Ziyu Qiu

COSI 166b

PA Movies2

Sep.15 2017

**Algorithm**

My algorithm takes three factors into consideration: pastscore (the average of the user's past scores, to decide whether the user is a harsh grader), simscore (the average score the most similar users gave this movie), popularity (the average score given by the general public).

If no one has reviewed this movie, then the pastscore becomes the single decider:

predictscore = pastscore

If similar users have watched this movie, then consider both pastscore and simscore:

predictscore = pastscore\*0.6+simscore\*0.4

If non similar users have watched this movie, then use the general public's view for reference:

predictscore = pastscore\*0.6+popularity\*0.4

Pros and Cons:

I tried to consider both the grading habit of the user and the peers' (especially similar users') opinion on the given movie.

However, even the most similar users can have very low similarity with the user, and the rating habit can be very different.

**Analysis**

Here are the runnign results of my code:

|  |  |  |
| --- | --- | --- |
| **Test Set** | **Mean** | **Stdv** |
| u1 | -0.03 | 1.48 |
| u2 | -0.03 | 1.44 |
| u2 | -0.04 | 1.41 |
| u4 | -0.02 | 1.41 |
| u5 | -0.02 | 1.43 |
| ua | -0.05 | 1.45 |
| ub | -0.05 | 1.51 |
| **Average** | **-0.03** | **1.45** |

The mean is around -0.03, which is an acceptably small number indicating the whole prediction is not deviating too much. However, the standard deviation is kind of large. I have tried my best to keep it under 1.5. After several tests, I decided to take 0.6 and 0.4 as the coefficients for smallest mean and standard deviation. I could probably improve that after reading more papers, but it might not worth the time, so I think it is acceptable for now.

**Benchmarking**

Here are the running time of my code:

|  |  |
| --- | --- |
| **Test Set** | **Running Time** |
| u1 | 35s |
| u2 | 1m13s |
| u2 | 1m39s |
| u4 | 1m45s |
| u5 | 1m47s |
| ua | 1m57s |
| ub | 1m59s |
| **Average** | **1m33s** |

I used the "time" command in terminal for timing. Interestingly, even with similar data size, the running time keeps increasing. And the running time could vary when I run the program with sublime or terminal, or run it with different numbers of background programs.