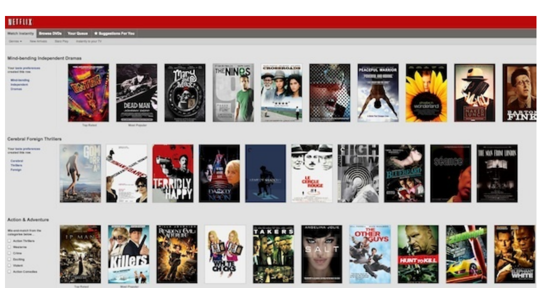
Case Study 4: Collaborative Filtering

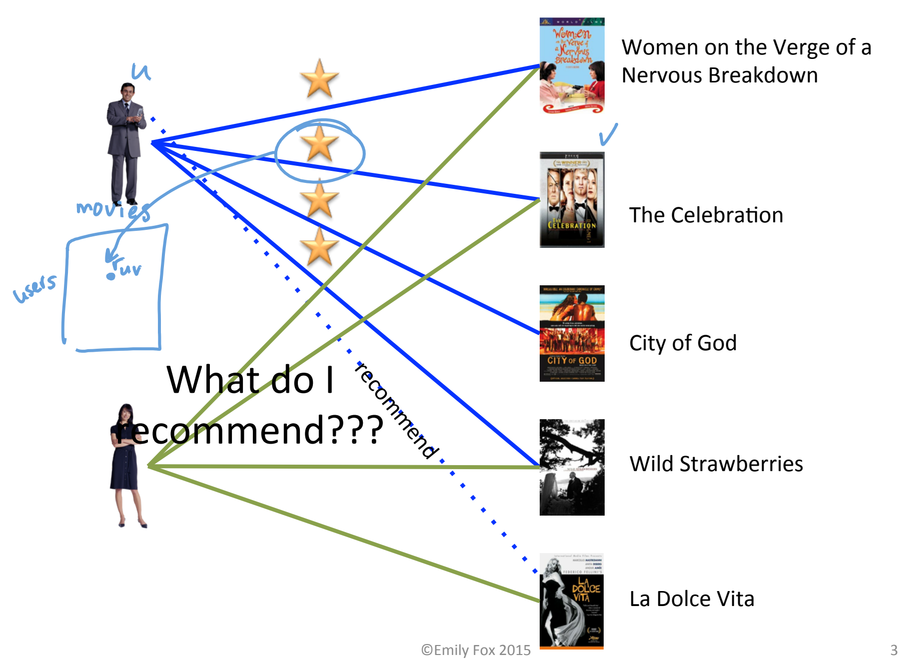
Collaborative Filtering
Matrix Completion
Alternating Least Squares

Machine Learning for Big Data
CSE547/STAT548, University of Washington
Emily Fox
May 5th, 2015

Collaborative Filtering

- **Goal:** Find movies of interest to a user based on movies watched by the user and others
- **Methods:** matrix factorization, GraphLab



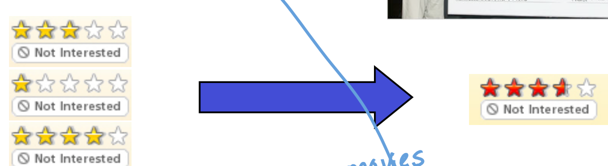


©Emily Fox 2015

3

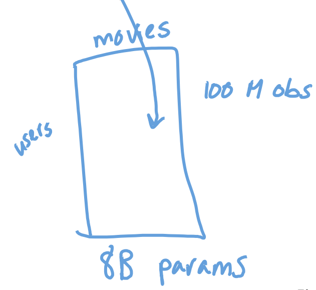
Netflix Prize

- Given 100 million ratings on a scale of 1 to 5, predict 3 million ratings to highest accuracy



- 17770 total movies
- 480189 total users
- Over 8 billion total ratings

- How to fill in the blanks?

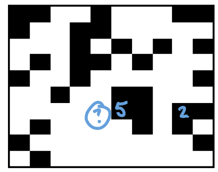


©Emily Fox 2015

Figures from Ben Recht

Matrix Completion Problem

movies

X = 

users

X_{ij} known for black cells
X_{ij} unknown for white cells
Rows index users
Columns index movies

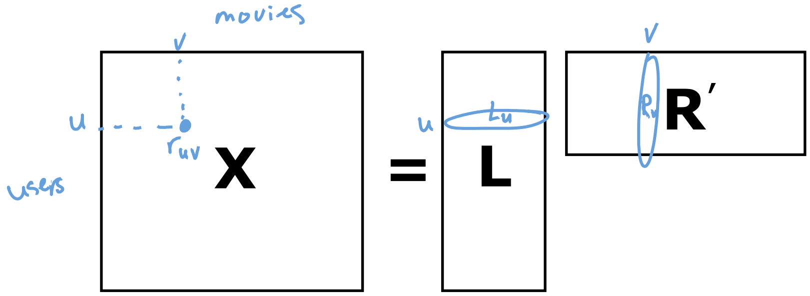
- Filling missing data? well-approx. by rank k matrix

$$\underbrace{X}_{n \times m} \approx \underbrace{L}_{n \times k} \; \underbrace{R'}_{k \times m}$$

n·m params

$$\underbrace{k\cdot n + k\cdot m}_{\text{learn this}} \ll \underbrace{n\cdot m}_{\text{predict this}}$$

Interpreting Low-Rank Matrix Completion (aka Matrix Factorization)



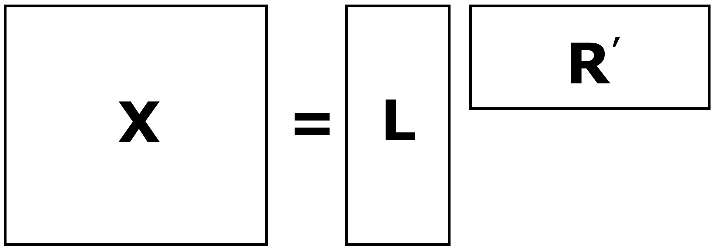
$r_{uv} \approx L_u \cdot R_v$

$L_u = [\ \ ]$ ← how much user u likes topic i

movie topic i "action"

$R_v = [\ \ ]$ ← how much movie v is about topic i

Identifiability of Factors



- If r_{uv} is described by L_u , R_v what happens if we redefine the "topics" as

  $\tilde{L}_u = L_u \Lambda \qquad \tilde{R}_v = R_v \Lambda$ where $\Lambda \Lambda^T = I$ (orthonormal)

- Then, $\tilde{L}_u \cdot \tilde{R}_v = L_u \Lambda \Lambda^T R_v^T = L_u \cdot R_v \approx r_{uv}$

  invariant to orthonormal transformations!

Matrix Completion via Rank Minimization

- Given observed values: $(u, v, r_{uv}) \in X$ some $r_{uv} = ?$
- Find matrix ${}_{n}\overset{m}{\Theta}$ ← filled in (not sparse)
- Such that: $\underbrace{\Theta_{uv} = r_{uv}}_{\text{fit } r_{uv} \neq ? \text{ perfectly}} \quad \forall\, r_{uv} \neq ?$
- But... want low-rank Θ ★
- Introduce bias:

  $$\min_{\Theta} \operatorname{rank}(\Theta) \quad \text{s.t. } \Theta_{uv} = r_{uv} \quad \forall\, r_{uv} \neq ?$$

- Two issues:
  - NP-hard
  - you can't hope to get exact matching

Approximate Matrix Completion

- Minimize squared error:
  - (Other loss functions are possible)

$$\min_{\theta} \sum_{(u,v):r_{uv}\neq ?} (\theta_{uv} - r_{uv})^2$$ allow for some error

- Choose rank *k:*

$${}_{n}\overset{m}{\theta} = {}_{n}\overset{k}{L}\,\overset{m}{R'}$$ ← fix rank k

- Optimization problem:

$$\min_{L,R} \sum_{r_{uv}\neq ?} (L_u \cdot R_v - r_{uv})^2$$

non-convex opt. problem ... local optima only

Coordinate Descent for Matrix Factorization

$$\min_{L,R} \sum_{(u,v):r_{uv}\neq ?} (L_u \cdot R_v - r_{uv})^2$$

- Fix movie factors R, optimize for user factors L
- First observation:

$$\min_{L_1,\dots,L_n} \sum_{(u,v):r_{uv}\neq ?} (L_u \cdot R_v - r_{uv})^2$$

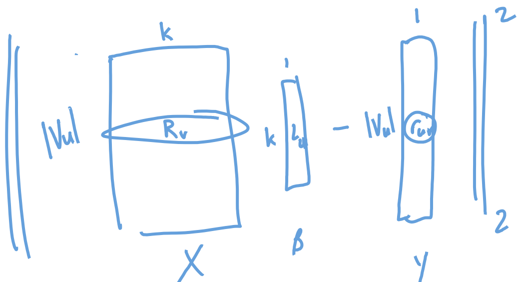
$V_u \triangleq$ set of movies user u rated

$$= \min_{L_1,\dots,L_n} \sum_u \sum_{v\in V_u} (L_u \cdot R_v - r_{uv})^2$$ ← ind. opt. problem for each user

$$= \sum_u \min_{L_u} \sum_{v\in V_u} (L_u \cdot R_v - r_{uv})^2$$ ← data parallel problem

Minimizing Over User Factors

- For each user *u*: $\min_{L_u} \sum_{v \in V_u} (L_u \cdot R_v - r_{uv})^2$
- In matrix form:



Think of as

$\|X\beta - y\|_2^2$

normal LS problem

- Second observation: Solve by

– matrix inversion

– gradient methods

Coordinate Descent for Matrix Factorization: Alternating Least-Squares

$$\min_{L,R} \sum_{(u,v):r_{uv} \neq ?} (L_u \cdot R_v - r_{uv})^2$$

- Fix movie factors, optimize for user factors
  - Independent least-squares over users $\min_{L_u} \sum_{v \in V_u} (L_u \cdot R_v - r_{uv})^2$
- Fix user factors, optimize for movie factors
  - Independent least-squares over movies $\min_{R_v} \sum_{u \in U_v} (L_u \cdot R_v - r_{uv})^2$
- System may be underdetermined:
- Converges to