UNIVERSITY OF NEW SOUTH WALES
School of Mathematics and Statistics

MATH2089 Numerical Methods and Statistics
Term 2, 2019

Numerical Methods Laboratory – Week 5 Solutions

1. Let $\mathbf{x} = (5, -4, 0, -6)^T$.

   (a) Calculate $\|\mathbf{x}\|_1$, $\|\mathbf{x}\|_2$ and $\|\mathbf{x}\|_\infty$ by hand.

   (b) Use MATLAB to check your answers.

   Answer

   (a) By hand

$$
\begin{aligned}
\|\mathbf{x}\|_1 &= \sum_{i=1}^{n} |x_i| = |5| + |-4| + |0| + |-6| = 15, \\
\|\mathbf{x}\|_2 &= \sqrt{\sum_{i=1}^{n} x_i^2} = \sqrt{|5|^2 + |-4|^2 + |0|^2 + |-6|^2} = \sqrt{77} \approx 8.7750, \\
\|\mathbf{x}\|_\infty &= \max_{i=1,\dots,n} |x_i| = \max\{5, 4, 0, 6\} = 6.
\end{aligned}
$$

   (b) Using MATLAB

```
x = [5; -4; 0; -6]
xn1 = norm(x, 1)
xn2 = norm(x, 2)
xni = norm(x, Inf)
```

2. Let

$$
A = \begin{bmatrix} 3 & -4 & 2 \\ 0 & 4 & -5 \\ 2 & -2 & 3 \end{bmatrix}, \qquad \mathbf{b} = \begin{bmatrix} 3 \\ 3 \\ 3 \end{bmatrix}.
$$

   (a) If possible, calculate $\|A\|_1$, $\|A\|_2$ and $\|A\|_\infty$ by hand.

   (b) Use MATLAB to check your answers.

   (c) Use MATLAB's backslash \ to solve the linear system $A\mathbf{x} = \mathbf{b}$.

   (d) Calculate the residual $\mathbf{r} = \mathbf{b} - A\mathbf{x}$. Is $\mathbf{r} = \mathbf{0}$?

   (e) Use the MATLAB function `lu` to calculate the LU factorization of A.

Answer

(a) By hand

$$\begin{aligned}
\|A\|_1 &= \max_{j=1\ldots,n} \sum_{i=1}^{m} |A_{ij}| = \max\{5, 10, 10\} = 10, \\
\|A\|_\infty &= \max_{i=1,\ldots,m} \sum_{j=1}^{n} |A_{ij}| = \max\{9, 9, 7\} = 9, \\
\|A\|_2 &= \max_{j=1,\ldots,n} \sqrt{|\lambda_j(A^T A)|} = \sqrt{\max\{1.3706, 8.5151, 77.1143\}} = 8.7815. \\
&\quad \text{as} \quad A^T A = \begin{pmatrix} 13 & -16 & 12 \\ -16 & 36 & -34 \\ 12 & -34 & 38 \end{pmatrix} \text{ has } \lambda_1 = 77.1143, \lambda_2 = 8.5151, \lambda_3 = 1.3706.
\end{aligned}$$

Calculating the eigenvalues of even a 3 by 3 matrix by hand is tedious.

(b) Using MATLAB

```
A = [3 -4 2; 0 4 -5; 2 -2 3]
norm(A, 1)
norm(A, inf)
ATA = A'*A
ev = eig(ATA)
max(sqrt(abs(ev)))
norm(A, 2)
```

(c) The MATLAB backslash \ operator gives (assuming A is defined as above)

```
b = [3; 3; 3]
x = A \ b
x =
    2.2000
    1.0000
    0.2000
```

(d) The MATLAB commands give

```
r = A*x - b
r =
   1.0e-15 *
      -0.4441
       0
      -0.4441
norm(r)
   6.2804e-16
```

The residual vector $\mathbf{r}$ is effectively zero (note the scaling factor of 10^{-15} which multiplies all elements), except for rounding error of the order of the relative machine precision ϵ.

Thus the calculated vector $\mathbf{x}$ is the (approximate) solution of $A\mathbf{x} = \mathbf{b}$.

(e) MATLAB's LU factorization produces

```
[L, U] = lu(A)
L =
    1.0000         0         0
         0    1.0000         0
    0.6667    0.1667    1.0000
U =
    3.0000   -4.0000    2.0000
         0    4.0000   -5.0000
         0         0    2.5000
```

so L is unit lower triangular ($L_{ii} = 1$ for all i and $L_{ij} = 0$ for all $j > i$) and U is upper triangular ($U_{ij} = 0$ for all $i > j$). You should also check that $LU = A$.

3. Consider the matrices

$$A = \begin{bmatrix} 0 & 3 & -2 \\ -1 & -4 & 2 \\ 5 & 14 & 26 \end{bmatrix}, \qquad B = \begin{bmatrix} -11/8 & -53/48 & -1/48 \\ 3/8 & 5/48 & 1/48 \\ 1/16 & 5/32 & 1/32 \end{bmatrix}$$

(a) Verify that $B = A^{-1}$ by showing that $AB = I$. What is $||AB - I||_1$?

(b) Calculate $\|A\|_1$, $\|A^{-1}\|_1$ and $\kappa_1(A)$ by hand and check using MATLAB.

(c) Calculate $\|A\|_\infty$, $\|A^{-1}\|_\infty$ and $\kappa_\infty(A)$ by hand and check using MATLAB.

(d) You want to calculate the 2-norm condition number $\kappa_2(A) = \|A\|_2 \, \|A^{-1}\|_2$.

   i. Use the MATLAB function `eig` to calculate the eigenvalues of A and A^{-1}

   ii. How are the eigenvalues of A and A^{-1} related?

   iii. Calculate $|\lambda_{\text{max}}|/|\lambda_{\text{min}}|$

   iv. Use MATLAB's `cond` to calculate $\kappa_2(A)$. Is this the same as in the previous part?

(e) Use row operations to reduce A to row-echelon form (by hand or using MATLAB just to do each row operation)

(f) The LU factorization produces a lower triangular matrix L and an upper triangular matrix U such that

$$PA = LU$$

where P is a permutation matrix reordering the rows of A (equations in a linear system).

   i. Use the MATLAB function `lu` to calculate the matrices P, L, and U.

   ii. Calculate $E = PA - LU$ and $\|E\|$.

Answer See the MATLAB M-file `tut05q3sol.m`.

```
% MATH2089: File = tut05q3sol.m
% Numerical Methods - Norms, Inverses, Condition Numbers and LU factorization

format compact
```

```
% Define the matrix A
A = [ 0  3 -2
     -1 -4  2
      5 14 26]

%B = inv(A)
% Provided B is the inverse of A, as AB = I = BA
B = [ -11/8   -53/48    -1/48
        3/8    5/48      1/48
       1/16    5/32     1/32]
E = A*B - eye(3)
Enrm1 = norm(E, 1)

% 1-norm is maximum of column sums of absolute values
Anrm1 = norm(A, 1)
AInrm1 = norm(B, 1)
kappa1 = Anrm1 * AInrm1
kappa1_chk = cond(A, 1)

% Infinity-norm is maximum of row sums of absolute values
Anrmi = norm(A, Inf)
AInrmi = norm(B, Inf)
kappai = Anrmi * AInrmi
kappai_chk = cond(A, Inf)

% Eigenvalues of inv(A) are 1/(eigenvlaues of A)
Aev = eig(A)
AIev = eig(B)
AIev_chk = 1./Aev

% Calcualte largest magnitude eigenvalue / smallest magnitude eigenvalue
evmaxmin = max(abs(Aev)) / min(abs(Aev))

% 2-norm of A is square root of largest eigenvalue of A'*A
% 2-norm of inv(A) is 1 over square root of smallest eigenvalue of A'*A
ATA = A'*A
% Sort eigenvalues in descending order: largest first, smallest last
ev = sort(eig(ATA), 'descend')
Anrm2 = sqrt(ev(1))
Anrm2_chk = norm(A, 2)
AInrm2 = 1/sqrt(ev(end))
AInrm2_chk = norm(B, 2)
kappa2 = sqrt(ev(1)/ev(end))
kappa2_chk = cond(A,2)
% NB As A is NOT symmetric, this is NOT the same as evmaxmin
```

```
% Row operations to get LU factorization
% Re-Order rows to get a non-zero pivot element and store in U
p = [2 1 3]
P = eye(3); P = P(p,:)
U = A(p, :)
% U(2,1) is already zero
% Make U(3,1) zero by subtracting a multiple of row 1
L31 = U(3,1)/U(1,1)
U(3,:) = U(3,:) - L31*U(1,:)
% Make U(3,2) zero by subtracting a multiple of row 2
L32 = U(3,2)/U(2,2)
U(3,:) = U(3,:) - L32*U(2,:)
% U is now upper triangular (row-echelon form)
L = eye(3);
L(3,1) = L31;
L(3,2) = L32
% Check
LUchk = P*A - L*U


% LU factroization: see help lu
% Is this the same permutation matrix P as above?
[L, U, P] = lu(A)
E = P*A - L*U
% Default norm is the 2-norm
Enrm = norm(E)
```

4. Consider the linear system $A\mathbf{x} = \mathbf{b}$, where you know A "exactly" and

```
norm(A) = 2.518
norm(A-A') = 9.3e-16
rcond(A) = 1e-12
min(eig(A)) = 1.3e-7
max(eig(A)) = 2.7e+6
```

(a) Is A symmetric?

(b) Show that if A is symmetric, then the eigenvalues of $A^T A$ are λ^2 where λ is an eigenvalue of A.

(c) What is the condition number $\kappa_2(A)$ of A using the 2-norm?

(d) Is this consistent with the given value of `rcond`?

(e) The elements of $\mathbf{b}$ come from measurements which are accurate to 4 significant decimal digits. Estimate the relative error in the computed solution to $A\mathbf{x} = \mathbf{b}$.

(f) If you want a computed solution that is accurate to 4 significant decimal digits, how accurate must the values of $\mathbf{b}$ be?

Answer

(a) As $\|A - A^T\|_2 = 9.3 \times 10^{-16} \approx \epsilon$, A is symmetric.

(b) If λ is an eigenvalue and $\mathbf{v}$ is an eigenvector of A, then $A\mathbf{v} = \lambda\mathbf{v}$. Thus

$$(A^T A)\mathbf{v} = A^T(A\mathbf{v}) = \lambda A^T\mathbf{v} = \lambda A\mathbf{v} = \lambda^2\mathbf{v}$$

so λ^2 is an eigenvalue of $A^T A$. Note that this used the fact that A is symmetric, that is $A^T = A$.

If A is not symmetric, then A and A^T have the same eigenvalues (why?), but not necessarily the same eigenvectors.

(c) As $\sqrt{\lambda^2} = |\lambda|$, the 2-norm condition number of A is

$$\kappa_2(A) = \|A\|_2 \, \|A^{-1}\|_2 = \frac{\max|\lambda_i(A)|}{\min|\lambda_i(A)|} = \frac{2.7 \times 10^6}{1.3 \times 10^{-7}} \approx 2 \times 10^{13}.$$

This is close to $1/\epsilon \approx 4.5 \times 10^{15}$ indicating the problem is very badly conditioned.

(d) The MATLAB function `rcond` gives the reciprocal of the 1-norm condition number, so

$$\kappa_1(A) = \frac{1}{\texttt{rcond(A)}} \approx 1 \times 10^{12}.$$

This is **not** inconsistent with the 2-norm condition number $\kappa_2(A) = 2 \times 10^{13}$ as the choice of norm can affect the precise value. Both indicate the problem is badly conditioned.

(e) Values which are accurate to 4 decimal digits after rounding have a **relative** error $\mathrm{re}\,(\mathbf{b}) \le 0.5 \times 10^{-4}$. All floating point numbers stored in computer have a relative error of at least ϵ, the relative machine precision. The relative error in the computed solution $\bar{\mathbf{x}}$ is estimated by

$$\begin{aligned} \mathrm{re}\,(\bar{\mathbf{x}}) &\approx \kappa(A)\,(\mathrm{re}\,(A) + \mathrm{re}\,(\mathbf{b})) \\ &= 2 \times 10^{13}\left(2 \times 10^{-16} + 5 \times 10^{-5}\right) \\ &= 10^9. \end{aligned}$$

Any relative error bigger than 1 indicates there are **no** significant digits in the computed solution!

(f) To get a computed solution $\bar{\mathbf{x}}$ accurate to 4 decimal digits you need $\mathrm{re}\,(\bar{\mathbf{x}}) \le 0.5 \times 10^{-4}$. This requires

$$\mathrm{re}\,(\bar{\mathbf{x}}) \approx 2 \times 10^{13}\left(2 \times 10^{-16} + \mathrm{re}\,(\mathbf{b})\right) \le 5 \times 10^{-5}.$$

However as $\mathrm{re}\,(\mathbf{b}) \ge 0$, the relative error in A means that the relative error in the computer solution is at least 4×10^{-3}. Thus the problem is so badly conditioned that no matter how accurately $\mathbf{b}$ is known, you cannot achieve more than around 2 significant decimal digits.

5. For a vector $\mathbf{x} \in \mathbb{R}^n$ the vector norms are related by

$$\begin{aligned} &\|\mathbf{x}\|_2 \le \|\mathbf{x}\|_1 \le \sqrt{n}\|\mathbf{x}\|_2, \\ &\|\mathbf{x}\|_\infty \le \|\mathbf{x}\|_2 \le \sqrt{n}\|\mathbf{x}\|_\infty, \\ &\|\mathbf{x}\|_\infty \le \|\mathbf{x}\|_1 \le n\|\mathbf{x}\|_\infty. \end{aligned}$$

(a) Let $\mathbf{e}_j \in \mathbb{R}^n$ be the jth unit vector (that is all elements are zero except the jth element which is 1). Calculate $\|\mathbf{e}_j\|_1$, $\|\mathbf{e}_j\|_2$ and $\|\mathbf{e}_j\|_\infty$. Hence show that the lower bounds above cannot be improved.

(b) Let $\mathbf{e} \in \mathbb{R}^n$ be the vector with all elements equal to 1. Calculate $\|\mathbf{e}\|_1$, $\|\mathbf{e}\|_2$ and $\|\mathbf{e}\|_\infty$. Hence show that the upper bounds above cannot be improved.

(c) The **relative error** in $\mathbf{x}$ measured using the ∞-norm is 3.4×10^{-7}.

   i. How many correct decimal places are there in $\mathbf{x}$ if $\|\mathbf{x}\|_\infty = 100$.

   ii. Does this apply to all elements of the vector $\mathbf{x}$?

Answer

(a) For the unit vectors $\mathbf{e}_j \in \mathbb{R}^n$, $\|\mathbf{e}_j\|_1 = 1$, $\|\mathbf{e}_j\|_2 = 1$ and $\|\mathbf{e}_j\|_\infty = 1$, showing that the unit vectors $\mathbf{e}_j$ achieve the lower bounds.

(b) For the vector $\mathbf{e} \in \mathbb{R}^n$ with each element equal to 1, $\|\mathbf{e}\|_1 = n$, $\|\mathbf{e}\|_2 = \sqrt{n}$ and $\|\mathbf{e}\|_\infty = 1$, showing that the vector $\mathbf{e}$ achieves the upper bounds.

(c) The relative error in $\mathbf{x}$ is $3.4 \times 10^{-7} = 0.34 \times 10^{-6} < 0.5 \times 10^{-6}$.

   i. Thus there are around 6 significant (decimal) figures in the elements of $\mathbf{x}$. If $\|\mathbf{x}\|_\infty \approx 100$, so the magnitude of the largest element is around 100, then the largest element will have at least 3 correct decimal places, eg $100.xxx$.

   ii. The infinity norm $\|\mathbf{x}\|_\infty = \max_{i=1,\ldots,n} |x_i|$ is the element of largest magnitude. Other elements may be more accurate.

6. (a) Some suggestions in MATLAB to test if $A \in \mathbb{R}^{n \times n}$ is equal to the zero matrix are

```
Ans1 = A == 0
Ans2 = max(abs(A)) == 0
Ans3 = max(max(abs(A))) == 0
Ans4 = norm(A, 1) == 0
```

For an n by n matrix what do each of the above commands produce? Try

$$A = \begin{bmatrix} 3 & -2 \\ 0 & -4 \end{bmatrix} \quad \text{and} \quad A = \begin{bmatrix} 3 & 0 & -4 \\ 0 & 0 & -4 \\ 1 & 0 & 0 \end{bmatrix}.$$

(b) Should the matrix `A = eps*randn(n,n)` be treated as zero?

(c) Propose a test for two matrices $B, C \in \mathbb{R}^{n \times n}$ to be equal, taking into account the size of B and C

Answer

(a) Let A be a n by n matrix. Then

   i. `Ans1 = A == 0` produces an n by n logical matrix, with all elements 0 or 1. The (i, j) element is the result of the logical comparison `A(i,j) == 0`, which gives 1 if $a_{ij} = 0$ and 0 otherwise. For example

$$A = \left[\begin{array}{cc} 3 & -2 \\ 0 & -4 \end{array} \right] \Longrightarrow \texttt{Ans1} = \left[\begin{array}{cc} 0 & 0 \\ 1 & 0 \end{array} \right]$$

ii. `Ans2 = max(abs(A)) == 0` first calculates the absolute value of all the elements of the array `A`. Then the `max` takes the maximum down all columns, so for the above matrix A

$$\texttt{max(abs(A))} = \begin{bmatrix} 3 & 4 \end{bmatrix}$$

The logical operator `==` then applies to each element of the vector, so `Ans2` is a 1 by n array of 0 and 1, with

$$\texttt{Ans2(j)} = \begin{cases} 1 & \text{if } \max_{i=1,\dots,n} |a_{ij}| = 0; \\ 0 & \text{otherwise.} \end{cases}$$

For example

$$A = \begin{bmatrix} 3 & 0 & -2 \\ 0 & 0 & -4 \\ 1 & 0 & 0 \end{bmatrix} \Longrightarrow \texttt{Ans2} = \begin{bmatrix} 0 & 1 & 0 \end{bmatrix}$$

iii. `Ans3 = max(max(abs(A))) == 0` takes the maximum of all the values in the row vector produced by taking the maximum down the columns of all the absolute values, that is the maximum magnitude element over all the elements of the array A. For the above matrix A this gives 4, which is not equal to 0, so `Ans3 = 0`.

iv. `Ans4 = norm(A,1) == 0` checks if the 1-norm of the matrix A is zero. The matrix 1-norm is the largest column sum of absolute values

$$\|A\|_1 = \max_{j=1,\dots,n} \sum_{i=1}^{n} |a_{ij}| = \texttt{max(sum(abs(A)))}$$

For any norm, $\|A\| = 0 \iff A = 0$, the zero matrix. Thus the result will be 1 if the matrix A is the zero matrix, 0 otherwise.

$$A = \begin{bmatrix} 3 & 0 & -2 \\ 0 & 0 & -4 \\ 1 & 0 & 0 \end{bmatrix} \Longrightarrow \texttt{norm(A,1)} = 6 \Longrightarrow \texttt{Ans4} = 0$$

(b) The matrix `A = eps*randn(n,n)` is an n by n matrix with each element $\epsilon\alpha$ where α is a sample from a standard normal distribution $N(0,1)$. Thus each element of A is around the relative machine precision ϵ, and we may have to accept that these just came from rounding error in computer arithmetic.

For example

```
>> A = eps*randn(4,4)
A =
  1.0e-015 *
   -0.0960   -0.2546    0.0727   -0.1306
   -0.3698    0.2644    0.0388    0.4848
    0.0278    0.2640   -0.0415   -0.0303
    0.0639   -0.0084    0.1612    0.0253
>> norm(A,1)
```

```
ans =
  7.9141e-016
>>
>> norm(A,2)
ans =
  6.8932e-016
>> A == 0
ans =
     0     0     0     0
     0     0     0     0
     0     0     0     0
     0     0     0     0
```

(c) Two matrices $B, C \in \mathbb{R}^{n\times n}$ are equal, $B = C \iff \|B - C\| = 0$ for some matrix norm. However errors in computer arithmetic mean that we cannot expect to get exactly 0 for the norm. A better test is

$$\|B - C\| \leq \tau$$

where the tolerance τ takes into account the relative machine precision, for example

```
tau = 10*eps*max([norm(B,1), norm(C,1), 1]);
chk = norm(B-C, 1) <= tau
```

Note that for large matrices the matrix 1 or infinity norms are quicker to calculate than the default 2 norm, which requires an eigenvalue calculation.