AMS 526 Homework 2

Due: Wednesday 09/24 in class

1. (10 points) Let $P \in \mathbb{R}^{n \times n}$ be an orthogonal projection matrix (projector).
   (a) Show that $I - P$ is also an orthogonal projection matrix.
   (b) Show that $I - 2P$ is an orthogonal matrix.

2. (10 points) Consider the matrices
   
   \[
   A = \begin{bmatrix}
   1 & 0 \\
   0 & 1 \\
   1 & 0
   \end{bmatrix}, \quad
   B = \begin{bmatrix}
   1 & 2 \\
   0 & 1 \\
   1 & 0
   \end{bmatrix}.
   \]

   Answer the following questions by hand calculation.
   (a) What is the orthogonal projector $P$ onto range($A$), and what is the image under $P$ of the vector $(1, 2, 3)^T$?
   (b) Answer the same questions for $B$.

3. (10 points) Consider the matrix
   
   \[
   A = \begin{bmatrix}
   2 & 11 \\
   10 & 5
   \end{bmatrix}.
   \]

   (a) By hand calculation, determine a real SVD of $A$ in the form of $A = U\Sigma V^T$. The SVD is not unique, so find the one that have the minimum number of minus signs in $U$ and $V$.
   (Hint: Compute the eigenvalue decomposition of $A^TA$ by hand.)
   (b) List the singular values, left singular vectors, and right singular vectors of $A$.

4. (10 points) Let $A = \begin{bmatrix} I & B \\ B^T & I \end{bmatrix}$, where $B \in \mathbb{R}^{m \times n}$ with $\|B\|_2 < 1$. Let $B = U\Sigma V^T$ denote the singular value decomposition of $B$.
   (a) Show that the left singular vectors of $A$ are the columns of the matrix
   
   \[
   X = \frac{1}{\sqrt{2}} \begin{bmatrix} U & U \\ V & -V \end{bmatrix}.
   \]
   (b) Show that the condition number of $A$ in 2-norm is
   
   \[
   \kappa(A) = 1 + \frac{\|B\|_2}{1 - \|B\|_2}.
   \]

5. (15 points) Gaussian elimination can be used to compute the inverse $A^{-1}$ of a nonsingular matrix $A \in \mathbb{R}^{n \times n}$, though it is rarely really necessary to do so.
(a) Describe an algorithm for computing $A^{-1}$ by solving $n$ systems of equations, and show that its asymptotic operation count is $\frac{8}{3}n^3$ flops.

(b) Describe a variant of your algorithm, taking advantage of sparsity, that reduces the operations count to $2n^3$ flops.

(c) Suppose one wishes to solve $k$ systems of equations $Ax_j = b_j$, or equivalently, a block system $AX = B$ with $B \in \mathbb{R}^{n \times k}$. What is the asymptotic operation count (a function of $n$ and $k$) for doing this (i) directly from the LU factorization and (ii) with a preliminary computation of $A^{-1}$?

6. (15 points) Suppose $A \in \mathbb{R}^{n \times n}$ is strictly column dominant, which means that for each $k$,

\[ |a_{kk}| > \sum_{j \neq k} |a_{jk}|. \]

Show that if Gaussian elimination with partial pivoting is applied to $A$, no row interchanges take place.

7. (30 points) Write a routine in MATLAB to estimate the condition number of a real matrix $A$ using 1-norm. You will need to compute $\|A\|_1$, which is easy, and estimate $\|A^{-1}\|_1$, which is more challenging. One way to estimate $\|A^{-1}\|_1$ is to take it as the ratio $\|z\|_1/\|y\|_1$, where $z$ is the solution to $Az = y$ and $y$ is picked by some heuristic to maximize the ratio.

We choose $y$ as the solution to the system $A^T y = c$, where $c$ is a vector each of whose components is $\pm 1$, with the sign for each component chosen by the following heuristic. Using the factorization $PA = LU$ (you may use MATLAB’s routine `lu`), the system $A^T y = c$ is solved in two stages, successively solving the triangular systems $U^T v = c$ and $L^T Py = v$. At each step of the first triangular solution, choose the corresponding component of $c$ to be 1 or $-1$, depending on which will make the resulting component of $v$ larger in magnitude. You will need to write a custom triangular solution routine to implement the selection of $c$. Then solve the second triangular system $L^T Py = v$ in the usual way for $y$, for which you can use MATLAB’s backslash “\” operator. The idea here is that any ill-conditioning in $A$ will be reflected in $U$, resulting in a relatively large $v$. The relative well-conditioning unit triangular matrix $L$ will then preserve this relationship, resulting in relatively large $y$.

Test your program on the Hilbert matrix of order $n = 2, 3, \ldots, 12$, which has entries $h_{ij} = 1/(i + j - 1)$. For example, a $3 \times 3$ Hilbert matrix has entries

\[
\begin{bmatrix}
1 & 1/2 & 1/3 \\
1/2 & 1/3 & 1/4 \\
1/3 & 1/4 & 1/5 \\
\end{bmatrix}.
\]

To check the quality of your estimates, compute $A^{-1}$ explicitly using the LU factorization that was already computed and then compute the condition number $\|A\|_1 \|A^{-1}\|_1$. Plot the estimated condition numbers and the explicitly computed condition numbers for the Hilbert matrix of order $n = 2, 3, \ldots, 12$ (using the horizontal axis for $n$ and the vertical axis for the condition numbers). Compare the required flops of the two approaches (i.e., estimation and explicit computation), and also plot their running times for different $n$. To obtain reliable timing of a procedure, you may need to run it repeatedly for hundreds of iterations and then take the average (use the tic() and toc() functions outside the iterations).

Submit your programs, the plots, and a brief discussion of your results.

Remarks: The condition-number estimator is from [1]. Modern software (e.g., MATLAB) estimates condition numbers using more advanced methods in [2, 3]. (You are not required to read these articles.)
References

