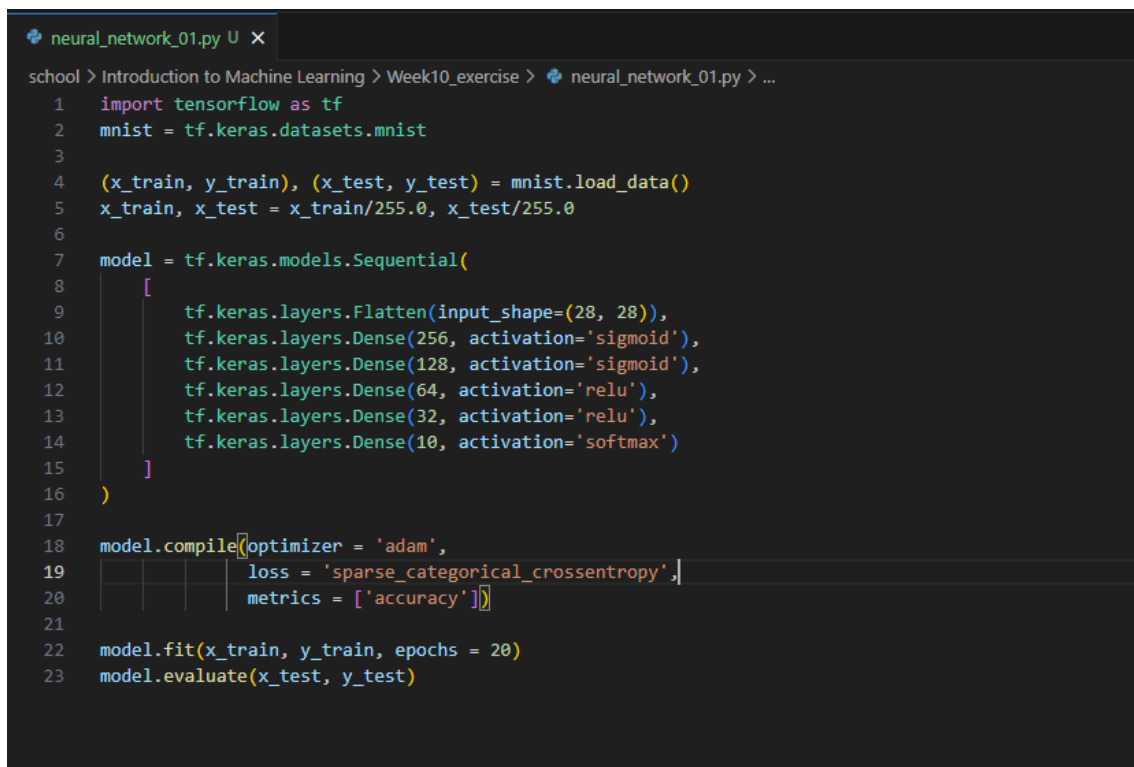


Week 10 Report

Neural Network 01

37237256 薛丁銘

The code of this week is as shown in Figure 1. Here I changed several parameters and I added layers of the network. The result showed that 'adam' for the optimizer and 'sparse_categorical_crossentropy' for the loss function leads to the best result. After 20 epochs, the accuracy reached 99.76%. The results are as shown in Figure 2.



```
neural_network_01.py U X
school > Introduction to Machine Learning > Week10_exercise > neural_network_01.py > ...
1 import tensorflow as tf
2 mnist = tf.keras.datasets.mnist
3
4 (x_train, y_train), (x_test, y_test) = mnist.load_data()
5 x_train, x_test = x_train/255.0, x_test/255.0
6
7 model = tf.keras.models.Sequential(
8     [
9         tf.keras.layers.Flatten(input_shape=(28, 28)),
10        tf.keras.layers.Dense(256, activation='sigmoid'),
11        tf.keras.layers.Dense(128, activation='sigmoid'),
12        tf.keras.layers.Dense(64, activation='relu'),
13        tf.keras.layers.Dense(32, activation='relu'),
14        tf.keras.layers.Dense(10, activation='softmax')
15    ]
16)
17
18 model.compile(optimizer = 'adam',
19               loss = 'sparse_categorical_crossentropy',
20               metrics = ['accuracy'])
21
22 model.fit(x_train, y_train, epochs = 20)
23 model.evaluate(x_test, y_test)
```

Figure 1. Code

```

PS D:\code> cd 'd:\code'; & 'c:\python379\python.exe' 'c:\Users\xding\.vscode\extensions\ms-python.python-2023.22.
\school\Introduction to Machine Learning\Week10_exercise\neural_network_01.py'
2024-01-05 03:52:36.842553: I tensorflow/core/platform/cpu_feature_guard.cc:193] This TensorFlow binary is optimized with
n performance-critical operations: AVX AVX2
To enable them in other operations, rebuild TensorFlow with the appropriate compiler flags.
Epoch 1/20
1875/1875 [=====] - 8s 4ms/step - loss: 0.4163 - accuracy: 0.8719
Epoch 2/20
1875/1875 [=====] - 8s 4ms/step - loss: 0.1499 - accuracy: 0.9543
Epoch 3/20
1875/1875 [=====] - 8s 4ms/step - loss: 0.1022 - accuracy: 0.9691
Epoch 4/20
1875/1875 [=====] - 8s 4ms/step - loss: 0.0759 - accuracy: 0.9761
Epoch 5/20
1875/1875 [=====] - 8s 4ms/step - loss: 0.0586 - accuracy: 0.9815
Epoch 6/20
1875/1875 [=====] - 9s 5ms/step - loss: 0.0477 - accuracy: 0.9847
Epoch 7/20
1875/1875 [=====] - 9s 5ms/step - loss: 0.0372 - accuracy: 0.9881
Epoch 8/20
1875/1875 [=====] - 9s 5ms/step - loss: 0.0309 - accuracy: 0.9899
Epoch 9/20
1875/1875 [=====] - 9s 5ms/step - loss: 0.0264 - accuracy: 0.9915
Epoch 10/20
1875/1875 [=====] - 8s 4ms/step - loss: 0.0221 - accuracy: 0.9926
Epoch 11/20
1875/1875 [=====] - 9s 5ms/step - loss: 0.0187 - accuracy: 0.9941
Epoch 12/20
1875/1875 [=====] - 9s 5ms/step - loss: 0.0176 - accuracy: 0.9941
Epoch 13/20
1875/1875 [=====] - 10s 6ms/step - loss: 0.0150 - accuracy: 0.9951
Epoch 14/20
1875/1875 [=====] - 9s 5ms/step - loss: 0.0120 - accuracy: 0.9961
Epoch 15/20
1875/1875 [=====] - 10s 5ms/step - loss: 0.0135 - accuracy: 0.9956
Epoch 16/20
1875/1875 [=====] - 19s 10ms/step - loss: 0.0111 - accuracy: 0.9961
Epoch 17/20
1875/1875 [=====] - 21s 11ms/step - loss: 0.0099 - accuracy: 0.9967
Epoch 18/20
1875/1875 [=====] - 21s 11ms/step - loss: 0.0100 - accuracy: 0.9966
Epoch 19/20
1875/1875 [=====] - 21s 11ms/step - loss: 0.0085 - accuracy: 0.9973
Epoch 20/20
1875/1875 [=====] - 21s 11ms/step - loss: 0.0074 - accuracy: 0.9976
313/313 [=====] - 2s 5ms/step - loss: 0.1171 - accuracy: 0.9790
PS D:\code> 

```

Figure 2. Results