Experiment No.2

Title: Implementation of Naïve Bayesian algorithm for classification

Batch: **Roll No.:** **Experiment No.:2**

Aim: Implementation of Naïve Bayesian algorithm for classification

---

Resources needed: Any RDBMS, Java

Theory:

A Bayesian classifier is a simple probabilistic classifier. Bayesian classifier can predict membership probabilities such as the probabilities that a sample belongs to a particular class or groupings.

Bayesian classification is based on Bayes theorem and this technique tends to be highly accurate and fast, making it useful on large databases.

Naïve Bayesian Classification Algorithm:

The operation of the Naïve Bayesian is as follows,
1) Let *D* be a training set of tuples and their associated class labels. As usual, each tuple is represented by an *n*-dimensional attribute vector, $\boldsymbol{X} = (x1, x2, :::, xn)$, depicting *n* measurements made on the tuple from *n* attributes, respectively, $A1, A2, :::, An$.

2) Suppose that there are m classes C1,C2,…….,Cm, Given a tuple, ***X***, the classifier will predict that ***X*** belongs to the class having the highest posterior probability, conditioned on ***X***. That is, the na¨ıve Bayesian classifier predicts that tuple ***X*** belongs to the class *Ci* if and only if,

$$P(C_i|X) > P(C_j|X) \quad \text{for } 1 \leq j \leq m, j \neq i.$$

The class *Ci* for which $P(Ci\,|\boldsymbol{X})$ is maximized is called the *maximum posteriori hypothesis*.

3) Using Bayes' theorem,

$$P(C_i|\boldsymbol{X}) = \frac{P(\boldsymbol{X}|C_i)P(C_i)}{P(\boldsymbol{X})}.$$

As $P(\boldsymbol{X})$ is constant for all classes, only $P(\boldsymbol{X}/C_i)P(C_i)$ needs to be maximized. If the class prior probabilities are not known, then it is commonly assumed that the classes are equally likely, that is, $P(C_1)= P(C_2)= \ldots\ldots = P(C_m)$, and we would therefore maximize $P(\boldsymbol{X}/C_i)$. Otherwise, we maximize $P(\boldsymbol{X}/C_i)P(C_i)$. Note that the class prior probabilities may be estimated by $P(C_i)= |C_{i,D}| / |D|$, where $|Ci,D|$ is the number of training tuples of class C_i in D.

This presumes that the attributes' values are conditionally independent of one another, given the class label of the tuple (i.e., that there are no dependence relationships among the attributes). Thus,

$$P(X|C_i) = \prod_{k=1}^{n} P(x_k|C_i)$$
$$= P(x_1|C_i) \times P(x_2|C_i) \times \cdots \times P(x_n|C_i)$$

We can easily estimate the probabilities $P(x_1/C_i)$, $P(x_2/C_i)$, …… , $P(x_n/C_i)$ from the training tuples. Recall that here x_k refers to the value of attribute A_k for tuple $\boldsymbol{X}$. For each attribute, we look at whether the attribute is categorical or continuous-valued.

4) Sample X is therefore assigned to class Ci if and only if P(X/Ci).P(Ci)>P(X/Cj).P(Cj) for i<=j<=m. y≠1 In other words if it is assigned to the class C for which P(X/Ci).P(Ci) is Max.

Procedure / Approach /Algorithm / Activity Diagram:

1. Identify attributes suitable for applying classification algorithm
2. Implement **Naïve Bayesian** on your dataset.
3. Apply **Naïve Bayesian** to classify unknown tuple.

Results: (Program printout with output / Document printout as per the format)

Questions:

1. What are advantages and disadvantages of Bayesian Classification?
2. Comment on Laplacian correction.

Outcomes:

Conclusion: (Conclusion to be based on the objectives and outcomes achieved)

Grade: AA / AB / BB / BC / CC / CD /DD

Signature of faculty in-charge with date

References:

Books/ Journals/ Websites:

1. Han, Kamber, "Data Mining Concepts and Techniques", Morgan Kaufmann 3rd Edition