import pandas as pd
import numpy as np
from sklearn.tree import DecisionTreeClassifier, export\_graphviz
from sklearn.model\_selection import train\_test\_split
from sklearn import metrics
import pydotplus
from IPython.display import Image

## Załadowanie danych

data=pd.read\_csv('tennis.csv')
data.head()

	Day	Outlook	Temperature	Humidity	Wind	PlayTennis
0	D1	Sunny	Hot	High	Weak	No
1	D2	Sunny	Hot	High	Strong	No
2	D3	Overcast	Hot	High	Weak	Yes
3	D4	Rain	Mild	High	Weak	Yes
4	D5	Rain	Cool	Normal	Weak	Yes

## Dane z tabeli (bez atrybutu PlayTennis) zapisujemy w kodowaniu one-hot

one\_hot\_data = pd.get\_dummies(data[ ['Outlook','Temperature'] ])
one\_hot\_data

	Outlook_Overcast	Outlook_Rain	Outlook_Sunny	Temperature_Cool	Temperature_Hot	Temperature_Mild
0	0	0	1	0	1	0
1	0	0	1	0	1	0
2	1	0	0	0	1	0
3	0	1	0	0	0	1
4	0	1	0	1	0	0
5	0	1	0	1	0	0
6	1	0	0	1	0	0
7	0	0	1	0	0	1
8	0	0	1	1	0	0
9	0	1	0	0	0	1
10	0	0	1	0	0	1
11	1	0	0	0	0	1
12	1	0	0	0	1	0
13	0	1	0	0	0	1

## Objekt DecisionTreeClassifier

```
clf = DecisionTreeClassifier()
clf train = clf.fit(one hot data, data['PlayTennis'])
Rysujemy drzewo decyzyjne
#print(export graphviz(clf train, None))
#Create Dot Data
dot data = export graphviz(clf train, out file=None, feature names=list(one hot data.columns.values),
                               class names=['Yes', 'No'], rounded=True, filled=True)
graph = pydotplus.graph_from_dot_data(dot_data)
# Show graph
Image(graph.create_png())
                                                      Outlook Overcast <= 0.5
                                                             gini = 0.459
                                                            samples = 14
                                                            value = [5, 9]
                                                             class = No
                                                        True
                                                                          False
                                           Temperature Hot <= 0.5
                                                                          aini = 0.0
                                                  gini = 0.5
                                                                         samples = 4
                                                samples = 10
                                                                         value = [0, 4]
                                                value = [5, 5]
                                                                          class = No
                                                 class = Yes
                              Temperature Mild <= 0.5
                                                              gini = 0.0
                                    gini = \overline{0}.469
                                                             samples = 2
                                    samples = 8
                                                             value = [2, 0]
                                    value = [3, 5]
                                                             class = Yes
                                     class = No
                                             Outlook Sunny <= 0.5
                 Outlook Sunny <= 0.5
                                                   gini = 0.48
                       gini = 0.444
                      samples = 3
                                                  samples = 5
                      value = [1, 2]
                                                  value = [2, 3]
                       class = No
                                                   class = No
       gini = 0.5
                          gini = 0.0
                                                gini = 0.444
                                                                    gini = 0.5
       samples = 2
                                                samples = 3
                         samples = 1
                                                                   samples = 2
      value = [1, 1]
                         value = [0, 1]
                                                value = [1, 2]
                                                                  value = [1, 1]
       class = Yes
                         class = No
                                                 class = No
                                                                   class = Yes
```

Testujemy model dla przypadku o parametrach: \*\* Rain,Hot\*\* (zakodowany jako one-hot)

```
prediction = clf_train.predict([[0,1,0,0,1,0]])
prediction
```

/usr/local/lib/python3.8/dist-packages/sklearn/base.py:450: UserWarning: X does not have valid feature names, but DecisionTreeClassifier was fitted with feature names warnings.warn(array(['No'], dtype=object)

✓ 0 s ukończono o 11:14

×