constrained Nonconvex optimization (lots of variants)

$$\min_x f_0(x) \quad \text{s.t. } f_i(x) \le 0, \; i = 1 \ldots m$$

want: "global convergence": for any feasible starting $x^{(1)}$, → local optimum

Typical structure: "trust region" algorithm

1. at current guess $x^{(k)}$, construct simple approximations $g_i(x) \approx f_i(x)$ near $x^{(k)}$
   (e.g. Taylor expansion)
   (often convex ⇒ 2 easy)

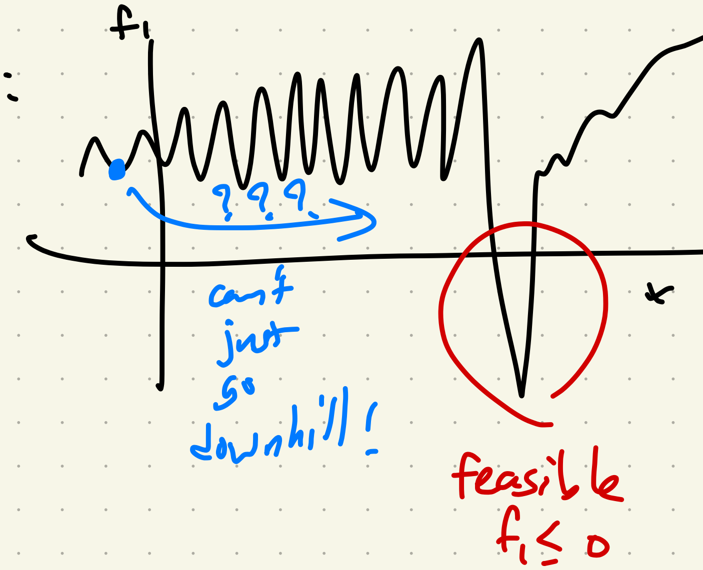
2. solve $\min_{x \in T} g_0(x)$ st. $g_i(x) \le 0$ — constrain $\|x - x^{(k)}\|$ to be small enough = trust region T
   ⇒ $x^{(k+1)}$? candidate

3. check is $x^{(k+1)}$ good enough?
   eval $f_i(x^{(k+1)})$: does f_0 decrease? feasible?
   if yes: go to 1, possibly update T
   if no: update g_i and/or shrink T + goto 2

details matter!

given infeasible starting point, finding a feasible point may be hard:

$\sim$ global optimization



ex: SLP = sequential linear programming:

$$g_i = f_i(x^{(k)}) + \nabla f_i\big|_{x^{(k)}} \cdot (x - x^{(k)})$$

$= 1^{st}$-order Taylor $\Longrightarrow$ (2) is LP if T = box (SOCP if T = sphere)

... still tricky
details in (3)

SQP: $f_0 \approx$ quadratic (convex?)

... tricky: computing $\frac{\partial f_0}{\partial x_i \partial x_j}$ = Hessian matrix

expensive ($\sim n$ gradients), big

$\Longrightarrow$ approximate Hessians: BFGS updates

CCSA algorithm (Svanberg, 2002)

= conservative, convex, separable approximations

solves $\min_{x \in X \subseteq \mathbb{R}^n} f_0(x)$

s.t. $f_i(x) \leq 0$

$i = 1 \ldots m$

X = box constraint
$= \{x \mid x_j^{min} \leq x_j \leq x_j^{max}\}$

globally convergent for any f_i with bounded 2^{nd} derivs as long as feasible set has non-empty interior: there must exist a strictly feasible x: $f_i(x) < 0$ for $i = 1 \ldots m$ (no equality constraints)

- relatively simple, robust
- illustrates many features: trust regions, convexity, duality (Lagrange multipliers), ... penalty terms
- scales to millions of variables, constraints
  - can take advantage of sparsity in ∇f_i
- commonly used in topology optimization of PDEs

core approximation (step 1): (several variants of CCSA)

for $x^{(k)}$ = current guess,

1

$$f_i(x) \approx g_i(x) = \underbrace{f_i(x^{(k)}) + \nabla f_i\Big|_{x^{(k)}} \cdot (x - x^{(k)})}_{\text{1st-order Taylor}}$$

$i = 0 \ldots m$

$$+ \frac{\rho_i}{2} \sum_{j=1}^{n} \frac{(x_j - x_j^{(k)})^2}{\sigma_j^2}$$

for penalty weight $\rho_i > 0$ to be determined

and σ_j = diameter of trust region (to be determined)

$$T = \left\{ x \,\Big|\, |x_j - x_j^{(k)}| \leq \sigma_j \right\} \cap X$$

2

$$\min_{x \in T} g_0(x)$$

$$\text{s.t. } g_i(x) \leq 0$$

$$i = 1 \ldots m$$

easy: g_i's are convex, separable ($= \sum_j$ functions of each x_j)

③ given candidate $x^{(k+1)}$ from ②,

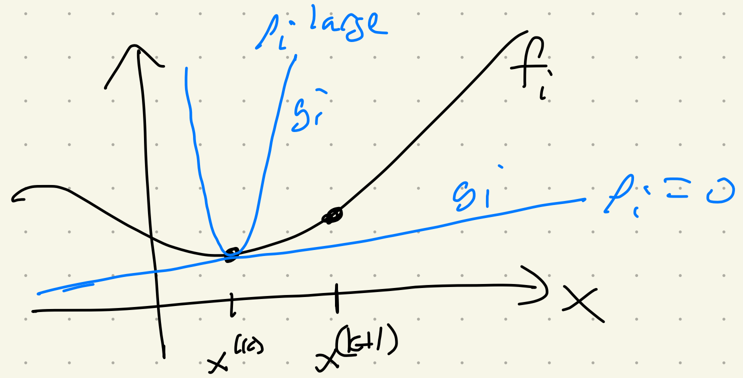
evaluate $f_i(x^{(k+1)})$, $g_i(x^{(k+1)})$

good step: in CCSA $g_i(x^{(k+1)}) \geq f_i(x^{(k+1)})$

for all $i = 0 \ldots m$

$\Rightarrow$ goto ① (outer iteration)

"conservative" approximation



outer iterations = sequence of feasible improvements

... can stop early + still satisfies constraints

if true: guarantees f_0 improved

and $f_i \leq 0$ (feasible)
$i = 1 \ldots m$

2^{nd}-deriv f_i bounded $\Rightarrow$ finite # of inner iterations

bad step: some $g_i < f_i$ at $x^{(k+1)}$

$\Rightarrow$ increase ρ_i for that i

and goto ② (inner iteration)

(e.g. $\rho_i \rightarrow \rho_i \times 2$)

on outer iteration (good $x^{(k+1)}$),

two additional updates:

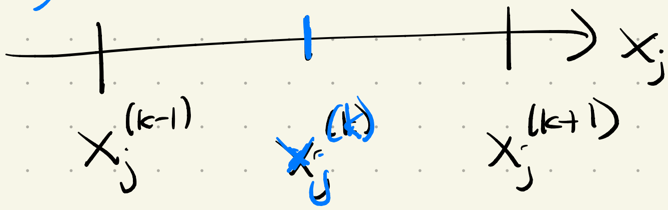
i) decrease all ρ_i by some factor, e.g. $\frac{1}{2}$,

(allows larger subsequent step, if possible)

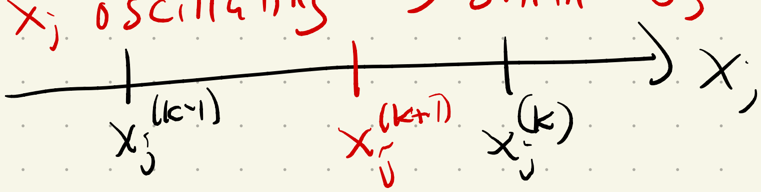
ii) update σ_j's (trust region diameters)

$$\sigma_j \to (\text{old } \sigma_j) \times \begin{cases} 0.5 \ \text{(x_j oscillating)} & \text{if } \left[x_j^{(k+1)} - x_j^{(k)}\right]\cdot\left[x_j^{(k)} - x_j^{(k-1)}\right] < 0 \\ 2.0 & \text{if } \left(x_j^{(k+1)} - x_j^{(k)}\right)\cdot\left(x_j^{(k)} - x_j^{(k-1)}\right) > 0 \\ 1.0 & \text{if } = 0 \end{cases}$$

x_j monotonic $\Rightarrow$ increase σ_j



x_j oscillating $\Rightarrow$ shrink σ_j



How do we solve step 2 ?

$$\boxed{\min_{x \in T} \; g_0(x) \quad \text{subject to } g_i(x) \le 0, \; i = 1 \ldots m}$$

for : $g_i(x) = f_i(x^{(k)}) + \nabla f_i \Big|_{x^{(k)}} \cdot (x - x^{(k)})$

$$+ \frac{\rho_i}{2} \sum_{j=1}^{n} \frac{(x_j - x_j^{(k)})^2}{\sigma_j^2}$$

convex problem $\Rightarrow$ use duality (Lagrange multipliers) to simplify

with nonempty feasible set

$\Rightarrow$ Slater's

strong duality

KKT

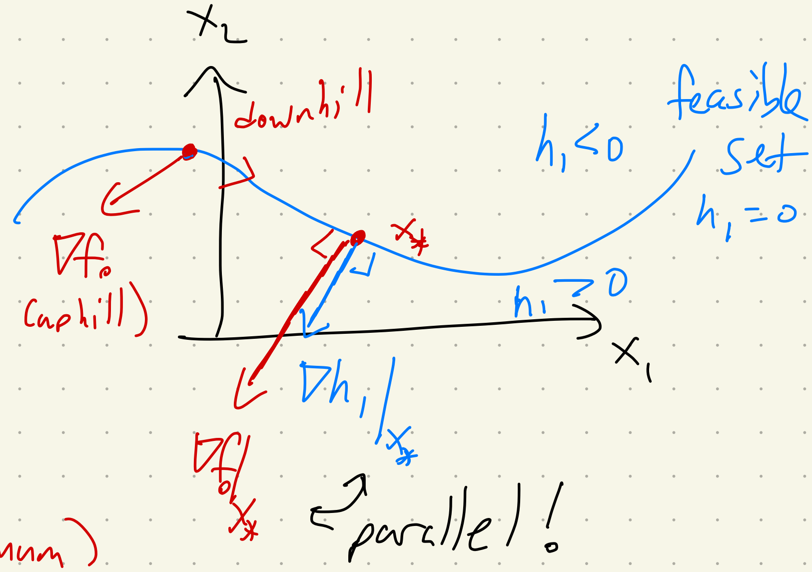
Lagrange multipliers (ala 18.02)

"primal" problem:

$$\min_{x \in \mathbb{R}^2} f_0(x)$$

$$\text{s.t.} \quad h_1(x) = 0$$

$\Rightarrow$ x_*, $f_0(x_*) = p_*$
(primal optimum)



$\Rightarrow$ unconstrained problem with extra variable ν_1 (Lagrange multiplier)

- find extremum of Lagrangian

$$L(x, \nu_1) = f_0(x) + \nu_1 h_1(x)$$

$$\Rightarrow \nabla_x L = 0 = \nabla_x f_0 + \nu_1 \nabla_x h_1 \quad \checkmark \text{ at } x_*$$

$$\nabla_\nu L = \frac{\partial L}{\partial \nu_1} = 0 = h_1(x) \quad \text{(our constraint)}$$

at x_*, ∇f_0 and ∇h_1 are parallel

$\Rightarrow$ exists <u>some</u> ν_1 such that $\nabla_x L = 0$ ☺

and ν_1 (solved from $\nabla f_0 + \nu_1 \nabla h_1 = 0$ at x_*)

tells you how "badly" f_0 "wants" to

move off the constraint

- if f_0 "wants" to move x into $h_1 > 0$ region ($h_1 > 0$ = downhill)

  $\Rightarrow \nabla f_0$ points into $h_1 < 0$

  = antiparallel to ∇h_1 $\Rightarrow \nu_1 > 0$

- if f_0 "wants" to move x into $h_1 < 0$,

  then $\nu_1 < 0$

- $\nu_1 = 0 \Rightarrow$ constraint irrelevant (still a local optimum even without constraint)

Lagrangian

primal:

standard form problem (not necessarily convex)

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & f_i(x) \leq 0, \quad i = 1, \ldots, m \\ & h_i(x) = 0, \quad i = 1, \ldots, p \end{array}$$

variable $x \in \mathbf{R}^n$, domain $\mathcal{D}$, optimal value $p^\star$ primal $= f_0(x_\star)$

Lagrangian: $L : \mathbf{R}^n \times \mathbf{R}^m \times \mathbf{R}^p \to \mathbf{R}$, with $\mathbf{dom}\, L = \mathcal{D} \times \mathbf{R}^m \times \mathbf{R}^p$,

$$L(x, \lambda, \nu) = f_0(x) + \sum_{i=1}^m \lambda_i f_i(x) + \sum_{i=1}^p \nu_i h_i(x)$$

- weighted sum of objective and constraint functions
- λ_i is Lagrange multiplier associated with $f_i(x) \leq 0$
- ν_i is Lagrange multiplier associated with $h_i(x) = 0$

Lagrange dual function

Lagrange dual function: $g : \mathbf{R}^m \times \mathbf{R}^p \to \mathbf{R}$,

$$
\begin{aligned}
g(\lambda, \nu) &= \inf_{x \in \mathcal{D}} L(x, \lambda, \nu) \\
&= \inf_{x \in \mathcal{D}} \left(f_0(x) + \sum_{i=1}^{m} \lambda_i f_i(x) + \sum_{i=1}^{p} \nu_i h_i(x) \right)
\end{aligned}
$$

g is concave, can be $-\infty$ for some λ, ν

notation: $\vec{v} \in \mathbb{R}^J$

lower bound property: if $\lambda \succeq 0$, then $g(\lambda, \nu) \leq p^\star = f_0(x_*)$

$\Leftrightarrow \lambda_i \geq 0$ for $i = 1..m$

$\vec{v} \succeq 0$

$\Leftrightarrow v_j \geq 0$ for $j = 1..J$

$\boxed{\succeq \text{ = elementwise } \geq 0}$

proof: if $\tilde{x}$ is feasible and $\lambda \succeq 0$, then

$$
f_0(\tilde{x}) \geq L(\tilde{x}, \lambda, \nu) \geq \inf_{x \in \mathcal{D}} L(x, \lambda, \nu) = g(\lambda, \nu)
$$

minimizing over all feasible $\tilde{x}$ gives $p^\star \geq g(\lambda, \nu)$

The dual problem

Lagrange dual problem

$$
\begin{array}{ll}
\underset{\lambda,\nu}{\text{maximize}} & g(\lambda, \nu) \\
\text{subject to} & \lambda \succeq 0
\end{array}
\longrightarrow \text{dual optimum}
$$

$$d^\star = g(\lambda^\star, \nu^\star)$$

- finds best lower bound on $p^\star$, obtained from Lagrange dual function
- a convex optimization problem; optimal value denoted $d^\star$
- λ, ν are dual feasible if $\lambda \succeq 0$, $(\lambda, \nu) \in \mathbf{dom}\, g$
- often simplified by making implicit constraint $(\lambda, \nu) \in \mathbf{dom}\, g$ explicit

example: standard form LP and its dual (page 5–5)

$$
\begin{array}{ll}
\text{minimize} & c^T x \\
\text{subject to} & Ax = b \\
 & x \succeq 0
\end{array}
\qquad\qquad
\begin{array}{ll}
\text{maximize} & -b^T \nu \\
\text{subject to} & A^T \nu + c \succeq 0
\end{array}
$$

Weak and strong duality

weak duality: $d^\star \leq p^\star$

- always holds (for convex and nonconvex problems)
- can be used to find nontrivial lower bounds for difficult problems

  for example, solving the SDP

$$\begin{array}{ll} \text{maximize} & -\mathbf{1}^T \nu \\ \text{subject to} & W + \mathbf{diag}(\nu) \succeq 0 \end{array}$$

  gives a lower bound for the two-way partitioning problem on page 5–7

strong duality: $d^\star = p^\star$

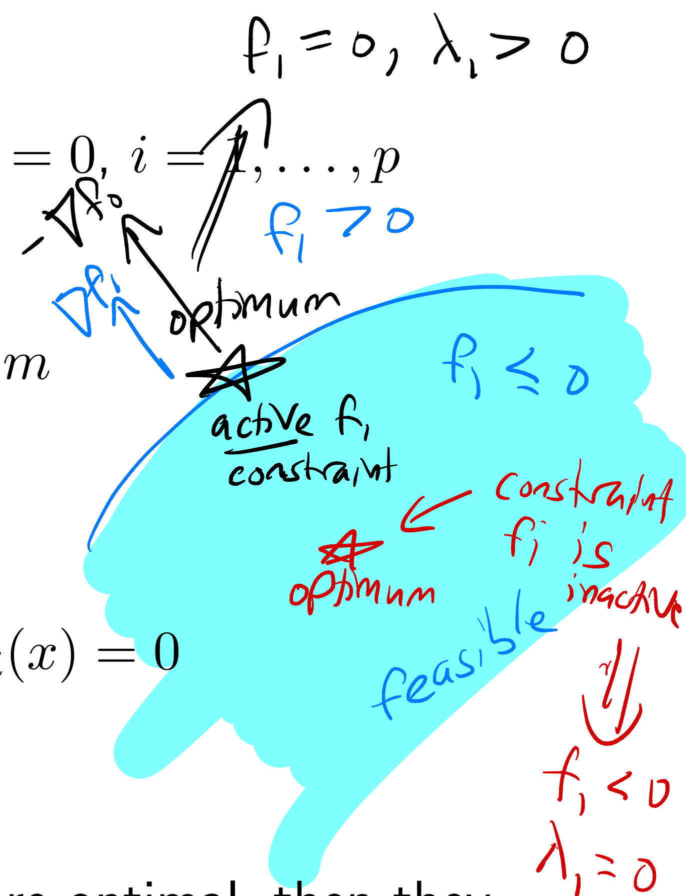
- does not hold in general
- (usually) holds for convex problems
- conditions that guarantee strong duality in convex problems are called **constraint qualifications**

Karush-Kuhn-Tucker (KKT) conditions

the following four conditions are called KKT conditions (for a problem with differentiable f_i, h_i):

1. primal constraints: $f_i(x) \leq 0$, $i = 1, \ldots, m$, $h_i(x) = 0$, $i = 1, \ldots, p$
2. dual constraints: $\lambda \succeq 0$
3. complementary slackness: $\lambda_i f_i(x) = 0$, $i = 1, \ldots, m$
4. gradient of Lagrangian with respect to x vanishes:

$$\nabla f_0(x) + \sum_{i=1}^{m} \lambda_i \nabla f_i(x) + \sum_{i=1}^{p} \nu_i \nabla h_i(x) = 0$$



from page 5–17: if strong duality holds and x, λ, ν are optimal, then they must satisfy the KKT conditions

(slide added by SGJ)

Sufficient conditions for KKT at optima:

- **Slater's condition**: for **convex** problems, strong duality holds and KKT holds at optimum if there is a *strictly feasible* point x where all $f_i(x) < 0$ (or = 0 if f_i is affine), and all $h_i(x) = 0$ are affine.

- **LICQ** (linearly independent constraint qualification): even for **nonconvex** problems, KKT must hold at any local optima if the $\{\nabla f_i, \nabla h_i\}$ are *linearly independent* for all *active* constraints.
  - **Active** constraints: $f_i(x) = 0$, $h_i(x) = 0$

    (equality constraints h_i are always active at feasible points)

weird:

violates LICQ:

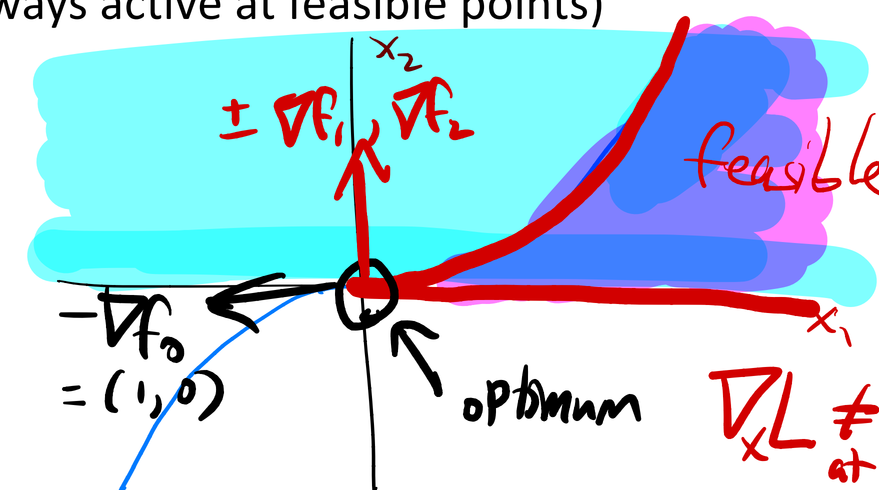
$x \in \mathbb{R}^2$

$\min \; x_1$

$\text{s.t.} \; x_2 \le x_1^3$

$x_2 \ge 0$

feasible set has "cusp"



$\nabla_x L \neq 0$ at opt!

CCSA via dual problem: ($x \in \mathbb{R}^n$, $\lambda \in \mathbb{R}^m$)

on ②:

$$L(x,\lambda) = g_0(x) + \sum_{i=1}^{m} \lambda_i g_i(x)$$

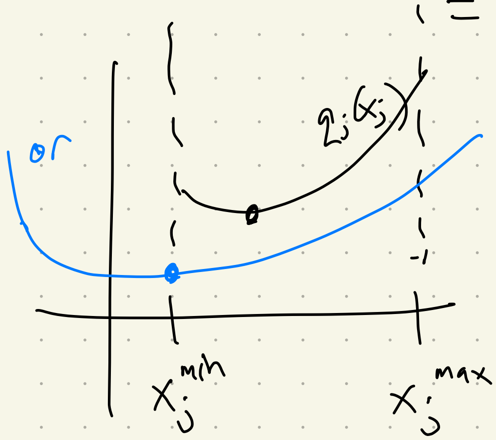
separable: $$= \sum_{j=1}^{n} \overset{\text{convex}}{\text{quadratic}_j}(x_j) + \text{constant}$$

$$\parallel$$

$$\left(\frac{\partial g_0}{\partial x_j} + \sum_{i=1}^{m} \lambda_i \frac{\partial g_i}{\partial x_j}\right)\Bigg|_{x^{(k)}} \left(x_j - x_j^{(k)}\right) + \frac{\rho_i}{2} \frac{\left(x_j - x_j^{(k)}\right)^2}{\sigma_j^2}$$

$$\Rightarrow g(\lambda) = \min_{x \in T} L(x,\lambda)$$

$$= \sum_{j=1}^{n} \boxed{\min_{\substack{x_j \in [x_j^{min}, x_j^{max}] \\ \text{of } T}} \overset{\text{convex}}{\text{quadratic}_j}(x_j) = q_j(x_j)} + \text{constant}$$



easy (high school):

solve $\frac{dq_j}{dx_j} = 0$, check bounds $\Longrightarrow x_j(\lambda)$

$\Rightarrow$ solve dual problem $\max_{\lambda \geq 0} g(\lambda)$

$\Rightarrow \lambda \in \mathbb{R}^m$

$m = \#$ constraints f_i

only bound constraints

nonlinear, convex

many possible algorithms

(e.g. Svanberg suggested an "active-set" variant of nonlinear CG)

elegant: apply CCSA <u>recursively</u>!

... on each step $\lambda^{(k)}$,

form $-g(\lambda) \approx \hat{g}_0(\lambda) = g(\lambda^{(k)}) + \nabla_\lambda g \cdot (\lambda - \lambda^{(k)}) + \frac{\hat{\rho}_0}{2} \sum_{i=1}^{m} \left(\frac{\lambda_i - \lambda_i^{(k)}}{\hat{\sigma}_i} \right)^2$

etc.

then $\min_{\lambda \in \hat{T}} \hat{g}_0(\lambda) \Rightarrow$ duality again!

no nonlinear constraints!

$\Rightarrow$ no Lagrange multipliers

$\hat{g}$ (no arguments)

$\Rightarrow$ trivial

alternatively (equivalently):

minimize $\hat{g}_0(\lambda) = \hat{L}$ analytically

$\Rightarrow$ .... recursive CCSA steps

$\lambda^{(k)} \longrightarrow$ dual optimum λ_*

strong duality: $\max_{\lambda \geq 0} g(\lambda) = \min_{\substack{x \in T \\ g_i(x) \leq 0}} g_0(x)$

$\Downarrow$

given $\lambda_* \Rightarrow x_* = x(\lambda_*)$

= candidate for $x^{(k+1)}$

Can we do better than CCSA?

- ideally, like to use <u>second</u> derivative information too!

- Hessian matrix $H_{ij} = \frac{\partial^2 f}{\partial x_i \partial x_j}$ (for f_0 etc)
  $n \times n$ (<u>big</u>!) (<u>expensive</u>!)

- amazing idea: approximate H iteratively using only ∇f
  $\Rightarrow$ BFGS updates