INDIAN INSTITUTE OF TECHNOLOGY KHARAGPUR

# CS60050 – Machine Learning

# Assignment 1
# Decision Trees

Submitted by:

Jothi Prakash (19EC39023)

Tushar Kishore Bokade (19CS30011)

# Theory

A tree has many analogies in real life, and turns out that it has influenced a wide area of machine learning, covering both classification and regression. In decision analysis, a decision tree can be used to visually and explicitly represent decisions and decision making. As the name goes, it uses a tree-like model of decisions. Though a commonly used tool in data mining for deriving a strategy to reach a particular goal, its also widely used in machine learning

Important Aspects of a Decision Tree -

**Recursive Binary Splitting**

In this procedure all the features are considered and different split points are tried and tested using a cost function. The split with the best cost (or lowest cost) is selected.

**Cost of a split**

$$\text{Regression : sum}(y — \text{prediction})^2$$

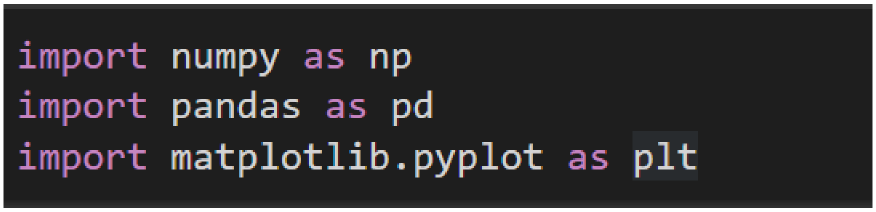
$$\text{Classification : } G = \text{sum}(pk * (1 — pk))$$

**Pruning**

The performance of a tree can be further increased by pruning. It involves removing the branches that make use of features having low importance. This way, we reduce the complexity of tree, and thus increasing its predictive power by reducing overfitting.

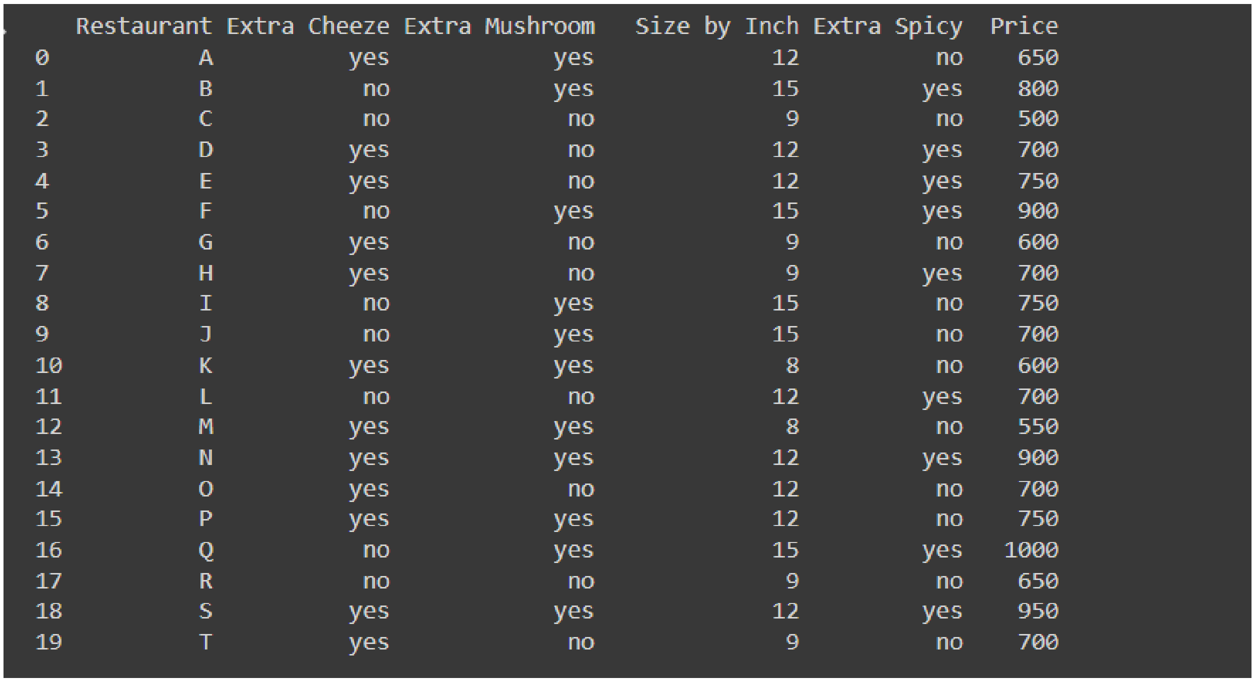
# Experimental Procedure

- Imported 3 libraries – Numpy, Pandas and Matplotlib



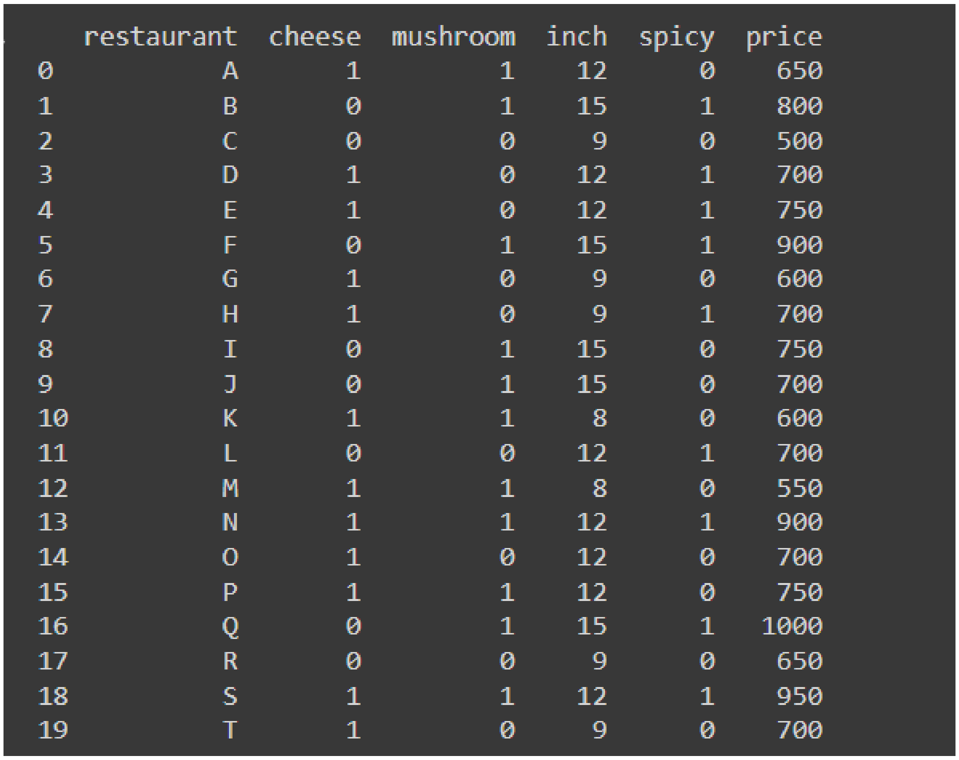
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
```

- Read the data



```
    Restaurant Extra Cheeze Extra Mushroom   Size by Inch Extra Spicy  Price
0            A          yes            yes             12          no    650
1            B           no            yes             15         yes    800
2            C           no             no              9          no    500
3            D          yes             no             12         yes    700
4            E          yes             no             12         yes    750
5            F           no            yes             15         yes    900
6            G          yes             no              9          no    600
7            H          yes             no              9         yes    700
8            I           no            yes             15          no    750
9            J           no            yes             15          no    700
10           K          yes            yes              8          no    600
11           L           no             no             12         yes    700
12           M          yes            yes              8          no    550
13           N          yes            yes             12         yes    900
14           O          yes             no             12          no    700
15           P          yes            yes             12          no    750
16           Q           no            yes             15         yes   1000
17           R           no             no              9          no    650
18           S          yes            yes             12         yes    950
19           T          yes             no              9          no    700
```

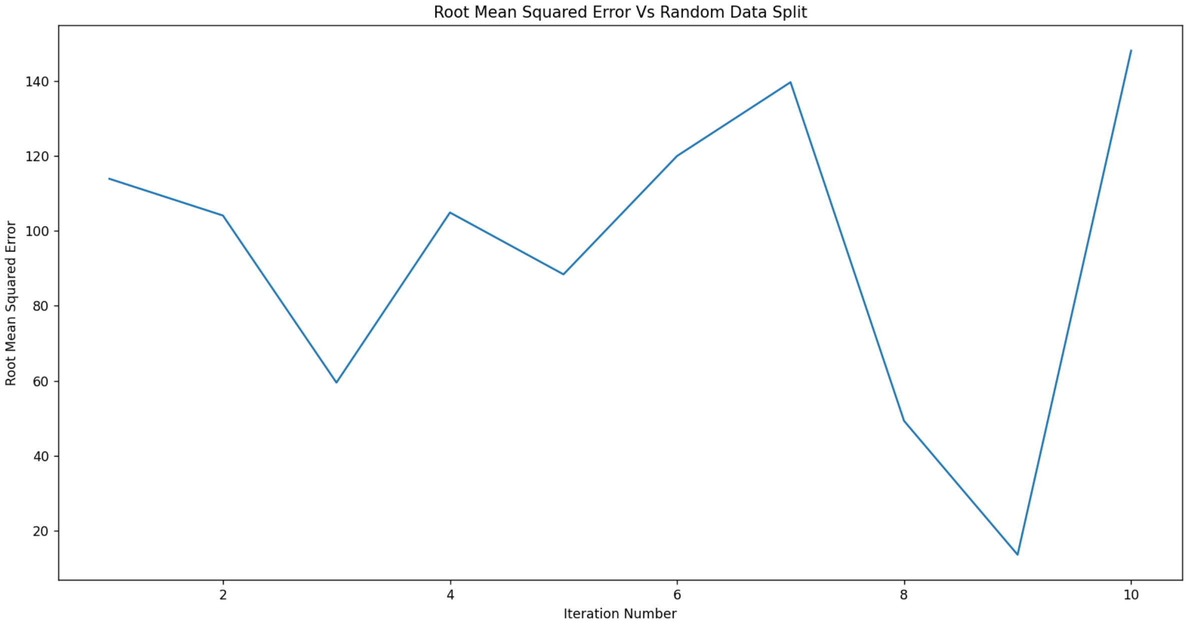
- Renaming the columns and the data for ease of use



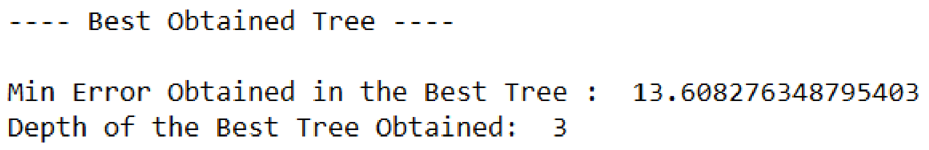
```
    restaurant  cheese  mushroom  inch  spicy  price
0            A       1         1    12      0    650
1            B       0         1    15      1    800
2            C       0         0     9      0    500
3            D       1         0    12      1    700
4            E       1         0    12      1    750
5            F       0         1    15      1    900
6            G       1         0     9      0    600
7            H       1         0     9      1    700
8            I       0         1    15      0    750
9            J       0         1    15      0    700
10           K       1         1     8      0    600
11           L       0         0    12      1    700
12           M       1         1     8      0    550
13           N       1         1    12      1    900
14           O       1         0    12      0    700
15           P       1         1    12      0    750
16           Q       0         1    15      1   1000
17           R       0         0     9      0    650
18           S       1         1    12      1    950
19           T       1         0     9      0    700
```

- Splitting the data as 70% train and 30% test over multiple iterations

## Root Mean Squared Error Over Multiple Iterations

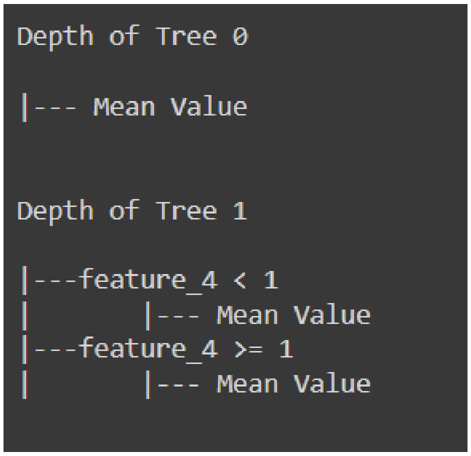


- Best Tree obtained is

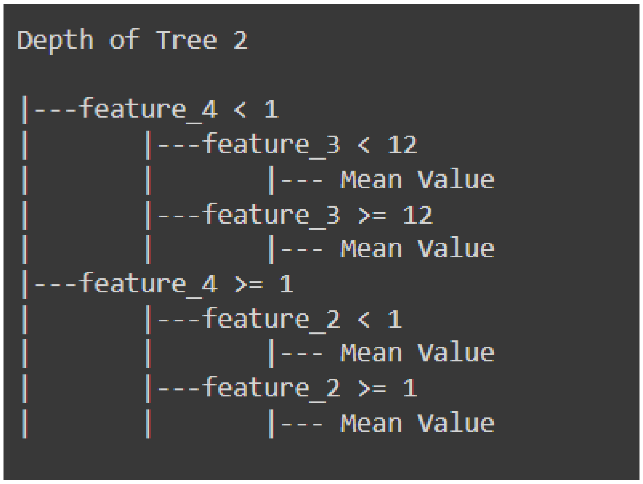


```
---- Best Obtained Tree ----

Min Error Obtained in the Best Tree :  13.608276348795403
Depth of the Best Tree Obtained:  3
```

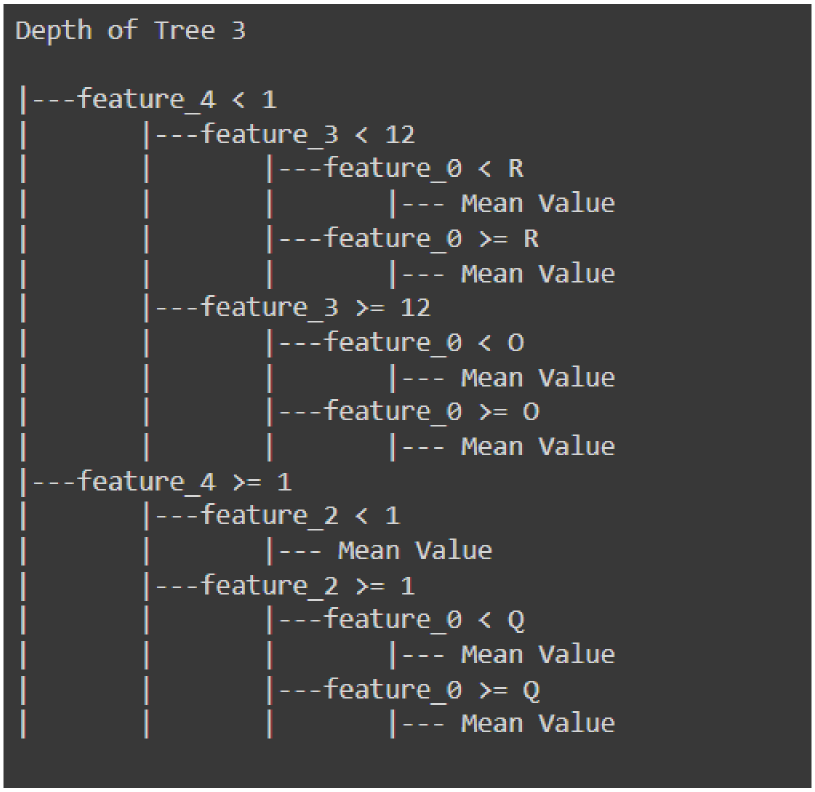
- Performing Rule based pruning while changing the depth of the decision tree



```
Depth of Tree 0

|--- Mean Value


Depth of Tree 1

|---feature_4 < 1
|       |--- Mean Value
|---feature_4 >= 1
|       |--- Mean Value
```

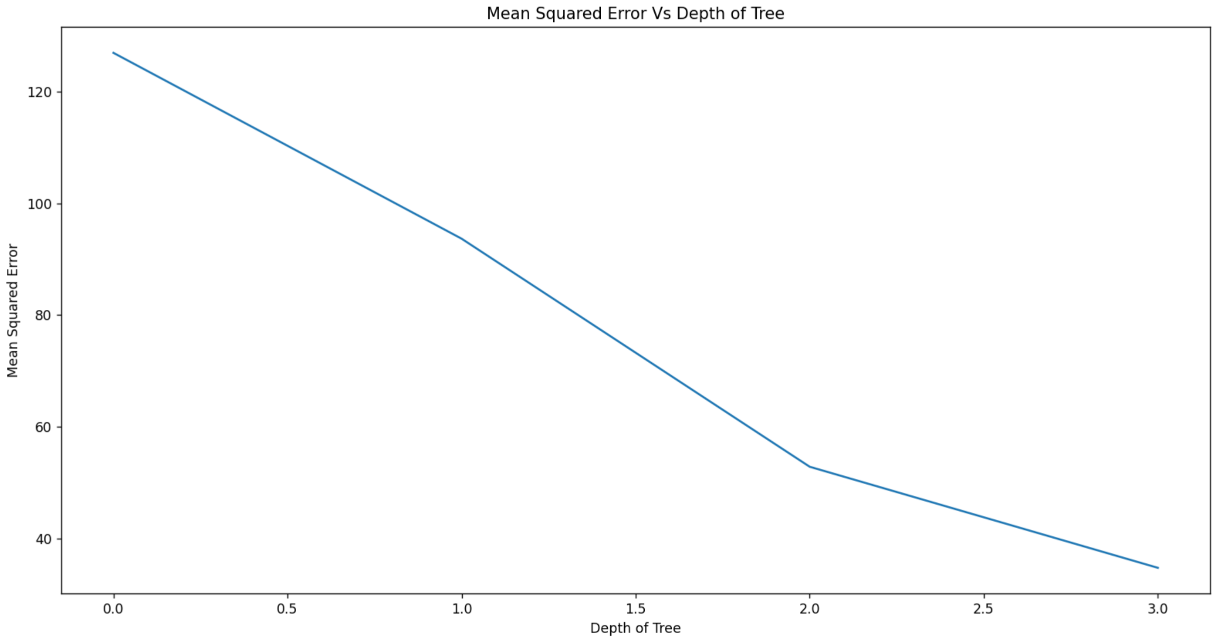


```
Depth of Tree 2

|---feature_4 < 1
|       |---feature_3 < 12
|       |       |--- Mean Value
|       |---feature_3 >= 12
|       |       |--- Mean Value
|---feature_4 >= 1
|       |---feature_2 < 1
|       |       |--- Mean Value
|       |---feature_2 >= 1
|       |       |--- Mean Value
```



```
Depth of Tree 3

|---feature_4 < 1
|       |---feature_3 < 12
|       |       |---feature_0 < R
|       |       |       |--- Mean Value
|       |       |---feature_0 >= R
|       |       |       |--- Mean Value
|       |---feature_3 >= 12
|       |       |---feature_0 < O
|       |       |       |--- Mean Value
|       |       |---feature_0 >= O
|       |       |       |--- Mean Value
|---feature_4 >= 1
|       |---feature_2 < 1
|       |       |--- Mean Value
|       |---feature_2 >= 1
|       |       |---feature_0 < Q
|       |       |       |--- Mean Value
|       |       |---feature_0 >= Q
|       |       |       |--- Mean Value
```

## Variation of Error with Depth of Tree



# Results

- The obtained Root Mean Squared Error fluctuates between 10 – 140 over different train and test split of the data
- The best tree gave a Root Mean Squared Error of 13.60 and had a depth of 3
- The best tree gave the best Mean Squared Error of 30.30 on the dataset
- With increasing depth of the tree it always performs better, this maybe due to the very limited availability of data of just 21 entries