# Analysis of Newton's Method

Oscar Dalmau
dalmau@cimat.mx

Centro de Investigación en Matemáticas
CIMAT A.C. Mexico

March 2, 2021

# Outline

# Steepest descent Method: Step size

## Steepest descent: Step size

1. In the Steepest descent Method with exact line search two consecutive directions are orthogonal, i.e. $\boldsymbol{g}_k \perp \boldsymbol{g}_{k+1}$.
2. The solution trajectory of the steepest-descent method with exact line search follows a zig-zag pattern.
3. Cuadratic case (exact step size): $\alpha_k = \frac{\boldsymbol{g}_k^T \boldsymbol{g}_k}{\boldsymbol{g}_k^T \mathbf{Q} \boldsymbol{g}_k}$
4. General case (step size): $\alpha_k = \frac{\boldsymbol{g}_k^T \boldsymbol{g}_k}{\boldsymbol{g}_k^T \mathbf{H}_k \boldsymbol{g}_k}$
5. General case (step size approximation): $\alpha_k = \frac{\boldsymbol{g}_k^T \boldsymbol{g}_k \hat{\alpha}^2}{2(\hat{f} - f_k + \hat{\alpha} \boldsymbol{g}_k^T \boldsymbol{g}_k)}$

   with $\hat{f} = f(x_k - \hat{\alpha} \boldsymbol{g}_k)$

# Steepest descent Method: Cuadratic case

## Steepest descent: Cuadratic case

1. In the steepest decent algorithm $\boldsymbol{x}_{k+1} = \boldsymbol{x}_k - \alpha_k \boldsymbol{g}_k$, with exact line search, i.e. $\alpha_k = \frac{\boldsymbol{g}_k^T \boldsymbol{g}_k}{\boldsymbol{g}_k^T \mathbf{Q} \boldsymbol{g}_k}$, we have that $\boldsymbol{x}_k \to \boldsymbol{x}^*$ for any $\boldsymbol{x}_0$, i.e. converges globally.
2. In the steepest decent algorithm $\boldsymbol{x}_{k+1} = \boldsymbol{x}_k - \alpha \boldsymbol{g}_k$, with with fixed step size, i.e. $\alpha_k = \alpha$ for all k, we have that $\boldsymbol{x}_k \to \boldsymbol{x}^*$ for any $\boldsymbol{x}_0$ iff $0 < \alpha < \frac{2}{\lambda_{max}(\mathbf{Q})}$, i.e. converges globally.
3. The method of steepest descent converges linearly with a ratio no greater than $1 - \frac{1}{\kappa}$.
4. The steepest descent algorithm with exact line search with $\boldsymbol{g}_k \neq \mathbf{0}$ converges in one iteration iff $\boldsymbol{g}_k$ is an eigenvector of $\mathbf{Q}$

# Non-quadratic case

## Theorem 1.1

*Non-quadratic case Suppose $f$ is defined on $\mathbb{R}^n$, has continuous second partial derivatives, and has a relative minimum at $\boldsymbol{x}^*$. Suppose further that the Hessian matrix of $f$, $\mathbf{H}(x^*)$, has smallest eigenvalue $a > 0$ and largest eigenvalue $A > 0$. If $\{\boldsymbol{x}_k\}$ is a sequence generated by the method of steepest descent that converges to $\boldsymbol{x}^*$, then the sequence of objective values $\{f(\boldsymbol{x}_k)\}$ converges to $f(\boldsymbol{x}^*)$ linearly with a convergence ratio no greater than $\left(\frac{A-a}{A+a}\right)^2$, i.e., for all k sufficiently large, we have*

$$f(\boldsymbol{x}_{k+1}) - f(\boldsymbol{x}^*) \leq \left(\frac{A-a}{A+a}\right)^2 [f(\boldsymbol{x}_k) - f(\boldsymbol{x}^*)]$$

# Newton's Method for Systems of NonLinear Equations

## Problem

An unconstraint optimization problems, in general, yields the following System of NonLinear Equations

$$\boldsymbol{g}(\boldsymbol{x}) = 0$$

where, in our case, $\boldsymbol{g} \stackrel{def}{=} \nabla f : \mathbb{R}^n \to \mathbb{R}^n$ is assumed to be continuously differentiable.

# Newton's Method for Systems of NonLinear Equations

**Algorithm**

Given $\boldsymbol{g} : \mathbb{R}^n \to \mathbb{R}^n$ continuously differentiable and $\boldsymbol{x}_0 \in \mathbb{R}^n$: at each iteration $k$, (Note: in our case $\boldsymbol{g} = \nabla f$ and $D\boldsymbol{g} = \nabla^2 f = \mathbf{H}$)

1. Solve $D\boldsymbol{g}(\boldsymbol{x}_k)\boldsymbol{d}_k = -\boldsymbol{g}(\boldsymbol{x}_k)$
2. Update $\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \boldsymbol{d}_k$

The derivative of $\boldsymbol{g} = \nabla f(\cdot)$ at $\boldsymbol{x}$ is the Jacobian (matrix) of $\boldsymbol{g} = \nabla f(\cdot)$ at $\boldsymbol{x}$, denoted here as $\mathbf{J}(\boldsymbol{x}_k) = D\boldsymbol{g}(\boldsymbol{x}_k)$, or the Hessian of $f$, ie, $\mathbf{H}(\boldsymbol{x}_k) = D\boldsymbol{g}(\boldsymbol{x}_k) = \nabla^2 f(\boldsymbol{x}_k)$.

### Advantages of Newton's method

1. Quadratically convergence from good starting guesses if $\mathbf{H}(\boldsymbol{x}^*) = \nabla^2 f(\boldsymbol{x}^*)$ is nonsingular.
2. Exact solution in one iteration for an affine $\nabla f$ (exact at each iteration for any affine component functions of $\nabla f$), i.e., for quadratic problem.

## Disadvantages of Newton's method

1. In general, it is not globally convergent.
2. Requires the computation of $\mathbf{H}(\boldsymbol{x}_k) = \nabla^2 f(\boldsymbol{x}_k)$ at each iteration.
3. It requires, at each iteration, the solution of a system of linear equations that may be singular or ill-conditioned.
4. $\boldsymbol{d}_k = -\mathbf{H}(\boldsymbol{x}_k)^{-1}\boldsymbol{g}(\boldsymbol{x}_k)$ could not be a descent direction.

# Local convergence of Newton's Method

**Theorem 2.1**

*Suppose that $f \in \mathcal{C}^3$ ,and $\boldsymbol{x}^* \in \mathbb{R}^n$ is a point such that $\nabla f(\boldsymbol{x}^*) = 0$ and $\mathbf{H}(\boldsymbol{x}^*)$ is invertible. Then, for all $\boldsymbol{x}_0$ sufficiently close to $\boldsymbol{x}^*$, Newton's method is well defined for all $k$, and converges to $\boldsymbol{x}^*$ with order of convergence at least $2$.*

### Theorem 2.2

*Let* $\{\boldsymbol{x}_k\}$ *a sequence generated by Newton's method for minimizing a function* $f(\boldsymbol{x})$*. If the Hessian* $\mathbf{H}(\boldsymbol{x}_k) \succ 0$ *and* $\boldsymbol{g}_k = \nabla f(\boldsymbol{x}_k) \neq \mathbf{0}$ *then, the direction*

$$\boldsymbol{d}_k \;\; = \;\; -\mathbf{H}(\boldsymbol{x}_k)^{-1}\boldsymbol{g}(\boldsymbol{x}_k) = \boldsymbol{x}_{k+1} - \boldsymbol{x}_k$$

*is a descent direction*

## Comments

1. Newton's method has superior convergence properties if the starting point is near the solution.
2. However, the method is not guaranteed to converge to the solution if we start far away from it (in fact, it may not even be well defined because the Hessian may be singular).
3. The method may not be a descent method; that is, it is possible that $f(\boldsymbol{x}_{k+1}) > f(\boldsymbol{x}_k)$.
4. Is it possible to modify the algorithm such that the descent property holds??.

# Hessian modification

- If the Hessian matrix $\nabla^2 f(\boldsymbol{x})$ is not positive definite, the Newton direction $\boldsymbol{d}_k^N$

$$\nabla^2 f(\boldsymbol{x}_k)\boldsymbol{d}_k^N = -\nabla f(\boldsymbol{x}_k)$$

may not be a descent direction.

- One alternative to solve the previous problem is to modify the Hessian, ie,

$$\mathbf{B}_k = \nabla^2 f(\boldsymbol{x}_k) + \mathbf{E}_k$$

such that $\mathbf{B}_k \succ 0$ and the new direction

$$\mathbf{B}_k \boldsymbol{d}_k = -\nabla f(\boldsymbol{x}_k)$$

is a descent direction

# Line Search Newton with Modification

**Require:** $\boldsymbol{x}_0$

$k = 0$
**while** $\|\nabla f_k\| > \tau_g$ **do**
  Factorize the matrix $\mathbf{B}_k = \nabla^2 f(\boldsymbol{x}_k) + \mathbf{E}_k$ where $\mathbf{E}_k = 0$ if $\nabla^2 f(\boldsymbol{x}_k)$ is sufficiently positive definite; otherwise, $\mathbf{E}_k$ is chosen to ensure that $\mathbf{B}_k$ is sufficiently positive definite
  Solve $\mathbf{B}_k \boldsymbol{d}_k = -\nabla f(\boldsymbol{x}_k)$
  Compute $\alpha_k$ (Wolfe, Goldstein, or Armijo conditions)
  Set $\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \alpha_k \boldsymbol{d}_k$
  $k = k + 1$
**end while**

# Comments

- The global convergence can be stablished is the sequence $\{\mathbf{B}_k\}$ have bounded condition number ( bounded modified factorization property ) whenever the sequence of Hessians $\{\nabla^2 f(\boldsymbol{x}_k)\}$ is bounded; ie,

$$\kappa(\mathbf{B}_k) \quad = \quad \|\mathbf{B}_k\| \|\mathbf{B}_k^{-1}\| \leq C$$

for $C > 0$, with $\|\cdot\| = \|\cdot\|_2$

- The global convergence follows from Zoutendijk's Theorem, becuase if $\kappa(\mathbf{B}_k) \leq C$ then $\cos\theta_k \geq 1/C$ which guarantees that

$$\lim_{k\to\infty} \|\nabla f(\boldsymbol{x}_k)\| \quad \to \quad 0$$

## Comments

Using the following inequalities

$$\begin{aligned}
\frac{\|\mathbf{A}^{-1}(\mathbf{A}\boldsymbol{x})\|_2}{\|\mathbf{A}\boldsymbol{x}\|_2} &\leq \|\mathbf{A}^{-1}\|_2 \Longrightarrow \|\boldsymbol{x}\| \leq \|\mathbf{A}^{-1}\|_2\|\mathbf{A}\boldsymbol{x}\| \\
\frac{\boldsymbol{x}^T\mathbf{A}^{-1}\boldsymbol{x}}{\|\boldsymbol{x}\|^2} &\geq \frac{1}{\lambda_M(\mathbf{A})} = \frac{1}{\|\mathbf{A}\|_2}
\end{aligned}$$

If $\mathbf{A} \succ 0$ is symmetric then $\lambda_M(\mathbf{A}) = \|\mathbf{A}\|_2$. Using $\mathbf{B}_k\boldsymbol{d}_k = -\boldsymbol{g}_k$ and $\mathbf{A} = \mathbf{B}_k$ we obtain

$$\begin{aligned}
\cos\theta_k &= \frac{-\boldsymbol{d}_k^T\boldsymbol{g}_k}{\|\boldsymbol{d}_k\|\|\boldsymbol{g}_k\|} \geq \frac{-\boldsymbol{d}_k^T\boldsymbol{g}_k}{\|\mathbf{B}_k^{-1}\|_2\|\mathbf{B}_k\boldsymbol{d}_k\|\|\boldsymbol{g}_k\|} \\
&= \frac{\boldsymbol{g}_k^T\mathbf{B}_k^{-1}\boldsymbol{g}_k}{\|\boldsymbol{g}_k\|^2}\frac{1}{\|\mathbf{B}_k^{-1}\|_2} \geq \frac{1}{\|\mathbf{B}_k\|_2}\frac{1}{\|\mathbf{B}_k^{-1}\|_2} \geq \frac{1}{C}
\end{aligned}$$

# Theorem

### Theorem 3.1

*Let $f$ be twice continuously differentiable on an open set $D$, and assume that the starting point $x_0$ of the previous Algorithm is such that the level set $L = \{\boldsymbol{x} \in D : f(\boldsymbol{x}) \leq f(\boldsymbol{x}_0)\}$ is compact. Then if the bounded modified factorization property holds, we have that*

$$\lim_{k\to\infty} \|\nabla f(\boldsymbol{x}_k)\| \quad \to \quad 0$$

# Cholesky with Added Multiple of the Identity

- We can simply select $\mathbf{E_k} = \tau_k \mathbf{I}$ then

$$\mathbf{B}_k \;=\; \nabla^2 f(\boldsymbol{x}_k) + \tau_k \mathbf{I}$$

  with $\tau_k \geq 0$ such that it ensures that $\mathbf{B}_k$ is sufficiently positive definite.
- We can compute $\tau_k$ based on the smallest eigenvalue of the Hessian $\nabla^2 f(\boldsymbol{x}_k)$, but this is not always possible or it is computationally expensive.

# Cholesky with Added Multiple of the Identity

Choose $\beta > 0$, ie, $\beta = 1e-3$
**if** $\min_i(a_{ii}) > 0$ **then**
    $\tau_0 = 0$
**else**
    $\tau_0 = -\min_i(a_{ii}) + \beta$
**end if**
**for** $k = 0, 1, \ldots$ **do**
    Attempt to apply the Cholesky to obtain $\mathbf{L}\mathbf{L}^T = \mathbf{A} + \tau_k \mathbf{I}$
    **if** The decomposition is successful **then**
        Return $\mathbf{L}$
    **else**
        $\tau_{k+1} = \max(2\tau_k, \beta)$
    **end if**
**end for**

## Recall: Sufficient decrease and Backtracking

- Choose $\hat{\alpha} > 0$, $\rho \in (0, 1)$, $c_1 \in (0, 1)$, set $\alpha = \hat{\alpha}$
- **Repeat** until $f(\boldsymbol{x}_k + \alpha \boldsymbol{d}_k) \leq f(\boldsymbol{x}_k) + c_1 \alpha \nabla f_k^T \boldsymbol{d}_k$

  $\alpha = \rho\alpha$
- **end** (repeat)
- Terminate with $\alpha_k = \alpha$

# Quadratic Interpolation

Suppose that a given $\alpha_0 > 0$ does not satisfies the Armijo's or sufficient descent condition, i.e.,
$f(\boldsymbol{x}_k + \alpha_0 \boldsymbol{d}_k) > f(\boldsymbol{x}_k) + c_1 \alpha_0 \nabla f(\boldsymbol{x}_k)^T \boldsymbol{d}_k$ with $\boldsymbol{d}_k = -\nabla f(\boldsymbol{x}_k)$
and $c_1 = 10^{-4}$, or equivalently

$$
\begin{aligned}
\phi(\alpha_0) &> \phi(0) + c_1 \alpha_0 \phi'(0) = \ell(\alpha_0) \\
\phi(\alpha) &= f(\boldsymbol{x}_k + \alpha \boldsymbol{d}_k) \\
\ell(\alpha) &= \phi(0) + c_1 \alpha \phi'(0)
\end{aligned}
$$

# Quadratic Interpolation

The next candidate $\alpha_1$ is computed as the minimum for the quadratic interpolation

$$\phi_q(\alpha) = a\alpha^2 + b\alpha + c$$

satisfying

$$\begin{aligned}\phi_q(0) &= \phi(0)\\ \phi_q'(0) &= \phi'(0)\\ \phi_q(\alpha_0) &= \phi(\alpha_0)\end{aligned}$$

# Quadratic Interpolation

$$\phi_q'(\alpha) \quad = \quad 2a\alpha + b$$

Then

$$\begin{aligned} \phi_q(0) = c &\quad = \quad \phi(0) = f(\boldsymbol{x}_k) \\ \phi_q'(0) = b &\quad = \quad \phi'(0) = \nabla f(\boldsymbol{x}_k)^T \boldsymbol{d}_k \\ \phi_q(\alpha_0) = a\alpha_0^2 + b\alpha_0 + c &\quad = \quad \phi(\alpha_0) \end{aligned}$$

# Quadratic Interpolation

Then

$$
\begin{aligned}
c &= \phi(0) = f(\boldsymbol{x}_k) \\
b &= \phi'(0) = \nabla f(\boldsymbol{x}_k)^T \boldsymbol{d}_k \\
a &= \frac{\phi(\alpha_0) - b\alpha_0 - c}{\alpha_0^2}
\end{aligned}
$$

and

$$
\begin{aligned}
2a\alpha + b &= 0 \\
\alpha^* &= \frac{-b}{2a}
\end{aligned}
$$

$$
\boxed{\alpha^* = \frac{-\alpha_0^2 \phi'(0)}{2(\phi(\alpha_0) - \phi'(0)\alpha_0 - \phi(0))}}
$$

# Quadratic Interpolation

Note that $\alpha^* \in (0, \alpha_0)$ due to $a > 0$ and $b < 0$

$$a \quad = \quad \frac{\phi(\alpha_0) - \phi'(0)\alpha_0 - \phi(0)}{\alpha_0^2}$$

Let $\tau(\alpha) := \phi'(0)\alpha + \phi(0)$ the tangent to $\phi(\alpha)$ at $\alpha = 0$. as $c_1 = 10^{-4} < 1$ then $c_1\phi'(0)\alpha > \phi'(0)\alpha$ for $\alpha > 0$ due to $\phi'(0) < 0$. Therefore

$$\tau(\alpha) \quad < \quad \ell(\alpha), \text{ for } \alpha > 0$$

then

$$\ell(\alpha_0) \quad > \quad \tau(\alpha_0)$$

# Quadratic Interpolation

As $\alpha_0$ does not satisfies Armijo's Condition and the previous conclusion, we obtain:

$$\begin{aligned} \phi(\alpha_0) &> \phi(0) + c_1 \alpha \phi'(0) = \ell(\alpha_0) \\ \ell(\alpha_0) &> \tau(\alpha_0) \end{aligned}$$

then $\phi(\alpha_0) - \phi'(0)\alpha_0 - \phi(0) = \phi(\alpha_0) - \tau(\alpha_0) > 0$. And

$$a = \frac{\phi(\alpha_0) - \phi'(0)\alpha_0 - \phi(0)}{\alpha_0^2} > 0$$

i.e., $\phi_q(\alpha)$ is a convex parabola (an open upward parabola).

# Quadratic Interpolation

In order to proof that $\alpha^* < \alpha_0$ it is sufficient to proof that the slope of $\phi_q(\alpha)$ and $\alpha_0$ is positive, i.e., $\alpha_0$ is at the right side of the minimum $\alpha^*$ or it is located at the increasing region of $\phi_q(\alpha)$.

$$\begin{aligned}
\phi_q'(\alpha_0) &= 2a\alpha_0 + b \\
&= 2\frac{\phi(\alpha_0) - \phi'(0)\alpha_0 - \phi(0)}{\alpha_0^2}\alpha_0 + \phi(0) \\
&= 2\frac{\phi(\alpha_0) - \delta\phi'(0)\alpha_0 - \phi(0)}{\alpha_0}
\end{aligned}$$

with $\delta = 0.5$.

## Quadratic Interpolation

As $\delta = 0.5 > c_1$ then

$$\phi(0) + c_1 \alpha \phi'(0) > \delta \phi'(0) \alpha_0 + \phi(0)$$

and due to $\alpha_0$ does not satisfies Armijo's Condition, i.e.

$$\phi(\alpha_0) \quad > \quad \phi(0) + c_1 \alpha \phi'(0)$$

we obtain

$$\phi(\alpha_0) \quad > \quad \delta \phi'(0) \alpha_0 + \phi(0)$$

# Quadratic Interpolation

Therefore

$$\phi(\alpha_0) - \delta\phi'(0)\alpha_0 - \phi(0) \quad > \quad 0$$

and

$$\phi_q'(\alpha_0) > 0$$

due to $\alpha_0 > 0$. We conclude that $\alpha^* < \alpha_0$. And finally $\alpha^* \in (0, \alpha_0)$ and we can select $\alpha_1 = \alpha^*$

# Quadratic Interpolation: Algorithm

Given $\alpha_0 > 0$, $\phi(\cdot), \phi'(\cdot)$. ($\alpha_0$ does no satisfy Armijo's condition)

$\alpha_1 = \frac{-\alpha_0^2 \phi'(0)}{2(\phi(\alpha_0) - \phi'(0)\alpha_0 - \phi(0))}$

while $\phi(\alpha_1) > \phi(0) + c_1 \alpha_1 \phi'(0)$:

$\alpha_0 = \alpha_1$

$\alpha_1 = \frac{-\alpha_0^2 \phi'(0)}{2(\phi(\alpha_0) - \phi'(0)\alpha_0 - \phi(0))}$

return $\alpha_1$

Note: In practice we simply set $\alpha_0 = \frac{-\alpha_0^2 \phi'(0)}{2(\phi(\alpha_0) - \phi'(0)\alpha_0 - \phi(0))}$

# Cubic Interpolation

- Suppose that $\alpha_0 > 0$ does not satisfies the Armijo's or sufficient descent condition and, $\alpha_1$, obtained by the quadratic interpolation algorithm, does not satisfies the Armijo's condition.
- We can compute $\alpha_2$ by using cubic interpolation.

# Cubic Interpolation

The next candidate $\alpha_2$ can be computed as the minimum of the Cubic interpolation

$$\phi_c(\alpha) = a\alpha^3 + b\alpha^2 + c\alpha + d$$

satisfying

$$\begin{aligned} \phi_c(0) &= \phi(0) \\ \phi_c'(0) &= \phi'(0) \\ \phi_c(\alpha_0) &= \phi(\alpha_0) \\ \phi_c(\alpha_1) &= \phi(\alpha_1) \end{aligned}$$

# Cubic Interpolation

From

$$\begin{aligned}\phi_c(\alpha) &= a\alpha^3 + b\alpha^2 + c\alpha + d \\ \phi_c'(\alpha) &= 3a\alpha^2 + b\alpha + c\end{aligned}$$

and the conditions

$$\begin{aligned}\phi_c(0) &= \phi(0) \\ \phi_c'(0) &= \phi'(0)\end{aligned}$$

we obtain that

$$\begin{aligned}d &= \phi(0) \\ c &= \phi'(0) < 0\end{aligned}$$

# Cubic Interpolation

From the conditions

$$
\begin{aligned}
\phi_q(\alpha_0) &= \phi(\alpha_0) \\
\phi_q(\alpha_1) &= \phi(\alpha_1)
\end{aligned}
$$

we obtain that

$$
\begin{aligned}
\alpha_1^3 a + \alpha_1^2 b &= \phi_c(\alpha_1) - \phi'(0)\alpha_1 - \phi(0) \\
\alpha_0^3 a + \alpha_0^2 b &= \phi_c(\alpha_0) - \phi'(0)\alpha_0 - \phi(0)
\end{aligned}
$$

Note: The right side of both equations is positive due to $\alpha_0, \alpha_1$ do not satisfy Armijo (see proof in quadratic interpolation case)

# Cubic Interpolation

Solving

$$\begin{aligned}\alpha_1^3 a + \alpha_1^2 b &= \phi_c(\alpha_1) - \phi'(0)\alpha_1 - \phi(0)\\ \alpha_0^3 a + \alpha_0^2 b &= \phi_c(\alpha_0) - \phi'(0)\alpha_0 - \phi(0)\end{aligned}$$

for $a$, $b$

$$\left[\begin{array}{c} a \\ b \end{array}\right] = \frac{1}{\alpha_1^2\alpha_0^2(\alpha_1-\alpha_0)}\left[\begin{array}{cc} \alpha_0^2 & -\alpha_1^2 \\ -\alpha_0^3 & \alpha_1^3 \end{array}\right]\left[\begin{array}{c} \phi_c(\alpha_1) - \phi'(0)\alpha_1 - \phi(0) \\ \phi_c(\alpha_0) - \phi'(0)\alpha_0 - \phi(0) \end{array}\right]$$

Note: Since $0 < \alpha_1 < \alpha_0$ and $\phi_c(\alpha_i) - \phi'(0)\alpha_i - \phi(0) > 0$, $i = 0, 1$, it can be proof that $a$ and $b$ have different sign.

# Cubic Interpolation

We already know $a$, $b$, $c$ and $d$ then we can compute the minimum of $\phi_c(\cdot)$.

$$\phi_c'(\alpha) = 3a\alpha^2 + 2b\alpha + c = 0$$

and

$$\alpha^* = \frac{-b + \sqrt{b^2 - 3ac}}{3a}$$

is the minimum, i.e., it can be proof that $\phi_c''(\alpha^*) > 0$ and $\alpha^* > 0$, due to $a, b$ have different sign, and we set $\alpha_2 = \alpha^*$

# Cubic Interpolation: Algorithm

Given $\alpha_0, \alpha_1 > 0$, $\phi(\cdot), \phi'(\cdot)$. ($\alpha_0, \alpha_1$ do no satisfy Armijo's condition and $\alpha_1$ is computed with the quadratic interpolation algorithm)

$\alpha_2 = \frac{-b+\sqrt{b^2-3ac}}{3a}$

while $\phi(\alpha_2) > \phi(0) + c_1\alpha_2\phi'(0)$:

$\alpha_0 = \alpha_1$, $\alpha_1 = \alpha_2$

$\alpha_2 = \frac{-b+\sqrt{b^2-3ac}}{3a}$

return $\alpha_2$

# Initial step length

1. For Newton and quasi Newton methods, the step $\alpha_0 = 1$ should always be used as the initial trial step length.
2. Another option is to assume that the first-order change in the function at iterate $\boldsymbol{x}_k$ will be the same as that obtained at the previous step, i.e., select the initial guess $\alpha_0$ that satisfies

$$\alpha_0 \nabla f(\boldsymbol{x}_k)^T \boldsymbol{d}_k = \alpha_{k-1} \nabla f(\boldsymbol{x}_{k-1})^T \boldsymbol{d}_{k-1}$$

**Initial step length**: $\boxed{\alpha_0 = \dfrac{\alpha_{k-1} \nabla f(\boldsymbol{x}_{k-1})^T \boldsymbol{d}_{k-1}}{\nabla f(\boldsymbol{x}_k)^T \boldsymbol{d}_k}}$

# Initial step length

Another strategy is to compute the quadratic interpolation of $f(\boldsymbol{x}_{k-1})$, $\nabla f(\boldsymbol{x}_{k-1})^T \boldsymbol{d}_{k-1}$ and $f(\boldsymbol{x}_k)$ where $\alpha_{k-1}$ is the minimun, i.e, $\phi(0)$, $\phi'(0)$ and $\phi(\alpha_{k-1}) = f(\boldsymbol{x}_k)$. Then,

$$\begin{aligned}
\phi_q(\alpha) &= a\alpha^2 + \alpha b + \phi(0) \\
a &= \frac{\phi(\alpha_{k-1}) - \phi'(0)\alpha_{k-1} - \phi(0)}{\alpha_{k-1}^2} \\
b &= \phi'(0)
\end{aligned}$$

# Initial step length

If $\alpha_{k-1}$ is the minimum and $f(\boldsymbol{x}_k)$ its corresponding minimum value, then

$$
\begin{aligned}
2a\alpha_{k-1} + b = 0; \quad \alpha_{k-1} &= -\frac{b}{2a} \\
f(\boldsymbol{x}_k) &= \phi(\alpha_{k-1})
\end{aligned}
$$

$$
\begin{aligned}
f(\boldsymbol{x}_k) &= a\alpha_{k-1}^2 + \alpha_{k-1}b + f(\boldsymbol{x}_{k-1}) = \alpha_{k-1}(a\alpha_{k-1} + b) + f(\boldsymbol{x}_{k-1}) \\
&= \alpha_{k-1}(-ab/(2a) + b) + f(\boldsymbol{x}_k) = b\alpha_{k-1}/2 + f(\boldsymbol{x}_{k-1}) \\
\alpha_{k-1} &= 2\frac{f(\boldsymbol{x}_k) - f(\boldsymbol{x}_{k-1})}{\phi'(0)}
\end{aligned}
$$

**Initial step length**: $\boxed{\alpha_0 = 2\frac{f(\boldsymbol{x}_k) - f(\boldsymbol{x}_{k-1})}{\nabla f(\boldsymbol{x}_{k-1})^T \boldsymbol{d}_{k-1}}}$