical stable center of this series of trials. It represents the possibility of qualification of an examined product. The possibility is called probability.

Definition 4.1 (Statistical Probability). If in a number of repeated trials, the frequency of event A inclines to a constant p stably, it represents the possibility of appearance of event A, and we call this constant p the probability of event A, shortly $P(A)$:

$$p = P(A).$$

So, a probability is the stable center of a frequency. A probability of any event A is a non-negative real number that is not bigger than 1:

$$0 \leq P(A) \leq 1.$$

The statistical definition of probability has a close relationship with frequency and is easily understood. But it is a tough problem to find the probability of an arbitrary event with experiments. Sometimes it is even impossible. So we often calculate probability with the classical probabilistic method or geometrical probabilistic method.

Definition 4.2 (Classical Probability). Let a trial have only finite N possible results, or N basic events. If event A contains K possible results, we call K/N the probability of event A, shortly $P(A)$:

$$P(A) = K/N. \tag{4.2}$$

To calculate classical probability, we need to know the number of all the basic events. So, classical probability is restricted to cases of finite population. In the case of an infinite population or if the total number of basic events is unknown, the geometrical probability model is used to calculate the probability. Besides, geometrical probability also gives a general definition of probability.

Geometrical random trial: Assume Ω is a bounded domain of M-dimensional space, and $L(\Omega)$ is the volume of Ω. We consider the random trial that we throw a random point into Ω evenly and assume the following: (a) The random point may fall in any domain of Ω, but cannot fall outside of Ω. (b) The distribution of the random point in Ω is even, viz. the possibility that the random point falls into a domain is proportional to the volume of the domain and is independent of the position or the shape of the domain in Ω. Under the restrictions above, we call a trial a geometrical random trial where Ω is basic event space.

Event in a geometrical random trial: Assume that Ω is the basic event space of geometrical random trial, and A is a subset of Ω that can be measured with volume, where $L(A)$ is the M-dimensional volume of A. Then the event "random point falls in domain A" is represented with A. In Ω, a subset that can be measured with volume is called a measurable set. Each measurable set can be viewed as an event. The set of all measurable subsets is represented by F.

Definition 4.3 (Geometrical Probability). Assume that Ω is a basic event space of a geometrical random trial and F is the set of all measurable subsets of Ω. Then the probability of any event A in F is the ratio between the volume of A and that of Ω:

$$P(A) = V(A)/V(\Omega). \tag{4.3}$$

Definition 4.4 (Conditional Probability). The probability of an event A under the condition that event B has happened is denoted by $P(A|B)$. We call it the conditional probability of event A under condition B. $P(A)$ is called unconditional probability.

Example 4.2. There are two white balls and one black ball in a bag. Now we take out two balls in turn. The following questions arise: (a) What is the probability of the event that a white ball is picked the first time? (b) What is the probability of the event that a white ball is picked the second time when a white ball has been picked the first time?

Solution: Assume A is the event that a white ball is picked the first time, and B is the event that a white ball is picked the second time. Then $\{B|A\}$ is the event that a white ball is picked the second time when a white ball has been picked the first time. According to Definition 4.4, we have the following:

(1) No matter whether it is repeated sampling or non-repeated sampling, $P(A) = 2/3$.
(2) When sampling is non-repeated, $P(B|A) = 1/2$; when sampling is repeated, $P(B|A) = P(B) = 2/3$. The conditional probability equals non-conditional probability.

If the appearance of any of event A or B will not affect the probability of the other event, viz. $P(A) = P(A|B)$ or $P(B) = P(B|A)$. We call events A and B independent events.

Theorem 4.1 (Addition Theorem). *The probability of the sum of two mutually exclusive events equals the sum of the probabilities of the two events, that is,*

$$P(A + B) = P(A) + P(B).$$

The sum of probabilities of two mutually inverse events is 1. In other words, if $A + A^{-1} = \Omega$, *A and* A^{-1} *are mutually inverse, then* $P(A) + P(A^{-1}) = 1$, *or* $P(A) = 1 - P(A^{-1})$.

If A and B are two arbitrary events, then

$$P(A + B) = P(A) + P(B) - P(AB)$$

holds. This theorem can be generalized to the case that involves more than three events:

$$P(A + B + C) = P(A) + P(B) + P(C) - P(AB) - P(BC) - P(CA) + P(ABC).$$

Theorem 4.2 (Multiplication Theorem). *Assume A and B are two mutually independent non-zero events, then the probability of the multiple event equals the multiplication of probabilities of events A and B, that is,*

$$P(A \cdot B) = P(A) \cdot P(B) \quad \text{or} \quad P(A \cdot B) = P(B) \cdot P(A).$$

Assume A and B are two arbitrary non-zero events, then the probability of the multiple event equals the multiplication of the probability of event A (or B) and the

conditional probability of event B (or A) under condition A (or B):

$$P(A \cdot B) = P(A) \cdot P(B|A) \quad \text{or} \quad P(A \cdot B) = P(B) \cdot P(A|B).$$

This theorem can be generalized to the case that involves more than three events. When the probability of multiple events $P(A_1 A_2 \ldots A_{n-1}) > 0$, *we have*

$$P(A_1 A_2 \ldots A_n) = P(A_1) \cdot P(A_2|A_1) \cdot P(A_3|A_1 A_2) \ldots P(A_n|A_1 A_{2\ldots} A_{n-1}).$$

If all the events are pairwise independent, we have

$$P(A_1 A_2 \ldots A_n) = P(A_1) \cdot P(A_2) \cdot P(A_3) \ldots P(A_n).$$

4.2.2 *Bayesian Probability*

(1) **Prior Probability:** A prior probability is the probability of an event that is gained from historical materials or subjective judgments. It is not verified and is estimated in the absence of evidence. So it is called prior probability. There are two kinds of prior probabilities. One is objective prior probability, which is calculated according to historical materials; the other is subjective prior probability, which is estimated purely based on the subjective experience when historical material is absent or incomplete.

(2) **Posterior Probability:** A posterior probability is the probability that is computed according to the prior probability and additional information from investigation via Bayesian formula.

(3) **Joint Probability:** The joint probability of two events is the probability of the intersection of the two events. It is also called multiplication formula.

(4) **Total Probability Formula:** Assume all the influence factors of event A are $B_1, B_2, \ldots,$ and they satisfy $B_i \cdot B_j = \emptyset, (i \neq j)$ and $P(\cup B_i) = 1, P(B_i) > 0, i = 1, 2, \ldots,$ then we have

$$P(A) = \sum P(B_i) P(A|B_i). \tag{4.4}$$

(5) **Bayesian Formula:** Bayesian formula, which is also called posterior probability formula or inverse probability formula, has wide applications.

Assume $P(B_i)$ is the prior probability, and $P(A_j|B_i)$ is the new information gained from investigation, where $i = 1, 2, \ldots, n$, and $j = 1, 2, \ldots, m$. Then the posterior probability calculated with Bayesian formula is

$$P(B_i|A_j) = \frac{P(B_i) P(A_j|B_i)}{\sum_{k=1}^{m} P(B_k) P(A_j|B_k)}. \tag{4.5}$$

Table 4.2. Daily outputs of three teams

	B_1	B_2	Total
Team A_1	2,000	1,000	3,000
Team A_2	1,500	500	2,000
Team A_3	500	500	1,000
Total	4,000	2,000	6,000

Example 4.3. One kind of product is made in a factory. Three work teams (A_1, A_2, and A_3) are in charge of two specifications (B_1 and B_2) of the product. Their daily outputs are listed in Table 4.2.

Now we randomly pick out one from the 6,000 products. Please answer the following questions.

1. Calculate the following probabilities with Classical Probability

(1) Calculate the probabilities that the picked product comes from the outputs of A_1, A_2, or A_3, respectively.

$$\text{Solution:} \quad P(A_1) = 3{,}000/6{,}000 = 1/2,$$
$$P(A_2) = 2{,}000/6{,}000 = 1/3,$$
$$P(A_3) = 1{,}000/6{,}000 = 1/6.$$

Calculate the probabilities that the picked product belongs to B_1 or B_2, respectively.

$$\text{Solution:} \quad P(B_1) = 4{,}000/6{,}000 = 2/3,$$
$$P(B_2) = 2{,}000/6{,}000 = 1/3.$$

(2) Calculate the probability that the pick product is B_1 and comes from A_1.

$$\text{Solution:} \quad P(A_1 \cdot B_1) = 2{,}000/6{,}000 = 1/3.$$

(3) If the fact that the product comes from A_1 is known, what is the probability that it belongs to B_1?

$$\text{Solution:} \quad P(B_1|A_1) = 2{,}000/3{,}000 = 2/3.$$

(4) If the product belongs to B_2, what is the probability that it comes from A_1, A_2, or A_3, respectively?

$$\begin{aligned} \text{Solution:} \quad & P(A_1|B_2) = 1{,}000/2{,}000 = 1/2, \\ & P(A_2|B_2) = 500/2{,}000 = 1/4, \\ & P(A_3|B_2) = 500/2{,}000 = 1/4. \end{aligned}$$

2. Calculate the following probabilities with Conditional Probability

(1) If a product comes from A_1, what is the probability that it belongs to B_1?

$$\text{Solution:} \quad P(B_1|A_1) = (1/3)/(1/2) = 2/3.$$

(2) If a product belongs to B_2, what is the probability that it comes from A_1, A_2, or A_3, respectively?

$$\begin{aligned} \text{Solution:} \quad & P(A_1|B_2) = (1/6)/(1/3) = 1/2, \\ & P(A_2|B_2) = (1/12)/(1/3) = 1/4, \\ & P(A_3|B_2) = (1/12)/(1/3) = 1/4. \end{aligned}$$

3. Calculate the following probabilities with Bayesian formula

(1) Known:
$$\begin{aligned} P(B_1) &= 4000/6000 = 2/3, \\ P(B_2) &= 2000/6000 = 1/3, \\ P(A_1|B_1) &= 1/2, \\ P(A_1|B_2) &= 1/2. \end{aligned}$$

Question: If a product comes from A_1, what is the probability that it belongs to B_2?
Solution: Calculate Joint Probabilities:

$$\begin{aligned} P(B_1)P(A_1|B_1) &= (2/3)(1/2) = 1/3, \\ P(B_2)P(A_1|B_2) &= (1/3)(1/2) = 1/6. \end{aligned}$$

Calculate Total Probability:

$$P(A_1) = (1/3) + (1/6) = 1/2.$$

Calculate posterior probability according to Bayesian formula:

$$P(B_2|A_1) = (1/6) \div (1/2) = 1/3.$$

(2) Known: $P(A_1) = 3{,}000/6{,}000 = 1/2,$

$$P(A_2) = 2{,}000/6{,}000 = 1/3,$$

$$P(A_3) = 1{,}000/6{,}000 = 1/6,$$

$$P(B_2|A_1) = 1{,}000/3{,}000 = 1/3,$$

$$P(B_2|A_2) = 500/2{,}000 = 1/4,$$

$$P(B_2|A_3) = 500/1{,}000 = 1/2.$$

Question: If a product belongs to B_2, what is the probability that it comes from A_1, A_2, or A_3?
Solution: Calculate Joint Probabilities:

$$P(A_1)P(B_2|A_1) = (1/2)(1/3) = 1/6,$$

$$P(A_2)P(B_2|A_2) = (1/3)(1/4) = 1/12,$$

$$P(A_3)P(B_2|A_3) = (1/6)(1/2) = 1/12.$$

Calculate Total Probability P(B2):

$$P(B_2) = \sum P(A_i)P(B_2|A_i)$$
$$= (1/2)(1/3) + (1/3)(1/4) + (1/6)(1/2) = 1/3.$$

Calculate posterior probability according to Bayesian formula:

$$P(A_1|B_2) = (1/6) \div (1/3) = 1/2,$$

$$P(A_2|B_2) = (1/12) \div (1/3) = 1/4,$$

$$P(A_3|B_2) = (1/12) \div (1/3) = 1/4.$$

4.3 Bayesian Problem Solving

Bayesian learning theory utilizes prior information and sample data to estimate unknown data. Probabilities (joint probabilities and conditional probabilities) are the representation of prior information and sample data in Bayesian learning theory.

How to get the estimation of these probabilities (also called probabilistic density estimation) is a topic of much controversy in Bayesian learning theory. Bayesian density estimation focuses on how to estimate the distribution of unknown variables (vectors) and its parameters based on sample data and prior knowledge from human experts. It includes two steps. One is to determine prior distributions of unknown variables; the other is to get the parameters of these distributions. If we know nothing about previous information, the distribution is called non-informative prior distribution. If we know the distribution and seek its proper parameters, the distribution is called informative prior distribution. Because learning from data is the most elementary characteristic of data mining, non-informative prior distribution is the main subject of Bayesian learning theory research.

The first step of Bayesian problem solving is to select a Bayesian prior distribution. This is a key step. There are two common methods to select a prior distribution, namely, subjective method and objective method. The former makes use of human experience and expert knowledge to assign prior distribution. The latter is done by analyzing the characteristics of the data to get the statistical features of the data. It requires having a sufficient amount of data to get the true distribution of data. In practice, these two methods are often combined. Several common methods for prior distribution selection are listed in the following. Before we discuss these methods, we give some definitions first.

Let θ be the parameter of a model, $\boldsymbol{X} = (x_1, x_2, \ldots, x_n)$ be the observed data, and $\pi(\theta)$ be the prior distribution of θ. $\pi(\theta)$ represents the brief of parameter θ when no evidence exists. $l(x_1, x_2, \ldots, x_n|\theta) \propto p(x_1, x_2, \ldots, x_n|\theta)$ is the likelihood function. It represents the brief of unknown data when parameter θ is known. $h(\theta|x_1, x_2, \ldots, x_n) \propto p(\theta|x_1, x_2, \ldots, x_n)$ is the brief of parameter θ after new evidence appears. Bayesian theorem describes the relation between them as follows:

$$h(\theta|x_1, x_2, \ldots, x_n) = \frac{\pi(\theta)p(x_1, x_2, \ldots, x_n|\theta)}{\int \pi(\theta)p(x_1, x_2, \ldots, x_n|\theta)d\theta} \propto \pi(\theta)l(x_1, x_2, \ldots, x_n|\theta). \quad (4.6)$$

Definition 4.5 (Kernel of Distribution Density). If $f(x)$, the distribution density of random variable z can be decomposed as $f(x), = cg(x)$, where c is a constant independent of x, we call $g(x)$ the kernel of $f(x)$, or short for $f(x) \propto g(x)$. If we know the kernel of distribution density, we can determine the corresponding constant according to the fact that the integral of distribution density in the whole space is 1. Therefore, the key to solving the distribution density of a random variable is to solve the kernel of its distribution density.

Definition 4.6 (Sufficient Statistic). To parameter θ, the statistic $t(x_1, x_2, \ldots, x_n)$ is sufficient if the posterior distribution of θ, $h(\theta|x_1, x_2, \ldots, x_n)$, is always a function of θ and $t(x_1, x_2, \ldots, x_n)$ in spite of its prior distribution.

This definition clearly states that the information of θ in data can be represented by its sufficient statistics. Sufficient statistics are connections between posterior distribution and data. Below, we give a theorem to judge whether a statistic is sufficient.

Theorem 4.3 (The Neyman–Fisher Factorization Theorem). *Let* $f\theta(\mathbf{x})$ *be the density or mass function for the random vector* $\mathbf{x}$, *parametrized by the vector* θ. *The statistic* $t = T(\mathbf{x})$ *is sufficient for* θ *if and only if there exist functions* $a(\mathbf{x})$ (*not depending on* θ) *and* $b\theta(t)$ *such that* $f\theta(\mathbf{x}) = \mathbf{a}(\mathbf{x})\ \mathbf{b}\theta(\mathbf{t})$ *for all possible values of* $\mathbf{x}$.

4.3.1 *Common Methods for Prior Distribution Selection*

1. Conjugate family of distributions

Raiffa and Schaifeer suggested using conjugate distributions as prior distributions, where the posterior distribution and the corresponding prior distribution are the same kind of distribution. The general description of conjugate distribution is as follows:

Definition 4.7. Let the conditional distribution of samples $x_1, x_2, \ldots, x_n$ under parameters θ be $p(x_1, \mathrm{x}_2, \ldots, x_n|\theta)$. If the prior density function $\pi(\theta)$ and its resulting posterior density function $\pi(\theta|x)$ are in the same family, the prior density function $\pi(\theta)$ is said to be conjugate to the conditional distribution $p(x|\theta)$.

Definition 4.8. Let $P = \{p(x|\theta) : \theta \in \Theta\}$ be the density function family with parameters θ. $H = \pi(\theta)$ is the prior distribution family of θ. If for any given $p \in P$ and $\pi \in H$, the resulting posterior distribution $\pi(\theta|x)$ is always in family H, H is said to be the conjugate family to P.

When the density functions of data distribution and its prior are all exponential functions, the resulting function of their multiplication is the sample kind of exponential function. The only difference is a factor of proportionality. So we have the following theorem:

Theorem 4.4. *If for random variable* Z, *the kernel of its density function* $f(x)$ *is an exponential function*, *the density function belongs to the conjugate family.*

All the distributions with exponential kernel function compose the exponential family, *which includes binary distribution*, *multinomial distribution*, *normal distribution*, *Gamma distribution*, *Poisson distribution*, *and Dirichlet distribution.*

Conjugate distributions can provide a reasonable synthesis of historical trials and a reasonable precondition for future trials. The computation of non-conjugate distribution is rather difficult. In contrast, the computation of conjugate distribution is easy, where only multiplication with the prior is required. So, in fact, the conjugate family makes a firm foundation for the practical application of Bayesian learning.

2. Principle of maximum entropy

Entropy is used to quantify the uncertainty of an event in information theory. If a random variable x takes two different possible values, namely a and b, compare the following two case:

$$(1) \quad p(x = a) = 0.98, \quad p(x = b) = 0.02,$$
$$(2) \quad p(x = a) = 0.45, \quad p(x = b) = 0.55.$$

Obviously, the uncertainty of case 1 is much less than that of case 2. Intuitively, we can see that the uncertainty will reach maximum when the probabilities of the two values are equal.

Definition 4.9. Let x be a discrete random variable. It takes at most countable values $a_1, a_2, \ldots, a_k, \ldots$, and $p(x = a_i) = p_i, i = 1, 2, \ldots$. The entropy of x is $H(x) = -\sum_i p_i \ln p_i$. For a continuous random variable x, if the integral $H(x) = -\int p(x) \ln p(x)$ is meaningful, where $p(x)$ is the density of variable x, the integral is called the entropy of a continuous random variable.

According to the definition, when two random variables have the same distribution, they have equal entropy. So, entropy is only related to distribution.

Principle of maximum entropy: For non-information data, the best prior distribution is the distribution which makes the entropy maximum under parameters θ.

It can be proved that the entropy of a random variable, or vector, reaches maximum if and only if its distribution is uniform. Hence, the Bayesian assumption, which assumes non-information prior to be uniform, fits the principle of maximum entropy. It makes the entropy of a random variable, or vector, maximum. Below is the proof of the case of limited valued random variable.

Theorem 4.5. *Let random variable x take limited values $a_1, a_2, \ldots, a_n$. The corresponding probabilities are $p_1, p_2, \ldots, p_n$. The entropy $H(x)$ is maximum if and only if $p_1 = p_2 = \cdots = p_n = 1/n$.*

Proof. Consider $G(p_1, p_2, \ldots, p_n) = -\sum_{i=1}^{n} p_i \ln p_i + \lambda(\sum_{i=1}^{n} p_i - 1)$. To find its maximum, we let the partial derivative of G with respect to p_i be 0, and thus get

the following equation:

$$0 = \frac{\partial G}{\partial p_i} = -\ln p_{i-1} + \lambda \qquad (i = 1, 2, \ldots, n).$$

Solving the equations, we get $p_1 = p_2 = \cdots = p_n$. Because $\sum_{i=1}^{n} p_i = 1$, we have $p_1 = p_2 = \cdots = p_n = 1/n$. Here, the corresponding entropy is $-\sum_{i=1}^{n} \frac{1}{n} \ln \frac{1}{n} = \ln n$.

For a continuous random variable, the result is the same.

From above, when there is no information to determine prior distribution, the principle of maximum distribution is a reasonable choice for prior selection. There are many cases where no information is available to determine the prior, so Bayesian assumption is very important in these cases. □

3. Jeffrey's principle

Jeffrey had made a significant contribution to prior distribution selection. He proposed an invariance principle, which solved a conflict in Bayesian assumption very well and gave an approach to find prior density. Jeffrey's principle is composed of two parts: one is a reasonable requirement to prior distribution; the other is giving out a concrete approach to find a correct prior distribution fitting the requirement.

There is a conflict in Bayesian assumption: If we choose uniform as the distribution of parameter θ, once we take function $g(\theta)$ as parameter, it should also obey uniform distribution and vice versa. Yet, the above precondition cannot lead to the expected result. To solve the conflict, Jeffrey proposed an invariance request. That is, a reasonable principle for prior selection should have invariance.

If we choose $\pi(\theta)$ as the prior distribution of parameter θ, according to invariance principle, $\pi_g(g(\theta))$, the distribution of function $g(\theta)$ should satisfy the following:

$$\pi(\theta) = \pi_g(g(\theta))|g'(\theta)|. \tag{4.7}$$

The key point is how to find a prior distribution $\pi(\theta)$ to satisfy the above condition. Jeffrey skillfully utilized the invariance of the Fisher information matrix to find a required $\pi(\theta)$.

The distribution of parameter θ has the kernel of the square root of information matrix $I(\theta)$, viz. $\pi(\theta) \propto |I(\theta)|^{1/2}$, where $I(\theta) = E\left(\frac{\partial \ln p(x_1, x_2, \ldots, x_n; \theta)}{\partial \theta}\right)\left(\frac{\partial \ln p(x_1, x_2, \ldots, x_n; \theta)}{\partial \theta}\right)'$. The concrete deriving process is not presented here. Interested readers can find them in related references. It is noted that Jeffrey's principle is just a

principle of finding a reasonable prior, while using the square root of the information matrix as the kernel of the prior is a concrete approach. They are different. In fact, we can seek other concrete approaches to embody the principle.

4.3.2 *Computational Learning*

Learning is the process by which a system can improve its behavior after running. Is the posterior distribution gained via Bayesian formula better than its corresponding prior? What is its learning mechanism? Here, we analyze normal distribution as an example to study the effect of prior information and sample data by changing parameters.

Let $x_1, x_2, \ldots, x_n$ be a sample from normal distribution $N(\theta, \sigma_1^2)$, where σ_1^2 is known and θ is unknown. To seek $\tilde{\theta}$, the estimation of θ, we take another normal distribution as the prior of θ. That is,

$$\pi(\theta) = N(\mu_0, \sigma_0^2).$$

The resulting posterior distribution of θ is also a normal distribution:

$$h(\theta|\bar{x}_1) = N(\alpha_1, d_1^2),$$

where

$$\bar{x}_1 = \sum_{i=1}^{n} \frac{x_i}{n}, \quad \alpha_1 = \left(\frac{1}{\sigma_0^2}\mu_0 + \frac{n}{\sigma_1^2}\bar{x}_1\right) \Big/ \left(\frac{1}{\sigma_0^2} + \frac{n}{\sigma_1^2}\right), \quad d_1^2 = \left(\frac{1}{\sigma_0^2} + \frac{n}{\sigma_1^2}\right)^{-1}.$$

Take α_1, the expectation of the posterior $h(\theta|\bar{x})$ as the estimation of θ, then we have

$$\tilde{\theta} = E(\theta|\bar{x}_1)\left(\frac{1}{\sigma_0^2}\mu_0 + \frac{n}{\sigma_1^2}\bar{x}_1\right) \cdot d_1^2. \tag{4.8}$$

Therefore, $\tilde{\theta}$, the estimation of θ, is the weighted average of μ_0, the expectation of prior, and $\bar{x}_1$, the sample mean. σ_0^2 is the variance of $N(\mu_0, \sigma_0^2)$, so its reciprocal, $1/\sigma_0^2$, is the precision of μ_0. Similarly, σ_1^2/n is the variance of sample mean $\bar{x}$, so its reciprocal is the precision of $\bar{x}_1$. Hence, we see that $\tilde{\theta}$ is the weighted average of μ_0 and $\bar{x}_1$, where the weights are their precisions, respectively. The smaller the variance, the bigger the weight. Besides, the bigger the sample size n, the smaller the variance σ_1^2/n, or the bigger the weight of sample mean. This means that when, n is quite large, the effect of the prior mean will be very small. This analysis illustrates that the posterior from the Bayesian formula integrates prior information and sample data.

The result is more reasonable than that based on merely prior information or sample data. The learning mechanism is effective. The analysis based on other conjugate prior distribution leads to a similar result.

According to the previous discussion, with the conjugate prior, we can use the posterior information as the prior of the next computation and seek the next posterior by integrating the information of more samples. If we repeat this process time after time, can we get posterior increasingly close to reality? We study this problem in the following.

Let new sample $x_1, x_2, \ldots, x_n$ be from normal distribution $N(\theta, \sigma_2^2)$, where σ_2^2 is known and θ is unknown. If we use the previous posterior $h(\theta|\bar{x}_1) = N(\alpha_1, d_1^2)$ as the prior of the next round of computation, then the new posterior is $h_1(\theta|\bar{x}_2) = N\big(\alpha_2, d_2^2\big)$, where

$$\bar{x}_2 = \sum_{i=1}^{n} \frac{x_i}{n}, \quad \alpha_2 = \left(\frac{1}{d_1^2}\alpha_1 + \frac{n}{\sigma_2^2}\bar{x}_2\right) \Big/ \left(\frac{1}{d_1^2} + \frac{n}{\sigma_2^2}\right), \quad d_2^2 = \left(\frac{1}{d_1^2} + \frac{n}{\sigma_2^2}\right)^{-1}$$

Now, $\alpha_2 = \left(\frac{1}{\sigma_0^2}\mu_0 + \frac{n}{\sigma_1^2}\bar{x}\right) \Big/ \left(\frac{1}{\sigma_0^2} + \frac{n}{\sigma_1^2}\right)$ uses the expectation of posterior $h_1(\theta|\bar{x}_2)$ as the estimation of θ. Because $\alpha_1 = \left(\frac{1}{\sigma_0^2}\mu_0 + \frac{n}{\sigma_1^2}\overline{x}_1\right) \cdot d_1^2$, we have

$$\begin{aligned}\alpha_2 &= \left(\frac{1}{d_1^2}\alpha_1 + \frac{n}{\sigma_2^2}\bar{x}_2\right) \cdot d_2^2 = \left(\frac{1}{\sigma_0^2}\mu_0 + \frac{n}{\sigma_1^2}\bar{x}_1 + \frac{n}{\sigma_2^2}\bar{x}_2\right) \cdot d_2^2 \\ &= \left(\frac{1}{\sigma_0^2}\mu_0 + \frac{n}{\sigma_1^2}\bar{x}_1\right) \cdot d_2^2 + \frac{n}{\sigma_2^2}\bar{x}_2 \cdot d_2^2, \end{aligned} \tag{4.9}$$

and $\frac{n}{\sigma_2^2} > 0$, so

$$d_2^2 = \left(\frac{1}{d_1^2} + \frac{n}{\sigma_2^2}\right)^{-1} = \left(\frac{1}{\sigma_0^2} + \frac{n}{\sigma_1^2} + \frac{n}{\sigma_2^2}\right)^{-1} < d_1^2 = \left(\frac{1}{\sigma_0^2} + \frac{n}{\sigma_1^2}\right)^{-1}.$$

In α_2, $\left(\frac{1}{\sigma_0^2}\mu_0 + \frac{n}{\sigma_1^2}\overline{x}_1\right) \cdot d_2^2 < \alpha_1$. It is clear that because of the addition of the new sample, the proportion of the original prior and the old sample declines. According to Equation (4.9), with the continuous increase of new sample (here we assume the sample size keeps invariant), we have

$$\begin{aligned}\alpha_m &= \left(\frac{1}{\sigma_0^2}\mu_0 + \frac{n}{\sigma_1^2}\bar{\bar{x}}_1 + \frac{n}{\sigma_2^2}\bar{x}_2 + \cdots + \frac{n}{\sigma_m^2}\bar{x}_m\right) \cdot d_m^2 \\ &= \left(\frac{1}{\sigma_0^2}\mu_0 + \sum_{k=1}^{m} \frac{n}{\sigma_k^2}\bar{x}_k\right) \cdot d_m^2 \qquad (k = 1, 2, \ldots, m). \end{aligned} \tag{4.10}$$

From Equation (4.10), if the variance of the new samples are the same, they equal a sample with the size $m \times n$. The above process weighted all the sample means with their precisions. The higher the precision, the bigger the weight. If the prior distribution is estimated precisely, we can use less sample data and only need a little computation. This is especially useful in the situation where the sample is hard to collect. It is also the point where the Bayesian approach outperforms other methods. Therefore, the determination of prior distribution in Bayesian learning is extremely important. If there is no prior information and we adopt non-information prior, with the increase of sample, the effect of the sample will become more and more salient. If the noise of the sample is small, the posterior will be increasingly close to its true value. The only issue is that significant computation is required.

4.3.3 *Steps in Bayesian Problem Solving*

The steps in Bayesian problem solving can be summarized as follows:

(1) Define the random variable. Set unknown parameters as random variable or vector, for short denoted as θ. The joint density $p(x_1, x_2, \ldots, x_n; \theta)$ of the sample $x_1, x_2, \ldots, x_n$ is regarded as the conditional density of $x_1, x_2, \ldots, x_n$ with respect to θ, denoted briefly as $p(x_1, x_2, \ldots, x_n|\theta)$ or $p(D|\theta)$.
(2) Determine the prior distribution density $p(\theta)$. Use conjugate distribution. If there is no information about prior distribution, then use Bayesian assumption of non-information prior distribution.
(3) Calculate posterior distribution density via Bayesian theorem.
(4) Make inference of the problem with the resulting posterior distribution density.

Take the case of a single variable and single parameter as an example. Consider the problem of thumbtack throwing. If we throw a thumbtack up in the air, the thumbtack will fall down and reset at one of two states: on its head or on its tail. Suppose we flip the thumbtack $N + 1$ times, from the first N observations, how can we get the probability of the case "head" in the $N + 1$th throw?

Step 1. Define a random variable Θ. The value θ corresponds to the possible value of the real probability of head. The density function $p(\theta)$ represents the uncertainty of Θ. The variable of the ith result is X_i ($i = 1, 2, \ldots, N + 1$), and the set of observations is $D = \{X_1 = x_1, \ldots, X_n = x_n\}$. Our objective is to calculate $p(x_{N+1}|D)$.

Step 2. According to Bayesian theorem, we have, $p(\theta|D) = \frac{p(\theta)p(D|\theta)}{p(D)}$, where $p(D) = \int p(D|\theta)p(\theta)d\theta$, $p(D|\theta)$ is the binary likelihood function of sample. If θ, the value of Θ, is known, the observation value in D is independent; the probability

of head (tail) is θ, the probability of tail is $(1-\theta)$, then

$$p(\theta|D) = \frac{p(\theta)\theta^h(1-\theta)^t}{p(D)}, \tag{4.11}$$

where h and t are the times of head and tail in the observation D, respectively. They are sufficient statistics of sample binary distribution.

Step 3. Seek the mean of Θ as the probability of case head in the $N+1$th toss:

$$\begin{aligned} p(X_{N+1} = \text{heads}|D) &= \int p(X_{N+1} = \text{heads}|\theta)p(\theta|D)d\theta \\ &= \int \theta \cdot p(\theta|D)d\theta \equiv E_{p(\theta|D)}(\theta), \end{aligned} \tag{4.12}$$

where $E_{p(\theta|D)}(\theta)$ is the expectation of θ under the distribution $p(\theta|D)$.

Step 4. Assign prior distribution and supper parameters for Θ.
The common method for prior assignment is to assume prior distribution first, and then to determine proper parameters. Here, we assume the prior distribution is Beta distribution:

$$p(\theta) = \text{Beta}(\theta|\alpha_h, \alpha_t) \equiv \frac{\Gamma(\alpha)}{\Gamma(\alpha_h)\Gamma(\alpha_t)}\theta^{\alpha_h-1}(1-\theta)^{\alpha_t-1}, \tag{4.13}$$

where $\alpha_h > 0$ and $\alpha_t > 0$ are parameters of Beta distribution, $\alpha = \alpha_h + \alpha_t$, and $\Gamma(\cdot)$ is the Gamma function. To distinguish from parameter θ, σh and α_t are called "Supper Parameters". Because Beta distribution belongs to the conjugate family, the resulting posterior is also Beta distribution:

$$\begin{aligned} p(\theta|D) &= \frac{\Gamma(\alpha+N)}{\Gamma(\alpha_h+h)\Gamma(\alpha_t+t)}\theta^{\alpha_h+h-1}(1-\theta)^{\alpha_t+t-1} \\ &= \text{Beta}(\theta|\alpha_h+h, \alpha_t+t). \end{aligned} \tag{4.14}$$

To this distribution, its expectation of θ has a simple form as follows:

$$\int \theta \cdot \text{Beta}(\theta|\alpha_h, \alpha_t)d\theta = \frac{\alpha_h}{\alpha}. \tag{4.15}$$

Therefore, for a given Beta prior, we get the probability of head in the $N+1$th toss as follows:

$$p(X_{N+1} = \text{heads}|D) = \frac{\alpha_h + h}{\alpha + N}. \tag{4.16}$$

There are many ways to determine the supper parameters of the prior Beta distribution $p(\theta)$, such as imagined future data and equivalent samples. Other methods

can be found in the works of Winkler, Chaloner, and Duncan. In the method of imagined future data, two equations can be deduced from Equation (4.16), and two supper parameters α_h and α_t can be solved accordingly.

In the case of single variable multiple parameters (a single variable with multiple possible states), commonly X is regarded as a continuous variable with Gaussian distribution. Assume its physical density is $p(x|\theta)$, then we have

$$p(x|\theta) = (2\pi v)^{-1/2} e^{-(x-\mu)^2/2v^2},$$

where $\theta = \{\mu, v\}$.

Similar to the previous approach on binary distribution, we first assign the prior of parameters and then solve the posterior with the data $D = \{X_1 = x_1, X_2 = x_2, \ldots, X_N = x_N\}$ via Bayesian theorem:

$$P(\theta|D) = p(D|\theta)p(\theta)/p(D).$$

Next, we use the mean of Θ as the prediction as follows:

$$p(x_{N+!}|D) = \int p(x_{N+1}|\theta)p(\theta|D)d\theta. \tag{4.17}$$

For an exponential family, the computation is effective and close. In the case of multi samples, if the observed value of X is discrete, Dirichlet distribution can be used as the prior distribution, which can simplify the computation.

The computational learning mechanism of Bayesian theorem is to get the weighted average of the expectation of prior distribution and the mean of sample, where the higher the precision, the bigger the weight. Under the precondition that the prior is conjugate distribution, posterior information can be used as the prior in the next round of computation, so that it can be integrated with further obtained sample information. If this process is repeated time and again, the effect of the sample will be increasingly prominent. Because Bayesian method integrates prior information and posterior information, it can both avoid the subjective bias when using only prior information and avoid numerous blind searching and computation when sample information is limited. Besides, it can also avoid the affect of noise when utilizing only posterior information. Therefore, it is suitable for problems of data mining with statistical features and problems of knowledge discovery, especially the problems where the sample is hard to collect or the cost of collecting the sample is high. The key of effective learning with Bayesian method is determining the prior reasonably and precisely. Currently, there are only some principles for prior determination, and there is no operable whole theory to determine priors. In many cases, the reasonability and precision of prior distribution is hard to evaluate. Further research is required to solve these problems.

4.4 Naïve Bayesian Learning Model

In naïve Bayesian learning models, the training sample I is decomposed into feature vector $\boldsymbol{X}$ and decision class variable C. Here, it is assumed that all the weights in a feature vector are independent given the decision variable. In other words, each weight affects the decision variable independently. Although the assumption to some extent limits the sphere of the naïve Bayesian model, in practical applications, naïve Bayesian model can both exponentially reduce the complexity for model construction and express striking robustness and effectiveness even when the assumption is unsatisfied (Nigam *et al.*, 1998). It has been successfully applied in many data mining tasks, such as classification, clustering, model selection, and so on. Currently, many researchers are working to relax the limitation of independence among variables (Heckerman, 1997) so that the model can be applied more widely.

4.4.1 *Naïve Bayesian Learning Model*

Bayesian theorem tells us how to predict the class of the incoming sample given training samples. The rule of classification is maximum posterior probability, which is given in the following equation:

$$P(C_i|A) = P(C_i) * P(A|C_i)/P(A). \tag{4.18}$$

Here, A is a test sample to be classified and $P(Y|X)$ is the conditional probability of Y under the condition of X. The probabilities at the right side of the equation can be estimated from the training data. Suppose that the sample is represented as a vector of features. If all features are independent for the given classes, $P(A|C_i)$ can be decomposed as a product of factors: $P(a_1|C_i) \times P(a_2|C_i) \times \cdots \times P(a_m|C_i)$, where a_i is the ith feature of the test sample. Accordingly, the posterior computation equation can be rewritten as follows:

$$P(C_i|A) = \frac{P(C_i)}{P(A)} \prod_{j=1}^{m} P(a_j|C_i). \tag{4.19}$$

The entire process is called naïve Bayesian classification. In the common sense, only when the independent assumption holds, or when the correlation of features is very weak can the naïve Bayesian classifier achieve the optimal or sub-optimal result. Yet, the strong limited condition seems inconsistent with the fact that the naïve Bayesian classifier gains striking performance in many fields, including some fields where there is obvious dependence among features. In 16 out of the total 28 data sets of UCI, naïve Bayesian classifier outperforms the C4.5 algorithms and

has similar performance with that of CN2 and PEBLS. Some research works report similar results (Clark and Niblett, 1989). At the same time, researchers have also successfully proposed some strategy to relax the limitation of independence among features (Nigam *et al.*, 1998).

The conditional probability in formula (4.19) can be gained using maximum likelihood estimation:

$$P(v_j|C_i) = \frac{\text{count}(v_j \wedge c_i)}{\text{count}(c_i)}. \tag{4.20}$$

To avoid zero probability, if the actual conditional probability is zero, it is assigned to be $0.5/N$, where N is the total number of examples.

Suppose that there are only two classes, namely class 0 and class 1, and $a_1, \ldots, a_k$ represent features of test set. Let $b_0 = P(C = 0)$, $b_1 = P(C = 1) = 1 - b_0$, $p_{j0} = P(A_j = a_j|C = 0)$, $p_{j1} = P(A_j = a_j|C = 0)$, then

$$p = P(C = 1|A_1 = a_1 \wedge \cdots \wedge A_k = a_k) = \left(\prod_{j=1}^{k} p_{j1}\right) b_1/z, \tag{4.21}$$

$$q = P(C = 0|A_1 = a_1 \wedge \cdots \wedge A_k = a_k) = \left(\prod_{j=1}^{k} p_{j0}\right) b_0/z, \tag{4.22}$$

where z is a constant. After taking logarithm on both sides of the above two equations, we subtract the second equation from the first one and get

$$\log\ p - \log\ q = \left(\sum_{j=1}^{k} \log\ p_{j1} - \log\ p_{j0}\right) + \log\ b_1 - \log\ b_0. \tag{4.23}$$

Here, let $w_j = \log\ p_{j1} - \log\ p_{j0}$, $b = \log\ b_1 - \log\ b_0$, then the above equation is written as:

$$\log\ (1 - p)/p = -\sum_{j=1}^{k} w_j - b. \tag{4.24}$$

After taking exponential on both sides of Equation (4.24) and rearranging, we have

$$p = \frac{1}{1 + e^{-\sum_{j=1}^{k} w_j - b}}. \tag{4.25}$$

To calculate this value, we assume that feature A_j has $v(j)$ possible values. Let

$$w_{jj'} = \log P(A_j = a_{jj'}|c = 1)$$
$$\log P(A_j = a_{jj'}|c = 0)(1 \leq j' \leq v(j)). \tag{4.26}$$

We have

$$P(C(x)=1)=\frac{1}{1+e^{-(\sum_{j=1}^{k}\sum_{j'}^{v(j)}I(A_j(x)=a_{jj'})w_{jj'}-b}}, \tag{4.27}$$

where I is a characteristic function. If φ is true, then $I(\varphi)=1$; else $I(\varphi)=0$. In practical computation, Equation (4.27) can be calculated similar to Equation (4.20).

In fact, Equation (4.27) is a perception function with a sigmoid activation function. The input of this function is the possible value of all features. So, to some extent, naïve Bayesian classifier is equal to a perception model. Further research demonstrated that naïve Bayesian classifier can be generalized to logical regression with numerical features.

Consider Equation (4.20). If A_j takes discrete values, count$(A_j = a_j \wedge C = c_i)$ can be calculated directly from the training samples. If A_j is continuous, it should be discretized. In unsupervised discretization, a feature is discretized into M equally wide sections, where $M = 10$ commonly. We can also utilize a more complicated discretization method, such as supervised discretization method.

Let each A_j be a numerical feature (discrete or continuous). The logical regression model is given as follows:

$$\log\frac{P(C=1|A_1=a_1,\ldots,A_k=a_k)}{P(C=0|A_1=a_1,\ldots,A_k=a_k)}=\sum_{j=1}^{k}b_ja_j+b_0. \tag{4.28}$$

After transforming similar to that of Equation (4.24), we have

$$p=\frac{1}{1+e^{-\sum_{j=1}^{k}b_ja_j-b_0}}. \tag{4.29}$$

Obviously, this is also a perception function with a sigmoid activation function. Its inputs are all feature values using function $f_j(\varphi)$ to replace b_ja_j. If the sphere of A_j is divided into M parts and the ith part is $[c_{j(i-1)},\ c_{ji}]$, the function $f_j(\varphi)$ is given as follows:

$$b_ja_j=f_j(a_j)=\sum_{i=1}^{M}b_{ji}I(c_{j(i-1)}<a_j\le c_{ji}), \tag{4.30}$$

where b_{ji} is a constant. According to Equations (4.29) and (4.30), we have

$$P(C(x)=1)=\frac{1}{1+e^{-(\sum_{j=1}^{k}\sum_{i}^{M}b_{ji}I(c_{j(i-1)}<a_j\le c_{ji}))-b_0}}. \tag{4.31}$$

This is the final regression function. So, naïve Bayesian classifier is a non-parametric and nonlinear extension of logical regression. By setting $b_{ji}=(c_{j(i-1)}+c_{ij})/(2b_j)$, we can get a standard logical regression formula.

4.4.2 *Boosting of Naïve Bayesian Model*

In boosting, a series of classifiers will be built, and in each classifier in the series misclassified by previous classifier will be given more attention. Concretely, after learning classifier k, the weights of training examples that are misclassified by classifier k will increase, and classifier $k+1$ will be learnt based on the newly weighted training examples. This process will be repeated T times. The final classifier is the synthesis of all the classifiers in series.

Initially, each training example is set with a weight. In the learning process, if some example is misclassified by one classifier, in the next learning round, the corresponding weight will be increased, so that the next classifier will pay more attention to it.

The boosting algorithm for binary classification problem is given by Freund and Schapire as the AdaBoost Algorithm (Freund and Schapire, 1995).

Algorithm 4.1 (AdaBoost Algorithm).

Input:
N training examples $< (x_1\ ,\ y_1\ >, \cdots (x_N, y_N) >$
Distribution of the N training examples, D: w, where w is the weight vector of training example.
T: the number of rounds for training.

1. Initialize:
2. Initial weight vector of training examples: $w_i\ = 1/N\ i = 1, \ldots, N$
3. for $t = 1$ to T
4. Given weights ${w_i}^t$, find a hypothesis $H^{(t)} : X \rightarrow [0, 1]$
5. Estimation the general error of hypothesis $H^{(t)}$:
   $e^{(t)} = \sum_{i=1}^{N} {w_i}^{(t)} \mid y_i\ -\ {h_i}^{(t)}\ (x_i)|$
6. Calculate $\beta^{(t)} = e^{(t)}/(1 - e^{(t)})$
7. Renew the next round weights of examples with
   ${w_i}^{(t+1)}\ =\ {w_i}^{(t)}\ (\beta^{(t)})^{1\ -\ |y_i\ -\ h_i{}^{(t)}(x_i)|}$
8. Normalize ${w_i}^{(t+1)}$, so that they are summed up to 1
9. End for
10. Output

$$h(x) = \begin{cases} 1 & \text{if } \sum_{t=1}^{T} \left(\log \frac{i}{\beta^{(t)}}\right) h^{(t)}(x) \geq \frac{1}{2} \sum_{t=1}^{T} \left(\log \frac{i}{\beta^{(t)}}\right), \\ 0 & \text{otherwise.} \end{cases}$$

Here we assume that all the classifiers are effective. In other words for each classifier, the examples correctly classified are more than the ones misclassified, $e^t < 0.5$. Hence, $\beta^{(t)} < 1$. When $|y_i - h_i^{(t)}(x_i)|$ increases, $w_i^{(t+1)}$ will increase accordingly. The algorithm fulfills the idea of boosting.

Some notes on the algorithm:

(1) $h^{(t)}(x)$ is calculated via the output formula, and the result is either 0 or 1.
(2) The calculation of conditional probability $P(A_j = a_{jj'}|C = c)$ in formula (4.20). If we do not consider the weights, the computational basis for *count*(*condition*) is 1. For example, if there are k examples satisfying the condition, then *count*(*condition*) $= k$. If we consider the weights, the computational basis for each example is its weight. For example, if there are k examples satisfying the condition, then $count(condition) = \sum_i^k w_i$. In this case, the adjustment of weights embodies the idea of boosting.
(3) The output of the algorithm means that for an incoming input x, according to Step 6 in the algorithm, we can use the result of learning to generate the output by voting.

The final combined hypothesis can be defined as

$$H(x) = \frac{1}{1 + \prod_{t=1}^{T} (\beta^{(t)})^{2r(x)-1}},$$

where $r(x) = \frac{\sum_{t=1}^{T} (\log 1/\beta^t) H^{(t)}(x)}{\sum_{t=1}^{T} (\log 1/\beta^t)}$.

Below we will demonstrate that after boosting the representative capability of combined naïve Bayesian classifier equals to that of multiple layered perception model with one hidden layer. Let $\alpha = \prod_{t=1}^{T} \beta^t$ and $v^{(t)} = \log \beta^{(t)} / \log \alpha$, then

$$H(x) = \frac{1}{1 + \alpha^{2(\sum_{t=1}^{T} v^{(t)} H^{(t)}(x)) - 1}} = \frac{1}{1 + e^{\sum_{t=1}^{T} 2 \log \beta^{(t)} H^{(t)}(x) - \sum_{t=1}^{T} \log \beta^{(t)}}}.$$

The output of the combined classifier is the output of a sigmoid function, which takes the outputs of single classifiers and their weights as its parameters. Since a naïve Bayesian classifier equals a perception machine, the combined classifier equals a perception network with a hidden layer.

The boosting naïve Bayesian method for multiple classification problems is as follows:

Algorithm 4.2 (Multiple Classification AdaBoost Algorithm).

Input:
N training examples $< (x_1, y_1) >, \cdots < (x_N, y_N) >$

Distribution of the N training examples, D: w, where w is the weight vector of training example.

T: the number of rounds for training.

1. Initialize:
   Initial weight vector of training examples $w_i = 1/N, \quad i = (1, \ldots, N)$
2. *for* $t = 1$ *to* T
3. Given weights ${w_i}^t$, find a hypothesis $H^{(t)} : X \to Y$
4. Estimation the general error of hypothesis $H^{(t)}$:
   $e^{(t)} = \sum_{i=1}^{N} {w_i}^{(t)} I(y_i \neq {h_i}^{(t)}(x_i))$
5. Calculate $\beta^{(t)} = e^{(t)}/(1 - e^{(t)})$
6. Renew the next round weights of examples with
   ${w_i}^{(t+1)} = {w_i}^{(t)} (\beta^{(t)})^{1 - I(y_i = {h_i}^{(t)}(x_i))}$
7. Normalize ${w_i}^{(t+1)}$, so that they are summed up to 1
8. End for
9. Output:

$$h(x) = \arg \max_{y \in Y} \sum_{t=1}^{T} \left(\log \frac{i}{\beta^{(t)}} \right) I(h^{(t)}(x) = y)$$

where $I(\phi) = 1$ if $\phi = T$; $I(\phi) = 0$ otherwise.

4.4.3 *The Computational Complexity*

Suppose a sample in the sample space has f features, and each feature takes v values. The naïve Bayesian classifier deduced by Formula (4.27) will have $fv + 1$ parameters. These parameters are accumulatively learnt $2fv + 2$ times. In each learning process, each feature value of each training example will improve the final precision. So, the time complexity for n training examples is O(nf), independent of v. Substantially, this time complexity is optimal. For boosting the naïve Bayesian classifier, the time complexity of each round is O(nf). T round training corresponds to O(Tnf). Notice that T is a constant. So the entire time complexity is still O(nf).

For naïve Bayesian classifier, the primary computation is counting. Training examples can be processed either sequentially or in batches from disk or tape. So this method is perfectly suited for knowledge discovery on a large data set. The training set is not necessarily loaded to memory entirely, and part of it can be kept in disks or tapes. Yet, the boosting naïve Bayesian model also has the following problems:

(1) From the idea of boosting, when noise exists in the training set, the boosting method will take it as useful information and amplify its effect with a large weight. This will reduce the performance of boosting. If there are many noisy data, boosting will lead to worse result.
(2) Although theoretically boosting can achieve zero error rate for the training set, in its practical application of naïve Bayesian model, the 0 classification error seen in the training set is generally hardly guaranteed.

4.5 Construction of a Bayesian Network

4.5.1 *Structure of a Bayesian Network and Its Construction*

In short, Bayesian network is a directed acyclic graph with probabilistic notes. The graphic model can be utilized to represent the (physical or Bayesian) joint distribution of a large variable set. It can also be used to analyze correlations among numerous variables. With the capability of learning and statistical reasoning under the Bayesian theorem, it can fulfill many data mining tasks, such as prediction, classification, clustering, casual analysis, and so on.

Given a series of variables $\boldsymbol{X} = \{x_1, x_2, \ldots, x_n\}$, a Bayesian network is composed of two components: one is network structure S, which represents the conditional independence among variables $\boldsymbol{X}$; the other is local distribution set P, which is related to every variable. The two components define the joint probability of $\boldsymbol{X}$. S is a directed acyclic graph. Nodes in S and variables in $\boldsymbol{X}$ are one to one correspondingly. Let x_i be a variable or node and $\boldsymbol{Pa}_\mathrm{i}$ be the parent nodes of x_i in S. The absence of an arc between the nodes usually represents conditional independence. The joint probability of X is represented as

$$p(\boldsymbol{X}) = \prod_{i=1}^{n} p(xi|\boldsymbol{Pa}_i), \tag{4.32}$$

where $p(x_i|\boldsymbol{Pa_i})(i = 1, 2, \ldots, n)$ is the local probabilistic distribution in formula (4.32). The pair (S, P) represents the joint probabilistic distribution $p(X)$. If a Bayesian network is constructed merely based on prior information, the probabilistic distribution is Bayesian, or subjective. If the Bayesian network is constructed purely based on data, the distribution is physical, or objective.

To construct a Bayesian network, we should do the following tasks.

Step 1. Determine all the related variables and their explanations. To do so, we need to (a) determine the objective of the model, or make a reasonable explanation of given problem; (b) find as many as possible problem-related observations, and determine

a subset that is worthy of constructing model; (c) translate these observations into mutually exclusive and exhaustive state variables. The result of these operations is not unique.

Step 2. Construct a directed acyclic graph which expresses a conditional independent assertion. According to the multiplication formula, we have

$$\begin{aligned} p(\boldsymbol{X}) &= \prod_{i=1}^{n} p(x_i|x_1, x_2, \ldots, x_{i-1}) \\ &= p(x_1)p(x_2|x_1)p(x_3|x_1, x_2)\ldots p(x_n|x_1, x_2, \ldots, x_{n-1}). \end{aligned} \tag{4.33}$$

For any variable $\boldsymbol{X}$, if there is a subset $\pi_i \subseteq \{x_1, x_2, \ldots, x_{i-1}\}$, then x_i and $\{x_1, x_2, \ldots, x_{i-1}\}\backslash\pi_i$ are conditional independent. That is, for any given $\boldsymbol{X}$, the following equation holds:

$$p(x_i|x_1, x_2, \ldots, x_{i-1}) = p(x_i|\pi_i), \quad (i = 1, 2, \ldots, n). \tag{4.34}$$

According to Formulas (4.33) and (4.34), we have $p(x) = \prod_{i=1}^{n} p(x_i|\pi_i)$. The variable set $(\pi_1, \ldots, \pi_n)$ corresponds to the parent set $(\boldsymbol{Pa}_1, \ldots, \boldsymbol{Pa}_n)$. So the above equation can also be written as $p(X) = \prod_{i=1}^{n} p(x_i|Pa_i)$. To determine the structure of the Bayesian network, we need to (a) sort variables $x_1, x_2, \ldots, x_i$; and (b) determine the variable set $(\pi_1, \ldots, \pi_n)$ that satisfies formula (4.34).

Theoretically, finding a proper conditional independent sequence from n variables is a combination explosion problem, for it will require comparison among $n!$ different sequences. In practice, casual relation is often used to solve this problem. Generally, casual relation will correspond to a conditional independent assertion. So we can find a proper sequence by adding arrowed arcs from reason variables to result variables.

Step 3. Assign local probabilistic distribution $p(x_i|\boldsymbol{Pa_i})$. In the discrete case, we need to assign a distribution for each variable on each state of its parent nodes.

Obviously, the steps above may be intermingled but not purely performed in sequence.

4.5.2 *Probabilistic Distribution of Learning the Bayesian Network*

Consider the following problem: given the structure of a Bayesian network, how can we learn the probabilistic distribution, or how can we update its original prior, based on observed data? Here we use Bayesian approach, which integrates prior knowledge and data to improve existing knowledge. This technique can be applied to data

mining. Assume that the physical joint distribution of variables $\boldsymbol{X} = (x_1, x_2, \ldots, x_n)$ can be coded in some network structure S:

$$P(x|\boldsymbol{\theta}_s, S^h) = \prod_{i=1}^{n} p(x_i|\boldsymbol{Pa_i}, \boldsymbol{\theta}_i, S^h), \tag{4.35}$$

where $\boldsymbol{\theta_i}$ is the parameter vector of distribution $p(x_i|\boldsymbol{Pa}_i, \boldsymbol{\theta}_i, S^h)$; $\boldsymbol{\theta_s}$ is the vector of parameter groups $(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \ldots, \boldsymbol{\theta}_n)$; and S^h is the hypothesis that physical joint distribution can be decomposed in accordance with structure S. It is noted that the decomposition is not cross, or overlapped. For example, given $\boldsymbol{X} = \{x_1, x_2\}$, any joint distribution of $\boldsymbol{X}$ can be decomposed to a no-arc network or a network with the only arc $x_1 \rightarrow x_2$. This is cross or overlapped. Besides, suppose we generate a random sample $D = \{x_1, \ldots, x_n\}$ based on the physical distribution of $\boldsymbol{X}$. An element x_i of D represents an observed value of the sample and is called a case. We define a vector-valued variable Θ_S corresponding to parameter vector $\boldsymbol{\theta}_s$ and assign a prior density function $p(\boldsymbol{\theta}_s|S^h)$ to represent the uncertainty of Θ_S. Then the probability learning of the Bayesian network is described as: given a random sample D, calculated the posterior $p(\boldsymbol{\theta}_s|D, S^h)$.

Below, we use unrestricted multinomial distribution to discuss the basic idea of probability learning. Assume that each variable $x_i \in X$ is discrete and has r_i possible values $x_i^1, x_i^2, \ldots, x_i^{r_i}$. Each local distribution function is a set of multinomial distributions, each of which corresponds to a composition of $\boldsymbol{Pa_i}$. That is to say, let

$$\begin{aligned} &p(x_i^k|\boldsymbol{pa}_i^j, \boldsymbol{\theta}_i, S^h) = \theta_{ijk} > 0 \\ &i = 1, 2, \ldots, n; \quad j = 1, 2, \ldots, q_i; \quad k = 1, 2, \ldots, r_i, \end{aligned} \tag{4.36}$$

where $\boldsymbol{pa}_i^1, \boldsymbol{pa}_i^2, \ldots, \boldsymbol{pa}_i^{q_i}$ represent the composition of $\boldsymbol{Pa}_i$; $q_i = \prod_{Xi \in Pa_i} ri$; $\theta_i = ((\theta_{ijk})_{k=2}^{r_i})_{j=1}^{q_i}$ is parameter; θ_{ij1} is not included for $\theta_{ij1} = 1 - \sum_{k=2}^{r_i} \theta_{ijk}$ can be calculated from other parameters. For convenience, we define the parameter vector as follows:

$$\theta_{ij} = (\theta_{ij2}, \theta_{ij3}, \ldots, \theta_{ijr_i}), \quad (i = 1, 2, \ldots, n; \quad j = 1, 2, \ldots, q_i).$$

Given the local distribution functions above, we still require two assumptions to make the calculation of the posterior $p(\theta_s|D, S^h)$ close:

(1) There is no missing data in sample D, or D is complete;
(2) Parameter vectors are mutually independent, viz. $p(\theta_s|S^h) = \prod_{i=1}^{n} \prod_{j=1}^{q_i} p(\theta_{ij}|S^h)$. This is called parameter independence.

Under the above assumptions, for a given random sample D, parameters are independent:

$$p(\boldsymbol{\theta}_s|D, S^h) = \prod_{i=1}^{n}\prod_{j=1}^{q_i} p(\boldsymbol{\theta}_{ij}|D, S^h). \tag{4.37}$$

Then we can update each parameter vector θ_{ij} independently. Suppose each parameter θ_{ij} has the prior distribution of Dirichlet distribution $\mathrm{Dir}(\boldsymbol{\theta}_{ij}|\alpha_{ij1}, \alpha_{ij2}, \ldots, \alpha_{ijr_i})$, we get the following posterior distribution:

$$p(\boldsymbol{\theta}_{ij}|D, S^h) = \mathrm{Dir}(\boldsymbol{\theta}_{ij}|\alpha_{ij1} + N_{ij1}, \quad \alpha_{ij2} + N_{ij2}, \ldots, \alpha_{ijr_i} + N_{ijr_i}), \tag{4.38}$$

where N_{ijk} is the number of cases in D that satisfy $X_i = x_i^k$ and $\boldsymbol{Pa}_i = \boldsymbol{pa}_i^j$.

Now we can make an interesting prediction by seeking the mean of possible $\boldsymbol{\theta}_s$. For example, for the $N + 1$th case,

$$p(X_{N+1}|D, S^h) = \underset{p(\theta s|D,S^h)}{E}\left(\prod_{i=1}^{r_i}\theta_{ijk}\right).$$

According to the parameter independence given D, we can calculate the expectation as follows:

$$\begin{aligned} p(x_{N+1}|D, S^h) &= \int \prod_{i=1}^{n} \theta ijk \; p(\boldsymbol{\theta}_s|D, S^h)\mathrm{d}\boldsymbol{\theta} \\ &= \prod_{i=1}^{n} \int \theta ijk \; p(\boldsymbol{\theta}_{\mathrm{ij}}|D, S^h)\mathrm{d}\boldsymbol{\theta}_{ij}, \end{aligned}$$

and finally get

$$p(x_{N+1}|D, S^h) = \prod_{i=1}^{n} \frac{\alpha_{ijk} + N_{ijk}}{\alpha_{ij} + N_{ij}}, \tag{4.39}$$

where $\alpha_{ij} = \sum_{k=1}^{r_i} \alpha_{ijk}$ and $N_{ij} = \sum_{k=1}^{r_i} N_{ijk}$. Because unrestricted multinomial distribution belongs to the exponential family, the above computation is rather easy.

A Bayesian network with respect to variables $\boldsymbol{X}$ represents the joint distribution of $\boldsymbol{X}$. So, no matter whether a Bayesian network is constructed from prior knowledge, or data, or integration of them, in principle, it can be used to deduce any interested probability. Yet, the precise or even approximately precise reasoning on a Bayesian network with discrete variables is NP hard. The current solution is to simplify computation based on some conditional independence, to construct a simple network topology for some specific reasoning problem, or to simplify the

network structure at the cost of less precision loss. Even though it often requires considerable computation to construct a Bayesian network, for some problems, such as naïve Bayesian classification, using conditional independence can largely reduce computation without loss of much precision.

When sample data are incomplete, except for some special cases, we need to borrow an approximation method, such as Monte Carlo method, Gaussian approximation, EM algorithm to find maximum likelihood (ML) or maximum *a posteriori* (MAP), and so on. Although these algorithms are mature, the computational cost is large.

4.5.3 *Structure of Learning the Bayesian Network*

When the structure of a Bayesian network is undetermined, it is possible to learn both the network structure and the probabilities from data. Because in data mining, there are huge amount of data, and it is hard to tell the relation among variables, the structure learning problem is practically meaningful.

The network structure that represents the physical joint probability of $\boldsymbol{X}$ is improvable. According to Bayesian approach, we define a discrete variable to represent the uncertainty of network structure. The states of the variable correspond to all possible network structure hypotheses S^h. We set its prior as $p(S^h)$. For a given random sample D, which comes from the physical distribution of $\boldsymbol{X}$, we calculate the posterior probability $p(S^h|D)$ and $p(\boldsymbol{\theta}_S|D, S^h)$, where $\boldsymbol{\theta}_S$ is a parameter vector. Then we use these posteriors to calculate the interested expectations.

The computation of $p(\boldsymbol{\theta}_S|D, S^h)$ is similar to what we have illustrated in the previous section. The computation of $p(S^h|D)$ is theoretically easy. According to Bayesian theorem, we have

$$p(S^h|D) = p(S^h, D)/p(D) = p(S^h)p(D|S^h)/p(D), \tag{4.40}$$

where $p(D)$ is a structure-independent normalizing constant and $p(D|S^h)$ is the marginal likelihood. To determine the posterior of the network structure, we need to only calculate marginal likelihood for each possible structure.

Under the precondition of unrestricted multinomial distribution, parameter independence, Dirichlet prior, and complete data, the parameter vector $\boldsymbol{\theta}_{ij}$ can be updated independently. The marginal likelihood of data is exactly the multiplication of marginal likelihoods of each $i-j$ pair:

$$p(D|S^h) = \prod_{i=1}^{n}\prod_{j=1}^{q_i}\frac{\Gamma(\alpha_{ij})}{\Gamma(\alpha_{ij}+N_{ij})}\cdot\prod_{k=1}^{r_i}\frac{\Gamma(\alpha_{ijk}+N_{ijk})}{\Gamma(\alpha_{ijk})}. \tag{4.41}$$

This formula was originally proposed by Cooper and Herskovits in 1992.

In common cases, the number of possible Bayesian networks with n variables is larger than exponential in n. It is intractable to exclude these hypotheses. Two approaches can be used to handle this problem, namely model selection and selective model averaging. The former approach is to select a "good" model from all the possible models (structure hypotheses) and use it as the correct model. The latter approach is to select a reasonable number of "good" models from all the possible models and pretend that these models are exhaustive. The questions are as follows: how to decide whether a model is "good" or not? How to search "good" models? Can a precise result can be obtained when these approaches are applied to Bayesian structure? There are some different definitions and corresponding computational methods about a "good" model. The last two questions are hardly capable of being answered theoretically. Some research work had demonstrated that using greedy algorithm to select a single good model often leads to precise prediction (Chickering and Heckerman, 1996). Applying the Monte Carlo method to perform selective model averaging is sometime effective as well. It may even result in better prediction. These results are somewhat largely responsible for the great deal of recent interest in learning with Bayesian network.

In 1995, Heckerman pointed out that under the precondition of parameter independence, parameter modularity, likelihood equivalence, and so on, the methods for learning Bayesian non-casual network can be applied to learning the casual network. In 1997, he suggested that under casual Markov condition, the casual relationship could be deduced from conditional independence and conditional correlation (Heckerman, 1997). This makes it possible for the corresponding effect to be predicted when interference is seen.

Below is a case study in which Heckerman *et al.* used Bayesian network to perform data mining and knowledge discovery. The data came from 10,318 Wisconsin high school seniors. Each student was described by the following variables and corresponding states

- Sex (SEX): male, female;
- Socioeconomic status (SES): low, lower middle, upper middle, high;
- Intelligence quotient (IQ): low, lower middle, upper middle, high;
- Parental encouragement (PE): low, high;
- College plans (CP): yes, no.

Our goal here is to discover the factors that affect the intention of high school seniors to attend college or to understand the possibly causal relationships among these variables. Data are described by the sufficient statistics in Table 4.3. In this

Table 4.3. Sufficient statistics

(Male)	4	349	13	64	9	207	33	72	12	126	38	54	10	67	49	43
	2	232	27	84	7	201	64	95	12	115	93	92	17	79	119	59
	8	166	47	91	6	120	74	110	17	92	148	100	6	42	198	73
	4	48	39	57	5	47	132	90	9	41	224	65	8	17	414	54
(Female)	5	454	9	44	5	312	14	47	8	216	20	35	13	96	28	24
	11	285	29	61	19	236	47	88	12	164	62	85	15	113	72	50
	7	163	36	72	13	193	75	90	12	174	91	100	20	81	142	77
	6	50	36	58	5	70	110	76	12	48	230	81	13	49	360	98

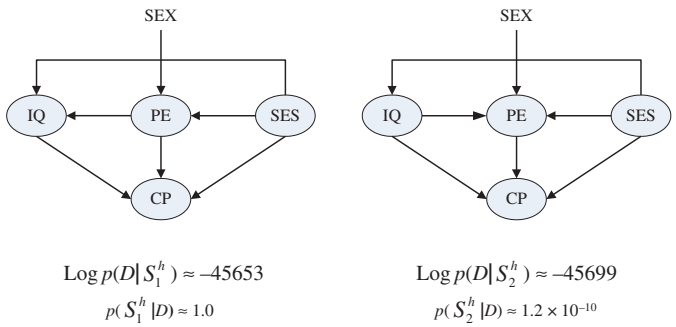


Fig. 4.1. Most likely network structures with no hidden variables

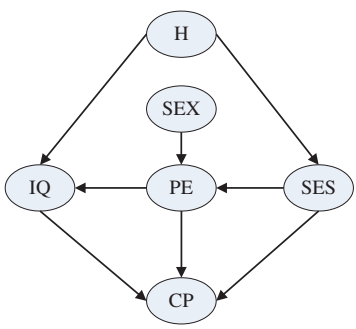
table, each entry represents a statistic of a state which cycles through all possible configurations. For example, the first entry indicates that the statistic for the configuration (SEX = male, SES = low, IQ = low, PE = low, CP = yes) is 4; the second entry states that the statistic for the configuration (SEX = male, SES = low, IQ = low, PE = low, CP = no) is 349. In the cycling of configuration of variables in the table, the last variable (CP) varies most quickly, and then PE, IQ, SES, and SEX vary more slowly. Thus, the upper four lines are the statistics for male students and the lower four lines are that for female students.

When analyzing data, we assume that there are no hidden variables. To generate priors for network parameters, we utilize an equivalent sample size of five and a prior network where $p(\boldsymbol{X}|S_c^h)$ is uniform. Except that we exclude the structure where SEX and/or SES have parents and/or CP has children, we assume that all network structures are equally likely. Because the data set is complete, we use formulas (4.40) and (4.41) to calculate the posterior of the network structure. After searching all network structures exhaustively, we find the two most likely networks which are shown in Figure 4.1. Note that the posterior probabilities of the two most

likely network structures are very close. If we adopt casual Markov assumption and assume that there are no hidden variables, the arcs in the two graphs can all be interpreted casually. Some of these results, such as the influence of socioeconomic status and IQ on the college plan, are not surprising. Some other results are very interesting: from both graphs, we can see that the influence of Sex on the College Plan is conveyed by the influence of Parental Encouragement. Besides, the only difference between the two graphs is the direction of the arc between PE and IQ. Two different casual relationships are seen, both of which seem reasonable. The network on the right in the below figure was in 1993 with a non-Bayesian method.

The most questionable result is whether socioeconomic status has a direct influence on IQ. To verify the result, we consider a novel model which replaces the direct influence in the original model with a hidden variable pointing to SES and IQ. Besides, we also consider such models where a hidden variable points to SES, IQ, and PE and none or one or both of two links of SES–PE and PE–IQ are removed. For each structure, the number of hidden variables in these models varies from two to six.

We use the Cheeseman–Stutz variant of Laplace approximation to compute the posterior probabilities of these models. To find the MAP, we use EM algorithm and take the largest local maximum from 100 runs with different random initials. The model with the highest MAP is shown in Figure 4.2. This model is 2×10^{10} times more likely than the best model containing no hidden variables about. Another most likely model contains a hidden variable and has an additional arc from the hidden variable to PE. This model is only 5×10^{-9} times less likely than the best model. Suppose that no reasonable models are ignored, strong evidence suggests that there is a hidden variable influencing the SES and IQ. An examination of the probabilities



$\text{Log}\, p(D|\, S^h) \approx -45629$

Fig. 4.2. The *a posteriori* most likely network structure with a hidden variable

in Figure 4.2 suggests that the hidden variable corresponds to some concept like "parent quality".

Using Bayesian method to learn the structure and probabilities of Bayesian network from prior information and sample information so as to construct a whole Bayesian network opens an avenue for applying Bayesian network to data mining and knowledge discovery. Compared with other data mining methods, such as rule-based method, decision tree, and artificial neural network, Bayesian network has the following characteristics:

(1) It can integrate prior and posterior information, so as to avoid subjective bias when using merely prior information, to avoid the large blind searching and computation when sample is lacking, and to avoid the influence from noise when using only posterior information. As long as the prior is determined properly, we can perform effective learning, especially when the sample is difficult or costly to gain.
(2) It can handle an incomplete data set.
(3) It can explore casual relations in data.
(4) There are mature and effective algorithms. Although probabilistic reasoning is NP hard for any arbitrary Bayesian network, in many practical problems, these operations can be either simplified by adding some constrains or solved by some approximation methods.

Yet, the computation of Bayesian network is huge. The Bayesian network seems less efficient than some other methods if a problem is also be resolved by other efficient approaches. Although there are some methods for prior determination, which is extremely important when a sample is hard to get, in practice, to find a reasonable prior involving many variables is still a difficult problem. Besides, a Bayesian network requires many assumptions as precondition. There are no ready rules to judge whether a practical problem satisfies the assumptions or not. These are problems that deserve further study. Still, it can be predicted that in data mining and knowledge discovery, especially in data mining with probabilistic statistical features, the Bayesian network will become a powerful tool.

4.6 Bayesian Latent Semantic Model

With the prevalence of Internet, Web information is increasing in an exponential manner. It has been a research focus of Web information processing as to how to organize the information reasonably so as to find the expected target in massive Web data, and how to effectively analyze the information so as to mine new and latently

useful patterns in the massive Web data. The classification of Web information is an effective approach for improving search effectiveness and efficiency. For example, when searching with a Web search engine, if the class information of the query is available, the searching sphere will be limited and recall will be improved. Meanwhile, classification can provide good organization of information so as to help the user browse and filter information. Many big websites adopt this kind of information organization. For example, Yahoo maintains its Web catalog structure manually; Google uses some sorting mechanism to let the most user-related pages rank ahead, so as to make users' browse convenient. Deerwester *et al.* take the advantage of linear algebra and perform information filtering and latent semantic index (LSI) via singular value decomposition (SVD) (Deerwester *et al.* 1990). They project the high-dimensional representation of documents in vector space model (VSM) to a low-dimensional latent semantic space (LSS). This approach on the one hand reduces the scale of the problem and on the other hand, to some extent, avoids the over-sparse data. It gains preferable effects in many applications including language modeling, video retrieval, and protein database.

Clustering is one of main approaches in text mining. Its primary effects include the following: (a) By clustering search results, a website can provide users the required Web pages in terms of classes, so that users can quickly locate their expected targets, (b) generating the catalog automatically, (c) analyzing the commonness in Web pages by clustering them. The typical clustering algorithm is K-means clustering. Besides, some new clustering algorithms, such as self-organizing map (SOM), clustering with neural networks, and probability-based hierarchical Bayesian clustering (HBC), are also under intensive study and have many applications. Yet, most clustering algorithms are unsupervised algorithms, which search the solution space somewhat blindly. Thus, the clustering results often lack semantic characteristics. Meanwhile, in high-dimensional cases, selecting a proper distance metric becomes very difficult.

Web classification is one kind of supervised learning. By analyzing training data, classifiers can predict the class labels for unseen Web pages. Currently, there are many effective algorithms to classify Web pages, such as naïve Bayesian method and SVM. It is a pity that obtaining a large amount of classified training samples, which are necessary for training highly precise classifiers, is very costly. Besides, in practice, different classification architectures are often inconsistent. This makes daily maintaining of the Web catalog difficult. Kamal Nigam *et al.* proposed a method that can utilize documents with class labels and those without class labels to train a classifier. It only requires a small amount of labeled training samples, and one can learn a Bayesian classifier by integrating knowledge in the unlabeled samples (Nigam *et al.*, 1998).

Our basic idea for solving this problem is as follows. If some Web pages $D = \{d_1, d_2, \ldots, d_n\}$ consist of a description of some latent class variables $Z = \{z_1, z_2, \ldots, z_k\}$, first, by introducing Bayesian latent semantic model, we assign documents containing latent class variables to the corresponding class; then we utilize naïve Bayesian model to classify the documents containing no latent class variables with the knowledge in the previous step. According to the characteristics of these two steps, we define two likelihood functions and use the EM algorithm to find the local optimal solution with the maximum likelihood. This approach on the one hand avoids blind search in the solution space like unsupervised learning; on the other hand, it requires only some class variables but not a large amount of labeled training samples. It will release website managers from fussy training document labeling and improve the efficiency of Web page automatic classification. To distinguish from supervised learning and unsupervised learning, this approach is named semi-supervised learning.

The basic idea of latent semantic analysis (LSA) is to project the documents in a high-dimensional VSM to a low-dimensional latent semantic space. This projection is performed via singular value decomposition (SVD) on entry/document matrix $N_{m\times n}$. Concretely, according to linear algebra, any matrix *Nm*n* can be decomposed as follows:

$$\boldsymbol{N} = \boldsymbol{U}\sum \boldsymbol{V}^T, \tag{4.42}$$

where $\boldsymbol{U}$, $\boldsymbol{V}$ are orthogonal matrixes ($\boldsymbol{U}\boldsymbol{U}^T = \boldsymbol{V}\boldsymbol{V}^T = \boldsymbol{I}$); $\sum = \text{diag}(a_1, a_2, \ldots, a_k, \ldots, a_v)$ ($a_1, a_2, \ldots, a_v$ are singular values) is a diagonal matrix. In latent semantic analysis, the approximation is gained by keeping k biggest singular values and setting others to 0:

$$\tilde{\boldsymbol{N}} = \boldsymbol{U}\tilde{\sum}\boldsymbol{V}^T \approx \boldsymbol{U}\sum \boldsymbol{V}^T = \boldsymbol{N}. \tag{4.43}$$

Because the similarity between two documents can be represented with $\boldsymbol{N}\boldsymbol{N}^{\mathrm{T}} \approx \tilde{\boldsymbol{N}}\tilde{\boldsymbol{N}}^T = U\tilde{\sum}^2\boldsymbol{U}^T$, the coordinate of a document in the latent semantic space can be approximated by $\boldsymbol{U}\tilde{\sum}$. After projecting the representation of a document from high-dimensional space to low-dimensional semantic space, the sparsity of data, which exists in high-dimensional space, does not exist any more in the low-dimensional latent semantic space. This also indicates that even if there is no common factor between the two documents in the high-dimensional space, we may still find meaningful connections between them in the low-dimensional semantic space.

After the SVD and projecting documents from a high-dimensional space to a low-dimensional latent semantic space, the scale of the problem is effectively reduced. LSA has been successfully applied to many fields, including information

filtering, text indexing, and video retrieval. Yet, SVD is sensitive to variation of data and seems stiff when prior information is lacking. These shortcomings limit its application.

According to our experiences, the description of any problem is developed centering on some theme. There are relative obvious boundaries between different themes. Because of differences in personal favors and interests, people's concerns on different themes vary. There is prior knowledge in different themes. Accordingly, we proposed the Bayesian latent semantic model for document generation.

Let document set be $D = \{d_1, d_2, \ldots, d_n\}$, and word set be $W = \{w_1, w_2, \ldots, w_m\}$. The generation model for document $d \in D$ can be expressed as follows:

(1) Choose document d at the probability of $P(d)$;
(2) Choose a latent theme z, which has the prior knowledge $p(z|\theta)$;
(3) Denote the probability that theme z contains document d by $p(z|d, \theta)$;
(4) Denote the probability of word $w \in W$ under the theme z by $p(w|z, \theta)$.

After this process, we get the observed pair (d, w). The latent theme z is omitted, and joint probability model is generated as follows:

$$p(d, w) = p(d)p(w|d), \tag{4.44}$$

$$p(w|d) = \sum_{z \in Z} p(w|z, \theta)p(z|d, \theta). \tag{4.45}$$

This model is a hybrid probabilistic model under the following independence assumptions:

(1) The generation of each observed pair (d, w) is relatively independent, and they are related via latent themes.
(2) The generation of word w is independent of any concrete document d. It only depends on latent theme variable z.

Formula (4.45) indicates that in some document d, the distribution of word w is the convex combination of latent themes. The weight of a theme in the combination is the probability at which document d belongs to the theme. Figure 4.3 illustrates the relationships between factors in the model.

According to Bayesian formula, we substitute Formula (4.45) into Formula (4.44) and get

$$p(d, w) = \sum_{z \in Z} p(z|\theta)p(w|z, \theta)p(d|z, \theta). \tag{4.46}$$

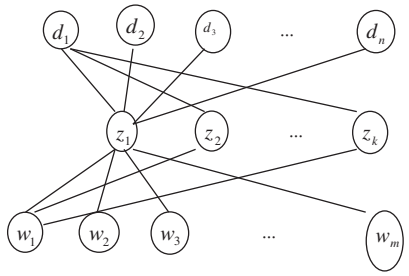


Fig. 4.3. Bayesian latent semantic model

Compared with LSA, Bayesian latent semantic model has a firm statistical foundation and avoids the data sensibility in LSA. It also utilizes prior information of latent theme variables to avoid being over stiff like the SVD. In the Bayesian latent semantic model, formula (4.42) can be rewritten as follows:

$$U = \{p(d_i|z_k)\}_{n\times k},$$
$$V = \{p(w_i|z_k)\}_{m\times k},$$
$$\tilde{\Sigma} = \mathrm{diag}(p(z_k), p(z_k), \ldots, p(z_k)).$$

So it has same representation form as that of SVD.

In LSA, the criterion for parameter selection is minimum least square loss. From the viewpoint of Bayesian learning, in our model, we have two applicable criteria: maximum *a posteriori* (MAP) and maximum likelihood (ML).

MAP estimation is applied to find the proper latent theme variable under the condition of document set D and word set W:

$$P(Z|D, W) = \prod_{z\in Z}\prod_{d\in D}\prod_{w\in W} p(z|d, w). \tag{4.47}$$

According to Bayesian formula, we have

$$p(z|d, w) = \frac{p(z)p(w|z)p(d|z)}{\sum_{z\in Z} p(z)p(w|z)p(d|z)}. \tag{4.48}$$

ML estimation is used to find a proper value of the following expression:

$$\prod_{d\in D}\prod_{w\in W} p(d, w)^{n(d,w)}, \tag{4.49}$$

where $n(d, w)$ represents the count of word w in document d. In practice, we often take a logarithm of the likelihood, shortened to log-likelihood:

$$\sum_{d \in D} \sum_{w \in W} n(d, w) \log p(d, w). \tag{4.50}$$

A general approach to maximize the two estimations is expectation maximum (EM), which is discussed in detail in Section 4.7.

4.7 Semi-Supervised Text Mining Algorithms

4.7.1 *Web Page Clustering*

Presently there are many algorithms for text classification, and they can achieve satisfactory precision and recall. Yet, the cost for obtaining labeled training documents is very high. Nigam *et al.* proposed an approach in which they used mix corpus including labeled and unlabeled documents to train a classifier and gained good classification results, but they still needed a certain number of labeled documents (Nigam *et al.* 1998). Web clustering merges related Web pages into one cluster with some similarity criterion. When dealing with high-dimensional and massive data, conventional clustering methods cannot achieve satisfactory effectiveness and efficiency. The reason is, on the one hand, unsupervised search in solution space is to some extent blind; on the other hand, common similarity metric, e.g., Euclidean distance, does not work well in high-dimensional space and it is hard to find proper similarity metric in this situation. Considering the characters of supervised learning and unsupervised learning, we proposed a semi-supervised learning algorithm. Under the framework of Bayesian latent semantic model, we can classify documents into different classes with some user-provided latent class variables. In this process, no labeled training documents are required.

The general model is described as follows: given a document set $D = \{d_1, d_2, \ldots, d_n\}$ and its word set $W = \{w_1, w_2, \ldots, w_m\}$, and a group of class variable $Z = \{z_1, z_2, \ldots, z_k\}$ with its prior information $\theta = \{\theta_1, \theta_2, \ldots, \theta_k\}$, try to seek a division $D_j (j \in (1, \ldots, k))$ of D, so that

$$\bigcup_{j=1}^{k} D_j = D, \quad D_i \cap D_j = \phi \ \ (i \neq j).$$

First, we divide D into two sets: $D = D_L \cup D_U$, where

$$D_L = \{d | \exists j, z_j \in d, j \in [1 \ldots k]\},$$

$$D_U = \{d | \forall j, z_j \notin d, j \in [1 \ldots k]\}.$$

In our algorithm, the classification process includes two stages

Stage 1. Utilize Bayesian latent semantic model with the parameters estimated based on EM algorithm to label the documents in D_L:

$$l(d) = z_j = \max_i \{p(d|z_i)\}. \tag{4.51}$$

Stage 2. Train a naïve Bayesian classifier with the labeled documents in D_L, and label documents in D_U with this classifier. Then, update the parameters of Bayesian latent semantic models with the EM algorithm.

4.7.2 *Label Documents with Latent Classification Themes*

Ideally, any document will not contain more than one latent class theme. In this case, we can easily label a document with its latent theme. In practice, however, the ideal status is hard to achieve. On the one hand, it is difficult to find such a latent theme; on the other hand, there may be multiple themes in one document. For example, a document labeled with "economics" may contain words of other themes, e.g., "politics" and/or "culture". We handle these cases by labeling them with the most related theme. Under ML criterion, after some rounds of EM iterations, we finally determine the theme of the test document according to Formula (4.51).

EM algorithm is one of the primary parameter estimation approaches for sparse data. It performs E step and M step alternately so as to find the most likely result. The general process of EM algorithm is described below.

(1) **E step:** calculate expectation based on the current parameters;
(2) **M step:** find the proper parameter with maximum likelihood based on the expectation in E step;
(3) Compute the likelihood with the renewed parameters. If the likelihood exceeds the predefined threshold or the number of iterations exceeds the predefined value, stop. Otherwise, go to Step (1).

In our algorithm, we adopt the following two steps to perform the iteration:

(1) In E step, we obtain the expectation via the following Bayesian formula:

$$P(z|d, w) = \frac{p(z)p(d|z)p(w|z)}{\sum_{z'} p(z')p(d|z')p(w|z')}. \tag{4.52}$$

In terms of probabilistic semantics, the formula explains the probability of word w in document d with latent theme variable z.

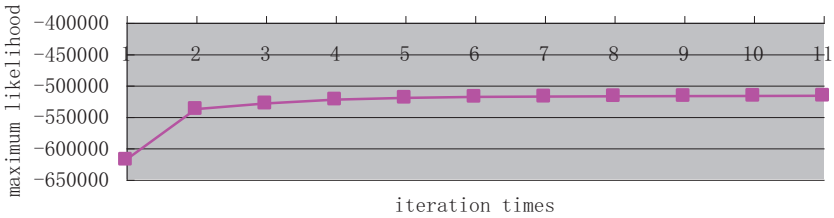


Fig. 4.4. Relationship of iteration times and maximum likelihood

(2) In M step, we use the expectation from the above step to estimate the density of parameters:

$$p(w|z) = \frac{\sum_d n(d,w)p(z|d,w)}{\sum_{d,w'} n(d,w')p(z|d,w')}, \tag{4.53a}$$

$$p(d|z) = \frac{\sum_w n(d,w)p(z|d,w)}{\sum_{d',w} n(d',w)p(z|d',w)}, \tag{4.53b}$$

$$p(z) = \frac{\sum_{d,w} n(d,w)p(z|d,w)}{\sum_{d,w} n(d,w)}. \tag{4.53c}$$

Compared with SVD in LSA, EM algorithm has linear convergence time. It is simple and easy to implement, and it results in local optimal of likelihood function. Figure 4.4 shows the relation between iteration times and the corresponding maximum likelihood in our experiment.

4.7.3 *Learning Labeled and Unlabeled Data Based on Naïve Bayesian Model*

Conventional classification methods usually learn classifiers based on labeled training samples to classify unlabeled data. Yet, obtaining a large amount of labeled training samples is very costly and fussy. Kamal Nigam *et al.*'s research indicated that unlabeled data also contain useful information for learning classifiers. Accordingly, we use naïve Bayesian model as classifier and label the unlabeled training samples with a special non-label status; then, we estimate these labels with EM algorithm.

Here, we present the general description of text classification with a naïve Bayesian classifier: given the training document set $D = \{d_1, d_2, \ldots, d_n\}$ and its word set $W = \{w_1, w_2, \ldots, w_m\}$, each training document is represented as an $m+1$ dimensional vector $\boldsymbol{d}_i = \langle w_1, w_2, \ldots, w_m, c_i \rangle$, where $c_i \in C = \{c_1, c_2, \ldots, c_k\}$ is a class variable. The classification task is to predict the class of unseen document

$\boldsymbol{d} = \langle w_1, w_2, \ldots, w_m \rangle$:

$$c = \max_{j \in 1,\ldots,k} \{p(c_j|d, \theta),$$

where θ is the parameter of model.

To calculate the above expression, we expend the factor in the expression and get

$$p(d|c_j, \theta) = p(|d|) \prod_{k=1}^{|d|} p(w_k|c_j; \theta; w_q, q < k). \tag{4.54}$$

When computing Formula (4.54) with the naïve Bayesian model, we need to introduce the following independence assumptions:

(1) The generation of words in documents is independent of the content. That is to say, same words at different position of a document are independent.
(2) Words in a document are independent of the class of the documents.

Based on the above independence assumption and Bayesian formula, Equation (4.54) can be rewritten as

$$\begin{aligned} p(c_j|d, \theta) &= \frac{p(c_j|\theta)p(d|c_j, \theta)}{p(d|\theta)} \\ &= \frac{p(c_j|\theta) \prod_{r=1}^{m} p(w_r|c_j, \theta)}{\sum_{i=1}^{k} p(c_i|\theta) \prod_{r=1}^{m} p(w_r|c_i, \theta)}. \end{aligned} \tag{4.55}$$

The learning task learns parameters of the model from prior information in training data. Here, we adopt multinomial distribution and Dirichlet conjugate distribution:

$$\theta_{c_j} = p(c_j|\theta) = \frac{\sum_{i=1}^{|D|} I(c(d_i) = c_j)}{|D|}, \tag{4.56a}$$

$$\theta_{w_t|c_j} = p(w_t|c_j, \theta) = \frac{\alpha_{jt} + \sum_{i=1}^{|D|} n(d_i, w_t) I(c(d_i) = c_j)}{\alpha_{j0} + \sum_{k=1}^{m} \sum_{i=1}^{|D|} n(d_i, w_k) I(c(d_i) = c_j)}, \tag{4.56b}$$

where $\alpha_{j0} = \sum_{i=1}^{k} \alpha_{ji}$ is the super-parameter of model; $c(\cdot)$ is the class labeling function $I(a = b)$ and is a characteristic function (if $a = b$, then $I(a = b) = 1$; otherwise $I(a = b) = 0$).

Although the applicable condition for naïve Bayesian model is somewhat harsh, numerous experiments demonstrate that even when independence assumption is

unsatisfied, naïve Bayesian model can still work robustly. It has been one of the most popular methods for text classification.

Below, we will classify unlabeled documents according to MAP criterion based on the knowledge in these unlabeled documents.

Consider the entire sample set $D = D_L \cup D_U$, where D_L is the set of documents that has been labeled in the first stage. Assume that the generation of all samples in D is mutually independent; then, the following equation holds:

$$p(D|\theta) = \prod_{d_i \in D_U} \sum_{j=1}^{|C|} p(c_j|\theta) p(d_i|c_j, \theta) \cdot \prod_{d_i \in D_L} p(c(d_i)|\theta) p(d_i|c(d_i), \theta). \quad (4.57)$$

In the above equation, unlabeled documents are regarded as a mix model. Our learning task is to gain the maximum estimation of model parameter θ with the sample set D, and according to Bayesian theorem, we have

$$p(\theta|D) = \frac{p(\theta) p(D|\theta)}{P(D)}. \quad (4.58)$$

For a fixed sample set, $p(\theta)$ and $p(D)$ are both constants. Take logarithm on the both sides of Equation (4.58). We have

$$\begin{aligned} l(\theta|D) &= \log p(\theta|D) \\ &= \log \frac{p(\theta)}{p(D)} + \sum_{d_i \in D_U} \log \sum_{j=1}^{|C|} p(c_j|\theta) p(\mathrm{d}_i|c_j, \theta) \\ &\quad + \sum_{d_i \in D_L} \log p(c(d_i)|\theta) p(\mathrm{d}_i|c(\mathrm{d}_i), \theta). \end{aligned} \quad (4.59)$$

To label the unlabeled documents, we need latent variables in LSA. Here we introduce k latent variables $Z = \{z_1, z_2, \ldots, z_k\}$, where each latent variable is an n-dimensional vector $z_i = \langle z_{\mathrm{i}1}, z_{\mathrm{i}2}, \ldots, z_{\mathrm{in}} \rangle$, and if $c(\boldsymbol{d}_j) = c_i$ then $z_{ij} = 1$, otherwise $z_{ij} = 0$. So, Equation (4.59) can be rewritten as follows:

$$l(\theta|D) = \log \frac{p(\theta)}{p(D)} + \sum_{i=1}^{|D|} \sum_{j=1}^{|C|} z_{ji} \log p(c_j|\theta) p(\mathrm{d}_i|c_j, \theta_j). \quad (4.60)$$

In Equation (4.59), z_{ji} for labeled documents is known. The learning task maximizes model parameters and estimates z_{ji} of unlabeled documents.

Here, we still apply EM algorithm to gain knowledge about unlabeled documents. Yet, the process is somewhat different from the previous stage. In the kth iteration in E step, we will use the naïve Bayesian classifier to find the class label

of unlabeled documents based on the current estimation of parameters:

$$p(d|c_j, \theta^k) = \frac{p(c_j|\theta^K) \prod_{r=1}^{m} p(w_r|c_j; \theta^k)}{\sum_{i=1}^{k} p(c_i|\theta^k) \prod_{r=1}^{m} p(w_r|c_i; \theta^k)}, \quad j \in 1, \ldots, k.$$

The class c_i corresponding to MAP is the expected label of the unlabeled documents

$$z_{id} = 1, \quad z_{jd} = 0 (j \neq i).$$

In step M, we maximize the estimation of the current parameters based on the expectation obtained from the just previous E step:

$$\theta_{c_j} = p(c_j|\theta) = \frac{\sum_{i=1}^{|D|} z_{ji}}{|D|}, \tag{4.61a}$$

$$\theta_{w_t|c_j} = p(w_t|c_j, \theta) = \frac{\alpha_j + \sum_{i=1}^{|D|} n(d_i, w_t) z_{ji}}{\alpha_0 + \sum_{k=1}^{m} \sum_{i=1}^{|D|} n(d_i, w_k) z_{ji}}. \tag{4.61b}$$

Organizing Web information into catalogs is an effective way to improve the effectiveness and efficiency of information retrieval. It can be achieved by learning classifiers with labeled documents and predicting the class label of a new Web page with the learnt classifiers. Yet, the acquisition of labeled training data is often costly and fussy. Web page clustering, which can cluster documents according to some similarity metric, can help to improve the retrieval. The problem is that the solution search of traditional clustering methods is somewhat blind and lacks semantic meaning. Thus, the effect of clustering is usually unsatisfactory. In this section, we proposed a semi-supervised learning algorithm. Under the framework of Bayesian latent semantic model, the new algorithm uses no labeled training data but only a few latent class/theme variables to assign documents to the corresponding class/theme. The algorithm includes two stages. In the first stage, it applies Bayesian latent semantic analysis to label documents which contain latent theme variable(s); in the second stage, it uses the naïve Bayesian model to label the documents without latent theme with the knowledge information in these documents. Experimental results demonstrate that the algorithm achieves high precision and recall. We will further investigate related issues, such as the influence of latent variable selection on the clustering result and how to implement word clustering under the framework of Bayesian latent semantic analysis.

Exercises

4.1 Explain conditional probability, prior probability, and posterior probability.
4.2 Describe Bayesian formula and explicate its significance thoroughly.

4.3 Describe some criteria for prior distribution selection.

4.4 What does "Naïve" mean in Naïve Bayesian classification? Briefly state the main ideas for improving Naïve Bayesian classification.

4.5 Describe the structure of Bayesian network and its construction and exemplify the usage of a Bayesian network.

4.6 What is semi-supervised text mining? Describe some applications of the Bayesian model in Web page clustering.

4.7 In recent years, with the development of Internet technology, Bayesian rules are widely applied. Exemplify two concrete applications of the Bayesian rules and explain the results.