


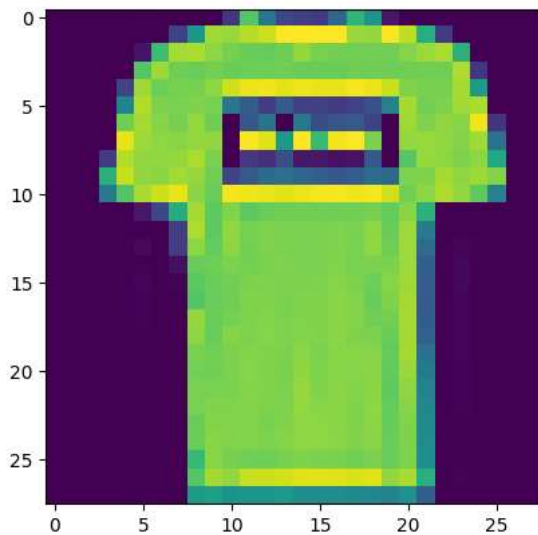
```
import tensorflow as tf
import matplotlib.pyplot as plt
from tensorflow import keras
import numpy as np
```

```
(x_train, y_train), (x_test, y_test) = keras.datasets.fashion_mnist.load_data()
```

```
Downloading data from https://storage.googleapis.com/tensorflow/tf-keras-datasets/train-labels-idx1-ubyte.gz
29515/29515 [=====] - 0s 0us/step
Downloading data from https://storage.googleapis.com/tensorflow/tf-keras-datasets/train-images-idx3-ubyte.gz
26421880/26421880 [=====] - 0s 0us/step
Downloading data from https://storage.googleapis.com/tensorflow/tf-keras-datasets/t10k-labels-idx1-ubyte.gz
5148/5148 [=====] - 0s 0us/step
Downloading data from https://storage.googleapis.com/tensorflow/tf-keras-datasets/t10k-images-idx3-ubyte.gz
4422102/4422102 [=====] - 0s 0us/step
```

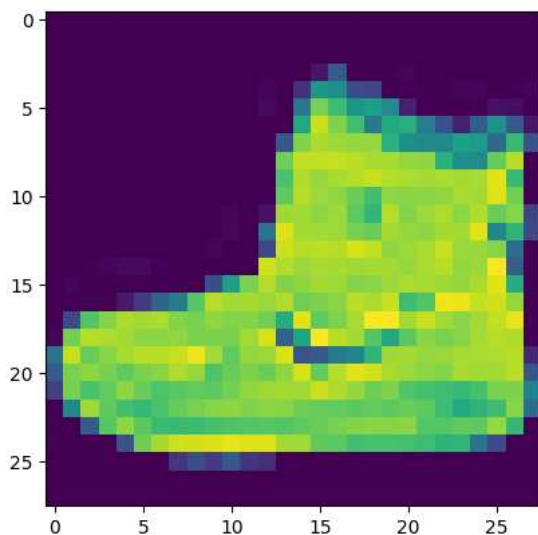
```
plt.imshow(x_train[1])
```

 <matplotlib.image.AxesImage at 0x7f856124ad70>



```
plt.imshow(x_train[0])
```

<matplotlib.image.AxesImage at 0x7f855ba46ec0>



```
x_train = x_train.astype('float32') / 255.0
x_test = x_test.astype('float32') / 255.0
```

```
x_train = x_train.reshape(-1, 28, 28, 1)
x_test = x_test.reshape(-1, 28, 28, 1)

x_train.shape
(60000, 28, 28)
x_test.shape
(10000, 28, 28, 1)
y_train.shape
(60000,)
y_test.shape
(10000,)

model = keras.Sequential([
keras.layers.Conv2D(32, (3,3), activation='relu', input_shape=(28,28,1)),

keras.layers.MaxPooling2D((2,2)),

keras.layers.Dropout(0.25),

keras.layers.Conv2D(64, (3,3), activation='relu'),
keras.layers.MaxPooling2D((2,2)),

keras.layers.Dropout(0.25),
keras.layers.Conv2D(128, (3,3), activation='relu'),

keras.layers.Flatten(),
keras.layers.Dense(128, activation='relu'),

keras.layers.Dropout(0.25),
keras.layers.Dense(10, activation='softmax')])

model.summary()

model.compile(optimizer='adam', loss='sparse_categorical_crossentropy', metrics=['accuracy'])
history = model.fit(x_train, y_train, epochs=10, validation_data=(x_test, y_test))
```

Model: "sequential"

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 26, 26, 32)	320
max_pooling2d (MaxPooling2D)	(None, 13, 13, 32)	0
dropout (Dropout)	(None, 13, 13, 32)	0
conv2d_1 (Conv2D)	(None, 11, 11, 64)	18496
max_pooling2d_1 (MaxPooling2D)	(None, 5, 5, 64)	0
dropout_1 (Dropout)	(None, 5, 5, 64)	0
conv2d_2 (Conv2D)	(None, 3, 3, 128)	73856
flatten (Flatten)	(None, 1152)	0
dense (Dense)	(None, 128)	147584
dropout_2 (Dropout)	(None, 128)	0
dense_1 (Dense)	(None, 10)	1290

Total params: 241,546  
Trainable params: 241,546  
Non-trainable params: 0

Epoch 1/10	1875/1875	[=====]	- 83s 43ms/step	- loss: 0.5470	- accuracy: 0.7968	- val_loss: 0.3708	- val_accuracy: 0.8629
Epoch 2/10	1875/1875	[=====]	- 77s 41ms/step	- loss: 0.3600	- accuracy: 0.8680	- val_loss: 0.3205	- val_accuracy: 0.8810
Epoch 3/10	1875/1875	[=====]	- 79s 42ms/step	- loss: 0.3182	- accuracy: 0.8830	- val_loss: 0.2943	- val_accuracy: 0.8888
Epoch 4/10	1875/1875	[=====]	- 82s 44ms/step	- loss: 0.2916	- accuracy: 0.8913	- val_loss: 0.2761	- val_accuracy: 0.8977

```
Epoch 5/10
1875/1875 [=====] - 79s 42ms/step - loss: 0.2735 - accuracy: 0.8978 - val_loss: 0.2680 - val_accuracy: 0.9013
Epoch 6/10
1875/1875 [=====] - 79s 42ms/step - loss: 0.2603 - accuracy: 0.9035 - val_loss: 0.3003 - val_accuracy: 0.8947
Epoch 7/10
1875/1875 [=====] - 79s 42ms/step - loss: 0.2524 - accuracy: 0.9073 - val_loss: 0.2590 - val_accuracy: 0.9053
Epoch 8/10
1875/1875 [=====] - 79s 42ms/step - loss: 0.2430 - accuracy: 0.9088 - val_loss: 0.2634 - val_accuracy: 0.9048
Epoch 9/10
1875/1875 [=====] - 78s 42ms/step - loss: 0.2337 - accuracy: 0.9123 - val_loss: 0.2554 - val_accuracy: 0.9055
Epoch 10/10
1875/1875 [=====] - 78s 42ms/step - loss: 0.2283 - accuracy: 0.9136 - val_loss: 0.2534 - val_accuracy: 0.9079

test_loss, test_acc = model.evaluate(x_test, y_test)
print('Test accuracy:', test_acc)

313/313 [=====] - 4s 14ms/step - loss: 0.2534 - accuracy: 0.9079
Test accuracy: 0.9078999757766724
```

