Numerical Methods for Computational Science and Engineering

Fall Semester 2017 (HS17)

Prof. Rima Alaifari, SAM, ETH Zurich

8.5 Unconstrained Optimization

Optimization problems we have already seen:

- Least-squares solution:

  Find $x \in \mathbb{K}^n$ s.t. $\|Ax-b\|_2 \to \min$

- generalized solution:

  Find least sq. solution x to $Ax=b$

  s.t. $\|x\|_2 \to \min$

- norm-constrained extrema

  Given $A \in \mathbb{K}^{m,n}$ $m \geq n$

  Find $x \in \mathbb{K}^n$, $\|x\|_2 = 1$ s.t. $\|Ax\|_2 \to \min$

- best low-rank approximation

  Given $A \in \mathbb{K}^{m,n}$, find $\tilde{A} \in \mathbb{K}^{m,n}$, $\operatorname{rank}(\tilde{A}) \leq k$

  s.t. $\|A-\tilde{A}\| \to \min$ over rank-k matrices

  $\uparrow$ 2-norm / F-norm

- total least-squares problem

  Given $A \in \mathbb{R}^{m,n}$, $m > n$, $\operatorname{rank}(A) = n$, $b \in \mathbb{R}^m$

  Find $\hat{A} \in \mathbb{R}^{m,n}$, $\hat{b} \in \mathbb{R}^m$ s.t.

  $\|[A \; b] - [\hat{A} \; \hat{b}]\|_F \to \min$ with $\hat{b} \in \mathcal{R}(\hat{A})$

General question: $F: \mathbb{R}^n \to \mathbb{R}$

How to find min/max of F? [unconstrained optimization]

Application: Maximum likelihood estimation [machine learning]

Suppose some quantity can be modelled with a probability distribution:

for example; weight of 5-year olds in Switzerland

$\rightsquigarrow$ normal distribution

Can we estimate mean μ & variance σ through randomized samples?

Sample $\{w_1, \dots, w_n\}$

$$f(w; \mu, \sigma) = \frac{1}{\sqrt{2\pi}\,\sigma} e^{-(w-\mu)^2/2\sigma^2}$$

$f(w_i; \mu, \sigma)$ likelihood to observe weight w_i for child i

weight of child i is independent of weight of child j

$$P(\{w_1, \dots, w_n\}; \mu, \sigma) = \prod_{j=1}^{n} f(w_j; \mu, \sigma)$$

$\uparrow$

view as function in μ & σ, ($\{w_1, \dots, w_n\}$ is fixed)

maximize P to estimate μ & σ.

In practice: maximize $\log P$ instead

[same location of max, but better numerical properties] maximizing $F \Leftrightarrow$ minimizing $-F$

Therefore: only consider minimization problems

Global vs local minimum:

- x^* is a global minimum of $F: \mathbb{R}^n \to \mathbb{R}$ if

$$F(x^*) \leq F(x) \qquad \forall x \in \mathbb{R}^n$$

- x^* is a local minimum of $F: \mathbb{R}^n \to \mathbb{R}$ if

there is $\varepsilon > 0$ s.t. $\forall x$ with $\underbrace{\|x - x^*\| \leq \varepsilon}_{\varepsilon\text{-ball around } x^*}$

$$F(x^*) \leq F(x)$$

8.5.1 Optimization with differentiable objective function

$F: \mathbb{R}^n \to \mathbb{R}$ differentiable

∇F direction of greatest increase

$-\nabla F$ direction of steepest descent

Why? Locally around $\bar{x}$

$$F(x) \approx F(\bar{x}) + \nabla F(\bar{x})^T (x - \bar{x})$$

$x - \bar{x} = \tau \nabla F(\bar{x})$:

$$F(\bar{x} + \tau \nabla F(\bar{x})) \approx F(\bar{x}) + \tau \|\nabla F(\bar{x})\|^2$$

$\tau > 0$ increase

$\tau < 0$ decrease

Stationary point: $\nabla F(x) = 0$

could be local/global max/min, saddle point

If F is twice diff. we can check the Hessian matrix at a stationary point:

$$H_F(x) = \left(\frac{\partial^2 F}{\partial x_i \, \partial x_j}(x) \right)_{i,j=1}^{n}$$

Taylor expansion:

$$F(x) \approx F(\bar{x}) + \underbrace{\nabla F(\bar{x})^T (x-\bar{x})}_{=0} + \frac{1}{2} (x-\bar{x})^T H_F(\bar{x}) (x-\bar{x})$$

if $\bar{x}$ stat.:

$$F(x) \approx F(\bar{x}) + \underbrace{\frac{1}{2} (x-\bar{x})^T H_F(\bar{x}) (x-\bar{x})}_{\text{increase/decrease/unclear}}$$

$H_F(\bar{x})$ pos. def.: $(x-\bar{x})^T H_F(\bar{x}) (x-\bar{x}) > 0$

locally: ε-ball around $\bar{x}$ s.t.

$F(x) \geq F(\bar{x}) \Rightarrow \bar{x}$ local minimum

$H_F(\bar{x})$ neg. def.: $\bar{x}$ local maximum

$H_F(\bar{x})$ indefinite: $\bar{x}$ saddle point

$H_F(\bar{x})$ not invertible: e.g. whole region of saddle points [unlikely]

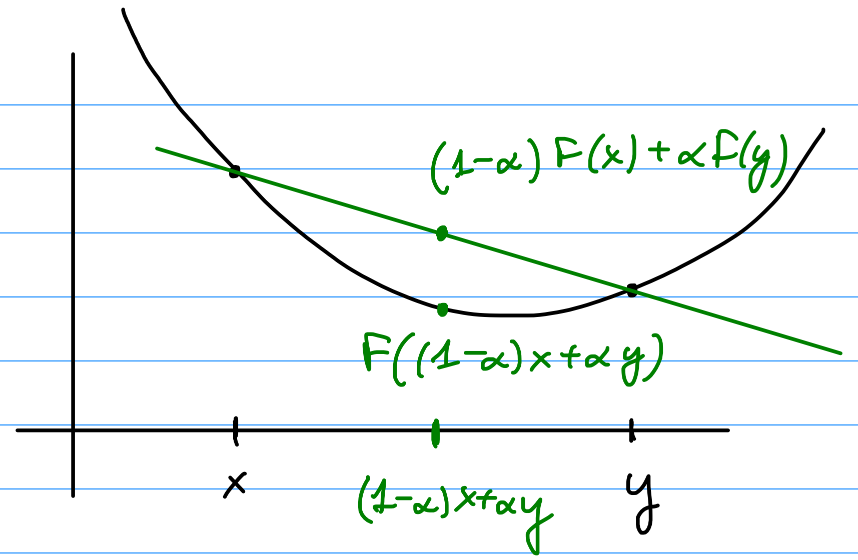
Check positive definiteness for example by checking whether Cholesky factorization exists [cf. Exercises]

8.5.2. Optimization with convex objective function

Definition [convex function]: A function $F: \mathbb{R}^n \to \mathbb{R}$ is called convex, if for all $x, y \in \mathbb{R}^n$ and all $\alpha \in (0,1)$:

$$F((1-\alpha)x + \alpha y) \leq (1-\alpha) F(x) + \alpha F(y)$$

$(<)$ (strictly convex)



Lemma [minimum of convex function]: If $\bar{x} \in \mathbb{R}^n$ is a local minimum of $F: \mathbb{R}^n \to \mathbb{R}$, then it is a global minimum.

Proof: Let $\bar{x} \in \mathbb{R}^n$ be a local minimum of F and $x_0 \in \mathbb{R}^n$ s.t. $F(x_0) < F(\bar{x})$ [i.e. $\bar{x}$ is not a global min]

For $\alpha \in (0,1)$, convexity implies

$$F(\underbrace{\bar{x} + \alpha(x_0 - \bar{x})}_{\alpha x_0 + (1-\alpha)\bar{x}}) \leq (1-\alpha)F(\bar{x}) + \alpha \underbrace{F(x_0)}_{< F(\bar{x})} < F(\bar{x})$$

Take sequence $\alpha_k \to 0$: $\bar{x}$ cannot be a local min.

(in every ε-neighborhood around $\bar{x}$ there is x with $F(x) < F(\bar{x})$).

$\lightning$ contradicts ass. that $\bar{x}$ is local min. $\square$

8.5.3 Methods in 1D

$f: \mathbb{R} \to \mathbb{R}$

- Newton's method (or variants) applied to f' if $f \in C^2$

  iterate $x_{k+1} = x_k - \dfrac{f'(x_k)}{f''(x_k)}$

Note: Newton's method for minimization

$\Leftrightarrow$ approximate function locally by parabola & look for its vertex

$$f(x) \approx \underbrace{f(x_k) + f'(x_k)(x-x_k) + \frac{1}{2} f''(x_k)(x-x_k)^2}_{\text{parabola with vertex } x_k - \frac{f'(x_k)}{f''(x_k)}}$$

- Golden Section Search

Algorithm for non-diff. f ?

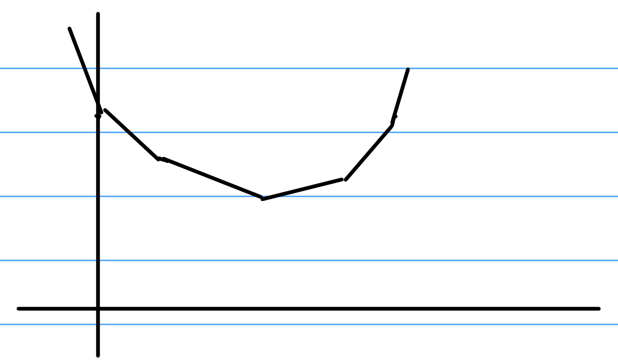
Definition [unimodal]: A function $f:[a,b] \to \mathbb{R}$ is called unimodal if there exists $x_u \in [a,b]$ s.t.

f is strict monotonically decreasing on $[a, x_u]$ and

— " — increasing on $[x_u, b]$.

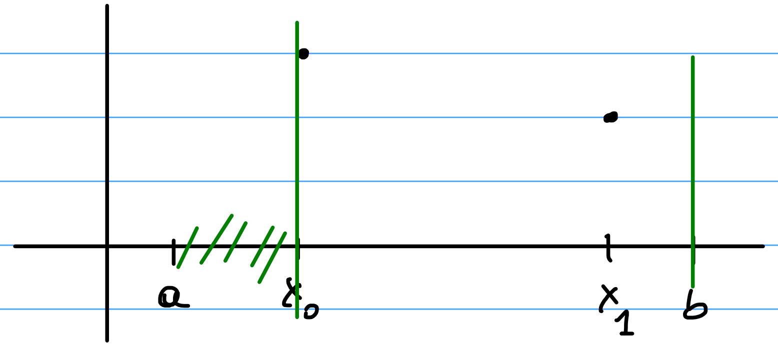
Example: $f(x) = |x|$ absolute value function.

unimodal function: local minimum is global minimum



Idea: Suppose for 2 values x_0, x_1 s.t. $a < x_0 < x_1 < b$

we know $f(x_0) \geq f(x_1)$



then: we can discard interval $[a, x_0]$!

if instead $f(x_0) \leq f(x_1)$: discard $[x_1, b]$!

$\to$ iterate!

Suppose $a=0, \quad b=1$

$$x_0^{(0)} = 1-\lambda \qquad \lambda \in \left(\tfrac{1}{2}, 1\right)$$

$$x_1^{(0)} = \lambda$$

If we can discard $[x_1^{(0)}, 1]$

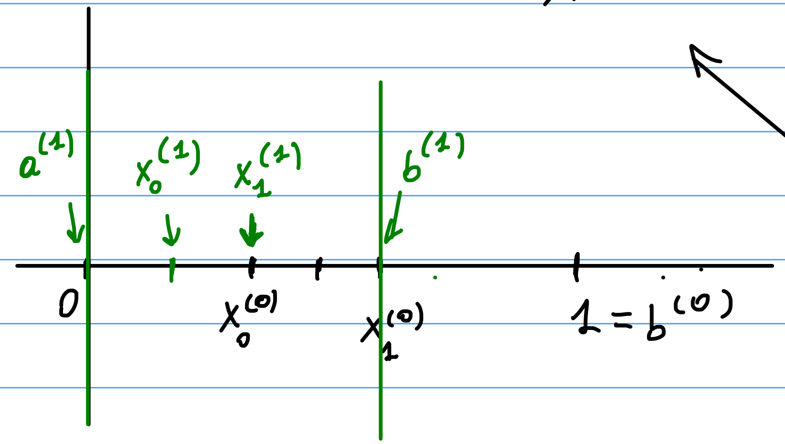
Search over $[0, x_1^{(0)}]$ and $f(1-\lambda)$ is already known $\to$ reuse

$[0, \lambda]$ now divided s.t. new points are $x_0^{(1)} = (1-\lambda)\lambda$

$$x_1^{(1)} = \lambda^2$$

How to reuse $f(1-\lambda)$? Define λ s.t.

$$\lambda^2 = 1-\lambda \qquad [\text{i.e. } x_1^{(1)} = x_0^{(0)}]$$

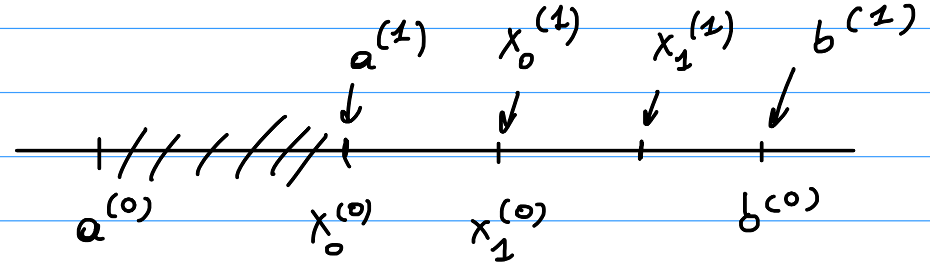


positive solution:

$$\lambda = \tfrac{1}{2}\left(\sqrt{5}-1\right)$$

($\varphi = \lambda + 1$ is golden ratio)

If we discard $[0, x_0^{(0)}]$:



Algorithm [Golden Section Search]:

- initialize $\quad x_0 = a + (1-\lambda)(b-a), \quad x_1 = a + \lambda(b-a)$

  $f_0 = f(x_0), \quad f_1 = f(x_1)$

- while $|b-a| > tol$

  if $f_0 \geq f_1$ $\quad$ [remove $[a^{(j-1)}, x_0^{(j-1)}]$]

  $a \leftarrow x_0 \qquad [a^{(j)} = x_0^{(j-1)}]$

  $x_0 \leftarrow x_1, \ f_0 \leftarrow f_1 \qquad [x_0^{(j)} = x_1^{(j-1)}]$

  $x_1 \leftarrow a + \lambda(b-a), \ f_1 \leftarrow f(x_1) \qquad [x_1^{(j)} = a^{(j)} + \lambda(b^{(j)} - a^{(j)})]$

if $f_1 > f_0$ $\quad$ [remove $[x_1^{(j-1)}, b^{(j-1)}]$]

$b \leftarrow x_1$

$x_1 \leftarrow x_0$, $f_1 \leftarrow f_0$

$x_0 \leftarrow a + (1-\lambda)(b-a)$, $f_0 \leftarrow f(x_0)$

if f is unimodal on $[a,b]$, this alg. converges to the global minimum

In each iteration: interval size is reduced by factor

$0.618 \approx \lambda < 1$

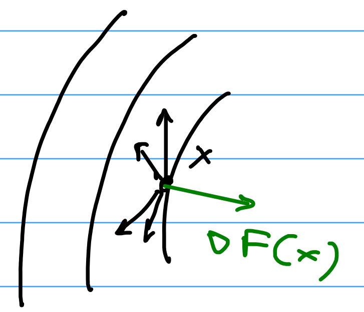
i.e. linear-type convergence as for bisection [root-finding]

if f has multiple local minima: Golden Section Search finds some local minimum

8.5.4 Methods in Higher Dimensions

8.5.4.1 Gradient descent

level sets



Δx is descent direction $\quad \nabla F(x)^T \Delta x < 0$

$\Delta x = -\nabla F(x)$ $\quad$ steepest descent / gradient descent direction

Guarantee for gradient descent direction:

if $\nabla F(x) \neq 0$ and $\alpha > 0$ sufficiently small, then

$$F(x - \alpha \nabla F(x)) \leq F(x)$$

Gradient descent iteration:

$$x^{(k+1)} = x^{(k)} - t^{(k)} \nabla F(x^{(k)})$$

$\uparrow$ finding step size is a 1D problem

algorithm:
- start with initial guess $x^{(0)}$
- while stopping criterion not satisfied

  (e.g.: $\|\nabla F\|_2 < \mathrm{tol}$)

  1.) take $g^{(k)}(t) = F(x^{(k)} - t \nabla F(x^{(k)}))$

  2.) find step size t^* through line search

  e.g.: $t^* = \underset{t \geq 0}{\operatorname{argmin}}\, g^{(k)}(t)$

  3.) take $x^{(k+1)} = x^{(k)} - t^* \nabla F(x^{(k)})$.

In each iteration $F(x^{(k)})$ decreases

terminates when $\nabla F(x^{(k)}) \approx 0$

How to find t^*?

- exact line search $\left(t^* = \underset{t \geq 0}{\operatorname{argmin}}\, g^{(k)}(t) \right)$

- backtracking line search:

$$F(x - t \nabla F(x)) \approx \underline{F(x) - t \|\nabla F(x)\|^2} \quad \text{for } t \text{ small enough}$$

$$< F(x) - \alpha t \|\nabla F(x)\|^2$$

for some $\alpha \in (0,1)$

Idea: Start with $t = 1$, fix $\alpha \in (0, 0.5)$:

decrease t until

$$F(x - t \nabla F(x)) < F(x) - \alpha t \|\nabla F(x)\|^2 \qquad (*)$$

$\nearrow$

(*) : iterate until "good decrease" is reached

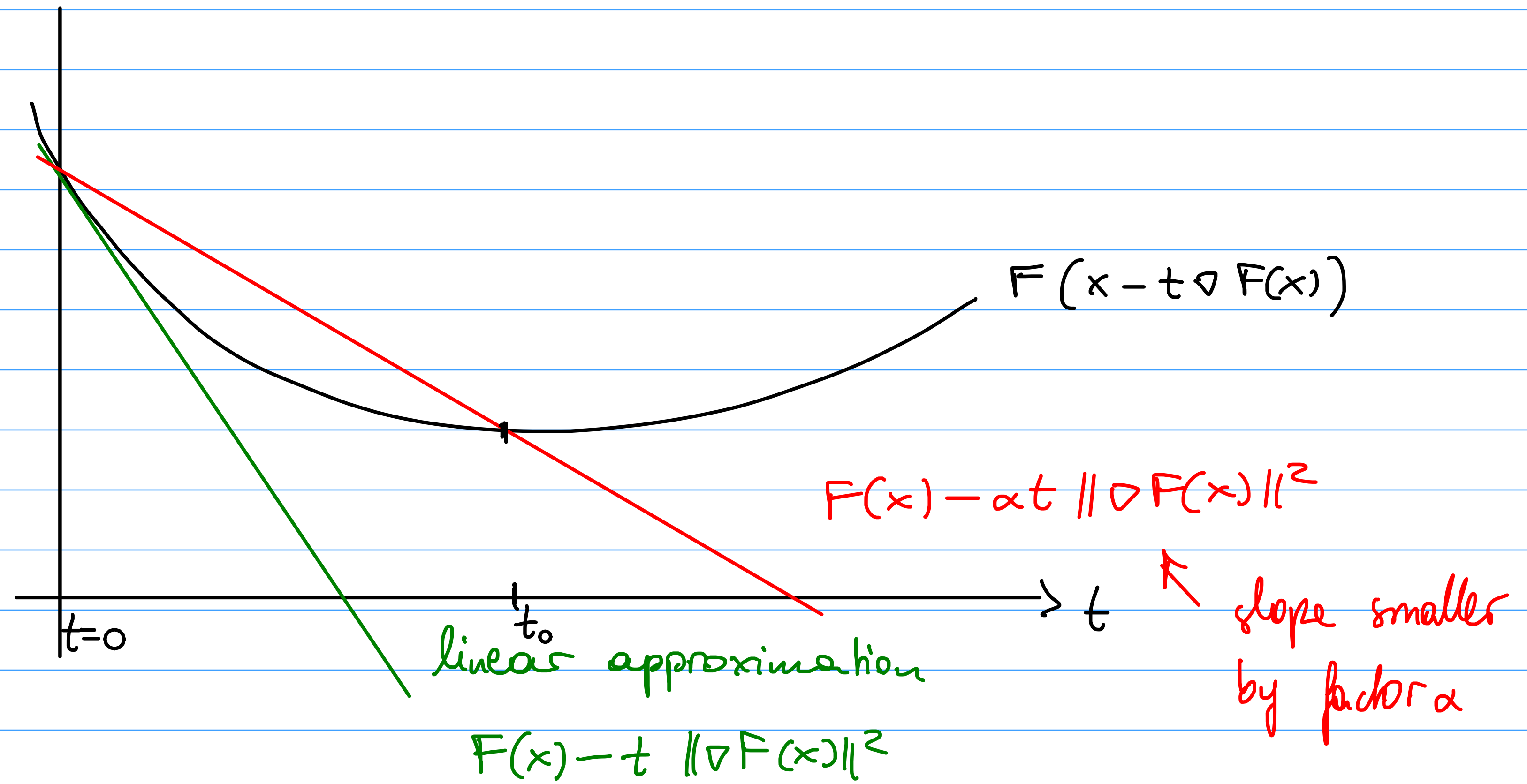
start with $t=1$, fix $\alpha \in (0, 0.5)$, $\beta \in (0,1)$:

while $F(x - t\nabla F(x)) > F(x) - \alpha t \|\nabla F(x)\|^2$

$t \leftarrow \beta t$

guarantees : $F(x^{(k)}) - F(x^{(k+1)}) > \alpha t \|\nabla F(x^{(k)})\|^2$

↑ decrease in F



backtracking: start at $t=1$ and stop when $t \leq t_0$ for the first time.

8.5.4.2. Newton's method

As in 1D: If F is twice diff.

$$F(x) \approx F(x^{(k)}) + \nabla F(x^{(k)})^T (x - x^{(k)})$$
$$+ \frac{1}{2} (x - x^{(k)})^T H_F(x^{(k)}) (x - x^{(k)})$$

Differentiate RHS and set to zero (minimum of quadr. approximation) suggests:

$$x^{(k+1)} = x^{(k)} - \left[H_F(x^{(k)}) \right]^{-1} \nabla F(x^{(k)})$$

near a minimum: quadratic convergence (faster than linear conv. of gradient descent)

Gradient descent:	Newton's method:
line search in every iteration	computing H_F & solving LSE in each iteration

But:
- Newton's method needs fewer iteration
- Gradient descent typically converges on a larger region than Newton's method

Note: both can get stuck at local minima or saddle points

8.5.4.3. BFGS method

↑

Quasi-Newton

Instead of computing & solving for the Hessian $H_F(x^{(k)})$ approximate by B_k s.t. B_{k+1} is obtained from simple update of B_k.

Newton's method:

$$x^{(k+1)} - x^{(k)} = - \left[H_F(x^{(k)}) \right]^{-1} \nabla F(x^{(k)})$$

Approximation B_k of $H_F(x^{(k)})$:

approximation of derivative of $\nabla F(x^{(k)})$

secant-like condition as for Broyden's method:

$$B_{k+1} (\underbrace{x^{(k+1)} - x^{(k)}}_{=: s^{(k)}}) = \underbrace{\nabla F(x^{(k+1)}) - \nabla F(x^{(k)})}_{=: y^{(k)}}$$

$$B_{k+1} s^{(k)} = y^{(k)} \qquad (*)$$

But now: B_{k+1} to be s.p.d.!

BFGS update:

$$B_{k+1} = B_k + \alpha u u^T + \beta v v^T$$

s.t. $(*)$ holds.

Choose: $u = y^{(k)}$, $v = B_k s^{(k)}$

$$\alpha = \frac{1}{y^{(k)T} s^{(k)}}, \quad \beta = - \frac{1}{s^{(k)T} B_k s^{(k)}}$$

$$\Rightarrow \quad (B_k + \alpha u u^T + \beta v v^T) s^{(k)} = y^{(k)} \quad \checkmark$$

BFGS update becomes:

$$B_{k+1} = B_k + \frac{y^{(k)} y^{(k)T}}{y^{(k)T} s^{(k)}} - \frac{B_k s^{(k)} s^{(k)T} B_k^T}{s^{(k)T} B_k s^{(k)}}$$

With Sherman-Morrison-Woodbury formula:

$$B_{k+1}^{-1} = \left(I - \frac{s^{(k)} y^{(k)T}}{y^{(k)T} s^{(k)}} \right) B_k^{-1} \left(I - \frac{y^{(k)} s^{(k)T}}{y^{(k)T} s^{(k)}} \right) + \frac{s^{(k)} s^{(k)T}}{y^{(k)T} s^{(k)}}$$

Variant : L-BFGS : no storage of dense matrix B_k

$\uparrow$ limited memory