Question 6: Classification statistics are shown in the table below.

Dataset	Classification Rate	Tree Size
Dummy 1	1.0	3
Dummy 2	0.65	11
Cars	0.94	408
Connect 4	0.76	41521

Discussion: The first dummy dataset likely had such a high classification rate because it appears that there was one feature (feature five) that uniquely determined the output labels for each of the testing and training examples. From examining the data, it seemed that having a value of 0 for feature five resulted in a 1 for the output label and vice versa in both the training and testing sets. This explains why the tree only needed one attribute to classify the testing examples with 100 percent accuracy. The second dummy dataset had a lower classification rate and a larger tree size, which may indicate that there was overfitting to the training set. For instance, it appeared that attribute two was a fairly good predictor of the final label in the training dataset but not in the testing dataset.

The classification rate for the cars may be high because decision trees are good at dealing with discrete data and, intuitively, because deciding whether or not a car is acceptable based on a set of six attributes is a fairly straightforward task for a human. The tree that was produced included all six features, which could indicate that they were all at least somewhat useful and/or that there was not a lot of noise in the dataset. The tree for the Connect4 dataset was orders of magnitude larger than all of the other trees, which suggests that there may have been some overfitting. Since each point in the dataset corresponded to one specific configuration of the board, it makes sense that the decision tree might overgeneralize from the training examples. It also appears that the input vector is not a particularly accurate representation of the classification problem—for instance, having two configurations with only one different slot could potentially lead down different branches of the tree.

Question 7:

Cars: A dataset similar to the cars dataset could be used as part of a recommendation feature for a web-based music service like Spotify or Pandora. This dataset might include discrete features for each song available on the service, such as its genre, the decade it was released, its instrumentation, et cetera. A decision tree could then be used to a produce a binary recommendation decision for a given user if. However, it would likely be even more effective to use continuous features such as length or number of plays, which don't lend themselves well to decision tree learning.

Connect4: The Connect4 dataset could improve the performance of a bot that determined what moves to take in a game of Connect4 using the minimax algorithm, which determines the optimal move in a two-player game (*http://www-cs-faculty.stanford.edu/people/eroberts/courses/soco/projects/2003-04/intelligent-search/minimax.html*). Specifically, the decision tree could be employed to calculate the value of the evaluation function, perhaps by using it to determine the probability that a certain action would result in a win.

Question 8:

Results: Results for the balance dataset (*https://archive.ics.uci.edu/ml/datasets/Balance+Scale*) are shown in the table below. This dataset had four attributes: the weights on the left and right sides of the scale and the distances of those weights from the center. All attributes had discrete integer values ranging from one to five. The labels indicated whether the scale was tilted to the left or right or was perfectly balanced.

Classification Rate	Tree Size
0.66	501

I would have expected better performance from this dataset since it seems like a relatively simple task and because all of the attributes were discrete. However, the large tree size relative to the number of training examples (501 to 562) suggests that overfitting may be the culprit. It is also possible that it would have been better to treat the attributes as continuous, in which case a decision tree classifier would not be the best choice.

Application: A dataset containing information about the balance of pieces in Jenga could be used in a setup similar to the Connect4 application described above. Specifically, the output of the decision tree could be aid in calculating the output of an evaluation function for the minimax algorithm. A similar dataset with numerical values of 1 through 5 for a student's SAT scores could be used to determine whether a student was prepared for college; this could act as a variable in a Bayes Net like the one described in class that involved variables such as GPA, hours spent studying, et cetera.