Applied Deep Learning - Assignment 2

Task 1: Solving Least Squares with Gradient Descent

We applied gradient descent to the full diabetes dataset to solve the least squares problem

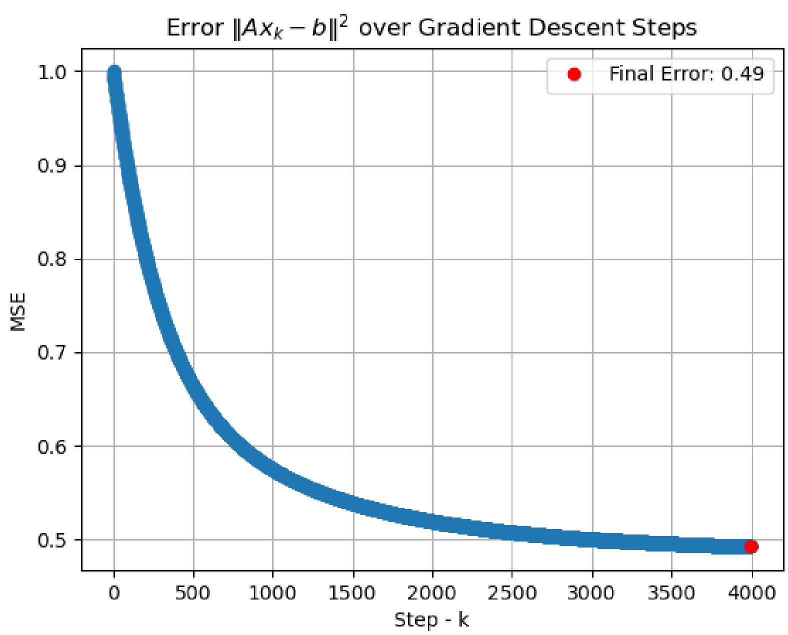
$$min_{\mathrm{x}}||\mathrm{Ax}-\mathrm{b}||^2$$

Where:

- the data matrix $A \in \mathbb{R}^{442\times10}$
- the normalized output vector $b \in \mathbb{R}^{442}$
- the coefficient vector to be optimized $x \in \mathbb{R}^{10}$

Hyperparameters:

- Initial point: $x = \vec{0}$
- Step size (learning rate) $\varepsilon = 0.1$
- Stopping condition: $||\nabla f(x)|| \leq 10^{-4}$
- Max iterations 4000



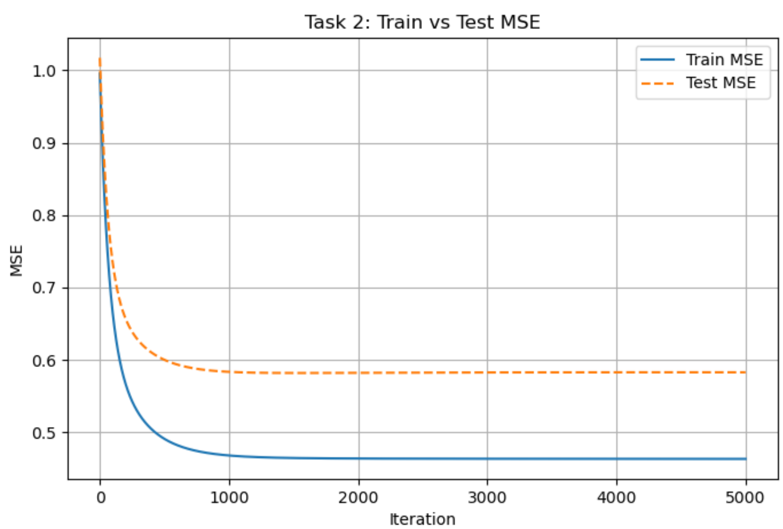
Conclusion:
The error curve converges smoothly, suggesting the gradient descent algorithm is functioning correctly.

Task 2: Train/Test Split and Evaluation

We randomly split the diabetes dataset into 80% training and 20% testing. Gradient descent was applied to the training set only, and both training and testing errors were tracked.

Hyperparameters:

- Step size $\varepsilon = 10^{-3}$
- Stopping condition $||\nabla f(x)|| \leq 10^{-4}$
- Max iterations 5000

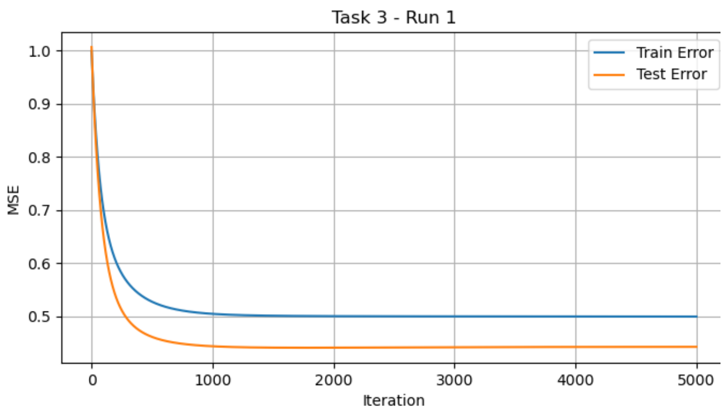


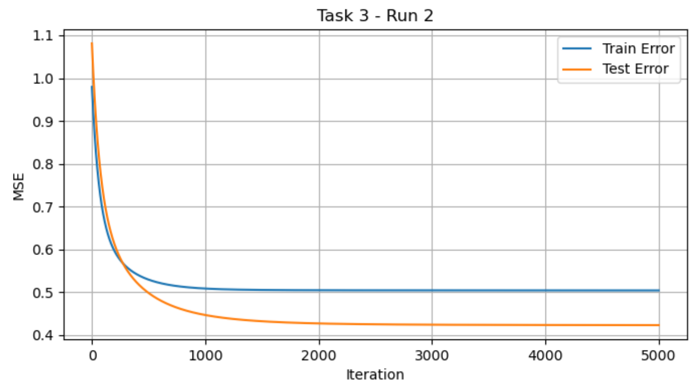
Conclusion:

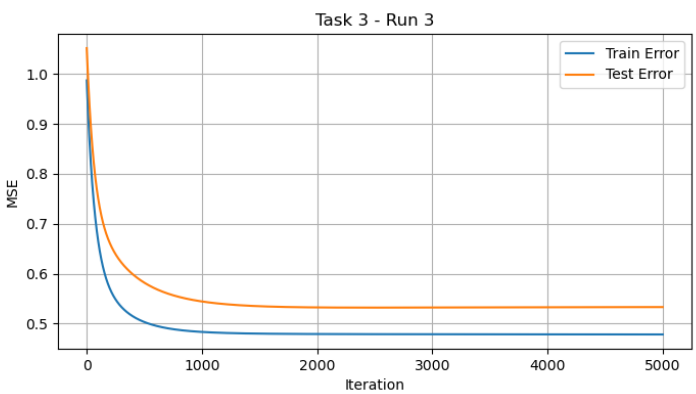
Based on the graph, we observe that both training and testing errors gradually decrease over the iterations and then stabilize. The test error does not increase significantly after reaching its minimum, which indicates the model is not heavily overfitting. There is a slight gap between train and test error, but this is expected. Overall, no strong signs of overfitting or underfitting were observed in this single run. To make a more robust conclusion, multiple splits and additional metrics should be examined.

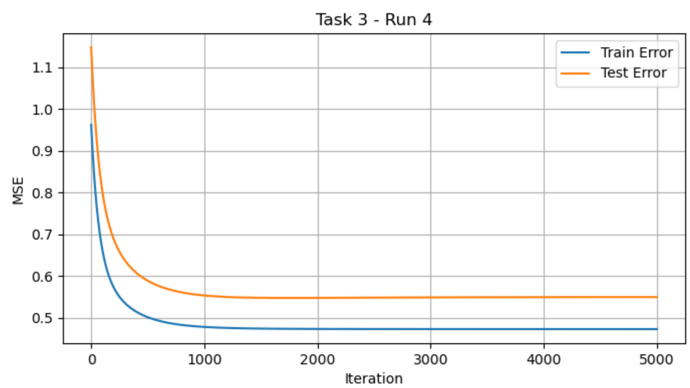
Task 3: 10 Random Splits + AVG and MIN Analysis

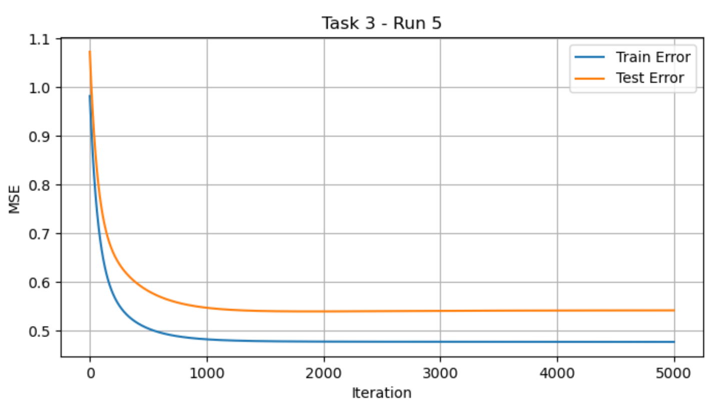
We repeated Task 2 **ten times**, each time using a new 80/20 random split. For each run, we plotted the training and test errors. Then we computed the per-iteration average (AVG) and minimum (MIN) errors across all runs.

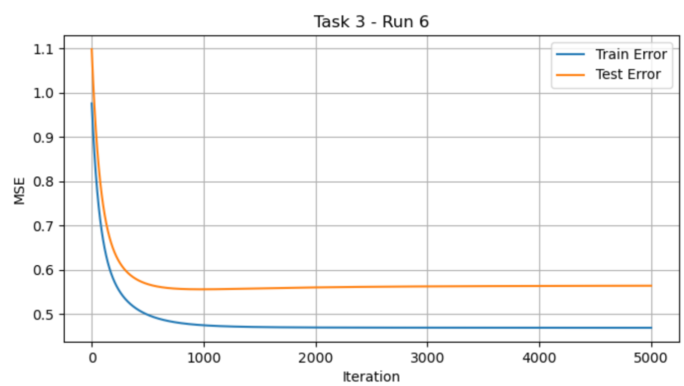


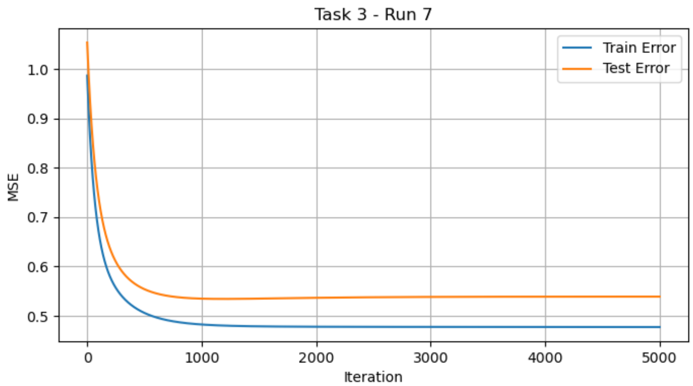




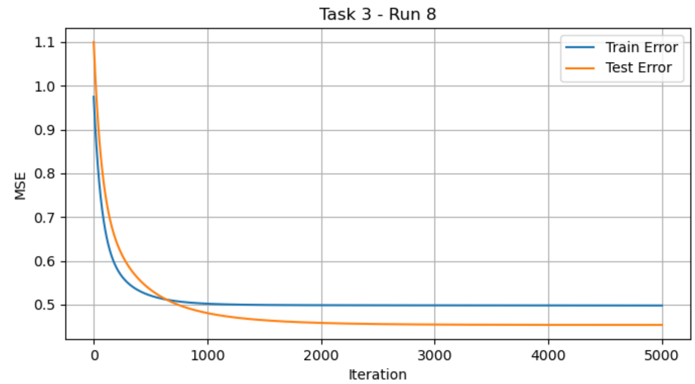




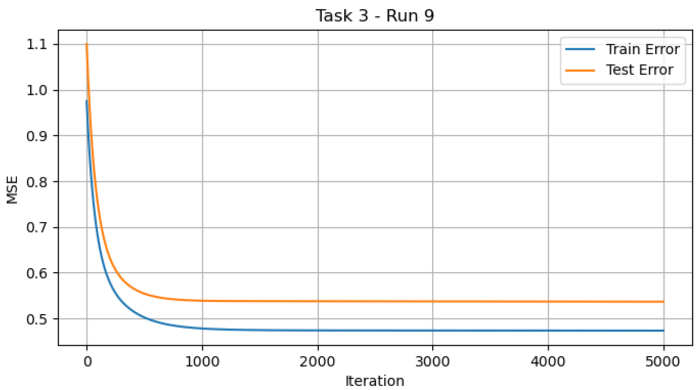




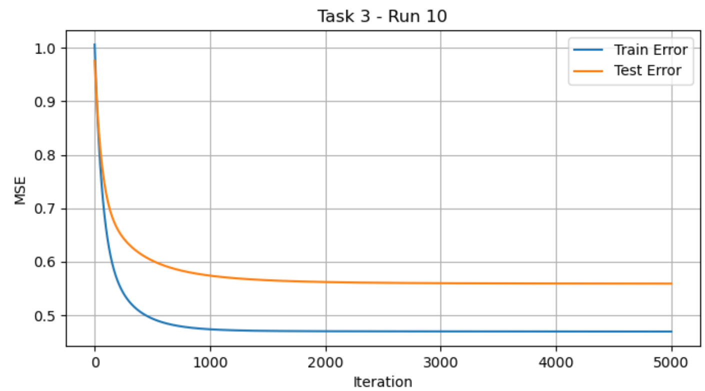
Task 3 - Run 7
Train Error
Test Error
MSE
Iteration



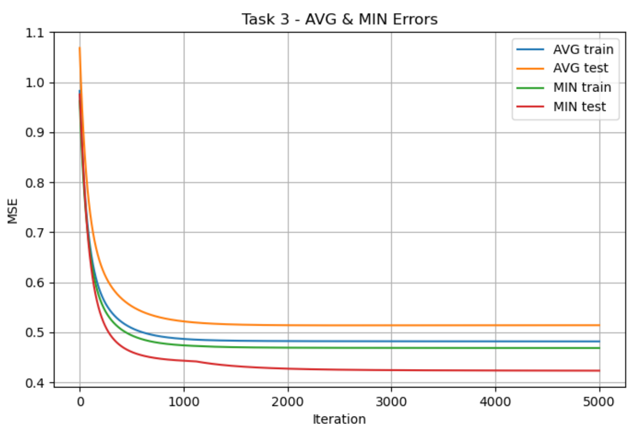
Task 3 - Run 8
Train Error
Test Error
MSE
Iteration



Task 3 - Run 9
Train Error
Test Error
MSE
Iteration



Task 3 - Run 10
Train Error
Test Error
MSE
Iteration



Task 3 - AVG & MIN Errors
AVG train
AVG test
MIN train
MIN test
MSE
Iteration

Analysis:
Each of the 10 runs was performed with a different 80/20 split of the dataset, generated randomly without a fixed seed. This introduces variation in training and testing data between runs. The average curves represent the mean of the errors across runs per iteration, while the minimum curves capture the best performance at each iteration.
Across all runs, the training error consistently decreased as iterations progressed. The test error typically decreased at first, then plateaued, and occasionally slightly increased—suggesting light overfitting in some cases. The growing gap between average training and testing errors indicates some generalization limits of the model.

Conclusion:
The average graph gives an overall view of the algorithm's behavior across different train/test splits, while the minimum test error curve shows the best generalization performance that was achieved. Although the model sometimes slightly overfits, the overfitting is not severe. These plots are useful for determining a practical stopping point for training that balances performance and generalization.