10-725/36-725: Convex Optimization **Fall 2015**

Lecture 20: November 4

Lecturer: Ryan Tibshirani *Scribes: Vineet Jain*

Note: *LaTeX template courtesy of UC Berkeley EECS dept.*

20.1 Last time: Coordinate descent

Consider the problem

$$\min_x f(x)$$

where $f(x) = g(x) + \sum_{i=1}^n h_i(x_i)$, with g convex and differentiable and each h_i convex.

Coordinate descent: let $x^{(0)} \in \mathbb{R}$ and repeat for $k = 1, 2, \ldots$

$$x^{(k)} = \underset{x_i}{\operatorname{argmin}} f\left(x_1^{(k)}, \ldots, x_{i-1}^{(k)}, x_i, x_{i+1}^{(k-1)}, \ldots, x_n^{(k-1)}\right), i = 1, 2, \ldots, n$$

A few points to note:

- This method can be applied to blocks of variables instead of individual variabels.
- For every update $x_i^{(k)}$, we use the 'most recent information' available for the other variables.
- Relatively simple to implement and can achieve state-of-the-art

20.2 Reminder: Conjugate Functions

Conjugate function of $f : \mathbb{R} \to \mathbb{R}$,

$$f^*(y) = \max_x y^T x - f(x)$$

and is always convex.

- Useful in formulation in dual programs, since

$$-f^*(y) = \min_x f(x) - y^T x$$

- If f is closed and convex, then $f^{**} = f$ and,

$$x \in \partial f^*(y) \iff y \in \partial f(x) \iff x \in \underset{z}{\operatorname{argmin}} f(z) - y^T z$$

  Since $x \in \operatorname{argmin}_z f(z) - y^T z \iff 0 \in \partial f(x) - y \iff y \in \partial f(x) \iff x \in \partial f^*(y)$
  For proving the other direction, use $f^{**} = f$.

- If f is strictly convex, then f^* is differentiable and $\nabla f^*(y) = \operatorname{argmin}_z f(z) - y^T z$

20.3 Dual first-order methods

Using the properties of conjugate functions, we can optimize the dual (conjugate) problem without calculating it's gradient directly. This is especially useful if we cannot obtain the dual or the conjugate in closed form. Consider a convex optimization problem with affine equality constraint,

$$\min_x f(x) \text{ subject to } Ax = b$$

Deriving its dual formulation, the Lagrangian is given by,

$$L(x, u) = f(x) + u^T(Ax - b)$$

and the dual function is,

$$\begin{aligned} g(u) &= \min_x L(x, u) = \min_x f(x) + u^T(Ax - b) \\ &= \min_x f(x) - (-A^T u)^T x - u^T b \\ &= -f^*(-A^T u) - u^T b \end{aligned}$$

The dual problem is therefore,

$$\max_u -f^*(-A^T u) - u^T b$$

The subgradient is given by,

$$\begin{aligned} \partial g(u) &= A\partial f^*(-A^T u) - b \\ &= Ax - b \text{ where } x \in \operatorname*{argmin}_z f(z) + u^T Az \end{aligned}$$

where we have used the property of conjugate functions (for closed and convex f).

20.3.1 Dual subgradient method

The above formulation of the problem can be used to develop the subgradient method, which maximizes the dual objective for the problem given above.

1. Start with an initial guess $u^{(0)}$
2. Repeat for $k = 1, 2, 3, \ldots$

$$\begin{aligned} x^{(k)} &\in \operatorname*{argmin}_z f(z) + (u^{(k-1)})^T Az && \text{(Note that this is KKT stationarity condition)} \\ u^{(k)} &= u^{(k-1)} + t_k(Ax^{(k)} - b) && \text{(Note that } (Ax^{(k)} - b) \text{ is subgradient of } g) \end{aligned}$$

The step sizes t_k can be chosen in standard ways.
Note that this method maximizes the dual objective without the need to obtain an expression for the dual function or the conjugate of f.

20.3.2 Dual gradient ascent

If f is strictly convex, then f^* is differentiable and the subgradient of f^* at x is now a singleton set, which is equal to the gradient $\nabla f^*(x)$, so the above method becomes dual gradient ascent,

1. Start with an initial guess $u^{(0)}$
2. Repeat for $k = 1, 2, 3, \ldots$

$$x^{(k)} = \underset{z}{\operatorname{argmin}}\, f(z) + (u^{(k-1)})^T Az \qquad \text{(Note that this is KKT stationarity condition)}$$
$$u^{(k)} = u^{(k-1)} + t_k(Ax^{(k)} - b) \qquad \text{(Note that } (Ax^{(k)} - b) \text{ is gradient of } g)$$

The step sizes t_k can be chosen in standard ways.
Proximal gradient descent and acceleration can also be applied to this framework.

20.3.3 Convergence analysis

Theorem 20.1 *Assume f is a closed and convex function. Then f is strongly convex with parameter m $\Longleftrightarrow \nabla f^*$ is Lipschitz with parameter $\frac{1}{m}$.*

Proof:
Proof of "$\Longrightarrow$":
If g is strongly convex with parameter m and x is the minimizer,

$$g(y) \geq g(x) + \frac{m}{2}\|y - x\|_2^2, \text{ for all } y$$

Define $g_u(x) = f(x) - u^T x$ and since strong convexity implies strict convexity, f^* is differentiable and using the property of conjugate functions,

$$\underset{z}{\operatorname{argmin}}\, g_u(z) = x_u = \nabla f^*(u)$$
$$\underset{z}{\operatorname{argmin}}\, g_v(z) = x_v = \nabla f^*(v)$$

Using the fact that x_u and x_v are minimizers of g_u and g_v respectively,

$$g_u(x_v) \geq g_u(x_u) + \frac{m}{2}\|x_v - x_u\|_2^2 \Rightarrow f(x_v) - u^T x_v \geq f(x_u) - u^T x_u + \frac{m}{2}\|x_v - x_u\|_2^2$$
$$g_v(x_u) \geq g_v(x_v) + \frac{m}{2}\|x_u - x_v\|_2^2 \Rightarrow f(x_u) - v^T x_u \geq f(x_v) - v^T x_v + \frac{m}{2}\|x_u - x_v\|_2^2$$

Adding these inequalities and applying Cauchy-Schwartz inequality,

$$f(x_v) + f(x_u) - u^T x_v - v^T x_u \geq f(x_u) + f(x_v) - u^T x_u - v^T x_v + m\|x_u - x_v\|_2^2$$
$$(u - v)^T(x_u - x_v) \geq m\|x_u - x_v\|_2^2 \Rightarrow \|u - v\|_2\|x_u - x_v\|_2 \geq m\|x_u - x_v\|_2^2$$
$$\|\nabla f^*(u) - \nabla f^*(v)\|_2^2 \leq \frac{1}{m}\|u - v\|_2$$

Proof of "$\Longleftarrow$":
Assume f^* has Lipschitz property, with constant $L = \frac{1}{m}$. Define $g_x(z)$, which is also Lipschitz with constant L,

$$\begin{aligned}
g_x(z) &= f^*(z) - \nabla f^*(x)^T z \\
\nabla g_x(z) &= \nabla f^*(z) - \nabla f^*(x) \\
\nabla^2 g_x(z) &= \nabla^2 f^*(z) \preceq LI
\end{aligned}$$

Using the Taylor expansion of $g_x(z)$, we get the following inequality,

$$g_x(z) \leq g_x(y) + \nabla g_x(y)^T (z - y) + \frac{L}{2}\|z - y\|_2^2$$

Minimizing both sides over z, where we use the minimizer for a quadratic function, $z^* = y - \nabla g_x(y)/L$,

$$\begin{aligned}
g_x(x) &\leq g_x(y) - \nabla g_x(y)^T \frac{\nabla g_x(y)}{L} + \frac{L}{2}\| - \frac{\nabla g_x(y)}{L}\|_2^2 \\
f^*(x) - \nabla f^*(x)^T x &\leq f^*(y) - \nabla f^*(x)^T y - \frac{1}{2L}\|\nabla f^*(y) - \nabla f^*(x)\|_2^2 \\
\frac{1}{2L}\|\nabla f^*(y) - \nabla f^*(x)\|_2^2 &\leq f^*(x) - f^*(y) + \nabla f^*(x)^T (x - y)
\end{aligned}$$

Similarly, we have

$$\frac{1}{2L}\|\nabla f^*(y) - \nabla f^*(x)\|_2^2 \leq f^*(y) - f^*(x) + \nabla f^*(y)^T (y - x)$$

Adding these inequalities, we get,

$$\frac{1}{L}\|\nabla f^*(y) - \nabla f^*(x)\|_2^2 \leq (\nabla f^*(y) - \nabla f^*(x))^T (y - x)$$

Let $u = \nabla f^*(x)$ and $v = \nabla f^*(y)$, then we have $x = \nabla f(u)$ and $y = \nabla f(v)$, since $f^{**} = f$. Then,

$$(\nabla f(u) - \nabla f(v))^T (u - v) \geq \frac{1}{L}\|u - v\|_2^2$$

which implies that f is strongly convex with parameter $1/L = m$. ■

20.3.4 Convergence Guarantees

Using the above theorem and the results from gradient descent convergence analysis:

- If f is strongly convex with parameter m, then dual gradient ascent with constant step sizes $t_k = m$ converges at sublinear rate $O(1/\epsilon)$. Note that in order to get a sublinear rate, we require strong convexity of f, which is a stronger condition that primal gradient descent (which required ∇f Lipschitz).

- If f is strongly convex with parameter m and ∇f is Lipschitz with parameter L, then dual gradient ascent with step sizes $t_k = 2/(1/m + 1/L)$ converges at linear rate $O(log(1/\epsilon))$.

20.4 Dual Decomposition

Consider the problem

$$\min_x \sum_{i=1}^{B} f_i(x_i) \text{ subject to } Ax = b$$

where $x = (x_1, \ldots, x_B) \in \mathbb{R}^n$ is divided into B blocks of variables where each $x_i \in \mathbb{R}^{n_i}$. Here, f is decomposable into B blocks, but the equality constraint does not decompose in a similar manner in the primal form of the problem.
The minimization problem in calculation of the sub-gradient allows us to decompose it for each individual block x_i. We can partition A accordingly, $A = [A_1, \ldots, A_B]$, where $A_i \in \mathbb{R}^{m \times n_i}$,

$$\begin{aligned} x^+ &\in \operatorname*{argmin}_z f(z) + u^T A z = \operatorname*{argmin}_z \sum_{i=1}^{B} (f_i(z_i) + u^T (A_i z_i)) \\ \iff x_i^+ &\in \operatorname*{argmin}_{z_i} f_i(z_i) + u^T (A_i z_i), \qquad i = 1, 2, \ldots, B \end{aligned}$$

Using the above, we have the dual decomposition algorithm: Repeat for $k = 1, 2, \ldots$

$$\begin{aligned} x_i^{(k)} &\in \operatorname*{argmin}_{z_i} f_i(z_i) + (u^{(k-1)})^T A_i z_i, \qquad i = 1, 2, \ldots, B \\ u^{(k)} &= u^{(k-1)} + t_k \left(\sum_{i=1}^{B} A_i x_i^{(k)} - b \right) \end{aligned}$$

The advantage of this decomposition is that it allows parallelized updates of each block x_i. It is helpful to think of it as a two-step process:

- **Broadcast:** send u to each of the B processors, each optimizes in parallel to find x_i.
- **Gather:** collect $A_i x_i$ from each processor and update the global dual variable u.

20.4.1 Inequality constraints

Consider the problem

$$\min_x \sum_{i=1}^{B} f_i(x_i) \text{ subject to } \sum_{i=1}^{B} A_i x_i \leq b$$

This again can be decomposed into minimzation over each of the individual x_i in the subgradient calculation step. The difference is that we now project the update onto $\mathbb{R}^n_+$, so it is a projected subgradient method: Repeat for $k = 1, 2, \ldots$

$$\begin{aligned} x_i^{(k)} &\in \operatorname*{argmin}_{z_i} f_i(z_i) + (u^{(k-1)})^T A_i z_i, \qquad i = 1, 2, \ldots, B \\ u^{(k)} &= \left(u^{(k-1)} + t_k \left(\sum_{i=1}^{B} A_i x_i^{(k)} - b \right) \right)_+ \end{aligned}$$

where u_+ denotes the positive part of u, i.e., $(u_+)_i = \max\{0, u_i\}, i = 1, \ldots, m$.

20.5 Augmented Lagrangian Method

One disadvantage of dual ascent is that it requires strong convexity of the objective function to ensure convergence. The augmented Lagrangian method, also known as the method of multipliers, gains better convergence properties by transforming the primal problem

$$\min_x f(x) + \frac{\rho}{2}\|Ax - b\|_2^2 \text{ subject to } Ax = b$$

where $\rho > 0$ is a parameter. This is clearly the same problem as the original, and when A has full column rank, the objective function is strongly convex. This ensures convergence of the dual ascent method,

$$\begin{aligned} x^{(k)} &= \operatorname*{argmin}_z f(z) + (u^{(k-1)})^T Az + \frac{\rho}{2}\|Az - b\|_2^2 \\ u^{(k)} &= u^{(k-1)} + \rho(Ax^{(k)} - b) \end{aligned}$$

Note that we have the step size as $t_k = \rho$. This is because,

$$\begin{aligned} x^{(k)} &= \operatorname*{argmin}_z f(z) + (u^{(k-1)})^T Az + \frac{\rho}{2}\|Az - b\|_2^2 \\ \iff 0 &\in \partial f(x^{(k)}) + A^T\left(u^{(k-1)} + \rho(Ax^{(k)} - b)\right) \\ &= \partial f(x^{(k)}) + A^T u^{(k)} \end{aligned}$$

So choosing this value of t_k gives us the stationarity condition in the original primal problem, under mild conditions $Ax^{(k)} - b \to 0$ as $k \to \infty$ (primal iterates approach feasibility), so the KKT conditions are satisfied in the limit. Hence, $x^{(k)}, u^{(k)}$ converge to the solutions x^*, u^*.

While we see much better convergence properties with the augmented Lagrangian method, we have lost the property of decomposability.

20.6 Alternating Direction Method of Multipliers (ADMM)

Alternating direction method of multipliers strives to get the best of both words: retain better convergence properties of the augmented Lagrangian method while maintaining decomposability. Consider problem,

$$\min_{x,z} f(x) + g(z) \text{ subject to } Ax + Bz = c$$

Even if the original problem is not in this form, we can often manipulate them to fit this form by introducing auxiliary variables. We can then augment the objective function,

$$\min_{x,z} f(x) + g(z) + \frac{\rho}{2}\|Ax + Bz - c\|_2^2 \text{ subject to } Ax + Bz = c$$

where $\rho > 0$ is some parameter. With this formulation, we can write the augmented Lagrangian

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

The augmented Lagrangian method would have jointly minimized the above function over x, z,

$$\left(x^{(k)}, z^{(k)}\right) = \operatorname*{argmin}_{x,z} L_\rho(x, z, u^{(k-1)})$$

ADMM splits this minimization into two steps, first minimizing over x and then over z (or vice versa), using the update value of the previous minimizer in the second step. Repeat for $k = 1, 2, \ldots$,

$$\begin{aligned} x^{(k)} &= \underset{x}{\operatorname{argmin}}\, L_\rho(x, z^{(k-1)}, u^{(k-1)}) \\ z^{(k)} &= \underset{z}{\operatorname{argmin}}\, L_\rho(x^{(k)}, z, u^{(k-1)}) \\ u^{(k)} &= u^{(k-1)} + \rho(Ax^{(k)} + Bz^{(k)} - c) \end{aligned}$$

Note that the update step for u no longer uses the gradient, since the result of separately minimizing x and z is not necessarily the same as jointly minimizing over both variables.

20.6.1 Convergence Guarantees

Under modest assumptions - f, g are closed and convex and A, B are not required to be full rank, ADMM iterates satisfy, for any $\rho > 0$,

- **Residual convergence:** $r^{(k)} = Ax^{(k)}Bz^{(k)}c \to 0$ as $k \to \infty$, i.e., primal iterates approach feasibility.
- **Objective convergence:** $f(x^{(k)}) + g(z^{(k)}) \to f^\star + g^\star$, where $f^\star + g^\star$ is the optimal objective value for the primal.
- **Dual convergence:** $u^{(k)} \to u^\star$, where $u^\star$ is a dual solution.

Note that ADMM does not guarantee that the primal iterates converge to the primal solution, in general.

ADMM roughly behaves like a first-order method, but it much more flexible and allows problems to be solved in parallel, even when it is not obvious from the problem structure. Theory on convergence rates are still being worked out: see Hong and Luo (2012), Deng and Yin (2012), Iutzeler et al. (2014), Nishihara et al. (2015).

20.6.2 Scaled form ADMM

Typically, ADMM is used in scaled form for convenience. Let $w = u/\rho$. Then the augmented Lagrangian is,

$$\begin{aligned} L_\rho(x, z, w) &= f(x) + g(z) + \rho w^T(Ax + Bz - c) + \frac{\rho}{2}\|Ax + Bz - c\|_2^2 \\ &= f(x) + g(z) + \left(\frac{\rho}{2}\right) 2w^T(Ax + Bz - c) + \frac{\rho}{2}\|Ax + Bz - c\|_2^2 + \frac{\rho}{2}\|w\|_2^2 - \frac{\rho}{2}\|w\|_2^2 \\ &= f(x) + g(z) + \frac{\rho}{2}\|Ax + Bz - c + w\|_2^2 - \frac{\rho}{2}\|w\|_2^2 \end{aligned}$$

The corresponding ADMM updates are,

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

Note that the $k^{th} iterate \mathrm{w}^{(k)}$ is a running sum of residuals

$$w^{(k)} = w^{(0)} + \sum_{i=1}^{k} \left(Ax^{(i)} + Bz^{(i)}c\right)$$

20.6.3 Example: alternating projections

Consider the problem of finding a point in the intersection of two convex sets $C, D \subseteq \mathbb{R}^n$, which we write as

$$\min_x I_C(x) + I_D(x)$$

As seen in a previous lecture, this problem can be reformulated as the maximum distance to each of the two sets, and apply the subgradient method with a chosen step-size. Alternatively, to get this into ADMM form, we can introduce an auxiliary variable z so the problem now becomes

$$\min_{x,z} I_C(x) + I_D(z) \text{ subject to } x - z = 0$$

Each ADMM cycle involves two projections

$$\begin{aligned} x^{(k)} &= \operatorname*{argmin}_x P_C(z^{(k-1)} - w^{(k-1)}) \\ z^{(k)} &= \operatorname*{argmin}_z P_D(x^{(k)} + w^{(k-1)}) \\ w^{(k)} &= w^{(k-1)} + x^{(k)} - z^{(k)} \end{aligned}$$

Comparing the above to the classic von Neumann alternating projections algorithm

$$\begin{aligned} x^{(k)} &= \operatorname*{argmin}_x P_C(z^{(k-1)}) \\ z^{(k)} &= \operatorname*{argmin}_z P_D(x^{(k)}) \end{aligned}$$

The difference in ADMM is that we now have a dual variable, w, which is often called the "offset" variable and is equal to the sum of the residuals. In this setting, the ADMM algorithm converges much quicker than standard alternating projections.

When one of the sets, say, C is a linear subspace, then due to linearity, w does not matter for the first projection. Initialized at $z^{(0)} = y$, ADMM then is equivalent to Dykstra's algorithm (which has better convergence properties than the von Neumann alternating projections algorithm) for finding the closest point in the intersection $C \cap D$ to y.

References

[Boyd 10] S. Boyd and N. Parikh and E. Chu and B. Peleato and J. Eckstein, (2010), "Distributed optimization and statistical learning via the alternating direction method of multipliers"

[Deng 12] W. Deng and W. Yin, (2012), "On the global and linear convergence of the generalized alternating direction method of multipliers"

[Hong 12] M. Hong and Z. Luo, (2012), "On the linear convergence of the alternating direction method of multipliers"

[Iutz 14] F. Iutzeler and P. Bianchi and Ph. Ciblat and W. Hachem, (2014), "Linear convergence rate for distributed optimization with the alternating direction method of multipliers"

[Nish 15] R. Nishihara and L. Lessard and B. Recht and A. Packard and M. Jordan, (2015), "A general analysis of the convergence of ADMM"

[LN] L. Vandenberghe, Lecture Notes for EE 236C, UCLA, Spring 2011-2012