An iterative improvement approach for the discretization of numeric attribute in Bayesian classifiers

Yue Zhang
Professor: Dr. Zaiane
Computing Science Department
University of Alberta

- **Introduction to Classification**
- **Introduction to Bayesian Classifier**
- **Iterative approach dealing with the numeric attributes using Bayesian Classifier**
- **Conclusion**

Introduction to Classification

- Definition of Classification
- A two step process
- Numerous application
- Criteria evaluating the Classifier

Introduction to Classification(Con't)

- Predictive accuracy
- speed
- robustness
- scalability
- interpretability

Introduction to Bayesian Classifier

- Bayes Theorem

X: a data sample whose class label is unknow

H: the hypothesis that X belongs to a specific class

$$P(H|X) = P(X|H)\,P(H) / P(X)$$

- Naïve Bayesian Classifier

Naïve Bayesian Classifier

•Sample, idiot Classifier

•Class conditional independence: Assume that the effect of an attribute value on a give class is independent of the values of the other attributes.

•The probability of each class may be computed and the example assigned to the class with the highest probability.

An iterative approach dealing with the numeric attributes

- Abstract
- Introduction
- Numeric Values in Bayesian Classifiers
- Deiscetizing numeric variables
- Searching for boundaries
- Expert defined thresholds

Abstract

- There are lots of numeric attributes in the real case
- Discretization of numeric attributes is critical to successful application of the Bayesian classifier
- A new method based on iterative improvement search was proposed in the paper

Introduction to the case

- Discovering how to troubleshoot a telephone network using a database of repair records.
- Four classes in the case: PDI, PDF, PDO, PDT
- There are lots of numeric attributes in this case

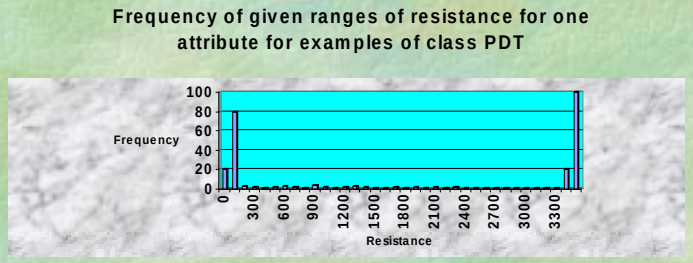
Reason of Selecting Naïve Bayesian

- It is accurate
- it can determine the most likely class of a training example
- The result of the Bayesian classifier is simple
- It is possible to take advantage of expert knowledge on the critical values of continuous variables

Numeric Values in Bayesian Classifiers

- It is easy when attributes have no numeric attributes
- One method to deal with numeric attributes: find the mean value of each numeric attribute and use this information to determine the probability that an attribute had a given value

The method doesn't work well

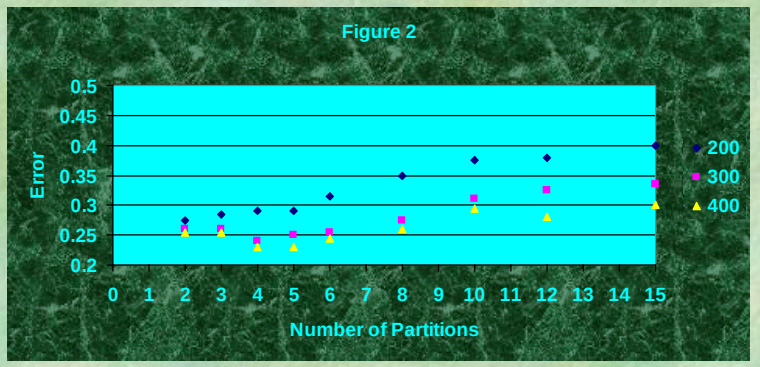


We can see sometimes the numeric attribute is not normally distributed

Discretizing numeric variables

- Discrete the numeric variables in Bayesian Classifiers into small partitions
- Treat these partitions as nominal values.
- Such as: First-Quarter, Second-Quarter, Third-Quarter and Fourth-Quarter instead of 12 months in one year.

Error - Partitions



Error- Partition (Con't)

- The choice of P has a major effect on the error rate. Too small or too large a value the P has chosen will not give us the good result
- The best value for P depends upon the size of the training set

Problems with partitioning data

- It's not clear how to choose the best value for P
- it's not clear that the same value for P should be used for every attribute.
- The approach doesn't look for critical values of the variable, but divides the variable into evenly spaced partitions

Searching for boundaries

- The problem of finding a good set of boundary points to discretize values for numeric attributes can be viewed as a search problem.
- Hard to generate all test all such boundary points since each value of each attribute of each example is a potential boundary point

Iterative approach for finding the best boundary point

- There are two operators that adjust boundary points:
  - merge two contiguous interval.
  - Split an interval into two intervals by considering introducing a new boundary point that is midway between every pair of contiguous attribute values within that interval.

Procedure of the method

1) Estimate the error (or misclassification cost) of each adjustment using LOOCV of the current set of boundary points, and reorder the examples such that those that are misclassified

2) Reorder the attributes randomly

Procedure of the method (con'd)

c) for each attribute in the set of attributes

1)Apply all operators in all possible ways to the current boundary points of the attribute.

2) Estimate the error (or misclassification cost) of each adjustment using LOOCV.

3) if the error of any adjustment is less than the error of the current boundary points. Then make that adjustment with the lowest error.

Procedure of the method (con'd)

d) if no boundary point was adjusted in Step C then return the current boundary points Else go to step A

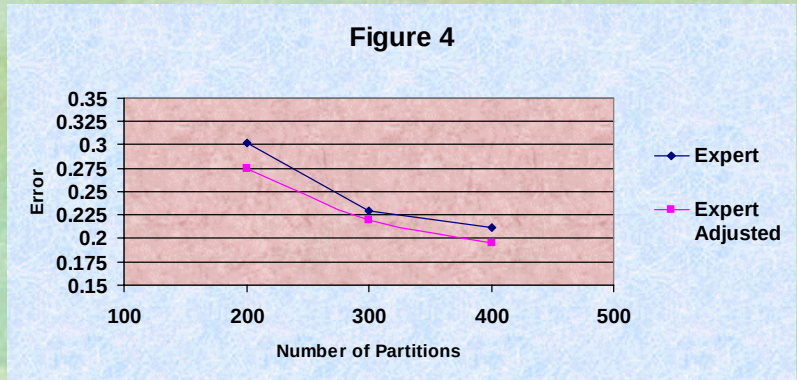
Efficiency issues make the algorithm practical in large data

- Make one change to each attribute in Step C rather than making the single change to a single attribute
- attributes are reordered randomly before the loop is executed
- LOOCV is used to estimate error
- examples that are misclassified are tested first

Expert defined thresholds

- Let the expert to select the boundary points
- It works well eventhough it is better for using adjustment process on it.
- We can see that results in the figure below

Expert defined boundary points



Conclusions

- An iterative improvement approach to discretizing a set of numeric attributes for use in a Bayesian classifier was proposed in this paper
- Finding the good set of boundary point is the most important
- It is efficient