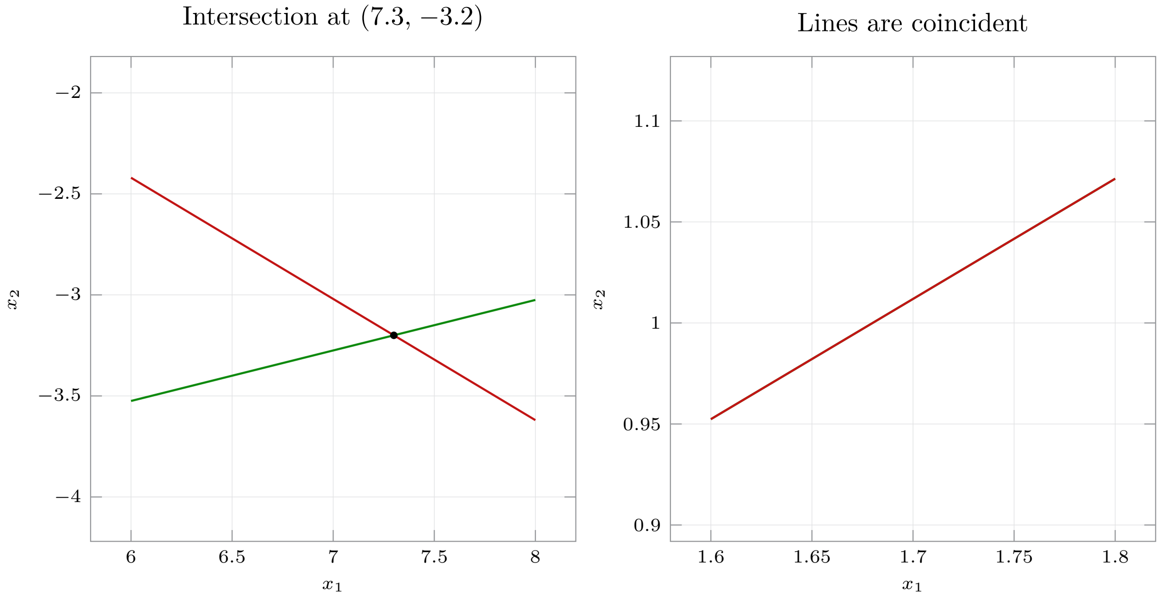
Chapter 20

Numeric Linear Algebra

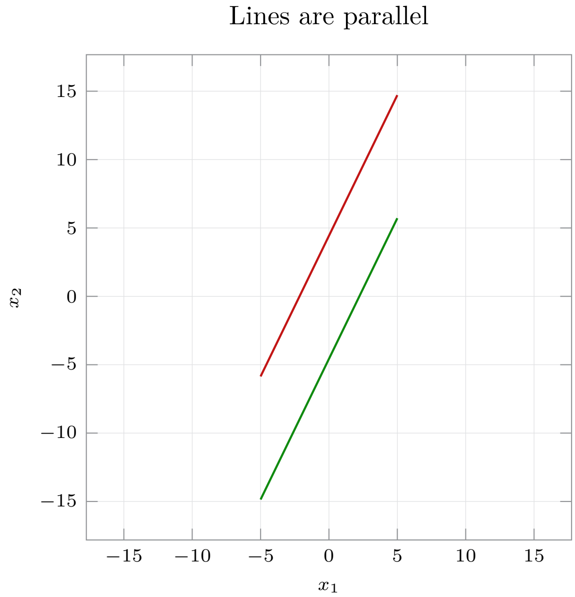
20.1 Linear Systems: Gauss Elimination

1. Solving geometrically,

2. Solving geometrically,



3. Solving geometrically,



4. Solving by Gauss elimination with pivoting,

$$\tilde{\mathbf{A}} = \left[\begin{array}{cc|c} 6 & 2 & -3 \\ 4 & -2 & 6 \end{array}\right] \qquad = \left[\begin{array}{cc|c} 6 & 2 & -3 \\ 0 & -10/3 & 8 \end{array}\right] \tag{20.1.1}$$

$$x_2 = -2.4 \qquad x_1 = 0.3 \tag{20.1.2}$$

5. Solving by Gauss elimination with pivoting,

$$\tilde{\mathbf{A}} = \left[\begin{array}{cc|c} 2 & -8 & -4 \\ 3 & 1 & 7 \end{array}\right] \qquad = \left[\begin{array}{cc|c} 3 & 1 & 7 \\ 2 & -8 & -4 \end{array}\right] \tag{20.1.3}$$

$$= \left[\begin{array}{cc|c} 3 & 1 & 7 \\ 0 & -\frac{26}{3} & -\frac{26}{3} \end{array}\right] \tag{20.1.4}$$

$$x_2 = 1 \qquad x_1 = 2 \tag{20.1.5}$$

6. Solving by Gauss elimination with pivoting,

$$\tilde{\mathbf{A}} = \left[\begin{array}{cc|c} 25.38 & -15.48 & 30.60 \\ -14.10 & 8.6 & -17 \end{array}\right] \qquad = \left[\begin{array}{cc|c} 25.38 & -15.48 & 30.60 \\ 0 & 0 & 0 \end{array}\right] \tag{20.1.6}$$

$$x_2 = t \qquad x_1 = \frac{30.6 + 15.48t}{25.38} \tag{20.1.7}$$

7. Solving by Gauss elimination with pivoting,

$$\tilde{\mathbf{A}} = \left[\begin{array}{ccc|c} -3 & 6 & -9 & -46.725 \\ 1 & -4 & 3 & 19.571 \\ 2 & 5 & -7 & -20.073 \end{array}\right] \qquad = \left[\begin{array}{ccc|c} -3 & 6 & -9 & -46.725 \\ 0 & -2 & 0 & 3.996 \\ 0 & 9 & -13 & -51.223 \end{array}\right] \tag{20.1.8}$$

$$= \left[\begin{array}{ccc|c} -3 & 6 & -9 & -46.725 \\ 0 & -2 & 0 & 3.996 \\ 0 & 0 & -13 & -33.241 \end{array}\right] \qquad x_3 = 2.557 \tag{20.1.9}$$

$$x_2 = -1.998 \qquad x_1 = 3.908 \tag{20.1.10}$$

8. Solving by Gauss elimination with pivoting,

$$\tilde{\mathbf{A}} = \left[\begin{array}{ccc|c} 5 & 3 & 1 & 2 \\ 0 & -4 & 8 & -3 \\ 10 & -6 & 26 & 0 \end{array}\right] \qquad = \left[\begin{array}{ccc|c} 10 & -6 & 26 & 0 \\ 5 & 3 & 1 & 2 \\ 0 & -4 & 8 & -3 \end{array}\right] \tag{20.1.11}$$

$$= \left[\begin{array}{ccc|c} 10 & -6 & 26 & 0 \\ 0 & 6 & -12 & 2 \\ 0 & -4 & 8 & -3 \end{array}\right] \qquad = \left[\begin{array}{ccc|c} 10 & -6 & 26 & 0 \\ 0 & 6 & -12 & 2 \\ 0 & 0 & 0 & -5/3 \end{array}\right] \tag{20.1.12}$$

$$\text{no solution} \tag{20.1.13}$$

9. Solving by Gauss elimination with pivoting,

$$\tilde{\mathbf{A}} = \left[\begin{array}{ccc|c} 0 & 6 & 13 & 137.86 \\ 6 & 0 & -8 & -85.88 \\ 13 & -8 & 0 & 178.54 \end{array}\right] \qquad = \left[\begin{array}{ccc|c} 13 & -8 & 0 & 178.54 \\ 0 & 6 & 13 & 137.86 \\ 6 & 0 & -8 & -85.88 \end{array}\right] \tag{20.1.14}$$

$$= \left[\begin{array}{ccc|c} 13 & -8 & 0 & 178.54 \\ 0 & 6 & 13 & 137.86 \\ 0 & \frac{48}{13} & -8 & -168.28 \end{array}\right] \qquad = \left[\begin{array}{ccc|c} 13 & -8 & 0 & 178.54 \\ 0 & 6 & 13 & 137.86 \\ 0 & 0 & -16 & -253.12 \end{array}\right] \tag{20.1.15}$$

$$x_3 = 15.82 \tag{20.1.16}$$

$$x_2 = -11.3 \qquad x_1 = 6.78 \tag{20.1.17}$$

10. Solving by Gauss elimination with pivoting,

$$\tilde{\mathbf{A}} = \left[\begin{array}{ccc|c} 4 & 4 & 2 & 0 \\ 3 & -1 & 2 & 0 \\ 3 & 7 & 1 & 0 \end{array}\right] \qquad = \left[\begin{array}{ccc|c} 4 & 4 & 2 & 0 \\ 0 & -4 & 0.5 & 0 \\ 0 & 4 & -0.5 & 0 \end{array}\right] \tag{20.1.18}$$

$$= \left[\begin{array}{ccc|c} 4 & 4 & 2 & 0 \\ 0 & -4 & 0.5 & 0 \\ 0 & 0 & 0 & 0 \end{array}\right] \tag{20.1.19}$$

$$x_3 = t \qquad x_2 = \frac{t}{8} \tag{20.1.20}$$

$$x_1 = -\frac{5t}{8} \tag{20.1.21}$$

11. Solving by Gauss elimination with pivoting,

$$\tilde{\mathbf{A}} = \left[\begin{array}{ccc|c} 3.4 & -6.12 & -2.72 & 0 \\ -1 & 1.8 & 0.8 & 0 \\ 2.7 & -4.86 & 2.16 & 0 \end{array}\right] \qquad = \left[\begin{array}{ccc|c} 3.4 & -6.12 & -2.72 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 4.32 & 0 \end{array}\right] \tag{20.1.22}$$

$$x_3 = 0 \qquad x_2 = t \tag{20.1.23}$$

$$x_1 = 1.8t \tag{20.1.24}$$

12. Solving by Gauss elimination with pivoting,

$$\tilde{\mathbf{A}} = \left[\begin{array}{ccc|c} 5 & 3 & 1 & 2 \\ 0 & -4 & 8 & -3 \\ 10 & -6 & 26 & 0 \end{array}\right] \qquad = \left[\begin{array}{ccc|c} 10 & -6 & 26 & 0 \\ 0 & 6 & -12 & 2 \\ 0 & -4 & 8 & -3 \end{array}\right] \tag{20.1.25}$$

$$= \left[\begin{array}{ccc|c} 10 & -6 & 26 & 0 \\ 0 & 6 & -12 & 2 \\ 0 & 0 & 0 & -5/3 \end{array}\right] \qquad \text{no solution} \tag{20.1.26}$$

13. Solving by Gauss elimination with pivoting,

$$\tilde{\mathbf{A}} = \left[\begin{array}{ccc|c} 0 & 3 & 5 & 1.20736 \\ 3 & -4 & 0 & -2.34066 \\ 5 & 0 & 6 & -0.329193 \end{array}\right] = \left[\begin{array}{ccc|c} 5 & 0 & 6 & -0.329193 \\ 3 & -4 & 0 & -2.34066 \\ 0 & 3 & 5 & 1.20736 \end{array}\right] \tag{20.1.27}$$

$$= \left[\begin{array}{ccc|c} 5 & 0 & 6 & -0.329193 \\ 0 & -4 & \frac{-18}{5} & -2.1431442 \\ 0 & 3 & 5 & 1.20736 \end{array}\right] = \left[\begin{array}{ccc|c} 5 & 0 & 6 & -0.329193 \\ 0 & -4 & \frac{-18}{5} & -2.1431442 \\ 0 & 0 & 2.3 & -0.39999815 \end{array}\right] \tag{20.1.28}$$

$$x_3 = -0.173912 \qquad x_2 = 0.692307 \tag{20.1.29}$$

$$x_1 = 0.142856 \tag{20.1.30}$$

14. Solving by Gauss elimination with pivoting,

$$\tilde{\mathbf{A}} = \left[\begin{array}{ccc|c} -47 & 4 & -7 & -118 \\ 19 & -3 & 2 & 43 \\ -15 & 5 & 0 & -25 \end{array}\right] = \left[\begin{array}{ccc|c} -47 & 4 & -7 & -118 \\ 0 & -\frac{65}{47} & -\frac{39}{47} & -\frac{221}{47} \\ 0 & \frac{175}{47} & \frac{105}{47} & \frac{595}{47} \end{array}\right] \tag{20.1.31}$$

$$= \left[\begin{array}{ccc|c} -47 & 4 & -7 & -118 \\ 0 & -65 & -39 & -221 \\ 0 & 0 & 0 & 0 \end{array}\right] \qquad x_3 = t \tag{20.1.32}$$

$$x_2 = \frac{17}{5} - \frac{3t}{5} \qquad x_1 = \frac{14}{5} - \frac{t}{5} \tag{20.1.33}$$

A computer may give a unique solution to this problem because of rounding erorrs arising from converting the rational numbers into machine numbers.

15. Solving by Gauss elimination with pivoting,

$$\tilde{\mathbf{A}} = \left[\begin{array}{cccc|c} 0 & 2.2 & 1.5 & -3.3 & -9.3 \\ 0.2 & 1.8 & 0 & 4.2 & 9.24 \\ -1 & -3.1 & 2.5 & 0 & -8.7 \\ 0.5 & 0 & -3.8 & 1.5 & 11.94 \end{array}\right] = \left[\begin{array}{cccc|c} -1 & -3.1 & 2.5 & 0 & -8.7 \\ 0 & 2.2 & 1.5 & -3.3 & -9.3 \\ 0.5 & 0 & -3.8 & 1.5 & 11.94 \\ 0.2 & 1.8 & 0 & 4.2 & 9.24 \end{array}\right] \tag{20.1.34}$$

$$= \left[\begin{array}{cccc|c} -1 & -3.1 & 2.5 & 0 & -8.7 \\ 0 & 2.2 & 1.5 & -3.3 & -9.3 \\ 0 & -1.55 & -2.55 & 1.5 & 7.59 \\ 0 & 1.18 & 0.5 & 4.2 & 7.5 \end{array}\right] = \left[\begin{array}{cccc|c} -1 & -3.1 & 2.5 & 0 & -8.7 \\ 0 & 2.2 & 1.5 & -3.3 & -9.3 \\ 0 & 0 & -\frac{657}{440} & -\frac{33}{40} & \frac{2283}{2200} \\ 0 & 0 & -\frac{67}{220} & \frac{597}{100} & \frac{13737}{1100} \end{array}\right] \tag{20.1.35}$$

$$= \left[\begin{array}{cccc|c} -1 & -3.1 & 2.5 & 0 & -8.7 \\ 0 & 2.2 & 1.5 & -3.3 & -9.3 \\ 0 & 0 & -657 & -363 & \frac{2283}{5} \\ 0 & 0 & -67 & \frac{6567}{5} & \frac{13737}{5} \end{array}\right] = \left[\begin{array}{cccc|c} -1 & -3.1 & 2.5 & 0 & -8.7 \\ 0 & 2.2 & 1.5 & -3.3 & -9.3 \\ 0 & 0 & -657 & -363 & \frac{2283}{5} \\ 0 & 0 & 0 & \frac{1478708}{1095} & \frac{2957416}{1095} \end{array}\right] \tag{20.1.36}$$

$$x_4 = 2 \qquad x_3 = -\frac{9}{5} \tag{20.1.37}$$

$$x_2 = 0 \qquad x_1 = \frac{21}{5} \tag{20.1.38}$$

A computer may give a different solution to this problem because of rounding erorrs arising from converting the rational numbers into machine numbers.

16. Solving by Gauss elimination with pivoting,

$$\tilde{\mathbf{A}} = \left[\begin{array}{cccc|c} 3.2 & 1.6 & 0 & 0 & -0.8 \\ 1.6 & -0.8 & 2.4 & 0 & 16 \\ 0 & 2.4 & -4.8 & 3.6 & -39 \\ 0 & 0 & 3.6 & 2.4 & 10.2 \end{array}\right] = \left[\begin{array}{cccc|c} 3.2 & 1.6 & 0 & 0 & -0.8 \\ 0 & -1.6 & 2.4 & 0 & 16.4 \\ 0 & 2.4 & -4.8 & 3.6 & -39 \\ 0 & 0 & 3.6 & 2.4 & 10.2 \end{array}\right] \tag{20.1.39}$$

$$= \left[\begin{array}{cccc|c} 3.2 & 1.6 & 0 & 0 & -0.8 \\ 0 & -1.6 & 2.4 & 0 & 16.4 \\ 0 & 0 & -1.2 & 3.6 & -14.4 \\ 0 & 0 & 3.6 & 2.4 & 10.2 \end{array}\right] = \left[\begin{array}{cccc|c} 3.2 & 1.6 & 0 & 0 & -0.8 \\ 0 & -1.6 & 2.4 & 0 & 16.4 \\ 0 & 0 & -1.2 & 3.6 & -14.4 \\ 0 & 0 & 0 & 13.2 & -33 \end{array}\right] \tag{20.1.40}$$

$$x_4 = -2.5 \qquad x_3 = 4.5 \tag{20.1.41}$$

$$x_2 = -3.5 \qquad x_1 = 1.5 \tag{20.1.42}$$

17. Coded in `numpy`. Only the sorting subroutine was used for partial pivoting. Results for all problems in this set used to verify.

18. Linear Systems

(a) The given system is,

$$\tilde{\mathbf{A}} = \left[\begin{array}{cc|c} a & 1 & b \\ 1 & 1 & 3 \end{array}\right] \qquad \det \mathbf{A} = a - 1 \tag{20.1.43}$$

$$a = 1,\ b = 3 \qquad \Longrightarrow \text{infinitely many solutions} \tag{20.1.44}$$

$$a = 1,\ b \neq 3 \qquad \Longrightarrow \text{no solution} \tag{20.1.45}$$

$$a \neq 1 \qquad \Longrightarrow \text{unique solution} \tag{20.1.46}$$

(b) For the first system,

$$\tilde{\mathbf{A}} = \left[\begin{array}{ccc|c} 1 & 1 & 1 & 3 \\ 4 & 2 & -1 & 5 \\ 9 & 5 & -1 & 13 \end{array}\right] = \left[\begin{array}{ccc|c} 9 & 5 & -1 & 13 \\ 1 & 1 & 1 & 3 \\ 4 & 2 & -1 & 5 \end{array}\right] \tag{20.1.47}$$

$$= \left[\begin{array}{ccc|c} 9 & 5 & -1 & 13 \\ 0 & \frac{4}{9} & \frac{10}{9} & \frac{14}{9} \\ 0 & -\frac{2}{9} & -\frac{5}{9} & -\frac{7}{9} \end{array}\right] = \left[\begin{array}{ccc|c} 9 & 5 & -1 & 13 \\ 0 & 4 & 10 & 14 \\ 0 & -2 & -5 & -7 \end{array}\right] \tag{20.1.48}$$

$$= \left[\begin{array}{ccc|c} 9 & 5 & -1 & 13 \\ 0 & 4 & 10 & 14 \\ 0 & 0 & 0 & 0 \end{array}\right] \qquad \text{infinitely many solutions} \tag{20.1.49}$$

For the second system,

$$\tilde{\mathbf{A}} = \left[\begin{array}{ccc|c} 1 & 1 & 1 & 3 \\ 4 & 2 & -1 & 5 \\ 9 & 5 & -1 & 12 \end{array}\right] = \left[\begin{array}{ccc|c} 9 & 5 & -1 & 12 \\ 1 & 1 & 1 & 3 \\ 4 & 2 & -1 & 5 \end{array}\right] \tag{20.1.50}$$

$$= \left[\begin{array}{ccc|c} 9 & 5 & -1 & 13 \\ 0 & \frac{4}{9} & \frac{10}{9} & \frac{5}{3} \\ 0 & -\frac{2}{9} & -\frac{5}{9} & -\frac{7}{9} \end{array}\right] = \left[\begin{array}{ccc|c} 9 & 5 & -1 & 13 \\ 0 & 4 & 10 & 15 \\ 0 & -2 & -5 & -7 \end{array}\right] \tag{20.1.51}$$

$$= \left[\begin{array}{ccc|c} 9 & 5 & -1 & 13 \\ 0 & 4 & 10 & 14 \\ 0 & 0 & 0 & 0.5 \end{array}\right] \qquad \text{no solution} \tag{20.1.52}$$

(c) Rounding errors might change the computed coefficient determinant to be nonzero, which makes the trivial solution a possible answer, when the system is homogeneous.

(d) First solving without pivoting,

$$\hat{\mathbf{A}} = \left[\begin{array}{cc|c} \epsilon & 1 & 1 \\ 1 & 1 & 2 \end{array}\right] \qquad = \left[\begin{array}{cc|c} \epsilon & 1 & 1 \\ 0 & 1-\frac{1}{\epsilon} & 2-\frac{1}{\epsilon} \end{array}\right] \tag{20.1.53}$$

$$x_2 = \frac{2\epsilon - 1}{\epsilon - 1} \qquad \lim_{\epsilon \to 0} x_2 = 1 \tag{20.1.54}$$

$$x_1 = \frac{-1}{\epsilon - 1} \qquad \lim_{\epsilon \to 0} x_1 = 1 \tag{20.1.55}$$

$$x_2^* = 1 \qquad x_1^* = 0 \tag{20.1.56}$$

Now, with pivoting,

$$\hat{\mathbf{A}} = \left[\begin{array}{cc|c} 1 & 1 & 2 \\ \epsilon & 1 & 1 \end{array}\right] \qquad = \left[\begin{array}{cc|c} 1 & 1 & 2 \\ 0 & 1-\epsilon & 1-2\epsilon \end{array}\right] \tag{20.1.57}$$

$$x_2^* = 1 \qquad x_1^* = 1 \tag{20.1.58}$$

(e) First solving without pivoting, and $3S$

$$\hat{\mathbf{A}} = \left[\begin{array}{cc|c} 4.03 & 2.16 & -4.61 \\ 6.21 & 3.35 & -7.19 \end{array}\right] \qquad = \left[\begin{array}{cc|c} 4.03 & 2.16 & -4.61 \\ 0 & 0.02 & -0.09 \end{array}\right] \tag{20.1.59}$$

$$x_2^* = -4.5 \qquad x_1^* = 1.27 \tag{20.1.60}$$

Now, with pivoting,

$$\hat{\mathbf{A}} = \left[\begin{array}{cc|c} 6.21 & 3.35 & -7.19 \\ 4.03 & 2.16 & -4.61 \end{array}\right] \qquad = \left[\begin{array}{cc|c} 6.21 & 3.35 & -7.19 \\ 0 & -0.01 & 0.06 \end{array}\right] \tag{20.1.61}$$

$$x_2^* = -6 \qquad x_1^* = 2.08 \tag{20.1.62}$$

First solving without pivoting, and $4S$

$$\hat{\mathbf{A}} = \left[\begin{array}{cc|c} 4.03 & 2.16 & -4.61 \\ 6.21 & 3.35 & -7.19 \end{array}\right] \qquad = \left[\begin{array}{cc|c} 4.03 & 2.16 & -4.61 \\ 0 & 0.022 & -0.086 \end{array}\right] \tag{20.1.63}$$

$$x_2^* = -3.909 \qquad x_1^* = 0.9511 \tag{20.1.64}$$

Now, with pivoting,

$$\hat{\mathbf{A}} = \left[\begin{array}{cc|c} 6.21 & 3.35 & -7.19 \\ 4.03 & 2.16 & -4.61 \end{array}\right] \qquad = \left[\begin{array}{cc|c} 6.21 & 3.35 & -7.19 \\ 0 & -0.014 & 0.055 \end{array}\right] \tag{20.1.65}$$

$$x_2^* = -3.928 \qquad x_1^* = 0.9613 \tag{20.1.66}$$

The true solution is $(-4, 1)$ and an increase in accuracy is observed when more significant digits are used.

20.2 Linear Systems: LU-Factorization, Matrix Inversion

1. LU factorization coded in `numpy`

$$\mathbf{A} = \begin{bmatrix} 4 & 5 \\ 12 & 14 \end{bmatrix} \qquad \mathbf{LU} = \begin{bmatrix} 1 & 0 \\ 3 & 1 \end{bmatrix} \begin{bmatrix} 4 & 5 \\ 0 & -1 \end{bmatrix} \tag{20.2.1}$$

$$\mathbf{Ly} = \mathbf{b} = \begin{bmatrix} 14 \\ 36 \end{bmatrix} \qquad \mathbf{y} = \begin{bmatrix} 14 \\ -6 \end{bmatrix} \tag{20.2.2}$$

$$\mathbf{Ux} = \mathbf{y} \qquad \mathbf{x} = \begin{bmatrix} -4 \\ 6 \end{bmatrix} \tag{20.2.3}$$

2. LU factorization coded in `numpy`

$$\mathbf{A} = \begin{bmatrix} 2 & 9 \\ 3 & -5 \end{bmatrix} \qquad \mathbf{LU} = \begin{bmatrix} 1 & 0 \\ 1.5 & 1 \end{bmatrix} \begin{bmatrix} 2 & 9 \\ 0 & -18.5 \end{bmatrix} \tag{20.2.4}$$

$$\mathbf{Ly} = \mathbf{b} = \begin{bmatrix} 82 \\ -62 \end{bmatrix} \qquad \mathbf{y} = \begin{bmatrix} 82 \\ -185 \end{bmatrix} \tag{20.2.5}$$

$$\mathbf{Ux} = \mathbf{y} \qquad \mathbf{x} = \begin{bmatrix} -4 \\ 10 \end{bmatrix} \tag{20.2.6}$$

3. LU factorization coded in `numpy`

$$\tilde{\mathbf{A}} = \left[\begin{array}{ccc|c} 5 & 4 & 1 & 6.8 \\ 10 & 9 & 4 & 17.6 \\ 10 & 13 & 15 & 38.4 \end{array}\right] \qquad \mathbf{LU} = \begin{bmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ 2 & 5 & 1 \end{bmatrix}\begin{bmatrix} 5 & 4 & 1 \\ 0 & 1 & 2 \\ 0 & 0 & 3 \end{bmatrix} \tag{20.2.7}$$

$$\mathbf{Ly} = \mathbf{b} \quad \Longrightarrow \quad \mathbf{y} = \begin{bmatrix} 6.8 \\ 4 \\ 4.8 \end{bmatrix} \qquad \mathbf{Ux} = \mathbf{y} \quad \Longrightarrow \quad \mathbf{x} = \begin{bmatrix} 0.4 \\ 0.8 \\ 1.6 \end{bmatrix} \tag{20.2.8}$$

4. LU factorization coded in `numpy`

$$\tilde{\mathbf{A}} = \left[\begin{array}{ccc|c} 2 & 0 & 2 & 0 \\ -2 & 2 & 1 & 0 \\ 1 & 2 & -2 & 18 \end{array}\right] \qquad \mathbf{LU} = \begin{bmatrix} 1 & 0 & 0 \\ -1 & 1 & 0 \\ 0.5 & 0.5 & 1 \end{bmatrix}\begin{bmatrix} 2 & 1 & 2 \\ 0 & 3 & 3 \\ 0 & 0 & -4.5 \end{bmatrix} \tag{20.2.9}$$

$$\mathbf{Ly} = \mathbf{b} \quad \Longrightarrow \quad \mathbf{y} = \begin{bmatrix} 0 \\ 0 \\ 18 \end{bmatrix} \qquad \mathbf{Ux} = \mathbf{y} \quad \Longrightarrow \quad \mathbf{x} = \begin{bmatrix} 2 \\ 4 \\ -4 \end{bmatrix} \tag{20.2.10}$$

5. LU factorization coded in `numpy`

$$\tilde{\mathbf{A}} = \left[\begin{array}{ccc|c} 3 & 9 & 6 & 4.6 \\ 18 & 48 & 39 & 27.2 \\ 9 & -27 & 42 & 9 \end{array}\right] \qquad \mathbf{LU} = \begin{bmatrix} 1 & 0 & 0 \\ 6 & 1 & 0 \\ 3 & 9 & 1 \end{bmatrix}\begin{bmatrix} 3 & 9 & 6 \\ 0 & -6 & 3 \\ 0 & 0 & -3 \end{bmatrix} \tag{20.2.11}$$

$$\mathbf{Ly} = \mathbf{b} \quad \Longrightarrow \quad \mathbf{y} = \begin{bmatrix} 4.6 \\ -0.4 \\ -1.2 \end{bmatrix} \qquad \mathbf{Ux} = \mathbf{y} \quad \Longrightarrow \quad \mathbf{x} = \begin{bmatrix} -\frac{1}{15} \\ \frac{4}{15} \\ \frac{2}{5} \end{bmatrix} \tag{20.2.12}$$

6. Crout's factorization

(a) Perform Doolittle factorization to obtain

$$\mathbf{A} = \mathbf{L}_d\,\mathbf{U}_d \qquad \mathbf{U}_d = \mathbf{DU} \tag{20.2.13}$$

$$\mathbf{L} = \mathbf{L}_d\,\mathbf{D} \qquad \mathbf{A} = \mathbf{LU} \tag{20.2.14}$$

Here, D is a diagonal matrix of $\mathbf{U}_d$

(b) Solving Problem 5 by Crout's factorization

$$\tilde{\mathbf{A}} = \left[\begin{array}{ccc|c} 3 & 9 & 6 & 4.6 \\ 18 & 48 & 39 & 27.2 \\ 9 & -27 & 42 & 9 \end{array}\right] \qquad \mathbf{LU} = \begin{bmatrix} 3 & 0 & 0 \\ 18 & -6 & 0 \\ 9 & -54 & -3 \end{bmatrix} \begin{bmatrix} 1 & 3 & 2 \\ 0 & 1 & -0.5 \\ 0 & 0 & 1 \end{bmatrix} \tag{20.2.15}$$

$$\mathbf{Ly} = \mathbf{b} \quad \Longrightarrow \quad \mathbf{y} = \begin{bmatrix} 4.6 \\ -0.4 \\ -1.2 \end{bmatrix} \qquad \mathbf{Ux} = \mathbf{y} \quad \Longrightarrow \quad \mathbf{x} = \begin{bmatrix} -\frac{1}{15} \\ \frac{4}{15} \\ \frac{2}{5} \end{bmatrix} \tag{20.2.16}$$

(c) Factorizing Problem 5 by all three methods

$$\mathbf{A} = \begin{bmatrix} 1 & -4 & 2 \\ -4 & 25 & 4 \\ 2 & 4 & 24 \end{bmatrix} \tag{20.2.17}$$

$$\mathbf{L}_d\ \mathbf{U}_d = \begin{bmatrix} 1 & 0 & 0 \\ -4 & 1 & 0 \\ 2 & 4/3 & 1 \end{bmatrix} \begin{bmatrix} 1 & -4 & 2 \\ 0 & 9 & 12 \\ 0 & 0 & 4 \end{bmatrix} \tag{20.2.18}$$

$$\mathbf{L}_c\ \mathbf{U}_c = \begin{bmatrix} 1 & 0 & 0 \\ -4 & 9 & 0 \\ 2 & 12 & 4 \end{bmatrix} \begin{bmatrix} 1 & -4 & 2 \\ 0 & 1 & 4/3 \\ 0 & 0 & 1 \end{bmatrix} \tag{20.2.19}$$

$$\mathbf{L}_y\ \mathbf{U}_y = \begin{bmatrix} 1 & 0 & 0 \\ -4 & 3 & 0 \\ 2 & 4 & 2 \end{bmatrix} \begin{bmatrix} 1 & -4 & 2 \\ 0 & 3 & 4 \\ 0 & 0 & 2 \end{bmatrix} \tag{20.2.20}$$

(d) Perform Doolittle factorization and then divide each row of $\mathbf{U}_d$ by the square root of the diagonal term. Then, its transpose is $\mathbf{L}$

(e) When the original matrix A is symmetric,

$$\mathbf{A} = \mathbf{L}_d\ \mathbf{U}_d \qquad \mathbf{A}^T = \mathbf{U}_d^T\ \mathbf{L}_d^T \tag{20.2.21}$$

$$\mathbf{U}_d^T = \mathbf{L}_c \qquad \mathbf{L}_d^T = \mathbf{U}_c \tag{20.2.22}$$

$$\mathbf{L}_c\ \mathbf{U}_c = \mathbf{A}^T = \mathbf{A} \tag{20.2.23}$$

Thus, the Doolittle and Crout's factorizationa re inter-convertible directly, when $\mathbf{A}$ is symmetric.

7. LU factorization coded in `numpy`

$$\tilde{\mathbf{A}} = \left[\begin{array}{ccc|c} 9 & 6 & 12 & 17.4 \\ 6 & 13 & 11 & 23.6 \\ 12 & 11 & 26 & 30.8 \end{array}\right] \qquad \mathbf{LU} = \begin{bmatrix} 3 & 0 & 0 \\ 2 & 3 & 0 \\ 4 & 1 & 3 \end{bmatrix} \begin{bmatrix} 3 & 2 & 4 \\ 0 & 3 & 1 \\ 0 & 0 & 3 \end{bmatrix} \tag{20.2.24}$$

$$\mathbf{Ly} = \mathbf{b} \implies \mathbf{y} = \begin{bmatrix} 17.4 \\ 12 \\ 3.6 \end{bmatrix} \qquad \mathbf{Ux} = \mathbf{y} \implies \mathbf{x} = \begin{bmatrix} 0.6 \\ 1.2 \\ 0.4 \end{bmatrix} \tag{20.2.25}$$

8. LU factorization coded in `numpy`

$$\tilde{\mathbf{A}} = \left[\begin{array}{ccc|c} 4 & 6 & 8 & 0 \\ 6 & 34 & 52 & -160 \\ 8 & 52 & 129 & -452 \end{array}\right] \qquad \mathbf{LU} = \begin{bmatrix} 2 & 0 & 0 \\ 3 & 5 & 0 \\ 4 & 8 & 7 \end{bmatrix} \begin{bmatrix} 2 & 3 & 4 \\ 0 & 5 & 8 \\ 0 & 0 & 7 \end{bmatrix} \tag{20.2.26}$$

$$\mathbf{Ly} = \mathbf{b} \implies \mathbf{y} = \begin{bmatrix} 0 \\ -160 \\ -196 \end{bmatrix} \qquad \mathbf{Ux} = \mathbf{y} \implies \mathbf{x} = \begin{bmatrix} 8 \\ 0 \\ -4 \end{bmatrix} \tag{20.2.27}$$

9. LU factorization coded in `numpy`

$$\tilde{\mathbf{A}} = \left[\begin{array}{ccc|c} 1 & 0 & 3 & 14 \\ 0 & 2 & 1 & 2 \\ 3 & 8 & 14 & 54 \end{array}\right] \qquad \mathbf{LU} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \sqrt{2} & 0 \\ 3 & 1/\sqrt{2} & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 3 \\ 0 & \sqrt{2} & 1/\sqrt{2} \\ 0 & 0 & 1 \end{bmatrix} \tag{20.2.28}$$

$$\mathbf{Ly} = \mathbf{b} \implies \mathbf{y} = \begin{bmatrix} 14 \\ 2 \\ 4 \end{bmatrix} \qquad \mathbf{Ux} = \mathbf{y} \implies \mathbf{x} = \begin{bmatrix} 2 \\ -1 \\ 4 \end{bmatrix} \tag{20.2.29}$$

10. LU factorization coded in `numpy`

$$\tilde{\mathbf{A}} = \left[\begin{array}{ccc|c} 4 & 0 & 2 & 1.5 \\ 0 & 4 & 1 & 4 \\ 2 & 1 & 2 & 2.5 \end{array}\right] \qquad \mathbf{LU} = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 1 & 0.5 & \sqrt{3/4} \end{bmatrix} \begin{bmatrix} 2 & 0 & 1 \\ 0 & 2 & 0.5 \\ 0 & 0 & \sqrt{3/4} \end{bmatrix} \tag{20.2.30}$$

$$\mathbf{Ly} = \mathbf{b} \implies \mathbf{y} = \begin{bmatrix} 1.5 \\ 4 \\ 3/4 \end{bmatrix} \qquad \mathbf{Ux} = \mathbf{y} \implies \mathbf{x} = \begin{bmatrix} -1/8 \\ 3/4 \\ 1 \end{bmatrix} \tag{20.2.31}$$

11. LU factorization coded in `numpy`

$$\tilde{\mathbf{A}} = \left[\begin{array}{cccc|c} 1 & -1 & 3 & 2 & 15 \\ -1 & 5 & -5 & -2 & -35 \\ 3 & -5 & 19 & 3 & 94 \\ 2 & -2 & 3 & 21 & 1 \end{array}\right] \qquad \mathbf{LU} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ -1 & 2 & 0 & 0 \\ 3 & -1 & 3 & 0 \\ 2 & 0 & -1 & 4 \end{bmatrix} \begin{bmatrix} 1 & -1 & 3 & 2 \\ 0 & 2 & -1 & 0 \\ 0 & 0 & 3 & -1 \\ 0 & 0 & 0 & 4 \end{bmatrix} \tag{20.2.32}$$

$$\mathbf{Ly} = \mathbf{b} \quad \Longrightarrow \quad \mathbf{y} = \begin{bmatrix} 15 \\ -20 \\ 39 \\ -16 \end{bmatrix} \qquad \mathbf{Ux} = \mathbf{y} \quad \Longrightarrow \quad \mathbf{x} = \begin{bmatrix} 2 \\ -3 \\ 4 \\ -1 \end{bmatrix} \tag{20.2.33}$$

12. LU factorization coded in `numpy`

$$\tilde{\mathbf{A}} = \left[\begin{array}{cccc|c} 4 & 2 & 4 & 0 & 20 \\ 2 & 2 & 3 & 2 & 36 \\ 4 & 3 & 6 & 3 & 60 \\ 0 & 2 & 3 & 9 & 122 \end{array}\right] \qquad \mathbf{LU} = \begin{bmatrix} 2 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 2 & 1 & 1 & 0 \\ 0 & 2 & 1 & 2 \end{bmatrix} \begin{bmatrix} 2 & 1 & 2 & 0 \\ 0 & 1 & 1 & 2 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 2 \end{bmatrix} \tag{20.2.34}$$

$$\mathbf{Ly} = \mathbf{b} \quad \Longrightarrow \quad \mathbf{y} = \begin{bmatrix} 20 \\ 26 \\ 14 \\ 56 \end{bmatrix} \qquad \mathbf{Ux} = \mathbf{y} \quad \Longrightarrow \quad \mathbf{x} = \begin{bmatrix} 6 \\ -2 \\ 0 \\ 14 \end{bmatrix} \tag{20.2.35}$$

13. Given $\mathbf{A}$ and $\mathbf{B}$ are positive definite,

$$\mathbf{x}^T(-\mathbf{A})\ \mathbf{x} < 0 \tag{20.2.36}$$

$$\mathbf{x}^T\mathbf{A}^T\mathbf{x} = (\mathbf{x}^T\mathbf{A}\mathbf{x})^T > 0 \tag{20.2.37}$$

$$\mathbf{x}^T(\mathbf{A}+\mathbf{B})\mathbf{x} = \mathbf{x}^T\mathbf{A}\mathbf{x} + \mathbf{x}^T\mathbf{B}\mathbf{x} > 0 \tag{20.2.38}$$

No such claim can be made for $\mathbf{A} - \mathbf{B}$ since the end result of subtracting one positive number from another need not be positive.

14. Cholesky method coded in `numpy`

(a) Using Cholesky method on Example 2,

$$\mathbf{A} = \left[\begin{array}{ccc|c} 4 & 2 & 14 & 14 \\ 2 & 17 & -5 & -101 \\ 14 & -5 & 83 & 155 \end{array}\right] \qquad \mathbf{LU} = \begin{bmatrix} 2 & 0 & 0 \\ 1 & 4 & 0 \\ 7 & -3 & 5 \end{bmatrix} \begin{bmatrix} 2 & 1 & 7 \\ 0 & 4 & -3 \\ 0 & 0 & 5 \end{bmatrix} \tag{20.2.39}$$

$$\mathbf{Ly} = \mathbf{b} \quad \Longrightarrow \quad \mathbf{y} = \begin{bmatrix} 14 \\ -108 \\ 25 \end{bmatrix} \qquad \mathbf{Ux} = \mathbf{y} \quad \Longrightarrow \quad \mathbf{x} = \begin{bmatrix} 3 \\ -6 \\ 1 \end{bmatrix} \tag{20.2.40}$$

The code was already used in solving Problems 7, 8, 9

(b) Applying Cholesky factorization to the spline matrices

$$\mathbf{A} = \begin{bmatrix} 2 & 1 & 0 \\ 1 & 4 & 1 \\ 0 & 1 & 2 \end{bmatrix} \tag{20.2.41}$$

$$\mathbf{LU} = \begin{bmatrix} \sqrt{2} & 0 & 0 \\ 1/\sqrt{2} & \sqrt{7/2} & 0 \\ 0 & \sqrt{2/7} & \sqrt{12/7} \end{bmatrix} \begin{bmatrix} \sqrt{2} & 1/\sqrt{2} & 0 \\ 0 & \sqrt{7/2} & \sqrt{2/7} \\ 0 & 0 & \sqrt{12/7} \end{bmatrix} \tag{20.2.42}$$

For the 4×4 matrix,

$$\mathbf{A} = \begin{bmatrix} 2 & 1 & 0 & 0 \\ 1 & 4 & 1 & 0 \\ 0 & 1 & 4 & 1 \\ 0 & 0 & 1 & 2 \end{bmatrix} \tag{20.2.43}$$

$$\mathbf{LU} = \begin{bmatrix} \sqrt{2} & 0 & 0 & 0 \\ 1/\sqrt{2} & \sqrt{7/2} & 0 & 0 \\ 0 & \sqrt{2/7} & \sqrt{26/7} & 0 \\ 0 & 0 & \sqrt{7/26} & \sqrt{45/26} \end{bmatrix} \begin{bmatrix} \sqrt{2} & 1/\sqrt{2} & 0 & 0 \\ 0 & \sqrt{7/2} & \sqrt{2/7} & 0 \\ 0 & 0 & \sqrt{26/7} & \sqrt{7/26} \\ 0 & 0 & 0 & \sqrt{45/26} \end{bmatrix} \tag{20.2.44}$$

15. Finding inverse using Gauss-Jordan elimination

$$\left[\begin{array}{cc|cc} 4 & 5 & 1 & 0 \\ 12 & 14 & 0 & 1 \end{array}\right] \qquad \left[\begin{array}{cc|cc} 4 & 5 & 1 & 0 \\ 0 & -1 & -3 & 1 \end{array}\right] \tag{20.2.45}$$

$$\left[\begin{array}{cc|cc} 4 & 0 & -14 & 5 \\ 0 & -1 & -3 & 1 \end{array}\right] \qquad \left[\begin{array}{cc|cc} 1 & 0 & -7/2 & 5/4 \\ 0 & 1 & 3 & -1 \end{array}\right] \tag{20.2.46}$$

16. Finding inverse using Gauss-Jordan elimination

$$\left[\begin{array}{ccc|ccc} 2 & 1 & 2 & 1 & 0 & 0 \\ -2 & 2 & 1 & 0 & 1 & 0 \\ 1 & 2 & -2 & 0 & 0 & 1 \end{array}\right] \qquad \left[\begin{array}{ccc|ccc} 2 & 1 & 2 & 1 & 0 & 0 \\ 0 & 3 & 3 & 1 & 1 & 0 \\ 0 & 1.5 & -3 & -0.5 & 0 & 1 \end{array}\right] \tag{20.2.47}$$

$$\left[\begin{array}{ccc|ccc} 2 & 1 & 2 & 1 & 0 & 0 \\ 0 & 3 & 3 & 1 & 1 & 0 \\ 0 & 0 & -4.5 & -1 & -0.5 & 1 \end{array}\right] \qquad \left[\begin{array}{ccc|ccc} 2 & 0 & 1 & 2/3 & -1/3 & 0 \\ 0 & 3 & 3 & 1 & 1 & 0 \\ 0 & 0 & -4.5 & -1 & -0.5 & 1 \end{array}\right] \tag{20.2.48}$$

$$\left[\begin{array}{ccc|ccc} 2 & 0 & 0 & 4/9 & -4/9 & 2/9 \\ 0 & 3 & 0 & 1/3 & 2/3 & 2/3 \\ 0 & 0 & -4.5 & -1 & -0.5 & 1 \end{array}\right] \qquad \left[\begin{array}{ccc|ccc} 1 & 0 & 0 & 2/9 & -2/9 & 1/9 \\ 0 & 1 & 0 & 1/9 & 2/9 & 2/9 \\ 0 & 0 & 1 & 2/9 & 1/9 & -2/9 \end{array}\right] \tag{20.2.49}$$

17. Finding inverse using Gauss-Jordan elimination

$$\left[\begin{array}{ccc|ccc} 1 & -4 & 2 & 1 & 0 & 0 \\ -4 & 25 & 4 & 0 & 1 & 0 \\ 2 & 4 & 24 & 0 & 0 & 1 \end{array}\right] \qquad \left[\begin{array}{ccc|ccc} 1 & -4 & 2 & 1 & 0 & 0 \\ 0 & 9 & 12 & 4 & 1 & 0 \\ 0 & 12 & 20 & -2 & 0 & 1 \end{array}\right] \tag{20.2.50}$$

$$\left[\begin{array}{ccc|ccc} 1 & -4 & 2 & 1 & 0 & 0 \\ 0 & 9 & 12 & 4 & 1 & 0 \\ 0 & 0 & 4 & -\frac{22}{3} & -\frac{4}{3} & 1 \end{array}\right] \qquad \left[\begin{array}{ccc|ccc} 1 & 0 & \frac{22}{3} & \frac{25}{9} & \frac{4}{9} & 0 \\ 0 & 9 & 12 & 4 & 1 & 0 \\ 0 & 0 & 4 & -\frac{22}{3} & -\frac{4}{3} & 1 \end{array}\right] \tag{20.2.51}$$

$$\left[\begin{array}{ccc|ccc} 1 & 0 & 0 & \frac{146}{9} & \frac{26}{9} & -\frac{11}{6} \\ 0 & 9 & 0 & 26 & 17 & -3 \\ 0 & 0 & 4 & -\frac{22}{3} & -\frac{4}{3} & 1 \end{array}\right] \qquad \left[\begin{array}{ccc|ccc} 1 & 0 & 0 & \frac{146}{9} & \frac{26}{9} & -\frac{11}{6} \\ 0 & 1 & 0 & \frac{26}{9} & \frac{17}{9} & -\frac{1}{3} \\ 0 & 0 & 1 & -\frac{11}{6} & -\frac{1}{3} & \frac{1}{4} \end{array}\right] \tag{20.2.52}$$

18. Finding inverse using Gauss-Jordan elimination

$$\left[\begin{array}{ccc|ccc} 0.01 & 0 & 0.03 & 1 & 0 & 0 \\ 0 & 0.16 & 0.08 & 0 & 1 & 0 \\ 0.03 & 0.08 & 0.14 & 0 & 0 & 1 \end{array}\right] \quad \left[\begin{array}{ccc|ccc} 0.01 & 0 & 0.03 & 1 & 0 & 0 \\ 0 & 0.16 & 0.08 & 0 & 1 & 0 \\ 0 & 0.08 & 0.05 & -3 & 0 & 1 \end{array}\right] \tag{20.2.53}$$

$$\left[\begin{array}{ccc|ccc} 0.01 & 0 & 0.03 & 1 & 0 & 0 \\ 0 & 0.16 & 0.08 & 0 & 1 & 0 \\ 0 & 0 & 0.01 & -3 & -0.5 & 1 \end{array}\right] \quad \left[\begin{array}{ccc|ccc} 0.01 & 0 & 0 & 10 & 1.5 & -3 \\ 0 & 0.16 & 0.08 & 0 & 1 & 0 \\ 0 & 0 & 0.01 & -3 & -0.5 & 1 \end{array}\right] \tag{20.2.54}$$

$$\left[\begin{array}{ccc|ccc} 0.01 & 0 & 0 & 10 & 1.5 & -3 \\ 0 & 0.16 & 0 & 24 & 5 & -8 \\ 0 & 0 & 0.01 & -3 & -0.5 & 1 \end{array}\right] \quad \left[\begin{array}{ccc|ccc} 1 & 0 & 0 & 1000 & 150 & -300 \\ 0 & 1 & 0 & 150 & 31.25 & -50 \\ 0 & 0 & 1 & -300 & -50 & 100 \end{array}\right] \tag{20.2.55}$$

19. Finding inverse using Gauss-Jordan elimination

$$\left[\begin{array}{cccc|cccc} 4 & 2 & 4 & 0 & 1 & 0 & 0 & 0 \\ 2 & 2 & 3 & 2 & 0 & 1 & 0 & 0 \\ 4 & 3 & 6 & 3 & 0 & 0 & 1 & 0 \\ 0 & 2 & 3 & 9 & 0 & 0 & 0 & 1 \end{array}\right] \quad \left[\begin{array}{cccc|cccc} 4 & 2 & 4 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 2 & -0.5 & 1 & 0 & 0 \\ 0 & 1 & 2 & 3 & -1 & 0 & 1 & 0 \\ 0 & 2 & 3 & 9 & 0 & 0 & 0 & 1 \end{array}\right] \tag{20.2.56}$$

$$\left[\begin{array}{cccc|cccc} 4 & 2 & 4 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 2 & -0.5 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & -0.5 & -1 & 1 & 0 \\ 0 & 0 & 1 & 5 & 1 & -2 & 0 & 1 \end{array}\right] \quad \left[\begin{array}{cccc|cccc} 4 & 2 & 4 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 2 & -0.5 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & -0.5 & -1 & 1 & 0 \\ 0 & 0 & 0 & 4 & 1.5 & -1 & -1 & 1 \end{array}\right] \tag{20.2.57}$$

$$\left[\begin{array}{cccc|cccc} 4 & 0 & 2 & -4 & 2 & -2 & 0 & 0 \\ 0 & 1 & 1 & 2 & -0.5 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & -0.5 & -1 & 1 & 0 \\ 0 & 0 & 0 & 4 & 1.5 & -1 & -1 & 1 \end{array}\right] \quad \left[\begin{array}{cccc|cccc} 4 & 0 & 0 & -6 & 3 & 0 & -2 & 0 \\ 0 & 1 & 0 & 1 & 0 & 2 & -1 & 0 \\ 0 & 0 & 1 & 1 & -0.5 & -1 & 1 & 0 \\ 0 & 0 & 0 & 4 & 1.5 & -1 & -1 & 1 \end{array}\right] \tag{20.2.58}$$

$$\left[\begin{array}{cccc|cccc} 4 & 0 & 0 & 0 & \frac{21}{4} & -\frac{3}{2} & -\frac{7}{2} & \frac{3}{2} \\ 0 & 1 & 0 & 0 & -\frac{3}{8} & \frac{9}{4} & -\frac{3}{4} & -\frac{1}{4} \\ 0 & 0 & 1 & 0 & -\frac{7}{8} & -\frac{3}{4} & \frac{5}{4} & -\frac{1}{4} \\ 0 & 0 & 0 & 4 & 1.5 & -1 & -1 & 1 \end{array}\right] \quad \left[\begin{array}{cccc|cccc} 1 & 0 & 0 & 0 & \frac{21}{16} & -\frac{3}{8} & \frac{-7}{8} & \frac{3}{8} \\ 0 & 1 & 0 & 0 & -\frac{3}{8} & \frac{9}{4} & -\frac{3}{4} & -\frac{1}{4} \\ 0 & 0 & 1 & 0 & -\frac{7}{8} & -\frac{3}{4} & \frac{5}{4} & -\frac{1}{4} \\ 0 & 0 & 0 & 1 & \frac{3}{8} & -\frac{1}{4} & -\frac{1}{4} & \frac{1}{4} \end{array}\right] \tag{20.2.59}$$

20. Finding the determinant

$$\mathbf{A} = \begin{bmatrix} 1/3 & 1/4 & 2 \\ -1/9 & 1 & 1/7 \\ 4/63 & -3/28 & 13/49 \end{bmatrix} \tag{20.2.60}$$

$$\begin{aligned} \det \mathbf{A} &= (1/3)[13/49 + (1/7)(3/28)] \\ &\quad + (1/9)[(1/4)(13/49) + (2)(3/28)] \\ &\quad + (4/63)[(1/4)(1/7) - 2] = 0 \end{aligned} \tag{20.2.61}$$

Using increasingly many significant digits,

$$\det \mathbf{A_2} = 0.089 + 0.031 + 0.063 = 0.18 \tag{20.2.62}$$

$$\det \mathbf{A_3} = 0.0932 + 0.0311 - 0.124 = 0.0003 \tag{20.2.63}$$

$$\det \mathbf{A_4} = 0.09352 + 0.03116 - 0.12469436 = -1.436 \times 10^{-5} \tag{20.2.64}$$

$$\det \mathbf{A_5} = 0.093539 + 0.031178 - 0.12471 = 7 \times 10^{-6} \tag{20.2.65}$$

The determinant gets closer to the real value with more significant digits.

20.3 Linear Systems: Solution by Iteration

1. Gauss-Seidel method coded in `sympy`,

$$\mathbf{x} = \begin{bmatrix} 87.5 \\ 87.5 \\ 62.5 \\ 62.5 \end{bmatrix} \qquad n = 14 \tag{20.3.1}$$

2. Checking the eigenvalues of the iteration matrix,

$$\mathbf{I} - \mathbf{A} = \begin{bmatrix} 0 & -0.5 & -0.5 \\ -0.5 & 0 & -0.5 \\ -0.5 & -0.5 & 0 \end{bmatrix} \qquad 0 = -\lambda^3 + 0.75\lambda - 0.25 \tag{20.3.2}$$

$$\lambda_i = \{0.5, 0.5, -1\} \tag{20.3.3}$$

Not all the eigenvalues have absolute value less than one, which means the Jacobi iteration diverges.

3. The row and column norms are,

$$\mathbf{C} = \begin{bmatrix} 0 & -0.5 & -0.5 \\ 0 & 0.25 & -0.25 \\ 0 & 1/8 & 3/8 \end{bmatrix} \qquad \|\mathbf{C}\|_j = \begin{bmatrix} 1 \\ 1/2 \\ 1/2 \end{bmatrix} \tag{20.3.4}$$

$$\|\mathbf{C}\|_k = \begin{bmatrix} 0 & 7/8 & 9/8 \end{bmatrix} \tag{20.3.5}$$

Since some terms in the row and column norms are ≥ 1, no determination can be made.

4. Using 5 full sweeps starting with the given initial guess.

$$\mathbf{x} = \begin{bmatrix} 2.99969 & -9.00015 & 5.99996 \end{bmatrix} \tag{20.3.6}$$

5. Using 5 full sweeps starting with the given initial guess.

$$\mathbf{x} = \begin{bmatrix} 0.5 & 0.5 & 0.5 \end{bmatrix} \tag{20.3.7}$$

6. Using 5 full sweeps starting with the given initial guess, after row swaps,

$$\mathbf{A}^* = \begin{bmatrix} 5 & 1 & 0 \\ 1 & 6 & 1 \\ 0 & 1 & 7 \end{bmatrix} \qquad \mathbf{b}^* = \begin{bmatrix} 0 \\ -10.5 \\ 25.5 \end{bmatrix} \tag{20.3.8}$$

$$\mathbf{x}^* = \begin{bmatrix} 0.499977 \\ -2.5 \\ 4 \end{bmatrix} \qquad \mathbf{x} = \begin{bmatrix} 4 \\ 0.499977 \\ -2.5 \end{bmatrix} \tag{20.3.9}$$

7. Using 5 full sweeps starting with the given initial guess.

$$\mathbf{x} = \begin{bmatrix} 1.99951 & -4.00012 & 7.99998 \end{bmatrix} \tag{20.3.10}$$

8. Using 5 full sweeps starting with the given initial guess.

$$\mathbf{x} = \begin{bmatrix} 2.05332 & 0.01272 & 0.960209 \end{bmatrix} \tag{20.3.11}$$

9. Using 5 full sweeps starting with the given initial guess.

$$\mathbf{x} = \begin{bmatrix} 2.00004 & 0.998059 & 4.00072 \end{bmatrix} \tag{20.3.12}$$

10. Using 5 full sweeps starting with the given initial guess, after row swaps,

$$\mathbf{A}^* = \begin{bmatrix} 8 & 2 & 1 \\ 1 & 6 & 2 \\ 4 & 0 & 5 \end{bmatrix} \qquad \mathbf{b}^* = \begin{bmatrix} -11.5 \\ 18.5 \\ 12.5 \end{bmatrix} \tag{20.3.13}$$

$$\mathbf{x}^* = \begin{bmatrix} -2.49475 \\ 1.99809 \\ 4.4958 \end{bmatrix} \qquad \mathbf{x} = \begin{bmatrix} 4.4958 \\ 1.99809 \\ -2.49475 \end{bmatrix} \tag{20.3.14}$$

11. Using 3 full sweeps starting with the given initial guess.

$$\mathbf{x}_a = \begin{bmatrix} 0.499825 \\ 0.500008 \\ 0.500017 \end{bmatrix} \qquad \mathbf{x}_b = \begin{bmatrix} 0.5033326 \\ 0.499845 \\ 0.499682 \end{bmatrix} \tag{20.3.15}$$

The initial guess being closer makes the result at the end of 3 iterations closer to the correct answer.

12. Computing the iteration matrix $\mathbf{C}$,

$$\mathbf{A}_1 = \begin{bmatrix} 10 & 1 & 1 \\ 1 & 10 & 1 \\ 1 & 1 & 10 \end{bmatrix} \qquad \mathbf{C}_1 = -(\mathbf{I} + \mathbf{L})^{-1}\,\mathbf{U} \tag{20.3.16}$$

$$\mathbf{C}_1 = \begin{bmatrix} 0 & -0.5 & -0.5 \\ 0 & 0.25 & -0.25 \\ 0 & 1/8 & 3/8 \end{bmatrix} \qquad \|\mathbf{C}\|_1 = 0.169 < 1 \tag{20.3.17}$$

Rearranging the rows,

$$\mathbf{A}_2 = \begin{bmatrix} 1 & 1 & 10 \\ 10 & 1 & 1 \\ 1 & 10 & 1 \end{bmatrix} \qquad \mathbf{C}_2 = -(\mathbf{I} + \mathbf{L})^{-1}\,\mathbf{U} \tag{20.3.18}$$

$$\mathbf{C}_2 = \begin{bmatrix} 0 & -1 & -10 \\ 0 & 10 & 99 \\ 0 & -99 & -980 \end{bmatrix} \qquad \|\mathbf{C}\|_2 = 990.052 > 1 \tag{20.3.19}$$

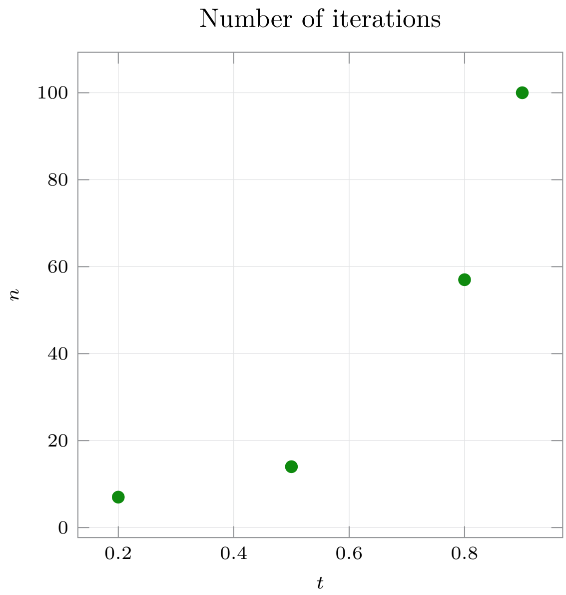
Clearly, the Gauss-Seidel method only converges when the diagonal elements are the largest in their respective rows.

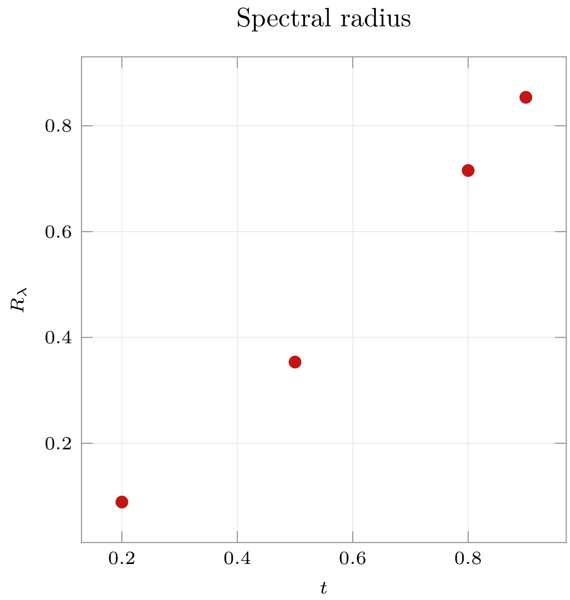
13. Gauss-Seidel method

(a) Code written in `numpy`

(b) Finding the number of steps to obtain a $6S$ solution.

value of t	number of iterations n	Spectral Radius
0.2	7	0.08944
0.5	14	0.3535
0.8	57	0.7155
0.9	100	0.8538





(c) Starting from the iteration formula

$$\mathbf{x}^{(m+1)} = \mathbf{b} - \mathbf{L}\mathbf{x}^{(m+1)} - \mathbf{U}\mathbf{x}^{(m)} \tag{20.3.20}$$

$$= \mathbf{x}^{(m)} + \mathbf{b} - \mathbf{L}\mathbf{x}^{(m+1)} - \mathbf{U}\mathbf{x}^{(m)} - \mathbf{I}\mathbf{x}^{(m)} \tag{20.3.21}$$

For these two examples, the improvement is minimal.

14. Jacobi method coded in `numpy`.

$$n_g = 16 \qquad\qquad n_J = 33 \tag{20.3.22}$$

15. Jacobi method coded in `numpy`.

$$n_g = 8 \qquad\qquad n_J = 12 \tag{20.3.23}$$

16. Jacobi method coded in `numpy`.

$$n_g = 10 \qquad\qquad n_J = 20 \tag{20.3.24}$$

17. The iteration matrix is,

$$\mathbf{C} = \begin{bmatrix} 0 & 1/4 & 1/8 \\ 1/6 & 0 & 1/3 \\ 4/5 & 0 & 0 \end{bmatrix} \qquad 0 = -\lambda^3 + \frac{17}{120}\,\lambda + \frac{1}{15} \tag{20.3.25}$$

$$\{\lambda_i\} = 0.519589,\, -0.259795 \pm \mathfrak{i}\; 0.246603 \tag{20.3.26}$$

18. Computing the three kinds of norms, after coding in `numpy`

$$\mathbf{A} = \begin{bmatrix} 8 & 2 & 1 \\ 1 & 6 & 2 \\ 4 & 0 & 5 \end{bmatrix} \qquad \|\mathbf{A}\|_F = 12.288 \tag{20.3.27}$$

$$\|\mathbf{A}\|_j = 11 \qquad \|\mathbf{A}\|_k = 13 \tag{20.3.28}$$

19. Computing the three kinds of norms, after coding in `numpy`

$$\mathbf{A} = \begin{bmatrix} 10 & 1 & 1 \\ 1 & 10 & 1 \\ 1 & 1 & 10 \end{bmatrix} \qquad \|\mathbf{A}\|_F = 17.493 \tag{20.3.29}$$

$$\|\mathbf{A}\|_j = 12 \qquad \|\mathbf{A}\|_k = 12 \tag{20.3.30}$$

A greater dominance of the diagonal terms means a greater difference between the Frobenius norms and the others.

20. Computing the three kinds of norms, after coding in `numpy`

$$\mathbf{A} = \begin{bmatrix} 2k & -k & -k \\ k & -2k & k \\ -k & -k & 2k \end{bmatrix} \qquad \|\mathbf{A}\|_F = \sqrt{18}k \tag{20.3.31}$$

$$\|\mathbf{A}\|_j = 4k \qquad \|\mathbf{A}\|_k = 4k \tag{20.3.32}$$

The differences scale linearly with k.

20.4 Linear Systems: Ill-Conditioning, Norms

1. Finding the three types of norm

$$\mathbf{x} = \begin{bmatrix} 1 & -3 & 8 & 0 & -6 & 0 \end{bmatrix} \qquad \|\mathbf{x}\|_1 = 18 \tag{20.4.1}$$

$$\|\mathbf{x}\|_2 = \sqrt{1+9+64+36} = \sqrt{110} \qquad \|\mathbf{x}\|_\infty = 8 \tag{20.4.2}$$

$$\mathbf{u} = \frac{1}{8}\begin{bmatrix} 1 & -3 & 8 & 0 & -6 & 0 \end{bmatrix} \tag{20.4.3}$$

2. Finding the three types of norm

$$\mathbf{x} = \begin{bmatrix} 4 & -1 & 8 \end{bmatrix} \qquad \|\mathbf{x}\|_1 = 13 \tag{20.4.4}$$

$$\|\mathbf{x}\|_2 = \sqrt{4+1+64} = \sqrt{69} \qquad \|\mathbf{x}\|_\infty = 8 \tag{20.4.5}$$

$$\mathbf{u} = \frac{1}{8}\begin{bmatrix} 4 & -1 & 8 \end{bmatrix} \tag{20.4.6}$$

3. Finding the three types of norm

$$\mathbf{x} = \begin{bmatrix} 0.2 & 0.6 & -2.1 & 3.0 \end{bmatrix} \qquad \|\mathbf{x}\|_1 = 5.9 \tag{20.4.7}$$

$$\|\mathbf{x}\|_2 = \sqrt{0.04+0.36+4.41+9} = \sqrt{13.81} \qquad \|\mathbf{x}\|_\infty = 3 \tag{20.4.8}$$

$$\mathbf{u} = \frac{1}{3}\begin{bmatrix} 0.2 & 0.6 & -2.1 & 3.0 \end{bmatrix} \tag{20.4.9}$$

4. Finding the three types of norm, given $k > 4$

$$\mathbf{x} = \begin{bmatrix} k^2 & 4k & k^3 \end{bmatrix} \qquad \|\mathbf{x}\|_1 = k^3 + k^2 + 4k \tag{20.4.10}$$

$$\|\mathbf{x}\|_2 = \sqrt{k^3 + k^2 + 4k} \qquad \|\mathbf{x}\|_\infty = k^3 \tag{20.4.11}$$

$$\mathbf{u} = \begin{bmatrix} \frac{1}{k} & \frac{4}{k^2} & 1 \end{bmatrix} \tag{20.4.12}$$

5. Finding the three types of norm, given $k > 4$

$$\mathbf{x} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \end{bmatrix} \qquad \|\mathbf{x}\|_1 = 5 \tag{20.4.13}$$

$$\|\mathbf{x}\|_2 = \sqrt{1+1+1+1+1} = \sqrt{5} \qquad \|\mathbf{x}\|_\infty = 1 \tag{20.4.14}$$

$$\mathbf{u} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \end{bmatrix} \tag{20.4.15}$$

6. Finding the three types of norm, given $k > 4$

$$\mathbf{x} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 \end{bmatrix} \qquad \|\mathbf{x}\|_1 = 1 \tag{20.4.16}$$

$$\|\mathbf{x}\|_2 = \sqrt{0+0+0+1+0} = 1 \qquad \|\mathbf{x}\|_\infty = 1 \tag{20.4.17}$$

$$\mathbf{u} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 \end{bmatrix} \tag{20.4.18}$$

7. Finding the three types of norm, given $k > 4$

$$\mathbf{x} = \begin{bmatrix} a & b & c \end{bmatrix} \qquad \|\mathbf{x}\|_1 = |a| + |b| + |c| \tag{20.4.19}$$

$$\|\mathbf{x}\|_2 = \sqrt{a^2 + b^2 + c^2} \qquad \|\mathbf{x}\|_1 = \|\mathbf{x}\|_2 \tag{20.4.20}$$

$$a^2 + b^2 + c^2 = (|a| + |b| + |c|)^2 \qquad ab + bc + ca = 0 \tag{20.4.21}$$

Two out of three elements must be zero.

8. Let the term with the greatest absolute value be x_j

$$\|\mathbf{x}\|_\infty = x_j \qquad \|\mathbf{x}\|_2 = \sqrt{\sum_{i=1}^{n} x_i^2} \tag{20.4.22}$$

$$x_1^2 + x_2^2 + \cdots + x_n^2 \geq x_j^2 \qquad \|\mathbf{x}\|_2 \geq \|\mathbf{x}\|_\infty \tag{20.4.23}$$

Now comparing the l_1 and Euclidean norms, the triangle inequality yields,

$$(|x_1| + |x_2| + \cdots + |x_n|)^2 \geq x_1^2 + x_2^2 + \ldots x_n^2 \tag{20.4.24}$$

$$|x_1| + |x_2| + \cdots + |x_n| \geq \sqrt{x_1^2 + x_2^2 + \ldots x_n^2} \tag{20.4.25}$$

$$\|\mathbf{x}\|_1 \geq \|\mathbf{x}\|_2 \tag{20.4.26}$$

9. Finding the inverse and then the condition number,

$$\mathbf{A} = \begin{bmatrix} 2 & 1 \\ 0 & 4 \end{bmatrix} \qquad \mathbf{A}^{-1} = \begin{bmatrix} 1/2 & -1/8 \\ 0 & 1/4 \end{bmatrix} \tag{20.4.27}$$

$$\|\mathbf{A}\|_1 = 5 \qquad \|\mathbf{A}^{-1}\|_1 = 1/2 \tag{20.4.28}$$

$$\kappa(\mathbf{A}) = 2.5 \tag{20.4.29}$$

10. Finding the inverse and then the condition number,

$$\mathbf{A} = \begin{bmatrix} 2.1 & 4.5 \\ 0.5 & 1.8 \end{bmatrix} \qquad \mathbf{A}^{-1} = \begin{bmatrix} 20/17 & -50/17 \\ -50/153 & 70/51 \end{bmatrix} \tag{20.4.30}$$

$$\|\mathbf{A}\|_1 = 6.3 \qquad \|\mathbf{A}^{-1}\|_1 = \frac{220}{51} \tag{20.4.31}$$

$$\kappa(\mathbf{A}) = \frac{462}{17} \tag{20.4.32}$$

11. Finding the inverse and then the condition number,

$$\mathbf{A} = \begin{bmatrix} \sqrt{5} & 5 \\ 0 & -\sqrt{5} \end{bmatrix} \qquad \mathbf{A}^{-1} = \frac{1}{5}\begin{bmatrix} \sqrt{5} & 5 \\ 0 & -\sqrt{5} \end{bmatrix} \tag{20.4.33}$$

$$\|\mathbf{A}\|_1 = 5 + \sqrt{5} \qquad \|\mathbf{A}^{-1}\|_1 = \frac{5+\sqrt{5}}{5} \tag{20.4.34}$$

$$\kappa(\mathbf{A}) = 6 + 2\sqrt{5} \tag{20.4.35}$$

12. Finding the inverse and then the condition number,

$$\mathbf{A} = \begin{bmatrix} 7 & 6 \\ 6 & 5 \end{bmatrix} \qquad \mathbf{A}^{-1} = \begin{bmatrix} -5 & -6 \\ -6 & -7 \end{bmatrix} \tag{20.4.36}$$

$$\|\mathbf{A}\|_1 = 13 \qquad \|\mathbf{A}^{-1}\|_1 = 13 \tag{20.4.37}$$

$$\kappa(\mathbf{A}) = 169 \tag{20.4.38}$$

13. Finding the inverse and then the condition number,

$$\mathbf{A} = \begin{bmatrix} -2 & 4 & -1 \\ -2 & 3 & 0 \\ 7 & -12 & 2 \end{bmatrix} \qquad \mathbf{A}^{-1} = \begin{bmatrix} 6 & 4 & 3 \\ 4 & 3 & 2 \\ 3 & 4 & 2 \end{bmatrix} \tag{20.4.39}$$

$$\|\mathbf{A}\|_1 = 19 \qquad \|\mathbf{A}^{-1}\|_1 = 13 \tag{20.4.40}$$

$$\kappa(\mathbf{A}) = 247 \tag{20.4.41}$$

14. Finding the inverse and then the condition number,

$$\mathbf{A} = \begin{bmatrix} 1 & 1/100 & 0 \\ 1/100 & 1 & 1/100 \\ 0 & 1/100 & 1 \end{bmatrix} \tag{20.4.42}$$

$$\mathbf{A}^{-1} = \begin{bmatrix} 9999/9998 & -50/4999 & 1/9998 \\ -50/4999 & 5000/4999 & -50/4999 \\ 1/9998 & -50/4999 & 9999/9998 \end{bmatrix} \tag{20.4.43}$$

$$\|\mathbf{A}\|_1 = 1.01 \tag{20.4.44}$$

$$\|\mathbf{A}^{-1}\|_1 = \frac{10200}{9998} \tag{20.4.45}$$

$$\kappa(\mathbf{A}) = \frac{5151}{4999} = 1.03 \tag{20.4.46}$$

15. Finding the inverse and then the condition number,

$$\mathbf{A} = \begin{bmatrix} -20 & 0 & 0 \\ 0 & 1/20 & 0 \\ 0 & 0 & 20 \end{bmatrix} \qquad \mathbf{A}^{-1} = \begin{bmatrix} -1/20 & 0 & 0 \\ 0 & 20 & 0 \\ 0 & 0 & 1/20 \end{bmatrix} \tag{20.4.47}$$

$$\|\mathbf{A}\|_1 = 20 \qquad \|\mathbf{A}^{-1}\|_1 = 20 \tag{20.4.48}$$

$$\kappa(\mathbf{A}) = 400 \tag{20.4.49}$$

16. Finding the inverse and then the condition number,

$$\mathbf{A} = \begin{bmatrix} 21 & 10.5 & 7 & 5.25 \\ 10.5 & 7 & 5.25 & 4.2 \\ 7 & 5.25 & 4.2 & 3.5 \\ 5.25 & 4.2 & 3.5 & 3 \end{bmatrix} \qquad \mathbf{A}^{-1} = \begin{bmatrix} 16/21 & -40/7 & 80/7 & -20/3 \\ -40/7 & 400/7 & -900/7 & 80 \\ 80/7 & -900/7 & 2160/7 & -200 \\ -20/3 & 80 & -200 & 400/3 \end{bmatrix} \tag{20.4.50}$$

$$\|\mathbf{A}\|_1 = 43.75 \qquad \|\mathbf{A}^{-1}\|_1 = \frac{4540}{7} \tag{20.4.51}$$

$$\kappa(\mathbf{A}) = 28375 \tag{20.4.52}$$

17. Verifying the relation,

$$\mathbf{A} = \begin{bmatrix} -2 & 4 & -1 \\ -2 & 3 & 0 \\ 7 & -12 & 2 \end{bmatrix} \qquad \mathbf{x} = \begin{bmatrix} 3 \\ 15 \\ -4 \end{bmatrix} \tag{20.4.53}$$

$$\|\mathbf{A}\|_\infty = 21 \qquad \|\mathbf{x}\|_\infty = 15 \tag{20.4.54}$$

$$\mathbf{Ax} = \begin{bmatrix} 58 \\ 39 \\ -167 \end{bmatrix} \qquad \|\mathbf{Ax}\|_\infty = 167 < 315 \tag{20.4.55}$$

18. Verifying the relation,

$$\mathbf{A} = \begin{bmatrix} 2 & 1 \\ 0 & 4 \end{bmatrix} \qquad \mathbf{B} = \begin{bmatrix} 2.1 & 4.5 \\ 0.5 & 1.8 \end{bmatrix} \tag{20.4.56}$$

$$\mathbf{AB} = \begin{bmatrix} 4.7 & 10.8 \\ 2 & 7.2 \end{bmatrix} \qquad \|\mathbf{AB}\|_\infty = 15.5 \leq 26.4 \tag{20.4.57}$$

19. Solving the two systems, which differ in **b**

$$x_1 = \begin{bmatrix} -2 \\ 4 \end{bmatrix} \qquad x_2 = \begin{bmatrix} -144 \\ 184 \end{bmatrix} \tag{20.4.58}$$

$$\kappa(\mathbf{A}) = 21318 \tag{20.4.59}$$

The system is unstable, and the condition number is high.

20. Solving the two systems, which differ in **b**

$$x_1 = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \qquad x_2 = \begin{bmatrix} 0.8454 \\ 1.2727 \end{bmatrix} \tag{20.4.60}$$

$$\kappa(\mathbf{A}) = 143.45 \tag{20.4.61}$$

The system is less unstable, and the condition number is lower.

21. The residual which might ahve been large, like the deviation in x, turns out to be

$$\mathbf{r} = \mathbf{A}\ (\mathbf{x} - \tilde{\mathbf{x}}) = \begin{bmatrix} 4.5 & 3.55 \\ 3.55 & 2.8 \end{bmatrix} \begin{bmatrix} 8 \\ -10.1 \end{bmatrix} \tag{20.4.62}$$

$$= \begin{bmatrix} 0.145 \\ 0.12 \end{bmatrix} \tag{20.4.63}$$

which is very small. This is a symptom of the ill-conditioned matrix $\mathbf{A}$

22. Looking at the condition number using l_1 and l_∞,

$$\|\mathbf{A}\mathbf{A}^{-1}\| \leq \|\mathbf{A}\|\ \|\mathbf{A}^{-1}\| \qquad \|\mathbf{I}\| \leq \|\mathbf{A}\|\ \|\mathbf{A}^{-1}\| \tag{20.4.64}$$

$$1 \leq \kappa(\mathbf{A}) \tag{20.4.65}$$

Now looking at the Frobenius form,

$$\|\mathbf{A}\mathbf{A}^{-1}\| \leq \|\mathbf{A}\|\ \|\mathbf{A}^{-1}\| \qquad \|\mathbf{I}\| \leq \|\mathbf{A}\|\ \|\mathbf{A}^{-1}\| \tag{20.4.66}$$

$$\sqrt{1^2 + 1^2 + \cdots + 1^2} \leq \kappa(\mathbf{A}) \qquad \sqrt{n} \leq \kappa(\mathbf{A}) \tag{20.4.67}$$

23. Computing the condition numbers using the l_1 norm,

n	$\kappa(\mathbf{H})$
3	748
4	943655
5	29070279
6	985194889

A linear fit of n vs $\ln \kappa$ gives,

$$\ln \kappa = 3.5n - 3.76 \qquad \kappa = 0.0233\ e^{3.5n} \tag{20.4.68}$$

24. Norms

(a) Proving the first relation, assuming x_j is the largest element in absolute value

$$|x_j| \leq |x_1| + \cdots + |x_n| \qquad \|\mathbf{x}\|_\infty \leq \|\mathbf{x}\|_1 \tag{20.4.69}$$

$$|x_1| + \cdots + |x_n| \leq |x_j| + \cdots + |x_j| \qquad \|\mathbf{x}\|_1 \leq n\|\mathbf{x}\|_\infty \tag{20.4.70}$$

Proving the second relation, assuming x_j is the largest element in absolute value

$$\|\mathbf{x}\|_1 \leq n\|\mathbf{x}\|_\infty \qquad \frac{1}{n} \|\mathbf{x}\|_1 \leq \|\mathbf{x}\|_\infty \tag{20.4.71}$$

(b) Proving the first relation, with $|x_j|$ being the infinity norm

$$|x_1|^2 + \cdots + |x_n|^2 \leq (|x_1| + \cdots + |x_n|)^2 \qquad \|\mathbf{x}\|_2 \leq \|\mathbf{x}\|_1 \tag{20.4.72}$$

$$\mathbf{y} = \begin{bmatrix} 1 & 1 & \dots & 1 \end{bmatrix} \qquad \left(\mathbf{x}^T\mathbf{y}\right)^2 = (x_1 + \cdots + x_n)^2 \tag{20.4.73}$$

$$\|\mathbf{x}\|_1^2 = n \ \|\mathbf{x}\|_2^2 \qquad \|\mathbf{x}\|_1 = \sqrt{n} \ \|\mathbf{x}\|_2 \tag{20.4.74}$$

The second inequalities follow from the first.

(c) The inequality starts from

$$\frac{\|\mathbf{Ax}\|}{\|\mathbf{x}\|} \leq c \qquad \mathbf{x} \neq 0 \tag{20.4.75}$$

$$\|\mathbf{x}\| = 1 \qquad \implies \qquad \|\mathbf{Ax}\| \leq c \tag{20.4.76}$$

$$\kappa(\mathbf{A}) = \max_{\|\mathbf{x}\|=1} \|\mathbf{Ax}\| \tag{20.4.77}$$

The same c that is valid for the original definition of κ, is also valid for this new relation. This means that the two formulas are equivalent.

(d) Examples TBC. Only considering the column sum norm,

$$\|\mathbf{A}\| = \max_k \sum_{j=1}^n |a_{jk}| \qquad |a_{jk}| \geq 0 \tag{20.4.78}$$

$$\implies \qquad \|\mathbf{A}\| \geq 0 \tag{20.4.79}$$

The aboslute value is zero only if the number itself is zero.

$$\|\mathbf{A}\| = 0 \qquad \iff \qquad \mathbf{A} = 0 \tag{20.4.80}$$

Multiplying a matrix by a scalar multiplies every element by that scalar.

$$\|k \ \mathbf{A}\| = |k| \ \|\mathbf{A}\| \tag{20.4.81}$$

The absolute value operation satisfies the triangle inequality.

$$|a_{jk} + b_{jk}| \leq |a_{jk}| + |b_{jk}| \tag{20.4.82}$$

$$\sum_{j=1}^{n} |a_{jk} + b_{jk}| \leq \sum_{j=1}^{n} |a_{jk}| + \sum_{j=1}^{n} |b_{jk}| \tag{20.4.83}$$

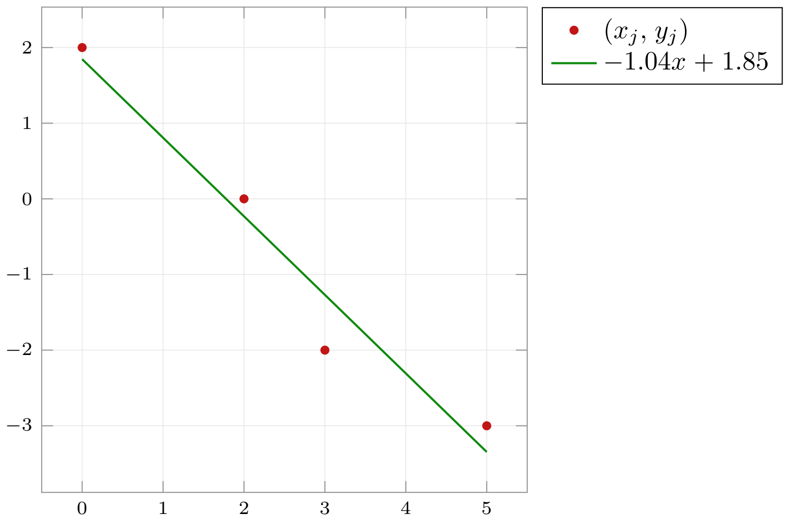
$$\max_k \sum_{j=1}^{n} |a_{jk} + b_{jk}| \leq \max_k \sum_{j=1}^{n} |a_{jk}| + \max_k \sum_{j=1}^{n} |b_{jk}| \tag{20.4.84}$$

$$\|\mathbf{A} + \mathbf{B}\| \leq \|\mathbf{A}\| + \|\mathbf{B}\| \tag{20.4.85}$$

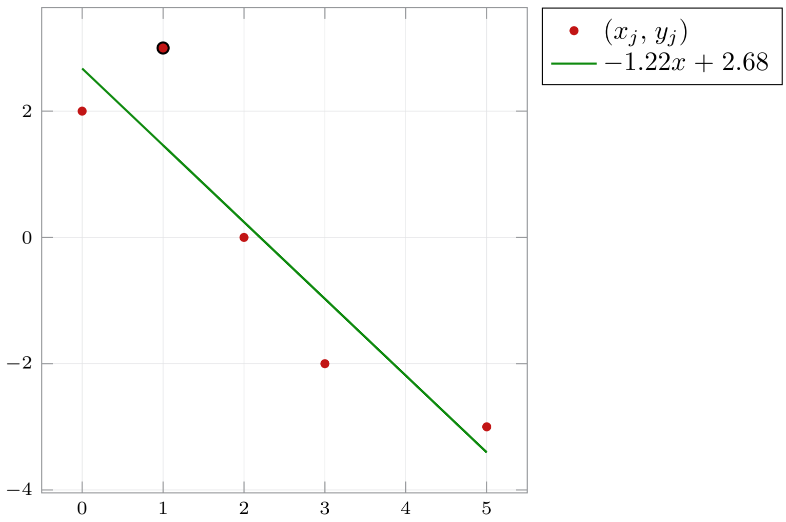
25. Refer notes. TBC

20.5 Least Squares Method

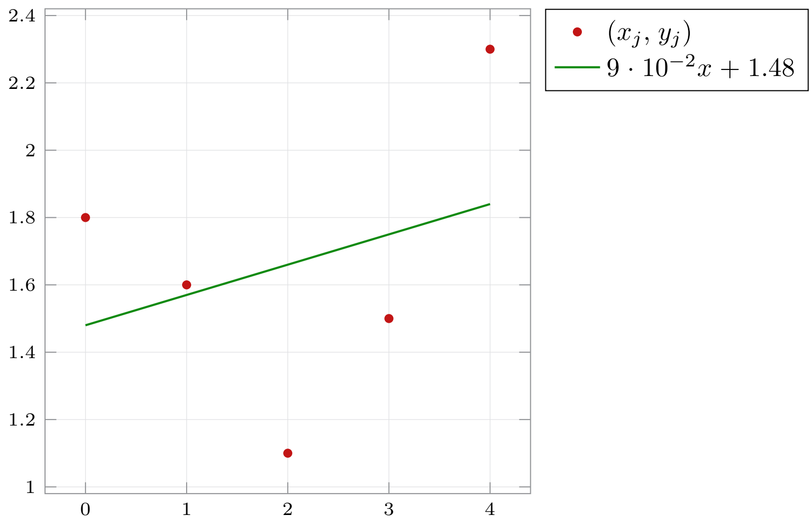
1. Plotting the best fit line,



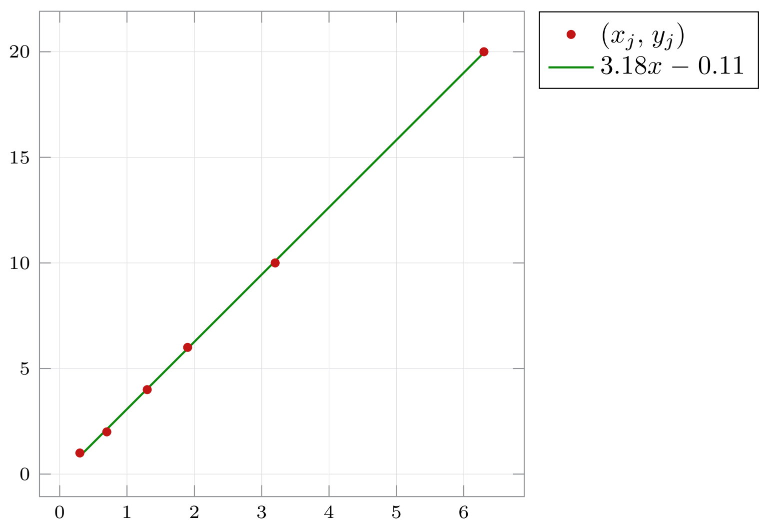
2. Adding a point far above, will decrease the slope and increase the intercept.



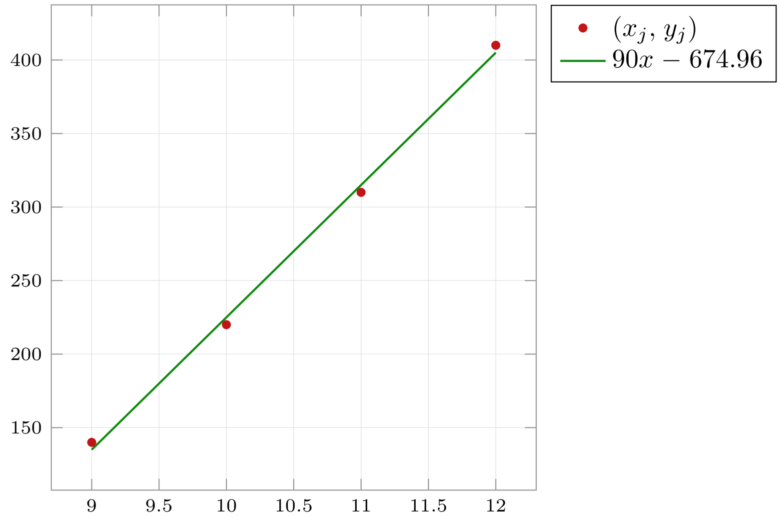
3. Plotting the best fit line, which is a bad fit



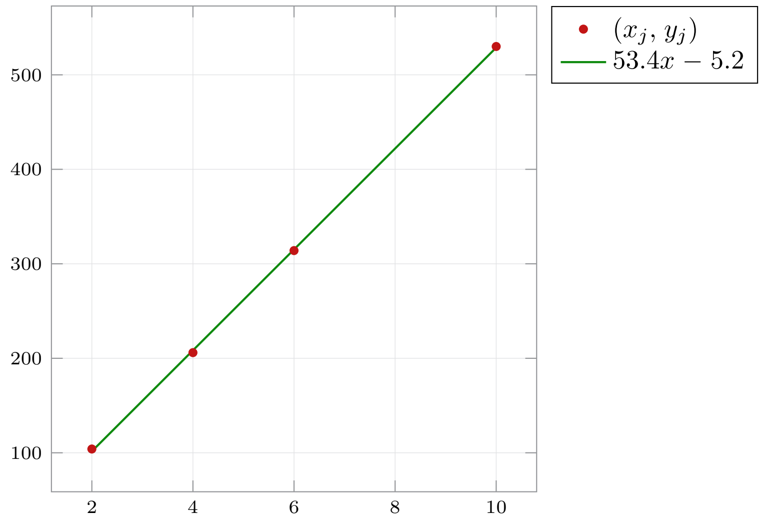
4. Plotting the best fit line, for the equation $F = ks$, gives $(1/k) = 0.31$ and thus $k = 3.226$ lb cm^{-1}



5. Plotting the best fit line, for the equation $x = vt$, gives $k = 90$ km h^{-1}



6. Plotting the best fit line, for the equation $U = Ri$, gives $R = 53.4$



7. The normal equations for a quadratic polynomial fit

$$p(x) = b_0 + b_1 x + b_2 x^2 \qquad q = \sum_{j=1}^{n} \left[y_j - b_0 - b_1 x_j - b_2 x_j^2 \right]^2 \tag{20.5.1}$$

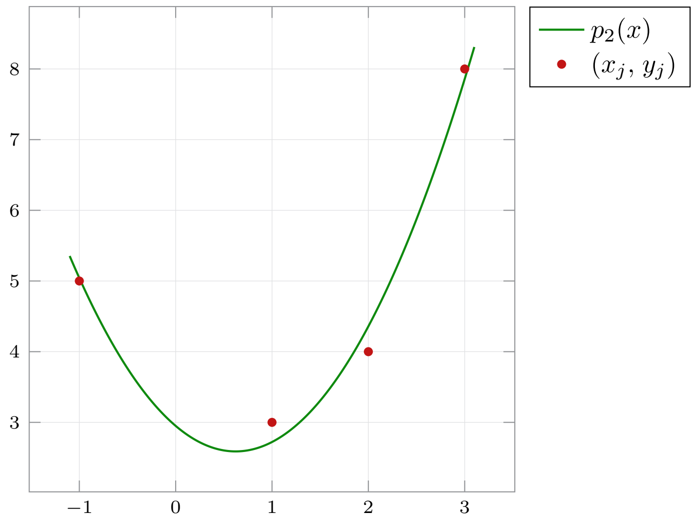
$$\frac{\partial q}{\partial b_0} = 0 \qquad \sum y_j = b_0 \sum 1 + b_1 \sum x_j + b_2 \sum x_j^2 \tag{20.5.2}$$

$$\frac{\partial q}{\partial b_1} = 0 \qquad \sum x_j y_j = b_0 \sum x_j + b_1 \sum x_j^2 + b_2 \sum x_j^3 \tag{20.5.3}$$

$$\frac{\partial q}{\partial b_2} = 0 \qquad \sum x_j^2 y_j = b_0 \sum x_j^2 + b_1 \sum x_j^3 + b_2 \sum x_j^4 \tag{20.5.4}$$

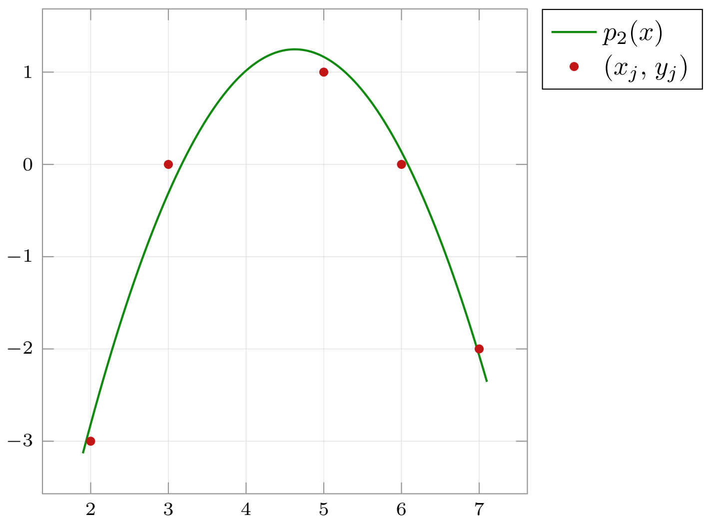
8. Plotting the best fit parabola,

$0.932 - 1.16x + 2.95x^2$



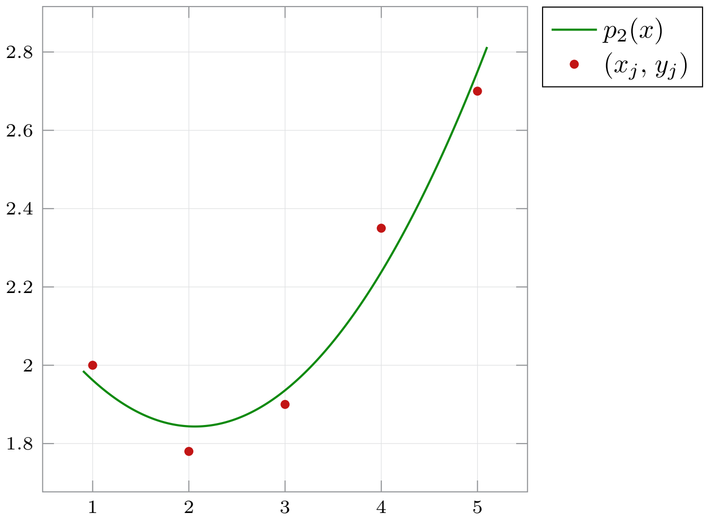
9. Plotting the best fit parabola,

$-11.36 - 5.45x - 0.589x^2$

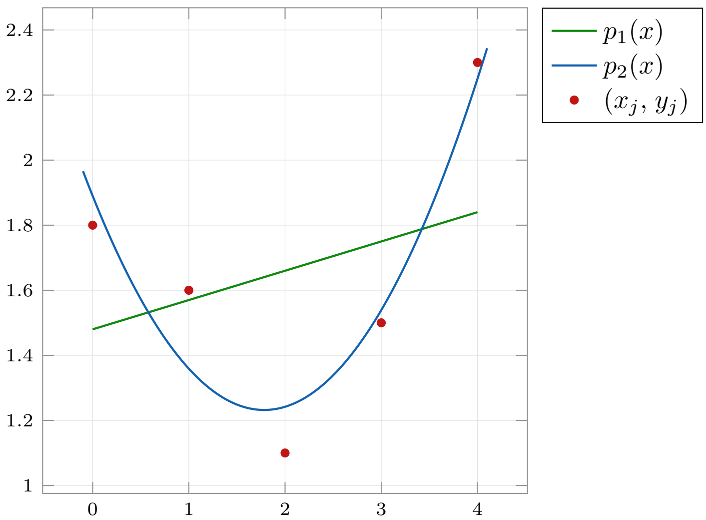


10. Plotting the best fit parabola,

$2.29 - 0.433x + 0.105x^2$



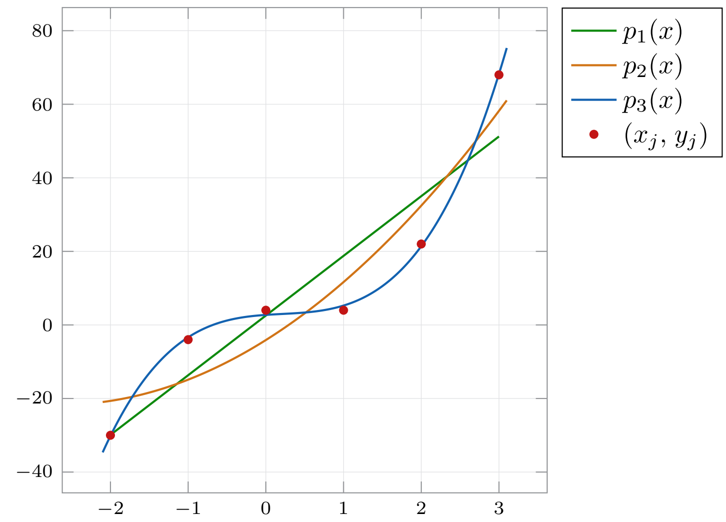
11. Plotting the best fit line, which is a bad fit, and the parabola, which is a better fit



12. Using the derivation of the quadratic polynomial fit, the cubic fit is given in matrix form by

$$\begin{bmatrix} \sum 1 & \sum x_j & \sum x_j^2 & \sum x_j^3 \\ \sum x_j & \sum x_j^2 & \sum x_j^3 & \sum x_j^4 \\ \sum x_j^2 & \sum x_j^3 & \sum x_j^4 & \sum x_j^5 \\ \sum x_j^3 & \sum x_j^4 & \sum x_j^5 & \sum x_j^6 \end{bmatrix} \begin{bmatrix} b_0 \\ b_1 \\ b_2 \\ b_3 \end{bmatrix} = \begin{bmatrix} \sum y_j \\ \sum x_j y_j \\ \sum x_j^2 y_j \\ \sum x_j^3 y_j \end{bmatrix}$$ 20.5.5

13. The highest degree polynomial fit is the best, simply due to the larger number of fitting parameters.



14. Least squares approximation of a function

(a) From the definition of the l_2 norm,

$$\|f - F_m\|^2 = \int_a^b \Big[f - F_m(x)\Big]^2 \ \mathrm{d}x \tag{20.5.6}$$

Using differentiation under the integral sign,

$$\frac{\partial}{\partial a_j} \ \|f(x) - F_m(x)\|^2 = 0 \tag{20.5.7}$$

$$2 \int_a^b y_j(x) \ \Big[f(x) - F_m(x)\Big] \ \mathrm{d}x = 0 \tag{20.5.8}$$

$$\int_a^b f(x) \ y_j(x) \ \mathrm{d}x = \int_a^b y_j(x) \ \sum_{k=0}^{m} a_k \ y_k(x) \ \mathrm{d}x \tag{20.5.9}$$

$$b_j = \sum_{k=0}^{m} a_k \ \int_a^b y_j(x) \ y_k(x) \ \mathrm{d}x \tag{20.5.10}$$

$$b_j = \sum_{k=0}^{m} a_k \ h_{jk} \tag{20.5.11}$$

(b) If the individual functions are powers of x, then

$$h_{jk} = \int_a^b x^{j+k} \ \mathrm{d}x \qquad\qquad b_j = \int_a^b f(x) \ x^j \ \mathrm{d}x \tag{20.5.12}$$

Now, for the special case where $[a,\, b] = [0,\, 1]$,

$$h_{jk} = \frac{1}{j+k+1} \tag{20.5.13}$$

$$j,\, k \in \{0,\, 1,\, 2,\, \ldots,\, m\} \tag{20.5.14}$$

$$\mathbf{H} = \begin{bmatrix} 1 & 1/2 & \ldots & 1/(m+1) \\ 1/2 & 1/3 & \vdots & 1/(m+2) \\ \vdots & \vdots & \ddots & \vdots \\ 1/(m+1) & 1/(m+2) & \ldots & 1/(2m+1) \end{bmatrix} \tag{20.5.15}$$

This is a Hilbert matrix.

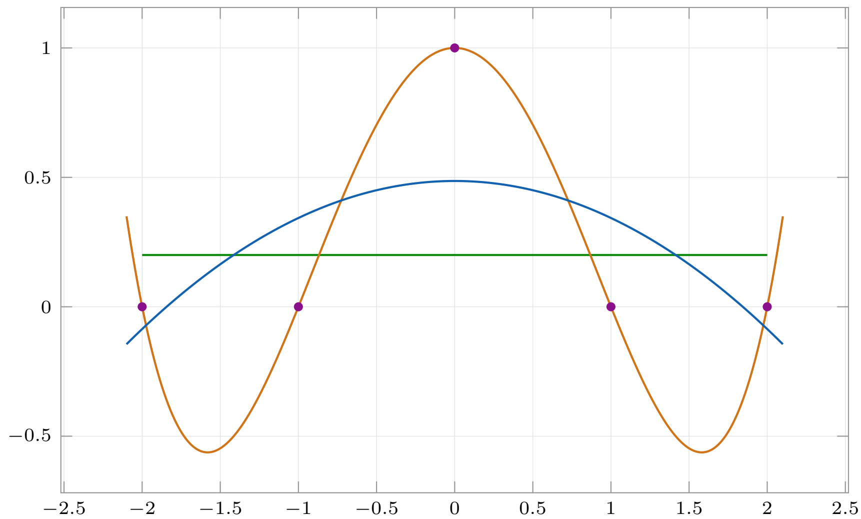
(c) The coefficient matrix reduces to a diagonal matrix, if the functions $\{y_k\}$ are orthogonal.

$$0\; a_0 + \cdots + h_{jj}\; a_j + \cdots + 0\; a_m = b_j \tag{20.5.16}$$

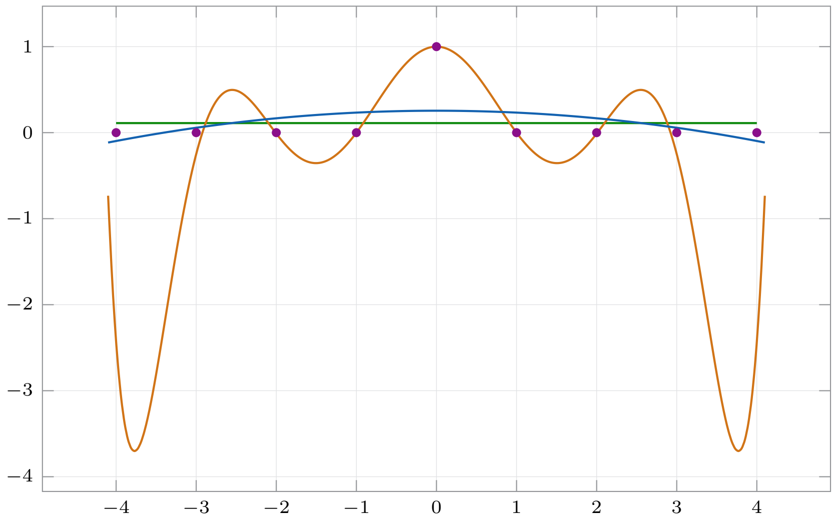
$$a_j = \frac{b_j}{h_{jj}} \tag{20.5.17}$$

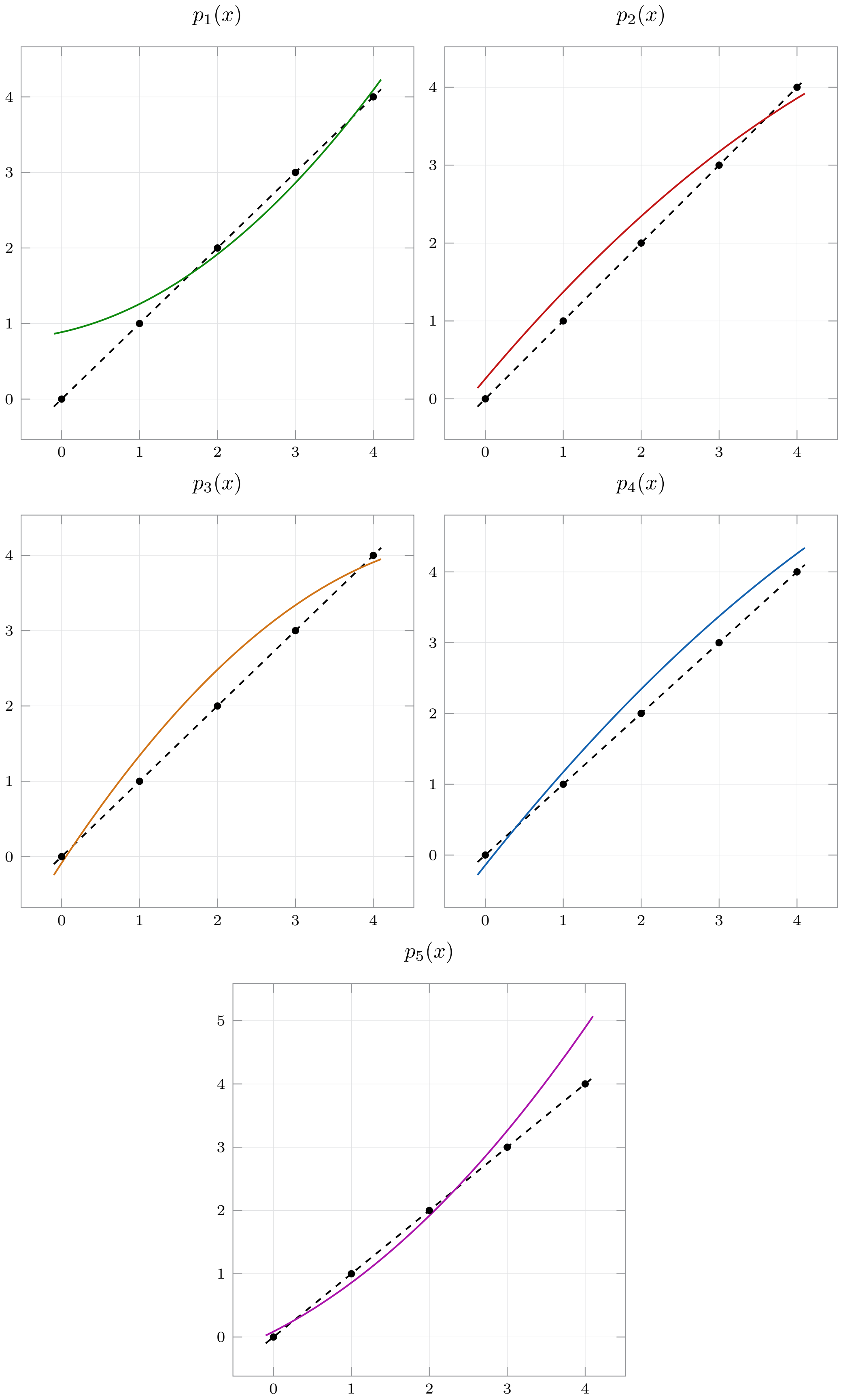
15. Plotting the linear and quadratic least squares best fit, along with the fourth degree interpolating polynomial,

Interpolant and Least Sq.



Interpolant and Least Sq.





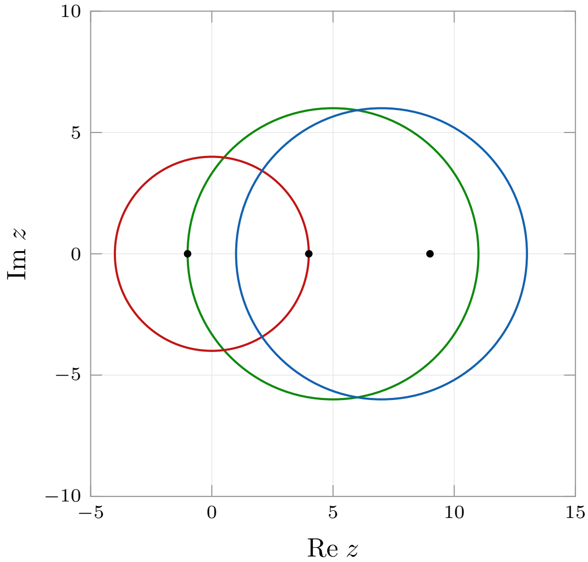
$p_1(x)$
$p_2(x)$
$p_3(x)$
$p_4(x)$
$p_5(x)$

20.6 Matrix Eigenvalue Problems: Introduction

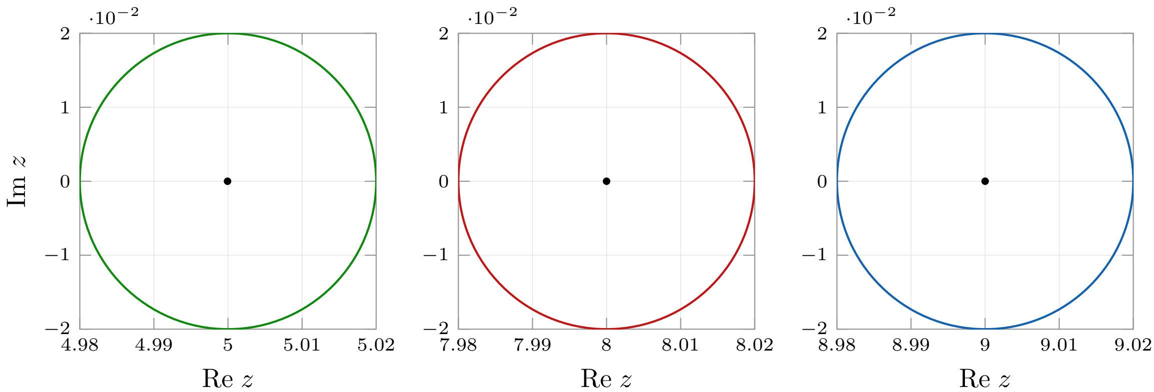
1. No problem set in this section.

20.7 Inclusion of Matrix Eigenvalues

1. Plotting the Gerschgorin disks, and using a CAS to plot the eigenvalues,

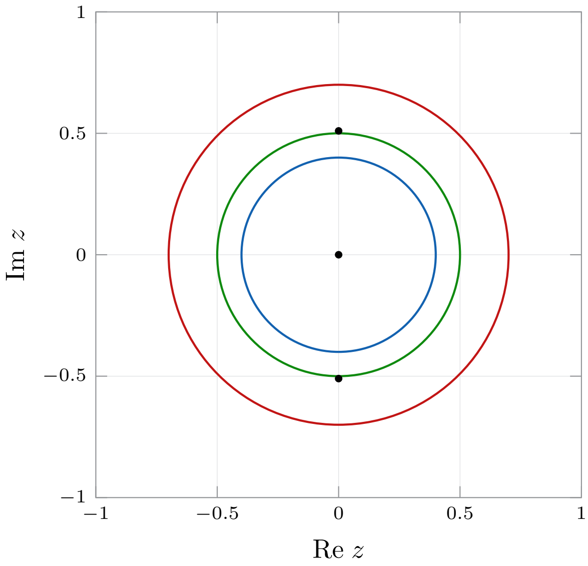


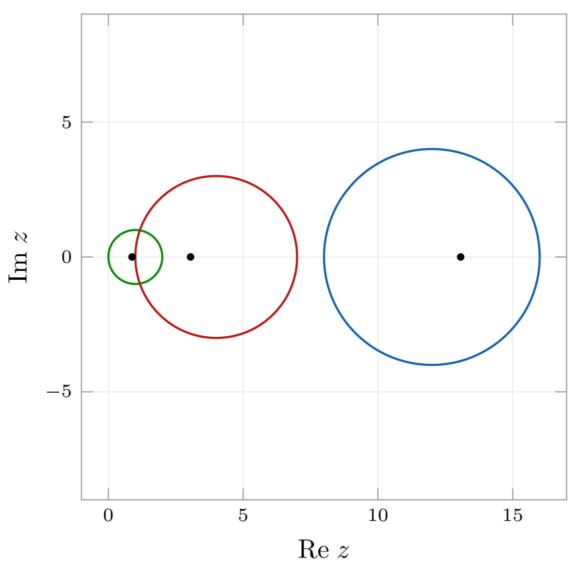
2. Plotting the Gerschgorin disks, and using a CAS to plot the eigenvalues,



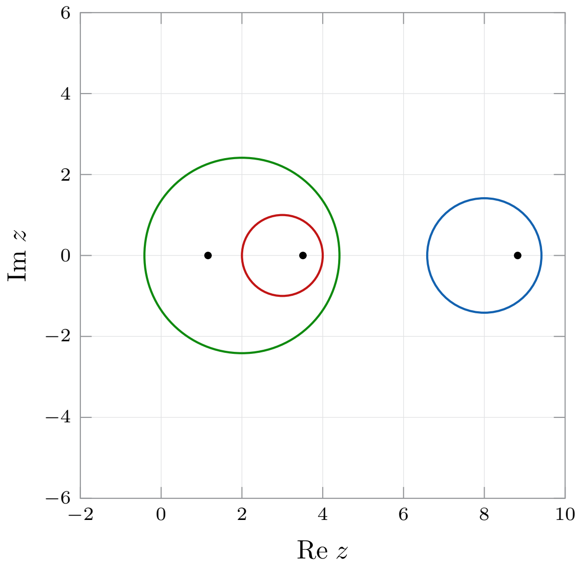
3. Plotting the Gerschgorin disks, and using a CAS to plot the eigenvalues,

4. Plotting the Gerschgorin disks, and using a CAS to plot the eigenvalues,

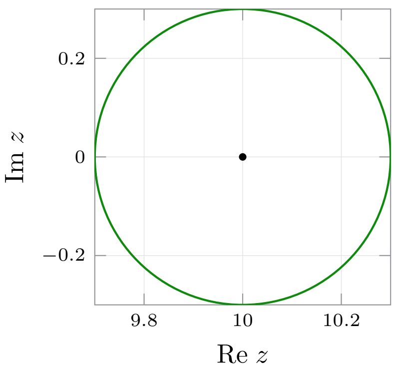


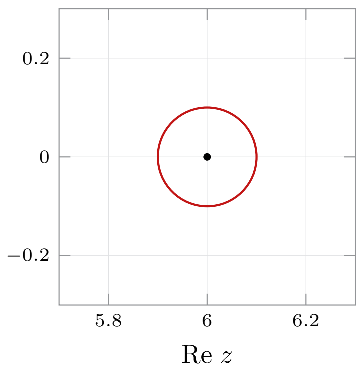


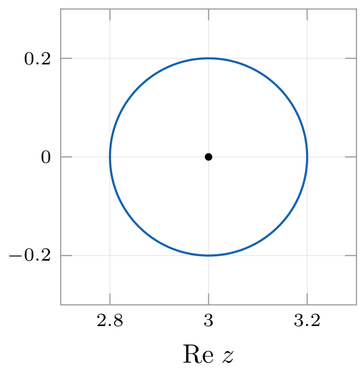
5. Plotting the Gerschgorin disks, and using a CAS to plot the eigenvalues,



6. Plotting the Gerschgorin disks, and using a CAS to plot the eigenvalues,







7. Using a similarity transformation,

$$\mathbf{T} = \begin{bmatrix} 100 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad \mathbf{T}^{-1}\mathbf{A}\mathbf{T} = \begin{bmatrix} 5 & 10^{-4} & 10^{-4} \\ 1 & 8 & 10^{-2} \\ 1 & 10^{-2} & 9 \end{bmatrix} \tag{20.7.1}$$

$$r_5 = 2 \times 10^{-4} \tag{20.7.2}$$

8. Using a similarity transformation,

$$\mathbf{T} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & k \end{bmatrix} \qquad \mathbf{A} = \begin{bmatrix} 10 & 0.1 & 0.2 \\ 0.1 & 6 & 0 \\ -0.2 & 0 & 3 \end{bmatrix} \tag{20.7.3}$$

$$\mathbf{T}^{-1}\mathbf{A}\mathbf{T} = \begin{bmatrix} 10 & 0.1 & -0.2k \\ 0.1 & 6 & 0 \\ -0.2/k & 0 & 3 \end{bmatrix} \tag{20.7.4}$$

Using Gerschgorin's extension, the disk for the third row has to remain disjoint from the other disks.

$$10 - 0.1 - 0.2k > 3 + \frac{0.2}{k} \qquad 0.2k^2 - 6.9k + 0.2 < 0 \tag{20.7.5}$$

$$k_1 = 34.37 \qquad k_2 = 0.03 \tag{20.7.6}$$

This means the largest possible integer is $k^* = 34$.

9. The eigenvalues are located at the diagonal entries with an error which is a disk of radius $(n-1) \cdot 10^{-5}$. Since the matrix has n rows, the other $(n-1)$ entries can at most have size 10^{-5}, which makes the maximum radius of the Gerschgorin disk proportional to $(n-1)$

10. From Problem 1, the eigenvalue $\lambda = 4$ lies on the disk $|z - 0| \leq |-2| + |2|$

11. Looking at the sum of aboslute values of elements within a row,

$$|a_{j1}| + \cdots + |a_{jn}| \geq |\lambda - a_{jj}| + |a_{jj}| \geq |\lambda| \tag{20.7.7}$$

using the triangle inqeuality. Maximizing the left side over all columns,

$$\max_j \sum_{k=1}^{n} |a_{jk}| \geq |\lambda| \tag{20.7.8}$$

$$\|\mathbf{A}\|_\infty \geq \rho(\mathbf{A}) \tag{20.7.9}$$

Since this is true for all the eigenvalues of the matrix, the quantity on the right is the spectral radius.

12. Using Schur's inequality to find an upper bound on the spectral radius,

$$\sum_{k=1}^{n}\sum_{j=1}^{n}|a_{jk}|^2 = 181 \qquad |\lambda| \le \sqrt{181} \tag{20.7.10}$$

13. Using Schur's inequality to find an upper bound on the spectral radius,

$$\sum_{k=1}^{n}\sum_{j=1}^{n}|a_{jk}|^2 = 122 \qquad |\lambda| \le \sqrt{122} \tag{20.7.11}$$

14. Using Schur's inequality to find an upper bound on the spectral radius,

$$\sum_{k=1}^{n}\sum_{j=1}^{n}|a_{jk}|^2 = 145.1 \qquad |\lambda| \le \sqrt{145.1} \tag{20.7.12}$$

15. Using Schur's inequality to find an upper bound on the spectral radius,

$$\sum_{k=1}^{n}\sum_{j=1}^{n}|a_{jk}|^2 = 0.52 \qquad |\lambda| \le \sqrt{0.52} \tag{20.7.13}$$

16. Using Schur's inequality to find an upper bound on the spectral radius,

$$\sum_{k=1}^{n}\sum_{j=1}^{n}|a_{jk}|^2 = 83 \qquad |\lambda| \le \sqrt{83} \tag{20.7.14}$$

17. Verifying that the matrix is normal,

$$\mathbf{A} = \begin{bmatrix} 2 & \mathrm{i} & 1+\mathrm{i} \\ -\mathrm{i} & 3 & 0 \\ 1-\mathrm{i} & 0 & 8 \end{bmatrix} \qquad \mathbf{A}^\dagger = \begin{bmatrix} 2 & \mathrm{i} & 1+\mathrm{i} \\ -\mathrm{i} & 3 & 0 \\ 1-\mathrm{i} & 0 & 8 \end{bmatrix} \tag{20.7.15}$$

$$\mathbf{A}^\dagger = \mathbf{A} \qquad \mathbf{A}^\dagger\mathbf{A} = \mathbf{A}\mathbf{A}^\dagger \tag{20.7.16}$$

18. For Hermitian matrices,

$$\mathbf{A}^\dagger = \mathbf{A} \qquad \mathbf{A}^\dagger\mathbf{A} = \mathbf{A}^2 = \mathbf{A}\mathbf{A}^\dagger \tag{20.7.17}$$

For skew-Hermitian matrices,

$$\mathbf{A}^\dagger = -\mathbf{A} \qquad \mathbf{A}^\dagger\mathbf{A} = -\mathbf{A}^2 = \mathbf{A}\mathbf{A}^\dagger \tag{20.7.18}$$

For unitary matrices,

$$\mathbf{A}^\dagger = \mathbf{A}^{-1} \qquad \mathbf{A}^\dagger\mathbf{A} = \mathbf{I} = \mathbf{A}\mathbf{A}^\dagger \tag{20.7.19}$$

19. From Gerschgorin's theorem and given the matrix is diagonally dominant,

$$|\lambda - a_{jj}| \leq \sum_{k\neq j} |a_{jk}| \qquad \sum_{k\neq j} |a_{jk}| \leq |a_{jj}| \tag{20.7.20}$$

$$|\lambda - a_{jj}| \leq |a_{jj}| \tag{20.7.21}$$

The eigenvalue is located inside a disk that excludes the origin. This means that zero is never one of the eigenvalues.

$$\lambda_i \neq 0 \qquad \forall \quad i \in \{1, 2, \dots, n\} \tag{20.7.22}$$

$$\det(\mathbf{A}) = \prod_{i=1}^{n} \lambda_i \neq 0 \tag{20.7.23}$$

Since the determinant is nonzero, the matrix is singular.

20. Split $\mathbf{A}$ into a diagonal matrix and the remainder, for some real scalar t,

$$\mathbf{A} = \mathbf{B} + \mathbf{C} \qquad \mathbf{A}_t = \mathbf{B} + t\mathbf{C} \tag{20.7.24}$$

$$|\lambda - a_{jj}| \leq t \sum_{k\neq j} |a_{jk}| \tag{20.7.25}$$

$$t = 0 \qquad \Longrightarrow \qquad \{\lambda_i\} = \{a_{ii}\} \tag{20.7.26}$$

Starting from the $t = 1$ case, where $\mathbf{A}_1 = \mathbf{A}$, assume that one of the disks does not contain an eigenvalue.

As t decreases to zero, the radii of these disks also decrease to zero continuously. At the other extreme, at $t = 0$, the disks reduce to a set of points that are the eigenvalues themselves.

This contradicts the initial assumption that the disk did not initially contain an eigenvalue, which proves the theorem.

20.8 Power Method for Eigenvalues

1. Finding the rayleigh criterion using the power method,

n	q	δ
1	10	3
2	10.99	0.30275
3	10.9999	0.0275

2. Finding the rayleigh criterion using the power method,

n	q	δ
1	0	4
2	6	4
3	7.85	1.231

3. Finding the rayleigh criterion using the power method,

n	q	δ
1	4	1.633
2	4.786	0.6186
3	4.917	0.3985

4. Finding the rayleigh criterion using the power method,

n	q	δ
1	1.333	2.175
2	6.574	1.742
3	7.158	0.474

5. Finding the rayleigh criterion using the power method with scaling,

n	q	δ
1	4	1.633
2	4.786	0.6186
3	4.917	0.3985

6. Finding the rayleigh criterion using the power method with scaling,

n	q	δ
1	12.33333	2.49443
2	12.96211	0.61442
3	12.99797	0.14230

7. Finding the rayleigh criterion using the power method with scaling,

n	q	δ
1	5.5	0.5
2	5.57737	0.31147
3	5.60179	0.18990

8. Finding the rayleigh criterion using the power method with scaling,

n	q	δ
1	10.5	2.9580
2	11.1302	1.3688
3	11.1831	0.9637

9. Given the initial guess is already an eigenvalue,

$$\mathbf{y} = \mathbf{A}\mathbf{x} = \lambda\mathbf{x} \qquad m_0 = \mathbf{x}^T\mathbf{x} \tag{20.8.1}$$

$$m_1 = \lambda\ \mathbf{x}^T\mathbf{x} = \lambda m_0 \qquad m_2 = \lambda^2\ \mathbf{x}^T\mathbf{x} = \lambda^2 m_0 \tag{20.8.2}$$

$$\delta^2 = \lambda^2 - (\lambda)^2 = 0 \qquad \delta = 0 \tag{20.8.3}$$

10. Using the set of real unit eigenvectors $\{\mathbf{z_1}, \mathbf{z_2}, \dots, \mathbf{z_n}\}$

$$\mathbf{x} = a_1\mathbf{z}_1 + \dots + a_n\mathbf{z}_n \qquad \mathbf{x}^T\mathbf{x} = a_1^2 + \dots + a_n^2 \tag{20.8.4}$$

$$\mathbf{y} = \lambda_1 a_1\mathbf{z}_1 + \dots + \lambda_n a_n\mathbf{z}_n \qquad \mathbf{x}^T\mathbf{y} = (\lambda_1 a_1)^2 + \dots + (\lambda_n a_n)^2 \tag{20.8.5}$$

Subsitutting these values into the rayleigh quotient, where λ_1 is the dominant eigenvalue,

$$q = \frac{m_1}{m_0} = \frac{\mathbf{x}^T\mathbf{y}}{\mathbf{x}^T\mathbf{x}} \qquad q^{(1)} = \frac{\sum_j \lambda_j^2\ a_j^2}{\sum_j a_j^2} \tag{20.8.6}$$

$$q^{(m)} = \frac{\sum_j \lambda_j^{2m+1}\ a_j^2}{\sum_j \lambda_j^{2m} a_j^2} \tag{20.8.7}$$

After a large number of iterations,

$$\text{numerator} = \lambda_1^{2m+1}\left[a_1^2 + \left(\frac{\lambda_2}{\lambda_1}\right)^{2m+1} a_2^2 + \dots + \left(\frac{\lambda_n}{\lambda_1}\right)^{2m+1} a_n^2\right] \tag{20.8.8}$$

$$\text{denominator} = \lambda_1^{2m}\left[a_1^2 + \left(\frac{\lambda_2}{\lambda_1}\right)^{2m} a_2^2 + \dots + \left(\frac{\lambda_n}{\lambda_1}\right)^{2m} a_n^2\right] \tag{20.8.9}$$

20.8.10

Since the fractions are all less than 1, they approach zero with increasing m,

$$\lim_{m\to\infty} q^{(m)} = \frac{\lambda_1^{2m+1}\ a_1^2}{\lambda_1^{2m}\ a_1^2} = \lambda_1 \tag{20.8.11}$$

11. Finding the rayleigh criterion using the power method,

n	q	δ
1	1	1.633
5	-2.0515	1.9603
10	-2.9798	0.3173

Using a CAS to look at the spectrum of the original matrix $\mathbf{A}$

$$\{\lambda_i\} = \{0, 3, 5\} \tag{20.8.12}$$

12. Power method

(a) Algorithm coded in `numpy`

n	q
1	16
5	32.3168
10	32.0022
15	32.00000608
20	32.00000012

(b) Using the shifting theorem for $\mathbf{B} = \mathbf{A} + k\mathbf{I}$, the best results for 10 iterations starting from the same initial guess are for $k^* = -10$

(c) Refer problem 8. Coded in `numpy`

(d) The eigenvalues from a CAS are $\{-1, 1\}$. Within machine number, $q \approxeq 0$ for all steps and $\delta = 1$ for all steps.

Since the vector oscillates between the same two values for this initial guess, the iterative process will not converge.

(e) A simple matrix resembling the identity matrix can yield errors larger than this theoretical bound

$$\mathbf{A} = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} \qquad \mathbf{x}_0 = \begin{bmatrix} 3 \\ 1 \end{bmatrix} \tag{20.8.13}$$

$$\mathbf{x}_1 = \begin{bmatrix} 4 \\ 1 \end{bmatrix} \qquad \mathbf{x}_m = \begin{bmatrix} (m+3) \\ 1 \end{bmatrix} \tag{20.8.14}$$

$$\delta_1 = 0.1 \qquad q_1 = 0.3 \tag{20.8.15}$$

This matrix is nonsymmetric but has no dominant eigenvalue. The actual error is larger than δ

(f) Using the `numpy` code, the convergence is faster when the second eigenvalue is smaller than first.

20.9 Tridiagonalization and QR-Factorization

1. Tridiagonalization algorithm written in `numpy`.

$$\mathbf{v}_1 = \begin{bmatrix} 0 \\ 0.7384 \\ 0.6743 \end{bmatrix} \qquad \mathbf{A}_1 = \mathbf{B} = \begin{bmatrix} 0.98 & -0.4418 & 0 \\ -0.4418 & 0.8701 & 0.3718 \\ 0 & 0.3718 & 0.4898 \end{bmatrix} \tag{20.9.1}$$

2. Tridiagonalization algorithm written in `numpy`.

$$\mathbf{v}_1 = \begin{bmatrix} 0 \\ 0.9239 \\ 0.3827 \end{bmatrix} \qquad \mathbf{A}_1 = \mathbf{B} = \begin{bmatrix} 0 & -1.4142 & 0 \\ -1.4142 & 1 & 0 \\ 0 & 0 & -1 \end{bmatrix} \tag{20.9.2}$$

3. Tridiagonalization algorithm written in `numpy`.

$$\mathbf{v}_1 = \begin{bmatrix} 0 \\ 0.8817 \\ 0.4719 \end{bmatrix} \qquad \mathbf{A}_1 = \mathbf{B} = \begin{bmatrix} 7 & -3.6056 & 0 \\ -3.6056 & 13.4615 & 3.6923 \\ 0 & 3.6923 & 3.5385 \end{bmatrix} \tag{20.9.3}$$

4. Tridiagonalization algorithm written in `numpy`.

$$\mathbf{v}_1 = \begin{bmatrix} 0 \\ 0.9856 \\ 0.1196 \\ 0.1196 \end{bmatrix} \qquad \mathbf{A}_1 = \begin{bmatrix} 5 & -4.2426 & 0 & 0 \\ -4.2426 & 6 & -1 & -1 \\ 0 & -1 & 3.5 & 1.5 \\ 0 & -1 & 1.5 & 3.5 \end{bmatrix} \tag{20.9.4}$$

$$\mathbf{v}_2 = \begin{bmatrix} 0 \\ 0 \\ 0.9239 \\ 0.3827 \end{bmatrix} \qquad \mathbf{B} = \begin{bmatrix} 5 & -4.2426 & 0 & 0 \\ -4.2426 & 6 & 1.4142 & 0 \\ 0 & 1.4142 & 5 & 0 \\ 0 & 0 & 0 & 2 \end{bmatrix} \tag{20.9.5}$$

5. Tridiagonalization algorithm written in `numpy`.

$$\mathbf{v}_1 = \begin{bmatrix} 0 \\ 0.9406 \\ 0.0787 \\ 0.3303 \end{bmatrix} \qquad \mathbf{A}_1 = \begin{bmatrix} 3 & -67.587 & -67.587 & 0 \\ -67.587 & 143.5324 & -45.223 & -3.4156 \\ -67.587 & -45.223 & 23.487 & 1.2062 \\ 0 & -3.4156 & 1.2062 & -34.0194 \end{bmatrix}$$ 20.9.6

$$\mathbf{v}_2 = \begin{bmatrix} 0 \\ 0 \\ 0.9993 \\ 0.0377 \end{bmatrix} \qquad \mathbf{B} = \begin{bmatrix} 3 & -67.587 & 0 & 0 \\ -67.587 & 143.5324 & 45.3518 & 0 \\ 0 & 45.3518 & 23.342 & 3.1262 \\ 0 & 0 & 3.1262 & -33.8744 \end{bmatrix}$$ 20.9.7

6. QR factorization coded in `numpy`

n	λ_1	λ_2	λ_3	Error
1	1.2925	0.8402	0.2072	0.2942
5	1.4393	0.7207	0.1800039	0.022478
10	1.439999	0.7200007	0.18	0.000703

7. QR factorization coded in `numpy`

n	λ_1	λ_2	λ_3	Error
1	11.2903	10.6144	2.0952	5.0173
5	15.9966	6.0034	2.000012	0.18492
10	16	6.0000	2	0.001372

8. QR factorization coded in `numpy`

n	λ_1	λ_2	λ_3	Error
1	14.200439177	−6.30462	2.1042282	0.06679
5	14.20048773	−6.30524639	2.10475866	0.001725
10	141.20048787	−6.30524661	2.10475874	$2.997\,766 \times 10^{-5}$

9. QR factorization coded in `numpy`

n	λ_1	λ_2	λ_3	Error
1	141.066	68.9666	−30.0326	4.926
5	141.39996	68.64025	−30.0402101	0.2747
10	141.401	68.63922	−30.04022035	0.0074

10. Coded in `numpy`. The convergence gets faster when the spectral shifting theorem is used to bring the smallest eigenvalue to zero, or when the smallest eigenvalue is already zero.