Robust Principal Component Analysis

Kirtan H. Modi
SEAS-Ahmedabad University
1401122

***Abstract*—Principal Component Analysis (PCA) is arguably the most widely used statistical tool for data analysis and dimensionality reduction today. However, its brittleness with respect to grossly corrupted observations often puts its validity in jeopardy. Even a single error in data points of matrix M could render the estimated L arbitrarily far from the true L_0. In this period of time where the world has a huge amount of data and there are some data which are corrupted we should use better algorithm to solve this problem. A number of natural approaches to robustifying PCA have been explored and proposed. This paper discuss about the mechanism of Robust Principal Component Analysis which is advanced version of PCA. We will talk about how low rank matrix will be recovered with the help of Convex Optimization program called Principal Component Pursuit (PCP). This convex optimization program is capable enough to recover the low rank matrix and identify the corrupted data.**

Keywords: Principal Component Analysis (PCA), Principal Component Pursuit (PCP), Video surveillance, Face Recognition, Latent Semantic Indexing, Ranking and Collaborative Filtering, Low Rank Matrix, Sparse Matrix, Convex Optimization.

I. Introduction

In today's world lots of works needed to be analysed and we need some kind of program to deal with large amount of Data.In order to better understand the behavior of the complex system, a natural approach is to break the system into simpler components. There are some real life applications where data can be modeled in terms of low-rank matrix and sparse matrix.Sparse structure of noise is common in many applications like: Video Surveillance,Face Recognition,Latent Semantic Indexing,Ranking and Collaborative Filtering.In recent time Robust Principal Component Analysis is widely used because of it's ability to recover low-rank model from the sparse noise.Suppose we have to separate foreground and background from a given image or given video frame.then,at that time this approach can help us to separate both the tings.

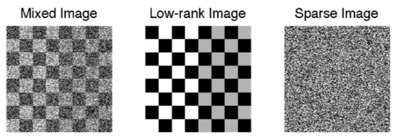

Fig. 1: Image seperation

So, here we are given a data matrix (M) which can be decomposed into sparse matrix S_0 and low-rank matrix L_0

$$M = L_0 + S_0,$$

Here, we do not know the low dimensional column and row space of L_0. Even we don't know the non-zero entries of sparse matrix S_0 and don't know how many non-zeroes entries there are actually.

The object of this paper is to propose a tractable solution for recovering the sparse and low-rank components, and to analyze when our approach recovers these components exactly.

A. Principle Component Analysis (PCA)

Principal component analysis (PCA) is one of the most widely used multivariate techniques in data analysis. It is commonly used to reduce the dimensionality of data in order to examine its underlying structure.Perhaps the simplest and most useful assumption is that the data all lie near some low-dimensional subspace.This says that,if we stack all the data points as column vectors of a matrix M, the matrix should (approximately) have low rank. mathematically,

$$M = L_0 + N_0$$

But a single corrupted value can change the estimated $\hat{L}$ arbitrary far from true L_0. Here, N_0 is perturbation matrix. In classical PCA we seek to find the best rank(k) of low rank matrix by given equation,

$$minimize \ \ ||M - L||$$
$$subject\ to \ \ rank(L) \leq k$$

But, this problem cannot be solved by classical PCA because a small corruption can change the low-rank matrix.So,low rank matrix will be recovered with the help of Convex Optimization program called Principal Component Pursuit (PCP).

B. Convex Optimization

Our aim is to separate low-rank matrix L_0 and sparse matrix S_0 from the given data matrix.This problem can be solved by tractable Convex Optimization.Here,the separation will happen so accurately that if we add both sparse matrix and low-rank matrix we can get data matrix.We can do this with low complexity with PCP

$$minimize \qquad ||L||_* + \lambda ||S||_1$$
$$subject\ to \qquad L + S = M$$

Here, $||L||_* := \sum_i \sigma_i(L)$ is the nuclear norm of the matrix L,which is the sum of all the singular values of L and $||S||_1 := \sum_{ij} |S_{ij}|$ denotes l_1 norm which is addition of all the entries of long vector of matrix S. In short, perfect recovery from incomplete and corrupted entries is possible by convex optimization.

II. When does separation make sense?

Suppose the matrix M is equal to e1e1* (this matrix has a one in the top left corner and zeros everywhere else).Then since M is both sparse and low-rank, how can we decide whether it is low-rank or sparse component?

”Low Rank Component cannot be sparse.”

Because if this condition is not being fulfilled then separation won’t make any sense. It’s like your signal looks like noise and noise looks like a signal.We cannot have rows and columns of our data matrix that are completely orthogonal to the rest.

Here,even if we know that which entries are corrupted we cannot correct them.So,low-rank component shouldn’t be sparse.

we will borrow the general notion of incoherence introduced in Cand‘es and Recht [2009] for the matrix completion problem.this is an assumption concerning the singular vectors of the low-rank component. Write the singular value decomposition of L_0 as

$$L = U \sum V^*$$

Here, $\sum$ is a matrix with singular values as it’s diagonal entries. U is a left singular vector and V is a right singular vector.

III. Incoherence condition

So,it’s the condition to prove that low rank component is not sparse.If principal components were very sparse then the kind of problem which we face is hard to solve.Now we will measure the correlation of column space and row space by taking the projection of basis vector onto the column space.The incoherence condition with parameter μ states that,

$$max_i ||U^* e_i||^2 \leq \frac{\mu r}{n_1}, \qquad max_i ||V^* e_i||^2 \leq \frac{\mu r}{n_2} \tag{1}$$

$$||UV^*||_\infty \leq \sqrt{\frac{\mu r}{n_1 n_2}} \tag{2}$$

The incoherence condition asserts that for small values of μ, the singular vectors are reasonably spread out—in other words, not sparse.

Now the second issue will arrive when sparse matrix has low-rank.

”Sparse Component cannot be low-rank.”

Assume that the non-zero entries of sparse matrix are in a column or some few columns then the rank of the matrix will become less.Now due to some issues all the columns of S_0 vanishes.Then S_0 and L_0 would look like similar.So,we cannot recover S_0 and L_0 from the data matrix. To avoid such meaningless situations, we will assume that the sparsity pattern of the sparse component is selected uniformly at random.

IV. Main Result

Under these minimal assumptions, the simple PCP solution perfectly recovers the low-rank and the sparse components.But,here the low-rank component shouldn’t be too large and the sparse component should be reasonably sparse.

THEOREM 1.1. Suppose L_0 is n×n matrix and suppose that the support set of S_0 is uniformly distributed among all sets of cardinality m.Then, there is a numerical constant c such that with probability at least 1-cn^{-10} (over the choice of support of S_0), Principal Component Pursuit with λ =1/n is exact, that is, $\hat{L}$= L_0 and $\hat{S}$= S_0, provided that

$$rank(L_0) \leq \rho_r n \mu^{-1} (logn)^{-2} \; and \; m \leq \rho_s n^2 \tag{3}$$

In this equation, ρ_r and ρ_s are positive numerical constants.We would like to emphasize that the only “piece of randomness” in our assumptions concerns the locations of the nonzero entries of S_0; everything else is deterministic.So,here on an average all entries has a positive chance of being corrupted.We are gonna solve this problem with λ, which is equal to the inverse of the square root of one of the highest dimension value from matrix M.

Here,the location of the entries have been choosen at random.

$$||L||_* + \frac{1}{\sqrt{n_{(1)}}} ||S||_1, \quad n_{(1)} = max(n_1, n_2)$$

This equation will tell us exactly where the corruption has occured in the data entry.

V. Matrix Completion

Now,Instead of having only corruption, we are also having missing entries in our matrix.Corrupted Robust PCA is much more harder to be implemented mathematically.

*THEOREM 1.2.*Suppose L_0 is n×n, obeys above conditions and Ω_{obs} is uniformly distributed among all sets of cardinality m obeying m = 0.1n^2.Suppose for simplicity, that each observed entry is corrupted with probability τ independently of the others. Then, there is a numerical constant c such that with probability at least 1-cn^{-10}, Principal Component Pursuit with $\lambda = \frac{1}{\sqrt{0.1n}}$ (where n is maximum dimension) is exact, that is, $\hat{L}$ = L_0. Provided that

$$rank(L_0) \leq \rho_r n \mu^{-1} (logn)^{-2} \; and \; \tau \leq \tau_s. \tag{4}$$

Where ρ_r and τ_s are positive numerical constants

VI. Algorithm

```
1: initialize: $S_0 = Y_0 = 0, \mu > 0$
2: while not converged do
3:     compute $L_{k+1} = D_{1/\mu}(M - S_k + \mu^{-1} Y_k)$
4:     compute $S_{k+1} = S_{\lambda/\mu}(M - L_{k+1} + \mu^{-1} Y_k)$
5:     compute $Y_{k+1} = Y_k + \mu(M - L_{k+1} - S_{k+1})$
6: end while
7: output: L,S
```

VII. Conclusion

In short,perfect recovery from incomplete and corrupted entries is possible by convex optimization.With the help of Convex Optimization it is possible to recover sparse matrix and low-rank matrix from the given data matrix M.

References

[1]https://www.youtube.com/watch?v=DK8RTamIoB8
[2]https://www.quora.com/How-does-the-Netflix-movie-recommendation-algorithm-work
[3]https://en.wikipedia.org/wiki/Convex$_optimization$
[4]$http://lmafit.blogs.rice.edu/$