Nhat Le  
CS 365  
Lab C: Design Sheet and Initial Design

Design Plan

1. Import and store the dataset

* Use basic Python I/O to read the data file
* Each example is stored as a tuple because they are not meant to be modified and it is easy to lookup the value for specific attribute by indexing the tuple
* The dataset will be stored as a Python list of example tuples because it will be easy for slicing, splitting, and iteration.

1. Build the decision tree

* Use a tree node structure to build the decision tree
* Each node will have the following attribute:

+ example\_set: the set of examples at that node. It will be the whole dataset for the root and the corresponding subsets for the children after splitting

+ attributes: the list of available attributes that can be used to split the example\_set at that node. It will be the all attributes for root node and {parent.attributes – parent.split\_variable} for the children

+ split\_variable: the attribute/variable that will be used to split the example set at that node, will be calculated and chosen from available attributes at that node to yield maximum information gain

+ distinct\_value: the common value for the example set at that node corresponding to the split\_variable of its parent

+ decision: the Boolean value to indicate whether the node is a terminal/leaf node

+ classification: “yes” or “no” if the node is a terminal/leaf node

+ parent, children: for linked tree structure

* A function to expand a node (split the data set) and add non-terminal node to a frontier for recursive splitting
* A function to calculate entropy of a set
* A function to calculate information gain by splitting a set by an attribute

1. Testing

* A function takes in the tree and a test example and puts the example through the right path in the tree to give the prediction.