Simon Basescu, Charlie Ferguson

Names of files required to run program: DecisionTree.java, Example.java, TestClassifier.java, TestClassifierHepatitis.java, TreeNode.java, (and all provided data files).

Group honor code: “All group members were present and contributing during all work on this project”

We have neither given nor received unauthorized aid on this assignment.

**1) Analysis of performance on big sets**

**Big set 1 performance:**

Positive examples correct: 100 out of 100

Negative examples correct: 100 out of 100

The decision tree was very simple, and only had to split on features 3, 4, and 7 because

either 3 is true, 4 is true, or 7 is true as stated in the problem. Since the positive examples always have this property, but the negative examples never do, it always cleanly splits the data so that if the feature is true, that node becomes a leaf. The decision tree may not always be the same due to different implementation methods which may choose the next feature to split on differently. For example, if there is a tie in terms of information gain between two features to split on, the tie could be broken differently based on the implementation.

**Big set 2 performance:**

Positive examples correct: 98 out of 100

Negative examples correct: 95 out of 100

This decision tree was much larger than the decision tree for Big set 1. The reason for this is that at each feature, 7, 3, and 4, there will not be a clean split like there was in the previous example due to the noise. Despite this, there will still be a large amount of information gain at each split for features 7, 3, and 4 because all of the positive examples will still be split, but along with 10% of the negative examples. Thus the tree must split on more features to parse out the decisions.

**2) Hepatitis performance**

It first looks at the feature *varicela.* If this is not a symptom, it looks at the feature *histology*. If this is not a symptom, then the example is labeled as positive. If *histology* is a symptom, then the example is labeled as negative. From this side of the decision tree, it appears that *varicela* and *histology* are big factors in determining if someone will perish (the absence of both of these symptoms is a good sign for a patient’s survival).

Positive examples correct: 38 out of 42

Negative examples correct: 7 out of 10

Percentage of correctly classified examples: 45/52 =~ 86.5%

False positives: 3

False negatives: 4

**3) Custom algorithm performance**

The name of this data set is *Titanic - Machine Learning from Disaster*, and it is part of a Kaggle competition found here: <https://www.kaggle.com/c/titanic/data>. It is a dataset that consists of passenger data for people on the Titanic and whether or not they survived. We split the fields into the following features: male or female, 1st class ticket, 2nd class ticket, 3rd class ticket, age less than 10, paid a fare of less than or equal to 10, paid a fare between 10 and 30, paid a fare more than 30, has a parent or has a child on board, and finally has a spouse or has a sibling on board. These features were used to predict whether or not the passenger survived. Overall, there are 10 features for each example, and we used 100 positive training examples, 100 negative training examples, 100 positive testing examples, and 100 negative testing examples.

**Results**: Overall, the decision tree predicted 167/200 = 83.5% of the examples correctly. It performed similarly for positive and negative examples, with the following breakdown:

Positive examples correct: 78 out of 100

Negative examples correct: 79 out of 100

This means that there were 22 false negatives, and 21 false positives.