Lecture 31

"ACCELERATED"

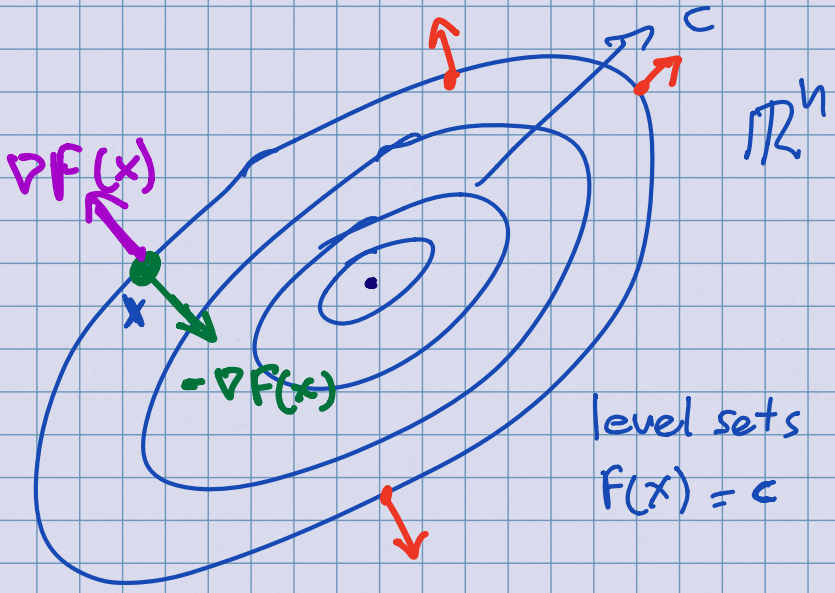
GRADIENT DESCENT

- MINIPROJECT (DUE TUESDAY 12/1)
- NEW PSET (FRIDAY)
- THANKSGIVING BREAK!

GRADIENT DESCENT

$$x_{k+1} = x_k - \gamma \nabla f(x_k)$$

- x_{k+1}: next point
- x_k: current point
- γ: stepsize
- $\nabla f(x_k)$: gradient at current point



(negative)

GRADIENT IS ORTHOGONAL TO LEVEL SETS

(steepest descent direction)

Convergence of gradient method.

(quadratic functions)

$$F(x) := \frac{1}{2} x^T A x$$

λ_i eigenvalues of A

Q : condition number

$$\lambda_1 / \lambda_n \geq 1$$

STRONGLY CONVEX

$$0 < \lambda_n \leq \cdots \leq \lambda_1$$

converges if $0 < \gamma < \frac{2}{\lambda_1}$

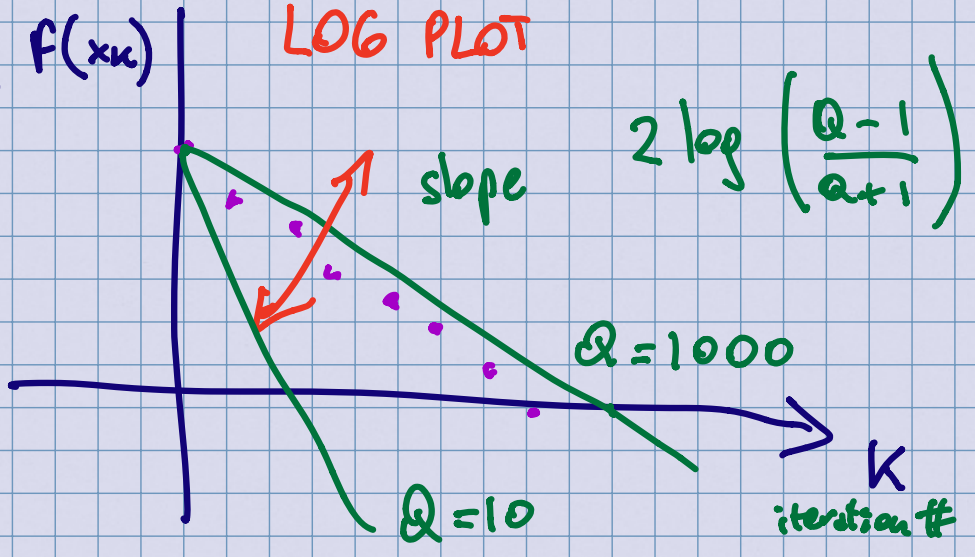
stepsize $\gamma = \frac{2}{\lambda_1 + \lambda_n}$

$$\Downarrow$$

$$F(x_k) \leq C \left(\frac{Q-1}{Q+1} \right)^{2k}$$

(fast)

$$\log F(x_k) \leq \underbrace{\log C + 2k \log\left(\frac{Q-1}{Q+1}\right)}_{\text{linear function of } k}$$



(actually $\log\left(F(x_k) - F(x_*)\right)$ since we assumed $F(x_*) = 0$)

SOUNDS PRETTY GOOD !!

BUT, CAN WE DO BETTER ?

(... without paying more !)

- FOR INSTANCE, IF WE ALLOW SECOND DERIVATIVES (i.e., Hessian) CAN DERIVE POTENTIALLY MUCH BETTER (BUT MORE EXPENSIVE) METHODS.

EXAMPLE: QUADRATIC FUNCTIONS !

$$f(x) = \frac{1}{2} x^T A x - b^T x$$

A psd

"better, but expensive"

$\nabla f = 0$

$\min_x f(x)$

"simple, cheap"

linear algebra

$$Ax = b$$

- more complicated (e.g. Gaussian elimination)
- fixed # iterations (typically $O(n^3)$)

$$x_{k+1} = x_k - \gamma \nabla f(x_k)$$

- Easy to implement
- may take many iterations (condition number)

(aside...)

NEWTON'S METHOD

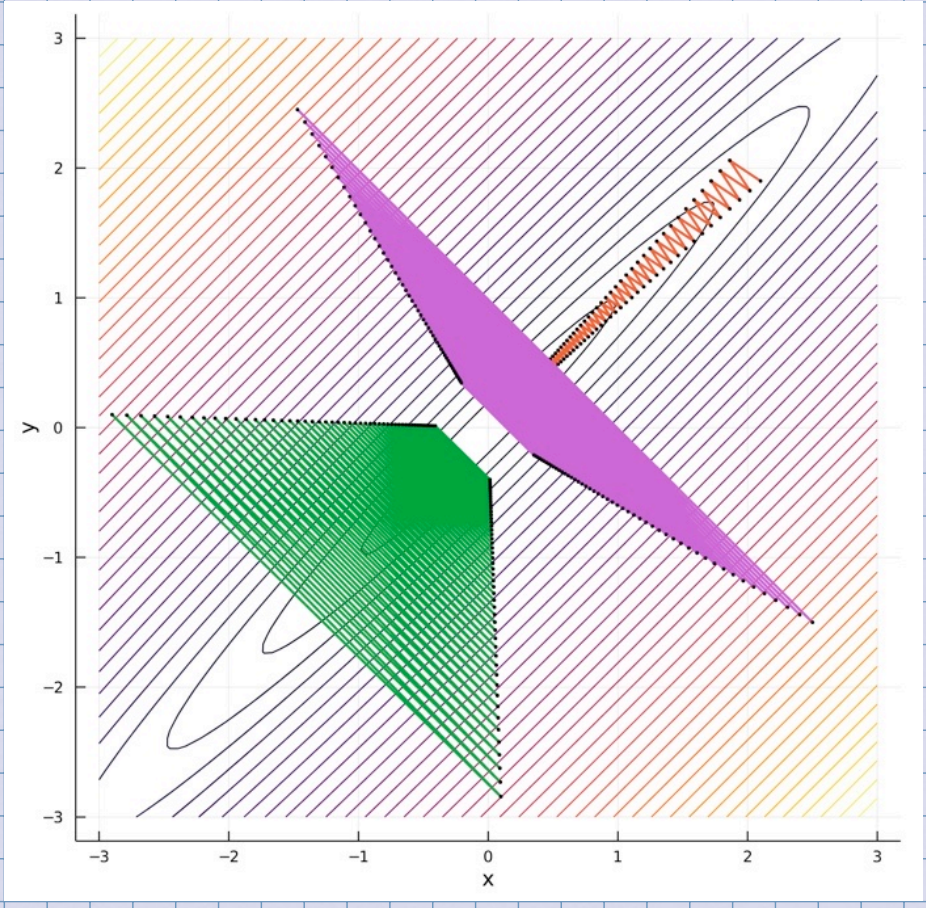
$$x_{k+1} = x_k - \underbrace{\left[\nabla^2 f(x_k)\right]^{-1}}_{\text{inverse of Hessian}} \underbrace{\nabla f(x_k)}_{\text{gradient}}$$

ALL KIND OF WONDERFUL PROPERTIES...

(but not in this course ☹)

WANT TO DO BETTER THAN "PLAIN" GRADIENT, BUT WITHOUT TOO MUCH ADDITIONAL EXPENSE.

Why is the gradient method slow sometimes?

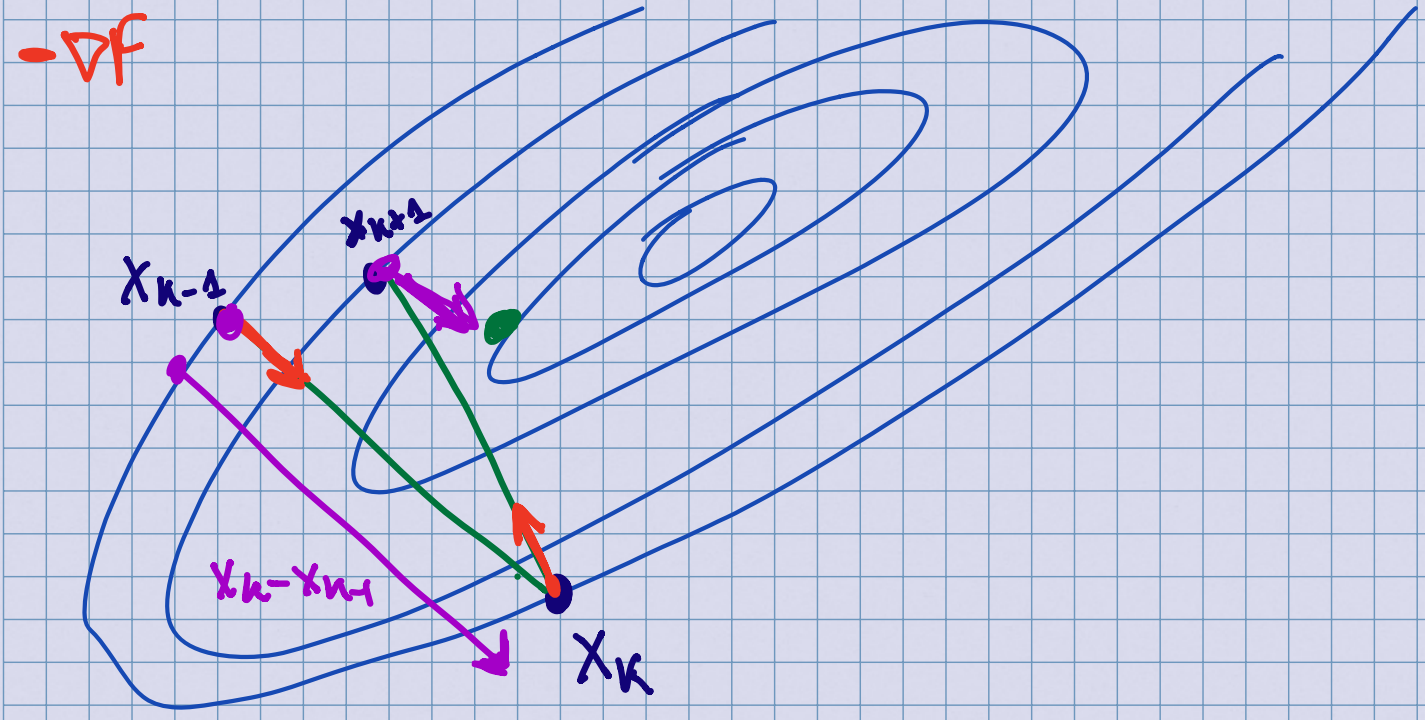


ZIG-ZAG BEHAVIOR

(worse with large condition numbers)

Any way around it?

Momentum (or "heavy ball") idea



$$x_{k+1} = \underbrace{x_k - \alpha\, Df(x_k)}_{\text{standard gradient method}} + \underbrace{\beta(x_k - x_{k-1})}_{\text{momentum term}}$$

($\beta = 0$ is GD)

Great! How to analyze it?

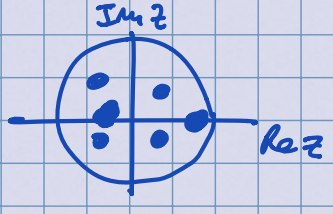
In the quadratic case

$$f(x) = \frac{1}{2} x^T A x \qquad \nabla f(x) = Ax$$

it's again a linear system!

$$x_{k+1} = \underbrace{x_k - \alpha \nabla f(x_k)}_{\text{standard gradient method}} + \underbrace{\beta (x_k - x_{k-1})}_{\text{momentum term}}$$

$$\begin{bmatrix} x_{k+1} \\ x_k \end{bmatrix} = \underbrace{\begin{bmatrix} (1+\beta)I - \alpha A & -\beta I \\ I & 0 \end{bmatrix}}_{\text{want eigenvalues of this to be inside a small disk in } \mathbb{C}} \begin{bmatrix} x_k \\ x_{k-1} \end{bmatrix}$$



- Can diagonalize (why?), so it is enough to consider the 2×2 matrix

$$M_i = \begin{bmatrix} 1+\beta-\alpha\lambda_i & -\beta \\ 1 & 0 \end{bmatrix}$$

λ_i are the eigenvalues of A. $\lambda_1 \dots \lambda_n$

- Given $\lambda_1, \dots \lambda_n$, find the best α, β, i.e., such that the eigenvalues of M_i are contained in the smallest possible disk.

Can show (some algebra)

that the best α, β are

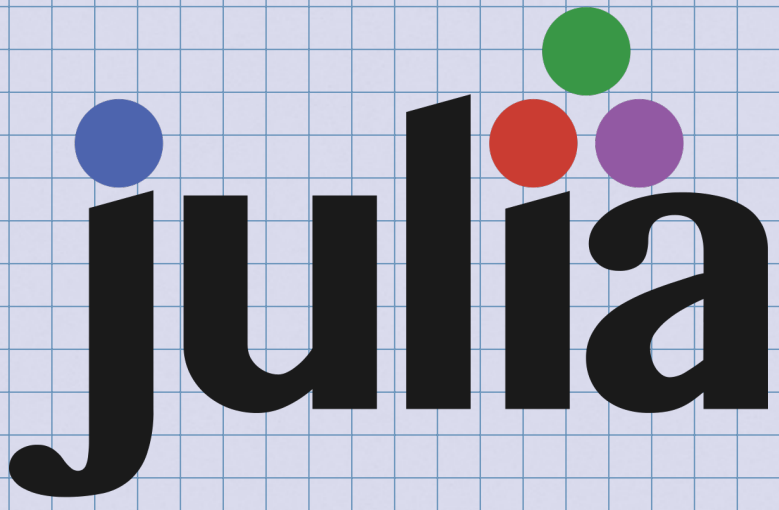
$$\alpha = \frac{4}{\left(\sqrt{\lambda_1} + \sqrt{\lambda_n}\right)^2} \qquad \beta = \left(\frac{\sqrt{\lambda_1} - \sqrt{\lambda_n}}{\sqrt{\lambda_1} + \sqrt{\lambda_n}}\right)^2$$

giving a convergence rate of:

$$\rho = \frac{\sqrt{\lambda_1} - \sqrt{\lambda_n}}{\sqrt{\lambda_1} + \sqrt{\lambda_n}} = \boxed{\frac{\sqrt{Q} - 1}{\sqrt{Q} + 1}}$$

SIMILAR TO THE EXPRESSION FOR GRADIENT DESCENT, BUT WITH Q REPLACED BY $\sqrt{Q}$

go to



THIS IS LOT BETTER!

FOR LARGE R, $\frac{R-1}{R+1} \sim 1 - \frac{2}{R}$

$\Rightarrow$

RATE

$$GD \approx 1 - \frac{2}{Q}$$

$$AGD \approx 1 - \frac{2}{\sqrt{Q}}$$

function value (or distance to optimum)

decreases by this multiplicative factor

at every iteration

GD $1 - \frac{2}{Q}$ $>$ $1 - \frac{2}{\sqrt{Q}}$ AGD

$$x_{k+1} = \underbrace{x_k - \alpha \nabla f(x_k)}_{\text{standard gradient method}} + \underbrace{\beta (x_k - x_{k-1})}_{\text{momentum term}}$$

GRADIENT	ACCELERATED GRADIENT
$\beta = 0$	$\beta \neq 0$
rate $\frac{Q-1}{Q+1}$	rate $\frac{\sqrt{Q}-1}{\sqrt{Q}+1}$
# iters $O\left(Q \log \frac{1}{\epsilon}\right)$	$O\left(\sqrt{Q} \log \frac{1}{\epsilon}\right)$

roughly, # iterations reduced by $\sqrt{Q}$ factor (for same error)

Some comments

- Marginally more expensive, but essentially the same cost
- Can be a lot faster, in some problems
- More sensitive to noise, and nonlinearities
- In practice, just add "a little bit" of momentum to GD.
- A variation (Nesterov's AGD) also works for convex functions (not necessarily quadratic)