Today: Conjugate Gradient

Goal: Let $A^T = A \in \mathbb{R}^{n\times n}$, solve $Ax = b$

A positive definite, i.e. $x^TAx > 0$, $\forall x \neq 0$

Idea: Turn $Ax=b$ into a minimization problem

Since A is positive definite, $\|x\|_A = \sqrt{x^TAx}$

Let $x_* \in \mathbb{R}^n$ be exact solution to $Ax = b$

x_* solves $Ax=b \iff x^* = \underset{x\in\mathbb{R}^n}{\arg\min}\ \|x - x_*\|_A^2$

$$\begin{aligned}\|x-x_*\|_A^2 &= (x-x_*)^TA(x-x_*) \\ &= x^TAx - 2(Ax_*)^Tx + \|x_*\|_A^2\end{aligned}$$

$\iff x^* = \underset{x\in\mathbb{R}^n}{\arg\min}\ (x^TAx - 2b^Tx)$

Let $f(x) = \frac{1}{2}x^TAx - b^Tx \in \mathbb{R}$.

want to find $x_* = \underset{x\in\mathbb{R}^d}{\arg\min}\ f(x) \longleftarrow$ optimization algorithms

---

• Method 1: Steepest gradient descent

Given $x_k \in \mathbb{R}^n$, try to find $x_{k+1} \in \mathbb{R}^n$

such that $x_{k+1} = x_k - \alpha_k \nabla f(x_k) \longleftarrow$ downhill direction

want $f(x_{k+1}) \leq f(x_k)$

so choose $\alpha_k = \underset{\alpha\in\mathbb{R}}{\arg\min}\ f(x_k - \alpha\underbrace{\nabla f(x_k)}_{-d_k}) \longleftarrow$ line search

Note that $\nabla f(x) = Ax - b$

$$\begin{aligned}f(x+\alpha y) &= \tfrac{1}{2}(x+\alpha y)^TA(x+\alpha y) - b^T(x+\alpha y) \\ &= \tfrac{1}{2}x^TAx - b^Tx + \alpha y^T(Ax-b) + \tfrac{\alpha^2}{2}y^TAy\end{aligned}$$

$$= f(x) + \alpha\, y^T(Ax-b) + \frac{\alpha^2}{2} y^T A y$$

$$d_k = -\nabla f(x_k) = b - Ax_k = r_k$$

$$f(x_k - \alpha d_k) = f(x_k) + \alpha\, d_k^T r_k + \frac{\alpha^2}{2} d_k^T A d_k$$

$$\Longrightarrow \quad \boxed{\alpha_k = \frac{d_k^T r_k}{d_k^T A d_k}} = \frac{r_k^T r_k}{r_k^T A r_k}$$

Note also that $r_k = b - A(x_{k-1} + \alpha_{k-1} d_{k-1})$

$$= r_{k-1} - \alpha_{k-1} A d_{k-1} = r_{k-1} - \alpha_{k-1} \underbrace{A r_{k-1}}_{w_{k-1}}$$

- Algorithm: Given $x_0 \in \mathbb{R}^n$, $r_0 = b - Ax_0 = d_0 \in \mathbb{R}^n$

For $k = 1, 2, 3, \cdots$

$$w_{k-1} = A d_{k-1}$$

$$\alpha_{k-1} = \frac{d_{k-1}^T r_{k-1}}{d_{k-1}^T w_{k-1}}$$

$$x_k = x_{k-1} + \alpha_{k-1} d_{k-1}$$

$$r_k = r_{k-1} - \alpha_{k-1} w_{k-1}$$

$$d_{k-1} = r_{k-1}$$

end

$\overbrace{x_{k-1}, r_{k-1}, d_{k-1}, w_{k-1}, \alpha_{k-1}}^{4n+1}$

- Cost: $O(n)$ memory

$O(n)$ flops

+ compute one Ad per iteration

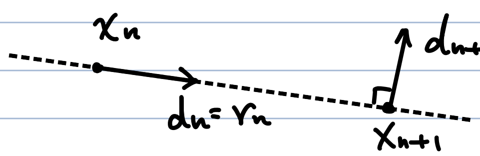
search direction has no memory

- Convergence of Steepest GD

1) Two consecutive search directions are orthogonal

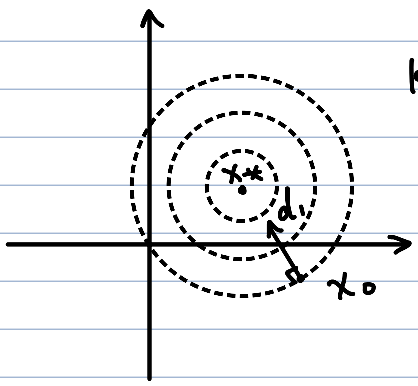
From $r_k = r_{k-1} - \alpha_{k-1} A r_{k-1}$

$$\xrightarrow{r_{k-1}^T \cdot} \quad r_{k-1}^T r_k = r_{k-1}^T r_{k-1} - \alpha_{k-1} r_{k-1}^T A r_{k-1} = 0$$



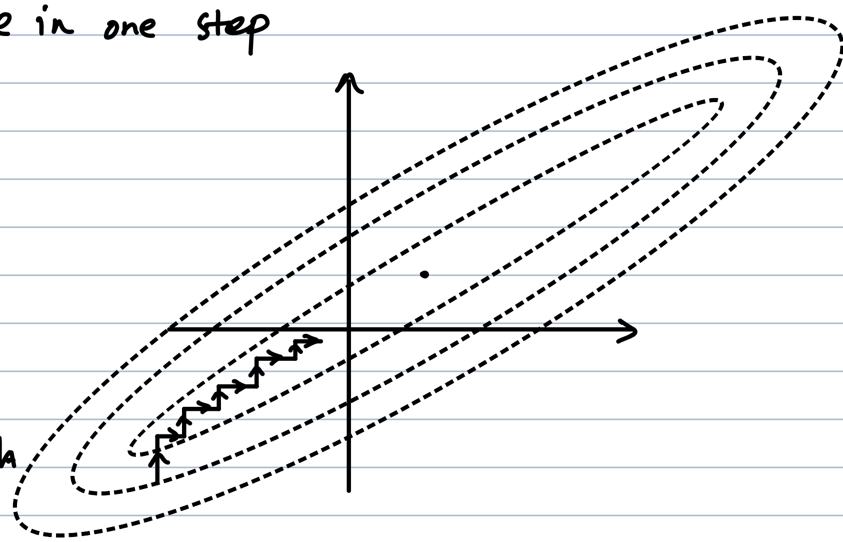
2) $$\|x_k - x_*\|_A \le \left(\frac{\kappa_2(A)-1}{\kappa_2(A)+1}\right)^k \|x_0 - x_*\|_A$$

Slow convergence if $\kappa_2(A) \gg 1$



$\kappa_2(A) = 1$

converge in one step



$\kappa_2(A) \gg 1$

$$\|x_k - x_*\|_A \le \left(1 - \frac{2}{\kappa_2(A)+1}\right)^k \|x_0 - x_*\|_A \le \varepsilon$$

$$\Rightarrow \quad k \ge O\left(\frac{\log(\varepsilon)}{\log\left(1-\frac{2}{\kappa_2(A)+1}\right)}\right) \approx O\left(\kappa_2(A)\log\left(\frac{1}{\varepsilon}\right)\right)$$

Pf: $$\|x_k - x_*\|_A^2 = (x_k - x_*)^T A (x_k - x_*)$$
$$= f(x_k) + x_*^T A x_*$$
$(\forall \alpha \in \mathbb{R})$ $$\le f(x_{k-1} + \alpha r_{k-1}) + x_*^T A x_*$$
$$= (x_{k-1} + \alpha r_{k-1} - x_*)^T A (x_{k-1} + \alpha r_{k-1} - x_*)^T$$
$$= \|(I - \alpha A)(x_{k-1} - x_*)\|_A$$

A eig. values $0 < \lambda_1 \le \lambda_2 \le \cdots \le \lambda_n$

$$\le \max_{1\le i\le n} |1 - \alpha\lambda_i| \, \|x_{k-1} - x_*\|_A$$

Note that $$\min_{\alpha\in\mathbb{R}} \max_{1\le i\le n} |1 - \alpha\lambda_i| = \frac{\lambda_n - \lambda_1}{\lambda_n + \lambda_1}$$

$$\Rightarrow \quad \|x_k - x_*\|_A \le \frac{\lambda_n - \lambda_1}{\lambda_n + \lambda_1} \|x_{k-1} - x_*\|_A \le \left(\frac{\lambda_n - \lambda_1}{\lambda_n + \lambda_1}\right)^k \|x_0 - x_*\|_A \quad \blacksquare$$

Method 2: Conjugate Gradient (Krylov subspace method)

Each step, we still do one-dimensional search

$$x_k = x_{k-1} + \alpha_{k-1} d_{k-1}$$

Choose $\alpha_{k-1} = \underset{\alpha \in \mathbb{R}}{\operatorname{argmin}} f(x_{k-1} + \alpha d_{k-1})$

$$\Rightarrow \boxed{\alpha_{k-1} = \frac{d_{k-1}^T r_{k-1}}{d_{k-1}^T A d_{k-1}}}$$

$$\Rightarrow \boxed{r_k = r_{k-1} - \alpha_{k-1} A d_{k-1}} \Rightarrow d_{k-1}^T r_k = 0 \longleftarrow \text{search direction}$$

- How to choose "best" d_k ?

– Because we do 1D search each step, after k steps,

$$x_k = x_0 + \sum_{i=0}^{k-1} \alpha_i d_i \quad , \quad \alpha_i \in \mathbb{R}$$

– Hope to choose $d_0, d_1, \dots, d_{k-1}$

s.t. 1) $\operatorname{span}\{d_0, \dots, d_{k-1}\} = \mathcal{K}_k(A, r_0) =: \mathcal{K}_k$

2) $f(x_k) = \min\limits_{x \in x_0 + \mathcal{K}_k} f(x)$

– Since x_k is optimal, we have

$$\partial_{\alpha_j} f(x_0 + \sum_{i=0}^{k-1} \alpha_i d_i)$$

$$= \partial_{\alpha_j} \left[f(x_0) + \left(\sum_{i=0}^{k-1} \alpha_i d_i\right)^T (Ax_0 - b) + \frac{1}{2} \left(\sum_{i=0}^{k-1} \alpha_i d_i\right)^T A \left(\sum_{i=0}^{k-1} \alpha_i d_i\right) \right]$$

$$= d_j^T (Ax_0 - b) + d_j^T A \left(\sum_{i=0}^{k-1} \alpha_i d_i\right)$$

$$= d_j^T(Ax_k - b) = -d_j^T r_k \longleftarrow$$ when $\partial_{\alpha_j} f(x_0 + \sum_{i=0}^{k-1} \alpha_i d_i) = 0$,

$\Rightarrow \boxed{d_j^T r_k = 0, \ \forall j = 0, \dots, k-1}$ (★)

$$= d_j^T(Ax_{j+1} - b) - d_j^T A \sum_{i=j+1}^{k-1} \alpha_i d_i$$

$$= \underbrace{-d_j^T r_{j+1}}_{=0} - \underbrace{d_j^T \sum_{i=j+1}^{k-1} \alpha_i A d_i}_{\sum_{i=j+1}^{k-1} \alpha_i d_j^T A d_i}$$

α_i's are fixed as long as d_i's are chosen

so want to choose d_i s.t. $d_j^T A d_i = 0 \quad j = 0, \dots, i-1$

$d_j \perp_A d_i$

<u>Idea:</u> As we generate the residual, use Gram-Schmidt to generate search directions that are A-conjugate.

- At the k^{th} step, we have compute $r_k, d_0, \dots, d_{k-1}$

then $d_k = r_k - \sum_{i=0}^{k-1} d_i \frac{d_i^T A r_k}{d_i^T A d_i} \longleftarrow$ need to store all d_i's not a great idea!

- We can simplify it a lot:

Actually, we have the following properties:

1) $d_j \perp r_k, \quad j = 0, \dots, k-1$

2) $r_i \perp r_k, \quad i = 0, \dots, k-1$

3) $d_i \perp_A r_k, \quad i = 0, \dots, k-2,$

Proof: 1) Follows from the optimality of α_{k-1} (★)

2) From (★), $d_i^T r_k = 0, \quad i = 0, 1, \dots, k-1$

$$\Rightarrow \left(r_i - \sum_{j=0}^{i-1} d_j \frac{d_j^T A r_i}{d_j^T A d_j}\right)^T r_k = 0 \Rightarrow \boxed{r_i^T r_k = 0}, \ i = 0, \dots, k-1$$

(★★)

3) Since $r_{i+1} = r_i - \alpha_i A d_i \Rightarrow A d_i = \dfrac{r_i - r_{i+1}}{\alpha_i}$

So $d_i^T A r_k = \left(\dfrac{r_i - r_{i+1}}{\alpha_i}\right)^T r_k$ $(*)$, $i = 0, \cdots, k-1$

$\overset{2)}{\Rightarrow}$ for $i = 0, \cdots, k-2$, $d_i^T A r_k = 0$. $\longleftarrow$ $d_i \perp_A r_k$, $i = 0, \cdots, k-2$

$\square$

- From 3),

$$d_k \overset{3)}{=} r_k - d_{k-1} \underbrace{\boxed{\frac{d_{k-1}^T A r_k}{d_{k-1}^T A d_{k-1}}}}_{=: \beta_k}$$

β_k can also be computed as follows (Just to be consistent with the usual form of CG)

$$\beta_k = \left(\frac{r_{k-1} - r_k}{\alpha_{k-1}}\right)^T r_k \Big/ (d_{k-1}^T A d_{k-1})$$

$$\overset{2) + \alpha_{k-1}}{=} \frac{r_k^T r_k}{d_{k-1}^T r_{k-1}}$$

$$\overset{\text{def of } d_{k-1} + 1)}{=} \frac{r_k^T r_k}{r_{k-1}^T r_{k-1}}$$

- Algorithm : Conjugate Gradient

Given initial $x_0 \in \mathbb{R}^n$, $r_0 = b - A x_0 \in \mathbb{R}^n$, $d_0 = r_0$

For $k = 1, 2, 3, \cdots$

$\overbrace{x_{k-1}, r_{k-1}, d_{k-1}, w_{k-1}, \alpha_{k-1}, \beta_{k-1}}^{4n+2}$

$w_{k-1} = A d_{k-1}$

$\alpha_{k-1} = \dfrac{d_{k-1}^T r_{k-1}}{d_{k-1}^T w_{k-1}}$

$x_k = x_{k-1} + \alpha_{k-1} d_{k-1}$

$r_k = r_{k-1} - \alpha_{k-1} w_{k-1}$

if $\|r_k\|_2 < \varepsilon$, break

- Cost : $O(n)$ memory

compute one Ad + $O(n)$ flops per iteration

$$d_k = r_k + \underbrace{d_{k-1} \frac{r_k^T r_k}{r_{k-1}^T r_{k-1}}}_{\beta_k}$$ ← only difference from Steepest Gradient Descent

end

- Does $\text{span}\{d_0, \dots, d_{k-1}\} = K_k(A, r_0)$? ✓

  Actually, also $K_k(A, r_0) = \text{span}\{r_0, r_1, \dots, r_{k-1}\}$

  By induction, $\text{span}\{d_0\} = \text{span}\{r_0\} = K_1(A, r_0)$

  Assume $\text{span}\{d_0, \dots, d_{k-1}\} = K_k(A, r_0)$

  then $r_k = r_{k-1} - \alpha_{k-1} A d_{k-1} \in K_{k+1}(A, r_0)$

  $d_k = r_k + d_{k-1} \beta_k \in K_{k+1}(A, r_0)$

  but $r_k \perp r_i, \; i = 0, 1, \dots, k-1$ linearly indep. $(r_k \neq 0)$

  $d_k \perp_A d_i, \; i = 0, 1, \dots, k-1$

  $\Rightarrow K_{k+1}(A, r_0) = \text{span}\{r_0, r_1, \dots, r_{k-1}\} = \text{span}\{d_0, \dots, d_{k-1}\}$

  CG is Krylov subspace method !

- Convergence of CG

  From $x_k = x_0 + \sum_{j=0}^{k-1} \alpha_j d_j \in x_0 + K_k$

  $$\Rightarrow f(x_k) = \min_{x \in x_0 + K_k} f(x)$$

  $$\Leftrightarrow \|x_k - x_*\|_A = \min_{x \in x_0 + K_k} \|x - x_*\|_A$$

  $$= \min_{p \in P_{k-1}} \|x_0 - x_* + p(A) r_0\|_A$$

$$= \min_{p \in P_{k-1}} \|(I - p(A)A)(x_0 - x_*)\|_A$$

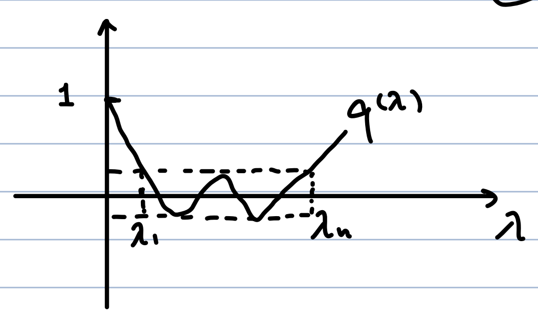
$$= \min_{\substack{q \in P_k \\ q(0)=1}} \| q(A)(x_0 - x_*)\|_A$$

Assume $A = Q^* \Lambda Q$ eigendecomposition $(A^T = A)$

$(0 < \lambda_1 \le \lambda_2 \le \cdots \le \lambda_n)$

$$\|x_k - x_*\|_A \le \underbrace{\min_{\substack{q \in P_k \\ q(0)=1}} \max_{1 \le i \le n} |q(\lambda_i)|}_{=: \mathcal{E}_k(A)} \|r_0\|_A$$

if $k \ge n$, $\mathcal{E}_k(A) = 0$
CG converges in n steps



Thm
$$\min_{\substack{q = a_k\lambda^k + \cdots + a_0 \in P_k \\ a_k = 1}} \max_{\lambda \in [-1,1]} |q(\lambda)| = 1$$

achieved by $q_* = T_k(\lambda) \longleftarrow$ chebyshev polynomial

$$T_k(\lambda) = \begin{cases} \cos(k \arccos(\lambda)), & |\lambda| \le 1 \\ \frac{1}{2}(\lambda + \sqrt{\lambda^2 - 1})^k + \frac{1}{2}(\lambda + \sqrt{\lambda^2 - 1})^{-k}, & |\lambda| \ge 1 \end{cases}$$ ∎

Apply this to $\mathcal{E}_k(A)$

$$\mathcal{E}_k(A) \le \min_{\substack{q \in P_k \\ q(0)=1}} \max_{\lambda \in [\lambda_1, \lambda_n]} |q(\lambda)|$$

$$= \max_{\lambda \in [\lambda_1, \lambda_n]} \left| \frac{T_k\left(1 + 2\frac{\lambda - \lambda_n}{\lambda_n - \lambda_1}\right)}{T_k\left(1 + 2\frac{-\lambda_n}{\lambda_n - \lambda_1}\right)} \right|$$

$$= \frac{1}{\left| T_k\left(1 - \frac{2\lambda_n}{\lambda_n - \lambda_1}\right)\right|} = \frac{1}{\left| T_k\left(\frac{K_2(A)+1}{K_2(A)-1}\right)\right|}$$

Note that $\frac{K_2(A)+1}{K_2(A)-1} + \sqrt{\left(\frac{K_2(A)+1}{K_2(A)-1}\right)^2 - 1} = \frac{\sqrt{K_2(A)}+1}{\sqrt{K_2(A)}-1}$

$$\Rightarrow \quad \mathcal{E}_k(A) \le \left(\frac{1}{2}\left(\frac{\sqrt{K_2(A)}+1}{\sqrt{K_2(A)}-1}\right)^k + \frac{1}{2}\left(\frac{\sqrt{K_2(A)}+1}{\sqrt{K_2(A)}-1}\right)^{-k} \right)^{-1}$$

$$\le 2\left(\frac{\sqrt{K_2(A)}-1}{\sqrt{K_2(A)}+1}\right)^k$$

when $A^* = A$ in GMRES, the error bound is similar !

$$\Rightarrow \quad \|x_k - x_*\|_A \le 2\left(\frac{\sqrt{K_2(A)}-1}{\sqrt{K_2(A)}+1}\right)^k \|x_0 - x_*\|_A$$

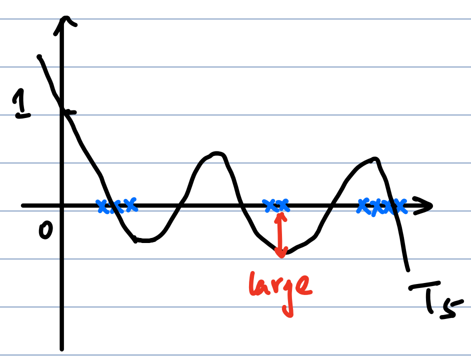
when $K_2(A) \gg 1$, $\left(\frac{\sqrt{K_2}-1}{\sqrt{K_2}+1}\right)^k \approx \left(1 - \frac{2}{\sqrt{K_2}}\right)^k \approx e^{-2k/\sqrt{K_2}}$

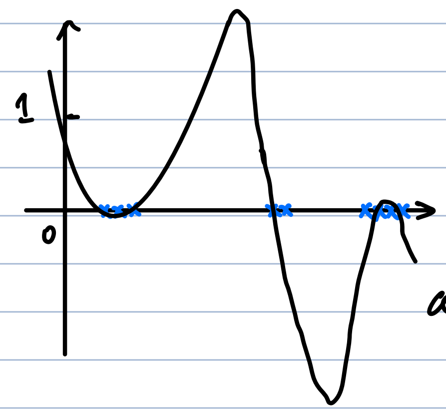
to make $\|x_k - x_*\|_A \le \varepsilon$

we need $k \geq O\left(\sqrt{K_2(A)} \log\left(\frac{1}{\varepsilon}\right)\right)$

better than steepest GD

— Remark 1: Only worst case bound, convergence can be faster





a better q_5

Actual convergence rate depends on the structure of the spectra of A. # cluster is important.

- Remark 2: CG is mathematically equivalent to FOM. Actually, we can derive CG from FOM directly by solving the equation $H_k y_k = \beta_1 e_1$ by LU decomposition.
(See pset 3)

- Remark 3: If A is not symmetric, can we use the idea of GMRES to extend CG? (CG is computationally saving)

In GMRES, $$\|b - Ax_k\|_2 = \min_{x \in x_0 + K_k} \|b - Ax\|_2$$
$$= \min_{x \in x_0 + K_k} \|A(x - x_*)\|_2$$
$$= \min_{x \in x_0 + K_k} \|x - x_*\|_{A^TA}$$

$\Rightarrow$ apply CG to $A^TAx = A^Tb$ $\longleftarrow$ CGN / CGNR / Conjugate Residual...

But convergence rate is only $O\left(\left(1 - \frac{1}{\kappa_2(A)}\right)^k\right)$