Entropy_ID3_Exercise.pdf

by Derek Dilger, for the team

	Color	Shape	Size	class
1	red	square	big	+
2	blue	square	big	+
3	red	round	small	-
4	green	square	small	-
5	red	round	big	+
6	green	round	big	-

each column header is an attribute

each row is like a tuple. here's tuple #1

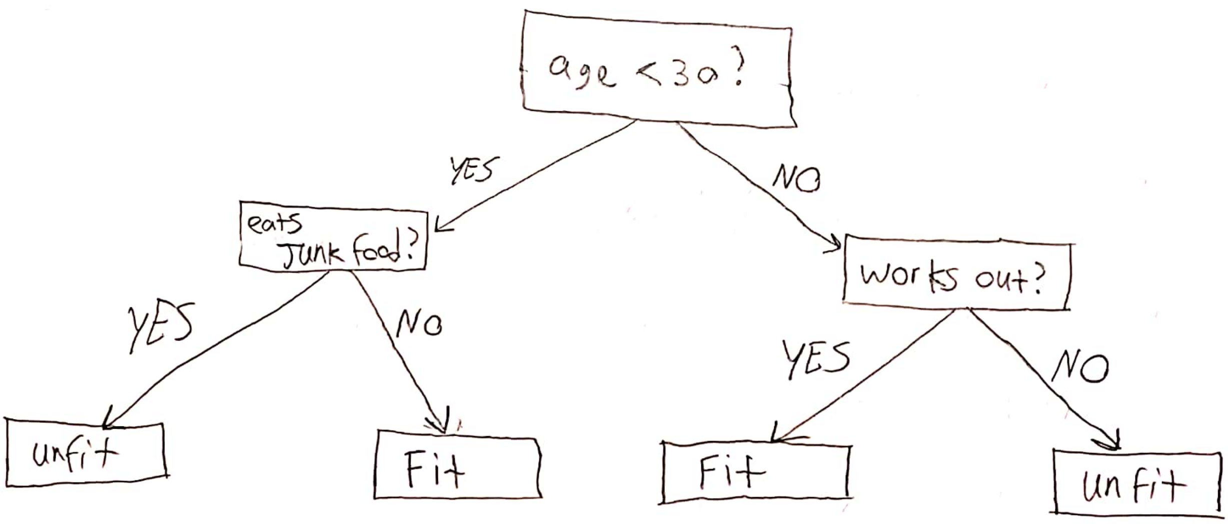
each tuple here has 4 elements, but this generalizes to n elements.

"data set"

this column is the "target attribute". Our target attribute may take on two values, "+" and "-".

Here's an example of what we're working towards. This example does not use our data.

Is the person fit?



How was this tree made? By using the ID3 algorithm. This small fitness example can be done by hand. But with large data sets, the ID3 algorithm is used to make a large tree.

Here we will make the decision tree one iteration at a time. I.D3 uses entropy to calculate which attribute is the most immediately discriminating. (as in, it's a greedy algorithm)

let our dataset be "S"

let n be the total number of classes in the target attribute (here n = 2. "+" or "-")

let P_i be the probability of being classed as "i", in other words, $\frac{\text{number of rows with class } i \text{ in the target column}}{\text{total number of rows}}$

let entropy be calculated as

$$\text{Entropy}(S) = -\sum_{i=1}^{n} P_i * \log_2(P_i)$$

let Information Gain be calculated as

$$IG(S, A) = \text{Entropy}(S) - \sum((|S_v| / |S|) * \text{Entropy}(S_v))$$

Where S_v is the set of rows in S for which the feature column A has value v, $|S_v|$ is the number of rows in S_v,

and $|S|$ is the number of rows in S.

Color	Shape	Size	class
red	square	big	+
blue	square	big	+
red	round	small	−
green	square	small	−
red	round	big	+
green	round	big	−

$$\text{Entropy}(S) = -\left(\frac{3}{6}\right) * \log_2\left(\frac{3}{6}\right) - \left(\frac{3}{6}\right) * \log_2\left(\frac{3}{6}\right) = -\left(\frac{1}{2}\right)(-1) - \left(\frac{1}{2}\right)(-1) = 1.0$$

Now calculate IG for each feature (i.e. find which is most discriminating)

IG for Color:

$|S| = 6$ (total rows)

For $v = \text{red}$, $|S_v| = 3$ ← (red rows)

$\text{Entropy}(S_v) = -\left(\frac{2}{3}\right)\left(\log_2 \frac{2}{3}\right) - \left(\frac{1}{3}\right)\left(\log_2 \frac{1}{3}\right) = 0.9182958$

For $v = \text{blue}$, $|S_v| = 1$

$\text{Entropy}(S_v) = -\left(\frac{1}{1}\right)\left(\log_2 \frac{1}{1}\right) - \underbrace{\left(\frac{0}{1}\right)\cancel{\left(\log_2 \frac{0}{1}\right)}}_{\text{nonsense artifact of formula}} = 0$

note zero entropy because "blue", with it's 1 sample size, appears to predict "class" (perfectly).

For $v = \text{green}$, $|S_v| = 2$

$\text{Entropy}(S_v) = 0$ ← zero, like blue because both greens predict "−" exactly and only.

$$IG(S, \text{Color}) = \text{Entropy}(S) - (|S_{red}|/|S|)(\text{Entropy}(S_{red})) - (|S_{blue}|/|S|)(\text{Entropy}(S_{blue})) - (|S_{green}|/|S|)(\text{Entropy}(S_{green}))$$

$$= 1.0 - \left(\frac{3}{6}\right)(0.9182958) - (0) - 0 = \boxed{0.54085}$$

IG for Shape:

Coincidentally same as Entropy(S_{red})

For v = square, $|S_v| = 3$

$$\text{Entropy}(S_v) = -\left(\tfrac{2}{3}\right)\left(\log_2\left(\tfrac{2}{3}\right)\right) - \left(\tfrac{1}{3}\right)\left(\log_2\left(\tfrac{1}{3}\right)\right) = 0.9182958$$

The computations are laborious and excessive. The general formulas given describe fully how to calculate each value as needed for input into IG(S, class). The rest are done on calculator. Edge cases have also been demonstrated, and how to deal with them.

Overall so far, we have

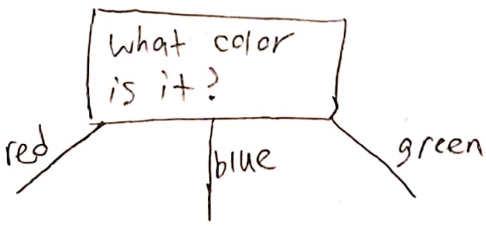
$$IG(S, \text{color}) = 0.54085$$

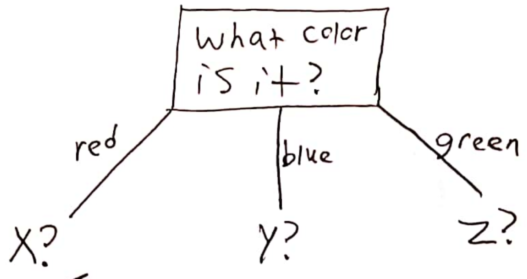
$$IG(S, \text{Shape}) = 0.08170$$

$$IG(S, \text{Size}) = 0.459148$$

So therefore we choose to discriminate based on color to begin with, as it has the highest information gain.

The tree so far is:





So what question do we put here?

We ask about the attribute with the highest Information gain, just as before.

However, now our dataset has changed to contain only tuples with color==red.

i.e

Color	Shape	Size	Class
red	square	big	+
red	round	small	−
red	round	big	+

S = dataset for "X?"

IG(S, color) = 0.811278

IG(S, shape) = 1.0

IG(S, size) = 1.5

So we ask about size at "X?"

dataset for "Y?"

Color	Shape	Size	Class
blue	square	big	+

note this degenerate dataset with one tuple. We don't ask a question here, but rather, conclude that the class is "+". The tuple correlates exactly with class's outcome.

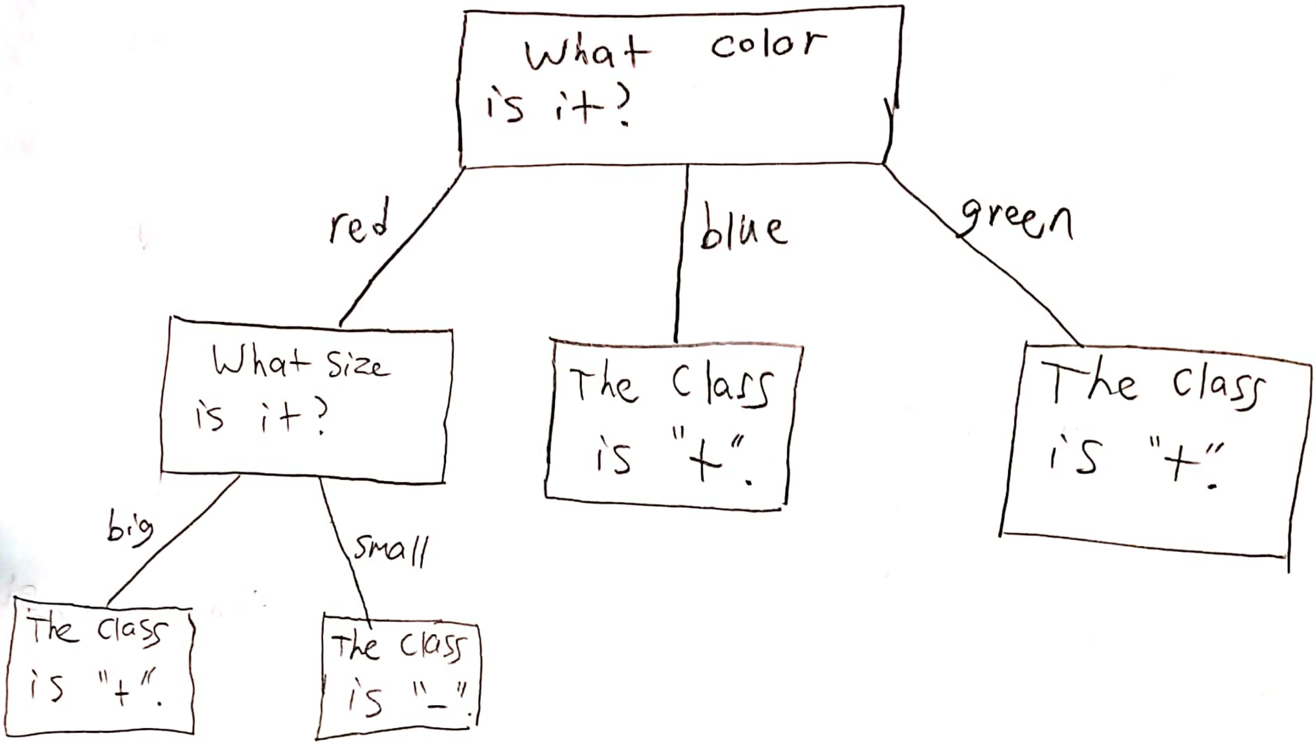
dataset for "Z?"

Color	Shape	Size	Class
green	square	small	—
green	round	big	—

degenerate. Perfect correlation between color and class.

We do the same for blue & green, Y? and Z?.

Now the decision tree is:



Note that

(red and big) = Class "+"

(red and small) = Class "_"

for all tuples.
(from the previous table. It would be computed as before.)
So we have classified everything.

<end of part (i) in project>

(ii) If "pattern of shirt" attribute is added, only a few possibilities may occur. Either it offers more information gain at a certain step of classification, or it does not.
In the case it does, it may or may not allow us to classify using fewer steps. In the case it does not, we will simply use the tree as shown, because the shirts didn't help.

With a data size of millions of shirts, almost certainly the shirt color attribute would help classify at some step in the tree.* This decision would influence millions of dollars of revenue.

If the manager and CEO are adequately paying attention (doing their job), then certainly the data scientist would make an impact.

*(unlike in our dwarf tree with small dataset)