Experiment No. 2

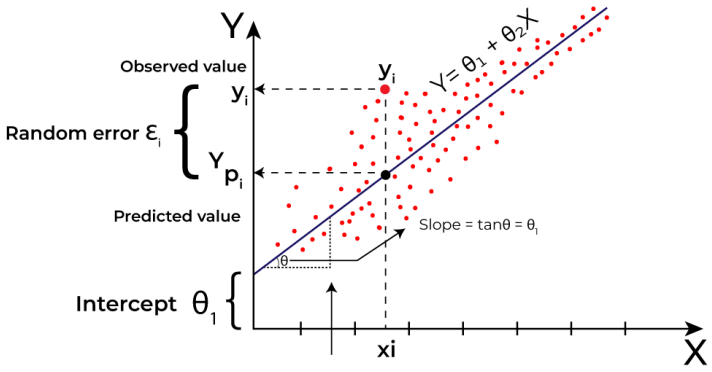
Aim: To implement and evaluate Linear Regression for predictive modeling using Python, and analyze the relationship between independent and dependent variables

Software tools: Google Colab, Python Libraries(Pandas,Scikit-learn,Matplotlib,Seaborn)

Theory:

1) Linear regression

Linear regression is a supervised machine learning algorithm used for modeling the relationship between a dependent variable and one or more independent variables by fitting a linear equation to the observed data. It is one of the simplest and most widely used statistical techniques for predictive modeling and data analysis.



Terminologies Related to the Regression Analysis:

● Dependent Variable: The main factor in Regression analysis which we want to predict or understand is called the dependent variable. It is also called target variable.
● Independent Variable: The factors which affect the dependent variables or which are used to predict the values of the dependent variables are called independent variables, also called as a predictor.
● Outliers: Outlier is an observation which contains either very low value or very high value in comparison to other observed values. An outlier may hamper the result, so it should be avoided.
● Multicollinearity: If the independent variables are highly correlated with each other than other variables, then such a condition is called Multicollinearity. It should not be

present in the dataset, because it creates a problem while ranking the most affecting variable.
● Underfitting and Overfitting: If our algorithm works well with the training dataset but not well with the test dataset, then such a problem is called Overfitting. And if our algorithm does not perform well even with a training dataset, then such a problem is called underfitting.

Linear Regression is a supervised machine learning technique used for predicting a dependent variable (Y) based on one or more independent variables (X). It assumes a linear relationship between the variables, represented by the equation: **Y=mX+c**

Where:

- **Y** = Dependent variable (Target)
- **X** = Independent variable (Feature)
- **m** = Slope or Coefficient (how much Y changes when X changes by one unit)
- **c** = Intercept (value of Y when X = 0)

In Python, we can implement Linear Regression using **Pandas** for data handling and **scikit-learn** for the regression model.

Scikit-learn :

Scikit-learn is a comprehensive machine learning library for Python. It provides a wide range of machine learning algorithms, including linear regression. Scikit-learn is easy to use and offers tools for data preprocessing, model selection, training and evaluation. The LinearRegression class in scikit-learn allows you to create and train linear regression models efficiently.

Matplotlib.pyplot:

Matplotlib is a popular data visualization library for creating static, animated, and interactive plots. matplotlib.pyplot is a module within Matplotlib that provides a MATLAB-like interface for creating plots and charts. It's often used to visualize data, including scatter plots, line charts, and regression lines, which can help in interpreting the results of linear regression.

It allows customization of figure size, axis labels, titles, legends, and gridlines.Commonly used function: plt.plot() and plt.scatter().

Seaborn:

Seaborn is a data visualization library based on Matplotlib that provides a higher-level interface for creating informative and attractive statistical graphics. It's commonly used for creating visually appealing plots for data exploration and presentation, including regression plots.

Functions like sns.scatterplot() and sns.lineplot() simplify regression visualization.

2) Multiple Linear Regression

Multiple Linear Regression is a statistical method used in predictive modeling and statistical analysis. It extends the concept of simple linear regression, which models the relationship between a dependent variable (the target) and a single independent variable (predictor or feature), to multiple independent variables.

The general equation is: $\mathbf{Y=b0+b1X1+b2X2+\cdots+bnXn+\varepsilon}$

Where:

- **Y** = Dependent variable (Target)
- $\mathbf{X_1, X_2, \ldots, X}$□ = Independent variables (Features)
- $\mathbf{b_0}$ = Intercept
- $\mathbf{b_1, b_2, \ldots, b}$□ = Coefficients showing the contribution of each independent variable
- **ε** = Error term (difference between actual and predicted value)

Applications

- Predicting sales based on advertising spend.
- Estimating house prices from area and location.
- Forecasting demand based on past data.
- Analyzing the effect of study time on exam scores.

Conclusion :

We applied Linear Regression using Pandas for data preparation, Scikit-learn for model training, and Matplotlib with Seaborn for visualization.