Machine Learning & Data Mining
Miniproject 2: MovieLens Visualization

Caltech CS/CNS/EE 155
Released February 21^{st}, 2020

---

1 Overview

- In this project, you will be creating visualizations of the MovieLens data set using matrix factorization.
- The report is due at 9pm on Friday, February 28^{th}, via Gradescope. All visualizations and code should be included in the report in PDF format. There is no fixed formatting for this report, but a set of guidelines is included below and will be posted on Piazza.
- Each team must post at least one interesting visualization and a brief discussion of it on Piazza by 9pm on Thursday, February 27^{th}.
- You must work in a group of size either 2 or 3. We encourage you to use the Search for Teammates feature on Piazza to help you find teammates. You may keep the same group as in the previous miniproject.

2 Introduction

In late 2006, Netflix challenged the world to create a recommender system that could predict whether a user would like a given movie based on his/her previous ratings on other movies. Netflix created their own recommender system, Cinematch, and hoped that the world could beat their performance by over 10% (in terms of how closely predicted ratings match subsequent actual ratings). The challenge ended in September 2009, when team "BellKor's Pragmatic Chaos" surpassed the 10% mark.

In this miniproject, we will be focusing on creating visualizations of this data rather than actual recommender systems used to predict user ratings on movies. We will be working with the much smaller MovieLens Dataset rather than the full Netflix Prize Dataset in order to reduce the computational time needed to produce these visualizations. We will start with some basic visualizations and then move on to more complicated ones.

3 Data Format

The MovieLens data set consists of 100,000 ratings from 943 users on 1682 movies, where each user has rated at least 20 movies. More information about the files can be found below:

- **movies.txt**: Each of the 1682 lines in this file contains a tab-delimited list of the following fields for a movie:

  Movie Id, Movie Title, Unknown, Action, Adventure, Animation, Childrens, Comedy, Crime, Documentary, Drama, Fantasy, Film-Noir, Horror, Musical, Mystery, Romance, Sci-Fi, Thriller, War, Western

  The last 19 fields are various movie genres. Here, a 1 indicates the movie is of the given genre, while a 0 indicates that it is not. Note that movies can be in several genres at once. The movie ids correspond to the movie ids specified in the **data.txt** file and range from 1 to 1682. There may be movies with duplicate titles but different ids in this dataset. Additionally, some of the movies do not have reviews at all. Try to deal with both of these issues in your initial data cleaning.

- **data.txt**: Each of the 100,000 lines in this file consists of a tab-delimited list of the following fields for a given rating instance:

  User Id, Movie Id, Rating

  Here, all ratings are integer values ranging from 1 to 5. User ids range from 1 to 943 and movie ids range from 1 to 1682, as in the previous file.

4 Basic Visualizations

First, you will create some basic visualizations of the MovieLens dataset described above. Using a method (e.g. histograms) of your choice, visualize the following:

1. All ratings in the MovieLens Dataset.
2. All ratings of the ten most popular movies (movies which have received the most ratings).
3. All ratings of the ten best movies (movies with the highest average ratings).
4. All ratings of movies from three genres of your choice (create three separate visualizations).

Note that in Parts 2 and 3 you only need to make one combined histogram for the ten most popular movies and one combined histogram for the ten best movies.

The Python packages Matplotlib and Seaborn are good choices for these visualizations, but there are also many other good visualization packages.

5 Matrix Factorization Visualizations

Let m, n be the number of users and movies, respectively, and Y be the m x n matrix of the movie ratings, where y_{ij} corresponds to user i's rating for the movie j. Note that most of the elements of the matrix are unknown. The goal of a recommender system is to predict these missing values.

Your job is to find the matrices U and V, such that $Y \simeq U^T V$. Note that U has dimension $k \times m$ and V has dimension $k \times n$. You must try at least three methods for finding U and V.

1. Use (and/or modify) your code for Homework 5.
2. Incorporate bias terms a and b for each user and movie, to model global tendencies of the various users and movies. See the guide for more information. You should write your own implementation of this method.
3. Use an off-the-shelf implementation[1]. Google "collaborative filtering python," "collaborative filtering matlab," etc. to see examples. Note that in this assignment, we want you to try an off-the-shelf matrix factorization method, rather than any collaborative filtering method in general.

[1] One off-the-shelf method we suggest is Surprise SVD, found at http://surpriselib.com/.

For the first two methods, choose *k = 20*, and justify your choices for any other parameters and the stopping criteria you use. For all of these methods, split the MovieLens dataset into a training set (of size 90,000) and a test set (of size 10,000), as given in the files **train.txt** and **test.txt**. You should then compare these methods by assessing their performance on the test set. Once you have obtained U, V, you will attempt to visualize and interpret your results.

1. In order to visualize the resulting latent factors, apply SVD to $V = A\Sigma B$ and use the first two columns of A to project U, V into a two-dimensional space. This projection is given by $\tilde{U} = A_{1:2}^T U \in \mathbb{R}^{2\times m}$ and $\tilde{V} = A_{1:2}^T V \in \mathbb{R}^{2\times n}$.

2. Now, construct creative 2D-visualizations of $\tilde{V}$, similar to the one in Figure 2 of the reference [1]. For each of the three matrix factorization methods, visualize the following:

   (a) Any ten movies of your choice from the MovieLens dataset.

   (b) The ten most popular movies (movies which have received the most ratings).

   (c) The ten best movies (movies with the highest average ratings).

   (d) Ten movies from the three genres you selected in Section 4, Basic Visualizations (for a total of 30 movies). Create one visualization, containing ten movies, for each of the three genres you select.

Report Guideline

Your report should consist of 4 sections: Introduction, Basic Visualizations, Matrix Factorization Methods, and Matrix Factorization Visualizations.

Introduction (worth 10 points)

For this section, you should include your group members, team name, and division of labour.

Basic Visualizations (worth 20 points)

For this section, include the visualizations requested in the Basic Visualizations section. What, in general, did you observe? Did the results match what you would expect to see? How do the ratings from the most popular movies compare to the ratings of the best movies? How do the ratings of the three genres you chose compare to one another?

Matrix Factorization Methods (worth 40 points)

For this section, include a description of the 3 different types of SVD that you implemented (normal SVD from Homework 5, SVD with a global bias term described on Slide 8 of the Miniproject guide, and SVD using an off-the-shelf matrix factorization method). How do each of these methods work? How do they differ? How did they perform in comparison to one another on the test set? Can these methods differences explain why they perform differently on the test set?

Additionally, in this section, you should include 6 graphs for each of the 3 different types of SVD that you implemented (described in Section 5, part 2 of the Miniproject 2 report), for a total of 18 graphs.

Matrix Factorization Visualizations (worth 30 points)

For this section, describe some general trends you observe in the data through your plots. What, in general, did you observe? Did the results match what you would expect to see? How does the visualization of the most popular movies compare to that of the best movies? How do the visualizations of the three genres you chose compare to one another? How do the visualizations produced by the different matrix factorization methods compare to one another? Be sure to include some plots to indicate which phenomena youre referring to with respect to your observations.

Additionally, we will be offering bonus points for particularly interesting blog posts, so have fun with those! We're looking forward to seeing your blog posts.

6 Submission Instructions

Only one group member is required to submit the project per team. The report should be submitted as a **single PDF file** to Gradescope at the aforementioned date and time. Be sure to include all team member names in the PDF document and be sure to appropriately title all visualizations wherever they appear so that it is clear which question each visualization corresponds to. Append code to the end of the document.

In addition to submitting the project on Gradescope, one group member of each team should make a post on Piazza at the aforementioned date and time of at least one interesting visualization the team created. The post should be made in the project2 folder with the title "[Team Name]: Visualization Submission," and the first line of the post should be "Submitted by: [Team Members]." The post should contain at least one visualization created using matrix factorization and a brief description (1 paragraph should suffice) of what makes the visualization so interesting. We will put a sample post on Piazza for reference.

References

1. Koren, Y., Bell, R.,& Volinsky, C. (2009). Matrix Factorization Techniques for Recommender Systems Computer, (8), 30-37.

2. Herlocker, J. L., Konstan, J. A., Borchers, A., & Riedl, J. (1999, August). An algorithmic framework for performing collaborative filtering. In Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval (pp. 230-237). ACM.