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 Data
   Ensure your dataset has input features (X) and a target variable (y).
2. Split the Dataset
   Typically into training and testing sets (e.g., 80/20 split).
3. Train the Model
   Fit the linear regression model to the training data.
4. Make Predictions
   Use the trained model to predict the target variable on test data.
5. Evaluate Performance
   Use metrics like MSE (Mean Squared Error) and R^2 (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
R^2 Score	Indicates how well the model explains the variability of the target variable $R^2 = 1$ means perfect fit, $R^2 = 0$ 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.