

CSE 6643 Homework 6

Schäfer, Spring 2023

Deadline: April 11 Tuesday, 8:00 am

- There are 2 sections in grade scope: Homework 6 and Homework 6 Programming. Submit your answers as a PDF file to Homework 6 (report the results that you obtain using programming by using plots, tables, and a description of your implementation like you would when writing a paper.) and also submit your code in a zip file to Homework 6 Programming.
- Programming questions are posted in Julia. You are allowed to use basic library functions like sorting, plotting, matrix-vector products etc, but nothing that renders the problem itself trivial. Please use your common sense and ask the instructors if you are unsure. You should never add additional packages to the environment.
- Late homework incurs a penalty of 20% for every 24 hours that it is late. Thus, right after the deadline, it will only be worth 80% credit, and after four days, it will not be worth any credit.
- We recommend the use of LaTeX for typing up your solutions. No credit will be given to unreadable handwriting.
- List explicitly with whom in the class you discussed which problem, if any. Cite all external resources that you were using to complete the homework. For details, consult the collaboration policy in the class syllabus on canvas.

1 Convergence of QR iteration [50 pts]

In this problem, we consider the convergence rate of the QR algorithm with a single-shift strategy. We consider a real matrix $\mathbf{A} \in \mathbb{R}^{m \times m}$. The QR iteration can be written as follows:

$$\mathbf{A}^{(0)} = \mathbf{A} \tag{1}$$

$$\mathbf{A}^{(k)} = \mu_k \mathbf{I} + \mathbf{Q}_k \mathbf{R}_k, \tag{2}$$

$$\mathbf{A}^{(k+1)} = \mathbf{R}_k \mathbf{Q}_k + \mu_k \mathbf{I}. \tag{3}$$

If we choose $\mu_k = \mathbf{A}_{m,m}^{(k)}$ to be the bottom-right entry of the matrix $\mathbf{A}^{(k)}$, then this is called the *single-shift QR iteration*.

Prove the following results. You may use figures to illustrate your explanations.

(a) [10 pts]

Show that if $\mathbf{A}^{(0)} = \mathbf{A}$ is an upper Hessenberg matrix, then $\mathbf{A}^{(k)}$ is upper Hessenberg for all $k \geq 0$. Thus, from now on, we always assume that the matrix \mathbf{A} is an upper Hessenberg matrix.

(b) [10 pts]

Prove that the total operation cost for each QR iteration is $O(m^2)$.

(c) [10 pts]

In the QR step, we perform $m - 1$ Givens rotations on the matrix $\mathbf{A}^{(k)} - \mu_k \mathbf{I}$. Suppose that after $m - 2$ Givens rotations, the bottom-right 2×2 sub-matrix of $\mathbf{A}^{(k)} - \mu_k \mathbf{I}$ is given by

$$\begin{pmatrix} a & b \\ \varepsilon & 0 \end{pmatrix}. \quad (4)$$

Explain why the (m, m) entry is 0 at that stage, and prove that

$$\mathbf{A}_{m,m-1}^{(k+1)} = -\frac{\varepsilon^2 b}{\varepsilon^2 + a^2}. \quad (5)$$

(d) [10 pts]

Based on the previous result, explain why we can expect the single-shift QR algorithm to converge quadratically (provided that it is converging).

(e) [10 pts]

We showed that the single-shift QR algorithm converges quite fast if the guess is sufficiently accurate. However, its convergence is not guaranteed. Give an example in which the single-shift QR algorithm fails to converge, and explain why.

2 Deflation upon Convergence [20 pts]

Consider an upper Hessenberg matrix $\mathbf{H} \in \mathbb{R}^{m \times m}$ with eigenvalue λ . We define

$$\mathbf{H} - \lambda \mathbf{I} = \mathbf{U}_1 \mathbf{R}_1 \quad (\text{QR factorization}) \quad (6)$$

$$\mathbf{H}_1 = \mathbf{R}_1 \mathbf{U}_1 + \lambda \mathbf{I}. \quad (7)$$

(a) [10 pts]

Prove that if $\mathbf{H}_{i+1,i} \neq 0, \forall 1 \leq i < m$ (\mathbf{H} is an unreduced Hessenberg matrix), then

$$\mathbf{H}_1(m, :) = \lambda \mathbf{e}_m^T. \quad (8)$$

(b) [10 pts]

Explain the connection between this result and the process of deflation in the QR iteration algorithm.

3 An implicit QR Factorization [15 bonus pts]

Denote $\mathbf{H} = \mathbf{H}_1$, and assume we generate a sequence of matrices \mathbf{H}_k via

$$\mathbf{H}_k - \mu_k \mathbf{I} = \mathbf{U}_k \mathbf{R}_k, \quad \mathbf{H}_{k+1} = \mathbf{R}_k \mathbf{U}_k + \mu_k \mathbf{I}. \quad (9)$$

Prove that

$$(\mathbf{U}_1 \cdots \mathbf{U}_j)(\mathbf{R}_j \cdots \mathbf{R}_1) = (\mathbf{H} - \mu_j \mathbf{I}) \cdots (\mathbf{H} - \mu_1 \mathbf{I}). \quad (10)$$

This result shows that we are implicitly computing a QR factorization of

$$(\mathbf{H} - \mu_j \mathbf{I}) \cdots (\mathbf{H} - \mu_1 \mathbf{I}). \quad (11)$$

4 QR with Shifts [30 pts]

(a) Almost upper triangular [7.5 pts]

Go to section (a) of the file `HW6_your_code.jl` and implement a function that reduces a symmetric matrix $\mathbf{A} \in \mathbb{R}^{m \times m}$ to Hessenberg form using Householder reflections. You should end up with a matrix \mathbf{T} in Hessenberg form. Your algorithm should operate in place, overwriting the input matrix and not allocating additional memory.

(b) Givens [7.5 pts]

Go to section (b) of the file `HW6_your_code.jl` and implement a function that runs a single iteration of the unshifted QR algorithm. Your function should take \mathbf{T}_k in Hessenberg form as an input and compute \mathbf{T}_{k+1} also in Hessenberg form. You should use Givens rotations to implement QR-factorization on \mathbf{T}_k .

(c) Single-Shift vs. Wilkson Shifts [7.5 pts]

Go to section (c) of the file `HW6_your_code.jl` and implement a function that runs the practical QR iteration with both the Single-Shift and Wilkinson Shift. Your function should have an input that allows you to select which type of shift you want to use. Your implementation should include deflation and a reasonable criteria for when to implement deflation and terminate your QR iterations. You can use your function from part (b) to do the QR iteration at each step.

(d) Breaking symmetry [7.5 pts]

Go to section (d) of the file `HW6_your_driver.jl` and design an experiment that evaluates your practical QR algorithm with shifts. You should include a semi-log plot showing the rate of convergence of your algorithm using Single-Shift and Wilkinson Shift. Compare the results with the rate of convergence you expected to see for both cases. Do you have a preference between the Wilkinson shift or the Rayleigh shift? If so which one do you prefer and why?