Lecture 15: Decision Trees

COMP90049
Introduction to Machine Learning
Semester 1, 2021

Lea Frermann, CIS

Roadmap

So far ... Classification and Evaluation

- KNN, Naive Bayes, Logistic Regression, Perceptron
- Probabilistic models
- Loss functions, and estimation
- Evaluation

Today... Decision Trees

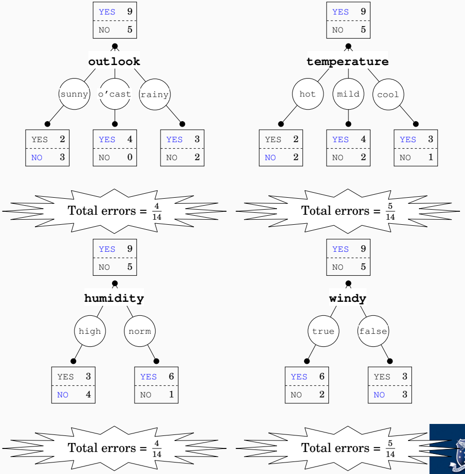
- Definition and motivation
- Estimation (ID3 Algorithm)
- Discussion

THE UNIVERSITY OF MELBOURNE

From Decision Stumps to Decision Trees

We have seen decision stumps in action in the context of 1-R

Given the obvious myopia of decision stumps, how can we construct **decision trees** (of arbitrary depth) which have the ability to capture complex feature interaction?

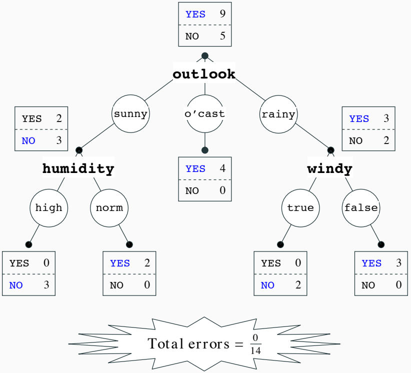


THE UNIVERSITY OF MELBOURNE

The Weather Dataset (again!)

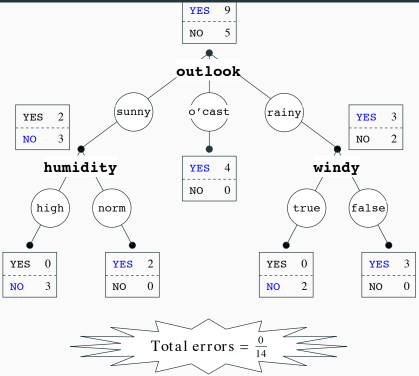
	Outlook	Temperature	Humidity	Windy	Play
a:	sunny	hot	high	FALSE	no
b:	sunny	hot	high	TRUE	no
c:	overcast	hot	high	FALSE	yes
d:	rainy	mild	high	FALSE	yes
e:	rainy	cool	normal	FALSE	yes
f:	rainy	cool	normal	TRUE	no
g:	overcast	cool	normal	TRUE	yes
h:	sunny	mild	high	FALSE	no
i:	sunny	cool	normal	FALSE	yes
j:	rainy	mild	normal	FALSE	yes
k:	sunny	mild	normal	TRUE	yes
l:	overcast	mild	high	TRUE	yes
m:	overcast	hot	normal	FALSE	yes
n:	rainy	mild	high	TRUE	no

Rule-based classification



- Construct the tree
- Extract one rule per leaf node
    1. if (outlook == o'cast) $\rightarrow$ yes
    2. if (outlook == sunny & humidity == normal) $\rightarrow$ yes
    3. if (outlook == rainy) & windy == false) $\rightarrow$ yes
    4. ...

Disjunctive descriptions



Decision Trees can be read as a disjunction; for example, `Yes`:

$$\begin{aligned} (outlook = sunny \wedge humidity = normal) \\ \vee(outlook = overcast) \\ \vee(outlook = rainy \wedge windy = false) \end{aligned}$$

Decision Trees: Classifying Novel Instances

At test time...

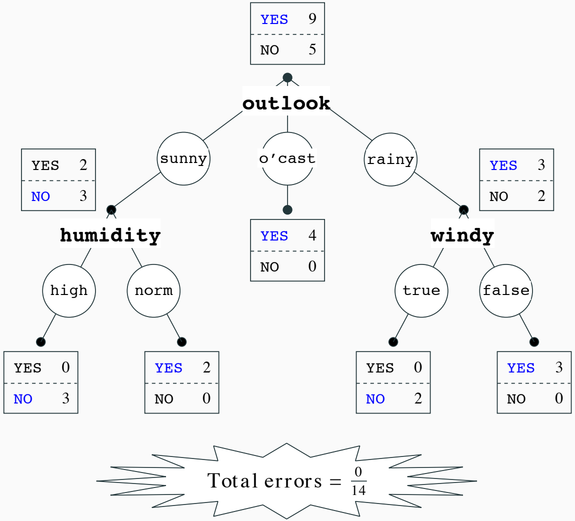
- Assume we have constructed a decision tree
- Now, classify novel instances by traversing down the tree and predict the class according to the label of the deepest reachable point in the tree structure (leaf)

Complications

- unobserved attribute–value pairs
- missing values

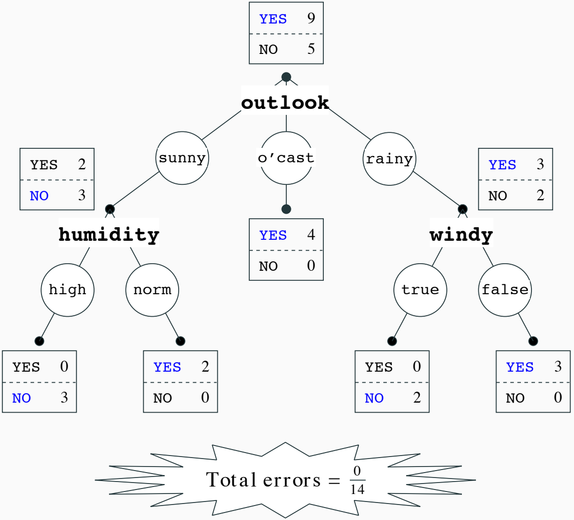
Classification Example

Classify test instance: **(sunny, hot, normal, False)**



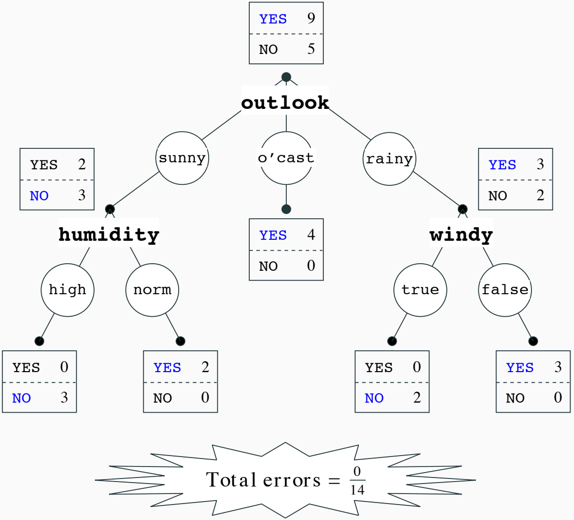
Classification Example

Classify test instance: **(rainy, hot, low, False)**



Classification Example

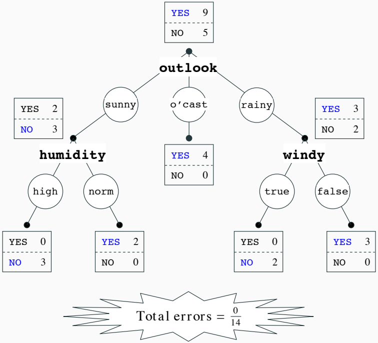
Classify test instance: **(?,cool, high, True)**



Decision Trees: Issues

Issues

- How to build an optimal tree?
- What does 'optimal' mean?
- How to choose attributes for decision points?
- When to stop growing the tree?



ID3 Algorithm

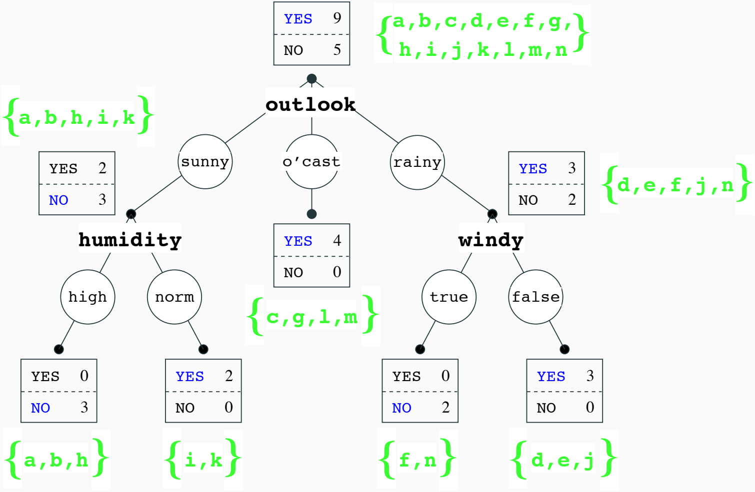
ID3: Overview

Optimal construction of a Decision Tree is **NP hard** (non-deterministic polynomial).

So we use heuristics:

- Choose an attribute to partition the data at the node such that each partition is as **pure** (homogeneous) as possible.
- In each partition most of the instances should belong to as few classes as possible
- Each partition should be as large as possible.

We can stop the growth of the tree if all the leaf nodes are (largely) dominated by a single class (that is the leaf nodes are nearly pure).


YES 9
NO 5
{a,b,c,d,e,f,g,h,i,j,k,l,m,n}
outlook
{a,b,h,i,k}
sunny
o'cast
rainy
YES 2
NO 3
YES 3
NO 2
{d,e,f,j,n}
humidity
YES 4
NO 0
windy
high
norm
{c,g,l,m}
true
false
YES 0
NO 3
YES 2
NO 0
YES 0
NO 2
YES 3
NO 0
{a,b,h}
{i,k}
{f,n}
{d,e,j}

Constructing Decision Trees: ID3

Basic method: recursive divide-and-conquer

FUNCTION ID3 (Root)

IF all instances at root have same class**

THEN stop

ELSE 1. Select a new attribute to use in partitioning root node instances

2. Create a branch for each attribute value and partition up root node instances according to each value

3. Call ID3($LEAF_i$) for each leaf node $LEAF_i$

**This is overly simplified, as we will discuss momentarily

Criterion for Attribute Selection

How do we choose the attribute to partition the instances at a given node?

We want to get the smallest tree (Occam's Razor; generalisability). Prefer the shortest hypothesis that fits the data.

In favor:

- Fewer short hypotheses than long hypotheses
  - a short hyp. that fits the data unlikely to be a coincidence
  - a long hyp. that fits data might be a coincidence

Against:

- Many ways to define small sets of hypotheses

Entropy and Information Gain (Intuition)

Information Gain: ‘Reduction of entropy before and after the data is partitioned using the attribute A’.

Entropy: The expected (average) level of surprise or uncertainty.

Given a random variable (e.g., a coinflip), how surprised am I when seeing a certain outcome?

Entropy and Information Gain (Intuition)

Information Gain: ‘Reduction of entropy before and after the data is partitioned using the attribute A’.

Entropy: The expected (average) level of surprise or uncertainty.

Given a random variable (e.g., a coinflip), how surprised am I when seeing a certain outcome?

- **Low probability** event: if it happens, it’s big news! High surprise! High information!
- **High probability** event: it was likely to happen anyway. Not very surprising. Low information!

Entropy (Definition)

- A measure of **unpredictability**
- Level of unpredictability (surprise) for a single event i: **self-information**

$$\text{self-info}(i) = \frac{1}{P(i)} = -\log_2 P(i)$$

- Given a probability distribution, the information (in bits) required to predict an event is the distribution's **entropy** or **information value**
- The entropy of a discrete random event x with possible outcomes $1, ..n$ is:

$$\begin{aligned} H(x) &= \sum_{i=1}^{n} P(i) \textit{self-info}(i) \\ &= -\sum_{i=1}^{n} P(i) \log_2 P(i) \end{aligned}$$

where $0 \log_2 0 =^{def} 0$

Entropy (Definition)

Example 1 Coin flips.

- Biased coin. 55 flips: 50x *head*, 5x *tail*:

$$\approx \quad 0.44 \text{ bits}$$

- Fair coin. 55 flips: 30x *head*, 25x *tail*:

$$\approx \quad 0.99 \text{ bits}$$

The more uncertainty, the higher the entropy.

Example 2 In the context of Decision Trees, we are looking at the class distribution at a node:

- 50 Y instances, 5 N instances:

$$\approx \quad 0.44 \text{ bits}$$

- 30 Y instances, 25 N instances:

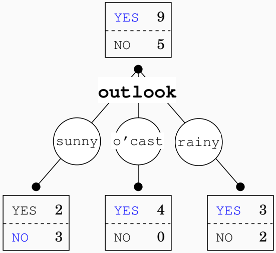
$$\approx \quad 0.99 \text{ bits}$$

We want to classify with high certainty. We want leaves with **low entropy**!

Entropy: Summary

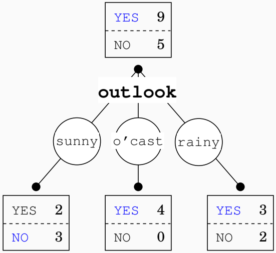
Entropy is a measure of **unpredictability**

- If the probability of a single class is **high**
  - Probability mass is centered
  - Entropy is low
  - The event is **predictable**
- If the probability is **evenly divided** between multiple classes
  - Probability mass is spread out
  - Entropy is high
  - The event is **unpredictable**



From Entropy to Information Gain

- Decision tree with low entropy: class is more predictable.
- **Information Gain** (reduction of entropy): measures how much uncertainty was reduced.
- Select the **attribute** that has largest information gain: the most entropy (uncertainty) is reduced, class is most predictable.



Information Gain

The expected reduction in entropy caused by knowing the value of an attribute.

Compare

- the entropy before splitting the tree using the attribute's values
- the weighted average of the entropy over the children after the split. This is called the (**Mean Information**)

If the entropy **decreases**, then we have a better tree (more predictable)

Mean Information Associated with a Decision Stump

- We calculate the mean information for a tree stump with m attribute values as:

$$\text{Mean Info}(x_1, .., x_m) = \sum_{i=1}^{m} P(x_i)H(x_i)$$

  where $H(x_i)$ is the entropy of the class distribution for the instances at node x_i

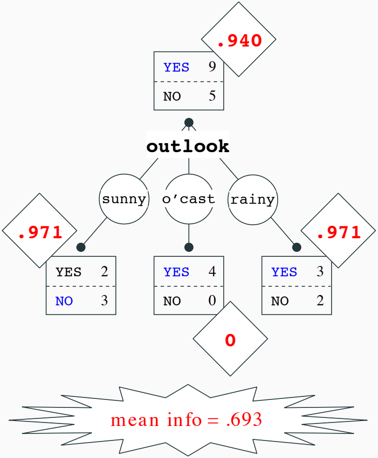
  and $P(x_i)$ is the proportion of instances at sub-node x_i

Mean Information (outlook)

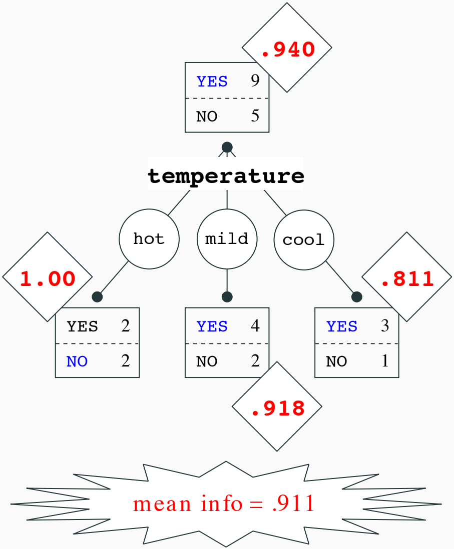
$H(x) = -\sum_i P(x_i) \log_2 P(x_i)$

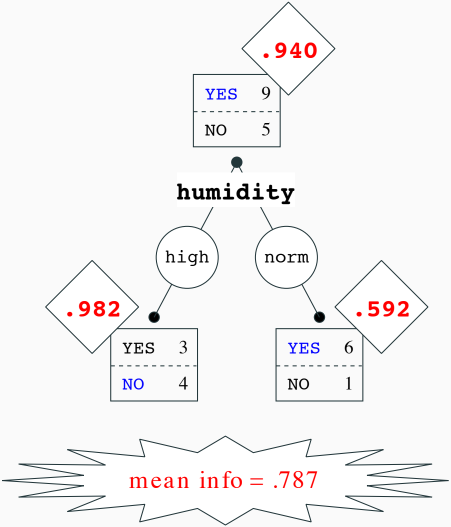
H(rainy) =

=0.971

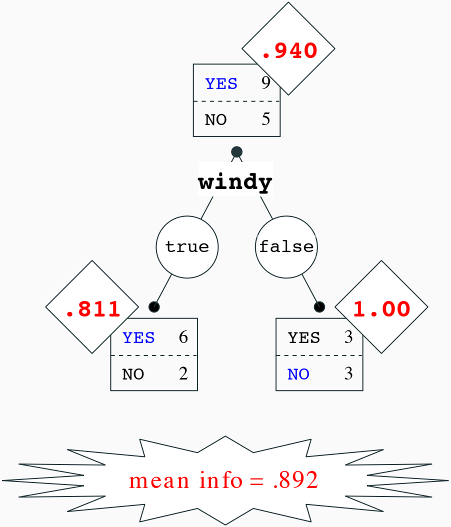


Mean Information (temperature)




.940
YES 9
NO 5
humidity
high
norm
.982
YES 3
NO 4
.592
YES 6
NO 1
mean info = .787

Mean Information (`windy`)



Attribute Selection: Information Gain

- We determine which attribute R_A (with values $x_1, ...x_m$) best partitions the instances at a given root node R according to **information gain** (IG):

$$\begin{aligned} IG(R_A|R) &= H(R) - \text{mean-info}(R_A) \\ &= H(R) - \sum_{i=1}^{m} P(x_i)H(x_i) \end{aligned}$$

$$\begin{aligned} IG(\mathtt{outlook}|R) &= 0.247 \\ IG(\mathtt{temperature}|R) &= 0.029 \\ IG(\mathtt{humidity}|R) &= 0.152 \\ IG(\mathtt{windy}|R) &= 0.048 \end{aligned}$$

$H(R) = 0.94$
$Mean\_info(outlook) = 0.693$
$Mean\_info(temperature) = 0.911$
$Mean\_info(humidity) = 0.787$
$Mean\_info(windy) = 0.892$

Attribute Selection: Information Gain

- We determine which attribute R_A (with values $x_1, ...x_m$) best partitions the instances at a given root node R according to **information gain**:

$$
\begin{aligned}
IG(R_A|R) &= H(R) - \text{mean-info}(R_A) \\
&= H(R) - \sum_{i=1}^{m} P(x_i)H(x_i)
\end{aligned}
$$

$$
\begin{aligned}
IG(\mathtt{outlook}|R) &= 0.247 \\
IG(\mathtt{temperature}|R) &= 0.029 \\
IG(\mathtt{humidity}|R) &= 0.152 \\
IG(\mathtt{windy}|R) &= 0.048
\end{aligned}
$$

$H(R) = 0.94$
$Mean\_info(outlook) = 0.693$
$Mean\_info(temperature) = 0.911$
$Mean\_info(humidity) = 0.787$
$Mean\_info(windy) = 0.892$

Shortcomings of Information Gain

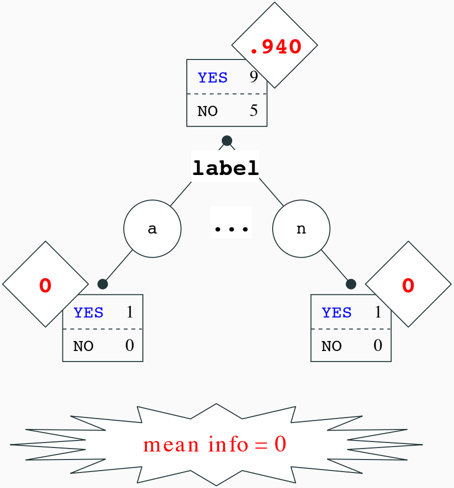
Information gain tends to **prefer highly-branching attributes**:

- A subset of instances is more likely to be homogeneous (pure) if there are only a few instances
- Attribute with many values will have fewer instances at each child node

This may result in **overfitting** / fragmentation

Mean Information (`label`)

Information gain tends to **prefer highly-branching attributes**:



Solution: Gain Ratio

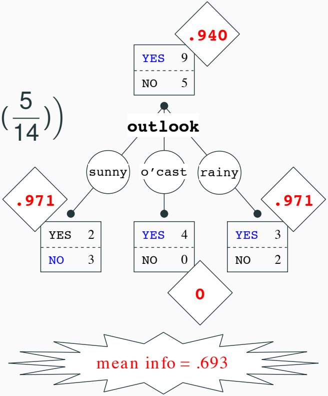
- **Gain ratio (GR)** reduces the bias for information gain towards highly-branching attributes by normalising relative to the **split information**
- **Split info (SI)** is the entropy of a given split (evenness of the distribution of instances to attribute values)

$$
\begin{aligned}
GR(R_A|R) &= \frac{IG(R_A|R)}{SI(R_A|R)} = \frac{IG(R_A|R)}{H(R_A)} \\
&= \frac{H(R) - \sum_{i=1}^{m} P(x_i)H(x_i)}{-\sum_{i=1}^{m} P(x_i)\log_2 P(x_i)}
\end{aligned}
$$

- Split Info sometimes called *Intrinsic Value*
- Discourages the selection of attributes with many uniformly distributed values

Split Info

$$SI(\texttt{outlook}|R)$$
$$= -\left((\frac{5}{14})\log_2(\frac{5}{14}) + (\frac{4}{14})\log_2(\frac{4}{14}) + (\frac{5}{14})\log_2(\frac{5}{14})\right)$$
$$= 1.577$$



NB: Entropy of distribution of instances to attribute *values* (disregarding classes, unlike Mean Info)

Gain Ratio: Example

$$
\begin{aligned}
IG(\texttt{outlook}|R) &= 0.247 \\
SI(\texttt{outlook}|R) &= 1.577 \\
GR(\texttt{outlook}|R) &= 0.156 \\
\\
IG(\texttt{humidity}|R) &= 0.152 \\
SI(\texttt{humidity}|R) &= 1.000 \\
GR(\texttt{humidity}|R) &= 0.152 \\
\\
\color{red}{IG(\texttt{label}|R)} &\color{red}{= 0.940} \\
SI(\texttt{label}|R) &= 3.807 \\
\color{red}{GR(\texttt{label}|R)} &\color{red}{= 0.247}
\end{aligned}
$$

$$
\begin{aligned}
IG(\texttt{temperature}|R) &= 0.029 \\
SI(\texttt{temperature}|R) &= 1.557 \\
GR(\texttt{temperature}|R) &= 0.019 \\
\\
IG(\texttt{windy}|R) &= 0.048 \\
SI(\texttt{windy}|R) &= 0.985 \\
GR(\texttt{windy}|R) &= 0.049
\end{aligned}
$$

Stopping criteria i

The definition of ID3 above suggests that:

- We recurse until the instances at a node are of the same class
- This is consistent with our usage of entropy: if all of the instances are of a single class, the entropy of the distribution is 0
- Considering other attributes cannot "improve" an entropy of 0 — the Info Gain is 0 by definition

This helps to ensure that the tree remains compact (Occam's Razor)

Stopping criteria ii

The definition of ID3 above suggests that:

- The Info Gain/Gain Ratio allows us to choose the (seemingly) best attribute at a given node
- However, it is also an approximate indication of how much absolute improvement we expect from partitioning the data according to the values of a given attribute
- An Info Gain of 0 means that there is no improvement; a very small improvement is often unjustifiable
- Typical modification of ID3: choose best attribute only if IG/GR is greater than some **threshold** τ
- Other similar approaches use **pruning** — post-process the tree to remove undesirable branches (with few instances, or small IG/GR improvements)

Stopping criteria iii

The definition of ID3 above suggests that:

- We might observe improvement through every layer of the tree
- We then run out of attributes, even though one or more leaves could be improved further
- Fall back to majority class label for instances at a leaf with a mixed distribution — unclear what to do with ties
- Possibly can be taken as evidence that the given attributes are insufficient for solving the problem

Discussion

Hypothesis Space Search in ID3

- ID3 can be characterized as searching a space of hypotheses for one that fits the training examples.
- The hypothesis space searched by ID3 is the set of possible decision trees.
- ID3 performs a simple-to-complex, hill-climbing search through this hypothesis space (with no backtracking),
  - beginning with the empty tree
  - considering progressively more elaborate hypotheses in search of a decision tree that correctly classifies the training data

Pros / Cons of Decision Trees

Pros

- Highly regarded among basic supervised learners
- Fast to train, even faster to classify
- Very transparent (probably the most interpretable of all classification algorithms!)

Cons

- Prone to Overfitting
- Loss of information for continuous variables
- Complex calculation if there are many classes
- No guarantee to return the globally optimal decision
- Information gain: Bias for attributes with greater no. of values.

Variants of Decision Trees

ID3 is not the only (nor most popular) Decision Tree learner:

- **Oblivious Decision Trees** require the same attribute at every node in a layer
- **Random Tree** only uses a sample of the possible attributes at a given node
    - Helps to account for irrelevant attributes
    - Basis for a better Decision Tree variant: **Random Forest**

Summary

- Describe the basic decision tree induction method used in ID3
- What is information gain, how is it calculated and what is its primary shortcoming?
- What is gain ratio, and how does it attempt to overcome the shortcoming of information gain?
- What are the theoretical and practical properties of ID3-style decision trees?

Mitchell, Tom (1997). Machine Learning. Chapter 3: *Decision Tree Learning*.

Tan et al (2006) Introduction to Data Mining. Section 4.3, pp 150-171.