Quadratic Envelopes; Read Me

In order to promote the use of Quadratic Envelopes for applications in compressed sensing, sparse recovery, low-rank recovery and similar problems, this site contains implemented versions of some key proximal operators. All code is written in MATLAB, and it is complemented with explanations of what is going on. Below, γ is the tuning parameter of the quadratic envelope and ρ is the "step-size" parameter in the proximal operator. We need $\rho > \gamma$ for these to work out, see [1] for the general idea behind quadratic envelopes. We first treat the case of vectors and then the case of matrices.

Sparse recovery

`ProxQmucard` This function computes

$$\mathrm{prox}_{\mathcal{Q}_\gamma(\mu\mathrm{card})/\rho}(y) = \operatorname*{arg\,min}_{x\in\mathbb{C}^n} \mathcal{Q}_\gamma(\mu\mathrm{card})(x) + \frac{\rho}{2}\|x-y\|^2$$

where $\gamma > 0$ and $\mu > 0$ are parameters, and $\mathrm{card}(y) = \|y\|_0$. More concretely we have that

$$(\mathrm{prox}_{\mathcal{Q}_\gamma(\mu\mathrm{card})/\rho}(y))_i = \begin{cases} y_i & \text{if } |y_i| \geq \sqrt{2\mu/\gamma} \\ \frac{\rho y_i - \sqrt{2\mu\gamma}\cdot\arg(y_i)}{\rho-\gamma} & \text{if } \sqrt{2\mu\gamma}/\rho \leq |y_i| \leq \sqrt{2\mu/\gamma} \\ 0 & \text{if } |y_i| \leq \sqrt{2\mu\gamma}/\rho. \end{cases}$$

Details are to be found in [2] and [1]. Note that in the limit case $\rho = \gamma$ this becomes hard thresholding with threshold $\sqrt{2\mu/\gamma}$. The proximal operator is identical to the one used for the Minimax Concave Penalty [6].

`ProxQgammaiota` This function computes

$$\mathrm{prox}_{\mathcal{Q}_\gamma(\iota_K)/\rho}(y) = \operatorname*{arg\,min}_{x\in\mathbb{C}^n} \mathcal{Q}_\gamma(\iota_K)(x) + \frac{\rho}{2}\|x-y\|^2$$

where $\iota_K(y) = \infty$ if $\mathrm{card}(y) > K$ and 0 else. The procedure is tricky, we here outline the key steps. The rationale behind the steps are described (in an almost identical situation) in [3].

Given a complex or real vector x let $|x|$ be vector of the corresponding absolute values. Consider the (family of) map(s) $\pi_x : \mathbb{C}^n \to \mathbb{C}^n$ such that $\pi_x(y)$ *first* reorders $|y|$ as $|x|$ and after gives to the reordered $|y|$ the same phases of x (so that $\arg(\pi_x(y)_i) = \arg(x_i)$). (In case two entries have the same modulus this is ambiguous, which reflects the fact that in such cases the proximal operator can be multi-valued, and any choice will lead to a minimizer.) Also with the symbol $|\widetilde{x}|$ we mean the vector of absolute values of x re-ordered decreasingly (first the component-wise absolute value is taken and then the vector is ordered decreasingly).

Now consider the vector x obtained by the following procedure:

1. Introduce the auxiliary vector r defined by $r_i = |\widetilde{y}|_i$ if $i \leq K$ and $r_i = \rho|\widetilde{y}|_i/\gamma$ else. Then, if $r_K > r_{K+1}$, set
$$x = \pi_y(r).$$
2. If not, compute the following indexes: j^* is the smallest index such that $r_{j^*} \leq r_{K+1}$ and l^* is the largest index such that $r_{l^*} \geq r_K$;
3. Sort the numbers $\{r_i\}_{i=j^*}^{l^*}$ decreasingly and call the corresponding vector z;
4. For $m \in \{1, \dots, l^* - j^* - 1\}$ (starting with $m = 1$) set
$$s = (z_m + z_{m+1})/2.$$
5. Let j be the first index such that $r_j \leq s_m$ and let l be the last index such that $r_l \geq s_m$. Compute
$$s_I = \frac{\rho \sum_{i=j}^{l} |\widetilde{y}|_i}{\rho(K + 1 - j) + \gamma(l - K)}.$$
6. If $z_{m+1} \leq s_I \leq z_m$; introduce a new vector $\hat{x}$ by setting $\hat{x}_i = \max(s_I, r_i)$ if $i \leq K$ and $\hat{x}_i = \min(s_I, r_i)$ if $i > K$. Return
$$x = \pi_y(\hat{x});$$
If not, increase m of one and repeat the steps $4 - 6$. This procedure will eventually terminate.

In conclusion we thus have that

$$\operatorname{prox}_{\mathcal{Q}_2(\iota_K)/\rho}(y) = \frac{\rho y - \gamma x}{\rho - \gamma} \tag{1}$$

where x is obtained via the steps $1, 2, \dots, 6$. Details of (1) can be found in [2] (Proposition 3.3). More details and proofs are found in [3] (Section 4.2).

The first proximal operator can be used when the sparsity degree of the sought solution is completely unknown, whereas the second can be used when it is exactly known. In between these there are many more advanced proximal operators which can be used when upper and lower limits of degree of sparsity is known, but we do not have online versions of them yet. Instructions for how to make them yourself is found in [5].

The coming two proximal operators come from quadratic envelopes on $\mathbb{R}^n$, where all vectors with negative values are banned. Let ι_+ be the indicator functional of the non-negative quadrant, i.e. $\iota_+(x) = 0$ if and only if $x_j \geq 0$ for all j, and ∞ else.

`ProxQmucardplus` This function solves

$$\underset{x\in\mathbb{R}^n}{\arg\min}\, \mathcal{Q}_\gamma(\mu\text{card} + \iota_+)(x) + \frac{\rho}{2}\|x-y\|^2.$$

It is simply `ProxQmucard` applied to the vector $\max(0, y)$.

`ProxQgammaiota` This function computes

$$\underset{x\in\mathbb{C}^n}{\arg\min}\, \mathcal{Q}_\gamma(\iota_K + \iota_+)(x) + \frac{\rho}{2}\|x-y\|^2.$$

The computations are a slight alteration of those in `ProxQgammaiota`. See [3] (Section 4.2) for documentation.

Low-rank recovery

Each of the above proximal operators has a counterpart for low rank matrix problems. Given a matrix X we let $\sigma(X)$ be its singular values and $\lambda(X)$ its eigenvalues (in the self-adjoint case). Note that e.g. $\text{card}(\sigma(X)) = \text{rank}(X)$, so each of the above functionals is low rank inducing when combined with either σ or λ. Thus, to compute e.g.

$$\underset{X}{\arg\min}\, \mathcal{Q}_\gamma(\mu\ \text{rank})(X) + \frac{\rho}{2}\|X-Y\|^2,$$

one proceeds by letting $U\text{diag}(y)V^*$ be the SVD of Y, then one computes $x =$ `ProxQmucard`(y, μ, γ, ρ) and the solution is $X = U\text{diag}(x)V^*$. This is further explained in [4].

For problems involving Hermitian matrices and Positive Semidefinite conditions, one does the same but using the eigendecomposition $Y = U\text{diag}(x)U^*$, see [3].

References

[1] Marcus Carlsson. On convex envelopes and regularization of non-convex functionals without moving global minima. *Journal of Optimization Theory and Applications*, 183:66–84.

[2] Marcus Carlsson. On convexification/optimization of functionals including an l2-misfit term. *arXiv preprint arXiv:1609.09378*, 2016.

[3] Marcus Carlsson and Daniele Gerosa. On phase retrieval via matrix completion and the estimation of low rank psd matrices. *Inverse Problems*, 36(1), 2019.

[4] Marcus Carlsson, Daniele Gerosa, and Carl Olsson. An unbiased approach to low rank recovery. *SIAM Journal on Optimization*, 32(4):2969–2996, 2022.

[5] Viktor Larsson and Carl Olsson. Convex low rank approximation. *International Journal of Computer Vision*, 120(2):194–214, 2016.

[6] Cun-Hui Zhang et al. Nearly unbiased variable selection under minimax concave penalty. *The Annals of Statistics*, 38(2):894–942, 2010.