

MAT 653: Statistical Simulation

Instructor: Dr. Wei Li

Scribe: Jingyao Tang

Sep 28th, 2021

Newton's Method

Take a second order Taylor expansion of $T(x_k)$ in the direction given by αp : $T(x_k, \alpha p) = f(x_k) + \alpha p^T \nabla f(x_k) + 1/2 \alpha^2 p^T \nabla^2 f(x_k) p$.

we travel to a stationary point of this quadratic approximation which is a minimizer if f is convex. Set $\alpha \equiv 1$.

Set gradient of $T(x_k, \alpha p)$ to zero and solve for p .

$$FOC : \nabla_p T(x_k, \alpha p) \underset{set}{=} 0$$

$$\nabla f(x_k) + \nabla^2 f(x_k) p = 0$$

$$\Rightarrow p = p_k = -\nabla f(x_k)^{-1} \cdot \nabla f(x_k), \text{ if } \nabla^2 f(x_k) \text{ is invertible}$$

$$x_{k+1} \leftarrow x_k - [\nabla^2 f(x_k)]^{-1} \cdot \nabla f(x_k)$$

There are two ways to solve the system:

1. If $\nabla^2 f(x_k)$ is not invertible, you can use $[\nabla^2 f(x_k)]^+$.

2. $\nabla f(x_k) + \nabla^2 f(x_k)(x_{k+1} - x_k) = 0$.

$$\Leftrightarrow \nabla^2 f(x_k) x_{k+1} = \nabla^2 f(x_k) x_k - \nabla f(x_k).$$

\Rightarrow solve for x_{k+1}

Remark: Newton method produces a sequence of points x_1, x_2, \dots that minimizes a sequence of the function by repeatedly creating a quadratic approximation to the function f .

With a quadratic approximation more closely mimicking the object function. Newton method is often more effective than the gradient method. However, the reliance on the quadratic approximation makes Newton's method more difficult to use, especially for non-convex function.

$\nabla^2 f(x_k)$ may not be invertible, so can use $[\nabla^2 f(x_k)]^+$ or solving for x_{k+1} from this system as above.

However, both approaches do **NOT** guarantee that:

$$p_k^T \nabla^2 f(x_k) < 0$$

Maximization Problem

Note: $\max f(x) = \min(-f(x))$

Newton method

$$x_{k+1} \leftarrow x_k - [\nabla^2 f(x_k)]^{-1} \cdot \nabla f(x_k)$$

note: $[\nabla^2 f(x_k)]^{-1}$ is n.d.

gradient descend method

$$x_{k+1} \leftarrow x_k - (-\nabla f(x_k))$$

Quasi Newton Method(minimization problem)

$$p_k = -[\nabla^2 f(x_k)]^{-1} \cdot \nabla f(x_k) \quad \text{Newton Method}$$

$$p_k = -B_k^{-1} \nabla f(x_k) \quad \text{Quasi -Newton Method}$$

here, B_k is some approximation to $\nabla^2 f(x_k)$. Note that

$$\begin{aligned} \nabla f(x_{k+1}) &= \nabla f(x_k) + \nabla^2 f(x_k)(x_{k+1} - x_k) + o(\|x_{k+1} - x_k\|) \\ \Leftrightarrow \nabla^2 f(x_k)(x_{k+1} - x_k) &\approx \nabla f(x_{k+1}) - \nabla f(x_k) \end{aligned}$$

I want my B_k to satisfy:

1. B_k is symmetric.
2. $B_k(x_{k+1} - x_k) = \nabla f(x_{k+1}) - \nabla f(x_k)$.

BFGS

$$B_{k+1} = B_k - \frac{B_k s_k s_k^T B_k}{s_k^T B_k s_k} + \frac{y_k y_k^T}{y_k^T s_k}$$

$$s_k = B_k(x_{k+1} - x_k)$$

$$y_k = \nabla f(x_{k+1}) - \nabla f(x_k)$$

Whenever you have : $B_0 > 0$ and $s_k^T y_k > 0$: (Secant Condition), the BFGS update produces positive definite approximation to Hessian.

Alternative version of BFGS

Update inverse of Hessians B_k :

Let $\tilde{B} = B_k^{-1}$.

$$\tilde{B}_{k+1}^{-1} = (I - p_k s_k y_k^T) \tilde{B} (I - p_k y_k s_k^T) + p_k s_k s_k^T$$

$$p_k = \frac{1}{y_k^T s_k}$$

$$p_k = -\tilde{B} \nabla f(x_k)$$

initial choice of $B_0 : I$, or some multiple of I .

Line Search

choose α assume p_k is found. An ideal choice of α is find a global minimizer of

$$\phi(\alpha) \equiv f(x_k + \alpha p_k), \alpha > 0$$

but too costly.

Instead, we use some inexact line search method to identify a step length that achieves adequate reduction in f .

Sufficient reduction condition(Armijo condition)

$$f(x_k + \alpha p_k) \leq f(x_k) + c_1 \alpha \nabla f(x_k)^T p_k,$$

here, c_1 can be some small number, say 10^{-4} . Let $\phi(\alpha) = f(x_k + \alpha p_k)$, and $l(\alpha) = f(x_k) + c_1 \alpha \nabla f(x_k)^T p_k$. The condition says we need to choose α such that there is sufficient reduction in $\phi(\alpha)$, i.e., $\phi(\alpha) \leq l(\alpha)$.

Procedure (Backtracking line search)

choose $\bar{\alpha} > 0, \rho \in (0, 1), c_1 = 10^{-4}$.

In the k -th step of iteration,

```
repeat until  $f(x_k + \alpha p_k) \leq f(x_k) + c_1 \alpha p_k^T \nabla f(x_k)$   
     $\alpha \leftarrow \rho \alpha$   
end(repeat)
```

When terminated, $\alpha_k \leftarrow \alpha$.

Algorithm (BFGS with line search)

$x_0, \epsilon > 0, \tilde{B}_0, k \leftarrow 0$

```
while  $\|\nabla f(x_k)\| > \epsilon$  :  
    compute  $p_k = -\tilde{B}_k \nabla f(x_k)$   
    line search for  $\alpha_k$   
     $x_{k+1} \leftarrow x_k + \alpha_k p_k$   
     $s_k \leftarrow x_{k+1} - x_k$   
     $y_k \leftarrow \nabla f(x_{k+1}) - \nabla f(x_k)$   
    compute  $\tilde{B}_{k+1}$   
     $k \leftarrow k + 1$   
end while
```