Alternating Direction Method of Multipliers

参考讲义：

1. Neal Parikh, Stephen Boyd. Proximal Algorithms.
2. S. Boyd. NIPS Workshop on Optimization for Machine Learning, 12/16/11.
3. Yuan Zhong. Alternating Direction Method of Multipliers. PPT.
4. Liang Zhang. Alternating Direction Method of Multipliers. PPT.

Dual problem

convex equality constrained optimization problem

$$\begin{array}{ll} \text{minimize} & f(x) \\ \text{subject to} & Ax = b \end{array}$$

Lagrangian: $L(x, y) = f(x) + y^T(Ax - b)$

dual function: $g(y) = \inf_x L(x, y)$

dual problem: $\quad$ maximize $g(y)$

Dual ascent

$y^{k+1} = y^k + \alpha^k \nabla g(y^k)$

$\nabla g(y^k) = A\tilde{x} - b$, where $\tilde{x} = \operatorname{argmin}_x L(x, y^k)$

dual ascent method is

$$
\begin{aligned}
x^{k+1} &:= \operatorname{argmin}_x L(x, y^k) && // \ x\text{-minimization} \\
y^{k+1} &:= y^k + \alpha^k (Ax^{k+1} - b) && // \ \text{dual update}
\end{aligned}
$$

Dual decomposition

- suppose f is separable:

$$f(x) = f_1(x_1) + \cdots + f_N(x_N), \quad x = (x_1, \ldots, x_N)$$

- then L is separable in x: $L(x,y) = L_1(x_1,y) + \cdots + L_N(x_N,y) - y^T b$,

$$L_i(x_i, y) = f_i(x_i) + y^T A_i x_i$$

- x-minimization in dual ascent splits into N separate minimizations

$$x_i^{k+1} := \underset{x_i}{\operatorname{argmin}} \, L_i(x_i, y^k)$$

which can be carried out in parallel

- dual decomposition (Everett, Dantzig, Wolfe, Benders 1960–65)

$$
\begin{aligned}
x_i^{k+1} &:= \operatorname{argmin}_{x_i} L_i(x_i, y^k), \quad i = 1, \ldots, N \\
y^{k+1} &:= y^k + \alpha^k \left(\textstyle\sum_{i=1}^N A_i x_i^{k+1} - b\right)
\end{aligned}
$$

- scatter y^k; update x_i in parallel; gather $A_i x_i^{k+1}$

Augmented Lagrangian

$$L_\rho(x, y) = f(x) + y^T(Ax - b) + (\rho/2)\|Ax - b\|_2^2$$

method of multipliers

$$\begin{aligned} x^{k+1} &:= \operatorname*{argmin}_x L_\rho(x, y^k) \\ y^{k+1} &:= y^k + \rho(Ax^{k+1} - b) \end{aligned}$$

optimality conditions(KKT)

$$Ax^\star - b = 0, \qquad \nabla f(x^\star) + A^T y^\star = 0$$ (primal and dual feasibility)

since x^{k+1} minimizes $L_\rho(x, y^k)$

$$\begin{aligned} 0 &= \nabla_x L_\rho(x^{k+1}, y^k) \\ &= \nabla_x f(x^{k+1}) + A^T\left(y^k + \rho(Ax^{k+1} - b)\right) \\ &= \nabla_x f(x^{k+1}) + A^T y^{k+1} \end{aligned}$$

dual update $y^{k+1} = y^k + \rho(x^{k+1} - b)$ makes (x^{k+1}, y^{k+1}) *dual feasible*

primal feasibility achieved in limit: $Ax^{k+1} - b \to 0$

(compared to dual decomposition)

- *good news*: converges under much more relaxed conditions (f can be nondifferentiable, take on value $+\infty$, ...)
- *bad news*: quadratic penalty destroys splitting of the x-update, so can't do decomposition

Alternating direction method of multipliers

ADMM problem form (with f, g convex)

$$\begin{array}{ll} \text{minimize} & f(x) + g(z) \\ \text{subject to} & Ax + Bz = c \end{array}$$

$$L_\rho(x, z, y) = f(x) + g(z) + y^T(Ax + Bz - c) + (\rho/2)\|Ax + Bz - c\|_2^2$$

ADMM:

$$\begin{array}{lll} x^{k+1} & := & \operatorname{argmin}_x L_\rho(x, z^k, y^k) \qquad // \ x\text{-minimization} \\ z^{k+1} & := & \operatorname{argmin}_z L_\rho(x^{k+1}, z, y^k) \qquad // \ z\text{-minimization} \\ y^{k+1} & := & y^k + \rho(Ax^{k+1} + Bz^{k+1} - c) \qquad // \ \text{dual update} \end{array}$$

ADMM and optimality conditions

optimality conditions (for differentiable case):

- primal feasibility: $Ax + Bz - c = 0$
- dual feasibility: $\nabla f(x) + A^T y = 0, \quad \nabla g(z) + B^T y = 0$

since z^{k+1} minimizes $L_\rho(x^{k+1}, z, y^k)$ we have

$$\begin{aligned} 0 &= \nabla g(z^{k+1}) + B^T y^k + \rho B^T(Ax^{k+1} + Bz^{k+1} - c) \\ &= \nabla g(z^{k+1}) + B^T y^{k+1} \end{aligned}$$

Augmented Lagrangian with Proximal version ADMM

$$\begin{array}{ll} \text{minimize} & f(x) + g(z) \\ \text{subject to} & x - z = 0, \end{array}$$

$$L_\rho(x, z, y) = f(x) + g(z) + y^T(x - z) + (\rho/2)\|x - z\|_2^2,$$

$$\begin{aligned} x^{k+1} &:= \operatorname*{argmin}_x \left(f(x) + y^{kT}x + (\rho/2)\|x - z^k\|_2^2 \right) \\ z^{k+1} &:= \operatorname*{argmin}_z \left(g(z) - y^{kT}z + (\rho/2)\|x^{k+1} - z\|_2^2 \right) \\ y^{k+1} &:= y^k + \rho(x^{k+1} - z^{k+1}), \end{aligned}$$

$$\downarrow$$

$$\begin{aligned} x^{k+1} &:= \operatorname*{argmin}_x \left(f(x) + (\rho/2)\|x - z^k + (1/\rho)y^k\|_2^2 \right) \\ z^{k+1} &:= \operatorname*{argmin}_z \left(g(z) + (\rho/2)\|x^{k+1} - z - (1/\rho)y^k\|_2^2 \right) \\ y^{k+1} &:= y^k + \rho(x^{k+1} - z^{k+1}). \end{aligned}$$

$$\longrightarrow$$

$$\begin{aligned} x^{k+1} &:= \mathbf{prox}_{\lambda f}(z^k - u^k) \\ z^{k+1} &:= \mathbf{prox}_{\lambda g}(x^{k+1} + u^k) \\ u^{k+1} &:= u^k + x^{k+1} - z^{k+1}, \end{aligned}$$

With $u^k = (1/\rho)y^k$ and $\lambda = 1/\rho$, this is the proximal form of ADMM.

Application of ADMM

1. Constrained convex optimization

2. Lasso

Constrained convex optimization

- consider ADMM for generic problem

$$\begin{array}{ll} \text{minimize} & f(x) \\ \text{subject to} & x \in \mathcal{C} \end{array}$$

- ADMM form: take g to be indicator of $\mathcal{C}$

$$\begin{array}{ll} \text{minimize} & f(x) + g(z) \\ \text{subject to} & x - z = 0 \end{array}$$

- algorithm:

$$\begin{aligned} x^{k+1} &:= \operatorname*{argmin}_x \left(f(x) + (\rho/2)\|x - z^k + u^k\|_2^2 \right) \\ z^{k+1} &:= \Pi_{\mathcal{C}}(x^{k+1} + u^k) \\ u^{k+1} &:= u^k + x^{k+1} - z^{k+1} \end{aligned}$$

Lasso

- lasso problem:

$$\text{minimize} \quad (1/2)\|Ax - b\|_2^2 + \lambda\|x\|_1$$

- ADMM form:

$$\begin{array}{ll} \text{minimize} & (1/2)\|Ax - b\|_2^2 + \lambda\|z\|_1 \\ \text{subject to} & x - z = 0 \end{array}$$

- ADMM:

$$\begin{aligned} x^{k+1} &:= (A^TA + \rho I)^{-1}(A^Tb + \rho z^k - y^k) \\ z^{k+1} &:= S_{\lambda/\rho}(x^{k+1} + y^k/\rho) \\ y^{k+1} &:= y^k + \rho(x^{k+1} - z^{k+1}) \end{aligned}$$

$$\begin{aligned} x^{k+1} &:= \mathbf{prox}_{\lambda f}(z^k - u^k) \\ z^{k+1} &:= \mathbf{prox}_{\lambda g}(x^{k+1} + u^k) \\ u^{k+1} &:= u^k + x^{k+1} - z^{k+1}, \end{aligned}$$