p6sldfb2m

August 2, 2023

[48]:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.linear_model import LogisticRegression
from sklearn.preprocessing import StandardScaler
```

[49]:

```
from google.colab import drive
drive.mount('/content/drive')
```

```
Drive already mounted at /content/drive; to attempt to forcibly remount, call
drive.mount("/content/drive", force_remount=True).
```

[50]:

```
df_train=pd.read_csv("/content/drive/MyDrive/mydatasets/C2_train.
  ↪gender_submission.csv")
df_test=pd.read_csv("/content/drive/MyDrive/mydatasets/C2_test.
  ↪gender_submission.csv")
```

[51]:

```
df_train.dropna(inplace=True)
df_test.dropna(inplace=True)
```

[52]:

```
df_train.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 183 entries, 1 to 889
Data columns (total 12 columns):
 #   Column       Non-Null Count  Dtype
---  ------       --------------  -----
 0   PassengerId  183 non-null    int64
 1   Survived     183 non-null    int64
 2   Pclass       183 non-null    int64
 3   Name         183 non-null    object
 4   Sex          183 non-null    object
 5   Age          183 non-null    float64
 6   SibSp        183 non-null    int64
 7   Parch        183 non-null    int64
 8   Ticket       183 non-null    object
 9   Fare         183 non-null    float64
 10  Cabin        183 non-null    object
 11  Embarked     183 non-null    object
dtypes: float64(2), int64(5), object(5)
memory usage: 18.6+ KB
```

[53]:
```
feature_matrix = df_train[['PassengerId', 'Pclass',  'Age', 'SibSp','Fare',
       'Parch']]
target_vector = df_train[['Embarked']]
```

[54]:
```
fs = StandardScaler().fit_transform(feature_matrix)
logr = LogisticRegression()
logr.fit(fs,target_vector)
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/utils/validation.py:1143:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
ravel().
  y = column_or_1d(y, warn=True)
```

[54]:
```
LogisticRegression()
```

[55]:
```
observation = df_test[['PassengerId', 'Pclass',  'Age', 'SibSp','Fare',
       'Parch']]
prediction = logr.predict(observation)
print(prediction)
```

```
['S' 'S' 'C' 'S' 'S' 'S' 'S' 'S' 'S' 'S' 'C' 'S' 'C' 'C' 'S' 'C' 'S' 'S'
 'S' 'S' 'C' 'S' 'S' 'S' 'S' 'S' 'C' 'S' 'S' 'S' 'S' 'C' 'S' 'S' 'S' 'S'
 'S' 'S' 'S' 'S' 'S' 'C' 'S' 'S' 'S' 'S' 'S' 'S' 'S' 'S' 'S' 'S' 'S' 'S'
 'S' 'S' 'S' 'S' 'S' 'S' 'S' 'S' 'S' 'S' 'S' 'S' 'S' 'S' 'C' 'S' 'S' 'S'
 'S' 'S' 'S' 'S' 'S' 'S' 'S' 'S' 'S' 'S' 'S' 'S' 'S' 'S' 'S']
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/base.py:432: UserWarning: X has
feature names, but LogisticRegression was fitted without feature names
  warnings.warn(
```

[56]:
```
logr.classes_
```

[56]:
```
array(['C', 'Q', 'S'], dtype=object)
```

[57]:
```
logr.predict_proba(observation)[0][0]
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/base.py:432: UserWarning: X has
feature names, but LogisticRegression was fitted without feature names
  warnings.warn(
```

[57]:
```
6.0897190477966995e-40
```

Random Forest

[65]:

```
df2=pd.read_csv("/content/drive/MyDrive/mydatasets/C2_test.gender_submission.
 ↪csv")
df2=df.dropna()
df2['Embarked'].value_counts()
```

[65]:

```
1    47
2    39
3     1
Name: Embarked, dtype: int64
```

[66]:

```
x=df2[['PassengerId', 'Pclass','Age', 'SibSp', 'Parch',
        'Fare']]
y=df2['Embarked']
```

[67]:

```
g1={"Embarked":{"C":1,"S":2,"Q":3}}
df2=df2.replace(g1)
df2
```

[67]:

```
     PassengerId  Pclass                                               Name   \
12           904       1      Snyder, Mrs. John Pillsbury (Nelle Stevenson)
14           906       1  Chaffee, Mrs. Herbert Fuller (Carrie Constance…
24           916       1    Ryerson, Mrs. Arthur Larned (Emily Maria Borie)
26           918       1                       Ostby, Miss. Helene Ragnhild
28           920       1                            Brady, Mr. John Bertram
..             …     …                                                 …
404         1296       1                       Frauenthal, Mr. Isaac Gerald
405         1297       2       Nourney, Mr. Alfred (Baron von Drachstedt")"
407         1299       1                         Widener, Mr. George Dunton
411         1303       1    Minahan, Mrs. William Edward (Lillian E Thorpe)
414         1306       1                       Oliva y Ocana, Dona. Fermina

        Sex   Age  SibSp  Parch         Ticket     Fare            Cabin   \
12   female  23.0      1      0          21228   82.2667              B45
14   female  47.0      1      0    W.E.P. 5734   61.1750              E31
24   female  48.0      1      3       PC 17608  262.3750  B57 B59 B63 B66
26   female  22.0      0      1         113509   61.9792              B36
28     male  41.0      0      0         113054   30.5000              A21
..      …   …    …    …            …       …              …
404    male  43.0      1      0          17765   27.7208              D40
405    male  20.0      0      0  SC/PARIS 2166   13.8625              D38
407    male  50.0      1      1         113503  211.5000              C80
411  female  37.0      1      0          19928   90.0000              C78
414  female  39.0      0      0       PC 17758  108.9000             C105

     Embarked
12          2
14          2
```

```
24           1
26           1
28           2
..         …
404          1
405          1
407          1
411          3
414          1

[87 rows x 11 columns]
```

[68]:
```
from sklearn.model_selection import train_test_split
```

[69]:
```
x_train,x_test,y_train,y_test=train_test_split(x,y,train_size=0.70)
```

[70]:
```
from sklearn.ensemble import RandomForestClassifier
```

[71]:
```
rfc=RandomForestClassifier()
rfc.fit(x_train,y_train)
```

[71]:
```
RandomForestClassifier()
```

[72]:
```
parameters={'max_depth':[1,2,3,4,5],
            'min_samples_leaf':[5,10,15,20,25],
            'n_estimators':[10,20,30,40,50]
}
```

[73]:
```
from sklearn.model_selection import GridSearchCV
grid_search␣
 ↪=GridSearchCV(estimator=rfc,param_grid=parameters,cv=2,scoring="accuracy")
grid_search.fit(x_train,y_train)
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/_split.py:700:
UserWarning: The least populated class in y has only 1 members, which is less
than n_splits=2.
  warnings.warn(
```

[73]:
```
GridSearchCV(cv=2, estimator=RandomForestClassifier(),
             param_grid={'max_depth': [1, 2, 3, 4, 5],
                         'min_samples_leaf': [5, 10, 15, 20, 25],
                         'n_estimators': [10, 20, 30, 40, 50]},
             scoring='accuracy')
```

[74]:
```
grid_search.best_score_
```

[74]:
```
0.7
```

[75]:

```
rfc_best=grid_search.best_estimator_
```

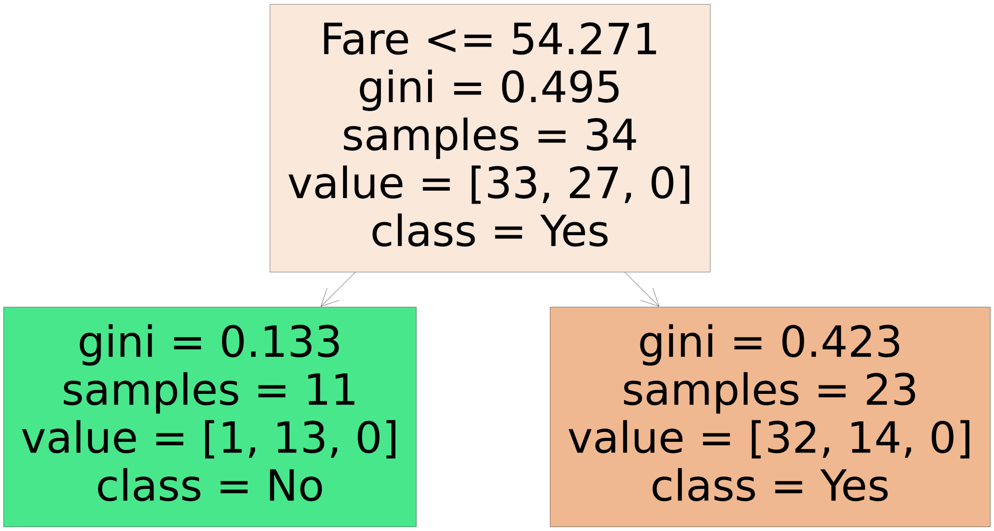
[76]:

```
from sklearn.tree import plot_tree

plt.figure(figsize=(80,40))
plot_tree(rfc_best.estimators_[5],feature_names=x.
 ↪columns,class_names=['Yes','No'],filled=True)
```

[76]:

```
[Text(0.5, 0.75, 'Fare <= 54.271\ngini = 0.495\nsamples = 34\nvalue = [33, 27,
0]\nclass = Yes'),
 Text(0.25, 0.25, 'gini = 0.133\nsamples = 11\nvalue = [1, 13, 0]\nclass = No'),
 Text(0.75, 0.25, 'gini = 0.423\nsamples = 23\nvalue = [32, 14, 0]\nclass =
Yes')]
```


Fare <= 54.271
gini = 0.495
samples = 34
value = [33, 27, 0]
class = Yes
gini = 0.133
samples = 11
value = [1, 13, 0]
class = No
gini = 0.423
samples = 23
value = [32, 14, 0]
class = Yes