

ML ASSIGNMENT REPORT-FINAL

Simple and Multiple Linear Regression

Fahim Wayez

21-44499-1

Section: B

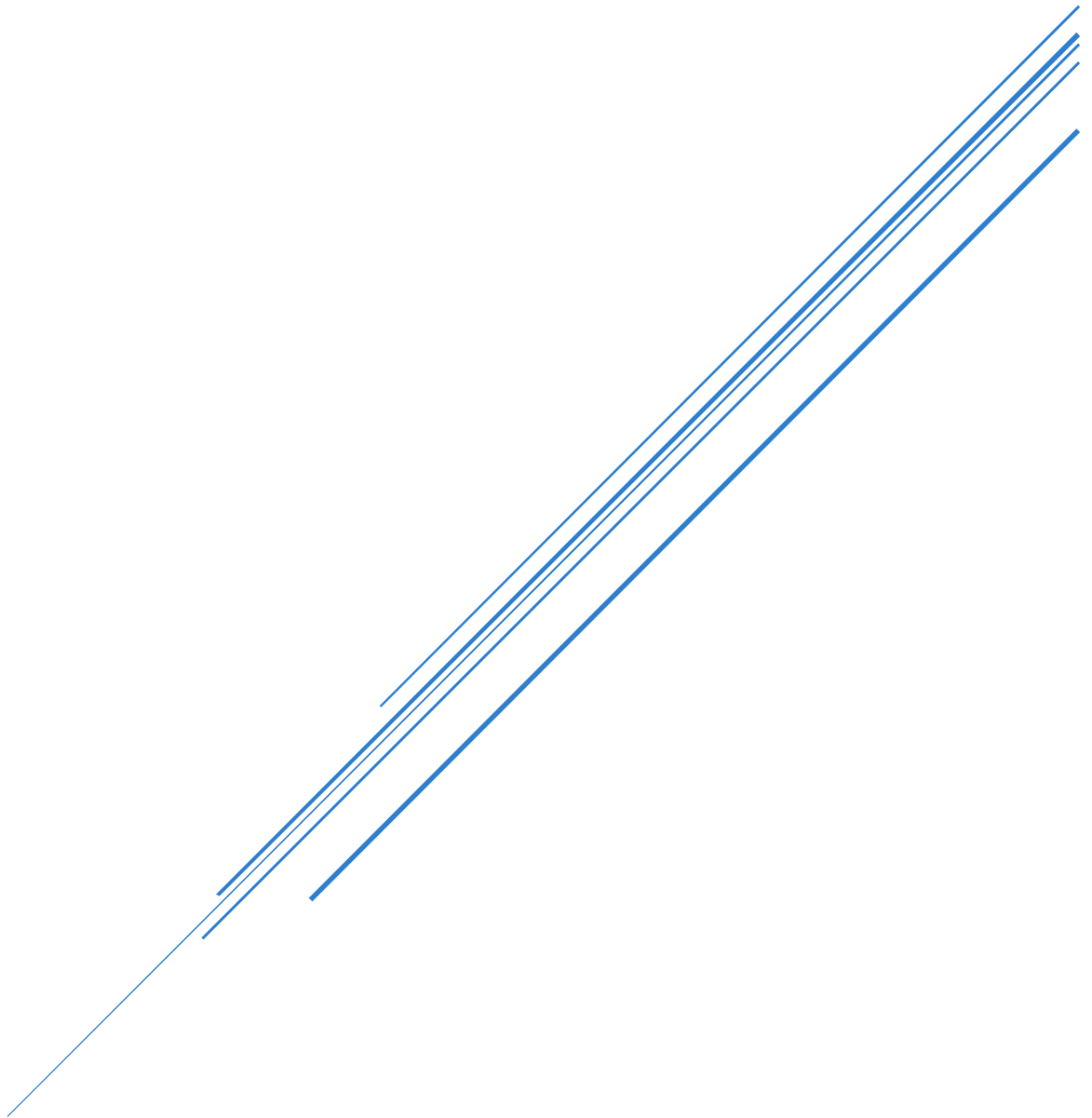


Table of Contents

Simple Regression	2
Terminal Output	2
Generated Plots	2
Multiple Regression	3
Terminal Output	3
Discussion	3

Terminal Output

```

PROBLEMS 36 OUTPUT DEBUG CONSOLE TERMINAL PORTS SQL CONSOLE POSTMAN CONSOLE SQL HISTORY ...
C:\Users\fahim\Downloads\Tenamoy Jibon\10th Semester\ML\Final\linear_regression>python -u "c:\Users\fahim\Downloads\Tenamoy Jibon\10th
Semester\ML\Final\linear_regression\code\simple.py"
Params GD: [254449.99982048 93308.92004027]
Params SGD: [254550.06767351 92946.32643744]
Training RMSE: 64083.51.
Training cost: 2053348364.32.
Test RMSE: 65773.19.
Test cost: 2163056350.22.
Training RMSE SGD: 64084.62.
Training cost SGD: 2053419108.17.
Test RMSE SGD: 65614.89.
Test cost SGD: 2152656638.19.

C:\Users\fahim\Downloads\Tenamoy Jibon\10th Semester\ML\Final\linear_regression>

```

Figure 1: Terminal Output of Simple Regression

Generated Plots

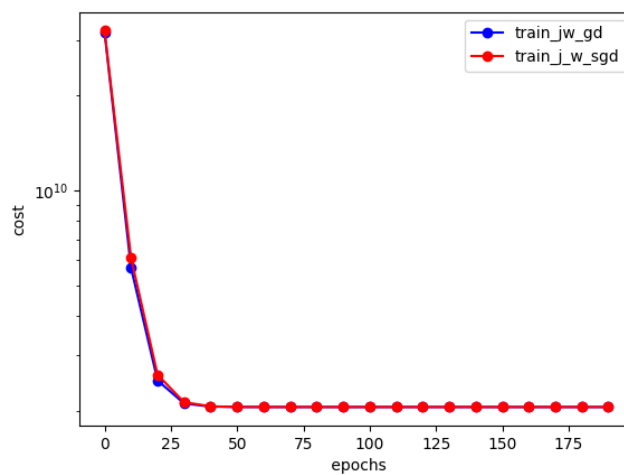


Figure 2: Epochs vs Cost plot

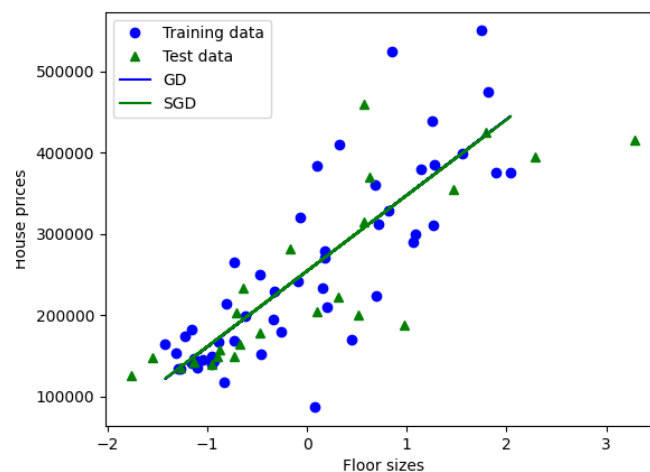


Figure 3: Linear approximation plot for each training method

Multiple Regression

Terminal Output

```
Microsoft Windows [Version 10.0.22631.3527]
(c) Microsoft Corporation. All rights reserved.

C:\Users\fahim\Downloads\Tenamoy Jibon\10th Semester\ML\Final\linear_regression>python -u "c:\Users\fahim\Downloads\Tenamoy Jibon\10th Semester\ML\Final\linear_regression\code\multiple.py"
Params GD: [254449.99982048 78079.18106675 24442.5758378 2075.95636731]
Params SGD: [254580.71771403 77853.15388901 24170.61766679 1508.27966708]
Training RMSE: 61070.62.
Training cost: 1864810304.94.
Test RMSE: 58473.59.
Test cost: 1709580288.69.
Training RMSE SGD: 61075.13.
Training cost SGD: 1865085468.61.
Test RMSE SGD: 58227.86.
Test cost SGD: 1695241650.30.

C:\Users\fahim\Downloads\Tenamoy Jibon\10th Semester\ML\Final\linear_regression>
```

Figure 4: Terminal Output of Multiple Regression

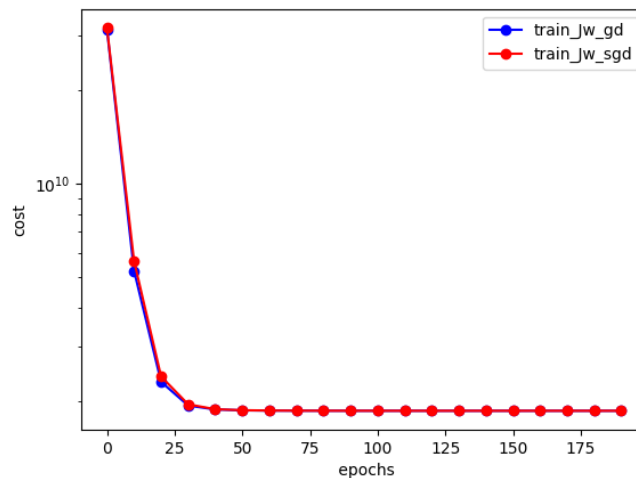


Figure 5: Epochs vs Cost plot

Discussion

The results from both simple and multiple regression experiment basically provide us valuable insights into the performance of gradient descent methods. In the simple regression problem, both batch gradient (GD) and stochastic gradient descent (SGD) yield similar parameter estimates, with only a little variation in the coefficients. But there is a difference in training and test RMSE and costs between the two methods as we can see.

RMSE (Root Mean Square Deviation) measures the average magnitude of the errors and is concerned with the deviations from the actual value. RMSE value with zero indicates that the model has a perfect fit. The lower the RMSE, the better the model and its predictions. And in the context of the cost function, also, lower values are generally better. The cost function represents the error or discrepancy between the predicted values of the model and the actual observed values in the data. Lower cost values imply better fitting of the model to the data, suggesting that the model is performing well in capturing the underlying patterns and relationships in the data.

For the simple regression, both training RMSE and cost are slightly higher for SGD (Training RMSE: 64084.62, Training cost: 2053419108.17) compared to GD (Training RMSE: 64083.51, Training cost: 2053348364.32). But test RMSE and cost are slightly lower for SGD (Test RMSE: 65614.89, Test cost: 2152656638.19) compared to GD (Test RMSE: 65773.19, Test cost: 2163056350.22). Which indicates that overall SGD may have slightly better generalization performance.

In contrast, for the multiple regression scenario, the differences between GD (Training RMSE: 61070.62, Training cost: 1864810304.94, Test RMSE: 58473.59, Test cost: 1709580288.69) and SGD (Training RMSE: 61075.13, Training cost: 1865085468.61, Test RMSE: 58227.86, Test cost: 1695241650.30) resulting performance are almost the same as simple regression. While both methods converge to similar parameter estimates, for training RMSE and cost SGD is slightly higher, and for test RMSE and cost, SGD is slightly lower, suggesting that SGD may be more effective in finding optimal solutions for more complex regression tasks with multiple features. Overall, the results demonstrate the effectiveness of both GD and SGD in minimizing the cost function, with SGD showing potential advantages in terms of computational efficiency and generalization performance for more complex regression problems.