Non-linear problems in n variable

Lectures for PHD course on
Unconstrained Numerical Optimization

Enrico Bertolazzi

DIMS – Università di Trento

May 2008

Outline

The problem to solve

Problem

Given $\mathbf{F} : D \subseteq \mathbb{R}^n \mapsto \mathbb{R}^n$
Find $\boldsymbol{x}_\star \in D$ *for which* $\mathbf{F}(\boldsymbol{x}_\star) = 0$.

Example

Let

$$\mathbf{F}(\boldsymbol{x}) = \begin{pmatrix} x_1^2 + x_2^3 + 7 \\ x_1 + x_2 + 1 \end{pmatrix}$$

which has $\mathbf{F}(\boldsymbol{x}_\star) = \mathbf{0}$ for $\boldsymbol{x}_\star = (1, -2)^T$.

Outline

The Newton procedure (1/3)

- Consider the following map

$$\mathbf{F}(\boldsymbol{x}) = \begin{pmatrix} x_1^2 + x_2^3 + 7 \\ x_1 + x_2 + 1 \end{pmatrix}$$

we known an approximation of a root $\boldsymbol{x}_0 \approx (1.1, -1.9)^T$.

- Setting $\boldsymbol{x}_1 = \boldsymbol{x}_0 + \boldsymbol{p}$ we obtain [1]

$$\mathbf{F}(\boldsymbol{x}_0 + \boldsymbol{p}) = \begin{pmatrix} 1.351 \\ 0.2 \end{pmatrix} + \begin{pmatrix} 2.2 & 10.83 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} p_1 \\ p_2 \end{pmatrix} + \vec{\mathcal{O}}(\|\boldsymbol{p}\|^2)$$

if $\boldsymbol{x}_0$ is a good approximation of a root of $\mathbf{F}(\boldsymbol{x})$ then $\vec{\mathcal{O}}(\|\boldsymbol{p}\|^2)$ is a small vector.

[1] Here $\vec{\mathcal{O}}(x)$ means $(\mathcal{O}(x), \ldots, \mathcal{O}(x))^T$

The Newton procedure (2/3)

- Neglecting $\vec{\mathcal{O}}(\|\boldsymbol{p}\|^2)$ and solving

$$\begin{pmatrix} 1.351 \\ 0.2 \end{pmatrix} + \begin{pmatrix} 2.2 & 10.83 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} p_1 \\ p_2 \end{pmatrix} = \mathbf{0}$$

we obtain $\boldsymbol{p} = (-0.094438, -0.105562)^T$.

- Now we set

$$\boldsymbol{x}_1 = \boldsymbol{x}_0 + \boldsymbol{p} = \begin{pmatrix} 1.005562 \\ -2.0055612 \end{pmatrix}$$

The Newton procedure (3/3)

- Considering

$$\mathbf{F}(\boldsymbol{x}_1 + \boldsymbol{q}) = \begin{pmatrix} -0.05576 \\ 8\,10^{-7} \end{pmatrix} + \begin{pmatrix} 2.0111 & 12.0668 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} q_1 \\ q_2 \end{pmatrix} + \vec{\mathcal{O}}(\|\boldsymbol{q}\|^2)$$

- Neglecting $\vec{\mathcal{O}}(\|\boldsymbol{q}\|^2)$ and solving

$$\begin{pmatrix} -0.05576 \\ 8\,10^{-7} \end{pmatrix} + \begin{pmatrix} 2.0111 & 12.0668 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} q_1 \\ q_2 \end{pmatrix} = \mathbf{0}$$

  we obtain $\boldsymbol{q} = (-0.0055466, 0.0055458)^T$.
- Now we set $\boldsymbol{x}_2 = \boldsymbol{x}_1 + \boldsymbol{q} = (1.000015, -2.000015)^T$

The Newton procedure: a modern point of view (1/2)

The previous procedure can be resumed as follows:

1. Consider the following function $\mathbf{F}(\boldsymbol{x})$. We known an approximation of a root $\boldsymbol{x}_0$.
2. Expand by Taylor series

$$\mathbf{F}(\boldsymbol{x}) = \mathbf{F}(\boldsymbol{x}_0) + \nabla\mathbf{F}(\boldsymbol{x}_0)(\boldsymbol{x} - \boldsymbol{x}_0) + \vec{\mathcal{O}}(\|\boldsymbol{x} - \boldsymbol{x}_0\|^2)$$

3. Drop the term $\vec{\mathcal{O}}(\|x - \boldsymbol{x}_0\|^2)$ and solve

$$\mathbf{0} = \mathbf{F}(\boldsymbol{x}_0) + \nabla\mathbf{F}(\boldsymbol{x}_0)(\boldsymbol{x} - \boldsymbol{x}_0)$$

   Call $\boldsymbol{x}_1$ this solution.
4. Repeat $1 - 3$ with $\boldsymbol{x}_1$, $\boldsymbol{x}_2$, $\boldsymbol{x}_3$, ...

The Newton procedure: a modern point of view (2/2)

Algorithm (Newton iterative scheme)

Let $\boldsymbol{x}_0$ *assigned, then for* $k = 0, 1, 2, \ldots$

1. *Solve for* $\boldsymbol{p}_k$*:*

$$\nabla\mathbf{F}(\boldsymbol{x}_k)\boldsymbol{p}_k + \mathbf{F}(\boldsymbol{x}_k) = \mathbf{0}$$

2. *Update*

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \boldsymbol{p}_k$$

Standard Assumptions

In the study of convergence of numerical scheme, some standard regularity assumption are assumed for the function $\mathbf{F}(\boldsymbol{x})$.

Assumption (Standard Assumptions)

The function $\mathbf{F} : D \subset \mathbb{R}^n \mapsto \mathbb{R}^n$ *is continuous, differentiable with Lipschitz derivative* $\nabla\mathbf{F}(\boldsymbol{x})$*. i.e.*

$$\|\nabla\mathbf{F}(\boldsymbol{x}) - \nabla\mathbf{F}(\boldsymbol{y})\| \leq \gamma \|\boldsymbol{x} - \boldsymbol{y}\| \qquad \forall \boldsymbol{x}, \boldsymbol{y} \in D \subset \mathbb{R}^n$$

Lemma (Taylor like expansion)

Let $\mathbf{F}(\boldsymbol{x})$ *satisfy the standard assumptions, then*

$$\|\mathbf{F}(\boldsymbol{y}) - \mathbf{F}(\boldsymbol{x}) - \nabla\mathbf{F}(\boldsymbol{x})(\boldsymbol{y} - \boldsymbol{x})\| \leq \frac{\gamma}{2} \|\boldsymbol{x} - \boldsymbol{y}\|^2 \quad \forall \boldsymbol{x}, \boldsymbol{y} \in D \subset \mathbb{R}^n$$

Proof.

From basic Calculus:

$$\mathbf{F}(\boldsymbol{y}) - \mathbf{F}(\boldsymbol{x}) = \int_0^1 \nabla\mathbf{F}(\boldsymbol{x} + t(\boldsymbol{y} - \boldsymbol{x}))(\boldsymbol{y} - \boldsymbol{x})\, dt$$

subtracting on both side $\nabla\mathbf{F}(\boldsymbol{x})(\boldsymbol{y} - \boldsymbol{x})$ we have

$$\mathbf{F}(\boldsymbol{y}) - \mathbf{F}(\boldsymbol{x}) - \nabla\mathbf{F}(\boldsymbol{x})(\boldsymbol{y} - \boldsymbol{x}) =$$

$$\int_0^1 \big[\nabla\mathbf{F}(\boldsymbol{x} + t(\boldsymbol{y} - \boldsymbol{x})) - \nabla\mathbf{F}(\boldsymbol{x})\big](\boldsymbol{y} - \boldsymbol{x})\, dt$$

and taking the norm

$$\|\mathbf{F}(\boldsymbol{y}) - \mathbf{F}(\boldsymbol{x}) - \nabla\mathbf{F}(\boldsymbol{x})(\boldsymbol{y} - \boldsymbol{x})\| \leq \int_0^1 \gamma t \,\|\boldsymbol{y} - \boldsymbol{x}\|^2\, dt$$

□

Lemma (Jacobian norm control)

Let $\mathbf{F}(\boldsymbol{x})$ satisfying standard assumptions, and $\nabla\mathbf{F}(\boldsymbol{x}_\star)$ non singular. Then there exists $\delta > 0$ such that for all $\|\boldsymbol{x} - \boldsymbol{x}_\star\| \leq \delta$ we have

$$2^{-1}\,\|\nabla\mathbf{F}(\boldsymbol{x})\| \leq \|\nabla\mathbf{F}(\boldsymbol{x}_\star)\| \leq 2\,\|\nabla\mathbf{F}(\boldsymbol{x})\|$$

and

$$2^{-1}\,\left\|\nabla\mathbf{F}(\boldsymbol{x})^{-1}\right\| \leq \left\|\nabla\mathbf{F}(\boldsymbol{x}_\star)^{-1}\right\| \leq 2\,\left\|\nabla\mathbf{F}(\boldsymbol{x})^{-1}\right\|$$

Proof. (1/3).

From standard assumptions choosing $\gamma\delta \leq 2^{-1}\|\nabla\mathbf{F}(\boldsymbol{x}_\star)\|$

$$\begin{aligned}\|\nabla\mathbf{F}(\boldsymbol{x})\| &\leq \|\nabla\mathbf{F}(\boldsymbol{x}) - \nabla\mathbf{F}(\boldsymbol{x}_\star)\| + \|\nabla\mathbf{F}(\boldsymbol{x}_\star)\| \\ &\leq \gamma\|\boldsymbol{x} - \boldsymbol{x}_\star\| + \|\nabla\mathbf{F}(\boldsymbol{x}_\star)\| \\ &\leq (3/2)\|\nabla\mathbf{F}(\boldsymbol{x}_\star)\| \leq 2\|\nabla\mathbf{F}(\boldsymbol{x}_\star)\|\end{aligned}$$

again choosing $\gamma\delta \leq 2^{-1}\|\nabla\mathbf{F}(\boldsymbol{x}_\star)\|$

$$\begin{aligned}\|\nabla\mathbf{F}(\boldsymbol{x}_\star)\| &\leq \|\nabla\mathbf{F}(\boldsymbol{x}_\star) - \nabla\mathbf{F}(\boldsymbol{x})\| + \|\nabla\mathbf{F}(\boldsymbol{x})\| \\ &\leq \gamma\|\boldsymbol{x} - \boldsymbol{x}_\star\| + \|\nabla\mathbf{F}(\boldsymbol{x})\| \\ &\leq 2^{-1}\|\nabla\mathbf{F}(\boldsymbol{x}_\star)\| + \|\nabla\mathbf{F}(\boldsymbol{x})\|\end{aligned}$$

so that $2^{-1}\|\nabla\mathbf{F}(\boldsymbol{x}_\star)\| \leq \|\nabla\mathbf{F}(\boldsymbol{x})\|$.

Proof. (2/3).

From the continuity of the determinant there exists a neighbor with $\nabla\mathbf{F}(\boldsymbol{x})$ non singular for all $\|\boldsymbol{x} - \boldsymbol{x}_\star\| \leq \delta$.

$$\begin{aligned}&\left\|\nabla\mathbf{F}(\boldsymbol{x})^{-1} - \nabla\mathbf{F}(\boldsymbol{x}_\star)^{-1}\right\| \\ &\qquad\leq \left\|\nabla\mathbf{F}(\boldsymbol{x})^{-1}\right\|\|\nabla\mathbf{F}(\boldsymbol{x}_\star) - \nabla\mathbf{F}(\boldsymbol{x})\|\left\|\nabla\mathbf{F}(\boldsymbol{x}_\star)^{-1}\right\| \\ &\qquad\leq \gamma\|\boldsymbol{x} - \boldsymbol{x}_\star\|\left\|\nabla\mathbf{F}(\boldsymbol{x})^{-1}\right\|\left\|\nabla\mathbf{F}(\boldsymbol{x}_\star)^{-1}\right\|\end{aligned}$$

and choosing δ such that $\gamma\delta\left\|\nabla\mathbf{F}(\boldsymbol{x}_\star)^{-1}\right\| \leq 2^{-1}$ we have

$$\left\|\nabla\mathbf{F}(\boldsymbol{x})^{-1} - \nabla\mathbf{F}(\boldsymbol{x}_\star)^{-1}\right\| \leq 2^{-1}\left\|\nabla\mathbf{F}(\boldsymbol{x})^{-1}\right\|$$

and using this last inequality

$$\begin{aligned}\left\|\nabla\mathbf{F}(\boldsymbol{x}_\star)^{-1}\right\| &\leq \left\|\nabla\mathbf{F}(\boldsymbol{x}_\star)^{-1} - \nabla\mathbf{F}(\boldsymbol{x})^{-1}\right\| + \left\|\nabla\mathbf{F}(\boldsymbol{x})^{-1}\right\| \\ &\leq (3/2)\left\|\nabla\mathbf{F}(\boldsymbol{x})^{-1}\right\| \leq 2\left\|\nabla\mathbf{F}(\boldsymbol{x})^{-1}\right\|\end{aligned}$$

Proof. (3/3).

Using last inequality again

$$\|\nabla\mathbf{F}(\boldsymbol{x})^{-1}\| \leq \|\nabla\mathbf{F}(\boldsymbol{x})^{-1} - \nabla\mathbf{F}(\boldsymbol{x}_\star)^{-1}\| + \|\nabla\mathbf{F}(\boldsymbol{x}_\star)^{-1}\|$$
$$\leq 2^{-1}\|\nabla\mathbf{F}(\boldsymbol{x})^{-1}\| + \|\nabla\mathbf{F}(\boldsymbol{x}_\star)^{-1}\|$$

so that

$$2^{-1}\|\nabla\mathbf{F}(\boldsymbol{x})^{-1}\| \leq \|\nabla\mathbf{F}(\boldsymbol{x}_\star)^{-1}\|$$

choosing δ such that for all $\|\boldsymbol{x} - \boldsymbol{x}_\star\| \leq \delta$ we have $\nabla\mathbf{F}(\boldsymbol{x})$ non singular and $\gamma\delta \leq 2^{-1}\|\nabla\mathbf{F}(\boldsymbol{x}_\star)\|$ and $\gamma\delta\|\nabla\mathbf{F}(\boldsymbol{x}_\star)^{-1}\| \leq 2^{-1}$ then the inequality of the lemma are true. □

Theorem (Local Convergence of Newton method)

Let $\mathbf{F}(\boldsymbol{x})$ satisfying standard assumptions, and $\boldsymbol{x}_\star$ a simple root (i.e. $\nabla\mathbf{F}(\boldsymbol{x}_\star)$ non singular). Then, if $\|\boldsymbol{x}_0 - \boldsymbol{x}_\star\| \leq \delta$ with $C\delta \leq 1$ where

$$C = \gamma\|\nabla\mathbf{F}(\boldsymbol{x}_\star)^{-1}\|$$

then, the sequence generated by Newton method satisfies:

1. $\|\boldsymbol{x}_k - \boldsymbol{x}_\star\| \leq \delta$ *for* $k = 0, 1, 2, 3, \ldots$
2. $\|\boldsymbol{x}_{k+1} - \boldsymbol{x}_\star\| \leq C\|\boldsymbol{x}_k - \boldsymbol{x}_\star\|^2$ *for* $k = 0, 1, 2, 3, \ldots$
3. $\lim_{k\mapsto\infty} \boldsymbol{x}_k = \boldsymbol{x}_\star$.

- The point 2 of the theorem is the second q-order of convergence of Newton method.

Proof.

Consider a Newton step with $\|\boldsymbol{x}_k - \boldsymbol{x}_\star\| \leq \delta$ and

$$
\begin{aligned}
\boldsymbol{x}_{k+1} - \boldsymbol{x}_\star &= \boldsymbol{x}_k - \boldsymbol{x}_\star - \nabla\mathbf{F}(\boldsymbol{x}_k)^{-1}\big[\mathbf{F}(\boldsymbol{x}_k) - \mathbf{F}(\boldsymbol{x}_\star)\big] \\
&= \nabla\mathbf{F}(\boldsymbol{x}_k)^{-1}\big[\nabla\mathbf{F}(\boldsymbol{x}_k)(\boldsymbol{x}_k - \boldsymbol{x}_\star) - \mathbf{F}(\boldsymbol{x}_k) + \mathbf{F}(\boldsymbol{x}_\star)\big]
\end{aligned}
$$

taking the norm and using Taylor like lemma

$$
\|\boldsymbol{x}_{k+1} - \boldsymbol{x}_\star\| \leq 2^{-1}\gamma \|\boldsymbol{x}_k - \boldsymbol{x}_\star\|^2 \left\|\nabla\mathbf{F}(\boldsymbol{x}_k)^{-1}\right\|
$$

from Jacobian norm control lemma (slide 12) there exist a δ such that $2\left\|\nabla\mathbf{F}(\boldsymbol{x}_k)^{-1}\right\| \geq \left\|\nabla\mathbf{F}(\boldsymbol{x}_\star)^{-1}\right\|$ for all $\|\boldsymbol{x}_k - \boldsymbol{x}_\star\| \leq \delta$.
Reducing eventually δ such that $\gamma\delta\left\|\nabla\mathbf{F}(\boldsymbol{x}_\star)^{-1}\right\| \leq 1$ we have

$$
\|\boldsymbol{x}_{k+1} - \boldsymbol{x}_\star\| \leq \gamma\left\|\nabla\mathbf{F}(\boldsymbol{x}_\star)^{-1}\right\| \delta \|\boldsymbol{x}_k - \boldsymbol{x}_\star\|^2 \leq \|\boldsymbol{x}_k - \boldsymbol{x}_\star\|,
$$

So that by induction we prove point 1. Point 2 and 3 follows trivially. □

Theorem (Newton-Kantorovich)

Let $\mathbf{F} : D \subset \mathbb{R}^n \mapsto \mathbb{R}^n$ be a differentiable mapping and let $\boldsymbol{x}_0 \in D$ be such that $\nabla\mathbf{F}(\boldsymbol{x}_0)$ is nonsingular. Let be

$$
\begin{aligned}
B(\boldsymbol{x}_0, \rho) &= \{\boldsymbol{y} \mid \|\boldsymbol{x}_0 - \boldsymbol{y}\| < \rho\}, \\
\alpha &= \left\|\nabla\mathbf{F}(\boldsymbol{x}_0)^{-1}\mathbf{F}(\boldsymbol{x}_0)\right\|,
\end{aligned}
$$

Moreover

- $\overline{B(\boldsymbol{x}_0, \rho)} \subset D$;
- $\left\|\nabla\mathbf{F}(\boldsymbol{x}_0)^{-1}(\mathbf{F}(\boldsymbol{x}) - \mathbf{F}(\boldsymbol{x}_0))\right\| \leq \omega \|\boldsymbol{x} - \boldsymbol{x}_0\|$ *for all* $\boldsymbol{x} \in D$;
- $\kappa := \alpha\omega \leq 2^{-1}$;

If the radius ρ is large enough, i.e.

$$
\hat{\rho} := \frac{1 - \sqrt{1 - 2\kappa}}{\omega} \leq \rho
$$

Then:

Theorem (cont.)

- $\mathbf{F}(\boldsymbol{x})$ *has a zero* $\boldsymbol{x}_\star \in \overline{B(\boldsymbol{x}_0, \hat{\rho})}$;
- *The open ball* $B(\boldsymbol{x}_0, \hat{\rho})$ *does not contain any zero of* $\mathbf{F}(\boldsymbol{x})$ *different from* $\boldsymbol{x}_\star$;
- *The Newton iterative procedure produce sequences belonging to* $B(\boldsymbol{x}_0, \hat{\rho})$ *that converge to* $\boldsymbol{x}_\star$;
- *If* $\kappa < 2^{-1}$ *then for Newton's method, we have*

$$\|\boldsymbol{x}_k - \boldsymbol{x}_\star\| \leq \frac{2\beta\lambda^{2^k}}{1-\lambda^{2^k}}$$

where

$$\beta = \frac{\sqrt{1-2\kappa}}{\omega}, \qquad \lambda = \frac{1-\kappa-\sqrt{1-2\kappa}}{\kappa}$$

Proof.

P. Deuflhard and G. Heindl
Affine Invariant Convergence Theorems for Newton's Method and Extensions to Related Methods
SIAM Journal on Numerical Analysis, **16**, 1979.

Florian A. Potra
The Kantorovich Theorem and interior point methods
Math. Program., Ser. A **102**, 2005.

J.M. Ortega
The Newton-Kantorovich theorem
Amer. Math. Monthly **75**, 1968.

□

- Newton method converge normally only when $\boldsymbol{x}_0$ is near $\boldsymbol{x}_\star$ a root of the nonlinear system.
- A way to make a more robust non linear solver is to use the techniques developed for minimization to make a globally convergent nonlinear solver.
- In particular if we consider the merit function

$$\mathsf{f}(\boldsymbol{x}) = \frac{1}{2}\|\mathbf{F}(\boldsymbol{x})\|^2$$

  we have that $\mathsf{f}(\boldsymbol{x}) \geq 0$ and if $\boldsymbol{x}_\star$ is such that $\mathsf{f}(\boldsymbol{x}_\star) = 0$ than we have that
  1. $\boldsymbol{x}_\star$ is a global minimum of $\mathsf{f}(\boldsymbol{x})$;
  2. $\mathbf{F}(\boldsymbol{x}_\star) = \mathbf{0}$, i.e. is a solution of the nonlinear system $\mathbf{F}(\boldsymbol{x})$.
- So that finding a global minimum of the merit function $\mathsf{f}(\boldsymbol{x})$ is the same of finding a solution of the nonlinear system $\mathbf{F}(\boldsymbol{x})$.

- We can apply for example the gradient method to the merit function $\mathsf{f}(\boldsymbol{x})$. This produce a slow method.
- Instead, we can use the Newton method to produce a search direction. The resulting method is the following
  1. Compute the search direction by solving $\nabla\mathbf{F}(\boldsymbol{x}_k)\boldsymbol{d}_k + \mathbf{F}(\boldsymbol{x}_k) = \mathbf{0}$;
  2. Find an approximate solution of the problem $\alpha_k = \arg\min_{\alpha\geq 0}\|\mathbf{F}(\boldsymbol{x}_k + \alpha\boldsymbol{d}_k)\|^2$;
  3. Update the solution $\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \alpha_k\boldsymbol{d}_k$.
- The previous algorithm work if the direction $\boldsymbol{d}_k$ is a descent direction.

Is $\boldsymbol{d}_k$ a descent direction? (1/2)

Lemma

The direction $\boldsymbol{d}$ computed as a solution of the problem

$$\nabla \mathbf{F}(\boldsymbol{x})\boldsymbol{d} + \mathbf{F}(\boldsymbol{x}) = \mathbf{0}$$

is a descent direction.

Proof.

Consider the gradient of $\mathsf{f}(\boldsymbol{x}) = (1/2)\,\|\mathbf{F}(\boldsymbol{x})\|^2$:

$$\frac{\partial \mathsf{f}(\boldsymbol{x})}{\partial x_k} = \frac{1}{2}\frac{\partial \|\mathbf{F}(\boldsymbol{x})\|^2}{\partial x_k} = \frac{1}{2}\frac{\partial}{\partial x_k}\sum_{i=1}^{n} F_i(\boldsymbol{x})^2 = \sum_{i=1}^{n}\frac{\partial F_i(\boldsymbol{x})}{\partial x_k}F_i(\boldsymbol{x})$$

this can be written as $\quad \nabla \mathsf{f}(\boldsymbol{x}) = \mathbf{F}(\boldsymbol{x})^T \nabla \mathbf{F}(\boldsymbol{x})$

(cont.)

Is $\boldsymbol{d}_k$ a descent direction? (2/2)

Proof.

Now we check $\nabla \mathsf{f}(\boldsymbol{x})\boldsymbol{d}$:

$$\begin{aligned}
\nabla \mathsf{f}(\boldsymbol{x})\boldsymbol{d} &= \mathbf{F}(\boldsymbol{x})^T \nabla \mathbf{F}(\boldsymbol{x})\boldsymbol{d} \\
&= -\mathbf{F}(\boldsymbol{x})^T \nabla \mathbf{F}(\boldsymbol{x}) \nabla \mathbf{F}(\boldsymbol{x})^{-1}\mathbf{F}(\boldsymbol{x}) \\
&= -\mathbf{F}(\boldsymbol{x})^T \mathbf{F}(\boldsymbol{x}) \\
&= -\|\mathbf{F}(\boldsymbol{x})\|^2 < 0
\end{aligned}$$

□

This lemma prove that Newton direction is a descent direction.

Is the angle between $\boldsymbol{d}_k$ and $\nabla \mathsf{f}(\boldsymbol{x}_k)$ bounded from $\pi/2$?

Let θ_k the angle between $\nabla \mathsf{f}(\boldsymbol{x}_k)$ and $\boldsymbol{d}_k$, then we have

$$\begin{aligned}\cos\theta_k &= -\frac{\nabla \mathsf{f}(\boldsymbol{x}_k)\boldsymbol{d}_k}{\|\mathbf{F}(\boldsymbol{x}_k)\|\,\|\nabla\mathbf{F}(\boldsymbol{x}_k)^{-1}\mathbf{F}(\boldsymbol{x}_k)\|}\\ &= \frac{\|\mathbf{F}(\boldsymbol{x}_k)\|}{\|\nabla\mathbf{F}(\boldsymbol{x}_k)^{-1}\mathbf{F}(\boldsymbol{x}_k)\|}\\ &\geq \frac{\|\mathbf{F}(\boldsymbol{x}_k)\|}{\|\nabla\mathbf{F}(\boldsymbol{x}_k)^{-1}\|\,\|\mathbf{F}(\boldsymbol{x}_k)\|}\\ &\geq \left\|\nabla\mathbf{F}(\boldsymbol{x}_k)^{-1}\right\|^{-1}\end{aligned}$$

so that, if for example $\left\|\nabla\mathbf{F}(\boldsymbol{x})^{-1}\right\|$ is bounded from below then the angle θ_k is strictly less then $\pi/2$ radiants. By the Zoutendijk theorem then the globalized Newton scheme is globally convergent.

Algorithm (The globalized Newton method)

$k \leftarrow 0$; $\boldsymbol{x}$ *assigned;*
$\boldsymbol{f} \leftarrow \mathbf{F}(\boldsymbol{x})$;
while $\|\boldsymbol{f}\| > \epsilon$ **do**
  — Evaluate search direction
  Solve $\nabla\mathbf{F}(\boldsymbol{x})\boldsymbol{d} + \mathbf{F}(\boldsymbol{x}) = \mathbf{0}$;
  — Evaluate dumping factor λ
  $\lambda \approx \arg\min_{\alpha>0} \|\mathbf{F}(\boldsymbol{x} + \alpha\boldsymbol{d}_k)\|^2$ *by line-search;*
  — perform step
  $\boldsymbol{x} \leftarrow \boldsymbol{x} + \lambda\boldsymbol{d}$;
  $\boldsymbol{f} \leftarrow \mathbf{F}(\boldsymbol{x})$;
  $k \leftarrow k + 1$;
end while

Outline

The Frobenius matrix norm

Definition

The Frobenius norm $\|\cdot\|_F$ of a matrix $\boldsymbol{A} \in \mathbb{R}^{n\times m}$ is defined as follows:

$$\|\boldsymbol{A}\|_F = \left(\sum_{i=1}^{n}\sum_{j=1}^{m} A_{ij}^2\right)^{1/2}$$

is a matrix norm, i.e. it satisfy:

1. $\|\boldsymbol{A}\|_F \geq 0$ and $\|\boldsymbol{A}\|_F = 0 \iff \boldsymbol{A} = \boldsymbol{0}$;
2. $\|\lambda \boldsymbol{A}\|_F = |\lambda| \, \|\boldsymbol{A}\|_F$;
3. $\|\boldsymbol{A} + \boldsymbol{B}\|_F \leq \|\boldsymbol{A}\|_F + \|\boldsymbol{B}\|_F$;
4. $\|\boldsymbol{A}\boldsymbol{B}\|_F \leq \|\boldsymbol{A}\|_F \, \|\boldsymbol{B}\|_F$;

The Frobenius norm is the length of the vector $\boldsymbol{A}$ if we consider $\boldsymbol{A}$ as a vector in $\mathbb{R}^{n^2}$.

The Frobenius matrix norm (2/4)

The first two point of the Frobenius norm $\|\cdot\|_F$ are trivial, to prove point 3 and 4 we need two classical inequality:

Cauchy–Schwartz inequality

$$\sum_{i=1}^{n} a_i b_i \leq \left(\sum_{i=1}^{n} a_i^2\right)^{1/2}\left(\sum_{i=1}^{n} b_i^2\right)^{1/2}$$

The inequality is strict unless $a_i = \lambda b_i$ for $i = 1, 2, \ldots, n$.

Triangular inequality

$$\left(\sum_{i=1}^{n} (a_i + b_i)^2\right)^{1/2} \leq \left(\sum_{i=1}^{n} a_i^2\right)^{1/2} + \left(\sum_{i=1}^{n} b_i^2\right)^{1/2}$$

The inequality is strict unless $a_i = \lambda b_i$ for $i = 1, 2, \ldots, n$.

The Frobenius matrix norm (3/4)

Proof of $\|\boldsymbol{A}+\boldsymbol{B}\|_F \leq \|\boldsymbol{A}\|_F + \|\boldsymbol{B}\|_F$.
By using triangular inequality

$$\begin{aligned}
\|\boldsymbol{A}+\boldsymbol{B}\|_F &= \left(\sum_{i,j=1}^{n} (A_{ij} + B_{ij})^2\right)^{1/2} \\
&\leq \left(\sum_{i,j=1}^{n} A_{ij}^2\right)^{1/2} + \left(\sum_{i,j=1}^{n} B_{ij}^2\right)^{1/2} \\
&= \|\boldsymbol{A}\|_F + \|\boldsymbol{B}\|_F .
\end{aligned}$$

The Frobenius matrix norm (4/4)

Proof of $\|\boldsymbol{AB}\|_F \leq \|\boldsymbol{A}\|_F \|\boldsymbol{B}\|_F$.
By using Cauchy–Schwartz inequality with

$$\begin{aligned}
\|\boldsymbol{AB}\|_F &= \Big(\sum_{i,j=1}^n \Big(\sum_{k=1}^n A_{ik}B_{kj}\Big)^2\Big)^{1/2} \\
&\leq \Big(\sum_{i,j=1}^n \Big(\sum_{k=1}^n A_{ik}^2\Big)\Big(\sum_{k'=1}^n B_{k'j}^2\Big)\Big)^{1/2} \\
&= \Big(\Big(\sum_{i=1}^n\sum_{k=1}^n A_{ik}^2\Big)\Big(\sum_{j=1}^n\sum_{k'=1}^n B_{k'j}^2\Big)\Big)^{1/2} \\
&= \|\boldsymbol{A}\|_F \|\boldsymbol{B}\|_F .
\end{aligned}$$

Lemma

Let $\boldsymbol{u}, \boldsymbol{w} \in \mathbb{R}^m$ column vector then the following equality is true:

$$\left\|\boldsymbol{u}\boldsymbol{w}^T\right\|_F \leq \|\boldsymbol{u}\|_2 \|\boldsymbol{w}\|_2$$

Proof.

$$\begin{aligned}
\left\|\boldsymbol{u}\boldsymbol{w}^T\right\|_F^2 &= \sum_{i,j=1}^n u_i^2 w_j^2 \\
&= \left(\sum_{i=1}^n u_i^2\right)\left(\sum_{j=1}^n w_j^2\right)
\end{aligned}$$

□

Lemma

Let $\boldsymbol{A} \in \mathbb{R}^{n\times m}$ *and* $\boldsymbol{x} \in \mathbb{R}^{m}$ *column vector then the following inequality is true:*

$$\|\boldsymbol{A}\boldsymbol{x}\|_2 \leq \|\boldsymbol{A}\|_F \|\boldsymbol{x}\|_2$$

Proof.

By using Cauchy-Schwarz inequality

$$\|\boldsymbol{A}\boldsymbol{x}\|_2^2 = \sum_{i=1}^{n}\Big(\sum_{j=1}^{m} A_{ij}x_j\Big)^2 \leq \sum_{i=1}^{n}\Big(\sum_{j=1}^{m} \boldsymbol{A}_{ij}^2\Big)\Big(\sum_{k} x_k^2\Big)$$

$$= \|\boldsymbol{A}\|_F^2 \|\boldsymbol{x}\|_2^2$$

□

The Frobenius matrix norm

Lemma

Let $\boldsymbol{a}, \boldsymbol{b} \in \mathbb{R}^{n}$ *and* $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^{m}$ *orthonormal vector. i.e.* $\boldsymbol{x}^T\boldsymbol{y} = 0$ *and* $\|\boldsymbol{x}\|_2 = \|\boldsymbol{y}\|_2 = 1$*, then the following equality is true*

$$\left\|\boldsymbol{a}\boldsymbol{x}^T + \boldsymbol{b}\boldsymbol{y}^T\right\|_F^2 = \|\boldsymbol{a}\|_2^2 + \|\boldsymbol{b}\|_2^2$$

Proof.

$$\left\|\boldsymbol{a}\boldsymbol{x}^T + \boldsymbol{b}\boldsymbol{y}^T\right\|_F^2 = \sum_{i=1}^{n}\sum_{j=1}^{m}(a_ix_j + b_iy_j)^2$$

$$= \sum_{i=1}^{n}\sum_{j=1}^{m}(a_i^2x_j^2 + b_i^2y_j^2 + 2a_ix_jb_iy_j)$$

$$= \|\boldsymbol{a}\|_2^2\|\boldsymbol{x}\|_2^2 + \|\boldsymbol{b}\|_2^2\|\boldsymbol{y}\|_2^2 + 2(\boldsymbol{a}^T\boldsymbol{b})\underbrace{(\boldsymbol{x}^T\boldsymbol{y})}_{=0}$$

□

Lemma

Let $\boldsymbol{A} \in \mathbb{R}^{n\times m}$ *and* $\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_n \in \mathbb{R}^m$ *a base of orthonormal vector for* $\mathbb{R}^m$*, then*

$$\|\boldsymbol{A}\|_F^2 = \sum_{k=1}^{n} \|\boldsymbol{A}\boldsymbol{v}_k\|_2^2$$

Proof.

consider a generic vector $\boldsymbol{u} = \alpha_1\boldsymbol{v}_1 + \cdots + \alpha_m\boldsymbol{v}_m$ and notice that

$$\left(\sum_{k=1}^{m} \boldsymbol{v}_k\boldsymbol{v}_k^T\right)\boldsymbol{u} = \left(\sum_{k=1}^{m} \boldsymbol{v}_k\boldsymbol{v}_k^T\right)\left(\sum_{j=1}^{m} \alpha_j\boldsymbol{v}_j\right) = \sum_{k=1}^{m}\sum_{j=1}^{m} \boldsymbol{v}_k\boldsymbol{v}_k^T\boldsymbol{v}_j\alpha_j$$

$$= \sum_{k=1}^{m} \alpha_k\boldsymbol{v}_k = \boldsymbol{u}$$

(cont.)

Proof.

Thus

$$\boldsymbol{I} = \sum_{k=1}^{m} \boldsymbol{v}_k\boldsymbol{v}_k^T$$

Using this relation we can write

$$\|\boldsymbol{A}\|_F^2 = \|\boldsymbol{A}\boldsymbol{I}\|_F^2 = \left\|\boldsymbol{A}\left(\sum_{k=1}^{m} \boldsymbol{v}_k\boldsymbol{v}_k^T\right)\right\|_F^2 = \left\|\sum_{k=1}^{m} \boldsymbol{w}_k\boldsymbol{v}_k^T\right\|_F^2 =$$

where $\boldsymbol{w}_k = \boldsymbol{A}\boldsymbol{v}_k$. Using the previous lemma we have

$$\|\boldsymbol{A}\|_F^2 = \sum_{k=1}^{m} \|\boldsymbol{w}_k\|_2^2 = \sum_{k=1}^{m} \|\boldsymbol{A}\boldsymbol{v}_k\|_2^2$$

□

Outline

The Broyden method (1/5)

- Newton method is a fast (q-order 2) numerical scheme to approximate the root of a function $\mathbf{F}(\boldsymbol{x})$ but needs the knowledge of the Jacobian $\nabla\mathbf{F}(\boldsymbol{x})$.
- Sometimes Jacobian is not available or too expensive to compute, in this case a numerical procedure to approximate the root which does not use derivative is mandatory.
- The Newton scheme find successively the root of the affine approximation

$$L_k(\boldsymbol{x}) \doteq \nabla\mathbf{F}(\boldsymbol{x}_k)(\boldsymbol{x}-\boldsymbol{x}_k)+\mathbf{F}(\boldsymbol{x}_k)=\mathbf{0}$$

- Substituting the Jacobian in the affine approximation by $\boldsymbol{A}_k$

$$M_k(\boldsymbol{x}) \doteq \boldsymbol{A}_k(\boldsymbol{x}-\boldsymbol{x}_k)+\mathbf{F}(\boldsymbol{x}_k)=\mathbf{0}$$

and solving successively this affine model produces the family of different methods:

The Broyden method (2/5)

Algorithm (Generic Secant iterative scheme)

Let $\boldsymbol{x}_0$ *and* $\boldsymbol{A}_0$ *assigned, then for* $k = 0, 1, 2, \ldots$

1. *Solve for* $\boldsymbol{p}_k$*:*
$$M_k(\boldsymbol{p}_k + \boldsymbol{x}_k) = \boldsymbol{A}_k\boldsymbol{p}_k + \mathbf{F}(\boldsymbol{x}_k) = \mathbf{0}$$
2. *Update the root approximation*
$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \boldsymbol{p}_k$$
3. *Update the affine model and produce* $\boldsymbol{A}_{k+1}$.

The Broyden method (3/5)

1. The update of $M_k \to M_{k+1}$ determine the algorithm.
2. A simple update is the forcing of a number of the secant relation:
$$M_{k+1}(\boldsymbol{x}_{k+1-\ell}) = \mathbf{F}(\boldsymbol{x}_{k+1-\ell}), \qquad \ell = 1, 2, \ldots, m$$
notice that $M_{k+1}(\boldsymbol{x}_{k+1}) = \mathbf{F}(\boldsymbol{x}_{k+1})$ for all $\boldsymbol{A}_{k+1}$.
3. If $\boldsymbol{A}_{k+1} \in \mathbb{R}^{n\times n}$ and $m = n$ and $\boldsymbol{d}_\ell = \boldsymbol{x}_{k+1-\ell} - \boldsymbol{x}_{k+1}$ are linearly independent then we have enough linear relation to determine $\boldsymbol{A}_{k+1}$.
4. Unfortunately vectors $\boldsymbol{d}_\ell$ tends to become linearly dependent so that this approach is very ill conditioned.
5. A more feasible approach uses less secant relation and other conditions to determine M_{k+1}.

The Broyden method (4/5)

1. The update of $M_k \to M_{k+1}$ in Broyden scheme is the following:
   1. $M_{k+1}(\boldsymbol{x}_k) = \mathbf{F}(\boldsymbol{x}_k)$;
   2. $M_{k+1}(\boldsymbol{x}) - M_k(\boldsymbol{x})$ is small in some sense;
2. The first condition imply

$$\boldsymbol{A}_{k+1}(\boldsymbol{x}_k - \boldsymbol{x}_{k+1}) + \mathbf{F}(\boldsymbol{x}_{k+1}) = \mathbf{F}(\boldsymbol{x}_k)$$

   which set n linear equation that do not determine the n^2 coefficients of $\boldsymbol{A}_{k+1}$.
3. The second condition become

$$M_{k+1}(\boldsymbol{x}) - M_k(\boldsymbol{x}) = (\boldsymbol{A}_{k+1} - \boldsymbol{A}_k)(\boldsymbol{x} - \boldsymbol{x}_k)$$

$$\|M_{k+1}(\boldsymbol{x}) - M_k(\boldsymbol{x})\| \leq \|\boldsymbol{A}_{k+1} - \boldsymbol{A}_k\| \, \|\boldsymbol{x} - \boldsymbol{x}_k\|$$

   where $\|\cdot\|$ is some norm. The term $\|\boldsymbol{x} - \boldsymbol{x}_k\|$ is not controllable, so a condition should be $\|\boldsymbol{A}_{k+1} - \boldsymbol{A}_k\|$ is minimum.

The Broyden method (5/5)

1. Defining

$$\boldsymbol{y}_k = \mathbf{F}(\boldsymbol{x}_{k+1}) - \mathbf{F}(\boldsymbol{x}_k), \qquad \boldsymbol{s}_k = \boldsymbol{x}_{k+1} - \boldsymbol{x}_k$$

   the Broyden scheme find the update $\boldsymbol{A}_{k+1}$ which satisfy:
   1. $\boldsymbol{A}_{k+1}\boldsymbol{s}_k = \boldsymbol{y}_k$;
   2. $\|\boldsymbol{A}_{k+1} - \boldsymbol{A}_k\| \leq \|\boldsymbol{B} - \boldsymbol{A}_k\|$ for all $\boldsymbol{B}$ such that $\boldsymbol{B}\boldsymbol{s}_k = \boldsymbol{y}_k$.
2. If we choose for the norm $\|\cdot\|$ the Frobenius norm $\|\cdot\|_F$

$$\|\boldsymbol{A}\|_F = \left(\sum_{i,j=1}^{n} A_{ij}^2\right)^{1/2}$$

   then the problem admits a unique solution.

With the Frobenius matrix norm it is possible to solve the following problem

Lemma

Let $\boldsymbol{A} \in \mathbb{R}^{n\times n}$ *and* $\boldsymbol{s}, \boldsymbol{y} \in \mathbb{R}^n$ *with* $\boldsymbol{s} \neq \boldsymbol{0}$ *and* $\boldsymbol{A}\boldsymbol{s} \neq \boldsymbol{y}$. *Consider the set*

$$\mathcal{B} = \left\{ \boldsymbol{B} \in \mathbb{R}^{n\times n} \,|\, \boldsymbol{B}\boldsymbol{s} = \boldsymbol{y} \right\}$$

then there exists a unique matrix $\boldsymbol{B} \in \mathcal{B}$ *such that*

$$\|\boldsymbol{A} - \boldsymbol{B}\|_F \leq \|\boldsymbol{A} - \boldsymbol{C}\|_F \qquad \text{for all } \boldsymbol{C} \in \mathcal{B}$$

moreover $\boldsymbol{B}$ *has the following form*

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

i.e. $\boldsymbol{B}$ *is a rank one perturbation of the matrix* $\boldsymbol{A}$.

Proof. (1/4).

First of all notice that

$$\frac{1}{\boldsymbol{s}^T\boldsymbol{s}}\boldsymbol{y}\boldsymbol{s}^T \in \mathcal{B} \qquad \left[\frac{1}{\boldsymbol{s}^T\boldsymbol{s}}\boldsymbol{y}\boldsymbol{s}^T\right]\boldsymbol{s} = \boldsymbol{y}$$

so that set $\mathcal{B}$ is not empty. Next we reformulate the problem as a constrained minimum problem:

$$\arg\min_{\boldsymbol{B}\in\mathbb{R}^{n\times n}} \frac{1}{2}\sum_{i,j=1}^{n}(A_{ij} - B_{ij})^2 \qquad \text{subject to } \boldsymbol{B}\boldsymbol{s} = \boldsymbol{y}.$$

The solution is a stationary point of the Lagrangian:

$$g(\boldsymbol{B}, \boldsymbol{\lambda}) = \frac{1}{2}\sum_{i,j=1}^{n}(A_{ij} - B_{ij})^2 + \sum_{i=1}^{n}\lambda_i\left(\sum_{j=1}^{n}B_{ij}s_j - y_i\right)$$

Proof. (2/4).

taking the gradient we have

$$\frac{\partial}{\partial B_{ij}} g(\boldsymbol{B},\boldsymbol{\lambda}) = A_{ij} - B_{ij} + \lambda_i s_j = 0$$

$$\frac{\partial}{\partial \lambda_i} g(\boldsymbol{B},\boldsymbol{\lambda}) = \sum_{j=1}^{n} B_{ij} s_j - y_j = 0$$

The previous equality can be written in matrix form

$$\boldsymbol{B} = \boldsymbol{A} + \boldsymbol{\lambda}\boldsymbol{s}^T \qquad \boldsymbol{B}\boldsymbol{s} = \boldsymbol{y}$$

so that we can solve for $\boldsymbol{\lambda}$

$$\boldsymbol{B}\boldsymbol{s} = \boldsymbol{A}\boldsymbol{s} + \boldsymbol{\lambda}\boldsymbol{s}^T\boldsymbol{s} = \boldsymbol{y} \qquad \boldsymbol{\lambda} = \frac{\boldsymbol{y} - \boldsymbol{A}\boldsymbol{s}}{\boldsymbol{s}^T\boldsymbol{s}}$$

next we prove that $\boldsymbol{B}$ is the unique minimum.

Proof. (3/4).

The matrix $\boldsymbol{B}$ is at minimum distance, in fact

$$\|\boldsymbol{B} - \boldsymbol{A}\|_F = \left\|\boldsymbol{A} + \frac{(\boldsymbol{y} - \boldsymbol{A}\boldsymbol{s})\boldsymbol{s}^T}{\boldsymbol{s}^T\boldsymbol{s}} - \boldsymbol{A}\right\|_F = \left\|\frac{(\boldsymbol{y} - \boldsymbol{A}\boldsymbol{s})\boldsymbol{s}^T}{\boldsymbol{s}^T\boldsymbol{s}}\right\|_F$$

for all $\boldsymbol{C} \in \mathcal{B}$ we have $\boldsymbol{C}\boldsymbol{s} = \boldsymbol{y}$ so that

$$\|\boldsymbol{B} - \boldsymbol{A}\|_F = \left\|\frac{(\boldsymbol{C}\boldsymbol{s} - \boldsymbol{A}\boldsymbol{s})\boldsymbol{s}^T}{\boldsymbol{s}^T\boldsymbol{s}}\right\|_F = \left\|(\boldsymbol{C} - \boldsymbol{A})\frac{\boldsymbol{s}\boldsymbol{s}^T}{\boldsymbol{s}^T\boldsymbol{s}}\right\|_F$$

$$\leq \|\boldsymbol{C} - \boldsymbol{A}\|_F \left\|\frac{\boldsymbol{s}\boldsymbol{s}^T}{\boldsymbol{s}^T\boldsymbol{s}}\right\|_F = \|\boldsymbol{C} - \boldsymbol{A}\|_F$$

because in general

$$\left\|\boldsymbol{u}\boldsymbol{v}^T\right\|_F = \left(\sum_{i,j=1}^{n} u_i^2 v_j^2\right)^{\frac{1}{2}} = \left(\sum_{i=1}^{n} u_i^2 \sum_{j=1}^{n} v_j^2\right)^{\frac{1}{2}} = \|\boldsymbol{u}\|\,\|\boldsymbol{v}\|$$

Proof. (4/4).

Let $\boldsymbol{B}'$ and $\boldsymbol{B}''$ two different minimum. Then $\frac{1}{2}(\boldsymbol{B}' + \boldsymbol{B}'') \in \mathcal{B}$ moreover

$$\left\| \boldsymbol{A} - \frac{1}{2}(\boldsymbol{B}' + \boldsymbol{B}'') \right\|_F \leq \frac{1}{2} \left\| \boldsymbol{A} - \boldsymbol{B}' \right\|_F + \frac{1}{2} \left\| \boldsymbol{A} - \boldsymbol{B}'' \right\|_F$$

If the inequality is strict we have a contradiction. From the Cauchy–Schwartz inequality we have an equality only when $\boldsymbol{A} - \boldsymbol{B}' = \lambda(\boldsymbol{A} - \boldsymbol{B}'')$ so that

$$\boldsymbol{B}' - \lambda \boldsymbol{B}'' = (1 - \lambda)\boldsymbol{A}$$

and

$$\boldsymbol{B}'\boldsymbol{s} - \lambda \boldsymbol{B}''\boldsymbol{s} = (1 - \lambda)\boldsymbol{A}\boldsymbol{s} \quad \Rightarrow \quad (1 - \lambda)\boldsymbol{y} = (1 - \lambda)\boldsymbol{A}\boldsymbol{s}$$

due to $\boldsymbol{A}\boldsymbol{s} \neq \boldsymbol{y}$ this is true only when $\lambda = 1$, i.e. $\boldsymbol{B}' = \boldsymbol{B}''$. □

Corollary

The update

$$\boldsymbol{A}_{k+1} = \boldsymbol{A}_k + \frac{(\boldsymbol{y}_k - \boldsymbol{A}_k \boldsymbol{s}_k)\boldsymbol{s}_k^T}{\boldsymbol{s}_k^T \boldsymbol{s}_k}$$

satisfy the secant condition:

$$\boldsymbol{A}_{k+1}\boldsymbol{s}_k = \boldsymbol{y}_k$$

moreover, $\boldsymbol{A}_{k+1}$ is the nearest matrix in the Frobenius norm that satisfy the secant condition.

Remark

Different the norm produce different results and in general you can loose uniqueness of the update.

The Broyden method (1/2)

Algorithm (The Broyden method)

$k \leftarrow 0$; $\boldsymbol{x}_0$ *and* $\boldsymbol{A}_0$ *assigned (for example* $\boldsymbol{A}_0 = \nabla\mathbf{F}(\boldsymbol{x}_0)$*);*
$\boldsymbol{f}_0 \leftarrow \mathbf{F}(\boldsymbol{x}_0)$;
while $\|\boldsymbol{f}_k\| > \epsilon$ **do**
  Solve for $\boldsymbol{s}_k$ *the linear system* $\boldsymbol{A}_k\boldsymbol{s}_k + \boldsymbol{f}_k = \mathbf{0}$;
  $\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \boldsymbol{s}_k$;
  $\boldsymbol{f}_{k+1} = \mathbf{F}(\boldsymbol{x}_{k+1})$;
  $\boldsymbol{y}_k = \boldsymbol{f}_{k+1} - \boldsymbol{f}_k$;
  Update: $\boldsymbol{A}_{k+1} = \boldsymbol{A}_k + \dfrac{(\boldsymbol{y}_k - \boldsymbol{A}_k\boldsymbol{s}_k)\boldsymbol{s}_k^T}{\boldsymbol{s}_k^T\boldsymbol{s}_k}$;
  $k \leftarrow k+1$;
end while

The Broyden method (2/2)

Notice that $\boldsymbol{y}_k - \boldsymbol{A}_k\boldsymbol{s}_k = \boldsymbol{f}_{k+1} - \boldsymbol{f}_k + \boldsymbol{f}_k$ so that the update can be written as $\boldsymbol{A}_{k+1} \leftarrow \boldsymbol{A}_k + \boldsymbol{f}_{k+1}\boldsymbol{s}_k^T/\boldsymbol{s}_k^T\boldsymbol{s}_k$ and $\boldsymbol{y}_k$ can be eliminated.

Algorithm (The Broyden method (alternative version))

$k \leftarrow 0$; $\boldsymbol{x}$ *and* $\boldsymbol{A}$ *assigned (for example* $\boldsymbol{A} = \nabla\mathbf{F}(\boldsymbol{x})$*);*
$\boldsymbol{f} \leftarrow \mathbf{F}(\boldsymbol{x})$;
while $\|\boldsymbol{f}\| > \epsilon$ **do**
  Solve for $\boldsymbol{s}$ *the linear system* $\boldsymbol{A}\boldsymbol{s} + \boldsymbol{f} = \mathbf{0}$;
  $\boldsymbol{x} \leftarrow \boldsymbol{x} + \boldsymbol{s}$;
  $\boldsymbol{f} \leftarrow \mathbf{F}(\boldsymbol{x})$;
  Update: $\boldsymbol{A} \leftarrow \boldsymbol{A} + \dfrac{\boldsymbol{f}\boldsymbol{s}^T}{\boldsymbol{s}^T\boldsymbol{s}}$;
  $k \leftarrow k+1$;
end while

Broyden algorithm properties (1/2)

Theorem

Let $\mathbf{F}(\boldsymbol{x})$ satisfy the standard regularity conditions with $\nabla\mathbf{F}(\boldsymbol{x}_\star)$ nonsingular. Then there exists positive constants ϵ, δ such that if $\|\boldsymbol{x}_0 - \boldsymbol{x}_\star\| \leq \epsilon$ and $\|\boldsymbol{A}_0 - \nabla\mathbf{F}(\boldsymbol{x}_\star)\| \leq \delta$, then the sequence $\{\boldsymbol{x}_k\}$ generated by the Broyden method is well defined and converge q-superlinearly to $\boldsymbol{x}_\star$, i.e.

$$\lim_{k\to\infty} \frac{\|\boldsymbol{x}_{k+1} - \boldsymbol{x}_k\|}{\|\boldsymbol{x}_k - \boldsymbol{x}_\star\|} = 0$$

C.G.Broyden, J.E.Dennis, J.J.Moré
On the local and super-linear convergence of quasi-Newton methods.
J. Inst. Math. Appl, **6** 222–236, 1973.

Broyden algorithm properties (2/2)

Theorem

Let $\mathbf{F}(\boldsymbol{x}) = \boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}$ where $\boldsymbol{A} \in \mathbb{R}^{n\times n}$. Then the Broyden method converge in at most $2n$ steps.

Theorem

Let $\mathbf{F} : \mathbb{R}^n \mapsto \mathbb{R}^n$ satisfy the standard regularity conditions with $\nabla\mathbf{F}(\boldsymbol{x}_\star)$ nonsingular. Then there exists positive constants ϵ, δ such that if $\|\boldsymbol{x}_0 - \boldsymbol{x}_\star\| \leq \epsilon$ and $\|\boldsymbol{A}_0 - \nabla\mathbf{F}(\boldsymbol{x}_\star)\| \leq \delta$, then the sequence $\{\boldsymbol{x}_k\}$ generated by the Broyden method satisfy

$$\|\boldsymbol{x}_{k+2n} - \boldsymbol{x}_\star\| \leq C \|\boldsymbol{x}_k - \boldsymbol{x}_\star\|^2$$

D.M. Gay
Some convergence properties of Broyden's method.
SIAM Journal of Numerical Analysis, **16** 623–630, 1979.

Reorganizing Broyden update

- Broyden method needs to solve a linear system for $\boldsymbol{A}_k$ at each step
- This can be onerous in terms of CPU cost
- it is possible to update directly the inverse of $\boldsymbol{A}_k$ i.e. it is possible to update $\boldsymbol{H}_k = \boldsymbol{A}_k^{-1}$.
- The update of $\boldsymbol{A}_k$ solve the problem of efficiency but do not alleviate the memory occupation
- The matrix $\boldsymbol{A}_k$ can be written as a product of simple matrix, this can save memory if the update are lesser respect to the system dimension.

Sherman-Morrison formula

Sherman-Morrison formula permit to explicity write the inverse of a matrix perturbed with a rank 1 matrix

Proposition (Sherman–Morrison formula)

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

where

$$\alpha = 1 + \boldsymbol{v}^T\boldsymbol{A}^{-1}\boldsymbol{u}$$

The Sherman–Morrison formula can be checked by a direct calculation.

Application of Sherman-Morrison formula (1/2)

- From the Broyden update formula

$$A_{k+1} = A_k + \frac{f_{k+1} s_k^T}{s_k^T s_k}$$

- By using Sherman–Morrison formula

$$A_{k+1}^{-1} = A_k^{-1} - \frac{1}{\beta_k} A_k^{-1} f_{k+1} s_k^T A_k^{-1}$$

$$\beta_k = s_k^T s_k + s_k^T A_k^{-1} f_{k+1}$$

- By setting $H_k = A_k^{-1}$ we have the update formula for H_k:

$$H_{k+1} = H_k - \frac{1}{\beta_k} H_k f_{k+1} s_k^T H_k$$

$$\beta_k = s_k^T s_k + s_k^T H_k f_{k+1}$$

Application of Sherman-Morrison formula (2/2)

- The update formula for H_k:

$$H_{k+1} = H_k - \frac{1}{\beta_k} H_k f_{k+1} s_k^T H_k$$

$$\beta_k = s_k^T s_k + s_k^T H_k f_{k+1}$$

- Can be reorganized as follows
  1. Compute $z_{k+1} = H_k f_{k+1}$;
  2. Compute $\beta_k = s_k^T s_k + s_k^T z_{k+1}$;
  3. Compute $H_{k+1} = \left(I - \beta_k^{-1} z_{k+1} s_k^T \right) H_k$;

The Broyden method with inverse updated

Algorithm (The Broyden method (updating inverse))

$k \leftarrow 0$; $\boldsymbol{x}_0$ *assigned*;
$\boldsymbol{f}_0 \leftarrow \mathbf{F}(\boldsymbol{x}_0)$;
$\boldsymbol{H}_0 \leftarrow \boldsymbol{I}$ *or better* $\boldsymbol{H}_0 \leftarrow \nabla\mathbf{F}(\boldsymbol{x}_0)^{-1}$;
while $\|\boldsymbol{f}_k\| > \epsilon$ **do**

— *perform step*

$$\begin{aligned} \boldsymbol{s}_k &= -\boldsymbol{H}_k\boldsymbol{f}_k; \\ \boldsymbol{x}_{k+1} &= \boldsymbol{x}_k + \boldsymbol{s}_k; \\ \boldsymbol{f}_{k+1} &= \mathbf{F}(\boldsymbol{x}_{k+1}); \end{aligned}$$

— *update* $\boldsymbol{H}$

$$\begin{aligned} \boldsymbol{z}_{k+1} &= \boldsymbol{H}_k\boldsymbol{f}_{k+1}; \\ \beta_k &= \boldsymbol{s}_k^T\boldsymbol{s}_k + \boldsymbol{s}_k^T\boldsymbol{z}_{k+1}; \\ \boldsymbol{H}_{k+1} &= \left(\boldsymbol{I} - \beta_k^{-1}\boldsymbol{z}_{k+1}\boldsymbol{s}_k^T\right)\boldsymbol{H}_k; \\ k &\leftarrow k+1; \end{aligned}$$

end while

- If n is very large then the storing of $\boldsymbol{H}_k$ can be very expensive.
- Moreover when n is very large we hope to find a good solution with a number m of iteration with $m \lll n$
- So that instead of storing $\boldsymbol{H}_k$ we can decide to store the vectors $\boldsymbol{z}_k$ and $\boldsymbol{s}_k$ plus the scalars β_k. With this vectors and scalars we can write

$$\boldsymbol{H}_k = \left(\boldsymbol{I} - \beta_{k-1}\boldsymbol{z}_k\boldsymbol{s}_{k-1}^T\right)\cdots\left(\boldsymbol{I} - \beta_1\boldsymbol{z}_2\boldsymbol{s}_1^T\right)\left(\boldsymbol{I} - \beta_0\boldsymbol{z}_1\boldsymbol{s}_0^T\right)\boldsymbol{H}_0$$

- Assuming $\boldsymbol{H}_0 = \boldsymbol{I}$ or can be computed on the fly we must store only $2\,n\,m + m$ real number instead of n^2 saving a lot of memory.
- However we can do better. It is possible to eliminate $\boldsymbol{z}_k$ ad store only $n\,m + m$ real numbers.

Elimination of z_k (1/3)

1. A step of the broyden iterative scheme can be rewritten as

$$
\begin{aligned}
d_k &= -H_k f_k \\
x_{k+1} &= x_k + d_k \\
f_{k+1} &= \mathbf{F}(x_{k+1}) \\
z_{k+1} &= H_k f_{k+1} \\
H_{k+1} &= \left(I - \frac{z_{k+1} d_k^T}{d_k^T d_k + d_k^T z_{k+1}} \right) H_k
\end{aligned}
$$

2. you can notice that z_k and d_k are similar and contains a lot of common information.
3. It is possible exploring the iteration to eliminate z_k from the update formula of H_k so that we can store the whole sequence without the vectors z_k.

Elimination of z_k (2/3)

$$
\begin{aligned}
-d_{k+1} = H_{k+1} f_{k+1} &= \left(I - \frac{z_{k+1} d_k^T}{d_k^T d_k + d_k^T z_{k+1}} \right) H_k f_{k+1} \\
&= \left(I - \frac{z_{k+1} d_k^T}{d_k^T d_k + d_k^T z_{k+1}} \right) z_{k+1} \\
&= z_{k+1} - \frac{z_{k+1} d_k^T z_{k+1}}{d_k^T d_k + d_k^T z_{k+1}} \\
&= \frac{d_k^T d_k}{d_k^T d_k + d_k^T z_{k+1}} z_{k+1}
\end{aligned}
$$

substituting in the update formula for H_{k+1} we obtain

$$
H_{k+1} \leftarrow \left(I + \frac{d_{k+1} d_k^T}{d_k^T d_k} \right) H_k
$$

Elimination of $\boldsymbol{z}_k$ (3/3)

Substituting into the step of the broyden iterative scheme and assuming $\boldsymbol{d}_k$ known

$$\begin{aligned}
\boldsymbol{x}_{k+1} &= \boldsymbol{x}_k + \boldsymbol{d}_k \\
\boldsymbol{f}_{k+1} &= \mathbf{F}(\boldsymbol{x}_{k+1}) \\
\boldsymbol{z}_{k+1} &= \boldsymbol{H}_k \boldsymbol{f}_{k+1} \\
\boldsymbol{d}_{k+1} &= -\frac{\boldsymbol{d}_k^T \boldsymbol{d}_k}{\boldsymbol{d}_k^T \boldsymbol{d}_k + \boldsymbol{d}_k^T \boldsymbol{z}_{k+1}} \boldsymbol{z}_{k+1} \\
\boldsymbol{H}_{k+1} &= \left(\boldsymbol{I} + \frac{\boldsymbol{d}_{k+1} \boldsymbol{d}_k^T}{\boldsymbol{d}_k^T \boldsymbol{d}_k} \right) \boldsymbol{H}_k
\end{aligned}$$

notice that $\boldsymbol{x}_{k+1}$, $\boldsymbol{f}_{k+1}$ and $\boldsymbol{z}_{k+1}$ are not used in $\boldsymbol{H}_{k+1}$ so that only $\boldsymbol{d}_k$ and its length need to be stored.

Algorithm (The Broyden method with low memory usage)

$k \leftarrow 0$; $\boldsymbol{x}$ *assigned*;
$\boldsymbol{f} \leftarrow \mathbf{F}(\boldsymbol{x})$; $\boldsymbol{H}_0 \leftarrow \nabla \mathbf{F}(\boldsymbol{x})^{-1}$; $\boldsymbol{d}_0 \leftarrow -\boldsymbol{H}_0 \boldsymbol{f}$; $\ell_0 \leftarrow \boldsymbol{d}_0^T \boldsymbol{d}_0$;
while $\|\boldsymbol{f}\| > \epsilon$ **do**
    — *perform step*
    $\boldsymbol{x} \leftarrow \boldsymbol{x} + \boldsymbol{d}_k$;
    $\boldsymbol{f} \leftarrow \mathbf{F}(\boldsymbol{x})$;
    — *evaluate* $\boldsymbol{H}_k \boldsymbol{f}$
    $\boldsymbol{z} \leftarrow \boldsymbol{H}_0 \boldsymbol{f}$;
    for $j = 0, 1, \ldots, k-1$ **do**
        $\boldsymbol{z} \leftarrow \boldsymbol{z} + \left[(\boldsymbol{d}_j^T \boldsymbol{z}) / \ell_j \right] \boldsymbol{d}_{j+1}$;
    end for
    — *update* $\boldsymbol{H}_{k+1}$
    $\boldsymbol{d}_{k+1} = -\left[\ell_k / (\ell_k + \boldsymbol{d}_k^T \boldsymbol{z}) \right] \boldsymbol{z}$;
    $\ell_{k+1} = \boldsymbol{d}_{k+1}^T \boldsymbol{d}_{k+1}$;
    $k \leftarrow k + 1$;
end while

Outline

Algorithm (The dumped Broyden method)

$k \leftarrow 0$; $\boldsymbol{x}_0$ *assigned*;
$\boldsymbol{f}_0 \leftarrow \mathbf{F}(\boldsymbol{x}_0)$; $\boldsymbol{H}_0 \leftarrow \nabla\mathbf{F}(\boldsymbol{x}_0)^{-1}$;
while $\|\boldsymbol{f}_k\| > \epsilon$ **do**
— *compute search direction*
$\boldsymbol{d}_k = -\boldsymbol{H}_k \boldsymbol{f}_k$;
Approximate $\arg\min_{\lambda>0} \|\mathbf{F}(\boldsymbol{x}_k + \lambda \boldsymbol{d}_k)\|^2$ *by line-search*;
— *perform step*
$\boldsymbol{s}_k = \lambda_k \boldsymbol{d}_k$;
$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \boldsymbol{s}_k$;
$\boldsymbol{f}_{k+1} = \mathbf{F}(\boldsymbol{x}_{k+1})$;
$\boldsymbol{y}_k = \boldsymbol{f}_{k+1} - \boldsymbol{f}_k$;
— *update* $\boldsymbol{H}_{k+1}$

$$\boldsymbol{H}_{k+1} = \boldsymbol{H}_k + \frac{(\boldsymbol{s}_k - \boldsymbol{H}_k \boldsymbol{y}_k)\boldsymbol{s}_k^T}{\boldsymbol{s}_k^T \boldsymbol{H}_k \boldsymbol{y}_k} \boldsymbol{H}_k;$$

$k \leftarrow k+1$;
end while

Elimination of $\boldsymbol{z}_k$ (1/5)

Notice that

$$\boldsymbol{H}_k\boldsymbol{y}_k = \boldsymbol{H}_k\boldsymbol{f}_{k+1} - \boldsymbol{H}_k\boldsymbol{f}_k = \boldsymbol{z}_{k+1} + \boldsymbol{d}_k, \quad \text{and} \quad \boldsymbol{s}_k = \lambda_k\boldsymbol{d}_k$$

and

$$\begin{aligned}
\boldsymbol{H}_{k+1} &= \boldsymbol{H}_k + \frac{(\boldsymbol{s}_k - \boldsymbol{H}_k\boldsymbol{y}_k)\boldsymbol{s}_k^T}{\boldsymbol{s}_k^T\boldsymbol{H}_k\boldsymbol{y}_k}\boldsymbol{H}_k \\
&= \boldsymbol{H}_k + \frac{(\lambda_k\boldsymbol{d}_k - \boldsymbol{z}_{k+1} - \boldsymbol{d}_k)\lambda_k\boldsymbol{d}_k^T}{\lambda_k\boldsymbol{d}_k^T(\boldsymbol{z}_{k+1} + \boldsymbol{d}_k)}\boldsymbol{H}_k \\
&= \left(\boldsymbol{I} + \frac{(\lambda_k\boldsymbol{d}_k - \boldsymbol{z}_{k+1} - \boldsymbol{d}_k)\boldsymbol{d}_k^T}{\boldsymbol{d}_k^T(\boldsymbol{z}_{k+1} + \boldsymbol{d}_k)}\right)\boldsymbol{H}_k \\
&= \left(\boldsymbol{I} - \frac{(\boldsymbol{z}_{k+1} + (1-\lambda_k)\boldsymbol{d}_k)\boldsymbol{d}_k^T}{\boldsymbol{d}_k^T\boldsymbol{d}_k + \boldsymbol{d}_k^T\boldsymbol{z}_{k+1}}\right)\boldsymbol{H}_k
\end{aligned}$$

Elimination of $\boldsymbol{z}_k$ (2/5)

A step of the broyden iterative scheme can be rewritten as

$$\begin{aligned}
\boldsymbol{d}_k &= -\boldsymbol{H}_k\boldsymbol{f}_k \\
\boldsymbol{x}_{k+1} &= \boldsymbol{x}_k + \lambda_k\boldsymbol{d}_k \\
\boldsymbol{f}_{k+1} &= \mathbf{F}(\boldsymbol{x}_{k+1}) \\
\boldsymbol{z}_{k+1} &= \boldsymbol{H}_k\boldsymbol{f}_{k+1} \\
\boldsymbol{H}_{k+1} &= \left(\boldsymbol{I} - \frac{(\boldsymbol{z}_{k+1} + (1-\lambda_k)\boldsymbol{d}_k)\boldsymbol{d}_k^T}{\boldsymbol{d}_k^T\boldsymbol{d}_k + \boldsymbol{d}_k^T\boldsymbol{z}_{k+1}}\right)\boldsymbol{H}_k
\end{aligned}$$

Elimination of $\boldsymbol{z}_k$ (3/5)

$$
\begin{aligned}
-\boldsymbol{d}_{k+1} &= \boldsymbol{H}_{k+1}\boldsymbol{f}_{k+1} \\
&= \left(\boldsymbol{I} - \frac{(\boldsymbol{z}_{k+1} + (1-\lambda_k)\boldsymbol{d}_k)\boldsymbol{d}_k^T}{\boldsymbol{d}_k^T\boldsymbol{d}_k + \boldsymbol{d}_k^T\boldsymbol{z}_{k+1}}\right)\boldsymbol{H}_k\boldsymbol{f}_{k+1} \\
&= \left(\boldsymbol{I} - \frac{(\boldsymbol{z}_{k+1} + (1-\lambda_k)\boldsymbol{d}_k)\boldsymbol{d}_k^T}{\boldsymbol{d}_k^T\boldsymbol{d}_k + \boldsymbol{d}_k^T\boldsymbol{z}_{k+1}}\right)\boldsymbol{z}_{k+1} \\
&= \boldsymbol{z}_{k+1} - \frac{(\boldsymbol{z}_{k+1} + (1-\lambda_k)\boldsymbol{d}_k)\boldsymbol{d}_k^T\boldsymbol{z}_{k+1}}{\boldsymbol{d}_k^T\boldsymbol{d}_k + \boldsymbol{d}_k^T\boldsymbol{z}_{k+1}} \\
&= \frac{(\boldsymbol{d}_k^T\boldsymbol{d}_k)\boldsymbol{z}_{k+1} + (\lambda_k - 1)(\boldsymbol{d}_k^T\boldsymbol{z}_{k+1})\boldsymbol{d}_k}{\boldsymbol{d}_k^T\boldsymbol{d}_k + \boldsymbol{d}_k^T\boldsymbol{z}_{k+1}}
\end{aligned}
$$

Elimination of $\boldsymbol{z}_k$ (4/5)

Solving for $\boldsymbol{z}_{k+1}$

$$
\boldsymbol{z}_{k+1} = -\boldsymbol{d}_{k+1} - \frac{(\boldsymbol{d}_k^T\boldsymbol{z}_{k+1})}{\boldsymbol{d}_k^T\boldsymbol{d}_k}(\boldsymbol{d}_{k+1} + (\lambda_k - 1)\boldsymbol{d}_k)
$$

and adding on both side $(1-\lambda_k)\boldsymbol{d}_k$

$$
\begin{aligned}
\boldsymbol{z}_{k+1} + (1-\lambda_k)\boldsymbol{d}_k &= -(\boldsymbol{d}_{k+1} + (\lambda_k - 1)\boldsymbol{d}_k)\left(1 + \frac{(\boldsymbol{d}_k^T\boldsymbol{z}_{k+1})}{\boldsymbol{d}_k^T\boldsymbol{d}_k}\right) \\
&= -(\boldsymbol{d}_{k+1} + (\lambda_k - 1)\boldsymbol{d}_k)\frac{\boldsymbol{d}_k^T\boldsymbol{d}_k + \boldsymbol{d}_k^T\boldsymbol{z}_{k+1}}{\boldsymbol{d}_k^T\boldsymbol{d}_k}
\end{aligned}
$$

and substituting in $\boldsymbol{H}_{k+1}$ we have

$$
\boldsymbol{H}_{k+1} = \left(\boldsymbol{I} + \frac{(\boldsymbol{d}_{k+1} + (\lambda_k - 1)\boldsymbol{d}_k)\boldsymbol{d}_k^T}{\boldsymbol{d}_k^T\boldsymbol{d}_k}\right)\boldsymbol{H}_k
$$

Elimination of $\boldsymbol{z}_k$ (5/5)

Substituting into the step of the broyden iterative scheme and assuming $\boldsymbol{d}_k$ known

$$
\begin{aligned}
\boldsymbol{x}_{k+1} &= \boldsymbol{x}_k + \lambda_k \boldsymbol{d}_k \\
\boldsymbol{f}_{k+1} &= \mathbf{F}(\boldsymbol{x}_{k+1}) \\
\boldsymbol{z}_{k+1} &= \boldsymbol{H}_k \boldsymbol{f}_{k+1} \\
\boldsymbol{d}_{k+1} &= -\frac{(\boldsymbol{d}_k^T\boldsymbol{d}_k)\boldsymbol{z}_{k+1} + (\lambda_k - 1)(\boldsymbol{d}_k^T\boldsymbol{z}_{k+1})\boldsymbol{d}_k}{\boldsymbol{d}_k^T\boldsymbol{d}_k + \boldsymbol{d}_k^T\boldsymbol{z}_{k+1}} \\
\boldsymbol{H}_{k+1} &= \left(\boldsymbol{I} + \frac{(\boldsymbol{d}_{k+1} + (\lambda_k - 1)\boldsymbol{d}_k)\boldsymbol{d}_k^T}{\boldsymbol{d}_k^T\boldsymbol{d}_k}\right)\boldsymbol{H}_k
\end{aligned}
$$

notice that $\boldsymbol{x}_{k+1}$, $\boldsymbol{f}_{k+1}$ and $\boldsymbol{z}_{k+1}$ are not used in $\boldsymbol{H}_{k+1}$ so that only $\boldsymbol{d}_k$ and its length need to be stored.

Algorithm (The dumped Broyden method)

$k \leftarrow 0$; $\boldsymbol{x}$ *assigned*;
$\boldsymbol{f} \leftarrow \mathbf{F}(\boldsymbol{x})$; $\boldsymbol{H}_0 \leftarrow \nabla\mathbf{F}(\boldsymbol{x})^{-1}$; $\boldsymbol{d}_0 \leftarrow -\boldsymbol{H}_0\boldsymbol{f}$; $\ell_0 \leftarrow \boldsymbol{d}_0^T\boldsymbol{d}_0$;
while $\|\boldsymbol{f}_k\| > \epsilon$ **do**
  Approximate $\arg\min_{\lambda>0} \|\mathbf{F}(\boldsymbol{x} + \lambda\boldsymbol{d}_k)\|^2$ *by line-search;*
  — *perform step*
  $\boldsymbol{x} \leftarrow \boldsymbol{x} + \lambda_k\boldsymbol{d}_k$;
  $\boldsymbol{f} \leftarrow \mathbf{F}(\boldsymbol{x})$;
  —- *evaluate* $\boldsymbol{H}_k\boldsymbol{f}$
  $\boldsymbol{z} \leftarrow \boldsymbol{H}_0\boldsymbol{f}$;
  for $j = 0, 1, \dots, k-1$ **do**
    $\boldsymbol{z} \leftarrow \boldsymbol{z} + \big[(\boldsymbol{d}_j^T\boldsymbol{z})/\ell_j\big]\big(\boldsymbol{d}_{j+1} + (\lambda_j - 1)\boldsymbol{d}_j\big)$;
  — *update* $\boldsymbol{H}_{k+1}$
  $\boldsymbol{d}_{k+1} = -\big[\ell_k\boldsymbol{z} + (\lambda_k - 1)(\boldsymbol{d}_k^T\boldsymbol{z})\boldsymbol{d}_k\big]/(\ell_k + \boldsymbol{d}_k^T\boldsymbol{z})$;
  $\ell_{k+1} = \boldsymbol{d}_{k+1}^T\boldsymbol{d}_{k+1}$;
  $k \leftarrow k + 1$;
end while

Some additional reference

C. G. Broyden
A Class of Methods for Solving Nonlinear Simultaneous Equations
Mathematics of Computation, **19**, No. 92, pp. 577–593

C.G. Broyden
On the discovery of the "good Broyden" method
Mathematical Programming, **87**, Number 2, 2000

E. Bertolazzi, F. Biral and M. Da Lio
Symbolic-numeric efficient solution of optimal control problems for multibody systems
Journal of Computational and Applied Mathematics, **185**, 2006

Outline

Stopping criteria for q-convergent sequences (1/2)

1. Consider an iterative scheme that produce a sequence $\{x_k\}$ which converge to α with q-order p.
2. This means that there exists a constant C such that
$$|x_{k+1}-\alpha| \leq C\,|x_k-\alpha|^p \qquad \text{for } k \geq m$$
3. If $\lim_{k\mapsto\infty} \dfrac{|x_{k+1}-\alpha|}{|x_k-\alpha|^p}$ exists and is say C we have
$$|x_{k+1}-\alpha| \approx C\,|x_k-\alpha|^p \qquad \text{for large } k$$
4. We can use this last expression to obtain an error estimate for the error and the values of p if unknown using the only known values.

Stopping criteria and q-order estimation

Stopping criteria q-convergent sequences (2/2)

1. If $|x_{k+1}-\alpha| \leq C\,|x_k-\alpha|^p$ we can write:
$$\begin{aligned}|x_k-\alpha| &\leq |x_k-x_{k+1}| + |x_{k+1}-\alpha| \\ &\leq |x_k-x_{k+1}| + C\,|x_k-\alpha|^p\end{aligned}$$
$$\Downarrow$$
$$|x_k-\alpha| \leq \frac{|x_k-x_{k+1}|}{1-C\,|x_k-\alpha|^{p-1}}$$
2. If x_k is so near the solution such that $C\,|x_k-\alpha|^{p-1} \leq \frac{1}{2}$ then
$$|x_k-\alpha| \leq 2\,|x_k-x_{k+1}|$$
3. This justify the stopping criteria
$$|x_{k+1}-x_k| \leq \tau \qquad \text{Absolute tolerance}$$
$$|x_{k+1}-x_k| \leq \tau\max\{|x_k|,|x_{k+1}|\} \qquad \text{Relative tolerance}$$

Estimation of the q-order (1/3)

1. Consider an iterative scheme that produce a sequence $\{x_k\}$ which converge to α with q-order p.
2. If $|x_{k+1} - \alpha| \approx C\,|x_k - \alpha|^p$ then the ratio:

$$\log \frac{|x_{k+1} - \alpha|}{|x_k - \alpha|} \approx \log \frac{C\,|x_k - \alpha|^p}{|x_k - \alpha|} = (p-1)\log C^{\frac{1}{p-1}}\,|x_k - \alpha|$$

and analogously

$$\log \frac{|x_{k+2} - \alpha|}{|x_{k+1} - \alpha|} \approx \log \frac{C^{1+p}\,|x_k - \alpha|^{p^2}}{C\,|x_k - \alpha|^p} = p(p-1)\log C^{\frac{1}{p-1}}\,|x_k - \alpha|$$

3. From this two ratio we can deduce p as

$$\log \frac{|x_{k+2} - \alpha|}{|x_{k+1} - \alpha|} \Big/ \log \frac{|x_{k+1} - \alpha|}{|x_k - \alpha|} \approx p$$

Stopping criteria and q-order estimation

Estimation of the q-order (2/3)

1. The ratio

$$\log \frac{|x_{k+2} - \alpha|}{|x_{k+1} - \alpha|} \Big/ \log \frac{|x_{k+1} - \alpha|}{|x_k - \alpha|} \approx p$$

uses the error which is not known.

2. If we are near the solution we can use the estimation $|x_k - \alpha| \approx |x_{k+1} - x_k|$ so that

$$\log \frac{|x_{k+2} - x_{k+3}|}{|x_{k+1} - x_{k+2}|} \Big/ \log \frac{|x_{k+1} - x_{k+2}|}{|x_k - x_{k+1}|} \approx p$$

so that 3 iteration are enough to estimate the q-order of a sequence.

Estimation of the q-order (3/3)

1. if the the step length is proportional to the value of $f(x)$ as in Newton-Raphson scheme, i.e. $|x_k - \alpha| \approx M\,|f(x_k)|$ we can simplify the previous formula as:

$$\log \frac{|f(x_{k+2})|}{|f(x_{k+1})|} \Big/ \log \frac{|f(x_{k+1})|}{|f(x_k)|} \approx p$$

2. Such estimation are useful to check code implementation. In fact if we expect order p and we see order $r \neq p$ there is something wrong in the implementation or in the theory!

References

J. Stoer and R. Bulirsch
Introduction to numerical analysis
Springer-Verlag, Texts in Applied Mathematics, **12**, 2002.

J. E. Dennis, Jr. and Robert B. Schnabel
Numerical Methods for Unconstrained Optimization and Nonlinear Equations
SIAM, Classics in Applied Mathematics, **16**, 1996.

Jorge Nocedal, and Stephen J. Wright
Numerical optimization
Springer, 2006