NAÏVE BAYES CLASSIFIER

Professor Tom Fomby
Department of Economics
Southern Methodist University
Dallas, Texas 75275

April 2008

The **Naïve Bayes classifier** is a classification method based on Bayes Theorem. Let C_j denote that an output belongs to the j-th class, $j = 1,2\cdots, J$ out of J possible classes. Let $P(C_j \mid X_1, X_2, \cdots, X_p)$ denote the (posterior) probability of belonging in the j-th class given the individual characteristics $X_1, X_2, \cdots, X_p$. Furthermore, let $P(X_1, X_2, \cdots, X_p \mid C_j)$ denote the probability of a case with individual characteristics $X_1, X_2, \cdots, X_p$ belonging to the j-th class and $P(C_j)$ denote the unconditional (i.e. without regard to individual characteristics) prior probability of belonging to the j-th class. For a total of J classes, Bayes theorem gives us the following probability rule for calculating the case-specific probability of falling into the j-th class:

$$P(C_j \mid X_1, X_2, \cdots, X_p) = \frac{P(X_1, X_2, \cdots, X_p \mid C_j) \cdot P(C_j)}{Denom} \qquad (1)$$

where $Denom = P(X_1, X_2, \cdots, X_p \mid C_1)P(C_1) + \cdots + P(X_1, X_2, \cdots, X_p \mid C_J)P(C_J)$. Of course the conditional class probabilities of (1) are exhaustive in that a case $(X_1, X_2, \cdots, X_p)$ has to fall in one of the J cases. That is, $\sum_{j=1}^{J} P(C_j \mid X_1, X_2, \cdots, X_p) = 1$.

The difficulty with using (1) is that in situations where the number of cases $(X_1, X_2, \cdots, X_p)$ is few and distinct and the number of classes J is large, there may be many instances where the probabilities of cases falling in specific classes, $P(X_1, X_2, \cdots, X_p \mid C_j)$, are frequently equal to zero for the majority of classes. This gives rise to the frequent occurrences of zero probabilities for the case-specific class probabilities of equation (1) excepting maybe for one or two classes. That is, one would have a classifier where the probability of falling in one class or only a few classes would be "all or none." Moreover, when applying a trained (on the training data set) Bayes Classifier (1) to an independent data set (i.e. the validation or test data sets) it could likely be the case that some of the cases $(X_1, X_2, \cdots, X_p)$ that occur in the independent data set do not appear in the training data set. In this case, then, how would one apply (1) in scoring the independent data set?

One way to overcome the above mentioned shortcomings is to make the classifier in equation (1) **"naïve"** in that one can assume that the inputs $X_1, X_2, \cdots, X_p$ are

independent of each other. This independence then allows us to calculate the case-specific class probabilities as in

$$P(X_1, X_2, \cdots, X_p \mid C_j) = P(X_1 \mid C_j) \cdot P(X_2 \mid C_j) \cdots P(X_p \mid C_j). \qquad (2)$$

Then in the independent case the terms on the right-hand-side of (2) can be calculated simply as the relative frequencies of the individual X_i's in the class C_j. For example, the training data set could be used to calculate the relative frequency

$$P(X_i \mid C_j) = [(\#\text{ of } X_i \text{ in } C_j)/(\text{total } \#\text{ of cases in } C_j)]. \qquad (3)$$

Obviously, holding other factors constant, the likelihood that the case-specific class probabilities $P(X_1, X_2, \cdots, X_p \mid C_j)$ turn out to be zero is reduced (both for the training data set as well as the independent data sets) and thus the posterior class probabilities are likely to be less sparse (i.e. equal to zero) across the various classes for a given input characteristic $(X_1, X_2, \cdots, X_p)$. Of course the independence assumption is very simplistic since the inputs (attributes) $X_1, X_2, \cdots, X_p$ are very likely to be correlated. However, quite surprisingly, this "Naïve Bayes" approach often works quite well in practice.

In the XLMINER program the training of the Naïve Bayes classifier proceeds in the following manner: The prior class probabilities $P(C_j), j = 1,2,\cdots, J$ are determined in one of two ways by the user. (i) Either the user specifies that the prior class probabilities are equal to each other as in

$$P(C_j) = 1/J, \; j = 1,2,\cdots, J \qquad (4)$$

or (ii) the user adopts the relative frequencies of the classes in the training data set as in

$$P(C_j) = (\#\text{ training cases falling into } C_j \text{ / total } \#\text{ of training cases}). \qquad (5)$$

After the prior class probabilities are determined, the training data set is used to calculate the case-specific class probabilities $P(X_1, X_2, \cdots, X_p \mid C_j)$ using the independence assumption (2) and the input-specific class probabilities (3) obtained from the training data set. Then when scoring the training data set or independent data sets, the Bayes formula (1) is applied.

Of course, as with other classifiers, the Naïve Bayes classifier has a tuning parameter, namely, the cut-off probability for choice. In the case that the user is comfortable with $P_{11} = P_{00} = 0$ and the assumption of symmetric payoffs, $P_{10} = P_{01} = L$, then the choice of a cut-off probability of 0.5 is natural and focusing on maximizing the accuracy rate (ACC) or, equivalently, minimizing the error rate (ERR) of the classifier in the validation data set is the approach to follow. However, if the payoffs are

asymmetric in the sense that $R = \left(\frac{P_{10}}{P_{01}}\right) \neq 1$ then one sets out to "shop over" various cut-off probabilities, for example 0.4, 0.45, 0.5, 0.55, and 0.6, so as to choose the cut-off probability that minimizes the weighted error rate, $ERR(weighted) = \frac{(Rn_{10} + n_{01})}{N}$, when applied to the validation data set.