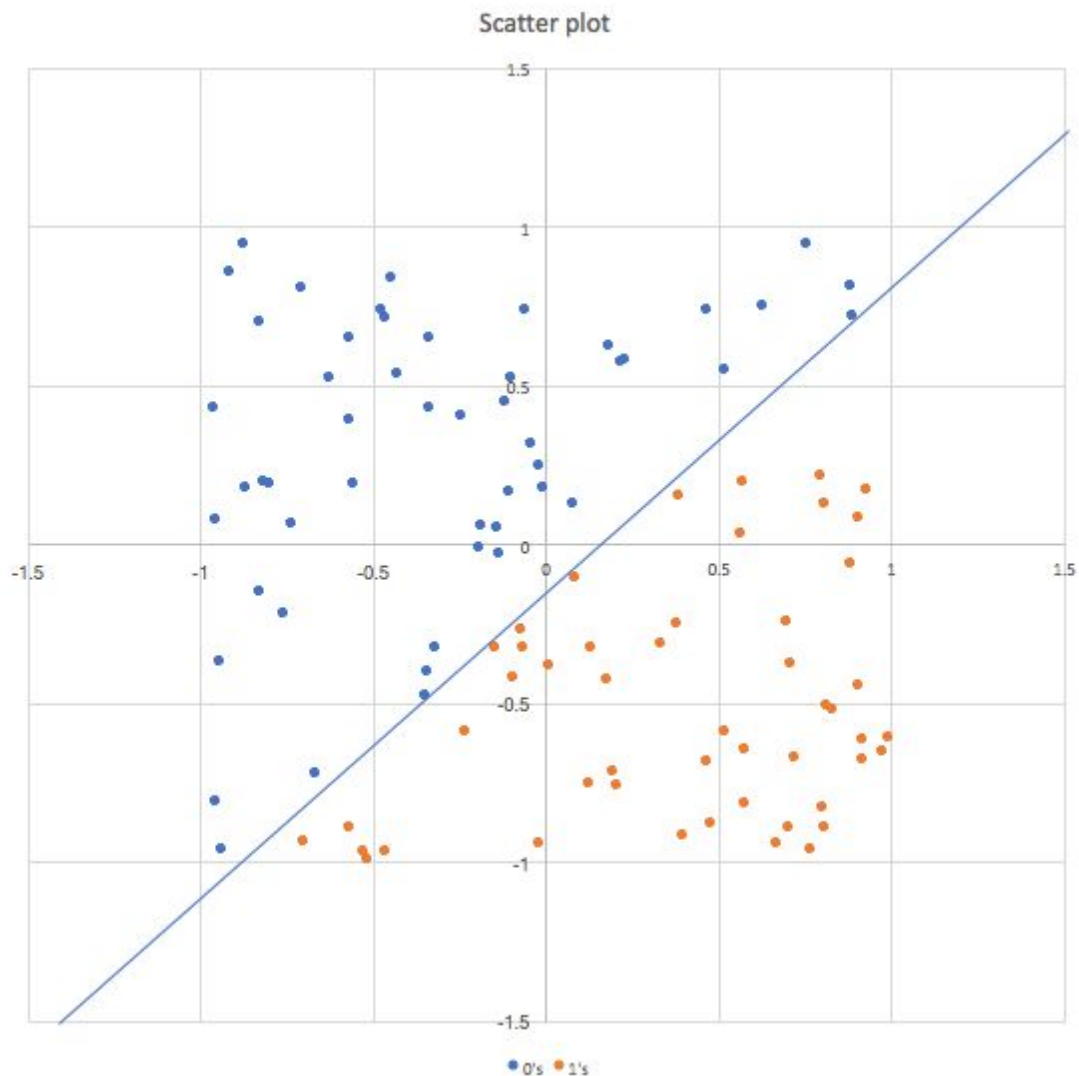


## Lab 6 Report

### Implementing Neural Networks - a Perceptron

#### Part 1

#### Scatter Plot of "linearly separable" test case

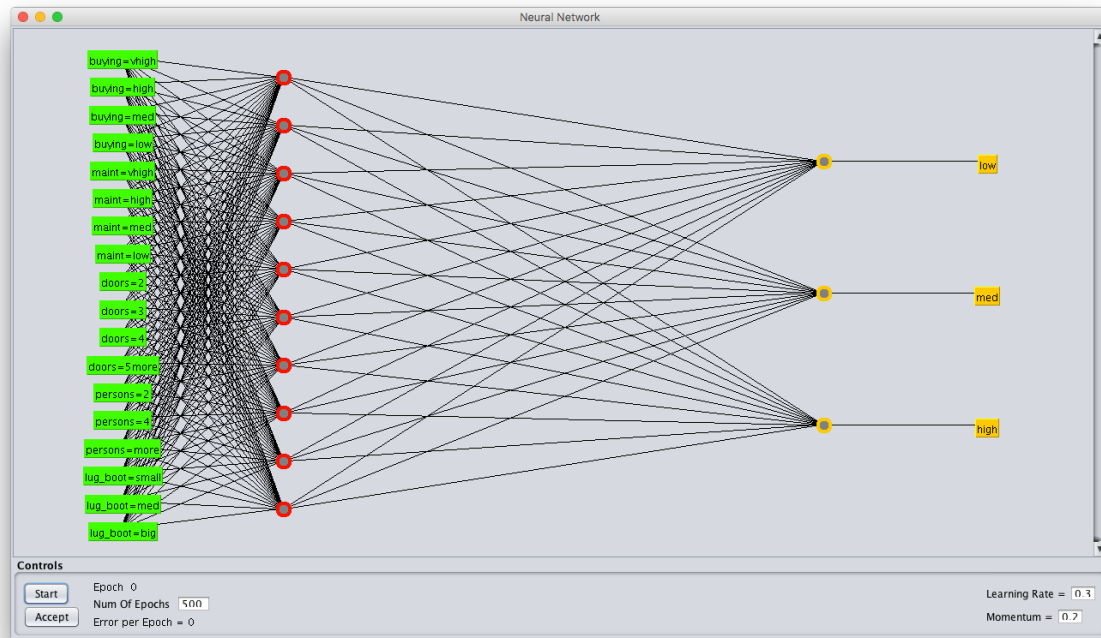


#### Equation of the slope

$$-0.01778477311442745 = 0.08513613730821404y + (-0.09563002173262877)x$$

#### Part 2

#### Analysis of ANN



\*\*\* LAYERS = a

\*\*\* LEARNING RATE = 0.3

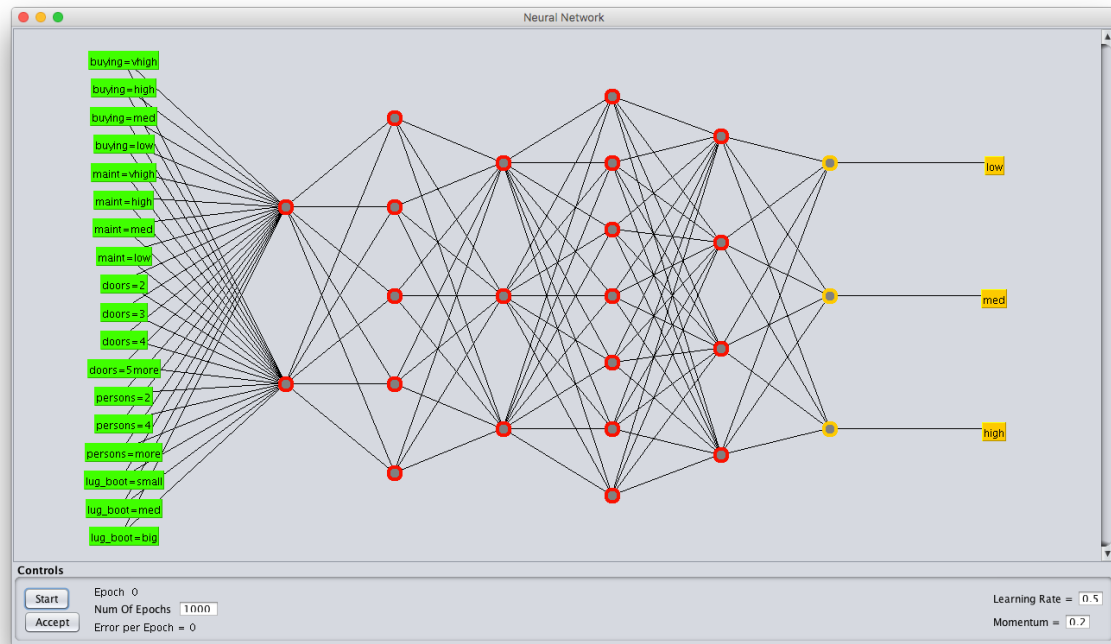
\*\*\* TRAINING TIME = 500

Time taken to build model: 4.11 seconds

=== Stratified cross-validation ===

=== Summary ===

Correctly Classified Instances	366	21.1806 %
Incorrectly Classified Instances	1362	78.8194 %
Kappa statistic	-0.1823	
Mean absolute error	0.4775	
Root mean squared error	0.5141	
Relative absolute error	107.4384 %	
Root relative squared error	109.06 %	
Total Number of Instances	1728	



\*\*\* LAYERS = 2,5,3,7,4

\*\*\* LEARNING RATE = 0.5

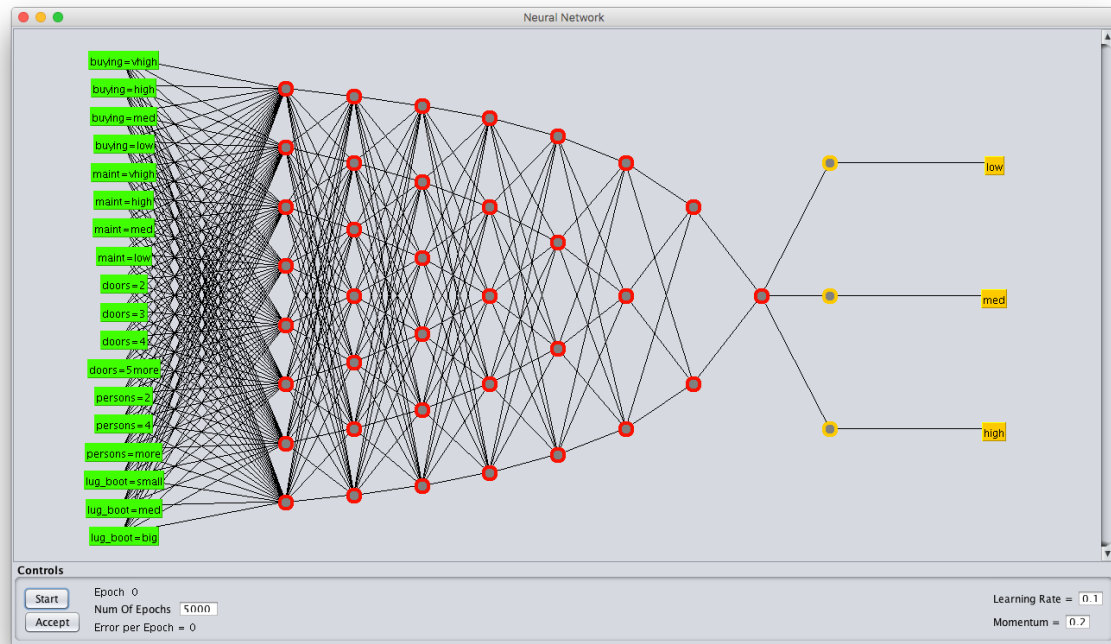
\*\*\* TRAINING TIME = 1000

Time taken to build model: 40.17 seconds

=== Stratified cross-validation ===

=== Summary ===

Correctly Classified Instances	574	33.2176 %
Incorrectly Classified Instances	1154	66.7824 %
Kappa statistic	-0.0017	
Mean absolute error	0.4446	
Root mean squared error	0.4753	
Relative absolute error	100.0293 %	
Root relative squared error	100.8178 %	
Total Number of Instances	1728	



\*\*\* LAYERS = 8,7,6,5,4,3,2,1

\*\*\* LEARNING RATE = 0.1

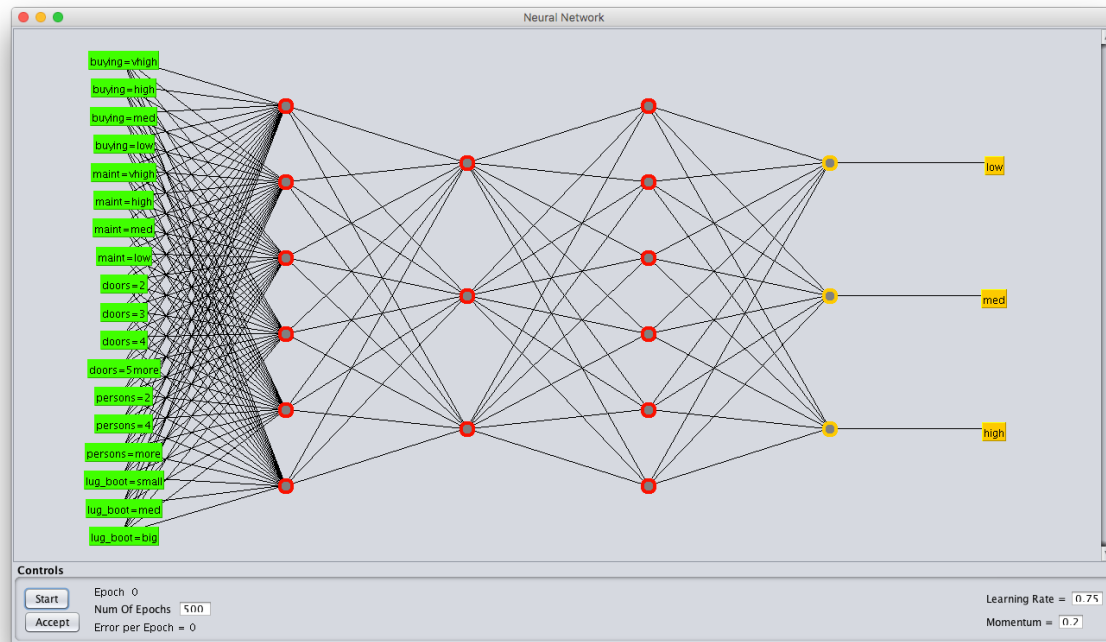
\*\*\* TRAINING TIME = 5000

Time taken to build model: 82.98 seconds

=== Stratified cross-validation ===

=== Summary ===

Correctly Classified Instances	575	33.2755 %
Incorrectly Classified Instances	1153	66.7245 %
Kappa statistic	-0.0009	
Mean absolute error	0.4445	
Root mean squared error	0.4721	
Relative absolute error	100.0092 %	
Root relative squared error	100.1417 %	
Total Number of Instances	1728	



\*\*\* LAYERS = 6,3,6

\*\*\* LEARNING RATE = 0.75

\*\*\* TRAINING TIME = 500

Time taken to build model: 3.92 seconds

=== Stratified cross-validation ===

=== Summary ===

Correctly Classified Instances	573	33.1597 %
Incorrectly Classified Instances	1155	66.8403 %
Kappa statistic	-0.0026	
Mean absolute error	0.4477	
Root mean squared error	0.4806	
Relative absolute error	100.7424 %	
Root relative squared error	101.9534 %	
Total Number of Instances	1728	

### Thoughts on WEKA's models

Even though the parameters of layers, learning rate and training time where changes drastically, the percentage of correctly classified instances didn't changed as much as we expected to; the sole thing that changed enough to be considered, was the time taken to build the mode.

It is important to state the the "test option" parameter was always on *Cross-validation*. So, maybe if we have had used another testing option, the Artificial Neural

Network would have behaved differently and the percentage of correct classifications would have turned to be reliable (and not the average of 33% that was obtained).

To finish, another observation is that maybe the dataset or the time taken to train the algorithm was not large enough to obtain a function that could classify this complicated data. The only thing that we suggest is to continue experimenting with the parameters, as we cannot stop at a given time the algorithm and check the values of each neuron as they would not make any sense at all.

### **Final reflections**

- Explanations as to what are ANNs good for.

Neural Networks are good for grouping data in different datasets based on similarity. Utilizing multiple nodes allows us to classify data even if the dataset is very large or complicated. Furthermore by increasing the number of samples the network can learn more to improve its accuracy.

- Where would you use them?

Anywhere where I would need to classify or create rules based on linearly separable data. Where the data is available in quantities enough to allow the creation of training datasets, and the data has been at least partially labeled. Some examples could be Stock Market prediction, Image compression, Filtering unwanted internet traffic and Email spam prediction.

- Are they worth the effort implementing or not?

Depends on the problem, but as long as you picked the right dataset, then yes.

- What kinds of problems do they not solve?

They do not solve problems where there is data that can not be linearly separated, or if the data has not been labeled.