

# CS 6210: HOMEWORK 2

Instructor: Anil Damle

Due: September 20, 2018

## 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 supporting plots and requested output from your code must be typeset and submitted via the CMS as a single pdf file. Additionally, please submit any code written for the assignment via the CMS as well. This can be done by either including it in your solution as an appendix, or uploading it as a zip file.

---

## QUESTION 1:

For the following four problems state whether you think it is plausible/reasonable to try and construct stable or backwards stable algorithms. Think carefully about what the input/output space of such algorithms is and based on your characterization of the problem justify your choice.

1. Computing the inner product  $x^*y$  given vectors  $x, y \in \mathbb{C}^n$ .
2. Computing the outer product  $xy^*$  given vectors  $x, y \in \mathbb{C}^n$ .
3. Computing Cholesky factorization ( $A = LL^*$ , with  $L$  lower triangular) of a given Hermitian positive definite matrix  $A \in \mathbb{C}^{n \times n}$ .
4. Computing the SVD of a given matrix  $A \in \mathbb{C}^{n \times n}$ .

## QUESTION 2:

For any square non-singular matrix  $A$ , prove that  $1/\kappa_2(A)$  is the relative distance to the nearest singular matrix to  $A$  in the two-norm. In other words, prove that

$$\frac{1}{\kappa(A)} = \min_{\text{rank}(X) < n} \frac{\|A - X\|_2}{\|A\|_2}.$$

## QUESTION 3:

Assume we solve a problem  $f$  with a stable algorithm  $\tilde{f}$ , prove that the relative accuracy is bounded by  $\mathcal{O}(\kappa(x)\mu)$  where  $\mu$  is machine precision and  $\kappa(x)$  is the relative condition number of  $f$  at  $x$ .

## QUESTION 4:

Implement LU factorizations with no, partial, and complete pivoting for  $A \in \mathbb{R}^{n \times n}$ . Then, address the following:

- Demonstrate that your implementations can accurately solve most well-conditioned square non-singular linear systems  $Ax = b$  given  $A$  and  $b$ , and scales as  $\mathcal{O}(n^3)$ . Clearly outline how you test this and include your results.

- Construct test cases where without the use of pivoting your algorithm will result in large errors in the solution—demonstrate this does indeed happen.
- Construct a test case where partial pivoting fails to yield a good solution but complete pivoting does—demonstrate this does indeed happen. Such examples exist, but are typically considered pathological and “rare” enough that simply using LU with partial pivoting (despite its poor worst case theoretical performance) is fine in practice.
- Consider computing the LU factorization (all three ways) of the so-called Hilbert matrix of size  $n$  (defined as  $H_{ij} = 1/(i+j-1)$  and a symmetric positive definite matrix). For moderate  $n$  what do you observe about  $\|LU - H\|_2$  (adding permutation matrices as applicable)? What about the accuracy of a solution to  $Hx = b$ ? Explain your observations.