Program 5 Foundations of Computational Math 1 Fall 2011

Due date: This is an optional assignment via email before the Final Exam begins on Before Final Exam

General Task

Implement codes to use preconditioned steepest descent and preconditioned conjugate gradient to solve $Ax = b$ where $A \in \mathbb{R}^{n\times n}$ is symmetric positive definite.

You must demonstrate your code on multiple examples for three situations:

1. Take $A = \Lambda = diag(\lambda_1, \lambda_2, \dots, \lambda_n)$, i.e., a diagonal matrix and apply CG and steepest descent **without** preconditioning, i.e., set your preconditioner to I. You should choose various combinations of values for the λ_i to demonstrate how the spectrum affects the convergence of the two methods. Make sure that you demonstrate for a range of n the key points made in the text and notes about the convergence of these methods as a function of the matrices, in particular, the distribution of the eigenvalues.

2. Any symmetric positive definite matrix A can be written $A = Q\Lambda Q^T$ where Λ is a diagonal matrix with the eigenvalues of A (which are postive) forming the diagonal, and Q is an orthogonal matrix containing eigenvectors of A. Therefore, choosing a particular set of eigenvalues and the matrix Q allows you to define a nondiagonal symmetric positive definite matrix with known eigenvalues. Compare the performance seen in the first part of the problem when A was chosen to be a diagonal matrix with the performance of steepest descent and CG without preconditioning on $A = Q\Lambda Q^T$ with corresponding Λ.

3. Again using $A = Q\Lambda Q^T$, investigate the performance of steepest descent and CG with preconditioning. Use two preconditioners, the Jacobi preconditioner (diagonal preconditioning) and the preconditioning matrix defined by symmetric Gauss-Seidel given in the notes and text book.

The notes contain pseudocodes for Preconditioned CG and Preconditioned Steepest Descent. When implementing your code you may treat A as a dense matrix, i.e., do not worry about the storage of A or computational complexity of a single step. You can limit the size of the problems, therefore, to a $n \leq 200$.

Choose your b vectors based on a known solution $\hat{x}$ and use

$$\frac{\|x_k - \hat{x}\|_A}{\|\hat{x}\|_A} \quad \text{and} \quad \frac{\|x_k - \hat{x}\|_2}{\|\hat{x}\|_2}$$

to describe the progress of the methods towards the solution.

Submission of Results

Expected results comprise:

1. A document describing the manner in which you generated the problems, why you generated them, what you expect to happen for the problems used, what you observed, and what you conclude based on your observations.

2. The source code, makefiles, and instructions on how to compile and execute your code including the math department's machine used if applicable.

3. Code documentation should be included in each routine.

These results should be emailed to gallivan@math.fsu.edu by the beginning of the final exam on the due date above. You may be asked to demonstrate your code if your document does not completely convince me that you tested your code sufficiently.

Also keep in mind that the program portion of your grade will use the highest 3 grades awarded.