



Honours Degree in Computing

Computational Intelligence: TensorFlow Assignment

Submitted by: Derek McCarthy, B00007439

Submission date: 14/04/2019

Declaration

I/We declare that this material, which I/We now submit for assessment, is entirely my/our own work and has not been taken from the work of others, except where otherwise stated. I/We have identified and included the source of all facts, ideas, opinions, and viewpoints of others in the assignment references. Direct quotations from books, journal articles, internet sources, module text, or any other source whatsoever are acknowledged, and the source cited are identified in the assignment references.

I/We understand that plagiarism, collusion, and copying are grave and serious offences and accept the penalties that would be imposed should I/we engage in plagiarism, collusion or copying. I acknowledge that copying someone else's assignment, or part of it, is wrong, and that submitting identical work to others constitutes a form of plagiarism. I/We have read and understood the colleges plagiarism policy 3AS08 (available [here](#)).

This material, or any part of it, has not been previously submitted for assessment for an academic purpose at this or any other academic institution.

I have not allowed anyone to copy my work with the intention of passing it off as their own work.

Name: Derek McCarthy

Dated: 14/04/2019

Contents

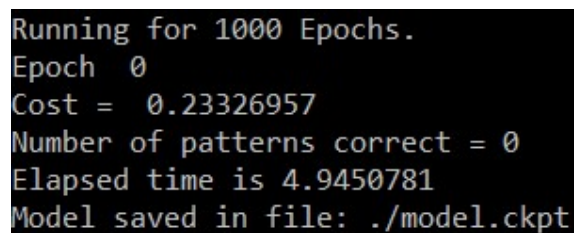
Introduction	4
Training	4
Evaluation.....	8
Conclusion	9

Introduction

The purpose of this report is to document the results of building a multilayer neural network using TensorFlow. TensorFlow is a math's library that is also used in machine learning and the construction of neural networks. There will be two phases to the project training phase and evaluation phase. The training and evaluation data will be supplied. For the training phase there will be 4000 examples of data. As for the evaluation phase this will use 1000 examples of data. Both the training and evaluation models will have 2 inputs and one hot encoded label. Our goal is to get the program to be accurate to within 0.1 of the target values for both training and evaluation.

Training

In this section we will discuss the results we achieved in the training phase of the project. In this phase we will be adjusting three variables values. These variables are the number of hidden layers, number of epochs and the learning rate. The default values for these variables were, 5 hidden layers, 1000 epochs and a learning rate of 0.1. To begin we ran the program 10 times with these values but were predicting 0% of the patterns correct during the 10 runs.

A screenshot of a terminal window with a black background and white text. The text displays the output of a TensorFlow training process for 1000 epochs. It shows the current epoch (0), the cost (0.23326957), the number of patterns correctly predicted (0), the elapsed time (4.9450781), and the file path where the model was saved (./model.ckpt).

```
Running for 1000 Epochs.  
Epoch 0  
Cost = 0.23326957  
Number of patterns correct = 0  
Elapsed time is 4.9450781  
Model saved in file: ./model.ckpt
```

Figure 1 - Result of Initial Runs

So, then we increased the learning rate to 0.5 and ran in 10 times. Again, this yield 0 patterns predicted correctly. After this we then increased the number of epochs to 5000.

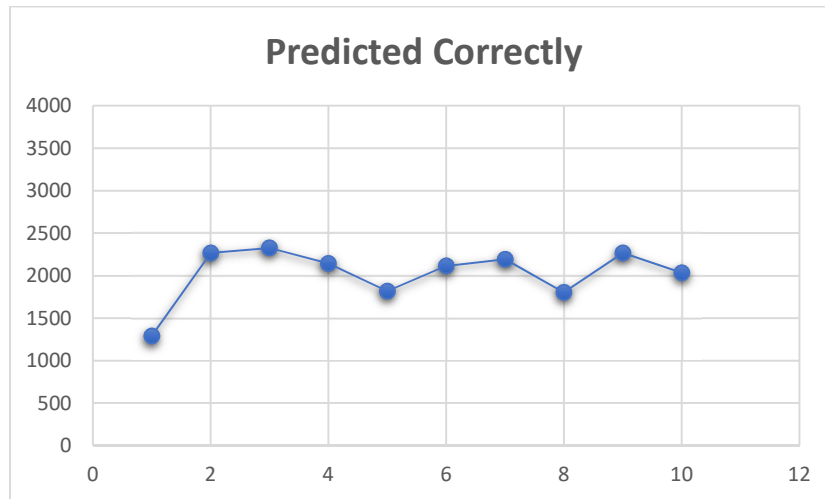


Figure 2

In figure 2 we can see the results of the number of patterns being predicted correctly over 10 runs. The average patterns being predicted correctly was 2025.4 out of 4000. This gives an accuracy of 50.63%, which is way off the target accuracy of 99.9%. The average time to run was 22.14 seconds. We also tested adjusting the hidden layer but found 5 to give the best accuracy with this number of epochs and learning rate.

After this we increased the learning rate to 1 and kept the number of epochs at 5000 and decreased hidden layers at 3.

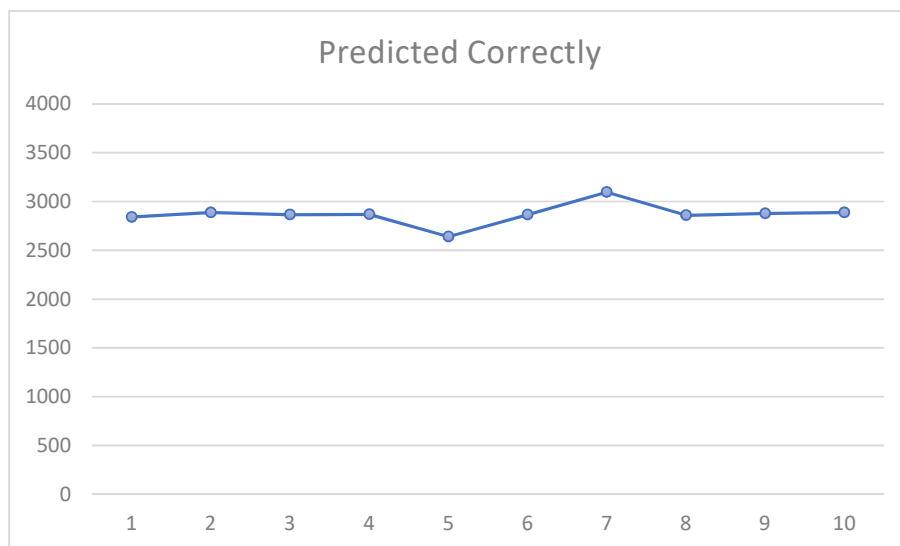


Figure 3

As you can see the number of predicted has increased from the previous runs. The average predicted correctly is now 2868.6 (72.72% accuracy). The average run time has also increased to 26.15 seconds.

From here we then increased the number of hidden layers in 0.5 increments, we also tried different combinations of hidden layers and incremented the number of epochs by 1000 after every set of ten runs.

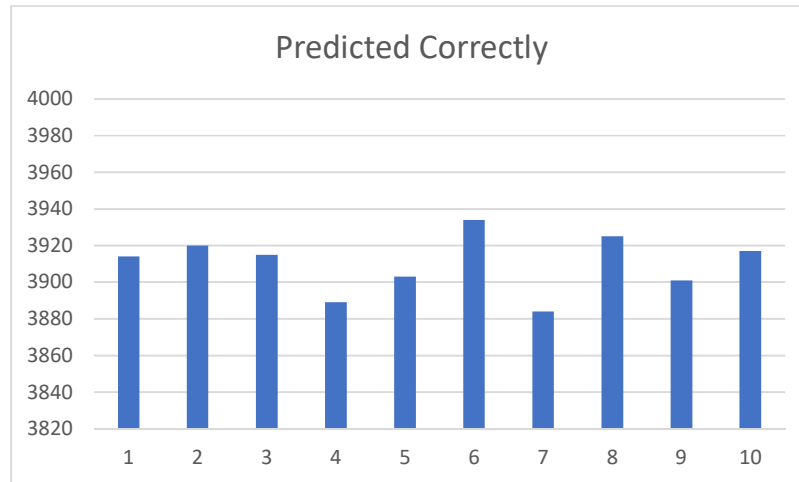


Figure 4

In figure 4 we can see the results from the 10 runs with the number of epochs at 10000, learning rate at 5 and the number of hidden layers at 3. The average predicted correctly is 3910.2 (97.76% accuracy) and the average run time has also increased to 40.57. Although this model's accuracy is high it is not as high as we would like it to be. We will now test the model by increasing the number of epochs in 1000 increments.

Epochs 11000			Epochs 12000			Epochs 13000			Epochs 14000			Epochs 15000		
Hidden Layers	3		Hidden Layers	3		Hidden Layers	3		Hidden Layers	3		Hidden Layers	3	
Learning Rate	5		Learning Rate	5		Learning Rate	5		Learning Rate	5		Learning Rate	5	
Run	Correct	Time	Run	Correct	Time	Run	Correct	Time	Run	Correct	Time	Run	Correct	Time
1	3919	54.43	1	3909	60.23	1	3914	66.06	1	3923	73.77	1	3938	75.39
2	3920	54.6	2	3924	62.55	2	3945	64.7	2	3930	71.38	2	3940	74.33
3	3896	54.2	3	3928	68.4	3	3934	69.4	3	3937	77.12	3	3942	77.45
4	3921	57.6	4	3919	66.7	4	3923	68.4	4	3927	67.54	4	3936	72.79
5	3912	55.3	5	3910	60.3	5	3941	61.2	5	3941	73.9	5	3934	68.74
6	3908	59.6	6	3927	60.1	6	3917	66.5	6	3936	65.45	6	3943	74.6
7	3928	54.2	7	3919	65.5	7	3933	65.5	7	3939	74.34	7	3927	75.68
8	3934	60.2	8	3928	64.9	8	3937	67.7	8	3932	66.43	8	3925	74.93
9	3898	55.1	9	3921	65.1	9	3935	64.97	9	3928	72.5	9	3931	72.26
10	3913	54.1	10	3936	61.5	10	3929	65.1	10	3929	69.23	10	3929	76.37
Average	3914.9	55.933	Average	3922.1	63.528	Average	3930.8	65.953	Average	3932.2	71.166	Average	3934.5	74.254

Figure 5

In figure 5 we can see the results from increasing the number of epochs by 1000 per every 10 runs. Using 3 hidden layers a learning rate of 5 and 11000 epochs we get an average of 3914.9 patterns predicted correctly (97.87% accuracy). With an average time per run of 56 seconds. Increasing the number of epochs to 12000 gives you an average of 3922 patterns predicted correctly (98.05% accuracy). With an average of 63.53 seconds per run. At 13000 and 14000 epochs the average number of patterns predicted correctly was 3930.8 and 3932.2 (accuracy 98.27% and 98.31%) respectively. For 15000 epochs the number predicted correctly has slightly increased to 3934.5 (98.36% accuracy). While the time per run has also

increased to an average of 74.25 per run. As there is very little to be gained from increasing the number of epochs we will again try increase the learning rate.

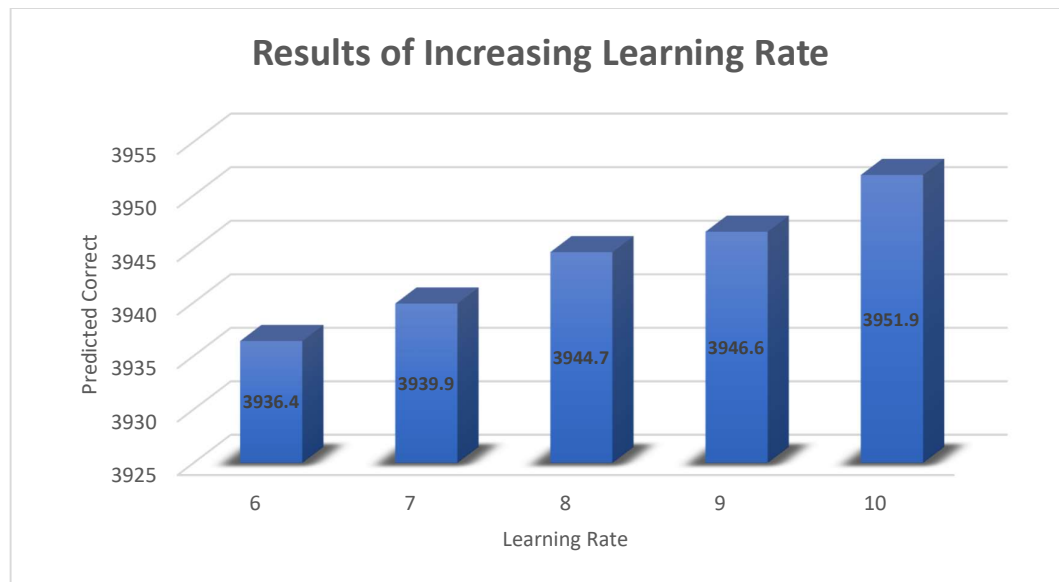


Figure 6

In figure 6 we can see the results from increasing the learning rate. As you can see a learning rate of 10 gives us the highest number of patterns predicted correctly with 3951.9 (98.80% accuracy). We now feel the learning rate we should use is 10. We chose to go no higher than 10 as we felt this could cause the model to overfit the results. So, we will try to improve this accuracy by focusing on the hidden layers.

Epochs	Learning Rate	Hidden Layers	Average Predicted Correct	Accuracy
15000	10	2	3951	98.78%
15000	10	3	3951.9	98.80%
15000	10	4	3949	98.73%
15000	10	5	3948.8	98.72%
15000	10	6	3948	98.70%
15000	10	7	3946.8	98.67%
15000	10	8	3945.6	98.64%
15000	10	9	3947.6	98.69%
15000	10	10	3947	98.68%

Figure 7

In figure 7 we can see the results of increasing the number of hidden layers from 2-10. As you can see 3 hidden layers gives the best results for the averaged pattern predicted correctly and accuracy. From this table we can also see that as the number of hidden layers goes up from 3 the accuracy of the model goes down, except for using 9 hidden layers. As we feel this trend is likely to continue as we increase the hidden layers we feel that using 3 hidden layers is best. The below formula was used to calculate the accuracy of the average patterns predicted correctly.

$$\frac{\text{Average Predicted Correctly}}{\text{Number of Samples in Data}} * 100$$

Now that we have the values for our training data we can now start the evaluation. The values we will now use for training are,

Number of Epochs	Hidden Layers	Learning Rate
15000	3	10

Evaluation

In this section we will discuss the results from the evaluation phase. During the evaluation phase of model, we will use 15000 epochs and 3 hidden layers. For the learning rate we will start at 0.1 and then set it to 0.5 and increment it by 0.5 until we reach 10.

Epochs	Learning Rate	Hidden Layers	Average Predicted Correct	Accuracy
15000	0.1	3	52	5.20%
15000	0.5	3	863	86.30%
15000	1	3	945	94.50%
15000	1.5	3	952	95.20%
15000	2	3	976	97.60%
15000	2.5	3	983	98.30%
15000	3	3	985	98.50%
15000	3.5	3	983	98.30%
15000	4	3	987	98.70%
15000	4.5	3	987	98.70%
15000	5	3	988	98.80%
15000	5.5	3	989	98.90%
15000	6	3	989	98.90%
15000	6.5	3	990	99.00%
15000	7	3	990	99.00%
15000	7.5	3	991	99.10%
15000	8	3	991	99.10%
15000	8.5	3	991	99.10%
15000	9	3	992	99.20%
15000	9.5	3	992	99.20%
15000	10	3	993	99.30%

Figure 8

In figure 8 we can see the results from running the evaluation phase. As you can see a learning rate of 0.1 gave the least accurate results with 5.2% patterns be predicted correctly. While a learning rate of 10 gave the best results 99.30%.

Conclusion

To conclude we feel the model is the best we could achieve with the data. The accuracy is good with 98.68% for training and 99.30% for evaluation. To get a higher accuracy for both training and evaluation we needed to use astronomical numbers to reach this. For example, to get 3983 (99.58% accuracy) patterns predicted correctly we needed to use 250000 epochs and a learning rate of 100 with 20 hidden layers. After consideration we decided not to use these values for our model because it took 17 times longer to run for just a 0.28% increase in accuracy and also because this increase in accuracy could be caused by overfitting of the data. Although a learning rate of ten is considered high and is recommended to between 0.1 and 0.75. We opted to use this value as it gave better accuracy overall in both models and felt the accuracy for a learning rate below 1 was too low.