HW3

Sheng Zhang

March 28, 2017

```
## Q1

# Read in data
book_ratings <- read.csv("./BX-Book-Ratings 2.csv", header = TRUE, sep = ";")
# book_ratings <- read.csv("./Spring 2017/Machine Learning/Rmd files/HW3/BX-Book-Ratings 2.csv", header

# Find top 100 active users
sort_user <- sort(table(book_ratings$User.ID), decreasing = TRUE)
top100 <- dimnames(sort_user[1:100])
book_ratings_top100 <- book_ratings[book_ratings$User.ID %in% top100[[1]],]

# Count unique books
unique_book <- unique(book_ratings_top100$ISBN)
length(unique_book)
```

```
## [1] 113421
```

```
# Count unique book ratings
dim(book_ratings_top100)
```

```
## [1] 203554      3
```

1) There are 113421 unique books and 203554 unique book ratings in the reduced dataset.

```
## Q2

set.seed(1)
book.train.id <- sample(1:nrow(book_ratings_top100), 100000)
book.train <- book_ratings_top100[book.train.id, ]
book.test <- book_ratings_top100[-book.train.id, ]
```

```
## Q3

library(tidyr)
```

```
## Warning: package 'tidyr' was built under R version 3.3.3
```

```
train.matrix <- spread(book.train, ISBN, Book.Rating)
test.matrix <- spread(book.test, ISBN, Book.Rating)
```

```
## Q4

row_means <- rowMeans(train.matrix[,-1], na.rm = TRUE)
train.matrix_pred1 <- train.matrix[-1]
```

```
for (i in 1:nrow(train.matrix))
{
  train.matrix_pred1[i, is.na(train.matrix_pred1[i,])] <- row_means[i]
}

SVD_pred1 <- svd(train.matrix_pred1)
train.matrix_pred2 <- SVD_pred1$u[, 1:10] %*% diag(SVD_pred1$d)[1:10, 1:10] %*% t(SVD_pred1$v [, 1:10])

train.matrix_pred2 <- as.data.frame(train.matrix_pred2)
colnames(train.matrix_pred2) <- colnames(train.matrix_pred1[,-1])
common_ISBN <- intersect(colnames(train.matrix[,-1]),colnames(test.matrix[,-1]))

common_train_matrix <- train.matrix_pred2[,common_ISBN]
common_test_matrix <- test.matrix[,common_ISBN]
diff_matrix <- common_test_matrix - common_train_matrix
mse_1 <- sum(diff_matrix^2, na.rm = TRUE)/length(which(!is.na(diff_matrix)))
mse_1
```

```
## [1] 7.380393
```

4) The MSE of the SVD method is about 7.380.

```
## Q5

train.matrix_pred3 <- train.matrix[,-1]

train.matrix_pred3[is.na(train.matrix_pred3)] <- train.matrix_pred2[is.na(train.matrix_pred3)]

SVD_pred3 <- svd(train.matrix_pred3)
train.matrix_pred4 <- SVD_pred3$u[, 1:10] %*% diag(SVD_pred3$d)[1:10, 1:10] %*% t(SVD_pred3$v [, 1:10])

train.matrix_pred4 <- as.data.frame(train.matrix_pred4)
colnames(train.matrix_pred4) <- colnames(train.matrix_pred2)

common_train_matrix_2 <- train.matrix_pred4[,common_ISBN]
diff_matrix_2 <- common_test_matrix - common_train_matrix_2
mse_2 <- sum(diff_matrix_2^2, na.rm = TRUE)/length(which(!is.na(diff_matrix_2)))
mse_2
```

```
## [1] 7.377347
```

5) The MSE of the SVD method with 2 iterations is now about 7.378. The recommendation system did not improve much.

6) First, I could try to use more iterations of the SVD method to try to improve the predictive performance. Second, I could specify a higher number of latent factors in the SVD process to retain more information about the original matrix to yield a better prediction. Third, I could use additional methods such as decision trees together with SVD to see if the performance will improve.