Summary of Robust Principal Component Analysis

Chintan Saraviya
Information and Communication Technology(ICT)
School of Engineering and Applied Science
Ahmedabad University
Email: chintan.saraviya@iet.ahduni.edu.in

***Abstract*—Principal Component Analysis (PCA) is a statistical technique widely used for the reduction of dimensionality. A major disadvantage of this method is that it is very sensitive to abnormal values, that is, a single corrupted data point in the data matrix will result in a significant deviation from expected output. In computer vision applications, outliers sample due to noises or sensor failures. This article evaluates a method named "Robust PCA" which under appropriate assumptions, does the decomposition of a data matrix into a lower rank component and a sparse component.**

I. Introduction

Principal Component Analysis (PCA) is a tool for providing an approximation to a data matrix M, in order to reduce the size or capture the main directions of variation of the data. More recently, more emphasis has been placed on robust PCA, which is more robust to realistic flaws in data such as mass-tailed outliers. The robust principal component analysis formulates a data decomposition, $M = L_0 + S_0$, which decomposes M into a low rank component L_0 (capturing trends through the data matrix) and a sparse component S_0 (capturing outliers that can mask the baseline values), which we seek to separate based only on the observation of the data matrix M. Depending on the application, we can focus on one or the other component:

- In some parameters, the sparse component S_0 may represent undesirable aberrant values, eg. corrupted data - we may want to clean up the data by removing the outliers and retrieving the lower-ranked component L_0.
- In other parameters, the sparse S_0 component can contain information of interest, for example, in image or video data, S_0 can capture the prominent objects that are of interest, while L_0 can capture components that we wish to remove.

II. Applications

- Video Surveillance: Given a sequence of video surveillance frames, we often need to identify activities that stand out from the background. kground and the sparse component S0 captures the moving objects in the foreground. However, each image frame has thousands or tens of thousands of pixels, and each video fragment contains hundreds or thousands of frames. It would be impossible to decompose M in such a way unless we have a truly scalable solution to this problem.
- Face Recognition: Images of a person's face can be approached by a small subspace. Being able to properly recover this subspace is crucial in many applications such as facial recognition and alignment. However, realistic face images often suffer from self-shading, speculation, or saturation in brightness, which make this a difficult task and subsequently compromise recognition performance.
- Latent Semantic Indexing: Web search engines often need to analyze and index the content of an enormous corpus of documents. The basic idea is to gather a document matrix M whose inputs generally encode the relevance of a term (or word) on a document such as the frequency it appears in the document.
- Ranking and Collaborative Filtering: The problem is to use incomplete classifications provided by users to predict a given user's preference. This problem is usually expressed as an end-of-matrix problem of lower rank. However, as the data collection process often lacks control or is sometimes even adhoc, a small part of the available rankings could be noisy and even falsified. The problem is more difficult because we have to complete the matrix and correct the errors.

Similar problems also arise in many other applications such as the graphic model Learning, linear identification of systems and decomposition of coherence in optical systems.

III. Assuptions

The objective is to decompose the matrix M into a L_0 component of lower rank and a similar component S_0 such that $M = L_0 + S_0$. Some assumptions are made for the proposed solution. The matrix of lower rank L_0 is not sparse and sparse matrix S_0 is not of low rank. Showing the singular value decomposition of L_0 as it is an $n_1 \times n_2$ matrix:

$$L_0 = U\Sigma V^T \tag{1}$$

Let r be the rank of matrix L_0. Then, the condition with parameter μ is written as:

$$max_i ||U^T e_i||^2 \leq \frac{\mu r}{n_1} \tag{2}$$

$$max_i ||V^T e_i||^2 \leq \frac{\mu r}{n_2} \tag{3}$$

$$||UV^T||\infty \leq \sqrt{\frac{\mu r}{n_1 n_2}} \tag{4}$$

Here, $||M||_\infty = \max_{ij}|M_{ij}|$ is the l_∞ norm of M seen as long vector. To avoid the sparse matrix S_0 from being low-rank matrix, the pattern of sparsity of sparse component is selected uniformly at random. If we have a matrix M of dimensions $n_1 \times n_2$ then under some weak assumptions the Principal Component Pursuit with $\lambda = \frac{1}{\sqrt{n}}$ is true, such that, $\hat{L} = L_0$ and $\hat{S} = S_0$, given that

$$rank(L_0) \leq \rho_r n \mu^{-1} (logn)^{-2} where, m \leq \rho_s n^2 \quad (5)$$

In the paper, for recovering low-rank matrix L_0, minimize $||L||_* + \lambda ||S||_1$ subject to L + S = M

IV. Algorithm

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
Output: *L,S*

Algorithm 1: PCP by Alternating Direction

V. Observations

λ	0.0208
Iterations	483
Rank(L)	4
Error	0.000010
Card(S)	403599
Elapsed Time	87.902417 seconds
Size of M	180 $\times$ 2304

VI. Implementation

Fig. 1: Original Matrix

Fig. 2: Low Rank Matrix

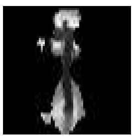

Fig. 3: Sparse Matrix

The above algorithm was implemented in matlab tested on a video surveillance frame for background - foreground seperation as shown in the images below

VII. CONCLUSIONS

From this paper, we understand that under few minimal assumptions and coherence conditions we can see that the low rank matrix L and sparse matrix S can be recovered. RPCA offers some advantages such as robustness to highly corrupted set of samples or data. It is indeed possible to decompose a data matrix into its corresponding low-rank matrix and sparse components and improve the data.

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] http://statweb.stanford.edu/ candes/papers/RobustPCA.pdf
[4] https://github.com/dlaptev/RobustPCA