Convexity

- Second-order condition
- Strict & strong convexity
- Gradient descent algo

FOC

- dom f convex $\forall \vec{x}, \vec{y} \in$ dom

$$f(\vec{y}) \geq f(\vec{x}) + \nabla f(x)^T (y - z)$$

SOC

- talking about Hessian, ie: means that matrix is PSD

$$\nabla^2 f(\vec{x}) \succeq 0$$

- dom (f) convex

↳ F(x): twice differentiable fcn

eg:

$$f(\vec{x}) = \vec{x}^T Q \vec{x} + \vec{b}^T \vec{x}$$

- strong convexity ⇒ strict convexity ⇒ convexity

Strict Convexity

Recall

∀ convex fcns, dom(f) must also be convex

$\forall x, y \in$ domain

$$f(\theta \vec{x} + (1-\theta)\vec{y}) < \theta f(\vec{x}) + (1-\theta) f(\vec{y})$$ 0th order cond.

↳ $0 < \theta < 1$

$$f(\vec{y}) > f(\vec{x}) + \nabla f(x)^T (y - x)$$ 1st order condition

$$\nabla^2 f(\vec{x}) \succ 0$$ 2nd order condition

positive definite

strong convexity : "is this kinda like a quadratic?"

- f diff, dom(f) convex
- f is μ-strongly convex if:

capTures degree of strong convexity

$\forall \vec{x}, \vec{y} \in$ domain

$\vec{x} \neq \vec{y}$

$$f(\vec{y}) \geq f(\vec{x}) + \nabla f(\vec{x})^T(\vec{y}-\vec{x}) + \underbrace{\frac{\mu}{2}\|\vec{y}-\vec{x}\|^2}$$

[Recall: Taylor expansion

$$f(\vec{y}) \approx f(\vec{x}) + \nabla f^T(x)(\vec{y}-\vec{x}) + \underbrace{\frac{1}{2}(\vec{y}-\vec{x})^T \nabla^2 f(\vec{x})(\vec{y}-\vec{x})}$$]

$$\frac{1}{2}(\vec{y}-\vec{x})^T \begin{bmatrix} \mu & & 0 \\ & \mu & \\ 0 & & \ddots \; \mu \end{bmatrix} (\vec{y}-\vec{x})$$

quadratic terms ⇒ replacing Hessian with $I \cdot \mu$ in order to lower bound the Hessian at all points in the domain

$$A \succeq B \Rightarrow A - B \succeq 0$$

Gradient Descent

$F(\vec{x}) = \|A\vec{x} - \vec{b}\|_2^2$

- use for unconstrained optimization problems (like least squares), eg:

$$p^* = \min_{\vec{x} \in \mathbb{R}^n} f_0(\vec{x})$$

$$\vec{p}^* = f(\vec{x}^*)$$

- particularly useful algo when we have explicit solutions
- instead of considering $f(\vec{x})$, consider $f(\vec{x} + \Delta\vec{x})$

(Taylor exp, 1st order approx.)

$$f(\vec{x} + \Delta\vec{x}) = f(\vec{x}) + \nabla f(\vec{x})^T \Delta\vec{x}$$

$$f(\vec{x} + s\vec{v}) = f(\vec{x}) + \nabla f(\vec{x})^T \vec{v}$$

↳ step size

$$= f(\vec{x}) + \langle \nabla f(\vec{x}), \vec{v} \rangle$$

} want to decrease fcn value
↳ what direction should we choose to minimize?

Recall: to maximize this value, choose $\vec{v}$ aligned to gradient; but want to minimize this by choosing $\vec{v} = -\nabla f(\vec{x})$

$\langle \nabla f(\vec{x}), \vec{v} \rangle \leq \|\nabla f(\vec{x})\|_2 \|\vec{v}\|_2$

↳ Cauchy-Schwartz

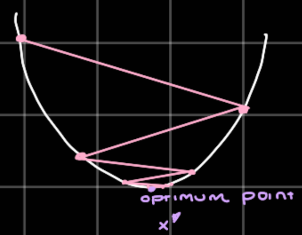
⇒ to minimize inner prod: $\vec{v} = -\nabla f(\vec{x})$

Note: we aren't normalizing out the $\vec{v}$ bc when we're close to the optimum, we don't want to move too much (bc ∇ is also small at the optimum)

Gradient Descent Algorithm

$$\vec{x}_{k+1} = \vec{x}_k - \eta \nabla F(\vec{x}_k) \quad , \quad \vec{x}_0 : \text{initial point}, \quad \eta : \text{step size}$$

↳ want to show that this converges to a value $\vec{x}^*$



Note:
if you reach $\vec{x}_*$ at k
(ie $\vec{x}_k = \vec{x}_*$), then $\vec{x}_{k+1} = \vec{x}_k - \eta \cancel{\nabla F(\vec{x}_k)}^{\,0}$
so it converges to $\vec{x}_{k+n} = \vec{x}_*$!

Least squares

$$F(\vec{x}) = \|A\vec{x} - \vec{b}\|_2^2$$

$$\nabla F(\vec{x}) = 2A^T(A\vec{x} - \vec{b})$$

Recall L-S solution: $\underbrace{(A^TA)^{-1}}_{\text{taking a matrix inv. is } O(n^3)} A^T\vec{b}$

↳ better to do gradient descent bc it's $O(n^2)$

$$\begin{aligned}
\vec{x}_{k+1} &= \vec{x}_k - \eta \nabla F(\vec{x}_k) \\
&= \vec{x}_k - \eta\, 2A^T(A\vec{x} - \vec{b}) \\
&= \underbrace{(I - 2\eta A^TA)}_{\text{symmetric}}\vec{x}_k - 2\eta A^T\vec{b}
\end{aligned}$$

symmetric (bc I & A^TA are symm) → can use spectral thm to do eigenval decomp

To see how good our soln is / to prove convergence:

$$\begin{aligned}
\vec{x}_{k+1} - \vec{x}^* &= \vec{x}_{k+1} - (A^TA)^{-1}A^T\vec{b} \\
&= (I - 2\eta A^TA)\vec{x}_k + 2\eta A^T\vec{b} - (A^TA)^{-1}A^T\vec{b} \\
&= (I - 2\eta A^TA)\vec{x}_k + 2\eta (A^TA)(A^TA)^{-1}A^T\vec{b} - (A^TA)^{-1}A^T\vec{b} \\
&= \underbrace{(I - 2\eta A^TA)\vec{x}_k}\; + (2\eta A^TA - I)(A^TA)^{-1}A^T\vec{b} \\
&= \underbrace{(I - 2\eta A^TA)}(\vec{x}_k - (A^TA)^{-1}A^T\vec{b})
\end{aligned}$$

if the eigenvalues of this are < 1, then Gradient Descent converges for least-squares