Q2.3 Questions

1. Compute the prediction accuracy (MAE on ml-100k/u1.test) of (Eq. 3). Report the result. Compute the difference between the Adjusted Cosine similarity and the baseline (cosine baseline). Is the prediction accuracy better or worst than the baseline (Eq. 5 from Milestone 1)? (Use a baseline MAE of 0.7669)

   MAE for cosine similarity: 0.7477653398324886

   Difference between adjusted cosine similarity and baseline: -0.01913466016751142

   The adjusted cosine based similarity method has smaller MAE, and hence better accuracy than the baseline.

2. Implement the Jaccard Coefficient[2]. Provide the mathematical formulation of your similarity metric in your report. Compute the prediction accuracy and report the result. Compute the difference between the Jaccard Coefficient and the Adjusted Cosine similarity (Eq. 1, jaccard cosine) Is the Jaccard Coefficient better or worst than Adjusted Cosine similarity?

   Formulation of Jaccard Coefficient. Let $s_{u,v}$ denote the Jaccard Coefficient between user u and user v. Let I(u), I(v) denote the set of items rated by user u,v.

   $S_{u,v} = | I(u) \cap I(v) | / | I(u) \cup I(v) | = | I(u) \cap I(v) | / (| I(u)| + |I(v)| - | I(u) \cap I(v) |)$

   MAE for Jaccard: 0.7622501056973406

   Diff between Jaccard and Adjusted Cosine: 0.014484765864852034

   Jaccard has larger MAE hence is worse than adjusted cosine similarity.

3. How many $s_{u,v}$ computations, as a function of the size of U (the set of users), have to be performed? How many does that represent for the 'ml-100k' dataset?

   As an upper bound, for an any user u, the similarity between u and any other user (including user u himself) needs to be computed. So $|U|^2$ many $s_{u,v}$ need to be computed. For ml-100k dataset, we have 943 users, so 943^2=889249 many $s_{u,v}$ need to be computed.

4. Compute the minimum number of multiplications required for each $s_{u,v}$ on the ml-100k/u1.base.[3] (Tip: This is the number of common items, |I(u) \ I(v)|, between u and v.) What are the min, max, average, and standard deviation for all similarities computed? Report those in a table.

min	0
max	685
average	12.205087663860263
standard deviation	18.1751839909868

5. How much memory, as a function of the size of U, is required to store all $s_{u,v}$, both zero and non-zero values? How many bytes are needed to store only the non-zero $s_{u,v}$ on the 'ml-100k' dataset, assuming each non-zero $s_{u,v}$ is stored as a double (64-bit floating point value)?

   $8|U|^2$ bytes are required to store all $s_{u,v}$ both zero and non-zero, because each $s_{u,v}$ needs 8 bytes and there are $|U|^2$ of them.
   6592280 bytes are needed to store only the non-zero $s_{u,v}$ on the 'ml-100k' dataset.

6. Measure the time required for computing predictions (with 1 Spark ex- ecutor), including computing the similarities $s_{u,v}$. (Tip: If you compute the similarities in batch prior to predictions, include the time for com- puting them all. If you compute similarities on a by-need basis, possibly with caching, include the time for all those that were actually computed.) Provide the min, max, average, and standard-deviation over five mea- surements. Discuss in your report whether the average is higher than the previous methods you measured in Milestone 1 (Q.3.1.5)? If this is so, discuss why.

Stats for time required for predictions	Values(microsecond)
Min	4.8059733958E7 (48059733.958)
Max	5.0527877166E7 (50527877.166)
Average	4.9112934483E7 (49112934.483)
Standard Deviation	940977.7648544562

   Yes, the average is higher than the previous methods in Milestone1. There are two main sources of time increasing for my M2 implementation: (1) additional time is required to compute nonzero similarities. (2) to implement the user-specific weighted-sum (equation 2 in project spec) , additional time is required to join similarities and deviations together.

7. Measure only the time for computing similarities (with 1 Spark execu- tor). (Tip: If you compute the similarities in batch prior to predictions, include the time for computing them all. If you compute similarities on a by-need basis, possibly with caching, include the time for all those that were actually computed.) Provide the min, max, average and standard- deviation over ten five measurements. What is the average time per $s_{u,v}$ in microseconds? On average, what is the ratio between the computation of similarities and the total time required to make predictions? Are the computation of similarities significant for predictions? (Tip: To lower your total running time, you can combine this measurement in with that of the previous question in the same runs.)

	Value (microsecond)
Min	1.3166184333E7 (13166184.333)
Max	1.520421425E7 (1520421.425)
Average	1.36432142746E7 (13643214.2746)
Standard Deviation	782493.8542634158

- The average time per $s_{u,v}$ is 16.556595623486867 microsecond.
- The ratio between the computation of similarities and the total time required to make predictions is 0.277792691848264.
- The computations of similarities in my implementation is relatively significant as it occupies more than ¼ of the total time.

Q3.2 Questions

1. What is the impact of varying k on the prediction accuracy? Provide the MAE (on ml-100k/u1.test) for k = 10, 30, 50, 100, 200, 300, 400, 800, 943. What is the lowest k such that the MAE is lower than for the baseline method (Eq. 5 of Milestone 1)? How much lower? (Use a baseline MAE of 0.7669, and compute lowestk baseline)

   Impact of varying k on the prediction accuracy: When k is relatively small (< 300) it seems that the accuracy increases as k increases. When k is relatively large (>=300) it seems that the accuracy decreases slightly as k increases (potentially overfitting).

k	MAE
10	0.8407036862423928
30	0.7914221792247477
50	0.7749407796360606
100	0.7561353222065882
200	0.7484528977469215
300	0.7469140388149915
400	0.7471389103638729
800	0.7475383223779437
943	0.7477653398324886

   Lowest K With Better Mae Than Baseline: 100,

   Lowest K Mae Minus Baseline Mae: -0.010764677793411837862423928, i.e. KNN with K=100's MAE is 0.010764677793411837862423928 lower than the baseline MAE.

2. What is the minimum number of bytes required, as a function of the size of U, to store only the k nearest similarity values for all possible users u, i.e. top k $s_{u,v}$ for every u, for all previous values of k (with the ml-100k dataset)? Assume an ideal implementation that stored only similiarity values with a double (64-bit floating point value) and did not use extra memory for the containing data structures (this represents a lower bound on memory usage). Provide the formula in the report. Compute the number of bytes for each value of k in your code.

   |U|*k similarities in total, each similarity needs 8 bytes, so
   minimum number of bytes required = 8k*|U|

k	Minimum number of bytes
10	75440
30	226320
50	377200
100	754400
200	1508800
300	2263200
400	3017600
800	6035200
943	7113992

3. Provide the RAM available in your laptop. Given the lowest k you have provided in Q.3.1.1, what is the maximum number of users you could store in RAM? Only count the similarity values, and assume you were storing values in a simple sparse matrix

implementation that used 3x the memory than what you have computed in the previous section (2 64-bit integers for indices and 1 double for similarity values).

RAM size 8GB = 8 *10^9 bytes

Lowest k better than baseline = 100. 100 similarities per user. Each similarity 24 bytes. 2400 bytes per user.

numUserInRAM = 8 *10^9 / 2400 = 3333333

4. Does varying k has an impact on the number of similarity values ($s_{u,v}$) to compute, to obtain the exact k nearest neighbours? If so, which? Provide the answer in your report.

   No. No matter what k is, to get the top k similarities for each user, we need to compute all the similarities for each user, as any similarity can be a top k similarity.

5. Report your personal top 5 recommendations with the neighbourhood predictor (Eq. 3) with k = 30 and k = 300. How much do they differ between the two dfferent values of k? How much do they differ from those of the previous Milestone?

   "Top5WithK=30" : [[26, "Brothers McMullen", 5.0], [39, "Strange Days (1995)", 5.0], [41, "Billy Madison (1995)", 5.0], [57, "Priest (1994)", 5.0], [114, "Wallace & Gromit: The Best of Aardman Animation (1996)", 5.0]],

   "Top5WithK=300" : [[1111, "Double Happiness (1994)", 5.0], [1189, "Prefontaine (1997)", 5.0], [1233, "Nénette et Boni (1996)", 5.0], [1242, "Old Lady Who Walked in the Sea", 5.0], [1302, "Late Bloomers (1996)", 5.0]]

   The recommendations generated by knn with k=30 is completely different from the recommendations generated by knn with k=300.

   K=30's recommendations are completely different from previous milestone's recommendations. K=300's recommendations have only one overlapping with previous milestone's recommendation [1189, "Prefontaine (1997)", 5.0].

   Both k=30 and k=300 are completely different from the recommendations generated by previous Milestone's recommendations for the bonus question.