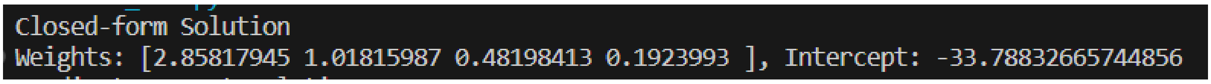
NYCU Introduction to Machine Learning, Homework 1

110550126, 曾家祐

Part. 1, Coding (50%):

(10%) Linear Regression Model - Closed-form Solution

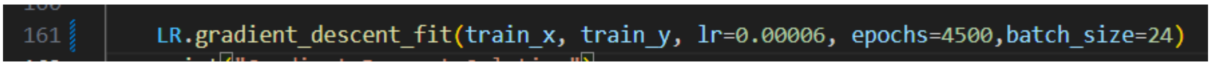
1. (10%) Show the weights and intercepts of your linear model.



```
Closed-form Solution
Weights: [2.85817945 1.01815987 0.48198413 0.1923993 ], Intercept: -33.78832665744856
```

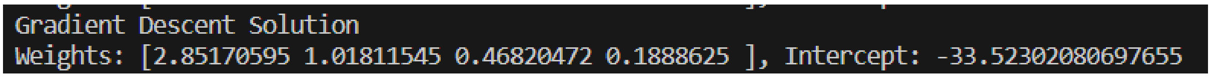
(40%) Linear Regression Model - Gradient Descent Solution

2. (0%) Show the learning rate and epoch (and batch size if you implement mini-batch gradient descent) you choose.



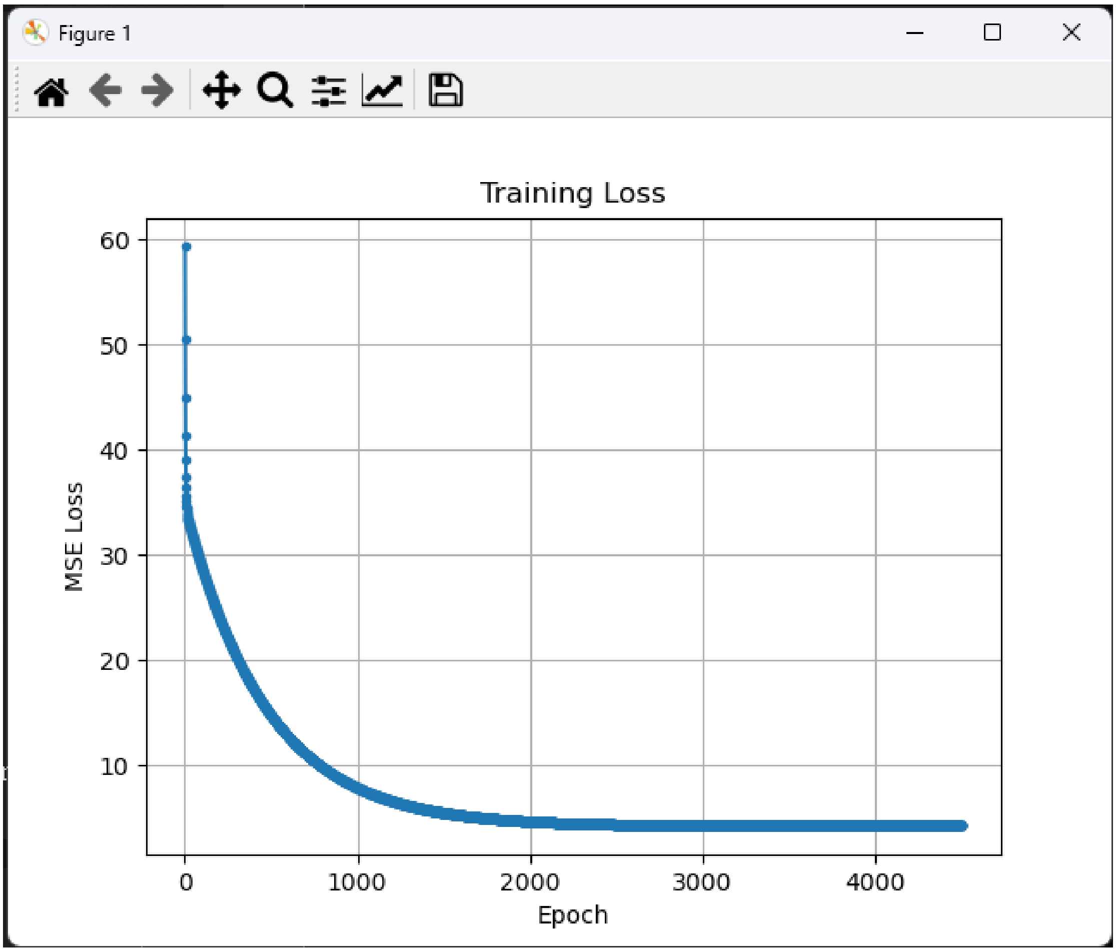
```
    LR.gradient_descent_fit(train_x, train_y, lr=0.00006, epochs=4500,batch_size=24)
```

3. (10%) Show the weights and intercepts of your linear model.

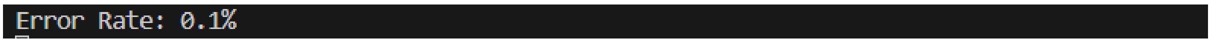


```
Gradient Descent Solution
Weights: [2.85170595 1.01811545 0.46820472 0.1888625 ], Intercept: -33.52302080697655
```

4. (10%) Plot the learning curve. (x-axis=epoch, y-axis=training loss)



5. (20%) Show your error rate between your closed-form solution and the gradient descent solution.



```
Error Rate: 0.1%
```

Part. 2, Questions (50%):

1. (10%) How does the value of learning rate impact the training process in gradient descent? Please explain in detail.

   In gradient descent, every epoch we get an direction to update the weights and and the learning rate impact the how much we update along this direction. (if visualize in hyperplane)

	high learning rate	good training rate	low learning rate
training time	updated fast, so need not much time(epochs) to train	moderate	updated slowly so need more time(epoch) to train
accuracy	may not be reach the best answer (might noe converge or might deverge)	good accuracy	may converge to a local minimun error but not the best answer,

2. (10%) There are some cases where gradient descent may fail to converge. Please provide at least two scenarios and explain in detail.

   1. High learning rate:

      If learning is too high, gradient descent may diverge instead of converge, and the weight and intercept we predict will be extremely high

   2. Low learning rate:

      If learning rate is too small traing may end before we reach the convergence, or stuck in local minimum error rather than global minimum error so the weight and intercept we get may not be the best.

   3. Ill-Conditioned Loss Surfaces:

      If the loss function's contours can be elongated or skewed, It is hard to find the optimal direction to minimize the loss efficiently. The gradient information may mislead the algorithm, causing it to take unnecessarily small steps or oscillate.

3. (15%) Is mean square error (MSE) the optimal selection when modeling a simple linear regression model? Describe why MSE is effective for resolving most linear regression probl

ems and list scenarios where MSE may be inappropriate for data modeling, proposing alternative loss functions suitable for linear regression modeling in those cases.

Usually MSE is optimal selection in simple linear regression model there are some reason

1. Averaging Errors: MSE will measures the average square error, this will punishing the larger error more. this will make the goal of minizing overall prediction errors.
2. convexity: MSE is a convex function. Therefore it will have a unique minimum, and we can use gradient-based optimization algorithm to reach the solution of linear regression.

etc.

1. Outlier:

   Since MSE is sensitive to the outlier (square the difference between prediction and ground truth), the MSE will not have good performance.

   alternative loss function:

   we can use Huberloss, Huber loss combine the benifit of mAE and MSE , if the difference is small it will square the error, if the error is big it will take the absolute error of it, this will make model not that sensitive to outlier. So in this situation it will have better performance than MSE.
2. Heteroscedasticity

   If the variance of the errors is not constant across the range of predictor variables MSE may not be appropriate because it assumes constant variance. I

   Alternative loss function:

   We can use Weighted MSE, assign different weights to different data points based on their estimated variances. This way, you can give more importance to data points with smaller variances.

4. (15%) In the lecture, we learned that there is a regularization method for linear regression models to boost the model’ s performance. (p18 in linear_regression.pdf)

$$E_D(\mathbf{w}) + \lambda E_W(\mathbf{w})$$

4.1. (5%) Will the use of the regularization term always enhance the model's performance? Choose one of the following options: "Yes, it will always improve," "No, it will always worsen," or "Not necessarily always better or worse."

4.2. We know that λ is a parameter that should be carefully tuned. Discuss the following situations: (both in 100 words)

4.2.1. (5%) Discuss how the model' s performance may be affected when λ is set too small. For example, λ=10^(-100) or λ=0

4.2.2. (5%) Discuss how the model' s performance may be affected when λ is set too large. For example, λ=1000000 or λ=10^100

4.1: Not necessarily always better or worse.

4.2.1: When λ is set too small, the regularization term may becomes negligible, and the model will essentially revert to original linear regression, sometimes this will leads to overfiting since model will capture noise because of poor regularization.

4.2.2: When λ is set too large, the regularization term will be much effective and dominate the loss function, this may leed to underfitting because of the low error. Therefore the model we train will be too simple, so unable to catch the relation between this data.