Dividing, Conquering, and Mixing Matrix Factorizations

Lester Mackey[†]

Collaborators: Ameet Talwalkar[*], David Weiss[‡], Michael I. Jordan[*]

[†]Stanford University [*]UC Berkeley [‡]University of Pennsylvania

June 5, 2013

Part I

Divide-Factor-Combine

Motivation: Large-scale Matrix Completion

Goal: Estimate a matrix $\mathbf{L}_0 \in \mathbb{R}^{m\times n}$ given a subset of its entries

$$\begin{bmatrix} ? & ? & 1 & \dots & 4 \\ 3 & ? & ? & \dots & ? \\ ? & 5 & ? & \dots & 5 \end{bmatrix} \rightarrow \begin{bmatrix} 2 & 3 & 1 & \dots & 4 \\ 3 & 4 & 5 & \dots & 1 \\ 2 & 5 & 3 & \dots & 5 \end{bmatrix}$$

Examples

- Collaborative filtering: How will user i rate movie j?
  - Netflix: 10 million users, 100K DVD titles
- Ranking on the web: Is URL j relevant to user i?
  - Google News: millions of articles, millions of users
- Link prediction: Is user i friends with user j?
  - Facebook: 500 million users

Motivation: Large-scale Matrix Completion

Goal: Estimate a matrix $\mathbf{L}_0 \in \mathbb{R}^{m\times n}$ given a subset of its entries

$$\begin{bmatrix} ? & ? & 1 & \dots & 4 \\ 3 & ? & ? & \dots & ? \\ ? & 5 & ? & \dots & 5 \end{bmatrix} \rightarrow \begin{bmatrix} 2 & 3 & 1 & \dots & 4 \\ 3 & 4 & 5 & \dots & 1 \\ 2 & 5 & 3 & \dots & 5 \end{bmatrix}$$

State of the art MC algorithms

- Strong estimation guarantees
- Plagued by expensive subroutines (e.g., truncated SVD)

This talk

- Present divide and conquer approaches for scaling up any MC algorithm while maintaining strong estimation guarantees

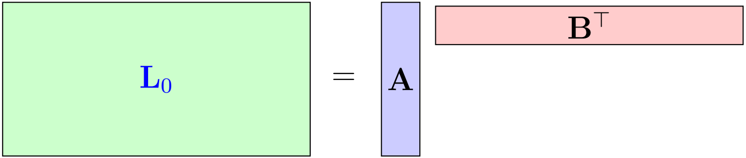
Exact Matrix Completion

Goal: Estimate a matrix $\mathbf{L}_0 \in \mathbb{R}^{m \times n}$ given a subset of its entries

Noisy Matrix Completion

Goal: Given entries from a matrix $\mathbf{M} = \mathbf{L}_0 + \mathbf{Z} \in \mathbb{R}^{m \times n}$ where $\mathbf{Z}$ is entrywise noise **and** $\mathbf{L}_0$ **has rank** $\mathbf{r} \ll m, n$, estimate $\mathbf{L}_0$

- Good news: $\mathbf{L}_0$ has $\sim (m+n)r \ll mn$ degrees of freedom



  - Factored form: $\mathbf{A}\mathbf{B}^\top$ for $\mathbf{A} \in \mathbb{R}^{m \times r}$ and $\mathbf{B} \in \mathbb{R}^{n \times r}$

- Bad news: Not all low-rank matrices can be recovered

Question: What can go wrong?

What can go wrong?

Entire column missing

$$\begin{bmatrix} 1 & 2 & ? & 3 & \dots & 4 \\ 3 & 5 & ? & 4 & \dots & 1 \\ 2 & 5 & ? & 2 & \dots & 5 \end{bmatrix}$$

- No hope of recovery!

Solution: Uniform observation model

Assume that the set of s observed entries Ω is drawn uniformly at random:

$$\Omega \sim \text{Unif}(m, n, s)$$

What can go wrong?

Bad spread of information

$$\mathbf{L} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \begin{bmatrix} 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

- Can only recover $\mathbf{L}$ if $\mathbf{L}_{11}$ is observed

Solution: Incoherence with standard basis (Candès and Recht, 2009)

A matrix $\mathbf{L} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^\top \in \mathbb{R}^{m \times n}$ with $\mathrm{rank}(\mathbf{L}) = r$ is *incoherent* if

Singular vectors are not too skewed: $\begin{cases} \max_i \|\mathbf{U}\mathbf{U}^\top \mathbf{e}_i\|^2 \leq \mu r/m \\ \max_i \|\mathbf{V}\mathbf{V}^\top \mathbf{e}_i\|^2 \leq \mu r/n \end{cases}$

and not too cross-correlated: $\|\mathbf{U}\mathbf{V}^\top\|_\infty \leq \sqrt{\frac{\mu r}{mn}}$

How do we estimate $\mathbf{L}_0$?

First attempt:

$$\begin{array}{ll} \text{minimize}_{\mathbf{A}} & \operatorname{rank}(\mathbf{A}) \\ \text{subject to} & \sum_{(i,j)\in\Omega}(\mathbf{A}_{ij} - \mathbf{M}_{ij})^2 \leq \Delta^2. \end{array}$$

Problem: Computationally intractable!

Solution: Solve **convex** relaxation (Fazel, Hindi, and Boyd, 2001; Candès and Plan, 2010)

$$\begin{array}{ll} \text{minimize}_{\mathbf{A}} & \|\mathbf{A}\|_* \\ \text{subject to} & \sum_{(i,j)\in\Omega}(\mathbf{A}_{ij} - \mathbf{M}_{ij})^2 \leq \Delta^2 \end{array}$$

where $\|\mathbf{A}\|_* = \sum_k \sigma_k(\mathbf{A})$ is the trace/nuclear norm of $\mathbf{A}$.

Questions:

- Will the nuclear norm heuristic successfully recover $\mathbf{L}_0$?
- Can nuclear norm minimization scale to large MC problems?

Noisy Nuclear Norm Heuristic: Does it work?

Yes, with high probability.

Typical Theorem

If $\mathbf{L}_0$ with rank r is incoherent, $s \gtrsim rn\log^2(n)$ entries of $\mathbf{M} \in \mathbb{R}^{m\times n}$ are observed uniformly at random, and $\hat{\mathbf{L}}$ solves the noisy nuclear norm heuristic, then

$$\|\hat{\mathbf{L}} - \mathbf{L}_0\|_F \leq f(m,n)\Delta$$

with high probability when $\|\mathbf{M} - \mathbf{L}_0\|_F \leq \Delta$.

- See Candès and Plan (2010); Mackey, Talwalkar, and Jordan (2011). See also Keshavan, Montanari, and Oh (2010); Negahban and Wainwright (2010)
- Implies **exact** recovery in the noiseless setting ($\Delta = 0$)

Noisy Nuclear Norm Heuristic: Does it scale?

Not quite...

- Standard interior point methods (Candès and Recht, 2009): $\mathrm{O}(|\Omega|(m+n)^3 + |\Omega|^2(m+n)^2 + |\Omega|^3)$
- More efficient, tailored algorithms:
  - Singular Value Thresholding (SVT) (Cai, Candès, and Shen, 2010)
  - Augmented Lagrange Multiplier (ALM) (Lin, Chen, Wu, and Ma, 2009)
  - Accelerated Proximal Gradient (APG) (Toh and Yun, 2010)
  - All require rank-k truncated SVD on **every** iteration

Take away: Many provably accurate MC algorithms are too expensive for large-scale or real-time matrix completion

Question: How can we scale up a given matrix completion algorithm and still retain estimation guarantees?

Divide-Factor-Combine (DFC)

Our Solution: Divide and conquer

1. Divide $\mathbf{M}$ into submatrices.
2. Complete each submatrix **in parallel**.
3. Combine submatrix estimates to estimate $\mathbf{L}_0$.

Advantages

- Submatrix completion is often much cheaper than completing $\mathbf{M}$
- Multiple submatrix completions can be carried out in parallel
- DFC works with **any** base MC algorithm
- With the right choice of division and recombination, yields estimation guarantees comparable to those of the base algorithm

DFC-Proj: Partition and Project

1. Randomly partition $\mathbf{M}$ into t column submatrices $\mathbf{M} = \begin{bmatrix} \mathbf{C}_1 & \mathbf{C}_2 & \cdots & \mathbf{C}_t \end{bmatrix}$ where each $\mathbf{C}_i \in \mathbb{R}^{m \times l}$
2. Complete the submatrices **in parallel** to obtain
$$\begin{bmatrix} \hat{\mathbf{C}}_1 & \hat{\mathbf{C}}_2 & \cdots & \hat{\mathbf{C}}_t \end{bmatrix}$$
   - Reduced cost: Expect t-fold speed-up per iteration
   - Parallel computation: Pay cost of one cheaper MC
3. Project submatrices onto a single low-dimensional column space
   - Estimate column space of $\mathbf{L}_0$ with column space of $\hat{\mathbf{C}}_1$
$$\hat{\mathbf{L}}^{proj} = \hat{\mathbf{C}}_1 \hat{\mathbf{C}}_1^{+} \begin{bmatrix} \hat{\mathbf{C}}_1 & \hat{\mathbf{C}}_2 & \cdots & \hat{\mathbf{C}}_t \end{bmatrix}$$
   - Common technique for randomized low-rank approximation (Frieze, Kannan, and Vempala, 1998)
   - Minimal cost: $\mathrm{O}(mk^2 + lk^2)$ where $k = \mathrm{rank}(\hat{\mathbf{L}}^{proj})$
4. **Ensemble:** Project onto column space of each $\hat{\mathbf{C}}_j$ and average

DFC: Does it work?

Yes, with high probability.

Theorem (Mackey, Talwalkar, and Jordan, 2011)

If $\mathbf{L}_0$ with rank r is incoherent and $s = \omega(r^2 n \log^2(n)/\epsilon^2)$ entries of $\mathbf{M} \in \mathbb{R}^{m \times n}$ are observed uniformly at random, then $l = o(n)$ random columns suffice to have

$$\|\hat{\mathbf{L}}^{proj} - \mathbf{L}_0\|_F \leq (2+\epsilon) f(m,n) \Delta$$

with high probability when $\|\mathbf{M} - \mathbf{L}_0\|_F \leq \Delta$ and the noisy nuclear norm heuristic is used as a base algorithm.

- Can sample vanishingly small fraction of columns ($l/n \to 0$)
- Implies exact recovery for noiseless ($\Delta = 0$) setting

DFC Estimation Error

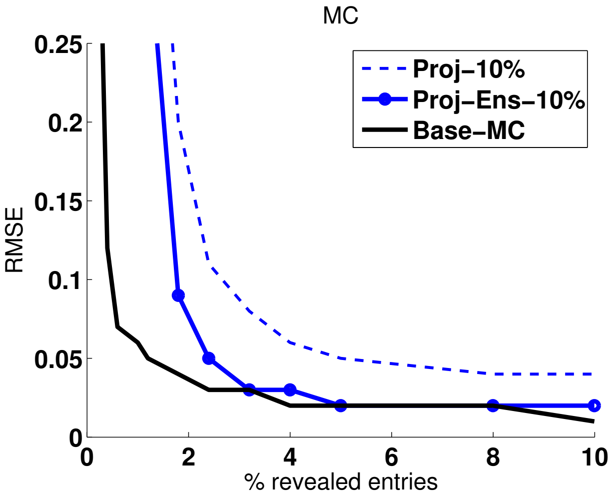


Figure : Estimation error of DFC and base algorithm (APG) with $m = 10K$ and $r = 10$.

DFC Speed-up

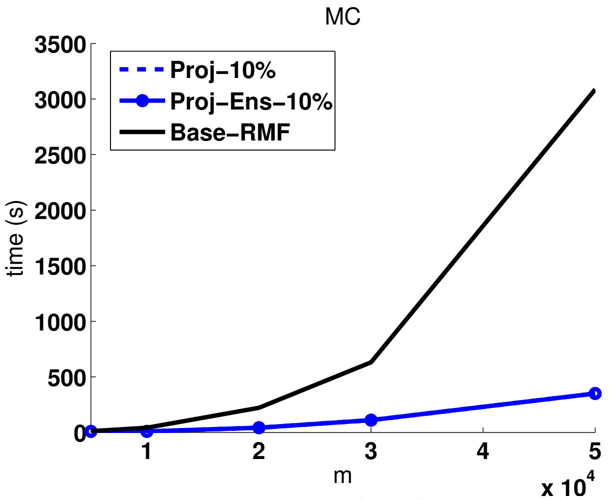


Figure : Speed-up over base algorithm (APG) for random matrices with $r = 0.001m$ and 4% of entries revealed.

Application: Collaborative filtering

Task: Given a sparsely observed matrix of user-item ratings, predict the unobserved ratings

Challenges

- Full-rank rating matrix
- Noisy, non-uniform observations

The Data

- **Netflix Prize Dataset**[1]
  - 100 million ratings in $\{1, \ldots, 5\}$
  - 17,770 movies, 480,189 users

[1] `http://www.netflixprize.com/`

Application: Collaborative filtering

Method	Netflix	
	RMSE	Time
Base algorithm (APG)	0.8433	2653.1s
DFC-Proj-25%	0.8436	689.5s
DFC-Proj-10%	0.8484	289.7s
DFC-Proj-Ens-25%	0.8411	689.5s
DFC-Proj-Ens-10%	0.8433	289.7s

Robust Matrix Factorization

Goal: Given a matrix $\mathbf{M} = \mathbf{L}_0 + \mathbf{S}_0 + \mathbf{Z}$ where $\mathbf{L}_0$ is low-rank, $\mathbf{S}_0$ is sparse, and $\mathbf{Z}$ is entrywise noise, recover $\mathbf{L}_0$ (Chandrasekaran, Sanghavi, Parrilo, and Willsky, 2009; Candès, Li, Ma, and Wright, 2011; Zhou, Li, Wright, Candès, and Ma, 2010)

$\mathbf{M}$ $\mathbf{L}_0$ $\mathbf{S}_0$

- $\mathbf{S}_0$ can be viewed as an outlier/gross corruption matrix
  - Ordinary PCA breaks down in this setting
- **Harder than MC:** outlier locations are unknown
- **More expensive than MC:** dense, fully observed matrices

Application: Video background modeling

Task

- Each video frame forms one column of matrix $\mathbf{M}$
- Decompose $\mathbf{M}$ into stationary background $\mathbf{L}_0$ and moving foreground objects $\mathbf{S}_0$

$\mathbf{M}$

$\mathbf{L}_0$

$\mathbf{S}_0$

Challenges

- Video is noisy
- Foreground corruption is often clustered, not uniform

Part II

Mixed Membership Matrix Factorization

Matrix Completion

Learning from Pairs

- Given two sets of objects
  - Set of users and set of items
- Observe labeled object pairs
  - User u gave item j a rating r_{uj} of 5
- Predict labels of unobserved pairs
  - How will user u rate item k?

Examples

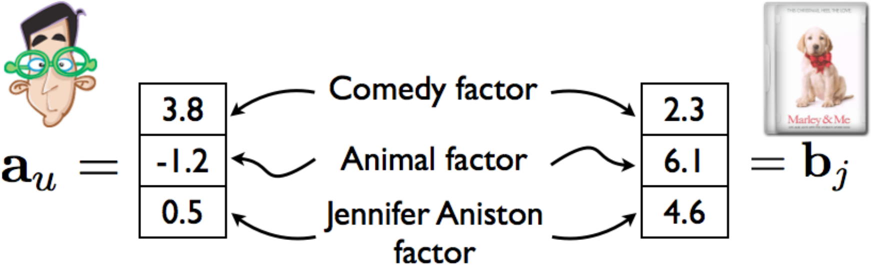
- Movie rating prediction in collaborative filtering
  - How will user u rate movie j?
- Click prediction in web search
  - Will user u click on URL j?
- Link prediction in a social network
  - Is user u friends with user j?

Prior Models for Matrix Completion

Latent Factor Modeling / Matrix Factorization

Rennie & Srebro (2005); DeCoste (2006); Salakhutdinov & Mnih (2008); Takács et al. (2009); Lawrence & Urtasun (2009)

- Associate latent factor vector, $\mathbf{a}_u \in \mathbb{R}^D$, with each user u
- Associate latent factor vector, $\mathbf{b}_j \in \mathbb{R}^D$, with each item j
- Generate expected rating via inner product



$$\mathbb{E}(r_{uj}) = \mathbf{a}_u \cdot \mathbf{b}_j = 3$$

Prior Models for Matrix Completion

Latent Factor Modeling / Matrix Factorization

Rennie & Srebro (2005); DeCoste (2006); Salakhutdinov & Mnih (2008); Takács et al. (2009); Lawrence & Urtasun (2009)

- Associate latent factor vector, $\mathbf{a}_u \in \mathbb{R}^D$, with each user u
- Associate latent factor vector, $\mathbf{b}_j \in \mathbb{R}^D$, with each item j
- Generate expected rating via inner product: $\mathbb{E}(r_{uj}) = \mathbf{a}_u \cdot \mathbf{b}_j$

Pro: State-of-the-art predictive performance

Con: Fundamentally static rating mechanism

- Assumes user u rates according to $\mathbf{a}_u$, regardless of context
- In reality, dyadic interactions are heterogeneous
  - User's ratings may be influenced by instantaneous mood
  - Distinct users may share single account or web browser

Prior Models for Matrix Completion

Mixed Membership Topic Modeling

Airoldi, Blei, Fienberg, and Xing (2008); Porteous, Bart, and Welling (2008)

- Each user u maintains distribution over topics, $\theta_u^U \in \mathbb{R}^{K^U}$
- Each item j maintains distribution over topics, $\theta_j^M \in \mathbb{R}^{K^M}$
- Expected rating $\mathbb{E}(r_{uj})$ determined by *interaction-specific* topics sampled from user and item topic distributions

$$\mathbb{E}(r_{uj}) = f(z_{uj}^U, z_{uj}^M)$$

Prior Models for Matrix Completion

Mixed Membership Topic Modeling

Airoldi, Blei, Fienberg, and Xing (2008); Porteous, Bart, and Welling (2008)

- Each user u maintains distribution over topics, $\theta_u^U \in \mathbb{R}^{K^U}$
- Each item j maintains distribution over topics, $\theta_j^M \in \mathbb{R}^{K^M}$
- Expected rating $\mathbb{E}(r_{uj})$ determined by *interaction-specific* topics sampled from user and item topic distributions

Pro: Context-sensitive clustering

- User moods: in the mood for comedy vs. romance
- Item contexts: opening night vs. in high school classroom
- Multiple raters per account: parent vs. child

Con: Purely groupwise interactions

- Assumes user and item interact only through their topics
- Relatively poor predictive performance

Mixed Membership Matrix Factorization (M^3F)

Goal: Leverage the complementary strengths of latent factor models and mixed membership models for improved matrix completion

General M^3F Framework (Mackey, Weiss, and Jordan, 2010):

- Users and items endowed both with latent factor vectors ($\mathbf{a}_u$ and $\mathbf{b}_j$) and with topic distribution parameters (θ_u^U and θ_j^M)
- To rate an item
  - User u draws topic i from θ_u^U
  - Item j draws topic k from θ_j^M
  - Expected rating

$$\mathbb{E}(r_{uj}) = \underbrace{\mathbf{a}_u \cdot \mathbf{b}_j}_{\text{static base rating}} + \underbrace{\beta_{uj}^{ik}}_{\text{context-sensitive bias}}$$

- M^3F models differ in specification of β_{uj}^{ik}
- Fully Bayesian framework

Mixed Membership Matrix Factorization (M^{3}F)

Goal: Leverage the complementary strengths of latent factor models and mixed membership models for improved matrix completion

General M^{3}F Framework (Mackey, Weiss, and Jordan, 2010):

- M^{3}F models differ in specification of β_{uj}^{ik}

Specific M^{3}F Models:

- M^{3}F Topic-Indexed Bias Model
- M^{3}F Topic-Indexed Factor Model

M^{3}F Models

M^{3}F Topic-Indexed Bias Model (M^{3}F-TIB)

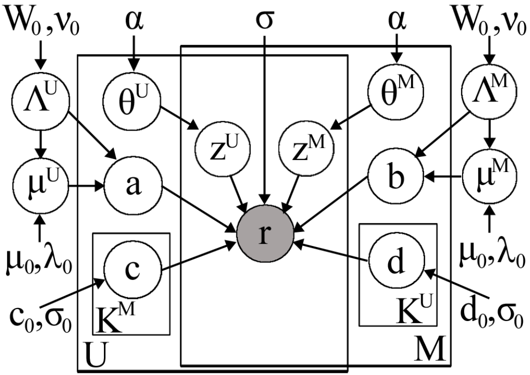
- Contextual bias decomposes into latent user and latent item bias

$$\beta_{uj}^{ik} = c_u^k + d_j^i$$

- Item bias d_j^i influenced by user topic i
  - Group predisposition toward liking/disliking item j
  - Captures polarizing *Napoleon Dynamite* effect
    - Certain movies provoke strongly differing reactions from otherwise similar users
- User bias c_u^k influenced by item topic k
  - Predisposition of u toward liking/disliking item group

M^{3}F Inference and Prediction

Goal: Predict unobserved labels given labeled pairs



- Posterior inference over latent topics and parameters intractable
- Use block Gibbs sampling with closed form conditionals
  - User parameters sampled in parallel (same for items)
  - Interaction-specific topics sampled in parallel

M^{3}F Inference and Prediction

Goal: Predict unobserved labels given labeled pairs

- Bayes optimal prediction under root mean squared error (RMSE)

$$\textbf{M}^3\textbf{F-TIB:}\ \frac{1}{T}\sum_{t=1}^{T}\left(\mathbf{a}_u^{(t)}\cdot\mathbf{b}_j^{(t)}+\sum_{k=1}^{K^M}c_u^{k(t)}\theta_{jk}^{M(t)}+\sum_{i=1}^{K^U}d_j^{i(t)}\theta_{ui}^{U(t)}\right)$$

Experimental Evaluation

The Setup

- Evaluate rating prediction performance on Netflix Prize Dataset[2]
  - 100 million ratings in $\{1, \ldots, 5\}$
  - 17,770 movies, 480,189 users
  - RMSE as primary evaluation metric
- Compare to state-of-the-art latent factor model
  - Bayesian Probabilistic Matrix Factorization[3] (BPMF)
    - M^{3}F reduces to BPMF when no topics are sampled
- Matlab/MEX implementation on dual quad-core CPUs

[2] `http://www.netflixprize.com/`

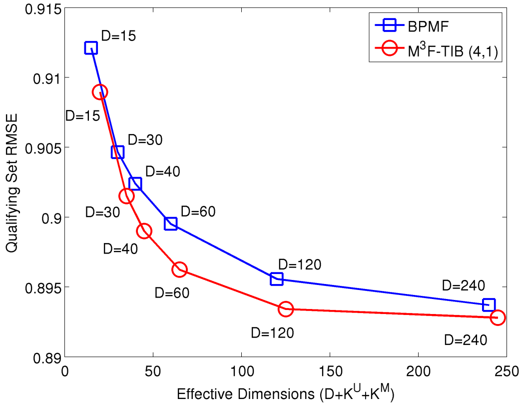
[3] Salakhutdinov and Mnih (2008)

Netflix Prize Data

Question: How does performance vary with latent dimensionality?

- Contrast M^{3}F-TIB $(K^U, K^M) = (4, 1)$ with BPMF
- 500 Gibbs samples for M^{3}F-TIB and BPMF

Method	RMSE	Time
BPMF/15	0.9121	27.8s
TIB/15	**0.9090**	46.3s
BPMF/30	0.9047	38.6s
TIB/30	**0.9015**	56.9s
BPMF/40	0.9027	48.3s
TIB/40	**0.8990**	70.5s
BPMF/60	0.9002	94.3s
TIB/60	**0.8962**	97.0s
BPMF/120	0.8956	273.7s
TIB/120	**0.8934**	285.2s
BPMF/240	0.8938	1152.0s
TIB/240	**0.8929**	1158.2s



Stratification

Question: Where are improvements over BPMF being realized?

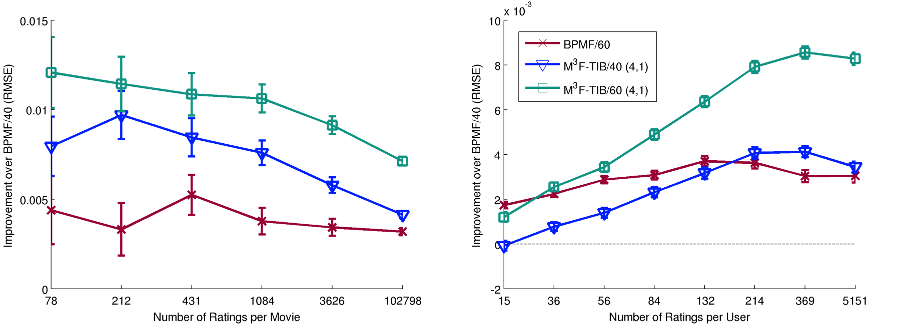


Figure : RMSE improvements over BPMF/40 on the Netflix Prize as a function of movie or user rating count. Left: Each bin represents 1/6 of the movie base. Right: Each bin represents 1/8 of the user base.

The *Napoleon Dynamite* Effect

Question: Do M^{3}F models capture polarization effects?

Table : Top 200 Movies from the Netflix Prize dataset with the highest and lowest cross-topic variance in $\mathbb{E}(d^i_j|\mathbf{r}^{(\mathrm{v})})$.

Movie Title	$\mathbb{E}(d^i_j\|\mathbf{r}^{(\mathrm{v})})$
Napoleon Dynamite	-0.11 $\pm$ 0.93
Fahrenheit 9/11	-0.06 $\pm$ 0.90
Chicago	-0.12 $\pm$ 0.78
The Village	-0.14 $\pm$ 0.71
Lost in Translation	-0.02 $\pm$ 0.70
LotR: The Fellowship of the Ring	0.15 $\pm$ 0.00
LotR: The Two Towers	0.18 $\pm$ 0.00
LotR: The Return of the King	0.24 $\pm$ 0.00
Star Wars: Episode V	0.35 $\pm$ 0.00
Raiders of the Lost Ark	0.29 $\pm$ 0.00

Conclusions

M^3F framework for matrix completion

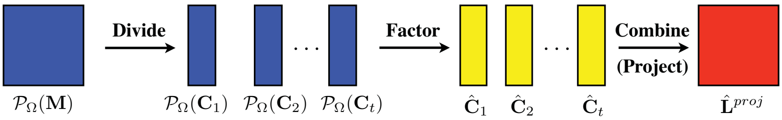
- Strong predictive performance and static specificity of latent factor models
- Clustered context-sensitivity of mixed membership topic models
- Outperforms pure latent factor modeling while fitting fewer parameters
- Greatest improvements for high-variance, sparsely rated items

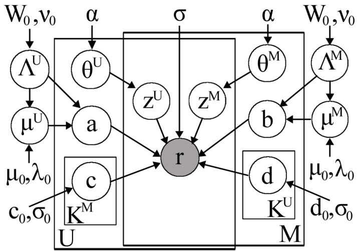
Future work

- Modeling user choice: missingness is informative
- Nonparametric priors on topic parameters
- Alternative approaches to inference

The End

Thanks!





References I

Airoldi, E., Blei, D., Fienberg, S., and Xing, E. Mixed membership stochastic blockmodels. *JMLR*, 9:1981–2014, 2008.

Cai, J. F., Candès, E. J., and Shen, Z. A singular value thresholding algorithm for matrix completion. *SIAM Journal on Optimization*, 20(4), 2010.

Candès, E. J. and Recht, B. Exact matrix completion via convex optimization. *Foundations of Computational Mathematics*, 9 (6):717–772, 2009.

Candès, E. J., Li, X., Ma, Y., and Wright, J. Robust principal component analysis? *Journal of the ACM*, 58(3):1–37, 2011.

Candès, E.J. and Plan, Y. Matrix completion with noise. *Proceedings of the IEEE*, 98(6):925 –936, 2010.

Chandrasekaran, V., Sanghavi, S., Parrilo, P. A., and Willsky, A. S. Sparse and low-rank matrix decompositions. In *Allerton Conference on Communication, Control, and Computing*, 2009.

DeCoste, D. Collaborative prediction using ensembles of maximum margin matrix factorizations. In *ICML*, 2006.

Fazel, M., Hindi, H., and Boyd, S. P. A rank minimization heuristic with application to minimum order system approximation. In *In Proceedings of the 2001 American Control Conference*, pp. 4734–4739, 2001.

Frieze, A., Kannan, R., and Vempala, S. Fast Monte Carlo algorithms for finding low-rank approximations. In *Foundations of Computer Science*, 1998.

Keshavan, R. H., Montanari, A., and Oh, S. Matrix completion from noisy entries. *Journal of Machine Learning Research*, 99: 2057–2078, 2010.

Lawrence, N.D. and Urtasun, R. Non-linear matrix factorization with Gaussian processes. In *ICML*, 2009.

Lin, Z., Chen, M., Wu, L., and Ma, Y. The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices. UIUC Technical Report UILU-ENG-09-2215, 2009.

Mackey, L., Weiss, D., and Jordan, M. I. Mixed membership matrix factorization. In *ICML*, June 2010.

Mackey, L., Talwalkar, A., and Jordan, M. I. Divide-and-conquer matrix factorization. In Shawe-Taylor, J., Zemel, R. S., Bartlett, P. L., Pereira, F. C. N., and Weinberger, K. Q. (eds.), *Advances in Neural Information Processing Systems 24*, pp. 1134–1142. 2011.

Negahban, S. and Wainwright, M. J. Restricted strong convexity and weighted matrix completion: Optimal bounds with noise. `arXiv:1009.2118v2[cs.IT]`, 2010.

References II

Porteous, I., Bart, E., and Welling, M. Multi-HDP: A non parametric Bayesian model for tensor factorization. In *AAAI*, 2008.

Rennie, J. and Srebro, N. Fast maximum margin matrix factorization for collaborative prediction. In *ICML*, 2005.

Salakhutdinov, R. and Mnih, A. Bayesian probabilistic matrix factorization using Markov chain Monte Marlo. In *ICML*, 2008.

Takács, G., Pilászy, I., Németh, B., and Tikk, D. Scalable collaborative filtering approaches for large recommender systems. *JMLR*, 10:623–656, 2009.

Toh, K. and Yun, S. An accelerated proximal gradient algorithm for nuclear norm regularized least squares problems. *Pacific Journal of Optimization*, 6(3):615–640, 2010.

Zhou, Z., Li, X., Wright, J., Candès, E. J., and Ma, Y. Stable principal component pursuit. In *IEEE International Symposium on Information Theory Proceedings (ISIT)*, pp. 1518 –1522, 2010.