Semester	T.E. Semester VI – Computer Engineering
Subject	QA
Subject Professor In-charge	Prof. Kavita Shirsat

Group Members	Deep Salunkhee 21102A0014 Omkar Patil 21102A0003 Pranav Redij 21102A0005 Sukant Thombare 21102A0037 Shantanu Lagvankar 21102A0043	
Grade and Subject Teacher's Signature		

Mini Project Title	Multiple Regression	
Resources / Apparatus Required	Hardware: Computer system	Software: Python
Description	1. **Theoretical Background:** • **Multiple Linear Regression:** The code implements multiple linear regression, a statistical technique used to model the relationship between multiple independent variables (X1, X2, ...) and a dependent variable (Y). • **Ordinary Least Squares (OLS):** The method used to estimate the parameters of the linear regression model is Ordinary Least Squares. OLS aims to minimize the sum of the squared differences between the observed and predicted values of the dependent variable. • **Assumptions of Linear Regression:** The validity of the regression results relies on several assumptions, including linearity, independence of errors, homoscedasticity, and normality of errors. 2. **Mathematical Formulation:** • **Model Equation:** The model equation for multiple linear regression is represented as: Y = b0 + b1*X1 + b2*X2 + ... + bn*Xn + ε where Y is the dependent variable, X1, X2, ..., Xn are the independent variables, b0 is the intercept term, b1, b2, ..., bn are the regression coefficients, and ε is the error term. • **Coefficients Estimation:** The coefficients (b1, b2, ..., bn) are estimated using the method of Ordinary Least Squares (OLS) to minimize the sum of squared errors between the observed and predicted values of the dependent variable.	

	• **Model Summary:** The **model.summary()** function provides a detailed summary of the regression results, including coefficients, standard errors, t-values, p-values, and various statistics such as R-squared and adjusted R-squared. 3. **Statistical Metrics:** • **R-squared (R^2):** R-squared is a measure of the proportion of variance in the dependent variable that is explained by the independent variables. It ranges from 0 to 1, where higher values indicate a better fit of the model to the data. • **Total Sum of Squares (SST):** SST measures the total variance in the dependent variable. • **Regression Sum of Squares (SSR):** SSR measures the variance explained by the regression model. • **Error Sum of Squares (SSE):** SSE measures the unexplained variance or residual variance. • **Mean Square Regression (MSR):** MSR is the average amount of variance explained by the regression model. • **Mean Square Error (MSE):** MSE is the average amount of unexplained variance or residual variance. • **Degrees of Freedom:** Degrees of freedom represent the number of independent pieces of information in the data used to estimate a statistic. In the context of regression, df_model represents the degrees of freedom for the model, and df_resid represents the degrees of freedom for the residuals.
Program	(code below)

```python
import pandas as pd
import statsmodels.api as sm

# Read data from Excel file
data = pd.read_excel("data.xlsx")

# Separate independent variables (X) and dependent variable (Y)
X = data[['X1', 'X2']]
Y = data['Y']

# Add constant term for intercept
X = sm.add_constant(X)

# Create and fit the regression model
model = sm.OLS(Y, X).fit()

# Print the model summary
print(model.summary())

# Calculate SST (Total Sum of Squares)
```

```
y_mean = Y.mean()
SST = ((Y - y_mean) ** 2).sum()

# Calculate SSR (Regression Sum of Squares)
SSR = ((model.predict(X) - y_mean) ** 2).sum()

# Calculate SSE (Error Sum of Squares)
SSE = ((Y - model.predict(X)) ** 2).sum()

# Calculate R-squared
R_squared = SSR / SST

# Calculate MSR (Mean Regression Sum of Squares)
MSR = SSR / model.df_model

# Calculate MSE (Mean Error Sum of Squares)
MSE = SSE / model.df_resid

# Print calculated values
print("SST:", SST)
print("SSR:", SSR)
print("SSE:", SSE)
print("R^2:", R_squared)
print("MSR:", MSR)
print("MSE:", MSE)

# Print model equation
print("Model Equation:")
print("Y = {:.2f} + {:.2f}*X1 + {:.2f}*X2".format(model.params[0],
model.params[1], model.params[2]))
```

Output

	A	B	C	D
1	Y	X1	X2	
2	32	160	5.5	
3	15	80	6	
4	30	112	9.5	
5	34	185	5	
6	35	152	8	
7	10	90	3	
8	39	170	9	
9	26	140	5	
10	11	115	0.5	
11	23	150	1.5	
12				
13				
14				

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                      Y   R-squared:                       0.988
Model:                            OLS   Adj. R-squared:                  0.984
Method:                 Least Squares   F-statistic:                     285.8
Date:                Sat, 23 Mar 2024   Prob (F-statistic):           1.95e-07
Time:                        17:36:50   Log-Likelihood:                -15.013
No. Observations:                  10   AIC:                             36.03
Df Residuals:                       7   BIC:                             36.93
Df Model:                           2
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const        -13.8246      1.795     -7.701      0.000     -18.069      -9.580
X1             0.2122      0.013     16.759      0.000       0.182       0.242
X2             1.9995      0.146     13.728      0.000       1.655       2.344
==============================================================================
Omnibus:                        0.567   Durbin-Watson:                   2.132
Prob(Omnibus):                  0.753   Jarque-Bera (JB):                0.550
Skew:                           0.240   Prob(JB):                        0.759
Kurtosis:                       1.956   Cond. No.                         610.
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly spec
SST: 974.5
SSR: 962.7104146993281
SSE: 11.789585300672167
R^2: 0.98790191349341
MSR: 481.35520734966406
MSE: 1.6842264715245954
Model Equation:
Y = -13.82 + 0.21*X1 + 2.00*X2
```