Programming Assignment 3 Foundations of Computational Mathematics 1 Fall 2024

The solutions are due on Canvas by 11:59 PM on Tuesday November 26, 2024

Part 1: Behavior of RF, SD, and CG in Eigen Coordinate System

This part of the assignment concerns implementing and observing the performance of Richardson's method (often called Richardson's First Order Stationary method, hence RF), Steepest Descent (SD), and Conjugate Gradient (CG). This part will exploit the fact that for symmetric positive definite $A = Q\Lambda Q^T$, where the columns of Q are the n orthonormal eigenvectors of A and Λ is a positive real diagonal matrix with the eigenvalues of A on the diagonal, the behavior of the methods for a problem can be seen in the coordinates defined by the eigenvectors. In practice, of course, this decomposition is not known but for developing understanding, analyzing and demonstrating the relative performance of these methods, it is sufficient to consider systems of the form $\Lambda\tilde{x} = \tilde{b}$ related to $Ax = b$ by $\tilde{x} = Q^T x$ and $\tilde{b} = Q^T b$ since, as seen in the notes, study questions, and homework problems $\|A^{-1}b - x_k\|_2 = \|\Lambda^{-1}\tilde{b} - \tilde{x}_k\|_2$ for the iterations defined by all three methods. (This is not true, in general, for the other stationary methods.)

The notes and lectures (Sets 7, 8, 9, and 10; Study Question Sets 4 and 5; and written problems from Homework 3) have discussed and derived the bounds on the behavior of the damping of the error and residuals. For RF and SD each step must reduce the error norm by at the factor

$$\frac{\kappa - 1}{\kappa + 1}$$

with SD's single step reduction being no less and typically better than RF. CG's one step damping factor is between

$$\frac{\sqrt{\kappa} - 1}{\sqrt{\kappa} + 1} \text{ and } 2\,\frac{\sqrt{\kappa} - 1}{\sqrt{\kappa} + 1}.$$

For CG, the performance, in exact arithmetic, is also related to the number of distinct eigenvalues, e.g., the identity matrix as a single eigenvalue, 1, with multiplicity n, as well as the eccentricity of the spectrum as measured by κ.

Therefore, part of your task is comparing your observed reductions in the norms of the error and resdiual to these bounds and to results across methods. This implies that you will be working with problems for which you know the solution. Your discussion for this part of the assignment should generate matrices of various sizes and for each matrix, Λ, you should generate a solution vector, $\tilde{x}$, and the associated righthand side $\tilde{b} = \Lambda\tilde{x}$. Additionally, for each problem $\Lambda\tilde{x} = \tilde{b}$ you should generate a set of initial guesses and run all three methods with each initial guess for each problem, for each matrix. Your main task is to collect the appropriate information from each of these runs and to analyze and present it in such a way that it clearly probes the behavior of each method relative to each expectation and the relative performances of the three methods, e.g., was SD typically better than RF?; Was CG typically better than both SD and RF? What is the characteristic of problems where methods achieved good or bad performance? All of these questions should be based on a clear understanding of the course material.

The experiments in this section can be run over fairly large problems since you do not need a large sparse or dense A; the problem is defined by three vectors, the eigenvalues $\lambda_1, \ldots, \lambda_n$, the solution $\tilde{x}$ and the righthand side vector $\tilde{b}$, i.e., $3n$ data. The work space is also $O(n)$ for all three methods. Computationally, the methods differ somewhat in the number of operations per step and you should determine this number as a function of n and compute and compare the total work defined as the number of operations times the number of iterations. You may find it useful to run at least some of your experiments interactively without a termination criterion in order to generate some insight into the peformance of the methods. However, at some point the large parameterized set of experiments should use a termination criterion of the ratio $\|r_k\|/\|r_0\| \leq \epsilon$ where $\epsilon = 10^s$ for some integer s that indicates by how many digits the residual norm has been reduced. You will have to probe this value to see what is effective in allowing you to make appropraite observations and conclusions about the performance of the methods.

The matrices Λ should be generated by a combination of problems with specific spectra, e.g.,

- all n eigenvalues the same;
- k distinct eigenvalues with the multiplicties of each chosen deterministically or randomly;
- k distinct eigenvalues that are used as the mean of a normally distributed selection of a "cloud" of eigenvalues around each distinct eigenvalue (the number in each must also be chosen as the multiplicities in the previous item);
- from a uniform distribution between parameters the two extreme eigenvalues included in the problem, λ_{min} and λ_{max}, that you chooose in order to set κ;
- from a normal distribution between parameters the two extreme eigenvalues included in the problem, λ_{min} and λ_{max}, that you chooose in order to set κ with a selected mean and variance.

Consider carefully and express clearly the predictions you are testing with each set of data from the experiments relative to what is known from the class material.

Part 2: Behavior of Stationary Methods Jacobi, Gauss-Seidel, and Symmetric Gauss-Seidel

For this part of the assignment, you should implement these stationary methods in a single code with a selectable choice of preconditioner that defines the method as given in Set 10 of the class notes and Study Questions Set 5. Since these methods are sensitive to the ordering of the variables in the vector in the coordinate system defined by the matrix A in $Ax = b$, problems cannot be run in the eigen coordinate system. For this part, you may implement a version of the code that uses a dense matrix and the associated computational primitives. You will modify this for large sparse matrices in the the third and final part of the assignment.

For this part of the assignment you are required to compare the predicted behavior of the methods based on a matrix norm and spectral radius of the iteration matrix G associated with each problem/method combination. As in part 1, you should generate many solutions and righthand side vectors for each matrix, and many initial guesses for each problem. These methods converge for problems other than symmetric positive definite, although that has not been discussed in detail in the notes, and there is a vast literature characterizing the properties of associated matrices.

The following examples contain matrices with various properties.

$$A_0 = \begin{pmatrix} 3 & 7 & -1 \\ 7 & 4 & 1 \\ -1 & 1 & 2 \end{pmatrix} \quad A_1 = \begin{pmatrix} 3 & 0 & 4 \\ 7 & 4 & 2 \\ -1 & -1 & 2 \end{pmatrix} \quad A_2 = \begin{pmatrix} -3 & 3 & -6 \\ -4 & 7 & -8 \\ 5 & 7 & -9 \end{pmatrix}$$

$$A_3 = \begin{pmatrix} 4 & 1 & 1 \\ 2 & -9 & 0 \\ 0 & -8 & -6 \end{pmatrix} \quad A_4 = \begin{pmatrix} 7 & 6 & 9 \\ 4 & 5 & -4 \\ -7 & -3 & 8 \end{pmatrix}$$

$$A_5 = \begin{pmatrix} 6 & -2 & 0 \\ -1 & 2 & -1 \\ 0 & -6/5 & 1 \end{pmatrix} \quad A_6 = \begin{pmatrix} 5 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -3/2 & 1 \end{pmatrix}$$

$$A_7 = \begin{pmatrix} 4 & -1 & 0 & 0 & 0 & 0 & 0 \\ -1 & 4 & -1 & 0 & 0 & 0 & 0 \\ 0 & -1 & 4 & -1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 4 & -1 & 0 & 0 \\ 0 & 0 & 0 & -1 & 4 & -1 & 0 \\ 0 & 0 & 0 & 0 & -1 & 4 & -1 \\ 0 & 0 & 0 & 0 & 0 & -1 & 4 \end{pmatrix}$$

$$A_8 = \begin{pmatrix} 2 & -1 & 0 & 0 & 0 & 0 & 0 \\ -1 & 2 & -1 & 0 & 0 & 0 & 0 \\ 0 & -1 & 2 & -1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 2 & -1 & 0 & 0 \\ 0 & 0 & 0 & -1 & 2 & -1 & 0 \\ 0 & 0 & 0 & 0 & -1 & 2 & -1 \\ 0 & 0 & 0 & 0 & 0 & -1 & 2 \end{pmatrix}$$

Consider first taking the b for a solution x that is all 1's and $x_0 = 0$ and iterated until $\|x_k - x\|_2 / \|x\|_2 \leq 10^{-6}$. For each matrix, compute, using an appropriate numerical linear algebra library, e.g., Matlab or LAPACK, the spectral radii: $\rho(G_J)$, $\rho(G_{GS})$ and $\rho(G_{SGS})$; the matrix norms $\|G_J\|_2$, $\|G_{GS}\|_2$, and $\|G_{GS}\|_2$; and the number of iterations: k_J, k_{GS}, and k_{SGS} required to satisfie the termination condition (or record lack of convergence).

Next, take each of the matrices and generate multiple known solutions x and corresponding b vectors. Solve each problem and compare the performance observed to the behavior seen with the reference problem above.

Part 3: Behavior of All Methods for Large Sparse Symmetric Positive Definite Matrix Problems

For this part of the problem you must implement a sparse symmetric matrix vector product as described in the lectures and notes based on the assumption that $A = D - L - L^T$ where D is the positive diagonal matrix comprising the diagonal elements of A stored as a separate vector, $-L$ is the strictly lower triangular sparse matrix stored in either a compressed row or compressed column storage data structure, and $-L^T$ is the strictly upper triangular sparse matrix that is not stored since it is already stored in the data structure for $-L$ when interpreted as the alternate storage scheme, i.e., compressed row if $-L$ is viewed as compressed column storage and vice versa. The sparse primitive should also be adapted to do matrix vector multiplication by $-L$ or $-L^T$ and to do a triangular solve $(D - L)v_1 = v_2$, $Dv_1 = v_2$ and $(D - L^T)v_1 = v_2$ as needed for the stationary methods. Note that as seen in the earlier programming assignments and homework solutions these system solve routines can be derived from the column-oriented or row-oriented matrix vector product in an organized and straightforward way since they share a similar control flow.

As with the other parts of the problem you should generate many sparse symmetric positive definite matrices with varying levels of sparsity, along with many solutions for each matrix, and many initial guesses for each problem. The matrices can be generated by first generating $-L$ by choosing the number of nonzeros below the diagonal and randomly generating values. You can place them anywhere in $-L$ but a reasonable approach is to generate a row or column at a time with a maximum number of elements prescribed and placing them in randomly selected column or row positions. The diagonal should be randomly selected positive numbers. In order to make the matrix positive definite the diagonal elements should be boosted so that the matrix A is strictly diagonally dominant, i.e., so that each diagonal element is larger than the sum of the magnitudes of the elements in its column and row (note the implication of symmetry that simplifies this constraint).

Consider carefully and express clearly the predictions you are testing with each set of data from the experiments relative to what is known from the class material in terms of expected performance. You may use any library to compute the eigenvalues of selected problems, e.g., those that are unexpectedly good or bad, in order to explain your observations. Note this may require you to move the associated matrix to a data structure required by that libary.