4.3 Proximal Algorithms

4.3.1 Proximal Operator

Definition:(Proximal operator) Let $f : \mathbb{R}^n \to \overline{\mathbb{R}}$ be a closed proper convex function. The proximal operator associated with f is defined by

$$\text{prox}_f(x) := \arg\min_u f(u) + \frac{1}{2}\|x-u\|^2$$

We can also consider the scaled proximal operator.

$$\text{prox}_{\lambda f}(x) := \arg\min_u f(u) + \frac{1}{2\lambda}\|x-u\|^2$$

Example:
Consider the indicator function of a closed convex set:

$$\delta_C(x) = \begin{cases} 0 & x \in C \\ \infty & \text{otherwise} \end{cases}$$

Then

$$\text{prox}_{\delta_C}(x) = \arg\min_u \delta_C(u) + \frac{1}{2}\|x-u\|^2 = P_C(x)$$

Hence, the proximal operator for a indicator function is just the projection to the set.
We can therefore consider prox_f as a generalized projection.

Example:
Let $f(x) := \|x\|_1$. Then

$$(\text{prox}_{\lambda f}(x))_i = T_\lambda(x_i) := \begin{cases} x_i - \lambda & x_i > \lambda \\ x_i + \lambda & x_i < -\lambda \\ 0 & \text{otherwise} \end{cases}$$

T_λ is also called the soft-thresholding operator.

4.3.2 Basic rules

Proposition:

1. If $f(x) = ag(x) + b$ with $a > 0$, then

$$\text{prox}_f(x) = \text{prox}_{ag}(x)$$

2. If $f(x) = g(x) + \langle a, x\rangle + b$, then

$$\text{prox}_f(x) = \text{prox}_g(x-a)$$

3. If $f(x) = g(ax+b)$ with $a \neq 0$, then

$$\mathrm{prox}_f(x) = \frac{1}{a}\big(\mathrm{prox}_{a^2 g}(ax+b) - b\big)$$

Proposition:(Moreau decomposition) Suppose f is closed, proper and convex. Then

$$x = \mathrm{prox}_f(x) + \mathrm{prox}_{f^*}(x)$$

If $\lambda > 0$, then

$$x = \mathrm{prox}_{\lambda f}(x) + \lambda \mathrm{prox}_{\frac{1}{\lambda} f^*}\big(\frac{x}{\lambda}\big)$$

Proof. Suppose $u = \mathrm{prox}_f(x)$. Then the optimal condition gives

$$0 \in \partial f(u) + u - x$$

That is $x - u \in \partial f(u)$. Then $u \in \partial f^*(x-u)$. (Exercise!)
This is equivalent to

$$0 \in \partial f^*(x-u) + (x-u) - x$$

By considering the optimal condition for prox_{f^*} problem,

$$x - u = \mathrm{prox}_{f^*}(x)$$

Therefore, $x = \mathrm{prox}_f(x) + \mathrm{prox}_{f^*}(x)$. Then extended case can be proved using the simple version. □

Remark:
Let's consider the optimal condition for the proximal problem.

$$\begin{aligned} & z = \arg\min_u \{f(u) + \frac{1}{2}\|x-u\|^2\} \\ & \Leftrightarrow 0 \in \partial f(x) + z - x \\ & \Leftrightarrow x \in (I + \partial f)(x) \end{aligned}$$

Therefore, we also write $z = (I+\partial f)^{-1}(x)$, which is called the resolvent operator of ∂f.

4.3.3 Proximal point algorithm

Consider the problem

$$\min f(x)$$

where f is convex but may not be differentiable.

Lemma: x^* minimizes f if and only if

$$x^* = \mathrm{prox}_f(x^*)$$

Therefore, a minimizer of f is also a fixed point of the proximal operator.

Proof. Suppose x^* minimizes f, then

$$f(x) + \frac{1}{2}\|x - x^*\|^2 \geq f(x^*) = f(x^*) + \frac{1}{2}\|x^* - x^*\|^2$$

This shows that $x^* = \text{prox}_f(x^*)$.
Suppose $x^* = \text{prox}_f(x^*)$. Then by the optimal condition

$$0 \in \partial f(x^*) + x^* - x^*$$

So $0 \in \partial f(x^*)$. Therefore, x^* minimizes f. $\square$

This motivates the proximal point algorithm:

$$x^{k+1} := \text{prox}_{\lambda f}(x^k)$$

If the proximal operator is a contraction, we can immediately prove the convergence of this algorithm.
This may not be true in general. Nonetheless, the proximal operator has a different property that helps prove the convergence.

Proposition:(Firm nonexpansiveness) Given x_1, x_2, we have

$$\|\text{prox}_f(x_1) - \text{prox}_f(x_2)\|^2 \leq \langle \text{prox}_f(x_1) - \text{prox}_f(x_2), x_1 - x_2 \rangle$$

In particular,

$$\|\text{prox}_f(x_1) - \text{prox}_f(x_2)\| \leq \|x_1 - x_2\|$$

Given an nonexpansive operator N and $\alpha \in (0, 1)$, the operator

$$T := (1 - \alpha)I + \alpha N$$

is called an averaged operator.
For averaged operator T, if it has a fixed point, then the iteration

$$x^{k+1} := T(x^k)$$

will converge to a fixed point of T.
This is known as the Kranoselskii-Mann theorem.
In particular, the firmly nonexpansiveness operators are $\frac{1}{2}$-averaged.
Therefore, we can prove the convergence of proximal point algorithm using the Kranoselskii-Mann theorem.

4.3.4 Proximal Gradient algorithm

Consider the optimization problem

$$\min F(x) = f(x) + g(x)$$

where f is convex and differentiable, g is convex.
Suppose the proximal operator of g is simple, we consider the proximal gradient algorithm:

$$x^{k+1} := \text{prox}_{t_k g}(x^k - t_k \nabla f(x^k))$$

This is also called the forward-backward splitting method.

To get convergence result, we assume that f is L-smooth.

Theorem: Suupose f is L-smooth and $t_k = \frac{1}{L}$. Then

$$F(x^k) - F^* \leq \frac{L}{2k} \|x^0 - x^*\|^2$$

In other words, to get error less than ϵ, we need $O(\frac{1}{\epsilon})$ iterations.

Similar to gradient descent, we can get faster convergence if we assume f is also μ-strongly convex.
Theorem Suppose f is μ-strongly convex and L-smooth. Let $t_k = \frac{1}{L}$. Then

$$\|x^k - x^*\|^2 \leq \big(1 - \frac{\mu}{L}\big)^k \|x^0 - x^*\|^2$$

Therefore, we get linear convergence if f is also strongly convex.

Example: Consider the LASSO problem:

$$\min \frac{1}{2} \|Ax - b\|_2^2 + \lambda \|x\|_1$$

Let $f(x) = \frac{1}{2} \|Ax - b\|_2^2$, $g(x) = \lambda \|x\|_1$.
Then $\nabla f(x) = A^T(Ax - b)$. Recall that $\text{prox}_{tg}(x) = \text{prox}_{\lambda \| \|_1}(x) = T_{t\lambda}(x)$.
Therefore, the proximal gradient iteration for LASSO is

$$x^{k+1} = T_{t\lambda}(x^k + tA^T(Ax^k - b))$$

4.4 ADMM

4.4.1 Dual ascent

Recall that if strong duality holds, then the primal optimal value is equal to the dual optimal value, that is

$$f(x^*) = g(\lambda^*, \mu^*)$$

where x^* (λ^*, μ^*) are primal (dual) optimal solution.
In particular $x^* \in \arg\min L(x, \lambda^*, \mu^*)$.

Consider the the problem

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

The Lagrangian is $L(x, \mu) = f(x) + \langle \mu, Ax - b \rangle$
The dual function is given by

$$g(\mu) = \inf_x L(x, \mu)$$

To maximize the dual function, we consider gradient ascent

$$\mu^{k+1} = \mu^k + t_k \nabla g(\mu^k)$$

$$\nabla g(\mu_0) = \nabla_\mu \inf_x L(x, \mu_0) = \nabla_\mu \inf_x (f(x) + \langle \mu_0, Ax - b \rangle)$$

Suppose $x^+ = \arg\min(f(x) + \langle \mu_0, Ax - b \rangle)$, then

$$\nabla g(\mu_0) = \nabla_\mu (f(x^+) + \langle \mu_0, Ax^+ - b \rangle) = Ax^+ - b$$

We alternatively minimize $L(x, \mu^k)$, and then update μ^k. This leads to the following algorithm:

$$x^{k+1} = \arg\min_x L(x, \mu^k)$$

$$\mu^{k+1} = \mu^k + t_k (Ax^{k+1} - b)$$

Under some conditions (eg. f is strongly convex), this methods converges.
We can also generalize this to problems with inequality constraints.

Pros: Decomposability
Cons: Poor convergence properties

4.4.2 Augmented Lagrangian Method

Consider

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

If $\rho \geq 0$, this problem has the same set of solution as

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

This motivates the definition of the augmented Lagrangian, which is given by

$$L_\rho(x, \mu) = f(x) + \frac{\rho}{2} \|Ax - b\|^2 + \langle \mu, Ax - b \rangle$$

We try to apply this to the dual ascent algorithm.
Recall the KKT conditions for the original problem are

$$Ax^* = b, \ \nabla f(x^*) + A^T \mu^* = 0$$

Since $x^{k+1} = \arg\min L_\rho(x, \mu^k)$, we have

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

If we choose ρ as the step size for updating μ, then we have $\nabla f(x^{k+1}) + A^T\mu^{k+1} = 0$.
Hence we get the following algorithm, which is called method of multipliers,

$$x^{k+1} = \arg\min_x L_\rho(x, \mu^k)$$

$$\mu^{k+1} = \mu^k + \rho(Ax^{k+1} - b)$$

Pros: Better convergence properties
Cons: Not decomposable

4.4.3 ADMM

Consider the problem

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

The augmented Lagrangian is given by

$$L_\rho(x, z, \mu) = f(x) + g(z) + \langle \mu, Ax + Bz - c\rangle + \frac{\rho}{2}\|Ax + Bz - c\|^2$$

Instead of minimizing L_ρ over x, z jointly, we split the minimization into 2 parts. This is called the general ADMM algorithm, which is given by

$$x^{k+1} = \arg\min_x L_\rho(x, z^k, \mu^k)$$

$$z^{k+1} = \arg\min_z L_\rho(x^{k+1}, z, \mu^k)$$

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

We can also consider the scaled version of ADMM. Let $\nu = \frac{1}{\rho}\mu$, then

$$\begin{aligned} L_\rho(x, z, \mu) &= f(x) + g(z) + \langle \mu, Ax + Bz - c\rangle + \frac{\rho}{2}\|Ax + Bz - c\|^2 \\ &= f(x) + g(z) + \frac{\rho}{2}\|Ax + Bz - c + \nu\|^2 - \frac{\rho}{2}\|\nu\|^2 \end{aligned}$$

Hence, we have the following scaled ADMM

$$x^{k+1} = \arg\min_x (f(x) + \frac{\rho}{2}\|Ax + Bz^k - c + \nu^k\|^2)$$

$$z^{k+1} = \arg\min_z (g(z) + \frac{\rho}{2}\|Ax^{k+1} + Bz - c + \nu^k\|^2)$$

$$\nu^{k+1} = \nu^k + Ax^{k+1} + Bz^{k+1} - c$$

We have good convergence properties for ADMM:
Assume f, g are closed,proper and convex and strong duality holds. Then:

1. $Ax^k + Bz^k - c \to 0$.
2. $f(x^k) + g(z^*) \to p^*$
3. $\mu^k \to \mu^*$

4.4.4 Examples

Convex constraints
Consider

$$\min_{x \in C} f(x)$$

where C is a closed convex set.
We first transform the problem into ADMM form

$$\min f(x) + g(z) \text{ subject to } x - z = 0$$

where g is the indicator function of C
The z update is given by

$$z^{k+1} = \arg\min_z (g(z) + \frac{\rho}{2}\|x^{k+1} - z + \nu^k\|^2) = P_C(x^{k+1} + \nu^k)$$

where $P_C(\cdot)$ denotes the projection onto C.
Hence the ADMM iteration is give by

$$x^{k+1} = \arg\min_x f(x) + \frac{\rho}{2}\|x - z^k + \nu^k\|^2$$

$$z^{k+1} = P_C(x^{k+1} + \nu^k)$$
$$\nu^{k+1} = \nu^k + x^{k+1} - z^{k+1}$$

LASSO
Consider the l_1-regularized least square problem:

$$\min_x \frac{1}{2}\|Ax - b\|_2^2 + \lambda\|x\|_1$$

Again, we transform the problem into ADMM form

$$\min_{x,z} \frac{1}{2}\|Ax - b\|_2^2 + \lambda\|z\|_1 \text{ subject to } x - z = 0$$

We first consider the x update:

$$x^{k+1} = \arg\min_x (\frac{1}{2}\|Ax - b\|_2^2 + \frac{\rho}{2}\|x - z^k + \nu^k\|_2^2)$$

This is equivalent to the least square problem

$$\min_x \left\| \begin{bmatrix} A \\ \sqrt{\rho} I \end{bmatrix} x - \begin{bmatrix} b \\ \sqrt{\rho}(z^k - \nu^k) \end{bmatrix} \right\|_2^2$$

Hence

$$\begin{aligned} x^{k+1} &= (A^T A + \rho I)^{-1} \left[A^T \sqrt{\rho} I\right] \begin{bmatrix} b \\ \sqrt{\rho}(z^k - \nu^k) \end{bmatrix} \\ &= (A^T A + \rho I)^{-1}(A^T b + \rho(z^k - \nu^k)) \end{aligned}$$

Now we consider the z update

$$z^{k+1} = \arg\min_z \lambda\|z\|_1 + \frac{\rho}{2}\|z - x^{k+1} - \nu^k\|_2^2$$

This problem is separable. Each component of z^{k+1} is given by

$$z_i^{k+1} = \arg\min_y \lambda|y| + \frac{\rho}{2}(y - x_i^{k+1} - \nu_i^k)^2$$

We differentiate the objective function (let's call it $g(y)$)

$$g'(y) = \begin{cases} \lambda + \rho(y - x_i^{k+1} - \nu_i^k) & y > 0 \\ -\lambda + \rho(y - x_i^{k+1} - \nu_i^k) & y < 0 \end{cases}$$

If $y^* > 0$, then $y^* = x_i^{k+1} + \nu_i^k - \frac{1}{\rho}\lambda$, and this holds if $x_i^{k+1} + \nu_i^k) > \frac{1}{\rho}\lambda$.
If $y^* < 0$, then $y^* = x_i^{k+1} + \nu_i^k + \frac{1}{\rho}\lambda$, and this holds if $x_i^{k+1} + \nu_i^k) < -\frac{1}{\rho}\lambda$.
Lastly, if $|x_i^{k+1} + \nu_i^k)| \leq \frac{1}{\rho}\lambda$, then $y^* = 0$.
We denote this by $T_{\lambda/\rho}(\cdot)$ (Soft-thresholding operator)
Hence

$$z^{k+1} = T_{\lambda/\rho}(x^{k+1} + \nu^k)$$

Therefore, the ADMM iteration for LASSO is given by

$$x^{k+1} = (A^T A + \rho I)^{-1}(A^T b + \rho(z^k - \nu^k))$$

$$z^{k+1} = T_{\lambda/\rho}(x^{k+1} + \nu^k)$$

$$\nu^{k+1} = \nu^k + x^{k+1} - z^{k+1}$$