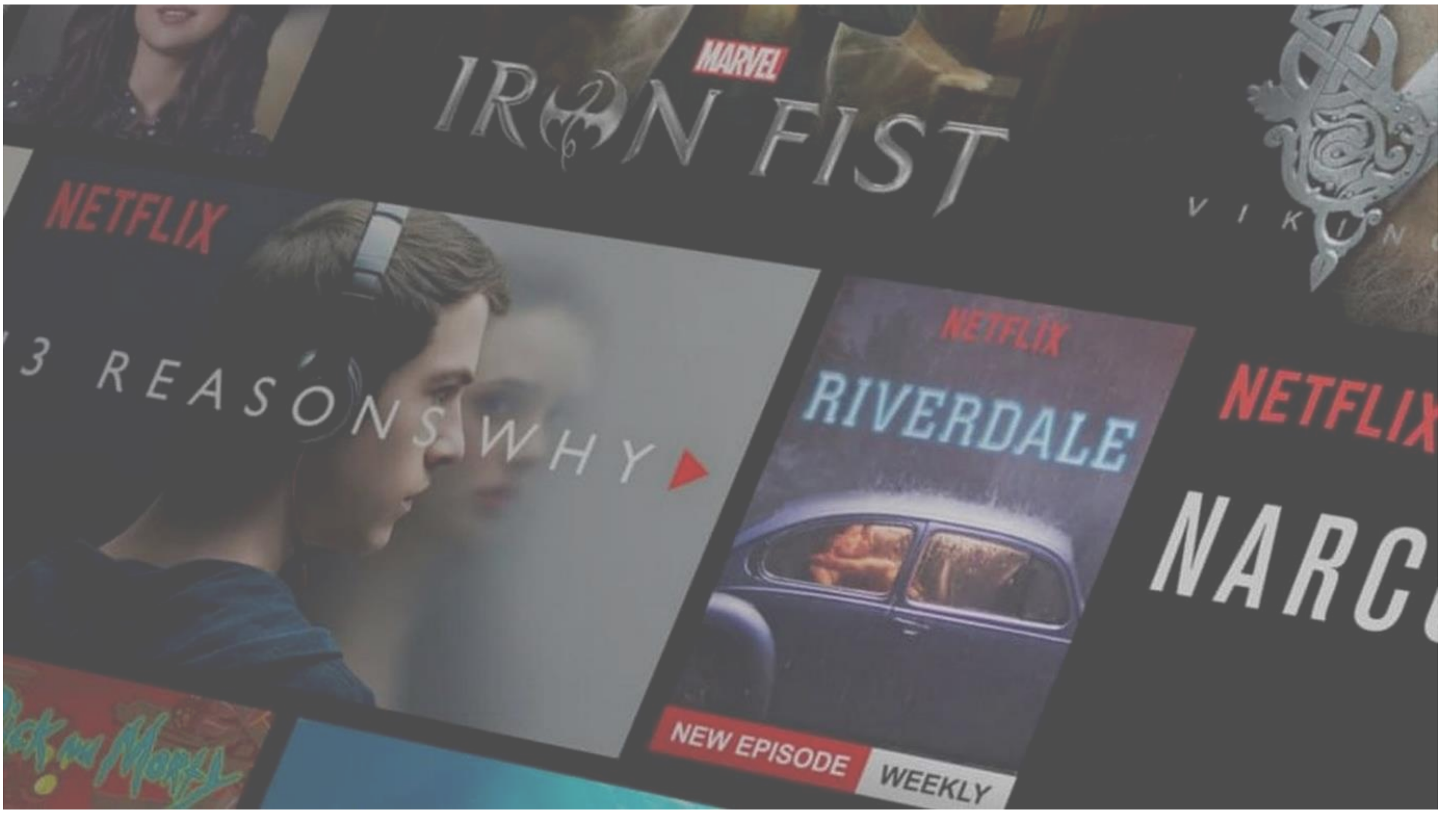


Free Movie Recommendation System

User Guide

Version 1.0 – 21 October 2020

Creative Team Name

Table of Contents

Introduction

About this Document

The purpose of this document is to define the functionality and operation of the Movie Recommendation Script delivered by the Creative Team Name. This document serves as the sole reference guide for the usage of the Movie Recommendation Script. Further information regarding the software specifications can be found in the Software Specifications Document delivered by Creative Team Name.

Required Data Structure

Three components are required for the operation of the Movie Recommendation System

- The MovieRecommendationSystem.R script
- Data for Movies
- Data for user ratings

It is important to note that the format for the Movies and Ratings data must come as a .csv file and be structured accordingly.

The movies data *must* be structured as such: it contains three columns named (case sensitive) *movieId*, *title* and *genres*.

- movieId: a unique identifying number or string for a movie. If there are duplicate movieIds, the script will remove any duplicates.
- title: The title of the movie.
- genres: the genres which the movie belongs to, delimited by the same delimiter. By default, the delimiter is the pipe ("|") character, but this can be modified as long as the delimiter is consistent. For example, two different rows having the genres: Drama|Thriller" and "Comedy|Fantasy|Children" is fine. Equally, "Drama.Thriller" and "Comedy.Fantasy.Children" is also fine, assuming the delimiter is specified to be the period character ("."). Including a mix of delimiters such as "Drama|Thriller" and Comedy.Fantasy.Children" will not work. See the provided movies.csv file for more examples.

The ratings data must be structured as such: it has four columns named (case sensitive) *userId*, *movieId*, *rating* and *timestamp*.

- userId: The ID of the user watching a given movie.
- movieId: The ID of the movie the userId watched.
- rating: The rating the user gave the movie, on a scale from 0.5 to 5. Values which are 0 or blank are assumed “missing” values where the user watched the movie but gave it no rating. Any values greater than 5 or less than 0 are removed.
- timestamp: Timestamp is an arbitrary piece of metadata which the recommendation system does not use. In the dataset provided, this value is an integer value representing seconds since midnight Coordinated Universal Time (UTC) of January 1, 1970. This value can be anything, however, the value will not influence the performance or results of the movie recommendation system.

Installing / Starting the System

First-time Users

An Internet Connection is also required when running the script for the first time. The script has a few dependant libraries which it will automatically download and install if its not already installed in the system.

Installing the Prerequisite System Requirements

FMRS is written in R, and thus requires an installation of the R language.

For Windows: https://cran.r-project.org/bin/windows/base/

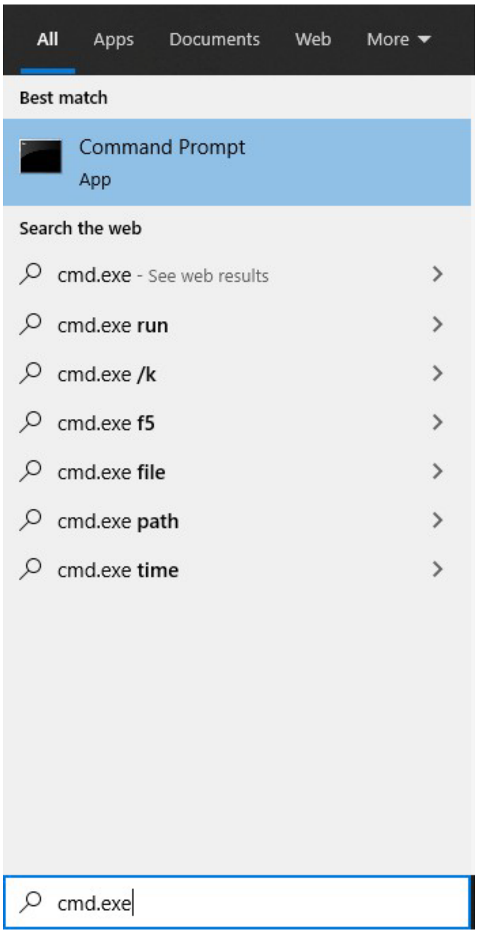
For MacOS: https://cran.r-project.org/ (Download R for (Mac) OS X

Version R-4.0.2 was used at the time of the development of this document.

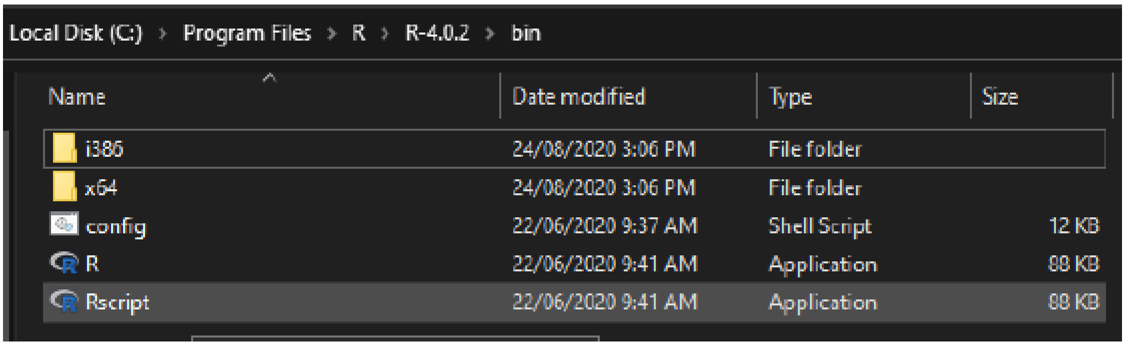
Step by Step Instructions

Windows Instructions

1. The script is executed via a system command. Open the command prompt and search for "cmd.exe" in the start menu. (Ensure that R is installed)



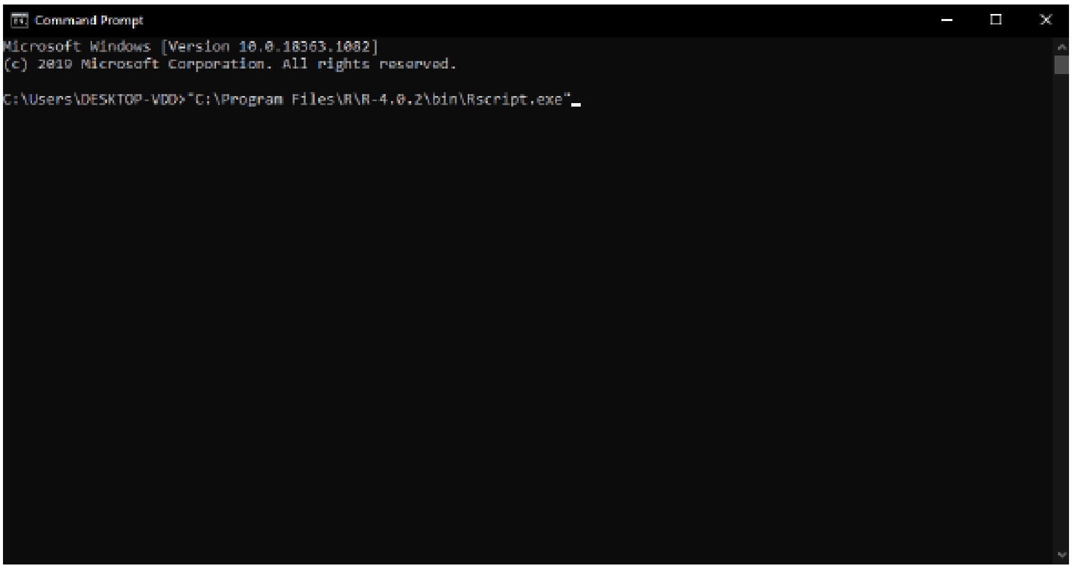
2. Locate where R has installed the Rscript.exe file. (This is usually under `c:\ProgramFiles\R\R-Rversion\bin`):



Local Disk (C:) › Program Files › R › R-4.0.2 › bin

Name	Date modified	Type	Size
i386	24/08/2020 3:06 PM	File folder	
x64	24/08/2020 3:06 PM	File folder	
config	22/06/2020 9:37 AM	Shell Script	12 KB
R	22/06/2020 9:41 AM	Application	88 KB
Rscript	22/06/2020 9:41 AM	Application	88 KB

3. Enter the located file path into the command prompt and surround it with double quotes, as such:



```
Microsoft Windows [Version 10.0.18363.1082]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\DESKTOP-VDD>"C:\Program Files\R\R-4.0.2\bin\Rscript.exe"_
```

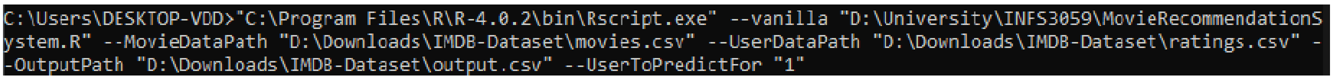
4. Enter --`vanilla` after the recently added file path of the Rscript.exe command, then provide the full path to the MovieRecommendationSystem.R file, again surrounded by double quotes, e.g. “D:\University\INFS3059\MovieRecommendationSystem.R”.

5. After this, the relevant arguments will be required for this script to function (see screenshot below for an example). All arguments must be surrounded by double quotes, the four required arguments are as follows:

- --MovieDataPath: the full path to the movies.csv data file. For example, “D:\Downlaods\IMDB-Dataset\movies.csv”. If this path is invalid, the script will throw an error.
- --userDataPath: the full path to the ratings.csv data file. For example, “D:\Downloads\IMDB-Dataset\ratings.csv”. If this path is invalid, the script will throw an error.
- --OutputPath: the path to where the outpuuted predictions file will be written to. For example, “D:\Downlaods\IMDB-Dataset\output.csv”. If this path does not exist, it will be created. Ensure that the intended directory path contains the required valid permissions, e.g. must be able to write to, otherwise the script will throw an error.

- --UserToPredictFor: A value which matches a userId in the list of unique users from UserDataPath. For example, in `ratings.csv`, userIds range from 1 to 669, meaning a valid value is any number from 1 to 669. This is the user which will have predictions made for them, e.g. a list of movies they may enjoy but have not already watched. If the UserToPredictFor value is invalid, the script will throw and error.

An example of the minimal required arguments:

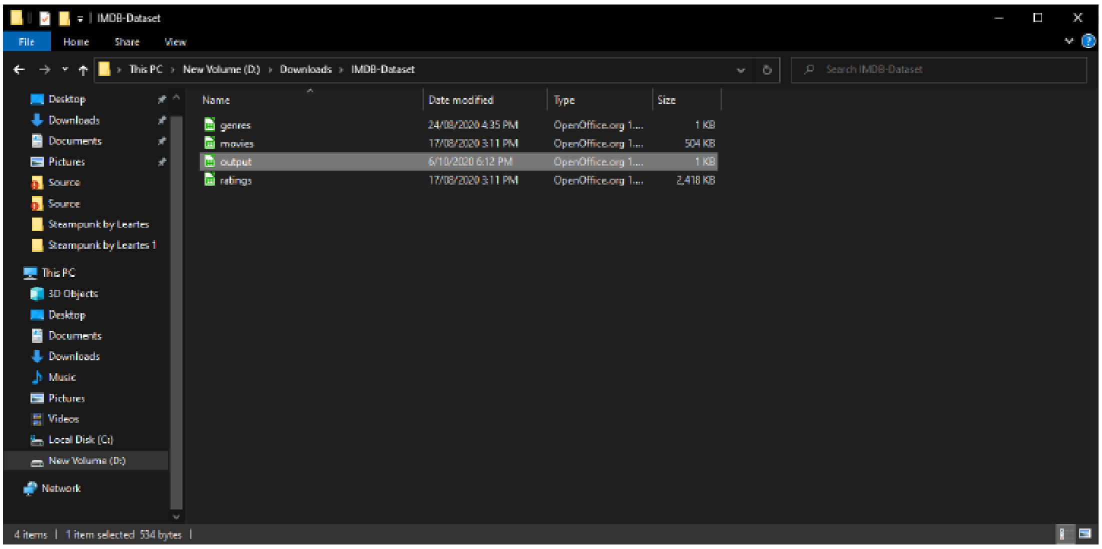


```
C:\Users\DESKTOP-VDD>"C:\Program Files\R\R-4.0.2\bin\Rscript.exe" --vanilla "D:\University\INFS3059\MovieRecommendationS
ystem.R" --MovieDataPath "D:\Downloads\IMDB-Dataset\movies.csv" --UserDataPath "D:\Downloads\IMDB-Dataset\ratings.csv" -
-OutputPath "D:\Downloads\IMDB-Dataset\output.csv" --UserToPredictFor "1"
```

6. (OPTIONAL) Additional arguments may also be added to the command line if you wish to adjust the final output:
   - --NoMoviesToPred: The number of movies to predict for a user. Default is 10.
   - --ContentFilter: A value which must either be “ibcf” or “ubcf”. Default is “ibcf”. Refer to the Software Specifications Document for more detail.
   - --NumNearestItems: The number of most alike movies to search for when weighting a user preference. If the value is less than 2, it will be set 2. Defaults to 1000. Prints a warning if greater than total number of movies in the dataset.

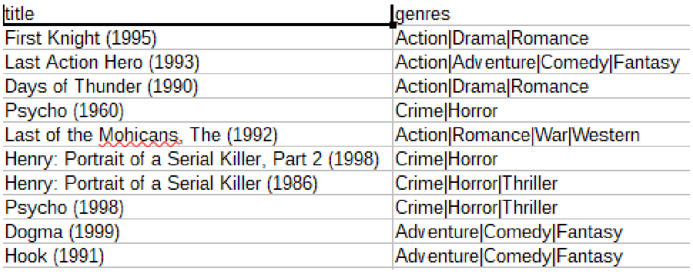
7. Simply press enter after entering in the required arguments, this will launch the script. *Note: The execution time for this script is proportional to the size of the data; execution time will be shorter if the data set is smaller and longer if the amount of data is larger. The provided example data will run for around 15 minutes.*

8. Once the command prompt finishes executing the script, locate the path specified in OutputPath to view the file called `output.csv` which presents the predicted movies for that user (by default 10 movies).



Name	Date modified	Type	Size
genres	24/08/2020 4:35 PM	OpenOffice.org 1....	1 KB
movies	17/08/2020 3:11 PM	OpenOffice.org 1....	504 KB
output	6/10/2020 6:12 PM	OpenOffice.org 1....	1 KB
ratings	17/08/2020 3:11 PM	OpenOffice.org 1....	2,418 KB

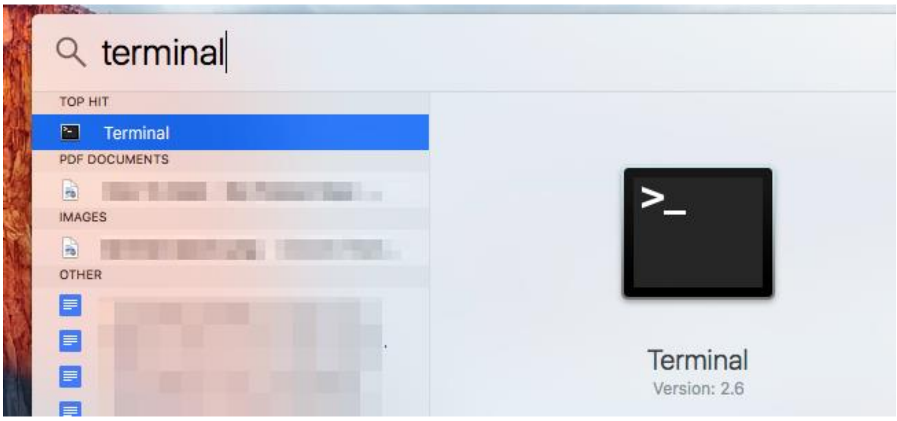
9. The output.csv file can be opened with programs like Excel or LibreOfficeCalc to view the outputted results.



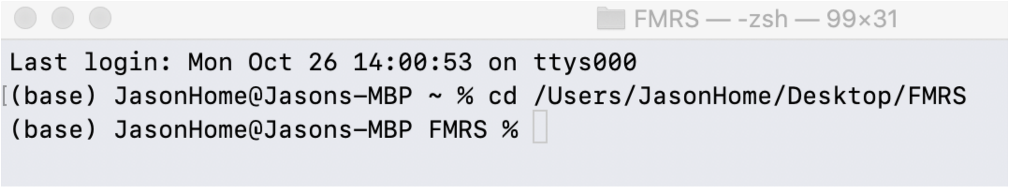
title	genres
First Knight (1995)	Action|Drama|Romance
Last Action Hero (1993)	Action|Adventure|Comedy|Fantasy
Days of Thunder (1990)	Action|Drama|Romance
Psycho (1960)	Crime|Horror
Last of the Mohicans, The (1992)	Action|Romance|War|Western
Henry: Portrait of a Serial Killer, Part 2 (1998)	Crime|Horror
Henry: Portrait of a Serial Killer (1986)	Crime|Horror|Thriller
Psycho (1998)	Crime|Horror|Thriller
Dogma (1999)	Adventure|Comedy|Fantasy
Hook (1991)	Adventure|Comedy|Fantasy

MacOS Instructions

1. The script is executed via a system command. Open Terminal by performing a Spotlight Search. (Ensure that R is installed)

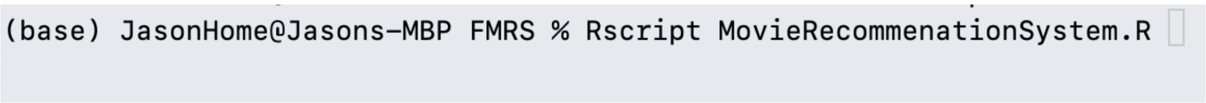


2. Locate the folder location of the MovieRecommendationSystem.R script and use the Change Directory (cd) command in Terminal to use the location of the folder. See image below as reference.



```
Last login: Mon Oct 26 14:00:53 on ttys000
[(base) JasonHome@Jasons-MBP ~ % cd /Users/JasonHome/Desktop/FMRS
(base) JasonHome@Jasons-MBP FMRS %
```

3. Enter “RScript MovieRecommendationSystem.R” into terminal.



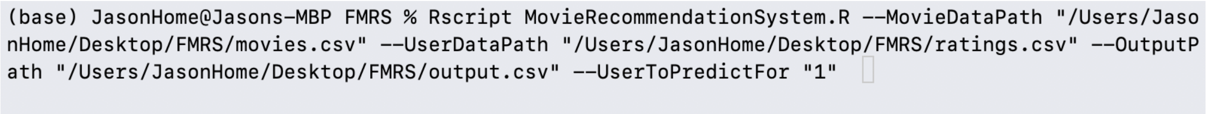
```
(base) JasonHome@Jasons-MBP FMRS % Rscript MovieRecommenationSystem.R
```

4. After this, the relevant arguments will be required for this script to function (see screenshot below for an example). All arguments must be surrounded by double quotes, the four required arguments are as follows:

    - --MovieDataPath: the full path to the movies.csv data file. For example, "/Users/JasonHome/Desktop/FMRS/movies.csv”. If this path is invalid, the script will throw an error.
    - --UserDataPath: the full path to the ratings.csv data file. For example, "/Users/JasonHome/Desktop/FMRS/ratings.csv". If this path is invalid, the script will throw an error.
    - --OutputPath: the path to where the outputted predictions file will be written to. For example, "/Users/JasonHome/Desktop/FMRS/output.csv". If this path does not exist, it will be created. Ensure that the intended directory path contains the required valid permissions, e.g. must be able to write to, otherwise the script will throw an error.

    - --UserToPredictFor: A value which matches a userId in the list of unique users from UserDataPath. For example, in `ratings.csv`, userIds range from 1 to 669, meaning a valid value is any number from 1 to 669. This is the user which will have predictions made for them, e.g. a list of movies they may enjoy but have not already watched. If the UserToPredictFor value is invalid, the script will throw and error.

An example of the minimal required arguments:

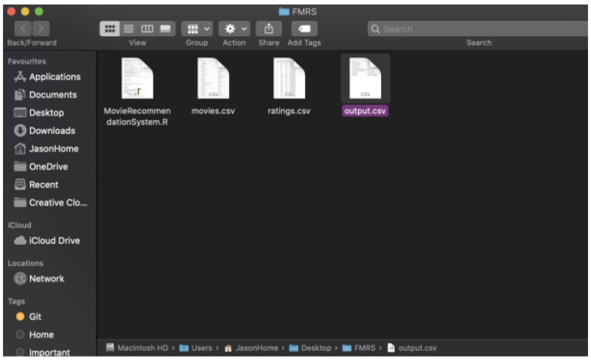


```
(base) JasonHome@Jasons-MBP FMRS % Rscript MovieRecommendationSystem.R --MovieDataPath "/Users/Jaso
nHome/Desktop/FMRS/movies.csv" --UserDataPath "/Users/JasonHome/Desktop/FMRS/ratings.csv" --OutputP
ath "/Users/JasonHome/Desktop/FMRS/output.csv" --UserToPredictFor "1"
```

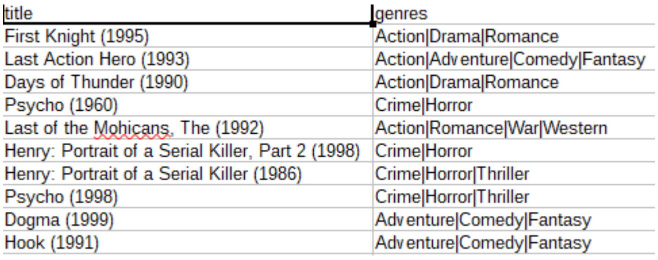
5. (OPTIONAL) Additional arguments may also be added to the command line if you wish to adjust the final output:
    - --NoMoviesToPred: The number of movies to predict for a user. Default is 10.
    - --ContentFilter: A value which must either be “ibcf” or “ubcf”. Default is “ibcf”. Refer to the Software Specifications Document for more detail.
    - --NumNearestItems: The number of most alike movies to search for when weighting a user preference. If the value is less than 2, it will be set 2. Defaults to 1000. Prints a warning if greater than total number of movies in the dataset.

6. Simply press enter after entering in the required arguments, this will launch the script. *Note: The execution time for this script is proportional to the size of the data; execution time will be shorter if the data set is smaller and longer if the amount of data is larger. The provided example data will run for around 15 minutes.*

7. Once the command prompt finishes executing the script, locate the path specified in OutputPath to view the file called `output.csv` which presents the predicted movies for that user (by default 10 movies).



8. The output.csv file can be opened with programs like Excel or LibreOfficeCalc to view the outputted results.



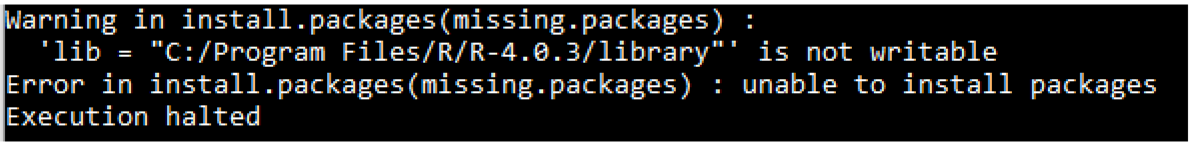
title	genres
First Knight (1995)	Action|Drama|Romance
Last Action Hero (1993)	Action|Adventure|Comedy|Fantasy
Days of Thunder (1990)	Action|Drama|Romance
Psycho (1960)	Crime|Horror
Last of the Mohicans, The (1992)	Action|Romance|War|Western
Henry: Portrait of a Serial Killer, Part 2 (1998)	Crime|Horror
Henry: Portrait of a Serial Killer (1986)	Crime|Horror|Thriller
Psycho (1998)	Crime|Horror|Thriller
Dogma (1999)	Adventure|Comedy|Fantasy
Hook (1991)	Adventure|Comedy|Fantasy

Errors & Trouble Shooting

Error in Installing Packages

<u>Issue:</u>

After executing the script command, if you encounter an issue where installing to a certain file location is not writable like the image below, perform the following steps:

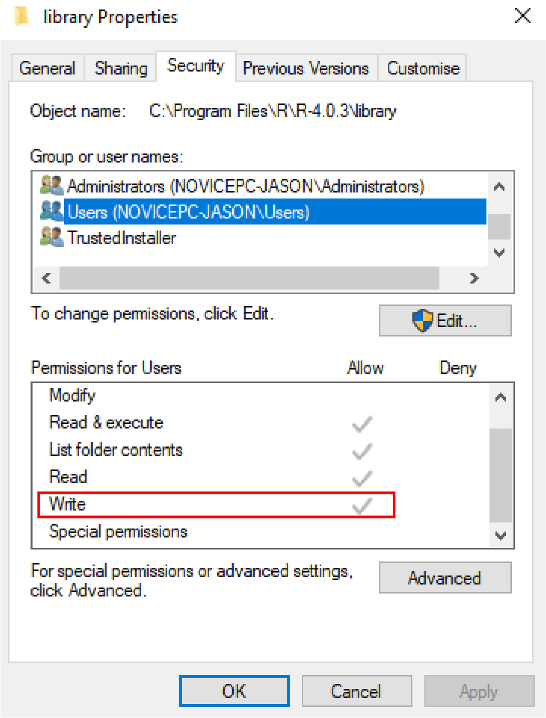


```
Warning in install.packages(missing.packages) :
  'lib = "C:/Program Files/R/R-4.0.3/library"' is not writable
Error in install.packages(missing.packages) : unable to install packages
Execution halted
```

Unable to install packages

<u>Possible Solution:</u>

1. Navigate to the file location of the installed R script library. Right click on the file and select Properties then to the Security Tab. Possible file locations:
   - For Windows: C:\Program Files\R\R-4.0.3\library
   - For MacOS: Macintosh HD/Library/Frameworks/R.framework/Versions/4.0/Resources/library
2. Ensure that the permissions for the User are allowed to *Write* to that file. See image below for reference guide for Windows and MacOS.



Object name: C:\Program Files\R\R-4.0.3\library

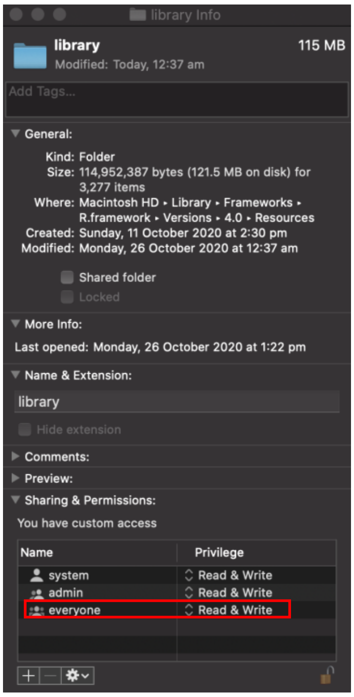
Group or user names:

- Administrators (NOVICEPC-JASON\Administrators)
- Users (NOVICEPC-JASON\Users)
- TrustedInstaller

To change permissions, click Edit.

Permissions for Users	Allow	Deny
Modify		
Read & execute	✓	
List folder contents	✓	
Read	✓	
Write	✓	
Special permissions		

For special permissions or advanced settings, click Advanced.



library 115 MB
Modified: Today, 12:37 am

General:

Kind: Folder
Size: 114,952,387 bytes (121.5 MB on disk) for 3,277 items
Where: Macintosh HD ▸ Library ▸ Frameworks ▸ R.framework ▸ Versions ▸ 4.0 ▸ Resources
Created: Sunday, 11 October 2020 at 2:30 pm
Modified: Monday, 26 October 2020 at 12:37 am

□ Shared folder
□ Locked

More Info:

Last opened: Monday, 26 October 2020 at 1:22 pm

Name & Extension:

library

□ Hide extension

Sharing & Permissions:

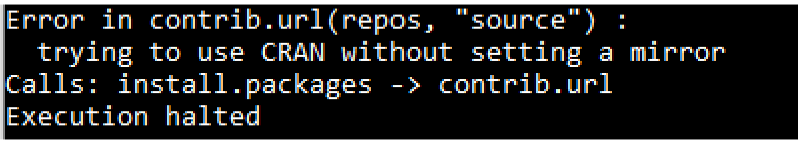
You have custom access

Name	Privilege
system	Read & Write
admin	Read & Write
everyone	Read & Write

Trying to use CRAN without setting a mirror

Issue:

After executing the script command, if you encounter an issue where trying to use CRAN without setting a mirror halts the execution like the image below, perform the following steps.

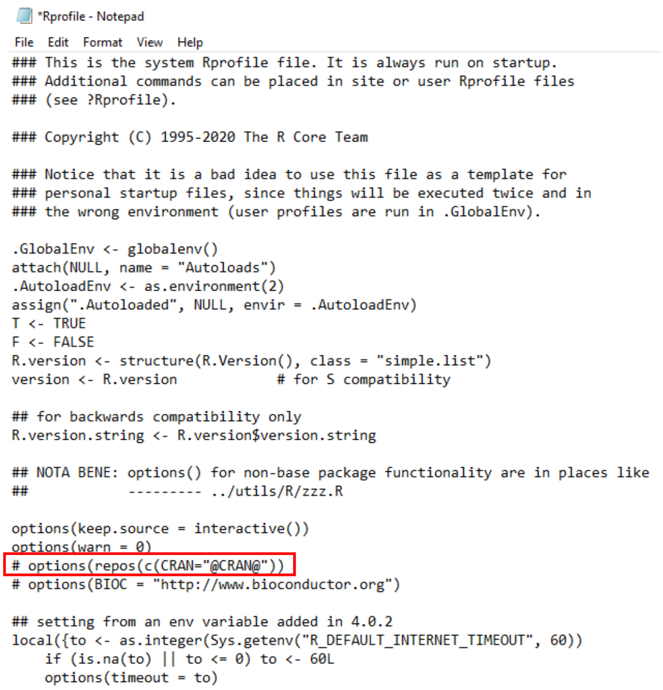


```
Error in contrib.url(repos, "source") :
  trying to use CRAN without setting a mirror
Calls: install.packages -> contrib.url
Execution halted
```

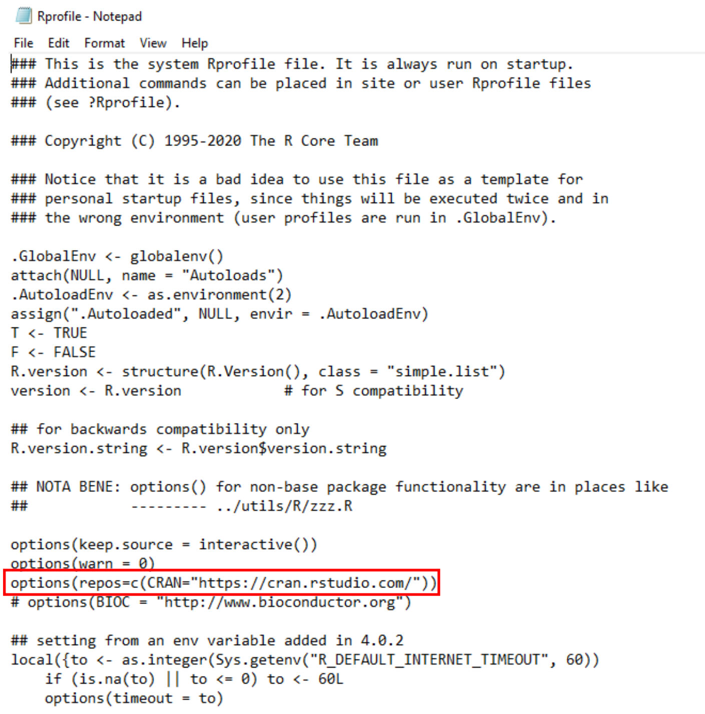
Trying to use CRAN without setting a mirror

Possible Solution:

1. Navigate to the file location of the installed R Script library and navigate to the R folder. Possible file locations:
   - For Windows: C:\Program Files\R\R-4.0.3\library\base\R
   - For MacOS: Macintosh HD/Library/Frameworks/R.framework/Versions/4.0/Resources/library/base/R
2. Once in the file location, open the file “RProfile” with any text editor.
3. Replace the line, “`# options(repos=c(CRAN=”@CRAN@”))`” with “`options(repos=c(CRAN=https://cran.rstudio.com/))`”. Note removal of “#”, see the image below for reference.



```
### This is the system Rprofile file. It is always run on startup.
### Additional commands can be placed in site or user Rprofile files
### (see ?Rprofile).

### Copyright (C) 1995-2020 The R Core Team

### Notice that it is a bad idea to use this file as a template for
### personal startup files, since things will be executed twice and in
### the wrong environment (user profiles are run in .GlobalEnv).

.GlobalEnv <- globalenv()
attach(NULL, name = "Autoloads")
.AutoloadEnv <- as.environment(2)
assign(".Autoloaded", NULL, envir = .AutoloadEnv)
T <- TRUE
F <- FALSE
R.version <- structure(R.Version(), class = "simple.list")
version <- R.version            # for S compatibility

## for backwards compatibility only
R.version.string <- R.version$version.string

## NOTA BENE: options() for non-base package functionality are in places like
##            --------- ../utils/R/zzz.R

options(keep.source = interactive())
options(warn = 0)
# options(repos=c(CRAN="@CRAN@"))
# options(BIOC = "http://www.bioconductor.org")

## setting from an env variable added in 4.0.2
local({to <- as.integer(Sys.getenv("R_DEFAULT_INTERNET_TIMEOUT", 60))
    if (is.na(to) || to <= 0) to <- 60L
    options(timeout = to)
```



```
### This is the system Rprofile file. It is always run on startup.
### Additional commands can be placed in site or user Rprofile files
### (see ?Rprofile).

### Copyright (C) 1995-2020 The R Core Team

### Notice that it is a bad idea to use this file as a template for
### personal startup files, since things will be executed twice and in
### the wrong environment (user profiles are run in .GlobalEnv).

.GlobalEnv <- globalenv()
attach(NULL, name = "Autoloads")
.AutoloadEnv <- as.environment(2)
assign(".Autoloaded", NULL, envir = .AutoloadEnv)
T <- TRUE
F <- FALSE
R.version <- structure(R.Version(), class = "simple.list")
version <- R.version            # for S compatibility

## for backwards compatibility only
R.version.string <- R.version$version.string

## NOTA BENE: options() for non-base package functionality are in places like
##            --------- ../utils/R/zzz.R

options(keep.source = interactive())
options(warn = 0)
options(repos=c(CRAN="https://cran.rstudio.com/"))
# options(BIOC = "http://www.bioconductor.org")

## setting from an env variable added in 4.0.2
local({to <- as.integer(Sys.getenv("R_DEFAULT_INTERNET_TIMEOUT", 60))
    if (is.na(to) || to <= 0) to <- 60L
    options(timeout = to)
```