Modulo 2 - 8. Dense + Dropout + Batch Normalization

Use the Student GPA dataset to predict student GPA.

Use previous concepts to create different Neural Network Architectures and compare your results. (Python Notebook)

Experiment 1: A single Dense Hidden Layer

Experiment 2: A set of three Dense Hidden Layers

Experiment 3: Add a dropout layer after each Dense Hidden Layer

Experiment 4: Add a Batch Normalization Layer after each Dropout Layer.

Create a comparative table and upload you code and the comparative table as the activity evidence. (Notebook and PDF)

1. Import Libraries

In [1]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import tensorflow as tf
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense
from tensorflow.keras.layers import Dropout
from tensorflow.keras.layers import BatchNormalization
from tabulate import tabulate
```

2. Load Data

In [2]:
```
data = pd.read_csv('Student_performance_data _.csv')
data.head(10)
```

Out[2]:

	StudentID	Age	Gender	Ethnicity	ParentalEducation	StudyTimeWeekly	Absences	Tutoring	ParentalSupport	Extracurricular	Sports	Mus
0	1001	17	1	0	2	19.833723	7	1	2	0	0	
1	1002	18	0	0	1	15.408756	0	0	1	0	0	
2	1003	15	0	2	3	4.210570	26	0	2	0	0	
3	1004	17	1	0	3	10.028829	14	0	3	1	0	
4	1005	17	1	0	2	4.672495	17	1	3	0	0	
5	1006	18	0	0	1	8.191219	0	0	1	1	0	
6	1007	15	0	1	1	15.601680	10	0	3	0	1	
7	1008	15	1	1	4	15.424496	22	1	1	1	0	
8	1009	17	0	0	0	4.562008	1	0	2	0	1	
9	1010	16	1	0	1	18.444466	0	0	3	1	0	

3. Remove Columns

In [3]:
```
dataset = data.drop(columns=['StudentID'])
```

4. Define Train and Test Sets

In [4]:
```
X = dataset.drop("GPA", axis=1)
y = dataset[["GPA"]]
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
scaler = StandardScaler()
X_train = scaler.fit_transform(X_train)
X_test = scaler.transform(X_test)
```

Experiment 1: A single Dense Hidden Layer

In [5]:
```
# Model Structure Definition
model1 = Sequential([
    Dense(64, input_dim=13, activation='relu'),
    Dense(64, activation='relu'),
    Dense(1)
])

# Model Compile
model1.compile(optimizer='adam', loss='mean_squared_error', metrics=['mean_absolute_error'])

# Model Train
history1 = model1.fit(X_train, y_train, batch_size=10, epochs=50, validation_split=0.2)
```

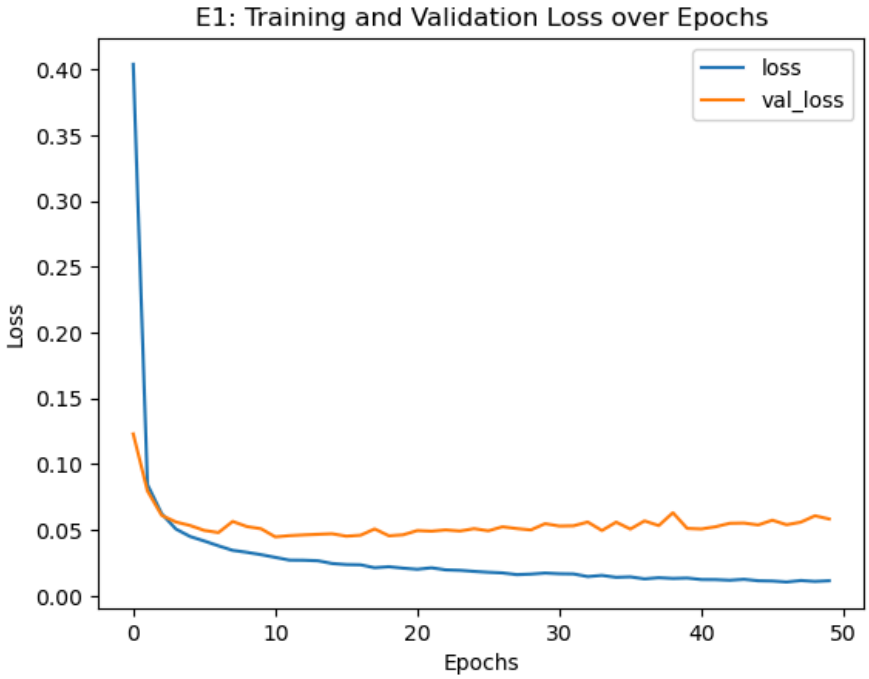
```
c:\Users\eryke\anaconda3\Lib\site-packages\keras\src\layers\core\dense.py:87: UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When using Sequential models, prefer using an `Input(shape)` object as the first layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
```

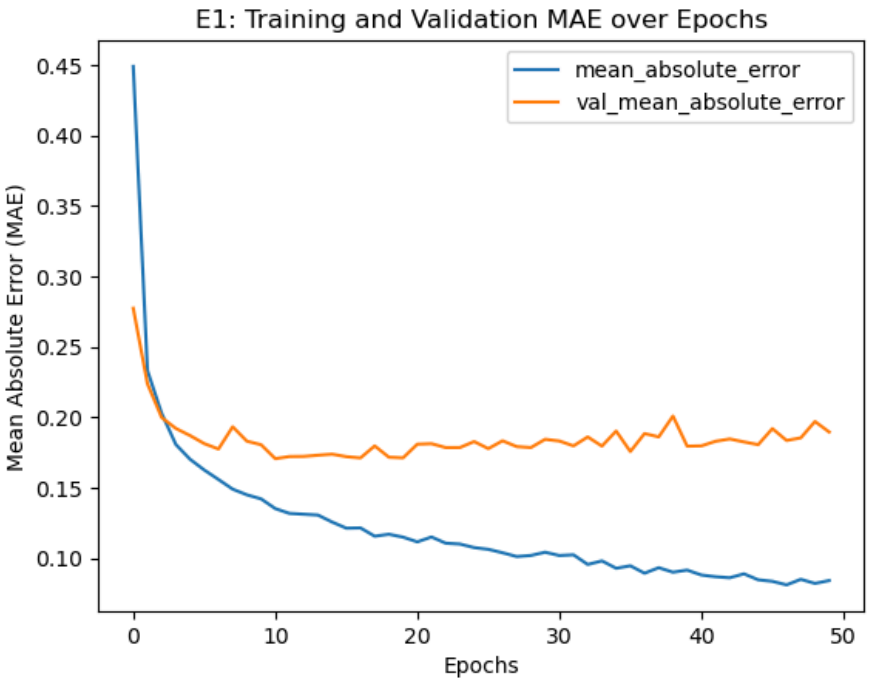
```
Epoch 1/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 4s 6ms/step - loss: 0.9356 - mean_absolute_error: 0.7175 - val_loss: 0.1230 - val_mean_absolute_error:
0.2773
Epoch 2/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - loss: 0.0917 - mean_absolute_error: 0.2449 - val_loss: 0.0797 - val_mean_absolute_error:
0.2236
Epoch 3/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - loss: 0.0680 - mean_absolute_error: 0.2099 - val_loss: 0.0611 - val_mean_absolute_error:
0.1997
Epoch 4/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - loss: 0.0501 - mean_absolute_error: 0.1802 - val_loss: 0.0561 - val_mean_absolute_error:
0.1920
Epoch 5/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - loss: 0.0434 - mean_absolute_error: 0.1689 - val_loss: 0.0534 - val_mean_absolute_error:
0.1869
Epoch 6/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 3ms/step - loss: 0.0416 - mean_absolute_error: 0.1625 - val_loss: 0.0497 - val_mean_absolute_error:
0.1812
Epoch 7/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - loss: 0.0398 - mean_absolute_error: 0.1588 - val_loss: 0.0480 - val_mean_absolute_error:
0.1773
Epoch 8/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 3ms/step - loss: 0.0347 - mean_absolute_error: 0.1489 - val_loss: 0.0565 - val_mean_absolute_error:
0.1931
Epoch 9/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - loss: 0.0352 - mean_absolute_error: 0.1490 - val_loss: 0.0526 - val_mean_absolute_error:
0.1829
Epoch 10/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 4ms/step - loss: 0.0308 - mean_absolute_error: 0.1428 - val_loss: 0.0509 - val_mean_absolute_error:
0.1804
Epoch 11/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - loss: 0.0295 - mean_absolute_error: 0.1351 - val_loss: 0.0448 - val_mean_absolute_error:
0.1705
Epoch 12/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 3ms/step - loss: 0.0259 - mean_absolute_error: 0.1268 - val_loss: 0.0457 - val_mean_absolute_error:
0.1720
Epoch 13/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - loss: 0.0262 - mean_absolute_error: 0.1284 - val_loss: 0.0463 - val_mean_absolute_error:
0.1721
Epoch 14/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - loss: 0.0256 - mean_absolute_error: 0.1288 - val_loss: 0.0467 - val_mean_absolute_error:
0.1730
Epoch 15/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - loss: 0.0239 - mean_absolute_error: 0.1201 - val_loss: 0.0471 - val_mean_absolute_error:
0.1737
Epoch 16/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - loss: 0.0239 - mean_absolute_error: 0.1205 - val_loss: 0.0453 - val_mean_absolute_error:
0.1719
Epoch 17/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - loss: 0.0216 - mean_absolute_error: 0.1151 - val_loss: 0.0460 - val_mean_absolute_error:
0.1710
Epoch 18/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - loss: 0.0213 - mean_absolute_error: 0.1165 - val_loss: 0.0508 - val_mean_absolute_error:
0.1795
Epoch 19/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - loss: 0.0198 - mean_absolute_error: 0.1108 - val_loss: 0.0456 - val_mean_absolute_error:
0.1715
Epoch 20/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - loss: 0.0200 - mean_absolute_error: 0.1134 - val_loss: 0.0463 - val_mean_absolute_error:
0.1711
Epoch 21/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - loss: 0.0176 - mean_absolute_error: 0.1039 - val_loss: 0.0495 - val_mean_absolute_error:
0.1808
Epoch 22/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - loss: 0.0216 - mean_absolute_error: 0.1156 - val_loss: 0.0491 - val_mean_absolute_error:
0.1811
Epoch 23/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - loss: 0.0178 - mean_absolute_error: 0.1045 - val_loss: 0.0501 - val_mean_absolute_error:
0.1784
Epoch 24/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - loss: 0.0185 - mean_absolute_error: 0.1082 - val_loss: 0.0492 - val_mean_absolute_error:
0.1784
Epoch 25/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - loss: 0.0163 - mean_absolute_error: 0.1019 - val_loss: 0.0510 - val_mean_absolute_error:
0.1826
Epoch 26/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - loss: 0.0179 - mean_absolute_error: 0.1065 - val_loss: 0.0494 - val_mean_absolute_error:
0.1776
Epoch 27/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - loss: 0.0155 - mean_absolute_error: 0.0982 - val_loss: 0.0524 - val_mean_absolute_error:
```

```
0.1832
Epoch 28/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 4ms/step - loss: 0.0156 - mean_absolute_error: 0.0996 - val_loss: 0.0511 - val_mean_absolute_error:
0.1791
Epoch 29/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - loss: 0.0162 - mean_absolute_error: 0.1012 - val_loss: 0.0500 - val_mean_absolute_error:
0.1784
Epoch 30/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - loss: 0.0157 - mean_absolute_error: 0.0996 - val_loss: 0.0548 - val_mean_absolute_error:
0.1842
Epoch 31/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - loss: 0.0160 - mean_absolute_error: 0.0997 - val_loss: 0.0530 - val_mean_absolute_error:
0.1830
Epoch 32/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - loss: 0.0158 - mean_absolute_error: 0.1000 - val_loss: 0.0532 - val_mean_absolute_error:
0.1796
Epoch 33/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - loss: 0.0133 - mean_absolute_error: 0.0905 - val_loss: 0.0560 - val_mean_absolute_error:
0.1859
Epoch 34/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - loss: 0.0152 - mean_absolute_error: 0.0971 - val_loss: 0.0496 - val_mean_absolute_error:
0.1794
Epoch 35/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 4ms/step - loss: 0.0127 - mean_absolute_error: 0.0883 - val_loss: 0.0560 - val_mean_absolute_error:
0.1901
Epoch 36/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - loss: 0.0144 - mean_absolute_error: 0.0957 - val_loss: 0.0507 - val_mean_absolute_error:
0.1756
Epoch 37/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - loss: 0.0126 - mean_absolute_error: 0.0879 - val_loss: 0.0569 - val_mean_absolute_error:
0.1883
Epoch 38/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 3ms/step - loss: 0.0133 - mean_absolute_error: 0.0921 - val_loss: 0.0534 - val_mean_absolute_error:
0.1860
Epoch 39/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 3ms/step - loss: 0.0118 - mean_absolute_error: 0.0858 - val_loss: 0.0631 - val_mean_absolute_error:
0.2008
Epoch 40/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0154 - mean_absolute_error: 0.0993 - val_loss: 0.0512 - val_mean_absolute_error:
0.1794
Epoch 41/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0119 - mean_absolute_error: 0.0862 - val_loss: 0.0508 - val_mean_absolute_error:
0.1796
Epoch 42/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0122 - mean_absolute_error: 0.0874 - val_loss: 0.0525 - val_mean_absolute_error:
0.1828
Epoch 43/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0113 - mean_absolute_error: 0.0848 - val_loss: 0.0551 - val_mean_absolute_error:
0.1845
Epoch 44/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0131 - mean_absolute_error: 0.0907 - val_loss: 0.0553 - val_mean_absolute_error:
0.1824
Epoch 45/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0108 - mean_absolute_error: 0.0822 - val_loss: 0.0538 - val_mean_absolute_error:
0.1804
Epoch 46/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0102 - mean_absolute_error: 0.0794 - val_loss: 0.0574 - val_mean_absolute_error:
0.1918
Epoch 47/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0104 - mean_absolute_error: 0.0795 - val_loss: 0.0540 - val_mean_absolute_error:
0.1834
Epoch 48/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0101 - mean_absolute_error: 0.0801 - val_loss: 0.0559 - val_mean_absolute_error:
0.1853
Epoch 49/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0112 - mean_absolute_error: 0.0843 - val_loss: 0.0608 - val_mean_absolute_error:
0.1969
Epoch 50/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0100 - mean_absolute_error: 0.0795 - val_loss: 0.0583 - val_mean_absolute_error:
0.1894
```

In [6]:
```python
# View History
df1 = pd.DataFrame(history1.history)
df1.plot(y=['loss', 'val_loss'], title='E1: Training and Validation Loss over Epochs', xlabel='Epochs', ylabel='Loss')
df1.plot(y=['mean_absolute_error', 'val_mean_absolute_error'], title='E1: Training and Validation MAE over Epochs', xlabel='Epochs
```

Out[6]:
```
<Axes: title={'center': 'E1: Training and Validation MAE over Epochs'}, xlabel='Epochs', ylabel='Mean Absolute Error (MAE)'>
```





In [7]:
```
# Model Evaluate
loss1, mae1 = model1.evaluate(X_test, y_test)
print("Loss:", round(loss1, 4))
print("MAE:", round(mae1, 4))
```

```
15/15 ━━━━━━━━━━━━━━━━━━━━ 0s 1ms/step - loss: 0.0646 - mean_absolute_error: 0.1988
Loss: 0.0618
MAE: 0.1986
```

Experiment 2: A set of three Dense Hidden Layers

In [8]:
```
# Model Structure Definition
model2 = Sequential([
    Dense(64, input_dim=13, activation='relu'),
    Dense(64, activation='relu'),
    Dense(64, activation='relu'),
    Dense(64, activation='relu'),
    Dense(1)
])

# Model Compile
model2.compile(optimizer='adam', loss='mean_squared_error', metrics=['mean_absolute_error'])

# Model Train
history2 = model2.fit(X_train, y_train, batch_size=10, epochs=50, validation_split=0.2)
```

```
Epoch 1/50
```

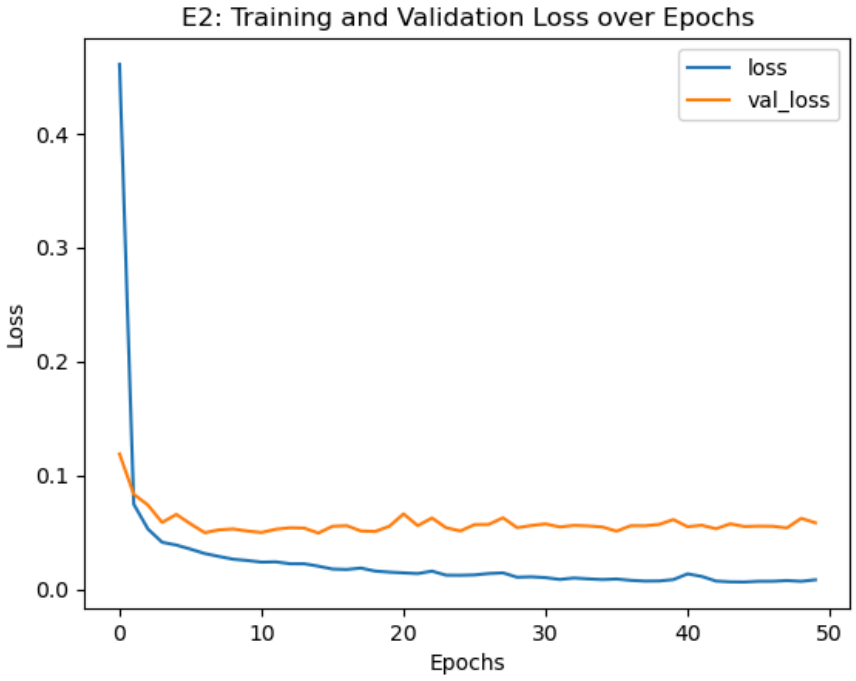
```
c:\Users\eryke\anaconda3\Lib\site-packages\keras\src\layers\core\dense.py:87: UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When using Sequential models, prefer using an `Input(shape)` object as the first layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
```

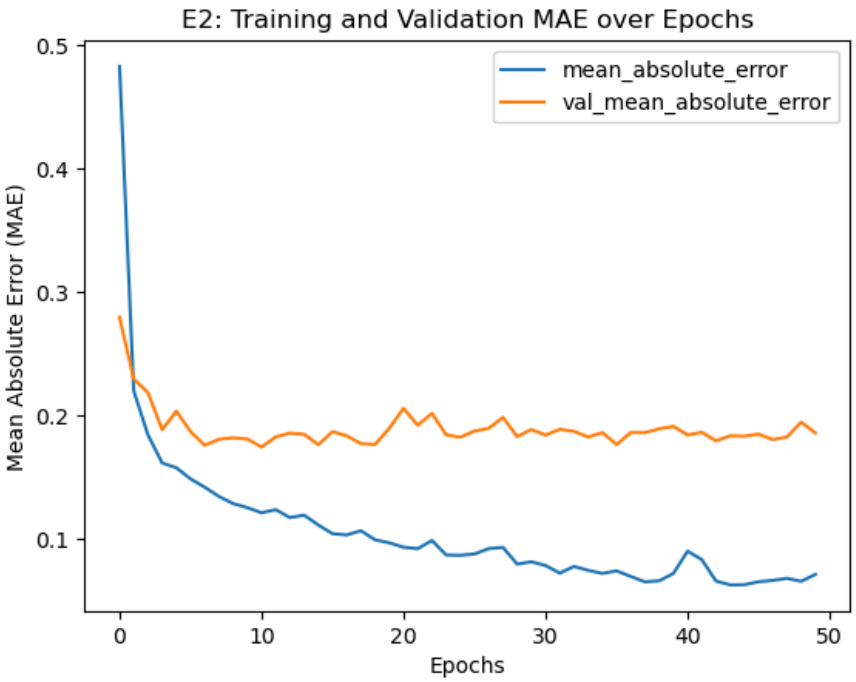
```
153/153 ━━━━━━━━━━━━━━━━━━━━ 3s 4ms/step - loss: 1.0879 - mean_absolute_error: 0.7655 - val_loss: 0.1186 - val_mean_absolute_error:
0.2793
Epoch 2/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0840 - mean_absolute_error: 0.2295 - val_loss: 0.0833 - val_mean_absolute_error:
0.2294
Epoch 3/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0510 - mean_absolute_error: 0.1798 - val_loss: 0.0738 - val_mean_absolute_error:
0.2184
Epoch 4/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0418 - mean_absolute_error: 0.1602 - val_loss: 0.0585 - val_mean_absolute_error:
0.1883
Epoch 5/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0350 - mean_absolute_error: 0.1496 - val_loss: 0.0656 - val_mean_absolute_error:
0.2033
Epoch 6/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0343 - mean_absolute_error: 0.1451 - val_loss: 0.0573 - val_mean_absolute_error:
0.1868
Epoch 7/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0314 - mean_absolute_error: 0.1425 - val_loss: 0.0495 - val_mean_absolute_error:
0.1756
Epoch 8/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0267 - mean_absolute_error: 0.1304 - val_loss: 0.0520 - val_mean_absolute_error:
0.1806
Epoch 9/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0266 - mean_absolute_error: 0.1292 - val_loss: 0.0528 - val_mean_absolute_error:
0.1817
Epoch 10/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0244 - mean_absolute_error: 0.1242 - val_loss: 0.0510 - val_mean_absolute_error:
0.1808
Epoch 11/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0221 - mean_absolute_error: 0.1171 - val_loss: 0.0497 - val_mean_absolute_error:
0.1742
Epoch 12/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0210 - mean_absolute_error: 0.1158 - val_loss: 0.0525 - val_mean_absolute_error:
0.1825
Epoch 13/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0214 - mean_absolute_error: 0.1148 - val_loss: 0.0539 - val_mean_absolute_error:
0.1856
Epoch 14/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0200 - mean_absolute_error: 0.1115 - val_loss: 0.0536 - val_mean_absolute_error:
0.1846
Epoch 15/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0208 - mean_absolute_error: 0.1126 - val_loss: 0.0492 - val_mean_absolute_error:
0.1762
Epoch 16/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0160 - mean_absolute_error: 0.1012 - val_loss: 0.0552 - val_mean_absolute_error:
0.1868
Epoch 17/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 3ms/step - loss: 0.0177 - mean_absolute_error: 0.1037 - val_loss: 0.0557 - val_mean_absolute_error:
0.1833
Epoch 18/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 3ms/step - loss: 0.0168 - mean_absolute_error: 0.1012 - val_loss: 0.0510 - val_mean_absolute_error:
0.1770
Epoch 19/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0147 - mean_absolute_error: 0.0950 - val_loss: 0.0506 - val_mean_absolute_error:
0.1764
Epoch 20/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0138 - mean_absolute_error: 0.0934 - val_loss: 0.0551 - val_mean_absolute_error:
0.1895
Epoch 21/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0139 - mean_absolute_error: 0.0922 - val_loss: 0.0660 - val_mean_absolute_error:
0.2056
Epoch 22/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0137 - mean_absolute_error: 0.0929 - val_loss: 0.0557 - val_mean_absolute_error:
0.1919
Epoch 23/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0168 - mean_absolute_error: 0.1009 - val_loss: 0.0624 - val_mean_absolute_error:
0.2015
Epoch 24/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 3ms/step - loss: 0.0131 - mean_absolute_error: 0.0903 - val_loss: 0.0539 - val_mean_absolute_error:
0.1841
Epoch 25/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0112 - mean_absolute_error: 0.0836 - val_loss: 0.0511 - val_mean_absolute_error:
0.1823
Epoch 26/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 3ms/step - loss: 0.0104 - mean_absolute_error: 0.0800 - val_loss: 0.0566 - val_mean_absolute_error:
0.1872
Epoch 27/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0122 - mean_absolute_error: 0.0859 - val_loss: 0.0568 - val_mean_absolute_error:
0.1894
```

```
Epoch 28/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0158 - mean_absolute_error: 0.0965 - val_loss: 0.0626 - val_mean_absolute_error:
0.1983
Epoch 29/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 3ms/step - loss: 0.0112 - mean_absolute_error: 0.0823 - val_loss: 0.0539 - val_mean_absolute_error:
0.1826
Epoch 30/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0099 - mean_absolute_error: 0.0786 - val_loss: 0.0558 - val_mean_absolute_error:
0.1885
Epoch 31/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0088 - mean_absolute_error: 0.0735 - val_loss: 0.0572 - val_mean_absolute_error:
0.1840
Epoch 32/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0082 - mean_absolute_error: 0.0710 - val_loss: 0.0546 - val_mean_absolute_error:
0.1886
Epoch 33/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 3ms/step - loss: 0.0087 - mean_absolute_error: 0.0727 - val_loss: 0.0559 - val_mean_absolute_error:
0.1870
Epoch 34/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 3ms/step - loss: 0.0088 - mean_absolute_error: 0.0737 - val_loss: 0.0555 - val_mean_absolute_error:
0.1824
Epoch 35/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0083 - mean_absolute_error: 0.0711 - val_loss: 0.0545 - val_mean_absolute_error:
0.1860
Epoch 36/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - loss: 0.0076 - mean_absolute_error: 0.0681 - val_loss: 0.0509 - val_mean_absolute_error:
0.1763
Epoch 37/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 3ms/step - loss: 0.0074 - mean_absolute_error: 0.0677 - val_loss: 0.0557 - val_mean_absolute_error:
0.1860
Epoch 38/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0059 - mean_absolute_error: 0.0594 - val_loss: 0.0556 - val_mean_absolute_error:
0.1859
Epoch 39/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0076 - mean_absolute_error: 0.0673 - val_loss: 0.0568 - val_mean_absolute_error:
0.1891
Epoch 40/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 3ms/step - loss: 0.0081 - mean_absolute_error: 0.0706 - val_loss: 0.0610 - val_mean_absolute_error:
0.1910
Epoch 41/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0117 - mean_absolute_error: 0.0844 - val_loss: 0.0548 - val_mean_absolute_error:
0.1840
Epoch 42/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0128 - mean_absolute_error: 0.0897 - val_loss: 0.0561 - val_mean_absolute_error:
0.1862
Epoch 43/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0065 - mean_absolute_error: 0.0630 - val_loss: 0.0530 - val_mean_absolute_error:
0.1793
Epoch 44/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0059 - mean_absolute_error: 0.0605 - val_loss: 0.0573 - val_mean_absolute_error:
0.1834
Epoch 45/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0065 - mean_absolute_error: 0.0641 - val_loss: 0.0550 - val_mean_absolute_error:
0.1831
Epoch 46/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0066 - mean_absolute_error: 0.0645 - val_loss: 0.0553 - val_mean_absolute_error:
0.1848
Epoch 47/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0068 - mean_absolute_error: 0.0662 - val_loss: 0.0552 - val_mean_absolute_error:
0.1803
Epoch 48/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0067 - mean_absolute_error: 0.0639 - val_loss: 0.0537 - val_mean_absolute_error:
0.1824
Epoch 49/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0068 - mean_absolute_error: 0.0656 - val_loss: 0.0621 - val_mean_absolute_error:
0.1944
Epoch 50/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.0073 - mean_absolute_error: 0.0673 - val_loss: 0.0582 - val_mean_absolute_error:
0.1857
```

In [9]:
```
# View History
df2 = pd.DataFrame(history2.history)
df2.plot(y=['loss', 'val_loss'], title='E2: Training and Validation Loss over Epochs', xlabel='Epochs', ylabel='Loss')
df2.plot(y=['mean_absolute_error', 'val_mean_absolute_error'], title='E2: Training and Validation MAE over Epochs', xlabel='Epochs
```

Out[9]:
```
<Axes: title={'center': 'E2: Training and Validation MAE over Epochs'}, xlabel='Epochs', ylabel='Mean Absolute Error (MAE)'>
```





In [10]:
```
# Model Evaluate
loss2, mae2 = model2.evaluate(X_test, y_test)
print("Loss:", round(loss2, 4))
print("MAE:", round(mae2, 4))
```

```
15/15 ━━━━━━━━━━━━━━━━━━━━ 0s 1ms/step - loss: 0.0667 - mean_absolute_error: 0.2013
Loss: 0.0618
MAE: 0.1927
```

Experiment 3: Add a dropout layer after each Dense Hidden Layer

In [11]:
```
# Model Structure Definition
model3 = Sequential([
    Dense(64, input_dim=13, activation='relu'),
    Dense(64, activation='relu'),
    Dropout(0.5),
    Dense(64, activation='relu'),
    Dropout(0.5),
    Dense(64, activation='relu'),
    Dropout(0.5),
    Dense(1)
])

# Model Compile
model3.compile(optimizer='adam', loss='mean_squared_error', metrics=['mean_absolute_error'])
```

```
# Model Train
history3 = model3.fit(X_train, y_train, batch_size=10, epochs=50, validation_split=0.2)
```

```
Epoch 1/50
```

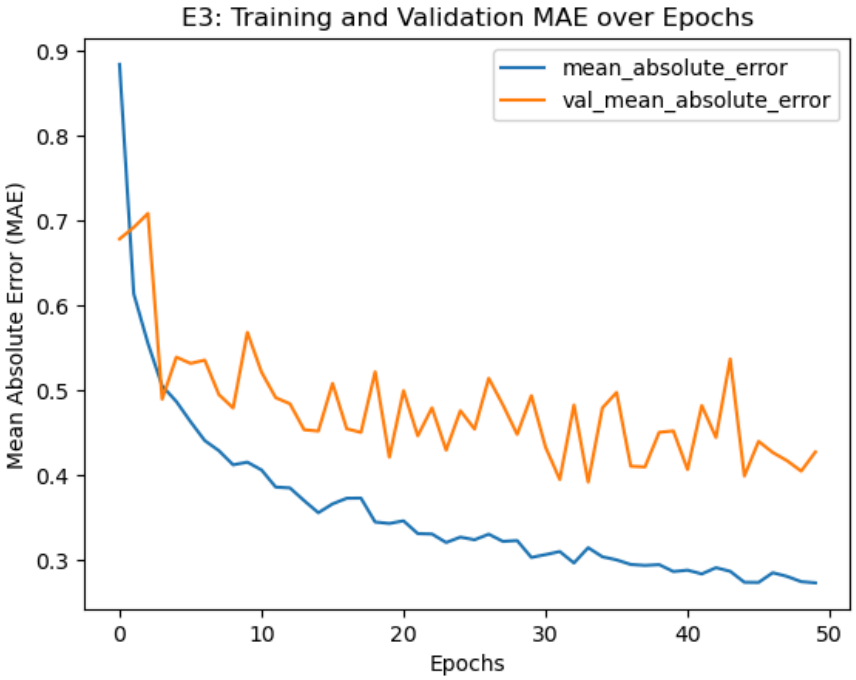
```
c:\Users\eryke\anaconda3\Lib\site-packages\keras\src\layers\core\dense.py:87: UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When using Sequential models, prefer using an `Input(shape)` object as the first layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
```

```
153/153 ━━━━━━━━━━━━━━━━━━━━ 3s 4ms/step - loss: 1.7526 - mean_absolute_error: 1.0710 - val_loss: 0.6120 - val_mean_absolute_error:
0.6783
Epoch 2/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.6840 - mean_absolute_error: 0.6484 - val_loss: 0.6485 - val_mean_absolute_error:
0.6920
Epoch 3/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.5443 - mean_absolute_error: 0.5587 - val_loss: 0.6736 - val_mean_absolute_error:
0.7084
Epoch 4/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.4593 - mean_absolute_error: 0.5191 - val_loss: 0.3187 - val_mean_absolute_error:
0.4894
Epoch 5/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.3786 - mean_absolute_error: 0.4763 - val_loss: 0.3945 - val_mean_absolute_error:
0.5391
Epoch 6/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.3630 - mean_absolute_error: 0.4794 - val_loss: 0.3858 - val_mean_absolute_error:
0.5317
Epoch 7/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.3291 - mean_absolute_error: 0.4412 - val_loss: 0.3898 - val_mean_absolute_error:
0.5356
Epoch 8/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 3ms/step - loss: 0.3236 - mean_absolute_error: 0.4369 - val_loss: 0.3418 - val_mean_absolute_error:
0.4949
Epoch 9/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.3048 - mean_absolute_error: 0.4215 - val_loss: 0.3205 - val_mean_absolute_error:
0.4792
Epoch 10/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.3106 - mean_absolute_error: 0.4213 - val_loss: 0.4574 - val_mean_absolute_error:
0.5683
Epoch 11/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 3ms/step - loss: 0.2762 - mean_absolute_error: 0.4091 - val_loss: 0.3960 - val_mean_absolute_error:
0.5213
Epoch 12/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 3ms/step - loss: 0.2592 - mean_absolute_error: 0.3804 - val_loss: 0.3411 - val_mean_absolute_error:
0.4913
Epoch 13/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.2441 - mean_absolute_error: 0.3798 - val_loss: 0.3353 - val_mean_absolute_error:
0.4841
Epoch 14/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.2504 - mean_absolute_error: 0.3789 - val_loss: 0.2967 - val_mean_absolute_error:
0.4535
Epoch 15/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 3ms/step - loss: 0.2079 - mean_absolute_error: 0.3523 - val_loss: 0.2970 - val_mean_absolute_error:
0.4520
Epoch 16/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 3ms/step - loss: 0.2317 - mean_absolute_error: 0.3625 - val_loss: 0.3792 - val_mean_absolute_error:
0.5082
Epoch 17/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.2566 - mean_absolute_error: 0.3781 - val_loss: 0.3024 - val_mean_absolute_error:
0.4546
Epoch 18/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 3ms/step - loss: 0.2178 - mean_absolute_error: 0.3544 - val_loss: 0.3042 - val_mean_absolute_error:
0.4504
Epoch 19/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.1909 - mean_absolute_error: 0.3386 - val_loss: 0.4000 - val_mean_absolute_error:
0.5219
Epoch 20/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.1913 - mean_absolute_error: 0.3311 - val_loss: 0.2631 - val_mean_absolute_error:
0.4211
Epoch 21/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.2246 - mean_absolute_error: 0.3572 - val_loss: 0.3738 - val_mean_absolute_error:
0.4996
Epoch 22/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.1952 - mean_absolute_error: 0.3376 - val_loss: 0.2900 - val_mean_absolute_error:
0.4464
Epoch 23/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.1855 - mean_absolute_error: 0.3269 - val_loss: 0.3255 - val_mean_absolute_error:
0.4793
Epoch 24/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.1674 - mean_absolute_error: 0.3112 - val_loss: 0.2651 - val_mean_absolute_error:
0.4294
Epoch 25/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.1838 - mean_absolute_error: 0.3299 - val_loss: 0.3310 - val_mean_absolute_error:
0.4760
Epoch 26/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.1610 - mean_absolute_error: 0.3122 - val_loss: 0.2925 - val_mean_absolute_error:
0.4543
Epoch 27/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 3ms/step - loss: 0.1781 - mean_absolute_error: 0.3124 - val_loss: 0.3891 - val_mean_absolute_error:
0.5142
```

```
Epoch 28/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.1806 - mean_absolute_error: 0.3234 - val_loss: 0.3353 - val_mean_absolute_error:
0.4827
Epoch 29/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.1891 - mean_absolute_error: 0.3274 - val_loss: 0.2916 - val_mean_absolute_error:
0.4481
Epoch 30/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.1703 - mean_absolute_error: 0.3118 - val_loss: 0.3540 - val_mean_absolute_error:
0.4937
Epoch 31/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.1667 - mean_absolute_error: 0.3111 - val_loss: 0.2793 - val_mean_absolute_error:
0.4329
Epoch 32/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 3ms/step - loss: 0.1781 - mean_absolute_error: 0.3112 - val_loss: 0.2292 - val_mean_absolute_error:
0.3949
Epoch 33/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 0.1489 - mean_absolute_error: 0.2869 - val_loss: 0.3542 - val_mean_absolute_error:
0.4827
Epoch 34/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 0s 3ms/step - loss: 0.1794 - mean_absolute_error: 0.3221 - val_loss: 0.2275 - val_mean_absolute_error:
0.3918
Epoch 35/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - loss: 0.1641 - mean_absolute_error: 0.3138 - val_loss: 0.3393 - val_mean_absolute_error:
0.4792
Epoch 36/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 4ms/step - loss: 0.1566 - mean_absolute_error: 0.2986 - val_loss: 0.3623 - val_mean_absolute_error:
0.4975
Epoch 37/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 4ms/step - loss: 0.1555 - mean_absolute_error: 0.2984 - val_loss: 0.2542 - val_mean_absolute_error:
0.4106
Epoch 38/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 4ms/step - loss: 0.1682 - mean_absolute_error: 0.3109 - val_loss: 0.2460 - val_mean_absolute_error:
0.4096
Epoch 39/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 4ms/step - loss: 0.1548 - mean_absolute_error: 0.2977 - val_loss: 0.3017 - val_mean_absolute_error:
0.4505
Epoch 40/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 4ms/step - loss: 0.1405 - mean_absolute_error: 0.2870 - val_loss: 0.2993 - val_mean_absolute_error:
0.4521
Epoch 41/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 4ms/step - loss: 0.1393 - mean_absolute_error: 0.2819 - val_loss: 0.2467 - val_mean_absolute_error:
0.4066
Epoch 42/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 4ms/step - loss: 0.1479 - mean_absolute_error: 0.2914 - val_loss: 0.3469 - val_mean_absolute_error:
0.4820
Epoch 43/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 4ms/step - loss: 0.1506 - mean_absolute_error: 0.2952 - val_loss: 0.2957 - val_mean_absolute_error:
0.4444
Epoch 44/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 4ms/step - loss: 0.1431 - mean_absolute_error: 0.2879 - val_loss: 0.4417 - val_mean_absolute_error:
0.5369
Epoch 45/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 4ms/step - loss: 0.1276 - mean_absolute_error: 0.2781 - val_loss: 0.2363 - val_mean_absolute_error:
0.3990
Epoch 46/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 4ms/step - loss: 0.1288 - mean_absolute_error: 0.2729 - val_loss: 0.2923 - val_mean_absolute_error:
0.4399
Epoch 47/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 4ms/step - loss: 0.1403 - mean_absolute_error: 0.2879 - val_loss: 0.2698 - val_mean_absolute_error:
0.4267
Epoch 48/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 4ms/step - loss: 0.1385 - mean_absolute_error: 0.2777 - val_loss: 0.2663 - val_mean_absolute_error:
0.4171
Epoch 49/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 4ms/step - loss: 0.1193 - mean_absolute_error: 0.2668 - val_loss: 0.2449 - val_mean_absolute_error:
0.4048
Epoch 50/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - loss: 0.1326 - mean_absolute_error: 0.2754 - val_loss: 0.2747 - val_mean_absolute_error:
0.4273
```

In [12]:
```python
# View History
df3 = pd.DataFrame(history3.history)
df3.plot(y=['loss', 'val_loss'], title='E3: Training and Validation Loss over Epochs', xlabel='Epochs', ylabel='Loss')
df3.plot(y=['mean_absolute_error', 'val_mean_absolute_error'], title='E3: Training and Validation MAE over Epochs', xlabel='Epochs
```

Out[12]:
```
<Axes: title={'center': 'E3: Training and Validation MAE over Epochs'}, xlabel='Epochs', ylabel='Mean Absolute Error (MAE)'>
```



In [13]:
```
# Model Evaluate
loss3, mae3 = model3.evaluate(X_test, y_test)
print("Loss:", round(loss3, 4))
print("MAE:", round(mae3, 4))
```

```
15/15 ━━━━━━━━━━━━━━━━━━━━ 0s 3ms/step - loss: 0.2535 - mean_absolute_error: 0.3981
Loss: 0.2547
MAE: 0.4039
```

Experiment 4: Add a Batch Normalization Layer after each Dropout Layer

In [14]:
```
# Model Structure Definition
model4 = Sequential([
    Dense(64, input_dim=13, activation='relu'),
    Dense(64, activation='relu'),
    Dropout(0.5),
    BatchNormalization(),
    Dense(64, activation='relu'),
    Dropout(0.5),
    BatchNormalization(),
    Dense(64, activation='relu'),
    Dropout(0.5),
    BatchNormalization(),
    Dense(1)
])
```

```python
# Model Compile
model4.compile(optimizer='adam', loss='mean_squared_error', metrics=['mean_absolute_error'])

# Model Train
history4 = model4.fit(X_train, y_train, batch_size=10, epochs=50, validation_split=0.2)
```

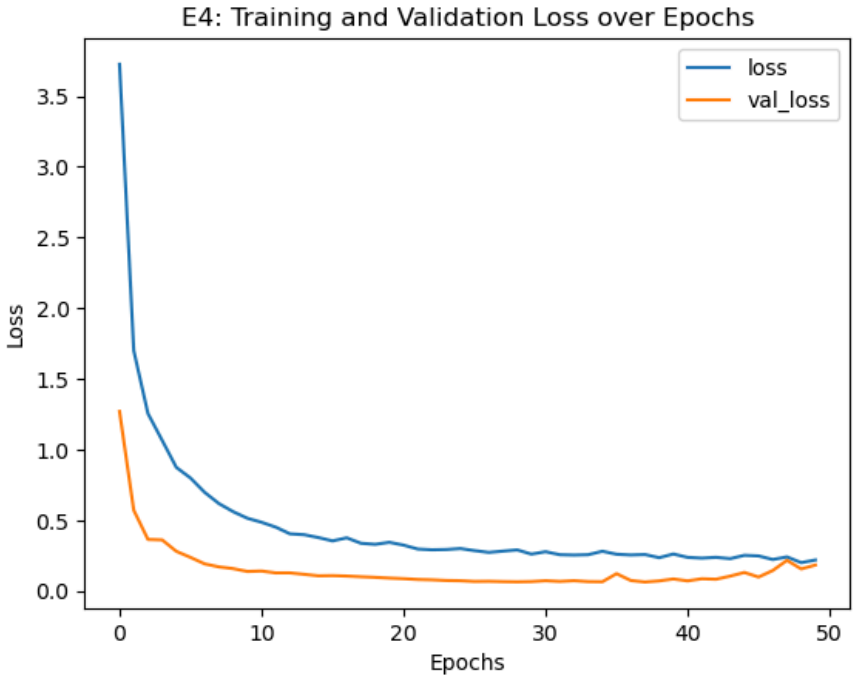
```
c:\Users\eryke\anaconda3\Lib\site-packages\keras\src\layers\core\dense.py:87: UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When using Sequential models, prefer using an `Input(shape)` object as the first layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
```

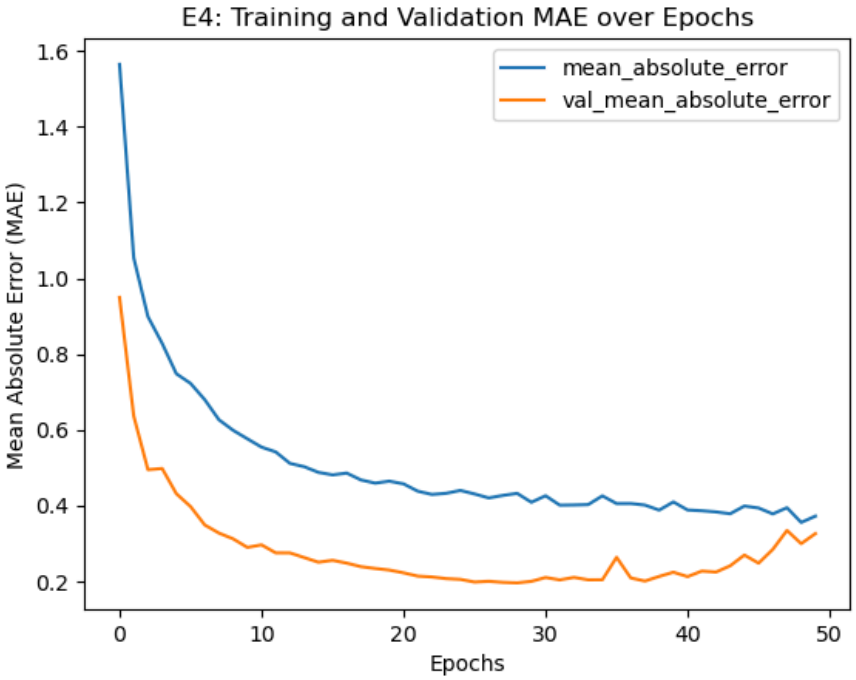
```
Epoch 1/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 11s 10ms/step - loss: 4.7467 - mean_absolute_error: 1.7937 - val_loss: 1.2692 - val_mean_absolute_erro
r: 0.9494
Epoch 2/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 2s 5ms/step - loss: 1.8716 - mean_absolute_error: 1.0954 - val_loss: 0.5704 - val_mean_absolute_error:
0.6361
Epoch 3/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 5ms/step - loss: 1.3143 - mean_absolute_error: 0.9229 - val_loss: 0.3639 - val_mean_absolute_error:
0.4945
Epoch 4/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 5ms/step - loss: 1.1469 - mean_absolute_error: 0.8576 - val_loss: 0.3606 - val_mean_absolute_error:
0.4977
Epoch 5/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 5ms/step - loss: 0.9079 - mean_absolute_error: 0.7610 - val_loss: 0.2796 - val_mean_absolute_error:
0.4311
Epoch 6/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 5ms/step - loss: 0.8589 - mean_absolute_error: 0.7590 - val_loss: 0.2366 - val_mean_absolute_error:
0.3973
Epoch 7/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 5ms/step - loss: 0.7142 - mean_absolute_error: 0.6944 - val_loss: 0.1907 - val_mean_absolute_error:
0.3487
Epoch 8/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 5ms/step - loss: 0.6327 - mean_absolute_error: 0.6353 - val_loss: 0.1697 - val_mean_absolute_error:
0.3272
Epoch 9/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 5ms/step - loss: 0.5308 - mean_absolute_error: 0.5824 - val_loss: 0.1573 - val_mean_absolute_error:
0.3119
Epoch 10/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 4ms/step - loss: 0.5182 - mean_absolute_error: 0.5829 - val_loss: 0.1373 - val_mean_absolute_error:
0.2894
Epoch 11/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 4ms/step - loss: 0.4805 - mean_absolute_error: 0.5498 - val_loss: 0.1402 - val_mean_absolute_error:
0.2961
Epoch 12/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 5ms/step - loss: 0.4503 - mean_absolute_error: 0.5378 - val_loss: 0.1272 - val_mean_absolute_error:
0.2749
Epoch 13/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 5ms/step - loss: 0.4124 - mean_absolute_error: 0.5200 - val_loss: 0.1272 - val_mean_absolute_error:
0.2749
Epoch 14/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 5ms/step - loss: 0.3903 - mean_absolute_error: 0.4998 - val_loss: 0.1170 - val_mean_absolute_error:
0.2626
Epoch 15/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 5ms/step - loss: 0.3827 - mean_absolute_error: 0.4906 - val_loss: 0.1066 - val_mean_absolute_error:
0.2503
Epoch 16/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 5ms/step - loss: 0.3417 - mean_absolute_error: 0.4748 - val_loss: 0.1078 - val_mean_absolute_error:
0.2552
Epoch 17/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 5ms/step - loss: 0.3792 - mean_absolute_error: 0.4927 - val_loss: 0.1047 - val_mean_absolute_error:
0.2479
Epoch 18/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 5ms/step - loss: 0.3247 - mean_absolute_error: 0.4590 - val_loss: 0.0999 - val_mean_absolute_error:
0.2385
Epoch 19/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 5ms/step - loss: 0.3290 - mean_absolute_error: 0.4563 - val_loss: 0.0956 - val_mean_absolute_error:
0.2337
Epoch 20/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 5ms/step - loss: 0.3289 - mean_absolute_error: 0.4592 - val_loss: 0.0904 - val_mean_absolute_error:
0.2296
Epoch 21/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 5ms/step - loss: 0.3199 - mean_absolute_error: 0.4562 - val_loss: 0.0862 - val_mean_absolute_error:
0.2223
Epoch 22/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 4ms/step - loss: 0.3018 - mean_absolute_error: 0.4448 - val_loss: 0.0807 - val_mean_absolute_error:
0.2136
Epoch 23/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 5ms/step - loss: 0.2953 - mean_absolute_error: 0.4347 - val_loss: 0.0782 - val_mean_absolute_error:
0.2114
Epoch 24/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 5ms/step - loss: 0.3082 - mean_absolute_error: 0.4462 - val_loss: 0.0734 - val_mean_absolute_error:
0.2070
Epoch 25/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 4ms/step - loss: 0.2927 - mean_absolute_error: 0.4330 - val_loss: 0.0718 - val_mean_absolute_error:
0.2050
Epoch 26/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 5ms/step - loss: 0.2869 - mean_absolute_error: 0.4301 - val_loss: 0.0667 - val_mean_absolute_error:
0.1979
Epoch 27/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 4ms/step - loss: 0.2606 - mean_absolute_error: 0.4092 - val_loss: 0.0678 - val_mean_absolute_error:
```

```
0.1999
Epoch 28/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 5ms/step - loss: 0.2815 - mean_absolute_error: 0.4268 - val_loss: 0.0653 - val_mean_absolute_error:
0.1970
Epoch 29/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 5ms/step - loss: 0.3037 - mean_absolute_error: 0.4424 - val_loss: 0.0639 - val_mean_absolute_error:
0.1958
Epoch 30/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 5ms/step - loss: 0.2273 - mean_absolute_error: 0.3811 - val_loss: 0.0657 - val_mean_absolute_error:
0.1997
Epoch 31/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 4ms/step - loss: 0.2736 - mean_absolute_error: 0.4293 - val_loss: 0.0719 - val_mean_absolute_error:
0.2099
Epoch 32/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 4ms/step - loss: 0.2580 - mean_absolute_error: 0.4013 - val_loss: 0.0669 - val_mean_absolute_error:
0.2036
Epoch 33/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 4ms/step - loss: 0.2469 - mean_absolute_error: 0.3952 - val_loss: 0.0721 - val_mean_absolute_error:
0.2101
Epoch 34/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 5ms/step - loss: 0.2607 - mean_absolute_error: 0.4063 - val_loss: 0.0655 - val_mean_absolute_error:
0.2038
Epoch 35/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 5ms/step - loss: 0.2640 - mean_absolute_error: 0.4124 - val_loss: 0.0644 - val_mean_absolute_error:
0.2039
Epoch 36/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 5ms/step - loss: 0.2651 - mean_absolute_error: 0.4088 - val_loss: 0.1216 - val_mean_absolute_error:
0.2632
Epoch 37/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 5ms/step - loss: 0.2631 - mean_absolute_error: 0.4139 - val_loss: 0.0734 - val_mean_absolute_error:
0.2086
Epoch 38/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 5ms/step - loss: 0.2694 - mean_absolute_error: 0.4129 - val_loss: 0.0626 - val_mean_absolute_error:
0.2006
Epoch 39/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 5ms/step - loss: 0.2347 - mean_absolute_error: 0.3902 - val_loss: 0.0706 - val_mean_absolute_error:
0.2127
Epoch 40/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 5ms/step - loss: 0.2537 - mean_absolute_error: 0.4081 - val_loss: 0.0842 - val_mean_absolute_error:
0.2239
Epoch 41/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 5ms/step - loss: 0.2197 - mean_absolute_error: 0.3783 - val_loss: 0.0707 - val_mean_absolute_error:
0.2122
Epoch 42/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 4ms/step - loss: 0.2348 - mean_absolute_error: 0.3890 - val_loss: 0.0856 - val_mean_absolute_error:
0.2269
Epoch 43/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 4ms/step - loss: 0.2420 - mean_absolute_error: 0.3846 - val_loss: 0.0823 - val_mean_absolute_error:
0.2243
Epoch 44/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 5ms/step - loss: 0.2373 - mean_absolute_error: 0.3836 - val_loss: 0.1042 - val_mean_absolute_error:
0.2408
Epoch 45/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 4ms/step - loss: 0.2535 - mean_absolute_error: 0.3984 - val_loss: 0.1302 - val_mean_absolute_error:
0.2692
Epoch 46/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 5ms/step - loss: 0.2529 - mean_absolute_error: 0.3954 - val_loss: 0.0977 - val_mean_absolute_error:
0.2477
Epoch 47/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 5ms/step - loss: 0.2137 - mean_absolute_error: 0.3747 - val_loss: 0.1441 - val_mean_absolute_error:
0.2841
Epoch 48/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 5ms/step - loss: 0.2272 - mean_absolute_error: 0.3803 - val_loss: 0.2160 - val_mean_absolute_error:
0.3342
Epoch 49/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 5ms/step - loss: 0.1973 - mean_absolute_error: 0.3523 - val_loss: 0.1551 - val_mean_absolute_error:
0.2993
Epoch 50/50
153/153 ━━━━━━━━━━━━━━━━━━━━ 1s 5ms/step - loss: 0.2159 - mean_absolute_error: 0.3686 - val_loss: 0.1826 - val_mean_absolute_error:
0.3258
```

In [15]:
```python
# View History
df4 = pd.DataFrame(history4.history)
df4.plot(y=['loss', 'val_loss'], title='E4: Training and Validation Loss over Epochs', xlabel='Epochs', ylabel='Loss')
df4.plot(y=['mean_absolute_error', 'val_mean_absolute_error'], title='E4: Training and Validation MAE over Epochs', xlabel='Epochs
```

Out[15]:
```
<Axes: title={'center': 'E4: Training and Validation MAE over Epochs'}, xlabel='Epochs', ylabel='Mean Absolute Error (MAE)'>
```





In [16]:
```
# Model Evaluate
loss4, mae4 = model4.evaluate(X_test, y_test)
print("Loss:", round(loss4, 4))
print("MAE:", round(mae4, 4))
```

```
15/15 ━━━━━━━━━━━━━━━━━━━━ 0s 4ms/step - loss: 0.1603 - mean_absolute_error: 0.2984
Loss: 0.1411
MAE: 0.2813
```

Model Comparison

In [17]:
```
head = ["E#", "Loss", "MAE"]
mydata = [
    ["E1", round(loss1, 4), round(mae1, 4)],
    ["E2", round(loss2, 4), round(mae2, 4)],
    ["E3", round(loss3, 4), round(mae3, 4)],
    ["E4", round(loss4, 4), round(mae4, 4)]
]
print(tabulate(mydata, headers=head, tablefmt="grid"))
```

E#	Loss	MAE
E1	0.0618	0.1986
E2	0.0618	0.1927
E3	0.2547	0.4039
E4	0.1411	0.2813

We observe that the best model with the lowest mean squared error loss and mean absolute error metric is 'Experiment 2: A set of three Dense Hidden Layers'. This is because it builds uppon the success of the first model by adding additional hidden layers, allowing the model to better train with the data provided. In comparison, Experiment 3: Add a dropout layer after each Dense Hidden Layer and Experiment 4: Add a Batch Normalization Layer after each Dropout Layer didn't perform as well as the previous experiments. This is because the addition of Dropout Layers in experiments 3 and 4, and the addition of Batch Normalization Layers in experiment 4 actually impacted the performance of the models negatively by reducing the already small amount of data and limiting the learning capabilities of the models, resulting in these decreases in performance in comparison to previous models.