Regression Analysis

Introduction

- Regression analysis is a statistical method used to model the relationship between a **dependent variable (Y)** and one or more **independent variables (X)**.

Objectives

- Understand and quantify relationships between variables.
- Predict the value of the dependent variable based on independent variables.
- Evaluate the impact of independent variables on outcomes.

Applications

- Predicting sales based on advertising expenditure.
- Estimating housing prices using location, size, and features.
- Assessing risk factors in healthcare outcomes.

Types of Regression

1. Linear Regression:

- **Objective**: Models the relationship between a dependent variable (Y) and a single independent variable (X).
- **Equation**:

$$Y = \beta_0 + \beta_1 X + \epsilon$$

  - Y Dependent variable (what we want to predict).
  - X Independent variable (the predictor or feature).
  - β_0 Intercept (the value of Y when $X = 0$).
  - β_1 Slope (the change in Y for a unit change in X).
  - ϵ Error term (captures unexplained variance or noise).
- **Example**: Predicting house prices based on size.

Implementation of Linear Regression in Python

1. Importing Necessary Libraries

```
import numpy as np
import matplotlib.pyplot as plt
from sklearn.linear_model import LinearRegression
```

2. Data Preparation

```
X = np.array([[1], [2], [3], [4], [5]])  # Independent variable
Y = np.array([1, 2, 2.5, 4, 5])  # Dependent variable
```

3. Model Initialization and Fitting:

```
model = LinearRegression()
model.fit(X, Y)
```

4. Predicting on custom input

```
Y_pred = model.predict(X)
```

5. Visualizing the results

```
plt.scatter(X, Y, color='blue')  # Actual data points
plt.plot(X, Y_pred, color='red')  # Regression line
plt.title("Linear Regression")
plt.xlabel("X")
plt.ylabel("Y")
plt.show()
```

Model Parameters

```
print("Intercept (β0):", model.intercept_)
print("Coefficient (β1):", model.coef_)
```