9.1 Dual (sub)gradient methods

Primal problem:

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

Dual is

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

where f^* is the conjugate of f. Let $g(u) := f^*(-A^T u) - b^T u$. Our goal is to minimize $g(u)$. The subdifferertial is given by

$$\partial g(u) = A\partial f^*(-A^T u) - b^T u,$$

but if $x \in \operatorname{argmin}_z L(z, u)$ then $\partial g(u) = Ax - b$. We may solve this as follows:

guess initial $u^{(0)}$
for $k = 1$ **to** ... **do**
  choose $x^{(k)} \in \operatorname{argmin}_x f(x) + (u^{(k-1)})^T Ax$
  $u^{(k)} = k^{(k-1)} + t_k \left(Ax^{(k-1)} - b\right)$
end for

Formally: if f is strictly convex then

1. conjugate f^* is differentiable
2. procedure is dual gradient ascent
3. $x^{(k)}$ is the unique minimizer

We can choose t_k as before and apply proximal methods (or acceleration).

9.2 Decomposable Dual

Example 9.1.

$$\min_x \sum_{i=1}^{P} f_i(x_i) \text{ s. t. } Ax = b$$

standard minimization decomposes into $x^+ \in \operatorname{argmin}_x \sum_{i=1}^{P} f_i(x_i) + u^T Ax$*, which is equivalent to solving separately for each* x_i*:*

$$x_i^+ \in \operatorname*{argmin}_{x_i} f_i(x_i) + u^T A_i x_i.$$

So we can iterate:

$$x_i^{(k)} \in \underset{x_i}{\operatorname{argmin}}\, f(x_i) + (u^{(k-1)})^T A_i x_i$$
$$u^{(k)} = k^{(k-1)} + t_k \left(\sum_{i=1}^{P} A_i x_i^{(k)} - b \right)$$

Strong duality holds in this particular example since we have no inequality constraints. If the constraints are inequalities, i.e. $Ax \leq b$, we make a slight modification to $u^{(k)}$:

$$u^{(k)} = \left(k^{(k-1)} + t_k \left(\sum_{i=1}^{B} A_i x_i^{(k)} - b \right) \right)_+$$

9.3 Augmented Lagrangian

We need some constraints on f for dual ascent to work ($\rightarrow g^*$), which the Augmented Lagrangian provides. Some simple sufficient conditions are:

1. f is strongly convex $\Rightarrow$ for accuracy ϵ we require $\mathcal{O}(1/\epsilon)$ iterations
2. f is strongly convex and ∇f Lipschitz $\Rightarrow$ $\mathcal{O}(log(1/\epsilon))$ iterations

Note: To achieve strong duality (primal optimality) the program must also satisfy one of the conditions mentioned earlier (e.g. Slater's condition).

Transform $\min_x f(x) + \frac{\rho}{2}||Ax - b||_2^2$ s. t. $Ax = b$. The objective is strongly convex if A has full column rank. Dual gradient ascent then becomes

$$x^{(k)} = \underset{x}{\operatorname{argmin}}\, f(x) + (u^{(k-1)})^T Ax + \frac{\rho}{2}||Ax - b||_2^2$$
$$u^{(k)} = k^{(k-1)} + \rho(Ax^{(k-1)} - b)$$

Replacing the step size t_k with ρ gives better convergence properties than the original DGA. But by introducing the norm we lose the decomposability property (if we had it) and attendant opportunity for parallelization. ρ balances primal feasibility with a small objective; a larger ρ places less weight on objective value and forces $x^{(k)}$ closer to primal feasible points.

9.4 Alternating Direction Method of Multipliers (ADMM)

Fixes the augmented Lagrangian

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

Add $\frac{\rho}{2}||Ax + Bz - c||_2^2$ to the objective, penalizing unfeasibility:

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

Iteratively update estimates of x^*, z^*, u^*:

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

Properties of ADMM (some of which do not require A and B to be full rank):

1. $Ax^{(k)} + Bz^{(k)} - c \to 0$ as $k \to \infty$ (primal feasibility)
2. $f^{(k)} + g^{(k)} \to f^* + g^*$ (primal optimality)
3. $u^{(k)} \to u^*$ (dual solution)
4. doesn't necessarily give $x^{(k)} \to x^*$ and $z^{(k)} \to z^*$

The exact convergence rate is unknown, but empirically seems close to $\mathcal{O}(1/\epsilon)$.

Example 9.2 (LASSO)**.**

$$\min_\beta \frac{1}{2}||y + X\beta||_2^2 + \lambda||\alpha|| \text{ s. t. } \alpha = \beta$$

ADMM update:

$$
\begin{aligned}
\beta^{(k)} &= \left(X^T X + \rho I\right)^{-1} \left(X^T y + \rho(\alpha^{(k-1)} - w^{(k-1)})\right) \\
\alpha^{(k)} &= S_{\lambda/\rho}(\beta^{(k)} + w^{(k-1)}) \\
w^{(k)} &= w^{(k-1)} + \beta^{(k)} - \alpha^{(k)}
\end{aligned}
$$

Issues with ADMM:

- How to choose ρ.
- Different ADMM formulations of the problem may have different convergence properties.

9.5 Consensus ADMM

$$\min_x \sum_{i=1}^{P} f_i(a_i^T x + b_i) + g(x)$$

Introduce blocks of RVs $x_1, ..., x_P$ and minimize:

$$\min_{x1,\ldots,x_P,x} \sum_{i=1}^{P} f_i(a_i^T x + b_i) + g(x) \text{ s. t. } x_i = x \,\forall i$$

Consensus ADMM update:

$$x_i^{(k)} = \underset{x_i}{\operatorname{argmin}} f_i(a_i^T x_i + b_i) + \frac{\rho}{2}||x_i - x^{(k-1)} + w_i^{(k-1)}||_2^2$$
$$x^{(k)} = \underset{x}{\operatorname{argmin}} \frac{\rho}{2}||x - \overline{x}^{(k)} + w_i^{(k-1)}||_2^2 + g(x)$$
$$w_i^{(k)} = w_i^{(k-1)} + x_i^{(k)} - x^{(k)}$$

9.6 Coordinate Descent

Works well with LASSO.If $f(x) = g(x) + \sum_{i=1}^n h_i(x_i)$ where g is convex and differentiable, h merely convex $\Rightarrow$ we can: Guess $x^{(0)}$. Update according to:

$$x_1^{(k)} \in \underset{x_1}{\operatorname{argmin}} f(x_1, x_2^{(k-1)}, ..., x_n^{(k-1)})$$
$$x_2^{(k)} \in \underset{x_2}{\operatorname{argmin}} f(x_1^{(k)}, x_2, ..., x_n^{(k-1)}) \quad \text{(minimize over whole vector or block)}$$
$$...$$

Example 9.3 (LASSO)**.** *(state of the art in LASSO software.)*

$$||\beta|| = \sum_{i=1}^{P} |\beta_i|$$
$$\beta_i = S_{\lambda/||x_i||_2^2}\left(\frac{X_i^T(y - X_{-i}\beta_{-i})}{X_i^T X_i}\right)$$

just take the derivative of the objective w. r. t. β_i.