CSE574
Project-2

Handwritten Digits Classification

Amulya Reddy Datla
UB Person number: 50560100
amulyare@buffalo.edu

Vignesh Thalur Jayachandrakumar
UB Person number: 50559900
vignesht@buffalo.edu

I. Explanation with supporting figures of how to choose the hyper-parameter for Neural Network

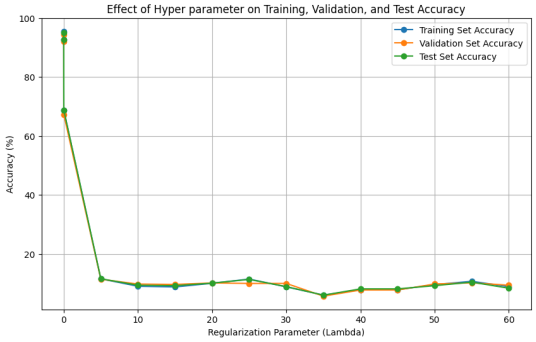

Fig. 1. figure 1

The plot in figure 1 illustrates that varying the regularization parameter lambda affects the accuracy of the neural network on training, validation, and test sets. The regularization parameter lambda controls the balance between minimizing the loss function and penalizing large weights in the neural network. Accuracy is highest across at lambda value 0 indicating strong performance but also a potential risk of over-fitting. This means the model could perform well on the training set but generalize poorly to unseen data.

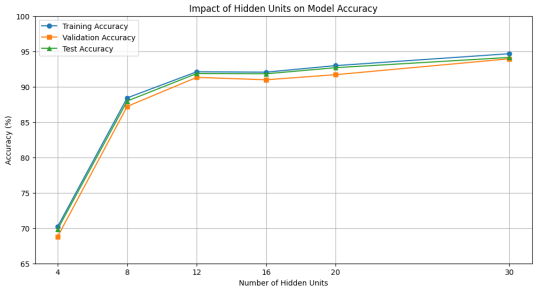

Fig. 2. figure 2

The plot in figure 2 shows that increasing the number of hidden units improves model performance up to a point. The most significant gains occur when increasing from 4 to 8 hidden units, indicating that a very small network was underfitting. Beyond about 12-16 hidden units, further increases lead to only marginal improvements in accuracy. The model does not exhibit significant overfitting even with a large number of hidden units (30), as validation and test accuracies remain close to training accuracy.

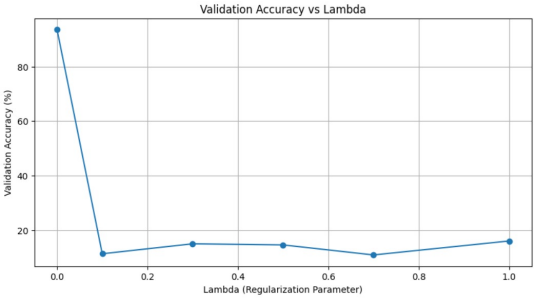

Fig. 3. figure 3

The graph in figure 3 shows the relationship between validation accuracy and the regularization parameter, lambda. As lambda increases from 0 to 1, the validation accuracy initially drops sharply from a high value to a much lower value. After this steep decline, the accuracy remains relatively low and stable as lambda continues to increase. This suggests that increasing regularization significantly reduces model performance after a certain point.

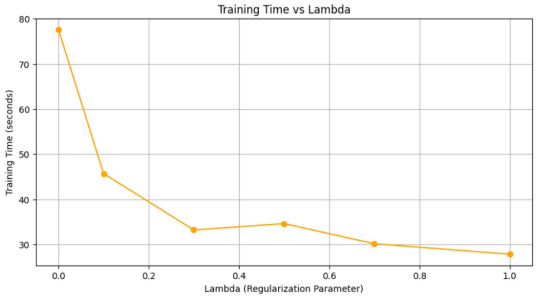

Fig. 4. figure 4

The graph in figure 4 shows the relationship between training time and lambda in a neural network. As lambda increases, the training time decreases significantly, particularly from 0 to 0.2, where the time drops sharply from around 80 seconds to 40 seconds. Beyond this point, the training time continues to decrease more gradually, stabilizing around 30 seconds as lambda approaches 1. This suggests that higher regularization leads to faster training times, likely due to reduced model complexity and overfitting.

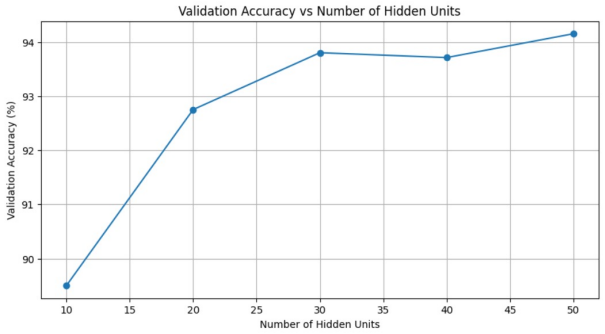

Fig. 5. figure 5

The graph in figure 5 illustrates the relationship between Validation Accuracy and the Number of Hidden Units in a neural network. As the number of hidden units increases the validation accuracy improves noticeably, reaching approximately 94%. Beyond 30 hidden units, the accuracy plateaus and fluctuates slightly, showing minimal gains up to 50 hidden units. This indicates that increasing the number of hidden units beyond a certain point offers diminishing returns in terms of validation accuracy.

Fig. 6. figure 6

The graph in figure 6 shows the relationship between training time and the number of hidden units in a neural network. As the number of hidden units increases the training time generally increases. Initially, there is a sharp rise in training time from 50 seconds at 10 hidden units to around 80 seconds at 20 hidden units. From 20 to 40 hidden units, the training time remains relatively stable, fluctuating slightly around 80 seconds. However, at 50 hidden units, there is a significant increase in training time, reaching approximately 95 seconds.

The best accuracy is achieved by choosing the hyper-parameters lambda as 0 and nhidden as 30. A training accuracy of 94.04%, a test accuracy of 93.84%, and a validation accuracy of 93.64%, is obtained indicating strong model performance across all datasets.

II. Accuracy of classification method on the handwritten digits test data

```
preprocess done
Training Time: 80.69 seconds

 Training set Accuracy:94.04400000000001%

 Validation set Accuracy:93.64%

 Test set Accuracy:93.84%
```

Fig. 7. figure 7

The classification method on the handwritten digits dataset achieved a Training set Accuracy of 94.04%, a Validation set Accuracy of 93.64%, and a Test set Accuracy of 93.84%. The model training is done in approximately in 80.69 seconds.

III. Accuracy of classification method on the CelebA data set

```
(myenv) C:\Users\ideal>python "C:\Users\ideal\Downloads\facennScript.py"
Training Time: 288.85 seconds

 Training set Accuracy:87.76303317535546%

 Validation set Accuracy:87.3545966228893%

 Test set Accuracy:87.66086298258895%
```

Fig. 8. figure 8

The classification method on the CelebA dataset achieved an accuracy of 87.76% on the training set, 87.35% on the validation set, and 87.66% on the test set. The model training is done in approximately 288.85 seconds.

IV. Comparison of your neural network with a deep neural network (using TensorFlow) in terms of accuracy and training time

```
preprocess done
Training Time: 80.69 seconds

 Training set Accuracy:94.04400000000001%

 Validation set Accuracy:93.64%

 Test set Accuracy:93.84%
```

Fig. 9. figure 9

```
(myenv) C:\Users\ideal>python "C:\Users\ideal\OneDrive\Desktop\final el\deepnnScript.py"
WARNING:tensorflow:From C:\Users\ideal\OneDrive\Desktop\final el\deepnnScript.py:81: softmax_cross_entropy_with_logits (from tensorflow.python.ops.nn_ops) i
s deprecated and will be removed in a future version.
Instructions for updating:

Future major versions of TensorFlow will allow gradients to flow
into the labels input on backprop by default.

See @{tf.nn.softmax_cross_entropy_with_logits_v2}.

2024-11-08 10:06:24.365888: I T:\src\github\tensorflow\tensorflow\core\platform\cpu_feature_guard.cc:140] Your CPU supports instructions that this TensorFlo
w binary was not compiled to use: AVX2
Optimization Finished!
Training Time: 191.52 seconds
Accuracy: 0.8865859
```

Fig. 10. figure 10

The comparison of neural network with a deep neural network in terms of accuracy and training time are visualized in the above images obtained by running nnScript.py and deepnnScript.py files.

V. Compare the results of deep neural network and neural network with one hidden layer on the CelebA data set

```
Training Single Layer Neural Network...
w1 shape: (256, 2377)
w2 shape: (2, 257)
output shape: (21100, 2)
a2 shape: (21100, 257)
a1 shape: (21100, 2377)
z2 shape: (21100, 256)
error shape: (21100, 2)
delta3 shape: (21100, 2)
delta2 shape: (21100, 257)
delta2[:, 1:] shape: (21100, 256)
Epoch 0, Accuracy: 0.4995260663507109
Single Layer NN Test Accuracy: 0.5

Comparison of Test Accuracies:
Deep Neural Network: 0.8353520035743713
Single Layer Neural Network: 0.5
```

Fig. 11. figure 10

The comparison between a Deep Neural Network and a Single Layer Neural Network on the CelebA dataset shows that the DNN significantly outperforms the SLNN. The DNN achieves a test accuracy of 83.53%, while the SLNN only reaches 50%. This difference highlights the ability of DNN to capture more complex features and patterns in the dataset due to its deeper architecture, whereas the SLNN, with its single hidden layer, struggles to model the complexity of the data effectively.

VI. Report the results from convolutional neural network in terms of accuracy and training time

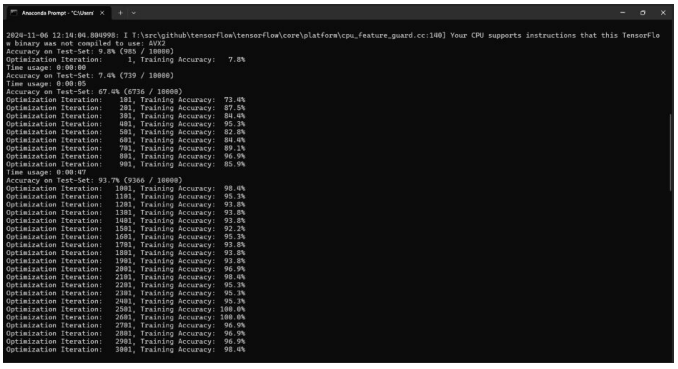

Fig. 12. figure 11

The images of output are obtained by running a Convolutional Neural Network (CNN) training script on MNIST dataset. The script performs multiple optimization iterations, displaying the training accuracy at each step. Initially, the accuracy on the test set starts low at 7.4% but gradually improves as training progresses. After 10,000 iterations, the training accuracy reaches 98.4%, and the final test set accuracy is 98.7% (9872/10000), indicating that the model has learned to generalize well to unseen data. The entire process took approximately 9 minutes.

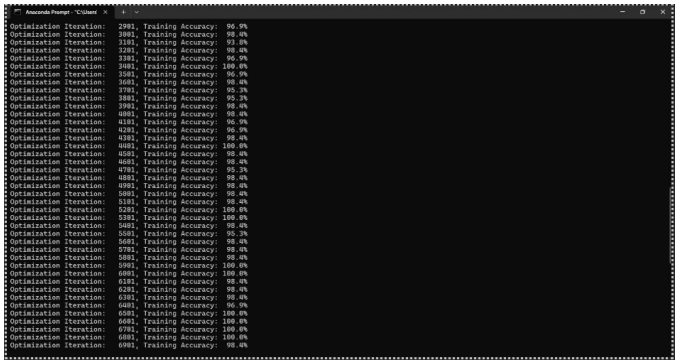

Fig. 13. figure 12

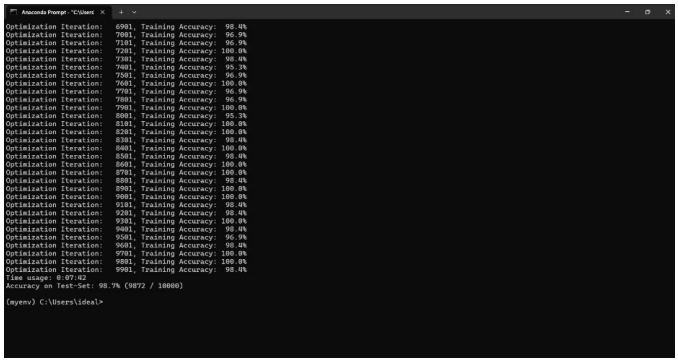

Fig. 14. figure 13