LLORMA: Local Low-Rank Matrix Approximation

Joonseok Lee, Seungyeon Kim, Guy Lebanon, Yoram Singer

2013/06/19

Matrix Completion Problem

- Problem: given a partially-observed noisy matrix M, we are asked to approximately complete it.

- Application: recommendation systems
  - $M_{u,i}$ is a rating on item i by user u.
  - Sparse in nature.
  - We want to estimate unseen ratings.

Items

Users

		3		
3				5
	5		4	
		2	5	
1			2	

Georgia Institute of Technology Google

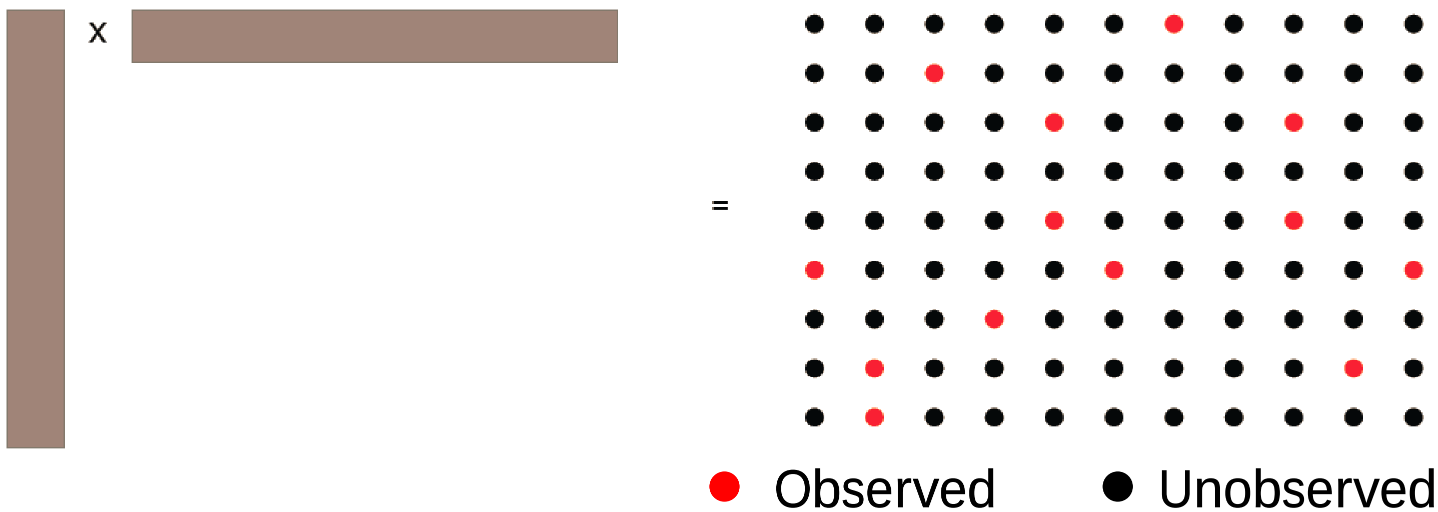
Matrix Completion Problem

- Common practice: low-rank assumption.

$$M \approx UV^T \in \mathbb{R}^{n_1 \times n_2}, \qquad U \in \mathbb{R}^{n_1 \times r}$$

$$V \in \mathbb{R}^{n_2 \times r}$$

$$r \ll \min(n_1, n_2)$$

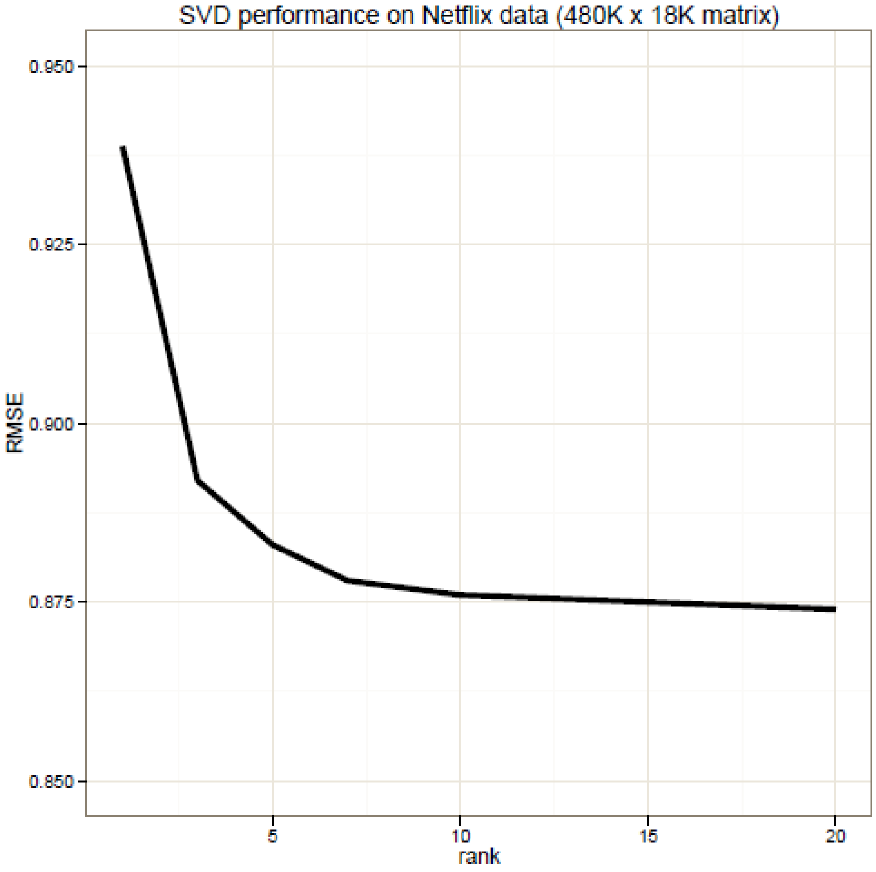


Matrix Completion Problem

- Incomplete SVD:
  - Minimizing Frobenius norm

$$\min_{U,V} \sum_{(u,i)\in A} \left([UV^T]_{u,i} - M_{u,i}\right)^2$$

  - Diminishing returns, as rank increases.



Georgia Institute of Technology Google

Matrix Completion Problem

- Diminishing returns. Why?
  - H1: M has low rank; diminishing returns reflect best possible prediction.
  - H2: M has high rank; diminishing returns due to over-fitting, or convergence to a poor local maximum.

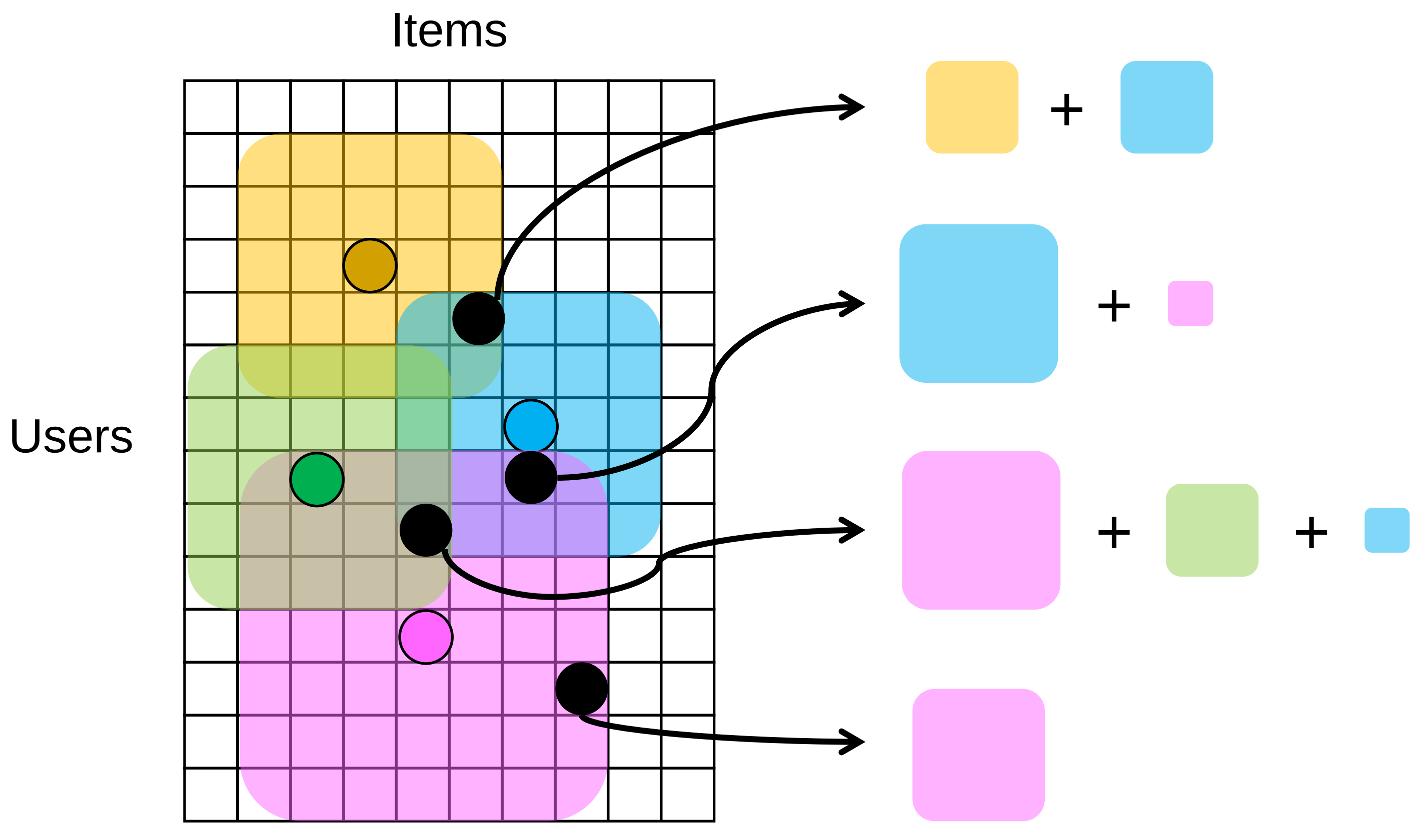
- Contribution: in recommendation systems,
  - H2 is correct.
  - H1 is incorrect globally, but it’s correct locally.

Local Low-Rank Assumption

- Instead of the global low-rank assumption, we assume that the matrix is only locally of low-rank.

- Locality?
  - A sub-matrix with similar users (rows), items (columns).
  - This sub-matrix is in low-rank.
  - The whole rating matrix is expressed with multiple local matrices.
  - SVD with non-parametric smoothing significantly postpones diminishing returns.

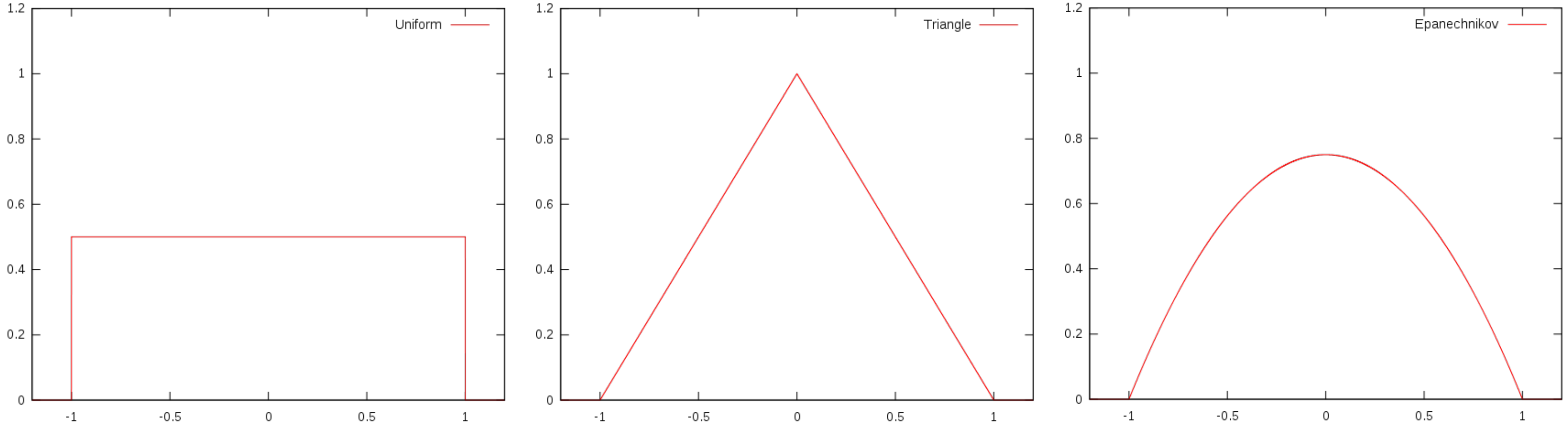
Illustration



Further Assumptions

- Local models are assumed to vary slowly.
  - For similar inputs, their estimations are also similar.

- Similar users/items are applied by smoothing kernels.



Learning Local Models

- Incomplete SVD: minimizing Frobenius norm

$$\min_{U,V} \sum_{(u,i)\in A} K((u^*,i^*),(u,i)) \left([UV^T]_{u,i} - M_{u,i}\right)^2$$

Global Approximation

- Learning a local model for every test point is computationally infeasible.

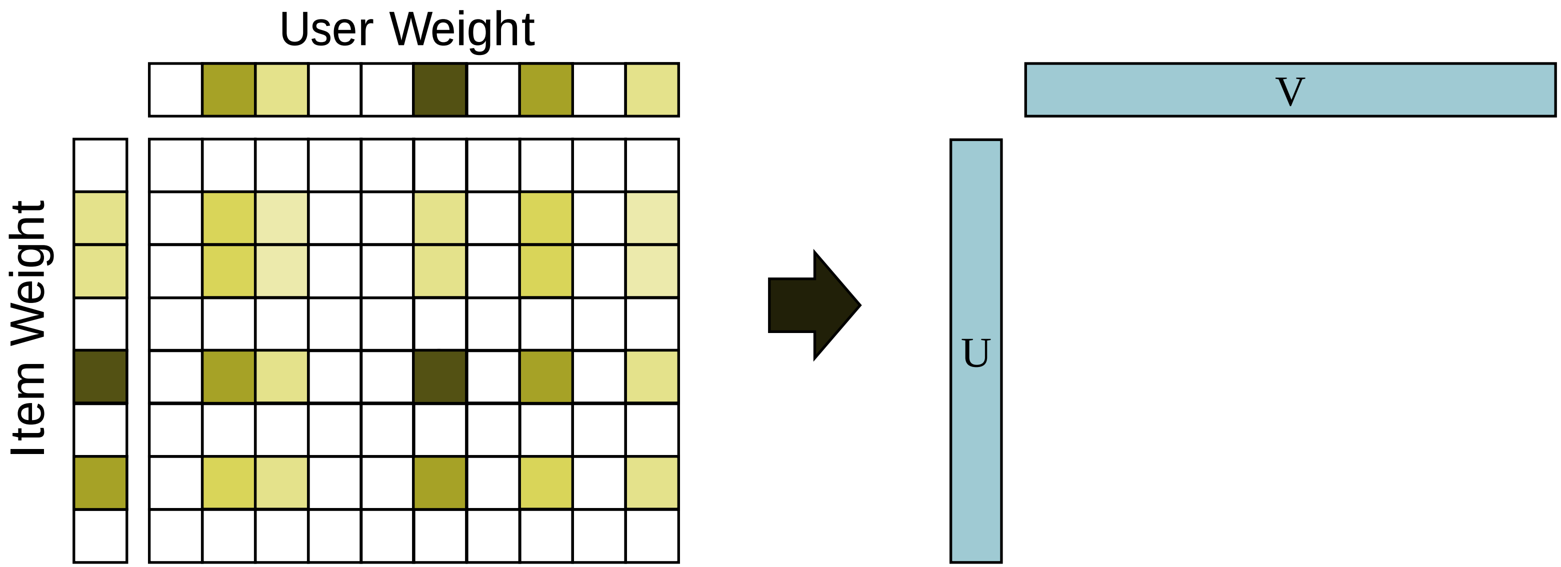
- Weighted average based on Nadaraya-Watson local regression:

$$\sum_k \frac{K((u_k, i_k), (u, i))}{\sum_j K((u_j, i_j), (u, i))} [U_k V_k^T]_{u,i}$$

  - Values of local models at indices close to queried point contribute more than indices further away from it.

Learning Algorithm

- Run in Parallel:
  - Step 1: Select an anchor point.
  - Step 2: Calculate user/item weight with kernel smoothing.
  - Step 3: Solve a weighted incomplete SVD problem.



Nuclear Norm Minimization

- Nuclear norm: sum of singular values.
  - a good surrogate for $\min_X rank(X)$. (Compressed Sensing)

- An alternative matrix completion:

$$\min_X \sum_i \sigma_i(X) \quad \text{s.t.} \quad \sum_{(u,i)\in A} (X_{u,i} - M_{u,i})^2 < \alpha$$

- Local variation:

$$\min_X \sum_i \sigma_i(X) \quad \text{s.t.} \quad \sum_{(u,i)\in A} K((u^*, i^*), (u, i))(X_{u,i} - M_{u,i})^2 < \alpha$$

Georgia Institute of Technology Google

Nuclear Norm Minimization

- Pros
  - Convex problem.
  - No need to specify rank in advance.
- Cons
  - Not scalable due to computational overhead.

Theoretical Bound

- [Candés & Plan, 2010]
  - If M has sufficient samples ($m \geq C\,\mu^2 rn \log^6 n$), (*)
  - and, the observed entries are distorted by noise Z, with $|Z| \leq \delta$,

$$\|\hat{M} - M\|_F \leq C\sqrt{\frac{\min(n_1, n_2)}{p}}\delta$$

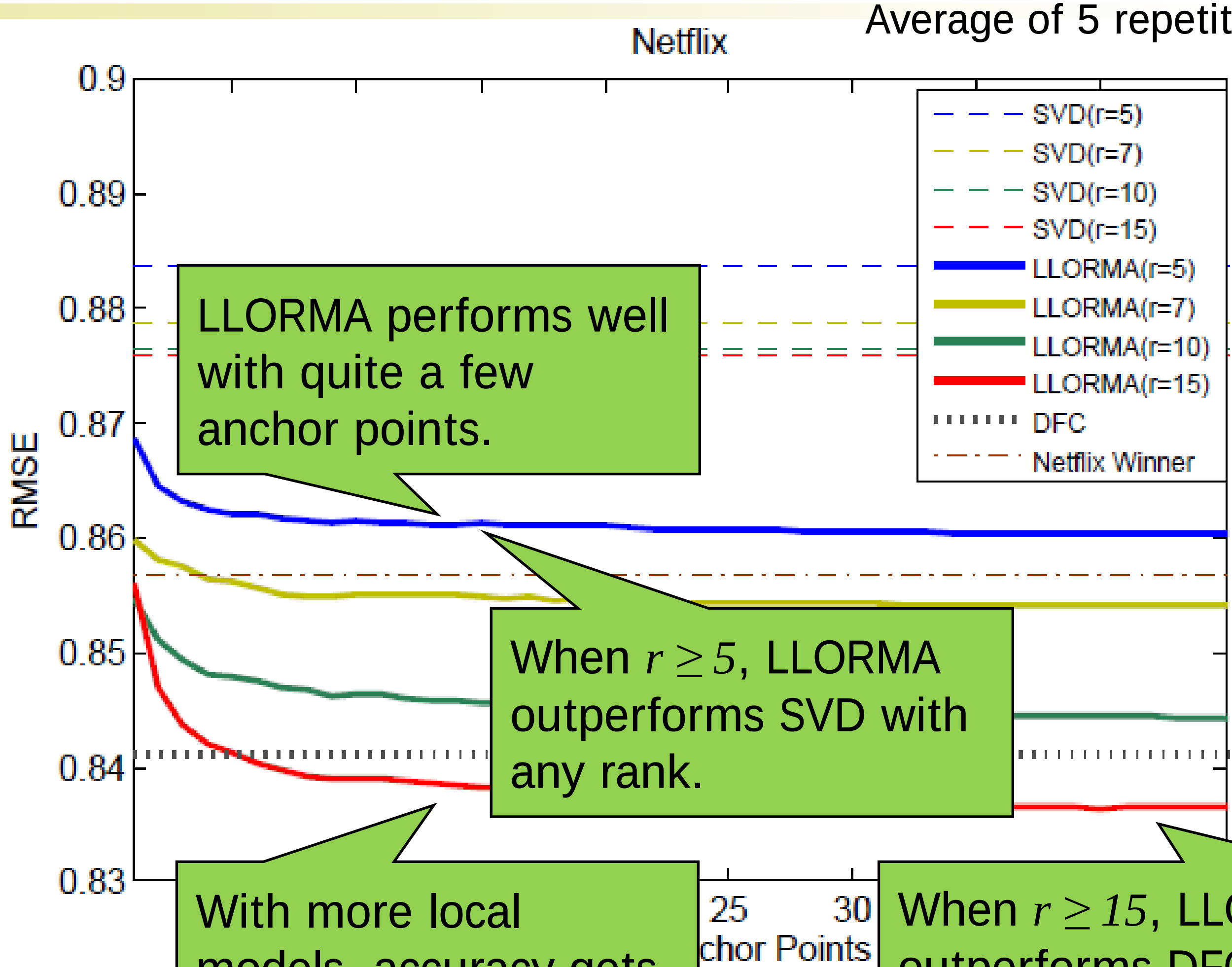
with high probability, where p is the observed ratio.

- Our extension
  - If (*) holds within a neighborhood (for a local rank r),
  - then with high probability the local completion error is bounded by

$$C_1 h^{\beta+1/2}\sqrt{\frac{2n}{m} + \frac{1}{n}} + \frac{C_2 h^{\beta+1}}{n}$$

Experimental Result

Average of 5 repetitions



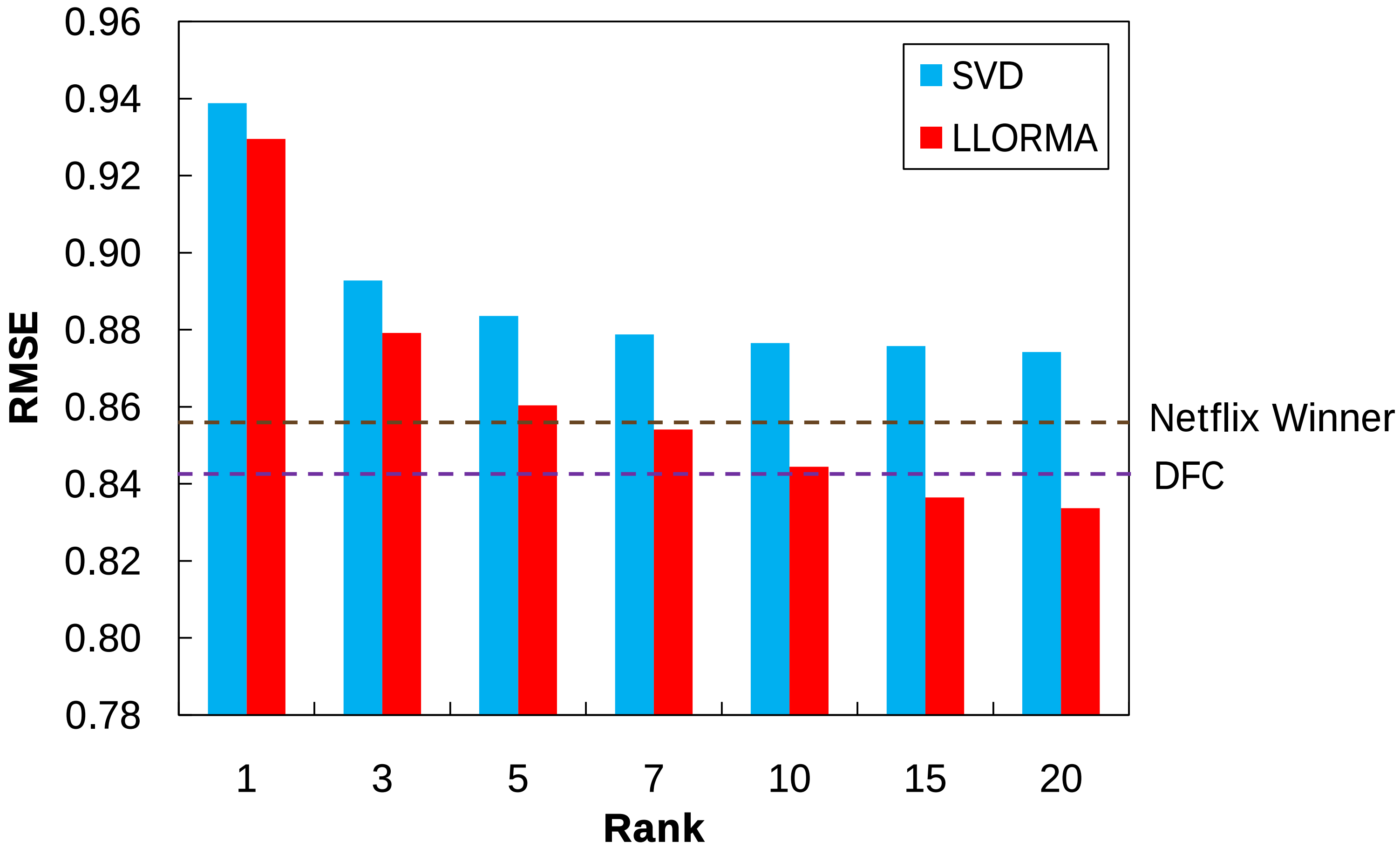
LLORMA performs well with quite a few anchor points.

When $r \geq 5$, LLORMA outperforms SVD with any rank.

With more local models, accuracy gets better.

When $r \geq 15$, LLORMA outperforms DFC.

Experimental Result



Comparison to Ensemble

- LLORMA approximates a matrix via convex combination of K local models, with input-dependent weights.

$$F(u,i) = \sum_k \underbrace{\frac{K((u_k,i_k),(u,i))}{\sum_j K((u_j,i_j),(u,i))}}_{w_k(u,\,i)} f_k(u,i)$$

  - Related to [Lee et al., NIPS 2012], [Sill et al., ArXiv 2009].

$$F(u,i) = \sum_k \underbrace{\beta_k h_k(u,i)}_{w_k(u,\,i)} f_k(u,i)$$

Comparison to Ensemble

- Ensemble CF
  - K models are assumed to be given in advance.
  - Learns only weights $\beta_k h_k(u, i)$.
  - Locality is applied only to the non-constant weights.
    - Each model may be learned in a global manner.

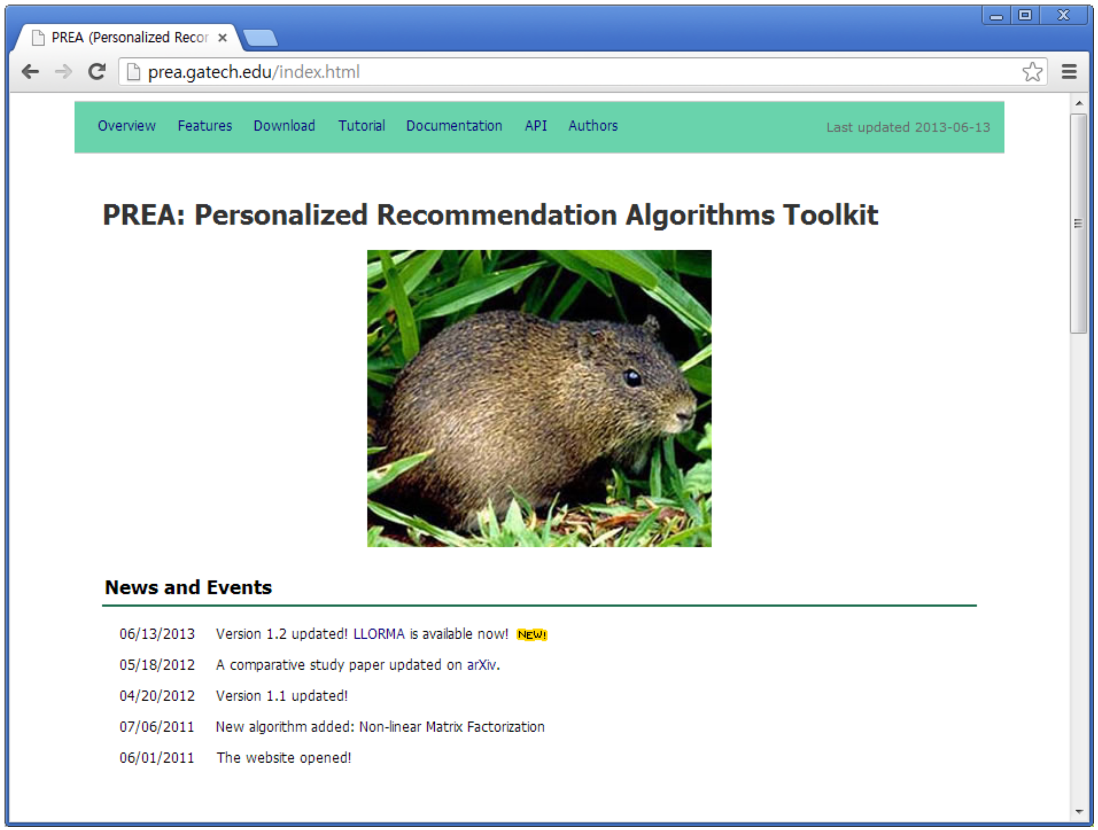
- LLORMA
  - Learns both local models and their weights at the same time.
  - The range of optimization is extended to local models as well.

Summary

- We proposed a matrix approximation under the local low-rank assumption.
- Our algorithm runs completely in parallel, leading to superior scalability.
- Experimental result supports that LLORMA outperforms several state-of-the-art methods without heavy computational overhead.
- With a formal analysis, we provide a theoretical bound in terms of matrix size, training set size, and locality.

Source code is online!

- PREA toolkit: http://prea.gatech.edu



Georgia Institute of Technology Google