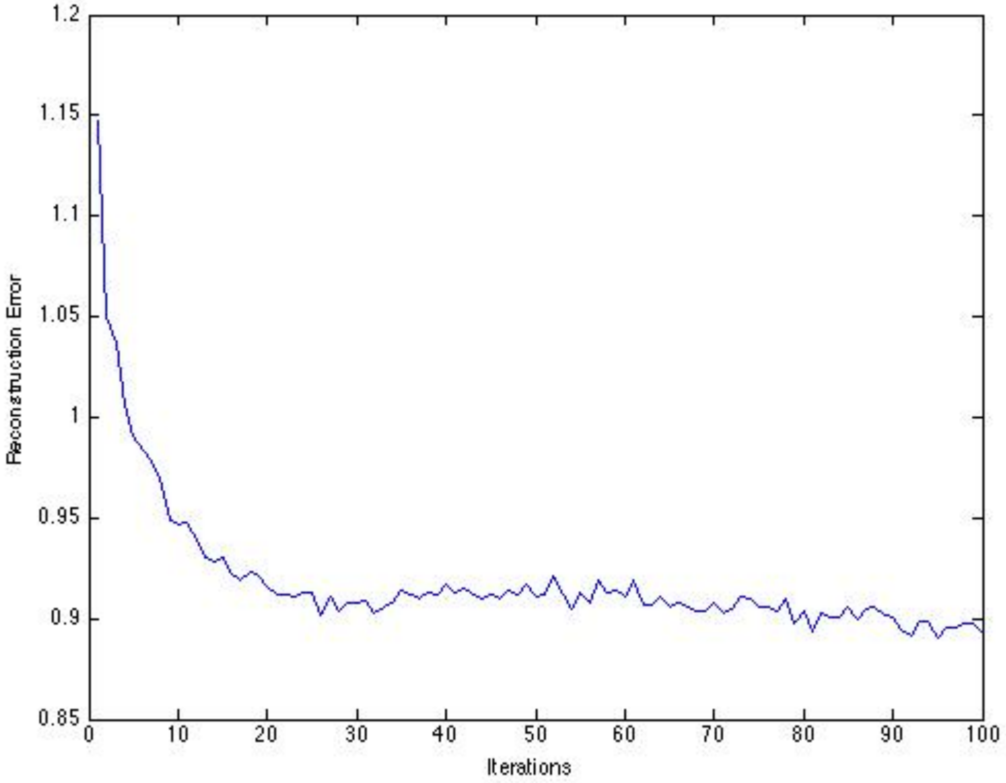
10-605 - HW 7 - Distributed SGD for Matrix Factorization on Spark

ANDREW ID: NILOYG NAME:NILOY GUPTA

EXPERIMENTS:
Set the number of workers B = 10, the number of factors F = 20, and β = 0.6. Plot the reconstruction error L_{NZSL} versus the iteration number i = 1, 2, .., 100. Explain the trend in your plot in the space provided below.

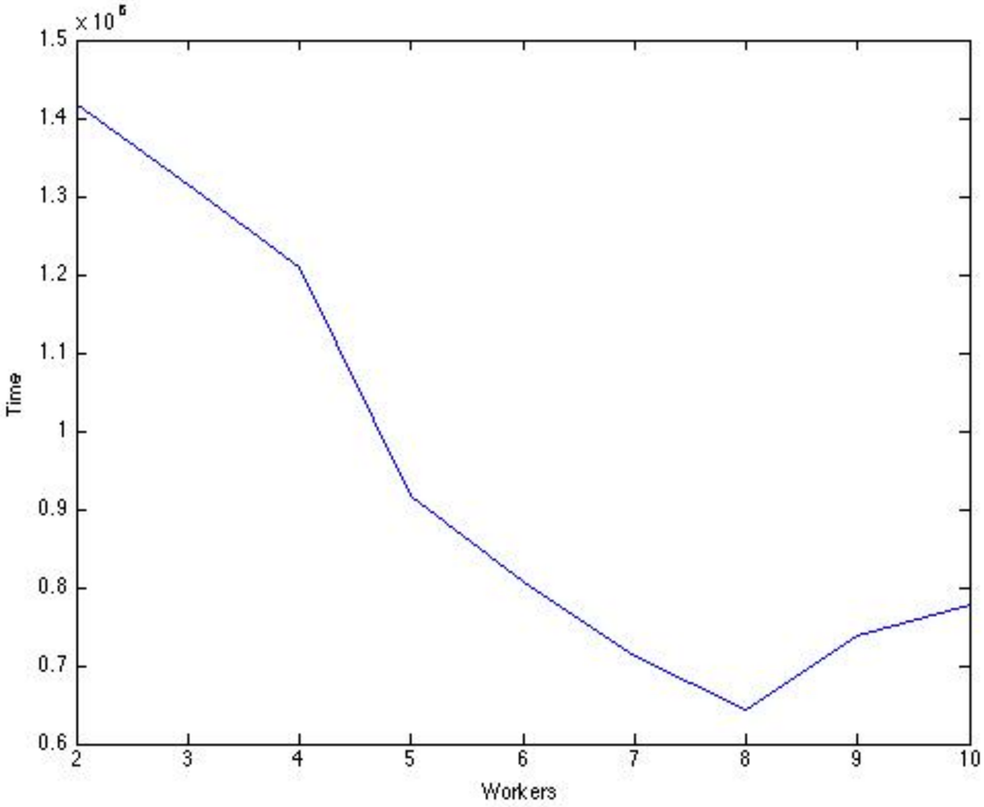
Ans:

The reconstruction error minimizses after 25 iterations and then ossilates between nearby local minima.



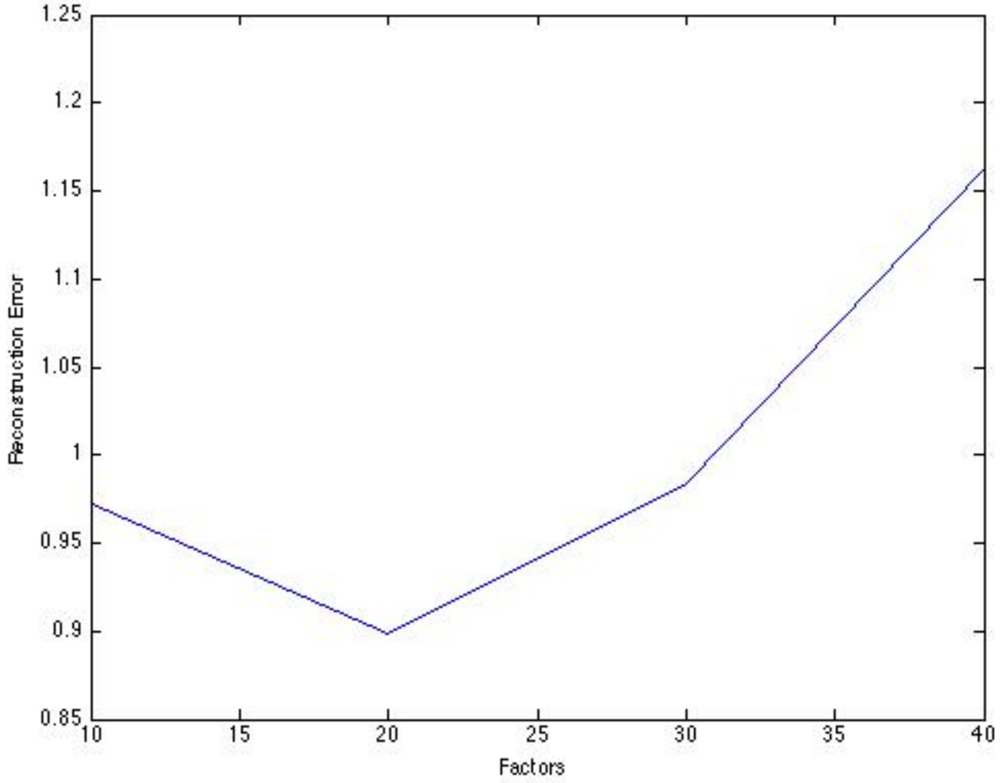
Set the number of iterations I = 30, the number of factors F = 20, and β = 0.6. Plot the runtime of your Spark code R versus number of workers B = 2, 3, .., 10 in steps of 1. Please ensure your local machine or Spark cluster can support the number of parallel workers you are requesting. Explain the trend in your plot in the space provided below.

Ans: The time decreases with number of workers but after point it increases as the network latency of communicating smaller RDDs increases.



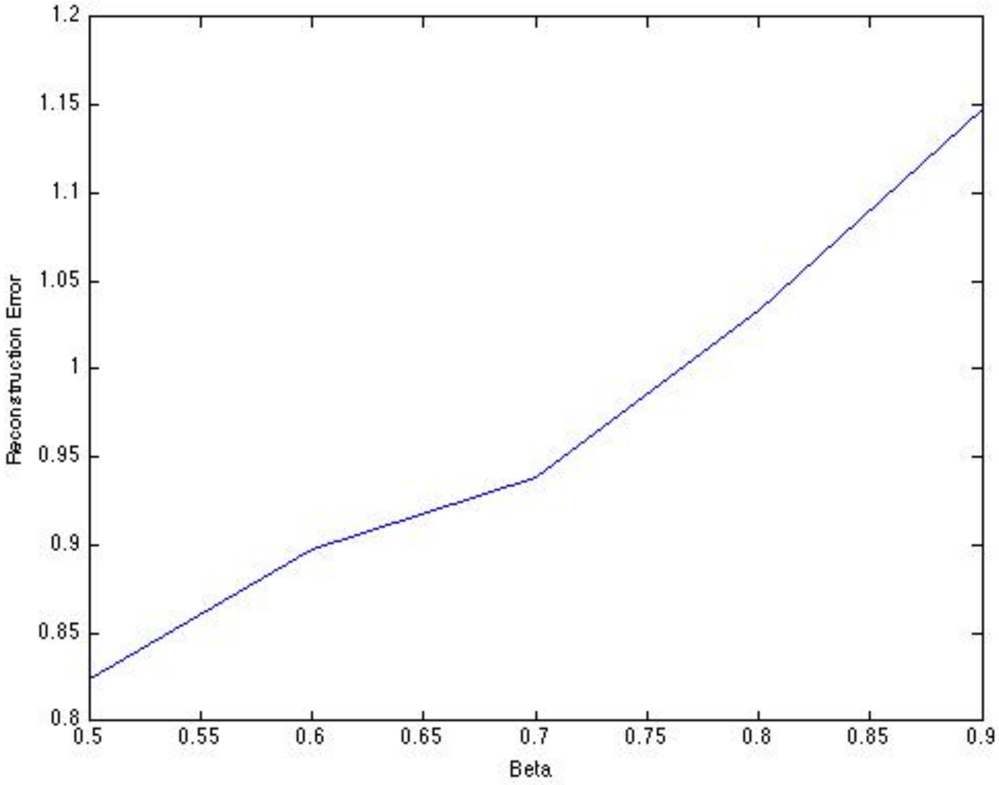
Set the number of iterations I = 30, the number of workers B = 10, and β = 0.6. Plot the reconstruction error L_{NZSL} versus number of factors F = 10, 20, .., 100 in steps of 10. Explain the trend in your plot in the space provided below.

Ans: The construction error reaches minima at for 20 factors. This is so because for fewer factors the number of latent features is too less to make any meaningful prediction. While for larger number of factors the latent feature values become close to each other and loose the distinguishing property required for making a prediction.



Set the number of workers B = 10, the number of factors F = 20, and the number of iterations I = 30. Plot the reconstruction error L_{NZSL} versus β = 0.5,0.6,..,0.9 in steps of 0.1. Explain the trend in your plot in the space provided below.

Ans: The increase in beta causes a reduction in learning rate and hence the convergence is slower and as a result, never reaches the minima as beta increases.



THEORY:

(1) Is there any advantage to using DSGD for Matrix Factorization instead of Singular Value Decomposition (SVD) which also finds a matrix decomposition that can be used in recommendation systems?

Ans:

In SVD unobserved ratings are treated as zero, while this case is handled in DSGD.

(2) Explain clearly and concisely your method (used in the code you have written) for creating strata at the beginning of every iteration of the DSGD-MF algorithm.

Ans:

We partition the matrix by the non-overlapping diagonals and store them as stratums. Each stratum is subsequently partitioned into blocks.

The pseudo code for getting the stratum id and block id given userId, movieid is as follows:

```
adjRow = maxUserId/num_workers # This gives the dimension of the block
adjCol = maxMovieId/num_workers

#First part of the equation gets the stratum id, while the second part gets the number of rotations/shifts
stratumId = (movieId/movieblock_size) - (userId/userblock_size)


# Take modulus by number of workers to convert negative ids to fit within 0 to numWorkers range

stratumId = stratumId%numWorkers
```

(3) If you were to implement two versions of DSGD-MF using MapReduce and Spark, do you think you will find a relative speedup factor between MapReduce and Spark implementations, keepingother parameters like the total number of iterations and number of workers fixed? Which implementation do you think will be faster? Why? If your answer depends on any general optimization tricks related to MapReduce or Spark that you know, please state them as well.

Ans:

The speedup performance on Spark would be greater than MapReduce provided the data can fit in memory. This is because computations in Spark are in memory while in MapReduce there is significant I/O during each iteration.

(4) Match the Spark RDD transformations to their descriptions. No explanation required.

Ans:

(1) -> (e)

(2) -> (a)

(3) -> (d)

(4) ->(b)

(5) -> (c)

(5) You need to document your code, upload it to a github repository with a README.md file at the root of the repository explaining how to run the code, and provide a link to the github repository. We will provide a Google form link on piazza after the assignment deadline where you can submit a link to your Github repository. The code will be checked for readability and documentation. Since this is an important but subjective aspect, we will provide explanation for any deducted marks.

Github Link: https://github.com/niloygupta/matrix-factorization-spark/tree/master/DSGD-Spark

(6) Answer the questions in the collaboration policy on page 1.

Did you receive any help whatsoever from anyone in solving this assignment? Yes.

- If you answered yes, give full details: Aswarth explained me the concept of mapPartitions and sc.parrallelize.

- Did you give any help whatsoever to anyone in solving this assignment? Yes

- If you answered yes, give full details: I explained the strata creation logic to Aswarth.