EN.600.692 Advanced Topics in Machine Learning
Final Project

Evaluation of Low Rank Subspace Clustering (LRSC) for Uncorrupted Data, Noisy Data, and Corrupted Data

Michelle Chyn, Barbara Kim

Introduction

We wanted to compare the efficiency of the Low Rank Subspace Clustering (LRSC) algorithms across a data set altered to have different errors via noise or corruption. We thought this would be an interesting project since we covered neither the non-convex LRSC methods nor the convex LRSC method for corruption in our previous assignments. Each of the algorithms have varying computational costs, which motivates the question of when it is worth it to use more complex algorithms.

Here we outline some of the differences between the optimization problems defined in each method. LRSC with uncorrupted data just seeks to find a low rank representation of the data that is subspace preserving when subspaces are independent in order to compute subspace clusters without having to use the entire data matrices. In LRSC with Noise, the non-convex problem models the data matrix $X = A + E$, where E contains all of the errors and A is the ideal uncorrupted data. The convex problem is a special case of this model where $A = XC$, which makes the model a linear problem, which can be solved using convex optimization. This assumption implies that we do not directly model the noise, which could lead to poorer performance in the case of data containing large amounts of noise. LRSC with Noise penalizes error using the Frobenius norm. In order to make the model robust against corruptions, the l1 norm is used when the error is sparse and not whole columns.

A cursory analysis of each algorithm shows that the non-convex LRSC with noisy data algorithm needs an additional singular value decomposition (SVD) compared to regular LRSC and even convex LRSC with noisy data. LRSC robust against corruptions adds another level of computation with the use of alternating minimization of augmented Lagrangians. Therefore, we seek to determine the levels at which the simpler algorithm performs best in data sets containing increasing levels of noise, leading to where the noise level is so high that those entries are considered corrupted.

Methods

Data

The top left 50 x 50 pixels of each image in the Yale B data set were used, because our computers could not compute the singular value decomposition of the entire data set. We used 10 images for each of 3 individuals from the Yale B data set.

We created 28 data sets to test on each algorithm with the unaltered original data set as a control. 24 of the sets contained additions of uniformly distributed noise (10%-60% noise added) in differing percentages of pixels. The remaining 4 sets were altered such that selected pixels were corrupted uniformly at random, where a corrupted entry is defined as an entry whose added error could possibly vary from 0 to 255. Within those each of those two categories, 20%, 40%, 60%, 80% of pixels were chosen randomly for modifications (noise or corruption).

Low Rank Subspace Clustering

We evaluated 5 different methods for our project: LRSC for uncorrupted data, convex LRSC for noisy data, non-convex LRSC for noisy data, convex LRSC for corrupted data (i.e. large errors), non-convex LRSC for corrupted data. We used a specific $\boldsymbol{\tau}$ that led to best fitting of the model for each LRSC method by running LRSC over a range of $\boldsymbol{\tau}$ from 10^{-7} to 100. All LRSC algorithms set q as 1 and $\boldsymbol{\beta}$ as 10.

Evaluation

The clustering for each method on each data set was calculated. Additionally, we visualized the images produced from the matrix of recovered data and compared it to the original data.

Results

Table 1: Range of τ used in Low Rank Subspace Clustering (LRSC) Algorithms

	Uncorrupt	Noisy, Convex	Noisy, Non-Convex	Corrupt, Convex	Corrupt, Non-Convex
Tau	N/A	10^{-6}	10^{-6} - 10^{-7}	-	0.0001 - 150

Table 2: Clustering error (%) of different LRSC algorithms on Uncorrupted, Noisy, and Corrupted Data Sets

	Uncorrupt	Noisy, Convex	Noisy, Non-Convex	Corrupt, Convex	Corrupt, Non-Convex
Uncorrupted Data	30	0	0	-	30
0.1 Noise, 20% Corruption	30	0	0	-	30
0.1 Noise, 40% Corruption	30	0	0	-	30
0.1 Noise, 60% Corruption	30	0	0	-	33
0.1 Noise, 80% Corruption	30	0	0	-	30
0.6 Noise, 20% Corruption	27	10	0	-	27
0.6 Noise, 40% Corruption	27	13	3	-	23
0.6 Noise, 60% Corruption	23	13	0	-	23
0.6 Noise, 80% Corruption	27	0	0	-	0

Table 3: Average computing time (seconds) of the different LRSC algorithms on the 3 subjects from the Yale B Data Set

	Uncorrupt	Noisy, Convex	Noisy, Non-Convex	Corrupt, Convex	Corrupt, Non-Convex
Computing Time (s)	0.0768	0.0796	4.5017	-	18.4892

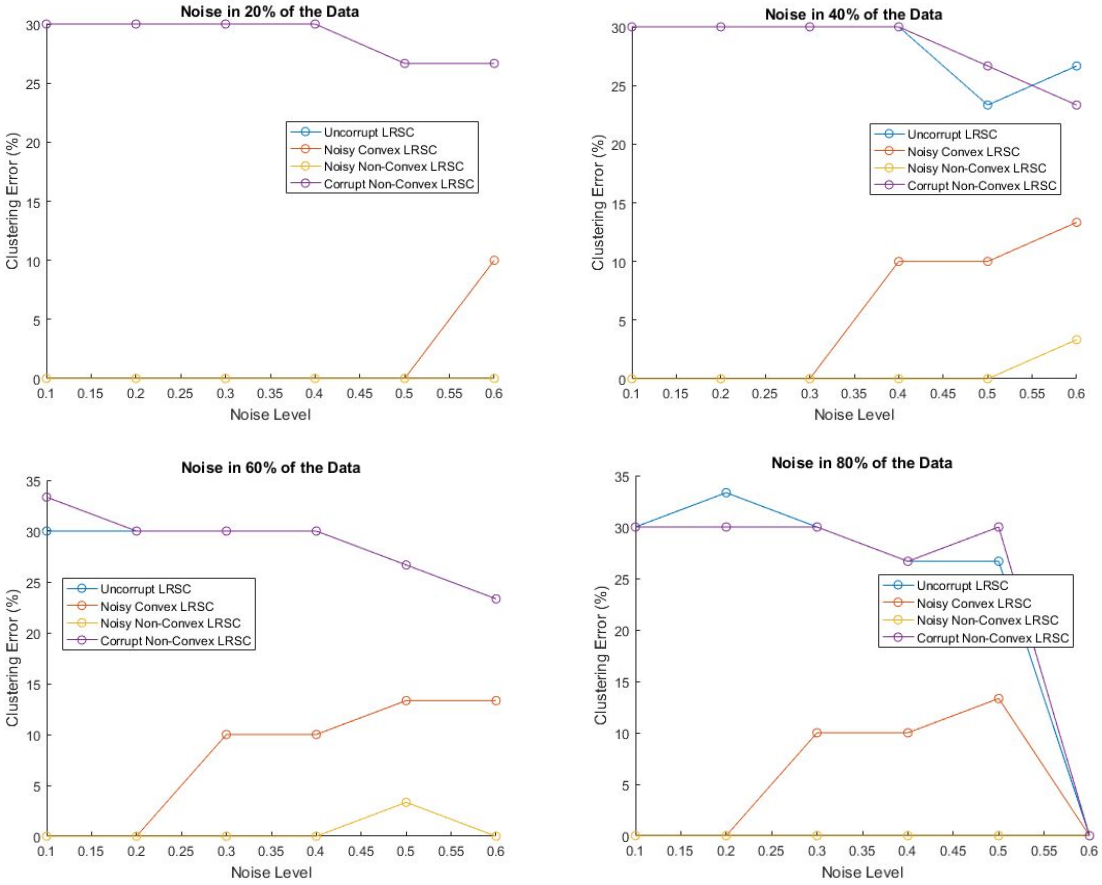
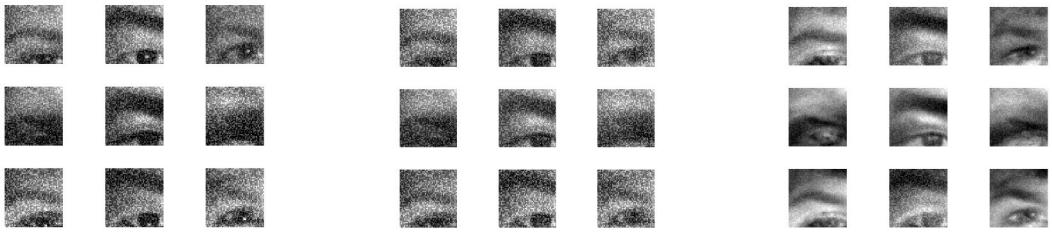


Fig. 1 Clustering errors of different variations of the Low Rank Subspace Clustering Algorithm as a function of the Noise Level, with noise making up 20% to 80% of the Yale B dataset.



a. b. c.

Fig. 2 a) First 3 images for 3 individuals, 10% noise, 80% corrupted. b) Images reconstructed from C matrix from noisy convex LRSC method. c) Images reconstructed from C matrix from noisy non-convex LRSC method

As noise level increases, the noisy convex LRSC algorithm produces increasing errors in clustering, but the error is still lower overall by between 20-30% compared to LRSC for uncorrupted data. As more of the data is altered by noise, the noisy convex LRSC algorithm results in around 10% error. The noisy non-convex LRSC algorithm performs very well, with close to 0% error regardless of percent noise or noise level. The reconstructed images via the noisy non-convex LRSC algorithm recovered more of the original image than the reconstructed images via the noisy convex LRSC algorithm, as seen on Fig. 2. However, the noisy non-convex LRSC takes around 4.5 seconds to run once on the Yale B dataset, which is an order of magnitude higher (~60x slower) than the computing time of the noisy convex LRSC algorithm. However, the corrupt non-convex LRSC algorithm takes about 4 times as long to run than the noisy non-convex LRSC method.

Discussion

Non-convex LRSC method for noisy data and convex LRSC method for noisy data seem to perform the best out of all the algorithms tested. Non-convex LRSC for corrupted data performed no better than the LRSC for uncorrupted data. We reason that this may be due to our implementation. One, due to the large computational power that running the algorithms on a full image dataset would require, we had to reduce our images to 50x50. However, this may have led to poorer clustering results. Second, non-convex algorithms may have required more iterations than we processed, as we capped off the iterations at 1000 or 2000.

Citations

1. Liu, Guangcan, et al. "Robust recovery of subspace structures by low-rank representation." *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 35.1 (2013): 171-184.
2. Vidal, Rene, Yi Ma, and Shankar Sastry. "Generalized principal component analysis (GPCA)." *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 27.12 (2005): 1945-1959.