Introduction to Recommender Systems

Code ▾

Getting Started

- Open a new R Script
- Install (if necessary) and load the *data.table* and *RANN* package

Import data

- Import the the books dataset as a **data.table** from https://github.com/zygmuntz/goodbooks-10k/raw/master/books.csv and assign it the variable *books*
- Import the the ratings dataset as a **data.table** from https://github.com/zygmuntz/goodbooks-10k/raw/master/ratings.csv and assign it the variable *ratings*
- Import the the book to tags dataset as a **data.table** from https://github.com/zygmuntz/goodbooks-10k/raw/master/book_tags.csv and assign it the variable *book_tags*
- Import the tags lookup dataset as a **data.table** from https://github.com/zygmuntz/goodbooks-10k/raw/master/tags.csv and assign it the variable *tags*

Processing the data

- Filter *book_tags* and keep only the top 3 tags (by counts), for each goodreads_book_id

- Run the following code to generate indicator columns for a combination of genre types. Explore the main_tags data.frame.

Hide

```
#Get the main categories from tags for each book

main_tags_labels = c('romance','fiction','young-adult','fantasy','science-fiction',
'children','best','covers','non-fiction', 'history','mystery',
'paranormal','love','horror','historical','gay','sci-fi',
'historical-fiction','nonfiction','series','literature',  'contemporary',
'thriller','women','novels','suspense','classics' ,'graphic-novels',
'historical-romance', 'christian')

main_tags = merge(x=book_tags,y=tags,by="tag_id")
main_tags = main_tags[,.(tags = paste(tag_name,collapse=",")),.(goodreads_book_id)]

for(j in main_tags_labels){
  set(main_tags,j = j,value = grepl(x = main_tags$tags,pattern = j)*1)

  print(j)
}
main_tags[,tags:=NULL]
```

- Add the following columns to the *books* data.table. 1. *primary_author*: The name of the first author of a book. 2. *english*: A binary (0/1) indicator for the letters “en” in a books language_code.
- Remove all other columns except book_id,work_id,goodreads_book_id,primary_author, original_publication_year,english,average_rating,ratings_1,ratings_2, ratings_3,ratings_4,ratings_5 from books

- Join *main_tags* data to *books* on *goodreads_book_id*

Exploratory Data Analysis

- Create a new books data.table called *books_wide* by “melting” the genre columns.

Use the books_wide data set for the following

- Calculate the average book rating by author

- Calculate the average book rating, and number of published book by author in each genre

- Calculate the three top rated authors in each genre

- Calculate the best genre of each author

Content-Based Filtering

- Create *books_cb* a copy of *books*, and delete the primary_author, goodreads_book_id and work_id column

- Normalize (subtract the min, divide by the range) all remaining numeric columns with the exception of *book_id*

- Randomly assign 500 unique user_ids from *ratings* into a variable to *test_user*. Assign the remaining to *train_user*

- Create *user_affinity* a data.table of the high rated books by user_id

- For each user in *test_user*, find the top 5 books(book_id) that are most related to their highest rated book based on the **Cosine Similarity** metric

- Print recommendation for the test_users!

Collaborative Filtering

- Create a user-item matrix ratings matrix using the *ratings* dataset called *user_item_mat*

- Create a subset of *user_item_mat*, for user_id's only in *test_users*

- Remove rows correspondig to *test_users* from *user_item_mat_test*

- In *user_item_mat* and *user_item_mat_test*, replace all NA values with 0

- For the first 10 rows (user) in *user_item_mat_test*:

- 1. Use the **Euclidean Distance** metric to find the 5 most similar users in *user_item_mat*.
- 2. Find the highest rated book_id's for the 5 most similar users

- Print recommendation for the 10 test_users!