

```
In [ ]: # #https://www.w3schools.com/python/python_ml_decision_tree.asp
# Machine Learning Lab2: Created by Jibrael Jos, PhD
# Topic: Decision Tree Explorations
# Student Name:
# Roll No:
# Date:
# Submission :
# Python Notebook as PDF (File Name MLDecisionTree21.pdf)
# and Observations in MLLab2_21.xlsx

# Where 21 can be replaced with your roll number
```

```
In [1]: import pandas

df = pandas.read_csv("dataTree1.csv")

print(df)
```

	Age	Experience	Rank	Nationality	Go
0	36	10	9	UK	NO
1	42	12	4	USA	NO
2	23	4	6	N	NO
3	52	4	4	USA	NO
4	43	21	8	USA	YES
5	44	14	5	UK	NO
6	66	3	7	N	YES
7	35	14	9	UK	YES
8	52	13	7	N	YES
9	35	5	9	N	YES
10	24	3	5	USA	NO
11	18	3	7	UK	YES
12	45	9	9	UK	YES

```
In [ ]: # 1.Upload data from a csv file
# 2.Upload from a text file where seperator is tab
# 3.Upload from an Excel Sheet
```

```
In [2]: d = {'UK': 0, 'USA': 1, 'N': 2}
df['Nationality'] = df['Nationality'].map(d)
d = {'YES': 1, 'NO': 0}
df['Go'] = df['Go'].map(d)
```

```
print(df)
```

	Age	Experience	Rank	Nationality	Go
0	36	10	9	0	0
1	42	12	4	1	0
2	23	4	6	2	0
3	52	4	4	1	0
4	43	21	8	1	1
5	44	14	5	0	0
6	66	3	7	2	1
7	35	14	9	0	1
8	52	13	7	2	1
9	35	5	9	2	1
10	24	3	5	1	0
11	18	3	7	0	1
12	45	9	9	0	1

```
In [ ]: # 4. Explore map function in a data frame
        # 5. Create a map function to convert a month column to Numbers
        # Jan-1, Feb-2 and so on
        # 6. Create a map function to convert True to 1 and False to Zero
```

```
In [3]: features = ['Age', 'Experience', 'Rank', 'Nationality']

X = df[features]
y = df['Go']

print(X)
print(y)
```

	Age	Experience	Rank	Nationality
0	36	10	9	0
1	42	12	4	1
2	23	4	6	2
3	52	4	4	1
4	43	21	8	1
5	44	14	5	0
6	66	3	7	2
7	35	14	9	0
8	52	13	7	2
9	35	5	9	2
10	24	3	5	1
11	18	3	7	0
12	45	9	9	0

0	0
1	0
2	0
3	0
4	1
5	0
6	1
7	1
8	1
9	1
10	0
11	1
12	1

Name: Go, dtype: int64

```
In [4]: import pandas
from sklearn import tree
from sklearn.tree import DecisionTreeClassifier
import matplotlib.pyplot as plt

df = pandas.read_csv("dataTree1.csv")

d = {'UK': 0, 'USA': 1, 'N': 2}
df['Nationality'] = df['Nationality'].map(d)
d = {'YES': 1, 'NO': 0}
df['Go'] = df['Go'].map(d)

features = ['Age', 'Experience', 'Rank', 'Nationality']
print(features)
X = df[features]
```

```

y = df['Go']

dtree = DecisionTreeClassifier(criterion='gini')
dtree = dtree.fit(X, y)

tree.plot_tree(dtree, feature_names=features)

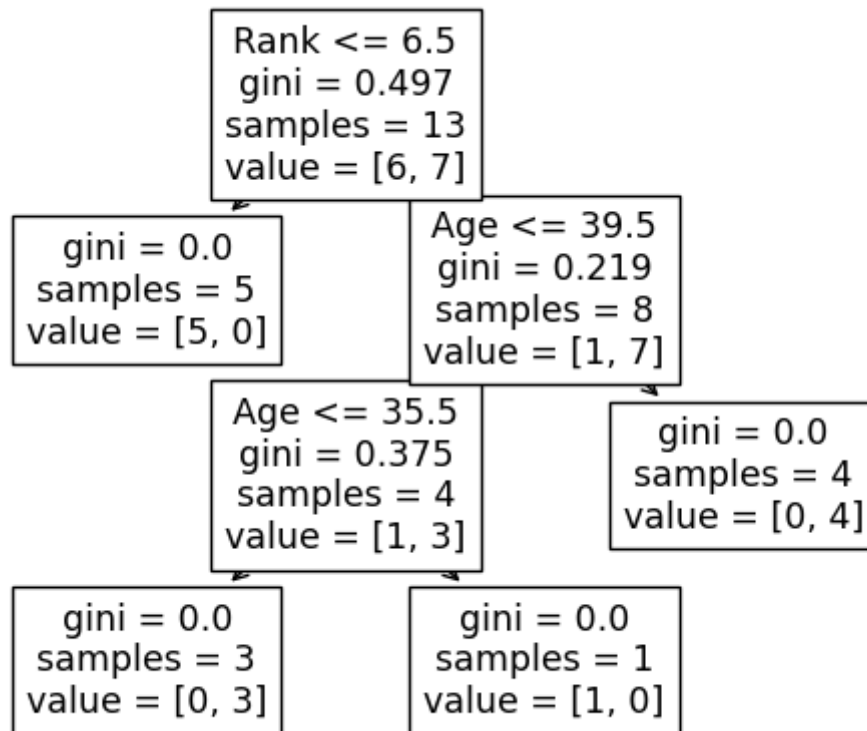
```

Out[4]:

```

['Age', 'Experience', 'Rank', 'Nationality']
[Text(0.4, 0.875, 'Rank <= 6.5\ngini = 0.497\nsamples = 13\nvalue = [6, 7]'),
 Text(0.2, 0.625, 'gini = 0.0\nsamples = 5\nvalue = [5, 0]'),
 Text(0.6, 0.625, 'Age <= 39.5\ngini = 0.219\nsamples = 8\nvalue = [1, 7]'),
 Text(0.4, 0.375, 'Age <= 35.5\ngini = 0.375\nsamples = 4\nvalue = [1, 3]'),
 Text(0.2, 0.125, 'gini = 0.0\nsamples = 3\nvalue = [0, 3]'),
 Text(0.6, 0.125, 'gini = 0.0\nsamples = 1\nvalue = [1, 0]'),
 Text(0.8, 0.375, 'gini = 0.0\nsamples = 4\nvalue = [0, 4]')]

```



In []: # 7. Run Code above
8. Check GINI value with Maths Calculation in an Excel Workbook

```
# 9. Change Gini to Entropy and check calculation
# 10. Change different parameters and study the impact
```

```
In [5]: print(dtree.predict([[47, 12, 4, 0]]))
```

```
[0]
```

```
C:\ProgramData\anaconda3\Lib\site-packages\sklearn\base.py:464: UserWarning: X does not have valid feature names, but DecisionTreeClassifier was fitted with feature names
  warnings.warn(
```

```
In [6]: print(dtree.predict([[40, 10, 6, 1]]))
```

```
[0]
```

```
C:\ProgramData\anaconda3\Lib\site-packages\sklearn\base.py:464: UserWarning: X does not have valid feature names, but DecisionTreeClassifier was fitted with feature names
  warnings.warn(
```

```
In [7]: #https://www.geeksforgeeks.org/decision-tree/
```

```
In [8]: # Import the necessary libraries
from sklearn.datasets import load_iris
from sklearn import tree
from sklearn.tree import DecisionTreeClassifier
from sklearn.tree import export_graphviz
from graphviz import Source

# Load the dataset
iris = load_iris()
X = iris.data[:, 2:] # petal length and width
y = iris.target
features = ['Length', 'Width']
# DecisionTreeClassifier
tree_clf = DecisionTreeClassifier(criterion='entropy',
                                  max_depth=2)

tree_clf.fit(X, y)
tree.plot_tree(tree_clf, feature_names=features)
# # Plot the decision tree graph
# export_graphviz(
#     tree_clf,
#     out_file="iris_tree.dot",
#     feature_names=iris.feature_names[2:],
#     class_names=iris.target_names,
#     rounded=True,
```

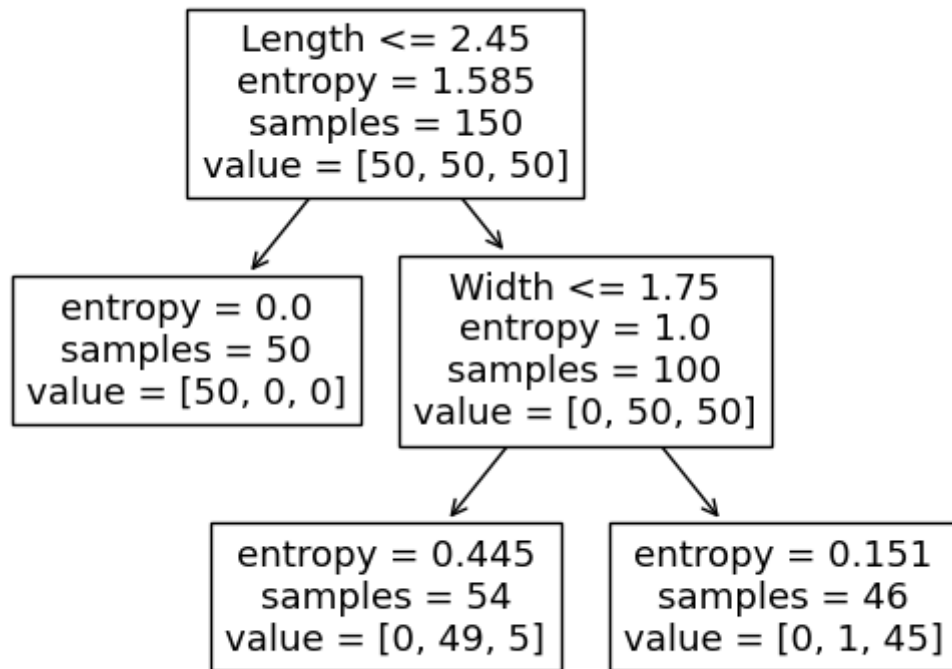
```
#         filled=True
# )

# with open("iris_tree.dot") as f:
#     dot_graph = f.read()

# Source(dot_graph)
```

Out[8]:

```
[Text(0.4, 0.8333333333333334, 'Length <= 2.45\nentropy = 1.585\nsamples = 150\nvalue = [50, 50, 50]'),
Text(0.2, 0.5, 'entropy = 0.0\nsamples = 50\nvalue = [50, 0, 0]'),
Text(0.6, 0.5, 'Width <= 1.75\nentropy = 1.0\nsamples = 100\nvalue = [0, 50, 50]'),
Text(0.4, 0.16666666666666666, 'entropy = 0.445\nsamples = 54\nvalue = [0, 49, 5]'),
Text(0.8, 0.16666666666666666, 'entropy = 0.151\nsamples = 46\nvalue = [0, 1, 45]')]
```



In []:

```
# 11. Check IRIS dataset with Entropy
# 12. Compare method we used in Excel and in Python for IRIS data
# 13. Read about Decision Tree (share URL read).
# Identify advantages and disadvantages
# 14. Watch a video on Decision Tree(share URL)
```

```
# Share some Learnings  
# 15 Decision Tree can be viewed using plottree and graphviz .. Explore both methods
```

In []: