Robust Matrix Completion*

Rongting Zhang, Zhao Liu

Dec 5, 2016

The problem of low-rank matrix completion has received significant interest in the past decade. It can be used as a building block for recommender systems especially for collaborative filtering, in reconstructing 3D path of particles from partial observation or image inpainting.

Low-rank matrix completion consists of recovering a rank-r matrix of size $m \times n$ from only a fraction of its entries. Denoting by Ω the set of observed entries, the problem can be stated as

$$\begin{aligned} \min_X \quad & \operatorname{rank}(X) \\ s.t. \quad & P_\Omega(X) = P_\Omega(M), \end{aligned} \tag{1}$$

where P_Ω the orthogonal projection onto to the space of non-zero entries. The problem, however, is known to NP-hard. A relaxation problem is proposed as an approximation to (1)

$$\begin{aligned} \min_X \quad & \|X\|_* \\ s.t. \quad & P_\Omega(X) = P_\Omega(M), \end{aligned} \tag{2}$$

which has been shown in various situations would result in exact recovery with certain restrictions. However, in real world, the data observed are often polluted with noise. Candès et al. [1] studies the setting where a small amount of noise is present in the data. Chen et al. [3] studied the problem of where a number of columns are arbitratily corrupted. Li [4] and Chen et al. [2] studied the problem when a constant fraction of the entries of the matrix are outliers.

In this report, we try to do an incomplete survey and comparison on some of the aforementioned papers.

1 Preliminaries

We briefly derive the following relations between nuclear norm minimization and thresholding on matrices, which we will to use repeatedly:

$$\arg\min_X \epsilon\|X\|_* + \frac{1}{2}\|X - W\|_F^2 = \mathfrak{S}_\epsilon[W] \tag{3}$$

$$\arg\min_X \epsilon\|X\|_1 + \frac{1}{2}\|X - W\|_F^2 = \mathfrak{L}_\epsilon[W] \tag{4}$$

$$\arg\min_X \epsilon\|X\|_{1,2} + \frac{1}{2}\|X - W\|_F^2 = \mathfrak{C}_\epsilon[W] \tag{5}$$

where:

*This is course project for 2016 Fall UT-Austin graduate course EE381K Convex Optimization

- $\mathfrak{L}_\epsilon[\cdot]$ is the soft thresholding operator:

$$\mathfrak{L}_\epsilon[A]_{i,j} = \begin{cases} A_{i,j} - \epsilon & \text{if } A_{i,j} > \epsilon \\ A_{i,j} + \epsilon & \text{if } A_{i,j} < -\epsilon \\ 0 & \text{otherwise} \end{cases} \tag{6}$$

- $\mathfrak{S}_\epsilon[\cdot]$ is the soft-thresholding operator on Sigular Values: $\mathfrak{S}_\epsilon[W] = U\mathfrak{S}_\epsilon[\Sigma]V^\intercal$ where $W = U\Sigma V^\intercal$
- $\mathfrak{C}_\epsilon[\cdot]$ is the column-wise (ℓ_2) soft thresholding operator: (denote A_j to be the $j-th$ column of A)

$$\mathfrak{C}_\epsilon[A]_j = \begin{cases} A_j - \epsilon A_j/\|A_j\|_2 & \text{if } \|A_j\|_2 > \epsilon \\ 0 & \text{otherwise} \end{cases} \tag{7}$$

- $\|X\|_{1,2} = \sum_i (\sum_j X_{i,j}^2)^{\frac{1}{2}}$.

We briefly sketch a proof using proximal map and Moreau's identity:

Proof. (1) Denote $g(X) = \|X\|_*$. By Moreau's identity,

$$\arg\min_X \epsilon\|X\|_* + \frac{1}{2}\|X - W\|_F^2 = prox_{\epsilon g}(W) = W - prox_{(\epsilon g)^*}(W)$$

$$\begin{aligned}(\epsilon g)^*(A) &= \sup_B\{< A, B > -\epsilon\|B\|_*\} \\ &= \epsilon \sup_B\{< \frac{1}{\epsilon}A, B > -\|B\|_*\} \\ &= \epsilon g^*(A) \\ &= \epsilon I_{\{\|A\|_2 \le 1\}}\end{aligned}$$

where the last equality follows from that the dual norm of nuclear norm $\|\cdot\|_*$ is the spectual norm $\|\cdot\|_2$ (proved in HW4), and I is the indecator function which takes 0 inside the region $\{A : \|A\|_2 \le 1\}$ and ∞ outside. Followed by the above result, the *prox* of the conjugate is

$$\begin{aligned}prox_{(\epsilon g)^*}(W) &= \arg\min_{\{A:\|A\|_2 \le 1\}} \epsilon + \frac{1}{2}\|W - A\|_F^2 \\ &= proj_{\{A:\|A\|_2 \le 1\}}(W) \\ &= U\Sigma_\epsilon V^T \\ \text{where:} & \\ U\Sigma V^T \quad &\text{is} \quad \text{the SVD of } W, \\ (\Sigma_\epsilon)_{i,i} &= \min(|\sigma_i|, \epsilon) sign(\sigma_i)\end{aligned}$$

Applying Moreau's identity, the conclusion (3) directly follows.

(2)(3) The proof is similar to (1), applying that the dual norm of $\|\cdot\|_1$ is $\|\cdot\|_\infty$, and the dual norm of $\|\cdot\|_{1,2}$ is $\|\cdot\|_{\infty,2}$. □

2 Matrix completion with Small Noise

It is proposed in Candès et. al [1], that matrix completion problem with small noise especially in the setting of Gaussian noise can be solved stably using

$$\begin{aligned} \min_X \quad & \|X\|_* \\ s.t. \quad & \|\mathcal{A}(X) - b\|_2 \le \theta, \end{aligned} \tag{8}$$

or its Lagrangian version

$$\min_X \quad \|X\|_* + \lambda \|\mathcal{A}(X) - b\|_2^2, \tag{9}$$

where θ and λ are parameters. Under matrix completion setting, $\mathcal{A} = \mathcal{A}^* = P_\Omega$, $b = M$ and $\|\cdot\|_2$ on vectors (expanded from matrices) is just the Frobenius norm $\|\cdot\|_F$.

To solve for (9), a **Fixed Point Continuation** method is used [5]. The fixed point continuation method essentially is based on solving a series of problems in the form of (9) with an increasing sequence of $\{\lambda_i\}_{i=1}^L$ with $\lambda_L = \lambda$. Each of the problem is solved using fixed point iteration

$$Y^k = X^k - \tau g(X^k) \tag{10}$$

$$X^{k+1} = \mathfrak{S}_{\frac{\tau}{2\lambda}}[Y^k]. \tag{11}$$

where $g(X) = \mathcal{A}^*(\mathcal{A}(X) - b)$, $\mathcal{A}^*$ is the dual operator of the linear operator $\mathcal{A}$.

To derive the fixed point iteration, we note that X^* is the optimal solution to (9) if and only if

$$\mathbf{0} \in \frac{1}{2\lambda} \partial \|X^*\|_* + g(X^*), \tag{12}$$

This is equivalent to

$$\mathbf{0} \in \frac{\tau}{2\lambda} \partial \|X^*\|_* + X^* - (X^* - \tau g(X^*)), \tag{13}$$

for any $\tau > 0$. If we let

$$Y^* = X^* - \tau g(X^*), \tag{14}$$

then we have X^* is the optimal solution to

$$\min_X \quad \tau \|X\|_* + \lambda \|X - Y\|_F^2, \tag{15}$$

with $Y = Y^*$. And it is known that (15) has a closed form solution which uses thresholding on singular values: $\mathfrak{S}_{\frac{\tau}{2\lambda}}[Y]$ (see (3)). A fixed point continuation algorithm is shown in Algorithm 1. The stopping condition for the inner loop in Algorithm 1 is such that

$$\frac{\|X^{k+1} - X^k\|_F}{\max\{1, \|X^k\|_F\}} < xtol, \tag{16}$$

where $xtol$ is a small positive number. We also put a restriction on number of iterations for the inner loop, the maximum iteration for inner loop is denoted as I_m.

Algorithm 1 Fixed Point Continuation(FPC)

Input: $X_0, \widehat{\lambda} > 0$.
  Initialize: Select $\lambda_1 < \lambda_2 < \ldots < \lambda_L = \widehat{\lambda}$, set $X = X_0$
  for $\lambda = \lambda_1, \ldots, \lambda_L$ **do**
    while not converged **do**
      select $\tau > 0$
      compute $Y = X - \tau\mathcal{A}^*(\mathcal{A}(X) - b)$
      compute $X = \mathfrak{L}_{\frac{\tau}{2\lambda}}(Y)$
    end while
  end for
Output: X

3 Matrix completion with corrupted columns

3.1 Problem and notations

Consider the case when some entries of the matrix is corrupted. Assume the matrix $M \in \mathbb{R}^{m\times n}$ have a subset of corrupted columns $\mathcal{I}_0$ ($|\mathcal{I}_0| = \gamma n$, $\gamma < 1$). Namely, there are no restrictions on values of these column entries. The position of $\mathcal{I}_0$ is unknown beforehand. And assume M is 'esentially' low rank, i.e. suppose M can be decomposed as

$$M = L_0 + C_0 \tag{17}$$

where:

- L_0 is a rank r matrix, and is nonzero on at most $(1-\gamma)n$ columns ($\mathcal{I}_0^c$). Further, assume that L_0 satisfies **incoherence conditions**: suppose its SVD $U_0\Sigma_0V_0^\intercal$, there exists constant $\mu_0 > 0$ s.t.

$$\begin{aligned} \max_i \|U_0^\intercal e_i\|^2 &\le \mu_0\frac{r}{m} \\ \min_j \|V_0^\intercal e_j\|^2 &\le \mu_0\frac{r}{(1-\gamma)n} \end{aligned}$$

  which means that both the left and right singular vectors are 'spread out', so the information about row/column space of L_0 are not very sparse among row/column space.

- C_0 is nonzero only on the γn columns ($\mathcal{I}_0$), which correspond to the corrupted entries of M.

Let $\Omega \subset [m]\times[n]$ denote the subset of samples taken from the matrix, which can also be decomposed of

$$\Omega = \Omega^{(L)} \cup \Omega^{(C)}$$

where:

- $\Omega^{(L)}$ is sampled uniformly from the noncorrupted columns ($[m]\times\mathcal{I}_0^c$), with $|\Omega^{(L)}| = N$;
- there are no restrictions on how $\Omega^{(C)}$ is sampled on corrputed columns, since we do not aim at (and there's no hope of) recovering C_0.

Denote $\mathcal{P}_\Omega$ the orthogonal projection onto the subspace of matrices supported on Ω:

$$\mathcal{P}_\Omega(X)_{i,j} = \begin{cases} X_{i,j} & \text{if } (i,j)\in\Omega \\ 0 & \text{otherwise} \end{cases}$$

3.2 The Augmented Lagrange Multiplier Method

Given $\mathcal{P}_\Omega(M)$, the **goal** is to recover matrices (L, C) such that:

1. The location of corrupted columns $\mathcal{I}_0$ is recovered, i.e. nonzero columns of $C = \mathcal{I}_0$;
2. L=L_0 on $\mathcal{I}_0^c$ columns;
3. L is in the column space of L_0, i.e. $\mathcal{P}_{U_0}(L) = L$ ($\mathcal{P}_{U_0}$is the orthogonal projection onto the space spanned by columns of U_0); combined with 2nd requirement, it means that the column space of L_0 is completely recovered.

The paper [3] proposed the following **optimization problem**:

$$\begin{aligned} \min_{L,C} \quad & \|L\|_* + \lambda \|C\|_{1,2} \\ s.t. \quad & \mathcal{P}_\Omega(L + C) = \mathcal{P}_\Omega(M) \end{aligned} \tag{18}$$

which can be formulated as

$$\begin{aligned} \min_{L,C,E} \quad & \|L\|_* + \lambda \|C\|_{1,2} \\ s.t. \quad & L + C + E = M \\ \text{and} \quad & \mathcal{P}_\Omega(E) = 0 \end{aligned} \tag{19}$$

3.3 The Augmented Lagrangian Multiplier Methods

The algorithm applies the Augmented Lagrangian Multiplier (ALM) to the minimization problem (19). The general idea is to solve the dual problem in a Dual Gradient Ascent way, by incorprating the constraint into the Augmented Lagrangian:

$$\mathcal{L}_u(L, C, E; Y) = \|L\|_* + \lambda \|C\|_{1,2} + < Y, M - L - C - E > + \frac{u}{2} \|M - L - C - E\|_F^2$$

and optimize L, C and E in an alternating way:

$$\begin{aligned} L^{(k+1)} &= \arg\min_L \mathcal{L}_u(L, C^{(k)}, E^{(k)}; Y^{(k)}) \\ C^{(k+1)} &= \arg\min_C \mathcal{L}_u(L^{(k+1)}, C, E^{(k)}; Y^{(k)}) \\ E^{(k+1)} &= \arg\min_{\mathcal{P}_\Omega(E)=0} \mathcal{L}_u(L^{(k+1)}, C^{(k+1)}, E; Y^{(k)}) \\ Y^{(k+1)} &= Y^{(k)} + \eta_k (M - L^{(k+1)} - C^{(k+1)} - E^{(k+1)}) \end{aligned}$$

If we apply the preliminary results (3),(5), completing the squares, the first three equations simplifies to:

$$
\begin{aligned}
L^{(k+1)} &= \arg\min_L \{\|L\|_* + < Y^{(k)}, M - L - C^{(k)} - E^{(k)} > \\
&\quad + \frac{u}{2}\|M - L - C^{(k)} - E^{(k)}\|_F^2\} \\
&= \arg\min_L \{\|L\|_* + \frac{u}{2}\|L - (M - C^{(k)} - E^{(k)} + u_k^{-1}Y^{(k)})\|_F^2\} \\
&= U\mathfrak{L}_{u_k^{-1}}[S]V^\intercal \\
(\text{where} &\quad USV^\intercal = M - C^{(k)} - E^{(k)} + u_k^{-1}Y^{(k)}) \\
C^{(k+1)} &= \mathfrak{E}_{\lambda u_k^{-1}}[M - E^{(k)} - L^{(k)} + u_k^{-1}\lambda Y^{(k)}] \\
E^{(k+1)} &= \mathcal{P}_{\Omega^c}(M - L^{(k+1)} - C^{(k+1)} + u_k^{-1}Y^{(k)})
\end{aligned}
$$

Set $u_{k+1} = \alpha u_k$ where $\alpha > 1$. Set stepsize $\eta_k = u_k$ (so we have decreasing regularization constants in the Lagrangian ($\frac{1}{u_k}$) and increasing stepsizes). We have the ALM algorithm for Robust Matrix Completion.

Algorithm 2 The ALM Algorithm for Robust Matrix Completion with corrupted columns

Input: $\mathcal{P}_\Omega(M) \in \mathbb{R}^{m\times n}$, $\Omega \subseteq [m] \times [n]$, λ. Assuming $\mathcal{P}_\Omega(M) = 0$.

Initialize: $Y^{(0)} = 0; L^{(0)} = 0; C^{(0)} = 0; E^{(0)} = 0; u_0 > 0; \alpha > 1; k = 0.$
while not converged **do**
  $(U, S, V) = \text{svd}(M - C^{(k)} - E^{(k)} + u_k^{-1}Y^{(k)})$;
  $L^{(k+1)} = U\mathfrak{L}_{u_k^{-1}}[S]V^\intercal$;
  $C^{(k+1)} = \mathfrak{E}_{\lambda u_k^{-1}}[M - E^{(k)} - L^{(k)} + u_k^{-1}\lambda Y^{(k)}]$;
  $Y^{(k+1)} = Y^{(k)} + u_k(M - L^{(k+1)} - C^{(k+1)} - E^{(k+1)})$;
  $E^{(k+1)} = \mathcal{P}_{\Omega^c}(M - L^{(k+1)} - C^{(k+1)} + u_k^{-1}Y^{(k)})$;
  $u_{k+1} = \alpha u_k$;
  $k = k + 1$;
end while
return $(L^{(k+1)}, C^{(k+1)})$

4 Experiments

The data we used is the 100×100 matrix M provided in Homework 4. The binary matrix O specifies the subset of samples Ω. We did two experiments to compare the FPC algorithm, ALM algorithm and the Projected Subgradient Method we implemented in Homework 4.

First, we corrupt the matrix in the last γn columns ($\gamma = 0, 0.1, 0.2, 0.3, n = 100$). The way we corrupt each column is to randomly select 10 entries and set them to be 1000 (the maximum absolute value of M is about 10). Since we wish to recover the part of the matrix which is not corrupted (i.e. L_0 in equation (17)), and don't care about how the corrupted entries are recovered, the error we calculated is the Frobenius norm on the non-corrupted columns. For example, when $\gamma = 0.1$, at each iteration we get X (from PGD or FPC) or L (from ALM), we compute $\|(X - L_0)_{100\times(0:90)}\|_F^2$. We also compute the Nuclear Norm on the non-corrupted columns, and expecting it to approach the Nuclear Norm of the low-rank matrix $L0$. The errors and Nuclear Norms are plotted with each iteration.

Next, we both corrupt the matrix on some columns and add noise on the observed entries. Other implementation details are the same, except that the errors are computed against the no-noisy data M instead of the noisy data.

Parameters of each algorithm are set as follows:

- For Projected Subgradient Method, apply two choices of varying stepsize: $\frac{1}{k}$ and $\frac{1}{\sqrt{k}}$ and random initialization.
- For the FPC algorithm, select $\eta = 0.25$, $\lambda_1 = \frac{1}{2\eta\|\mathcal{A}^* b\|_2} = \frac{1}{2\eta\|\mathcal{P}_\Omega(M)\|_F}$, $xtol = 10^{-10}$, $I_m = 200$, $X_0 = \mathbf{0}$.
- For the ALM algorithm, select $\lambda = \frac{1}{2}\sqrt{\frac{m^{\frac{1}{4}}}{n}}$(as in the Convergence Theorem and Corollaries in [3]),$u_0 = 1.0/\|M\|_{1,2}$, $\alpha = 1.1$, and the criterion for convergence is:

$$\|M - C^{(k)} - L^{(k)} - E^{(k)}\|_F^2 / \|M\|_F^2 \leq 10^{-6}$$

In Figure 1, we show the result where there is no noise present in the observed data excluding corruptions. Both FPC and ALM converges considerably faster than projected subgradient methods. FPC does not converge as fast as ALM, even in the case with no corruption, probably due to the choice of a small stopping threshold for different λ_i. ALM is shown to able to reconstruct the corrupted columns faithfully and showed the best performance and lowest nuclear norm on all the test cases. However, its performance decreases as γ increases. For FPC, in some of the cases, the nuclear norm becomes much larger with minor decrease in the Frobenius norm error, this potentially means that the selection of λ in FPC should be adjusted in such case.

In Figure 2, we show the result where there is Gaussian noise present in the observed data. ALM does not perform as well as the previous case, which is understandable as ALM does explicitly inforce exact match on uncorrupted columns. Again, we need to properly adjust the λ in FPC to make it work better when the number of corrupted columns increases.

5 Conclusion

In this report, we compared two different algorithms FPC, which is designed for noisy matrix completion and ALM, which is designed for matrix completion with corrupted columns. As a benchmark, we also included the usual subgradient method. FPC in general works quite well and converges much faster than subgradient under both settings we are testing. ALM, on the other hand, performs even better than

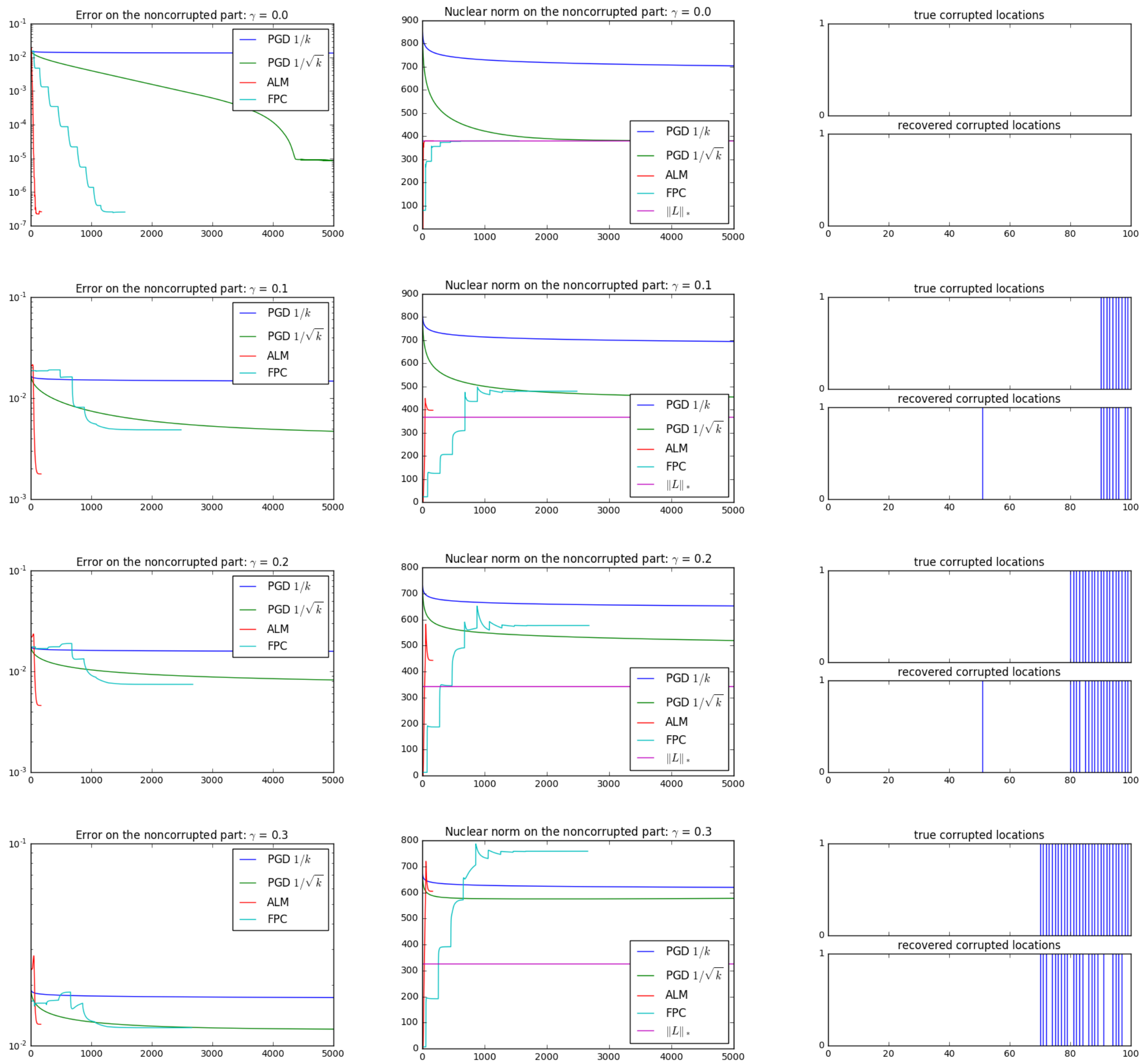

Figure 1: Matrix completion with corrupted columns

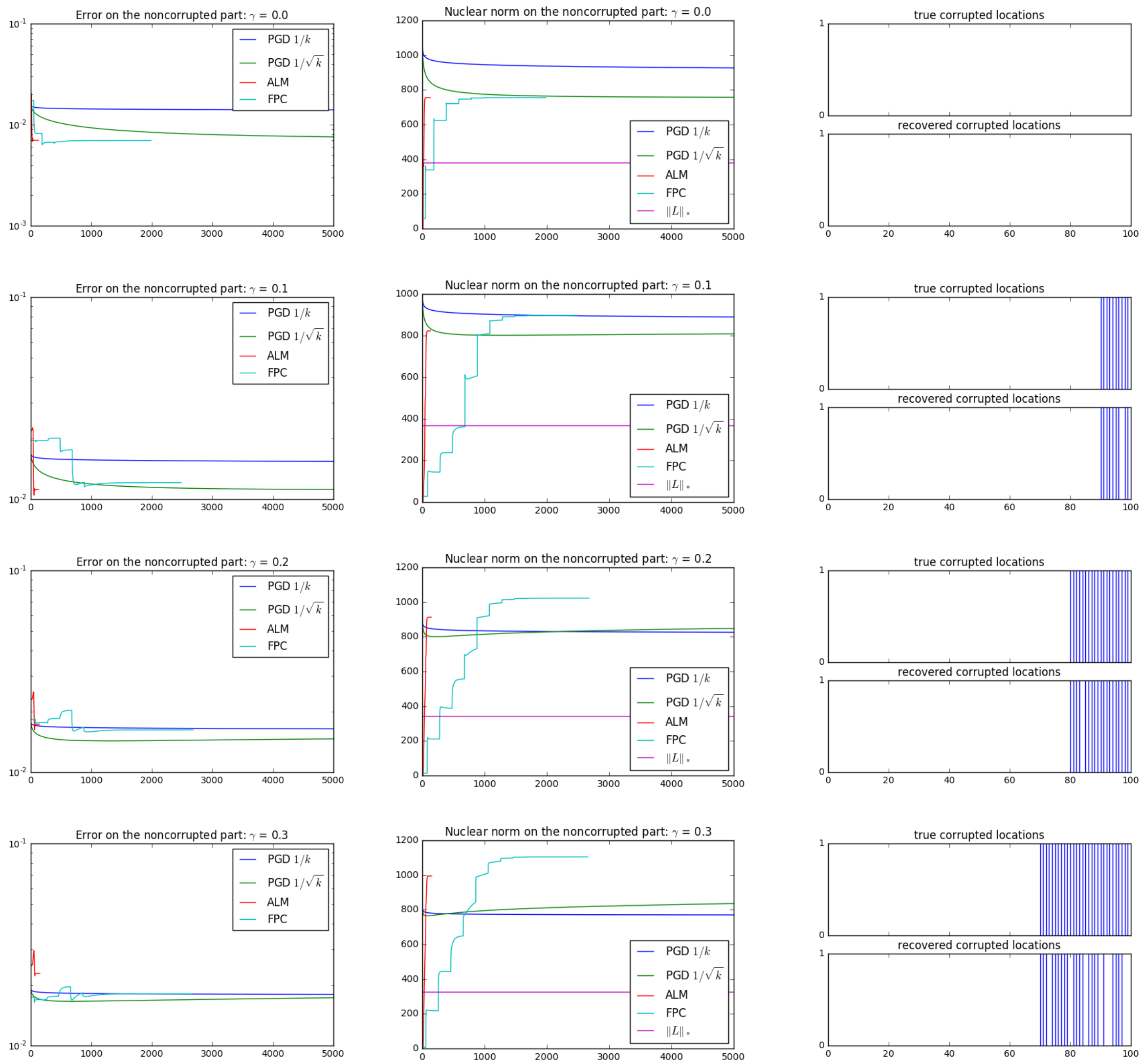

Figure 2: Matrix completion with corrupted column and noisy entries

FPC on data with corrupted columns. However, it is sensitive to noise and does not work quite as well. It would be of interest to see how one can device an algorithm that can generalized to the case with corrupted columns along with noise in observed data.

References

[1] E. J. Candes and Y. Plan, *Matrix completion with noise*, Proceedings of the IEEE, 98 (2010), pp. 925–936.

[2] Y. Chen, A. Jalali, S. Sanghavi, and C. Caramanis, *Low-rank matrix recovery from errors and erasures*, IEEE Transactions on Information Theory, 59 (2013), pp. 4324–4337.

[3] Y. Chen, H. Xu, C. Caramanis, and S. Sanghavi, *Robust matrix completion and corrupted columns*, in Proceedings of the 28th International Conference on Machine Learning (ICML-11), 2011, pp. 873–880.

[4] X. Li, *Compressed sensing and matrix completion with constant proportion of corruptions*, Constructive Approximation, 37 (2013), pp. 73–99.

[5] S. Ma, D. Goldfarb, and L. Chen, *Fixed point and bregman iterative methods for matrix rank minimization*, Mathematical Programming, 128 (2011), pp. 321–353.