

Tutorial 9:

Current Best Learning and Decision Trees

Question 1

Note:

“<nil> <=> Yes” is the most specific hypothesis possible, because it matches no patterns at all

a) Size of the hypothesis space.

We are only considering single hypotheses like the following:

<*, Warm, High, Strong, Cool, Change> <=> Yes

So the number of possible hypotheses of this type

= # possible Sky values (Sunny, Cloudy, Rain, *) * same for each other attribute or output.

= $4 \cdot 3^5 = 972$.

Remember: each has to match Yes on the output (here: EnjoySport).

There is also the <nil> <=> Yes hypothesis, which always produces a No (because no test pattern can match <nil>).

So the total number is 973.

With a little thought, we can see that these single hypotheses are by no means the only possible hypotheses on these kinds of space.

E.g.

(<Sunny, Warm, High, Strong, Cool, Change> or

<Cloudy, Warm, High, Strong, Cool, Change>) <=> Yes

has not been considered. But we don't consider this further in COMP3702/7702.

b) Ranking the hypotheses by specificity from most specific to least.

Here, specificity is based on the number of possible patterns which will match the hypothesis.

of matching patterns:

A = <*, *, High, *, *, * > <=> Yes. $\rightarrow 3 \cdot 2^4 = 48$

B = <*, *, Normal, Weak, *, Change > <=> Yes. $\rightarrow 3 \cdot 2 \cdot 2 = 12$

C = < Rain, Cold, High, Strong, Warm, Change > <=> Yes. $\rightarrow 1$

D = < nil > <=> Yes. $\rightarrow 0$

E = < *, Cold, Normal, Strong, Cool, Same > <=> Yes. $\rightarrow 3$

So: ranked hypotheses: D C E B A

c) Current best hypothesis learning:

H1 = <*, Cold, High, *, *, * > <=> Yes

Example 2: false negative \rightarrow generalise

H2 = <*, *, High, *, *, *> <=> Yes

Example 3: false positive -> specialise

H3 = <Sunny, *, High, *, *, *> <=> Yes

Example 4: classified correctly

H4 = no change (H3 again).

Question 2

Notes:

- When calculating the information gain of an attribute, you need to figure out how much information the data set as a whole contains, and then you figure out the amount of information that is left *after* classifying based on that attribute. In other words, the remainder is the quantity of information that that attribute failed to get rid of.
- When choosing an attribute, we want the one with largest gain. The first term in the gain equation is the same for all the attributes, so we actually want the attribute with the smallest remainder.
- At every node in the tree, ID3 makes a decision to divide based on the maximum quantity of entropy (information) it can get rid of in its current data set.
- Whenever you make a decision at a node in ID3, you're reducing the size of the dataset that you're considering (for any further branching).

a) Finding the attribute with greatest information gain (ie: smallest remainder).

There are 4 options: Height, Weight, Age, Gender.

Remainder(A) = sum (i over all possible values of attribute A) $(\frac{p_i+n_i}{p+n}) I(\frac{p_i}{p_i+n_i}, \frac{n_i}{p_i+n_i})$

Worth remembering: $I(0,1) = 0 = I(1,0)$ and that $I(0.5,0.5) = 1$.

Height:

$p_{Tall} = 10$; $n_{Tall} = 0$

$p_{Medium} = 1$; $n_{Medium} = 4$

$p_{Short} = 0$; $n_{Short} = 5$

Remainder(Height) = $\frac{10}{20} I(\frac{10}{10}, \frac{0}{10}) + \frac{5}{20} I(\frac{1}{5}, \frac{4}{5}) + \frac{5}{20} I(\frac{0}{5}, \frac{5}{5})$
= $0 + .25 * (-(\frac{1}{5}) * \log_2(\frac{1}{5}) - (\frac{4}{5}) * \log_2(\frac{4}{5})) + 0$
= 0.180

Weight:

$p_{Fat} = 3$; $n_{Fat} = 3$

$p_{Medium} = 1$; $n_{Medium} = 4$

$p_{Thin} = 7$; $n_{Thin} = 2$

Remainder(Weight) = $\frac{6}{20} I(\frac{3}{6}, \frac{3}{6}) + \frac{5}{20} I(\frac{1}{5}, \frac{4}{5}) + \frac{9}{20} I(\frac{7}{9}, \frac{2}{9})$
= $(\frac{6}{20}) * 1 + 0.18 + (\frac{9}{20}) * (-(\frac{7}{9}) * \log_2(\frac{7}{9}) - (\frac{2}{9}) * \log_2(\frac{2}{9}))$
= 0.824

Age:

$p_{\text{Young}} = 7 ; n_{\text{Young}} = 3$
 $p_{\text{Middleage}} = 2 ; n_{\text{Middleage}} = 2$
 $p_{\text{Old}} = 2 ; n_{\text{Old}} = 4$

$$\begin{aligned}
 \text{Remainder(Age)} &= 10/20 I(7/10,3/10) + 4/20 I(2/4,2/4) + 6/20 I(2/6,4/6) \\
 &= 0.5 * (- (7/10) * \log_2(7/10) - (3/10) * \log_2(3/10)) + 0.2 * 1 + 0.3 * (- (1/3) * \log_2(1/3) - \\
 &\quad (2/3) * \log_2(2/3)) \\
 &= 0.916
 \end{aligned}$$

Gender:

$p_{\text{Female}} = 7 ; n_{\text{Female}} = 4$
 $p_{\text{Male}} = 4 ; n_{\text{Male}} = 5$

$$\begin{aligned}
 \text{Remainder(Gender)} &= 11/20 I(7/11,4/11) + 9/20 I(4/9,5/9) \\
 &= (11/20) * (- (7/11) * \log_2(7/11) - (4/11) * \log_2(4/11)) + (9/20) * (- (4/9) * \log_2(4/9) - \\
 &\quad (5/9) * \log_2(5/9)) \\
 &= 0.966
 \end{aligned}$$

So Height has the smallest remainder and so the largest information gain and so is the best attribute for the top of the decision tree.

b) Finding the greatest information gain.

$$\text{Gain(A)} = I(p/(p+n), n/(p+n)) - \text{Remainder(A)}.$$

First need to find information content of entire data set
 $I(p/(p+n), n/(p+n)) = I(11/20, 9/20)$
 $= - (11/20) * \log_2(11/20) - (9/20) * \log_2(9/20)$
 $= 0.993$ bits per response (Sick = Yes or No).

Greatest information gain is for height, which had the smallest remainder

$$\text{Gain(Height)} = 0.993 - 0.180 = 0.813$$

c) Creating the rest of the decision tree.

Height=Tall branch:

- all examples have Sick=Yes, so just label that branch Yes.

Height=Medium branch:

#	Weight	Age	Gender	Sick ?
4	Med	Old	F	N
5	Fat	Young	M	N
8	Fat	Young	F	N
15	Thin	Old	F	Y
19	Fat	Old	F	N

Without doing any calculations, do any of these three attributes (Weight, Age, Gender) offer perfect discrimination on this set of examples?

Answer: yes: Weight does. (Note: Age and Gender don't).

So no need to calculate anything: branch here on Weight.

->

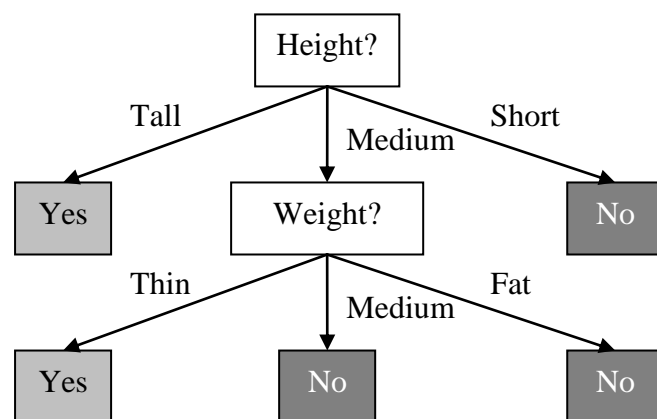
Weight= Med -> Sick = No

Weight= Fat -> Sick = No

Weight= Thin -> Sick = Yes

Height=Short branch

None of these people got sick, so just label that branch No.



d)

This decision tree is possibly not sufficient to warn everyone, as with only 20 examples, all possible combinations of the variables have not been experienced.

Also, there may be problems with dividing age, height, and weight into discrete categories.