Lab: Recommendation systems

In this lab you will work through some examples of the computations needed to create a basic recommendation system and then go on to make a recommendation system based on a sample of survey data. The activities below are written to be completed using Octave, which you can access at `octave-online.net`. To become familiar with basic matrix and vector commands in Octave, it is recommended that you complete the introductory lab before this activity.

To complete the entire lab, you will need to sign in to Octave Online with Google or by using an email address. Once you sign in (under the Menu button), you will need to upload the following files: `distance_sim.m` and `cosine_sim.m`, as well as the survey data `SURVEYDATA.txt`. The button to upload files is on the left-hand side of the screen. Once uploaded, you will see the list of files, and by clicking on each file name individually you can read its contents.

1. In the presentation, you saw two ways of comparing movie ratings data: user-to-user and movie-to-movie. The matrix of data from the presentation (3 users ranking 2 movies) is given again below:

$$A = \begin{bmatrix} 1 & 5 & 5 \\ 1 & 4 & 5 \end{bmatrix}.$$

Enter the matrix `A` into the Command Prompt, and then complete the following to check for your understanding:

(a) What information does the first column $\begin{bmatrix} 1 \\ 1 \end{bmatrix}$ contain? Type `A(:,1)` in the Command Prompt to access this information in Octave.

(b) What information does the first row $\begin{bmatrix} 1 & 5 & 5 \end{bmatrix}$ contain? Enter `A(1,:)` in the Command Prompt.

(c) What does `norm(A(:,1))` compute? How about `norm(A(:,1)-A(:,2))`?

User-to-user comparisons

2. Complete the table of **distance measures** between all the user vectors from Problem 1. Remember that the formula to find the distance between two vectors $\vec{v}, \vec{w}$ is $\|\vec{v} - \vec{w}\|$. You can make this calculation in Octave by typing

```
norm(A(:,i)-A(:,j))
```

with `i` and `j` replaced by particular numbers.

Distance measures

	User 1	User 2	User 3
User 1	0	5	5.6569
User 2	5	0	
User 3	5.6569		

3. Do you notice anything about the diagonal entries of the table in Problem 2? Explain.

4. Do you notice any other symmetries in the table in Problem 2? Is this a coincidence, or can you think of why some entries are the same? Are there any computations that we can skip to save time?

5. Make the heat map for the distance measure. This step requires that you have uploaded the file `distance_sim.m`. To call on the script, type the filename `distance_sim` (leave off the file extension) in the Octave Command Prompt.
If you are curious about the script, you can see its contents on the left-side of your screen.

6. Complete the table of **cosine measures** between all the user vectors from Problem 1. Remember that the formula to compute the angle θ between two vectors $\vec{v}$, $\vec{w}$ is

$$\theta = \arccos\left(\frac{\vec{v}^T\vec{w}}{\|\vec{v}\}\|\vec{w}\|}\right).$$

In Octave, the syntax is

```
        acos(A(:,i)'*A(:,j)/(norm(A(:,i))*norm(A(:,j))))
```

with `i` and `j` replaced by particular numbers.

Cosine measures

	User 1	User 2	User 3
User 1	0		
User 2	0.11066		
User 3	0	0.11066	

7. Compare the tables in Problems 2 and 6. Which pair of users are the best match according to each measure? Which users have the most dissimilar tastes in movies?

8. Make a heat map with the cosine measure data by calling the script `cosine_sim`.

Movie-to-movie comparisons

9. Let's suppose we asked each of the users to rank three additional movies in a follow-up survey. The movie rating matrix that we obtain is given below.

$$A = \begin{bmatrix} 1 & 5 & 5 \\ 1 & 4 & 5 \\ 2 & 2 & 1 \\ 5 & 3 & 1 \\ 2 & 4 & 4 \end{bmatrix}$$

Did User 1 like the fourth movie or not?

10. Enter `A` into the Octave Command Prompt. What does the command `A'` give? What does a row of `A'` correspond to, a user or a movie?

11. Complete the table of **distance measures** between all the movie vectors above. Why are so many comparisons pre-filled with an `x`?

Distance measures

	Movie 1	Movie 2	Movie 3	Movie 4	Movie 5
Movie 1	x	1	5.0990	6	1.7321
Movie 2	x	x	4.5826	5.7446	1.4142
Movie 3	x	x	x	3.1623	
Movie 4	x	x	x	x	
Movie 5	x	x	x	x	x

12. Set `A = A'` so that columns correspond to movies, and make heat maps for movie-to-movie comparisons by running both `distance_sim` and `cosine_sim`.

13. Which two movies are most similar under each method?

14. If a new user liked Movie 1, what are some other movies they would be likely to enjoy?

Working with a larger set of survey data

15. The file `SURVEYDATA.txt` that you uploaded contains actual data collected by asking 31 users to rank 24 movies on a scale of 1 – 5. Import the survey data into Octave as a matrix by entering the following series of commands in the Command Prompt (press enter after each command):

```
survey='SURVEYDATA.txt'
[A,delimiterOut]=importdata(survey)
```

Your data is now stored in a matrix `A`. The columns of `A` correspond to *users* and the rows of `A` correspond to *movies*.

16. Create the heat maps (you can use distance measure, cosine measure, or both!) for your data for user-to-user comparisons as well as movie-to-movie comparisons.

17. You can alter the scripts slightly to prevent the computer from doing redundant calculations (the calculations corresponding to x's in the table in Problem 11, for instance). In both distance_sim and cosine_sim, change the for loops to the following:

```
for i=1:size(A,2);
for j=i+1:size(A,2);
```

    Save the .m file after you make any changes. Then run both distance_sim and cosine_sim again to see how the heat maps look different.

18. The similarity measure programs create a matrix of distances (in the case of distance_sim) or angles (in the case of cosine_sim) called final like the ones you filled in above, and you can use the commands

```
[M,I] = min(final(final>0));
[row,col] = find(final==M)
```

    to find the location (row and column) of the smallest non-zero distance/angle in this matrix after you run each program. Which users are most similar? Which movies are most similar?

19. In the end, which measure of similarity – distance or cosine – do you think is more realistic for predicting movie ratings?

20. Which comparison (user-to-user or item-to-item) is more interesting to you?