Analyzing Movie Data from MovieLens

Data Science with R

https://raw.githubusercontent.com/dnchari/DS501_MovieLens/master/Results/unifiedMLDataMulti.csv

```
movielens = 'https://raw.githubusercontent.com/dnchari/DS501_MovieLens/master/Results/unifiedMLDataMulti
mlData = read.csv(movielens)
```

How many movies have an average rating over 4.5 overall? 11

```
movielens = 'https://raw.githubusercontent.com/dnchari/DS501_MovieLens/master/Results/unifiedMLDataMulti
mlData = read.csv(movielens)
# number of movies with average rating over 4.5
movie_avg_rating = aggregate(mlData$rating, list(mlData$movie_title), FUN=mean)
names(movie_avg_rating)[1] <- "movie_title"
names(movie_avg_rating)[2] <- "rating"
count_movie = nrow(movie_avg_rating[movie_avg_rating$rating > 4.5,])
print(paste("Number of movies with an average rating over 4.5:", count_movie))
```

```
## [1] "Number of movies with an average rating over 4.5: 11"
```

How many movies have an average rating over 4.5 among men? How about women?

```
# number of movies with average rating over 4.5 per gender
movie_avg_rating_gend = aggregate(mlData$rating, list(mlData$movie_title, mlData$gender), FUN=mean)
names(movie_avg_rating_gend)[1] <- "movie_title"
names(movie_avg_rating_gend)[2] <- "gender"
names(movie_avg_rating_gend)[3] <- "rating"

count_movie_F = nrow(movie_avg_rating_gend[movie_avg_rating_gend$rating > 4.5 & movie_avg_rating_gend ==
print(paste("Number of movies with an average rating over 4.5 among women:", count_movie_F))
```

```
## [1] "Number of movies with an average rating over 4.5 among women: 16"
```

```
count_movie_M = nrow(movie_avg_rating_gend[movie_avg_rating_gend$rating > 4.5 & movie_avg_rating_gend ==
print(paste("Number of movies with an average rating over 4.5 among men:", count_movie_M))
```

```
## [1] "Number of movies with an average rating over 4.5 among men: 18"
```

How many movies have a median rating over 4.5 among men over age 30? How about women over age 30?

```
# number of movies with average rating over 4.5 per gender and over 30 years old

# removes rows that the age is not above 30
mlData_remove_under = mlData[mlData$age > 30,]
movie_median_rating <- aggregate(mlData_remove_under$rating, list(mlData_remove_under$movie_title, mlDat

# renaming columns
names(movie_median_rating)[1] <- "movie_title"
names(movie_median_rating)[2] <- "gender"
names(movie_median_rating)[3] <- "rating"

count_movie_F_over = nrow(movie_avg_rating_gend[movie_median_rating$rating > 4.5 & movie_median_rating =
print(paste("Number of movies with a median rating over 4.5 among women over age 30:", count_movie_F_ove
```

```
## [1] "Number of movies with a median rating over 4.5 among women over age 30: 70"
```

```
count_movie_M_over = nrow(movie_avg_rating_gend[movie_median_rating$rating > 4.5 & movie_median_rating =
print(paste("Number of movies with a median rating over 4.5 among men over age 30:", count_movie_M_over)
```

```
## [1] "Number of movies with a median rating over 4.5 among men over age 30: 47"
```

What are the ten most popular movies?

```
#The top 10 movies based on highest average rating are

movie_avg_rating = aggregate(mlData$rating, list(mlData$movie_title), FUN=mean)
names(movie_avg_rating)[1] <- "movie_title"
names(movie_avg_rating)[2] <- "rating"

top_movies = movie_avg_rating[movie_avg_rating$rating == 5,]
top_movies$movie_title
```

```
##  [1] "Aiqing wansui (1994)"
##  [2] "Entertaining Angels: The Dorothy Day Story (1996)"
##  [3] "Great Day in Harlem, A (1994)"
##  [4] "Marlene Dietrich: Shadow and Light (1996) "
##  [5] "Prefontaine (1997)"
##  [6] "Saint of Fort Washington, The (1993)"
##  [7] "Santa with Muscles (1996)"
##  [8] "Someone Else's America (1995)"
##  [9] "Star Kid (1997)"
## [10] "They Made Me a Criminal (1939)"
```

Basic Statistics - How many movies have an average rating over 4.5 overall? - How many movies have an average rating over 4.5 among men? How about women? - How many movies have an median rating over 4.5 among men over age 30? How about women over age 30? - What are the ten most popular movies? + Choose what you consider to be a reasonable definition of "popular". + Be prepared to defend this choice. - Make some conjectures about how easy various groups are to please? Support your answers with data! + For example, one might conjecture that people between the ages of 1 and 10 are the easiest to please since they are all young children. This conjecture may or may not be true, but how would you support or disprove either conclusion with with data? + Be sure to come up with your own conjectures and support them with data!

```r
# looking into age and rating
mlData2 = mlData

# converting the age ranges
mlData2$age[mlData2$age > 0 & mlData2$age <= 10] <- 0
mlData2$age[mlData2$age > 10 & mlData2$age <= 20] <- 1
mlData2$age[mlData2$age > 20 & mlData2$age <= 30] <- 2
mlData2$age[mlData2$age > 30 & mlData2$age <= 40] <- 3
mlData2$age[mlData2$age > 40 & mlData2$age <= 50] <- 4
mlData2$age[mlData2$age > 50 & mlData2$age <= 60] <- 5
mlData2$age[mlData2$age > 60 & mlData2$age <= 70] <- 6
mlData2$age[mlData2$age > 70 & mlData2$age <= 80] <- 7
# average rating by age
age_rating = aggregate(mlData2$rating, list(mlData2$age), FUN=mean)
names(age_rating)[1] <- "age"
names(age_rating)[2] <- "average_rating"
age_rating_sort = age_rating[order(-age_rating$average_rating),]

age_rating_sort
```

```
##   age average_rating
## 8   7       4.042017
## 6   5       3.705056
## 7   6       3.625108
## 2   1       3.591653
## 5   4       3.591063
## 4   3       3.586259
## 1   0       3.570000
## 3   2       3.468967
```

```r
# occupation, movie genre, and rating
occupation_movie_avg_rating = aggregate(mlData$rating, list(mlData$occupation), FUN=mean)
names(occupation_movie_avg_rating)[1] <- "occupation"
names(occupation_movie_avg_rating)[2] <- "rating"
occupation_movie_avg_rating_sort = occupation_movie_avg_rating[order(-occupation_movie_avg_rating$ratin

genre_movie_avg_rating = aggregate(mlData$rating, list(mlData$genre), FUN=mean)
names(genre_movie_avg_rating)[1] <- "genre"
names(genre_movie_avg_rating)[2] <- "rating"
genre_movie_avg_rating_sort = genre_movie_avg_rating[order(-genre_movie_avg_rating$rating),]
genre_movie_avg_rating_sort
```

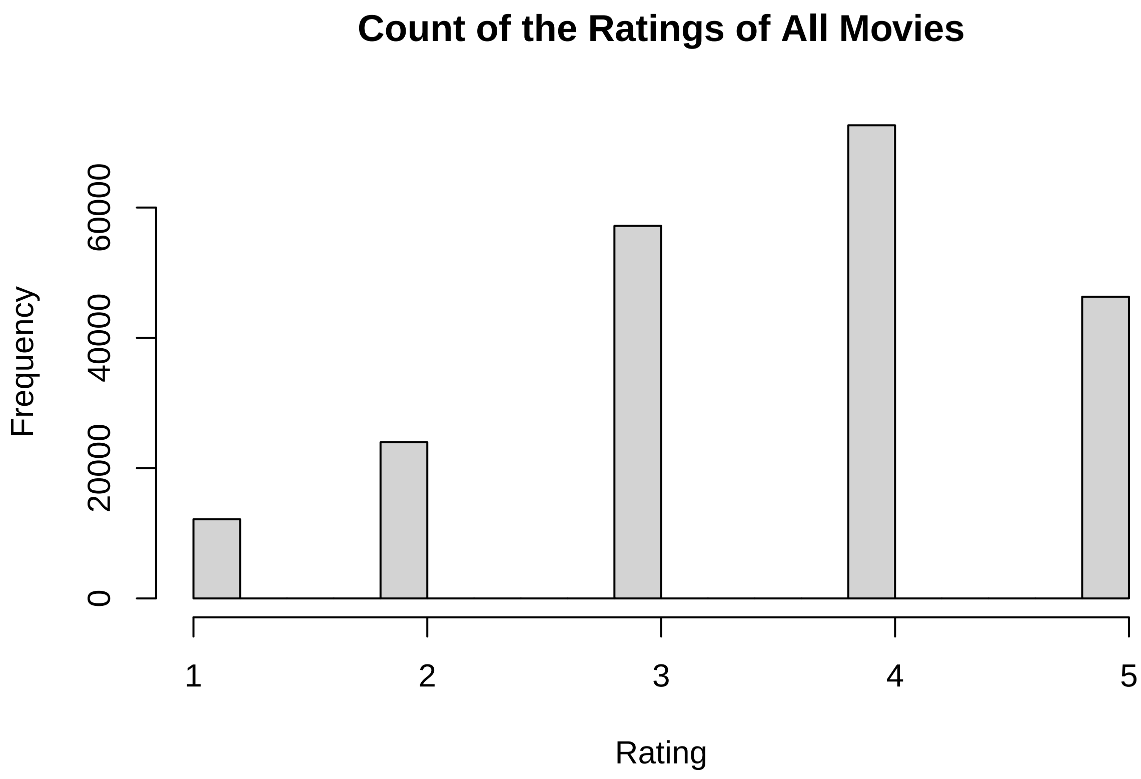
```
##          genre   rating
## 10   Film-Noir 3.921523
## 17         War 3.815812
## 8        Drama 3.686978
## 7  Documentary 3.672823
## 13     Mystery 3.639154
## 6        Crime 3.633039
## 14     Romance 3.620384
## 18     Western 3.613269
## 3    Animation 3.576699
## 15      Sci-Fi 3.560968
## 12     Musical 3.521397
## 16    Thriller 3.509335
## 2    Adventure 3.504520
## 1       Action 3.481567
## 5       Comedy 3.394775
## 4    Childrens 3.352818
## 11      Horror 3.290875
## 9      Fantasy 3.215237
```

Problem 2: Expand our investigation to histograms

An obvious issue with any inferences drawn from Problem 1 is that we did not consider how many times a movie was rated.

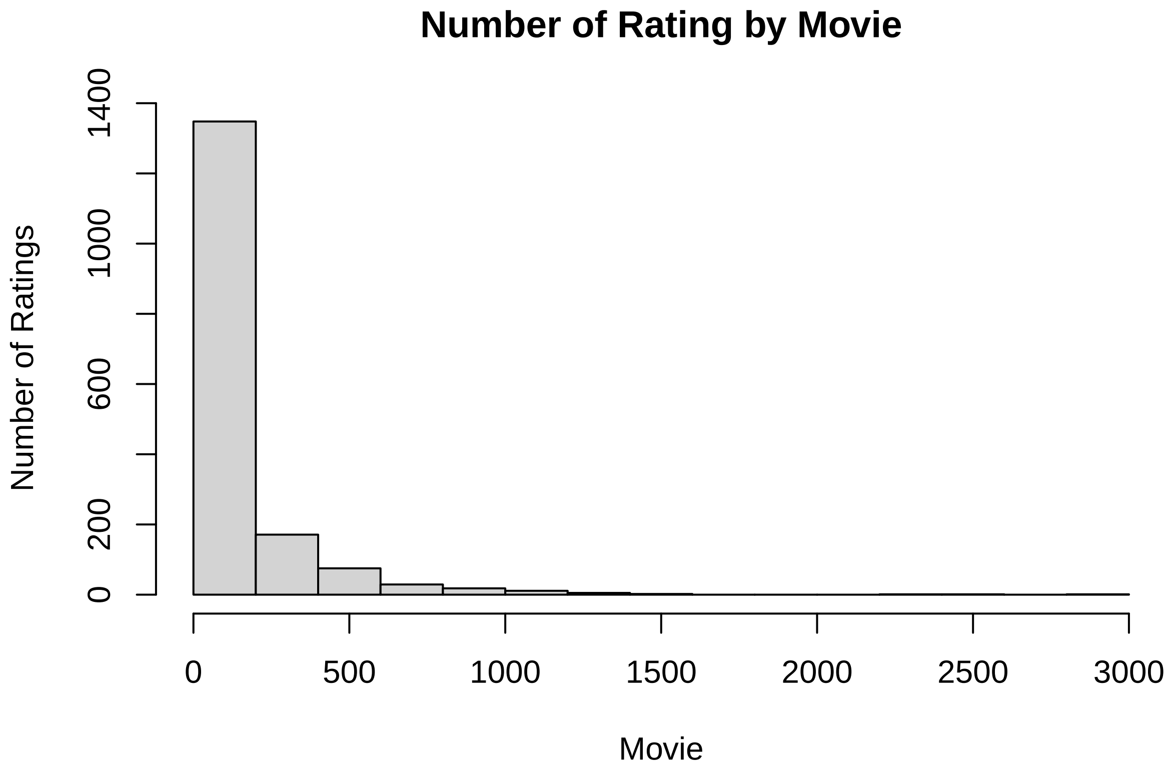
- Plot a histogram of the ratings of all movies.

```
Rating <- mlData$rating
hist(Rating, main="Count of the Ratings of All Movies")
```



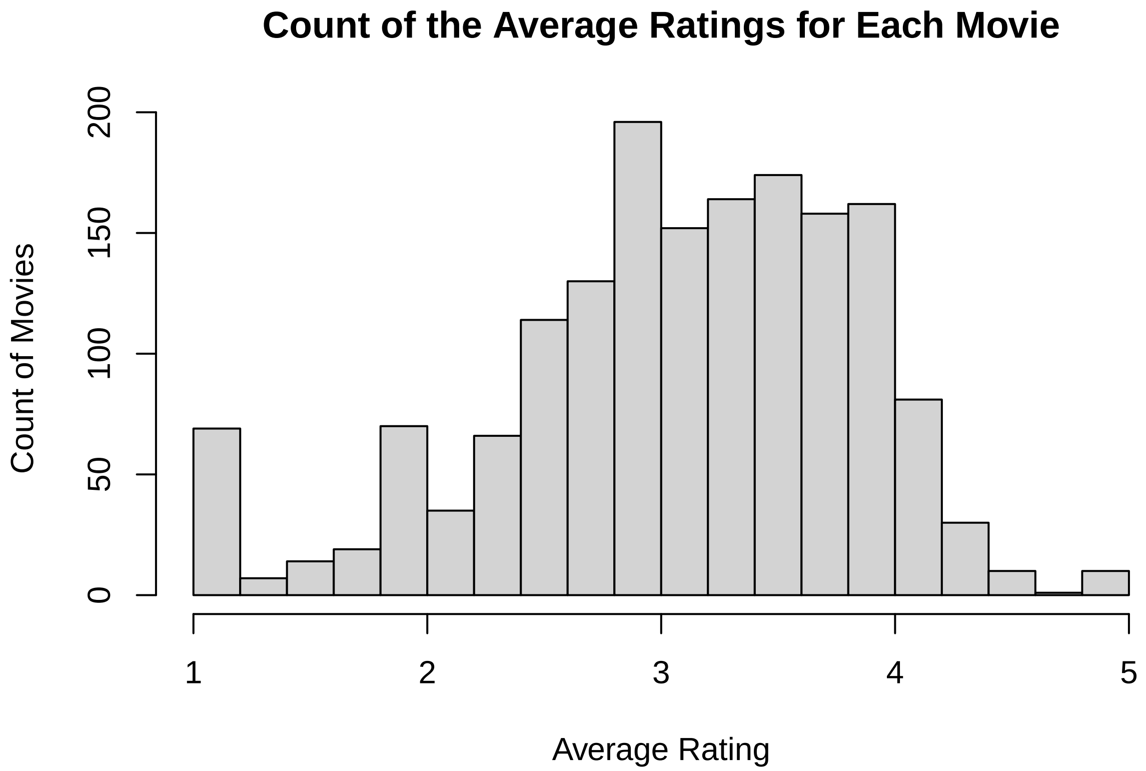
- Plot a histogram of the number of ratings each movie received.

```
count_ratings_per_movie = aggregate(mlData$rating, list(mlData$movie_title),FUN =length)
names(count_ratings_per_movie)[1] <- "movie_title"
names(count_ratings_per_movie)[2] <- "count"

hist(count_ratings_per_movie$count, main="Number of Rating by Movie", xlab="Movie", ylab="Number of Rat
```



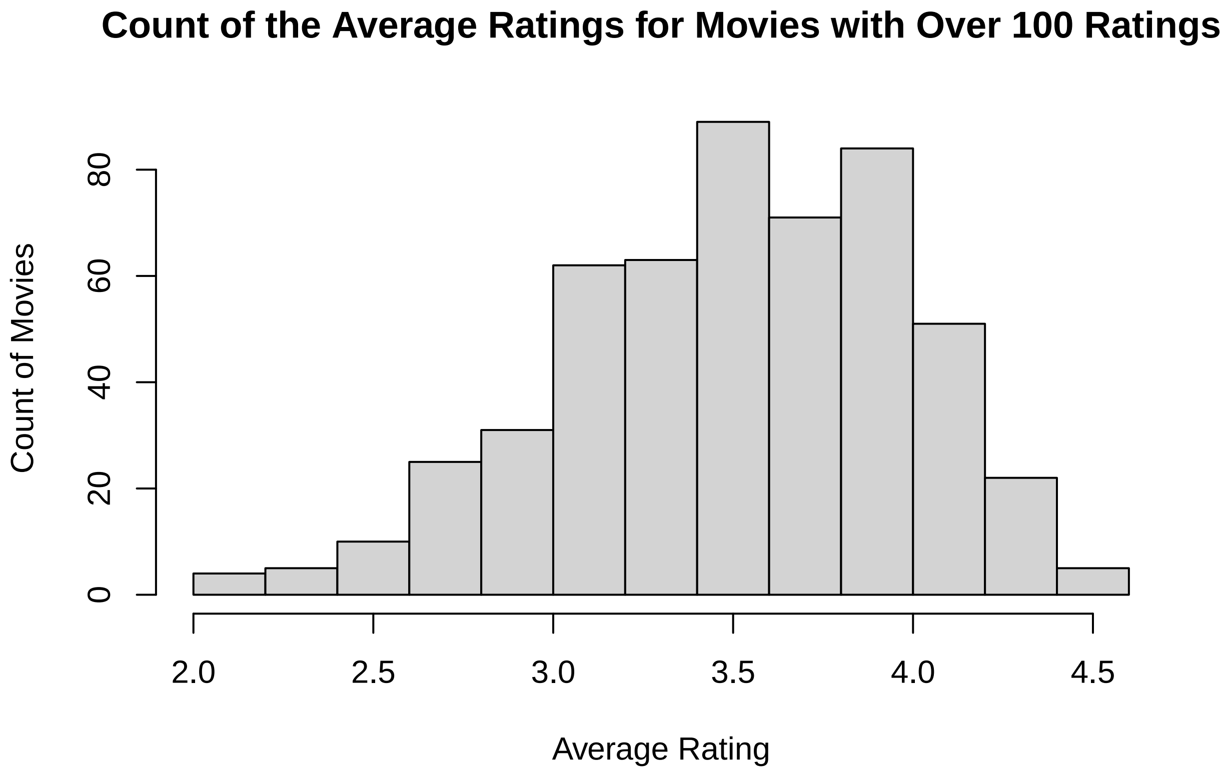
- Plot a histogram of the average rating for each movie.

```
hist(movie_avg_rating$rating, main="Count of the Average Ratings for Each Movie", xlab="Average Rating"
```



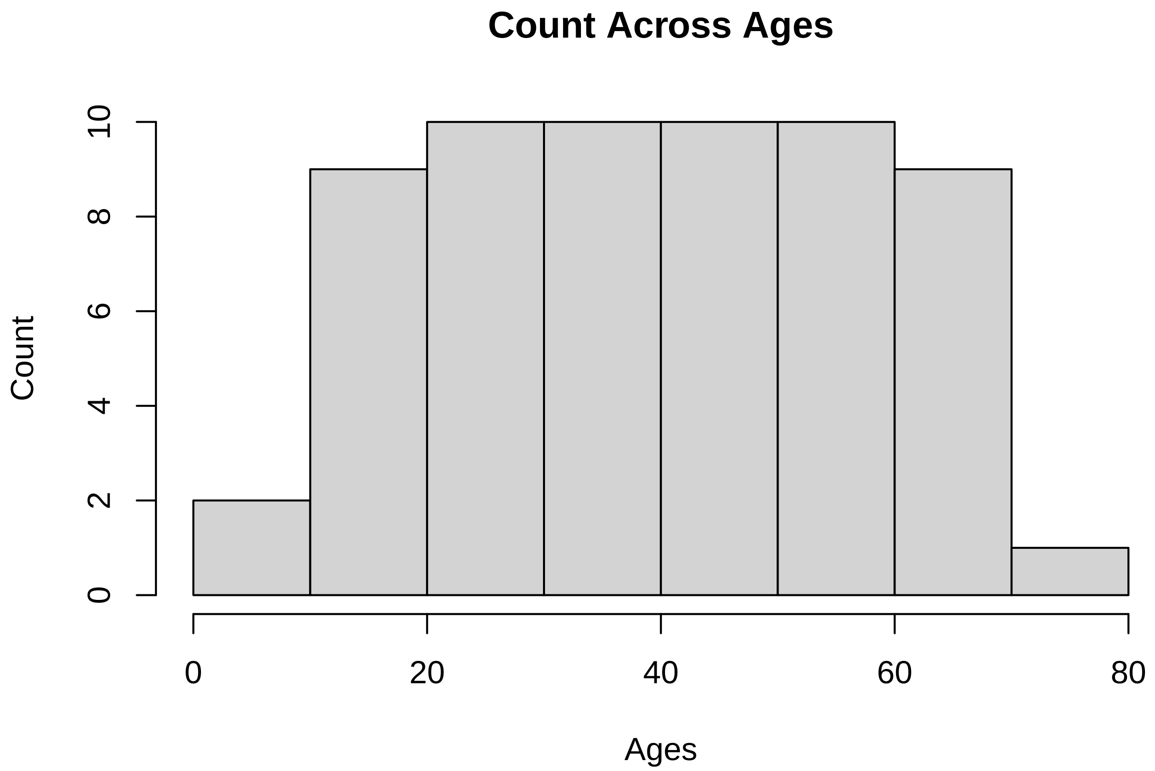
- Plot a histogram of the average rating for movies which are rated more than 100 times.
  - What do you observe about the tails of the histogram where you use all the movies versus the one where you only use movies rated more than 100 times?
  - Which highly rated movies would you trust are actually good? Those rated more than 100 times or those rated less than 100 times?

```
#Histogram of the average rating for movies which were rated more than 100

count_average_df = merge(count_ratings_per_movie,movie_avg_rating,by="movie_title")
count_average_df_cleaned = count_average_df[count_average_df$count > 100,]

hist(count_average_df_cleaned$rating, main="Count of the Average Ratings for Movies with Over 100 Ratin
```



- Make some conjectures about the distribution of ratings? Support your answers with data!

```
count_ratings_per_age = aggregate(mlData$rating, list(mlData$age),FUN =length)
names(count_ratings_per_age)[1] <- "age_range"
names(count_ratings_per_age)[2] <- "count"

hist(count_ratings_per_age$age_range, breaks = 7, main = 'Count Across Ages', xlab = 'Ages', ylab = 'Co
```



```
# count for age ranges 0 and 7 with rartings of 1 and 5
count_movie_5_0 = nrow(mlData2[mlData2$rating == 5 & mlData2$age == '0',])
count_movie_5_0
```

```
## [1] 49
```

```
count_movie_1_0 = nrow(mlData2[mlData2$rating == 1 & mlData2$age == '0',])
count_movie_1_0
```

```
## [1] 15
```

```
count_movie_5_7 = nrow(mlData2[mlData2$rating == 5 & mlData2$age == '7',])
count_movie_5_7
```

```
## [1] 36
```

```
count_movie_1_7 = nrow(mlData2[mlData2$rating == 1 & mlData2$age == '7',])
count_movie_1_7
```

```
## [1] 3
```

```
# total 1 and 5 ratings
count_movie_5 = nrow(mlData2[mlData2$rating == 5,])
count_movie_5
```

```
## [1] 46319
```

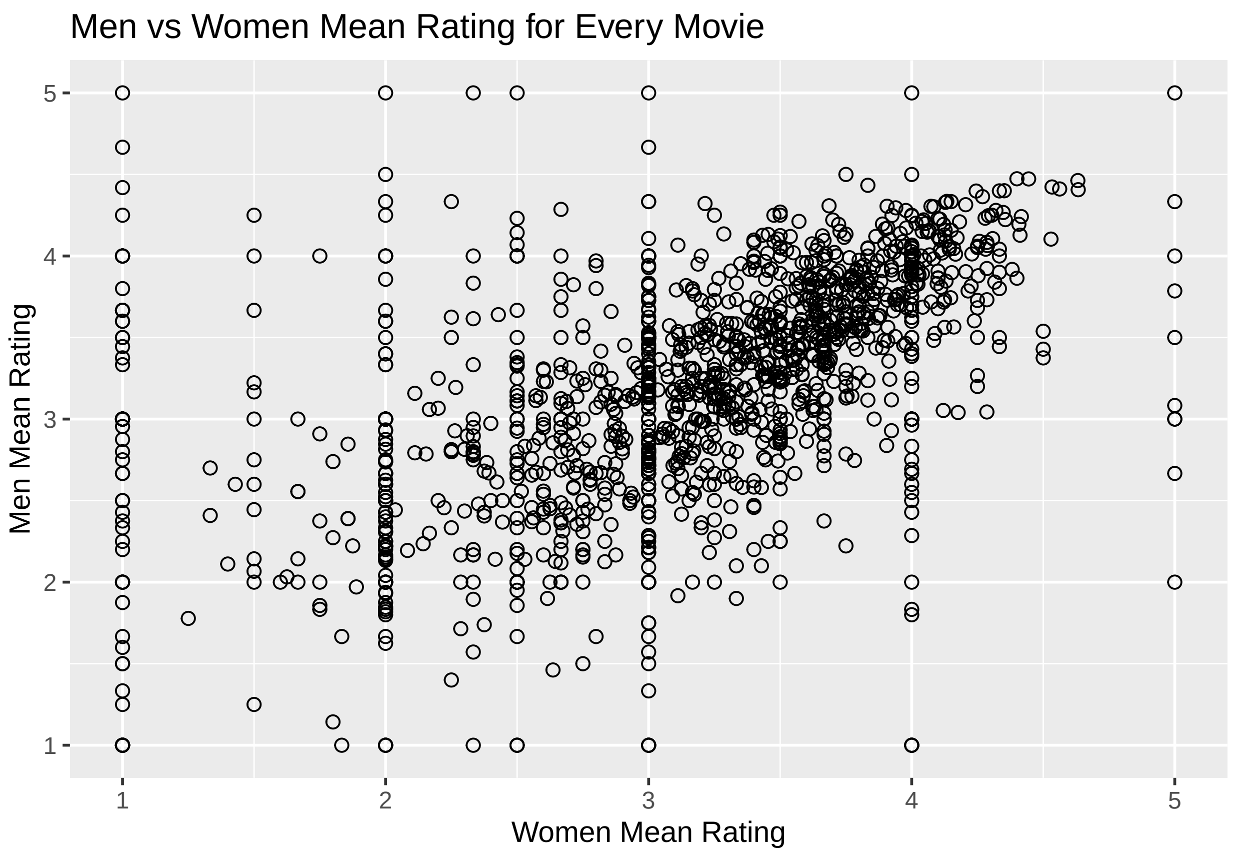
```
count_movie_1 = nrow(mlData2[mlData2$rating == 1 ,])
count_movie_1
```

```
## [1] 12146
```

Problem 3: Correlation: Men versus women

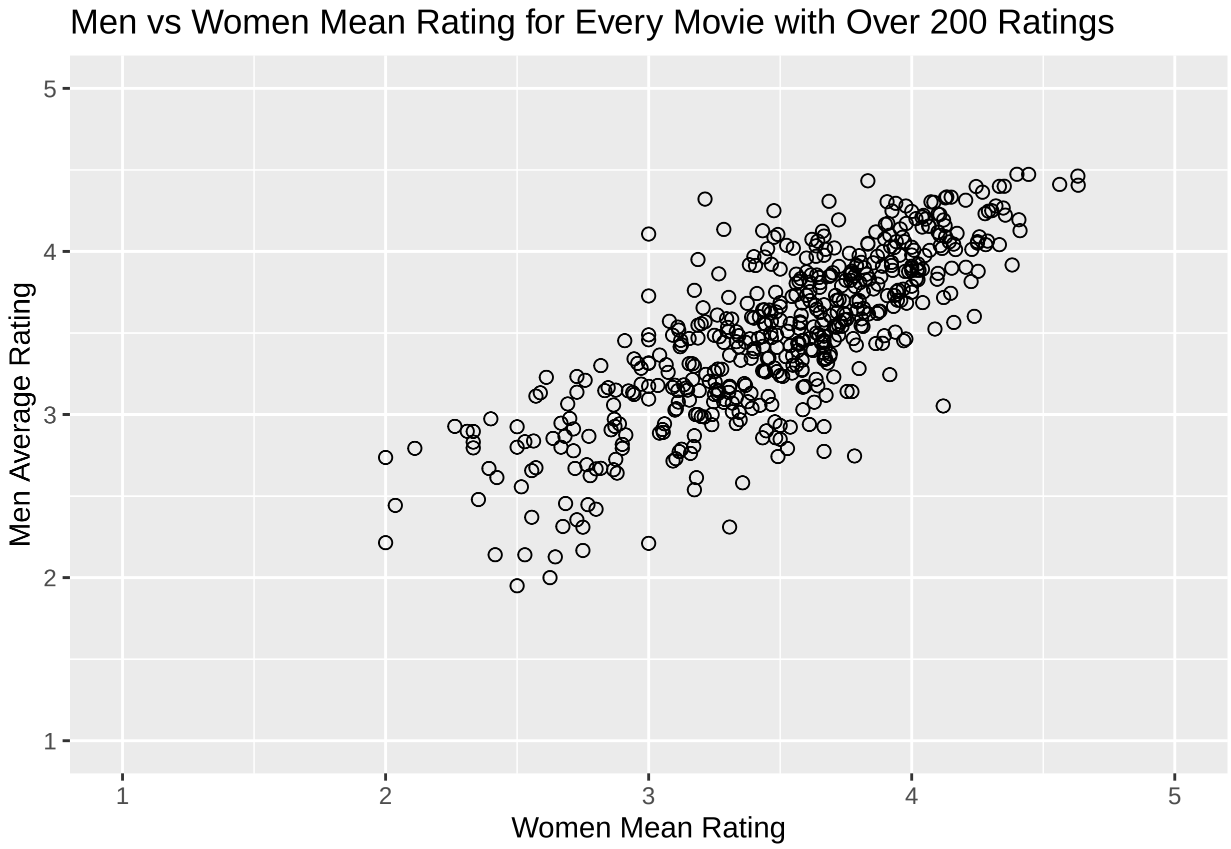
- Make a scatter plot of men versus women and their mean rating for every movie.

```
#scatter plot for men and women for their average rating for every movie
library(ggplot2)
movie_avg_rating_gend_F = movie_avg_rating_gend[movie_avg_rating_gend$gender == 'F',]
movie_avg_rating_gend_M = movie_avg_rating_gend[movie_avg_rating_gend$gender == 'M',]

names(movie_avg_rating_gend_F)[3] <- "rating_F"
names(movie_avg_rating_gend_M)[3] <- "rating_M"

gender_movie_rating = merge(movie_avg_rating_gend_F,movie_avg_rating_gend_M,by="movie_title")

ggplot(gender_movie_rating, aes(x=rating_F, y=rating_M)) + geom_point(size=2, shape=1)+
labs(title="Men vs Women Mean Rating for Every Movie",x="Women Mean Rating", y="Men Mean Rating")
```



- Make a scatter plot of men versus women and their mean rating for movies rated more than 200 times.

```
# scatter plot for men and women for movies with over 200 ratings
gender_movie_rating_1 = merge(count_ratings_per_movie,gender_movie_rating,by="movie_title")


gender_movie_rating_cleaned = gender_movie_rating_1[gender_movie_rating_1$count > 100,]

ggplot(gender_movie_rating_cleaned, aes(x=rating_F, y=rating_M)) + geom_point(size=2, shape=1)+
labs(title="Men vs Women Mean Rating for Every Movie with Over 200 Ratings",x="Women Mean Rating", y="M
```



- Compute the correlation coefficent between the ratings of men and women.

```
# correlation coefficient between rating of men and women
man_women_all_cor = cor(gender_movie_rating$rating_F, gender_movie_rating$rating_M)
print(paste("The correlation coefficient between all the ratings of men and women of movies is", man_wo
```

```
## [1] "The correlation coefficient between all the ratings of men and women of movies is 0.51519610319
```

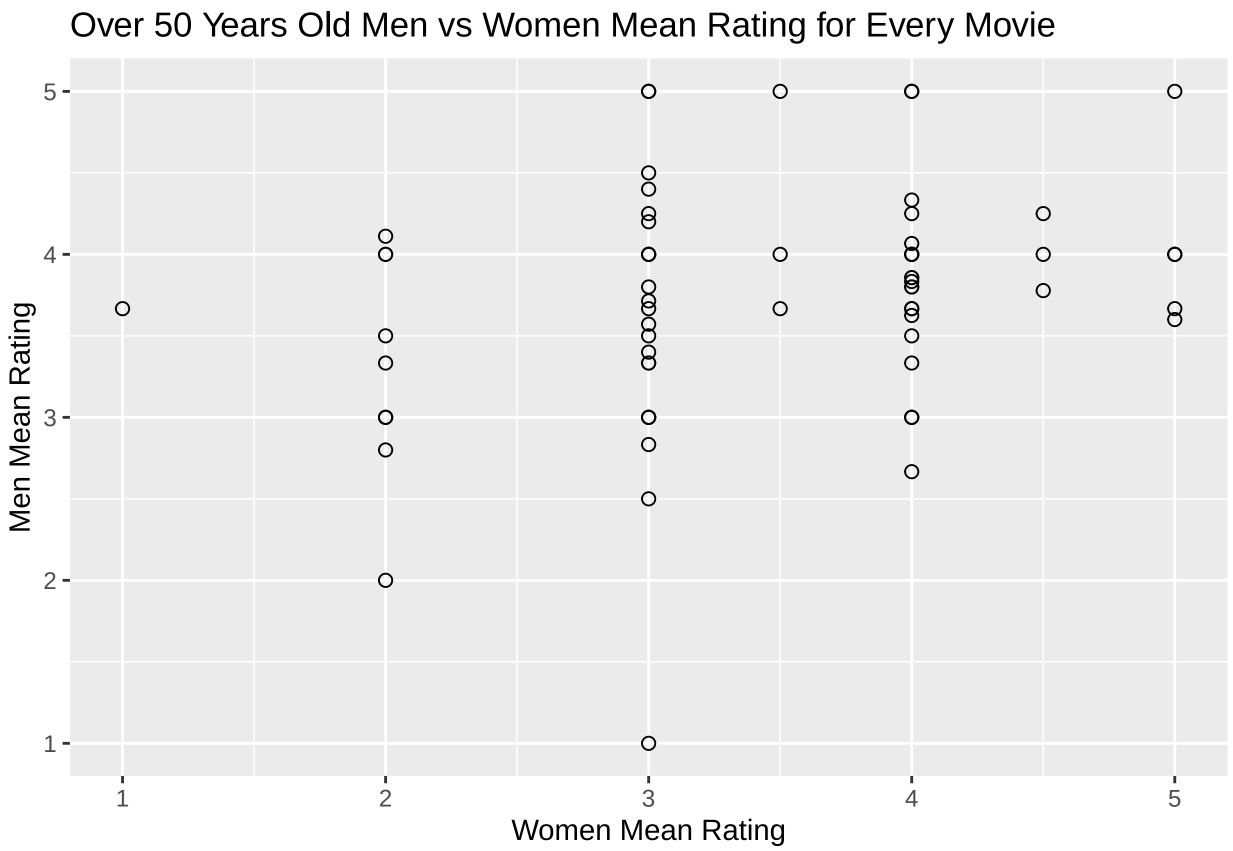
```
# correlation coefficient between rating of men and women
man_women_under_200_cor = cor(gender_movie_rating_cleaned$rating_F, gender_movie_rating_cleaned$rating_
print(paste("The correlation coefficient between the ratings of men and women of movies with over 200 r
```

```
## [1] "The correlation coefficient between the ratings of men and women of movies with over 200 rating
```

- Conjecture under what circumstances the rating given by one gender can be used to predict the rating given by the other gender.

```
#men and women 50+ scatter plot
movie_avg_rating_gend_age = aggregate(mlData2$rating, list(mlData2$movie_title, mlData2$gender, mlData2
names(movie_avg_rating_gend_age)[1] <- "movie_title"
names(movie_avg_rating_gend_age)[2] <- "gender"
names(movie_avg_rating_gend_age)[3] <- "age"
names(movie_avg_rating_gend_age)[4] <- "rating"
```

```
movie_avg_rating_gend_age_F = movie_avg_rating_gend_age[movie_avg_rating_gend_age$gender == 'F'& movie_a
movie_avg_rating_gend_age_M = movie_avg_rating_gend_age[movie_avg_rating_gend_age$gender == 'M' & movie_

names(movie_avg_rating_gend_age_F)[4] <- "rating_F"
names(movie_avg_rating_gend_age_M)[4] <- "rating_M"

gender_movie_rating_age_50 = merge(movie_avg_rating_gend_age_F,movie_avg_rating_gend_age_M,by="movie_ti

ggplot(gender_movie_rating_age_50, aes(x=rating_F, y=rating_M)) + geom_point(size=2, shape=1)+
labs(title="Over 50 Years Old Men vs Women Mean Rating for Every Movie",x="Women Mean Rating", y="Men Me
```

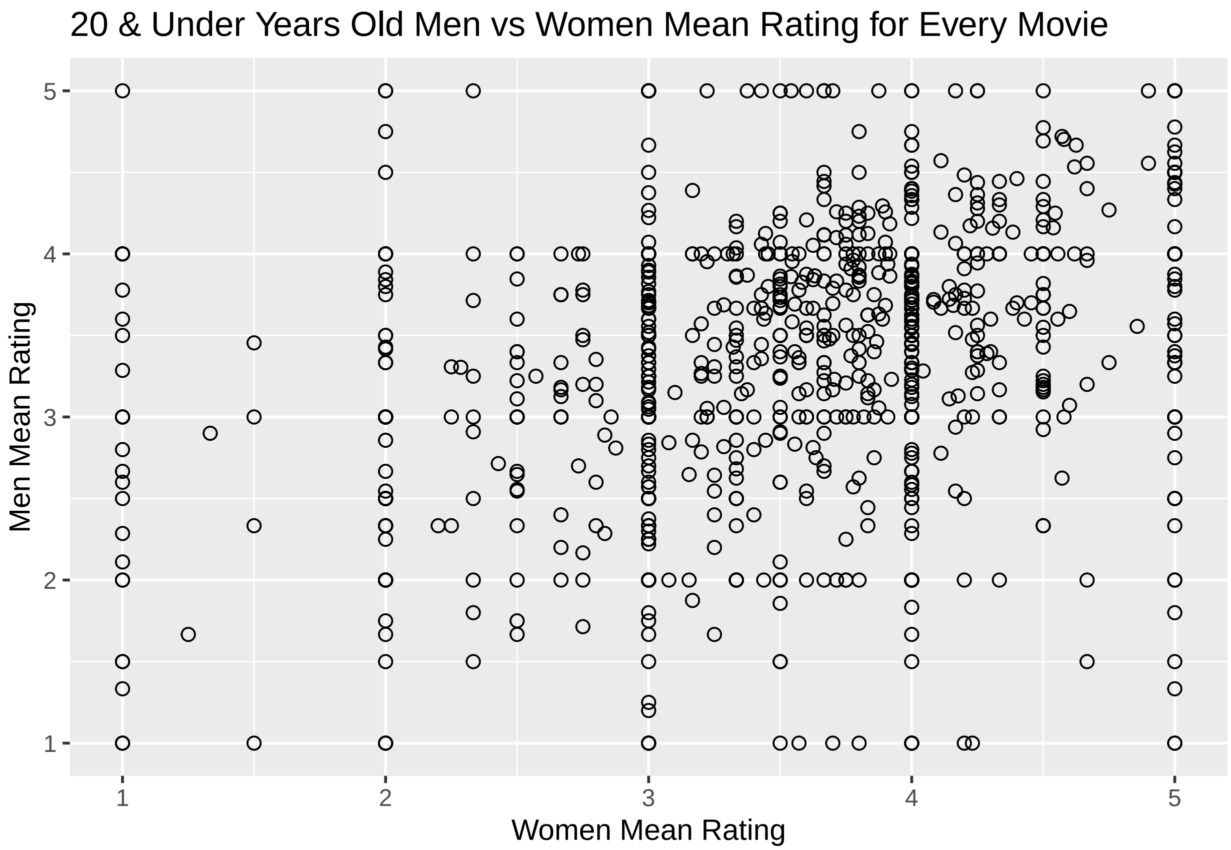


```
# men and women 50+ correlation
man_women_over_50 = cor(gender_movie_rating_age_50$rating_F, gender_movie_rating_age_50$rating_M)
man_women_over_50
```

```
## [1] 0.2924418
```

```
# men and women 20 and under scatter plot
movie_avg_rating_gend_age_F_2 = movie_avg_rating_gend_age[movie_avg_rating_gend_age$gender == 'F'& movie
movie_avg_rating_gend_age_M_2 = movie_avg_rating_gend_age[movie_avg_rating_gend_age$gender == 'M' & movi

names(movie_avg_rating_gend_age_F_2)[4] <- "rating_F"
names(movie_avg_rating_gend_age_M_2)[4] <- "rating_M"
```

```r
gender_movie_rating_age_20= merge(movie_avg_rating_gend_age_F_2,movie_avg_rating_gend_age_M_2,by="movie

ggplot(gender_movie_rating_age_20, aes(x=rating_F, y=rating_M)) + geom_point(size=2, shape=1)+
  labs(title="20 & Under Years Old Men vs Women Mean Rating for Every Movie",x="Women Mean Rating", y="
```



```r
# men and women 20 and under correlation
man_women_over_20 = cor(gender_movie_rating_age_20$rating_F, gender_movie_rating_age_20$rating_M)
man_women_over_20
```

```
## [1] 0.3083634
```