Lecture 17

Last time

- Douglas-Rachford
- Concensus optimization

Today

- Alternating Directions Method of Multipliers
- Examples

Alternating Directions Method of Multipliers

Following the general idea of DRM we consider a more "splitted template":

$$p^* = \begin{cases} \min & f(x) + g(z) \\ \text{s.t.} & Ax + Bz = c \end{cases}$$

where X, Z are Euclidean spaces $f: X \to \overline{\mathbb{R}}$, $g: Z \to \overline{\mathbb{R}}$ are closed, convex and proper, the maps $A: X \to \mathbb{R}^m$, $B: Z \to \mathbb{R}^m$ are linear, and $c \in \mathbb{R}^m$.

The augmented Lagrangian in this case corresponds to

$$\bar{L}(x, z; \lambda)$$

$$= f(x) + g(z) + \lambda^T(Ax + Bz - c) + \frac{\alpha}{2}\|Ax + Bz - c\|^2$$

Completing the square

$$= f(x) + g(z) + \frac{\alpha}{2}\|Ax + Bz - c + \underbrace{\tfrac{1}{\alpha}\lambda}_{u}\|^2 - \frac{\alpha}{2}\|\underbrace{\tfrac{1}{\alpha}\lambda}_{u}\|^2$$

$$=: \hat{L}(x, z; u).$$

Just as we did before, it is easy to show

$$p^* = \inf_{x, z} \sup_{u \in \mathbb{R}^m} \hat{L}(x, z; u)$$

This naturally suggests an algorithm where we alternate between minimizing for x and z, and then maximize for u:

ADMM

Input: x_0, y_0, u_0

Loop $k \geq 0$:

$x_{k+1} \leftarrow \underset{x}{\operatorname{argmin}} \hat{L}(x, z_k; u_k)$

$z_{k+1} \leftarrow \underset{z}{\operatorname{argmin}} \hat{L}(x_{k+1}, z; u_k)$

$u_{k+1} \leftarrow u_k + (Ax_{k+1} + Bz_{k+1} - c)$

This is akin to ALM, but now we are "splitting" the objective into two variables.

Remark Just as with ALM, the x_{k+1}, z_{k+1} updates might not be well-defined. They can be taken to be any minimizer. To analyze this algorithm let's try to reformula te the problem as that of finding the zero of the sum of operators. For this, notice that the Fenchel dual of p^* is ← (why?)

$$d^* = \sup_y \; c^T y - f^*(A^* y) - g^*(B^* y).$$

If a constrained qualification condition holds, i.e., $c \in \operatorname{int}\{A \operatorname{dom} f + B \operatorname{dom} g\}$, then $p^* = d^*$. Moreover, if a minimizer (x, z) of p^* is attained, then a minimizer y of the dual exist and it satisfies:

$$x \in \partial f^*(A^* y) \text{ and } z \in \partial g^*(B^* y)$$

Therefore we get that

$$y \text{ is dual optimal}$$
$$\Longleftrightarrow$$
$$c \in A\,\partial f^*(A^* y) + B\,\partial g^*(B^* y)$$
$$\Longleftrightarrow$$
$$0 \in \underbrace{A\,\partial f^*(A^* y)}_{T} + \underbrace{B\,\partial g^*(B^* y) - c}_{S}$$

Claim: S and T are monotone.

Therefore, we could apply DRM.

ADMM via DRM

Input: $y_0 \in \mathbb{R}^m$, R_S, R_T

Loop $k \geq 0$:

When is T maximal?

- $w_{k+1} \leftarrow R_T(y_k)$
- $\hat{w}_{k+1} \leftarrow R_S(2w_{k+1} - y_k)$
- $y_{k+1} \leftarrow y_k - \hat{w}_{k+1} + w_{k+1}$

Thanks to the theory we developed, we know that $y_k \to y^*$ a solution to the dual problem. In turn, ADMM and ADMM via DRM are essentially the same algorithm.

To see this, note that

$$w = R_{\overline{T}}(y)$$

$$\Leftrightarrow \quad y \in (I + \alpha T)w$$

$$\Leftrightarrow \quad y - w \in \alpha A \partial f^*(A^* w)$$

$$\Leftrightarrow \quad \exists\, x \in \partial f^*(A^* w) \text{ with } y - w = \alpha A x$$

$$\Leftrightarrow \quad A^* w \in \partial f(x) \text{ with } w = y - \alpha A x$$

$$\Leftrightarrow A^*(y - \alpha Ax) \in \partial f(x) \text{ with } w = y - \alpha Ax$$

$$\Leftrightarrow 0 \in \partial f(x) - A^*y + \alpha A^*Ax \text{ with } w = y - \alpha Ax$$

$$\Leftrightarrow x \in \operatorname{argmin} f(\cdot) - \langle y, A\cdot\rangle + \frac{\alpha}{2}\|A\cdot\|^2$$

Implies Ax is unique

$$\text{with } w = y - \alpha A x$$

What we have discovered is that if we can minimize

$$(\text{😀}) \quad f(\cdot) - \langle y, A\cdot\rangle + \frac{\alpha}{2}\|A\cdot\|^2$$

then, we can compute

$$R_T(y) = y - \alpha Ax.$$

A similar argument applies for R_S.

In turn, after the change of variables $y = \alpha(u + c - Bx)$ one can show the x-step in ADMM is equivalent to minimizing (ö). Some extra dry algebra yields that the two algorithms are equivalent after this change of basis.

This argument yields convergence of u_k. Under additinal conditions x_k, and z_k also converge, but we will not cover that here.

Examples

There are a number of important examples that we can cover if we take

$$\min_{\substack{x \in \mathbb{R}^n \\ z \in \mathbb{R}^n}} \quad f(x) + g(z)$$
$$x = z$$
$$(\ B = -I, \quad c = 0 \ )$$

In this case the x, z updates compute proximals.

▷ Intersections

If we take $f = \iota_{C_1}$, $g = \iota_{C_2}$ with C_1, C_2 convex, closed, non empty sets.

Then ADMM recovers the so-called Dykstra's alternating projection method.

▷ Compressed sensing

If we set $f = \|\cdot\|_1$ and $g = \tau_{\{x \mid Ax = b\}}(\cdot)$, then ADMM recovers the popular Basis pursuit method.

▷ LASSO

If we set $f = \|A\cdot - b\|_2^2$ and $g = \lambda\|\cdot\|_1$ ADMM solves the famous LASSO problem.

▷ Linear Programming

If we set

$$f = \langle c, \cdot \rangle + \tau_{\{x \mid Ax = b\}}(\cdot)$$

$$g = \tau_{\mathbb{R}^n_+}(\cdot)$$

then, we can solve linear programming.

- **Conic programming.**

More generally if we set $g = \iota_K(\cdot)$ with K a cone, we obtain an algorithm for conic programming.

Remark: ADMM is the backbone of popular solvers such as OSQP and SCS.