NNDL_Assignment6

NAME:JYOSHNA YARRAGUNTLA

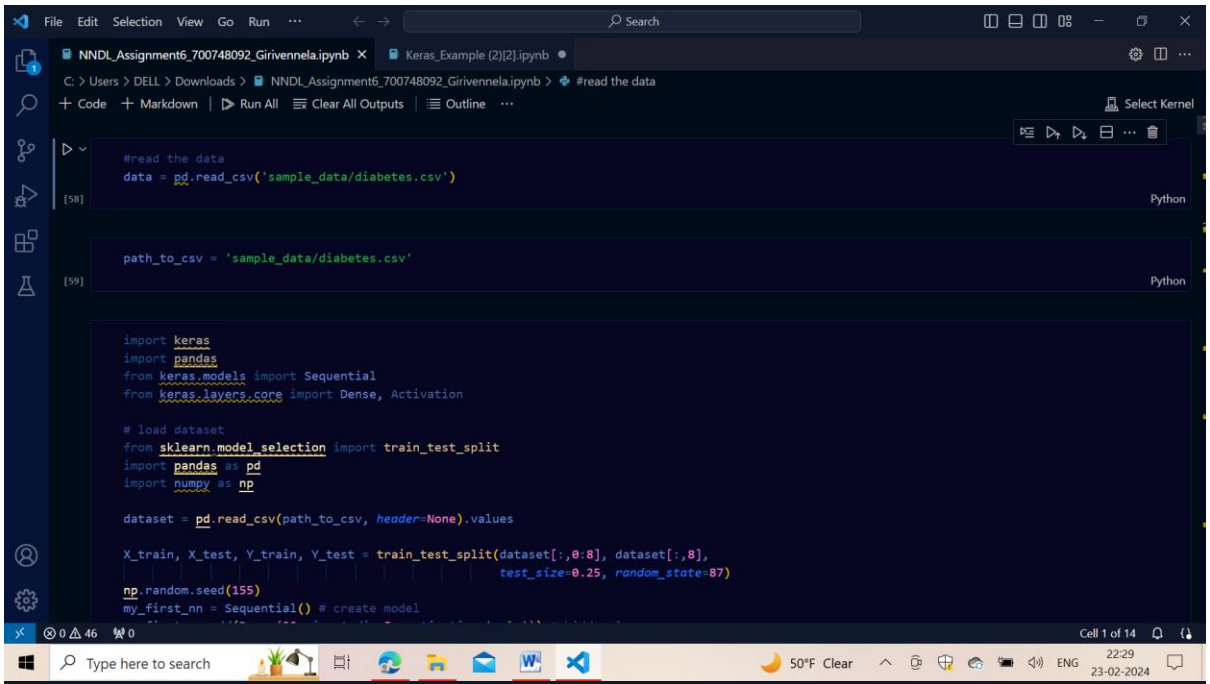
STUDENT ID:700759948

In class programming:

1. Reporting Accuracy of model using breast cancer dataset.
2. Normaling the data before feeding the data to the model and checking how the normalization changing the accuracy.
3. Ploting the loss and accuracy for both training data and validation data using the history object in the source code.
4. Ploting the images in the test data, and then do inferencing to check what is the prediction of the model on the single image.

GitHub:- https://github.com/Jyoshna200/neural-network-ass-6

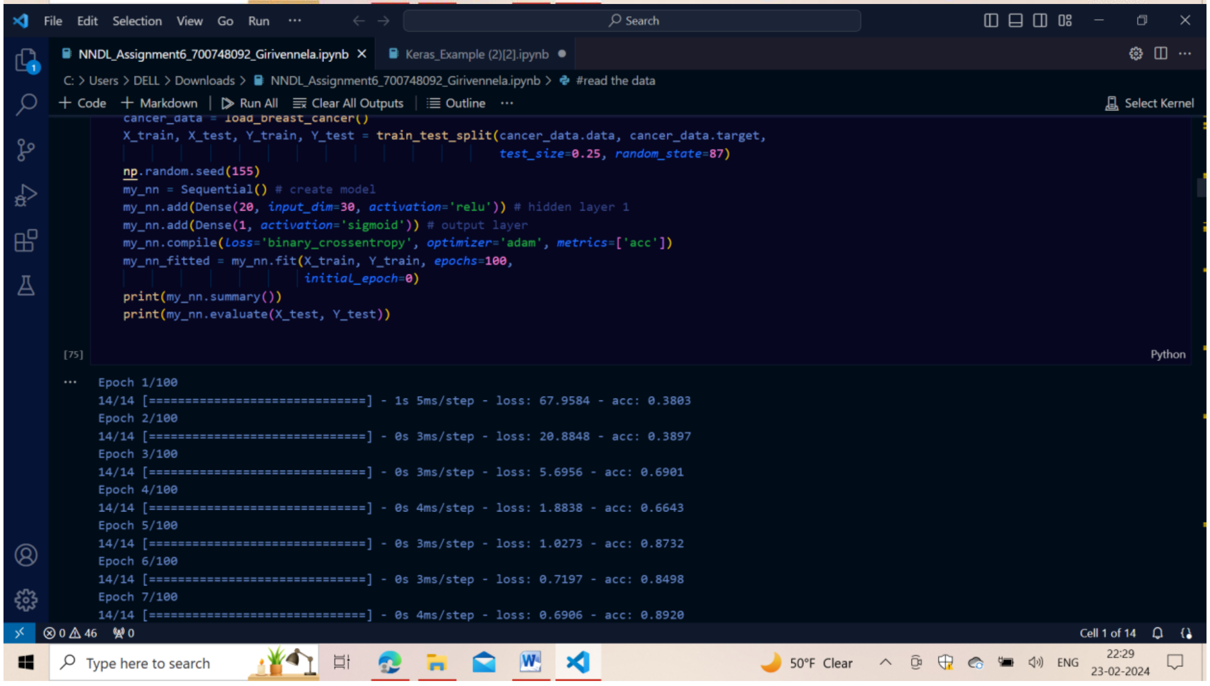
Video Link:- https://drive.google.com/file/d/1mr52OnNT0RyruuCeoz8HVnWbNeicDU2U/view?usp=sharing


File Edit Selection View Go Run
Search
NNDL_Assignment6_700748092_Girivennela.ipynb
Keras_Example (2)[2].ipynb
C: > Users > DELL > Downloads > NNDL_Assignment6_700748092_Girivennela.ipynb > #read the data
Code
Markdown
Run All
Clear All Outputs
Outline
Select Kernel
Python
Cell 1 of 14
Type here to search
50°F Clear
ENG
22:29
23-02-2024

```
#read the data
data = pd.read_csv('sample_data/diabetes.csv')
```
[58]

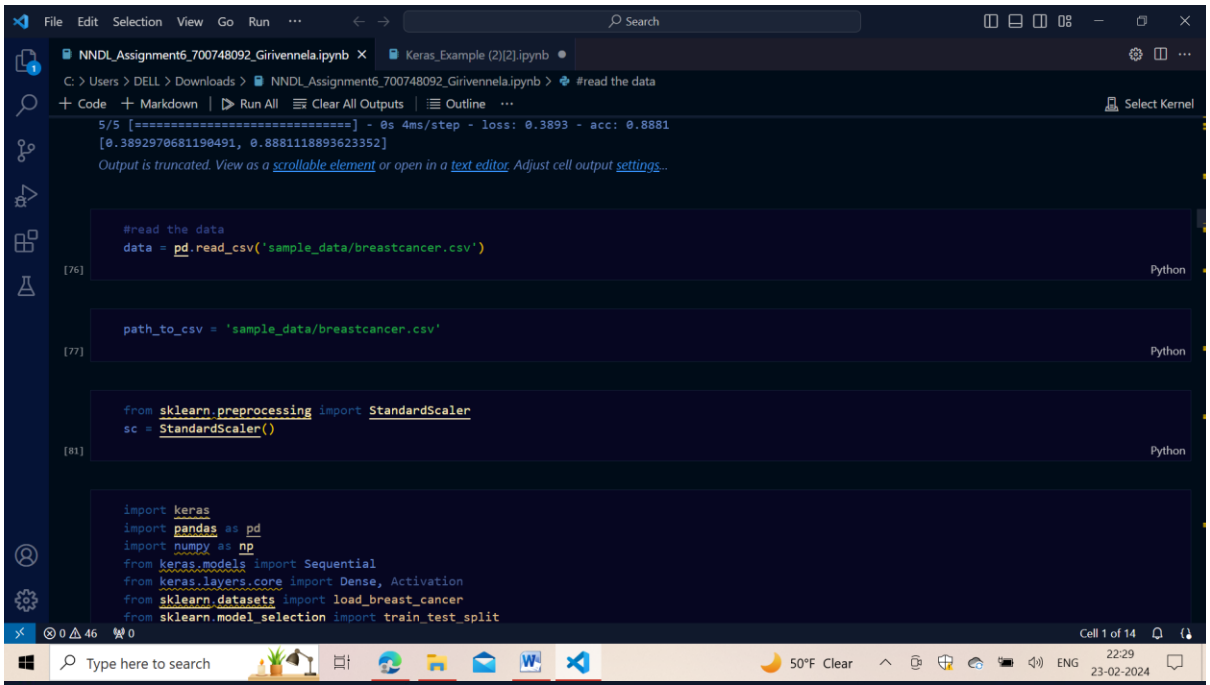
```
path_to_csv = 'sample_data/diabetes.csv'
```
[59]

```
import keras
import pandas
from keras.models import Sequential
from keras.layers.core import Dense, Activation

# load dataset
from sklearn.model_selection import train_test_split
import pandas as pd
import numpy as np

dataset = pd.read_csv(path_to_csv, header=None).values

X_train, X_test, Y_train, Y_test = train_test_split(dataset[:,0:8], dataset[:,8],
                                                    test_size=0.25, random_state=87)
np.random.seed(155)
my_first_nn = Sequential() # create model
```


File Edit Selection View Go Run
Search
NNDL_Assignment6_700748092_Girivennela.ipynb
Keras_Example (2)[2].ipynb
C: > Users > DELL > Downloads > NNDL_Assignment6_700748092_Girivennela.ipynb > #read the data
Code
Markdown
Run All
Clear All Outputs
Outline
Select Kernel
Python
Cell 1 of 14
Type here to search
50°F Clear
ENG
22:29
23-02-2024

```
cancer_data = load_breast_cancer()
X_train, X_test, Y_train, Y_test = train_test_split(cancer_data.data, cancer_data.target,
                                                    test_size=0.25, random_state=87)
np.random.seed(155)
my_nn = Sequential() # create model
my_nn.add(Dense(20, input_dim=30, activation='relu')) # hidden layer 1
my_nn.add(Dense(1, activation='sigmoid')) # output layer
my_nn.compile(loss='binary_crossentropy', optimizer='adam', metrics=['acc'])
my_nn_fitted = my_nn.fit(X_train, Y_train, epochs=100,
                         initial_epoch=0)
print(my_nn.summary())
print(my_nn.evaluate(X_test, Y_test))
```
[75]

```
Epoch 1/100
14/14 [==============================] - 1s 5ms/step - loss: 67.9584 - acc: 0.3803
Epoch 2/100
14/14 [==============================] - 0s 3ms/step - loss: 20.8848 - acc: 0.3897
Epoch 3/100
14/14 [==============================] - 0s 3ms/step - loss: 5.6956 - acc: 0.6901
Epoch 4/100
14/14 [==============================] - 0s 4ms/step - loss: 1.8838 - acc: 0.6643
Epoch 5/100
14/14 [==============================] - 0s 3ms/step - loss: 1.0273 - acc: 0.8732
Epoch 6/100
14/14 [==============================] - 0s 3ms/step - loss: 0.7197 - acc: 0.8498
Epoch 7/100
14/14 [==============================] - 0s 4ms/step - loss: 0.6906 - acc: 0.8920
```


File Edit Selection View Go Run ···
Search
NNDL_Assignment6_700748092_Girivennela.ipynb
Keras_Example (2)[2].ipynb
C: > Users > DELL > Downloads > NNDL_Assignment6_700748092_Girivennela.ipynb > #read the data
+ Code + Markdown Run All Clear All Outputs Outline
Select Kernel
Python
0 46 0
Cell 1 of 14
Type here to search
50°F Clear
ENG
22:29
23-02-2024

```
5/5 [==============================] - 0s 4ms/step - loss: 0.3893 - acc: 0.8881
[0.3892970681190491, 0.8881118893623352]
```

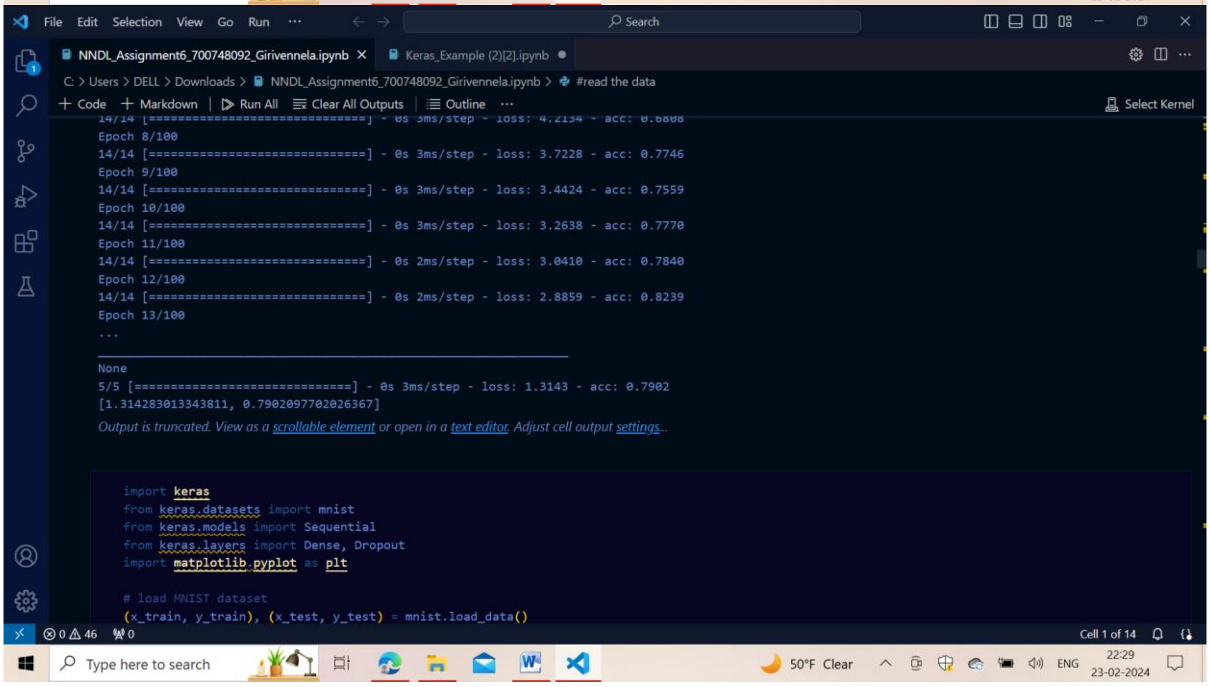
Output is truncated. View as a scrollable element or open in a text editor. Adjust cell output settings...

```
#read the data
data = pd.read_csv('sample_data/breastcancer.csv')
```
[76]

```
path_to_csv = 'sample_data/breastcancer.csv'
```
[77]

```
from sklearn.preprocessing import StandardScaler
sc = StandardScaler()
```
[81]

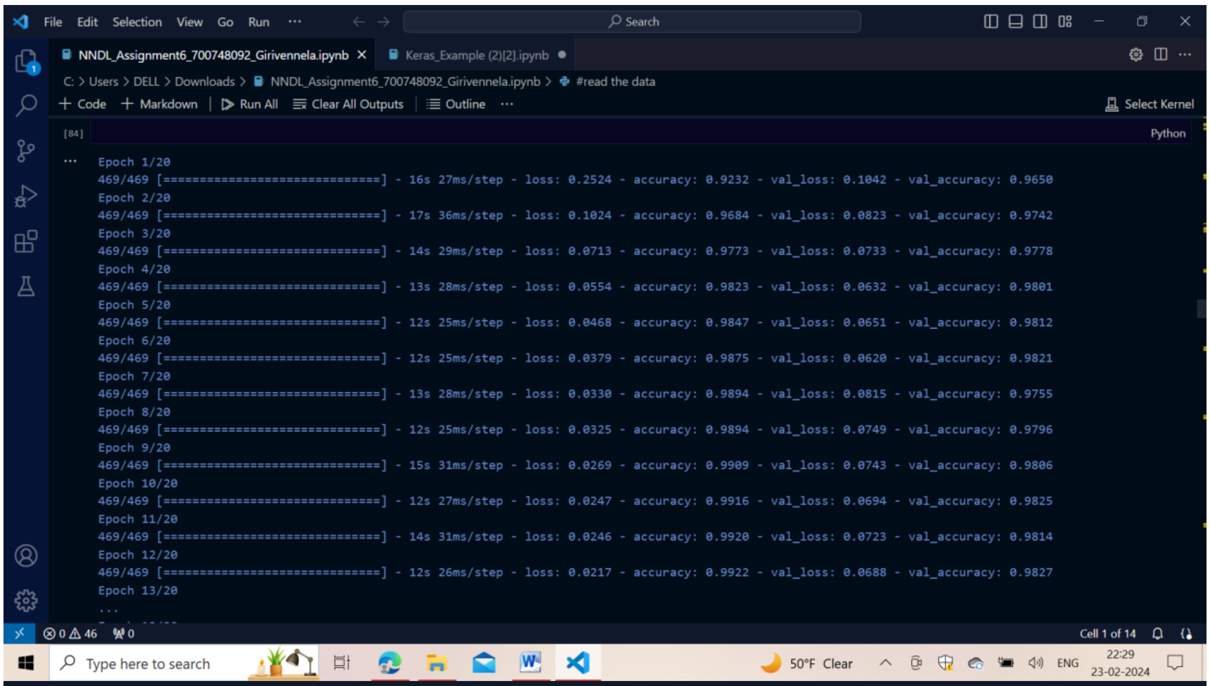
```
import keras
import pandas as pd
import numpy as np
from keras.models import Sequential
from keras.layers.core import Dense, Activation
from sklearn.datasets import load_breast_cancer
from sklearn.model_selection import train_test_split
```


File Edit Selection View Go Run ···
Search
NNDL_Assignment6_700748092_Girivennela.ipynb
Keras_Example (2)[2].ipynb
C: > Users > DELL > Downloads > NNDL_Assignment6_700748092_Girivennela.ipynb > #read the data
+ Code + Markdown Run All Clear All Outputs Outline
Select Kernel
0 46 0
Cell 1 of 14
Type here to search
50°F Clear
ENG
22:29
23-02-2024

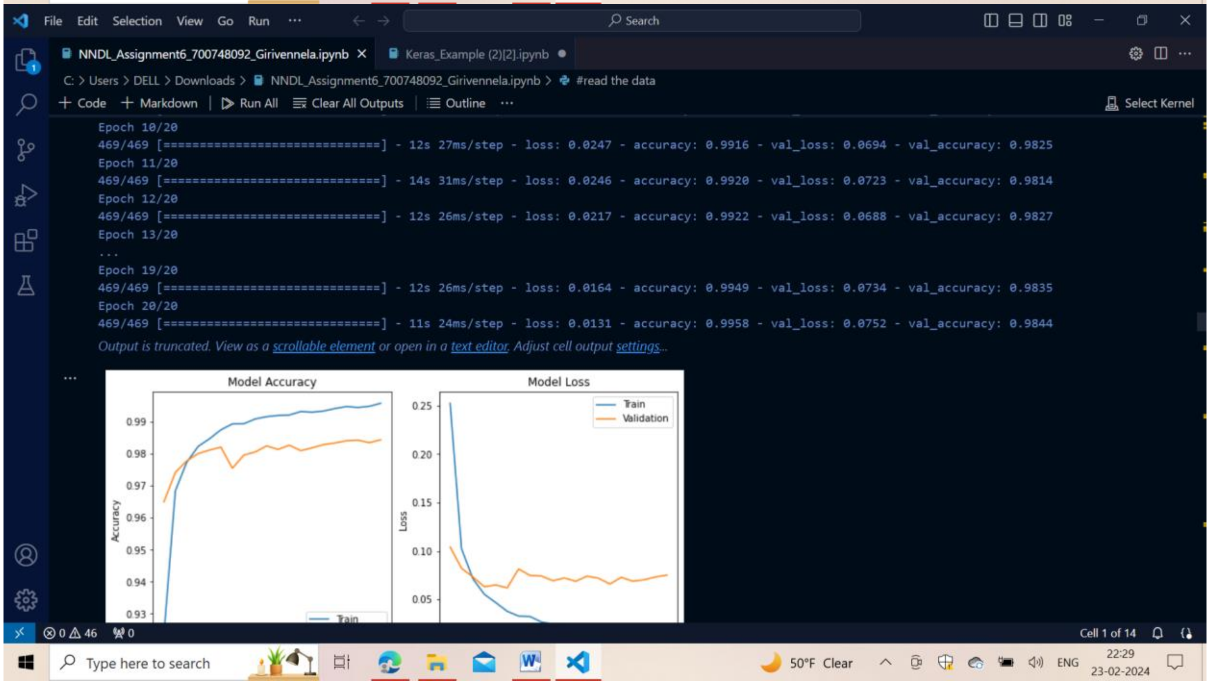
```
14/14 [==============================] - 0s 3ms/step - loss: 4.2134 - acc: 0.6808
Epoch 8/100
14/14 [==============================] - 0s 3ms/step - loss: 3.7228 - acc: 0.7746
Epoch 9/100
14/14 [==============================] - 0s 3ms/step - loss: 3.4424 - acc: 0.7559
Epoch 10/100
14/14 [==============================] - 0s 3ms/step - loss: 3.2638 - acc: 0.7770
Epoch 11/100
14/14 [==============================] - 0s 2ms/step - loss: 3.0410 - acc: 0.7840
Epoch 12/100
14/14 [==============================] - 0s 2ms/step - loss: 2.8859 - acc: 0.8239
Epoch 13/100
...

None
5/5 [==============================] - 0s 3ms/step - loss: 1.3143 - acc: 0.7902
[1.314283013343811, 0.7902097702026367]
```

Output is truncated. View as a scrollable element or open in a text editor. Adjust cell output settings...

```
import keras
from keras.datasets import mnist
from keras.models import Sequential
from keras.layers import Dense, Dropout
import matplotlib.pyplot as plt

# load MNIST dataset
(x_train, y_train), (x_test, y_test) = mnist.load_data()
```

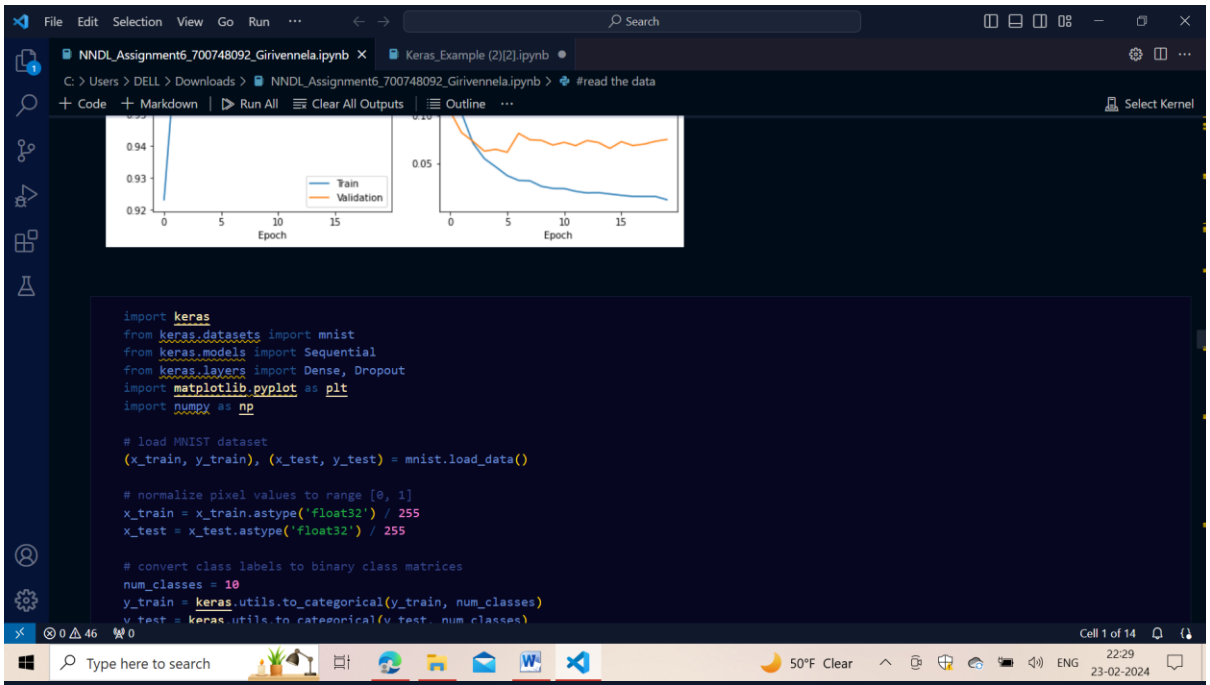

File Edit Selection View Go Run ···
Search
NNDL_Assignment6_700748092_Girivennela.ipynb
Keras_Example (2)[2].ipynb
C: > Users > DELL > Downloads > NNDL_Assignment6_700748092_Girivennela.ipynb > #read the data
+ Code + Markdown Run All Clear All Outputs Outline ···
Select Kernel
[84]
Python
Cell 1 of 14
Type here to search
50°F Clear
ENG
22:29
23-02-2024

```
Epoch 1/20
469/469 [==============================] - 16s 27ms/step - loss: 0.2524 - accuracy: 0.9232 - val_loss: 0.1042 - val_accuracy: 0.9650
Epoch 2/20
469/469 [==============================] - 17s 36ms/step - loss: 0.1024 - accuracy: 0.9684 - val_loss: 0.0823 - val_accuracy: 0.9742
Epoch 3/20
469/469 [==============================] - 14s 29ms/step - loss: 0.0713 - accuracy: 0.9773 - val_loss: 0.0733 - val_accuracy: 0.9778
Epoch 4/20
469/469 [==============================] - 13s 28ms/step - loss: 0.0554 - accuracy: 0.9823 - val_loss: 0.0632 - val_accuracy: 0.9801
Epoch 5/20
469/469 [==============================] - 12s 25ms/step - loss: 0.0468 - accuracy: 0.9847 - val_loss: 0.0651 - val_accuracy: 0.9812
Epoch 6/20
469/469 [==============================] - 12s 25ms/step - loss: 0.0379 - accuracy: 0.9875 - val_loss: 0.0620 - val_accuracy: 0.9821
Epoch 7/20
469/469 [==============================] - 13s 28ms/step - loss: 0.0330 - accuracy: 0.9894 - val_loss: 0.0815 - val_accuracy: 0.9755
Epoch 8/20
469/469 [==============================] - 12s 25ms/step - loss: 0.0325 - accuracy: 0.9894 - val_loss: 0.0749 - val_accuracy: 0.9796
Epoch 9/20
469/469 [==============================] - 15s 31ms/step - loss: 0.0269 - accuracy: 0.9909 - val_loss: 0.0743 - val_accuracy: 0.9806
Epoch 10/20
469/469 [==============================] - 12s 27ms/step - loss: 0.0247 - accuracy: 0.9916 - val_loss: 0.0694 - val_accuracy: 0.9825
Epoch 11/20
469/469 [==============================] - 14s 31ms/step - loss: 0.0246 - accuracy: 0.9920 - val_loss: 0.0723 - val_accuracy: 0.9814
Epoch 12/20
469/469 [==============================] - 12s 26ms/step - loss: 0.0217 - accuracy: 0.9922 - val_loss: 0.0688 - val_accuracy: 0.9827
Epoch 13/20
...
```

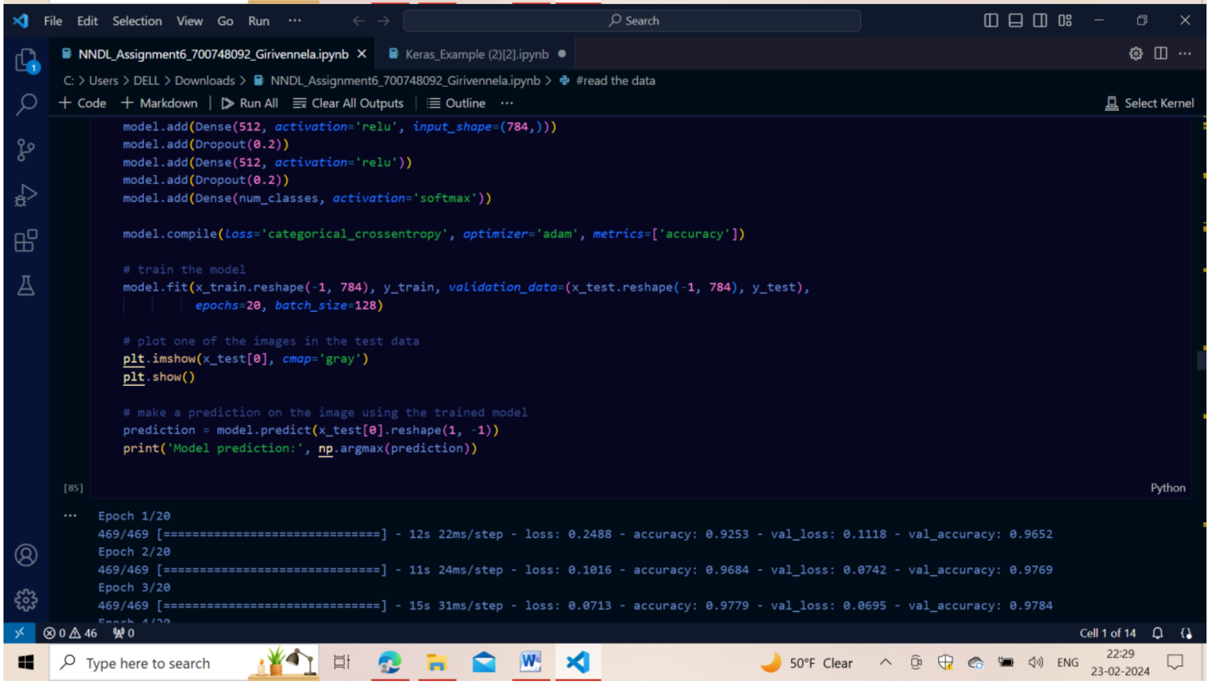

File Edit Selection View Go Run ···
Search
NNDL_Assignment6_700748092_Girivennela.ipynb
Keras_Example (2)[2].ipynb
C: > Users > DELL > Downloads > NNDL_Assignment6_700748092_Girivennela.ipynb > #read the data
+ Code + Markdown Run All Clear All Outputs Outline ···
Select Kernel
Model Accuracy
Accuracy
0.99
0.98
0.97
0.96
0.95
0.94
0.93
Train
Model Loss
Loss
0.25
0.20
0.15
0.10
0.05
Train
Validation
Cell 1 of 14
Type here to search
50°F Clear
ENG
22:29
23-02-2024

```
Epoch 10/20
469/469 [==============================] - 12s 27ms/step - loss: 0.0247 - accuracy: 0.9916 - val_loss: 0.0694 - val_accuracy: 0.9825
Epoch 11/20
469/469 [==============================] - 14s 31ms/step - loss: 0.0246 - accuracy: 0.9920 - val_loss: 0.0723 - val_accuracy: 0.9814
Epoch 12/20
469/469 [==============================] - 12s 26ms/step - loss: 0.0217 - accuracy: 0.9922 - val_loss: 0.0688 - val_accuracy: 0.9827
Epoch 13/20
...
Epoch 19/20
469/469 [==============================] - 12s 26ms/step - loss: 0.0164 - accuracy: 0.9949 - val_loss: 0.0734 - val_accuracy: 0.9835
Epoch 20/20
469/469 [==============================] - 11s 24ms/step - loss: 0.0131 - accuracy: 0.9958 - val_loss: 0.0752 - val_accuracy: 0.9844
```

Output is truncated. View as a scrollable element or open in a text editor. Adjust cell output settings...


File Edit Selection View Go Run ···
Search
NNDL_Assignment6_700748092_Girivennela.ipynb
Keras_Example (2)[2].ipynb
C: > Users > DELL > Downloads > NNDL_Assignment6_700748092_Girivennela.ipynb > #read the data
+ Code + Markdown Run All Clear All Outputs Outline
Select Kernel
0.94
0.93
0.92
Train
Validation
0.05
Epoch
Epoch
Cell 1 of 14
Type here to search
50°F Clear
ENG
22:29
23-02-2024

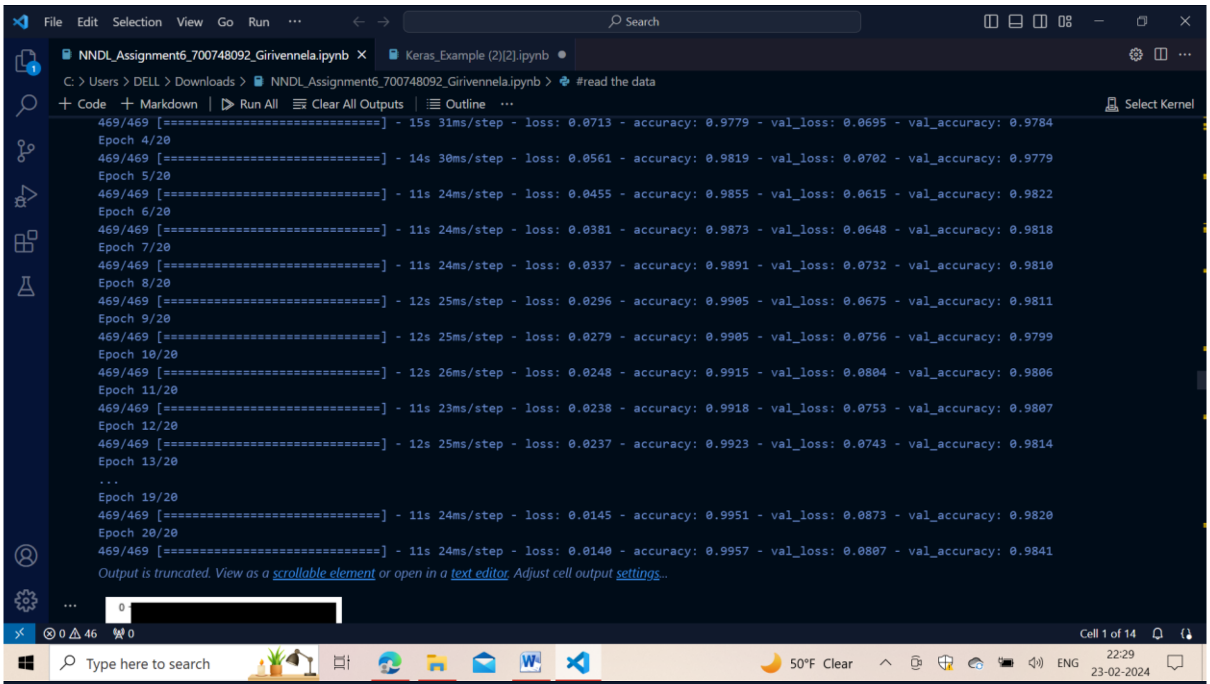
```
import keras
from keras.datasets import mnist
from keras.models import Sequential
from keras.layers import Dense, Dropout
import matplotlib.pyplot as plt
import numpy as np

# load MNIST dataset
(x_train, y_train), (x_test, y_test) = mnist.load_data()

# normalize pixel values to range [0, 1]
x_train = x_train.astype('float32') / 255
x_test = x_test.astype('float32') / 255

# convert class labels to binary class matrices
num_classes = 10
y_train = keras.utils.to_categorical(y_train, num_classes)
```


File Edit Selection View Go Run ···
Search
NNDL_Assignment6_700748092_Girivennela.ipynb
Keras_Example (2)[2].ipynb
C: > Users > DELL > Downloads > NNDL_Assignment6_700748092_Girivennela.ipynb > #read the data
+ Code + Markdown Run All Clear All Outputs Outline
Select Kernel
Python
Cell 1 of 14
Type here to search
50°F Clear
ENG
22:29
23-02-2024

```
model.add(Dense(512, activation='relu', input_shape=(784,)))
model.add(Dropout(0.2))
model.add(Dense(512, activation='relu'))
model.add(Dropout(0.2))
model.add(Dense(num_classes, activation='softmax'))

model.compile(loss='categorical_crossentropy', optimizer='adam', metrics=['accuracy'])

# train the model
model.fit(x_train.reshape(-1, 784), y_train, validation_data=(x_test.reshape(-1, 784), y_test),
          epochs=20, batch_size=128)

# plot one of the images in the test data
plt.imshow(x_test[0], cmap='gray')
plt.show()

# make a prediction on the image using the trained model
prediction = model.predict(x_test[0].reshape(1, -1))
print('Model prediction:', np.argmax(prediction))
```

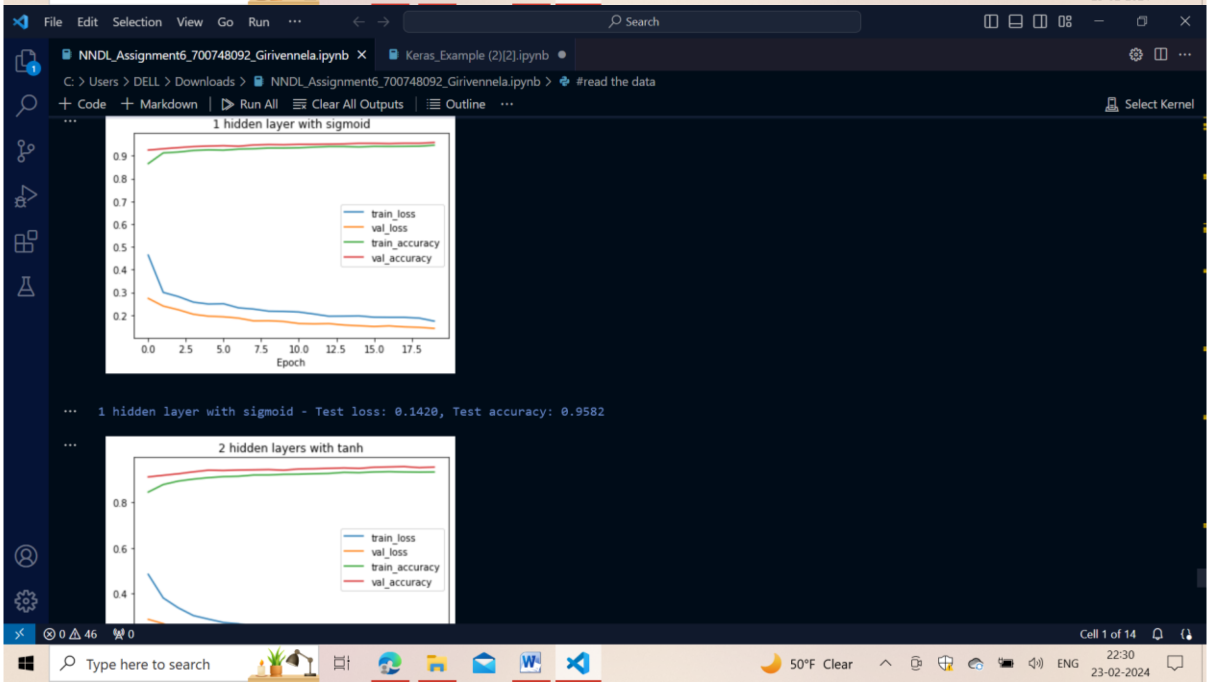
[85]

```
Epoch 1/20
469/469 [==============================] - 12s 22ms/step - loss: 0.2488 - accuracy: 0.9253 - val_loss: 0.1118 - val_accuracy: 0.9652
Epoch 2/20
469/469 [==============================] - 11s 24ms/step - loss: 0.1016 - accuracy: 0.9684 - val_loss: 0.0742 - val_accuracy: 0.9769
Epoch 3/20
469/469 [==============================] - 15s 31ms/step - loss: 0.0713 - accuracy: 0.9779 - val_loss: 0.0695 - val_accuracy: 0.9784
```

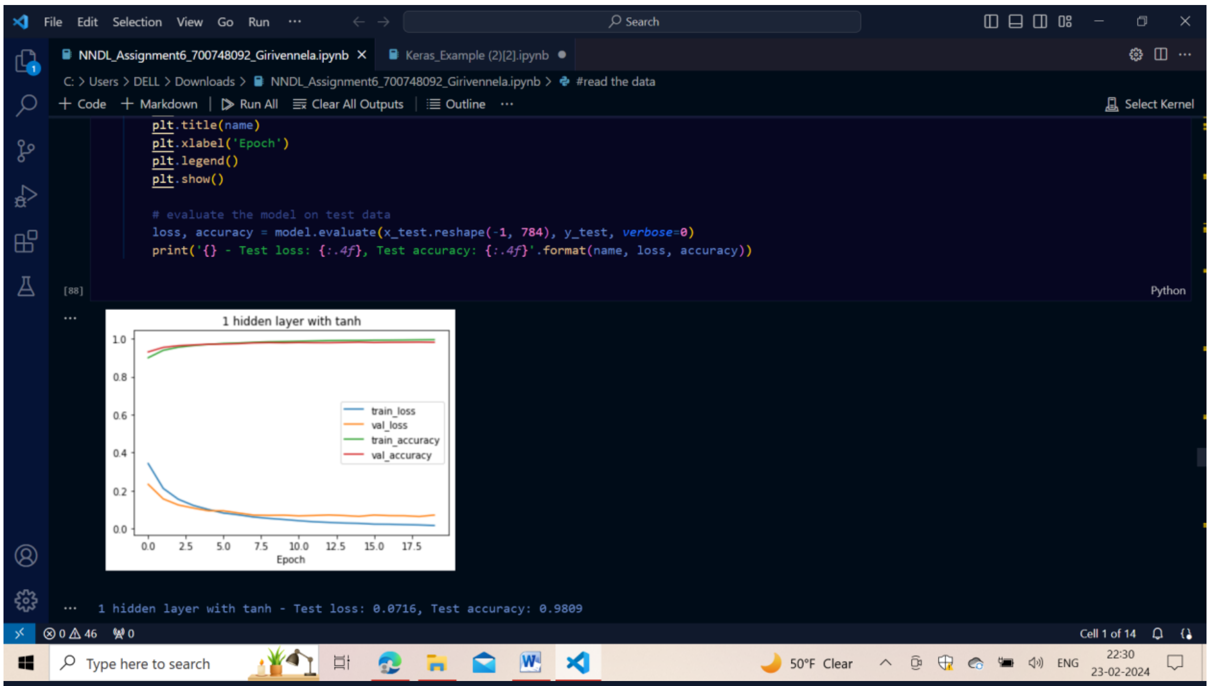

File Edit Selection View Go Run
Search
NNDL_Assignment6_700748092_Girivennela.ipynb
Keras_Example (2)[2].ipynb
C: > Users > DELL > Downloads > NNDL_Assignment6_700748092_Girivennela.ipynb > #read the data
+ Code + Markdown Run All Clear All Outputs Outline
Select Kernel
Cell 1 of 14
Type here to search
50°F Clear
ENG
22:29
23-02-2024

```
469/469 [==============================] - 15s 31ms/step - loss: 0.0713 - accuracy: 0.9779 - val_loss: 0.0695 - val_accuracy: 0.9784
Epoch 4/20
469/469 [==============================] - 14s 30ms/step - loss: 0.0561 - accuracy: 0.9819 - val_loss: 0.0702 - val_accuracy: 0.9779
Epoch 5/20
469/469 [==============================] - 11s 24ms/step - loss: 0.0455 - accuracy: 0.9855 - val_loss: 0.0615 - val_accuracy: 0.9822
Epoch 6/20
469/469 [==============================] - 11s 24ms/step - loss: 0.0381 - accuracy: 0.9873 - val_loss: 0.0648 - val_accuracy: 0.9818
Epoch 7/20
469/469 [==============================] - 11s 24ms/step - loss: 0.0337 - accuracy: 0.9891 - val_loss: 0.0732 - val_accuracy: 0.9810
Epoch 8/20
469/469 [==============================] - 12s 25ms/step - loss: 0.0296 - accuracy: 0.9905 - val_loss: 0.0675 - val_accuracy: 0.9811
Epoch 9/20
469/469 [==============================] - 12s 25ms/step - loss: 0.0279 - accuracy: 0.9905 - val_loss: 0.0756 - val_accuracy: 0.9799
Epoch 10/20
469/469 [==============================] - 12s 26ms/step - loss: 0.0248 - accuracy: 0.9915 - val_loss: 0.0804 - val_accuracy: 0.9806
Epoch 11/20
469/469 [==============================] - 11s 23ms/step - loss: 0.0238 - accuracy: 0.9918 - val_loss: 0.0753 - val_accuracy: 0.9807
Epoch 12/20
469/469 [==============================] - 12s 25ms/step - loss: 0.0237 - accuracy: 0.9923 - val_loss: 0.0743 - val_accuracy: 0.9814
Epoch 13/20
...
Epoch 19/20
469/469 [==============================] - 11s 24ms/step - loss: 0.0145 - accuracy: 0.9951 - val_loss: 0.0873 - val_accuracy: 0.9820
Epoch 20/20
469/469 [==============================] - 11s 24ms/step - loss: 0.0140 - accuracy: 0.9957 - val_loss: 0.0807 - val_accuracy: 0.9841
```

Output is truncated. View as a scrollable element or open in a text editor. Adjust cell output settings...

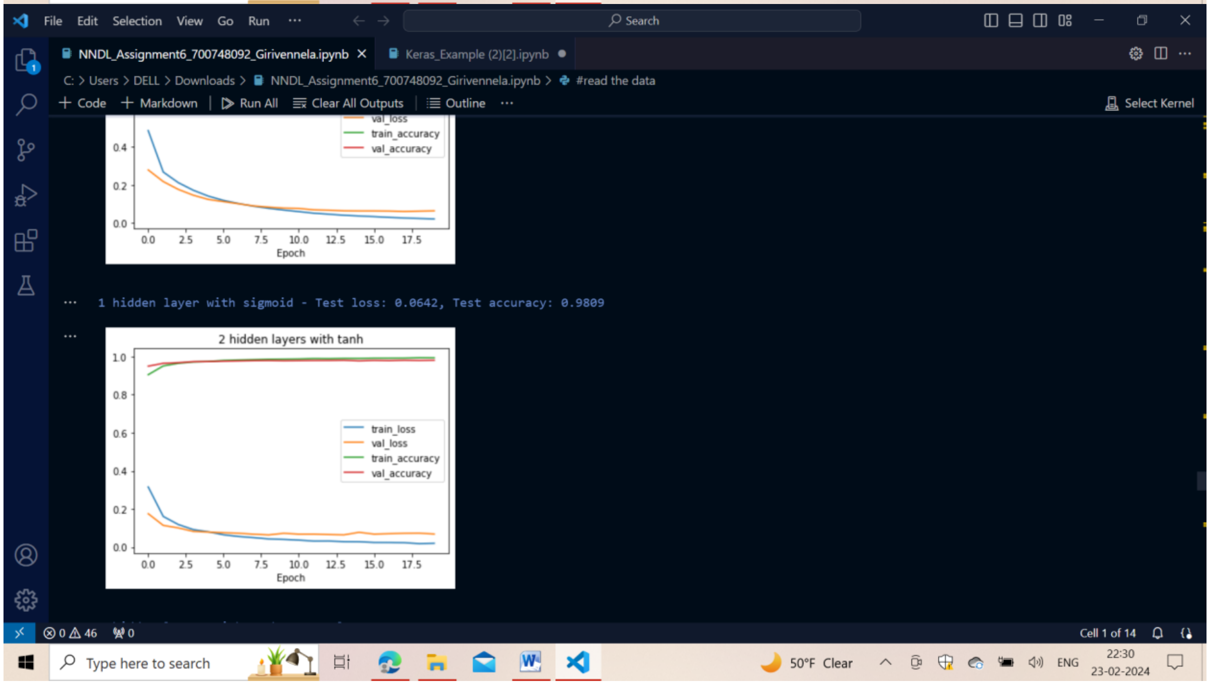

File Edit Selection View Go Run
Search
NNDL_Assignment6_700748092_Girivennela.ipynb
Keras_Example (2)[2].ipynb
C: > Users > DELL > Downloads > NNDL_Assignment6_700748092_Girivennela.ipynb > #read the data
+ Code + Markdown Run All Clear All Outputs Outline
Select Kernel
1 hidden layer with sigmoid
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
Epoch
train_loss
val_loss
train_accuracy
val_accuracy
2 hidden layers with tanh
0.8
0.6
0.4
train_loss
val_loss
train_accuracy
val_accuracy
Cell 1 of 14
Type here to search
50°F Clear
ENG
22:30
23-02-2024

```
1 hidden layer with sigmoid - Test loss: 0.1420, Test accuracy: 0.9582
```

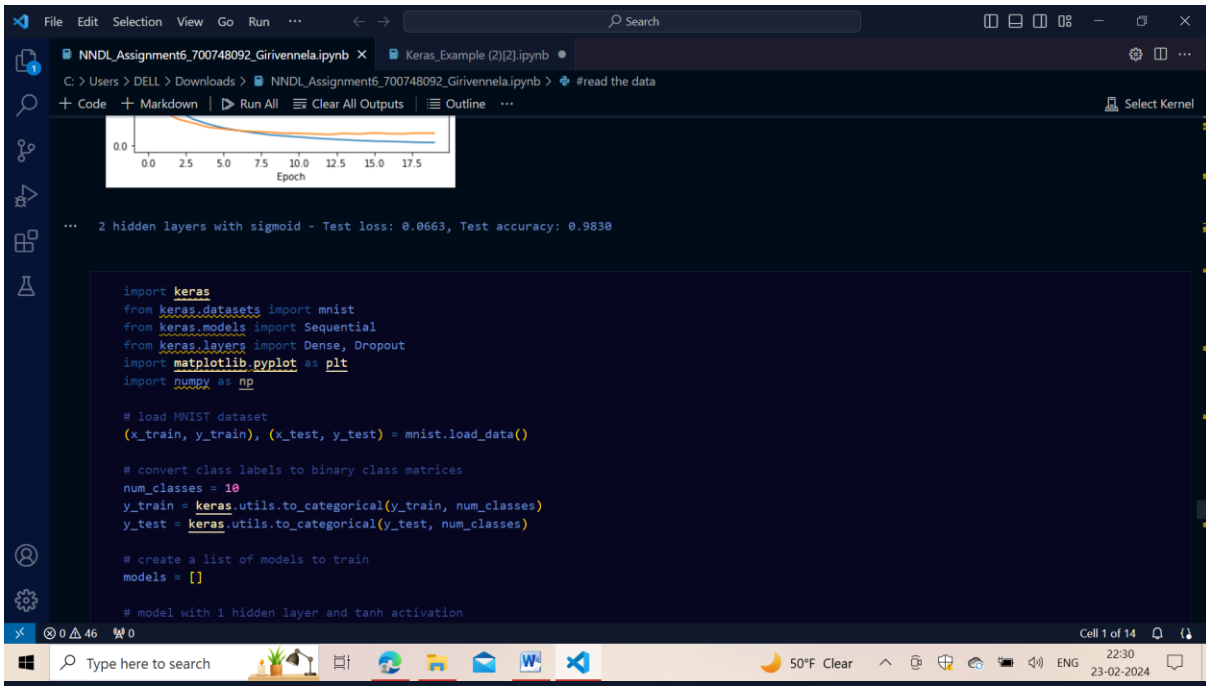

File Edit Selection View Go Run ···
Search
NNDL_Assignment6_700748092_Girivennela.ipynb
Keras_Example (2)[2].ipynb
C: > Users > DELL > Downloads > NNDL_Assignment6_700748092_Girivennela.ipynb > #read the data
+ Code + Markdown | Run All Clear All Outputs | Outline ···
Select Kernel
[88]
Python
1 hidden layer with tanh
1.0
0.8
0.6
0.4
0.2
0.0
0.0 2.5 5.0 7.5 10.0 12.5 15.0 17.5
Epoch
train_loss
val_loss
train_accuracy
val_accuracy
Cell 1 of 14
Type here to search
50°F Clear
ENG
22:30
23-02-2024

```
    plt.title(name)
    plt.xlabel('Epoch')
    plt.legend()
    plt.show()

    # evaluate the model on test data
    loss, accuracy = model.evaluate(x_test.reshape(-1, 784), y_test, verbose=0)
    print('{} - Test loss: {:.4f}, Test accuracy: {:.4f}'.format(name, loss, accuracy))
```

```
1 hidden layer with tanh - Test loss: 0.0716, Test accuracy: 0.9809
```


File Edit Selection View Go Run ···
Search
NNDL_Assignment6_700748092_Girivennela.ipynb
Keras_Example (2)[2].ipynb
C: > Users > DELL > Downloads > NNDL_Assignment6_700748092_Girivennela.ipynb > #read the data
+ Code + Markdown | Run All Clear All Outputs | Outline ···
Select Kernel
val_loss
train_accuracy
val_accuracy
0.4
0.2
0.0
0.0 2.5 5.0 7.5 10.0 12.5 15.0 17.5
Epoch
2 hidden layers with tanh
1.0
0.8
0.6
0.4
0.2
0.0
0.0 2.5 5.0 7.5 10.0 12.5 15.0 17.5
Epoch
train_loss
val_loss
train_accuracy
val_accuracy
Cell 1 of 14
Type here to search
50°F Clear
ENG
22:30
23-02-2024

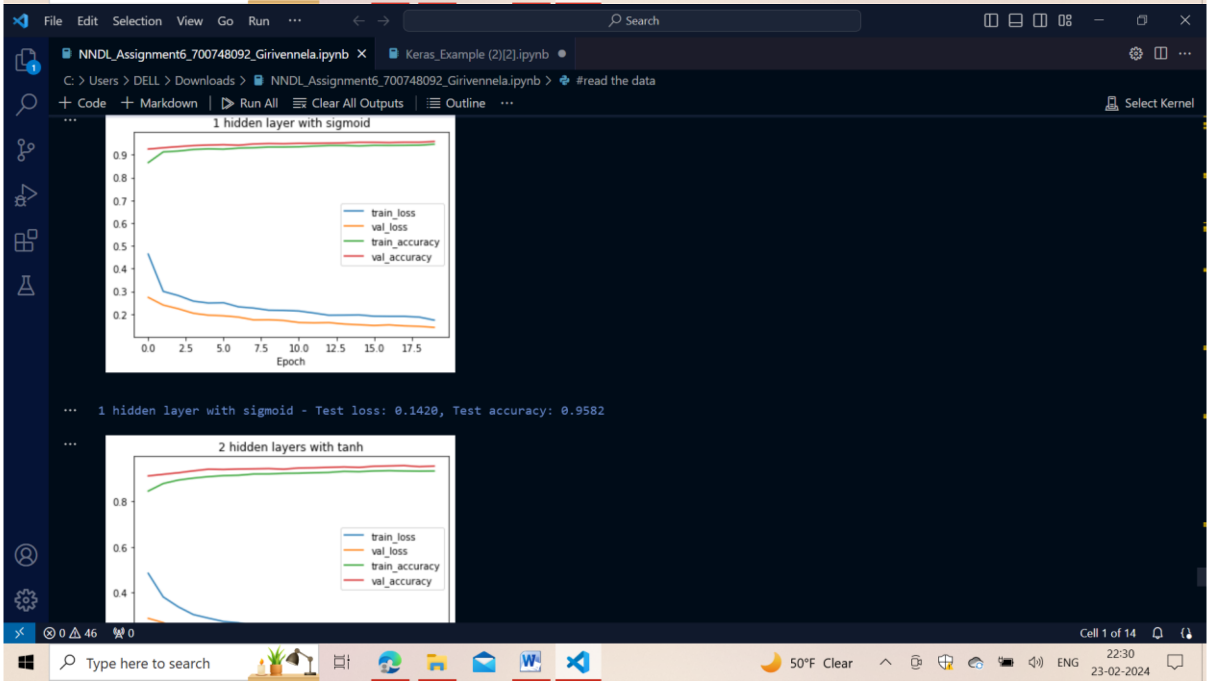
```
1 hidden layer with sigmoid - Test loss: 0.0642, Test accuracy: 0.9809
```



File Edit Selection View Go Run
Search
NNDL_Assignment6_700748092_Girivennela.ipynb
Keras_Example (2)[2].ipynb
C: > Users > DELL > Downloads > NNDL_Assignment6_700748092_Girivennela.ipynb > #read the data
Code
Markdown
Run All
Clear All Outputs
Outline
Select Kernel
0.0
0.0 2.5 5.0 7.5 10.0 12.5 15.0 17.5
Epoch
Cell 1 of 14
Type here to search
50°F Clear
ENG
22:30
23-02-2024

```
2 hidden layers with sigmoid - Test loss: 0.0663, Test accuracy: 0.9830
```

```python
import keras
from keras.datasets import mnist
from keras.models import Sequential
from keras.layers import Dense, Dropout
import matplotlib.pyplot as plt
import numpy as np

# load MNIST dataset
(x_train, y_train), (x_test, y_test) = mnist.load_data()

# convert class labels to binary class matrices
num_classes = 10
y_train = keras.utils.to_categorical(y_train, num_classes)
y_test = keras.utils.to_categorical(y_test, num_classes)

# create a list of models to train
models = []

# model with 1 hidden layer and tanh activation
```



```
1 hidden layer with sigmoid - Test loss: 0.1420, Test accuracy: 0.9582
```