Alex Riley & Katelyn Stringer
March 23, 2018

STAT 689 Class Project:
Building a Recommendation Engine for Movies

MovieLens is a non-commercial website run by the research group GroupLens that provides users with movie recommendations. Users can use the site to search for a movie based on its title, its genre, the actors/actresses, and more. They can also rate movies, tag them with identifying words, such as the name of the director, a topic the movie addresses, or a feeling the movie's atmosphere evokes, etc. This identifying information helps other users find the movie in either recommendations or their own searches. MovieLens has been around for over 20 years, creating a rich and detailed data set for building accurate recommendation algorithms.

Description of dataset:
For this project, we will use the stable benchmark version of the MovieLens dataset https://grouplens.org/datasets/movielens/. This version includes 2,000,026 ratings and 465,564 tags for 27,278 movies uploaded by 138,493 users between January 9, 1995 and March 31, 2015. Each user included in the data set had rated at least 20 movies, but was otherwise chosen at random. No identifying information on the user was recorded except for their movie ratings, tags, and timestamps for the two. All the movies included had at least one rating.

Plan for building Recommendation Engine:
We plan to build a recommendation engine for the data set using the collaborative filtering technique. To do so, we will ramp the complexity of the algorithm:

- Simple models: average of responses for a movie/user (see class slides on Netflix Prize)
- Weighted averages: test a variety of possible ways to measure similarity between users and provide a recommendation based on previous ratings by users with similar preferences (http://www.dataperspective.info/2014/05/basic-recommendation-engine-using-r.html)

For this section we will focus on "user-based" collaborative filtering (see reach goals for more). We will consider a predicted rating of 4 or greater as a recommendation (this number might also change).

Evaluation of Results:
To train the collaborative filtering algorithm, we will use most of the ratings as a training set, and intentionally blind some ratings to use as a test case. If the predicted ratings for most of the test set are within 1 star (may change) of their true values, we will consider the algorithm a success.

Reach Goals:
If we get the collaborative filtering working successfully, we will also build a recommendation model based on clustering methods ("item-based" methodology) and compare/contrast the results between the two models. This will serve as an independent check of our recommendation algorithm.