Homework #7
Tanner J. Evans

CS/Math 375-003

October 12, 2020

1. Find the norm $||A||_\infty$ of each of the following matrices:

   (a) The infinity norm of a matrix is simply the maximum value of each of the row's element's absolute values' sums. For part a:

$$A = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \quad \Bigg| \quad max((|1| + |2|), (|3| + |4|)) = max(3, 7) = 7$$

   (b) For part b:

$$A = \begin{bmatrix} 1 & 5 & 1 \\ -1 & 2 & -3 \\ 1 & -7 & 0 \end{bmatrix} \quad \Bigg| \quad max((|1| + |5| + |1|), (|-1| + |2| + |-3|), (|1| + |-7| + |0|)) = max(7, 6, 8) = 8$$

2. Find the infinity norm condition number of:

   (a) The infinity norm condition number can by found by calculating $||A|| * ||A^{-1}||$:

$$A = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \quad \Bigg| \quad ||A|| = 7 \quad \Bigg| \quad A^{-1} = \frac{1}{4-6} \begin{bmatrix} 4 & -2 \\ -3 & 1 \end{bmatrix} \quad \Bigg| \quad ||A^{-1}|| = 3 \quad \Bigg| \quad ||A|| * ||A^{-1}|| = 21$$

   (b)

$$A = \begin{bmatrix} 1 & 2.01 \\ 3 & 6 \end{bmatrix} \quad \Bigg| \quad ||A|| = 9 \quad \Bigg| \quad A^{-1} = \frac{1}{6-6.03} \begin{bmatrix} 6 & -2.01 \\ -3 & 1 \end{bmatrix} \quad \Bigg| \quad ||A^{-1}|| = \frac{8.01}{0.03} = 267 \quad \Bigg| \quad ||A|| * ||A^{-1}|| = 9 * 267 = 24$$

   (c) I am not sure how to deal with this. Since the determinant of A is zero, $\frac{1}{0}$ yields NaN, and A is not invertible.

3. (a) Find the condition number of the coefficient matrix in the system $\begin{bmatrix} 1 & 1 \\ 1+\delta & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 2 \\ 2+\delta \end{bmatrix}$:

$$A = \begin{bmatrix} 1 & 1 \\ 1+\delta & 1 \end{bmatrix} \quad \Bigg| \quad ||A|| = 2+\delta \quad \Bigg| \quad A^{-1} = \frac{1}{-\delta} \begin{bmatrix} 1 & -1 \\ -1-\delta & 1 \end{bmatrix} \quad \Bigg| \quad ||A^{-1}|| = \frac{2+\delta}{\delta} \quad \Bigg| \quad ||A|| * ||A^{-1}|| = \frac{(2+\delta)^2}{\delta}$$

   (b)

$$\frac{||x - \widetilde{x}||/||x||}{||b - \widetilde{b}||/||b||} = \frac{2/3}{\delta/2} = \frac{4}{3\delta}$$

   This seems innocuous enough, until we realize that when δ is sufficiently small, 3 becomes exceedingly small, and the overall value of the error magnification factor can become quite large.

4.
```
clear
n = [ 50 100 200 300 400 ];
results = zeros(length(n), 5);

for k = 1 : length(n)
    A = zeros(n(k), n(k));
    for j = 1 : n(k)
       for i = 1 : n(k)
           A(i, j) = sqrt(2*(i-j)^2 + n(k)/5);
       end
    end
    xExact = ones(n(k), 1);
    b = A*xExact;
    xComptd = A\b;
    relFwdErr = norm((xComptd - xExact), inf)/norm(xExact, inf);
    relBwdErr = eps;
    magFac = relFwdErr/relBwdErr;
    infNormCondA = cond(A, inf);

    results(k, 1) = n(k);
```

```
    results(k, 2) = relFwdErr;
    results(k, 3) = relBwdErr;
    results(k, 4) = magFac;
    results(k, 5) = infNormCondA;
end
format shortg
vars = {'n', 'rel fwd err', 'rel bwd err', 'mag. factor', ...
    'inf norm cond #'};
resultTable = table(results(:,1), results(:,2), results(:,3), ...
    results(:,4), results(:,5), 'VariableNames', vars)
```

```
>> hw7q4
resultTable =
  5×5 table
     n     rel fwd err    rel bwd err    mag. factor    inf norm cond #
    ___    ___________    ___________    ___________    _______________
     50    4.9553e-11     2.2204e-16     2.2317e+05       9.8251e+05
    100    2.7739e-09     2.2204e-16     1.2492e+07       6.1843e+07
    200     8.125e-07     2.2204e-16     3.6592e+09       1.2907e+10
    300    4.8093e-05     2.2204e-16     2.1659e+11       6.2039e+11
    400     0.0010159     2.2204e-16     4.5754e+12       1.4752e+13
```

The function that determines the elements of A is formulated to perform floating point operations. Each of these involves an error of epsilon. Each element of b is then a sum of n values, each with an error of epsilon. As we increase the size of the array, this error is increased. When we numerically compute x using these values, we perform a great many floating point operations on values that have already had epsilon introduced, and in the case of b, many times over. This results in a large error, and an increasingly large error for larger values of n.

5. (a) I prefer notating my Vandermonde matrices such that they appear consistent:

$$\begin{bmatrix} x_1^0 & x_1^1 & x_1^2 & \dots & x_1^{n-1} \\ x_2^0 & x_2^1 & x_2^2 & & \\ x_3^0 & x_3^1 & x_3^2 & & \\ \vdots & & & \ddots & \\ x_n^0 & & & & x_n^{n-1} \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \\ c_3 \\ \vdots \\ c_n \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_n \end{bmatrix}$$

(b)

```
% function v = vandermonde_hw7q5b(x)
% Supplies the Vandermonde matrix for any column matrix x.
function v = vandermonde_hw7q5b(x)
    v = zeros(length(x), length(x));
    for i = 1 : length(x)
       for j = 1 : length(x)
           v(i,j) = x(i)^(j-1);
       end
    end
end
```

(c)

```
results = zeros(4, 2);
for i = 1 : 4
    x = linspace(0, 1, i*10);
    V = vandermonde_hw7q5b(x);
    condNumInf = cond(V, inf);
    results(i, 1) = i*10;
    results(i, 2) = condNumInf;
end
```

```
vars = {'n', 'inf norm cond #'};
resultTable = table(results(:,1), results(:,2), 'VariableNames', vars)
```

```
>> vandermonde_hw7q5c
Warning: Matrix is close to singular or badly scaled. Results may be inaccurate. RCOND
=  2.300996e-17.
> In cond (line 46)
In vandermonde_hw7q5c (line 5)

Warning: Matrix is close to singular or badly scaled. Results may be inaccurate. RCOND
=  1.950861e-19.
> In cond (line 46)
In vandermonde_hw7q5c (line 5)

Warning: Matrix is close to singular or badly scaled. Results may be inaccurate. RCOND
=  9.944713e-20.
> In cond (line 46)
In vandermonde_hw7q5c (line 5)

resultTable =
  4×2 table
    n      inf norm cond #
    __     _______________
    10       4.8184e+07
    20       4.9796e+16
    30        5.054e+18
    40       1.7841e+19
```