

```

import tensorflow as tf

import numpy as np
import matplotlib.pyplot as plt

fashion_mnist = tf.keras.datasets.fashion_mnist

(train_images, train_labels), (test_images, test_labels) = fashion_mnist.load_data()

class_names = ['T-shirt/top', 'Trouser', 'Pullover', 'Dress', 'Coat',
               'Sandal', 'Shirt', 'Sneaker', 'Bag', 'Ankle boot']

train_images.shape
len(train_labels)
test_images.shape
len(test_labels)

plt.figure()
plt.imshow(train_images[0])
plt.colorbar()
plt.grid(False)
plt.show()

train_images = train_images / 255.0

test_images = test_images / 255.0

plt.figure(figsize=(10,10))
for i in range(25):
    plt.subplot(5,5,i+1)
    plt.xticks([])
    plt.yticks([])
    plt.grid(False)
    plt.imshow(train_images[i], cmap=plt.cm.binary)
    plt.xlabel(class_names[train_labels[i]])
plt.show()

model = tf.keras.Sequential([
    tf.keras.layers.Flatten(input_shape=(28, 28)),
    tf.keras.layers.Dense(128, activation='relu'),
    tf.keras.layers.Dense(10)
])

model.compile(optimizer='adam',
              loss=tf.keras.losses.SparseCategoricalCrossentropy(from_logits=True),
              metrics=['accuracy'])

model.fit(train_images, train_labels, epochs=10)

test_loss, test_acc = model.evaluate(test_images, test_labels, verbose=2)

print('\nTest accuracy:', test_acc)

probability_model = tf.keras.Sequential([model,
                                         tf.keras.layers.Softmax()])

predictions = probability_model.predict(test_images)

predictions[0]
np.argmax(predictions[0])
test_labels[0]

def plot_image(i, predictions_array, true_label, img):
    true_label, img = true_label[i], img[i]
    plt.grid(False)
    plt.xticks([])
    plt.yticks([])

    plt.imshow(img, cmap=plt.cm.binary)

    predicted_label = np.argmax(predictions_array)
    if predicted_label == true_label:
        color = 'blue'
    else:
        color = 'red'

```

```

plt.xlabel("{} {:.20f}% ({}).format(class_names[predicted_label],
                                   100*np.max(predictions_array),
                                   class_names[true_label]),
          color=color)

def plot_value_array(i, predictions_array, true_label):
    true_label = true_label[i]
    plt.grid(False)
    plt.xticks(range(10))
    plt.yticks([])
    thisplot = plt.bar(range(10), predictions_array, color="#777777")
    plt.ylim([0, 1])
    predicted_label = np.argmax(predictions_array)

    thisplot[predicted_label].set_color('red')
    thisplot[true_label].set_color('blue')

i = 0
plt.figure(figsize=(6,3))
plt.subplot(1,2,1)
plot_image(i, predictions[i], test_labels, test_images)
plt.subplot(1,2,2)
plot_value_array(i, predictions[i], test_labels)
plt.show()

i = 12
plt.figure(figsize=(6,3))
plt.subplot(1,2,1)
plot_image(i, predictions[i], test_labels, test_images)
plt.subplot(1,2,2)
plot_value_array(i, predictions[i], test_labels)
plt.show()

num_rows = 5
num_cols = 3
num_images = num_rows*num_cols
plt.figure(figsize=(2*2*num_cols, 2*num_rows))
for i in range(num_images):
    plt.subplot(num_rows, 2*num_cols, 2*i+1)
    plot_image(i, predictions[i], test_labels, test_images)
    plt.subplot(num_rows, 2*num_cols, 2*i+2)
    plot_value_array(i, predictions[i], test_labels)
plt.tight_layout()
plt.show()

img = test_images[1]

print(img.shape)

img = (np.expand_dims(img,0))

print(img.shape)

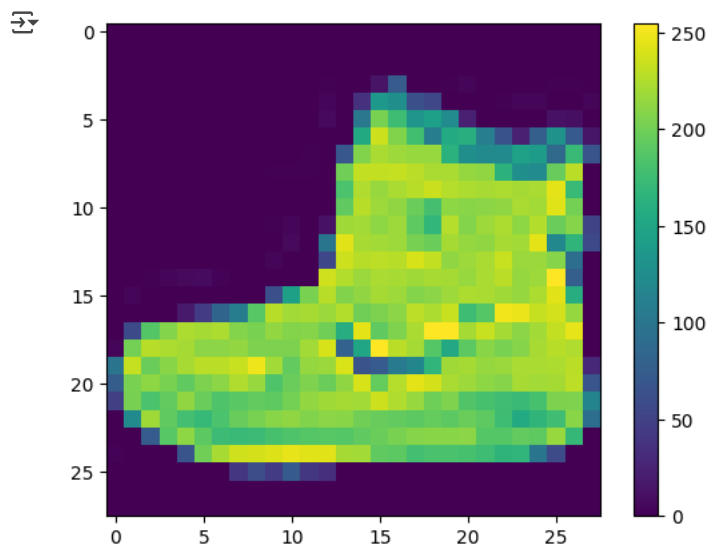
predictions_single = probability_model.predict(img)

print(predictions_single)

plot_value_array(1, predictions_single[0], test_labels)
_ = plt.xticks(range(10), class_names, rotation=45)
plt.show()

np.argmax(predictions_single[0])

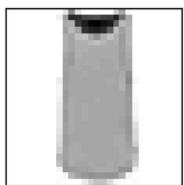
```



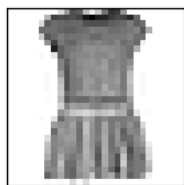
Ankle boot



T-shirt/top



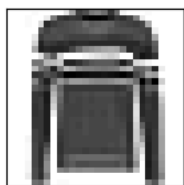
T-shirt/top



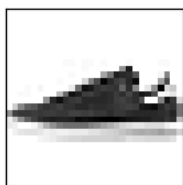
Dress



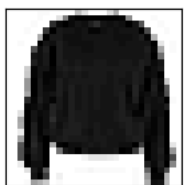
T-shirt/top



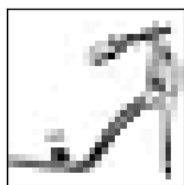
Pullover



Sneaker



Pullover



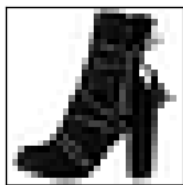
Sandal



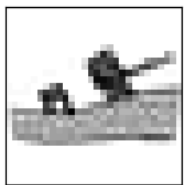
Sandal



T-shirt/top



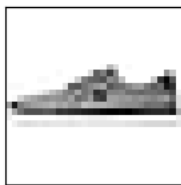
Ankle boot



Sandal



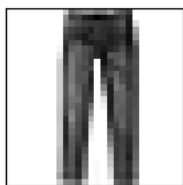
Sandal



Sneaker



Ankle boot



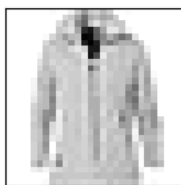
Trouser



T-shirt/top



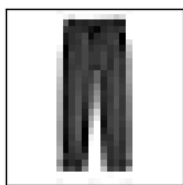
Shirt



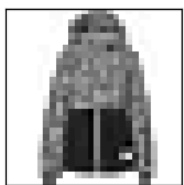
Coat



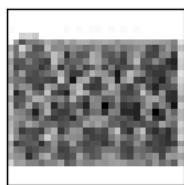
Dress



Trouser



Coat



Bag



Coat

Epoch 1/10

1875/1875 ————— 7s 3ms/step - accuracy: 0.7788 - loss: 0.6268

Epoch 2/10

1875/1875 ————— 8s 4ms/step - accuracy: 0.8630 - loss: 0.3806

Epoch 3/10

1875/1875 ————— 7s 4ms/step - accuracy: 0.8741 - loss: 0.3372

Epoch 4/10

1875/1875 ————— 9s 5ms/step - accuracy: 0.8858 - loss: 0.3083

Epoch 5/10

1875/1875 ————— 10s 5ms/step - accuracy: 0.8927 - loss: 0.2941

Epoch 6/10

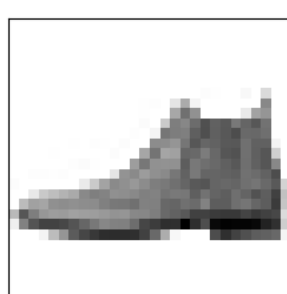
1875/1875 ————— 8s 4ms/step - accuracy: 0.8971 - loss: 0.2753

Epoch 7/10

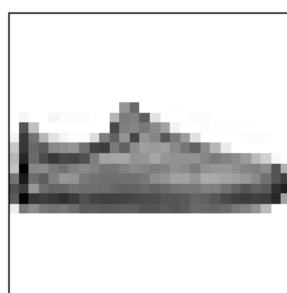
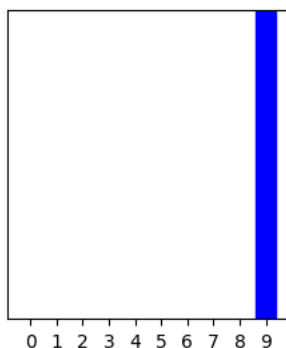
1875/1875 ————— 12s 5ms/step - accuracy: 0.9030 - loss: 0.2581

Epoch 8/10
 1875/1875 ————— 10s 5ms/step - accuracy: 0.9056 - loss: 0.2526
 Epoch 9/10
 1875/1875 ————— 8s 4ms/step - accuracy: 0.9097 - loss: 0.2434
 Epoch 10/10
 1875/1875 ————— 11s 4ms/step - accuracy: 0.9126 - loss: 0.2377
 313/313 - 1s - 2ms/step - accuracy: 0.8805 - loss: 0.3403

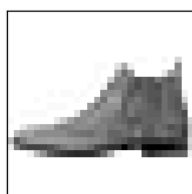
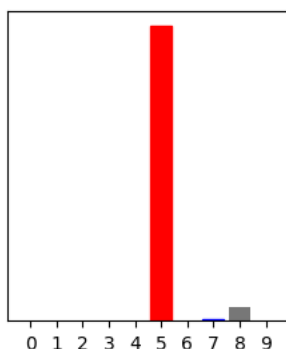
Test accuracy: 0.8805000185966492
 313/313 ————— 1s 2ms/step



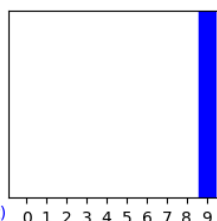
Ankle boot 100% (Ankle boot)



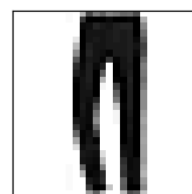
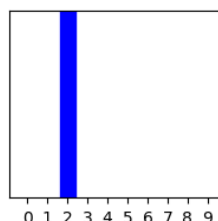
Sandal 95% (Sneaker)



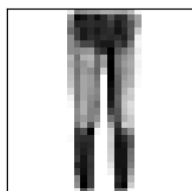
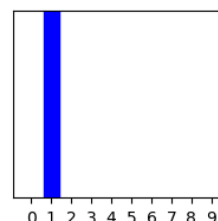
Ankle boot 100% (Ankle boot)



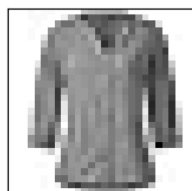
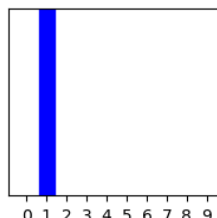
Pullover 100% (Pullover)



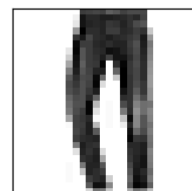
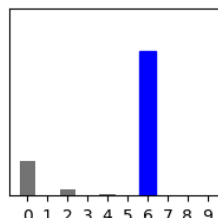
Trouser 100% (Trouser)



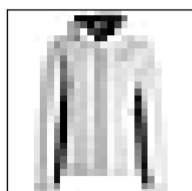
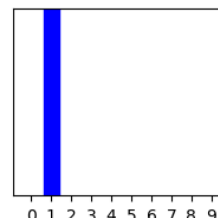
Trouser 100% (Trouser)



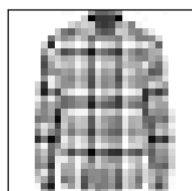
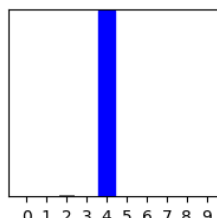
Shirt 77% (Shirt)



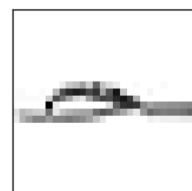
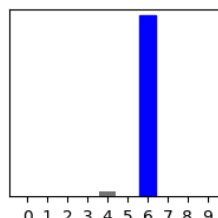
Trouser 100% (Trouser)



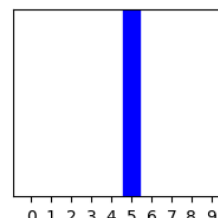
Coat 99% (Coat)



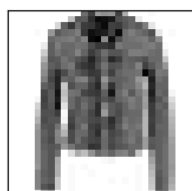
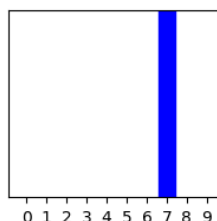
Shirt 97% (Shirt)



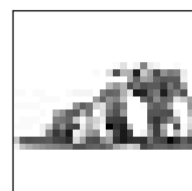
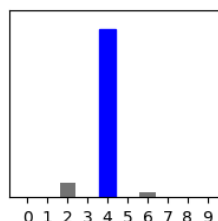
Sandal 100% (Sandal)



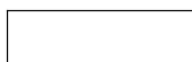
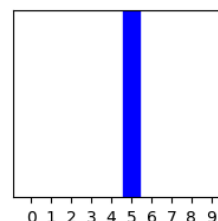
Sneaker 100% (Sneaker)



Coat 90% (Coat)

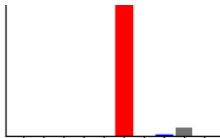


Sandal 100% (Sandal)

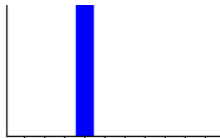




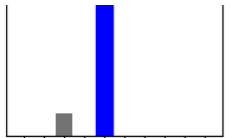
Sandal 95% (Sneaker)



Dress 100% (Dress)



Coat 88% (Coat)



(28, 28)

(1, 28, 28)

1/1 0s 68ms/step

[[3.6099329e-05 1.3937782e-13 9.9924624e-01 1.5083781e-11 6.2801113e-04
1.7261953e-13 8.9689638e-05 4.0066847e-14 4.9377037e-11 8.7570051e-16]]

