Efficient and Practical Stochastic Subgradient Descent for Nuclear Norm Regularization

ICML2012

Authors (IBM): Haim Avron, Satyen Kale, Shiva Kasiviswanathan, Vikas Sindhwani

Presenter: Wang Yuxiang

Outline of presentation

1. Problem setup
   - ✓ Matrix completion
   - ✓ Challenges in practice
2. Stochastic Gradient Descent
   - ✓ SGD at a glance
   - ✓ SSGD for solving Nuclear Norm Minimization
   - ✓ Contributions and guarantees
3. Experiments
4. Discussion: Convex vs. Non-Convex

Problem setup

- Convex optimization

$$\min_{X \in \mathbb{R}^{m \times n}} \; f(X) + \lambda \|X\|_*$$

- f(X) is any convex loss function, $||X||_*$ is nuclear norm defined to be sum of singular values.
- Nuclear norm is used to promote low-rank solutions.
  - It is the tight convex relaxation of rank.
  - Use it as a tractable replacement of rank.

Matrix completion

- When f(X) is indicator function

$$\min_X \|X\|_*$$
$$s.t.\, P_\Omega(X - M) = 0$$

- It is shown by Candes&Tao that under certain conditions, its solution is exactly the solution of

$$\min_X \operatorname{rank}(X)$$
$$s.t.\, P_\Omega(X - M) = 0$$

Practical challenges

- Noise

$$\min_X \|P_\Omega(X - M)\|_F^2 + \lambda \|X\|_*$$

- Corruptions (RPCA with missing data)

$$\min_X \|P_\Omega(X - M)\|_1 + \lambda \|X\|_*$$

- Noise and corruptions

$$\min_{X,E} \|P_\Omega(X - M - E)\|_F^2 + \lambda_1 \|X\|_* + \lambda_2 \|E\|_1$$

Practical challenges

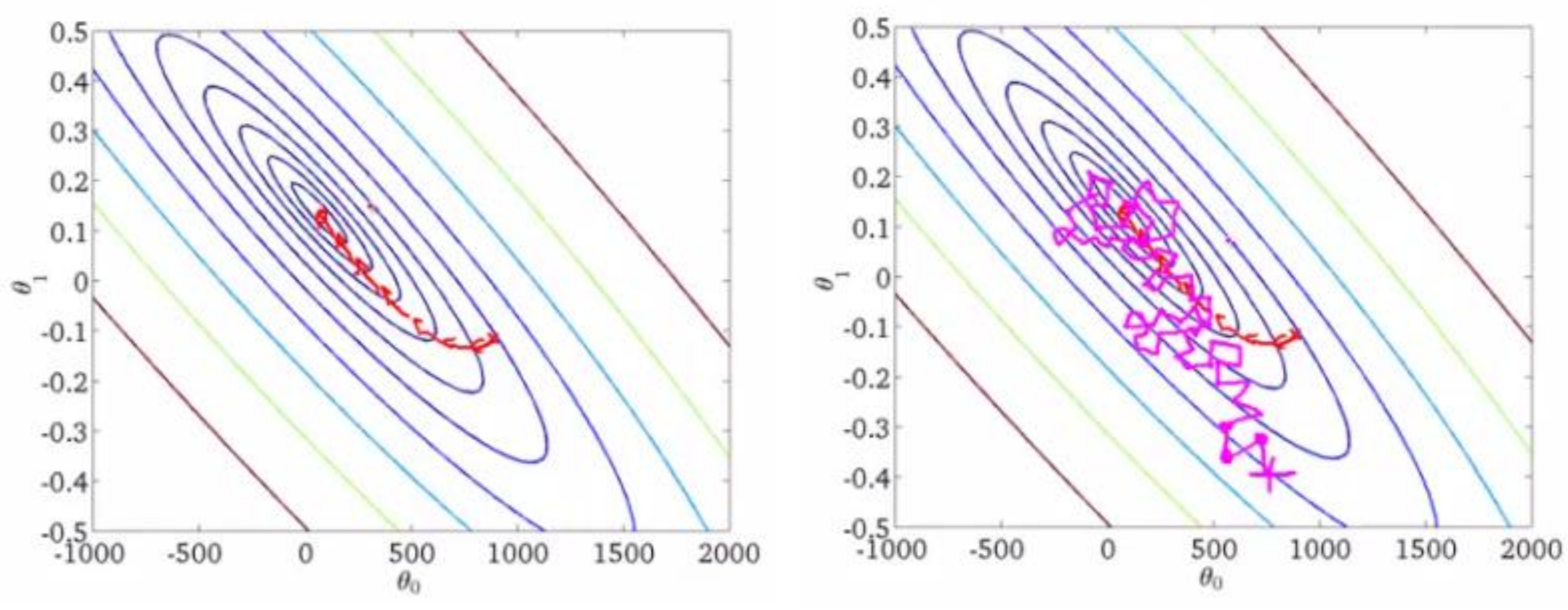
- Scalability
  - Divide-and-conquer (Mackey et al, NIPS11)
  - Stochastic gradient descent
    - GROUSE/GRASTA by Eriksson, Balzano, Recht. SGD on Grassmanian.
  - Parallel Stochastic Gradient
    - Jellyfish by Recht and Re
- The later two need fixed rank hence are non-convex algorithms.

This paper's contributions

- SGD algorithm for convex nuclear norm minimization.
  - Provide rate of convergence
  - More efficient variations.
- A very clear discussion of
  - theory and practice;
  - nuances of different low-rank promoting algorithms.

SGD at a glance

- Batch gradient descent vs. Stochastic Gradient Descent



SGD at a glance

- In expectation, SGD is converging to optimal solution.
- It takes many more update steps, but each step is much cheaper than batch methods.
- Subgradient descent the extension of gradient descent to non-differentiable functions.
  - Usually it requires only one subgradient in the set.

Master theorem for SSGD

- Solve: $$\min_{X \in \mathcal{K}} F(X)\,.$$
- By iterates: $$X^{(t+1)} = \Pi_{\mathcal{K}}(X^{(t)} - \eta^{(t)} g^{(t)})$$

Theorem 2.2 (Convergence of Stochastic Subgradient Descent). *Apply T iterations of the update $X^{(t+1)} = \Pi_{\mathcal{K}}(X^{(t)} - \eta^{(t)} g^{(t)})$ where $g^{(t)}$ is an unbiased estimator of a subgradient of F at $X^{(t)}$ (that is, $\mathbb{E}[g^{(t)}|X^{(t)}] \in \partial F(X^{(t)})$) satisfying $\mathbb{E}[\|g^{(t)}\|_{\mathrm{F}}^2|X^{(t)}] \leq G^2$. Then*

$$\frac{1}{T}\sum_{t=1}^{T} \mathbb{E}[F(X^{(t)})] - F(X_{\mathrm{opt}}) \leq \frac{\|X_{\mathrm{opt}} - X^{(0)}\|_{\mathrm{F}}^2 + \sum_{t=1}^{T}(\eta^{(t)})^2 G^2}{2\sum_{t=1}^{T}\eta^{(t)}}$$

Master theorem for SSGD

Corollary 2.3. *Set* $\eta^{(t)} = \beta \frac{\|X_{\mathrm{opt}}\|_{\mathrm{F}}}{G\sqrt{T}}$ *where* $\beta > 0$, *then*

$$\begin{aligned}\mathbb{E}[F(X^{(\ell)})] - F(X_{\mathrm{opt}}) &\leq \frac{1}{T}\sum_{t=1}^{T}\mathbb{E}[F(X^{(t)})] - F(X_{\mathrm{opt}}) \\ &\leq 4\frac{G\|X_{\mathrm{opt}}\|_{\mathrm{F}}}{\sqrt{T}}\max\left\{\beta, \frac{1}{\beta}\right\}.\end{aligned}$$

Thus, the above corollary implies that the output iterate is $O(\frac{1}{\sqrt{T}})$ close to the optimum solution in expected F-value.

Bounded solution space

- Additional constraint

$$\mathcal{K} = \{X \in \mathbb{R}^{m \times n} : \|X\|_{\mathrm{F}} \leq \Delta\}$$

Definition 2 (Projection Operator for $\mathcal{K}$). *Define* $\Pi_{\mathcal{K}}(P) = \operatorname{argmin}_{Q \in \mathcal{K}} \|P - Q\|_{\mathrm{F}} = \min\{1, \frac{\Delta}{\|P\|_{\mathrm{F}}}\}P.$

- It doesn't change the optimization because we know optimal solution $||X^*||_F < \Delta$.

Compute subgradient

- SVD of X

$$X = U\Sigma V^{\top}$$

- Subgradient of nuclear norm

$$U_{1:m,1:r}V_{1:n,1:r}^{\top} \in \partial\|X\|_*$$

- Subgradient of objective function

$$\mathcal{G}(F(X)) \stackrel{\text{def}}{=} \nabla f(X) + \lambda \cdot U_{\text{rank}}V_{\text{rank}}^{\top} \in \partial_X F(X)$$

SSGD for Nuclear norm

- What is left is to provide an efficient unbiased estimator.
- Probing Matrix Y: n*k. E(YY')=Identity.
  - We use $\mathcal{G}(F(X))YY^\top$ as an unbiased estimator of $\mathcal{G}(F(X))$
- When Y is scaled identity matrix, computation of $\mathcal{G}(F(X))Y$ is more efficient.

Probing matrix

- It can be anything that satisfies
  - $Y \in R^{n \times k}$
  - E(YY')=Identity
- Example:
  - $n = 3, k = 2$
  - $Y = \begin{pmatrix} \sqrt{\frac{3}{2}} & 0 \\ 0 & 0 \\ 0 & \sqrt{\frac{3}{2}} \end{pmatrix}, \ YY^T = \begin{pmatrix} \frac{3}{2} & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \frac{3}{2} \end{pmatrix}$
  - $E(YY^T) = \frac{1}{3}\begin{pmatrix} \frac{3}{2} & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \frac{3}{2} \end{pmatrix} + \frac{1}{3}\begin{pmatrix} \frac{3}{2} & 0 & 0 \\ 0 & \frac{3}{2} & 0 \\ 0 & 0 & 0 \end{pmatrix} + \frac{1}{3}\begin{pmatrix} \frac{3}{2} & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \frac{3}{2} \end{pmatrix} = I_3$

Basic-SSGD

- Here comes the algorithm.

Algorithm 1 Basic-SSGD

Input: f, λ, T, step sizes $\eta^{(1)}, \ldots, \eta^{(T-1)}$, and k

Initialize $X^{(0)} = \mathbf{0}_{m \times n}$
for $t = 0$ **to** $T-1$ **do**
  Generate an $n \times k$ probing matrix Y
  $g^{(t)} \longleftarrow \mathcal{G}(F(X^{(t)}))YY^{\top}$
  $X^{(t+1)} \longleftarrow \Pi_{\mathcal{K}}(X^{(t)} - \eta^{(t)} g^{(t)})$
end for
Return $X^{(\ell)} = \text{argmin}_{X^{(t)}, 0 \le t \le T} F(X^{(t)})$

- Note it requires one SVD for each iteration!

Fast-SSGD

- A Fast-SSGD update for sum loss function and low rank X using QR factorization of SVD.

Algorithm 2 FAST-SSGD-UPDATE

Input: $U \in \mathbb{R}^{m \times r^{(t)}}$, $\Sigma \in \mathbb{R}^{r^{(t)} \times r^{(t)}}$, $V \in \mathbb{R}^{n \times r^{(t)}}$, $Y \in \mathbb{R}^{n \times k}$, and $\eta^{(t)} > 0$

1. $S^{(t)} \longleftarrow \mathcal{G}(F(X))Y$ {without forming $\mathcal{G}(F(X))$}
2. $\widehat{U}^{(t+1)} \longleftarrow [U^{(t)}\Sigma^{(t)} \quad S^{(t)}]$
3. $\widehat{V}^{(t+1)} \longleftarrow [V^{(t)} \quad -\eta^{(t)}Y]$
4. Factorize: $\widehat{U}^{(t+1)} = Q_U R_U$
5. Factorize: $\widehat{V}^{(t+1)} = Q_V R_V$
6. $T \longleftarrow R_U R_V^\top$
7. SVD computation: $T = M\bar{\Sigma}^{(t+1)}N^\top$
8. $\bar{U}^{(t+1)} \longleftarrow Q_U M$
9. $\bar{V}^{(t+1)} \longleftarrow Q_V N$
10. Return $\bar{U}^{(t+1)}$, $\bar{\Sigma}^{(t+1)}$, and $\bar{V}^{(t+1)}$

Guarantee for SSGD

- Theorem 3.3

$$\mathbb{E}[X^{(l)}] - F(X_{\text{opt}}) \leq 4\sqrt{n}\frac{(G + \lambda\sqrt{r})\Delta}{\sqrt{kT}} \max\left\{\beta, \frac{1}{\beta}\right\}.$$

  - Rate of convergence at $O\left(\frac{1}{\sqrt{k}}\right)$
  - Need $O\left(\frac{n}{k\epsilon^2}\right)$ to converge to error ϵ
- Complexity of each iteration
  - Basic-SSGD: $O(mn^2)$
  - Fast-SSGD: $O(m(r^{(t)} + k)^2)$
  - $r^{(t)}$ increases as t becomes large…

Restrict r for fast computation

- Solution space becomes non-convex!
  - In theory, it's NP-hard to compute.
  - But in practice, it works great.
  - Equivalent shown in http://arxiv.org/abs/1203.1570
- Empirical evidence
  - nuclear norm regularization still useful
  - even though explicit rank constraint is imposed

Regularized matrix factorization:

Min $||X-UV'|| + ||U||^2_F + ||V||^2_F$

In fact: Min $||U||^2_F + ||V||^2_F = ||UV'||_F$

Experiments

- Netflix data: 480k user, 18k movie, 10M movie ratings.
- Movielens data: 70k user, 10k movie, 100 million movie ratings.
- Results:
  - It gives faster and better results w.r.t. convex methods such as Soft-impute.
  - It gives slower and worse results w.r.t. non-convex factorization methods, e.g., Jellyfish.

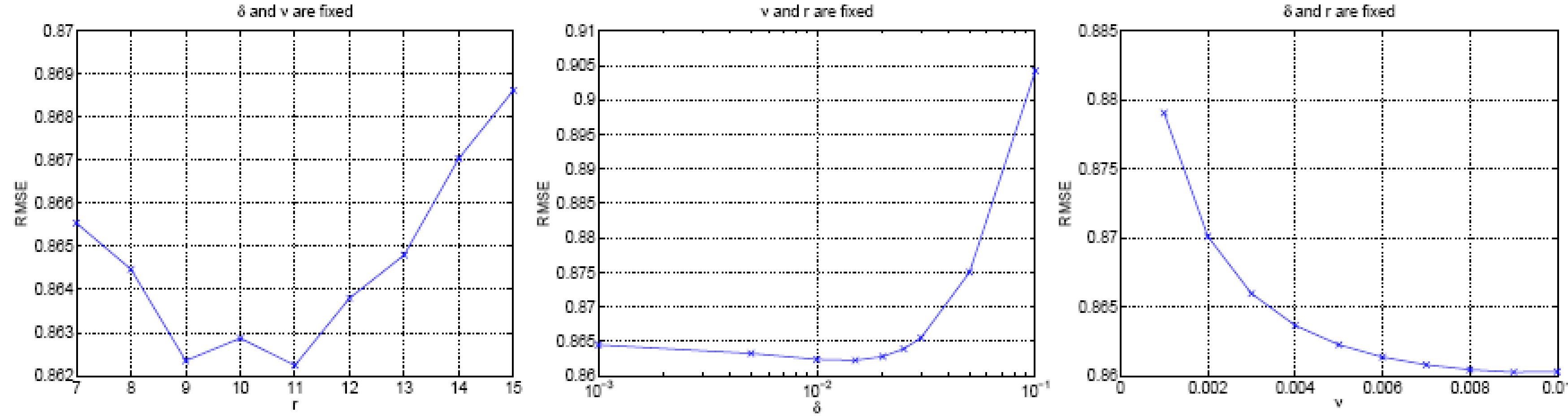
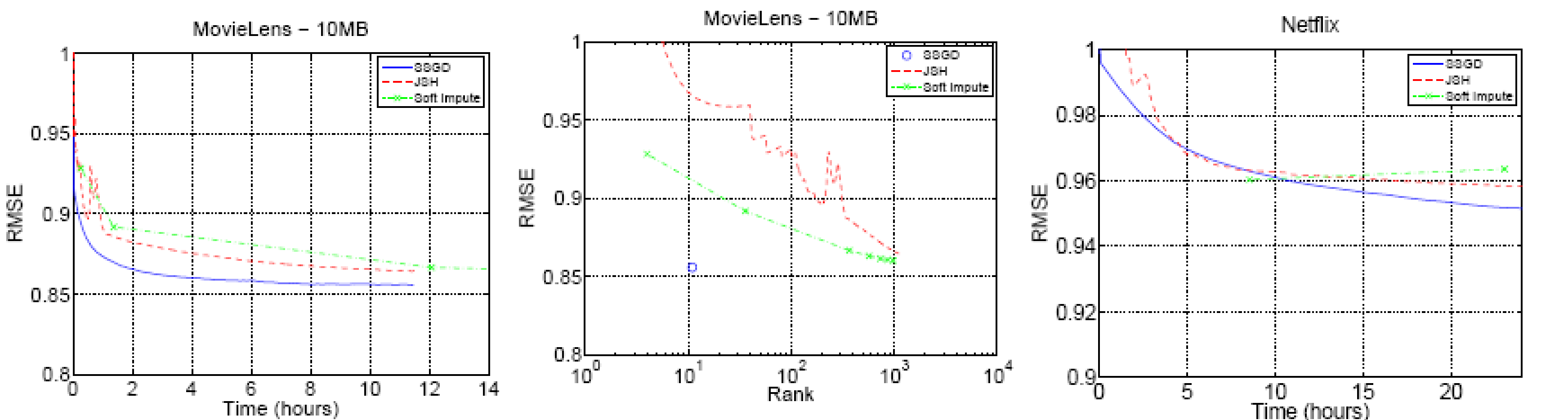


Figure 1. Sensitivity of SSGD to parameters on the MovieLens 10M dataset. We run SSGD for 45 super-iterations. In each of the three graphs, we fix two parameters and vary the third.



(a) (b) (c)

Figure 2. Figure (a): test RMSE vs time on MovieLens 10M. Figure (b): test RMSE vs rank MovieLens 10M (for SSGD-MATRIX-COMPLETION, rank $r = 11$; figure 1 shows its RMSE vs rank). Figure (c): shows test RMSE vs time on the Netflix dataset.

Discussion

- Convex relaxation is tractable, but less desirable under noise.
- Explicit rank will help in practice, especially when the physical rank of the data is known.
- It's good to add nuclear norm regularization even if the rank constraint is already imposed.

Questions from class

- Jiaming: In algorithm I, why does it still need to select X(l) from 0 <= l <= T? Is the algorithm convergent?
  - That's to keep track of the solution with best objective value thus far. Convergence is not a problem.

- Jiaming: What does it mean "For both JSH and Soft-Impute, we needed to go to a much larger rank to obtain a RMSE comparable to that"?
  - It explains the middle of Figure 2.

Questions from class

- Shahzor: In pdf file.

Thank you!