**Assignment2. Decision Tree**

2017029561 Nam Jihoon

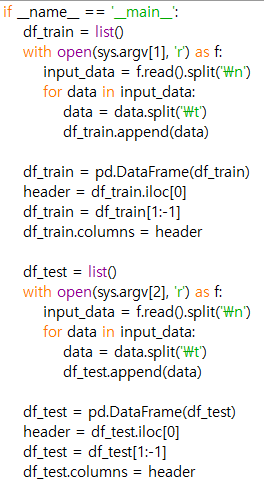
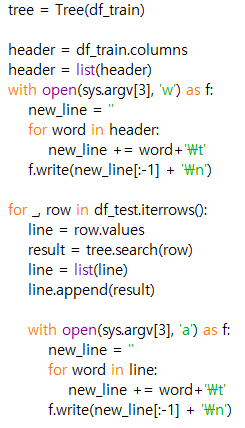
1. **Summary**

The Decision Tree is a model that enables data on various items to be implemented as a tree and used for decision making. Through the node of the tree until the leaf node is reached, the data can be analyzed for items given in common and branched out according to the results.

The tree is constructed according to the data given in the learning stage, and in the test stage, the tree is searched through recursions to find the correct answer.

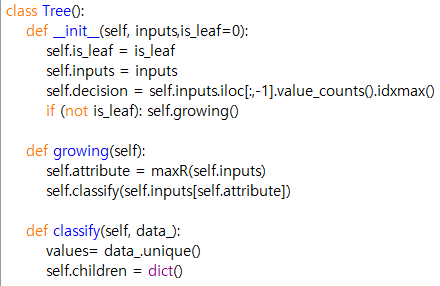
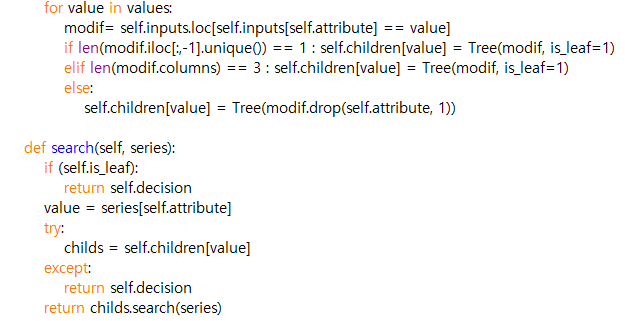
If you simply branch out to form a tree, there is a state of decision that is more than necessary. This is called overfitting. To prevent this, they give up the result of 100% purity and find a way to simplify it by reducing the number of branches as much as possible. This is called pruning.

1. **Functions description**
2. **Main function**

This is the main function, which is automatically called as soon as the program starts. Parameters can be received through the console, which is stored in sys.argv. In this task, it will be used to specify the addresses of input and output files. The main function of the main function is to receive the address of the input file to process the data, then read the input file and save it in the list. Afterwards, it invokes another function to configure and utilize the decision tree, and then stores the results with the specified output file address.

1. **class Tree()**

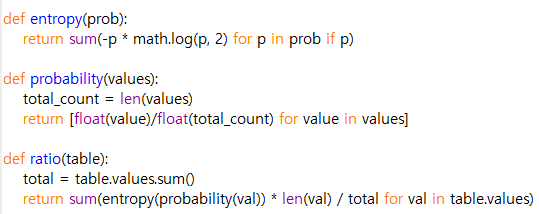
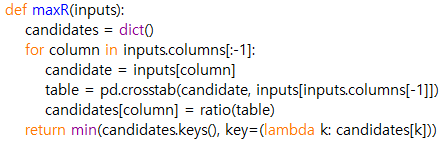
Class was used to construct the tree. The nodes in the tree have their own characteristics according to the type, so they are abstracted and managed through this class.

Each object has a value for data and is\_leaf. Each time you create a new object in the Tree class, the init() function is called. This function initializes the value of the variable that the object has.

Initially, the function used to create a tree through a train dataset is a grouting function. Functions that assist the composition of the tree by calculating the entropy value are defined later. Invoke the auxiliary function and configure the tree through the grouting function.

Use the classify function and search function to deal with the test dataset afterwards. The items of each test data are received and the class label is recursively toured in forecasting models made of training data.

1. **functions to obtain and handle entropy values**

The probability function was used to calculate at once. The value is obtained through the entropy function by utilizing the p obtained through the probability function as a parameter. For this purpose, the sum() function and the log() function of the math function are used inside the function.

1. **Compiling Instruction**

The Program written by Python Idle and used Python 3.7.2.

This program was tested on the Windows 10 and executed using Anaconda 3

This program use several libraries. If these are ready, it also run on command prompt

[1] download dt.py and dt\_answer.txt, dt\_test.txt, dt\_train.txt in same directory

[2] get in to directory that dt.py is downloaded and enter the command “python dt.py dt\_train.txt dt\_test.txt dt\_result.txt” or “python dt.py dt\_train1.txt dt\_test1.txt dt\_resul1t.txt”

[3] To check the test result, enter the command “dt\_test.exe dt\_answer.txt dt\_result.txt” or “dt\_test1.exe dt\_answer1.txt dt\_result1.txt”