Robust Principle Component Analysis Project Report

Vidit C. Shah
1401078
School of Engineering and Applied Science

***Abstract*—The robust principal component analysis (robust PCA) problem has been considered in many Machine learning and Big data applications, where the goal is to decompose the data matrix into a low rank and a sparse matrix. In this paper we are going to study how we can recover a low rank component and sparse component using the convex program called Principal Component Pursuit. In given article we will solve the convex Principal Component Pursuit using Augmented Lagrange Multiplier(ALM) algorithm. In addition, we will also see real life example that how our algorithm solves the problem.**

I. INTRODUCTION

Robust principal component analysis (robust PCA) has received much attention in recent studies for ability to recover the low rank component and sparse component. So we can say that this is separation problem the motivation to solve this problem is suppose we have given a number 07 which addition of two numbers a and b and we are tell to find numbers a and b so now if we solve this type of problem then it will have enormous practical consequences. Here we have given data matrix M which is superposition typically a classical PCA of low rank matrix and perturbation matrix.In given data matrix M the columns of this matrix are data points so when we plot this points may be in high dimensional space.So when we do dimensionality reduction using the method of PCA is to give collection of points and try to find best low rank approximation of given data matrix. In given article we are going to solve the problem which is to recover the low rank matrix from highly corrupted data matrix (M) by efficient and scalable algorithm, so our equation form will be like $M = L_0 + S_0$.Where, L_0 is the low rank matrix and S_0 is assumed to be sparse and the sparsity pattern is uniformly random.The Principal Component Pursuits (PCP) estimates that solving

$$minimise \parallel L \parallel_* + \lambda \parallel S \parallel_1$$

$$subject\ to\ L + S = M$$

exactly recovers the low rank L_0 and the sparse component S_0.Here $\parallel L \parallel_*$ is $\sum_{n=1}^{rank(L)} \sigma_i$ is the nuclear norm of L and $\parallel S \parallel_1$ is $\sum_{i,j} \mid S_ij \mid$ is elementwise one norm of S.

II. ORGANIZATION OF REPORT

In above section we have seen the Introduction and motivation behid this paper now In Section 3 we will see where this our algorithm works means application of our algorithm.In next section 4 we will see which are difficulties are being faced ,assumptions and significant harder problem then Matrix completion.In next Section 5 we will see our Algorithm and the implemented code output in section 6.The conclusion is state in Section 7.

III. APPLICATION

We are decomposing the Data matrix into two components one is Low rank and other is sparse. So we can apply this algorithm on applications like Video Surveillance, Face Recognition, etc. In this type of application we are trying that our algorithm is capable enough to identify the foreground and background from given video frame. If we stack the video frames as a columns of data matrix M, then the L_0 corresponds to the stationary background and sparse component S_0 which are moving objects in video frame are the foreground.

IV. CHALLENGES AND ASSUMPTIONS

As mention above that this is separation problem means to identify the low rank component and sparse component from given data matrix so the issue is if our data matrix M is in form of $e_1e_2^*$ then it became very hard to identify the low rank matrix and sparse matrix. So we are imposing that low rank component is not sparse and the sparsity pattern is uniformly random. The other issue which is mentation in this article that we wish to recover the components but we can see only fractions of entry and from those fractions of entries we dont know which one is corrupted. So we can say that this is significant extension to the matrix completion problem. The algorithm which is given to us in this article, we are able to recover the components.As discuss above the Sparse matrix should be uniformly random and the low rank matrix decomposition as

$$L_0 = U\Sigma V^* = \sum_{i=1}^{r} \sigma_i u_i v_i^*$$

where r is the rank of the matrix,σ_1,..,σ_r are the positive singular values,U and V are the matrices of left and right-singular vectors.There are conditions given to us for convex programming to succeed.these are phrased in two quantities.The first is the maximum ratio between the L_∞ norm and the operator norm, restricted to the subspace generated by matrices whose row or column spaces agree with those of L_0. The second is the maximum ratio between the operator norm and the L_∞ norm, restricted to the subspace of matrices that vanish off the support of S_0.when product of this two quantities is small then the recovery is exact which is proved in Chandrasekaran et al. [2009].

A. Theorem 1.1

Let, L_0 is $n1 * n2$, where PCP with $\lambda = \frac{1}{\sqrt{n_1}}$ succeeds with the probability at least $1 - cn_1^{-10}$, provided that rank(L_0) $\leq \rho_r \ n_2 \ \mu^{-1} \ (\log n_1)^{-2}$ and $m \leq \rho_s n_1 n_2$,Here $\rho_s and \rho_r$ are the numerical constants, m is cardinality and $n_1 = max\,(n_1, n_2)$, $n_2 = min\,(n_1, n_2)$.

V. Algorithm

In Theorem 1.1 of given article it is prove that we can recover the low rank component L_0 of dimension $n_1 * n_2$ in polynomial time. this is practically possible due to the progress in scalable algorithm for nonsmooth convex optimization means we are minimizing the l_1 and nuclear norm. one algorithm which is suggested is iterative thresholding algorithm to minimize the l_1 norm and nuclear norm by shrinking the singular value of given matrix ,this means reducing the complexity to each iteration is the cost of SVD,but this algorithm converge very slowly, requiring approx 10^4 iterations. In article the other two algorithms which are Accelerated Proximal Gradient (APG) and augmented Lagrange multiplier(ALM) to get appropriate Low rank component and sparse component from grossly corrupted data matrix. The APG algorithm can solve this problem at least 50 times faster then iterative thresholding algorithm,but the issue with this APG algorithm is that it depends strongly on the design of good continuation schemes, So due to this the Augmented Lagrange multiplier(ALM) algorithm introduce in Lin et al. and Yuan and Yang is used to solve convex PCP problem. ALM algorithm works stably across a wide range of problem with no tuning parameter. From the experiment it is prove that this algorithm achieves much higher accuracy then ALM algorithm.
The ALM algorithm operates on the augmented Lagrangian

$$l\,(L, S, Y) = \|L\|_* + \lambda \|S\|_1 + \langle Y, M - L - S\rangle + \frac{\mu}{2} \|M - L - S\|_F^2$$

This Lagrange multiplier algorithm solves the PCP by repeatedly setting $(L_k, S_k) = argmin_{L,S} l\,(L, S, Y_k)$.Then we are updating the Lagrange multiplier matrix. For our low rank and sparse component we are finding $min_L l(L, S, Y)$ and $min_S l(L, S, Y)$ insted of solving a sequence of convex programs. In our algorithm first we are minimizing l with respect to L (fixing S) $\arg\min_L l(L, S, Y) = D_{\frac{1}{\mu}}(M - S + \mu^{-1}Y)$,then we are minimizing l with respect to S (fixing L),$\arg\min_S l(L, S, Y) = S_{\frac{\lambda}{\mu}}(M - L + \mu^{-1}Y)$.

In ALM algorithm very well depends on choice of μ and terminate condition.So we are selecting $\mu = n_1 n_2 / 4 \|M\|_1$ and terminate the algorithm when $\|M - L - S\|_F \leq \delta \|M\|_F$ with $\delta = 10^{-7}$.

Algorithm 1 Principal Component Pursuit by Alternating Directions [Lin et al. 2009a;Yuan and Yang 2009]

Initialize: $S_0 = Y_0 = 0, \mu > 0$
while not converged **do**
  compute $L_{k+1} = D_{\frac{1}{\mu}}(M - S_k + \mu^{-1}Y_k)$;
  compute $S_{k+1} = S_{\frac{\lambda}{\mu}}(M - L_{k+1} + \mu^{-1}Y_k)$;
  compute $Y_{k+1} = Y_k + \mu(M - L_{k+1} - S_{k+1})$;
end while
Output: *L,S*.

VI. Implementation

Figure 1.1 : Input Matrix M

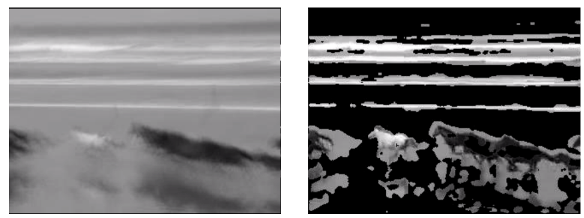

Figure 1.2: Low Rank and Sparse Matrix

VII. CONCLUSIONS

The problem is to recover the Low rank component and Sparse component from given data matrix.The low rank and sparse subspace are naturally distinguishable under incoherence assumptions. It is solved using Augmented Lagrange Multiplier (ALM) algorithm not for only square matrix but also for rectangular and grossly corrupted data matrix.

VIII. APPENDIX

Convex Optimization Problem - A Convex optimization problem is a problem where all constraints are convex functions, and the objective is a convex function if minimizing ,or a concave function maximizing.
Norm - In this article they have use five norms. The operator norm or 2-norm denoted by $\|X\|$, then the Frobenius norm is denoted by $\|X\|_F$.the other two norms are l_1 norm and l_∞ norm.

References

[1] ”Robust Principal Component Analysis?”,EMMANUEL J. CANDE‘S and XIAODONG LI,YI MA,JOHN WRIGHT.
[2] ”Robust Principal Component Analysis with Side Information”,Kai-Yang Chiang,Cho-Jui Hsieh.
[3] Video Link: https://www.youtube.com/watch?v=DK8RTamIoB8