Alternating Direction Method of Multipliers

Goran Banjac

Large-Scale Convex Optimization
ETH Zurich

April 28, 2020

Composite minimization problem

- consider the following convex optimization problem:

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

  where f and g are proper closed convex, $A \in \mathbb{R}^{p \times n}$, $B \in \mathbb{R}^{p \times m}$
- many convex problems can be represented in this form
- we cannot apply operator splitting methods directly, but we can apply them to the dual problem:

because operator splitting is used for unconstrainted problem

$$\text{minimize} \quad \underbrace{g^*(-B^T\mu)}_{p_1(\mu)} + \underbrace{f^*(-A^T\mu) + c^T\mu}_{p_2(\mu)}$$

ADMM

- DR splitting (with $\alpha = \frac{1}{2}$) applied to the dual problem reduces to

$$
\begin{aligned}
x^{k+1} &\in \underset{x}{\operatorname{argmin}} \left\{ f(x) + \frac{\rho}{2} \| Ax + Bz^k - c + \frac{1}{\rho} y^k \|_2^2 \right\} \\
z^{k+1} &\in \underset{z}{\operatorname{argmin}} \left\{ g(z) + \frac{\rho}{2} \| Ax^{k+1} + Bz - c + \frac{1}{\rho} y^k \|_2^2 \right\} \\
y^{k+1} &= y^k + \rho(Ax^{k+1} + Bz^{k+1} - c)
\end{aligned}
$$

- the method is known as the *alternating direction method of multipliers*
- note that the subproblems are not necessarily strongly convex
- the only conditions are that f and g are convex and that the subproblems have solutions
- if a solution exists and a certain constraint qualification holds, then (x^k, z^k) converges to a primal and y^k to a dual solution for any $\rho > 0$

Augmented Lagrangian

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

- consider the *augmented Lagrangian* of the problem

$$\mathcal{L}_\rho(x, z, y) \coloneqq f(x) + g(z) + y^T(Ax + Bz - c) + \frac{\rho}{2}\|Ax + Bz - c\|_2^2$$

- it can be seen as the standard Lagrangian for the equivalent problem

$$\begin{array}{ll} \text{minimize} & f(x) + g(z) + \frac{\rho}{2}\|Ax + Bz - c\|_2^2 \\ \text{subject to} & Ax + Bz = c \end{array}$$

- iteration of ADMM can be written as

$$\begin{aligned} x^{k+1} &\in \operatorname*{argmin}_x \mathcal{L}_\rho(x, z^k, y^k) \\ z^{k+1} &\in \operatorname*{argmin}_z \mathcal{L}_\rho(x^{k+1}, z, y^k) \\ y^{k+1} &= y^k + \rho \underbrace{(Ax^{k+1} + Bz^{k+1} - c)}_{\nabla_y \mathcal{L}_\rho(x^{k+1}, z^{k+1}, y^k)} \end{aligned}$$

looks like gradient ascent?
but it does enforce constraint on rau, positive then converges, so it is more powerful a Lagrangian splitting method.

Convergence analysis

- optimality conditions for the problem are given by

$$\begin{cases} 0 = Ax + Bz - c \\ 0 \in \partial f(x) + A^T y \\ 0 \in \partial g(z) + B^T y \end{cases}$$

- from the z-update, it follows that the last condition is always satisfied

$$0 \in \partial g(z^{k+1}) + B^T y^{k+1}$$

- from the x-update, we have

$$\rho A^T B(z^{k+1} - z^k) \in \partial f(x^{k+1}) + A^T y^{k+1}$$

- we define the following *optimality residuals*:

$$\boxed{\begin{aligned} r_{\text{prim}}^{k+1} &:= Ax^{k+1} + Bz^{k+1} - c \\ r_{\text{dual}}^{k+1} &:= \rho A^T B(z^{k+1} - z^k) \end{aligned}}$$

- if the problem has a solution, then r_{prim}^k and r_{dual}^k converge to zero
- ADMM is usually terminated when $\|r_{\text{prim}}^k\|$ and $\|r_{\text{dual}}^k\|$ are small

can use any norm, two-norm scaled by dimension, infinity norm consider all rows.

Extensions

we also don't want to use any positive matrix, R means a ellipsoidal shaped level. When R is diagonal then it works still because it still align with the box coordinates then it is component wise separable

$$x^{k+1} \in \underset{x}{\operatorname{argmin}} \left\{ f(x) + \frac{\rho}{2}\|Ax + Bz^k - c + \frac{1}{\rho}y^k\|_2^2 \right\}$$

$$z^{k+1} \in \underset{z}{\operatorname{argmin}} \left\{ g(z) + \frac{\rho}{2}\|Ax^{k+1} + Bz - c + \frac{1}{\rho}y^k\|_2^2 \right\}$$

$$y^{k+1} = y^k + \rho(Ax^{k+1} + Bz^{k+1} - c)$$

- the algorithm is derived from DR splitting for $\alpha = \frac{1}{2}$
- for $\alpha \in (0,1)$, Ax^{k+1} in z and y updates can be replaced with

over relaxiation and under relaxiation.

$$x_A^{k+1} = 2\alpha Ax^{k+1} - (1-2\alpha)Bz^k$$

- the penalty parameter ρ can be replaced by varying ρ^k or by a positive definite matrix $R \in \mathbb{S}_{++}^n$
- ADMM will converge even when the x- and z-minimization steps are not carried out exactly provided that minimization errors are summable

when minimize x and z is difficult and we just can get a "good enough" solution instead of an exact solution.

Constrained minimization with ADMM

- consider the following convex optimization problem

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

- applying ADMM to the equivalent problem

$$\begin{array}{ll} \text{minimize} & f(x) + \mathcal{I}_{\mathcal{C}}(z) \\ \text{subject to} & Ax - z = 0 \end{array}$$

reduces to the following iteration

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

Solving feasibility problems with ADMM

- consider the following convex feasibility problem

$$\begin{array}{ll} \text{find} & x \\ \text{subject to} & x \in \mathcal{C} \cap \mathcal{D} \end{array}$$

- applying ADMM to the equivalent problem

$$\begin{array}{ll} \text{minimize} & \mathcal{I}_{\mathcal{C}}(x) + \mathcal{I}_{\mathcal{D}}(z) \\ \text{subject to} & x - z = 0 \end{array}$$

  reduces to the following iteration

$$\begin{aligned} x^{k+1} &= \Pi_{\mathcal{C}}\left(z^k - y^k\right) \\ z^{k+1} &= \Pi_{\mathcal{D}}\left(x^{k+1} + y^k\right) \\ y^{k+1} &= y^k + x^{k+1} - z^{k+1} \end{aligned}$$

Lasso

- ℓ_1-regularized linear regression or *lasso* is given by

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

  where $\lambda > 0$ is a regularization parameter

- ADMM for solving the lasso problem has the following form

$$\begin{aligned} x^{k+1} &= (A^T A + \rho I)^{-1}\left(A^T b + \rho z^k - y^k\right) \\ z^{k+1} &= \mathrm{soft}_{\lambda/\rho}\left(x^{k+1} + \tfrac{1}{\rho} y^k\right) \\ y^{k+1} &= y^k + \rho(x^{k+1} - z^{k+1}) \end{aligned}$$

  where soft_κ is the *soft thresholding operator* given by

$$\mathrm{soft}_\kappa(a) = \begin{cases} a - \kappa & a > \kappa \\ 0 & |a| < \kappa \\ a + \kappa & a < -\kappa \end{cases}$$

Conic programs

general problems, we need a
general/commential method

- consider the *primal-dual pair* of convex conic programs:

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax + s = b \\ & \boxed{(x, s) \in \mathbb{R}^n \times \mathcal{K}} \end{array} \qquad \begin{array}{ll} \text{maximize} & b^T y \\ \text{subject to} & -A^T y + r = c \\ & \boxed{(r, y) \in \{0\}^n \times \mathcal{K}^*} \end{array}$$

- optimality conditions for the conic program are given by

$$Ax + s = b, \quad s \in \mathcal{K}, \quad A^T y + c = r, \quad r = 0, \quad \underline{y \in \mathcal{K}^*, \quad y^T s = 0}$$

- the complementary slackness condition can be replaced by the zero duality gap condition

$$0 = y^T s = y^T (b - Ax) = y^T b - (A^T y)^T x = c^T x + b^T y$$

- the optimality conditions then take the following form:

$$\begin{bmatrix} r \\ s \\ 0 \end{bmatrix} = \begin{bmatrix} 0 & A^T \\ -A & 0 \\ c^T & b^T \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} c \\ b \\ 0 \end{bmatrix}, \quad (r, s, x, y) \in \{0\}^n \times \mathcal{K} \times \mathbb{R}^n \times \mathcal{K}^*$$

Infeasibility conditions

- if there exists some $\bar{y}$ such that

$$A^T\bar{y} = 0, \quad b^T\bar{y} < 0, \quad \text{and} \quad \boxed{\bar{y} \in \mathcal{K}^*,}$$

  then the primal conic program is infeasible
- proof: the problem is infeasible if and only if the sets $\mathcal{S} := \{b - Ax \mid x \in \mathbb{R}^n\}$ and $\mathcal{K}$ do not intersect
- due to the conditions above, we have

$$\sup_{s\in\mathcal{S}} \left\{s^T\bar{y}\right\} = \sup_{x} \left\{(b - Ax)^T\bar{y}\right\} = b^T\bar{y} - \inf_{x} \left\{x^T(A^T\bar{y})\right\} < 0$$

$$\boxed{\inf_{z\in\mathcal{K}} \left\{z^T\bar{y}\right\} = 0}$$

derived because of definition of dual cone.

  which means that the hyperplane $\left\{z \mid \bar{y}^T z = 0\right\}$ separates $\mathcal{S}$ and $\mathcal{K}$
- similarly, if there exists some $\bar{x}$ such that

$$-A\bar{x} \in \mathcal{K} \quad \text{and} \quad c^T\bar{x} < 0,$$

  then the dual conic program is infeasible

Homogeneous self-dual embedding

- the *homogeneous self-dual embedding (HSDE)* is the following set of equations and inclusions:

$$\underbrace{\begin{bmatrix} r \\ s \\ \kappa \end{bmatrix}}_{v} = \underbrace{\begin{bmatrix} 0 & A^T & c \\ -A & 0 & b \\ -c^T & -b^T & 0 \end{bmatrix}}_{Q} \underbrace{\begin{bmatrix} x \\ y \\ \tau \end{bmatrix}}_{u}, \quad \begin{array}{l} (r, s, \kappa) \in \{0\}^n \times \mathcal{K} \times \mathbb{R}_+ \\ (x, y, \tau) \in \underbrace{\mathbb{R}^n \times \mathcal{K}^* \times \mathbb{R}_+}_{\mathcal{C}} \end{array}$$

complementary condition

- Q is skew-symmetric, which implies $u^T v = 0$ and thus $\tau\kappa = 0$
- any solution to the HSDE falls into one of three cases:
  - if $\tau > 0$ and $\kappa = 0$, then $(x/\tau, s/\tau, y/\tau)$ is a primal-dual solution
  - if $\tau = 0$ and $\kappa > 0$, then the problem is primal or dual infeasible
  - if $\tau = 0$ and $\kappa = 0$, then nothing can be concluded about the problem
- observe that zero is always a solution to HSDE

Splitting conic solver (SCS)

- HSDE can be expressed as the following feasibility problem:

$$
\begin{array}{ll}
\text{find} & (u,v) \\
\text{subject to} & v = Qu \\
& (u,v) \in \mathcal{C} \times \mathcal{C}^*
\end{array}
$$

- applying ADMM to the problem gives the following algorithm:

$$
\begin{aligned}
\tilde{u}^{k+1} &= \boxed{(I+Q)^{-1}} \left(u^k - v^k\right) \\
u^{k+1} &= \Pi_{\mathcal{C}} \left(\tilde{u}^{k+1} + v^k\right) \\
v^{k+1} &= v^k + \tilde{u}^{k+1} - u^{k+1}
\end{aligned}
$$

- since Q is skew-symmetric, the matrix $(I+Q)$ is always invertible
- projecting onto $\mathcal{C}$ is as difficult as projecting onto $\mathcal{K}$
- in each iteration k we have: $u^k \in \mathcal{C}$ and $v^k \in \mathcal{C}^*$
- the equality constraint holds asymptotically: $Qu^k - v^k \to 0$
- the algorithm is used in the open-source solver SCS

Problems with quadratic objective

- consider the following optimization problem:

$$\begin{array}{ll}\text{minimize} & \frac{1}{2}x^TPx + q^Tx \\ \text{subject to} & Ax \in \mathcal{C}\end{array}$$

  with $P \in \mathbb{S}^n_+$, $A \in \mathbb{R}^{m\times n}$, and $\mathcal{C} \subset \mathbb{R}^m$ a nonempty closed convex set
- we assume that we can efficiently project onto $\mathcal{C}$
- when $\mathcal{C} = [l, u]$, the problem reduces to a QP
- a QP can be reformulated as an SOCP and then solved with SCS
- solving a QP directly is usually more efficient

Optimality and infeasibility conditions

support function of the set c
S_C(y) = sup_z\inC z'y

- primal-dual pair of the convex problem with quadratic objective

$$\begin{array}{lll} \text{minimize} & \frac{1}{2}x^T P x + q^T x & \qquad \text{maximize} & -\frac{1}{2}x^T P x - S_{\mathcal{C}}(y) \\ \text{subject to} & Ax = z & \qquad \text{subject to} & Px + A^T y = -q \\ & z \in \mathcal{C} & & y \in (\mathcal{C}^\infty)^\circ \end{array}$$

we can add it explicitly, otherwise the objective would become negative infinity.

- optimality conditions

for example, if Ax=z z<=b, C is a level set, then Nc(z)=z, Sc(y)=

$$Ax - z = 0, \quad Px + q + A^T y = 0, \quad z \in \mathcal{C}, \quad \text{and} \quad y \in N_{\mathcal{C}}(z)$$

normal cone such that y*(x-z)<=0 for all x in C;

- sufficient primal infeasibility conditions

$$A^T \bar{y} = 0 \quad \text{and} \quad S_{\mathcal{C}}(\bar{y}) < 0,$$

or y\in partial C(z), which is the Fermat's rule

- sufficient dual infeasibility conditions

$$P\bar{x} = 0, \quad A\bar{x} \in \mathcal{C}^\infty, \quad \text{and} \quad q^T \bar{x} < 0,$$

ADMM for problems with quadratic objective

- we can reformulate the problem into the following form:

$$\begin{array}{ll} \text{minimize} & \frac{1}{2}\tilde{x}^T P\tilde{x} + q^T\tilde{x} + \mathcal{I}_{Ax=z}(\tilde{x},\tilde{z}) + \mathcal{I}_{\mathcal{C}}(z) \\ \text{subject to} & (\tilde{x},\tilde{z}) = (x,z) \end{array}$$

- applying ADMM to the problem gives the following algorithm:

$$\begin{aligned} (x^{k+1},\tilde{z}^{k+1}) &= \underset{(x,\tilde{z}):Ax=\tilde{z}}{\operatorname{argmin}} \left\{ \tfrac{1}{2}x^T P x + q^T x + \boxed{\tfrac{\rho}{2}\|x - x^k\|_2^2} + \tfrac{\rho}{2}\|\tilde{z} - z^k + \tfrac{1}{\rho}y^k\|_2^2 \right\} \\ z^{k+1} &= \Pi_{\mathcal{C}}\left(\tilde{z}^{k+1} + \tfrac{1}{\rho}y^k\right) \\ y^{k+1} &= y^k + \rho(\tilde{z}^{k+1} - z^{k+1}) \end{aligned}$$

therefore it becomes unique.
(strongly convex and invertible)

- in each iteration k we have: $z^{k+1} \in \mathcal{C}$ and $y^{k+1} \in N_{\mathcal{C}}(z^{k+1})$
- when the problem has a solution, then the following optimality residuals converge to zero asymptotically:

$$r_{\text{prim}}^k = Ax^k - z^k, \quad r_{\text{dual}}^k = Px^k + q + A^T y^k$$

- the algorithm is used in the open-source solver OSQP

Solving the linear system

- minimizing the equality-constrained QP in ADMM reduces to solving the following linear system

$$\underbrace{\begin{bmatrix} P+\rho I & A^T \\ A & -\frac{1}{\rho} I \end{bmatrix}}_{K} \begin{bmatrix} x^{k+1} \\ \nu^{k+1} \end{bmatrix} = \begin{bmatrix} \rho x^k - q \\ z^k - \frac{1}{\rho} y^k \end{bmatrix}$$

- $\tilde{z}^{k+1}$ can be recovered as $\tilde{z}^{k+1} = z^k + \frac{1}{\rho}(\nu^{k+1} - y^k)$
- since the matrix is quasi-definite, it can be factored as $K = LDL^T$, where L is a lower triangular and D a diagonal matrix
- eliminating ν^{k+1} from the linear system, it reduces to

$$\underbrace{(P + \rho I + \rho A^T A)}_{K'} x^{k+1} = \rho x^k - q + A^T(\rho z^k - y^k)$$

- $\tilde{z}^{k+1}$ can then be recovered as $\tilde{z}^{k+1} = A x^{k+1}$
- since $K' \succ 0$, we can solve the linear system using the CG method

Infeasibility detection

- even when the pair (x^k, y^k) does not converge, the difference between consecutive iterates always converges:

$$(x^{k+1} - x^k, y^{k+1} - y^k) \to (\delta x, \delta y)$$

- when the problem is primal infeasible, then

$$A^T \delta y = 0 \quad \text{and} \quad S_{\mathcal{C}}(\delta y) < 0$$

- similarly, when the problem is dual infeasible, then

$$P\delta x = 0, \quad A\delta x \in \mathcal{C}^\infty, \quad \text{and} \quad q^T \delta x < 0$$

- therefore, even with no HSDE, ADMM can detect infeasible problems
- the termination criteria are implemented by checking that $(y^{k+1} - y^k)$ and $(x^{k+1} - x^k)$ satisfy the above conditions (approximately)

Preconditioning problem data

- a known limitation of ADMM is that it can sometimes converge slowly
- introducing $\hat{x} = S^{-1}x$, the problem can be reformulated as

$$\begin{array}{ll} \text{minimize} & \frac{1}{2}\hat{x}^T\hat{P}\hat{x} + \hat{q}^T\hat{x} \\ \text{subject to} & \hat{A}\hat{x} \in \mathcal{C} \end{array}$$

  with $\hat{P} = SPS$, $\hat{q} = Sq$, and $\hat{A} = AS$
- the change of variables effectively modifies the problem data
- the procedure is called *data scaling* or *preconditioning*
- selecting a suitable scaling matrix S can improve the convergence rate of ADMM
- there exist effective heuristics for finding a good preconditioner