Naïve Bayes Classifier

Q1. Why is *naïve Bayes classification* called naïve? Briefly outline the major ideas of naïve Bayes classification.

[Answer]
Naïve Bayesian classification is called naïve because it assumes class conditional independence. That is, the effect of an attribute value on a given class is independent of the values of the other attributes. This assumption is made to reduce computational costs, and hence is considered naïve. The major idea behind naïve Bayesian classification is to try and classify data by maximising $P(X_j|C_i)P(C_i)$ (where i is an index of the class) using the Bayes theorem of posterior probability. In general,

- We are given a set of unknown data tuples, where each tuple is represented by an n-dimensional vector, $X = (x_1, x_2,\ldots, x_n)$ depicting n measurements made on the tuple from n attributes, respectively, $A_1, A_2, \ldots, A_n$. We are also given a set of m classes, $C_1, C_2, \ldots, C_m$.
- Using Bayes theorem, the naïve Bayesian classifier calculates the posterior probability of each class conditioned on X. X is assigned the class label of the class with the maximum posterior probability conditioned on X. Therefore, we try to maximize $P(C_{ij}|X) = P(X_j|C_i)P(Ci)=P(X)$. However, since $P(X)$ is constant for all classes, only $P(X_j|C_i)P(C_i)$ need be maximized. If the class prior probabilities are not known, then it is commonly assumed that the classes are equally likely, i.e. $P(C_1) = P(C_2) =\ldots = P(C_m)$, and we would therefore maximize $P(X_j|C_i)$. Otherwise, we maximize $P(X_j|C_i)P(C_i)$. The class prior probabilities may be estimated by $P(Ci) = s_i/s$, where s_i is the number of training tuples of class C_i, and s is the total number of training tuples.
- In order to reduce computation in evaluating $P(Xj|Ci)$, the naïve assumption of class conditional independence is made. This presumes that the values of the attributes are conditionally independent of one another, given the class label of the tuple, i.e., that there are no dependence relationships among the attributes.
    - If A_k is a categorical attribute then $P(A_k=x_{kj}|C_i)$ is equal to the number of training tuples in C_i that have x_k as the value for that attribute, divided by the total number of training tuples in C_i.
    - If A_k is a continuous attribute then $P(A_k=x_{kj}|C_i)$ can be calculated using a Gaussian density function.

Q2. The following table consists of training data from an employee database. The data have been generalised. For example, "31…35" and "46K…50K" stands for the age range of 31 to 35 years old and the salary range of 46K to 50K pounds.

department	*status*	*age*	*salary*
sales	senior	31...35	46K...50K
sales	junior	26...30	26K...30K
sales	junior	31...35	31K...35K
systems	junior	21...25	46K...50K
systems	senior	31...35	66K...70K
systems	junior	26...30	46K...50K
systems	senior	41...45	66K...70K
marketing	senior	36...40	46K...50K
marketing	junior	31...35	41K...45K
secretary	senior	46...50	36K...40K
secretary	junior	26...30	26K...30K

You are asked to use the data in the above table to train a naïve Bayesian classifier as the *status* attribute is the class label and all the remaining attributes are regarded as input. Once you have your naïve Bayesian classifier, test the following unseen instances: a) (*marketing, 31…35, 46K…50K*), and b) (*sale, 31…35, 66K-70K*).

[Answer]

We first estimate prior probabilities for the "status" class labels.

P(senior) = 5/11 and *P(junior) = 6/11*.

Now we need to estimate conditional probability for each attribute and store them in tables.

P(department | status)

Class	sales	systems	marketing	secretary
senior	1/5	2/5	1/5	1/5
junior	2/6=1/3	2/6=1/3	1/6	1/6

P(age | status)

Class	21…25	26…30	31…35	36…40	41…45	46…50
senior	0/5=0	0/5=0	2/5	1/5	1/5	1/5
junior	1/6	3/6=1/2	2/6=1/3	0/6=0	0/6=0	0/6=0

P(salary | status)

Class	26K-30K	31K-35K	36K-40K	41K-45K	46K-50K	66K-70K
senior	0/5=0	0/5=0	1/5	0/5=0	2/5	2/5
junior	2/6=1/3	1/6=0	0/6=0	1/6	2/6=1/3	0/6

So for the test instance **a**) (marketing, 31…35, 46K…50K),

$P(senior \mid \boldsymbol{a}) = P(senior)P(marketing \mid senior)P(31…35 \mid senior)P(46K…50K \mid senior)$

$= 5/11*1/5*\ 2/5*2/5$

$P(junior \mid \boldsymbol{a}) = P(junior)\ P(marketing \mid junior)P(31…35 \mid junior)P(46K…50K \mid junior)$

$= 6/11*1/6*\ 1/3*1/3$

The label for this instance (person) is "junior" since $P(junior \mid \boldsymbol{a}) > P(senior \mid \boldsymbol{a})$.

For the test instance **b**) (sale, 31…35, 66K-70K),

$P(senior \mid \boldsymbol{b}) = P(senior)P(sale \mid senior)P(31…35 \mid senior)P(66K…70K \mid senior)$

$= 5/11*1/5*\ 2/5*2/5$

$P(junior \mid \boldsymbol{a}) = P(junior)\ P(sale \mid junior)P(31…35 \mid junior)P(66K…70K \mid junior)$

$= 6/11*1/3*\ 1/3*0/3$

The label for this instance (person) is "senior" since $P(junior \mid \boldsymbol{a}) < P(senior \mid \boldsymbol{a})$.

Note: Strictly speaking, there are some missing attribute values in the training set and therefore the m-estimate method should be used in the conditional probability estimate. Nevertheless, the same results should be achieved if you use the m-estimate method.