Raport z zadania numerycznego 2

Bartłomiej Galek

29 października 2024

1 Treść zadania

(Zadanie numeryczne NUM2)

Zadane są macierze

$$\mathbf{A_1} = \begin{pmatrix} 5.8267103432 & 1.0419816676 & 0.4517861296 & -0.2246976350 & 0.7150286064 \\ 1.0419816676 & 5.8150823499 & -0.8642832971 & 0.6610711416 & -0.3874139415 \\ 0.4517861296 & -0.8642832971 & 1.5136472691 & -0.8512078774 & 0.6771688230 \\ -0.2246976350 & 0.6610711416 & -0.8512078774 & 5.3014166511 & 0.5228116055 \\ 0.7150286064 & -0.3874139415 & 0.6771688230 & 0.5228116055 & 3.5431433879 \end{pmatrix}$$

oraz

$$\mathbf{A_2} = \begin{pmatrix} 5.4763986379 & 1.6846933459 & 0.3136661779 & -1.0597154562 & 0.0083249547 \\ 1.6846933459 & 4.6359087874 & -0.6108766748 & 2.1930659258 & 0.9091647433 \\ 0.3136661779 & -0.6108766748 & 1.4591897081 & -1.1804364456 & 0.3985316185 \\ -1.0597154562 & 2.1930659258 & -1.1804364456 & 3.3110327980 & -1.1617171573 \\ 0.0083249547 & 0.9091647433 & 0.3985316185 & -1.1617171573 & 2.1174700695 \end{pmatrix}.$$

Zdefiniujmy wektor

$$\mathbf{b} \equiv (-2.8634904630, -4.8216733374, -4.2958468309, -0.0877703331, -2.0223464006)^T$$

Używając wybranego pakietu algebry komputerowej lub biblioteki numerycznej, rozwiąż równania macierzowe $\mathbf{A_i y} = \mathbf{b}$ dla $i = 1, 2$. Ponadto, rozwiąż analogiczne równania z zaburzonym wektorem wyrazów wolnych, $\mathbf{A_i y} = \mathbf{b} + \Delta\mathbf{b}$. Zaburzenie $\Delta\mathbf{b}$ wygeneruj jako losowy wektor o małej normie euklidesowej (np. $\|\Delta\mathbf{b}\|_2 \approx 10^{-6}$). Przeanalizuj jak wyniki dla macierzy $\mathbf{A_1}$ i $\mathbf{A_2}$ zależą od $\Delta\mathbf{b}$ i zinterpretuj zaobserwowane różnice.

2 Wstęp

Szukamy rozwiązań dwóch układów równań. W tym celu przygotowujemy program w Pythonie, który (korzystając z faktoryzacji LU z częściowym pivotingiem) rozwiąże te dwa równania. Tworzymy funkcję która będzie losowo generować zaburzenie rzędu $\approx 10^{-6}$. Mając już wygenerowane zaburzenie, rozwiązujemy równania dla zaburzonego wektora b. W celu sprawdzenia wpływu zaburzenia na wynik, znajdujemy różnicę między wektorem b bez zaburzenia oraz zaburzonym b. Następnie obliczamy wskaźnik uwarunkowania obu macierzy. Mając różnice w wynikach i wskaźniki, możemy przeanalizować wpływ $\Delta\mathbf{b}$ na wyniki.

3 Implementacja

3.1 Opis implementacji

W programie num2.py użyto biblioteki numpy, a konkretniej modułu linalg w celu użycia funkcji norm, cond i solve, gdzie:

- norm - funkcja obliczająca normę wektora/macierzy
- cond - funkcja obliczająca wskaźnik uwarunkowania
- solve - funkcja rozwiązująca układy równań liniowych (oblicza rozwiązanie metodą gesv pakietu LAPACK, która korzysta z faktoryzacji LU z częściowym pivotingiem)

3.2 Kod źródłowy

```
import numpy as np
from numpy.linalg import norm, cond, solve

A1 = np.array([
    [5.8267103432, 1.0419816676, 0.4517861296,
       -0.2246976350, 0.7150286064],
    [1.0419816676, 5.8150823499, -0.8642832971,
       0.6610711416, -0.3874139415],
    [0.4517861296, -0.8642832971, 1.5136472691,
       -0.8512078774, 0.6771688230],
    [-0.2246976350, 0.6610711416, -0.8512078774,
       5.3014166511, 0.5228116055],
```

```
    [0.7150286064, -0.3874139415, 0.6771688230, 0.5228116055, 3.5431433879]
])

A2 = np.array([
    [5.4763986379, 1.6846933459, 0.3136661779, -1.0597154562, 0.0083249547],
    [1.6846933459, 4.6359087874, -0.6108766748, 2.1930659258, 0.9091647433],
    [0.3136661779, -0.6108766748, 1.4591897081, -1.1804364456, 0.3985316185],
    [-1.0597154562, 2.1930659258, -1.1804364456, 3.3110327980, -1.1617171573],
    [0.0083249547, 0.9091647433, 0.3985316185, -1.1617171573, 2.1174700695]
])

b = np.array([-2.8634904630, -4.8216733374, -4.2958468309, -0.0877703331, -2.0223464006])

X1 = solve(A1, b)
X2 = solve(A2, b)
print(f"Base solution for matrix A1:\n{X1}")
print(f"Base solution for matrix A2:\n{X2}")

def generate_perturbation(size, magnitude = 1e-6):
    db = np.random.normal(0, magnitude, size)
    db = db * (magnitude / norm(db))
    return db

perturbation = b + generate_perturbation(len(b))

dX1 = solve(A1, perturbation)
dX2 = solve(A2, perturbation)

print(f"Perturbated solution for matrix A1:\n{dX1}")
print(f"Perturbated solution for matrix A2:\n{dX2}")


differenceX1 = abs(X1 - dX1)
differenceX2 = abs(X2 - dX2)
print(f"Difference between base and perturbated solution for matrix A1:\n{differenceX1}")
```

```python
print(f"Difference between base and perturbated solution
    for matrix A2:\n{differenceX2}")

condA1 = cond(A1)
condA2 = cond(A2)

print(f"Condition number for matrix A1:\n{condA1}")
print(f"Condition number for matrix A2:\n{condA2}")
```

4 Analiza wyników

4.1 Prezentacja wyników

1. Rozwiązania bazowe

Macierz A_1:

$$y_1 = \begin{bmatrix}0.02556195 & -1.35714283 & -3.94075752 & -0.48893629 & 0.10097805\end{bmatrix}^T$$

Macierz A_2:

$$y_2 = \begin{bmatrix}-0.408759 & -0.56030065 & -4.11200041 & -1.5242021 & -0.7752022\end{bmatrix}^T$$

2. Rozwiązania dla układów zaburzonych

Macierz A_1:

$$\tilde{y}_1 = \begin{bmatrix}0.02556206 & -1.35714292 & -3.94075749 & -0.48893612 & 0.10097803\end{bmatrix}^T$$

Macierz A_2:

$$\tilde{y}_2 = \begin{bmatrix}3740.36811226 & -6863.70801905 & 1470.79138773 & 8915.15001986 & 7545.70593169\end{bmatrix}^T$$

3. Różnice między rozwiązaniami bazowymi a zaburzonymi

Macierz A_1:

$$\Delta y_1 = \begin{bmatrix}1.13 \times 10^{-7} & 8.60 \times 10^{-8} & 2.71 \times 10^{-8} & 1.72 \times 10^{-7} & 2.39 \times 10^{-8}\end{bmatrix}^T$$

Macierz A_2:

$$\Delta y_2 = \begin{bmatrix}3740.77687126 & 6863.14771840 & 1474.90338814 & 8916.67422196 & 7546.48113390\end{bmatrix}^T$$

4. Wskaźniki uwarunkowania macierzy

- Macierz A_1: $\kappa(A_1) = 7.000000008382097$
- Macierz A_2: $\kappa(A_2) = 1.16452 \times 10^{11}$

4.2 Analiza wyników

1. Macierz A_1 jest znacznie lepiej uwarunkowana niż macierz A_2, co widać po wskaźnikach uwarunkowania różniących się o kilka rzędów wielkości.
2. Dla macierzy A_1 różnice między rozwiązaniem bazowym a zaburzonym są rzędu 10^{-7} - 10^{-8}, co świadczy o stabilności rozwiązania.
3. Dla macierzy A_2 zaburzenia prowadzą do dramatycznych zmian w rozwiązaniu - różnice są rzędu 10^3 - 10^4, co potwierdza złe uwarunkowanie układu.

5 Podsumowanie

Rozwiązywanie równań liniowych korzystając z metod numerycznych jest skuteczne tylko wtedy, gdy nasz problem jest dobrze uwarunkowany numerycznie. Wtedy małe zmiany w warunkach początkowych prowadzą do małych zmian w wyniku. Jednakże, jeśli nasz problem jest źle uwarunkowany (czyli posiada wysoki wskaźnik uwarunkowania jak macierz A2) to bez wględu na precyzję naszej metody rozwiązującej, małe zmiany będą prowadziły do drastycznych różnic w wyniku. Dlatego przed rozpoczęciem rozwiązywania danego problemu, trzeba sprawdzić czy posiada niski wskaźnik uwarunkowania, ponieważ tylko wtedy warto jest poświęcać czas i zasoby. W przeciwnym przypadku stracimy je na niestabilne rozwiązanie.