**10-725/36-725: Convex Optimization** **Fall 2016**

# Lecture 18: November 2

*Lecturer: Lecturer: Javier Pena* *Scribes: Scribes: Yizhu Lin, Pan Liu*

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



This lecture's notes illustrate some uses of various LaTeX macros. Take a look at this and imitate.

## 18.1 Review on Primal-dual interior-poit methods

Consider a convex minimization problem, assuming $f, h$ convex and differetiable, and strong duality holds,

$$\begin{aligned} \min_x \ & f(x) \\ \text{subject to } & Ax = b \\ & h(x) \leq 0 \end{aligned}$$

The central path eqution:

$$\begin{aligned} \nabla f(x) + \nabla h(x)u + A^T v &= 0 \\ Uh(x) + \tau \mathbf{1} &= 0 \\ Ax - b &= 0 \\ u, -h(x) &> 0 \end{aligned}$$

Let $w = (x, u, v)$, then the residuals

$$r(w) = r(x, u, v) := \begin{bmatrix} \nabla f(x) + \nabla h(x)u + A^T v \\ Uh(x) + \tau \mathbf{1} \\ Ax - b \end{bmatrix}$$

The Primal-dual interior-point algorithm is to apply Newton method on the central path equation, we first compute current residual based on current $(x, u, v)$, then compute the Newton step $(\Delta x, \Delta u \Delta v)$ by solving

$$\begin{bmatrix} \nabla^2 f(x) + \sum_i u_i \nabla^2 h_i(x) & \nabla h(x) & A^T \\ U \nabla h(x)^T & H(x) & 0 \\ A & 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta u \\ \Delta v \end{bmatrix} = -r(x, u, v)$$

Finally we compute a step length $\theta$, and update

$$(x^+, u^+, v^+) := (x, u, v) + \theta(\Delta x, \Delta u, \Delta v)$$

At each Newton update

$$\begin{aligned} \text{Suppose } \Delta w = (\Delta x, \Delta u, \Delta v) &= -r'(w)r(w) \\ r(w + \theta w) &\approx r(w) + r'(w)\theta \Delta w \\ &\approx r(w) + \theta r'(w)\Delta w, \text{ where } r'(w)\Delta w = -r(w) \text{ by construction} \\ &\approx (1 - \theta)r(w), \end{aligned}$$

Note if r is linear, consider the pure Newton method where $\theta = 1$ , we have

$$r(w + \theta w) \approx (1 - \theta)r(w) = 0,$$

So it takes one complete Newton step to get the solution. This is also true if a block of r is linear, you can reduce that linear block to zero in one Newton step.

## 18.2 Motivation for quasi-Newton methods

Consider a unconstrained, smooth optimization problem

$$\min_x f(x)$$

where $f$ is twice differentiable. Note $f$ don't have to be convex, since quasi-Newton Method can be applied byond convex optimization problems.
Recall two classic method, Gradient descent and Newton method. Newton method has quadratic convergence property, Gradient descent has linear convergence. But Newton's method has its trade-offs, it requires computing Hessian $\nabla^2 f(x)$ and the Newton step $p$, where $-\nabla^2 f(x)p = -\nabla f(x)$. Each of the two can be expensive.
The idea of quasi-Newton is use something like a scaled gradient, $B$, than is an approximation of Hessian, but $B$ is easy to solve and $Bp = -\nabla f(x)$ is easy to solve.

## 18.3 Quasi-Newton algorithm

### 18.3.1 Scant equation

The key idea to make the computation easier is to use information from the previous update in the current update, i.e., use $B^k$ as a warm start to compute $B^{k+1}$.
A reasonable requirement for $B^{k+1}$:

$$\nabla f(x^{k+1}) = \nabla f(x^k) + B^{k+1}s^k$$

or equivalently

$$B^{k+1}s^k = \nabla f(x^{k+1}) - \nabla f(x^k)$$

This is called the **secant equation** and written as

$$B^{k+1}s^k = y^k \text{or simply} B^+ s = y$$

where $s^k = x^{k+1} - x^k$ and $y^k = \nabla f(x^{k+1}) - \nabla f(x^k)$.
Consider the one dimensional case, secant equation tells that $B^k$ is the slope between $\nabla f(x)$ and $s$. In addition to secant equation, we also want

1. $B^+$ symmetric;
2. $B^+$ close to $B$;
3. $B \Rightarrow B^+$ maintains positive definte

### 18.3.2 Quasi-Newton algorithm

1. Compute the direction $p^k = -B^{k-1}\nabla f(x^k)$ ;
2. Set $x^{k+1} = x^k + t_k p^k$ ;
3. Update $B^{k+1}$ .

## 18.4 Most popular updates: SR1, DFP, BFGS, Broyden class

### 18.4.1 Symmetric rank-one update (SR1)

Update $B$ by adding an rank-one matrix (we want the changes of $B$ be parsimony):

$$B^{k+1} = B + auu^T$$

Plug-in secant equation:

$$y = B^+ s = Bs + auu^T s = Bs + a(u^T s)u$$

or

$$a(u^T s)u = y - Bs$$

where $a(u^T s)$ is a scalar. Thus $u$ has to be a multiple of $y - Bs$ so secant equation can hold.
Solve for a, we get:

$$a = \frac{1}{(y - Bs)^T s}$$
$$B^+ = B + \frac{(y - Bs)(y - Bs)^T}{(y - Bs)^T s}$$

### 18.4.2 Sherman-Morrision-Woodbury formula

Note that when updating $x$,

$$x^+ = x + tp = x - tB^{-1}\nabla f(x)$$

so we are also interested in updating the inverse of $B$.
In fact, by Sherman-Morrision-Woodbury formula, given a low-rank update of a matrix, the update on its inverse is also easy.

**Theorem 18.1** *Sherman-Morrision-Woodbury formula*
*Assume $A \in R^{nxn}$ , $U, V \in R^{nxd}$, $d \leq n$. Then $A + UV^T$ is nonsingular iff. $I + V^T A^1 U$ is nonsingular. In that case*

$$(A + UV^T)^{-1} = A^{-1} - A^{-1}U(I + V^T A^{-1}U)^{-1}V^T A^{-1}$$

A special case of SMW when the update is rank-one,

$$(A + UV^T)^{-1} = A^{-1} - \frac{A^{-1}uv^TA^{-1}}{1 + v^TA^{-1}u}$$

Thus for SR1, the update on the inverse of $B$, $H$, is

$$H^+ = H + \frac{(s - Hy)(s - Hy)^T}{(s - Hy)^Ty}$$

In SR1, there's a shortcomings that if the denominator $\approx 0$, SR1 may fail; and SR1 does not preserve positive definiteness.

### 18.4.3 Davidon-Fletcher-Powell (DFP) update

To overcome shortcomings in SR1, we can try rank-two update

$$H^+ = H + auu^T + bvv^T$$

Look at the secant equation on $H$ instead of $B$:

$$B^+s = y \Leftrightarrow H^+y = s$$

We have

$$s - Hy = (au^Ty)u + (bv^Ty)v$$

$u$, $v$ are not uniquely determined, but we can take $u = s, v = Hy$ to satisfy secant equation, then solving for a, b we get

$$H^+ = H - \frac{Hyy^TH}{y^THy} + \frac{ss^T}{y^Ts}$$

By SMW we also have update on $B$:

$$B^+ = (I - \frac{ys^T}{y^Ts})B(I - \frac{sy^T}{y^Ts}) + \frac{yy^T}{y^Ts}$$

We can see if $B$ is positive definite, the first term in above equation is SPD, and the second term is positive, so $B^+$ is also positive definite, i.e, DFP preserves positive definiteness.

An alternative way to compute DFP is to find a closest $B^+$ to $B$ that is symmetric and satisfies secant equation.
Solve:

$$\begin{aligned} \min_{B^+} \ & ||W^{-1}(B^+ - B)W^{-T}||_F \\ \text{subject to } & B^+ = (B^+)^T \\ & B^+s = y \end{aligned}$$

where $W \in R^{nxn}$ is nonsingular and such that $WW^Ts = y$.

### 18.4.4 Broyden-Flentcher-Goldfarb-Shanno (BFGS) update

Same ideas as the DFP update but with roles of B and H exchanged, do update on H instead of B. The solution to BFGS is similar to DFG, with swapping B and H, s and y:

$$B^+ = B - \frac{Bss^TB}{s^TBs} + \frac{yy^T}{y^Ts}$$
$$H^+ = (I - \frac{sy^T}{y^Ts})H(I - \frac{ys^T}{y^Ts}) + \frac{ss^T}{y^Ts}$$

From the equation above we can see that similar to DFP, BFGS also preserves positive definiteness. BFGS is more popular than DFP for it has a self-correcting property, thus more robust.

### 18.4.5 The Broyden class

The Broyden class is an entire class of updates that have similar shape:

$$B^+ = (1-\phi)B^+_{BFGS} + \phi B^+_{DFP}, \text{ for } \phi \in R.$$

i.e, a combination of BFGS and DFP.

## 18.5 Superlinear convergence

The standard quasi-Newton Method algorithm:

Pick initial $x^0$ and $B^0$
For $k = 0, 1, ...$
- •Solve $B^k p^k = -\nabla f(x^k)$
- •Pick $t_k$ and let $x^{k+1} = x^k + t_k p^k$
- •Update $B^k$ to $B^{k+1}$

end for

Note if take $B = I$, this becomes Gradient descent, which has linear convergence; if take $B =$ Hessian, this becomes Newton method, which has quadratic convergence under certain conditions; if $B \in$ Broyden class, this is quasi-Newton method, the convergence rate is in between, superlinear convergence:

$$\lim_{k\to\infty} \frac{||x^{k+1} - x^*||}{||x^k - x^*||} = 0.$$

This convergence rate relies on a more careful choice of step length:

$$f(x + tp) \leq f(x) + \alpha_1 t \nabla f(x)^T p, \text{ this ensures } t \text{ not too large}$$

and

$$\nabla f(x + tp)^T p \geq \alpha_2 t |\nabla f(x)^T p|, \text{ this ensures } t \text{ not too small}$$

for $0 < \alpha_1 < \alpha_2 < 1$.

The convergence rate is a result from Dennis-Moré. Under suitable assumption, assumptions, DFP and BFGS updates ensure

$$\lim_{k\to\infty} \frac{||\nabla f(x^k) - \nabla^2 f(x^k)p^k||}{||p^k||} = 0$$

for $p^k = -H^k \nabla f(x^k)$ and we get superlinear convergence.

## 18.6 Limited memory BFGS (LBFGS)

For large problems, exact quasi-Newton updates becomes too costly, since storing the complete $H$ matrix is expensive by itself.
In LBFGS, Instead of computing and storing $H$, compute an implicit modied version of $H$ by maintaining the last $m$ pairs $(y, s)$.
The BFGS method computes direction

$$p = -Hg, \text{ where } g = \nabla f(x),$$

Plug-in BFGS update on $H$, we have

$$\begin{aligned} H^+ g &= (I - \frac{sy^T}{y^T s})H(I - \frac{ys^T}{y^T s})g + \frac{ss^T g}{y^T s} \\ &= (I - \frac{sy^T}{y^T s})H(g - \frac{s^T g}{y^T s}y) + \frac{s^T g}{y^T s}s \\ &= (I - \frac{sy^T}{y^T s})p + \alpha s \\ &= p + (\alpha - \beta)s \end{aligned}$$

where

$$\begin{aligned} \alpha &= \frac{s^T g}{y^T s} \\ q &= g - \frac{s^T g}{y^T s}y = g - \alpha y \\ p &= Hq \\ \beta &= \frac{sy^T}{y^T s} \end{aligned}$$

Hence $Hg$ can be computed via two loops of length $k$ if $H$ is obtained after $k$ BFGS updates. LBFGS algorithm is then using information of the last few iterations:

1. $q := -\nabla f(x^k)$
2. For $i = k-1, ..., \min(k-m, 0)$
   $\alpha := \frac{(s^i)^T g}{(y^i)^T s^i}$
   $q := g - \alpha y$
   end for

3. $p := H^{0,k}q$
4. For $i = \min(k-m, 0), ..., k-1$
   $\beta := \frac{(y^i)^T p}{(y^i)^T s^i}$
   $q := g - \alpha y$
   end for
5. return $p$

In step 3 $H^{0,k}q$ is the initial H. Popular choice:

$$H^{0,k}q := \frac{((y^{k-1})^T s^{k-1}}{(y^{k-1})^T y^{k-1}} I$$