Supplementary of SVD-free Convex-Concave Approaches for Nuclear Norm Regularization

Yichi Xiao[1], **Zhe Li**[2], **Tianbao Yang**[2], **Lijun Zhang**[1]
[1]National Key Laboratory for Novel Software Technology, Nanjing University, Nanjing 210023, China
{xiaoyc, zhanglj}@lamda.nju.edu.cn
[2]Department of Computer Science, the University of Iowa, Iowa City, IA 52242, USA
{zhe-li-1, tianbao-yang}@uiowa.edu

1 Proof of Theorem 2

In this section, we provide the detail proof of the Theorem 2.

Denote $v_1, v_2, \cdots, v_T$ be the sequence of stochastic subgradient $\partial f(A_t, \xi_t)$. For short, let $v_{1:T}$ denote this sequence $v_1, v_2, \cdots, v_T$. Let $L(A, U) = \mathbf{E}_\xi[f(A;\xi)] + \lambda tr(U^\top A) - \rho[\|U\|_2 - 1]_+$.Note that A_{t+1} is the update of stochastic subgradient descent applied to $L(A_t, U_t)$. By the law of total expectation and convexity of $L(A, U)$ w.r.t. A, we have

$$\begin{aligned}&\mathbf{E}_{v_{1:T}}[L(A_t, U_t) - L(A, U_t)]\\ \leq&\mathbf{E}_{v_{1:T}}[\langle A_t - A, \partial f(A_t;\xi_t) + \lambda U_t\rangle]\end{aligned}$$

By the updating rule of SECONE-S, we know

$$\begin{aligned}&\langle A_t - A, \partial f(A_t;\xi_t) + \lambda U_t\rangle \leq \frac{\eta_t}{2}\|\partial f(A_t;\xi_t) + \lambda U_t\|_F^2\\ &+ \frac{1}{2\eta_t}(\|A - A_t\|_F^2 - \|A - A_{t+1}\|_F^2)\end{aligned}$$

Then taking the expectation on the above inequality and combining the above two inequalities, we have

$$\begin{aligned}&\mathbf{E}_{v_{1:T}}[L(A_t, U_t) - L(A, U_t)]\\ \leq&\mathbf{E}_{v_{1:T}}[\frac{\eta_t}{2}\|\partial f(A_t;\xi_t) + \lambda U_t\|_F^2]\\ +&\mathbf{E}_{v_{1:T}}[\frac{1}{2\eta_t}(\|A - A_t\|_F^2 - \|A - A_{t+1}\|_F^2)]\end{aligned}$$

By Jensen's inequality, we have

$$\mathbf{E}_{v_{1:T}}[\|\partial f(A_t;\xi_t)\|_F] \leq \sqrt{\mathbf{E}_{v_{1:T}}[\|\partial f(A_t;\xi_t)\|_F^2]} \leq G$$

Thus,

$$\begin{aligned}&\mathbf{E}_{v_{1:T}}[\|\partial f(A_t;\xi_t) + \lambda U_t\|_F^2]\\ =&\mathbf{E}_{v_{1:T}}[\|\partial f(A_t;\xi_t)\|_F^2] + 2\langle \lambda U_t, \mathbf{E}_{v_{1:T}}[\|\partial f(A_t;\xi_t)\|_F]\rangle\\ +&\|\lambda U_t\|_F^2\\ \leq&(G + \lambda\sigma)^2\end{aligned}$$

Similarly, U_{t+1} is the update of subgradient descent applied to $L(A_t, U_t)$, hence for any $U \in \mathbb{R}^{n\times m}$

$$\begin{aligned}L(A_t, U) - L(A_t, U_t) \leq& \frac{1}{2\tau_t}(\|U - U_t\|_F^2 - \|U - U_{t+1}\|_F^2)\\ &+ \frac{\tau_t}{2}\|\lambda A_t - \rho\partial[\|U_t\|_2 - 1]_+\|_F^2\end{aligned}$$

Combining the above two inequalities, we obtain an inequality of the gap

$$\begin{aligned}&\mathbf{E}_{v_{1:T}}[L(A_t, U) - L(A, U_t)]\\ \leq&\frac{1}{2\eta_t}\mathbf{E}_{v_{1:T}}[\|A - A_t\|_F^2 - \|A - A_{t+1}\|_F^2]\\ +&\frac{1}{2\tau_t}\mathbf{E}_{v_{1:T}}[\|U - U_t\|_F^2 - \|U - U_{t+1}\|_F^2]\\ +&\frac{\eta_t}{2}(G + \lambda\sigma)^2 + \frac{\tau_t}{2}(\rho + \lambda\sigma)^2\end{aligned}$$

Due to the linearity of expectation, we shall adopt the same procedure as in the proof of Theorem 1 to handle the diminishing step size $\eta_t = c_1/\sqrt{t}$ and $\tau_t = c_2/\sqrt{t}$. By summing up the resulting inequality over $t = 1, \cdots, T$, we have

$$\begin{aligned}&\mathbf{E}_{v_{1:T}}[\sum_{t=1}^T L(A_t, U) - L(A, U_t)] \leq \frac{\sqrt{T}}{2c_1}D_1^2 + \frac{\sqrt{T}}{2c_2}\mathbf{E}_{v_{1:T}}[D_2^2]\\ &+ c_1\sqrt{T}(G + \lambda\sigma)^2 + c_2\sqrt{T}(\rho + \lambda\sigma)^2\end{aligned}$$

The remaining proof is similar to the part II of Theorem 1 and we could conclude the proof by following that.

2 Experiments

We present more numerical experiments on real datasets to demonstrate the efficiency of the proposed algorithms.

2.1 Robust Low-rank Matrix Approximation

We compare our method with two classical methods: subgradient descent (GD) and proximal subgradient descent (PGD) [Duchi and Singer, 2009] on the Gisette[1] dataset, which contains $n = 6000$ instances, each of which has $m = 5000$ features. According to Theorem 1, we set step sizes in Algorithm 1 as $\eta_t = c_1/\sqrt{t}$ and $\tau_t = c_2/\sqrt{t}$, where c_1, c_2 are some constants. The same step size $\eta_t = c_1/\sqrt{t}$ is also used for GD and PGD. We tune the value of c_1 and c_2 in a range of $\{10^{-5}, 10^{-4}, \ldots, 10^{10}\}$ and report the best results based on the objective value.

In Fig. 1, we plot the objective value versus the running time for $\lambda = 5 \times 10^{-6}$. We choose this value of λ because it can produce a low-rank output, and the convergence behavior

[1]https://archive.ics.uci.edu/ml/datasets/Gisette

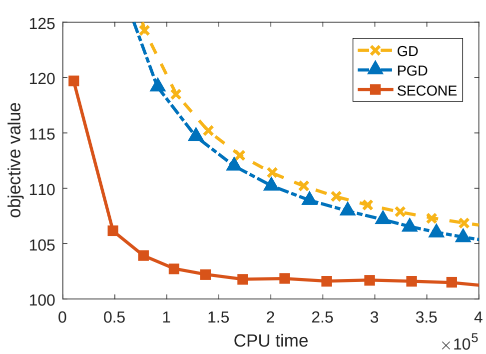

Figure 1: Results of robust low-rank matrix approximation

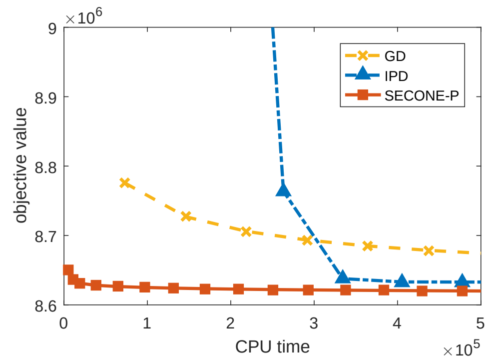

Figure 2: Results of sparse and low-rank link prediction

Table 1: Statistics for matrix approximation

Method	c_1	c_2	T	Total CPU time
SECONE	$1e9$	10	36000	$4.05e5$
PGD	$1e9$		500	$4.10e5$
GD	$1e9$		500	$3.93e5$

Table 2: Statistics for link prediction

Method	c_1 or θ	c_2	T	Total CPU time
SECONE	1	$1e{-}5$	15500	$5.02e5$
IPD	0.01		450	$5.13e5$
GD	1		420	$5.10e5$

is insensitive to λ. As can be seen, SECONE decreases much faster than GD and PGD. This is as expected as SECONE is SVD-free and time-efficient, which is also convinced by the statistics shown in Table 1. As can be seen, each iteration of SECONE takes much less time than other two methods.

2.2 Sparse and Low-rank Link Prediction

Following the setting in [Richard *et al.*, 2012], we perform experiments on the Facebook100 dataset which contains the friendship relations between students. We select a single university with $41,554$ students and keep only the 10% users with the highest degree (e.g. $m = n = 4155$). We flip 15% of randomly chosen entries and the goal is to learn a sparse and low-rank matrix from the noisy adjacency matrix Y.

We compare Algorithm 3 (SECONE-P) with subgradient descent (GD) and Incremental Proximal Decent (IPD), which is an iterative algorithm designed for the above problem but with no theoretical guarantees [Richard *et al.*, 2012]. The step sizes in SECONE-P and GD are set in the same way as in Section 2.1. The parameter θ of IPD is searched in the range of $\{10^{-3}, 10^{-2}, \ldots, 10\}$.

In Fig. 2, we plot objective value versus the running time when $\lambda = 8$ and $\gamma = 0.4$. As can be seen, SECONE-P converges much faster than other methods, and GD performs the worst. The statistics of different methods are shown in Table 2. Again, the running time per iteration of SECONE-P is much smaller than other methods.

References

[Baccini *et al.*, 1996] A. Baccini, Ph. Besse, and A. de Falguerolles. A l_1-norm PCA and a heuristic approach. In *Proceedings of the International Conference on Ordinal and Symbolic Data Analysis*, pages 359–368, 1996.

[Croux and Filzmoser, 1998] Christophe Croux and Peter Filzmoser. Robust factorization of a data matrix. In *Proceedings in Computational Statistics*, pages 245–250, 1998.

[Duchi and Singer, 2009] John Duchi and Yoram Singer. Efficient online and batch learning using forward backward splitting. *Journal of Machine Learning Research*, 10(Dec):2899–2934, 2009.

[Ke and Kanade, 2005] Qifa Ke and Takeo Kanade. Robust l_1 norm factorization in the presence of outliers and missing data by alternative convex programming. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 1, pages 739–746, 2005.

[Richard *et al.*, 2012] Emile Richard, Pierre-Andre Savalle, and Nicolas Vayatis. Estimation of simultaneously sparse and low rank matrices. In *Proceedings of the 29th International Conference on Machine Learning*, pages 1351–1358, 2012.