Artificial Intelligence

Tutorial week 2 solution– Machine Learning

COMP3411/9814

1. Decision Trees

There are 3 objects in class '+' and 5 in '–', so the entropy is:

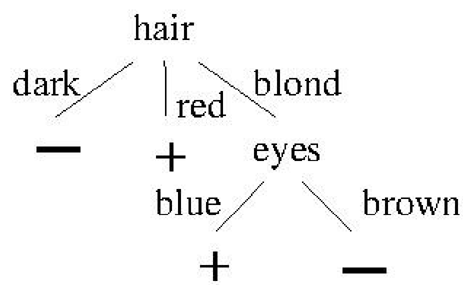
height	hair	eyes	hired
short	blond	blue	+
tall	red	blue	+
tall	blond	blue	+
tall	blond	brown	-
short	dark	blue	-
tall	dark	blue	-
tall	dark	brown	-
short	blond	brown	-

Entropy (parent) = $\Sigma_i P_i log_2 P_i = -(3/8)log(3/8) - (5/8)log(5/8) = 0.954$.
Suppose we split on height, [3, 5] parent would be divided to [1, 2] and [2, 3]:

- Of the 3 'short' items, 1 is '+' and 2 are '–', so Entropy (short) = $-(1/3)log(1/3) - (2/3)log(2/3) = 0.918$.
- Of the 5 'tall' items, 2 are '+' and 3 are '–', so Entropy (tall) = $-(2/5)log(2/5) - (3/5)log(3/5) = 0.971$.
- The average entropy after splitting on 'height' is: Entropy (height) = $(3/8)(0.918) + (5/8)(0.971) = 0.951$.
- The information gained by testing this attribute is: 0.954 - 0.951 = 0.003 (i.e. very little)

If we try splitting on 'hair' we find that the branch for 'dark' has 3 items, all '–' and the branch for 'red' has 1 item, in '+'. Thus, these branches require no further information to make a decision. The branch for 'blond' has 2 '+' and 2 '–' items and so requires 1 bit. That is, Entropy (hair) = (3/8)(0) + (1/8)(0) + (4/8)(1) = 0.5 and the information gained by testing hair is 0.954 - 0.5 = 0.454 bits. By a similar calculation, the entropy for testing 'eyes' is (5/8)(0.971) + (3/8)(0) = 0.607, so the information gained is: 0.954 - 0.607 = 0.347 bits. Thus 'hair' gives us the maximum information gain.

Since the 'blond' branch for hair still contains a mixed population, we need to apply the procedure recursively to these four items. Note that we now only need to test 'height' and 'eyes' since the 'hair' attribute has already been used. If we split on 'height', the branch for 'tall' and 'short' will each contain one '+' and one '–', so the entropy gain is zero. If we split on 'eyes', the 'blue' brach contains two '+'s and the 'brown' branch two '–'s, so the tree is complete:



For the last part, yes, a person with those characteristics would likely be hired.

2. The Laplace error estimate for pruning a node in a Decision Tree is given by:

$$E = 1 - \frac{n+1}{N+K}$$

where N is the total number of items, n is the number of items in the majority class and k is the number of classes. Given a subtree where parents are [4, 7], left child is [2, 1], and the right child is [2, 6], should the children be pruned or not? Show your calculations.
Error (Parent) = 1 - (7+1)/(11+2) = 1 - 8/13 = 5/13 = 0.385
Error(Left) = 1 - (2+1)/(3+2) = 1 - 3/5 = 2/5 = 0.4
Error(Right) = 1 - (6+1)/(8+2) = 1 - 7/10 = 3/10 = 0.3
Backed Up Error = (3/11)*(0.4) + (8/11)*(0.3) = 0.327 < 0.385
Since Error of Parent is larger than Backed Up Error –> Don't Prune.