Vadym Semkovych

296669

11.03.2020

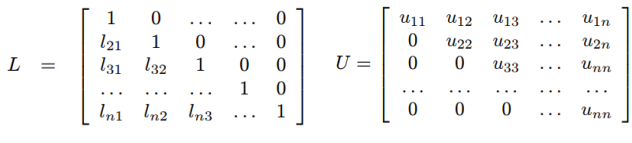
Sprawozdanie 2

Odwracanie macierzy, obliczanie wyznacznika i wskaźnika uwarunkowania macierzy przy użyciu rozkładu LU

1. **Wstęp teoretyczny**

**Metoda LU** - metoda, służąca do rozwiązywania układu równań liniowych. Rozkład LU polega na podziale macierzy na macierz (dolna macierz trójkątna z jedynkami na diagonali) i macierz (górna macierz trójkątna z niezerowymi elementami na diagonali).

gdzie:



Rozwiązywanie układu równań polega na tym, że sprowadzamy:



(gdzie **A –** macierz współczynników, – wektor (macierz ) zmiennych , - wektor (macierz ) wyrazów wolnych.)

do postaci:



Dzięki właściwościom macierzy trójkątnych taki układ łatwo da się rozwiązać.

Metoda LU pozwala na obliczanie wyznacznika również. Wzór podany poniżej:



przy czym = 1, a równy jest iloczynowi elementów stojących na diagonali tej macierzy.

Wskaźnik uwarunkowania macierzy można obliczyć za pomocą iloczynu normy macierzy **A** i normy macierzy .

Wskaźnik uwarunkowania: .

1. **Problem**

Podczas laboratorium analizowaliśmy macierz za pomocą rozkładu LU. Elementy macierzy zdefiniowane są następująco:

gdzie:

***i, j = 1,2 ….. n, ,*** ponieważ korzystamy z biblioteki NR.

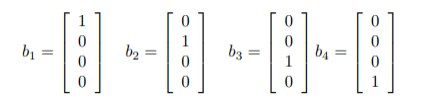
Dalej dokonaliśmy rozkładu LU, zastępując macierz **A** na macierz **L** oraz **U**. Zrobiliśmy to korzystając z funkcji biblioteki NR:

ludcmp(float A[n][n], int n, int indx[n], float &d),

gdzie: A - macierz, n - rozmiar macierzy, indx - wektor permutacji wierszy, d - określa liczbę permutacji.

Następnym krokiem było obliczenie wyznacznika macierzy. Funkcja ludcmp w wyniku podaje nam macierz LU, więc aby obliczyć wyznacznik wystarczyło wymnożyć elementy stojące na diagonali przez zmienną d.

Kolejne zadanie, to wyliczanie macierzy odwrotnej . Znaleźliśmy macierz odwrotną rozwiązując n układów równań z wektorami wyrazów wolnych:



Tutaj znowu korzystamy z funkcji biblioteki NR:

lubksb(float LU[n][n],int n, int indx[n], float b[n]),

gdzie: LU - to rozkład LU (wpisany do macierzy A), b - aktualny wektor wyrazów wolnych

Następnie szukaliśmy iloczyn **,** żeby sprawdzić skuteczność metody LU (w najlepszym wypadku powinniśmy dostać macierz jednostkową). Skorzystaliśmy tutaj z funkcji mnożenia macierzy:

for (int i = 1; i <= 4; i++) {

for (int j = 1; j <= 4; j++) {

C[i][j] = 0;

for(int k = 1; k <= 4; k++)

C[i][j] += A[i][k] \* B[k][j];

}

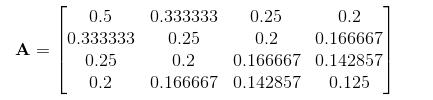
}

(pętle startują od 1, ponieważ korzystamy z biblioteki NR).

Ostatnie zadanie, to obliczanie wskaźnika uwarunkowania, obliczyliśmy go korzystając ze wzoru wymienionego we wstępie teoretycznym. W naszym przypadku korzystamy z normy maksymalnej:

1. **Wyniki**

Macierz wejściowa **A:**



1. **Elementy diagonalne macierzy U:**

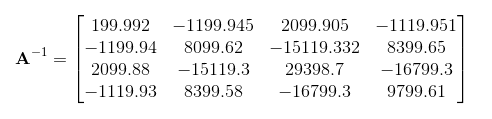
|  |  |
| --- | --- |
|  | 0.2 |
|  | −0.0833333 |
|  | −0.00238095 |
|  | **-**0.00005953 |

*Tabela 1. Wartości diagonali macierzy U*

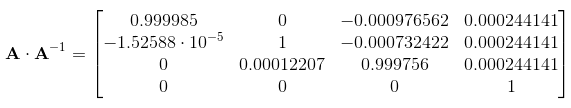
1. **Wyznacznik macierzy A:**

Korzystając z elementów zapisanych w Tabeli 1 obliczyliśmy wyznacznik macierzy za pomocą wzoru zdefiniowanego we wstępie teoretycznym, wynosi on:

1. **Macierz odwrotna :**



1. **Iloczyn :**



1. **Wskaźnik uwarunkowania:**
2. **Wnioski**

Korzystając z metody LU odwróciliśmy macierz **A**. Także obliczyliśmy jej wyznacznik oraz wskaźnik uwarunkowania. Aby sprawdzić skuteczność metody znaleźliśmy iloczyn , oczekiwaliśmy w wyniku dostać macierz jednostkową, ale jak widzimy elementy na diagonali są jedynkami ,gdyż elementy leżące nie na przekątnej są przybliżone do zera, ale niezerowe. Wskaźnik uwarunkowania jest duży, świadczy to o tym że macierz **A** jest źle uwarunkowana.

Metoda LU nie gwarantuje dokładności wyników, jednak za pomocą tej metody można szybko wyliczyć wyznacznik oraz odwrócić macierz.