Numerical Method Homework4

Xiangbo Wang

June 10, 2011

1 Problem1

Let's assume x is any eigenvalue of M which is a **real** positive definite matrix. Then $Mx = \lambda x$ and therefore $x^T Mx = x^T \lambda x = \lambda x^T x = \lambda x \cdot x = \lambda \|x\|^2$. Since M is a positive definite matrix, it holds that $x^T Mx > 0$. Then $\lambda \|x\|^2$ must be positive and therefore λ must be positive.

Let's assume y is any eigenvalue of M which is a **complex** positive definite matrix. Then $My = \lambda y$ and therefore $\Re(y^* My) = \Re(y^* \lambda y) = \lambda \Re(y^* y) = \lambda \Re(y^*)\Re(y)$. Since $y^* = y^T$ if y is real, then $\lambda \Re(y^*)\Re(y) = \lambda \Re(y^T)\Re(y) = \lambda \Re(y) \cdot \Re(y) = \lambda \|\Re(y)\|^2$. Since M is a positive definite matrix, it holds that $\Re(y^* My) > 0$. Then $\lambda \|\Re(y)\|^2$ must be positive and therefore λ must be positive.

In conclusion, the eigenvalues of a positive definite matrix are all positive.

2 Problem2

The condition number is defined be be the maximum ratio of the relative error in x divided by the relative error in b of linear equation $Ax = b$. Let's assume the error in the solution $A^{-1}b$ is $A^{-1}e$. Then the ratio of the relative error in the solution to the relative error in b is

$$\frac{\|A^{-1}e\|/\|A^{-1}b\|}{\|e\|/\|b\|} = (\|A^{-1}e\|/\|e\|)(\|b\|/\|A^{-1}b\|) \tag{1}$$

Then the maximum value could be seen to be the product of the two operator norms

$$\kappa(A) = \|A\| \cdot \|A^{-1}\| \tag{2}$$

Since the norm of a matrix could be obtain by

$$\|A\| = \sqrt{\text{largest eigenvalue of } A^* A} \tag{3}$$

and the matrix A in this question is an unitary matrix which has the properties of $A^\dagger A = I_n$ and $A^\dagger = A^{-1}$ ($A^\dagger$ is the same as A^*), thus from (3) we could obtain that

$$\|A\| = \sqrt{\text{largest eigenvalue of } I_n} \tag{4}$$

$$\begin{aligned} \|A^{-1}\| &= \sqrt{\text{largest eigenvalue of } A^{-1^*}A^{-1}} \\ &= \sqrt{\text{largest eigenvalue of } A^{\dagger^*}A^\dagger} \\ &= \sqrt{\text{largest eigenvalue of } AA^\dagger} \\ &= \sqrt{\text{largest eigenvalue of } I_n} \end{aligned} \tag{5}$$

Therefore from (2), (4) and (5) we finally have that the condition number of an unitary matrix is

$$\kappa(A) = \sqrt{\text{largest eigenvalue of } I_n}\sqrt{\text{largest eigenvalue of } I_n} = 1 \tag{6}$$

3 Problem3

(a) The cubic spline interpolator we need for this system is like following equations:

$$\begin{aligned} s_i(x_i) &= a_i(x-x_i)^3 + b_i(x-x_i)^2 + c_i(x-x_i) + d_i \\ s_i(x_i) &= a_i h_i^3 + b_i h_i^2 + c_i h_i + d_i \end{aligned} \tag{7}$$

And for each spline, it must satisfy $s_i(x) = d_i$.

(b) My code is in $p3.m$ and the test file is $p3\_test.m$. Usage: Having an array of data as input and call like $p3(N)$.

4 Problem4

(a) From the system of equations $x' = Ax$ in the question, we can obtain that $\frac{dx}{dt} = Ax$ so that $x = e^{At} + c$. While $t = 0$ and $x = x_0$, then we can obtain that $x = e^{At} - tx_0$ and $e^A = ve^\lambda v$. Then with the *Forward Euler*:

$$\frac{du}{dt} = \lambda u \tag{8}$$

$$\frac{u^{n+1} - u^n}{dt} = \lambda u^n \tag{9}$$

$$u^{n+1} = u^n + \lambda dt u^n \tag{10}$$

$$u^{n+1} = u^n + A dt u^n \tag{11}$$

$$\frac{u^{n+1}}{u^n} = 1 + Adt \tag{12}$$

where $u^{n+1} = u(x_{n+1})$ and $c = 0 andu = u^0$. The limitation should be imposed on the timesetp in order to get the numerical stability. Therefore, to get more stability of the solution, the discrete dt should be more discrete.

(b) With the *Backward Euler*, we will obtain that:

$$\frac{u^{n+1} - u^n}{dt} = Au^{n+1} \tag{13}$$

$$u^{n+1} - u^n = Adtu^{n+1} \tag{14}$$

$$u^n = u^{n+1}(1 - Adt) \tag{15}$$

$$\frac{u^n}{u^{n+1}} = 1 - Adt \tag{16}$$

The limitation should be imposed on the timesetp in order to get the numerical stability. However, since the *Backward Euler* are numerically stable, therefore the size of the timestep is not important.

5 Problem5

Since $Av = \lambda v$, given $(I - vv')Av$, we can obtain that

$$\begin{aligned}(I - vv')Av &= (I - vv')\lambda v \\ &= \lambda(I - vv')v \\ &= \lambda(Iv - v(v'v))\end{aligned} \tag{17}$$

Since $Iv = v$ and $v'v = \|v\|$, we can obtain that

$$\lambda(Iv - v(v'v)) = \lambda(v - v\|v\|) \tag{18}$$

Since $\|v\| = 1$, we can get that

$$\begin{aligned}\lambda(v - v\|v\|) &= \lambda(v - v \times 1) \\ &= \lambda \times 0\end{aligned} \tag{19}$$

Therefore, $(I - vv')Av = 0$.

6 Problem6

(a) To find sufficient conditions, the matrix A has to be strictly dominant and irresolvable. A matrix is said to be strictly dominant if

$$|a_{ii}| > \sum_{j=1, j\neq i}^{n} |a_{ij}|,\ i \in N = 1, 2, ..., n \tag{20}$$

So in this case, $|5| > |\alpha|$ which means $\alpha > 5$ or $\alpha < -5$. Also, A matrix is said to be irresolvable if sets I and J such that $I \neq \phi$, $J \neq \phi$, $I \cap J \neq \phi$, $I \cup J = N = 1, 2, ..., n$, $a_{ij} = 0$ do not exist.

Therefore $\alpha > 5$ or $\alpha < -5$.

(b) For the methods, if the spectral radius of the iteration matrix is less than one, the convergence will be guaranteed. Based on this, I found a matrix Jacob method got convergence while Gauss-Seidel method doesn't.

$$A = \begin{pmatrix} 3 & -5 & 2 \\ 5 & 4 & 3 \\ 2 & 5 & 3 \end{pmatrix} \tag{21}$$

With $b = (1\ 1\ 1)'$ and $x_0 = (0\ 0\ 0)'$, Jacob method could get the solution with 204 iterations, while Gauss-Seidel can't. And the $\rho(B) = 0.91 < 1$ for Jacob method but $\rho(B) = 2.16 > 1$ for Gauss-Seidel.

7 Problem7

My code for Jacobi is in *p7_j.m* and the results are in *result_7j.xls*. It works pretty good when the N is not large [10, 20 and 40]. However, it works bad on $N = 80$ where it needs 20222 iterations. And it doesn't stop when $N = 160$.

My code for SSOR is in *p7_s.m* and the results are in *result_7s.xls*. It works pretty good for all the N equals to 10, 20, 40, 80 and 160.

In conclusion, the SSOR is much better.

8 Problem8

My code for Non-preconditioner, Jacobi and SSOR are all in *p8.m* and the results are in *result_8.xls*. Non-preconditioner and Jacobi got the same iterations which means that it didn't help much. However, SSOR gave me nothing which made things even worse.