Applications of Convex Optimization in Image Processing

Anirudh K. Agrawal (11098)
Department of Electrical Engineering
Indian Institute of Technology
kanirudh@iitk.ac.in

Vivek Kumar (11821)
Department of Electrical Engineering
Indian Institute of Technology
viveks@iitk.ac.in

***Abstract*—We planning to highlight the different areas of image processing where convex optimization finds its uses. Some of the general areas where we apply convex optimization are denoising , deblurring of images, segmentation, 3d reconstruction and face recognition. In the following report we will discuss different areas of Image Processing in relation to convex optimization, namely de-noising and face-recognition. To solve the image denoising problem we discuss an efficient iterative algorithm to solve a certain class of linearly constrained non-convex optimization problems, which is used to implement non-convex norms for regularization tasks in image processing and computer vision and compare different variants of it. We also discuss Augmented Lagrangian Methods (ALM) for fast and scalable ℓ_1-minimization in real-world Robust Face Recognition and also compares several other popular ℓ_1-minimization algorithm in sparse representation based classification. In this report we verify the results obtained by the authors and present our own comments.**

I. Introduction

In many problems related to Image Processing and Computer Vision we find many optimization problems, many of which are of convex nature. We can use the various Convex optimization techniques to solve them.

For solving the Face Recognition problem a sparsity based classification framework has been used which boils down to solving a ℓ_1 minimization problem as which can be written as follows

$$\min_x \quad ||x||_1 \quad \text{subject to} \quad b = Ax \tag{1}$$

But in practical cases b has noise in it, so we solve a unconstrained *Basis pursuit denoising* problem which can be written as follows

$$\min_x \quad \frac{1}{2}||b - Ax||_2 + \lambda||x||_1 \tag{2}$$

Here the matrix 'A' is made for each class, whose column vectors are the images of size(w x h) and 'y' is the test image. We already know that minimizing the ℓ_1 produces a sparse results, this finds applications in image compression also. Although the above problem can be solved as a simple linear program due to high dimensionality of images the time and computational complexity is vary high. Many efficient methods have been developed over the years, we will discuss some of them and compare them based on their computational complexity.

There are also many computer vision problems which falls under linearly constrained convex plus concave optimization problems which are actually non-convex problems but can be efficiently minimized using state-of-the-art algorithms for convex optimization. We will study how to efficiently solve a wide class of optimization problems including ℓ_2-ℓ_p and ℓ_1-ℓ_p problems with $0 < p < 1$ which are non-convex in nature. The model considered is a linearly constrained minimization on a finite dimensional Hilbert space $\mathcal{H}$ of the form

$$\min_{x \in \mathcal{H}} \ F(x), \qquad s.t. \qquad Ax = b, \tag{3}$$

with $F : \mathcal{H} \rightarrow \mathbb{R}$ being a sum of two Lipschitz continuous terms

$$F(x) := F_1(x) + F_2(|x|) \tag{4}$$

We assume that F is bounded below, $F_1 : \mathcal{H} \rightarrow \mathbb{R} \cup \{\infty\}$ is a proper convex and $F_2 : \mathcal{H}_+ \rightarrow \mathbb{R}$ is concave and increasing. Here, $\mathcal{H}_+$ denotes the non-negative orthant of the space $\mathcal{H}$; increasingness and the absolute value $|x|$ are to be understood coordinate-wise. The linear constraint Ax = b is given by a linear operator $A : \mathcal{H} \rightarrow \mathcal{H}_1$. For the study purpose, we take formulation of F_1 and F_2 as

$$F_1(x) = ||Tx - g||_2^2, \qquad and \qquad F_2(|x|) = \lambda||x||_{\varepsilon,p}^p, \tag{5}$$

where, $||x||_{\varepsilon,p}^p, = \sum_i(|x_i|) + \varepsilon)^p$ is a non-convex norm for $0 < p < 1, \lambda \in \mathbb{R}_+$, T is a linear operator, and g is a vector to be approximated. To solve the optimization problem given in (3) the following iterative algorithm is proposed in the paper:

$$x^{k+1} = \underset{Ax=b}{argmin} F^k(x) \tag{6}$$

$$:= \underset{Ax=b}{argmin} F_1(x) + ||w^k.x||_1, \tag{7}$$

where ∂F_2 denotes the superdifferential of the concave function F_2. This iterative algorithm proceeds by iteratively solving ℓ_1 problems which approximate the original problem. We have worked on the implementation of above algorithm in image denoising by Rudin, Osher, and Fatemi (ROF) model and iteratively reweighted ℓ_1 (IRL1) algorithm and compared the results for both.

II. Image Denoising

The general problem (3) is formulated for specific computer vision applications as:

$$\underset{Ax=b}{min}\ F(x) = \underset{Ax=b}{min} F_1(x) + F_2(|x|) \tag{8}$$

$$:= \underset{Ax=b}{min} ||Tx - g||_q^q + \Lambda^T \mathbf{F}_2(|x|), \tag{9}$$

where $\mathbf{F}_2 : \mathcal{H}_+ \to \mathcal{H}_+$ is a coordinate-wise increasing and concave funtion. $A : \mathcal{H} \to \mathcal{H}_\infty, T\mathcal{H} \to \mathcal{H}_1$ are linear operators acting between finite dimensional Hilbert spaces $\mathcal{H}_1$ or $\mathcal{H}_1$. The weight $\Lambda \in \mathcal{H}_+$ has non-negative entries. The *data-term* is the convex ℓ_p-norm with $q \geq 1$. Various prototypes for $\mathbf{F}_2(|x|)$ are

$$|x_i| \mapsto (|x_i| + \varepsilon)^p \quad or \quad |x_i| \mapsto log(1 + \beta|x_i|), \quad \forall i, \tag{10}$$

i.e., the regularized ℓ_1-norm, $0 < p < 1, \varepsilon \in \mathbb{R}_+$, or a convex-log function. The ℓ_p-norm becomes Lipschitz by the ε-regularization and the log-function naturally is Lipschitz.Algorithm (7) simplifies to

$$x^{k+1} = \underset{Ax=b}{argmin} ||Tx - g||_q^q + ||diag(\Lambda)(w^k.x||_1, \tag{11}$$

where the weights given by the superdifferential of F_2 are

$$w_i^k = \frac{p}{(|x_i^k| + \varepsilon)^{1-p}} \quad or \quad w_i^k = \frac{\beta}{(1 + \beta|x_i|)}, \tag{12}$$

Further in the report, we will describe the two main algorithms for Image denoising which are optimized using the above described algorithm and do their comparison.

A. ROF Image Denoising Model

Now, we'll touch upon ROF image model [5] which aims at obtaining a clean image x from a noisy image $f : \Omega \to \mathbb{R}$, where Ω is bounder open subset of $\mathbb{R}^2$. For solving this problem they proposed the following minimization problem:

$$u = \underset{u}{argmin} ||_p + \lambda||f - u||_q^2. \tag{13}$$

Here $F_1(u) = |x|_p$ and $F_2(u) = \lambda||f - u||_q^2$ and subdifferential $\partial F_2 = 2\lambda(u - f)$. Formally, we can write $\partial F_2(u) = -div\left(\frac{\nabla u}{|\nabla u|}\right)$, and we can write the Euler-lagrange differential equation for the RUdin Osher Fatemi model as follows:

$$0 = -div\Big(\frac{\nabla u}{|\nabla u|}\Big) + 2\lambda(u - f) \tag{14}$$

or equivalently,

$$u = f + \frac{1}{2\lambda} div\Big(\frac{\nabla u}{|\nabla u|}\Big) \tag{15}$$

Here, The parameter λ determines the denoising level. if λ is small, then it is equivalent to penalizing the fidelity term more and the minimizer u is very close to original image. On the other hand, for large λ the minimizer is a cartoon representation of the image f.

B. Iterative Reweighted ℓ_1 Algorithm

Paper [6] describes an iterative reweighted ℓ_1 algorithm for non-convex l_p pseudo-norms, $p \in (0, 1)$, which are non-differentiable at zero. It solves a sequence of non-smooth ℓ_1 problems and can be seen as counterpart to IRLS (Iteratively Reweighted Least Squares) algorithm. Originally, it was proposed to improve the sparsity properties in ℓ_1 regularized compressed sensing problems, but this algorithm found its uses in various computer vision algorithms. In the formulation of the IRL1 algorithm, we have minimization problem as explained in equations (10), (11) and (12) as

$$|x_i| \mapsto log(1 + \beta|x_i|), \quad \forall i,$$

$$x^{k+1} = \underset{Ax=b}{argmin} ||Tx - g||_q^q + ||diag(\Lambda)(w^k.x||_1,$$

where the weights given by the superdifferential of F_2 are

$$w_i^k = \frac{p}{(|x_i^k| + \varepsilon)^{1-p}} \quad or \quad w_i^k = \frac{\beta}{(1 + \beta|x_i|)}$$

The results for both the algorithms discussed above are provided in the Section 4 (Results and Simulations).

III. ℓ_1 Minimization Algorithms

Here we are gonna discuss various ℓ_1 minimization techniques.

A. Homotopy Methods

The method [4] uses the observation that while solving (2) the objective function undergoes a homotopy from the ell_2 constraint to the ell_1 objective as λ decreases, i.e when $\lambda \to 0$, x_λ^* converges to the solution of (1). Also we can show that the solution path $\mathcal{X} = \{x_\lambda^* : \lambda \in [0, \infty)\}$ is a piecewise constant function of λ. Therefore if we can the breakpoints that lead to changes in the support of x_λ^* i.e either removal of a previously non-zero coefficients or addition of a new non-zero coefficient, we can construct a decreasing sequence of λ.
In order to computer the gradient of objective function of the problem defined in (2), let us define a subdifferential of $||x||_1$ defined as follows :

$$u(x) = \partial||x||_1 = \begin{cases} u_i = sgn(x_i) & x_i \neq 0 \\ u_i \in [-1, 1] & x_i = 0 \end{cases} \tag{16}$$

Setting initial value $x^{(0)} = 0$ we can iterate w.r.t a non-zero λ, and using the condition $0 \in \partial F(x)$ we derive the following

$$c(x) = A^T b - A^T Ax \in \lambda u(x) \tag{17}$$

A sparse support set $\mathcal{I} = \{i : |c_i^{(k)}(x)| = \lambda\}$ can be generated according to (16) at the k-th iteration. Then we can compute the update directions for non-zero coefficients of x^k. The newton update equations derived involve only the non-zero coefficients of x^k, which will usually be very small number since **x** is sparse. The complexity of Homotopy Algorithms is bounded by $\mathcal{O}(dm^2 + dmn)$ which is much better than Interior Point Methods .

B. Templates for Convex Cone Solvers (TFOCS)

It is a recently developed novel method [3] which can be used to solve the general constrained problem of the type

$$\min_{\mathcal{A}x+b\in\mathcal{K}} \quad f(x)$$

where $\mathcal{A}$ is a linear operator and $\mathcal{K}$ is a convex cone.We can observe that that equation (1) and (2) fall into the above category. TFOCS proposes to use the soft-thersholding operator to solve the dual problem of ℓ_1 minimization.

In this algorithm we wish to use the generalized project gradient descent to obtain the solution of dual problem of (1) and in turn get the primal solution. But if the dual cost function is not smooth the gradient descent cannot be applied. In order to overcome the above problem we use a smoothing term $\phi(x)$ to the primal problem. For the (1) problem we get the following problem

$$\min_{x} \quad ||x||_1 + \mu\phi(x) \quad \text{subject to} \quad \mathbf{b} = A\mathbf{x} \tag{18}$$

here $\mu > 0$ is the smoothing parameter and $\phi(x)$ is strongly convex function. $\phi(x)$ is usually taken as $\phi(x) = \frac{1}{2}||\mathbf{x} - \mathbf{x}_0||_2^2$, x_0 has to be specified. We can form the Lagrangian for as follows

$$\mathcal{L}_\mu(x,\theta) = ||\mathbf{x}||_1 + \frac{\mu}{2}||\mathbf{x} - \mathbf{x}_0||_2^2 - \theta^T(A\mathbf{x} - \mathbf{b}) \tag{19}$$

here θ is the Lagrangian multiplier.

We can define the dual problem for above as $max_\theta \mathcal{L}_\mu(\theta)$, where $\mathcal{L}_\mu(\theta) = min_x \mathcal{L}_\mu(x,\theta)$. We also note that $\mathcal{L}_\mu(\theta)$ is smooth concave function whose gradient can be calculated and is given by

$$\nabla\mathcal{L}_\mu(\theta) = b - A\hat{x}_\mu(\theta)$$

Using *first-order project gradient methods* we can update the primal and dual variables as follows

$$\begin{cases} x_k = soft(x_0 + \frac{1}{\mu}(A^T\theta_k, \frac{1}{\mu}) \\ \theta_{k+1} = \theta_k + t_k(b - Ax_k) \end{cases}$$

where $\{t_k\}$ are the step-sizes satisfying $t_k \leq \frac{\mu}{||A^TA||}$ for all k. Note that for a small μ the solution to () reduces to the solution of (1) . The values of μ and x_0 determine the number of iterations.

C. Augmented Lagrangian Methods (PALM)

We will discuss here a special class of first-order methods called the *Augmented Lagrangian Methods* [2] We know that Lagrangian eliminates the equality constraints by including a penalty term in the cost function for the minimization problem. Let us consider a modified cost function for (1)

$$\min_{x} \quad g(x) + \frac{\xi}{2}||h(x)||_2^2 \quad \text{subj. to} \quad h(x) = 0 \tag{20}$$

here $g(x) = ||x||_1$ and $h(x) = b - Ax$, observe that for g and h are continuous and convex functions. (20) has the same optimal solution as that of (1) for any $\xi > 0$. Although we can use other kinds of penalty functions by the quadratic is preferred because of its smoothness.

The Lagrangian of (20) is given by the following

$$\mathcal{L}_\xi(x,\theta) = g(x) + \frac{\xi}{2}||h(x)||_2^2 + \theta^T h(x) \tag{21}$$

here $\theta \in \mathbb{R}^m$ is a Lagrange multiplier. $\mathcal{L}_\xi(x,\theta)$ is the *Augmented Lagrangian* of (1). It can be shown that there exists $\theta^* \in \mathbb{R}^m$ and $\xi^* \in \mathbb{R}$ such that

$$x^* = \arg\min_x \mathcal{L}_\xi(x,\theta^*) \qquad \forall\xi > \xi^* \tag{22}$$

To find the optimal solution for (1) we can use the method of multipliers for minimizing the augmented Lagrangian function $\mathcal{L}_\xi(x,\theta)$. We get the following iterative scheme for solving (22)

$$x_{k+1} = argmin_x \mathcal{L}_{\xi_k}(x,\theta_k) \tag{23}$$

$$\theta_{k+1} = \theta_k + \xi_k h(x_{k+1}) \tag{24}$$

here $\{\xi_k\}$ is a pre-defined positive sequence. Convergence is guaranteed provided that $\{\theta_k\}$ is a bounded sequence and $\{\xi_k\}$ is sufficiently large after a certain index.

The choice of $\{\xi_k\}$ is problem dependent. Higher the value of ξ_k more difficult it is to minimize $\mathcal{L}_{\xi_k}(x,\theta_k)$. For experiments of synthetic data we let it be that $\xi_k \to \infty$ while for real face data we chose some fixed $\xi_k = \xi$.

D. ALM for dual problems (DALM)

The concepts described above can also be dual problem of (1) which is

$$\underset{y}{b^T y} \quad \text{subj. to} A^T y \in B_1^\infty \tag{25}$$

here $B_1^\infty = \{x \in \mathbb{R}^n \ : \ ||x||_\infty \leq 1$. Writing the augmented Lagrangian for above we get

$$\min_{y,z} - b^T y - x^T(z - A^T y) + \frac{\beta}{2}||z - A^T y||_2^2$$
$$\text{subj. to} \quad z \in B_1^\infty$$

here $\mathbf{x}$ is the Lagrange Multiplier of the dual problem. For ease of computation we adopt an alternation strategy to solve the above problem. In this we iteratively minimize the cost function with respect to on variable while holding the rest constant.

Given (x_k, y_k) the update rule for z is

$$z_{k+1} = \mathcal{P}_{B_1^\infty}(A^T y_k + x_k/\beta) \tag{26}$$

here $\mathcal{P}_{B_1^\infty}$ is the projection operator onto B_1^∞. Given (x_k, z_{k+1}) minimizing w.r.t to y we get a Least Squares problem which can be solved as follows :

$$\beta AA^T\mathbf{y} = \beta A z_{k+1} - (Ax_k - b) \tag{27}$$

and the update equation for $\mathbf{x}$ is

$$x_{k+1} = x_k - \beta(z_{k+1} - A^T y_{k+1} \tag{28}$$

The dual algorithm always converges because all the subproblems are solved exactly.

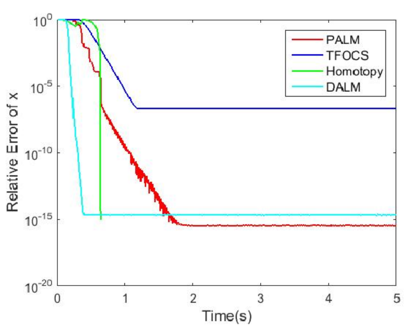
((a)) Original Image

((b)) After adding gaussian noise (PSNR=20.14)

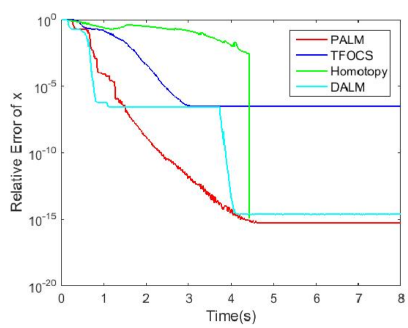
((c)) Denoising using ROF model (PSNR=26.08)

((d)) Denoising using IRL1 (PSNR=24.80)

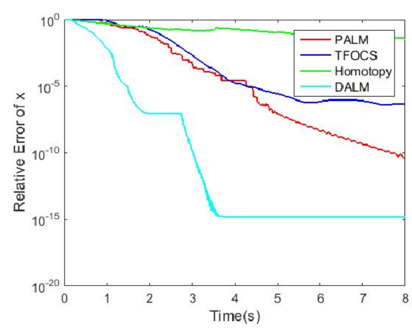
Fig. 1: Image denoising problem. Denoising the image using ROF model (100 iterations, time elapsed:1.13 sec) and IRL1 algorithm (40 iterations,time elapsed:24.57 sec)



((a)) m = 800,n=1000,d=100



((b)) m = 800,n=1000,d=100



((c)) m = 800,n=2000,d=200

Fig. 2: Relative error v/s CPU time

IV. Results and Simulations

A. Image Denoising

Figure 1 shows original image (Figure 1(a)) in consideration and the noisy image by adding a gaussian noise of $\sigma = 25$. The peak signal to noise ration (PSNR) for the noisy image is 20.143045 db.

1) ROF Image Denoising Model: ROF algorithm was run for noisy image for $\lambda = 10$ and 100 iterations (figure 1(c)). The peak SNR of denoised Image by ROF denoising model is: 26.088432 db and the time taken for 100 iterations is 24.57 seconds. On increasing the number of iterations the PSNR value increases and saturates after around 2000 iterations.

2) Iterative Reweighted ℓ_1 Algorithm: The output of denoised image by IRL1 algorithm is shown in figure 1(d). Peak SNR of denoised image by IRL1 algorithm found out to be 24.80 db after 40 iterations which is less than as compared to by ROF model. Also the time taken IRL1 is much higer (24.57 seconds).

B. ℓ_1 Minimization algorithm comparison

To compare the various ℓ_1 minimization algorithms we plot solve the (2) problem for each of them and plot the relative error $r_k(x) = ||x - x_0||_2/||x_0||_2$ w.r.t to number of iterations. We have tested the algorithms by generating and sparse signal and under determined linear system based on Gaussian Distribution. The matrix A of size m x n (m ¡ n) is generated by setting each entry a i.i.d gaussian. We have normalized in column to have unit ℓ_2 norm. The support of x_0 is chosen based on a uniform distribution in the interval [-10,10]. The support of x_0 is also chosen randomly with $||x_0||_0 = d$ i.e there are d non-zero entries in x_0. We plot the the relative error for different values of d and n keeping $m = 800$ fixed as done in the paper [2].From Fig. 2 we can see that DALM has the best performance among all other methods discussed above.

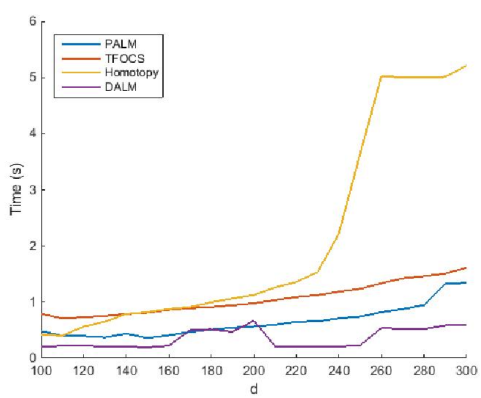


Fig. 3: Variation of Time Taken v/s d

In the above Fig. 3 we calculate the time is takes for $r_k(x) = ||x - x_0||_2/||x_0||_2 < 10^{-5}$ and plot that against the size of support of x. As can be seen from Fig. 3 we can see that we increase the size of support of x , Homotopy performance degrades strongly. While DALM,PALM,TFOCS have comparable performances. From the these two we can say DALM works best among

V. Conclusion

We studied an efficient iterative algorithm to solve certain class of linearly constrain non-convex problems and used it to implement two image denoising algorithms ROF image denoising and iterative reweighted ℓ_1 algorithm and compared the results for both of them.

We presented various el_1 minimzation approaches that can be used in face recognition. From our simulation results we found that DALM has the best performance. WE can observe from Fig. 2 c) that in terms on time complexity and scaling in the size of support of x. Though , we still have to test the above algorithms on real face data.

References

[1] Ochs. P; Dosovitskiy. A; Brox. T;Pock. T;"An iterated L1 Algorithm for Non-Smooth Non-Convex Optimization in Computer Vision"IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2013.

[2] Yang, A.Y.; Zihan Zhou; Balasubramanian, A.G.; Sastry, S.S.; Yi Ma, "Fast ℓ_1 -Minimization Algorithms for Robust Face Recognition," Image Processing, IEEE Transactions on , vol.22, no.8, pp.3234,3246, Aug. 2013

[3] S. R. Becker, E. J. Cands, and M. Grant, Templates for convex cone problems with applications to sparse signal recovery, Math. Program. Compon., vol. 3, no. 3, pp. 165218, 2010.

[4] D. Malioutov, M. Cetin, and A. Willsky, Homotopy continuation for sparse signal representation, in Proc. IEEE Int. Conf. Acoust., Speech, Signal Process., Mar. 2005, pp. 733736.

[5] Rudin, Leonid I., Stanley Osher, and Emad Fatemi. "Nonlinear total variation based noise removal algorithms." Physica D: Nonlinear Phenomena 60.1 (1992): 259-268.

[6] Candes, Emmanuel J., Michael B. Wakin, and Stephen P. Boyd. "Enhancing sparsity by reweighted 1 minimization." Journal of Fourier analysis and applications 14.5-6 (2008): 877-905.