ASSIGNMENT 1

Group Details:

Group - 002

18CS10069 - Siba Smarak Panigrahi

18CS30051 - Debajyoti Dasgupta

October 3, 2020

A. Procedure:

1. We have used the standard ID3 algorithm for designing the Decision Tree. The data used for training was provided in .csv format. The data is the percentage increase in COVID cases (worldwide). We have shuffled the data. We considered the split of 60:20:20 as Train, Validation, and Test set respectively. The data had the following attributes:
    a. Date
    b. Confirmed
    c. Recovered
    d. Deaths
    e. Increase rate (Used as Target variable)

2. Major functions:

(This contains the procedure followed. For a detailed function signature, refer the README.md file available with this)

a. **construct_tree:** Uses the helper functions to build the Decision Tree. It is a recursive method, which chooses an attribute at each height to classify the current node into two children nodes. The attribute to be chosen is decided on the basis of the decrease in variance from the current node. The following equations help to determine the attribute to be used (LS denote the learning sample, A denotes an attribute):

$$I(LS) = Var_{y|LS}(y) = E_{y|LS}(y - E_{y|LS}(y))^2$$

$$\triangle I(LS, A) = Var_{y|LS}(y) - \sum_a \frac{|LS_a|}{|LS|} Var_{y|LS_a}(y)$$

b. **good_attr:** This function uses the above-mentioned equations and returns the attribute to be used at a given height to **construct_tree** method. It uses the current data at a given node.
c. **prune:** Prunes the Decision Tree built with the height provided to construct_tree. We have used the validation set to prune the tree.
d. **predict:** Provided a decision tree root and data, this function predicts the output on the basis of training. It outputs the prediction values and MSE value.
e. **randomize_select_best_tree:** Provides the accuracy by averaging over 10 random 80/20 splits. The depth to be considered is provided as a parameter in **construct_tree**. Also returns that particular tree which provides the best test accuracy as the desired one.
f. **print_decision_tree:** Uses Digraph (from Graphviz package) to print the tree. Stores the output in ".gv" and ".pdf" format.

3. Helper functions:

a. **get_date:** To covert date into a float. Further also helps in better decision making and also allows us to consider the date as an attribute.
b. **build_data:** Converts the DataFrame object obtained from read_csv to a collection of dictionaries. Each row is converted to a dictionary, with keys as attributes, and values as the corresponding entry in that row.
c. **train_test_split:** Splits the data into three parts i.e, Train, Validation, and Test set in the ratio of 60:20:20.
d. **remove_children:** Removes the children of the current node. Used in the **prune** method.
e. **restore:** Restores the children of the current node. Used in the **prune** method.
f. **Count_node**: Provided the root of a Decision Tree, it counts and returns the number of nodes.
g. **variance:** Calculates the variance of the provided data. Used by **good_attr** method.
h. **predict_one:** Outputs the predicted value for a single data. It uses the attribute and threshold (split) value at a node and recursively moves from the root to a leaf. The predicted value is the mean value of that leaf node. Used by **predict** method.

i. **get_node:** Returns the text to be printed at a particular node. Used by **print_decision_tree** method.
j. **is_leaf:** Returns true if the given node is a leaf node. Used to terminate the recursion in the **predict_one** and **get_node** method.
k. **r2_score:** Used to find accuracy.

4. Pruning

a. **Steps:**
    i. We constructed the entire Decision Tree with the height provided in **construct_tree**. We take a bottom-up approach.
    ii. Pruning begins from the parent of leaf nodes. We consider a node. We temporarily remove the children of the node, say it **New_Temporary_Tree**. Evaluate the mean-squared error (**MSE**) of the New_Temporary_Tree on the validation set.
    iii. If the MSE decreases on the validation set, then we remove the children of that node and update the attributes of the current node, thus updating the tree.
    iv. We repeat these steps until no more pruning is possible (overfitting has reduced significantly).
b. The above pruning approach is also termed as **reduced-error pruning**. To summarize, we have used the train set for building the tree, validation set for pruning, and test set for estimating the final accuracy.

B. Results

1. Best Test Accuracy with a given height:

a. The accuracy by averaging over 10 random 80/20 splits: **25.95%** (with height = -1 i.e. Full Tree)
b. The tree with the best test accuracy (the desired one):
Train Acc: 100.0, Train MSE: 0.0, Test Acc: 87.61, Test MSE: 5.12

2. Height Variation and Selection of Best Possible Depth

We have selected the depth of 40 as the best possible depth. This depth provides the least MSE value on the Test set. The detailed tabular result and plot of Test-MSE and height are presented below:
(the actual code considers all the height from 1 to 49. To make the following table look a little less clumsy, we provide the results at depths which are multiples of 5)

BEST TREE: Height = 40
Train acc: 100%, Train MSE: 0.0, Test acc: 93.68%, Test MSE: 2.61

Table 1: Accuracy and MSE Results at different heights

Height	Train Acc. (in %)	Train MSE	Test Acc. (in %)	Test MSE
1	-0.09	77.55	-3.82	42.93
5	75.76	18.78	79.34	8.54
10	99.1	0.7	88.94	4.57
15	99.75	0.19	88.05	4.94
20	96.76	2.51	87.38	5.22
25	92.16	6.08	86.04	5.77
30	89.04	8.49	83.62	6.77
35	81.53	14.31	92.74	3.0
40	**85.27**	**11.41**	**93.68**	**2.61**
45	93.42	5.1	88.26	4.86

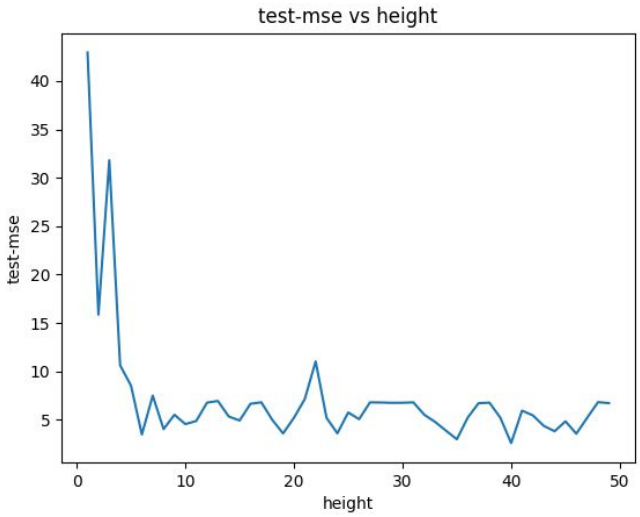

Fig 1: Test MSE vs Heights

3. Results of Pruning:

a. **Before pruning**: Valid acc: 35.77%, Valid MSE: 45.66, number of nodes = 203

b. **After Pruning**: Valid acc: 36.41%, Valid MSE: 45.20, number of nodes = 81

c. **Pruned Tree**:

Train acc: 89.38%, Train MSE: 8.42, Test acc: 97.76%, Test MSE: 0.92

4. Final Pruned decision tree with the hierarchical representation of Attributes:

It is available in the submitted zip folder.

The name of the file is: **Decision Tree Image/decision_tree.gv.pdf**