

Rozwiązywanie układu równań liniowych $Ax = b$ metodą blokową Gaussa Seidla Projekt I MN

Gabriela Majstrak

08.01.2024

1. Opis metody

Metoda Gaussa-Seidla jest metodą iteracyjną wyznaczania rozwiązania równania $Ax = b$. W metodzie iteracyjnej startujemy z przybliżenia początkowego $x^{(0)}$ i tworzymy ciąg kolejnych przybliżeń.

$$x^{(0)} = \begin{bmatrix} x_1^{(0)} \\ \vdots \\ x_n^{(0)} \end{bmatrix}$$

W naszym przypadku zajmujemy się blokową metodą Gaussa-Seidla. Macierz A o wymiarach $n \times n$ jest symetryczna i ma następującą postać blokową.

$$A = \begin{bmatrix} A_{11} & A_{12} & A_{13} \\ A_{12}^T & -A_{11} & A_{23} \\ A_{13}^T & A_{23}^T & A_{11} \end{bmatrix}$$

Gdzie każda z macierzy A_{ii} ma wymiar $p \times p$, $n = 3p$.

Aby rozwiązać równanie $Ax = b$ metodą blokową Gaussa Seidla początkowy wektor przybliżeń $x^{(0)} \in R^n$ dzielimy na 3 podwektory $X_1^{(0)}, X_2^{(0)}, X_3^{(0)} \in R^p$.

$$X_1^{(0)} = \begin{bmatrix} x_1^{(0)} \\ \vdots \\ x_p^{(0)} \end{bmatrix} \quad X_2^{(0)} = \begin{bmatrix} x_{p+1}^{(0)} \\ \vdots \\ x_{2p}^{(0)} \end{bmatrix} \quad X_3^{(0)} = \begin{bmatrix} x_{2p+1}^{(0)} \\ \vdots \\ x_n^{(0)} \end{bmatrix}$$

Analogicznie postępujemy z wektorem b wyrazów wolnych, dzielimy b na $B_1, B_2, B_3 \in R^p$. Następnie iteracyjnie wyznaczamy ciągi przybliżeń każdego z tych podwektorów wykorzystując poniższe równania, które dostajemy z mnożenia macierzowego Ax .

$$A_{11}X_1^{k+1} = B_1 - A_{12}X_2^k - A_{13}X_3^k \quad (1)$$

Mając wyznaczone X_1^{k+1} możemy z tego przybliżenia skorzystać wyznaczając X_2^{k+1} . Wzór iteracyjny dla wyznaczenia X_2^{k+1} będzie więc wyglądał następująco:

$$A_{11}X_2^{k+1} = A_{12}^T X_1^{k+1} + A_{23}X_3^k - B_2 \quad (2)$$

Teraz mając wyznaczone zarówno X_1^{k+1} jak i X_2^{k+1} , możemy skorzystać z tych przybliżeń przy wyznaczaniu X_3^{k+1} . Wzór iteracyjny dla X_3^{k+1} :

$$A_{11}X_3^{k+1} = B_3 - A_{13}^T X_1^{k+1} - A_{23}^T X_2^{k+1} \quad (3)$$

Równania (1), (2) i (3) rozwiązujemy metodą eliminacji Gaussa z częściowym wyborem elementu głównego (GEPP)

2. Opis programu

2.1 Funkcja createMatrix

Ta funkcja tworzy macierz o rozmiarze $p \times p$ w zależności od wartości klucza **key**. Oto krótki opis działań funkcji:

- Jeśli klucz **key** to **'random'**, funkcja generuje macierz losową o wymiarach $p \times p$ przy użyciu funkcji **rand(p)**.
- Jeśli klucz **key** to **'magic'**, funkcja generuje macierz magiczną o wymiarach $p \times p$ przy użyciu funkcji **magic(p)**.
- Jeśli klucz **key** to **'other'**, funkcja losowo wybiera wartości z zakresu od 1 do 1000, permutuje je i tworzy macierz $p \times p$ przy użyciu tych wartości.
- Jeśli klucz **key** to **'custom'**, funkcja prosi użytkownika o wprowadzenie wartości macierzy, oddzielonych przecinkami. Następnie przekształca te wartości do macierzy $p \times p$, pod warunkiem, że liczba wprowadzonych wartości jest równa p^2 .

Jeśli użytkownik poda nieznany klucz, zostanie wygenerowany błąd, a użytkownikowi zostanie podana lista dostępnych kluczy.

2.2 Funkcja createblockMatrix

Funkcja **createblockmatrix** tworzy macierz blokową o wymiarach $3p \times 3p$, gdzie poszczególne bloki zależą od macierzy **C**, generowanej przez funkcję **createMatrix**. Macierz tworzona przez tą funkcję ma podaną już wyżej postać:

$$A = \begin{bmatrix} A_{11} & A_{12} & A_{13} \\ A_{12}^T & -A_{11} & A_{23} \\ A_{13}^T & A_{23}^T & A_{11} \end{bmatrix}$$

2.3 Funkcja GEPP

Funkcja GEPP, to funkcja napisana w celu rozwiązywania równań $p \times p$ (1), (2) i (3) czyli wyznaczania kolejnych przybliżeń $X_1^{(k)}$, $X_2^{(k)}$, $X_3^{(k)}$. Rozwiązywanie tych równań opiera się o metodę eliminacji Gaussa z częściowym wyborem elementu głównego

2.4 Funkcja Gauss-Seidel

Funkcja służy do blokowego wyznaczania rozwiązania układu $AX = B$, gdzie macierz współczynników A jest tryblokowa (bloki rozmiaru $p \times p$). Algorytm wykorzystuje metodę Gaussa z częściowym wyborem elementu głównego do iteracyjnego rozwiązania mniejszych układów równań związanych z poszczególnymi blokami macierzy A .

Opis

1. Funkcja, w pętli iteracyjnej, korzysta z metody GEPP do rozwiązywania układów równań związanych z kolejnymi blokami macierzy A .
2. Proces iteracyjny kontynuowany jest, dopóki norma różnicy między kolejnymi przybliżeniami rozwiązania nie osiągnie zadanego poziomu precyzji (10^{-10}).
3. Funkcja zwraca wektor rozwiązania X oraz informację o liczbie wykonanych iteracji.

W przypadku, gdy macierz nie jest diagonalnie dominująca na którymś z wierszy, funkcja wypisuje stosowne ostrzeżenie. Istnieje również ograniczenie na maksymalną liczbę iteracji (300), aby zapobiec potencjalnym problemom z nieskończoną pętlą.

3. Ciekawe przykłady, analiza ich wyników i błędów numerycznych

W skrypcie testującym udało mi się odkryć parę ciekawych przypadków działania funkcji Gauss-Seidel, a co za tym idzie ustalić kiedy metoda będzie zbieżna. Warto dodać, że w każdym przykładzie wektor wyrazów wolnych generowałam losowo.

3.1. macierz w której bloki są generowane losowo

Aby macierz była zbieżna dla bloków generowanych losowo, musielibyśmy mieć duże szczęście przy jej generowaniu. Jest tak, w związku z wymaganą pewną strukturą takich blokowych macierzy, którą określimy, pokazując kolejne przykłady.

3.2. macierz, której bloki są macierzami magicznymi

Podobnie jak w przykładzie wyżej, metoda nie będzie dla takich macierzy zbieżna.

3.3. macierze silnie diagonalnie dominująca, w której każdy blok jest również silnie diagonalnie dominujący

Macierz silnie diagonalnie dominująca to taka macierz, w której elementy na przekątnej co do modułu są większe niż suma bezwzględna pozostałych elementów w ich odpowiednich wierszach. Jeśli każdy blok tej macierzy również jest silnie diagonalnie dominujący, oznacza to cała macierz również.

W przypadku metod numerycznych, takich jak metoda Gaussa-Seidela, macierze silnie diagonalnie dominujące z silnie dominującymi blokami mają tendencję do szybszej zbieżności. W testowanych przykładach udało się zauważyć że im silniejsza diagonalna dominacja w każdym bloku, tym szybciej metoda iteracyjna będzie zbliżać się do rozwiązania układu równań.

Dla następującej macierzy silnie diagonalnie dominującej z każdym blokiem również dominującym porównania czasów wykonania były podobne

$$A = \begin{bmatrix} 40 & -1 & 9 & -5 & 3 & 1 \\ 2 & 30 & 8 & 6 & 2 & 4 \\ 9 & 8 & -40 & 1 & -7 & 2 \\ -5 & 6 & -2 & -30 & 5 & 6 \\ 3 & 2 & -7 & 5 & 40 & -1 \\ 1 & 4 & 2 & 6 & 2 & 30 \end{bmatrix}$$

Porównanie czasów wykonania:

Metoda Gaussa-Seidela: 0.000994 sekund

Funkcja wbudowana: 0.000102 sekund

Porównanie rozwiązań:

Błąd bezwzględny: 2.062621e-12

Błąd względny: 4.789494e-11

Rysunek 1: błędy bezwzględne i względne między funkcją wbudowaną i metodą Gaussa Seidela

$$A = \begin{bmatrix} 200 & -1 & 9 & -5 & 3 & 1 \\ 2 & 100 & 8 & 6 & 2 & 4 \\ 9 & 8 & -200 & 1 & -7 & 2 \\ -5 & 6 & -2 & -100 & 5 & 6 \\ 3 & 2 & -7 & 5 & 200 & -1 \\ 1 & 4 & 2 & 6 & 2 & 100 \end{bmatrix}$$

Porównanie czasów wykonania:

Metoda Gaussa-Seidela: 0.000635 sekund

Funkcja wbudowana: 0.000081 sekund

Porównanie rozwiązań:

Błąd bezwzględny: 4.012139e-14

Błąd względny: 3.270414e-12

Rysunek 2: błędy bezwzględne i względne między funkcją wbudowaną i metodą Gaussa Seidela

Im silniejsza dominacja, tym metoda szybciej zbieżna i generuje mniejsze błędy.

3.4. Macierz silnie diagonalnie dominująca, w której tylko bloki macierzy leżące na przekątnej są silnie diagonalnie dominujące

Metoda Gaussa-Seidela może efektywnie działać, nawet gdy tylko bloki na przekątnej macierzy blokowej są diagonalnie dominujące, a pozostałe bloki nie spełniają tego warunku. W takim przypadku, jeśli główna macierz A pozostaje diagonalnie dominująca, to metoda iteracyjna może nadal zbiegać do rozwiązania układu równań.

$$A = \begin{bmatrix} 2000 & -1 & 0 & 1 & 0 & 1 \\ 2 & 1000 & 1 & 0 & 1 & 0 \\ 0 & 1 & -2000 & 1 & 0 & 1 \\ 1 & 0 & -2 & -1000 & 1 & 0 \\ 0 & 1 & 0 & 1 & 2000 & -1 \\ 1 & 0 & 1 & 0 & 2 & 1000 \end{bmatrix}$$

Dla tej macierzy metoda jest zbieżna mimo tego, że bloki silnie diagonalnie dominujące są tylko na przekątnej.

```

Porównanie czasów wykonania:
Metoda Gaussa-Seidela: 0.000643 sekund
Funkcja wbudowana: 0.000077 sekund

Porównanie rozwiązań:
Błąd bezwzględny: 1.842701e-16
Błąd względny: 1.837047e-13

```

Rysunek 3: błędy bezwzględne i względne między funkcją wbudowaną i metodą Gaussa Seidela

3.5. Bloki na przekątnej dominujące diagonalnie, pozostałe bloki nie oraz cała macierz nie

W przypadku, gdy tylko bloki na przekątnej macierzy są diagonalnie dominujące, a pozostałe bloki oraz cała macierz nie spełniają tego warunku, metoda Gaussa-Seidela może się rozbiegać. Przykład takiej macierzy (ze skryptu testującego)

$$A = \begin{bmatrix} 200 & -1 & 0 & 1000 & 0 & 1000 \\ 2 & 100 & 1000 & 0 & 1000 & 0 \\ 0 & 1000 & -200 & 1000 & 0 & 1000 \\ 1000 & 0 & -2 & -100 & 1000 & 0 \\ 0 & 1000 & 0 & 1000 & 200 & -1 \\ 1000 & 0 & 1000 & 0 & 2 & 100 \end{bmatrix}$$

3.6. Zbieżność metody, gdy macierz A nie jest silnie diagonalnie dominująca

Niekiedy metoda Gaussa-Seidela może zbiegać nawet w sytuacji, gdy warunek silnej diagonalnej dominacji nie jest spełniony dla całej macierzy. Wystarczy, że bloki na przekątnej nie są zerowe, a cała reszta bloków już tak. Mimo że warunek silnej diagonalnej dominacji nie jest spełniony globalnie, specyficzna struktura bloków może wpłynąć na efektywność iteracyjnego rozwiązania układu równań.

$$A = \begin{bmatrix} 1 & 2 & 3 & 0 & 0 & 0 & 0 & 0 & 0 \\ 2 & -1 & 4 & 0 & 0 & 0 & 0 & 0 & 0 \\ 3 & 4 & 5 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & -2 & -3 & 0 & 0 & 0 \\ 0 & 0 & 0 & -2 & 1 & -4 & 0 & 0 & 0 \\ 0 & 0 & 0 & -3 & -4 & -5 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 2 & 3 \\ 0 & 0 & 0 & 0 & 0 & 0 & 2 & -1 & 4 \\ 0 & 0 & 0 & 0 & 0 & 0 & 3 & 4 & 5 \end{bmatrix}$$

Porównanie czasów wykonania:

Metoda Gaussa-Seidela: 0.000472 sekund

Funkcja wbudowana: 0.000077 sekund

Porównanie rozwiązań:

Błąd bezwzględny: 1.872959e-16

Błąd względny: 1.392354e-16

Rysunek 4: błędy bezwzględne i względne między funkcją wbudowaną i metodą Gaussa Seidela

3.7. Analiza wyników

- Macierze generowane losowo:
 - Metoda Gaussa-Seidela nie jest zbieżna dla macierzy, w których bloki są generowane losowo.
 - Wymaga to specyficznej struktury blokowej, co sprawia, że zbieżność jest trudna do osiągnięcia.
- Macierze magiczne:
 - Metoda również nie jest zbieżna dla macierzy, które składają się z bloków będących macierzami magicznymi.
 - Brak struktury i specyficzne wartości elementów sprawiają, że metoda nie radzi sobie efektywnie.
- Macierze silnie diagonalnie dominujące z dominującymi blokami:
 - Macierze silnie diagonalnie dominujące z dominującymi blokami skutkują zbieżnością metody Gaussa-Seidela.
 - Im silniejsza diagonalna dominacja w macierzy A , tym szybsza jest metoda iteracyjna.
- Macierze, gdzie tylko bloki na przekątnej są silnie diagonalnie dominujące, ale też cała macierz A jest silnie diagonalnie dominująca:

- Metoda Gaussa-Seidela może być zbieżna, nawet gdy tylko bloki na przekątnej macierzy są diagonalnie dominujące.
- Struktura bloków wpływa na zbieżność, a elastyczność metody pozwala na skuteczne radzenie sobie z różnorodnymi warunkami.
- Macierze z blokami silnie dominującymi na przekątnej, reszta nie i cała macierz również nie:
 - Metoda nie jest zbieżna, gdy tylko bloki na przekątnej są dominujące, a reszta bloków i cała macierz nie spełniają tego warunku.
- Macierze niekoniecznie diagonalnie dominujące, gdzie bloki na przekątnej są niezerowe, reszta zerowa:
 - Metoda jest zbieżna w przypadku, gdy bloki na przekątnej są niezerowe, a pozostałe bloki to macierze zerowe.
 - Specyficzna struktura bloków wpływa na zbieżność, nawet gdy globalna diagonalna dominacja nie jest spełniona.

3.8. Analiza błędów numeryczny

Na podstawie powyżej przedstawionych przykładów, można stwierdzić, że przy blokach silnie diagonalnie dominujących błędy będą mniejsze im silniej macierz A i każdy blok będzie dominujący. Błędy wychodzą też bardzo małe dla macierzy, które bloki poza tymi na przekątnej są zerowe

4. Zastosowania metody

Metoda Gaussa-Seidela znajduje praktyczne zastosowanie w różnych obszarach, zwłaszcza w Machine Learning i Data Science. Poniżej kilka przykładów:

Rozwiązywanie układów równań liniowych w Machine Learning

Metoda Gaussa-Seidela jest używana do iteracyjnego rozwiązywania układów równań, co jest przydatne w algorytmach Machine Learning, na przykład w regresji liniowej.

Algorytmy faktoryzacji i analiza składników

Wykorzystywana w algorytmach faktoryzacji macierzy, co ma zastosowanie w analizie składników głównych (PCA) czy algorytmach rekomendacyjnych.

Optymalizacja funkcji kosztu w trenowaniu modeli

Stosowana do iteracyjnej optymalizacji parametrów modelu w procesie trenowania, zwłaszcza w metodach gradientowych.

Rozwiązywanie równań różniczkowych

Używana w numerycznych metodach rozwiązywania równań różniczkowych, zwłaszcza w przypadku równań eliptycznych.

Algorytmy iteracyjne w analizie danych

Metoda Gaussa-Seidela znajduje zastosowanie w iteracyjnych algorytmach analizy danych, takich jak klastrowanie czy algorytmy grupowania.

Ta elastyczna metoda wymaga jednak uwzględnienia właściwości macierzy i monitorowania zbieżności. W praktyce może być skuteczna, zwłaszcza w przypadku dużych zbiorów danych, gdzie metody iteracyjne mogą być bardziej efektywne niż analityczne.