Inteligentna Analiza Danych **2019/2020**

Prowadzący: mgr inż. Paweł Tarasiuk środa, 14:30

Daniel Modrzejewski 229963 229963@edu.p.lodz.pl
Mateusz Srebnik 230004 230004@edu.p.lodz.pl

Zadanie 2.: metoda iteracyjna Jacobiego (iteracji prostej)

1. Cel

Celem zadania drugiego jest zaimplementowanie programu mającego na celu rozwiązywania układu N równań liniowych z N niewiadomymi za pomocą metody iteracyjnej Jacobiego (iteracji prostej).

2. Wprowadzenie

Metoda pozwala na obliczenie układu N równań liniowych z N niewiadomymi. Przybliżonym wynikiem układu równań będzie wektor x . Na początku wektor będzie zawierał same 0, z każdą następną iteracją będzie przybliżał się do prawidłowego wyniku. W tym celu korzystamy ze wzoru

$x^{n+1} = Mx^n + NB$

,gdzie

x^{n+1} - wynik iteracji algorytmu

x^n- wynik poprzedniej iteracji algorytmu

A - to macierz współczynników Ax=B

B - to macierz wyników równań

$N - D^{-1}$, gdzie D jest macierzą składającą się z elementów na przekątnej macierzy A

$M = N(L + U)$, gdzie$(L + U)$ to macierz A z zastąpionymi wartościami 0 na przekątnej

możemy zapisać, że

A = L + U + D

3. Opis implementacji

Program został napisany w języku Python, składa się z dwóch klas. Klasa main odpowiada za backendowa część programu , a klasa gui za frontendowa. Klasa main zawiera w sobie funkcje:
- sprawdzająca czy podana macierz ma odpowiedni rozmiar oraz czy ma 0 na przekątnej,
- zmiana kolejności wierszy,
- sprawdzenie czy macierz jest przekątnie dominująca,
- mnożenie macierzy,
- algorytm iteracyjny

Jeśli macierz nie jest zbieżna program nie zatrzymuje się ze względu na chęć pokazania jak będzie się zachowywał program w przypadku nie spełnienia założeń.

4. Materiały i metody

Badania zostały przeprowadzone dla 5 macierzy , trzech przekątnie dominujących oraz 2 nie przekątnie dominujących (nieoznaczonej i sprzecznej).

Pierwszą czynnością do przeprowadzenia badania jest załadowanie macierzy z pliku, jeśli będzie ona poprawna pojawi się w oknie programu, jeśli nie, pojawi się komunikat. Następnie ustawiamy wartość ilości iteracji lub dokładności (to samo miejsce przeznaczone na wpisanie) i wybieramy jeden z przycisków odpowiadający za warunek stopu. Wyniki pojawią się od razu po naciśnięciu przycisku. Badania zostały przeprowadzone z argumentami umieszczonymi pod każdą z macierzy.

Badania:

1. Przykład 1

$$\begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 3 \\ 7 \\ 5 \end{bmatrix}$$

Pierwszy warunek badania to ilość iteracji 5
Drugi warunek badania to dokładność 0.0001

Przykład przede wszystkim mający na celu zmianę kolejności wierszy, tak aby macierz był przekątniowo dominująca.

2. Przykład 2

$$\begin{bmatrix} 0.5 & -0.0625 & 0.1875 & 0.0625 \\ -0.0625 & 0.5 & 0 & 0 \\ 0.1875 & 0 & 0.375 & 0.125 \\ 0.0625 & 0 & 0.125 & 0.25 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 1.5 \\ -1.625 \\ 1 \\ 0.4375 \end{bmatrix}$$

Pierwszy warunek badania to ilość iteracji 10
Drugi warunek badania to dokładność 0.0001

3. Przykład 3

$$\begin{bmatrix} 1 & 0.2 & 0.3 \\ 0.1 & 1 & -0.3 \\ -0.1 & -0.2 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 1.5 \\ 0.8 \\ 0.7 \end{bmatrix}$$

Pierwszy warunek badania to ilość iteracji 10
Drugi warunek badania to dokładność 0.0001

4. Przykład 4

$$\begin{bmatrix} 3 & 3 & 1 \\ 2 & 5 & 7 \\ -4 & -10 & -14 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 1 \\ 20 \\ -40 \end{bmatrix}$$

Pierwszy warunek badania to ilość iteracji 10
Drugi warunek badania to dokładność 0.0001

Przykład przede wszystkim mający na celu sprawdzenie zachowania programu w przypadku układu nieoznaczonego.

5. Przykład 5

$$\begin{bmatrix} 3 & 3 & 1 \\ 2 & 5 & 7 \\ -4 & -10 & -14 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 1 \\ 20 \\ -20 \end{bmatrix}$$

Pierwszy warunek badania to ilość iteracji 10
Drugi warunek badania to dokładność 0.0001

Przykład przede wszystkim mający na celu sprawdzenie zachowania programu w przypadku układu sprzecznego.

5. Wyniki

1. Przykład 1

Tabela 1.Iteracje 5

IT	x_1	x_2	x_3
1	7.0	5.0	3.0
2	7.0	5.0	3.0
3	7.0	5.0	3.0
4	7.0	5.0	3.0
5	7.0	5.0	3.0

Tabela 2.Dokładność 0.0001

IT	x_1	x_2	x_3
1	7.0	5.0	3.0
2	7.0	5.0	3.0

2. Przykład 2

Tabela 3.Iteracje 10

IT	x_1	x_2	x_3	x_4
1	3.0	-3.25	2.6667	1.75
2	1.375	-2.875	0.5833	-0.3333
3	2.4635	-3.0781	2.0903	1.1146
4	1.6921	-2.9421	1.0634	0.089
5	2.2224	-3.0385	1.791	0.7953
6	1.8492	-2.9722	1.2904	0.2989
7	2.1072	-3.0189	1.6424	0.6425
8	1.9264	-2.9866	1.3989	0.402
9	2.0518	-3.0092	1.5695	0.569
10	1.9642	-2.9935	1.4511	0.4523

Tabela 4.Dokładność 0.0001

IT	x_1	x_2	x_3	x_4
1	3.0	-3.25	2.6667	1.75
2	1.375	-2.875	0.5833	-0.3333
3	2.4635	-3.0781	2.0903	1.1146
4	1.6921	-2.9421	1.0634	0.089
5	2.2224	-3.0385	1.791	0.7953
6	1.8492	-2.9722	1.2904	0.2989
7	2.1072	-3.0189	1.6424	0.6425
8	1.9264	-2.9866	1.3989	0.402
9	2.0518	-3.0092	1.5695	0.569
10	1.9642	-2.9935	1.4511	0.4523
...	...	...	...	...
24	1.9998	-3.0	1.4997	0.4997
25	2.0002	-3.0	1.5002	0.5002
26	1.9999	-3.0	1.4998	0.4999
27	2.0001	-3.0	1.5001	0.5001
28	1.9999	-3.0	1.4999	0.4999
29	2.0	-3.0	1.5001	0.5

3. Przykład 3

Tabela 5.Iteracje 10

IT	x_1	x_2	x_3
1	1.5	0.8	0.7
2	1.13	0.86	1.01
3	1.025	0.99	0.985
4	1.0065	0.993	1.0005
5	1.0013	0.9995	0.9992
6	1.0003	0.9996	1.0
7	1.0001	1.0	1.0
8	1.0	1.0	1.0
9	1.0	1.0	1.0
10	1.0	1.0	1.0

Tabela 6.Dokładność 0.0001

IT	x_1	x_2	x_3
1	1.5	0.8	0.7
2	1.13	0.86	1.01
3	1.025	0.99	0.985
4	1.0065	0.993	1.0005
5	1.0013	0.9995	0.9992
6	1.0003	0.9996	1.0
7	1.0001	1.0	1.0

4. Przykład 4

Tabela 7.Iteracje 10

IT	x_1	x_2	x_3
1	0.3333	4.0	2.8571
2	-4.619	-0.1333	-0.0952
3	0.4984	5.981	4.2721
4	-7.0717	-2.1803	-1.5574
5	3.0328	9.009	6.435
6	-10.8206	-6.2221	-4.4443
7	8.0369	14.5503	10.3931
8	-17.6814	-13.7651	-9.8322
9	17.3758	24.8376	17.7412
10	-30.418	-27.788	-19.8486

Tabela 8.Dokładność 0.0001

IT	x_1	x_2	x_3
1	0.3333	4.0	2.8571
2	-4.619	-0.1333	-0.0952
3	0.4984	5.981	4.2721
4	-7.0717	-2.1803	-1.5574
5	3.0328	9.009	6.435
6	-10.8206	-6.2221	-4.4443
7	8.0369	14.5503	10.3931
8	-17.6814	-13.7651	-9.8322
9	17.3758	24.8376	17.7412
10	-30.418	-27.788	-19.8486
...	...	...	...
2289	1.42e+308	1.56e+308	1.12e+308
2290	-inf	-inf	-1.52e+308
2291	nan	nan	inf

5. Przykład 5

Tabela 9.Iteracje 10

IT	x_1	x_2	x_3
1	0.3333	4.0	1.4286
2	-4.1429	1.8667	-1.5238
3	-1.0254	7.7905	1.2789
4	-7.8834	2.6197	-3.8431
5	-1.0053	12.5337	1.8098
6	-12.8036	1.8684	-7.2368
7	0.8772	19.253	3.7522
8	-20.1704	-1.6039	-12.5742
9	6.1286	29.672	8.3372
10	-32.1178	-10.1235	-21.5168

Tabela 10.Dokładność 0.0001

IT	x_1	x_2	x_3
1	0.3333	4.0	1.4286
2	-4.1429	1.8667	-1.5238
3	-1.0254	7.7905	1.2789
4	-7.8834	2.6197	-3.8431
5	-1.0053	12.5337	1.8098
6	-12.8036	1.8684	-7.2368
7	0.8772	19.253	3.7522
8	-20.1704	-1.6039	-12.5742
9	6.1286	29.672	8.3372
10	-32.1178	-10.1235	-21.5168
...	...	...	...
2290	-1.51e+308	-1.66e+308	-1.19e+308
2291	inf	inf	1.62e+308
2292	nan	nan	-inf

6. Dyskusja

W przypadku przykładu pierwszego mamy wynik już przy pierwszej iteracji. Dla algorytmu uruchomionego z warunkiem dokładności mamy dwie iteracje ze względu na potrzebne w algorytmie porównanie poprzedniego wyniku z następnym. W przypadku przykładu drugiego mamy wynik dość dokładny już po 10 iteracjach. Dalsze iteracje pozwalają dojść do dokładności 4 miejsc po przecinku, a powyżej 20 iteracji różnica rozstrzyga się na dalszych miejscach po przecinku i wartości krążą wokół właściwego wyniku. W przykładzie 3 wystarczy 7 iteracji do osiągnięcia dokładnego wyniku W przypadku macierzy nie spełniających warunku zbieżności, nieoznaczonych lub sprzecznych wyniki dążą do nieskończoności lub minus nieskończoności.

7. Wnioski

Im większa ilość iteracji tym większa dokładność. Każda następna iteracja algorytmu bazuje na wyniku poprzedniej. Algorytm sprawdza się w przypadku macierzy zbieżnych, w przypadku macierzy sprzecznych lub nieoznaczonych wynik będzie dążył do nieskończoności.Póki algorytm nie osiągnie zadanej dokładności wyniki wektora będą przybliżone do właściwego wyniku.

Literatura

[1] T. Oetiker, H. Partl, I. Hyna, E. Schlegl. *Nie za krótkie wprowadzenie do systemu LaTeX2e*, 2007, dostępny online. `https://ctan.org/tex-archive/info/lshort/polish/lshort2e.pdf`.

https://college.cengage.com/mathematics/larson/elementary$_{l}inear/5e/students/ch08-10/chap_{1}0_{2}.pdf$
$http : //www.algorytm.org/procedury - numeryczne/metoda - jacobiego.html$