INF 553 – Fall 2017
The 2nd USC Informatics Data Mining Competition

Starting Date: 11/1 2017 11:59 PM PST
End Date: 11/22 2017 11:59 PM PST
Competition Overview
You will build this on your assignment 3 to implement a recommendation system on MovieLens dataset. You can use any method (including SparkMLlib) to improve your rating prediction. You can use Scala or Python for this competition.
Write your own code!

Datasets
The MovieLens datasets can be found in the following link:
https://grouplens.org/datasets/movielens/

You will use the datasets ml-latest-small.zip for the competition. You can use other data sources (e.g., IMDB) to improve the performance of your recommendation system.

You will also use the testing file *testing_small.csv*, separate the test data from the training data in *ml-latest-small.zip*. Your goal is to predict the ratings of every <userId> and <movieId> combination in the test files. You CANNOT use the ratings in the testing datasets to train your recommendation system. Specifically, you should first extract training data from the *ratings.csv file* downloaded from MovieLenses using the testing data. Then by using the training data,
You will need to **predict** rate for movies in the testing datasets. You can use the testing data as your ground truth to evaluate the accuracy of your recommendation system. These are exactly the same steps as the ones in your assignment 3.

Example: Assuming ratings.csv contains 1 million records and the testing_small.csv contains two records: (12345, 2, 3) and (12345, 13, 4). You will need to first remove the ratings of user ID 12345 on movie IDs 2 and 13 from ratings.csv. You will then use the remaining records in the ratings.csv to train a recommendation system (1 million – 2 records). Finally, given the user ID 12345 and movie IDs 2 and 13, your system should produce rating predictions as close as 3 and 4, respectively.

Result format:
1, **Save the predication results in a text file**. The result is ordered by <userId> and <movieId> in ascending order.

2, **Print the accuracy information** in terminal, and **copy this value** in your description file.

```
>=0 and <1:1530
>=1 and <2:464
>=2 and <3:106
>=3 and <4:17
>=4:1
Outliers:0
```

```
RMSE = 1.0131646962216851
```

Description File

Please include the following content in your description file:
1. The Spark version and Python version
2. A short description about how to run your program for both tasks:
Order of the arguments should be followed as arg0 - training file, arg1 - testing file.

For example, to run jar package, you should to write the commend as:

```
./bin/spark-submit --class Priyambada_Jain_Task1
/usr/local/spark/Implementation/Priyambada_Jain_Task1.jar
/usr/local/spark/data/ml-latest-small/ratings.csv
/usr/local/spark/data/Testing_data1.csv
```

3. The accuracy of your system (the format is described above)
4. A short description about your method (less than 200 words)

Submission Details

Your submission must be a .zip file with name: *<Firstname>_<Lastname>_comp.zip*
Please include all the files as following:
1. A description file: *<Firstname>_<Lastname>_desription.txt (or pdf)...*
2. A Scala or Python script for your system: *<Firstname>_<Lastname>_ comp.scala* or *<Firstname>_<Lastname>_ comp.py*
3. If you use Scala, please submit the jar package as well and name it as *<Firstname>_<Lastname>_ comp.jar*
4. One result file: <Firstname>_<Lastname>_result.txt

Competition Criteria:

1. **If your programs cannot run with the commands you provide,** we will not consider your submission.
2. **If the files generated are not sorted based on the specifications,** we will not consider your submission.
3. **If your program generates more than one file,** we will not consider your submission.
4. **If your prediction result files miss any records**, we will not consider your submission.
5. **If you don't provide the source code,** we will not consider your submission.
6. **If you don't describe how to run your code, which Spark/Python version you used, or the accuracy result,** we will not consider your submission.
7. If your code does not finish the prediction in 2 minutes, we will not consider your submission.
8. You can submit your code and result anytime you want, but **we will grade only your latest submission before every Wednesday 11:59pm PST.** We will announce the leaderboard on the following Fridays.

Ranking Criteria:
We will rank the competition based on your **RMSE accuracy**. After the last day of the competition, the submission having the highest accuracy will receive a 4% bonus on their final grade. The 2nd highest accuracy will receive a 3% bonus on their final grade. The 3rd highest accuracy will receive a 2% bonus on their final grade. Others will receive a 1% bonus on their final grade. In addition, you need to beat Priyambada's system to be able to receive the bonus. Priyambada will continuously improve her system and will announce her accuracy every Friday along with the rankings for the previous week.

Here are the submission dates explicitly: **11/1, 11/8, 11/15 and finally 11/22.**