Quasi-Newton Methods

Ryan Tibshirani
Convex Optimization 10-725

Last time: primal-dual interior-point methods

Given the problem

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

where f, $h = (h_1, \ldots, h_m)$, all convex and twice differentiable, and strong duality holds. Central path equations:

$$r(x, u, v) = \begin{pmatrix} \nabla f(x) + Dh(x)^T u + A^T v \\ -\mathrm{diag}(u) h(x) - 1/t \\ Ax - b \end{pmatrix} = 0$$

subject to $u > 0$, $h(x) < 0$

Primal dual interior point method: repeat updates

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

where $(\Delta x, \Delta u, \Delta v)$ is defined by Newton step:

$$\begin{bmatrix} H_{\mathrm{pd}}(x) & Dh(x)^T & A^T \\ -\mathrm{diag}(u) Dh(x) & -\mathrm{diag}(h(x)) & 0 \\ A & 0 & 0 \end{bmatrix} \begin{pmatrix} \Delta x \\ \Delta u \\ \Delta v \end{pmatrix} = -r(x, u, v)$$

and $H_{\mathrm{pd}}(x) = \nabla^2 f(x) + \sum_{i=1}^m u_i \nabla^2 h_i(x)$

- Step size $s > 0$ is chosen by backtracking, while maintaining $u > 0$, $h(x) < 0$
- Primal-dual iterates are not necessarily feasible (but they are once we take one full Newton step)
- Often converges faster than barrier method

Outline

Today:

- Quasi-Newton motivation
- SR1, BFGS, DFP, Broyden class
- Convergence analysis
- Limited memory BFGS
- Stochastic quasi-Newton

Gradient descent and Newton revisited

Back to unconstrained, smooth convex optimization

$$\min_x \ f(x)$$

where f is convex, twice differentable, and $\mathrm{dom}(f) = \mathbb{R}^n$. Recall gradient descent update:

$$x^+ = x - t\nabla f(x)$$

and Newton's method update:

$$x^+ = x - t(\nabla^2 f(x))^{-1}\nabla f(x)$$

- Newton's method has (local) quadratic convergence, versus linear convergence of gradient descent
- But Newton iterations are much more expensive ...

Quasi-Newton methods

Two main steps in Newton iteration:

- Compute Hessian $\nabla^2 f(x)$
- Solve the system $\nabla^2 f(x) s = -\nabla f(x)$

Each of these two steps could be expensive

Quasi-Newton methods repeat updates of the form

$$x^+ = x + ts$$

where direction s is defined by linear system

$$Bs = -\nabla f(x)$$

for some approximation B of $\nabla^2 f(x)$. We want B to be easy to compute, and $Bs = g$ to be easy to solve

Some history

- In the mid 1950s, W. Davidon was a mathematician/physicist at Argonne National Lab
- He was using coordinate descent on an optimization problem and his computer kept crashing before finishing
- He figured out a way to accelerate the computation, leading to the first quasi-Newton method (soon Fletcher and Powell followed up on his work)
- Although Davidon's contribution was a major breakthrough in optimization, his original paper was rejected
- In 1991, after more than 30 years, his paper was published in the first issue of the SIAM Journal on Optimization
- In addition to his remarkable work in optimization, Davidon was a peace activist (see the book "The Burglary")

Quasi-Newton template

Let $x^{(0)} \in \mathbb{R}^n$, $B^{(0)} \succ 0$. For $k = 1, 2, 3, \ldots$, repeat:

1. Solve $B^{(k-1)} s^{(k-1)} = -\nabla f(x^{(k-1)})$
2. Update $x^{(k)} = x^{(k-1)} + t_k s^{(k-1)}$
3. Compute $B^{(k)}$ from $B^{(k-1)}$

Different quasi-Newton methods implement Step 3 differently. As we will see, commonly we can compute $(B^{(k)})^{-1}$ from $(B^{(k-1)})^{-1}$

Basic idea: as $B^{(k-1)}$ already contains info about the Hessian, use suitable matrix update to form $B^{(k)}$

Reasonable requirement for $B^{(k)}$ (motivated by secant method):

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

Secant equation

We can equivalently write latter condition as

$$\nabla f(x^+) = \nabla f(x) + B^+ s$$

Letting $y = \nabla f(x^+) - \nabla f(x)$, this becomes

$$B^+ s = y$$

This is called the secant equation

In addition to the secant equation, we want:

- B^+ to be symmetric
- B^+ to be "close" to B
- $B \succ 0 \Rightarrow B^+ \succ 0$

Symmetric rank-one update

Let's try an update of the form:

$$B^+ = B + auu^T$$

The secant equation $B^+ s = y$ yields

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

This only holds if u is a multiple of $y - Bs$. Putting $u = y - Bs$, we solve the above, $a = 1/(y - Bs)^T s$, which leads to

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

called the symmetric rank-one (SR1) update

How can we solve $B^+ s^+ = -\nabla f(x^+)$, in order to take next step? In addition to propogating B to B^+, let's propogate inverses, i.e., $C = B^{-1}$ to $C^+ = (B^+)^{-1}$

Sherman-Morrison formula:

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

Thus for the SR1 update the inverse is also easily updated:

$$C^+ = C + \frac{(s - Cy)(s - Cy)^T}{(s - Cy)^T y}$$

In general, SR1 is simple and cheap, but has key shortcoming: it does not preserve positive definiteness

Broyden-Fletcher-Goldfarb-Shanno update

Let's now try a rank-two update:

$$B^+ = B + auu^T + bvv^T.$$

The secant equation $y = B^+ s$ yields

$$y - Bs = (au^T s)u + (bv^T s)v$$

Putting $u = y,\ v = Bs$, and solving for a, b we get

$$B^+ = B - \frac{Bss^T B}{s^T Bs} + \frac{yy^T}{y^T s}$$

called the Broyden-Fletcher-Goldfarb-Shanno (BFGS) update

Woodbury formula (generalization of Sherman-Morrison):

$$(A + UCV)^{-1} = A^{-1} - A^{-1}U(C^{-1} + VA^{-1}U)^{-1}VA^{-1}$$

Applied to our case, we get a rank-two update on inverse C:

$$\begin{aligned} C^+ &= C + \frac{(s - Cy)s^T}{y^Ts} + \frac{s(s - Cy)^T}{y^Ts} - \frac{(s - Cy)^Ty}{(y^Ts)^2}ss^T \\ &= \left(I - \frac{sy^T}{y^Ts}\right)C\left(I - \frac{ys^T}{y^Ts}\right) + \frac{ss^T}{y^Ts} \end{aligned}$$

The BFGS update is thus still quite cheap: $O(n^2)$ operations

Importantly, BFGS update preserves positive definiteness. Recall this means $B \succ 0 \Rightarrow B^+ \succ 0$. Equivalently, $C \succ 0 \Rightarrow C^+ \succ 0$

To see this, compute

$$x^T C^+ x = \left(x - \frac{s^T x}{y^T s} y\right)^T C \left(x - \frac{s^T x}{y^T s} y\right) + \frac{(s^T x)^2}{y^T s}$$

Now observe:

- Both terms are nonnegative
- Second term is only zero when $s^T x = 0$
- In that case first term is only zero when $x = 0$

Davidon-Fletcher-Powell update

We could have pursued same idea to update inverse C:

$$C^+ = C + auu^T + bvv^T.$$

Multiplying by y, using the secant equation $s = C^+y$, and solving for a, b, yields

$$C^+ = C - \frac{Cyy^TC}{y^TCy} + \frac{ss^T}{y^Ts}$$

Woodbury then shows

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

This is the Davidon-Fletcher-Powell (DFP) update. Also cheap: $O(n^2)$, preserves positive definiteness. Not as popular as BFGS

Curvature condition

Observe that $B^+ \succ 0$ and $B^+ s = y$ imply

$$y^T s = s^T B^+ s > 0.$$

called the curvature condition. Fact: if $y^T s > 0$, then there exists $M \succ 0$ such that $Ms = y$

Interesting alternate derivation for DFP update: find B^+ "closest" to B w.r.t. appropriate conditions, i.e., solve

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

where W is nonsingular and such that $WW^T s = y$. And BFGS solves same problem but with roles of B and C exchanged

Broyden class

SR1, DFP, and BFGS are some of numerous possible quasi-Newton updates. The Broyden class of updates is defined by:

$$B^+ = (1-\phi)B^+_{\mathsf{BFGS}} + \phi B^+_{\mathsf{DFP}}, \quad \phi \in \mathbb{R}$$

By putting $v = y/(y^T s) - Bs/(s^T Bs)$, we can rewrite the above as

$$B^+ = B - \frac{Bss^T B}{s^T Bs} + \frac{yy^T}{y^T s} + \phi(s^T Bs)vv^T$$

Note:

- BFGS corresponds to $\phi = 0$
- DFS corresponds to $\phi = 1$
- SR1 corresponds to $\phi = y^T s/(y^T s - s^T Bs)$

Convergence analysis

Assume that f convex, twice differentiable, having $\mathrm{dom}(f) = \mathbb{R}^n$, and additionally

- ∇f is Lipschitz with parameter L
- f is strongly convex with parameter m
- $\nabla^2 f$ is Lipschitz with parameter M

(same conditions as in the analysis of Newton's method)

Theorem: Both DFP and BFGS, with backtracking line search, converge globally. Furthermore, for all $k \geq k_0$,

$$\|x^{(k)} - x^\star\|_2 \leq c_k \|x^{(k-1)} - x^\star\|_2$$

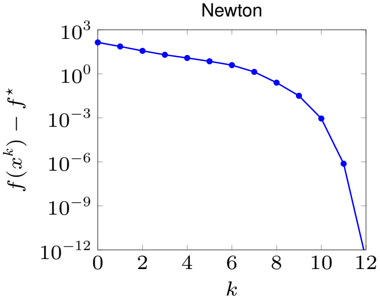
where $c_k \to 0$ as $k \to \infty$. Here k_0, c_k depend on L, m, M

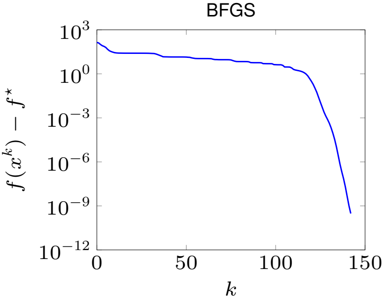
This is called local superlinear convergence

Example: Newton versus BFGS

Example from Vandenberghe's lecture notes: Newton versus BFGS on LP barrier problem, for $n = 100$, $m = 500$

$$\min_x \; c^T x - \sum_{i=1}^{m} \log(b_i - a_i^T x)$$





Note that Newton update is $O(n^3)$, quasi-Newton update is $O(n^2)$. But quasi-Newton converges in less than 100 times the iterations

Implicit-form quasi-Newton

For large problems, quasi-Newton updates can become too costly

Basic idea: instead of explicitly computing and storing C, compute an implicit version of C by maintaining all pairs (y, s)

Recall BFGS updates C via

$$C^+ = \left(I - \frac{sy^T}{y^T s}\right) C \left(I - \frac{ys^T}{y^T s}\right) + \frac{ss^T}{y^T s}$$

Observe this leads to

$$C^+ g = p + (\alpha - \beta) s, \quad \text{where}$$
$$\alpha = \frac{s^T g}{y^T s},\ q = g - \alpha y,\ p = Cq,\ \beta = \frac{y^T p}{y^T s}$$

We see that C^+g can be computed via too loops of length k (if C^+ is the approximation to the inverse Hessian after k iterations):

1. Let $q = -\nabla f(x^{(k)})$
2. For $i = k-1, \ldots, 0$:
   (a) Compute $\alpha_i = (s^{(i)})^T q / ((y^{(i)})^T s^{(i)})$
   (b) Update $q = q - \alpha y^i$
3. Let $p = C^{(0)} q$
4. For $i = 0, \ldots, k-1$:
   (a) Compute $\beta = (y^{(i)})^T p / ((y^{(i)})^T s^{(i)})$
   (b) Update $p = p + (\alpha_i - \beta) s^{(i)}$
5. Return p

Limited memory BFGS

Limited memory BFGS (LBFGS) simply limits each of these loops to be length m:

1. Let $q = -\nabla f(x^{(k)})$
2. For $i = k-1, \ldots, k-m$:
   (a) Compute $\alpha_i = (s^{(i)})^T q / ((y^{(i)})^T s^{(i)})$
   (b) Update $q = q - \alpha y^i$
3. Let $p = \bar{C}^{(k-m)} q$
4. For $i = k-m, \ldots, k-1$:
   (a) Compute $\beta = (y^{(i)})^T p / ((y^{(i)})^T s^{(i)})$
   (b) Update $p = p + (\alpha_i - \beta) s^{(i)}$
5. Return p

In Step 3, $\bar{C}^{(k-m)}$ is our guess at $C^{(k-m)}$ (which is not stored). A popular choice is $\bar{C}^{(k-m)} = I$, more sophisticated choices exist

Stochastic quasi-Newton methods

Consider now the problem

$$\min_x \; \mathbb{E}_\xi[f(x,\xi)]$$

for a noise variable ξ. Tempting to extend previous ideas and take stochastic quasi-Newton updates of the form:

$$x^{(k)} = x^{(k-1)} - t_k C^{(k-1)} \nabla f(x^{(k-1)}, \xi_k)$$

But there are challenges:

- Can have at best sublinear convergence (recall lower bound by Nemirovski et al.) So is additional overhead of quasi-Newton, worth it, over plain SGD?
- Updates to C depend on consecutive gradient estimates; noise in the gradient estimates could be a hindrance

The most straightforward adaptation of quasi-Newton methods is to use BFGS (or LBFGS) with

$$s^{(k-1)} = x^{(k)} - x^{(k-1)},\ y^{(k-1)} = \nabla f(x^{(k)}, \xi_k) - \nabla f(x^{(k-1)}, \xi_k)$$

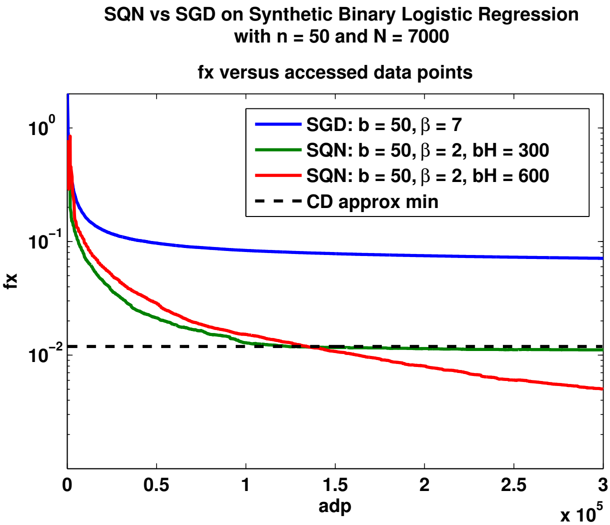
The key is to use the same noise variable ξ_k in the two stochastic gradients. This is due to Schraudolph et al. (2007)

More recently, Byrd et al. (2015) propose a stochastic version of LBFGS with three main changes:

- Perform an LBFGS update only every L iterations
- Compute s to be an average over L last search directions
- Compute y using Hessian approximation based on sampling

With proper tuning, either approach can give improvements over SGD

Example from Byrd et al. (2015):



References and further reading

- L. Bottou, F. Curtis, J. Nocedal (2016), "Optimization methods for large-scale machine learning"
- R. Byrd, S. Hansen, J. Nocedal, Y. Singer (2015), "A stochastic quasi-Newton method for large-scale optimization"
- J. Dennis and R. Schnabel (1996), "Numerical methods for unconstrained optimization and nonlinear equations"
- J. Nocedal and S. Wright (2006), "Numerical optimization", Chapters 6 and 7
- N. Schraudolph, J. Yu, S. Gunter (2007), "A stochastic quasi-Newton method for online convex optimization"
- L. Vandenberghe, Lecture notes for EE 236C, UCLA, Spring 2011-2012