Riemannian gossip algorithms for decentralized matrix completion

Hiroyuki Kasai[†], Bamdev Mishra[‡], and Atul Saroop[‡]

[†]The University of Electro-Communications, Japan

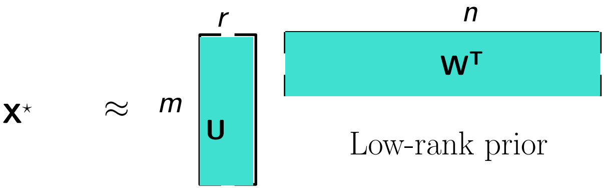
[‡]Amazon Development Centre India, India

IEICE meeting 2016

The matrix completion problem

n Users

$$m \text{ Movies} \begin{bmatrix} ? & ? & * & ? \\ * & * & ? & * \\ ? & * & * & ? \\ * & ? & * & ? \end{bmatrix}$$



$$(n+m-r)r, \; r \ll (m, n)$$

U and **W** factor matrices.
[Netflix Challenge, 2006]

Our interest is to look at the decentralized scenario

$$
\begin{array}{c}
\begin{array}{cc} n_1 \text{ Users} & n_2 \text{ Users} \end{array} \\
m \text{ Movies} \left[\begin{array}{cc|cc}
? & ? & * & ? \\
* & * & ? & * \\
? & * & * & ? \\
* & ? & * & ?
\end{array}\right] \approx \mathbf{U}[\mathbf{W}_1^{\mathbf{T}} \;\; \mathbf{W}_2^{\mathbf{T}}] \\
\begin{array}{cc} \mathbf{X}_1^\star & \mathbf{X}_2^\star \end{array}
\end{array}
$$

- An agent i has access to its own data matrix $\mathbf{X}_i^\star$.
- The matrix $\mathbf{U}$ is common across all the agents.

Contributions

We develop a nonlinear gossip algorithm with minimal communication between agents.

The optimization formulation is based on a weighted combination of matrix completion and consensus terms.

We develop a parallel variant of the proposed gossip algorithm.

Paper and codes available online

at

www.bamdevmishra.com.

Outline

- Problem formulation on the Riemannian Grassmann manifold.
- Proposed gossip algorithms.
- Numerical comparisons on synthetic and Netflix data.

Outline

- Problem formulation on the Riemannian Grassmann manifold.
- Proposed gossip algorithms.
- Numerical comparisons on synthetic and Netflix data.

Batch problem formulation

$$\min_{\mathbf{U}\in \mathrm{St}(r,m)} \min_{\mathbf{W}\in\mathbb{R}^{n\times r}} \|\mathcal{P}_\Omega(\mathbf{U}\mathbf{W}^T) - \mathcal{P}_\Omega(\mathbf{X}^\star)\|_F^2.$$

$\mathbf{W} \in \mathbb{R}^{n\times r}$ and
$\mathbf{U} \in \mathrm{St}(r, m)$, the set of $m \times r$ matrices with orthonormal columns.

$\mathcal{P}_\Omega$ is the sampling operator, a convenient way to denote known entries.

Eliminate **W**

$$\min_{\mathbf{U}\in \mathrm{St}(r,m)} \min_{\mathbf{W}\in \mathbb{R}^{n\times r}} \|\mathcal{P}_\Omega(\mathbf{U}\mathbf{W}^T) - \mathcal{P}_\Omega(\mathbf{X}^\star)\|_F^2$$

$$\equiv$$

$\min_{\mathbf{U}\in \mathrm{St}(r,m)} f(\mathbf{U}, \mathbf{W}_\mathbf{U})$, a Grassmann optimization problem.

- Solve blue problem in closed form to obtain $\mathbf{W}_\mathbf{U}$.
- Final optimization problem is on Grassmann manifold, i.e., variable is ‘column space’ of **U**.

[Boumal and Absil, LAA, 2015]

Decentralized problem formulation

$\mathbf{X}^\star = [\mathbf{X}_1^\star, \mathbf{X}_2^\star, \ldots, \mathbf{X}_N^\star]$.

$$\sum_i \min_{\mathbf{U} \in \mathrm{St}(r,m), \mathbf{W}_i \in \mathbb{R}^{n_i \times r}} \frac{1}{2} \| \mathcal{P}_{\Omega_i}(\mathbf{U}{\mathbf{W}_i}^T) - \mathcal{P}_{\Omega_i}(\mathbf{X}_i^\star) \|_F^2$$

$$= \min_{\mathbf{U} \in \mathrm{St}(r,m)} \frac{1}{2} \sum_i \| \mathcal{P}_{\Omega_i}(\mathbf{U}\mathbf{W}_{i\mathbf{U}}^T) - \mathcal{P}_{\Omega_i}(\mathbf{X}_i^\star) \|_F^2,$$

where $\mathbf{W}_{i\mathbf{U}}$ is computed by agent i independently.

Although the problem is distributed, we still need to learn a common $\mathbf{U}$.

We add a consensus term to our optimization formulation

Key idea: introduce multiple copies of $\mathbf{U}$ among N agents, but allow them to reach consensus.

$$\min_{\mathbf{U}_1,\ldots,\mathbf{U}_N \in \mathrm{St}(r,m)} \frac{1}{2} \sum_i \underbrace{\|\mathcal{P}_{\Omega_i}(\mathbf{U}_i \mathbf{W}_{i\mathbf{U}_i}^T) - \mathcal{P}_{\Omega_i}(\mathbf{X}_i^\star)\|_F^2}_{\text{completion task handled by agent } i}$$
$$+ \frac{\rho}{2} \underbrace{\left(d(\mathbf{U}_1, \mathbf{U}_2)^2 + d(\mathbf{U}_2, \mathbf{U}_3)^2 + \ldots + d(\mathbf{U}_{N-1}, \mathbf{U}_N)^2\right)}_{\text{consensus among agents}}.$$

d is the Riemannian distance on the Grassmann manifold.

A large ρ trades-off completion with consensus.

Minimizing only consensus $\Rightarrow \mathbf{U}_1 = \mathbf{U}_2 = \ldots = \mathbf{U}_{N-1} = \mathbf{U}_N$.

Outline

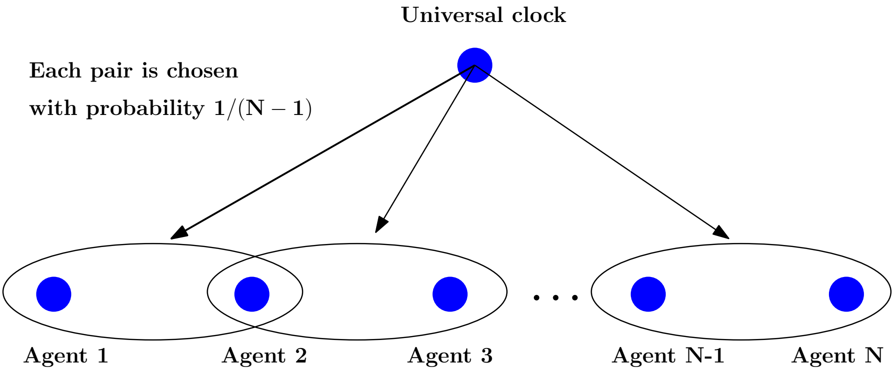
- Problem formulation on the Riemannian Grassmann manifold.
- Proposed gossip algorithms.
- Numerical comparisons on synthetic and Netflix data.

Riemannian online gossip on Grassmann

1. Agents i and $i+1$ are neighbors for all $i \leqslant N-1$. (ordering of agents)

2. At each time slot, say t, we pick an agent $i \leqslant N-1$ randomly with uniform probability. (SGD updates)
   - Equivalently, we also pick agent $i+1$ (the neighbor of agent i).
   - Agents i and $i+1$ update $\mathbf{U}_i$ and $\mathbf{U}_{i+1}$, respectively, by taking a *gradient descent step* with stepsize γ_t on Grassmann manifold.
   - $\sum \gamma_t^2 < \infty$ and $\sum \gamma_t = +\infty$.

A graphical illustration



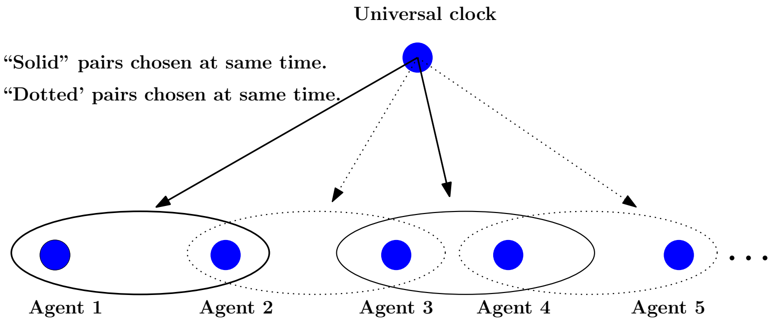
Convergence of Riemannian online gossip

Asymptotic convergence follows standard SGD analysis on manifold.

The proposed algorithm is readily implementable, e.g., with the toolbox Manopt.

[Bonnabel, IEEE TAC, 2013; Absil, Mahoney, and Sepulchre, Princeton Press, 2008; Boumal et al., JMLR, 2014]

Parallelizing Riemannian gossip with particular sampling



“Dotted’ and “solid” groups are chosen with probability $1/2$.

Convergence guarantees remain the same.

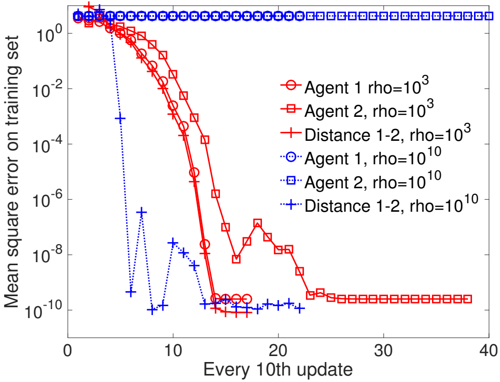
$(N-1)/2$ times faster.

Outline

- Problem formulation on the Riemannian Grassmann manifold.
- Proposed gossip algorithms.
- Numerical comparisons on synthetic and Netflix data.

Effect of ρ

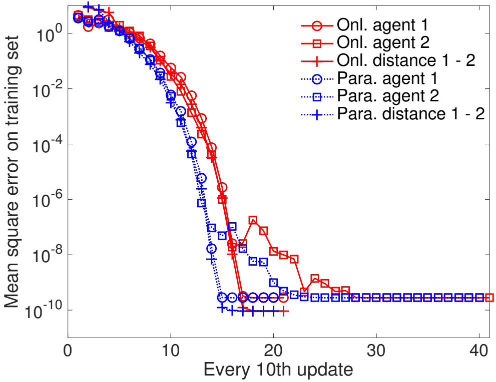
10 000$\times$100 000 matrix with $N = 6$.



A very large ρ minimizes only achieves consensus among agents.
A tuned ρ achieves both completion and consensus.

Performance of online and parallel variants

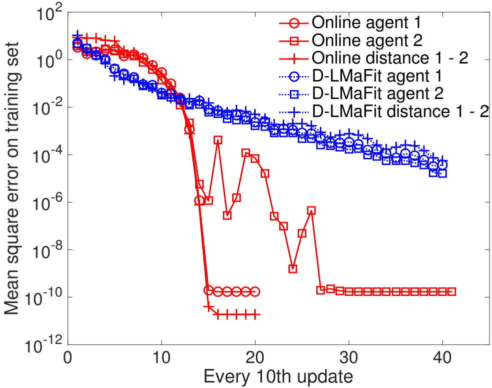
10 000×100 000 matrix with $N = 6$ and $\rho = 10^3$.



There is no loss of performance in parallelizing the updates.

Comparison with D-LMaFit

500×12000 and $N = 6$.



D-LMaFit code is not scalable to large data.

Netflix data: different number of agents

Rank $= 10$

$N = \{2, 5, 10, 15, 20\}$ agents

10 random 80/20 - train/test - 80/20 million split

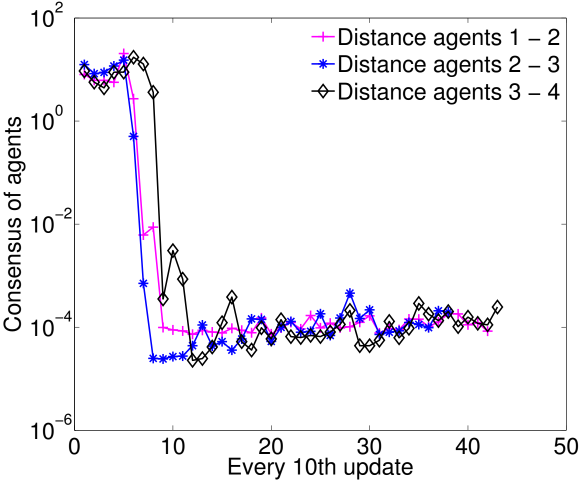
Online gossip:

	$N = 2$	$N = 5$	$N = 10$	$N = 15$	$N = 20$
Test RMSE	0.877	0.885	0.891	0.894	0.900

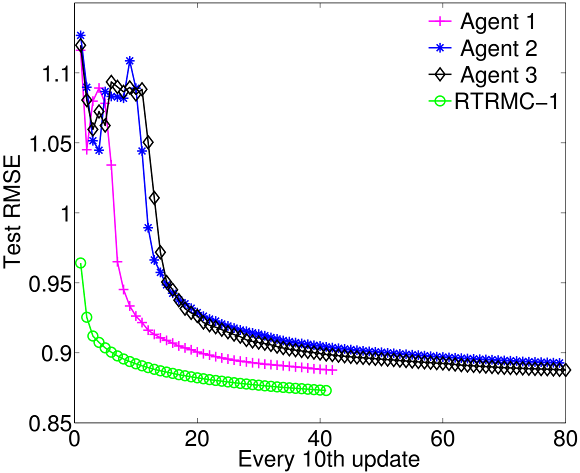
Batch gradient descent algorithm RTRMC benchmark: 0.873.

[Boumal and Absil, LAA, 2015]

Netflix data: consensus of agents



Netflix data: test RMSE with updates



Summary and future work

- We proposed a Riemannian gossip approach to the decentralized matrix completion problem.
- We minimize weighted sum of completion and consensus terms on Grassmann manifold.
- Numerical comparisons show good performance of the proposed algorithms, e.g., on Netflix dataset.
- Currently, we intend to explore asynchronous updating of agents on Grassmann manifold.

Riemannian gossip algorithms for decentralized matrix completion

Hiroyuki Kasai†, Bamdev Mishra‡, and Atul Saroop‡

†The University of Electro-Communications, Japan

‡Amazon Development Centre India, India

IEICE meeting 2016