



1 Det vil ikke bli gitt ut noen løsningsforslag for denne oppgaven, men du kan bruke `numpy.linalg.norm` for å få svarene dine bekreftet.

a) For $\mathbf{x} = (1.2, -4.3, 2.5)^\top$, beregn $\|\mathbf{x}\|_1$, $\|\mathbf{x}\|_2$ og $\|\mathbf{x}\|_\infty$.

b) For

$$A = \begin{pmatrix} 3.0 & 1.4 & -1.2 \\ 2.6 & 5.1 & 1.2 \\ 2.2 & -1.1 & 5.1 \end{pmatrix}$$

Beregn $\|A\|_1$ og $\|A\|_\infty$.

I resten av settet er

$$\mathbf{x} = \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} \in \mathbb{R}^n \text{ og } A = \begin{pmatrix} a_{11} & \dots & a_{1n} \\ \vdots & & \vdots \\ a_{n1} & \dots & a_{nn} \end{pmatrix} \in \mathbb{R}^{n \times n}.$$

2

a) Vis at $\|A\|_1 = \max_j \sum_{i=1}^n |a_{ij}|$ er den tilordnede (naturlige) matrisenormen til $\|\cdot\|_1$.

b) Vis følgende ulikheter:

$$\|\mathbf{x}\|_\infty \leq \|\mathbf{x}\|_2 \leq \|\mathbf{x}\|_1$$

$$\|\mathbf{x}\|_1 \leq n \|\mathbf{x}\|_\infty$$

$$\|\mathbf{x}\|_2 \leq \sqrt{n} \|\mathbf{x}\|_\infty$$

3 Gitt igjen ligningssystemet $\mathbf{Ax} = \mathbf{b}$, og anta at diagonalelementene $a_{ii} \neq 0$. Ligningssystemet kan da skrives som

$$\sum_{j=1}^n a_{ij}x_j = b_i, \quad i = 1, 2, \dots, n, \quad (3.1)$$

som kan skrives om til

$$a_{ii}x_i + \sum_{\substack{j=1 \\ j \neq i}}^n a_{ij}x_j = b_i, \quad i = 1, 2, \dots, n \quad (3.2)$$

eller

$$x_i = \frac{1}{a_{ii}} \left(b_i - \sum_{\substack{j=1 \\ j \neq i}}^n a_{ij}x_j \right), \quad i = 1, 2, \dots, n, \quad (3.3)$$

som igjen (puh!) kan brukes som basis for følgende iterative algoritme (Jakobi-iterasjoner):

Gitt $\mathbf{x}^{(0)}$. For $k = 0, 1, 2, \dots$:

$$x_i^{(k+1)} = \frac{1}{a_{ii}} \left(b_i - \sum_{\substack{j=1 \\ j \neq i}}^n a_{ij}x_j^{(k)} \right), \quad i = 1, 2, \dots, n.$$

I praksis avsluttes iterasjonene hvis $\|\mathbf{x}^{(k+1)} - \mathbf{x}^{(k)}\| \leq \text{Tol}$, der Tol er en eller annen gitt brukertoleranse (vellykket), alternativt hvis maksimalt antall iterasjoner er oppnådd (mislykket).

a) Vis at (3.3) kan skrives på matriseform som

$$\mathbf{x}^{(k+1)} = G\mathbf{x}^{(k)} + \mathbf{c}.$$

Skriv ned G og $\mathbf{c}$.

b) Vis at $\|G\|_\infty < 1$ dersom A er strengt diagonal dominant.

c) Implementer algoritmen på formen (3.3) i Python, og test implementasjonen på matrisen i oppgave 1. Velg for eksempel en høyreside $\mathbf{b}$ slik at den eksakte løsningen av problemet blir $\mathbf{x} = (1, 1, \dots, 1)^\top$, og bruk (igjen for eksempel) $x_i^{(0)} = b_i/a_{ii}$, $i = 1, \dots, n$ som startverdi. Velg selv en toleranse. Vil iterasjonene konvergere? Og hvordan stemmer det med teorien?