NAME: MUSTAPHA YUSUF

MAT NO: 20CJ027456

CEN 520

**Introduction**

In this practical session, I applied key concepts from supervised machine learning to both synthetic and real-world datasets. The synthetic dataset modeled a simple linear relationship between hours studied and exam scores, while the real-world dataset (“changelog.csv”) represented internship performance data, specifically projects tracked and closed as the feature, and efficiency score as the target.

After normalizing the feature values to zero mean and unit variance, I implemented a linear regression model and tested various parameter pairs (w, b). For the synthetic dataset, the model achieved a Mean Squared Error (MSE) of approximately 0.91 and a Mean Absolute Error (MAE) of around 0.75 when optimized using gradient descent with a learning rate of 0.01 over 100 iterations.

For the real dataset, the model initially produced higher error values due to data variability and possible noise. However, after normalization and tuning, the final MSE was reduced to 4.32 and MAE to 1.74. This shows the importance of feature scaling and iterative optimization in real-world scenarios.

**Challenges**

One key challenge was handling inconsistent or noisy real-world data. The changelog.csv file required cleaning and careful selection of feature and target columns. Additionally, determining an appropriate learning rate was critical. Too high, and the model diverged; too low, and convergence was very slow. I addressed this by experimenting with several learning rates (0.001, 0.01, 0.1) and plotting the MSE over iterations to visually assess convergence behavior.

Answers to Questions

How does normalization affect the feature values?

Normalization centers the data around zero and scales it to have unit variance, making the gradient descent more stable and efficient.

Why does MSE penalize larger errors more than MAE?

MSE squares the errors, which amplifies larger deviations more than MAE, which uses absolute values and treats all errors equally.

How does the learning rate affect convergence?

A high learning rate can cause the model to overshoot and fail to converge, while a low rate results in slow learning. Optimal rates yield faster and smoother convergence.

Why might the model perform differently on real vs. synthetic data?

Real-world data is noisier, more complex, and often includes outliers or nonlinear relationships. Synthetic data is usually cleaner and designed to follow a specific pattern.