Lecture 23 Augmented Lagrangian and ADMM

Can Li

ChE 597: Computational Optimization
Purdue University

Augmented Lagrangian and ADMM

- These methods can be seen as an extension of Lagrangian decomposition.
- In LD, we focused on MILP problems.
- Augmented Lagrangian and ADMM for MILPs are not as mature (under active research). We will focus on continuous convex problems in this lecture.

Preliminary: Dual Ascent

Consider the linearly constrained convex optimization problem,

$$\min \quad f(x) \text{ s.t. } Ax = b$$

- Lagrangian: $L(x, y) = f(x) + y^T(Ax - b)$
- Dual function: $g(y) = \min_x L(x, y)$, dual problem: $\max_y g(y)$

Dual ascent algorithm (subgradient method when g is nonsmooth):

$$x^{k+1} := \text{argmin}_x \, L\left(x, y^k\right)$$
$$y^{k+1} := y^k + \alpha^k \left(Ax^{k+1} - b\right)$$

where $\alpha^k > 0$ is the step size.

Lagrangian Decomposition (Dual Decomposition)

Suppose that objective f is separable:

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

then Lagrangian $L(x, y)$ can be written as:

$$L(x, y) = \sum_{i=1}^{N} L_i(x_i, y) = \sum_{i=1}^{N} \left(f_i(x_i) + y^T A_i x_i - (1/N) y^T b \right),$$

which is also separable in x and where A is conformably partitioned such that $Ax = \sum_{i=1}^{N} A_i x_i$.
This means that the x-minimization in dual ascent splits into into N separate minimizations,

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

which can be carried out in parallel.

To solve the dual problem, the iterates can be written as

$$x_i^{k+1} := \text{argmin}_{x_i} L_i \left(x_i, y^k \right), \quad i = 1, \ldots, N$$

$$y^{k+1} := y^k + \alpha^k \left(\sum_{i=1}^{N} A_i x_i^{k+1} - b \right)$$

- In this case, we refer to the dual ascent method as *dual decomposition*.
- Each iteration requires following operarions: scatter y^k; update x_i in parallel; gather $A_i x_i^{k+1}$

Dual Decomposition is an old concept in optimization that dates back to the early 1960s. It is generally used to solve large problems. However, it works only under many assumptions and is often slow.

Issues with Dual Decomposition

- Convergence can be slow.
- It may not end up with a solution that satisfies the primal constraint $Ax = b$ even with strong duality. This happens when the solution to $\min_x L(x, y^*)$ is not unique.

Augmented Lagrangian

For the optimization problem considered earlier, the augmented Lagrangian is (with $\rho > 0$)

$$L_\rho(x, y) := \underbrace{f(x) + y^T(Ax - b)}_{\text{Lagrangian}} + \underbrace{\frac{\rho}{2}\|Ax - b\|_2^2}_{\text{'augmentation'}}.$$

- Where $\rho > 0$ is called the penalty parameter

It can also be viewed as as the (unaugmented) Lagrangian associated with the problem

$$\begin{array}{ll} \text{minimize} & f(x) + (\rho/2)\|Ax - b\|_2^2 \\ \text{subject to} & Ax = b. \end{array}$$

This problem is clearly equivalent to the original problem, since for any feasible x the term added to the objective is zero.

Method of Multipliers

Applying dual ascent to the modified problem yield the below iterates,

$$x^{k+1} := \underset{x}{\operatorname{argmin}} L_\rho\left(x, y^k\right)$$

$$y^{k+1} := y^k + \alpha^k\left(Ax^{k+1} - b\right)$$

This is known as method of multipliers. Now, assuming f to be differentiable, we look at choosing α^k.
The KKT optimality conditions for our original problem are:

$$Ax^* - b = 0 \quad \text{(primal feasibility)}, \quad \nabla f\left(x^*\right) + A^T y^* = 0 \quad \text{(stationarity)}$$

For our iterates, by definition x^{k+1} minimizes $L_\rho\left(x, y^k\right)$, so

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

But since, we need the iterates to satisfy stationarity i.e. $\nabla_x f\left(x^{k+1}\right) + A^T y^{k+1} = 0$. This motivates $\alpha^k = \rho$.

Practical Algorithm for Method of Multipliers

- Finding the exact minimizer for $L_\rho\left(x, y^k\right)$ at each step might not be practical. Instead, we obtain iterates with increasing approximation and terminate based on a global criterion.
- We do so by increasing ρ to equivalently turn the augmentation to behave like an indicator function of the feasible set.

Algorithm 1: Algorithm for Equality Constraints

Data: Given $\rho_0 > 0$, tolerance $\tau_0 > 0$, starting points x_s^0 and y^0.
Result: Find an approximate solution x^k.

for $k = 0, 1, 2, \ldots$ **do**
  Find an approx minimizer x^k of $L_{\rho_k}(x, y^k)$, starting at x_s^k, and terminating when $\|\nabla_x L_{\rho_k}(x, y^k)\| \leq \tau_k$.
  if *final convergence test satisfied* **then**
    STOP with approximate solution x^k.
  Update Lagrange multipliers y^{k+1} using dual update step.
  Choose new penalty parameter $\rho_{k+1} \in (\rho_k, \infty)$.
  Set starting point for the next iteration to $x_s^{k+1} = x^k$.

Method of Multipliers for Inequality Constraints

Consider the optimization problem with linear inequality constraints:

$$\min \quad f(x) \text{ s.t. } Ax \geq b$$

- Let $c(x) = Ax - b$, we can convert it to a problem with equality constraints by introducing slack variables s to replace the inequalities. Equivalent formulation is:

$$\min \quad f(x) \text{ s.t. } c(x) = s, \quad s \geq 0$$

- For $y \geq 0$, the augmented Lagrangian is convex w.r.t x and s (convex quadratic w.r.t. x and s).

$$L_\rho(x, s, y) := f(x) - y^T(c(x) - s) + \frac{\rho}{2}\|c(x) - s\|_2^2, \text{ s.t. } s \geq 0$$

For fixed $y \geq 0$, the optimality conditions are $\nabla_x L_\rho(x, s, y) = 0$, $\nabla_s L_\rho(x, s, y) = 0$

- We perform explicit minimization by $\nabla_s L_\rho(x, s, y) = 0$. Which upon solving gives, $s_{i,opt} = c_i(x) - \frac{y_i}{\rho}$.
- If this unconstrained minimizer ($s_{i,opt}$) is smaller than the lower bound of 0, then since $L_\rho(x, s, y)$ is convex in s, the optimal value of s_i is 0. Conclusively we write:

$$s_{i,opt} = \max(c_i(x) - \frac{y_i}{\rho}, 0)$$

- Substituting, we can write -$y_i(c_i(x) - s_i) + \frac{\rho}{2}(c_i(x) - s_i)^2$ as:

$$= \begin{cases} -y_i c_i(x) + \frac{\rho}{2} c_i^2(x) & \text{if } c_i(x) - y_i/\rho \leq 0, \\ -\frac{1}{2\rho} y_i^2 & \text{otherwise.} \end{cases}$$

- This extension renders Lagrangian to depend only on x, y and ρ. Hence, algorithm proposed earlier for equality constrained problem can be used. With one modification (since $y \geq 0$):

$$y_i^{k+1} = \max(y_i^k - \rho c_i(x^k), 0)$$

- As the method of multipliers proceeds, the primal residual $Ax^{k+1} - b$ converges to zero, yielding optimality.
- Adding the penalty term helps to robustify dual ascent, i.e., convergence under much more relaxed conditions.
  (f can be non-differentiable, take on value $+\infty$, . . .)
- However, improved convergence properties comes at a cost, quadratic penalty destroys splitting of the x-update when f is separable. Thus, cannot be used for decomposition.
- We next see alternating direction method of multipliers (ADMM), which aims to blend the decomposability of dual ascent with the superior convergence properties of the method of multipliers.
- Therefore, ADMM proposed by Gabay, Mercier, Glowinski, Marrocco in 1976 is also known as "robust dual decomposition" or "decomposable method of multipliers".

Alternating Direction Method of Multipliers (ADMM)

- Consider now problems with a separable objective of the form

$$\min_{(x,z)} f(x) + h(z) \quad \text{s.t.} \quad Ax + Bz = c,$$

for which the augmented Lagrangian is

$$L_\rho(x, z, y) := f(x) + h(z) + y^T(Ax + Bz - c) + \frac{\rho}{2}\|Ax - Bz - c\|_2^2.$$

- Standard AL would minimize $L_\rho(x, z, y)$ w.r.t. (x, z) jointly. However, since coupled in the quadratic term, separability is lost.
- In ADMM, minimize over x and z separately and sequentially:

$$x^{k+1} = \operatorname*{argmin}_x L_\rho(x, z^k, y^k)$$
$$z^{k+1} = \operatorname*{argmin}_z L_\rho(x^{k+1}, z, y^k)$$
$$y^{k+1} = y^k + \rho\left(Ax^{k+1} + Bz^{k+1} - c\right)$$

Features of ADMM

Main features of ADMM:

- Does one cycle of block-coordinate descent in (x, z).
- The minimizations over x and z add only a quadratic term to f and h, respectively. Usually does not alter the cost much.
- Can perform the (x, z) minimizations inexactly.
- Can add explicit (separated) constraints: $x \in \Omega_x, z \in \Omega_z$.
- Recent applications to federated learning, distributed optimization, sparse principal components, image processing...

ADMM and optimality conditions

Optimality conditions (for differentiable case):

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

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

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

- So with ADMM dual variable update, $\left(x^{k+1}, z^{k+1}, y^{k+1}\right)$ satisfies second stationarity condition (since step-size, α^k for dual update is chosen as ρ).
- Primal feasibility and first stationarity condition are achieved as $k \to \infty$.

Convergence

Many convergence results exist for ADMM under various assumptions. We limit ourselves to a basic, yet very general, result for introductory purposes.

- Assumption 1: The (extended-real-valued) functions $f : \mathbf{R}^n \to \mathbf{R} \cup \{+\infty\}$ and $g : \mathbf{R}^m \to \mathbf{R} \cup \{+\infty\}$ are closed, proper, and convex.
- Assumption 2: The unaugmented Lagrangian L_0 has a saddle point.

Under these assumptions,

- *Residual convergence*: $Ax^k + Bz^k - c \to 0$, i.e., the iterates approach feasibility.
- *Objective convergence*: $f\left(x^k\right) + g\left(z^k\right) \to p^*$ as $k \to \infty$, i.e., the objective function approaches the optimal value.
- *Dual variable convergence*: $y^k \to y^*$ as $k \to \infty$, where y^* is a dual optimal point.

Consensus Optimization

Consider the case with a single global variable, with the objective split into N parts:

$$\text{minimize} \sum_{i=1}^{N} f_i(x)$$

- e.g., f_i is the loss function for ith block of training data.

ADMM form:

$$\begin{array}{ll} \text{minimize} & \sum_{i=1}^{N} f_i(x_i) \\ \text{subject to} & x_i - z = 0 \quad \forall i = 1 \ldots, N \end{array}$$

- x_i are *local variables*.
- z is the *global variable*.
- $x_i - z = 0$ are consistency or consensus constraints (also called *global consensus problem*).
- We can add regularization using a $g(z)$ term.

Consensus Optimization via ADMM

- Lagrangian is written as:

$$L_\rho(x, z, y) = \sum_{i=1}^{N} \left(f_i(x_i) + y_i^T (x_i - z) + (\rho/2) \|x_i - z\|_2^2 \right)$$

ADMM iterated can be motivated as:

$$x_i^{k+1} := \underset{x_i}{\text{argmin}} \left(f_i(x_i) + y_i^{kT} \left(x_i - z^k \right) + (\rho/2) \left\| x_i - z^k \right\|_2^2 \right)$$

$$z^{k+1} := \frac{1}{N} \sum_{i=1}^{N} \left(x_i^{k+1} + (1/\rho) y_i^k \right)$$

$$y_i^{k+1} := y_i^k + \rho \left(x_i^{k+1} - z^{k+1} \right)$$

Algorithms Related to ADMM

- Operator splitting methods
  (Douglas, Peaceman, Rachford, Lions, Mercier, ... 1950s, 1979)
- Proximal point algorithm (Rockafellar 1976)
- Dykstra's alternating projections algorithm (1983)
- Spingarn's method of partial inverses (1985)
- Rockafellar-Wets progressive hedging (1991)
- Proximal methods (Rockafellar, many others, 1976-present)
- Bregman iterative methods (2008-present)

Most of these are special cases of the proximal point algorithm.

Augmented Lagrangian and ADMM for MILP

Augmented Lagrangian and ADMM for MILP are still under active research. Here are some recent references.

- Feizollahi, M. J., Ahmed, S., & Sun, A. (2017). Exact augmented Lagrangian duality for mixed integer linear programming. Mathematical Programming, 161, 365-387.
- Sun, K., Sun, M., & Yin, W. (2024). Decomposition Methods for Global Solution of Mixed-Integer Linear Programs. SIAM Journal on Optimization, 34(2), 1206-1235.
- Knueven, B., Mildebrath, D., Muir, C., Siirola, J. D., Watson, J. P., & Woodruff, D. L. (2023). A parallel hub-and-spoke system for large-scale scenario-based optimization under uncertainty. Mathematical Programming Computation, 15(4), 591-619.

References

- S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction methods of multipliers," Foundations and Trends in Machine Learning, 3, pp. 1-122, 2011.
- Numerical Optimization (Jorge Nocedal and Stephen J. Wright), Springer, 2006.
- `https://pages.cs.wisc.edu/~swright/nd2016/IMA_augmentedLagrangian.pdf`