Project 2

Max Wagner

June 18, 2016

Data

The data was generated in excel. It is similar to the data from class on the movie rating, but with 26 imaginary people, and 10 movies. It uses a similar 1-5 scale, with 0 representing a movie that they have not seen. In this case, the idea of the recommendation system would be to recommend the best movies to someone. The main method will be collabortative filtering.

The head of the csv file can be seen below:

```
movies <- read.csv('movies.csv', stringsAsFactors = FALSE)
users <- movies[,1]
movies <- movies[,2:11]
movie.names <- colnames(movies)
head(movies)
```

```
##   m1 m2 m3 m4 m5 m6 m7 m8 m9 m10
## 1  0  1  1  4  0  1  1  0  3   3
## 2  3  4  0  3  3  3  2  5  3   5
## 3  4  4  4  0  0  5  1  1  2   0
## 4  2  4  2  1  0  2  5  2  1   0
## 5  3  2  2  4  1  0  1  4  3   5
## 6  4  4  2  1  5  3  5  4  1   5
```

User Based CF

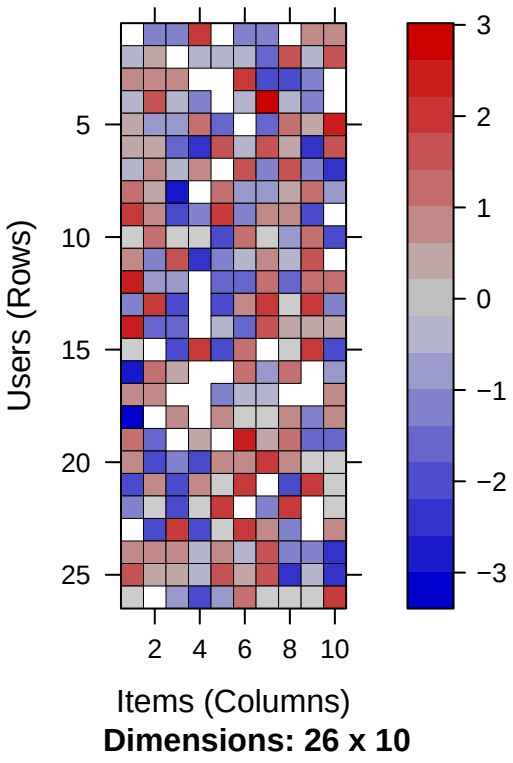
I'll use 'recommenderlab' to get a few different types of read outs. The first is user based collaborative filtering. In this case recommendations are given based on proximity to other users' ratings.

```
library(recommenderlab)
movies <- read.csv('movies.csv', stringsAsFactors = FALSE)
movies[movies==0] <- NA
movies <- as.matrix(movies[,2:11])

#make it a rating matrix
movies.rm <- as(movies, "realRatingMatrix")

#normalize it, should do this automatically in Recommender as well
movies.rmn <- normalize(movies.rm)

#graph it
plot(image(movies.rmn, main = "Normalized Rating"))
```




```
#create a recommender for what the users should see next
user.recom <- Recommender(movies.rm[1:20], method = "UBCF")
users.predict <- predict(user.recom, movies.rm[21:26],n=9)
as(users.predict, "list")
```

```
## [[1]]
## [1] "m7"
##
## [[2]]
## [1] "m6" "m9"
##
## [[3]]
## [1] "m9" "m1"
##
## [[4]]
## character(0)
##
## [[5]]
## character(0)
##
## [[6]]
## [1] "m2"
```

The above list turned out slightly differently than I had predicted at first, but it made sense when I realized it was recommending users to see the movies they had no seen yet. In the case of users that had seen all the movies, it had no recommendations for them.

Item Based CF

Similar to the user based, but instead is based purely on how movies are rated, are individual user ratings are fairly irrelevant.

```
movies.recom <- Recommender(movies.rm[1:20], method = "IBCF")
movies.predict <- predict(movies.recom, movies.rm[21:26],n=9)
as(movies.predict, "list")
```

```
## [[1]]
## [1] "m7"
##
## [[2]]
## [1] "m9" "m6"
##
## [[3]]
## [1] "m9" "m1"
##
## [[4]]
## character(0)
##
## [[5]]
## character(0)
##
## [[6]]
## [1] "m2"
```

Interestingly enough the first few recommendations are nearly identical. The one difference being that user 2 would be recommended movie 9 before movie 6 in this method. Which is better is impossible to know with a faux data set. However we can check the error of prediction for these last few rows between the two methods.

```
users.predict <- predict(user.recom, movies.rm[21:26], type="ratings")
movies.predict <- predict(movies.recom, movies.rm[21:26], type="ratings")
user.error <- calcPredictionAccuracy(users.predict, movies.rm[21:26])
movie.error <- calcPredictionAccuracy(movies.predict, movies.rm[21:26])
rbind(user.error,movie.error)
```

```
##             RMSE MSE MAE
## user.error   NaN NaN NaN
## movie.error  NaN NaN NaN
```

Unfortunately the data set seems too small to get accurate error measurements, but the method should work in most realistic cases.