Decision Trees

Yuntian Deng

Lecture 18

Readings: RN 19.3, PM 7.3.1.

Outline

Learning Goals

- Describe the components of a decision tree.
- Construct a decision tree given an order of testing the features.
- Determine the prediction accuracy of a decision tree on a test set.
- Compute the entropy of a probability distribution.
- Compute the expected information gain for selecting a feature.
- Trace the execution of and implement the algorithm for learning a decision tree.

Examples of Decision Trees

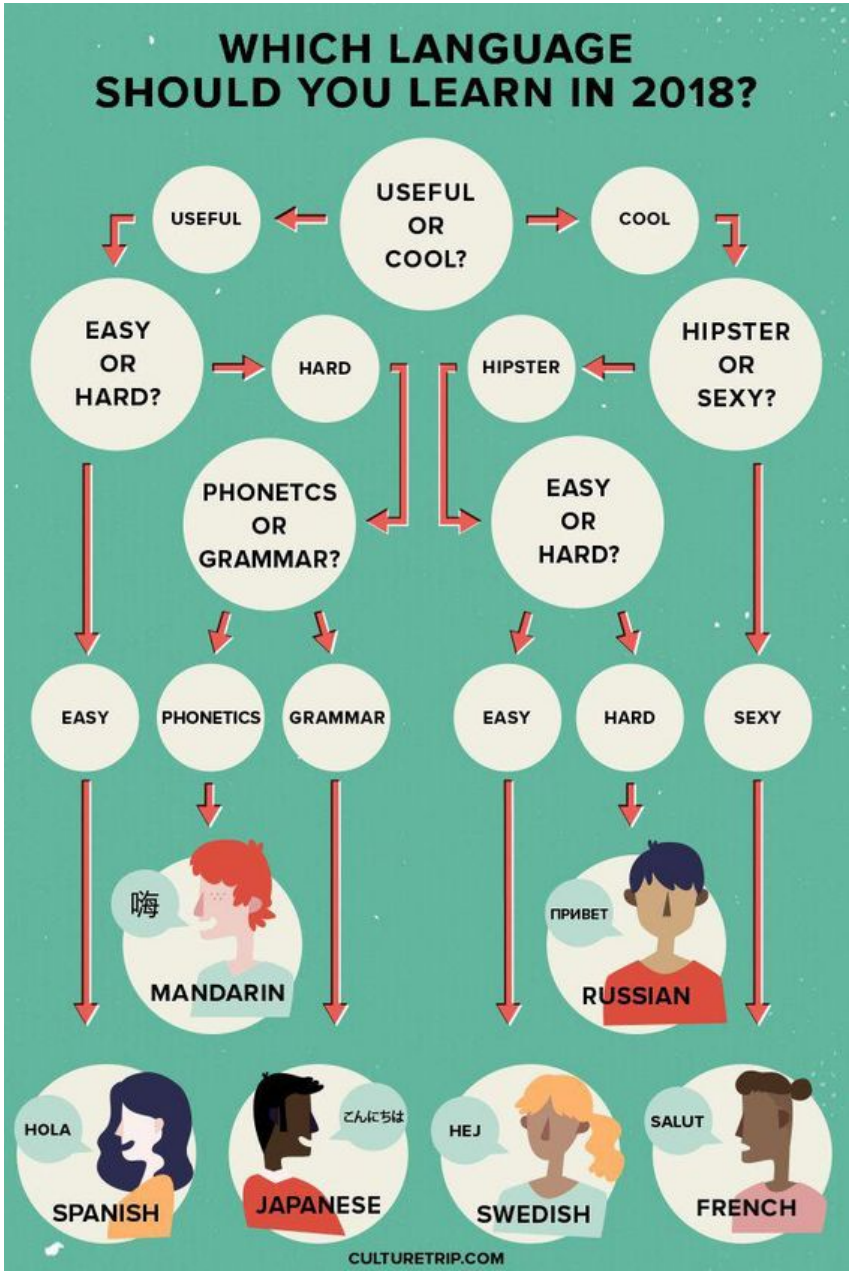
Definition and Classifying an Example

Grow a Full Tree

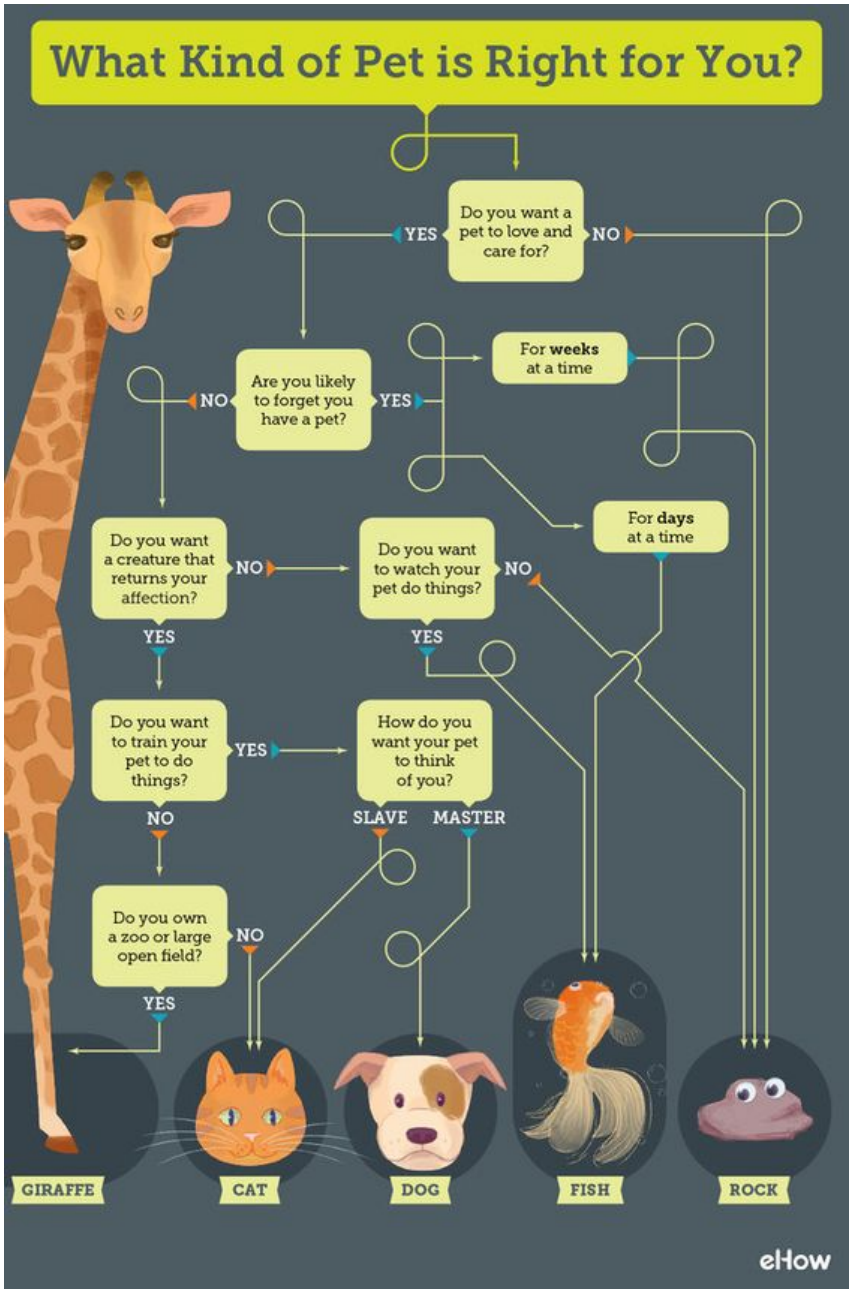
Decision Tree Learning Algorithm

Determine the Order of Testing Features

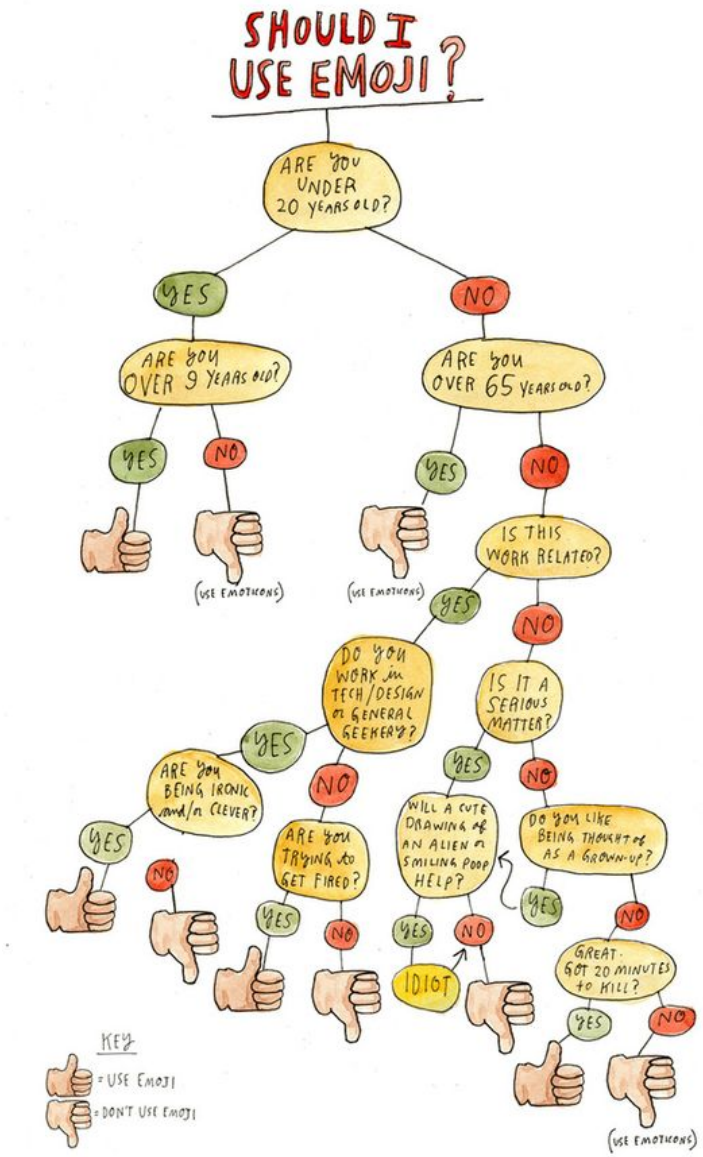
Which language should you learn?



What kind of pet is right for you?



Should I use an emoji?



Running example: Jeeves the valet

Jeeves is a valet to Bertie Wooster. On some days, Bertie likes to play tennis and asks Jeeves to lay out his tennis things and book the court. Jeeves would like to predict whether Bertie will play tennis (and so be a better valet). Each morning over the last two weeks, Jeeves has recorded whether Bertie played tennis on that day and various attributes of the weather (training set).

Jeeves would like to evaluate the classifier he has come up with for predicting whether Bertie will play tennis. Each morning over the next two weeks, Jeeves records more data (test set).

Jeeves the valet - training set

Day	Outlook	Temp	Humidity	Wind	Tennis?
1	Sunny	Hot	High	Weak	No
2	Sunny	Hot	High	Strong	No
3	Overcast	Hot	High	Weak	Yes
4	Rain	Mild	High	Weak	Yes
5	Rain	Cool	Normal	Weak	Yes
6	Rain	Cool	Normal	Strong	No
7	Overcast	Cool	Normal	Strong	Yes
8	Sunny	Mild	High	Weak	No
9	Sunny	Cool	Normal	Weak	Yes
10	Rain	Mild	Normal	Weak	Yes
11	Sunny	Mild	Normal	Strong	Yes
12	Overcast	Mild	High	Strong	Yes
13	Overcast	Hot	Normal	Weak	Yes
14	Rain	Mild	High	Strong	No

Jeeves the valet - test set

Day	Outlook	Temp	Humidity	Wind	Tennis?
1	Sunny	Mild	High	Strong	No
2	Rain	Hot	Normal	Strong	No
3	Rain	Cool	High	Strong	No
4	Overcast	Hot	High	Strong	Yes
5	Overcast	Cool	Normal	Weak	Yes
6	Rain	Hot	High	Weak	Yes
7	Overcast	Mild	Normal	Weak	Yes
8	Overcast	Cool	High	Weak	Yes
9	Rain	Cool	High	Weak	Yes
10	Rain	Mild	Normal	Strong	No
11	Overcast	Mild	High	Weak	Yes
12	Sunny	Mild	Normal	Weak	Yes
13	Sunny	Cool	High	Strong	No
14	Sunny	Cool	High	Weak	No

Examples of Decision Trees

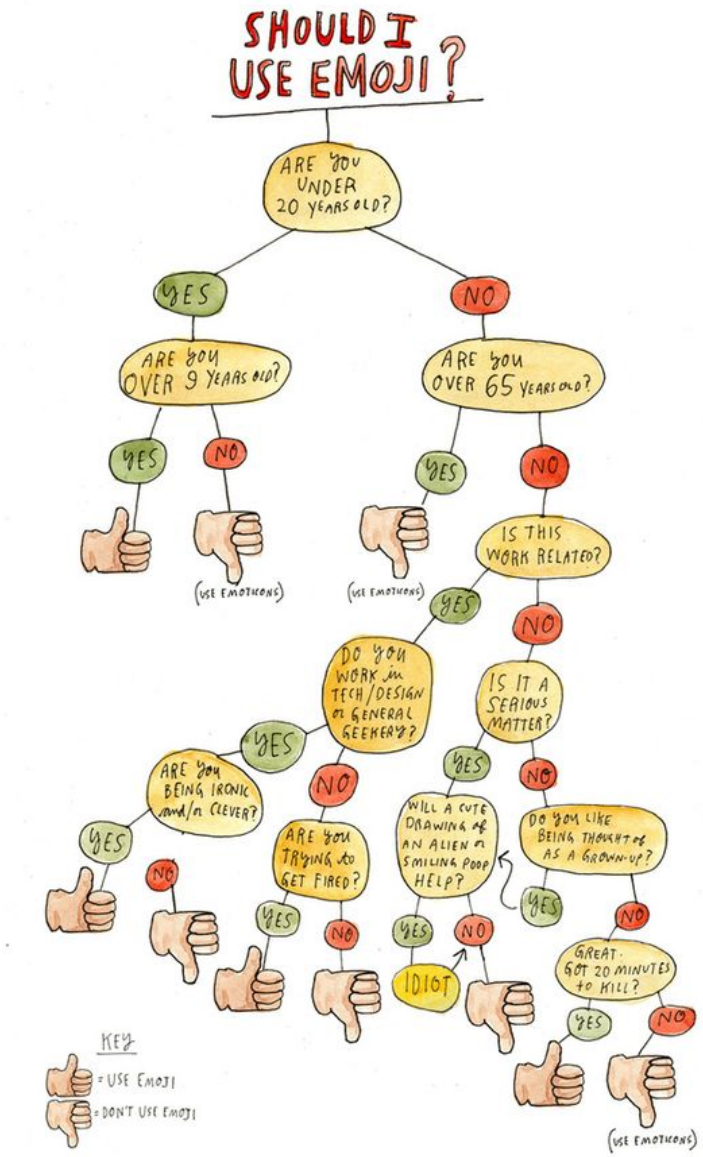
Definition and Classifying an Example

Grow a Full Tree

Decision Tree Learning Algorithm

Determine the Order of Testing Features

What is a decision tree?



What is a decision tree?

- A simple model for supervised classification.

  → One of the simplest, yet useful and widely successful ML algorithms.

- A single discrete target feature.

  → Each value in the domain of the target feature is a class.

- Each internal node performs a Boolean test on an input feature.

- The edges are labeled with values of the input feature.

  → yes/no. true/false.

- Each leaf node specifies a value for the target feature.

  → specifies a class.

Classify an example using a decision tree

- Go down the tree, evaluate each test and follow the corresponding edge.
- When a leaf is reached, return the classification on that leaf.

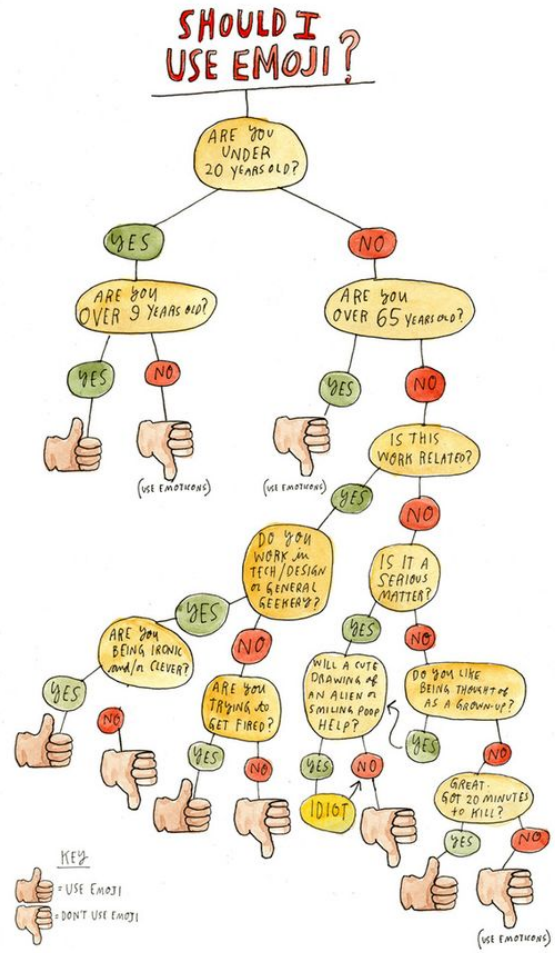
E.g. Should I use an emoji?

- I am 30 years old.
- This is work related.
- I am an accountant.
- I am not trying to get fired.

$\rightarrow$ No.

If we convert a decision tree to a program, what does it look like?

$\rightarrow$ A nested if-then-else structure.



Issues in learning a decision tree

How should we build a decision tree?

- We need to determine an order of testing the input features.
- Given an order of testing the input features,
  we can build a decision tree by splitting the examples.

Issues in learning a decision tree

How should we build a decision tree?

- We need to determine an order of testing the input features.
- Given an order of testing the input features,
  we can build a decision tree by splitting the examples.

Which decision tree should we generate?

- Which order of testing the input features should we use?
  - The search space is too big for systematic search.
  - Solution: greedy (myopic) search.
- Should we grow a full tree or not?
  - A decision tree can represent any discrete function of input features.
  - Need a bias. For example, prefer the smallest tree. (Least depth? Fewest nodes?)

Examples of Decision Trees

Definition and Classifying an Example

Grow a Full Tree

Decision Tree Learning Algorithm

Determine the Order of Testing Features

Construct a Decision Tree
Given an Order of Testing Features

$\rightarrow$ Given an order of testing features, it's straightforward to generate a decision tree.

Construct a decision tree
using the following order of testing features.

- First, test Outlook.
- For Outlook = Sunny, test Temp.
- For Outlook = Rain, test Wind.
- For other branches, test Humidity before testing Wind.

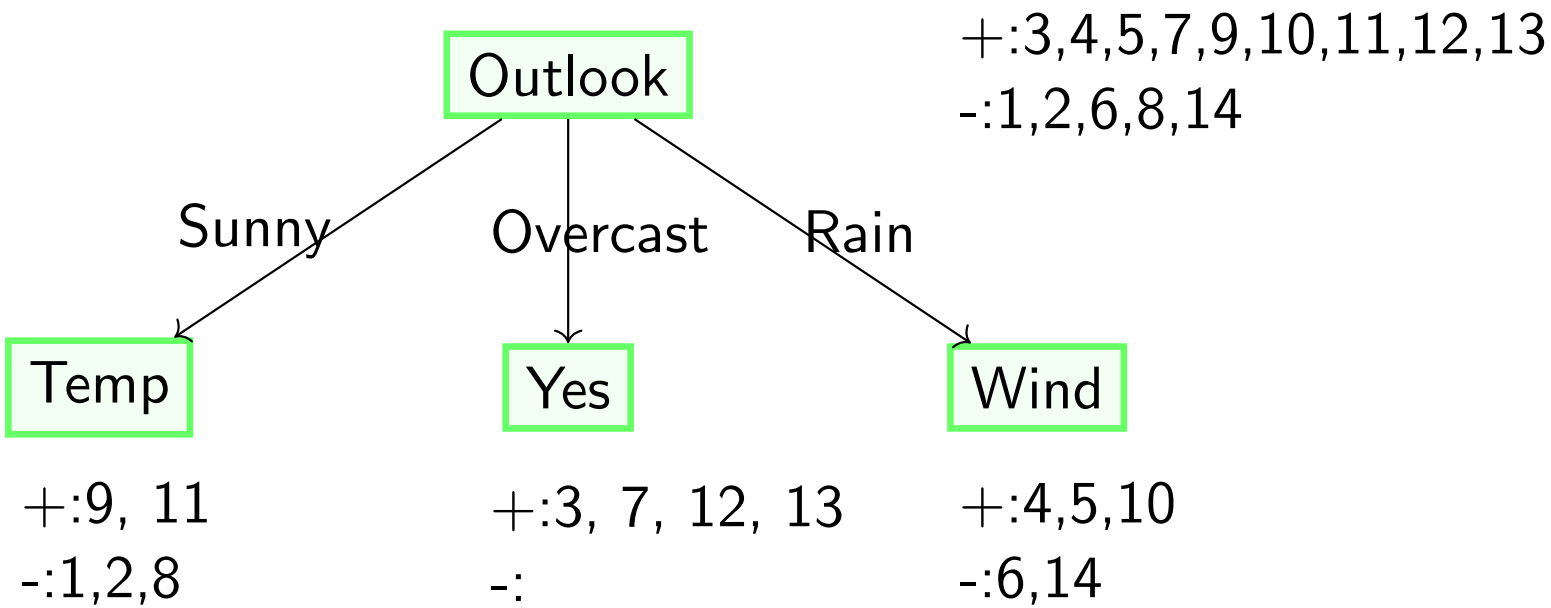
Let's draw the decision tree.

Example: Decision tree

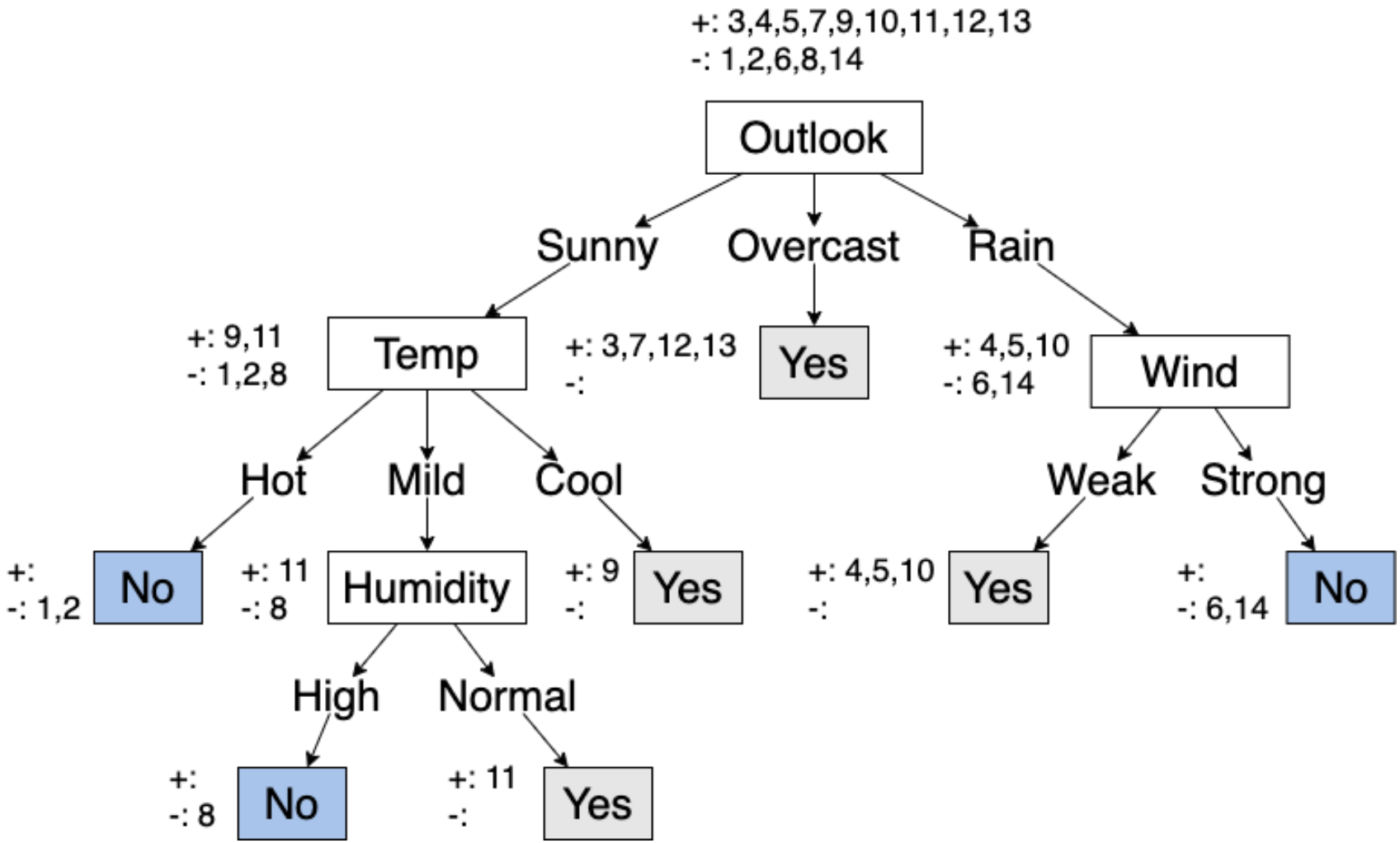
Outlook

+:3,4,5,7,9,10,11,12,13
-:1,2,6,8,14

Example: Decision tree



Example: Decision tree



When do we stop?

We stop in any of the following scenarios:

- All the examples belong to the same class.
- There are no more features to test.
- There are no more examples.

No features left

Day	Outlook	Temp	Humidity	Wind	Tennis?
1	Sunny	Hot	High	Weak	No
2	Sunny	Hot	High	Strong	No
3	Overcast	Hot	High	Weak	Yes
4	Rain	Mild	High	Weak	Yes
5	Rain	Cool	Normal	Weak	Yes
6	Rain	Cool	Normal	Strong	No
7	Overcast	Cool	Normal	Strong	Yes
8	Sunny	Mild	High	Weak	No
9	Sunny	Cool	Normal	Weak	Yes
10	Rain	Mild	Normal	Weak	Yes
11	Sunny	Mild	Normal	Strong	Yes
12	Overcast	Mild	High	Strong	Yes
13	Overcast	Hot	Normal	Weak	Yes
14	Rain	Mild	High	Strong	No
15	Sunny	Mild	High	Weak	No
16	Sunny	Mild	High	Weak	Yes
17	Sunny	Mild	High	Strong	Yes

No features left

Complete the branch of the tree where

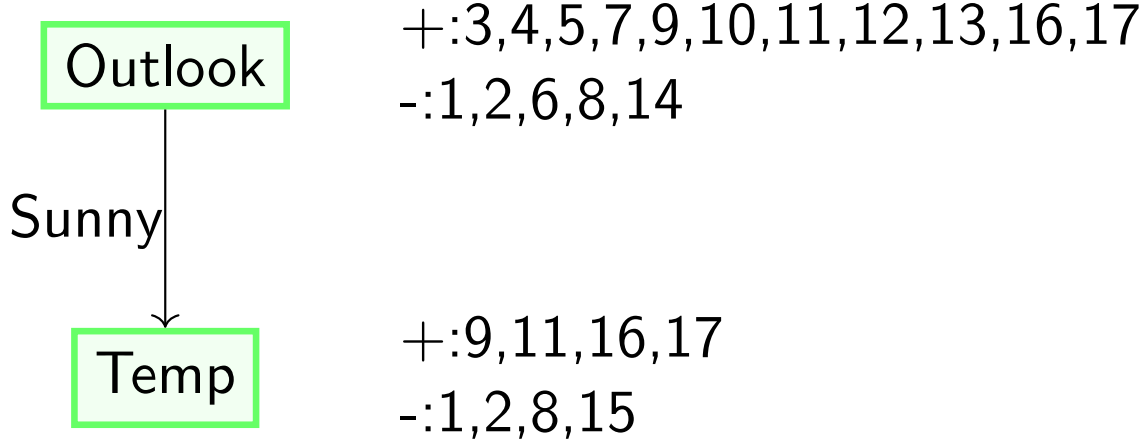
Outlook = Sunny, Temp = Mild, Humidity = High

No features left

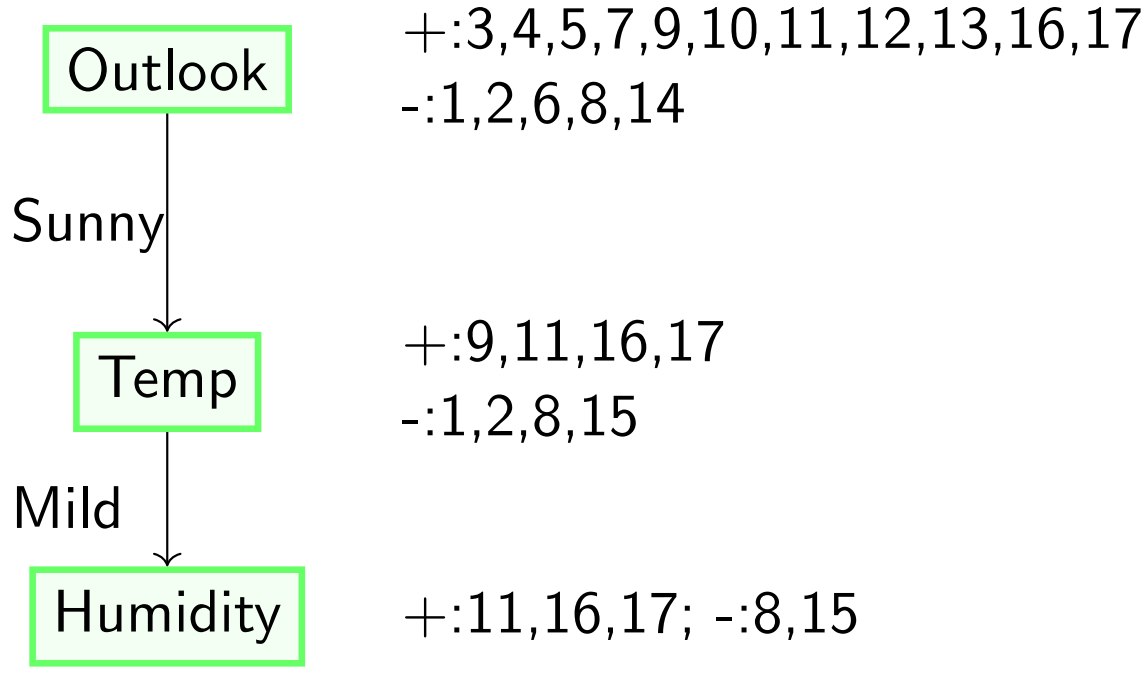
Outlook

+:3,4,5,7,9,10,11,12,13,16,17
-:1,2,6,8,14

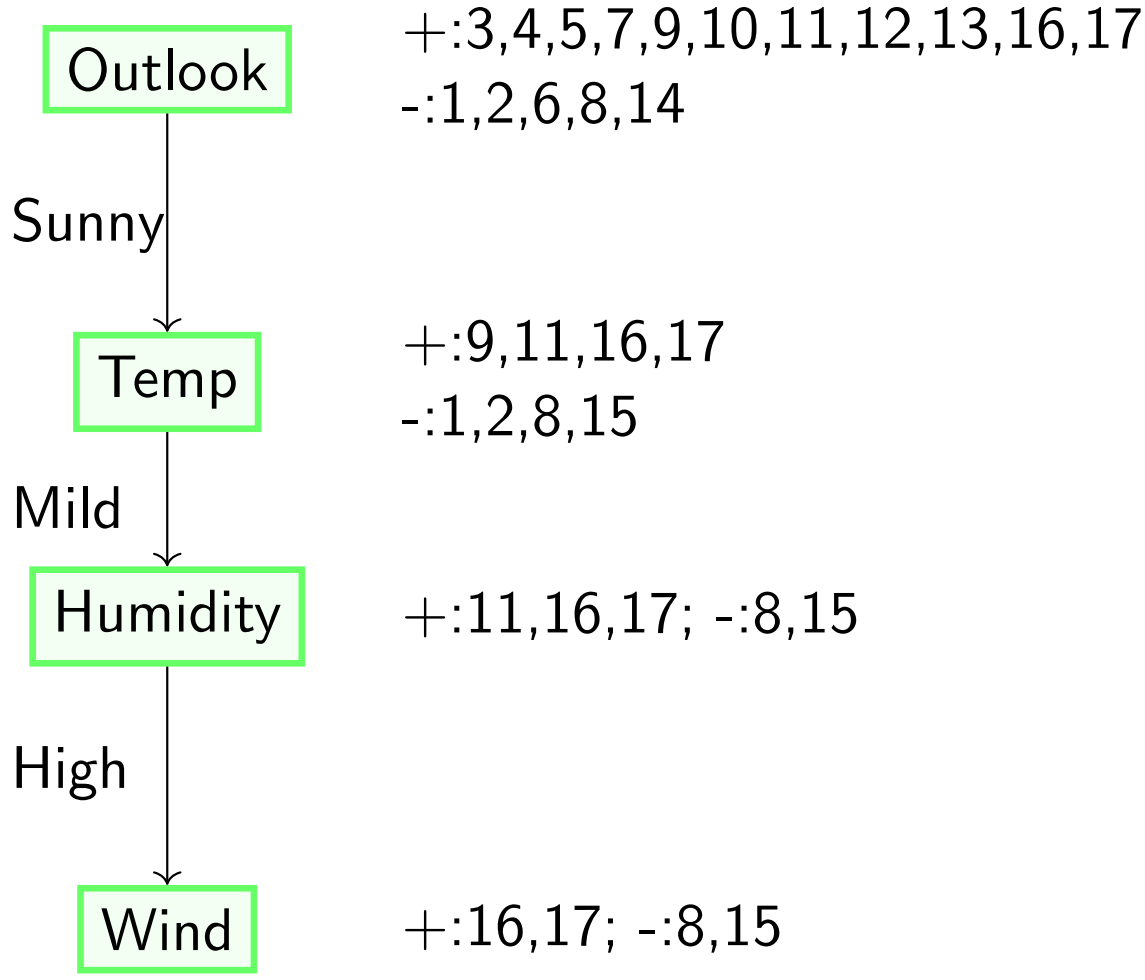
No features left



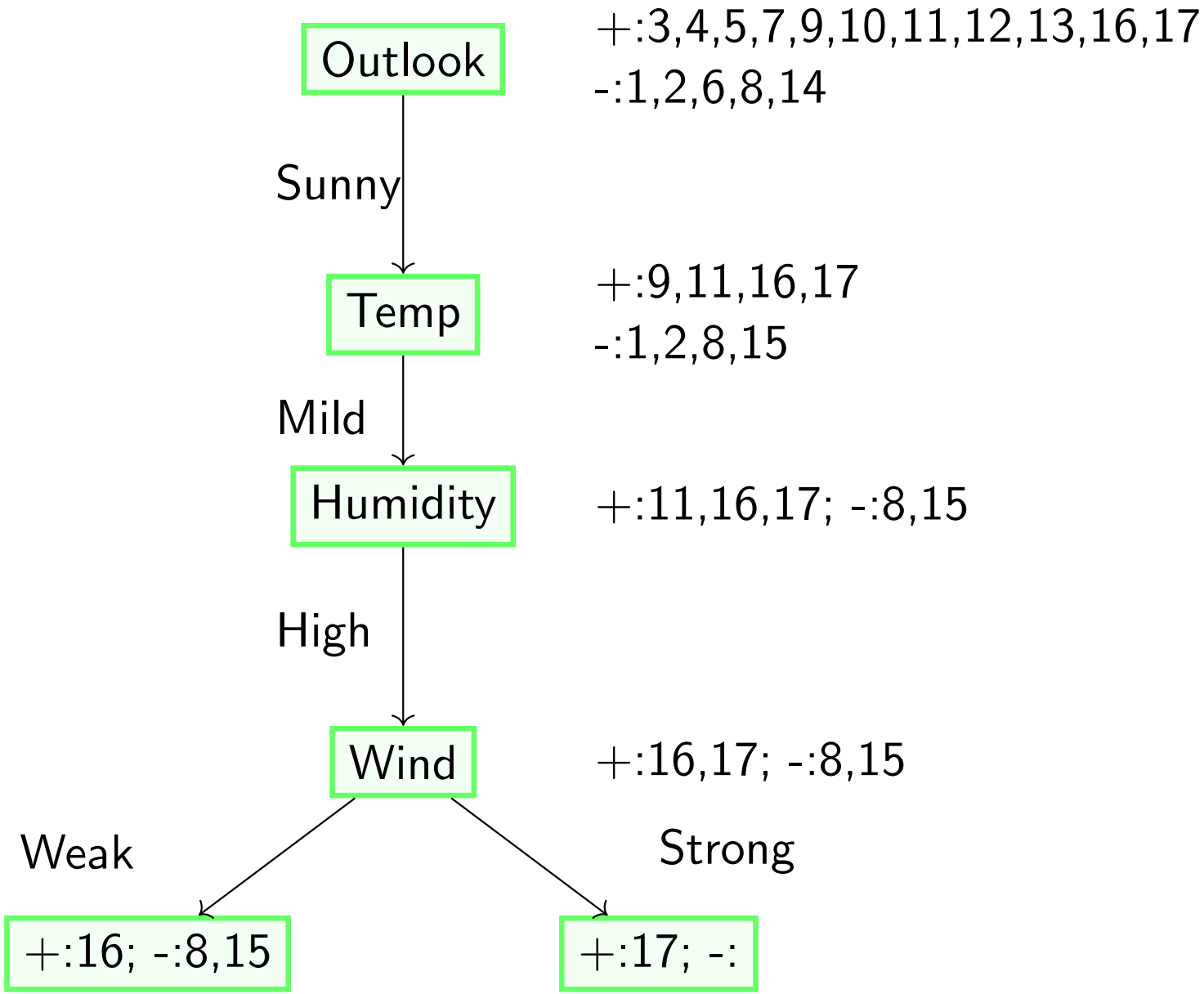
No features left



No features left



No features left



No features left

Why do we have this special case?

$\rightarrow$ The data is noisy. Even if we know the values of all the features, we still cannot make a deterministic decision. Perhaps, the outcome is influenced by a feature that we don't observe.

What are some possible ways of handling this case?

$\rightarrow$

- Predict the majority class.
- Predict the labels probabilistically.

No examples left

Consider the revised training set below.

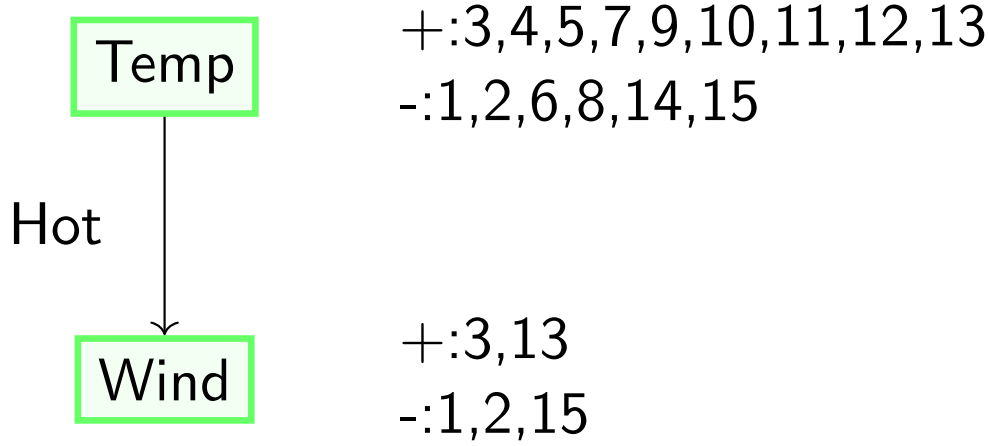
Day	Outlook	Temp	Humidity	Wind	Tennis?
1	Sunny	Hot	High	Weak	No
2	Sunny	Hot	High	Strong	No
3	Overcast	Hot	High	Weak	Yes
4	Rain	Mild	High	Weak	Yes
5	Rain	Cool	Normal	Weak	Yes
6	Rain	Cool	Normal	Strong	No
7	Overcast	Cool	Normal	Strong	Yes
8	Sunny	Mild	High	Weak	No
9	Sunny	Cool	Normal	Weak	Yes
10	Rain	Mild	Normal	Weak	Yes
11	Sunny	Mild	Normal	Strong	Yes
12	Overcast	Mild	High	Strong	Yes
13	Overcast	Hot	Normal	Weak	Yes
14	Rain	Mild	High	Strong	No
15	Sunny	Hot	High	Weak	No

No examples left

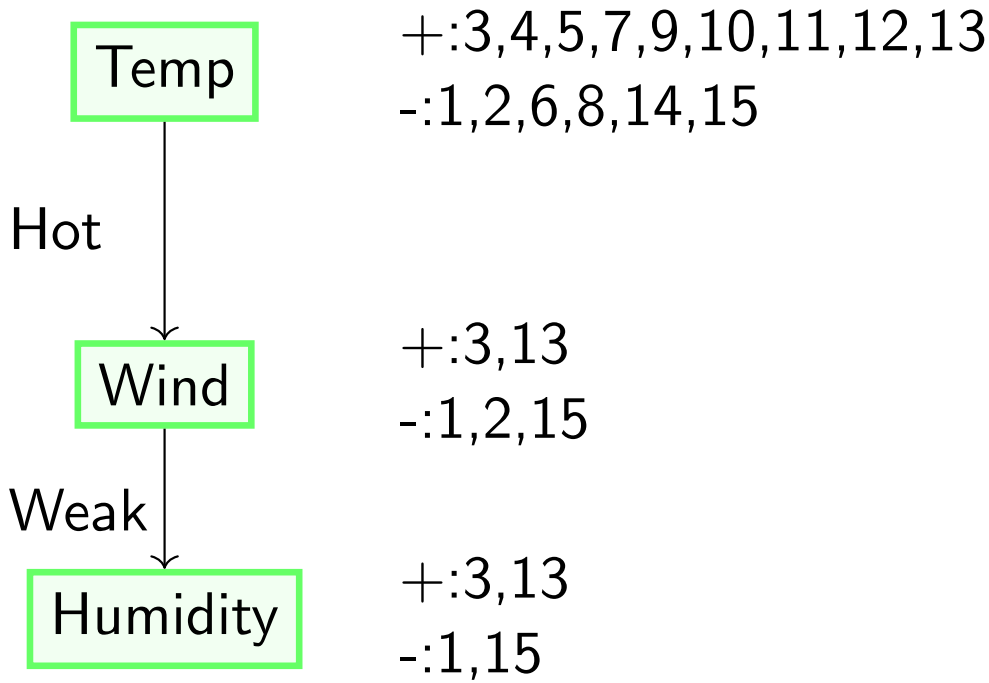
Complete the branch of the tree where

Temp = Hot, Wind = Weak, and Humidity = High.

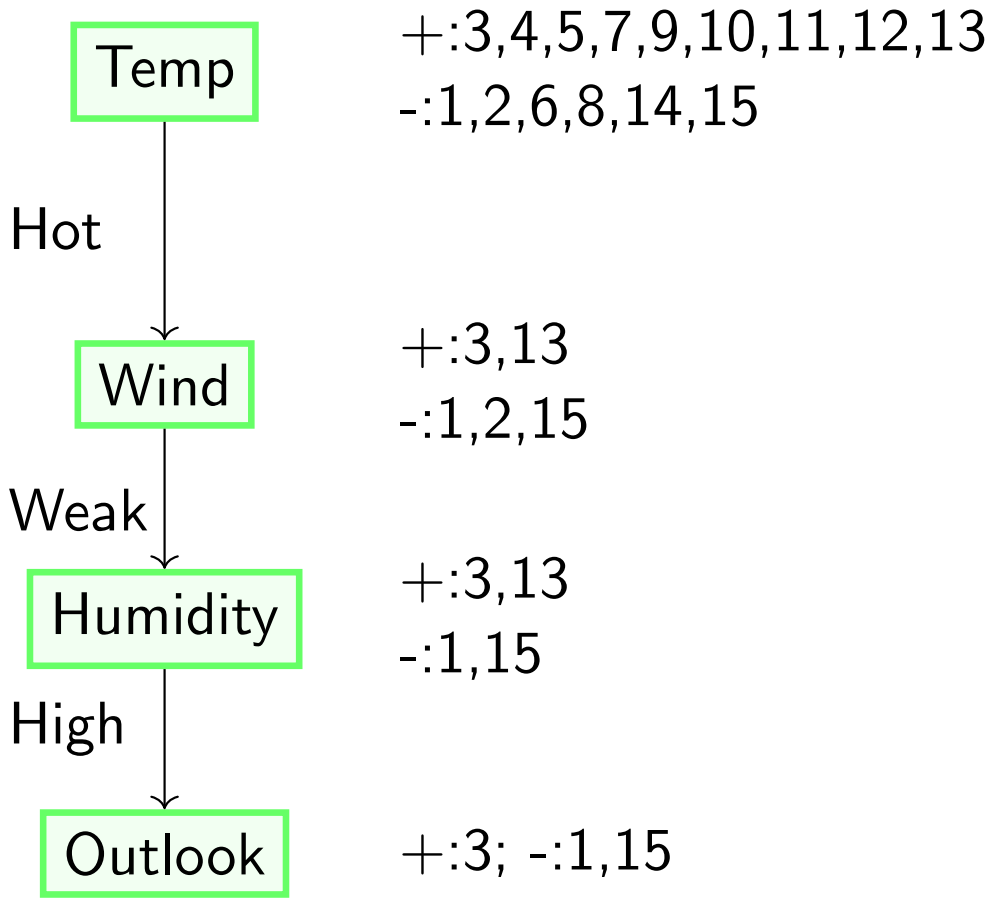
No examples left



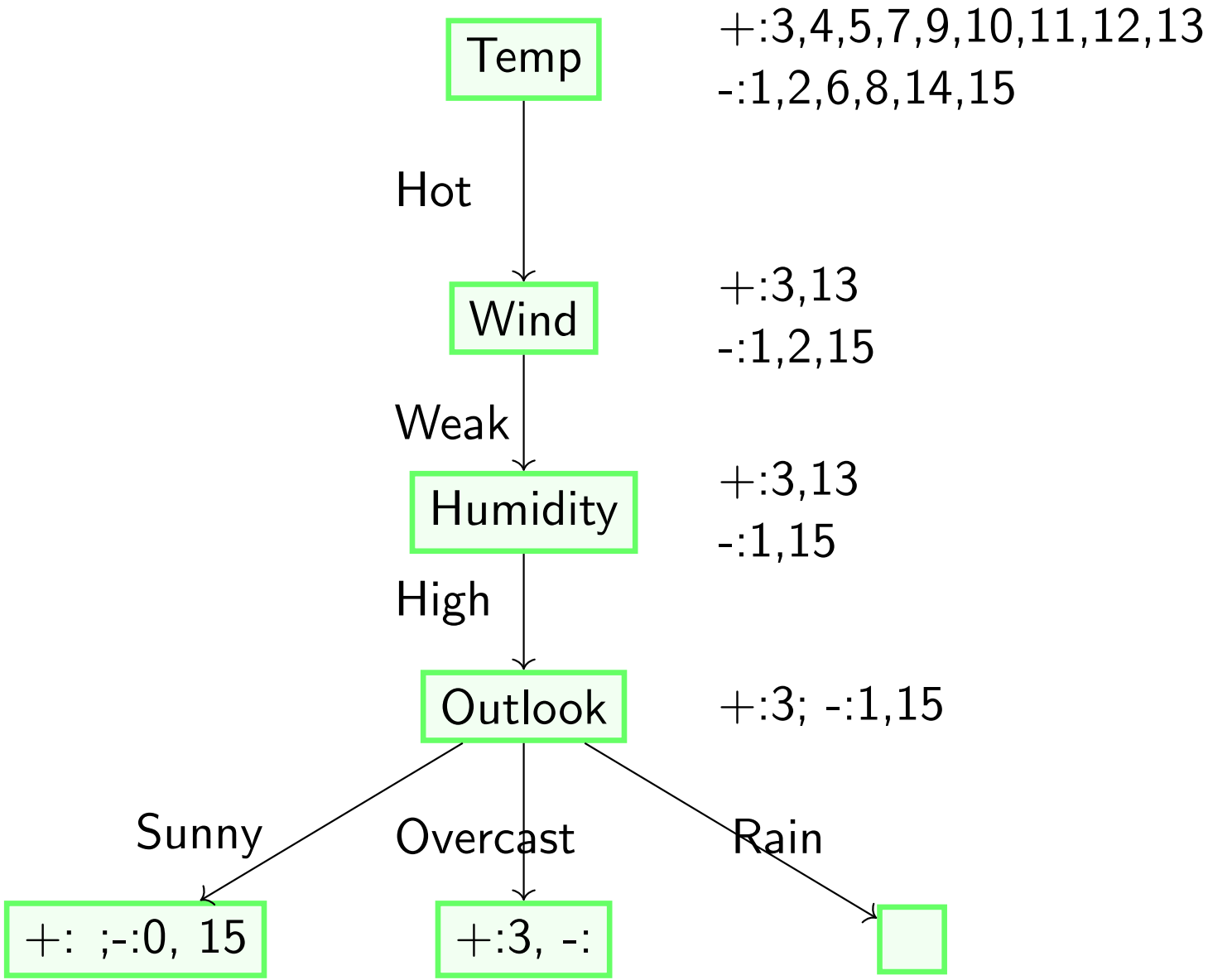
No examples left



No examples left



No examples left



No examples left

Why do we have this special case?

$\rightarrow$ A combination of features is not present in the training set. If we never observe this combination, we don't know how to predict it.

What are some possible ways of handling this case?

$\rightarrow$ Use the majority decision in the examples at the parent node.

Examples of Decision Trees

Definition and Classifying an Example

Grow a Full Tree

Decision Tree Learning Algorithm

Determine the Order of Testing Features

The Decision Tree Learner algorithm

Algorithm 1 Decision Tree Learner (examples, features)

```
1: if all examples are in the same class then
2:     return the class label.
3: else if no features left then
4:     return the majority decision.
5: else if no examples left then
6:     return the majority decision at the parent node.
7: else
8:     choose a feature f.
9:     for each value v of feature f do
10:        build edge with label v.
11:        build sub-tree using examples where the value of f is v.
```

Which feature should we test at each step?

Learning the "best" decision tree

- Each order of testing features leads to a different decision tree. Many decision trees can predict the training examples perfectly. Which tree do we prefer?

  $\rightarrow$ We prefer the smallest tree, since a simple model is more likely to predict unseen data well.

Learning the "best" decision tree

- Each order of testing features leads to a different decision tree. Many decision trees can predict the training examples perfectly. Which tree do we prefer?

  $\rightarrow$ We prefer the smallest tree, since a simple model is more likely to predict unseen data well.

- How many possible decision trees are there?

  How do we find the smallest decision tree?

  $\rightarrow$ A lot. Can represent all discrete functions of features. Cannot do systematic search. Intractable to find the optimal order of testing features.

  Do greedy search.

  Greedy search: make the myopic best decision at each step. (Best for that step, may not be best for minimizing the tree)

Examples of Decision Trees

Definition and Classifying an Example

Grow a Full Tree

Decision Tree Learning Algorithm

Determine the Order of Testing Features

Which feature should we test at each step?

- How can we create a shallow/small tree?
  Need to minimize the number of tests.

- Finding the optimal order of testing features is difficult.

  Greedy search — make the best myopic choice at each step.

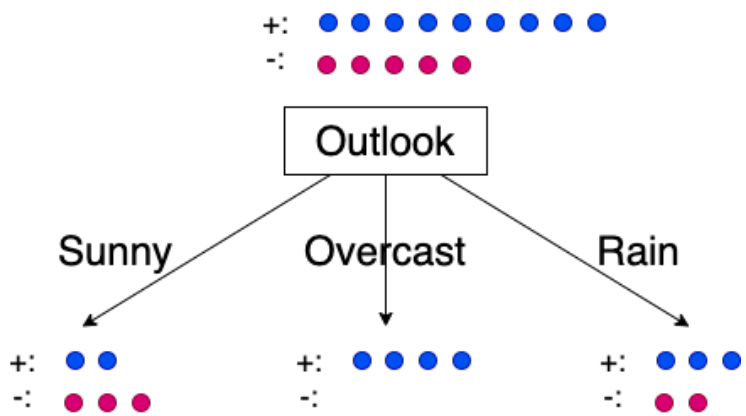
- At each step, test a feature that
  makes the biggest difference to the classification.

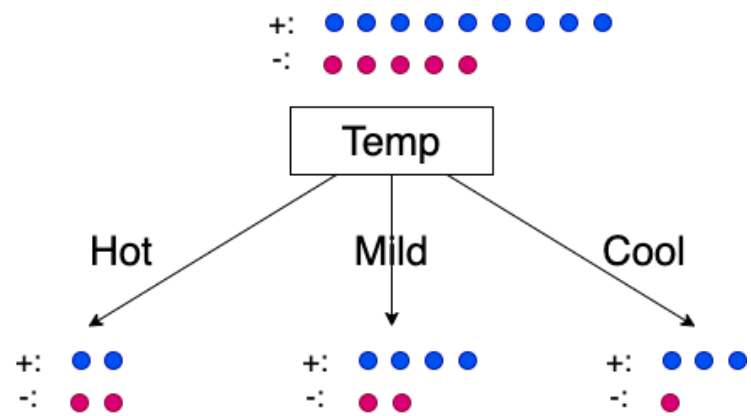
Which feature should we test?

Which of the two features below should we test?

(A) Outlook

(B) Temp



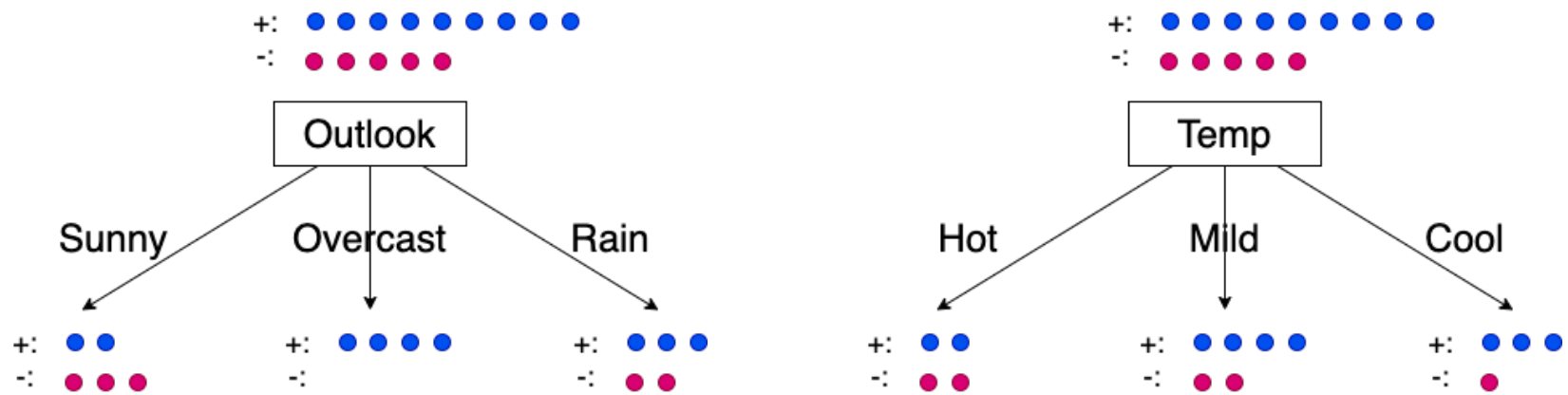


Which feature should we test?

Which of the two features below should we test?

(A) Outlook

(B) Temp



→ (A) We should test Outlook since it reduces uncertainty more than Temp.

Identifying the most important feature

- We want to make a decision as soon as possible
  $\rightarrow$ reduce uncertainty as much as possible
- Information content of a feature
  = uncertainty before testing the feature $-$ uncertainty after testing the feature
- Measure uncertainty using the notion of entropy.

Given a distribution $P(c_1), ..., P(c_k)$ over k outcomes $c_1, ..., c_k$, the **entropy** of the distribution is

$$I(P(c_1), ..., P(c_k)) = -\sum_{i=1}^{k} P(c_i) \log_2(P(c_i))$$

$\rightarrow$ Base 2 since we measure info content in bits.

Q: Entropy of a distribution over two outcomes

Q #1: What is the entropy of the distribution $(0.5, 0.5)$?

$$I(P(c_1), ..., P(c_k)) = -\sum_{i=1}^{k} P(c_i) \log_2(P(c_i))$$

(A) 0.2

(B) 0.4

(C) 0.6

(D) 0.8

(E) 1

Q: Entropy of a distribution over two outcomes

Q #1: What is the entropy of the distribution $(0.5, 0.5)$?

$$I(P(c_1), ..., P(c_k)) = -\sum_{i=1}^{k} P(c_i) \log_2(P(c_i))$$

(A) 0.2

(B) 0.4

(C) 0.6

(D) 0.8

(E) 1

$\rightarrow I(0.5, 0.5) = -\frac{1}{2} log_2(1/2) - \frac{1}{2} log_2(1/2) = 1.$

There is one bit of uncertainty in this distribution.
There is a lot of uncertainty in this distribution.

Q: Entropy of a distribution over two outcomes

Q #2: What is the entropy of the distribution $(0.01, 0.99)$?

$$I(P(c_1), ..., P(c_k)) = -\sum_{i=1}^{k} P(c_i) \log_2(P(c_i))$$

(A) 0.02

(B) 0.04

(C) 0.06

(D) 0.08

(E) 0.1

Q: Entropy of a distribution over two outcomes

Q #2: What is the entropy of the distribution $(0.01, 0.99)$?

$$I(P(c_1), ..., P(c_k)) = -\sum_{i=1}^{k} P(c_i) \log_2(P(c_i))$$

(A) 0.02

(B) 0.04

(C) 0.06

(D) 0.08

(E) 0.1

$\rightarrow I(0.99, 0.01) = -0.99 log_2(0.99) - 0.01 log_2(0.01) = 0.08.$

There is 0.08 bits of uncertainty in this distribution.

There is very little uncertainty in this distribution. We almost know for sure that the outcome will be the second one.

Entropy of a distribution over two outcomes

Consider a distribution $(p, 1-p)$ where $0 \leq p \leq 1$.

- What is the maximum entropy of this distribution?

  $\rightarrow$ maximum is at $p = 1/2$.

- What is the minimum entropy of this distribution?

  $\rightarrow$ minimum is at $p = 0$ and $p = 1$.

  By definition $I(1, 0) = 0$ and $I(0, 1) = 0$.

- Plot the entropy of the distribution $(p, 1-p)$ with respect to p.

Expected information gain of testing a feature

- The feature has k values $v_1, ..., v_k$.
- Before testing the feature,
  we have p positive and n negative examples.
- After testing the feature,
  for each value v_i of the feature,
  we have p_i positive and n_i negative examples.

Let's calculate the expected information gain of testing this feature.

Expected information gain of testing a feature

Entropy before testing the feature:

$\rightarrow$

$$H_{before} = I\left(\frac{p}{p+n}, \frac{n}{p+n}\right)$$

Expected entropy after testing the feature:

$\rightarrow$

$$H_{after} = \sum_{i=1}^{k} \frac{p_i + n_i}{p+n} * I\left(\frac{p_i}{p_i+n_i}, \frac{n_i}{p_i+n_i}\right)$$

Expected information gain (entropy reduction) is

$\rightarrow$

$$InformationGain = H_{before} - H_{after}$$

$\rightarrow$ can be generalized to more than two classes as well.

Q: Entropy and Information Gain

Q #3: What is the entropy of the examples before we select a feature for the root node of the tree?

$$I(P(c_1), ..., P(c_k)) = -\sum_{i=1}^{k} P(c_i) \log_2(P(c_i))$$

$$H_{before} = I\left(\frac{p}{p+n}, \frac{n}{p+n}\right)$$

(A) 0.54

(B) 0.64

(C) 0.74

(D) 0.84

(E) 0.94

Day	Outlook	Temp	Humidity	Wind	Tennis?
1	Sunny	Hot	High	Weak	No
2	Sunny	Hot	High	Strong	No
3	Overcast	Hot	High	Weak	Yes
4	Rain	Mild	High	Weak	Yes
5	Rain	Cool	Normal	Weak	Yes
6	Rain	Cool	Normal	Strong	No
7	Overcast	Cool	Normal	Strong	Yes
8	Sunny	Mild	High	Weak	No
9	Sunny	Cool	Normal	Weak	Yes
10	Rain	Mild	Normal	Weak	Yes
11	Sunny	Mild	Normal	Strong	Yes
12	Overcast	Mild	High	Strong	Yes
13	Overcast	Hot	Normal	Weak	Yes
14	Rain	Mild	High	Strong	No

Q: Entropy and Information Gain

Q #3: What is the entropy of the examples before we select a feature for the root node of the tree?

$$I(P(c_1), ..., P(c_k)) = -\sum_{i=1}^{k} P(c_i) \log_2(P(c_i))$$

$$H_{before} = I\left(\frac{p}{p+n}, \frac{n}{p+n}\right)$$

(A) 0.54

(B) 0.64

(C) 0.74

(D) 0.84

(E) 0.94

Day	Outlook	Temp	Humidity	Wind	Tennis?
1	Sunny	Hot	High	Weak	No
2	Sunny	Hot	High	Strong	No
3	Overcast	Hot	High	Weak	Yes
4	Rain	Mild	High	Weak	Yes
5	Rain	Cool	Normal	Weak	Yes
6	Rain	Cool	Normal	Strong	No
7	Overcast	Cool	Normal	Strong	Yes
8	Sunny	Mild	High	Weak	No
9	Sunny	Cool	Normal	Weak	Yes
10	Rain	Mild	Normal	Weak	Yes
11	Sunny	Mild	Normal	Strong	Yes
12	Overcast	Mild	High	Strong	Yes
13	Overcast	Hot	Normal	Weak	Yes
14	Rain	Mild	High	Strong	No

$\rightarrow$ Correct answer is 0.940 (E).

Q: Entropy and Information Gain

Q #4: What is the expected information gain if we select **Outlook** as the root node of the tree?

$$I(P(c_1), ..., P(c_k)) = -\sum_{i=1}^{k} P(c_i) \log_2(P(c_i))$$

$$H_{before} = I\left(\frac{p}{p+n}, \frac{n}{p+n}\right)$$

$$H_{after} = \sum_{i=1}^{k} \frac{p_i+n_i}{p+n} * I\left(\frac{p_i}{p_i+n_i}, \frac{n_i}{p_i+n_i}\right)$$

(A) 0.237

(B) 0.247

(C) 0.257

(D) 0.267

(E) 0.277

Day	Outlook	Temp	Humidity	Wind	Tennis?
1	Sunny	Hot	High	Weak	No
2	Sunny	Hot	High	Strong	No
3	Overcast	Hot	High	Weak	Yes
4	Rain	Mild	High	Weak	Yes
5	Rain	Cool	Normal	Weak	Yes
6	Rain	Cool	Normal	Strong	No
7	Overcast	Cool	Normal	Strong	Yes
8	Sunny	Mild	High	Weak	No
9	Sunny	Cool	Normal	Weak	Yes
10	Rain	Mild	Normal	Weak	Yes
11	Sunny	Mild	Normal	Strong	Yes
12	Overcast	Mild	High	Strong	Yes
13	Overcast	Hot	Normal	Weak	Yes
14	Rain	Mild	High	Strong	No

Q: Entropy and Information Gain

Q #4: What is the expected information gain if we select **Outlook** as the root node of the tree?

$$I(P(c_1), ..., P(c_k)) = -\sum_{i=1}^{k} P(c_i) \log_2(P(c_i))$$

$$H_{before} = I\left(\frac{p}{p+n}, \frac{n}{p+n}\right)$$

$$H_{after} = \sum_{i=1}^{k} \frac{p_i+n_i}{p+n} * I\left(\frac{p_i}{p_i+n_i}, \frac{n_i}{p_i+n_i}\right)$$

(A) 0.237

(B) 0.247

(C) 0.257

(D) 0.267

(E) 0.277

$\rightarrow$ Correct answer is 0.247 (B).

Day	Outlook	Temp	Humidity	Wind	Tennis?
1	Sunny	Hot	High	Weak	No
2	Sunny	Hot	High	Strong	No
3	Overcast	Hot	High	Weak	Yes
4	Rain	Mild	High	Weak	Yes
5	Rain	Cool	Normal	Weak	Yes
6	Rain	Cool	Normal	Strong	No
7	Overcast	Cool	Normal	Strong	Yes
8	Sunny	Mild	High	Weak	No
9	Sunny	Cool	Normal	Weak	Yes
10	Rain	Mild	Normal	Weak	Yes
11	Sunny	Mild	Normal	Strong	Yes
12	Overcast	Mild	High	Strong	Yes
13	Overcast	Hot	Normal	Weak	Yes
14	Rain	Mild	High	Strong	No

Q: Entropy and Information Gain

Q #5: What is the expected information gain if we select **Humidity** as the root node of the tree?

$$I(P(c_1), ..., P(c_k)) = -\sum_{i=1}^{k} P(c_i) \log_2(P(c_i))$$

$$H_{before} = I\left(\frac{p}{p+n}, \frac{n}{p+n}\right)$$

$$H_{after} = \sum_{i=1}^{k} \frac{p_i+n_i}{p+n} * I\left(\frac{p_i}{p_i+n_i}, \frac{n_i}{p_i+n_i}\right)$$

(A) 0.151

(B) 0.251

(C) 0.351

(D) 0.451

(E) 0.551

Day	Outlook	Temp	Humidity	Wind	Tennis?
1	Sunny	Hot	High	Weak	No
2	Sunny	Hot	High	Strong	No
3	Overcast	Hot	High	Weak	Yes
4	Rain	Mild	High	Weak	Yes
5	Rain	Cool	Normal	Weak	Yes
6	Rain	Cool	Normal	Strong	No
7	Overcast	Cool	Normal	Strong	Yes
8	Sunny	Mild	High	Weak	No
9	Sunny	Cool	Normal	Weak	Yes
10	Rain	Mild	Normal	Weak	Yes
11	Sunny	Mild	Normal	Strong	Yes
12	Overcast	Mild	High	Strong	Yes
13	Overcast	Hot	Normal	Weak	Yes
14	Rain	Mild	High	Strong	No

Q: Entropy and Information Gain

Q #5: What is the expected information gain if we select **Humidity** as the root node of the tree?

$$I(P(c_1), ..., P(c_k)) = -\sum_{i=1}^{k} P(c_i) \log_2(P(c_i))$$

$$H_{before} = I\left(\frac{p}{p+n}, \frac{n}{p+n}\right)$$

$$H_{after} = \sum_{i=1}^{k} \frac{p_i+n_i}{p+n} * I\left(\frac{p_i}{p_i+n_i}, \frac{n_i}{p_i+n_i}\right)$$

(A) 0.151

(B) 0.251

(C) 0.351

(D) 0.451

(E) 0.551

Day	Outlook	Temp	Humidity	Wind	Tennis?
1	Sunny	Hot	High	Weak	No
2	Sunny	Hot	High	Strong	No
3	Overcast	Hot	High	Weak	Yes
4	Rain	Mild	High	Weak	Yes
5	Rain	Cool	Normal	Weak	Yes
6	Rain	Cool	Normal	Strong	No
7	Overcast	Cool	Normal	Strong	Yes
8	Sunny	Mild	High	Weak	No
9	Sunny	Cool	Normal	Weak	Yes
10	Rain	Mild	Normal	Weak	Yes
11	Sunny	Mild	Normal	Strong	Yes
12	Overcast	Mild	High	Strong	Yes
13	Overcast	Hot	Normal	Weak	Yes
14	Rain	Mild	High	Strong	No

→ Correct answer is 0.151 (A).

Q: Which feature should we pick to build the tree?

Q #6: Which feature should we use to build the tree? If Humidity could decrease the entropy to 0.48 and Outlook could decrease the entropy to 0.77.

(A) Using Humidity is better

(B) Using Outlook is better

(C) They are equivalent

Q: Which feature should we pick to build the tree?

Q #6: Which feature should we use to build the tree? If Humidity could decrease the entropy to 0.48 and Outlook could decrease the entropy to 0.77.

(A) Using Humidity is better

(B) Using Outlook is better

(C) They are equivalent

$\rightarrow$ Correct answer is Humidity is better (A).

ID3 Algorithm

- Calculate the entropy of every attribute a of the data set S
- Partition ("split") the set S into subsets using the attribute for which the resulting entropy after splitting is minimized; or, equivalently, information gain is maximum.
- Make a decision tree node containing that attribute.
- Recurse on subsets using the remaining attributes.

Revisiting Learning Goals

- Describe the components of a decision tree.
- Construct a decision tree given an order of testing the features.
- Determine the prediction accuracy of a decision tree on a test set.
- Compute the entropy of a probability distribution.
- Compute the expected information gain for selecting a feature.
- Trace the execution of and implement the algorithm for learning a decision tree.