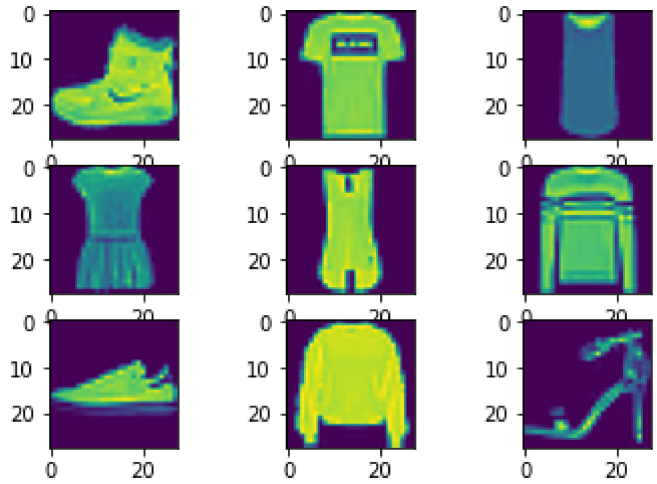
```
from keras.datasets import fashion_mnist
(X_train,y_train),(X_test,y_test)=fashion_mnist.load_data()
import matplotlib.pyplot as plt
for i in range(9):
 plt.subplot(330+i+1)
 plt.imshow(X_train[i])
plt.show()
```



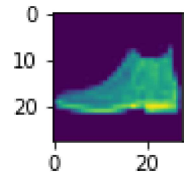
```
X = X_test
X_train =X_train.reshape(60000,784)
X_test = X_test.reshape(10000,784)
X_train = X_train.astype('float32')
X_test = X_test.astype('float32')
X_train/=255
X_test/=255
from tensorflow.keras.utils import to_categorical
y_train=to_categorical(y_train,10)
y_test=to_categorical(y_test,10)
from keras.models import Sequential
from keras.layers import Dense,Activation,Dropout
model = Sequential()
model.add(Dense(512,activation='relu',input_shape=(784,)))
model.add(Dropout(0.2))
model.add(Dense(512,activation='relu'))
model.add(Dropout(0.1))
model.add(Dense(10,activation="softmax"))
model.summary()
from tensorflow.keras.optimizers import RMSprop
model.compile(loss='categorical_crossentropy',optimizer=RMSprop(),metrics=['accuracy'])
history = model.fit(X_train,y_train,batch_size=128,epochs=50,verbose=1,validation_data=(X_
```

```
Epoch 23/50
469/469 [==============================] - 3s 6ms/step - loss: 0.2493 - accuracy: 0.
Epoch 24/50
469/469 [==============================] - 3s 6ms/step - loss: 0.2438 - accuracy: 0.
Epoch 25/50
```

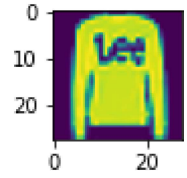
```
469/469 [==============================] - 3s 6ms/step - loss: 0.2476 - accuracy: 0.
Epoch 26/50
469/469 [==============================] - 3s 6ms/step - loss: 0.2440 - accuracy: 0.
Epoch 27/50
469/469 [==============================] - 3s 6ms/step - loss: 0.2385 - accuracy: 0.
Epoch 28/50
469/469 [==============================] - 3s 6ms/step - loss: 0.2443 - accuracy: 0.
Epoch 29/50
469/469 [==============================] - 3s 6ms/step - loss: 0.2363 - accuracy: 0.
Epoch 30/50
469/469 [==============================] - 3s 6ms/step - loss: 0.2341 - accuracy: 0.
Epoch 31/50
469/469 [==============================] - 3s 6ms/step - loss: 0.2331 - accuracy: 0.
Epoch 32/50
469/469 [==============================] - 3s 6ms/step - loss: 0.2295 - accuracy: 0.
Epoch 33/50
469/469 [==============================] - 3s 6ms/step - loss: 0.2310 - accuracy: 0.
Epoch 34/50
469/469 [==============================] - 3s 6ms/step - loss: 0.2257 - accuracy: 0.
Epoch 35/50
469/469 [==============================] - 3s 6ms/step - loss: 0.2254 - accuracy: 0.
Epoch 36/50
469/469 [==============================] - 3s 6ms/step - loss: 0.2259 - accuracy: 0.
Epoch 37/50
469/469 [==============================] - 3s 6ms/step - loss: 0.2227 - accuracy: 0.
Epoch 38/50
469/469 [==============================] - 3s 6ms/step - loss: 0.2250 - accuracy: 0.
Epoch 39/50
469/469 [==============================] - 3s 6ms/step - loss: 0.2184 - accuracy: 0.
Epoch 40/50
469/469 [==============================] - 3s 6ms/step - loss: 0.2231 - accuracy: 0.
Epoch 41/50
469/469 [==============================] - 3s 6ms/step - loss: 0.2212 - accuracy: 0.
Epoch 42/50
469/469 [==============================] - 3s 6ms/step - loss: 0.2159 - accuracy: 0.
Epoch 43/50
469/469 [==============================] - 3s 6ms/step - loss: 0.2124 - accuracy: 0.
Epoch 44/50
469/469 [==============================] - 3s 6ms/step - loss: 0.2179 - accuracy: 0.
Epoch 45/50
469/469 [==============================] - 3s 6ms/step - loss: 0.2133 - accuracy: 0.
Epoch 46/50
469/469 [==============================] - 3s 6ms/step - loss: 0.2103 - accuracy: 0.
Epoch 47/50
469/469 [==============================] - 3s 6ms/step - loss: 0.2147 - accuracy: 0.
Epoch 48/50
469/469 [==============================] - 3s 6ms/step - loss: 0.2090 - accuracy: 0.
Epoch 49/50
469/469 [==============================] - 3s 6ms/step - loss: 0.2117 - accuracy: 0.
Epoch 50/50
469/469 [==============================] - 3s 6ms/step - loss: 0.2061 - accuracy: 0.
```

```
1 import numpy as np
```

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

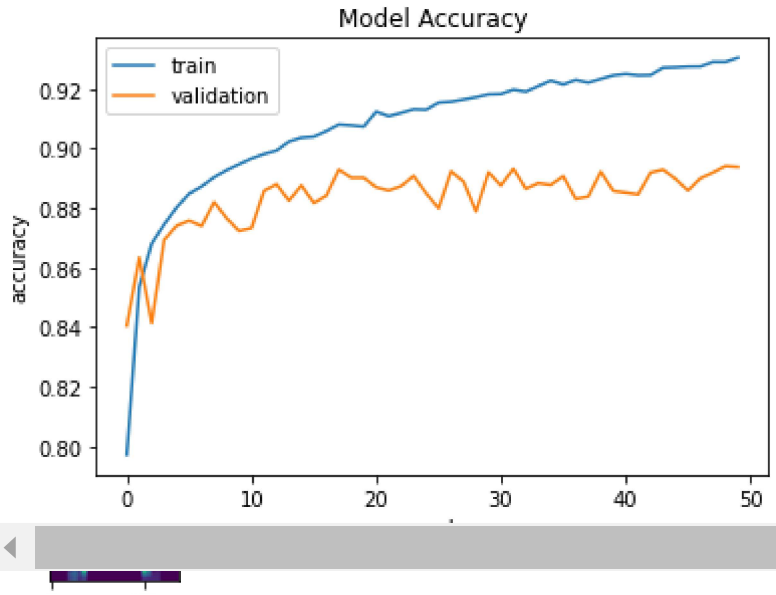


```
[0. 0. 0. 0. 0. 0. 0. 0. 0. 1.]
```

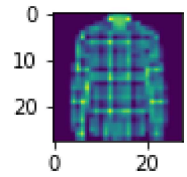


```
1 score=model.evaluate(X_test,y_test,verbose=1)
2 print('Test loss=',score[0])
3 print('Test accuracy=',score[1])
4 plt.plot(history.history['accuracy'])
5 plt.plot(history.history['val_accuracy'])
6 plt.title('Model Accuracy')
7 plt.ylabel('accuracy')
8 plt.xlabel('epoch')
9 plt.legend(['train','validation'],loc='upper left')
```

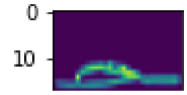
```
313/313 [==============================] - 1s 3ms/step - loss: 0.6549 - accuracy: 0.8938
Test loss= 0.6549438834190369
Test accuracy= 0.8938000202178955
<matplotlib.legend.Legend at 0x7efb7238b890>
```




```
1
```



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




20