Computers and Electrical Engineering 000 (2017) 1–12

Contents lists available at ScienceDirect

Computers and Electrical Engineering

journal homepage: www.elsevier.com/locate/compeleceng

A Bayesian assumption based forecasting probability distribution model for small samples☆

Zonglei Lu[a,b,*], Xiaohan Geng[a], Guoming Chen[b]

[a] *Department of Computer Science & Technology, Civil Aviation University of China, Tianjin, China*
[b] *Information Technology Research Base of Civil Aviation Administration of China, Tianjin, China*

ARTICLE INFO

Article history:
Received 15 July 2017
Revised 20 November 2017
Accepted 20 November 2017
Available online xxx

Keywords:
Small sample size problem
Probability distribution forecasting
Naïve Bayes
Bayesian network

ABSTRACT

In this work, a novel forecasting probability distribution model is presented. Probability distribution plays a role in the function of probability values. Therefore, forecasting the probability distribution function is a challenging process. To that end, the method described in this work loosens the control conditions of the given data set. Subsequently, statistical methods can be applied to the resulting sample data. The distribution functions are then fitted using the cubic spline interpolation method. In this work, the naive Bayes and the Bayesian network methods are adjusted to handle the small sample problem. In addition, the maximal extension clusters are used to determine the conditional function. Two data sets from the UCI repository and a custom data set are used to validate the forecasting model. The experiments show the proposed method can generate an accurate distribution function.

1. Introduction

Estimating the probability distribution function is a classical machine learning problem. Probability distribution is a useful tool for describing random variables because a single probability value does not adequately describe the variable. Probability distribution is a mathematical description of a random phenomenon. The base of this description is the probability of events. Thus, probability distribution requires a more complicated model than the probability of events. Moreover, probability distribution has been widely applied in many fields, such as lightning current amplitude [1], wind speed [2], and multimeric systems [3]. Obtaining the probability distribution function of a large sample set is a relatively easy task. However, in the case of a small sample set, this traditional method may not be applicable.

The small sample size problem (SSSP) is a hot topic in current academic research. For example, in tasks pertaining face recognition [4] or speech emotion [5], the lack of large sample size is a challenge. To that end, loosen control condition (LCC) [6] has been applied as valid method for a small sample set. Virtual sample generation (VSG) [7] has also been used to address the SSSP. Li Der-Chiang proposed a genetic algorithm based on virtual sample generation, derived from LCC and VSG [8]. In [9], Zhang Cui-Cui proposed an ensemble framework to generate new data from the distribution of the original samples.

The bootstrap method has been widely used to address the SSSP. First, several bootstrap samples are generated by resampling the original data set. Then, the probability distribution of each bootstrap sample is calculated, thereby enabling the

☆ Reviews processed and recommended for publication to the Editor-in-Chief by Associate Editor Dr. Huimin Lu.

* Corresponding author.
E-mail addresses: zllv@cauc.edu.cn (Z. Lu), xhgengyjs_16@cauc.edu.cn (X. Geng), gmchenyjs_13@cauc.edu.cn (G. Chen).

https://doi.org/10.1016/j.compeleceng.2017.11.025
0045-7906/© 2017 Elsevier Ltd. All rights reserved.

estimation of the probability distribution of the original dataset. Thus, the bootstrap method can be utilized for estimating the parameters in the case of small samples.

The conventional small sample learning methods mainly focus on a possible value of the target attribute; however, this value is only related to the probability of events. Even so, forecasting probability distribution of in the case of small samples is a challenging problem. To that end, the proposed model, which is based on LCC and the Bayesian learning model, attempts to address this challenge.

The remainder of this paper is organized as follows. Related work is reviewed in Section 2. The proposed method is introduced in Section 3. Experiments and results of the experimentation are described in Section 4. The paper is concluded and the findings are presented in Section 5.

2. Basic concepts and principles

Probability is a measure of the likelihood of an event occurring, and as such, there are two different theoretical explanations about probability. The first is a frequentist view that defines probability as an objective concept. The probability of an event is the limiting proportion of times that the event occurs from a long series of independent identical opportunities. The second explanation is the Bayesian probability view (also called subjective probability view), wherein the probability is regarded as an inner state rather than an objective property of the outside world [10]. The probability only denotes the degree of personal beliefs that the event occurred.

According to the Bayesian view, all probabilities are conditional probabilities. The definition of conditions is related to subjective recognition. Moreover, different conditions are likely to result in different outcomes. Thus, several different schemes can be generated to select instances from the data set. Consider a data set of seismic records from all regions of the world. The task is to obtain the distribution of intermediate-focus earthquakes in a certain region. Therefore, the sample should consist of seismic records from the same location. However, the probability of finding an earthquake instance that matches the exact condition is very low. The probability improves, however, when the definition of the location extends to a specific area surrounding the specific location. If the definition extends to across the world, it is equivalent to encompassing all the data sets. As the definition extends, the set contains seismic data with different characteristics. Therefore, the extension should be confined to meet the statistical requirements.

According to the Bayesian probability view, the sample is the result of selection. The sample set can be extended by omitting some selected attribute value. Let us consider a universal set, and the sample set is just a subset of the universal set. The generation of the samples mainly depends on the selection criteria. Thus, a small sample may be generated by a highly strict selection criterion. In this work, *selection* [11] is defined as follows:

Definition 1. Selection produces a horizontal subset of a given data set D, which consists of all the instances that satisfy the condition set C. The selection by C from D is denoted as $\sigma_{\hat{C}}(D) = \{d | d \in D \wedge \hat{C}(d) = T\}$, where D is a subset of the Cartesian product of conditional attributes and a target attribute, namely $D = X \times Y$, $X = X_1 \times X_2 \times \ldots \times X_k$ and $\hat{C}$ denotes a selection condition generated by the conjunction of conditions in C.

Here, C is a group of constraints on X. Therefore, $\hat{C}(\cdot)$ is a logical expression with a value of "T" or "F". For example, if $C = \{A_1 = a, A_2 = b, A_3 = c\}$, then $\hat{C}(d) = (d.A_1 = a) \wedge (d.A_2 = b) \wedge (d.A_3 = c)$.

Theorem 1. *If* $D_1 = \sigma_{\hat{C}_1}(D)$, $D_2 = \sigma_{\hat{C}_2}(D)$ *and* $C_1 \subseteq C_2$, *then* $D_2 \subseteq D_1$.

Proof. 1) If $C_1 = C_2$, then we establish that $D_1 = D_2$.
2) If $C_1 \subset C_2$, and because $D_2 = \sigma_{\hat{C}_2}(D)$,
We have for each $d \in D_2$, $\hat{C}_2(d) = T$.
Moreover, given that $C_1 \subset C_2$, we have $\hat{C}_1(d) = T$.
Thus, $d \in D_1$ is established.
Therefore, $D_2 \subseteq D_1$.

Corollary 1. *If* $D_1 = \sigma_{\hat{C}_1}(D)$, $D_2 = \sigma_{\hat{C}_2}(D)$ *and* $C_1 \subseteq C_2$, *then* $|D_1| \geq |D_2|$.

Here, Corollary 1 shows that when the conditions become more stringent, the number of data will be reduced, and vice versa. Therefore, the data set may be expanded or shrunk by adjusting the condition set.

Expansion of the data set is a conventional method applied to solving the SSSP. Therefore, the approach utilized to expand the data set, i.e., to loosen the conditions, is significant. Each constraint in the given condition set can generate a new single-constraint condition set. Moreover, each single-constraint condition can generate a conditional data set. Thus, the generated dataset may have more instances than the original one. Therefore, for sufficient instances, there may exist a function that fits the single-constraint conditions. Moreover, these conditional functions contain the complete information about the final distribution function.

Parameter learning has been commonly applied to predict the conditional distribution. Parameter learning depends on the prior distribution type in the corresponding field. Unfortunately, in some cases, there is little information about the prior distribution type. According to the analytic geometry, the process of forecasting distribution is a process of fitting the plane curve. Therefore, the interpolation function can fit the distribution. Moreover, the cubic spline interpolation function is second-order continuous. It avoids Runge's phenomenon, which often appears in higher-order polynomial interpolation. In [2], the sectioned cubic spline interpolation function has been applied. The position of its inflection point determines the number of segments. However, cubic spline interpolation is unstable in forecasting non-monotonic functions as some knots in fitting curve function exist below the X-axis. Therefore, to address this problem, the equation is adjusted as shown in Eq. (1).

$$\begin{cases} f(x_i) = y_i \\ f(x_{i+1}) = y_{i+1} \\ f'(x_i) = 0 \\ f'(x_{i+1}) = 0 \end{cases} \tag{1}$$

In Eq. (1), $f(x) = ax^3 + bx^2 + cx + d$. $f'(x)$is the derived function of $f(x)$, where x_i denotes the different target attribute values in increasing order based on i. The symbol y_i denotes the count number (or the ratio) of x_i. The coefficients (a, b, c, and d) of the ith segment are calculated by Eq. (1). The $f(x)$ is a part of the conditional function.

Given sufficient predefined data, the cubic spline function can fit the target function well. Moreover, knots are calculated based on the frequency of the given value to fit the probability distribution function. With sufficient knots, the over-fitting and the under-fitting problems can be avoided.

Subsequent to adjustment, the cubic spline interpolation function can be used to calculate the conditional function. Several acceptable conditional functions can be obtained; however, only one function is real. In this work, we describe the ability of the Bayesian method to handle conditional probability. The naïve Bayesian assumption assumes that conditional attributes are mutually independent when given the target attribute [12]. Eq. (2) illustrates the Bayes assumption, where y, C, and c_i are the target random variable, the conditional data set, and its ith constraint in C, respectively. In Eq. (3), we replace the probability value with its distribution function, which can handle the probability distribution case; α is a constant coefficient.

$$P(c_1, c_2, \cdots, c_k|y) = \prod_{i=1}^{k} P(c_i|y) \tag{2}$$

$$f(y|C) = \alpha f(y) \prod_{i=1}^{k} f(c_i|y) \tag{3}$$

In [12], Pedro Domingos et al. pointed out that even though the naïve Bayesian assumption was not applicable in some samples, it still achieves a satisfactory result. Therefore, the Bayesian assumption can be considered as suiting most samples. In general, the original attributes can be converted to conditional independent attributes by principal component analysis, thus satisfying the naive Bayesian assumption. Therefore, the task now is transformed into forecasting the product of many single-constraint functions.

3. Forecasting probability distribution algorithm based on Bayesian network

Forecasting function based on the naïve Bayes is a relatively simple prediction method. We know that determining the remaining parts of condition sets can lead to the generation of more instances. Consequently, a sufficiently large number of instances can meet the requirements of statistics. However, the newly generated instances do not meet all the conditions, which may lead to reduced accuracy of the result. Therefore, the condition set should have the appropriate size for the sample set size to fit.

Definition 2. Valid condition set

The condition set C is called a valid condition set if C satisfies $|\sigma_{\hat{C}}(D)| \geq 30$.

The statistical principle requires that the sample size is not less than 30. Thus, the number of instances in the valid condition set should also be greater than 30. According to Definition 2, the valid condition set exists, only if $|D| \geq 30$. For example, an empty set must be a valid condition set, with the same choices as the original dataset.

Definition 3. Extension set

The valid condition set $C_2$is called an extension of C_1 if $C_1$is a valid condition set and $C_2 \supset C_1$. In particular, the valid condition set C_2 is called a gradual extension of C_1 if C_2 is an extension of C_1 and $|C_2| = |C_1| + 1$. The valid condition set$C_2$is called a maximal extension set if any valid condition set is not the extension of C_2.

Definition 3 implies that if a set is not the maximal expansion set, the set must be able to extend. Thus, the maximal extension sets can be determined by extending the empty set. Thus, the greatest extended cluster can be defined as the collection of all the greatest extended sets. Algorithm 1 shows the process of searching the maximal extension cluster, which is denoted by A_C.

Given a condition set C, let n be the number of constraints in C. Then, the time complexity of searching all its gradual extensions is $O(n)$. As can be inferred from Algorithm 1, the process of calculating NT requires scanning T twice. Intuitively, all possible combinations of the condition sets must be scanned to search the maximal extension cluster. Given that the algorithm is applicable to small samples, the time overhead is within the acceptable range.

Algorithm 1 Maximal extension cluster algorithm.

```
Input: sample set $D$ and condition set $C$
Output: the maximal extension cluster $A_C$
Step 1: Let n be the number of constraints in $C$, the temp cluster $T=\{\Phi\}$
Step 2: The new temporary cluster $NT$ consists of the gradual expansion of any element in $T$
Step 3: For $i=1$ to $n$
    Begin
Step 3.1: $T=NT$
Step 3.2: The new temporary cluster $NT$ consists of the gradual expansion of any element in $T$
Step 3.3: For each element $E$ in $T$
Step 3.4: If $E$ is not a subset of any element in $NT$, then $E$ is the maximal extension set and add $E$ into $A_C$
Step 3.5: If $NT$ is empty, then go to Step4
    End
Step 4: $A_C$ is the maximal extension cluster.
```

The maximal extension cluster algorithm provides a mechanism for loosening the control condition. The loosening of the control condition can be generalized as selecting the single-constraint conditions. This mechanism must strike a balance between the sample size and the condition set size. The maximal extension cluster generates samples that meet the statistical requirements. These samples are generated by loosening control conditions; each sample retains some characteristics from the original small sample. Given that the size of each sample is greater than 30, these samples can be used to train models. However, the same constraints may exist in different elements in a cluster. Thus, the repeated constraints will result in deviations in the final synthetic model.

Let A denote the Cartesian product of all attributes. Let $V=A$and $E=\{(X,\ Y)|X,\ Y \in A\wedge X \neq Y\wedge X\cap Y \neq \Phi\}$ be the vertex set and edge set of the graph, respectively. The undirected graph structure $G=(V,\ E)$ denotes the relationship among the maximal extension sets.

Given the directions of the edges, the undirected graph will be converted to a directed graph, which forms the central infrastructure of the Bayesian network. To describe the probability distribution governing a set of variables, the Bayesian network specifies a set of conditional independence assumptions and a set of conditional probabilities.

In this paper, the relationships among vertexes are extended to functions. Each directed edge denotes a conditional distribution table. The target attribute T is added into vertex set V as a vertex to calculate the distribution. Meanwhile, the edges from T to other vertices are added to the edge set E, too. Then, the Bayesian network can infer the value of some target variables when the observed values of the other variables are given. The interpolation functions could be built according to the graph and conditional values. Moreover, the variable is independent of its non-successor vertex in the Bayesian network. Thus, the final distribution can be calculated by Eq. (4).

$$f(C|y) = \prod_{i=1}^{n-1} f(V_i|Parents(V_i)) \tag{4}$$

Here, n denotes the size of the vertex set, and V_i denotes the ith vertex in vertex set V. Given that the target attribute is the sink node, $f(y)$ and $f(V_n)$ can be regarded as the same. Therefore, from Eqs. (3) and (4), the final function can be rewritten as shown in Eq. (5). Eq. (6) is the regularized form of Eq. (5).

$$f(y|C) = \alpha\prod_{i=1}^{n} f(V_i|Parents(V_i)) \tag{5}$$

$$f(y|C) = f(y|C)/\int_{-\infty}^{\infty} f(y|C)dy \tag{6}$$

In the above, "*Parents*" represents the union of all the predecessor vertices in vertex V_i. For each i, $f(V_i \mid Parents(V_i)$is a cubic spline interpolation function. The time complexity of polynomial multiplication is governed by the number of items being multiplied. Hence, the process of composing the conditional functions will not be time-intensive.

Minimum description length (MDL), described in [13], is used to direct the edges of the graph and is capable of playing the role of the score function. The MDL score is shown in Eq. (7), where M, G_S, $|G_S|$ are the sample size, the Bayesian network, and the dimension of the given Bayesian network, respectively.

$$Score(G_S : D) = -\log_2 P(D|G_S) + \frac{lbM \times |G_S|}{2} \tag{7}$$

For the entire data set, Eq. (7) can be decomposed into independent factors about its parent vertices [14], as shown in Eq. (8).

$$Score(G_S : D) = \sum_{i=1}^{n}\left(\frac{\log_2 M}{2}\|\pi_i\|(\|V_i\|-1) - \sum_{j=1}^{\|\pi_i\|}\sum_{k=1}^{\|V_i\|} N_{ijk} lb\theta_{ijk}\right) \tag{8}$$

Please cite this article as: Z. Lu et al., A Bayesian assumption based forecasting probability distribution model for small samples, Computers and Electrical Engineering (2017), https://doi.org/10.1016/j.compeleceng.2017.11.025

Algorithm 2 Forecasting probability distribution algorithm based on Bayesian network.

```
Input: sample set D and condition set C
Output: the final distribution function $f(y|C)$
Step 1: Search maximal extension cluster $A_C$
Step 2: Build a preliminary graph ($A_C$ as vertex set) and list all the possible structures
Step 3: Choose the minimal MDL among directed acyclic graphs and denote with $G_S$
Step 4: Calculate the conditional function of the vertices
Step 5: Obtain the final distribution from Eqs. (7) and (8).
```

Here, π_i denotes the parent vertex of V_i, and N_{ijk} denotes the number of the instances satisfying both $\pi_i = \pi_i{}^j$ and $V_i = V_i{}^j$. $\|\pi_i\|$ and $\|V_i\|$ denote the number of possible value in π_i and V_i, respectively. N_{ij} and θ_{ijk} are defined as $N_{ij} = \sum_{k=1}^{\|V_i\|} N_{ijk}$ and $\theta_{ijk} = N_{ijk}/N_{ij}$, respectively.

The directed acyclic graph with the lowest MDL score is the best structure for the given sample. Obviously, the possible number of Bayesian network structures increases exponentially with increasing size of the edge set. Therefore, for a large number of edges, random intelligent optimization algorithms can be used. These algorithms can search the possible directed acyclic graph. Fortunately, most small samples are relatively simple and can be enumerated completely. The forecasting probability distribution algorithm is presented in Algorithm 2.

Algorithm 2 outputs the probability distribution version of the Bayesian network. The time complexity of Algorithm 2 is the product of the time complexity of training original Bayesian network and the time complexity of cubic spline interpolation. In fact, for a small sample, as analyzed in Algorithm 1, Step1 is executed almost instantly. However, the time complexity in Step2 and Step3 is largely dependent on the accuracy of the intelligent optimization algorithms. Time complexity in Step4 is O(n), where n is the number of points. Step5 is a polynomial multiplication; therefore, the time complexity is only dependent on to the number of items contained.

4. Experiments

Based on the Bayesian assumption, the forecasting probability distribution algorithm is applied to a custom dataset, and three data sets—IRIS, QUAKE, WAVEFORM-500—from the UCI repository. The process of forecasting distribution function and the accuracy are described experimentally. Then, the relationship between the sample size and the accuracy is discussed.

The bootstrap method is a useful method for solving the SSSP. Thus, the bootstrap method is used as a baseline in this study. The performance of the proposed method and the bootstrap method are compared based on IRIS dataset. In the IRIS dataset, there are 150 instances with four numeric attributes. The four attributes are *Sepal_length, Sepal_width, Petal_length*, and *Petal_width*. In the experiment, the attribute *Sepal_length* was used as the target attribute. The other three attributes were used as conditional attributes. Each conditional attribute was discretized into five distinct ranges of equal size, wherein each range comprised 30 instances. The objective of the experiment was to find the probability distribution of *Sepal_length* under different conditions. There were 46 legal conditions to build a nonempty sample set. The following table lists the results of the experiment.

In Table 1 the first three columns represent the range of attribute values of the original dataset. For example, the first instance in IRIS is ⟨*Sepal_length* = 5.1, *Sepal_width* = 3.5, *Petal_length* = 1.4, *Petal_width* = 0.2⟩. The *Sepal_width* value of this instance is in the fifth range. The *Sepal_width* value and *Petal_length* value are both in the first range. Thus, the instance is converted to ⟨5.1, 5, 1, 1⟩. The fourth column indicates the number of instances that meet the conditions. The fifth and sixth columns list the mathematical expectations predicted by the bootstrap method and the proposed method, respectively.

From Table 1, it can be seen that the two methods predict quite similar expectations of *Sepal_length*. The difference between the two methods decreases when the sample size increases. Let us consider a particular case, where only one instance in the sample is present. Regardless of the kind of resampling method used, the result is only a repetition of the single instance. Thus, the bootstrap method is not applicable in this case. The proposed method extends the samples by loosening the conditions. Thus, unlike the bootstrap method, the proposed method can generate more similar instances. The mean root errors of these two methods for varying sample sizes are shown in Fig. 1. The X-axis denotes the size of the sample, while the Y-axis denotes the mean root error.

As can be seen in Fig. 1, there is little difference between the two methods for large sample sets. Therefore, the proposed method can be used to solve the SSSP.

There are three conditional attributes (A, B and C) and a target attribute (*target*) in the custom data set. The conditional attributes in the custom data set are randomly generated from 1, 2, and 3.Let m represents the number of the instances that meet one single-condition. The target attribute is randomly generated by the normal distribution N(0, σ2), where $\sigma = 3$/m. For example, let $A = 1$, $B = 1$, and $C = 1$. If m is equal to zero, its target attribute value follows a uniform distribution (from −3.5 to 3.5). Thus, the function $f(y|A = 1,\ B = 1,\ C = 1)$ is a standard normal distribution while the conditional functions are not. There are more than 10,000 instances in the custom data set. By equal-ratio sampling method, we select 300 instances to form a small sample. The naive Bayesian and the Bayesian network methods are evaluated using this sample. The statistical method is not applicable for this sample because the number of valid instances is 11.

Table 1
Experiment on IRIS dataset.

Sepal_width	Petal_length	Petal_width	Size	Expectation of Sepal_length predicted by bootstrap	Expectation of Sepal_length predicted by the proposed method
1	1	2	1	4.5	5.142747489
1	2	2	8	5.273	5.156954013
1	2	3	1	5.2	5.156954013
1	3	2	2	5.891333333	5.578646189
1	3	3	7	5.703555556	5.578646189
1	3	4	2	5.493666667	5.578646189
1	4	4	4	6.020222222	5.501407163
1	4	5	2	6.026	5.501407163
1	5	3	1	6.1	5.766342932
1	5	4	1	6.7	5.766342932
1	5	5	1	7.7	5.766342932
2	1	1	4	4.589555556	4.850929208
2	2	3	1	5.6	5.502378585
2	3	3	8	6.231888889	5.667004917
2	3	4	1	6.104666667	5.667004917
2	4	3	3	6.353555556	5.30586322
2	4	4	5	6.181777778	5.30586322
2	4	5	2	5.691333333	5.30586322
2	5	4	3	7.028444444	5.66239485
2	5	5	3	6.800111111	5.66239485
3	1	1	3	4.626555556	4.973698843
3	1	2	1	4.8	4.973698843
3	2	1	4	4.901444444	5.588990216
3	3	3	6	6.067222222	5.936393687
3	3	4	2	5.491333333	5.936393687
3	4	3	1	6.9	6.033368124
3	4	4	4	6.157888889	6.033368124
3	4	5	2	6.591333333	6.033368124
3	5	4	2	6.880333333	6.300025518
3	5	5	5	7.167777778	6.300025518
4	1	1	4	4.754666667	5.719480011
4	1	2	3	4.744888889	5.719480011
4	2	1	4	4.903555556	5.775556799
4	2	2	2	5.045666667	5.775556799
4	3	3	1	6.4	6.525725648
4	4	3	1	7	6.438357791
4	4	4	3	6.312888889	6.438357791
4	4	5	3	6.611888889	6.438357791
4	5	4	2	6.765333333	6.578643794
4	5	5	7	6.714888889	6.578643794
5	1	1	8	5.253777778	5.545545507
5	1	2	6	5.218666667	5.545545507
5	2	1	3	5.163777778	5.402306776
5	2	2	7	5.276777778	5.402306776
5	3	4	1	6	6.275232016
5	5	5	5	7.022777778	7.185734283

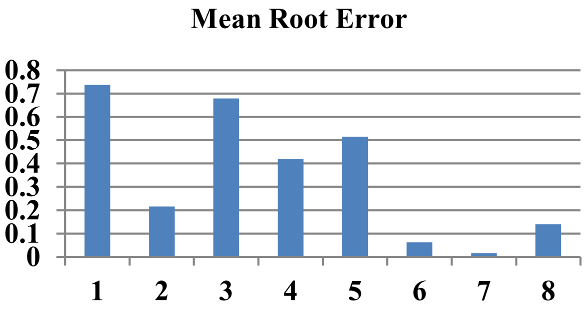

Fig. 1. Mean root error between bootstrap and proposed method for increasing sample sizes.

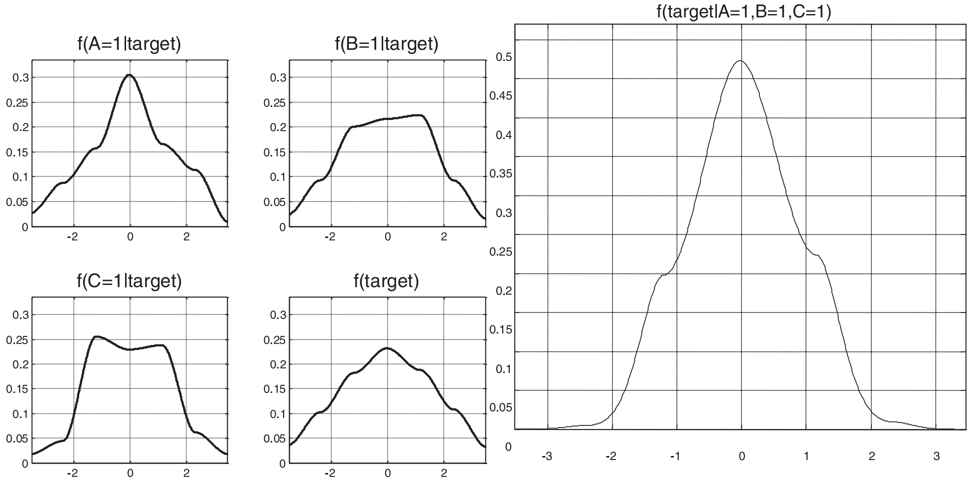

Fig. 2. Conditional functions and final function (naïve Bayes).

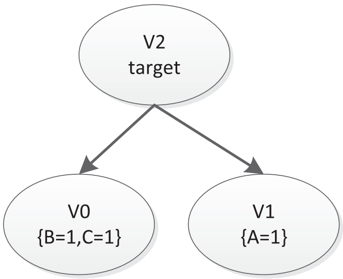

Fig. 3. Graph structure of the Bayesian Network method.

As regards the evaluation of the naïve Bayesian method, the conditional functions are shown in Fig. 2(left). The *f*(*target*) is generated using all instances in the small sample. The final forecasting function is shown in Fig. 1(right). The forecasting function may differ from the standard normal distribution. This deviation may be attributed to the influence of random factors.

In addition, there may exist some conditional attributes that are not independent. In this case, the maximal extension cluster is calculated using Eq. (9). The network structure is shown in Fig. 3 and the conditional functions are shown in Fig. 4(left). The final function is shown in Fig. 4(right).

$$A_C = \{\{B = 1, C = 1\}, \{A = 1\}\} \tag{9}$$

Fig. 5 shows the final functions obtained using different methods. The similarity score is used as a measure of accuracy. To that end, the similarity of two different functions is defined in Eq. (10). Let f_1, f_2be two distribution functions and $\int_{-\infty}^{\infty} f_1(x)dx = \int_{-\infty}^{\infty} f_2(x)dx = 1$. Then, Eq. (10) represents the area overlapped by the two functions.

$$Similar(f_1, f_2) = 1 - \frac{1}{2}\int_{-\infty}^{\infty} |f_1(x) - f_2(x)|dx \tag{10}$$

In Fig. 5, the standard normal distribution represents the forecasting result of the ideal world. The dashed and dotted line is the final function obtained using the naïve Bayesian method; this line is higher than the standard normal distribution. The dashed line is the final function obtained using the Bayesian network method; this line is lower than the standard normal distribution. For the small sample case, the dashed and dotted line and the dashed line seem to deviate from the standard normal distribution. The dotted line is the benchmark function, which is used to measure the accuracy of other methods. The benchmark function of the distribution function is predicted using statistical methods wherein all instances are used. In the given custom data set, the benchmark function is a standard normal distribution. However, a benchmark can be established only with sufficient data.

As can be seen in Fig. 5, the shapes of the four distribution functions images are similar. This implies that the expectations of the four distributions are equal. However, there may be a slight difference between the standard deviation of these four distribution functions. Thus, most of the confidence intervals of the distribution functions will be coincident. Most of these functions will do the same decision whether to accept the value or not.

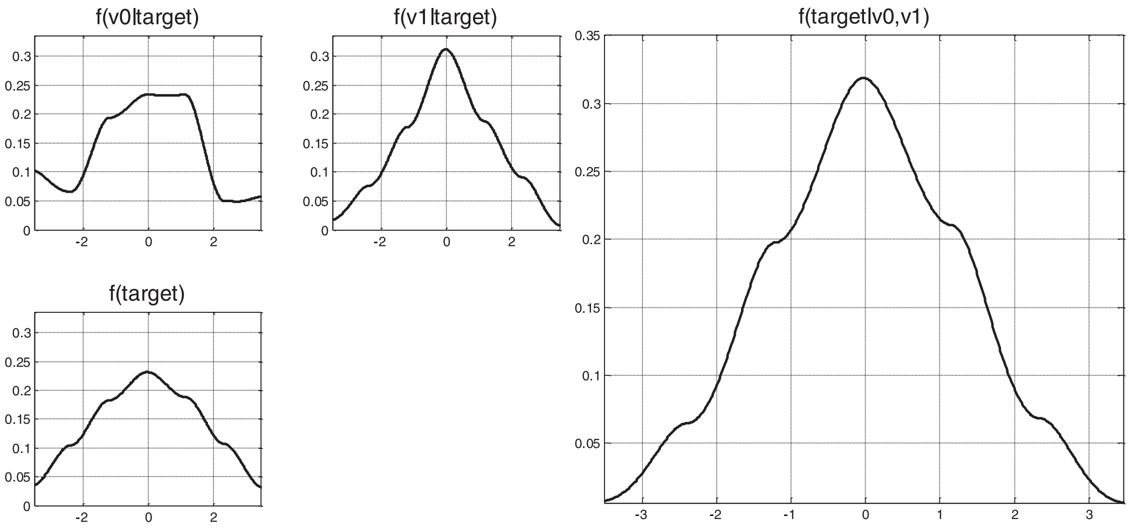

Fig. 4. Conditional functions and final function (Bayesian network).

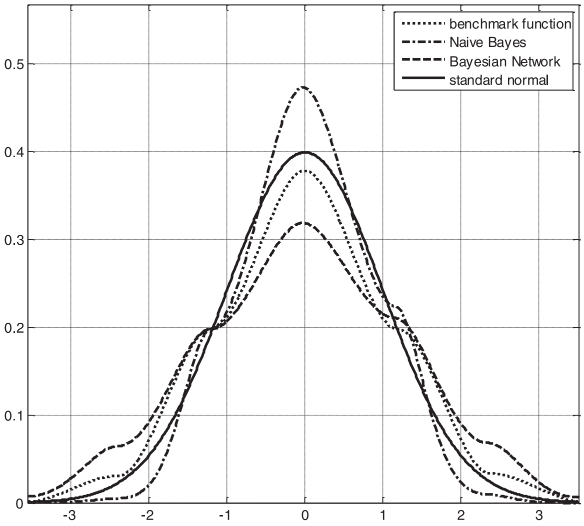

Fig. 5. Distributions function obtained using different methods.

The benchmark function tests the accuracy of an unknown real distribution function. As shown in Fig. 5, there is stronger independence between $B = 1$ and $C = 1$, and $f(v0|target)$ is affected by random factors. Thus, the accuracy of the function based on the naïve Bayes is better than the one based on the Bayesian network. In the following experiments, the benchmark function is a substitute of the statistical results.

The QUAKE data set consists of three conditional attributes and one target attribute. The four attributes can be regarded as real attributes. The earthquakes in the dataset are divided into shallow-focus earthquakes (below 60 km), intermediate-focus earthquakes (between 60 and 300 km) and deep-focus earthquakes (above 300 km) on the basis of the hypocenter depth. The conditional attribute *focal_depth* denotes the depth of the hypocenter, and it is a real attribute. The other two conditional attributes are latitude and longitude. The target attribute *Richter* is an index of earthquake intensity. The task is to predict the distribution function of shallow-focus earthquakes in China. $C = \{0 < focal_depth \le 60,\ 4 < latitude \le 53,\ 73 < longitude \le 135\}$is the condition set in this case. First, using all the instances, the benchmark function is generated by statistical methods. The training set consists of 200 randomly selected instances. The distribution function is then forecasted using the Naive Bayes and the Bayesian network method.

In Fig. 6, the conditional functions used in the naïve Bayesian method as well as the final function (normalized) based on the naïve Bayesian method are. Eq. (11) represents the maximal extension cluster used in the Bayesian network. In Fig. 7, the conditional functions used in the Bayesian network as well as the final function based on the Bayesian network are

Please cite this article as: Z. Lu et al., A Bayesian assumption based forecasting probability distribution model for small samples, Computers and Electrical Engineering (2017), https://doi.org/10.1016/j.compeleceng.2017.11.025

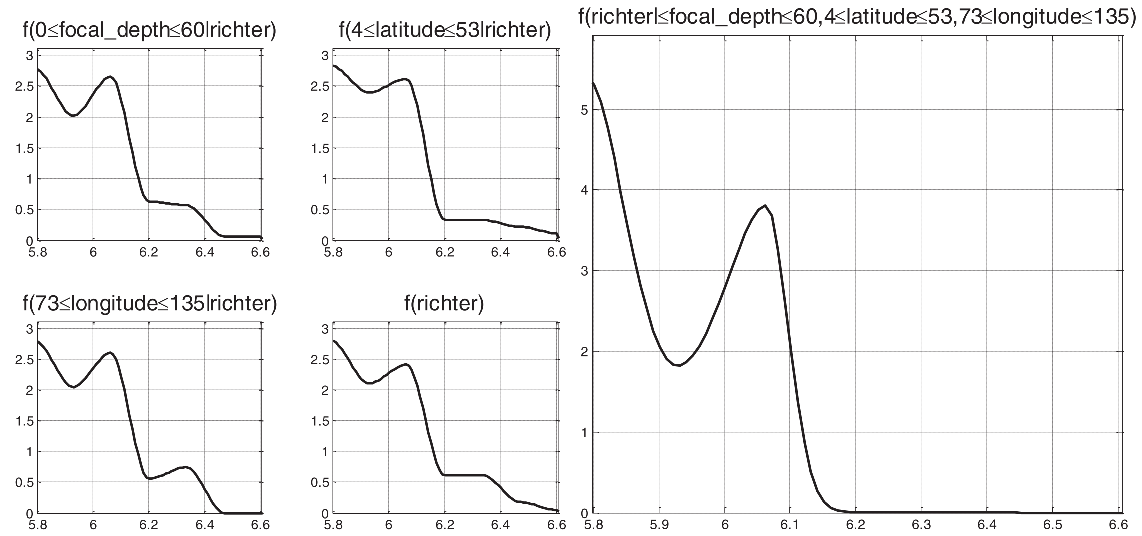

Fig. 6. Conditional functions and final function (right) (naïve Bayes).

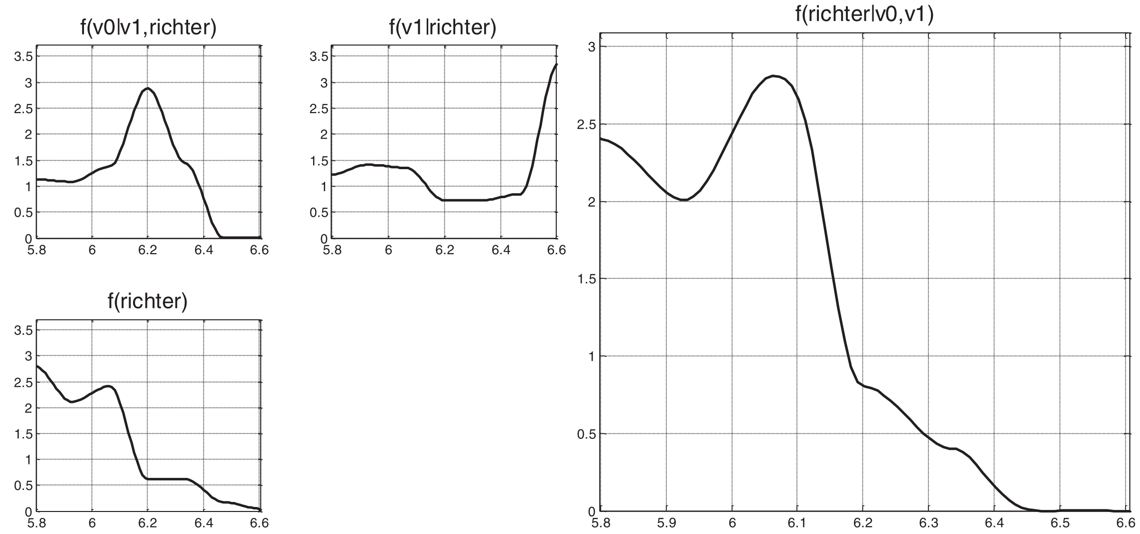

Fig. 7. Conditional functions and final function (right) (Bayesian network).

shown.

$$A_C = \left\{ \begin{matrix} \{0 < focal\_depth \le 60, 73 < longitude \le 135\}, \\ \{0 < focal\_depth \le 60, 4 < latitude \le 53\} \end{matrix} \right\} \tag{11}$$

The similarity scores (defined in Eq. (10)) for the naïve Bayes and the Bayesian network are 0.6854 and 0.8810, respectively. The comparisons are shown in Fig. 7. Intuitively, the distribution function based on the Bayesian network is better than the one based on the Naïve Bayes. This indicates that the attributes are not completely independent. The comparison of the final distribution functions is shown in Fig. 8.

To test the relationship between accuracy and sample size, we used the WAVE-5000 data set in UCI repository. The first three attributes were used as conditional attributes, and the fourth attribute was used as the target attribute. Suppose that all constraints are satisfied in the full sample set. Then, the Benchmark function, whose condition set is $C=\{-1 < x_1 \le 0, 0 < x_2 \le 1, -1<x_3 \le 1\}$, is used to test the effect of different sample sizes. The obtained similarity scores for the naïve Bayes and the Bayesian network are listed in Table 2.

If the size of the sample set is small (such as 100, 150, or 200), the similarity measures of the two methods are almost identical. However, the accuracy of the Bayesian network is improved if the sample size increases to 250. Note that the sample with 400 instances leads to a better result than that with 450 instances. When the number of instances exceeds 400, the maximal extension cluster contains all the conditions. Thus, the formulas used in the function based on the Bayesian

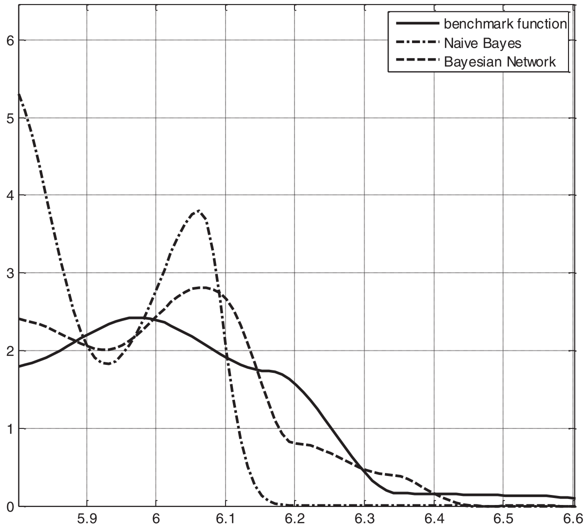

Fig. 8. Comparison of naïve Bayes and Bayesian network against the benchmark function.

Table 2
Similarity scores for varying sample sizes.

Sample size	Naïve Bayes	Bayesian network
100	0.649708	0.649708
150	0.709402	0.709402
200	0.719484	0.719484
250	0.765042	0.916195
300	0.693234	0.919937
350	0.707992	0.923796
400	0.722457	0.938641
450	0.702773	0.925855
500	0.732272	0.924836
550	0.722921	0.931464
600	0.735799	0.938296
650	0.723905	0.936330
700	0.733199	0.943108

network are the same as those used in the bootstrap method in addition to some exceptions. In addition, if the maximal extension sets are not single conditions, the accuracy of the function based on the Bayesian network is greater than 90%. As the sample size becomes larger, the accuracy of the function based on the naïve Bayes is keeping shaking rise. Moreover, the low accuracy may be attributed to the presence of some dependent attributes.

5. Conclusions

In this study, we presented a forecasting probability distribution model. The proposed model outputs the probability distribution of the Bayesian learning model. The model combines cubic spline interpolation with the original Bayesian learning model.

In general, the SSSP forbids the application of the statistical principle. As a result, most recent learning methods cannot handle small samples. The proposed model expands the sample set by loosening the control conditions, in order to meet the statistical requirements. Thus, each sample can provide a distribution that describes the target distribution. Moreover, based on the route of loosening, a Bayesian network is built, wherein each node denotes the probability distribution of the expansion sample. Following this, the Bayesian network is used to calculate the final probability distribution.

Acknowledgments

This work is supported by Science &Technology Project of Civil Aviation Administration of China under grant No. MHRD20140315, Fundamental Research Funds for the Central Universities of CAUC under grant No. 3122013z004, National Key Technology Support Program under grant No. 2014BAJ04B02, Tianjin Application Basis and Cutting-Edge Technology Research under Grant No. 14JCZDJC32500.

References

[1] Liu G, Zhang X, Chen X-Y, et al. Sectioned fitting method of probability distribution function of lightning current amplitude. J South China Univ Technol 2014;42(4):40–5.
[2] Shamshirband S, Petkovic D, Tong CW, et al. Trend detection of wind speed probability distribution by adaptive neuro-fuzzy methodology. Flow Meas Instrum 2015;45(8):43–8.
[3] Albert J, Rooman M. Probability distributions for multimeric systems. J Math Biol 2015;72(1-2):157–69.
[4] Wang C, Zhang J, Chang G, et al. Singular value decomposition projection for solving the small sample size problem in face recognition. J Visual Commun Image Represent 2015;26(8):265–74.
[5] Mao Q-R, Zhao X-L, Bai L-J, et al. Recognition of speech emotion on small samples by over-complete dictionary learning and PCA dimension reduction. J Jiangsu Univ 2013;34(1):60–5.
[6] Wu H-Y, Wang J-M, Dai G-Z. Personalized interaction techniques of vision-based 3D dynamic gestures based on small sample learning. Acta Electron Sin 2013;41(9):2230–6.
[7] Yang Y, Wang X-Q. Attribute reduction based on the grey relational analysis and dynamic programming. In: Natural computation (ICNC), 2013 ninth international conference on. IEEE; 2013. p. 697–701.
[8] Li D-C, Lin L-S, Peng L-J. Improving learning accuracy by using synthetic samples for small datasets with non-linear attribute dependency. Decis Support Syst 2014;59:286–95.
[9] Li D-C, Wen I-H. A genetic algorithm-based virtual sample generation technique to improve small data set learning. Neurocomputing 2014;143:222–30.
[10] David H, Heikki M, Padhraic S. Principles of data mining. The MIT Press; 2001.
[11] Ozsu MT, Valduriez P. Principles of distributed database systems. Prentice Hall, Inc; 1999.
[12] Domingos P, Pazzani M. Beyond independence: conditions for the optimality of the simple Bayesian classifier. In: Proceedings of thirteenth international conference on machine learning, vol. 5; 1996. p. 105–20.
[13] Ching TS, Hamzah NA, Moin NH. An introduction to multiple structural breaks estimation with minimum description length approach. In: AIP conference proceedings, vol. 16(3); 2014. p. 991–6.
[14] Li W-W, Wang J, Fang L, et al. Edge-oriented approach of Bayesian networks based on tabu genetic algorithm. Comput Eng 2009(4):178–80.

Lu Zonglei was born in 1981. He is a Ph.D. in computer applied technology, and he is an associate professor of the Civil Aviation University of China. His current research interests include data mining, machine learning, and knowledge engineering.

Geng Xiaohan was born in 1996. She is a postgraduate in computer science and technology. Her current research interests include data mining, machine learning.

Chen Guoming was born in 1990. He is a master of computer science and technology. His current research interests include data mining, machine learning.

Please cite this article as: Z. Lu et al., A Bayesian assumption based forecasting probability distribution model for small samples, Computers and Electrical Engineering (2017), https://doi.org/10.1016/j.compeleceng.2017.11.025