Sep 26, 2023 (Due: 08:00 Oct 10, 2023)

1. Describe how to avoid cancellation when constructing Householder reflections in the Householder triangularization algorithm for complex matrices.

2. Estimate the cost for Householder triangularization process, as well as the cost of explicitly forming the QR factorization (if requested by user).

3. Write a program to compute the QR factorization of a general complex matrix $A \in \mathbb{C}^{m\times n}$ with $m \geq n$ with
(1) Cholesky QR (i.e., through the Cholesky factorization of A^*A);
(2) Householder triangularization.
Visualize the loss of orthogonality $|Q^*Q - I_n|$.
(If you use MATLAB/Octave, you may find `imagesc()` helpful.

4. Let

$$A = \begin{bmatrix} \alpha_1 & \rho_2 & \rho_3 & \cdots & \cdots & \rho_n \\ \beta_2 & \alpha_2 & 0 & \cdots & \cdots & 0 \\ \beta_3 & 0 & \alpha_3 & \ddots & & \vdots \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \vdots & & \ddots & \alpha_{n-1} & 0 \\ \beta_n & 0 & \cdots & \cdots & 0 & \alpha_n \end{bmatrix} \in \mathbb{R}^{n\times n}.$$

Design an efficient algorithm to compute the QR factorization of A.

5. Let $Q \in \mathbb{R}^{n\times n}$ be an orthogonal matrix. Show that Q can be factorized as the product of finitely many Householder reflections, and if, in addition, $\det(Q) = 1$, Q can be factorized as the product of finitely many Givens rotations.

6. (optional) Let $w_1, w_2, \ldots, w_k \in \mathbb{C}^n$ be unit vectors. Try to find a matrix $T \in \mathbb{C}^{k\times k}$ such that

$$(I - 2w_1w_1^*)(I - 2w_2w_2^*)\cdots(I - 2w_kw_k^*) = I - [w_1, w_2, \ldots, w_k]T[w_1, w_2, \ldots, w_k]^*.$$

7. (optional) Design an efficient algorithm to compute the QR factorization of $R + uv^\top$, where $R \in \mathbb{R}^{n\times n}$ is an upper triangular matrix and $u, v \in \mathbb{R}^n$ are column vectors.

8. (H) Use pseudocode to describe a block algorithm (either left-looking or right-looking) for Householder triangularization. Make sure all indices are correct. For simplicity, you may assume that the number of columns of the matrix is a multiple of the block size.