Answers

Friday, October 7, 2022 12:23 AM

1. Theoretically, a dataset at a particular node that has all instances with same label already provides us with the classfication task on-hand. Any further splits won't provide us with a different prediction (the majority class will not change however we try to branch the tree, thus will result in same prediction). And according to our hypothesis/assumption: the simplest tree that classifies the training instances accurately will generalize. As "Occam's razor" quote: "Entities should not be multiplied beyond necessity" with the simplist hypothesis being better.

   Technically, any split from this root or subtree node will give zero information gain, and thus satisfy the third base case condition (as per the algorithm instructions).

h_1 hypothesis – mark the node with all same label instances as leaf node with prediction being that label.

h_2 hypothesis – choose best split threshold value c among all possible splits (cross-feature to generalize).
Any such split will result in entropy zero.

let $Y = \{+, -\}$ represent random distribution of binary class.
$S_1, S_2 \in S$ split subsets (of a particular c threshold split)

$$H_S(Y \mid S) = -\sum_{s \in S} \frac{|s|}{|S|} H(s) = -\left[\frac{|S_1|}{|S|} H_S(S_1)\right] - \left[\frac{|S_2|}{|S|} H_S(S_2)\right]$$

$$H_S(Y) = -\sum_{y \in Y} P(y) \log_2 P(y)$$

either total, S_1, or S_2

we know that all elements are the same, then either $P_+ = 0 \wedge P_- = 1$ or $P_- = 0 \wedge P_+ = 1$

$\Downarrow$ then in either case:

$$H_S(Y) = -[1 \log 1] - [0 \log 0] = 0 \quad \text{by definition.}$$

either total, S_1, or S_2

$\Downarrow$

↓

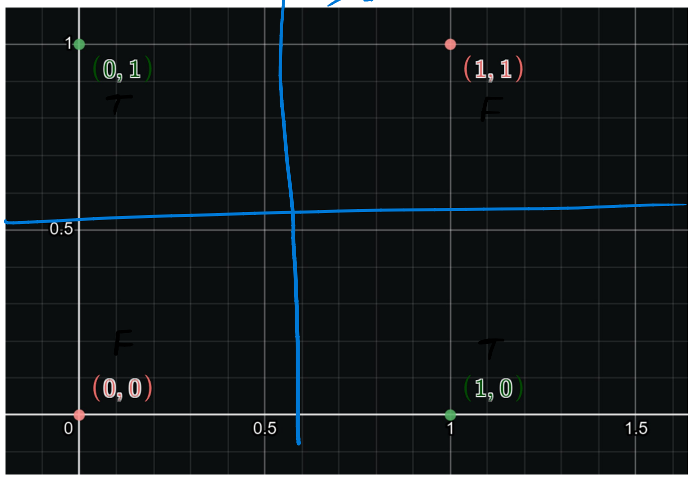
any split will result in combined entropy of zero and subsequently info gain ratio of 0.

2.

```
dataset > refuseToSplitDataset.txt
x1   1 0 1 T
x2   1 1 0 F
x3   0 1 1 T
x4   0 0 0 F
```

instances   feature 1   feature 2   labels

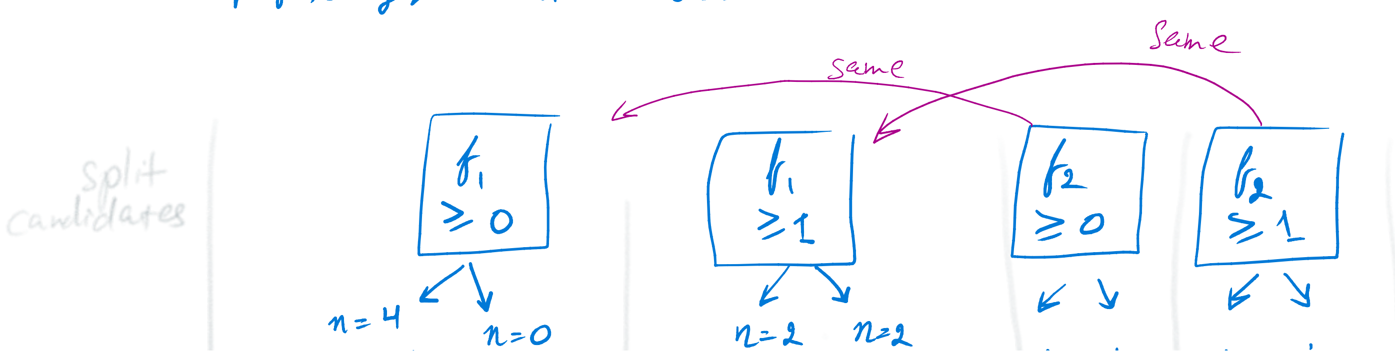
f1 split ≥ 1



f2 split ≥ 1

Any dataset that falls into one of the base cases (Gain Ratio of zero; Entropy of any candidate split is zero) will prevent spliting of the node.
To achieve the 2nd requirement that the algorithm will continue to split if forced, we can replicate similar split results as a different feature.
Because the InfoGain = 0   in all split candidates the algorithm is inevitably going to stop on the parent node.
As seen in the plot if we force split on f1 with c=1 then on the next iteration the decision tree algorithm wil lsplit on feature 2 where each subset will have one true label and one false label.

choices of splits by feature & threshold :



	$n=4$ $+:2, -:2$; $n=0$	$n=2$ $+:1, -:1$	$n=2$ $+:1, -:2$	⋮ ⋮	⋮	⋮
Entropy: of split	no split	1	1	no split	1	1
Info Gain:		$H_D(Y) = 1$ Initial entropy $H_D(Y\|S) = -\sum_{s_1, s_2} \frac{1}{2} \cdot 2$ $\Downarrow$ $InfoGain(D, S) = 1 - 1 = 0$			0	

3. (Not sure I understood the instructions, therefore I posted more information to be comprehensive)

↓ Root node candidates:

[Entropy(parent)=0.845351 Entropy(leftChild)=0.845351 Entropy(rightChild)=0.000000]
Feature=0 Threshold=0.000000 GainRatio=0.000000 InfoGain=0.000000

[Entropy(parent)=0.845351 Entropy(leftChild)=0.000000 Entropy(rightChild)=0.881291]
Feature=0 Threshold=0.100000 GainRatio=0.100518 InfoGain=0.044177

[Entropy(parent)=0.845351 Entropy(leftChild)=0.845351 Entropy(rightChild)=0.000000]
Feature=1 Threshold=-2.000000 GainRatio=0.000000 InfoGain=0.000000

[Entropy(parent)=0.845351 Entropy(leftChild)=0.881291 Entropy(rightChild)=0.000000]
Feature=1 Threshold=-1.000000 GainRatio=0.100518 InfoGain=0.044177

[Entropy(parent)=0.845351 Entropy(leftChild)=0.764205 Entropy(rightChild)=1.000000]
Feature=1 Threshold=0.000000 GainRatio=0.055954 InfoGain=0.038275

[Entropy(parent)=0.845351 Entropy(leftChild)=0.811278 Entropy(rightChild)=0.918296]
Feature=1 Threshold=1.000000 GainRatio=0.005780 InfoGain=0.004886

[Entropy(parent)=0.845351 Entropy(leftChild)=0.863121 Entropy(rightChild)=0.811278]
Feature=1 Threshold=2.000000 GainRatio=0.001144 InfoGain=0.001082

[Entropy(parent)=0.845351 Entropy(leftChild)=0.918296 Entropy(rightChild)=0.721928]

corresponding tree:

infoGain=0.189053 Feature=1 Threshold=8.000000 GainRatio=0.430157 [+:3 , -:8]

- LEFT
  infoGain=0.000000 Leaf with label: true [+:1 , -:0]
- RIGHT
  infoGain=0.034144 Feature=0 Threshold=0.100000 GainRatio=0.072802 [+:2 , -:8]
  - LEFT
    infoGain=0.000000 Leaf with label: false [+:0 , -:1]
  - RIGHT
    infoGain=0.000000 Leaf with label: false [+:2 , -:7]

Feature=1 Threshold=3.000000 GainRatio=0.016411 InfoGain=0.016313

[Entropy(parent)=0.845351 Entropy(leftChild)=0.970951 Entropy(rightChild)=0.650022]
Feature=1 Threshold=4.000000 GainRatio=0.049749 InfoGain=0.049452

[Entropy(parent)=0.845351 Entropy(leftChild)=1.000000 Entropy(rightChild)=0.591673]
Feature=1 Threshold=5.000000 GainRatio=0.111240 InfoGain=0.105196

[Entropy(parent)=0.845351 Entropy(leftChild)=0.918296 Entropy(rightChild)=0.543564]
Feature=1 Threshold=6.000000 GainRatio=0.236100 InfoGain=0.199587

[Entropy(parent)=0.845351 Entropy(leftChild)=1.000000 Entropy(rightChild)=0.764205]
Feature=1 Threshold=7.000000 GainRatio=0.055954 InfoGain=0.038275

[Entropy(parent)=0.845351 Entropy(leftChild)=0.000000 Entropy(rightChild)=0.721928]
Feature=1 Threshold=8.000000 GainRatio=0.430157 InfoGain=0.189053

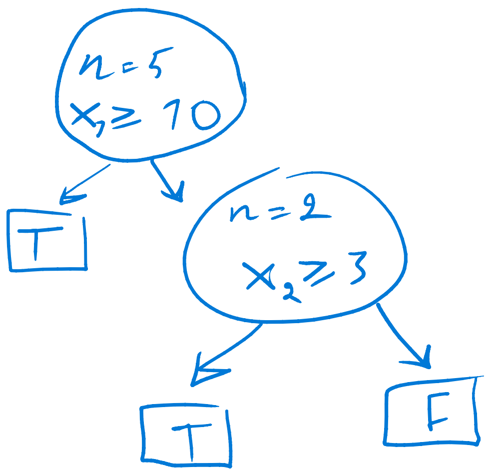
↓ Children iterations candidates:

[Entropy(parent)=0.000000 Entropy(leftChild)=0.000000 Entropy(rightChild)=0.000000]
Feature=0 Threshold=0.000000 GainRatio=0.000000 InfoGain=0.000000

[Entropy(parent)=0.721928 Entropy(leftChild)=0.721928 Entropy(rightChild)=0.000000]
Feature=0 Threshold=0.000000 GainRatio=0.000000 InfoGain=0.000000

[Entropy(parent)=0.721928 Entropy(leftChild)=0.000000 Entropy(rightChild)=0.764205]
Feature=0 Threshold=0.100000 GainRatio=0.072802 InfoGain=0.034144

4.



⇐

infoGain=0.321928 Feature=0 Threshold=10 GainRatio=0.331560 [+:4 , -:1]

- LEFT
  infoGain=0.000000 Leaf with label: true
- RIGHT
  infoGain=1.000000 Feature=1 Threshold=3 GainRatio=1.000000 [+:1 , -:1]
  - LEFT
    infoGain=0.000000 Leaf with label: true
  - RIGHT
    infoGain=0.000000 Leaf with label: false

given an input instance x, assign label as follows:

if $x \geq 10$

given an input instance x, assign label as

if $x_1 \geq 10$

  $x \to T$

else if $x_2 \geq 3$

  $x \to T$

else

  $x \to F$

5.

h_{01}

infoGain=0.669016
Feature=1 Threshold=0.201829 GainRatio=1.000000
[+:825 , -:175]

- LEFT
  infoGain=0.000000 Leaf with label: true [+:825 , -:0]
- RIGHT
  infoGain=0.000000 Leaf with label: false [+:0 , -:175]

The decision tree splits the data precisely into 2 subsets each with uniform class type. Thus, it models the problem through one condition x >= 0.201829 ; satisfying instance are labeled true, otherwise they are labeled as false.
As the feature split is the second in order we can say that the area above (and including) the line parallel to x-axis x = 0.201829 includes positive examples while below it are negative examples.

h_{02}

infoGain=0.080135
Feature=1 Threshold=0.919236 GainRatio=0.210507
[+:492 , -:508]

- LEFT
  infoGain=0.000000 Leaf with label: true [+:74 , -:0]
- RIGHT
  infoGain=0.269671 Feature=0 Threshold=0.533076 GainRatio=0.269904 [+:418 , -:508]
  - LEFT
    infoGain=0.000000 Leaf with label: true [+:338 , -:109]
  - RIGHT
    infoGain=0.000000 Leaf with label: false [+:80 , -:399]

Note: The specification doesn't clearly state wether the implementation should allow spliting on features more than one, therefore I've done mine to create a tree of max depth of # of features. This is one of the parts I found vague about the instructions, as there can be different levels of accuracy depending on the objective of the model use case.

Interpretation of D2: The decision tree indicates that all instances above $x_1 = 0.919$ should be labeled true/positive. While if the value is below $x_1 = 0.919$ the we consider the other attribute $x_0 = 0.533$, anything larger than it we can consider true, while smaller value are mostly false;
(If the implementation allows for further bounding of features in nested trees, then defenitely the tree would be much larger than the case for D1 dataset, as entropy is present and the split is not uniform.)

6.