Project 1: A Report on Robust Principal Component Analysis

Aashima Yuthika - 1401071
School of Engineering and Applied Sciences
Ahmedabad University
Subject: Algorithms and Optimisation for Big Data
February 08, 2017

***Abstract*—Principal Component Analysis (PCA) is an important technique in data analysis and dimensionality reduction in a lot of cases. However, it may not always be very 'robust' in many real life scenarios. This is where Robust PCA comes in. It has been assumed that the data matrix that we have is the superposition of a low-rank matrix and sparse matrix. We are looking to extract each component individually. Thus, essentially what we want is to see if we can recover the principal components of a data matrix, even if a fraction of its entries are either corrupted or missing entirely. This paper in essence talks about Taking a matrix with errors or missing data and completing it (read: Data Completion). The authors have discussed this technique's application in video surveillance, where it helps in object detection in a cluttered background, and in face recognition in, where it tries to remove shadows and other peculiarities in images of faces.**

I. Introduction

We are given a large data matrix M which can be decomposed as:

$$M = L_0 + S_0$$

Where, L_0 is a low rank matrix and S_0 is a sparse matrix. There is no knowledge of the low-dimensional column and row-space of L_0 and no knowledge of of the locations of the non-zero entries of S_0. We now want to know if we can recover the low-rank and the sparse components of both of these matrices as accurately as possible and as efficiently as we can.

The data matrix, as mentioned previously, is large and hence needs to be handled in a scalable manner too. PCS is a tool that is very widely used for data analysis and dimensionality reduction. However, it's not very vigorous when it comes to highly corrupted observations and it is evident from the fact that an excessively corrupted entry in M can easily render the estimate $\hat{L}$ very far from the original L_0.

The problem thus studied here is an idealized version of the Robust PCA in which a low-rank matrix L_0 is being tried to recover from a grossly corrupted data matrix $M = L_0 + S_0$. This very different from the small noise term N_0 in the classic PCA.

A. Important Assumptions

Let's suppose that the matrix M is equal to $e_1 e_1^*$ (that is, it has a *1* in the top left corner and zeros everywhere else). Then, as M is both sparse and low-rank, it's not possible to decide whether it is just low rank or sparse. Hence, it has been imposed that the low-rank component L_0 is not sparse. The notion of coherence is thus used for the matrix completion problem. This is an assumption concerning the singular vectors of the low-rank component. The singular value decomposition of $L_0 \epsilon R^{n_1 \times n_2}$ as:

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

where r is the rank of the matrix, $\sigma_1, ..., \sigma_r$ are the positive singular values, and $U = [u_1, ..., u_r]$, $V = [v_1, ..., v_r]$ are the matrices of left and right singular vectors. Then the incoherence condition with parameter μ states that

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

and

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

Here, $\|M\|_\infty = max_i |M_{ij}|$, that is, the l_∞ norm of M seen as a long vector.

Another issue may be that the sparse matrix S_0 can be low-rank. This will happen if all the non-zero entries of S occur in a column or in a few columns. Furthermore, if the first columns of S_0 is the opposite of that of L_0 and all other columns in S_0 vanish. Then it is clear that L_0 and $S_0 4$ cannot be recovered as $M = L_0 + S_0$ will have a column space equal to or included in that of L_0. Thus, it is assumed that the sparsity pattern of the sparse component is randomly uniform.

II. Theorems

A. Theorem 1.1

Suppose L_0 is $n \times n$, obeys the incoherence conditions. Fix any $n \times n$ matrix Σ of signs. Suppose that the support set Ω of S_0 is uniformly distributed among all sets of cardinality m, and that $sgn([S_0]_{ij}) = \Sigma_{ij}$ for all $(i,j)\epsilon\Omega$. Then ,there is a numerical constant c such that with probability at least $1 - cn^{-10}$ (over the choice of support of S_0), the Principal Component Pursuit with $\lambda = \frac{1}{\sqrt{n}}$ is exact, that is, $\hat{L} = L_0$ and $\hat{S} = S_0$, provided that

$$rank(L_0) \leq \rho_r n \mu^{-1} (\log n)^{-2}$$

and

$$m \leq \rho_s n^2$$

In this equation, ρ_r and ρ_s are positive numerical constants. In the general rectangular case, where L_0 is $n_1 \times n_2$, PCP with $\lambda = \frac{1}{\sqrt{n_{(1)}}}$ succeeds with probability at least $1 - cn_{(1)}^{-10}$, provided that $rank(L_0) \leq \rho_r n_{(2)} \mu^{-1} (\log n_{(1)})^{-2}$ and $m \leq \rho_s n_1 n_2$

Thus, matrices L_0 whose singular vectors - or the Principal Components - are spread and can be recovered almost perfectly from completely unknown corruption patterns (as long as they are randomly distributed). All that is required is that the singular vectors of L_0 are not too spiky (i.e, they are more or less uniformly distributed).

B. Theorem 1.2

Suppose L_0 is $n \times n$, obeys the incoherence conditions and that 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 $\lambda = \frac{1}{\sqrt{0.1n}}$ is exact, that is, $\hat{L} = L_0$, provided that

$$rank(L_0) \leq \rho_r n \mu^{-1} (\log n)^{-2}$$

and

$$\tau \leq \tau_s$$

In this equation, ρ_r and τ_s are positive numerical constants. For general $n_1 \times n_2$ rectangular matrices, PCP with $\lambda = \frac{1}{\sqrt{0.1 n_{(1)}}}$ succeeds from $m = 0.1 n_1 n_2$ corrupted entries with probability at least $1 - cn^{-10}$, provided that $rank(L_0) \leq \rho_r n_{(2)} \mu^{-1} (\log n_{(1)})^{-2}$.

The above theorem basically tries to say that we can recover the corrupted data perfectly by convex optimisation from both incomplete and corrupted data.

III. Algorithm

Algorithm 1 Principal Component Pursuit by Alternating Directions

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

IV. Applications

There are some very important real life applications of Robust PCA wherein it is used to separate the low-rank and sparse distributions from a given data set. Some of them are as follows:

1) **Video Surveillance**
It is often required in a set of video frames to identify the activities that stand out from the background. If these various video frames are stacked upon on another to form a data matrix M then the low-rank component L_0 refers to the stationary background and the sparse components S_0 are the moving objects in the foreground as these occupy only a small fraction of the whole video frame. Thus, by separating the low-rank and the sparse components of the data matrix we can in essence separate the foreground from the background.
2) **Face Recognition**
It is established that images of a human's face can be approximated by a low-dimensional sub-space. Being able to correctly recognise a face is crucial to many applications such as face recognition and alignment. However, realistic faces and scenarios often suffer from self-shadowing, specularities, or saturations in brightness, which make it a difficult task and subsequently compromise the recognition performance. Thus, Robust PCA can be used to remove these shadows from faces for a more proper recognition.

V. Results

Given below, is an image that has been corrupted by a text superimposed on it. The figure below shows the various kinds of image matrices recovered from the data image.

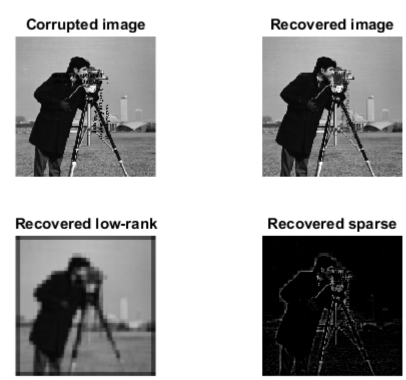

Fig. 1: Corrupted Image is the input image. Recovered Image is the actual recovered image. Low Rank Image is the low rank pixel data recovered from the matrix. Sparse Components have also been recovered from it.

As we can see the low rank image that has been obtained is practically free from the corruption. Even the total recovered image is almost free from corruption, so much so that it's hardly visible.

REFERENCES

[1] E.J.Candes, Xiaodong Li, Yi Ma, John Wright, *Robust Principal Component Analysis*, Journal of the ACM, Vol. 58, No. 3, Article 11, May 2011.

[2] https://github.com/dlaptev/RobustPCA