SQL Codes to Implement the Bayesian Classification

Na LI*, Lin-tong ZHANG and Zhi-gang ZHANG
University of Science and Technology Beijing, Beijing, China

*Corresponding author

Keywords: Bayesian classification, Recursive algorithm, SQL.

Abstract. Bayesian classification is a hotspot of machine learning, the purpose of this paper is to train an efficient Bayesian classification based on SQL. It expounded the Bayes' theorem and the concept of Bayes classifier as the foundation, and then, used the recursive algorithm to create a Bayesian data set; the SQL query codes to count classification properties, decision attribute values and class-condition in the training sample, and to calculate the corresponding probability; and the SQL query - update codes to mark the most optimal decision value, thus, the high accuracy of Bayesian classifier model had been trained. Further, we applied the above classification system to predict and determinate the students' grades. It is preliminarily shown that Bayesian classification method has better application prospect in predicting students’ grades.

Introduction

Bayesian decision theory is the basic method of decision-making in probability framework. Its classification task is based on the given probability and misjudgment loss to select the most optimal classification marks. This paper will introduce a Bayesian classifier system—apply the students' given grades to train out a Bayesian classifier with a higher performance, thus, by doing this, the students’ grades can be well predicted, and such Bayesian classifier has a higher rate of anticipation.

Bayesian Classifier

Bayesian Decision Theory

The principle of the Bayesian decision theory \([1, 3, \text{and} 4]\) is to decide the credibility of discriminating as a certain decision value by calculating the conditional probability of an event, this ultimate result is the probability distribution of random variables. Bayesian decision theory, based on the Bayesian theory, will apply the prior probability to calculate the posterior probability. Finally, it will make a reasonable decision by calculating the probability.

Bayes' theorem is used to describe the relationship between two conditional probabilities:

\[
P(Y = y | X_1 = x_1, X_2 = x_2, \ldots, X_n = x_n) = \frac{P(X_1 = x_1, X_2 = x_2, \ldots, X_n = x_n, Y = y)}{P(X_1 = x_1, X_2 = x_2, \ldots, X_n = x_n)}
\]

\[
= \frac{P(Y = y)P(X_1 = x_1, X_2 = x_2, \ldots, X_n = x_n | Y = y)}{P(X_1 = x_1, X_2 = x_2, \ldots, X_n = x_n)}
\]

\[
\text{(1)}
\]

\(Y\) : the decision classes; \(y\) : the decision values; \(x_i\) : the attribute values in the \(i\)th attribute \(X_i\) of the sample \(X\). \((i = 1, 2, \ldots, n)\)

There into, \(P(Y = y)\) is the class “prior” probability: \(P(X_1 = x_1, X_2 = x_2, \ldots, X_n = x_n | Y = y)\) is the class-conditional probability of sample \(X\) relative to class notation \(Y\); \(P(X_1 = x_1, X_2 = x_2, \ldots, X_n = x_n)\) is the "evidence" factor for normalization, that is, the same property values for the same attribute of the sample \(X\) can be grouped together. Therefore, the question of evaluating \(P(Y = y | X_1 = x_1, X_2 = x_2, \ldots, X_n = x_n)\) is converted to estimate class” prior” probability
\( P(Y = y) \) and class-conditional probability \( P(X_1 = x_1, X_2 = x_2, \ldots, X_n = x_n | Y = y) \) based on training samples \( X \).

**Naive Bayesian Classifier**

According to formula (Eq. 1), the difficulty of obtaining the posterior probability \( P(Y = y | X_1 = x_1, X_2 = x_2, \ldots, X_n = x_n) \) is that: the class-conditional probability \( P(X_1 = x_1, X_2 = x_2, \ldots, X_n = x_n | Y = y) \) is the joint probability of all attributes. Therefore, we make an "attribute conditional independence assumption"[2], that is, each property is independent of each other, and the classifier under this assumption is called the naive Bayesian classifier.

\[
P(Y = y | X_1 = x_1, X_2 = x_2, \ldots, X_n = x_n) = \frac{P(Y = y) P(X_1 = x_1, X_2 = x_2, \ldots, X_n = x_n)}{P(X_1 = x_1, X_2 = x_2, \ldots, X_n = x_n)} = \frac{P(Y = y)}{P(X_1 = x_1, X_2 = x_2, \ldots, X_n = x_n)} \prod_{i=1}^{n} P(X_i = x_i | Y = y) \quad (2)
\]

\( Y \): the decision classes; \( y \): the decision values; \( X_i \): the attribute values in the \( i \)th attribute \( X_i \) of the sample \( X \). (\( i = 1, 2, \ldots, n \))

Therefore, the expression of the naive Bayesian classifier [5] is:

\[
h_{\text{naive}}(X_1 = x_1, X_2 = x_2, \ldots, X_n = x_n) = \arg \max_{y \in \text{Y}} P(Y = y) \prod_{i=1}^{n} P(X_i = x_i | Y = y). \quad (3)
\]

**Bayesian Classifier Instance**

The following table is an existing data examples (Table 1), the method is to create a Bayesian data set by using the Cartesian product operation, and to train a naive Bayesian classifier by the relationship between the classification attributes and decision attributes, thus, the data are optimally discriminated:

<table>
<thead>
<tr>
<th>ID</th>
<th>House</th>
<th>Marriage</th>
<th>Default Loans</th>
<th>ID</th>
<th>House</th>
<th>Marriage</th>
<th>Default Loans</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Yes</td>
<td>Single</td>
<td>No</td>
<td>6</td>
<td>No</td>
<td>Married</td>
<td>No</td>
</tr>
<tr>
<td>2</td>
<td>No</td>
<td>Married</td>
<td>No</td>
<td>7</td>
<td>Yes</td>
<td>Divorced</td>
<td>No</td>
</tr>
<tr>
<td>3</td>
<td>No</td>
<td>Single</td>
<td>No</td>
<td>8</td>
<td>No</td>
<td>Single</td>
<td>Yes</td>
</tr>
<tr>
<td>4</td>
<td>Yes</td>
<td>Married</td>
<td>No</td>
<td>9</td>
<td>No</td>
<td>Married</td>
<td>No</td>
</tr>
<tr>
<td>5</td>
<td>No</td>
<td>Divorced</td>
<td>Yes</td>
<td>10</td>
<td>No</td>
<td>Single</td>
<td>Yes</td>
</tr>
</tbody>
</table>

First step: estimate the class "prior" probability \( P(Y = y) \), which represents the classification attribute of "default loans":

\[
P(\text{default loans} = "Yes") = \frac{3}{10} = 0.3,
\]

\[
P(\text{default loans} = "No") = \frac{7}{10} = 0.7,
\]

Second step: by using the formula (Eq. 1), determine the conditional probability of each attribute \( P(X_i = x_i | Y = y) \), \( i = 1, 2 \). Here \( X_1 \) is the attribute of "house", and \( X_2 \) is the attribute of "marriage":

\[
P_{\text{own-house}} = P(\text{house} = "Yes" | \text{default loans} = "No") = \frac{3}{7} \approx 0.429,
\]
\[ P_{\text{Single} = \text{"Yes"}} = P(\text{Marriage} = \text{"Single"} | \text{default loans} = \text{"Yes"}) = \frac{2}{3} \approx 0.667, \]
\[ P_{\text{Single} = \text{"No"}} = P(\text{Marriage} = \text{"Single"} | \text{default loans} = \text{"No"}) = \frac{2}{7} \approx 0.286, \]

Third step: by using the formula (Eq. 2), make decisions on the sample data set:

\[ P(\text{default loans} = \text{"Yes"}) \times P_{\text{own-house} = \text{"Yes"}} \times P_{\text{Single} = \text{"Yes"}} = 0, \]
\[ P(\text{default loans} = \text{"No"}) \times P_{\text{own-house} = \text{"No"}} \times P_{\text{Single} = \text{"No"}} \approx 0.086, \]

“own-house” and “single” people have a larger probability of not-defaulting on the loans, therefore, the naive Bayesian classifier discriminates the “own-house” and “single” people as "not-default loans".

**SQL Implements Bayes Classifier**

**Algorithm Principle**

In order to better present the thorough consideration of Bayes classification, the first step, use recursive algorithm to compute the Cartesian product of multiple attribute values corresponding to each attribute in the sample, and then, create a data set of Bayesian classification, this step is the foundation of the success within the whole algorithm; the second step, for each case in the data set, use the SQL codes to query the records of the classification attribute, decision values and the class-condition in the sample table, and then, count and calculate the corresponding probability, this step is the core of the success within the whole algorithm; the third step, the assumptions is the "naive Bayes classifier", therefore, calculate the joint probability of a given decision value, (which is the continuous multiplication of the class-conditional probability) take the most probable decision value, and finally, the optimization of Bayesian classifier is realized.

**Algorithm Implementation**

The algorithm flow chart is as follows (Figure 1):

![Figure 1. A flow chart about Bayes classification algorithm.](image)
Some SQL codes are as follows:
1. Delete the Bayesian data table
   ```sql
   drop table FL_Bayes_Demo
   ```
2. Create the Bayesian data table
   ```sql
   create table FL_Bayes_Demo ( ID integer, X1 varchar(10), X2 varchar(10), Y varchar(10),
   mX integer, pX float, mY integer, pY float, mY_x integer, pY_x float, mX1_y integer, mX2_y integer,
   pX1_y float, pX2_y float, B_pY_X float, cY X varchar(1))
   ```
   The meaning of each field is:
   - \(X_1\): availability of housing; \(X_2\): marital status; \(Y\): the situation of delinquent loans;
   - \(x_1\): attribute values of housing “Yes/No”;
   - \(x_2\): attribute values of marital status “single/married/divorced”;
   - \(mX\): count of \(X_1 = x_1, X_2 = x_2\);
   - \(pX\): frequency of \(X_1 = x_1, X_2 = x_2\);
   - \(y\): the decision value of the default loans “Yes/No”;
   - \(mY\): count of \(Y = y\);
   - \(pY\): frequency of \(Y = y\);
   - \(mY_x\): under the condition of \(X_1 = x_1, X_2 = x_2\), calculate the count of \(Y = y\);
   - \(pY_x\): under the condition of \(X_1 = x_1, X_2 = x_2\), calculate the frequency of \(Y = y\);
   - \(mX1_y\): under the condition of \(Y = y\), calculate the count of \(X_1 = x_1\);
   - \(mX2_y\): under the condition of \(Y = y\), calculate the count of \(X_2 = x_2\);
   - \(B_pY_X\): apply formula (Eq. 3), to Continuous multiplication \(pY \times pX1 \times pX2 \times Y\);
   - \(cY X\): apply formula (Eq. 3), Take the same property value \(X_1 = x_1, X_2 = x_2\), the maximum value under different decision values \(Y\) is used to the classification.
3. Create a Bayesian classification data set —— recursive algorithm
   The principle of this algorithm is to create a custom recursive function that iterates through each attribute value of each attribute from \(X_1, \ldots, X_n\) in turn. In the loop, when \(k = 1, \ldots, n - 1\), the next attribute \(k + 1\) is retrieved by calling the recursive function. When \(k = n\), recursive algorithm has been finished, and then, output the previous \(n\) traversal values, and create the Bayesian classification data set at this time (Table 2).

   The example flow chart is as follows:

   ![Flow chart about recursive algorithm](Figure 2)
Table 2. Create a Bayesian classification data set.

<table>
<thead>
<tr>
<th>ID</th>
<th>X1</th>
<th>X2</th>
<th>Y</th>
<th>ID</th>
<th>X1</th>
<th>X2</th>
<th>Y</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>No</td>
<td>Single</td>
<td>No</td>
<td>7</td>
<td>Yes</td>
<td>Single</td>
<td>No</td>
</tr>
<tr>
<td>2</td>
<td>No</td>
<td>Single</td>
<td>Yes</td>
<td>8</td>
<td>Yes</td>
<td>Single</td>
<td>Yes</td>
</tr>
<tr>
<td>3</td>
<td>No</td>
<td>Divorced</td>
<td>No</td>
<td>9</td>
<td>Yes</td>
<td>Divorced</td>
<td>No</td>
</tr>
<tr>
<td>4</td>
<td>No</td>
<td>Divorced</td>
<td>Yes</td>
<td>10</td>
<td>Yes</td>
<td>Divorced</td>
<td>Yes</td>
</tr>
<tr>
<td>5</td>
<td>No</td>
<td>Married</td>
<td>No</td>
<td>11</td>
<td>Yes</td>
<td>Married</td>
<td>No</td>
</tr>
<tr>
<td>6</td>
<td>No</td>
<td>Married</td>
<td>Yes</td>
<td>12</td>
<td>Yes</td>
<td>Married</td>
<td>Yes</td>
</tr>
</tbody>
</table>

4. Select the count and value of the decision attributes \( mY \)
   
   select distinct default_loans, count(*) as con from Bayes_Demo order by default_loans

5. Select the count of each combination of the classification attributes \( mY \_X \)
   
   select house, marriage, count(*) as con from Bayes_Demo
   group by house, marriage order by house, marriage

6. Select the count of each value of the decision attributes \( mX_1 \_y \) \( mX_2 \_y \)
   
   select default_loans, count(*) as con from Bayes_Demo
   group by default_loans order by default_loans

7. Calculate the total number of sample tables
   
   select count(*) as con from Bayes_Demo

8. Update the count of each combination of the classification attributes in the Bayesian data table \( mX \)
   
   update FL_Bayes_Demo set mX=1, pX=0.1 where Y=Y and X1='Yes' and X2='Marriage'

9. Update the frequency of each combination of the classification attributes and decision attributes
   in the Bayesian data table \( pY \_X \)
   
   update FL_Bayes_Demo set \( pY \_X = mY \_X / mX \) where mX>0

10. Update the probability of each combination of the classification attributes and decision attributes in the Bayesian data table \( B \_ pY \_X \)
    (According to formula (Eq. 3), the probability: \( P(Y = y) \prod_{i=1}^{n} P(X_i = x_i \mid Y = y) \))
    
   update FL_Bayes_Demo set \( B \_ pY \_X = pY \_X \_Y \_Y \_X \_Y \_X \_X \_Y \)

11. According to the maximum value of the classification probability of each combination, update the final form
    (According to formula (Eq. 3), that is to select: \( \max_{y \in Y} P(Y = y) \prod_{i=1}^{n} P(X_i = x_i \mid Y = y) \))
    
   The first step: select the maximum of joint probability \( B \_ pY \_X \) in the Bayesian data table,

   SQL codes are:
   select max(B\_pY\_X) from FL_Bayes_Demo where X1=A\_X1 and X2=A\_X2

   The second step: select the record of the maximum of joint probability \( B \_ pY \_X \), SQL codes are:
   select X1,X2,Y from FL_Bayes_Demo A where B\_pY\_X=
   (select max(B\_pY\_X) from FL_Bayes_Demo where X1=A\_X1 and X2=A\_X2 )
   order by X1,X2,Y

   The third step: update \( cY \_X \), and mark \( cY \_X \) at the maximum of \( B \_ pY \_X \), SQL codes are:
   update FL_Bayes_Demo A set cY\_X='*' where B\_pY\_X=
   (select max(B\_pY\_X) from FL_Bayes_Demo where X1=A\_X1 and X2=A\_X2 )

**Algorithm Analysis**

The results of Bayesian data set are as follows:
From the analysis of the results, it can be seen that "no-house &"single", and "no-house &"divorced", which can be discriminated as "default loans" by the naive Bayes classifier; the "married" can be discriminated as "not-default loans" by the classifier; The above data shows that, "own-house" is a sufficient but unnecessary condition for "not-default loans"; At the same time, by comparing the data set with the original data, the classifier has a high accuracy rate of 90%, so it has high practical value.

Algorithm Application

The above classification system can be applied to students' grades, which can implement the prejudgment of the “Probability Theory” courses, and the results are as follows:

<table>
<thead>
<tr>
<th></th>
<th>X1</th>
<th>X2</th>
<th>X3</th>
<th>Y</th>
<th>mX</th>
<th>pX</th>
<th>mY</th>
<th>mY_x</th>
<th>pY_x</th>
<th>mY pY</th>
<th>B_pY</th>
<th>cY X</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Fail</td>
<td>Fail</td>
<td>Fail</td>
<td>Fail</td>
<td>18</td>
<td>0.009</td>
<td>9</td>
<td>0.333</td>
<td>0.667</td>
<td>107</td>
<td>0.581</td>
<td>0.419</td>
</tr>
<tr>
<td>2</td>
<td>Fail</td>
<td>Fail</td>
<td>Fail</td>
<td>pass</td>
<td>18</td>
<td>0.009</td>
<td>10</td>
<td>0.333</td>
<td>0.667</td>
<td>107</td>
<td>0.581</td>
<td>0.419</td>
</tr>
<tr>
<td>3</td>
<td>Fail</td>
<td>Fail</td>
<td>pass</td>
<td>Fail</td>
<td>61</td>
<td>0.025</td>
<td>26</td>
<td>0.667</td>
<td>0.333</td>
<td>64</td>
<td>0.305</td>
<td>0.695</td>
</tr>
<tr>
<td>4</td>
<td>Fail</td>
<td>Fail</td>
<td>pass</td>
<td>pass</td>
<td>61</td>
<td>0.025</td>
<td>35</td>
<td>0.667</td>
<td>0.333</td>
<td>131</td>
<td>0.907</td>
<td>0.093</td>
</tr>
<tr>
<td>5</td>
<td>Fail</td>
<td>pass</td>
<td>Fail</td>
<td>Fail</td>
<td>9</td>
<td>0.004</td>
<td>2</td>
<td>0.222</td>
<td>0.778</td>
<td>64</td>
<td>0.305</td>
<td>0.695</td>
</tr>
<tr>
<td>6</td>
<td>Fail</td>
<td>pass</td>
<td>Fail</td>
<td>pass</td>
<td>9</td>
<td>0.004</td>
<td>7</td>
<td>0.222</td>
<td>0.778</td>
<td>131</td>
<td>0.907</td>
<td>0.093</td>
</tr>
<tr>
<td>7</td>
<td>Fail</td>
<td>pass</td>
<td>pass</td>
<td>Fail</td>
<td>107</td>
<td>0.048</td>
<td>28</td>
<td>0.667</td>
<td>0.333</td>
<td>64</td>
<td>0.305</td>
<td>0.695</td>
</tr>
<tr>
<td>8</td>
<td>Fail</td>
<td>pass</td>
<td>pass</td>
<td>pass</td>
<td>107</td>
<td>0.048</td>
<td>37</td>
<td>0.667</td>
<td>0.333</td>
<td>131</td>
<td>0.907</td>
<td>0.093</td>
</tr>
<tr>
<td>9</td>
<td>pass</td>
<td>Fail</td>
<td>Fail</td>
<td>Fail</td>
<td>20</td>
<td>0.009</td>
<td>12</td>
<td>0.667</td>
<td>0.333</td>
<td>131</td>
<td>0.907</td>
<td>0.093</td>
</tr>
<tr>
<td>10</td>
<td>pass</td>
<td>Fail</td>
<td>Fail</td>
<td>pass</td>
<td>20</td>
<td>0.009</td>
<td>8</td>
<td>0.667</td>
<td>0.333</td>
<td>131</td>
<td>0.907</td>
<td>0.093</td>
</tr>
<tr>
<td>11</td>
<td>pass</td>
<td>Fail</td>
<td>pass</td>
<td>Fail</td>
<td>162</td>
<td>0.072</td>
<td>42</td>
<td>0.667</td>
<td>0.333</td>
<td>131</td>
<td>0.907</td>
<td>0.093</td>
</tr>
<tr>
<td>12</td>
<td>pass</td>
<td>Fail</td>
<td>pass</td>
<td>pass</td>
<td>162</td>
<td>0.072</td>
<td>42</td>
<td>0.667</td>
<td>0.333</td>
<td>131</td>
<td>0.907</td>
<td>0.093</td>
</tr>
<tr>
<td>13</td>
<td>pass</td>
<td>Fail</td>
<td>Fail</td>
<td>Fail</td>
<td>33</td>
<td>0.015</td>
<td>6</td>
<td>0.667</td>
<td>0.333</td>
<td>131</td>
<td>0.907</td>
<td>0.093</td>
</tr>
<tr>
<td>14</td>
<td>pass</td>
<td>Fail</td>
<td>Fail</td>
<td>pass</td>
<td>33</td>
<td>0.015</td>
<td>47</td>
<td>0.667</td>
<td>0.333</td>
<td>131</td>
<td>0.907</td>
<td>0.093</td>
</tr>
<tr>
<td>15</td>
<td>pass</td>
<td>Fail</td>
<td>pass</td>
<td>Fail</td>
<td>1842</td>
<td>0.081</td>
<td>88</td>
<td>0.667</td>
<td>0.333</td>
<td>131</td>
<td>0.907</td>
<td>0.093</td>
</tr>
<tr>
<td>16</td>
<td>pass</td>
<td>Fail</td>
<td>pass</td>
<td>pass</td>
<td>1842</td>
<td>0.081</td>
<td>88</td>
<td>0.667</td>
<td>0.333</td>
<td>131</td>
<td>0.907</td>
<td>0.093</td>
</tr>
</tbody>
</table>

$X_1$: a passing situation of "Advanced Mathematics I";( classification attribute)

$X_2$: a passing situation of "Advanced Mathematics II";( classification attribute)

$X_3$: a passing situation of "Linear Algebra";( classification attribute)

$Y$: a passing situation of “Probability Theory”.(decision attribute)

With the same classification system, $X_1$, $X_2$, $X_3$ are as classification attributes, $Y$ is as a decision attribute, a Bayesian classifier was trained by the grades of the previous three courses, in order to prejudge the passing situation of $Y$ “ probability theory “ course.

From the analysis of Table 4, there are 2252 students, the number of “Probability Theory” result discriminant right is 2030, so the Bayesian classifier in this set of data has a discriminant accuracy rate of 90.14%,it is a high accuracy; At the same time, While from the intuitive, under the condition of “Advanced Mathematics I” fail, "Advanced Mathematics II” passed, "Linear Algebra" fail, the number of students who pass the “Probability Theory” is more, but by discriminant classifier, it will predict “Probability Theory” as "fail". By careful analysis, the reason is that the number of people who don’t pass it is relatively small, therefore, the probability of the joint probability is larger, and so, "probability theory" exam will be judged as “fail”.

Summary and Conclusion

This paper uses a training data set, program the recursive algorithm, and SQL query codes to select the related records about the count and the frequency. Finally, by using SQL query – update codes,
the maximum value of combinatorial classification probability is found, and a Bayesian classifier with 90% accuracy can be trained. Further, under the condition of classification attributes ("Advanced Mathematics I", "Advanced Mathematics II", "Linear Algebra"), by applying such classification system, an efficient Bayesian classifier can be trained. It can prejudge the situation of "probability theory" grades, by compared with the actual result, the forecasting results’ accuracy is as high as 90.14%.

Therefore, the Bayesian classifier model mainly has the following obvious advantages: 1) the attributes of the sample are easy to extract, the complexity is low, and the processing speed is fast. 2) using naive Bayes classification algorithm, the model is simple and easy to implement. There are some defects in this algorithm: 1) low classification attribute, not very general; (2) the Bayesian classifier, although simple, is not necessarily the optimal classification algorithm. It should be compared with other algorithms.

Generally, a relatively new classification system has been introduced in this paper, under the condition of given classification attributes and decision attributes, can train a kind of high precision Bayesian classifier to predict student grades, it has the high availability and ease of use.

Acknowledgements

This article is funded by the Beijing University of Science and Technology Education Research Project (JG2017Z10).

References


