Zadanie numeryczne 2

Pola Dudek

29 października 2024

1 Wstęp

Rozwiązywanie układów równań macierzowych to jedno z fundamentalnych zagadnień algebry liniowej. Proces ten może być realizowany przy pomocy metod numerycznych, a co za tym idzie, powinien być analizowany pod kątem stabilności numerycznej.

Celem zadania jest analiza rozwiązań równań macierzowych $A_i x = b$ dla dwóch zadanych macierzy rzeczywistych symetrycznych

$$A_1 = \begin{pmatrix} 5.8267 & 1.0420 & 0.4518 & -0.2247 & 0.7150 \\ 1.0420 & 5.8151 & -0.8643 & 0.6611 & -0.3874 \\ 0.4518 & -0.8643 & 1.5136 & -0.8512 & 0.6772 \\ -0.2247 & 0.6611 & -0.8512 & 5.3014 & 0.5228 \\ 0.7150 & -0.3874 & 0.6772 & 0.5228 & 3.5431 \end{pmatrix}$$

$$A_2 = \begin{pmatrix} 5.4764 & 1.6847 & 0.3137 & -1.0597 & 0.0083 \\ 1.6847 & 4.6359 & -0.6109 & 2.1931 & 0.9092 \\ 0.3137 & -0.6109 & 1.4592 & -1.1804 & 0.3985 \\ -1.0597 & 2.1931 & -1.1804 & 3.3110 & -1.1617 \\ 0.0083 & 0.9092 & 0.3985 & -1.1617 & 2.1175 \end{pmatrix}$$

oraz wektora wyrazów wolnych b.

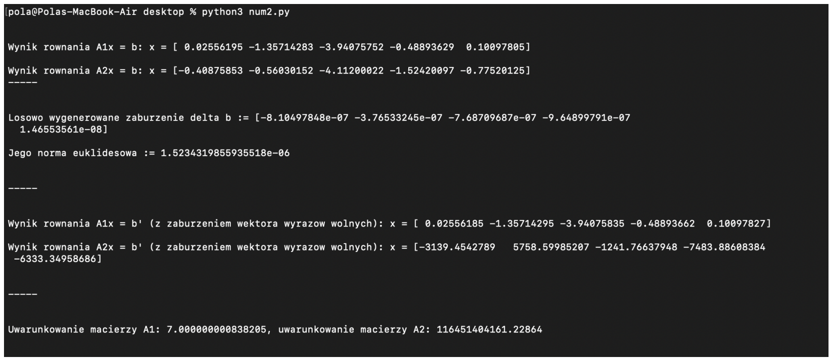
$$b = \begin{pmatrix} -2.8635 \\ -4.8217 \\ -4.2958 \\ -0.0878 \\ -2.0223 \end{pmatrix}$$

wraz ze wpływem losowo wygenerowanego wektora zaburzenia Δb o bardzo małej normie euklidesowej $||\Delta b||_2 \approx 10^{-6}$: $A_i x = b + \Delta b$

Program napisany został w języku Python i używa biblioteki NumPy, w szczególności jej modułu algebry liniowej *linalg*, i jest podzielony na poniższe etapy:

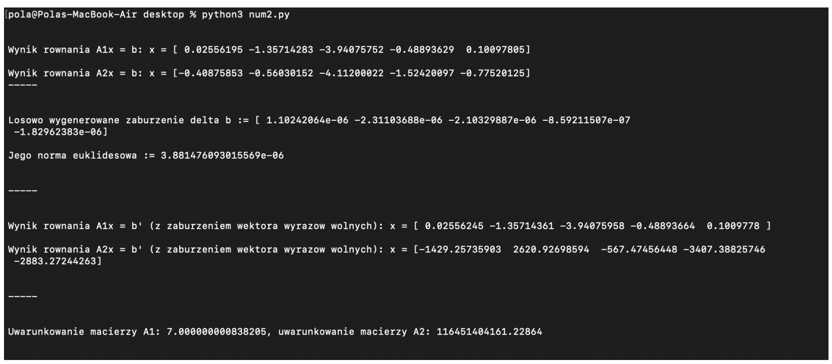
- Rozwiązanie równań macierzowych $A_i x = b$, dla $i = 1, 2$ funkcją `solve()` oraz wypisanie wyniku.
- Rozwiązanie równań $A_i x = b + \Delta b$, gdzie Δb to losowy wektor o małej normie euklidesowej, dla $i = 1, 2$ oraz wypisanie wyniku.
- Jawne wypisanie losowo wygenerowanego wektora zaburzeń Δb oraz podanie jego normy euklidesowej - funkcja `norm()`. Wektor ten generowany jest przy każdym uruchomieniu programu.
- Dodatkowo obliczenie liczby uwarunkowania macierzy A_1 i A_2 funkcją `cond()`, w celu potwierdzenia hipotezy o ich uwarunkowaniu.

2 Wyniki



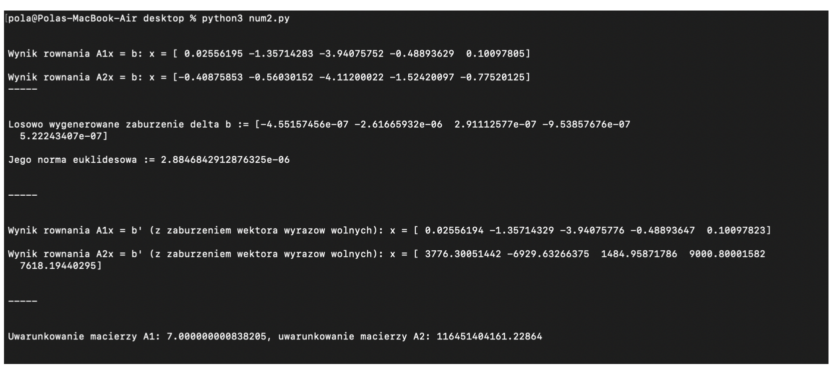
```
pola@Polas-MacBook-Air desktop % python3 num2.py

Wynik rownania A1x = b: x = [ 0.02556195 -1.35714283 -3.94075752 -0.48893629  0.10097805]

Wynik rownania A2x = b: x = [-0.40875853 -0.56030152 -4.11200022 -1.52420097 -0.77520125]
-----

Losowo wygenerowane zaburzenie delta b := [-8.10497848e-07 -3.76533245e-07 -7.68709687e-07 -9.64899791e-07
  1.46553561e-08]

Jego norma euklidesowa := 1.5234319855935518e-06

-----

Wynik rownania A1x = b' (z zaburzeniem wektora wyrazow wolnych): x = [ 0.02556185 -1.35714295 -3.94075835 -0.48893662  0.10097827]

Wynik rownania A2x = b' (z zaburzeniem wektora wyrazow wolnych): x = [-3139.4542789   5758.59985207 -1241.76637948 -7483.88608384
 -6333.34958686]

-----

Uwarunkowanie macierzy A1: 7.000000000838205, uwarunkowanie macierzy A2: 116451404161.22864
```

Rysunek 1: Terminal, zrzut ekranu przy pierwszym uruchomieniu programu



```
pola@Polas-MacBook-Air desktop % python3 num2.py

Wynik rownania A1x = b: x = [ 0.02556195 -1.35714283 -3.94075752 -0.48893629  0.10097805]

Wynik rownania A2x = b: x = [-0.40875853 -0.56030152 -4.11200022 -1.52420097 -0.77520125]
-----

Losowo wygenerowane zaburzenie delta b := [ 1.10242064e-06 -2.31103688e-06 -2.10329887e-06 -8.59211507e-07
 -1.82962383e-06]

Jego norma euklidesowa := 3.881476093015569e-06

-----

Wynik rownania A1x = b' (z zaburzeniem wektora wyrazow wolnych): x = [ 0.02556245 -1.35714361 -3.94075958 -0.48893664  0.1009778 ]

Wynik rownania A2x = b' (z zaburzeniem wektora wyrazow wolnych): x = [-1429.25735903  2620.92698594  -567.47456448 -3407.38825746
 -2883.27244263]

-----

Uwarunkowanie macierzy A1: 7.000000000838205, uwarunkowanie macierzy A2: 116451404161.22864
```

Rysunek 2: Terminal, zrzut ekranu przy drugim uruchomieniu programu



```
pola@Polas-MacBook-Air desktop % python3 num2.py

Wynik rownania A1x = b: x = [ 0.02556195 -1.35714283 -3.94075752 -0.48893629  0.10097805]

Wynik rownania A2x = b: x = [-0.40875853 -0.56030152 -4.11200022 -1.52420097 -0.77520125]
-----

Losowo wygenerowane zaburzenie delta b := [-4.55157456e-07 -2.61665932e-06  2.91112577e-07 -9.53857676e-07
  5.22243407e-07]

Jego norma euklidesowa := 2.8846842912876325e-06

-----

Wynik rownania A1x = b' (z zaburzeniem wektora wyrazow wolnych): x = [ 0.02556194 -1.35714329 -3.94075776 -0.48893647  0.10097823]

Wynik rownania A2x = b' (z zaburzeniem wektora wyrazow wolnych): x = [ 3776.30051442 -6929.63266375  1484.95871786  9000.80001582
  7618.19440295]

-----

Uwarunkowanie macierzy A1: 7.000000000838205, uwarunkowanie macierzy A2: 116451404161.22864
```

Rysunek 3: Terminal, zrzut ekranu przy trzecim uruchomieniu programu

2.1 Porównanie wyników dla A_1 i A_2 wraz ze wpływem Δb

Rozwiązania uzyskane w obecności zaburzenia Δb, jak można było się spodziewać, różnią się od poprzednio otrzymanych wyników. Dla macierzy A_1 wartości wektora x różnią się nieznacznie, jednak w przypadku macierzy A_2 wpływ zaburzenia jest bardzo zauważalny, a wynik uległ zmianie o nawet tysięczne wartości.

Tabela 1: Wartości wektora x oraz x' dla macierzy A_1, gdzie x' to wynik otrzymany w obecności zaburzenia Δb dla pierwszego uruchomienia programu.

Indeks	x	x'	Różnice
1	0.02556195	0.02556185	0.00000010
2	-1.35714283	-1.35714295	-0.00000012
3	-3.94075752	-3.94075835	-0.00000083
4	-0.48893629	-0.48893662	-0.00000033
5	0.10097805	0.10097827	-0.00000022

Tabela 2: Jak wyżej, dla macierzy A_2.

Indeks	x	x'	Różnice
1	-0.40875853	-3139.4542789	-3139.04552037
2	-0.56030152	5758.59985207	5759.16015359
3	-4.11200022	-1241.76637948	2872.34562074
4	-1.52420097	-7483.88608384	-7482.36188287
5	-0.77520125	-6333.34958686	-6332.57438561

3 Dyskusja

Analizując wyniki otrzymane dla obu macierzy, można wyprowadzić następujące spostrzeżenia:

- Macierz A_2 wykazuje zdecydowanie dużo większą wrażliwość na zaburzenia Δb niż macierz A_1. Opierając się na analizie numerycznej nasuwa to wniosek, że macierz A_2 jest źle warunkowana - to znaczy ma dużą liczbę uwarunkowania.

  Liczba uwarunkowania macierzy jest pojęciem analizy numerycznej i mówi o tym jaki wpływ na wyniki mają zmiany danych wejściowych. Liczba uwarunkowania macierzy A, oznaczana jest jako $cond(A)$, z definicji:

  $$cond(A) = \|A\| \cdot \|A^{-1}\|$$

  gdzie

  - $\|A\|$ to norma macierzy A (na przykład norma euklidesowa),
  - $\|A^{-1}\|$ to norma odwrotności macierzy A.

Nie zaleca się jednak stosowania powyższego wzoru, ponieważ wymaga on obliczenia odwrotności macierzy A, co jest niezwykle kosztowne obliczeniowo. Istnieje jednak alternatywny wzór:

$$cond(A) = \frac{\sigma_{\max}}{\sigma_{\min}}$$

gdzie $\sigma_{\max}$ oraz $\sigma_{\min}$ to odpowiednio największa i najmniejsza wartość singularna macierzy A.

W celu potwierdzenia przypuszczeń, użyłam funkcji bibliotki Numpy.linalg `cond()` która zwraca liczbę uwarunkowania danej macierzy. Zgodnie z przewidywaniami otrzymana liczba jest wielocyfrowa i wynosi dokładnie 116451404161.22864 - to poświadcza złe uwarunkowanie macierzy A_2

- Dla macierzy A_1 zaburzenie powoduje natomiast nieznaczne zmiany w danych wyjściowych - świadczy to o jej dobrym uwarunkowaniu. Potwierdza to jej liczba uwarunkowania uzyskana funkcją `cond()`, która wynosi 7.000000000838205.

- Ponadto dla danych z pierwszego uruchomienia programu (Tabela 1 i 2) sprawdziłam normę euklidesową wektora różnicy wyników (nazwijmy go d:

  Dla macierzy A_1:

$$\mathbf{d_1} = \begin{bmatrix} 0.00000010 \\ -0.00000012 \\ -0.00000083 \\ -0.00000033 \\ -0.00000022 \end{bmatrix}$$

  Dla macierzy A_2:

$$\mathbf{d_2} = \begin{bmatrix} -3139.04552037 \\ 5759.16015359 \\ 2872.34562074 \\ -7482.36188287 \\ -6332.57438561 \end{bmatrix}$$

  Po wykonaniu obliczeń, kolejno otrzymano wyniki: $\|d_1\|_2 \approx 9.33 \times 10^{-7}$, $\|d_2\|_2 \approx 12139.16$. Norma wektora różnic dla macierzy A_1 jest bardzo mała - wyniki x i x' są bardzo zbliżone. Natomiast norma wektora różnicy dla macierzy A_2 jest znacznie większa - wyniki x i x' znacznie od siebie odbiegają. Można więc zauważyć zależność pomiędzy normą wektora różnicy wyników, a uwarunkowaniem macierzy.

- Skupiając się na ogólnym przypadku złego uwarunkowania macierzy, można stwierdzić, że tak uwarunkowane macierze zwracają wyniki niestabilne

- gdy dane wejściowe zawierają błędy, na przykład wynikające z zaokrągleń, obliczone wyniki mogą być znacznie zaburzone i obarczone dużym błędem numerycznym. W rezultacie takie dane stają się kolokwialnie "zakłamane", a uzyskane rozwiązania mogą się znacznie od siebie różnić w zależności od przyjętej precyzji obliczeń.

- Przeciwnie natomiast będą zachowywać się macierze dobrze uwarunkowana takie jak macierz A_1 - niewielkie zmiany w danych wejściowych, będą powodować nieznaczne zmiany wyników. Dobrze uwarunkowane macierze są stabilne numerycznie, a ich rozwiązania są mniej podatne na błędy numeryczne, a co za tym idzie otrzymane wyniki będą zbliżone do rzeczywistych i realnych rozwiązań - czyli będą spełniać warunki, na których najbardziej nam zależy w kontekście optymalizacji metod numerycznych.

4 Wnioski

Na podstawie przeprowadzonej dyskusji można sformułować następujące wnioski:

1. Dodanie zaburzenia Δb prowadzi do różnicy w otrzymanych wynikach równań macierzowych. Dla macierzy A_1 są to różnice bardzo niewielkiego rzędu, natomiast dla macierzy A_2 wartości ulegają znacznemu odkształceniu.

2. Dodanie zaburzenia do danych wejściowych pozwala zobrazować pojęcie liczby uwarunkowania macierzy - źle uwarunkowane macierze będą bardzo podatne na perturbacje danych wejściowych i widoczne będą znaczne zmiany danych wyjściowych, natomiast macierze dobrze uwarunkowane będą stablilne numerycznie, a zmiany te będą niewielkie.

3. Optymalizacja metod numerycznych dąży do uzyskiwania wyników jak najmniej obarczonych błędem i jak najbardziej stabilnych, przypominających te analityczne. Zatem decydując się na wybór macierzy, powinno się zwrócić uwagę na jej liczbę uwarunkowania i zdecydować się na taką macierz, dla której ta liczba będzie jak najmniejsza, jeżeli dostępne są inne macierze, które mogą modelować ten sam problem. Ponadto, można próbować przekształcać macierz, tak aby poprawić jej uwarunkowanie, na przykład używając metody skalowania.