**Parallelizing Alternating Least Squares Algorithm on CUDA runtime Environment**

**Project Description:**

Parallelize the sequential ALS algorithm using c based CUDA and analyze the performance results of both sequential and parallel programs.

**Implementation:**

* Used **cuBLAS API:** gpublasDgemm

To calculate the dot products of any intermediate results within the program in parallel on GPU Device.

* Used **cuSOLVER Library:**

This library does the QR decomposition to solve the linear system, which is Ax = B. Where A is a dense matrix with equal number of rows and columns.Main computations needed for this decomposition are done in parallel on Device.

The code uses three steps:

Step 1: A = Q\*R by geqrf.

Step 2: B := Q^T\*B by ormqr.

Step 3: solve R\*X = B by trsm.

Finally, x = R \ Q^T\*B

* I have written a kernel named **“trans”** to find the Transpose of any 1-Dimensional matrix whenever necessary.
* Trying to use **OpenMP:**

I tried to parallelize the main part of the program where I can do large amount of computations using OpenMP.

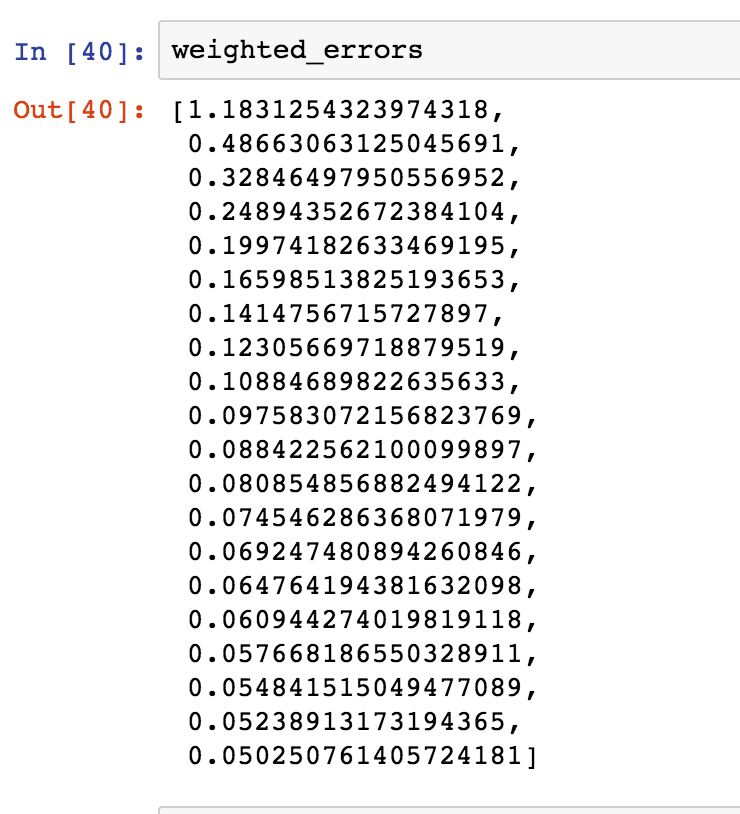
**Data: Input to the program**

An input matrix is developed with the ratings given by a set of users to a set of users. This data is collected from ratings.txt file.

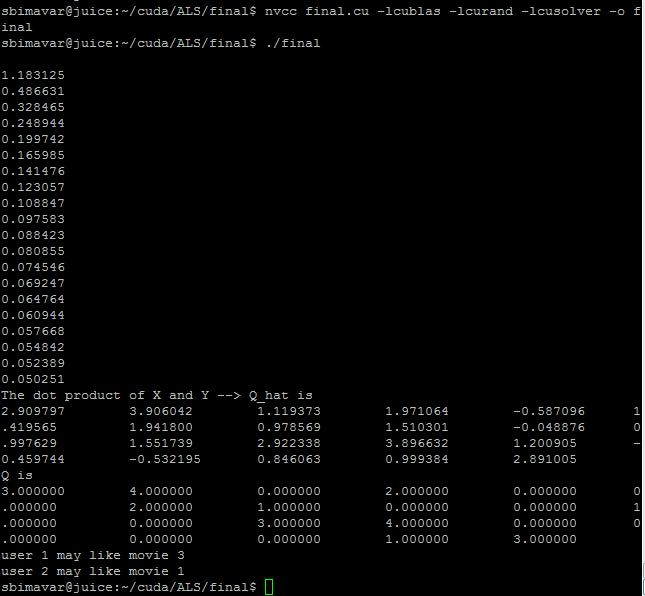
1. user-id as row-index 2) movie-id as column-index

**Performance Results:**

1. **Sequential ALS:**

****

1. **My\_ALS: Running on CUDA runtime environment**

****