Decision trees

Inductive inference.

robust to noise in data and searches in complete hypothesis space.

approximating discrete valued functions

Representation:

A decision tree (DT)

A DT can be converted to sets of
if-then-else rules

Medical diagnosis to credit risk

In general DT represents a disjunction of conjunctions of constraints on the attribute values of instances.

Each branch represents an OR (disjuntion)

A path to leaf represents an AND (conjunction)

connect values of attributes using ANDs and ORs

```
         Sunny AND Normal
OR
         Overcast
OR
         Rain  AND Weak
```

The Algorithm.

Find the attribute that best classifies instances
Make a node for this attribute
Make branches for each value of this attribute
Sort training instances to each branch
        B_i gets all instances with
        value V_i for this attribute

Recur/Repeat for each branch with current set of instances at each branch

What is the best attribute?

We use a statistical measure called Information Gain defined in terms of Entropy

Given S = [9+,5-]

Entropy(S) = -P+logP+ - P-logP-

Log base 2.

= - 9/14 log(9/14) - 5/14 log (5/14)

= .940

```
Suppose S = [100+, 0-]

Entropy(S)
  = - 100/100 log 100/100 - 0/100 log 0/100

  = 0

Suppose S = [0+, 100-]

Entropy(S)
  = -0/100 log 0/100 - 100/100 log 100/100

  = 0

Suppose S = [50+, 50-]

Entropy S
  = - 50/100 log 50/100 - 50/100 log 50/100

  = -1/2 (-1) - 1/2 (-1)

  = 1/2 + 1/2

  = 1
```

```
In general:
For C target attribute values

Entropy(S) = Sum(1,c)[ -P_i log2 P_i ]

Information Gain:

Expected reduction in Entropy in we
partition examples on Attribute A

Gain(S,A) : Gain of an attribute A
relative to collection of
examples/(training instances) S, is
defined as:

Gain(S,A)
  = Entropy(S)
     - Sum [|S_v|/|S| * Entropy (S_v)]
```