DecisionTree

March 25, 2023

[77]:
```
import pandas as pd
d = pd.read_csv('student-por.csv', sep=';')
len(d)
```

[77]:
```
649
```

[78]:
```
d['pass'] = d.apply(lambda row: 1 if (row['G1']+row['G2']+row ['G3'])>= 35 else␣
 ↪0,axis=1)
d = d.drop(['G1','G2','G3'], axis=1)
d.head()
```

[78]:
```
  school sex  age address famsize Pstatus  Medu  Fedu     Mjob      Fjob  …  \
0     GP   F   18       U     GT3       A     4     4  at_home   teacher  …
1     GP   F   17       U     GT3       T     1     1  at_home     other  …
2     GP   F   15       U     LE3       T     1     1  at_home     other  …
3     GP   F   15       U     GT3       T     4     2   health  services  …
4     GP   F   16       U     GT3       T     3     3    other     other  …

  internet romantic  famrel  freetime  goout Dalc Walc health absences pass
0       no       no       4         3      4    1    1      3        4    0
1      yes       no       5         3      3    1    1      3        2    0
2      yes       no       4         3      2    2    3      3        6    1
3      yes      yes       3         2      2    1    1      5        0    1
4       no       no       4         3      2    1    2      5        0    1

[5 rows x 31 columns]
```

[79]:
```
d = pd.
 ↪get_dummies(d,columns=['sex','school','address','famsize','Pstatus','Mjob','Fjob','reason',
d.head()
```

[79]:
```
   age  Medu  Fedu  traveltime  studytime  failures  famrel  freetime  goout  \
0   18     4     4           2          2         0       4         3      4
1   17     1     1           1          2         0       5         3      3
2   15     1     1           1          2         0       4         3      2
3   15     4     2           1          3         0       3         2      2
4   16     3     3           1          2         0       4         3      2
```

```
   Dalc  …  activities_no  activities_yes  nursery_no  nursery_yes  \
0     1  …              1               0           0            1
1     1  …              1               0           1            0
2     2  …              1               0           0            1
3     1  …              0               1           0            1
4     1  …              1               0           0            1

   higher_no  higher_yes  internet_no  internet_yes  romantic_no  romantic_yes
0          0           1            1             0            1             0
1          0           1            0             1            1             0
2          0           1            0             1            1             0
3          0           1            0             1            0             1
4          0           1            1             0            1             0

[5 rows x 57 columns]
```

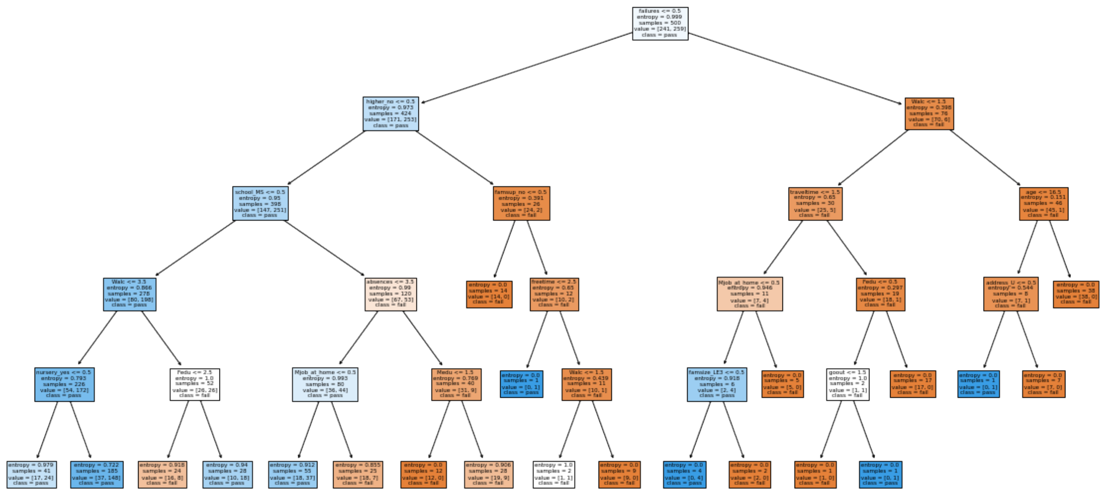
[80]:
```
d = d.sample(frac=1)
d_train = d[:500]
d_test = d[500:]

d_train_att = d_train.drop(['pass'], axis=1)
d_train_pass = d_train['pass']

d_test_att = d_test.drop(['pass'], axis=1)
d_test_pass = d_test['pass']

d_att = d.drop(['pass'], axis=1)
d_pass = d['pass']

import numpy as np
("Passing: %d out of %d (%.2f%%)" % (np.sum(d_pass), len(d_pass), 100*float(np.
 ↪sum(d_pass)) / len(d_pass)))
```

[80]:
```
'Passing: 328 out of 649 (50.54%)'
```

[81]:
```
from sklearn import tree
t = tree.DecisionTreeClassifier(criterion="entropy", max_depth=5)
t = t.fit(d_train_att, d_train_pass)
print(tree.export_text(t))
```

```
|--- feature_5 <= 0.50
|   |--- feature_50 <= 0.50
|   |   |--- feature_16 <= 0.50
|   |   |   |--- feature_10 <= 3.50
|   |   |   |   |--- feature_49 <= 0.50
|   |   |   |   |   |--- class: 1
|   |   |   |   |--- feature_49 >  0.50
|   |   |   |   |   |--- class: 1
|   |   |   |--- feature_10 >  3.50
|   |   |   |   |--- feature_2 <= 2.50
|   |   |   |   |   |--- class: 0
|   |   |   |   |--- feature_2 >  2.50
|   |   |   |   |   |--- class: 1
|   |   |--- feature_16 >  0.50
|   |   |   |--- feature_12 <= 3.50
|   |   |   |   |--- feature_23 <= 0.50
|   |   |   |   |   |--- class: 1
|   |   |   |   |--- feature_23 >  0.50
|   |   |   |   |   |--- class: 0
|   |   |   |--- feature_12 >  3.50
|   |   |   |   |--- feature_1 <= 1.50
|   |   |   |   |   |--- class: 0
|   |   |   |   |--- feature_1 >  1.50
|   |   |   |   |   |--- class: 0
|   |--- feature_50 >  0.50
|   |   |--- feature_42 <= 0.50
|   |   |   |--- class: 0
|   |   |--- feature_42 >  0.50
|   |   |   |--- feature_7 <= 2.50
|   |   |   |   |--- class: 1
|   |   |   |--- feature_7 >  2.50
|   |   |   |   |--- feature_10 <= 1.50
|   |   |   |   |   |--- class: 0
|   |   |   |   |--- feature_10 >  1.50
|   |   |   |   |   |--- class: 0
|--- feature_5 >  0.50
|   |--- feature_10 <= 1.50
|   |   |--- feature_3 <= 1.50
|   |   |   |--- feature_23 <= 0.50
|   |   |   |   |--- feature_20 <= 0.50
|   |   |   |   |   |--- class: 1
|   |   |   |   |--- feature_20 >  0.50
|   |   |   |   |   |--- class: 0
|   |   |   |--- feature_23 >  0.50
|   |   |   |   |--- class: 0
|   |   |--- feature_3 >  1.50
|   |   |   |--- feature_2 <= 0.50
|   |   |   |   |--- feature_8 <= 1.50
|   |   |   |   |   |--- class: 0
|   |   |   |   |--- feature_8 >  1.50
|   |   |   |   |   |--- class: 1
|   |   |   |--- feature_2 >  0.50
|   |   |   |   |--- class: 0
|   |--- feature_10 >  1.50
|   |   |--- feature_0 <= 16.50
|   |   |   |--- feature_18 <= 0.50
|   |   |   |   |--- class: 1
|   |   |   |--- feature_18 >  0.50
|   |   |   |   |--- class: 0
|   |   |--- feature_0 >  16.50
|   |   |   |--- class: 0
```

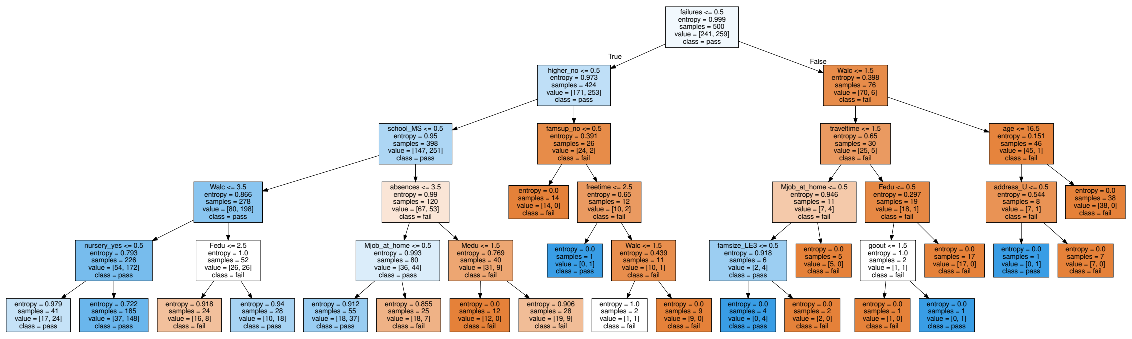
[82]:
```
fig = plt.figure(figsize=(20,10))
_ = tree.plot_tree(t,
                   feature_names=list(d_train_att),
                   class_names=["fail", "pass"],
                   filled=True)
```


failures <= 0.5 entropy = 0.999 samples = 500 value = [241, 259] class = pass
higher_no <= 0.5 entropy = 0.973 samples = 424 value = [171, 253] class = pass
Walc <= 1.5 entropy = 0.398 samples = 76 value = [70, 6] class = fail
school_MS <= 0.5 entropy = 0.95 samples = 398 value = [147, 251] class = pass
famsup_no <= 0.5 entropy = 0.391 samples = 26 value = [24, 2] class = fail
traveltime <= 1.5 entropy = 0.65 samples = 30 value = [25, 5] class = fail
age <= 16.5 entropy = 0.151 samples = 46 value = [45, 1] class = fail
Walc <= 3.5 entropy = 0.866 samples = 278 value = [80, 198] class = pass
absences <= 3.5 entropy = 0.99 samples = 120 value = [67, 53] class = fail
entropy = 0.0 samples = 14 value = [14, 0] class = fail
freetime <= 2.5 entropy = 0.65 samples = 12 value = [10, 2] class = fail
Mjob_at_home <= 0.5 entropy = 0.946 samples = 11 value = [7, 4] class = fail
Fedu <= 0.5 entropy = 0.297 samples = 19 value = [18, 1] class = fail
address_U <= 0.5 entropy = 0.544 samples = 8 value = [7, 1] class = fail
entropy = 0.0 samples = 38 value = [38, 0] class = fail
nursery_yes <= 0.5 entropy = 0.793 samples = 226 value = [54, 172] class = pass
Fedu <= 2.5 entropy = 1.0 samples = 52 value = [26, 26] class = fail
Mjob_at_home <= 0.5 entropy = 0.993 samples = 80 value = [36, 44] class = pass
Medu <= 1.5 entropy = 0.769 samples = 40 value = [31, 9] class = fail
entropy = 0.0 samples = 1 value = [0, 1] class = pass
Walc <= 1.5 entropy = 0.439 samples = 11 value = [10, 1] class = fail
famsize_LE3 <= 0.5 entropy = 0.918 samples = 6 value = [2, 4] class = pass
entropy = 0.0 samples = 5 value = [5, 0] class = fail
goout <= 1.5 entropy = 1.0 samples = 2 value = [1, 1] class = fail
entropy = 0.0 samples = 17 value = [17, 0] class = fail
entropy = 0.0 samples = 1 value = [0, 1] class = pass
entropy = 0.0 samples = 7 value = [7, 0] class = fail
entropy = 0.979 samples = 41 value = [17, 24] class = pass
entropy = 0.722 samples = 185 value = [37, 148] class = pass
entropy = 0.918 samples = 24 value = [16, 8] class = fail
entropy = 0.94 samples = 28 value = [10, 18] class = pass
entropy = 0.912 samples = 55 value = [18, 37] class = pass
entropy = 0.855 samples = 25 value = [18, 7] class = fail
entropy = 0.0 samples = 12 value = [12, 0] class = fail
entropy = 0.906 samples = 28 value = [19, 9] class = fail
entropy = 1.0 samples = 2 value = [1, 1] class = fail
entropy = 0.0 samples = 9 value = [9, 0] class = fail
entropy = 0.0 samples = 4 value = [0, 4] class = pass
entropy = 0.0 samples = 2 value = [2, 0] class = fail
entropy = 0.0 samples = 1 value = [1, 0] class = fail
entropy = 0.0 samples = 1 value = [0, 1] class = pass

[83]:
```
fig.savefig("decistion_tree.png")
```

[84]:
```
import graphviz

dot_data = tree.export_graphviz(t, out_file=None,
                                feature_names=list(d_train_att),
                                class_names=["fail", "pass"],
                                filled=True)

graph = graphviz.Source(dot_data, format="png")
graph
```

[84]:


failures <= 0.5 entropy = 0.999 samples = 500 value = [241, 259] class = pass
True
False
higher_no <= 0.5 entropy = 0.973 samples = 424 value = [171, 253] class = pass
Walc <= 1.5 entropy = 0.398 samples = 76 value = [70, 6] class = fail
school_MS <= 0.5 entropy = 0.95 samples = 398 value = [147, 251] class = pass
famsup_no <= 0.5 entropy = 0.391 samples = 26 value = [24, 2] class = fail
traveltime <= 1.5 entropy = 0.65 samples = 30 value = [25, 5] class = fail
age <= 16.5 entropy = 0.151 samples = 46 value = [45, 1] class = fail
Walc <= 3.5 entropy = 0.866 samples = 278 value = [80, 198] class = pass
absences <= 3.5 entropy = 0.99 samples = 120 value = [67, 53] class = fail
entropy = 0.0 samples = 14 value = [14, 0] class = fail
freetime <= 2.5 entropy = 0.65 samples = 12 value = [10, 2] class = fail
Mjob_at_home <= 0.5 entropy = 0.946 samples = 11 value = [7, 4] class = fail
Fedu <= 0.5 entropy = 0.297 samples = 19 value = [18, 1] class = fail
address_U <= 0.5 entropy = 0.544 samples = 8 value = [7, 1] class = fail
entropy = 0.0 samples = 38 value = [38, 0] class = fail
nursery_yes <= 0.5 entropy = 0.793 samples = 226 value = [54, 172] class = pass
Fedu <= 2.5 entropy = 1.0 samples = 52 value = [26, 26] class = fail
Mjob_at_home <= 0.5 entropy = 0.993 samples = 80 value = [36, 44] class = pass
Medu <= 1.5 entropy = 0.769 samples = 40 value = [31, 9] class = fail
entropy = 0.0 samples = 1 value = [0, 1] class = pass
Walc <= 1.5 entropy = 0.439 samples = 11 value = [10, 1] class = fail
famsize_LE3 <= 0.5 entropy = 0.918 samples = 6 value = [2, 4] class = pass
entropy = 0.0 samples = 5 value = [5, 0] class = fail
goout <= 1.5 entropy = 1.0 samples = 2 value = [1, 1] class = fail
entropy = 0.0 samples = 17 value = [17, 0] class = fail
entropy = 0.0 samples = 1 value = [0, 1] class = pass
entropy = 0.0 samples = 7 value = [7, 0] class = fail
entropy = 0.979 samples = 41 value = [17, 24] class = pass
entropy = 0.722 samples = 185 value = [37, 148] class = pass
entropy = 0.918 samples = 24 value = [16, 8] class = fail
entropy = 0.94 samples = 28 value = [10, 18] class = pass
entropy = 0.912 samples = 55 value = [18, 37] class = pass
entropy = 0.855 samples = 25 value = [18, 7] class = fail
entropy = 0.0 samples = 12 value = [12, 0] class = fail
entropy = 0.906 samples = 28 value = [19, 9] class = fail
entropy = 1.0 samples = 2 value = [1, 1] class = fail
entropy = 0.0 samples = 9 value = [9, 0] class = fail
entropy = 0.0 samples = 4 value = [0, 4] class = pass
entropy = 0.0 samples = 2 value = [2, 0] class = fail
entropy = 0.0 samples = 1 value = [1, 0] class = fail
entropy = 0.0 samples = 1 value = [0, 1] class = pass

[85]:
```python
graph.render("decision_tree_graphivz")
```

[85]: 'decision_tree_graphivz.png'

[87]:
```python
tree.export_graphviz(clf, out_file="student-performance.dot", label="all",␣
 ↪impurity=False, proportion=True,
                         feature_names=list(d_train_att), class_names=["fail",␣
 ↪"pass"],
                         filled=True, rounded=True)
```

[88]:
```python
t.score(d_test_att, d_test_pass)
```

[88]: 0.6577181208053692

[89]:
```python
from sklearn.model_selection import cross_val_score
scores = cross_val_score(clf, d_att, d_pass, cv=5)
print("Accuracy: %0.2f (+/- %0.2f)" % (scores.mean(), scores.std() * 2))
```

```
Accuracy: 0.66 (+/- 0.07)
```

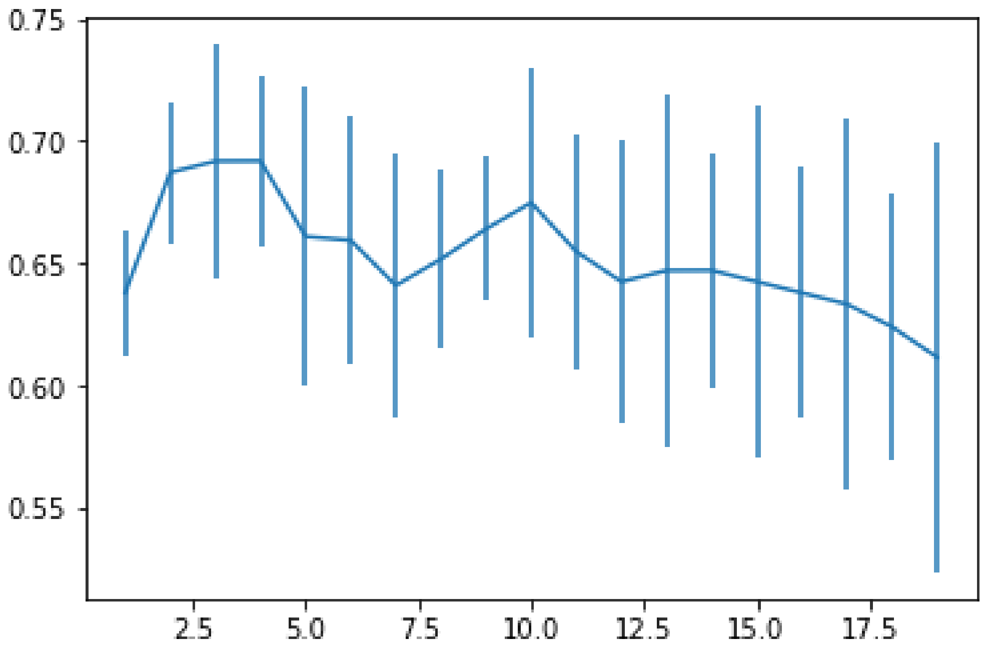
[90]:
```python
for max_depth in range(1, 20):
    t = tree.DecisionTreeClassifier(criterion="entropy", max_depth=max_depth)
    scores = cross_val_score(t, d_att, d_pass, cv=5)
    print("Max depth: %d, Accuracy: %0.2f (+/- %0.2f)" % (max_depth, scores.
 ↪mean(), scores.std() * 2))
```

```
Max depth: 1, Accuracy: 0.64 (+/- 0.03)
Max depth: 2, Accuracy: 0.69 (+/- 0.03)
Max depth: 3, Accuracy: 0.69 (+/- 0.05)
Max depth: 4, Accuracy: 0.69 (+/- 0.04)
Max depth: 5, Accuracy: 0.66 (+/- 0.07)
Max depth: 6, Accuracy: 0.65 (+/- 0.07)
Max depth: 7, Accuracy: 0.65 (+/- 0.04)
Max depth: 8, Accuracy: 0.64 (+/- 0.06)
Max depth: 9, Accuracy: 0.65 (+/- 0.07)
Max depth: 10, Accuracy: 0.66 (+/- 0.03)
Max depth: 11, Accuracy: 0.65 (+/- 0.04)
Max depth: 12, Accuracy: 0.65 (+/- 0.07)
Max depth: 13, Accuracy: 0.64 (+/- 0.08)
Max depth: 14, Accuracy: 0.63 (+/- 0.05)
Max depth: 15, Accuracy: 0.64 (+/- 0.08)
Max depth: 16, Accuracy: 0.65 (+/- 0.05)
Max depth: 17, Accuracy: 0.64 (+/- 0.05)
Max depth: 18, Accuracy: 0.64 (+/- 0.10)
Max depth: 19, Accuracy: 0.61 (+/- 0.08)
```

[91]:
```
depth_acc = np.empty((19,3), float)
i = 0
for max_depth in range(1, 20):
    t = tree.DecisionTreeClassifier(criterion="entropy", max_depth=max_depth)
    scores = cross_val_score(t, d_att, d_pass, cv=5)
    depth_acc[i,0] = max_depth
    depth_acc[i,1] = scores.mean()
    depth_acc[i,2] = scores.std() * 2
    i += 1

depth_acc
```

[91]:
```
array([[ 1.        ,  0.63790101,  0.02584084],
       [ 2.        ,  0.68723912,  0.02897674],
       [ 3.        ,  0.69179487,  0.0477116 ],
       [ 4.        ,  0.69186643,  0.03531423],
       [ 5.        ,  0.66097794,  0.06087809],
       [ 6.        ,  0.65942755,  0.05043652],
       [ 7.        ,  0.64093023,  0.0536498 ],
       [ 8.        ,  0.65172332,  0.03627785],
       [ 9.        ,  0.66407871,  0.02941166],
       [10.        ,  0.67482409,  0.05521507],
       [11.        ,  0.65478831,  0.04813747],
       [12.        ,  0.64248062,  0.05760953],
       [13.        ,  0.64706023,  0.07178249],
       [14.        ,  0.64710793,  0.04833142],
       [15.        ,  0.64249255,  0.07192719],
       [16.        ,  0.63791294,  0.05120491],
       [17.        ,  0.6332737 ,  0.07587234],
       [18.        ,  0.62400716,  0.0544644 ],
       [19.        ,  0.61160405,  0.08790366]])
```

[92]:
```
import matplotlib.pyplot as plt
fig, ax = plt.subplots()
ax.errorbar(depth_acc[:,0], depth_acc[:,1], yerr=depth_acc[:,2])
plt.show()
```


0.75
0.70
0.65
0.60
0.55
2.5
5.0
7.5
10.0
12.5
15.0
17.5

```
[ ]:
```