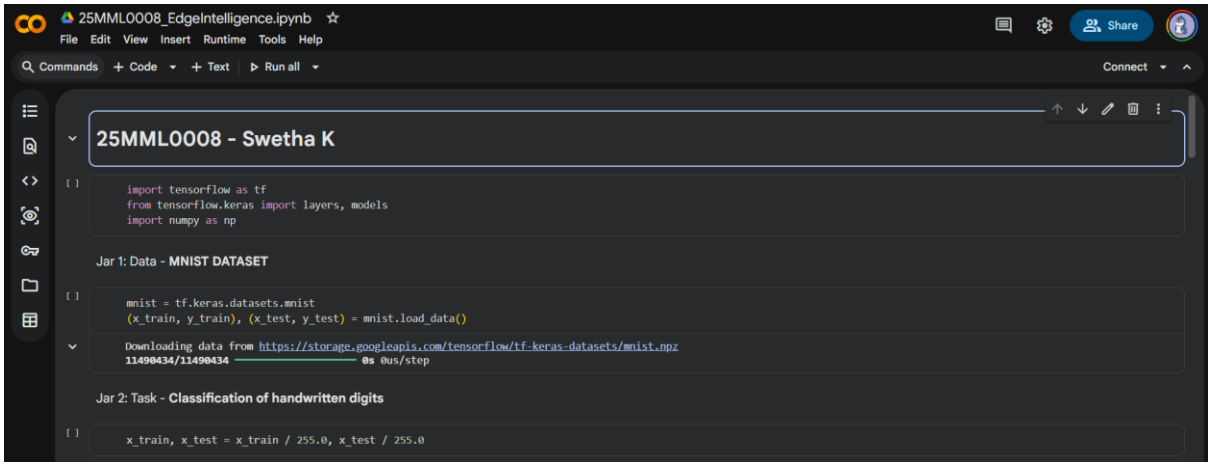
EDGE INTELLIGENCE LAB ASSIGNMENT 2

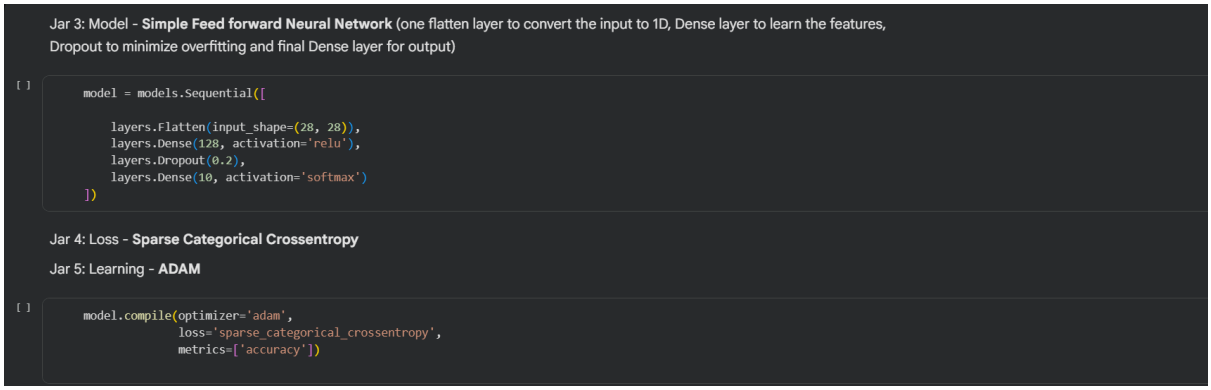
SWETHA K

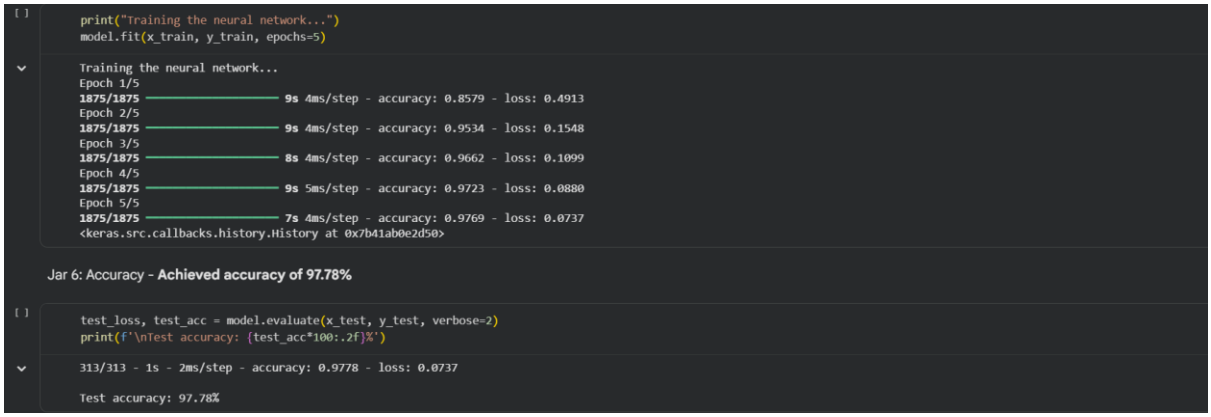
25MML0008

Task 1:

Analysing MNIST dataset applying Aritificial Neural Network, saving the model and finding the weight of the model







```
predictions = model.predict(x_test)
print(f"Prediction for first test image: {np.argmax(predictions[0])}")
print(f"Actual label: {y_test[0]}")
```

```
313/313 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step
Prediction for first test image: 7
Actual label: 7
```

Saving the model

```
import pickle
weights = model.get_weights()
with open('mnist_weights.pkl', 'wb') as f:
    pickle.dump(weights, f)
```

Weight of the model

```
import os

# Get the size in bytes
file_size_bytes = os.path.getsize('mnist_weights.pkl')

# Convert bytes to KB
file_size_kb = file_size_bytes / 1024

print(f"File Size: {file_size_kb:.2f} KB")
```

```
File Size: 397.80 KB
```

Load the model

```
with open('mnist_weights.pkl', 'rb') as f:
    loaded_weights = pickle.load(f)
model.set_weights(loaded_weights)
```

Using the pickled model to get the accuracy from test data

```
# Load MNIST test data (ensure it's normalized to 0-1)
(_, _), (x_test, y_test) = tf.keras.datasets.mnist.load_data()
x_test = x_test / 255.0

# Evaluate
loss, accuracy = model.evaluate(x_test, y_test, verbose=0)

print(f"Test Loss: {loss:.4f}")
print(f"Test Accuracy: {accuracy * 100:.2f}%")
```

```
Test Loss: 0.0737
Test Accuracy: 97.78%
```

```
model.summary()
```

```
model.summary()
```

Model: "sequential_1"

Layer (type)	Output Shape	Param #
flatten_1 (Flatten)	(None, 784)	0
dense_2 (Dense)	(None, 128)	100,480
dropout_1 (Dropout)	(None, 128)	0
dense_3 (Dense)	(None, 10)	1,290

Total params: 305,312 (1.16 MB)
Trainable params: 101,770 (397.54 KB)
Non-trainable params: 0 (0.00 B)
Optimizer params: 203,542 (795.09 KB)

Findings: Since the model is using ANN (Simple Neural Network) the number of parameters of the model are lesser compared to a CNN model, thus the weight of the model is 397.54KB.

Task 2:

1. Creating an account on Edge Impulse
2. Data acquisition
3. Choosing data connect -> Scan QR through phone
4. Selecting labels and uploading dataset
5. Splitting dataset into train and test.