wine

April 2, 2023

[ ]:
```python
## Import Necessary Functions
import numpy as np
import matplotlib.pyplot as plt



#You will use Vine Dataset you can get the detail in the URL below
#https://archive.ics.uci.edu/ml/datasets/Wine

from sklearn.datasets import load_wine
vine = load_wine()
```

[ ]:
```python
# PRINT FOR YOUR INTUITION SO YOU KNOW THE DIMENSIONS YOU ARE WORKING WITH
print(vine.data.shape) #.data contains the features
print(vine.target.shape)
print(vine.target[0]) #.target contains the target variables
print(vine.target_names.shape)
print(vine.feature_names)
```

```
(178, 13)
(178,)
0
(3,)
['alcohol', 'malic_acid', 'ash', 'alcalinity_of_ash', 'magnesium',
'total_phenols', 'flavanoids', 'nonflavanoid_phenols', 'proanthocyanins',
'color_intensity', 'hue', 'od280/od315_of_diluted_wines', 'proline']
```

[ ]:
```python
#SPLIT THE DATA INTO TRAINING AND TESTING SET
#WHEN CALLING THE DATA use vine.data instance as well as vine.target.
 ↪astype('int')
#THE SPLIT OF DATA IS UPTO YOU
from tensorflow.keras.utils import to_categorical
from sklearn.model_selection import train_test_split
X, y = load_wine(return_X_y=True)
y = to_categorical(y)
```

```
X2_train, X2_test, y2_train, y2_test = train_test_split(X,
 ↪y,random_state=104,test_size=0.3,shuffle=True)
```

[ ]:
```
X2_train.shape, X2_test.shape
```

[ ]: ((124, 13), (54, 13))

[ ]:
```
y2_train.shape, y2_test.shape
```

[ ]: ((124, 3), (54, 3))

[ ]:
```
print(y2_train)
```

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

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

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

[ ]:
```
print(X2_train)
```

```
[[1.165e+01 1.670e+00 2.620e+00 … 1.360e+00 3.210e+00 5.620e+02]
 [1.383e+01 1.570e+00 2.620e+00 … 1.130e+00 2.570e+00 1.130e+03]
 [1.184e+01 8.900e-01 2.580e+00 … 7.900e-01 3.080e+00 5.200e+02]
```

```
 …
 [1.345e+01 3.700e+00 2.600e+00 … 8.500e-01 1.560e+00 6.950e+02]
 [1.352e+01 3.170e+00 2.720e+00 … 8.900e-01 2.060e+00 5.200e+02]
 [1.221e+01 1.190e+00 1.750e+00 … 1.280e+00 3.070e+00 7.180e+02]]
```

[ ]:
```python
import tensorflow as tf

print(tf.__version__)
```

```
2.11.0
```

[ ]:
```python
input_dim = X2_train.shape[1:]
output_dim = y.shape[1]
```

[ ]:
```python
model = tf.keras.Sequential([
    tf.keras.layers.Dense(128, activation='relu',input_shape=input_dim),
    tf.keras.layers.Dense(256, activation='relu'),
    tf.keras.layers.Dense(output_dim, activation='softmax')
])
```

[ ]:
```python
model.compile(
    loss='categorical_crossentropy',
    optimizer='adam',
    metrics=[
        tf.keras.metrics.BinaryAccuracy(name='accuracy'),
        tf.keras.metrics.Precision(name='precision'),
        tf.keras.metrics.Recall(name='recall')
    ]
)
```

[ ]:
```python
testing = model.fit(X2_train, y2_train, epochs=100)
```

```
Epoch 1/100
4/4 [==============================] - 0s 1ms/step - loss: 30.1028 - accuracy:
0.5591 - precision: 0.3387 - recall: 0.3387
Epoch 2/100
4/4 [==============================] - 0s 1ms/step - loss: 17.9924 - accuracy:
0.5645 - precision: 0.3468 - recall: 0.3468
Epoch 3/100
4/4 [==============================] - 0s 1ms/step - loss: 19.6014 - accuracy:
0.6183 - precision: 0.4274 - recall: 0.4274
Epoch 4/100
4/4 [==============================] - 0s 1ms/step - loss: 15.0139 - accuracy:
0.6667 - precision: 0.5000 - recall: 0.5000
Epoch 5/100
4/4 [==============================] - 0s 1ms/step - loss: 6.8271 - accuracy:
0.6613 - precision: 0.4919 - recall: 0.4919
Epoch 6/100
```

```
4/4 [==============================] - 0s 1ms/step - loss: 4.5598 - accuracy:
0.6774 - precision: 0.5161 - recall: 0.5161
Epoch 7/100
4/4 [==============================] - 0s 1ms/step - loss: 4.9822 - accuracy:
0.7473 - precision: 0.6210 - recall: 0.6210
Epoch 8/100
4/4 [==============================] - 0s 1ms/step - loss: 3.2929 - accuracy:
0.6156 - precision: 0.4228 - recall: 0.4194
Epoch 9/100
4/4 [==============================] - 0s 1ms/step - loss: 2.6168 - accuracy:
0.6720 - precision: 0.5082 - recall: 0.5000
Epoch 10/100
4/4 [==============================] - 0s 1ms/step - loss: 1.5067 - accuracy:
0.7581 - precision: 0.6393 - recall: 0.6290
Epoch 11/100
4/4 [==============================] - 0s 1ms/step - loss: 2.2389 - accuracy:
0.6774 - precision: 0.5161 - recall: 0.5161
Epoch 12/100
4/4 [==============================] - 0s 1ms/step - loss: 1.4263 - accuracy:
0.7930 - precision: 0.6942 - recall: 0.6774
Epoch 13/100
4/4 [==============================] - 0s 1ms/step - loss: 1.2111 - accuracy:
0.7849 - precision: 0.6774 - recall: 0.6774
Epoch 14/100
4/4 [==============================] - 0s 1ms/step - loss: 1.1126 - accuracy:
0.8038 - precision: 0.7107 - recall: 0.6935
Epoch 15/100
4/4 [==============================] - 0s 1ms/step - loss: 0.7713 - accuracy:
0.8629 - precision: 0.8017 - recall: 0.7823
Epoch 16/100
4/4 [==============================] - 0s 1ms/step - loss: 1.0332 - accuracy:
0.8387 - precision: 0.7623 - recall: 0.7500
Epoch 17/100
4/4 [==============================] - 0s 1ms/step - loss: 0.9291 - accuracy:
0.8387 - precision: 0.7623 - recall: 0.7500
Epoch 18/100
4/4 [==============================] - 0s 1ms/step - loss: 1.3703 - accuracy:
0.7527 - precision: 0.6290 - recall: 0.6290
Epoch 19/100
4/4 [==============================] - 0s 1ms/step - loss: 1.2593 - accuracy:
0.7742 - precision: 0.6613 - recall: 0.6613
Epoch 20/100
4/4 [==============================] - 0s 1ms/step - loss: 0.8627 - accuracy:
0.8199 - precision: 0.7355 - recall: 0.7177
Epoch 21/100
4/4 [==============================] - 0s 1ms/step - loss: 0.9627 - accuracy:
0.7849 - precision: 0.6774 - recall: 0.6774
Epoch 22/100
```

```
4/4 [==============================] - 0s 1ms/step - loss: 1.0311 - accuracy:
0.7849 - precision: 0.6774 - recall: 0.6774
Epoch 23/100
4/4 [==============================] - 0s 1ms/step - loss: 0.8313 - accuracy:
0.8468 - precision: 0.7815 - recall: 0.7500
Epoch 24/100
4/4 [==============================] - 0s 1ms/step - loss: 0.9216 - accuracy:
0.8414 - precision: 0.7642 - recall: 0.7581
Epoch 25/100
4/4 [==============================] - 0s 1ms/step - loss: 0.9790 - accuracy:
0.8118 - precision: 0.7250 - recall: 0.7016
Epoch 26/100
4/4 [==============================] - 0s 1ms/step - loss: 1.0288 - accuracy:
0.8280 - precision: 0.7419 - recall: 0.7419
Epoch 27/100
4/4 [==============================] - 0s 1ms/step - loss: 0.7955 - accuracy:
0.8602 - precision: 0.7951 - recall: 0.7823
Epoch 28/100
4/4 [==============================] - 0s 1ms/step - loss: 0.8484 - accuracy:
0.8978 - precision: 0.8525 - recall: 0.8387
Epoch 29/100
4/4 [==============================] - 0s 1ms/step - loss: 0.8420 - accuracy:
0.8118 - precision: 0.7213 - recall: 0.7097
Epoch 30/100
4/4 [==============================] - 0s 1ms/step - loss: 0.7015 - accuracy:
0.9194 - precision: 0.8790 - recall: 0.8790
Epoch 31/100
4/4 [==============================] - 0s 1ms/step - loss: 0.6288 - accuracy:
0.8871 - precision: 0.8417 - recall: 0.8145
Epoch 32/100
4/4 [==============================] - 0s 1ms/step - loss: 0.8802 - accuracy:
0.9032 - precision: 0.8548 - recall: 0.8548
Epoch 33/100
4/4 [==============================] - 0s 1ms/step - loss: 0.8373 - accuracy:
0.8414 - precision: 0.7686 - recall: 0.7500
Epoch 34/100
4/4 [==============================] - 0s 1ms/step - loss: 0.7252 - accuracy:
0.8306 - precision: 0.7480 - recall: 0.7419
Epoch 35/100
4/4 [==============================] - 0s 1ms/step - loss: 0.7673 - accuracy:
0.8387 - precision: 0.7581 - recall: 0.7581
Epoch 36/100
4/4 [==============================] - 0s 1ms/step - loss: 0.6000 - accuracy:
0.8898 - precision: 0.8487 - recall: 0.8145
Epoch 37/100
4/4 [==============================] - 0s 1ms/step - loss: 0.6056 - accuracy:
0.8871 - precision: 0.8417 - recall: 0.8145
Epoch 38/100
```

```
4/4 [==============================] - 0s 1ms/step - loss: 0.7641 - accuracy:
0.9086 - precision: 0.8629 - recall: 0.8629
Epoch 39/100
4/4 [==============================] - 0s 1ms/step - loss: 0.9525 - accuracy:
0.8575 - precision: 0.7886 - recall: 0.7823
Epoch 40/100
4/4 [==============================] - 0s 1ms/step - loss: 1.1454 - accuracy:
0.7849 - precision: 0.6803 - recall: 0.6694
Epoch 41/100
4/4 [==============================] - 0s 1ms/step - loss: 0.6701 - accuracy:
0.8280 - precision: 0.7459 - recall: 0.7339
Epoch 42/100
4/4 [==============================] - 0s 1ms/step - loss: 0.5887 - accuracy:
0.8737 - precision: 0.8182 - recall: 0.7984
Epoch 43/100
4/4 [==============================] - 0s 1ms/step - loss: 1.0246 - accuracy:
0.7742 - precision: 0.6613 - recall: 0.6613
Epoch 44/100
4/4 [==============================] - 0s 1ms/step - loss: 0.8647 - accuracy:
0.8226 - precision: 0.7417 - recall: 0.7177
Epoch 45/100
4/4 [==============================] - 0s 1ms/step - loss: 0.7437 - accuracy:
0.8226 - precision: 0.7339 - recall: 0.7339
Epoch 46/100
4/4 [==============================] - 0s 1ms/step - loss: 0.7502 - accuracy:
0.8253 - precision: 0.7398 - recall: 0.7339
Epoch 47/100
4/4 [==============================] - 0s 1ms/step - loss: 0.4976 - accuracy:
0.8925 - precision: 0.8500 - recall: 0.8226
Epoch 48/100
4/4 [==============================] - 0s 1ms/step - loss: 0.9952 - accuracy:
0.8118 - precision: 0.7177 - recall: 0.7177
Epoch 49/100
4/4 [==============================] - 0s 1ms/step - loss: 1.1356 - accuracy:
0.7876 - precision: 0.6829 - recall: 0.6774
Epoch 50/100
4/4 [==============================] - 0s 1ms/step - loss: 0.6093 - accuracy:
0.8414 - precision: 0.7642 - recall: 0.7581
Epoch 51/100
4/4 [==============================] - 0s 1ms/step - loss: 0.9813 - accuracy:
0.8145 - precision: 0.7273 - recall: 0.7097
Epoch 52/100
4/4 [==============================] - 0s 1ms/step - loss: 0.7654 - accuracy:
0.8468 - precision: 0.7724 - recall: 0.7661
Epoch 53/100
4/4 [==============================] - 0s 1ms/step - loss: 0.7408 - accuracy:
0.8495 - precision: 0.7833 - recall: 0.7581
Epoch 54/100
```

```
4/4 [==============================] - 0s 1ms/step - loss: 0.9612 - accuracy:
0.8468 - precision: 0.7815 - recall: 0.7500
Epoch 55/100
4/4 [==============================] - 0s 1ms/step - loss: 0.9118 - accuracy:
0.8683 - precision: 0.8099 - recall: 0.7903
Epoch 56/100
4/4 [==============================] - 0s 1ms/step - loss: 0.7329 - accuracy:
0.8710 - precision: 0.8065 - recall: 0.8065
Epoch 57/100
4/4 [==============================] - 0s 1ms/step - loss: 0.6445 - accuracy:
0.8710 - precision: 0.8167 - recall: 0.7903
Epoch 58/100
4/4 [==============================] - 0s 1ms/step - loss: 0.9829 - accuracy:
0.8226 - precision: 0.7339 - recall: 0.7339
Epoch 59/100
4/4 [==============================] - 0s 1ms/step - loss: 0.6380 - accuracy:
0.9086 - precision: 0.8689 - recall: 0.8548
Epoch 60/100
4/4 [==============================] - 0s 1ms/step - loss: 1.0275 - accuracy:
0.8414 - precision: 0.7642 - recall: 0.7581
Epoch 61/100
4/4 [==============================] - 0s 1ms/step - loss: 0.7460 - accuracy:
0.8387 - precision: 0.7581 - recall: 0.7581
Epoch 62/100
4/4 [==============================] - 0s 1ms/step - loss: 1.3347 - accuracy:
0.7634 - precision: 0.6452 - recall: 0.6452
Epoch 63/100
4/4 [==============================] - 0s 1ms/step - loss: 0.7390 - accuracy:
0.8602 - precision: 0.7903 - recall: 0.7903
Epoch 64/100
4/4 [==============================] - 0s 1ms/step - loss: 0.6679 - accuracy:
0.8737 - precision: 0.8182 - recall: 0.7984
Epoch 65/100
4/4 [==============================] - 0s 1ms/step - loss: 0.4269 - accuracy:
0.9274 - precision: 0.9008 - recall: 0.8790
Epoch 66/100
4/4 [==============================] - 0s 1ms/step - loss: 0.4202 - accuracy:
0.9032 - precision: 0.8667 - recall: 0.8387
Epoch 67/100
4/4 [==============================] - 0s 1ms/step - loss: 0.5796 - accuracy:
0.8710 - precision: 0.8065 - recall: 0.8065
Epoch 68/100
4/4 [==============================] - 0s 1ms/step - loss: 0.6632 - accuracy:
0.8548 - precision: 0.7869 - recall: 0.7742
Epoch 69/100
4/4 [==============================] - 0s 1ms/step - loss: 0.4619 - accuracy:
0.9086 - precision: 0.8689 - recall: 0.8548
Epoch 70/100
```

```
4/4 [==============================] - 0s 1ms/step - loss: 0.4181 - accuracy:
0.9005 - precision: 0.8537 - recall: 0.8468
Epoch 71/100
4/4 [==============================] - 0s 1ms/step - loss: 0.4253 - accuracy:
0.9382 - precision: 0.9106 - recall: 0.9032
Epoch 72/100
4/4 [==============================] - 0s 1ms/step - loss: 0.2892 - accuracy:
0.9409 - precision: 0.9180 - recall: 0.9032
Epoch 73/100
4/4 [==============================] - 0s 1ms/step - loss: 0.4312 - accuracy:
0.9355 - precision: 0.9032 - recall: 0.9032
Epoch 74/100
4/4 [==============================] - 0s 1ms/step - loss: 0.3425 - accuracy:
0.9516 - precision: 0.9274 - recall: 0.9274
Epoch 75/100
4/4 [==============================] - 0s 1ms/step - loss: 0.6206 - accuracy:
0.8656 - precision: 0.8033 - recall: 0.7903
Epoch 76/100
4/4 [==============================] - 0s 1ms/step - loss: 0.5217 - accuracy:
0.8656 - precision: 0.8033 - recall: 0.7903
Epoch 77/100
4/4 [==============================] - 0s 1ms/step - loss: 0.4521 - accuracy:
0.8925 - precision: 0.8500 - recall: 0.8226
Epoch 78/100
4/4 [==============================] - 0s 1ms/step - loss: 0.6069 - accuracy:
0.8602 - precision: 0.7903 - recall: 0.7903
Epoch 79/100
4/4 [==============================] - 0s 1ms/step - loss: 2.0105 - accuracy:
0.7796 - precision: 0.6694 - recall: 0.6694
Epoch 80/100
4/4 [==============================] - 0s 1ms/step - loss: 1.7041 - accuracy:
0.7634 - precision: 0.6475 - recall: 0.6371
Epoch 81/100
4/4 [==============================] - 0s 1ms/step - loss: 4.0134 - accuracy:
0.6640 - precision: 0.4959 - recall: 0.4919
Epoch 82/100
4/4 [==============================] - 0s 1ms/step - loss: 0.5758 - accuracy:
0.8898 - precision: 0.8374 - recall: 0.8306
Epoch 83/100
4/4 [==============================] - 0s 1ms/step - loss: 0.7820 - accuracy:
0.8683 - precision: 0.8049 - recall: 0.7984
Epoch 84/100
4/4 [==============================] - 0s 1ms/step - loss: 0.6703 - accuracy:
0.9032 - precision: 0.8607 - recall: 0.8468
Epoch 85/100
4/4 [==============================] - 0s 1ms/step - loss: 0.7654 - accuracy:
0.8925 - precision: 0.8387 - recall: 0.8387
Epoch 86/100
```

```
4/4 [==============================] - 0s 1ms/step - loss: 0.5690 - accuracy:
0.9274 - precision: 0.8943 - recall: 0.8871
Epoch 87/100
4/4 [==============================] - 0s 1ms/step - loss: 0.5864 - accuracy:
0.9194 - precision: 0.8852 - recall: 0.8710
Epoch 88/100
4/4 [==============================] - 0s 1ms/step - loss: 0.3759 - accuracy:
0.9462 - precision: 0.9194 - recall: 0.9194
Epoch 89/100
4/4 [==============================] - 0s 1ms/step - loss: 0.3613 - accuracy:
0.9516 - precision: 0.9274 - recall: 0.9274
Epoch 90/100
4/4 [==============================] - 0s 1ms/step - loss: 0.3542 - accuracy:
0.9597 - precision: 0.9431 - recall: 0.9355
Epoch 91/100
4/4 [==============================] - 0s 1ms/step - loss: 0.3624 - accuracy:
0.9301 - precision: 0.8952 - recall: 0.8952
Epoch 92/100
4/4 [==============================] - 0s 1ms/step - loss: 0.7455 - accuracy:
0.8898 - precision: 0.8374 - recall: 0.8306
Epoch 93/100
4/4 [==============================] - 0s 1ms/step - loss: 1.1263 - accuracy:
0.8441 - precision: 0.7661 - recall: 0.7661
Epoch 94/100
4/4 [==============================] - 0s 1ms/step - loss: 1.4408 - accuracy:
0.8011 - precision: 0.7049 - recall: 0.6935
Epoch 95/100
4/4 [==============================] - 0s 1ms/step - loss: 1.2716 - accuracy:
0.8226 - precision: 0.7339 - recall: 0.7339
Epoch 96/100
4/4 [==============================] - 0s 1ms/step - loss: 1.1527 - accuracy:
0.8414 - precision: 0.7642 - recall: 0.7581
Epoch 97/100
4/4 [==============================] - 0s 1ms/step - loss: 0.7229 - accuracy:
0.9005 - precision: 0.8537 - recall: 0.8468
Epoch 98/100
4/4 [==============================] - 0s 1ms/step - loss: 1.1797 - accuracy:
0.8575 - precision: 0.7886 - recall: 0.7823
Epoch 99/100
4/4 [==============================] - 0s 1ms/step - loss: 0.5180 - accuracy:
0.9409 - precision: 0.9113 - recall: 0.9113
Epoch 100/100
4/4 [==============================] - 0s 1ms/step - loss: 0.7762 - accuracy:
0.9113 - precision: 0.8699 - recall: 0.8629
```

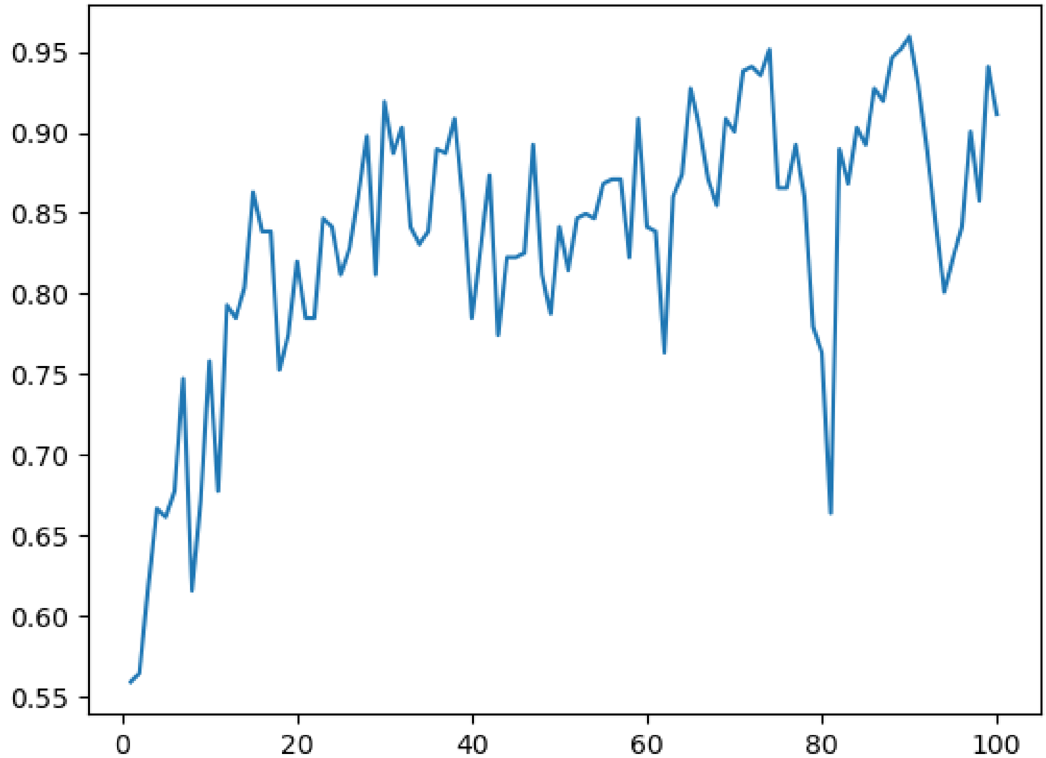
[ ]:
```
print('\nTest accuracy:', testing.history['accuracy'])
```

```
Test accuracy: [0.5591397881507874, 0.5645161271095276, 0.6182795763015747,
0.6666666865348816, 0.6612903475761414, 0.6774193644523621, 0.7473118305206299,
0.6155914068222046, 0.6720430254936218, 0.7580645084381104, 0.6774193644523621,
0.7930107712745667, 0.7849462628364563, 0.8037634491920471, 0.8629032373428345,
0.8387096524238586, 0.8387096524238586, 0.7526881694793701, 0.774193525314331,
0.8198924660682678, 0.7849462628364563, 0.7849462628364563, 0.8467742204666138,
0.8413978219032288, 0.8118279576301575, 0.8279569745063782, 0.8602150678634644,
0.897849440574646, 0.8118279576301575, 0.9193548560142517, 0.8870967626571655,
0.9032257795333862, 0.8413978219032288, 0.8306451439857483, 0.8387096524238586,
0.8897849321365356, 0.8870967626571655, 0.9086021780967712, 0.8575268983840942,
0.7849462628364563, 0.8279569745063782, 0.8736559152603149, 0.774193525314331,
0.8225806355476379, 0.8225806355476379, 0.8252688050270081, 0.8924731016159058,
0.8118279576301575, 0.7876344323158264, 0.8413978219032288, 0.8145161271095276,
0.8467742204666138, 0.8494623899459839, 0.8467742204666138, 0.8682795763015747,
0.8709677457809448, 0.8709677457809448, 0.8225806355476379, 0.9086021780967712,
0.8413978219032288, 0.8387096524238586, 0.7634408473968506, 0.8602150678634644,
0.8736559152603149, 0.9274193644523621, 0.9032257795333862, 0.8709677457809448,
0.8548387289047241, 0.9086021780967712, 0.9005376100540161, 0.9381720423698425,
0.9408602118492126, 0.9354838728904724, 0.9516128897666931, 0.8655914068222046,
0.8655914068222046, 0.8924731016159058, 0.8602150678634644, 0.7795698642730713,
0.7634408473968506, 0.6639785170555115, 0.8897849321365356, 0.8682795763015747,
0.9032257795333862, 0.8924731016159058, 0.9274193644523621, 0.9193548560142517,
0.9462365508079529, 0.9516128897666931, 0.9596773982048035, 0.9301075339317322,
0.8897849321365356, 0.8440860509872437, 0.801075279712677, 0.8225806355476379,
0.8413978219032288, 0.9005376100540161, 0.8575268983840942, 0.9408602118492126,
0.9112903475761414]
```

[ ]:
```python
test_acc = model.evaluate(X2_test, y2_test)


print('\nTest accuracy:', test_acc)

plt.plot(
    np.arange(1, 101),
    testing.history['accuracy'], label='Accuracy'
)
plt.show
```

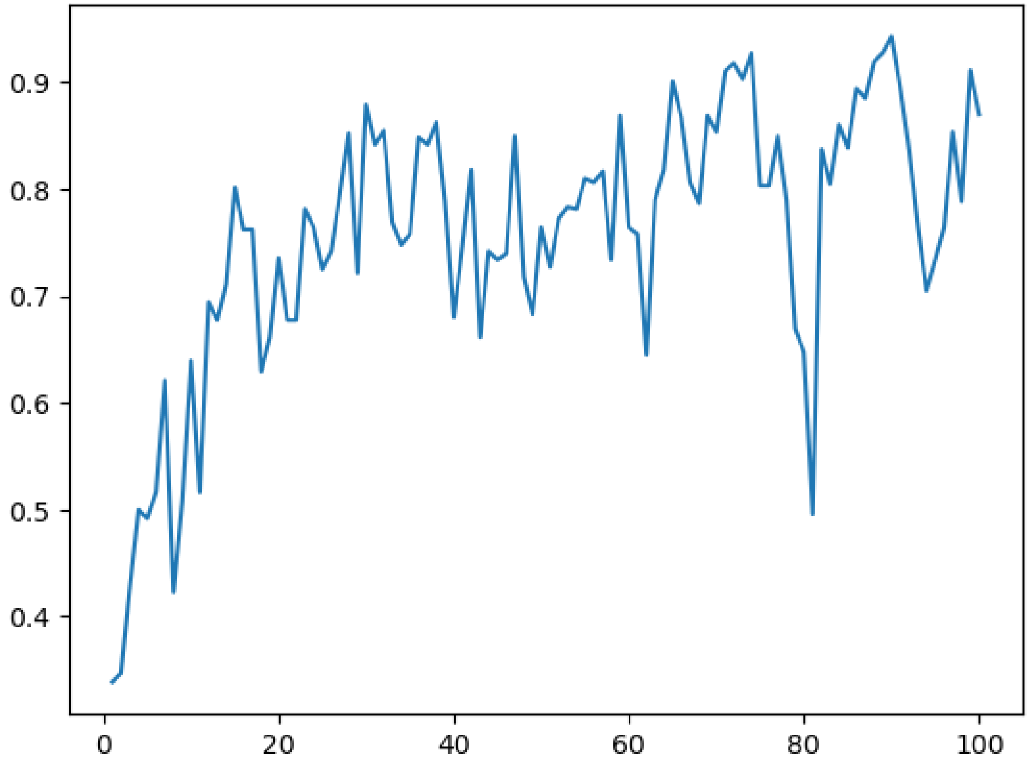
```
2/2 [==============================] - 0s 3ms/step - loss: 0.2309 - accuracy:
0.9259 - precision: 0.8889 - recall: 0.8889

Test accuracy: [0.2308574914932251, 0.9259259104728699, 0.8888888955116272,
0.8888888955116272]
```

[ ]:
```
<function matplotlib.pyplot.show(close=None, block=None)>
```



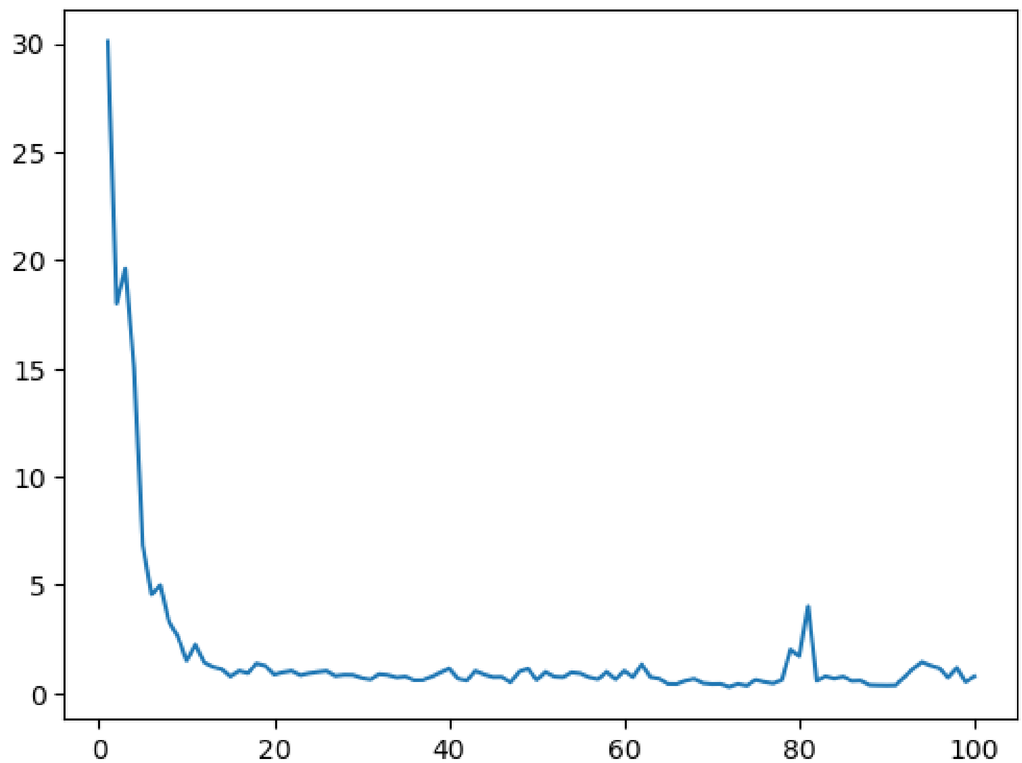
[ ]:
```
plt.plot(
    np.arange(1, 101),
    testing.history['precision'], label='Precision'
)
```

[ ]: [<matplotlib.lines.Line2D at 0x7f7ca41e2860>]



[ ]:
```
plt.plot(
    np.arange(1, 101),
    testing.history['loss'], label='Loss'
)
```

[ ]: [<matplotlib.lines.Line2D at 0x7f7ca449be20>]



[ ]:
```
model.evaluate(X2_test,y2_test)
```

```
2/2 [==============================] - 0s 2ms/step - loss: 0.2309 - accuracy:
0.9259 - precision: 0.8889 - recall: 0.8889
```

[ ]:
```
[0.2308574914932251,
 0.9259259104728699,
 0.8888888955116272,
 0.8888888955116272]
```

Accuracy: 92.59%