Learning to Rank using Linear Regression

Shraddha Dwivedi
UB Person Number: 50290266
Computer Science and Engineering
University at Buffalo
sdwivedi@buffalo.edu

1. Introduction

This report is about solving the Learning to Rank problem using Linear Regression. We solved this problem using the two approaches of Linear Regression.
1. Closed-form solution
2. Stochastic Gradient Descent.

Linear Regression

Linear regression is a linear approach to modelling the relationship between a scalar response (or dependent variable) and one or more explanatory variables (or independent variables).

Closed-form Solution

Closed-form solutions are non-iterative solution which gives the equation to direct calculate the weight matrix.

Stochastic Gradient Descent

Stochastic gradient descent (often shortened to SGD), also known as incremental gradient descent, is an iterative method for optimizing a differentiable objective function, a stochastic approximation of gradient descent optimization. In stochastic (or "on-line") gradient descent, the true gradient of is approximated by a gradient at a single example:

$$w := w - \eta \nabla Q_i(w).$$

As the algorithm sweeps through the training set, it performs the above update for each training example.

2 Hyperparameters

Hyperparameter are the variable which determines the structure of the network. It also determines that how the network is trained. We set the hyperparameters before training the model. Hyperparameter varies for different model training algorithm.

2.1 Stochastic Gradient Descent

2.1.1 Learning Rate

Learning rate is the hyperparameter that controls how much we are adjusting the weights of our neural network

Table 1. Varying Learning rate

Lambda (La)=2, initial_weight scaling factor=220

S.no	Learning rate	E_rms Testing
1	0.01	0.623
2	0.1	0.656

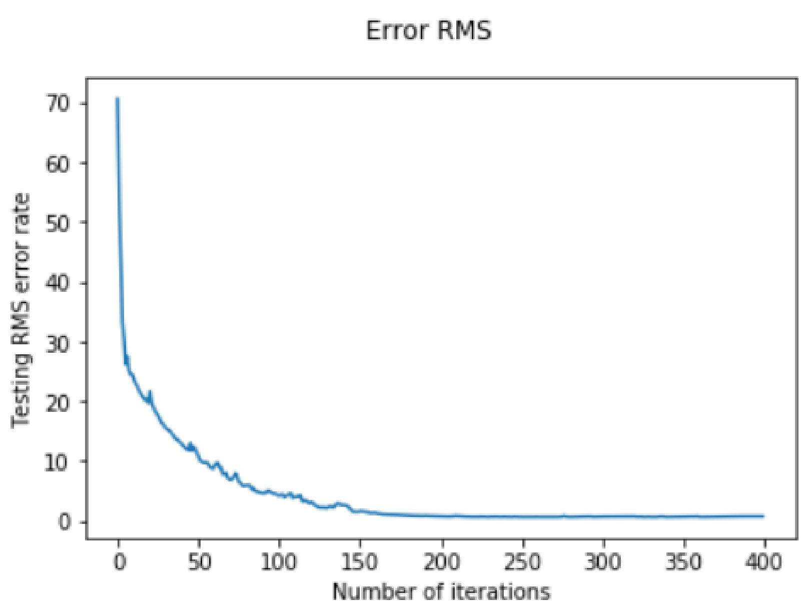

Fig 2.1.1.1: Learning rate: 0.01

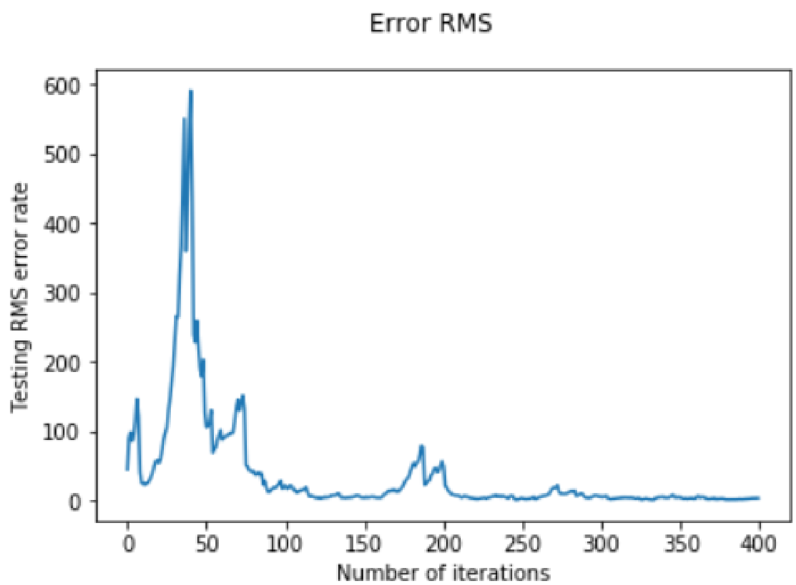

Fig 2.1.1.2: Learning rate: 0.1

2.1.2 Initial Weight Scaling Factor

Initially we will multiply the W(weight calculated in closed form solution) by some weight, which is called as initial scaling factor.

Table 2. Varying Weight Scaling Factor

Lambda(La) : 2, Learning rate: 0.01

S.no	Initial weight Scaling factor	Testing Accuracy
1	220	0.623
2	800	0.623

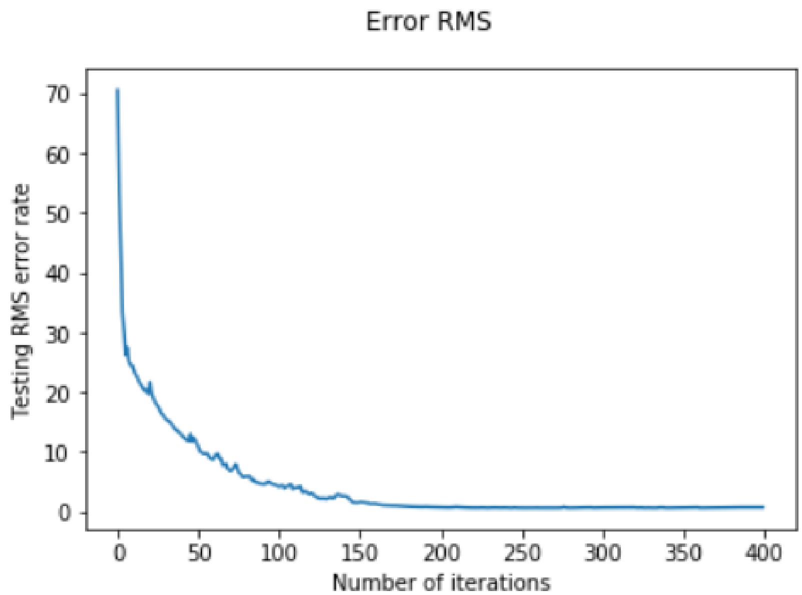

Fig 2.1.2.1: Initial weight scaling factor: 220

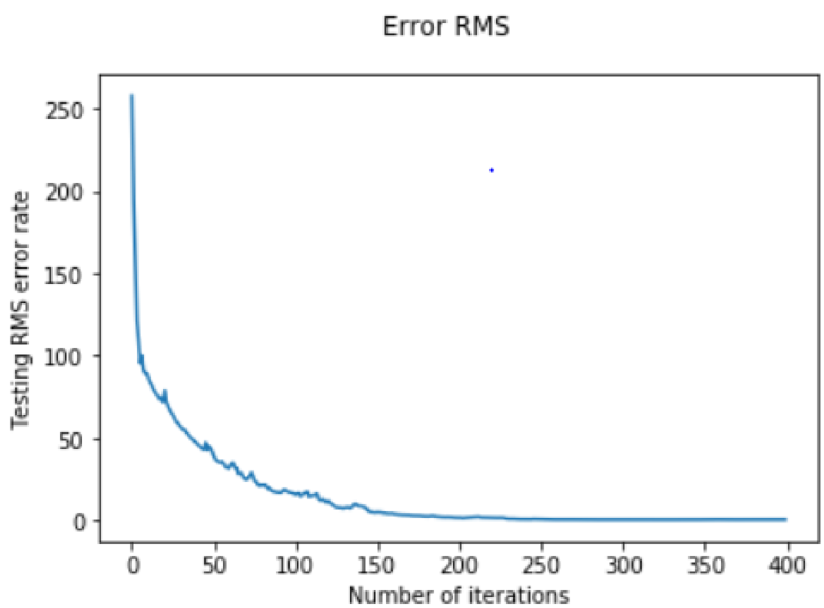

Fig 2.1.2.2: Initial weight scaling factor: 800

2.1.3 Lambda (λ)

Lambda is a hyperparameter that affects what hypothesis is chosen. Hypothesis is chosen that minimizes the cost. Lambda governs the relative importance of the regularization term.

Table 3. Varying Lambda

Learning rate: 0.1 , Initial weight scaling factor: 220

S.no	Lambda	Testing Accuracy
1	1	0.634
2	2	0.656
3	4	30

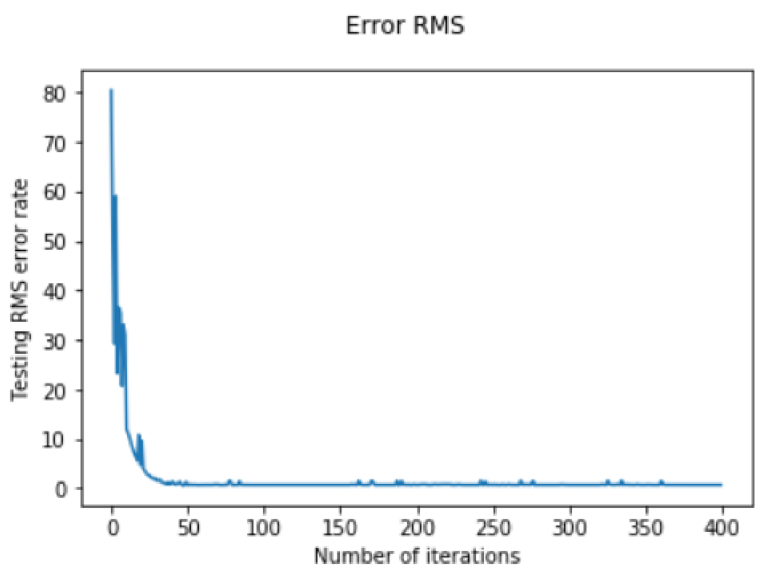

Fig 2.1.13.1: λ=1

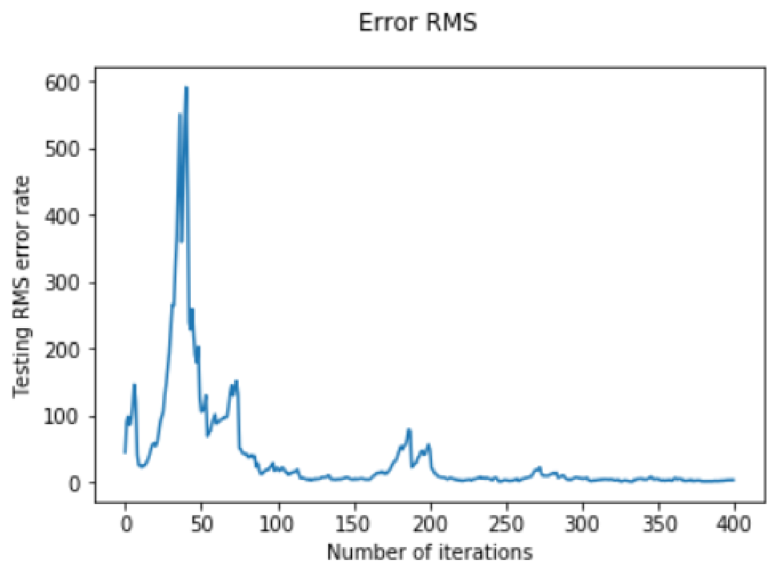

Fig 2.1.3.2: λ=2

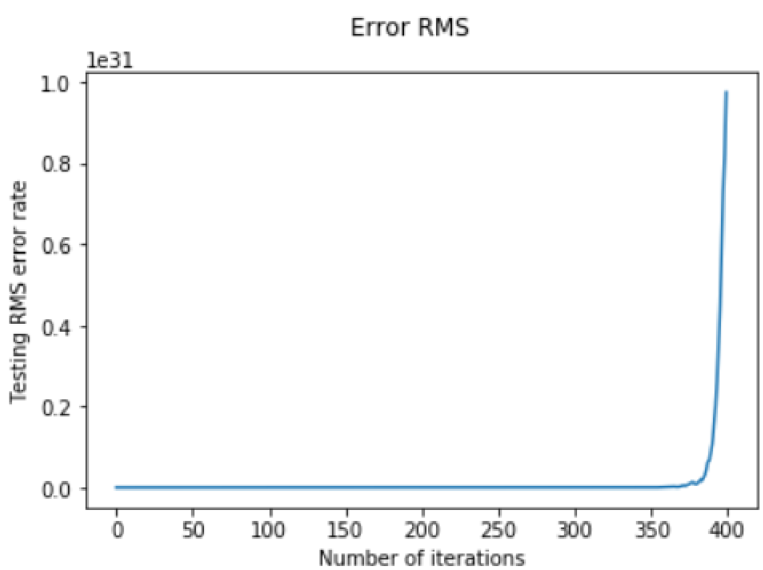

Fig 2.3.3: λ=4

2.2 Closed-form Solution

2.2.1 Lambda

Lambda is a hyperparameter that affects what hypothesis is chosen.

Hypothesis is chosen that minimizes the cost. Lambda governs the relative importance of the regularization term.

Table 4. Varying Lambda

M: 10

S.no	Lambda	Testing Accuracy
1	0.03	0.634
2	0.5	0.656

2.2.1 M

M is the number of basis function we have in the .

Table 5. Varying M

Lambda: 0.03

S.no	M	Testing Accuracy
1	10	0.627
2	15	0.627
3	25	0.627

3. Conclusion

We can infer that stochastic gradient descent method is better than closed form solution for solving this Learning to Rank problem using Linear Regression. For stochastic gradient descent, we get the best result when the learning rate is 0.01, lambda is 2 and initial weight factor is 220. For closed form solution, result won't vary much by changing the hyper parameter. But we can infer that we get the better result when M is 10 and lambda is 0.03.

Training data set	Validation data set	Test data set	Lambda	initial weight factor	Learning rate	E_rms Training	E_rms Validation	E_rms Testing
55699	6962	6962	2	220	0.01	0.549	0.538	0.623
55699	6962	6962	2	220	0.1	0.572	0.56	0.656
55699	6962	6962	4	400	0.1	54.6	53.9	54.8
55699	6962	6962	4	220	0.1	29.9	29.5	30
55699	6962	6962	1	220	0.1	0.563	0.552	0.634
55699	6962	6962	2	220	0.1	0.566	0.556	0.636
55699	6962	6962	2	300	0.05	0.559	0.552	0.634

Fig 3.1 Summarized table when hyperparameters are changed for Stochastic Gradient Descent

Training data set	Validation data set	Test data set	Lambda(C_Lambda)	M(no of clusters)	E_rms Training	E_rms Validation	E_rms Testing
55699	6962	6962	0.03	10	0.549	0.538	0.627
55699	6962	6962	0.5	10	0.549	0.538	0.628
55699	6962	6962	0.8	25	0.549	0.538	0.627
55699	6962	6962	0.03	10	0.549	0.538	0.627
55699	6962	6962	0.03	10	0.549	0.538	0.627
55699	6962	6962	0.03	10	0.549	0.538	0.627

Fig 3.1 Summarized table when hyperparameters are changed for Closed form solution

References

[1] https://www.geeksforgeeks.org/

[2] https://medium.com/the-theory-of-everything/

[3] https://en.wikipedia.org/