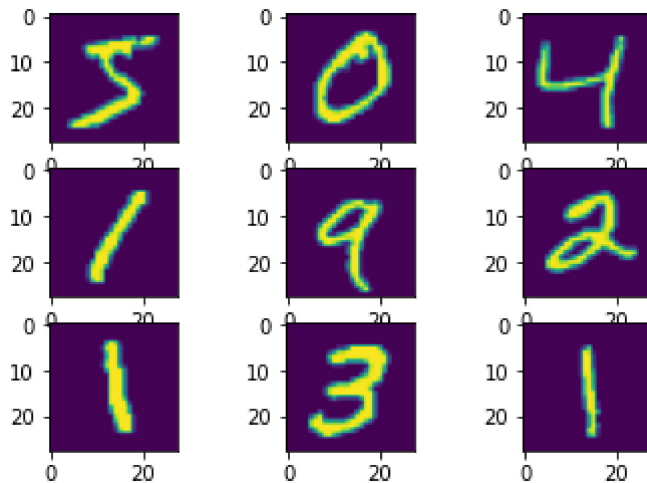


#Trần Nam Phương 20146470

```
from keras.datasets import mnist
import matplotlib.pyplot as plt
import numpy as np
from keras.utils.np_utils import to_categorical
(X_train,y_train),(X_test,y_test)=mnist.load_data()
for i in range(9):
    plt.subplot(330+i+1)
    plt.imshow(X_train[i])
plt.show()
```



X=X\_test

```
X_train=X_train.astype('float32')
X_test=X_test.astype('float32')
X_train=X_train/255
X_test=X_test/255
from tensorflow.keras.utils import to_categorical
```

```
y_train=to_categorical(y_train,10)
y_test=to_categorical(y_test,10)
y_train.shape
```

(60000, 10)

```
from keras.layers import Dense, Flatten
from keras.layers import Conv2D, MaxPooling2D
import numpy as np
from keras.models import Sequential
from keras.layers import Dense,Activation,Dropout
```

```

model = Sequential()

model.add(Conv2D(32, (3, 3), activation='relu',kernel_initializer='he_uniform',padding='Same')
model.add(Conv2D(32, (3, 3), activation='relu',kernel_initializer='he_uniform',padding='Same')
model.add(MaxPooling2D((2,2)))

model.add(Conv2D(64, (3, 3), activation='relu',kernel_initializer='he_uniform',padding='Same')
model.add(Conv2D(64, (3, 3), activation='relu',kernel_initializer='he_uniform',padding='Same')
model.add(MaxPooling2D((2,2)))

model.add(Conv2D(128, (3, 3), activation='relu',kernel_initializer='he_uniform',padding='Same')
model.add(Conv2D(128, (3, 3), activation='relu',kernel_initializer='he_uniform',padding='Same')
model.add(Flatten())
model.add(Dense(128, activation='relu',kernel_initializer='he_uniform'))
model.add(Dense(10,activation='softmax'))

from tensorflow.keras.optimizers import SGD
opt= SGD(lr=0.01, momentum=0.9)
model.compile(optimizer=opt,loss='categorical_crossentropy', metrics=['accuracy'])
history = model.fit(X_train, y_train, batch_size=64, epochs=4,validation_data=(X_test,y_test))

```

```

Epoch 1/4
/usr/local/lib/python3.7/dist-packages/keras/optimizer_v2/gradient_descent.py:102: UserWarning
  super(SGD, self).__init__(name, **kwargs)
938/938 [=====] - 358s 381ms/step - loss: 2.2797 - accuracy: 0
Epoch 2/4
938/938 [=====] - 360s 384ms/step - loss: 2.3015 - accuracy: 0
Epoch 3/4
938/938 [=====] - 360s 383ms/step - loss: 2.2892 - accuracy: 0
Epoch 4/4
938/938 [=====] - 357s 381ms/step - loss: 2.3022 - accuracy: 0
938/938 [=====] - 357s 381ms/step - loss: 2.3022 - accuracy: 0

```

```

score = model.evaluate(X_test, y_test, verbose=1)
print('Test loss =', score[0])
print('Test accuracy =', score[1])

```

```

313/313 [=====] - 15s 47ms/step - loss: 2.3015 - accuracy: 0.11
Test loss = 2.3015193939208984
Test accuracy = 0.11349999904632568

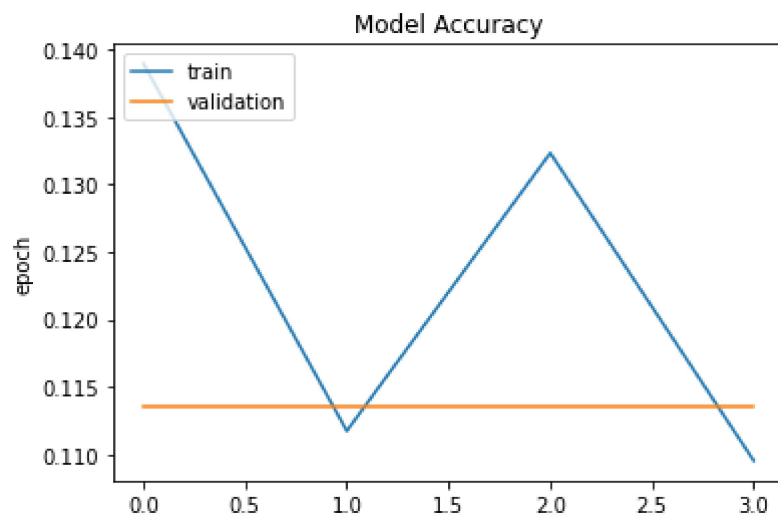
```

```

plt.plot(history.history['accuracy'])
plt.plot(history.history['val_accuracy'])
plt.title('Model Accuracy')
plt.ylabel('accuracy')
plt.ylabel('epoch')
plt.legend(['train', 'validation'],loc='upper left')

```

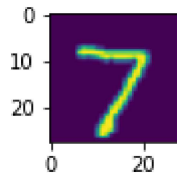
&lt;matplotlib.legend.Legend at 0x7f1cf8355d90&gt;



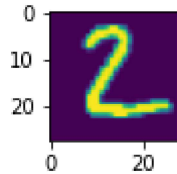
```
y_pred = model.predict(X_test)
for i in range(9):
    plt.subplot(330+i+1)
    plt.imshow(X[i])
    print(y_pred[i])
    plt.show()
```



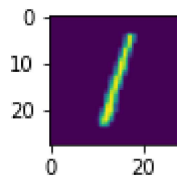
```
[0.10422329 0.10587932 0.09988899 0.10342236 0.09933028 0.08634588  
0.09769671 0.10431349 0.0955336 0.10336607]
```



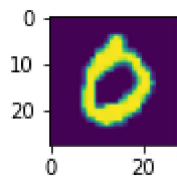
```
[0.10422329 0.10587932 0.0998889 0.10342236 0.09933028 0.08634588  
0.09769671 0.10431349 0.0955336 0.10336607]
```



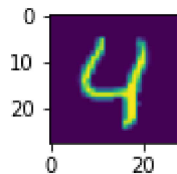
```
[0.10422329 0.10587932 0.09988899 0.10342236 0.09933028 0.08634588  
0.09769671 0.10431349 0.0955336 0.10336608]
```



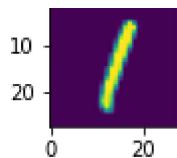
```
[0.1042233 0.10587932 0.09988899 0.10342236 0.09933028 0.08634589  
0.09769671 0.10431349 0.0955336 0.10336608]
```



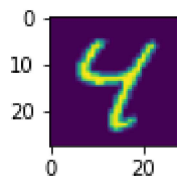
```
[0.10422329 0.10587932 0.09988899 0.10342236 0.09933028 0.08634588  
0.09769671 0.10431349 0.0955336 0.10336608]
```



```
[0.10422329 0.10587932 0.09988899 0.10342236 0.09933028 0.08634588  
0.09769671 0.10431349 0.0955336 0.10336608]
```



```
[0.10422329 0.10587931 0.09988899 0.10342236 0.09933027 0.08634588  
0.09769671 0.10431349 0.09553359 0.10336606]
```



```
[0.10422329 0.10587932 0.09988899 0.10342236 0.09933028 0.08634588  
0.09769671 0.10431349 0.09553359 0.10336608]
```