COMP9321

Data Services Engineering

Term 1, 2020

Week 7 Thursday Lecture

Decision Tree

Supervised Learning

Background

- Decision trees have a long history in machine learning
- The first popular algorithm dates back to 1979
- Very popular in many real world problems
  - Intuitive to understand
  - Easy to build

Motivation

- How do people make decisions?
    - Consider a variety of factors
    - Follow a logical path of checks
- An Example
    - Should I eat at this restaurant?
        - If there is no wait
            - Yes
    - If there is short wait and I am hungry
        - Yes
    - Else
        - No

Advantages

- Handling of categorical variables
- Handling of missing values and unknown labels
- Detection of nonlinear relationships
- Visualization and interpretation in decision trees

When to consider Decision Trees

- Instances describable by attribute-value pairs
- Target function is discrete valued
- Disjunctive hypothesis may be required
  分离的
- Possibly noisy training data
- Missing attribute values

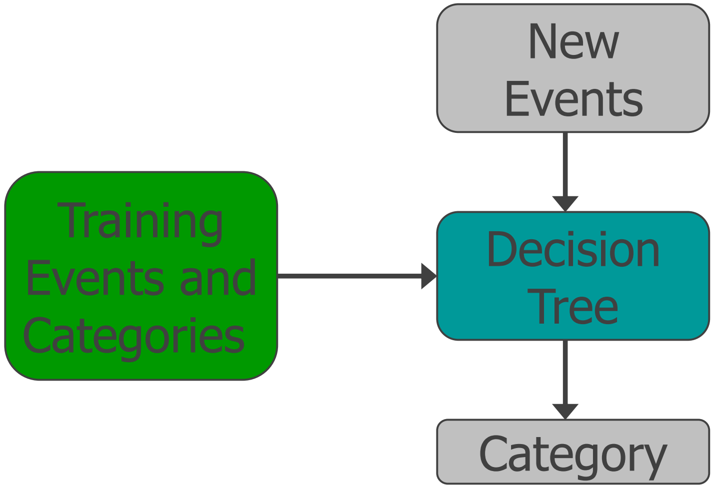
Examples:

- Medical diagnosis
- Credit risk analysis
- Object classification for robot manipulator (Tan 1993)

Introduction

Use a decision tree to predict categories for new events.

Use training data to build the decision tree.



Introduction

A decision tree has 2 kinds of nodes

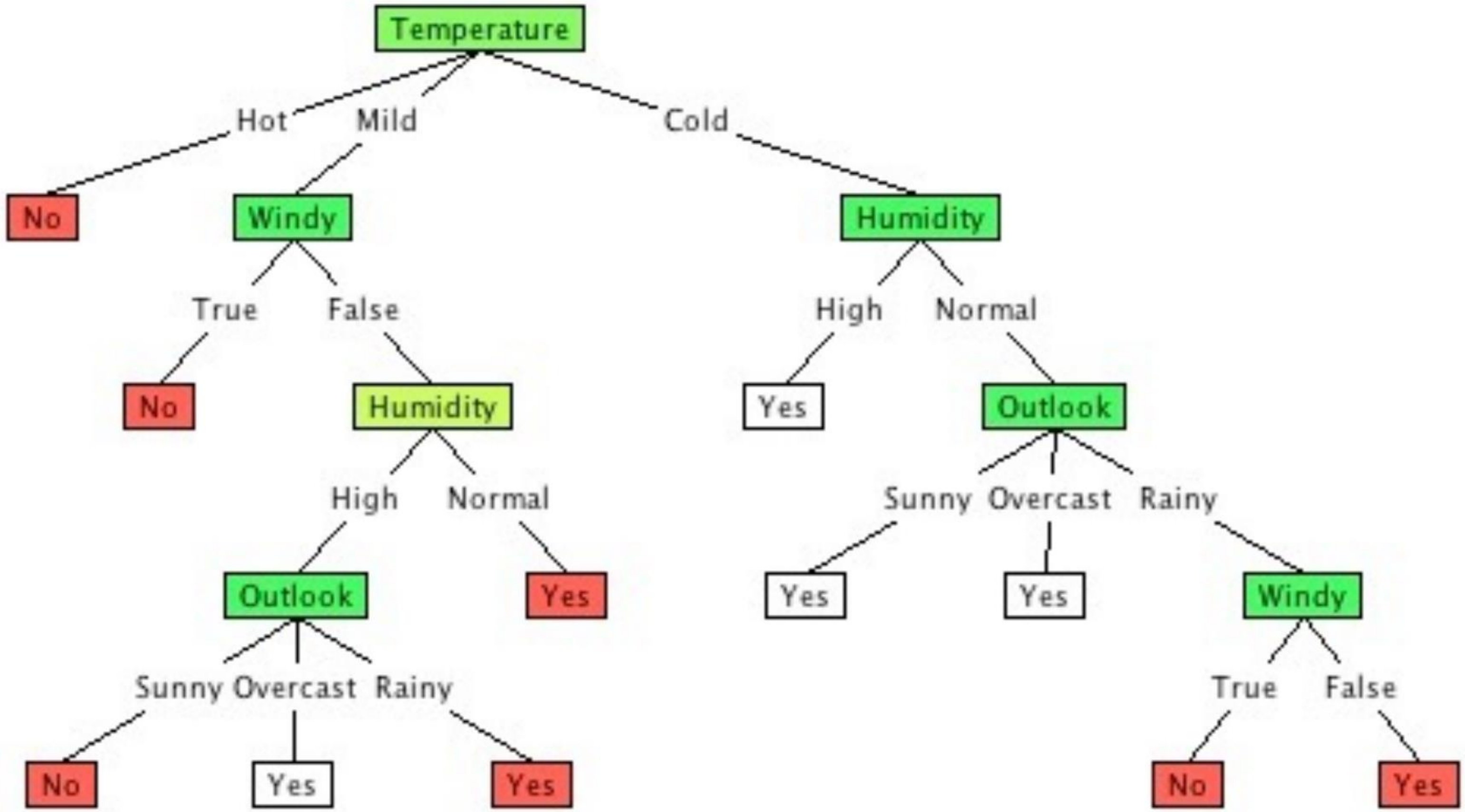
1. Each **leaf node** has a class label, determined by majority vote of training examples reaching that leaf.
2. Each **internal node** is a question on features. It branches out according to the answers.

Decision Tree for Play Tennis

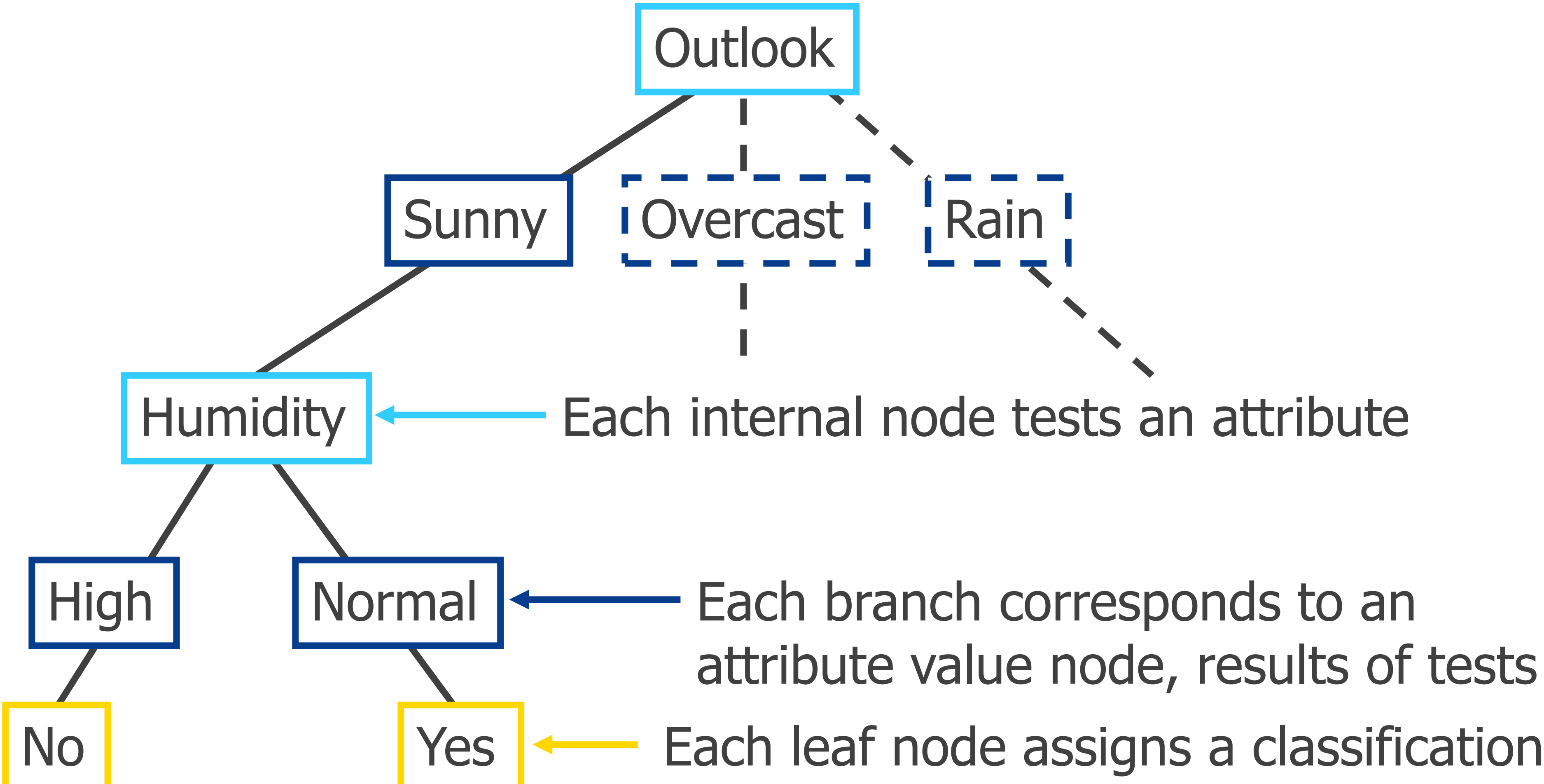
Play Tennis	Outlook	Temperature	Humidity	Windy
No	Sunny	Hot	High	No
No	Sunny	Hot	High	Yes
Yes	Overcast	Hot	High	No
Yes	Rainy	Mild	High	No
Yes	Rainy	Cold	Normal	No

If temperature is not hot → Play
If outlook is overcast → Play
Otherwise → Don’t play tennis

Decision Tree for Play Tennis



Decision Tree for PlayTennis - Structure



What Makes a Good Tree

Small tree:

- Occam’s razor: a guideline to help us choose. Simpler is better
- Avoids over-fitting

What Makes a Good Tree

- Occam’s razor: a guideline to help us choose. Simpler is better

What Makes a Good Tree

- Occam's razor: a guideline to help us choose. Simpler is better

Prank explanation requires:	**Licking explanation requires:**
1. Human (not observed)	1. Cat (observed)
2. Ability to enter house (unknown)	2. Licking (observed)
3. Motive to play prank (unknown)	
4. Leaving no other trace (observed)	

https://www.youtube.com/watch?v=AQNxNeQ9cxw

What Makes a Good Tree

- A decision tree may be human readable, but not use human logic!
- How do we build small trees that accurately capture data?
- Learning an optimal decision tree is computationally intractable

Greedy Algorithm

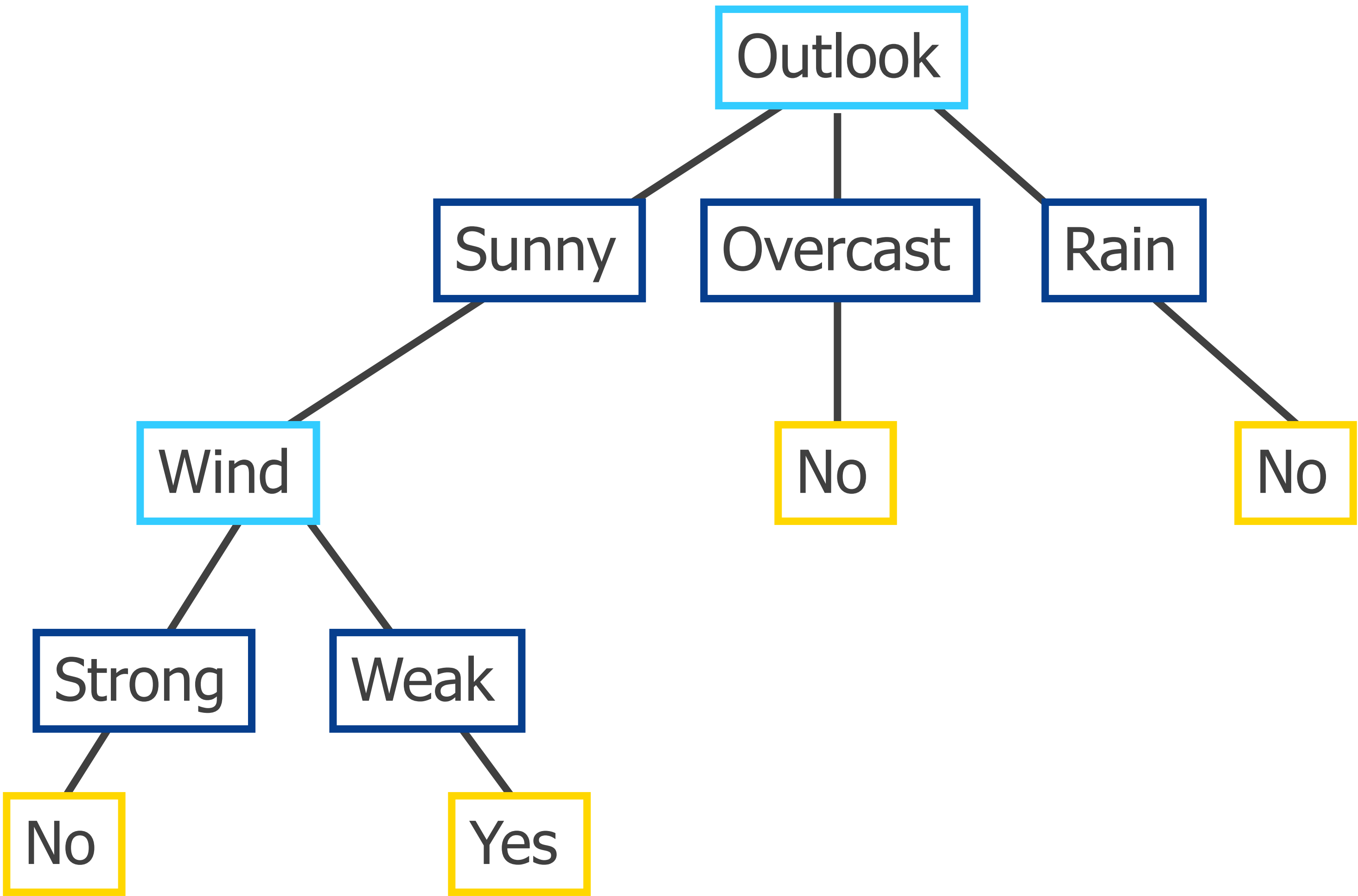
- We can get good trees by simple greedy algorithms
- Adjustments are usually to fix greedy selection problems

Recursive:

1. Select the “best” variable, and generate child nodes: One for each possible value;
2. Partition samples using the possible values, and assign these subsets of samples to the child nodes;
3. Repeat for each child node until all samples associated with a node that are either all positive or all negative.

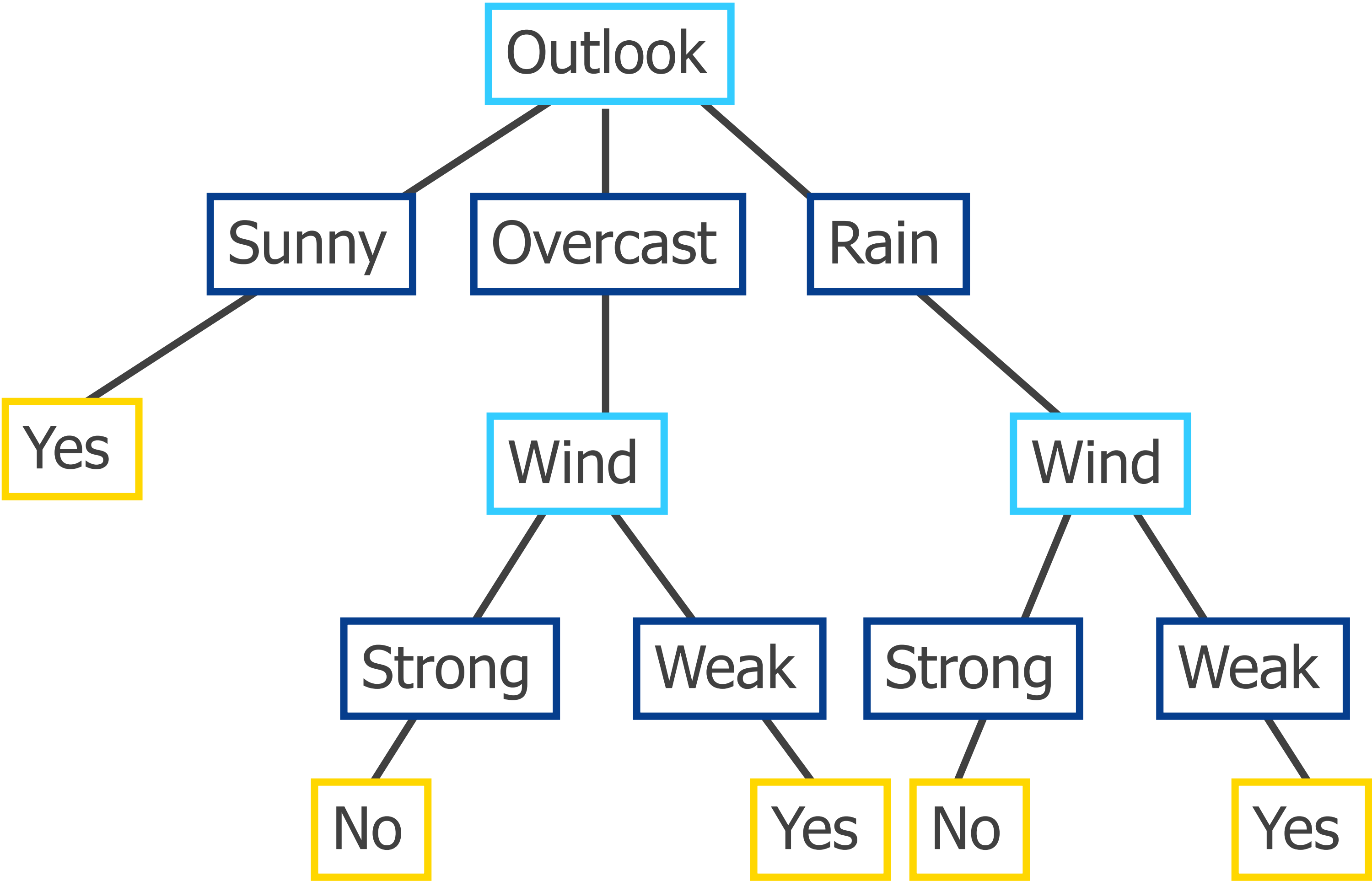
Decision Tree for Conjunction

Outlook=Sunny $\wedge$ Wind=Weak



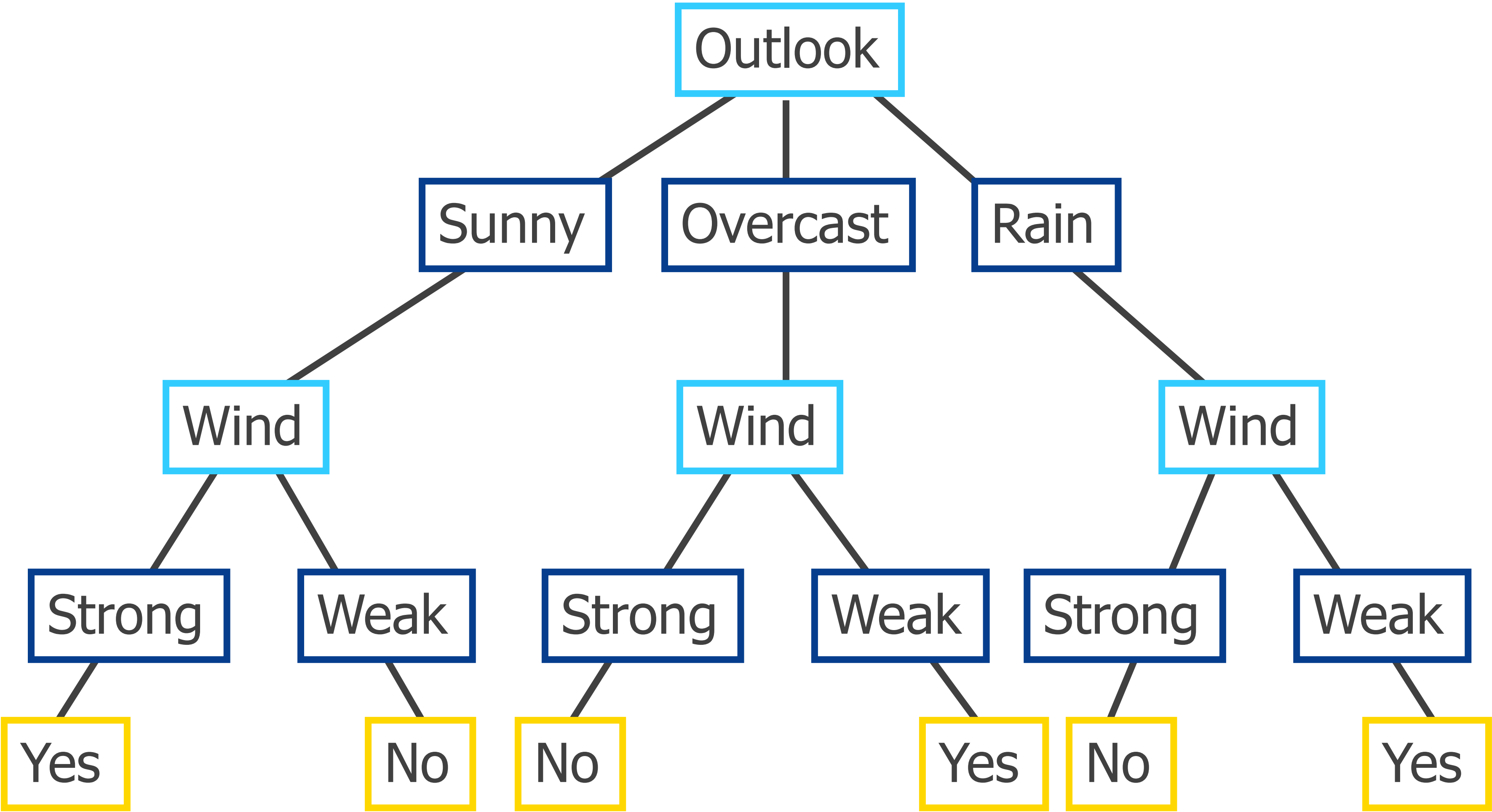
Decision Tree for Disjunction

Outlook=Sunny $\vee$ Wind=Weak



Decision Tree for XOR

Outlook=Sunny XOR Wind=Weak



UNSW SYDNEY

Top-Down Induction of Decision Trees ID3

1. A $\leftarrow$ the "best" decision attribute for next *node*
2. Assign A as decision attribute for *node*
3. For each value of A create new descendant
4. Sort training examples to leaf node according to the attribute value of the branch
5. If all training examples are perfectly classified (same value of target attribute) stop, else iterate over new leaf nodes.

Variable Selection

The best variable for partition

- The most informative variable
- Select the variable that is most informative about the labels

The quantification of information

Founded by Claude Shannon

Basic concepts

Entropy: $H(X) = -\sum_x \mathbb{P}(X = x) \log \mathbb{P}(X = x)$

Conditional Entropy: $H(Y|X) = \sum_x \mathbb{P}(X = x) H(Y|X = x)$

Information Gain: $\mathsf{IG}(\mathsf{Y}|\mathsf{X}) = \mathsf{H}(\mathsf{Y}) - \mathsf{H}(\mathsf{Y}|\mathsf{X})$

Select the variable with the highest information gain

Entropy

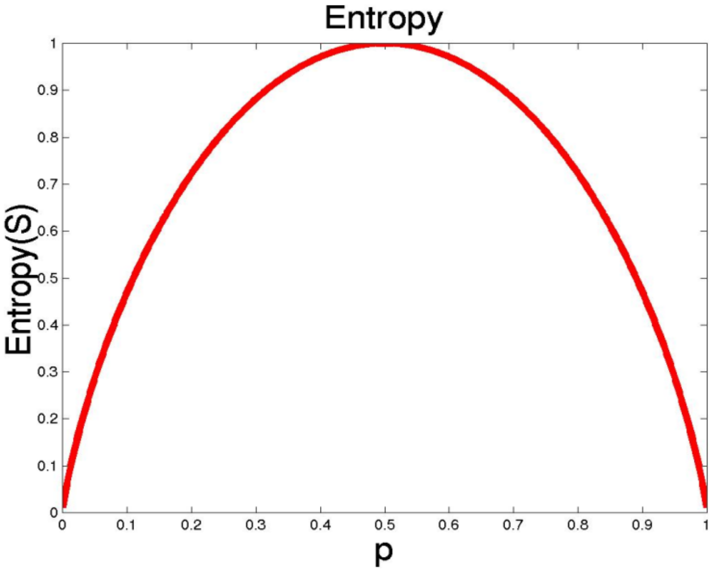
S is a sample of training examples

p_+ is the proportion of positive examples

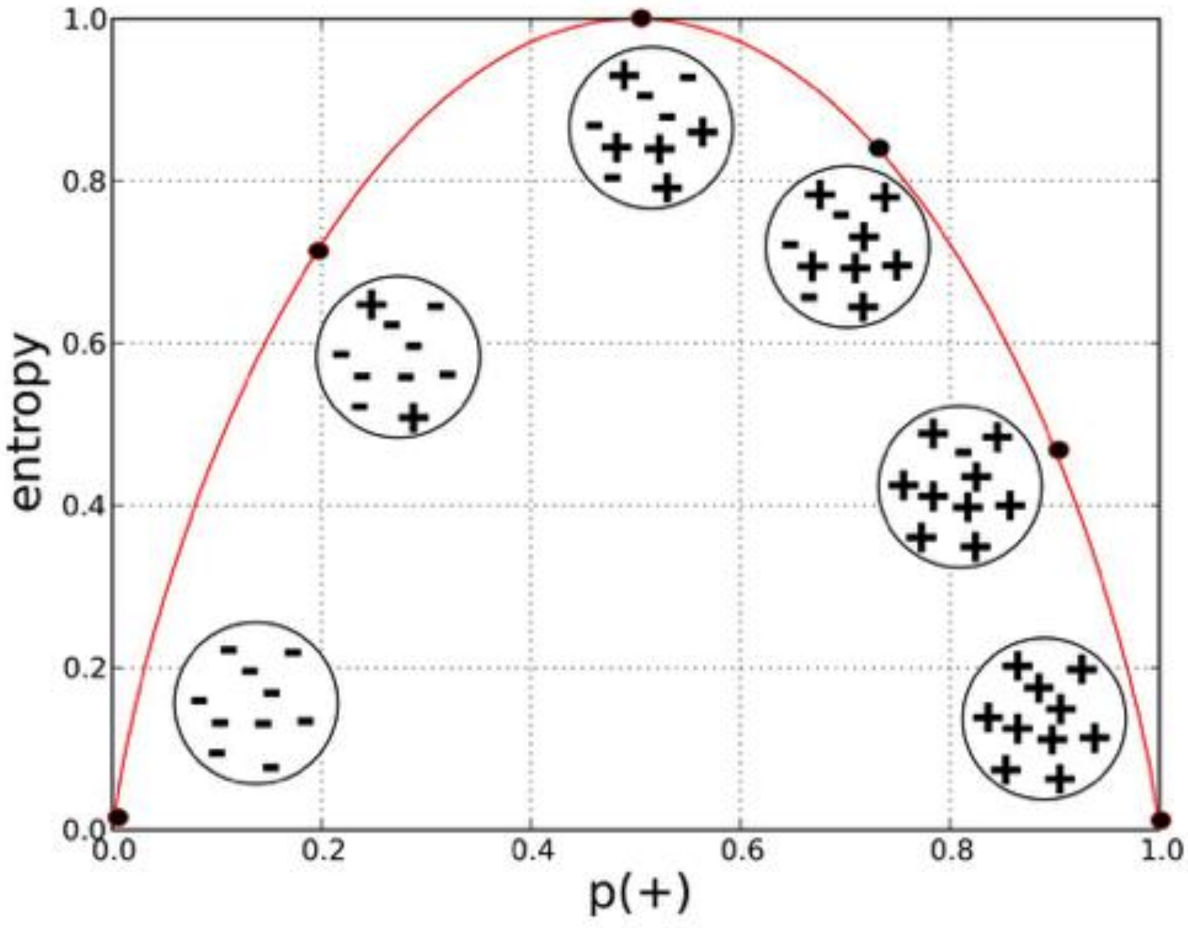
p_- is the proportion of negative examples

Entropy measures the impurity of S

- Entropy(S) = $-p_+ \log_2 p_+ - p_- \log_2 p_-$



Entropy

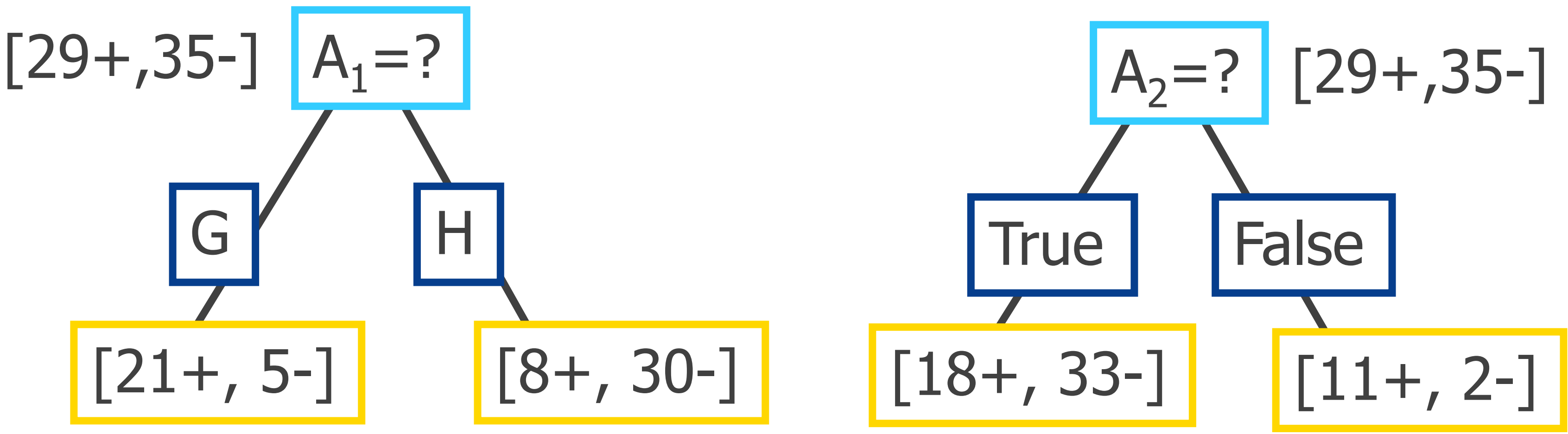


Information Gain (S=E)

Gain(S,A): expected reduction in entropy due to sorting S on attribute A

Entropy([29+,35-]) = -29/64 log2 29/64 – 35/64 log2 35/64
= 0.99

$$Gain(S,A) \equiv Entropy(S) - \sum_{v \in D_A} \frac{|S_v|}{|S|} Entropy(S_v)$$



Training Examples

Day	Outlook	Temp.	Humidity	Wind	Play Tennis
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	Weak	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cold	Normal	Weak	Yes
D10	Rain	Mild	Normal	Strong	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

Selecting the Next Attribute

The information gain values for the 4 attributes are:

- Gain(S,Outlook) =0.247
- Gain(S,Humidity) =0.151
- Gain(S,Wind) =0.048
- Gain(S,Temperature) =0.029

where S denotes the collection of training examples

Occam’s Razor

”If two theories explain the facts equally weel, then the simpler theory is to be preferred”

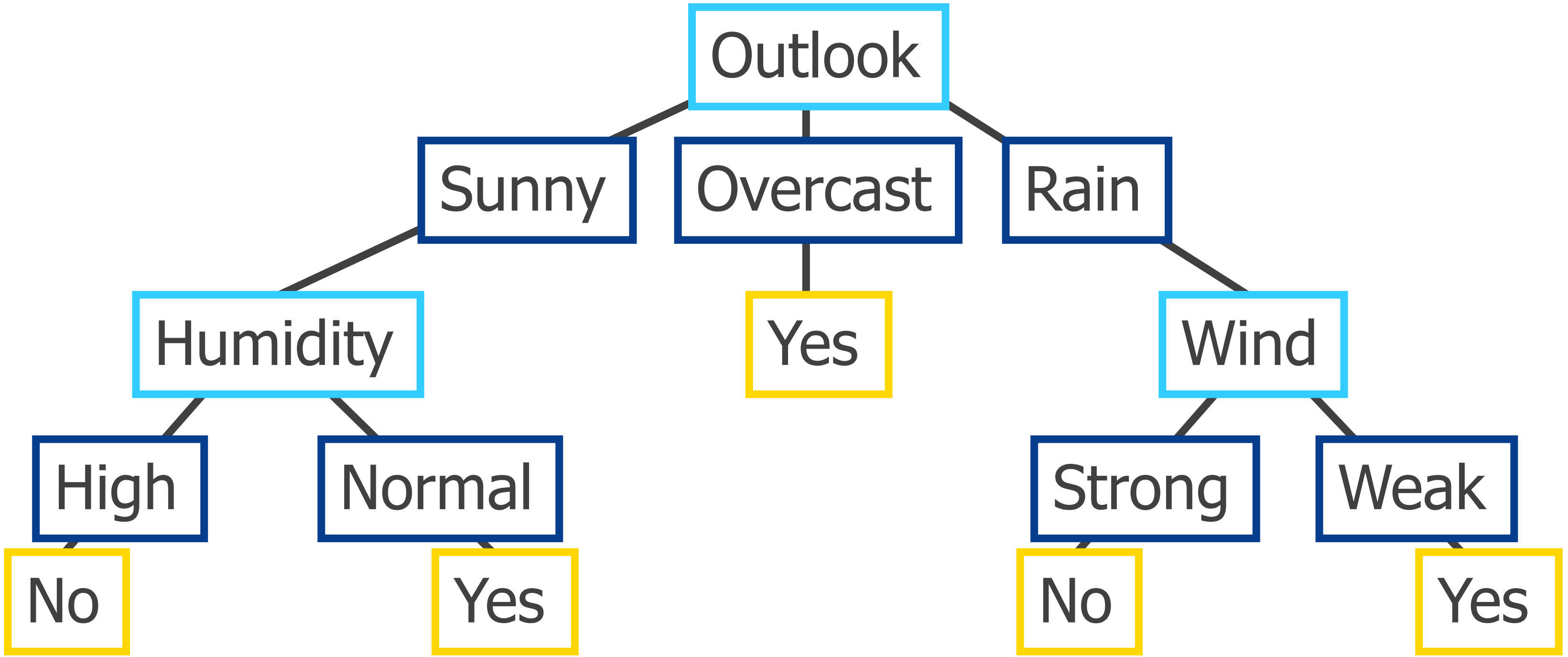
Arguments in favor:

- Fewer short hypotheses than long hypotheses
- A short hypothesis that fits the data is unlikely to be a coincidence
- A long hypothesis that fits the data might be a coincidence

Arguments opposed:

- There are many ways to define small sets of hypotheses

Converting a Tree to Rules



R_1: If (Outlook=Sunny) $\wedge$ (Humidity=High) Then PlayTennis=No
R_2: If (Outlook=Sunny) $\wedge$ (Humidity=Normal) Then PlayTennis=Yes
R_3: If (Outlook=Overcast) Then PlayTennis=Yes
R_4: If (Outlook=Rain) $\wedge$ (Wind=Strong) Then PlayTennis=No
R_5: If (Outlook=Rain) $\wedge$ (Wind=Weak) Then PlayTennis=Yes

Continuous Valued Attributes

Create a discrete attribute to test continuous

Temperature = 24.50C

(Temperature > 20.00C) = {true, false}

Where to set the threshold?

Temperature	15^{0}C	18^{0}C	19^{0}C	22^{0}C	24^{0}C	27^{0}C
PlayTennis	No	No	Yes	Yes	Yes	No

Unknown Attribute Values

What if some examples have missing values of A?

Use training example anyway sort through tree

If node n tests A, assign most common value of A among other examples sorted to node n.

Assign most common value of A among other examples with same target value

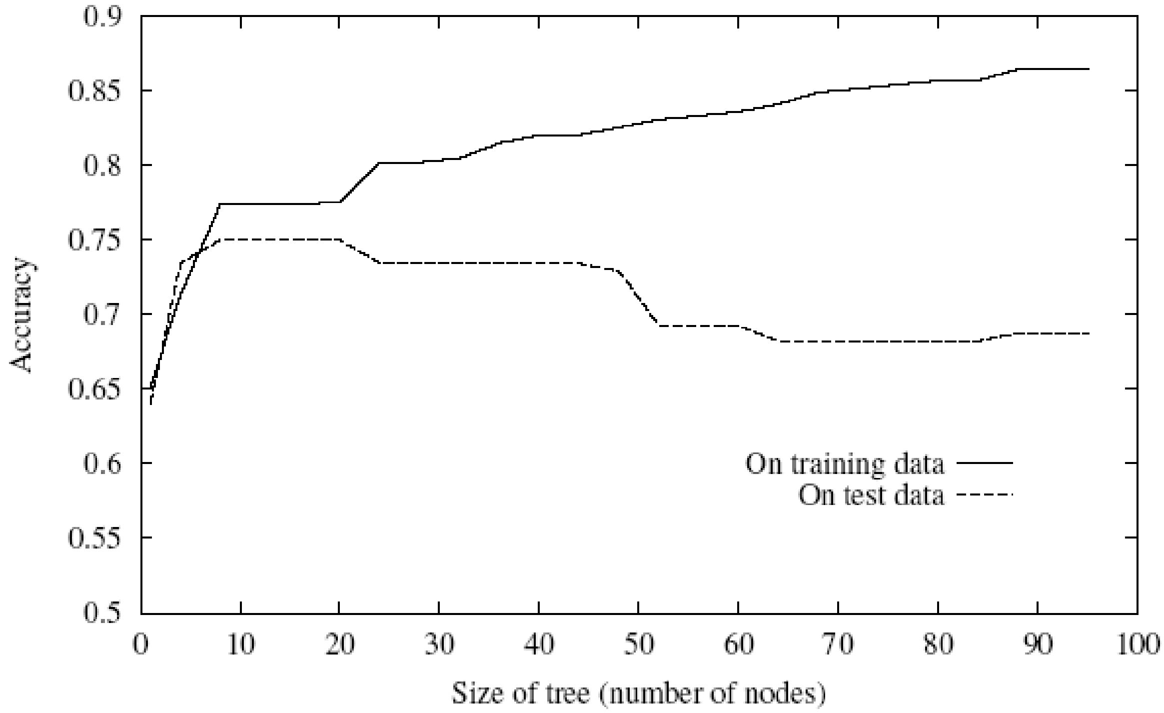
Assign probability pi to each possible value vi of A

- Assign fraction pi of example to each descendant in tree

Classify new examples in the same fashion

Overfitting

One of the biggest problems with decision trees is Overfitting



Avoid Overfitting

stop growing when split not statistically significant

grow full tree, then post-prune

Select “best” tree:

measure performance over training data

measure performance over separate validation data set

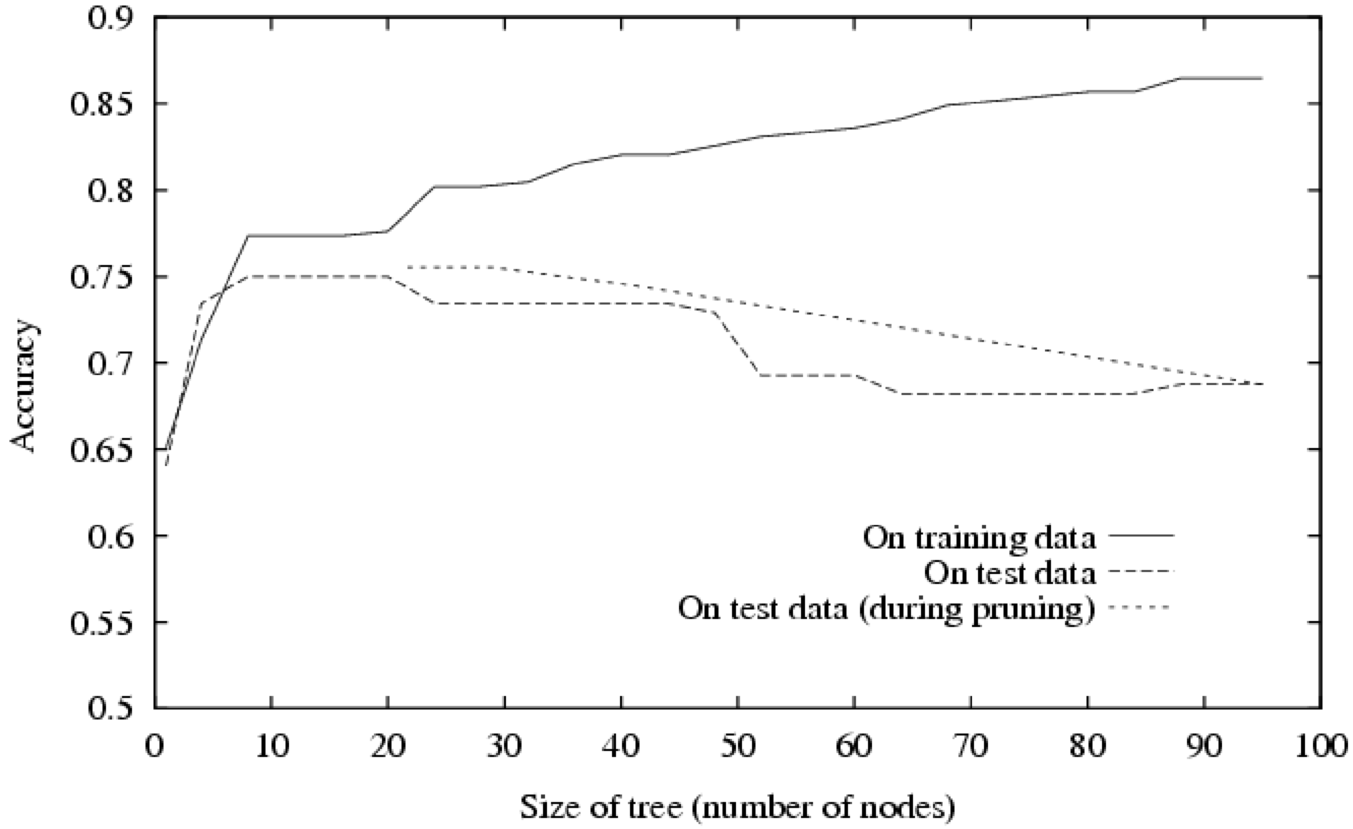
min(|tree|+|misclassifications(tree)|)

Avoid Overfitting

Idea 1: Stop growing the tree when the error doesn't drop by more than a threshold with any new cut.

Idea 2: Prune a large tree from the leaves to the root. Weakest link pruning:

Effect of Reduced Error Pruning



Useful resources

- https://dl.acm.org/citation.cfm?id=541177
- https://towardsdatascience.com/a-guide-to-decision-trees-for-machine-learning-and-data-science-fe2607241956
- https://towardsdatascience.com/entropy-how-decision-trees-make-decisions-2946b9c18c8
- https://victorzhou.com/blog/information-gain/

Questions?