Machine Learning Assignment 1

Report

Baye-Saliou Fall

22 April 2024

Introduction

The information laid out in this document is relating to the use of Artificial Neural Networks within the field of Machine Learning.

A feedforward artificial neural network model to predict the label of a handwritten digit was used for the assignment. The data used was the MNIST dataset which has the images of the handwritten digits. The digits are from 0 to 9.

The main objective of the assignment is to find the best parameters for the model, namely, Learning Rate, Number of layers within the hidden layer, Activation Function, Optimiser and Epochs, to effectively predict the labels of images that are given as input to the model by the user.

Findings

Regularisation

The method used for regularisation is the dropout method. This dropout method reduces feature co-adaptation. The rate used is 0.3, as mentioned in an article by Andrian Tam, the best rate should be between 20% and 50%, anything lower will lead to minimal effect (Tam A. 2023).

Furthermore, the rate should not be too high as this may damage the data making it difficult for the model to predict.

Number of Layers within the hidden layer

According to an article written by Mausam Gaurav in 2019, the optimum number of layers within a hidden layer is 3 (Gaurav M. 2019). Mausam conducted an experiment to prove this using the GridSearchCV method (Gaurav M. 2019). Therefore, the Artificial Neural Network used in this assessment has 3 layers within the hidden layer.

Activation Function

The activation function used in the hidden layer is ReLu because it does not allow neurons that produce negative input for the next layer to get activated and this makes activation easy for the network (Aswini S. 2023). Moreover, the easier the activation the better the model.

The softmax function is used in the model for the output layer as it produces a probability distribution of the classes- 0 to 9. As seen in figures 1.1 and 1.2, the model learns slower with softmax function compared to “nn.Linear(hidden_size, num_classes)” , but it is better when it comes to predicting.

Figure 1.1 nn.Linear(hidden_size, num_classes)

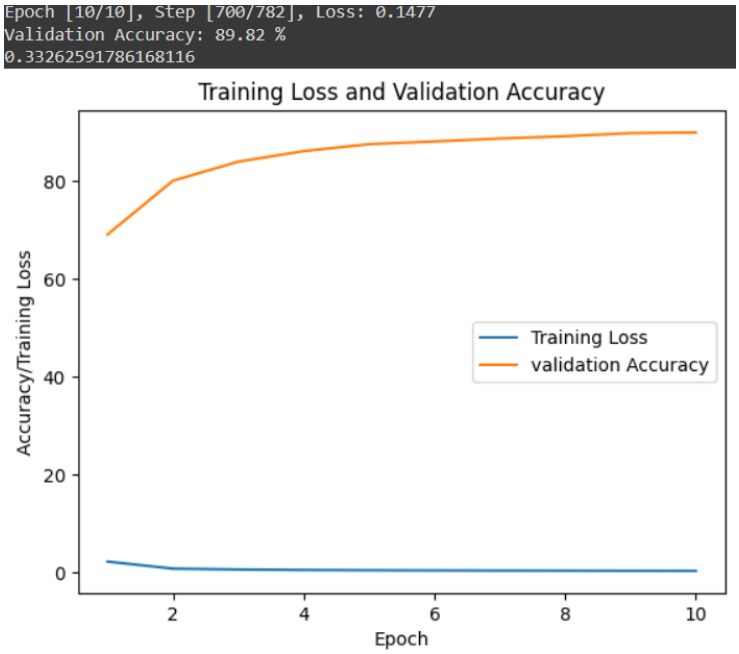
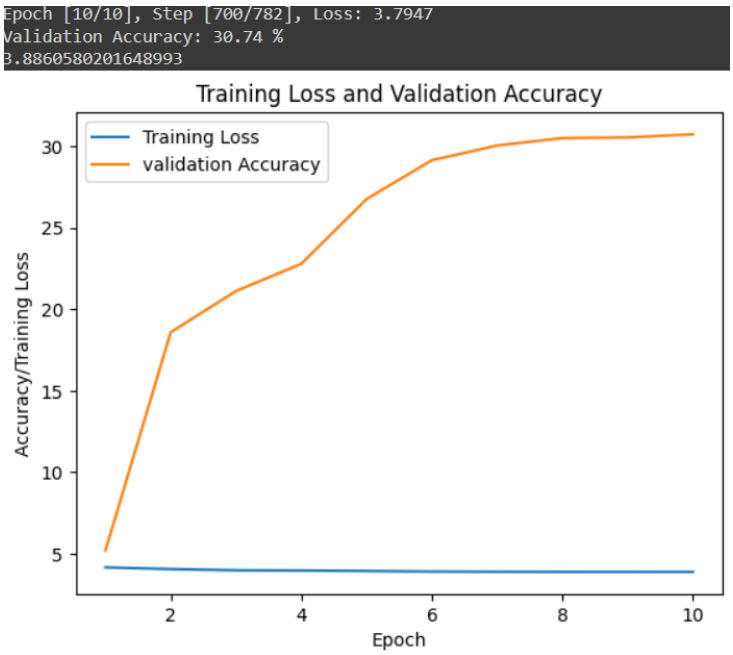


Figure 1.2 Softmax



Optimiser

The optimiser used in the experiment is the Adam instead of the SGD, gradient descent optimiser. This decision was aided by an experiment held by freeCodeCamp to find the best optimizer and learning rate (freeCodeCamp. 2018).

Please see the figures below illustrating how the two optimizers perform relative to each other. As seen in the figures below Adam produces higher validation accuracy than SGD, and a lower loss than SGD.

Figure 2.1 Adam

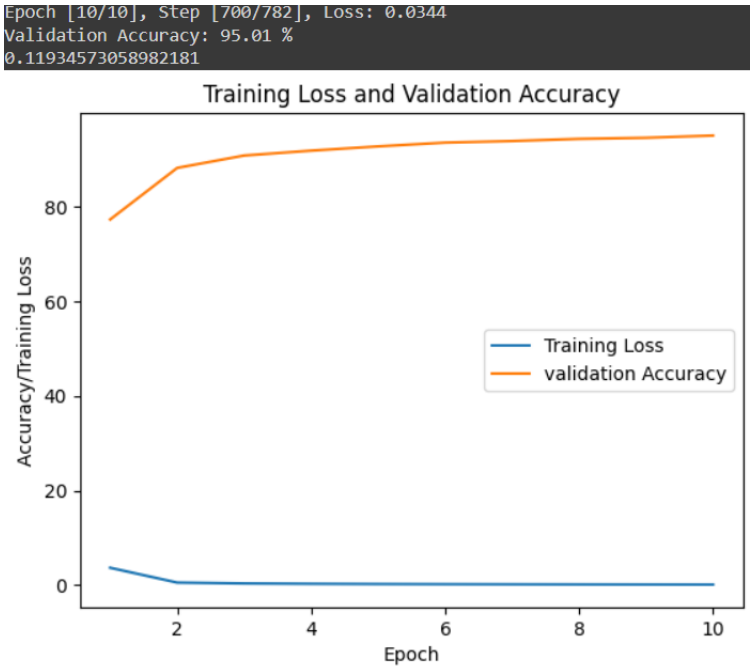
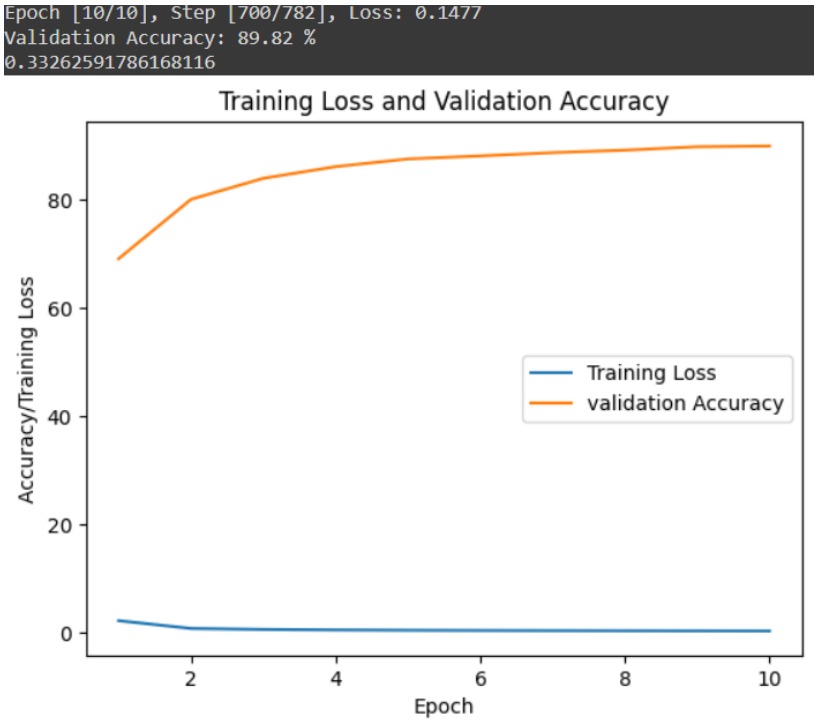


Figure 2.2 SGD



Epochs and Learning Rate

For the number of epochs, it is better to have many, but because the model is not implementing a stopping criterion, 100 epochs are used in the model. This ensures that the model has enough time to train and does not take too long either.

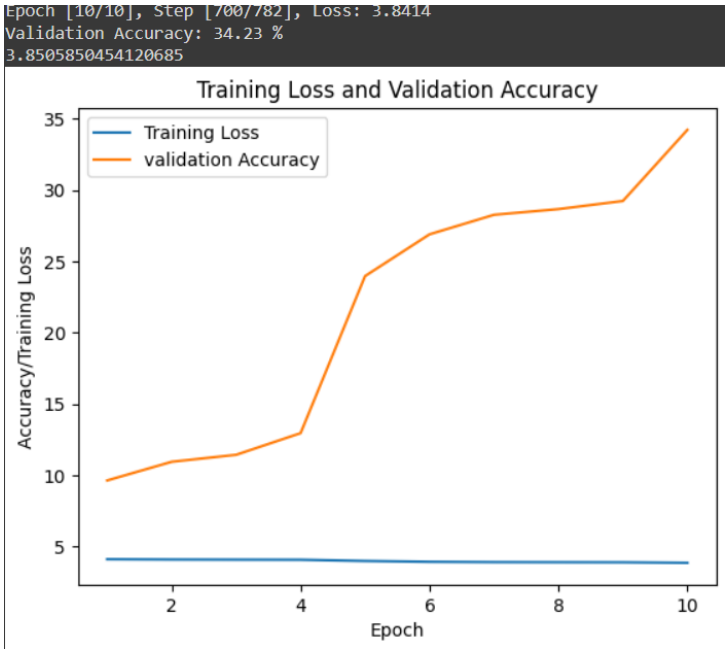
As for the learning rate used was 0.0001.

Conclusion

Before the assessment took place, the default values were as follows:

- Learning Rate = 0.0001
- Epochs = 10
- No regularisation
- One layer within hidden layer
- Optimizer = SGD
- Activation functions, ReLu(Hidden layer) and softmax (output layer)

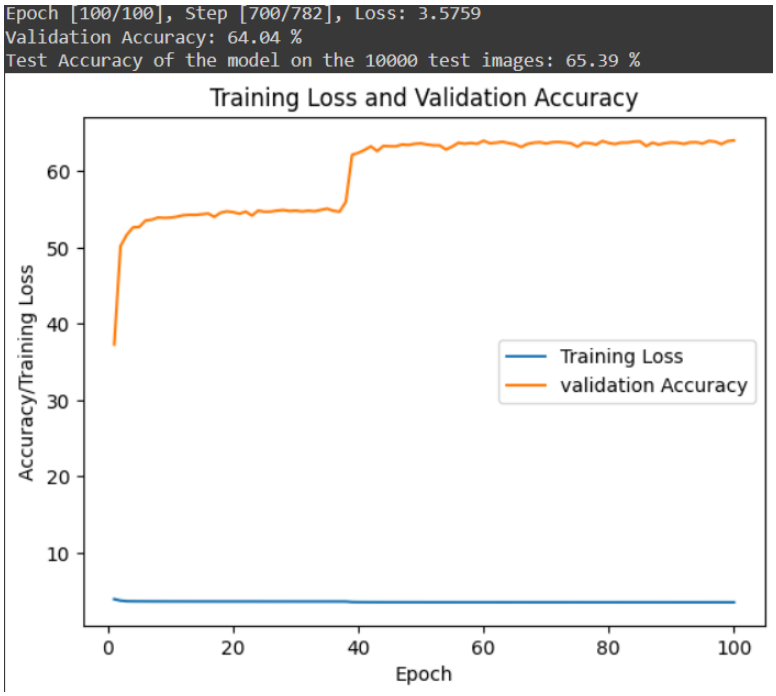
Figure 3.1 Results with default values for parameters.



After some investigation the final parameters are:

- Learning rate = 0.0001
- Epochs = 100
- Dropout Regularisation method
- 3 Layers within hidden layer
- Optimizer = Adam
- Activation functions, ReLu(Hidden layer) softmax (output layer)

Figure 3.2 Results with final values for parameters.



These results were obtained from the assessment.

References

- Gaurav M. 2019 ,**How the find the optimum number of hidden layers and nodes in a neural network model?,** Available at (How to find the optimum number of hidden layers and nodes in a neural network model? (datagraphi.com))
- Tam A. 2023 ,**Using Dropout Regularization in PyTorch Models ,** Available at (Using Dropout Regularization in PyTorch Models - MachineLearningMastery.com)
- Aswini S. 2023 ,**RELU and SIGMOID activation functions in Neural Networks,** Available at (https://www.shiksha.com/online-courses/articles/relu-and-sigmoid-activation-function/#:~:text=Advantage%20of%20ReLU%20Function%3A,the%20network%20easy%20for%20computation.)