Lab 3. Data Exploration

Initial exploration of data types and dimensions

```
dim(MovieLense)
## [1]  943 1664
slotNames(MovieLense)
## [1] "data"      "normalize"
class(MovieLense@data)
## [1] "dgCMatrix"
## attr(,"package")
## [1] "Matrix"
dim(MovieLense@data)
## [1]  943 1664
```

Exploring values of ratings

```
vector_ratings <- as.vector(MovieLense@data)
unique(vector_ratings) # what are unique values of ratings
## [1] 5 4 0 3 1 2
table_ratings <- table(vector_ratings) # what is the count of each rating value
table_ratings
## vector_ratings
##       0       1       2       3       4       5
## 1469760    6059   11307   27002   33947   21077
```

Visualize the rating:

```
vector_ratings <- vector_ratings[vector_ratings != 0] # rating == 0 are NA values
vector_ratings <- factor(vector_ratings)

qplot(vector_ratings) +
  ggtitle("Distribution of the ratings")
```

Exploring viewings of movies:

```
views_per_movie <- colCounts(MovieLense) # count views for each movie

table_views <- data.frame(movie = names(views_per_movie),
```

```
                            views = views_per_movie) # create dataframe of views
table_views <- table_views[order(table_views$views,
                                 decreasing = TRUE), ] # sort by number of views

ggplot(table_views[1:6, ], aes(x = movie, y = views)) +
  geom_bar(stat="identity") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
ggtitle("Number of views of the top movies")
```

Exploring average ratings:

```
average_ratings <- colMeans(MovieLense)

qplot(average_ratings) +
  stat_bin(binwidth = 0.1) +
  ggtitle("Distribution of the average movie rating")
```

```
average_ratings_relevant <- average_ratings[views_per_movie > 100]
qplot(average_ratings_relevant) +
  stat_bin(binwidth = 0.1) +
  ggtitle(paste("Distribution of the relevant average ratings"))
```

Visualizing the matrix:

```
image(MovieLense, main = "Heatmap of the rating matrix") # hard to read-too many dimensio
ns
```

```
image(MovieLense[1:10, 1:15], main = "Heatmap of the first rows and columns")
```

Visualize most relevant users/movies only:

```
min_n_movies <- quantile(rowCounts(MovieLense), 0.99)
min_n_users <- quantile(colCounts(MovieLense), 0.99)
min_n_movies
##    99%
```

```
## 440.96
```

```
min_n_users
```

```
##    99%
## 371.07
```

```
image(MovieLense[rowCounts(MovieLense) > min_n_movies,
                 colCounts(MovieLense) > min_n_users],
main = "Heatmap of the top users and movies")
```