18.06 Problem Set 1 - Solutions

Due Wednesday, 12 September 2007 at 4 pm in 2-106.

Problem 0: from the book. *(50=10+10+10+10+10)*

(a) problem set 1.2, problem 8.

(a) F.

A Counterexample: $u = (1, 0, 0)$, $v = (0, 1, 0)$ and $w = (0, 1, 1)$.

(b) T.

Proof: $u \cdot (v + 2w) = u \cdot v + 2u \cdot w = 0 + 0 = 0$.

(c) T.

Proof: $\|u - v\| = \sqrt{(u - v) \cdot (u - v)} = \sqrt{u \cdot u - 2u \cdot v + v \cdot v} = \sqrt{1 + 1} = \sqrt{2}$.

(b) problem set 2.1, problem 4.

Answer: The planes in row picture, the column picture and the coefficient matrix are changed. The solution is not changed.

(c) problem set 2.2, problem 19.

(a) For any real number a, $\mathbf{u} = (x, y, z) + a(x - X, y - Y, z - Z)$ is another solution, since $(x - X, y - Y, z - Z)$ lies in the "null space", i.e. $A[x - X \ \ y - Y \ \ z - Z]^T = 0$, which implies $A\mathbf{u} = A[x \ y \ z]^T$.

NOTE: $(x, y, z) + (X, Y, Z)$ is not a solution, since $A[x \ y \ z]^T + A[X \ Y \ Z]^T = 2\mathbf{b}$!

(b) They also meet at all points in the line which is determined by the two points. In fact, when a changes, the points $(x, y, z) + a(x - X, y - Y, z - Z)$ in (a) forms the line through (x, y, z) and (X, Y, Z).

(d) problem set 2.3, problem 11.

For example: $\begin{pmatrix} 1 & 1 & 2 \\ 2 & 1 & 4 \\ 1 & 3 & 1 \end{pmatrix}$.

In general we have

$$\begin{pmatrix} 1 & a & b \\ c & 1 & d \\ e & f & 1 \end{pmatrix} \rightsquigarrow \begin{pmatrix} 1 & a & b \\ 0 & 1 - ac & d - bc \\ e & f & 1 \end{pmatrix} \rightsquigarrow \begin{pmatrix} 1 & a & b \\ 0 & 1 - ac & d - bc \\ 0 & f - ae & 1 - be \end{pmatrix}$$

$$\rightsquigarrow \begin{pmatrix} 1 & a & b \\ 0 & 1 - ac & d - bc \\ 0 & 0 & 1 - be - \frac{f - ae}{1 - ac}(d - bc) \end{pmatrix}$$

Thus the condition for the matrix to have negative pivot is $ac > 1$ and $(1-ac)(1-be) > (f-ae)(d-bc)$.

(e) problem set 2.3, problem 19.

$$PQ = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix}, \; QP = \begin{pmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}, \; P^2 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

There are many matrices whose square is the identity. For example, P, Q, $-P$, $-Q$, $R = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix}$, $-R$, $S = \begin{pmatrix} -1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix}$, $\begin{pmatrix} \cos t & \sin t & 0 \\ \sin t & -\cos t & 0 \\ 0 & 0 & 1 \end{pmatrix}$ etc.

Problem 1: elimination matrices. *(30=10+10+10)*

(a) Solve the following equations by elimination and back-substitution:

$$x + 3y - 4z = 7 \qquad (1)$$
$$2x - 4y + 2z = 0 \qquad (2)$$
$$3x + 19y + z = 2 \qquad (3)$$

Solution:

The elimination is given by

$$\begin{pmatrix} 1 & 3 & -4 & 7 \\ 2 & -4 & 2 & 0 \\ 3 & 19 & 1 & 2 \end{pmatrix} \overset{E_{21}}{\rightsquigarrow} \begin{pmatrix} 1 & 3 & -4 & 7 \\ 0 & -10 & 10 & -14 \\ 3 & 19 & 1 & 2 \end{pmatrix}$$
$$\overset{E_{31}}{\rightsquigarrow} \begin{pmatrix} 1 & 3 & -4 & 7 \\ 0 & -10 & 10 & -14 \\ 0 & 10 & 13 & -19 \end{pmatrix}$$
$$\overset{E_{32}}{\rightsquigarrow} \begin{pmatrix} 1 & 3 & -4 & 7 \\ 0 & -10 & 10 & -14 \\ 0 & 0 & 23 & -33 \end{pmatrix}$$

Thus back-substitution gives

$z = -33/23 = -1.4348$,

$y = \frac{10z-14}{10} = -4/115 = -0.0348$

and $x = 7 - 3y + 4z = 157/115 = 1.3652$.

(b) Let the above system be denoted by $Ax = b$. Find the three elimination matrices E_{21}, E_{31}, and E_{32} that put A into upper triangular form $U = E_{32}E_{31}E_{21}A$. Compute $M = E_{32}E_{31}E_{21}$.

From the solution to part (a) one get

$$E_{21} = \begin{pmatrix} 1 & 0 & 0 \\ -2 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \qquad E_{31} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -3 & 0 & 1 \end{pmatrix}, \qquad E_{32} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 1 \end{pmatrix}$$

and their product is

$$M = E_{32}E_{31}E_{21} = \begin{pmatrix} 1 & 0 & 0 \\ -2 & 1 & 0 \\ -5 & 1 & 1 \end{pmatrix}.$$

(c) Change a single number in the above equations so as to get *no* solutions. Then change another number to get *infinitely many* solutions.

The solution to the first question is not unique. To get a system with no solutions, we need to make the third row in the last step of elimination to be $[0\ 0\ 0\ *]$ with $*$ a nonzero number. One way to do this is to change the $(3,3)$-entry of the coefficient matrix from 1 to a, then after elimination as in part (a), the last row becomes $[0\ 0\ a+22\ \ -33]$. Thus if we take $a = -22$, the system will have no solution:

$$\begin{aligned} x + 3y - 4z &= 7 \qquad (1) \\ 2x - 4y + 2z &= 0 \qquad (2) \\ 3x + 19y\underline{-22}z &= 2 \qquad (3) \end{aligned}$$

In principle, you can change any single number in the 3×3 coefficient matrix to get a new system with no solution. To do so, for example, you can exchange rows, and exchange the name of unknowns x, y, z if necessary, such that the position you want to change is in at the $(3,3)$-entry. Then do the procedure above.

The solution to the second question is also not unique. Having a system with no solutions, if we want to get a system with infinitely many solutions by changing only 1 entry, we must change the vector b such that the last row of the matrix after elimination (the matrix U) is [0 0 0 0]. We may replace the 2 above by an arbitrary number c. Then after elimination the last row will be $[0\ 0\ 0\ c - 35]$. Thus we may change 2 to 35, i.e.

$$\begin{aligned} x + 3y - 4z &= 7 \qquad (1) \\ 2x - 4y + 2z &= 0 \qquad (2) \\ 3x + 19y - 22z &= \underline{35} \qquad (3) \end{aligned}$$

However, you may also change 7 or 0 by the same principle to get a system with infinitely many solutions.

Problem 2: Matlab, timings, etc.*(20=10+10)*

Warmup: Solve the equations in problem 1 using Matlab. You can do this by the commands:

```
>> A = [1 3 4; 2 -4 2; 3 19 1]
>> b = [7; 0; 2]
>> x = A \ b
```

Verify that you get the same answer as in 1(a), of course! Now, *time* how long Matlab takes to solve the equations, using the Matlab commands *tic* and *toc* (like a clock: tic toc tic toc):

```
>> tic; x=A \ b; toc
```

This prints the time in seconds for the command(s) between tic and toc. It should be quite a small number...in fact, this matrix is so small that you are mostly measuring the overhead of the Matlab interpreter rather than the solution of the equations per se.

(a) Now, let's solve larger systems. Much larger systems. To save the trouble of coming up with equations by hand, we'll let Matlab choose them at random using the *rand(m,n)* command, which creates a random $m \times n$ matrix:

```
>> A = rand(100,100);
>> b = rand(100,1);
>> tic; x = A \ b; toc
```

Notice the semicolons (;) after the commands: this suppresses the output, which is useful if you don't want to print out the 100×100 matrix A.

The above code was for 100×100. Now try doubling this to 200. Then to 400. Then to 800. Then to 1600. By what factor, on average, does the computation time increase each time you double the number of rows and columns? (You may want to repeat each computation a few times to make sure you get consistent timing results.)

I tried ten times for each case, and my average result is:

For n=100: time = 0.001 seconds

For n=200: time = 0.0044 seconds

For n=400: time = 0.0285 seconds

For n=800: time = 0.162 seconds

For n=1600: time = 0.983 seconds

For n=3200: time = 6.44 seconds

On average, the time increase by a factor about 6 when we double the number of rows and columns. However, the factor increases when the number n is bigger and bigger.

(b) Random linear equations like the ones above are very unlikely to have no solution. Construct a 2×2 or 3×3 set of equations by hand that (you think) have no solution. What does Matlab do when you give it `A \ b` with this A?

I used $\begin{pmatrix} 1 & 1 \\ 2 & 2 \end{pmatrix}$ and $b = (7, 2)$, and the out put is "Warning: matrix is singular to working precision" and "ans=(-inf, inf)".