Nonlinear Optimization Lecture 17

Garrick Aden-Buie

Thursday, March 24, 2016

Last time

- Penalty
- Barrier
- Augmented Lagrangian
- Gradient Projection
  - Use steepest descent but project onto feasible region
  - Special Case: Linear equality constraints

In the above, we are interested in finding a direction that is both *feasible* and *improving.*

Continuing from last time where we found that in the special linear equality constraints case:

$$d = \left[I - A^T(AA^T)^{-1}A\right](-\nabla f(\bar{x}))$$

In this case, we update $x^{k+1} = x^k + \lambda d$.

Today we will focus on the projection matrix

$$P = I - A^T(AA^T)^{-1}A$$

Properties of P

1. $P^T = I - A^T(AA^T)^{-1}A = P$, that is P is symmetric.
2. $PA^T = A^T - A^T(AA^T)^{-1}AA^T = A^T - A^T = 0$, which is implied from d being projected onto the null space of A.
3. $PP = P$
4. $x^T Px = x^T PPx = x^T P^T Px = (Px)^T Px = \|Px\|^2 \geq 0 \; \forall x \Rightarrow P$ is PSD.

Note that is is only for linear equality constraints and when A is full-rank. Also note that when A is very, very large the inverse $(AA^T)^{-1}$ is difficult to compute.

When not linear equality…

$$\begin{aligned} \min \quad & f(x) \\ \text{s.t} \quad & Ax \leq b \\ & Qx = q \end{aligned}$$

Let $\bar{x}$ be a feasible solution, $A_1\bar{x} = b_1$ the binding constraints and $A_2\bar{x} < b_2$ be non-binding. This works when $\begin{bmatrix} A_1 \\ Q \end{bmatrix}$ is full-rank.

Frank-Wolfe Algorithm

$$\begin{aligned} \min \quad & f(x) \\ \text{s.t} \quad & Ax = b \\ & x \geq 0 \end{aligned}$$

Let $\bar{x}$ be a feasible solution and f convex, differentiable. Approximate $f(x)$ using a first-order Taylor expansion:

$$f(x) \approx f(\bar{x}) + \nabla f(\bar{x})^T (x - \bar{x})$$

And then we solve the following problem instead of the original.

$$\begin{aligned} \min \quad & f(\bar{x}) + \nabla f(\bar{x})^T (x - \bar{x}) \\ \text{s.t} \quad & Ax = b \\ & x \geq 0 \end{aligned}$$

Note that $f(\bar{x})$ and $\bar{x}$ are constants, we can convert the above into a new problem

$$\begin{aligned} \min \quad & \nabla f(\bar{x})^T x \\ \text{s.t} \quad & Ax = b \\ & x \geq 0 \end{aligned}$$

which we refer to as $ALP(\bar{x})$ or Approximate LP.

Let $\bar{y}$ b a solution of $ALP(\bar{x})$, then at the next iteration approximate $f(x)$ at $\bar{y}$.

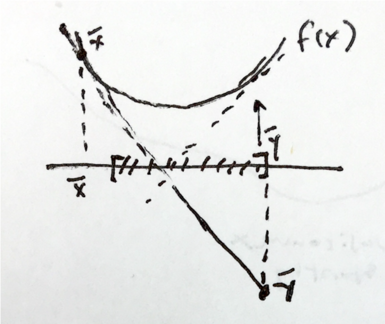

Figure 1: Illustration of the Frank-Wolfe Algorithm

Consider a direction $d = \bar{y} - \bar{x}$. Is this an improving or feasible direction?

For *feasible*

$$\begin{aligned}
\bar{x} + \lambda d &= \bar{x} + \lambda(\bar{y} - \bar{x}) \\
&= (1-\lambda)\bar{x} + \lambda\bar{y} \\
X &= \{x\colon Ax = b,\ x \geq 0\} \\
\textit{We know...}\quad & \bar{x} \in X,\ \bar{y} \in X \Rightarrow \bar{x} + \lambda d \in X\ \forall \lambda \in [0,1]
\end{aligned}$$

For *improving*, we know that $\bar{y}$ is a solution of $ALP(\bar{x})$. Also note that we have assumed that $f(x)$ is convex and differentiable.

$$\begin{aligned}
f(\bar{x}) + \nabla f(\bar{x})^T(x - \bar{x}) \geq f(\bar{x}) + \nabla f(\bar{x})^T(\bar{y} - \bar{x}) &\quad \forall x \in X \\
f(x) \geq f(\bar{x}) + \nabla f(\bar{x})^T(x - \bar{x}) &\quad \forall x \in X \\
\Rightarrow f(x) \geq f(\bar{x}) + \nabla f(\bar{x})^T(\bar{y} - \bar{x}) &\quad \forall x \in X \\
\Rightarrow \nabla f(\bar{x})^T(\bar{y} - \bar{x}) \leq f(x) - f(\bar{x}) &\quad \forall x \in X \\
\Rightarrow \nabla f(\bar{x})^T d \leq f(x^*) - f(\bar{x}) \leq 0 & \\
\Rightarrow \nabla f(\bar{x})^T d \leq 0 &
\end{aligned}$$

In summary, we know that d is a *feasible* direction. If $\nabla f(\bar{x})^T d < 0$ we know that d is an improving direction, and if $\nabla f(\bar{x})^T d = 0$ then we can show that $\bar{x}$ is optimal.

Returning to... Unconstrained Optimization: Multi-Dimensional Problem

- Multi-Dimensional
  - *Cyclic search*
  - *Steepest Descent*
    - Fast in first iterations, slow afterwards

Newton's Method

$$\begin{aligned}
q(x) &= f(x^k) + \nabla f(x^k)^T(x - x^k) + \frac{1}{2}(x - x^k)^T H(x^k)(x - x^k) \\
x^{k+1} &= x^k - H(x^k)^{-1}\nabla f(x^k)
\end{aligned}$$

The problem with this algorithm is that it involves calculating a matrix inverse, which is computationally expensive. But this algorithm is good when you are close to the solution. So you can use the steepest descent first, and then switch to Newton's method.

Quasi-Newton Methods

There are also a group of methods called *quasi-Newton methods* that combine the above methods. In essence, they start like the steepest descent and end like Newton's method.

$$\begin{aligned} x^{k+1} &= x^k - \lambda_k D_k \nabla f(x^k) \\ \lambda_k &: \text{ an optimal step length} \end{aligned}$$

If $D_k = I$, then we have the steepest descent method. If $\lambda_k D_k = H_k^{-1}$, then we have Newton's method.

This is the general scheme, but our objective is to wisely choose D_k.

D_k

> When will $d^k = -\lambda_k D_k \nabla f(x^k)$ be a descent direction?

We need to check the sign of the gradient to see that it's negative.

$$\begin{aligned} \nabla f(x^k)^T d^k &< 0 \\ \lambda_k \nabla f(x^k)^T D_k \nabla f(x^k) &< 0 \end{aligned}$$

If D_k is PD, then d^k is a descent direction.

- Start $D_1 = I$
- We want $\{D_k\} \to H^{-1}$
- We'll skip the details but we can accomplish this with the following method…

Davidon-Fletcher-Powell (DFP) Method

$$\begin{aligned} x^{k+1} &= \lambda_k d^k \\ d^k &= -D_k \nabla f(x^k) \\ D_{k+1} &= D_k + \frac{p^k (p^k)^T}{(p^k)^T q^k} - \frac{D_k q^k (q^k)^T D_k}{(q^k)^T D_k q^k} \end{aligned}$$

Where $p^k = x^{k+1} - x^k$ and $q^k = \nabla f(x^{k+1}) - \nabla f(x^k)$.