ADMM

DSA5103 Lecture 9

Yangjing Zhang

16-Mar-2023

NUS

Today's content

1. Robust PCA
2. ADMM
3. ADMM for robust PCA
4. ADMM for Lasso

Robust PCA

Recover sparse and low-rank matrices

Suppose we are given a matrix $M \in \mathbb{R}^{m \times n}$

$$M = \underbrace{L_0}_{\text{low}-\text{rank}} + \underbrace{S_0}_{\text{sparse}}$$

Robust PCA aims to recover both a low-rank L and a sparse S from M

There are many important applications in which the data can naturally be modeled as a low-rank plus a sparse contribution, for example,

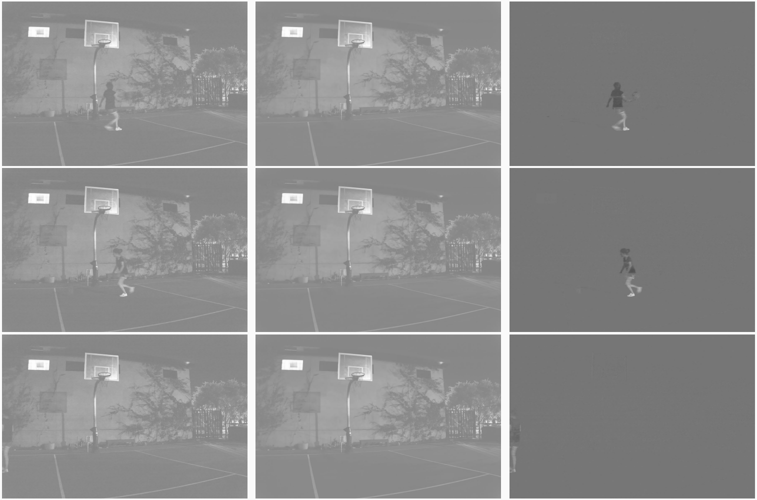
- Given a sequence of surveillance video frames (stacking the video frames as columns of a matrix M)
- we often need to identify activities that stand out from the background
- the low-rank component L_0 naturally corresponds to the stationary background
- the sparse component S_0 captures the moving objects in the foreground

Application: video data

- Basketball player video
- The video contains $n = 112$ frames
- Each frame's resolution 918×1374, $m = 1,261,332$
- Data matrix $M \in \mathbb{R}^{1,261,332 \times 112}$

Image = background + moving object

$M = L + S$



Original M Low-rank L Sparse S

Figure 1: The 30-th, 60-th, 90-th frames of the basketball player video. Column 1: raw frame M, Col 2: background L, Col 3: moving foreground S.

Surveillance video

Separation of background (low-rank) and moving objects (sparse)

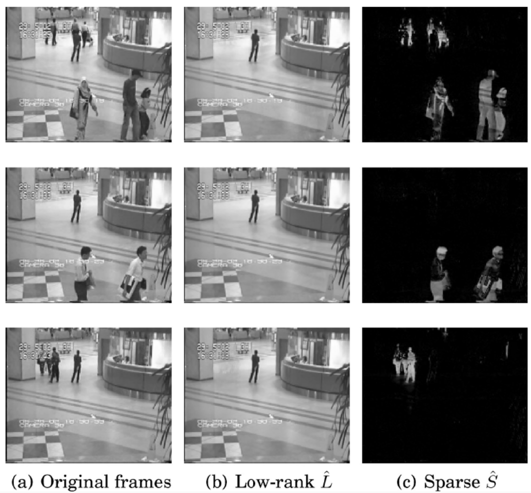


Figure 2: Image from [1, Fig.2]

PCA

Classical PCA seeks the best rank-k estimate of L_0 by solving

$$\begin{aligned} \min_{L} \quad & \sigma_{\max}(M - L) \\ \text{s.t.} \quad & \operatorname{rank}(L) \leq k \end{aligned}$$

- The principal components can be computed by eigenvalue decomposition of the data covariance matrix (as in Lecture 1)
- The principal components can also be computed by singular value decomposition of the data matrix

However, PCA is sensitive to outliers

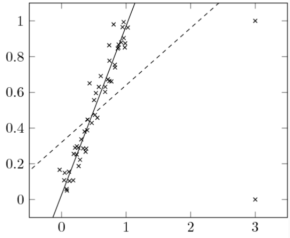


Figure 3: Image from internet

Robust PCA

Suppose we are given an $m \times n$ matrix $M = \underbrace{L_0}_{\text{low-rank}} + \underbrace{S_0}_{\text{sparse}}$. The low-rank L_0 and sparse S_0 can be recovered (under some assumptions) via solving the convex optimization

$$\begin{aligned} \min_{L,S} \quad & \|L\|_* + \lambda \|S\|_1 \\ \text{s.t.} \quad & L + S = M \end{aligned}$$

- Minimizing the nuclear norm $\|L\|_*$ will promote low-rankness of L
  - ▷ $\|L\|_*$ is the "best" convex approximation of $\text{rank}(L)$
- Minimizing the ℓ_1 norm $\|S\|_1$ will promote sparsity of S
  - ▷ $\|S\|_1$ is the "best" convex approximation of $\text{nnz}(L)$, which denotes the number of nonzero entries in L
- λ controls the trade-off between low-rankness and sparsity. In practice, $\lambda = \dfrac{1}{\sqrt{\max(m,n)}}$ always returns a good recovery

ADMM

Target problem

Our target problem is a convex optimization problem with 2-block separable structure — two variables, separable objective

$$\begin{aligned} \min_{y,z} \quad & f(y) + g(z) \\ \text{s.t.} \quad & Ay + Bz = c \end{aligned}$$

- Let $\mathcal{X}, \mathcal{Y}, \mathcal{Z}$ be finite dimensional real Euclidean spaces, e.g.,
  - ▷ the space of vectors $\mathbb{R}^n$
  - ▷ the space of matrices $\mathbb{R}^{m \times n}$
- $f : \mathcal{Y} \to (-\infty, +\infty]$ closed proper convex function
- $g : \mathcal{Z} \to (-\infty, +\infty]$ closed proper convex function
- $A : \mathcal{Y} \to \mathcal{X}$ linear map
- $B : \mathcal{Z} \to \mathcal{X}$ linear map

Target problem

For simplicity, we consider the target problem in space of vectors

$$\begin{aligned} \min_{y \in \mathbb{R}^m, z \in \mathbb{R}^n} \quad & f(y) + g(z) \qquad \text{(P)} \\ \text{s.t.} \quad & Ay + Bz = c \end{aligned}$$

given $A \in \mathbb{R}^{p \times m}$, $B \in \mathbb{R}^{p \times n}$, and $c \in \mathbb{R}^p$.

- Let $x \in \mathbb{R}^p$. Lagrangian $L(y, z, x) = f(y) + g(z) + \langle x, Ay + Bz - c \rangle$
- Dual function $\theta(x) = \min_{y,z} L(y, z, x)$
- Dual problem $\max_x \theta(x)$
- **Idea**: Gradient method for dual problem

Dual ascent

- Dual problem $\max\limits_{x} \theta(x)$
- Gradient method for dual problem $x^{(k+1)} = x^{(k)} + \tau \nabla\theta(x^{(k)})$
- Gradient descent $\rightarrow$ "ascent" for maximization problem
- $\theta(x^{(k)}) = \min\limits_{y,z} L(y, z, x^{(k)})$. If $(\bar{y}, \bar{z}) = \arg\min\limits_{y,z} L(\bar{y}, \bar{z}, x^{(k)})$, then

$$\nabla\theta(x^{(k)}) = \frac{\partial}{\partial x} L(\bar{y}, \bar{z}, x^{(k)}) = A\bar{y} + B\bar{z} - c$$

- Dual ascent

$$\begin{aligned} \left(y^{(k+1)}, z^{(k+1)}\right) &= \arg\min_{y,z} L(y, z, x^{(k)}) \\ x^{(k+1)} &= x^{(k)} + \tau\left(Ay^{(k+1)} + Bz^{(k+1)} - c\right) \end{aligned}$$

Method of multipliers

- A method to robustify dual ascent
- For $\sigma > 0$, the augmented Lagrangian function is

$$L_{\sigma}(y, z, x) = \underbrace{f(y) + g(z) + \langle x, Ay + Bz - c\rangle}_{L(y,z,x)} + \underbrace{\frac{\sigma}{2}\|Ay + Bz - c\|^2}_{\text{quadratic penalty}}$$

augmented Lagrangian = Lagrangian + quadratic penalty

- Method of multipliers

$$\left(y^{(k+1)}, z^{(k+1)}\right) = \arg\min_{y,z} L_{\sigma}(y, z, x^{(k)})$$

$$x^{(k+1)} = x^{(k)} + \sigma\left(Ay^{(k+1)} + Bz^{(k+1)} - c\right)$$

  Dual step length $= \sigma$

- Update (y, z) jointly is usually difficult. Consider alternating minimization.

Alternating direction method of multipliers

$$\min_{y,z} \quad f(y) + g(z) \quad \text{s.t.} \quad Ay + Bz = c \qquad \text{(P)}$$

Algorithm (Alternating direction method of multipliers (ADMM) for (P))

Choose $\sigma > 0$, $0 < \tau < \frac{1+\sqrt{5}}{2}$, $x^{(0)} \in \mathbb{R}^p$, $z^{(0)} \in \mathrm{dom}(g)$. Set $k \leftarrow 0$

repeat until convergence

$$y^{(k+1)} \leftarrow \arg\min_{y} \, L_\sigma(y, z^{(k)}, x^{(k)})$$

$$z^{(k+1)} \leftarrow \arg\min_{z} \, L_\sigma(y^{(k+1)}, z, x^{(k)})$$

$$x^{(k+1)} \leftarrow x^{(k)} + \tau\sigma(Ay^{(k+1)} + Bz^{(k+1)} - c)$$

$$k \leftarrow k + 1$$

end(repeat)

return $y^{(k)}, z^{(k)}, x^{(k)}$

$y^{(k+1)}$ and $z^{(k+1)}$ in the subproblems should be easy to update

Dual of target problem

$$\min_{y,z} f(y) + g(z) \text{ s.t. } Ay + Bz = c \quad \text{(P)}$$

The Lagrange dual problem of (P) is

$$\max_{x} \quad -f^*(-A^T x) - g^*(-B^T x) - \langle c, x \rangle \quad \text{(D)}$$

where $f^*(z) = \max_y \{\langle z, y \rangle - f(y)\}$ is the (Fenchel) conjugate[1] of f.

Derivation*

[1] Lecture 4, page 28

Dual of target problem

$$\min_{y,z} f(y) + g(z) \text{ s.t. } Ay + Bz = c \quad \text{(P)}$$

The Lagrange dual problem of (P) is

$$\max_{x} \quad -f^*(-A^T x) - g^*(-B^T x) - \langle c, x \rangle \quad \text{(D)}$$

where $f^*(z) = \max_y \{\langle z, y \rangle - f(y)\}$ is the (Fenchel) conjugate[1] of f.

Derivation*

1. $\forall\, y \in \mathbb{R}^m,\, x \in \mathbb{R}^p,\, A \in \mathbb{R}^{p \times m},\, \langle Ay, x \rangle = \langle y, A^T x \rangle$
2. The Lagrange dual function $\theta(x) = \min_{y,z} L(y, z, x) =$ $\min_y \{f(y) + \langle A^T x, y \rangle\} + \min_z \{g(z) + \langle B^T x, z \rangle\} - \langle c, x \rangle$
3. $\min_y \{f(y) + \langle A^T x, y \rangle\} = -\max_y \{\langle -A^T x, y \rangle - f(y)\} = -f^*(-A^T x)$

[1] Lecture 4, page 28

Convergence

$$\min_{y,z} \quad f(y) + g(z) \quad \text{s.t.} \quad Ay + Bz = c \quad \text{(P)}$$

$$\max_{x} \quad -f^*(-A^T x) - g^*(-B^T x) - \langle c, x \rangle \quad \text{(D)}$$

Theorem. Under some conditions:

(1) a constraint qualification holds (Assume there exists $(\hat{y}, \hat{z})$ in the relative interior of $\mathrm{dom}(f) \times \mathrm{dom}(g)$ such that $A\hat{y} + B\hat{z} = c$);

(2) every subproblem is well defined $\Sigma_f + \sigma AA^T$ and $\Sigma_g + \sigma BB^T$ are positive definite;

the sequence $\{(y^{(k)}, z^{(k)}, x^{(k)})\}$ generated by ADMM converges to $(\bar{y}, \bar{z}, \bar{x})$ with $(\bar{y}, \bar{z})$ optimal to (P) and $\bar{x}$ optimal to the Lagrange dual problem of (P).

$$\begin{aligned} y^{(k)}, z^{(k)} &\to \bar{y}, \bar{z} \quad && \text{primal optimal} \\ x^{(k)} &\to \bar{x} && \text{dual optimal} \end{aligned}$$

ADMM is a primal-dual method, solving both primal and dual problems

Revisit inner product and norm

- Inner product $\langle X, Y\rangle = \mathrm{Tr}(X^T Y)$
  - ▷ X and Y are vectors/matrices of the same dimension
- Norm $\|X\|^2 = \langle X, X\rangle$
  - ▷ In particular, $\|\cdot\| = \|\cdot\|_2$ for vectors and $\|\cdot\| = \|\cdot\|_F$ for matrice
- $\langle X, Y\rangle = \langle Y, X\rangle$
- $\|X+Y\|^2 = \|X\|^2 + \|Y\|^2 + 2\langle X, Y\rangle$
- $\langle X, Y\rangle + \frac{\sigma}{2}\|Y\|^2 = \frac{\sigma}{2}\|Y + \sigma^{-1}X\|^2 - \frac{1}{2\sigma}\|X\|^2$

Example: constrained convex problem

Apply ADMM for

$$\min_{y} \quad f(y) \quad \text{s.t.} \quad y \in C$$

- Transform it into the form that ADMM can handle

$$\begin{aligned} \min_{y,z} \quad & f(y) + \delta_C(z) \\ \text{s.t.} \quad & y - z = 0 \end{aligned}$$

Constraint "$Ay + Bz = c$": $A = I, B = -I, c = 0$

- The augmented Lagrangian function

$$\begin{aligned} L_\sigma(y, z, x) &= f(y) + \delta_C(z) + \langle x, y - z \rangle + \frac{\sigma}{2}\|y - z\|^2 \\ &= f(y) + \delta_C(z) + \frac{\sigma}{2}\|y - z + \sigma^{-1}x\|^2 - \frac{1}{2\sigma}\|x\|^2 \end{aligned}$$

Example: constrained convex problem

$$L_\sigma(y,z,x) = f(y) + \delta_C(z) + \frac{\sigma}{2}\|y - z + \sigma^{-1}x\|^2 - \frac{1}{2\sigma}\|x\|^2$$

Subproblem-y:

$$\begin{aligned} &\arg\min_y \left\{ f(y) + \frac{\sigma}{2}\|y - z + \sigma^{-1}x\|^2 \right\} \\ = \; &\arg\min_y \left\{ \frac{1}{\sigma} f(y) + \frac{1}{2}\|y - (z - \sigma^{-1}x)\|^2 \right\} = P_{\frac{1}{\sigma}f}(z - \sigma^{-1}x) \end{aligned}$$

Subproblem-z: $\Pi_C(y + \sigma^{-1}x)$

ADMM iteration:

$$\begin{aligned} y^{(k+1)} &= P_{\frac{1}{\sigma}f}(z^{(k)} - \sigma^{-1}x^{(k)}) \ \text{the difficulty depends on } f \\ z^{(k+1)} &= \Pi_C(y^{(k+1)} + \sigma^{-1}x^{(k)}) \\ x^{(k+1)} &= x^{(k)} + \tau\sigma(y^{(k+1)} - z^{(k+1)}) \end{aligned}$$

Example: Consensus optimization

Apply ADMM for consensus optimization

$$\min_{y} \quad \sum_{i=1}^{n} f_i(y)$$

f_i is loss function for i-th training data e.g., in linear regression / SVM

- Transform it into the form that ADMM can handle

$$\begin{aligned} \min_{y_i, z} \quad & \sum_{i=1}^{n} f_i(y_i) \\ \text{s.t.} \quad & y_i - z = 0, \, \forall\, i \in [n] \end{aligned}$$

  - ▷ y_i are local variables, z is the global variable
  - ▷ $y_i - z = 0$ are consensus constraints

- One block $\{y_1, \ldots, y_n\}$, the other block z, constraint:

$$\begin{bmatrix} I & & \\ & \ddots & \\ & & I \end{bmatrix} \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix} - \begin{bmatrix} I \\ \vdots \\ I \end{bmatrix} z = 0$$

Example: Consensus optimization

- The augmented Lagrangian

$$L_\sigma(y_1, \ldots, y_n, z, x) = \sum_{i=1}^n \left(f_i(y_i) + \langle x_i, y_i - z \rangle + \frac{\sigma}{2}\|y_i - z\|^2 \right)$$
$$= \sum_{i=1}^n \left(f_i(y_i) + \frac{\sigma}{2}\|y_i - z + \sigma^{-1}x_i\|^2 - \frac{1}{2\sigma}\|x_i\|^2 \right)$$

- $(y_1, \ldots, y_n)$ can be updated jointly

$$\arg\min_{y_i} \left\{ \frac{1}{\sigma} f_i(y_i) + \frac{1}{2}\|y_i - (z - \sigma^{-1}x_i)\|^2 \right\} = P_{\frac{1}{\sigma} f_i}(z - \sigma^{-1}x_i)$$

- Set $0 = \nabla_z = \sigma \sum_{i=1}^n (z - y_i - \sigma^{-1}x_i) \Rightarrow z = \frac{1}{n}\sum_{i=1}^n (y_i + \sigma^{-1}x_i)$
- Multipliers $x_i \leftarrow x_i + \tau\sigma(y_i - z)$

ADMM for robust PCA

ADMM for robust PCA

Apply ADMM for

$$\begin{aligned} \min_{L,S} \quad & \underbrace{\|L\|_*}_{f(L)} + \underbrace{\lambda\|S\|_1}_{g(S)} \\ \text{s.t.} \quad & L + S = M \end{aligned}$$

The augmented Lagrangian function

$$\begin{aligned} L_\sigma(L, S, Z) &= \|L\|_* + \lambda\|S\|_1 + \langle Z, L + S - M\rangle + \frac{\sigma}{2}\|L + S - M\|_F^2 \\ &= \|L\|_* + \lambda\|S\|_1 + \frac{\sigma}{2}\|L + S - M + \sigma^{-1}Z\|_F^2 - \frac{1}{2\sigma}\|Z\|_F^2 \end{aligned}$$

Two subproblems: subproblem-L, subproblem-S

Update multipliers: $Z^{(k+1)} = Z^{(k)} + \tau\sigma(L^{(k+1)} + S^{(k+1)} - M)$

Subproblem-L

$$L_\sigma(L,S,Z) = \|L\|_* + \lambda\|S\|_1 + \frac{\sigma}{2}\|L+S-M+\sigma^{-1}Z\|_F^2 - \frac{1}{2\sigma}\|Z\|_F^2$$

Consider minimizing w.r.t. L

$$\begin{aligned}
&\arg\min_L \quad \|L\|_* + \frac{\sigma}{2}\|L+S-M+\sigma^{-1}Z\|_F^2 \\
=&\arg\min_L \quad \frac{1}{\sigma}\|L\|_* + \frac{1}{2}\|L-(M-S-\sigma^{-1}Z)\|_F^2 \\
=&P_{\frac{1}{\sigma}\|\cdot\|_*}(M-S-\sigma^{-1}Z)
\end{aligned}$$

Therefore, the k-th iteration can be obtained by[2]

1. Compute SVD: $M - S^{(k)} - \sigma^{-1}Z^{(k)} = U^{(k)}\mathrm{Diag}(d^{(k)})(V^{(k)})^T$
2. Soft-thresholding: $\gamma^{(k)} = S_{\frac{1}{\sigma}}(d^{(k)})$
3. $L^{(k+1)} = U^{(k)}\mathrm{Diag}(\gamma^{(k)})(V^{(k)})^T$

[2] Lecture 8, page 39

Subproblem-S

$$L_\sigma(L,S,Z) = \|L\|_* + \lambda\|S\|_1 + \frac{\sigma}{2}\|L+S-M+\sigma^{-1}Z\|_F^2 - \frac{1}{2\sigma}\|Z\|_F^2$$

Consider minimizing w.r.t. S

$$\begin{aligned}
&\arg\min_S \quad \lambda\|S\|_1 + \frac{\sigma}{2}\|L+S-M+\sigma^{-1}Z\|_F^2 \\
=&\arg\min_S \quad \frac{\lambda}{\sigma}\|S\|_1 + \frac{1}{2}\|S-(M-L-\sigma^{-1}Z)\|_F^2 \\
=&P_{\frac{\lambda}{\sigma}\|\cdot\|_1}(M-L-\sigma^{-1}Z) \\
=&S_{\frac{\lambda}{\sigma}}(M-L-\sigma^{-1}Z)
\end{aligned}$$

The k-th iteration

$$S^{(k+1)} = S_{\frac{\lambda}{\sigma}}(M-L^{(k+1)}-\sigma^{-1}Z^{(k)})$$

ADMM framework

Algorithm (ADMM for robust PCA)

Choose $\sigma > 0$, $0 < \tau < \frac{1+\sqrt{5}}{2}$, $S^{(0)} \in \mathbb{R}^{m\times n}$, $Z^{(0)} \in \mathbb{R}^{m\times n}$.

Set $k \leftarrow 0$

repeat until convergence

$$
\begin{aligned}
T^{(k)} &\leftarrow M - S^{(k)} - \sigma^{-1} Z^{(k)} \\
T^{(k)} &= U^{(k)} \mathrm{Diag}(d^{(k)}) (V^{(k)})^T \quad \text{(SVD of } T^{(k)}\text{)} \\
\gamma^{(k)} &\leftarrow S_{\frac{1}{\sigma}}(d^{(k)}) \\
L^{(k+1)} &\leftarrow U^{(k)} \mathrm{Diag}(\gamma^{(k)}) (V^{(k)})^T \\
S^{(k+1)} &\leftarrow S_{\frac{\lambda}{\sigma}}(M - L^{(k+1)} - \sigma^{-1} Z^{(k)}) \\
Z^{(k+1)} &\leftarrow Z^{(k)} + \tau\sigma(L^{(k+1)} + S^{(k+1)} - M) \\
k &\leftarrow k+1
\end{aligned}
$$

end(repeat)

return $L^{(k)}, S^{(k)}, Z^{(k)}$

ADMM for Lasso

1. ADMM for Lasso

First, we attempt to apply ADMM for Lasso: given $X \in \mathbb{R}^{n\times p}$, $Y \in \mathbb{R}^n$

$$\min_{\beta\in\mathbb{R}^p} \quad \frac{1}{2}\|X\beta - Y\|^2 + \lambda\|\beta\|_1$$

- Transform it into the form that ADMM can handle ($u = Y - X\beta$)

$$\begin{aligned} \min_{u,\beta} \quad & \underbrace{\frac{1}{2}\|u\|^2}_{f(u)} + \underbrace{\lambda\|\beta\|_1}_{g(\beta)} \\ \text{s.t.} \quad & u + X\beta = Y \end{aligned}$$

  Constraint "$Au + B\beta = c$": $A = I$, $B = X, c = Y$

- Let $\xi \in \mathbb{R}^n$. The augmented Lagrangian function

$$\begin{aligned} L_\sigma(u,\beta,\xi) &= \frac{1}{2}\|u\|^2 + \lambda\|\beta\|_1 + \langle \xi, Y - u - X\beta\rangle + \frac{\sigma}{2}\|Y - u - X\beta\|^2 \\ &= \frac{1}{2}\|u\|^2 + \lambda\|\beta\|_1 + \frac{\sigma}{2}\|Y - u - X\beta + \sigma^{-1}\xi\|^2 - \frac{1}{2\sigma}\|\xi\|^2 \end{aligned}$$

1. ADMM for Lasso

$$L_\sigma(u,\beta,\xi) = \frac{1}{2}\|u\|^2 + \lambda\|\beta\|_1 + \frac{\sigma}{2}\|Y - u - X\beta + \sigma^{-1}\xi\|^2 - \frac{1}{2\sigma}\|\xi\|^2$$

- Subproblem-u: minimize a smooth convex function, explicit solution.
- Subproblem-β

$$\min_{\beta} \quad \frac{\sigma}{2}\|X\beta - (Y - u + \sigma^{-1}\xi)\|^2 + \lambda\|\beta\|_1$$

  It is as difficult as in solving the original Lasso problem

- Therefore, ADMM is not suitable for the transformation

$$\min_{\beta} \frac{1}{2}\|X\beta - Y\|^2 + \lambda\|\beta\|_1 \Rightarrow \min_{u,\beta} \frac{1}{2}\|u\|^2 + \lambda\|\beta\|_1 \text{ s.t. } u + X\beta = Y$$

2. ADMM for Lasso

Second, we attempt to apply ADMM for Lasso with a different transformation

$$\min_{\beta \in \mathbb{R}^p} \quad \frac{1}{2}\|X\beta - Y\|^2 + \lambda\|\beta\|_1$$

- Transform it with a slack variable $u = \beta$

$$\begin{aligned} \min_{\beta, u} \quad & \frac{1}{2}\|X\beta - Y\|^2 + \lambda\|u\|_1 \\ \text{s.t.} \quad & \beta - u = 0 \end{aligned}$$

- Let $\xi \in \mathbb{R}^p$. The augmented Lagrangian function

$$\begin{aligned} L_\sigma(\beta, u, \xi) &= \frac{1}{2}\|X\beta - Y\|^2 + \lambda\|u\|_1 + \langle \xi, \beta - u \rangle + \frac{\sigma}{2}\|\beta - u\|^2 \\ &= \frac{1}{2}\|X\beta - Y\|^2 + \lambda\|u\|_1 + \frac{\sigma}{2}\|\beta - u + \sigma^{-1}\xi\|^2 - \frac{1}{2\sigma}\|\xi\|^2 \end{aligned}$$

Subproblem-β

$$L_\sigma(\beta, u, \xi) = \frac{1}{2}\|X\beta - Y\|^2 + \lambda\|u\|_1 + \frac{\sigma}{2}\|\beta - u + \sigma^{-1}\xi\|^2 - \frac{1}{2\sigma}\|\xi\|^2$$

$$\min_\beta \left\{ \frac{1}{2}\|X\beta - Y\|^2 + \frac{\sigma}{2}\|\beta - u + \sigma^{-1}\xi\|^2 \right\}$$

The objective function is a smooth convex function. Set the gradient to be zero:

Subproblem-β

$$L_\sigma(\beta, u, \xi) = \frac{1}{2}\|X\beta - Y\|^2 + \lambda\|u\|_1 + \frac{\sigma}{2}\|\beta - u + \sigma^{-1}\xi\|^2 - \frac{1}{2\sigma}\|\xi\|^2$$

$$\min_\beta \left\{ \frac{1}{2}\|X\beta - Y\|^2 + \frac{\sigma}{2}\|\beta - u + \sigma^{-1}\xi\|^2 \right\}$$

The objective function is a smooth convex function. Set the gradient to be zero:

$$\begin{aligned} \nabla_\beta &= X^T(X\beta - Y) + \sigma(\beta - u + \sigma^{-1}\xi) \\ &= \left(\sigma I + X^T X\right)\beta - (X^T Y + \sigma u - \xi) \\ \Rightarrow \beta &= \left(\sigma I + X^T X\right)^{-1}(X^T Y + \sigma u - \xi) \end{aligned}$$

In k-th iteration $\beta^{(k+1)} = \left(\sigma I + X^T X\right)^{-1}\left(X^T Y + \sigma u^{(k)} - \xi^{(k)}\right)$

- The coefficient matrix $\sigma I + X^T X \in \mathbb{S}^p_{++}$ is constant over iterations. Therefore, one may save $\left(\sigma I + X^T X\right)^{-1}$ (or its decomposition) at the beginning to save computational cost

Subproblem-u

$$L_\sigma(\beta, u, \xi) = \frac{1}{2}\|X\beta - Y\|^2 + \lambda\|u\|_1 + \frac{\sigma}{2}\|\beta - u + \sigma^{-1}\xi\|^2 - \frac{1}{2\sigma}\|\xi\|^2$$

$$\begin{aligned}
& \arg\min_u \left\{ \lambda\|u\|_1 + \frac{\sigma}{2}\left\|\beta - u + \sigma^{-1}\xi\right\|^2 \right\} \\
= \; & \arg\min_u \left\{ \frac{\lambda}{\sigma}\|u\|_1 + \frac{1}{2}\left\|u - \left(\beta + \sigma^{-1}\xi\right)\right\|^2 \right\} \\
= \; & P_{\frac{\lambda}{\sigma}\|\cdot\|_1}\left(\beta + \sigma^{-1}\xi\right) \\
= \; & S_{\frac{\lambda}{\sigma}}\left(\beta + \sigma^{-1}\xi\right)
\end{aligned}$$

In k-th iteration $u^{(k+1)} = S_{\frac{\lambda}{\sigma}}\left(\beta^{(k+1)} + \sigma^{-1}\xi^{(k)}\right)$ soft-thresholding

ADMM framework

$$\min_{\beta, u} \quad \frac{1}{2}\|X\beta - Y\|^2 + \lambda\|u\|_1 \quad \text{s.t.} \quad \beta - u = 0$$

Algorithm (ADMM for Lasso)

Choose $\sigma > 0$, $0 < \tau < \frac{1+\sqrt{5}}{2}$, $u^{(0)} \in \mathbb{R}^p$, $\xi^{(0)} \in \mathbb{R}^p$. Set $k \leftarrow 0$

repeat until convergence

$$\beta^{(k+1)} \leftarrow \left(\sigma I + X^T X\right)^{-1} \left(X^T Y + \sigma u^{(k)} - \xi^{(k)}\right)$$

$$u^{(k+1)} \leftarrow S_{\frac{\lambda}{\sigma}} \left(\beta^{(k+1)} + \sigma^{-1}\xi^{(k)}\right)$$

$$\xi^{(k+1)} \leftarrow \xi^{(k)} + \tau\sigma \left(\beta^{(k+1)} - u^{(k+1)}\right)$$

$$k \leftarrow k + 1$$

end(repeat)

return $\beta^{(k)}, u^{(k)}, \xi^{(k)}$

When p is large, updating $\beta^{(k+1)}$ is time-consuming

3. ADMM for dual of Lasso

Third, we consider ADMM for solving the dual of Lasso

$$\min_{\beta \in \mathbb{R}^p} \quad \frac{1}{2}\|X\beta - Y\|^2 + \lambda\|\beta\|_1$$

Derive its dual

1. With a slack variable $u = Y - X\beta$, it is equivalent to

$$\begin{aligned} \min_{\beta, u} \quad & \frac{1}{2}\|u\|^2 + \lambda\|\beta\|_1 \\ \text{s.t.} \quad & u + X\beta = Y \end{aligned}$$

2. Let $y \in \mathbb{R}^n$. The Lagrangian is

$$\begin{aligned} L(\beta, u, y) &= \frac{1}{2}\|u\|^2 + \lambda\|\beta\|_1 + \langle y, Y - u - X\beta \rangle \\ &= \frac{1}{2}\|u\|^2 - \langle y, u \rangle + \lambda\|\beta\|_1 - \langle X^T y, \beta \rangle + \langle y, Y \rangle \end{aligned}$$

3. ADMM for dual of Lasso

3. The Lagrange dual function $\theta(y) = \min\limits_{\beta,u} L(\beta, u, y)$

$$= \min_{\beta,u} \left\{ \frac{1}{2}\|u\|^2 - \langle y, u \rangle + \lambda\|\beta\|_1 - \langle X^T y, \beta \rangle + \langle y, Y \rangle \right\}$$

$$= \min_{\beta} \left\{ \lambda\|\beta\|_1 - \langle X^T y, \beta \rangle \right\} + \min_{u} \left\{ \frac{1}{2}\|u\|^2 - \langle y, u \rangle \right\} + \langle y, Y \rangle$$

[3] Lecture 4, page 31

3. ADMM for dual of Lasso

3. The Lagrange dual function $\theta(y) = \min_{\beta,u} L(\beta, u, y)$

$$= \min_{\beta,u} \left\{ \frac{1}{2}\|u\|^2 - \langle y, u \rangle + \lambda \|\beta\|_1 - \langle X^T y, \beta \rangle + \langle y, Y \rangle \right\}$$

$$= \min_{\beta} \left\{ \lambda \|\beta\|_1 - \langle X^T y, \beta \rangle \right\} + \min_{u} \left\{ \frac{1}{2}\|u\|^2 - \langle y, u \rangle \right\} + \langle y, Y \rangle$$

- Set $\nabla_u = u - y = 0 \Rightarrow u = y$. $\min_u \left\{ \frac{1}{2}\|u\|^2 - \langle y, u \rangle \right\} = -\frac{1}{2}\|y\|^2$
- $\min_{\beta} \left\{ \lambda \|\beta\|_1 - \langle X^T y, \beta \rangle \right\} = -\lambda \max_{\beta} \left\{ \langle \frac{X^T y}{\lambda}, \beta \rangle - \underbrace{\|\beta\|_1}_{h(\beta)} \right\} =$

  $-\lambda h^* \left(\frac{X^T y}{\lambda} \right) \overset{3}{=} -\delta_{B_1} \left(\frac{X^T y}{\lambda} \right)$, $B_1 = \{\beta \in \mathbb{R}^p \mid \|\beta\|_\infty \le 1\}$

[3] Lecture 4, page 31

Application 1: dual of Lasso

3. The Lagrange dual function

$$\theta(y) = -\delta_{B_1}\left(\frac{X^T y}{\lambda}\right) - \frac{1}{2}\|y\|^2 + \langle y, Y\rangle, \quad B_1 = \{\beta \in \mathbb{R}^p \mid \|\beta\|_\infty \le 1\}$$

4. The Lagrange dual problem

$$\begin{aligned}
&\max_{y} \quad \theta(y) \\
=& -\min_{y} \quad \delta_C\left(\frac{X^T y}{\lambda}\right) + \frac{1}{2}\|y\|^2 - \langle y, Y\rangle \\
=& -\min_{y} \quad \frac{1}{2}\|y\|^2 - \langle y, Y\rangle \quad \text{s.t.} \quad \left\|X^T y\right\|_\infty \le \lambda \\
=& -\min_{y,v} \quad \frac{1}{2}\|y\|^2 - \langle y, Y\rangle + \delta_{B_\lambda}(v) \\
&\quad \text{s.t.} \quad X^T y + v = 0
\end{aligned}$$

where $B_\lambda = \{v \mid \|v\|_\infty \le \lambda\}$

Application 1: dual of Lasso

Dual problem

$$
\begin{aligned}
\min_{\xi, v} \quad & \underbrace{\frac{1}{2}\|y\|^2 - \langle y, Y\rangle}_{f(y)} + \underbrace{\delta_{B_\lambda}(v)}_{g(v)} \\
\text{s.t.} \quad & X^T y + v = 0
\end{aligned}
$$

Constraint "$Ay + Bv = c$": $A = X^T$, $B = I, c = 0$

- Let $\beta \in \mathbb{R}^p$. The augmented Lagrangian function

$$
\begin{aligned}
& L_\sigma(y, v, \beta) \\
= & \frac{1}{2}\|y\|^2 - \langle y, Y\rangle + \delta_{B_\lambda}(v) + \langle \beta, X^T y + v\rangle + \frac{\sigma}{2}\|X^T y + v\|^2 \\
= & \frac{1}{2}\|y\|^2 - \langle y, Y\rangle + \delta_{B_\lambda}(v) + \frac{\sigma}{2}\|X^T y + v + \sigma^{-1}\beta\|^2 - \frac{1}{2\sigma}\|\beta\|^2
\end{aligned}
$$

Subproblem-y

$$L_\sigma(y,v,\beta) = \frac{1}{2}\|y\|^2 - \langle y, Y\rangle + \delta_{B_\lambda}(v) + \frac{\sigma}{2}\|X^T y + v + \sigma^{-1}\beta\|^2 - \frac{1}{2\sigma}\|\beta\|^2$$

$$\min_y \left\{ \frac{1}{2}\|y\|^2 - \langle y, Y\rangle + \frac{\sigma}{2}\|X^T y + v + \sigma^{-1}\beta\|^2 \right\}$$

The objective function is a smooth convex function. Set the gradient to be zero:

$$\begin{aligned} \nabla_y &= y - Y + \sigma X(X^T y + v + \sigma^{-1}\beta) \\ &= \left(I + \sigma X X^T\right) y - (Y - X\beta - \sigma X v) \\ \Rightarrow y &= \left(I + \sigma X X^T\right)^{-1} (Y - X\beta - \sigma X v) \end{aligned}$$

In k-th iteration $y^{(k+1)} = \left(I + \sigma X X^T\right)^{-1} \left(Y - X(\beta^{(k)} + \sigma v^{(k)})\right)$

- The coefficient matrix $I + \sigma X X^T \in \mathbb{S}^n_{++}$ is constant over iterations. Therefore, one may save $\left(I + \sigma X X^T\right)^{-1}$ (or its decomposition) at the beginning to save computational cost
- Cost of $X(\beta^{(k)} + \sigma v^{(k)})$ < Cost of $X\beta^{(k)} + \sigma X v^{(k)}$

Subproblem-v

$$L_\sigma(y, v, \beta) = \frac{1}{2}\|y\|^2 - \langle y, Y\rangle + \delta_{B_\lambda}(v) + \frac{\sigma}{2}\|X^T y + v + \sigma^{-1}\beta\|^2 - \frac{1}{2\sigma}\|\beta\|^2$$

$$\begin{aligned}
&\arg\min_v \left\{\delta_{B_\lambda}(v) + \frac{\sigma}{2}\|X^T y + v + \sigma^{-1}\beta\|^2\right\} \\
= \;&\arg\min_v \left\{\delta_{B_\lambda}(v) + \frac{\sigma}{2}\|v - (-X^T y - \sigma^{-1}\beta)\|^2\right\} \\
= \;&\Pi_{B_\lambda}(-X^T y - \sigma^{-1}\beta)
\end{aligned}$$

In k-th iteration $v^{(k+1)} = \Pi_{B_\lambda}\left(-X^T y^{(k+1)} - \sigma^{-1}\beta^{(k)}\right)$

Recall that $B_\lambda = \{v \mid \|v\|_\infty \leq \lambda\}$. Thus the projection $v = \Pi_{B_\lambda}(s)$ can be computed elementwisely

$$v_i = \begin{cases} \lambda, & \text{if } s_i > \lambda \\ s_i, & \text{if } -\lambda \leq s_i \leq \lambda \\ -\lambda, & \text{if } s_i < -\lambda \end{cases}$$

ADMM framework

$$\min_{y,v} \quad \frac{1}{2}\|y\|^2 - \langle y, Y \rangle + \delta_{B_\lambda}(v) \quad \text{s.t.} \quad X^T y + v = 0$$

Algorithm (ADMM for dual of Lasso)

Choose $\sigma > 0$, $0 < \tau < \frac{1+\sqrt{5}}{2}$, $\beta^{(0)} \in \mathbb{R}^p$, $v^{(0)} \in B_\lambda$. Set $k \leftarrow 0$

repeat until convergence

$$y^{(k+1)} \leftarrow \left(I + \sigma X X^T\right)^{-1} \left(Y - X(\beta^{(k)} + \sigma v^{(k)})\right)$$

$$v^{(k+1)} \leftarrow \Pi_{B_\lambda}\left(-X^T y^{(k+1)} - \sigma^{-1}\beta^{(k)}\right)$$

$$\beta^{(k+1)} \leftarrow \beta^{(k)} + \tau\sigma\left(X^T y^{(k+1)} + v^{(k+1)}\right)$$

$$k \leftarrow k + 1$$

end(repeat)

return $y^{(k)}, v^{(k)}, \beta^{(k)}$

When n is large, updating $y^{(k+1)}$ is time-consuming

Primal vs. Dual Lasso

Primal Lasso

$$\min_{\beta} \quad \frac{1}{2}\|X\beta - Y\|^2 + \lambda\|\beta\|_1$$

Primal ADMM

$$\beta \leftarrow \left(\sigma I + X^T X\right)^{-1} \left(X^T Y + \sigma u - \xi\right)$$

$$u \leftarrow S_{\frac{\lambda}{\sigma}} \left(\beta + \sigma^{-1}\xi\right)$$

$$\xi \leftarrow \xi + \tau\sigma \left(\beta - u\right)$$

Use primal ADMM if $p < n$

Dual Lasso

$$\min_{y,v} \quad \frac{1}{2}\|y\|^2 - \langle y, Y\rangle + \delta_{B_\lambda}(v)$$

$$\text{s.t.} \quad X^T y + v = 0$$

Dual ADMM

$$y \leftarrow \left(I + \sigma X X^T\right)^{-1} (Y - X(\beta + \sigma v))$$

$$v \leftarrow \Pi_{B_\lambda} \left(-X^T y - \sigma^{-1}\beta\right)$$

$$\beta \leftarrow \beta + \tau\sigma \left(X^T y + v\right)$$

Use dual ADMM if $n < p$

ADMM in practice

$$\min_{y,z} f(y) + g(z) \text{ s.t. } Ay + Bz = c$$

$$L_\sigma(y,z,x) = f(y) + g(z) + \langle x, Ay + Bz - c\rangle + \frac{\sigma}{2}\|Ay + Bz - c\|^2$$

1. Choice of σ can greatly affect the practical convergence of ADMM:
   - σ too large: not enough emphasis on minimizing $f(y) + g(z)$
   - σ too small: not enough emphasis on feasibility $Ay + Bz = c$

   Theoretically, σ is fixed through the algorithm. In practice, one can tune σ according to the progresses of feasibilities. Roughly, $\|Ay + Bz - c\|$ being very small means σ is too large

ADMM in practice

$$\min_{y,z} f(y) + g(z) \text{ s.t. } Ay + Bz = c$$

$$L_\sigma(y, z, x) = f(y) + g(z) + \langle x, Ay + Bz - c\rangle + \frac{\sigma}{2}\|Ay + Bz - c\|^2$$

1. Choice of σ can greatly affect the practical convergence of ADMM:
   - σ too large: not enough emphasis on minimizing $f(y) + g(z)$
   - σ too small: not enough emphasis on feasibility $Ay + Bz = c$

   Theoretically, σ is fixed through the algorithm. In practice, one can tune σ according to the progresses of feasibilities. Roughly, $\|Ay + Bz - c\|$ being very small means σ is too large
2. Step length $0 < \tau < \frac{1+\sqrt{5}}{2}$. In practice, a larger step length generally results in faster convergence and common choices are $\tau = 1$ or $\tau = 1.618 \approx \frac{1+\sqrt{5}}{2}$
3. Transforming a problem into the form with 2-block separable structure that ADMM can handle is tricky and different forms can lead to different ADMM.

References i

E. J. Candès, X. Li, Y. Ma, and J. Wright.
Robust principal component analysis?
Journal of the ACM (JACM), 58(3):1–37, 2011.