NYCU Introduction to Machine Learning, Homework 1

110550080, 何曉嫻

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

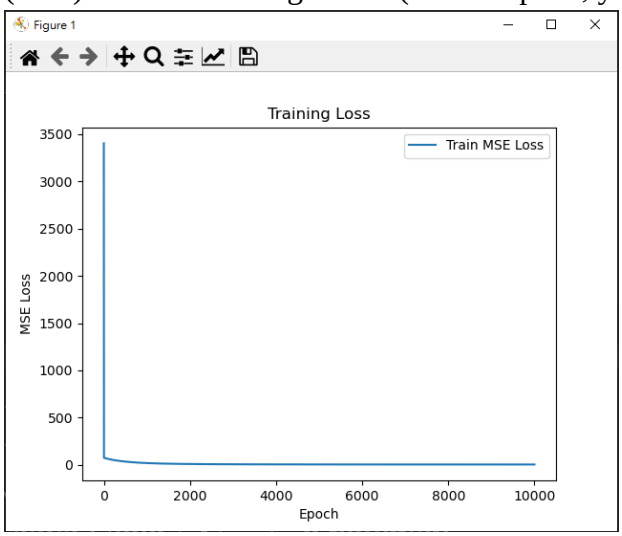
2. (0%) Show the learning rate and epoch (and batch size if you implement mini-batch gradient descent) you choose.

```
LR.gradient_descent_fit(train_x, train_y, lr=0.0001, epochs=10000)
```

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

```
Gradient Descent Solution
Weights: [2.84516406 1.01499322 0.4084954  0.18334555], Intercept: -32.96843949495309
```

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: 0.2%
```

Part. 2, Questions (50%):

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

   The learning rate controls how quickly the step size at each iteration when updating the model's parameters to minimize the loss function. Smaller learning rates need more training epochs given the smaller changes made to the weights each update, the larger learning rates result in rapid changes and require fewer training epochs.

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

   (1) Learning Rate is Too High or Too Low : If the learning rate is too high, the algorithm may overshoot the optimal parameters and fail to converge. Conversely, if the learning rate is too low, the algorithm may take too long to converge or get stuck in a local minimum.

   (2) Poor Initialization of Parameters : If the initial values of the model parameters are too far from the optimal values, gradient descent may not converge. This is because the gradient descent may get stuck in a local minimum.

   (3) Poor Feature Scaling : If the features in dataset are not properly scaled, and gradient descent may not converge. Because features with larger ranges will take longer to converge.

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.

   Yes.

   (1) MSE make a convex optimization problem in the linear regression. Convex optimization problems have a single global minimum, which ensures that gradient-based optimization algorithms like gradient descent will converge to the global minimum. Thus, obtain a unique and optimal solution for linear regression problems.

   MSE leads to a closed-form solution for the regression coefficients. This closed-form solution is able to compute the values of the coefficients without the need for iterative optimization algorithms. It simplifies the computation and can be particularly advantageous when dealing with smaller datasets or when you require quick model parameter estimation.

   (2) If the dataset contains outliers or extreme values, MSE may be inappropriate for data modeling. MSE is sensitive to outliers in the data, since a single large error can dominate the total loss and pull the line of best fit away from the majority of

the data points. Alternative loss functions like Huber loss, which is less sensitive to outliers, may be more appropriate.
MSE is expressed in squared units of the dependent variable, which can make it difficult to compare across different datasets or units of measurement. MSE gets pronounced based on whether the data is scaled or not. Alternative loss functions like Weighted Least Squares, where weights are assigned based on the variance of errors, may be more appropriate.

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

"Not necessarily always better or worse."

(1) Overfitting: When λ is very small or equal to zero, the regularization term has minimal impact, and the model reduces to standard linear regression without regularization. In such cases, the model may be prone to overfitting, where it fits the training data very closely, capturing noise and outliers. Overfitting models perform well on the training data but poorly on testing data.
High Variance: When λ is very small, the model's coefficients are not penalized much for being large. As a result, the model can have high variance, which means it is highly sensitive to small changes in the training data. This makes it less robust and generalizable.

(2) Underfitting: When λ is extremely large, the regularization term dominates the loss function, causing the model to heavily penalize the magnitude of the coefficients. This leads to underfitting, where the model is too simplistic and fails to capture the underlying patterns in the data. Underfit models perform poorly on both the training and validation/test data.
High Bias: High λ results in high bias because it forces the model to be too simple and constrained. Such models cannot adapt well to the data, and their predictive power is severely limited.