Robust Principal Component Analysis

Karan Patel, 1401113
School of Engineering and Applied Science
Ahmedabad University
Email: karan.u.btechi14@ahduni.edu.in

***Abstract*—We have use the use these phenomena at many places. We have a data matrix M which is the superposition of two things low-rank matrix also known as original data matrix and sparse matrix also known as corrupted matrix.As all the feasible decomposition techniques of a matrix the output of this program is addition of l_1 norm of low-rank matrix and weighted nuclear form of sparse matrix .It has been proved that you can recover it under certain assumptions by doing a complex program which is called as Principal Component Pursuit and in addition to that the data matrix given to us is corrupted and some time the fraction of the entries missing as well.We will discuss it's application in video surveillance where you can detect the steady object and the objects which are moving and in Netflix Recommendation Systems.**

Key words: Principal Component Pursuit, l_1 norm, nuclear norm, low-rank matrices, sparse matrices, Singular Value Decomposition(SVD), Augmented Lagrange Multiplier, Robust Regression

I. Introduction

Principal Component Analysis is mainly used in area of optimization of data, statistics, and machine learning. It has decomposed the bigger matrix in to smaller and easily understandable matrix form. but the situation is different in real life as we know the data is not small it is way too big so you cannot do it by your Classical PCA or SVD.

$$M = L_0 + N_0$$

where M is data matrix L_0 is low-rank matrix and N_0 is perturbation matrix.

As Robust PCA can solve this problem as it is robust to outliers and corrupted observations as shown in figure 1 the data matrix M is the sum of L_0 low-rank matrix and S_0 sparse matrix.The L_0 is same as in the classical pca but the different part from the classical pca is S_0 instead of N_0. As N_0 is small noise term which is replaced by S_0 which can take large magnitude and there support is assumed to be sparse but unknown.

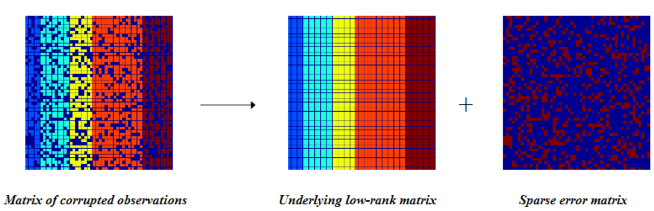


Figure 1

II. Applications

There are many applications of the Robust PCA like Video Surveillance, Face Recognition, Latent Semantic Indexing, Ranking and Collaborative Filtering.As we have done only th video surveillance in these the each frame of the video is our data matrix M and we need to differentiate the moving objects and the stationary object in the foreground.Here L_0 naturally corresponds to the stationary object and S_0 corresponds to the moving objects.As we know that each frame had so many pixels and so we cannot get the truly scalable solution of this problem.

III. Separation of Matrix

The data matrix is M an it is separated in two parts L_0 and S_0 under some constrains.Then

$$M = L_0 + S_0$$

and we have to get the low-rank matrix and sparse matrix.But in the beginning we don't know the number of errors in the low rank matrix L_0 and it is hard to extract them so taken some constraints and we will relax these complex problem

$$minimize \quad ||L||_* + \lambda||S||_1$$
$$Subject\ to \quad M = L_0 + S_0$$

Where, $||L||_* = \Sigma_i \sigma_i(L)$ denote the nuclear norm of matrix L and $||S||_1 = \Sigma_{ij}|M_{ij}|$ of the matrix L denote the l_1 norm

So under this condition the PCP will work best an give us the low-rank matrix and sparse matrix exactly.AS theoretically this also can be proven as L_0 grows almost linearly in the dimension of the matrix, and the errors in S_0 are up to a constant fraction of all entries.

IV. Assumptions

In many cases it is possible that the data matrix M has 1 in starting rows and columns and other rows and columns may be 0.Therefore M can be low-rank or it can be sparse and to make problem easy we have taken two assumptions.

A. Low-rank matrix L_0 can not be sparse.

B. Sparse matrix S_0 can not be low-rank

V. Algorithms

First we have written PCP algorithm for geetting the low-rank matrix and sparse matrix from the data matrix and we have also done the algorithm for the large data by the convex optimization algorithm.we had used the ALM and taken it's special case known as Alternating Direction Method.

A. *Principal Component Pursuit*

We have the PCP in the matrix separation part and now we this is the result of PCP. In this paper we had define $n(1) = max(n_1, n_2)$ and $n(2) = min(n_1, n_2)$. Suppose L_0 is a square matrix of rank any arbitrary rank nxn, such that it obeys the assumptions given above. Suppose that the support set Ω of S_0 is uniformly distributed among all sets of cardinality m, and that $sgn([S_0]_{ij}) = \sum_{ij} for\ all(i,j) \in \Omega$.Then there is a numerical constant c such that with probability at least $1 - cn^{-10}$, *Principal Component Pursuit* with $\lambda = 1/\sqrt{n}$, returns exact low-rank and sparse matrix provided that

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

In the above equation ρ_r and ρ_s are positive numerical constants. In general case this nxn;dimension of L_0 is $n_1 x n_2$, PCP with $\lambda = 1/\sqrt{n_{(1)}}$, succeeds with the probability at least $1 - cn_{(1)}^{-10}$, provided that $rank(L_0) \leq \rho_r n_{(2)} \mu^{-1}(logn_{(1)})^{-2}$ and $m \leq \rho_s n_1 n_2$ Thus the the claim we made can be restated

$$minimize \quad ||L||_* + 1/\sqrt{n_{(1)}}||S||_1$$
$$subject\ to \quad L + S = M$$

Here it is to note that the parameter λ has not to be balanced between L_0 and S_0 and is independently found to be $\lambda = 1/\sqrt{n_{(1)}}$.

B. *Augmented Lagrange Multiplier*

We had used ALM's special case known as Alternating Directions Method.

ALGORITHM 1: (Principal Component Pursuitby Alternating Directions [Lin et al. 2009a; Yuan and Yang 2009])

initialize: $S_0 = Y_0 = 0, \mu > 0$.
while not converged **do**
  compute $L_{k+1} = \mathcal{D}_{1/\mu}(M - S_k + \mu^{-1}Y_k)$;
  compute $S_{k+1} = \mathcal{S}_{\lambda/\mu}(M - L_{k+1} + \mu^{-1}Y_k)$;
  compute $Y_{k+1} = Y_k + \mu(M - L_{k+1} - S_{k+1})$;
end while
output: L, S.

The value S_0 and L_0 are updated in this algorithm. Let $S_\tau : R \longrightarrow R$ denote the shrinkage operator $S_\tau[x] = sgn(x)max(|x| - \tau, 0)$ and extend it to matrices by applying it to each element it shows that

$$arg\ min_S\ l(L, S, Y) = S_{\lambda/\mu}(M - L + \mu^{-1}Y)$$

Similarly for matrices X, let $D_\tau(X)$ denote singular value threshold operator given by $D_\tau(X) = US_\tau(\sum)V^*$ and thus

$$arg\ min_L\ l(L, S, Y) = D_{1/\mu}(M - S + \mu^{-1}Y)$$

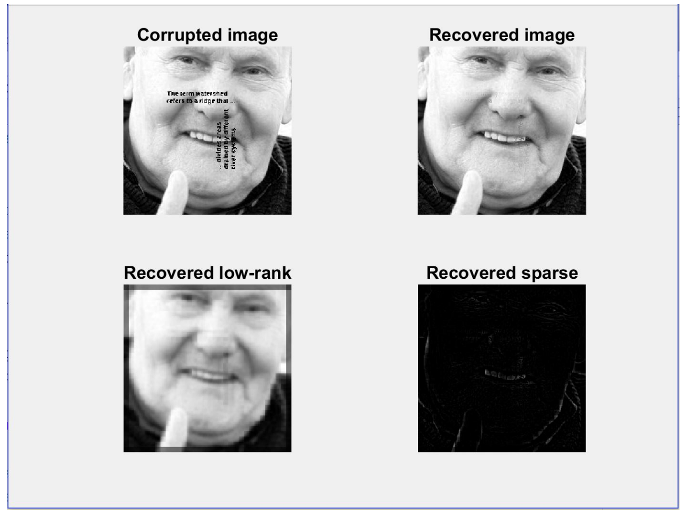
Here we suggest $\mu = n_1 n_2/4||M||_1$ and we terminate the algorithm when $||M - L - S||_F \leq \delta||M||_F$ with $\delta = 10^{-7}$

VI. Results

We had used

The results for the above proposed *Principal Component Pursuit* and *Alternating Direction methods* are simulated using Matlab as a tool and a user input image.

We have taken a corrupted data matrix M which is a corrupted image and from this corrupted data matrix we have



achieved the L_0 low-rank matrix and S_0 completely.Here the speculation and the shadowing effect of the image are stored in S_0.

As shown above from the corrupted image M having dimensions 578x400, thus $M \epsilon R^{578x400}$ we have successfully recovered L_0 and have recorded speculations and image shadowing in S_0. The program took 5.13 $seconds$ to converge with 143 iterations. The rank of low-rank matrix ; **rank(L)= 36** which suggests it is indeed a low rank. The cardinality of set of the sparse matrix S_0 is **card(S)= 227902**. The error rate observed was 2.38.

VII. Conclusion

The Low-rank matrix L_0 and Sparse matrix S_0 has successfully recovered from the corrupted image M.and we can get it for bigger image but the computation time will increase as you increase the pixels in the image. The algorithm can be extended further if we neglect some of the assumptions.

References

[1] Cands, E. J., Li, X., Ma, Y., Wright, J. (2011). Robust principal component analysis?. Journal of the ACM (JACM), 58(3), 11. Chicago

[2] H. Kopka and P. W. Daly, *A Guide to LaTeX*, 3rd ed. Harlow, England: Addison-Wesley, 1999.

[3] Robust Principal Component Analysis: Exact Recovery of Corrupted Low-Rank Matrices by Convex Optimization. John Wright , Yigang Peng, Yi Ma Visual Computing Group Microsoft Research Asia

[4] Analysis of Robust PCA via Local Incoherence. Huishuai Zhang, Yi Zhou, Yingbin Liang.

[5] //http://ieeexplore.ieee.org/abstract/document/937541/

[6] http://papers.nips.cc/paper/3704-robust-principal-component-analysis-exact-recovery-of-corrupted-low-rank-matrices-via-convex-optimization