About Split Proximal Algorithms for the Q-Lasso

Abdellatif Moudafi

Aix Marseille Université, CNRS-L.S.I.S UMR 7296, France
e-mail : abdellatif.moudafi@univ-amu.fr

Abstract : Numerous problems in signal processing and imaging, statistical learning and data mining, or computer vision can be formulated as optimization problems which consist in minimizing a sum of convex functions, not necessarily differentiable, possibly composed with linear operators. Each function is typically either a data fidelity term or a regularization term enforcing some properties on the solution, see for example [1, 2] and references therein. In this note we are interested in the general form of Q-Lasso introduced in [3] which generalized the well-known Lasso of Tibshirani [4]. Q is a closed convex subset of a Euclidean m-space, for some integer $m \geq 1$, that can be interpreted as the set of errors within given tolerance level when linear measurements are taken to cover a signal/image via the Lasso. Only the unconstrained case was discussed in [3], we discuss here some split proximal algorithms for solving the general case. It is worth mentioning that the lasso model a number of applied problems arising from machine learning and signal/image processing due to the fact it promotes the sparsity of a signal.

Keywords : Q-Lasso; basic pursuit; proximal algorithm; soft-thresholding; regularization; Douglas-Rachford algorithm; forward-backward iterations.
2010 Mathematics Subject Classification : 49J53; 65K10; 49M37; 90C25.

1 Introduction

The lasso of Tibshirani [4] is the minimization problem

$$\min_{x \in \mathbb{R}^n} \frac{1}{2} \|Ax - b\|_2^2 + \gamma \|x\|_1, \tag{1.1}$$

where A is an $m \times n$ real matrix, $b \in I\!\!R^m$ and $\gamma > 0$ is a tuning parameter. It is equivalent to the the basic pursuit (BP) of Chen et al. [5]

$$\min_{x \in I\!\!R^n} \|x\|_1 \quad \text{subject to } Ax = b. \tag{1.2}$$

However, due to errors of measurements, the constraint $Ax = b$ is actually inexact; It turns out that problem (1.2) is reformulated as

$$\min_{x \in I\!\!R^n} \|x\|_1 \quad \text{subject to } \|Ax - b\|_p \leq \varepsilon, \tag{1.3}$$

where $\varepsilon > 0$ is the tolerance level of errors and p is often $1, 2$ or ∞. It is noticed in [3] that if we let $Q := B_\varepsilon(b)$, the closed ball in $I\!\!R^n$ with center b and radius ε, then (1.3) is rewritten as

$$\min_{x \in I\!\!R^n} \|x\|_1 \quad \text{subject to } Ax \in Q. \tag{1.4}$$

With Q a nonempty closed convex set of $I\!\!R^m$ and P_Q the projection from $I\!\!R^m$ onto Q and since that the constraint is equivalent to the condition $Ax - P_Q(Ax) = 0$, this leads to the following equivalent Lagrangian formulation

$$\min_{x \in I\!\!R^n} \frac{1}{2}\|(I - P_Q)Ax\|_2^2 + \gamma\|x\|_1, \tag{1.5}$$

with $\gamma > 0$ a Lagrangian multiplier. A connection is also made in [3] with the so-called split feasibility problem [6] which is stated as finding x verifying

$$x \in C, \quad Ax \in Q, \tag{1.6}$$

where C and Q are closed convex subsets of $I\!\!R^n$ and $I\!\!R^m$, respectively. An equivalent minimization formulation of (1.6) is

$$\min_{x \in C} \frac{1}{2}\|(I - P_Q)Ax\|_2^2. \tag{1.7}$$

Its l_1 regularization is given as

$$\min_{x \in C} \frac{1}{2}\|(I - P_Q)Ax\|_2^2 + \gamma\|x\|_1, \tag{1.8}$$

where $\gamma > 0$ is a regularization parameter.

The main difficulty in solving (1.8) stems form the fact that m, n are typically of high, hence the denomination of large-scale optimization. It is not possible to manipulate, at every iteration of an algorithm, matrices of size $n \times n$, like the Hessian of a function. So, proximal algorithms, which only exploit first-order information of the functions, are often the only viable way to solve (1.8). In this note, we propose two proximal algorithms to solve the problem (1.8) by full splitting; that is, at every iteration, the only operations involved are evaluations

of the gradient of the function $\frac{1}{2}\|(I - P_Q)A(\cdot)\|_2^2$, the proximal mapping of $\gamma\|\cdot\|_1$, A, or its transpose A^t.

In [3], properties and iterative methods for (1.5) are investigated. Remember also that many authors devoted their works to the unconstrained minimization problem $\min_{x \in H} f_1(x) + f_2(x)$ with f_1, f_2 are two proper, convex lower semi continuous functions defined on a Hilbert space H and f_2 differentiable with a β-Lipschitz continuous gradient for some $\beta > 0$ and an effective method to solve it is the forward-backward algorithm which from an initial value x_0 generates a sequence (x_k) by the following iteration

$$x_{k+1} = (1 - \lambda_k)x_k + \lambda_k prox_{\gamma_k f_1}(x_k - \gamma_k \nabla f_2(x_k)), \qquad (1.9)$$

where $\gamma_k > 0$ is the algorithm step-size, $0 < \lambda_k < 1$ is a relaxation parameter.
It is well-known, see for instance [1], that if (γ_k) is bounded and (λ_k) is bounded from below, then (x_k) weakly converges to a solution of $\min_{x \in H} f_1(x) + f_2(x)$ provided that the set of solutions is nonempty.

To relaxing the assumption on the differentiability of f_2, the Douglas-Rachford algorithm was introduce. It generates a sequence (y_k) as follows

$$\begin{cases} y_{k+1/2} = prox_{\kappa f_2} y_k; \\ y_{k+1} = y_k + \tau_k\big(prox_{\kappa f_1}(2y_{k+1/2} - y_k) - y_{k+1/2}\big) \end{cases} \qquad (1.10)$$

where $\kappa > 0$, (τ_k) is a sequence of positive reals. It is well-known that (y_k) converges weakly to y such that $prox_{\kappa f_2} y$ is a solution of the unconstrained minimization problem above provided that: $\forall k \in \mathbb{N}$, $\tau_k \in]0, 2[$ and $\sum_{k=0}^{\infty} \tau_k(2 - \tau_k) = +\infty$ and the set of solutions is nonempty.

In this note we are interested in (1.8) which is more general and reduced to (1.5) when the set of constraints is the entire space $\mathbb{R}^n$. The involvement of the convex set C brings some technical difficulties which are overcome in what follows. Since the same properties for problem (1.8) may be obtained by mimicking the analysis developed in [3] for (1.5), we will focus our attention on the algorithmic aspect.

2   Split Proximal Algorithms

In this section, we introduce several proximal iterative methods for solving Q-Lasso (1.8) in its general form. For the sake of our purpose we confine ourselves to the finite dimensional setting, but our analysis is still valid in Hilbert spaces, just replace the transpose A^t of A by its adjoint operator A^* and the convergence by the weak convergence.

To begin with, recall that the proximal mapping (or the Moreau envelope) of a proper, convex and lower semicontinuous function φ of parameter $\lambda > 0$ is defined by

$$prox_{\lambda\varphi}(x) := arg \min_{v \in \mathbb{R}^n} \{\varphi(v) + \frac{1}{2}\|v - x\|^2\}, \ x \in \mathbb{R}^n,$$

and that it has closed-form expression in some important cases. For example, if
$\varphi = \| \cdot \|_1$, then for $x \in I\!\!R^n$

$$prox_{\lambda\|\cdot\|_1}(x) = (prox_{\lambda|\cdot|}(x_1), prox_{\lambda|\cdot|}(x_n)), \qquad (2.1)$$

where $prox_{\lambda|\cdot|}(x_k) = sgn(x_k) \max_{k=1,2,\cdots n}\{|x_k| - \lambda, 0\}$.
If $\varphi(x) = \frac{1}{2}\|Ax - y\|^2$, then

$$Prox_{\gamma\varphi} = (I + \gamma A^t A)^{-1} A^t = A^t(I + \gamma AA^t)^{-1},$$

and if $\varphi = i_C$, we have

$$Prox_{\gamma\varphi}(x) = Proj_C(x) := arg \min_{z \in C} \|x - z\|,$$

where

$$i_C(x) = \left\{ \begin{array}{l} 0 \text{ if } x \in C; \\ +\infty \, otherwise \end{array} \right.$$

such function is convenient to enforce hard constraints on the solution. Observe
that the minimization problem (1.8) can be written as

$$\min_{x \in I\!\!R^n} \frac{1}{2}\|(I - P_Q)Ax\|_2^2 + \gamma\|x\|_1 + i_C(x). \qquad (2.2)$$

Remember also that the partial differential is defined as

$$\partial\varphi(x) := \{u \in I\!\!R^n; \varphi(z) \geq \varphi(x) + \langle u, z - x \rangle \; \forall z \in I\!\!R^n\}.$$

It is easily seen that

$$\partial\frac{1}{2}\|Ax - y\|^2 = \nabla\frac{1}{2}\|Ax - y\|^2 = A^t(Ax - y) = A^t(Ax - y)$$

and that

$$\partial\| \cdot \|_1(x) = \left\{ \begin{array}{l} sign(x_i) \text{ if } x_i \neq 0; \\ \, [-1, 1] \text{ if } x_i = 0 \end{array} \right.$$

and ∂i_C is nothing but the normal cone to C.

Very recently several split proximal algorithms were proposed to minimize the
problem

$$\min_{x \in I\!\!R^n} f(x) + g(x) + i_C(x), \qquad (2.3)$$

where f, g are two proper, convex and lower semicontinuous functions and C a
nonempty closed convex set.

Based in the algorithms introduced in [5], we propose two solutions for solv-
ing the more general problem (2.2). Both of them are obtained by coupling the
forward-backward algorithm and the Douglas-Rachford algorithm. Then we men-
tion other solutions introduced in [7] founded on a combination of the Krasnoselski-
Mann iteration and the Douglas-Rachford algorithm or an alternating algorithm.
Other solutions may be considered by using primal-dual algorithms proposed in
[2].

2.1 Insertion of a Forward-Backward Step in the Douglas-Rachford Algorithm

To apply the Douglas-Rachford algorithm when $g_1 = \gamma \| \cdot \|_1$ et $g_2 = \frac{1}{2}\|(I - P_Q)A(\cdot)\|_2^2 + i_C$, we need to determine their proximal mappings. The main difficulty lies in the computation of the second one, namely $prox_{\kappa \frac{1}{2}\|(I-P_Q)A(\cdot)\|_2^2 + i_C}$. As in [5], we can use a forward-backward algorithm to achieve this goal. The resulting algorithm is

Algorithm 1:

1. Set $\underline{\gamma} \in]0, 2\kappa^{-1}\|A\|^{-1}], \underline{\lambda} \in]0, 1]$ and $\kappa \in]0, +\infty[$.

   Choose $(\tau_k)_{k \in \mathbb{N}}$ satisfying $\forall k \in \mathbb{N}, \ \tau_k \in]0, 2[$ and $\sum_{k=0}^{\infty} \tau_k(2 - \tau_k) = +\infty$.

2. Set $k = 0, y_0 = y_{-1/2} \in C$.

3. Set $x_{k,0} = y_{k-1/2}$.

4. Pour $i = 0, ..., N_k - 1$,

   a) choose $\gamma_{k,n} \in [\underline{\gamma}, 2\kappa^{-1}\|A\|^{-1}[$ and $\lambda_{k,i} \in [\underline{\lambda}, 1]$;

   b) compute

   $$x_{k,i+1} = x_{k,i} + \lambda_{k,i}\big(P_C\big(\frac{x_{k,i} - \gamma_{k,i}(\kappa A^t(I - P_Q)Ax_{k,i} - y_k)}{1 + \gamma_{k,i}}\big) - x_{k,i}\big).$$

5. Set $y_{k+1/2} = x_{k,N_k}$.

6. Set $y_{k+1} = y_k + \tau_k(prox_{\kappa\|\|_1}(2y_{k+1/2} - y_k) - y_{k+1/2})$.

7. Increment $k \leftarrow k + 1$ and go to 3.

By a judicious choose of of N_k, the convergence of the sequence (y_k) to y such that $prox_{\kappa \frac{1}{2}\|(I-P_Q)A(\cdot)\|_2^2 + i_C}(y)$ solves problem (2.2), follows directly by applying [5]-Proposition 4.1.

2.2 Insertion of a Douglas-Rachford Step in the Forward-Backward Algorithm

We consider $f_1 = \kappa\| \cdot \|_1 + i_C$ et $f_2 = \frac{1}{2}\|(I - P_Q)A(\cdot)\|_2^2$. Since f_2 has a $\|A\|^2$-Lipschitz gradient, we can apply the forward-backward algorithm. This requires however to compute $prox_{i_C + \gamma_k\|\cdot\|}$ which can be performed with Douglas-Rachford iterations. The resulting algorithm is

Algorithm 2:

1. Choose γ_k and λ_k satisfying assumptions $0 < \inf_k \gamma_k \leq \sup_k \gamma_k < 2/\|A\|^2$, $0 < \underline{\lambda} \leq \lambda_k \leq 1$.

   Set $\underline{\tau} \in]0, 2]$.

2. Set $k = 0, x_0 \in C$.

3. Set $x'_k = x_k - \gamma_k A^t(I - P_Q)Ax_n$.

4. Set $y_{k,0} = 2 prox_{\gamma_k \|\cdot\|_1} x'_k - x'_k$.

5. For $i = 0, \cdots, M_k - 1$,

   a) compute

   $$y_{k,i+1/2} = P_C(\frac{y_{k,i} + x'_k}{2});$$

   b) choose $\tau_{k,i} \in [\underline{\tau}, 2]$;

   c) compute $y_{k,i+1} = y_{k,i} + \tau_{k,i}(prox_{\gamma_k \|\cdot\|_1}(2y_{n,i+1/2} - y_{k,i}) - y_{n,i+1/2})$;

   d) if $y_{k,i+1} = y_{k,i}$, then goto 6.

6. Set $x_{k+1} = x_k + \lambda_k(y_{k,i+1/2} - x_k)$.

7. Increment $k \leftarrow k + 1$ and go to 3.

A direct application of [5]-Proposition 4.2 ensures the existence of positive integers $(\overline{M}_k)$ such that if for all $k \geq 0$ $M_k \geq \overline{M}_k$, then the sequence (x_k) weakly convergences to a solution of problem (2.2).

Remark 2.1. *Other split proximal algorithms may be designed by combining the fixed-point idea to compute the composite of a convex function with a linear operator introduced in [8] and the analysis developed for computing the proximal mapping of the sum of two convex functions developed in [7]. Primal-dual algorithms considered in [2] can also be used. Note that there are often several ways to assign the functions of (2.2) to the terms used in the generic problem investigated in [2, 8, 9].*

References

[1] P.L. Combettes, J.-C. Pesquet, A Douglas-Rachford splitting approach to nonsmooth convex variational signal recovery, IEEE J. Selected Topics Signal Process 1 (2007) 564-574.

[2] L. Condat, A Generic Proximal Algorithm for Convex Optimization: Application to Total Variation Minimization, IEEE Signal Proc. Letters 21 (8) (2014) 985-989.

[3] M.A. Alghamdi, M. Ali Alghamdi, N. Shahzad, Naseer, H.-K. Xu, Properties and Iterative Methods for the Q-Lasso, Abstract & Applied Analysis (2013), Article ID 250943, 8 pages.

[4] R. Tibshirani, Regression Shrinkage and Selection Via the Lasso, Journal of the Royal Statistical Society (Series B) 58 (1) (1996) 267-288.

[5] S.S. Chen, D.L. Donoho, M.A. Saunders, Atomic decomposition by basis pursuit, SIAM journal on scientific computing 20 (1) (1998) 33-61.

[6] Y. Censor, T. Elfving, A multiprojection algorithm using Bregman projections in a product space, Numer. Algorithms 8 (1994) 221-239.

[7] A. Moudafi, Computing the resolvent of composite operators, CUBO A Mathematical Journal 16 (3) (2014) 87-96.

[8] Ch.A. Micchelli, L. Shen, Y. Xu, X. Zeng, Proximity algorithms for the L_1/TV image denoising model, Adv Comput Math 38 (2013) 401-426.

[9] C. Chaux, J.-C. Pesquet, N. Pustelnik, Nested iterative algorithms for convex constrained image recovery problems, SIAM Journal on Imaging Sciences 2 (2) (2009) 730-762.