Lab 5 Report — A Little Linear Regression

Jack Hilton-Jones (jhj1g23@soton.ac.uk)

Within this lab, a Convolution Neural Network (CNN) is implemented which takes an image of a scatter plot and predicts the parameters of the line of best fit.

I. A SIMPLE CNN BASELINE

To begin, a CNN model is implemented and trained using the "Adam" optimiser for 100 epochs and batches of 128 items. Mean Squared Error (MSE) is implemented as the loss function, chosen due to its ability to predict continuous output values. MSE calculates the distance between the predicted and the actual values while also being sensitive to outliers, heavily penalising them due to the squaring of errors. It is also differentiable, this is crucial for the Adam optimiser, ensuring there is a gradient at every point. Therefore, when using the Adam optimiser, the MSE loss function is chosen as it complements the optimiser well. The CNN is built, and the results can be seen in Table I.

Metric	Value
Training MSE Loss	0.628
Test MSE Loss	14.544

TABLE I
TRAINING AND TEST LOSS METRICS

The results shown from the CNN built with the results from Table I suggest a low training loss indicating the model has been able to learn the training data well. However, the model does not generalise well over new data, therefore, a very high test MSE is shown. The model is overfitting to training data, capturing noise and irrelevant patterns. This is because the model is too complex due to the many parameters in its final hidden layers.

II. A SIMPLE CNN WITH GLOBAL POOLING

The CNN built in Section I is overfitted due to the amount of parameters in its final hidden layers, this will be fixed in this section. To do this, Global Max Pooling is used to flatten the feature maps into a vector.

Metric	Value
Training MSE Loss	12.612
Test MSE Loss	14.935

TABLE II
TRAINING AND TEST LOSS METRICS

From Table II the model's training loss and test loss are high, as the model is not able to learn the data and provide an accurate fit. Even though the train and test loss is close, this is irrelevant as they both provide a high loss and underfitting.

III. LET'S REGRESS

In the final section, the CNN from Section I and II are modified to improve the performance of the model. Firstly, the number of input channels to the first convolutional layer is changed from 1 to 3. In addition, the input for the forward pass is modified, and the code turns the input into a 3 channel, where the first channel is the original data and the second and third channels represent normalised coordinate grids, which are stacked. Adding these channels provides spatial information to the model which is crucial to identify the position of the line within the image as this allows the model to learn where the features are located and also what features are present. Combining these two additions to the original CNN will provide the model with more information to accurately predict the parameters of the line of best fit within an image. The model is trained and then tested on the test data, which is shown in Table III and in Figure 1.

Metric	Value
Training MSE Loss	1.073
Test MSE Loss	1.570

TABLE III
TRAINING AND TEST LOSS METRICS

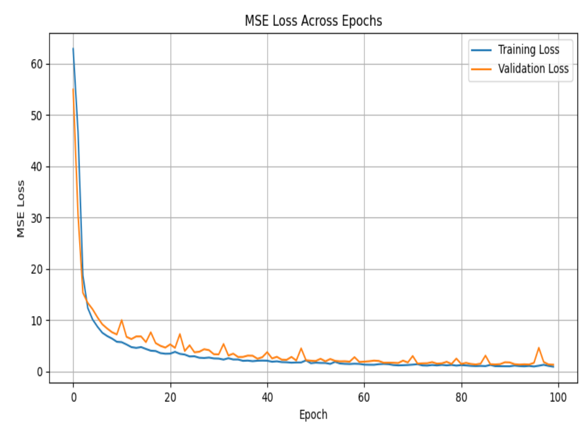

Fig. 1. CNN with modified input channels MSE training loss over the number of epochs

The results from the modified CNN show much improvement from the models found in Section I and II. It provides a low training loss and test loss, which shows the model has been able to learn the training data well while not overfitting. It can generalise well over new data, proven by a low test loss. The low test and training loss provides evidence for the increased spatial information and increased amount of channels added to the network.