Collaborative Filtering for Movie Recommendations

Alex Riley
Katelyn Stringer

Types of Recommendation Systems

- Content-based
- User-based
- Hybrid
- More sophisticated models...

*

Types of Recommendation Systems

- Content-based
- User-based
- Hybrid
- More sophisticated models...

*

User-Based Collaborative Filtering

1. Quantify similarities between users by comparing previous ratings
2. Predict ratings using a weighted combination of other ratings

    -- Weights = similarity of users

movielens

grouplens

UNIVERSITY OF MINNESOTA

20 Million movie ratings

- 1995 – 2015
- Users selected at random
- Must have ≥ 20 ratings
- No identifying information

https://movielens.org

The MovieLens Data Set

- Ratings are 0.5 - 5 stars
  - 0.5 star increments
- Users are identified by ID number
- Movie IDs match titles in sep. file
- Other quantities available:
  - Tags, genre, time stamps

	userId	movieId	rating	timestamp
20000258	138493	68954	4.5	1258126920
20000259	138493	69526	4.5	1259865108
20000260	138493	69644	3.0	1260209457
20000261	138493	70286	5.0	1258126944
20000262	138493	71619	2.5	1255811136

The MovieLens Data Set

- Ratings are 0.5 - 5 stars
  - 0.5 star increments
- Users are identified by ID number
- Movie IDs match titles in sep. file
- Other quantities available:
  - Tags, genre, time stamps

	userId	movieId	rating	timestamp
20000258	138493	68954	4.5	125[illegible]126[illegible]20
20000259	138493	69526	4.5	12598[illegible]5108
20000260	138493	69644	3.0	12602[illegible]9457
20000261	138493	70286	5.0	125[illegible]126[illegible]44
20000262	138493	71619	2.5	12[illegible]58111[illegible]6

The MovieLens Data Set

- Ratings are 0.5 - 5 stars
  - 0.5 star increments
- Users are identified by ID number
- Movie IDs match titles in sep. file
- Other quantities available:
  - Tags, genre, time stamps

	userId	movieId	rating	~~timestamp~~
20000258	138493	68954	4.5	~~125[illegible]126[illegible]20~~
20000259	138493	69526	4.5	~~12598[illegible]5108~~
20000260	138493	69644	3.0	~~12602[illegible]9457~~
20000261	138493	70286	5.0	~~125[illegible]126[illegible]44~~
20000262	138493	71619	2.5	~~12[illegible]5811[illegible]6~~

20 Million → 100,000 ratings

*

Formatting the Data

Convert to matrix for analysis

User	Movie	Rating
25661	50	1
32890	50	5
50987	60	3
32890	60	2

pandas.pivot_table

Movie	50	60
User		
25661	1	NaN
32890	5	2
50987	NaN	3

*

Calculating Similarity

1. Pearson Correlation Coefficient
2. Cosine Similarity

$$similarity(A,B) = \frac{\sum_{i=1}^{n}(A_i - \bar{A})(B_i - \bar{B})}{\sqrt{\sum_{i=1}^{n}(A_i - \bar{A})^2 \sum_{i=1}^{n}(B_i - \bar{B})^2}}$$

Pros:

- Easy (pandas.corr)
- Accounts for “niceness” of user-- measures scatter around mean of ratings

Cons:

- Overestimates similarity for very few overlaps between users
    - same rating for one movie gives sim=1

Image Credit: http://guides.neo4j.com/sandbox/recommendations

Calculating Similarity

1. Pearson Correlation Coefficient
2. Cosine Similarity

$$similarity(A,B) = \frac{A \cdot B}{\|A\| \times \|B\|} = \frac{\sum_{i=1}^{n} A_i \times B_i}{\sqrt{\sum_{i=1}^{n} A_i^2} \times \sqrt{\sum_{i=1}^{n} B_i^2}}$$

Pros:

- Boils down to direct matrix math
  - check with sklearn’s pairwise_distances
- Weight factor [0,1] means no negative predictions

Cons:

- Matrix multiplication with NaNs -> replace with 0s, depresses similarity

Image Credit: http://guides.neo4j.com/sandbox/recommendations

Result: Similarity Matrix

User	25661	32890	50987	60453
User				
25661	1	NaN	0.98	0.66
32890	NaN	1	-0.23	0.59
50987	0.98	-0.23	1	NaN
60453	0.66	0.59	NaN	1

*

Predicting Ratings

$$\mathrm{pred}_A = \frac{1}{\sum_{k \in R_j} \mathrm{sim}(A, k)} \sum_{k \in R_j} \mathrm{sim}(A, j)\mathrm{rating}_{k,j}$$

A = user

k = index of other user

R_j = set of users who rated movie j

*

User	25661	32890	50987	60453
User				
25661	1	NaN	0.98	0.66
32890	NaN	1	-0.23	0.59
50987	0.98	-0.23	1	NaN
60453	0.66	0.59	NaN	[illegible]

Mo	50	60	89	103	973
User					
25661	1	NaN / 0	3	5	2
32890	5	2	NaN /0	3	1
50987	NaN / 0	3	4	4	NaN / 0
60453	5	1	4.5	3	2.5

*

Predicting Ratings

$$\frac{1*NaN + NaN*2 + 0.98*3 + 0.66*1}{(NaN + NaN + 0.98 + 0.66)} = 3.34 \text{ for movie \# 60 for user 25661}$$

Measuring Performance

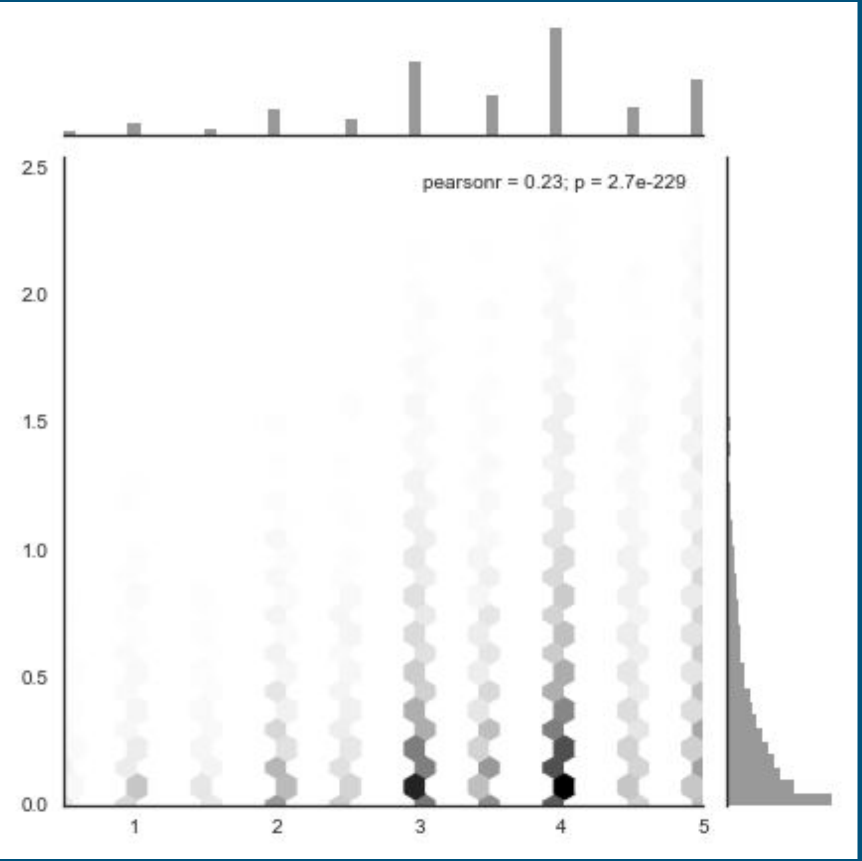
Split data into training (80%) & test (20%) sets

Movie	50	60	89	103	973
User					
25661	1	NaN / 0	3	5	2
32890	5	2	NaN /0	3	1
50987	NaN / 0	3	4	4	NaN / 0
60453	5	1	4.5	3	2.5

Measuring Performance: Pearson

Use the training data to predict the test ratings

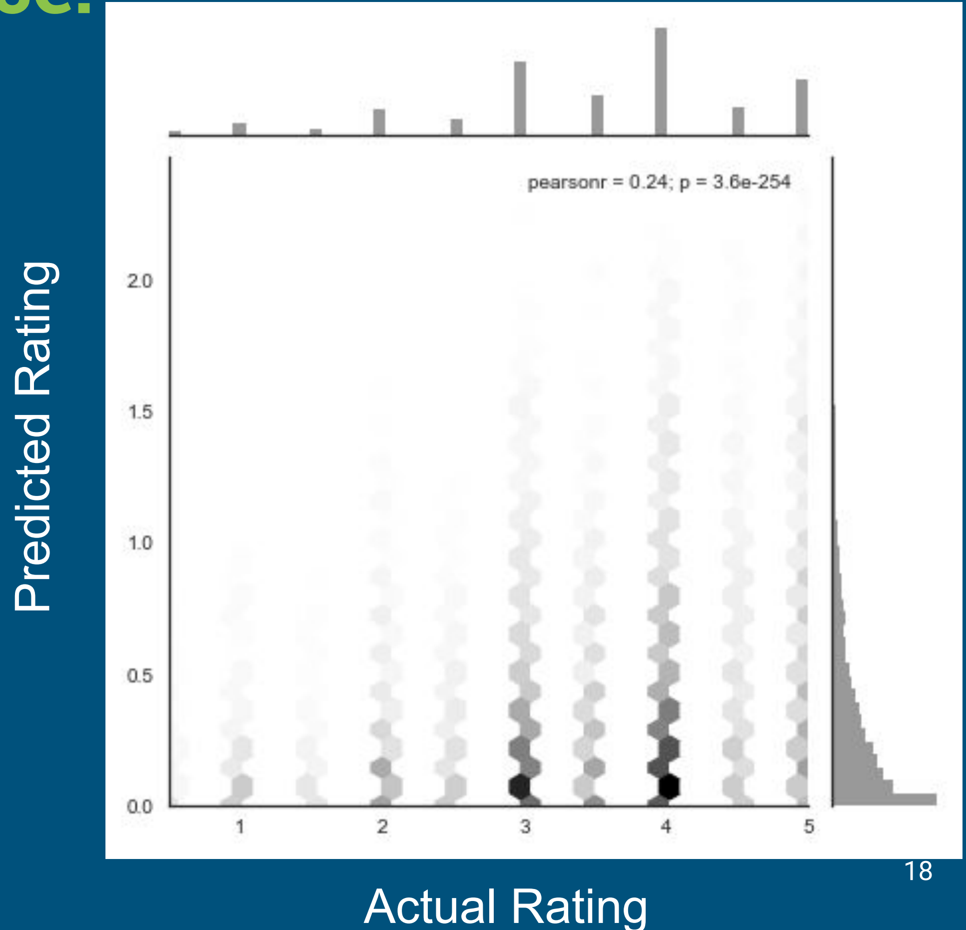
MSE ≅ 10.8667



Measuring Performance: Cosine

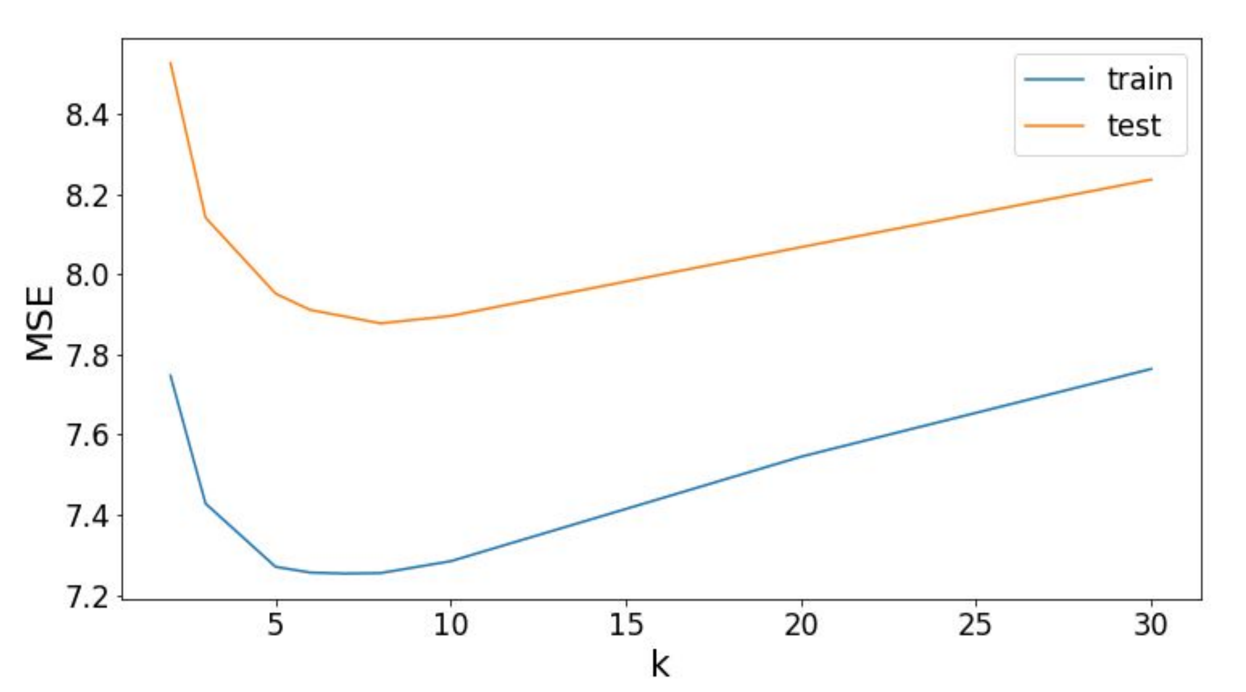
Use the training data to predict the test ratings

MSE ≅ 10.7631



Top-*k* Filtering: Cosine

How many similar users should be considered when making recommendations?



*

The Ultimate Test...

Katelyn’s Input

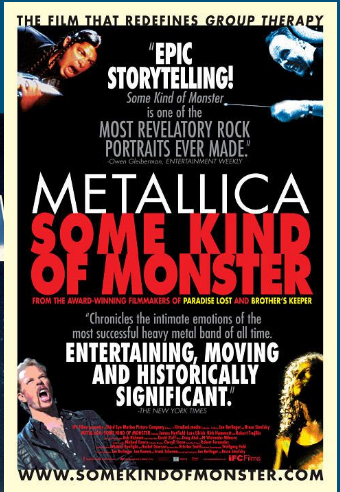
	title	rating
2	Illusionist, The (L'illusionniste) (2010)	5.0
3	Catch Me If You Can (2002)	5.0
5	Bridge of Spies (2015)	5.0
20	WALL·E (2008)	5.0
0	13 Going on 30 (2004)	4.5
22	The Man from U.N.C.L.E. (2015)	4.5
8	Jonah: A VeggieTales Movie (2002)	4.5
11	Ant-Man (2015)	4.5
14	Remember the Titans (2000)	4.0
19	The Age of Adaline (2015)	4.0
18	Chronicles of Narnia: The Lion, the Witch and ...	4.0
17	Little Miss Sunshine (2006)	4.0

16	Little Rascals, The (1994)	4.0
13	Lucky One, The (2012)	4.0
12	Rookie, The (1990)	4.0
7	Mr. Bean's Holiday (2007)	4.0
15	Batman: The Dark Knight Returns, Part 1 (2012)	3.5
25	Wedding Crashers (2005)	3.5
10	Hangover, The (2009)	3.0
9	Twilight Saga: New Moon, The (2009)	3.0
4	Talladega Nights: The Ballad of Ricky Bobby (2...	3.0
23	Peanuts Movie, The (2015)	3.0
24	Baby's Day Out (1994)	3.0
21	Grease 2 (1982)	2.0
6	Sabrina (1954)	1.5
1	Without a Paddle (2004)	1.0

*

Katelyn’s Results: Pearson Similarity

Recommended:



NOT Recommended:

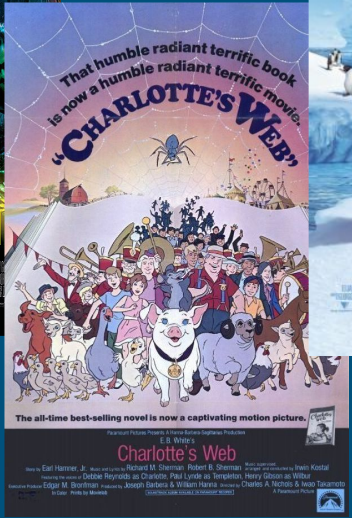


Image Credit: IMDb

Alex's Input

	title	rating
23	Santa Clause, The (1994)	5.0
21	Scooby-Doo (2002)	5.0
19	Looney, Looney, Looney Bugs Bunny Movie, The (...	5.0
17	Meet the Robinsons (2007)	5.0
14	The Imitation Game (2014)	5.0
7	Phantom of the Opera, The (2004)	5.0
13	Bruce Almighty (2003)	4.5
10	Pink Panther Strikes Again, The (1976)	4.5
25	Home Alone 3 (1997)	4.5
2	Giver, The (2014)	4.0
1	Cloudy with a Chance of Meatballs (2009)	4.0
4	Bill Cosby, Himself (1983)	4.0
8	Legend of Zorro, The (2005)	3.5
20	Jesus Christ Superstar (1973)	3.5
6	The Theory of Everything (2014)	3.5
15	Doctor Dolittle (1967)	3.5
9	Legend of Zorro, The (2005)	3.0
0	Shutter Island (2010)	3.0
16	Parent Trap, The (1961)	3.0
12	American Sniper (2014)	2.5
3	The Island (2006)	2.0
22	Snow White and the Huntsman (2012)	2.0
18	Fast Five (Fast and the Furious 5, The) (2011)	0.5
11	Resident Evil: Apocalypse (2004)	0.5
24	Nightmare on Elm Street 2: Freddy's Revenge, A...	0.5
5	Taken 3 (2015)	0.5

Alex’s Results: Cosine Similarity

Recommended:

NOT Recommended:

Image Credit: IMDb

How Can We Make it Better?

Alex and Katelyn need to watch more movies

More ratings per user → Better recommendations?

Adapt code to handle missing values (NaN or 0) better

Make a hybrid model of user-based & item-based clustering using some of the data we ignored: genres, tags, etc.

Thank you!

Moral of the Story: Don’t take good recommendations for granted!

https://github.com/stringkm/movie-matchmaker