Determining the top 10 movies

Gosuddin Siddiqi

January 31, 2017

The following report is a part of the coursework at UW INFX 573 class. The datasets used was provided as a part of coursework and is a subset of the original MovieLens dataset

```
movies <- read.csv("movie.titles.csv")
ratings <- read.csv("ratings.csv")

summary(ratings)
```

```
##      userId       movieId           rating           year
##  Min.   :  1   Min.   :     1   Min.   :0.500   Min.   :1902
##  1st Qu.:182   1st Qu.:  1028   1st Qu.:3.000   1st Qu.:1987
##  Median :367   Median :  2406   Median :4.000   Median :1995
##  Mean   :347   Mean   : 12549   Mean   :3.544   Mean   :1992
##  3rd Qu.:520   3rd Qu.:  5418   3rd Qu.:4.000   3rd Qu.:2001
##  Max.   :671   Max.   :163949   Max.   :5.000   Max.   :2016
##                                                 NA's   :4
##                   genre
##  Drama                : 7757
##  Comedy               : 6748
##  Comedy|Romance       : 3973
##  Drama|Romance        : 3462
##  Comedy|Drama         : 3272
##  Comedy|Drama|Romance: 3204
##  (Other)              :71588
```

Merging the dataset

```
movies_With_Ratings <- merge(movies, ratings, by = "movieId")
```

Counting the number of ratings per movie

```
legi_movies <- movies_With_Ratings %>% select(movieId, title, rating) %>% group_by(movieId,
    title, rating) %>% summarize(n_r = n(), avg_ratings = mean(rating)) %>%
    arrange(desc(n_r))
```

I filtered the movies that have legitmate number of votes before the average ratings could be calculated. This would eliminate the errorneous single votes that falsely make the movie #1. I choose a threshold of 200 movies in the decreasing order of the number of votes.

```
popular200 <- head(legi_movies, 200)
```

I then arranged the movies according to the decreasing order of the avg_ratings. This could be used to select the top 10 movies.

```
top10 <- popular200 %>% group_by(movieId, title) %>% filter(sum(n_r) > 100) %>%
    summarise(avg_ratings = mean(rating), w_r = weighted.mean(rating, n_r)) %>%
    arrange(desc(w_r))


head(top10$title, 10)
```

```
##  [1] Godfather, The (1972)
##  [2] Shawshank Redemption, The (1994)
##  [3] Schindler's List (1993)
##  [4] Lord of the Rings: The Fellowship of the Ring, The (2001)
##  [5] Star Wars: Episode IV - A New Hope (1977)
##  [6] Matrix, The (1999)
##  [7] Usual Suspects, The (1995)
##  [8] Fargo (1996)
##  [9] Star Wars: Episode V - The Empire Strikes Back (1980)
## [10] Fight Club (1999)
## 9123 Levels: 'burbs, The (1989) ... Zulu (2013)
```

Top 90s movies

More can be done on this dataset to produce top 10 movies based on the years such as 70s, 80s, 90s and so on. Also, the dataset could also be filtered based on the genre. I have implemented an example of each.

```
legi_movies_90 <- movies_With_Ratings %>% filter(year.x >= 1990 & year.x < 2000) %>%
    select(movieId, title, rating, year.x) %>% group_by(movieId, title, rating) %>%
    summarize(n_r = n(), avg_ratings = mean(rating)) %>% arrange(desc(n_r))

popular100_90 <- head(legi_movies_90, 100)  ## re adjusting to top 100 because of subset created

top10_90 <- popular100_90 %>% group_by(movieId, title) %>% filter(sum(n_r) >
    100) %>% summarise(avg_ratings = mean(rating), w_r = weighted.mean(rating,
    n_r)) %>% arrange(desc(w_r))

head(top10_90$title, 10)
```

```
##  [1] Shawshank Redemption, The (1994) Schindler's List (1993)
##  [3] Matrix, The (1999)               Usual Suspects, The (1995)
##  [5] Fargo (1996)                     Fight Club (1999)
##  [7] American Beauty (1999)           Braveheart (1995)
##  [9] Sixth Sense, The (1999)          Silence of the Lambs, The (1991)
## 9123 Levels: 'burbs, The (1989) ... Zulu (2013)
```

Top Action Movies

```
legi_movies_action <- movies_With_Ratings %>% filter(grepl("Action", genre,
    ignore.case = T)) %>% select(movieId, title, rating, year.x) %>% group_by(movieId,
    title, rating) %>% summarize(n_r = n(), avg_ratings = mean(rating)) %>%
    arrange(desc(n_r))
```

```
popular_action <- head(legi_movies_action, 25)
## re adjusting to top 20 because of subset created

top10_action <- popular_action %>% group_by(movieId, title) %>% filter(sum(n_r) >
    75) %>% summarise(avg_ratings = mean(rating), w_r = weighted.mean(rating,
    n_r)) %>% arrange(desc(w_r))

head(top10_action$title, 10)
```

```
## [1] Matrix, The (1999)
## [2] Star Wars: Episode IV - A New Hope (1977)
## [3] Star Wars: Episode V - The Empire Strikes Back (1980)
## [4] Braveheart (1995)
## [5] Raiders of the Lost Ark (Indiana Jones and the Raiders of the Lost Ark) (1981)
## [6] Star Wars: Episode VI - Return of the Jedi (1983)
## [7] Terminator 2: Judgment Day (1991)
## [8] Jurassic Park (1993)
## [9] Batman (1989)
## 9123 Levels: 'burbs, The (1989) ... Zulu (2013)
```

IMBD rankings

According to the imdb formula, I calculated the top 10 movies. They claim that they use true Bayesian estimate.

Top 10

```
C <- mean(ratings$rating)
m <- 84  #lowest number of votes as per my quantiles calculation

top10_imdb_formula <- legi_movies %>% mutate(weighted_ratings = ((n_r/(n_r +
    m)) * avg_ratings) + (m/(n_r + m)) * C) %>% arrange(desc(weighted_ratings))

head(top10_imdb_formula$title, 10)
```

```
##  [1] Shawshank Redemption, The (1994)
##  [2] Pulp Fiction (1994)
##  [3] Star Wars: Episode IV - A New Hope (1977)
##  [4] Schindler's List (1993)
##  [5] Godfather, The (1972)
##  [6] Forrest Gump (1994)
##  [7] Silence of the Lambs, The (1991)
##  [8] Fargo (1996)
##  [9] Matrix, The (1999)
## [10] Star Wars: Episode V - The Empire Strikes Back (1980)
## 9123 Levels: 'burbs, The (1989) ... Zulu (2013)
```

Top 10 90s Movies

```
top10_imdb_formula <- legi_movies_90 %>% mutate(weighted_ratings = ((n_r/(n_r +
    m)) * avg_ratings) + (m/(n_r + m)) * C) %>% arrange(desc(weighted_ratings))

head(top10_imdb_formula$title, 10)
```

```
##  [1] Shawshank Redemption, The (1994) Pulp Fiction (1994)
##  [3] Schindler's List (1993)          Forrest Gump (1994)
##  [5] Silence of the Lambs, The (1991) Fargo (1996)
##  [7] Matrix, The (1999)               Usual Suspects, The (1995)
##  [9] American Beauty (1999)           Braveheart (1995)
## 9123 Levels: 'burbs, The (1989) ... Zulu (2013)
```

Top 10 Action movies

```
top10_imdb_formula <- legi_movies_action %>% mutate(weighted_ratings = ((n_r/(n_r +
    m)) * avg_ratings) + (m/(n_r + m)) * C) %>% arrange(desc(weighted_ratings))

head(top10_imdb_formula$title, 10)
```

```
##  [1] Star Wars: Episode IV - A New Hope (1977)
##  [2] Matrix, The (1999)
##  [3] Star Wars: Episode V - The Empire Strikes Back (1980)
##  [4] Raiders of the Lost Ark (Indiana Jones and the Raiders of the Lost Ark) (1981)
##  [5] Braveheart (1995)
##  [6] Fight Club (1999)
##  [7] Terminator 2: Judgment Day (1991)
##  [8] Star Wars: Episode VI - Return of the Jedi (1983)
##  [9] Lord of the Rings: The Return of the King, The (2003)
## [10] Princess Bride, The (1987)
## 9123 Levels: 'burbs, The (1989) ... Zulu (2013)
```