CSE 574 Machine Learning

Project 2 Report

- Gautam Shende 50245840

- Sujit Singh 50247206

In this project we are performing linear regression on LETOR (Source: Microsoft) dataset and synthetically generated dataset to find out the optimal weights giving minimal error. For this, we are using two approaches

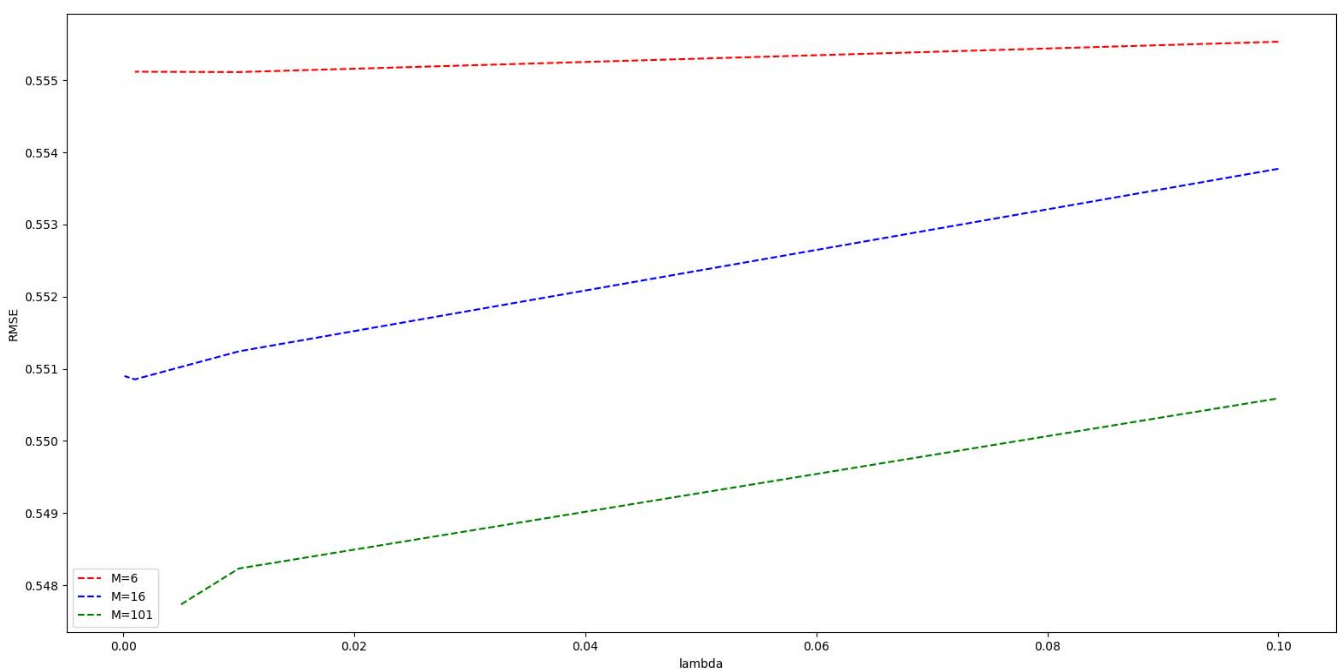
1. Closed Form solution

2. Stochastic gradient descent

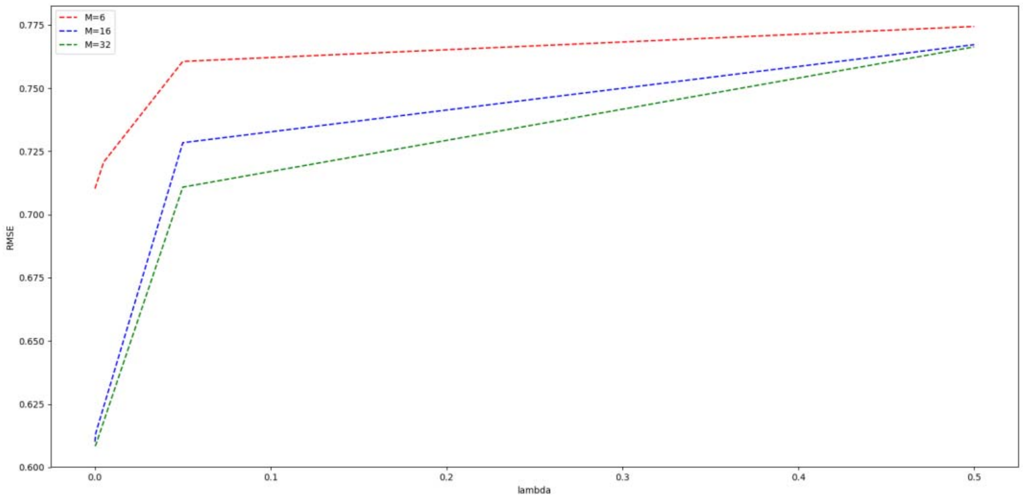
Hyper Parameters tuning:

CLOSED FORM SOLUTION:

Tuning the hyper parameters in the closed form solution was easier as the only hyper-parameter was lambda which was for error regularization.



Lambda vs RMSE in LETOR



Lambda vs RMSE in Synthetic dataset

Please note that the given graphs are sharp because we have plotted only intermediate points. Otherwise the graph should have been a curve. However, this helped in general understanding of the RMSE with lambda. Very high value of lambda increased the error as expected because lambda introduces a penalty for higher values of coefficients to avoid overfitting.

For synthetic data, we tried various values of M and lambdas to obtain the following results

Please note here the calculated RMSE is sqrt 2 * original RMSE as we were using a factor of sqrt 2 for convenience while comparing models.

M = 6 lambda = 5e-05 RMSE = 1.00446901755

M = 6 lambda = 0.005 RMSE = 1.01930431164

M = 6 lambda = 0.05 RMSE = 1.07561490878

M = 6 lambda = 0.5 RMSE = 1.0951576188

M = 16 lambda = 5e-05 RMSE = 0.862886361467

M = 16 lambda = 0.0005 RMSE = 0.867656167931

M = 16 lambda = 0.05 RMSE = 1.03012805499

M = 16 lambda = 0.5 RMSE = 1.08497607961

M = 32 lambda = 5e-05 RMSE = 0.860428861814

M = 32 lambda = 0.0005 RMSE = 0.861013081583

M = 32 lambda = 0.5 RMSE = 1.08374451554

M = 32 lambda = 0.05 RMSE = 1.00528843159

Decreasing the value of lambda increased accuracy but we choose not to take lambda as low as 5e-05 because that would lead to overfitting. We picked 0.0005 instead.

For the LETOR data set, we had the following results :

M = 6 lambda = 0.1 RMSE = 0.78564243338

M = 6 lambda = 0.01 RMSE = 0.78504783115

M = 6 lambda = 0.001 RMSE = 0.78505481561

M = 16 lambda = 0.1 RMSE = 0.783151973714

M = 16 lambda = 0.01 RMSE = 0.779573535892

M = 16 lambda = 0.001 RMSE = 0.779024463378

M = 16 lambda = 0.0001 RMSE = 0.779089265979

M = 101 lambda = 0.0001 RMSE = 0.772867684297

M = 101 lambda = 50000 RMSE = 0.803967422135

M = 101 lambda = 500 RMSE = 0.803400643375

M = 101 lambda = 0.5 RMSE = 0.781500523101

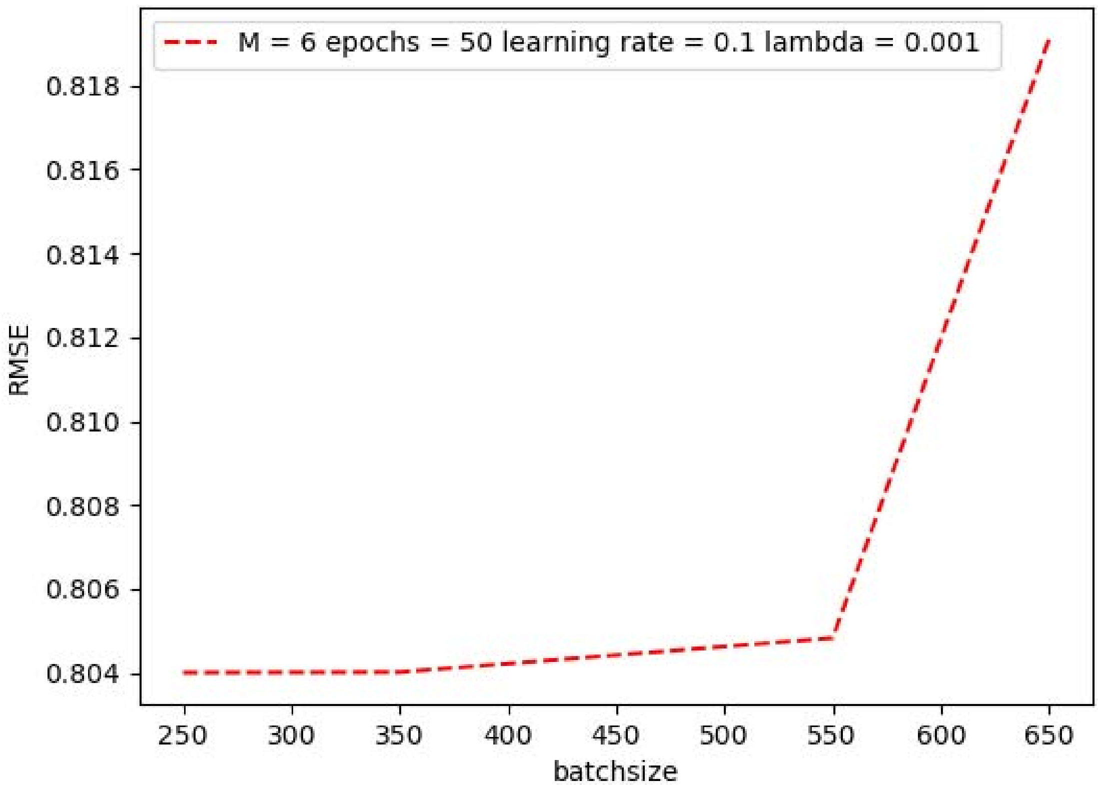
M = 101 lambda = 0.005 RMSE = 0.774613226669

M = 101 lambda = 0.01 RMSE = 0.775316726553

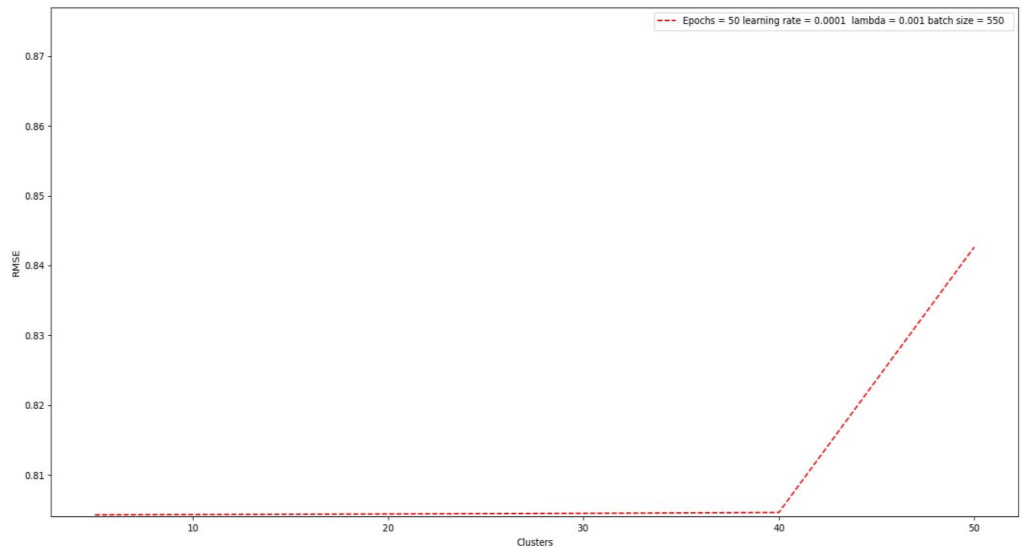
M = 101 lambda = 0.1 RMSE = 0.778652324828

So we choose M = 101, lambda = 0.0001

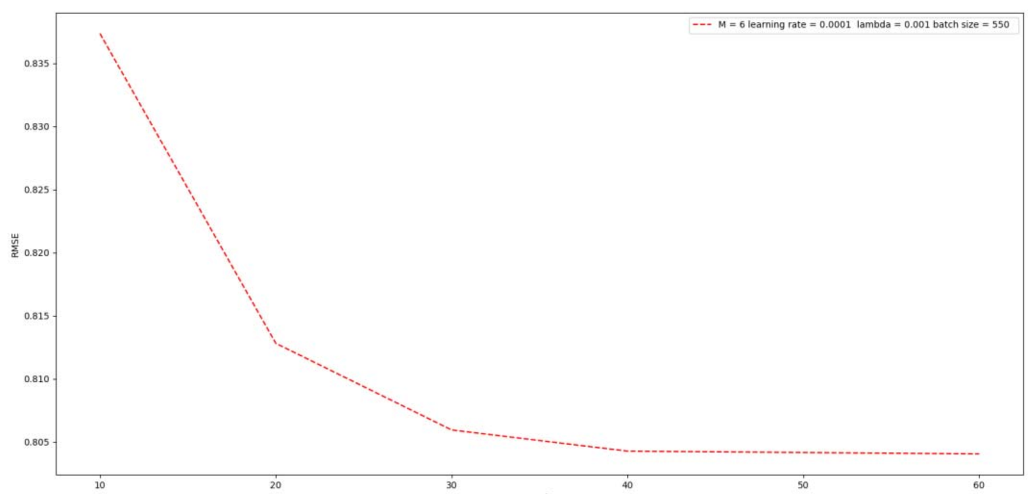
Keeping all other features constant, we varied single parameter to understand the flow of RMSE with each feature. These are the graphs on the LETOR dataset:

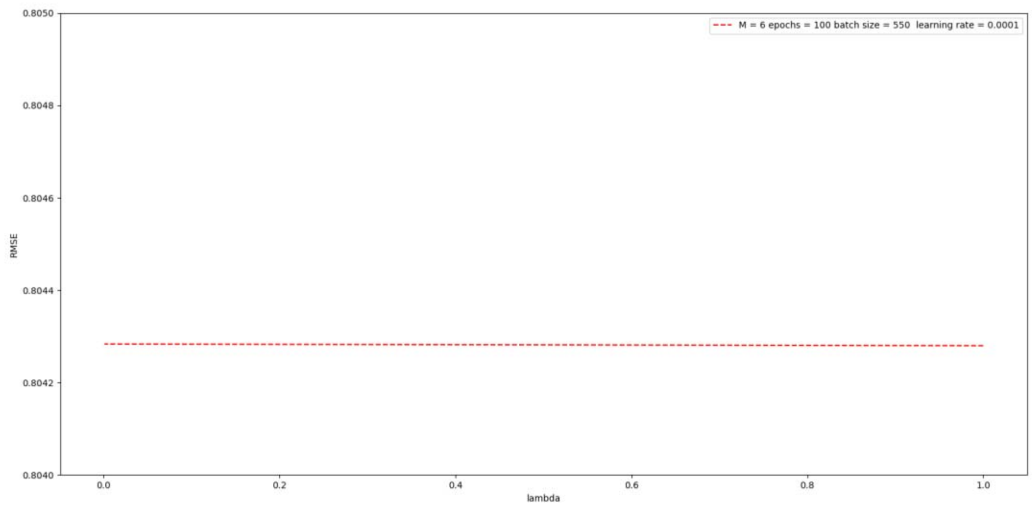


Taking too high batch size increases error.

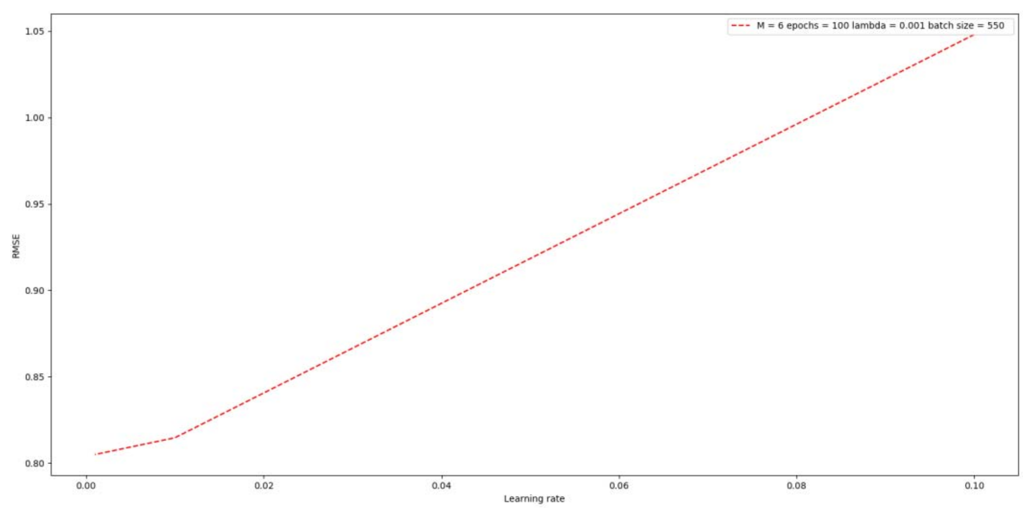


Dividing the dataset into high number of clusters increases the redundancy. It was observed that too many clusters introduced high error.





The value of lambda did not affect the error. It is just to keep in check the overfitting condition.



Learning Rate : Taking too high lambda value does not help in finding local minimum. Very low value of lambda fails to converge. Hence we need to take a value that is not too low to converge and not too high to skip local minimum.

Using this information and various results for the 2 datasets:

Testing on LETOR dataset:

M = 6 epochs = 100 batch size = 5500 learning rate = 0.001 lambda = 0.1 RMSE = 0.975197901462

M = 6 epochs = 100 batch size = 550 learning rate = 0.001 lambda = 0.1 RMSE = 0.804798661668

M = 6 epochs = 100 batch size = 550 learning rate = 0.1 lambda = 0.1 RMSE = 1.04819118267

M = 6 epochs = 100 batch size = 550 learning rate = 1e-05 lambda = 0.1 RMSE = 0.83747954552

M = 6 epochs = 100 batch size = 550 learning rate = 1e-05 lambda = 0.01 RMSE = 0.837479114581

M = 6 epochs = 100 batch size = 550 learning rate = 0.0001 lambda = 0.01 RMSE = 0.804249881522

M = 6 epochs = 100 batch size = 550 learning rate = 0.0001 lambda = 0.01 RMSE = 0.804283509712

M = 6 epochs = 100 batch size = 1600 learning rate = 0.0001 lambda = 0.01 RMSE = 0.825866239285

M = 6 epochs = 500 batch size = 550 learning rate = 1e-05 lambda = 0.01 RMSE = 0.803979505342

M = 16 epochs = 100 batch size = 1600 learning rate = 0.0001 lambda = 0.01 RMSE = 0.817699715055

M = 16 epochs = 500 batch size = 550 learning rate = 1e-05 lambda = 0.01 RMSE = 0.804260048507

M = 16 epochs = 500 batch size = 550 learning rate = 1e-05 lambda = 0.01 RMSE = 0.804260048507

M = 16 epochs = 500 batch size = 1000 learning rate = 1e-05 lambda = 0.001 RMSE = 0.804132096905

M = 16 epochs = 100 batch size = 1000 learning rate = 0.0001 lambda = 0.001 RMSE = 0.804268344804

*M = 16 epochs = 100 batch size = 1000 learning rate = 0.0001 lambda = 0.001 RMSE = 0.804025956505

*M = 16 epochs = 100 batch size = 10000 learning rate = 0.0001 lambda = 0.001 RMSE = 0.913959122444

*M = 16 epochs = 100 batch size = 10000 learning rate = 1e-05 lambda = 0.001 RMSE = 0.916815480462

M = 16 epochs = 100 batch size = 100 learning rate = 1e-05 lambda = 0.001 RMSE = 0.80442090063

M = 16 epochs = 500 batch size = 100 learning rate = 1e-05 lambda = 0.001 RMSE = 0.804423218557

M = 16 epochs = 500 batch size = 100 learning rate = 1e-06 lambda = 0.001 RMSE = 0.804252625489

M = 16 epochs = 500 batch size = 100 learning rate = 1e-06 lambda = 0.01 RMSE = 0.804252561163

M = 16 epochs = 500 batch size = 100 learning rate = 1e-06 lambda = 1 RMSE = 0.804245536615

M = 16 epochs = 2000 batch size = 100 learning rate = 1e-06 lambda = 1 RMSE = 0.804418247782

M = 31 epochs = 2000 batch size = 100 learning rate = 1e-06 lambda = 1 RMSE = 0.804382360253

M = 101 epochs = 2000 batch size = 100 learning rate = 1e-06 lambda = 1 RMSE = 0.804408763002

We fixed the hyper paramters as follows :

M = 16, epochs = 100, batchsize = 1000, learning_rate = 0.0001, lambda = 0.001

2)Testing on SYNTHETIC Data Set :

M = 6 epochs = 2000 batch size = 100 learning rate = 1e-06 lambda = 1 RMSE = 1.12453667003

M = 6 epochs = 2000 batch size = 100 learning rate = 1e-05 lambda = 1 RMSE = 1.09895093467

M = 6 epochs = 2000 batch size = 100 learning rate = 0.0001 lambda = 1 RMSE = 1.09885778049

M = 6 epochs = 2000 batch size = 100 learning rate = 0.001 lambda = 1 RMSE = 1.09819087511

*M = 6 epochs = 2000 batch size = 100 learning rate = 0.01 lambda = 1 RMSE = 1.09810001407

*M = 6 epochs = 2000 batch size = 100 learning rate = 0.1 lambda = 1 RMSE = 1.59184402008

*M = 16 epochs = 2000 batch size = 100 learning rate = 0.1 lambda = 1 RMSE = 1.67067111912

M = 16 epochs = 2000 batch size = 100 learning rate = 0.001 lambda = 0.1 RMSE = 1.09781658613

M = 16 epochs = 2000 batch size = 300 learning rate = 0.001 lambda = 1 RMSE = 1.11392836214

M = 16 epochs = 2000 batch size = 75 learning rate = 0.001 lambda = 1 RMSE = 1.09864567907

M = 16 epochs = 2000 batch size = 100 learning rate = 1e-06 lambda = 1 RMSE = 1.09932031615

M = 16 epochs = 2000 batch size = 200 learning rate = 1e-05 lambda = 1 RMSE = 1.09889625907

M = 16 epochs = 2000 batch size = 200 learning rate = 1e-05 lambda = 0.01 RMSE = 1.09887820289

*M = 16 epochs = 1000 batch size = 500 learning rate = 1e-05 lambda = 0.01 RMSE = 1.74109404329

M = 16 epochs = 1000 batch size = 500 learning rate = 0.01 lambda = 0.01 RMSE = 1.09998691639

M = 16 epochs = 1000 batch size = 500 learning rate = 0.005 lambda = 0.01 RMSE = 1.10177353129

M = 16 epochs = 1000 batch size = 200 learning rate = 0.005 lambda = 0.01 RMSE = 1.09928686331

M = 16 epochs = 1000 batch size = 200 learning rate = 0.005 lambda = 1e-05 RMSE = 1.09928707512

M = 16 epochs = 1000 batch size = 150 learning rate = 0.005 lambda = 0.001 RMSE = 1.14764811597

M = 32 epochs = 1000 batch size = 150 learning rate = 0.005 lambda = 0.001 RMSE = 1.22026226081

M = 32 epochs = 2000 batch size = 100 learning rate = 5e-05 lambda = 0.001 RMSE = 1.09864430866

M = 32 epochs = 2000 batch size = 100 learning rate = 0.001 lambda = 0.001 RMSE = 1.09787733619

M = 32 epochs = 5000 batch size = 100 learning rate = 0.001 lambda = 0.001 RMSE = 1.09787763324

For synthetic, we fixed the hyper parameters as follows :

M = 16, epochs = 2000, batch_size = 100,learning_rate = 0.001, lambda = 0.1