

```

1 C:\Users\Adrian\Desktop\ML\KNO_repo\.venv\Scripts\
  python.exe C:\Users\Adrian\Desktop\NAI_toolsAI\
  NeuralNetworkClassification\CNN_exercises.py
2 2024-12-19 23:03:27.167947: I tensorflow/core/util/
  port.cc:153] oneDNN custom operations are on. You may
  see slightly different numerical results due to
  floating-point round-off errors from different
  computation orders. To turn them off, set the
  environment variable `TF_ENABLE_ONEDNN_OPTS=0`.
3 2024-12-19 23:03:28.066788: I tensorflow/core/util/
  port.cc:153] oneDNN custom operations are on. You may
  see slightly different numerical results due to
  floating-point round-off errors from different
  computation orders. To turn them off, set the
  environment variable `TF_ENABLE_ONEDNN_OPTS=0`.
4 C:\Users\Adrian\Desktop\ML\KNO_repo\.venv\Lib\site-
  packages\keras\src\layers\core\dense.py:87:
  UserWarning: Do not pass an `input_shape`/`input_dim`
  ` argument to a layer. When using Sequential models,
  prefer using an `Input(shape)` object as the first
  layer in the model instead.
5   super().__init__(activity_regularizer=
  activity_regularizer, **kwargs)
6 2024-12-19 23:03:30.251707: I tensorflow/core/
  platform/cpu_feature_guard.cc:210] This TensorFlow
  binary is optimized to use available CPU instructions
  in performance-critical operations.
7 To enable the following instructions: AVX2 FMA, in
  other operations, rebuild TensorFlow with the
  appropriate compiler flags.
8 Epoch 1/20
9 2/2 _____ 1s 94ms/step - accuracy: 0.
  3163 - loss: 0.7966 - val_accuracy: 0.3333 - val_loss
  : 0.6866
10 Epoch 2/20
11 2/2 _____ 0s 15ms/step - accuracy: 0.
  5000 - loss: 0.6989 - val_accuracy: 0.6667 - val_loss
  : 0.6480
12 Epoch 3/20
13 2/2 _____ 0s 15ms/step - accuracy: 0.
  6629 - loss: 0.6187 - val_accuracy: 0.8333 - val_loss

```

```
13 : 0.6141
14 Epoch 4/20
15 2/2 _____ 0s 16ms/step - accuracy: 0.
    8674 - loss: 0.5543 - val_accuracy: 1.0000 - val_loss
      : 0.5836
16 Epoch 5/20
17 2/2 _____ 0s 15ms/step - accuracy: 0.
    8466 - loss: 0.5032 - val_accuracy: 1.0000 - val_loss
      : 0.5550
18 Epoch 6/20
19 2/2 _____ 0s 16ms/step - accuracy: 0.
    9489 - loss: 0.4496 - val_accuracy: 1.0000 - val_loss
      : 0.5279
20 Epoch 7/20
21 2/2 _____ 0s 15ms/step - accuracy: 0.
    9697 - loss: 0.3869 - val_accuracy: 1.0000 - val_loss
      : 0.5035
22 Epoch 8/20
23 2/2 _____ 0s 15ms/step - accuracy: 0.
    9489 - loss: 0.3597 - val_accuracy: 1.0000 - val_loss
      : 0.4814
24 Epoch 9/20
25 2/2 _____ 0s 16ms/step - accuracy: 0.
    9489 - loss: 0.3268 - val_accuracy: 1.0000 - val_loss
      : 0.4623
26 Epoch 10/20
27 2/2 _____ 0s 15ms/step - accuracy: 0.
    9697 - loss: 0.2968 - val_accuracy: 0.8333 - val_loss
      : 0.4462
28 Epoch 11/20
29 2/2 _____ 0s 15ms/step - accuracy: 0.
    9697 - loss: 0.2671 - val_accuracy: 0.8333 - val_loss
      : 0.4316
30 Epoch 12/20
31 2/2 _____ 0s 15ms/step - accuracy: 1.
    0000 - loss: 0.2438 - val_accuracy: 0.8333 - val_loss
      : 0.4189
32 Epoch 13/20
33 2/2 _____ 0s 17ms/step - accuracy: 1.
    0000 - loss: 0.2199 - val_accuracy: 0.8333 - val_loss
      : 0.4079
```

```
34 Epoch 14/20
35 2/2 _____ 0s 18ms/step - accuracy: 1.
    0000 - loss: 0.2071 - val_accuracy: 0.8333 - val_loss
      : 0.3980
36 Epoch 15/20
37 2/2 _____ 0s 15ms/step - accuracy: 1.
    0000 - loss: 0.1894 - val_accuracy: 0.8333 - val_loss
      : 0.3894
38 Epoch 16/20
39 2/2 _____ 0s 16ms/step - accuracy: 1.
    0000 - loss: 0.1695 - val_accuracy: 0.8333 - val_loss
      : 0.3833
40 Epoch 17/20
41 2/2 _____ 0s 14ms/step - accuracy: 1.
    0000 - loss: 0.1605 - val_accuracy: 0.8333 - val_loss
      : 0.3799
42 Epoch 18/20
43 2/2 _____ 0s 15ms/step - accuracy: 1.
    0000 - loss: 0.1405 - val_accuracy: 0.8333 - val_loss
      : 0.3774
44 Epoch 19/20
45 2/2 _____ 0s 16ms/step - accuracy: 1.
    0000 - loss: 0.1429 - val_accuracy: 0.8333 - val_loss
      : 0.3759
46 Epoch 20/20
47 2/2 _____ 0s 15ms/step - accuracy: 1.
    0000 - loss: 0.1349 - val_accuracy: 0.8333 - val_loss
      : 0.3761
48 1/1 _____ 0s 26ms/step
49 Neural Network Test Accuracy: 0.79
50
51 Confusion Matrix - Neural Network
52      b      g
53 b   7      6
54 g   0     15
55 Logistic Regression Test Accuracy: 0.75
56
57 Confusion Matrix - Logistic Regression
58      b      g
59 b   6      7
60 g   0     15
```

```

61 Found 2400 images belonging to 3 classes.
62 Found 600 images belonging to 3 classes.
63 C:\Users\Adrian\Desktop\ML\KN0_repo\.venv\Lib\site-
    packages\keras\src\layers\convolutional\base_conv.py
    :107: UserWarning: Do not pass an `input_shape`/`
    input_dim` argument to a layer. When using
    Sequential models, prefer using an `Input(shape)`
    object as the first layer in the model instead.
64     super().__init__(activity_regularizer=
        activity_regularizer, **kwargs)
65 C:\Users\Adrian\Desktop\ML\KN0_repo\.venv\Lib\site-
    packages\keras\src\trainers\data_adapters\
    py_dataset_adapter.py:121: UserWarning: Your `
    PyDataset` class should call `super().__init__(**
    kwargs)` in its constructor. `**kwargs` can include
    `workers`, `use_multiprocessing`, `max_queue_size
    `. Do not pass these arguments to `fit()`, as they
    will be ignored.
66     self._warn_if_super_not_called()
67 Epoch 1/10
68 75/75 _____ 3s 34ms/step - accuracy:
    0.4546 - loss: 1.0530 - val_accuracy: 0.6650 -
    val_loss: 0.7710
69 Epoch 2/10
70 75/75 _____ 2s 32ms/step - accuracy:
    0.6765 - loss: 0.7283 - val_accuracy: 0.6750 -
    val_loss: 0.7278
71 Epoch 3/10
72 75/75 _____ 2s 32ms/step - accuracy:
    0.7447 - loss: 0.5890 - val_accuracy: 0.6883 -
    val_loss: 0.7122
73 Epoch 4/10
74 75/75 _____ 2s 32ms/step - accuracy:
    0.7912 - loss: 0.4972 - val_accuracy: 0.6917 -
    val_loss: 0.7314
75 Epoch 5/10
76 75/75 _____ 2s 31ms/step - accuracy:
    0.8312 - loss: 0.4286 - val_accuracy: 0.6917 -
    val_loss: 0.7322
77 Epoch 6/10
78 75/75 _____ 2s 33ms/step - accuracy:

```

```

78 0.8567 - loss: 0.3545 - val_accuracy: 0.7017 -
   val_loss: 0.7458
79 Epoch 7/10
80 75/75 _____ 2s 32ms/step - accuracy:
   0.8893 - loss: 0.2965 - val_accuracy: 0.6817 -
   val_loss: 0.8196
81 Epoch 8/10
82 75/75 _____ 2s 33ms/step - accuracy:
   0.9311 - loss: 0.2226 - val_accuracy: 0.7117 -
   val_loss: 0.8891
83 Epoch 9/10
84 75/75 _____ 2s 32ms/step - accuracy:
   0.9512 - loss: 0.1519 - val_accuracy: 0.6917 -
   val_loss: 0.9282
85 Epoch 10/10
86 75/75 _____ 2s 32ms/step - accuracy:
   0.9772 - loss: 0.1057 - val_accuracy: 0.6950 -
   val_loss: 1.0627
87 19/19 _____ 0s 19ms/step
88 Test Accuracy: 0.69
89 C:\Users\Adrian\Desktop\NAI_toolsAI\
   NeuralNetworkClassification\CNN_exercises.py:146:
   UserWarning: FigureCanvasAgg is non-interactive, and
   thus cannot be shown
90 plt.show()
91 Downloading data from https://storage.googleapis.com
   /tensorflow/tf-keras-datasets/train-labels-idx1-
  ubyte.gz
92 29515/29515 _____ 0s 0us/step
93 Downloading data from https://storage.googleapis.com
   /tensorflow/tf-keras-datasets/train-images-idx3-
  ubyte.gz
94 26421880/26421880 _____ 1s 0us/step
95 Downloading data from https://storage.googleapis.com
   /tensorflow/tf-keras-datasets/t10k-labels-idx1-ubyte
   .gz
96 5148/5148 _____ 0s 0us/step
97 Downloading data from https://storage.googleapis.com
   /tensorflow/tf-keras-datasets/t10k-images-idx3-ubyte
   .gz
98 4422102/4422102 _____ 0s 0us/step

```

```
99 C:\Users\Adrian\Desktop\ML\KN0_repo\.venv\Lib\site-  
   packages\keras\src\layers\core\dense.py:87:  
   UserWarning: Do not pass an `input_shape`/`input_dim`  
   argument to a layer. When using Sequential models  
   , prefer using an `Input(shape)` object as the first  
   layer in the model instead.  
100   super().__init__(activity_regularizer=  
   activity_regularizer, **kwargs)  
101 Epoch 1/10  
102 150/150 _____ 1s 1ms/step - accuracy  
   : 0.6035 - loss: 1.1881 - val_accuracy: 0.7858 -  
   val_loss: 0.6033  
103 Epoch 2/10  
104 150/150 _____ 0s 811us/step -  
   accuracy: 0.8094 - loss: 0.5560 - val_accuracy: 0.  
   7967 - val_loss: 0.5776  
105 Epoch 3/10  
106 150/150 _____ 0s 810us/step -  
   accuracy: 0.8313 - loss: 0.4882 - val_accuracy: 0.  
   8025 - val_loss: 0.5420  
107 Epoch 4/10  
108 150/150 _____ 0s 812us/step -  
   accuracy: 0.8572 - loss: 0.4252 - val_accuracy: 0.  
   8250 - val_loss: 0.4881  
109 Epoch 5/10  
110 150/150 _____ 0s 811us/step -  
   accuracy: 0.8648 - loss: 0.3799 - val_accuracy: 0.  
   8150 - val_loss: 0.4912  
111 Epoch 6/10  
112 150/150 _____ 0s 803us/step -  
   accuracy: 0.8667 - loss: 0.3735 - val_accuracy: 0.  
   8258 - val_loss: 0.4998  
113 Epoch 7/10  
114 150/150 _____ 0s 807us/step -  
   accuracy: 0.8853 - loss: 0.3235 - val_accuracy: 0.  
   8258 - val_loss: 0.4915  
115 Epoch 8/10  
116 150/150 _____ 0s 819us/step -  
   accuracy: 0.8872 - loss: 0.3058 - val_accuracy: 0.  
   8258 - val_loss: 0.4552  
117 Epoch 9/10
```

```

118 150/150 _____ 0s 812us/step -
    accuracy: 0.8892 - loss: 0.2957 - val_accuracy: 0.
    8292 - val_loss: 0.4899
119 Epoch 10/10
120 150/150 _____ 0s 809us/step -
    accuracy: 0.8989 - loss: 0.2846 - val_accuracy: 0.
    8383 - val_loss: 0.4655
121 188/188 _____ 0s 535us/step
122 Fashion-MNIST Neural Network Test Accuracy: 0.84
123
124 Confusion Matrix - Fashion-MNIST
125           T-shirt/top  Trouser  Pullover  ...
    Sneaker  Bag  Ankle boot
126 T-shirt/top           502           3           3
    ...           0      11           0
127 Trouser              0          556           0
    ...           0       0           0
128 Pullover             8           3          376
    ...           0       2           0
129 Dress                33          26           3
    ...           0       4           0
130 Coat                 0           1          37
    ...           0       6           0
131 Sandal               1           0           0
    ...          22       1          15
132 Shirt                94           2          34
    ...           0      14           0
133 Sneaker              0           0           0
    ...          558       0          19
134 Bag                  5           1           3
    ...           4      570           0
135 Ankle boot           0           0           0
    ...          30       0          542
136
137 [10 rows x 10 columns]
138
139 Process finished with exit code 0
140

```