Welcome back to Ma221! Lecture 35, Nov 15

Splitting Methods for $Ax=b$

Goal: Given initial guess x_0
for solution of $Ax=b$, cheaply
compute sequence $x_i \to A^{-1}b$

Def: Splitting of $A = M-K$, M nonsingular

$$Ax=b \Leftrightarrow Mx = Kx+b$$

compute x_{i+1} by solving $Mx_{i+1} = Kx_i+b$

$$(*) \quad x_{i+1} = M^{-1}Kx_i + M^{-1}b = Rx_i + c$$

For $(*)$ to work well need:

(1) x_i should converge to $A^{-1}b$, for any x_0

(2) Solving $Mx_{i+1} = Kx_i+b$ for x_{i+1}
should be much cheaper than solving with A

Lemma: Let $\|\cdot\|$ be any operator norm

$$\|R\| = \max_{x\neq 0} \frac{\|Rx\|}{\|x\|}$$

then if $\|R\|<1$ $(*)$ converges to $A^{-1}b$
for any x_0

Proof: Subtract $x = Rx+c$ from $(*)$

$$x_{i+1} - x = R(x_i - x) = R^{i+1}(x_0 - x)$$

$$\|x_{i+1}-x\| \le \|R\|^{i+1}\|x_0 - x\|$$

$$\to 0 \text{ if } \|R\|<1$$

Def: The spectral radius $\rho(R)$ of R is

$$\rho(R) = \max_{\substack{\lambda \text{ an eval} \\ \text{of } R}} |\lambda|$$

Thm: $x_{i+1} = Rx_i + c$ converges to $A^{-1}b$ for all x_0 iff $\rho(R) < 1$

Lemma: For all operator norms, $\rho(R) \le \|R\|$

For any R and $\varepsilon > 0$, there exist $\|\cdot\|^*$ such that $\|R\|^* \le \rho(R) + \varepsilon$

Proof: To show $\rho(R) \le \|R\| = \max_{x \ne 0} \frac{\|Rx\|}{\|x\|}$

choose $x =$ evec for λ, $|\lambda| = \rho(R)$

$$\|R\| \ge \frac{\|Rx\|}{\|x\|} = \frac{\|\lambda x\|}{\|x\|} = |\lambda| = \rho(R)$$

to construct $\|\cdot\|^*$, use Jordan Form

$$S^{-1}RS = J = \begin{bmatrix} \square & & & \\ & \square & & \\ & & \square & \\ & & & \ddots \end{bmatrix} \qquad \square = \begin{bmatrix} \lambda & 1 & & \\ & \ddots & \ddots & \\ & & \ddots & 1 \\ & & & \lambda \end{bmatrix}$$

$$D = \begin{bmatrix} 1 & & & \\ & \varepsilon & & & \\ & & \varepsilon^2 & & \\ & & & \ddots & \\ & & & & \varepsilon^{n-1} \end{bmatrix}$$

$D^{-1}JD \to$ all 1's turn into ε's

$$J_\varepsilon = D^{-1}S^{-1}RSD = \begin{bmatrix} \lambda_1 & \varepsilon & & \\ & \ddots & 0 & \\ & & \ddots & \varepsilon \\ & & & \lambda_n \end{bmatrix}$$

to define $\|R\|^*$ define $\|x\|^* = \|(SD^{-1})x\|_\infty$

$$\|R\|^* = \max_{x\neq 0} \frac{\|Rx\|^*}{\|x\|^*} = \max_{x\neq 0} \frac{\|(SD)^{-1}Rx\|_\infty}{\|\underbrace{(SD)^{-1}x}_{y}\|_\infty}$$

$$= \max_{y\neq 0} \frac{\|(SD)^{-1}R(SD)y\|_\infty}{\|y\|_\infty} = \max_{y\neq 0} \frac{\|J_\varepsilon y\|_\infty}{\|y\|_\infty} = \|J_\varepsilon\|_\infty$$

$\leq \rho(R) + \varepsilon$ QED

Proof of Thm: if $\rho(R) \geq 1$ choose x_0 so that
$x_0 - x$ = evec for λ where $|\lambda| = \rho(R) \geq 1$
$x_i - x = R^i(x_0 - x) = \lambda^i(x_0 - x) \rightarrow$ does not converge to 0

if $\rho(R) < 1$, use Lemma to construct $\|R\|^*$
such that $\|R\|^* = \underbrace{\rho(R)}_{<1} + \varepsilon$, choose ε
so $\|R\|^* < 1$
$\Rightarrow$ convergence $\forall x_0$ by first Lemma

Goal for $A = M - K$: $\rho(R)$ as small as possible
but still cheap to solve $Mx_{i+1} = Kx_i + b$

Ex: $M = I$, $K = I - A \Rightarrow$ solving for x_{i+1} very cheap, but no guarantees on $\rho(R) = \rho(M^{-1}K) = \rho(I - A)$

Ex: $K = 0 \Rightarrow R = 0 \Rightarrow$ converge in one step
but need $c = M^{-1}b = A^{-1}b$, no savings

Describe Jacobi's, Gauss-Seidel (GS)
Successive Overrelaxation (SOR)

$$A = \begin{bmatrix} \ddots & & -U' \\ & D & \\ -L' & & \ddots \end{bmatrix} = D - L' - U' = D(I - L - U)$$

Jacobi: In words: for $j = 1$ to n, pick $x_{i+1}(j)$ to exactly solve equation j

As a Loop: for $j = 1$ to n

$$x_{i+1}(j) = \Big(b_j - \sum_{k \neq j} A_{jk} x_i(k)\Big) / A_{jj}$$

As a Splitting: $D x_{i+1} = (L' + U') x_i + b$

$$A = M - K = D - (L' + U')$$

$$R_J = M^{-1} K = D^{-1}(L' + U') = L + U$$

For 2D Poisson: $T_N V + V T_N = h^2 F$

$V^{N \times N}$ unknowns

T^N 1D Poisson

to get from V_i to V_{i+1}

for $j = 1:N$, for $k = 1:N$

$$V_{i+1}(j,k) = \big(V_i(j-1,k) + V_i(j+1,k) + V_i(j,k-1) + V_i(j,k+1) + h^2 F(j,k)\big) / 4$$

= "average" of 4 nearest neighbors and right hand side F

Gauss-Seidel

In words: improve on Jacobi by using most recently updated values of x

As a loop for $j=1:n$

$$x_{i+1}(j) = \Big(b_j - \underbrace{\sum_{k<j} A_{jk} x_{i+1}(k)}_{\text{... updated } x} - \underbrace{\sum_{k>j} A_{jk} x_i(k)}_{\text{... old } x}\Big) / A_{jj}$$

As a Splitting: $A = (D - L') - U' = M - K$

each step of GS is a triangular solve

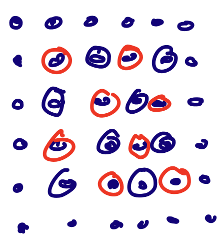
$$R_{GS} = M^{-1}K = (D-L')^{-1}U'$$
$$= (D(I-L))^{-1}U'$$
$$= (I-L)^{-1}U$$

In contrast to Jacobi, order of updating $x_{i+1}(j)$ matters

for 2D Poisson

Natural Order (rowwise or columnwise updating of $v(j,k)$)

Red-Black Ordering



Red $\equiv$ $j+k$ even

Black $\equiv$ $j+k$ odd

Number all Red node nodes before Black nodes

Red nodes only have Black neighbors (vice versa)

$\Rightarrow$ When updating Red nodes, can update them in any order (Parallelism), all Black nodes have "old data". Then update all Black nodes (parallel) using updated Red data

forall Red (j,k) $\quad j+k$ even

$$\left.\begin{aligned} V_{i+1}(j,k) = (&V_i(j-1,k) + V_i(j+1,k) \\ &+ V_i(j,k-1) + V_i(j,k+1) \\ &+ h^2 F(j,k))/4 \end{aligned}\right) \text{old black data}$$

for all Black (j,k) $\quad j+k$ odd

$$\left.\begin{aligned} V_{i+1}(j,k) = (&V_{i+1}(j-1,k) + V_{i+1}(j+1,k) \\ &+ V_{i+1}(j,k-1) + V_{i+1}(j,k+1) \\ &+ h^2 F(j,k))/4 \end{aligned}\right) \text{new red data}$$

SOR:

In words: Depends on parameter w

Result of SOR = weighted comb of old x and result of GS

$$x^{SOR}_{w,i+1}(j) = (1-w)x_i(j) + w\, x^{GS}_{i+1}(j)$$

$w=1 \Rightarrow$ same as GS

$w<1 \Rightarrow$ "underrelaxation", not useful

$w>1 \Rightarrow$ "over relaxation": good idea since x_{i+1}^{GS} better than x_i, go farther in same direction

Later will pick ω optimally for Poisson

As a loop:

for $j = 1:n$

$$x_{i+1}(j) = (1-\omega)\, x_i(j) + \omega\Big(b_j - \sum_{k<j} A_{jk}\, x_{i+1}(k) - \sum_{k>j} A_{jk}\, x_i(k)\Big) / A_{jj}$$