hw3

February 13, 2025

```
[3]: # Exercise 1
     # 1.1
     # As the classes are categorical, use one-hot encoding to represent the set of␣
      ↪classes.
     # You will find this useful when developing the output layer of the neural␣
      ↪network.
     # Note that the dataset doesn't include data points from all grading levels,
     # but you may still need to create the one-hot encoding by 10 cols.

     import pandas as pd
     from sklearn.preprocessing import OneHotEncoder, MinMaxScaler

     data = pd.read_csv('winequality-white.csv')

     encoder = OneHotEncoder(categories=[range(1, 11)], sparse=False)
     encoded_labels = encoder.fit_transform(data[['quality']])
     encoded_labels_df = pd.DataFrame(encoded_labels, columns=[f'quality_{i}' for i␣
      ↪in range(1, 11)])
     data = pd.concat([data.drop('quality', axis=1), encoded_labels_df], axis=1)

     # 1.2
     scaler = MinMaxScaler()
     scaled_data = scaler.fit_transform(data.iloc[:, :-10])
     scaled_data_df = pd.DataFrame(scaled_data, columns=data.columns[:-10])
     final_data = pd.concat([scaled_data_df, encoded_labels_df], axis=1)

     final_data.to_csv('preprocessed_winequality_data.csv', index=False)
```

```
/opt/anaconda3/lib/python3.11/site-
packages/sklearn/preprocessing/_encoders.py:868: FutureWarning: `sparse` was
renamed to `sparse_output` in version 1.2 and will be removed in 1.4.
`sparse_output` is ignored unless you leave `sparse` to its default value.
  warnings.warn(
```

```
[14]: # Exercise 3
      import numpy as np
      import pandas as pd
      import tensorflow as tf
```

```python
from tensorflow import keras
from sklearn.model_selection import train_test_split
from sklearn.metrics import confusion_matrix, mean_squared_error
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.utils.class_weight import compute_class_weight
from imblearn.over_sampling import RandomOverSampler


# Load the data
data = pd.read_csv('preprocessed_winequality_data.csv')

# input data
X = data.iloc[:, :-10].values.astype(np.float32)

# output data
y = data.iloc[:, -10:].values.astype(np.float32)

# Compute class weights
y_labels = np.argmax(y, axis=1)  # Convert one-hot encoding to class labels
class_weights = compute_class_weight(class_weight="balanced", classes=np.
 ↪unique(y_labels), y=y_labels)
class_weight_dict = {i: class_weights[i] for i in range(len(class_weights))}

# Apply oversampling to balance classes
ros = RandomOverSampler(random_state=42)
X_resampled, y_resampled = ros.fit_resample(X, y)

# Split the data
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.1,␣
 ↪random_state=42)

# Build a model
model = keras.Sequential([
    keras.layers.Dense(23, activation='sigmoid', input_shape=(11,)),
    keras.layers.Dense(17, activation='sigmoid'),
    keras.layers.Dense(13, activation='sigmoid'),
    keras.layers.Dense(10, activation='sigmoid')
])

# Compile a model
model.compile(optimizer=keras.optimizers.SGD(learning_rate=0.4),
              loss='mean_squared_error',
              metrics=['accuracy'])

model.summary()
```

```python
# Train the model
history = model.fit(X_train, y_train, epochs=500, batch_size=32,␣
 ↪validation_data=(X_test, y_test), class_weight=class_weight_dict)

# Evaluate the data
y_pred = model.predict(X_test)

y_test_labels = np.argmax(y_test, axis=1)
y_pred_labels = np.argmax(y_pred, axis=1)

conf_matrx = confusion_matrix(y_test_labels, y_pred_labels)

plt.figure(figsize=(10, 7))
sns.heatmap(conf_matrx, annot=True, fmt='d', cmap='Blues')
plt.xlabel("Predicted Label")
plt.ylabel("True Label")
plt.title("Confusion Matrix")
plt.show()
```

/opt/anaconda3/lib/python3.11/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.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)

Model: "sequential_8"

Layer (type)	Output Shape	Param #
dense_32 (Dense)	(None, 23)	276
dense_33 (Dense)	(None, 17)	408
dense_34 (Dense)	(None, 13)	234
dense_35 (Dense)	(None, 10)	140

 Total params: 1,058 (4.13 KB)

 Trainable params: 1,058 (4.13 KB)

 Non-trainable params: 0 (0.00 B)

Epoch 1/500

```
138/138              1s 2ms/step -
accuracy: 0.1743 - loss: 0.7020 - val_accuracy: 0.1918 - val_loss: 0.1465
Epoch 2/500
138/138              0s 1ms/step -
accuracy: 0.1792 - loss: 0.3896 - val_accuracy: 0.1918 - val_loss: 0.1444
Epoch 3/500
138/138              0s 971us/step -
accuracy: 0.1912 - loss: 0.3865 - val_accuracy: 0.1918 - val_loss: 0.1437
Epoch 4/500
138/138              0s 948us/step -
accuracy: 0.1799 - loss: 0.3933 - val_accuracy: 0.1918 - val_loss: 0.1439
Epoch 5/500
138/138              0s 963us/step -
accuracy: 0.1775 - loss: 0.3942 - val_accuracy: 0.1918 - val_loss: 0.1425
Epoch 6/500
138/138              0s 933us/step -
accuracy: 0.1901 - loss: 0.3864 - val_accuracy: 0.1918 - val_loss: 0.1431
Epoch 7/500
138/138              0s 950us/step -
accuracy: 0.1814 - loss: 0.3916 - val_accuracy: 0.1918 - val_loss: 0.1420
Epoch 8/500
138/138              0s 947us/step -
accuracy: 0.1850 - loss: 0.3874 - val_accuracy: 0.1918 - val_loss: 0.1424
Epoch 9/500
138/138              0s 947us/step -
accuracy: 0.1739 - loss: 0.3910 - val_accuracy: 0.1918 - val_loss: 0.1449
Epoch 10/500
138/138              0s 2ms/step -
accuracy: 0.1759 - loss: 0.3992 - val_accuracy: 0.1918 - val_loss: 0.1455
Epoch 11/500
138/138              0s 938us/step -
accuracy: 0.1850 - loss: 0.3858 - val_accuracy: 0.1918 - val_loss: 0.1430
Epoch 12/500
138/138              0s 937us/step -
accuracy: 0.1731 - loss: 0.3926 - val_accuracy: 0.1918 - val_loss: 0.1456
Epoch 13/500
138/138              0s 926us/step -
accuracy: 0.1738 - loss: 0.3887 - val_accuracy: 0.1918 - val_loss: 0.1432
Epoch 14/500
138/138              0s 964us/step -
accuracy: 0.1747 - loss: 0.3931 - val_accuracy: 0.1918 - val_loss: 0.1438
Epoch 15/500
138/138              0s 926us/step -
accuracy: 0.1833 - loss: 0.3799 - val_accuracy: 0.1918 - val_loss: 0.1438
Epoch 16/500
138/138              0s 970us/step -
accuracy: 0.1695 - loss: 0.3899 - val_accuracy: 0.1918 - val_loss: 0.1441
Epoch 17/500
```

```
138/138                0s 933us/step -
accuracy: 0.1788 - loss: 0.3887 - val_accuracy: 0.1918 - val_loss: 0.1443
Epoch 18/500
138/138                0s 925us/step -
accuracy: 0.1841 - loss: 0.3838 - val_accuracy: 0.1918 - val_loss: 0.1445
Epoch 19/500
138/138                0s 949us/step -
accuracy: 0.1711 - loss: 0.3916 - val_accuracy: 0.1918 - val_loss: 0.1431
Epoch 20/500
138/138                0s 1ms/step -
accuracy: 0.1647 - loss: 0.3930 - val_accuracy: 0.1918 - val_loss: 0.1471
Epoch 21/500
138/138                0s 943us/step -
accuracy: 0.1679 - loss: 0.3944 - val_accuracy: 0.1918 - val_loss: 0.1452
Epoch 22/500
138/138                0s 919us/step -
accuracy: 0.1712 - loss: 0.3873 - val_accuracy: 0.1918 - val_loss: 0.1448
Epoch 23/500
138/138                0s 1ms/step -
accuracy: 0.1745 - loss: 0.3941 - val_accuracy: 0.1918 - val_loss: 0.1421
Epoch 24/500
138/138                0s 965us/step -
accuracy: 0.1749 - loss: 0.3865 - val_accuracy: 0.1918 - val_loss: 0.1458
Epoch 25/500
138/138                0s 961us/step -
accuracy: 0.1777 - loss: 0.3882 - val_accuracy: 0.1918 - val_loss: 0.1428
Epoch 26/500
138/138                0s 948us/step -
accuracy: 0.1815 - loss: 0.3932 - val_accuracy: 0.1918 - val_loss: 0.1426
Epoch 27/500
138/138                0s 914us/step -
accuracy: 0.1782 - loss: 0.3902 - val_accuracy: 0.1918 - val_loss: 0.1429
Epoch 28/500
138/138                0s 934us/step -
accuracy: 0.1734 - loss: 0.3951 - val_accuracy: 0.1918 - val_loss: 0.1450
Epoch 29/500
138/138                0s 959us/step -
accuracy: 0.1774 - loss: 0.3872 - val_accuracy: 0.1918 - val_loss: 0.1455
Epoch 30/500
138/138                0s 924us/step -
accuracy: 0.1804 - loss: 0.3839 - val_accuracy: 0.1918 - val_loss: 0.1456
Epoch 31/500
138/138                0s 911us/step -
accuracy: 0.1720 - loss: 0.3930 - val_accuracy: 0.1918 - val_loss: 0.1449
Epoch 32/500
138/138                0s 1ms/step -
accuracy: 0.1807 - loss: 0.3924 - val_accuracy: 0.1918 - val_loss: 0.1422
Epoch 33/500
```

```
138/138            0s 949us/step -
accuracy: 0.1852 - loss: 0.3889 - val_accuracy: 0.1918 - val_loss: 0.1421
Epoch 34/500
138/138            0s 1ms/step -
accuracy: 0.1754 - loss: 0.3888 - val_accuracy: 0.1918 - val_loss: 0.1430
Epoch 35/500
138/138            0s 955us/step -
accuracy: 0.1823 - loss: 0.3860 - val_accuracy: 0.1918 - val_loss: 0.1426
Epoch 36/500
138/138            0s 911us/step -
accuracy: 0.1693 - loss: 0.3877 - val_accuracy: 0.1918 - val_loss: 0.1450
Epoch 37/500
138/138            0s 950us/step -
accuracy: 0.1895 - loss: 0.3891 - val_accuracy: 0.1918 - val_loss: 0.1419
Epoch 38/500
138/138            0s 914us/step -
accuracy: 0.1699 - loss: 0.3945 - val_accuracy: 0.1918 - val_loss: 0.1439
Epoch 39/500
138/138            0s 934us/step -
accuracy: 0.1757 - loss: 0.3867 - val_accuracy: 0.1918 - val_loss: 0.1437
Epoch 40/500
138/138            0s 933us/step -
accuracy: 0.1732 - loss: 0.3916 - val_accuracy: 0.1918 - val_loss: 0.1433
Epoch 41/500
138/138            0s 953us/step -
accuracy: 0.1783 - loss: 0.3868 - val_accuracy: 0.1918 - val_loss: 0.1451
Epoch 42/500
138/138            0s 2ms/step -
accuracy: 0.1860 - loss: 0.3907 - val_accuracy: 0.1918 - val_loss: 0.1445
Epoch 43/500
138/138            0s 967us/step -
accuracy: 0.1739 - loss: 0.3940 - val_accuracy: 0.1918 - val_loss: 0.1447
Epoch 44/500
138/138            0s 946us/step -
accuracy: 0.1778 - loss: 0.3913 - val_accuracy: 0.1918 - val_loss: 0.1451
Epoch 45/500
138/138            0s 938us/step -
accuracy: 0.1678 - loss: 0.3957 - val_accuracy: 0.1918 - val_loss: 0.1441
Epoch 46/500
138/138            0s 936us/step -
accuracy: 0.1880 - loss: 0.3850 - val_accuracy: 0.1918 - val_loss: 0.1440
Epoch 47/500
138/138            0s 946us/step -
accuracy: 0.1690 - loss: 0.3897 - val_accuracy: 0.1918 - val_loss: 0.1453
Epoch 48/500
138/138            0s 905us/step -
accuracy: 0.1840 - loss: 0.3880 - val_accuracy: 0.1918 - val_loss: 0.1439
Epoch 49/500
```

```
138/138          0s 961us/step -
accuracy: 0.1764 - loss: 0.3890 - val_accuracy: 0.1918 - val_loss: 0.1447
Epoch 50/500
138/138          0s 997us/step -
accuracy: 0.1827 - loss: 0.3916 - val_accuracy: 0.1918 - val_loss: 0.1434
Epoch 51/500
138/138          0s 1ms/step -
accuracy: 0.1728 - loss: 0.3958 - val_accuracy: 0.1918 - val_loss: 0.1459
Epoch 52/500
138/138          0s 953us/step -
accuracy: 0.1797 - loss: 0.3975 - val_accuracy: 0.1918 - val_loss: 0.1454
Epoch 53/500
138/138          0s 931us/step -
accuracy: 0.1742 - loss: 0.3884 - val_accuracy: 0.1918 - val_loss: 0.1438
Epoch 54/500
138/138          0s 913us/step -
accuracy: 0.1665 - loss: 0.3942 - val_accuracy: 0.1918 - val_loss: 0.1438
Epoch 55/500
138/138          0s 941us/step -
accuracy: 0.1763 - loss: 0.3939 - val_accuracy: 0.1918 - val_loss: 0.1443
Epoch 56/500
138/138          0s 912us/step -
accuracy: 0.1794 - loss: 0.3896 - val_accuracy: 0.1918 - val_loss: 0.1400
Epoch 57/500
138/138          0s 917us/step -
accuracy: 0.1750 - loss: 0.3909 - val_accuracy: 0.1918 - val_loss: 0.1432
Epoch 58/500
138/138          0s 1ms/step -
accuracy: 0.1826 - loss: 0.3835 - val_accuracy: 0.1918 - val_loss: 0.1442
Epoch 59/500
138/138          0s 959us/step -
accuracy: 0.1793 - loss: 0.3867 - val_accuracy: 0.1918 - val_loss: 0.1443
Epoch 60/500
138/138          0s 960us/step -
accuracy: 0.1758 - loss: 0.3885 - val_accuracy: 0.1918 - val_loss: 0.1422
Epoch 61/500
138/138          0s 926us/step -
accuracy: 0.1782 - loss: 0.3944 - val_accuracy: 0.1918 - val_loss: 0.1440
Epoch 62/500
138/138          0s 1ms/step -
accuracy: 0.1877 - loss: 0.3827 - val_accuracy: 0.1918 - val_loss: 0.1435
Epoch 63/500
138/138          0s 926us/step -
accuracy: 0.1748 - loss: 0.3883 - val_accuracy: 0.1918 - val_loss: 0.1470
Epoch 64/500
138/138          0s 917us/step -
accuracy: 0.1639 - loss: 0.3957 - val_accuracy: 0.1918 - val_loss: 0.1435
Epoch 65/500
```

```
138/138              0s 992us/step -
accuracy: 0.1839 - loss: 0.3872 - val_accuracy: 0.1918 - val_loss: 0.1429
Epoch 66/500
138/138              0s 938us/step -
accuracy: 0.1693 - loss: 0.3911 - val_accuracy: 0.1918 - val_loss: 0.1452
Epoch 67/500
138/138              0s 928us/step -
accuracy: 0.1690 - loss: 0.3928 - val_accuracy: 0.1918 - val_loss: 0.1446
Epoch 68/500
138/138              0s 914us/step -
accuracy: 0.1870 - loss: 0.3844 - val_accuracy: 0.1918 - val_loss: 0.1425
Epoch 69/500
138/138              0s 971us/step -
accuracy: 0.1828 - loss: 0.3853 - val_accuracy: 0.1918 - val_loss: 0.1434
Epoch 70/500
138/138              0s 1ms/step -
accuracy: 0.1775 - loss: 0.3944 - val_accuracy: 0.1918 - val_loss: 0.1449
Epoch 71/500
138/138              0s 965us/step -
accuracy: 0.1726 - loss: 0.3871 - val_accuracy: 0.1918 - val_loss: 0.1459
Epoch 72/500
138/138              0s 949us/step -
accuracy: 0.1746 - loss: 0.3846 - val_accuracy: 0.1918 - val_loss: 0.1444
Epoch 73/500
138/138              0s 924us/step -
accuracy: 0.1719 - loss: 0.3897 - val_accuracy: 0.1918 - val_loss: 0.1445
Epoch 74/500
138/138              0s 939us/step -
accuracy: 0.1759 - loss: 0.3848 - val_accuracy: 0.1918 - val_loss: 0.1414
Epoch 75/500
138/138              0s 1ms/step -
accuracy: 0.1749 - loss: 0.3891 - val_accuracy: 0.1918 - val_loss: 0.1442
Epoch 76/500
138/138              0s 963us/step -
accuracy: 0.1724 - loss: 0.3876 - val_accuracy: 0.1918 - val_loss: 0.1452
Epoch 77/500
138/138              0s 967us/step -
accuracy: 0.1828 - loss: 0.3870 - val_accuracy: 0.1918 - val_loss: 0.1446
Epoch 78/500
138/138              0s 940us/step -
accuracy: 0.1728 - loss: 0.3808 - val_accuracy: 0.1918 - val_loss: 0.1439
Epoch 79/500
138/138              0s 938us/step -
accuracy: 0.1826 - loss: 0.3901 - val_accuracy: 0.1918 - val_loss: 0.1425
Epoch 80/500
138/138              0s 924us/step -
accuracy: 0.1721 - loss: 0.3954 - val_accuracy: 0.1918 - val_loss: 0.1446
Epoch 81/500
```

```
138/138              0s 926us/step -
accuracy: 0.1849 - loss: 0.3783 - val_accuracy: 0.1918 - val_loss: 0.1420
Epoch 82/500
138/138              0s 947us/step -
accuracy: 0.1770 - loss: 0.3923 - val_accuracy: 0.1918 - val_loss: 0.1443
Epoch 83/500
138/138              0s 901us/step -
accuracy: 0.1747 - loss: 0.3847 - val_accuracy: 0.1918 - val_loss: 0.1451
Epoch 84/500
138/138              0s 1ms/step -
accuracy: 0.1835 - loss: 0.3876 - val_accuracy: 0.1918 - val_loss: 0.1436
Epoch 85/500
138/138              0s 940us/step -
accuracy: 0.1758 - loss: 0.3863 - val_accuracy: 0.1918 - val_loss: 0.1428
Epoch 86/500
138/138              0s 910us/step -
accuracy: 0.1727 - loss: 0.3901 - val_accuracy: 0.1918 - val_loss: 0.1430
Epoch 87/500
138/138              0s 930us/step -
accuracy: 0.1658 - loss: 0.3893 - val_accuracy: 0.1918 - val_loss: 0.1460
Epoch 88/500
138/138              0s 952us/step -
accuracy: 0.1768 - loss: 0.3930 - val_accuracy: 0.1918 - val_loss: 0.1465
Epoch 89/500
138/138              0s 941us/step -
accuracy: 0.1764 - loss: 0.3944 - val_accuracy: 0.1918 - val_loss: 0.1442
Epoch 90/500
138/138              0s 932us/step -
accuracy: 0.1750 - loss: 0.3842 - val_accuracy: 0.1918 - val_loss: 0.1424
Epoch 91/500
138/138              0s 909us/step -
accuracy: 0.1838 - loss: 0.3820 - val_accuracy: 0.1918 - val_loss: 0.1420
Epoch 92/500
138/138              0s 907us/step -
accuracy: 0.1742 - loss: 0.3883 - val_accuracy: 0.1918 - val_loss: 0.1449
Epoch 93/500
138/138              0s 970us/step -
accuracy: 0.1756 - loss: 0.3900 - val_accuracy: 0.1918 - val_loss: 0.1439
Epoch 94/500
138/138              0s 1ms/step -
accuracy: 0.1805 - loss: 0.3868 - val_accuracy: 0.1918 - val_loss: 0.1438
Epoch 95/500
138/138              0s 912us/step -
accuracy: 0.1863 - loss: 0.3851 - val_accuracy: 0.1918 - val_loss: 0.1421
Epoch 96/500
138/138              0s 1ms/step -
accuracy: 0.1810 - loss: 0.3855 - val_accuracy: 0.1918 - val_loss: 0.1445
Epoch 97/500
```

```
138/138              0s 912us/step -
accuracy: 0.1882 - loss: 0.3888 - val_accuracy: 0.1918 - val_loss: 0.1451
Epoch 98/500
138/138              0s 934us/step -
accuracy: 0.1765 - loss: 0.3940 - val_accuracy: 0.1918 - val_loss: 0.1447
Epoch 99/500
138/138              0s 917us/step -
accuracy: 0.1687 - loss: 0.3957 - val_accuracy: 0.1918 - val_loss: 0.1431
Epoch 100/500
138/138              0s 922us/step -
accuracy: 0.1812 - loss: 0.3932 - val_accuracy: 0.1918 - val_loss: 0.1421
Epoch 101/500
138/138              0s 1ms/step -
accuracy: 0.1800 - loss: 0.3816 - val_accuracy: 0.1918 - val_loss: 0.1414
Epoch 102/500
138/138              0s 908us/step -
accuracy: 0.1820 - loss: 0.3786 - val_accuracy: 0.1918 - val_loss: 0.1442
Epoch 103/500
138/138              0s 926us/step -
accuracy: 0.1804 - loss: 0.3861 - val_accuracy: 0.1918 - val_loss: 0.1434
Epoch 104/500
138/138              0s 908us/step -
accuracy: 0.1771 - loss: 0.3887 - val_accuracy: 0.1918 - val_loss: 0.1416
Epoch 105/500
138/138              0s 1ms/step -
accuracy: 0.1712 - loss: 0.3850 - val_accuracy: 0.1918 - val_loss: 0.1440
Epoch 106/500
138/138              0s 915us/step -
accuracy: 0.1805 - loss: 0.3825 - val_accuracy: 0.1918 - val_loss: 0.1450
Epoch 107/500
138/138              0s 1ms/step -
accuracy: 0.1717 - loss: 0.3859 - val_accuracy: 0.1918 - val_loss: 0.1445
Epoch 108/500
138/138              0s 910us/step -
accuracy: 0.1757 - loss: 0.3879 - val_accuracy: 0.1918 - val_loss: 0.1425
Epoch 109/500
138/138              0s 941us/step -
accuracy: 0.1744 - loss: 0.3846 - val_accuracy: 0.1918 - val_loss: 0.1405
Epoch 110/500
138/138              0s 922us/step -
accuracy: 0.1912 - loss: 0.3790 - val_accuracy: 0.1918 - val_loss: 0.1427
Epoch 111/500
138/138              0s 922us/step -
accuracy: 0.1774 - loss: 0.3890 - val_accuracy: 0.1918 - val_loss: 0.1409
Epoch 112/500
138/138              0s 927us/step -
accuracy: 0.1821 - loss: 0.3879 - val_accuracy: 0.1918 - val_loss: 0.1428
Epoch 113/500
```

```
138/138              0s 909us/step -
accuracy: 0.1782 - loss: 0.3921 - val_accuracy: 0.1918 - val_loss: 0.1420
Epoch 114/500
138/138              0s 919us/step -
accuracy: 0.1817 - loss: 0.3803 - val_accuracy: 0.1918 - val_loss: 0.1428
Epoch 115/500
138/138              0s 926us/step -
accuracy: 0.1788 - loss: 0.3907 - val_accuracy: 0.1918 - val_loss: 0.1451
Epoch 116/500
138/138              0s 932us/step -
accuracy: 0.1752 - loss: 0.3828 - val_accuracy: 0.1918 - val_loss: 0.1446
Epoch 117/500
138/138              0s 922us/step -
accuracy: 0.1760 - loss: 0.3887 - val_accuracy: 0.1918 - val_loss: 0.1438
Epoch 118/500
138/138              0s 1ms/step -
accuracy: 0.1718 - loss: 0.3874 - val_accuracy: 0.1918 - val_loss: 0.1429
Epoch 119/500
138/138              0s 946us/step -
accuracy: 0.1813 - loss: 0.3902 - val_accuracy: 0.1918 - val_loss: 0.1409
Epoch 120/500
138/138              0s 942us/step -
accuracy: 0.1799 - loss: 0.3886 - val_accuracy: 0.1918 - val_loss: 0.1451
Epoch 121/500
138/138              0s 914us/step -
accuracy: 0.1848 - loss: 0.3868 - val_accuracy: 0.1918 - val_loss: 0.1428
Epoch 122/500
138/138              0s 946us/step -
accuracy: 0.1728 - loss: 0.3897 - val_accuracy: 0.1918 - val_loss: 0.1457
Epoch 123/500
138/138              0s 939us/step -
accuracy: 0.1743 - loss: 0.3938 - val_accuracy: 0.1918 - val_loss: 0.1449
Epoch 124/500
138/138              0s 932us/step -
accuracy: 0.1857 - loss: 0.3805 - val_accuracy: 0.1918 - val_loss: 0.1420
Epoch 125/500
138/138              0s 967us/step -
accuracy: 0.1777 - loss: 0.3887 - val_accuracy: 0.1918 - val_loss: 0.1425
Epoch 126/500
138/138              0s 928us/step -
accuracy: 0.1841 - loss: 0.3847 - val_accuracy: 0.1918 - val_loss: 0.1418
Epoch 127/500
138/138              0s 923us/step -
accuracy: 0.1707 - loss: 0.3895 - val_accuracy: 0.1918 - val_loss: 0.1450
Epoch 128/500
138/138              0s 1ms/step -
accuracy: 0.1831 - loss: 0.3829 - val_accuracy: 0.1918 - val_loss: 0.1435
Epoch 129/500
```

```
138/138            0s 941us/step -
accuracy: 0.1749 - loss: 0.3832 - val_accuracy: 0.1918 - val_loss: 0.1445
Epoch 130/500
138/138            0s 942us/step -
accuracy: 0.1785 - loss: 0.3915 - val_accuracy: 0.1918 - val_loss: 0.1443
Epoch 131/500
138/138            0s 1ms/step -
accuracy: 0.1767 - loss: 0.3863 - val_accuracy: 0.1918 - val_loss: 0.1413
Epoch 132/500
138/138            0s 966us/step -
accuracy: 0.1756 - loss: 0.3895 - val_accuracy: 0.1918 - val_loss: 0.1406
Epoch 133/500
138/138            0s 961us/step -
accuracy: 0.1750 - loss: 0.3920 - val_accuracy: 0.1918 - val_loss: 0.1421
Epoch 134/500
138/138            0s 1ms/step -
accuracy: 0.1722 - loss: 0.3916 - val_accuracy: 0.1918 - val_loss: 0.1448
Epoch 135/500
138/138            0s 909us/step -
accuracy: 0.1791 - loss: 0.3847 - val_accuracy: 0.1918 - val_loss: 0.1371
Epoch 136/500
138/138            0s 941us/step -
accuracy: 0.1830 - loss: 0.3916 - val_accuracy: 0.1918 - val_loss: 0.1415
Epoch 137/500
138/138            0s 928us/step -
accuracy: 0.1800 - loss: 0.3905 - val_accuracy: 0.1918 - val_loss: 0.1424
Epoch 138/500
138/138            0s 908us/step -
accuracy: 0.1809 - loss: 0.3849 - val_accuracy: 0.1918 - val_loss: 0.1433
Epoch 139/500
138/138            0s 929us/step -
accuracy: 0.1706 - loss: 0.3858 - val_accuracy: 0.1918 - val_loss: 0.1428
Epoch 140/500
138/138            0s 914us/step -
accuracy: 0.1778 - loss: 0.3856 - val_accuracy: 0.1918 - val_loss: 0.1392
Epoch 141/500
138/138            0s 942us/step -
accuracy: 0.1853 - loss: 0.3825 - val_accuracy: 0.1918 - val_loss: 0.1406
Epoch 142/500
138/138            0s 939us/step -
accuracy: 0.1779 - loss: 0.3771 - val_accuracy: 0.1918 - val_loss: 0.1396
Epoch 143/500
138/138            0s 932us/step -
accuracy: 0.1758 - loss: 0.3858 - val_accuracy: 0.1918 - val_loss: 0.1374
Epoch 144/500
138/138            0s 950us/step -
accuracy: 0.1786 - loss: 0.3817 - val_accuracy: 0.1918 - val_loss: 0.1382
Epoch 145/500
```

```
138/138            0s 909us/step -
accuracy: 0.1922 - loss: 0.3853 - val_accuracy: 0.1918 - val_loss: 0.1407
Epoch 146/500
138/138            0s 917us/step -
accuracy: 0.1743 - loss: 0.3854 - val_accuracy: 0.1918 - val_loss: 0.1382
Epoch 147/500
138/138            0s 911us/step -
accuracy: 0.1874 - loss: 0.3853 - val_accuracy: 0.1918 - val_loss: 0.1409
Epoch 148/500
138/138            0s 922us/step -
accuracy: 0.1774 - loss: 0.3774 - val_accuracy: 0.1918 - val_loss: 0.1377
Epoch 149/500
138/138            0s 1ms/step -
accuracy: 0.1913 - loss: 0.3790 - val_accuracy: 0.1918 - val_loss: 0.1344
Epoch 150/500
138/138            0s 972us/step -
accuracy: 0.1758 - loss: 0.3768 - val_accuracy: 0.1918 - val_loss: 0.1345
Epoch 151/500
138/138            0s 917us/step -
accuracy: 0.1750 - loss: 0.3813 - val_accuracy: 0.1918 - val_loss: 0.1382
Epoch 152/500
138/138            0s 934us/step -
accuracy: 0.1771 - loss: 0.3829 - val_accuracy: 0.1918 - val_loss: 0.1358
Epoch 153/500
138/138            0s 931us/step -
accuracy: 0.1719 - loss: 0.3810 - val_accuracy: 0.1918 - val_loss: 0.1395
Epoch 154/500
138/138            0s 995us/step -
accuracy: 0.1838 - loss: 0.3793 - val_accuracy: 0.1918 - val_loss: 0.1329
Epoch 155/500
138/138            0s 928us/step -
accuracy: 0.1789 - loss: 0.3797 - val_accuracy: 0.1918 - val_loss: 0.1350
Epoch 156/500
138/138            0s 900us/step -
accuracy: 0.1740 - loss: 0.3846 - val_accuracy: 0.1918 - val_loss: 0.1345
Epoch 157/500
138/138            0s 1ms/step -
accuracy: 0.1852 - loss: 0.3806 - val_accuracy: 0.1918 - val_loss: 0.1322
Epoch 158/500
138/138            0s 918us/step -
accuracy: 0.1748 - loss: 0.3784 - val_accuracy: 0.1918 - val_loss: 0.1379
Epoch 159/500
138/138            0s 928us/step -
accuracy: 0.1804 - loss: 0.3758 - val_accuracy: 0.1918 - val_loss: 0.1361
Epoch 160/500
138/138            0s 915us/step -
accuracy: 0.1709 - loss: 0.3804 - val_accuracy: 0.1918 - val_loss: 0.1423
Epoch 161/500
```

```
138/138              0s 903us/step -
accuracy: 0.1826 - loss: 0.3806 - val_accuracy: 0.1918 - val_loss: 0.1372
Epoch 162/500
138/138              0s 912us/step -
accuracy: 0.1711 - loss: 0.3834 - val_accuracy: 0.1918 - val_loss: 0.1391
Epoch 163/500
138/138              0s 1ms/step -
accuracy: 0.1745 - loss: 0.3800 - val_accuracy: 0.1918 - val_loss: 0.1240
Epoch 164/500
138/138              0s 1ms/step -
accuracy: 0.1830 - loss: 0.3776 - val_accuracy: 0.1918 - val_loss: 0.1398
Epoch 165/500
138/138              0s 901us/step -
accuracy: 0.1724 - loss: 0.3776 - val_accuracy: 0.1918 - val_loss: 0.1357
Epoch 166/500
138/138              0s 932us/step -
accuracy: 0.1695 - loss: 0.3799 - val_accuracy: 0.1918 - val_loss: 0.1318
Epoch 167/500
138/138              0s 932us/step -
accuracy: 0.1779 - loss: 0.3815 - val_accuracy: 0.1918 - val_loss: 0.1384
Epoch 168/500
138/138              0s 914us/step -
accuracy: 0.1787 - loss: 0.3795 - val_accuracy: 0.1918 - val_loss: 0.1367
Epoch 169/500
138/138              0s 913us/step -
accuracy: 0.1763 - loss: 0.3811 - val_accuracy: 0.1918 - val_loss: 0.1390
Epoch 170/500
138/138              0s 930us/step -
accuracy: 0.1742 - loss: 0.3823 - val_accuracy: 0.1918 - val_loss: 0.1358
Epoch 171/500
138/138              0s 917us/step -
accuracy: 0.1725 - loss: 0.3757 - val_accuracy: 0.1918 - val_loss: 0.1410
Epoch 172/500
138/138              0s 925us/step -
accuracy: 0.1788 - loss: 0.3774 - val_accuracy: 0.1918 - val_loss: 0.1403
Epoch 173/500
138/138              0s 936us/step -
accuracy: 0.1825 - loss: 0.3776 - val_accuracy: 0.1918 - val_loss: 0.1183
Epoch 174/500
138/138              0s 1ms/step -
accuracy: 0.1828 - loss: 0.3785 - val_accuracy: 0.1918 - val_loss: 0.1397
Epoch 175/500
138/138              0s 931us/step -
accuracy: 0.1838 - loss: 0.3789 - val_accuracy: 0.1918 - val_loss: 0.1345
Epoch 176/500
138/138              0s 933us/step -
accuracy: 0.1779 - loss: 0.3701 - val_accuracy: 0.1918 - val_loss: 0.1336
Epoch 177/500
```

```
138/138              0s 904us/step -
accuracy: 0.1833 - loss: 0.3699 - val_accuracy: 0.1918 - val_loss: 0.1381
Epoch 178/500
138/138              0s 928us/step -
accuracy: 0.1785 - loss: 0.3813 - val_accuracy: 0.1918 - val_loss: 0.1367
Epoch 179/500
138/138              0s 1ms/step -
accuracy: 0.1763 - loss: 0.3767 - val_accuracy: 0.1918 - val_loss: 0.1428
Epoch 180/500
138/138              0s 966us/step -
accuracy: 0.1750 - loss: 0.3769 - val_accuracy: 0.1918 - val_loss: 0.1364
Epoch 181/500
138/138              0s 920us/step -
accuracy: 0.1933 - loss: 0.3741 - val_accuracy: 0.1918 - val_loss: 0.1335
Epoch 182/500
138/138              0s 934us/step -
accuracy: 0.1729 - loss: 0.3724 - val_accuracy: 0.1918 - val_loss: 0.1333
Epoch 183/500
138/138              0s 926us/step -
accuracy: 0.1758 - loss: 0.3753 - val_accuracy: 0.1918 - val_loss: 0.1295
Epoch 184/500
138/138              0s 928us/step -
accuracy: 0.1812 - loss: 0.3711 - val_accuracy: 0.1918 - val_loss: 0.1338
Epoch 185/500
138/138              0s 1ms/step -
accuracy: 0.1824 - loss: 0.3723 - val_accuracy: 0.1918 - val_loss: 0.1199
Epoch 186/500
138/138              0s 940us/step -
accuracy: 0.1789 - loss: 0.3768 - val_accuracy: 0.1918 - val_loss: 0.1334
Epoch 187/500
138/138              0s 944us/step -
accuracy: 0.1863 - loss: 0.3739 - val_accuracy: 0.1918 - val_loss: 0.1264
Epoch 188/500
138/138              0s 908us/step -
accuracy: 0.1716 - loss: 0.3709 - val_accuracy: 0.1918 - val_loss: 0.1313
Epoch 189/500
138/138              0s 934us/step -
accuracy: 0.1785 - loss: 0.3730 - val_accuracy: 0.1918 - val_loss: 0.1298
Epoch 190/500
138/138              0s 955us/step -
accuracy: 0.1831 - loss: 0.3689 - val_accuracy: 0.1918 - val_loss: 0.1276
Epoch 191/500
138/138              0s 935us/step -
accuracy: 0.1794 - loss: 0.3667 - val_accuracy: 0.1918 - val_loss: 0.1270
Epoch 192/500
138/138              0s 921us/step -
accuracy: 0.1765 - loss: 0.3656 - val_accuracy: 0.1918 - val_loss: 0.1307
Epoch 193/500
```

```
138/138              0s 1ms/step -
accuracy: 0.1771 - loss: 0.3705 - val_accuracy: 0.1918 - val_loss: 0.1219
Epoch 194/500
138/138              0s 956us/step -
accuracy: 0.1789 - loss: 0.3740 - val_accuracy: 0.1918 - val_loss: 0.1325
Epoch 195/500
138/138              0s 931us/step -
accuracy: 0.1899 - loss: 0.3724 - val_accuracy: 0.1918 - val_loss: 0.1316
Epoch 196/500
138/138              0s 1ms/step -
accuracy: 0.1728 - loss: 0.3677 - val_accuracy: 0.1918 - val_loss: 0.1331
Epoch 197/500
138/138              0s 916us/step -
accuracy: 0.1707 - loss: 0.3729 - val_accuracy: 0.1918 - val_loss: 0.1390
Epoch 198/500
138/138              0s 932us/step -
accuracy: 0.1948 - loss: 0.3637 - val_accuracy: 0.1918 - val_loss: 0.1396
Epoch 199/500
138/138              0s 920us/step -
accuracy: 0.1821 - loss: 0.3668 - val_accuracy: 0.1939 - val_loss: 0.1300
Epoch 200/500
138/138              0s 1ms/step -
accuracy: 0.1869 - loss: 0.3601 - val_accuracy: 0.2020 - val_loss: 0.1193
Epoch 201/500
138/138              0s 942us/step -
accuracy: 0.1823 - loss: 0.3715 - val_accuracy: 0.1918 - val_loss: 0.1361
Epoch 202/500
138/138              0s 955us/step -
accuracy: 0.1853 - loss: 0.3711 - val_accuracy: 0.1939 - val_loss: 0.1288
Epoch 203/500
138/138              0s 924us/step -
accuracy: 0.1914 - loss: 0.3673 - val_accuracy: 0.2000 - val_loss: 0.1250
Epoch 204/500
138/138              0s 950us/step -
accuracy: 0.1879 - loss: 0.3695 - val_accuracy: 0.1939 - val_loss: 0.1364
Epoch 205/500
138/138              0s 917us/step -
accuracy: 0.1888 - loss: 0.3650 - val_accuracy: 0.1939 - val_loss: 0.1391
Epoch 206/500
138/138              0s 948us/step -
accuracy: 0.1768 - loss: 0.3682 - val_accuracy: 0.2000 - val_loss: 0.1300
Epoch 207/500
138/138              0s 931us/step -
accuracy: 0.1844 - loss: 0.3701 - val_accuracy: 0.1939 - val_loss: 0.1374
Epoch 208/500
138/138              0s 942us/step -
accuracy: 0.1870 - loss: 0.3695 - val_accuracy: 0.2000 - val_loss: 0.1282
Epoch 209/500
```

```
138/138              0s 952us/step -
accuracy: 0.1767 - loss: 0.3734 - val_accuracy: 0.2041 - val_loss: 0.1247
Epoch 210/500
138/138              0s 920us/step -
accuracy: 0.1964 - loss: 0.3744 - val_accuracy: 0.1939 - val_loss: 0.1355
Epoch 211/500
138/138              0s 975us/step -
accuracy: 0.1943 - loss: 0.3670 - val_accuracy: 0.2245 - val_loss: 0.1109
Epoch 212/500
138/138              0s 960us/step -
accuracy: 0.2032 - loss: 0.3637 - val_accuracy: 0.2041 - val_loss: 0.1211
Epoch 213/500
138/138              0s 896us/step -
accuracy: 0.1908 - loss: 0.3631 - val_accuracy: 0.2020 - val_loss: 0.1254
Epoch 214/500
138/138              0s 922us/step -
accuracy: 0.1859 - loss: 0.3727 - val_accuracy: 0.2000 - val_loss: 0.1348
Epoch 215/500
138/138              0s 932us/step -
accuracy: 0.1895 - loss: 0.3610 - val_accuracy: 0.1939 - val_loss: 0.1428
Epoch 216/500
138/138              0s 930us/step -
accuracy: 0.1921 - loss: 0.3696 - val_accuracy: 0.2020 - val_loss: 0.1274
Epoch 217/500
138/138              0s 909us/step -
accuracy: 0.2038 - loss: 0.3623 - val_accuracy: 0.1939 - val_loss: 0.1407
Epoch 218/500
138/138              0s 904us/step -
accuracy: 0.2019 - loss: 0.3599 - val_accuracy: 0.2143 - val_loss: 0.1170
Epoch 219/500
138/138              0s 980us/step -
accuracy: 0.1879 - loss: 0.3698 - val_accuracy: 0.2000 - val_loss: 0.1328
Epoch 220/500
138/138              0s 933us/step -
accuracy: 0.1913 - loss: 0.3628 - val_accuracy: 0.2000 - val_loss: 0.1320
Epoch 221/500
138/138              0s 1ms/step -
accuracy: 0.1903 - loss: 0.3672 - val_accuracy: 0.2000 - val_loss: 0.1343
Epoch 222/500
138/138              0s 946us/step -
accuracy: 0.1830 - loss: 0.3779 - val_accuracy: 0.2082 - val_loss: 0.1214
Epoch 223/500
138/138              0s 938us/step -
accuracy: 0.1979 - loss: 0.3652 - val_accuracy: 0.2000 - val_loss: 0.1361
Epoch 224/500
138/138              0s 905us/step -
accuracy: 0.1862 - loss: 0.3697 - val_accuracy: 0.2020 - val_loss: 0.1276
Epoch 225/500
```

```
138/138              0s 937us/step -
accuracy: 0.2017 - loss: 0.3557 - val_accuracy: 0.2000 - val_loss: 0.1320
Epoch 226/500
138/138              0s 907us/step -
accuracy: 0.1945 - loss: 0.3597 - val_accuracy: 0.1939 - val_loss: 0.1427
Epoch 227/500
138/138              0s 928us/step -
accuracy: 0.1940 - loss: 0.3585 - val_accuracy: 0.2020 - val_loss: 0.1296
Epoch 228/500
138/138              0s 977us/step -
accuracy: 0.1830 - loss: 0.3573 - val_accuracy: 0.2041 - val_loss: 0.1263
Epoch 229/500
138/138              0s 1ms/step -
accuracy: 0.1912 - loss: 0.3637 - val_accuracy: 0.2082 - val_loss: 0.1222
Epoch 230/500
138/138              0s 902us/step -
accuracy: 0.1906 - loss: 0.3579 - val_accuracy: 0.2061 - val_loss: 0.1237
Epoch 231/500
138/138              0s 999us/step -
accuracy: 0.2049 - loss: 0.3578 - val_accuracy: 0.2000 - val_loss: 0.1341
Epoch 232/500
138/138              0s 922us/step -
accuracy: 0.2089 - loss: 0.3565 - val_accuracy: 0.2000 - val_loss: 0.1388
Epoch 233/500
138/138              0s 974us/step -
accuracy: 0.1975 - loss: 0.3631 - val_accuracy: 0.2041 - val_loss: 0.1272
Epoch 234/500
138/138              0s 908us/step -
accuracy: 0.2016 - loss: 0.3638 - val_accuracy: 0.2000 - val_loss: 0.1339
Epoch 235/500
138/138              0s 1ms/step -
accuracy: 0.2041 - loss: 0.3568 - val_accuracy: 0.2061 - val_loss: 0.1251
Epoch 236/500
138/138              0s 939us/step -
accuracy: 0.1949 - loss: 0.3635 - val_accuracy: 0.2061 - val_loss: 0.1233
Epoch 237/500
138/138              0s 915us/step -
accuracy: 0.1962 - loss: 0.3635 - val_accuracy: 0.2061 - val_loss: 0.1249
Epoch 238/500
138/138              0s 937us/step -
accuracy: 0.2010 - loss: 0.3659 - val_accuracy: 0.2020 - val_loss: 0.1313
Epoch 239/500
138/138              0s 996us/step -
accuracy: 0.1953 - loss: 0.3654 - val_accuracy: 0.2020 - val_loss: 0.1326
Epoch 240/500
138/138              0s 927us/step -
accuracy: 0.1910 - loss: 0.3680 - val_accuracy: 0.2000 - val_loss: 0.1382
Epoch 241/500
```

```
138/138          0s 917us/step -
accuracy: 0.1929 - loss: 0.3613 - val_accuracy: 0.2061 - val_loss: 0.1244
Epoch 242/500
138/138          0s 919us/step -
accuracy: 0.1970 - loss: 0.3675 - val_accuracy: 0.2000 - val_loss: 0.1359
Epoch 243/500
138/138          0s 928us/step -
accuracy: 0.1905 - loss: 0.3720 - val_accuracy: 0.2061 - val_loss: 0.1250
Epoch 244/500
138/138          0s 941us/step -
accuracy: 0.1982 - loss: 0.3697 - val_accuracy: 0.2122 - val_loss: 0.1189
Epoch 245/500
138/138          0s 949us/step -
accuracy: 0.2023 - loss: 0.3595 - val_accuracy: 0.2020 - val_loss: 0.1334
Epoch 246/500
138/138          0s 931us/step -
accuracy: 0.1993 - loss: 0.3605 - val_accuracy: 0.2061 - val_loss: 0.1247
Epoch 247/500
138/138          0s 1ms/step -
accuracy: 0.2010 - loss: 0.3608 - val_accuracy: 0.2082 - val_loss: 0.1226
Epoch 248/500
138/138          0s 1ms/step -
accuracy: 0.1973 - loss: 0.3527 - val_accuracy: 0.2082 - val_loss: 0.1208
Epoch 249/500
138/138          0s 908us/step -
accuracy: 0.1917 - loss: 0.3668 - val_accuracy: 0.2041 - val_loss: 0.1284
Epoch 250/500
138/138          0s 931us/step -
accuracy: 0.2013 - loss: 0.3572 - val_accuracy: 0.2143 - val_loss: 0.1184
Epoch 251/500
138/138          0s 917us/step -
accuracy: 0.2069 - loss: 0.3689 - val_accuracy: 0.2061 - val_loss: 0.1232
Epoch 252/500
138/138          0s 926us/step -
accuracy: 0.1967 - loss: 0.3600 - val_accuracy: 0.1939 - val_loss: 0.1430
Epoch 253/500
138/138          0s 925us/step -
accuracy: 0.1922 - loss: 0.3674 - val_accuracy: 0.2041 - val_loss: 0.1315
Epoch 254/500
138/138          0s 933us/step -
accuracy: 0.1932 - loss: 0.3681 - val_accuracy: 0.2020 - val_loss: 0.1333
Epoch 255/500
138/138          0s 926us/step -
accuracy: 0.1951 - loss: 0.3559 - val_accuracy: 0.1939 - val_loss: 0.1427
Epoch 256/500
138/138          0s 927us/step -
accuracy: 0.2107 - loss: 0.3700 - val_accuracy: 0.2020 - val_loss: 0.1267
Epoch 257/500
```

```
138/138              0s 978us/step -
accuracy: 0.1906 - loss: 0.3636 - val_accuracy: 0.2020 - val_loss: 0.1322
Epoch 258/500
138/138              0s 1ms/step -
accuracy: 0.1943 - loss: 0.3598 - val_accuracy: 0.2082 - val_loss: 0.1201
Epoch 259/500
138/138              0s 952us/step -
accuracy: 0.1926 - loss: 0.3776 - val_accuracy: 0.2082 - val_loss: 0.1214
Epoch 260/500
138/138              0s 929us/step -
accuracy: 0.2012 - loss: 0.3610 - val_accuracy: 0.2000 - val_loss: 0.1359
Epoch 261/500
138/138              0s 914us/step -
accuracy: 0.1869 - loss: 0.3735 - val_accuracy: 0.2041 - val_loss: 0.1285
Epoch 262/500
138/138              0s 942us/step -
accuracy: 0.1988 - loss: 0.3529 - val_accuracy: 0.2041 - val_loss: 0.1260
Epoch 263/500
138/138              0s 937us/step -
accuracy: 0.1979 - loss: 0.3585 - val_accuracy: 0.2020 - val_loss: 0.1322
Epoch 264/500
138/138              0s 952us/step -
accuracy: 0.2001 - loss: 0.3600 - val_accuracy: 0.2000 - val_loss: 0.1360
Epoch 265/500
138/138              0s 922us/step -
accuracy: 0.2023 - loss: 0.3541 - val_accuracy: 0.2082 - val_loss: 0.1216
Epoch 266/500
138/138              0s 969us/step -
accuracy: 0.1938 - loss: 0.3740 - val_accuracy: 0.2082 - val_loss: 0.1216
Epoch 267/500
138/138              0s 1ms/step -
accuracy: 0.1971 - loss: 0.3540 - val_accuracy: 0.2020 - val_loss: 0.1265
Epoch 268/500
138/138              0s 913us/step -
accuracy: 0.2075 - loss: 0.3474 - val_accuracy: 0.2082 - val_loss: 0.1233
Epoch 269/500
138/138              0s 940us/step -
accuracy: 0.2032 - loss: 0.3705 - val_accuracy: 0.2000 - val_loss: 0.1358
Epoch 270/500
138/138              0s 2ms/step -
accuracy: 0.1946 - loss: 0.3540 - val_accuracy: 0.2061 - val_loss: 0.1234
Epoch 271/500
138/138              0s 1ms/step -
accuracy: 0.1983 - loss: 0.3586 - val_accuracy: 0.2041 - val_loss: 0.1245
Epoch 272/500
138/138              0s 1ms/step -
accuracy: 0.2024 - loss: 0.3561 - val_accuracy: 0.2102 - val_loss: 0.1220
Epoch 273/500
```

```
138/138              0s 1ms/step -
accuracy: 0.1950 - loss: 0.3559 - val_accuracy: 0.2204 - val_loss: 0.1187
Epoch 274/500
138/138              0s 986us/step -
accuracy: 0.2060 - loss: 0.3652 - val_accuracy: 0.2408 - val_loss: 0.1096
Epoch 275/500
138/138              0s 1ms/step -
accuracy: 0.2071 - loss: 0.3554 - val_accuracy: 0.2000 - val_loss: 0.1376
Epoch 276/500
138/138              0s 910us/step -
accuracy: 0.1973 - loss: 0.3635 - val_accuracy: 0.2286 - val_loss: 0.1169
Epoch 277/500
138/138              0s 942us/step -
accuracy: 0.1926 - loss: 0.3747 - val_accuracy: 0.2061 - val_loss: 0.1230
Epoch 278/500
138/138              0s 946us/step -
accuracy: 0.1914 - loss: 0.3633 - val_accuracy: 0.2082 - val_loss: 0.1220
Epoch 279/500
138/138              0s 912us/step -
accuracy: 0.2052 - loss: 0.3473 - val_accuracy: 0.2265 - val_loss: 0.1163
Epoch 280/500
138/138              0s 1ms/step -
accuracy: 0.1961 - loss: 0.3680 - val_accuracy: 0.2041 - val_loss: 0.1293
Epoch 281/500
138/138              0s 921us/step -
accuracy: 0.1960 - loss: 0.3670 - val_accuracy: 0.2041 - val_loss: 0.1246
Epoch 282/500
138/138              0s 919us/step -
accuracy: 0.2005 - loss: 0.3643 - val_accuracy: 0.2041 - val_loss: 0.1243
Epoch 283/500
138/138              0s 1ms/step -
accuracy: 0.2011 - loss: 0.3613 - val_accuracy: 0.2041 - val_loss: 0.1253
Epoch 284/500
138/138              0s 912us/step -
accuracy: 0.1938 - loss: 0.3632 - val_accuracy: 0.2061 - val_loss: 0.1234
Epoch 285/500
138/138              0s 1ms/step -
accuracy: 0.2048 - loss: 0.3576 - val_accuracy: 0.2061 - val_loss: 0.1231
Epoch 286/500
138/138              0s 930us/step -
accuracy: 0.2001 - loss: 0.3565 - val_accuracy: 0.2286 - val_loss: 0.1164
Epoch 287/500
138/138              0s 901us/step -
accuracy: 0.2105 - loss: 0.3611 - val_accuracy: 0.2041 - val_loss: 0.1304
Epoch 288/500
138/138              0s 940us/step -
accuracy: 0.1983 - loss: 0.3640 - val_accuracy: 0.2184 - val_loss: 0.1208
Epoch 289/500
```

```
138/138              0s 929us/step -
accuracy: 0.2081 - loss: 0.3607 - val_accuracy: 0.2041 - val_loss: 0.1290
Epoch 290/500
138/138              0s 931us/step -
accuracy: 0.2053 - loss: 0.3588 - val_accuracy: 0.2265 - val_loss: 0.1174
Epoch 291/500
138/138              0s 1ms/step -
accuracy: 0.2043 - loss: 0.3604 - val_accuracy: 0.2327 - val_loss: 0.1160
Epoch 292/500
138/138              0s 1ms/step -
accuracy: 0.2158 - loss: 0.3467 - val_accuracy: 0.2041 - val_loss: 0.1256
Epoch 293/500
138/138              0s 916us/step -
accuracy: 0.1996 - loss: 0.3610 - val_accuracy: 0.2041 - val_loss: 0.1251
Epoch 294/500
138/138              0s 882us/step -
accuracy: 0.2154 - loss: 0.3497 - val_accuracy: 0.2082 - val_loss: 0.1222
Epoch 295/500
138/138              0s 932us/step -
accuracy: 0.2058 - loss: 0.3671 - val_accuracy: 0.2041 - val_loss: 0.1359
Epoch 296/500
138/138              0s 902us/step -
accuracy: 0.2126 - loss: 0.3535 - val_accuracy: 0.2041 - val_loss: 0.1267
Epoch 297/500
138/138              0s 910us/step -
accuracy: 0.2048 - loss: 0.3610 - val_accuracy: 0.2041 - val_loss: 0.1349
Epoch 298/500
138/138              0s 945us/step -
accuracy: 0.2016 - loss: 0.3610 - val_accuracy: 0.2041 - val_loss: 0.1296
Epoch 299/500
138/138              0s 1ms/step -
accuracy: 0.2008 - loss: 0.3570 - val_accuracy: 0.2082 - val_loss: 0.1228
Epoch 300/500
138/138              0s 936us/step -
accuracy: 0.1939 - loss: 0.3696 - val_accuracy: 0.2041 - val_loss: 0.1257
Epoch 301/500
138/138              0s 905us/step -
accuracy: 0.1973 - loss: 0.3608 - val_accuracy: 0.2163 - val_loss: 0.1208
Epoch 302/500
138/138              0s 937us/step -
accuracy: 0.2050 - loss: 0.3582 - val_accuracy: 0.2041 - val_loss: 0.1336
Epoch 303/500
138/138              0s 1ms/step -
accuracy: 0.1989 - loss: 0.3586 - val_accuracy: 0.2041 - val_loss: 0.1362
Epoch 304/500
138/138              0s 936us/step -
accuracy: 0.2038 - loss: 0.3533 - val_accuracy: 0.2020 - val_loss: 0.1272
Epoch 305/500
```

```
138/138           0s 905us/step -
accuracy: 0.2023 - loss: 0.3602 - val_accuracy: 0.2163 - val_loss: 0.1195
Epoch 306/500
138/138           0s 1ms/step -
accuracy: 0.2031 - loss: 0.3591 - val_accuracy: 0.2041 - val_loss: 0.1300
Epoch 307/500
138/138           0s 1ms/step -
accuracy: 0.2010 - loss: 0.3580 - val_accuracy: 0.2041 - val_loss: 0.1368
Epoch 308/500
138/138           0s 949us/step -
accuracy: 0.1963 - loss: 0.3538 - val_accuracy: 0.2041 - val_loss: 0.1303
Epoch 309/500
138/138           0s 940us/step -
accuracy: 0.2144 - loss: 0.3508 - val_accuracy: 0.2041 - val_loss: 0.1367
Epoch 310/500
138/138           0s 943us/step -
accuracy: 0.2023 - loss: 0.3686 - val_accuracy: 0.2041 - val_loss: 0.1253
Epoch 311/500
138/138           0s 926us/step -
accuracy: 0.1974 - loss: 0.3656 - val_accuracy: 0.2041 - val_loss: 0.1270
Epoch 312/500
138/138           0s 908us/step -
accuracy: 0.1989 - loss: 0.3561 - val_accuracy: 0.2041 - val_loss: 0.1271
Epoch 313/500
138/138           0s 1ms/step -
accuracy: 0.2069 - loss: 0.3617 - val_accuracy: 0.2102 - val_loss: 0.1227
Epoch 314/500
138/138           0s 910us/step -
accuracy: 0.2108 - loss: 0.3599 - val_accuracy: 0.2388 - val_loss: 0.1128
Epoch 315/500
138/138           0s 1ms/step -
accuracy: 0.2026 - loss: 0.3589 - val_accuracy: 0.2122 - val_loss: 0.1225
Epoch 316/500
138/138           0s 959us/step -
accuracy: 0.2011 - loss: 0.3667 - val_accuracy: 0.2041 - val_loss: 0.1278
Epoch 317/500
138/138           0s 944us/step -
accuracy: 0.1994 - loss: 0.3491 - val_accuracy: 0.2020 - val_loss: 0.1277
Epoch 318/500
138/138           0s 959us/step -
accuracy: 0.2058 - loss: 0.3537 - val_accuracy: 0.2041 - val_loss: 0.1297
Epoch 319/500
138/138           0s 944us/step -
accuracy: 0.2032 - loss: 0.3552 - val_accuracy: 0.2367 - val_loss: 0.1149
Epoch 320/500
138/138           0s 974us/step -
accuracy: 0.2103 - loss: 0.3555 - val_accuracy: 0.2204 - val_loss: 0.1196
Epoch 321/500
```

```
138/138            0s 968us/step -
accuracy: 0.2060 - loss: 0.3614 - val_accuracy: 0.2082 - val_loss: 0.1235
Epoch 322/500
138/138            0s 1ms/step -
accuracy: 0.1970 - loss: 0.3681 - val_accuracy: 0.2306 - val_loss: 0.1171
Epoch 323/500
138/138            0s 1ms/step -
accuracy: 0.2046 - loss: 0.3648 - val_accuracy: 0.2082 - val_loss: 0.1245
Epoch 324/500
138/138            0s 935us/step -
accuracy: 0.2118 - loss: 0.3546 - val_accuracy: 0.2041 - val_loss: 0.1273
Epoch 325/500
138/138            0s 943us/step -
accuracy: 0.1954 - loss: 0.3663 - val_accuracy: 0.2224 - val_loss: 0.1188
Epoch 326/500
138/138            0s 914us/step -
accuracy: 0.1983 - loss: 0.3617 - val_accuracy: 0.2041 - val_loss: 0.1293
Epoch 327/500
138/138            0s 980us/step -
accuracy: 0.2007 - loss: 0.3528 - val_accuracy: 0.1959 - val_loss: 0.1421
Epoch 328/500
138/138            0s 961us/step -
accuracy: 0.2089 - loss: 0.3515 - val_accuracy: 0.2041 - val_loss: 0.1315
Epoch 329/500
138/138            0s 897us/step -
accuracy: 0.2102 - loss: 0.3584 - val_accuracy: 0.2020 - val_loss: 0.1271
Epoch 330/500
138/138            0s 1ms/step -
accuracy: 0.2103 - loss: 0.3544 - val_accuracy: 0.2061 - val_loss: 0.1257
Epoch 331/500
138/138            0s 921us/step -
accuracy: 0.2020 - loss: 0.3625 - val_accuracy: 0.2347 - val_loss: 0.1155
Epoch 332/500
138/138            0s 953us/step -
accuracy: 0.2164 - loss: 0.3599 - val_accuracy: 0.2041 - val_loss: 0.1269
Epoch 333/500
138/138            0s 1ms/step -
accuracy: 0.2088 - loss: 0.3669 - val_accuracy: 0.2041 - val_loss: 0.1277
Epoch 334/500
138/138            0s 971us/step -
accuracy: 0.2008 - loss: 0.3705 - val_accuracy: 0.2041 - val_loss: 0.1302
Epoch 335/500
138/138            0s 957us/step -
accuracy: 0.2090 - loss: 0.3624 - val_accuracy: 0.2000 - val_loss: 0.1382
Epoch 336/500
138/138            0s 929us/step -
accuracy: 0.2181 - loss: 0.3502 - val_accuracy: 0.2041 - val_loss: 0.1309
Epoch 337/500
```

```
138/138              0s 1ms/step -
accuracy: 0.2043 - loss: 0.3531 - val_accuracy: 0.2020 - val_loss: 0.1385
Epoch 338/500
138/138              0s 940us/step -
accuracy: 0.1871 - loss: 0.3621 - val_accuracy: 0.2041 - val_loss: 0.1281
Epoch 339/500
138/138              0s 947us/step -
accuracy: 0.2037 - loss: 0.3600 - val_accuracy: 0.2061 - val_loss: 0.1257
Epoch 340/500
138/138              0s 912us/step -
accuracy: 0.2033 - loss: 0.3707 - val_accuracy: 0.2184 - val_loss: 0.1228
Epoch 341/500
138/138              0s 953us/step -
accuracy: 0.2187 - loss: 0.3477 - val_accuracy: 0.2408 - val_loss: 0.1147
Epoch 342/500
138/138              0s 948us/step -
accuracy: 0.2067 - loss: 0.3749 - val_accuracy: 0.2265 - val_loss: 0.1193
Epoch 343/500
138/138              0s 1ms/step -
accuracy: 0.2096 - loss: 0.3525 - val_accuracy: 0.2041 - val_loss: 0.1333
Epoch 344/500
138/138              0s 978us/step -
accuracy: 0.2056 - loss: 0.3590 - val_accuracy: 0.2347 - val_loss: 0.1167
Epoch 345/500
138/138              0s 928us/step -
accuracy: 0.2007 - loss: 0.3544 - val_accuracy: 0.2041 - val_loss: 0.1274
Epoch 346/500
138/138              0s 924us/step -
accuracy: 0.1998 - loss: 0.3612 - val_accuracy: 0.2041 - val_loss: 0.1280
Epoch 347/500
138/138              0s 959us/step -
accuracy: 0.2042 - loss: 0.3512 - val_accuracy: 0.2041 - val_loss: 0.1372
Epoch 348/500
138/138              0s 935us/step -
accuracy: 0.2067 - loss: 0.3576 - val_accuracy: 0.2245 - val_loss: 0.1191
Epoch 349/500
138/138              0s 947us/step -
accuracy: 0.2098 - loss: 0.3580 - val_accuracy: 0.2122 - val_loss: 0.1241
Epoch 350/500
138/138              0s 990us/step -
accuracy: 0.2088 - loss: 0.3552 - val_accuracy: 0.2061 - val_loss: 0.1253
Epoch 351/500
138/138              0s 924us/step -
accuracy: 0.2025 - loss: 0.3626 - val_accuracy: 0.2449 - val_loss: 0.1136
Epoch 352/500
138/138              0s 912us/step -
accuracy: 0.2027 - loss: 0.3828 - val_accuracy: 0.2041 - val_loss: 0.1298
Epoch 353/500
```

```
138/138                0s 1ms/step -
accuracy: 0.2025 - loss: 0.3667 - val_accuracy: 0.2204 - val_loss: 0.1210
Epoch 354/500
138/138                0s 994us/step -
accuracy: 0.2034 - loss: 0.3438 - val_accuracy: 0.2204 - val_loss: 0.1205
Epoch 355/500
138/138                0s 927us/step -
accuracy: 0.2177 - loss: 0.3520 - val_accuracy: 0.2102 - val_loss: 0.1254
Epoch 356/500
138/138                0s 932us/step -
accuracy: 0.2028 - loss: 0.3711 - val_accuracy: 0.2184 - val_loss: 0.1234
Epoch 357/500
138/138                0s 992us/step -
accuracy: 0.2103 - loss: 0.3549 - val_accuracy: 0.2061 - val_loss: 0.1267
Epoch 358/500
138/138                0s 940us/step -
accuracy: 0.2109 - loss: 0.3455 - val_accuracy: 0.2061 - val_loss: 0.1367
Epoch 359/500
138/138                0s 916us/step -
accuracy: 0.2076 - loss: 0.3583 - val_accuracy: 0.2061 - val_loss: 0.1266
Epoch 360/500
138/138                0s 930us/step -
accuracy: 0.1975 - loss: 0.3694 - val_accuracy: 0.2347 - val_loss: 0.1169
Epoch 361/500
138/138                0s 917us/step -
accuracy: 0.2062 - loss: 0.3561 - val_accuracy: 0.2408 - val_loss: 0.1149
Epoch 362/500
138/138                0s 1ms/step -
accuracy: 0.2209 - loss: 0.3511 - val_accuracy: 0.2041 - val_loss: 0.1308
Epoch 363/500
138/138                0s 1ms/step -
accuracy: 0.1979 - loss: 0.3635 - val_accuracy: 0.2061 - val_loss: 0.1260
Epoch 364/500
138/138                0s 938us/step -
accuracy: 0.2079 - loss: 0.3563 - val_accuracy: 0.2041 - val_loss: 0.1291
Epoch 365/500
138/138                0s 921us/step -
accuracy: 0.2053 - loss: 0.3543 - val_accuracy: 0.2020 - val_loss: 0.1380
Epoch 366/500
138/138                0s 924us/step -
accuracy: 0.1972 - loss: 0.3626 - val_accuracy: 0.2286 - val_loss: 0.1176
Epoch 367/500
138/138                0s 955us/step -
accuracy: 0.2130 - loss: 0.3659 - val_accuracy: 0.2041 - val_loss: 0.1286
Epoch 368/500
138/138                0s 961us/step -
accuracy: 0.2064 - loss: 0.3552 - val_accuracy: 0.2184 - val_loss: 0.1219
Epoch 369/500
```

```
138/138          0s 952us/step -
accuracy: 0.2128 - loss: 0.3549 - val_accuracy: 0.2286 - val_loss: 0.1178
Epoch 370/500
138/138          0s 1ms/step -
accuracy: 0.2140 - loss: 0.3557 - val_accuracy: 0.2061 - val_loss: 0.1272
Epoch 371/500
138/138          0s 1ms/step -
accuracy: 0.2024 - loss: 0.3548 - val_accuracy: 0.2408 - val_loss: 0.1144
Epoch 372/500
138/138          0s 919us/step -
accuracy: 0.2141 - loss: 0.3571 - val_accuracy: 0.2041 - val_loss: 0.1310
Epoch 373/500
138/138          0s 923us/step -
accuracy: 0.1933 - loss: 0.3556 - val_accuracy: 0.2163 - val_loss: 0.1238
Epoch 374/500
138/138          0s 956us/step -
accuracy: 0.2088 - loss: 0.3497 - val_accuracy: 0.2041 - val_loss: 0.1301
Epoch 375/500
138/138          0s 916us/step -
accuracy: 0.2009 - loss: 0.3670 - val_accuracy: 0.2061 - val_loss: 0.1258
Epoch 376/500
138/138          0s 1ms/step -
accuracy: 0.2144 - loss: 0.3482 - val_accuracy: 0.2204 - val_loss: 0.1206
Epoch 377/500
138/138          0s 937us/step -
accuracy: 0.2176 - loss: 0.3609 - val_accuracy: 0.2000 - val_loss: 0.1396
Epoch 378/500
138/138          0s 923us/step -
accuracy: 0.2086 - loss: 0.3504 - val_accuracy: 0.2041 - val_loss: 0.1321
Epoch 379/500
138/138          0s 921us/step -
accuracy: 0.1952 - loss: 0.3627 - val_accuracy: 0.2041 - val_loss: 0.1279
Epoch 380/500
138/138          0s 1ms/step -
accuracy: 0.2018 - loss: 0.3551 - val_accuracy: 0.2061 - val_loss: 0.1268
Epoch 381/500
138/138          0s 893us/step -
accuracy: 0.2158 - loss: 0.3461 - val_accuracy: 0.2041 - val_loss: 0.1365
Epoch 382/500
138/138          0s 1ms/step -
accuracy: 0.2052 - loss: 0.3504 - val_accuracy: 0.2286 - val_loss: 0.1185
Epoch 383/500
138/138          0s 958us/step -
accuracy: 0.2112 - loss: 0.3553 - val_accuracy: 0.2184 - val_loss: 0.1220
Epoch 384/500
138/138          0s 932us/step -
accuracy: 0.2041 - loss: 0.3548 - val_accuracy: 0.2041 - val_loss: 0.1324
Epoch 385/500
```

27

```
138/138              0s 972us/step -
accuracy: 0.2063 - loss: 0.3614 - val_accuracy: 0.2143 - val_loss: 0.1247
Epoch 386/500
138/138              0s 942us/step -
accuracy: 0.2004 - loss: 0.3549 - val_accuracy: 0.2061 - val_loss: 0.1266
Epoch 387/500
138/138              0s 956us/step -
accuracy: 0.2015 - loss: 0.3622 - val_accuracy: 0.2184 - val_loss: 0.1224
Epoch 388/500
138/138              0s 905us/step -
accuracy: 0.2060 - loss: 0.3489 - val_accuracy: 0.2061 - val_loss: 0.1263
Epoch 389/500
138/138              0s 1ms/step -
accuracy: 0.2072 - loss: 0.3596 - val_accuracy: 0.2041 - val_loss: 0.1274
Epoch 390/500
138/138              0s 964us/step -
accuracy: 0.2040 - loss: 0.3617 - val_accuracy: 0.2122 - val_loss: 0.1250
Epoch 391/500
138/138              0s 916us/step -
accuracy: 0.2124 - loss: 0.3606 - val_accuracy: 0.2204 - val_loss: 0.1211
Epoch 392/500
138/138              0s 970us/step -
accuracy: 0.2116 - loss: 0.3585 - val_accuracy: 0.2061 - val_loss: 0.1277
Epoch 393/500
138/138              0s 960us/step -
accuracy: 0.2092 - loss: 0.3553 - val_accuracy: 0.2020 - val_loss: 0.1363
Epoch 394/500
138/138              0s 945us/step -
accuracy: 0.2103 - loss: 0.3563 - val_accuracy: 0.2041 - val_loss: 0.1318
Epoch 395/500
138/138              0s 1ms/step -
accuracy: 0.2037 - loss: 0.3640 - val_accuracy: 0.2041 - val_loss: 0.1284
Epoch 396/500
138/138              0s 931us/step -
accuracy: 0.2154 - loss: 0.3584 - val_accuracy: 0.2041 - val_loss: 0.1340
Epoch 397/500
138/138              0s 925us/step -
accuracy: 0.2039 - loss: 0.3652 - val_accuracy: 0.2041 - val_loss: 0.1282
Epoch 398/500
138/138              0s 1ms/step -
accuracy: 0.2036 - loss: 0.3677 - val_accuracy: 0.2286 - val_loss: 0.1187
Epoch 399/500
138/138              0s 948us/step -
accuracy: 0.2082 - loss: 0.3532 - val_accuracy: 0.2163 - val_loss: 0.1241
Epoch 400/500
138/138              0s 961us/step -
accuracy: 0.2023 - loss: 0.3763 - val_accuracy: 0.2061 - val_loss: 0.1268
Epoch 401/500
```

```
138/138              0s 1ms/step -
accuracy: 0.2087 - loss: 0.3550 - val_accuracy: 0.2143 - val_loss: 0.1251
Epoch 402/500
138/138              0s 938us/step -
accuracy: 0.2113 - loss: 0.3539 - val_accuracy: 0.2041 - val_loss: 0.1281
Epoch 403/500
138/138              0s 925us/step -
accuracy: 0.2089 - loss: 0.3639 - val_accuracy: 0.2061 - val_loss: 0.1267
Epoch 404/500
138/138              0s 946us/step -
accuracy: 0.2032 - loss: 0.3593 - val_accuracy: 0.2163 - val_loss: 0.1240
Epoch 405/500
138/138              0s 928us/step -
accuracy: 0.2155 - loss: 0.3540 - val_accuracy: 0.2184 - val_loss: 0.1238
Epoch 406/500
138/138              0s 952us/step -
accuracy: 0.2117 - loss: 0.3589 - val_accuracy: 0.2204 - val_loss: 0.1218
Epoch 407/500
138/138              0s 2ms/step -
accuracy: 0.2118 - loss: 0.3563 - val_accuracy: 0.2061 - val_loss: 0.1285
Epoch 408/500
138/138              0s 918us/step -
accuracy: 0.2174 - loss: 0.3573 - val_accuracy: 0.2041 - val_loss: 0.1333
Epoch 409/500
138/138              0s 946us/step -
accuracy: 0.2083 - loss: 0.3540 - val_accuracy: 0.2204 - val_loss: 0.1236
Epoch 410/500
138/138              0s 936us/step -
accuracy: 0.2102 - loss: 0.3592 - val_accuracy: 0.2204 - val_loss: 0.1235
Epoch 411/500
138/138              0s 939us/step -
accuracy: 0.2163 - loss: 0.3512 - val_accuracy: 0.2224 - val_loss: 0.1213
Epoch 412/500
138/138              0s 1ms/step -
accuracy: 0.1974 - loss: 0.3645 - val_accuracy: 0.2143 - val_loss: 0.1241
Epoch 413/500
138/138              0s 935us/step -
accuracy: 0.2125 - loss: 0.3538 - val_accuracy: 0.2204 - val_loss: 0.1238
Epoch 414/500
138/138              0s 921us/step -
accuracy: 0.2218 - loss: 0.3580 - val_accuracy: 0.2306 - val_loss: 0.1183
Epoch 415/500
138/138              0s 910us/step -
accuracy: 0.2176 - loss: 0.3513 - val_accuracy: 0.2143 - val_loss: 0.1244
Epoch 416/500
138/138              0s 910us/step -
accuracy: 0.2163 - loss: 0.3485 - val_accuracy: 0.2102 - val_loss: 0.1267
Epoch 417/500
```

```
138/138              0s 912us/step -
accuracy: 0.2053 - loss: 0.3738 - val_accuracy: 0.2163 - val_loss: 0.1258
Epoch 418/500
138/138              0s 1ms/step -
accuracy: 0.2090 - loss: 0.3683 - val_accuracy: 0.2429 - val_loss: 0.1140
Epoch 419/500
138/138              0s 913us/step -
accuracy: 0.2053 - loss: 0.3732 - val_accuracy: 0.2306 - val_loss: 0.1195
Epoch 420/500
138/138              0s 922us/step -
accuracy: 0.2076 - loss: 0.3666 - val_accuracy: 0.2204 - val_loss: 0.1224
Epoch 421/500
138/138              0s 918us/step -
accuracy: 0.2085 - loss: 0.3631 - val_accuracy: 0.2347 - val_loss: 0.1186
Epoch 422/500
138/138              0s 915us/step -
accuracy: 0.2080 - loss: 0.3585 - val_accuracy: 0.2204 - val_loss: 0.1238
Epoch 423/500
138/138              0s 894us/step -
accuracy: 0.2018 - loss: 0.3624 - val_accuracy: 0.2102 - val_loss: 0.1276
Epoch 424/500
138/138              0s 1ms/step -
accuracy: 0.2141 - loss: 0.3498 - val_accuracy: 0.2020 - val_loss: 0.1359
Epoch 425/500
138/138              0s 907us/step -
accuracy: 0.2202 - loss: 0.3550 - val_accuracy: 0.2041 - val_loss: 0.1310
Epoch 426/500
138/138              0s 973us/step -
accuracy: 0.2121 - loss: 0.3489 - val_accuracy: 0.2245 - val_loss: 0.1213
Epoch 427/500
138/138              0s 1ms/step -
accuracy: 0.2066 - loss: 0.3602 - val_accuracy: 0.2204 - val_loss: 0.1230
Epoch 428/500
138/138              0s 921us/step -
accuracy: 0.2014 - loss: 0.3650 - val_accuracy: 0.2204 - val_loss: 0.1242
Epoch 429/500
138/138              0s 1ms/step -
accuracy: 0.2152 - loss: 0.3502 - val_accuracy: 0.2122 - val_loss: 0.1263
Epoch 430/500
138/138              0s 917us/step -
accuracy: 0.2176 - loss: 0.3599 - val_accuracy: 0.2102 - val_loss: 0.1272
Epoch 431/500
138/138              0s 924us/step -
accuracy: 0.2101 - loss: 0.3531 - val_accuracy: 0.2429 - val_loss: 0.1152
Epoch 432/500
138/138              0s 888us/step -
accuracy: 0.2201 - loss: 0.3481 - val_accuracy: 0.2061 - val_loss: 0.1287
Epoch 433/500
```

```
138/138            0s 940us/step -
accuracy: 0.2093 - loss: 0.3644 - val_accuracy: 0.2327 - val_loss: 0.1194
Epoch 434/500
138/138            0s 930us/step -
accuracy: 0.2158 - loss: 0.3531 - val_accuracy: 0.2429 - val_loss: 0.1162
Epoch 435/500
138/138            0s 1ms/step -
accuracy: 0.2249 - loss: 0.3516 - val_accuracy: 0.2245 - val_loss: 0.1223
Epoch 436/500
138/138            0s 910us/step -
accuracy: 0.2096 - loss: 0.3650 - val_accuracy: 0.2245 - val_loss: 0.1211
Epoch 437/500
138/138            0s 907us/step -
accuracy: 0.2141 - loss: 0.3490 - val_accuracy: 0.2245 - val_loss: 0.1208
Epoch 438/500
138/138            0s 904us/step -
accuracy: 0.2093 - loss: 0.3633 - val_accuracy: 0.2429 - val_loss: 0.1152
Epoch 439/500
138/138            0s 894us/step -
accuracy: 0.2288 - loss: 0.3656 - val_accuracy: 0.2367 - val_loss: 0.1180
Epoch 440/500
138/138            0s 1ms/step -
accuracy: 0.2228 - loss: 0.3488 - val_accuracy: 0.2041 - val_loss: 0.1292
Epoch 441/500
138/138            0s 1ms/step -
accuracy: 0.2151 - loss: 0.3516 - val_accuracy: 0.2347 - val_loss: 0.1194
Epoch 442/500
138/138            0s 955us/step -
accuracy: 0.2111 - loss: 0.3623 - val_accuracy: 0.2184 - val_loss: 0.1243
Epoch 443/500
138/138            0s 912us/step -
accuracy: 0.2104 - loss: 0.3466 - val_accuracy: 0.2184 - val_loss: 0.1240
Epoch 444/500
138/138            0s 972us/step -
accuracy: 0.2143 - loss: 0.3533 - val_accuracy: 0.2245 - val_loss: 0.1215
Epoch 445/500
138/138            0s 1ms/step -
accuracy: 0.2110 - loss: 0.3603 - val_accuracy: 0.2204 - val_loss: 0.1234
Epoch 446/500
138/138            0s 900us/step -
accuracy: 0.2082 - loss: 0.3541 - val_accuracy: 0.2020 - val_loss: 0.1350
Epoch 447/500
138/138            0s 920us/step -
accuracy: 0.2021 - loss: 0.3639 - val_accuracy: 0.2429 - val_loss: 0.1172
Epoch 448/500
138/138            0s 914us/step -
accuracy: 0.2072 - loss: 0.3626 - val_accuracy: 0.2245 - val_loss: 0.1216
Epoch 449/500
```

```
138/138            0s 947us/step -
accuracy: 0.2140 - loss: 0.3517 - val_accuracy: 0.2204 - val_loss: 0.1240
Epoch 450/500
138/138            0s 931us/step -
accuracy: 0.2114 - loss: 0.3513 - val_accuracy: 0.2041 - val_loss: 0.1396
Epoch 451/500
138/138            0s 1ms/step -
accuracy: 0.2123 - loss: 0.3688 - val_accuracy: 0.2245 - val_loss: 0.1218
Epoch 452/500
138/138            0s 928us/step -
accuracy: 0.2252 - loss: 0.3551 - val_accuracy: 0.2041 - val_loss: 0.1321
Epoch 453/500
138/138            0s 1ms/step -
accuracy: 0.2081 - loss: 0.3472 - val_accuracy: 0.2429 - val_loss: 0.1142
Epoch 454/500
138/138            0s 1ms/step -
accuracy: 0.2200 - loss: 0.3493 - val_accuracy: 0.1959 - val_loss: 0.1410
Epoch 455/500
138/138            0s 1ms/step -
accuracy: 0.1920 - loss: 0.3643 - val_accuracy: 0.2469 - val_loss: 0.1139
Epoch 456/500
138/138            0s 1ms/step -
accuracy: 0.2213 - loss: 0.3586 - val_accuracy: 0.2429 - val_loss: 0.1164
Epoch 457/500
138/138            0s 960us/step -
accuracy: 0.2137 - loss: 0.3528 - val_accuracy: 0.2082 - val_loss: 0.1287
Epoch 458/500
138/138            0s 991us/step -
accuracy: 0.2157 - loss: 0.3579 - val_accuracy: 0.2245 - val_loss: 0.1222
Epoch 459/500
138/138            0s 937us/step -
accuracy: 0.2054 - loss: 0.3446 - val_accuracy: 0.2020 - val_loss: 0.1385
Epoch 460/500
138/138            0s 920us/step -
accuracy: 0.2112 - loss: 0.3556 - val_accuracy: 0.2184 - val_loss: 0.1238
Epoch 461/500
138/138            0s 1ms/step -
accuracy: 0.2197 - loss: 0.3484 - val_accuracy: 0.2143 - val_loss: 0.1261
Epoch 462/500
138/138            0s 932us/step -
accuracy: 0.2120 - loss: 0.3529 - val_accuracy: 0.2143 - val_loss: 0.1251
Epoch 463/500
138/138            0s 956us/step -
accuracy: 0.2121 - loss: 0.3540 - val_accuracy: 0.2184 - val_loss: 0.1241
Epoch 464/500
138/138            0s 942us/step -
accuracy: 0.2088 - loss: 0.3540 - val_accuracy: 0.2204 - val_loss: 0.1230
Epoch 465/500
```

```
138/138              0s 923us/step -
accuracy: 0.2042 - loss: 0.3596 - val_accuracy: 0.2102 - val_loss: 0.1280
Epoch 466/500
138/138              0s 1ms/step -
accuracy: 0.2140 - loss: 0.3550 - val_accuracy: 0.2020 - val_loss: 0.1338
Epoch 467/500
138/138              0s 940us/step -
accuracy: 0.2147 - loss: 0.3570 - val_accuracy: 0.2286 - val_loss: 0.1216
Epoch 468/500
138/138              0s 929us/step -
accuracy: 0.2126 - loss: 0.3471 - val_accuracy: 0.2061 - val_loss: 0.1282
Epoch 469/500
138/138              0s 1ms/step -
accuracy: 0.1991 - loss: 0.3659 - val_accuracy: 0.2082 - val_loss: 0.1297
Epoch 470/500
138/138              0s 916us/step -
accuracy: 0.2168 - loss: 0.3448 - val_accuracy: 0.2143 - val_loss: 0.1268
Epoch 471/500
138/138              0s 1ms/step -
accuracy: 0.2046 - loss: 0.3558 - val_accuracy: 0.2020 - val_loss: 0.1334
Epoch 472/500
138/138              0s 940us/step -
accuracy: 0.2054 - loss: 0.3572 - val_accuracy: 0.2020 - val_loss: 0.1317
Epoch 473/500
138/138              0s 942us/step -
accuracy: 0.2131 - loss: 0.3513 - val_accuracy: 0.2082 - val_loss: 0.1292
Epoch 474/500
138/138              0s 958us/step -
accuracy: 0.1985 - loss: 0.3646 - val_accuracy: 0.2184 - val_loss: 0.1243
Epoch 475/500
138/138              0s 943us/step -
accuracy: 0.2165 - loss: 0.3461 - val_accuracy: 0.2143 - val_loss: 0.1267
Epoch 476/500
138/138              0s 948us/step -
accuracy: 0.2152 - loss: 0.3524 - val_accuracy: 0.2061 - val_loss: 0.1311
Epoch 477/500
138/138              0s 1ms/step -
accuracy: 0.2101 - loss: 0.3574 - val_accuracy: 0.2388 - val_loss: 0.1181
Epoch 478/500
138/138              0s 907us/step -
accuracy: 0.2205 - loss: 0.3474 - val_accuracy: 0.2020 - val_loss: 0.1319
Epoch 479/500
138/138              0s 949us/step -
accuracy: 0.2156 - loss: 0.3545 - val_accuracy: 0.2122 - val_loss: 0.1278
Epoch 480/500
138/138              0s 933us/step -
accuracy: 0.2189 - loss: 0.3582 - val_accuracy: 0.2469 - val_loss: 0.1129
Epoch 481/500
```

```
138/138            0s 972us/step -
accuracy: 0.2152 - loss: 0.3517 - val_accuracy: 0.2082 - val_loss: 0.1295
Epoch 482/500
138/138            0s 1ms/step -
accuracy: 0.2134 - loss: 0.3471 - val_accuracy: 0.2265 - val_loss: 0.1200
Epoch 483/500
138/138            0s 904us/step -
accuracy: 0.2179 - loss: 0.3550 - val_accuracy: 0.2265 - val_loss: 0.1213
Epoch 484/500
138/138            0s 928us/step -
accuracy: 0.2131 - loss: 0.3549 - val_accuracy: 0.2122 - val_loss: 0.1275
Epoch 485/500
138/138            0s 914us/step -
accuracy: 0.2155 - loss: 0.3569 - val_accuracy: 0.2143 - val_loss: 0.1264
Epoch 486/500
138/138            0s 1ms/step -
accuracy: 0.2050 - loss: 0.3605 - val_accuracy: 0.2143 - val_loss: 0.1274
Epoch 487/500
138/138            0s 1ms/step -
accuracy: 0.2133 - loss: 0.3547 - val_accuracy: 0.2388 - val_loss: 0.1190
Epoch 488/500
138/138            0s 909us/step -
accuracy: 0.2130 - loss: 0.3528 - val_accuracy: 0.2163 - val_loss: 0.1261
Epoch 489/500
138/138            0s 912us/step -
accuracy: 0.2143 - loss: 0.3484 - val_accuracy: 0.2143 - val_loss: 0.1260
Epoch 490/500
138/138            0s 930us/step -
accuracy: 0.2196 - loss: 0.3544 - val_accuracy: 0.2020 - val_loss: 0.1321
Epoch 491/500
138/138            0s 924us/step -
accuracy: 0.2041 - loss: 0.3647 - val_accuracy: 0.2306 - val_loss: 0.1214
Epoch 492/500
138/138            0s 1ms/step -
accuracy: 0.2079 - loss: 0.3561 - val_accuracy: 0.2367 - val_loss: 0.1206
Epoch 493/500
138/138            0s 919us/step -
accuracy: 0.2134 - loss: 0.3618 - val_accuracy: 0.2245 - val_loss: 0.1232
Epoch 494/500
138/138            0s 1ms/step -
accuracy: 0.2035 - loss: 0.3563 - val_accuracy: 0.2102 - val_loss: 0.1294
Epoch 495/500
138/138            0s 909us/step -
accuracy: 0.2213 - loss: 0.3527 - val_accuracy: 0.2143 - val_loss: 0.1250
Epoch 496/500
138/138            0s 1ms/step -
accuracy: 0.2118 - loss: 0.3537 - val_accuracy: 0.2061 - val_loss: 0.1297
Epoch 497/500
```

```
138/138                0s 946us/step -
accuracy: 0.2118 - loss: 0.3593 - val_accuracy: 0.2102 - val_loss: 0.1290
Epoch 498/500
138/138                0s 1ms/step -
accuracy: 0.2077 - loss: 0.3535 - val_accuracy: 0.2306 - val_loss: 0.1212
Epoch 499/500
138/138                0s 943us/step -
accuracy: 0.2157 - loss: 0.3559 - val_accuracy: 0.2673 - val_loss: 0.1092
Epoch 500/500
138/138                0s 974us/step -
accuracy: 0.2179 - loss: 0.3545 - val_accuracy: 0.2143 - val_loss: 0.1256
16/16                  0s 3ms/step
```


Confusion Matrix
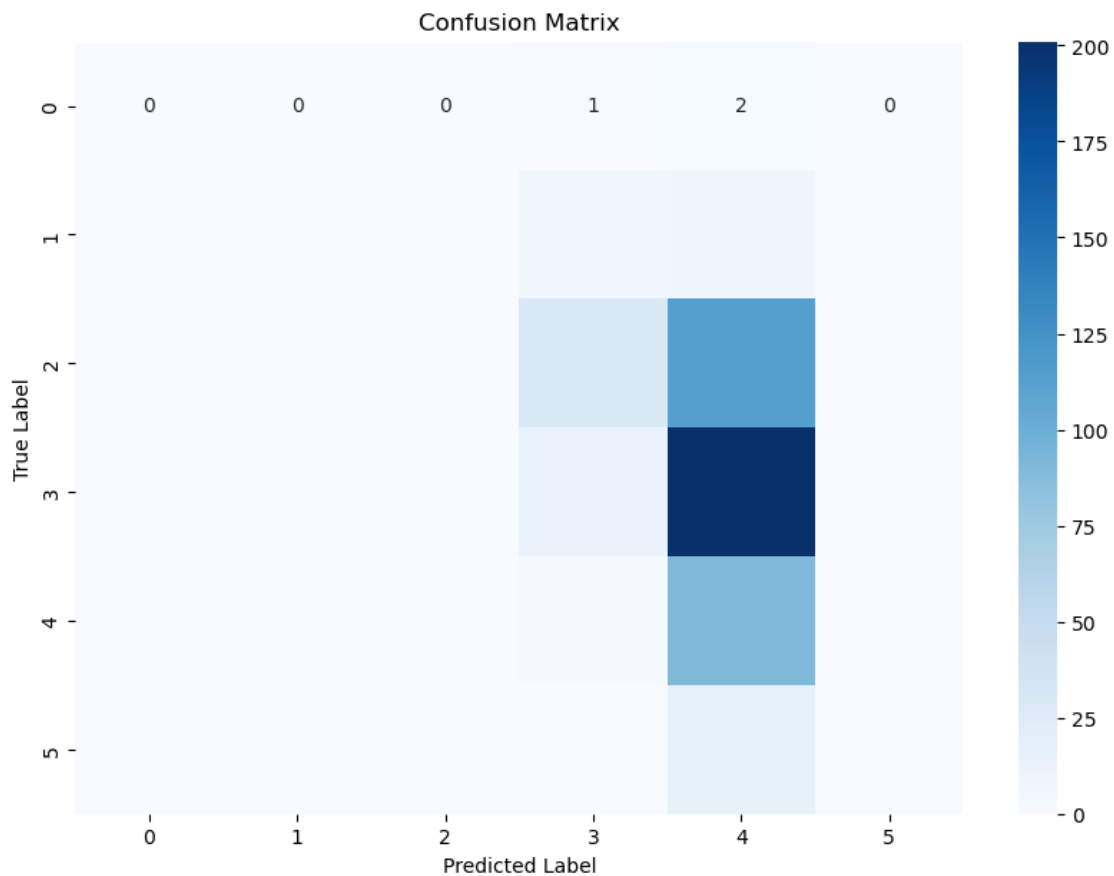
```python
# Exercise 4
import numpy as np
import pandas as pd
import tensorflow as tf
from tensorflow import keras
from sklearn.model_selection import KFold
```

```python
from sklearn.metrics import mean_squared_error

data = pd.read_csv('preprocessed_winequality_data.csv')

X = data.iloc[:, :-10].values.astype(np.float32)

y = data.iloc[:, -10:].values.astype(np.float32)

kf = KFold(n_splits=10, shuffle=True, random_state=42)

mse_scores = []

for fold, (train_index, test_index) in enumerate(kf.split(X)):
    print(f"Training fold {fold+1}...")

    X_train, X_test = X[train_index], X[test_index]
    y_train, y_test = y[train_index], y[test_index]

    model = keras.Sequential([
        keras.layers.Dense(23, activation='sigmoid', input_shape=(11,)),
        keras.layers.Dense(17, activation='sigmoid'),
        keras.layers.Dense(13, activation='sigmoid'),
        keras.layers.Dense(10, activation='sigmoid')
    ])

    model.compile(optimizer=keras.optimizers.SGD(learning_rate=0.4),
                  loss='mean_squared_error',
                  metrics=['accuracy'])

    model.fit(X_train, y_train, epochs=500, batch_size=32, verbose=0)

    y_pred = model.predict(X_test)

    mse = mean_squared_error(y_test, y_pred)
    mse_scores.append(mse)

    print(f"Fold {fold+1} MSE: {mse:.4f}")

average_mse = np.mean(mse_scores)

print("\nFinal Results:")
print("MSE values for each fold:", mse_scores)
print(f"Overall Average MSE: {average_mse:.4f}")
```

Training fold 1…

/opt/anaconda3/lib/python3.11/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.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)

16/16   0s 3ms/step
Fold 1 MSE: 0.0636
Training fold 2…

/opt/anaconda3/lib/python3.11/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.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)

16/16   0s 3ms/step
Fold 2 MSE: 0.0671
Training fold 3…

/opt/anaconda3/lib/python3.11/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.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)

16/16   0s 4ms/step
Fold 3 MSE: 0.0605
Training fold 4…

/opt/anaconda3/lib/python3.11/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.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)

16/16   0s 3ms/step
Fold 4 MSE: 0.0670
Training fold 5…

/opt/anaconda3/lib/python3.11/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.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)

16/16   0s 3ms/step
Fold 5 MSE: 0.0696
Training fold 6…

/opt/anaconda3/lib/python3.11/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.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)

```
16/16               0s 3ms/step
Fold 6 MSE: 0.0606
Training fold 7…
```

/opt/anaconda3/lib/python3.11/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.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)

```
16/16               0s 3ms/step
Fold 7 MSE: 0.0683
Training fold 8…
```

/opt/anaconda3/lib/python3.11/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.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)

```
16/16               0s 3ms/step
Fold 8 MSE: 0.0660
Training fold 9…
```

/opt/anaconda3/lib/python3.11/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.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)

```
16/16               0s 4ms/step
Fold 9 MSE: 0.0607
Training fold 10…
```

/opt/anaconda3/lib/python3.11/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.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)

```
16/16               0s 4ms/step
Fold 10 MSE: 0.0676

Final Results:
MSE values for each fold: [0.06359987, 0.06713602, 0.060517024, 0.06702243,
0.069573365, 0.060612537, 0.06827153, 0.06600944, 0.060737234, 0.06755147]
Overall Average MSE: 0.0651
```

```python
[17]:  # Exercise 5
       # Use random search
       import numpy as np
```

```python
import tensorflow as tf
from tensorflow import keras
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_squared_error
import random

data = pd.read_csv('preprocessed_winequality_data.csv')

X = data.iloc[:, :-10].values.astype(np.float32)

y = data.iloc[:, -10:].values.astype(np.float32)

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.1,␣
 ↪random_state=42)

hidden_layer_sizes = [(20, 15, 10), (25, 18, 12), (30, 20, 15)]
learning_rates = [0.1, 0.2, 0.4, 0.5]
epochs_list = [100, 300, 500]

best_params = None
best_mse = float('inf')

for _ in range(10):
    # Choose parameter randomly
    hidden_layers = random.choice(hidden_layer_sizes)
    learning_rate = random.choice(learning_rates)
    epochs = random.choice(epochs_list)

    print(f"Testing with layers={hidden_layers}, lr={learning_rate},␣
 ↪epochs={epochs}...")

    model = keras.Sequential([
        keras.layers.Dense(hidden_layers[0], activation='sigmoid',␣
 ↪input_shape=(11,)),
        keras.layers.Dense(hidden_layers[1], activation='sigmoid'),
        keras.layers.Dense(hidden_layers[2], activation='sigmoid'),
        keras.layers.Dense(10, activation='sigmoid')
    ])

    model.compile(optimizer=keras.optimizers.SGD(learning_rate=learning_rate),
                  loss='mean_squared_error',
                  metrics=['accuracy'])

    model.fit(X_train, y_train, epochs=epochs, batch_size=32, verbose=0)

    y_pred = model.predict(X_test)
```

```python
    mse = mean_squared_error(y_test, y_pred)

    print(f"MSE: {mse:.4f}")

    # Update the best parameter
    if mse < best_mse:
        best_mse = mse
        best_params = (hidden_layers, learning_rate, epochs)

print("\nBest Hyperparameters:")
print(f"Hidden Layers: {best_params[0]}")
print(f"Learning Rate: {best_params[1]}")
print(f"Epochs: {best_params[2]}")
print(f"Minimum MSE: {best_mse:.4f}")
```

Testing with layers=(20, 15, 10), lr=0.4, epochs=300…

/opt/anaconda3/lib/python3.11/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.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)

16/16   0s 4ms/step
MSE: 0.0682
Testing with layers=(30, 20, 15), lr=0.1, epochs=500…

/opt/anaconda3/lib/python3.11/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.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)

16/16   0s 4ms/step
MSE: 0.0682
Testing with layers=(20, 15, 10), lr=0.2, epochs=300…

/opt/anaconda3/lib/python3.11/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.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)

16/16   0s 3ms/step
MSE: 0.0682
Testing with layers=(20, 15, 10), lr=0.1, epochs=500…

/opt/anaconda3/lib/python3.11/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.

```
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
```

16/16   **0s** 4ms/step
MSE: 0.0682
Testing with layers=(20, 15, 10), lr=0.1, epochs=500…

```
/opt/anaconda3/lib/python3.11/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.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
```

16/16   **0s** 5ms/step
MSE: 0.0682
Testing with layers=(20, 15, 10), lr=0.4, epochs=300…

```
/opt/anaconda3/lib/python3.11/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.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
```

16/16   **0s** 3ms/step
MSE: 0.0683
Testing with layers=(25, 18, 12), lr=0.4, epochs=300…

```
/opt/anaconda3/lib/python3.11/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.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
```

16/16   **0s** 3ms/step
MSE: 0.0682
Testing with layers=(25, 18, 12), lr=0.2, epochs=300…

```
/opt/anaconda3/lib/python3.11/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.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
```

16/16   **0s** 4ms/step
MSE: 0.0682
Testing with layers=(20, 15, 10), lr=0.2, epochs=500…

```
/opt/anaconda3/lib/python3.11/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.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
```

```
16/16              0s 3ms/step
MSE: 0.0682
Testing with layers=(25, 18, 12), lr=0.4, epochs=300…

/opt/anaconda3/lib/python3.11/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.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)

16/16              0s 3ms/step
MSE: 0.0681

Best Hyperparameters:
Hidden Layers: (25, 18, 12)
Learning Rate: 0.4
Epochs: 300
Minimum MSE: 0.0681
```