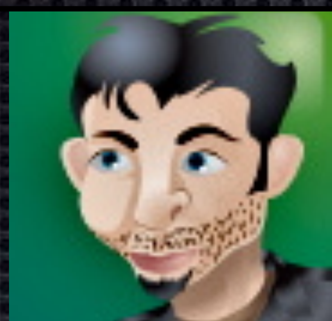
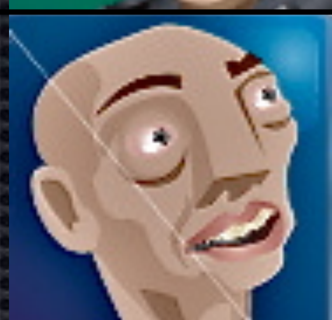
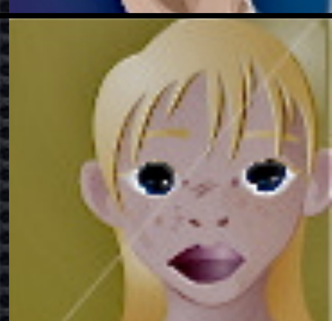
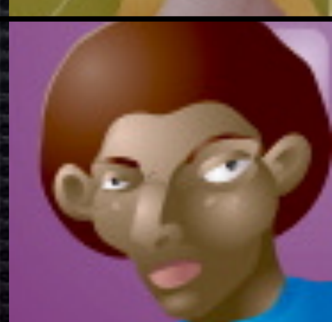
NETFLIX
Movie Recommendations

Virgil Pavlu
Shahzad Rajput
Keshi Dai

Movie ratings: 1(bad) - 5(good)

5	3	2	1	5

Movie ratings

					
	5	3	2	?	5
	3	1	5	4	?
	4	4	?	3	5
	?	5	3	2	4

COLLABORATIVE FILTERING; PEARSON FORMULA

compute for each user u mean and variance. Let N_u = number of movies rated by user u; R_{um} is the rating of user u for movie m

$$\mu_u = \frac{\sum_m R_{um}}{N_u}$$

$$\sigma_u = \frac{\sum_m R_{um}^2}{N_u} - \mu_u^2$$

normalize each ratings by substracting the user mean and dividing by user variance

$$\bar{r}_{um} = \frac{R_{um} - \mu_u}{\sigma_u}$$

compute user similarity between any two users u and v

$$\rho_{uv} = \frac{1}{\text{movies in common } m} \sum_m \bar{r}_{um} \cdot \bar{r}_{vm}$$

predict the rating for a new movie by accounting for all other users' v rating on the movie

$$predict(u, m) = \mu_u + \frac{\sum_v \rho_{uv} \cdot \bar{r}_{vm}}{\sum_v |\rho_{uv}|} \cdot \sigma_u$$

Users-item-ratings problem

- Usually very sparse
- Many applications
  - article recommendation
- Amazon, Netflix, iTunes and many others
  - pretty much all online stores/services
  - “automatic” reviews
  - some items (movie, books) easier than others
- Content vs Collaborative approach

NETFLIX dataset

- Rent movies via postal service
  - recently also online
- 18000 movies
- .5 million users
- Training: 100 million ratings
- Testing : 1 million ratings
  - measure perfomance : RMSE

37918 teams / 180 countries

Netflix Prize

Home | Rules | Leaderboard | Register | Update | Submit | Download

Leaderboard

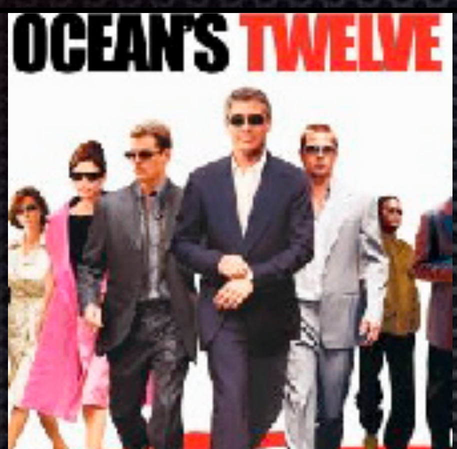
Display top 40 leaders.

Rank	Team Name	Best Score	% Improvement	Last Submit Time
--	No Grand Prize candidates yet	--	--	--
Grand Prize - RMSE <= 0.8563				
1	PragmaticTheory	0.8537	9.64	2009-03-14 02:00:01
2	BellKor in BigChaos	0.8538	9.63	2009-01-05 22:05:26
3	Dace	0.8606	9.54	2009-03-11 00:12:12
4	Grand Prize Team	0.8609	9.51	2009-03-12 17:56:36
Progress Prize 2008 RMSE = 0.8616 Winning Team: BellKor in BigChaos				
5	BigChaos	0.8624	9.35	2009-02-07 13:06:32
6	BellKor	0.8628	9.31	2008-12-31 11:50:49
7	Gravity	0.8651	9.07	2009-01-23 06:58:01
8	Ces	0.8654	9.04	2009-03-09 03:03:22
9	Opera Solutions	0.8654	9.04	2009-03-13 08:00:07
10	NewNetflixTeam	0.8657	9.01	2009-03-12 05:53:42
11	J Dennis Su	0.8658	9.00	2009-03-11 09:41:54
12	BruceDengDaoCiYiYou	0.8660	8.98	2009-03-11 01:24:48
13	acmehill	0.8661	8.97	2009-03-11 10:39:16
14	Feeds2	0.8665	8.92	2009-03-10 17:34:20
15	pengpengzhou	0.8666	8.91	2009-03-11 00:49:53
16	My Brain and His Chain	0.8668	8.89	2008-09-30 02:19:47
17	Just a guy in a garage	0.8669	8.88	2009-02-17 18:10:59
18	scientist	0.8670	8.87	2009-03-11 23:45:07
19	When Gravity and Dinosaurs Unite	0.8675	8.82	2008-10-05 14:16:53
20	IDEA2	0.8675	8.82	2009-03-13 10:15:13

Collaborative Filtering

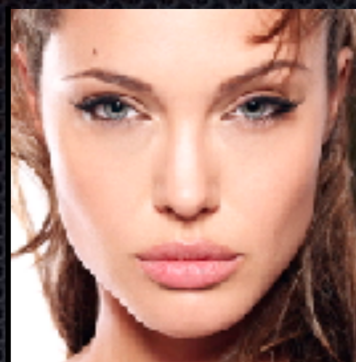
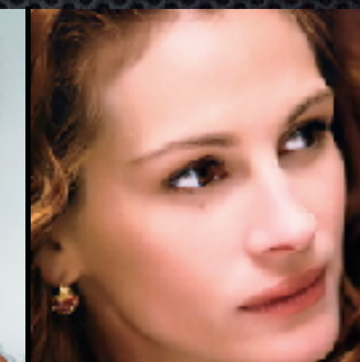
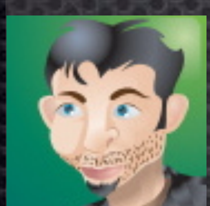
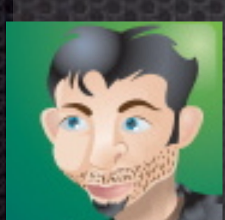
- Use similarity between users/items
- Many solutions, old and new
  - Simple : Pearson's formula
    - measure statistical correlation between users/items
  - Simple : Rule-based
  - k-Nearest Neighbor/k-Means + regression
  - Model effects due to user/movie/time etc
    - Star Wars may not be as likeable now as 30 years ago
  - Matrix factorization

Content-based training

		x			x	x

- Identify movies by content features
  - Actors, genre, director, writer etc
  - 6000 features to cover 90% of NETFLIX dataset
  - We use content data from IMDB
- Learn a profile for each user

User profile

						
 movie r=**4**	4	4			4	4
 movie r=**1**	1			1	1	
 movie r=**5**			5	5	5	
 profile	2.5	4	5	3	3.3	4

Content + Collaborative

- Fix a movie m
- Build a training set with content+collab features

profile collaborative

training testing

	date	c_1	c_2	c_3	c_4 ...	m_1	m_2	m_3...	rating
u_1	.28	1.2	4.3	-	3.8 ...	5	2	1 ...	3
u_2	.35	2.5	2.1	1.5	4.1 ...	4	3	4 ...	4
u_3	.78	1.4	1.2	-	3.2 ...	-	-	1 ...	1
u_4	.32	-	-	1.7	2.8 ...	3	1	- ...	5
u_5	.34	2.1	4.0	2.3	2.0 ...	-	2	1 ...	1
u_6	.31	2.8	3.5	2.6	3.4 ...	2	-	1 ...	2
u_7	.38	-	4.2	2.9	2.8 ...	4	3	- ...	?
u_8	.29	2.4	4.5	-	2.0 ...	-	2	2 ...	?
u_9	.30	1.9	3.8	3.1	3.4 ...	-	4	3 ...	?

- Run decision tree + regression

Content + Collaborative

- On some movies content features dominant
- On others, collab features dominant

			profile				collaborative			
		date	c_1	c_2	c_3	c_4 ...	m_1	m_2	m_3...	rating
training	u_1	.28	1.2	4.3	-	3.8 ...	5	2	1 ...	3
	u_2	.35	2.5	2.1	1.5	4.1 ...	4	3	4 ...	4
	u_3	.78	1.4	1.2	-	3.2 ...	-	-	1 ...	1
	u_4	.32	-	-	1.7	2.8 ...	3	1	- ...	5
	u_5	.34	2.1	4.0	2.3	2.0 ...	-	2	1 ...	1
	u_6	.31	2.8	3.5	2.6	3.4 ...	2	-	1 ...	2
testing	u_7	.38	-	4.2	2.9	2.8 ...	4	3	- ...	?
	u_8	.29	2.4	4.5	-	2.0 ...	-	2	2 ...	?
	u_9	.30	1.9	3.8	3.1	3.4 ...	-	4	3 ...	?

[Preliminary] results

- About 600 movies, chosen randomly
  - Train on 90% of data
  - Test on 10% of data
- Overall RMSE=.95
- Problems with movies with few ratings