

Experiment No. 2

Expt No. 2	Implementation of Multiple Linear Regression.
Date :	

Aim:

To implement and understand the working of Multiple Linear Regression, a fundamental predictive modeling technique in Machine Learning that models the relationship between one dependent variable and multiple independent variables.

System Requirements:

- Operating System: Windows 8 or above / Linux / macOS
- Memory (RAM): Minimum 4 GB
- Processor: Minimum 2.33 GHz (Dual Core or higher)

Software/Tools Required:

Jupyter Notebook / Anaconda Navigator / Google Colaboratory / Spyder
Python 3.x with the following libraries:

- NumPy
- Pandas
- Matplotlib
- Seaborn
- Scikit-learn

Purpose of the Experiment:

- To understand how multiple independent variables can be used to predict a target variable.

- To gain hands-on experience with implementing and interpreting Multiple Linear Regression using Python.
- To learn the significance of feature selection, model fitting, and evaluation in predictive modeling.

Expected Outcomes:

- Understand the concept and mathematical foundation of Multiple Linear Regression.
- Load, visualize, and prepare a dataset for regression analysis.
- Implement Multiple Linear Regression using scikit-learn.
- Interpret model coefficients, intercept, and performance metrics such as R^2 score.
- Predict target values based on multiple input features.

Theory:

Multiple Linear Regression (MLR) is a statistical method used to analyze the relationship between a dependent variable and two or more independent variables. It extends the concept of simple linear regression, which involves predicting a dependent variable based on a single independent variable, to cases where there are multiple predictors. It serves to predict the change in the dependent variable based on the difference in the independent variable; this could also be called a Multiple regression line.

The Multiple Linear Regression Equation

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \dots + \beta_n X_n + \varepsilon$$

Where:

Y is the dependent variable being predicted.

β_0 is the intercept or constant term.

$\beta_1, \beta_2, \dots, \beta_n$ are the coefficients associated with each independent variable X_1, X_2, \dots, X_n respectively, representing their impact on Y.

ε is the error term accounting for the difference between the predicted and actual values of Y.

This multiple linear regression equation signifies that the dependent variable Y is a linear combination of the independent variables X_1, X_2, \dots, X_n weighted by their respective coefficients $\beta_1, \beta_2, \dots, \beta_n$, along with an intercept term β_0 . The goal of multiple linear regression is to estimate the coefficients that minimize the overall difference between predicted and actual values of the dependent variable.

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \dots + \epsilon$$

Labels and their corresponding parts in the equation:

- Dependent Variable (Response Variable) → Y
- Independent Variables (Predictors) → X_1, X_2, \dots
- Y intercept → β_0
- Slope Coefficient → β_1, β_2
- Error Term → ϵ

MLR uses the method of least squares to minimize the error between predicted and actual values. Scikit-learn provides `LinearRegression()` to model this efficiently.

Dataset:

The following dataset is used for regression. It includes Area (sq. ft), Number of Bedrooms, Age of House (in years), and Price:

Area	Bedroom	Age	Price
2600	3	20	550000
3000	4	15	565000
3200	NaN	18	610000
3600	3	30	595000
4000	5	8	760000
4100	6	8	810000

Real-World Applications of Linear Regression

Multiple Linear Regression helps in making more accurate predictions when outcomes depend on multiple factors. Some real-world applications include:

1. Real Estate: Predict house prices based on multiple features such as area (sq. ft.), number of bedrooms, age of the property, and proximity to amenities.

2. Healthcare: Estimate a patient's Body Mass Index (BMI) or disease risk based on inputs like age, weight, height, and activity level.
3. Finance: Forecast stock prices or investment returns by considering market indicators such as inflation rate, interest rate, and GDP growth.
4. Education: Predict student performance or exam scores using study hours, attendance, assignment scores, and internal assessment marks.
5. Manufacturing: Predict product quality or machine output based on temperature, pressure, input material quality, and operation time.

Procedure to implement Multiple Linear Regression –

Step1: Import required libraries

Step 2: Create a dataset with missing values.

Step 3: Convert it into a DataFrame using pandas.

Step 4: Handle missing data using fillna().

Step 5: Plot the heatmap to view feature correlations.

Step 6: Plot the scatter plot for visualization.

Step 7: Fit a Multiple Linear Regression model using
`sklearn.linear_model.LinearRegression`

Step 8: Print the intercept and coefficients.

Step 9: Make prediction for a sample input.

Procedure with Sample Code:

Step 1: Import necessary libraries

These libraries help in data manipulation, visualization, and model building

```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.linear_model import LinearRegression
```

Step 2: Create a Dataset with missing Values

Dataset with Area, Bedrooms (one missing value), Age, and Price

```
data = {  
    'Area': [2600, 3000, 3200, 3600, 4000, 4100],  
    'Bedroom': [3, 4, NaN, 3, 5, 6],  
    'Age': [20, 15, 18, 30, 8, 8],  
    'Price': [550000, 565000, 610000, 595000, 760000, 810000]  
}
```

Step 3: Convert it into a DataFrame using pandas.

```
df = pd.DataFrame(data)
```

Step 3: Handle missing data using fillna().

```
print("Before filling NaN:\n", df)  
df.fillna(0, inplace=True) # Replacing missing values with 0  
print("\nAfter filling NaN with 0:\n", df)
```

Step 5: Plot the heatmap to view feature correlations.

#Heatmap: Feature Correlation

```
sns.heatmap(df.corr(), annot=True)  
plt.title("Feature Correlation Heatmap")  
plt.show()
```

Step 6: Plot the scatter plot for visualization.

#Scatter Plot: Area vs Price

```
sns.relplot(x='Area', y='Price', data=df)  
plt.title("Area vs Price")  
plt.show()  
# Independent variables (Area, Bedroom, Age)
```

```
X = df[['Area', 'Bedroom', 'Age']]
```

```
# Dependent variable (Price)
```

```
y = df['Price']
```

Step 7: Fit a Multiple Linear Regression model

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

Step 8: Print the intercept and coefficients.

```
# Model parameters
```

```
print("\nIntercept:", model.intercept_)  
print("Coefficients:", model.coef_)
```

Step 9: Make prediction for a sample input

```
# Predicting price for a house with Area=3000, Bedroom=3, Age=15)
```

```
predicted_price = model.predict([[3000, 3, 15]])  
print("\nPredicted Price for Area=3000, Bedroom=3, Age=15:", predicted_price[0])
```

Observation

I. Record the following in observation table:

1. Regression Equation: $y = (m_1 \times \text{Area}) + (m_2 \times \text{Bedroom}) + (m_3 \times \text{Age}) + b$
2. Coefficient for Area (m_1): _____
3. Coefficient for Bedroom (m_2): _____
4. Coefficient for Age (m_3): _____
5. Intercept (b): _____
6. Predicted Price for (3000, 3, 15) : _____
7. Screenshot of Heatmap
8. Screenshot of Scatter Plot (Area vs Price)
9. Screenshot of Regression Output (print result)

II. Perform the following tasks and record results by applying Multiple Linear Regression. Use the datasets given for each task.

Task 1: Salary Prediction Based on Experience and Education		
1	10	30
2	12	35
3	14	45
4	16	55
5	18	60

Answer the Following:

1. Write the multiple regression equation using Experience and Education Level.
2. Predict the salary for a person with 6 years of experience and 20 years of education.
3. Plot the 3D surface or scatter plot with fitted regression plane and interpret the slopes.

Task 2: House Price Prediction Based on Area, Bedrooms, and Age			
Area (sqft)	Bedrooms	Age (years)	Price (₹ in Lakhs)
1500	3	5	45
2000	4	10	65
1800	3	8	58
2500	4	12	75
2200	5	6	80

Answer the Following:

1. Derive the multiple regression equation to predict House Price.
2. Predict the price of a 2400 sqft house with 4 bedrooms and 7 years of age.
3. Visualize the regression surface or 2D projections (e.g., Area vs Price) and interpret results.

Task 3: Insurance Premium Estimation Based on Age, BMI, and Smoking Status			
Age	BMI (Body Mass Index)	Smoker (1 = Yes, 0 = No)	Premium (₹ in 1000s)
25	22.0	0	12
35	28.0	1	35
45	26.5	0	25
30	30.1	1	40
50	27.0	0	28

Answer the Following:

1. Build a multiple regression model to estimate insurance premium.
2. Predict premium for a 40-year-old with BMI 27.5 who is a smoker.
3. Explain on the influence of the smoking variable (binary) on the premium amount.

Conclusion:

The experiment provides students with practical experience in applying Multiple Linear Regression to analyze the effect of multiple variables on an outcome. Through hands-on experience, students learned to build predictive models, interpret coefficients, and visualize model outcomes using real-world data.

Viva Questions



Sr.No	Question	CO
1.	What is Multiple Linear Regression?	1ICPC406_2
2.	Differentiate between Simple and Multiple Linear Regression.	1ICPC406_2
3.	What is the equation of a Multiple Linear Regression model?	1ICPC406_2
4.	What do the coefficients in the regression model represent?	1ICPC406_2
5.	How is the intercept interpreted in regression?	1ICPC406_2
6.	What are independent and dependent variables in your experiment?	1ICPC406_2
7.	Which Python library is used to implement linear regression?	1ICPC406_2
8.	What are some assumptions of linear regression?	1ICPC406_2
9.	How do you handle missing values in your dataset?	1ICPC406_2
10.	What does R^2 (R-squared) indicate in regression?	1ICPC406_2

11.	Why do we reshape the input data using <code>.values</code> or <code>np.array()</code> ?	1ICPC406_2
12.	What is overfitting in regression models?	1ICPC406_2
13.	How can you visually evaluate a regression model?	1ICPC406_2
14.	Why is feature scaling not mandatory for Linear Regression?	1ICPC406_2
15.	Can Multiple Linear Regression work with categorical data? If yes, how?	1ICPC406_2

References –

a. Textbook –

- i. Machine Learning with Python – An approach to Applied ML – Abhishek Vijayvargiya, BPB Publications, 1st Edition 2018
- ii. Machine Learning, Tom Mitchell, McGraw Hill Education, 1st Edition 1997

b. Online references –

- i. <http://www.analyticsvidhya.com/blog/2021/10/multiple-linear-regression/>
- ii. <http://www.javatpoint.com/multiple-linear-regression-in-machine-learning>
- iii. <http://www.geeksforgeeks.org/ml-multiple-linear-regression-using-python/>