```
from google.colab import drive
drive.mount('/content/drive/')
```

```
Mounted at /content/drive/
```

```
# import libraries
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from keras.models import Sequential
from keras.layers import Dense,Dropout
from keras.utils import to_categorical
```

```
# load the datasets for TRINING
datasets_train = pd.read_csv('/content/drive/My Drive/Colab Notebook

x_train = datasets_train.loc[:,'1x1':]
y_train = datasets_train['label']

x_train = np.reshape(x_train,(-1, 28*28))
x_train /=255
print(x_train.head(10))

```

```
   1x1  1x2  1x3  1x4  1x5  1x6  ...  28x23  28x24  28x25  28x26  28x27  28x28
0  0.0  0.0  0.0  0.0  0.0  0.0  ...    0.0    0.0    0.0    0.0    0.0    0.0
1  0.0  0.0  0.0  0.0  0.0  0.0  ...    0.0    0.0    0.0    0.0    0.0    0.0
2  0.0  0.0  0.0  0.0  0.0  0.0  ...    0.0    0.0    0.0    0.0    0.0    0.0
3  0.0  0.0  0.0  0.0  0.0  0.0  ...    0.0    0.0    0.0    0.0    0.0    0.0
4  0.0  0.0  0.0  0.0  0.0  0.0  ...    0.0    0.0    0.0    0.0    0.0    0.0
5  0.0  0.0  0.0  0.0  0.0  0.0  ...    0.0    0.0    0.0    0.0    0.0    0.0
6  0.0  0.0  0.0  0.0  0.0  0.0  ...    0.0    0.0    0.0    0.0    0.0    0.0
7  0.0  0.0  0.0  0.0  0.0  0.0  ...    0.0    0.0    0.0    0.0    0.0    0.0
8  0.0  0.0  0.0  0.0  0.0  0.0  ...    0.0    0.0    0.0    0.0    0.0    0.0
9  0.0  0.0  0.0  0.0  0.0  0.0  ...    0.0    0.0    0.0    0.0    0.0    0.0

[10 rows x 784 columns]
```

```
# load the datasets fot TESTING
datasets_test = pd.read_csv('/content/drive/My Drive/Colab Notebooks

x_test = datasets_test.loc[:,'1x1':]
y_test = datasets_test['label']

```

```
x_test = np.reshape(x_test,(-1,784))
x_test /=255
print(x_test.head(10))
```

```
   1x1  1x2  1x3  1x4  1x5  1x6  ...  28x23  28x24  28x25  28x26  28x27  28x28
0  0.0  0.0  0.0  0.0  0.0  0.0  ...    0.0    0.0    0.0    0.0    0.0    0.0
1  0.0  0.0  0.0  0.0  0.0  0.0  ...    0.0    0.0    0.0    0.0    0.0    0.0
2  0.0  0.0  0.0  0.0  0.0  0.0  ...    0.0    0.0    0.0    0.0    0.0    0.0
3  0.0  0.0  0.0  0.0  0.0  0.0  ...    0.0    0.0    0.0    0.0    0.0    0.0
4  0.0  0.0  0.0  0.0  0.0  0.0  ...    0.0    0.0    0.0    0.0    0.0    0.0
5  0.0  0.0  0.0  0.0  0.0  0.0  ...    0.0    0.0    0.0    0.0    0.0    0.0
6  0.0  0.0  0.0  0.0  0.0  0.0  ...    0.0    0.0    0.0    0.0    0.0    0.0
7  0.0  0.0  0.0  0.0  0.0  0.0  ...    0.0    0.0    0.0    0.0    0.0    0.0
8  0.0  0.0  0.0  0.0  0.0  0.0  ...    0.0    0.0    0.0    0.0    0.0    0.0
9  0.0  0.0  0.0  0.0  0.0  0.0  ...    0.0    0.0    0.0    0.0    0.0    0.0

[10 rows x 784 columns]
```

```
# defining the keras model
model = Sequential()
# input layer
model.add(Dense(1024,input_shape=(784,),activation='relu'))
model.add(Dense(128, activation='tanh'))
model.add(Dropout(rate=0.05))
model.add(Dense(64, activation='relu'))
model.add(Dense(10,activation='softmax'))

```

```
# compiling the model
model.compile(optimizer='adam',loss='categorical_crossentropy',metri
```

```
# fitting the model
history = model.fit(x=x_train,y=to_categorical(y_train),batch_size=1
```

```
Epoch 1/5
600/600 [==============================] - 8s 14ms/step - loss: 0.2059 - accuracy: 0.939
Epoch 2/5
600/600 [==============================] - 8s 14ms/step - loss: 0.0782 - accuracy: 0.975
Epoch 3/5
600/600 [==============================] - 8s 14ms/step - loss: 0.0511 - accuracy: 0.983
Epoch 4/5
600/600 [==============================] - 8s 14ms/step - loss: 0.0396 - accuracy: 0.987
Epoch 5/5
600/600 [==============================] - 8s 14ms/step - loss: 0.0279 - accuracy: 0.990
```

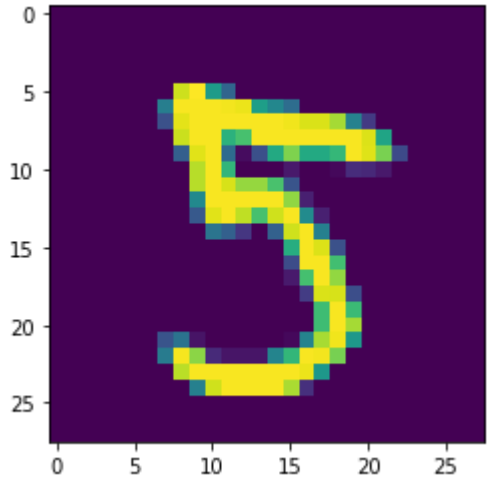
```
# Evaluate the loss and accuracy of the model
loss,accuracy = model.evaluate(x=x_train,y=to_categorical(y_train),b
```

```
print(f'\nloss: {loss} \naccuracy: {accuracy} ')
model.save('/content/drive/My Drive/Colab Notebooks/handwritten_mode
print ('model saved successfully!')
```

```
600/600 [==============================] - 4s 6ms/step - loss: 0.0159 - accuracy: 0.994

loss: 0.015853187069296837
accuracy: 0.9948833584785461
INFO:tensorflow:Assets written to: /content/drive/My Drive/Colab Notebooks/handwritten_m
model saved successfully!
```

```
# Make predictions
v=x_test.loc[2545,:]
a=np.reshape(np.asarray(v),(28,28))
plt.imshow(a)
plt.show()
predictions = model.predict(x=np.asarray(v).reshape((-1,784)))
print(np.round(predictions))
```




```
[[0. 0. 0. 0. 0. 1. 0. 0. 0. 0.]]
```

```
# plot accuracy
plt.xlim(0,1000)
plt.plot(history.history['accuracy'],'r-')
plt.title('accuracy with epochs')
plt.xlabel('epochs')
plt.ylabel('accuracy')
plt.legend(['train','test'],loc='upper right')
plt.show()

# plot loss
plt.plot(history.history['loss'],'r-')
plt.title('loss with epochs')
```

```
plt.title('loss with epochs')
plt.xlabel('epochs')
plt.ylabel('loss')
plt.legend(['train','test'],loc='upper right')
plt.show()
```