Classification Methods

Tozammel Hossain

Data Science & Analytics
University of Missouri

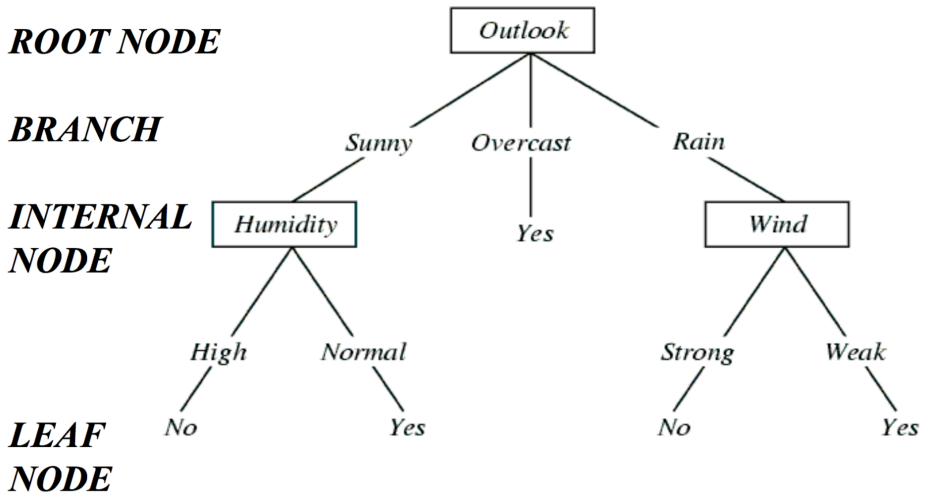
Decision Tree

- **Very popular; Easy to interpret**
- **Rule based classification method**
  - Classifier is defined by a set of decision rules
  - Expressed in terms of **if-else** conditions with logical operators (**AND, OR, NOT**)
- **Solve both classification and regression problem**
  - Focus: classification

Induced Tree

PlayTennis: training examples

Day	Outlook	Temperature	Humidity	Wind	PlayTennis
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rain	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No



Rules:

- if outlook = sunny AND Humidity = Normal: classify Yes
- if outlook = sunny AND Humidity = High: classify No
- if outlook = overcast : classify Yes
- …..

Applications

- Equipment diagnosis
- Medical diagnosis
- Credit card risk analysis
- Robot movement
- Face recognition
- …

Will I play tennis today?

	O	T	H	W	Play?
1	S	H	H	W	-
2	S	H	H	S	-
3	O	H	H	W	+
4	R	M	H	W	+
5	R	C	N	W	+
6	R	C	N	S	-
7	O	C	N	S	+
8	S	M	H	W	-
9	S	C	N	W	+
10	R	M	N	W	+
11	S	M	N	S	+
12	O	M	H	S	+
13	O	H	N	W	+
14	R	M	H	S	-

Outlook: S(unny), O(vercast), R(ainy)

Temperature: H(ot), M(edium), C(ool)

Humidity: H(igh), N(ormal), L(ow)

Wind: S(trong), W(eak)

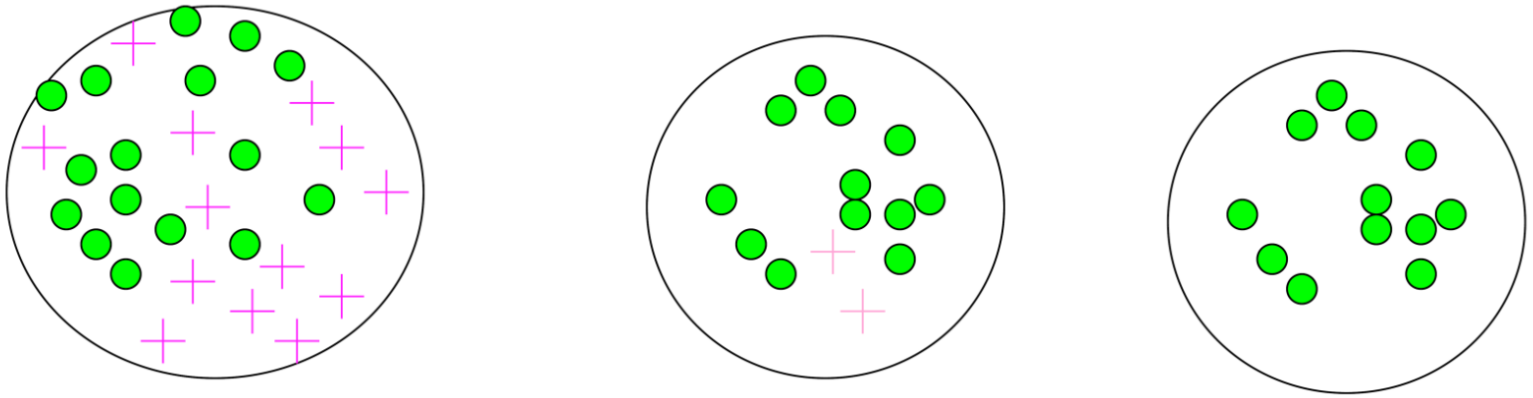
Learning Task:

PlayTennis? = f(outlook, temp, humidity, wind)

Algorithm

- **Finding the optimal tree is a computationally-intractable problem**
- **Rely on heuristics**
  - Popular algorithms
    - ID3, C4.5, C5
  - Greedy, top-down approach, iterative
  - Information gain is used to measure purity of a leaf node
- **Occam's Razor**
  - Simpler is better

Feature selection criterion: Entropy & Information Gain



- **Entropy measures impurity (disorder) of a sample of examples**
- **Pick feature with smallest entropy to split the examples at current iteration**
- **Information Gain**
  - The expected reduction in a set of samples conditioned on a feature

Will I play tennis today?

	O	T	H	W	Play?
1	S	H	H	W	-
2	S	H	H	S	-
3	O	H	H	W	+
4	R	M	H	W	+
5	R	C	N	W	+
6	R	C	N	S	-
7	O	C	N	S	+
8	S	M	H	W	-
9	S	C	N	W	+
10	R	M	N	W	+
11	S	M	N	S	+
12	O	M	H	S	+
13	O	H	N	W	+
14	R	M	H	S	-

Current entropy of "Play?" Variable

P(+) = 9/14; P(-) = 5/14

$H(Play) = - P(+) * \log_2 P(+) - P(-) * \log_2 P(-) = 0.94$

	O	T	H	W	Play?
1	S	H	H	W	-
2	S	H	H	S	-
3	O	H	H	W	+
4	R	M	H	W	+
5	R	C	N	W	+
6	R	C	N	S	-
7	O	C	N	S	+
8	S	M	H	W	-
9	S	C	N	W	+
10	R	M	N	W	+
11	S	M	N	S	+
12	O	M	H	S	+
13	O	H	N	W	+
14	R	M	H	S	-

Outlook = sunny:

P(+) = 2/5; P(-) = 3/5;

H(Play | Outlook = sunny) = 0.971

Outlook = overcast:

P(+) = 4/4; P(-) = 0;

H(Play | Outlook = overcast) = 0

Outlook = rainy:

P(+) = 3/5; P(-) = 2/5;

H(Play | Outlook = rainy) = 0.971

H(Play | outlook) = 5/15 * H(Play | Outlook = sunny) + 4/14 * H(Play | Outlook = overcast) + 5/15 * H(Play | Outlook = rainy)

= 0.694

Information Gain = H(Play) - H(Play | outlook)

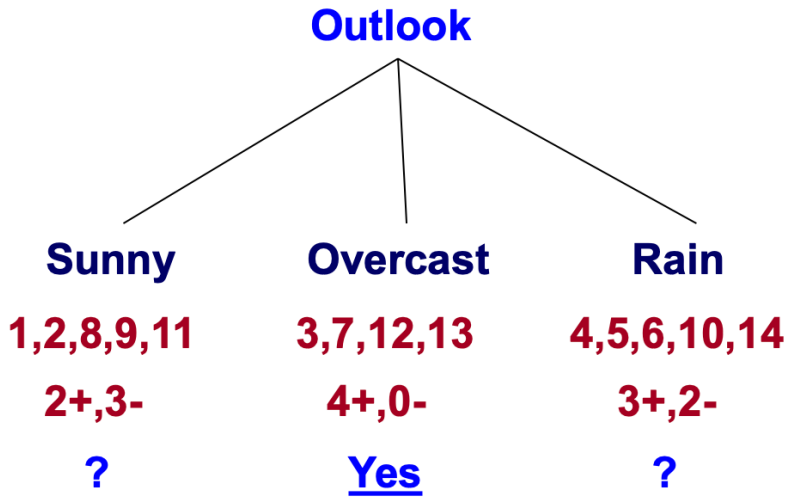
= 0.94 – 0.694

Which feature to split on?

- **Information gain:**
  - Outlook: 0.246
  - Humidity: 0.151
  - Wind: 0.048
  - Temperature: 0.029
- **Split on outlook**
  - Greedy decision, why?

Tree after first iteration

	O	T	H	W	Play?
1	S	H	H	W	-
2	S	H	H	S	-
3	O	H	H	W	+
4	R	M	H	W	+
5	R	C	N	W	+
6	R	C	N	S	-
7	O	C	N	S	+
8	S	M	H	W	-
9	S	C	N	W	+
10	R	M	N	W	+
11	S	M	N	S	+
12	O	M	H	S	+
13	O	H	N	W	+
14	R	M	H	S	-



- **Continue**
  - Every attribute is included in the path
  - or examples within node has the same label

Sklearn Decision Tree

sklearn.tree.DecisionTreeClassifier

class sklearn.tree.**DecisionTreeClassifier**(*, *criterion='gini'*, *splitter='best'*, *max_depth=None*, *min_samples_split=2*, *min_samples_leaf=1*, *min_weight_fraction_leaf=0.0*, *max_features=None*, *random_state=None*, *max_leaf_nodes=None*, *min_impurity_decrease=0.0*, *min_impurity_split=None*, *class_weight=None*, *ccp_alpha=0.0*) ¶ [source]

- **Key parameters**
  - criterion
  - max_depth
  - min_samples_split
  - min_samples_leaf
  - max_leaf_nodes
  - min_impurity_decrease