Machine Learning & Data Mining — Caltech CS/CNS/EE 155
Miniproject 3: Unsupervised Learning for Visualization — Released March 3rd, 2017

TAs: Avishek Dutta, Jagriti Agrawal

1 Overview

- In this project, you will be creating visualizations of the MovieLens data set using matrix factorization.
- All visualizations and code used to generate the visualizations are due at 9pm on Friday, March 10th via Moodle. The reports are due at 9pm on Monday, March 13th, again via Moodle.
- You can work in groups of up to three. 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 miniprojects.

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 that 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.

- **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 gotten 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 for this problem).

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

Report Deliverable

Your report should contain a section dedicated to basic visualizations. 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?

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.

First, use your code or the solution code for Homework 6 to find the matrices U and V, such that $Y \simeq U^T V$. You may also use off-the-shelf implementations if you want but be aware that these may or may yield correct results. Note that U has dimension $k \times m$ and V has dimension $k \times n$. Choose $k = 20$, and justify your choices for any other parameters and the stopping criteria you use. 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 $\tilde{V} = A_{1:2}^T V \in \mathbb{R}^{2\times n}$ and $\tilde{U} = A_{1:2}^T U \in \mathbb{R}^{2\times m}$.
2. Now construct creative 2D-visualizations of $\tilde{V}$, similar to the one in Figure 2 of the reference [1]. Visualize the following:
   (a) Ten random movies of your choice from the MovieLens Dataset.
   (b) The ten most popular movies (movies which have gotten 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 Deliverable

Your report should contain a section dedicated to matrix factorization visualizations. 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?

6 Submission Instructions

Only one group member is required to submit the project per team. The code used to generate the visualizations and a **single PDF file** containing all the visualizations should be submitted to Moodle at the aforementioned date and time in a **.zip archive**. Use your team name for the name of the file (replacing any spaces with underscores). The report should be submitted as a **single PDF file** to Moodle at the aformentioned date and time. Again, use your team name for the name of the file.

Be sure to include all team member names in both PDF documents and be sure to appropriately title all visualizations wherever they appear so that it is clear which question each visualization corresponds to.

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.

[3] GraphLab's Collaborative Filtering Library