**LeNet5 Implementation using Tensorflow**

**Brief Report and Analysis Report**

MNIST dataset contains images of handwritten digits.

The dataset is loaded from a CSV file and pre-processed by reshaping, normalizing, and splitting into training and test sets.

There are two models created using different activation functions to check the accuracy.

The first LeNet5 model has the following layers:

Conv2D layer with 32 filters, kernel size of (5,5), and activation function 'tanh'

MaxPooling2D layer with pool size (2,2)

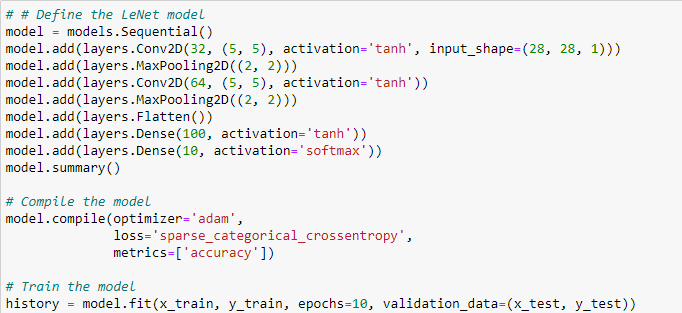
Conv2D layer with 64 filters, kernel size of (5,5), and activation function 'tanh'

MaxPooling2D layer with pool size (2,2)

Flatten layer

Dense layer with 100 neurons and activation function 'tanh'

Dense output layer with 10 neurons and activation function 'softmax'



The second LeNet5 model has the following layers:

Conv2D layer with 6 filters, kernel size of (5,5), and activation function 'relu'

MaxPooling2D layer with pool size (2,2)

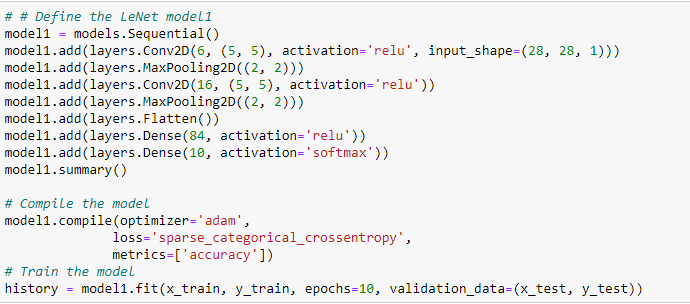
Conv2D layer with 16 filters, kernel size of (5,5), and activation function 'relu'

MaxPooling2D layer with pool size (2,2)

Flatten layer

Dense layer with 84 neurons and activation function 'relu'

Dense output layer with 10 neurons and activation function 'softmax'



Both models are trained using 'adam' optimizer, 'sparse\_categorical\_crossentropy' loss function, and 'accuracy' metric for 10 epochs on the training data with a batch size of 32.

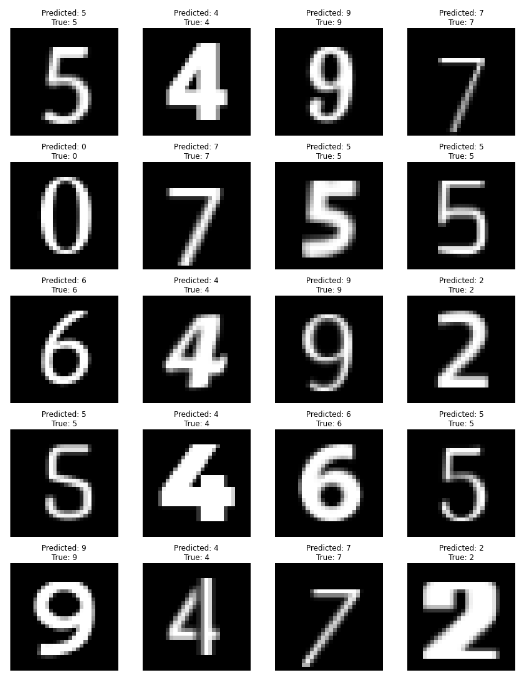
The first model achieves a test accuracy of 98.91%, while the second model achieves a test accuracy of 97.62%.

The first model outperforms the second one slightly.

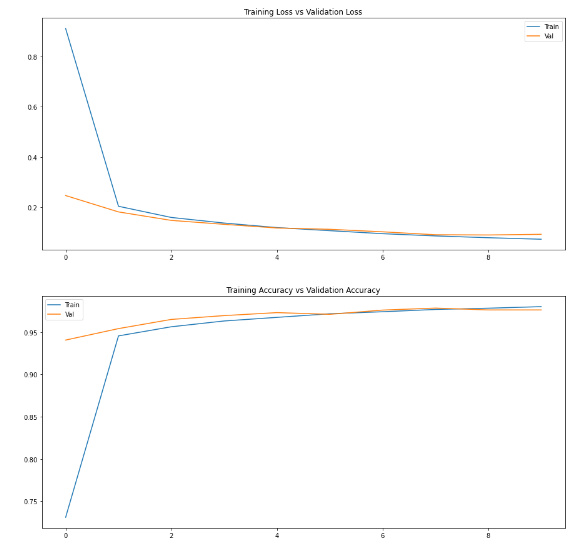
Finally, the code plots some random test images with their predicted and actual labels for the first model using matplotlib.

**Predictions using first model**

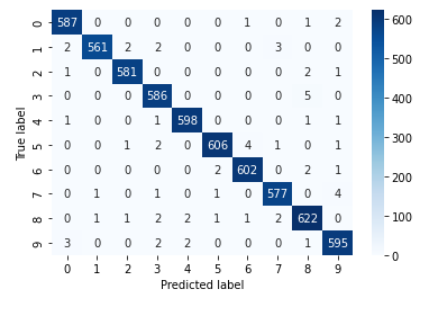
Below is a screenshot of predicted values and the actual values



Below is a screenshot of training loss vs Validation loss and Training accuracy vs Validation Accuracy

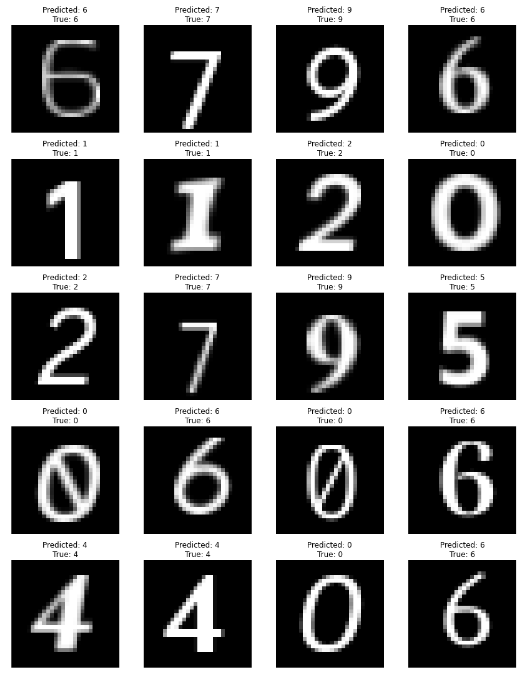


Below is a screenshot of correlation of predicted values and the actual values

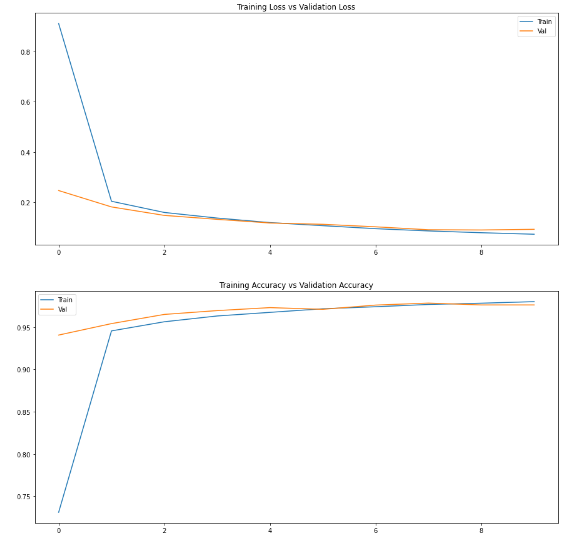


**Predictions using second model**

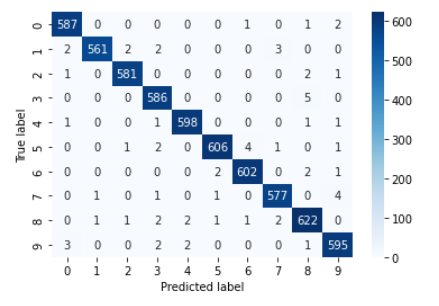
Below is a screenshot of predicted values and the actual values



Below is a screenshot of training loss vs Validation loss and Training accuracy vs Validation Accuracy



Below is a screenshot of correlation of predicted values and the actual values



**References:**

<https://www.kaggle.com/code/niranjanjagannath/lenet-5-architecture-for-mnist-using-tensorflow>

<https://gist.github.com/Moataz-E/6751b1b92fe8f4ff617f10c7f9f9d315>