NYCU Introduction to Machine Learning, Homework 1

109550085 , 陳欣妤

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: -33.78832665744835
```

(40%) Linear Regression Model - Gradient Descent Solution

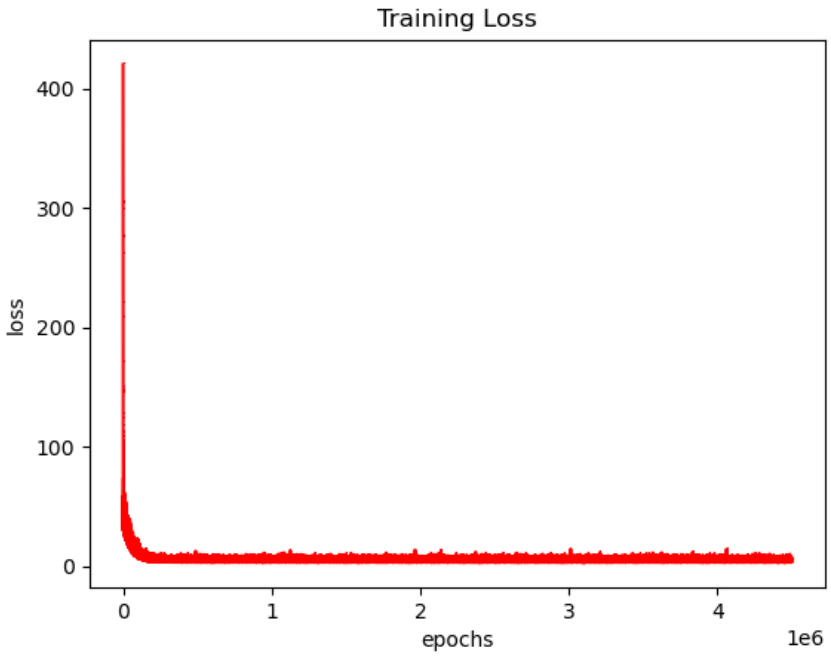
2. (0%) Show the learning rate and epoch (and batch size if you implement mini-batch gradient descent) you choose.

```
LR.gradient_descent_fit(train_x, train_y, lr=3e-3, epochs=4500000)
```

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

```
Gradient Descent Solution
Weights: [2.8585007  1.01962668 0.48426827 0.19044701], Intercept: -33.786687807340854
```

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.

   A high learning rate causes the model parameters to update by large steps at each iteration. This can result in rapid convergence, where the loss decreases quickly. However, it can also lead to overshooting the minimum point, causing the optimization process to diverge or oscillate around the minimum, failing to converge.

   A low learning rate makes small adjustments to the model parameters at each step. While this leads to stable updates and can eventually reach the minimum, it often requires a large number of iterations for convergence. It can be computationally expensive and may get stuck in local minima if the learning rate is too low.

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

   - High Learning Rates

     Using an excessively high learning rate can lead to non-convergence or divergence. With a high learning rate, the parameter updates are large, and the algorithm may not settle at the minimum but rather bounce around it or move away from it. The optimization process may exhibit oscillations or even diverge completely.

   - Ill-Conditioned Loss Surfaces

     Some loss surfaces are ill-conditioned, meaning they have steep and shallow regions with significantly different curvatures. In the presence of an ill-conditioned loss surface, the gradient can be very small or even close to zero in certain directions, making it difficult for the optimization algorithm to make progress. Additionally, saddle points can trap the optimization process as the gradient is zero but it is not a local minimum.

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.

   Whether MSE is the optimal choice for modeling a simple linear regression model depends on the specific characteristics of data and the goals of modeling task.

MSE is a straightforward and convex loss function, which makes it easy to work with mathematically. It can be minimized efficiently using gradient-based optimization methods.

MSE gives substantial weight to large errors, making it sensitive to outliers in data. A hybrid loss function that combines the advantages of both MSE and absolute error (L1 loss). It is less sensitive to outliers than MSE.

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."

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

When λ is set too small, the regularization term becomes negligible, effectively vanishing from the loss function. This can lead to overfitting, where the model becomes highly flexible and fits the training data extremely closely. The model may capture noise in the data, resulting in poor generalization to unseen data. Essentially, the model will prioritize minimizing the squared error term, potentially leading to excessively complex models with high variance.

When λ is set too large, the regularization term dominates the loss function. This can result in underfitting, where the model's flexibility is severely constrained, and it may struggle to capture meaningful patterns in the data. The model will prioritize minimizing the regularization term, causing it to become too simple and biased. Consequently, it may perform poorly on both the training and validation datasets, failing to capture the underlying relationships.