

```

# ...
import numpy as np
import tensorflow as tf
from tensorflow import keras
import pandas as pd
import seaborn as sns
from pylab import rcParams
import matplotlib.pyplot as plt

sns.set(style='whitegrid', palette='muted', font_scale=1.5)
rcParams['figure.figsize'] = 16, 10 # Adjusted the figsize attribute
RANDOM_SEED = 42
np.random.seed(RANDOM_SEED)
tf.random.set_seed(RANDOM_SEED)

# data
data_time = np.arange(0, 200, 0.1)
sin_values = np.sin(data_time) + np.random.normal(scale=0.5, size=len(data_time))
plt.plot(data_time, sin_values, label='sine (with noise)')

# data pre-processing
data_full = pd.DataFrame(dict(sine=sin_values), index=data_time, columns=['sine'])
data_full.head()

len_train = int(len(data_full) * 0.8)
len_test = len(data_full) - len_train
train, test = data_full.iloc[0:len_train], data_full.iloc[len_train:] # Corrected this line

def gen_data(X, y, num_steps=1):
    Xs, ys = [], []
    for i in range(len(X) - num_steps):
        Xs.append(X.iloc[i:(i + num_steps)].values)
        ys.append(y.iloc[i + num_steps])
    return np.array(Xs), np.array(ys)

num_steps = 10
trainX, trainY = gen_data(train, train['sine'], num_steps)
testX, testY = gen_data(test, test['sine'], num_steps)

# implementing sequential model
lstm_model = keras.Sequential()
lstm_model.add(keras.layers.LSTM(128, input_shape=(trainX.shape[1], trainX.shape[2])))
lstm_model.add(keras.layers.Dense(1))
lstm_model.compile(loss='mean_squared_error', optimizer=keras.optimizers.Adam(0.001)) # Corrected the loss function name

# early stopping
callbacks = [tf.keras.callbacks.EarlyStopping(monitor='loss', patience=3)]

# model training
history = lstm_model.fit(
    trainX, trainY,
    epochs=30,
    batch_size=16,
    validation_split=0.1,
    shuffle=False,
    callbacks=callbacks, # Corrected this line
)

# evaluation and print the result
result = lstm_model.evaluate(testX, testY)
print("Test Loss:", result)

# Plotting
plt.plot(history.history['loss'], label='train')
plt.plot(history.history['val_loss'], label='test')
plt.legend()
plt.show()

```



```

Epoch 1/30
90/90 [=====] - 8s 32ms/step - loss: 0.3727 - val_loss: 0.36
Epoch 2/30
90/90 [=====] - 2s 23ms/step - loss: 0.3188 - val_loss: 0.35
Epoch 3/30
90/90 [=====] - 2s 21ms/step - loss: 0.3167 - val_loss: 0.35
Epoch 4/30
90/90 [=====] - 2s 26ms/step - loss: 0.3157 - val_loss: 0.35
Epoch 5/30
90/90 [=====] - 3s 32ms/step - loss: 0.3149 - val_loss: 0.35
Epoch 6/30
90/90 [=====] - 2s 24ms/step - loss: 0.3141 - val_loss: 0.34
Epoch 7/30
90/90 [=====] - 2s 22ms/step - loss: 0.3133 - val_loss: 0.34
Epoch 8/30
90/90 [=====] - 2s 20ms/step - loss: 0.3126 - val_loss: 0.34
Epoch 9/30
90/90 [=====] - 2s 24ms/step - loss: 0.3120 - val_loss: 0.34
Epoch 10/30
90/90 [=====] - 3s 31ms/step - loss: 0.3116 - val_loss: 0.34
Epoch 11/30
90/90 [=====] - 2s 23ms/step - loss: 0.3111 - val_loss: 0.34
Epoch 12/30
90/90 [=====] - 2s 20ms/step - loss: 0.3108 - val_loss: 0.34
Epoch 13/30
90/90 [=====] - 2s 21ms/step - loss: 0.3104 - val_loss: 0.34
Epoch 14/30
90/90 [=====] - 1s 11ms/step - loss: 0.3101 - val_loss: 0.34
Epoch 15/30
90/90 [=====] - 1s 11ms/step - loss: 0.3098 - val_loss: 0.34
Epoch 16/30
90/90 [=====] - 1s 11ms/step - loss: 0.3095 - val_loss: 0.34
Epoch 17/30
90/90 [=====] - 1s 11ms/step - loss: 0.3091 - val_loss: 0.34
Epoch 18/30
90/90 [=====] - 1s 11ms/step - loss: 0.3088 - val_loss: 0.34
Epoch 19/30
90/90 [=====] - 1s 15ms/step - loss: 0.3084 - val_loss: 0.34
Epoch 20/30
90/90 [=====] - 2s 17ms/step - loss: 0.3080 - val_loss: 0.34
Epoch 21/30
90/90 [=====] - 1s 12ms/step - loss: 0.3076 - val_loss: 0.34
Epoch 22/30
90/90 [=====] - 1s 11ms/step - loss: 0.3071 - val_loss: 0.34
Epoch 23/30
90/90 [=====] - 1s 11ms/step - loss: 0.3065 - val_loss: 0.34
Epoch 24/30
90/90 [=====] - 1s 11ms/step - loss: 0.3058 - val_loss: 0.34
Epoch 25/30
90/90 [=====] - 1s 12ms/step - loss: 0.3050 - val_loss: 0.34
Epoch 26/30
90/90 [=====] - 1s 11ms/step - loss: 0.3043 - val_loss: 0.34
Epoch 27/30
90/90 [=====] - 1s 11ms/step - loss: 0.3035 - val_loss: 0.34
Epoch 28/30
90/90 [=====] - 1s 12ms/step - loss: 0.3029 - val_loss: 0.34
Epoch 29/30
90/90 [=====] - 1s 11ms/step - loss: 0.3023 - val_loss: 0.34
Epoch 30/30
90/90 [=====] - 1s 11ms/step - loss: 0.3016 - val_loss: 0.34
13/13 [=====] - 0s 7ms/step - loss: 0.3038
Test Loss: 0.303804486989975

```



