Exercise 10.1 (June 1, 2020)

-B9TB1707

Question:

• Ratings of 20 songs are available (rating1.txt by 5 people, rating2.txt by 15 people)
• Suppose a new (i.e., 16th) person gives ratings for three songs: song1=4, song3=2, song7=3
• Estimate ratings by this person for other songs
• The following steps should be performed for each rating date (rating1.txt and rating2.txt)
• First, find a rank-3 approximation of R, i.e., obtain 5x3 P and 3x20 S
• Second, find p16 that satisfies the following equations using S:
• Finally, calculate prediction of ratings by
• True ratings of R16 are:

4 3 2 2 3 3 3 2 3 1 2 3 2 2 3 4 3 3 3 3

Solution:

My code for the solution is as follows:

CAPS_10_B9TB1707_10.1.m

```
1  R=load('D:\Eric\My CAPS homework\rating1.txt');
2  [U1,W1,V1]=svd(R.R);
3  w1=W1(1:3,1:3);
4  v1=(V1')(1:3,:);
5  j1=w1*v1;
6  K1=[j1(:,1) j1(:,3) j1(:,7)];
7  input=[4 2 3];
8  pi1=input/K1;
9  Result_1=pi1*j1;
10 Result_1=round(Result_1);
11 Real=[4 3 2 2 3 3 3 2 3 1 2 3 2 2 3 4 3 3 3 3]
12 disp(" These are the predictions bsed on 5 people");
13 Result_1
14 Error_1=Real-Result_1
15 R=load('D:\Eric\My CAPS homework\rating2.txt');
16 [U,W,V]=svd(R.R);
17 w=W(1:3,1:3);
18 v=(V')(1:3,:);
19 j2=w*v;
20 K2=[j2(:,1) j2(:,3) j2(:,7)];
21 pi2=input/K;
22 Result_2=pi2*j2;
23 Result_2=round(Result_2);
24 disp(" These are the predictions based on 15 people");
25 Result_2
26 Error_2=Real-Res
```

And the output is as follows:

```
Command Window
>> Real =

   4   3   2   2   3   3   3   2   3   1   2   3   2   2   3   4   3   3   3   3

 These are the predictions bsed on 5 people
Result_1 =

   4   4   2   4   4   4   3   3   3   3   3   3   2   2   3   3   4   4   3   3

Error_1 =

   0  -1   0  -2  -1  -1   0  -1   0  -2  -1   0   0   0   0   1  -1  -1   0   0

 These are the predictions based on 15 people
Result_2 =

   4   3   2   1   2   2   3   1   3   0   1   3   1   1   2   3   4   4   3   3

Error_2 =

   0   0   0   1   1   1   0   1   0   1   1   0   1   1   1   1  -1  -1   0   0
```

How it works:

1. Line 1 loads the txt file (ratings1.txt) holding the ratings of the songs of 5 people.
2. Line two performs the singular value decomposition using the svd() function and stores the output into U1, W1, V1.
3. Line 3 reduces the diagonal matrix into rank 3 and stores it in w1.
4. Line 4 transpose and reduces V1 into a rank 3 matrix and stores it in v1.
5. Line 5 stores the product of w1 and v1.
6. Line 6 stores the columns we need in K1 (The songs for which we have the ratings of the new person).
7. Line 7 stores the ratings of the new person in array input.
8. Line 8 calculates and stores the pseudo-inverse.
9. Line 9 calculates the result and line 10 rounds in to the nearest integer.
10. Line 11 stores the actual ratings given by the new person.
11. Line 12 describes the output, Line 13 prints our predictions.
12. Line 14 prints the error in our calculations but subtracting our result from the real values.
13. Line 15-26 repeat the process for ratings2.txt.

Conclusion:

Thus I have predicted the ratings a person might give to songs from ratings given by other people and data of 3 of the new person's rating. This method employs singular value decomposition to compute the predictions. This problem was very interesting for me, because I found out that Netflix uses a similar algorithm to recommend movies (more specifically a SVD inspired algorithm called SVD++ and another technique called the restricted Boltzmann machine). I also decided to leave zeroes in the final result after the rounding to the nearest integer to more accurately reflect the error generated by our numerical techniques.