```
import numpy as np
import pandas as pd
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, Dropout
from tensorflow.keras.callbacks import EarlyStopping
import matplotlib.pyplot as plt

# Simulated Dataset (Replace with real data)
def generate_synthetic_data(num_samples=10000):
    np.random.seed(42)
    product_features = np.random.rand(num_samples, 5)  # Features like rating, stock, reviews, etc.
    customer_behavior = np.random.rand(num_samples, 3)  # Clicks, views, etc.
    demand = (product_features[:, 0] * 0.4 + customer_behavior[:, 0] * 0.6 +
              np.random.normal(0, 0.05, num_samples))
    price = 100 - demand * 50  # Dynamic pricing logic (inverse relation to demand)
    price = np.clip(price, 10, 100)  # Ensuring prices are within a range
    features = np.hstack((product_features, customer_behavior))
    return features, price

# Generate Data
features, prices = generate_synthetic_data()
X_train, X_test, y_train, y_test = train_test_split(features, prices, test_size=0.2, random_state=42)

# Standardize the features
scaler = StandardScaler()
X_train = scaler.fit_transform(X_train)
X_test = scaler.transform(X_test)

# Build the Deep Learning Model
model = Sequential([
    Dense(64, input_dim=X_train.shape[1], activation='relu'),
    Dropout(0.2),
    Dense(32, activation='relu'),
    Dropout(0.2),
    Dense(1, activation='linear')  # Output is the predicted price
])

model.compile(optimizer='adam', loss='mse', metrics=['mae'])

# Train the Model
early_stopping = EarlyStopping(monitor='val_loss', patience=10, restore_best_weights=True)
history = model.fit(X_train, y_train, validation_data=(X_test, y_test),
                    epochs=100, batch_size=64, callbacks=[early_stopping], verbose=1)

# Evaluate the Model
loss, mae = model.evaluate(X_test, y_test, verbose=0)
print(f"Test Mean Absolute Error: {mae:.2f}")

# Predict prices for test set
predicted_prices = model.predict(X_test)

# Visualize the Results
plt.figure(figsize=(10, 5))
plt.scatter(y_test, predicted_prices, alpha=0.6)
plt.xlabel("Actual Prices")
plt.ylabel("Predicted Prices")
plt.title("Actual vs Predicted Prices")
plt.show()

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

```
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87: UserWarning: Do not pass an `input_shape`/`input_dim` argu
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
Epoch 1/100
125/125 ━━━━━━━━━━━━━━━━━━━━ 2s 4ms/step - loss: 5355.0474 - mae: 72.3169 - val_loss: 2993.1592 - val_mae: 53.7499
Epoch 2/100
125/125 ━━━━━━━━━━━━━━━━━━━━ 0s 3ms/step - loss: 1849.8729 - mae: 39.7366 - val_loss: 110.9115 - val_mae: 8.4763
Epoch 3/100
125/125 ━━━━━━━━━━━━━━━━━━━━ 0s 3ms/step - loss: 236.6779 - mae: 12.3956 - val_loss: 91.2584 - val_mae: 7.6427
Epoch 4/100
125/125 ━━━━━━━━━━━━━━━━━━━━ 1s 6ms/step - loss: 230.0129 - mae: 12.2514 - val_loss: 81.4526 - val_mae: 7.2054
Epoch 5/100
125/125 ━━━━━━━━━━━━━━━━━━━━ 1s 8ms/step - loss: 223.9629 - mae: 11.9895 - val_loss: 73.9429 - val_mae: 6.8802
Epoch 6/100
125/125 ━━━━━━━━━━━━━━━━━━━━ 1s 8ms/step - loss: 209.7952 - mae: 11.6185 - val_loss: 67.1072 - val_mae: 6.5435
Epoch 7/100
125/125 ━━━━━━━━━━━━━━━━━━━━ 1s 7ms/step - loss: 203.1453 - mae: 11.3604 - val_loss: 61.5082 - val_mae: 6.2695
Epoch 8/100
125/125 ━━━━━━━━━━━━━━━━━━━━ 1s 4ms/step - loss: 199.8428 - mae: 11.3192 - val_loss: 55.3147 - val_mae: 5.9372
Epoch 9/100
125/125 ━━━━━━━━━━━━━━━━━━━━ 1s 5ms/step - loss: 191.6376 - mae: 11.0749 - val_loss: 52.1763 - val_mae: 5.7714
Epoch 10/100
125/125 ━━━━━━━━━━━━━━━━━━━━ 1s 6ms/step - loss: 191.0204 - mae: 11.0079 - val_loss: 47.9139 - val_mae: 5.5600
Epoch 11/100
125/125 ━━━━━━━━━━━━━━━━━━━━ 1s 5ms/step - loss: 182.8524 - mae: 10.7115 - val_loss: 43.8638 - val_mae: 5.3273
Epoch 12/100
125/125 ━━━━━━━━━━━━━━━━━━━━ 1s 4ms/step - loss: 174.1781 - mae: 10.5336 - val_loss: 40.4944 - val_mae: 5.1262
Epoch 13/100
125/125 ━━━━━━━━━━━━━━━━━━━━ 1s 5ms/step - loss: 168.3750 - mae: 10.4081 - val_loss: 37.9945 - val_mae: 4.9826
Epoch 14/100
125/125 ━━━━━━━━━━━━━━━━━━━━ 1s 6ms/step - loss: 170.2426 - mae: 10.4265 - val_loss: 31.3046 - val_mae: 4.5138
Epoch 15/100
125/125 ━━━━━━━━━━━━━━━━━━━━ 1s 5ms/step - loss: 163.0575 - mae: 10.1171 - val_loss: 27.4216 - val_mae: 4.2085
Epoch 16/100
125/125 ━━━━━━━━━━━━━━━━━━━━ 1s 5ms/step - loss: 156.8917 - mae: 9.9747 - val_loss: 24.7353 - val_mae: 3.9996
Epoch 17/100
125/125 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - loss: 150.9225 - mae: 9.7335 - val_loss: 22.4661 - val_mae: 3.8247
Epoch 18/100
125/125 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - loss: 150.2227 - mae: 9.7006 - val_loss: 20.7576 - val_mae: 3.6822
Epoch 19/100
125/125 ━━━━━━━━━━━━━━━━━━━━ 0s 3ms/step - loss: 149.0192 - mae: 9.6358 - val_loss: 17.0699 - val_mae: 3.3255
Epoch 20/100
125/125 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - loss: 147.9985 - mae: 9.6605 - val_loss: 17.1224 - val_mae: 3.3467
Epoch 21/100
125/125 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - loss: 146.2356 - mae: 9.5507 - val_loss: 15.6900 - val_mae: 3.2094
Epoch 22/100
125/125 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 133.2541 - mae: 9.1681 - val_loss: 14.4147 - val_mae: 3.0688
Epoch 23/100
125/125 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 141.3667 - mae: 9.4432 - val_loss: 14.0082 - val_mae: 3.0458
Epoch 24/100
125/125 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 138.1913 - mae: 9.2672 - val_loss: 15.6741 - val_mae: 3.2371
Epoch 25/100
125/125 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 143.5442 - mae: 9.4724 - val_loss: 12.8570 - val_mae: 2.9125
Epoch 26/100
125/125 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - loss: 135.5305 - mae: 9.2085 - val_loss: 18.1208 - val_mae: 3.5515
Epoch 27/100
125/125 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 144.6989 - mae: 9.5604 - val_loss: 10.3739 - val_mae: 2.5862
Epoch 28/100
125/125 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 138.8586 - mae: 9.3330 - val_loss: 9.5850 - val_mae: 2.4823
Epoch 29/100
125/125 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 135.6491 - mae: 9.1854 - val_loss: 12.1328 - val_mae: 2.8444
Epoch 30/100
125/125 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 134.7421 - mae: 9.1745 - val_loss: 10.0194 - val_mae: 2.5520
Epoch 31/100
125/125 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 130.6765 - mae: 9.0394 - val_loss: 10.3162 - val_mae: 2.6033
Epoch 32/100
125/125 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 127.2939 - mae: 8.9150 - val_loss: 7.9928 - val_mae: 2.2493
Epoch 33/100
125/125 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 129.8516 - mae: 9.0715 - val_loss: 11.8475 - val_mae: 2.8156
Epoch 34/100
125/125 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 132.6461 - mae: 9.1513 - val_loss: 7.3296 - val_mae: 2.1472
Epoch 35/100
125/125 ━━━━━━━━━━━━━━━━━━━━ 0s 3ms/step - loss: 129.4657 - mae: 9.0107 - val_loss: 10.9785 - val_mae: 2.6901
Epoch 36/100
125/125 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 129.6733 - mae: 9.0514 - val_loss: 12.1907 - val_mae: 2.8756
Epoch 37/100
125/125 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 126.4600 - mae: 8.8883 - val_loss: 8.0565 - val_mae: 2.2627
Epoch 38/100
125/125 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 123.0905 - mae: 8.8109 - val_loss: 8.4759 - val_mae: 2.3363
Epoch 39/100
125/125 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 132.2499 - mae: 9.0618 - val_loss: 9.3007 - val_mae: 2.4544
Epoch 40/100
125/125 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 126.4090 - mae: 8.8552 - val_loss: 10.0070 - val_mae: 2.5513
Epoch 41/100
125/125 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - loss: 128.9491 - mae: 8.9451 - val_loss: 9.0551 - val_mae: 2.4288
```