Homework 6 Foundations of Computational Math 2 Spring 2019

Problem 6.1

Let Λ be a diagonal matrix whose diagonal elements are positive and also are the eigenvalues of Λ. Given x_0, define the sequence x_k as the sequence of vectors produced by steepest descent applied to $\Lambda x = b$.

Show that, in general, steepest descent does not converge in one step to $x = \Lambda^{-1} b$ for all x_0. Your proof should also make it clear when steepest descent will converge in one step for all x_0.

Problem 6.2

Let $A \in \mathbb{R}^{n \times n}$ be a symmetric positive definite matrix, $C \in \mathbb{R}^{n \times n}$ be a symmetric nonsingular matrix, and $b \in \mathbb{R}^n$ be a vector. The matrix $M = C^2$ is therefore symmetric positive definite. Also, let $\tilde{A} = C^{-1} A C^{-1}$ and $\tilde{b} = C^{-1} b$.

The preconditioned Steepest Descent algorithm to solve $Ax = b$ is:

A, M are symmetric positive definite
x_0 arbitrary; $r_0 = b - Ax_0$; solve $Mz_0 = r_0$

do $k = 0, 1, \ldots$ until convergence

$w_k = Az_k$
$\alpha_k = \frac{z_k^T r_k}{z_k^T w_k}$
$x_{k+1} \leftarrow x_k + z_k \alpha_k$
$r_{k+1} \leftarrow r_k - w_k \alpha_k$
solve $Mz_{k+1} = r_{k+1}$

end

The Steepest Descent algorithm to solve $\tilde{A}\tilde{x} = \tilde{b}$ is:

$\tilde{A}$ is symmetric positive definite
$\tilde{x}_0$ arbitrary; $\tilde{r}_0 = \tilde{b} - \tilde{A}\tilde{x}_0$; $\tilde{v}_0 = \tilde{A}\tilde{r}_0$

do $k = 0, 1, \ldots$ until convergence

$\tilde{\alpha}_k = \frac{\tilde{r}_k^T \tilde{r}_k}{\tilde{r}_k^T \tilde{v}_k}$
$\tilde{x}_{k+1} \leftarrow \tilde{x}_k + \tilde{r}_k \tilde{\alpha}_k$
$\tilde{r}_{k+1} \leftarrow \tilde{r}_k - \tilde{v}_k \tilde{\alpha}_k$

$\tilde{v}_{k+1} \leftarrow \tilde{A}\tilde{r}_{k+1}$

end

Show that given the appropriate consistency between initial guesses the preconditioned steepest descent recurrences to solve $Ax = b$ can be derived from the steepest descent recurrences to solve $\tilde{A}\tilde{x} = \tilde{b}$.

Problem 6.3

Consider the generic Conjugate Direction algorithm for solving the minimization problem

$$\min_{x \in \mathbb{R}^n} f(x)$$

where $f(x) = 0.5\, x^T Ax - x^T b$, $b \in \mathbb{R}^n$, and $A \in \mathbb{R}^{n\times n}$ is symmetric positive definite.

Denote the A-orthogonal direction vectors $d_0, d_1, \ldots$ and let $r_k = b - Ax_k$. Show that

$$\frac{d_{k-1}^T r_0}{d_{k-1}^T A d_{k-1}} = \frac{d_{k-1}^T r_{k-1}}{d_{k-1}^T A d_{k-1}}$$

Problem 6.4

Recall the basic CD/CG properties that hold given the assumption that CG has not converged at step k,

- $x_k = \alpha_0 d_0 + \cdots + \alpha_{k-1} d_{k-1}$ is optimal (inherited from CD), i.e.,

$$\forall x \in x_0 + span[d_0, d_1, \ldots, d_{k-1}], \quad \|x_k - A^{-1}b\|_A \leq \|x - A^{-1}b\|_A$$

- $< d_k,\ d_j >_A = 0$ $i \neq j$ for $0 \leq i, j \leq k-1$ (inherited from CD).
- $< r_k,\ d_j >= 0$ for $0 \leq j \leq k-1$ (inherited from CD).
- $< r_k,\ r_j >= 0$ for $0 \leq j \leq k-1$ (CG-specific).
- $span[d_0, d_1, \ldots, d_k] = span[r_0, r_1, \ldots, r_k]$ (CG-specific).
- $span[r_0, r_1, \ldots, r_k] = span[r_0, Ar_0, \ldots, A^k r_0]$ (CG-specific).

Given the inherited properties prove the three CG-specific properties.

Problem 6.5

When solving $Ax = b$ or equivalently the associated quadratic definite minimization problem using CG, we have

$$x_{k+1} = x_0 + \alpha_0 p_0 + \cdots + \alpha_k p_k$$

where the p_j are $A-$orthogonal vectors. It can be shown that

$$span[p_0, \ldots, p_k] = span[r_0,\ Ar_0, \ldots, A^k r_0]$$

where $r_0 = b - Ax_0$ and x_0 is the initial guess at the solution $x^* = A^{-1}b$. Therefore,

$$x_{k+1} = x_0 + \gamma_0 r_0 + \gamma_1 A r_0 + \cdots + \gamma_k A^k r_0 = x_0 - P_k(A) r_0$$

where $P_k(A) = \gamma_0 I + \gamma_1 A + \cdots + \gamma_k A^k$ is a matrix that is called a matrix polynomial evaluated at A. (A space whose span can be defined by a matrix polynomial is called a Krylov space.)

Denote $d_j = A^j r_0$ for $j = 0,\ 1, \ldots$ and determine the relationship between the coefficients $\alpha_0, \ldots, \alpha_k$ and the coefficients $\gamma_0, \ldots, \gamma_k$.

Problem 6.6

Suppose $T \in \mathbb{R}^{n \times n}$ is a symmetric positive definite tridiagonal matrix. This implies that all of the diagonal elements are positive, i.e., $e_i^T T e_i > 0$. Supose that T is also an irreducible tridiagonal matrix. This implies that the first subdiagonal (and by symmetry the first superdiagonal) are all nonzero, i.e.,$e_{i+1}^T T e_i = e_i^T T e_{i+1} \neq 0$. Jacobi and CG both converge when solving $Tx = b$ for any x_0. Assume that all arithmetic operations take 1 unit of time. You may ignore the cost of the termination check.

How much faster must CG converge compared to Jacobi in order to require fewer operations to achieve the same accuracy in the solution?

Problem 6.7

Consider the block tridiagonal matrix associated with an $n \times n$ grid discretization of the partial differential $u_{\xi,\xi} + u_{\eta,\eta} = g$ on a two-dimensional domain.

The matrix is $n^2 \times n^2 = N \times N$ where $N = n^2$, with $n \times n$ blocks $T_i \in \mathbb{R}^{n \times n}$ $1 \leq i \leq n$ $E_i = -I_n \in \mathbb{R}^{n \times n}$ $1 \leq i \leq n$ and block tridiagonal structure given by

$$A = \begin{pmatrix} T_1 & E_1 & 0 & \cdots & \cdots & \cdots & 0 \\ E_2 & T_2 & E_2 & 0 & & & \vdots \\ 0 & E_3 & T_3 & E_3 & 0 & & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & E_{n-1} & T_{n-2} & E_{n-2} & 0 \\ 0 & & \cdots & 0 & E_{n-1} & T_{n-1} & E_{n-1} \\ 0 & & & \cdots & 0 & E_n & T_n \end{pmatrix}$$

where T_i are tridiagonal and E_i are diagonal and dimensions $n \times n$

$$T_i = \begin{pmatrix} 4 & -1 & 0 & 0 & 0 & \dots & 0 \\ -1 & 4 & -1 & 0 & 0 & \dots & 0 \\ 0 & -1 & 4 & -1 & 0 & \dots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & & \vdots \\ 0 & \dots & 0 & -1 & 4 & -1 & 0 \\ 0 & \dots & 0 & 0 & -1 & 4 & -1 \\ 0 & \dots & 0 & 0 & 0 & -1 & 4 \end{pmatrix}$$

$$E_i = -I_n = \begin{pmatrix} -1 & 0 & 0 & 0 & 0 & \dots & 0 \\ 0 & -1 & 0 & 0 & 0 & \dots & 0 \\ 0 & 0 & -1 & 0 & 0 & \dots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & & \vdots \\ 0 & \dots & 0 & 0 & -1 & 0 & 0 \\ 0 & \dots & 0 & 0 & 0 & -1 & 0 \\ 0 & \dots & 0 & 0 & 0 & 0 & -1 \end{pmatrix}$$

Note: Be careful with N vs. n in your answer to this problem.

6.7.a

i. Determine the computational complexity the matrix-vector product $Av \to z$ where $v,\ z \in \mathbb{R}^N$.

ii. Determine the computational complexity of one step of CG without preconditioning to solve $Ax = b$.

For both answers express the complexity as $CN^k + O(N^{k-1})$ where k is an appropriate integer and C is a constant. (You must give both k and C)

6.7.b

Suppose you use an incomplete Cholesky preconditioner where $M = LL^T$. L is a lower triangular matrix with a nonzero structure identical to the diagonal and strict lower triangular part of A, i.e.,

$$L = \begin{pmatrix} L_1 & 0 & 0 & \cdots & \cdots & \cdots & 0 \\ D_2 & L_2 & 0 & 0 & & & \vdots \\ 0 & D_3 & L_3 & 0 & 0 & & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & D_{n-1} & L_{n-2} & 0 & 0 \\ 0 & & \cdots & 0 & D_{n-1} & L_{n-1} & 0 \\ 0 & & & \cdots & 0 & D_n & L_n \end{pmatrix}$$

where L_i are lower triangular and D_i are diagonal and dimensions $n \times n$ where the nonzero positions are marked with $*$ (the actual values are not important).

$$L_i = \begin{pmatrix} * & 0 & 0 & 0 & 0 & \dots & 0 \\ * & * & 0 & 0 & 0 & \dots & 0 \\ 0 & * & * & 0 & 0 & \dots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & & \vdots \\ 0 & \dots & 0 & * & * & 0 & 0 \\ 0 & \dots & 0 & 0 & * & * & 0 \\ 0 & \dots & 0 & 0 & 0 & * & * \end{pmatrix}, \quad D_i = \begin{pmatrix} * & 0 & 0 & 0 & 0 & \dots & 0 \\ 0 & * & 0 & 0 & 0 & \dots & 0 \\ 0 & 0 & * & 0 & 0 & \dots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & & \vdots \\ 0 & \dots & 0 & 0 & * & 0 & 0 \\ 0 & \dots & 0 & 0 & 0 & * & 0 \\ 0 & \dots & 0 & 0 & 0 & 0 & * \end{pmatrix}$$

i. Determine the computational complexity of one step of CG **with** preconditioning to solve $Ax = b$ using $M = LL^T$. Express the complexity as $CN^k + O(N^{k-1})$ where k is an appropriate integer and C is a constant. (You must give both k and C)

ii. Compare the complexity of one step of CG with preconditioning and one step without preconditioning. How effective does the preconditioner have to be in reducing the number of iterations in order for CG with preconditioning to take fewer computations to reach the same accuracy?