Assignment 1

Mariam Walaa

March 23, 2024

Question 2

For the n by n matrix with entries $A_{ij} = \frac{5}{i+2j-1}$, set $x = [1, 1, ..., 1]^T$ and $b = Ax$. Use the Matlab's backslash command to compute $\bar{x}$, the double-precision computed solution. Find the infinity norm of the forward error, $||x - \bar{x}||_\infty$, and of the backward error, $||A\bar{x} - b||_\infty$. Compute the ratio between the relative forward error and the relative backward error of $Ax = b$ in the infinity norm and compare this ratio with the condition number of A for $n = 6$.

Solution:

```
% Set the size of the matrix
n = 6;

% Initialize the matrix A
A = zeros(n, n);

% Populate the matrix A with the given formula
for i = 1:n
    for j = 1:n
        A(i, j) = 5 / (i + 2*j - 1);
    end
end

% Create the vector x
x = ones(n, 1);

% Compute b = Ax
b = A * x;

% Use the backslash operator to solve Ax = b
x_bar = A \ b;

% Compute the infinity norm of the forward error ||x - x_bar||_inf
forward_error = norm(x - x_bar, inf);

% Compute Ax_bar
Ax_bar = A * x_bar;

% Compute the infinity norm of the backward error ||Ax_bar - b||_inf
backward_error = norm(Ax_bar - b, inf);

% Compute the relative forward error
relative_forward_error = forward_error / norm(x, inf);

% Compute the relative backward error
relative_backward_error = backward_error / norm(b, inf);

% Compute the condition number of A
condition_number_A = cond(A, inf);

% Compute the ratio between the relative forward error and the relative backward error
error_ratio = relative_forward_error / relative_backward_error;
```

```
Forward error: 6.4408e-10
Backward error: 4.4409e-16
Relative forward error: 6.4408e-10
Relative backward error: 7.2504e-17
```

```
Ratio between relative forward error and relative backward error: 8883341.6875
Condition number of A for n = 6: 70342013.9293
```

Here is what the system looks like for $n = 6$.

$$\begin{pmatrix} \frac{5}{2} & \frac{5}{4} & \frac{5}{6} & \frac{5}{8} & \frac{1}{2} & \frac{5}{12} \\ \frac{5}{3} & 1 & \frac{5}{7} & \frac{5}{9} & \frac{5}{11} & \frac{5}{13} \\ \frac{5}{4} & \frac{5}{6} & \frac{5}{8} & \frac{1}{2} & \frac{5}{12} & \frac{5}{14} \\ 1 & \frac{5}{7} & \frac{5}{9} & \frac{5}{11} & \frac{5}{13} & \frac{1}{3} \\ \frac{5}{6} & \frac{5}{8} & \frac{1}{2} & \frac{5}{12} & \frac{5}{14} & \frac{5}{16} \\ \frac{5}{7} & \frac{5}{9} & \frac{5}{11} & \frac{5}{13} & \frac{1}{3} & \frac{5}{17} \end{pmatrix} \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{pmatrix} = \begin{pmatrix} \frac{49}{8} \\ \frac{43024}{9009} \\ \frac{223}{56} \\ \frac{31012}{9009} \\ \frac{341}{112} \\ \frac{6161944556856429}{2251799813685248} \end{pmatrix}$$

The error ratio is $8,883,341.6875$ and the condition number is $70,342,013.9293$. The condition number is larger, implying that the system is ill-conditioned. It is also much larger than the error ratio.

Question 3(A)

Let A be an n by n matrix. Show that if A is symmetric and positive definite, then $a_{ii} > 0$ for any $i = 1, 2, ..., n$.

Solution:

Suppose A is an n by n, symmetric and positive definite matrix. Since A is positive definite, $x^T Ax > 0$ for all non-zero x. Choose $x = e_i$. Then $e_i^T Ae_i > 0$. Since A is symmetric, $A = A^T$. So, $e_i^T Ae_i = e_i^T A^T e_i = (Ae_i)^T e_i$. Let $y = Ae_i$. Then $(Ae_i)_i = a_{ii}$. So, $e_i^T Ae_i = y^T e_i = a_{ii} > 0$, as wanted.

Question 3(B)

Let A be an n by n matrix. Show that if A is symmetric and positive definite, then the matrix A is non-singular.

Solution:

Suppose A is an n by n, symmetric and positive definite matrix.

Assume, for contradiction, that A is singular. Then $det(A) = 0$. Since A is symmetric, it has real eigenvalues. Let λ be an eigenvalue of A with corresponding eigenvector $v \neq 0$. Thus, $Av = \lambda v$.

Furthermore, since A is positive definite, we have $v^T Av = v^T \lambda v = \lambda ||v||^2$. If A is singular, there must be at least one $\lambda = 0$, so $v^T Av = v^T \lambda v = 0$, contradicting $v^T Av > 0$ for all v. Therefore, A must be non-singular.

Question 3(C)

Show that if A is a strictly diagonally dominant matrix, then $(-A)$ is a strictly diagonally dominant matrix.

Solution:

Suppose A is a strictly diagonally dominant matrix with entries (a_{ii}). By definition then, $|a_{ii}| > \sum_{j\neq i} |a_{ij}|$. Consider $-A$ with entries $(-a_{ii})$. Then $|-a_{ii}| = |a_{ii}| > \sum_{j\neq i} |a_{ij}| = \sum_{j\neq i} |-a_{ij}|$. This is the definition of a strictly diagonally dominant matrix, therefore $-A$ is strictly diagonally dominant.

Question 4

Apply the Jacobi method to approximate in Matlab the solution of the sparse system for $Ax = b$, with three correct decimal places (forward error in the infinity norm) for $n = 100$. The correct solution is $[1, -1, 1, -1, ..., 1, -1]^T$. Give the number of iterations needed. The system is

$$\begin{bmatrix} 2 & 1 & 0 & \cdots & 0 & 0 \\ 1 & 2 & 1 & \cdots & 0 & 0 \\ 0 & 1 & 2 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 2 & 1 \\ 0 & 0 & 0 & \cdots & 1 & 2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_{n-1} \\ x_n \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \\ -1 \end{bmatrix}$$

Solution:

```
% Define the size of the system
n = 100;

% Create the sparse matrix A using diag function
A = diag(ones(1, n-1), -1) + 2*diag(ones(1, n), 0) + diag(ones(1, n-1), 1);

% Create the right-hand side vector b
b = [1; zeros(n-2, 1); -1];

% Initialize the solution vector x with zeros
x = zeros(n, 1);

% Set the tolerance for convergence
tolerance = 1e-6;

% Set the maximum number of iterations
maxIterations = 100000;

% Perform Jacobi iteration
for iter = 1:maxIterations
    xNew = zeros(n, 1);
    for i = 1:n
        xNew(i) = (b(i) - A(i, :)*x + A(i, i)*x(i)) / A(i, i);
    end

    % Check for convergence
    if norm(xNew - x, inf) < tolerance
        break;
    end

    % Update solution for the next iteration
    x = xNew;
end

% Display the result
disp(['Approximate solution: ', num2str(x', '%.3f')]);
disp(['Number of iterations needed: ', num2str(iter)]);
```

```
Approximate solution: 1.000-1.000 1.000-1.000 1.000-1.000 1.000-0.999 0.999-0.999 0.999-0.999 0.999-0.999 0
Number of iterations needed: 13276
```

Question 5

Solve the following problem $Ax = b$ as below by carrying out the conjugate gradient method by hand:

$$\begin{bmatrix} 2 & -1 \\ -1 & 2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

Solution:

Following the algorithm, we start with $x^{(0)} = \vec{0}$ and $d^{(0)} = \pi^{(0)} = \vec{b}$ and stop when $\pi(k) = \vec{0}$.

For the first iteration, we have $k = 1$:

$$\begin{aligned}
\alpha_1 &= \frac{(\pi^{(0)})^T \pi^{(0)}}{(d^{(0)})^T A d^{(0)}} \\
&= \frac{\begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix}}{\begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} 2 & -1 \\ -1 & 2 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix}} \\
&= \frac{1}{\begin{bmatrix} 2 & -1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix}} \\
&= \frac{1}{2} \\
x^{(1)} &= x^{(0)} + \alpha_1 \cdot d^{(0)} \\
&= \begin{bmatrix} 0 \\ 0 \end{bmatrix} + \frac{1}{2} \begin{bmatrix} 1 \\ 0 \end{bmatrix} \\
&= \begin{bmatrix} \frac{1}{2} \\ 0 \end{bmatrix} \\
\pi^{(1)} &= \pi^{(0)} - \alpha_1 \cdot A \cdot d^{(0)} \\
&= \begin{bmatrix} 1 \\ 0 \end{bmatrix} - \frac{1}{2} \begin{bmatrix} 2 \\ -1 \end{bmatrix} \\
&= \begin{bmatrix} 1 \\ 0 \end{bmatrix} - \begin{bmatrix} 1 \\ -\frac{1}{2} \end{bmatrix} \\
&= \begin{bmatrix} 0 \\ \frac{1}{2} \end{bmatrix} \\
\beta_1 &= \frac{(\pi^{(1)})^T \pi^{(1)}}{(d^{(0)})^T A d^{(0)}} \\
&= \frac{\begin{bmatrix} 0 & \frac{1}{2} \end{bmatrix} \begin{bmatrix} 0 \\ \frac{1}{2} \end{bmatrix}}{\begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix}} \\
&= \frac{1}{4} \\
d^{(1)} &= \pi^{(1)} + \beta_1 \cdot d^{(0)} \\
&= \begin{bmatrix} 0 \\ \frac{1}{2} \end{bmatrix} + \frac{1}{4} \begin{bmatrix} 1 \\ 0 \end{bmatrix} \\
&= \begin{bmatrix} \frac{1}{4} \\ \frac{1}{2} \end{bmatrix}
\end{aligned}$$

For the second iteration, we have $k = 2$:

$$
\begin{aligned}
\alpha_2 &= \frac{(\pi^{(1)})^T \pi^{(1)}}{(d^{(1)})^T A d^{(1)}} \\
&= \frac{\begin{bmatrix} 0 & \frac{1}{2} \end{bmatrix} \begin{bmatrix} 0 \\ \frac{1}{2} \end{bmatrix}}{\begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} 2 & -1 \\ -1 & 2 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix}} \\
&= \frac{\frac{1}{4}}{\begin{bmatrix} 0 & \frac{3}{4} \end{bmatrix} \begin{bmatrix} \frac{1}{4} \\ \frac{1}{2} \end{bmatrix}} \\
&= \frac{2}{3} \\
x^{(2)} &= x^{(1)} + \alpha_2 \cdot d^{(1)} \\
&= \begin{bmatrix} \frac{1}{2} \\ 0 \end{bmatrix} + \frac{2}{3} \begin{bmatrix} \frac{1}{4} \\ \frac{1}{2} \end{bmatrix} \\
&= \begin{bmatrix} \frac{2}{3} \\ \frac{1}{3} \end{bmatrix} \\
\pi^{(2)} &= \pi^{(1)} - \alpha_2 \cdot A \cdot d^{(1)} \\
&= \begin{bmatrix} 0 \\ \frac{1}{2} \end{bmatrix} - \frac{2}{3} \begin{bmatrix} 0 \\ \frac{3}{4} \end{bmatrix} \\
&= \begin{bmatrix} 0 \\ \frac{1}{2} \end{bmatrix} - \begin{bmatrix} 0 \\ \frac{1}{2} \end{bmatrix} \\
&= \begin{bmatrix} 0 \\ 0 \end{bmatrix}
\end{aligned}
$$

Since $\pi^{(2)} = \vec{0}$, $x^{(2)} = \begin{bmatrix} \frac{2}{3} \\ \frac{1}{3} \end{bmatrix}$ is the approximate (and exact) solution to $Ax = b$.

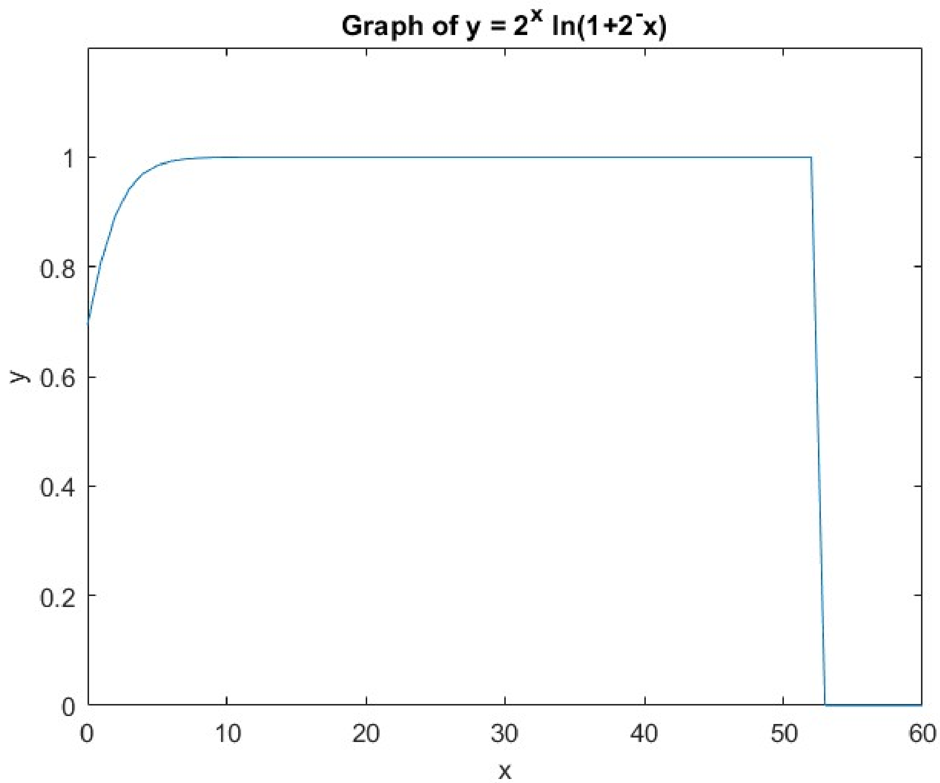
Question 6

Use Matlab to plot a graph of the function $y = 2^x \ln(1 + 2^{-x})$ for x going from 0 to 60.

What is the value of $\lim_{x\to\infty} y$?

Can you explain the graph you obtained?

Solution:



It is clear that the function approaches 1 as x goes to infinity, and reduces to zero after around $x = 52$. The limit of $y = 2^x \ln(1 + 2^{-x})$ as $x \to \infty$ is given by:

$$\begin{aligned}
\lim_{x\to\infty} 2^x \ln(1+2^{-x}) &= \lim_{x\to\infty} \frac{\ln(1+2^{-x})}{2^{-x}} \\
&= \lim_{x\to\infty} \frac{\ln(1+2^{-x})}{2^{-x}} \\
&\overset{\text{L'H}}{=} \lim_{x\to\infty} \frac{-\ln(2)2^{-x}}{-\ln(2)2^{-x}(1+2^{-x})} \\
&= \lim_{x\to\infty} \frac{1}{1+2^{-x}} \\
&= \frac{1}{1+0} \\
&= 1
\end{aligned}$$

As above, the value of the limit is 1. The graph I obtained shows that y approaches 1 until about 52, and then drops to zero. There is a numerical error issue since the graph uses a limited precision to plot the graph, causing errors when 2^{-x} is close enough to zero, which gives $\ln(1 + 0) = \ln(1) = 0$, multiplied by a very large number 2^x. This can be avoided by using higher precision or arbitrary precision to compute the graph after