NYCU Introduction to Machine Learning, Homework 1

110550168, 賴御安

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: -33.78832665744901
```

(40%) Linear Regression Model - Gradient Descent Solution

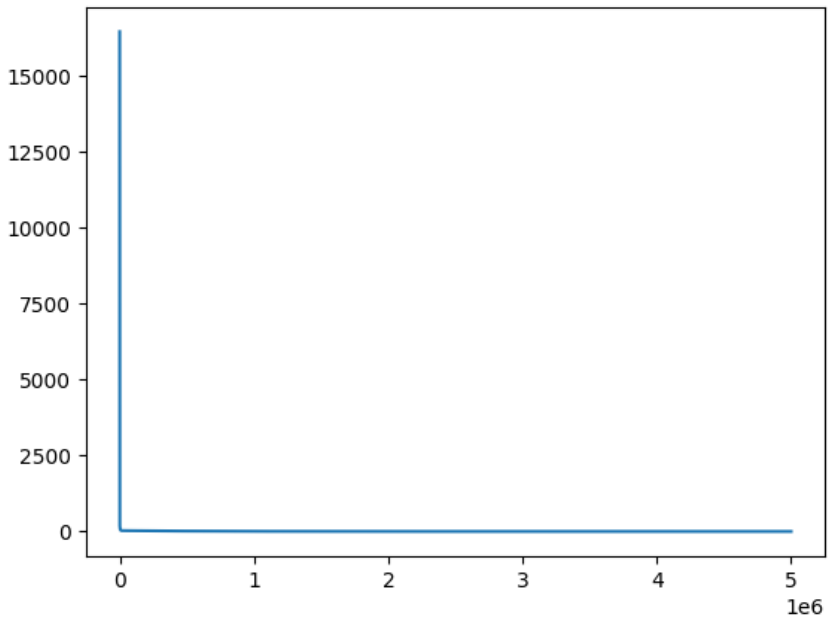
2. (0%) Show the learning rate and epoch (and batch size if you implement mini-batch gradient descent) you choose.

```
LR.gradient_descent_fit(train_x, train_y, lr=0.000032, epochs=5000000)
```

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

```
Gradient Descent Solution
Weights: [2.8487281  1.01520075 0.45280546 0.18551298], Intercept: -33.30090875371474
```

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.**

   Write or type your answer here.

   The learning rate controls the convergence speed by multiplying the gradient by the learning rate. When the learning rate is large, it takes larger steps toward the optimal solution, while using a small learning rate, it gets close to the optimal point slowly. We need to set a reasonable rate so that when we subtract the product from the current weight and intercept for many times, the model will learn to best approximate the function.

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

   Write or type your answer here.

   Gradient Exploding : This means that the learning rate is too high, so that we will jump over the minimum of the loss function and lead to oscillations or divergence.
   Gradient Vanishing Problem: This means that when getting close to the optimal point, the gradient will approach zero, so that gradient descent can struggle to converge or be stuck. Then we need a large epoch and require huge computation.
   One of the solutions to problems above is to set a reasonable rate.

   Saddle Point : Gradient descent may be stuck at the saddle point where the gradient is zero but is not the local minima. This happens more in high dimensional space.
   One of the solutions is to use momentum-based optimizations. It works by adding a fraction of the previous weight update to the current weight update, then builds momentum when descending the loss function.

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

   Write or type your answer here.

   Yes, MSE is the optimal selection when modeling a simple linear regression. The reasons are that MSE is easy to differentiate, and also has a straightforward mathematical form, so that MSE is effective to optimize gradient descent.

   The scenarios that MSE may be inappropriate for data modeling :

   outliers : MSE is sensitive to outliers because it squares the error, and the model may prioritize fitting these extreme data points at the expense of the majority of the data. Huber loss or Tukeys's biweight loss are less sensitive to outliers, may be more suitable.

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

Write or type your answer here.

4.1.A. Not necessarily always better or worse.

4.2.1.A. When λ is small, the regulation term is weak to the loss function. Thus, the model can still lead to overfitting, so we will not prevent it. This means that there may still be overfitting and has a large weight.

4.2.2.A. When λ is large, compared to the loss function, the regulation term is strong, so that the model's coefficients are close to zero. This makes the model focus on regularization more than the weight.