In [3]:
```
%%shell
jupyter nbconvert --to html /content/Student_performance.ipynb
```

Problem Statement

You are a data scientist working for a school

You are asked to predict the GPA of the current students based on the following provided data:

0 StudentID int64

1 Age int64

2 Gender int64

3 Ethnicity int64

4 ParentalEducation int64

5 StudyTimeWeekly float64 6 Absences int64

7 Tutoring int64

8 ParentalSupport int64

9 Extracurricular int64

10 Sports int64

11 Music int64

12 Volunteering int64

13 GPA float64 14 GradeClass float64

The GPA is the Grade Point Average, typically ranges from 0.0 to 4.0 in most educational systems, with 4.0 representing an 'A' or excellent performance.

The minimum passing GPA can vary by institution, but it's often around 2.0. This usually corresponds to a 'C' grade, which is considered satisfactory.

You need to create a Deep Learning model capable to predict the GPA of a Student based on a set of provided features. The data provided represents 2,392 students.

In this excersice you will be requested to create a total of three models and select the most performant one.

1) Import Libraries

First let's import the following libraries, if there is any library that you need and is not in the list bellow feel free to include it

In [4]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import tensorflow as tf
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense
from tensorflow.keras.layers import Dropout
```

```
from tensorflow.keras.layers import Conv1D, MaxPooling1D, Flatten
from tensorflow.keras.regularizers import l2
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
```

2) Load Data

- You will be provided with a cvs (comma separated value) file.
- You will need to add that file into a pandas dataframe, you can use the following code as reference
- The file will be available in canvas

In [5]:
```
data = pd.read_csv("Student_performance_data _.csv")
data
```

Out[5]:

	StudentID	Age	Gender	Ethnicity	ParentalEducation	StudyTimeWeekly	Absences	Tutoring	I
0	1001	17	1	0	2	19.833723	7	1	
1	1002	18	0	0	1	15.408756	0	0	
2	1003	15	0	2	3	4.210570	26	0	
3	1004	17	1	0	3	10.028829	14	0	
4	1005	17	1	0	2	4.672495	17	1	
...	...	...	...	...	...	...	...	...	
2387	3388	18	1	0	3	10.680555	2	0	
2388	3389	17	0	0	1	7.583217	4	1	
2389	3390	16	1	0	2	6.805500	20	0	
2390	3391	16	1	1	0	12.416653	17	0	
2391	3392	16	1	0	2	17.819907	13	0	

2392 rows × 15 columns

3) Review you data:

Make sure you review your data. Place special attention of null or empty values.

In [6]:
```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2392 entries, 0 to 2391
Data columns (total 15 columns):
 #   Column             Non-Null Count  Dtype  
---  ------             --------------  -----  
 0   StudentID          2392 non-null   int64  
 1   Age                2392 non-null   int64  
 2   Gender             2392 non-null   int64  
 3   Ethnicity          2392 non-null   int64  
 4   ParentalEducation  2392 non-null   int64  
 5   StudyTimeWeekly    2392 non-null   float64
 6   Absences           2392 non-null   int64  
 7   Tutoring           2392 non-null   int64  
 8   ParentalSupport    2392 non-null   int64  
 9   Extracurricular    2392 non-null   int64  
 10  Sports             2392 non-null   int64  
 11  Music              2392 non-null   int64  
 12  Volunteering       2392 non-null   int64  
 13  GPA                2392 non-null   float64
 14  GradeClass         2392 non-null   float64
dtypes: float64(3), int64(12)
memory usage: 280.4 KB
```

4. Remove the columns not needed for Student performance prediction

- Choose only the columns you consider to be valuable for your model training.
- For example, StudentID might not be a good feature for your model, and thus should be removed from your main dataset, which other columns should also be removed?
- You can name that final dataset as 'dataset'

In [7]:

```
columns_to_remove = ['StudentID']
dataset = data.drop(columns=columns_to_remove)

dataset.head()
```

Out[7]:

	Age	Gender	Ethnicity	ParentalEducation	StudyTimeWeekly	Absences	Tutoring	ParentalSupport
0	17	1	0	2	19.833723	7	1	2
1	18	0	0	1	15.408756	0	0	1
2	15	0	2	3	4.210570	26	0	2
3	17	1	0	3	10.028829	14	0	3
4	17	1	0	2	4.672495	17	1	3

In [8]:

```
final_dataset = dataset.drop(columns=['Gender', 'Ethnicity','Volunteering'])

final_dataset.head()
```

Out[8]:

	Age	ParentalEducation	StudyTimeWeekly	Absences	Tutoring	ParentalSupport	Extracurricular	S
0	17	2	19.833723	7	1	2	0	
1	18	1	15.408756	0	0	1	0	
2	15	3	4.210570	26	0	2	0	
3	17	3	10.028829	14	0	3	1	
4	17	2	4.672495	17	1	3	0	

5. Check if the columns has any null values:

- Here you now have your final dataset to use in your model training.
- Before moving foward review your data check for any null or empty value that might be needed to be removed

In [9]:

```
missing_values = final_dataset.isnull().sum()

missing_values
```

Out[9]:

	0
Age	0
ParentalEducation	0
StudyTimeWeekly	0
Absences	0
Tutoring	0
ParentalSupport	0
Extracurricular	0
Sports	0
Music	0
GPA	0
GradeClass	0

dtype: int64

6. Prepare your data for training and for testing set:

- First create a dataset named X, with all columns but GPA. These are the features
- Next create another dataset named y, with only GPA column. This is the label
- If you go to your Imports, you will see the following import: **'from sklearn.model_selection import train_test_split'**

- Use that *train_test_split* function to create: X_train, X_test, y_train and y_test respectively. Use X and y datasets as parameters. Other parameters to use are: Test Size = 0.2, Random State = 42.

- Standarize your features (X_train and X_test) by using the StandardScaler (investigate how to use fit_transform and transform functions). This will help the training process by dealing with normilized data.

  Note: Your X_train shape should be around (1913, 10). This means the dataset has 10 columns which should be the input.

In [10]:
```
X = final_dataset.drop(columns=['GPA'])

y = final_dataset['GPA']

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=

X_train_shape = X_train.shape
X_test_shape = X_test.shape

scaler = StandardScaler()
X_train_scaled = scaler.fit_transform(X_train)
X_test_scaled = scaler.transform(X_test)

X_train_shape, X_test_shape
```

Out[10]:
```
((1913, 10), (479, 10))
```

7. Define your Deep Neural Network.

- This will be a Sequential Neural Network.
- With a Dense input layer with 64 units, and input dimention of 10 and Relu as the activation function.
- A Dense hidden layer with 32 units, and Relu as the activation function.
- And a Dense output layer with 1 unit, do not define an activation function so it defaults to linear, suitable for regression tasks. e.g. Dense(1)

This last part of the output layer is super important, since we want to predict the GPA, this means that we want a regression and not a classification. Linear activation function is best for regression and Sigmoid is best for Binary Classification

In [11]:
```
model = Sequential()

model.add(Dense(64, input_dim=10, activation='relu'))

model.add(Dense(32, activation='relu'))

model.add(Dense(1))

model.compile(optimizer='adam', loss='mean_squared_error')

model.summary()
```

```
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87: UserWarnin
g: Do not pass an `input_shape`/`input_dim` argument to a layer. When using Sequentia
l models, prefer using an `Input(shape)` object as the first layer in the model inste
ad.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
```

Model: "sequential"

Layer (type)	Output Shape	Pa
dense (Dense)	(None, 64)	
dense_1 (Dense)	(None, 32)	
dense_2 (Dense)	(None, 1)	

Total params: 2,817 (11.00 KB)

Trainable params: 2,817 (11.00 KB)

Non-trainable params: 0 (0.00 B)

8. Compile your Neural Network

- Choose Adam as the optimizer
- And MSE as the Loss function
- Also add the following metrics: Mean Absolute Error

In [12]:
```python
model.compile(optimizer='adam',
              loss='mean_squared_error',
              metrics=['mean_absolute_error'])

model.summary()
```

Model: "sequential"

Layer (type)	Output Shape	Pa
dense (Dense)	(None, 64)	
dense_1 (Dense)	(None, 32)	
dense_2 (Dense)	(None, 1)	

Total params: 2,817 (11.00 KB)

Trainable params: 2,817 (11.00 KB)

Non-trainable params: 0 (0.00 B)

9. Fit (or train) your model

- Use the X_train and y_train datasets for the training

- Do 50 data iterations
- Choose the batch size = 10
- Also select a validation_split of 0.2
- Save the result of the fit function in a variable called 'history'

In [13]:
```
history = model.fit(X_train_scaled, y_train,
                    epochs=50,
                    batch_size=10,
                    validation_split=0.2)
```

```
Epoch 1/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 3s 6ms/step - loss: 1.3757 - mean_absolute_error: 0.8708
- val_loss: 0.1178 - val_mean_absolute_error: 0.2791
Epoch 2/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - loss: 0.0999 - mean_absolute_error: 0.2548
- val_loss: 0.0727 - val_mean_absolute_error: 0.2172
Epoch 3/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - loss: 0.0664 - mean_absolute_error: 0.2061
- val_loss: 0.0570 - val_mean_absolute_error: 0.1931
Epoch 4/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - loss: 0.0537 - mean_absolute_error: 0.1889
- val_loss: 0.0546 - val_mean_absolute_error: 0.1875
Epoch 5/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - loss: 0.0472 - mean_absolute_error: 0.1738
- val_loss: 0.0509 - val_mean_absolute_error: 0.1782
Epoch 6/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - loss: 0.0384 - mean_absolute_error: 0.1593
- val_loss: 0.0479 - val_mean_absolute_error: 0.1745
Epoch 7/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - loss: 0.0359 - mean_absolute_error: 0.1537
- val_loss: 0.0424 - val_mean_absolute_error: 0.1655
Epoch 8/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - loss: 0.0332 - mean_absolute_error: 0.1457
- val_loss: 0.0461 - val_mean_absolute_error: 0.1739
Epoch 9/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0345 - mean_absolute_error: 0.1487
- val_loss: 0.0436 - val_mean_absolute_error: 0.1634
Epoch 10/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - loss: 0.0345 - mean_absolute_error: 0.1492
- val_loss: 0.0434 - val_mean_absolute_error: 0.1672
Epoch 11/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - loss: 0.0290 - mean_absolute_error: 0.1349
- val_loss: 0.0428 - val_mean_absolute_error: 0.1641
Epoch 12/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - loss: 0.0299 - mean_absolute_error: 0.1380
- val_loss: 0.0539 - val_mean_absolute_error: 0.1839
Epoch 13/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - loss: 0.0293 - mean_absolute_error: 0.1357
- val_loss: 0.0431 - val_mean_absolute_error: 0.1636
Epoch 14/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - loss: 0.0296 - mean_absolute_error: 0.1366
- val_loss: 0.0455 - val_mean_absolute_error: 0.1674
Epoch 15/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - loss: 0.0280 - mean_absolute_error: 0.1318
- val_loss: 0.0441 - val_mean_absolute_error: 0.1647
Epoch 16/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0255 - mean_absolute_error: 0.1266
- val_loss: 0.0472 - val_mean_absolute_error: 0.1714
Epoch 17/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - loss: 0.0249 - mean_absolute_error: 0.1244
- val_loss: 0.0431 - val_mean_absolute_error: 0.1635
Epoch 18/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - loss: 0.0261 - mean_absolute_error: 0.1288
- val_loss: 0.0465 - val_mean_absolute_error: 0.1670
Epoch 19/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - loss: 0.0246 - mean_absolute_error: 0.1259
- val_loss: 0.0439 - val_mean_absolute_error: 0.1640
Epoch 20/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - loss: 0.0246 - mean_absolute_error: 0.1236
- val_loss: 0.0488 - val_mean_absolute_error: 0.1701
```

```
Epoch 21/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - loss: 0.0252 - mean_absolute_error: 0.1261
- val_loss: 0.0472 - val_mean_absolute_error: 0.1705
Epoch 22/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 4ms/step - loss: 0.0238 - mean_absolute_error: 0.1220
- val_loss: 0.0443 - val_mean_absolute_error: 0.1623
Epoch 23/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 3ms/step - loss: 0.0241 - mean_absolute_error: 0.1245
- val_loss: 0.0473 - val_mean_absolute_error: 0.1703
Epoch 24/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - loss: 0.0243 - mean_absolute_error: 0.1238
- val_loss: 0.0422 - val_mean_absolute_error: 0.1606
Epoch 25/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - loss: 0.0225 - mean_absolute_error: 0.1179
- val_loss: 0.0458 - val_mean_absolute_error: 0.1708
Epoch 26/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - loss: 0.0230 - mean_absolute_error: 0.1188
- val_loss: 0.0467 - val_mean_absolute_error: 0.1671
Epoch 27/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0216 - mean_absolute_error: 0.1168
- val_loss: 0.0473 - val_mean_absolute_error: 0.1709
Epoch 28/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - loss: 0.0205 - mean_absolute_error: 0.1137
- val_loss: 0.0460 - val_mean_absolute_error: 0.1679
Epoch 29/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - loss: 0.0222 - mean_absolute_error: 0.1177
- val_loss: 0.0504 - val_mean_absolute_error: 0.1745
Epoch 30/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - loss: 0.0221 - mean_absolute_error: 0.1171
- val_loss: 0.0440 - val_mean_absolute_error: 0.1621
Epoch 31/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - loss: 0.0211 - mean_absolute_error: 0.1144
- val_loss: 0.0440 - val_mean_absolute_error: 0.1625
Epoch 32/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - loss: 0.0219 - mean_absolute_error: 0.1173
- val_loss: 0.0451 - val_mean_absolute_error: 0.1673
Epoch 33/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - loss: 0.0222 - mean_absolute_error: 0.1177
- val_loss: 0.0463 - val_mean_absolute_error: 0.1685
Epoch 34/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - loss: 0.0188 - mean_absolute_error: 0.1094
- val_loss: 0.0432 - val_mean_absolute_error: 0.1603
Epoch 35/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - loss: 0.0207 - mean_absolute_error: 0.1135
- val_loss: 0.0563 - val_mean_absolute_error: 0.1883
Epoch 36/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - loss: 0.0189 - mean_absolute_error: 0.1096
- val_loss: 0.0488 - val_mean_absolute_error: 0.1744
Epoch 37/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - loss: 0.0201 - mean_absolute_error: 0.1114
- val_loss: 0.0474 - val_mean_absolute_error: 0.1687
Epoch 38/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - loss: 0.0200 - mean_absolute_error: 0.1108
- val_loss: 0.0461 - val_mean_absolute_error: 0.1667
Epoch 39/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - loss: 0.0210 - mean_absolute_error: 0.1177
- val_loss: 0.0452 - val_mean_absolute_error: 0.1635
Epoch 40/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 3ms/step - loss: 0.0184 - mean_absolute_error: 0.1056
- val_loss: 0.0459 - val_mean_absolute_error: 0.1639
```

```
Epoch 41/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - loss: 0.0198 - mean_absolute_error: 0.1096
- val_loss: 0.0480 - val_mean_absolute_error: 0.1678
Epoch 42/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - loss: 0.0195 - mean_absolute_error: 0.1120
- val_loss: 0.0463 - val_mean_absolute_error: 0.1685
Epoch 43/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - loss: 0.0180 - mean_absolute_error: 0.1062
- val_loss: 0.0440 - val_mean_absolute_error: 0.1632
Epoch 44/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 4ms/step - loss: 0.0184 - mean_absolute_error: 0.1071
- val_loss: 0.0464 - val_mean_absolute_error: 0.1673
Epoch 45/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - loss: 0.0172 - mean_absolute_error: 0.1045
- val_loss: 0.0445 - val_mean_absolute_error: 0.1643
Epoch 46/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0173 - mean_absolute_error: 0.1044
- val_loss: 0.0448 - val_mean_absolute_error: 0.1627
Epoch 47/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - loss: 0.0174 - mean_absolute_error: 0.1032
- val_loss: 0.0482 - val_mean_absolute_error: 0.1705
Epoch 48/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - loss: 0.0172 - mean_absolute_error: 0.1033
- val_loss: 0.0451 - val_mean_absolute_error: 0.1647
Epoch 49/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - loss: 0.0167 - mean_absolute_error: 0.1027
- val_loss: 0.0531 - val_mean_absolute_error: 0.1806
Epoch 50/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0176 - mean_absolute_error: 0.1040
- val_loss: 0.0490 - val_mean_absolute_error: 0.1746
```

10. View your history variable:

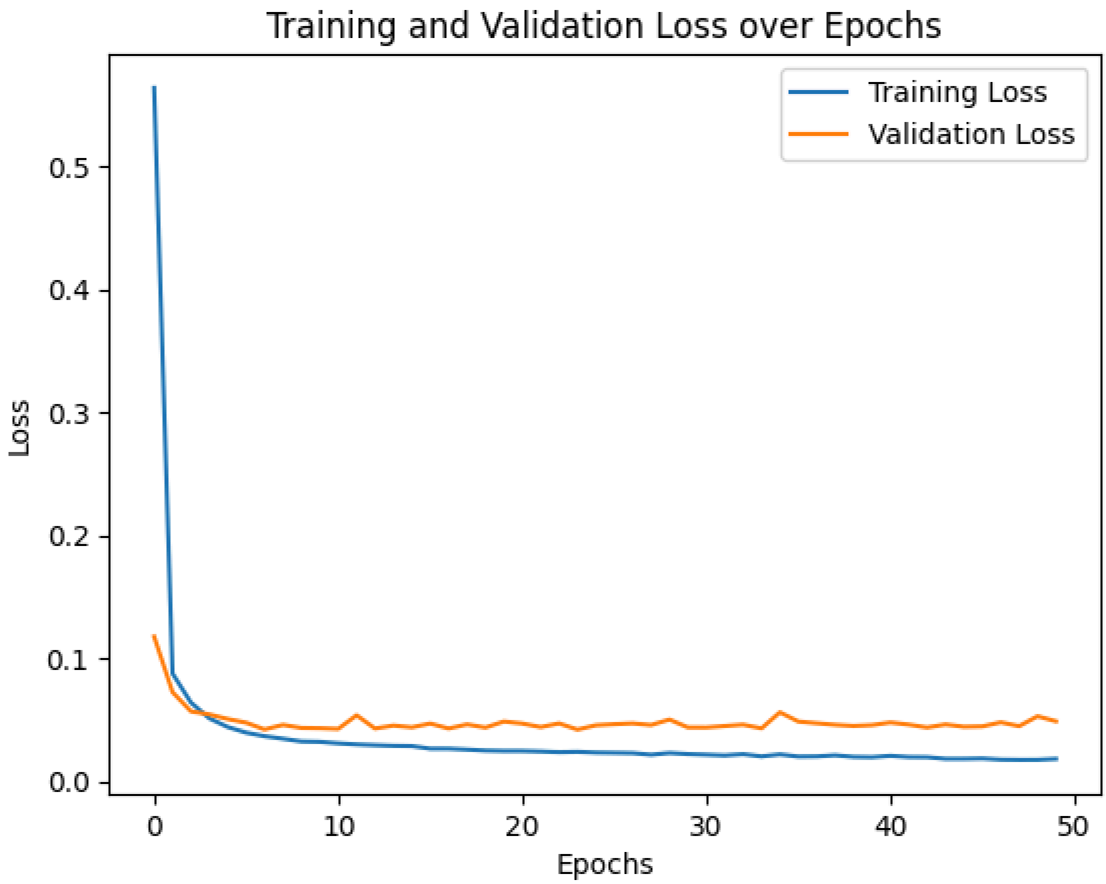
- Use Matplotlib.pyplot to show graphs of your model traning history
- In one graph:
  - Plot the Training Loss and the Validation Loss
  - X Label = Epochs
  - Y Label = Loss
  - Title = Training and Validation Loss over Epochs
- In a second graph:
  - Plot the Training MAE and the Validation MAE
  - X Label = Epochs
  - Y Label = Mean Absolute Error (MAE)
  - Title = Training and Validation MAE over Epochs

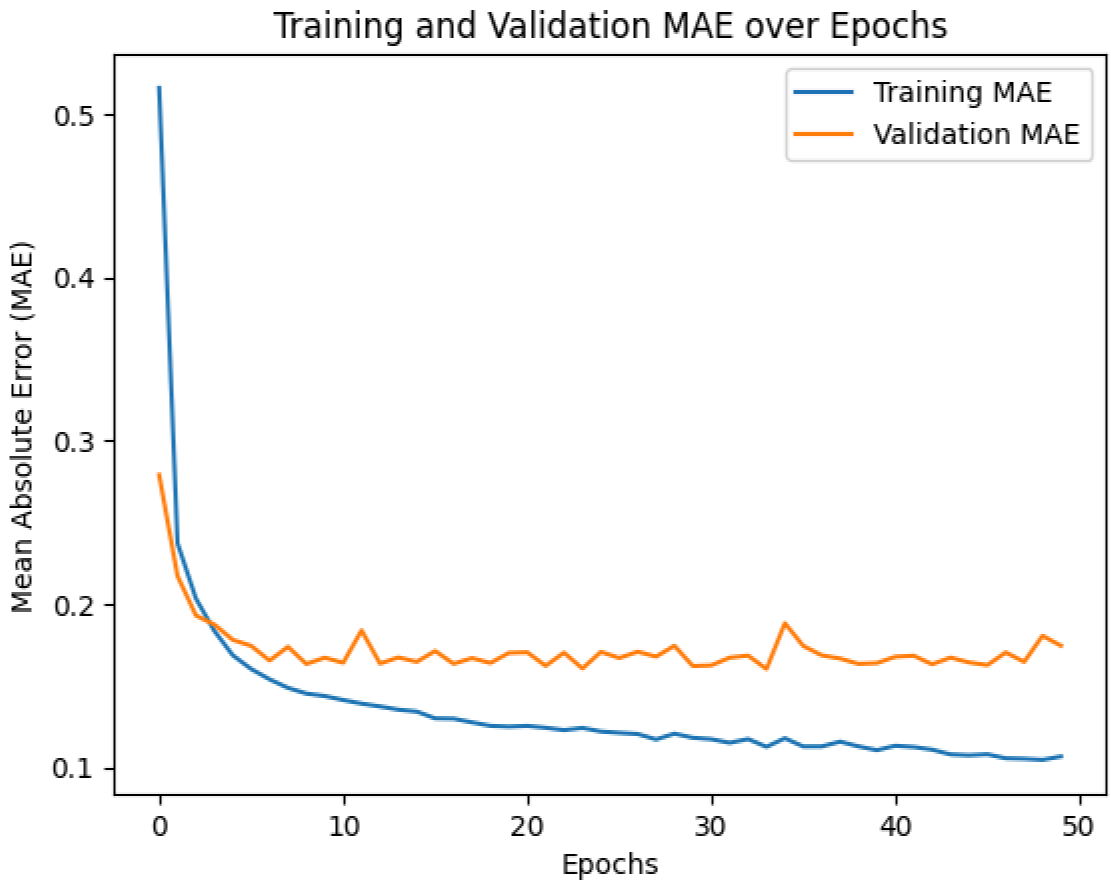
In [14]:

```python
plt.plot(history.history['loss'], label='Training Loss')
plt.plot(history.history['val_loss'], label='Validation Loss')
plt.xlabel('Epochs')
plt.ylabel('Loss')
plt.title('Training and Validation Loss over Epochs')
plt.legend()
plt.show()

plt.plot(history.history['mean_absolute_error'], label='Training MAE')
plt.plot(history.history['val_mean_absolute_error'], label='Validation MAE')
```

```
plt.xlabel('Epochs')
plt.ylabel('Mean Absolute Error (MAE)')
plt.title('Training and Validation MAE over Epochs')
plt.legend()
plt.show()
```





11. Evaluate your model:

- See the result of your loss function.
- What can you deduct from there?

In [15]:
```
loss, mae = model.evaluate(X_test_scaled, y_test)
print(f'Loss: {loss}, Mean Absolute Error: {mae}')
```

```
15/15 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0475 - mean_absolute_error: 0.1703
Loss: 0.045884568244218826, Mean Absolute Error: 0.16879822313785553
```

12. Use your model to make some predictions:

- Make predictions of your X_test dataset
- Print the each of the predictions and the actual value (which is in y_test)
- How good was your model?

In [19]:
```
y_pred = model.predict(X_test_scaled)

for i in range(len(y_test)):
    print(f"Predicted: {y_pred[i][0]}, Actual: {y_test.iloc[i]}")
```

```
15/15 ━━━━━━━━━━━━━━━━━━━━ 0s 7ms/step
Predicted: 1.434401035308838, Actual: 1.4277243762746905
Predicted: 2.8287594318389893, Actual: 3.117354434785501
Predicted: 1.7671840190887451, Actual: 2.037768574636005
Predicted: 3.5548267364501953, Actual: 3.5485205508668662
Predicted: 0.6627643704414368, Actual: 0.2489771312307257
Predicted: 2.522404909133911, Actual: 2.627693905554347
Predicted: 1.6261200904846191, Actual: 2.057378500596372
Predicted: 2.2186262607574463, Actual: 2.248337588471201
Predicted: 2.1604838371276855, Actual: 2.1947065208246226
Predicted: 1.042617678642273, Actual: 0.7581829737450007
Predicted: 2.796381950378418, Actual: 2.370893096932428
Predicted: 0.6036919355392456, Actual: 0.7664048694920337
Predicted: 2.7774555683135986, Actual: 2.952721567213245
Predicted: 2.6910226345062256, Actual: 2.3433313526833226
Predicted: 2.711111545562744, Actual: 2.7718106588704914
Predicted: 0.23863865435123444, Actual: 0.2878673233291232
Predicted: 1.135642170906067, Actual: 1.0182646498699195
Predicted: 1.5282914638519287, Actual: 1.629355895809393
Predicted: 2.2549798488616943, Actual: 2.0744387503601613
Predicted: 2.4520339965820312, Actual: 2.423800751639832
Predicted: 1.9141418933868408, Actual: 1.7562115530004156
Predicted: 1.6901865005493164, Actual: 1.5662885180613493
Predicted: 1.7320659160614014, Actual: 1.7062124885863237
Predicted: 3.1103603839874268, Actual: 3.161436270258364
Predicted: 1.7054874897003174, Actual: 1.733364046560005
Predicted: 0.4939981698989868, Actual: 0.8419632253726905
Predicted: 1.6261416673660278, Actual: 1.3791671997209602
Predicted: 2.675720453262329, Actual: 3.026983310961493
Predicted: 2.1112916469573975, Actual: 2.191998419606377
Predicted: 1.7739057540893555, Actual: 2.315769874969324
Predicted: 2.02156662940979, Actual: 2.068111784968204
Predicted: 0.6155768036842346, Actual: 0.869123386308555
Predicted: 2.8575026988983154, Actual: 2.900096239205548
Predicted: 3.1621928215026855, Actual: 3.468581349135728
Predicted: 1.496173620223999, Actual: 1.5674124377048069
Predicted: 1.7900030612945557, Actual: 1.7946671055341392
Predicted: 3.1724724769592285, Actual: 3.1813076022771107
Predicted: 2.858459711074829, Actual: 2.8973550040674096
Predicted: 3.147073984146118, Actual: 3.2448822032661777
Predicted: 0.7472177147865295, Actual: 0.3578088919508027
Predicted: 2.793205976486206, Actual: 2.6523548127186087
Predicted: 3.6052398681640625, Actual: 3.680961344427839
Predicted: 0.9261228442192078, Actual: 1.0363787383257312
Predicted: 2.215752363204956, Actual: 2.017218038843316
Predicted: 0.8922378420829773, Actual: 0.9633750092514732
Predicted: 2.4408156871795654, Actual: 2.23946398594873
Predicted: 2.77579665184021, Actual: 2.735960967147571
Predicted: 1.2090227603912354, Actual: 1.3619328272119078
Predicted: 3.0705392360687256, Actual: 2.70295861751592
Predicted: 1.2638720273971558, Actual: 1.441918756451196
Predicted: 3.0492827892303467, Actual: 3.219531247903172
Predicted: 3.020312547683716, Actual: 3.339094362200312
Predicted: 1.4384541511535645, Actual: 1.556218796080208
Predicted: 1.1068354845046997, Actual: 1.3423871779577343
Predicted: 1.7711048126220703, Actual: 1.756186167808898
Predicted: 3.373246908187866, Actual: 3.421837670026282
Predicted: 2.544795274734497, Actual: 2.3169599744650875
Predicted: 3.3455865383148193, Actual: 3.286585133610396
Predicted: 1.0808054208755493, Actual: 0.684651926072042
```

```
Predicted: 2.10282039642334, Actual: 2.1370681063433614
Predicted: 1.7453349828720093, Actual: 1.6884896091579062
Predicted: 1.775601863861084, Actual: 2.1962173498725552
Predicted: 2.339693784713745, Actual: 2.4777903742896354
Predicted: 1.4846220016479492, Actual: 1.1979333311054492
Predicted: 1.1393500566482544, Actual: 0.9881530769380218
Predicted: 1.9596748352050781, Actual: 1.8589248556209292
Predicted: 3.0572032928466797, Actual: 3.040729612939948
Predicted: 2.294420003890991, Actual: 2.3741309720150148
Predicted: 1.1449859142303467, Actual: 1.2212994205081906
Predicted: 3.348600387573242, Actual: 3.2742334588098787
Predicted: 3.4258930683135986, Actual: 3.5451594318003106
Predicted: 0.837295413017273, Actual: 1.101939932271634
Predicted: 2.9309537410736084, Actual: 2.968807133269986
Predicted: 2.5167155265808105, Actual: 2.5761746968185832
Predicted: 0.8214935660362244, Actual: 0.4541144380742554
Predicted: 2.690354824066162, Actual: 2.6741625507699083
Predicted: 1.9990085363388062, Actual: 2.05996665628182
Predicted: 1.761317491531372, Actual: 1.983133992222757
Predicted: 0.37076425552368164, Actual: 0.2123670952038988
Predicted: 1.4323649406433105, Actual: 1.279370611476852
Predicted: 2.060152053833008, Actual: 2.029736582447598
Predicted: 2.8940012454986572, Actual: 3.189217169770364
Predicted: 2.100888729095459, Actual: 2.2564883117007817
Predicted: 1.4926873445510864, Actual: 1.636403110402885
Predicted: 1.5755939483642578, Actual: 1.4015509339508203
Predicted: 0.6164355874061584, Actual: 0.4436406552124723
Predicted: 3.7982048988342285, Actual: 4.0
Predicted: 1.1243520975112915, Actual: 1.2712295298373406
Predicted: 1.4107937812805176, Actual: 1.3026414038177534
Predicted: 1.5547125339508057, Actual: 1.5970165723817666
Predicted: 2.4614484310150146, Actual: 2.5353912836886785
Predicted: 1.2290183305740356, Actual: 1.382935704905427
Predicted: 0.9283017516136169, Actual: 1.0384883524708666
Predicted: 0.6668896079063416, Actual: 0.1306542378736745
Predicted: 1.5966777801513672, Actual: 1.2738181762310288
Predicted: 1.2264760732650757, Actual: 1.2714946229661936
Predicted: 3.124021530151367, Actual: 3.3250668431789085
Predicted: 1.0921810865402222, Actual: 1.0707079518820537
Predicted: 1.6330459117889404, Actual: 1.574306088258861
Predicted: 1.6892263889312744, Actual: 1.6336793710032238
Predicted: 0.5535646080970764, Actual: 0.0
Predicted: 2.221374273300171, Actual: 2.122102757166825
Predicted: 1.379076361656189, Actual: 1.2912304259430896
Predicted: 1.6088178157806396, Actual: 1.4547228982036302
Predicted: 0.8596975207328796, Actual: 0.7742848136453381
Predicted: 2.1862971782684326, Actual: 2.293087191388992
Predicted: 2.7408666610717773, Actual: 2.6827762879633723
Predicted: 1.3048464059829712, Actual: 1.3154643711031746
Predicted: 1.46249520778656, Actual: 1.4693892718306072
Predicted: 0.748594343662262, Actual: 0.61316598823277
Predicted: 1.5529158115386963, Actual: 1.8819787410279116
Predicted: 2.0489065647125244, Actual: 2.268905391743987
Predicted: 2.984683036804199, Actual: 3.545153035874012
Predicted: 0.9132096171379089, Actual: 0.4258137609932144
Predicted: 0.28936630487442017, Actual: 0.5403751565523403
Predicted: 1.3423526287078857, Actual: 1.1110684623439582
Predicted: 2.150566339492798, Actual: 2.035367286041147
Predicted: 2.513868570327759, Actual: 2.995458295719109
Predicted: 0.8954193592071533, Actual: 1.2178731513333656
```

```
Predicted: 1.8968169689178467, Actual: 2.1269889706667486
Predicted: 0.4747321605682373, Actual: 0.5297986094952745
Predicted: 3.08994197845459, Actual: 3.0168978681026624
Predicted: 3.5386054515838623, Actual: 3.5921289778598062
Predicted: 1.074942708015442, Actual: 1.0583113523259429
Predicted: 1.9012229442596436, Actual: 1.8425841347566336
Predicted: 1.0008982419967651, Actual: 1.254572618142762
Predicted: 2.5888426303863525, Actual: 2.794885282534225
Predicted: 0.9569627642631531, Actual: 1.3478638291935137
Predicted: 2.1357595920562744, Actual: 2.226983393791037
Predicted: 2.7624869346618652, Actual: 2.81347034515229
Predicted: 1.034543752670288, Actual: 1.1055870981431069
Predicted: 3.86680269241333, Actual: 4.0
Predicted: 2.800910711288452, Actual: 2.898151345753362
Predicted: 0.6685284376144409, Actual: 0.4239919484053893
Predicted: 0.6261377334594727, Actual: 0.691477418739295
Predicted: 1.019150733947754, Actual: 0.930028645913522
Predicted: 2.034313440322876, Actual: 2.330783482945125
Predicted: 0.20446325838565826, Actual: 0.3375541414467983
Predicted: 0.24006818234920502, Actual: 0.5022261210258885
Predicted: 1.1756112575531006, Actual: 0.9893299205987556
Predicted: 2.5865180492401123, Actual: 2.745858798383193
Predicted: 1.6322667598724365, Actual: 1.8041546917762192
Predicted: 3.0064902305603027, Actual: 3.3432781941994127
Predicted: 1.6992199420928955, Actual: 1.5730510214028783
Predicted: 2.756789445877075, Actual: 2.964009263152522
Predicted: 1.034479022026062, Actual: 1.1936177713396423
Predicted: 1.482954978942871, Actual: 1.3492561219436314
Predicted: 2.4208359718322754, Actual: 2.520175258554297
Predicted: 2.85603928565979, Actual: 3.2537859623175915
Predicted: 1.493833065032959, Actual: 1.4633796485356
Predicted: 3.0396621227264404, Actual: 3.161961511352394
Predicted: 0.8501110076904297, Actual: 0.7696361214560695
Predicted: 3.347162961959839, Actual: 3.5037432645192648
Predicted: 1.5099838972091675, Actual: 1.2687721230682183
Predicted: 0.7181736826896667, Actual: 1.1708122846030742
Predicted: 0.5678004622459412, Actual: 0.4712831088434319
Predicted: 1.3908019065856934, Actual: 1.3978971899557189
Predicted: 0.7272745370864868, Actual: 0.9689861311254953
Predicted: 1.4453941583633423, Actual: 1.511276997165948
Predicted: 0.37794244289398193, Actual: 0.5242135665585334
Predicted: 1.5543091297149658, Actual: 1.4913821292254057
Predicted: 1.2134532928466797, Actual: 1.519441725815139
Predicted: 0.8352088332176208, Actual: 0.7132676650608875
Predicted: 0.7294268012046814, Actual: 0.4005694786322673
Predicted: 0.3471962809562683, Actual: 0.5813557994949716
Predicted: 0.6785313487052917, Actual: 0.4050526130408124
Predicted: 0.9183739423751831, Actual: 1.1544100089135072
Predicted: 1.2650227546691895, Actual: 1.150153548429549
Predicted: 2.2696268558502197, Actual: 2.45790140704778
Predicted: 3.53698992729187, Actual: 3.4555094110343934
Predicted: 1.5406439304351807, Actual: 1.4414095429135232
Predicted: 3.443507432937622, Actual: 3.2879731984208025
Predicted: 2.90193772315979, Actual: 3.043931624402609
Predicted: 1.1569586992263794, Actual: 0.189818059909026
Predicted: 1.367412805557251, Actual: 1.1657871621526066
Predicted: 2.5589897632598877, Actual: 2.565368052695248
Predicted: 2.228776693344116, Actual: 2.2346855491679
Predicted: 1.1191612482070923, Actual: 1.25865485443205
Predicted: 2.272770643234253, Actual: 2.116723382234456
```

```
Predicted: 1.492063045501709, Actual: 1.674676218445066
Predicted: 2.906343698501587, Actual: 2.8211081074612654
Predicted: 1.7563567161560059, Actual: 1.9961073864422316
Predicted: 0.8155516386032104, Actual: 0.8090678531539729
Predicted: 2.960014581680298, Actual: 2.546634577748015
Predicted: 3.143364429473877, Actual: 2.735041610764964
Predicted: 2.695948362350464, Actual: 2.7535902317989964
Predicted: 3.1853299140930176, Actual: 3.283916521897384
Predicted: 1.6526999473571777, Actual: 1.2878189977208636
Predicted: 0.21819238364696503, Actual: 0.3106695325979863
Predicted: 1.9213175773620605, Actual: 1.83151128144078
Predicted: 1.2741365432739258, Actual: 0.911568270646688
Predicted: 2.2275636196136475, Actual: 2.109686328016939
Predicted: 1.4025145769119263, Actual: 1.0444999950877136
Predicted: 0.6613853573799133, Actual: 0.9654824453045304
Predicted: 1.7494280338287354, Actual: 1.5504493308923313
Predicted: 1.2107150554656982, Actual: 1.223828264622006
Predicted: 1.5460931062698364, Actual: 1.8934409155419147
Predicted: 1.9339990615844727, Actual: 2.295362280360723
Predicted: 2.6647744178771973, Actual: 2.8393132491560324
Predicted: 2.3079140186309814, Actual: 2.251847104187132
Predicted: 1.5850903987884521, Actual: 1.4053156279015715
Predicted: 3.0066754817962646, Actual: 3.03210533252155
Predicted: 0.7055887579917908, Actual: 0.5694198849254213
Predicted: 1.9111199378967285, Actual: 2.220050546998637
Predicted: 2.888082981109619, Actual: 2.7373779261627598
Predicted: 1.862846851348877, Actual: 2.340811012860678
Predicted: 0.6458139419555664, Actual: 0.5601746057023475
Predicted: 2.8411529064178467, Actual: 2.913887649138132
Predicted: 2.1976115703582764, Actual: 2.259451153743707
Predicted: 1.4923804998397827, Actual: 1.7419681672894356
Predicted: 1.6639337539672852, Actual: 1.2658914451513914
Predicted: 1.02984619140625, Actual: 1.298729106820987
Predicted: 2.019035577774048, Actual: 1.9137997969013387
Predicted: 2.7364537715911865, Actual: 2.7808169956104924
Predicted: 2.5117905139923096, Actual: 2.553484846074773
Predicted: 0.9206003546714783, Actual: 0.8922661351679123
Predicted: 2.826253652572632, Actual: 2.669749932596335
Predicted: 3.139582872390747, Actual: 3.088512305616677
Predicted: 1.8380144834518433, Actual: 1.5635740338353443
Predicted: 2.779873847961426, Actual: 2.875735286006944
Predicted: 0.854584276676178, Actual: 1.184519764401664
Predicted: 2.459192991256714, Actual: 2.584174278596964
Predicted: 3.4654102325439453, Actual: 3.812757187598897
Predicted: 0.4390970468521118, Actual: 0.6729361977994386
Predicted: 1.5382599830627441, Actual: 1.9558725265628367
Predicted: 1.691450595855713, Actual: 1.773293697386631
Predicted: 3.094693899154663, Actual: 3.055386295398076
Predicted: 2.762086868286133, Actual: 2.718465435794503
Predicted: 1.3632055521011353, Actual: 1.2705295622953123
Predicted: 1.9288766384124756, Actual: 1.9879647265294176
Predicted: 1.5935111045837402, Actual: 1.7209071500753406
Predicted: 1.775460958480835, Actual: 2.1105837595046006
Predicted: 2.809520959854126, Actual: 2.8816123214605245
Predicted: 1.064257264137268, Actual: 0.9707643470928292
Predicted: 2.8902652263641357, Actual: 2.9728834414945147
Predicted: 2.5649285316467285, Actual: 2.5210277348652257
Predicted: 0.4626639485359192, Actual: 0.5152393995715008
Predicted: 0.6791861057281494, Actual: 1.0085719028530793
Predicted: 2.822789192199707, Actual: 2.791339403107066
```

```
Predicted: 0.6887026429176331, Actual: 0.586837405483887
Predicted: 3.450500249862671, Actual: 3.666290826527483
Predicted: 1.202465295791626, Actual: 0.9030855768578068
Predicted: 0.8664146065711975, Actual: 0.7045952434646587
Predicted: 2.340118169784546, Actual: 2.3983144793001743
Predicted: 1.5670568943023682, Actual: 1.817789572439356
Predicted: 0.5281149744987488, Actual: 0.7995911976901089
Predicted: 2.4165079593658447, Actual: 2.127408865912826
Predicted: 1.0101231336593628, Actual: 1.1767815056705988
Predicted: 0.5685091018676758, Actual: 0.4291372796949342
Predicted: 1.9228354692459106, Actual: 2.0469388171096283
Predicted: 0.9679123759269714, Actual: 1.0861513973549914
Predicted: 2.915221929550171, Actual: 2.9869427371126283
Predicted: 2.3751072883605957, Actual: 2.474039536809681
Predicted: 1.9135222434997559, Actual: 1.581931480635446
Predicted: 2.1601784229278564, Actual: 2.3278327626806856
Predicted: 1.852914571762085, Actual: 1.5252820489261911
Predicted: 1.3525595664978027, Actual: 1.6831172628416484
Predicted: 2.2331643104553223, Actual: 2.4610469629080955
Predicted: 0.3017435073852539, Actual: 0.0278186247324679
Predicted: 1.4484013319015503, Actual: 1.5364194793970165
Predicted: 2.186326742172241, Actual: 2.0998851567205854
Predicted: 0.3536549210548401, Actual: 0.3751478505639292
Predicted: 1.3840532302856445, Actual: 1.2945928801079334
Predicted: 0.44141685962677, Actual: 0.7697406121390962
Predicted: 1.7776261568069458, Actual: 1.821442173929074
Predicted: 2.942898988723755, Actual: 3.4982573417733205
Predicted: 2.1138756275177, Actual: 2.378050042081803
Predicted: 0.9159853458404541, Actual: 0.5185224297099189
Predicted: 2.5920941829681396, Actual: 2.754877493270432
Predicted: 2.22906494140625, Actual: 2.207001981395421
Predicted: 1.213460922241211, Actual: 1.1416385674703
Predicted: 3.0783112049102783, Actual: 3.137624350847544
Predicted: 1.3618731498718262, Actual: 1.3638410472397342
Predicted: 1.3570547103881836, Actual: 1.6884763841942343
Predicted: 1.410080909729004, Actual: 1.3503298063610123
Predicted: 2.4165232181549072, Actual: 2.0624682672770884
Predicted: 1.0074716806411743, Actual: 1.022641733675116
Predicted: 1.7355681657791138, Actual: 1.7922893205354298
Predicted: 0.2704230546951294, Actual: 0.0
Predicted: 2.70068621635437, Actual: 2.8728793655039597
Predicted: 3.174628973007202, Actual: 3.08869986818596
Predicted: 2.7882306575775146, Actual: 2.937454182316864
Predicted: 1.835256814956665, Actual: 1.9900604524480008
Predicted: 1.5722310543060303, Actual: 1.5165082679194366
Predicted: 3.1566269397735596, Actual: 3.300928568970799
Predicted: 2.250347137451172, Actual: 2.172740551700977
Predicted: 1.3011925220489502, Actual: 1.11944101772116
Predicted: 1.4921019077301025, Actual: 1.5717864335294334
Predicted: 0.7869974970817566, Actual: 0.4958590410264636
Predicted: 0.4027708172798157, Actual: 0.155004788411154
Predicted: 3.408130407333374, Actual: 3.8649768612963977
Predicted: 2.0844955444335938, Actual: 1.8044174585321688
Predicted: 2.6285083293914795, Actual: 2.784237837142815
Predicted: 1.1543079614639282, Actual: 1.2058569757056352
Predicted: 2.586243152618408, Actual: 2.823600084682192
Predicted: 1.1829266548156738, Actual: 0.8952499802854423
Predicted: 2.965270757675171, Actual: 3.1122811365205387
Predicted: 1.6188112497329712, Actual: 1.8154017176022967
Predicted: 0.2207823544740677, Actual: 0.1005921480894177
```

```
Predicted: 3.010019063949585, Actual: 3.065473133634235
Predicted: 2.750500440597534, Actual: 2.5538644982316425
Predicted: 1.9899628162384033, Actual: 1.9863780592187463
Predicted: 0.9138733744621277, Actual: 0.9941294103867458
Predicted: 3.264963388442993, Actual: 3.3501309740346112
Predicted: 2.0727450847625732, Actual: 2.3116704996467785
Predicted: 2.5451338291168213, Actual: 2.504167767054588
Predicted: 0.3393136262893677, Actual: 0.763471998256058
Predicted: 2.4854471683502197, Actual: 2.71222974268319
Predicted: 0.7936924695968628, Actual: 0.9252693613945324
Predicted: 1.7352527379989624, Actual: 1.849760707133483
Predicted: 1.3973945379257202, Actual: 1.3970629516250068
Predicted: 1.6067609786987305, Actual: 1.8889919304287464
Predicted: 0.6356110572814941, Actual: 0.9000416924553322
Predicted: 0.5439927577972412, Actual: 1.027727966185627
Predicted: 0.8722530603408813, Actual: 0.7016799102374711
Predicted: 1.412558674812317, Actual: 1.4119815069059265
Predicted: 2.8839457035064697, Actual: 2.977408432892682
Predicted: 2.327009916305542, Actual: 2.2103713279948844
Predicted: 2.463689088821411, Actual: 2.438731252650425
Predicted: 1.8358601331710815, Actual: 1.9212942080249984
Predicted: 1.788067102432251, Actual: 1.7339724542334658
Predicted: 1.254281997680664, Actual: 1.5892697074168551
Predicted: 1.8801485300064087, Actual: 1.503416679305313
Predicted: 2.0890696048736572, Actual: 2.222002853095735
Predicted: 2.9562318325042725, Actual: 3.1062304249844512
Predicted: 1.408050775527954, Actual: 1.5420637396404375
Predicted: 0.8908218741416931, Actual: 1.2523485891697117
Predicted: 1.2222436666488647, Actual: 1.710466837709535
Predicted: 2.5782575607299805, Actual: 2.693598515909963
Predicted: 1.0936752557754517, Actual: 0.9657009889491656
Predicted: 1.889742374420166, Actual: 1.86827692299816
Predicted: 2.5925254821777344, Actual: 2.806877505411242
Predicted: 2.3939368724823, Actual: 2.342836773191855
Predicted: 2.6968305110931396, Actual: 2.583738162772638
Predicted: 3.0577473640441895, Actual: 3.2485727923536656
Predicted: 2.593996286392212, Actual: 2.5844682625115074
Predicted: 2.48109769821167, Actual: 2.587565796370635
Predicted: 1.220760464668274, Actual: 1.5660204997024247
Predicted: 0.5037482380867004, Actual: 0.8670343739276987
Predicted: 2.0665316581726074, Actual: 1.8476004181084256
Predicted: 1.3068697452545166, Actual: 1.1221109734183077
Predicted: 1.4413912296295166, Actual: 1.7181817231817458
Predicted: 1.2335281372070312, Actual: 1.1645391660958937
Predicted: 1.8109571933746338, Actual: 1.6833300315712254
Predicted: 2.1976842880249023, Actual: 2.018579782190191
Predicted: 2.453808069229126, Actual: 2.512488758731327
Predicted: 3.0393805503845215, Actual: 3.168988610106379
Predicted: 3.076615571975708, Actual: 3.0526409582889382
Predicted: 1.0107991695404053, Actual: 1.1528666911541523
Predicted: 3.126746416091919, Actual: 3.25829468502287
Predicted: 3.2242960929870605, Actual: 3.363670491095614
Predicted: 2.2298967838287354, Actual: 2.404089819888117
Predicted: 2.1015563011169434, Actual: 2.137692088152197
Predicted: 2.444669008255005, Actual: 2.427614635957672
Predicted: 1.923379898071289, Actual: 1.810037822647263
Predicted: 1.909618854522705, Actual: 1.756035494723187
Predicted: 2.1813762187957764, Actual: 2.053147894166025
Predicted: 1.0180610418319702, Actual: 1.6014725497408242
Predicted: 0.46234285831451416, Actual: 0.4695533233798704
```

```
Predicted: 2.2329933643341064, Actual: 2.0046273170074884
Predicted: 2.3833425045013428, Actual: 2.2887682028383933
Predicted: 3.046457529067993, Actual: 3.091715028356484
Predicted: 2.4327409267425537, Actual: 2.117233031215417
Predicted: 1.1191900968551636, Actual: 1.3476334931151803
Predicted: 1.655390739440918, Actual: 1.5304278892477712
Predicted: 2.7066802978515625, Actual: 2.734401494419992
Predicted: 3.4315943717956543, Actual: 3.8302053754229743
Predicted: 2.666236400604248, Actual: 2.6394472915746228
Predicted: 2.1027355194091797, Actual: 2.471395779180272
Predicted: 2.1753909587860107, Actual: 2.2286086189122747
Predicted: 0.5818309187889099, Actual: 0.6566169233161951
Predicted: 2.2093920707702637, Actual: 2.2598041504961186
Predicted: 2.5602080821990967, Actual: 2.3327839636333105
Predicted: 2.789968252182007, Actual: 3.0234824828446896
Predicted: 2.420935869216919, Actual: 2.964668230929299
Predicted: 0.9763807058334351, Actual: 0.983576997394541
Predicted: 1.421317458152771, Actual: 1.465549386433812
Predicted: 2.3959484100341797, Actual: 2.268850734169383
Predicted: 3.291919708251953, Actual: 3.5729453198762804
Predicted: 0.8878996968269348, Actual: 0.6822651648416035
Predicted: 1.8258450031280518, Actual: 1.9846599185571745
Predicted: 2.516392946243286, Actual: 2.519724822511747
Predicted: 0.4919050931930542, Actual: 0.5496018253589585
Predicted: 2.882930040359497, Actual: 2.9868916209305363
Predicted: 2.6506295204162598, Actual: 2.6404761356980795
Predicted: 1.5369913578033447, Actual: 1.4772887703973712
Predicted: 1.9958252906799316, Actual: 1.9395846660340497
Predicted: 1.821760654449463, Actual: 1.8298105464885008
Predicted: 2.6323673725128174, Actual: 2.6696408286087285
Predicted: 1.7705953121185303, Actual: 1.5961281034309902
Predicted: 2.263143539428711, Actual: 2.278951504956563
Predicted: 0.9183775782585144, Actual: 1.1378011102056016
Predicted: 1.0025562047958374, Actual: 0.6768179731112823
Predicted: 2.388307809829712, Actual: 2.5040052870066454
Predicted: 2.1412813663482666, Actual: 2.122638529628868
Predicted: 2.5892090797424316, Actual: 2.977851918315743
Predicted: 2.444669246673584, Actual: 2.827614868021281
Predicted: 2.783627510070801, Actual: 2.942198841728139
Predicted: 2.061018228530884, Actual: 2.309975847769393
Predicted: 2.626006841659546, Actual: 2.7778858935835875
Predicted: 2.5948662757873535, Actual: 2.5244226584154354
Predicted: 2.3115384578704834, Actual: 2.135266354890094
Predicted: 2.283226490020752, Actual: 2.224197461212005
Predicted: 3.034635305404663, Actual: 3.2389317177358112
Predicted: 2.34891939163208, Actual: 2.230253521333785
Predicted: 2.6866424083709717, Actual: 3.060805402033993
Predicted: 0.9624164700508118, Actual: 1.3635615499322744
Predicted: 3.2780308723449707, Actual: 3.603507572240497
Predicted: 2.084505319595337, Actual: 2.1672829561454443
Predicted: 3.04679012298584, Actual: 3.323902960329112
Predicted: 1.957099437713623, Actual: 2.0165967745176614
Predicted: 1.6172759532928467, Actual: 1.3620437691526197
Predicted: 2.0462124347686768, Actual: 2.1156039687255843
Predicted: 2.437228202819824, Actual: 2.465306489467062
Predicted: 3.237593412399292, Actual: 3.060490750087925
Predicted: 2.009826421737671, Actual: 1.9915084140251773
Predicted: 0.302803635597229, Actual: 0.264924162952827
Predicted: 3.3276708126068115, Actual: 3.64573804877044
Predicted: 2.797380208969116, Actual: 2.5172289818586204
```

```
Predicted: 1.4660730361938477, Actual: 1.5487102111350304
Predicted: 1.4235897064208984, Actual: 1.599593887728051
Predicted: 1.5703387260437012, Actual: 1.5538733401089513
Predicted: 2.2236146926879883, Actual: 2.595889050696416
Predicted: 1.842270851135254, Actual: 1.8790984603853385
Predicted: 1.3408987522125244, Actual: 1.5066627733506968
Predicted: 1.1342869997024536, Actual: 1.1048937876999445
Predicted: 0.7622981071472168, Actual: 0.3413935419913523
Predicted: 2.2529613971710205, Actual: 2.4052675257641485
Predicted: 1.8306490182876587, Actual: 1.6848422066992197
Predicted: 2.774724721908569З, Actual: 2.993502391205279
Predicted: 2.9869513511657715, Actual: 3.4154133242434344
Predicted: 1.3720698356628418, Actual: 1.7097013783889978
Predicted: 1.7662129402160645, Actual: 1.5386892509239083
Predicted: 2.8236160278320312, Actual: 2.921386757069629
Predicted: 2.3696954250335693, Actual: 2.605247445547851
Predicted: 2.265432357788086, Actual: 2.4400060965135784
Predicted: 1.4045453071594238, Actual: 1.42899323811113
Predicted: 3.2221431732177734, Actual: 3.3721260428259656
Predicted: 2.239773750305176, Actual: 2.1516179803357813
Predicted: 1.682082176208496, Actual: 1.5950547587554351
Predicted: 1.396370530128479, Actual: 1.4145556778225787
Predicted: 3.00049090385437, Actual: 3.128858646003016
Predicted: 2.9231367111206055, Actual: 2.553439811443938
Predicted: 2.853463888168335, Actual: 3.1296400506375512
Predicted: 2.8336799144744873, Actual: 2.981992255406158
Predicted: 2.8505425453186035, Actual: 2.981786874282664
Predicted: 0.9293222427368164, Actual: 1.2010533026138683
Predicted: 2.6611907482147217, Actual: 2.79138610847234
Predicted: 0.5791555047035217, Actual: 0.1530318116508149
Predicted: 0.7632426619529724, Actual: 0.6193509314675341
Predicted: 2.2743990421295166, Actual: 2.281345006080266
Predicted: 0.7046546339988708, Actual: 0.4937411351889655
Predicted: 2.8192782402038574, Actual: 2.966548034294946
Predicted: 1.948339581489563, Actual: 1.989455195588546
Predicted: 3.51948618888855, Actual: 3.5433159148930136
Predicted: 1.9613840579986572, Actual: 1.970600159244711
Predicted: 1.1647638082504272, Actual: 1.3440857175872982
Predicted: 0.8370093107223511, Actual: 1.027015599936749
Predicted: 0.5502614974975586, Actual: 0.4277633541749704
Predicted: 1.8493523597717285, Actual: 1.9746552454467765
Predicted: 1.0523204803466797, Actual: 1.242176324159397
Predicted: 0.7026596665382385, Actual: 0.384375395876116
Predicted: 0.6877423524856567, Actual: 0.4474828563605549
Predicted: 1.598934531211853, Actual: 1.7290731666537256
Predicted: 2.2158493995666504, Actual: 2.136399546240388
Predicted: 0.22412844002246857, Actual: 0.2109791787447653
Predicted: 0.48548150062561035, Actual: 0.33096690773874
Predicted: 2.724203586578369, Actual: 2.888891669272444
Predicted: 3.1624717712402344, Actual: 3.270373860776488
Predicted: 2.6151278018951416, Actual: 2.500113299959695
Predicted: 1.2388062477111816, Actual: 1.212880751488374
Predicted: 1.9674663543701172, Actual: 2.009298479722319
Predicted: 1.2143813371658325, Actual: 1.543837471085622
Predicted: 1.444828987121582, Actual: 1.4382049683676248
Predicted: 1.78999662399292, Actual: 1.562359564441758
Predicted: 1.8504608869552612, Actual: 2.17490279620988
Predicted: 2.4247419834136963, Actual: 2.3325403195354064
Predicted: 2.5379319190979004, Actual: 2.777966932491008
Predicted: 0.9285194873809814, Actual: 0.863545351686935
```

13. Compete against this model:

- Create two more different models to compete with this model
- Here are a few ideas of things you can change:
  - During Dataset data engineering:
    - You can remove features that you think do not help in the training and prediction
    - Feature Scaling: Ensure all features are on a similar scale (as you already did with StandardScaler)
  - During Model Definition:
    - You can change the Model Architecture (change the type or number of layers or the number of units)
    - You can add dropout layers to prevent overfitting
  - During Model Compile:
    - You can try other optimizer when compiling your model, here some optimizer samples: Adam, RMSprop, or Adagrad.
    - Try another Loss Function
  - During Model Training:
    - Encrease the number of Epochs
    - Adjust the size of your batch
- Explain in a Markdown cell which changes are you implementing
- Show the comparison of your model versus the original model

Model 2:

- Changes:
  - Dataset Data Engineering
  - Model Definition
  - Model Compile
  - Model Training

In [21]:
```
model2 = Sequential()
model2.add(Dense(128, input_dim=10, activation='relu'))
model2.add(Dense(64, activation='relu'))
model2.add(Dropout(0.2))
model2.add(Dense(32, activation='relu'))
model2.add(Dense(1))

model2.compile(optimizer='RMSprop', loss='mean_squared_error', metrics=['mean_absolute

history2 = model2.fit(X_train_scaled, y_train, epochs=50, batch_size=10, validation_sp

final_dataset2 = dataset.drop(columns=['Gender', 'Ethnicity', 'Volunteering', 'Music']

X2 = final_dataset2.drop(columns=['GPA'])
y2 = final_dataset2['GPA']

X_train2, X_test2, y_train2, y_test2 = train_test_split(X2, y2, test_size=0.2, random_

scaler2 = StandardScaler()
```

```
X_train_scaled2 = scaler2.fit_transform(X_train2)
X_test_scaled2 = scaler2.transform(X_test2)
```

```
Epoch 1/50
```

```
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87: UserWarnin
g: Do not pass an `input_shape`/`input_dim` argument to a layer. When using Sequentia
l models, prefer using an `Input(shape)` object as the first layer in the model inste
ad.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
```

```
153/153 ━━━━━━━━━━━━━━━━━━━━ 3s 8ms/step - loss: 0.6306 - mean_absolute_error: 0.5730
- val_loss: 0.0725 - val_mean_absolute_error: 0.2199
Epoch 2/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 2s 4ms/step - loss: 0.1204 - mean_absolute_error: 0.2749
- val_loss: 0.0774 - val_mean_absolute_error: 0.2304
Epoch 3/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 4ms/step - loss: 0.0919 - mean_absolute_error: 0.2429
- val_loss: 0.0586 - val_mean_absolute_error: 0.1959
Epoch 4/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 4ms/step - loss: 0.0738 - mean_absolute_error: 0.2130
- val_loss: 0.0524 - val_mean_absolute_error: 0.1826
Epoch 5/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0663 - mean_absolute_error: 0.2033
- val_loss: 0.0473 - val_mean_absolute_error: 0.1740
Epoch 6/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - loss: 0.0617 - mean_absolute_error: 0.1980
- val_loss: 0.0447 - val_mean_absolute_error: 0.1679
Epoch 7/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 6ms/step - loss: 0.0550 - mean_absolute_error: 0.1852
- val_loss: 0.0684 - val_mean_absolute_error: 0.2134
Epoch 8/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 4ms/step - loss: 0.0536 - mean_absolute_error: 0.1838
- val_loss: 0.0417 - val_mean_absolute_error: 0.1616
Epoch 9/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 4ms/step - loss: 0.0540 - mean_absolute_error: 0.1834
- val_loss: 0.0566 - val_mean_absolute_error: 0.1900
Epoch 10/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - loss: 0.0482 - mean_absolute_error: 0.1724
- val_loss: 0.0466 - val_mean_absolute_error: 0.1697
Epoch 11/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - loss: 0.0491 - mean_absolute_error: 0.1785
- val_loss: 0.0426 - val_mean_absolute_error: 0.1628
Epoch 12/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - loss: 0.0419 - mean_absolute_error: 0.1616
- val_loss: 0.0453 - val_mean_absolute_error: 0.1676
Epoch 13/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0436 - mean_absolute_error: 0.1665
- val_loss: 0.0446 - val_mean_absolute_error: 0.1674
Epoch 14/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - loss: 0.0433 - mean_absolute_error: 0.1644
- val_loss: 0.0544 - val_mean_absolute_error: 0.1845
Epoch 15/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - loss: 0.0425 - mean_absolute_error: 0.1655
- val_loss: 0.0568 - val_mean_absolute_error: 0.1883
Epoch 16/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - loss: 0.0379 - mean_absolute_error: 0.1523
- val_loss: 0.1131 - val_mean_absolute_error: 0.2777
Epoch 17/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - loss: 0.0410 - mean_absolute_error: 0.1602
- val_loss: 0.0462 - val_mean_absolute_error: 0.1647
Epoch 18/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - loss: 0.0411 - mean_absolute_error: 0.1580
- val_loss: 0.0461 - val_mean_absolute_error: 0.1692
Epoch 19/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - loss: 0.0432 - mean_absolute_error: 0.1644
- val_loss: 0.0515 - val_mean_absolute_error: 0.1803
Epoch 20/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - loss: 0.0399 - mean_absolute_error: 0.1554
- val_loss: 0.0419 - val_mean_absolute_error: 0.1607
Epoch 21/50
```

```
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - loss: 0.0382 - mean_absolute_error: 0.1554
- val_loss: 0.0857 - val_mean_absolute_error: 0.2446
Epoch 22/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - loss: 0.0339 - mean_absolute_error: 0.1452
- val_loss: 0.0581 - val_mean_absolute_error: 0.1948
Epoch 23/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 3ms/step - loss: 0.0365 - mean_absolute_error: 0.1517
- val_loss: 0.0601 - val_mean_absolute_error: 0.2002
Epoch 24/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - loss: 0.0341 - mean_absolute_error: 0.1459
- val_loss: 0.0540 - val_mean_absolute_error: 0.1855
Epoch 25/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0369 - mean_absolute_error: 0.1514
- val_loss: 0.0497 - val_mean_absolute_error: 0.1773
Epoch 26/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - loss: 0.0317 - mean_absolute_error: 0.1410
- val_loss: 0.0658 - val_mean_absolute_error: 0.2101
Epoch 27/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 4ms/step - loss: 0.0362 - mean_absolute_error: 0.1487
- val_loss: 0.0723 - val_mean_absolute_error: 0.2198
Epoch 28/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - loss: 0.0330 - mean_absolute_error: 0.1424
- val_loss: 0.0514 - val_mean_absolute_error: 0.1818
Epoch 29/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 7ms/step - loss: 0.0306 - mean_absolute_error: 0.1366
- val_loss: 0.0612 - val_mean_absolute_error: 0.2033
Epoch 30/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 4ms/step - loss: 0.0336 - mean_absolute_error: 0.1451
- val_loss: 0.0592 - val_mean_absolute_error: 0.1948
Epoch 31/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - loss: 0.0301 - mean_absolute_error: 0.1354
- val_loss: 0.0677 - val_mean_absolute_error: 0.2098
Epoch 32/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - loss: 0.0327 - mean_absolute_error: 0.1405
- val_loss: 0.0532 - val_mean_absolute_error: 0.1848
Epoch 33/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - loss: 0.0314 - mean_absolute_error: 0.1389
- val_loss: 0.0652 - val_mean_absolute_error: 0.2092
Epoch 34/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - loss: 0.0315 - mean_absolute_error: 0.1415
- val_loss: 0.0832 - val_mean_absolute_error: 0.2408
Epoch 35/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0293 - mean_absolute_error: 0.1315
- val_loss: 0.0992 - val_mean_absolute_error: 0.2599
Epoch 36/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - loss: 0.0285 - mean_absolute_error: 0.1300
- val_loss: 0.0787 - val_mean_absolute_error: 0.2295
Epoch 37/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0274 - mean_absolute_error: 0.1282
- val_loss: 0.0527 - val_mean_absolute_error: 0.1858
Epoch 38/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - loss: 0.0297 - mean_absolute_error: 0.1342
- val_loss: 0.0431 - val_mean_absolute_error: 0.1600
Epoch 39/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - loss: 0.0279 - mean_absolute_error: 0.1341
- val_loss: 0.1151 - val_mean_absolute_error: 0.2786
Epoch 40/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - loss: 0.0298 - mean_absolute_error: 0.1335
- val_loss: 0.0761 - val_mean_absolute_error: 0.2266
Epoch 41/50
```

```
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - loss: 0.0284 - mean_absolute_error: 0.1326
- val_loss: 0.1087 - val_mean_absolute_error: 0.2733
Epoch 42/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 3ms/step - loss: 0.0268 - mean_absolute_error: 0.1301
- val_loss: 0.0560 - val_mean_absolute_error: 0.1873
Epoch 43/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - loss: 0.0264 - mean_absolute_error: 0.1272
- val_loss: 0.0875 - val_mean_absolute_error: 0.2468
Epoch 44/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - loss: 0.0268 - mean_absolute_error: 0.1293
- val_loss: 0.0632 - val_mean_absolute_error: 0.2010
Epoch 45/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - loss: 0.0242 - mean_absolute_error: 0.1229
- val_loss: 0.0612 - val_mean_absolute_error: 0.2000
Epoch 46/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - loss: 0.0236 - mean_absolute_error: 0.1197
- val_loss: 0.0521 - val_mean_absolute_error: 0.1858
Epoch 47/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - loss: 0.0240 - mean_absolute_error: 0.1222
- val_loss: 0.0812 - val_mean_absolute_error: 0.2361
Epoch 48/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 4ms/step - loss: 0.0258 - mean_absolute_error: 0.1268
- val_loss: 0.0721 - val_mean_absolute_error: 0.2172
Epoch 49/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - loss: 0.0238 - mean_absolute_error: 0.1186
- val_loss: 0.0663 - val_mean_absolute_error: 0.2088
Epoch 50/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 4ms/step - loss: 0.0244 - mean_absolute_error: 0.1222
- val_loss: 0.0915 - val_mean_absolute_error: 0.2513
```

Model 3:

- Changes:
  - Dataset Data Engineering
  - Model Definition
  - Model Compile
  - Model Training

In [22]:
```
model3 = Sequential()
model3.add(Dense(64, input_dim=X_train_scaled2.shape[1], activation='relu'))
model3.add(Dense(32, activation='relu'))
model3.add(Dense(1))

model3.compile(optimizer='Adam', loss='mean_squared_error', metrics=['mean_absolute_er

history3 = model3.fit(X_train_scaled2, y_train2, epochs=50, batch_size=15, validation_
```

```
Epoch 1/50
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87: UserWarnin
g: Do not pass an `input_shape`/`input_dim` argument to a layer. When using Sequentia
l models, prefer using an `Input(shape)` object as the first layer in the model inste
ad.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
```

```
102/102 ━━━━━━━━━━━━━━━━━━━━ 1s 4ms/step - loss: 2.3189 - mean_absolute_error: 1.2584
- val_loss: 0.2466 - val_mean_absolute_error: 0.4064
Epoch 2/50
102/102 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.1723 - mean_absolute_error: 0.3305
- val_loss: 0.1149 - val_mean_absolute_error: 0.2792
Epoch 3/50
102/102 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0933 - mean_absolute_error: 0.2487
- val_loss: 0.0811 - val_mean_absolute_error: 0.2325
Epoch 4/50
102/102 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0700 - mean_absolute_error: 0.2141
- val_loss: 0.0657 - val_mean_absolute_error: 0.2051
Epoch 5/50
102/102 ━━━━━━━━━━━━━━━━━━━━ 0s 3ms/step - loss: 0.0578 - mean_absolute_error: 0.1970
- val_loss: 0.0590 - val_mean_absolute_error: 0.1956
Epoch 6/50
102/102 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0482 - mean_absolute_error: 0.1776
- val_loss: 0.0534 - val_mean_absolute_error: 0.1839
Epoch 7/50
102/102 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0429 - mean_absolute_error: 0.1674
- val_loss: 0.0519 - val_mean_absolute_error: 0.1803
Epoch 8/50
102/102 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0418 - mean_absolute_error: 0.1653
- val_loss: 0.0499 - val_mean_absolute_error: 0.1747
Epoch 9/50
102/102 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0431 - mean_absolute_error: 0.1682
- val_loss: 0.0452 - val_mean_absolute_error: 0.1685
Epoch 10/50
102/102 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0420 - mean_absolute_error: 0.1633
- val_loss: 0.0453 - val_mean_absolute_error: 0.1700
Epoch 11/50
102/102 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0363 - mean_absolute_error: 0.1530
- val_loss: 0.0434 - val_mean_absolute_error: 0.1662
Epoch 12/50
102/102 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0354 - mean_absolute_error: 0.1494
- val_loss: 0.0431 - val_mean_absolute_error: 0.1654
Epoch 13/50
102/102 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0359 - mean_absolute_error: 0.1504
- val_loss: 0.0424 - val_mean_absolute_error: 0.1648
Epoch 14/50
102/102 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0349 - mean_absolute_error: 0.1464
- val_loss: 0.0418 - val_mean_absolute_error: 0.1620
Epoch 15/50
102/102 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0323 - mean_absolute_error: 0.1407
- val_loss: 0.0412 - val_mean_absolute_error: 0.1623
Epoch 16/50
102/102 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0309 - mean_absolute_error: 0.1396
- val_loss: 0.0410 - val_mean_absolute_error: 0.1600
Epoch 17/50
102/102 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0300 - mean_absolute_error: 0.1369
- val_loss: 0.0418 - val_mean_absolute_error: 0.1596
Epoch 18/50
102/102 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0306 - mean_absolute_error: 0.1383
- val_loss: 0.0439 - val_mean_absolute_error: 0.1649
Epoch 19/50
102/102 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0303 - mean_absolute_error: 0.1365
- val_loss: 0.0404 - val_mean_absolute_error: 0.1562
Epoch 20/50
102/102 ━━━━━━━━━━━━━━━━━━━━ 0s 3ms/step - loss: 0.0312 - mean_absolute_error: 0.1398
- val_loss: 0.0401 - val_mean_absolute_error: 0.1578
Epoch 21/50
```

```
102/102 ━━━━━━━━━━━━━━━━━━━━ 1s 4ms/step - loss: 0.0287 - mean_absolute_error: 0.1329
- val_loss: 0.0448 - val_mean_absolute_error: 0.1654
Epoch 22/50
102/102 ━━━━━━━━━━━━━━━━━━━━ 1s 4ms/step - loss: 0.0288 - mean_absolute_error: 0.1354
- val_loss: 0.0424 - val_mean_absolute_error: 0.1624
Epoch 23/50
102/102 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - loss: 0.0307 - mean_absolute_error: 0.1376
- val_loss: 0.0410 - val_mean_absolute_error: 0.1598
Epoch 24/50
102/102 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - loss: 0.0306 - mean_absolute_error: 0.1384
- val_loss: 0.0422 - val_mean_absolute_error: 0.1611
Epoch 25/50
102/102 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - loss: 0.0268 - mean_absolute_error: 0.1293
- val_loss: 0.0398 - val_mean_absolute_error: 0.1566
Epoch 26/50
102/102 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0262 - mean_absolute_error: 0.1284
- val_loss: 0.0488 - val_mean_absolute_error: 0.1733
Epoch 27/50
102/102 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0275 - mean_absolute_error: 0.1309
- val_loss: 0.0405 - val_mean_absolute_error: 0.1575
Epoch 28/50
102/102 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0251 - mean_absolute_error: 0.1247
- val_loss: 0.0410 - val_mean_absolute_error: 0.1589
Epoch 29/50
102/102 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0267 - mean_absolute_error: 0.1294
- val_loss: 0.0404 - val_mean_absolute_error: 0.1578
Epoch 30/50
102/102 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0250 - mean_absolute_error: 0.1247
- val_loss: 0.0399 - val_mean_absolute_error: 0.1551
Epoch 31/50
102/102 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0259 - mean_absolute_error: 0.1291
- val_loss: 0.0421 - val_mean_absolute_error: 0.1604
Epoch 32/50
102/102 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0256 - mean_absolute_error: 0.1239
- val_loss: 0.0460 - val_mean_absolute_error: 0.1661
Epoch 33/50
102/102 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0262 - mean_absolute_error: 0.1280
- val_loss: 0.0424 - val_mean_absolute_error: 0.1626
Epoch 34/50
102/102 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0246 - mean_absolute_error: 0.1244
- val_loss: 0.0442 - val_mean_absolute_error: 0.1619
Epoch 35/50
102/102 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0266 - mean_absolute_error: 0.1274
- val_loss: 0.0408 - val_mean_absolute_error: 0.1564
Epoch 36/50
102/102 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0247 - mean_absolute_error: 0.1252
- val_loss: 0.0422 - val_mean_absolute_error: 0.1594
Epoch 37/50
102/102 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0244 - mean_absolute_error: 0.1237
- val_loss: 0.0448 - val_mean_absolute_error: 0.1661
Epoch 38/50
102/102 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0253 - mean_absolute_error: 0.1245
- val_loss: 0.0426 - val_mean_absolute_error: 0.1592
Epoch 39/50
102/102 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0234 - mean_absolute_error: 0.1196
- val_loss: 0.0445 - val_mean_absolute_error: 0.1618
Epoch 40/50
102/102 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0254 - mean_absolute_error: 0.1257
- val_loss: 0.0427 - val_mean_absolute_error: 0.1593
Epoch 41/50
```

```
102/102 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0235 - mean_absolute_error: 0.1217
- val_loss: 0.0430 - val_mean_absolute_error: 0.1601
Epoch 42/50
102/102 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0230 - mean_absolute_error: 0.1184
- val_loss: 0.0438 - val_mean_absolute_error: 0.1642
Epoch 43/50
102/102 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0242 - mean_absolute_error: 0.1198
- val_loss: 0.0453 - val_mean_absolute_error: 0.1658
Epoch 44/50
102/102 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0243 - mean_absolute_error: 0.1238
- val_loss: 0.0476 - val_mean_absolute_error: 0.1692
Epoch 45/50
102/102 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0241 - mean_absolute_error: 0.1211
- val_loss: 0.0435 - val_mean_absolute_error: 0.1620
Epoch 46/50
102/102 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0229 - mean_absolute_error: 0.1189
- val_loss: 0.0442 - val_mean_absolute_error: 0.1636
Epoch 47/50
102/102 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0218 - mean_absolute_error: 0.1157
- val_loss: 0.0476 - val_mean_absolute_error: 0.1731
Epoch 48/50
102/102 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0235 - mean_absolute_error: 0.1201
- val_loss: 0.0470 - val_mean_absolute_error: 0.1681
Epoch 49/50
102/102 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0236 - mean_absolute_error: 0.1220
- val_loss: 0.0483 - val_mean_absolute_error: 0.1730
Epoch 50/50
102/102 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0234 - mean_absolute_error: 0.1217
- val_loss: 0.0473 - val_mean_absolute_error: 0.1688
```

In [23]:

```python
loss1, mae1 = model.evaluate(X_test_scaled, y_test)
loss2, mae2 = model2.evaluate(X_test_scaled, y_test)
loss3, mae3 = model3.evaluate(X_test_scaled2, y_test2)

print(f"Model 1 - Loss: {loss1}, MAE: {mae1}")
print(f"Model 2 - Loss: {loss2}, MAE: {mae2}")
print(f"Model 3 - Loss: {loss3}, MAE: {mae3}")
```

```
15/15 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0475 - mean_absolute_error: 0.1703
15/15 ━━━━━━━━━━━━━━━━━━━━ 0s 3ms/step - loss: 0.0860 - mean_absolute_error: 0.2364
15/15 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0473 - mean_absolute_error: 0.1715
Model 1 - Loss: 0.045884568244218826, MAE: 0.16879822313785553
Model 2 - Loss: 0.0872996523976326, MAE: 0.24258054792881012
Model 3 - Loss: 0.043446801602840424, MAE: 0.16504491865634918
```