Math 609

L # 7

Tuesday
Sept. 20, 2011

Convergence of the simplest iteration methods

The fundamental theorem of iterative methods for

(1) $\quad x^{(k+1)} = G x^{(k)} + c \qquad x^{(0)}$ – arbitrary given

Th. $\rho(G) < 1$ is sufficient and necessary for the convergence of the iteration (1)

We shall study the following methods for $A = D - L - U$

Jacobi $\qquad G = I - D^{-1}A = D^{-1}(L+U)$

Gauss-Seidel: $\quad G = I - (D-L)^{-1}A = (D-L)^{-1}U$

SOR $\qquad G = I - (\frac{1}{\omega}D - L)^{-1}A = (\frac{1}{\omega}D - L)^{-1}((1-\frac{1}{\omega})D - U)$

$$A = \omega^{-1}D - L + ((1-\tfrac{1}{\omega})D - U)$$

<u>Theorem</u> Gauss-Seidel converges for strictly diagonally dominant matrix A.

G-S $\quad x^{(k+1)} = \underbrace{(D-L)^{-1}U}_{G} x^{(k)} + (D-L)^{-1}b$

$Gx = \lambda x \qquad (D-L)^{-1}Ux = \lambda x$

$\lambda \in \sigma(G) \qquad Ux = \lambda(D-L)x$

$$\boxed{(\lambda L + U)x = \lambda D x}$$

λ let us look at i-th equation of this system

$(\lambda, x) \quad \begin{cases} -\lambda a_{i1}x_1 - \lambda a_{i2}x_2 - \dots - \lambda a_{i,i-1}x_{i-1} - a_{i,i+1}x_{i+1} - \dots = \lambda a_{ii} x_i \\ \quad i = 1, \dots, n \end{cases}$

assume $x_k = \max_i |x_i| = 1$

$$\lambda a_{kk} = -\lambda \sum_{j=1}^{k-1} a_{kj} x_j - \sum_{j=k+1}^{n} a_{kj} x_j$$

$$|\lambda| |a_{kk}| \le |\lambda| \sum_{j=1}^{k-1} |a_{kj}| + \sum |a_{kj}| \qquad \lambda \neq 0$$

$$|a_{kk}| \le \sum_{j=1}^{k-1} |a_{kj}| + \frac{1}{|\lambda|} \sum_{j=k+1}^{n} |a_{kj}|$$

if $|\lambda| \ge 1 \Rightarrow \frac{1}{|\lambda|} \le 1 \qquad |a_{kk}| \le \sum_{\substack{j=1 \\ j \neq k}}^{n} |a_{kj}|$ impossible

$|\lambda|$ cannot be ≥ 1 ↯

Let us go back to basic iterative methods for $Ax=b$ when A is an SPD matrix.
$A = D - L - L^T$

splitting Q operator $G = (I - Q^{-1}A)$

Richardson	$Q = \frac{1}{\tau} I$	$G = I - \tau A$
Jacobi	$Q = D$	$G = I - D^{-1}(D - L - L^T) = L + L^T$
Gauss-Seidel	$Q = D - L$	$G = (D - L)U$
SOR	$Q = \frac{1}{\omega} D - L$	$G = (D - \omega L)^{-1}((1-\omega)D + \omega U)$

Richardson $\quad \lambda(A) \quad 0 < \lambda_1(A) \leq \ldots \leq \lambda_n(A)$

(λ_j, ψ_j) eigenpairs $\quad \tau = \frac{1}{\lambda_n(A)} \quad \rho(G) = 1 - \frac{\lambda_1(A)}{\lambda_n(A)}$

But G is symmetric

$$\rho(G) = \|G\|_2 = 1 - \frac{\lambda_1(A)}{\lambda_n(A)} < 1$$

$$e^{(k+1)} = \sum_j (1 - \tau\lambda_j) c_j^{(k)} \psi_j \qquad \tau = \lambda_n(A)$$

$1 - \tau\lambda_{98} = 10^{-4} \quad 1 - \tau\lambda_{97} \approx 10^{-3}$

$$e^{(1)} = \sum_{j=1}^{n} (1 - \tau\lambda_j) c_j^{(0)} \psi_j = \sum_{j=1}^{n-1} (1 - \tau\lambda_j) c_j^{(0)} \psi_j$$

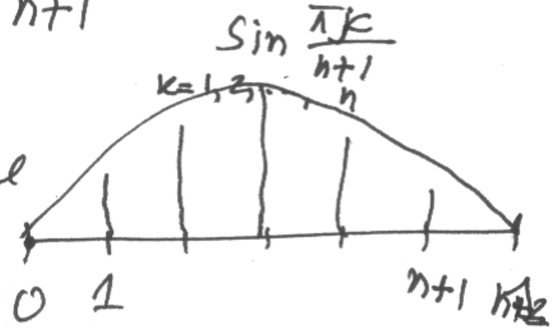
the last term associated with the error component along ψ_n is gone!

$$A = \begin{vmatrix} 2 & -1 & 0 & & 0 & 0 \\ -1 & 2 & -1 & & 0 & 0 \\ & & & & & \\ 0 & & & & 2 & -1 \\ 0 & & & & -1 & 2 \end{vmatrix} \in \mathbb{R}^{n\times n}$$

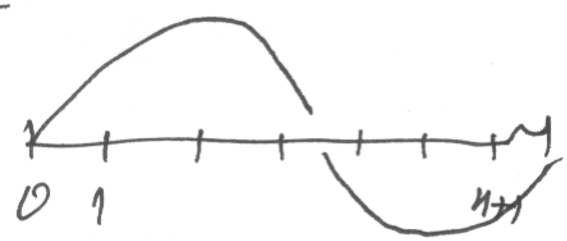
$$A\psi_j = \lambda_j \cdot \psi_j$$

$$\lambda_j = 4\sin^2\frac{\pi j}{2(n+1)} \qquad (\psi_j)_k = \sin\frac{\pi j k}{n+1}$$

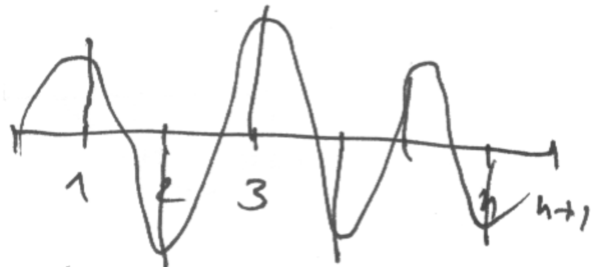
$j=1$ the lowest eigenvalue



$j=2$ the second smallest



$j=n$



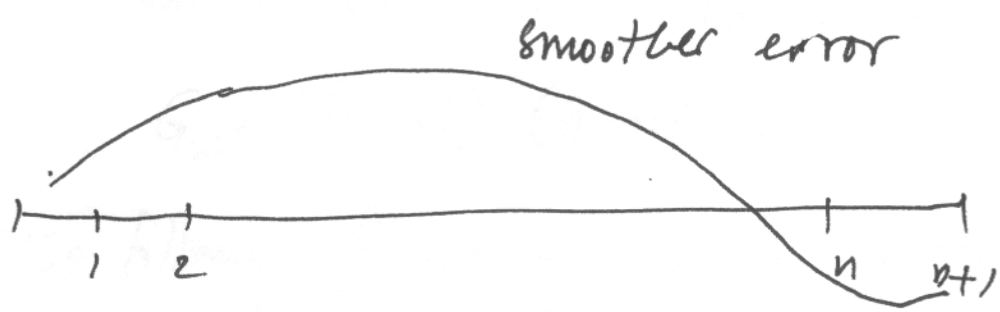
highly oscilatory

$$\boxed{(\psi_j, \psi_i) = \delta_{ij}\cdot 2}$$

$$\frac{\lambda_{n-1}}{\lambda_n} \sim 1 \qquad 1-\frac{\lambda_{n-1}}{\lambda_n} \approx \text{small } \underline{\underline{\varepsilon}}$$

$\boxed{\text{if } n=99 \quad \varepsilon = 0.74 * 10^{-3}}$

This means that after 2-3 iterations the highly oscilatory part of the error will be gone



Richardson & Gauss-Seidel are known to be good smoothers for A SPD!

SOR Method (as an extension of Gauss-Seidel)

Gauss-Seidel

$$a_{i1} x_1^{(k+1)} + \dots + a_{i,i-1} x_{i-1}^{(k+1)} + \underbrace{a_{ii} x_i^{(k+1)}} + a_{i,i+1} x_{i+1}^{(k)} + \dots + a_{in} x_n^{(k)} = b_i$$

for $i = 1, \dots, n$

$$\xi_i = \left(b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k+1)} - \sum_j a_{ij} x_j^{(k)}\right) / a_{ii}$$

G-S $\qquad x_i^{(k+1)} = \xi_i$

SOR $\qquad x_i^{(k+1)} = \omega \xi_i + (1-\omega) x_i^{(k)}$

end i $\qquad \omega$ parameter (extrapolation)

In matrix from $\qquad x^{(k+1)} = G_\omega x^{(k)} + c$

$$G_\omega = (D - \omega L)^{-1} \left((1-\omega) D + \omega U\right)$$

The following fundamental results concerns the choice of the parameter ω.

Theorem: Necessary condition for convergence of SOR method for any initial guess is $0 < \omega < 2$

Proof. It is rather simple consequence of the Fundamental Theorem for iterative methods. We shall prove that $\rho(G_\omega) < 1$ $\rho(G_\omega) = \max_{\lambda \in \sigma(G_\omega)} |\lambda|$

Recall the spectrum of G_ω are all complex number s.t. $\det |G_\omega - \lambda I| = 0$

$$\det(G_\omega - \lambda I) = (-1)^n \lambda^n + a_1 \lambda^{n-1} + \dots + \det(G_\omega)$$

From the Viet formula we have

$$\lambda_1 \lambda_2 \dots \lambda_n = \det(G_\omega)$$

No look at G_ω

$$G_\omega = (D - \omega L)^{-1}((1-\omega)D + \omega U) =$$

$$G_\omega = \left[D(I - \omega D^{-1} L)\right]^{-1} \cdot D\left[(1-\omega)I + D^{-1}U\right]$$

$$(I - \omega D^{-1} L)^{-1} \cancel{D^{-1}} \cancel{D} ((1-\omega)I + D^{-1}U)$$

$$\boxed{G_\omega = (I - \omega D^{-1} L)^{-1}((1-\omega)I + D^{-1}U)}$$

$$\det(G_\omega) = \underbrace{\det(I - \omega D^{-1} L)^{-1}}_{1} \underbrace{\det((1-\omega)I + D^{-1}U)}_{(1-\omega)^n}$$

$$|\lambda_1 \lambda_2 \dots \lambda_n| = |1-\omega|^n$$

The necessary condition for convergence of SOR means $\rho(G_\omega) < 1$ that is all $|\mu_i| < 1$. But in this case

$$|1-\omega|^n < 1 \Rightarrow |1-\omega| < 1 \Rightarrow -1 < 1-\omega < 1$$

$$\boxed{\text{that is} \quad 0 < \omega < 2}$$

If $|1-\omega|^n \geq 1$ means that at least one $|\mu_{i_0}| \geq 1$ and therefore $\rho(G_\omega) \geq 1$ and SOR diverges. Therefore, necessarily ~~we~~ $|1-\omega|^n < 1$, i.e.

$$0 < \omega < 2$$

SSOR one step forward SOR
one step backward SOR

forward $Q_f = \frac{1}{\omega} D - L$ $\qquad G_f = I - Q_f^{-1} A$
backward $Q_b = \frac{1}{\omega} D - U$ $\qquad G_b = I - Q_b^{-1} A$

$$G_{SSOR} = (I - Q_b^{-1} A)(I - Q_f^{-1} A)$$

$$x^{(k+\frac{1}{2})} = (I - Q_f^{-1} A) x^{(k)} + \omega Q_f^{-1} b \qquad \text{intermediate step}$$

$$x^{(k+1)} = (I - Q_b^{-1} A) x^{(k+1/2)} + \omega Q_b^{-1} b$$

this is how you implement SSOR

To analyze the method we shall write it down in the form

$$x^{(k+1)} = G_{SSOR}\, x^{(k)} + c$$

by eliminating the intermediate step

$$x^{(k+1)} = (I - Q_b^{-1}A)\left[(I - Q_f^{-1}A)x^{(k)} + \omega Q_f^{-1} b\right] + \omega Q_b^{-1} b$$

$$x^{(k+1)} = (I - Q_b^{-1}A)(I - Q_f^{-1}A)x^{(k)} + c \equiv G x^{(k)} + c$$

$$c = \omega (I - Q_b^{-1}A) Q_f^{-1} b + \omega Q_b^{-1} b$$

Very importan observation is that G is symmetric in the inner product (Ax, y) for A an SPD matrix.

Indeed

$$(AGx, y) \overset{?}{=} (Ax, Gy) \quad \text{is this true}$$

$$\| $$

$$(A(I - Q_b^{-1}A)(I - Q_f^{-1}A)x, y) = ((I - AQ_b^{-1})(I - AQ_f)Ax, y)$$

$$= ((I - AQ_f^{-1})Ax, (I - AQ_b^{-1})^T y) = (Ax, \underbrace{(I - AQ_f^{-1})^T}_{I - Q_f^{-1T}A^T}(I - Q_f^{-1}A)y)$$

$$I - (Q_b^{-1})^T A^T$$

$$= (Ax, (I - Q_b^{-1}A)(I - Q_f^{-1}A)y)$$

$$= (Ax, Gy) \quad \text{Yes}$$

Issues of implementation

Important in implementation is to develope a strategy of matrix free computations

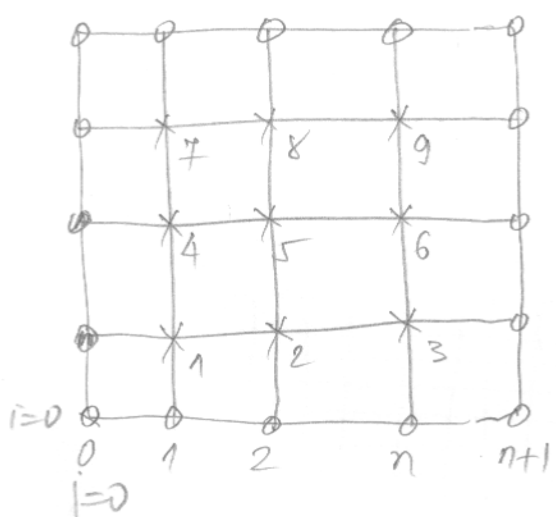
$$D\left(x^{(k+1)} - x^{(k)}\right) = f - Ax^{(k)}$$

What do you need to implement the iteration.

① You need to keep two consecutive vectors

$x^{(k+1)} = x_{new}$ $\quad x^{(k)} = x_{old}$ in $\mathbb{R}^{N}$

② You need for a given vector $x^{(k)}$ to produce a vector $Ax^{(k)}$ in $\mathbb{R}^N$



0 – zero values

x – unknown values

x_{ij} $\quad i = 1, \dots, n$
$\quad j = 1, \dots, n$

all together we have

x_{ij} $\quad i,j = 0, \dots, n+1$

$i=1$

$$\left\{ (Ax)_{ij} = (4+h^2)\,x_{ij} - x_{i+1j} - x_{i-1j} - x_{ij+1} - x_{ij-1} \right.$$

$i,j = 1, \dots, n$ $\quad \left\{ \begin{array}{ll} i=1 & j=1,\dots,n \\ i=2 & j=1,\dots,n \\ \vdots & \\ i=n & j=1,\dots,n \end{array} \right.$ lexicographical ordering

You can produce every $b_{ij} - (Ax)_{ij} = b_{ij} - (Ax)_{ij}$

But smooth functions can be represented by fewer unknowns. The notion is that we do not need any longer all eigenvectors to represent the remaining part of the solution.

The idea of multigrid or multilevel preconditioners is to project the residual to a smaller space and continue to iterate but with say 4-time less unknowns that before.

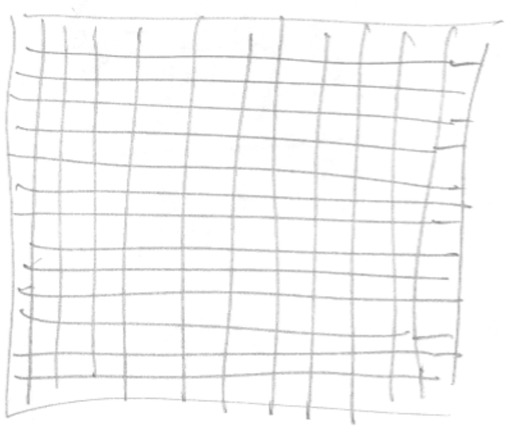
After a few iterations you get smoothed the error again and you can go even to a smaller space

$-\Delta u = f$

on 1000×1000 mesh

10^6 unknowns

500×500



250×250

100×100

Good smoothers are
Jacobi, Richardson, Gauss-Seidel

50×50

Next you need the diagonal matrix D

— for Jacobi diag. matrix D

— for Richardson the parameter τ

— for SOR & Gauss-Seidel

$D^{-1}(L+U)$

$(D-L)x^{(k+1)} = +Ux^{(k)} + b$

$A = a_{ij}$

$(D-L)x_s^{(k+1)} - Ux^{(k)} = b$

Write the s-the equation

$$a_{s1}x_1^{(k+1)} + a_{s2}x_2^{(k+1)} + \dots + a_{ss}x_s^{(k+1)} + a_{ss+1}x_{s+1}^{(k)} + \dots a_{sn}x_n^{(k)} = b_s$$

$$b_s - \sum_{j=1}^{s-1} a_{sj} \cdot x_j^{(k+1)} - \sum_{j=s+1}^{n} a_{sj} \cdot x_j^{(k)} = a_{ss}\, x_s^{(k+1)}$$

$$\xi_s = \left(b_s - \sum_{j=1}^{s-1} a_{sj}\, x_j^{(k+1)} - \sum_{j=s+1}^{n} a_{sj} \cdot x_j^{(k)}\right) / a_{ss}$$

G-S $\quad x_s^{(k+1)} = \xi_s \qquad s = 1, \dots, n$

SOR $\quad x_s^{(k+1)} = \omega\, \xi_s + (1-\omega)\, x_s^{(k)}$

$s = 1, \dots n$