Birla Institute of Applied Sciences
बिरला इंस्टिट्यूट ऑफ़ अप्लाइड साइंसेस
Bhimtal, Distt: Nainital, Uttarakhand- 263136

Pattern Recognition

- S. S. Samant

Decision Tree Classifier

These are *multistage* decision systems in which classes are sequentially rejected until we reach a finally accepted class. To this end, the feature space is split into unique regions, corresponding to the classes, *in a sequential manner*.

- A *splitting criterion* must be adopted according to which the best split from the set of candidate ones is chosen.
- A stop-splitting rule is required that controls the growth of the tree, and a node is declared as a terminal one (*leaf*).
- A rule is required that assigns each leaf to a specific class.

Information gain (entropy)

RID	*age*	*income*	*student*	*credit_rating*	*Class: buys_computer*
1	youth	high	no	fair	no
2	youth	high	no	excellent	no
3	middle_aged	high	no	fair	yes
4	senior	medium	no	fair	yes
5	senior	low	yes	fair	yes
6	senior	low	yes	excellent	no
7	middle_aged	low	yes	excellent	yes
8	youth	medium	no	fair	no
9	youth	low	yes	fair	yes
10	senior	medium	yes	fair	yes
11	youth	medium	yes	excellent	yes
12	middle_aged	medium	no	excellent	yes
13	middle_aged	high	yes	fair	yes
14	senior	medium	no	excellent	no

$$Info(D) = -\frac{9}{14}\log_2\left(\frac{9}{14}\right) - \frac{5}{14}\log_2\left(\frac{5}{14}\right) = 0.940 \text{ bits.}$$

What if we split on age?

age?

youth

income	*student*	*credit_rating*	*class*
high	no	fair	no
high	no	excellent	no
medium	no	fair	no
low	yes	fair	yes
medium	yes	excellent	yes

middle_aged

income	*student*	*credit_rating*	*class*
high	no	fair	yes
low	yes	excellent	yes
medium	no	excellent	yes
high	yes	fair	yes

senior

income	*student*	*credit_rating*	*class*
medium	no	fair	yes
low	yes	fair	yes
low	yes	excellent	no
medium	yes	fair	yes
medium	no	excellent	no

Information gain (entropy)

Birla Institute of Applied Sciences
बिरला इंस्टिट्यूट ऑफ़ अप्लाइड साइंसेस
Bhimtal, Distt: Nainital, Uttarakhand- 263136

RID	age	income	student	credit_rating	Class: buys_computer
1	youth	high	no	fair	no
2	youth	high	no	excellent	no
3	middle_aged	high	no	fair	yes
4	senior	medium	no	fair	yes
5	senior	low	yes	fair	yes
6	senior	low	yes	excellent	no
7	middle_aged	low	yes	excellent	yes
8	youth	medium	no	fair	no
9	youth	low	yes	fair	yes
10	senior	medium	yes	fair	yes
11	youth	medium	yes	excellent	yes
12	middle_aged	medium	no	excellent	yes
13	middle_aged	high	yes	fair	yes
14	senior	medium	no	excellent	no

$$Info(D) = -\frac{9}{14}\log_2\left(\frac{9}{14}\right) - \frac{5}{14}\log_2\left(\frac{5}{14}\right) = 0.940 \text{ bits.}$$

$$\begin{aligned} Info_{age}(D) = {} & \frac{5}{14} \times \left(-\frac{2}{5}\log_2\frac{2}{5} - \frac{3}{5}\log_2\frac{3}{5}\right) \\ & + \frac{4}{14} \times \left(-\frac{4}{4}\log_2\frac{4}{4} - \frac{0}{4}\log_2\frac{0}{4}\right) \\ & + \frac{5}{14} \times \left(-\frac{3}{5}\log_2\frac{3}{5} - \frac{2}{5}\log_2\frac{2}{5}\right) \\ = {} & 0.694 \text{ bits.} \end{aligned}$$

$$Info_A(D) = \sum_{j=1}^{v} \frac{|D_j|}{|D|} \times Info(D_j)$$

Information gain (entropy)

Birla Institute of Applied Sciences
बिरला इंस्टिट्यूट ऑफ़ अप्लाइड साइंसेस
Bhimtal, Distt: Nainital, Uttarakhand- 263136

RID	age	income	student	credit_rating	Class: buys_computer
1	youth	high	no	fair	no
2	youth	high	no	excellent	no
3	middle_aged	high	no	fair	yes
4	senior	medium	no	fair	yes
5	senior	low	yes	fair	yes
6	senior	low	yes	excellent	no
7	middle_aged	low	yes	excellent	yes
8	youth	medium	no	fair	no
9	youth	low	yes	fair	yes
10	senior	medium	yes	fair	yes
11	youth	medium	yes	excellent	yes
12	middle_aged	medium	no	excellent	yes
13	middle_aged	high	yes	fair	yes
14	senior	medium	no	excellent	no

$$Gain(age) = Info(D) - Info_{age}(D) = 0.940 - 0.694 = 0.246 \text{ bits}$$

$$Gain(income) = 0.029 \text{ bits}$$

$$Gain(student) = 0.151 \text{ bits}$$

$$Gain(credit\_rating) = 0.048 \text{ bits}$$

$$Info_A(D) = \sum_{j=1}^{v} \frac{|D_j|}{|D|} \times Info(D_j)$$

Representing nominal variables

Birla Institute of Applied Sciences
बिरला इंस्टिट्यूट ऑफ़ अप्लाइड साइंसेस
Bhimtal, Distt: Nainital, Uttarakhand- 263136

Python only accepts numeric values for attributes, how to represent **categorical** variables such as age and income?

So, if age takes 3 values ***youth***, ***middle-aged***, **and** ***senior***
We can use 3 – 1 (=2) **dummy variables** for each age-value the variable takes

Dummy variable : a numeric variable that represents **categorical** data, such as gender, race, etc.

Representing nominal variables

Python only accepts numeric values for attributes, how to represent **categorical** variables such as age and income?

So, if age takes 3 values ***youth***, ***middle-aged***, **and** ***senior***
We can use 3 – 1 (=2) **dummy variables** for each age-value the variable takes

Dummy variable : a numeric variable that represents **categorical** data, such as gender, race, etc.

For ex. age can take three values, so we can use 2 dummy variables as:

middle-aged	**senior**	**youth**
1	0	
0	1	
0	0	

Representing nominal variables

Python only accepts numeric values for attributes, how to represent **categorical** variables such as age and income?

So, if age takes 3 values ***youth***, ***middle-aged***, **and** ***senior***
We can use 3 – 1 (=2) **dummy variables** for each age-value the variable takes

Dummy variable : a numeric variable that represents **categorical** data, such as gender, race, etc.

For ex. age can take three values, so we can use 2 dummy variables as:

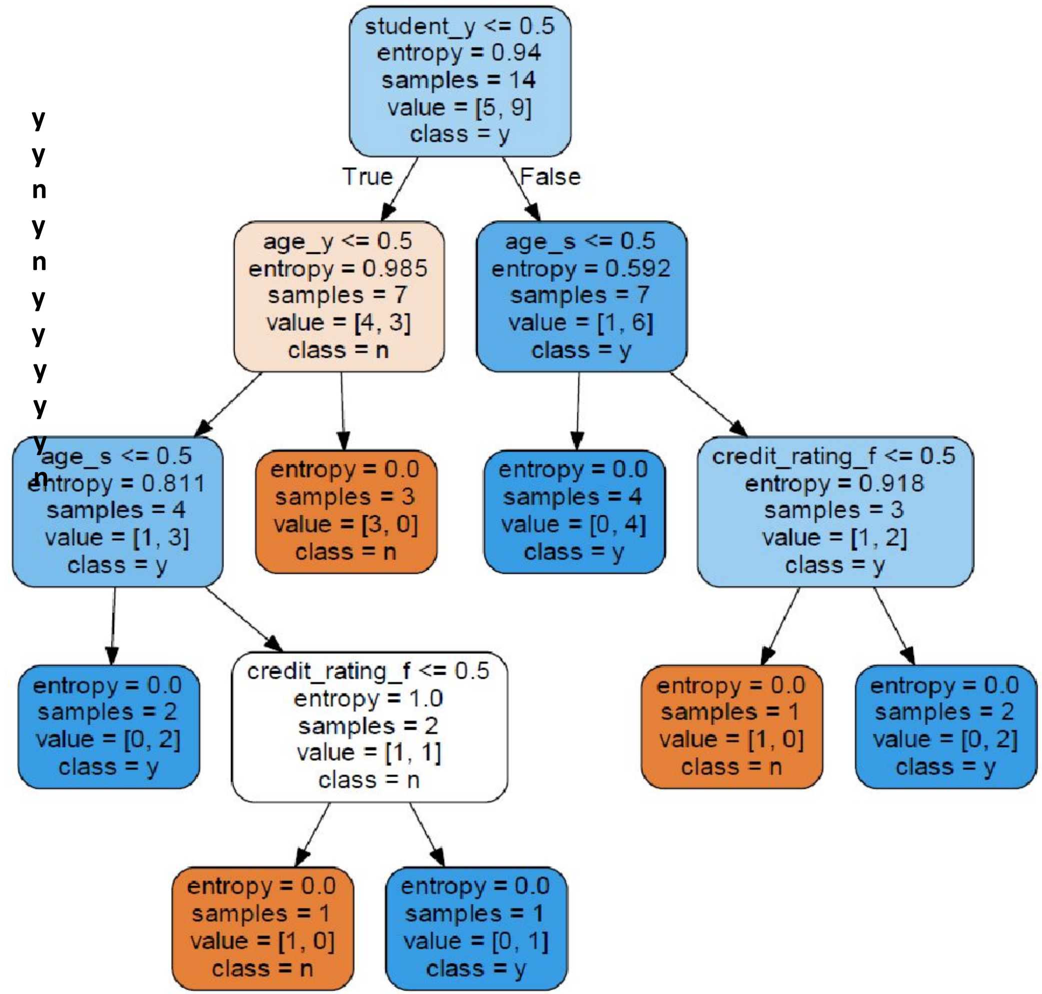
middle-aged	senior	youth
1	0	
0	1	
0	0	

Can infer this value from the other two

Attribute Selection Measures

	age_s	age_y	credit_rating_f	income_l	income_m	student_y	Class
0	0	1	1	0	0	0	n
1	0	1	0	0	0	0	n
2	0	0	1	0	0	0	y
3	1	0	1	0	1	0	y
4	1	0	1	1	0	1	y
5	1	0	0	1	0	1	n
6	0	0	0	1	0	1	y
7	0	1	1	0	1	0	n
8	0	1	1	1	0	1	y
9	1	0	1	0	1	1	y
10	0	1	0	0	1	1	y
11	0	0	0	0	1	0	y
12	0	0	1	0	0	1	y
13	1	0	0	0	1	0	n

RID	age	income	student	credit_rating	Class: buys_computer
1	youth	high	no	fair	no
2	youth	high	no	excellent	no
3	middle_aged	high	no	fair	yes
4	senior	medium	no	fair	yes
5	senior	low	yes	fair	yes
6	senior	low	yes	excellent	no
7	middle_aged	low	yes	excellent	yes
8	youth	medium	no	fair	no
9	youth	low	yes	fair	yes
10	senior	medium	yes	fair	yes
11	youth	medium	yes	excellent	yes
12	middle_aged	medium	no	excellent	yes
13	middle_aged	high	yes	fair	yes
14	senior	medium	no	excellent	no



Dataset

	age_s	age_y	credit_rating_f	income_l	income_m	student_y	Class
0	0	1	1	0	0	0	n
1	0	1	0	0	0	0	n
2	0	0	1	0	0	0	y
3	1	0	1	0	1	0	y
4	1	0	1	1	0	1	y
5	1	0	0	1	0	1	n
6	0	0	0	1	0	1	y
7	0	1	1	0	1	0	n
8	0	1	1	1	0	1	y
9	1	0	1	0	1	1	y
10	0	1	0	0	1	1	y
11	0	0	0	0	1	0	y
12	0	0	1	0	0	1	y
13	1	0	0	0	1	0	n

Initial entropy = 0.9402, entropy on splits (ages_s=?, age_y=?, credit_rating_f=?, income_l=?, income_m=?, student_y=?)

Dataset

	age_s	age_y	credit_rating_f	income_l	income_m	student_y	Class
0	0	1	1	0	0	0	n
1	0	1	0	0	0	0	n
2	0	0	1	0	0	0	y
3	1	0	1	0	1	0	y
4	1	0	1	1	0	1	y
5	1	0	0	1	0	1	n
6	0	0	0	1	0	1	y
7	0	1	1	0	1	0	n
8	0	1	1	1	0	1	y
9	1	0	1	0	1	1	y
10	0	1	0	0	1	1	y
11	0	0	0	0	1	0	y
12	0	0	1	0	0	1	y
13	1	0	0	0	1	0	n

Initial entropy = 0.9402, entropy on splits (ages_s=0.9371, age_y=0.838, credit_rating_f=0.8921, income_l=0.9253, income_m=0.9389, student_y=0.7884)

Dataset

	age_s	age_y	credit_rating_f	income_l	income_m	student_y	Class
0	0	1	1	0	0	0	n
1	0	1	0	0	0	0	n
2	0	0	1	0	0	0	y
3	1	0	1	0	1	0	y
7	0	1	1	0	1	0	n
11	0	0	0	0	1	0	y
13	1	0	0	0	1	0	n

Initial entropy = 0.9852, entropy on splits (ages_s=0.9792, age_y=0.4635, credit_rating_f=0.9649, income_l=0.9852, income_m=0.9649)

Dataset

	age_s	age_y	credit_rating_f	income_l	income_m	student_y	Class
2	0	0	1	0	0	0	y
3	1	0	1	0	1	0	y
11	0	0	0	0	1	0	y
13	1	0	0	0	1	0	n

Initial entropy = 0.8112, entropy on splits (ages_s=0.5, credit_rating_f=0.5, income_l=0.8112, income_m=0.6887)

Dataset

	age_s	age_y	credit_rating_f	income_l	income_m	student_y	Class
2	0	0	1	0	0	0	y
11	0	0	0	0	1	0	y

Initial entropy = 0, so Class='y'

DT implementation

```
#   install graphviz from anaconda prompt by running
#       conda install python-graphviz
#       conda install graphviz
import pandas as pd
from sklearn.tree import DecisionTreeClassifier
from sklearn.tree import export_graphviz
import graphviz

data = pd.DataFrame()
data['age']     = ['y','y','m','s','s','s','m','y','y','s','y','m','m','s']
data['income']  = ['h','h','h','m','l','l','l','m','l','m','m','m','h','m']
data['student']= ['n','n','n','n','y','y','y','n','y','y','y','n','y','n']
data['credit_rating']= ['f','e','f','f','f','e','e','f','f','f','e','e','f','e’]
#print (data)
df = pd.DataFrame([['s','m','y','f'], ['s','m','y','e']],
columns=['age','income','student','credit_rating'])
df2 = data.append(df,ignore_index=True )
fea = pd.get_dummies(df2,drop_first=True)
y =['n','n','y','y','y','n','y','n','y','y','y','y','y','n']
df_train, df_test = fea[:14], fea[14:]
dt = DecisionTreeClassifier(criterion='entropy', random_state=1)
dt.fit(df_train, y)
dot_data = export_graphviz(dt, out_file=None,
     feature_names =list(fea.columns.values), class_names = ['n', 'y'], filled=True,
rounded=True)
graph = graphviz.Source(dot_data)
graph.render('buys_computer')  # saves Dtree in a file buys_computer.pdf
```

Thank You!