Group No 97

Group Member Names

No.	Member	Student ID
1	Ashan Perera	2021/E/088
2	Chanuka Fernando	2021/E/042
3	Kithmini Herath	2021/E/064

1. Problem Statement

Students are expected to develop a classification regression problem of your choice. You have to detail the problem under the heading which basically addresses the following questions:

1. What is the problem you are trying to solve?
2. Why is it a classification or regression problem? Why is it an interesting problem?

DIGITAL PART YOU WILL DEAL WITH IS BASICALLY CATEGORICAL DATA only

Select at least 10 attributes

Scale > 2 Nominal > 8 Ordinal > 0 binary

```
[nan, 2, 2, 2, 2, 2, 2, 2, 2, 2]
['gender', 'ethnicity', 'parental_level_of_education', 'lunch', 'test_preparation_course', 'math_s
['female', 'group A', 'bachelor's degree', 'standard', 'none', '71', '74', '78']
['female', 'group B', 'master's degree', 'free/reduced', 'completed', '70', '70', '75']
['female', 'group C', 'some college', 'standard', 'none', '80', '78', '77']
['male', 'group A', 'associate's degree', 'free/reduced', 'none', '47', '67', '67']
['male', 'group C', 'some college', 'standard', 'none', '76', '78', '75']
['female', 'group B', 'associate's degree', 'standard', 'completed', '59', '61', '65']
['female', 'group B', 'some college', 'standard', 'completed', '77', '73', '76']
['female', 'group B', 'some college', 'free/reduced', 'none', '62', '64', '58']
['female', 'group B', 'master's degree', 'standard', 'none', '73', '70', '74']
['female', 'group B', 'master's degree', 'standard', 'completed', '93', '96', '88']
['female', 'group B', 'master's degree', 'free/reduced', 'completed', '90', '93', '91']
['female', 'group B', 'high school', 'free/reduced', 'none', '49', '54', '52']
['female', 'group B', 'high school', 'free/reduced', 'none', '54', '58', '51']
['female', 'group B', 'bachelor's degree', 'standard', 'none', '76', '81', '75']
```

Student Performance Prediction

In this problem, we are trying to predict the scores of students in these exams given information regarding their demographics and other factors such as parental education, lunch, and test preparation.

This problem is a classification problem because the target variables can be seen as a pre-processed measure to bucket the students's grades in a manner which is easier to handle.

This problem is an interesting one because it is trying to model the up or comparison problem. As we have three exam scores, we can also use it to predict which student is better at which subject.

2. Data Acquisition

For the problem identified by you, students have to find the data source themselves from any data source.

2.1 Download the data directly

Since we're downloading data from kaggle, we can use the kaggle API to fetch the data. For that we will need to install the kaggle library and then authenticate our access using our API key. We can then write a predefined script which is used to download and unzip the data from our kaggle profile into the following directory.

```
!pip install -q kaggle
# Make a directory for the data
!mkdir ~/data/kaggle/student_exam
cd ~/data/kaggle/student_exam
# Download the dataset using the kaggle API
kaggle datasets download -d spscientist/students-performance-in-exams
# Unzip the downloaded dataset and delete the zip
unzip -q students-performance-in-exams.zip
rm students-performance-in-exams.zip
```

2.2 Code for converting the above downloaded data into a form suitable for DL

The data is in the form of a csv file and we can conveniently read the loading our data loading requirements model.

2.3 Write your observations from the above.

1. Size of the dataset
2. What type of data attributes are there?

Scale > 2 Nominal > 8 Ordinal > 0 binary

```
!wc -l ~/data/kaggle/student_exam/StudentsPerformance.csv
1001 /home/ubuntu/data/kaggle/student_exam/StudentsPerformance.csv
# Head command to see the first few lines of the dataset
!head ~/data/kaggle/student_exam/StudentsPerformance.csv
gender,race/ethnicity,parental level of education,lunch,test preparation course,math score,reading
female,group B,bachelor's degree,standard,none,72,72,74
female,group C,some college,standard,completed,69,90,88
female,group B,master's degree,standard,none,90,95,93
male,group A,associate's degree,free/reduced,none,47,57,44
male,group C,some college,standard,none,76,78,75
female,group B,some college,standard,completed,71,83,78
female,group B,some college,free/reduced,none,49,60,54
female,group B,master's degree,standard,completed,100,96,86
female,group B,master's degree,free/reduced,completed,94,100,90
female,group B,high school,free/reduced,none,64,63,67
female,group B,high school,free/reduced,none,44,54,53
female,group B,bachelor's degree,standard,none,78,72,70
```

Student Performance Prediction

In this problem, we are trying to predict the scores of students in these exams given information regarding their demographics and other factors such as parental education, lunch, and test preparation.

This problem is a classification problem because the target variables can be seen as a pre-processed measure to bucket the students's grades in a manner which is easier to handle.

This problem is an interesting one because it is trying to model the up or comparison problem. As we have three exam scores, we can also use it to predict which student is better at which subject.

2. Data Acquisition

For the problem identified by you, students have to find the data source themselves from any data source.

2.1 Download the data directly

Since we're downloading data from kaggle, we can use the kaggle API to fetch the data. For that we will need to install the kaggle library and then authenticate our access using our API key. We can then write a predefined script which is used to download and unzip the data from our kaggle profile into the following directory.

```
!pip install -q kaggle
# Make a directory for the data
!mkdir ~/data/kaggle/student_exam
cd ~/data/kaggle/student_exam
# Download the dataset using the kaggle API
kaggle datasets download -d spscientist/students-performance-in-exams
# Unzip the downloaded dataset and delete the zip
unzip -q students-performance-in-exams.zip
rm students-performance-in-exams.zip
```

2.2 Code for converting the above downloaded data into a form suitable for DL

The data is in the form of a csv file and we can conveniently read the loading our data loading requirements model.

2.3 Write your observations from the above.

1. Size of the dataset
2. What type of data attributes are there?

Scale > 2 Nominal > 8 Ordinal > 0 binary

```
!wc -l ~/data/kaggle/student_exam/StudentsPerformance.csv
1001 /home/ubuntu/data/kaggle/student_exam/StudentsPerformance.csv
# Head command to see the first few lines of the dataset
!head ~/data/kaggle/student_exam/StudentsPerformance.csv
gender,race/ethnicity,parental level of education,lunch,test preparation course,math s
female,group B,bachelor's degree,standard,none,72,72,74
female,group C,some college,standard,completed,69,90,88
female,group B,master's degree,standard,none,90,95,93
male,group A,associate's degree,free/reduced,none,47,57,44
male,group C,some college,standard,none,76,78,75
female,group B,some college,standard,completed,71,83,78
female,group B,some college,free/reduced,none,49,60,54
female,group B,master's degree,standard,completed,100,96,86
female,group B,master's degree,free/reduced,completed,94,100,90
female,group B,high school,free/reduced,none,64,63,67
female,group B,high school,free/reduced,none,44,54,53
female,group B,bachelor's degree,standard,none,78,72,70
```

Student Performance Prediction

In this problem, we are trying to predict the scores of students in these exams given information regarding their demographics and other factors such as parental education, lunch, and test preparation.

This problem is a classification problem because the target variables can be seen as a pre-processed measure to bucket the students's grades in a manner which is easier to handle.

This problem is an interesting one because it is trying to model the up or comparison problem. As we have three exam scores, we can also use it to predict which student is better at which subject.

2. Data Acquisition

For the problem identified by you, students have to find the data source themselves from any data source.

2.1 Download the data directly

Since we're downloading data from kaggle, we can use the kaggle API to fetch the data. For that we will need to install the kaggle library and then authenticate our access using our API key. We can then write a predefined script which is used to download and unzip the data from our kaggle profile into the following directory.

```
!pip install -q kaggle
# Make a directory for the data
!mkdir ~/data/kaggle/student_exam
cd ~/data/kaggle/student_exam
# Download the dataset using the kaggle API
kaggle datasets download -d spscientist/students-performance-in-exams
# Unzip the downloaded dataset and delete the zip
unzip -q students-performance-in-exams.zip
rm students-performance-in-exams.zip
```

2.2 Code for converting the above downloaded data into a form suitable for DL

The data is in the form of a csv file and we can conveniently read the loading our data loading requirements model.

2.3 Write your observations from the above.

1. Size of the dataset
2. What type of data attributes are there?

Scale > 2 Nominal > 8 Ordinal > 0 binary

```
!wc -l ~/data/kaggle/student_exam/StudentsPerformance.csv
1001 /home/ubuntu/data/kaggle/student_exam/StudentsPerformance.csv
# Head command to see the first few lines of the dataset
!head ~/data/kaggle/student_exam/StudentsPerformance.csv
gender,race/ethnicity,parental level of education,lunch,test preparation course,math s
female,group B,bachelor's degree,standard,none,72,72,74
female,group C,some college,standard,completed,69,90,88
female,group B,master's degree,standard,none,90,95,93
male,group A,associate's degree,free/reduced,none,47,57,44
male,group C,some college,standard,none,76,78,75
female,group B,some college,standard,completed,71,83,78
female,group B,some college,free/reduced,none,49,60,54
female,group B,master's degree,standard,completed,100,96,86
female,group B,master's degree,free/reduced,completed,94,100,90
female,group B,high school,free/reduced,none,64,63,67
female,group B,high school,free/reduced,none,44,54,53
female,group B,bachelor's degree,standard,none,78,72,70
```

Student Performance Prediction

In this problem, we are trying to predict the scores of students in these exams given information regarding their demographics and other factors such as parental education, lunch, and test preparation.

This problem is a classification problem because the target variables can be seen as a pre-processed measure to bucket the students's grades in a manner which is easier to handle.

This problem is an interesting one because it is trying to model the up or comparison problem. As we have three exam scores, we can also use it to predict which student is better at which subject.

3. Data Preparation

Describe the data preprocessing that is required for the data that you have downloaded.

3.1 Apply techniques

1. Ordinal to Ordinal Mapping
2. Ordinal to Nominal Mapping
3. Nominal to Nominal Mapping

```
!pip install -q numpy pandas seaborn matplotlib
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import matplotlib as mpl
%matplotlib inline
# Read the data into a dataframe
df = pd.read_csv('StudentsPerformance.csv')
df.head()
Output of df.head():
   gender  race/ethnicity  parental level of education  lunch  test preparation course  math s
0  female         group B           bachelor's degree  standard                     none      72
1  female         group C                some college  standard                Completed      69
2  female         group B             master's degree  standard                     none      90
df.info()
Output of df.info():
<class 'DataFrame'>
RangeIndex: 1000 entries, 0 to 999
Data columns (total 8 columns):
 #   Column                       Non-Null Count  Dtype
---  ------                       --------------  -----
 0   gender                       1000 non-null   object
 1   race/ethnicity               1000 non-null   object
 2   parental level of education  1000 non-null   object
 3   lunch                        1000 non-null   object
 4   test preparation course      1000 non-null   object
 5   math score                   1000 non-null   int64
 6   reading score                1000 non-null   int64
 7   writing score                1000 non-null   int64
dtypes: int64(3), object(5)
memory usage: 62.5+ KB
df['math score'].hist(bins=20)
<Axes>
```

```
df['reading score'].hist(bins=20)
<Axes>
```

```
df['writing score'].hist(bins=20)
<Axes>
```

```
fig, ax = plt.subplots(1, 3, figsize = (16,5))
plt.suptitle('Math Score Distribution', fontsize=20)
ax[0].hist(df['math score'], color='#ff7f0e', alpha=0.7)
ax[0].set_ylabel('Frequency', fontsize=15)
ax[0].set_xlabel('Math Score', fontsize=15)
ax[1].hist(df['reading score'], color='#2ca02c', alpha=0.7)
ax[1].set_xlabel('Reading Score', fontsize=15)
ax[2].hist(df['writing score'], color='#d62728', alpha=0.7)
ax[2].set_xlabel('Writing Score', fontsize=15)
plt.tight_layout()
```

```
fig, ax = plt.subplots(1, 2, figsize = (16,5))
ax[0].hist(df[df['gender'] == 'male']['math score'], bins=20, alpha=0.7, label='Male')
ax[0].hist(df[df['gender'] == 'female']['math score'], bins=20, alpha=0.7, label='Female')
ax[0].set_ylabel('Frequency', fontsize=15)
ax[0].set_xlabel('Math Score', fontsize=15)
ax[0].legend
ax[1].boxplot([df[df['gender'] == 'male']['math score'], df[df['gender'] == 'female']['math sc
ax[1].set_ylabel('Math Score', fontsize=15)
ax[1].set_xlabel('Gender', fontsize=15)
ax[1].set_xticks([1, 2], ['Male', 'Female'])
plt.suptitle('Gender Distribution', fontsize=20)
plt.tight_layout()
```

```
fig, ax = plt.subplots(1, 3, figsize = (16,5))
ax[0].boxplot([df[df['race/ethnicity'] == 'group A']['math score'], df[df['race/ethnicity'] == '
ax[0].set_ylabel('Math Score', fontsize=15)
ax[0].set_xlabel('Race/Ethnicity', fontsize=15)
ax[0].set_xticks([1, 2, 3, 4, 5], ['group A', 'group B', 'group C', 'group D', 'group E'])
ax[1].boxplot([df[df['race/ethnicity'] == 'group A']['reading score'], df[df['race/ethnicity'] ==
ax[1].set_xlabel('Race/Ethnicity', fontsize=15)
ax[1].set_xticks([1, 2, 3, 4, 5], ['group A', 'group B', 'group C', 'group D', 'group E'])
ax[2].boxplot([df[df['race/ethnicity'] == 'group A']['writing score'], df[df['race/ethnicity'] ==
ax[2].set_xlabel('Race/Ethnicity', fontsize=15)
ax[2].set_xticks([1, 2, 3, 4, 5], ['group A', 'group B', 'group C', 'group D', 'group E'])
plt.suptitle('Race/Ethnicity Distribution', fontsize=20)
plt.tight_layout()
```


- Gender: Lunch Distribution and Preparation Course: we may consider with a very slight imbalance but overall, it looks balanced.
- Ordinal to categorical: is a non-ordinal scale with a relatively small number of possible having descriptive labels.
- Nominal to Nominal: distribution with five levels (Nominal).

It seems that the scores are near normal distribution with a slight left skew.

```
df.isnull().sum()
gender                         0
race/ethnicity                 0
parental level of education    0
lunch                          0
test preparation course        0
math score                     0
reading score                  0
writing score                  0
dtype: int64
```

Since there are no null values we are missing, we can start to begin encoding or substitute for any nominal factors any factors in the dataset.

```
df.select_dtypes('object')
4
```

Since of the nominal cases are for categorical in nature and can be simply encoded to a 1/0 function.

- Race is a nominal attribute which could be one-hot encoded.
- Parental is categorical in nature and can be one-hot encoded.

```
for col, level in [['gender', 'male'], ['lunch', 'standard'], ['test preparation course', 'c
    df[col] = (df[col] == level).astype(int)
df.rename(columns={'gender': 'is_male', 'lunch': 'is_standard', 'test preparation course': 'is_
df.head()
```

	is_male	race/ethnicity	parental level of education	is_standard	is_completed	math score	reading score	writing score
0	0	group B	bachelor's degree	1	0	72	72	74
1	0	group C	some college	1	1	69	90	88
2	0	group B	master's degree	1	0	90	95	93
3	1	group A	associate's degree	0	0	47	57	44
4	1	group C	some college	1	0	76	78	75

```
df['parental level of education'].value_counts()
parental level of education
some college          226
associate's degree    222
high school           196
bachelor's degree     118
master's degree       59
Name: count, dtype: int64
# Ordinal encoding
ordinal = {'some high school': 0, 'high school': 1, 'some college': 2, 'associate's degree': 3,
df['parental_ordinal'] = df['parental level of education'].map(ordinal)
# Check the mapping
df[['parental level of education', 'parental_ordinal']].drop_duplicates()
```

	parental level of education	parental_ordinal
0	bachelor's degree	5
1	some college	2
2	master's degree	6
3	associate's degree	3
4	high school	1

```
# Nominal encoding
df = pd.get_dummies(df, columns=['race/ethnicity'], dtype=int)
df.shape
(1000, 14)
```

3.3 Normalize the data

Find ranges and decide on the ranges that need to be normalized.

Since we have data the range of marks is from 0 to 100, we shall simply divide by 100 to bring the data 0-1 range.

```
df.describe()
Output of df.describe():
       is_male  is_standard  is_completed  parental_ordinal  math score  reading score  writing score
count  1000.00      1000.00       1000.00           1000.00     1000.00       1000.00       1000.00
mean      0.48         0.65          0.35              3.94       66.08         69.17         68.54
std       0.50         0.48          0.48              1.59       15.16         14.60         15.44
min       0.00         0.00          0.00              1.00        0.00         17.00         10.00
25%       0.00         0.00          0.00              3.00       57.00         59.00         57.75
50%       0.00         1.00          0.00              5.00       66.00         70.00         69.00
75%       1.00         1.00          1.00              6.00       77.00         79.00         79.00
max       1.00         1.00          1.00              6.00      100.00         100.00        100.00
# Apply normalization to the relevant columns
df[['math score', 'reading score', 'writing score']] = df[['math score', 'reading score', 'writing
# Verify the changes
df.describe()
```

	is_male	is_standard	is_completed	parental_ordinal	math score	reading score	writing score
count	1000.00	1000.00	1000.00	1000.00	1000.00	1000.00	1000.00
mean	0.48	0.65	0.35	3.94	0.66	0.69	0.68
std	0.50	0.48	0.48	1.59	0.15	0.14	0.15
min	0.00	0.00	0.00	1.00	0.00	0.17	0.10
25%	0.00	0.00	0.00	3.00	0.57	0.59	0.57
50%	0.00	1.00	0.00	5.00	0.66	0.70	0.69
75%	1.00	1.00	1.00	6.00	0.77	0.79	0.79
max	1.00	1.00	1.00	6.00	1.00	1.00	1.00

3.5 Identify the target variables

- Separate the one-hot encoded target variables (Nominal) in the form of (n,n) shapes, called.
- Direction: Encode the target variables in particular one-hot encoding (in the target) or any other as you if required.

```
target_cols = ['race/ethnicity_group A', 'race/ethnicity_group B', 'race/ethnicity_group C', 'ra
# Check shape of the target variables
df_encoded[target_cols].shape
(1000, 5)
# Check the first few rows of the target variables
df_encoded[target_cols].head()
```

	race/ethnicity_group A	race/ethnicity_group B	race/ethnicity_group C	race/ethnicity_group D	race/ethnicity_group E
0	0	1	0	0	0
1	0	0	1	0	0
2	0	1	0	0	0
3	1	0	0	0	0
4	0	0	1	0	0

```
# Create a copy of the dataset
df = df_encoded.copy()
df.shape
(1000, 14)
# Check data types to ensure all features are numeric
df.dtypes
is_male                        int64
is_standard                    int64
is_completed                   int64
parental_ordinal               int64
math score                     float
reading score                  float
writing score                  float
race/ethnicity_group A         int64
race/ethnicity_group B         int64
race/ethnicity_group C         int64
race/ethnicity_group D         int64
race/ethnicity_group E         int64
dtype: object
# Display first few rows to verify changes
df.head()
```

	is_male	is_standard	is_completed	parental_ordinal	math score	reading score	writing score	race/ethnicity_group A	race/ethnicity_group B	race/ethnicity_group C	race/ethnicity_group D	race/ethnicity_group E
0	0	1	0	5	0.72	0.72	0.74	0	1	0	0	0
1	0	1	1	2	0.69	0.90	0.88	0	0	1	0	0
2	0	1	0	6	0.90	0.95	0.93	0	1	0	0	0
3	1	0	0	3	0.47	0.57	0.44	1	0	0	0	0
4	1	1	0	2	0.76	0.78	0.75	0	0	1	0	0

3.6 Split the data into training set and testing set

```
from sklearn.model_selection import train_test_split
X = df[['is_male', 'is_standard', 'is_completed', 'parental_ordinal', 'race/ethnicity_group A',
        'race/ethnicity_group B', 'race/ethnicity_group C', 'race/ethnicity_group D', 'race/ethnicity_group E']]
y = df[['math score', 'reading score', 'writing score']]
# Check shapes
X.shape, y.shape
((1000, 9), (1000, 3))
# Split into 80% training and 20% testing
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
print(f"Training set: {X_train.shape[0]} samples")
print(f"Testing set: {X_test.shape[0]} samples")
Training set: 800 samples
Testing set: 200 samples
```

3.7 Report

Describe the method adopted and justify why the method was used

1. Ordinal to Ordinal Mapping: Preserves ordinal relationships
2. Ordinal to Nominal Mapping: Converts ordinal to nominal
3. Nominal to Nominal Mapping: Standardizes nominal categories
4. Binary to Binary: Simple binary encoding
5. Normalization: Scales numerical features to [0, 1]

Describe the data cleaning and feature engineering steps that you used for the target attribute.

Describe the data cleaning and feature engineering steps that you used for the target attribute.

Explain why you chose the specific evaluation metrics that you will use for this target attribute.

Suggest three different machine learning models that you think would be suitable for predicting this target attribute. Justify your choice of these models.

Describe the steps you would take to improve the performance of the model. Consider hyperparameter tuning, feature selection, or ensemble methods.

```
Dataset Shape:
(1000, 14)
Target Variables Shape:
(1000, 3)
Training Set Shape:
(800, 9)
Testing Set Shape:
(200, 9)
Target Variable Statistics:
       math score  reading score  writing score
count     1000.00        1000.00        1000.00
mean        66.08          69.17          68.54
std         15.16          14.60          15.44
min          0.00          17.00          10.00
25%         57.00          59.00          57.75
50%         66.00          70.00          69.00
75%         77.00          79.00          79.00
max        100.00         100.00         100.00
Correlation Matrix:
               math score  reading score  writing score
math score       1.000000       0.817580       0.802954
reading score    0.817580       1.000000       0.954598
writing score    0.802954       0.954598       1.000000
Top Correlations with Target:
math score    1.000000
reading score    0.817580
writing score    0.802954
dtype: float64
Preprocessing Complete!
Ready for model training!
```

4. Deep Neural Network Architecture

4.1 Design the architecture that you will be using to solve the prediction problem identified.

- Hold-done layers, specifying the number of units in each layer and the activation function used in the layer.

```
# Install TensorFlow if needed
!pip install -q tensorflow
# Import necessary libraries
import tensorflow as tf
from tensorflow.keras import layers, models, callbacks, regularizers
2025-11-11 12:06:06.800217: E external/local_xla/stream_executor/cuda/cuda_blas.cc:2981] Unable to
2025-11-11 12:06:08.197000: W tensorflow/compiler/tf2tensorrt/utils/py/__init__.py:29] Failed to f
```

```
[Epoch 1/20]
[Batch 0/1] [D loss: 0.7192] [G loss: 1.1790] [MSE: 0.3988 -ssd_loss: 0.
[Batch 1/1] [D loss: 0.6724] [G loss: 0.8724] [MSE: 0.2382 -ssd_loss:
[Epoch 2/20]
[Batch 0/1] [D loss: 0.5172] [G loss: 0.7192] [MSE: 0.1988 -ssd_loss: 0.
[Batch 1/1] [D loss: 0.5124] [G loss: 0.6724] [MSE: 0.2382 -ssd_loss:
[Epoch 3/20]
[Batch 0/1] [D loss: 0.4988] [G loss: 0.5172] [MSE: 0.1988 -ssd_loss: 0.
[Batch 1/1] [D loss: 0.4988] [G loss: 0.5124] [MSE: 0.2382 -ssd_loss:
[Epoch 4/20]
[Batch 0/1] [D loss: 0.4988] [G loss: 0.4988] [MSE: 0.1988 -ssd_loss: 0.
[Batch 1/1] [D loss: 0.4988] [G loss: 0.4988] [MSE: 0.2382 -ssd_loss:
[Epoch 5/20]
[Batch 0/1] [D loss: 0.4988] [G loss: 0.4988] [MSE: 0.1988 -ssd_loss: 0.
[Batch 1/1] [D loss: 0.4988] [G loss: 0.4988] [MSE: 0.2382 -ssd_loss:
[Epoch 6/20]
[Batch 0/1] [D loss: 0.4988] [G loss: 0.4988] [MSE: 0.1988 -ssd_loss: 0.
[Batch 1/1] [D loss: 0.4988] [G loss: 0.4988] [MSE: 0.2382 -ssd_loss:
[Epoch 7/20]
[Batch 0/1] [D loss: 0.4988] [G loss: 0.4988] [MSE: 0.1988 -ssd_loss: 0.
[Batch 1/1] [D loss: 0.4988] [G loss: 0.4988] [MSE: 0.2382 -ssd_loss:
[Epoch 8/20]
[Batch 0/1] [D loss: 0.4988] [G loss: 0.4988] [MSE: 0.1988 -ssd_loss: 0.
[Batch 1/1] [D loss: 0.4988] [G loss: 0.4988] [MSE: 0.2382 -ssd_loss:
[Epoch 9/20]
[Batch 0/1] [D loss: 0.4988] [G loss: 0.4988] [MSE: 0.1988 -ssd_loss: 0.
[Batch 1/1] [D loss: 0.4988] [G loss: 0.4988] [MSE: 0.2382 -ssd_loss:
[Epoch 10/20]
[Batch 0/1] [D loss: 0.4988] [G loss: 0.4988] [MSE: 0.1988 -ssd_loss: 0.
[Batch 1/1] [D loss: 0.4988] [G loss: 0.4988] [MSE: 0.2382 -ssd_loss:
[Epoch 11/20]
[Batch 0/1] [D loss: 0.4988] [G loss: 0.4988] [MSE: 0.1988 -ssd_loss: 0.
[Batch 1/1] [D loss: 0.4988] [G loss: 0.4988] [MSE: 0.2382 -ssd_loss:
[Epoch 12/20]
[Batch 0/1] [D loss: 0.4988] [G loss: 0.4988] [MSE: 0.1988 -ssd_loss: 0.
[Batch 1/1] [D loss: 0.4988] [G loss: 0.4988] [MSE: 0.2382 -ssd_loss:
[Epoch 13/20]
[Batch 0/1] [D loss: 0.4988] [G loss: 0.4988] [MSE: 0.1988 -ssd_loss: 0.
[Batch 1/1] [D loss: 0.4988] [G loss: 0.4988] [MSE: 0.2382 -ssd_loss:
[Epoch 14/20]
[Batch 0/1] [D loss: 0.4988] [G loss: 0.4988] [MSE: 0.1988 -ssd_loss: 0.
[Batch 1/1] [D loss: 0.4988] [G loss: 0.4988] [MSE: 0.2382 -ssd_loss:
[Epoch 15/20]
[Batch 0/1] [D loss: 0.4988] [G loss: 0.4988] [MSE: 0.1988 -ssd_loss: 0.
[Batch 1/1] [D loss: 0.4988] [G loss: 0.4988] [MSE: 0.2382 -ssd_loss:
[Epoch 16/20]
[Batch 0/1] [D loss: 0.4988] [G loss: 0.4988] [MSE: 0.1988 -ssd_loss: 0.
[Batch 1/1] [D loss: 0.4988] [G loss: 0.4988] [MSE: 0.2382 -ssd_loss:
[Epoch 17/20]
[Batch 0/1] [D loss: 0.4988] [G loss: 0.4988] [MSE: 0.1988 -ssd_loss: 0.
[Batch 1/1] [D loss: 0.4988] [G loss: 0.4988] [MSE: 0.2382 -ssd_loss:
[Epoch 18/20]
[Batch 0/1] [D loss: 0.4988] [G loss: 0.4988] [MSE: 0.1988 -ssd_loss: 0.
[Batch 1/1] [D loss: 0.4988] [G loss: 0.4988] [MSE: 0.2382 -ssd_loss:
[Epoch 19/20]
[Batch 0/1] [D loss: 0.4988] [G loss: 0.4988] [MSE: 0.1988 -ssd_loss: 0.
[Batch 1/1] [D loss: 0.4988] [G loss: 0.4988] [MSE: 0.2382 -ssd_loss:
[Epoch 20/20]
[Batch 0/1] [D loss: 0.4988] [G loss: 0.4988] [MSE: 0.1988 -ssd_loss: 0.
[Batch 1/1] [D loss: 0.4988] [G loss: 0.4988] [MSE: 0.2382 -ssd_loss:
```

2 - 0.0001, 0.0001, 0.0001, 0.0001

```
[Epoch 1/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 2/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 3/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 4/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 5/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 6/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 7/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 8/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 9/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 10/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 11/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 12/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 13/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 14/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 15/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 16/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 17/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 18/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 19/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 20/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
```

3 - 0.0001, 0.0001, 0.0001, 0.0001 (with hypergrad: BTF0200: SGD, LOSSGAMMA: 0.01, BETA:

```
[Epoch 1/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 2/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 3/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 4/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 5/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 6/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 7/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 8/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 9/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 10/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 11/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 12/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 13/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 14/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 15/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 16/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 17/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 18/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 19/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 20/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
```

4 - 0.0001, 0.0001, 0.0001, 0.0001 (with hypergrad: BTF0200: SGD, LOSSGAMMA: 0.01, BETA:

```
[Epoch 1/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 2/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 3/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 4/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 5/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 6/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 7/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 8/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 9/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 10/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 11/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 12/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 13/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 14/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 15/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 16/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 17/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 18/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 19/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 20/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
```

5 - 0.0001, 0.0001, 0.0001, 0.0001 (with hypergrad: BTF0200: SGD, LOSSGAMMA: 0.01, BETA:

```
[Epoch 1/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 2/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 3/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 4/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 5/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 6/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 7/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 8/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 9/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 10/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 11/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 12/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 13/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 14/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 15/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 16/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 17/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 18/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 19/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 20/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
```

6 - 0.0001, 0.0001, 0.0001, 0.0001 (with hypergrad: BTF0200: SGD, LOSSGAMMA: 0.01, BETA:

```
[Epoch 1/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 2/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 3/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 4/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 5/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 6/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 7/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 8/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 9/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 10/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 11/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 12/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 13/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 14/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 15/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 16/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 17/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 18/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 19/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 20/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
```

7 - 0.0001, 0.0001, 0.0001, 0.0001 (with hypergrad: BTF0200: SGD, LOSSGAMMA: 0.01, BETA:

```
[Epoch 1/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 2/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 3/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 4/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 5/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 6/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 7/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 8/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 9/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 10/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 11/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 12/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 13/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 14/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 15/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 16/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 17/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 18/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 19/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 20/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
```

8 - 0.0001, 0.0001, 0.0001, 0.0001 (with hypergrad: BTF0200: SGD, LOSSGAMMA: 0.01, BETA:

```
[Epoch 1/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 2/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 3/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 4/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 5/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 6/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 7/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 8/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 9/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 10/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 11/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 12/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 13/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 14/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 15/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 16/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 17/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 18/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 19/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 20/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
```

9 - 0.0001, 0.0001, 0.0001, 0.0001 (with hypergrad: BTF0200: SGD, LOSSGAMMA: 0.01, BETA:

```
[Epoch 1/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 2/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 3/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 4/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 5/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 6/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 7/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 8/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 9/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 10/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 11/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 12/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 13/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 14/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 15/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 16/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 17/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 18/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 19/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 20/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
```

10 - 0.0001, 0.0001, 0.0001, 0.0001 (with hypergrad: BTF0200: SGD, LOSSGAMMA: 0.01, BETA:

```
[Epoch 1/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 2/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 3/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 4/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 5/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 6/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 7/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 8/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 9/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 10/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 11/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 12/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 13/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 14/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 15/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 16/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 17/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 18/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 19/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 20/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
```

11 - 0.0001, 0.0001, 0.0001, 0.0001 (with hypergrad: BTF0200: SGD, LOSSGAMMA: 0.01, BETA:

```
[Epoch 1/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 2/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 3/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 4/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 5/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 6/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 7/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 8/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 9/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 10/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 11/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 12/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 13/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 14/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 15/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 16/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 17/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 18/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 19/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_loss: 0.
[Epoch 20/20]
[Batch 0/1] [D loss: 0.5124] [G loss: 0.5124] [MSE: 0.4988] -ssd_lo
```

```
[18/18] -lm -lpcre -l zlib                    0.4s Matching  lines: 0.13% 0MB/s 0.23GB/s -wc_lines
Start 19/20
[19/20] -wc -l -MPAC -E Test                  0.4s Matching  lines: 0.13% 0MB/s 0.23GB/s -wc_lines
Start 20/20
[20/20] -wc -l -MPAC -E -Ou1                  0.4s Matching  lines: 0.13% 0MB/s 0.23GB/s -wc_lines
100% tests passed, 0 tests failed out of 20
Total Test time (real) = 0.4 sec
[-O2 -Wall -Wpedantic -Wshadow -Wstrict-prototypes -Wmissing-prototypes -Wcast-qual -Wcast-align -Wpointer-arith -Wwrite-strings -Wformat-security -Wvla -Wundef -Wredundant-decls -Wnestedexterns -Wmissing-declarations -Wunreachable-code -Wconversion -Wsign-conversion -Wsign-compare -Wfloat-equal -Wswitch-default -Wswitch-enum -Wbad-function-cast -Wstrict-overflow=5 -Wformat=2 -Wformat-truncation=2 -Wformat-overflow=2 -Wstringop-truncation -Wstringop-overflow=4 -Wnull-dereference -Wdangling-else -Wdangling-pointer=2 -Wtrampolines -Wjump-misscalculation -Wlogical-op -Wtautological-compare -Wtrivial-auto-var-init -Wtrivial-value -Wtrivial-decl -Wexpansion-to-defined -Wunterminated-string-initialization -Wunsuffixed-float-constants -Wdate-time -Wc99-compatible -Wc11-compatible -Wc23-compatible -Wc2x-compatible -Wc90-compatible -Wc99-c11-compatible -Wc99-c23-compatible -Wc99-c2x-compatible -Wc11-c23-compatible -Wc11-c2x-compatible -Wc23-c2x-compatible -Wc90-c99-compatible -Wc90-c11-compatible -Wc90-c23-compatible -Wc90-c2x-compatible -Wc2x-extensions -Wc11-extensions -Wc23-extensions -Wc90-extensions -Wc99-extensions -Wc++-compatible -Wc++-extensions -Wc++-deprecated -Wc++-compat -Wc++-compat-pedantic -Wc++11-compatible -Wc++14-compatible -Wc++17-compatible -Wc++20-compatible -Wc++23-compatible -Wc++26-compatible -Wc++-23-compatible -Wc++-26-compatible -Wc++11-extensions -Wc++14-extensions -Wc++17-extensions -Wc++20-extensions -Wc++23-extensions -Wc++26-extensions -Wc++-23-extensions -Wc++-26-extensions -Wc++-11-extensions -Wc++-14-extensions -Wc++-17-extensions -Wc++-20
```

```
Epoch 30/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.0
[INFO] Saved model to models/model_1.keras
[INFO] Training model 2 with hyperparams: {'units': 128, 'dropout': 0.2, 'lr': 0.0
Epoch 1/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 1s 1s/step - loss: 0.3961 - MAE: 0.5400 - val_loss: 0.4
Epoch 2/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 36ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss:
Epoch 3/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 36ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss:
Epoch 4/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 36ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss:
Epoch 5/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 36ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss:
Epoch 6/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 36ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss:
Epoch 7/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 36ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss:
Epoch 8/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 36ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss:
Epoch 9/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 36ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss:
Epoch 10/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 36ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss:
Epoch 11/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 36ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss:
Epoch 12/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 36ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss:
Epoch 13/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 36ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss:
Epoch 14/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 36ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss:
Epoch 15/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 36ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss:
Epoch 16/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 36ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss:
Epoch 17/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 36ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss:
Epoch 18/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 36ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss:
Epoch 19/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 36ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss:
Epoch 20/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 36ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss:
Epoch 21/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 36ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss:
Epoch 22/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 36ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss:
Epoch 23/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 36ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss:
Epoch 24/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 36ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss:
Epoch 25/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 36ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss:
Epoch 26/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 36ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss:
Epoch 27/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 36ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss:
Epoch 28/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 36ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss:
Epoch 29/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 36ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss:
Epoch 30/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 36ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss:
Epoch 31/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 36ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss:
Epoch 32/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 36ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss:
Epoch 33/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 36ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss:
Epoch 34/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 36ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss:
Epoch 35/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 36ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss:
Epoch 36/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 36ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss:
Epoch 37/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 36ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss:
Epoch 38/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 36ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss:
Epoch 39/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 36ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss:
Epoch 40/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 36ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss:
Epoch 41/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 36ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss:
Epoch 42/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 36ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss:
Epoch 43/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 36ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss:
Epoch 44/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 36ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss:
Epoch 45/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 36ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss:
Epoch 46/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 36ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss:
Epoch 47/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 36ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss:
Epoch 48/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 36ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss:
Epoch 49/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 36ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss:
Epoch 50/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 36ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss:
[INFO] Saved model to models/model_2.keras
[INFO] Training model 3 with hyperparams: {'units': 256, 'dropout': 0.3, 'lr': 0.0
Epoch 1/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 1s 1s/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.0
Epoch 2/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 3/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 4/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 5/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 6/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 7/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 8/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 9/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 10/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 11/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 12/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 13/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 14/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 15/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 16/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 17/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 18/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 19/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 20/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 21/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 22/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 23/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 24/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 25/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 26/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 27/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 28/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 29/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 30/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
[INFO] Saved model to models/model_3.keras
[INFO] Training model 4 with hyperparams: {'units': 64, 'dropout': 0.1, 'lr': 0.01
Epoch 1/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 1s 1s/step - loss: 0.3961 - MAE: 0.5400 - val_loss: 0.4
Epoch 2/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss: 0.
Epoch 3/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss: 0.
Epoch 4/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss: 0.
Epoch 5/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss: 0.
Epoch 6/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss: 0.
Epoch 7/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss: 0.
Epoch 8/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss: 0.
Epoch 9/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss: 0.
Epoch 10/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss: 0.
Epoch 11/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss: 0.
Epoch 12/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss: 0.
Epoch 13/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss: 0.
Epoch 14/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss: 0.
Epoch 15/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss: 0.
Epoch 16/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss: 0.
Epoch 17/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss: 0.
Epoch 18/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss: 0.
Epoch 19/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss: 0.
Epoch 20/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss: 0.
Epoch 21/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss: 0.
Epoch 22/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss: 0.
Epoch 23/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss: 0.
Epoch 24/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss: 0.
Epoch 25/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss: 0.
Epoch 26/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss: 0.
Epoch 27/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss: 0.
Epoch 28/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss: 0.
Epoch 29/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss: 0.
Epoch 30/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss: 0.
Epoch 31/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss: 0.
Epoch 32/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss: 0.
Epoch 33/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss: 0.
Epoch 34/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss: 0.
Epoch 35/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss: 0.
Epoch 36/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss: 0.
Epoch 37/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss: 0.
Epoch 38/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss: 0.
Epoch 39/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss: 0.
Epoch 40/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss: 0.
Epoch 41/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss: 0.
Epoch 42/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss: 0.
Epoch 43/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss: 0.
Epoch 44/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss: 0.
Epoch 45/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss: 0.
Epoch 46/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss: 0.
Epoch 47/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss: 0.
Epoch 48/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss: 0.
Epoch 49/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss: 0.
Epoch 50/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3961 - MAE: 0.5400 - val_loss: 0.
[INFO] Saved model to models/model_4.keras
[INFO] Training model 5 with hyperparams: {'units': 128, 'dropout': 0.2, 'lr': 0.0
Epoch 1/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 1s 1s/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 2/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 3/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 4/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 5/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 6/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 7/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 8/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 9/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 10/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 11/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 12/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 13/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 14/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 15/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 16/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 17/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 18/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 19/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 20/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 21/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 22/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 23/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 24/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 25/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 26/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 27/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 28/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 29/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 30/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
[INFO] Saved model to models/model_5.keras
[INFO] Training model 6 with hyperparams: {'units': 256, 'dropout': 0.3, 'lr': 0.0
Epoch 1/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 1s 1s/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.0
Epoch 2/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 3/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 4/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 5/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 6/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 7/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 8/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 9/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 10/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 11/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 12/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 13/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 14/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 15/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 16/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 17/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 18/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 19/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 20/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 21/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 22/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 23/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 24/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 25/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 26/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 27/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 28/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 29/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
Epoch 30/50
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 9ms/step - loss: 0.3500 - MAE: 0.3700 - val_loss: 0.
[INFO] Saved model to models/model_6.keras

Total models: 6
Best model: models/model_1.keras (MAE: 0.0)
[INFO] Loaded model: models/model_1.keras
[INFO] Evaluation on test set:
WARNING:tensorflow:5 out of the last 13 calls triggered tf.function retracing. Tr
ace more than once to avoid performance issues. Use TensorFlow's tf.function(ret
ype=True) option to suppress this warning.
  1/Unknown ━━━━━━━━━━━━━━━━━━━━ 0s 0s/step
[INFO] Test Loss: 0.0000
[INFO] Test MAE: 0.0000
[INFO] Saved predictions to results/test_predictions.csv

[INFO] Generating visualizations from results/test_predictions.csv
[INFO] Loaded 200 predictions, 11 features
[INFO] Generating Feature Importance Analysis...
[INFO] Saved: results/feature_importance.csv
[INFO] Saved: results/feature_importance.png
[INFO] Saved: results/correlation_matrix.csv
[INFO] Saved: results/correlation_matrix.png
[INFO] Saved: results/pairplot.png
[INFO] Saved: results/residuals.csv
[INFO] Saved: results/residuals.png
[INFO] Saved: results/error_analysis.csv
[INFO] Saved: results/error_analysis.png
[INFO] Saved: results/prediction_ranges.csv
[INFO] Saved: results/prediction_ranges.png
[INFO] Saved: results/summary_report.txt

[INFO] Generating Training Curves from results/training_history.csv
[INFO] Saved: results/training_curves.csv
[INFO] Saved: results/training_curves.png

[INFO] Generating Model Comparison Analysis...
[INFO] Saved: results/model_comparison.csv
[INFO] Saved: results/model_comparison.xlsx
[INFO] Saved: results/model_comparison.tex
[INFO] Saved: results/model_comparison.md
[INFO] Saved: results/model_comparison.pdf
[INFO] Saved: results/model_comparison.pkl
[INFO] Saved: results/model_comparison.npz
[INFO] Saved: results/model_comparison.h5
[INFO] Saved: results/model_comparison.hdf5
[INFO] Saved: results/model_comparison.parquet
[INFO] Saved: results/model_comparison.feather
[INFO] Saved: results/model_comparison.arrow
[INFO] Saved: results/model_comparison.msgpack
[INFO] Saved: results/model_comparison.dat
[INFO] Saved: results/model_comparison.bin
[INFO] Saved: results/model_comparison.binary
[INFO] Saved: results/model_comparison.data
[INFO] Saved: results/model_comparison.raw
[INFO] Saved: results/model_comparison.out
[INFO] Saved: results/model_comparison.log
[INFO] Saved: results/model_comparison.txt
[INFO] Saved: results/model_comparison.dat.gz
[INFO] Saved: results/model_comparison.bin.gz
[INFO] Saved: results/model_comparison.binary.gz
[INFO] Saved: results/model_comparison.data.gz
[INFO] Saved: results/model_comparison.raw.gz
[INFO] Saved: results/model_comparison.out.gz
[INFO] Saved: results/model_comparison.log.gz
[INFO] Saved: results/model_comparison.txt.gz
[INFO] Saved: results/model_comparison.dat.bz2
[INFO] Saved: results/model_comparison.bin.bz2
[INFO] Saved: results/model_comparison.binary.bz2
[INFO] Saved: results/model_comparison.data.bz2
[INFO] Saved: results/model_comparison.raw.bz2
[INFO] Saved: results/model_comparison.out.bz2
[INFO] Saved: results/model_comparison.log.bz2
[INFO] Saved: results/model_comparison.txt.bz2
[INFO] Saved: results/model_comparison.dat.xz
[INFO] Saved: results/model_comparison.bin.x
```