

Earn a Verified Certificate for this course!

[Get Started](#)

Programming Assignment 1: Instructions

[Help](#)

When you're ready to submit your solution, go to the [assignments list](#).

Overview

This assignment will explore non-personalized recommendations. You will be given a program stub and a data set in .csv format. You will use these to write a program that makes basic, non-personalized recommendations.

[Download Ratings Data](#) — This is a comma-separated values file, with the user number, movie ID, and rating, in that order.

[Download Movies List](#) — Decoding the movie title is not required for this assignment, but are you curious which movie is which in the ratings file? Use this file to find out!

[Download Users List](#) — Decoding the unique user id is not required for this assignment, but use this file to find your user number using the unique identifier that was provided to you after rating the movies.

[Download Java Stub](#) (be sure you can successfully compile and run before making changes)

Notes

This assignment requires you to write the code needed to parse the ratings file. It is up to you how you do this (including whether you skip ahead and use LensKit data structures or simply build your own matrix). It is particularly important to make sure you can distinguish between rated and non-rated cells in your matrix.

Deliverables

There are 2 deliverables for this assignment. Each deliverable represents a different analysis of the data provided to you. For each deliverable, you will submit a list of the top 5 movies that occur with movies A, B, and C; where A, B, and C will be [uniquely assigned to you](#). Do this for each of the two association formulas described in class (remember, your movie is x, and you are looking for the other movies y that maximize the formula values):

1. Simple: $(x \text{ and } y) / x$
2. Advanced: $((x \text{ and } y) / x) / ((!x \text{ and } y) / !x)$

[Get my 3 movie assignments](#) (your user ID can be found on the [assignments index page](#))

Output Format

For each formula, your output should be as CSV file (a file of comma-separated values) defined as follows:

Each file will have three rows (one for each movie you're computing associations for). Each row will have the movie ID of the movie assigned to you, followed by five pairs of "movie ID,predicted-score", from first to last, showing the top-five associated movies using that formula.

Note: You will be graded on both choosing the right movies and getting correct scores (rounded to the hundredths place), therefore you should provide at least two decimal places precision on your predicted scores.

Examples

Suppose that you were assigned movie IDs 11, 121, and 8587. Your submission for the first part (simple formula) would be:

```
11,603,0.96,1892,0.94,1891,0.94,120,0.93,1894,0.93
121,120,0.95,122,0.95,603,0.94,597,0.89,604,0.88
8587,603,0.92,597,0.90,607,0.87,120,0.86,13,0.86
```

...and your submission for the second part (advanced formula) would be:

```
11,1891,5.69,1892,5.65,243,5.00,1894,4.72,2164,4.11
121,122,4.74,120,3.82,2164,3.40,243,3.26,1894,3.22
8587,10020,4.18,812,4.03,7443,2.63,9331,2.46,786,2.39
```

Note that with rounding, some entries will appear to tie. Be sure to preserve the order of the output from the original algorithm.
