

PRACTICAL No. 05

AIM :

To implement multivariable (Multiple) Linear Regression using python to predict the dependent variable based on two or more independent variables.

INPUT :

- A dataset with :
 - Two or more independent variables ($x_1, x_2, x_3 \dots$)
 - one dependent variable (y)

Example :

- x_1 = Study hours
- x_2 = Sleep hours
- y = marks scored

OUTPUT :

- A trained regression model.

- predicted values of x based on Inputs.

- Accuracy scores & graphical representation results (optional)

THEORY :

Multivariable (multiple) Linear Regression is an extension of simple linear regression where we use two or more independent variables to predict a single dependent variable.

It models the relationship as :

$$Y = m_1x_1 + m_2x_2 + \dots + x_n m_n + c$$

where

- Y = dependent variable

- x_1, x_2, \dots, x_n = independent variables

- m_1, m_2, \dots, m_n = coefficients / slopes

- c = intercept

The goal is to find the best fitting value of m_1 to m_n & c that minimize the prediction error.

Example

predict (Y) marks based on :

x_1 = study Hours

x_2 = sleep Hours.

Given Data :

x_1	x_2	y
1	2	50
2	1	60

Formula :

$$r = m_1 x_1 + m_2 x_2 + c$$

Step (1) : set up eqⁿ

$$50 = m_1(1) + m_2(2) + c \quad \dots (1)$$

$$60 = m_1(2) + m_2(1) + c \quad \dots (2)$$

Step (2) : solve for m_1 , m_2 & c

Subtraction of eqⁿ (1) & (2)

$$(2m_1 + m_2 + c) - (m_1 + 2m_2 + c) = 60 - 50$$

$$2m_1 + m_2 - m_1 - 2m_2 = 10$$

$$m_1 - m_2 = 10 \quad \dots (3)$$

Solve for c :

from eqⁿ (1)

$$c = 50 - m_1 - 2m_2$$

$$c = 50 - (m_2 + 10) - 2m_2 \quad \dots \text{from (3)}$$

$$c = 40 - 3m_2$$

Substitute eqⁿ (4) in (2)

$$2(m_2 + 10) + m_2 + (40 - 3m_2) = 60$$

$$2m_2 + 20 + m_2 + 40 - 3m_2 = 60$$

$$60 = 60$$

Since this holds we have infinitely many solutions based on a choice of m_2 . one possible solution can be obtained by assuming a value (e.g. setting $m_2 = 5$) :

$$m_1 = 15, m_2 = 5, c = ?$$

$$c = 40 - 3m_2 \quad \text{--- from (4)}$$

$$c = 40 - 3 \times 5$$

$$c = 25$$

$$y = 15x_1 + 5x_2 + 25$$

ALGORITHM:

- Step (1) : Import required libraries
- Step (2) : Load the dataset
- Step (3) : select input (x) & output (y)
- Step (4) : split data into training & testing sets.
- Step (5) : create a linear Regression object
- Step (6) : fit the model to training data using $\text{fit}(x_{\text{train}}, y_{\text{train}})$
- Step (7) : predict output using $\text{predict}(x_{\text{test}})$
- Step (8) : Evaluate model using metrics like : R^2 score
: MSE
- Step (9) : Plot predicted vs actual values.

CONCLUSION :

Multivariable linear regression is a powerful technique to analyze & predict a target variable based on multiple factors. It improves accuracy over simple linear regression when the output depends on more than one feature. In this practical, we successfully built a regression model.

that can predict outcomes using multiple inputs

REFERENCES :

- ① scikit-learn linear Regression tutorial.
- ② GeeksforGeeks. multiple Linear Regression
- ③ kaggle - Datasets & ML Projects

FLOWCHART :

