Homework: The Condition Number

Definition: The condition number of the matrix A is defined to be the maximum error magnification factor when solving $A\mathbf{x} = \mathbf{b}$ over all right-hand sides $\mathbf{b}$. We show below that

$$\mathrm{cond}(A) = \|A\| \|A^{-1}\|$$

Also below we look at some exercises that will illustrate the following **rule of thumb**: Expect to lose $\log_{10}(\mathrm{cond}(A))$ significant digits in solving $A\mathbf{x} = \mathbf{b}$.

One question that comes up: If we're worried about the matrix A being close to singular, why do we compute the condition number instead of the determinant? Hopefully we'll start to answer this question below.

Exercise Set 3

You might find the solutions to Exercise Set 1 and 2 useful when you're solving these.

1. Prove the formula for the condition number.

2. Let B_n be a square matrix with $1's$ along the diagonal, -1's above the diagonal and 0's below the diagonal (See the Matlab code attached to construct such a B). Show that, as n increases, the determinant of B_n is always 1, but the condition number increases exponentially with n.

3. Show that our rule of thumb is fairly good using the B_n above with $n = 40$. Some Matlab code is attached; we'll use a randomly chosen vector $\mathbf{b}$ for our right-hand side.

4. We might also have a sequence of matrices for which the determinant goes to zero, but the condition numbers stay good.

   Show that (in Matlab code), for increasing n,

   ```
   Dn=diag((1/10)ones(1,n));
   ```

   D_n has a determinant that goes to zero (as $n \to \infty$), but the condition number is always 1 (under the 1, 2 or infinity norms).

5. As one more example, re-do Example 2.12 on page 95 (Write a script file and verify the results).

Here is the Matlab code for Exercise 2.

```
function B=MatrixB(n)
B=eye(n);
for j=1:n
    for k=j+1:n
        B(j,k)=-1;
    end
end
```

Here is some code to test our rule of thumb:

```
B=MatrixB(40);
b=10*(round(100*rand(40,1))/100);
xc=inv(B)*b;
norm(B*xc-b)
log10(cond(B))
```