

1. We defined the Chebyshev polynomial in class as  $\tau_k(z) = \cos(k \arccos z)$  when  $z$  is a real number with  $|z| \leq 1$ .

a. Show that the roots of  $\tau_k(z)$  are  $\left\{ \cos\left(\frac{(2j-1)\pi}{2k}\right) \mid j = 1, 2, \dots, k \right\}$ .

b. For  $k \geq 1$ , let  $\tilde{\tau}_k(z) = \left(\frac{1}{2^{k-1}}\right) \tau_k(z)$ , so that  $\tilde{\tau}_k(z)$  is a polynomial of degree  $k$  with leading coefficient equal to 1. Prove that

$$\max_{z \in [-1, 1]} |\tau_k(z)| \leq \max_{z \in [-1, 1]} |T_k(z)|$$

where  $T_k$  is any other polynomial of degree  $k$  with leading coefficient 1.

c. For  $|z| > 1$ , let  $\tau_k(z) = \cosh(k \operatorname{arccosh} z)$ . Show that in this case,  $\tau_k(z)$  satisfies the same recurrence derived in class,

$$\tau_{k+1}(z) = 2z\tau_k(z) - \tau_{k-1}(z).$$

d. Prove that

$$\tau_k(t) = \frac{1}{2} \left[ (t + \sqrt{t^2 - 1})^k + (t - \sqrt{t^2 - 1})^k \right].$$

2. Let  $Ax = b$  be a linear system of equations where  $A$  is symmetric and positive-definite. Starting from an arbitrary initial iterate  $x_0$ , the steepest descent method generates a sequence of iterates  $x_1, x_2, \dots$  by the computation

$$x_{k+1} = x_k + \alpha_k r_k,$$

where  $r_k$  is the residual  $b - Ax_k$  and  $\alpha_k$  is a scalar chosen so that the norm  $\|x - x_{k+1}\|_A$  is minimal.

a. Explain the name “steepest descent method.”

b. Show that the error  $e_k = x - x_k$  satisfies

$$\|e_k\|_A \leq \left(\frac{\kappa - 1}{\kappa + 1}\right)^k \|e_0\|_A.$$

where  $\kappa = \Lambda/\lambda$  is the *condition number* of  $A$ , that is, the ratio of the largest eigenvalue of  $A$  to its smallest eigenvalue.

3. Given  $Ax = b$  as in Problem 2, let  $x_k$  be the  $k$ th iterate generated by the conjugate gradient method. Show that if  $x_k \neq x$ , then the vectors generated by CG satisfy

- (i)  $\langle r_k, p_j \rangle = \langle r_k, r_j \rangle = 0, \quad j < k,$
- (ii)  $\langle Ap_k, p_j \rangle = 0, \quad j < k,$
- (iii)  $\text{span}\{r_0, r_1, \dots, r_{k-1}\} = \text{span}\{p_0, p_1, \dots, p_{k-1}\}$   
 $= \mathcal{K}(A, r_0) \equiv \text{span}\{r_0, Ar_0, \dots, A^{k-1}r_0\}.$

Hint: Prove (i) and (ii) simultaneously by induction on  $k$ , and use a dimensionality argument for (iii).

4a. Write your own version of the conjugate gradient algorithm and use it to solve the problem  $Ax = b$  where  $A$  and  $b$  are given in the Matlab mat-files `hw1*.mat`. The program should take as input the matrix  $A$ , right hand side  $b$ , a maximum number of iterations, and a tolerance  $\tau$ . For output, it should produce the computed solution, a vector containing all the residual norms, and a flag indicating whether the stopping tolerance

$$\|r_k\|_2 \leq \tau \|b\|_2$$

is satisfied. Use this algorithm to compute an approximate solution  $x_k$  for which the Euclidean norm of the residual satisfies

$$\|r_k\|_2 \leq 10^{-8} \|b\|_2,$$

where  $x_0 = 0$ . Plot the residual norm (on a logarithmic scale) against the iteration counts, and identify the number of iterations required.

b. Modify the program so that at each step, the matrix

$$S_k = \text{tridiag} \left[ -\frac{1}{\alpha_{j-1}}, \frac{1}{\alpha_j} + \frac{\beta_{j-1}}{\alpha_{j-1}}, -\frac{\beta_j}{\alpha_j} \right]$$

is also constructed. (Note that  $S_k$  contains  $S_{k-1}$  as a principal minor.) As the iteration proceeds, compute the eigenvalues of  $S_k$  and compare them to those of  $A$ . You can use the Matlab function `eig` to compute the eigenvalues in each case. What do you observe?