Assignment 4: Classification

Due: November 23 2016 11:59PM

Two datasets (Golf, Car) can be found on Piazza. In each dataset, each row corresponds to a record. The last column corresponds to the class label, and the remaining columns are the attributes. For each dataset, we provide two versions: One is the original data, and the other (processed version) is obtained by mapping attribute values to integers. README presents the meanings of the attributes in these two datasets.

In this assignment, you are asked to implement Decision Tree algorithm.

Templates for Python and Java are provided. The use of templates is not required. It's fine if you implement the algorithm from scratch.

You cannot directly call a function or package that implements Decision Tree algorithm. You need to implement the algorithm by yourself. If you are not sure about whether it is OK to use a certain function, please post your question on Piazza.

Please take the following steps:

1. Implement Decision Tree algorithm as follows:

DTree(*records*, *attributes*) returns a tree
       If all *records* belong to the same class, return a leaf node with that class.
       Else pick an *attribute* F based on Gini Index and create a node *R* for it
   For each possible value *v* of *F*:
                    Let *Sv* be the subset of records that have value *v* for *F*
                    Add an out-going edge *E* to node *R* labeled with the value *v*.
                    call DTree(*Sv*, *attributes* – {*F*}) and attach the resulting tree as the subtree under edge *E*.
              Return the subtree rooted at *R*.

2. Test your Decision Tree algorithm on Golf dataset. Based on your output, you can either automatically or manually draw the tree. The resulting tree for the Golf dataset should look like:
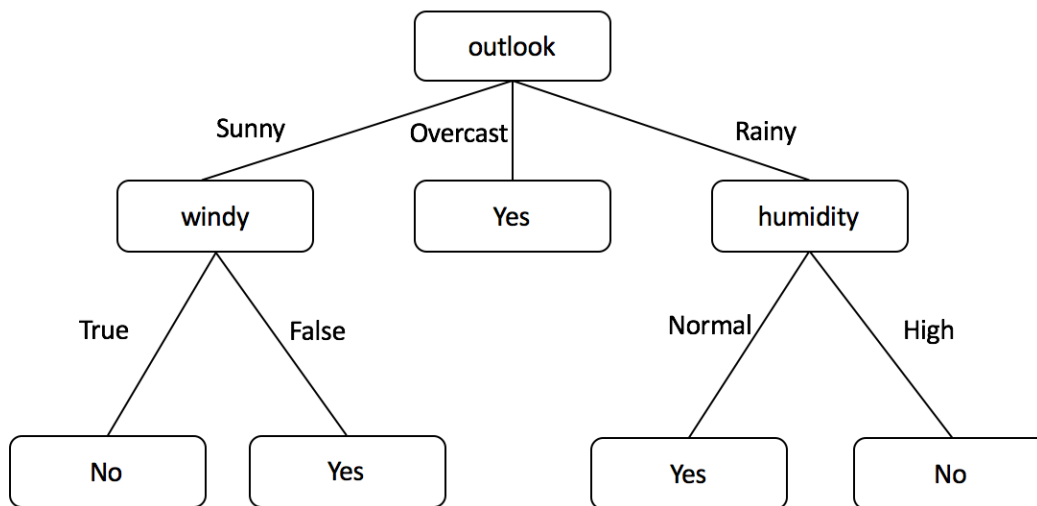
Note that you can choose your own way of representing the tree. For example, the output of your algorithm on the Golf dataset could look like:

```
node 1:  if outlook=Sunny then node 2; elseif outlook=Overcast then node 3; elseif outlook=Rainy then node 4
node 2:  if windy=True then node 5; elseif windy=False then node 6
node 3:  class=Yes
node 4:  if humidity=Normal then node 7; elseif humidity=High then node 8
node 5:  class=No
node 6:  class=Yes
node 7:  class=Yes
node 8:  class=No
```

Also, the tree can be drawn in different ways. It does not matter which output format you choose and which type of tree you draw, as long as we can easily verify your tree and verify the consistency between your algorithm's output and the tree.

To draw the tree, you can either use a tree plotting software, or just draw the tree using Excel or PowerPoint, or draw the tree on a piece of paper and include a scanned copy of the tree in the report.

3. If you get the correct tree, then apply your algorithm on the Car dataset and draw the tree.

4. Prepare your submission. Your final submission should be a zip file named as Assignment4.zip. In the zip file, you should include:
- A folder "Code", which contains all the codes used in this assignment. Inside the folder, please have a file "README" which describes how to run your code. If you use the template, then you don't need to include "README" unless you change the other functions that you are not asked to change.
- Report: A doc or pdf file named as Assignment4.doc or Assignment4.pdf. The report should consist of the following parts: 1) The output of your Decision Tree algorithm on the Car Dataset. Explain the format of the output you use. If the

format is straight-forward to understand, you don't need to explain it. 2) The tree drawn based on the output obtained from the Car Dataset using your algorithm. 3) The code of the Decision Tree algorithm you implement.

5. Log in any CSE department server and submit your zip file as follows:
submit_cse469 Assignment4.zip

Please refer to Course Syllabus for late submission policy. We will take the submission time recorded by the server as the time of your submission.

This assignment must be done independently. Running your submitted code should be able to reproduce the results in the report. Note that copying code/report from another student or source is not allowed and may result in an F. Academic integrity policy can be found at http://www.cse.buffalo.edu/shared/policies/academic.php