

Han-Hung Liu

Yutian Dai

1003871349

1002958980

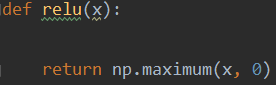
1. Neural Networks using Numpy

Han-Hung Liu Yutian Dai

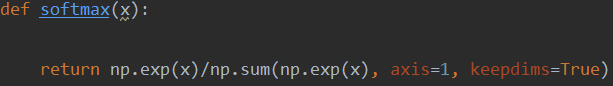
Contribution: 50% Contribution: 50%

1.1 Helper Functions

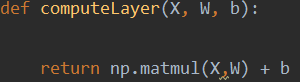
1. ReLU():



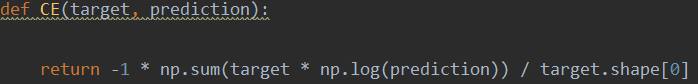
1. softmax():



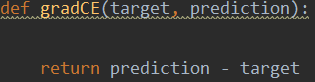
1. compute():



1. averageCE():



1. gradCE():



Analytical expression:

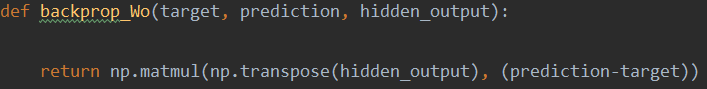
Calculate :

Calculate :

Apply chain rule:

1.2 Backpropagation Derivation

1. :

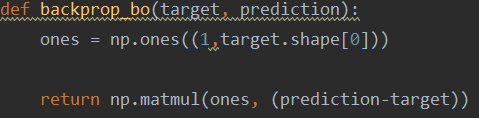


Analytical expression:

Calculate :

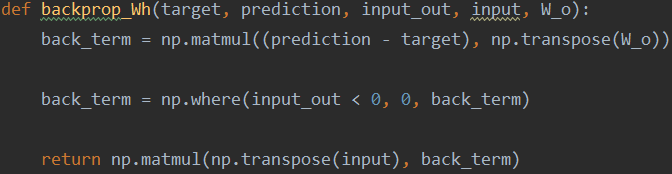
Then:

2. :



Analytical expression:

3. :



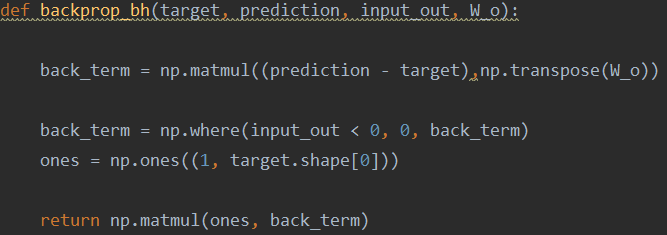
Analytical expression:

Calculate :

Calculate :

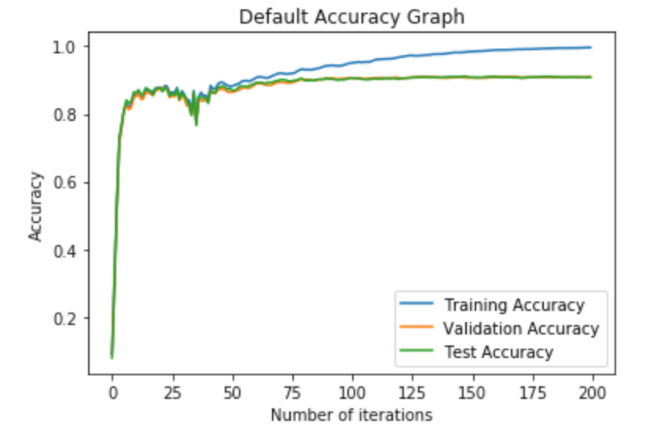
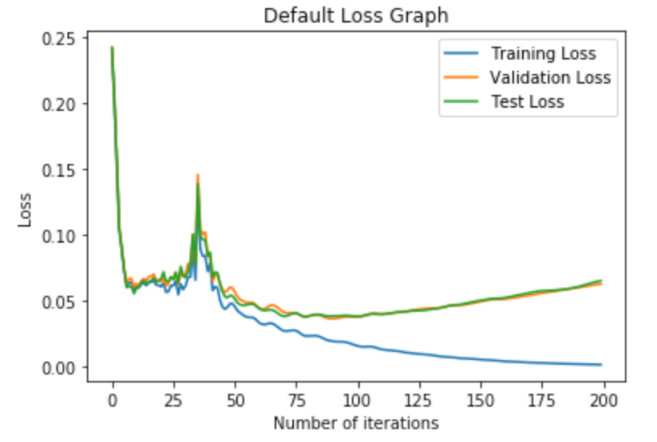
Then:

4. :



Analytical expression:

1.3 Learning

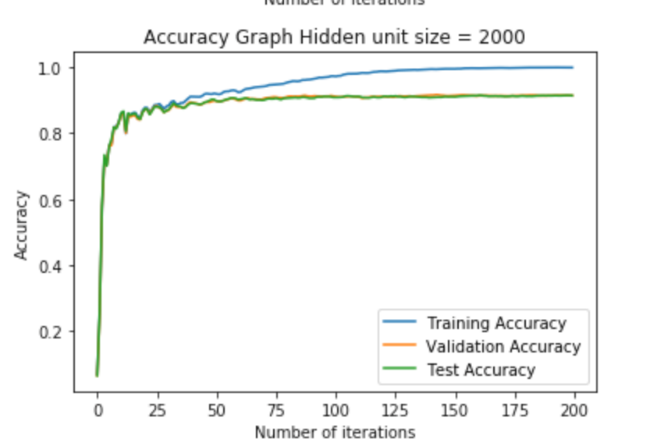
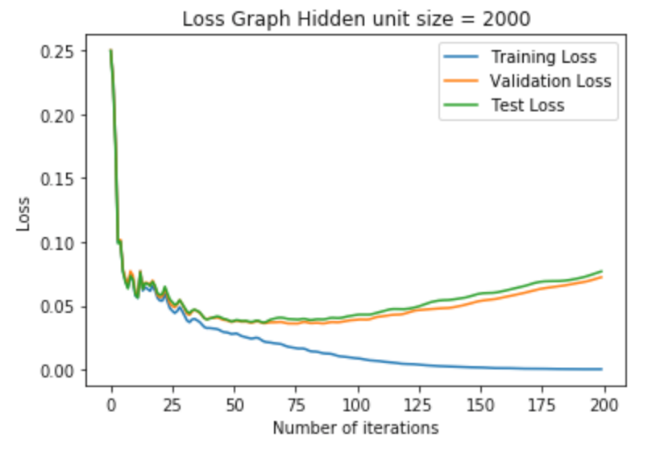
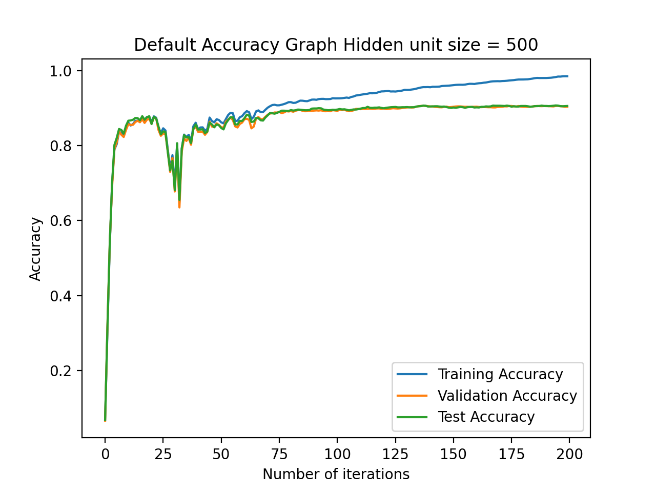
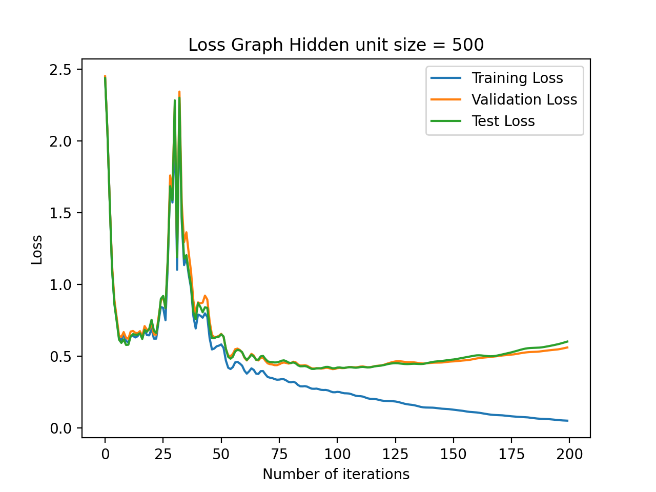
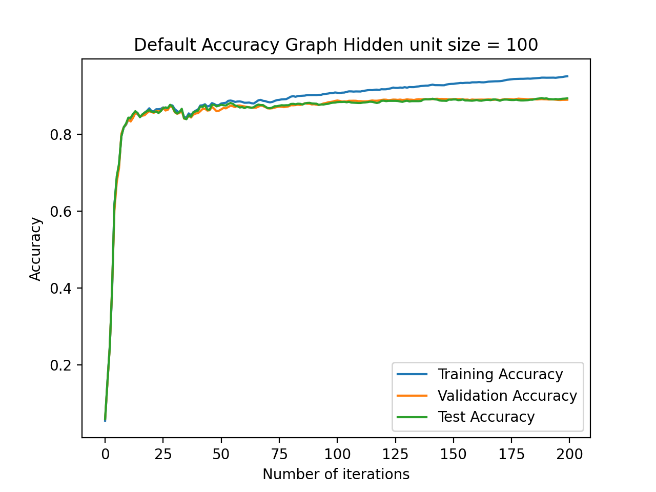
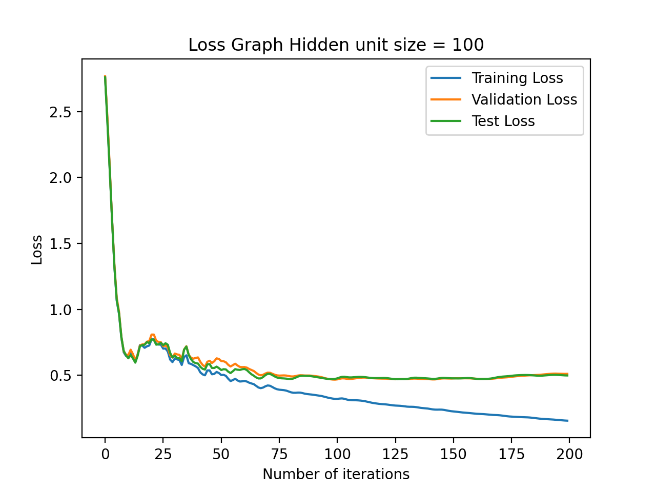


|  |  |  |
| --- | --- | --- |
| Training data accuracy | Valid data accuracy | Test data accuracy |
| 0.9964 | 0.9091 | 0.9078 |

Table 1: Accuracy with default setting

1.4 Hyperparameter Investigation

1. Number of hidden units



|  |  |  |  |
| --- | --- | --- | --- |
|  | Hidden units = 100 | Hidden units = 500 | Hidden units = 2000 |
| Training data accuracy | 0.9513 | 0.9849 | 0.9989 |
| Valid data accuracy | 0.8902 | 0.9036 | 0.914 |
| Test data accuracy | 0.8939 | 0.9057 | 0.9141 |

Table 2: Accuracy with different number of hidden units

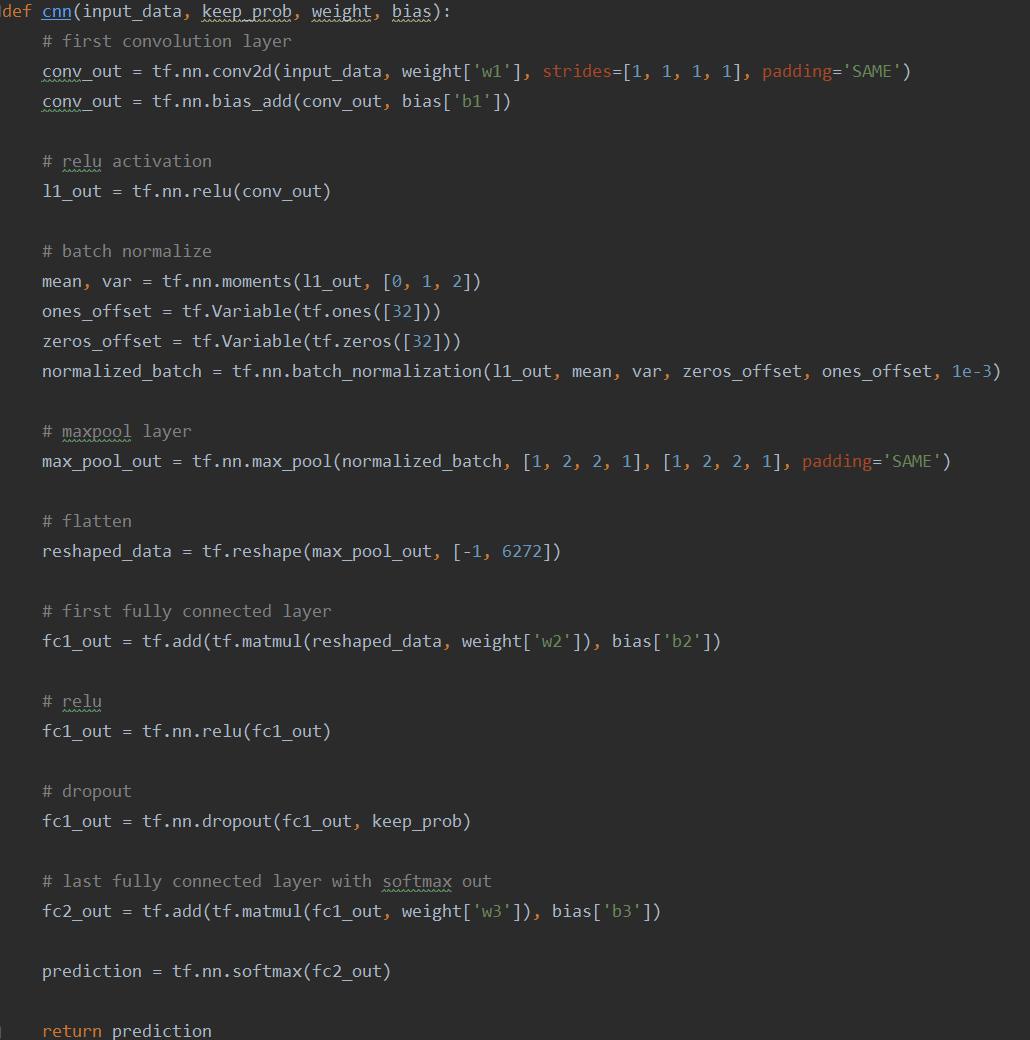
According to the table and figures, hidden units = 100 results in the worst accuracy, and hidden units = 2000 results in best accuracy. However, the model with 2000 hidden units requires the longest time to train. In conclusion, a larger number of hidden units will results in higher accuracy but lower performance.

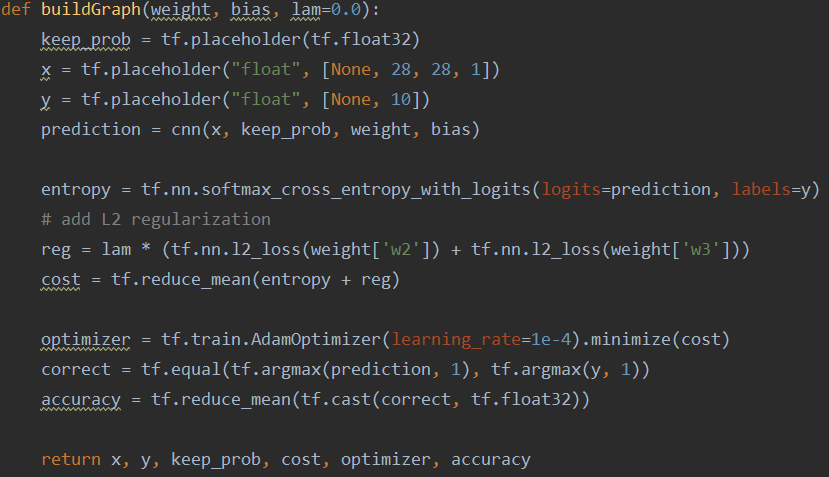
2. Early stopping:

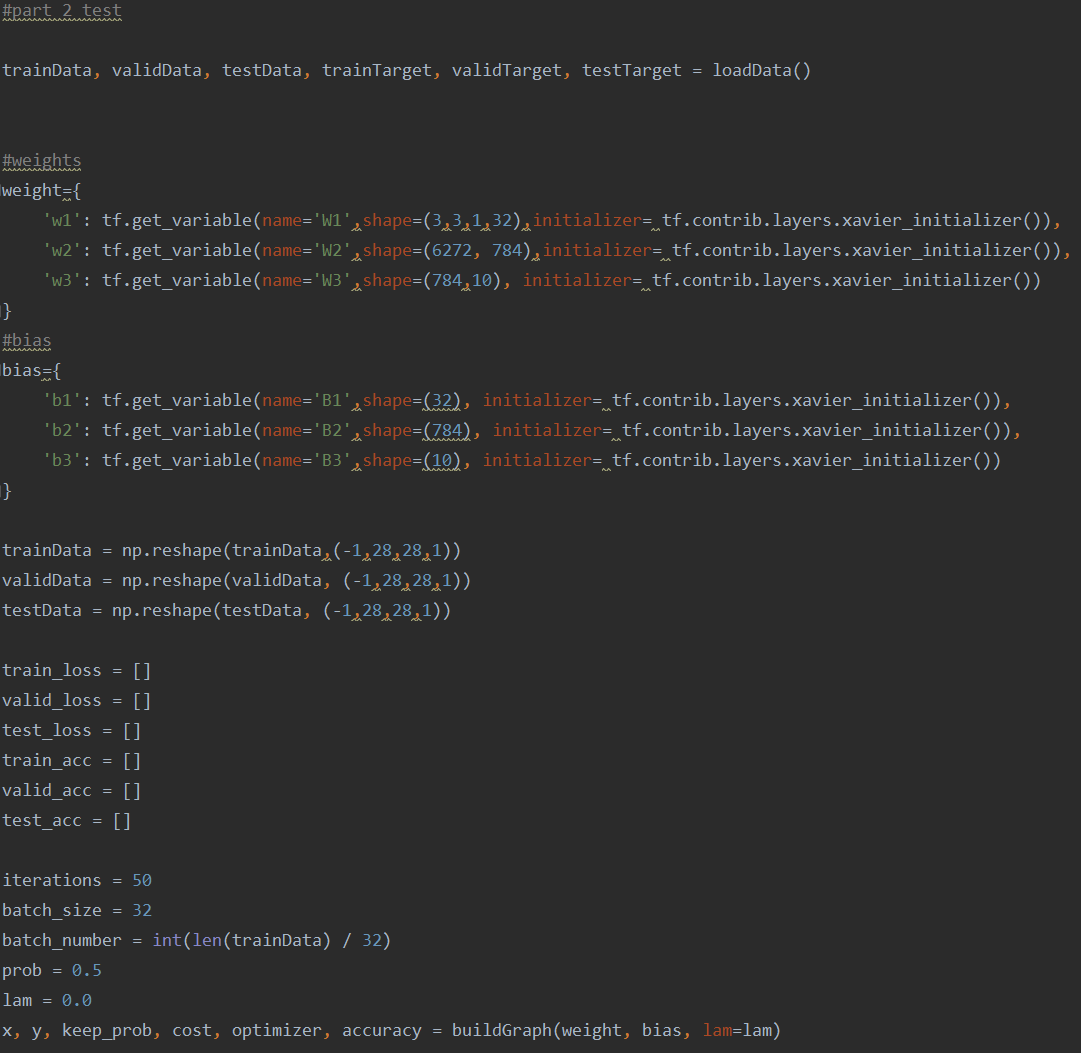
According to the figures, the accuracy of the validation data reached the highest accuracy after 80 iterations. At this point, training data accuracy is 0.94, validation data accuracy is 0.91, and test data accuracy is 0.91. After this point, a difference between training data accuracy and validation data accuracy occurs, which represents the occurrence of overfitting.

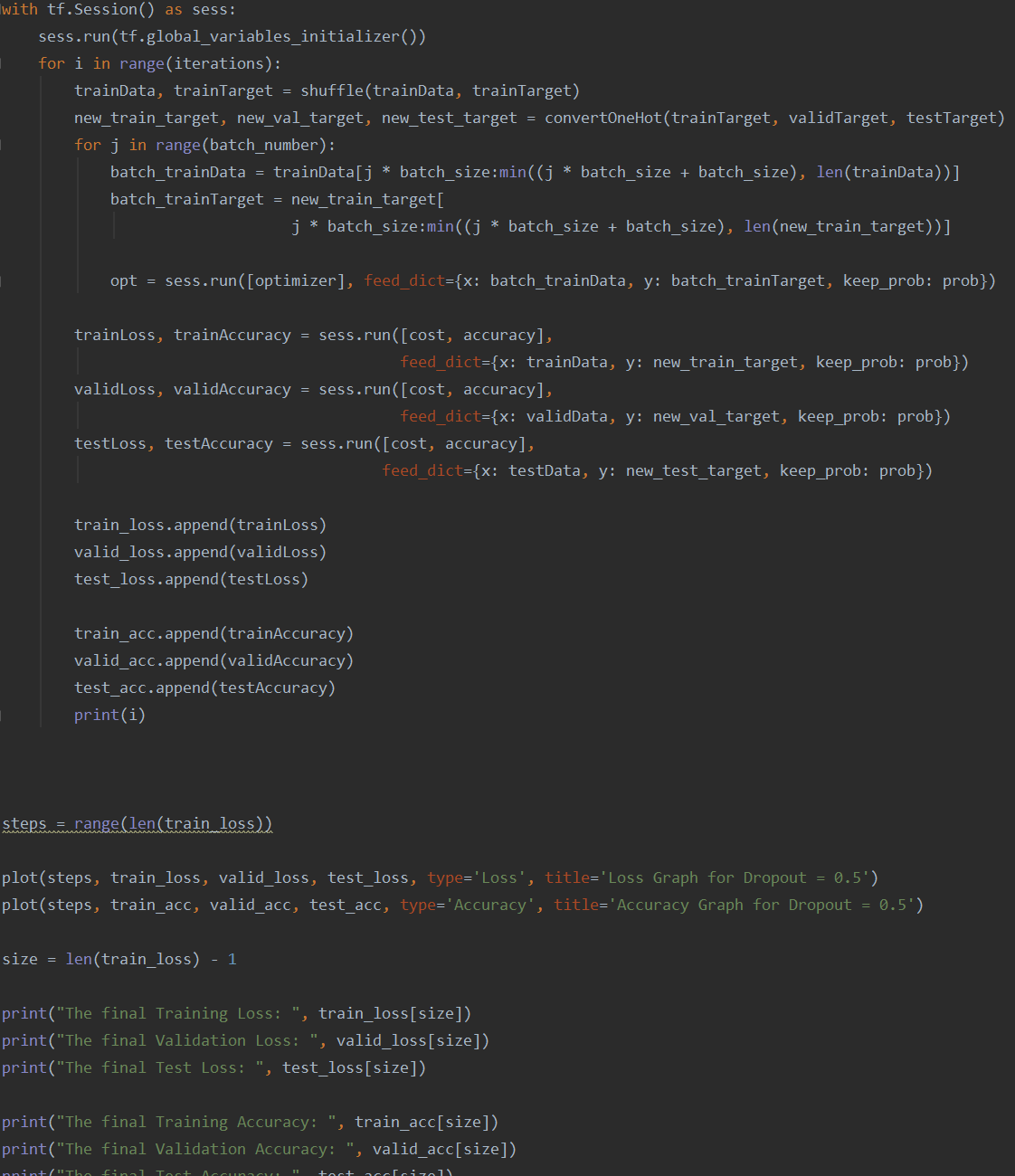
1. Neural Networks in Tensorflow

2.1 Model implementation

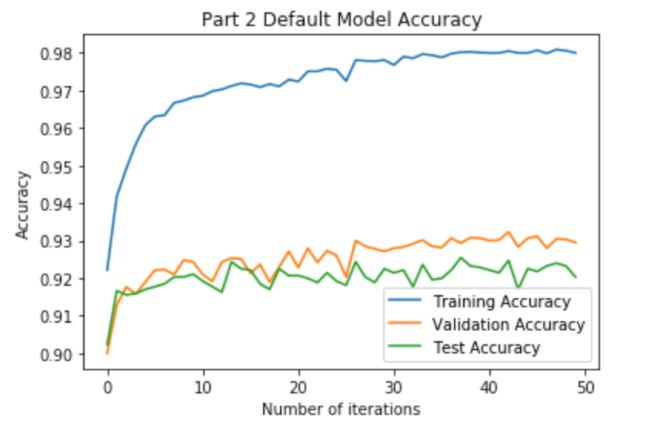
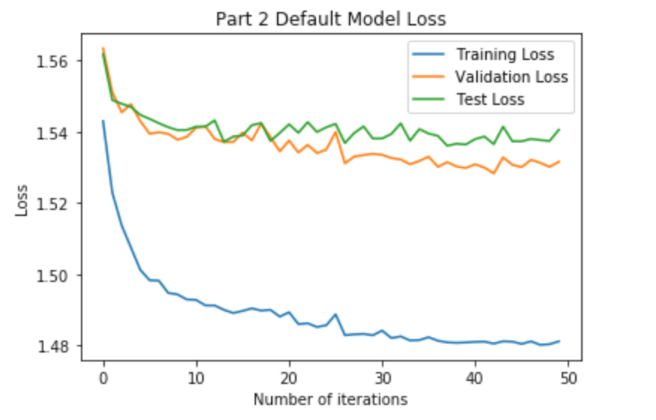






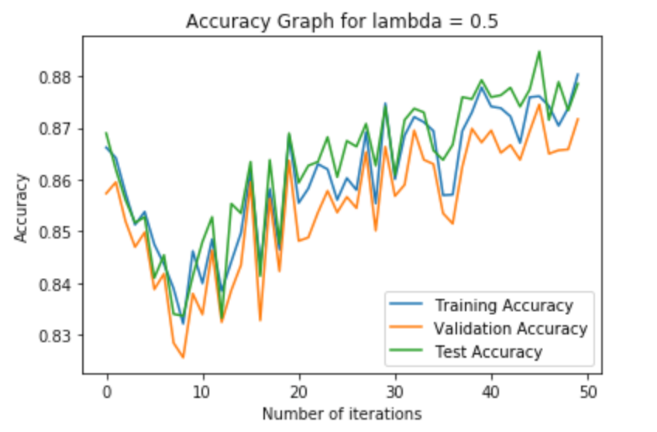
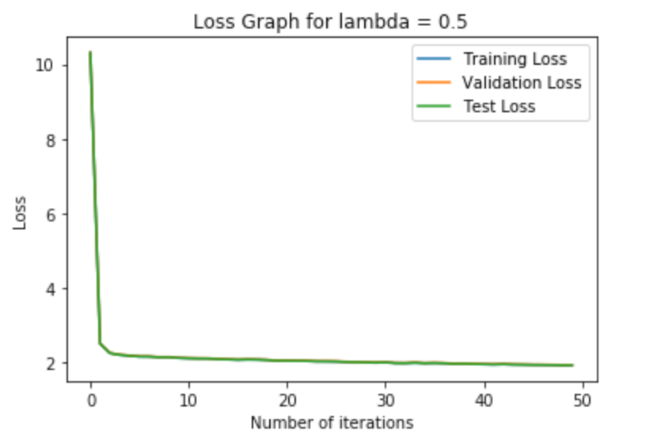
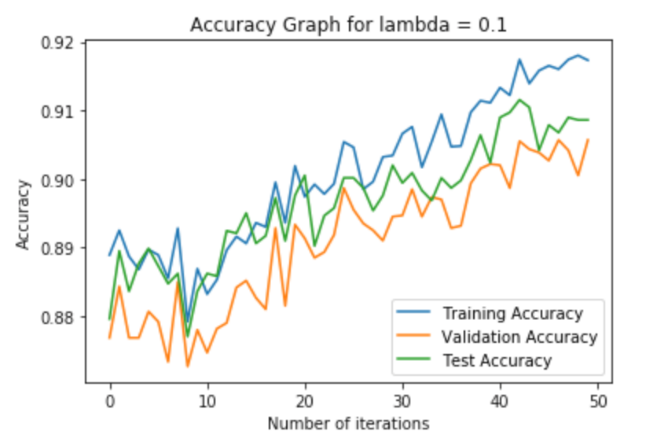
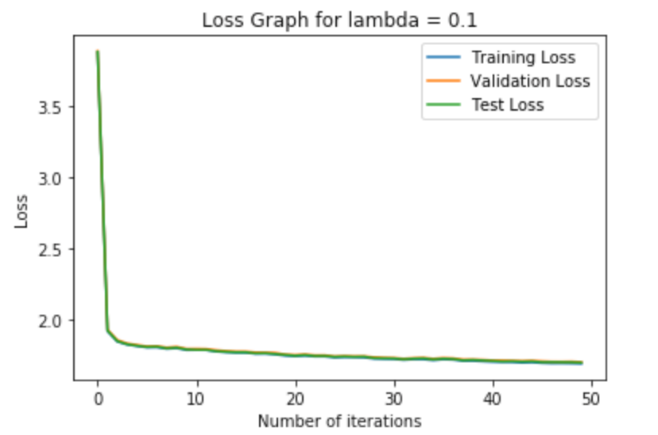
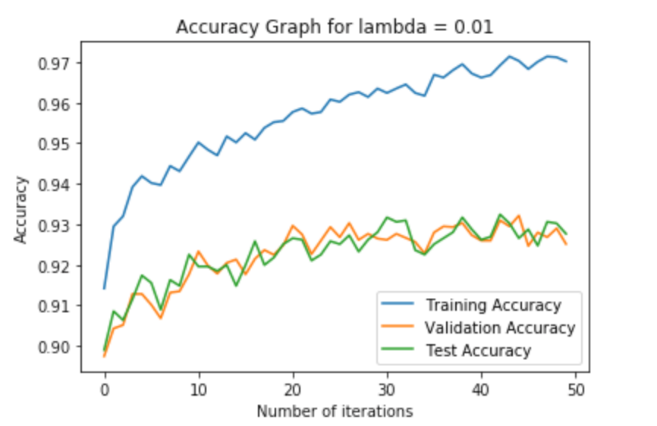
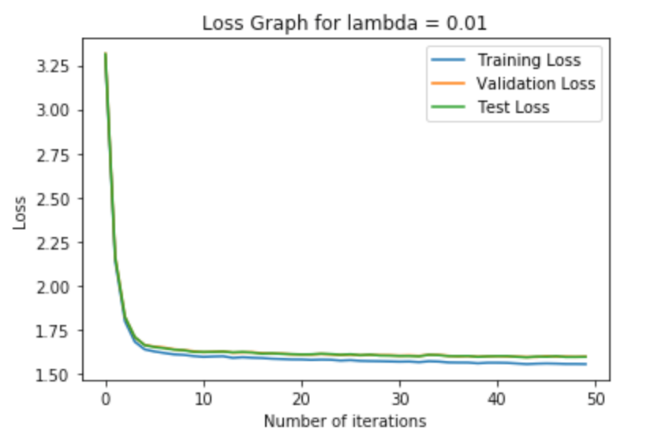


2.2 Model Training



2.3 Hyperparameter Investigation

1. L2 Regularization:

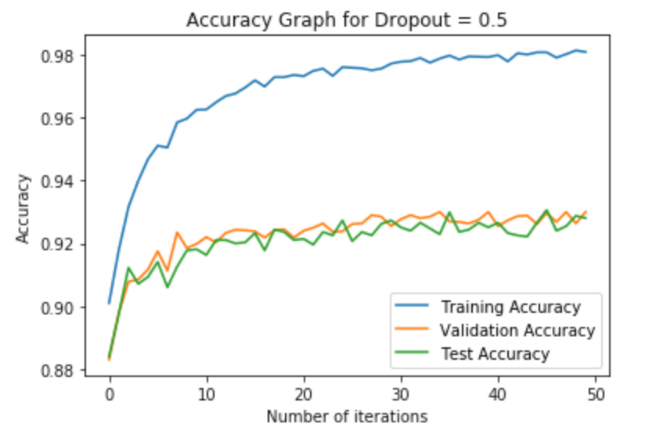
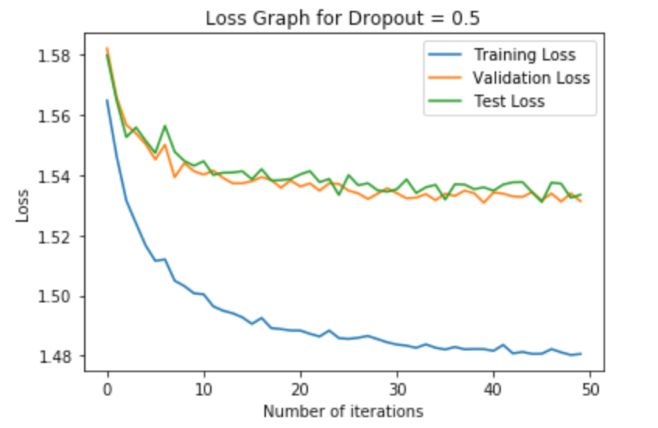
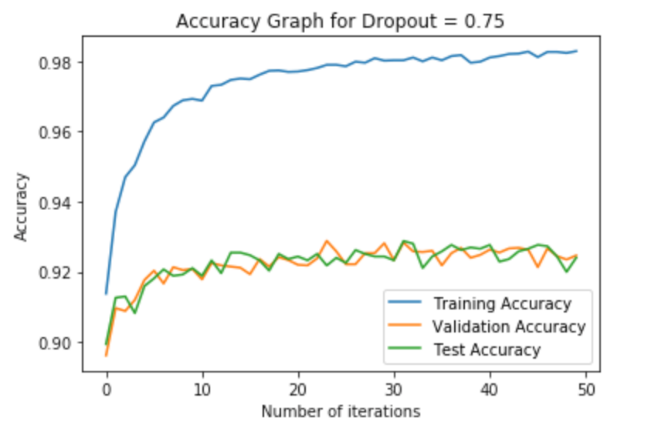
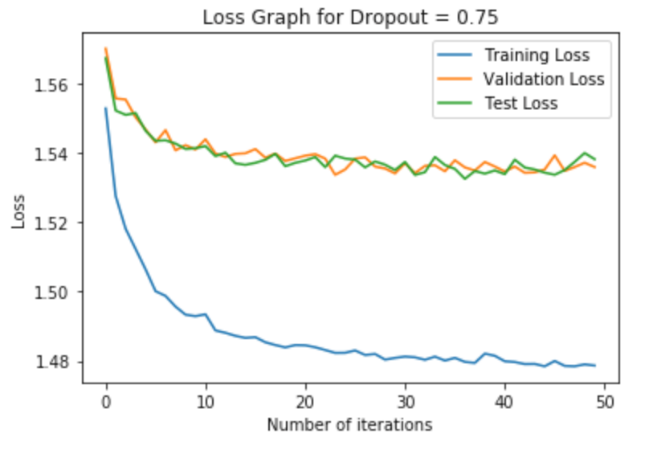
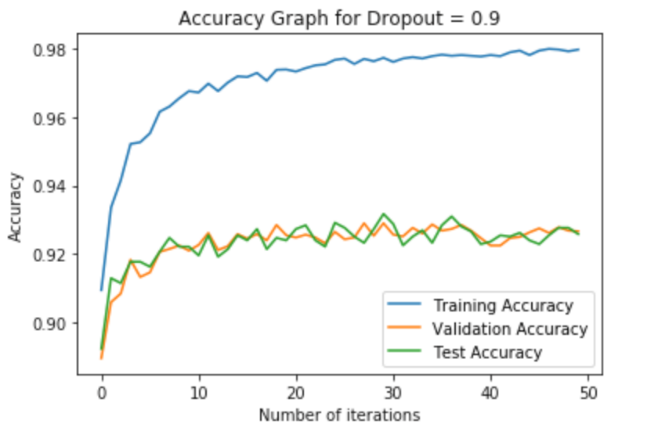
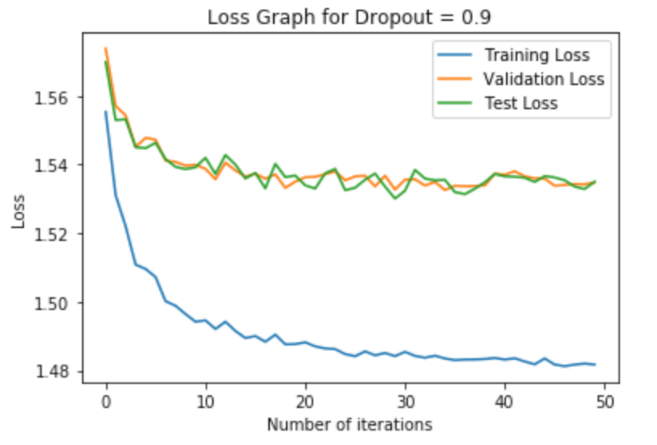


|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | λ = 0 | λ = 0.01 | λ = 0.1 | λ = 0.5 |
| Training data accuracy | 0.98 | 0.97 | 0.92 | 0.88 |
| Valid data accuracy | 0.93 | 0.93 | 0.91 | 0.87 |
| Test data accuracy | 0.92 | 0.93 | 0.91 | 0.88 |

Table 3: Accuracy with different λ

According to the table and figures, λ = 0 results in the best accuracy, and λ = 0.5 results in worst accuracy. Also, as the λ increases, the difference between training data accuracy and validation data accuracy becomes smaller. Therefore, adding L2 regularization can prevent the occurrence of overfitting. Moreover, as the λ increases, the model will have more noises.

2. Dropout:



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Dropout = 0 | Dropout = 0.9 | Dropout = 0.75 | Dropout = 0.5 |
| Training data accuracy | 0.98 | 0.98 | 0.98 | 0.98 |
| Valid data accuracy | 0.93 | 0.93 | 0.92 | 0.93 |
| Test data accuracy | 0.92 | 0.93 | 0.92 | 0.93 |

Table 4: Accuracy with different Dropout