Algorithms and Optimization for big data

Robust Principal Component Analysis

***Abstract*— We have a data matrix, which is the superposition of a low-rank component and a sparse component.Paper proves that under some suitable assumptions, 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.The paper uses principled approach of robust principal component analysis which assert that one can recover the principal components of a data matrix even though a positive fraction of its entries are arbitrarily corrupted. This extends to the situation where a fraction of the entries are missing as well. An algorithm for solving this optimization problem, and present applications in the area of video surveillance, where our methodology allows for the detection of objects in a cluttered background, and in the area of face recognition, where it offers a principled way of removing shadows and specularity in images of faces.**

I. INTRODUCTION

Given a large data matrix M,

$$M = L_0 + N_0 \tag{1}$$

where L_0 is low rank component and N_0 is sparse component of arbitrary magnitude.We have no information regarding the dimensions of both components as well as non zero entries location in sparse matrix.A scalable and efficient solution to this problem is possible which is result of this paper.

First Assumption : Data must lie in some small low dimensional subspace.If we stack all data points as column vectors of matrix M,the matrix should have low rank.

$$M = L_0 + N_0 \tag{2}$$

where L_0 has low-rank and N_0 is a small perturbation matrix Classic PCA gives best rank k estimate of L_0 by solving

$$\begin{gathered}\text{minimize } \| \text{M - L} \| \\ \text{subject to rank(L)} \leq \text{K}\end{gathered}$$

$\| M \|$ denotes the 2-norm; that is, the largest singular value of M.This problem can be efficiently solved via the singular value decomposition (SVD) and enjoys a number of optimality properties when the noise N_0 is small and independent and identically distributed Gaussian(**Second Assumption**).

Robust PCA is widely used as statistical tool for data analysis and dimension reduction.But if entries are corrupted there is high probability that we can get L_0 far from the original one.There are functional approaches to this problem like multivariate training,alternating minimization and random sampling techniques out of which none of them gives polynomial time bound.The problem we study here can be considered an idealized version of Robust PCA, in which we aim to recover a low-rank matrix L_0 from highly corrupted measurements M = L_0 + S_0 . Unlike the small noise term N_0 in classical PCA, the entries in S_0 can have arbitrarily large magnitude, and their support is assumed to be sparse but unknown.

There are various real life applications like video surveillance,face recognition,latent semantic indexing,ranking and collaborative filtering.

II. PROBLEM AND APPROACH TO SOLUTION

The separation problem seems impossible to solve since the number of unknowns to infer for L_0 and S_0 is twice as many as the given measurements in

$$M \in R^{n1*n2}$$

.In this article, we are going to see that it can be solved by tractable convex optimization. Let $\| M \|$:= $\sigma_i j|M_i j|$ denote the nuclear norm of matrix M,sum of singular values of M. Under weak assumptions ,the Principal Component Pursuit(PCP) estimate solving

$$\begin{gathered}\text{minimize } \| \text{L} \|_* + \lambda \| \text{S} \|_1 \\ \text{subject to rank(L)} \leq \text{K}\end{gathered}$$

exactly recovers the low-rank L_0 and the sparse S_0. It is guaranteed to work even if the rank of 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.Algorithmically, we will see that this problem can be solved by efficient and scalable algorithms, at a cost not so much higher than the classical PCA. Empirically, our simulations and experiments suggest this works under surprisingly broad conditions for many types. of real data.

A. Idea of separation

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?To make the problem meaningful, we need to **impose that the low-rank component L_0 is not sparse**.Using incoherence introduced in Candes and Recht[2009] for the matrix completion problem;it is an assumption concerning the singular vectors of the low rank component.SVD of $L_0 \in R^{n_1 * n_2}$ as

$$L_0 = \text{U}\Sigma V^*$$

where r is the rank of the matrix, σ1 ,..., σr are the positive singular values, and U = [u1 ,...,ur], V = [v1 ,...,vr] are the

matrices of left-singular and right-singular vectors.
Another identifiability issue arises if the sparse matrix has low-rank. This will occur if, say, all the nonzero entries of S occur in a column or in a few columns. Suppose for instance, that the first column of S_0 is the opposite of that of L_0 , and that all the other columns of S_0 vanish. Then it is clear that we would not be able to recover L_0 and S_0 by any method whatsoever since M = L_0 + S_0 would have a column space equal to, or included in that of L_0 . To avoid such meaningless situations, we will assume that the **sparsity pattern of the sparse component is selected uniformly at random.(Assumption 2)**

III. Main Result

Under these minimal assumptions, the simple PCP solution perfectly recovers the low-rank and the sparse components, provided the rank of the **low-rank component is not too large**, and that the **sparse component is reasonably sparse**.Throughout this paper, n(1) = max(n1,n2) and n(2) = min(n1,n2)

A. Theorem 1.1

Suppose L_0 is n x n, obeys (1.2)(1.3).Fix any n 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($[S_0]\_ij$) = Σ_ij for all (i,j) ıΩ. Then, there is a numerical constant c such that with probability at least $1cn^{10}$ (over the choice of support of S_0),Principal Component Pursuit (1.1) with λ = $1/[\sqrt{n}]$is exact, that is, L = L_0 and S = S_0 , provided that

$$\text{rank}(L_0) \leq \rho \text{n} \mu^{-1}(logn)^{-2}$$

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 (as long as these are randomly distributed). In fact, this works for large values of the rank, that is, on the order of $n/(logn)^2$ when is not too large. We emphasize that the only **"piece of randomness in our assumptions concerns the locations of the nonzero entries of** S_0 ; everything else **is deterministic**. In particular, all we require about L_0 is that its singular vectors are not spiky. Also, we make no assumption about the magnitudes or signs of the nonzero entries of S_0 . To avoid any ambiguity, our model for S_0 is this: take an arbitrary matrix S and set to zero its entries on the random set Ω^c; this gives S_0 .
A rather remarkable fact is that there is no tuning parameter in our algorithm. This is surprising because one might have expected that one would have to choose the right scalar λ to balance the two terms in $\|$ L $\|_*$ + λ $\|$ S $\|_1$ appropriately.The choice λ = 1/$\sqrt{n_(1)}$ is universal. Further, it is not a priori very clear why λ = 1/$\sqrt{n_(1)}$ is a correct choice no matter what L_0 and S_0 are. It is the mathematical analysis which reveals the correctness of this value. In fact, the proof of the theorem gives a whole range of correct values, and we have selected a sufficiently simple value in that range. One can obtain results with larger probabilities of success at the expense of reducing the value of ρ_r