Identifying Salient Features in MR images using Outlier Pursuit

Abhinav V. Sambasivan

December 20, 2015

Abstract

This project focuses on identifying salient features/outliers in MR images using Outlier Pursuit. We consider a model where the matrix of interest can be decomposed as a sum of low rank, and column sparse (with only a few non-zero columns) matrices. Usually, the outliers in such large matrices are identified by solving a (linear constrained) convex optimization program. Here, we propose an ADMM based algorithm to solve this nonsmooth convex problem. The proposed algorithm converges linearly to the optimal solution set. Existing tools and results in the field of Robust PCA are used to show that the obtained solution, recovers the outlier locations and the low-dimensional subspace exactly. We also briefly describe how MR images fit the modeled considered here, and show experimental results for identifying the salient features (lesion) in brain MR images using our algorithm.

1 Introduction

In this age of *Big-Data*, there have been incessant efforts on processing data efficiently. Identifying low dimensional structure from such high dimensional data sets has been the primary focus in statistical inference settings. We consider here, the problem of matrix *outlier identification*. Let $X \in \mathbb{R}^{n_1 \times n_2}$ be the data matrix of interest, and we know that it can be decomposed as

$$X = L_0 + C_0, \tag{1.1}$$

where L_0 is a low rank matrix and C_0 is column sparse, i.e. only a fraction of the columns in C_0 is non-zeros. Also, the rank of L_0 and the fraction of non-zero columns in C_0 are a priori unknown. Our focus here is to *accurately* identify the location of the non-zero columns of C_0, and also the column space of the low rank matrix L_0. This investigation of ours falls within the realm of Robust Principal Component Analysis (see, e.g., [1–4]). More recently, [6] gave results on outlier identification via compressive sampling techniques.

Principal Component Analysis (PCA), is the most widely used method for dimensionality reduction in statistical analysis. The vanilla PCA problem entails identifying the best low-dimensional approximation (in terms of least square error) for data points sampled from a much higher dimensional space. Singular Value Decomposition (SVD) when applied to the matrix formed by considering each data sample as a column, can be used to find the column space of the low rank approximation. However, it is well known that PCA is highly sensitive to the presence of *outliers*.

In such settings (where it is possible to have spurious or malicious data points), a natural question that follows is whether we can recover the low dimensional subspace of the "uncorrupted" points and also the identities of outliers.

1.1 Motivation

Several image processing and computer vision applications (see e.g., [7, 8]) aim to identify the "saliency map" of a given image, which ideally indicates the regions of the image that tend to attract the attention of a human viewer. The idea behind this project is to use the framework established in the Robust PCA and saliency map estimation literature to identify tumors and/or clots (which we shall henceforth refer to as "salient features") in an MR image.

Here, we interpret the MR image as a collection of distinct non-overlapping patches, so that we can represent the image equivalently as a matrix whose columns are vectorized versions of the patches. It has been demonstrated in [9] that such local patches extracted from images can be well approximated as sample points lying in a union of low dimensional linear subspaces. The approach we use is based on the assumption that the salient regions in the MR image (say anomalous growths, tumors, etc.,) may be modeled as outliers from a single common low dimensional subspace. Since the number of such outliers will (in general) be small, MR images fit into the model (1.1).

Given an MR image, we equivalently represent the image as a matrix, X (of size $n_1 \times n_2$), which is a collection of *vectorized* patches (as described above). Thus each patch in the original image is mapped uniquely to a column of X. The location of outliers in X, once identified (by the column support, as will be discussed in the next section), can then be mapped onto the corresponding patches, thus giving us the location of the salient features. The interpretation of MR images as described above, is pictorially shown in Figure 1.

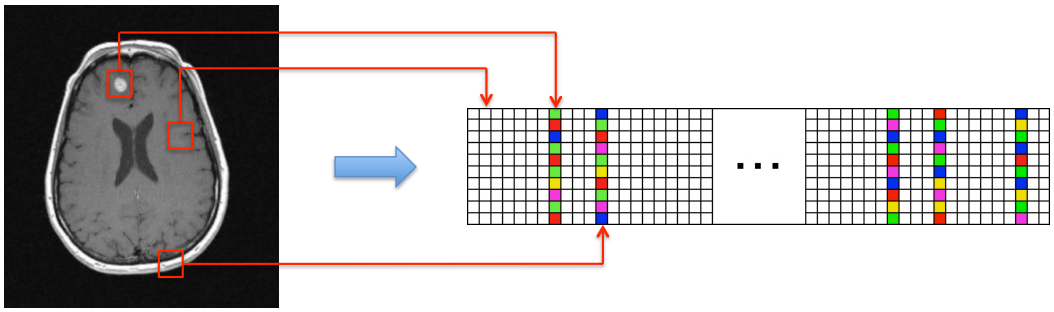

Figure 1: Vizualizing the representation of the MR image using the Matrix X. Patches in the image become columns of X. (a) Original MR image. (b) The $n_1 \times n_2$ matrix, X.

The salient feature patches (like the tumor and also the skull) are represented by the colored columns, and the rest of the tissue patches (which are similar to each other) correspond to vectors lying in the low-dimensional subspace. We shall henceforth concern ourselves only with the matrix X, keeping in mind that locating the outlier columns in X is equivalent to finding the salient image patches.

2 Problem Setup

In this section, we formalize the problem of outlier pursuit and define the optimization problem considered here. Following this, in Section 3 we provide a ADMM based algorithm to optimize the objective function and discuss convergence guarantees for the same. Lastly, in Section 4 we report the results obtained when the proposed algorithm is applied to an actual MR image.

We consider here a model, where the matrix of interest is a sum of low rank matrix, and a column sparse matrix (which corresponds to the outliers), i.e. the matrix $X \in \mathbb{R}^{n_1 \times n_2}$ can be decomposed as,

$$X = L_0 + C_0, \tag{2.2}$$

where L_0 is a low-rank matrix (with $rank(L_0) = r < \min(n_1, n_2)$), and C_0 is column sparse (i.e. only a fraction, $0 < \gamma < 1$, of columns in C_0 are non-zero). Our aim is to recover the column support of the sparse matrix C_0, which gives the location of the outliers; the column support set of C_0 is denoted by $\mathcal{I}_0$, with $|\mathcal{I}_0| = \gamma n_2$

The problem of exactly recovering the low-rank, and the (support of the) column sparse matrices involves solving the following convex optimization problem,

$$\text{(P1)}: \quad \underset{L,C}{\text{argmin}} \quad ||L||_* + \lambda ||C||_{1,2} \quad \text{subject to} \quad L + C = X, \tag{2.3}$$

where $\lambda > 0$ is a regularization parameter, $||\cdot||_*$ is the nuclear norm (sum of singular values), and $||\cdot||_{1,2}$ is the sum of ℓ_2 norm of the columns of the matrix. This problem has been studied by [1–4] who propose (2.3) as a convex surrogate of the following natural recovery formulation for the Robust PCA/Outlier Pursuit problem:

$$\text{(P2)}: \quad \underset{L,C}{\text{argmin}} \quad \text{rank}(L) + \lambda ||C||_{0,2} \quad \text{subject to} \quad L + C = X, \tag{2.4}$$

where $||\cdot||_{0,2}$ denotes the number of non-zero columns in the matrix. It can be easily seen that (2.4) is combinatorial in nature and intractable. Hence we work with its convex relaxation (2.3) to make the optimization problem feasible. Before we go into the specifics of how to solve the problem (P1), let us look at the conditions under which we this recovery problem is feasible (i.e., the solutions are identifiable).

2.1 Identifiability conditions and the notion of *Incoherence*

If there are no additional structural assumptions imposed on the low rank and column sparse components, then the recovery of outlier columns (and also the "true" low dimensional subspace) could be an ill-defined problem. To understand this statement, consider the trivial case when the matrix X has just one non-zero column. Such an X is both low rank and column sparse at the same time, thus making the problem unidentifiable.

To make the problem meaningful (or well-posed), we need to impose additional conditions on the low rank component L_0, to ensure that it is not column sparse as well. This is formalized via what is called the *incoherence condition* which is commonly used in the compressive sensing and matrix completion literature (see e.g., [1, 6]).

Definition 1 (Column Incoherence Property). *Let $L \in \mathbb{R}^{n_1 \times n_2}$ be a rank-r matrix, with at most $n_L \leq n_2$ non-zero columns, and a compact SVD, $L = U\Sigma V^T$. The matrix L is said to satisfy the* ***column incoherence property*** *with parameter μ_L if*

$$\max_i ||V^T e_i||_2^2 \leq \mu_L \frac{r}{n_L}, \tag{2.5}$$

where $\{e_i\}$ are the canonical basis (coordinate vectors) in $\mathbb{R}^{n_2}$.

It should be noted here that $\mu_L \in [1, n_L/r]$. A small incoherence parameter μ_L for the low rank component essentially implies that the column support of L_0 spread out and is hence more desirable. For the purposes of this project, we assume that L_0 is column incoherent with some parameter $\mu_L > 1$. It is worth noting here that, the value of μ_L, the fraction of non-zero columns in C_0 (denoted by γ), and the rank of L_0 (denoted by r), are a priori unknown and are not required for the execution of the algorithm presented here.

3 Outlier Pursuit Algorithm using ADMM

The Robust PCA problem entails efficiently solving a convex optimization problem (P1), and the current methods propose an Alternating Minimization (AltMin) algorithm for the same. The AltMin algorithm has been shown to converge linearly to an optimal point. In contrast, we present here a scalable ADMM based algorithm to solve (P1). The proposed method is also shown to converge linearly to the optimal solution.

3.1 Forming the ADMM problem

Our aim here is to solve the nonsmooth convex program (P1). The problem as described in (2.3) is an equality constrained optimization with a convex objective function (it is well known that both $||\cdot||_*$ and $||\cdot||_{1,2}$ are nonsmooth and convex). Thus (P1) is well within the framework of ADMM. The Augmented Lagrangian for this can be written as,

$$\mathcal{L}_\rho(L, C, \nu) = ||L||_* + \lambda||C||_{1,2} + \langle \nu, L + C - X\rangle + \frac{\rho}{2}||L + C - X||_F^2, \tag{3.6}$$

where $\rho \geq 0$ is a constant. The augmented dual function is given by,

$$d(\nu) = \min_{L,C} \ \mathcal{L}_\rho(L, C, \nu), \tag{3.7}$$

and the dual problem (which is equivalent to the primal problem P1 under mild conditions) is

$$\text{(DP):} \quad \max_\nu \ d(\nu). \tag{3.8}$$

If $\rho > 0$, then $L + C - X$ is constant over the set of minimizers of (3.7) (see [10] for proof). This implies that the gradient of the dual function is well defined and given by,

$$\nabla d(\nu) = L(\nu) + C(\nu) - X,$$

where $(L(\nu), C(\nu))$ is the minimizer of (3.7). Thus we can use a dual ascent strategy to solve the primal problem (P1) using the dual formulation (3.8)

$$\nu := \nu + \alpha(\nu) = \nu + \alpha \left(L(\nu) + C(\nu) - X \right)$$

where α is a suitable step size. The update steps involved in the ADMM algorithm is given in the box below. Executing these 3 steps recursively will, solve the Primal problem (P1).

General Steps involved in ADMM

Step 1: Update L_{k+1} - Subproblem (SP1)

$$\begin{aligned} L_{k+1} &= \underset{L}{\text{argmin}} \ \mathcal{L}_\rho(L, C_k, \nu_k) \\ &= \underset{L}{\text{argmin}} \left\{ ||L||_* + \langle \nu_k, L + C_k - X \rangle + \frac{\rho}{2} ||L + C_k - X||_F^2 \right\}. \end{aligned} \tag{3.9}$$

Step 2: Update C_{k+1} - Subproblem (SP2)

$$\begin{aligned} C_{k+1} &= \underset{C}{\text{argmin}} \ \mathcal{L}_\rho(L_{k+1}, C, \nu_k) \\ &= \underset{C}{\text{argmin}} \left\{ \lambda ||C||_{1,2} + \langle \nu_k, L_{k+1} + C - X \rangle + \frac{\rho}{2} ||L_{k+1} + C - X||_F^2 \right\}. \end{aligned} \tag{3.10}$$

Step 3: Update the Multiplier ν_{k+1} (Dual variable)

$$\nu_{k+1} = \nu_k + \rho \left(L_{k+1} + C_{k+1} - X \right), \tag{3.11}$$

where we fix the step size for the ascent step to be ρ.

3.2 Solving the Subproblems involved in ADMM

The most important steps involved the ADMM strategy is to solve the subproblem (SP1) given by (3.9), and subproblem (SP2) given by (3.10) to global minimum. Let us further look into how these subproblems are solved.

3.2.1 Subproblem (SP1)

The update step for L given by (3.9) involves solving the following nonsmooth convex optimization problem:

$$\widehat{L} = \underset{L}{\text{argmin}} \left\{ ||L||_* + \langle \nu, L + C - X \rangle + \frac{\rho}{2} ||L + C - X||_F^2 \right\}.$$

After completing the squares and ignoring the terms involving just C or ν (since they won't affect the minimization problem here), we get,

$$\widehat{L} = \underset{L}{\operatorname{argmin}} \left\{ ||L||_* + \frac{\rho}{2} \left\| L + C - X + \frac{\nu}{\rho} \right\|_F^2 \right\}. \tag{3.12}$$

The objective function of (3.12) is a sum of smooth and nonsmooth functions of L, (both of which are convex). A closer look at the objective function helps us identify that the solution of (3.12) is the proximity operator associated with the nuclear norm function. There happens to be a closed form solution to this problem and is studied in [11]. Before we obtain the closed form solution, let us first define some preliminary operations.

Consider a matrix $Z \in \mathbb{R}^{n_1 \times n_2}$ of rank r, with an SVD, $Z = U\Sigma V^T$, where $\Sigma = \operatorname{diag}(\{\sigma_i\}_{1 \le i \le r})$. For each $\tau \ge 0$, let us introduce the *singular value shrinkage* operator $\mathfrak{D}_\tau$ defined as

$$\mathfrak{D}_\tau(Z) := U\mathfrak{D}_\tau(\Sigma)V^T, \quad \mathfrak{D}_\tau(\Sigma) = \operatorname{diag}(\{\sigma_i - \tau\}_+), \tag{3.13}$$

where the t_+ is the positive part of t, given by, $t_+ = \max(0, t)$. It should be noted that $\mathfrak{D}_\tau$ applies a soft-thresholding rule to the singular values of the matrix. It is intuitive from the definition (3.13) that the *singular value shrinkage* operator promotes "low rank" property in some sense (i.e., if few singular values are smaller τ, then it makes them zero, thus reducing the rank of the resultant matrix).

We now state without proving a result from [11], which we shall use to solve the subproblem (SP1).

Result (from Theorem 2.1 in [11]). *For each $\tau \ge 0$ and $M \in \mathbb{R}^{n_1 \times n_2}$, the singular value shrinkage operator (3.13) obeys*

$$\mathfrak{D}_\tau(M) = \underset{Z}{\operatorname{argmin}} \left\{ \frac{1}{2}||Z - M||_F^2 + \tau||Z||_* \right\}. \tag{3.14}$$

From (3.14) it follows directly that the minimizer of (3.12) is given by,

$$\widehat{L} = \mathfrak{D}_{1/\rho}\left(X - C - \frac{\nu}{\rho} \right). \tag{3.15}$$

3.2.2 Subproblem (SP2)

The solution to (SP2) is similar in flavor to what was shown for the previous subproblem. The update step for C given in (3.10) involves solving the following nonsmooth convex optimization problem:

$$\widehat{C} = \underset{C}{\operatorname{argmin}} \left\{ \lambda||C||_{1,2} + \langle \nu, L + C - X \rangle + \frac{\rho}{2}||L + C - X||_F^2 \right\}.$$

After completing the squares and ignoring the terms involving just L or ν (since they won't affect the minimization problem here), we get,

$$\widehat{C} = \underset{C}{\operatorname{argmin}} \left\{ \underbrace{\frac{\lambda}{\rho}||C||_{1,2} + \frac{1}{2} \left\| L + C - X + \frac{\nu}{\rho} \right\|_F^2}_{=h_0(C)} \right\}. \tag{3.16}$$

The objective function of (3.16), as in the previous case, is a sum of smooth and nonsmooth functions of C, (both of which are convex). A closer look at the objective function of (3.16) helps us identify that its solution is the proximity operator associated with the group norm $||\cdot||_{1,2}$.

Since the function $h_0(C)$ is strictly convex, it has a unique minimizer (say $\widehat{C}$). Now, the first order optimality conditions imply that, $\widehat{C}$ minimizes h_0 if and only if $\mathbf{0}$ is a subgradient of h_0 at the point $C = \widehat{C}$, i.e.

$$\mathbf{0} \in C - \left(X - L - \frac{\nu}{\rho}\right) + \epsilon \cdot \partial||\widehat{C}||_{1,2} \ , \tag{3.17}$$

where $\epsilon = \lambda/\rho$, and $\partial||\widehat{C}||_{1,2}$ is the subdifferential of $||\cdot||_{1,2}$ at $C = \widehat{C}$. By definition, we have $||C||_{1,2} = \sum_i ||C^{(i)}||_2$ (where $C^{(i)}$ is the i-th column of C), and (3.17) is hence separable (decouples) column wise. So, the optimality condition (3.17) can be rewritten as,

$$\mathbf{0} \in C^{(i)} - \left(X^{(i)} - L^{(i)} - \frac{\nu^{(i)}}{\rho}\right) + \epsilon \cdot \left(\partial||\widehat{C}||_{1,2}\right)^{(i)}, \quad \forall i = 1, 2, \ldots, n_2.$$

In other words, for some subgradient vector $g^{(i)} \in \left(\partial||\widehat{C}||_{1,2}\right)^{(i)}$, we have

$$\mathbf{0} = \widehat{C}^{(i)} - G_C^{(i)} + \epsilon \cdot g^{(i)}, \quad \forall i = 1, 2, \ldots, n_2,$$

where we have defined $G_C^{(i)} = \left(X^{(i)} - L^{(i)} - \frac{\nu^{(i)}}{\rho}\right)$. We can rewrite the above equation as,

$$\widehat{C}^{(i)} = G_C^{(i)} - \epsilon \cdot g^{(i)}, \quad \forall i = 1, 2, \ldots, n_2. \tag{3.18}$$

Let us now define the *column wise shrinkage operator*, $\mathfrak{C}_\epsilon$ as,

$$\mathfrak{C}_\epsilon(G_C) = \left(1 - \frac{\epsilon}{||G_C^{(i)}||_2}\right)_+ G_C^{(i)} \quad \forall i = 1, 2, \ldots, n_2. \tag{3.19}$$

First if we note that value of the subgradient vector $g^{(i)} = \widehat{C}^{(i)}/||\widehat{C}^{(i)}||_2$ whenever $\widehat{C}^{(i)} \neq 0$ (by definition), and it can take any value in the unit ℓ_2-norm ball when $\widehat{C}^{(i)} = 0$. Using the above fact it is easy to show that, the *column wise shrinkage operator* defined in (3.20) satisfies the optimality condition (3.18). Hence we can write the optimal solution for the subproblem (SP2) as

$$\widehat{C} = \mathfrak{C}_{\lambda/\rho}\left(X - L - \frac{\nu}{\rho}\right). \tag{3.20}$$

The overall ADMM algorithm for the Outlier pursuit problem is summarized and provided below. Algorithm 1 was implemented in Matlab and the results are summarized in the following section.

4 Results and Discussion

The Algorithm 1 was first tested for synthetic data (generated by a Matlab script). This was also used as test bench to find the optimal values of the regularization parameter λ. Once the

Algorithm 1 Outlier-Pursuit Algorithm using ADMM

Input: X

Initialize: $L_0 = \nu_0 = \mathbf{0}^{n_1 \times n_2}, C_0 = X - L_0, \lambda = 0.71, \rho = 1$

while not converged **do**

(1) Update L_{k+1}:

$G_L^k \leftarrow X - C_k - \frac{1}{\rho}\nu_k$

$L_{k+1} \leftarrow \mathfrak{D}_{1/\rho}(G_L^k)$

(2) Update C_{k+1}:

$G_C^k \leftarrow X - L_{k+1} - \frac{1}{\rho}\nu_k$

$C_{k+1} \leftarrow \mathfrak{C}_{\lambda/\rho}(G_C^k)$

(3) Update ν_{k+1}:

$\nu_{k+1} \leftarrow \nu_k + \rho(L_{k+1} + C_{k+1} - X)$

$k \leftarrow k + 1$

end while

Output: $\widehat{L} := L_k$, and $\widehat{C} := C_k$

regularization parameter λ was tuned, test runs were performed with randomly generated test matrices X, and the objective function value and runtimes were observed.

Algorithm 1 converged within 40 iterations for matrices as big as 100×1000 with rank, $r = 5$, and the fraction of non-zero columns $\gamma = 0.01$. The total run time of the algorithm was less than 2 seconds on a desktop computer. Also the algorithm was able to recover the low dimensional subspace and the *exact* column support (almost) every time.

It was also observed that the number of iterations required for convergence seemed to be unaffected (in the order of magnitude sense) to the problem dimension. However the time taken per iteration seemed to heavily affected by the problem size. It was also interesting to note that the number of rows in X seemed to have lesser effect on the runtime when compared to the column size n_2. This seems to be intuitive as we run soft-thresholding operations column wise (for the C-update), and also find the compact SVD (for the L-update) at every iteration.

While discussing the convergence of the algorithm, it is necessary to elaborate a bit on the **stopping criteria** used (note that an explicit stoppping criterion is not specified in Algorithm 1 for the sake of brevity). Both objective function values and primal feasibility condition were taken into consideration to specify the stopping criteria. In particular, the algorithm keeps track of the following two things at every iteration, and the loop terminates when both the conditions (given below) are satisfied.

Condition 1: The change in the objective function value - The change in the value of the Augmented Lagrangian function between successive iterations is small. More precisely,

$$\left| \mathcal{L}_\rho^{(k+1)}(L, C, \nu) - \mathcal{L}_\rho^{(k)}(L, C, \nu) \right| < \delta_1 \ (= 0.1).$$

Condition 2: Primal feasibility - When the value of the normalized (per-element) error corre-

sponding to the primal constraint ($L + C = X$) is very small. More precisely,

$$\frac{\|L_{k+1} + C_{k+1} - X\|_F^2}{n_1 n_2} < \delta_2 \ (= 10^{-5})$$

As a performance benchmark, we implemented the AltMin algorithm presented in [1], and tested the outlier pursuit problem using synthetic data on both AltMin and Algorithm 1. With the stopping criteria as defined above, and initializing with a feasible point (refer Algorithm 1), it was observed that our ADMM based algorithm achieves much better (faster) primal feasibility than the AltMin algorithm. It was also observed that the AltMin algorithm converged faster (about 3x faster than ours).

That being said, we can say that the difference in convergence rate is not orders of magnitude apart. The authors of [1] have shown that AltMin converges linearly to the optimal solution. We have previously seen in Section 3 that our algorithm converges to an optimal point (it is also well known that ADMM with a convex objective converges to the optimal solution set). Now we seek the results established in [10], where the authors prove linear convergence for the sum of convex separable functions with linear constraints. It is easy to observe that our problem clearly falls in the above mentioned realm. Thus we can safely claim here that the algorithm presented here also achieve linear convergence. The convergence analysis involved is out of the scope of this project and is hence omitted.

Having discussed that qualitative and quantitative aspects of our algorithm, and its performance on synthetic data, let see how it performed on an actual MR test image. We tested the Algorithm 1 using brain MR image from [12]. The Axial T1 contrast-enhanced MRI of the brain demonstrates a 1-cm right frontal lesion (tumor).

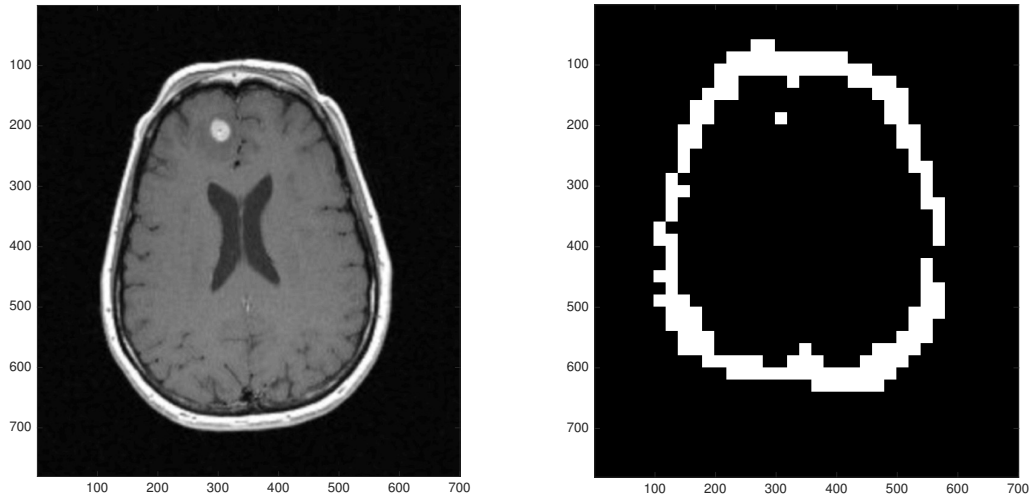

Figure 2: (Left) Given MR image with a 1-cm right frontal lesion, (Right) The outliers as recovered by Algorithm 1. The white patches correspond to the outliers. The lesion was correctly identified as an outlier by our algorithm

Figure 2 shows that our algorithm correctly identifies the location of the anomaly in the brain MRI. This means that the MR image fits the model proposed by us. Apart from the lesion, the

algorithm also identifies the skull periphery as outliers (which is still acceptable and expected). A plot showing the objective function values, and the amount of deviation from the primal feasibility condition as a function of iterations is shown in Figure 3.

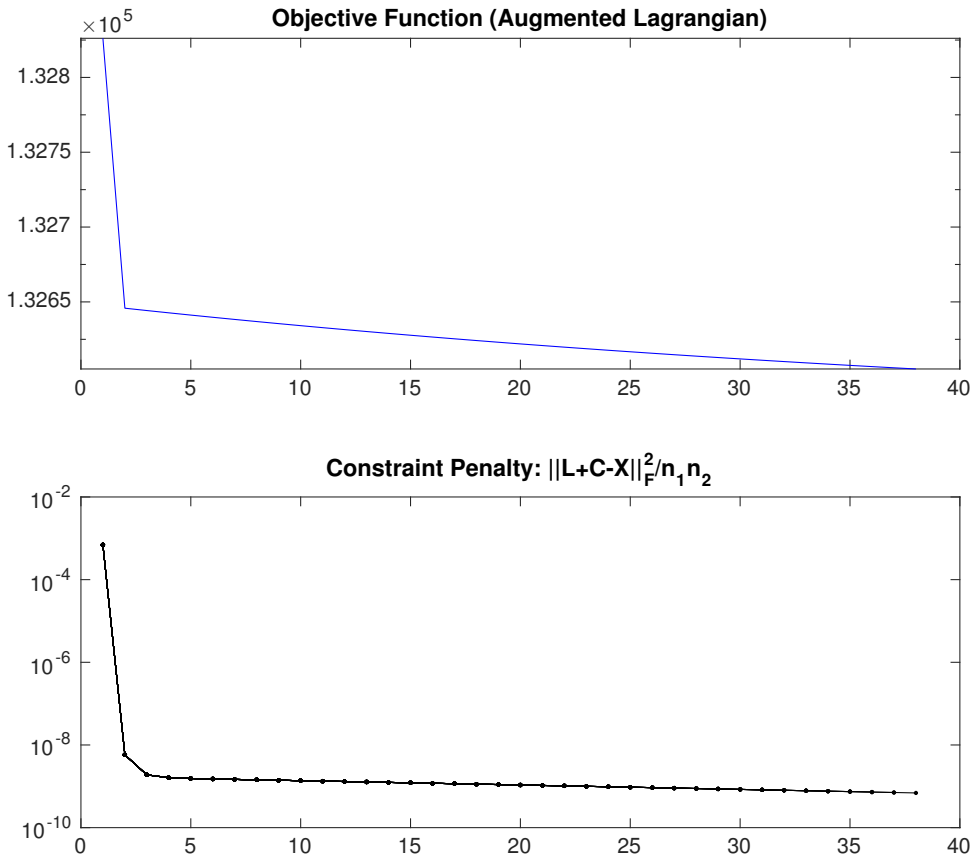

Figure 3: (Top) Augmented Lagrangian: $\mathcal{L}_\rho^{(k)}(L, C, \nu)$, (Bottom) Normalized (per-element) error corresponding to the primal constraint: $\frac{\|L_k+C_k-X\|_F^2}{n_1 n_2}$. The X-axis shows the number of iterations: k, and it is worth noting that these are semilog plots (Y-axis is given in logarithmic scale)

.

For a 794×719 input image, and patch size of 20×20 (in other words $n_1 = 400$ and $n_2 = 1365$, we can see that the algorithm has converged in 38 iterations, taking about 10 seconds to run.

5 Conclusion

The ADMM based algorithm proposed here, has been able to identify salient features in MR images as shown. Also the algorithm has linear convergence and is hence on par with the state-of-the-art. One big challenge in this project was to tune the regularization parameter λ, especially for actual MR images. With availability of large data sets and more sophisticated software tools, it might be feasible to build robust systems which accurately identify anomalies in MR images. Such systems will be of immense utility in clinical diagnosis.

Also, it was observed in the course of the project that, the algorithm in its current state might not be able to locate tumors/lesions in all types of MR images. In general MR images present themselves with varied contrasts, and a wide variety of settings, which significantly affect the image quality and texture. In those cases, the simple model presented here will not be sufficient. A more generic, union of subspaces model can be used to learn a dictionary, which can in turn be analyzed in settings similar to that established here. Such investigations could be prove to be interesting future directions in this field.

References

[1] H. Xu, C. Caramanis, and S. Sanghavi. "Robust PCA via outlier pursuit." Advances in Neural Information Processing Systems. 2010.

[2] E. J. Candès, et al. "Robust principal component analysis?." Journal of the ACM (JACM) 58.3 (2011): 11.

[3] M. Soltanolkotabi, E. Elhamifar, and E. J. Candès. "Robust subspace clustering." The Annals of Statistics 42.2 (2014): 669-699.

[4] J. Wright, and A. Ganesh, and K. Min, and Y. Ma. "Compressive principal component pursuit." Information and Inference 2.1 (2013): 32-68.

[5] E. J. Candès, and B. Recht. "Exact matrix completion via convex optimization." Foundations of Computational mathematics 9.6 (2009): 717-772.

[6] X. Li, and J. Haupt. "Identifying Outliers in Large Matrices via Randomized Compressive Sampling." IEEE Transactions on Signal Processing, vol. 63, no. 7, pp. 1792-1807, April 2015

[7] T. Liu, and J. Sun, and N. Zheng, and X. Tang, and H. Shum. "Learning to detect a salient object, in Proceedings IEEE Conference on Computer Vision and Pattern Recognition, 2007.

[8] N. Rao, and J. Harrison, and T. Karrels, and R. Nowak, and T. T. Rogers. "Using machines to improve human saliency detection, in Proceedings Asilomar Conference on Signals, Systems and Computers, 2010, pp. 8084.

[9] G. Yu, and G. Sapiro. "Statistical compressed sensing of Gaussian mixture models, IEEE Transactions on Signal Processing, vol. 59, no. 12, pp. 58425858, 2011.

[10] M. Hong, and Z. Q. Luo. "On the Linear Convergence of the Alternating Direction Method of Multipliers," March 2013, arXiv:1208.3922v3

[11] J. F. Cai, and E. J. Candes, and Z. Shen. "A Singular Value Thresholding Algorithm for Matrix Completion," SIAM Journal on Optimization, 20(4), pp.1956-1982, 2010

[12] A. L. Stockham, and S. T. Chao, and J. H. Suh. "Wanted: Dead or alive? Distinguishing radiation necrosis from tumor progression after stereotactic radiosurgery."