assignment-3

September 24, 2024

```
[2]: import pandas as pd
```

```
[3]: df = pd.read_csv('Admission_Predict.csv')
```

```
[4]: df.head()
```

```
[4]:    Serial No.  GRE Score  TOEFL Score  University Rating  SOP  LOR   CGPA  \
     0           1        337          118                  4  4.5   4.5  9.65
     1           2        324          107                  4  4.0   4.5  8.87
     2           3        316          104                  3  3.0   3.5  8.00
     3           4        322          110                  3  3.5   2.5  8.67
     4           5        314          103                  2  2.0   3.0  8.21

        Research  Chance of Admit
     0         1              0.92
     1         1              0.76
     2         1              0.72
     3         1              0.80
     4         0              0.65
```

```
[5]: from sklearn.preprocessing import Binarizer
     bi = Binarizer(threshold=0.75) # here we are changing values less than 0.75 to␣
      ↪0 and above 0.75 to 1
     df['Chance of Admit '] = bi.fit_transform(df[['Chance of Admit ']])
```

```
[6]: x = df.drop('Chance of Admit ',axis=1) # dropping the admitted column
     y = df['Chance of Admit ']
```

```
[7]: y=y.astype('int')
```

```
[8]: y.value_counts()
```

```
[8]: 0    228
     1    172
     Name: Chance of Admit , dtype: int64
```

```
[11]: from sklearn.model_selection import train_test_split
```

[13]:

```
x_train,x_test,y_train,y_test = train_test_split(x,y,random_state=0,test_size=0.
 ↪25)
```

[14]:

```
from sklearn.tree import DecisionTreeClassifier
```

[15]:

```
classifier = DecisionTreeClassifier(random_state=0)
```

[16]:

```
classifier.fit(x_train,y_train)
```

[16]:

```
DecisionTreeClassifier(random_state=0)
```

[18]:

```
y_pred = classifier.predict(x_test)
```

[19]:

```
result = pd.DataFrame(
{
    'actual':y_test,
    'predicted':y_pred
})
```

[20]:

```
result
```

[20]:

```
     actual  predicted
132       0          0
309       0          0
341       1          1
196       0          0
246       0          1
..      …          …
146       0          0
135       1          1
390       0          0
264       0          0
364       1          1

[100 rows x 2 columns]
```

[21]:

```
from sklearn.metrics import confusion_matrix,accuracy_score
from sklearn.metrics import classification_report
```

[ ]:

```
cm = confusion_matrix(y_test,y_pred)
disp = ConfusionMatrixDisplay(confusion_matrix=cm)
dip.plot()
```

[26]:

```
accuracy_score(y_test,y_pred)
```

[26]:

```
0.9
```

[27]:

```
print(classification_report(y_test,y_pred))
```

```
              precision    recall  f1-score   support

           0       0.92      0.92      0.92        59
           1       0.88      0.88      0.88        41

    accuracy                           0.90       100
   macro avg       0.90      0.90      0.90       100
weighted avg       0.90      0.90      0.90       100
```

[28]:

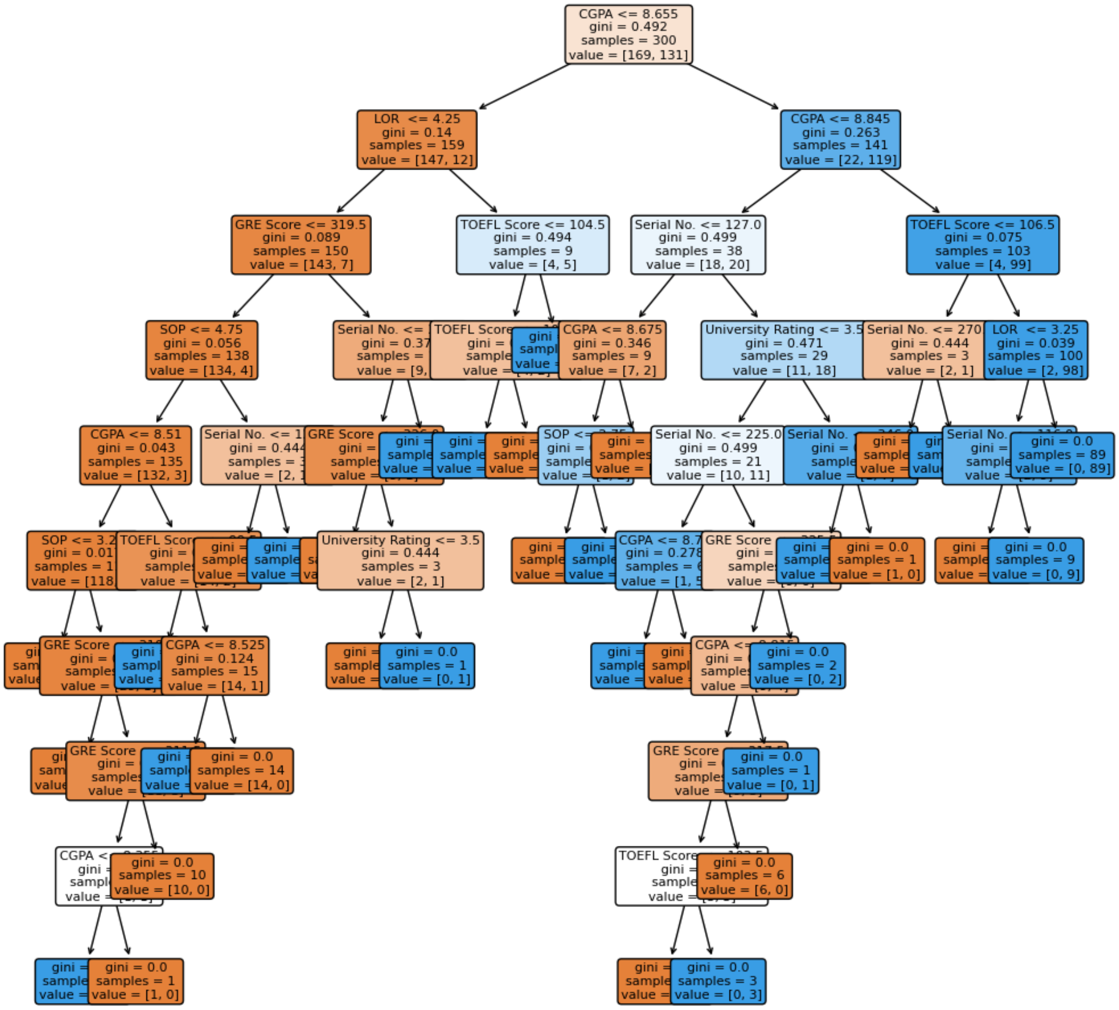
```
from sklearn.tree import plot_tree
```

[29]:

```
import matplotlib.pyplot as plt
```

[32]:

```
plt.figure(figsize=(12,12))
plot_tree(classifier,fontsize=8,filled=True,rounded=True,feature_names=x.
  ↪columns);
```


CGPA <= 8.655
gini = 0.492
samples = 300
value = [169, 131]
LOR <= 4.25
gini = 0.14
samples = 159
value = [147, 12]
CGPA <= 8.845
gini = 0.263
samples = 141
value = [22, 119]
GRE Score <= 319.5
gini = 0.089
samples = 150
value = [143, 7]
TOEFL Score <= 104.5
gini = 0.494
samples = 9
value = [4, 5]
Serial No. <= 127.0
gini = 0.499
samples = 38
value = [18, 20]
TOEFL Score <= 106.5
gini = 0.075
samples = 103
value = [4, 99]
SOP <= 4.75
gini = 0.056
samples = 138
value = [134, 4]
CGPA <= 8.675
gini = 0.346
samples = 9
value = [7, 2]
University Rating <= 3.5
gini = 0.471
samples = 29
value = [11, 18]
Serial No. <= 270
gini = 0.444
samples = 3
value = [2, 1]
LOR <= 3.25
gini = 0.039
samples = 100
value = [2, 98]
CGPA <= 8.51
gini = 0.043
samples = 135
value = [132, 3]
Serial No. <= 225.0
gini = 0.499
samples = 21
value = [10, 11]
gini = 0.0
samples = 89
value = [0, 89]
University Rating <= 3.5
gini = 0.444
samples = 3
value = [2, 1]
gini = 0.0
samples = 1
value = [1, 0]
gini = 0.0
samples = 9
value = [0, 9]
CGPA <= 8.525
gini = 0.124
samples = 15
value = [14, 1]
gini = 0.0
samples = 1
value = [0, 1]
gini = 0.0
samples = 2
value = [0, 2]
gini = 0.0
samples = 14
value = [14, 0]
gini = 0.0
samples = 1
value = [0, 1]
gini = 0.0
samples = 10
value = [10, 0]
gini = 0.0
samples = 6
value = [6, 0]
gini = 0.0
samples = 1
value = [1, 0]
gini = 0.0
samples = 3
value = [0, 3]

[ ]: