

SUMMARY & USAGE LICENSE

=====

Movielens data sets were collected by the GroupLens Research Project at the University of Minnesota.

This data set consists of:

- * 100,000 ratings (1-5) from 943 users on 1682 movies.
- * Each user has rated at least 20 movies.
- * Simple demographic info for the users (age, gender, occupation, zip)

The data was collected through the MovieLens web site (movielens.umn.edu) during the seven-month period from September 19th, 1997 through April 22nd, 1998. This data has been cleaned up - users who had less than 20 ratings or did not have complete demographic information were removed from this data set. Detailed descriptions of the data file can be found at the end of this file.

Neither the University of Minnesota nor any of the researchers involved can guarantee the correctness of the data, its suitability for any particular purpose, or the validity of results based on the use of the data set. The data set may be used for any research purposes under the following conditions:

- * The user may not state or imply any endorsement from the University of Minnesota or the GroupLens Research Group.
- * The user must acknowledge the use of the data set in publications resulting from the use of the data set (see below for citation information).
- * The user may not redistribute the data without separate permission.
- * The user may not use this information for any commercial or revenue-bearing purposes without first obtaining permission from a faculty member of the GroupLens Research Project at the University of Minnesota.

If you have any further questions or comments, please contact GroupLens <grouplens-info@cs.umn.edu>.

CITATION

=====

To acknowledge use of the dataset in publications, please cite the following paper:

F. Maxwell Harper and Joseph A. Konstan. 2015. The MovieLens Datasets: History and Context. ACM Transactions on Interactive Intelligent Systems (TiiS) 5, 4, Article 19 (December 2015), 19 pages. DOI=<http://dx.doi.org/10.1145/2827872>

ACKNOWLEDGEMENTS

=====

Thanks to Al Borchers for cleaning up this data and writing the accompanying scripts.

PUBLISHED WORK THAT HAS USED THIS DATASET

=====

Herlocker, J., Konstan, J., Borchers, A., Riedl, J.. An Algorithmic Framework for Performing Collaborative Filtering. Proceedings of the

1999 Conference on Research and Development in Information Retrieval. Aug. 1999.

FURTHER INFORMATION ABOUT THE GROUPLENS RESEARCH PROJECT

=====

The GroupLens Research Project is a research group in the Department of Computer Science and Engineering at the University of Minnesota. Members of the GroupLens Research Project are involved in many research projects related to the fields of information filtering, collaborative filtering, and recommender systems. The project is lead by professors John Riedl and Joseph Konstan. The project began to explore automated collaborative filtering in 1992, but is most well known for its world wide trial of an automated collaborative filtering system for Usenet news in 1996. The technology developed in the Usenet trial formed the base for the formation of Net Perceptions, Inc., which was founded by members of GroupLens Research. Since then the project has expanded its scope to research overall information filtering solutions, integrating in content-based methods as well as improving current collaborative filtering technology.

Further information on the GroupLens Research project, including research publications, can be found at the following web site:

<http://www.grouplens.org/>

GroupLens Research currently operates a movie recommender based on collaborative filtering:

<http://www.movielens.org/>

DETAILED DESCRIPTIONS OF DATA FILES

=====

Here are brief descriptions of the data.

ml-data.tar.gz -- Compressed tar file. To rebuild the u data files do this:
 gunzip ml-data.tar.gz
 tar xvf ml-data.tar
 mku.sh

u.data -- The full u data set, 100000 ratings by 943 users on 1682 items. Each user has rated at least 20 movies. Users and items are numbered consecutively from 1. The data is randomly ordered. This is a tab separated list of
 user id | item id | rating | timestamp.
 The time stamps are unix seconds since 1/1/1970 UTC

u.info -- The number of users, items, and ratings in the u data set.

u.item -- Information about the items (movies); this is a tab separated list of
 movie id | movie title | release date | video release date |
 IMDb URL | unknown | Action | Adventure | Animation |
 Children's | Comedy | Crime | Documentary | Drama | Fantasy |
 Film-Noir | Horror | Musical | Mystery | Romance | Sci-Fi |
 Thriller | War | Western |
 The last 19 fields are the genres, a 1 indicates the movie is of that genre, a 0 indicates it is not; movies can be in several genres at once.
 The movie ids are the ones used in the u.data data set.

u.genre -- A list of the genres.

```
u.user      -- Demographic information about the users; this is a tab
              separated list of
              user id | age | gender | occupation | zip code
              The user ids are the ones used in the u.data data set.

u.occupation -- A list of the occupations.

u1.base      -- The data sets u1.base and u1.test through u5.base and u5.test
u1.test      are 80%/20% splits of the u data into training and test data.
u2.base      Each of u1, ..., u5 have disjoint test sets; this is for
u2.test      5 fold cross validation (where you repeat your experiment
u3.base      with each training and test set and average the results).
u3.test      These data sets can be generated from u.data by mku.sh.
u4.base
u4.test
u5.base
u5.test

ua.base      -- The data sets ua.base, ua.test, ub.base, and ub.test
ua.test      split the u data into a training set and a test set with
ub.base      exactly 10 ratings per user in the test set. The sets
ub.test      ua.test and ub.test are disjoint. These data sets can
              be generated from u.data by mku.sh.

allbut.pl    -- The script that generates training and test sets where
              all but n of a users ratings are in the training data.

mku.sh       -- A shell script to generate all the u data sets from u.data.
```