Assignment 2

Moises A. Martinez

College of Computing and Engineering, Nova Southeastern University

CISC 0670: Artificial Intelligence

Dr. Wei Li

November 16th, 2022

Genetic Algorithm

Question 2.1

Given we have (x1, y1) = ((1 0 1 0 1 0) (1 1 1 1 1 1)), the following are possible:

- (1, 1 1 1 1 1)
- (1, 0 1 0 1 0)
- (1 0, 1 1 1 1)
- (1 1, 1 0 1 0)
- (1 0 1, 1 1 1)
- (1 1 1, 0 1 0)
- (1 0 1 0, 1 1)
- (1 1 1 1, 1 0)
- (1 0 1 0 1, 1)
- (1 1 1 1 1, 0)

This is because given the requirements of the problem, the crossover site must result in 2 non-zero strings, so we can have at most 5 elements from one parent string combined with 1 element from the other. So all the possible combinations are 5 and 1; 4 and 2; 3 and 3; 2 and 4; 1 and 5 from x1 and y1, along with the opposites (y1 and x1).

Question 2.2

Given the starting string (1, 1, 1, 1), the probability the output is:

- (0, 0, 0, 0) is <u>0.00000001</u> (0.01 * 0.01 * 0.01 * 0.01)

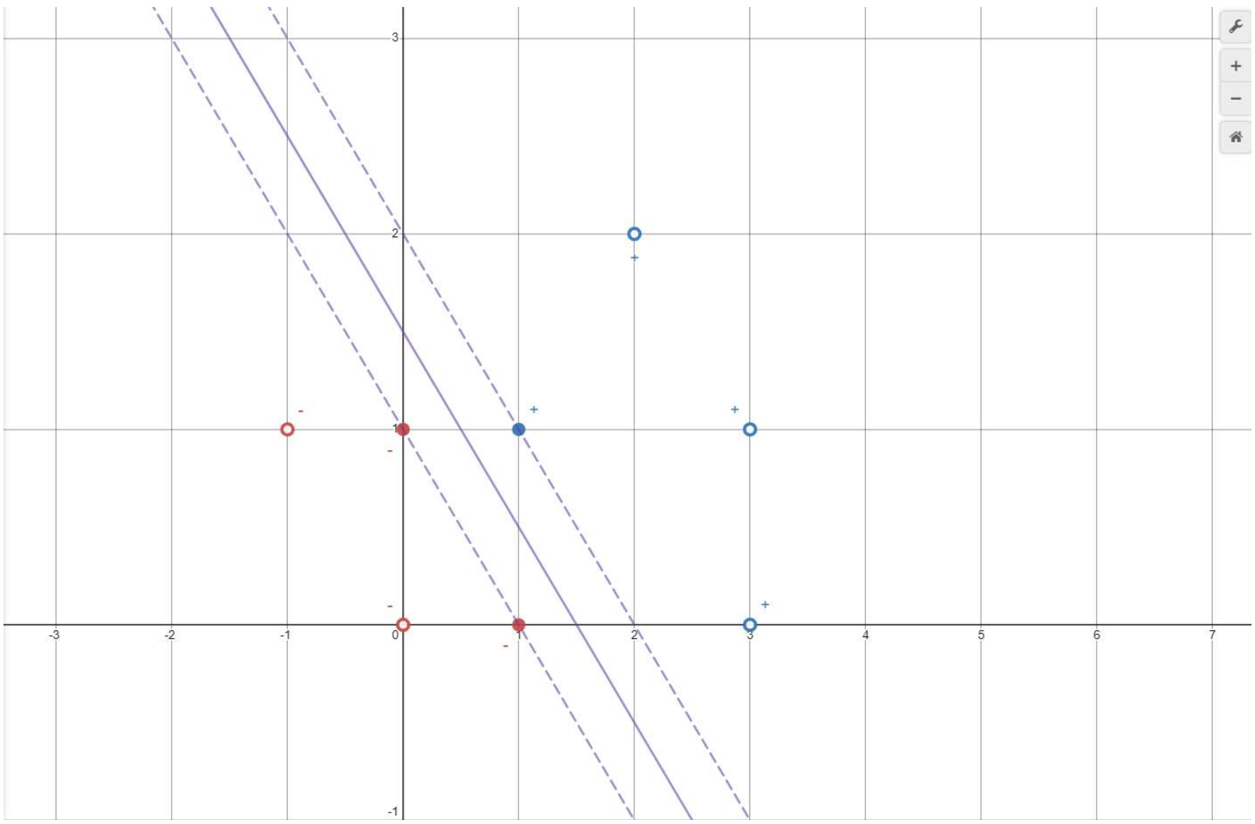
- (0, 1, 0, 0) is 0.00000099 (0.01 * 0.99 * 0.01 * 0.01)
- (1, 0, 1, 0) is 0.00009801 (0.99 * 0.01 * 0.99 * 0.01)
- (1, 1, 1, 1) is 0.96059601 (0.99 * 0.99 * 0.99 * 0.99)

*Note that since the probabilities are independent (as mentioned in the question), we are free to just multiply the odds for each bit together

Support Vector Machine

Question 2.4

(i)



The following data points are linearly separable. There are no points that exist such that we are unable to create a line that separates the two classes.

(ii) The weight vector is represented by the solid purple line on the graph in (i); more specifically, it is the line of equation $x_2 = -x_1 + 1.5$. The support vectors can be seen as the solid points on the graph in (i). More specifically, they are (1, 1), (1, 0), and (0, 1).

(iii) If you remove a support vector, the optimal margin will either stay the same or increase in size (it will never decrease). This is because since the optimal margin uses the points from the two classes that are closest to one-another, by removing a point naturally we will have to use the next closest point to calculate the new optimal margin. Since this hypothetical "second closest point" has a further distance that the original points, the new support vectors will be more spread out thus leading to a bigger optimal margin (or in the worst case the "second closest point" has the same distance as the first, resulting in no change).

(iv) Yes, the previous answer still holds true for any 2-dimensional data set. For the same reason, removing a support vector will result in the new set of support vectors consisting of points that either are the same distance or slightly further away from one-another (they'll never be closer because otherwise that point would've been a support vectors in the original data set before it was removed). This idea is universal across and n-dimensional data set, since the idea of points being distant isn't reliant on dimensions. In other words, even for an n-dimensional data set, the support vectors will consist of the points closest to one-another such that we can calculate a weight vector with the largest margin. Thus, removing a support vector will mean the next closest points will be used in creating the new weight vector, and since those new points are slightly further away (because they are guaranteed to be either equidistant or farther than the original points), the new margin will be either the same or bigger than the original.

Neural Network (Report)

Question 2.3.1: Setup

The experiment makes use of the “neuralnet” package tool in the R programming language and was developed in RStudio. The data used is the provided data in the “data.txt” file, with the “Temp”, ”Pres”, ”Flow”, and “Process” fields used as predictors and the “Rejects” field being the field the neural network is tasked with predicting (we filter out the “No.” field since it is not relevant to the data and is only needed for identification, which the neural network package can do on its own). Prior to beginning the tests, first a random seed is set for all tests (to prevent randomness from affecting the experiment) using r set.seed() function. Then the data is normalized; a min-max method is used along the 0-1 scale to normalize the data. Then, the normalized data is randomly split into the test and training groups: the training group consists of 80% of the original data and the test the remaining 20% (this leads to 225 training points vs. 75 testing points). From there, tests are carried out using these test/training set in the following general flow:

- Run the neuralnet() function, which creates the neural network with the test data. Here we can change various parameters for our liking for each test (explained in next section).
- Plot out the neural network created from the previous step
- Predict the “Rejects” values from the test set using our generated neural network
- Scale back the predicted data into the unnormalized (original) form
- Compare against the actual test data and calculate sum of squared errors
- Plot the results

From there, we can observe the results and determine how effective the predictions were and adjust our parameters as necessary to repeat the above steps for a more accurate prediction.

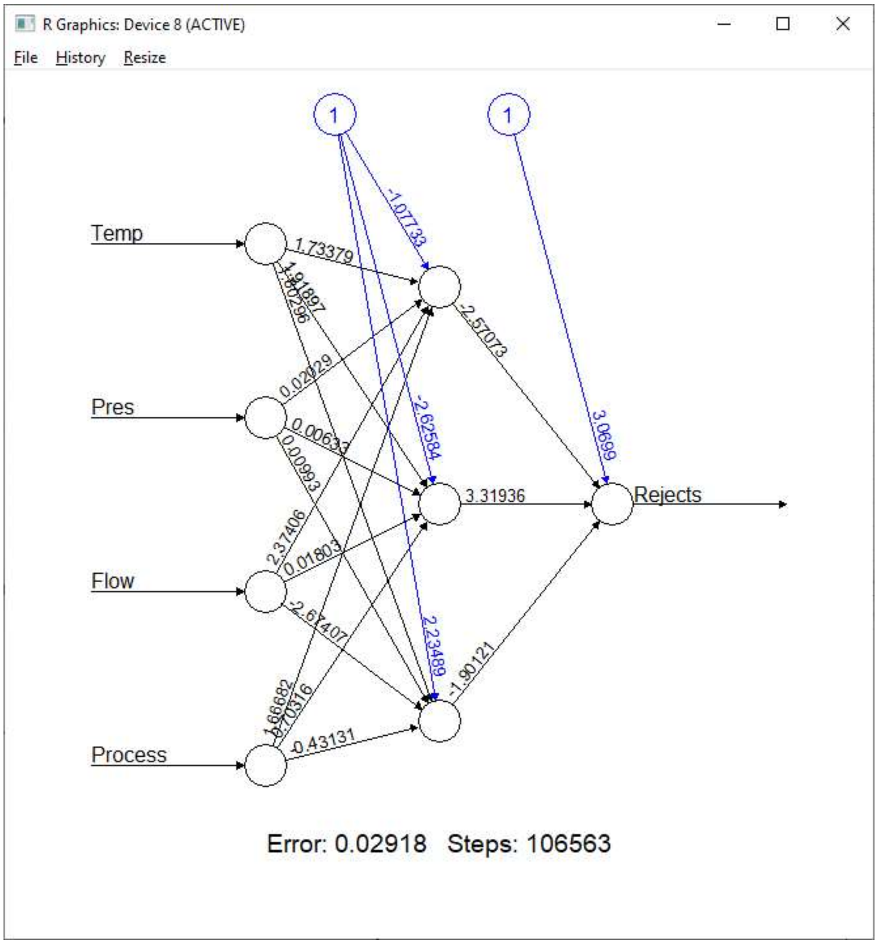
Question 2.3.2: Parameters

As mentioned, several parameters for the neuralnet() function can be adjusted to improve (or worsen) the prediction results. In this experiment, the following values were adjusted:

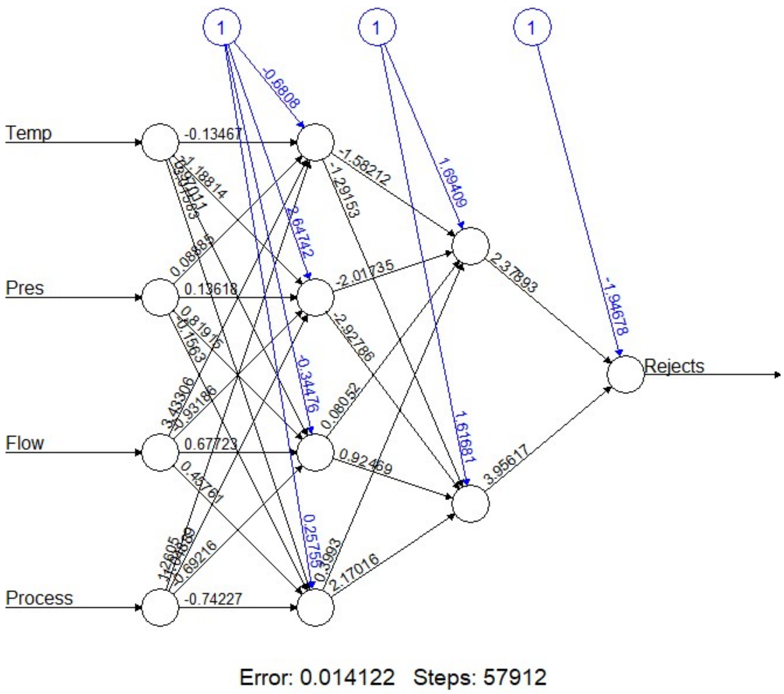
- Hidden layer count – the amount of hidden layers created for the neural network.
- Neuron count – the amount of neurons for each hidden layer
- Learning rate – the value used for the “learning rate” in the neural network generation
- act.fct – a parameter in the neuralnet function used to determine what type of differential function to used for generating the network (we can choose between logistic or tangent)
- linear.output – a parameter in the neuralnet function used to determine whether the function from “act.fct” should be applied to the last neuron or not (T if we do not apply it, F if we do)

Suffice to say, changing these values can have a significant impact on the resulting neural networks. Below you can see the resulting neural networks generated with various parameters, along with the sum of squared errors and number of steps:

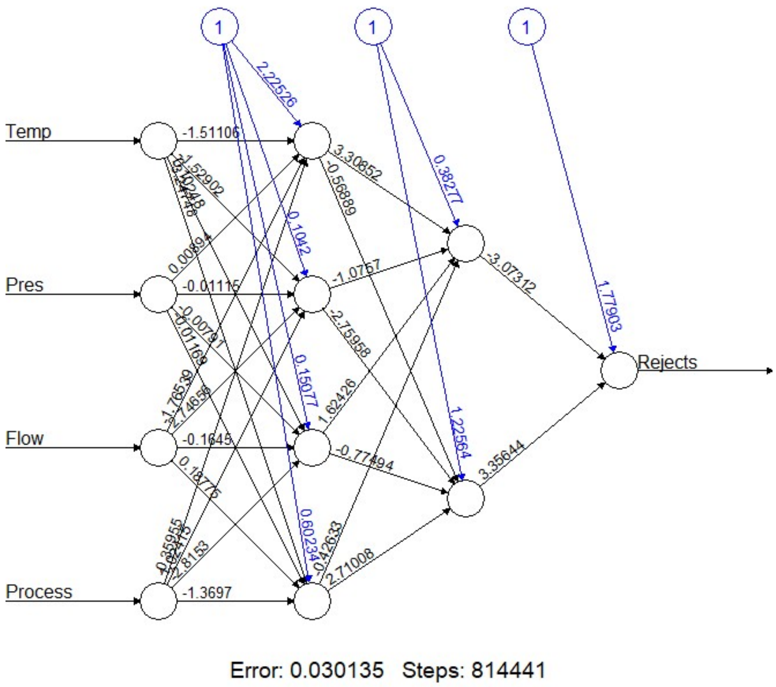
- 1 hidden layer, 3 neurons in first layer, learning rate = 0.001, act.fct = logistic, linear.output = true



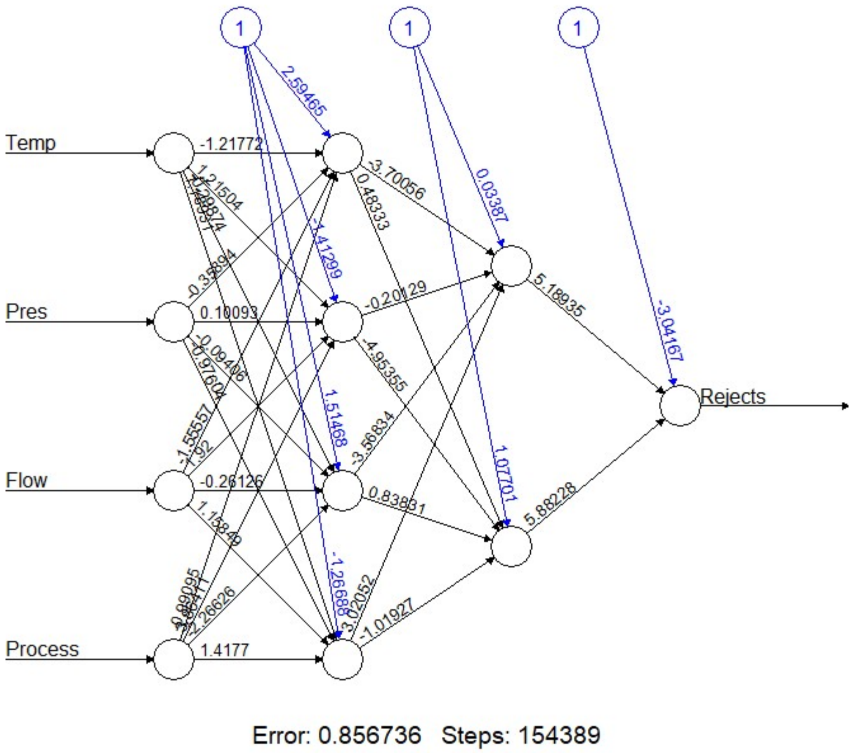
- 2 hidden layers, 4 neurons in first layer, 2 neurons in second layer, learning rate = 0.001, act.fct = logistic, linear.output = true



- 2 hidden layers, 4 neurons in first layer, 2 neurons in second layer, learning rate = 0.0001, act.fct = logistic, linear.output = true (note this one took longer to generate)



- 2 hidden layers, 4 neurons in first layer, 2 neurons in second layer, learning rate = 0.001, act.fct = logistic, linear.output = false (note this significantly raised our error)



- 2 hidden layers, 4 neurons in first layer, 2 neurons in second layer, learning rate = 0.001, act.fct = tanh, linear.output = true

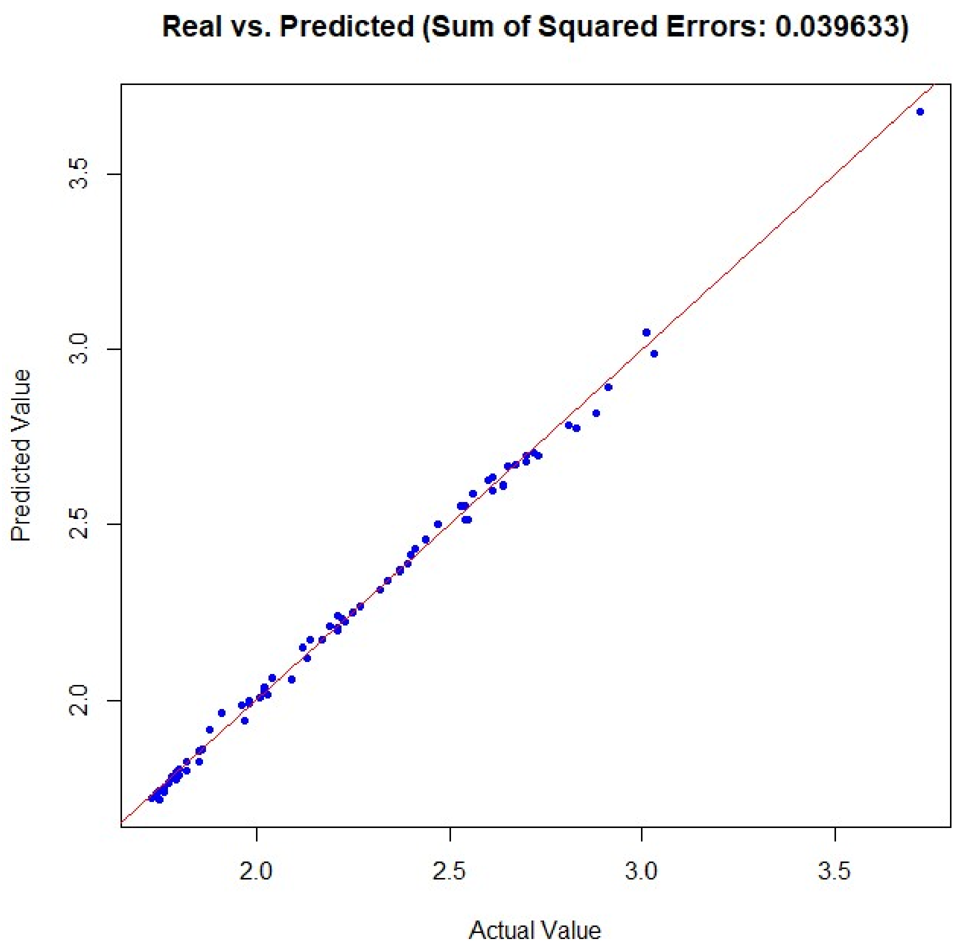
This result took so long to generate that a neural network was ultimately never generated

- 2 hidden layers, 4 neurons in first layer, 2 neurons in second layer, learning rate = 0.003, act.fct = logistic, linear.output = true

Similarly, this result also took too long to generate (>10mins)

Question 2.3.3: Results

Based on the above results, the parameters "2 hidden layers, 4 neurons in first layer, 2 neurons in second layer, learning rate = 0.001, act.fct = logistic, linear.output = true" provided us with the lowest error and thus the most accurate prediction. Below is the final prediction graph using this neural network



The following graph plots the actual values from the test set against the predicted values generated by the neural network. A point where the actual value and the predicted value match means it was a perfect prediction; this can be represented by the red line on the graph. The line is of the equation y=x, meaning all points on the line are values whose x and y coordinates match (aka our perfect predictions). The closer points are to the line, the more accurate the prediction was. The graph also displays the sum of errors for the test data.

College of Computing and Engineering
NOVA SOUTHEASTERN UNIVERSITY | NSU Florida

Certification of Authorship

Submitted to (Professor's Name): **Dr. Wei Li**

Student's Name: **Moises Alfonso Martinez**

Date of Submission: **November 16th, 2022**

Purpose and Title of Submission: **Homework Assignment 2**

Certification of Authorship: I hereby certify that I am the author of this document and that any assistance I received in its preparation is fully acknowledged and disclosed in the document. I have also cited all sources from which I obtained data, ideas, or words that are copied directly or paraphrased in the document. Sources are properly credited according to accepted standards for professional publications. I also certify that this paper was prepared by me for this purpose.

Student's Signature: **Moises Alfonso Martinez**