Mark Gluzman

Mg3627

Homework 4

ELEN 4903

Due 4/15

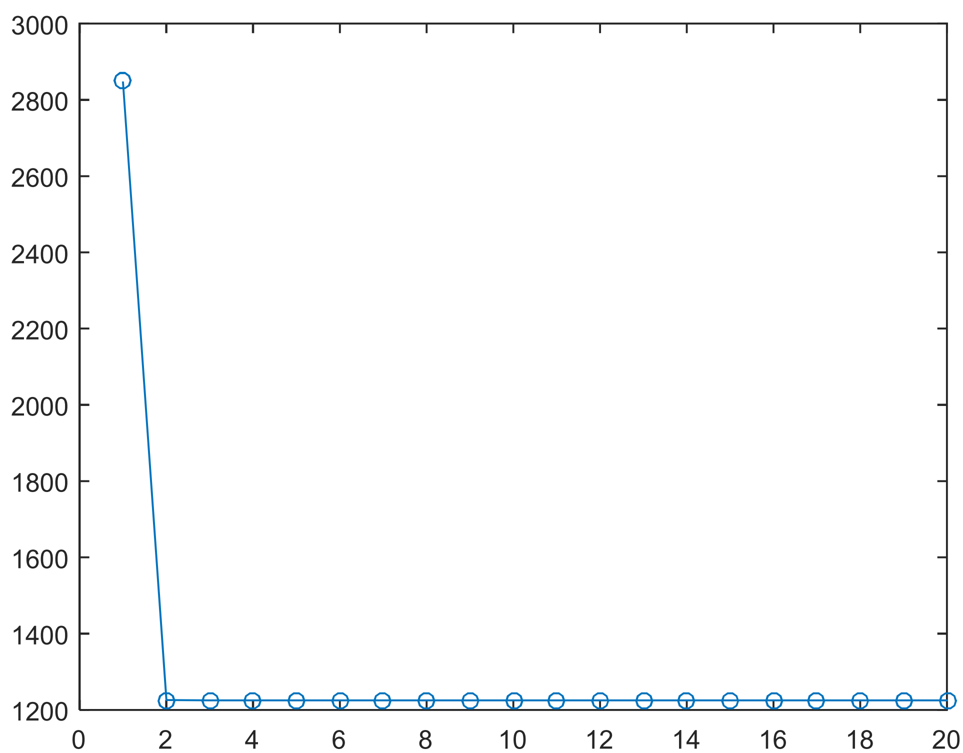
Problem 1 (K-means) – 35 points

Implement the K-means algorithm discussed in class. Generate 500 observations from a mixture of three Gaussians on $\mathbb{R}^2$ with mixing weights $\pi = [0.2, 0.5, 0.3]$ and means μ and covariances Σ,

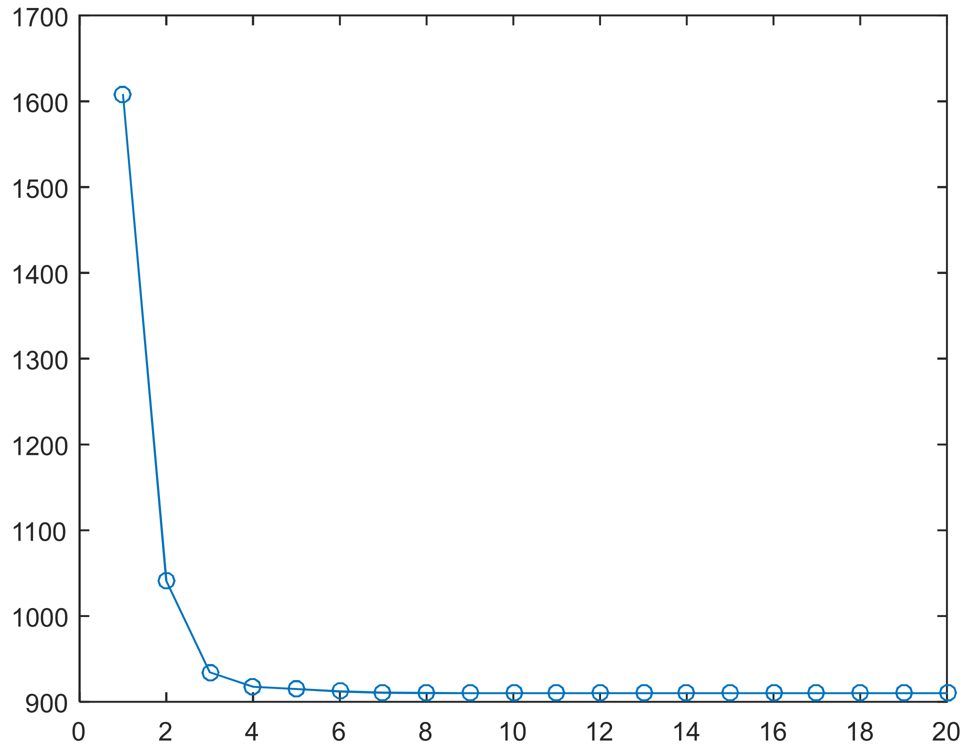
$$\mu_1 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \Sigma_1 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \qquad \mu_2 = \begin{bmatrix} 3 \\ 0 \end{bmatrix}, \Sigma_2 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \qquad \mu_3 = \begin{bmatrix} 0 \\ 3 \end{bmatrix}, \Sigma_3 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}.$$

1. For $K = 2, 3, 4, 5$, plot the value of the K-means objective function per iteration for 20 iterations (the algorithm may converge before that).

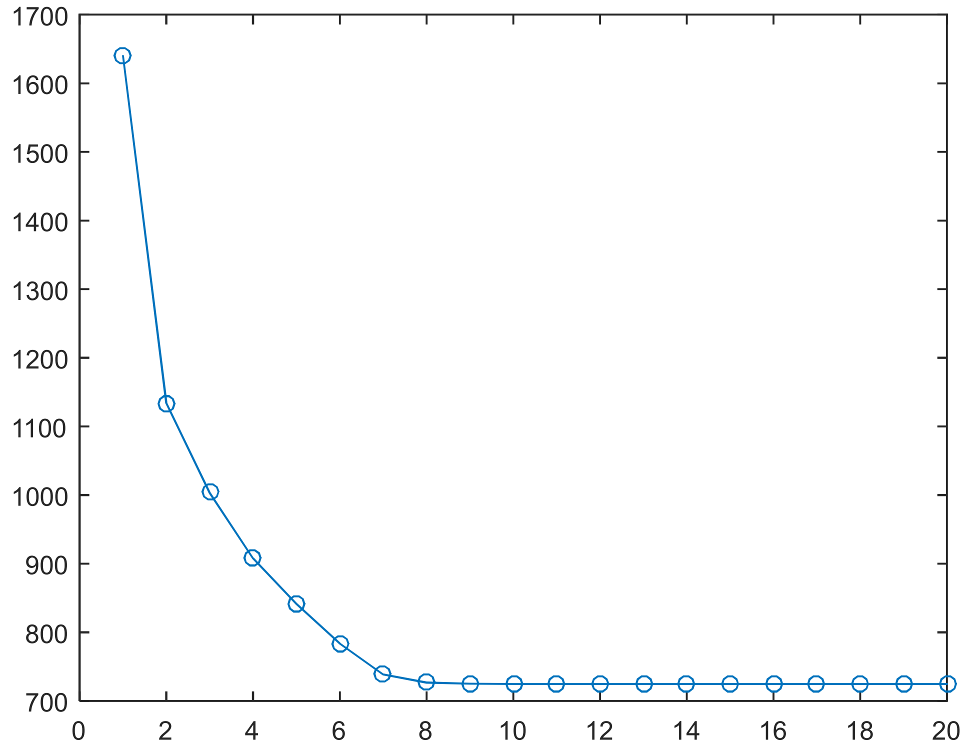
K=2



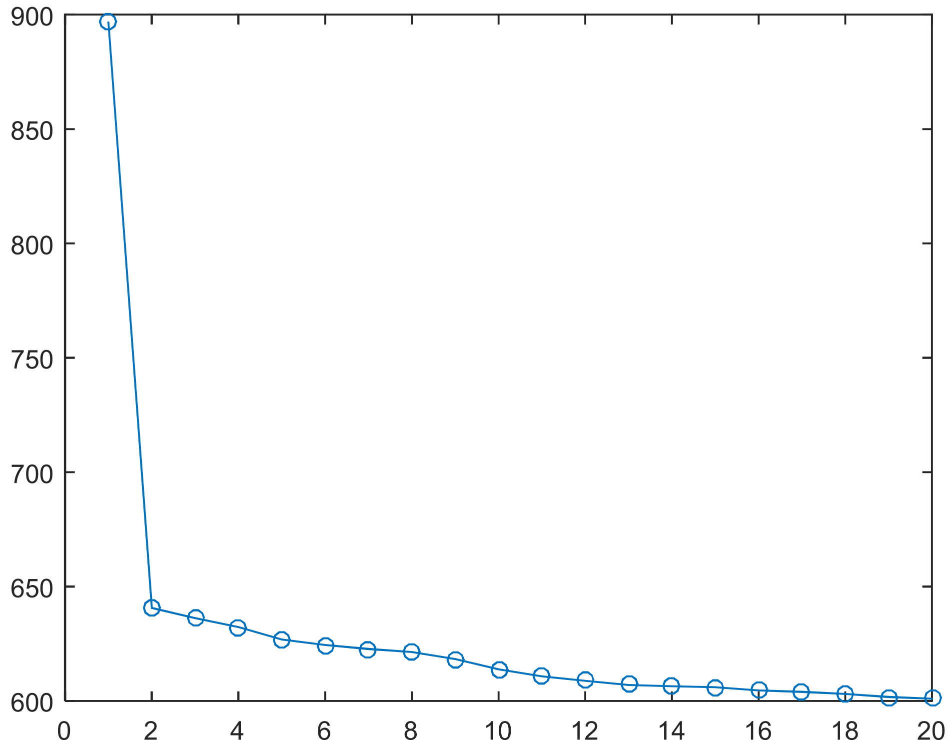
K=3


1700
1600
1500
1400
1300
1200
1100
1000
900
0
2
4
6
8
10
12
14
16
18
20

K=4

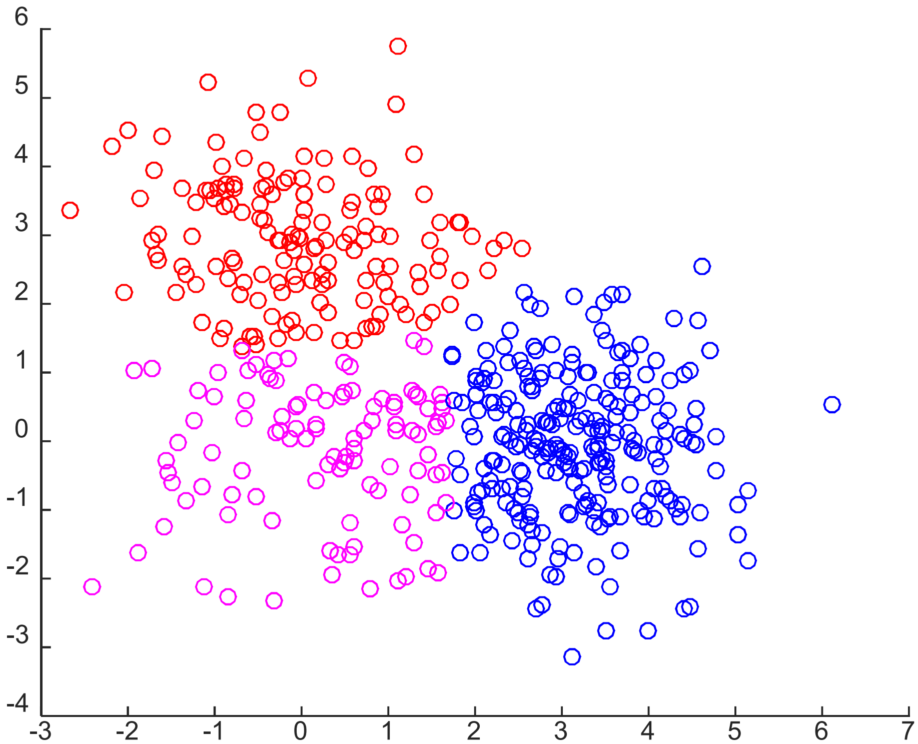

1700
1600
1500
1400
1300
1200
1100
1000
900
800
700
0
2
4
6
8
10
12
14
16
18
20

K=5

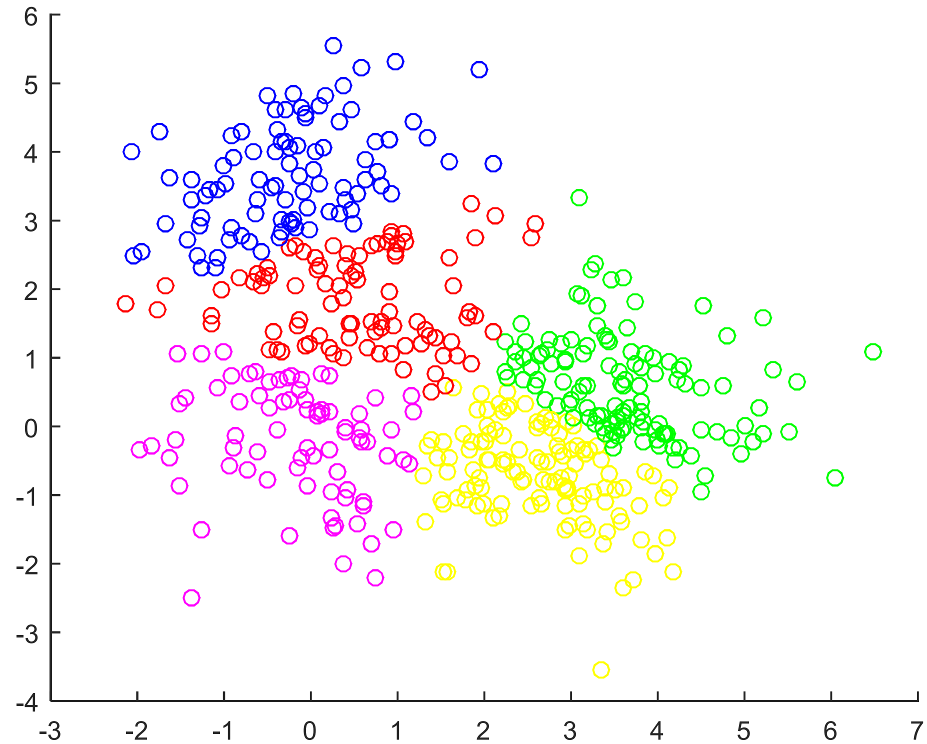


2. For $K = 3, 5$, plot the 500 data points and indicate the cluster of each for the final iteration by marking it with a color or a symbol.

K=3



K=5



Problem 2 (Matrix factorization) –85 points

In this problem, you will implement the MAP inference algorithm for the matrix completion problem discussed in class. As a reminder, for users $u \in \mathbb{R}^d$ and movies $v \in \mathbb{R}^d$, we have

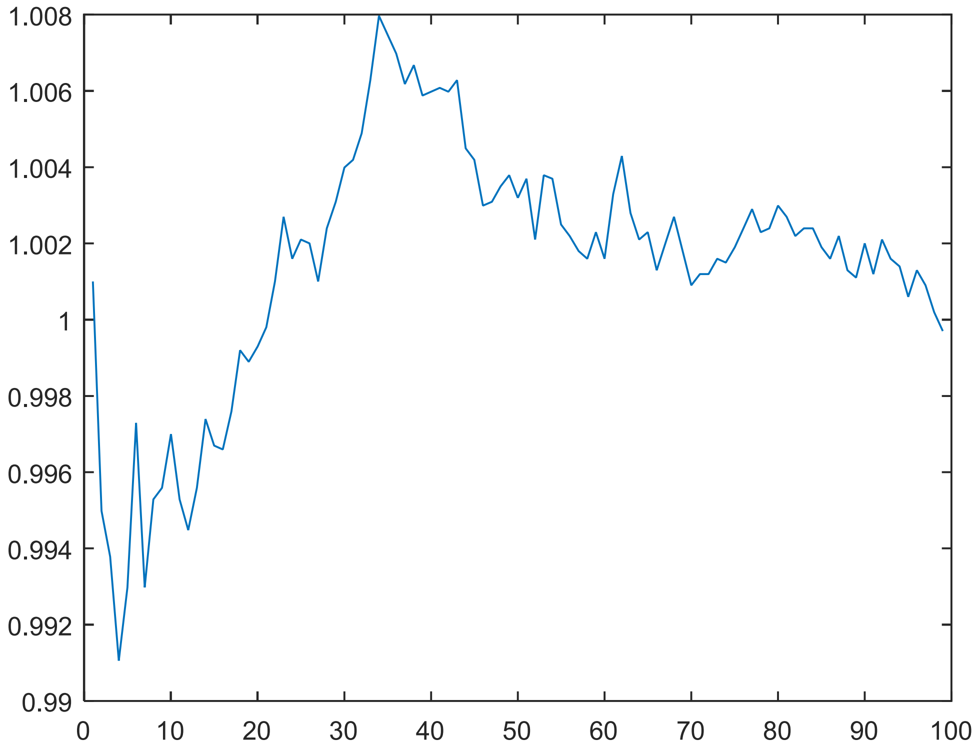
$$u_i \sim N(0, \lambda^{-1}I), \quad i = 1, \ldots, N_1, \qquad v_j \sim N(0, \lambda^{-1}I), \quad j = 1, \ldots, N_2.$$

We are given an $N_1 \times N_2$ matrix M with missing values. Given the set $\Omega = \{(i,j) : M_{ij} \text{ is measured}\}$, for each $(i,j) \in \Omega$ we model $M_{ij} \sim N(u_i^T v_j, \sigma^2)$.

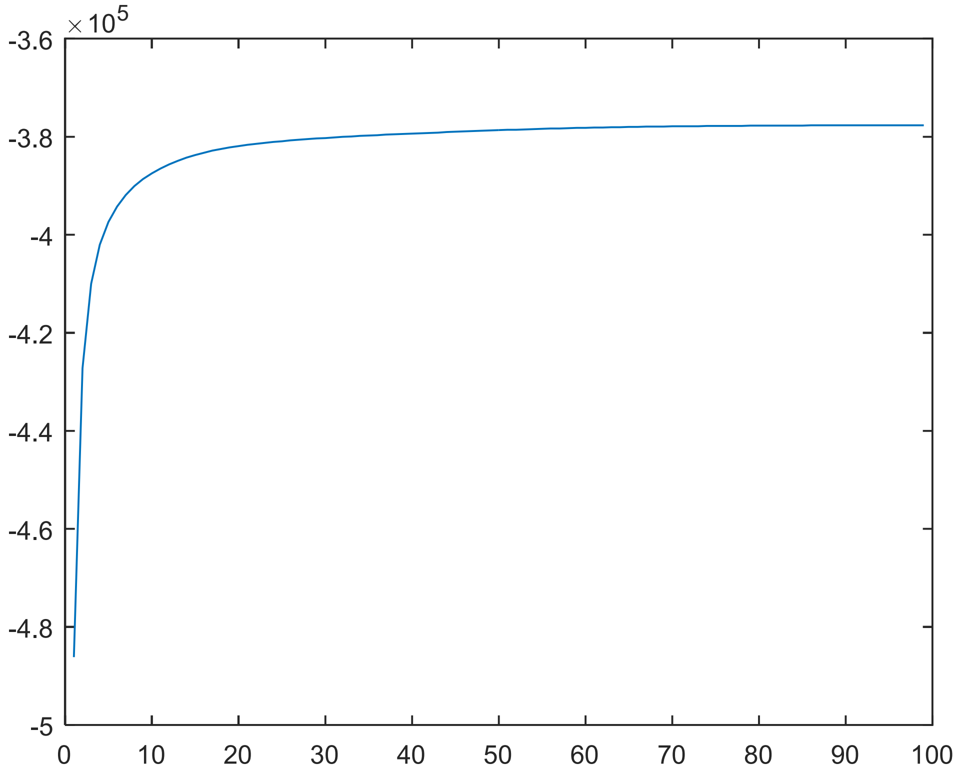
Run your code on the user-movie ratings dataset provided on Courseworks and the course website. For your algorithm, set $\sigma^2 = 0.25$, $d = 10$ and $\lambda = 10$. Train the model on the larger training set for 100 iterations. For each user-movie pair in the test set, predict the rating by mapping the relevant dot product to the closest integer from 1 to 5. Since the equations are in the slides, there's no need to re-derive it.

We get 2856 right predictions for test data.

1. Plot the RMSE of your predictions on the held out test set provided for iteration number 2 to 100.



2. On a separate plot, show the log joint likelihood for iterations 2 to 100.



3. After 100 iterations, pick three reasonably well-known movies from the list provided and for each movie find the 5 closest movies according to Euclidean distance using their respective locations v_j. List the query movie, the five nearest movies and their distances. A mapping from index to movie is provided with the data.

Star Trek: First Contact (1996) (j=222)

j	name	norm
265	Hunt for Red October, The (1990)	0.720150181219108
257	Men in Black (1997)	0.762281766781272
96	Terminator 2: Judgment Day (1991)	0.770290926327502
751	Tomorrow Never Dies (1997)	0.800504226487610
227	Star Trek VI: The Undiscovered Country (1991)	0.812414918469992

Schindler's List (1993) (j=318)

j	name	norm

193	Right Stuff, The (1983)	0.721615915802672
527	Gandhi (1982)	0.728778487306746
496	It's a Wonderful Life (1946)	0.745928165011728
648	Quiet Man, The (1952)	0.765626906619497
191	Amadeus (1984)	0.788624310444579

Men in Black (1997) (j=257)

j	name	norm
751	Tomorrow Never Dies (1997)	0.483932631781324
210	Indiana Jones and the Last Crusade (1989)	0.582607768141017
28	Apollo 13 (1995)	0.670110739017594
73	Maverick (1994)	0.671661119087559
204	Back to the Future (1985)	0.699454320524803

4. After 100 iterations, perform K-means on the vectors $u_1, \ldots, u_{N_1}$ learned by your algorithm. Set $K = 20$, which is an arbitrary number. The centroids can be interpreted as personality types (as far as movies are concerned).

   - Pick the 5 centroids corresponding to the 5 clusters that have the most data. For each cluster selected, give the number of users allocated to that cluster.
   - For each of these 5 centroids, list the 10 movies with the largest (most positive) dot product with that centroid. Also give the value of the dot product next to the movie.

1)K=5

Users=86

Movies, product:

```
127    4.811536
318    4.725766
483    4.699937
 64    4.671357
357    4.663574
187    4.637208
511    4.615301
 98    4.594775
603    4.565392
 50    4.554406
```

2)K=15

Users=75

Movies, product:

285	4.550719
100	4.544112
474	4.502968
59	4.502309
124	4.472952
60	4.457969
408	4.448727
9	4.414366
654	4.405369
242	4.403207

3)K=7

Users=72

Movies, product:

169	4.454813
963	4.452296
663	4.414917
285	4.384483
64	4.367692
114	4.356898
56	4.348003
1019	4.32295
900	4.321017
313	4.320852

4)K=2

Users=63

Movies, product:

272	4.541741
114	4.440276
315	4.43941
316	4.415547

64	4.390447
173	4.377597
169	4.356413
12	4.348799
515	4.281796
50	4.223

5)K=1

Users=62

Movies, product:

251	4.427672
408	4.41937
316	4.412835
315	4.358765
170	4.354415
169	4.350802
272	4.305891
114	4.296356
87	4.29002
318	4.28138

5. After 100 iterations, perform K-means on the vectors $v_1, \ldots, v_{N_2}$ learned by your algorithm. Set $K = 20$, which is an arbitrary number.
   - Pick the 5 centroids corresponding to the 5 clusters that have the most data. For each cluster selected, give the number of movies allocated to that cluster.
   - For each of these 5 centroids, list the 10 movies with the smallest Euclidean distance to that centroid. Also give the value of the Euclidean distance next to the movie.

1) K=20 Movies=181
   Movies, norms:

685	0.580977
102	0.836419
155	0.884509
855	0.909295
915	0.954312
656	0.973071
824	0.976256
445	0.979414

594	0.986478
26	0.98777

2) K=10 Movies-128
   Movies, norms:

832	0.83329
275	0.904779
102	0.914897
685	0.941197
181	0.953546
205	0.962934
633	0.978939
646	0.981474
827	0.991828
454	1.037712

3) K=7 Movies= 126
   Movies, norms:

685	1.208627
181	1.324258
445	1.394391
102	1.402213
302	1.432047
205	1.48524
832	1.496041
700	1.496434
926	1.496472
827	1.503925

4) K=13 Movies=106
   Movies, norms:

874	0.542698
860	0.569779
571	0.572453
227	0.605792
829	0.605801
273	0.616113
583	0.621318
573	0.625397
345	0.649258

360	0.660692

5) K=11 Movies 105
   Movies, norms:

106	0.534191
908	0.574216
378	0.577228
844	0.581871
70	0.619979
616	0.643111
924	0.645678
786	0.648098
411	0.676988
579	0.68727