Lab 5. ITEM-based Collaborative Filtering

The starting point is a rating matrix in which rows correspond to users and columns correspond to items.

Defining training/test sets

```
which_train <- sample(x = c(TRUE, FALSE),
                      size = nrow(ratings_movies),
                      replace = TRUE,
                      prob = c(0.8, 0.2))
head(which_train)
```

```
## [1]  TRUE  TRUE  TRUE FALSE  TRUE FALSE
```

```
recc_data_train <- ratings_movies[which_train, ]
recc_data_test <- ratings_movies[!which_train, ]


which_set <- sample(x = 1:5,
                    size = nrow(ratings_movies),
                    replace = TRUE)
for(i_model in 1:5) {
  which_train <- which_set == i_model
  recc_data_train <- ratings_movies[which_train, ]
  recc_data_test <- ratings_movies[!which_train, ]
  # build the recommender
}
```

Building the recommendation model

```
recommender_models <- recommenderRegistry$get_entries(dataType ="realRatingMatrix")
recommender_models$IBCF_realRatingMatrix$parameters
```

```
## $k
## [1] 30
##
## $method
## [1] "Cosine"
##
## $normalize
## [1] "center"
##
```

```
## $normalize_sim_matrix
## [1] FALSE
##
## $alpha
## [1] 0.5
##
## $na_as_zero
## [1] FALSE
##
## $minRating
## [1] NA
```

```
recc_model <- Recommender(data = recc_data_train,
                          method = "IBCF",
                          parameter = list(k = 30))


recc_model
```

```
## Recommender of type 'IBCF' for 'realRatingMatrix'
## learned using 111 users.
```

```
class(recc_model)
```

```
## [1] "Recommender"
## attr(,"package")
## [1] "recommenderlab"
```

Exploring the recommender model:

```
model_details <- getModel(recc_model)
model_details$description
```

```
## [1] "IBCF: Reduced similarity matrix"
```

```
model_details$k
```

```
## [1] 30
```

```
class(model_details$sim) # this contains a similarity matrix
```

```
## [1] "dgCMatrix"
## attr(,"package")
## [1] "Matrix"
```

```
dim(model_details$sim)
```

```
## [1] 332 332
```

```
n_items_top <- 20
image(model_details$sim[1:n_items_top, 1:n_items_top],
```

```
      main = "Heatmap of the first rows and columns")
```

```
row_sums <- rowSums(model_details$sim > 0)
table(row_sums)
## row_sums
##  30
## 332
col_sums <- colSums(model_details$sim > 0)
qplot(col_sums) + stat_bin(binwidth = 1) + ggtitle("Distribution of the column count")
## stat_bin: binwidth defaulted to range/30. Use 'binwidth = x' to adjust this.
```

Which movies have the most elements:

```
which_max <- order(col_sums, decreasing = TRUE)[1:6]
rownames(model_details$sim)[which_max]
## [1] "Sling Blade (1996)"         "Usual Suspects, The (1995)"
## [3] "Fargo (1996)"               "Vertigo (1958)"
## [5] "Stargate (1994)"            "Godfather, The (1972)"
```

Applying recommender system on the dataset:

```
n_recommended <- 6 # the number of items to recommend to each user


recc_predicted <- predict(object = recc_model, newdata = recc_data_test, n = n_recommende
d)
recc_predicted
## Recommendations as 'topNList' with n = 6 for 449 users.
```

Explore the results:

```
class(recc_predicted)
## [1] "topNList"
## attr(,"package")
## [1] "recommenderlab"
slotNames(recc_predicted)
## [1] "items"      "itemLabels" "n"
recc_user_1 <- recc_predicted@items[[1]] # recommendation for the first user
movies_user_1 <- recc_predicted@itemLabels[recc_user_1]
```

```
movies_user_1
```

```
## [1] "Schindler's List (1993)"          "Secrets & Lies (1996)"
## [3] "Trainspotting (1996)"             "Deer Hunter, The (1978)"
## [5] "L.A. Confidential (1997)"         "Manchurian Candidate, The (1962)"
```

```
recc_matrix <- sapply(recc_predicted@items,
                      function(x){ colnames(ratings_movies)[x] }) # matrix with the recom
mendations for each user
dim(recc_matrix)
```

```
## [1]   6 449
```

```
recc_matrix[, 1:4]
```

```
##      [,1]                                [,2]
## [1,] "Schindler's List (1993)"           "Babe (1995)"
## [2,] "Secrets & Lies (1996)"             "Usual Suspects, The (1995)"
## [3,] "Trainspotting (1996)"              "Taxi Driver (1976)"
## [4,] "Deer Hunter, The (1978)"           "Blade Runner (1982)"
## [5,] "L.A. Confidential (1997)"          "Welcome to the Dollhouse (1995)"
## [6,] "Manchurian Candidate, The (1962)" "Silence of the Lambs, The (1991)"
##      [,3]
## [1,] "Batman Forever (1995)"
## [2,] "Stargate (1994)"
## [3,] "Star Trek III: The Search for Spock (1984)"
## [4,] "Tin Cup (1996)"
## [5,] "Courage Under Fire (1996)"
## [6,] "Dumbo (1941)"
##      [,4]
## [1,] "Strictly Ballroom (1992)"
## [2,] "L.A. Confidential (1997)"
## [3,] "Cold Comfort Farm (1995)"
## [4,] "12 Angry Men (1957)"
## [5,] "Vertigo (1958)"
## [6,] "Room with a View, A (1986)"
```

Identify the most recommended movies:

```
number_of_items <- factor(table(recc_matrix))


chart_title <- "Distribution of the number of items for IBCF"
qplot(number_of_items) + ggtitle(chart_title)
```

```
number_of_items_sorted <- sort(number_of_items, decreasing = TRUE)
number_of_items_top <- head(number_of_items_sorted, n = 4)
table_top <- data.frame(names(number_of_items_top),
                        number_of_items_top)
table_top
```

```
##                                           names.number_of_items_top.
## Mr. Smith Goes to Washington (1939) Mr. Smith Goes to Washington (1939)
## Babe (1995)                                                 Babe (1995)
## Maltese Falcon, The (1941)                 Maltese Falcon, The (1941)
## L.A. Confidential (1997)                     L.A. Confidential (1997)
##                                     number_of_items_top
## Mr. Smith Goes to Washington (1939)                  55
## Babe (1995)                                          38
## Maltese Falcon, The (1941)                           35
```

```
## L.A. Confidential (1997)
```