Classification: Decision Trees

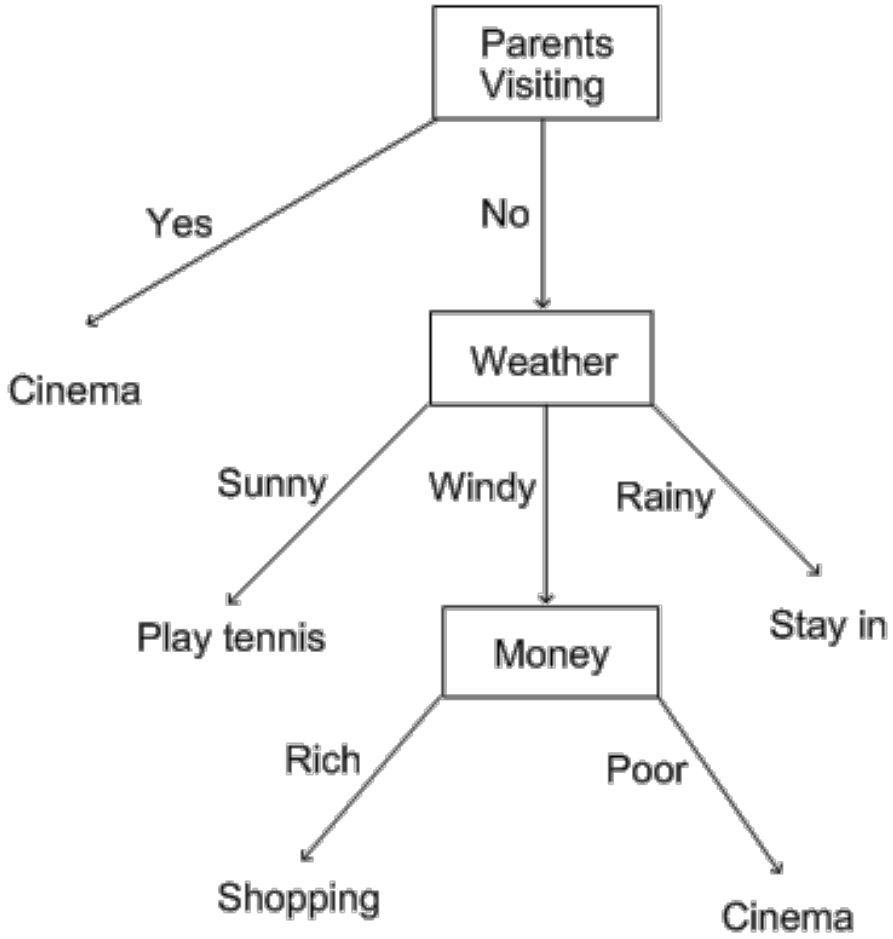
What is classification?

The task of determining / predicting the class that an instance belongs to; the assumptions:

- each instance is defined by a set of attributes;
- a set of possible classes is given

Decision trees

Example: Deciding what to do on a Sunday afternoon



Source: http://study.com/cimages/multimages/16/decision_tree.gif

ID3 algorithm

- ID3 - Iterative Dichotomiser 3
- One of the well-known algorithms for generating decision trees based on a given set of examples (dataset)
- The resulting tree allows for classifying future (unknown) instances

Example: Forecasting whether the play will be performed

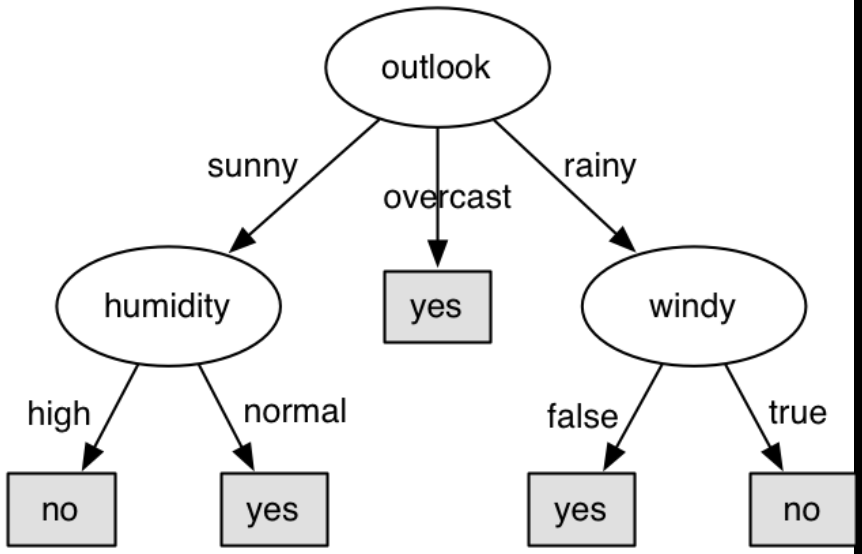
ToPlayOtNotToPlay.arff dataset

Outlook	Temp.	Humidity	Windy	Play
sunny	hot	high	false	no
sunny	hot	high	true	no
overcast	hot	high	false	yes
rainy	mild	high	false	yes
rainy	cool	normal	false	yes
rainy	cool	normal	true	no
overcast	cool	normal	true	yes
sunny	mild	high	false	no
sunny	cool	normal	false	yes
rainy	mild	normal	false	yes
sunny	mild	normal	true	yes
overcast	mild	high	true	yes
overcast	hot	normal	false	yes
rainy	mild	high	true	no

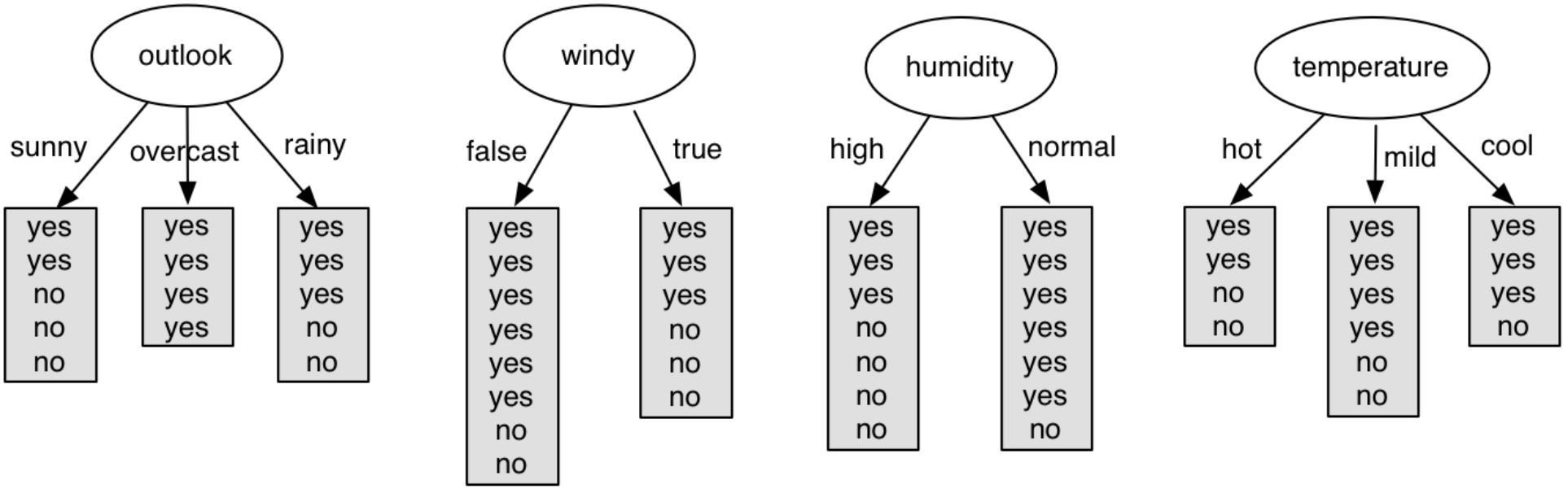
Top-down approach

Recursive divide-and-conquer:

- **Select** an attribute for the root node
  - create a branch for each of the attribute's values (assumption: attributes are nominal)
- **Split** instances into subsets
  - one for each branch extending from the node
- **Repeat** recursively for each branch
  - using only instances that reach the branch
- **Stop**
  - if all instances have the same class



Which attribute to select?



Which attribute to select?

- **Aim**: to get the smallest tree with good predictive power
- **How**: by relying on the notions of ***Entropy*** and ***Information Gain***
- **Entropy** quantifies the amount of uncertainty involved in the value of a random variable or the outcome of a random process
- **Information Gain** is the reduction in entropy (uncertainty) by learning (or obtaining data) about the random variable / process

Dataset entropy

In the context of a classification problem, entropy (H) is about the uncertainty associated with making predictions about the class membership of instances in the given dataset

$$H(S) = -\sum_{i=1}^{N} p_i \log_2 p_i$$

where:

- H(S) – entropy of the dataset S
- p_i – probability of the outcome (class value) i
- N – number of distinct outcomes (class values)

Dataset entropy

From the total of 14 instances in the example dataset (S) we have:

- 9 instances “yes”
- 5 instances “no”

$$H(S) = -\sum_{i=1}^{N} p_i \log_2 p_i$$

$$H(S) = -\frac{9}{14}\log_2 \frac{9}{14} - \frac{5}{14}\log_2 \frac{5}{14} = 0.940$$

Information gain

Information gain of an attribute A over the set of instances S *Gain(A, S)* represents

- the amount of information we would gain by knowing the value of the attribute A of instances in the set S
- the difference between the entropy before branching and the entropy after branching over the attribute A

Information gain

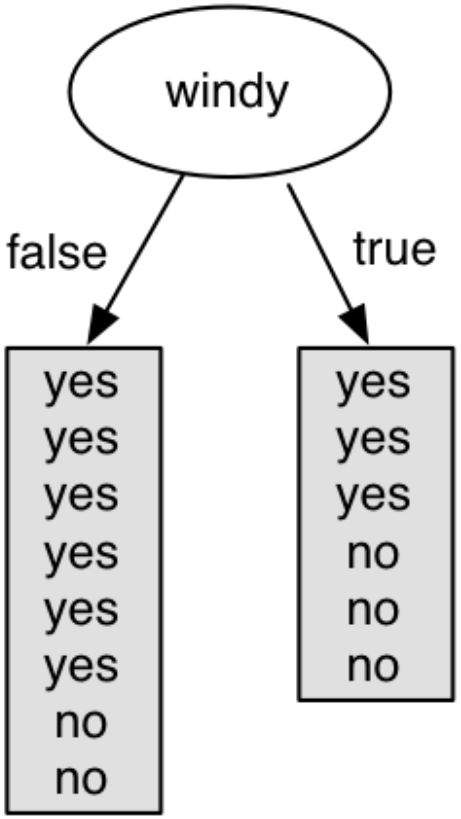
$$Gain(A,S) = H(S) - H(A,S) = H(S) - \sum_{j=1}^{v} \frac{|S_j|}{|S|} \cdot H(S_j)$$

where:

- H(S) – entropy of the whole dataset S
- H(A, S) – entropy of S when the attribute A is known
- $|S_j|$ – number of instance with j^{th} value of the attribute A
- |S| – total number of instances in dataset S
- v – number of distinct values of the attribute A
- $H(S_j)$ – entropy of the subset of instances with j^{th} value for the attribute A

Information gain of attribute "windy"

- From the total of 14 instances we have:
  - 8 instances "false"
  - 6 instances "true"

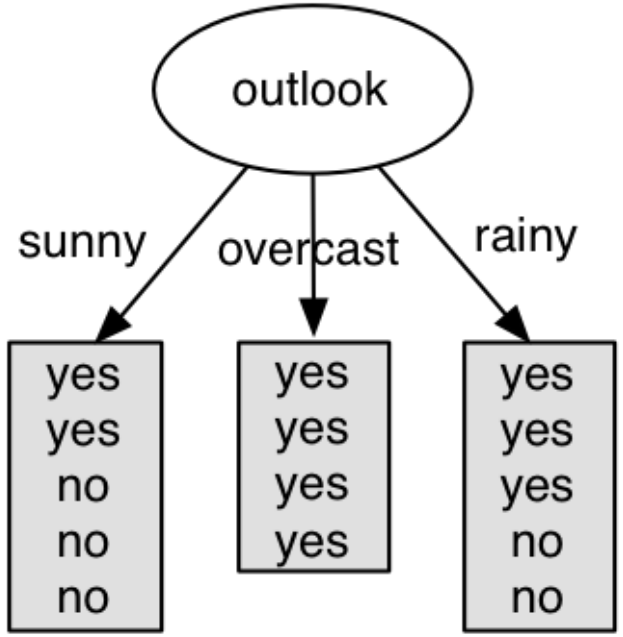


$$Gain(A,S) = H(S) - \sum_{j=1}^{v} \frac{|S_j|}{|S|} \cdot H(S_j)$$

$$Gain(A_{Windy}, S) = 0.940 - \frac{8}{14} \cdot \left(-\left(\frac{6}{8} \cdot \log_2 \frac{6}{8} + \frac{2}{8} \cdot \log_2 \frac{2}{8} \right) \right) - \frac{6}{14} \cdot \left(-\left(\frac{3}{6} \cdot \log_2 \frac{3}{6} + \frac{3}{6} \cdot \log_2 \frac{3}{6} \right) \right) = 0.048$$

Information gain of attribute "outlook"

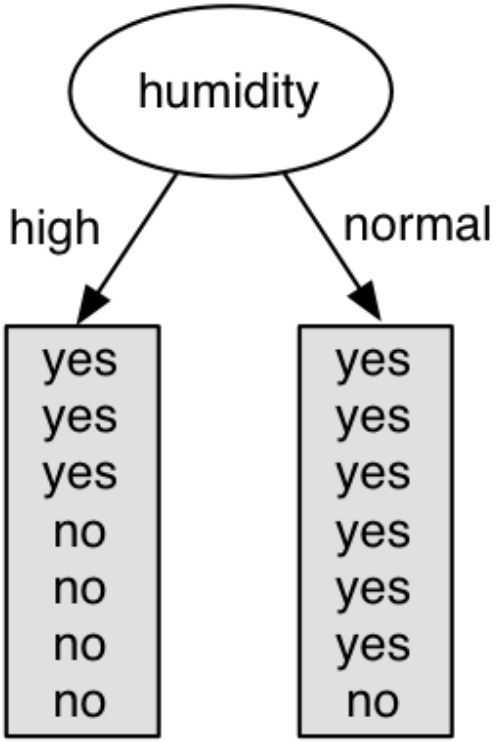
- From the total of 14 instances we have:
  - 5 instances "sunny"
  - 4 instances "overcast"
  - 5 instances "rainy"



$$Gain(A_{Outlook}, S) = 0.940 - \frac{5}{14} \cdot \left(-\left(\frac{2}{5} \log_2 \frac{2}{5} + \frac{3}{5} \log_2 \frac{3}{5} \right) \right) - \frac{4}{14} \cdot \left(-\left(\frac{4}{4} \log_2 \frac{4}{4} \right) \right) - \frac{5}{14} \cdot \left(-\left(\frac{3}{5} \cdot \log_2 \frac{3}{5} + \frac{2}{5} \cdot \log_2 \frac{2}{5} \right) \right) = 0.247$$

Information gain of attribute "humidity"

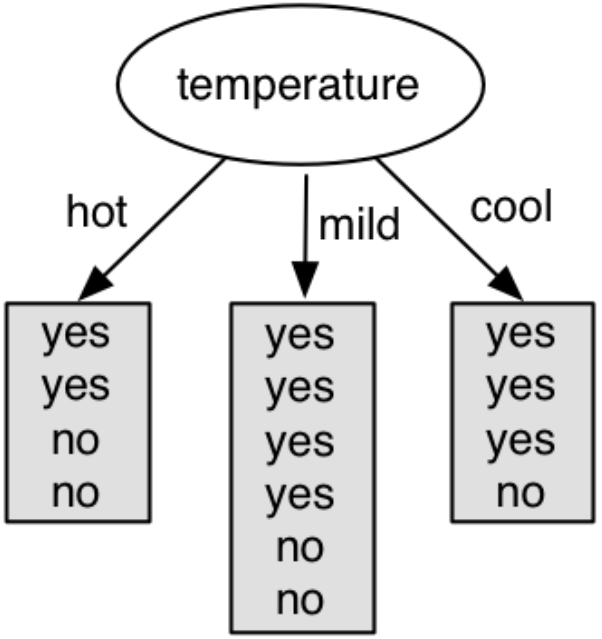
- From the total of 14 instances we have:
  - 7 instances "high"
  - 7 instances "normal"



$$Gain(A_{Humidity}, S) = 0.940 - \frac{7}{14} \cdot \left(-\left(\frac{3}{7} \cdot \log_2 \frac{3}{7} + \frac{4}{7} \cdot \log_2 \frac{4}{7}\right)\right) - \frac{7}{14} \cdot \left(-\left(\frac{6}{7} \cdot \log_2 \frac{6}{7} + \frac{1}{7} \cdot \log_2 \frac{1}{7}\right)\right) = 0.151$$

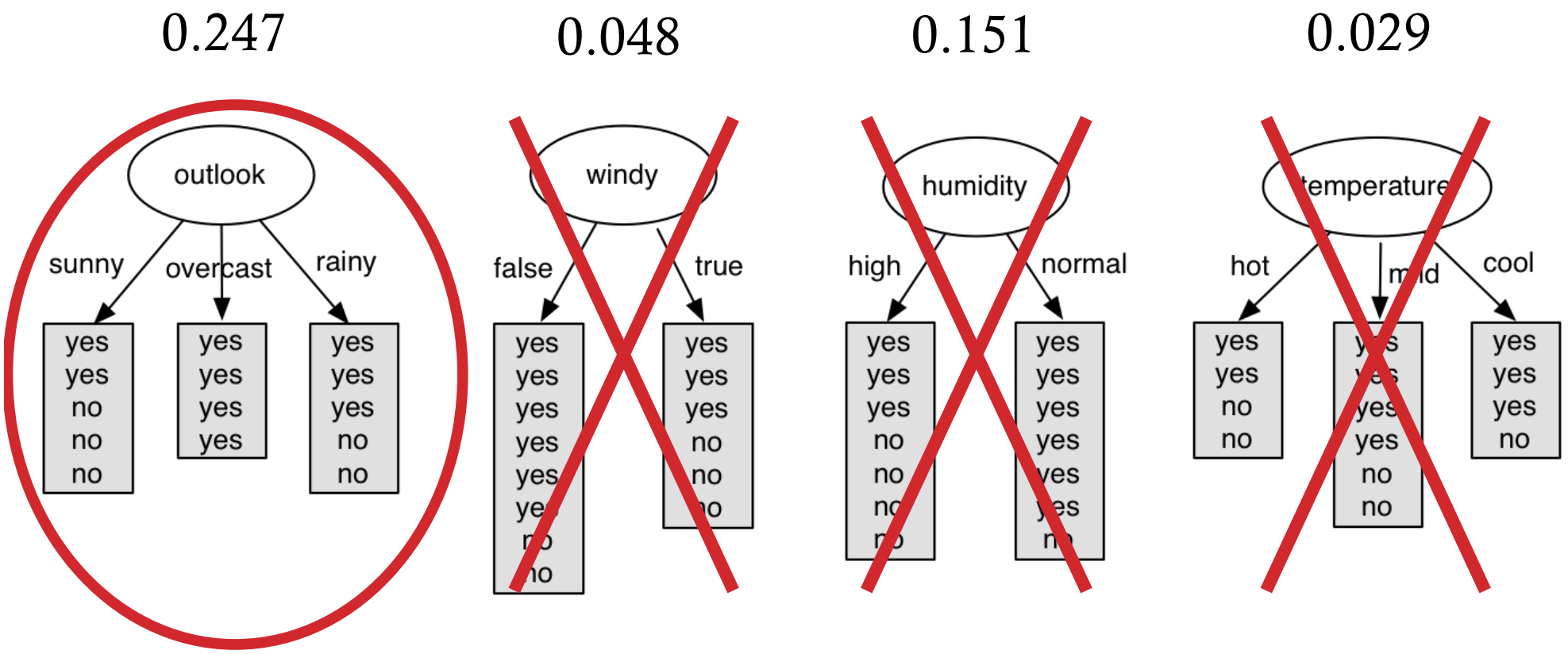
Information gain of attribute "temperature"

- From the total of 14 instances we have:
  - 4 instances "hot"
  - 6 instances "mild"
  - 4 instances "cool"



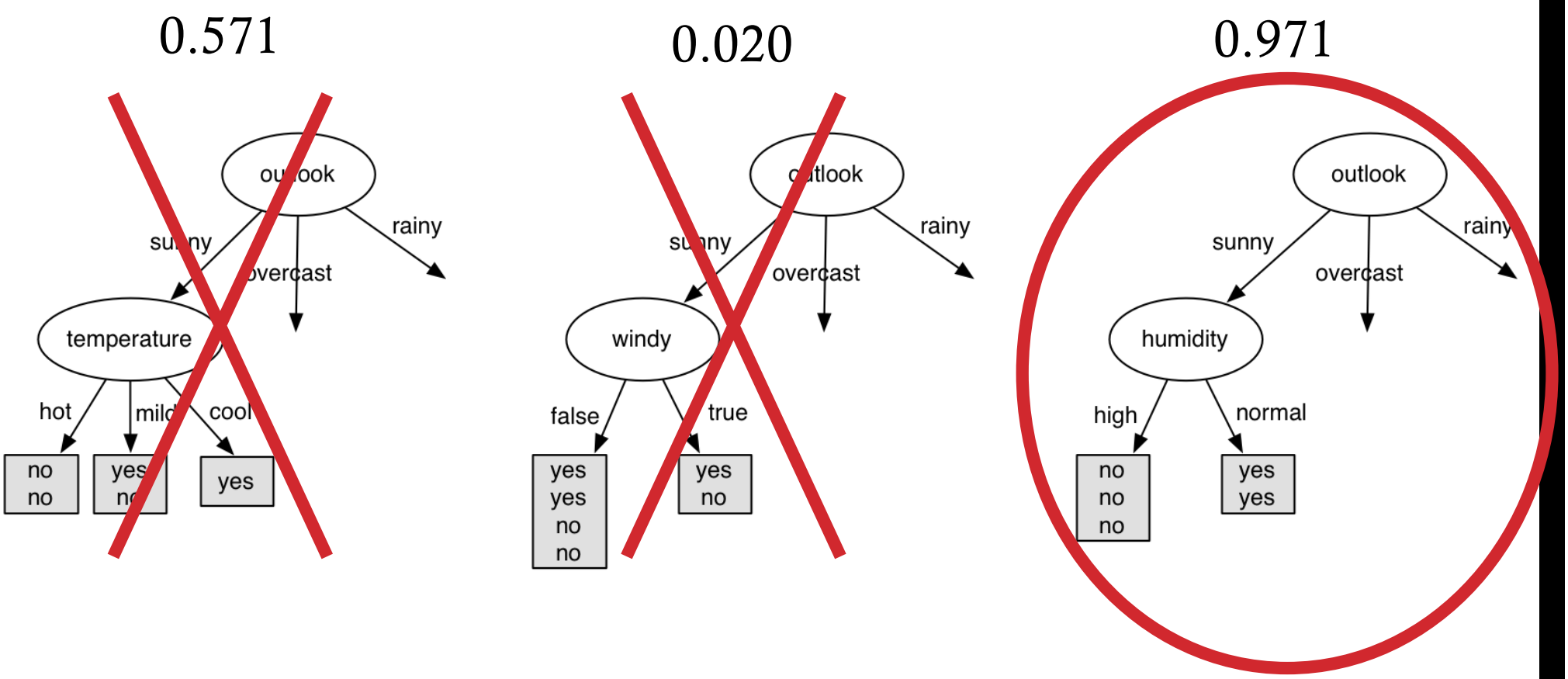
$$Gain(A_{Temperature}, S) = 0.940 -$$
$$\frac{4}{14} \cdot \left(-\left(\frac{2}{4} \cdot \log_2 \frac{2}{4} + \frac{2}{4} \cdot \log_2 \frac{2}{4}\right)\right) -$$
$$\frac{6}{14} \cdot \left(-\left(\frac{4}{6} \cdot \log_2 \frac{4}{6} + \frac{2}{6} \cdot \log_2 \frac{2}{6}\right)\right) -$$
$$\frac{4}{14} \cdot \left(-\left(\frac{3}{4} \cdot \log_2 \frac{3}{4} + \frac{1}{4} \cdot \log_2 \frac{1}{4}\right)\right) = 0.029$$

Which attribute to select?

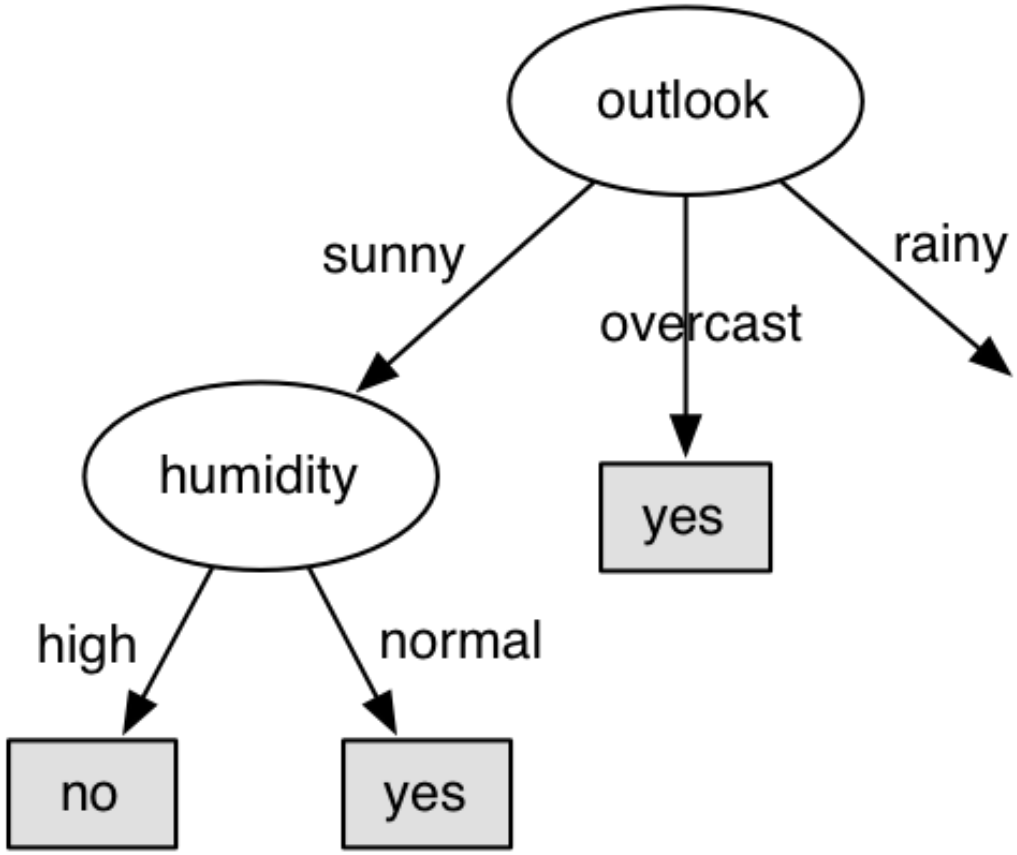


Choose the attribute with highest information gain

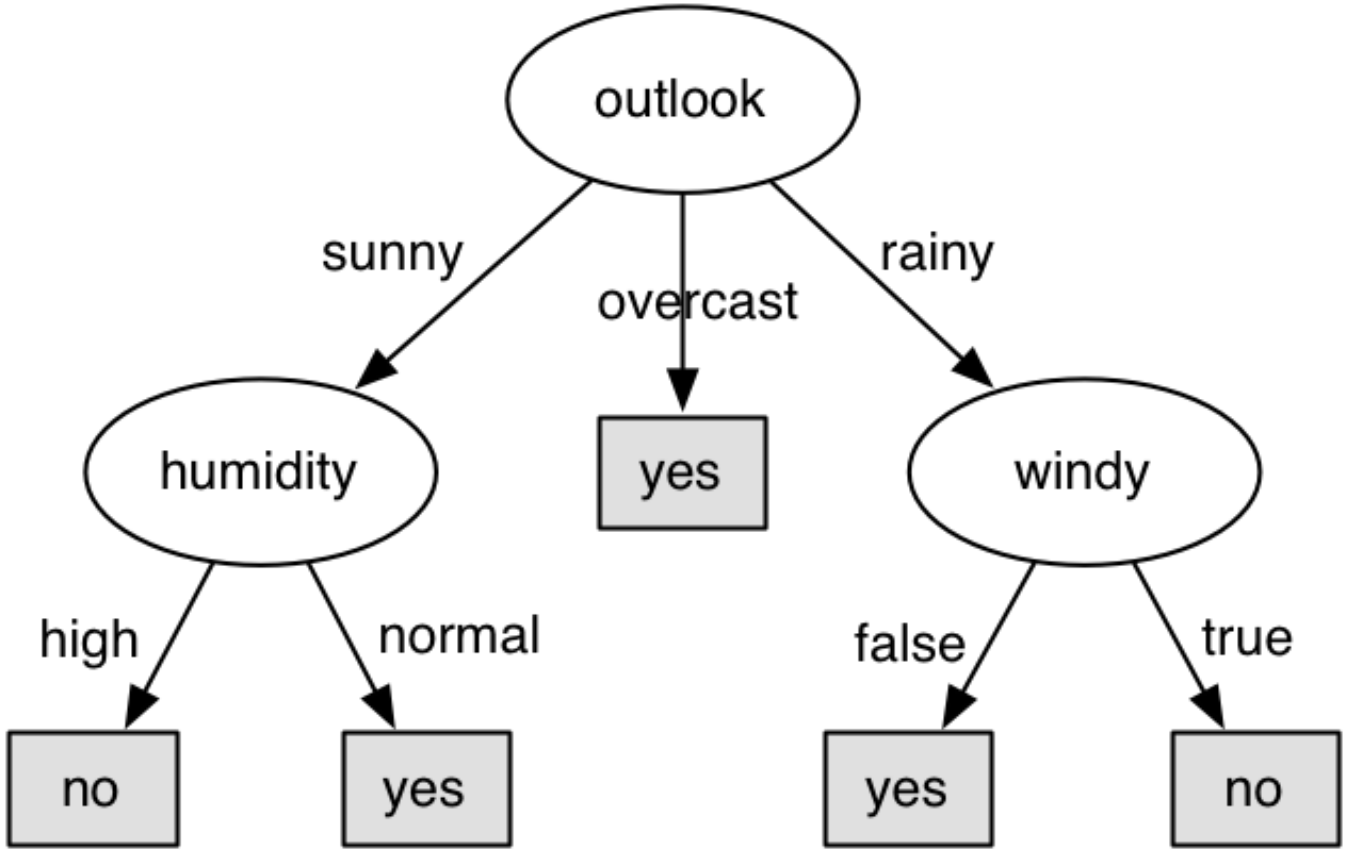
Iteration 2: Repeat recursively for each branch



Iteration 2: Repeat recursively for each branch



Iteration 2: Repeat recursively for each branch



Weka

- Software for data mining in Java
- Set of algorithms for machine learning and data mining
- Developed at the University of Waikato, New Zealand
- Open-source
- Website: http://www.cs.waikato.ac.nz/ml/weka

ARFF file

- Attribute-Relation File Format – ARFF
- Textual file

Attributes can be:
- Numerical
- Nominal

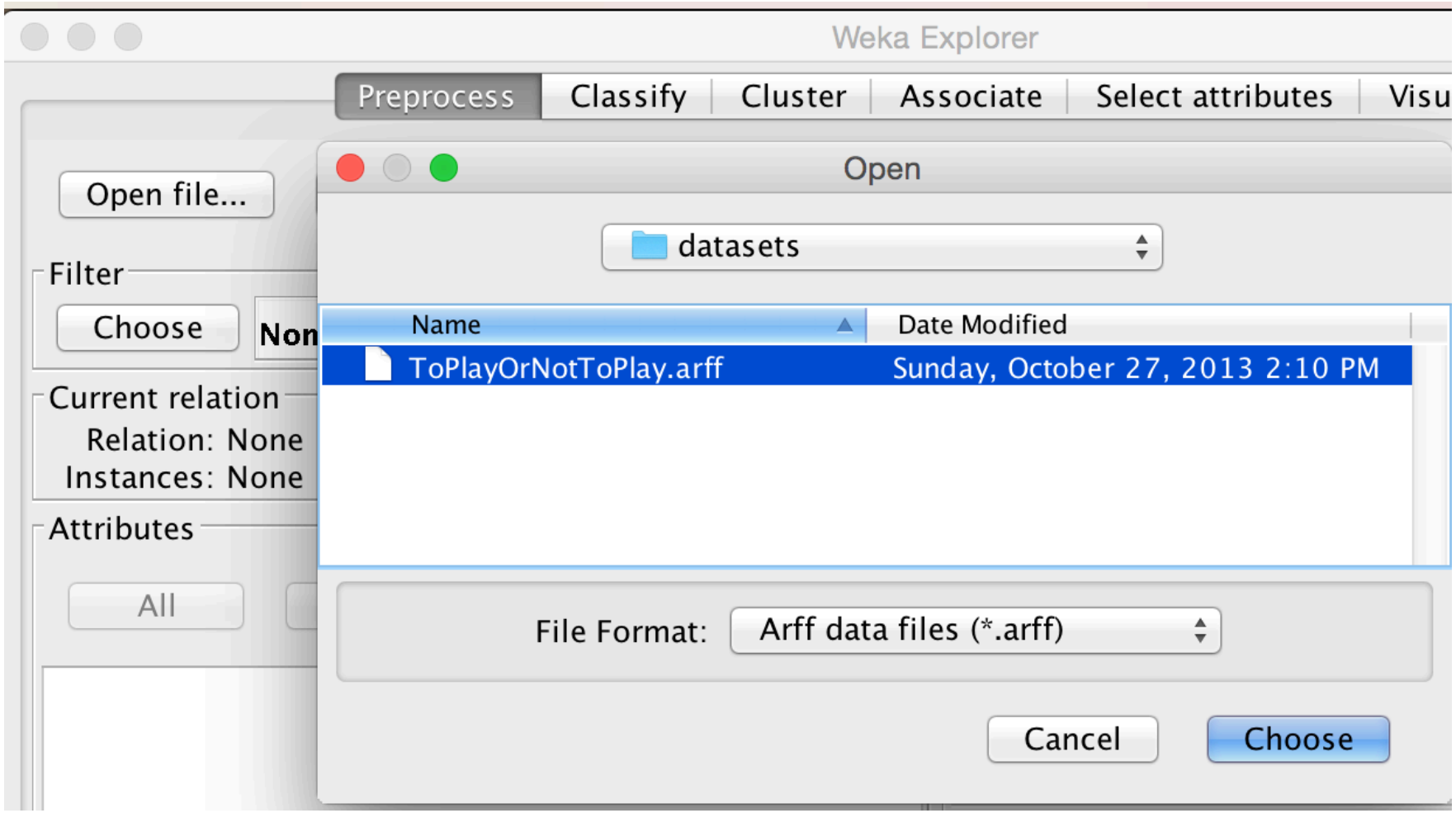
```
@relation TPONTPNom

@attribute Outlook {sunny, overcast, rainy}
@attribute Temp. {hot, mild, cool}
@attribute Humidity {high, normal}
@attribute Windy {false, true}
@attribute Play {no, yes}

@data
sunny, hot, high, false, no
sunny, hot, high, true, no
overcast, hot, high, false, yes
...
```

Datasets used for this class

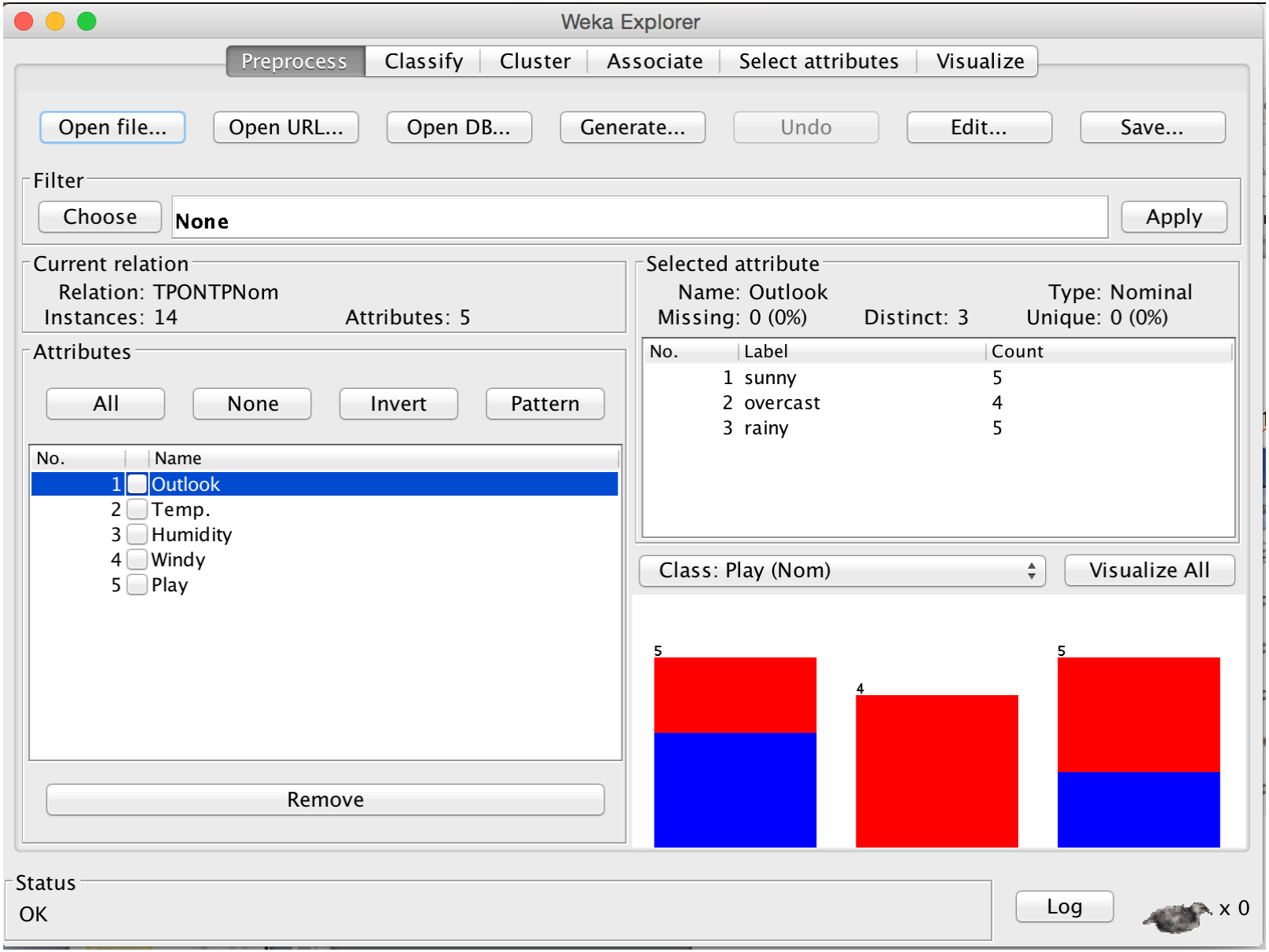
- Datasets from the website Technology Forge:

http://www.technologyforge.net/Datasets

Loading dataset



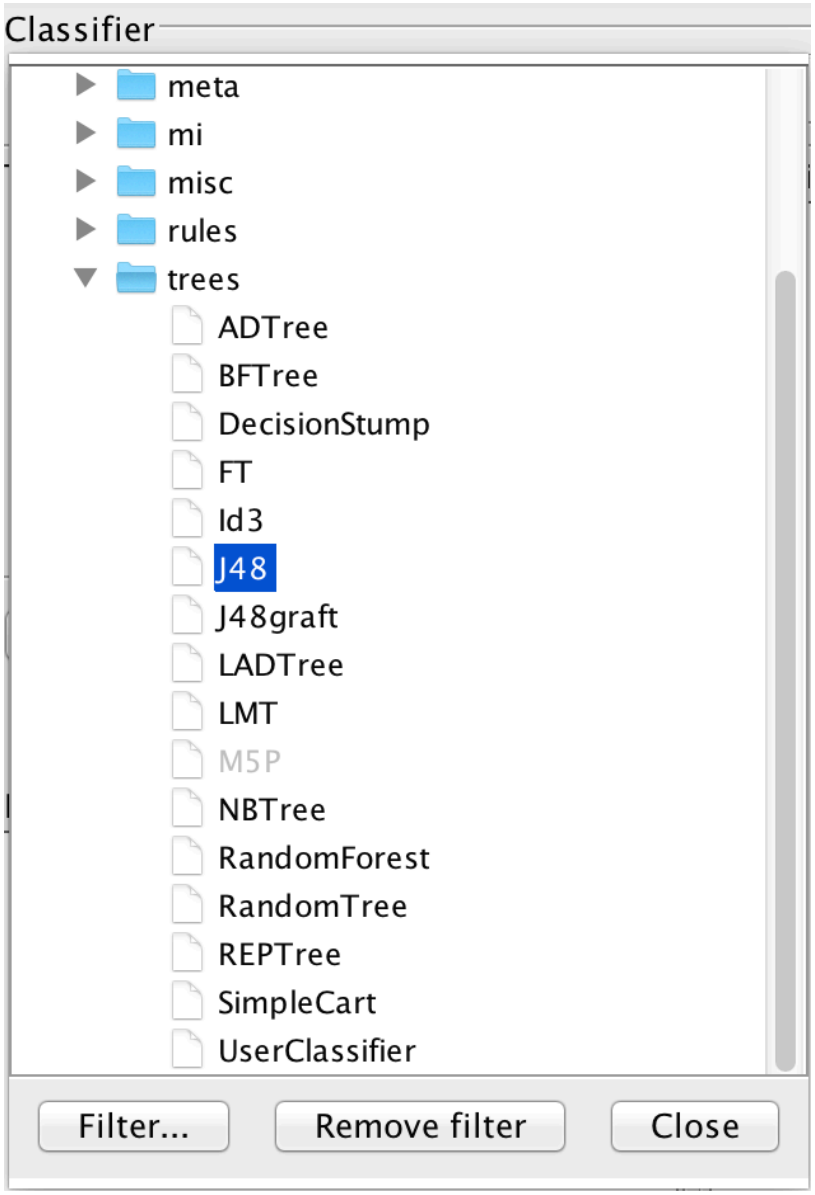
Dataset overview



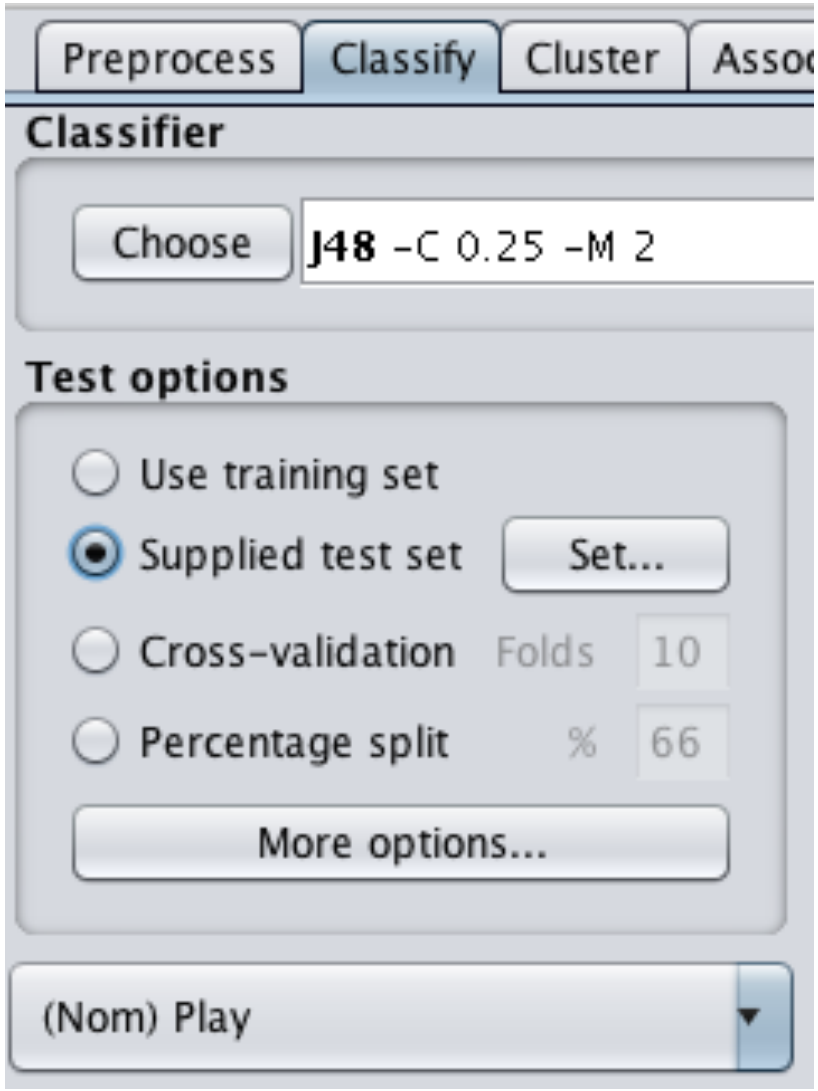
J48 class

- Implementation of C4.5 algorithm for generating decision trees
- C4.5 algorithm is an extension of the ID3 algorithm
- It extends the ID3 algorithm by:
  - supporting continuous and discrete attributes
  - supporting missing values (excludes instances with missing values when calculating entropy and information gain)
  - tree pruning
- Ross Quinlan (1993). C4.5: Programs for Machine Learning. Morgan Kaufmann Publishers, San Mateo, CA.

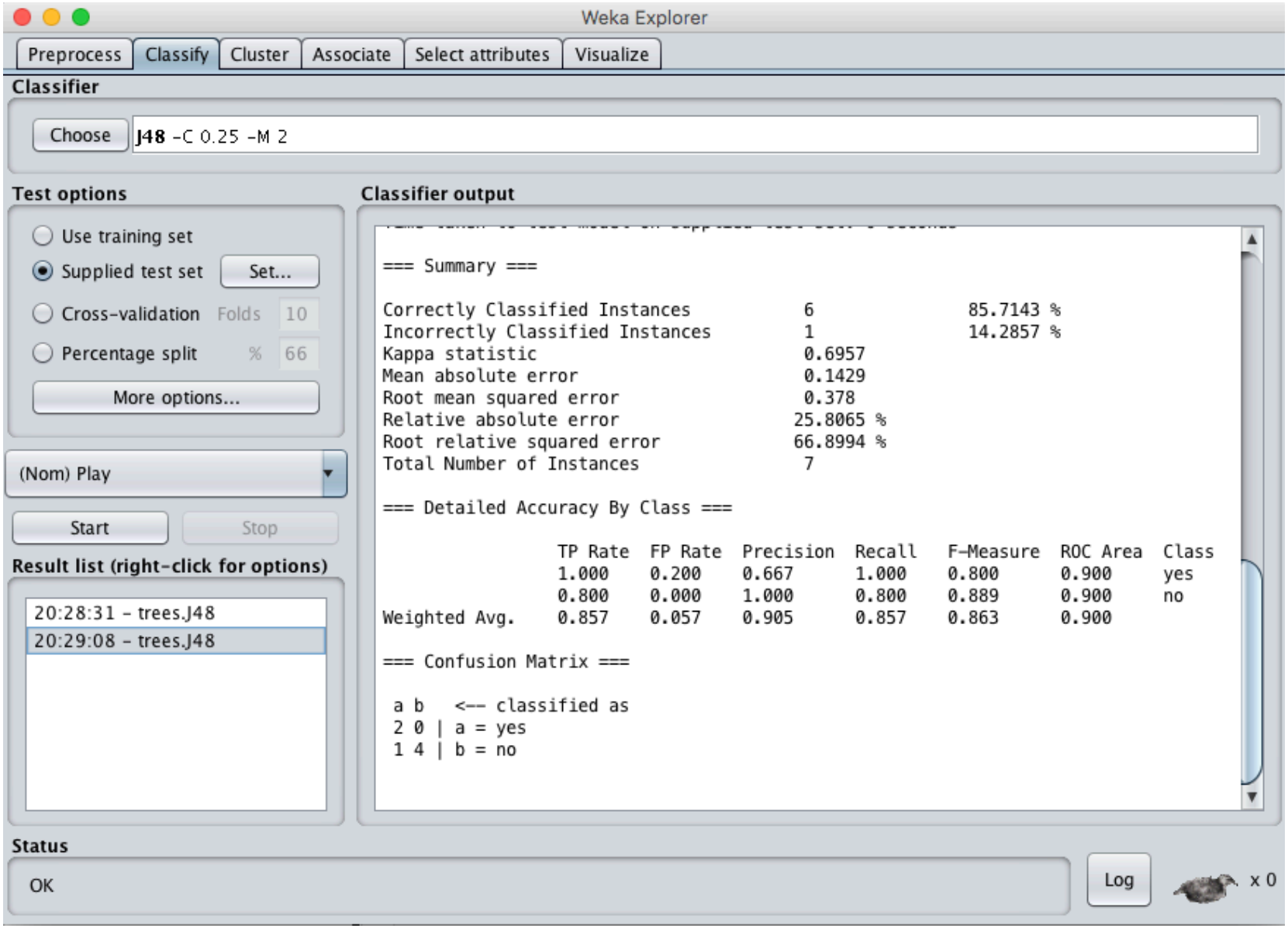
Choosing J48 classifier



Training the classifier



Overview of classification results



Confusion Matrix

		Predicted Class: Yes	Predicted Class: No
Actual Class	Yes	TP	FN
	No	FP	TN

TP = True Positive

FP = False Positive

TN = True Negative

FN = False Negative

```
=== Confusion Matrix ===

 a b   <-- classified as
 2 0 | a = yes
 1 4 | b = no
```

Precision, Recall and F measure

```
=== Detailed Accuracy By Class ===

                 TP Rate   FP Rate   Precision   Recall   F-Measure   ROC Area  Class
                 0.556     0.600     0.625       0.556    0.588       0.633     yes
                 0.400     0.444     0.333       0.400    0.364       0.633     no
Weighted Avg.    0.500     0.544     0.521       0.500    0.508       0.633
```

Precision = $\frac{TP}{(TP + FP)}$

TP Rate (Recall) = $\frac{TP}{(TP + FN)}$

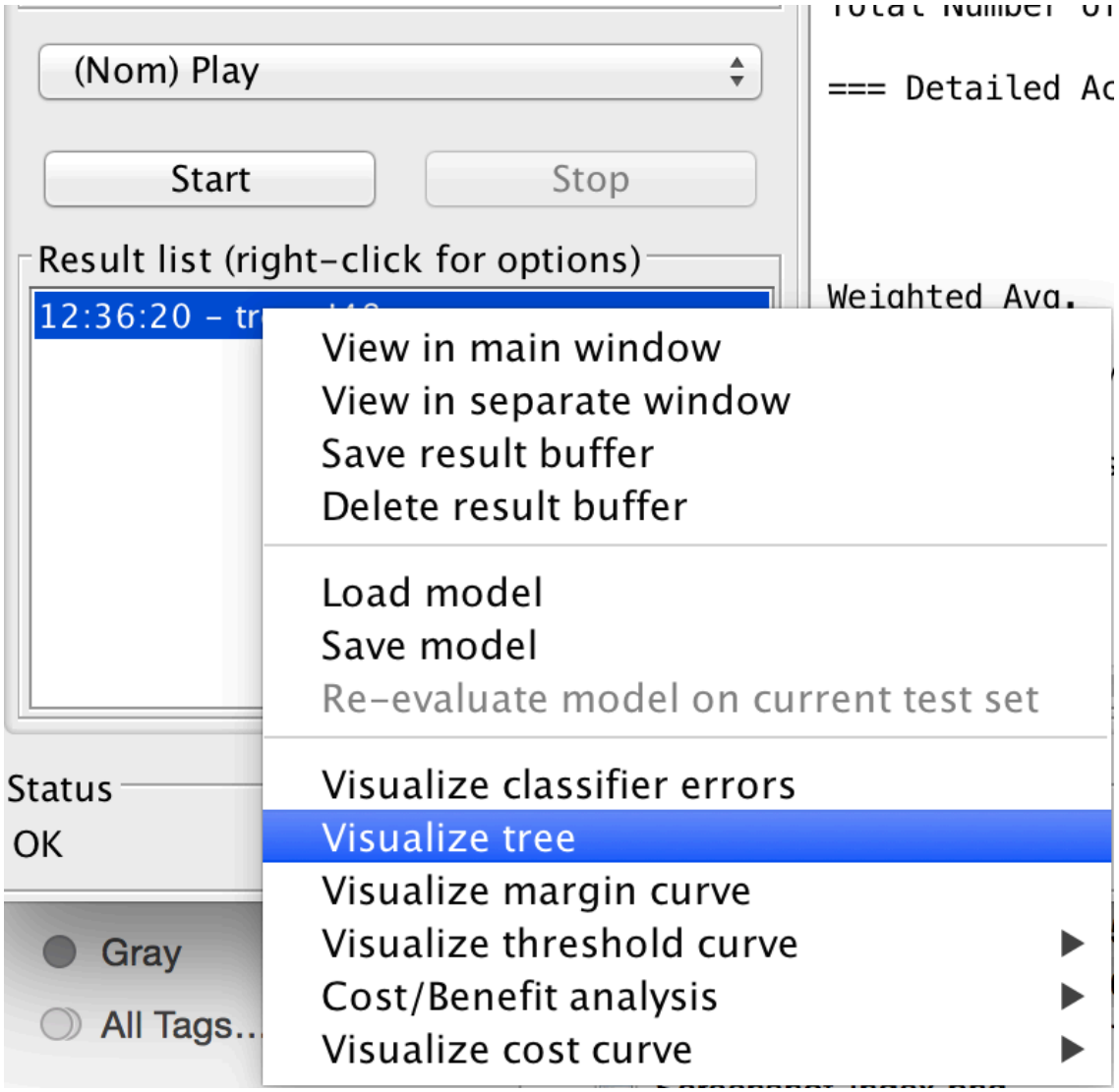
Recall (TP Rate) = $\frac{TP}{(TP + FN)}$

FP Rate = $\frac{FP}{(FP + TN)}$

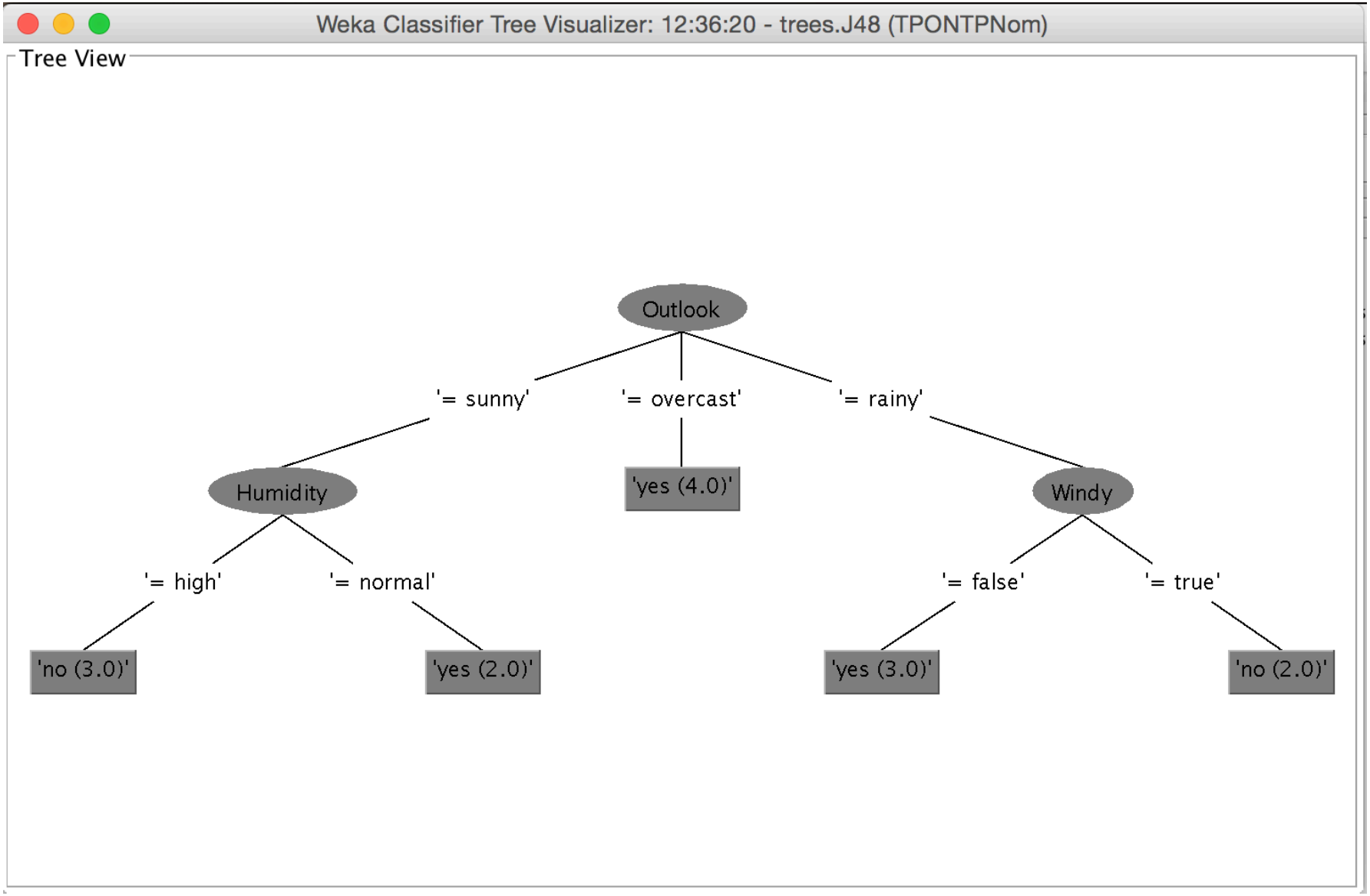
F1 measure = $\frac{2 * Precision * Recall}{Precision + Recall}$

```
=== Confusion Matrix ===

 a b   <-- classified as
 2 0 | a = yes
 1 4 | b = no
```

Visualizing decision tree



Visualizing decision tree



Tree pruning

Tree pruning

- Pruning is the process of reducing the tree size by removing its parts, that is, one or more sub-trees, in order to:
  - avoid over-fitting the model to the training data
  - improve the tree's prediction power on unseen data
- Sub-tree with classification error (on unseen data) bigger than the error of one of its leaf (terminal) nodes is removed and replaced by the leaf node with min. error

Example 2 – "Diabetes" dataset

- Dataset "Pima Indians Diabetes Database" contains data about female Pima Indians age 21 years or higher and tested for diabetes.
- Donated by the Johns Hopkins University, Maryland, USA.
- There are 768 instances described by 8 numerical attributes about the patients' conditions and annotated with a class determining whether the patients were positive or negative for diabetes
- The goal is to predict whether a new patient will be diagnosed positive or negative for diabetes

Example 3 – “Breast cancer” dataset

- “Breast cancer ” dataset contains information about patients diagnosed with breast cancer; it was donated by Institute of Oncology, Ljubljana, Slovenia.
- This dataset includes 201 instances of one class and 85 instances of another class. The instances are described by 9 nominal attributes
- Our goal is to predict whether there will be recurrent events of cancer or not.

Credits

"Data Mining with Weka" and "More Data Mining with Weka": MOOCs from the University of Waikato.

Link: https://www.youtube.com/user/WekaMOOC/

(Anonymous) survey for your comments ad suggestions:

http://goo.gl/cqdp3l