### Course: CS420 - Artificial Intelligence

06 – ID3 Decision trees and MLP

**Question 1.** Roger Federer is one of the greatest tennis players since tennis have been invented. We want to learn a little about what makes Federer win or lose a match. To do so, we gathered data from games played by R. Federer, shown in the below table.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID | Time | Match Type | Court Surface | Best Effort | Outcome |
| 1 | Morning | Master | Grass | True | Win |
| 2 | Afternoon | Grand slam | Clay | True | Win |
| 3 | Night | Friendly | Hard | False | Win |
| 4 | Afternoon | Friendly | Mixed | False | Lose |
| 5 | Afternoon | Master | Clay | True | Lose |
| 6 | Afternoon | Grand slam | Grass | True | Win |
| 7 | Afternoon | Grand slam | Hard | True | Win |
| 8 | Afternoon | Grand slam | Hard | True | Win |
| 9 | Morning | Master | Grass | True | Win |
| 10 | Afternoon | Grand slam | Clay | True | Lose |
| 11 | Night | Friendly | Hard | False | Win |
| 12 | Night | Master | Mixed | True | Lose |
| 13 | Afternoon | Master | Clay | True | Lose |
| 14 | Afternoon | Master | Grass | True | Win |
| 15 | Afternoon | Grand slam | Hard | True | Win |
| 16 | Afternoon | Grand slam | Clay | True | Win |

1. Build a classification model using ID3 decision tree from the gathered data.

The entropy of the whole dataset

* H(Dataset) = -11/16\*log211/16 – 5/16\*log25/16 = 0.896 (11 Win – 5 Lose)

The information gain of the attribute Time

* H(Time = Morning) = 0 (2 Win – 0 Lose)
* H(Time = Afternoon) = -7/11\*log27/11 – 4/11\*log24/11 = 0.946

(7 Win – 4 Lose)

* H(Time = Night) = -2/3\*log22/3 – 1/3\*log21/3 = 0.918 (2 Win – 1 Lose)
* AE(Time) = 2/16\*0 + 11/16\*0.946 + 3/16\*0.918 = 0.823
* IG(Time) = 0.896 – 0.823 = 0.073

The information gain of the attribute Match Type

* H(Match Type = Master) = 1 (3 Win – 3 Lose)
* H(Match Type = Grand slam) = -6/7\*log26/7 – 1/7\*log21/7 = 0.592

(6 Win – 1 Lose)

* H(Match Type = Friendly) = -2/3\*log22/3 – 1/3\*log21/3 = 0.918

(2 Win – 1 Lose)

* AE(Match Type) = 6/16\*1 + 7/16\*0.592 + 3/16\*0.918 = 0.806
* IG(Match Type) = 0.896 – 0.806 = 0.09

The information gain of the attribute Court Surface

* H(Court Surface = Grass) = 0 (4 Win – 0 Lose)
* H(Court Surface = Hard) = 0 (5 Win – 0 Lose)
* H(Court Surface = Clay) = -2/5\*log2(2/5) – 3/5\*log23/5 = 0.971

(2 Win – 3 Lose)

* H(Court Surface = Mixed) = 0 (0 Win – 2 Lose)
* AE(Court Surface) = 5/16\*0.971 = 0.303
* IG(Court Surface) = 0.896 – 0.303 = 0.593

The information gain of the attribute Best Effort

* H(Best Effort = True) = -9/13\*log29/13 – 4/13\*log24/13 = 0.890

(9 Win – 4 Lose)

* H(Best Effort = False) = -2/3\*log22/3 – 1/3\*log21/3 = 0.918

(2 Win – 1 Lose)

* AE(Best Effort) = 13/16\*0.890 + 3/16\*0.918 = 0.895
* IG(Best Effort) = 0.001

The root attribute will be Court Surface due to its largest IG.

A picture containing sitting, table, computer, dark

Description automatically generated

**Repeat for every branch of the root attribute that has examples not fully classified into a single class**

Consider the branch Court Surface = Clay

* H(DCourt Surface = Clay) = 0.971

The information gain of the attribute Time

* AE(Time, DCourt Surface = Clay) = 0.971
* IG(Time, DCourt Surface = Clay) = 0.971 – 0.971 = 0

The information gain of the attribute Match Type

* H(Match Type = Master) = 0 (0 Win – 2 Lose)
* H(Match Type = Grand slam) = -2/3\*log22/3 – 1/3\*log21/3 = 0.918

(2 Win – 1 Lose)

* AE(Match Type) = 2/5\*0 + 3/5\*0.918 = 0.551
* IG(Match Type) = 0.971 – 0.551 = 0.42

The information gain of the attribute Best Effort

* AE(Best Effort, DCourt Surface = Clay) = 0.971
* IG(Best Effort, DCourt Surface = Clay) = 0.971 – 0.971 = 0

Thus, the chose attribute is Match Type

A picture containing text, map

Description automatically generated

Consider the branch Court Surface = Clay → Match Type = Grand slam.

For Time and Best Effort, each attribute has a single value. Thus, it is pointless to continue to grow the tree. Instead, we use the rule of Majority Voting.

A picture containing computer

Description automatically generated

1. Knowing the conditions in which a tennis match takes place, we would like to predict whether R. Federer will win or lose the match, using ID3 decision tree in a.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Time | Match type | Court Surface | Best Effort | Outcome |
| Morning | Grand Slam | Grass | Yes | ? |
| Afternoon | Friendly | Clay | No | ? |

The first example: From the root Court Surface, follow the branch Court Surface = Grass to the conclusion Outcome = Win

The second example: From the root Court Surface, follow the branch Court Surface = Clay to the attribute Match Type. We have no such branch Match Type = Friendly there. What should we do? Choose either of the following two solutions

* Conclude that Outcome cannot be decided due to the lack of training data
* Follow Rule 3 (Lecture 09, slide 40) to create a default value: Among 16 examples used to construct the root node, there are 3 examples that have Match Type = Friendly. Among those 3 examples, there are 2 Win and 1 Lose. Thus, the label created for Court Surface = Clay → Match Type = Friendly is Win.

**Question 2.** A perceptron has two input units, a unipolar step function, weights and , and a threshold (note that, θ can be considered as a weight for an extra input which is always set to ).

1. The actual output for the input pattern is step[ 0.2\***1** + (-0.5)\***1** + 0.2\*(**-1**) ] = 0
2. The perceptron is now trained using the learning rule , where is the input vector, is the learning rate, is the weight vector, is the desired output, and is the actual output. What are the new values of the weights and threshold after one step of training with the input vector and desired output 1, with ?

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
| 0 | 0.2 | -0.3 | 0 |

1. Show the equation of the decision line for the perceptron **before** training

1. Show the equation of the decision line for the perceptron **after** training with the input vector in b.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Question 3.** In the network shown aside, all the units have binary inputs (0 or 1), unipolar step functions and binary outputs (0 or 1). The weights for this network are and . The threshold of the hidden unit (3) is 1.5 and the threshold of the output unit (4) is 0.5. The threshold of both input units (1 and 2) is 0.5, so the output of these units is the same as the input.  Compute the output at every neuron for all pairs of input values.   |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | |  |  | Neurons | | | | | 1 | 2 | 3 | 4 | | 0 | 0 | 0 | 0 | 0 | 0 | | 0 | 1 | 0 | 1 | 0 | 1 | | 1 | 0 | 1 | 0 | 0 | 1 | | 1 | 1 | 1 | 1 | 1 | 0 | | Diagram  Description automatically generated |

Which Boolean functions can be computed by this network? XOR(

**Question 4.** A single-layer network of perceptrons has 3 input units and 3 output units. No threshold (or bias) is considered.

The network has 3\*3 = 9 weights.

**Question 5.** A multi-layer feedforward network has 5 input units, a first hidden layer with 4 units, a second hidden layer with 3 units, and 2 output units.

If no threshold (or bias) is used the network has 5\*4+4\*3+3\*2 = 38 weights.

If threshold (bias) is used, the network has [(5+1)\*4]+[(4+1)\*3]+[(3+1)\*2] = 38 weights.

|  |  |
| --- | --- |
| **Question 6.** Suppose you have the following three-layered multi-layer neural networks. No threshold (or bias) is considered. The pattern to be learned is with desired output . The learning rate is 0.25.  The weight vectors are | Shape  Description automatically generated with low confidence |

1. Compute the output at every neuron when the pattern is propagated through the net.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| A | B | C | D | E |
| 0 | 1 | 0.634 | 0.458 | 0.644 |

1. Adjust the weights when the backpropagation takes place.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |
| 0.62 | 0.547 | 0.42 | -0.177 | 0.327 | 0.793 |