

Math 426/CISC 410 08F, All Sections

R.J. Braun

Homework 5 Solutions, Hints and Answers

Problem 2.4.1(d). Take limits.

$$\lim_{n \rightarrow \infty} \frac{n + \sqrt{n}}{n} = \lim_{n \rightarrow \infty} 1 + \frac{1}{\sqrt{n}} = 1;$$

this implies that $n + \sqrt{n} \sim n$. Likewise,

$$\lim_{n \rightarrow \infty} \frac{n + 2\sqrt{n}}{n} = \lim_{n \rightarrow \infty} 1 + \frac{2}{\sqrt{n}} = 1;$$

this implies that $n + 2\sqrt{n} \sim n$. Since both are asymptotic to the same function n , they are asymptotic to each other.

One could also compute

$$\lim_{n \rightarrow \infty} \frac{n + \sqrt{n}}{n + 2\sqrt{n}} = \lim_{n \rightarrow \infty} \frac{1 + \frac{1}{\sqrt{n}}}{1 + \frac{2}{\sqrt{n}}} = 1;$$

this implies that they are asymptotic to each other.

Problem 2.4.3. Here is a code fragment that illustrates matrix multiplication:

```
for i = 1:n
  for j = 1:n
    C(i,j) = A(i,:)*B(:,j);
  end
end
```

The inner statement is a vector inner product requiring n multiplications and $n - 1$ additions, for $2n - 1$ flops. The total work summed over the loops is

$$\sum_{i=1}^n \sum_{j=1}^n (2n - 1) = (2n - 1) \sum_{i=1}^n \sum_{j=1}^n 1 = (2n - 1) \sum_{i=1}^n n = n(2n - 1) \sum_{i=1}^n 1 = n^2(2n - 1) \sim 2n^3.$$

Problem 2.4.4. Starting from (2.9), we get

$$2 \sum_{k=1}^{n-1} k^2 + 3 \sum_{k=1}^{n-1} k = 2 \frac{(n-1)(n)(2n-1)}{6} + 3 \frac{(n-1)n}{2} = \frac{2}{3}n^3 + \frac{1}{2}n^2 - \frac{7}{6}n$$

Dividing by n^3 and taking the limit $n \rightarrow \infty$ yield a limit of 2, showing that the flop count is $\frac{2}{3}n^3$.

The following script will do the plots. Try it!

```

% Script: Prob_2_4_4.m
% solves problem 2.4.4

close all;

n = logspace(1,4);
f_approx = 2*n.^3/3;
f_exact = f_approx + n.^2/2 - 7*n/6;

figure
loglog(n,f_approx,'-',n,f_exact,'--','LineWidth',2)
xlabel('n'); ylabel('count');
legend('Approximate','Exact','Location','NorthWest');
legend('boxoff');
title('Operation count, LU');

figure
semilogx(n,f_approx./f_exact,'LineWidth',2)
xlabel('n'); ylabel('Approximate/Exact');
title('Asymptotic behavior');

```

Problem 2.5.2. This little script will do the job.

```

% Script: Prob2_5_2.m
% try n = 4; example was for n=5.
% feel free to try other n

format compact;
format long;
n = 4;
A = magic(n)
x_exact = rand(n,1)
b = A*x_exact
x = A\b
disp('The residual is:')
res = b-A*x
disp(sprintf('The determinant is det(A) = %6.3e',det(A)))
disp('Row echelon form:')
rref(A)

```

The results follow; this matrix is singular but Matlab still finds an answer, and with residual $b-A*x$ being very small! We would expect one free parameter based on row echelon form (shown below) and an infinite number of solutions theoretically. More about this in class soon... But, you can compute the error, and the relative error is about 37%, so it is not a good answer really, even if the residual is small. The more poorly conditioned a matrix is, the larger the error will be compared to the residual.

```

A =
    16     2     3    13
     5    11    10     8
     9     7     6    12
     4    14    15     1
x_exact =
    0.814723686393179
    0.905791937075619
    0.126986816293506
    0.913375856139019
b =
    27.102009435129872
    22.614204751844920
    25.395487908497216
    18.758159965172993
Warning: Matrix is close to singular or badly scaled.
        Results may be inaccurate. RCOND = 1.306145e-017.
> In Prob2_5_2 at 11
x =
    0.478099542532198
   -0.104080494507322
    1.136859247876447
    1.250000000000000
The residual is:
res =
    1.0e-014 *
                0
   -0.355271367880050
                0
    0.355271367880050
The determinant is det(A) = -0.000e+000
Row echelon form:
ans =
     1     0     0     1
     0     1     0     3
     0     0     1    -3
     0     0     0     0

```

Problem 2.5.3. The backslash is theoretically the same premultiplying by inverse of the matrix on the left. Thus $\underline{U}\underline{L}$, given the precedence of operations from left to right for the same operation, is the same as $(U^{-1}L)^{-1}b = L^{-1}Ub$, which is not the solution to the original system $Ax = b$, or $LUx = b$.

Problem 2.5.5. (a) The following script does more than was requested. The third and fourth columns show results for `lufact` and the backslash operator. The line with `A(1,6)=1e10` is the answer for this part. There error is around 10^{-7} either way, so partial pivoting available

in the backslash operator is making no difference at all.

(b) The following script does more than requested, but does investigate the effect of scaled row pivoting for different sizes of the upper right element with other elements fixed. The error increases by about a factor of 100 for each order increase of 100 in the upper right element with out scaling. Warning messages occur for the last three values because the matrix is so badly conditioned. But, with scaled row pivoting, the answer comes out very well.

```
% Script: Solution of problem 2.5.5
% fancy solutions with loop for values of A(1,6)

n = 10;
big = logspace(2,20,n);
abserror = zeros(n);
err2 = abserror;  err3 = abserror;

A = diag(ones(6,1),0) + diag(ones(5,1),-1);
y = [0; 0.2; 0.4; 0.6; 0.8; 1];
D = eye(6);

for k = 1:n
    A(1,end) = big(k);  b = A*y;

    for jj = 1:6
        D(jj,jj) = 1/max(abs(A(jj,:)));
    end

    [L,U] = lufact(A);  z = forwardsub(L,b); x = backsub(U,z);
    abserror(k) = max(abs(y-x));

    % part (a)
    x2 = A\b; err2(k) = max(abs(y-x2));

    %part (b)
    x3 = (D*A)\(D*b);  err3(k) = max(abs(y-x3));
end

disp(' ') % a blank line
disp(' Problem 2.5.5          (a)          (b) ')
disp(' k    A(1,6)          lufact        with \        scaled \')
disp('-----')
for k=1:n
    disp(sprintf('%2.0f    %3.1e    %5.3e    %5.3e    %5.3e',...
        k,big(k),abserror(k),err2(k),err3(k)))
end
```

The output is as follows.

Warning: Matrix is close to singular or badly scaled.
 Results may be inaccurate. RCOND = 1.000000e-016.
 > In Prob2_5_5 at 24
 Warning: Matrix is close to singular or badly scaled.
 Results may be inaccurate. RCOND = 1.000000e-018.
 > In Prob2_5_5 at 24
 Warning: Matrix is close to singular or badly scaled.
 Results may be inaccurate. RCOND = 1.000000e-020.
 > In Prob2_5_5 at 24

| Problem 2.5.5 | | | (a) | (b) |
|---------------|----------|------------|------------|------------|
| k | A(1,6) | lufact | with \ | scaled \ |
| 1 | 1.0e+002 | 8.549e-015 | 8.549e-015 | 1.110e-016 |
| 2 | 1.0e+004 | 7.276e-013 | 7.276e-013 | 2.220e-016 |
| 3 | 1.0e+006 | 4.657e-011 | 4.657e-011 | 1.110e-016 |
| 4 | 1.0e+008 | 8.941e-009 | 8.941e-009 | 3.331e-016 |
| 5 | 1.0e+010 | 7.629e-007 | 7.629e-007 | 1.110e-016 |
| 6 | 1.0e+012 | 4.883e-005 | 4.883e-005 | 2.220e-016 |
| 7 | 1.0e+014 | 9.375e-003 | 9.375e-003 | 1.110e-016 |
| 8 | 1.0e+016 | 1.200e+000 | 1.200e+000 | 2.220e-016 |
| 9 | 1.0e+018 | 8.000e-001 | 8.000e-001 | 1.110e-016 |
| 10 | 1.0e+020 | 8.000e-001 | 8.000e-001 | 1.110e-016 |

Problem 2.6.2(a). By definition,

$$\|x\|_2 = \left[\sum_{i=1}^n |x_i|^2 \right]^{1/2}.$$

Let $|x_p| = \max_{1 \leq i \leq n} |x_i| = \|x\|_\infty$; then

$$\|x\|_2 = |x_p| \left[1 + \sum_{i=1, i \neq p}^n \left| \frac{x_i}{x_p} \right|^2 \right]^{1/2} \geq |x_p| = \|x\|_\infty.$$

Problem 2.6.3. Let $x, y \in R^n$. Then

$$|x^T y| = \left| \sum_{i=1}^n x_i y_i \right| \leq \sum_{i=1}^n |x_i y_i| = \sum_{i=1}^n |x_i| |y_i|.$$

Let $|y_p| = \max_{1 \leq i \leq n} |y_i| = \|y\|_\infty$; then

$$|x^T y| \leq \sum_{i=1}^n |x_i| |y_i| \leq \sum_{i=1}^n |x_i| \|y\|_\infty = \|y\|_\infty \left(\sum_{i=1}^n |x_i| \right) = \|y\|_\infty \|x\|_1.$$

Problem 2.6.4(a). Note that we are using the ∞ -norm, so unit vectors have ± 1 as the biggest element in the vector. $\|A\|_\infty = 2$, so we need to satisfy

$$\|Ax\|_\infty = \max \{ |2x_1|, |x_1 - x_2| \} = 2.$$

From the first case, we need $2x_1 = 2$ or $-2x_1 = 2$, so we get $x_1 = \pm 1$, and $x_2 = \alpha$, with $-1 \leq \alpha \leq 1$, will give $\|x\|_\infty = 1$. For the other component, we need to satisfy

$$x_1 - x_2 = 2 \text{ or } -(x_1 - x_2) = 2;$$

we can only satisfy these is if we choose $x_1 = -x_2 = 1$ in the first case or $x_1 = -x_2 = -1$. But, we can still get unit vectors with $|x_1 - x_2| < 2$ by using the solution from the first case, so we don't get any restriction from the second part if $x_1 = \pm 1$. Thus the set of all vectors that satisfy both is then $\{[1 \ -1]^T, [-1 \ 1]^T\} \cup \{[\pm 1 \ \alpha]\}$ with $-1 \leq \alpha \leq 1$; this union is just the solution to the first case. This corresponds to the two vertical sides of the "unit circle" (a square) for the ∞ -norm shown in Figure 2.2 of the text.

Problem 2.6.6. Here is the short way.

$$\|Px\|_2 = \sqrt{(Px)^T(Px)} = \sqrt{x^T P^T P x} = \sqrt{x^T I x} = \sqrt{x^T x} = \|x\|_2.$$

Now use the definition of the norm:

$$\|P\|_2 = \max_{\|x\|_2=1} \|Px\|_2 = \max_{\|x\|_2=1} \|x\|_2 = 1.$$

Problem 2.6.7. Start with $AA^{-1} = I$; then take norms of both sides: $\|I\| = \|AA^{-1}\| \leq \|A\|\|A^{-1}\|$. Now we need to know about $\|I\|$; use the definition of a norm to do this. $\|I\| = \max_{\|x\|=1} \|Ix\| = \max_{\|x\|=1} \|x\| = 1$. Then, divide by the norm of A to get $\|A^{-1}\| \geq 1/\|A\|$.