Let status be the class label attribute.

(a) How would you modify the basic decision tree algorithm to take into consideration the count of each generalized data tuple (i.e., of each row entry)?

(b) Use your algorithm to construct a decision tree from the given data.

(c) Given a data tuple having the values “systems”, “26...30”, and “46-50K” for the attributes department, age, and salary, respectively, what would a naive Bayesian classification of the status for the tuple be?

(d) Design a multilayer feed-forward neural network for the given data. Label the nodes in the input and output layers.

(e) Using the multilayer feed-forward neural network obtained above, show the weight values after one iteration of the backpropagation algorithm, given the training instance “(sales, senior, 31...35, 46K...50K)”. Indicate your initial weight values and biases, and the learning rate used.

Answer:

(a) How would you modify the basic decision tree algorithm to take into consideration the count of each generalized data tuple (i.e., of each row entry)?

The basic decision tree algorithm should be modified as follows to take into consideration the count of each generalized data tuple:

- The count of each tuple must be integrated into the calculation of the attribute selection measure (such as information gain).
- Take the count into consideration to determine the most common class among the tuples.

(b) Use your algorithm to construct a decision tree from the given data.

The resulting tree is:

\[
\begin{align*}
\text{department} & \quad \text{status} & \quad \text{age} & \quad \text{salary} & \quad \text{count} \\
\text{sales} & \quad \text{senior} & \quad 31...35 & \quad 46K...50K & \quad 30 \\
\text{sales} & \quad \text{junior} & \quad 26...30 & \quad 26K...30K & \quad 40 \\
\text{sales} & \quad \text{junior} & \quad 31...35 & \quad 31K...35K & \quad 40 \\
\text{systems} & \quad \text{junior} & \quad 21...25 & \quad 46K...50K & \quad 20 \\
\text{systems} & \quad \text{senior} & \quad 31...35 & \quad 66K...70K & \quad 5 \\
\text{systems} & \quad \text{junior} & \quad 26...30 & \quad 46K...50K & \quad 3 \\
\text{systems} & \quad \text{senior} & \quad 41...45 & \quad 66K...70K & \quad 3 \\
\text{marketing} & \quad \text{senior} & \quad 36...40 & \quad 46K...50K & \quad 10 \\
\text{marketing} & \quad \text{junior} & \quad 31...35 & \quad 41K...45K & \quad 4 \\
\text{secretary} & \quad \text{senior} & \quad 46...50 & \quad 36K...40K & \quad 4 \\
\text{secretary} & \quad \text{junior} & \quad 26...30 & \quad 26K...30K & \quad 6
\end{align*}
\]
(c) Given a data tuple with the values "systems", "junior", and "26...30" for the attributes department, status, and age, respectively, what would a naïve Bayesian classification of the salary for the tuple be? $P(X|senior) = 0; P(X|junior) = 0.018$. Thus, a naïve Bayesian classification predicts "junior".

(d) Design a multilayer feed-forward neural network for the given data. Label the nodes in the input and output layers.

No standard answer. Every feasible solution is correct. Discrete-valued attributes may be encoded such that there is one input unit per domain value. For hidden layer units, the number should be smaller than that of input units, but larger than that of output units.

(e) Using the multilayer feed-forward neural network obtained above, show the weight values after one iteration of the backpropagation algorithm, given the training instance "(sales, senior, 31...35, 46K...50K)". Indicate your initial weight values and biases and the learning rate used.

No standard answer. Every feasible solution is correct.

6.12. The support vector machine (SVM) is a highly accurate classification method. However, SVM classifiers suffer from slow processing when training with a large set of data tuples. Discuss how to overcome this difficulty and develop a scalable SVM algorithm for efficient SVM classification in large datasets.

**Answer:**

We can use the micro-clustering technique in “Classifying large data sets using SVM with hierarchical clusters” by Yu, Yang, and Han, in Proc. 2003 ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining (KDD’03), pages 306-315, Aug. 2003 [YYH03]. A Cluster-Based SVM (CB-SVM) method is described as follows:

1. Construct the microclusters using a CF-Tree (Chapter 7).
2. Train an SVM on the centroids of the microclusters.
3. Decluster entries near the boundary.
4. Repeat the SVM training with the additional entries.
5. Repeat the above until convergence.

6.13. Write an algorithm for k-nearest neighbor classification given $k$ and $n$, the number of attributes describing each tuple.

**Answer:**

Figure 6.1 presents an algorithm for k-nearest neighbor classification.

6.14. The following table shows the midterm and final exam grades obtained for students in a database course.