

# COMP3330 Assignment1

Sebastian Hadley

University Of Newcastle

**Abstract.** In this report, we assess the effectiveness of different methods of solving Machine Learning problems by solving a selection of problems with different methods and judging how different factors can impact the performance of the solutions.

## 1 Question 1

The 2 spiral-problem is quite difficult as it is a relatively small dataset with complex relations that determine the way each data point is classified, because of this it is difficult to build a Neural Network complex enough to identify the correct decision boundaries while also not over-fitting to the training data.

### 1.1 Part a

For the "original dataset" of Lang and Witbrock (1988), different feed-forward Neural Network architectures were experimented with, including varying the number of layers, neurons per layer, learning rate and activation functions. The performance was evaluated on the different configurations using 10-fold cross validation with each fold being broken up into test and training sets to ensure ability to generalize was being assessed. The quantitative metrics that were judged were, loss and accuracy as well as a qualitative analysis of the decision boundaries. The decision boundaries were output for each fold, as well as a final decision boundary that took the mode class predicted for each point within the grid over all the decision boundaries to output an average decision boundary over the 10-folds.

In the initial attempt the model was constructed with 2 hidden layers each containing 8 neurons, that used the Relu activation function with the output layer using SoftMax. This model was initially tested with a learning rate of 0.01 over 100 epochs, with a few attempts at adjusting the hyper-parameters to tune the results, this architecture performed fairly poorly for all the different hyper parameters that were attempted. [Table 1.2](#) shows the results of a few attempted hyper parameters.

The most successful of these quantitatively is likely the 8 neuron, 100 epoch version, however the version that used 1000 epochs had a promising average decision boundary, that seemed to resemble a spiral however still had a lot of jagged edges and holes in the spiral shapes. [Figure 1.2](#)

These results indicated that despite the small dataset, the relationship between the points is complex enough that we would need to add additional layers and

neurons. After experimenting with different options, the most consistent results were achieved by a model containing 5 hidden layers with the first using Relu and the rest using the LeakyRelu activation function, with SoftMax in the output layer as well as the hyper parameters shown in this table. [Table 1.3](#)

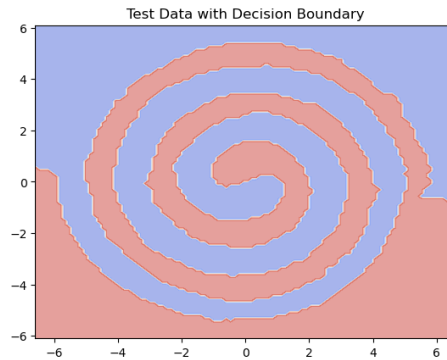
This Model performed quite well on both the training and test sets, with accuracy being better then the loss, 1 of the 10 folds seemed to go astray leading to it not accurately classifying the test sets however even those folds still outperformed the previous models discussed above.

**Table 1.1.** Average Results of Final Model Architecture

Training Loss	Training Accuracy	Test Loss	Test Accuracy	Time Taken
0.11	0.95	1.06	0.84	FILL IN LATER

These results are promising, however the time taken was significantly longer then the previous models, the speed is especially poor in relation to the size of the dataset however the relationships that determine which spiral a point fell into are complex which does make the time more acceptable.

The decision boundaries for this model also are very good showing a clear spiral shape on all but 1 of the folds, with the average looking particularly clean. This boundary shows that the model is successfully able to learn the shape of the data consistently over 10 folds, which is even better considering that the model is also successful in generalizing to unseen data. Feature Engineering perhaps could be used to improve the performance of the model, by increasing the input neurons with things such as  $(X_1)^2$  and  $(X_2)^2$  however the model already accurately captures the data.



**Fig. 1.1.** Average Decision Boundary over 10 folds

## 1.2 Part b

A variation of the 2 spiral task was generated that has much less steep angles using more steps to curve making it less dense in the middle proportionally. Instead of using pre-defined steps followed by predefined angle changes as the original dataset did, this one is constructed using random point generation which introduces some more noise to the dataset and makes it much less consistent in comparison to the original dataset, it was formed using the code provided in the specification. [45deg (n.d.)] Text for Part b.

## 1.3 Part c

Text for Part c.

# 2 Question 2

## 2.1 Part a

Text for Part a.

## 2.2 Part b

Text for Part b.

# 3 Conclusion

Conclusion text.

## References

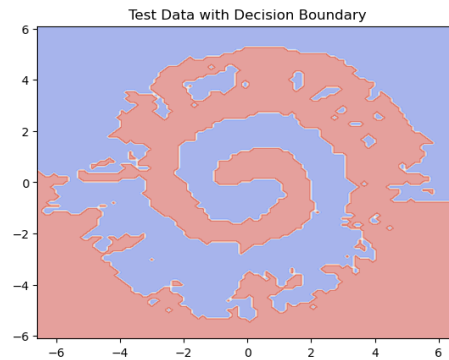
45deg (n.d.). 45deg. (n.d.). *2-spiral task generator* [Code]. Retrieved from <https://gist.github.com/45deg/e731d9e7f478de134def5668324c44c5>

## A Appendix

### A.1 Question 1

**Table 1.2.** Results of 2 Layer Architectures

Epochs	Hidden Neurons	Learning Rate	Average Test Loss	Average Test Accuracy
100	8	0.01	0.84	0.38
300	16	0.02	1.61	0.37
1000	16	0.02	4.91	0.42



**Fig. 1.2.** Average Decision Boundary over 10 folds

**Table 1.3.** Final Model Architecture

Epochs	Hidden Neurons	Learning Rate
1000	64	0.01

Figures and tables that were placed into the appendix.