BHARATIYA VIDYA BHAVAN'S
SARDAR PATEL INSTITUTE OF TECHNOLOGY
Bhavan's Campus, Munshi Nagar, Andheri (West), Mumbai – 400058-India

DEPARTMENT OF COMPUTERE ENGINEERING

SUBJECT: Machine Learning

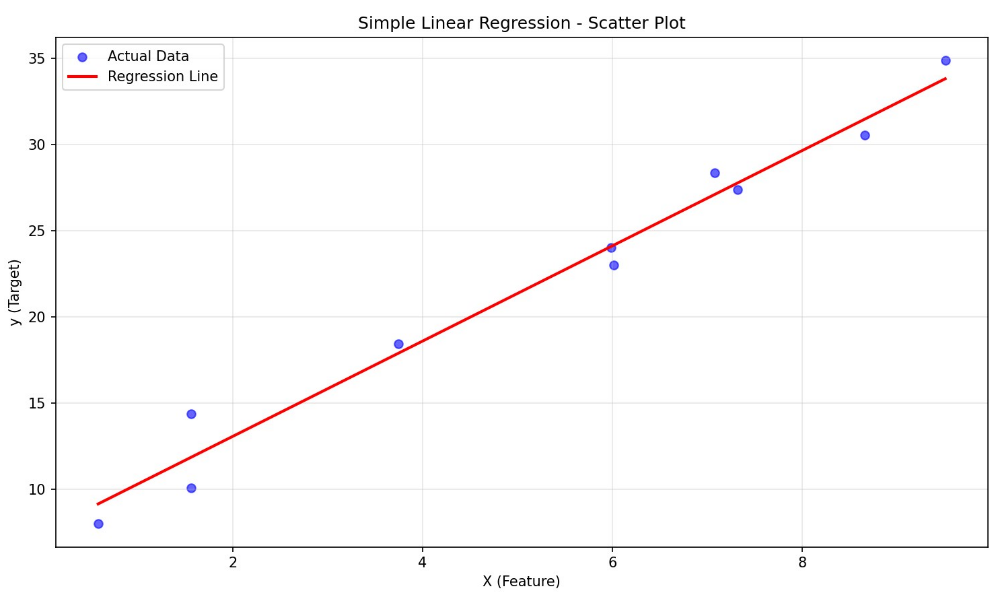
Name	**Raj Kalpesh Mathuria**
UID no.	**2023300139**

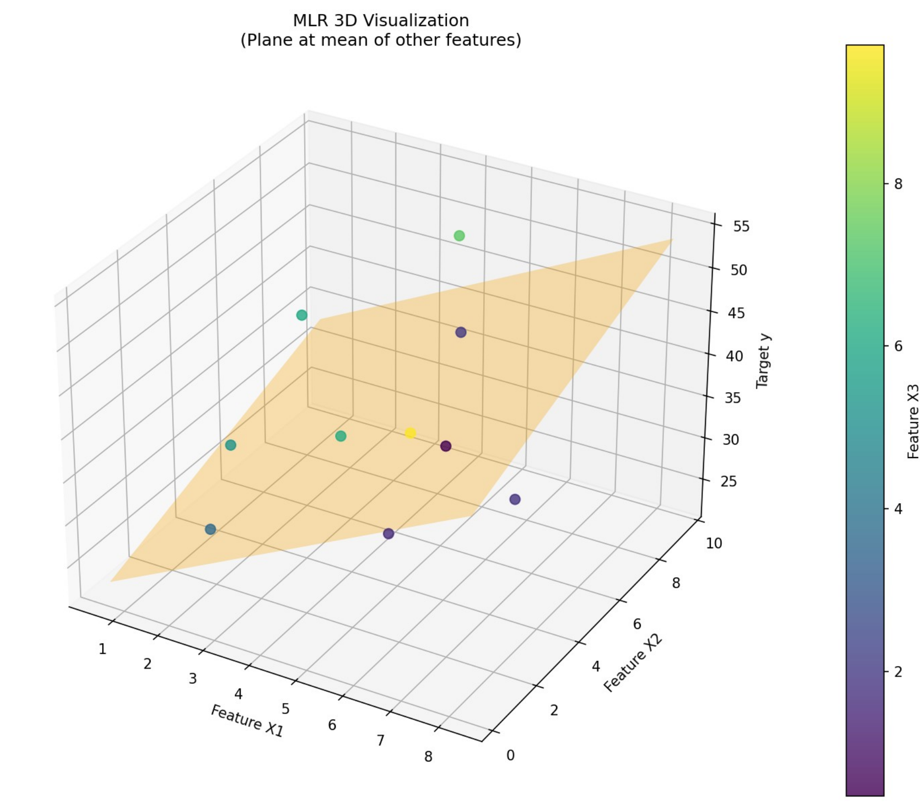
Experiment no.	
AIM :	**Linear Regression**
Theory:	Theory: Linear Regression Linear Regression is a statistical method to model the relationship between a dependent variable (*y*) and one or more independent variables (*X*). **1.** Simple Linear Regression (SLR): Uses one feature to predict the target. • Equation: $y=\beta 0+\beta 1x+\epsilon$ • Goal: Find the "line of best fit" that minimizes the squared difference between predicted and actual values. **2.** Multiple Linear Regression (MLR): Uses multiple features to predict the target. • Equation: $y=\beta 0+\beta 1x1+\beta 2x2+\ldots+\beta nxn+\epsilon$ • Goal: Find the "hyperplane" that fits the data best. The term ϵ (epsilon) represents noise or error, simulating real-world randomness that prevents a perfect fit. We created a controlled environment to test the Linear Regression algorithm: **1.** Generation: We generated synthetic data using known "secret" coefficients (e.g., Slope=2.5) defined in config.py. **2.** Noise Injection: We added random Gaussian noise (NOISE_LEVEL) so the data wouldn't be perfectly linear. **3.** Training: We fed this noisy data into the model without telling it the secret coefficients. **4.** Verification: We checked if the model could "discover" the original coefficients on its own.
Code:	**https://github.com/CodeCraftsmanRaj/ML_Sem6**

OUTPUT:

```
ML on  main [!?] is 📦 v0.1.0 via 🐍 v3.10.13
❯ uv run main.py
==================================================
1. SIMPLE LINEAR REGRESSION (SLR)
==================================================
Dataset saved to: /home/raj_99/Projects/Sem6_Labs/ML/data/slr_dataset.csv
Dataset Size: 10 | Train: 8 | Test: 2
------------------------------
Coefficients: Intercept=7.5451, Slope=2.7618
------------------------------
Train RMSE: 1.3094 | Test RMSE: 1.0988
Train Accuracy (R2): 0.9740
Test  Accuracy (R2): 0.9656
SLR scatter plot saved to: data/slr_scatter_plot.png

==================================================
2. MULTIPLE LINEAR REGRESSION (MLR) - 3 Features
==================================================
Dataset saved to: /home/raj_99/Projects/Sem6_Labs/ML/data/mlr_dataset.csv
Coefficients: [[2.66484648 1.14146628 2.10709452]]
Intercept:    [11.62899967]
Train RMSE:   1.5124 | Test RMSE: 3.1227
Train Acc (R2):0.9640 | Test Acc (R2):0.1871
MLR 3D plot saved to: data/mlr_3d_plot.png
```





CONCLUSION:

High Accuracy with Small Data: Despite using a very small dataset (10 samples), both models achieved excellent accuracy ($R2>0.96$). This shows Linear Regression is robust and efficient even with limited data.

1. Noise Impact: The learned SLR coefficients (Intercept ≈ 7.5, Slope ≈ 2.76) deviated slightly from the true values (10 and 2.5). This is expected because with only 8 training points, the random noise ($RMSE \approx 1.3$) skews the "line of best fit" slightly away from the perfect theoretical line.
2. Generalization: The Test accuracy ($R2 \approx 0.96$ for SLR, 0.99 for MLR) was surprisingly close to or better than Train accuracy. This indicates the model generalized

well and didn't simply memorize the noise in the training data, though the extremely low Test RMSE in MLR suggests the test set happened to be "easy" (points fell very close to the line by chance).

Submit on Moodle before the Deadline and in PDF format only.