# Recommender Systems

Insight

# Outline

Automatic Collaborative Filtering:

- People who bought this also bought...

Commercial examples

- also content-based recommendation

ACF algorithms

Latent space / matrix factorization methods

ACF versus content-based recommendation

Attacks on ACF systems

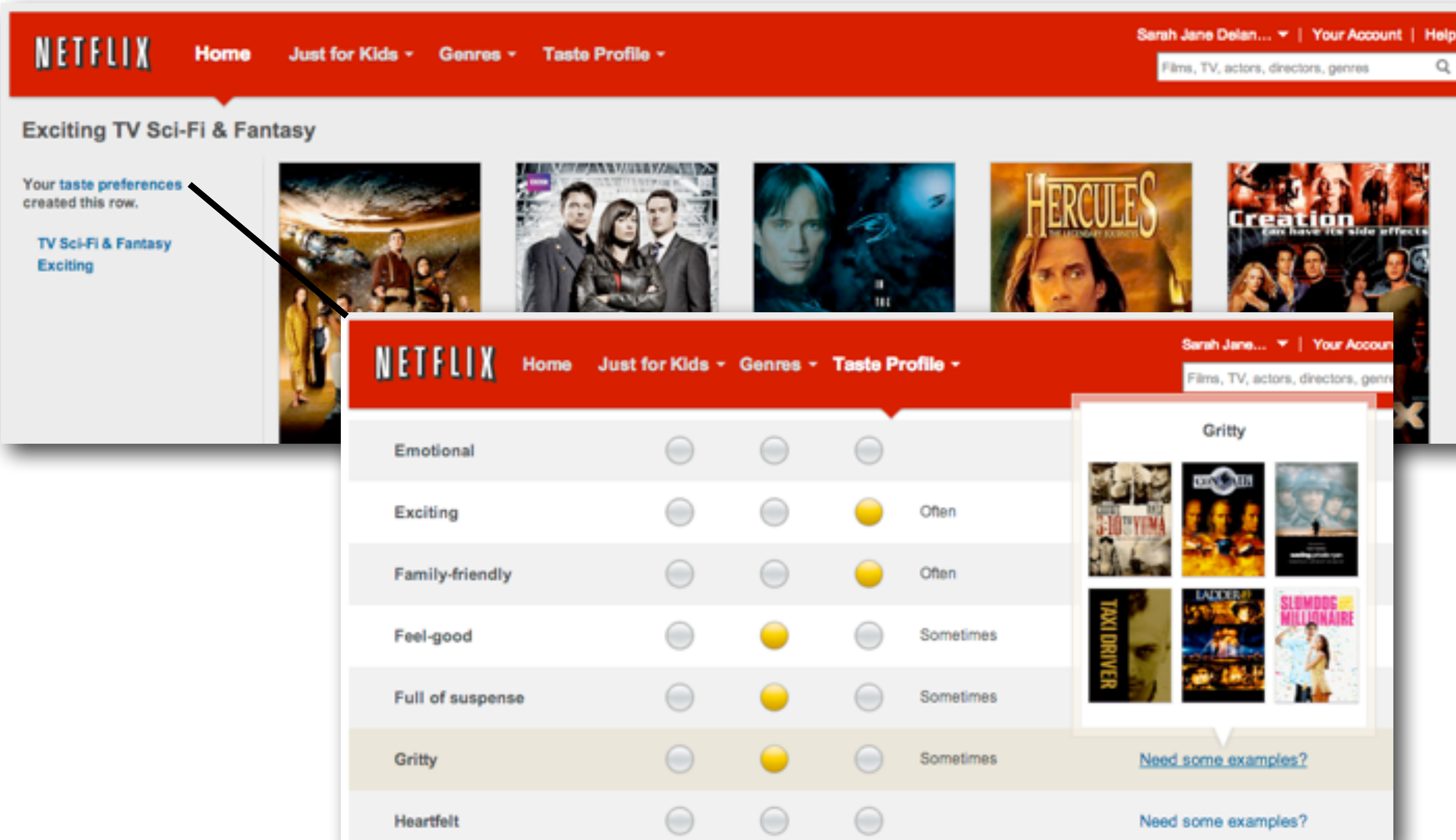
# Key Technology for Amazon


Insight

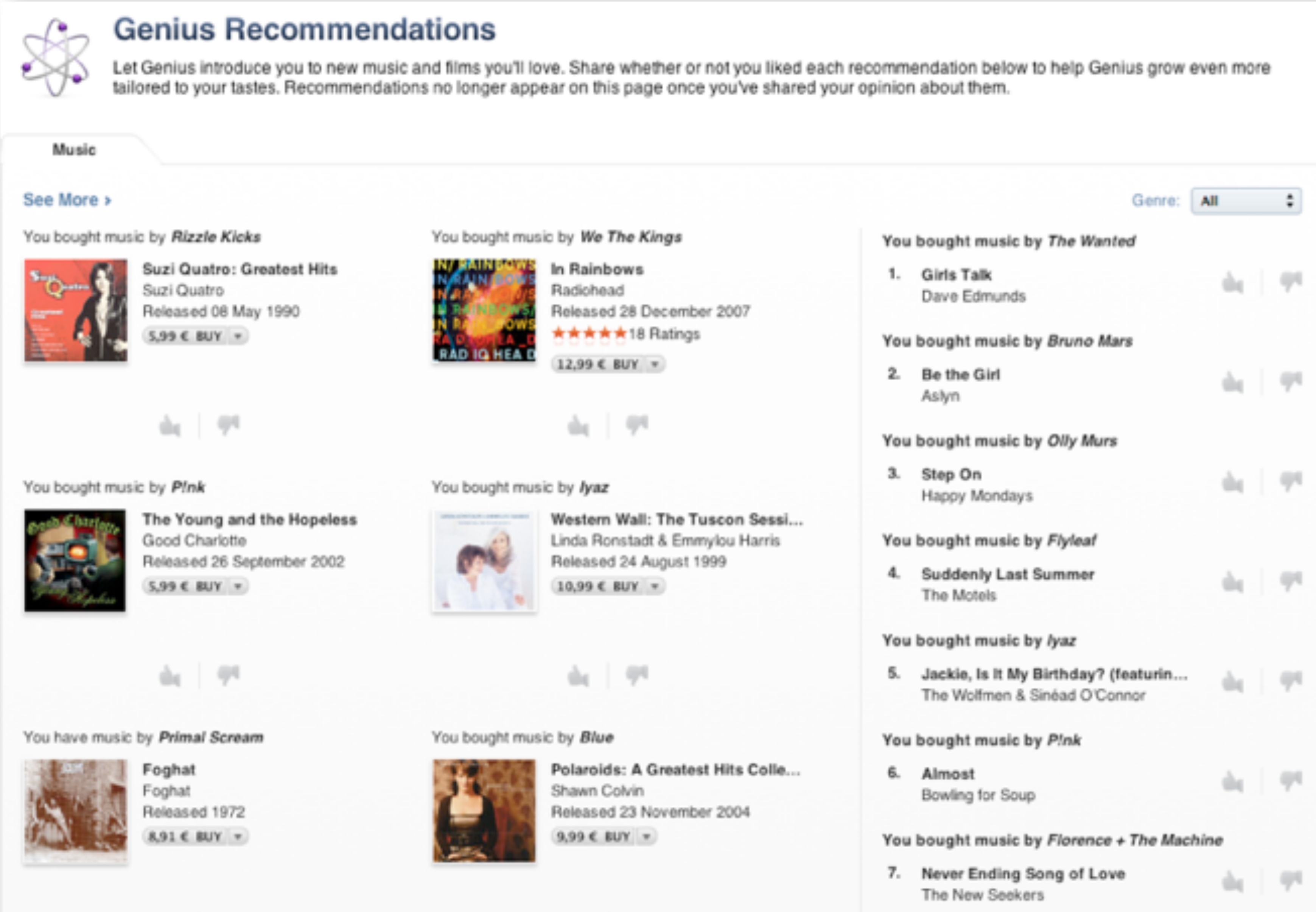
# And Netflix


Although multiple users on a single account can mess up your preferences...

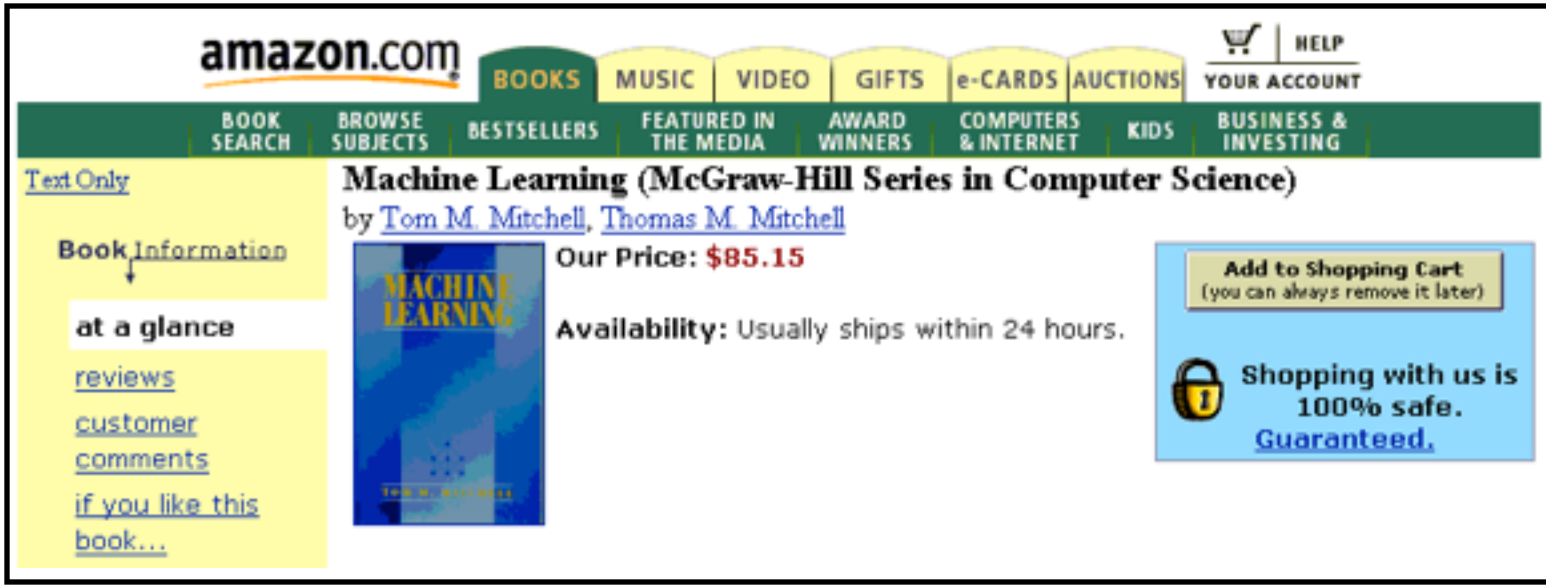
# Netflix also do content-based recommendation



# Also iTunes


Genius Recommendations
Let Genius introduce you to new music and films you'll love. Share whether or not you liked each recommendation below to help Genius grow even more tailored to your tastes. Recommendations no longer appear on this page once you've shared your opinion about them.
Music
See More ›
Genre: All
You bought music by Rizzle Kicks
Suzi Quatro: Greatest Hits
Suzi Quatro
Released 08 May 1990
5,99 € BUY
You bought music by We The Kings
In Rainbows
Radiohead
Released 28 December 2007
18 Ratings
12,99 € BUY
You bought music by P!nk
The Young and the Hopeless
Good Charlotte
Released 26 September 2002
5,99 € BUY
You bought music by Iyaz
Western Wall: The Tuscon Sessi...
Linda Ronstadt & Emmylou Harris
Released 24 August 1999
10,99 € BUY
You have music by Primal Scream
Foghat
Foghat
Released 1972
8,91 € BUY
You bought music by Blue
Polaroids: A Greatest Hits Colle...
Shawn Colvin
Released 23 November 2004
9,99 € BUY
You bought music by The Wanted
1. Girls Talk
Dave Edmunds
You bought music by Bruno Mars
2. Be the Girl
Aslyn
You bought music by Olly Murs
3. Step On
Happy Mondays
You bought music by Flyleaf
4. Suddenly Last Summer
The Motels
You bought music by Iyaz
5. Jackie, Is It My Birthday? (featurin...
The Wolfmen & Sinéad O'Connor
You bought music by P!nk
6. Almost
Bowling for Soup
You bought music by Florence + The Machine
7. Never Ending Song of Love
The New Seekers

# Automatic Collaborative Filtering (ACF)




## Customers who bought this book also bought:

- **Reinforcement Learning : An Introduction**; R. S. Sutton, A. G. Barto
- **Advances in Knowledge Discovery and Data Mining**; U. M. Fayyad, et al
- **Probabilistic Reasoning in Intelligent Systems**; J. Pearl

# Collaborative Recommendation (in one slide)

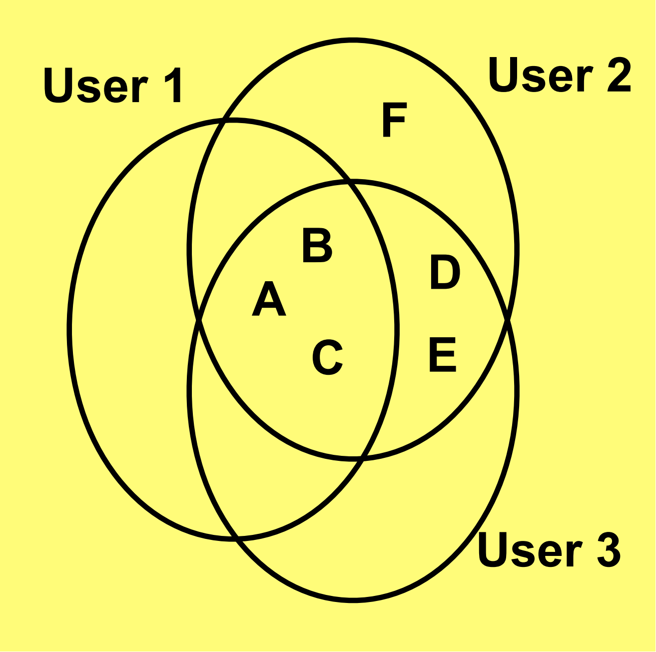
Users 1, 2 and 3 in same group

- {A,B,C} in common

Assets D & E can be recommended to User 1 based on this shared interest

Recommendation based on observations

- no detailed representation of D or E
- users must be identified




Step 1: Identify neighbours
Step 2: Recommend stuff they liked



Insight

# Automatic Collaborative Filtering

Recommend assets to users on the basis of how they have rated other assets.

**Representation-less** approach, since no descriptions of assets involved.

Example data (no asset description):

| | Song 1 | Song 2 | Song 3 | Song 4 | Song 5 | Song 6 |
|---|---|---|---|---|---|---|
| User 1 | 0.8 | 1.0 | | 0.8 | 0.2 | |
| User 2 | 0.2 | 1.0 | 0.0 | | 0.2 | 1.0 |
| User 3 | 0.6 | 0.6 | 1.0 | | | 0.4 |
| User 4 | | | | | 0.8 | 0.2 |
| User 5 | 0.4 | 0.6 | 1.0 | 1.0 | | |
| User 6 | | 0.8 | 0.0 | 0.6 | 1.0 | 0.4 |
| User 7 | 0.0 | | | 0.6 | | |
| User 8 | | 0.4 | 0.0 | 0.8 | 0.6 | 1.0 |

# Compare: content-based recommendation (*k*-NN)

Assets and users have a case-like description

Users are matched to assets that best meet their interests

| **4W&1F** | |
|---|---|
| Title: | Four Weddings and a Funeral |
| Year: | 1994 |
| Genre: | Comedy, Romance |
| Director: | Mike Newell? |
| Starring: | Hugh Grant, Andie MacDo[illegible] |
| Runtime: | 116 |
| Country: | UK |
| Language: | English |
| Certification: | USA:R (UK:15) |

$$Sim(u,p) = \sum_{i \in Features} w_i \times \sigma(f_i^u, f_i^p)$$

| **JB-7** | |
|---|---|
| Name: | Joe Bloggs |
| Preferred Era: | 1988 --> |
| Genre: | Thriller, Comedy, War, Romance |
| Director: | S. Spielburg, F. F. Coppola |
| Actors:A12 | Sharon Stone, Sylvester Stallone, Julia Roberts, Keanu Reeves, Liam Neeson, Andie MacDowell |
| [illegible]: | < 150 |
| [illegible]: | UK, US |
| Language: | English |
| Certification: | Any |

AKA: feature-based or case-based recommendation

Insight

# Representation for ACF is very different

| | A | B | C | D | E | F | G |
|---|---|---|---|---|---|---|---|
| **User 1** | 0.6 | 0.6 | 0.8 | | | 0.8 | 0.5 |
| **User 2** | | 0.8 | 0.8 | 0.3 | 0.7 | | |
| **User 3** | 0.6 | 0.6 | 0.3 | 0.5 | | 0.7 | 0.5 |
| **User 4** | | | | | 0.7 | 0.8 | 0.7 |
| **User 5** | 0.6 | 0.6 | 0.8 | | | 0.7 | |
| **User 6** | | 0.8 | 0.8 | 0.7 | 0.7 | | |
| **User 7** | 0.7 | 0.5 | | | 0.7 | | |
| **User 8** | | | | | 0.7 | 0.7 | 0.8 |

If ratings are available

In the absence of explicit ratings

| | A | B | C | D | E | F | G |
|---|---|---|---|---|---|---|---|
| **User 1** | 1 | 1 | 1 | | | 1 | 1 |
| **User 2** | | 1 | 1 | 1 | 1 | | |
| **User 3** | 1 | 1 | 1 | 1 | | 1 | 1 |
| **User 4** | | | | | 1 | 1 | 1 |
| **User 5** | 1 | 1 | 1 | | | 1 | |
| **User 6** | | 1 | 1 | 1 | 1 | | |
| **User 7** | 1 | 1 | | | 1 | | |
| **User 8** | | | | | 1 | 1 | 1 |

# ACF: How it works

Form virtual communities using clustering

- e.g. $k$-Means clustering
- requires a similarity (difference) metric,
  - e.g. mean squared difference

$$\delta_{UJ} = \frac{1}{|InCommon|} \sum_{f \in InCommon} (U_f = J_f)^2$$

  - where $U_f$ is $Us$ rating of asset $f$
  - Pearson correlation coefficient may be better: $r_{UJ}$

- (Shardanand & Maes 1995)

# ACF Rating Assets (Lazy approach)

Build personalised community at run time:

- Identify others with similarity above a threshold
- Rating for an item is a weighted average of rating of similar users for that item.

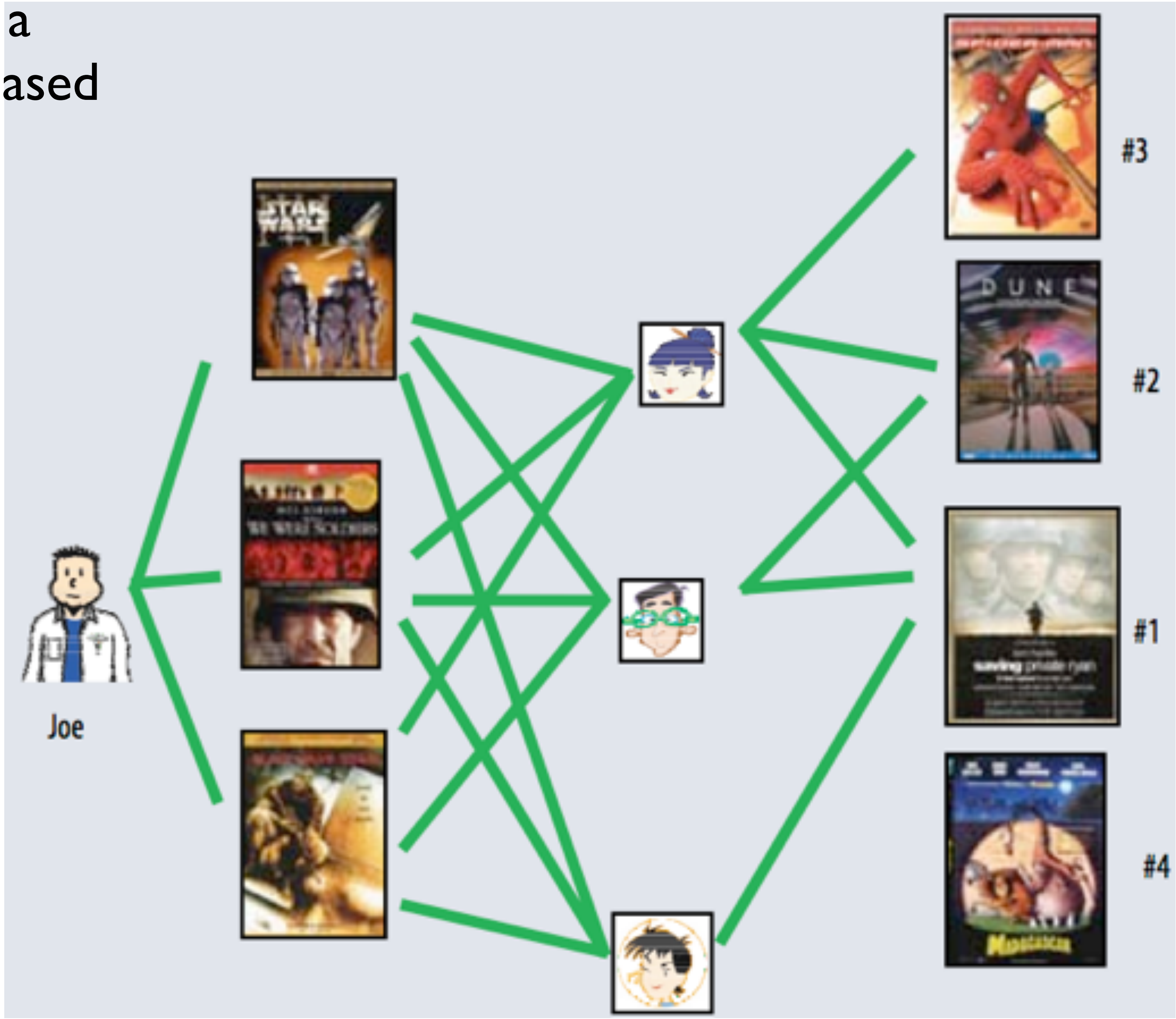
$$U_x = \bar{U} + \frac{\sum_{J \in \text{Raters of } x} (J_x - \bar{J}) r_{UJ}}{\sum_{J \in \text{Raters of } x} |r_{UJ}|}$$

score to be the expected value of the rating of the user for that asset



Insight

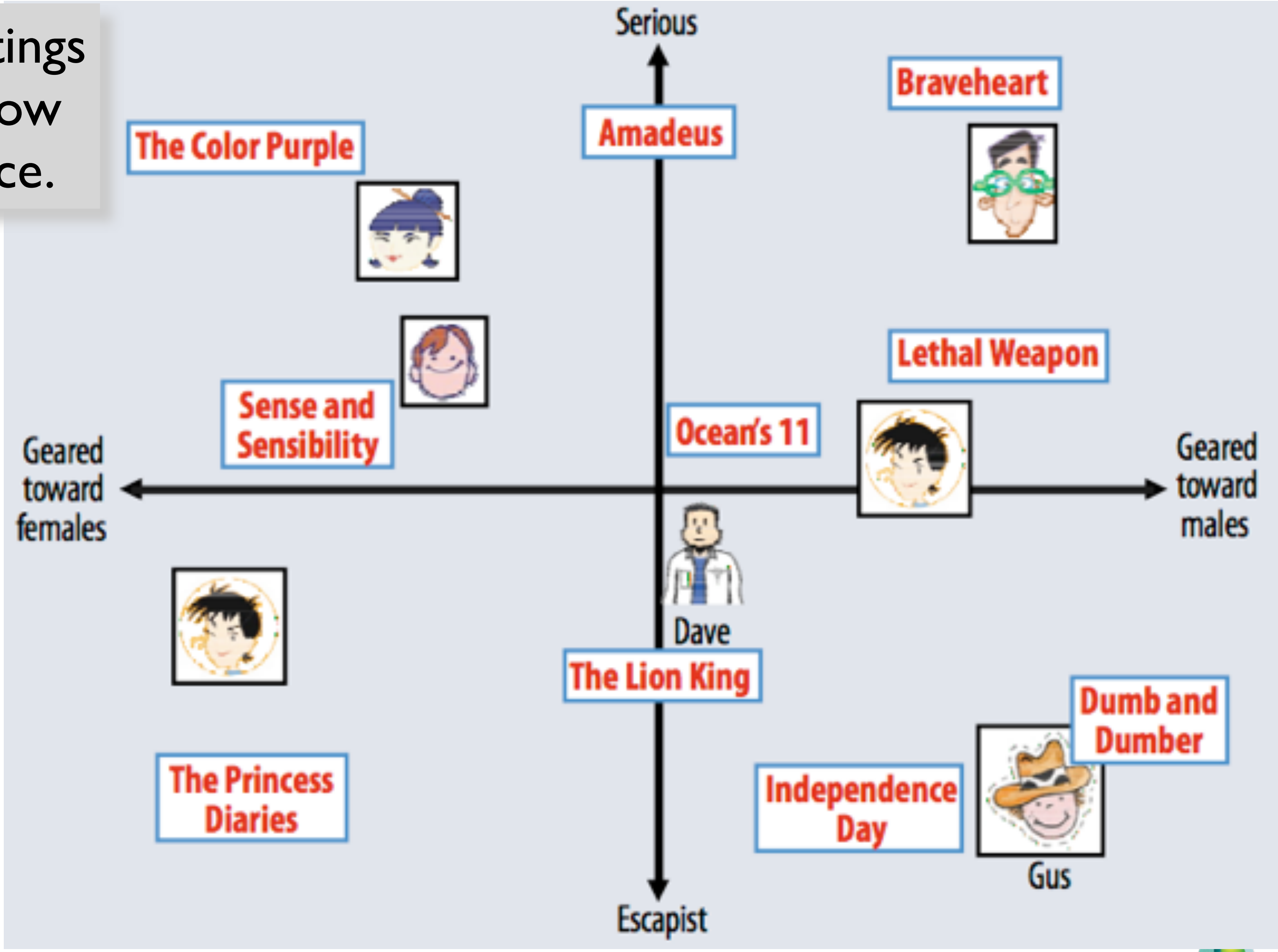
# This strategy is a *neighbourhood*-based approach




(Koren, Bell, Volinski, IEEE Computer 2009)

# Latent Space Approach

Project the ratings matrix into a low dimension space.




(Koren, Bell, Volinski, IEEE Computer 2009)

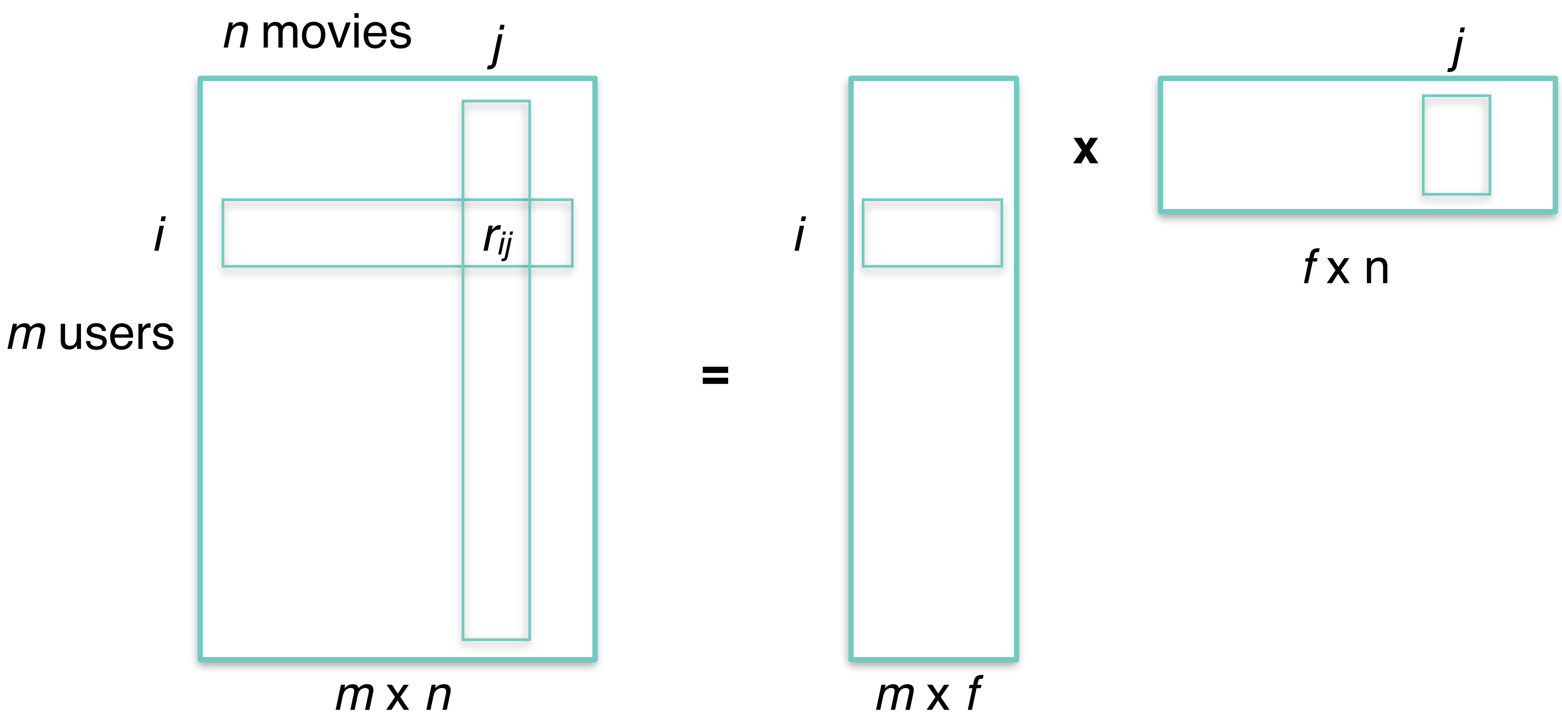
# Netflix Prize

## 100M ratings, 500k users, 18k movies

- <user, movie, grade date, grade (1-5*)>

## Matrix Analysis methods scored well

- project movies and users into a *f* dimension latent space



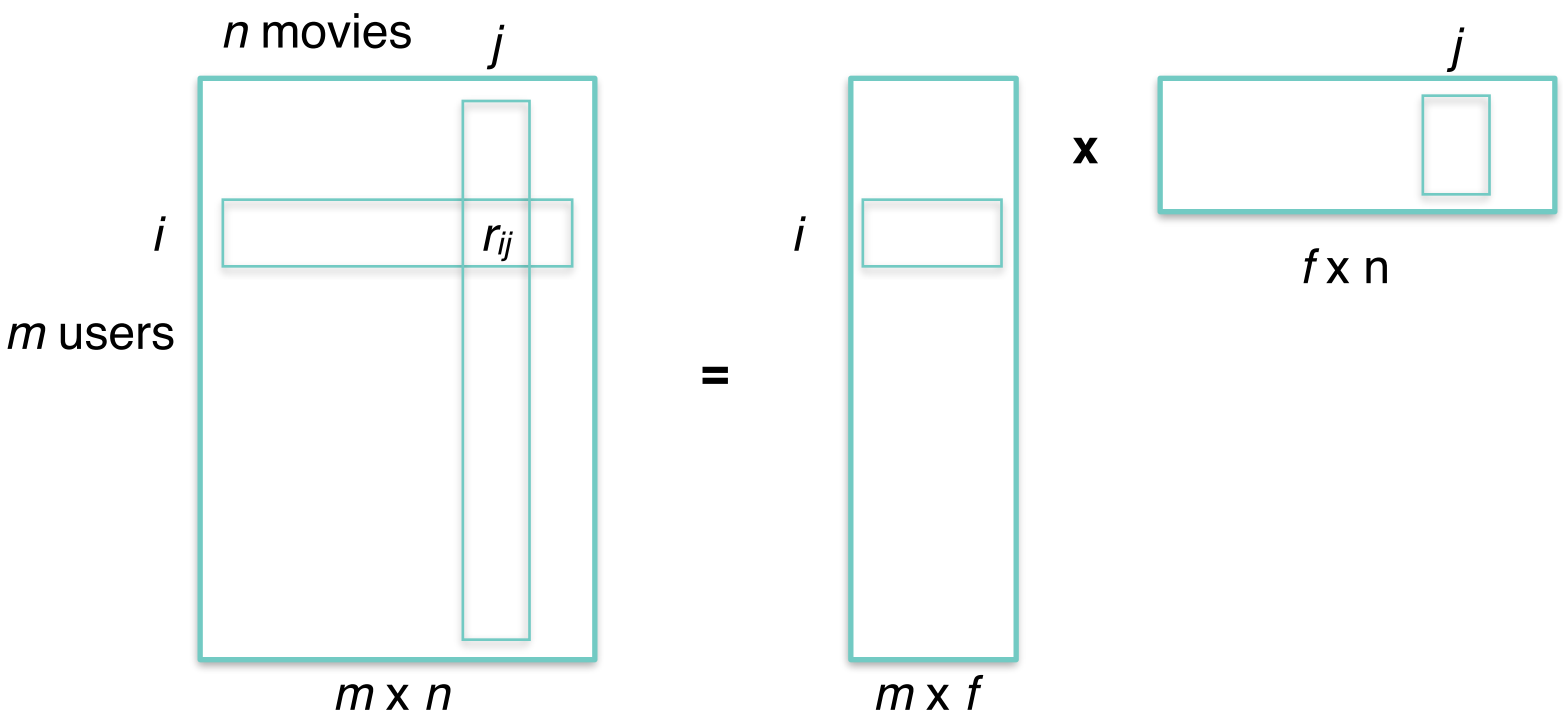
# RecSys using Matrix Factorisation

Ratings matrix is sparse (< 1%)

1st factor matrix represents each user as an *f*-dimension vector

2nd factor matrix represents each movie as an *f*-dimension vector

Affinity of user *i* to movie *j* is the product of these vectors



# ACF as matrix factorization

## Matrix factorization (e.g. SVD)

- map both users and items to a $f$ dimension space
  - each item $i$ represented by a vector $q_i \in \mathbb{R}^f$
    - each entry in $q_i$ indicates how those factors describe $i$
  - each user $u$ represented by a vector $p_u \in \mathbb{R}^f$
    - each entry in $p_u$ indicates $u$'s interest in items high in that factor
  - $q^T{}_i$ x $p_u$ indicates $u$'s interest in $i$
- cannot be done directly using SVD because of missing values

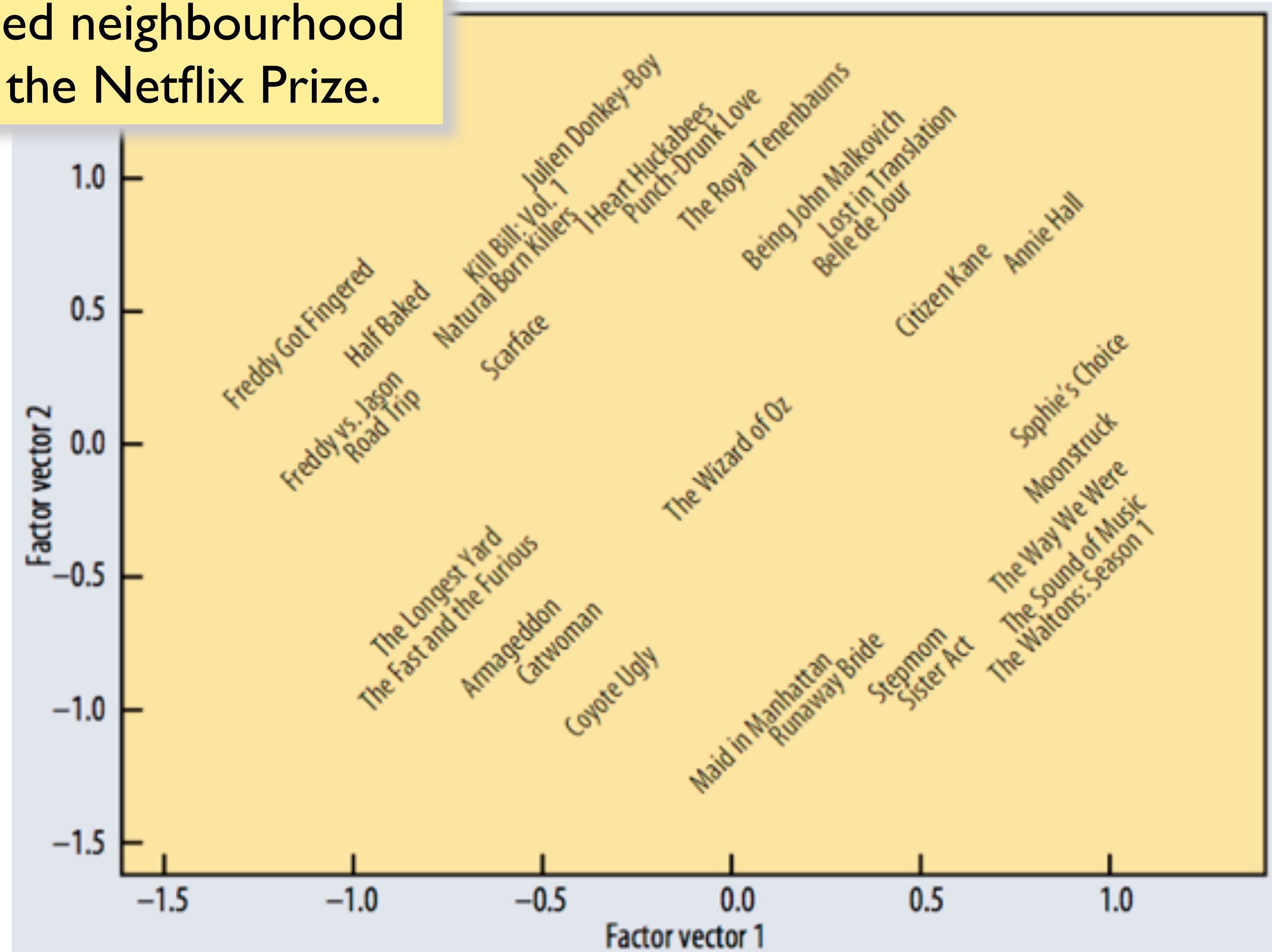
- Alternative

error term　　　　magnitude penalty

$$\min_{q*,p*} \sum_{(u,i)\in\kappa} (r_{ui} - q_i^T p_u)^2 + \lambda(\| q_i \|^2 + \| p_u \|^2)$$

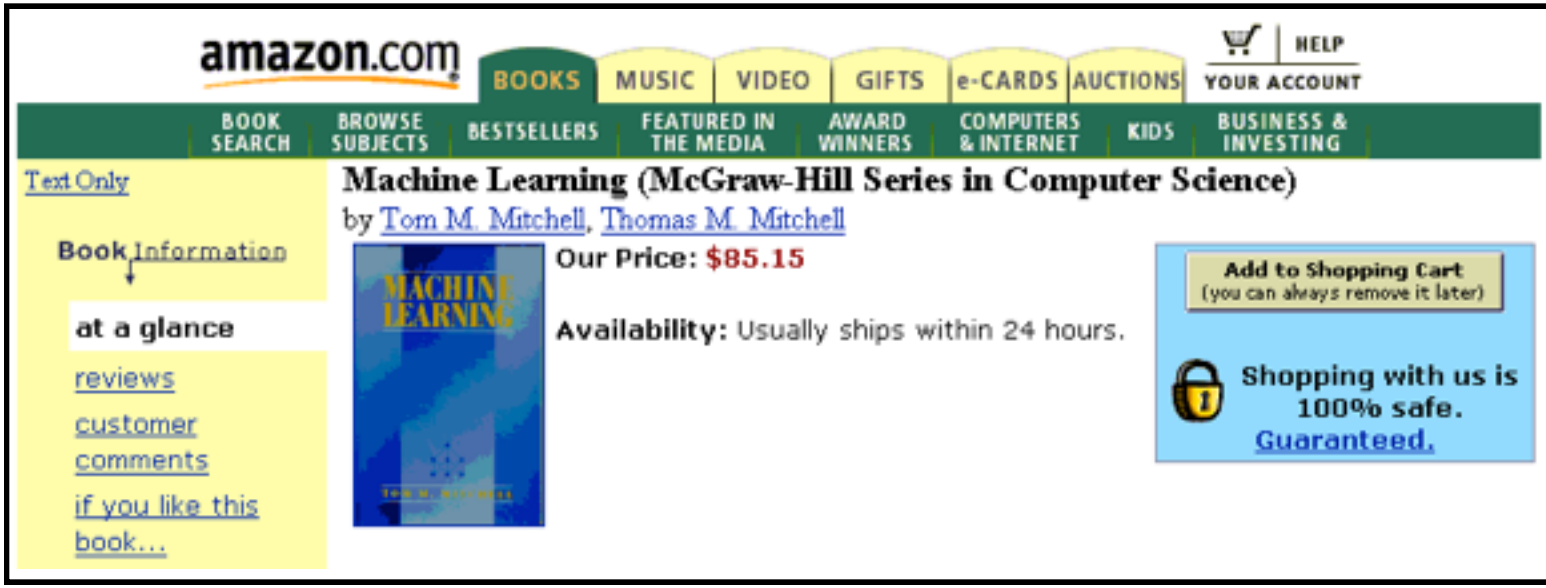
- see (Koren, Bell, Volinski, IEEE Computer 2009) for methods.

Matrix factorization techniques outperformed neighbourhood methods in the Netflix Prize.




(Koren, Bell, Volinski, IEEE Computer 2009)

# Recommendations can be made without explicit ratings:




## Customers who bought this book also bought:

- **Reinforcement Learning : An Introduction**; R. S. Sutton, A. G. Barto
- **Advances in Knowledge Discovery and Data Mining**; U. M. Fayyad, et al
- **Probabilistic Reasoning in Intelligent Systems**; J. Pearl

- **No anonymity**
- **The pump priming problem**: not enough observations
  ⇒ poor recommendations


**Other people who bought this item also purchased:**
The C++ Programming Language, Third Edition
Programming Perl, Second Edition
Advanced Programming in the UNIX Environment

# Content Based Recommendation

- In a content-based recommendation system, recommendation is based on how well an asset matches a user's profile.
- Example representations:

| TB-2 | |
|---|---|
| **Title:** | Unbreak My heart |
| **Year:** | 1996 |
| **Genre:** | Pop, soul |
| **Artist** | Toni Braxton |

| JB-7 | |
|---|---|
| **Name:** | Joe Bloggs |
| **Preferred Era:** | 1990 + |
| **Genre:** | Soul, RnB, Pop |
| **Fav_Artists:** | Lauryn Hill, Macy Gray, George Michael |

The major drawback is the problem of coming up with appropriate features (such as Genre) and marking up the assets using these features.

# ACF: Active v's Passive

## Active:

- Explicit rating supports finer recommendations
- but intrusive

## Passive:

- 0/1 : buy / didn't buy : listened didn't listen
- or implicit rating from usage data
- more noise
  - bought but didn't like
  - how to interpret interruptions in listening



Insight

# Knowledge-based or Data Driven?