

# ECE 532 - lecture 17 - iterative methods

1

\* iterative methods are a computational technique for solving optimization problems. They are essentially the only way to solve problems that are very large like the kinds solved in practice.

Simple example: minimize  $\|y - Ax\|^2$  (least squares).

suppose  $A$  has rank  $n$  (full column rank), then the

normal equations have solution  $\hat{x} = \underbrace{(A^T A)^{-1}}_{n \times n} \underbrace{(A^T b)}_{n \times 1}$ .

computing  $(A^T A)^{-1}$  requires inverting an  $n \times n$  matrix, which takes  $O(n^3)$  operations. If  $n$  is large, we may need to wait awhile... Also, it's an "all or nothing" proposition:

- 1) wait a while and get the exact solution
- 2) or give up and get nothing.

Iterative methods provide an intermediate alternative, a.k.a. "iterative solvers" or "iterative algorithms".

2

basic form :  $x_{k+1} = f(x_k)$ .

Note:  $x_k$  is the  $k^{\text{th}}$  iterate, NOT the  $k^{\text{th}}$  component of  $x$ !

i.e.  $x_k \in \mathbb{R}^n$  for all  $n, k$  and  $f: \mathbb{R}^n \rightarrow \mathbb{R}^n$ . The goal is to

design  $f$  so that:

compared to solving the optimization problem!

1) evaluating  $f(\cdot)$  has low cost, such as  $O(n)$  or  $O(n^2)$ .

2) the iterates converge to the true solution:  $\lim_{k \rightarrow \infty} x_k = A^+ y$ .

★ Advantage to such a method is that we can stop once we're close enough. The longer we compute, the more accurate the solution.

→ useful in ML applications, where it's often good enough to be "close".

→ used all the time e.g. when you need to calculate  $\sqrt{2}$ .

→ often a direct method doesn't exist or is too costly.

★ Potential disadvantages:

→ may take a long time to converge.

→ may not know how close you are to the solution.

→ requires tuning, this can be difficult.

## Landweber iteration

for solving  $\min_x \|Ax - y\|^2$ ,

the Landweber iteration is the iterative method:

$$x_{k+1} = x_k - \tau A^T (Ax_k - y).$$

where  $\tau > 0$  is a parameter that must be chosen. We also must choose  $x_0$  (initial iterate), but that can be anything, or we can use a "best guess" as well.

★ if it converges, then what must it converge to?

if  $x_k \rightarrow x_*$  then when  $x_k$  is very close to  $x_*$ ,

$$x_* = x_* - \tau A^T (Ax_* - y)$$

$$\Rightarrow A^T A x_* = A^T y.$$

so  $x_*$  satisfies the normal equations!

★ how do we prove convergence? We'll see two methods.

First, define  $r_k = Ax_k - y$ .

let's see what happens to  $r_k$  as  $k$  gets large.

method 1 : linear analysis (only works when  $f(\cdot)$  is affine).

$$x_{k+1} = x_k - \tau A^T (Ax_k - y).$$

$$\Rightarrow Ax_{k+1} - y = Ax_k - y - \tau AA^T (Ax_k - y).$$

$$\Rightarrow r_{k+1} = r_k - \tau AA^T r_k.$$

$$\Rightarrow \boxed{r_{k+1} = (I - \tau AA^T) r_k.}$$

we can also look at the normal residual:  $A^T r_k = A^T A x_k - A^T y$

$$\Rightarrow \boxed{(A^T r_{k+1}) = (I - \tau A^T A) (A^T r_k)}$$

if  $A$  tall + full rank,  $A = U \begin{bmatrix} \Sigma \\ 0 \end{bmatrix} V^T$ ,  $I - \tau A^T A = V (I - \tau \Sigma^T \Sigma) V^T$ .

$$\begin{aligned} \Rightarrow (A^T r_{k+1}) &= (I - \tau A^T A)^k (A^T r_0) = \\ &= \underbrace{[V(I - \tau \Sigma^T \Sigma) V^T] [V(I - \tau \Sigma^T \Sigma) V^T] \dots}_{k \text{ times}} (A^T r_0). \end{aligned}$$

$$= \underbrace{V(I - \tau \Sigma^T \Sigma)^k V^T}_{\text{matrix}} (A^T r_0)$$

$$\rightarrow \begin{bmatrix} (1 - \tau \sigma_1^2)^k & & & 0 \\ & \ddots & & \\ & & \ddots & \\ 0 & & & (1 - \tau \sigma_n^2)^k \end{bmatrix}$$

if we let  $z_k = V^T A^T r_k$ , then

(5)

$$z_k = \begin{bmatrix} (1-\tau\sigma_1^2)^k \\ \vdots \\ (1-\tau\sigma_n^2)^k \end{bmatrix} z_0.$$

converges if  $|1-\tau\sigma_i^2| < 1$  for all  $i$ .

$$\Rightarrow -1 < 1-\tau\sigma_i^2 < 1 \text{ for all } i.$$

$$\Rightarrow 0 < \tau\sigma_i^2 < 2 \text{ for all } i$$

$$\Rightarrow 0 < \tau\sigma_1^2 < 2$$

sufficient to ensure  
it for the largest  $\sigma_i$ .

$$\Rightarrow 0 < \tau < \frac{2}{\sigma_1^2}$$

$$\text{i.e. } 0 < \tau < \frac{2}{\|A\|^2}$$

in this case, we have  $z_k \rightarrow 0$ , i.e.  $A^T r_k \rightarrow 0$ .

Note:  $r_k \neq 0$  since for  $r_{k+1} = \underbrace{(I - \tau A A^T)}_{\downarrow}$   $r_k$ ,

$$V \begin{bmatrix} (1-\tau\sigma_1^2) \\ \vdots \\ (1-\tau\sigma_n^2) \\ \vdots \\ 0 \end{bmatrix} V^T$$

← zeros!

so  $0 < \tau\sigma_i^2 < 2$  does not hold for  $i > n$ .

method 2: bound analysis (ad-hoc method).

(6)

This time, expand the norm of the residual.

$$\begin{aligned}\|Ax_{k+1} - y\|^2 &= \|A(x_k - \tau A^T(Ax_k - y)) - y\|^2 \\ &= \|Ax_k - y - \tau AA^T(Ax_k - y)\|^2 \\ &= \|Ax_k - y\|^2 + \tau^2 \|AA^T(Ax_k - y)\|^2 - 2\tau (Ax_k - y)^T AA^T(Ax_k - y)\end{aligned}$$

(we use the fact that  $\|u+v\|^2 = (u+v)^T(u+v) = u^T u + 2u^T v + v^T v = \|u\|^2 + \|v\|^2 + 2u^T v$ )

$$= \|Ax_k - y\|^2 + \tau \left[ \tau \|AA^T(Ax_k - y)\|^2 - 2 \|A^T(Ax_k - y)\|^2 \right]$$

$$\leq \|Ax_k - y\|^2 + \tau \left[ \tau \|A\|^2 \|A^T(Ax_k - y)\|^2 - 2 \|A^T(Ax_k - y)\|^2 \right]$$

(we used the fact that  $\|Ax\| \leq \|A\| \cdot \|x\|$ .)

$$= \|Ax_k - y\|^2 + \tau \|A^T(Ax_k - y)\|^2 (\tau \|A\|^2 - 2).$$

so we'll have  $\|Ax_{k+1} - y\|^2 \leq \|Ax_k - y\|^2$  as long as  $0 < \tau < \frac{2}{\|A\|^2}$

(same result as before!)

7

Note: the proof above doesn't actually show that

$x_k \rightarrow A^+ y$ , but this can be done by taking

the SVD of  $A$  and using a similar bounding process.

[as long as  $x_0 \in N(A)^\perp$ , e.g.  $x_0 = 0$ ].

We can also directly expand  $\|x_{k+1} - A^+ y\|^2$  and express in terms of  $\|x_k - A^+ y\|$ .

let  $A = U_1 \Sigma_1 V_1^T$  (from SVD). let  $e_k := x_k - A^+ y = x_k - V_1 \Sigma_1^{-1} U_1^T y$ .

$$\begin{aligned}
e_{k+1} &= x_{k+1} - V_1 \Sigma_1^{-1} U_1^T y \\
&= x_k - \tau (V_1 \Sigma_1^2 V_1^T x_k - V_1 \Sigma_1 U_1^T y) - V_1 \Sigma_1^{-1} U_1^T y \\
&= e_k - \tau V_1 \Sigma_1^2 V_1^T (x_k - V_1 \Sigma_1^{-1} U_1^T y) \\
&= (I - \tau V_1 \Sigma_1^2 V_1^T) e_k.
\end{aligned}$$

$$\Rightarrow \text{split into components: } e_k = \underbrace{V_1 (V_1^T e_k)}_{\in N(A)^\perp} + \underbrace{V_2 (V_2^T e_k)}_{\in N(A)}.$$

$$\begin{cases}
(V_1^T e_{k+1}) = (I - \tau \Sigma_1^2) (V_1^T e_k). & \text{so component of } e_k \text{ along } N(A)^\perp \text{ goes to zero.} \\
(V_2^T e_{k+1}) = (V_2^T e_k). & \text{and component of } e_k \text{ along } N(A) \text{ is unchanged.}
\end{cases}$$

if we choose  $e_0 \in N(A)^\perp$ , then  $e_k \rightarrow 0$ . Note:  $e_0 = x_0 - \underbrace{V_1 \Sigma_1^{-1} U_1^T y}_{\in N(A)^\perp}$ .

an easy way to ensure this is  $x_0 = 0$ .