

# CSE 6643 Homework 5

Karl Hiner, Spring 2023

## 1 Power Method [30 pts]

We consider a matrix  $\mathbf{A}$  such that

$$\mathbf{Q}^T \mathbf{A} \mathbf{Q} = \text{diag}(\lambda_1, \dots, \lambda_m), \quad (1)$$

where  $\mathbf{Q}$  is orthogonal. We denote by  $\mathbf{q}_i$  the  $i$ th column of  $\mathbf{Q}$ .

We now consider the power method and the sequence  $\boldsymbol{\nu}^{(k)}$  defined as

$$\mathbf{z}^{(k)} = \mathbf{A} \boldsymbol{\nu}^{(k-1)} \quad (2)$$

$$\boldsymbol{\nu}^{(k)} = \mathbf{z}^{(k)} / \|\mathbf{z}^{(k)}\|_2, \quad (3)$$

where we assume that  $\|\boldsymbol{\nu}^{(0)}\|_2 = 1$ . Assume that we have  $\theta_k$  such that

$$\cos(\theta_k) = \mathbf{q}_1^T \boldsymbol{\nu}^{(k)} \quad (4)$$

with  $\cos(\theta_0) \neq 0$ . Prove that

$$1 - \cos(\theta_k)^2 \leq \frac{1}{a_1^2} \sum_{i=2}^m a_i^2 \left( \frac{\lambda_i}{\lambda_1} \right)^{2k}, \quad a_i = \mathbf{q}_i^T \boldsymbol{\nu}^{(0)} \quad (5)$$

$$\begin{aligned}
\cos(\theta_k) &= \mathbf{q}_1^T \boldsymbol{\nu}^{(k)} && \text{(given)} \\
&= \mathbf{q}_1^T \frac{\mathbf{A}^k \boldsymbol{\nu}^{(0)}}{\|\mathbf{A}^k \boldsymbol{\nu}^{(0)}\|_2} && \text{(def. of power method)} \\
&= \frac{\mathbf{q}_1^T \mathbf{Q} \boldsymbol{\Lambda}^k \mathbf{Q}^T \boldsymbol{\nu}^{(0)}}{\|\mathbf{Q} \boldsymbol{\Lambda}^k \mathbf{Q}^T \boldsymbol{\nu}^{(0)}\|_2} && \text{(def. of } \mathbf{A} \text{)} \\
&= \frac{\mathbf{q}_1^T \mathbf{Q} \boldsymbol{\Lambda}^k \mathbf{a}}{\|\mathbf{Q} \boldsymbol{\Lambda}^k \mathbf{a}\|_2} && (\mathbf{a} \equiv \mathbf{Q}^T \boldsymbol{\nu}^{(0)} \rightarrow a_i = \mathbf{q}_i^T \boldsymbol{\nu}^{(0)}) \\
\cos(\theta_k)^2 &= \frac{(\mathbf{q}_1^T \mathbf{Q} \boldsymbol{\Lambda}^k \mathbf{a})^2}{\|\mathbf{Q} \boldsymbol{\Lambda}^k \mathbf{a}\|_2^2} && \text{(square both sides)} \\
&= \frac{(\mathbf{q}_1^T \mathbf{Q} \boldsymbol{\Lambda}^k \mathbf{a})^2}{(\mathbf{Q} \boldsymbol{\Lambda}^k \mathbf{a})^T (\mathbf{Q} \boldsymbol{\Lambda}^k \mathbf{a})} && \text{(def. of 2-norm)} \\
&= \frac{(\mathbf{q}_1^T \mathbf{Q} \boldsymbol{\Lambda}^k \mathbf{a})^2}{\mathbf{a}^T \boldsymbol{\Lambda}^k \mathbf{Q}^T \mathbf{Q} \boldsymbol{\Lambda}^k \mathbf{a}} && \text{(apply transpose in den.)} \\
&= \frac{(\mathbf{q}_1^T \mathbf{Q} \boldsymbol{\Lambda}^k \mathbf{a})^2}{\boldsymbol{\Lambda}^{2k} \mathbf{a}^T \mathbf{a}} && \text{(simplify den. } \mathbf{Q} \text{ orth., } \boldsymbol{\Lambda} \text{ diag.)} \\
&= \frac{(\mathbf{q}_1^T \mathbf{Q} \boldsymbol{\Lambda}^k \mathbf{a})^2}{\sum_{i=1}^m \lambda_i^{2k} a_i^2} && \text{(expand den.)} \\
&= \frac{(\mathbf{e}_1 \boldsymbol{\Lambda}^k \mathbf{a})^2}{\sum_{i=1}^m \lambda_i^{2k} a_i^2} && (\mathbf{q}_1^T \cdot \mathbf{q}_1 = 1, \mathbf{q}_1^T \cdot \mathbf{q}_{i \neq 1} = 0) \\
&= \frac{\lambda_1^{2k} a_1^2}{\sum_{i=1}^m \lambda_i^{2k} a_i^2} && \text{(evaluate num.)} \\
1 - \cos(\theta_k)^2 &= 1 - \frac{\lambda_1^{2k} a_1^2}{\sum_{i=1}^m \lambda_i^{2k} a_i^2} && \text{(negate \& add 1 to both sides)} \\
&= \frac{\sum_{i=1}^m \lambda_i^{2k} a_i^2 - \lambda_1^{2k} a_1^2}{\sum_{i=1}^m \lambda_i^{2k} a_i^2} && \text{(common den.)} \\
&= \frac{\sum_{i=2}^m \lambda_i^{2k} a_i^2}{\lambda_1^{2k} a_1^2 + \sum_{i=2}^m \lambda_i^{2k} a_i^2} && \text{(rearrange summation terms)} \\
&\leq \frac{\sum_{i=2}^m \lambda_i^{2k} a_i^2}{\lambda_1^{2k} a_1^2} && \left( \text{since } \sum_{i=2}^m \lambda_i^{2k} a_i^2 \geq 0 \right) \\
&= \frac{1}{a_1^2} \sum_{i=2}^m a_i^2 \left( \frac{\lambda_i}{\lambda_1} \right)^{2k} && \text{(QED)}
\end{aligned}$$

## 2 The LU iteration algorithm [30 pts]

We consider the following iteration, starting with some full rank  $\mathbf{G}_0 \in \mathbb{C}^{m \times m}$ :

$$\mathbf{Z}_k = \mathbf{A}\mathbf{G}_{k-1}, \quad (6)$$

$$\mathbf{G}_k \mathbf{R}_k = \mathbf{Z}_k, \quad \text{LU factorization with no pivoting,} \quad (7)$$

where  $\mathbf{G}_k$  is lower-triangular and  $\mathbf{R}_k$  is upper-triangular. We define

$$\mathbf{T}_k = \mathbf{G}_k^{-1} \mathbf{A} \mathbf{G}_k. \quad (8)$$

Find an algorithm that computes the sequence  $\mathbf{T}_k$  in a manner similar to the QR iteration. This algorithm is a variant of the QR iteration. The eigenvalues of  $\mathbf{A}$  appear on the diagonal of  $\mathbf{T}_k$ .

The LU iteration algorithm is similar to the QR iteration algorithm, except that we use the LU factorization instead of the QR factorization.

Here is the LU iteration algorithm:

- Initialize  $\mathbf{T}_0 = \mathbf{A}$ .
- For  $k = 1, 2, \dots$ :
  - Compute the LU factorization of  $\mathbf{T}_{k-1}$ :  $\mathbf{LU} = \mathbf{T}_{k-1}$ .
  - Set  $\mathbf{T}_k = \mathbf{UL}$ .

As  $k \rightarrow \infty$ , the diagonals of  $\mathbf{T}_k$  converge to the eigenvalues of  $\mathbf{A}$ .

*Note: I have implemented this algorithm in Julia, and it works for square matrices with real eigenvalues, similar to the base QR iteration algorithm. As expected (and proven in (4b) below), the algorithm converges twice as slowly as the QR iteration algorithm. However, following guidance on Piazza and in office hours, I provide further details and a different approach below. We can think of the following as deriving an algorithm similar to the "Simultaneous Iteration" algorithm, but using LU factorization instead of QR factorization.*

In the QR iteration algorithm, we have:

- $\mathbf{QR} = \mathbf{Z}_{k-1}$
- $\mathbf{Z}_k = \mathbf{RQ}$

We want to find expressions for  $\mathbf{T}_k$  in the LU iteration that is similar to  $\mathbf{RQ}$ , such that  $\mathbf{T}_k = \mathbf{G}_k^{-1} \mathbf{A} \mathbf{G}_k$ . That is, we want to find some matrix  $\mathbf{B}_k$  such that  $\mathbf{T}_k = \mathbf{R}_k \mathbf{B}_k \mathbf{G}_k = \mathbf{G}_k^{-1} \mathbf{A} \mathbf{G}_k$ :

$$\begin{aligned} \mathbf{T}_k &= \mathbf{G}_k^{-1} \mathbf{A} \mathbf{G}_k && \text{(given)} \\ \mathbf{R}_k \mathbf{B}_k \mathbf{G}_k &= \mathbf{G}_k^{-1} \mathbf{A} \mathbf{G}_k && (\mathbf{T}_k \rightarrow \text{desired matrix similar to } \mathbf{R}_k \mathbf{G}_k) \\ \mathbf{R}_k \mathbf{B}_k &= \mathbf{G}_k^{-1} \mathbf{A} && \text{(right-multiply by } \mathbf{G}_k^{-1}) \\ \mathbf{B}_k &= \mathbf{R}_k^{-1} \mathbf{G}_k^{-1} \mathbf{A} && \text{(left-multiply by } \mathbf{R}_k^{-1} \text{ to isolate } \mathbf{B}_k) \\ &= (\mathbf{G}_k \mathbf{R}_k)^{-1} \mathbf{A} && \text{(group r.h.s. inverses)} \\ &= \mathbf{Z}_k^{-1} \mathbf{A} && \text{(from (7) above)} \\ &= (\mathbf{G}_{k-1})^{-1} && \text{(from (6) above)} \end{aligned}$$

Thus, we have

$$\mathbf{T}_k = \mathbf{G}_k^{-1} \mathbf{A} \mathbf{G}_k = \mathbf{R}_k \mathbf{B}_k \mathbf{G}_k = \mathbf{R}_k (\mathbf{G}_{k-1})^{-1} \mathbf{G}_k.$$

Here is a "Simultaneous LU iteration" algorithm:

- Initialize  $\mathbf{T}_0 = \mathbf{A}$
- Initialize  $\mathbf{G}_0$  to be a full-rank matrix.
- For  $k = 1, 2, \dots$ :
  - Compute  $\mathbf{Z}_k = \mathbf{A} \mathbf{G}_{k-1}$ .
  - Perform the LU factorization of  $\mathbf{Z}_k$  with no pivoting to find  $\mathbf{G}_k \mathbf{R}_k = \mathbf{Z}_k$ .
  - Solve the system  $\mathbf{G}_{k-1} \mathbf{X} = \mathbf{G}_k$  for  $\mathbf{X}$  using forward substitution to obtain  $\mathbf{X} = (\mathbf{G}_{k-1})^{-1} \mathbf{G}_k$ . (Note that  $\mathbf{G}_{k-1}$  is unit lower-diagonal.)
  - Compute  $\mathbf{T}_k = \mathbf{R}_k \mathbf{X}$

As  $k \rightarrow \infty$ , the diagonals of  $\mathbf{T}_k$  converge to the eigenvalues of  $\mathbf{A}$ .

See attached `lu_iteration.jl` for Julia implementations of the "LU iteration" and "Simultaneous LU iteration" algorithms, along with other iteration algorithms.

### 3 Convergence of Orthogonal Iteration [40 pts]

In this problem, you will explore the convergence behavior of the orthogonal iteration algorithm. In the problems below, only the asymptotic convergence will match the theoretical estimates. For small  $k$ , you may see deviations. To simplify grading, please use the provided (empty) file `HW4_your_code.jl` for your code. Please also attach all results to your report, both plots and print statements. The TAs should be able to grade your homework without running your code.

#### (a) [10 pts]

Write Julia code to create a matrix in  $\mathbb{R}^{m \times m}$  of size  $m = 8$  with eigenvalues  $1, 3, 9, \dots, 3^{m-1}$ . Explain your code (as comments *and* in your report) and describe what it does.

The method `random_matrix(m,  $\Lambda$ )` performs the following to produce a matrix in  $\mathbb{R}^{m \times m}$  with eigenvalues  $\Lambda$  using the following algorithm:

1. Create a random matrix  $\mathbf{A} \in \mathbb{R}^{m \times m}$  of normally distributed values with  $\mu = 0$  and  $\sigma = 1$ .
2. Perform a QR decomposition of  $\mathbf{A}$  to create an orthogonal matrix  $\mathbf{Q}$ .
3. Return  $\mathbf{Q} \text{diag}(\Lambda) \mathbf{Q}^T$ .

#### (b) [10 pts]

Implement the orthogonal iteration algorithm. Print the values along the diagonal of  $\mathbf{R}_k$  at each iteration  $k$  for  $k = 1, \dots, 5$ . Print each number using at most four significant digits.

```
$ julia --project=. HW5_your_code.jl
[-152.1, -555.1, -140.0, -47.68, -215.6, -3.844, -10.86, 4.508]
[1888.0, 639.5, 274.7, 62.78, 40.42, 3.566, 6.945, 1.098]
[2177.0, 716.7, 247.8, -76.64, -28.58, 6.24, 4.286, 1.01]
[2186.0, 727.7, 243.5, 80.46, 27.18, -8.452, -3.191, 1.001]
[2187.0, 728.9, 243.1, 80.94, 27.02, 8.933, 3.022, 1.0]
```

**(c) [10 pts]**

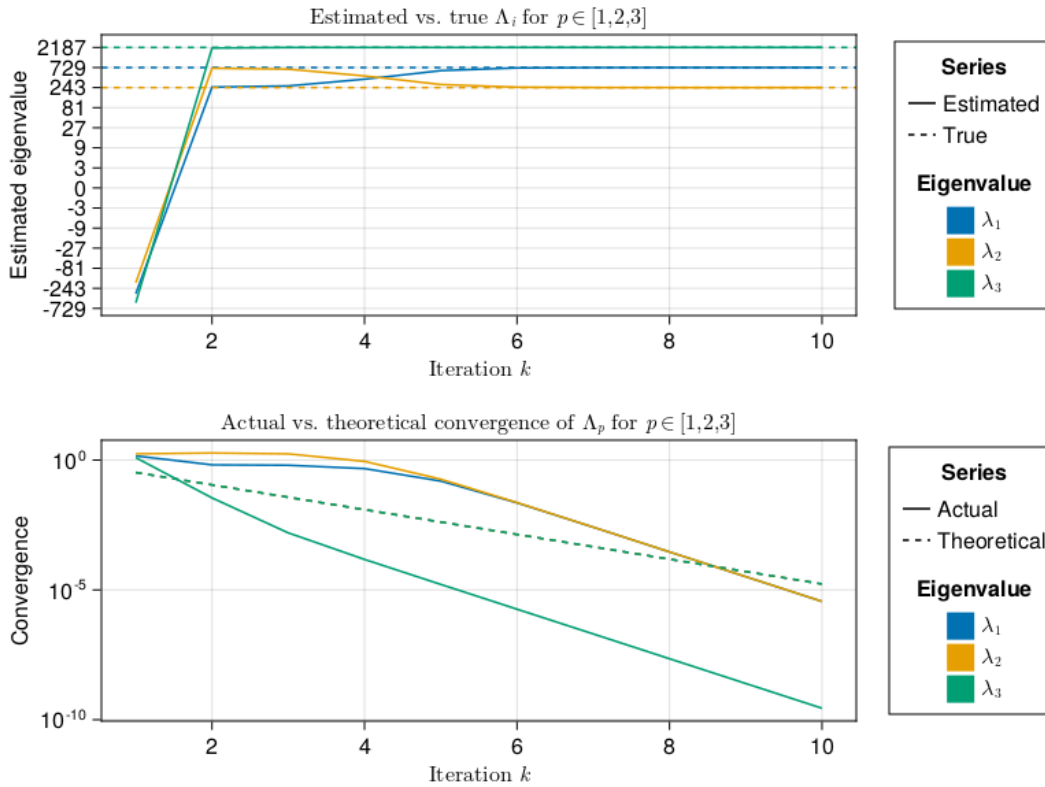
Considering entry  $p$  along the diagonal, plot the convergence of the  $p$ th eigenvalue. Choose  $p = 1, 2$ , and  $3$ . Compare with the theoretical rate of convergence at step  $k$  for entry  $p$ , which is given by

$$\max(|\lambda_{p+1}/\lambda_p|^k, |\lambda_p/\lambda_{p-1}|^k), \quad 1 < p < m, \quad (9)$$

$$|\lambda_2/\lambda_1|^k, \quad p = 1 \quad (10)$$

$$|\lambda_m/\lambda_{m-1}|^k, \quad p = m. \quad (11)$$

Use a semi-log plot (meaning that the  $y$ -axis of your plot should be logarithmic).



**Figure 1** Eigenvalue convergence for  $p = 1, 2, 3$

Figure 1 shows plots of the estimated and true eigenvalues, and the actual and theoretical convergence rates for  $p = 1, 2, 3$  and  $k \in 1, 2, \dots, 10$ . Notice that the actual convergence rates are somewhat faster than the theoretical convergence rates.

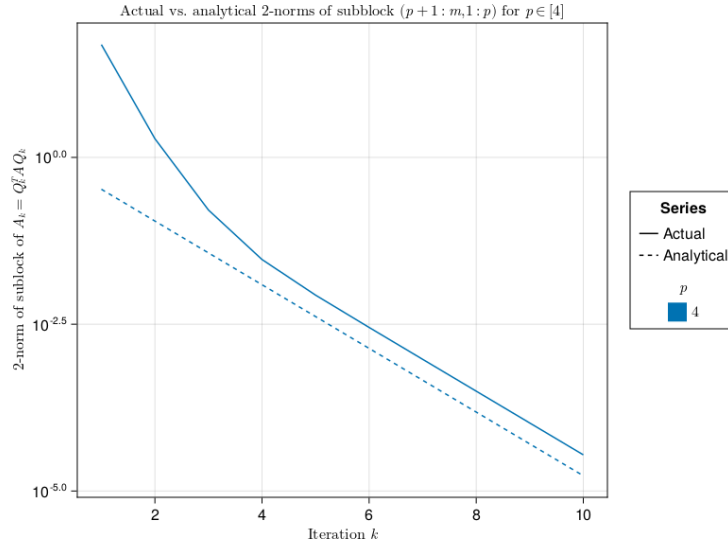
*Note: Depending on the generated random matrix, the convergence rates differ between runs.*

**(d) [10 pts]**

Consider the block  $(p+1:m, 1:p)$  in the matrix

$$\mathbf{A}_k = \mathbf{Q}_k^T \mathbf{A} \mathbf{Q}_k. \quad (12)$$

Plot the 2-norm of this block as a function of  $k$  for  $p = 4$ . Compare with the analytical estimate, which states that it should decay like  $|\lambda_{p+1}/\lambda_p|^k$ .



**Figure 2** Analytical vs. actual 2-norm decay rates

Figure 2 shows a plot of the true and analytical 2-norms of these subblocks. The analytical norm estimates have exactly the same decay rate as the true norm estimates, though the magnitudes are lower.

*Note: The exact magnitude of the difference in analytic vs true norms is dependent on the generated random matrix.*

## 4 LU and QR iteration [20 bonus pts]

We consider the LU iteration applied to a symmetric matrix. Assume that  $\mathbf{A}_0 \in \mathbb{R}^{m \times m}$  is symmetric and positive-definite. We produce a sequence of matrices  $\mathbf{A}_i$  using the following algorithm:

$$\mathbf{A}_i = \mathbf{G}_i^T \mathbf{G}_i, \quad \mathbf{A}_{i+1} = \mathbf{G}_i \mathbf{G}_i^T, \quad (13)$$

where  $\mathbf{G}_i$  are upper-triangular matrices.

Consider now  $\mathbf{A}'$ , the matrix obtained after one step of the QR iteration, that is,

$$\mathbf{A}_0 = \mathbf{Q}\mathbf{R}, \quad \mathbf{A}' = \mathbf{R}\mathbf{Q}. \quad (14)$$

We assume that the diagonal of  $\mathbf{R}$  is positive.

**(a) [10 bonus pts]**

Use  $\mathbf{A}_0^2$  to show that

$$\mathbf{R} = \mathbf{G}_1 \mathbf{G}_0 \quad (15)$$

$$\begin{aligned}
A_0^2 &= A_0^2 \\
A_0^T A_0 &= A_0 A_0 && \text{(since } A \text{ is symmetric, } A = A^T) \\
(QR)^T(QR) &= (G_0^T G_0)(G_0^T G_0) && \text{(substitute with LU/QR definitions)} \\
R^T R &= G_0^T (G_0 G_0^T) G_0 && \text{(simplify l.h.s., regroup r.h.s.)} \\
R^T R &= G_0^T A_1 G_0 && \text{(by def. of } A_1) \\
R^T R &= G_0^T (G_1^T G_1) G_0 && \text{(by def. of } A_1) \\
R^T R &= (G_1 G_0)^T G_1 G_0 && \text{(rearrange r.h.s.)} \\
R &= G_1 G_0 && \text{(QED)}
\end{aligned}$$

**(b) [10 bonus pts]**

Show that

$$A' = A_2 \tag{16}$$

$$\begin{aligned}
A' &= A_2 && \text{(given)} \\
A'^2 &= A_2^2 && \text{(square both sides)} \\
(RQ)(RQ) &= (G_1 G_1^T)(G_1 G_1^T) && \text{(substitute)} \\
RA_0 Q &= G_1 A_1 G_1^T && \text{(substitute)} \\
R(R^T Q^T)Q &= G_1 A_1 G_1^T && (A_0 \text{ is symmetric)} \\
RR^T &= G_1 A_1 G_1^T && \text{(simplify l.h.s.)} \\
(G_1 G_0)(G_1 G_0)^T &= G_1 (G_0 G_0^T) G_1^T && (A_1 = G_0 G_0^T, R = G_1 G_0 \text{ (a)}) \\
G_1 G_0 G_0^T G_1^T &= G_1 G_0 G_0^T G_1^T && \text{(rearrange l.h.s. QED)}
\end{aligned}$$

## 5 Sensitivity of Eigenvalues [20 bonus pts]

We are interested in determining the sensitivity of eigenvalues with respect to perturbations in the matrix.

Prove that if  $A \in \mathbb{C}^{m \times m}$  is diagonalizable with  $A = V \Lambda V^{-1}$ , and if  $\mu$  is an eigenvalue of  $A + E$ ,

$$\min_{\lambda \in \lambda(A)} |\mu - \lambda| \leq \kappa_p(V) \|E\|_p.$$

Here  $\|\cdot\|_p$  denotes the  $p$ -norm and  $\kappa_p(V) := \|V\|_p \|V^{-1}\|_p$  is the condition number associated with the  $p$ -norm.

*Hint:* assume that  $I + F$  is singular. Then, there is an  $x \neq 0$  such that  $x + Fx = 0$ . Therefore,

$\mathbf{F}\mathbf{x} = -\mathbf{x}$  and  $\|\mathbf{F}\|_p \geq 1$ . For the proof, identify the proper singular matrix and use this result.

$$\begin{aligned}
(\mathbf{A} + \mathbf{E})\mathbf{v} &= \mu\mathbf{v} && \text{(let } \mathbf{v} \text{ be the eigenvector for eigenvalue } \mu) \\
\mathbf{A}\mathbf{v} + \mathbf{E}\mathbf{v} &= \mu\mathbf{v} && \text{(distribute } \mathbf{v}) \\
\mathbf{V}\mathbf{\Lambda}\mathbf{V}^{-1}\mathbf{v} + \mathbf{E}\mathbf{v} &= \mu\mathbf{v} && \text{(substitute } \mathbf{A} = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^{-1}) \\
\mathbf{V}\mathbf{\Lambda}\mathbf{c} + \mathbf{E}\mathbf{V}\mathbf{c} &= \mu\mathbf{V}\mathbf{c} && \text{(Let } \mathbf{v} = \mathbf{V}\mathbf{c}) \\
\mathbf{V}\mathbf{\Lambda} + \mathbf{E}\mathbf{V} &= \mu\mathbf{V} && \text{(drop common } \mathbf{c}) \\
\mathbf{E}\mathbf{V} &= (\mu\mathbf{I} - \mathbf{\Lambda})\mathbf{V} && \text{(rearrange)} \\
\mathbf{F} := (\mathbf{V}^{-1}\mathbf{E}\mathbf{V}) &= \mathbf{V}^{-1}(\mu\mathbf{I} - \mathbf{\Lambda})\mathbf{V} && \text{(left-multiply by } \mathbf{V}^{-1})
\end{aligned}$$

Observe that  $\mathbf{I} + \mathbf{F}$  is singular. Thus, there exists a nonzero vector  $\mathbf{x}$  such that

$$\begin{aligned}
\mathbf{x} + \mathbf{F}\mathbf{x} &= \mathbf{0} \\
\mathbf{F}\mathbf{x} &= -\mathbf{x} \\
\|\mathbf{F}\|_p &\geq 1 && \text{(property of induced matrix norms)} \\
\|\mathbf{V}^{-1}(\mu\mathbf{I} - \mathbf{\Lambda})\mathbf{V}\|_p &\geq 1 && \text{(substitute } \mathbf{F}) \\
\|\mathbf{V}^{-1}\|_p \|\mu\mathbf{I} - \mathbf{\Lambda}\|_p \|\mathbf{V}\|_p &\geq 1 && \text{(submultiplicative property of matrix norms)} \\
\|\mu\mathbf{I} - \mathbf{\Lambda}\|_p &\geq \frac{1}{\kappa_p(\mathbf{V})} && \text{(divide by } \kappa_p(\mathbf{V}) := \|\mathbf{V}\|_p \|\mathbf{V}^{-1}\|_p) \\
\max_{\lambda \in \lambda(\mathbf{A})} |\mu - \lambda| &\geq \frac{1}{\kappa_p(\mathbf{V})} && \text{(def. of } p\text{-norm)} \\
\min_{\lambda \in \lambda(\mathbf{A})} |\mu - \lambda| &\leq \kappa_p(\mathbf{V}) && \text{(take the reciprocal of the inequality)} \\
\min_{\lambda \in \lambda(\mathbf{A})} |\mu - \lambda| &\leq \kappa_p(\mathbf{V}) \|\mathbf{E}\|_p && (\|\mathbf{E}\|_p \geq 0. \text{ QED})
\end{aligned}$$