Fall 2014 - Math 401 Section 0401
Applications of Linear Algebra

Matlab Project 1 - Due: October 15, 2014

Note: To complete this project you have to download the files: `GE.m`, `ltrisol.m`, `utrisol.m`, `lbidisol.m`, `elim.m`, `partic.m`, and `nullbasis.m` from the website: `http://www.math.umd.edu/~eotarol1/teaching.html`.

You should complete each problem in a separate `m`-file and bring a copy of each problem to class. You can also use the new MATLAB© command `publish`.

To prevent MATLAB© from outputing large matrices and/or vectors, you should add ';' at the end of each MATLAB© sentence (type 'help ;'). Do not show the matrices and vectors in the papers you turn in. Show only the codes, outputs, your comments and answers.

Problem 1. (25 pts) In this problem we compare three different alternatives to compute the solution of a linear system $A\mathbf{x} = \mathbf{b}$. Let $n = 10$ and define the $n \times n$ tridiagonal matrix A and the n-vector $\mathbf{b}$ using the instructions

```
>> A = diag(2*ones(1,n)) - diag(ones(1,n-1),1) - diag(ones(1,n-1),-1);
>> b = [0:1:n/2-1 n/2-1:-1:0]';      % don't forget the '
```

Type `help ones` and `help diag` to learn how these commands work.

To solve the linear equation $A^5\mathbf{x} = \mathbf{b}$, we propose three different alternatives:

(a) The first alternative is to use the MATLAB© command "\":

```
>>  x = (A^5) \ b;
```

(b) The second one is based on the fact that solving $A^5\mathbf{x} = \mathbf{b}$ is equivalent to solve $A(A(A(A(A\mathbf{x})))) = \mathbf{b}$. Then, by solving the sequence of linear systems $A\mathbf{x}_1 = \mathbf{b}$, $A\mathbf{x}_2 = \mathbf{x}_1$ $A\mathbf{x}_3 = \mathbf{x}_2$, $A\mathbf{x}_4 = \mathbf{x}_3$, $A\mathbf{x} = \mathbf{x}_4$, the desired solution $\mathbf{x}$ can be obtained. Each linear system can be solved by using the \ command.

(c) Finally, the third alternative is through the computation of the LU decomposition of A by using the function `GE.m`. Then solve the five linear systems of (b) by using `lbidisol.m` and `ubidisol.m` **without** re-decomposing the matrix A.

Solve the system $A^5\mathbf{x} = \mathbf{b}$ using these three alternatives. To complete (c), you should write a MATLAB© function `x = ubidisol(u,f,b)` to solve the involved upper bidiagonal system.

Estimate the number of operations (flops) in each method as a function of n and compare. Draw conclusions.

Problem 2. (25 pts) Let

$$A = \begin{bmatrix} 1 & 2 & 3 \\ 2 & 4 & -5 \\ -1 & 3 & -3 \end{bmatrix}.$$

(a) Use the MATLAB© function `[L, U] = GE(A)` to compute the LU decomposition of A without pivoting. Explain the obtained results.

(b) Consider the following MATLAB© function to find the LU factorization of A with row exchanges:

```
function [L,U,piv] = GEpiv(A)
[n,n] = size(A);
piv=1:n;
for k=1:n-1
    [maxv,s]=max(abs(A(k:n,k)));
    q=s+k-1;
    piv([k,q])=piv([q,k]);
    A([k,q],:)=A([q,k],:);
   A(k+1:n,k) = A(k+1:n,k)/A(k,k);
   A(k+1:n,k+1:n) = A(k+1:n,k+1:n) - A(k+1:n,k)*A(k,k+1:n);
end
L = eye(n,n) + tril(A,-1);
U = triu(A);
```

In this code `piv` is a permutation vector. Explain how to find the permutation matrix P from `piv` such that $PA = LU$. Check that $PA = LU$.

(c) Let $\mathbf{b} = [5, 4, 3]^T$. Use `ltrisol.m`, `utrisol.m` and the permuted decomposition described above to solve the linear system $A\mathbf{x} = \mathbf{b}$.

Problem 3. (25 pts) There are some matrices that are difficult to work with, even with sophisticated pivoting strategies: *ill-conditioned* matrices. Such matrices are typically characterized by being "almost" singular. A famous example of an ill-conditioned matrix is the *Hilbert* matrix $H_n = (h_{ij})_{i,j=1}^n$ of order n, which is defined by

$$h_{ij} = \frac{1}{i+j-1} \qquad \forall i, j \in \{1, \cdots, n\}.$$

This matrix is nonsingular and has an explicit inverse. However, as n becomes larger, H_n becomes closer to being singular. The MATLAB© functions `hilb(n)` and `invhilb(n)` give H_n and H_n^{-1} respectively.

Given $\mathbf{b}_n = (1, 0, \ldots, 0)$, we want to solve $H_n\mathbf{x}_n = \mathbf{b}_n$.

(a) Solve for $n = 5, 10$ using the Matlab command "\", and call the computed result $\mathbf{x}_n^*$.

(b) Compute the exact solution $\mathbf{x}_n = H_n^{-1}\mathbf{b}_n$, the *error* $\mathbf{e}_n = \mathbf{x}_n - \mathbf{x}_n^*$, and the residual $\mathbf{r}_n = \mathbf{b}_n - H_n\mathbf{x}_n^*$.

(c) Find the *condition number* of H_n, which is denoted by $\mathrm{cond}(H_n)$, using the command `cond`. This number is an estimate of the expected relative accuracy of the solution: if $\mathrm{cond}(H_n) \approx 10^t$ with $t \geq 0$ then, the number of correct decimal digits in the solution is expected to be $16 - t$. How many correct decimal digits do you expect for $n = 5$ and $n = 10$?

Problem 4. (25 pts) Let

$$A = \begin{bmatrix} 1 & 2 & 3 & 4 & 5 \\ 6 & 12 & 8 & 9 & 10 \\ 2 & 4 & 4 & 5 & 6 \end{bmatrix}$$

(a) Use `rref` to find the reduced row echelon form R of A together with the pivot columns of A.

(b) Use `elim.m` to find the reduced row echelon form R of A, and the elimination matrix E. The latter satisfies $R = EA$.

(c) Use the results of (a) and (b) to find a basis for the solution space of $A\mathbf{x} = 0$.

(d) Use `nulbasis.m` to find a basis for $N(A) := \{\mathbf{x} \in \mathbb{R}^5 : A\mathbf{x} = 0\}$. Relate with (c).

(e) What is the general solution to the linear system $A\mathbf{x} = 0$?

(f) Use `rank` to find the rank of A. Relate to the dimensions of A and $N(A)$.

(g) Find the condition on $\mathbf{b} = [b_1, b_2, b_3]^T$ that ensures $A\mathbf{x} = \mathbf{b}$ has at least one solution. To do this perform row reduction on the augmented matrix $[A \,|\, \mathbf{b}]$ through hand computations.

(h) Use `partic.m` to find a particular solution to $A\mathbf{x} = [0, 5, 1]^T$. Does $[0, 5, 1]^T$ satisfy the condition of (g)?

(i) Use the result in (e) and (h) to write the general solution to $A\mathbf{x} = [0, 5, 1]^T$.