KEELE UNIVERSITY

Open Book Assessment,

2023/24 FHEQ Level 5 January 2024

COMPUTER SCIENCE

CSC-20043

COMPUTATIONAL AND ARTIFICIAL INTELLIGENCE I

STUDENT NO. 22018575

Table of Contents

1.1.0 Defining Intelligence
Intelligence is measuring the rate of an entity to acquire new knowledge, organise that knowledge and recombine it with old knowledge to produce a new and more complex understanding. This allows for more complex problem-solving ability.

An example is teaching mathematics to young children - they are first taught addition and then multiplication. Recombining that knowledge shows that multiplication is just repetitive addition of the same number for N times. Therefore, this show the rate at which they acquire this knowledge is their intelligence.

1.1.1 Defining Artificial Intelligence
Artificial Intelligence is humanity's pursuit of creating an entity in their own image to better understand themselves. It is a branch of computer science focused on creating machines and systems which replicate a human's ability to perform complex tasks that cannot be hard-coded.

1.1.2 Defining an Agent
An Agent is an independent entity that gathers information from an environment, processes the information and then performs a rational action in that environment based on maximizing performance.

1.1.3 Defining Rationality
Rationality is when an action or decision is made on the current information available using a set of logic (rules) to achieve an optimal and desired outcome.

1.1.4 Defining Logical Reasoning
Logical reasoning is when an action or decision is taken using a set of formal logic (rules) which is given from the current available information.

1.2.0 "Surely computers cannot be intelligent—they can do only what their programmers tell them." Is the latter statement true, and does it imply the former?
Firstly, the statement shows a lack of confidence in asserting itself as fact and is more a question asking for validation of itself. Breaking down the first part of the statement, the speaker is likely combining the idea of intelligence and consciousness to be the same thing. Intelligence is whether an entity can acquire a new knowledge, organise that knowledge and recombine it with old knowledge to solve a problem with a satisfactory outcome. Humans have achieved this through techniques, such as machine learning algorithms and Neural Networks which simulate the process of learning in humans emulating intelligence. An example of this being 'Image Recognition' where distinguishing between cats and dogs is examined. This differs from the idea of consciousness; these entities do not exhibit a consciousness (or in other words, have wants and dreams). They simply intake and output data based on logic (a set of rules).

The second half of the statement is incorrect. With current techniques, a program can be given a set of parameters for the output that a programmer wants. The way it achieves this is not pre-determined by what the programmer tells the program to do. The program learns this by itself. An example of this is described in Lecture two of 'Evolutionary Algorithms CSC-

20043' where Thompson's hardware evolution experiment: ‘Evolved a Field Programmable Gate Array (FPGA) to discriminate between 1kHz and 10kHz tones’ is discussed. Thompson’s FGPA unintentionally became a radio receiver, and some circuit would only propagate on electrical signals at certain parts of the day whilst being a silent broadcast at other parts. Therefore, showing that machines can evolve to achieve a task beyond what was programmed by the programmer.

.2.1. “Surely animals cannot be intelligent—they can do only what their genes tell them.” Is the latter statement true, and does it imply the former?

The relationship of the statement above states that the genes directly determine all animals’ actions based on a set of rules in the genome, making them unable to learn new behaviours. This statement can be discredited by Pavlov’s dog experiment. This experiment shows the capability of dogs to learn new behaviours and be conditioned to react to external stimuli, such as a bell to signal food. A bell is man-made device and not found in nature.

The Baldwin effect - ‘A New factor in evolution’ - also discredits the statement. Its effect indicates that intelligences are not solely determined through genetic instructions but through a combination of genetic and learned factors, which highlight the capability to learn and for animals to interact with their environment. A quote to contrast this statement is from Darwin’s 1871 ‘Descent of Man’: "Nevertheless the difference in mind between man and the higher animals, great as it is, certainly is one of degree and not of kind.” If both statements were correct, then humans would also simply follow what their ‘genes’ tell them to do, as we are in fact also considered animals. Therefore, the statement is untrue.

1.3.1 A Political Parties neural network

- Area of the country (Scotland, Wales, Nothern Ireland and England). 4 Categorical nodes.
- Average age of voters. 1 Continuous node.
- % votes received by the party at last general election. 1 Continuous node.
- Whether the part won in constituency at the last general election. 1 Discrete node (Binary).
- The % of people who say they will vote for the party at next general election based on a recent opinion poll. 1 Continuous node.

There are 8 Nodes in total.

1.3.1.1 Normalisation

- Area of the country should use One hot encoding.
- Voter's age should be normalized using linear normalisation.
- % votes received by the party at last general election and the % of people who say they will vote for the party at next general election based on a recent opinion poll. Should be normalized using log normalisation.

1.3.2 Calculating Fitness

The neural network role is to predict the percentage of votes that a political party will receive in each electoral area. There are two previous general elections with 1300 examples.
To calculate the fitness, the NN predictions are compared to the real-world results.

Fitness = Number of correct predictions across the entire dataset (maximise fitness).
This achieves the goal of creating a neural network to predict percentage votes that a political party will receive in each electoral area by deeming the network with the highest correct predictions to be the more fit network. This approach maximises the number of correct predictions.

1.3.3 Selection

I propose a Steady-State selection method.
In political campaigns, there can be a range of both unpredictable and diverse factors which affect voting outside of the party's hand. I propose a Steady State selection because it maintains diversity in the population and allows for continuous evolution. This allows for more solutions to be available at a given time as a dramatic world event could occur (e.g. COVID-19 pandemic greatly affecting results). Therefore, having more diverse solutions available could help in future-proofing the A.I. and prevent having to re-train it if major changes occur. Steady State is also a balanced approach, allowing for both exploitation and exploration of a population to happen.

Steady state selection works by evaluating three individuals eliminating the least fit individual and using the two fittest to reproduce a new individual – as illustrated below in figure 1:

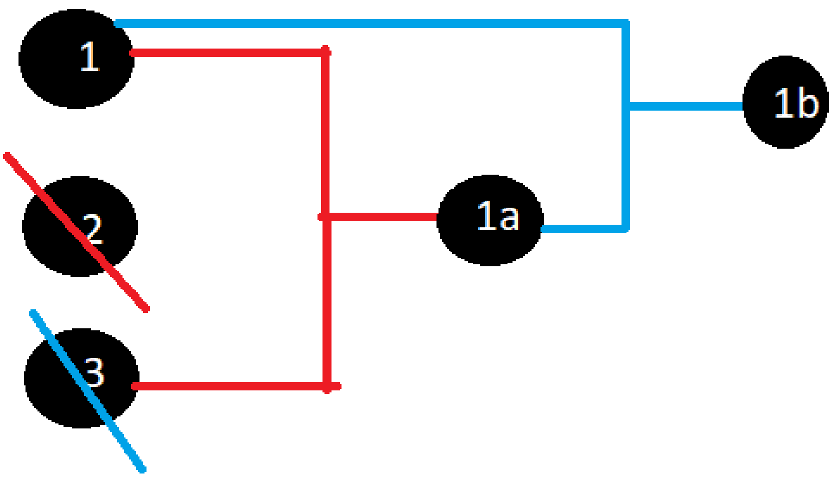


Figure 1

2.1.0 Perceptron Learning Rule

The Perceptron Learning Rule is a supervised learning algorithm and used on single layer perceptron. It is an algorithm that adjusts the weights of the input connections for making binary classifications. The below figure is an example of the algorithm.

```
REPEAT for each training example{
    IF ( out != t ) {          // output value != target value t
        θ = θ - η(t - out)          // update threshold θ
        w_i = w_i + η (t - out) a_i          // update each weight w_i
} UNTIL (network output accurate on all patterns OR max epochs)
```

Figure 2 – From CSC-20043 lecture slides

Limitations

The Perceptron Learning Rule can only be used on linearly solvable problems and fails on nonlinear problems. An example includes the XOR problem as it is impossible to classify with a single layer (SLP) and requires multiple layers (MLP) instead. However, an advantage is that it guarantees convergence in linearly separable data.

2.2.0 AND Gate

For an AND gate, a 1 should only be outputted when a1 and a2 are both 1. This means that we must find a threshold value that is only activated when both inputs are 1, which can be represented as: if $\Sigma >= \theta$ then output 1, else output 0. Any threshold between $1< \theta<=2$ could be considered suitable.

In the example below, 1.5 is chosen. The weights chosen are 1, but any combination weight that can meet the equation is also valid. $W1*1 + W2*1 >= \theta$ (threshold value).

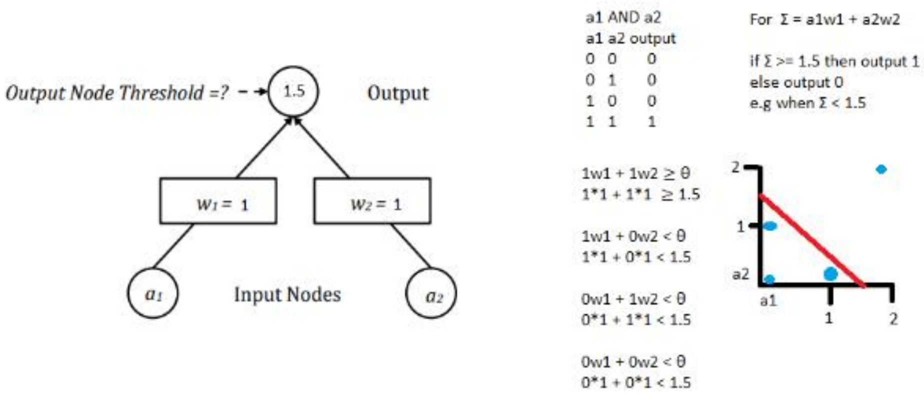
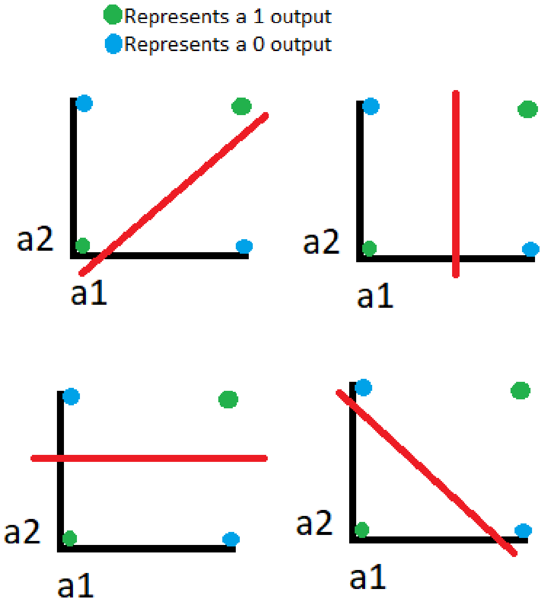


Figure 3

2.2.1 XOR Gate

XOR is not linearly solvable. This can be seen in the graph below as a line or hyperplane cannot be drawn to separate the two output classes 0 and 1.



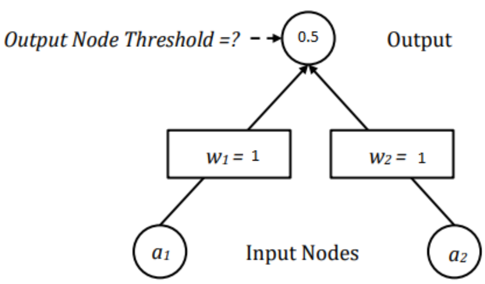
a1 XOR a2

a1	a2	output
0	0	1
0	1	0
1	0	0
1	1	1

Figure 4

2.2.2 OR Gate

For an OR gate, a 1 should only be outputted when either a1 or a2 has a 1 or both. Therefore, the threshold value should only be activated when at least one input has a 1. This can be represented as: if Σ >= θ then output 1, else output 0. The threshold value can lie between 0> θ <1. The weights chosen are 1, but any combination weight that can meet the equation is also valid. W1*1 + W2*0 >= θ (threshold value).



a1 OR a2

a1	a2	output
0	0	0
0	1	1
1	0	1
1	1	1

$1w1 + 1w2 \geq \theta$
$1*1 + 1*1 \geq 0.5$

$1w1 + 0w2 \geq \theta$
$1*1 + 0*1 \geq 0.5$

$0w1 + 1w2 \geq \theta$
$0*1 + 1*1 \geq 0.5$

$0w1 + 0w2 < \theta$
$0*1 + 0*1 < 0.5$

For Σ = a1w1 + a2w2

if Σ >= 0.5 then output 1 else output 0
E.g when Σ < θ

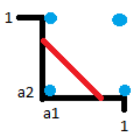


Figure 5

2.3 SLP in problems that are not linearly separable

No, it is impossible to use a SLP to solve problems that are not linearly separable. This can be seen in attempting to solve the XOR gate. XOR is not linearly separable and cannot be solved as seen in above figure.

2.4 The Sigmoid activation function

The sigmoid function is smooth and continuous, allowing for derivatives. The main advantage of this allows us to classify data in a continuous way instead of using the step activations binary method of 1 or 0. Therefore, the line can be model as a quadratic allowing for better classification of data

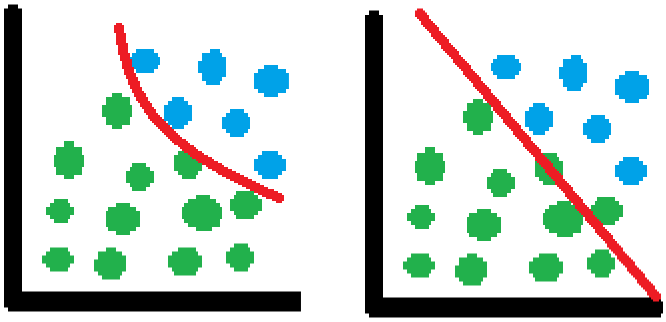


Figure 6

2.5 Gradient Descent algorithm

To train a multi-layer perceptron, start with randomly initialised weights. Then applying gradient descent, the derivative of the Loss function with respect to each weight is taken. Next is taking the weights' derivatives which will produce a gradient. The learning rate is then applied to calculate the step size that should be taken to remove error. If there are high amounts of error, the gradient descent will take a larger step size. If there is minimal error, the weights will take a smaller step size. Thus, rate of change is continuous and not constant.

New weights are now calculated. This is considered 'one epoch' or 'one pass through' the dataset. The process is then repeated for multiple epochs until either the predetermined number of epochs is hit or the change becomes nearly zero which signals for an early stop. This prevents over-fitting.

2.6 Generalisation

To estimate the generalization ability of a neural network, take the original dataset and spilt it into 3 subsets of the data. One dataset will be called the training dataset in which the network will be trained on with ~70% of the data. The second will be the validation dataset, which is a small partition of the training data and is used to check if the network needs further training or is ready to process the validation dataset. The validation dataset is unseen and check the network generalization ability on new unseen data allowing us to gauge its generalization ability. To enhance this, the data should be spilt at random to prevent unwanted biases and ensuring all datasets have a mixture of data.

2.7 High learning rates

Excessively high learning rates can create the problem of over-shooting. Over-Shooting is when the learning rate is too high and creates a step size which steps over the minimum value and can oscillate between two points preventing convergence from occurring.

References

Channon, A.C. 2023, CSC-20043 Computational and artificial intelligence I. Evolutionary Algorithms. 2nd Lecture

Thompson, A.T. 1997, 'An Evolved Circuit, Intrinsic in Silicon, Entwined with Physics.' DOI 10.1007/3-540-63173-9_61

Thompson's hardware evolution experiment: 'Evolved a Field Programmable Gate Array (FPGA) to discriminate between 1kHz and 10kHz tones'

Pavlov, I.P. 1927, 'Conditioned Reflexes: An Investigation of the Physiological Activity of the Cerebral Cortex,' Oxford University Press.

Baldwin, J.M. 1896, 'A New Factor in Evolution,' The American Naturalist, Volume 30, pp. 441-451.

Darwin, C. 1871, The Descent of Man, and Selection in Relation to Sex. John Murray.