Data Science

Assignment #2

|  |  |
| --- | --- |
| Course name | Data science |
| Assignment | Decision tree |
| Major | Computer |
| Student ID | 2011004028 |
| Name | Seonghun Lee |

1. Summary of my algorithm

There are two parts of my algorithm. First, the main part of this program is building a decision tree. And this could be divided into generate function and choosing attribute method. Second, the other part of this program is following to the tree and predict the class name of test file.

1. Building decision tree

There are several rules to build decision tree.

1. Stop if class is same in whole data set
2. Stop if there is no attributes remained
3. Stop if there is no data sets remained

I made two functions to find best attribute in this program. Gain ratio and gini index. However, the performance of those two are same.

1. Following tree function

In test file, there is no class name, so I should predict and write the solution.

1. Detailed description of my codes (for each functions)

|  |
| --- |
| Generate\_decision\_tree |
| * This is main function of this program * It is working as a recursive way to make myTree and the data structure of this is dictionary in python * First 4 lines of this function, if the condition of stop happen, it stop and return appropriate class name. this function is explained below. * And I just made two attribute selection methods (gini index and Gain Ratio) |
| Countclass and majority function |
| * As you can see above, countclass function is calculate the number of class in the data set * Majority function return the major class name |
| Gain ratio function |
| * It is the important function in this program. Attribute selection method! * It is divided into two function. First calculation function called info() and information gain function which is practically working. * In information gain function calculate original info value and find all the info value depending on the partition. And find best attribute which is making higher gain ratio value. |
| Gini index function |
| * Gini index function is similar to information gain function * But, the main difference is calculation formula. |
| Follow decision tree function |
| * It is just follow decision tree and find the class name |
| Main function |
| * Just adjust the form of output file |

1. Instructions for compiling my codes at TA’s computer (e.g. screenshot)

|  |
| --- |
| 1. Put the decisionTree.py and train.txt, test.txt files into same folder |
|  |
| 1. Execute command prompt and go to this folder |
|  |
| 1. Execute decisionTree.py with command line arguments |
|  |
| 1. Check the output file in your folder |
|  |

1. Any other specification of my implementations and testing

|  |
| --- |
| Test result |
| * in the result of this program, the answer accuracy is 5/5 in the first test and 315/346 in the second test. * This result is same using gain ratio and using gini index. |