Sheng Xu

MATH 226, HOMEWORK 3, DUE NOV. 13, 2015

For problems that include programming, please include the code and all outputted figures and tables. Please label these clearly and refer to them appropriately in your answers to the questions.

(1) Show that Jacobi method converges for all 2 by 2 symmetric positive definite matrices.
Proof: Let $A = \begin{pmatrix} a & b \\ b & c \end{pmatrix}$ Therefore:

$$D = \begin{pmatrix} a & \\ & c \end{pmatrix}$$

;

$$R = \begin{pmatrix} & b \\ b & \end{pmatrix}$$

We know that if we want the method to be convergent, $\rho(I_n - BA) < 1$, which is $\rho(I_n - D^{-1}A) < 1$. Therefore, since:

$$I_n - D^{-1}A = D^{-1}(D - A) = -D^{-1}R$$

We have $\rho(I_n - D^{-1}A) = \rho(-D^{-1}R) = \rho(D^{-1}R)$. Hence, the method is convergent if $\rho(D^{-1}R) < 1$.
Since A is a symmetric positive definite matric, $ac - b^2 > 0$. Thus $0 < \dfrac{b^2}{ac} < 1$. Now Let's compute $D^{-1}R$:

$$D^{-1}R = \begin{pmatrix} a^{-1} & \\ & c^{-1} \end{pmatrix} \cdot \begin{pmatrix} & b \\ b & \end{pmatrix} = \begin{pmatrix} & \dfrac{b}{a} \\ \dfrac{b}{c} & \end{pmatrix}$$

Now we try to calculate the eigenvalues of $D^{-1}R$ where $|\lambda I_2 - D^{-1}R| = 0$ Thus $\lambda^2 = \dfrac{b^2}{ac} < 1$, which means the absolute value of every eigenvalue is less than 1. Hence, $\rho(D^{-1}R) < 1$, which means Jacobi method converges.

(2) (a) Implement (all the programs should be functions that take matrix A, right hand side b, initial guess x^0, tolerance and maximal number of iterations as input arguments and output the approximate solution and number of iterations):
   (i) Richardson method
   (ii) Jacobi method
   (iii) Guass-Seidel method
   (iv) Successive Overrelaxation (SOR) method

(b) Consider the following $n \times n$ system:

$$\begin{pmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & 2 \end{pmatrix} \begin{pmatrix} x_1 \\ \\ \vdots \\ \\ x_n \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \\ -1 \end{pmatrix},$$

for $n = 2^l$, $l = 4, 5, 6, 7, 8$. Use the programs above to solve it until $||b - Ax^k||_2/||b||_2 < 10^{-6}$. Make a table to report the number of iterations of each iterative method and discuss the results. For Richardson method and SOR, please try different values of ω and discuss how the results depends on the choice of ω.

Table 2-1 Relationship between numbers of iteration and method

Method	$Richardson$	$Jacobi$	$Gauss-seidel$	SOR
Times($i = 4$)	644	644	909	668
Times($i = 5$)	2217	2217	3090	2216
Times($i = 6$)	7741	7741	10666	7576
Times($i = 7$)	27033	27033	36780	26091
Times($i = 8$)	93463	93463	125163	89216

Where the $w_{Richardson} = \dfrac{2}{\lambda_{min}(A) + \lambda_{max}(A)}$, $w_{SOR} = \dfrac{2}{1 + \sqrt{1 - \rho(I - D^{-1}A)^2}}$.
From the table we can conclude that with the optional value of $w_{Richardson}$ given in class, the method of Richardson iteration seem to be as efficient as Jacobi method. When we check their B matrixes, they are the same with this particular A matrix given. The Gauss-Seidel method seems to converge slowly in this example, and SOR method seems to converge a little more quickly than Jacobi and Richardson.

Table 2-2 Relationship between numbers of iteration and w for Richardson

w	wR_1	wR_2	wR_3
Times($i = 4$)	1161	1247	644
Times($i = 5$)	3971	4281	2217
Times($i = 6$)	13699	14888	7741
Times($i = 7$)	47050	51727	27033
Times($i = 8$)	159092	177648	93463

Where the $wR_1 = \dfrac{1}{\lambda_{max}(A)}$, $wR_2 = \dfrac{2}{\lambda_{min}(A) + 0.5\lambda_{max}(A)}$, $wR_3 = \dfrac{2}{\lambda_{min}(A) + \lambda_{max}(A)}$
We do not spot a better convergence in the table than the given $w = \dfrac{2}{\lambda_{min}(A) + \lambda_{max}(A)}$ in class. Also if we choose bigger w($w = \dfrac{3}{\lambda_{max}(A)}$), the method may not converge.

Table 2-3 Relationship between numbers of iteration and w for SOR

w	$wSOR_1$	$wSOR_2$	$wSOR_3$	3
Times($i=4$)	668	909	846	535
Times($i=5$)	2216	3090	2974	1828
Times($i=6$)	7576	10666	10460	6352
Times($i=7$)	26091	36780	36422	22099
Times($i=8$)	89216	125163	124563	76119

Where the $wSOR_1 = \dfrac{2}{1+\sqrt{1-\rho(I-D^{-1}A)^2}}$, $wSOR_2 = 1$, $wR_3 = 1+\sqrt{1-\rho(I-D^{-1}A)^2}$

Still, We do not spot a better convergence in the table than the given $w = \dfrac{2}{1+\sqrt{1-\rho(I-D^{-1}A)^2}}$ in class while obeying the Rule in PPT. However if we implement $w = 3$ we will need less iteration but this may not convergent in other matrix. Also, we can see that when $w = 1$, SOR method is equals to Gauss-Seidel Method.

(3) Verify the Shern-Morrison-Woodbury formula. If $\widetilde{\mathbf{A}} = \mathbf{A} + \mathbf{u}\mathbf{w}^T$, then

$$\widetilde{\mathbf{A}}^{-1} = \mathbf{A}^{-1} - \frac{1}{1+\mathbf{w}^T\mathbf{A}^{-1}\mathbf{u}}\mathbf{A}^{-1}\mathbf{u}\mathbf{w}^T\mathbf{A}^{-1}$$

Proof: First we construct the formula:

$$(I_n + \mathbf{A}^{-1}\mathbf{u}\mathbf{w}^T)(I_n - \frac{\mathbf{A}^{-1}\mathbf{u}\mathbf{w}^T}{I_n + \mathbf{w}^T\mathbf{A}^{-1}\mathbf{u}})$$

Then we verify this formulary equals to $\mathbf{I}$

$$\begin{aligned}
&(I_n + \mathbf{A}^{-1}\mathbf{u}\mathbf{w}^T)(I_n - \frac{\mathbf{A}^{-1}\mathbf{u}\mathbf{w}^T}{I_n + \mathbf{w}^T\mathbf{A}^{-1}\mathbf{u}}) \\
&= I_n + \mathbf{A}^{-1}\mathbf{u}\mathbf{w}^T - \frac{\mathbf{A}^{-1}\mathbf{u}\mathbf{w}^T}{I_n + \mathbf{w}^T\mathbf{A}^{-1}\mathbf{u}} - \frac{(\mathbf{A}^{-1}\mathbf{u}\mathbf{w}^T)^2}{I_n + \mathbf{w}^T\mathbf{A}^{-1}\mathbf{u}} \\
&= I_n + \mathbf{A}^{-1}\mathbf{u}\mathbf{w}^T - \frac{\mathbf{A}^{-1}\mathbf{u}(I + \mathbf{w}^T\mathbf{A}^{-1}\mathbf{u})\mathbf{w}^T}{I_n + \mathbf{w}^T\mathbf{A}^{-1}\mathbf{u}} \\
&= I_n
\end{aligned}$$

Hence,

$$(I_n + \mathbf{A}^{-1}\mathbf{u}\mathbf{w}^T)^{-1} = (I_n - \frac{\mathbf{A}^{-1}\mathbf{u}\mathbf{w}^T}{I_n + \mathbf{w}^T\mathbf{A}^{-1}\mathbf{u}})$$

Note that $\widetilde{\mathbf{A}}^{-1} = A \cdot (I_n + \mathbf{A}^{-1}\mathbf{u}\mathbf{w}^T)$, therefore:

$$\begin{aligned}
\widetilde{\mathbf{A}}^{-1} &= (I_n + \mathbf{A}^{-1}\mathbf{u}\mathbf{w}^T)^{-1} \cdot \mathbf{A}^{-1} \\
&= (I_n - \frac{\mathbf{A}^{-1}\mathbf{u}\mathbf{w}^T}{I_n + \mathbf{w}^T\mathbf{A}^{-1}\mathbf{u}}) \cdot \mathbf{A}^{-1} \\
&= \mathbf{A}^{-1} - \frac{1}{1+\mathbf{w}^T\mathbf{A}^{-1}\mathbf{u}}\mathbf{A}^{-1}\mathbf{u}\mathbf{w}^T\mathbf{A}^{-1}
\end{aligned}$$

(4) Consider

$$x_1^2 + x_2^2 = 1$$
$$(x_1 - 1)^2 + x_2^2 = 1$$

(a) Implement Newton's Method and find all solutions.
(b) Implement Broyden I Method and use $B_0 = I$ to find all solutions.
(c) Implement Broyden II Method and use $H_0 = I$ to find all solutions.

Table 4-1 Result and Iteration times when $x0 = (1,1)^T$

$Method$	$Newton$	$BroydenI$	$BroydenII$
$x1$	0.500000000000000	0.500000000000000	0.500000000000000
$x2$	0.866025403784439	0.866025403784439	0.866025403784439
$times$	5	12	12

Table 4-2 Result and Iteration times when $x0 = (1,-1)^T$

$Method$	$Newton$	$BroydenI$	$BroydenII$
$x1$	0.500000000000000	0.500000000000000	0.500000000000000
$x2$	-0.866025403784439	-0.866025403784439	-0.866025403784439
$times$	5	9	9

Also, when use $x0 = (0,0)^T$ or $x0 = (1,0)^T$, there will be errors.