C2: Methods for non-linear problems
The Newton method

Niclas Börlin

5DA001 Non-linear Optimization

Questions

Consider the general, non-linear, continuous minimization problem

$$\min_{x \in \Re^n} f(x).$$

1. How do we handle the fact that $f(x)$ is non-linear?
2. How do we construct a solver for a minimization problem?
3. How do we ensure that an algorithm converges...
   - ...if we start "close" to the solution?
   - ...even if we start "far" from the solution?
4. How do we compare optimization algorithms?

Convergence rate

Convergence rate
Motivation

- In order to compare different iterative methods, we need a measure of efficiency.
- The number of iterations varies, so computational complexity cannot be used.
- Instead the concept of a convergence rate is defined.

Convergence rate
Definition

- Assume we have a series $\{x_k\}$ that converges to a solution x^*. Define the sequence of errors as

$$e_k = x_k - x^*$$

  and note that

$$\lim_{k\to\infty} e_k = 0.$$

- We say that the sequence $\{x_k\}$ converges to x^* with rate r and rate constant C if

$$\lim_{k\to\infty} \frac{\|e_{k+1}\|}{\|e_k\|^r} = C$$

  and $C < \infty$.

Convergence rate
In practice

In practice there are three important rates of convergence:

- linear convergence, for $r = 1$ and $0 < C < 1$;
- quadratic convergence, for $r = 2$.
- super-linear convergence, for $r = 1$ and $C = 0$.

Linear convergence
Examples

- Linear convergence is controlled by the constant C:
  - For $r = 1$, $C = 0.1$ and $\|e_0\| = 1$, the norm of the error sequence becomes

$$\underbrace{1, 10^{-1}, 10^{-2}, \ldots, 10^{-7}}_{\text{7 iterations}}$$

  - For $C = 0.99$ the corresponding sequence is

$$\underbrace{1, 0.9, 0.9801, \ldots, 0.997 \cdot 10^{-7}}_{\text{1604 iterations}}.$$

Quadratic convergence
Examples

- Quadratic convergence is controlled by the starting error $\|e_0\|$:
  - For $r = 2$, $C = 0.1$ och $\|e_0\| = 1$, the sequence becomes

$$1, 10^{-1}, 10^{-3}, 10^{-7}, \ldots$$

  - For $r = 2$, $C = 3$ och $\|e_0\| = 1$, the sequence diverges

$$1, 3, 27, \ldots$$

  - For $r = 2$, $C = 3$ och $\|e_0\| = 0.1$, the sequence becomes

$$0.1, 0.03, 0.0027, \ldots,$$

    i.e. it converges despite $C > 1$.

Local and global convergence
Definition

- A method is called locally convergent if it produces a convergent sequence toward a minimizer x^* provided a close enough starting approximation.
- A method is called globally convergent if it produces a convergent sequence toward a minimizer x^* provided any starting approximation.
- Note that global convergence does *not* imply convergence towards a global minimizer.

Questions

Consider the general, non-linear, continuous minimization problem

$$\min_{x \in \Re^n} f(x).$$

1. How do we handle the fact that $f(x)$ is non-linear?
   - Approximate $f(x)$ by a sequence of simpler functions!
2. How do we construct a solver for a minimization problem?
   - Formulate and solve an equation (system) based on the first order conditions!
   - For every iteration k, use the solution as the next point x_{k+1}.
3. How do we ensure that an algorithm converges...
   - ...if we start "close" to the solution?
   - ...even if we start "far" from the solution?
4. How do we compare optimization algorithms?

Déjà vu all over again

Hold on! Haven't I heard about simplify-solve-iterate before...?

The Newton-Raphson method in $\Re^1$

The Newton-Raphson method in $\Re^1$

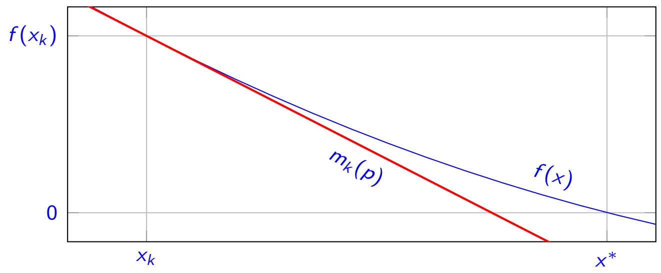
Consider the non-linear problem

$$f(x) = 0,$$

where $f, x \in \Re$.

1. Replace the function f by a simpler model function m_k; its linear Taylor approximation around x_k

$$f(x_k + p) \approx f(x_k) + pf'(x_k) = m_k(p).$$



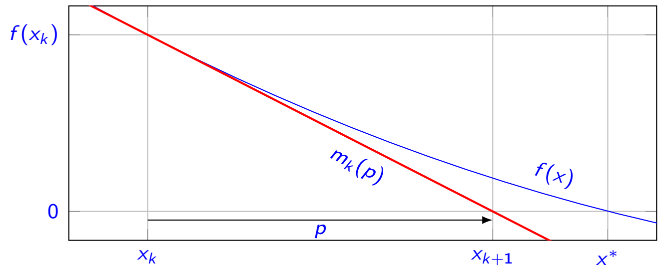
The Newton-Raphson method in $\Re^1$ (Cont'd)

2a. Find the solution for the model function, i.e. solve

$$m_k(p) = 0 = f(x_k) + pf'(x_k).$$

for p and get

$$p = -f(x_k)/f'(x_k).$$



- In general, the new iterate is given by

$$x_{k+1} = x_k + p_k = x_k - f(x_k)/f'(x_k).$$

2b. Assume the new point is better. (*What!? Yep, sometimes it fails, but we'll deal with that later.*)

But the Newton-Raphson method is for $f(x) = 0$, not $\min f(x)$...?

The Newton method for minimization in $\Re^n$

The classical Newton minimization method in $\Re^n$

- Apply the first-order necessary conditions on a function $f : \Re^n \to \Re$:

$$\nabla f(x^*) = 0, \qquad\qquad f'(x^*) = 0.$$

- This results in the Newton sequence

$$x_{k+1} = x_k - (\nabla^2 f(x_k))^{-1} \nabla f(x_k), \quad x_{k+1} = x_k - f'(x_k)/f''(x_k).$$

- This is often written as

$$x_{k+1} = x_k + p_k,$$

where p_k is the solution of the Newton equation:

$$\nabla^2 f(x_k) p_k = -\nabla f(x_k).$$

The Newton equation

$$\boxed{\nabla^2 f(x_k) p_k = -\nabla f(x_k)}$$

Hey, this is the 300+ year old Original! Beware of cheap copies!

Geometrical interpretation

- The approximation of the non-linear function $\nabla f(x)$ with the linear (in p) polynomial

$$\nabla f(x_k + p) \approx \nabla f(x_k) + \nabla^2 f(x_k) p$$

corresponds to approximating the non-linear function $f(x)$ with the quadratic (in p) Taylor expansion

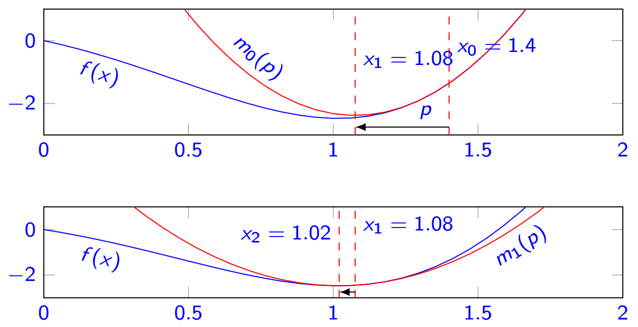
$$m_k(x_k + p) \equiv f(x_k) + \nabla f(x_k)^T p + \frac{1}{2} p^T \nabla^2 f(x_k) p.$$

- Thus, at each iteration, Newton's method:
  1. approximates f by its quadratic Taylor expansion m_k around x_k, and
  2. calculates x_{k+1} as the stationary point of m_k.

1D function — $f(x) = -e^x \sin 2x$

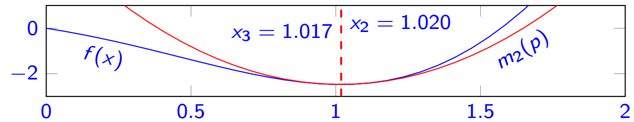
$$p = \frac{-f'(x_0)}{f''(x_0)} = \frac{-6.28}{19.36} = -0.32$$

$$p = \frac{-f'(x_1)}{f''(x_1)} = \frac{-0.76}{13.78} = -0.06$$

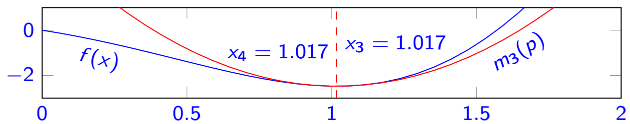


1D function — $f(x) = -e^x \sin 2x$

$$p = \frac{-f'(x_2)}{f''(x_2)} = \frac{-0.04}{12.44} = -0.003$$



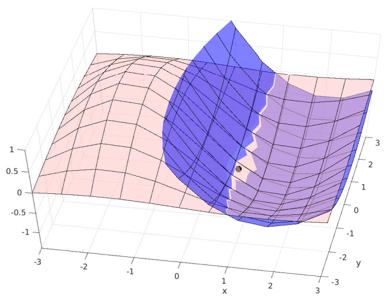
$$p = \frac{-f'(x_3)}{f''(x_3)} = \frac{-0.0001}{12.37} = -9 \cdot 10^{-6}$$

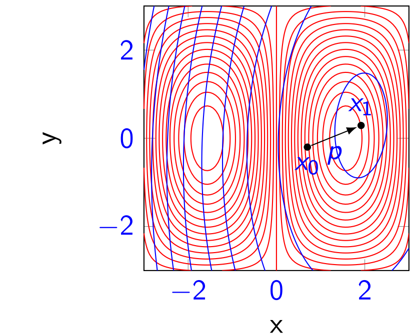


2D function — $f(x) = -\sin x_1 \cos x_2/2$

$$x_0 = \begin{pmatrix} 0.70 \\ -0.20 \end{pmatrix}, \nabla f(x_0) = \begin{pmatrix} -0.76 \\ -0.03 \end{pmatrix}, \nabla^2 f(x_0) = \begin{pmatrix} 0.64 & -0.04 \\ -0.04 & 0.16 \end{pmatrix},$$

$$p = \nabla^2 f(x_0)^{-1}(-\nabla f(x_0)) = \begin{pmatrix} 1.22 \\ 0.49 \end{pmatrix}, x_1 = x_0 + p = \begin{pmatrix} 1.92 \\ 0.29 \end{pmatrix}.$$

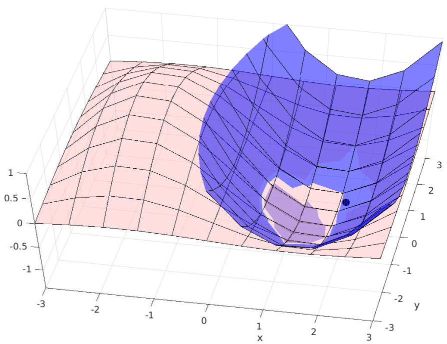


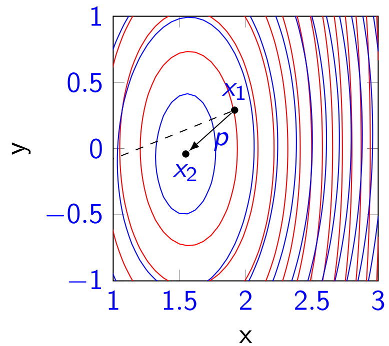


2D function — $f(x) = -\sin x_1 \cos x_2/2$

$$x_1 = \begin{pmatrix} 1.92 \\ 0.29 \end{pmatrix}, \nabla f(x_1) = \begin{pmatrix} 0.34 \\ 0.07 \end{pmatrix}, \nabla^2 f(x_1) = \begin{pmatrix} 0.93 & -0.02 \\ -0.02 & 0.23 \end{pmatrix},$$

$$p = \nabla^2 f(x_1)^{-1}(-\nabla f(x_1)) = \begin{pmatrix} -0.37 \\ -0.33 \end{pmatrix}, x_2 = x_1 + p = \begin{pmatrix} 1.55 \\ -0.04 \end{pmatrix}.$$

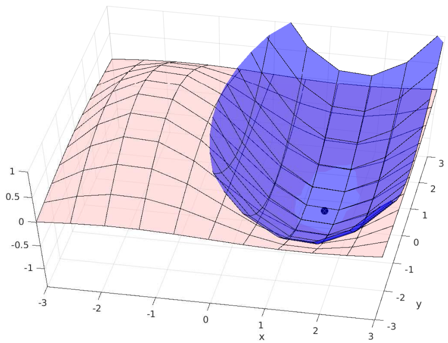


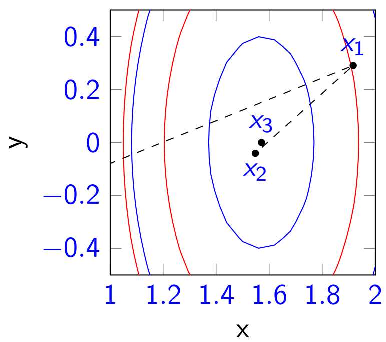


2D function — $f(x) = -\sin x_1 \cos x_2/2$

$$x_2 = \begin{pmatrix} 1.55 \\ -0.04 \end{pmatrix}, \nabla f(x_2) = \begin{pmatrix} -0.02 \\ -0.01 \end{pmatrix}, \nabla^2 f(x_2) = \begin{pmatrix} 1.00 & -0.00 \\ -0.00 & 0.25 \end{pmatrix},$$

$$p = \nabla^2 f(x_2)^{-1}(-\nabla f(x_2)) = \begin{pmatrix} 0.02 \\ 0.04 \end{pmatrix}, x_3 = x_2 + p = \begin{pmatrix} 1.57 \\ 0.00 \end{pmatrix}.$$

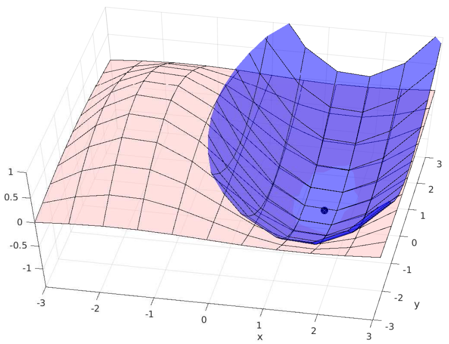


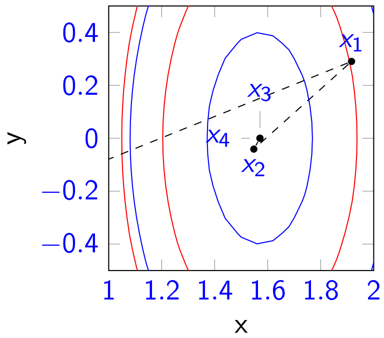


2D function — $f(x) = -\sin x_1 \cos x_2/2$

$$x_3 = \begin{pmatrix}1.57\\0.00\end{pmatrix}, \nabla f(x_3) = \begin{pmatrix}0.00\\0.00\end{pmatrix}, \nabla^2 f(x_3) = \begin{pmatrix}1.00 & -0.00\\-0.00 & 0.25\end{pmatrix},$$

$$p = \nabla^2 f(x_3)^{-1}(-\nabla f(x_3)) = \begin{pmatrix}-0.00\\-0.00\end{pmatrix}, x_4 = x_3 + p = \begin{pmatrix}1.57\\-0.00\end{pmatrix}.$$





Convergence of the Newton method

- For the Newton method:

$$\begin{aligned}
0 = f(x_*) &= f(x_k - e_k) = f(x_k) - e_k f'(x_k) + \frac{1}{2}e_k^2 f''(\xi),\\
-\frac{f(x_k) - e_k f'(x_k)}{f'(x_k)} &= \frac{1}{2}e_k^2\frac{f''(\xi)}{f'(x_k)},\\
e_k - \frac{f(x_k)}{f'(x_k)} &= \frac{1}{2}e_k^2\frac{f''(\xi)}{f'(x_k)},\\
\underbrace{x_k - \frac{f(x_k)}{f'(x_k)}}_{x_{k+1}} - x_* &= \frac{1}{2}e_k^2\frac{f''(\xi)}{f'(x_k)},\\
e_{k+1} &= \frac{1}{2}e_k^2\frac{f''(\xi)}{f'(x_k)},\\
\lim_{k\to\infty}\frac{\|e_{k+1}\|}{\|e_k\|^2} &= \frac{1}{2}\left|\frac{f''(x_*)}{f'(x_*)}\right|.
\end{aligned}$$

- Thus, the Newton method will have quadratic convergence if $f'(x_*) \neq 0$ and $f''(x_*)$ is bounded.

Convergence of the Newton method

- The Newton method converges quadratically[1] towards a stationary point[2] if the starting approximation is close enough[3].

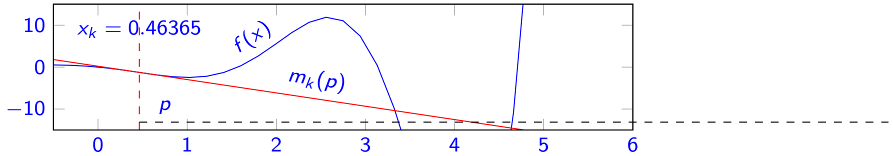
[1] *Wow, that's fast!*
[2] *Well, that's what we want, right?*
[3] *Hmm. . . , is that a problem?*

Problems with the Newton method

Problems with the Newton method (1)

- The Newton method will fail if $\nabla^2 f(x_k)$ is non-invertible for some k.

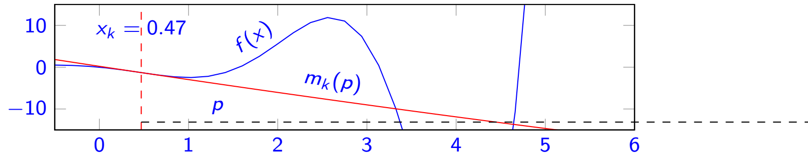
$$p = \frac{-f'(x_k)}{f''(x_k)} = \frac{3.18}{0} = \infty$$



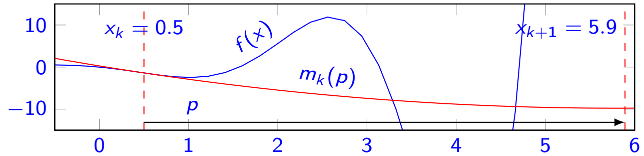
Problems with the Newton method (2)

- The Newton method will generate large updates *near* points where $\nabla^2 f(x_k)$ is non-invertible.

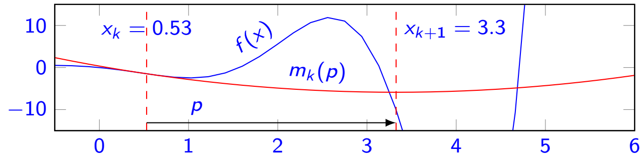
$$p = \frac{-f'(x_k)}{f''(x_k)} = \frac{3.18}{0.10} = 31$$



$$p = \frac{-f'(x_k)}{f''(x_k)} = \frac{3.17}{0.60} = 5.3$$



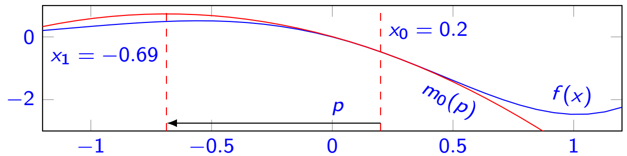
$$p = \frac{-f'(x_k)}{f''(x_k)} = \frac{3.14}{1.12} = 2.8$$



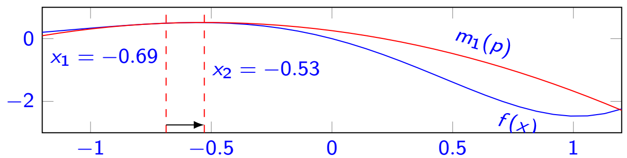
Problems with the Newton method (3)

- The Newton method converges towards a stationary point, not necessarily a minimizer.

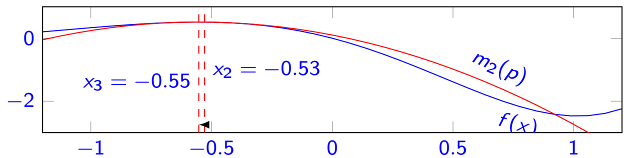
$$p = \frac{-f'(x_0)}{f''(x_0)} = \frac{2.73}{-3.07} = -0.89$$



$$p = \frac{-f'(x_1)}{f''(x_1)} = \frac{-0.30}{-1.87} = 0.16$$



$$p = \frac{-f'(x_2)}{f''(x_2)} = \frac{0.07}{-2.70} = -0.02$$



Problems with the Newton method (4)

- The Newton method requires explicit second-order information in the Hessian $\nabla^2 f(x_k)$.

  (Well, that's kind of obvious. Why is it a problem?)
- The second derivatives are generally complex.
  - They may take a long time to derive and implement.
  - Both steps may introduce bugs.

  (Nah, I have plenty of time and never make mstakes!)
- For large n, the Hessian
  - may be expensive to compute,
  - may require a lot of memory to store.

  (Ok, can't argue with that.)

If the Newton method has so many problems, why have we wasted time to talk about it?

Properties of the Newton method (summary)

- The Newton method converges quadratically towards a stationary point if the starting approximation is close enough. *(Yeah, hallelujah, you said so, but what about the problems...?)*
- It is possible to modify the Newton method (cheaply) to avoid most of the problems and still get quick[4] convergence. *(Ok...)*
- Furthermore, some methods use approximations of the Hessian and are faster[5] than Newton's method! *(That's neat, I guess, but why didn't we talk about them instead?)*
- In fact, many common optimization methods can be seen as approximations of the Newton method. The methods try to emulate the positive properties while avoiding the negative.

[4] measured in number of iterations
[5] measured in execution time

Convergence and the Newton method

- In other words:
  - The Newton method is locally convergent.
  - The Newton method is not globally convergent.

Questions

Consider the general, non-linear, continuous minimization problem

$$\min_{x \in \Re^n} f(x).$$

1. How do we handle the fact that $f(x)$ is non-linear?
2. How do we construct a solver for a minimization problem?
3. How do we ensure that an algorithm...
   - ...converges if we start "close" to the solution? ...becomes locally convergent?
     - Use one that is! The Newton method!
   - ...converges even if we start "far" from the solution? ...becomes globally convergent?
4. How do we compare optimization algorithms?

How to make the Newton method globally convergent (Part 1)

Descent methods and directions

- A descent method is a method that guarantees that

$$f(x_{k+1}) < f(x_k), k = 0, 1, \ldots$$

- A descent direction is a direction p_k such that

$$f(x_k + \alpha p_k) < f(x_k),$$

for some small value of $\alpha > 0$.

Descent directions

- Consider the Taylor expansion of the objective function along a search direction p

$$f(x_k + \alpha p) = f(x_k) + \alpha p^T \nabla f_k + \frac{1}{2}\alpha^2 p^T \nabla^2 f(x_k + \tau p)p,$$
$$\text{for some } \tau \in (0, \alpha)$$

- Any direction p such that $p^T \nabla f_k < 0$ will produce a reduction of the objective function for a short enough step.
- A direction p such that

$$p^T \nabla f_k < 0$$

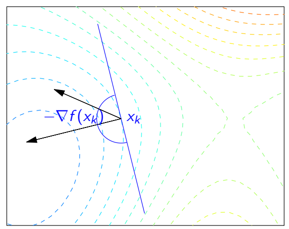
is a descent direction.

Descent directions

- Since

$$\cos\theta = \frac{-p^T \nabla f_k}{\|p\| \|\nabla f_k\|}$$

is the angle between the search direction and the negative gradient, descent directions are in the same half-plane as the negative gradient.
- The search direction corresponding to the negative gradient $p = -\nabla f_k$ is called the direction of steepest descent.



Descent Newton

- We will modify the Newton method to be a descent method.
- We will do this by splitting each iteration into two subproblem:
  - Compute a search direction p_k that is a descent direction (and more).
  - Perform a line search to compute a step length α such that

$$f(x_k + \alpha p_k) < f(x_k)$$

  (and more).

Why $f(x_{k+1}) < f(x_k)$ is not enough

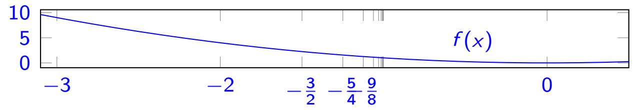
- Consider the minimization problem

$$\min_x f(x) = x^2.$$

- Assume that at each iteration, the search direction $p_k = 1$ and the step length $\alpha_k = 2^{-k}$. Hence,

$$x_{k+1} = x_k + 2^{-k}.$$

- Starting with $x_0 = -3$, the sequence becomes $-3, -2, -\frac{3}{2}, -\frac{5}{4}, -\frac{9}{8}$, with $x_k = -(1 + 2^{1-k})$.



Why $f(x_{k+1}) < f(x_k)$ is not enough

- The method does not converge to the minimizer $x_* = 0$!
- Each search direction is a descent direction since

$$p_k^T \nabla f(x_k) = 1 \cdot 2x_k = -2(1 + 2^{1-k}) < 0.$$

- However, the sequence does not converge to a stationary point!

$$\lim_{k \to \infty} x_k = -1$$

  and $f'(-1) = -2 \neq 0$.
- Thus, the condition that $f(x_{k+1}) < f(x_k)$ is not enough to guarantee convergence.

Globally convergent methods

Requirements

- One way to guarantee global convergence is to place a few requirement on the search direction and the step length:
  1. Each search direction p_k produces "sufficient descent".
  2. Each search direction p_k is "gradient related".
  3. Each step length α_k produces "sufficient decrease".
  4. Each step length α_k is not "too small".

Conditions on the search direction — the angle condition

- The "sufficient descent" condition corresponds to that $p_k^T \nabla f(x)$ cannot be arbitrarily close to 0.
- Instead, it must satisfy

$$-\frac{p_k^T \nabla f(x_k)}{\|p_k\| \cdot \|\nabla f(x_k)\|} \geq \varepsilon > 0,$$

  for all k and some $\varepsilon > 0$.
- This condition may be re-written as

$$\cos\theta \geq \varepsilon > 0,$$

  where θ is the angle between the search direction p_k and the negative gradient $-\nabla f(x_k)$.
- This condition is sometimes referred to as the *angle condition* and means that the angle between the search direction p_k and the negative gradient $-\nabla f(x_k)$ may not become arbitrarily close to being orthogonal.

Conditions on the search direction — gradient related

- The search direction p_k is said to be *gradient related* if

$$\|p_k\| \geq m\|\nabla f(x_k)\| \text{ or } \frac{\|p_k\|}{\|\nabla f(x_k)\|} \geq m > 0$$

  for all k and some $m > 0$.
- This condition states that the search direction may not be arbitrarily short compared to the gradient.

Sufficient descent and the Newton method

- Denote the gradient $\nabla f(x_k)$ by g and the Hessian $\nabla^2 f(x_k)$ by H. Let $\|\cdot\| = \|\cdot\|_2$.
- For the Newton method,

$$p_k = -(\nabla^2 f(x_k))^{-1} \nabla f(x_k) = -H^{-1} g.$$

- Thus, the sufficient descent condition becomes

$$-\frac{p_k^T \nabla f(x_k)}{\|p_k\| \|\nabla f(x_k)\|} = -\frac{-g^T H^{-1} g}{\|H^{-1} g\| \|g\|} = \frac{g^T H^{-1} g}{\|H^{-1} g\| \|g\|}.$$

- If H is positive definite, H will have positive eigenvalues, i.e. the exists a factorization $H = V \Lambda V^T$, where Λ is a diagonal matrix with positive diagonal element and $V^T V = I$.
- Then $H^{-1} = V \Lambda^{-1} V^T$.

Sufficient descent and the Newton method

- Let $\hat{g} = g / \|g\|$. Then

$$\frac{g^T H^{-1} g}{\|H^{-1} g\| \|g\|} = \frac{\hat{g}^T V \Lambda^{-1} V^T \hat{g} \|g\|^2}{\|V \Lambda^{-1} V^T \hat{g}\| \|g\|^2} = [u = V^T \hat{g}]$$
$$= \frac{u^T \Lambda^{-1} u}{\|V \Lambda^{-1} u\|} = \frac{u^T \Lambda^{-1} u}{\|\Lambda^{-1} u\|} \geq \frac{1/\lambda_{max}}{1/\lambda_{min}} = \frac{\lambda_{min}}{\lambda_{max}},$$

  where $\lambda_{max} = \max \lambda_{ii}$ and $\lambda_{min} = \min \lambda_{ii}$ are the largest and smallest eigenvalues of H, respectively.
- Thus, if $H = \nabla^2 f(x_k)$ is $\boxed{\text{positive definite}}$ for all k and

$$\frac{\ell_{min}}{\ell_{max}} = \frac{\min_k \lambda_{min}}{\max_k \lambda_{max}} = \epsilon > 0,$$

  i.e. the eigenvalues of H are $\boxed{\text{bounded from above and below}}$ for all k, then the Newton direction is guaranteed to be a sufficient descent direction.

Gradient related and the Newton method

- Similarly, for the gradient related condition:

$$\frac{\|p_k\|}{\|\nabla f(x_k)\|} = \frac{\|H^{-1}g\|}{\|g\|} = \frac{\|V\Lambda^{-1}V^T\hat{g}\|\|g\|}{\|g\|}$$
$$= \|V\Lambda^{-1}u\| = \|\Lambda^{-1}u\| \geq \frac{1}{\lambda_{max}}.$$

- Thus, if $H = \nabla^2 f(x_k)$ is $\boxed{\text{positive definite}}$ for all k and the eigenvalues are $\boxed{\text{bounded from above}}$

$$\frac{1}{\ell_{max}} = \frac{1}{\max_k \lambda_{max}} = m > 0,$$

then the Newton direction is gradient related.

The Newton direction and descent

- The Newton search direction p^N is written as

$$p^N = -B_k^{-1}\nabla f_k,$$

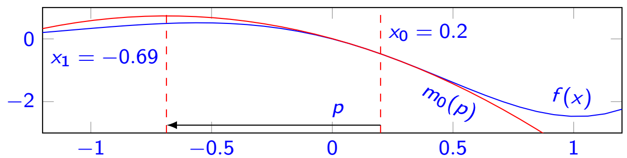
with $B_k = \nabla^2 f_k$.
- Thus, p^N will be a descent direction if $\nabla^2 f_k$ is positive definite.
- If $\nabla^2 f_k$ is *not* positive definite, the Newton direction p^N may not a descent direction.
- In that case we may choose B_k as a positive definite approximation of $\nabla^2 f_k$.

The Newton direction and descent

Modifying the Hessian

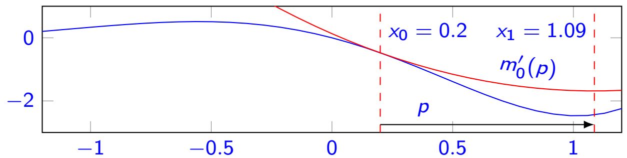
- If the Hessian is not positive definite, approximate it by another matrix B_k that is:
- Without modification:

$$p = \frac{-f'(x_0)}{f''(x_0)} = \frac{2.73}{-3.07} = -0.89$$



- With modification (1):

$$p = \frac{-f'(x_0)}{|f''(x_0)|} = \frac{2.73}{3.07} = 0.89$$

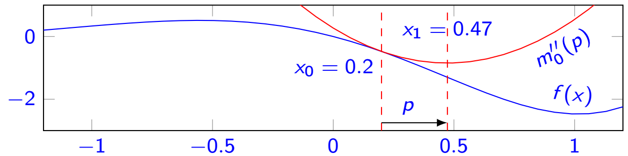


The Newton direction and descent

Modifying the Hessian

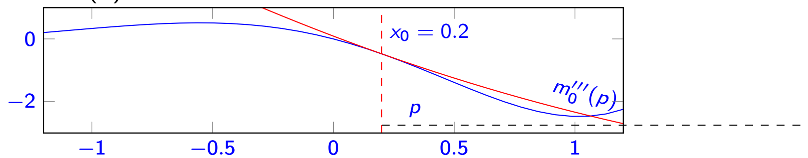
- With modification (2):

$$p = \frac{-f'(x_0)}{100} = \frac{2.73}{10} = 0.27$$



- With modification (3):

$$p = \frac{-f'(x_0)}{1} = 2.73$$



The Newton direction and descent

Computation

- The positive definite approximation B_k of the Hessian may be found with minimal extra effort: The search direction p is calculated as the solution of

$$\nabla^2 f(x)p = -\nabla f(x).$$

- If $\nabla^2 f(x)$ is positive definite, the matrix factorization

$$\nabla^2 f(x) = LDL^T$$

  may be used, where the diagonal elements of D are positive.
- If $\nabla^2 f(x)$ is *not* positive definite, at some point during the factorization, a diagonal element will be $d_{ii} \leq 0$.
- In this case, the d_{ii} may be replaced with a suitable positive entry.
- Finally, the factorization is used to calculate the search direction

$$(LDL^T)p = -\nabla f(x).$$