Recommendation Engines - Non-Personalised

Import libraries

```
library(readr)
```

USING THE CRITICS DATASET

Load the critics dataset

```
critics <- read.csv('/Users/taniaelachkar/Desktop/MBD/Term 2/Recommendation Engine
s/Lab1/Lab1_Tania_ElAchkar/critics.csv', sep=',', dec=',')
critics <- as.data.frame(critics)
colnames(critics) <- gsub('.', ' ', colnames(critics), fixed=T)
critics
```

```
##        User Star Wars IV   A New Hope Star Wars VI   Return of the Jedi
## 1      John                          1                                5
## 2     Maria                          5                                3
## 3     Anton                         NA                               NA
## 4     Roger                         NA                                3
## 5   Martina                          4                                3
## 6       Ana                          2                                4
## 7     Sergi                         NA                               NA
## 8      Marc                          4                               NA
## 9       Jim                          5                                1
## 10    Chris                          4                                2
## 11  Bernard                          2                                1
## 12    Nuria                          3                                5
## 13    Nerea                          2                                3
## 14   Carles                          3                               NA
## 15 Victoria                          4                                4
## 16     Ivan                         NA                               NA
## 17   Rachel                         NA                               NA
## 18    Nadia                          4                                5
## 19    Oriol                          5                                1
## 20   Valery                          1                                2
##    Forrest Gump The Shawshank Redemption The Silence of the Lambs
## 1             2                       NA                        4
## 2            NA                        2                        4
## 3            NA                        5                        2
## 4            NA                       NA                       NA
## 5             4                        1                        4
## 6             4                        4                       NA
## 7             3                        1                        1
## 8            NA                       NA                        3
```

```
## 9             NA                      4                       2
## 10            NA                      5                       3
## 11             5                     NA                      NA
## 12             2                     NA                       2
## 13            NA                      5                       4
## 14             3                     NA                       2
## 15            NA                     NA                       5
## 16             1                     NA                       3
## 17            NA                     NA                       4
## 18             1                      5                       1
## 19            NA                     NA                      NA
## 20             2                      4                       5
##    Gladiator Toy Story Saving Private Ryan Pulp Fiction Stand by Me
## 1          4         2                    2           NA            3
## 2          2         1                   NA           NA            4
## 3         NA         4                   NA           NA            1
## 4          1         2                    3            4           NA
## 5          1        NA                    4           NA            1
## 6         NA         3                    1            4            4
## 7          4        NA                    5            2           NA
## 8          2         2                   NA            3           NA
## 9          4         4                    4           NA            1
## 10        NA         4                    3            4           NA
## 11         5         5                   NA           NA           NA
## 12        NA         2                   NA            1           NA
## 13        NA         4                    5           NA           NA
## 14        NA         3                   NA           NA            4
## 15         5         2                   NA            3            5
## 16         2        NA                    2           NA            1
## 17        NA         2                   NA            2           NA
## 18         1         4                   NA            5           NA
## 19         4         2                    1            3            3
## 20        NA         2                    3            2            2
##    Shakespeare in Love Total Recall Independence Day Blade Runner
## 1                    2           NA                5            2
## 2                    3            2                2           NA
## 3                   NA            1                4            4
## 4                   NA            4                1            3
## 5                    5            1               NA            4
## 6                    5            2                4           NA
## 7                    1           NA               NA            3
## 8                   NA            2                3            2
## 9                    2            3                1           NA
## 10                  NA           NA                2           NA
## 11                  NA            3                2           NA
## 12                   3           NA                3           NA
## 13                  NA            2                4           NA
## 14                  NA            1                2            2
## 15                  NA            1                3           NA
## 16                   5           NA               NA           NA
```

```
## 17                   2           NA                NA           NA
## 18                  NA           NA                NA            4
## 19                   3            1                NA           NA
## 20                   1           NA                NA            5
##    Groundhog Day The Matrix Schindler s List The Sixth Sense
## 1             NA          4                 2               5
## 2              2         NA                 5               1
## 3              1          1                 2               3
## 4              5         NA                 5               1
## 5             NA          3                 5               5
## 6              1         NA                NA               3
## 7             NA          1                NA              NA
## 8              4         NA                 1               3
## 9              5         NA                NA              NA
## 10            NA          2                 5               1
## 11            NA          1                NA               2
## 12             2          5                NA              NA
## 13             3         NA                NA              NA
## 14             3          5                 1              NA
## 15             2         NA                 3              NA
## 16             5          2                 2               4
## 17            NA          4                NA              NA
## 18            NA         NA                 2               1
## 19            NA          2                NA              NA
## 20             5          4                 3               5
##    Raiders of the Lost Ark Babe
## 1                        NA   NA
## 2                         3   NA
## 3                         1    3
## 4                         1    2
## 5                        NA   NA
## 6                        NA    2
## 7                         5    2
## 8                         5   NA
## 9                        NA    5
## 10                       NA   NA
## 11                        1    4
## 12                       NA    2
## 13                        5   NA
## 14                       NA   NA
## 15                        3    1
## 16                        3    4
## 17                       NA   NA
## 18                        2    5
## 19                       NA   NA
## 20                        3   NA
```

Get the top 5 movies, ordered by the mean of their ratings

```
# Calculate the mean of each column(movie), except for the first one (the User col
umn)
means1 <- colMeans(critics[,-1], na.rm=T)

# Store the column names of the original critics dataframe, except for the first c
olumn (User)
column <- colnames(critics)
columns <- column[2:21]

# Create a new dataframe called means2 and store the mean rating for each movie in
that dataframe
means2 <- data.frame(movie_name=columns, rating=means1, row.names=NULL)

# Order this previous dataframe and store the top 5 movies (by rating) in a new da
taframe called means4
means3 <- means2[order(means2$rating,decreasing=T), ]
means4 <- means3[1:5,]

# Print the top 5 movies and their ratings
paste(means4$rating, means4$movie_name, sep=',')
```

```
## [1] "3.6,The Shawshank Redemption"
## [2] "3.26666666666667,Star Wars IV   A New Hope"
## [3] "3.22222222222222,Blade Runner"
## [4] "3.16666666666667,Groundhog Day"
## [5] "3.0625,The Silence of the Lambs"
```

Get the top 5 movies, ordered by their ratings

```
# Calculate, for each movie, the number of rows that are not NA (the number of rat
ings given for each movie)
count <- sapply(critics[,-1], function(x) {sum(!is.na(x))})

# Calculate, for each movie, the number of ratings that are greater than or equal
to 4
greater <- sapply(critics[,-1], function(x) {length(which(x>=4))})

# Calculate, for each movie, the percentage of ratings that are greater than or eq
ual to 4 and store the output in a dataframe
ratings <- greater / count
ratings2 <- data.frame(ratings)
top <- data.frame(movies=rownames(ratings2),ratings,row.names=NULL)

# Order the movies by rating distribution and store the top 5 in a new dataframe c
alled top5
top2 <- top[order(top$ratings,decreasing=T),]
top5 <- top2[1:5,]

# Print the top 5 movies by rating, along with their rating
paste(top5$ratings,top5$movies,sep=',')
```

```
## [1] "0.7,The Shawshank Redemption"
## [2] "0.533333333333333,Star Wars IV   A New Hope"
## [3] "0.5,Gladiator"
## [4] "0.444444444444444,Blade Runner"
## [5] "0.4375,The Silence of the Lambs"
```

Get the top 5 movies, ordered by the number of ratings they received

```
# Calculate, for each movie, the number of rows that are not NA (the number of rat
ings given for each movie) and store the output in a
# dataframe
counts <- sapply(critics[,-1],function(x) {sum(!is.na(x))})
counts2 <- data.frame(counts)
counts3 <- data.frame(movies=rownames(counts2),numb_ratings=counts,row.names=NULL)

# Order the movies by quantity of ratings and store the top 5 in a new dataframe c
alled ordered2
ordered <- counts3[order(counts3$numb_ratings,decreasing=T),]
ordered2 <- ordered[1:5,]

# Print the top 5 movies by quantity of ratings, along with their ratings
paste(ordered2$numb_ratings,ordered2$movies,sep=',')
```

```
## [1] "17,Toy Story"
## [2] "16,The Silence of the Lambs"
## [3] "15,Star Wars IV   A New Hope"
## [4] "14,Star Wars VI   Return of the Jedi"
## [5] "13,Independence Day"
```

Get the top 5 movies recommended to users who also watched The Wizard of Oz

```
# Remove the first column of the critics dataframe. It contains user names
critics2 <- critics[,-1]

# Subset the previous dataframe to contain the rows where there are no NA values f
or the Star Wars IV   A New Hope movie. Since we're comparing the
# percentage of other movie raters who also rated that movie, we need to keep the
rows of the users who rate that movie, hence no NA
critics3 <- critics2[which(!is.na(critics2$`Star Wars IV   A New Hope`)), ]

# Calculate the number of users who rated each movie, divided by the number of use
rs who rated Star Wars IV   A New Hope. The result, which is the
# percentage of other movie raters who also rated Star Wars IV   A New Hope is sto
red in a dataframe
percents <- sapply(critics3, function(x){sum(!is.na(x))/nrow(critics3)})
percents2 <- data.frame(percents)
percents3 <- data.frame(movies=rownames(percents2),occurences=percents,row.names=N
ULL)

# Order the movies based on the highest percentage
p_order <- percents3[order(percents3$occurences,decreasing=T),]

# Store the top 5 in a new dataframe called top5. We start the index at 2 because
our reference movie, Star Wars IV   A New Hope, has the highest
# result of 1 because it's compared to itself
p_order1 <- p_order[2:6,]

# Print the top 5 movies by percentage of other users who also rated Star Wars IV
A New Hope
paste(p_order1$occurences,p_order1$movies,sep=',')
```

```
## [1] "0.933333333333333,Toy Story"
## [2] "0.866666666666667,Star Wars VI   Return of the Jedi"
## [3] "0.8,The Silence of the Lambs"
## [4] "0.733333333333333,Independence Day"
## [5] "0.666666666666667,Total Recall"
```

Get the top 5 movie recommendations for users who also liked Babe

```
# Subset the original critics dataframe to only include the rows of users who rate
d the movie Babe (no NA values)
critics4 <- critics[!is.na(critics$Babe),c('User','Babe')]

# Subset the previous dataframe to keep the rows of users who gave either 4 or 5 s
tars to Babe, meaning that they liked the movie
critics5 <- critics4[(critics4$Babe==4|critics4$Babe==5),]

# Subset the previous dataframe to remove the Babe column because it is our refere
nce movie
critics6 <- critics[,-ncol(critics)]

# Subset the previous dataframe to keep the users who rated Babe with 4 or 5 stars
, but keeping all the movies, except for Babe
critics7 <- critics6[which(critics$User %in% critics5$User),]

# For these users, calculate the average rating they gave to other movies, and sto
re the result in a dataframe
mean1 <- colMeans(critics7[,-1], na.rm=T)
mean2 <- data.frame(movie=colnames(critics6[,-1]),rating=mean1,row.names=NULL)

# Order the dataframe by ratings and select the top 5
mean3 <- mean2[order(mean2$rating,decreasing=T),]
mean4 <- mean3[1:5,]

# Print the top 5 movies for people who gave 4 or 5 stars to Babe
paste(mean4$rating,mean4$movie,sep=',')
```

```
## [1] "5,Pulp Fiction"               "5,Groundhog Day"
## [3] "4.5,The Shawshank Redemption" "4.33333333333333,Toy Story"
## [5] "4,Blade Runner"
```

USING ANOTHER DATASET, MOVIELENS, WHICH HAS A DIFFERENT STORAGE LAYOUT THAN THE CRITICS DATASET WE HAVE BEEN WORKING WITH PREVIOUSLY

Load the MovieLens dataset

```
movies <- read.csv("/Users/taniaelachkar/Desktop/MBD/Term 2/Recommendation Engines
/Lab1/Lab1_Tania_ElAchkar/movies.csv", sep=',')
ratings <- read.csv("/Users/taniaelachkar/Desktop/MBD/Term 2/Recommendation Engine
s/Lab1/Lab1_Tania_ElAchkar/ratings.csv", sep=',')
head(movies)
```

```
##   movieId                               title
## 1       1                    Toy Story (1995)
## 2       2                      Jumanji (1995)
## 3       3             Grumpier Old Men (1995)
## 4       4            Waiting to Exhale (1995)
## 5       5 Father of the Bride Part II (1995)
## 6       6                           Heat (1995)
##                                         genres
## 1 Adventure|Animation|Children|Comedy|Fantasy
## 2                  Adventure|Children|Fantasy
## 3                               Comedy|Romance
## 4                        Comedy|Drama|Romance
## 5                                      Comedy
## 6                        Action|Crime|Thriller
```

```
head(ratings)
```

```
##   userId movieId rating  timestamp
## 1      1      31    2.5 1260759144
## 2      1    1029    3.0 1260759179
## 3      1    1061    3.0 1260759182
## 4      1    1129    2.0 1260759185
## 5      1    1172    4.0 1260759205
## 6      1    1263    2.0 1260759151
```

Merge both datasets into a new one and remove the movieId, genres, and timestamp variables

```
mr0 <- merge(movies, ratings, by.x='movieId')
mr <- mr0[,-c(1,3,6)]
```

Filter the dataframe and keep the rows/movies that are rated by at least 100 users

```
agg <- aggregate(mr$rating, by=list(mr$title), FUN=function(x) {length(x)})
agg1 <- subset(agg, x>100)
agg2 <- agg1[, -2]
mr1 <- mr[mr$title %in% agg2,]
head(mr1)
```

```
##              title userId rating
## 1 Toy Story (1995)     23    3.0
## 2 Toy Story (1995)    623    4.5
## 3 Toy Story (1995)    559    4.0
## 4 Toy Story (1995)    306    3.0
## 5 Toy Story (1995)    361    3.0
## 6 Toy Story (1995)    357    5.0
```

Get the top 5 movies ordered by the mean of their ratings

```
# Use the aggregate function to get the mean rating of each movie, grouping the re
sult by movie
mean_r <- aggregate(mr1$rating, by=list(mr1$title), FUN=function(x) {sum(x)/length
(x)})

# Convert the output to a data frame and ordering the ratings and keeping the top
5 results
mean_r1 <- data.frame(movies=mean_r$Group.1, ratings=mean_r$x, row.names=NULL)
mean_r2 <- mean_r1[order(mean_r1$ratings, decreasing=T), ]
mean_r3 <- mean_r2[1:5,]

# Print the top 5 movies by movie ratings
paste(mean_r3$ratings, mean_r3$movies, sep=',')
```

```
## [1] "4.4875,Godfather, The (1972)"
## [2] "4.48713826366559,Shawshank Redemption, The (1994)"
## [3] "4.38518518518519,Godfather: Part II, The (1974)"
## [4] "4.37064676616915,Usual Suspects, The (1995)"
## [5] "4.30327868852459,Schindler's List (1993)"
```

Get the top 5 movies, ordered by their ratings

```
# For each movie, calculate the percentage of ratings that are 4 stars or grater
percent_r <- aggregate(mr1$rating, by=list(mr1$title), FUN=function(x) {(length(wh
ich(x>=4)))/(sum(!is.na(x)))})

# Rename the columns of this new data frame
colnames(percent_r) <- c('movie', 'percent >= 4stars')

# Order the movies by rating distribution
percent_r1 <- percent_r[order(percent_r$`percent >= 4stars`, decreasing=T),]
percent_r2 <- percent_r1[1:5,]

# Print the top 5 movies by rating distribution
paste(percent_r2$`percent >= 4stars`, percent_r2$movie, sep=',')
```

```
## [1] "0.89,Godfather, The (1972)"
## [2] "0.881028938906752,Shawshank Redemption, The (1994)"
## [3] "0.860696517412935,Usual Suspects, The (1995)"
## [4] "0.844444444444444,Godfather: Part II, The (1974)"
## [5] "0.844262295081967,Schindler's List (1993)"
```

Get the top 5 movies, ordered by the number of ratings they received

```
# For each movie, calculate the number of rows that are not NA
counts_r <- aggregate(mr1$rating, by=list(mr1$title), FUN=function(x) {sum(!is.na(
x))})
colnames(counts_r) <- c('movie', 'number_of_ratings')

# Order the movies by quantity of ratings
order_r <- counts_r[order(counts_r$number_of_ratings, decreasing=T), ]
order_r1 <- order_r[1:5,]

# Print the top 5 movies by quantity of ratings
paste(order_r1$number_of_ratings, order_r1$movie, sep=',')
```

```
## [1] "341,Forrest Gump (1994)"
## [2] "324,Pulp Fiction (1994)"
## [3] "311,Shawshank Redemption, The (1994)"
## [4] "304,Silence of the Lambs, The (1991)"
## [5] "291,Star Wars: Episode IV - A New Hope (1977)"
```

Get the top 5 movies recommended to users who also watched Toy Story

```
# Subset the initial dataframe mr1 to include the users who rated Toy Story
mr2 <- mr1[mr1$title=='Toy Story (1995)',]
ts <- mr1[which(mr1$userId %in% mr2$userId), ]

# Calculate the number of users who rated each movie, divided by the number of use
rs who rated Toy Story. The result is the percentage of
# other movie raters who also rated Toy Story
numb_users <- length(unique(mr2$userId))
ts1 <- aggregate(ts$userId, by=list(ts$title), FUN=function(x) {(length(unique(x))
)/numb_users})
colnames(ts1) <- c('movie', 'percentage')

# Order the movies based on the highest percentage
ts2 <- ts1[order(ts1$percentage, decreasing=T), ]

# Store the top 5 in a new dataframe called ts3. We start the index at 2 because o
ur reference movie, Toy Story, has the highest result of
# 1 because it's compared to itself
ts3 <- ts2[2:6,]

# Print the top 5 movies that people who watched Toy Story also watched
paste(ts3$percentage, ts3$movie, sep=',')
```

```
## [1] "0.712550607287449,Forrest Gump (1994)"
## [2] "0.663967611336032,Star Wars: Episode IV - A New Hope (1977)"
## [3] "0.619433198380567,Pulp Fiction (1994)"
## [4] "0.595141700404858,Jurassic Park (1993)"
## [5] "0.591093117408907,Shawshank Redemption, The (1994)"
```

Get the top 5 movie recommendations for users who also liked Toy Story

```
# Subset our mr1 dataframe to keep the rows for the movie Toy Story, meaning that
we want to keep the users who have rated this movie
mr2 <- mr1[mr1$title=='Toy Story (1995)',]

# Subset the previous dataframe to keep the rows of users who gave either 4 or 5 s
tars to Toy Story
mr3 <- mr2[(mr2$rating==4|mr2$rating==5),]

# Subset the initial dataframe mr1 to remove the rows for the movie Toy Story beca
use it is our reference movie
mr4 <- mr1[!(mr1$title=='Toy Story (1995)'), ]

# Subset the previous dataframe to keep the users who rated Toy Story with 4 or 5
stars, but keeping all the movies, except for Toy Story
mr5 <- mr4[which(mr4$userId %in% mr3$userId), ]

# For these users, calculate the average rating they gave to other movies
avg_rating <- aggregate(mr5$rating, by=list(mr5$title), FUN=function(x) {sum(x)/le
ngth(x)})
colnames(avg_rating) <- c('movie', 'average_rating')

# Order the dataframe by ratings and select the top 5
avg_rating1 <- avg_rating[order(avg_rating$average_rating, decreasing=T), ]
avg_rating2 <- avg_rating1[1:5, ]

# Print the top 5 movies that people who watched Toy Story also liked
paste(avg_rating2$average_rating, avg_rating2$movie, sep=',')
```

```
## [1] "4.63125,Shawshank Redemption, The (1994)"
## [2] "4.50819672131148,Schindler's List (1993)"
## [3] "4.46875,Dark Knight, The (2008)"
## [4] "4.46296296296296,Godfather, The (1972)"
## [5] "4.43965517241379,Usual Suspects, The (1995)"
```