This python notebook was downloaded from:

nanoHub

Using neural networks to predict and classify crystal structures of elements

Outline:

1. Getting a dataset
2. Processing and Organizing Data
3. Creating the Model
4. Plotting

1. Getting a dataset

Datasets containing properties for the elements in the periodic table are available online; however, it would be thematic to create our own, using the tools from the first tutorial on MSEML Query_Viz. In this section we will query both Pymatgen and Mendeleev to get a complete set of properties per element. We will use this data to create the cases from which the model will train and test.

In this first snippet of code we will import all relevant libraries, the elements that will be turned into cases and the properties that will serve as the attributes for the cases. We will get 47 entries (which is a small dataset), but should give us a somewhat accurate prediction. It is important to note that more entries would move the prediction closer to the real value, and so would more attributes.

The elements listed were chosen because querying them for these properties yields a dataset with no unknown values, and because they represent the three most common crystallographic structures.

In [ ]:
```
import tensorflow as tf
from tensorflow import keras
from keras import initializers
from keras.layers import Dense
```

```
from keras.models import Sequential

import pymatgen.core as pymat
import mendeleev as mendel
import pandas as pd
import numpy as np
import random

%matplotlib inline
import matplotlib.pyplot as plt
import sys

fcc_elements = ["Ag", "Al", "Au", "Cu", "Ir", "Ni", "Pb", "Pd", "Pt", "Rh", "Th", "Yb"]
bcc_elements = ["Ba", "Ca", "Cr", "Cs", "Eu", "Fe", "Li", "Mn", "Mo", "Na", "Nb", "Rb", "Ta", "V", "W" ]
hcp_elements = ["Be", "Cd", "Co", "Dy", "Er", "Gd", "Hf", "Ho", "Lu", "Mg", "Re",
                "Ru", "Sc", "Tb", "Ti", "Tl", "Tm", "Y", "Zn", "Zr"]

elements = fcc_elements + bcc_elements + hcp_elements

random.Random(1).shuffle(elements)

#Note that different properties have been added from the two databases.
querable_mendeleev = ["atomic_number", "atomic_volume", "boiling_point", "en_ghosh",  "evaporation_heat", "heat_of_formation",
                     "lattice_constant", "melting_point", "specific_heat"]
querable_pymatgen = ["atomic_mass", "atomic_radius", "electrical_resistivity","molar_volume", "bulk_modulus", "youngs_modulus"
                     "average_ionic_radius", "density_of_solid", "coefficient_of_linear_thermal_expansion"]
querable_values = querable_mendeleev + querable_pymatgen
```

As before, we will use the database queries to populate lists which can be displayed by the Pandas library in a user-friendly table with the properties as the column headers.

In [ ]:
```
all_values = [] # Values for Attributes
all_labels = [] # Crystal structure labels (0 = fcc, 1 = bcc, 2 = hcp)

for item in elements:
    element_values = []

    # This section queries Mendeleev
    element_object = mendel.element(item)
    for i in querable_mendeleev:
```

```
        element_values.append(getattr(element_object,i))

    # This section queries Pymatgen
    element_object = pymat.Element(item)
    for i in querable_pymatgen:
        element_values.append(getattr(element_object,i))

    all_values.append(element_values) # All lists are appended to another list, creating a List of Lists

    if (item in fcc_elements):
        all_labels.append([1, 0, 0]) # The crystal structure labels are assigned here
    elif (item in bcc_elements):
        all_labels.append([0, 1, 0]) # The crystal structure labels are assigned here
    elif (item in hcp_elements):
        all_labels.append([0, 0, 1]) # The crystal structure labels are assigned here

#print("What is inside element_values")
#print(element_values)
#print("What is inside all_labels")
#print(all_labels)

# Pandas Dataframe
df = pd.DataFrame(all_values, columns=querable_values)

# We will patch some of the values that are not available in the datasets.

# Value for the CTE of Cesium
index_Cs = df.index[df['atomic_number'] == 55]
df.iloc[index_Cs, df.columns.get_loc("coefficient_of_linear_thermal_expansion")] = 0.000097
# Value from: David R. Lide (ed), CRC Handbook of Chemistry and Physics, 84th Edition. CRC Press. Boca Raton, Florida, 2003

# Value for the CTE of Rubidium
index_Rb = df.index[df['atomic_number'] == 37]
df.iloc[index_Rb, df.columns.get_loc("coefficient_of_linear_thermal_expansion")] = 0.000090
# Value from: https://www.azom.com/article.aspx?ArticleID=1834

# Value for the Evaporation Heat of Ruthenium
index_Ru = df.index[df['atomic_number'] == 44]
df.iloc[index_Ru, df.columns.get_loc("evaporation_heat")] = 595 # kJ/mol
# Value from: https://www.webelements.com/ruthenium/thermochemistry.html
```

```python
# Value for the Bulk Modulus of Zirconium
index_Zr = df.index[df['atomic_number'] == 40]
df.iloc[index_Zr, df.columns.get_loc("bulk_modulus")] = 94 # GPa
# Value from: https://materialsproject.org/materials/mp-131/

df.head
```

```
c:\Users\manda\AppData\Local\Programs\Python\Python311\Lib\site-packages\pymatgen\core\periodic_table.py:212: UserWarning:

No data available for coefficient_of_linear_thermal_expansion for Cs

c:\Users\manda\AppData\Local\Programs\Python\Python311\Lib\site-packages\pymatgen\core\periodic_table.py:212: UserWarning:

No data available for coefficient_of_linear_thermal_expansion for Rb

c:\Users\manda\AppData\Local\Programs\Python\Python311\Lib\site-packages\pymatgen\core\periodic_table.py:212: UserWarning:

No data available for bulk_modulus for Zr
```

```
Out[ ]: <bound method NDFrame.head of     atomic_number  atomic_volume  boiling_point  en_ghosh  evaporation_heat  \
        0              27           6.70        3200.15  0.143236             389.1
        1              69          18.10        2223.15  0.216724             232.0
        2              39          19.80        3618.15  0.121699             367.0
        3              75           8.85        5863.15  0.243516             704.0
        4              28           6.60        3186.15  0.147207             378.6
        5              67          18.70        2973.15  0.207795             301.0
        6              79          10.20        3109.15  0.261370             340.0
        7              21          15.00        3109.15  0.119383             332.7
        8              45           8.30        3968.15  0.140838             494.0
        9              74           9.53        5828.15  0.239050             824.0
        10             64          19.90        3546.15  0.194400             398.0
        11             65          19.20        3503.15  0.198863             389.0
        12             72          13.60        4873.15  0.229987             575.0
        13             70          24.80        1469.15  0.221190             159.0
        14             55          70.00         944.15  0.154213              68.3
        15             30           9.20        1180.15  0.155152             114.8
        16             56          39.00        2118.15  0.158679             142.0
        17             25           7.39        2334.15  0.135284             221.0
        18             26           7.10        3134.15  0.139253             340.0
        19             42           9.40        4912.15  0.131267             590.0
        20             11          23.70        1156.09  0.093214              97.9
        21             71          17.80        3675.15  0.225650             414.0
        22             90          19.80        5058.15  0.102770             513.7
        23             29           7.10        2833.15  0.151172             304.6
        24              3          13.10        1615.15  0.105093             148.0
        25             81          17.20        1746.15  0.173447             162.4
        26             23           8.35        3680.15  0.127334             460.0
        27             37          55.90         961.15  0.104686              75.8
        28             40          14.10        4679.15  0.124889             567.0
        29             24           7.23        2944.15  0.131305             342.0
        30             41          10.80        5014.15  0.128078             680.0
        31             47          10.30        2435.15  0.147217             254.1
        32              4           5.00        2741.15  0.144986             309.0
        33             44           8.30        4420.15  0.137649             595.0
        34             13          10.00        2792.15  0.150078             284.1
        35             22          10.60        3560.15  0.123364             422.6
        36             82          18.30        2022.15  0.177911             177.8
        37             20          29.90        1757.15  0.115412             153.6
        38             73          10.90        5728.15  0.234581             758.0
```

39	66	19.00	2840.15	0.203330	291.0
40	48	13.10	1040.15	0.150407	59.1
41	68	18.40	3141.15	0.212261	317.0
42	46	8.90	3236.15	0.144028	372.4
43	63	28.90	1802.15	0.189935	176.0
44	77	8.54	4701.15	0.251060	604.0
45	12	14.00	1363.15	0.121644	131.8
46	78	9.10	4098.15	0.256910	470.0

	heat_of_formation	lattice_constant	melting_point	specific_heat	\
0	426.70	2.51	1768.150	0.421	
1	232.20	3.54	1818.150	0.160	
2	424.70	3.65	1795.150	0.298	
3	774.00	2.76	3458.150	0.137	
4	430.10	3.52	1728.150	0.444	
5	300.60	3.58	1745.150	0.165	
6	368.20	4.08	1337.330	0.129	
7	377.80	3.31	1814.150	0.568	
8	556.00	3.80	2236.150	0.243	
9	851.00	3.16	3687.150	0.132	
10	397.50	3.64	1586.150	0.236	
11	388.70	3.60	1632.150	0.182	
12	618.40	3.20	2506.150	0.144	
13	155.60	5.49	1097.150	0.155	
14	76.50	6.05	301.650	0.242	
15	130.40	2.66	692.677	0.388	
16	179.10	5.02	1000.150	0.204	
17	283.30	8.89	1519.150	0.479	
18	415.50	2.87	1811.150	0.449	
19	658.98	3.15	2895.150	0.251	
20	107.50	4.23	370.944	1.228	
21	427.60	3.51	1936.150	0.154	
22	602.00	5.08	2023.150	0.118	
23	337.40	3.61	1357.770	0.385	
24	159.30	3.49	453.650	3.582	
25	182.20	3.46	577.150	0.129	
26	515.50	3.02	2183.150	0.489	
27	80.90	5.59	312.450	0.363	
28	610.00	3.23	2127.150	0.278	
29	397.48	2.88	2180.150	0.449	
30	733.00	3.30	2750.150	0.265	

31	284.90	4.09	1234.930	0.235
32	324.00	2.29	1560.150	1.825
33	650.60	2.70	2606.150	0.238
34	330.90	4.05	933.473	0.897
35	473.00	2.95	1943.150	0.523
36	195.20	4.95	600.612	0.130
37	177.80	5.58	1115.150	0.647
38	782.00	3.31	3290.150	0.140
39	290.40	3.59	1685.150	0.173
40	111.80	2.98	594.219	0.232
41	316.40	3.56	1802.150	0.168
42	376.60	3.89	1827.950	0.246
43	177.40	4.61	1095.150	0.182
44	669.00	3.84	2719.150	0.131
45	147.10	3.21	923.150	1.023
46	565.70	3.92	2041.350	0.133

	atomic_mass	atomic_radius	electrical_resistivity	molar_volume	\
0	58.933195	1.35	6.000000e-08	6.67	
1	168.934210	1.75	6.760000e-07	19.10	
2	88.905850	1.80	6.000000e-07	19.88	
3	186.207000	1.35	1.800000e-07	8.86	
4	58.693400	1.35	7.200000e-08	6.59	
5	164.930320	1.75	8.140000e-07	18.74	
6	196.966569	1.35	2.200000e-08	10.21	
7	44.955912	1.60	5.500000e-07	15.00	
8	102.905500	1.35	4.300000e-08	8.28	
9	183.840000	1.35	5.400000e-08	9.47	
10	157.250000	1.80	1.310000e-06	19.90	
11	158.925350	1.75	1.150000e-06	19.30	
12	178.490000	1.55	3.400000e-07	13.44	
13	173.040000	1.75	2.500000e-07	24.84	
14	132.905452	2.60	2.100000e-07	70.94	
15	65.409000	1.35	6.000000e-08	9.16	
16	137.327000	2.15	3.400000e-07	38.16	
17	54.938045	1.40	1.440000e-06	7.35	
18	55.845000	1.40	1.000000e-07	7.09	
19	95.940000	1.45	5.500000e-08	9.38	
20	22.989769	1.80	4.900000e-08	23.78	
21	174.967000	1.75	5.800000e-07	17.78	
22	232.038060	1.80	1.500000e-07	19.80	

23	63.546000	1.35	1.720000e-08	7.11
24	6.941000	1.45	9.500000e-08	13.02
25	204.383300	1.90	1.500000e-07	17.22
26	50.941500	1.35	2.000000e-07	8.32
27	85.467800	2.35	1.330000e-07	55.76
28	91.224000	1.55	4.330000e-07	14.02
29	51.996100	1.40	1.270000e-07	7.23
30	92.906380	1.45	1.520000e-07	10.83
31	107.868200	1.60	1.630000e-08	10.27
32	9.012182	1.05	3.800000e-08	4.85
33	101.070000	1.30	7.100000e-08	8.17
34	26.981539	1.25	2.700000e-08	10.00
35	47.867000	1.40	4.000000e-07	10.64
36	207.200000	1.80	2.100000e-07	18.26
37	40.078000	1.80	3.400000e-08	26.20
38	180.947880	1.45	1.350000e-07	10.85
39	162.500000	1.75	9.260000e-07	19.01
40	112.411000	1.55	7.000000e-08	13.00
41	167.259000	1.75	8.600000e-07	18.46
42	106.420000	1.40	1.080000e-07	8.56
43	151.964000	1.85	9.000000e-07	28.97
44	192.217000	1.35	4.700000e-08	8.52
45	24.305000	1.50	4.400000e-08	14.00
46	195.084000	1.35	1.060000e-07	9.09

	bulk_modulus	youngs_modulus	average_ionic_radius	density_of_solid	\
0	180.0	209.0	0.768333	8900.0	
1	45.0	74.0	1.095000	9321.0	
2	41.0	64.0	1.040000	4472.0	
3	370.0	463.0	0.712500	21020.0	
4	180.0	200.0	0.740000	8908.0	
5	40.0	65.0	1.041000	8795.0	
6	220.0	78.0	1.070000	19300.0	
7	57.0	74.0	0.885000	2985.0	
8	380.0	275.0	0.745000	12450.0	
9	310.0	411.0	0.766667	19250.0	
10	38.0	55.0	1.075000	7901.0	
11	38.7	56.0	0.981500	8219.0	
12	110.0	78.0	0.850000	13310.0	
13	31.0	24.0	1.084000	6570.0	
14	1.6	1.7	1.810000	1879.0	

15	70.0	108.0	0.880000	7140.0
16	9.6	13.0	1.490000	3510.0
17	120.0	198.0	0.648333	7470.0
18	170.0	211.0	0.852500	7874.0
19	230.0	329.0	0.775000	10280.0
20	6.3	10.0	1.160000	968.0
21	48.0	69.0	1.001000	9841.0
22	54.0	79.0	1.080000	11724.0
23	140.0	130.0	0.820000	8920.0
24	11.0	4.9	0.900000	535.0
25	43.0	8.0	1.332500	11850.0
26	160.0	128.0	0.777500	6110.0
27	2.5	2.4	1.660000	1532.0
28	94.0	68.0	0.860000	6511.0
29	160.0	279.0	0.940000	7140.0
30	170.0	105.0	0.820000	8570.0
31	100.0	83.0	1.086667	10490.0
32	130.0	287.0	0.590000	1848.0
33	220.0	447.0	0.661000	12370.0
34	76.0	70.0	0.675000	2700.0
35	110.0	116.0	0.851667	4507.0
36	46.0	16.0	1.122500	11340.0
37	17.0	20.0	1.140000	1550.0
38	200.0	186.0	0.820000	16650.0
39	41.0	61.0	1.131000	8551.0
40	42.0	50.0	1.090000	8650.0
41	44.0	70.0	1.030000	9066.0
42	180.0	121.0	0.846250	12023.0
43	8.3	18.0	1.198500	5244.0
44	320.0	528.0	0.765000	22650.0
45	45.0	45.0	0.860000	1738.0
46	230.0	168.0	0.805000	21090.0

	coefficient_of_linear_thermal_expansion
0	0.000013
1	0.000013
2	0.000011
3	0.000006
4	0.000013
5	0.000011
6	0.000014

7	0.000010
8	0.000008
9	0.000005
10	0.000009
11	0.000010
12	0.000006
13	0.000026
14	0.000097
15	0.000030
16	0.000021
17	0.000022
18	0.000012
19	0.000005
20	0.000071
21	0.000010
22	0.000011
23	0.000017
24	0.000046
25	0.000030
26	0.000008
27	0.000090
28	0.000006
29	0.000005
30	0.000007
31	0.000019
32	0.000011
33	0.000006
34	0.000023
35	0.000009
36	0.000029
37	0.000022
38	0.000006
39	0.000010
40	0.000031
41	0.000012
42	0.000012
43	0.000035
44	0.000006
45	0.000008
46	0.000009 >

2. Processing and Organizing Data

We again normalize the data and organize it into training and testing sets as before.

SETS

We have 47 elements for which the crystal structure is known and we will use 40 of these as a training set and the remaining 7 as testing set.

NORMALIZATION

We will again use the Standard Score Normalization, which subtracts the mean of the feature and divide by its standard deviation.

$$\frac{X - \mu}{\sigma}$$

While our model might converge without feature normalization, the resultant model would be difficult to train and would be dependent on the choice of units used in the input.

In [ ]:
```
# SETS

all_values = [list(df.iloc[x]) for x in range(len(all_values))]

# List of lists are turned into Numpy arrays to facilitate calculations in steps to follow (Normalization).
all_values = np.array(all_values, dtype = float)
print("Shape of Values:", all_values.shape)
all_labels = np.array(all_labels, dtype = int)
print("Shape of Labels:", all_labels.shape)

# Training Set
train_values = all_values[:40, :]
train_labels = all_labels[:40, :]

# Testing Set
test_values = all_values[-7:, :]
test_labels = all_labels[-7:, :]

# NORMALIZATION
```

```
mean = np.nanmean(train_values, axis = 0) # mean, np.nanmean is a version of mean function in numpy that throws errors if all

std = np.nanstd(train_values, axis = 0) # standard deviation

train_values = (train_values - mean) / std # input scaling
test_values = (test_values - mean) / std # input scaling

print(train_values[0]) # print a sample entry from the training set
#print(train_labels[0])
```

```
Shape of Values: (47, 18)
Shape of Labels: (47, 3)
[-0.80084167 -0.75983551  0.02340813 -0.40732945  0.15599373  0.16654528
 -1.09549525  0.09167774 -0.03493069 -0.82400017 -0.80570946 -0.67799461
 -0.75661221  0.70972845  0.6516648  -0.77257498  0.11409173 -0.3075323 ]
```

3. Creating the Model

For this classification, we will use a simple sequential neural network with one densely connected hidden layer. We will try many optimizers.

To learn more about Root Mean Squared Propagation, click here.

The key difference between the regression model and the classification model is our metric to measure network performance. While we used mean squared error (between the true outputs and the network's predicted output) for the regression task, we use categorical cross entropy (click here to learn more about it), using classification accuracy as a metric where higher accuracy implies a better network.

In [ ]:

```
from tensorflow.keras import optimizers
# DEFINITION OF THE MODEL

# The weights of our neural network will be initialized in a random manner, using a seed allows for reproducibility
kernel_init = initializers.RandomNormal(seed=14)

model  = Sequential()
model.add(Dense(32, activation='relu', input_shape=(train_values.shape[1],), kernel_initializer=kernel_init))
#model.add(Dense(16, activation='relu', kernel_initializer=kernel_init))
model.add(Dense(3, activation=tf.nn.softmax))  # Output Layer

# DEFINITION OF THE OPTIMIZER
```

```python
#optimizer = optimizers.RMSprop(0.002) # AdaM Optimizer. 0.002 is the learning rate.
#optimizer = optimizers.Adam(0.002) # AdaM Optimizer. 0.002 is the learning rate.
# optimizer = optimizers.SGD(0.002) # AdaM Optimizer. 0.002 is the learning rate.

# This line matches the optimizer to the model and states which metrics will evaluate the model's accuracy
model.compile(loss='categorical_crossentropy', optimizer=optimizer, metrics=['accuracy'])
model.summary()
```

```
Model: "sequential_15"
_________________________________________________________________
 Layer (type)                Output Shape              Param #   
=================================================================
 dense_30 (Dense)            (None, 32)                608       
                                                                 
 dense_31 (Dense)            (None, 3)                 99        
                                                                 
=================================================================
Total params: 707 (2.76 KB)
Trainable params: 707 (2.76 KB)
Non-trainable params: 0 (0.00 Byte)
_________________________________________________________________
```

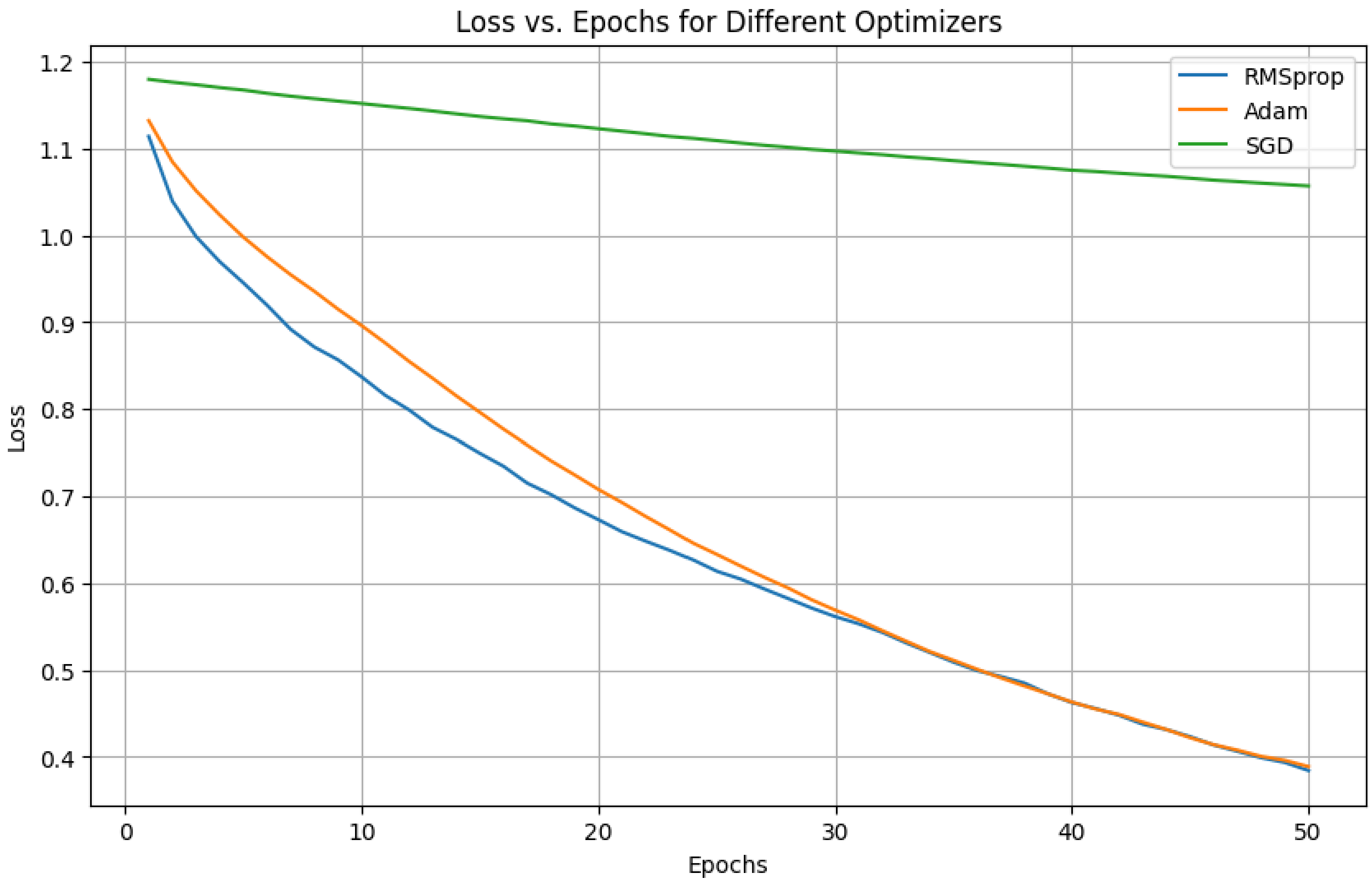

```
_________________________________________________________________
 Layer (type)                Output Shape              Param #   
=================================================================
 dense_30 (Dense)            (None, 32)                608       
                                                                 
 dense_31 (Dense)            (None, 3)                 99        
                                                                 
=================================================================
Total params: 707 (2.76 KB)
Trainable params: 707 (2.76 KB)
Non-trainable params: 0 (0.00 Byte)
_________________________________________________________________
```


In [ ]:

```python
import numpy as np
import matplotlib.pyplot as plt
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense
from tensorflow.keras import initializers
from tensorflow.keras import optimizers
```

```python
# Assuming you have training data in 'train_values' and corresponding labels in 'train_labels'

# Create a Sequential model
kernel_init = initializers.RandomNormal(seed=14)
model = Sequential()
model.add(Dense(32, activation='relu', input_shape=(train_values.shape[1],), kernel_initializer=kernel_init))
model.add(Dense(3, activation='softmax'))  # Output Layer

# Define the optimizers
optimizers_list = {
    'RMSprop': optimizers.RMSprop(0.002),
    'Adam': optimizers.Adam(0.002),
    'SGD': optimizers.SGD(0.002)
}

# Compile the model
model.compile(loss='categorical_crossentropy', optimizer='adam', metrics=['accuracy'])

# Plot loss vs epochs for each optimizer on the same graph
epochs = 50  # Define the number of epochs
plt.figure(figsize=(10, 6))
for optimizer_name, optimizer in optimizers_list.items():
    # Reinitialize the model
    model = Sequential()
    model.add(Dense(32, activation='relu', input_shape=(train_values.shape[1],), kernel_initializer=kernel_init))
    model.add(Dense(3, activation='softmax'))  # Output Layer
    model.compile(loss='categorical_crossentropy', optimizer=optimizer, metrics=['accuracy'])

    # Train the model
    history = model.fit(train_values, train_labels, epochs=epochs, verbose=0)

    # Plot loss vs epochs
    plt.plot(np.arange(1, epochs + 1), history.history['loss'], label=optimizer_name)

    # Reinitialize optimizer
    if optimizer_name != list(optimizers_list.keys())[-1]:
        optimizer = optimizers.get(optimizer_name)
        optimizer.learning_rate.assign(0.002)

plt.title('Loss vs. Epochs for Different Optimizers')
plt.xlabel('Epochs')
```

```
plt.ylabel('Loss')
plt.legend()
plt.grid(True)
plt.show()
```



TRAINING

This model is trained for 500 epochs, and we record the training accuracy in the history object. This way, by plotting "history" we can see the evolution of the "learning" of the model, that is the decrease of the Mean Absolute Error. Models in Keras are fitted to the training set using the **fit** method.

One **Epoch** occurs when you pass the entire dataset through the model. One **Batch** contains a subset of the dataset that can be fed to the model at the same time. A more detailed explanation of these concepts can be found in this blog. As we have a really small dataset compared to the ones that are usually considered to be modeled by these neural networks, we are feeding all entries at the same time, so our batch is the entire dataset, and an epoch occurs when the batch is processed.

In [ ]:
```
class PrintEpNum(keras.callbacks.Callback): # This is a function for the Epoch Counter
    def on_epoch_end(self, epoch, logs):
        sys.stdout.flush()
        sys.stdout.write("Current Epoch: " + str(epoch+1) + '\r') # Updates current Epoch Number


EPOCHS = 500 # Number of EPOCHS

# HISTORY Object which contains how the model learned
history = model.fit(train_values, train_labels, batch_size=train_values.shape[0], \
                    epochs=EPOCHS, validation_split=0.1, verbose = True, callbacks=[PrintEpNum()])

# PLOTTING HISTORY USING MATPLOTLIB

plt.figure()
plt.xlabel('Epoch')
plt.ylabel('Accuracy')
#plt.ylim(0.98,1.05)
plt.plot(history.epoch, np.array(history.history['accuracy']),label='Training Accuracy')
plt.plot(history.epoch, np.array(history.history['val_accuracy']),label = 'Validation Accuracy')
plt.legend()
plt.show()
```

```
Epoch 1/500
1/1 [==============================] - ETA: 0s - loss: 1.0634 - accuracy: 0.5278
```

```
1/1 [==============================] - 0s 234ms/step - loss: 1.0634 - accuracy: 0.5278 - val_loss: 0.9823 - val_accuracy: 0.750
0
Epoch 2/500
1/1 [==============================] - 0s 51ms/step - loss: 1.0625 - accuracy: 0.5278 - val_loss: 0.9817 - val_accuracy: 0.7500
Epoch 3/500
1/1 [==============================] - 0s 42ms/step - loss: 1.0615 - accuracy: 0.5278 - val_loss: 0.9811 - val_accuracy: 0.7500
Epoch 4/500
1/1 [==============================] - 0s 73ms/step - loss: 1.0606 - accuracy: 0.5278 - val_loss: 0.9805 - val_accuracy: 0.7500
Epoch 5/500
1/1 [==============================] - 0s 75ms/step - loss: 1.0597 - accuracy: 0.5278 - val_loss: 0.9800 - val_accuracy: 0.7500
Epoch 6/500
1/1 [==============================] - 0s 72ms/step - loss: 1.0588 - accuracy: 0.5278 - val_loss: 0.9794 - val_accuracy: 0.7500
Epoch 7/500
1/1 [==============================] - 0s 47ms/step - loss: 1.0579 - accuracy: 0.5278 - val_loss: 0.9788 - val_accuracy: 0.7500
Epoch 8/500
1/1 [==============================] - 0s 63ms/step - loss: 1.0570 - accuracy: 0.5278 - val_loss: 0.9782 - val_accuracy: 0.7500
Epoch 9/500
1/1 [==============================] - 0s 63ms/step - loss: 1.0561 - accuracy: 0.5278 - val_loss: 0.9777 - val_accuracy: 0.7500
Epoch 10/500
1/1 [==============================] - 0s 63ms/step - loss: 1.0552 - accuracy: 0.5278 - val_loss: 0.9771 - val_accuracy: 0.7500
Epoch 11/500
1/1 [==============================] - 0s 47ms/step - loss: 1.0543 - accuracy: 0.5278 - val_loss: 0.9765 - val_accuracy: 0.7500
Epoch 12/500
1/1 [==============================] - 0s 63ms/step - loss: 1.0534 - accuracy: 0.5556 - val_loss: 0.9760 - val_accuracy: 0.7500
Epoch 13/500
1/1 [==============================] - 0s 31ms/step - loss: 1.0525 - accuracy: 0.5556 - val_loss: 0.9754 - val_accuracy: 0.7500
Epoch 14/500
1/1 [==============================] - 0s 47ms/step - loss: 1.0517 - accuracy: 0.5556 - val_loss: 0.9748 - val_accuracy: 0.7500
Epoch 15/500
1/1 [==============================] - 0s 62ms/step - loss: 1.0508 - accuracy: 0.5556 - val_loss: 0.9743 - val_accuracy: 1.0000
Epoch 16/500
1/1 [==============================] - 0s 47ms/step - loss: 1.0499 - accuracy: 0.5556 - val_loss: 0.9737 - val_accuracy: 1.0000
Epoch 17/500
1/1 [==============================] - 0s 31ms/step - loss: 1.0490 - accuracy: 0.5556 - val_loss: 0.9731 - val_accuracy: 1.0000
Epoch 18/500
1/1 [==============================] - 0s 47ms/step - loss: 1.0482 - accuracy: 0.5556 - val_loss: 0.9726 - val_accuracy: 1.0000
Epoch 19/500
1/1 [==============================] - 0s 31ms/step - loss: 1.0473 - accuracy: 0.5556 - val_loss: 0.9720 - val_accuracy: 1.0000
Epoch 20/500
1/1 [==============================] - 0s 63ms/step - loss: 1.0464 - accuracy: 0.5556 - val_loss: 0.9714 - val_accuracy: 1.0000
Epoch 21/500
```

```
1/1 [==============================] - 0s 32ms/step - loss: 1.0455 - accuracy: 0.5556 - val_loss: 0.9709 - val_accuracy: 1.0000
Epoch 22/500
1/1 [==============================] - 0s 31ms/step - loss: 1.0447 - accuracy: 0.5556 - val_loss: 0.9703 - val_accuracy: 1.0000
Epoch 23/500
1/1 [==============================] - 0s 47ms/step - loss: 1.0438 - accuracy: 0.5833 - val_loss: 0.9698 - val_accuracy: 1.0000
Epoch 24/500
1/1 [==============================] - 0s 31ms/step - loss: 1.0429 - accuracy: 0.5833 - val_loss: 0.9692 - val_accuracy: 1.0000
Epoch 25/500
1/1 [==============================] - 0s 63ms/step - loss: 1.0421 - accuracy: 0.5833 - val_loss: 0.9687 - val_accuracy: 1.0000
Epoch 26/500
1/1 [==============================] - 0s 47ms/step - loss: 1.0412 - accuracy: 0.5833 - val_loss: 0.9682 - val_accuracy: 1.0000
Epoch 27/500
1/1 [==============================] - 0s 31ms/step - loss: 1.0404 - accuracy: 0.5833 - val_loss: 0.9676 - val_accuracy: 1.0000
Epoch 28/500
1/1 [==============================] - 0s 47ms/step - loss: 1.0396 - accuracy: 0.5833 - val_loss: 0.9671 - val_accuracy: 1.0000
Epoch 29/500
1/1 [==============================] - 0s 47ms/step - loss: 1.0387 - accuracy: 0.5833 - val_loss: 0.9666 - val_accuracy: 1.0000
Epoch 30/500
1/1 [==============================] - 0s 31ms/step - loss: 1.0379 - accuracy: 0.5833 - val_loss: 0.9661 - val_accuracy: 1.0000
Epoch 31/500
1/1 [==============================] - 0s 79ms/step - loss: 1.0371 - accuracy: 0.5833 - val_loss: 0.9655 - val_accuracy: 1.0000
Epoch 32/500
1/1 [==============================] - 0s 47ms/step - loss: 1.0363 - accuracy: 0.5833 - val_loss: 0.9650 - val_accuracy: 1.0000
Epoch 33/500
1/1 [==============================] - 0s 31ms/step - loss: 1.0355 - accuracy: 0.6111 - val_loss: 0.9644 - val_accuracy: 1.0000
Epoch 34/500
1/1 [==============================] - 0s 57ms/step - loss: 1.0347 - accuracy: 0.6111 - val_loss: 0.9638 - val_accuracy: 1.0000
Epoch 35/500
1/1 [==============================] - 0s 47ms/step - loss: 1.0338 - accuracy: 0.6111 - val_loss: 0.9632 - val_accuracy: 1.0000
Epoch 36/500
1/1 [==============================] - 0s 47ms/step - loss: 1.0330 - accuracy: 0.6111 - val_loss: 0.9627 - val_accuracy: 1.0000
Epoch 37/500
1/1 [==============================] - 0s 42ms/step - loss: 1.0322 - accuracy: 0.6111 - val_loss: 0.9621 - val_accuracy: 1.0000
Epoch 38/500
1/1 [==============================] - 0s 32ms/step - loss: 1.0314 - accuracy: 0.6111 - val_loss: 0.9615 - val_accuracy: 1.0000
Epoch 39/500
1/1 [==============================] - 0s 31ms/step - loss: 1.0306 - accuracy: 0.6389 - val_loss: 0.9609 - val_accuracy: 1.0000
Epoch 40/500
1/1 [==============================] - 0s 48ms/step - loss: 1.0299 - accuracy: 0.6389 - val_loss: 0.9603 - val_accuracy: 1.0000
Epoch 41/500
1/1 [==============================] - 0s 47ms/step - loss: 1.0291 - accuracy: 0.6389 - val_loss: 0.9598 - val_accuracy: 1.0000
```

```
Epoch 42/500
1/1 [==============================] - 0s 42ms/step - loss: 1.0283 - accuracy: 0.6389 - val_loss: 0.9593 - val_accuracy: 1.0000
Epoch 43/500
1/1 [==============================] - 0s 32ms/step - loss: 1.0275 - accuracy: 0.6389 - val_loss: 0.9587 - val_accuracy: 1.0000
Epoch 44/500
1/1 [==============================] - 0s 47ms/step - loss: 1.0267 - accuracy: 0.6389 - val_loss: 0.9582 - val_accuracy: 1.0000
Epoch 45/500
1/1 [==============================] - 0s 31ms/step - loss: 1.0259 - accuracy: 0.6389 - val_loss: 0.9577 - val_accuracy: 1.0000
Epoch 46/500
1/1 [==============================] - 0s 47ms/step - loss: 1.0251 - accuracy: 0.6389 - val_loss: 0.9571 - val_accuracy: 1.0000
Epoch 47/500
1/1 [==============================] - 0s 31ms/step - loss: 1.0244 - accuracy: 0.6389 - val_loss: 0.9566 - val_accuracy: 1.0000
Epoch 48/500
1/1 [==============================] - 0s 47ms/step - loss: 1.0236 - accuracy: 0.6389 - val_loss: 0.9560 - val_accuracy: 1.0000
Epoch 49/500
1/1 [==============================] - 0s 32ms/step - loss: 1.0229 - accuracy: 0.6389 - val_loss: 0.9555 - val_accuracy: 1.0000
Epoch 50/500
1/1 [==============================] - 0s 47ms/step - loss: 1.0221 - accuracy: 0.6389 - val_loss: 0.9549 - val_accuracy: 1.0000
Epoch 51/500
1/1 [==============================] - 0s 32ms/step - loss: 1.0214 - accuracy: 0.6389 - val_loss: 0.9544 - val_accuracy: 1.0000
Epoch 52/500
1/1 [==============================] - 0s 31ms/step - loss: 1.0206 - accuracy: 0.6389 - val_loss: 0.9539 - val_accuracy: 1.0000
Epoch 53/500
1/1 [==============================] - 0s 57ms/step - loss: 1.0199 - accuracy: 0.6389 - val_loss: 0.9533 - val_accuracy: 1.0000
Epoch 54/500
1/1 [==============================] - 0s 47ms/step - loss: 1.0192 - accuracy: 0.6389 - val_loss: 0.9528 - val_accuracy: 1.0000
Epoch 55/500
1/1 [==============================] - 0s 47ms/step - loss: 1.0185 - accuracy: 0.6389 - val_loss: 0.9523 - val_accuracy: 1.0000
Epoch 56/500
1/1 [==============================] - 0s 55ms/step - loss: 1.0177 - accuracy: 0.6389 - val_loss: 0.9518 - val_accuracy: 1.0000
Epoch 57/500
1/1 [==============================] - 0s 47ms/step - loss: 1.0170 - accuracy: 0.6389 - val_loss: 0.9512 - val_accuracy: 1.0000
Epoch 58/500
1/1 [==============================] - 0s 62ms/step - loss: 1.0163 - accuracy: 0.6389 - val_loss: 0.9507 - val_accuracy: 1.0000
Epoch 59/500
1/1 [==============================] - 0s 70ms/step - loss: 1.0156 - accuracy: 0.6389 - val_loss: 0.9502 - val_accuracy: 1.0000
Epoch 60/500
1/1 [==============================] - 0s 47ms/step - loss: 1.0148 - accuracy: 0.6389 - val_loss: 0.9497 - val_accuracy: 1.0000
Epoch 61/500
1/1 [==============================] - 0s 47ms/step - loss: 1.0141 - accuracy: 0.6389 - val_loss: 0.9492 - val_accuracy: 1.0000
Epoch 62/500
```

```
1/1 [==============================] - 0s 47ms/step - loss: 1.0134 - accuracy: 0.6389 - val_loss: 0.9487 - val_accuracy: 1.0000
Epoch 63/500
1/1 [==============================] - 0s 72ms/step - loss: 1.0127 - accuracy: 0.6389 - val_loss: 0.9482 - val_accuracy: 1.0000
Epoch 64/500
1/1 [==============================] - 0s 47ms/step - loss: 1.0120 - accuracy: 0.6389 - val_loss: 0.9477 - val_accuracy: 1.0000
Epoch 65/500
1/1 [==============================] - 0s 47ms/step - loss: 1.0113 - accuracy: 0.6389 - val_loss: 0.9472 - val_accuracy: 1.0000
Epoch 66/500
1/1 [==============================] - 0s 40ms/step - loss: 1.0106 - accuracy: 0.6389 - val_loss: 0.9467 - val_accuracy: 1.0000
Epoch 67/500
1/1 [==============================] - 0s 47ms/step - loss: 1.0099 - accuracy: 0.6389 - val_loss: 0.9462 - val_accuracy: 1.0000
Epoch 68/500
1/1 [==============================] - 0s 47ms/step - loss: 1.0092 - accuracy: 0.6389 - val_loss: 0.9457 - val_accuracy: 1.0000
Epoch 69/500
1/1 [==============================] - 0s 62ms/step - loss: 1.0085 - accuracy: 0.6389 - val_loss: 0.9452 - val_accuracy: 1.0000
Epoch 70/500
1/1 [==============================] - 0s 47ms/step - loss: 1.0078 - accuracy: 0.6389 - val_loss: 0.9448 - val_accuracy: 1.0000
Epoch 71/500
1/1 [==============================] - 0s 63ms/step - loss: 1.0071 - accuracy: 0.6389 - val_loss: 0.9443 - val_accuracy: 1.0000
Epoch 72/500
1/1 [==============================] - 0s 79ms/step - loss: 1.0064 - accuracy: 0.6389 - val_loss: 0.9438 - val_accuracy: 1.0000
Epoch 73/500
1/1 [==============================] - 0s 63ms/step - loss: 1.0057 - accuracy: 0.6389 - val_loss: 0.9434 - val_accuracy: 1.0000
Epoch 74/500
1/1 [==============================] - 0s 47ms/step - loss: 1.0051 - accuracy: 0.6389 - val_loss: 0.9429 - val_accuracy: 1.0000
Epoch 75/500
1/1 [==============================] - 0s 31ms/step - loss: 1.0044 - accuracy: 0.6389 - val_loss: 0.9424 - val_accuracy: 1.0000
Epoch 76/500
1/1 [==============================] - 0s 47ms/step - loss: 1.0037 - accuracy: 0.6389 - val_loss: 0.9419 - val_accuracy: 1.0000
Epoch 77/500
1/1 [==============================] - 0s 47ms/step - loss: 1.0030 - accuracy: 0.6389 - val_loss: 0.9415 - val_accuracy: 1.0000
Epoch 78/500
1/1 [==============================] - 0s 31ms/step - loss: 1.0023 - accuracy: 0.6389 - val_loss: 0.9410 - val_accuracy: 1.0000
Epoch 79/500
1/1 [==============================] - 0s 63ms/step - loss: 1.0016 - accuracy: 0.6389 - val_loss: 0.9405 - val_accuracy: 1.0000
Epoch 80/500
1/1 [==============================] - 0s 47ms/step - loss: 1.0010 - accuracy: 0.6389 - val_loss: 0.9400 - val_accuracy: 1.0000
Epoch 81/500
1/1 [==============================] - 0s 31ms/step - loss: 1.0003 - accuracy: 0.6667 - val_loss: 0.9396 - val_accuracy: 0.7500
Epoch 82/500
1/1 [==============================] - 0s 36ms/step - loss: 0.9996 - accuracy: 0.6667 - val_loss: 0.9391 - val_accuracy: 0.7500
```

```
Epoch 83/500
1/1 [==============================] - 0s 27ms/step - loss: 0.9989 - accuracy: 0.6667 - val_loss: 0.9386 - val_accuracy: 0.7500
Epoch 84/500
1/1 [==============================] - 0s 47ms/step - loss: 0.9983 - accuracy: 0.6667 - val_loss: 0.9382 - val_accuracy: 0.7500
Epoch 85/500
1/1 [==============================] - 0s 41ms/step - loss: 0.9976 - accuracy: 0.6667 - val_loss: 0.9377 - val_accuracy: 0.7500
Epoch 86/500
1/1 [==============================] - 0s 78ms/step - loss: 0.9969 - accuracy: 0.6667 - val_loss: 0.9373 - val_accuracy: 0.7500
Epoch 87/500
1/1 [==============================] - 0s 63ms/step - loss: 0.9963 - accuracy: 0.6667 - val_loss: 0.9368 - val_accuracy: 0.7500
Epoch 88/500
1/1 [==============================] - 0s 78ms/step - loss: 0.9956 - accuracy: 0.6667 - val_loss: 0.9363 - val_accuracy: 0.7500
Epoch 89/500
1/1 [==============================] - 0s 63ms/step - loss: 0.9950 - accuracy: 0.6667 - val_loss: 0.9359 - val_accuracy: 0.7500
Epoch 90/500
1/1 [==============================] - 0s 47ms/step - loss: 0.9943 - accuracy: 0.6667 - val_loss: 0.9354 - val_accuracy: 0.7500
Epoch 91/500
1/1 [==============================] - 0s 31ms/step - loss: 0.9937 - accuracy: 0.6667 - val_loss: 0.9349 - val_accuracy: 0.7500
Epoch 92/500
1/1 [==============================] - 0s 47ms/step - loss: 0.9930 - accuracy: 0.6389 - val_loss: 0.9345 - val_accuracy: 0.7500
Epoch 93/500
1/1 [==============================] - 0s 31ms/step - loss: 0.9924 - accuracy: 0.6389 - val_loss: 0.9340 - val_accuracy: 0.7500
Epoch 94/500
1/1 [==============================] - 0s 47ms/step - loss: 0.9918 - accuracy: 0.6389 - val_loss: 0.9336 - val_accuracy: 0.7500
Epoch 95/500
1/1 [==============================] - 0s 31ms/step - loss: 0.9911 - accuracy: 0.6389 - val_loss: 0.9331 - val_accuracy: 0.7500
Epoch 96/500
1/1 [==============================] - 0s 31ms/step - loss: 0.9905 - accuracy: 0.6389 - val_loss: 0.9327 - val_accuracy: 0.7500
Epoch 97/500
1/1 [==============================] - 0s 47ms/step - loss: 0.9898 - accuracy: 0.6389 - val_loss: 0.9322 - val_accuracy: 0.7500
Epoch 98/500
1/1 [==============================] - 0s 47ms/step - loss: 0.9892 - accuracy: 0.6389 - val_loss: 0.9317 - val_accuracy: 0.7500
Epoch 99/500
1/1 [==============================] - 0s 47ms/step - loss: 0.9886 - accuracy: 0.6389 - val_loss: 0.9313 - val_accuracy: 0.7500
Epoch 100/500
1/1 [==============================] - 0s 31ms/step - loss: 0.9879 - accuracy: 0.6389 - val_loss: 0.9308 - val_accuracy: 0.7500
Epoch 101/500
1/1 [==============================] - 0s 35ms/step - loss: 0.9873 - accuracy: 0.6389 - val_loss: 0.9303 - val_accuracy: 0.7500
Epoch 102/500
1/1 [==============================] - 0s 32ms/step - loss: 0.9867 - accuracy: 0.6389 - val_loss: 0.9299 - val_accuracy: 0.7500
Epoch 103/500
```

```
1/1 [==============================] - 0s 31ms/step - loss: 0.9861 - accuracy: 0.6389 - val_loss: 0.9294 - val_accuracy: 0.7500
Epoch 104/500
1/1 [==============================] - 0s 31ms/step - loss: 0.9854 - accuracy: 0.6389 - val_loss: 0.9289 - val_accuracy: 0.7500
Epoch 105/500
1/1 [==============================] - 0s 26ms/step - loss: 0.9848 - accuracy: 0.6389 - val_loss: 0.9285 - val_accuracy: 0.7500
Epoch 106/500
1/1 [==============================] - 0s 47ms/step - loss: 0.9842 - accuracy: 0.6389 - val_loss: 0.9280 - val_accuracy: 0.7500
Epoch 107/500
1/1 [==============================] - 0s 78ms/step - loss: 0.9836 - accuracy: 0.6389 - val_loss: 0.9276 - val_accuracy: 0.7500
Epoch 108/500
1/1 [==============================] - 0s 31ms/step - loss: 0.9830 - accuracy: 0.6389 - val_loss: 0.9271 - val_accuracy: 0.7500
Epoch 109/500
1/1 [==============================] - 0s 44ms/step - loss: 0.9824 - accuracy: 0.6389 - val_loss: 0.9267 - val_accuracy: 0.7500
Epoch 110/500
1/1 [==============================] - 0s 48ms/step - loss: 0.9818 - accuracy: 0.6389 - val_loss: 0.9263 - val_accuracy: 0.7500
Epoch 111/500
1/1 [==============================] - 0s 47ms/step - loss: 0.9812 - accuracy: 0.6389 - val_loss: 0.9258 - val_accuracy: 0.7500
Epoch 112/500
1/1 [==============================] - 0s 47ms/step - loss: 0.9806 - accuracy: 0.6389 - val_loss: 0.9254 - val_accuracy: 0.7500
Epoch 113/500
1/1 [==============================] - 0s 47ms/step - loss: 0.9801 - accuracy: 0.6111 - val_loss: 0.9250 - val_accuracy: 0.7500
Epoch 114/500
1/1 [==============================] - 0s 31ms/step - loss: 0.9795 - accuracy: 0.6111 - val_loss: 0.9245 - val_accuracy: 0.7500
Epoch 115/500
1/1 [==============================] - 0s 47ms/step - loss: 0.9789 - accuracy: 0.6111 - val_loss: 0.9242 - val_accuracy: 0.7500
Epoch 116/500
1/1 [==============================] - 0s 63ms/step - loss: 0.9784 - accuracy: 0.6111 - val_loss: 0.9237 - val_accuracy: 0.7500
Epoch 117/500
1/1 [==============================] - 0s 31ms/step - loss: 0.9778 - accuracy: 0.6111 - val_loss: 0.9233 - val_accuracy: 0.7500
Epoch 118/500
1/1 [==============================] - 0s 47ms/step - loss: 0.9773 - accuracy: 0.6111 - val_loss: 0.9229 - val_accuracy: 0.7500
Epoch 119/500
1/1 [==============================] - 0s 47ms/step - loss: 0.9767 - accuracy: 0.6111 - val_loss: 0.9225 - val_accuracy: 0.7500
Epoch 120/500
1/1 [==============================] - 0s 47ms/step - loss: 0.9761 - accuracy: 0.6111 - val_loss: 0.9221 - val_accuracy: 0.7500
Epoch 121/500
1/1 [==============================] - 0s 31ms/step - loss: 0.9756 - accuracy: 0.6111 - val_loss: 0.9216 - val_accuracy: 0.7500
Epoch 122/500
1/1 [==============================] - 0s 47ms/step - loss: 0.9750 - accuracy: 0.6111 - val_loss: 0.9212 - val_accuracy: 0.7500
Epoch 123/500
1/1 [==============================] - 0s 47ms/step - loss: 0.9745 - accuracy: 0.6111 - val_loss: 0.9208 - val_accuracy: 0.7500
```

```
Epoch 124/500
1/1 [==============================] - 0s 23ms/step - loss: 0.9739 - accuracy: 0.6111 - val_loss: 0.9203 - val_accuracy: 0.7500
Epoch 125/500
1/1 [==============================] - 0s 31ms/step - loss: 0.9734 - accuracy: 0.6111 - val_loss: 0.9200 - val_accuracy: 0.7500
Epoch 126/500
1/1 [==============================] - 0s 31ms/step - loss: 0.9728 - accuracy: 0.6111 - val_loss: 0.9195 - val_accuracy: 0.7500
Epoch 127/500
1/1 [==============================] - 0s 94ms/step - loss: 0.9723 - accuracy: 0.6111 - val_loss: 0.9191 - val_accuracy: 0.7500
Epoch 128/500
1/1 [==============================] - 0s 31ms/step - loss: 0.9717 - accuracy: 0.6111 - val_loss: 0.9186 - val_accuracy: 0.7500
Epoch 129/500
1/1 [==============================] - 0s 46ms/step - loss: 0.9712 - accuracy: 0.6111 - val_loss: 0.9182 - val_accuracy: 0.7500
Epoch 130/500
1/1 [==============================] - 0s 30ms/step - loss: 0.9706 - accuracy: 0.6389 - val_loss: 0.9178 - val_accuracy: 0.7500
Epoch 131/500
1/1 [==============================] - 0s 62ms/step - loss: 0.9701 - accuracy: 0.6667 - val_loss: 0.9174 - val_accuracy: 0.7500
Epoch 132/500
1/1 [==============================] - 0s 62ms/step - loss: 0.9695 - accuracy: 0.6667 - val_loss: 0.9169 - val_accuracy: 0.7500
Epoch 133/500
1/1 [==============================] - 0s 47ms/step - loss: 0.9690 - accuracy: 0.6667 - val_loss: 0.9166 - val_accuracy: 0.7500
Epoch 134/500
1/1 [==============================] - 0s 62ms/step - loss: 0.9684 - accuracy: 0.6667 - val_loss: 0.9162 - val_accuracy: 0.7500
Epoch 135/500
1/1 [==============================] - 0s 63ms/step - loss: 0.9679 - accuracy: 0.6667 - val_loss: 0.9158 - val_accuracy: 0.7500
Epoch 136/500
1/1 [==============================] - 0s 47ms/step - loss: 0.9673 - accuracy: 0.6667 - val_loss: 0.9153 - val_accuracy: 0.7500
Epoch 137/500
1/1 [==============================] - 0s 68ms/step - loss: 0.9668 - accuracy: 0.6667 - val_loss: 0.9149 - val_accuracy: 0.7500
Epoch 138/500
1/1 [==============================] - 0s 51ms/step - loss: 0.9663 - accuracy: 0.6667 - val_loss: 0.9145 - val_accuracy: 0.7500
Epoch 139/500
1/1 [==============================] - 0s 55ms/step - loss: 0.9657 - accuracy: 0.6667 - val_loss: 0.9141 - val_accuracy: 0.7500
Epoch 140/500
1/1 [==============================] - 0s 56ms/step - loss: 0.9652 - accuracy: 0.6667 - val_loss: 0.9137 - val_accuracy: 0.7500
Epoch 141/500
1/1 [==============================] - 0s 58ms/step - loss: 0.9647 - accuracy: 0.6667 - val_loss: 0.9133 - val_accuracy: 0.7500
Epoch 142/500
1/1 [==============================] - 0s 42ms/step - loss: 0.9641 - accuracy: 0.6667 - val_loss: 0.9129 - val_accuracy: 0.7500
Epoch 143/500
1/1 [==============================] - 0s 42ms/step - loss: 0.9636 - accuracy: 0.6667 - val_loss: 0.9125 - val_accuracy: 0.7500
Epoch 144/500
```

```
1/1 [==============================] - 0s 62ms/step - loss: 0.9631 - accuracy: 0.6667 - val_loss: 0.9120 - val_accuracy: 0.7500
Epoch 145/500
1/1 [==============================] - 0s 47ms/step - loss: 0.9626 - accuracy: 0.6944 - val_loss: 0.9117 - val_accuracy: 0.7500
Epoch 146/500
1/1 [==============================] - 0s 41ms/step - loss: 0.9620 - accuracy: 0.6944 - val_loss: 0.9112 - val_accuracy: 0.7500
Epoch 147/500
1/1 [==============================] - 0s 47ms/step - loss: 0.9615 - accuracy: 0.6944 - val_loss: 0.9108 - val_accuracy: 0.7500
Epoch 148/500
1/1 [==============================] - 0s 47ms/step - loss: 0.9610 - accuracy: 0.6944 - val_loss: 0.9104 - val_accuracy: 0.7500
Epoch 149/500
1/1 [==============================] - 0s 40ms/step - loss: 0.9605 - accuracy: 0.6944 - val_loss: 0.9100 - val_accuracy: 0.7500
Epoch 150/500
1/1 [==============================] - 0s 32ms/step - loss: 0.9600 - accuracy: 0.6944 - val_loss: 0.9096 - val_accuracy: 0.7500
Epoch 151/500
1/1 [==============================] - 0s 68ms/step - loss: 0.9595 - accuracy: 0.6944 - val_loss: 0.9092 - val_accuracy: 0.7500
Epoch 152/500
1/1 [==============================] - 0s 32ms/step - loss: 0.9589 - accuracy: 0.6944 - val_loss: 0.9088 - val_accuracy: 0.7500
Epoch 153/500
1/1 [==============================] - 0s 41ms/step - loss: 0.9584 - accuracy: 0.6944 - val_loss: 0.9083 - val_accuracy: 0.7500
Epoch 154/500
1/1 [==============================] - 0s 33ms/step - loss: 0.9579 - accuracy: 0.6944 - val_loss: 0.9079 - val_accuracy: 0.7500
Epoch 155/500
1/1 [==============================] - 0s 47ms/step - loss: 0.9574 - accuracy: 0.6944 - val_loss: 0.9076 - val_accuracy: 0.7500
Epoch 156/500
1/1 [==============================] - 0s 47ms/step - loss: 0.9569 - accuracy: 0.6944 - val_loss: 0.9071 - val_accuracy: 0.7500
Epoch 157/500
1/1 [==============================] - 0s 42ms/step - loss: 0.9564 - accuracy: 0.6944 - val_loss: 0.9068 - val_accuracy: 0.7500
Epoch 158/500
1/1 [==============================] - 0s 67ms/step - loss: 0.9558 - accuracy: 0.6944 - val_loss: 0.9064 - val_accuracy: 0.7500
Epoch 159/500
1/1 [==============================] - 0s 66ms/step - loss: 0.9553 - accuracy: 0.6944 - val_loss: 0.9060 - val_accuracy: 0.7500
Epoch 160/500
1/1 [==============================] - 0s 33ms/step - loss: 0.9548 - accuracy: 0.6944 - val_loss: 0.9056 - val_accuracy: 0.7500
Epoch 161/500
1/1 [==============================] - 0s 42ms/step - loss: 0.9543 - accuracy: 0.6944 - val_loss: 0.9052 - val_accuracy: 0.7500
Epoch 162/500
1/1 [==============================] - 0s 51ms/step - loss: 0.9538 - accuracy: 0.6667 - val_loss: 0.9048 - val_accuracy: 0.7500
Epoch 163/500
1/1 [==============================] - 0s 57ms/step - loss: 0.9533 - accuracy: 0.6667 - val_loss: 0.9045 - val_accuracy: 0.7500
Epoch 164/500
1/1 [==============================] - 0s 85ms/step - loss: 0.9527 - accuracy: 0.6667 - val_loss: 0.9041 - val_accuracy: 0.7500
```

```
Epoch 165/500
1/1 [==============================] - 0s 44ms/step - loss: 0.9522 - accuracy: 0.6667 - val_loss: 0.9037 - val_accuracy: 0.7500
Epoch 166/500
1/1 [==============================] - 0s 82ms/step - loss: 0.9517 - accuracy: 0.6667 - val_loss: 0.9033 - val_accuracy: 0.7500
Epoch 167/500
1/1 [==============================] - 0s 50ms/step - loss: 0.9512 - accuracy: 0.6667 - val_loss: 0.9029 - val_accuracy: 0.7500
Epoch 168/500
1/1 [==============================] - 0s 41ms/step - loss: 0.9507 - accuracy: 0.6667 - val_loss: 0.9026 - val_accuracy: 0.7500
Epoch 169/500
1/1 [==============================] - 0s 85ms/step - loss: 0.9502 - accuracy: 0.6667 - val_loss: 0.9022 - val_accuracy: 0.7500
Epoch 170/500
1/1 [==============================] - 0s 42ms/step - loss: 0.9497 - accuracy: 0.6667 - val_loss: 0.9019 - val_accuracy: 0.7500
Epoch 171/500
1/1 [==============================] - 0s 64ms/step - loss: 0.9492 - accuracy: 0.6667 - val_loss: 0.9015 - val_accuracy: 0.7500
Epoch 172/500
1/1 [==============================] - 0s 34ms/step - loss: 0.9486 - accuracy: 0.6667 - val_loss: 0.9011 - val_accuracy: 0.7500
Epoch 173/500
1/1 [==============================] - 0s 82ms/step - loss: 0.9481 - accuracy: 0.6667 - val_loss: 0.9007 - val_accuracy: 0.7500
Epoch 174/500
1/1 [==============================] - 0s 82ms/step - loss: 0.9476 - accuracy: 0.6667 - val_loss: 0.9003 - val_accuracy: 0.7500
Epoch 175/500
1/1 [==============================] - 0s 67ms/step - loss: 0.9471 - accuracy: 0.6667 - val_loss: 0.9001 - val_accuracy: 0.7500
Epoch 176/500
1/1 [==============================] - 0s 40ms/step - loss: 0.9466 - accuracy: 0.6389 - val_loss: 0.8997 - val_accuracy: 0.7500
Epoch 177/500
1/1 [==============================] - 0s 32ms/step - loss: 0.9461 - accuracy: 0.6389 - val_loss: 0.8994 - val_accuracy: 0.7500
Epoch 178/500
1/1 [==============================] - 0s 50ms/step - loss: 0.9456 - accuracy: 0.6389 - val_loss: 0.8989 - val_accuracy: 0.7500
Epoch 179/500
1/1 [==============================] - 0s 45ms/step - loss: 0.9451 - accuracy: 0.6389 - val_loss: 0.8986 - val_accuracy: 0.7500
Epoch 180/500
1/1 [==============================] - 0s 33ms/step - loss: 0.9446 - accuracy: 0.6389 - val_loss: 0.8982 - val_accuracy: 0.7500
Epoch 181/500
1/1 [==============================] - 0s 67ms/step - loss: 0.9441 - accuracy: 0.6389 - val_loss: 0.8979 - val_accuracy: 0.7500
Epoch 182/500
1/1 [==============================] - 0s 49ms/step - loss: 0.9436 - accuracy: 0.6389 - val_loss: 0.8976 - val_accuracy: 0.7500
Epoch 183/500
1/1 [==============================] - 0s 50ms/step - loss: 0.9431 - accuracy: 0.6389 - val_loss: 0.8972 - val_accuracy: 0.7500
Epoch 184/500
1/1 [==============================] - 0s 42ms/step - loss: 0.9426 - accuracy: 0.6389 - val_loss: 0.8969 - val_accuracy: 0.7500
Epoch 185/500
```

```
1/1 [==============================] - 0s 41ms/step - loss: 0.9421 - accuracy: 0.6389 - val_loss: 0.8965 - val_accuracy: 0.7500
Epoch 186/500
1/1 [==============================] - 0s 33ms/step - loss: 0.9416 - accuracy: 0.6389 - val_loss: 0.8962 - val_accuracy: 0.7500
Epoch 187/500
1/1 [==============================] - 0s 79ms/step - loss: 0.9411 - accuracy: 0.6389 - val_loss: 0.8959 - val_accuracy: 0.7500
Epoch 188/500
1/1 [==============================] - 0s 42ms/step - loss: 0.9406 - accuracy: 0.6389 - val_loss: 0.8955 - val_accuracy: 0.7500
Epoch 189/500
1/1 [==============================] - 0s 40ms/step - loss: 0.9401 - accuracy: 0.6389 - val_loss: 0.8951 - val_accuracy: 0.7500
Epoch 190/500
1/1 [==============================] - 0s 42ms/step - loss: 0.9395 - accuracy: 0.6389 - val_loss: 0.8948 - val_accuracy: 0.7500
Epoch 191/500
1/1 [==============================] - 0s 51ms/step - loss: 0.9390 - accuracy: 0.6389 - val_loss: 0.8944 - val_accuracy: 0.7500
Epoch 192/500
1/1 [==============================] - 0s 67ms/step - loss: 0.9385 - accuracy: 0.6389 - val_loss: 0.8941 - val_accuracy: 0.7500
Epoch 193/500
1/1 [==============================] - 0s 70ms/step - loss: 0.9380 - accuracy: 0.6389 - val_loss: 0.8938 - val_accuracy: 0.7500
Epoch 194/500
1/1 [==============================] - 0s 57ms/step - loss: 0.9375 - accuracy: 0.6389 - val_loss: 0.8934 - val_accuracy: 0.7500
Epoch 195/500
1/1 [==============================] - 0s 50ms/step - loss: 0.9370 - accuracy: 0.6389 - val_loss: 0.8932 - val_accuracy: 0.7500
Epoch 196/500
1/1 [==============================] - 0s 45ms/step - loss: 0.9365 - accuracy: 0.6667 - val_loss: 0.8928 - val_accuracy: 0.7500
Epoch 197/500
1/1 [==============================] - 0s 59ms/step - loss: 0.9360 - accuracy: 0.6667 - val_loss: 0.8924 - val_accuracy: 0.7500
Epoch 198/500
1/1 [==============================] - 0s 72ms/step - loss: 0.9355 - accuracy: 0.6667 - val_loss: 0.8921 - val_accuracy: 0.7500
Epoch 199/500
1/1 [==============================] - 0s 50ms/step - loss: 0.9350 - accuracy: 0.6667 - val_loss: 0.8918 - val_accuracy: 0.7500
Epoch 200/500
1/1 [==============================] - 0s 56ms/step - loss: 0.9345 - accuracy: 0.6667 - val_loss: 0.8915 - val_accuracy: 0.7500
Epoch 201/500
1/1 [==============================] - 0s 66ms/step - loss: 0.9341 - accuracy: 0.6667 - val_loss: 0.8912 - val_accuracy: 0.7500
Epoch 202/500
1/1 [==============================] - 0s 34ms/step - loss: 0.9336 - accuracy: 0.6667 - val_loss: 0.8908 - val_accuracy: 0.7500
Epoch 203/500
1/1 [==============================] - 0s 75ms/step - loss: 0.9331 - accuracy: 0.6667 - val_loss: 0.8905 - val_accuracy: 0.7500
Epoch 204/500
1/1 [==============================] - 0s 57ms/step - loss: 0.9326 - accuracy: 0.6667 - val_loss: 0.8901 - val_accuracy: 0.7500
Epoch 205/500
1/1 [==============================] - 0s 67ms/step - loss: 0.9321 - accuracy: 0.6667 - val_loss: 0.8899 - val_accuracy: 0.7500
```

```
Epoch 206/500
1/1 [==============================] - 0s 55ms/step - loss: 0.9316 - accuracy: 0.6667 - val_loss: 0.8895 - val_accuracy: 0.7500
Epoch 207/500
1/1 [==============================] - 0s 66ms/step - loss: 0.9311 - accuracy: 0.6667 - val_loss: 0.8892 - val_accuracy: 0.7500
Epoch 208/500
1/1 [==============================] - 0s 84ms/step - loss: 0.9306 - accuracy: 0.6667 - val_loss: 0.8889 - val_accuracy: 0.7500
Epoch 209/500
1/1 [==============================] - 0s 100ms/step - loss: 0.9301 - accuracy: 0.6667 - val_loss: 0.8885 - val_accuracy: 0.750
0
Epoch 210/500
1/1 [==============================] - 0s 51ms/step - loss: 0.9297 - accuracy: 0.6667 - val_loss: 0.8882 - val_accuracy: 0.7500
Epoch 211/500
1/1 [==============================] - 0s 55ms/step - loss: 0.9292 - accuracy: 0.6944 - val_loss: 0.8879 - val_accuracy: 0.7500
Epoch 212/500
1/1 [==============================] - 0s 67ms/step - loss: 0.9287 - accuracy: 0.6944 - val_loss: 0.8876 - val_accuracy: 0.7500
Epoch 213/500
1/1 [==============================] - 0s 66ms/step - loss: 0.9282 - accuracy: 0.6944 - val_loss: 0.8872 - val_accuracy: 0.7500
Epoch 214/500
1/1 [==============================] - 0s 50ms/step - loss: 0.9277 - accuracy: 0.6944 - val_loss: 0.8869 - val_accuracy: 0.7500
Epoch 215/500
1/1 [==============================] - 0s 50ms/step - loss: 0.9273 - accuracy: 0.6944 - val_loss: 0.8866 - val_accuracy: 0.7500
Epoch 216/500
1/1 [==============================] - 0s 33ms/step - loss: 0.9268 - accuracy: 0.6944 - val_loss: 0.8863 - val_accuracy: 0.7500
Epoch 217/500
1/1 [==============================] - 0s 37ms/step - loss: 0.9263 - accuracy: 0.6944 - val_loss: 0.8859 - val_accuracy: 0.7500
Epoch 218/500
1/1 [==============================] - 0s 33ms/step - loss: 0.9258 - accuracy: 0.6944 - val_loss: 0.8856 - val_accuracy: 0.7500
Epoch 219/500
1/1 [==============================] - 0s 40ms/step - loss: 0.9253 - accuracy: 0.6944 - val_loss: 0.8853 - val_accuracy: 0.7500
Epoch 220/500
1/1 [==============================] - 0s 50ms/step - loss: 0.9249 - accuracy: 0.6944 - val_loss: 0.8849 - val_accuracy: 0.7500
Epoch 221/500
1/1 [==============================] - 0s 46ms/step - loss: 0.9244 - accuracy: 0.6944 - val_loss: 0.8847 - val_accuracy: 0.7500
Epoch 222/500
1/1 [==============================] - 0s 37ms/step - loss: 0.9239 - accuracy: 0.6944 - val_loss: 0.8843 - val_accuracy: 0.7500
Epoch 223/500
1/1 [==============================] - 0s 55ms/step - loss: 0.9234 - accuracy: 0.6944 - val_loss: 0.8840 - val_accuracy: 0.7500
Epoch 224/500
1/1 [==============================] - 0s 34ms/step - loss: 0.9230 - accuracy: 0.6944 - val_loss: 0.8837 - val_accuracy: 0.7500
Epoch 225/500
1/1 [==============================] - 0s 51ms/step - loss: 0.9225 - accuracy: 0.6944 - val_loss: 0.8833 - val_accuracy: 0.7500
```

```
Epoch 226/500
1/1 [==============================] - 0s 33ms/step - loss: 0.9220 - accuracy: 0.6944 - val_loss: 0.8830 - val_accuracy: 0.7500
Epoch 227/500
1/1 [==============================] - 0s 75ms/step - loss: 0.9216 - accuracy: 0.6944 - val_loss: 0.8827 - val_accuracy: 0.7500
Epoch 228/500
1/1 [==============================] - 0s 49ms/step - loss: 0.9211 - accuracy: 0.6944 - val_loss: 0.8823 - val_accuracy: 0.7500
Epoch 229/500
1/1 [==============================] - 0s 33ms/step - loss: 0.9206 - accuracy: 0.6944 - val_loss: 0.8820 - val_accuracy: 0.7500
Epoch 230/500
1/1 [==============================] - 0s 48ms/step - loss: 0.9202 - accuracy: 0.6944 - val_loss: 0.8818 - val_accuracy: 0.7500
Epoch 231/500
1/1 [==============================] - 0s 81ms/step - loss: 0.9197 - accuracy: 0.6944 - val_loss: 0.8814 - val_accuracy: 0.7500
Epoch 232/500
1/1 [==============================] - 0s 76ms/step - loss: 0.9192 - accuracy: 0.6944 - val_loss: 0.8810 - val_accuracy: 0.7500
Epoch 233/500
1/1 [==============================] - 0s 84ms/step - loss: 0.9188 - accuracy: 0.6944 - val_loss: 0.8807 - val_accuracy: 0.7500
Epoch 234/500
1/1 [==============================] - 0s 75ms/step - loss: 0.9183 - accuracy: 0.6944 - val_loss: 0.8803 - val_accuracy: 0.7500
Epoch 235/500
1/1 [==============================] - 0s 57ms/step - loss: 0.9178 - accuracy: 0.6944 - val_loss: 0.8801 - val_accuracy: 0.7500
Epoch 236/500
1/1 [==============================] - 0s 50ms/step - loss: 0.9174 - accuracy: 0.6944 - val_loss: 0.8797 - val_accuracy: 0.7500
Epoch 237/500
1/1 [==============================] - 0s 59ms/step - loss: 0.9169 - accuracy: 0.6944 - val_loss: 0.8794 - val_accuracy: 0.7500
Epoch 238/500
1/1 [==============================] - 0s 73ms/step - loss: 0.9164 - accuracy: 0.6944 - val_loss: 0.8791 - val_accuracy: 0.7500
Epoch 239/500
1/1 [==============================] - 0s 52ms/step - loss: 0.9160 - accuracy: 0.6944 - val_loss: 0.8787 - val_accuracy: 0.7500
Epoch 240/500
1/1 [==============================] - 0s 48ms/step - loss: 0.9155 - accuracy: 0.6944 - val_loss: 0.8784 - val_accuracy: 0.7500
Epoch 241/500
1/1 [==============================] - 0s 58ms/step - loss: 0.9151 - accuracy: 0.6944 - val_loss: 0.8780 - val_accuracy: 0.7500
Epoch 242/500
1/1 [==============================] - 0s 33ms/step - loss: 0.9146 - accuracy: 0.6944 - val_loss: 0.8777 - val_accuracy: 0.7500
Epoch 243/500
1/1 [==============================] - 0s 49ms/step - loss: 0.9141 - accuracy: 0.6944 - val_loss: 0.8773 - val_accuracy: 0.7500
Epoch 244/500
1/1 [==============================] - 0s 34ms/step - loss: 0.9137 - accuracy: 0.6944 - val_loss: 0.8770 - val_accuracy: 0.7500
Epoch 245/500
1/1 [==============================] - 0s 25ms/step - loss: 0.9132 - accuracy: 0.6944 - val_loss: 0.8767 - val_accuracy: 0.7500
Epoch 246/500
```

```
1/1 [==============================] - 0s 67ms/step - loss: 0.9128 - accuracy: 0.6944 - val_loss: 0.8764 - val_accuracy: 0.7500
Epoch 247/500
1/1 [==============================] - 0s 100ms/step - loss: 0.9123 - accuracy: 0.6944 - val_loss: 0.8761 - val_accuracy: 0.750
0
Epoch 248/500
1/1 [==============================] - 0s 33ms/step - loss: 0.9118 - accuracy: 0.6944 - val_loss: 0.8757 - val_accuracy: 0.7500
Epoch 249/500
1/1 [==============================] - 0s 67ms/step - loss: 0.9114 - accuracy: 0.6944 - val_loss: 0.8753 - val_accuracy: 0.7500
Epoch 250/500
1/1 [==============================] - 0s 43ms/step - loss: 0.9109 - accuracy: 0.6944 - val_loss: 0.8750 - val_accuracy: 0.7500
Epoch 251/500
1/1 [==============================] - 0s 34ms/step - loss: 0.9104 - accuracy: 0.6944 - val_loss: 0.8747 - val_accuracy: 0.7500
Epoch 252/500
1/1 [==============================] - 0s 78ms/step - loss: 0.9100 - accuracy: 0.6944 - val_loss: 0.8745 - val_accuracy: 0.7500
Epoch 253/500
1/1 [==============================] - 0s 49ms/step - loss: 0.9095 - accuracy: 0.6944 - val_loss: 0.8741 - val_accuracy: 0.7500
Epoch 254/500
1/1 [==============================] - 0s 41ms/step - loss: 0.9091 - accuracy: 0.6944 - val_loss: 0.8738 - val_accuracy: 0.7500
Epoch 255/500
1/1 [==============================] - 0s 75ms/step - loss: 0.9086 - accuracy: 0.6944 - val_loss: 0.8734 - val_accuracy: 0.7500
Epoch 256/500
1/1 [==============================] - 0s 60ms/step - loss: 0.9081 - accuracy: 0.6944 - val_loss: 0.8731 - val_accuracy: 0.7500
Epoch 257/500
1/1 [==============================] - 0s 81ms/step - loss: 0.9077 - accuracy: 0.6944 - val_loss: 0.8727 - val_accuracy: 0.7500
Epoch 258/500
1/1 [==============================] - 0s 77ms/step - loss: 0.9072 - accuracy: 0.6944 - val_loss: 0.8724 - val_accuracy: 0.7500
Epoch 259/500
1/1 [==============================] - 0s 50ms/step - loss: 0.9068 - accuracy: 0.6944 - val_loss: 0.8722 - val_accuracy: 0.7500
Epoch 260/500
1/1 [==============================] - 0s 80ms/step - loss: 0.9063 - accuracy: 0.6944 - val_loss: 0.8718 - val_accuracy: 0.7500
Epoch 261/500
1/1 [==============================] - 0s 50ms/step - loss: 0.9059 - accuracy: 0.6944 - val_loss: 0.8715 - val_accuracy: 0.7500
Epoch 262/500
1/1 [==============================] - 0s 43ms/step - loss: 0.9054 - accuracy: 0.6944 - val_loss: 0.8711 - val_accuracy: 0.7500
Epoch 263/500
1/1 [==============================] - 0s 34ms/step - loss: 0.9050 - accuracy: 0.6944 - val_loss: 0.8708 - val_accuracy: 0.7500
Epoch 264/500
1/1 [==============================] - 0s 112ms/step - loss: 0.9045 - accuracy: 0.6944 - val_loss: 0.8704 - val_accuracy: 0.750
0
Epoch 265/500
1/1 [==============================] - 0s 33ms/step - loss: 0.9041 - accuracy: 0.6944 - val_loss: 0.8701 - val_accuracy: 0.7500
```

```
Epoch 266/500
1/1 [==============================] - 0s 33ms/step - loss: 0.9036 - accuracy: 0.6944 - val_loss: 0.8698 - val_accuracy: 0.7500
Epoch 267/500
1/1 [==============================] - 0s 67ms/step - loss: 0.9032 - accuracy: 0.6944 - val_loss: 0.8695 - val_accuracy: 0.7500
Epoch 268/500
1/1 [==============================] - 0s 50ms/step - loss: 0.9028 - accuracy: 0.6944 - val_loss: 0.8691 - val_accuracy: 0.7500
Epoch 269/500
1/1 [==============================] - 0s 50ms/step - loss: 0.9023 - accuracy: 0.6944 - val_loss: 0.8688 - val_accuracy: 0.7500
Epoch 270/500
1/1 [==============================] - 0s 50ms/step - loss: 0.9019 - accuracy: 0.6944 - val_loss: 0.8684 - val_accuracy: 0.7500
Epoch 271/500
1/1 [==============================] - 0s 50ms/step - loss: 0.9014 - accuracy: 0.6944 - val_loss: 0.8681 - val_accuracy: 0.7500
Epoch 272/500
1/1 [==============================] - 0s 50ms/step - loss: 0.9010 - accuracy: 0.6944 - val_loss: 0.8678 - val_accuracy: 0.7500
Epoch 273/500
1/1 [==============================] - 0s 67ms/step - loss: 0.9006 - accuracy: 0.6944 - val_loss: 0.8674 - val_accuracy: 0.7500
Epoch 274/500
1/1 [==============================] - 0s 49ms/step - loss: 0.9001 - accuracy: 0.6944 - val_loss: 0.8671 - val_accuracy: 0.7500
Epoch 275/500
1/1 [==============================] - 0s 54ms/step - loss: 0.8997 - accuracy: 0.6944 - val_loss: 0.8667 - val_accuracy: 0.7500
Epoch 276/500
1/1 [==============================] - 0s 40ms/step - loss: 0.8993 - accuracy: 0.6944 - val_loss: 0.8664 - val_accuracy: 0.7500
Epoch 277/500
1/1 [==============================] - 0s 34ms/step - loss: 0.8988 - accuracy: 0.6944 - val_loss: 0.8660 - val_accuracy: 0.7500
Epoch 278/500
1/1 [==============================] - 0s 65ms/step - loss: 0.8984 - accuracy: 0.6944 - val_loss: 0.8657 - val_accuracy: 0.7500
Epoch 279/500
1/1 [==============================] - 0s 50ms/step - loss: 0.8980 - accuracy: 0.6944 - val_loss: 0.8653 - val_accuracy: 0.7500
Epoch 280/500
1/1 [==============================] - 0s 42ms/step - loss: 0.8975 - accuracy: 0.6944 - val_loss: 0.8650 - val_accuracy: 0.7500
Epoch 281/500
1/1 [==============================] - 0s 41ms/step - loss: 0.8971 - accuracy: 0.6944 - val_loss: 0.8647 - val_accuracy: 0.7500
Epoch 282/500
1/1 [==============================] - 0s 79ms/step - loss: 0.8967 - accuracy: 0.6944 - val_loss: 0.8643 - val_accuracy: 0.7500
Epoch 283/500
1/1 [==============================] - 0s 44ms/step - loss: 0.8962 - accuracy: 0.6944 - val_loss: 0.8640 - val_accuracy: 0.7500
Epoch 284/500
1/1 [==============================] - 0s 40ms/step - loss: 0.8958 - accuracy: 0.6944 - val_loss: 0.8637 - val_accuracy: 0.7500
Epoch 285/500
1/1 [==============================] - 0s 83ms/step - loss: 0.8954 - accuracy: 0.6944 - val_loss: 0.8633 - val_accuracy: 0.7500
Epoch 286/500
```

```
1/1 [==============================] - 0s 32ms/step - loss: 0.8950 - accuracy: 0.6944 - val_loss: 0.8630 - val_accuracy: 0.7500
Epoch 287/500
1/1 [==============================] - 0s 43ms/step - loss: 0.8945 - accuracy: 0.6944 - val_loss: 0.8626 - val_accuracy: 0.7500
Epoch 288/500
1/1 [==============================] - 0s 40ms/step - loss: 0.8941 - accuracy: 0.6944 - val_loss: 0.8625 - val_accuracy: 0.7500
Epoch 289/500
1/1 [==============================] - 0s 34ms/step - loss: 0.8937 - accuracy: 0.6944 - val_loss: 0.8621 - val_accuracy: 0.7500
Epoch 290/500
1/1 [==============================] - 0s 50ms/step - loss: 0.8933 - accuracy: 0.6944 - val_loss: 0.8618 - val_accuracy: 0.7500
Epoch 291/500
1/1 [==============================] - 0s 45ms/step - loss: 0.8929 - accuracy: 0.6944 - val_loss: 0.8614 - val_accuracy: 0.7500
Epoch 292/500
1/1 [==============================] - 0s 34ms/step - loss: 0.8924 - accuracy: 0.6944 - val_loss: 0.8610 - val_accuracy: 0.7500
Epoch 293/500
1/1 [==============================] - 0s 67ms/step - loss: 0.8920 - accuracy: 0.6944 - val_loss: 0.8608 - val_accuracy: 0.7500
Epoch 294/500
1/1 [==============================] - 0s 60ms/step - loss: 0.8916 - accuracy: 0.6944 - val_loss: 0.8604 - val_accuracy: 0.7500
Epoch 295/500
1/1 [==============================] - 0s 68ms/step - loss: 0.8912 - accuracy: 0.6944 - val_loss: 0.8601 - val_accuracy: 0.7500
Epoch 296/500
1/1 [==============================] - 0s 82ms/step - loss: 0.8908 - accuracy: 0.6944 - val_loss: 0.8598 - val_accuracy: 0.7500
Epoch 297/500
1/1 [==============================] - 0s 48ms/step - loss: 0.8904 - accuracy: 0.6944 - val_loss: 0.8595 - val_accuracy: 0.7500
Epoch 298/500
1/1 [==============================] - 0s 42ms/step - loss: 0.8900 - accuracy: 0.6944 - val_loss: 0.8592 - val_accuracy: 0.7500
Epoch 299/500
1/1 [==============================] - 0s 62ms/step - loss: 0.8896 - accuracy: 0.6944 - val_loss: 0.8588 - val_accuracy: 0.7500
Epoch 300/500
1/1 [==============================] - 0s 59ms/step - loss: 0.8891 - accuracy: 0.6944 - val_loss: 0.8585 - val_accuracy: 0.7500
Epoch 301/500
1/1 [==============================] - 0s 48ms/step - loss: 0.8887 - accuracy: 0.6944 - val_loss: 0.8581 - val_accuracy: 0.7500
Epoch 302/500
1/1 [==============================] - 0s 42ms/step - loss: 0.8883 - accuracy: 0.6944 - val_loss: 0.8579 - val_accuracy: 0.7500
Epoch 303/500
1/1 [==============================] - 0s 67ms/step - loss: 0.8879 - accuracy: 0.6944 - val_loss: 0.8576 - val_accuracy: 0.7500
Epoch 304/500
1/1 [==============================] - 0s 100ms/step - loss: 0.8875 - accuracy: 0.6944 - val_loss: 0.8573 - val_accuracy: 0.750
0
Epoch 305/500
1/1 [==============================] - 0s 41ms/step - loss: 0.8871 - accuracy: 0.6944 - val_loss: 0.8569 - val_accuracy: 0.7500
Epoch 306/500
```

```
1/1 [==============================] - 0s 69ms/step - loss: 0.8867 - accuracy: 0.6944 - val_loss: 0.8565 - val_accuracy: 0.7500
Epoch 307/500
1/1 [==============================] - 0s 40ms/step - loss: 0.8863 - accuracy: 0.6944 - val_loss: 0.8562 - val_accuracy: 0.7500
Epoch 308/500
1/1 [==============================] - 0s 50ms/step - loss: 0.8859 - accuracy: 0.6944 - val_loss: 0.8560 - val_accuracy: 0.7500
Epoch 309/500
1/1 [==============================] - 0s 69ms/step - loss: 0.8855 - accuracy: 0.6944 - val_loss: 0.8555 - val_accuracy: 0.7500
Epoch 310/500
1/1 [==============================] - 0s 56ms/step - loss: 0.8851 - accuracy: 0.6944 - val_loss: 0.8553 - val_accuracy: 0.7500
Epoch 311/500
1/1 [==============================] - 0s 33ms/step - loss: 0.8847 - accuracy: 0.6944 - val_loss: 0.8549 - val_accuracy: 0.7500
Epoch 312/500
1/1 [==============================] - 0s 50ms/step - loss: 0.8843 - accuracy: 0.6944 - val_loss: 0.8547 - val_accuracy: 0.7500
Epoch 313/500
1/1 [==============================] - 0s 50ms/step - loss: 0.8839 - accuracy: 0.6944 - val_loss: 0.8543 - val_accuracy: 0.7500
Epoch 314/500
1/1 [==============================] - 0s 39ms/step - loss: 0.8835 - accuracy: 0.6944 - val_loss: 0.8540 - val_accuracy: 0.7500
Epoch 315/500
1/1 [==============================] - 0s 41ms/step - loss: 0.8831 - accuracy: 0.6944 - val_loss: 0.8536 - val_accuracy: 0.7500
Epoch 316/500
1/1 [==============================] - 0s 51ms/step - loss: 0.8827 - accuracy: 0.6944 - val_loss: 0.8533 - val_accuracy: 0.7500
Epoch 317/500
1/1 [==============================] - 0s 60ms/step - loss: 0.8823 - accuracy: 0.6944 - val_loss: 0.8531 - val_accuracy: 0.7500
Epoch 318/500
1/1 [==============================] - 0s 65ms/step - loss: 0.8820 - accuracy: 0.6944 - val_loss: 0.8527 - val_accuracy: 0.7500
Epoch 319/500
1/1 [==============================] - 0s 66ms/step - loss: 0.8816 - accuracy: 0.6944 - val_loss: 0.8524 - val_accuracy: 0.7500
Epoch 320/500
1/1 [==============================] - 0s 67ms/step - loss: 0.8812 - accuracy: 0.6944 - val_loss: 0.8520 - val_accuracy: 0.7500
Epoch 321/500
1/1 [==============================] - 0s 42ms/step - loss: 0.8808 - accuracy: 0.6944 - val_loss: 0.8517 - val_accuracy: 0.7500
Epoch 322/500
1/1 [==============================] - 0s 83ms/step - loss: 0.8804 - accuracy: 0.6944 - val_loss: 0.8514 - val_accuracy: 0.7500
Epoch 323/500
1/1 [==============================] - 0s 33ms/step - loss: 0.8800 - accuracy: 0.6944 - val_loss: 0.8510 - val_accuracy: 0.7500
Epoch 324/500
1/1 [==============================] - 0s 42ms/step - loss: 0.8796 - accuracy: 0.6944 - val_loss: 0.8507 - val_accuracy: 0.7500
Epoch 325/500
1/1 [==============================] - 0s 49ms/step - loss: 0.8792 - accuracy: 0.6944 - val_loss: 0.8504 - val_accuracy: 0.7500
Epoch 326/500
1/1 [==============================] - 0s 43ms/step - loss: 0.8788 - accuracy: 0.6944 - val_loss: 0.8501 - val_accuracy: 0.7500
```

```
Epoch 327/500
1/1 [==============================] - 0s 41ms/step - loss: 0.8784 - accuracy: 0.6944 - val_loss: 0.8498 - val_accuracy: 0.7500
Epoch 328/500
1/1 [==============================] - 0s 74ms/step - loss: 0.8780 - accuracy: 0.6944 - val_loss: 0.8495 - val_accuracy: 0.7500
Epoch 329/500
1/1 [==============================] - 0s 57ms/step - loss: 0.8776 - accuracy: 0.6944 - val_loss: 0.8492 - val_accuracy: 0.7500
Epoch 330/500
1/1 [==============================] - 0s 52ms/step - loss: 0.8773 - accuracy: 0.6944 - val_loss: 0.8488 - val_accuracy: 0.7500
Epoch 331/500
1/1 [==============================] - 0s 69ms/step - loss: 0.8769 - accuracy: 0.6944 - val_loss: 0.8485 - val_accuracy: 0.7500
Epoch 332/500
1/1 [==============================] - 0s 40ms/step - loss: 0.8765 - accuracy: 0.6944 - val_loss: 0.8482 - val_accuracy: 0.7500
Epoch 333/500
1/1 [==============================] - 0s 32ms/step - loss: 0.8761 - accuracy: 0.6944 - val_loss: 0.8479 - val_accuracy: 0.7500
Epoch 334/500
1/1 [==============================] - 0s 47ms/step - loss: 0.8757 - accuracy: 0.6944 - val_loss: 0.8476 - val_accuracy: 0.7500
Epoch 335/500
1/1 [==============================] - 0s 47ms/step - loss: 0.8753 - accuracy: 0.6944 - val_loss: 0.8473 - val_accuracy: 0.7500
Epoch 336/500
1/1 [==============================] - 0s 47ms/step - loss: 0.8749 - accuracy: 0.6944 - val_loss: 0.8469 - val_accuracy: 0.7500
Epoch 337/500
1/1 [==============================] - 0s 31ms/step - loss: 0.8745 - accuracy: 0.6944 - val_loss: 0.8466 - val_accuracy: 0.7500
Epoch 338/500
1/1 [==============================] - 0s 47ms/step - loss: 0.8741 - accuracy: 0.6944 - val_loss: 0.8462 - val_accuracy: 0.7500
Epoch 339/500
1/1 [==============================] - 0s 31ms/step - loss: 0.8738 - accuracy: 0.6944 - val_loss: 0.8460 - val_accuracy: 0.7500
Epoch 340/500
1/1 [==============================] - 0s 47ms/step - loss: 0.8734 - accuracy: 0.6944 - val_loss: 0.8457 - val_accuracy: 0.7500
Epoch 341/500
1/1 [==============================] - 0s 44ms/step - loss: 0.8730 - accuracy: 0.6944 - val_loss: 0.8454 - val_accuracy: 0.7500
Epoch 342/500
1/1 [==============================] - 0s 110ms/step - loss: 0.8726 - accuracy: 0.6944 - val_loss: 0.8450 - val_accuracy: 0.750
0
Epoch 343/500
1/1 [==============================] - 0s 78ms/step - loss: 0.8722 - accuracy: 0.6944 - val_loss: 0.8447 - val_accuracy: 0.7500
Epoch 344/500
1/1 [==============================] - 0s 63ms/step - loss: 0.8718 - accuracy: 0.6944 - val_loss: 0.8444 - val_accuracy: 0.7500
Epoch 345/500
1/1 [==============================] - 0s 78ms/step - loss: 0.8714 - accuracy: 0.6944 - val_loss: 0.8440 - val_accuracy: 0.7500
Epoch 346/500
1/1 [==============================] - 0s 56ms/step - loss: 0.8711 - accuracy: 0.6944 - val_loss: 0.8437 - val_accuracy: 0.7500
```

```
Epoch 347/500
1/1 [==============================] - 0s 47ms/step - loss: 0.8707 - accuracy: 0.6944 - val_loss: 0.8434 - val_accuracy: 0.7500
Epoch 348/500
1/1 [==============================] - 0s 47ms/step - loss: 0.8703 - accuracy: 0.6944 - val_loss: 0.8431 - val_accuracy: 0.7500
Epoch 349/500
1/1 [==============================] - 0s 31ms/step - loss: 0.8699 - accuracy: 0.6944 - val_loss: 0.8427 - val_accuracy: 0.7500
Epoch 350/500
1/1 [==============================] - 0s 40ms/step - loss: 0.8695 - accuracy: 0.6944 - val_loss: 0.8424 - val_accuracy: 0.7500
Epoch 351/500
1/1 [==============================] - 0s 47ms/step - loss: 0.8692 - accuracy: 0.6944 - val_loss: 0.8421 - val_accuracy: 0.7500
Epoch 352/500
1/1 [==============================] - 0s 47ms/step - loss: 0.8688 - accuracy: 0.6944 - val_loss: 0.8417 - val_accuracy: 0.7500
Epoch 353/500
1/1 [==============================] - 0s 47ms/step - loss: 0.8684 - accuracy: 0.6944 - val_loss: 0.8414 - val_accuracy: 0.7500
Epoch 354/500
1/1 [==============================] - 0s 48ms/step - loss: 0.8680 - accuracy: 0.6944 - val_loss: 0.8412 - val_accuracy: 0.7500
Epoch 355/500
1/1 [==============================] - 0s 63ms/step - loss: 0.8676 - accuracy: 0.6944 - val_loss: 0.8409 - val_accuracy: 0.7500
Epoch 356/500
1/1 [==============================] - 0s 47ms/step - loss: 0.8673 - accuracy: 0.6944 - val_loss: 0.8405 - val_accuracy: 0.7500
Epoch 357/500
1/1 [==============================] - 0s 47ms/step - loss: 0.8669 - accuracy: 0.6944 - val_loss: 0.8402 - val_accuracy: 0.7500
Epoch 358/500
1/1 [==============================] - 0s 31ms/step - loss: 0.8665 - accuracy: 0.6944 - val_loss: 0.8398 - val_accuracy: 0.7500
Epoch 359/500
1/1 [==============================] - 0s 47ms/step - loss: 0.8661 - accuracy: 0.6944 - val_loss: 0.8395 - val_accuracy: 0.7500
Epoch 360/500
1/1 [==============================] - 0s 47ms/step - loss: 0.8657 - accuracy: 0.6944 - val_loss: 0.8392 - val_accuracy: 0.7500
Epoch 361/500
1/1 [==============================] - 0s 31ms/step - loss: 0.8654 - accuracy: 0.6944 - val_loss: 0.8388 - val_accuracy: 0.7500
Epoch 362/500
1/1 [==============================] - 0s 78ms/step - loss: 0.8650 - accuracy: 0.6944 - val_loss: 0.8386 - val_accuracy: 0.7500
Epoch 363/500
1/1 [==============================] - 0s 31ms/step - loss: 0.8646 - accuracy: 0.6944 - val_loss: 0.8383 - val_accuracy: 0.7500
Epoch 364/500
1/1 [==============================] - 0s 47ms/step - loss: 0.8642 - accuracy: 0.6944 - val_loss: 0.8380 - val_accuracy: 0.7500
Epoch 365/500
1/1 [==============================] - 0s 31ms/step - loss: 0.8639 - accuracy: 0.6944 - val_loss: 0.8376 - val_accuracy: 0.7500
Epoch 366/500
1/1 [==============================] - 0s 47ms/step - loss: 0.8635 - accuracy: 0.6944 - val_loss: 0.8373 - val_accuracy: 0.7500
Epoch 367/500
```

```
1/1 [==============================] - 0s 47ms/step - loss: 0.8631 - accuracy: 0.6944 - val_loss: 0.8370 - val_accuracy: 0.7500
Epoch 368/500
1/1 [==============================] - 0s 63ms/step - loss: 0.8627 - accuracy: 0.6944 - val_loss: 0.8366 - val_accuracy: 0.7500
Epoch 369/500
1/1 [==============================] - 0s 62ms/step - loss: 0.8624 - accuracy: 0.6944 - val_loss: 0.8364 - val_accuracy: 0.7500
Epoch 370/500
1/1 [==============================] - 0s 62ms/step - loss: 0.8620 - accuracy: 0.6944 - val_loss: 0.8361 - val_accuracy: 0.7500
Epoch 371/500
1/1 [==============================] - 0s 31ms/step - loss: 0.8616 - accuracy: 0.6944 - val_loss: 0.8358 - val_accuracy: 0.7500
Epoch 372/500
1/1 [==============================] - 0s 47ms/step - loss: 0.8612 - accuracy: 0.6944 - val_loss: 0.8354 - val_accuracy: 0.7500
Epoch 373/500
1/1 [==============================] - 0s 45ms/step - loss: 0.8609 - accuracy: 0.6944 - val_loss: 0.8351 - val_accuracy: 0.7500
Epoch 374/500
1/1 [==============================] - 0s 31ms/step - loss: 0.8605 - accuracy: 0.6944 - val_loss: 0.8348 - val_accuracy: 0.7500
Epoch 375/500
1/1 [==============================] - 0s 36ms/step - loss: 0.8601 - accuracy: 0.6944 - val_loss: 0.8345 - val_accuracy: 0.7500
Epoch 376/500
1/1 [==============================] - 0s 32ms/step - loss: 0.8598 - accuracy: 0.6944 - val_loss: 0.8342 - val_accuracy: 0.7500
Epoch 377/500
1/1 [==============================] - 0s 39ms/step - loss: 0.8594 - accuracy: 0.6944 - val_loss: 0.8340 - val_accuracy: 0.7500
Epoch 378/500
1/1 [==============================] - 0s 40ms/step - loss: 0.8590 - accuracy: 0.6944 - val_loss: 0.8337 - val_accuracy: 0.7500
Epoch 379/500
1/1 [==============================] - 0s 41ms/step - loss: 0.8586 - accuracy: 0.6944 - val_loss: 0.8334 - val_accuracy: 0.7500
Epoch 380/500
1/1 [==============================] - 0s 69ms/step - loss: 0.8583 - accuracy: 0.6944 - val_loss: 0.8330 - val_accuracy: 0.7500
Epoch 381/500
1/1 [==============================] - 0s 31ms/step - loss: 0.8579 - accuracy: 0.7222 - val_loss: 0.8328 - val_accuracy: 0.7500
Epoch 382/500
1/1 [==============================] - 0s 31ms/step - loss: 0.8575 - accuracy: 0.7222 - val_loss: 0.8325 - val_accuracy: 0.7500
Epoch 383/500
1/1 [==============================] - 0s 41ms/step - loss: 0.8572 - accuracy: 0.7222 - val_loss: 0.8322 - val_accuracy: 0.7500
Epoch 384/500
1/1 [==============================] - 0s 52ms/step - loss: 0.8568 - accuracy: 0.7222 - val_loss: 0.8319 - val_accuracy: 0.7500
Epoch 385/500
1/1 [==============================] - 0s 28ms/step - loss: 0.8564 - accuracy: 0.7222 - val_loss: 0.8317 - val_accuracy: 0.7500
Epoch 386/500
1/1 [==============================] - 0s 47ms/step - loss: 0.8561 - accuracy: 0.7222 - val_loss: 0.8313 - val_accuracy: 0.7500
Epoch 387/500
1/1 [==============================] - 0s 40ms/step - loss: 0.8557 - accuracy: 0.7222 - val_loss: 0.8311 - val_accuracy: 0.7500
```

```
Epoch 388/500
1/1 [==============================] - 0s 31ms/step - loss: 0.8553 - accuracy: 0.7222 - val_loss: 0.8307 - val_accuracy: 0.7500
Epoch 389/500
1/1 [==============================] - 0s 78ms/step - loss: 0.8550 - accuracy: 0.7222 - val_loss: 0.8305 - val_accuracy: 0.7500
Epoch 390/500
1/1 [==============================] - 0s 41ms/step - loss: 0.8546 - accuracy: 0.7222 - val_loss: 0.8301 - val_accuracy: 0.7500
Epoch 391/500
1/1 [==============================] - 0s 31ms/step - loss: 0.8542 - accuracy: 0.7222 - val_loss: 0.8299 - val_accuracy: 0.7500
Epoch 392/500
1/1 [==============================] - 0s 31ms/step - loss: 0.8539 - accuracy: 0.7222 - val_loss: 0.8297 - val_accuracy: 0.7500
Epoch 393/500
1/1 [==============================] - 0s 47ms/step - loss: 0.8535 - accuracy: 0.7222 - val_loss: 0.8294 - val_accuracy: 0.7500
Epoch 394/500
1/1 [==============================] - 0s 31ms/step - loss: 0.8531 - accuracy: 0.7222 - val_loss: 0.8291 - val_accuracy: 0.7500
Epoch 395/500
1/1 [==============================] - 0s 96ms/step - loss: 0.8528 - accuracy: 0.7222 - val_loss: 0.8288 - val_accuracy: 0.7500
Epoch 396/500
1/1 [==============================] - 0s 30ms/step - loss: 0.8524 - accuracy: 0.7222 - val_loss: 0.8284 - val_accuracy: 0.7500
Epoch 397/500
1/1 [==============================] - 0s 63ms/step - loss: 0.8520 - accuracy: 0.7222 - val_loss: 0.8282 - val_accuracy: 0.7500
Epoch 398/500
1/1 [==============================] - 0s 62ms/step - loss: 0.8517 - accuracy: 0.7222 - val_loss: 0.8279 - val_accuracy: 0.7500
Epoch 399/500
1/1 [==============================] - 0s 78ms/step - loss: 0.8513 - accuracy: 0.7222 - val_loss: 0.8277 - val_accuracy: 0.7500
Epoch 400/500
1/1 [==============================] - 0s 62ms/step - loss: 0.8510 - accuracy: 0.7222 - val_loss: 0.8274 - val_accuracy: 0.7500
Epoch 401/500
1/1 [==============================] - 0s 78ms/step - loss: 0.8506 - accuracy: 0.7222 - val_loss: 0.8271 - val_accuracy: 0.7500
Epoch 402/500
1/1 [==============================] - 0s 47ms/step - loss: 0.8503 - accuracy: 0.7222 - val_loss: 0.8268 - val_accuracy: 0.7500
Epoch 403/500
1/1 [==============================] - 0s 62ms/step - loss: 0.8499 - accuracy: 0.7222 - val_loss: 0.8265 - val_accuracy: 0.7500
Epoch 404/500
1/1 [==============================] - 0s 62ms/step - loss: 0.8495 - accuracy: 0.7222 - val_loss: 0.8262 - val_accuracy: 0.7500
Epoch 405/500
1/1 [==============================] - 0s 57ms/step - loss: 0.8492 - accuracy: 0.7222 - val_loss: 0.8259 - val_accuracy: 0.7500
Epoch 406/500
1/1 [==============================] - 0s 78ms/step - loss: 0.8488 - accuracy: 0.7222 - val_loss: 0.8256 - val_accuracy: 0.7500
Epoch 407/500
1/1 [==============================] - 0s 78ms/step - loss: 0.8485 - accuracy: 0.7222 - val_loss: 0.8255 - val_accuracy: 0.7500
Epoch 408/500
```

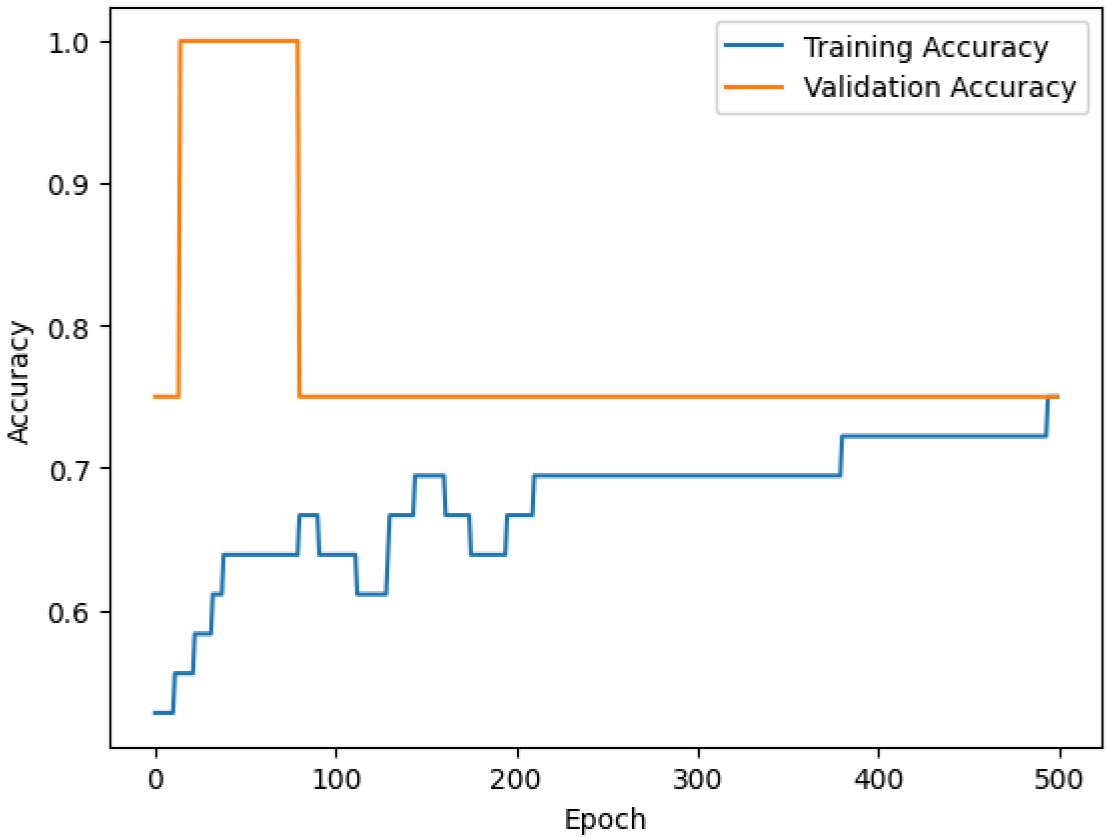
```
1/1 [==============================] - 0s 47ms/step - loss: 0.8481 - accuracy: 0.7222 - val_loss: 0.8252 - val_accuracy: 0.7500
Epoch 409/500
1/1 [==============================] - 0s 31ms/step - loss: 0.8478 - accuracy: 0.7222 - val_loss: 0.8249 - val_accuracy: 0.7500
Epoch 410/500
1/1 [==============================] - 0s 79ms/step - loss: 0.8474 - accuracy: 0.7222 - val_loss: 0.8246 - val_accuracy: 0.7500
Epoch 411/500
1/1 [==============================] - 0s 47ms/step - loss: 0.8470 - accuracy: 0.7222 - val_loss: 0.8243 - val_accuracy: 0.7500
Epoch 412/500
1/1 [==============================] - 0s 39ms/step - loss: 0.8467 - accuracy: 0.7222 - val_loss: 0.8240 - val_accuracy: 0.7500
Epoch 413/500
1/1 [==============================] - 0s 39ms/step - loss: 0.8463 - accuracy: 0.7222 - val_loss: 0.8237 - val_accuracy: 0.7500
Epoch 414/500
1/1 [==============================] - 0s 40ms/step - loss: 0.8460 - accuracy: 0.7222 - val_loss: 0.8235 - val_accuracy: 0.7500
Epoch 415/500
1/1 [==============================] - 0s 48ms/step - loss: 0.8456 - accuracy: 0.7222 - val_loss: 0.8232 - val_accuracy: 0.7500
Epoch 416/500
1/1 [==============================] - 0s 70ms/step - loss: 0.8453 - accuracy: 0.7222 - val_loss: 0.8229 - val_accuracy: 0.7500
Epoch 417/500
1/1 [==============================] - 0s 63ms/step - loss: 0.8450 - accuracy: 0.7222 - val_loss: 0.8227 - val_accuracy: 0.7500
Epoch 418/500
1/1 [==============================] - 0s 78ms/step - loss: 0.8446 - accuracy: 0.7222 - val_loss: 0.8224 - val_accuracy: 0.7500
Epoch 419/500
1/1 [==============================] - 0s 47ms/step - loss: 0.8443 - accuracy: 0.7222 - val_loss: 0.8221 - val_accuracy: 0.7500
Epoch 420/500
1/1 [==============================] - 0s 62ms/step - loss: 0.8439 - accuracy: 0.7222 - val_loss: 0.8218 - val_accuracy: 0.7500
Epoch 421/500
1/1 [==============================] - 0s 62ms/step - loss: 0.8436 - accuracy: 0.7222 - val_loss: 0.8215 - val_accuracy: 0.7500
Epoch 422/500
1/1 [==============================] - 0s 41ms/step - loss: 0.8432 - accuracy: 0.7222 - val_loss: 0.8213 - val_accuracy: 0.7500
Epoch 423/500
1/1 [==============================] - 0s 78ms/step - loss: 0.8429 - accuracy: 0.7222 - val_loss: 0.8211 - val_accuracy: 0.7500
Epoch 424/500
1/1 [==============================] - 0s 62ms/step - loss: 0.8426 - accuracy: 0.7222 - val_loss: 0.8208 - val_accuracy: 0.7500
Epoch 425/500
1/1 [==============================] - 0s 78ms/step - loss: 0.8422 - accuracy: 0.7222 - val_loss: 0.8205 - val_accuracy: 0.7500
Epoch 426/500
1/1 [==============================] - 0s 62ms/step - loss: 0.8419 - accuracy: 0.7222 - val_loss: 0.8202 - val_accuracy: 0.7500
Epoch 427/500
1/1 [==============================] - 0s 94ms/step - loss: 0.8415 - accuracy: 0.7222 - val_loss: 0.8200 - val_accuracy: 0.7500
Epoch 428/500
1/1 [==============================] - 0s 31ms/step - loss: 0.8412 - accuracy: 0.7222 - val_loss: 0.8196 - val_accuracy: 0.7500
```

```
Epoch 429/500
1/1 [==============================] - 0s 42ms/step - loss: 0.8408 - accuracy: 0.7222 - val_loss: 0.8194 - val_accuracy: 0.7500
Epoch 430/500
1/1 [==============================] - 0s 62ms/step - loss: 0.8405 - accuracy: 0.7222 - val_loss: 0.8191 - val_accuracy: 0.7500
Epoch 431/500
1/1 [==============================] - 0s 62ms/step - loss: 0.8402 - accuracy: 0.7222 - val_loss: 0.8189 - val_accuracy: 0.7500
Epoch 432/500
1/1 [==============================] - 0s 62ms/step - loss: 0.8398 - accuracy: 0.7222 - val_loss: 0.8186 - val_accuracy: 0.7500
Epoch 433/500
1/1 [==============================] - 0s 63ms/step - loss: 0.8395 - accuracy: 0.7222 - val_loss: 0.8184 - val_accuracy: 0.7500
Epoch 434/500
1/1 [==============================] - 0s 78ms/step - loss: 0.8392 - accuracy: 0.7222 - val_loss: 0.8180 - val_accuracy: 0.7500
Epoch 435/500
1/1 [==============================] - 0s 47ms/step - loss: 0.8388 - accuracy: 0.7222 - val_loss: 0.8178 - val_accuracy: 0.7500
Epoch 436/500
1/1 [==============================] - 0s 47ms/step - loss: 0.8385 - accuracy: 0.7222 - val_loss: 0.8175 - val_accuracy: 0.7500
Epoch 437/500
1/1 [==============================] - 0s 40ms/step - loss: 0.8381 - accuracy: 0.7222 - val_loss: 0.8172 - val_accuracy: 0.7500
Epoch 438/500
1/1 [==============================] - 0s 50ms/step - loss: 0.8378 - accuracy: 0.7222 - val_loss: 0.8170 - val_accuracy: 0.7500
Epoch 439/500
1/1 [==============================] - 0s 50ms/step - loss: 0.8375 - accuracy: 0.7222 - val_loss: 0.8167 - val_accuracy: 0.7500
Epoch 440/500
1/1 [==============================] - 0s 57ms/step - loss: 0.8371 - accuracy: 0.7222 - val_loss: 0.8165 - val_accuracy: 0.7500
Epoch 441/500
1/1 [==============================] - 0s 80ms/step - loss: 0.8368 - accuracy: 0.7222 - val_loss: 0.8161 - val_accuracy: 0.7500
Epoch 442/500
1/1 [==============================] - 0s 50ms/step - loss: 0.8365 - accuracy: 0.7222 - val_loss: 0.8159 - val_accuracy: 0.7500
Epoch 443/500
1/1 [==============================] - 0s 50ms/step - loss: 0.8361 - accuracy: 0.7222 - val_loss: 0.8156 - val_accuracy: 0.7500
Epoch 444/500
1/1 [==============================] - 0s 67ms/step - loss: 0.8358 - accuracy: 0.7222 - val_loss: 0.8153 - val_accuracy: 0.7500
Epoch 445/500
1/1 [==============================] - 0s 39ms/step - loss: 0.8354 - accuracy: 0.7222 - val_loss: 0.8151 - val_accuracy: 0.7500
Epoch 446/500
1/1 [==============================] - 0s 51ms/step - loss: 0.8351 - accuracy: 0.7222 - val_loss: 0.8148 - val_accuracy: 0.7500
Epoch 447/500
1/1 [==============================] - 0s 66ms/step - loss: 0.8348 - accuracy: 0.7222 - val_loss: 0.8145 - val_accuracy: 0.7500
Epoch 448/500
1/1 [==============================] - 0s 67ms/step - loss: 0.8344 - accuracy: 0.7222 - val_loss: 0.8143 - val_accuracy: 0.7500
Epoch 449/500
```

```
1/1 [==============================] - 0s 55ms/step - loss: 0.8341 - accuracy: 0.7222 - val_loss: 0.8139 - val_accuracy: 0.7500
Epoch 450/500
1/1 [==============================] - 0s 49ms/step - loss: 0.8338 - accuracy: 0.7222 - val_loss: 0.8136 - val_accuracy: 0.7500
Epoch 451/500
1/1 [==============================] - 0s 60ms/step - loss: 0.8334 - accuracy: 0.7222 - val_loss: 0.8134 - val_accuracy: 0.7500
Epoch 452/500
1/1 [==============================] - 0s 60ms/step - loss: 0.8331 - accuracy: 0.7222 - val_loss: 0.8132 - val_accuracy: 0.7500
Epoch 453/500
1/1 [==============================] - 0s 50ms/step - loss: 0.8328 - accuracy: 0.7222 - val_loss: 0.8129 - val_accuracy: 0.7500
Epoch 454/500
1/1 [==============================] - 0s 51ms/step - loss: 0.8324 - accuracy: 0.7222 - val_loss: 0.8126 - val_accuracy: 0.7500
Epoch 455/500
1/1 [==============================] - 0s 74ms/step - loss: 0.8321 - accuracy: 0.7222 - val_loss: 0.8123 - val_accuracy: 0.7500
Epoch 456/500
1/1 [==============================] - 0s 51ms/step - loss: 0.8318 - accuracy: 0.7222 - val_loss: 0.8120 - val_accuracy: 0.7500
Epoch 457/500
1/1 [==============================] - 0s 47ms/step - loss: 0.8315 - accuracy: 0.7222 - val_loss: 0.8117 - val_accuracy: 0.7500
Epoch 458/500
1/1 [==============================] - 0s 45ms/step - loss: 0.8311 - accuracy: 0.7222 - val_loss: 0.8115 - val_accuracy: 0.7500
Epoch 459/500
1/1 [==============================] - 0s 69ms/step - loss: 0.8308 - accuracy: 0.7222 - val_loss: 0.8112 - val_accuracy: 0.7500
Epoch 460/500
1/1 [==============================] - 0s 82ms/step - loss: 0.8305 - accuracy: 0.7222 - val_loss: 0.8110 - val_accuracy: 0.7500
Epoch 461/500
1/1 [==============================] - 0s 68ms/step - loss: 0.8301 - accuracy: 0.7222 - val_loss: 0.8107 - val_accuracy: 0.7500
Epoch 462/500
1/1 [==============================] - 0s 48ms/step - loss: 0.8298 - accuracy: 0.7222 - val_loss: 0.8105 - val_accuracy: 0.7500
Epoch 463/500
1/1 [==============================] - 0s 74ms/step - loss: 0.8295 - accuracy: 0.7222 - val_loss: 0.8101 - val_accuracy: 0.7500
Epoch 464/500
1/1 [==============================] - 0s 66ms/step - loss: 0.8291 - accuracy: 0.7222 - val_loss: 0.8099 - val_accuracy: 0.7500
Epoch 465/500
1/1 [==============================] - 0s 50ms/step - loss: 0.8288 - accuracy: 0.7222 - val_loss: 0.8096 - val_accuracy: 0.7500
Epoch 466/500
1/1 [==============================] - 0s 55ms/step - loss: 0.8285 - accuracy: 0.7222 - val_loss: 0.8093 - val_accuracy: 0.7500
Epoch 467/500
1/1 [==============================] - 0s 44ms/step - loss: 0.8282 - accuracy: 0.7222 - val_loss: 0.8090 - val_accuracy: 0.7500
Epoch 468/500
1/1 [==============================] - 0s 128ms/step - loss: 0.8278 - accuracy: 0.7222 - val_loss: 0.8088 - val_accuracy: 0.750
0
Epoch 469/500
```

```
1/1 [==============================] - 0s 68ms/step - loss: 0.8275 - accuracy: 0.7222 - val_loss: 0.8086 - val_accuracy: 0.7500
Epoch 470/500
1/1 [==============================] - 0s 46ms/step - loss: 0.8272 - accuracy: 0.7222 - val_loss: 0.8083 - val_accuracy: 0.7500
Epoch 471/500
1/1 [==============================] - 0s 42ms/step - loss: 0.8268 - accuracy: 0.7222 - val_loss: 0.8080 - val_accuracy: 0.7500
Epoch 472/500
1/1 [==============================] - 0s 56ms/step - loss: 0.8265 - accuracy: 0.7222 - val_loss: 0.8077 - val_accuracy: 0.7500
Epoch 473/500
1/1 [==============================] - 0s 50ms/step - loss: 0.8262 - accuracy: 0.7222 - val_loss: 0.8074 - val_accuracy: 0.7500
Epoch 474/500
1/1 [==============================] - 0s 50ms/step - loss: 0.8259 - accuracy: 0.7222 - val_loss: 0.8071 - val_accuracy: 0.7500
Epoch 475/500
1/1 [==============================] - 0s 75ms/step - loss: 0.8255 - accuracy: 0.7222 - val_loss: 0.8069 - val_accuracy: 0.7500
Epoch 476/500
1/1 [==============================] - 0s 67ms/step - loss: 0.8252 - accuracy: 0.7222 - val_loss: 0.8066 - val_accuracy: 0.7500
Epoch 477/500
1/1 [==============================] - 0s 36ms/step - loss: 0.8249 - accuracy: 0.7222 - val_loss: 0.8063 - val_accuracy: 0.7500
Epoch 478/500
1/1 [==============================] - 0s 106ms/step - loss: 0.8246 - accuracy: 0.7222 - val_loss: 0.8060 - val_accuracy: 0.750
0
Epoch 479/500
1/1 [==============================] - 0s 47ms/step - loss: 0.8242 - accuracy: 0.7222 - val_loss: 0.8057 - val_accuracy: 0.7500
Epoch 480/500
1/1 [==============================] - 0s 54ms/step - loss: 0.8239 - accuracy: 0.7222 - val_loss: 0.8055 - val_accuracy: 0.7500
Epoch 481/500
1/1 [==============================] - 0s 79ms/step - loss: 0.8236 - accuracy: 0.7222 - val_loss: 0.8051 - val_accuracy: 0.7500
Epoch 482/500
1/1 [==============================] - 0s 55ms/step - loss: 0.8233 - accuracy: 0.7222 - val_loss: 0.8049 - val_accuracy: 0.7500
Epoch 483/500
1/1 [==============================] - 0s 67ms/step - loss: 0.8229 - accuracy: 0.7222 - val_loss: 0.8046 - val_accuracy: 0.7500
Epoch 484/500
1/1 [==============================] - 0s 58ms/step - loss: 0.8226 - accuracy: 0.7222 - val_loss: 0.8043 - val_accuracy: 0.7500
Epoch 485/500
1/1 [==============================] - 0s 41ms/step - loss: 0.8223 - accuracy: 0.7222 - val_loss: 0.8041 - val_accuracy: 0.7500
Epoch 486/500
1/1 [==============================] - 0s 53ms/step - loss: 0.8220 - accuracy: 0.7222 - val_loss: 0.8037 - val_accuracy: 0.7500
Epoch 487/500
1/1 [==============================] - 0s 41ms/step - loss: 0.8216 - accuracy: 0.7222 - val_loss: 0.8035 - val_accuracy: 0.7500
Epoch 488/500
1/1 [==============================] - 0s 50ms/step - loss: 0.8213 - accuracy: 0.7222 - val_loss: 0.8032 - val_accuracy: 0.7500
Epoch 489/500
```

```
1/1 [==============================] - 0s 45ms/step - loss: 0.8210 - accuracy: 0.7222 - val_loss: 0.8029 - val_accuracy: 0.7500
Epoch 490/500
1/1 [==============================] - 0s 109ms/step - loss: 0.8207 - accuracy: 0.7222 - val_loss: 0.8025 - val_accuracy: 0.750
0
Epoch 491/500
1/1 [==============================] - 0s 50ms/step - loss: 0.8204 - accuracy: 0.7222 - val_loss: 0.8024 - val_accuracy: 0.7500
Epoch 492/500
1/1 [==============================] - 0s 62ms/step - loss: 0.8200 - accuracy: 0.7222 - val_loss: 0.8021 - val_accuracy: 0.7500
Epoch 493/500
1/1 [==============================] - 0s 72ms/step - loss: 0.8197 - accuracy: 0.7222 - val_loss: 0.8018 - val_accuracy: 0.7500
Epoch 494/500
1/1 [==============================] - 0s 46ms/step - loss: 0.8194 - accuracy: 0.7222 - val_loss: 0.8015 - val_accuracy: 0.7500
Epoch 495/500
1/1 [==============================] - 0s 69ms/step - loss: 0.8191 - accuracy: 0.7500 - val_loss: 0.8012 - val_accuracy: 0.7500
Epoch 496/500
1/1 [==============================] - 0s 49ms/step - loss: 0.8188 - accuracy: 0.7500 - val_loss: 0.8009 - val_accuracy: 0.7500
Epoch 497/500
1/1 [==============================] - 0s 66ms/step - loss: 0.8184 - accuracy: 0.7500 - val_loss: 0.8006 - val_accuracy: 0.7500
Epoch 498/500
1/1 [==============================] - 0s 50ms/step - loss: 0.8181 - accuracy: 0.7500 - val_loss: 0.8004 - val_accuracy: 0.7500
Epoch 499/500
1/1 [==============================] - 0s 106ms/step - loss: 0.8178 - accuracy: 0.7500 - val_loss: 0.8001 - val_accuracy: 0.750
0
Epoch 500/500
1/1 [==============================] - 0s 40ms/step - loss: 0.8175 - accuracy: 0.7500 - val_loss: 0.7998 - val_accuracy: 0.7500
```



TESTING

Models in Keras are tested using the method **evaluate**. This method returns the classification accuracy on the training and the testing sets.

In [ ]:

```
loss, acc = model.evaluate(train_values, train_labels, verbose=1)

print("Training Set Accuracy: %f" %(acc))

loss, acc = model.evaluate(test_values, test_labels, verbose=1)

print("Testing Set Accuracy: %f" %(acc))
```

```
2/2 [==============================] - 0s 13ms/step - loss: 0.8154 - accuracy: 0.7500
2/2 [==============================] - 0s 13ms/step - loss: 0.8154 - accuracy: 0.7500
Training Set Accuracy: 0.750000
1/1 [==============================] - 0s 33ms/step - loss: 1.0503 - accuracy: 0.2857
Testing Set Accuracy: 0.285714
```

MAKING PREDICTIONS

The last step in a Regression Model is to make predictions for values not in the training set, which are determined by the method **predict**. In the following cell we print the Elements in the testing set, the real values for their Young's Moduli and the predictions generated by the Machine Learning model.

In [ ]:

```
train_predictions = model.predict(train_values)
test_predictions = model.predict(test_values)

all_labels = np.vstack((train_labels, test_labels))
all_predictions = np.vstack((train_predictions, test_predictions))

predicted_labels = []
true_labels = []

for i in range(all_predictions.shape[0]):
    if (np.argmax(all_predictions[i]) == 0): # np.argmax returns the index of maximum value along an axis.
        # Here we are looking for the value
        predicted_labels.append("FCC")
    if (np.argmax(all_labels[i]) == 0):
        true_labels.append("FCC")
    if (np.argmax(all_predictions[i]) == 1):
        predicted_labels.append("BCC")
    if (np.argmax(all_labels[i]) == 1):
        true_labels.append("BCC")
    if (np.argmax(all_predictions[i]) == 2):
        predicted_labels.append("HCP")
    if (np.argmax(all_labels[i]) == 2):
        true_labels.append("HCP")

predicted_labels = np.array(predicted_labels).reshape((-1, 1))
true_labels = np.array(true_labels).reshape((-1, 1))
headings = ["Atomic number", "True crystal structure", "Predicted crystal structure"]
```

```
atomic_number_array = np.array(df.iloc[:, 0]).reshape((-1, 1))
plot_table = np.concatenate((atomic_number_array, true_labels, predicted_labels), axis=1)

plot_df = pd.DataFrame(plot_table, columns=headings)
```

```
2/2 [==============================] - 0s 6ms/step
2/2 [==============================] - 0s 6ms/step
1/1 [==============================] - 0s 48ms/step
```

In [ ]:

```
plot_df
```

Out[ ]:

	Atomic number	True crystal structure	Predicted crystal structure
0	27	HCP	BCC
1	69	HCP	HCP
2	39	HCP	HCP
3	75	HCP	HCP
4	28	FCC	BCC
5	67	HCP	HCP
6	79	FCC	FCC
7	21	HCP	HCP
8	45	FCC	BCC
9	74	BCC	HCP
10	64	HCP	HCP
11	65	HCP	HCP
12	72	HCP	HCP
13	70	FCC	HCP
14	55	BCC	BCC
15	30	HCP	HCP
16	56	BCC	BCC
17	25	BCC	BCC
18	26	BCC	BCC
19	42	BCC	BCC
20	11	BCC	BCC
21	71	HCP	HCP

	Atomic number	True crystal structure	Predicted crystal structure
22	90	FCC	FCC
23	29	FCC	HCP
24	3	BCC	BCC
25	81	HCP	HCP
26	23	BCC	BCC
27	37	BCC	BCC
28	40	HCP	HCP
29	24	BCC	BCC
30	41	BCC	BCC
31	47	FCC	HCP
32	4	HCP	HCP
33	44	HCP	BCC
34	13	FCC	BCC
35	22	HCP	HCP
36	82	FCC	FCC
37	20	BCC	BCC
38	73	BCC	HCP
39	66	HCP	HCP
40	48	HCP	FCC
41	68	HCP	HCP
42	46	FCC	FCC
43	63	BCC	HCP

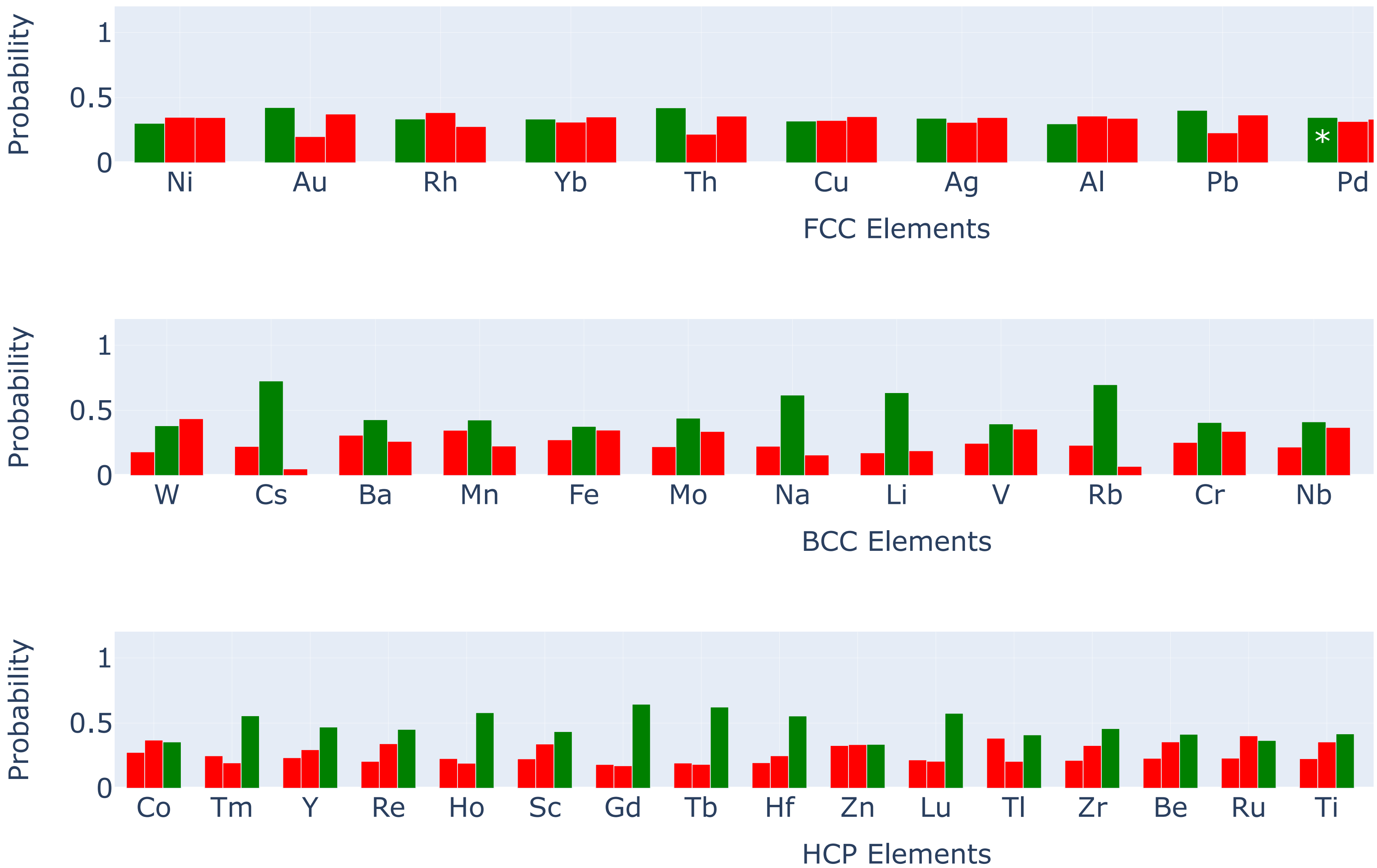
	Atomic number	True crystal structure	Predicted crystal structure
44	77	FCC	HCP
45	12	HCP	BCC
46	78	FCC	HCP

In [ ]:
```
crystal_structures = ["FCC", "BCC", "HCP"]
FCC_prediction = []
BCC_prediction = []
HCP_prediction = []

for item in range(len(all_predictions)):
    FCC_prediction.append(all_predictions[item].tolist()[0])
    BCC_prediction.append(all_predictions[item].tolist()[1])
    HCP_prediction.append(all_predictions[item].tolist()[2])

# -------------------------------------------------------------

# This block will be used to sort the elements by their atomic number

atomic_number = list(df.iloc[:, 0]) # From the Pandas Dataset
order = np.argsort(atomic_number) # Sorting Indexes

# Sorting the lists by the indexes
# elements = [elements[x] for x in order]
# FCC_prediction = [FCC_prediction[x] for x in order]
# BCC_prediction = [BCC_prediction[x] for x in order]
# HCP_prediction =[HCP_prediction[x] for x in order]

# # -------------------------------------------------------------
```

In [ ]:
```
import plotly as py
import plotly.graph_objs as go
from plotly.subplots import make_subplots
from plotly.offline import iplot

py.offline.init_notebook_mode(connected=True)
```

```
fig = make_subplots(rows=3, cols=1, vertical_spacing=0.2)

# ---------
fig.append_trace(go.Bar(x=[_ for _ in elements if _ in fcc_elements], y=[FCC_prediction[_] for _ in range(len(FCC_prediction))
                        text=['*'  if _ in elements[-7:] else None for _ in [_ for _ in elements if _ in fcc_elements]]), row=

fig.append_trace(go.Bar(x=[_ for _ in elements if _ in fcc_elements], y=[BCC_prediction[_] for _ in range(len(BCC_prediction))
fig.append_trace(go.Bar(x=[_ for _ in elements if _ in fcc_elements], y=[HCP_prediction[_] for _ in range(len(HCP_prediction))
# ---------

# ---------

fig.append_trace(go.Bar(x=[_ for _ in elements if _ in bcc_elements], y=[FCC_prediction[_] for _ in range(len(FCC_prediction))
fig.append_trace(go.Bar(x=[_ for _ in elements if _ in bcc_elements], y=[BCC_prediction[_] for _ in range(len(BCC_prediction))
                        text=['*'  if _ in elements[-7:] else None for _ in [_ for _ in elements if _ in bcc_elements]]), row=

fig.append_trace(go.Bar(x=[_ for _ in elements if _ in bcc_elements], y=[HCP_prediction[_] for _ in range(len(HCP_prediction))
# ---------

# ---------
fig.append_trace(go.Bar(x=[_ for _ in elements if _ in hcp_elements], y=[FCC_prediction[_] for _ in range(len(FCC_prediction))
fig.append_trace(go.Bar(x=[_ for _ in elements if _ in hcp_elements], y=[BCC_prediction[_] for _ in range(len(BCC_prediction))
fig.append_trace(go.Bar(x=[_ for _ in elements if _ in hcp_elements], y=[HCP_prediction[_] for _ in range(len(HCP_prediction))
                        text=['*'  if _ in elements[-7:] else None for _ in [_ for _ in elements if _ in hcp_elements]]), row=
# ---------


fig.update_xaxes(title=go.layout.xaxis.Title(text="FCC Elements", font=dict(size=18)),showgrid=True, tickfont=dict(size=18), r
fig.update_xaxes(title=go.layout.xaxis.Title(text="BCC Elements", font=dict(size=18)),showgrid=True, tickfont=dict(size=18), r
fig.update_xaxes(title=go.layout.xaxis.Title(text="HCP Elements", font=dict(size=18)),showgrid=True, tickfont=dict(size=18), r

fig.update_yaxes(title=go.layout.yaxis.Title(text="Probability", font=dict(size=18)),showgrid=True, tickfont=dict(size=18),ran
fig.update_yaxes(title=go.layout.yaxis.Title(text="Probability", font=dict(size=18)),showgrid=True, tickfont=dict(size=18),ran
fig.update_yaxes(title=go.layout.yaxis.Title(text="Probability", font=dict(size=18)),showgrid=True, tickfont=dict(size=18),ran




fig.update_layout(height=700, width=1200, barmode='group', bargap=0.3)
```

```
fig.show()
```



Probability
1
0.5
0
Ni
Au
Rh
Yb
Th
Cu
Ag
Al
Pb
Pd
*
FCC Elements
Probability
1
0.5
0
W
Cs
Ba
Mn
Fe
Mo
Na
Li
V
Rb
Cr
Nb
BCC Elements
Probability
1
0.5
0
Co
Tm
Y
Re
Ho
Sc
Gd
Tb
Hf
Zn
Lu
Tl
Zr
Be
Ru
Ti
HCP Elements

In [ ]:

```

```