CS 2601 Linear and Convex Optimization

7. Newton's method

Bo Jiang

John Hopcroft Center for Computer Science
Shanghai Jiao Tong University

Fall 2022

Outline

- Newton's method and properties
- Analysis of Newton's method
- Damped Newton's method

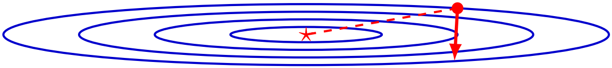
Better descent direction

Gradient descent uses first-order information (i.e. gradient),

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k - t_k \nabla f(\boldsymbol{x}_k)$$

Locally $-\nabla f(\boldsymbol{x}_k)$ is the max-rate descending direction, but globally it may not be the "right" direction.

Example. For $f(\boldsymbol{x}) = \frac{1}{2}\boldsymbol{x}^T \boldsymbol{Q} \boldsymbol{x}$ with $\boldsymbol{Q} = \text{diag}\{0.01, 1\}$, minimum is $\boldsymbol{x}^* = \boldsymbol{0}$.



The negative gradient is

$$-\nabla f(\boldsymbol{x}) = -\boldsymbol{Q}\boldsymbol{x} = -(0.01 x_1, x_2)^T$$

quite different from the "right" descent direction $\boldsymbol{d} = -\boldsymbol{x}$. Note

$$\boldsymbol{d} = -\boldsymbol{Q}^{-1} \nabla f(\boldsymbol{x}) = -[\nabla^2 f(\boldsymbol{x})]^{-1} \nabla f(\boldsymbol{x})$$

加入二阶信息

With second-order information (i.e. Hessian), we hope to do better.

Newton's method

Gradient step $\boldsymbol{x}_{k+1} = \boldsymbol{x}_k - t_k \nabla f(\boldsymbol{x}_k)$ can be interpreted as minimizing a quadratic approximation of f at $\boldsymbol{x}_k$,   具体内容写在黑板上

$$f(\boldsymbol{x}) \approx \hat{f}_{\text{gd}}(\boldsymbol{x}) \triangleq f(\boldsymbol{x}_k) + \nabla f(\boldsymbol{x}_k)^T(\boldsymbol{x} - \boldsymbol{x}_k) + \frac{1}{2t_k}\|\boldsymbol{x} - \boldsymbol{x}_k\|^2$$

Newton's method minimizes the second-order Taylor approximation,

$$f(\boldsymbol{x}) \approx \hat{f}_{\text{nt}}(\boldsymbol{x}) \triangleq f(\boldsymbol{x}_k) + \nabla f(\boldsymbol{x}_k)^T(\boldsymbol{x} - \boldsymbol{x}_k) + \frac{1}{2}(\boldsymbol{x} - \boldsymbol{x}_k)^T \nabla^2 f(\boldsymbol{x}_k)(\boldsymbol{x} - \boldsymbol{x}_k)$$

Newton step. Assuming $\nabla^2 f(\boldsymbol{x}_k) \succ \boldsymbol{O}$,   需要可逆

$$\boxed{\boldsymbol{x}_{k+1} = \boldsymbol{x}_k - [\nabla^2 f(\boldsymbol{x}_k)]^{-1} \nabla f(\boldsymbol{x}_k)}$$

Note. If f is quadratic, then $f = \hat{f}_{\text{nt}}$, and Newton's method gets to the minimum in a single step starting from any $\boldsymbol{x}_0$.

Newton's method (cont'd)

```
initialization x ← x_0 ∈ ℝ^n
while ‖∇f(x)‖ > δ do
    x ← x − [∇²f(x)]^{-1} ∇f(x)
end while
return x
```

$x \leftarrow x_0 \in \mathbb{R}^n$; **while** $\|\nabla f(\boldsymbol{x})\| > \delta$ **do** $\boldsymbol{x} \leftarrow \boldsymbol{x} - [\nabla^2 f(\boldsymbol{x})]^{-1}\nabla f(\boldsymbol{x})$ **end while**; **return** $\boldsymbol{x}$

Note. As in the case of gradient descent, other stopping criteria can be used. [BV] uses $\nabla f(\boldsymbol{x})[\nabla^2 f(\boldsymbol{x})]^{-1}\nabla f(\boldsymbol{x}) > \delta$.

The Newton step is a special case of $\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + t_k \boldsymbol{d}_k$ with

- Newton direction $\boldsymbol{d}_k = -[\nabla^2 f(\boldsymbol{x}_k)]^{-1}\nabla f(\boldsymbol{x}_k)$
- constant step size $t_k = 1$

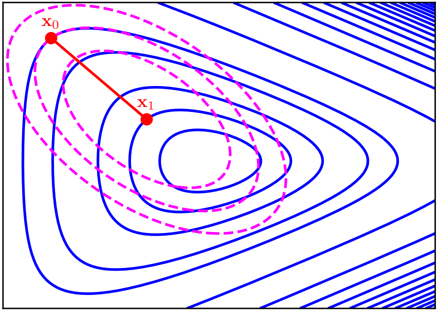
For $\nabla^2 f(\boldsymbol{x}_k) \succ \boldsymbol{O}$, the Newton direction is a descent direction

$$\nabla f(\boldsymbol{x}_k)^T \boldsymbol{d}_k = -\nabla f(\boldsymbol{x}_k)^T [\nabla^2 f(\boldsymbol{x}_k)]^{-1}\nabla f(\boldsymbol{x}_k) < 0 \quad \text{if } \nabla f(\boldsymbol{x}_k) \neq \boldsymbol{0}$$

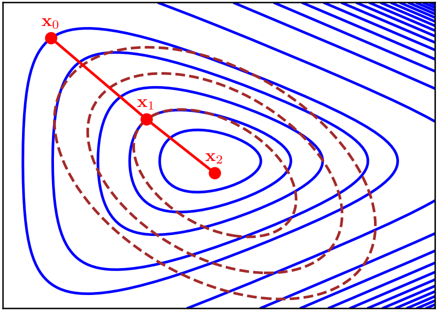
Newton's method (cont'd)

洋红色

The magenta curves are the level curves of the quadratic approximation of f at $\boldsymbol{x}_0$



The brown curves are the level curves of the quadratic approximation of f at $\boldsymbol{x}_1$.



Affine invariance

Given f, and invertible $A \in \mathbb{R}^{n \times n}$, let $g(y) = f(Ay)$. By the chain rule,

$$\nabla g(y) = A^T \nabla f(Ay), \qquad \nabla^2 g(y) = A^T \nabla^2 f(Ay) A$$

If we run Newton's method on f and g with $x_0 = Ay_0$, then

$$\begin{aligned}
y_1 &= y_0 - [\nabla^2 g(y_0)]^{-1} \nabla g(y_0) \\
&= y_0 - [A^T \nabla^2 f(x_0) A]^{-1} A^T \nabla f(x_0) \\
&= y_0 - A^{-1} [\nabla^2 f(x_0)]^{-1} \nabla f(x_0) \\
&= A^{-1} [x_0 - [\nabla^2 f(x_0)]^{-1} \nabla f(x_0)] \\
&= A^{-1} x_1
\end{aligned}$$

By induction $x_k = Ay_k$. Same progress independent of scaling by A.

For gradient descent, if $AA^T \neq I$, then in general,

$$x_1 = x_0 - t\nabla f(x_0) \neq Ay_1 = A(y_0 - tA^T \nabla f(y_0)) = x_0 - tAA^T \nabla f(x_0)$$

Connection to root finding

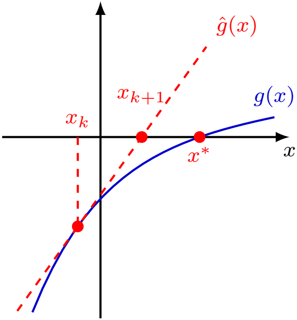
Newton's method is originally an algorithm for solving $g(x) = 0$.

By the first-order Taylor expansion,

$$g(x) \approx \hat{g}(x) \triangleq g(x_k) + g'(x_k)(x - x_k)$$

Use the root of $\hat{g}(x)$ as the next approximation

$$x_{k+1} = x_k - \frac{g(x_k)}{g'(x_k)}$$



Example (computing $\sqrt{C}$). $\sqrt{C}$ is a root of $g(x) = x^2 - C$. Newton's method yields

$$x_{k+1} = x_k - \frac{x_k^2 - C}{2x_k} = \frac{1}{2}\left(x_k + \frac{C}{x_k}\right)$$

For $x_0 > 0$, x_k converges to $\sqrt{C}$.

Connection to root finding (cont'd)

Back to the optimization problem,

$$x^* = \underset{x}{\operatorname{argmin}}\ f(x) \iff f'(x^*) = 0$$

Letting $g = f'$ in Newton's root finding algorithm,

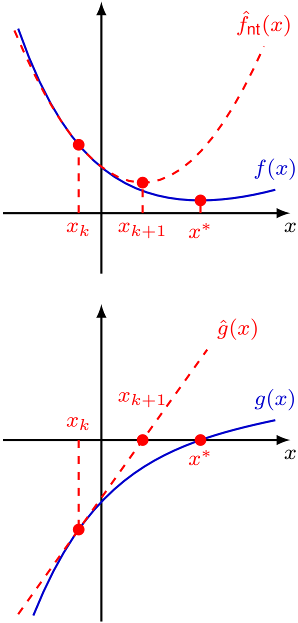
$$x_{k+1} = x_k - \frac{f'(x_k)}{f''(x_k)} = x_k - [f''(x_k)]^{-1} f'(x_k)$$

In n-dimension, $f' \to \nabla f$, $f'' \to \nabla^2 f$.

$$\boldsymbol{x}^* = \underset{\boldsymbol{x}}{\operatorname{argmin}}\ f(\boldsymbol{x}) \iff \nabla f(\boldsymbol{x}^*) = \mathbf{0}$$

Newton's algorithm becomes

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k - [\nabla^2 f(\boldsymbol{x}_k)]^{-1} \nabla f(\boldsymbol{x}_k)$$



Example

$$f(x_1, x_2) = e^{x_1+3x_2-0.1} + e^{x_1-3x_2-0.1} + e^{-x_1-0.1}$$

Newton step at $\boldsymbol{x}_0 = (-2, 1)^T$.

- gradient

$$\nabla f(\boldsymbol{x}_0) = e^{-0.1} \begin{pmatrix} e^{x_1+3x_2} + e^{x_1-3x_2} - e^{-x_1} \\ 3e^{x_1+3x_2} - 3e^{x_1-3x_2} \end{pmatrix} \Bigg|_{\boldsymbol{x}=\boldsymbol{x}_0} = \begin{pmatrix} -4.22019458 \\ 7.36051909 \end{pmatrix}$$

- Hessian

$$\nabla^2 f(\boldsymbol{x}_0) = e^{-0.1} \begin{pmatrix} e^{x_1+3x_2} + e^{x_1-3x_2} + e^{-x_1} & 3e^{x_1+3x_2} - 3e^{x_1-3x_2} \\ 3e^{x_1+3x_2} - 3e^{x_1-3x_2} & 9e^{x_1+3x_2} + 9e^{x_1-3x_2} \end{pmatrix} \Bigg|_{\boldsymbol{x}=\boldsymbol{x}_0}$$

$$= \begin{pmatrix} 9.1515943 & 7.36051909 \\ 7.36051909 & 22.19129872 \end{pmatrix}$$

- Newton direction

$$\nabla^2 f(\boldsymbol{x}_0)\boldsymbol{d} = -\nabla f(\boldsymbol{x}_0) \implies \boldsymbol{d} = (0.99274936, -0.66096491)^T$$

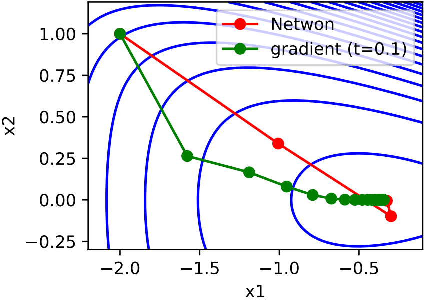
- Newton step

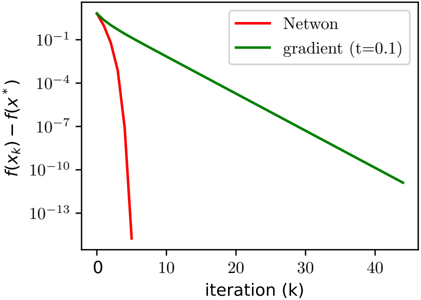
$$\boldsymbol{x}_1 = \boldsymbol{x}_0 + \boldsymbol{d} = (-1.00725064, 0.33903509)^T$$

Example (cont'd)

$$f(x_1, x_2) = e^{x_1+3x_2-0.1} + e^{x_1-3x_2-0.1} + e^{-x_1-0.1}$$

Solution using Newton's method and gradient descent with constant step size 0.1. Initial point $\boldsymbol{x}_0 = (-2, 1)^T$.





- Newton's method takes a more "direct" path
- Newton's method requires much fewer iterations, but each iteration is more expensive

Outline

- Newton's method and properties
- Analysis of Newton's method
- Damped Newton's method

Convergence of Newton's method

Example. Consider the minimization of $f(x) = \sqrt{1+x^2}$.

$$f'(x) = \frac{x}{\sqrt{1+x^2}}, \quad f''(x) = \frac{1}{(1+x^2)^{3/2}}$$

The Newton direction is

$$d_k = -\frac{f'(x_k)}{f''(x_k)} = -x_k - x_k^3$$

The Newton step is

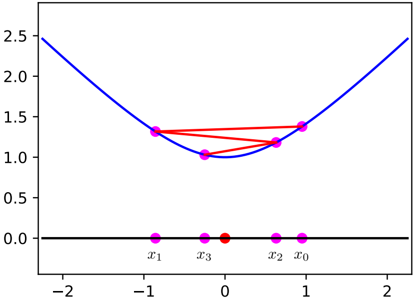
$$x_{k+1} = x_k + d_k = -x_k^3$$

Note $x_k \to x^* = 0$ iff $|x_0| < 1$.

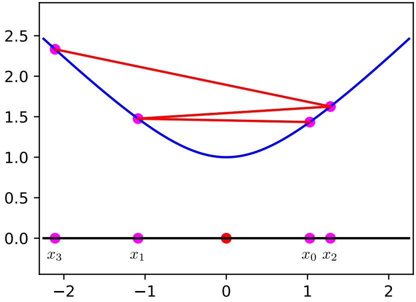
When $|x_0| > 1$, x_k diverges, and

$$f(x_{k+1}) > f(x_k)$$

Convergence of Newton's method (cont'd)



$\boldsymbol{x}_0 = 0.95$



$\boldsymbol{x}_0 = 1.02$

In general, Newton's method does not guarantee global convergence.
When it does converge, the convergence is usually very fast.

Convergence analysis: 1D case

Theorem. If f is m-strongly convex, f'' is M-Lipschitz continuous, and x^* is a minimum of f, then the sequence $\{x_k\}$ produced by Newton's method satisfies

$$|x_{k+1} - x^*| \leq \frac{M}{2m}|x_k - x^*|^2$$

Notes. Let $\xi_k = \frac{M}{2m}|x_k - x^*|$. The above inequality becomes $\xi_{k+1} \leq \xi_k^2$.

- If $\xi_k = 10^{-p}$, then $\xi_{k+1} \leq 10^{-2p}$, the number of significant digits doubles in each iteration!
- If $\xi_0 < 1$ i.e. $|x_0 - x^*| < \frac{2m}{M}$, then $\xi_k \leq \xi_0^{2^k}$ converges to 0 extremely fast. The number of iterations to ensure $\xi_k \leq \epsilon$ is $k \geq \log_2 \log_{\frac{1}{\xi_0}} \frac{1}{\epsilon}$.

  For $\epsilon = 10^{-p}$, $k \geq \log_2 p + \log_2 \log_{\frac{1}{\xi_0}} 10$, only logarithmic in the number of digits. Very few iterations are required!
- This theorem is a local convergence result. Fast convergence if x_0 is close enough to x^*, i.e. $|x_0 - x^*| < \frac{2m}{M}$. No guarantee if $|x_0 - x^*|$ is large.

和初始点选取很有关系！否则可能不会收敛

Proof: 1D case

f ′′ is M–Lipschitz continuous

$$
\begin{aligned}
& |x_{k+1} - x^*| \\
= & |x_k - x^* - [f''(x_k)]^{-1} f'(x_k)| && \text{Newton step} \\
= & |f''(x_k)|^{-1} \cdot |f'(x^*) - f'(x_k) - f''(x_k)(x^* - x_k)| && f'(x^*) = 0 \\
= & \frac{|x_k - x^*|}{|f''(x_k)|} \cdot \left| \int_0^1 [f''(x_k + t(x^* - x_k)) - f''(x_k)] dt \right| && \text{Newton-Leibniz} \\
\leq & \frac{|x_k - x^*|}{|f''(x_k)|} \cdot \int_0^1 |f''(x_k + t(x^* - x_k)) - f''(x_k)| dt && \left| \int f \right| \leq \int |f| \\
\leq & \frac{|x_k - x^*|}{|f''(x_k)|} \cdot \int_0^1 Mt|x_k - x^*| dt && M\text{-Lipschitz of } f'' \\
= & \frac{M}{2|f''(x_k)|} |x_k - x^*|^2 \\
\leq & \frac{M}{2m} |x_k - x^*|^2 && m\text{-strong convexity}
\end{aligned}
$$

用中值定理也可以做，就是得到的常数不一样

Matrix norm

Any function satisfies following conditions can be called matrix norm

The set of $m \times n$ matrices $\mathbb{R}^{m\times n}$ is a mn-dimensional vector space

A matrix norm on $\mathbb{R}^{m\times n}$ is a function $\|\cdot\| : \mathbb{R}^{m\times n} \to \mathbb{R}$ s.t.

1. $\|\boldsymbol{A}\| \geq 0$, $\forall \boldsymbol{A} \in \mathbb{R}^{m\times n}$
2. $\|\boldsymbol{A}\| = 0$ iff $\boldsymbol{A} = \boldsymbol{O}$
3. $\|c\boldsymbol{A}\| = |c| \cdot \|\boldsymbol{A}\|$, $\forall c \in \mathbb{R}, \boldsymbol{A} \in \mathbb{R}^{m\times n}$ (positive homogeneity)
4. $\|\boldsymbol{A} + \boldsymbol{B}\| \leq \|\boldsymbol{A}\| + \|\boldsymbol{B}\|$, $\forall \boldsymbol{A}, \boldsymbol{B} \in \mathbb{R}^{m\times n}$ (triangle inequality)

Example. The Frobenius norm on $\mathbb{R}^{m\times n}$ is the 2-norm on $\mathbb{R}^{mn}$.

$$\|\boldsymbol{A}\|_F = \sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n} a_{ij}^2} \quad \text{for } \boldsymbol{A} = (a_{ij}) \in \mathbb{R}^{m\times n}$$

Operator norm

定义了一种映射

A matrix $\boldsymbol{A} \in \mathbb{R}^{m\times n}$ defines a linear transformation from $\mathbb{R}^n$ to $\mathbb{R}^m$

$$\begin{aligned} \boldsymbol{A} : \mathbb{R}^n &\to \mathbb{R}^m \\ \boldsymbol{x} &\mapsto \boldsymbol{A}\boldsymbol{x} \end{aligned}$$

Given two vector norms $\|\cdot\|_a$ and $\|\cdot\|_b$ on $\mathbb{R}^n$ and $\mathbb{R}^m$, respectively, the operator norm or induced norm of $\boldsymbol{A}$ is defined by

$$\|\boldsymbol{A}\|_{a,b} = \max_{\boldsymbol{x}:\boldsymbol{x}\neq\boldsymbol{0}} \frac{\|\boldsymbol{A}\boldsymbol{x}\|_b}{\|\boldsymbol{x}\|_a} = \max_{\boldsymbol{x}:\|\boldsymbol{x}\|_a=1} \|\boldsymbol{A}\boldsymbol{x}\|_b = \max_{\boldsymbol{x}:\|\boldsymbol{x}\|_a\leq 1} \|\boldsymbol{A}\boldsymbol{x}\|_b$$

可以被证明

Exercise. Show the three definitions are equivalent.

The induced norm has the following important property.

Proposition (compatibility of norms).

$$\|\boldsymbol{A}\boldsymbol{x}\|_b \leq \|\boldsymbol{A}\|_{a,b}\|\boldsymbol{x}\|_a$$

Spectral norm

When the norms on $\mathbb{R}^n$ and $\mathbb{R}^m$ are both 2-norms, the induced norm on $\mathbb{R}^{n\times m}$ is simply called the 2-norm or spectral norm, denoted by $\|\cdot\|_2$.

Proposition.

$$\|\boldsymbol{A}\|_2 = \sqrt{\lambda_{\max}(\boldsymbol{A}^T\boldsymbol{A})},$$

where $\lambda_{\max}(\boldsymbol{A}^T\boldsymbol{A})$ is the maximum eigenvalue of $\boldsymbol{A}^T\boldsymbol{A}$.

Proof. Let $\|\boldsymbol{x}\|_2 = 1$. By slide 32 of §2,

$$\|\boldsymbol{A}\boldsymbol{x}\|_2^2 = \boldsymbol{x}^T\boldsymbol{A}^T\boldsymbol{A}\boldsymbol{x} \le \lambda_{\max}(\boldsymbol{A}^T\boldsymbol{A})\|\boldsymbol{x}\|_2^2 = \lambda_{\max}(\boldsymbol{A}^T\boldsymbol{A}), \quad \forall \boldsymbol{x} \in \mathbb{R}^n$$

with equality iff $\boldsymbol{x}$ is an eigenvector of $\boldsymbol{A}^T\boldsymbol{A}$ associated with $\lambda_{\max}(\boldsymbol{A}^T\boldsymbol{A})$.

Corollary. If $\boldsymbol{A}$ is symmetric,

$$\|\boldsymbol{A}\|_2 = \max\{|\lambda_{\max}(\boldsymbol{A})|, |\lambda_{\min}(\boldsymbol{A})|\}$$

If $\boldsymbol{A} \succeq \boldsymbol{O}$, then $\|\boldsymbol{A}\|_2 = \lambda_{\max}(\boldsymbol{A})$.

Examples

Example.

$$A = \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}$$

To find the 2-norm,

$$A^T A = \begin{pmatrix} 1 & 3 \\ 2 & 4 \end{pmatrix} \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix} = \begin{pmatrix} 10 & 14 \\ 14 & 20 \end{pmatrix}$$

$$\|A\|_2 = \sqrt{\lambda_{\max}(A^T A)} = \sqrt{15 + \sqrt{221}} \approx 5.465$$

Example.

$$A = \begin{pmatrix} 1 & 2 \\ 2 & 4 \end{pmatrix} \succeq O$$

$$\|A\|_2 = \sqrt{\lambda_{\max}(A^T A)} = \sqrt{\lambda_{\max}(A^2)} = \sqrt{\lambda_{\max}^2(A)} = \lambda_{\max}(A) = 5$$

Convergence analysis

$\nabla^2 f$ is M-Lipschitz continuous if

$$\|\nabla^2 f(\boldsymbol{x}) - \nabla^2 f(\boldsymbol{y})\|_2 \leq M\|\boldsymbol{x} - \boldsymbol{y}\|_2, \quad \forall \boldsymbol{x}, \boldsymbol{y}$$

Theorem. If f is m-strongly convex, $\nabla^2 f$ is M-Lipschitz continuous, and $\boldsymbol{x}^*$ is a minimum of f, then the sequence $\{\boldsymbol{x}_k\}$ produced by Newton's method satisfies

$$\|\boldsymbol{x}_{k+1} - \boldsymbol{x}^*\| \leq \frac{M}{2m}\|\boldsymbol{x}_k - \boldsymbol{x}^*\|^2$$

Note. The same remarks on slide 14 apply here with $|x_k - x^*|$ replaced by $\|\boldsymbol{x}_k - \boldsymbol{x}^*\|$. In particular, if $\|\boldsymbol{x}_0 - \boldsymbol{x}^*\| < \frac{2m}{M}$, then

$$\|\boldsymbol{x}_k - \boldsymbol{x}^*\| \leq \frac{2m}{M}\left(\frac{M}{2m}\|\boldsymbol{x}_0 - \boldsymbol{x}^*\|\right)^{2^k}$$

The proof is also very similar with only minor modifications.

Proof

$$
\begin{aligned}
& \|\boldsymbol{x}_{k+1} - \boldsymbol{x}^*\| \\
= {} & \|\boldsymbol{x}_k - \boldsymbol{x}^* - [\nabla^2 f(\boldsymbol{x}_k)]^{-1}[\nabla f(\boldsymbol{x}_k) - \nabla f(\boldsymbol{x}^*)]\| && (1) \\
\leq {} & \|[\nabla^2 f(\boldsymbol{x}_k)]^{-1}\| \cdot \|\nabla f(\boldsymbol{x}^*) - \nabla f(\boldsymbol{x}_k) - \nabla^2 f(\boldsymbol{x}_k)(\boldsymbol{x}^* - \boldsymbol{x}_k)\| && (2) \\
= {} & \|[\nabla^2 f(\boldsymbol{x}_k)]^{-1}\| \cdot \left\| \int_0^1 [\nabla^2 f(\boldsymbol{x}_k + t(\boldsymbol{x}^* - \boldsymbol{x}_k)) - \nabla^2 f(\boldsymbol{x}_k)](\boldsymbol{x}^* - \boldsymbol{x}_k) dt \right\| && (3) \\
\leq {} & \|[\nabla^2 f(\boldsymbol{x}_k)]^{-1}\| \int_0^1 \left\| [\nabla^2 f(\boldsymbol{x}_k + t(\boldsymbol{x}^* - \boldsymbol{x}_k)) - \nabla^2 f(\boldsymbol{x}_k)](\boldsymbol{x}^* - \boldsymbol{x}_k) \right\| dt && (4) \\
\leq {} & \|[\nabla^2 f(\boldsymbol{x}_k)]^{-1}\| \int_0^1 \left\| \nabla^2 f(\boldsymbol{x}_k + t(\boldsymbol{x}^* - \boldsymbol{x}_k)) - \nabla^2 f(\boldsymbol{x}_k) \right\| \cdot \|\boldsymbol{x}^* - \boldsymbol{x}_k\| dt && (5) \\
\leq {} & \|[\nabla^2 f(\boldsymbol{x}_k)]^{-1}\| \int_0^1 Mt \|\boldsymbol{x}^* - \boldsymbol{x}_k\|^2 dt && (6) \\
= {} & \|[\nabla^2 f(\boldsymbol{x}_k)]^{-1}\| \cdot \frac{M}{2} \|\boldsymbol{x}^* - \boldsymbol{x}_k\|^2 && (7) \\
\leq {} & \frac{M}{2m} \|\boldsymbol{x}_k - \boldsymbol{x}^*\|^2 && (8)
\end{aligned}
$$

Proof (cont'd)

1. Step (1) uses the Newton updating rule

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k - [\nabla^2 f(\boldsymbol{x}_k)]^{-1} \nabla f(\boldsymbol{x}_k)$$

   and the optimality condition $\nabla f(\boldsymbol{x}^*) = \mathbf{0}$.

2. Step (2) applies the compatibility of norms on slide 17 to

$$[\nabla^2 f(\boldsymbol{x}_k)]^{-1}[\nabla f(\boldsymbol{x}^*) - \nabla f(\boldsymbol{x}_k) - \nabla^2 f(\boldsymbol{x}_k)(\boldsymbol{x}^* - \boldsymbol{x}_k)]$$

3. Step (3) applies the Newton-Leibniz formula to the function $\boldsymbol{h}(t) = \nabla f(\boldsymbol{x}_k + t(\boldsymbol{x}^* - \boldsymbol{x}_k))$,

$$\nabla f(\boldsymbol{x}^*) - \nabla f(\boldsymbol{x}_k) = \boldsymbol{h}(1) - \boldsymbol{h}(0) = \int_0^1 \boldsymbol{h}'(t) dt$$

   where $\boldsymbol{h}'(t)$ is given by the chain rule,

$$\boldsymbol{h}'(t) = \nabla^2 f(\boldsymbol{x}_k + t(\boldsymbol{x}^* - \boldsymbol{x}_k))(\boldsymbol{x}^* - \boldsymbol{x}_k)$$

Proof (cont'd)

4. Step (4) uses the following inequality

$$\left\| \int \boldsymbol{f}(t)dt \right\| \leq \int \|\boldsymbol{f}(t)\|dt$$

   Proof. Let $\boldsymbol{z} = \int \boldsymbol{f}(t)dt$.

$$\|\boldsymbol{z}\|^2 = \boldsymbol{z}^T \int \boldsymbol{f}(t)dt \stackrel{(a)}{=} \int \boldsymbol{z}^T\boldsymbol{f}(t)dt \stackrel{(b)}{\leq} \int \|\boldsymbol{z}\| \cdot \|\boldsymbol{f}(t)\|dt = \|\boldsymbol{z}\| \int \|\boldsymbol{f}(t)\|dt,$$

   where (a) uses linearity of integration and (b) Cauchy-Schwarz.
5. Step (5) again applies the compatibility of norms on slide 17
6. Step (6) uses the Lipschitz continuity of $\nabla^2 f$
7. Step (7) performs the integration over t
8. Step (8) uses the m-strong convexity of f

$$\|[\nabla^2 f(\boldsymbol{x}_k)]^{-1}\| = \lambda_{\max}([\nabla^2 f(\boldsymbol{x}_k)]^{-1}) = \frac{1}{\lambda_{\min}(\nabla^2 f(\boldsymbol{x}_k))} \leq \frac{1}{m}$$

Outline

- Newton's method and properties
- Analysis of Newton's method
- Damped Newton's method

Damped Newton's method

The Newton direction $-[\nabla^2 f(\boldsymbol{x})]^{-1}\nabla f(\boldsymbol{x})$ is a descent direction, but with step size 1, Newton's method does not guarantee $f(\boldsymbol{x}_{k+1}) < f(\boldsymbol{x}_k)$.

To ensure $f(\boldsymbol{x}_{k+1}) < f(\boldsymbol{x}_k)$, damped Newton's method does backtracking line search along the Newton direction.

Damped Newton's method

converge globally

```
initialization x ← x_0 ∈ ℝ^n
while ‖∇f(x)‖ > δ do
    d ← −[∇²f(x)]^{−1} ∇f(x)          ▷ solve ∇²f(x)d = −∇f(x)
    t ← 1
    while f(x + td) > f(x) + αt∇f(x)^T d do
        t ← βt
    end while
    x ← x + td
end while
return x
```

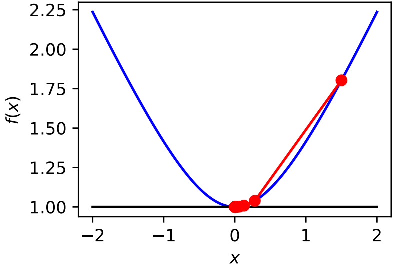
where $\alpha, \beta \in (0, 1)$

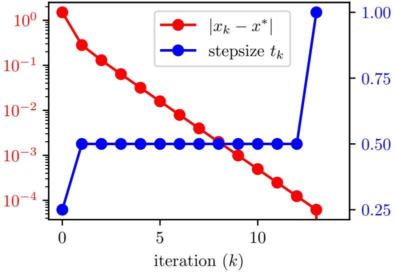
Example

$$f(x) = \sqrt{1 + x^2}$$

Recall pure Newton's method converges iff $|x_0| < 1$.

Damped Newton's method converges globally, e.g. for $x_0 = 1.5$.





一旦到了下一个阶段，就会进入
纯牛顿法，会很快，第一阶段比较慢

这里stepsize变为1了，其实
基本分为two phaze

Convergence analysis

Theorem. Assume f is m-strongly convex and L-smooth, $\nabla^2 f$ is M-Lipschitz, and $\boldsymbol{x}^*$ is a minimum of f. Damped Newton's method satisfies the following error bounds

$$f(\boldsymbol{x}_k) - f(\boldsymbol{x}^*) \le \begin{cases} f(\boldsymbol{x}_0) - f(\boldsymbol{x}^*) - \gamma k, & \text{if } k \le k_0 \\ \frac{2m^3}{M^2}\left(\frac{1}{2}\right)^{2^{k-k_0+1}}, & \text{if } k > k_0 \end{cases}$$

where $\gamma = 2\alpha\bar{\alpha}\beta\eta^2 m/L^2$, $\eta = \min\{1, 3(1-2\alpha)\}m^2/M$, and k_0 is the number of steps until $\|\nabla f(\boldsymbol{x}_{k_0+1})\| \le \eta$. 其实这就告诉我们此时离optima还很远

Notes.

- Damped Newton's method guarantees global convergence.
- To get $f(\boldsymbol{x}_k) - f(\boldsymbol{x}^*) \le \epsilon$, we need at most

第二阶段

第一阶段 $$\frac{f(\boldsymbol{x}_0) - f(\boldsymbol{x}^*)}{\gamma} + \log_2 \log_2 \frac{\epsilon_0}{\epsilon}$$

where $\epsilon_0 = \frac{2m^3}{M^2}$. It can be slow if γ is small.

Convergence analysis (cont'd)

Detailed analysis shows that the convergence follows two stages

- Damped Newton phase. When $\|\nabla f(\boldsymbol{x}_k)\| > \eta$, backtracking selects a step size $t_k \le 1$, and

$$f(\boldsymbol{x}_{k+1}) - f(\boldsymbol{x}_k) \le -\gamma$$

  Summing over k from 0 to $k_0 - 1$,

$$f(\boldsymbol{x}^*) - f(\boldsymbol{x}_0) \le f(\boldsymbol{x}_{k_0}) - f(\boldsymbol{x}_0) \le -k_0\gamma \implies k_0 \le \frac{f(\boldsymbol{x}_0) - f(\boldsymbol{x}^*)}{\gamma}$$

- Pure Newton phase. When $\|\nabla f(\boldsymbol{x}_k)\| \le \eta$, backtracking selects step size $t_k = 1$, and

$$\|\nabla f(\boldsymbol{x}_{k+1})\| \le \frac{M}{2m^2}\|\nabla f(\boldsymbol{x}_k)\|^2 \le \frac{1}{2}\|\nabla f(\boldsymbol{x}_k)\| \le \eta$$

  By induction, $\|\nabla f(\boldsymbol{x}_k)\| \le \eta$ for all $k \ge k_0$, so we will stay in the pure Newton phase with $t_k = 1$.