Robust Principal Component Analysis

Vatsal Patel
1401123
SEAS-AU

***Abstract*—In this era of Big Data there are many important applications in which the data under study can naturally be modeled as a low-rank plus a sparse contribution. PCA(Principal Componant Analysis) is widely used statistical tool for data analysis and dimensionality reduction today in field of big data. But what if there are corroupted values in given data martix? A single grossly corrupted entry in M could render the estimated low rank matrix arbitrarily far from the true Low rank matrix. In this case it is possible to recover both the low-rank and the sparse components exactly by solving a very convenient convex program called Principal Component Pursuit(PCP); we can recover principle component of given data matrix even if some of the entries are arbitrarily courrepted. Application of this topic is widely used in area of Video Surveillance, Face Recognition, Latent Semantic Indexing and Ranking and Collaborative Filtering.**

Keywords: PCA(principal componant analysis), Low rank matrix, Sparse matrix, principal component pursuit(PCP).

I. Introduction

In the area of image processing PCA is vital and widely used stastical tool. Suppose we are given a image or a video and we want to seperate out the background and foreground,In this case Robust principal component analysis is useful for its ability to recover the low rank model from sparse noise where we can say that low rank matrix happens to be background and sparse matrix happens to be foreground. Other than that there are many important applications in which the data under study can naturally be modeled as a low-rank plus a sparse contribution. One of the application where this occurs is in video Surveillance where we are given video and each frame can be consider as a image or data matrix and we want to separate a background from moving foreground objects.

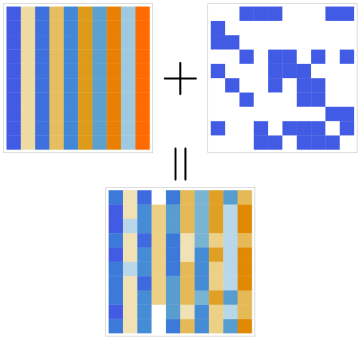

Fig. 1: Image seperation

In above example suppose we are given a large data matrix which can be decomposed as,

$$M = L_0 + S_0,$$

where low rank matrix is L_0 and sparse matrix is S_0, We do not know the low-dimensional column and row space of L_0, not even their dimension. Similarly, we do not know the locations of the nonzero entries of S_0, not even how many there are.

A. Classical Principle Component Analysis

in classical PCA, the entries in S_0 can have arbitrarily large magnitude, and their support is assumed to be sparse but unknown. More precisely, this says that if we stack all the data points as column vectors of a matrix M, the matrix should have low rank would look like,

$$M = L_0 + N_0$$

where N_0 is Purturbation matrix. In classical PCA we seek to find the best rank(k) of low rank matrix by given equation,

$$minimize \;\; ||M - L||$$

$$subject\ to \;\; rank(L) \leq k$$

A single grossly corrupted entry in M could render the estimated $\hat{L}$ arbitrarily far from the true L_0. Thats why this problem can not be solved by classical PCA therefore a substantial amount of low rank matrices is recoverable by solving a convenient convex program called Principal Component Pursuit(PCP).

B. Convex Optimizationon or Robust Principle Component Analysis

A small portion of the available rankings could be noisy and even tampered with. The problem is more challenging since we need to simultaneously complete the matrix and correct the errors. Robust PCA, in which we aim to recover a low-rank matrix L_0 from highly corrupted measurements M. Algorithmically this problem can be solved by efficient and scalable algorithms, at a cost not so much higher than the classical PCA. We can do that by PCP with low complexity.

$$minimize \qquad ||L||_* + \lambda||S||_1$$

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

Where $||L||_* := \sum_i \sigma_i(L)$ is the nuclear norm of the matrix L, that is the sum of the singular values of L, and $||S||_1 := \sum_{ij} |S_{ij}|$ denotes l_1 norm which is basically addition of all the

entries of long vector of matrix S. However, it is often possible to improve performance by choosing λ in accordance with prior knowledge about the solution. For example, if we know that S is very sparse, increasing λ will allow us to recover matrices L of larger rank.

II. Condition required for PCP to work

When we apply PCP to any data matrix there seems to not be enough information to perfectly disentangle the low-rank and the sparse components. For instance, suppose the matrix M is equal to $e_1 e_1^*$ (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? We can solve this problem by general notion of incoherence for the matrix completion problem, the incoherence condition asserts that for small values of μ, the singular vectors are reasonably spread out-in other words,not sparse.

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

where U is left singular vector and V is right singular vector of matrix L_0 and V^* is transpose of V, the incoherence condition with parameter μ look like,

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

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

from this condition we can say that when we do projection of basis vector of U on column space, then large value of μ indicates that L_0 and S_0 looks similar and when μ is small it means that they are not same and in other word entries are spreaded out and are not is one perticular coulmn.

Theorem 1.1 Given that given low rank component is not too large and sparse component is reasonably sparse simple PCP will work perfectly. Now Suppose that the support set Ω 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 $1 - cn^{-10}$, with $\lambda = 1/\sqrt{n}$ is exact, that is, $\hat{L} = L_0$ and $\hat{S} = S_0$, which means that $\hat{L}$ which is basically courrepted L_0 is almost same as L_0

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

where ρ_r and ρ_s are positive numerical constants. If above condition setisfy then and then only we can say that given data matrix is recoverable. In other words, matrices L_0 whose singular vectors or principal components are reasonably spread can be recovered with probability nearly one from arbitrary and completely unknown corruption patterns, and minimizing the below condition will always returns the correct answer.

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

III. Matrix completion problem

The matrix completion problem is that of recovering a low-rank matrix from only a small fraction of its entries, and by extension, from a small number of linear functionals. In other word not only some of the entries are courrepted but some of the entries of data matrix are even missing. In this case a fraction of observed entries available and the other missing, but we do not know which one are available, while the other is not missing but entirely corrupted altogether. One of the solution of this is to simultaneously detects the corrupted entries, and perfectly fits the low-rank component to the remaining entries that are deemed reliable.

Suppose L_0 is $n \times n$, obeys the conditions (1) and (2), 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,

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

where ρ_r and τ_s are positive numerical constants.By this equation we cab perfect recovery from incomplete and corrupted entries is possible by convex optimization.

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

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

Fig. 2: Pseudo code of Algorithm

IV. Conclusion

From this articel we can conclude that, perfect recovery from corrupted and incomplete data entries is possible by convex optimization program Principle Component Pursuit(PCP). Using 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=DK8RTamIoB8t=3083s

[2] https://en.wikipedia.org/wiki/Convex-optimization

[3]https://en.wikipedia.org/wiki/Principal-component-analysis

[4]http://nlp.stanford.edu/IR-book/html/htmledition/low-rank-approximations-1.html