

# CS 6210: HOMEWORK 4

Instructor: Anil Damle

Due: March 31, 2022

## POLICIES

You may discuss the homework problems freely with other students, but please refrain from looking at their code or writeups (or sharing your own). Ultimately, you must implement your own code and write up your own solution to be turned in. Your solution, including plots and requested output from your code should be typeset and submitted via the Gradescope as a pdf file. Additionally, please submit any code written for the assignment. This can be done by either including it in your solution as an appendix, or uploading it as a zip file to the separate Gradescope assignment.

---

## QUESTION 1:

Let us assume that we are using a stationary iterative method to solve  $Ax = b$  with the splitting  $A = M - N$  and initial guess  $x^{(0)}$ . Furthermore, assume that  $x^{(1)}, \dots, x^{(k)}$  have been computed using the iteration  $Mx^{(j+1)} = Nx^{(j)} + b$ . Normally we would consider  $x^{(k)}$  as our current approximation of the solution. However, maybe there is some process that allows us to accelerate the convergence of our method and draw an interesting connection to Krylov subspace methods.

Now, we are going to consider one way of accomplishing our goal. Specifically, we would like to construct coefficients  $\{v_j^{(k)}\}_{j=1}^k$  for each iteration  $k$  such that

$$y^{(k)} = \sum_{j=0}^k v_j^{(k)} x^{(j)}$$

gives a better approximation to our true solution, denoted  $x$ . Let us define  $G = M^{-1}N$  and

$$p_k(z) = \sum_{j=0}^k v_j^{(k)} z^j$$

- (a) Assuming  $x^{(0)} = 0$ , is  $y^{(k)}$  consistently part of a Krylov subspace? If so, which Krylov subspace? (By consistently, I mean can you define a matrix  $H$  and vector  $w$  that do not depend on  $k$  such that  $y^{(k)} \in \mathcal{K}_k(H, w)$ .)
- (b) Now, let us further assume that  $p_k(1) = 1$ . Prove that

$$y^{(k)} - x = p_k(G) e^{(0)}$$

- (c) Prove that if  $B$  is similar to a Hermitian matrix then

$$\rho(p_k(B)) = \max_{\lambda_i \in \lambda(B)} |p_k(\lambda_i)|$$

where  $\rho(p_k(B))$  is the spectral radius of  $p_k(B)$ .

We now assume that the iteration matrix  $G$  is similar to a Hermitian matrix and has real eigenvalues  $\lambda_1, \dots, \lambda_n$ . Recall that for convergence we must have

$$-1 < \lambda_n \leq \lambda_1 < 1$$

Moreover let  $\alpha$  and  $\beta$  be such that

$$-1 < \alpha \leq \lambda_n \leq \dots \leq \lambda_1 \leq \beta < 1.$$

- (d) Since we may write  $\|y^{(k)} - x\| = \|p_k(G)e^{(0)}\|$ , what choice of  $p_k$  would give us zero error? Is such a choice feasible?

Since we may write

$$\max_{\lambda_i \in \lambda(B)} |p_k(\lambda_i)| \leq \max_{\alpha \leq \lambda \leq \beta} |p_k(\lambda)|,$$

it seems reasonable to pick a  $p_k$  that is small on the interval  $[\alpha, \beta]$ . The ideal choice, given the constraint on  $p_k(1)$ , is a scaled and shifted version of the  $k^{\text{th}}$  Chebyshev polynomial. These polynomials may be defined by the recursion  $c_j(z) = 2zc_{j-1}(z) - c_{j-2}(z)$  where  $c_0(z) = 1$  and  $c_1(z) = z$ . Alternatively, we may write  $c_j(z) = \cos(j\theta)$  where  $\theta = \arccos(z)$ . Specifically, we may choose our polynomial to be

$$p_k(z) = \frac{c_k\left(-1 + 2\frac{z-\alpha}{\beta-\alpha}\right)}{c_k(\mu)},$$

where  $\mu = 1 + 2\frac{1-\beta}{\beta-\alpha}$ . You can verify that  $p_k(1) = 1$ . Moreover,  $c_k(z)$  has the property that it is bounded between  $-1$  and  $1$  in the interval  $[-1, 1]$ , but then grows rapidly outside of this interval. So for example, in the formula above,  $c_k(\mu)$  becomes large as  $k \rightarrow \infty$ . With the chosen scaling we ensure that  $p(z)$  is small in the interval  $[\alpha, \beta]$  while satisfying  $p(1) = 1$ .

- (e) Given the above choice for  $p_k(z)$  prove that there exists a constant  $C$  such that

$$\|y^{(k)} - x\|_2 \leq C \left( \frac{1}{c_k(\mu)} \right) \|x - x^{(0)}\|_2,$$

where  $C$  may depend on the matrix  $G$ .

- (f) Let  $\alpha = -0.9$  and  $\beta = 0.9$ . Plot  $c_k(\mu)$ , on a logarithmic scale, for  $k = 0, 1, \dots, 100$ .

We will now consider using this acceleration method in conjunction with the Jacobi iteration. For the remainder of this problem assume that  $A$  is a real symmetric matrix that is strictly diagonally dominant and has positive diagonal entries.

- (g) Under the aforementioned assumptions, prove that the iteration matrix associated with  $A$  is similar to a Hermitian matrix.
- (h) Implement the Jacobi method both with and without Chebyshev acceleration. You can find pseudo-code in Golub and Van Loan 4th edition, section 11.2.8 (3rd edition, section 10.1.5) that leverages the three-term recurrence for Chebyshev polynomials for an efficient implementation.

Build some non-singular test problems via matrices  $A$ , vectors  $x^{(0)}$  and  $b$  where the eigenvalues of  $G$  approach  $\pm 1$ . You may use a built in routine to compute eigenvalues and set the

eigenvalue bounds as  $\alpha = (-1 + \lambda_n)/2$  and  $\beta = (1 + \lambda_1)/2$ . Use your algorithm both with and without the acceleration to solve  $Ax = b$ . You may stop your algorithm when the 2 norm of the residual is less than  $10^{-6}$  or you have run 1000 iterations. Provide error plots, on a logarithmic scale, of the 2 norm of the residual vs iteration both with and without the acceleration. Comment on your observations.

- (i) How does the scheme here relate to our discussion of Krylov methods?

**QUESTION 2 (UNGRADED BUT INTERESTING; WE USED THIS RESULT IN CLASS):**

Here we will characterize the behavior of  $A^k$  in terms of its spectral radius.

- (a) Prove that if  $\rho(A) < 1$  then there exists a sub-multiplicative matrix norm  $\|\cdot\|$  such that  $\|A\| < 1$ . Hint: start with the Schur decomposition and note that for any non-singular matrix  $S$  and norm  $\|\cdot\|$ ,  $\|A\|_S = \|S^{-1}AS\|$  is also a norm.
- (b) Given  $A \in \mathbb{R}^{n \times n}$ , a matrix norm  $\|\cdot\|$ , and some  $\epsilon > 0$  prove that there exist constants  $\alpha$  (depending on the norm) and  $\beta_{A,\epsilon}$  (depending on  $A$ , the norm, and  $\epsilon$ ) such that

$$\alpha\rho(A)^k \leq \|A^k\| \leq \beta_{A,\epsilon}(\rho(A) + \epsilon)^k.$$

**QUESTION 3:**

We are now going to consider some details about how Krylov methods act in specific situations.

- (a) Given a symmetric positive definite matrix  $A$  and vector  $b$ , prove that if the Lanczos process breaks down at some point (*i.e.*  $\beta_k = 0$  using the notation from class and Trefethen and Bau) then the subspace  $\mathcal{K}_k(A, b)$  contains a solution to the linear system  $Ax = b$ . In principle we might be worried that if  $\beta_k = 0$  things have gone horribly wrong since we cannot construct the next vector in our orthonormal basis. However, this result shows that in this context everything has actually gone remarkably well.
- (b) Given a non-singular diagonalizable matrix  $A$  with at most  $p$  distinct eigenvalues and a vector  $b$ , show that a solution to  $Ax = b$  exists in  $\mathcal{K}_k(A, b)$  for some  $k \leq p$ . In other words, we certainly have a solution in the  $p^{\text{th}}$  Krylov subspace, though we may find one sooner in some special circumstances.

**QUESTION 4:**

In class we discussed how the Lanczos based formulation of CG can be made efficient (both in terms of compute and storage) by maintaining a  $LDL^T$  factorization of  $T_k$ .<sup>1</sup> More specifically, consider the factorization

$$T_k = L_k D_k L_k^T,$$

where  $L_k$  is unit lower bidiagonal and  $D_k$  is diagonal. Because of how  $T_{k+1}$  is related to  $T_k$ , given  $L_k$  and  $D_k$  we can compute  $L_{k+1}$  and  $D_{k+1}$  in a constant number of operations given the new entries in  $T_{k+1}$ .

However, even given that fact it still takes a bit of work to make the CG iteration efficient. The key step is to rewrite the CG update in terms of the vector  $z^{(k)} = L_k^T y^{(k)}$  instead of  $y^{(k)}$  directly and utilize an auxiliary sequence of vectors to  $V_k$ .<sup>2</sup> We will show why this helps in two steps:

<sup>1</sup>A similar argument can be made for MINRES using a QR factorization.

<sup>2</sup>Remember that in class we said the CG iterate can be defined as  $x^{(k)} = V_k y^{(k)}$  where  $y^{(k)}$  solves  $T_k y^{(k)} = \|b\|_2 e_1$  and  $V_k$  is the basis for  $\mathcal{K}_k(A, b)$  computed by the Lanczos procedure.

(a) If  $z^{(k)}$  solves the system  $L_k D_k z^{(k)} = \|b\|_2 e_1$ , show that

$$z^{(k)} = \begin{bmatrix} z^{(k-1)} \\ \xi_k \end{bmatrix}$$

and that  $\xi_k$  can be computed with a constant (in  $k$ ) number of arithmetic operations. In other words show that  $z^{(k)}$  is constructed by appending a new entry, denoted  $\xi_k$ , to  $z^{(k-1)}$  and that we can compute  $\xi_k$  efficiently.

(b) If we define the matrix  $W_k$  as  $W_k L_k^T = V_k$  show that

$$W_k = [W_{k-1} \quad w_k]$$

and that  $w_k$  can be computed using only  $v_k$  (the last column of  $V_k$ ) and  $w_{k-1}$ . In other words, show that  $W_k$  is constructed by simply appending a new vector to  $W_{k-1}$ , and that the new vector can be computed without needing to store all of  $V_k$  or  $W_k$ .

(c) Show that we can rewrite  $x^{(k)}$  as

$$x^{(k)} = x^{(k-1)} + \xi_k w_k.$$

Why do the preceding three points imply that the cost per iteration of CG is independent of  $k$  and, more specifically, that it is  $T_{\text{mult}}(A) + \mathcal{O}(n)$ ? (Recall that  $T_{\text{mult}}(A)$  is the cost of doing a matrix vector product with  $A$ .)