**Lesson 8: Linear Regression Model – Part 2**

**🔹 Introduction**

Welcome back! In Part 1, we introduced linear regression and its core concepts. Now, it’s time to get hands-on.

In this lesson, you will:

* Learn how to train a linear regression model
* Understand how to evaluate it
* See an example in Python using scikit-learn

**🔹 Steps to Build a Linear Regression Model**

* 1. Prepare the DataEnsure your dataset has input features (X) and a target variable (y).
  2. Split the DatasetTypically into training and testing sets (e.g., 80/20 split).
  3. Train the ModelFit the linear regression model to the training data.
  4. Make PredictionsUse the trained model to predict the target variable on test data.
  5. Evaluate PerformanceUse metrics like MSE (Mean Squared Error) and R² (coefficient of determination).

**🔹 Python Implementation**

import numpy as np

import pandas as pd

from sklearn.linear\_model import LinearRegression

from sklearn.model\_selection import train\_test\_split

from sklearn.metrics import mean\_squared\_error, r2\_score

import matplotlib.pyplot as plt

# Sample data

data = {

'Area (sq ft)': [1000, 1500, 2000, 2500, 3000],

'Price ($)': [200000, 250000, 300000, 350000, 400000]

}

df = pd.DataFrame(data)

# Features and target

X = df[['Area (sq ft)']]

y = df['Price ($)']

# Train-test split

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.2, random\_state=42)

# Model

model = LinearRegression()

model.fit(X\_train, y\_train)

# Predictions

y\_pred = model.predict(X\_test)

# Evaluation

print("Mean Squared Error (MSE):", mean\_squared\_error(y\_test, y\_pred))

print("R² Score:", r2\_score(y\_test, y\_pred))

# Visualization

plt.scatter(X, y, color='blue', label='Actual data')

plt.plot(X, model.predict(X), color='red', label='Regression Line')

plt.xlabel("Area (sq ft)")

plt.ylabel("Price ($)")

plt.legend()

plt.title("Linear Regression Example")

plt.show()

**🔹 Evaluation Metrics Explained**

|  |  |
| --- | --- |
| **Metric** | **What It Means** |
| MSE | Measures the average squared error between predicted and actual values |
| R² Score | Indicates how well the model explains the variability of the target variable  R² = 1 means perfect fit, R² = 0 means no predictive power |

**🧠 Mini Exercise (Optional)**

Try this yourself:  
Modify the dataset to include more houses, or try predicting salaries based on years of experience using a dataset from Kaggle.

**🔹 Outro**

Awesome work! In this lesson, you learned:

✅ How to train and test a linear regression model  
✅ How to evaluate model performance  
✅ How to use Python and scikit-learn to bring theory into practice

Linear regression is your launchpad for more advanced models like polynomial regression, logistic regression, and beyond.