Lecture 10
Some Facts About Linear Systems

Some inconvenient truths
In the last lecture we learned how to solve a linear system using Matlab. Input the following:

```matlab
> A = ones(4,4)
> b = randn(4,1)
> x = A \ b
```

As you will find, there is no solution to the equation $A\mathbf{x} = \mathbf{b}$. This unfortunate circumstance is mostly the fault of the matrix, $A$, which is too simple, its columns (and rows) are all the same. Now try

```matlab
> b = ones(4,1)
> x = [ 1 0 0 0]
> A*x
```

So the system $A\mathbf{x} = \mathbf{b}$ does have a solution. Still unfortunately, that is not the only solution. Try

```matlab
> x = [ 0 1 0 0]
> A*x
```

We see that this $x$ is also a solution. Next try

```matlab
> x = [ -4 5 2.27 -2.27]
> A*x
```

This $x$ is a solution! It is not hard to see that there are endless possibilities for solutions of this equation.

Basic theory
The most basic theoretical fact about linear systems is

**Theorem 1** A linear system $A\mathbf{x} = \mathbf{b}$ may have 0, 1, or infinitely many solutions.

Obviously, in most engineering applications we would want to have exactly one solution. The following two theorems show that having one and only one solution is a property of $A$.

**Theorem 2** Suppose $A$ is a square $(n \times n)$ matrix. The following are all equivalent:

1. The equation $A\mathbf{x} = \mathbf{b}$ has exactly one solution for any $\mathbf{b}$.
2. $\det(A) \neq 0$.
3. $A$ has an inverse.
4. The only solution of $A\mathbf{x} = \mathbf{0}$ is $\mathbf{x} = \mathbf{0}$.
5. The columns of $A$ are linearly independent (as vectors).
6. The rows of $A$ are linearly independent.

*If $A$ has these properties then it is called non-singular.*

On the other hand, a matrix that does not have these properties is called *singular.*
**Theorem 3** Suppose $A$ is a square matrix. The following are all equivalent:

1. The equation $Ax = b$ has $0$ or $\infty$ many solutions depending on $b$.
2. $\det(A) = 0$.
3. $A$ does not have an inverse.
4. The equation $Ax = 0$ has solutions other than $x = 0$.
5. The columns of $A$ are linearly dependent as vectors.
6. The rows of $A$ are linearly dependent.

To see how the two theorems work, define two matrices (type in \texttt{A1} then scroll up and modify to make \texttt{A2}):

$$A1 = \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix}, \quad A2 = \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 8 \end{pmatrix},$$

and two vectors:

$$b1 = \begin{pmatrix} 0 \\ 3 \\ 6 \end{pmatrix}, \quad b2 = \begin{pmatrix} 1 \\ 3 \\ 6 \end{pmatrix}.$$

First calculate the determinants of the matrices:

\texttt{> det(A1)} \quad \texttt{> det(A2)}

Then attempt to find the inverses:

\texttt{> inv(A1)} \quad \texttt{> inv(A2)}

Which matrix is singular and which is non-singular? Finally, attempt to solve all the possible equations $Ax = b$:

\texttt{> x = A1 \ b1} \quad \texttt{> x = A1 \ b2} \quad \texttt{> x = A2 \ b1} \quad \texttt{> x = A2 \ b2}

As you can see, equations involving the non-singular matrix have one and only one solution, but equation involving a singular matrix are more complicated.

**The residual vector**

Recall that the residual for an approximate solution $x$ of an equation $f(x) = 0$ is defined as $r = f(x)$. It is a measure of how close the equation is to being satisfied. For a linear system of equations we define the residual of an approximate solution $x$ by

$$r = Ax - b.$$

Notice that $r$ is a vector. Its size (norm) is an indication of how close we have come to solving $Ax = b$. 
Exercises

10.1 By hand, find all the solutions (if any) of the following linear system using the augmented matrix and Gaussian elimination:
\[ x_1 + 2x_2 + 3x_3 = 4, \]
\[ 4x_1 + 5x_2 + 6x_3 = 10, \quad \text{and} \]
\[ 7x_1 + 8x_2 + 9x_3 = 14. \]

10.2 (a) Write a well-commented MATLAB function program mysolvecheck with input a number \( n \) that makes a random \( n \times n \) matrix \( A \) and a random vector \( b \), solves the linear system \( Ax = b \), calculates the norm of the residual \( r = Ax - b \), and outputs that number as the error \( e \).

(b) Write a well-commented MATLAB script program that calls mysolvecheck 5 times each for \( n = 5, 10, 50, 100, 500, \) and 1000, records and averages the results, and makes a log-log plot of the average \( e \) vs. \( n \).

Turn in the plot and the two programs.