Metody numeryczne

Program wykładu

1. Dokładność
2. Układy równań liniowych
3. Równania nieliniowe – metody iteracyjne
4. Interpolacja
5. Aproksymacja
6. Różniczkowanie
7. Metody rozwiązywania warunków brzegowych równań różniczkowych zwyczajnych
8. Całkowanie
9. Miejsca zerowe wielomianów
10. Generatory liczb pseudolosowych i metody ich testowania
11. Metoda Monte Carlo
12. Metody geometrii obliczeniowej

Literatura

1. Kordecki W., Selwat K., Metody numeryczne dla informatyków, Helion, 2020.
2. Mikołajczak P., Ważny M., Metody numeryczne w C++, Instytut Informatyki UMCS, 2012.
3. Majchrzak E., Michnacki B., Metody numeryczne, Podstawy teoretyczne, aspekty praktyczne i algorytmy, WPŚ, Gliwice 2004
4. Burden R.L., Faires.D., Numerical Analysis. PWS-KENT Publishing Company, Boston 1984.
5. Tatjewski P., Metody numeryczne. OWPW, Warszawa 2013.
6. Dahlquist G., Bjorck L., Metody numeryczne. PWN, Warszawa 1983.
7. Fortuna Z., Macukow B., Wąsowski J., Metody numeryczne. WN-T, Warszawa 1982.
8. Harel D., Rzecz o istocie informatyki, algorytmika. WN-T, Warszawa 1992.
9. Kiełbasiński A., Schwetlick H., Numeryczna algebra liniowa. WN-T, Warszawa 1992.
10. Ralston A., Wstęp do analizy numerycznej. PWN, Warszawa 1983.
11. Stoer J., Bulirsch R., Wstęp do analizy numerycznej. PWN, Warszawa 1987.

Metody numeryczne

Metody numeryczne są działem matematyki stosowanej. Zajmują się badaniem sposobów umożliwiających rozwiązywanie zadań matematycznych za pomocą działań arytmetycznych.

Metody numeryczne:

- wykorzystywane środki - analiza matematyczna, algebra (dowody matematyczne dowodzą poprawności metod),

- nowoczesne metody numeryczne - znajomość języków programowania i odpowiednich bibliotek,
- jedno z podstawowych narzędzi pracy inżyniera,
- służą do analiz i symulacji problemów z zakresu fizyki, mechaniki, elektrotechniki, medycyny, ekonomii i wielu innych,
- praktyczne ich zastosowanie sprowadza się najczęściej do wykonania ściśle określonego algorytmu w postaci skończonej liczby działań arytmetycznych oraz logicznych (często również opracowania indywidualnej metody dla bardziej złożonego zagadnienia),
- korzystając z metod numerycznych możemy otrzymać rozwiązanie dokładne lub przybliżone.

Narzędzia

Lista narzędzi wykorzystywanych w analizie numerycznej:

https://en.wikipedia.org/wiki/List_of_numerical-analysis_software (https://en.wikipedia.org/wiki/List_of_numerical-analysis_software)

Porównanie narzędzi:

https://en.wikipedia.org/wiki/Comparison_of_numerical-analysis_software (https://en.wikipedia.org/wiki/Comparison_of_numerical-analysis_software)

Najważniejsze pojęcia

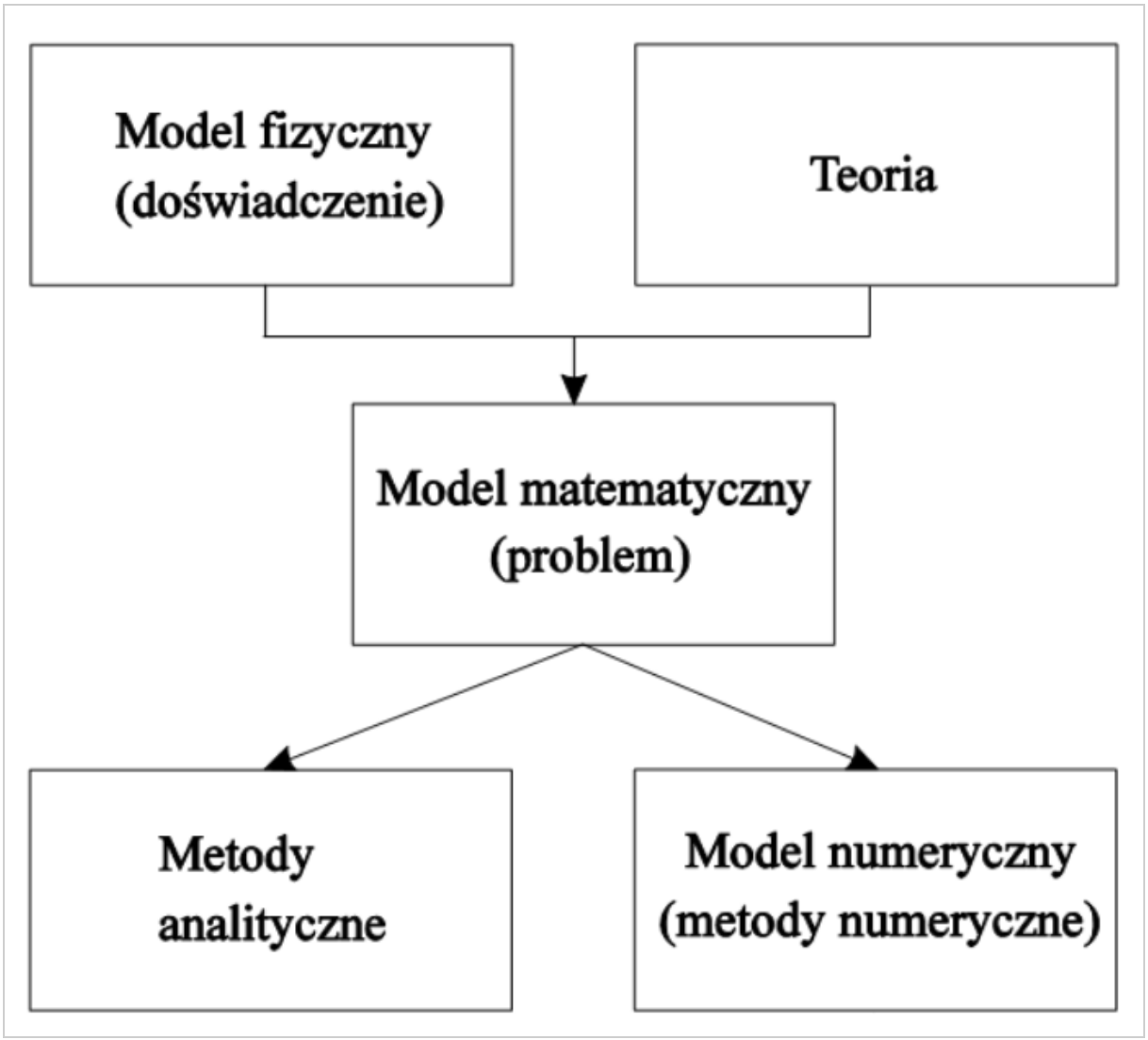
Dokładność (miara dokładności) - dopuszczalny błąd wyniku, ustalany najczęściej arbitralnie przez numeryka na podstawie jego doświadczenia.

Zbieżność iteracji - stopniowe zbliżanie się do wartości dokładnej (poszukiwanej) w procesie iteracji. Proces iteracyjnego poszukiwania wartości kończy się, gdy wartość ta zostanie osiągnięta z zadaną dokładnością.

Dyskretyzacja - zastąpienie rozważanego obiektu, składającego się z nieskończonej liczby punktów, obiektem równoważnym rozpatrywanemu, ale składającym się ze skończonej liczby punktów charakterystycznych, zwanych węzłami. Obiekt ciągły zastąpiony zostaje obiektem nieciągłym.

Model - reprezentacja badanego obiektu w postaci innej, niż ta, w której występuje on w rzeczywistości, w nauce model jest rozumiany jako celowo uproszczona reprezentacja rzeczywistości.

Model matematyczny (problem) - reprezentacja istniejącego lub hipotetycznego fragmentu rzeczywistości, tworzona w określonym celu, z wykorzystaniem skończonego zbioru symboli i operatorów matematycznych, z którymi związane są ścisłe zasady posługiwania się nimi. Model matematyczny pozbawiony jest szczegółów i cech nieistotnych dla osiągnięcia postawionego celu. Symbole i operatory matematyczne zawarte w modelu mają interpretację odnoszącą je do konkretnych elementów modelowanego fragmentu rzeczywistości.



Wykład 1. Dokładność

Plan

1. Błędy obliczeń
2. Typy całkowite i zmiennopozycyjne
3. Uwarunkowanie zadania
4. Algorytm i jego numeryczne realizacje
5. Stabilność i niestabilność numeryczna

Błędy obliczeń

Błąd jest różnicą między wartością dokładną i wartością przybliżoną.

Błąd bezwzględny Δ liczby x - wartość bezwzględna różnicy pomiędzy liczbą dokładną x liczbą przybliżoną $\bar{x}$:

$$\Delta = |x - \bar{x}|$$

Jeśli wartość dokładna nie jest znana, to zamiast błędu bezwględnego oblicza się kres górny (oszacowanie od góry) błędu bezwględnego.

Błąd wzgledny δ liczby x - stosunek błędu bezwzględnego Δ tej liczby do wartości bezwzględnej liczby dokładnej x ($x \neq 0$):

$$\delta = \frac{\Delta}{|x|} = \frac{|x - \bar{x}|}{|x|}$$

Zródła błedów w obliczeniach numerycznych moga miec różny charakter.

Moga to byc zwłaszcza:

- błędy zaookrągleń wynikające z arytmetyki komputera,
- błedy obciecia (np. konieczność zakończenia obliczeń na pewnym etapie dla procesów nieskończonych)
- błedy programisty
- błedy danych wejsciowych (np. dane zaokraglone z wczesniejszych obliczen, dane z pomiarów, stałe fizyczne),
- błędy modelu (np. zbytnie uproszczenie modelu matematycznego),
- błędy metody (np. duże błędy dla pewnego rodzaju danych wejściowych).

Błędy zaokrągleń

W sposób ścisły można reprezentowaś w komputerze tylko liczby całkowite (z pewnego zakresu) oraz liczby wymierne, posiadające skończone rozwinięcia binarne (z pewnego zakresu). Wszystkie inne liczby można reprezentoważ tylko w sposób przybliżony. Są one zatem obarczone pewnym błędem, zwanym błędem zaokrąglenia. Zaokrągleniem liczby nazywamy zatem odrzucenie z niej wszystkich cyfr, poczynając od pewnego miejsca.

Rozwinięciem binarnym liczby $\frac{1}{5}$ jest rozwinięcie nieskończone, okresowe $0.(0011)_2$. Analogicznie $\frac{1}{10} = \frac{1}{2} \cdot \frac{1}{5} = 0.0(0011)_2$. Zatem, ułamki typu $\frac{1}{10}$, $\frac{1}{100}$ itd nie mogą być dokładnie reprezentowane w pamięci. Komputer zapamiętuje tylko ich skończone przybliżenia, obarczone błędem zaokrąglenia.

Cyfry znaczące

Niech x będzie liczbą rzeczywistą, mającą ogólnie nieskończone rozwinięcie dziesiętne. Cyfry tego rozwinięcia numerujemy w następujący sposób: cyfra jedności ma numer zero, cyfra dziesiątek ma numer jeden, cyfra setek ma numer dwa itd. Cyfry części ułamkowej rozwinięcia dziesiętnego mają numery ujemne.

Liczba x jest poprawnie zaokrąglona na d-ej pozycji do liczby, którą oznaczamy $x^{(d)}$, jeśli błąd zaokrąglenia ϵ jest taki, że:

$$\epsilon = |x - x^{(d)}| \leq \frac{1}{2} \cdot 10^d$$

Jeśli $\bar{x}$ jest przybliżeniem dokładnej wartości x, to k-tą cyfrę dziesiętną liczby x nazywamy **znaczącą**, jeśli :

$$|x - \bar{x}| \leq \frac{1}{2} \cdot 10^k$$

Wynika stąd, że każda cyfra poprawnie zaaokrąglonej liczby, począwszy od pierwszej cyfry różnej od 0, jest znacząca. Liczba cyfr znaczących jest pewną miarą błędu zaokrąglenia. Jeśli zatem, w wyniku obliczeń otrzymamy pewną wielkość $\bar{x}$ z dokładnością do ϵ, to możemy jedynie stwierdzić, że dokładna wartość leży w przedziale $[\bar{x} - \frac{\epsilon}{2}, \bar{x} + \frac{\epsilon}{2}]$.

Typy całkowite

Dowolną liczbę całkowitą x można przedstawić w postaci rozwinięcia dwójkowego:

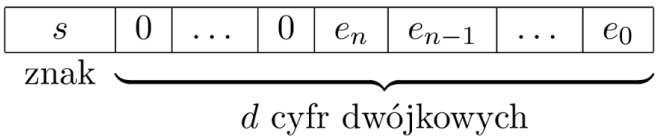
$$x = s \sum_{i=0}^{n} e_i \times 2^i$$

gdzie

s - znak liczby ($a = 1$ lub $s = -1$)

$e_i = 0$ lub $e_i = 1$

Jeżeli do zapisu liczby wykorzystujemy $d + 1$ bitów, to gdy $n < d$ — liczba x może być reprezentowana w wybranej arytmetyce i zapisana jako:



W ten sposób mogą być reprezentowane liczby z zakresu $[\ -2^d,\ 2^d\ -\ 1]$. Liczby całkowite w zależności od d mogą przyjmować wartości z różnych zakresów.

W zasadzie wszystkie obliczenia na liczbach całkowitych dokonywane są dokładnie. Wyjątki od tej reguły są dwa:

- Operacja dzielenia zazwyczaj nie wyprowadza poza typ całkowity. Jest to dzielenie całkowitoliczbowe („z resztą”).
- Wynik działania wykracza poza dopuszczalny zakres; podawany jest wówczas modulo $2d$

Typy zmiennopozycyjne

Struktura liczby zmiennopozycyjnej (zmiennoprzecinkowej) jest następująca:

$$x = m \cdot p^c,$$

gdzie:

m -mantysa,

c - cecha (wykładnik),

p - podstawa systemu (np. 2, 10).

W tym zapisie liczba rzeczywista jest przedstawiona za pomocą dwóch grup bitów: mantysy (części ułamkowej) oraz wykładnika (cechy, liczby całkowitej).

W realizacji komputerowej wykładnik (cecha) liczby x zapisywana jest na $d - t$ bitach, pozostałe t bitów przeznaczonych jest na reprezentację mantysy m. Zatem zamiast (na ogół nieskończonego) rozwinięcia mantysy:

$$m = \sum_{i=1}^{\infty} e_{-i} \times 2^{-i}$$

$e_{-1} = 1$, $e_i = 0$ lub $e_i = 1$ dla $i > 1$

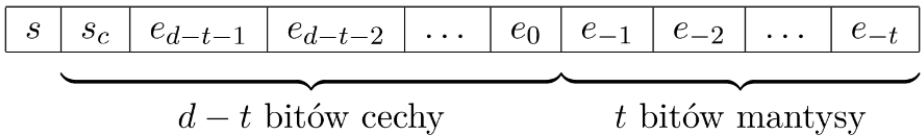
korzystamy (zakładając, że mantysa została prawidłowo zaaokrąglona do t cyfr

$$m_t = \sum_{i=1}^{t} e_{-i} \times 2^{-i}$$

wówczas

$$|m - m_t| \leq \frac{1}{2} \cdot 2^{-t}$$

Liczba x może być zapisana w następujący sposób:



Oznaczmy przez $rd(x)$ reprezentację zmiennopozycyjną (maszynowe przybliżenie) liczby x, czyli $rd(x) = s \cdot 2^c \cdot m_t$.

Jeśli $x \neq 0$ spełniona jest następująca nierówność:

$$\left| \frac{rd(x) - x}{x} \right| \leq 2^{-t}$$

czyli

$$rd(x) = x(1 + \epsilon), \quad \text{gdzie } |\epsilon| \leq 2^{-t} = \text{eps}$$

Liczby rzeczywiste reprezentowane są, na ogół, niedokładnie. Błąd względny ϵ jest nie większy

od 2^{-t} . We współczesnych komputerach t przyjmuje wartości: 24 dla liczb typu float (32-bitowych) lub 53 (double; 64 bity).

Liczba cyfr mantysy decyduje o dokładności liczb rzeczywistych, a liczba cyfr cechy — o ich zakresie. Cecha $w \in [w_{min},\ w_{max}]$, gdzie $w_{min} = -w_{max} - 1 = 2^{d-t-1}$.

Arytmetyka zmiennopozycyjna

Cztery elementarne działania arytmetyczne: dodawanie (+), odejmowanie (−), mnożenie (·), dzielenie (/). Wynikiem działań na liczbach maszynowych jest na ogół liczba maszynowa. Przez „fl" oznaczymy wynik działania zmiennopozycyjnego, wtedy wyniki elementarnych działań arytmetycznych możemy zapisać w postaci:

- $fl(x \pm y) = (x \pm y) \cdot (1 + \epsilon)$,
- $fl(x \cdot y) = (x \cdot y) \cdot (1 + \epsilon)$,
- $fl(x/y) = (x/y) \cdot (1 + \epsilon)$

gdzie $|\epsilon| \leq \mathrm{eps}$. Przyjmujemy, że błąd elementarnych operacji arytmetycznych to jedynie błąd zaokrąglenia dokładniego wyniku tych operacji.

Błędy i dokładność - podsumowanie

- Wykorzystywane w obliczeniach dane wejściowe to najczęściej liczby obarczone błędami reprezentacji maszynowej.
- Elementarne operacje arytmetyczne (całe ciągi tych operacji) również wprowadzają błędy w trakcie obliczeń.
- Błędy te ulegają propagacji i modyfikacji, może nastąpić kumulacja błędów.
- Całkowity błąd zadania obliczanego numerycznie jest z reguły znacznie większy od dokładności maszynowej.
- Błędy takie można analizować różnymi metodami, np. metodą probabilistyczną lub metodą najgorszego przypadku.
- Błąd algorytmu - łączny wpływ na obliczony wynik wszystkich błędów zaokrągleń, występujących podczas realizacji algorytmu
- Koszt algorytmu - mierzony najczęściej liczbą niezbędnych działań oraz niezbędnym obciążeniem pamięci komputera. Innym miernikiem kosztu algorytmu może być czas wykonywania obliczeń.

Zadanie numeryczne

Zadanie numeryczne - jasny i jednoznaczny opis powiązania funkcjonalnego między danymi wejściowymi, czyli zmiennymi niezależnymi zadania i danymi wyjściowymi, tj. szukanymi wynikami.

Zadania rozwiązywane numerycznie najczęściej nie występują samodzielnie, ale pojedynczo lub grupowo, jako zadania częściowe przy rozwiązywaniu złożonych problemów z różnych dziedzin zastosowań. Mówimy wtedy o klasie zadań.

Matematycznie zadanie obliczeniowe można przedstawić jako pewne odwzorowanie

$\Phi : R_n \to R_m$:

$$w = \Phi(d),$$

- $d = [d_1, d_2, \dots, d_n]^T$ - wektor danych wejściowych
- $w = [w_1, w_2, \dots, w_n]^T$ - wektor wyniku, rozwiązanie

W praktyce dysponujemy jedynie reprezentacjami maszynowymi danych:

$$rd(d_i) = d_i(1 + \epsilon_i)$$

gdzie $|\epsilon| \leq \text{eps}, \quad i = 1, 2, \dots, n.$

Uwarunkowanie zadania

Zadanie jest źle uwarunkowane, jeśli względnie małe błędy danych początkowych powodują duże błędy wyników obliczeń. Zadnie źle uwarunkowane obarczone jest dużymi błędami wyników niezależnie od zastosowanej metody lub algorytmu obliczania.

Uwarunkowanie zadania numerycznego to wrażliwość jego rozwiązania na poprawność danych początkowych.

Wskaźnikiem uwarukowania nazywamy wielkość charakteryzującą ilościowo maksymalny możliwy stosunek (iloraz) względnego błędu wyniku do względnego błędu (zaburzenia) danych. Dla błędów względnych definicja ta dotyczy względnego wskaźnika uwarunkowania.

Algorytm i jego numeryczne realizacje

Rozróżniamy trzy podstawowe pojęcia:

- Zadanie obliczeniowe (matematyczne): $w = \Phi(d)$
- Algorytm $A(d)$ obliczenia wyniku zadania $\varphi(d)$, sposób wyznaczenia wyniku zgodnie z jednoznacznie określoną kolejnością wykonywania elementarnych działań arytmetycznych
- Numeryczna realizacja $fl(A(d))$ algorytmu $A(d)$, polegająca na:
  - zastąpieniu liczb występujących w sformułowaniu $A(d)$ ich reprezentacjami zmiennopozycyjnymi,
  - wykonaniu operacji arytmetycznych w arytmetyce zmiennopozycyjnej „fl" czyli w sposób przybliżony,
  - operacje arytmetyczne - działania elementarne, obliczanie wartości funkcji standardowych.

Przykład

Załóżmy, że mamy funkcję $\Phi(a, b) = a^2 - b^2$

Zakładamy że $rd(a) = a$ i $rd(b) = b$ czyli zakładamy brak błędów reprezentacji danych. Interesuje nas tylko wpływ błędów elementarnych operacji arytmetycznych.

Rozważmy dwa algorytmy:

- $A_1(a,b) = a \cdot a - b \cdot b$
- $A_2(a,b) = (a+b) \cdot (a-b)$

(realizacja numeryczna - notatka)

Stabilność i niestabilność numeryczna

Algorytm jest niestabilny numerycznie jeśli w trakcie obliczeń "akumuluje się" błąd numeryczny i w rezultacie powstaje mało dokładny wynik. Zatem z niestabilnością numeryczną mamy do czynienia wtedy, gdy małe błędy danych lub popełniane w trakcie obliczeń rosną szybko w trakcie dalszych obliczeń powodując istotne/duże błędy/zniekształcenia wyników obliczeń.

W trakcie różnych operacji arytmetycznych może dochodzić do kumulacji błędów, np. jeśli dwie liczby obarczone są pewnymi znanymi błędami danych wejściowych, to w wyniku wykonania operacji na tych liczbach błędy również zostaną poddane tej operacji powodując kumulację możliwych błędów.

Dla niektórych wartości zmiennych algorytm może generować niedokładne wyniki obliczeń (np. pierwiastkowanie). Wynika to z błędów zaokrągleń wyników pośrednich, które mogą mieć wpływ na brak dokładności wyników końcowych. Jednym z przykładów takich algorytmów jest algorytm rozwiązujący równanie kwadratowe za pomocą "delty".

Algorytmy stabilne eliminują błędy zaokrągleń, dzięki zwiększa się dokładność wyników końcowych. Aby uniknąć błędów zaokrągleń w algorytmie rozwiązującym równanie kwadratowe, do obliczania pierwiastków równania kwadratowego należy zastosować "deltę" w połączeniu ze wzorami Viéte'a. Wynika to stąd, że błąd zaokrąglenia spowodowany jest odejmowaniem bliskich co do wartości liczb. Zachodzi to w liczniku wzorów na pierwiastki równania w algorytmie "delty".

In [ ]:

Metody numeryczne - wykład nr 2
Macierze

Aneta Wróblewska

Maria Curie-Skłodowska University, Lublin, Poland

March 4, 2024

Definicja macierzy

Macierzą nazywamy prostokątną tablicę $m \times n$ liczb umieszczonych w m wierszach i n kolumnach.

$$A = \begin{bmatrix} a_{1,1} & a_{1,2} & \dots & a_{1,n-1} & a_{1,n} \\ a_{2,1} & a_{2,2} & \dots & a_{2,n-1} & a_{2,n} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ a_{m-1,1} & a_{m-1,2} & \dots & a_{m-1,n-1} & a_{m-1,n} \\ a_{m,1} & a_{m,2} & \dots & a_{m,n-1} & a_{m,n} \end{bmatrix}$$

W skrócie oznaczamy ją $A = [a_{i,i}]$.

Macierz - oznaczenia

Dla macierzy $A = [a_{i,j}]$ mamy następujące oznaczenia:

- $a_{i,j}$ – element macierzy A,
- wskaźnik i - numerem wiersza, w którym znajduje się element $a_{i,j}$,
- wskaźnik j - numerem kolumny, w którym znajduje się element $a_{i,j}$,
- $a_{1,1}$, $a_{2,2}$, $a_{3,3}$, ... określają diagonalę macierzy,
- $m \times n$ - wymiar macierzy.

Macierze szczególne

Macierz wierszowa - macierz o wymiarze $1 \times n$:

$$A = \begin{bmatrix} a_1 & a_2 & a_3 & \dots & a_{n-1} & a_n \end{bmatrix}$$

Macierz kolumnowa - macierz o wymiarze $m \times 1$:

$$A = \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_{m-1} \\ a_m \end{bmatrix} = \begin{bmatrix} a_1 & a_2 & a_3 & \dots & a_{m-1} & a_m \end{bmatrix}^T$$

Macierz $A = [a_{i,j}]$ nazywamy macierzą zerową, jeśli zawiera same 0.

Macierz kwadratowa

Macierz kwadratowa - macierz, w której liczba wierszy i kolumn jest jednakowa. Tę liczbę nazywamy **stopniem macierzy**.
Na przykład następująca macierz

$$A = \begin{bmatrix} a_{1,1} & a_{1,2} & a_{1,3} \\ a_{2,1} & a_{2,2} & a_{2,3} \\ a_{3,1} & a_{3,2} & a_{3,3} \end{bmatrix}$$

jest macierzą kwadratową stopnia trzeciego.
Macierz symetryczna - macierz, której elementy spełniają równość $a_{i,j} = a_{j,i}$.

Macierz diagonalna

Macierz diagonalna - macierz kwadratowa, w której wszystkie elementy z wyjatkiem leżącej na diagonali sa równe 0

$$A = \begin{bmatrix} a_{1,1} & & \dots & 0 & 0 \\ 0 & a_{2,2} & \dots & 0 & 0 \\ 0 & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & a_{n-1,n-1} & 0 \\ 0 & 0 & \dots & 0 & a_{n,n} \end{bmatrix}$$

Szczególnym przypadkiem macierzy diagonalnej jest **macierz jednostkowa**, zawierająca na diagonali same jedynki.

Macierz górna trójkątna

Macierz górna trójkątna - macierz kwadratowa, w której elementy leżące na diagonali i powyżej niej są różne od 0.

$$\mathbf{U} = \begin{bmatrix} a_{1,1} & a_{1,2} & \dots & a_{1,n-1} & a_{1,n} \\ 0 & a_{2,2} & \dots & a_{2,n-1} & a_{2,n} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & a_{n-1,n-1} & a_{n-1,n} \\ 0 & 0 & \dots & 0 & a_{n,n} \end{bmatrix}$$

Macierz dolna trójkątna - macierz kwadratowa, w której elementy leżące na diagonali i poniżej niej są różne od 0.

$$\mathbf{L} = \begin{bmatrix} a_{1,1} & 0 & \dots & 0 & 0 \\ a_{2,1} & a_{2,2} & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ a_{n-1,1} & a_{n-1,2} & \dots & a_{n-1,n-1} & 0 \\ a_{n,1} & a_{n,2} & \dots & a_{n,n-1} & a_{n,n} \end{bmatrix}$$

Dodawanie i odejmowanie macierzy

Aby dodać (odjąć) macierze, muszą one mieć takie same wymiary.

$$\begin{bmatrix} a_{1,1} & a_{1,2} & \dots & a_{1,n} \\ a_{2,1} & a_{2,2} & \dots & a_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m,1} & a_{m,2} & \dots & a_{m,n} \end{bmatrix} \pm \begin{bmatrix} b_{1,1} & b_{1,2} & \dots & b_{1,n} \\ b_{2,1} & b_{2,2} & \dots & b_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ b_{m,1} & b_{m,2} & \dots & b_{m,n} \end{bmatrix} =$$

$$= \begin{bmatrix} a_{1,1} \pm b_{1,1} & a_{1,2} \pm b_{1,2} & \dots & a_{1,n} \pm b_{1,n} \\ a_{2,1} \pm b_{2,1} & a_{2,2} \pm b_{2,2} & \dots & a_{2,n} \pm b_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m,1} \pm b_{2,n} & a_{m,2} \pm b_{m,2} & \dots & a_{m,n} \pm b_{m,n} \end{bmatrix}$$

Dodawanie i odejmowanie macierzy jest operacją przemienną i łączną.

Mnożenie macierzy przez liczbę rzeczywistą

Macierz A pomnożona przez liczbę rzeczywistą k jest następującej postaci:

$$\begin{bmatrix} ka_{1,1} & ka_{1,2} & \dots & ka_{1,n} \\ ka_{2,1} & ka_{2,2} & \dots & ka_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ ka_{m,1} & ka_{m,2} & \dots & ka_{m,n} \end{bmatrix}$$

Transpozycja macierzy

Macierz transponowana do macierzy A to macierz A^T, która powstaje przez zamianę jej wierszy na kolumny i kolumn na wiersze.

Własności transpozycji

- $(A^T)^T = A$
- $(A \pm B)^T = A^T \pm B^T$
- $(kA)^T = kA^T$
- $(AB)^T = B^T A^T$
- $\det(A^T) = \det(A)$

Macierz symetryczna

Macierz symetryczna - macierz kwadratowa A, dla której zachodzi $A^T = A$

Macierz ortogonalna - macierz wymiaru $m \times n$, dla której zachodzi $A^T A = \mathbf{I}$. W przypadku, gdy $m = n$ mamy $A^T A = AA^T = \mathbf{I}$ oraz $A^T = A^{-1}$

Mnożenie macierzy

Iloczynem macierzy

$$A = \begin{bmatrix} a_{1,1} & a_{1,2} & \dots & a_{1,n} \\ a_{2,1} & a_{2,2} & \dots & a_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m,1} & a_{m,2} & \dots & a_{m,n} \end{bmatrix} \text{ i } B = \begin{bmatrix} b_{1,1} & b_{1,2} & \dots & b_{1,m} \\ b_{2,1} & b_{2,2} & \dots & b_{2,m} \\ \vdots & \vdots & \ddots & \vdots \\ b_{n,1} & b_{n,2} & \dots & b_{n,m} \end{bmatrix}$$

jest macierz kwadratowa $C = AB = \begin{bmatrix} c_{1,1} & c_{1,2} & \dots & c_{1,m} \\ c_{2,1} & c_{2,2} & \dots & c_{2,m} \\ \vdots & \vdots & \ddots & \vdots \\ c_{m,1} & c_{m,2} & \dots & c_{m,m} \end{bmatrix}$

której elementy oblicza się ze wzoru:
$c_{i,k} = \sum_{j=1}^{n} a_{i,j} b_{j,k}$ $(i, k = 1, 2, \dots, m)$

Mnożenie macierzy - własności

- mnożenie macierzy nie jest przemienne: $AB \neq BA$
- mnożenie macierzy jest łączne $(AB)C = A(BC)$
- mnożenie macierzy jest rozłączne względem dodawania $A(B + C) = AB + AC$
- macierz jednostkowa jest elementem neutralnym $AI = A$ oraz $IA = A$

Przekształcenia elementarne

Elementarne przekształcenia macierzy dzielą się na przekształcenia pierwszego i drugiego rodzaju.

Przekształceniami pierwszego rodzaju macierzy A są:

- przestawienie dwóch wierszy,
- pomnożenie dowolnego wiersza przez liczbę różną od zera,
- dodanie do wiersza krotności innego wiersza.

Przekształcenia drugiego rodzaju są analogiczne i dotyczą kolumn.

Przekształceniami drugiego rodzaju macierzy A są:

- przestawienie dwóch kolumn,
- pomnożenie dowolnej kolumny przez liczbę różną od zera,
- dodanie do kolumny krotności innej kolumny.

Znaczenie przekształceń elementarnych

liczenie rzędu macierzy

- na rząd macierzy nie mają wpływu żadne operacje elementarne, wszystkie je można stosować

liczenie wyznacznika macierzy kwadratowej

- wartości wyznacznika nie zmienia tylko dodanie wielokrotności jednego wiersza do drugiego albo jednej kolumny do drugiej
- pomnożenie wiersza lub kolumny o wartość k zwiększa wyznacznik k razy
- zmiana miejscami wierszy lub kolumn zmienia znak wyznacznika

Znaczenie przekształceń elementarnych

rozwiązywanie układów równań liniowych

- można działać wyłącznie na wierszach, zmiany w kolumnach wprowadzają do układu nowe zmienne

znajdowanie macierzy odwrotnej

- można działać na wierszach albo na kolumnach ale nie wolno mieszać jednych operacji z drugimi

Wyznacznik macierzy

Wyznacznik – funkcja przyporządkowująca każdej macierzy kwadratowej $A_{n,n}$ pewną liczbę.
Wyznacznik macierzy kwadratowej A oznaczany jest przez $\det(A)$ albo $|A|$

Jeżeli $\det(A) \neq 0$, to macierz A nazywa się **nieosobliwą**. Jeżeli $\det(A) = 0$, to macierz A nazywa się **osobliwą**.
Własności:

- $\det(A) = \det(A^T)$
- $\det(AB) = \det(A)\det(B)$

Wyznaczanie wyznacznika macierzy - rozwinięcie Laplace'a

Rozważmy macierz A o wymiarach $n \times n$:

$$A = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{bmatrix}$$

Wyznacznik macierzy A za pomocą rozwinięcia Laplace'a:

- $\det(A) = a_{11}$ jeśli $n = 1$
- $\det(A) = \sum_{i=1}^{n} (-1)^{i+j} a_{ij} M_{ij}$ jeśli $n > 1$

gdzie M_{ij} to macierz stopnia $n-1$ powstałą z macierzy A poprzez skreślenie i-tego wiersza i j-tej kolumny (minor). Powyższa definicja opiera się o rozwinięcie wzdłuż j-tej kolumny.

det$A$ - przypadki elementarne

Dla macierzy stopnia 1 ($n = 1$):

$$A = [a_{11}], \quad \det(A) = a_{11}$$

Dla macierzy stopnia 2 ($n = 2$):

$$A = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}$$

$$\det(A) = a_{11} \cdot a_{22} - a_{12} \cdot a_{21}$$

det$A$ - przypadki elementarne

Dla macierzy stopnia 3 ($n = 3$):

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}$$

$$\det(A) = a_{11} \cdot a_{22} \cdot a_{33} + a_{12} \cdot a_{23} \cdot a_{31} + a_{13} \cdot a_{21} \cdot a_{32}$$

$$- a_{13} \cdot a_{22} \cdot a_{31} - a_{12} \cdot a_{21} \cdot a_{33} - a_{11} \cdot a_{23} \cdot a_{32}$$

Macierz odwrotna

Rozważmy kwadratową macierz A o wymiarach $n \times n$. Macierzą odwrotną do macierzy A jest macierz A^{-1}, spełniająca następujący warunek:

$$A \cdot A^{-1} = A^{-1} \cdot A = I$$

gdzie I to macierz jednostkowa.
Warunkiem koniecznym i wystarczającym istnienia macierzy odwrotnej dla macierzy A jest:

$$\det(A) \neq 0$$

Odwracanie macierzy przy pomocy macierzy dopełnień

Niech A będzie nieosobliwą macierzą kwadratową, czyli $\det(A) \neq 0$. Element A_{ij} to **dopełnienie algebraiczne** elementu a_{ij}, obliczone ze wzoru:

$$A_{ij} = (-1)^{i+j} M_{ij}$$

gdzie M_{ij} to wyznacznik macierzy powstałej poprzez usunięcie i-tego wiersza i j-tej kolumny z macierzy A.

Macierz dopełnień

Przez A^D oznaczmy macierz dopełnień algebraicznych, a przez $(A^D)^T$ jej transpozycję:

$$(A^D)^T = \begin{bmatrix} A_{11} & A_{21} & \dots & A_{n1} \\ A_{12} & A_{22} & \dots & A_{n2} \\ \vdots & \vdots & \ddots & \vdots \\ A_{1n} & A_{2n} & \dots & A_{nn} \end{bmatrix}$$

gdzie A_{ij} to dopełnienie algebraiczne elementu a_{ij} macierzy A.

Odwracanie macierzy przy pomocy macierzy dopełnień

Transponowaną macierz dopełnień algebraicznych nazywamy **macierzą dołączoną** do macierzy A. **Macierz odwrotną** A^{-1} można otrzymać dzieląc wszystkie elementy macierzy dołączonej przez $\det(A)$:

$$A^{-1} = \frac{1}{\det(A)}(A^D)^T$$

Odwracanie macierzy - przykłady

Przykład 1: Rozważmy macierz A o wymiarach 2×2:

$$A = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$$

Jeśli $\det(A) \neq 0$, to macierz odwrotna A^{-1} istnieje i jest równa:

$$A^{-1} = \frac{1}{ad - bc} \begin{bmatrix} d & -b \\ -c & a \end{bmatrix}$$

Przykład 2: Dla macierzy A o wymiarach 3×3 można skorzystać z rozwinięcia Laplace'a - przykład na tablicy.

Zależności dla macierzy odwrotnej

- $(A^{-1})^{-1} = A$
- $(kA)^{-1} = \frac{1}{k}A^{-1}$, gdzie k jest liczbą różną od zera
- $(A^T)^{-1} = (A^{-1})^T$, gdzie A^T to macierz transponowana
- $(AB)^{-1} = B^{-1}A^{-1}$, gdzie AB to iloczyn macierzy
- $\det(A^{-1}) = \frac{1}{\det(A)}$

Te zależności są istotne przy manipulacjach na macierzach odwrotnych w różnych kontekstach matematycznych.

Przekształcenia elementarne a odwracanie macierzy

Z definicji mnożenia macierzy wynika, że dla dowolnej macierzy A: operacji elementarnej na wierszach macierzy A odpowiada pomnożenie macierzy A z lewej strony przez macierz, która powstaje z macierzy jednostkowej I przez wykonanie na niej tej samej operacji.
Stosując operacje elementarne na wierszach nieosobliwej macierzy A (tzn. takiej, że $\det(A) \neq 0$), możemy ją przekształcić do macierzy jednostkowej I. Wynika stąd, że istnieją macierze $B_1, B_2, \ldots, B_s$ takie, że

$$B_s \cdot \ldots \cdot B_2 \cdot B_1 \cdot A = I.$$

Zatem $A^{-1} = B_s \cdot \ldots \cdot B_2 \cdot B_1$, czyli $A^{-1} = B_s \cdot \ldots \cdot B_2 \cdot B_1 \cdot I$.
Stąd macierz A^{-1} powstaje z macierzy I przez wykonanie na niej tych samych operacji elementarnych, co na macierzy A.

Przekształcenia elementarne a odwracanie macierzy

W praktyce przy obliczaniu macierzy odwrotnej do macierzy nieosobliwej A przy pomocy operacji elementarnych na wierszach (dodawanie wielokrotności jednego wiersza do innego; mnożenie wiersza przez skalar lub zamiana miejscami dwóch wierszy) postępujemy w sposób następujący.
Z prawej strony macierzy A dopisujemy macierz jednostkową tego samego stopnia. Na wierszach otrzymanej w ten sposób macierzy blokowej $[A|I]$ wykonujemy operacje elementarne aż do uzyskania macierzy blokowej postaci $[I|B]$. Macierz B jest szukaną macierzą odwrotną do macierzy A, tj. $B = A^{-1}$. Symbolicznie można zapisać: $[A|I] \sim [I|A^{-1}]$.

Metody numeryczne
Wykład nr 3
Układy równań liniowych

Aneta Wróblewska

UMCS, Lublin

March 12, 2024

Układ równań liniowych, zapis macierzowy

Układ m równań liniowych z n niewiadomymi:

$$\begin{aligned} a_{11}x_1 + a_{12}x_2 + a_{13}x_3 + \ldots + a_{1n}x_n &= b_1, \\ a_{21}x_1 + a_{22}x_2 + a_{23}x_3 + \ldots + a_{2n}x_n &= b_2, \\ &\ldots \\ a_{m1}x_1 + a_{m2}x_2 + a_{m3}x_3 + \ldots + a_{mn}x_n &= b_m. \end{aligned}$$

można zapisać następująco:

$$\sum_{j=1}^{n} a_{ij}x_j = b_i \quad (i = 1, 2, \ldots, m),$$

lub w zapisie macierzowym

$$Ax = b$$

gdzie $A = [a_{ij}]$ jest macierzą główną układu równań, złożoną ze współczynników układu, wektora niewiadomych x i wektora wyrazów wolnych b.

Układ równań liniowych, zapis macierzowy

$$x = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}, \quad b = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{bmatrix}$$

Zatem powyższy układ równań $Ax = b$ może być przedstawiony w zapisie macierzowym w następujący sposób:

$$\begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \dots & a_{mn} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{bmatrix}$$

Macierz rozszerzona

Symbolem A_b oznaczymy macierz o rozmiarach $m \times (n+1)$, która powstaje z macierzy A przez dołączenie do niej wektora b jako $(n+1)$-szej kolumny. Macierz A_b (macierz główna i kolumna wyrazów wolnych) nazywamy **macierzą rozszerzoną** .

$$A_b = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} & b_1 \\ a_{21} & a_{22} & \dots & a_{2n} & b_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ a_{m1} & a_{m2} & \dots & a_{mn} & b_m \end{bmatrix}$$

Rząd macierzy, istnienie rozwiązania

Dowolną macierz A, np. macierz główną układu równań, można uważać za zbudowaną z jej wektorów kolumnowych lub jej wektorów wierszowych:

$$A = (a_{.1}, a_{.2}, \ldots, a_{.n})$$

$$A = (a_{1.}^T, a_{2.}^T, \ldots, a_{m.}^T)^T$$

Największa liczba niezależnych liniowo wektorów kolumnowych w A jest równa największej liczbie niezależnych liniowo wektorów wierszowych w A. Ta liczba r nazywana jest **rzędem macierzy** A

$$\text{rank}(A) = r$$

Rozwiązania w układach równań liniowych

Twierdzenie 1 Kroneckera-Capellego

- Jeżeli rząd macierzy głównej jest równy rzędowi macierzy rozszerzonej i liczbie niewiadomych w układzie $\text{rank}(A) = \text{rank}(A_b) = n$, to układ ma dokładnie jedno rozwiązanie (układ oznaczony).
- Jeżeli rzędy macierzy głównej i macierzy rozszerzonej są sobie równe ale są mniejsze od liczby niewiadomych $\text{rank}(A) = \text{rank}(A_b) < n$, to układ ma nieskończenie wiele rozwiązań (układ nieoznaczony).
- Jeżeli rząd macierzy głównej jest mniejszy niż rząd macierzy rozszerzonej $\text{rank}(A) < \text{rank}(A_b)$, to układ równań nie ma rozwiązań (układ sprzeczny).

Typy układów równań liniowych

Typy układów równań liniowych:

- **Układ jednorodny:** Jeżeli wszystkie wyrazy wolne są równe 0, to układ nazywamy jednorodnym. Taki układ ma zawsze rozwiązanie.
- **Układ kwadratowy:** Jeżeli liczba wierszy równa jest liczbie kolumn w macierzy głównej (m = n), to układ nazywamy kwadratowym.

Metody rozwiązywania układów równań liniowych

Metody rozwiązywania układów równań liniowych:

- **Bezpośrednie (dokładne):** dają dokładne rozwiązanie po skończonej liczbie przekształceń układu wejściowego, pomijając oczywiście błędy zaokrągleń.
  - efektywne dla układów o macierzach pełnych
  - mocno obciążają pamięć
  - ze względu na błędy zaokrągleń mogą być niestabilne
- **Iteracyjne (przybliżone):** tworzą ciąg wektorów zbieżny do szukanego rozwiązania.
  - liczba kroków nie jest z góry znana
  - dobrze się sprawdzają dla macierzy rzadkich o dużych rozmiarach
  - obciążenie pamięci nie jest zbyt duże
  - mogą wystąpić problemy ze zbieżnością rozwiązania

Typy zagadnień w rozwiązaniach numerycznych

Spotykane w praktyce inżynierskiej macierze współczynników dzielą się ogólnie na dwie grupy:

1. **Macierze pełne (ale nieduże):** Mają one mało elementów niezerowych. Nieduże oznacza, że są stopnia mniejszego np. od 30 (od 1 000). Macierze takie występują w różnych zadaniach statystki, w fizyce matematycznej i w technice, np. przy obliczaniu reakcji w podporach belek, sił w prętach kratownicy itp.
2. **Macierze rzadkie (ale najczęściej bardzo duże):** Mają one mało elementów niezerowych. Macierze te można tak skonstruować, że elementy niezerowe leżą zawsze bardzo blisko głównej diagonali. Przez duże rozumie się macierze stopnia 100 (10 000) lub większe. Występują one powszechnie w rozwiązywaniu numerycznym równań różniczkowych częściowych, np. w wyznaczaniu pól temperatury, pól przemieszczeń, odkształceń i naprężeń, itd.

Metody bezpośrednie i iteracyjne

Do metod bezpośrednich należą m.in.:

1. użycie macierzy odwrotnej,
2. wzory Cramera,
3. układ równań z macierzą trójkątną,
4. metoda eliminacji Gaussa,
5. rozkłady trójkątne macierzy,
6. Gaussa-Jordana,
7. Doolittle'a,
8. Crout'a,
9. Choleski'ego (Banachiewicza).

Do metod iteracyjnych należą m.in.:

1. Jacobi'ego,
2. Gaussa-Seidel'a,
3. Czebyszewa.

Użycie macierzy odwrotnej

Pierwsza, omówiona tutaj, metoda bezpośrednia rozwiązania układ równań będzie polegała na wykorzystaniu macierzy odwrotnej odpowiadającej danemu układowi równań.
Przypomnijmy z poprzedniego wykładu, że dla macierzy odwrotnej A^{-1} do macierzy kwadratowej A, zachodzi:

$$AA^{-1} = A^{-1}A = I$$

gdzie I jest macierzą jednostkową.

Rozwiązanie układu równań z użyciem macierzy odwrotnej

Układ równań w postaci macierzowej:

$$Ax = b$$

można rozwiązać następująco, jeśli znamy macierz odwrotną:

$$A^{-1}Ax = A^{-1}b \Rightarrow Ix = A^{-1}b$$

czyli:

$$x = A^{-1}b$$

Wzory Cramera

Kolejnym sposobem rozwiązania układu równań są **wzory Cramera**.
Jeżeli wyznacznik macierzy głównej układu równań $Ax = b$ jest różny od zera, $\det(A) \neq 0$, tzn. jeśli macierz A składa się z wektorów niezależnych liniowo (jest nieosobliwa), to

$$x_k = \frac{\det(A_k)}{\det(A)} \quad k = 1, 2, \ldots, n$$

gdzie A_k jest macierzą powstałą przez zastąpienie k-tej kolumny wektorem wyrazów wolnych b.

Wzory Cramera, zapis wektorowy

Układ równań $Ax = b$ można zapisać w postaci wektorowej następująco

$$\alpha_1 x_1 + \alpha_2 x_2 + \ldots + \alpha_n x_n = b$$

gdzie $\alpha_k (k = 1, 2, \ldots, n)$ oznacza wektor o składowych $(a_{1k}, a_{2k}, \ldots, a_{nk})$, natomiast b - wektor o składowych $(b_1, b_2, \ldots, b_n)$.
Jeżeli $\det(\alpha_1, \alpha_2, \ldots, \alpha_n) \neq 0$, tzn. jeśli wektory $\alpha_1, \alpha_2, \ldots, \alpha_n$ są niezależne liniowo (macierz A jest nieosobliwa), to

$$x_k = \frac{\det(\alpha_1, \ldots, \alpha_{k-1}, \beta, \alpha_{k+1}, \ldots, \alpha_n)}{\det(\alpha_1, \alpha_2, \ldots, \alpha_n)} \quad k = 1, 2, \ldots, n$$

Wzory noszą nazwę wzorów Cramera.

Wzory Cramera - przykład 3x3

Rozwiązać układ równań posługując się wzorami Cramera

$$\begin{aligned} a_{11}x_1 + a_{12}x_2 + a_{13}x_3 &= b_1 \\ a_{21}x_1 + a_{22}x_2 + a_{23}x_3 &= b_2 \\ a_{31}x_1 + a_{32}x_2 + a_{33}x_3 &= b_3 \end{aligned}$$

Obliczamy wyznacznik macierzy głównej:

$$\det(A) = \det \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}$$

Wzory Cramera - przykład 3x3 cd.

Obliczamy kolejno

$$x_1 = \frac{1}{\det(A)} \det \begin{bmatrix} b_1 & a_{12} & a_{13} \\ b_2 & a_{22} & a_{23} \\ b_3 & a_{32} & a_{33} \end{bmatrix} \equiv \frac{\det(A_1)}{\det(A)}$$

$$x_2 = \frac{1}{\det(A)} \det \begin{bmatrix} a_{11} & b_1 & a_{13} \\ a_{21} & b_2 & a_{23} \\ a_{31} & b_3 & a_{33} \end{bmatrix} \equiv \frac{\det(A_2)}{\det(A)}$$

$$x_3 = \frac{1}{\det(A)} \det \begin{bmatrix} a_{11} & a_{12} & b_1 \\ a_{21} & a_{22} & b_2 \\ a_{31} & a_{32} & b_3 \end{bmatrix} \equiv \frac{\det(A_3)}{\det(A)}$$

Rozwiązanie układu równań z macierzą trójkątną

Jezeli macierz układu równan liniowych jest macierzą trójkątną, rozwiązuje się go szczególnie łatwo. Dla ustalenia uwagi przyjmijmy, że A jest macierzą trójkątną górną. Aby istniało jednoznaczne rozwiązanie, macierz musi być nieosobliwa. Innymi słowy, wszystkie elementy na głównej przekątnej macierzy A muszą być różne od zera.

Rozwiązanie układu równań z macierzą trójkątną

Nasz układ będzie miał postać:

$$\begin{aligned} a_{11}x_1 + a_{12}x_2 + \ldots + a_{1n}x_n &= b_1 \\ a_{22}x_2 + \ldots + a_{2n}x_n &= b_2 \quad (1) \\ \ldots & \\ \ldots \quad a_{nn}x_n &= b_n \end{aligned}$$

Zwróćmy uwagę, że składową x_n wektora niewiadomych x obliczymy natychmiast z ostatniego równania powyższego układu. Podstawiając wynik do przedostatniego równania, wyznaczamy x_{n-1}. Procedurę kontynuujemy aż do wyliczenia x_1.

Rozwiązanie układu równań z macierzą trójkątną

Rozwiązanie zapisze się wzorem:

$$x_n = \frac{b_n}{a_{nn}}, \quad x_i = \frac{b_i - \sum_{k=i+1}^{n} a_{ik} x_k}{a_{ii}}, \quad i = n-1, n-2, \ldots, 1. \quad (2)$$

Ponieważ obliczenia zaczynamy od ostatniej składowej wektora niewiadomych, metoda ta nazywana jest podstawianiem w tył. Do znalezienia x musimy wykonać $M = \frac{1}{2}n^2 + \frac{1}{2}n$ mnożeń i dzielenia oraz $D = \frac{1}{2}n^2 - \frac{1}{2}n$ dodawań. Koszt obliczeń jest więc niewiele większy od kosztu mnożenia wektora przez macierz trójkątną!

Rozwiązanie układu równań z macierzą trójkątną

W podobny sposób znajdziemy rozwiązanie układu z macierzą trójkątną dolną, tym razem wykonując podstawianie w przód:

$$x_1 = \frac{b_1}{a_{11}} \quad (3)$$

$$x_i = \frac{b_i - \sum_{k=1}^{i-1} a_{ik} x_k}{a_{ii}}, \quad i = 2, 3, \ldots, n.$$

Koszt obliczeń jest oczywiście taki sam jak poprzednio. Wiele metod numerycznego rozwiązywania układów równań z dowolnymi macierzami polega na sprowadzeniu układu wyjściowego do postaci trójkątnej, a następnie zastosowaniu jednego ze wzorów (2)-(3).

Eliminacja Gaussa

Jedną z takich właśnie metod jest eliminacja Gaussa. Chociaż nazwana tak na cześć Carla Friedricha Gaussa, po raz pierwszy została zaprezentowana dużo wcześniej, bo już około 150 roku p.n.e w słynnym chińskim podręczniku matematyki „Dziewięć rozdziałów sztuki matematycznej".
Dla ułatwienia założymy, że macierz układu jest wymiaru 3×3, tzn.

$$\begin{aligned} a_{11}x_1 + a_{12}x_2 + a_{13}x_3 &= b_1, \\ a_{21}x_1 + a_{22}x_2 + a_{23}x_3 &= b_2, \\ a_{31}x_1 + a_{32}x_2 + a_{33}x_3 &= b_3. \end{aligned} \quad (4)$$

Eliminacja Gaussa

Naszym celem jest sprowadzenie tego układu do postaci trójkątnej. Odejmując od drugiego wiersza układu (4) pierwszy pomnożony przez a_{21}/a_{11}, a od trzeciego pierwszy pomnożony przez a_{31}/a_{11}, otrzymamy:

$$\begin{aligned} a_{11}^{(0)}x_1 + a_{12}^{(0)}x_2 + a_{13}^{(0)}x_3 &= b_1^{(0)}, \\ a_{22}^{(1)}x_2 + a_{23}^{(1)}x_3 &= b_2^{(1)}, \\ a_{32}^{(1)}x_2 + a_{33}^{(1)}x_3 &= b_3^{(1)}, \end{aligned} \quad (5)$$

gdzie

$$a_{ij}^{(0)} = a_{ij}, \quad b_i^{(0)} = b_i, \quad i,j = 1,2,3,$$

$$a_{ij}^{(1)} = a_{ij}^{(0)} - \frac{a_{i1}^{(0)}a_{1j}^{(0)}}{a_{11}^{(0)}}, \quad b_i^{(1)} = b_i^{(0)} - \frac{a_{i1}^{(0)}b_1^{(0)}}{a_{11}^{(0)}}, \quad i,j = 2,3.$$

Eliminacja Gaussa

Aby wyeliminować zmienną x_2 z trzeciego równania w układzie (5), odejmujemy od niego drugie pomnożone przez $a_{32}^{(1)}/a_{22}^{(1)}$:

$$\begin{aligned} a_{11}^{(0)}x_1 + a_{12}^{(0)}x_2 + a_{13}^{(0)}x_3 &= b_1^{(0)}, \\ a_{22}^{(1)}x_2 + a_{23}^{(1)}x_3 &= b_2^{(1)}, \\ a_{33}^{(2)}x_3 &= b_3^{(2)}, \end{aligned}$$

gdzie

$$a_{ij}^{(2)} = a_{ij}^{(1)} - \frac{a_{i2}^{(1)}a_{2j}^{(1)}}{a_{22}^{(1)}},$$

$$b_i^{(2)} = b_i^{(1)} - \frac{a_{i2}^{(1)}b_2^{(1)}}{a_{22}^{(1)}}, \quad i,j = 3.$$

Eliminacja Gaussa

Wzory na współczynniki macierzy i wyrazy wolne w każdym kroku eliminacji Gaussa łatwo jest uogólnić na przypadek macierzy dowolnego rozmiaru:

$$a_{ij}^{(k)} = a_{ij}^{(k-1)} - \frac{a_{ik}^{(k-1)} a_{kj}^{(k-1)}}{a_{kk}^{(k-1)}}, \quad i,j = k+1, k+2, \ldots, n,$$

$$b_{i}^{(k)} = b_{i}^{(k-1)} - \frac{a_{ik}^{(k-1)} b_{k}^{(k-1)}}{a_{kk}^{(k-1)}}, \quad i,j = k+1, k+2, \ldots, n.$$

Tym sposobem rzeczywiście otrzymaliśmy układ trójkątny, który można teraz rozwiązać podstawianiem wstecz (6):

$$x_i = \frac{b_i^{(i-1)} - \sum_{j=i+1}^{n} a_{ij}^{(i-1)} x_j}{a_{ii}^{(i-1)}}, \quad i = n, n-1, \ldots, 1. \quad (6)$$

Wybór elementu podstawowego

Eliminacja Gaussa w formie przedstawionej powyżej nie jest niezawodna.
Jeśli macierz jest nieosobliwa, to istnieje jednoznaczne rozwiązanie.
W przypadku, gdy $a_{11} = 0$ eliminacja Gaussa zawodzi już w pierwszym kroku, ponieważ algorytm wymaga dzielenia przez a_{11}.
Dlatego najczęściej stosuje się pewną modyfikację metody Gaussa, zwana **częściowym wyborem elementu podstawowego**.
Elementem podstawowym nazywamy ten element macierzy A, za pomocą którego dokonujemy eliminacji zmiennej z dalszych równań.

Wybór elementu podstawowego

- W przypadku, gdy któryś z elementów na głównej diagonali, tzn. któryś z elementów podstawowych (ang. pivot) jest równy zero, tj. $a_{kk}^{(k)} = 0$, należy tak przekształcić układ równań, o ile tylko jest on nieosobliwy, aby wartość $a_{kk}^{(k)} \neq 0$.
- Należy w tym celu wykonać operację przestawienia wierszy, tj. $(E_k) \leftrightarrow (E_p)$ gdzie $k + 1 \leqslant p \leqslant n$.
- W praktyce często żąda się dodatkowo, aby element podstawowy przyjmował jak największą wartość absolutną.

Wybór elementu podstawowego

Przykład: Rozważmy układ, biorąc pod uwagę jego macierz rozszerzoną w następującej postaci:

$$\left[\begin{array}{ccc|c} 0 & 2 & 2 & 1 \\ 3 & 3 & 0 & 3 \\ 1 & 0 & 1 & 2 \end{array}\right]$$

Zamiana wierszy w macierzy układu

Wiemy już, że $a_{11} = 0$ nie może być elementem podstawowym. Dlatego skorzystamy ze wspomnianej powyżej własności układów równań i zamienimy ze sobą wiersze w macierzy układu, tak aby nowy element diagonalny w jej pierwszym wierszu był różny od zera.

Wybór elementu podstawowego

Ze względu na błędy zaokrągleń w i-tym kroku eliminacji Gaussa powinniśmy się kierować wartościami elementów w i-tej kolumnie i wybierać wiersz, który ma największy element (wartość bezwzględną).
Po zamianie wierszy otrzymujemy następująca macierz, na której możemy wykonać pierwszy krok eliminacji Gaussa:

$$\left[\begin{array}{ccc|c} 3 & 3 & 0 & 3 \\ 0 & 2 & 2 & 1 \\ 0 & -1 & 1 & 1 \end{array}\right]$$

Wybór elementu podstawowego

W kolejnym kroku nie musimy zamieniać wierszy ze sobą:

$$\left[\begin{array}{ccc|c} 3 & 3 & 0 & 3 \\ 0 & 2 & 2 & 1 \\ 0 & 0 & 2 & \frac{3}{2} \end{array}\right]$$

Końcowe rozwiązanie znajdziemy ze wzoru (6):

$$x_3 = \frac{b_3^{(3)}}{a_{33}^{(3)}} = \frac{3}{4},$$

$$x_2 = \frac{b_2^{(3)} - a_{23}^{(3)} x_3}{a_{22}^{(3)}} = -\frac{1}{4},$$

$$x_1 = \frac{b_1^{(3)} - a_{12}^{(3)} x_2 - a_{13}^{(3)} x_3}{a_{11}^{(3)}} = \frac{5}{4}.$$

Wybór elementu podstawowego

Taka metoda postępowania jest częściowym wyborem elementu głównego. Pełny wybór realizowalibyśmy wyszukując współczynnika największego co do modułu nie tylko w kolumnie pod napotkanym zerem ale w całej podmacierzy "w dół i w prawo". Może to jeszcze poprawić dokładność ale jest znacznie bardziej czasochłonne.

Rozkład LU

Wiemy już, że rozwiązanie układu równań z macierzami trójkątnymi jest szczególnie łatwe. Przypuśćmy zatem, że macierz A dowolnego układu da się przedstawić w postaci iloczynu macierzy trójkątnej dolnej L i trójkątnej górnej U,

$$A = LU.$$

Jeżeli macierz A jest nieosobliwa, zachodzi

$$A^{-1} = (LU)^{-1} = U^{-1}L^{-1},$$

a więc rozwiązanie układu da się przedstawić w postaci

$$\mathrm{x} = A^{-1}\mathrm{b} = U^{-1}\left(L^{-1}\mathrm{b}\right).$$

Aby znaleźć rozwiązanie x układu dysponując rozkładem LU jego macierzy, wystarczy rozwiązać dwa układy trójkątne:

$$L\mathrm{y} = \mathrm{b},$$

$$U\mathrm{x} = \mathrm{y}.$$

Eliminacja Gaussa a rozkład LU

Okazuje się, że jednym ze sposobów uzyskania rozkładu LU jest omówiona już eliminacja Gaussa.
W wyniku eliminacji Gaussa zostanie uzyskana macierz górnotrójkątna U. Z kolei macierz dolnotrójkątną L należy wyznaczyć w następujący sposób. Pierwszy krok eliminacji Gaussa polega na wyzerowaniu kolumny poniżej elementu a_{11}. Zerując i-tym wierszem j-ty wiersz należy wpisać na pozycji L_{ji} współczynnik przez który został pomnożony wiersz i-ty. Ponadto macierz dolnotrójkątna L ma zawsze na diagonali wartości 1. Po wykonaniu eliminacji Gaussa zostaną uzyskane dwie macierze L i U.

Eliminacja Gaussa a rozkład LU

Nie każda macierz nieosobliwa można przedstawić w postaci $A = LU$. Aby rozkład istniał, wszystkie minory główne macierzy muszą być różne od zera. Jednak, jeżeli eliminacje Gaussa można przeprowadzić do końca, rozkład LU na pewno istnieje.
Jeżeli eliminacja Gaussa wymaga zamiany wierszy, wówczas zamiast rozkładu LU macierzy A znajdziemy rozkład permutacji jej wierszy,

$$PA = LU,$$

gdzie P to macierz permutacji. Jej znaczenie najlepiej jest zilustrować na przykładzie:

$$PA = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} = \begin{bmatrix} a_{31} & a_{32} & a_{33} \\ a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \end{bmatrix}$$

Eliminacja Gaussa a rozkład LU

Macierz permutacji ma następującą własność:

$$P^T P = I \quad \Rightarrow \quad P^T = P^{-1}.$$

Stąd wynika

$$A = P^T LU.$$

Metoda Doolittle'a

Rozkład LU możemy poszukać również w inny sposób, traktując równość (7) jako układ n^2 równań dla n^2 niewiadomych l_{ij} i u_{ij}:

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ l_{21} & 1 & 0 \\ l_{31} & l_{32} & 1 \end{bmatrix} \cdot \begin{bmatrix} u_{11} & u_{12} & u_{13} \\ 0 & u_{22} & u_{23} \\ 0 & 0 & u_{33} \end{bmatrix}$$

Stąd

$$u_{11} = a_{11}, \quad u_{12} = a_{12}, \quad u_{13} = a_{13},$$

$$l_{21} = \frac{a_{21}}{u_{11}}, \quad u_{22} = a_{22} - l_{21}u_{12}, \quad u_{23} = a_{23} - l_{21}u_{13},$$

$$l_{31} = \frac{a_{31}}{u_{11}}, \quad l_{32} = \frac{a_{32} - l_{31}u_{12}}{u_{22}}, \quad u_{33} = a_{33} - l_{31}u_{13} - l_{32}u_{23}.$$

Metoda Doolittle'a

W przypadku ogólnym elementy macierzy L i U obliczamy kolejno dla $i = 1, 2, \ldots, n$ z następujących wzorów:

$$u_{ij} = a_{ij} - \sum_{k=1}^{i-1} l_{ik} u_{kj}, \qquad j = i, i+1, \ldots, n,$$

$$l_{ji} = \frac{a_{ji} - \sum_{k=1}^{i-1} l_{jk} u_{ki}}{u_{ii}}, \qquad j = i+1, i+2, \ldots, n. \quad (8)$$

Metoda Doolittle'a

Metoda Doolittle'a staje się niezawodna dopiero w połączeniu z wyborem elementu podstawowego. Wiersze powinniśmy zamieniać ze sobą miejscami tak, aby element u_{ii} we wzorze (8) był jak największy.

Metoda Crouta

Do tej pory zakładaliśmy po cichu, że elementy diagonalne macierzy L są równe 1. Jeżeli dla odmiany przyjmiemy, że to U ma na głównej przekątnej same jedynki:

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} = \begin{bmatrix} l_{11} & 0 & 0 \\ l_{21} & l_{22} & 0 \\ l_{31} & l_{32} & l_{33} \end{bmatrix} \begin{bmatrix} 1 & u_{12} & u_{13} \\ 0 & 1 & u_{23} \\ 0 & 0 & 1 \end{bmatrix}.$$

Metoda Crouta

Ponownie potraktujemy powyższe wyrażenie jak równanie na niewiadome elementy macierzy trójkątnych. Otrzymamy metodę zaproponowaną przez Crouta:

$$
\begin{aligned}
& u_{12} = a_{12}, \quad u_{13} = a_{13}, \\
& l_{21} = \frac{a_{21}}{u_{11}}, \\
& u_{23} = a_{23} - l_{21} u_{13}, \\
& l_{31} = \frac{a_{31}}{u_{11}}, \quad l_{32} = \frac{a_{32} - l_{31} u_{12}}{u_{22}}, \\
& u_{33} = a_{33} - l_{31} u_{13} - l_{32} u_{23}.
\end{aligned}
$$

Inne rozkłady macierzy

Rozkład macierzy A na iloczyn LU nie jest jedynym możliwym rozkładem. Omówimy krótko jeszcze jeden z nich - rozkład Cholesky'ego (Banachiewicza).

Rozkład Cholesky'ego (Banachiewicza)

Jeżeli macierz układu jest macierzą symetryczną, tzn.

$$a_{ij} = a_{ji}, \quad i, j = 1, \ldots, n,$$

i dodatnio określona,

$$\mathrm{x}^T A\mathrm{x} > 0 \quad \text{dla każdego x},$$

to istnieje dla niej bardziej wydajny od LU rozkład na macierze trójkątne, a mianowicie

$$A = LL^T,$$

gdzie L to macierz trójkątna dolna. Traktując to jako układ równań ze względu na elementy macierzy L, znajdziemy:

$$l_{ii} = \sqrt{a_{ii} - \sum_{k=1}^{i-1} l_{ik}^2},$$

$$l_{ji} = \frac{1}{l_{ii}} \left(a_{ij} - \sum_{k=1}^{i-1} l_{jk} l_{ik} \right), \quad j = i+1, i+2, \ldots, n.$$

Rozkład Cholesky'ego

Ilość operacji potrzebna do znalezienia rozkładu Cholesky'ego jest o połowę mniejsza w porównaniu z LU. Dodatkową zaletą, związaną z własnościami macierzy A, jest niezawodność (metoda nie wymaga wyboru elementu podstawowego) i stabilność numeryczna.

Metody numeryczne
Wykład nr 4
Układy równań liniowych II - metody iteracyjne

Aneta Wróblewska

UMCS, Lublin

March 17, 2024

Wprowadzenie do metod iteracyjnych

- Metody iteracyjne są alternatywą dla metod bezpośrednich w rozwiązywaniu układów równań liniowych.
- Umożliwiają aproksymację rozwiązania poprzez iteracyjne poprawianie przybliżenia początkowego.
- Są szczególnie przydatne dla dużych, rzadkich układów równań, gdzie metody bezpośrednie mogą być nieefektywne z powodu wymagań pamięciowych lub obliczeniowych.

Kiedy stosować metody iteracyjne?

- Gdy układ równań jest zbyt duży dla metod bezpośrednich.
- W przypadkach, gdy macierz systemu jest rzadka i dobrze uwarunkowana.
- Gdy potrzebne jest szybkie znalezienie przybliżonego rozwiązania, a niekoniecznie dokładnego.
- W środowiskach obliczeniowych o ograniczonych zasobach pamięciowych.

Przykłady metod iteracyjnych

- Metoda Jacobiego
- Metoda Gaussa-Seidla
- Metoda sukcesywnych nadrelaksacji (SOR)
- Metoda gradientów sprzężonych (dla macierzy symetrycznych i dodatnio określonych)

Ogólny schemat metod iteracyjnych

1. Wybierz początkowe przybliżenie rozwiązania $x^{(0)}$.
2. Powtarzaj następujące kroki do spełnienia kryterium zbieżności:
   - Uaktualnij przybliżenie rozwiązania stosując wybraną metodę iteracyjną.
   - Sprawdź kryterium zbieżności (np. różnicę między kolejnymi przybliżeniami).
3. Zakończ, gdy rozwiązanie jest wystarczająco dokładne.

Wprowadzenie do metod iteracyjnych

W przybliżonych metodach rozwiązywania układu równań $Ax = b$, zawierającego n niewiadomych, poszukiwanie rozwiązania x rozpoczyna się od zastosowania pewnego rozwiązania początkowego $x^{(0)}$, generując ciąg wektorów $\{x^{(k)}\}_{k=0}^{\infty}$, zbieżny z x.

Przekształcenie układu równań

W większości metod iteracyjnych układ równań $Ax = b$ zamieniany jest na ekwiwalentny układ równań

$$x = Wx + Z \quad (1)$$

gdzie W jest macierzą $n \times n$, natomiast Z jest wektorem.

Iteracyjne poszukiwanie rozwiązania

Mając dany wektor początkowy $x^{(0)}$, iteracyjne poszukiwanie rozwiązania można zapisać w postaci

$$x^{(k)} = Wx^{(k-1)} + Z \quad k = 1, 2, 3, \ldots \qquad (2)$$

Metoda Jacobiego - Wstęp

W metodzie Jacobiego wykorzystujemy wzory (1) i (2)

$$x = Wx + Z$$

$$x^{(k)} = Wx^{(k-1)} + Z \quad k = 1, 2, 3, \ldots$$

Należy tylko określić, co to są macierze W i Z.

Przekształcenie układu do postaci iteracyjnej

Dany jest układ równań: $Ax = b$
Macierz główna układu A rozbijamy na sumę dwóch macierzy $D + R$:

$$A = D + R$$

gdzie D to macierz diagonalna, a R to reszta elementów macierzy A.

$$\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix} = \begin{bmatrix} a_{11} & 0 & \cdots & 0 \\ 0 & a_{22} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & a_{nn} \end{bmatrix} + \begin{bmatrix} 0 & a_{12} & \cdots & a_{1n} \\ a_{21} & 0 & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & 0 \end{bmatrix}$$

Przekształcenie układu do postaci iteracyjnej

W literaturze spotykamy też podział macierzy A na trzy macierze:
$A = D + L + U$
gdzie:

- D - macierz diagonalna,
- L - macierz dolnotrójkątna z zerami na diagonali,
- U - macierz górnotrójkątna z zerami na diagonali.

$$A = \begin{bmatrix} a_{11} & 0 & \cdots & 0 \\ 0 & a_{22} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & a_{nn} \end{bmatrix} + \begin{bmatrix} 0 & 0 & \cdots & 0 \\ a_{21} & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & 0 \end{bmatrix} + \begin{bmatrix} 0 & a_{12} & \cdots & a_{1n} \\ 0 & 0 & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 \end{bmatrix}$$

Przekształcenie układu do postaci iteracyjnej

Mamy serię następujących przekształceń:

$$Ax = b$$

$$(D + R)x = b$$

$$Dx + Rx = b$$

Przenosimy na prawą stronę:

$$Dx = -Rx + b$$

Mnożymy obustronnie przez macierz D^{-1}:

$$D^{-1}Dx = -D^{-1}Rx + D^{-1}b$$

$$x = -D^{-1}Rx + D^{-1}b \quad (3)$$

Przekształcenie układu do postaci iteracyjnej

W równaniu (3):

$$x = -D^{-1}Rx + D^{-1}b$$

wprowadzamy macierz W i wektor Z następująco:

$$W = -D^{-1}R$$

$$Z = D^{-1}b$$

Ostatecznie otrzymujemy wprowadzone wcześniej równania (1) i (2)

$$x = Wx + Z$$

$$x^{(k)} = Wx^{(k-1)} + Z \quad k = 1, 2, 3, \ldots$$

Odwrotność macierzy diagonalnej D

Macierz odwrotna do macierzy diagonalnej D też jest macierzą diagonalną, główna przekątna tworzy odwrotności elementów macierzy D:

$$D^{-1} = \begin{bmatrix} \frac{1}{d_{11}} & 0 & \cdots & 0 \\ 0 & \frac{1}{d_{22}} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \frac{1}{d_{nn}} \end{bmatrix} = \begin{bmatrix} \frac{1}{a_{11}} & 0 & \cdots & 0 \\ 0 & \frac{1}{a_{22}} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \frac{1}{a_{nn}} \end{bmatrix}$$

W naszym przypadku zastąpienia macierzy A sumą macierzy $D + R$, na głównej przekątnej macierzy D^{-1} będą odwrotności elementów z głównej przekątnej macierzy A.

Wyznaczenie wartości macierzy W

Dla macierzy W mamy:

$$W = -D^{-1}R$$

gdzie D^{-1} jest macierzą odwrotną do macierzy diagonalnej D, a R reprezentuje pozostałą część macierzy A po odjęciu D. Konkretnie:

$$D^{-1} = \begin{bmatrix} \frac{1}{a_{11}} & 0 & \cdots & 0 \\ 0 & \frac{1}{a_{22}} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \frac{1}{a_{nn}} \end{bmatrix}$$

$$R = \begin{bmatrix} 0 & a_{12} & \cdots & a_{1n} \\ a_{21} & 0 & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & 0 \end{bmatrix}$$

Wyznaczenie wartości macierzy W

Stąd:

$$W = -D^{-1}R = \begin{bmatrix} 0 & -\frac{a_{12}}{a_{11}} & \dots & -\frac{a_{1n}}{a_{11}} \\ -\frac{a_{21}}{a_{22}} & 0 & \dots & -\frac{a_{2n}}{a_{22}} \\ \vdots & \vdots & \ddots & \vdots \\ -\frac{a_{n1}}{a_{nn}} & -\frac{a_{n2}}{a_{nn}} & \dots & 0 \end{bmatrix}$$

Elementy macierzy W będą następujące:

$$W_{ij} = -\frac{a_{ij}}{a_{ii}}; \quad i,j = 1,2,\dots,n \quad \text{and} \quad j \neq i$$

Wyznaczenie wartości wektora Z

Wyznaczenie wartości wektora Z wykorzystuje macierz odwrotną do macierzy diagonalnej D i wektor b. Obliczenie Z przedstawia się następująco:

$$Z = D^{-1}b = \begin{bmatrix} \frac{1}{a_{11}} & 0 & \cdots & 0 \\ 0 & \frac{1}{a_{22}} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \frac{1}{a_{nn}} \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix} = \begin{bmatrix} \frac{b_1}{a_{11}} \\ \frac{b_2}{a_{22}} \\ \vdots \\ \frac{b_n}{a_{nn}} \end{bmatrix}$$

Elementy wektora Z będą następujące:

$$Z_i = \frac{b_i}{a_{ii}}; \quad i = 1, 2, \ldots, n$$

Metoda Jacobiego - zapis macierzowy

Równanie iteracyjne w postaci ogólnej: $x = Wx + Z$ zapisane macierzowo prezentuje się następująco:

$$x = \begin{bmatrix} 0 & -\frac{a_{12}}{a_{11}} & \dots & -\frac{a_{1n}}{a_{11}} \\ -\frac{a_{21}}{a_{22}} & 0 & \dots & -\frac{a_{2n}}{a_{22}} \\ \vdots & \vdots & \ddots & \vdots \\ -\frac{a_{n1}}{a_{nn}} & -\frac{a_{n2}}{a_{nn}} & \dots & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} + \begin{bmatrix} \frac{b_1}{a_{11}} \\ \frac{b_2}{a_{22}} \\ \vdots \\ \frac{b_n}{a_{nn}} \end{bmatrix}$$

Zapis dla poszczególnych niewiadomych

W przypadku, gdy każda z wartości $x_i^{(k)}$ obliczana jest na podstawie wektora $x^{(k-1)}$ dla $k \geqslant 1$:

$$x_i^{(k)} = \sum_{\substack{j=1 \\ j \neq i}}^{n} \left(-\frac{a_{ij} x_j^{(k-1)}}{a_{ii}} \right) + \frac{b_i}{a_{ii}}; \quad i = 1, 2, \ldots, n \quad (5)$$

W praktyce równanie (2) jest stosowane w rozwazaniach teoretycznych, natomiast do obliczen uzywane jest równanie (5)

Zapis metody przy podziale na macierze D, L i U

W równaniu $Ax = b$ macierz współczynników A można zapisać jako sumę macierzy diagonalnej D, trójkątnej dolnej L i trójkątnej górnej U:

$$A = D + L + U$$

Stąd, równanie wyjściowe $Ax = b$ może przyjąć postać:

$$Dx = -(L + U)x + b$$

i ostatecznie:

$$x = -D^{-1}(L + U)x + D^{-1}b$$

Postać przystosowana do metody iteracyjnej Jacobiego jest następująca:

$$x^{(k)} = -D^{-1}(L + U)x^{(k-1)} + D^{-1}b; \quad k = 1, 2, \ldots \quad (6)$$

Badanie zbieżności - liczenie norm macierzy W

Nie dla każdej macierzy W kolejne iteracje będą zbieżne.
Zbieżność macierzy W trzeba sprawdzić.

Poprawne rozwiązanie układu metoda iteracyjna uzyskamy, jeśli największa co do wartości modułu wartość własna macierzy W jest mniejsza od jedności.

Spełnienie tego warunku wymaga, aby którakolwiek z norm macierzy W była mniejsza niż 1.

$$\|W\| < 1$$

Badanie zbieżności - liczenie norm macierzy W

Normę macierzy można definiować na różne sposoby:

Norma dla wierszy:

$$\|W\|_\infty = \max_i \sum_{j=1}^{n} |w_{ij}|$$

Norma dla kolumn:

$$\|W\|_1 = \max_j \sum_{i=1}^{n} |w_{ij}|$$

Norma Frobeniusa:

$$\|W\|_F = \sqrt{\sum_{i=1}^{n} \sum_{j=1}^{n} w_{ij}^2}$$

Metoda Jacobiego - przykład 1

Stosując technikę iteracyjną rozwiązać układ równań:

$$
\begin{aligned}
E1 &: 4x_1 - 2x_2 = 0 \\
E2 &: -2x_1 + 5x_2 - x_3 = 2 \\
E3 &: -x_2 + 4x_3 + 2x_4 = 3 \\
E4 &: 2x_3 + 3x_4 = -2
\end{aligned}
$$

którego rozwiązanie dokładne wynosi:

$$
x = \begin{pmatrix} 0.5 \\ 1 \\ 2 \\ -2 \end{pmatrix}
$$

Metoda Jacobiego - przykład 1

Przekształcamy ten układ równań do postaci $x = Wx + Z$, obliczając z każdego równania E_i niewiadomą $x_i (i = 1, 2, 3, 4)$. Otrzymujemy:

$$
\begin{aligned}
x_1 &= \frac{1}{2}x_2 \\
x_2 &= \frac{2}{5} + \frac{1}{5}x_1 + \frac{1}{5}x_3 \\
x_3 &= \frac{3}{4} - \frac{1}{4}x_2 + \frac{2}{4}x_4 \\
x_4 &= -\frac{2}{3} - \frac{2}{3}x_3
\end{aligned}
$$

Metoda Jacobiego - przykład 1

gdzie:

$$W = \begin{bmatrix} 0 & \frac{2}{4} & 0 & 0 \\ \frac{2}{5} & 0 & \frac{1}{5} & 0 \\ 0 & \frac{1}{4} & 0 & -\frac{2}{4} \\ 0 & 0 & -\frac{2}{3} & 0 \end{bmatrix}, Z = \begin{bmatrix} 0 \\ \frac{2}{5} \\ \frac{3}{4} \\ -\frac{2}{3} \end{bmatrix}$$

Sprawdzamy zbieżność (norma dla wierszy):

$$\|W\| = \max_i \sum_{j=1}^{n} |w_{ij}| = \max\left(\frac{2}{4}, \frac{3}{5}, \frac{3}{4}, \frac{2}{3}\right) = \frac{3}{4}$$

Metoda Jacobiego - przykład 1

Stosując równanie $x^{(k)} = Wx^{(k-1)} + Z$ i przyjmując rozwiązanie początkowe $x^{(0)} = (0, 0, 0, 0)^T$, możemy zapisać kolejne iteracje:

$$
\begin{aligned}
x_1^{(k)} &= \frac{1}{4} x_2^{(k-1)} \\
x_2^{(k)} &= \frac{2}{5} + \frac{1}{5} x_1^{(k-1)} + \frac{1}{5} x_3^{(k-1)} \\
x_3^{(k)} &= \frac{1}{4} x_2^{(k-1)} - \frac{2}{4} x_4^{(k-1)} + \frac{3}{4} \\
x_4^{(k)} &= -\frac{2}{3} x_3^{(k-1)} - \frac{2}{3}
\end{aligned}
$$

gdzie $x_i^{(k)}$ oznacza wartość i-tej niewiadomej w k-tej iteracji, a $x_i^{(k-1)}$ oznacza wartość i-tej niewiadomej w poprzedniej iteracji. Przyjmując dokładność obliczeń $\epsilon = 10^{-3}$, w kolejnych iteracjach otrzymuje się następujące wartości:

Wyniki iteracji metody Jacobiego

i	x_1	x_2	x_3	x_4	błąd
0	0.0000	0.4000	0.7500	-0.6667	0.7500
1	0.2000	0.5500	1.1833	-1.1667	0.5000
2	0.2750	0.7167	1.4708	-1.4556	0.2889
3	0.3583	0.8042	1.6569	-1.6472	0.1917
4	0.4021	0.8747	1.7747	-1.7713	0.1241
5	0.4374	0.9158	1.8543	-1.8498	0.0797
6	0.4579	0.9458	1.9038	-1.9029	0.0531
7	0.4729	0.9639	1.9379	-1.9359	0.0341
8	0.4820	0.9767	1.9589	-1.9586	0.0227
9	0.4884	0.9846	1.9735	-1.9726	0.0146
10	0.4923	0.9900	1.9824	-1.9823	0.0097
11	0.4950	0.9934	1.9887	-1.9883	0.0062
12	0.4967	0.9957	1.9925	-1.9924	0.0041
13	0.4979	0.9972	1.9952	-1.9950	0.0027
14	0.4986	0.9982	1.9968	-1.9968	0.0018
15	0.4991	0.9988	1.9979	-1.9979	0.0011
16	0.4994	0.9992	1.9986	-1.9986	0.0008

Warunek stopu

Zakończenie obliczeń może nastąpić po spełnieniu warunku stopu, na przykład gdy stosunek normy różnicy kolejnych przybliżeń do normy bieżącego przybliżenia jest mniejszy od zadanej tolerancji ϵ:

$$\frac{\|x^{(k)} - x^{(k-1)}\|}{\|x^{(k)}\|} \leqslant \epsilon$$

W naszym przykładzie mamy:

$$\frac{\|x^{(k)} - x^{(k-1)}\|}{\|x^{(k)}\|} = \frac{1.9986 - 1.9979}{1.9986} \approx 0.00035 \leqslant 10^{-3}$$

Korzystamy tutaj z normy maksimum dla oceny zbieżności:

$$\|W\|_{\max} = \max_{ij} |w_{ij}|$$

U nas dla macierzy A mamy:

$$\|A\|_{\max} = \max_{ij} |a_{ij}|$$

Metoda Jacobiego - podsumowanie

- Prosty algorytm Jacobiego rozwiązania układu równań metodą Jacobiego wymaga, aby $a_{ii} \neq 0$ dla każdego $i = 1, 2, \ldots, n$.
- Jeżeli warunek ten nie jest spełniony, a układ równań jest nieosobliwy, to układ równań można tak przekształcić, by warunek $a_{ii} \neq 0$ dla każdego $i = 1, 2, \ldots, n$ był spełniony.
- Przyspieszenie zbieżności otrzymuje się w ten sposób, że jako a_{ii} wybiera się elementy największe co do ich wartości bezwzględnej.

Metoda Gaussa-Seidla

Metoda Gaussa-Seidla jest jedną z metod iteracyjnych rozwiązywania układów równań liniowych. Podobnie jak metoda Jacobiego, służy do znalezienia rozwiązania układu równań $Ax = b$, ale różni się sposobem aktualizacji wartości niewiadomych.

Ogólny schemat metody Gaussa-Seidla

Metoda Gaussa-Seidla aktualizuje wartości niewiadomych sekwencyjnie, wykorzystując już zaktualizowane wartości:

$$x_i^{(k)} = \frac{1}{a_{ii}} \left(b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k)} - \sum_{j=i+1}^{n} a_{ij} x_j^{(k-1)} \right),$$

dla $i = 1, 2, \ldots, n$.

Zalety i warunki stosowania metody Gaussa-Seidla

Metoda Gaussa-Seidla często zbiega szybciej niż metoda Jacobiego, szczególnie dla macierzy dobrze uwarunkowanych i dominującej przekątnej.
Stosuje się ją, gdy:

- Macierz systemu jest duża i rzadka.
- Preferowana jest szybsza konwergencja kosztem większej złożoności obliczeniowej pojedynczej iteracji.
- Macierz spełnia warunki zbieżności, np. jest silnie diagonalnie dominująca.

Metoda Gaussa-Seidla - przedstawienie w zapisie macierzowym

W celu przedstawienia tej metody w zapisie macierzowym, pomnóżmy każde równanie (7) przez a_{ii}. Mamy wtedy następujące zależności dla $i = 1, 2, \ldots, n$:

$$a_{i1}x_1^{(k)} + a_{i2}x_2^{(k)} + \cdots + a_{ii}x_i^{(k)} = -a_{i,i+1}x_{i+1}^{(k-1)} - \cdots - a_{in}x_n^{(k-1)} + b_i;$$

Dla wszystkich n równań otrzymujemy:

$$\begin{aligned} a_{11}x_1^{(k)} &= -a_{12}x_2^{(k-1)} - a_{13}x_3^{(k-1)} - \cdots - a_{1n}x_n^{(k-1)} + b_1 \\ a_{21}x_1^{(k)} + a_{22}x_2^{(k)} &= -a_{23}x_3^{(k-1)} - \cdots - a_{2n}x_n^{(k-1)} + b_2 \\ &\vdots \\ a_{n1}x_1^{(k)} + a_{n2}x_2^{(k)} + \cdots + a_{nn}x_n^{(k)} &= b_n \end{aligned}$$

Metoda Gaussa-Seidla - przedstawienie w zapisie macierzowym

lub w formie skróconej:

$$(L + D)x^{(k)} = -Ux^{(k-1)} + b$$

gdzie:

- L jest macierzą trójkątną dolną,
- D jest macierzą diagonalną,
- U jest macierzą trójkątną górną.

Przekształcenia równań metody Gaussa-Seidla

Po przekształceniach otrzymujemy równanie iteracyjne w postaci:

$$x^{(k)} = -(L+D)^{-1}Ux^{(k-1)} + (L+D)^{-1}b; \quad k = 1, 2, \ldots$$

Aby macierz trójkątna dolna $D - L$ była nieosobliwa, wymagane jest, aby:

$$a_{ii} \neq 0 \quad \text{dla wszystkich} \quad i = 1, 2, \ldots, n.$$

Metoda Gaussa-Seidla - Przykład 3

Rozwiązanie układu równań z przykładu 1, stosując wzór (7). Dla układu równań:

$$
\begin{aligned}
E1 &: 4x_1 - 2x_2 = 0 \\
E2 &: -2x_1 + 5x_2 - x_3 = 2 \\
E3 &: -x_2 + 4x_3 + 2x_4 = 3 \\
E4 &: 2x_3 + 3x_4 = -2
\end{aligned}
$$

Wzór (7) dla metody Gaussa-Seidla

Stosując wzór (7) dla powyższego układu równań, otrzymujemy kolejne iteracje:

$$
\begin{aligned}
x_1^{(k)} &= \frac{1}{2} x_2^{(k-1)} \\
x_2^{(k)} &= \frac{2 + 2x_1^{(k)} - \frac{1}{5} x_3^{(k-1)}}{5} \\
x_3^{(k)} &= \frac{3 + x_2^{(k)} + \frac{2}{4} x_4^{(k-1)}}{4} \\
x_4^{(k)} &= \frac{-2 - 2x_3^{(k)}}{3}
\end{aligned}
$$

Wyniki iteracji metody Gaussa-Seidla

i	x_1	x_2	x_3	x_4	błąd
0	0.0000	0.4000	0.8500	-1.2333	1.2333
1	0.2000	0.6500	1.5292	-1.6861	0.6792
2	0.3250	0.8358	1.8020	-1.8680	0.2728
3	0.4179	0.9276	1.9159	-1.9439	0.1139
4	0.4638	0.9687	1.9641	-1.9761	0.0482
5	0.4843	0.9866	1.9847	-1.9898	0.0206
6	0.4933	0.9943	1.9935	-1.9956	0.0089
7	0.4971	0.9975	1.9972	-1.9981	0.0038
8	0.4988	0.9989	1.9988	-1.9992	0.0016
9	0.4995	0.9996	1.9995	-1.9997	0.0007

Kryterium zakończenia obliczeń

Dla $\epsilon = 10^{-3}$ zakończenie obliczeń nastąpiło po 9 iteracjach, ponieważ

$$\frac{\|x^{(k)} - x^{(k-1)}\|}{\|x^{(k)}\|} = \frac{1.9997 - 1.9992}{1.9997} \approx 0.00025 \leqslant 10^{-3}$$

Tutaj również korzystamy z normy maksimum:

$$\|A\|_{\max} = \max_{ij} |a_{ij}|$$

Podsumowanie metody Gaussa-Seidla

Metoda Gaussa-Seidla, dzięki swojej iteracyjnej naturze i sposobowi aktualizacji wartości niewiadomych, może być bardziej efektywna od metody Jacobiego w wielu praktycznych zastosowaniach. Jej zastosowanie jest szczególnie korzystne w przypadku dużych układów równań z macierzami spełniającymi odpowiednie warunki zbieżności.

Metody numeryczne
Wykład nr 5
Równania nieliniowe

Aneta Wróblewska

UMCS, Lublin

April 11, 2024

Wprowadzenie

Równania nieliniowe są fundamentem wielu zagadnień naukowych i inżynierskich. Rozwiązania tych równań często nie mogą być wyrażone przez proste wzory analityczne, co prowadzi do potrzeby stosowania metod numerycznych.

Charakterystyka równań nieliniowych

Równania nieliniowe charakteryzują się tym, że zależność między zmiennymi nie jest liniowa, co oznacza, że zmiana jednej zmiennej nie prowadzi do proporcjonalnej zmiany drugiej zmiennej. Takie zachowanie wprowadza dodatkową złożoność do analizy i rozwiązania problemu.

Rozwiązanie równania

Poszukiwanie miejsc zerowych funkcji to zagadnienie znalezienia takiego x^*, że

$$f(x^*) = 0,$$

czyli rozwiązania równania.

Zakładamy, że nie jesteśmy w stanie znaleźć miejsca zerowego analitycznie - pozostają nam metody iteracyjne.

Problemy przy rozwiązywaniu równań nieliniowych

- Dobór odpowiedniego punktu startowego może znacząco wpłynąć na zbieżność metody i szybkość odnalezienia rozwiązania.
- Niektóre metody mogą nie zbiegać do rozwiązania lub zbiegać bardzo wolno, jeśli nie są spełnione pewne warunki.
- W przypadku wielu pierwiastków metoda może konwergować do różnych rozwiązań w zależności od punktu startowego.

Podstawowe metody rozwiązywania równań nieliniowych

- Metoda bisekcji - metoda podziału przedziału na pół, wykorzystująca własność ciągłości funkcji.
- Metoda Newtona (stycznych) - wykorzystuje pochodne funkcji do szybkiego odnalezienia pierwiastka.
- Metoda siecznych - podobna do metody Newtona, ale nie wymaga obliczania pochodnej funkcji.

Przykłady równań nieliniowych

Równania nieliniowe mogą przybierać różne formy, oto kilka przykładów:

1. Równania wielomianowe, np. $x^3 - 6x^2 + 11x - 6 = 0$.
2. Równania trygonometryczne, np. $\sin(x) + x^2 - 1 = 0$.
3. Równania eksponencjalne, np. $e^x - x - 2 = 0$.
4. Równania logarytmiczne, np. $\ln(x) + x^2 - 3 = 0$.
5. Równania zawierające funkcje specjalne, np. $J_0(x) + x^2 - 2 = 0$, gdzie $J_0(x)$ jest funkcją Bessela pierwszego rodzaju zerowego rzędu.

Różnorodność równań nieliniowych ukazuje złożoność problemu ich rozwiązywania oraz potrzebę stosowania specjalistycznych metod numerycznych.

Ważne twierdzenia

Twierdzenie Darboux

Jeżeli funkcja $f(x)$ jest ciągła w przedziale domkniętym $[a, b]$, zaś u jest liczbą z przedziału $[f(a), f(b)]$ to istnieje c z przedziału $[a, b]$, że $u = f(c)$.

Twierdzenie Bolzano-Cauchy'ego

Jeżeli funkcja $f(x)$ jest ciągła w przedziale domkniętym $[a, b]$ i $f(a) \cdot f(b) < 0$, to między punktami a i b znajduje się co najmniej jeden pierwiastek równania $f(x) = 0$.

Twierdzenie

Jeżeli w przedziale $[a, b]$ spełnione są założenia twierdzenia Bolzano-Cauchy'ego i dodatkowo $sgnf'(x) = const$ dla $x \in [a, b]$, to przedział ten jest przedziałem izolacji pierwiastka równania $f(x) = 0$.

Znaczenie twierdzeń w metodach rozwiązywania równań nieliniowych

Powyższe twierdzenia mają kluczowe znaczenie przy rozwiązywaniu równań nieliniowych, ponieważ:

- pozwalają na weryfikację istnienia rozwiązania w danym przedziale,
- pomagają określić przedział izolacji pierwiastka, co jest ważne dla metody bisekcji oraz innych metod iteracyjnych,
- ułatwiają wybór odpowiedniego przedziału startowego dla metod numerycznych.

Przykłady zastosowania

Przykład 1:
Rozważmy funkcję $f(x) = x^2 - 4$ na przedziale $[1, 5]$. Ponieważ $f(1) \cdot f(5) < 0$, to na mocy twierdzenia Bolzano-Cauchy'ego istnieje co najmniej jeden pierwiastek równania $f(x) = 0$ w przedziale $[1, 5]$.

Przykład 2:
Funkcja $f(x) = \sin(x) - \frac{x}{2}$ w przedziale $[3, 4]$. Mamy $f(3) \cdot f(4) < 0$ oraz $f'(x) = \cos(x) - \frac{1}{2}$ nie zmienia znaku w $[3, 4]$, co wskazuje na istnienie dokładnie jednego pierwiastka w tym przedziale.

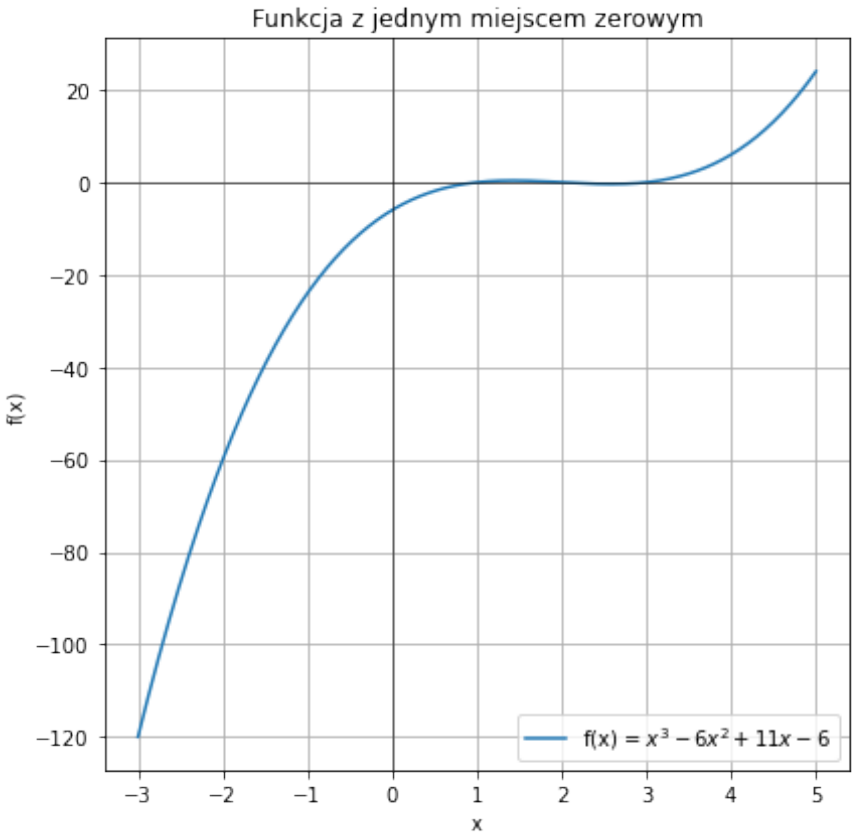
Funkcja signum

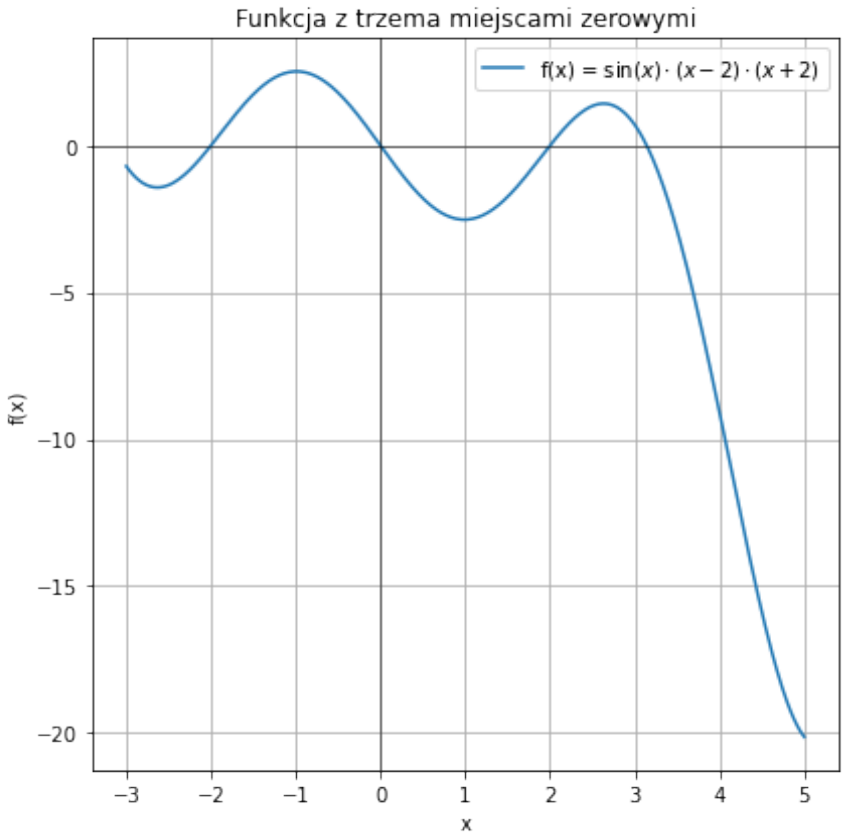
Funkcja signum, oznaczana jako *sgn*, jest funkcją zmiennej rzeczywistej zdefiniowaną w sposób następujący:

$$sgn(x) = \begin{cases} -1 & \text{dla } x < 0, \\ 0 & \text{dla } x = 0, \\ 1 & \text{dla } x > 0. \end{cases}$$

- Zwraca -1 dla wartości ujemnych argumentu.
- Zwraca 0 dla argumentu równego zero.
- Zwraca 1 dla wartości dodatnich argumentu.

Przykłady funkcji z 1 i 3 miejscami zerowymi





Charakterystyka pierwszej pochodnej

Pierwsza pochodna funkcji dostarcza cennych informacji o jej zachowaniu:

- Miejsca zerowe pochodnej wskazują potencjalne punkty ekstremalne funkcji.
- Dodatnie wartości pochodnej w pewnym przedziale sugerują, że funkcja w tym obszarze rośnie.
- Ujemne wartości pochodnej świadczą o malejącym charakterze funkcji w danym przedziale.
- Funkcja rosnąca przed punktem zerowym pochodnej, która następnie maleje, osiąga w tym punkcie maksimum lokalne.
- Funkcja malejąca, a następnie rosnąca po przekroczeniu miejsca zerowego pochodnej, osiąga minimum lokalne.

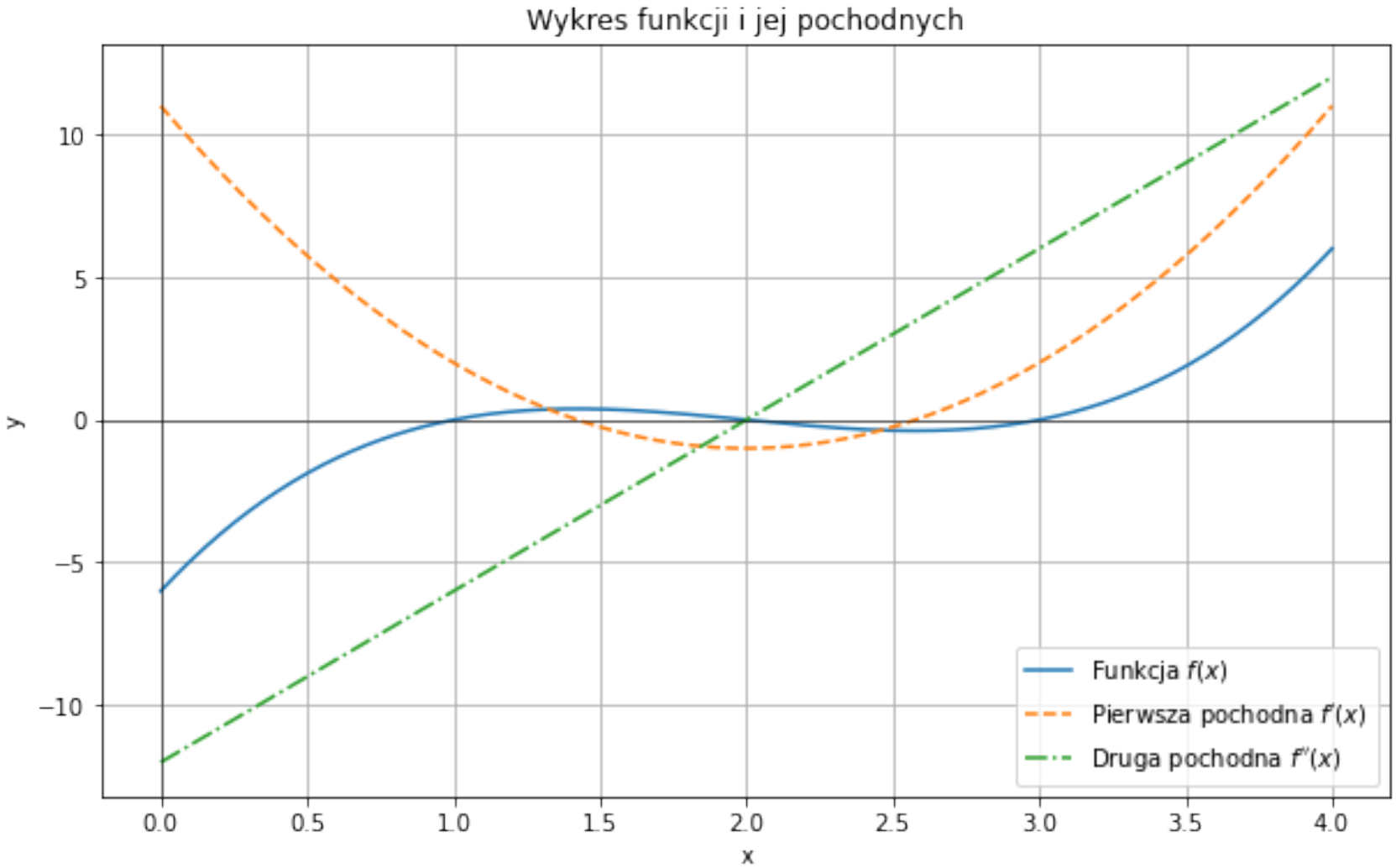
Charakterystyka drugiej pochodnej

Druga pochodna funkcji rzuca światło na jej krzywiznę:

- Punkty, w których druga pochodna przyjmuje wartość zero, mogą oznaczać punkty przegięcia funkcji.
- Dodatnia wartość drugiej pochodnej w określonym przedziale wskazuje, że funkcja jest wypukła w tym obszarze.
- Ujemne wartości drugiej pochodnej sygnalizują wklęsłość funkcji w danym przedziale.

Przykład wykresu funkcji i jej pochodnych

$$f(x) = x^3 - 6x^2 + 11x - 6$$



Metoda bisekcji

Metoda bisekcji

Metoda bisekcji jest techniką numeryczną służącą do znajdowania pierwiastka równania $f(x) = 0$ w określonym przedziale $[a, b]$. Rozpoczynamy od przyjęcia, że pierwsze dwie wartości ciągu to $x_1 = a$ i $x_2 = b$.

Obliczanie kolejnych przybliżeń

Dla każdego kroku iteracji $i = 3, 4, \ldots$, nową wartość x_i obliczamy jako średnią x_{i-1} i x_k:

$$x_i = \frac{x_{i-1} + x_k}{2}$$

gdzie k jest jedną z wartości $\{i-3, i-2\}$. Wybór k jest taki, aby spełnione były warunki:

- $|x_i - x_{i-1}| = |x_i - x_k|$
- $f(x_{i-1}) \cdot f(x_k) < 0$, co wskazuje na obecność pierwiastka w tym przedziale.

Zmniejszanie przedziału izolacji i szybkość iteracji

W każdej iteracji przedział izolacji pierwiastka jest redukowany o połowę, co prowadzi nas bliżej do poszukiwanego rozwiązania. Choć metoda ta jest niezawodna, to charakteryzuje się stosunkowo wolną konwergencją.

Kryteria zakończenia

Kryteria zakończenia iteracji metodą bisekcji:

- zadana liczba kroków - iteracji,
- dostatecznie mały błąd (problem, gdy funkcja jest płaska w pobliżu miejsca zerowego),
- wartość funkcji dostatecznie bliska zeru (problem, gdy funkcja jest nieciągła w pobliżu miejsca zerowego).

Wyznaczanie błędu metody bisekcji

Dokładność przybliżenia w i-tym kroku można określić jako:

$$|x_i - x^*| < \frac{b-a}{2^{i-2}}$$

co pokazuje, jak błąd zmniejsza się w miarę postępu iteracji. Oznacza to, że z każdym krokiem jesteśmy coraz bliżej pierwiastka równania.

Potencjalne problemy i praktyczne wskazówki

- metoda daje jedno miejsce zerowe, a nie wszystkie w przedziale $[a, b]$,
- błędy zaokrągleń powodują, że otrzymanie $f(x) = 0$ jest mało prawdopodobne, dlatego ta równość nie powinna stanowić kryterium zakończenia obliczeń,
- punkt środkowy lepiej obliczyć ze wzoru $a + \frac{b-a}{2}$ zamiast $\frac{a+b}{2}$ - w obliczeniach numerycznych lepiej jest obliczać nową wielkość, dodając do poprzedniej małą poprawkę,
- zmianę znaku wartości funkcji lepiej badać zaa pomocą nierówności $sgn(f(x_i)) \neq sgn(f(x_j))$ zamiast $f(x_i) \cdot f(x_j) < 0$ - unikamy zbędnego mnożenia.

Przykład metody bisekcji

Rozważamy funkcję kwadratową $f(x) = x^3 - 3x^2 - 2x + 5$ i szukamy pierwiastka tej funkcji w przedziale $[1, 2]$ przy użyciu metody bisekcji.

Krok 1

Początkowy przedział izolacji pierwiastka: $[a_1, b_1] = [1, 2]$

Punkt środkowy: $x_1 = \frac{1+2}{2} = 1.5$

$f(a_1) = f(1) = 1,$

$f(b_1) = f(2) = -3,$

$f(x_1) = f(1.5) = -1.375$

$f(1) \cdot f(1.5) < 0 \quad \longrightarrow$ kontynuujemy z przedziałem $[1, 1.5]$.

Przykład metody bisekcji

Krok 2

Nowy przedział: $[a_2, b_2] = [1, 1.5]$

Punkt środkowy: $x_2 = \frac{1+1.5}{2} = 1.25$

$f(x_2) = f(1.25) = -0.234$

$f(1) \cdot f(1.25) < 0 \quad \longrightarrow$ kontynuujemy z przedziałem $[1, 1.25]$.

Krok 3

Nowy przedział: $[a_3, b_3] = [1, 1.25]$

Punkt środkowy: $x_3 = \frac{1+1.25}{2} = 1.125$

$f(x_3) = f(1.125) = 0.376$

$f(1.125) \cdot f(1.25) < 0 \quad \longrightarrow$ kontynuujemy z przedziałem $[1.125, 1.25]$.

Przykład metody bisekcji

Krok 4

Nowy przedział: $[a_4, b_4] = [1.125, 1.25]$

Punkt środkowy: $x_4 = \frac{1.125+1.25}{2} = 1.1875$

$f(x_4) = f(1.1875) = 0.069$

$f(.1.875) \cdot f(1.25) < 0 \quad \longrightarrow$ kontynuujemy z przedziałem $[1.1875, 1.25]$.

Krok 5

Nowy przedział: $[a_5, b_5] = [1, 1.125]$

Punkt środkowy: $x_5 = \frac{1+1.125}{2} = 1.0625$

$f(x_5) = f(1.0625) = 0.152344$

$f(1) \cdot f(1.0625) < 0 \quad \longrightarrow$ kontynuujemy z przedziałem $[1, 1.0625]$.

Wyznaczanie błędu przybliżenia

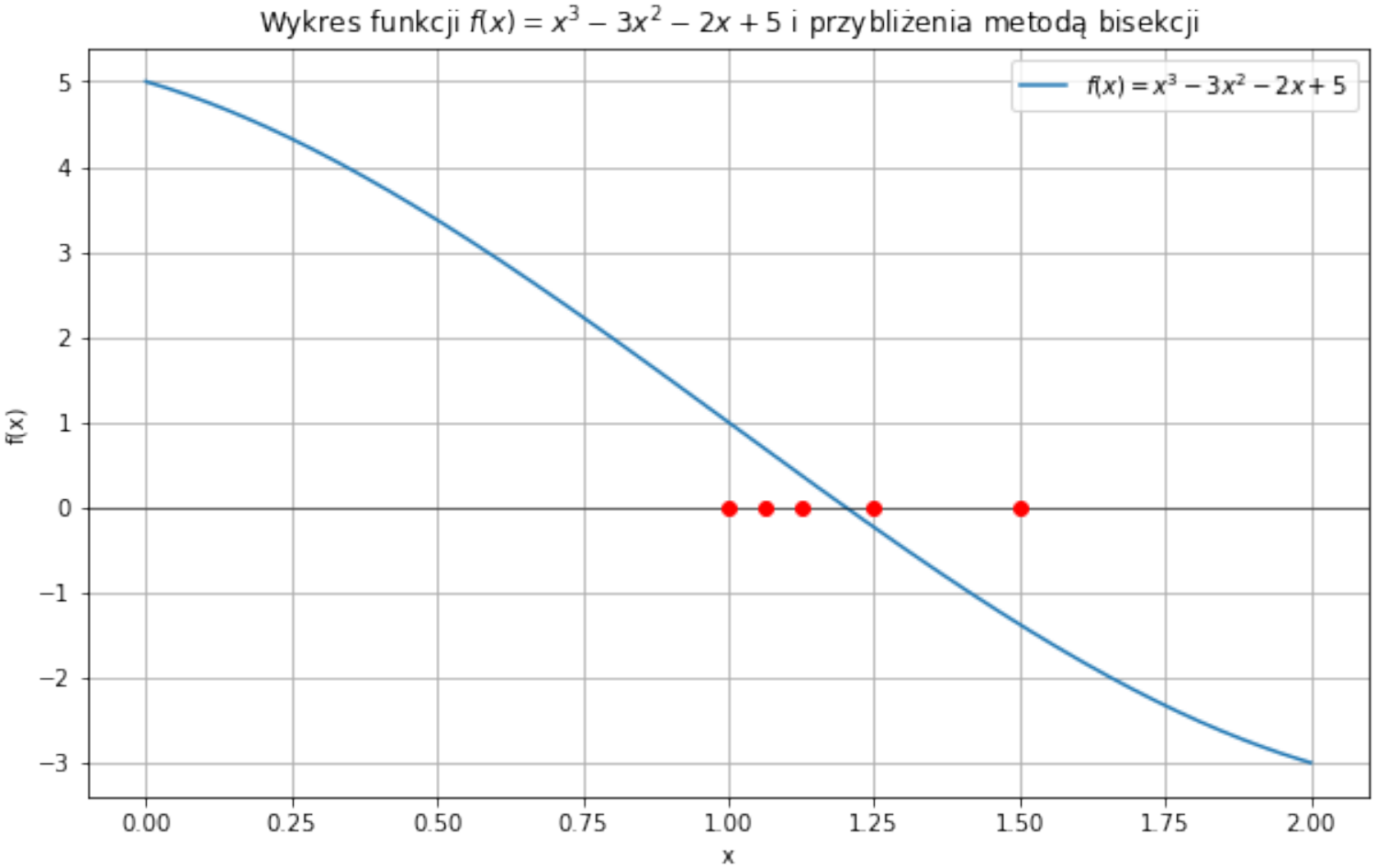
Analogicznie obliczamy kolejne kroki metody, aż do osiągnięcia zadanej dokładności.

Na przykład po 12 krokach metody bisekcji, dokładność przybliżenia możemy określić jako:

$$|x_{12} - x^*| < \frac{2-1}{2^{12-2}} = \frac{1}{2^{10}}$$

Metoda bisekcji - na wykresie

Wykres przedstawiający pięc kroków metody bisekcji dla funkcji $f(x) = x^3 - 3x^2 - 2x + 5$ na przedziale $[1, 2]$



Metoda stycznych (Newtona)

Metoda stycznych (Newtona)

Metoda stycznych, znana także jako **metoda Newtona**, polega na przybliżeniu pierwiastka równania $f(x) = 0$ w przedziale $[a, b]$ poprzez konstrukcję ciągu punktów będących miejscami zerowymi stycznych do krzywej funkcji $f(x)$.

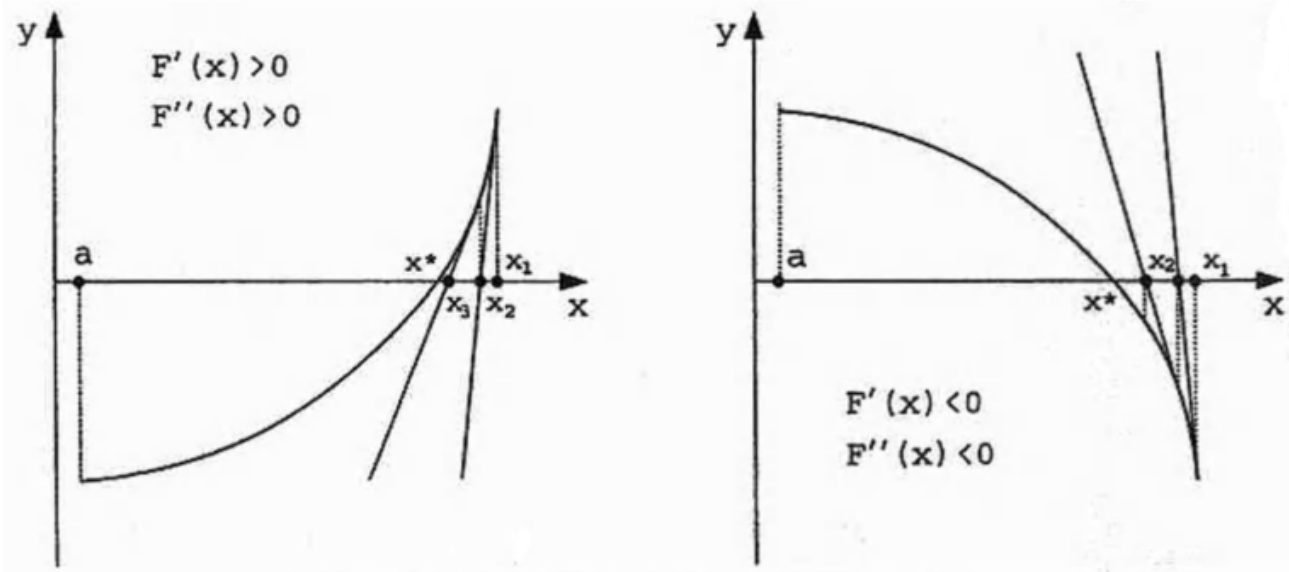
$$x_i = x_{i-1} - \frac{f(x_{i-1})}{f'(x_{i-1})}, \quad i = 2, 3, 4, \ldots$$

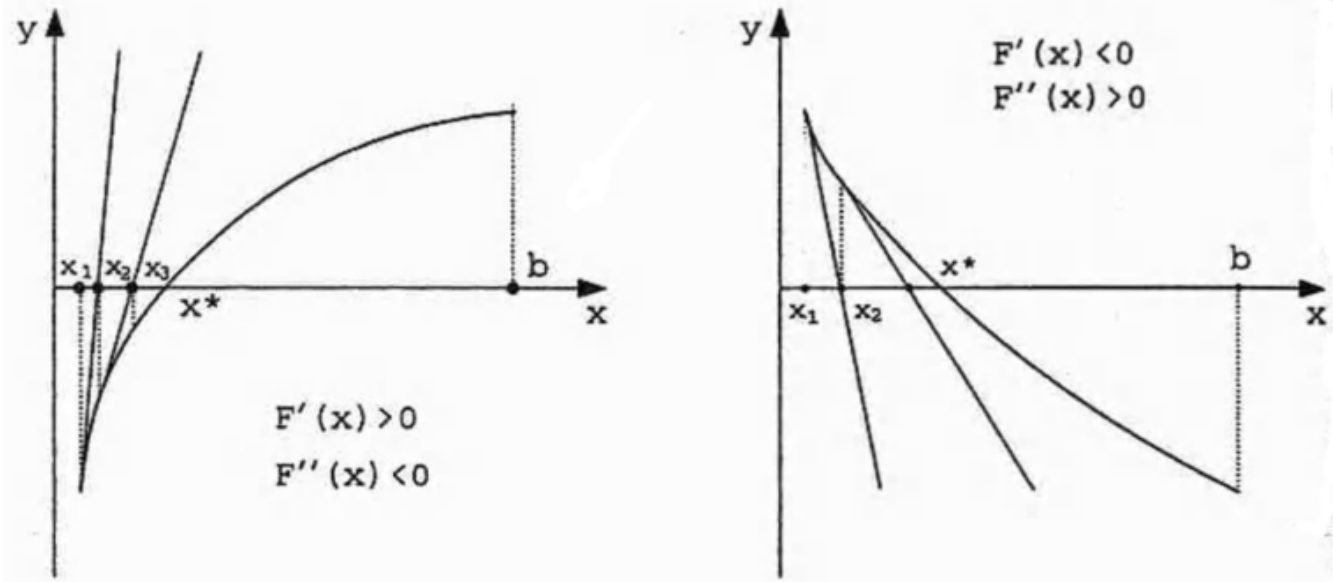
Wybór pierwszego przybliżenia x_1

Pierwsze przybliżenie x_1 często wybieramy spośród końców przedziału $[a, b]$. Kryterium wyboru między a a b stanowi zachowanie monotoniczności funkcji oraz jej krzywizna w rozważanym przedziale.

- Jeżeli dla $x \in [a, b]$, $f'(x) \cdot f''(x) < 0$, wtedy $x_1 = a$.
- Jeżeli $f'(x) \cdot f''(x) > 0$, wtedy $x_1 = b$.

Wybór pierwszego przybliżenia x_1





Oszacowanie błędu i zbieżność metody

Oszacowanie błędu

Błąd w każdym kroku iteracji możemy oszacować używając wzoru:

$$\Delta \approx |x_i - x_{i-1}|$$

Zbieżność metody

Metoda stycznych wykazuje szybką zbieżność (kwadratową, tzw. współczynnik zbieżności wynosi 2) pod warunkiem, że wybrane pierwsze przybliżenie jest wystarczająco blisko rzeczywistego pierwiastka. Metoda może być rozbieżna, jeżeli:

- pierwsze przybliżenie jest zbyt daleko od pierwiastka,
- funkcja $f(x)$ nie jest dostatecznie "gładka" w otoczeniu pierwiastka.

Warunki zakończenia obliczeń

Obliczenia kończymy, gdy osiągnięty zostanie zadany próg dokładności, np.:

$$|x_i - x_{i-1}| < \varepsilon$$

gdzie ε jest wcześniej zdefiniowanym progiem błędu.

Potencjalne problemy

Podczas stosowania metody stycznych można napotkać na problemy takie jak:

- rozbieżność iteracji przy niefortunnym wyborze x_1,
- zatrzymanie postępu w przypadku, gdy $f'(x_{i-1})$ jest bliskie zero,
- trudności z oszacowaniem globalnej dokładności przybliżenia.

Metoda stycznych - przykład

Analizujemy funkcję $f(x) = x^2 - 4x + 3$ z zamiarem zastosowania metody stycznych (Newtona) do znalezienia jej pierwiastka.
Pierwsze przybliżenie: $x_1 = 2$.
Pochodna funkcji: $f'(x) = 2x - 4$.

Krok 1

$x_1 = 2$

$f(x_1) = 2^2 - 4 \cdot 2 + 3 = -1$

$f'(x_1) = 2 \cdot 2 - 4 = 0$

$x_2 = x_1 - \frac{f(x_1)}{f'(x_1)} = 2 - \frac{-1}{0}$

Uwaga: W tym kroku nie można zastosować metody, ponieważ $f'(x_1) = 0$. Potrzebujemy wybrać inny punkt startowy.

Przykład cd - korekta i nowy punkt startowy

Krok 1

Początkowy punkt: $x_0 = 1.5$

$f(x_0) = (1.5)^2 - 4 \cdot 1.5 + 3 = -0.75$

$f'(x_0) = 2 \cdot 1.5 - 4 = -1$

$x_1 = x_0 - \frac{f(x_0)}{f'(x_0)} = 1.5 - \frac{-0.75}{-1} = 0.75$

Krok 2

$x_1 = 0.75$

$f(x_1) = (0.75)^2 - 4 \cdot 0.75 + 3 = 0.5625$

$f'(x_1) = 2 \cdot 0.75 - 4 = -2.5$

$x_2 = x_1 - \frac{f(x_1)}{f'(x_1)} = 0.75 - \frac{0.5625}{-2.5} = 0.975$

Przykład - dalsze kroki

Dalsze kroki obliczeniowe wykonujemy zgodnie z powyższą metodą otrzymując następujące wartości:

Iteracja (i)	Kolejne przybliżenie	Błąd przybliżenia
1	0.7500000000	0.7500000000
2	0.9750000000	0.2250000000
3	0.9996951220	0.0246951220
4	0.9999999535	0.0003048316
5	1.0000000000	0.0000000465

Metoda siecznych

Metoda siecznych

Metoda siecznych służy do aproksymacji pierwiastka równania $f(x) = 0$ w obrębie przedziału $[a, b]$. Jest to realizowane poprzez generowanie sekwencji miejsc zerowych dla cięciw rozpiętych między punktami $(x_i, f(x_i))$ i $(x_{i-1}, f(x_{i-1}))$, które określają granice poszczególnych przedziałów.

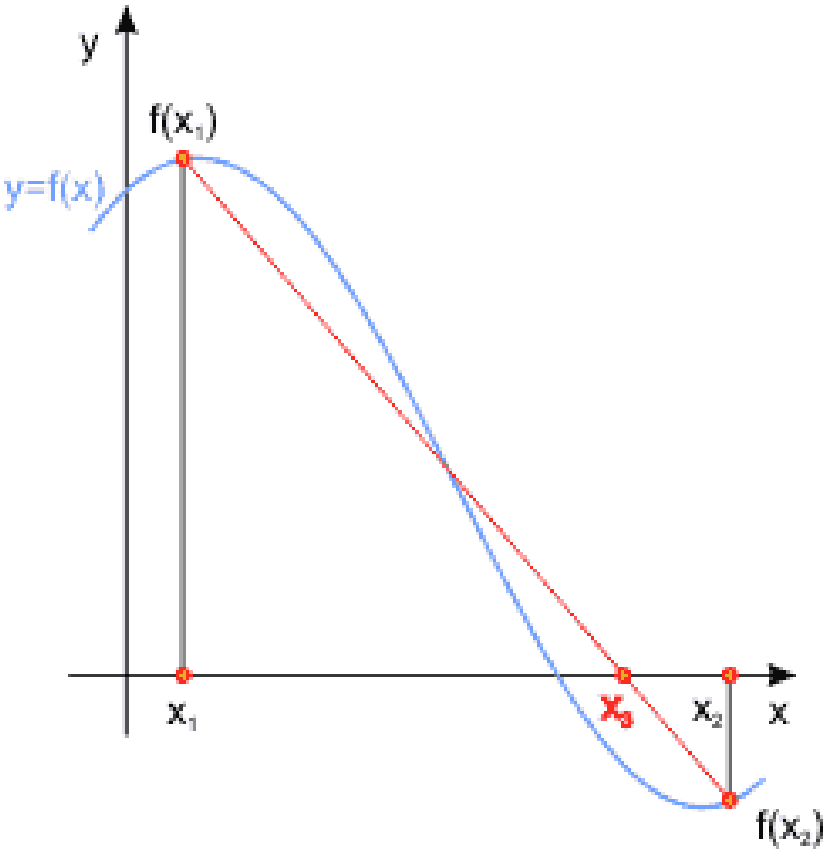
Wzór na kolejne przybliżenie pierwiastka prezentuje się następująco:

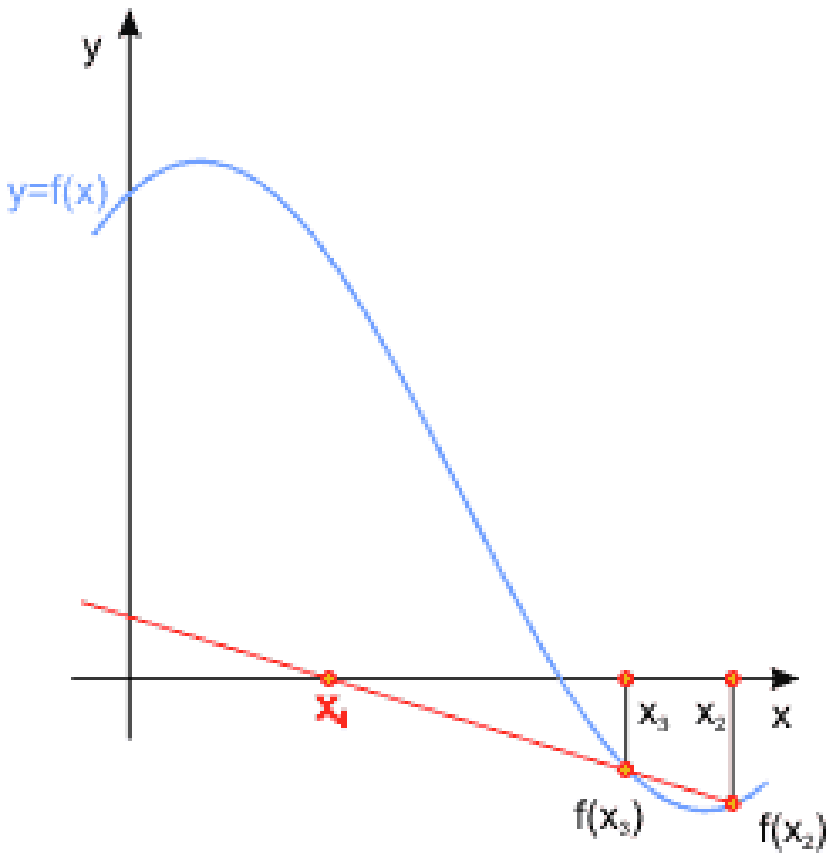
$$x_{i+1} = x_i - f(x_i) \cdot \frac{x_i - x_{i-1}}{f(x_i) - f(x_{i-1})}, \quad i = 2, 3, 4, \ldots$$

Błąd przybliżenia w każdym kroku obliczeniowym można oszacować za pomocą wzoru:

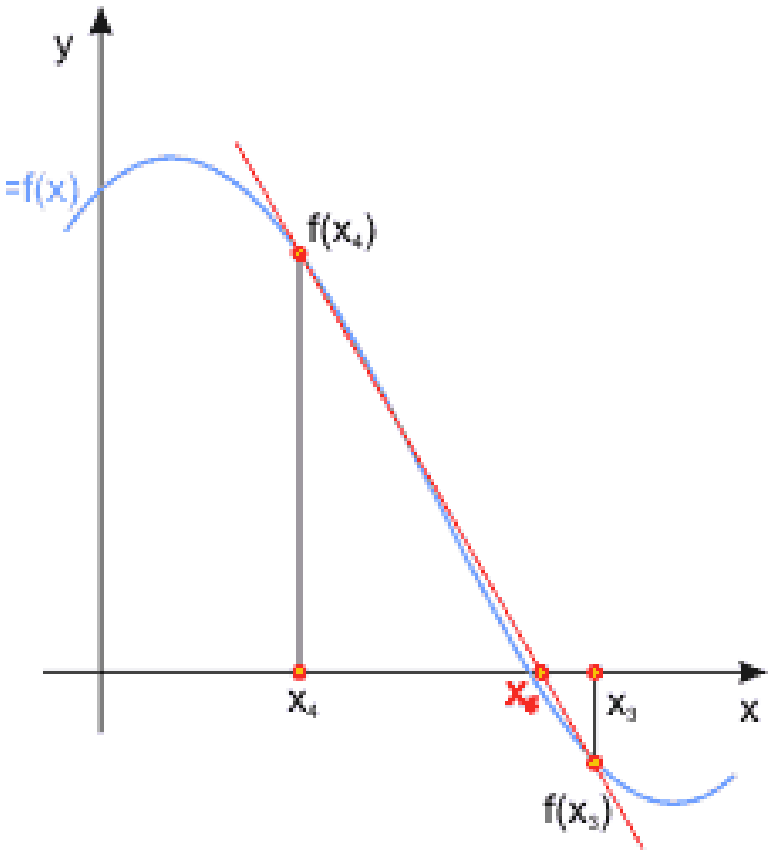
$$\Delta \approx |x_{i+1} - x_i|$$

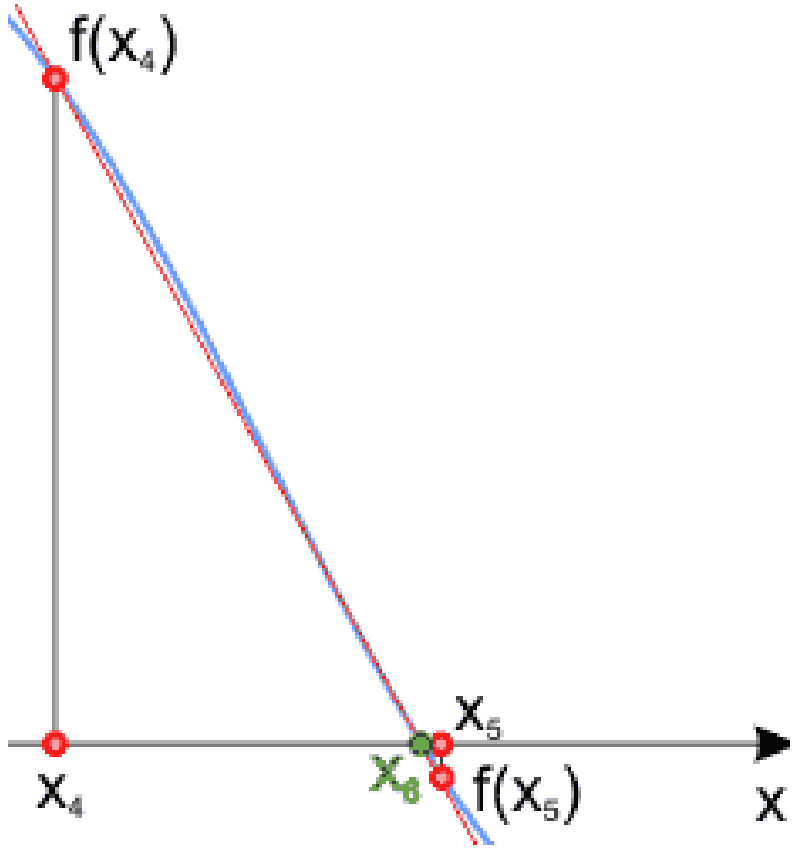
Metoda siecznych na wykresie





Metoda siecznych na wykresie - cd





Wybór punktów startowych - modyfikacja

Rozpatrujemy znaczenie znaku pierwszej i drugiej pochodnej funkcji w danym przedziale i jego wpływ na przybliżenie pierwiastka metody numeryczne.
W zależności od znaków pochodnych pierwszego i drugiego rzędu, możemy wyróżnić dwa główne scenariusze:

- Przybliżenie pierwiastka z niedomiarem,
- Przybliżenie pierwiastka z nadmiarem.

Wybór punktów startowych - modyfikacja

Przybliżenie z niedomiarem

Gdy $f'(x) > 0, f''(x) > 0$ lub $f'(x) < 0, f''(x) < 0$, kolejne przybliżenia pierwiastka układają się jak:

$$x_i < x_{i+1} < x_{i+2} < \ldots < x^*$$

Przybliżenie z nadmiarem

Gdy $f'(x) > 0, f''(x) < 0$ lub $f'(x) < 0, f''(x) > 0$, mamy do czynienia z sytuacją, gdzie:

$$x_i > x_{i+1} > x_{i+2} > \ldots > x^*$$

Biorąc powyższe własności pod uwagę, dla $x \in [a, b]$ punkty startowe x_2 i x_1 możemy wybrać zależnie od iloczynu pochodnych $f'(x) \cdot f''(x)$:

- Jeśli $f'(x) \cdot f''(x) < 0$, wówczas $x_2 = a$ i $x_1 = b$.
- Jeśli $f'(x) \cdot f''(x) > 0$, wówczas $x_2 = b$ i $x_1 = a$.

Metoda siecznych - przykład

Analizujemy funkcję $f(x) = x^2 - 4x + 3$ w przedziale $[0, 2]$ za pomocą metody siecznych, gdzie punkty startowe to $x_0 = 0$ i $x_1 = 2$.

Krok 1

$x_0 = 0, \quad x_1 = 2$

Obliczamy x_2:

$$x_2 = 2 - f(2)\frac{0-2}{f(0) - f(2)}$$

$$x_2 = 2 - (-1)\frac{0-2}{3-(-1)}$$

$$x_2 = 1.5$$

Błąd: $\Delta \approx |1.5 - 2| = 0.5$

Krok 2

W poprzednim kroku obliczyliśmy $x_2 = 1.5$. Teraz, użyjemy x_2 i x_1 do obliczenia x_3.

Obliczamy x_3:

$$x_3 = x_2 - f(x_2)\frac{x_1 - x_2}{f(x_1) - f(x_2)}$$

$$f(x_1) = 2^2 - 4 \cdot 2 + 3 = -1$$

$$f(x_2) = 1.5^2 - 4 \cdot 1.5 + 3 = -0.75$$

Stąd,

$$x_3 = 1.5 - (-0.75)\frac{2 - 1.5}{-1 - (-0.75)}$$

$$x_3 = 0$$

Błąd: $\Delta \approx |x_3 - x_2| = |0 - 1.5| = 1.5$

Wyniki iteracji metody siecznych

Iteracja (i)	Kolejne przybliżenie	Błąd przybliżenia
1	1.5000000000	0.5000000000
2	0.0000000000	1.5000000000
3	1.2000000000	1.2000000000
4	1.0714285714	0.1285714286
5	0.9917355372	0.0796930342
6	1.0003047851	0.0085692479
7	1.0000012545	0.0003035307
8	0.9999999998	0.0000012546
9	1.0000000000	0.0000000002
10	1.0000000000	0.0000000000

Table: Wyniki iteracji metody siecznych.

Metoda siecznych - podsumowanie i wnioski

- metoda siecznych ma tę zaletę, że do wykonywania operacji nie ma potrzeby wyliczać pochodnych,
- metoda siecznych może nie być zbieżna, gdy wybieżemy zbyt mały przedział $[a, b]$, np. przedział $[0.5, 1]$ dla funkcji $f(x) = 2x - x^3 - 1$ daje nam rozwiązania na przemian 0.5 i 1 – w takim przypadku trzeba zastosować metodę alternatywną,
- w ogólnym przypadku metoda siecznych wymaga większej liczby iteracji niż metoda stycznych (współczynnik zbieżności 1.62), jednak w niektórych przypadkach metoda stycznych może działać wolniej pomimo mniejszej liczby iteracji (zależy to od trudności w wyznaczeniu pochodnej).

Metoda regula falsi

Metoda regula falsi

Metoda regula falsi, znana również jako metoda fałszywej pozycji, jest metodą numeryczną rozwiązywania równań nieliniowych, która łączy zalety metody siecznych i metod przedziałowych. Podobnie jak metoda siecznych, polega na konstrukcji linii prostych łączących punkty na wykresie funkcji, ale zachowuje gwarancję zbieżności metod przedziałowych poprzez zachowanie przedziału izolacji pierwiastka.

Zasada działania metody regula falsi

Metoda działa na zasadzie wybierania końców przedziału $[a, b]$, tak aby wartości funkcji $f(a)$ i $f(b)$ miały przeciwne znaki, co wskazuje na istnienie pierwiastka w przedziale. Następnie, podobnie jak w metodzie siecznych, rysuje się cięciwę między punktami $(a, f(a))$ i $(b, f(b))$, a punkt przecięcia tej cięciwy z osią X stanowi nowe przybliżenie pierwiastka. W przeciwieństwie do metody siecznych, metoda Regula Falsi aktualizuje jeden z końców przedziału, zawsze utrzymując pierwiastek w obrębie przedziału, co zapewnia jej zbieżność.

- Metoda zawsze zbieżna, jeśli tylko dobrze wybrano przedział początkowy.
- Metoda zawiedzie, gdy f(x) styczne z osią x dla f(x)=0.
- Wolna zbieżność p=1 (metoda liniowa)

Metody numeryczne
Wykład nr 6
Interpolacja

Aneta Wróblewska

UMCS, Lublin

April 8, 2024

Co to jest interpolacja?

Interpolacja to proces znajdowania takiej funkcji, która przechodzi przez zadany zbiór punktów danych. W matematyce i informatyce jest to podstawowa technika wykorzystywana do estymacji wartości między znanymi punktami danych.

Do czego służy interpolacja?

Interpolacja ma szerokie zastosowanie, w tym:

- Przetwarzanie sygnałów i obrazów,
- Aproksymacja funkcji w celu analizy numerycznej,
- Symulacje komputerowe i grafika komputerowa,
- Inżynieria i nauki przyrodnicze do modelowania zjawisk.

Rodzaje interpolacji

Wyróżniamy kilka podstawowych rodzajów interpolacji:

- Interpolacja wielomianowa, np. metoda Lagrange'a, Newtona,
- Interpolacja funkcji sklejanych, w tym splajny liniowe, kwadratowe i sześcienne,
- Interpolacja za pomocą krzywych Bezjera,
- Inne metody, takie jak interpolacja Hermite'a.

Co to jest interpolacja?

Interpolacja to proces znajdowania funkcji interpolacyjnej $W(x)$, która dokładnie przechodzi przez zbiór danych punktów (węzłów interpolacji) (x_i, y_i), gdzie $y_i = f(x_i)$, dla $i = 0, 1, 2, \ldots, n$. Celem jest przybliżenie wartości funkcji w punktach poza danymi węzłami oraz oszacowanie błędu tych przybliżonych wartości.

Konstrukcja funkcji interpolacyjnej

Funkcja interpolacyjna $W(x)$ jest konstruowana jako kombinacja liniowa $n+1$ funkcji bazowych $\phi_i(x)$, zapisywana ogólnie jako:

$$W(x) = \sum_{i=0}^{n} a_i \phi_i(x) \quad \text{lub} \quad W(x) = a_0\phi_0(x) + a_1\phi_1(x) + \ldots + a_n\phi_n(x)$$

gdzie:

- $\phi_i(x)$ są funkcjami bazowymi,
- a_i są współczynnikami, wyznaczanymi na podstawie danych węzłów interpolacji.

Konstrukcja funkcji interpolacyjnej

Wprowadzając macierz bazową $\Phi = [\phi_0(x), \phi_1(x), \ldots, \phi_n(x)]$ i wektor współczynników $A = [a_0, a_1, \ldots, a_n]^T$, możemy zapisać $W(x)$ jako:

$$W(x) = \Phi(x) \cdot A$$

Po podstawieniu $n+1$ węzłów tworzy nam układ $n+1$ równań z $n+1$ niewiadomymi:

$$\begin{aligned}
W(x_0) &= a_0\phi_0(x_0) + a_1\phi_1(x_0) + \ldots + a_n\phi_n(x_0) = y_0,\\
W(x_1) &= a_0\phi_0(x_1) + a_1\phi_1(x_1) + \ldots + a_n\phi_n(x_1) = y_1,\\
&\vdots\\
W(x_n) &= a_0\phi_0(x_n) + a_1\phi_1(x_n) + \ldots + a_n\phi_n(x_n) = y_n.
\end{aligned}$$

Konstrukcja funkcji interpolacyjnej

W zapisie macierzowym:

$$X \cdot A = Y :$$

$$\begin{bmatrix} \phi_0(x_0) & \phi_1(x_0) & \dots & \phi_n(x_0) \\ \phi_0(x_1) & \phi_1(x_1) & \dots & \phi_n(x_1) \\ \vdots & \vdots & \ddots & \vdots \\ \phi_0(x_n) & \phi_1(x_n) & \dots & \phi_n(x_n) \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_n \end{bmatrix} = \begin{bmatrix} y_0 \\ y_1 \\ \vdots \\ y_n \end{bmatrix}$$

gdzie:

- X - macierz główna układu, $\det(X)$ musi być różne od 0,
- A - wektor współczynników, niewiadomych,
- Y - wektor z wartościami funkcji.

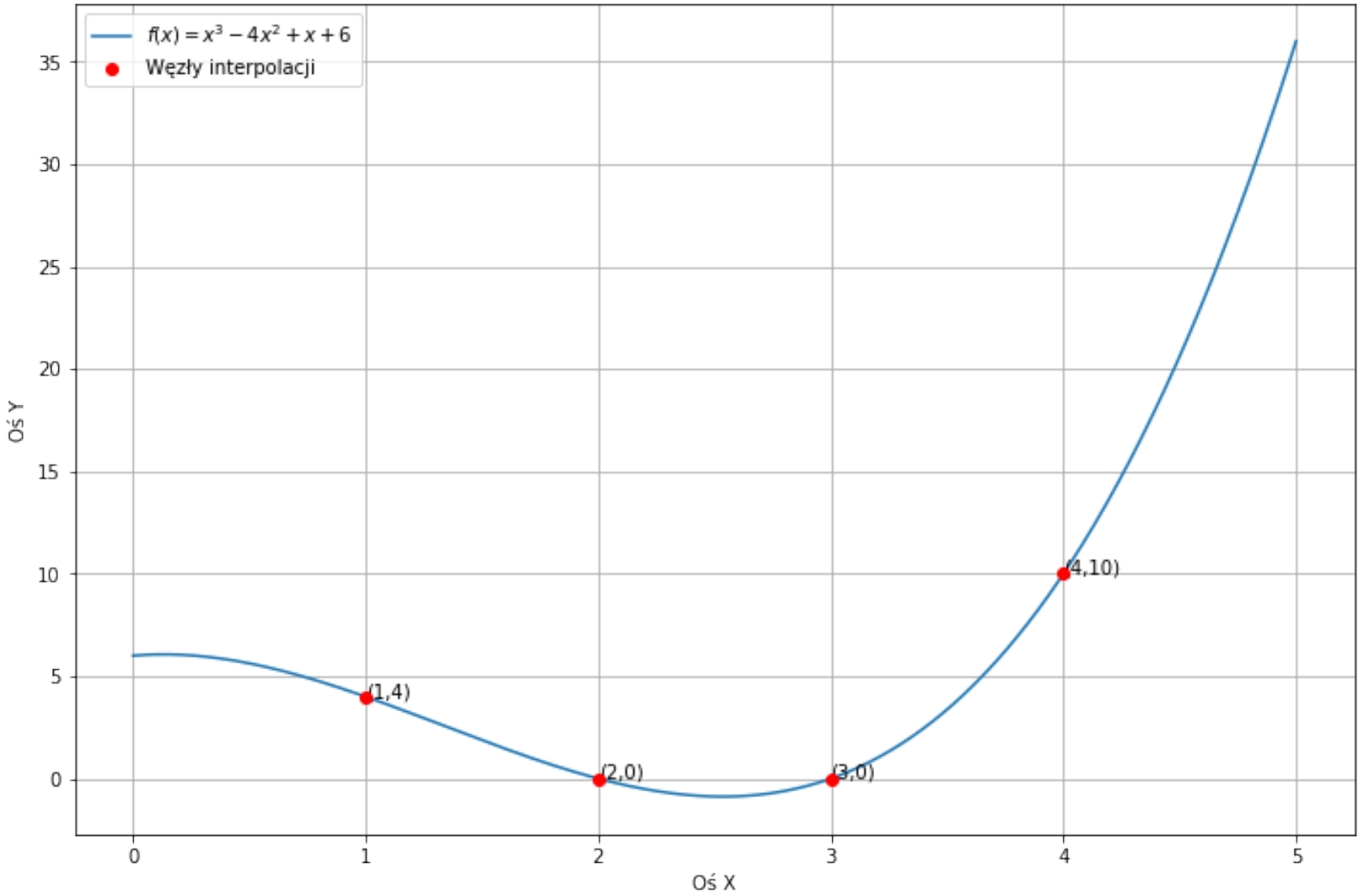
Konstrukcja funkcji interpolacyjnej

Jeżeli macierz X jest nieosobliwa, współczynniki A wyznaczane są jako $A = X^{-1} \cdot Y$, co prowadzi do postaci wielomianu interpolacyjnego:

$$W(x) = \Phi(x) \cdot X^{-1} \cdot Y$$

Tak skonstruowany wielomian interpolacyjny jest wynikiem iloczynu macierzy bazowej, macierzy interpolacyjnej X^{-1} i wektora wartości funkcji Y w węzłach interpolacji.

Przykład interpolacji



Tablicowanie vs. Interpolacja

Interpolacja można rozumieć jako proces odwrotny do tablicowania funkcji. Gdzie tablicowanie umożliwia tworzenie zestawu wartości na podstawie znanej formuły funkcji, interpolacja zajmuje się wyznaczaniem analitycznej formy funkcji bazując na zestawie danych wartości.

Tablicowanie:

- Polega na stworzeniu tablicy wartości dla danej funkcji.
- Używane, gdy znana jest analityczna postać funkcji.

Interpolacja:

- Polega na wyznaczeniu analitycznej formy funkcji, opierając się na zestawie jej wartości.
- Często wykorzystywana do określenia funkcji, której dokładna forma nie jest znana.

Poszukiwanie funkcji interpolacyjnej

W procesie interpolacji zazwyczaj dąży się do znalezienia funkcji interpolacyjnej o konkretnej, wcześniej założonej formie. Przykłady takich funkcji to:

- Wielomiany algebraiczne,
- Funkcje trygonometryczne,
- Inne postacie funkcji, które najlepiej pasują do charakteru danych.

Interpolacja wielomianowa

Wstęp do interpolacji wielomianowej

Interpolacja wielomianowa jest powszechnie stosowaną metodą w dziedzinie inżynierii, opartą na bazie jednomianów:

$$\phi_0(x) = 1, \phi_1(x) = x, \phi_2(x) = x^2, \ldots, \phi_n(x) = x^n$$

- Baza ta jest zamknięta dla funkcji ciągłych na skończonym przedziale $[x_0, x_n]$, co oznacza, że każdą funkcję z tej klasy można przedstawić jako szereg funkcji bazowych.

Forma wielomianu funkcyjnego

Wielomian interpolacyjny przyjmuje postać:

$$W(x) = a_0 + a_1 x + a_2 x^2 + \ldots + a_n x^n$$

Po podstawieniu wartości w węzłach otrzymujemy następujący układ równań:

$$\begin{aligned} W(x_0) &= a_0 + a_1 x_0 + \ldots + a_n x_0^n = y_0 \\ W(x_1) &= a_0 + a_1 x_1 + \ldots + a_n x_1^n = y_1 \\ &\vdots \\ W(x_n) &= a_0 + a_1 x_n + \ldots + a_n x_n^n = y_n \end{aligned}$$

Warunek interpolacji wymaga, aby $W(x_i) = y_i$ dla $i = 0, 1, \ldots, n$. Oznacza to, że układ równań tworzony przez te warunki ma jednoznaczne rozwiązanie, gdy wszystkie x_i są różne.

Wyznacznik macierzy i macierz odwrotna

Wartość wyznacznika macierzy głównej X, macierzy Vandermonde'a wynosi:

$$D = \det X = \begin{vmatrix} 1 & x_0 & \dots & x_0^n \\ 1 & x_1 & \dots & x_1^n \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_n & \dots & x_n^n \end{vmatrix} = \prod_{0 \leqslant j < i \leqslant n} (x_i - x_j) \neq 0$$

- Macierz odwrotna do bazy wielomianowej jest czasami nazywana macierzą Lagrange'a.
- Choć ta metoda interpolacji jest matematycznie elegancka, może nie być efektywna numerycznie. Macierz X jest pełna i może być źle uwarunkowana, co prowadzi do ryzyka dużych błędów w obliczeniach numerycznych.

Obliczanie wartości a_i

Wartości a_i oblicza się ze wzoru wynikającego z twierdzenia Cramera:

$$a_i = \frac{1}{D} \sum_{j=0}^{n} y_j X_{j+1,i+1} \qquad (1)$$

gdzie D jest wyznacznikiem macierzy głównej układu, a $X_{j+1,i+1}$ są kolejnymi dopełnieniami algebraicznymi elementów $i+1$-tej kolumny macierzy głównej.

Przykład - wyznacznik Vandermonde'a

Rozważmy obliczenie wartości wyznacznika Vandermonde'a dla punktów $x_0 = 2$, $x_1 = 3$, $x_2 = 4$.

Wyznacznik Vandermonde'a jest dany przez:

$$D = \begin{vmatrix} 1 & x_0 & x_0^2 \\ 1 & x_1 & x_1^2 \\ 1 & x_2 & x_2^2 \end{vmatrix} = \begin{vmatrix} 1 & 2 & 4 \\ 1 & 3 & 9 \\ 1 & 4 & 16 \end{vmatrix}$$

Wykorzystując wzór na wyznacznik Vandermonde'a:

$$D = (x_1 - x_0)(x_2 - x_0)(x_2 - x_1)$$

Podstawiając dane wartości, otrzymujemy:

$$D = (3 - 2)(4 - 2)(4 - 3) = 2$$

Funkcje interpolacyjne

Funkcje odgrywają kluczową rolę w modelowaniu matematycznym, służąc jako narzędzia do opisu relacji między różnymi zmiennymi. W kontekście obliczeń numerycznych pojawia się wyzwanie przybliżonego przedstawiania funkcji, aby możliwe było obliczenie jej wartości dla dowolnych argumentów z przedziału $\langle a, b \rangle$ za pomocą ograniczonej liczby operacji arytmetycznych i logicznych.

Funkcje interpolacyjne

Wybór funkcji przybliżającej $\tilde{f}$, która będzie reprezentować oryginalną funkcję f, może zależeć od wielu czynników. Ważne jest, aby taka funkcja umożliwiała proste obliczenie jej wartości. Dlatego często wybiera się wielomiany algebraiczne jako funkcje przybliżające ze względu na ich łatwość definiowania i obliczania.

Funkcje interpolacyjne

Wielomian algebraiczny postaci:

$$W_n(x) = a_0 + a_1 x + a_2 x^2 + a_3 x^3 + \ldots + a_n x^n$$

jest często stosowany jako funkcja przybliżająca ze względu na jego prostotę definiowania za pomocą skończonej liczby współczynników oraz łatwość obliczeń.

Funkcje interpolacyjne

Interpolacja za pomocą wielomianów umożliwia przybliżenie dowolnej funkcji, gdzie wartość $W_n(x)$ dla argumentów x, które nie są węzłami, reprezentuje estymację wartości y. Błąd przybliżenia, czyli różnica między funkcją oryginalną a przybliżającą, jest oceniany poprzez porównanie ich wartości w wybranych punktach poza węzłami interpolacji.

Przykład - interpolacja funkcji $\sin(\pi x)$

Rozważmy zadanie obliczenia współczynników wielomianu interpolującego funkcję $\sin(\pi x)$ w przedziale $< -1, 1 >$, z pięcioma węzłami interpolacji.
Dane węzły interpolacji i wartości funkcji są następujące:

- $x_0 = -1, y_0 = 0$
- $x_1 = -0.5, y_1 = -1$
- $x_2 = 0, y_2 = 0$
- $x_3 = 0.5, y_3 = 1$
- $x_4 = 1, y_4 = 0$

Formułujemy układ równań na podstawie wzoru na wielomian interpolacyjny $W_n(x)$.

Postać macierzy Vandermonde'a

Dla zadanych węzłów interpolacji i funkcji $\sin(\pi x)$, macierz Vandermonde'a ma postać:

$$X = \begin{pmatrix} 1 & x_0 & x_0^2 & x_0^3 & x_0^4 \\ 1 & x_1 & x_1^2 & x_1^3 & x_1^4 \\ 1 & x_2 & x_2^2 & x_2^3 & x_2^4 \\ 1 & x_3 & x_3^2 & x_3^3 & x_3^4 \\ 1 & x_4 & x_4^2 & x_4^3 & x_4^4 \end{pmatrix}$$

Podstawiając wartości węzłów:

$$X = \begin{pmatrix} 1 & -1 & 1 & -1 & 1 \\ 1 & -0.5 & 0.25 & -0.125 & 0.0625 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 0.5 & 0.25 & 0.125 & 0.0625 \\ 1 & 1 & 1 & 1 & 1 \end{pmatrix}$$

Macierz i wyznacznik Vandermonde'a

Wyznacznik $D = det(X)$ obliczamy jako:

$$D = (x_1 - x_0)(x_2 - x_0)(x_3 - x_0)(x_4 - x_0) \cdot \ldots \cdot (x_4 - x_3)$$

Co daje nam:

$$D = 9/32$$

Obliczanie współczynników i postaci wielomianu interpolacyjnego

Współczynniki a_i wielomianu interpolacyjnego obliczamy używając wzoru Cramera (1). Po podstawieniu wartości y i obliczeniach otrzymujemy:

$$a_0 = 0, a_1 = 8/3, a_2 = 0, a_3 = -8/3, a_4 = 0$$

Ostatecznie, wielomian interpolacyjny przyjmuje postać:

$$W_3(x) = \frac{8}{3}x - \frac{8}{3}x^3$$

Interpolacja Lagrange'a

Interpolacja Lagrange'a i funkcje bazowe

W interpolacji Lagrange'a dla $n+1$ węzłów:

$$(x_0, y_0), \ldots, (x_n, y_n)$$

funkcje bazowe są zdefiniowane jako:

$$\phi_i(x) = \prod_{\substack{j=0 \\ j \neq i}}^{n} (x - x_j),$$

gdzie dla każdego i, funkcja $\phi_i(x)$ pomija czynnik $(x - x_i)$, będąc wielomianem stopnia n.

Układ równań interpolacji Lagrange'a

Równania dla współczynników a_i wynikają z:

$$W(x_i) = y_i, \quad \text{dla } i = 0, 1, \ldots, n,$$

co prowadzi do macierzy głównej układu równań, gdzie każda funkcja bazowa $\phi_i(x)$ zeruje się dla $x = x_i$ oprócz jednego węzła, co upraszcza rozwiązanie układu.

Rozwiązanie układu równań

Wielomian interpolacyjny Lagrange'a można zapisać w uproszczonej formie:

$$W(x) = \sum_{i=0}^{n} y_i \left(\prod_{\substack{j=0 \\ j \neq i}}^{n} \frac{x - x_j}{x_i - x_j} \right),$$

co daje nam bezpośrednią metodę obliczania wartości wielomianu interpolacyjnego w dowolnym punkcie x, nie będącym węzłem interpolacji.

(przykład na tablicy)

Interpolacja Newtona

Wzór interpolacyjny Newtona

Wielomian $p \in P_n$, spełniający dla danej funkcji f warunki

$$p(x_i) = f(x_i),$$

dla $i = 0, 1, \ldots, n$ w parami różnych węzłach x_i, można wyrazić za pomocą wzoru interpolacyjnego Newtona:

$$p(x) = \sum_{k=0}^{n} f[x_0, x_1, \ldots, x_k] \prod_{j=0}^{k-1} (x - x_j)$$

Ilorazy różnicowe $f[x_0, x_1, \ldots, x_k]$

Wartość $f[x_0, x_1, \ldots, x_k]$ zależy tylko od węzłów $x_0, x_1, \ldots, x_k$ i wartości funkcji w tych węzłach. Jest to iloraz różnicowy rzędu k dla funkcji f i wymienionych wyżej węzłów.
Ilorazy różnicowe:

- rzędu zerowego

$$f[x_i] = f(x_i),$$

- rzędu pierwszego

$$f[x_i, x_{i+1}] = \frac{f[x_{i+1}] - f[x_i]}{x_{i+1} - x_i}$$

- rzędu k:

$$f[x_i, x_{i+1}, \ldots, x_{i+k}] = \frac{f[x_{i+1}, \ldots, x_{i+k}] - f[x_i, \ldots, x_{i+k-1}]}{x_{i+k} - x_i}$$

dla $i = 0, \ldots, n$

Ilorazy różnicowe $f[x_0, x_1, \ldots, x_k]$

Ilorazy różnicowe wyznaczamy przy pomocy tablicy trójkątnej, którą można utworzyć znając węzły i wartości funkcji w tych węzłach:

$$
\begin{array}{lllll}
x_0 & f[x_0] & & & \\
 & & f[x_0, x_1] & & \\
x_1 & f[x_1] & & f[x_0, x_1, x_2] & \\
 & & f[x_1, x_2] & & f[x_0, x_1, x_2, x_3] \\
x_2 & f[x_2] & & f[x_1, x_2, x_3] & \\
 & & f[x_2, x_3] & & \\
x_3 & f[x_3] & & &
\end{array}
$$

(przykład na tablicy)

Zbieżność wielomianów interpolacyjnych i błąd interpolacji

Twierdzenia o zbieżności

Twierdzenie 1 (Fabera)

Dla dowolnego ciągu układów węzłów $a \leqslant x_0 < x_1 < \ldots < x_n \leqslant b$, istnieje taka funkcja ciągła w $[a, b]$, że ciąg wielomianów interpolacyjnych zbudowanych dla tych węzłów nie jest do niej zbieżny.

Twierdzenie 2

Jeżeli f jest funkcją ciągłą w $[a, b]$, to istnieje taki ciąg układów węzłów $a \leqslant x_0 < x_1 < \ldots < x_n \leqslant b$, że zbudowane dla nich wielomiany interpolacyjne tworzą ciąg zbieżny do f.

Błąd interpolacji

Dość naturalne wydaje się przyjęcie, że zwiększenie liczby węzłów interpolacji (lub stopnia wielomianu interpolacyjnego) pociąga za sobą coraz lepsze przybliżenie funkcji $f(x)$ wielomianem $L_n(x)$.
Idealna byłaby zależność:

$$\lim_{n\to\infty} L_n(x) = f(x),$$

tj. dla coraz większej liczby węzłów wielomian interpolacyjny staje się „coraz bardziej podobny" do interpolowanej funkcji.
Dla węzłów równo odległych tak być nie musi $\longrightarrow$ efekt Rungego.

Błąd interpolacji

Można dowieść, że dla każdego wielomianu interpolacyjnego stopnia n, przybliżającego funkcję $f(x)$ w przedziale $[a, b]$ na podstawie $n+1$ węzłów, istnieje taka liczba ξ zależna od x, że dla reszty interpolacji $r(x)$:

$$\frac{f^{(n+1)}(\xi)}{(n+1)!} \cdot p_n(x) = r(x),$$

gdzie $p_n(x) = (x - x_0)(x - x_1) \dots (x - x_n)$, a $\xi \in (a; b)$ jest liczbą zależną od x.
Do oszacowania z góry wartości $r(x)$ można posłużyć się wielomianami Czebyszewa stopnia $n+1$ do oszacowania wartości $p_n(x)$ dla $x \in [-1, 1]$. Dla przedziału $[a, b]$ wystarczy dokonać przeskalowania wielomianu $p_n(x)$.

Efekt Rungego

Kiedy występuje problem?

- Interpolowane funkcje mają wysoki stopień.
- Węzły interpolacyjne są równoodległe.

Problem ten pojawia się zwłaszcza na końcach przedziału interpolacji i jest szczególnie widoczny dla funkcji, które mają duże drugie lub wyższe pochodne w obrębie przedziału interpolacji, ponieważ wysokiego stopnia wielomiany mają tendencję do wykazywania oscylacji między punktami węzłowymi, co prowadzi do znacznego wzrostu błędu interpolacji na krańcach przedziału.

Efekt Rungego

Rozwiązanie problemu

- Użycie interpolacji kawałkowej, takiej jak funkcje sklejane (splajny), które zamiast jednego wielomianu wysokiego stopnia używają serii wielomianów niskiego stopnia na mniejszych podprzedziałach.
- Użycie węzłów Czebyszewa zamiast równoodległych punktów.

Interpolacja funkcjami sklejanymi

Funkcje sklejane

Interpolacja funkcjami sklejanymi (ang. spline interpolation) to metoda przybliżania funkcji za pomocą kawałkami wielomianów niskiego stopnia, które są "sklejane" w taki sposób, aby całość była gładka.

Funkcja sklejana stopnia k to funkcja, która na każdym podprzedziale zadanego przedziału interpolacji jest wielomianem stopnia co najwyżej k i która ma ciągłe pochodne do rzędu $k-1$ w punktach sklejenia.

Interpolacja funkcjami sklejanymi oferuje elastyczne i wydajne narzędzie do przybliżania funkcji, szczególnie gdy potrzebujemy wysokiej dokładności i gładkości interpolowanej funkcji.

Funkcje sklejane liniowe

Najprostszy typ funkcji sklejanych. Dla każdego podprzedziału $[x_i, x_{i+1}]$ interpolacja liniowa definiowana jest wzorem:

$$S(x) = y_i + \frac{y_{i+1} - y_i}{x_{i+1} - x_i}(x - x_i)$$

Funkcje sklejane kubiczne

Są to funkcje sklejane stopnia 3, najczęściej używane w praktyce ze względu na dobrą równowagę między złożonością obliczeniową a jakością interpolacji. Warunki sklejenia zapewniają gładkość przejść między wielomianami.

$$S_i(x) = a_i + b_i(x - x_i) + c_i(x - x_i)^2 + d_i(x - x_i)^3$$

Warunki brzegowe

Warunki interpolacji i naturalności

- W węzłach zewnętrznych $s_0(x_0) = f(x_0)$ i $s_{n-1}(x_n) = f(x_n)$.
- Warunek naturalności: $s_0''(x_0) = s_{n-1}''(x_n) = 0$.

Warunki w węzłach wewnętrznych Dla każdego węzła wewnętrznego x_i dla $i = 1, 2, \ldots, n-1$:

- $s_{i-1}(x_i) = s_i(x_i) = f(x_i)$,
- $s_{i-1}'(x_i) = s_i'(x_i)$,
- $s_{i-1}''(x_i) = s_i''(x_i)$.

Zastosowanie krzywych Béziera w interpolacji

Krzywej Béziera

Krzywe Béziera są szeroko stosowane w grafice komputerowej, animacji oraz w projektowaniu CAD (Computer-Aided Design) do modelowania gładkich i łatwo kontrolowalnych kształtów.

Krzywa Béziera stopnia n jest zdefiniowana jako kombinacja liniowa punktów kontrolnych $P_0, P_1, ..., P_n$ z wykorzystaniem wielomianów bazowych Bernsteina:

$$B(t) = \sum_{i=0}^{n} P_i B_{i,n}(t)$$

gdzie $B_{i,n}(t) = \binom{n}{i} t^i (1-t)^{n-i}$ dla $t \in [0, 1]$.

Właściwości krzywych Béziera i obliczanie punktów na krzywej

- Krzywa zaczyna się w P_0 i kończy w P_n.
- Tylko punkty kontrolne P_0 i P_n leżą na krzywej. Pozostałe wpływają na jej kształt.
- Krzywa jest zawsze zawarta w otoczce wypukłej swoich punktów kontrolnych.

Wyznaczeniu punktów na krzywej Béziera pomaga **algorytm de Casteljau** Jest to podstawowa metoda rekurencyjna służąca rysowaniu krzywych Béziera.

Podsumowanie - problemy i wyzwania

Interpolacja, pomimo swej użyteczności, wiąże się z pewnymi problemami:

- Efekt Rungego przy interpolacji wielomianowej na równoodległych węzłach.
- Wybór odpowiedniej metody interpolacji dla danego problemu.
- Obliczeniowa złożoność niektórych metod interpolacyjnych.

Metody numeryczne
Wykład nr 7
Aproksymacja

Aneta Wróblewska

UMCS, Lublin

April 15, 2024

Zagadnienie aproksymacji

Aproksymacja polega na zastąpieniu funkcji f inną funkcją f^* lub na znalezieniu funkcji f^* na podstawie pewnego znanego ciągu wartości funkcji f. Wartości te często mogą być obarczone dużym błędem (wartości empiryczne). Funkcja aproksymująca f^* powinna mieć tę własność, że łatwo przeprowadza się na niej operacje matematyczne (różniczkowanie, całkowanie). Dlatego jako funkcje aproksymujące stosuje się wielomiany algebraiczne, funkcje wymierne lub wielomiany trygonometryczne.

Źródła błędów w aproksymacji

W aproksymacji występują dwa **źródła błędów**: danych wejściowych, są to tzw. błędy pomiarów, oraz konkretnego modelu (klasy funkcji), który zamierza się dostosować do danych wejściowych, są to tzw. błędy modelu. W praktyce zarówno dane wejściowe, jak i model nie są doskonałe.
Aproksymację można traktować jako **problem dostosowania modelu matematycznego do danych**, którymi się dysponuje i do innych znanych faktów.

Aproksymacja w postaci ogólnej

W aproksymacji liniowej funkcję f zastępuje się (aproksymuje) funkcją f^*, wyrażoną następującą kombinacją liniową:

$$f^*(x) = a_0\phi_0(x) + a_1\phi_1(x) + \ldots + a_k\phi_k(x)$$

w której znanych jest $k+1$ funkcji $\phi_0, \phi_1, \ldots, \phi_k$.
W aproksymacji tej oblicza się wartości parametrów $a_0, a_1, \ldots, a_k$.
Jeżeli np. $\phi_i(x) = x^i$, to klasą dopuszczalnych funkcji f^* jest klasa wielomianów stopnia k.
Układ $1, x, x^2, \ldots, x^k$ nazywa się **bazą zbioru wszystkich wielomianów stopnia** k. Możliwe są też inne bazy tego zbioru.

Zagadnienie aproksymacji

Zagadnienie aproksymacji można zatem zapisać w skrócie następująco: mając dany ciąg t_i, poszukujemy ciągłej funkcji, która najlepiej przybliża daną funkcję $f(t)$.
Rozważając zbiór punktów:

$$(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n),$$

dążymy do znalezienia funkcji $f(x)$ danej klasy, która w punktach $x_1, x_2, \ldots, x_n$ najefektywniej przybliża wartości y_i.
Podobne zagadnienie można sformułować dla funkcji. Dla danej funkcji $g(x)$, która ma być przybliżona przez funkcję $f(x)$ danej klasy, potrzebujemy określić miarę jakości przybliżenia - odległość między zbiorem $\{y_1, y_2, \ldots, y_n\}$ a zbiorem $\{f(x_1), f(x_2), \ldots, f(x_n)\}$ lub też między funkcją $g(x)$ a przybliżającą ją funkcją $f(x)$.

Wprowadzenie do metryk i norm

Zdefiniujemy miarę odległości w zbiorze X, zwaną metryką. Przestrzeń metryczna to para (X, d), gdzie X jest zbiorem, a d jest funkcją zdefiniowaną na $X \times X$

$$d : (x, y) \mapsto [0, 1),$$

która spełnia warunki:

1. $d(x, y) = 0 \Leftrightarrow x = y$,
2. $d(x, y) = d(y, x)$,
3. $d(x, y) + d(y, z) \geqslant d(x, z)$.

Ostatni warunek jest znany jako warunek trójkąta. Wartość metryki $d(x, y)$ reprezentuje odległość między punktami x i y.

Normy funkcji

Niech F będzie rodziną funkcji rzeczywistych, ciągłych i ograniczonych, określonych na odcinku $K = [a, b]$ lub funkcji określonych na zbiorze $K = \{x_1, x_2, \ldots, x_n\}$. Norma funkcji to odwzorowanie

$$\|\cdot\| : F \to [0, 1),$$

które funkcji $f \in F$ przypisuje nieujemną liczbę $\|f\|$ i które spełnia warunki:

1. $\|f\| = 0 \Leftrightarrow f \equiv 0$,
2. $\|\lambda f\| = |\lambda|\|f\|$,
3. $\|f\| + \|g\| \geqslant \|f + g\|$.

Ostatni warunek to warunek trójkąta. Norma określa metrykę w rodzinie funkcji. Jeśli F jest rodziną funkcji określonych na zbiorze K, to wzór

$$d_{\|\cdot\|}(f, g) = \|f - g\|$$

określa metrykę w tej rodzinie. Para $(F, d_{\|\cdot\|})$ stanowi przestrzeń metryczną.

Norma jednostajna

Definiujemy normę jednostajną wzorem

$$\|f\| = \sup_{x \in K} |f(x)|,$$

gdzie supremum (sup) oznacza wartość największą w zbiorze. Jest to norma, ponieważ spełnione są warunki (1) i (2). Warunek (3) jest spełniony, ponieważ dla dowolnych $a, b \in \mathbb{R}$

$$|a| + |b| \geqslant |a + b|.$$

Przykład 1

Na przedziale $K = [-5, 5]$ mamy zdefiniowane funkcje

$$f(x) = \frac{x}{x^2 + 1} - \frac{10x^2}{10x^2 + 1},$$

$$g(x) = \frac{x}{10}.$$

Interesuje nas znalezienie maksimum ich różnicy

$$h(x) = f(x) - g(x) = \frac{9x}{100x^2 + 10}.$$

Przykład 1 cd

Pochodna funkcji $h(x)$:

$$h'(x) = \frac{-9(10x^2 - 1)}{10(10x^2 + 1)^2}.$$

Dla $x > 0$, pochodna zeruje się w punkcie

$$x_0 = \sqrt{\frac{1}{10}} \approx 0.3162277660168379,$$

i to w tym punkcie funkcja h osiąga swoje maksimum.

Przykład 1 cd

Maksymalna różnica między funkcjami f i g wynosi

$$\|f - g\| = h(x_0) = \frac{9}{2 \cdot 10^{3/2}} \approx 0.142302494707577.$$

Norma L2

Zdefiniujemy normę L2, zwaną także normą kwadratową. Dla funkcji f należącej do przestrzeni F, norma L2 jest wyrażona wzorem:

$$\|f\|_2 = \sqrt{\int_a^b f^2(x)\,dx},$$

co można również zapisać jako:

$$\|f\| = \sqrt{\int_a^b f^2(x)\,dx}.$$

Przykład 2

Rozważamy funkcje na przedziale $K = [-5, 5]$:

- $f(x) = \frac{x}{x^2+1} - \frac{10x^2}{10x^2+1}$
- $g(x) = \frac{x}{10}$

Szukamy normy L2 ich różnicy:

$$h(x) = f(x) - g(x) = \frac{9x}{100x^2 + 10}.$$

Przykład 2 cd

Obliczamy normę L2 różnicy funkcji $h(x)$:

$$\int_0^5 h^2(x)\,dx = \frac{20331}{\sqrt{10}\,\text{arc tg}\left(\frac{5}{\sqrt{10}}\right)} - \frac{4050}{5020000}$$
$$\approx 0.0185018457452533,$$

co oznacza, że norma L2 funkcji h jest równa:

$$\|h\| = \sqrt{\int_{-5}^5 h^2(x)\,dx} = \sqrt{\frac{20331}{\sqrt{10}\,\text{arc tg}\left(\frac{5}{\sqrt{10}}\right)} - \frac{4050}{100\sqrt{251}}}$$
$$\approx 0.192363435950043 9.$$

Norma L1

Definiujemy normę L1 dla funkcji f należącej do rodziny funkcji F za pomocą wzoru:

$$\|f\|_1 = \int_a^b |f(x)|\, dx.$$

Norma ta jest dobrze zdefiniowana, o ile całka niewłaściwa jest zbieżna.

Przykład 3

Rozważamy funkcję $h(x)$ określoną wzorem:

$$h(x) = \frac{9x}{100x^2 + 10}.$$

Obliczamy całkę normy L1 dla funkcji $h(x)$ na przedziale $[0, 5]$:

$$\int_0^5 h(x)\,dx = \frac{9 \ln 210}{200} \approx 0.248645382260930 2,$$

co daje normę L1 funkcji h na przedziale $[-5, 5]$ równą:

$$\|h\|_1 = \int_{-5}^5 |h(x)|\,dx \approx 0.497290764521860 5.$$

Normy funkcji określone na zbiorach skończonych lub ciągach

Normy mogą być również zdefiniowane dla funkcji określonych na zbiorach skończonych lub ciągach. Rozważmy zbiór $K = \{x_1, x_2, \ldots, x_n\}$ lub $K = \{x_1, x_2, x_3, \ldots\}$, gdzie $a_i = f(x_i)$.

1. Norma jednostajna $\|f\| = \sup\{|a_1|, |a_2|, \ldots\}$.
2. Norma L2 $\|f\|_2 = \sqrt{\sum_{i=1}^{\infty} a_i^2}$.
3. Norma L1 $\|f\|_1 = \sum_{i=1}^{\infty} |a_i|$.

Aproksymacja wielomianowa

1. Szeregi potęgowe i ich zastosowania w aproksymacji

Szeregi potęgowe

Szereg potęgowy zdefiniowany dla pewnego punktu x_0 wyraża się jako:

$$f(x) = \sum_{k=0}^{\infty} a_k (x - x_0)^k,$$

gdzie $x \in (x_0 - r, x_0 + r)$, a r jest promieniem zbieżności szeregu. Dla x spoza tego przedziału, szereg jest rozbieżny.

Forma szeregu potęgowego

Możemy zapisać szereg potęgowy jako:

$$f(x) = \sum_{k=0}^{n-1} a_k (x - x_0)^k + R_n,$$

gdzie R_n jest resztą szeregu.

Aproksymacja szeregu potęgowego - wzór Taylora i Maclaurina

Jeśli:

$$f(x) = \sum_{k=0}^{n-1} a_k (x - x_0)^k + R_n$$

i $x \in (x_0 - r, x_0 + r)$, to

$$f(x) \approx \sum_{k=0}^{n-1} a_k (x - x_0)^k.$$

Powyższy wzór, gdzie $a_k = \frac{f^{(k)}(x_0)}{k!}$ nazywamy wzorem Taylora.

Dla $x_0 = 0$ wzór Taylora nazywamy wzorem Maclaurina:

$$f(x) \approx \sum_{k=0}^{n-1} a_k x^k.$$

Przykład 4 - aproksymacja $\sin x$

Przybliżenie funkcji $\sin x$ wzorem Maclaurina ma następującą formę:

$$\sin x \approx \sum_{k=1}^{n} (-1)^{k-1} \frac{x^{2k-1}}{(2k-1)!} = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \cdots + (-1)^{n-1} \frac{x^{2n-1}}{(2n-1)!}.$$

2. Wielomiany Czebyszewa i ich zastosowanie w aproksymacji

Definicja wielomianów Czebyszewa

Wielomiany Czebyszewa definiuje się rekurencyjnie:

$$\begin{aligned} T_0(x) &= 1, \\ T_1(x) &= x, \\ T_k(x) &= 2xT_{k-1}(x) - T_{k-2}(x), \quad k \geqslant 2. \end{aligned}$$

W przedziale $[-1, 1]$, wielomiany Czebyszewa wyrażają się jako:

$$T_k(x) = \cos(k \arccos x),$$

dla $k = 0, 1, 2, \ldots$.

Aproksymacja funkcji za pomocą wielomianów Czebyszewa

Aproksymacja funkcji $f(x)$ sumami wielomianów Czebyszewa $T_k(x)$:

$$f(x) \approx \frac{c_0}{2} + \sum_{k=1}^{n} c_k T_k(x),$$

gdzie

$$c_k = \frac{2}{\pi} \int_{-1}^{1} \frac{f(x) T_k(x)}{\sqrt{1-x^2}} dx.$$

Przykład 5 - funkcja signum (sgn)

Rozważamy funkcję $sgn(x)$, określoną jako:

$$sgn(x) = \begin{cases} 1 & \text{dla } x > 0, \\ -1 & \text{dla } x < 0, \\ 0 & \text{dla } x = 0. \end{cases}$$

Ograniczając do dziedziny $(-1, 1)$, otrzymujemy:

$$f(x) = \begin{cases} 1 & \text{dla } x \in (0, 1), \\ -1 & \text{dla } x \in (-1, 0), \\ 0 & \text{dla } x = 0. \end{cases}$$

Można wykazać, że współczynniki c_k dla funkcji $sgn(x)$ wynoszą:

$$c_k = \begin{cases} 0 & \text{dla } k = 2i, \\ (-1)^{k+1} \frac{4}{\pi k} & \text{dla } k = 2i+1, \quad k = 0, 1, \ldots. \end{cases}$$

Aproksymacja szeregami trygonometrycznymi

Szeregi trygonometryczne Fouriera

Szereg trygonometryczny Fouriera dla funkcji okresowej ma postać:

$$f(x) = \frac{a_0}{2} + \sum_{k=1}^{\infty}(a_k \cos(kx) + b_k \sin(kx)),$$

gdzie $x \in [-\pi, \pi]$, pod warunkiem zbieżności szeregu. Funkcja $f(x)$ jest okresowa z okresem 2π.
Aproksymację funkcji $f(x)$ uzyskujemy, biorąc początkowe składniki sumy szeregu trygonometrycznego:

$$f(x) \approx \frac{a_0}{2} + \sum_{k=1}^{n}(a_k \cos(kx) + b_k \sin(kx)).$$

Przykład 6 - funkcja signum (sgn)

Rozważamy funkcję $sgn(x)$ na dziedzinie $(-\pi, \pi)$:

$$f(x) = \begin{cases} 1 & \text{dla } x \in (0, \pi), \\ -1 & \text{dla } x \in (-\pi, 0), \\ 0 & \text{dla } x = 0. \end{cases}$$

Współczynniki szeregu Fouriera dla funkcji $sgn(x)$ są takie, że $a_k = 0$ oraz:

$$b_k = \begin{cases} 0 & \text{dla } k = 2i, \\ \frac{4}{\pi k} & \text{dla } k = 2i + 1, \end{cases}$$

gdzie $k = 0, 1, 2, \ldots$.

Przykład 6 cd

Aproksymacja funkcji $sgn(x)$ szeregiem trygonometrycznym:

$$f(x) \approx \sum_{k=0}^{n} b_{2k+1} \sin((2k+1)x),$$

gdzie $s(k,x) = b_k \sin(kx)$, a więc:

$$f(x) \approx \sum_{k=0}^{n} s(2k+1, x).$$

Aproksymacja średniokwadratowa

Aproksymacja średniokwadratowa

Kolejnym zagadnieniem jest aproksymacja średniokwadratowa (metoda najmniejszych kwadratów) i jej zastosowanie do przybliżania funkcji z użyciem normy L_2.

Dla zbioru punktów (x_i, y_i), gdzie $y_i = f(x_i)$, szukamy funkcji $F(x)$ takiej, że dla pewnej funkcji wagowej $w(x)$ wyrażenie:

$$||F - f||_2 = \sum_{i=1}^{n} w(x_i)(F(x_i) - y_i)^2$$

osiąga minimum.
Dla uproszczenia w dalszych rozważaniach przyjmujemy $w(x) = 1$.

Aproksymacja liniowa

Najprostszym przypadkiem aproksymacji średniokwadratowej jest aproksymacja liniowa. W aproksymacji liniowej szukamy $F(x) = ax + b$, minimalizując:

$$h(a, b) = \sum_{i=1}^{n} (ax_i + b - y_i)^2.$$

Obliczamy pochodne cząstkowe $h(a, b)$ po a i b, a następnie rozwiązujemy układ równań:

$$\begin{cases} \sum_{i=1}^{n} (y_i - ax_i - b)x_i = 0, \\ \sum_{i=1}^{n} (y_i - ax_i - b) = 0. \end{cases}$$

Po przekształceniach otrzymujemy:

$$a = \frac{nA - BC}{nD - B^2}, \quad b = \frac{CD - AB}{nD - B^2},$$

gdzie:

$$A = \sum_{i=1}^{n} x_i y_i, \quad B = \sum_{i=1}^{n} x_i, \quad C = \sum_{i=1}^{n} y_i, \quad D = \sum_{i=1}^{n} x_i^2.$$

Przykład 9 - aproksymacja liniowa

Mając dane przedstawione poniższą tabelą

x_i	y_i
1	1
3	12
5	25
7	38

możemy obliczyć współczynniki a i b jako:

$$a = 6.2, \quad b = -5.8.$$

Aproksymacja wielomianowa

Przyjmujemy formę funkcji aproksymującej:

$$F(x) = a_0\phi_0(x) + a_1\phi_1(x) + \ldots + a_m\phi_m(x),$$

gdzie dla uproszczenia $\phi_k(x) = x^k$.
Zadaniem jest minimalizacja funkcji błędu:

$$h(a_0, a_1, \ldots, a_m) = \sum_{i=1}^{n} \left(\sum_{j=0}^{m} a_j x_i^j - y_i \right)^2 .$$

Aproksymacja wielomianowa

Dla każdego $i = 0, 1, \ldots, m$ otrzymujemy układ $m + 1$ równań z $m + 1$ niewiadomymi:

$$\frac{\partial h}{\partial a_i} = 2 \sum_{i=1}^{n} \left(\sum_{j=0}^{m} a_j x_i^j - y_i \right) x_i^i = 0.$$

Budowa układu równań - wielomian 1-go stopnia

Układ równań w postaci macierzowej dla dwóch niewiadomych a_0 i a_1 (wielomian 1-go stopnia, czyli równanie prostej: $y = ax + b$).

$$\begin{bmatrix} n & \sum_{i=1}^{n} x_i \\ \sum_{i=1}^{n} x_i & \sum_{i=1}^{n} x_i^2 \end{bmatrix} \cdot \begin{bmatrix} b \\ a \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^{n} y_i \\ \sum_{i=1}^{n} x_i y_i \end{bmatrix}$$

Budowa układu równań - ciąg dalszy

$$nb + a\sum_{i=1}^{n} x_i = \sum_{i=1}^{n} y_i$$

$$b\sum_{i=1}^{n} x_i + a\sum_{i=1}^{n} x_i^2 = \sum_{i=1}^{n} x_i y_i$$

Przykład 10 - aproksymacja wielomianem drugiego stopnia

Dla danych z poniższej tabeli znajdziemy wielomian aproksymujący drugiego stopnia.

x_i	y_i
0	2.00
0.5	2.48
1.0	2.84
1.5	3.00
2.0	2.91

Przykład 10 cd

Poszukujemy wielomianu $F(x) = ax^2 + bx + c$. Równania wynikające z minimalizacji błędu to:

$$a\sum_{i=1}^{n} x_i^4 + b\sum_{i=1}^{n} x_i^3 + c\sum_{i=1}^{n} x_i^2 = \sum_{i=1}^{n} x_i^2 y_i,$$

$$a\sum_{i=1}^{n} x_i^3 + b\sum_{i=1}^{n} x_i^2 + c\sum_{i=1}^{n} x_i = \sum_{i=1}^{n} x_i y_i,$$

$$a\sum_{i=1}^{n} x_i^2 + b\sum_{i=1}^{n} x_i + nc = \sum_{i=1}^{n} y_i.$$

Przykład 10 cd

Rozwiązując powyższy układ równań, uzyskujemy:

$$a = -\frac{67}{175}, \quad b = \frac{2159}{1750}, \quad c = \frac{6953}{3500}.$$

Wielomian aproksymujący:

$$F(x) = -\frac{67}{175}x^2 + \frac{2159}{1750}x + \frac{6953}{3500}.$$

Metody numeryczne
Wykład nr 8
Różniczkowanie numeryczne

Aneta Wróblewska

UMCS, Lublin

April 22, 2024

Wstęp

Różniczkowanie numeryczne

Różniczkowanie numeryczne jest metodą obliczania przybliżonych wartości pochodnych funkcji, wykorzystując wartości tej funkcji w skończonej liczbie punktów.

Różniczkowanie numeryczne jest niezbędne w sytuacjach, gdzie rozwiązania analityczne są trudne lub niemożliwe do uzyskania, na przykład w przypadku skomplikowanych modeli fizycznych, ekonomicznych czy biologicznych, które nie są wyrażone w prostych wzorach matematycznych.

Zastosowania różniczkowania numerycznego

Inżynieria i fizyka

Różniczkowanie numeryczne jest szeroko stosowane do analizowania dynamiki systemów, np. w symulacjach mechaniki płynów, gdzie oblicza się gradienty prędkości i ciśnienia. Obliczanie prędkości i przyspieszenia na podstawie zarejestrowanej trajektorii ruchu.

Na przykładzie mechaniki płynów, różnica skończona do obliczenia pochodnej ciśnienia p w punkcie może być wyrażona jako:

$$\frac{\Delta p}{\Delta x} = \frac{p(x+h) - p(x)}{h}$$

gdzie h jest małym krokiem w przestrzeni.

Zastosowania różniczkowania numerycznego

Ekonomia i finanse

W ekonomii i finansach, różniczkowanie numeryczne pozwala na obliczanie stóp zmian, np. w modelowaniu opcji finansowych za pomocą równania Blacka-Scholesa.

Biologia i medycyna

Wykorzystywane do modelowania rozprzestrzeniania się substancji chemicznych lub leków w organizmach żywych, obliczając szybkość zmian stężenia w czasie.

Informatyka

Stosowane w algorytmach uczenia maszynowego, szczególnie w technikach optymalizacyjnych jak spadek gradientu, gdzie pochodne funkcji kosztu są niezbędne do aktualizacji wag modeli.

Metoda różniczkowania Newtona

Metoda różniczkowania Newtona

Metoda różniczkowania Newtona, często nazywana metodą różnic skończonych Newtona, stosowana jest do numerycznego obliczania pochodnych funkcji. Metoda ta jest szczególnie przydatna, gdy nie dysponujemy analityczną formą funkcji, ale znamy jej wartości w dyskretnych punktach.

Pochodna funkcji

Dla funkcji f określonej w punktach x oraz $x + h$, gdzie h jest małym przyrostem, pochodna funkcji w punkcie x może być przybliżona jako:

$$f'(x) \approx \frac{f(x+h) - f(x)}{h}$$

To jest tak zwana różnica w przód (różnica zwykła).

Metoda różniczkowania Newtona

Przykład 1

Załóżmy, że chcemy obliczyć pochodną funkcji $f(x) = x^2$ w punkcie $x = 2$ z $h = 0.01$:

$$f'(2) \approx \frac{f(2.01) - f(2)}{0.01} = \frac{4.0401 - 4}{0.01} = 4.01$$

Teoretyczna wartość pochodnej, $2x$ w $x = 2$, wynosi 4. Dlatego nasze przybliżenie jest bardzo bliskie wartości rzeczywistej.

Metoda różniczkowania Newtona jest prosta w implementacji i nie wymaga złożonych obliczeń, jednak jej dokładność zależy od wielkości kroku h oraz zachowania funkcji f. Mała wartość h może prowadzić do błędów numerycznych związanych z precyzją arytmetyki komputerowej.

Metoda różnic skończonych

Metoda różnic skończonych

Co to są różnice skończone?

Różnice skończone to metoda przybliżania pochodnych funkcji poprzez wykorzystanie wartości funkcji w skończonej liczbie punktów.

Metoda jest szeroko stosowana w numerycznym rozwiązywaniu równań różniczkowych, gdzie analityczne rozwiązanie jest trudne lub niemożliwe do uzyskania. Można ją wyprowadzić wprost z ilorazu różnicowego, bądź z rozwinięcia w szereg Taylora.

Podstawowe wzory różnic skończonych

Różnica w przód (zwykła)

$$f'(x) \approx \frac{f(x+h) - f(x)}{h}$$

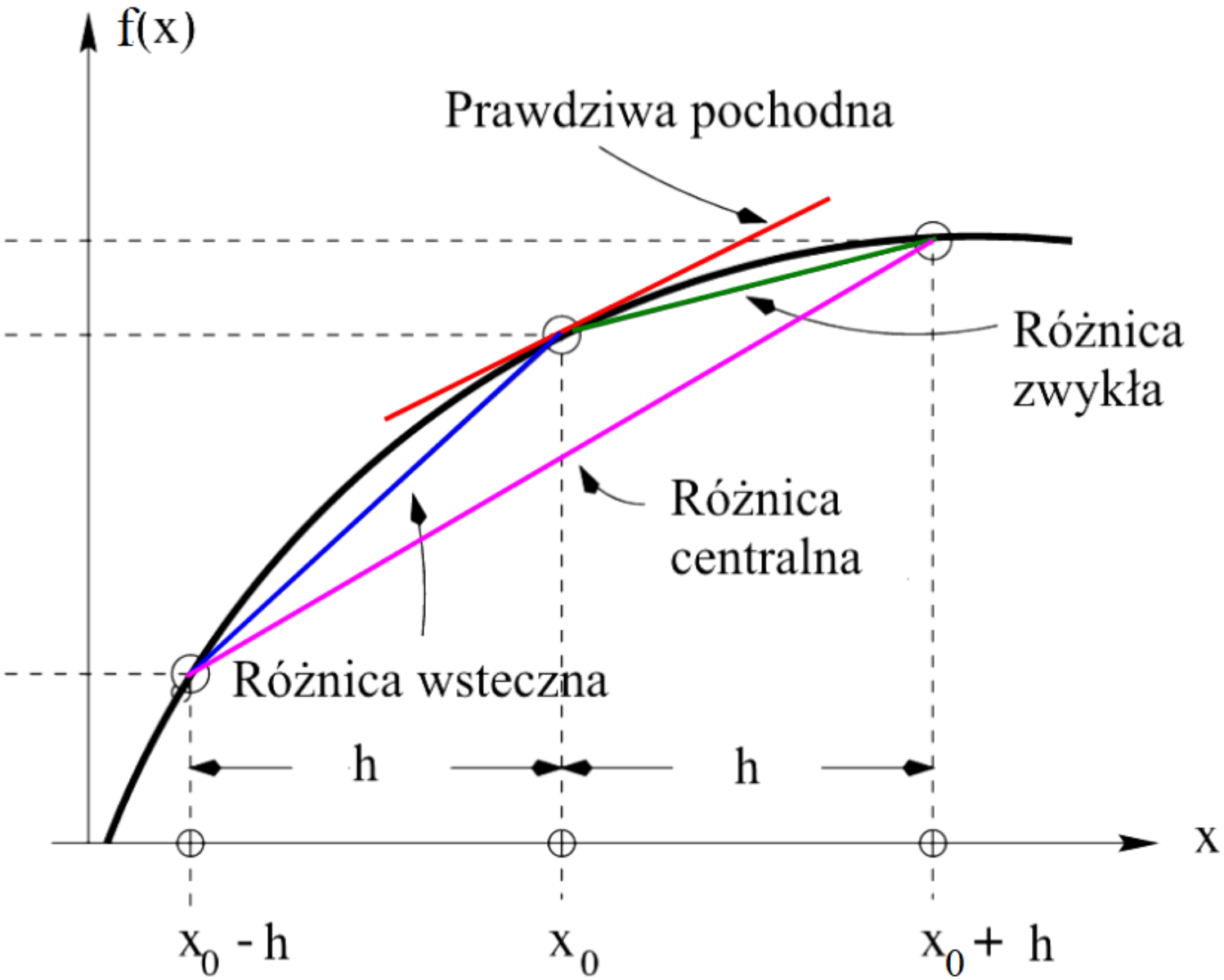
Różnica wsteczna

$$f'(x) \approx \frac{f(x) - f(x-h)}{h}$$

Różnica centralna

$$f'(x) \approx \frac{f(x+h) - f(x-h)}{2h}$$

Różnice skończone - porównanie



Różnice centralne - przykład

Przykład 2

Załóżmy, że mamy funkcję $f(x) = x^2$ i chcemy obliczyć $f'(2)$ przy $h = 0.1$ za pomocą różnicy centralnej:

$$f'(2) \approx \frac{(2.1)^2 - (1.9)^2}{0.2} = \frac{4.41 - 3.61}{0.2} = 4.0$$

Teoretyczna wartość pochodnej $f'(x) = 2x$ w $x = 2$ wynosi 4, więc wynik jest dokładny.

Wyprowadzenie metody różnic skończonych ze wzoru Taylora

- Rozwinięcie funkcji analitycznej $f(x)$ w otoczeniu punktu x w szereg Taylora można wyrazić w postaci:

$$f(x+h) = f(x) + hf'(x) + \frac{h^2}{2!}f''(x) + \dots$$

- Definiujemy operator różniczkowania $D^k f(x) = f^{(k)}(x)$
- Zatem:

$$f(x+h) = (1 + \frac{hD}{1!} + \frac{h^2D^2}{2!} + \dots)f(x) = e^{hD}f(x)$$

- Definiujemy operatory różnicy zwykłej Δ i wstecznej ∇:

$$\Delta f(x) = f(x+h) - f(x)$$

$$\nabla f(x) = f(x) - f(x-h)$$

Równość operatorów i logarytmowanie

- Z porównania zależności uzyskujemy wzór na równość operatorów:

$$e^{hD} = \Delta + 1$$

$$1 - \nabla = e^{-hD}$$

- Logarytmując obustronnie otrzymujemy:

$$\ln(1 + \Delta) = hD \quad \longrightarrow \quad D = \frac{1}{h} \ln(1 + \Delta)$$

$$\ln(1 - \nabla) = -hD \quad \longrightarrow \quad D = -\frac{1}{h} \ln(1 - \nabla)$$

Wzory na pochodne dla różnicy zwykłej

- Podnosząc (w pierwszej kolejności) równanie dla różnicy zwykłej obustronnie do potęgi k-tej, uzyskujemy:

$$D^k = \frac{1}{h^k}(\ln(1+\Delta))^k$$

- Możemy zatem wyprowadzić wzory na dowolne pochodne funkcji $f(x)$ wyrażone za pomocą różnic zwykłych i wstecznych.

Wyprowadzenie wzoru dla różnicy zwykłej

Rozwijając funkcję logarytmiczną w szereg wokół 1 (zwykle Δ to mała wielkość): $\ln(1+\Delta) = \Delta - \frac{\Delta^2}{2} + \frac{\Delta^3}{3} - \frac{\Delta^4}{4} + \ldots$, uzyskujemy następujące powiązanie operatora różniczkowania D z operatorem różnicy zwykłej ("w przód") Δ:

$$D^{(k)} = \frac{1}{h^k}(\Delta - \frac{\Delta^2}{2} + \frac{\Delta^3}{3} - \frac{\Delta^4}{4} + \ldots)^k$$

Wzory na pochodne funkcji dla różnicy zwykłej

Możemy zatem wyprowadzić wzory na dowolne pochodne funkcji $f(x)$ wyrażone za pomocą różnic zwykłych:

$$k = 1 \quad f^{(1)}(x) = \frac{1}{h}(\Delta f(x) - \frac{1}{2}\Delta^2 f(x) + \frac{1}{3}\Delta^3 f(x) - \frac{1}{4}\Delta^4 f(x) + \ldots)$$

$$k = 2 \quad f^{(2)}(x) = \frac{1}{h^2}(\Delta^2 f(x) - \Delta^3 f(x) + \frac{11}{12}\Delta^4 f(x) - \frac{10}{12}\Delta^5 f(x) + \ldots)$$

$$k = 3 \quad f^{(3)}(x) = \frac{1}{h^3}(\Delta^3 f(x) - \frac{3}{2}\Delta^4 f(x) + \frac{7}{4}\Delta^5 f(x) - \frac{45}{24}\Delta^6 f(x) + \ldots$$

Wzory na pochodne funkcji

Pokażemy (dowód na tablicy), że pochodną pierwszego rzędu (i analogicznie drugiego rzędu) można wyrazić w różnoraki sposób korzystając z większej lub mniejszej ilości wyrazów w powyższym rozwinięciu.

$$f'(x) = \frac{f(x_{i+1}) - f(x_i)}{h}$$

$$f'(x) = \frac{-3f(x_i) + 4f(x_{i+1}) - f(x_{i+2})}{2h}$$

$$f''(x) = \frac{f(x_i) - 2f(x_{i+1}) + f(x_{i+2})}{h^2}$$

$$f''(x) = \frac{2f(x_i) - 5f(x_{i+1}) + 4f(x_{i+2}) - 3f(x_{i+3})}{h^2}$$

Wzory na pochodne dla różnicy wstecznej

- Podnosząc następnie obustronnie równanie dla różnicy wstecznej do potęgi k-tej, uzyskujemy:

$$D^k = \frac{1}{h^k}(-\ln(1-\nabla))^k$$

- Dalej postępujemy analogicznie jak w przypadku różnic zwykłych.

Wzory na pochodne dla różnicy wstecznej

Ponieważ $\ln(1-\nabla) = -(\nabla + \frac{\nabla^2}{2} + \frac{\nabla^3}{3} + \frac{\nabla^4}{4} + \ldots)$,

$$D^k = \frac{1}{h^k}(\nabla + \frac{\nabla^2}{2} + \frac{\nabla^3}{3} + \frac{\nabla^4}{4} + \ldots)^k$$

Wzory na pochodne funkcji dla różnicy wstecznej

Możemy zatem wyprowadzić wzory na dowolne pochodne funkcji $f(x)$ wyrażone za pomocą różnic wstecznych:

$$k = 1 \quad f^{(1)}(x) = \frac{1}{h}(\nabla f(x) + \frac{1}{2}\nabla^2 f(x) + \frac{1}{3}\nabla^3 f(x) + \ldots)$$

$$k = 2 \quad f^{(2)}(x) = \frac{1}{h^2}(\nabla^2 f(x) + \nabla^3 f(x) + \frac{11}{12}\nabla^4 f(x) + \ldots)$$

$$k = 3 \quad f^{(3)}(x) = \frac{1}{h^3}(\nabla^3 f(x) + \frac{3}{2}\nabla^4 f(x) + \frac{34}{24}\nabla^5 f(x) + \ldots)$$

Różnice centralne

- Wzory różniczkowania numerycznego funkcji $f(x)$ w punkcie $x = x_0$ dla różnicy zwykłej i wstecznej wykorzystują jedynie wartości funkcji $f(x)$ dla argumentów leżących tylko z jednej strony x_0
- Wady tej nie posiadają wzory wykorzystujące wartości funkcji $f(x)$ po prawej i po lewej stronie punktu x_0. Są to wzory symetryczne, oparte na różnicach centralnych

Definicja operatora różnicy centralnej

$$\delta f(x) = \frac{f(x+h) - f(x-h)}{2h}$$

Zatem,

$$\delta^2 f(x) = \frac{f(x+h) - 2f(x) + f(x-h)}{h^2}$$

Rozwinięcie w szereg Taylora

$$f(x+h) = f(x) + f'(x)h + \frac{f''(x)}{2}h^2 + \frac{f'''(x)}{6}h^3 + O(h^4)$$

$$f(x-h) = f(x) - f'(x)h + \frac{f''(x)}{2}h^2 - \frac{f'''(x)}{6}h^3 + O(h^4)$$

Różnice centralne i błąd

Podstawiając te wzory do definicji $\delta f(x)$ i $\delta^2 f(x)$, otrzymujemy:

$$\delta f(x) = f'(x) + O(h^2)$$

$$\delta^2 f(x) = f''(x) + O(h^2)$$

Wzory na pochodne - podsumowanie

Dwupunktowe różnice zwykłe:

$$f'(x) \approx \frac{f(x+h) - f(x)}{h}, \qquad O(h)$$

Trzypunktowe różnice zwykłe:

$$f'(x) \approx \frac{-3f(x) + 4f(x+h) - f(x+2h)}{2h}, \qquad O(h^2)$$

Dwupunktowe różnice wsteczne:

$$f'(x) \approx \frac{f(x) - f(x-h)}{h}, \qquad O(h)$$

Trzypunktowe różnice wsteczne:

$$f'(x) \approx \frac{3f(x) - 4f(x-h) + f(x-2h)}{2h}, \qquad O(h^2)$$

Wzory na pochodne - podsumowanie cd

Dwupunktowe różnice centralne:

$$f'(x) \approx \frac{f(x+h) - f(x-h)}{2h}, \qquad O(h^2)$$

Czteropunktowe różnice centralne:

$$f'(x) \approx \frac{f(x-2h) - 8f(x-h) + 8f(x+h) - f(x+2h)}{12h}, \qquad O(h^4)$$

Błędy metody różnic skończonych

Błąd przybliżenia

Błąd metody różnic skończonych zależy od wartości h oraz od wyższych pochodnych funkcji f. Ogólnie, błąd dla różnicy centralnej jest rzędu $O(h^2)$, co oznacza, że mniejsze h daje dokładniejsze wyniki, ale zwiększa ryzyko błędu numerycznego.

Błąd w różniczkowaniu numerycznym - przykład

Rozważmy funkcję $f(x) = e^x$.
Policzymy pochodną w punkcie $x = 0$ korzystając z dwupunktowych różnic centralnych:

$$f'(x) = \frac{f(x+h) - f(x-h)}{2h} + O(h^2)$$

gdzie $i = 0, h \neq 0$.

$$f'(0) = \frac{e^h - e^{-h}}{2h} + O(h^2)$$

gdzie $i = 0, h \neq 0$.
Podczas obliczeń komputer wprowadza błąd zaokrąglenia:

$$e^h \longleftrightarrow e^h + R_1 \qquad e^{-h} \longleftrightarrow e^{-h} + R_2$$

Błąd w różniczkowaniu numerycznym - przykład (cd.)

Wartości dokładne:

$$f'(0) = \frac{e^h + R_1 - e^{-h} - R_2}{2h} + O(h^2) = \frac{e^h - e^{-h}}{2h} + \frac{R_1 - R_2}{2h} + O(h^2)$$

Gdy zmniejszamy h, błąd obcięcia ($O(h^2)$) maleje, ale błąd zaokrąglenia ($\frac{R_1 - R_2}{2h}$) rośnie.

Różniczkowanie funkcji aproksymującej

Różniczkowanie funkcji aproksymującej

Aproksymacja funkcji

Często funkcja f nie jest znana dokładnie, ale mamy jej przybliżone wartości w punktach. W takim przypadku używamy różnic skończonych do aproksymowania pochodnej na podstawie tych przybliżonych wartości.

Różniczkowanie za pomocą wielomianów Lagrange'a

Metoda interpolacyjna

Różnice skończone można połączyć z interpolacją wielomianową Lagrange'a, co pozwala na obliczanie pochodnych w punktach między znanymi wartościami funkcji.

$$f'(x) \approx \frac{\sum_{k=0}^{n} f(x_k) l'_k(x)}{\sum_{k=0}^{n} l_k(x)}$$

gdzie $l_k(x)$ to wielomiany interpolacyjne Lagrange'a.

Różniczkowanie za pomocą wielomianów Lagrange'a

Zapiszmy wielomian przechodzący przez trzy punkty (x_i, y_i), (x_{i+1}, y_{i+1}), (x_{i+2}, y_{i+2}):

$$f(x) = \frac{(x - x_{i+1})(x - x_{i+2})}{(x_i - x_{i+1})(x_i - x_{i+2})} y_i + \frac{(x - x_i)(x - x_{i+2})}{(x_{i+1} - x_i)(x_{i+1} - x_{i+2})} y_{i+1} + \frac{(x - x_i)(x - x_{i+1})}{(x_{i+2} - x_i)(x_{i+2} - x_{i+1})} y_{i+2}$$

Różniczkowanie za pomocą wielomianów Lagrange'a (cd.)

Różniczkujemy po x, a następnie podstawiamy $x = x_{i+1}$:

$$f'(x) = \frac{2x - x_i - x_{i+1}}{(x_{i+2} - x_i)(x_{i+2} - x_{i+1})} y_i + \frac{2x - x_i - x_{i+2}}{(x_{i+1} - x_i)(x_{i+1} - x_{i+2})} y_{i+1} + \frac{2x - x_{i+1} - x_{i+2}}{(x_i - x_{i+1})(x_i - x_{i+2})} y_{i+2}$$

Uwagi

1. Gdy punkty są równomiernie rozłożone, czyli $x_{i+2} - x_{i+1} = x_{i+1} - x_i = h$, korzystając z różnicy centralnej otrzymujemy:
$$f'(x_{i+1}) = \frac{y_{i+2} - y_i}{2h}$$
2. Zaleta nr 1: punkty nie muszą być równomiernie rozłożone
3. Zaleta nr 2: możemy policzyć pochodną w dowolnym punkcie między x_i a x_{i+2}

Ekstrapolacja Richardsona

Ekstrapolacja Richardsona

Weźmy szereg Taylora:

$$f(x \pm h) = \sum_{k=0}^{\infty} (\pm 1)^k \frac{f^{(k)}(x)}{k!} h^k$$

Odejmijmy stronami dwa powyższe równania i wyznaczmy pierwszą pochodną:

$$f'(x) = \frac{1}{2h}[f(x+h) - f(x-h)] - \left[\frac{1}{3!}h^2 f^{(3)}(x) + \frac{1}{5!}h^4 f^{(5)}(x) + \ldots\right]$$

$$L = D(h) + a_2 h^2 + a_4 h^4 + a_6 h^6 + \ldots \quad (h \neq 0) \qquad (1)$$

gdzie:
L - pierwsza pochodna
$D(h)$ - przybliżenie pochodnej
$a_2 h^2 + a_4 h^4 + a_6 h^6 + \ldots$ - błąd przybliżenia

Ekstrapolacja Richardsona

W równaniu (1) zamieńmy h na $h/2$ i pomnóżmy stronami przez 4:

$$4L = 4D(h/2) + a_2h^2 + \frac{a_4h^4}{4} + \frac{a_6h^6}{16} + \dots \quad (2)$$

Odejmijmy równanie (1) od (2) i podzielmy przez 3:

$$L = \frac{4}{3}D(h/2) - \frac{1}{3}D(h) - \frac{a_4h^4}{4} - \frac{a_6h^6}{16} - \dots \quad (3)$$

przejście od (1) do (3) jest pierwszym krokiem ekstrapolacji Richardsona

Ekstrapolacja Richardsona

Kombinacja $D(h)$ i $D(h/2)$ jest przybliżeniem L, którego błąd jest rzędu $O(h^4)$ w porównaniu z $O(h^2)$ dla wzoru (1).
Rozumowanie nie zależy od interpretacji L i $D(h)$ może być zastosowane do innych zagadnień.

Przykład 3

Zastosuj pierwszy krok ekstrapolacji Richardsona do znalezienia pochodnej funkcji $f(x) = 2^x$ w punkcie 3 znając dokładne wartości funkcji w punktach: $1, 2, 3, 4, 5$ oraz $h = 2$.

(Rozwiązanie na tablicy)

Ekstrapolacja Richardsona

Teraz rozumujemy podobnie jak przy przejściu z równania (1) do (3).
Oznaczmy $D^{(1)}(h) := \frac{4}{3}D\left(\frac{h}{2}\right) - \frac{1}{3}D(h)$.
Po zastosowaniu wzoru (3) otrzymamy:

$$L = D^{(1)}(h) + b_4 h^4 + b_6 h^6 + \cdots \quad (4)$$

i ogólnie:

$$D^{(k)}(h) := \frac{4^k D^{(k-1)}(\frac{h}{2}) - D^{(k-1)}(h)}{4^k - 1}$$

Ekstrapolacja Richardsona - schemat

Schemat M kroków ekstrapolacji Richardsona:

- Wybieramy h, np. $h = 1$.
- Obliczamy $D(n, 0) := D(h/2^n)$ $(0 \leqslant n \leqslant M)$.
- Dla $k = 1, 2, \ldots, M$ i $n = k, k+1, \ldots, M$ stosujemy wzór rekurencyjny:

$$D(n,k) := D(n,k-1) + \frac{D(n,k-1) - D(n-1,k-1)}{4k-1}$$
$$= \frac{4D(n,k-1) - D(n-1,k-1)}{4k-1} \quad (6)$$

Ekstrapolacja Richardsona

Ekstrapolacja Richardsona daje następującą trójkątną tablicę przybliżeń L:

$$\begin{array}{ccccc} D(0,0) & & & & \\ D(1,0) & D(1,1) & & & \\ D(2,0) & D(2,1) & D(2,2) & & \\ \vdots & \vdots & \vdots & \ddots & \\ D(M,0) & D(M,1) & D(M,2) & \cdots & D(M,M) \end{array}$$

Tablicę, przy założeniu, że funkcja D jest dana, można skonstruować przy użyciu algorytmu:

input h, M
for $n = 0$ **to** M **do**
    $D(n,0) \leftarrow D(h/2^n)$
end do
for $k = 1$ **to** M **do**
    for $n = k$ **to** M **do**
        $D(n,k) \leftarrow D(n,k-1) + \left[\frac{D(n,k-1)-D(n-1,k-1)}{4k-1}\right]$
        output $D(n,k)$
    end do
end do

Przykład 4

$f(x) = \operatorname{arc\,tg} x, \quad x = \sqrt{2}, \quad f'(x) = (x^2 - 1)^{-1} \quad \Rightarrow \quad f'(\sqrt{2}) = \frac{1}{3}$

n	D(n, 0)	D(n, 1)	D(n, 2)	D(n, 3)	D(n, 4)
0	0.3926991				
1	0.3487710	0.3341283			
2	0.3371938	0.3333348	0.3332819		
3	0.3342981	0.3333329	0.3333328	0.3333336	
4	0.3335748	0.3333336	0.3333337	0.3333337	0.3333337

Metody numeryczne
Wykład nr 9
Całkowanie numeryczne

Aneta Wróblewska

UMCS, Lublin

May 6, 2024

Cele całkowania numerycznego

- **Całkowanie numeryczne**, znane również jako **kwadratura numeryczna**, ma na celu znajdowanie przybliżonej wartości całek oznaczonych, zwłaszcza kiedy analityczne rozwiązania są trudne lub niemożliwe do uzyskania.
- Głównym celem jest obliczenie obszaru pod krzywą funkcji, co odpowiada sumowaniu nieskończenie małych prostokątów pod krzywą.
- Umożliwia efektywne i skuteczne rozwiązywanie problemów w inżynierii i naukach przyrodniczych, gdzie bezpośrednie metody są niepraktyczne.

Definicje i podstawy

- **Całkowanie numeryczne** to metoda obliczania przybliżonej wartości całki za pomocą dyskretnych sum.
- **Kwadratura** – tradycyjny termin używany w całkowaniu numerycznym, odnosi się do procesu obliczania wartości całek jednowymiarowych. Dwu- i wielowymiarowe całkowania nazywane są czasami **kubaturami**, choć nazwa kwadratura odnosi się również do całkowania w wyższych wymiarach.
- **Błąd całkowania numerycznego**: różnica między wartością przybliżoną a dokładną wartością całki.

Zastosowania całkowania numerycznego

- Całkowanie numeryczne jest stosowane w wielu dziedzinach, takich jak fizyka, inżynieria, ekonomia i finanse, do modelowania i symulacji systemów fizycznych, ekonomicznych i biologicznych.
- Jest niezbędne w przypadkach, gdzie nie można uzyskać rozwiązania analitycznego, np. w dynamice płynów, optyce czy w metodach elementów skończonych.
- Używane także do obliczania wartości oczekiwanych, prawdopodobieństw i innych parametrów statystycznych.

Kwadratura i metody całkowania

- **Metoda prostokątów**: najprostsza forma kwadratury, wykorzystuje sumę prostokątów do aproksymacji obszaru pod krzywą.
- **Metoda trapezów**: ulepszona metoda prostokątów, sumuje obszary trapezów zamiast prostokątów.
- **Metoda Simpsona**: używa parabol do aproksymacji obszaru pod krzywą, znacznie zwiększając dokładność w porównaniu do metody trapezów.
- Metody te są łatwe do implementacji i mogą być adaptowane do złożonych problemów całkowania.

W tym wykładzie zapoznamy się z podstawowymi metodami przybliżonego obliczania całek oznaczonych funkcji jednej zmiennej, tj. całek postaci

$$I = \int_a^b f(x)$$

Będziemy zakładać, że funkcja f jest przynajmniej ciągła w domkniętym przedziale $[a, b]$ (oznacza to automatycznie, że funkcja f jest ograniczona w tym przedziale).

Całkowanie

- Funkcja pierwotna $F(x)$ funkcji $f(x)$ jest funkcją spełniającą warunek
$$F'(x) = f(x).$$
- Jeśli $F(x)$ jest funkcją pierwotną funkcji $f(x)$, to $F(x) + C$, gdzie C jest dowolną stałą, też jest funkcją pierwotną funkcji $f(x)$.

Całka nieoznaczona

Klasa takich funkcji pierwotnych jest całką nieoznaczoną funkcji $f(x)$:

$$\int f(x)\,dx = F(x) + C.$$

Funkcje elementarne

- Przypomnijmy, że funkcjami elementarnymi są funkcje: stałe, potęgowe, wykładnicze, logarytmiczne, trygonometryczne i cyklometryczne oraz wszystkie funkcje otrzymane z nich za pomocą skończonej ilości działań arytmetycznych bądź działań składania funkcji.

Przykłady funkcji pierwotnych

- Funkcje pierwotne mogą nie być funkcjami elementarnymi. Na przykład:

$$F(x) = \int e^{-x^2}\, dx,$$

$$G(x) = \int \frac{\sin x}{x}\, dx,$$

$$H(x) = \int \frac{e^x}{\sqrt{x}}\, dx$$

nie są funkcjami elementarnymi, mimo że funkcje podcałkowe są elementarne.

Całka oznaczona

- Całka oznaczona funkcji $f(x)$ w granicach od a do b jest liczbą

$$\int_a^b f(x)\,dx = F(x)\Big|_a^b = F(b) - F(a),$$

gdzie $F(x)$ jest dowolną funkcją pierwotną funkcji $f(x)$.

- Zauważmy, że

$$\int_a^x f(t)\,dt = F(x)$$

jest pewną funkcją pierwotną funkcji $f(x)$.

Rozwinięcie funkcji podcałkowej w szereg potęgowy

Rozwinięcie funkcji podcałkowej w szereg potęgowy

Problem obliczenia całki można rozwiązać przez rozwiniecie funkcji w szereg potęgowy.

Przykład

Znajdźmy całkę z funkcji e^{-x^2}. Mamy rozwinięcie funkcji eksponencjalnej w szereg potęgowy:

$$e^x = \sum_{n=0}^{\infty} \frac{x^n}{n!}$$

dla $x \in (-1, 1)$. Zatem:

$$e^{-x^2} = \sum_{n=0}^{\infty} \frac{(-1)^n x^{2n}}{n!}.$$

Obliczanie całki z rozwinięcia funkcji

Stąd:

$$\int e^{-x^2}\,dx = \sum_{n=0}^{\infty} \int \frac{(-1)^n x^{2n}}{n!}\,dx = \sum_{n=0}^{\infty} \frac{(-1)^n x^{2n+1}}{(2n+1)n!},$$

$$\int e^{-x^2}\,dx \approx \sum_{m=0}^{n} \frac{(-1)^m x^{2m+1}}{(2m+1)m!}.$$

m musi być dostatecznie duże, aby błąd był dostatecznie mały.

Wartości funkcji $F(x)$

Niech

$$F(x) = \int_0^x e^{-t^2}\, dt.$$

W poniższej tabeli podane są wartości funkcji $F(x)$ danej wzorem (2) dla argumentów $x = 0.1, 0.2, \ldots, 1.0$, obliczone w kolejnych kolumnach programem Maxima oraz według wzoru (1) dla $m = 2$ i dla $m = 3$.

x	Maxima	$m = 2$	$m = 3$
0.1	0.0997	0.0998	0.0998
0.2	0.1974	0.1981	0.1981
$\vdots$	$\vdots$	$\vdots$	$\vdots$
1.0	0.7468	0.7667	0.7429

Kwadratury interpolacyjne

Kwadratury interpolacyjne

Metody Newtona-Cotesa – zbiór metod numerycznych całkowania, zwanego również kwadraturą.

- Dana jest funkcja $f(x)$ ciągła i ograniczona w przedziale $[a, b]$.
- Przedział $[a, b]$ dzielimy na skończoną liczbę podprzedziałów - kolejne punkty na osi OX:

$$a = x_0 < x_1 < x_2 < \ldots < x_i < x_{i+1} < \ldots < x_n = b$$

  gdzie $i = 0, 1, \ldots, n$.
- Zwykle punkty te rozmieszczone są równomiernie: $h = x_{i+1} - x_i = \text{const}$. W takim wypadku $h = \frac{b-a}{n}$. Dla węzłów nierówno oddalonych od siebie maja zastosowanie inne wzory np. kwadratura gaussowska.
- $\int_{x_0=a}^{x_n=b} f(x)\,dx = \sum_{i=0}^{n-1} \int_{x_i}^{x_{i+1}} f(x)\,dx$ (1)

Interpolacyjne przybliżenie całki

- Poszczególne składniki sumy oznaczymy sobie przez σ_i:

$$\sigma_i = \int_{x_i}^{x_{i+1}} f(x)\,dx$$

- Istotą metody kwadratur interpolacyjnych jest przybliżenie funkcji $f(x)$ w przedziale $[x_i, x_{i+1}]$ (lub odpowiednio poszerzonym) wzorem interpolacyjnym. Stąd:

$$\sigma_i \approx \int_{x_i}^{x_{i+1}} W(x)\,dx$$

gdzie $W(x)$ jest wielomianem interpolacyjnym.

Interpolacja Lagrange'a w całkowaniu numerycznym

Jeżeli mamy zdefiniowany zestaw równoodległych węzłów interpolacji:

$$a = x_0 < x_1 < x_2, \ldots, < x_{n-1} < x_n = b$$

dla funkcji $f(x)$, gdzie x_i są punktami, w których znamy wartości $f(x_i) = y_i$, to całkę:

$$\int_a^b f(x)\,dx$$

można przybliżyć przez:

$$\int_a^b L_n(x)\,dx$$

Interpolacja Lagrange'a w całkowaniu numerycznym

gdzie $L_n(x)$ jest wielomianem interpolacyjnym Lagrange'a stopnia co najwyżej n, aproksymującym funkcję $f(x)$ w węzłach interpolacji, tj.:

$$L_n(x_0) = y(x_0),\ L_n(x_1) = y(x_1),\ \ldots,\ L_n(x_n) = y(x_n)$$

Interpolacja Lagrange'a w całkowaniu numerycznym

Niech $h_n = \frac{b-a}{n}$ będzie długością kroku dzielącą dwa węzły interpolacji. Wprowadzając zmienną pomocniczą t, tak że $x = a + th$, można zapisać:

$$\lambda_i(x) = \lambda_i(a + th) = \prod_{\substack{j=0 \\ j \neq i}}^{n} \frac{t - j}{i - j} = g(t).$$

Interpolacja Lagrange'a w całkowaniu numerycznym

Wtedy przybliżenie całki daje się zapisać jako:

$$\int_a^b L_n(x)\,dx = \int_a^b \sum_{i=0}^{n} f(x_i)\cdot\lambda_i(x)\,dx = \sum_{i=0}^{n} f(x_i)\cdot\int_a^b \lambda_i(x)\,dx$$

$$x = a + t\cdot h, \quad f(x_i) = f(a + i\cdot h), \quad x_i = a + i\cdot h$$

Interpolacja Lagrange'a w całkowaniu numerycznym

Zmieniając zmienną oraz granice całkowania, otrzymujemy:

$$\int_a^b L_i(x)\,dx = h \cdot \int_0^n g(t)\,dt.$$

Ostatecznie, korzystając z aproksymacji Newtona-Cotesa dla $n+1$ równoodległych węzłów:

$$\int_a^b f(x)\,dx = \sum_{i=0}^{n} f(x_i) \cdot h \cdot \int_0^n \prod_{\substack{j=0 \\ j\neq i}}^{n} \frac{t-j}{i-j}\,dt.$$

Rodzaje wzorów Newtona-Cotesa

Wyróżniamy dwa główne rodzaje wzorów Newtona-Cotesa:

- **Otwarte**, które pomijają wartości funkcji w skrajnych punktach przedziału.
- **Zamknięte**, które uwzględniają wartości funkcji we wszystkich punktach, włącznie ze skrajnymi.

Zamknięty wzór Newtona-Cotesa

Dla zamkniętego wzoru Newtona-Cotesa rzędu n:

$$\int_a^b f(x)\,dx \approx \sum_{i=0}^{n} w_i f(x_i),$$

gdzie $x_i = h \cdot i + x_0$, a $h = \frac{x_n - x_0}{n}$, i w_i to wagi uzyskane z wielomianów bazowych Lagrange'a.

Przykładami wzoru zamkniętego są wzory metody trapezów i Simpsona.

Otwarty wzór Newtona-Cotesa

Dla otwartego wzoru Newtona-Cotesa rzędu n:

$$\int_a^b f(x)\,dx \approx \sum_{i=1}^{n-1} w_i f(x_i)$$

Wagi są wyznaczone podobnie jak w przypadku zamkniętego wzoru.

Przykładem wzoru otwartego jest wzory metody prostokątów.

Zastosowania i błędy wzorów Newtona-Cotesa

Wzory te umożliwiają przybliżenie wartości całki na podstawie wartości funkcji w dyskretnych punktach. Dokładność wzoru zależy od rzędu użytego wzoru oraz od rozmiaru kroku h. Błąd przybliżenia maleje z mniejszym krokiem h.

Wzory Newtona-Cotesa mogą być stosowane zarówno w formie otwartej, jak i zamkniętej, w zależności od dostępności danych w skrajnych punktach przedziału. Przy odpowiednim doborze kroku h i liczby podprzedziałów, można efektywnie przybliżyć wartość całki określonej funkcji.

Metoda prostokątów

Metoda prostokątów

Jeśli $W(x) = f(x_i)$ dla $x \in [x_i, x_{i+1}]$, funkcję podcałkową $f(x)$ przybliżamy wzorem interpolacyjnym ograniczonym do pierwszego składnika.
Oznacza to zastąpienie funkcji $f(x)$ na odcinku $[x_i, x_{i+1}]$ linią poziomą o wartości y_i. W przypadku węzłów równoodległych mamy zazwyczaj $y_i = f(x_i + h/2)$
Dla przybliżenia mamy:

$$\sigma_i = \int_{x_i}^{x_{i+1}} f(x)\,dx \approx \int_{x_i}^{x_{i+1}} y_i\,dx$$

Obliczając wartość całki:

$$\sigma_i \approx \int_{x_i}^{x_{i+1}} y_i\,dx = [y_i x]_{x_i}^{x_{i+1}} = y_i(x_{i+1} - x_i) = y_i h$$

Metoda prostokątów

Otrzymujemy zatem:

$$\int_a^b f(x)\,dx = h\sum_{i=0}^{n-1} y_i$$

Przybliżona wartość całki jest sumą prostokątów, odpowiadającą sumie prostokątów o podstawie h i wysokości y_i.

Metoda prostokątów - przykład

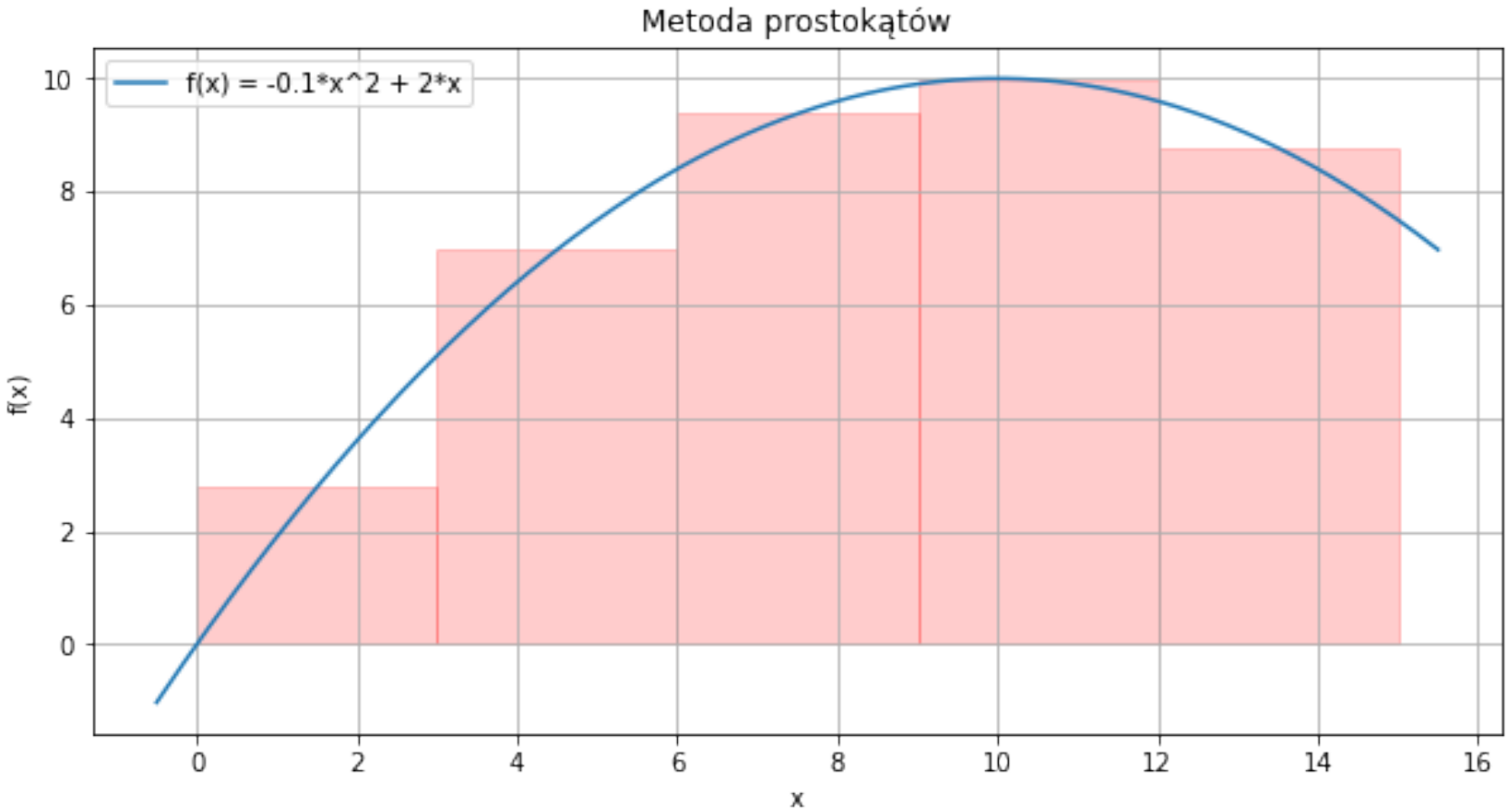
Dana jest funkcja

$$f(x) = -0.1x^2 + 2x$$

Obliczmy wartość całki oznaczonej w przedziale $[0, 15]$ metodą prostokątów, dzieląc przedział na odcinki długości 3.

Uzyskujemy w ten sposób wartość całki równą 113.625. Dokładny wynik to 112.5.

Metoda prostokątów - przykład



Metoda trapezów

Metoda trapezów polega na przybliżeniu funkcji podcałkowej wzorem interpolacyjnym z dokładnością do dwóch pierwszych składników:

- $W(x) = f(x_i) + q\Delta y_i$, gdzie Δy_i to różnica skończona pierwszego rzędu.
- Oznacza to aproksymację prostą przechodzącą przez punkty $(x_i, f(x_i))$, $(x_{i+1}, f(x_{i+1}))$.

Metoda trapezów

Wyprowadzenie wzoru sprowadza się to do policzenia powierzchni trapezu o podstawach $f(x_i), f(x_{i+1})$ i wysokości h:

$$\sigma_i = \frac{1}{2}h(y_{i+1} + y_i)$$

Sumując kolejne pola σ_i dla całego przedziału $[a, b]$ otrzymujemy:

$$\int_a^b f(x)\,dx = \frac{1}{2}h\sum_{i=0}^{n-1}(y_{i+1} + y_i) = h\left[\frac{1}{2}(y_0 + y_n) + \sum_{i=1}^{n-1} y_i\right]$$

Przybliżona wartość całki jest sumą trapezów.

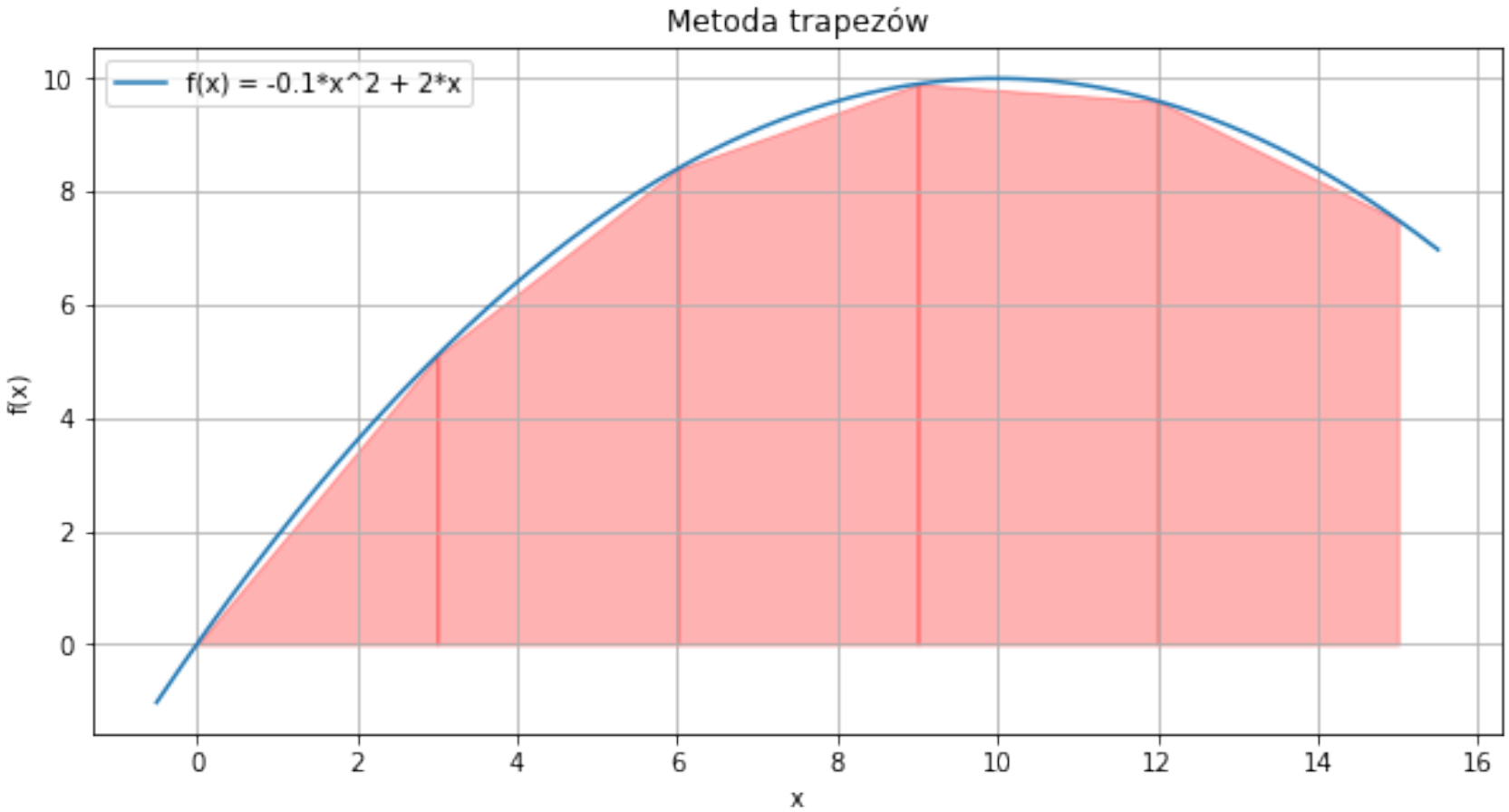
Metoda trapezów

Powyższy wzór jest dokładny, jeżeli f jest wielomianem stopnia co najwyżej piewszego. W innych przypadkach błąd przybliżenia wynosi:

$$\delta = \frac{1}{12}|f''(\xi)|(b-a)^3,$$

gdzie ξ jest pewną liczbą z przedziału (a, b), zaś $h = (b-a)/n$.

Metoda trapezów - przykład



Metoda Simpsona

Metoda Simpsona - wstęp

Metoda Simpsona pozwala uzyskać bardziej precyzyjne wyniki niż metoda trapezów.

- Pola trapezów są zastępowane polami pod parabolami.
- Każde dwa pola trapezów są zastępowane jednym polem pod parabolą.
- Parabola jest wyznaczona przez trzy punkty: u, v i w.

$$\int_a^b f(x)\,dx \approx \sum_{k=0}^{n/2-1} \int_{x_{2k}}^{x_{2k+2}} W(x)\,dx$$

Metoda Simpsona - przybliżenie

Przybliżenie wartości całki oznaczonej S_n jest sumą pól pod parabolami:

$$S_n = \frac{h}{3}\Big[f(x_0) + 4(f(x_1) + f(x_3) + \ldots + f(x_{n-1})) + 2(f(x_2) + f(x_4) + \ldots + f(x_{n-2})) + f(x_n)\Big]$$

gdzie:

- $h = \frac{b-a}{n}$
- $x_i = a + ih$ dla $i = 0, 1, \ldots, n$
- n jest parzyste

Metoda Simpsona - błąd

Błąd przybliżenia wynosi:

$$\epsilon = \left| f^{(4)}(\xi) \right| \frac{(b-a)h^4}{180}$$

gdzie:

- $f^{(4)}(\xi)$ to czwarta pochodna funkcji f w punkcie $\xi \in (a, b)$
- $h = \frac{b-a}{n}$

Wzór z założenia jest dokładny dla wielomianów stopnia co najwyżej 3 stopnia.

Metody numeryczne
Wykład nr 10
Miejsca zerowe wielomianów

Aneta Wróblewska

UMCS, Lublin

May 13, 2024

Wprowadzenie

Poszukiwanie miejsc zerowych wielomianów jest fundamentalnym problemem w matematyce stosowanej, fizyce i inżynierii. Miejsce zerowe wielomianu to wartość, dla której wielomian przyjmuje wartość zero. Znalezienie tych miejsc ma kluczowe znaczenie w analizie równań różniczkowych, optymalizacji oraz w naukach inżynieryjnych.

Wprowadzenie

Główne zagadnienia wykładu:

- Algorytm Hornera
- Wpływ zaburzeń współczynników
- Metoda Laguerre'a
- Obniżanie stopnia wielomianu i wygładzanie

Algorytm Hornera

Algorytm Hornera to efektywna metoda obliczania wartości wielomianu przy danej wartości zmiennej oraz wyznaczania jego pochodnych. Dzięki swojej strukturze pozwala na szybkie obliczenie wartości wielomianu, redukując liczbę mnożeń i dodawań, co jest szczególnie użyteczne w poszukiwaniu pierwiastków.

Wpływ zaburzeń współczynników

Wielomiany są bardzo wrażliwe na zaburzenia ich współczynników, co może prowadzić do znaczących zmian w lokalizacji ich miejsc zerowych. Zrozumienie tej wrażliwości jest kluczowe przy analizie błędów numerycznych oraz w przypadkach, gdy dane wejściowe są obarczone niepewnością.

Metoda Laguerre'a

Metoda Laguerre'a to jedna z zaawansowanych technik numerycznych służąca do znajdowania pierwiastków wielomianów. Jest to iteracyjna metoda, która może być stosowana do wszystkich typów wielomianów i jest znana z szybkiej zbieżności, szczególnie przy odpowiednim dobieraniu punktów startowych.

Obniżanie stopnia wielomianu i wygładzanie

Redukcja stopnia wielomianu oraz wygładzanie są technikami przetwarzania przedwstępnego, które mogą ułatwić znalezienie miejsc zerowych. Obniżanie stopnia polega na eliminowaniu terminów o niskim wpływie, podczas gdy wygładzanie może pomóc zredukować efekt oscylacji, szczególnie w wielomianach wysokiego stopnia.

Podstawowe Twierdzenie Algebry i schemat Hornera

Podstawowe Twierdzenie Algebry

Rozwiązanie równań wielomianowych

$$P_n(z) = a_n z^n + a_{n-1} z^{n-1} + \cdots + a_1 z + a_0 = 0 \qquad (1)$$

jest jednym z niewielu przypadków, w których interesuje nas znajomość wszystkich pierwiastków równania nieliniowego.

Podstawowe Twierdzenie Algebry

Wielomian stopnia n ma na płaszczyźnie zespolonej dokładnie n pierwiastków, przy czym pierwiastki wielokrotne liczy się z ich krotnościami.

Uwaga

W przypadku rzeczywistym wielomian stopnia n może nie mieć wcale pierwiastków, a jeśli ma, to jest ich co najwyżej n. Natomiast każdy wielomian rzeczywisty stopnia nieparzystego ma przynajmniej jeden pierwiastek rzeczywisty (wynika to z faktu, że granice niewłaściwe wielomianu rzeczywistego stopnia nieparzystego są różnych znaków, a także z faktu, że wielomian jako funkcja ciągła ma własność Darboux – a zatem musi przyjąć wartość pośrednią 0).

Podstawowe Twierdzenie Algebry

- **Charakter twierdzenia:** Podstawowe twierdzenie algebry nie jest konstruktywne — nie wskazuje sposobu poszukiwania pierwiastków — jednak dostarcza informacji o tym, czego szukać.
- **Przydatność teoretyczna:** Jest to częsty przykład, gdzie silny, chociaż niekonstruktywny wynik teoretyczny znacznie ułatwia stosowanie metod numerycznych.

Schemat Hornera

- **Znaczenie:** Schemat Hornera to efektywny sposób obliczania wartości wielomianu oraz jego pochodnych w danym punkcie.
- **Koncepcja:** Polega na przekształceniu postaci wielomianu tak, aby minimalizować liczbę operacji arytmetycznych potrzebnych do jego wartości.

Schemat Hornera - koncepcja

- Przekształcamy wielomian $P_n(z)$ w postać:

$$P_n(z) = (((a_n z + a_{n-1})z + a_{n-2})z + \cdots + a_1)z + a_0 \qquad (2)$$

- Pamiętamy, że z każdym dodaniem kolejnego wyrazu redukujemy stopień wielomianu o jeden.
- Dzięki temu, potrzebujemy jedynie n mnożeń i n dodawań, aby obliczyć wartość wielomianu w punkcie z.

Schemat Hornera - algorytm

Algorithm 1: Algorytm schematu Hornera

Wejście: Wielomian

$$P_n(z) = a_n z^n + a_{n-1} z^{n-1} + \cdots + a_1 z + a_0 = 0,$$

wartość z

Wyjście: Wartość wielomianu $P_n(z)$

```
1 p ← a_n;
2 for i ← n − 1 to 0 do
3 |   p ← p · z + a_i;
4 end
5 return p;
```

Schemat Hornera - przykład

Przykład: Rozważmy wielomian $P(z) = 3z^3 + 2z^2 - z + 5$ i chcemy obliczyć jego wartość dla $z = 2$.

Krok 1: $p \leftarrow 3$
Krok 2: $p \leftarrow p \cdot 2 + 2 = 3 \cdot 2 + 2 = 8$
Krok 3: $p \leftarrow p \cdot 2 - 1 = 8 \cdot 2 - 1 = 15$
Krok 4: $p \leftarrow p \cdot 2 + 5 = 15 \cdot 2 + 5 = 35$

Wartość wielomianu $P(z)$ dla $z = 2$ wynosi 35.

Schemat Hornera - podsumowanie

- **Zalety:** Schemat Hornera pozwala na efektywne obliczanie wartości wielomianu przy minimalnym zużyciu zasobów.
- **Zastosowania:** Jest szeroko stosowany w obliczeniach numerycznych oraz analizie numerycznej.
- **Wnioski:** Znając schemat Hornera, możemy zwiększyć wydajność naszych obliczeń, szczególnie w przypadku wielomianów o dużych stopniach.

Wpływ zaburzeń współczynników na miejsca zerowe wielomianów

Wpływ zaburzeń współczynników

W obliczeniach praktycznych współczynniki wielomianów, których pierwiastków poszukujemy, rzadko znamy w sposób dokładny. Najczęściej są one wynikiem jakichś poprzednich obliczeń, są zatem obarczone pewnymi błędami.
Jaki jest wpływ błędów współczynników na wartości znalezionych numerycznie miejsc zerowych?

Wpływ zaburzeń współczynników (cd.)

Niech dokładny wielomian ma postać jak w równaniu 1 i niech z_0 będzie jego dokładnym pierwiastkiem. Przypuśćmy dalej, że dokładne wartości współczynników a_k nie są znane - zamiast tego znamy wartości przybliżone $\tilde{a}_k = a_k + \delta_k$, przy czym $\forall k : |\delta_k| \ll 1$. Poszukujemy pierwiastków wielomianu zaburzonego:

$$\widetilde{P}_n(z) = \widetilde{a}_n z^n + \widetilde{a}_{n-1} z^{n-1} + \cdots + \widetilde{a}_1 z + \widetilde{a}_0 = 0 \qquad (3)$$

Wpływ zaburzeń współczynników (cd.)

Spodziewamy się, że miejsce zerowe wielomianu 3 jest zaburzonym miejscem zerowym wielomianu1: $\tilde{P}_n(\tilde{z}_0) = 0 \rightarrow \tilde{z}_0 = z_0 + \varepsilon$, $|\varepsilon| \ll 1$.
Mamy

$$\begin{aligned} 0 &= P_n(z_0) \\ &= a_n z_0^n + a_{n-1} z_0^{n-1} + \cdots + a_1 z_0 + a_0 \\ &= (\widetilde{a}_n - \delta_n)(\widetilde{z}_0 - \varepsilon)^n + (\widetilde{a}_{n-1} - \delta_{n-1})(\widetilde{z}_0 - \varepsilon)^{n-1} + \ldots \\ &+ (\widetilde{a}_1 - \delta_1)(\widetilde{z}_0 - \varepsilon) + (\widetilde{a}_0 - \delta_0). \end{aligned} \tag{4}$$

Wpływ zaburzeń współczynników (cd.)

Zauważmy, że

$$(\widetilde{z}_0 - \varepsilon)^k = \sum_{l=0}^{k} \binom{k}{l} \widetilde{z}_0^{k-l} (-1)^l \varepsilon^l \approx \widetilde{z}_0^k - k\widetilde{z}_0^{k-1}\varepsilon, \qquad (5)$$

gdyż wyższe potęgi ε możemy zaniedbać. Zaniedbujemy również iloczyny $\delta_k \varepsilon$.

Wpływ zaburzeń współczynników (cd.)

Zatem

$$
\begin{aligned}
0 &= P_n(z_0) \\
&\approx \widetilde{a}_n \widetilde{z}_0^n - n\widetilde{a}_n \widetilde{z}_0^{n-1}\varepsilon + \delta_n \widetilde{z}_0^n + \\
&\qquad + \widetilde{a}_{n-1}\widetilde{z}_0^{n-1} - (n-1)\widetilde{a}_{n-1}\widetilde{z}_0^{n-2}\varepsilon + \delta_{n-1}\widetilde{z}_0^{n-1} + \dots \\
&= \sum_{k=0}^{n} \widetilde{a}_k \widetilde{z}_0^k - \left(\sum_{k=1}^{n} k\widetilde{a}_k \widetilde{z}_0^{k-1} \right) \varepsilon + \sum_{k=0}^{n} \delta_k \widetilde{z}_0^k . \qquad (6)
\end{aligned}
$$

Wpływ zaburzeń współczynników (cd.)

Rozwijając wyrażenie, otrzymujemy:

$$\widetilde{P}_n(\widetilde{z}_0) = 0 + \left(\sum_{k=1}^{n} k\widetilde{a}_k \widetilde{z}_0^{k-1} \right) \varepsilon - \sum_{k=0}^{n} \delta_k \widetilde{z}_0^k$$

Ostatecznie otrzymujemy następujące oszacowanie wpływu zaburzeń współczynników na zaburzenie miejsca zerowego wielomianu:

$$|\varepsilon| \approx \frac{|\sum_{k=0}^{n} \delta_k \tilde{z}_0^k|}{|\tilde{P}_n'(\tilde{z}_0)|} \tag{7}$$

Przykład Wilkinsona

Przykład Wilkinsona przedstawia wielomian:

$$W(z) = (z+1)(z+2)\dots(z+20). \tag{8}$$

Jego miejsca zerowe to liczby całkowite ujemne: $-1, -2, \dots, -20$. Załóżmy, że zaburzamy tylko jeden współczynnik w wielomianie (8): $\delta_{19} = 2^{-23} \approx 10^{-7}$, $\delta_{k\neq 19} = 0$. Jak zmieni się położenie miejsca zerowego $z_0 = -20$?

Przykład Wilkinsona (cd.)

Obliczamy $W'(-20) = -19!$. Oszacowanie (7) daje:

$$|\varepsilon| \approx \frac{10^{-7} \cdot 20^{19}}{19!} \approx 4.4. \tag{9}$$

Zaburzenie miejsca zerowego jest siedem rzędów wielkości większe niż zaburzenie pojedynczego współczynnika! W rzeczywistości miejsca zerowe tak zaburzonego wielomianu stają się nawet zespolone. Zagadnienie znajdowania miejsc zerowych wielomianów może być źle uwarunkowane!

Zaburzenia wielokrotnych miejsc zerowych

Oszacowanie (7) załamuje się dla miejsc zerowych o krotności większej od jeden, co oznacza, że znikają także pochodne wielomianu. To sugeruje, że wielokrotne miejsce zerowe może zmienić się drastycznie po niewielkim zaburzeniu współczynników wielomianu.

Zaburzenia wielokrotnych miejsc zerowych (cd.)

Rozważmy przykład wielomianu:

$$\begin{aligned} Q(x) &= 39205740x^6 - 147747493x^5 + 173235338x^4 + 2869080x^3 - \\ &\quad 158495872x^2 + 118949888x - 28016640 \\ &= 17^3 \cdot 19 \cdot 20 \cdot 21 \left(x + \frac{20}{21}\right) \left(x - \frac{16}{17}\right)^3 \left(x - \frac{18}{19}\right) \left(x - \frac{19}{20}\right). \end{aligned} \tag{10}$$

Zaburzenia wielokrotnych miejsc zerowych (cd.)

Znalezienie miejsc zerowych tego wielomianu w postaci iloczynowej jest trywialne. Jednak numeryczne znalezienie miejsc zerowych tego samego wielomianu w postaci ogólnej może być bardzo trudne.

Zaburzenia wielokrotnych miejsc zerowych (cd.)

Mała zmiana w powyższym wielomianie , np. zwiększenie lub zmniejszenie wyrazu wolnego o 1, może spowodować przesunięcie potrójnych miejsc zerowych z osi rzeczywistej na płaszczyznę zespoloną.

Zaburzenia wielokrotnych miejsc zerowych (cd.)

W praktyce numerycznej, gdy otrzymujemy grupę miejsc zerowych bliskich "numerycznych miejsc zerowych", czasami trudno jest określić, czy są one naprawdę różne, czy też są one efektem skończonej dokładności, z jaką znamy współczynniki, i reprezentują "rozszczepione" miejsce wielokrotne.

Poszukiwanie miejsc zerowych wielomianów

Co robić przy poszukiwaniu miejsc zerowych?

W kontekście poszukiwania miejsc zerowych wielomianów, niezbędne są dwie kwestie:

- Specjalistyczna metoda numeryczna do wyznaczania pierwiastków wielomianów. Taką metodą jest **metoda Laguerre'a**.
- Skuteczna strategia postępowania, która obejmuje:
  - Obniżanie stopnia wielomianu (tzw. deflacja) poprzez dzielenie wielomianu przez $(z - z_0)$, gdzie z_0 jest znalezionym pierwiastkiem.
  - Wygładzanie odkrytych miejsc zerowych za pomocą pierwotnego wielomianu, jeszcze przed jego deflacją.

Wprowadzenie do metody Laguerre'a

Niech $P_n(z)$ będzie wielomianem stopnia n. **Metoda Laguerre'a** określona jest przez następującą iterację:

$$z_{i+1} = z_i - \frac{nP_n(z_i)}{P'_n(z_i) \pm \sqrt{(n-1)\left((n-1)(P'_n(z_i))^2 - nP_n(z_i)P''_n(z_i)\right)}}, \tag{11}$$

gdzie znak w mianowniku wybierany jest tak, aby maksymalizować jego wartość bezwzględną.

Zbieżność metody Laguerre'a

- Jeśli wszystkie pierwiastki P_n są pojedyncze i rzeczywiste, metoda jest zbieżna sześciennie dla dowolnego rzeczywistego przybliżenia początkowego, tj. jeśli $|z_i - \bar{z}| < \epsilon \ll 1$, to $|z_{i+1} - \bar{z}| \sim \epsilon^3$, gdzie $\bar{z}$ jest poszukiwanym pierwiastkiem.
- W ogólności metoda jest zbieżna sześciennie do wszystkich pojedynczych pierwiastków (rzeczywistych i zespolonych).

Ograniczenia i zalety metody Laguerre'a

- Metoda jest zbieżna liniowo do wielokrotnych pierwiastków.
- Metoda może nie zbiegać się w rzadkich przypadkach; stagnację można przełamać wykonując jeden-dwa kroki metody Newtona, a następnie wracając do metody Laguerre'a.
- Jest to metoda preferowana do wyszukiwania pierwiastków wielomianów, zarówno rzeczywistych, jak i zespolonych.

Zastosowania i porównania

- Metoda Laguerre'a może być również stosowana do poszukiwania miejsc zerowych funkcji analitycznych rozwijalnych lokalnie w szereg potęgowy do rzędu n.
- Metoda Laguerre'a jest podobna do metody opartej o rozwinięcie w szereg Taylora do drugiego rzędu, ale uwzględnia stopień wielomianu, co czyni ją bardziej efektywną.

Specyfika metody

- Metoda wymaga obliczania drugiej pochodnej, co jest prostym zadaniem dla wielomianów.
- Nawet rozpoczynając od rzeczywistych punktów początkowych, metoda może prowadzić do zespolonych miejsc zerowych, co jest typowe dla wielomianów.

Deflacja - obniżanie stopnia wielomianu

Podczas numerycznego poszukiwania pierwiastków wielomianu może dojść do sytuacji, gdy różne próby zbiegają się do tego samego, już znalezionego pierwiastka. Aby tego unikać, stosujemy **deflację**, czyli technikę obniżania stopnia wielomianu poprzez faktoryzację:

$$P_n(z) = (z - z_1)P_{n-1}(z),$$

gdzie z_1 to pierwiastek wcześniej znaleziony dla wielomianu $P_n(z)$. Następnie szukamy pierwiastka dla nowo powstałego wielomianu $P_{n-1}(z)$.

Wygładzanie miejsc zerowych

Drobne zaburzenia współczynników wielomianu, które mogą powstać na skutek ograniczonej precyzji obliczeń, mogą znacząco wpłynąć na znalezione pierwiastki. Dlatego stosuje się **proces wygładzania**:

- Używamy pełnego, niepodzielonego wielomianu $P_n(z)$ do poprawy znalezionych miejsc zerowych.
- Załóżmy, że z_2 to przybliżone miejsce zerowe P_{n-1}. Używamy tego jako punktu startowego dla metody Laguerre'a.

Zastosowanie metody Laguerre'a

Zakładając, że numerycznie znalezione z_2 leży blisko prawdziwego pierwiastka, możemy spodziewać się szybkiej zbieżności przy użyciu metody Laguerre'a na wielomianie P_n. W rezultacie uzyskujemy dokładniejsze i wygładzone miejsce zerowe z_2.

Kontynuacja faktoryzacji

Po znalezieniu i wygładzeniu z_2 dla P_n, proces jest kontynuowany:

$$P_{n-1}(z) = (z - z_2)P_{n-2}(z)$$

co oznacza, że $P_n(z) = (z - z_1)(z - z_2)P_{n-2}(z)$. Następnie powtarzamy procedurę dla $P_{n-2}(z)$.

Deflacja wielomianu

Załóżmy, że z_0 jest pierwiastkiem wielomianu $P_n(z)$. Wtedy musi być spełniony związek:

$$\begin{aligned}(z - z_0) \cdot P_{n-1}(z) &= (z - z_0)\left(b_{n-1}z^{n-1} + b_{n-2}z^{n-2} + \cdots + b_1 z + b_0\right) \\ &= P_n(z). \end{aligned} \tag{12}$$

To równanie pozwala na obliczenie współczynników b_k rozwińmy to jako:

$$\begin{aligned} b_{n-1} &= a_n \\ -z_0 b_{n-1} + b_{n-2} &= a_{n-1} \\ &\dots \\ -z_0 b_1 + b_0 &= a_1 \\ -z_0 b_0 &= a_0 \end{aligned} \tag{13}$$

Układ równań do deflacji

Wynikający układ równań liniowych z powyższego rozkładu można rozwiązać metodą podstawienia w przód:

$$\begin{bmatrix} 1 & 0 & 0 & \dots & 0 \\ -z_0 & 1 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \dots & 1 & 0 \\ 0 & 0 & \dots & -z_0 & 1 \end{bmatrix} \begin{bmatrix} b_{n-1} \\ b_{n-2} \\ \vdots \\ b_1 \\ b_0 \end{bmatrix} = \begin{bmatrix} a_n \\ a_{n-1} \\ \vdots \\ a_2 \\ a_1 \end{bmatrix}. \tag{14}$$

Strategia postępowania

Załóżmy, że znaleźliśmy już miejsca zerowe $z_1, z_2, \ldots, z_k$ wielomianu $P_n(z)$, używając deflacji do redukcji stopnia wielomianu, otrzymując nowy wielomian $P_{n-k}(z)$. Następnie:

1. Rozpoczynamy z dowolnym przybliżeniem i stosujemy metodę Laguerre'a do znalezienia kolejnego pierwiastka $\tilde{z}_{k+1}$ wielomianu P_{n-k}.
2. W celu wygładzenia miejsce zerowego używamy $\tilde{z}_{k+1}$ jako warunku początkowego do ponownego użycia metody Laguerre'a na pełnym wielomianie P_n, dla szybkiego uzyskania dokładniejszego wyniku z_{k+1}.
3. Wielomian $P_{n-k}(z)$ jest następnie faktoryzowany do $P_{n-k-1}(z)$, kontynuując proces aż do osiągnięcia wielomianu stopnia 2.

Końcowe uwagi

- Jeśli pierwotny wielomian $P_n(z)$ ma rzeczywiste współczynniki, jego pierwiastki są rzeczywiste lub tworzą sprzężone pary zespolone. Jeśli znajdziemy zespolony pierwiastek $z_k = x_k + iy_k$, to $z_{k+1} = x_k - iy_k$ również jest pierwiastkiem.
- Gdy pierwotny wielomian ma całkowite współczynniki, warto najpierw sprawdzić, czy posiada pierwiastki wymierne, zanim przystąpimy do numerycznych obliczeń.

Przykład

- Rozważany wielomian: $P(x) = (x-1)^3 = x^3 - 3x^2 + 3x - 1$
- Cel: Znalezienie wszystkich pierwiastków wielomianu.
- Metoda: Użycie metody Laguerre'a oraz techniki deflacji do systematycznego obniżania stopnia wielomianu i izolowania pierwiastków.

Metoda Laguerre'a

- Metoda Laguerre'a jest używana do efektywnego znajdowania pojedynczego pierwiastka wielomianu.
- Startujemy z przybliżonego pierwiastka, a metoda iteracyjnie poprawia to przybliżenie.
- Dla wielomianu $P(x)$, iteracja wygląda następująco:

$$z_{i+1} = z_i - \frac{nP(z_i)}{P'(z_i) \pm \sqrt{(n-1)[(n-1)(P'(z_i))^2 - nP(z_i)P''(z_i)]}}$$

- Wybór znaku w mianowniku zależy od tego, który daje większy moduł mianownika.

Przykład cd

Pierwsza iteracja i deflacja

- Startowy punkt: $z_0 = 1.5$ (przybliżenie)
- Znaleziono pierwiastek: $z = 1.0$ (dla przykładu, w metodzie Laguerre'a ustalona została maksymalna liczba iteracji 100 w każdym kroku)
- Po znalezieniu pierwiastka $z1 = 1.0$, stosujemy deflację:

$$P(x) = (x - 1)(x^2 - 2x + 1)$$

Przykład cd

Druga iteracja i kolejna deflacja

- Teraz bierzemy $P(x) = x^2 - 2x + 1$ i szukamy kolejnych pierwiastków.
- Ponownie znajdujemy $z = 1.0$
- Deflacja daje nam:

$$P(x) = (x - 1)(x - 1)(x - 1)$$

Przykład cd

Ostatnia iteracja

- Dla wielomianu $P(x) = x - 1$, pierwiastek jest oczywisty: $z = 1.0$
- Wszystkie pierwiastki wielomianu to $1.0, 1.0, 1.0$
- Proces jest zakończony, wszystkie pierwiastki zostały znalezione.

Przykład cd

Podsumowanie

- Wszystkie pierwiastki wielomianu $(x-1)^3$ zostały skutecznie znalezione za pomocą metody Laguerre'a i deflacji.
- Metoda pozwala na efektywne izolowanie i precyzyjne wyznaczanie pierwiastków, nawet jeśli są one wielokrotne.

Metody numeryczne
Wykład nr 11
Generatory liczb pseudolosowych

Aneta Wróblewska

UMCS, Lublin

May 28, 2024

Losowość, liczby losowe i pseudolosowe oraz ich wykorzystanie

Losowość

- W naturze, technice, ekonomii i życiu społecznym często spotykamy zjawiska, które wydają się być losowe.
- Trudność w przewidywaniu ich przyszłych zachowań lub określeniu przyczyn wynika z braku informacji, błędów w obserwacji, czy ograniczeń technicznych w dostępie do danych.
- Przyczyny losowości mogą być związane z właściwościami fizycznymi zjawisk lub ich złożonością, uniemożliwiającą modelowanie deterministyczne.

Losowość w matematyce i kryptografii

- Rozkład liczb pierwszych wśród liczb naturalnych.Chociaż można określić średnią częstość ich występowania, dokładne rozmieszczenie liczb pierwszych jest nieznane i wydaje się losowe. Pojawienie się liczby pierwszej w ciągu liczb naturalnych jest trudne do przewidzenia bez szczegółowej analizy.
- Znajdują zastosowanie w statystycznych badaniach reprezentatywnych, kontroli jakości, badaniach rynkowych oraz w naukach eksperymentalnych do planowania eksperymentów.
- Liczby losowe są niezbędne w metodach Monte Carlo, używanych do obliczeń prawdopodobieństw i optymalizacji.
- W kryptografii, liczby losowe pełnią kluczową rolę jako klucze w szyfrach, zwiększając bezpieczeństwo przesyłanych informacji.

Losowość w symulacjach i systemach komunikacyjnych

- Symulacje komputerowe używają liczb losowych do imitacji realnych procesów opisanych równaniami, uwzględniając losowe czynniki wpływające na te zjawiska.
- Badania te często stanowią jedyną metodę ilościowej analizy złożonych procesów, które nie mogłyby być inaczej zbadane.
- Liczby losowe tworzą złudzenie realizmu w grach komputerowych, symulatorach treningowych oraz w grach strategicznych.
- Rozwój technologii komunikacyjnych, jak telefonia komórkowa i sieci komputerowe, intensyfikuje potrzebę skutecznych generatorów liczb losowych.

Ciąg losowy

- **Ciąg** liczbowy nazywamy **losowym**, jeśli nie istnieje krótszy algorytm, który by go opisał niż sam ciąg.
- Z ciągu takiego nie można wywnioskować żadnych reguł umożliwiających jego odtworzenie bez znajomości wszystkich jego elementów.
- Nie jest też możliwe przewidzenie żadnego elementu ciągu na podstawie pozostałych.

Ciągi pseudolosowe

- Mimo, że istnieją reguły opisujące ciąg, nasza niewiedza może uniemożliwić ich identyfikację.
- Taki **ciąg** nazywamy **pseudolosowym** i w wielu przypadkach może być traktowany jak ciąg losowy.
- W sytuacjach praktycznych pseudolosowe ciągi liczb mogą efektywnie symulować losowe zachowania.
- W przeciwieństwie do prawdziwie losowych, liczby pseudolosowe generowane są przez algorytmy deterministyczne.
- Ważne jest, aby generowane ciągi liczb były nieprzewidywalne i miały właściwości statystyczne podobne do ciągów losowych.

Przykłady generatorów ciągów liczb pseudolosowych

- **LCG (Linear Congruential Generator)**: Jedna z najstarszych i najprostszych metod generowania liczb pseudolosowych.
- **Mersenne Twister**: Znany z bardzo długiego okresu i wysokiej jakości generowanych ciągów.
- **Cryptographically Secure Pseudorandom Number Generators (CSPRNG)**: Generatory o wysokiej nieprzewidywalności, stosowane w kryptografii.

Historyczne metody otrzymywania liczb losowych

Wprowadzenie do historycznych metod liczb losowych

- Od dawna istnieje zapotrzebowanie na liczby losowe w badaniach statystycznych.
- Jednymi z pierwszych źródeł liczb losowych były tablice liczb losowych.
- Przykłady historycznych zbiorów liczb losowych obejmują różne techniki ich generacji.

Wczesne tablice liczb losowych

- W 1927 roku L.H. Tippett opublikował pierwszą tablicę losowych cyfr, składającą się z 41600 cyfr pochodzących z danych spisu powszechnego w Wielkiej Brytanii.
- W 1939 roku R.A. Fisher i F. Yates wydali tablicę 15000 losowych cyfr, które zaczerpnęli z cyfr od 15 do 19 z tablic logarytmicznych.
- Kendall, Babington i Smith w tym samym roku zaprezentowali tablicę 100000 cyfr losowych uzyskanych za pomocą "elektrycznej ruletki".

Tablice liczb losowych w Polsce i RAND Corporation

- W 1951 roku w Polsce GUS opracował własną tablicę liczb losowych, korzystając z pasków drukujących maszyn liczbowych.
- W 1955 roku RAND Corporation stworzyła tablicę miliona cyfr losowych, używając impulsów binarnych, co umożliwiało łatwe zastosowanie w obliczeniach komputerowych.
- Wady tablic liczb losowych obejmowały ograniczoną długość i konieczność rozwoju algorytmów do generowania nowych ciągów losowych.

Generowanie ciągów liczb losowych na podstawie tablicy cyfr losowych

1. Wybór losowej pięciocyfrowej liczby z tablicy.
2. Modyfikacja pierwszej cyfry liczby modulo 2. Tak zmieniona liczba pięciocyfrowa wskazuje numer wiersza w tablicy.
3. Zredykowana dwucyfrowego końcówka liczby modulo 50 wskazuje numer kolumny.
4. Proces rozpoczyna się od wybranej pozycji w tablicy, co tworzy losowy ciąg.

Współczesne metody otrzymywania liczb losowych

Generatory fizyczne liczb losowych

- Współczesne metody generacji liczb losowych dzielą się na algorytmiczne i fizyczne.
- Algorytmy matematyczne pozwalają na wielokrotne otrzymanie tego samego ciągu pseudolosowego.
- Generatory fizyczne opierają się na mierzalnych parametrach procesów fizycznych, które zachodzą w sposób losowy.

Generatory fizyczne

Przykłady generatorów fizycznych

- Mechaniczne urządzenia losujące, takie jak moneta, kostka do gry, czy ruletka.
- Licznik Geigera, mierzący promieniowanie jądrowe, które zachodzi losowo.
- Elektroniczne liczniki impulsów, np. z dysków komputerowych, monitorów czy kart dźwiękowych.
- Urządzenia generujące losowe bity z klawiatury lub arytmometru w intensywnie używanych komputerach.
- Specjalnie skonstruowane elektroniczne urządzenia generujące liczby losowe, np. z diod szumowych, często dostępne jako karty komputerowe.

Ważność testowania generatorów fizycznych

- Każdy wygenerowany ciąg liczb losowych musi być testowany przed użyciem, aby zapewnić, że zachowuje on właściwości losowości.
- W przypadku awarii urządzenia, wygenerowane ciągi mogą stracić swoje losowe właściwości.
- Dla celów kryptograficznych, ciągi liczb losowych powinny być zapisywane w dwóch kopiah na zewnętrznych nośnikach danych.

Generatory algorytmiczne

Podstawą algorytmicznego generowania liczb losowych jest uzyskanie ciągu liczb całkowitych z przedziału [1;M] w sposób równomierny.

Z losowych liczb całkowitych $X_i, i = 1, 2, \ldots$ o równomiernym rozkładzie w [1;M] uzyskuje się liczby $R_i, i = 1, 2, \ldots$ o ciągłym rozkładzie równomiernym na [0; 1] przez przekształcenie $R_i = \frac{X_i}{M}$.

Generatory kongruencyjne

Najbardziej znanym sposobem generowania liczb pseudolosowych jest metoda opracowana przez Lehmera w 1951 zwana **liniowym generatorem kongruentnym** (ang. LCG - Linear Congruential Generator). Można wyróżnić dwa podstawowe wzory do obliczania liczb pseudolosowych z wykorzystaniem generatora LCG.

Addytywny LCG:

$$X_{n+1} = a \cdot X_n + c \mod M$$

Multiplikatywny LCG:

$$X_{n+1} = a \cdot X_n \mod M$$

gdzie:

- X_n - n-ta liczba psudolosowa
- a - mnożnik
- c - parametr, dla generatora multiplikatywnego $c = 0$
- M - ilość generowanych liczb

Niech N będzie okresem generatora. Ważne informacje dotyczące długości generatora liniowego zawarte są w następujących twierdzeniach:

Twierdzenie 1

Maksymalny okres generatora liniowego wynosi $N = 2^{m-2}$, gdy $M = 2^m$ dla $m \geqslant 3$ i $a \equiv 3 \mod 8$ lub $a \equiv 5 \mod 8$.

Twierdzenie 2

Dla $M = p$, gdzie p jest liczbą pierwszą, generator liniowy posiada maksymalny okres równy p. Ten okres jest osiągany, gdy a jest pierwiastkiem pierwotnym liczby p.

Generatory kongruencyjne

- Liczby generowane przez addytywny LCG (Linear Congruential Generator) mogą przyjmować wartości z przedziału od 0 do $M - 1$.
- Multiplikatywny LCG generuje wartości z przedziału od 1 do $M - 1$.
- Po M wykonaniach, wartości zaczynają się powtarzać, co wynika z własności działania modulo.

Konfiguracja generatora liczb losowych

- Konieczne jest podanie wartości oznaczającej ziarno (ang. *seed*), które jest początkową wartością X_0.
- Aby generator działał poprawnie, wartości a, c i M nie mogą być przypadkowe.

Zasady doboru parametrów

- Wartość zmiennej M jest o jeden większa od największej wartości losowej, jaką algorytm będzie mógł wygenerować. Często wybiera się tutaj potęgi 10 albo 2,
- Parametr c musi być względnie pierwszy z M.
- Wartość wyrażenia $a - 1$ jest wielokrotnością wszystkich dzielników zmiennej M. Wartość a nie powinna być zbyt duża, ale też nie za mała - dobrym wyborem jest tutaj liczba o jeden rząd mniejsza (o jedną cyfrę krótsza) od M,

D.E.Knuth pokazał, iż niektóre wartości mogą prowadzić do wygenerowania bardzo krótkich cykli, co oczywiście jest wysoce niepożądane. Na przykład dla $b = 19$, $M = 381$ oraz $X_1 = 0$ otrzymamy sekwencję: $0, 1, 20, 0, 1, 20, \cdots$

- Główną wadą generatorów liniowych jest ich przewidywalność - punkty o współrzędnych (X_n, X_{n+1}) układają się na linii prostej.

Popularne generatory liczb losowych

Nazwa	**M**	**a**	**c**
Numerical Recipes	2^{32}	1664525	1013904223
Borland C/C++	2^{32}	22695477	1
GNU Compiler Collection	2^{32}	69069	5
ANSI C	2^{32}	1103515245	12345
Borland Delphi, Virtual Pascal	2^{32}	134775813	1
Microsoft Visual/Quick C/C++	2^{32}	214013	2531011
ANSIC	2^{31}	1103515245	12345
MINSTD	$2^{31} - 1$	16807	0

Table: Wybrane generatory liczb losowych

Przykład generatora addytywnego LCG

- Parametry generatora: $a = 4,\ c = 2,\ M = 9,\ X_0 = 0.$
- Wzór:

$$X_n = (4 \cdot X_{n-1} + 2) \mod 9$$

Rozwiązanie - kolejne wartości X_n

- $X_0 = (4 \cdot 0 + 2) \mod 9 = 2$
- $X_1 = (4 \cdot 2 + 2) \mod 9 = 1$
- $X_2 = (4 \cdot 1 + 2) \mod 9 = 6$
- $X_3 = (4 \cdot 6 + 2) \mod 9 = 8$
- $X_4 = (4 \cdot 8 + 2) \mod 9 = 7$
- $X_5 = (4 \cdot 7 + 2) \mod 9 = 3$
- $X_6 = (4 \cdot 3 + 2) \mod 9 = 5$
- $X_7 = (4 \cdot 5 + 2) \mod 9 = 4$
- $X_8 = (4 \cdot 4 + 2) \mod 9 = 0$
- $X_9 = (4 \cdot 0 + 2) \mod 9 = 2$
- $X_{10} = (4 \cdot 2 + 2) \mod 9 = 1$
- $X_{11} = (4 \cdot 1 + 2) \mod 9 = 6$
- $X_{12} = (4 \cdot 6 + 2) \mod 9 = 8$
- $X_{13} = (4 \cdot 8 + 2) \mod 9 = 7$
- $X_{14} = (4 \cdot 7 + 2) \mod 9 = 3$
- $X_{15} = (4 \cdot 3 + 2) \mod 9 = 5$
- $X_{16} = (4 \cdot 5 + 2) \mod 9 = 4$
- $X_{17} = (4 \cdot 4 + 2) \mod 9 = 0$

Generator Lehmera

Generator Lehmera, nazywany czasami generatorem liczb losowych Parka-Millera (od Stephena K. Parka i Keitha W. Millera), to rodzaj Liniowego Generatora Kongruencyjnego (LCG), który operuje w multiplikatywnej grupie modulo M. Jego działanie określa wzór:

$$X_{k+1} = a \cdot X_k \mod M,$$

gdzie:

- M – liczba pierwsza lub potęga liczby pierwszej,
- a – element mający wysoki rząd modulo M (np. pierwiastek pierwotny),
- X_0 – ziarno, liczba względnie pierwsza z M.

Generator Lehmera

W 1988 roku, Park i Miller zasugerowali wartości $M = 2^{31} - 1$ (liczba Mersenne'a M_{31}) i $a = 7^5 = 16807$ (pierwiastek pierwotny modulo M_{31}), obecnie znane jako MINSTD. Mimo krytyki, te parametry pozostały w użyciu do dzisiaj, włącznie z zastosowaniami w CarbonLib i minstd_rand0 z C++11. Park, Miller i Stockmeyer zasugerowali później użycie mnożnika $a = 48271$ zamiast 16807.

Generator Lehmera

W Sinclair ZX-81 i jego następcach użyto wartości $M = 2^{16} + 1$ (liczba Fermata F_4) i $a = 75$. CRAY użył generatora Lehmera z $M = 2^{48} - 1$ i $a = 44485709377909$. GNU Scientific Library zawiera kilka generatorów Lehmera, w tym MINSTD i RANF.

Generator Lehmera, choć jest specjalnym przypadkiem LCG (z $c = 0$), posiada unikalne ograniczenia i właściwości. W szczególności, ziarno X_0 musi być względnie pierwsze do M. Maksymalny okres generatora Lehmera, kiedy M jest liczbą pierwszą i a jest pierwiastkiem pierwotnym, równa się $M - 1$.

Uogólniony generator liniowy

- Uogólnienie generatora liniowego polega na uwzględnieniu kilku poprzednich wartości przy obliczaniu bieżącej wartości:

$$X_n = a_1 X_{n-1} + \ldots + a_k X_{n-k} + b \mod M$$

- Przy odpowiednim doborze stałych $a_1, \ldots, a_k, b < M$, generator ten może osiągnąć maksymalny okres równy M.
- Uogólniony generator liniowy, mimo większej złożoności, nadal nie nadaje się do zastosowań kryptograficznych.

Generator Fibonacciego jest jednym z odmian uogólnionego generatora liniowego liczb losowych, który opiera się na ciągu Fibonacciego:

$$X_M = X_{M-1} + X_{M-2} \mod M \quad (M \geqslant 2)$$

Ten rodzaj generatora charakteryzuje się lepszymi parametrami jakościowymi niż inne generatory liniowe, jednak jego główną wadą są duże korelacje między wyrazami ciągu, co oznacza, że ciągi te spełniają warunek rozkładu, ale nie warunek niezależności.

Lagged Fibonacci Generator

Uogólnienie generatora Fibonacciego

W celu zmniejszenia korelacji, można uogólnić generator Fibonacciego do postaci zwaną "lagged Fibonacci generator":

$$X_M = X_{M-p} + X_{M-q} \mod M \quad (M \geqslant p, p > q \geqslant 1)$$

gdzie p i q są opóźnieniami generatora. Generator taki można modyfikować zastępując działanie dodawania innymi operacjami, takimi jak odejmowanie, mnożenie, XOR:

$$X_M = X_{M-q} \diamond X_{M-p} \mod M \quad (M \geqslant p, p > q \geqslant 1),$$

gdzie $\diamond$ reprezentuje wybrany operator.

Przykład zastosowania generatora

$$m = 17, \quad p = 3, \quad q = 1, \quad X_0 = 7, \quad X_1 = 16, \quad X_2 = 5$$

$$X_M = X_{M-q} + X_{M-p} \mod m$$

$$\begin{aligned} X_3 &= X_0 + X_2 \mod 17 = 7 + 5 \mod 17 = 12, \\ X_4 &= X_1 + X_3 \mod 17 = 16 + 12 \mod 17 = 11, \\ X_5 &= X_2 + X_4 \mod 17 = 5 + 11 \mod 17 = 16. \end{aligned}$$

Kolejne wartości ciągu: 7, 16, 5, 12, 11, 16, 11, 5, 4, 15, 3, 7,...

Zaawansowane uogólnienia

Generator Fibonacciego można rozbudować o więcej poprzednich punktów:

$$X_M = X_{M-p_1} \diamond X_{M-p_2} \diamond \ldots \diamond X_{M-p_k} \mod M \quad (M \geqslant p_k > \ldots > p_1 \geqslant 1)$$

Przykłady znanych generatorów tego typu to generator Marsagli (Marsa-LFIB4) i generator Ziffa (Ziff98), które wykorzystują różne operacje i mają różne parametry p_i.

Kwadratowy generator kongruencyjny

Aby uniknąć liniowej zależności kolejnych wartości ciągu, zastosowano zależność kwadratową:

$$X_{n+1} = aX_n^2 + bX_n + c \mod M$$

Dla odpowiednich wartości $a, b, c, X_0 < M$, maksymalny okres tego generatora może być równy M.

Generator wykorzystujący wielomiany permutacyjne

Generator wykorzystuje wielomiany permutacyjne postaci:

$$g(X) = \sum_{k=0}^{n} a_k X^k \quad a_1, \ldots, a_n \in \{0, 1, \ldots, M-1\}$$

Maksymalny okres generatora jest równy M. W przeciwieństwie do generatora liniowego, nie jest przewidywalny.

Generator inwersyjny

Generator inwersyjny, używany do generowania liczb losowych o równomiernym rozkładzie, wykorzystuje wzór:

$$X_{n+1} = \begin{cases} (aX_n^{-1} + b) \mod p & \text{dla } X_n \neq 0; \\ b & \text{dla } X_n = 0; \end{cases}$$

gdzie p jest liczbą pierwszą. Maksymalny okres generatora, przy odpowiednich wartościach a i b, może wynosić $p - 1$.

Metody testowania generatorów

Aby ocenić, czy przyjąć, czy odrzucić generator lub wygenerowany ciąg bitów stosujemy w praktyce zestawy odpowiednich testów statystycznych. Testy te powinny potwierdzać:

- równomierny rozkład ciągu bitów,
- losowość rozkładu,
- niezależność kolejnych bitów.

W literaturze dostępnych jest wiele testów do oceny generatorów liczb losowych. Można wyróżnić testy ogólne, dotyczące rozkładów liczb całkowitych, oraz specyficzne, skoncentrowane na ciągach binarnych. Zestawy testów powinny być kompletne, ale także wystarczająco ograniczone, aby były wykonalne w praktyce podczas użytkowania generatora.

Testy według normy FIPS-140-3

Praktyczne zastosowanie testów

Weryfikacja generatorów często korzysta ze standardów amerykańskich, np. normy FIPS-140-3, która określa cztery podstawowe testy dla ciągów długości 20 000 bitów:

1. Test monobitowy - sprawdza, czy liczba jedyneczek jest w określonych granicach.
2. Test pokerowy - bada równomierność rozkładu segmentów czterobitowych.
3. Test serii - analizuje ciągłość serii takich samych bitów.
4. Test długich serii - sprawdza, czy występują zbyt długie serie jednego bitu.

Norma FIPS-140-3 stawia wysokie wymagania dla akceptacji ciągów bitów, wymagając, aby poziom istotności testów (prawdopodobieństwo odrzucenia ciągu bitów mogącego pochodzić z prawidowego źródła) wynosił $0,0001$.

Test monobitowy

Test monobitowy jest podstawowym testem statystycznym używanym do sprawdzania generatorów liczb losowych. Jego głównym celem jest ocena, czy liczba jedynkowych bitów ('1') w wygenerowanym ciągu bitów leży w granicach, które są statystycznie prawdopodobne dla równomiernie losowego ciągu. Test wykorzystuje statystykę chi-kwadrat do oceny, czy zaobserwowane odchylenie liczby jedynek od oczekiwanej średniej jest statystycznie istotne.

Test monobitowy

Procedura testu monobitowego składa się z kilku kroków:

1. Obliczenie liczby jedynek w ciągu 20000 bitów:

$$X = \text{liczba jedynek w ciągu 20000 bitów}$$

2. Ocena, czy wygenerowany ciąg można uznać za akceptowalny, na podstawie kryterium:

$$9725 < X < 10275$$

Jeżeli liczba jedynkowych bitów spełnia ten warunek, ciąg jest uznawany za zgodny z oczekiwaniami dla równomiernego rozkładu.

Test pokerowy

Test pokerowy to metoda statystyczna używana do sprawdzania generatorów liczb losowych, zwłaszcza w kontekście ich zdolności do produkcji ciągów bitów, które nie wykazują widocznych wzorców i są równomiernie rozłożone. Test skupia się na analizie segmentów czterobitowych w ciągu bitów.
Jest to krytyczne dla zastosowań, które wymagają wysokiej jakości losowości, takich jak algorytmy kryptograficzne.

Test pokerowy

Procedura testu pokerowego obejmuje następujące kroki:

1. Dzielenie ciągu 20000 bitów na 5000 segmentów czterobitowych.
2. Liczenie częstości pojawienia się każdego z 16 możliwych czterobitowych segmentów (od 0000 do 1111).
3. Obliczanie statystyki chi-kwadrat z wyników częstości:

$$X = \frac{16}{5000}\left(\sum_{i=0}^{15}[f(i)]^2\right) - 5000$$

   gdzie $f(i)$ to liczba wystąpień i-tego czterobitowego segmentu.
4. Ocena zgodności rozkładu segmentów z oczekiwanym rozkładem równomiernym, gdzie zakres akceptowalnych wartości statystyki X wynosi:

$$2.16 < X < 46.17$$

Test serii

Test serii ocenia, czy liczba serii różnej długości w ciągu bitów jest zgodna z oczekiwaniami dla ciągu bitów o równomiernym rozkładzie losowości. Serią nazywamy ciąg kolejnych bitów o tej samej wartości.
Test serii pozwala zidentyfikować ciągi bitów, które mogą nie wykazywać oczekiwanej losowości ze względu na nieodpowiednią liczbę lub rozkład długości serii. Jest to kluczowe dla zastosowań kryptograficznych i innych, gdzie jakość losowości bezpośrednio wpływa na bezpieczeństwo i efektywność systemów.

Test serii

Test serii skupia się na analizie długości serii bitów w wygenerowanym ciągu:

1. Podział ciągu 20000 bitów na serie jednorodnych bitów (1 lub 0).
2. Liczenie liczby serii o różnej długości. Specyficzne kategorie długości to 1, 2, 3, 4, 5 oraz 6 i więcej.
3. Porównanie wyników z teoretycznymi wartościami oczekiwanymi dla idealnie losowego ciągu bitów. Akceptowalne zakresy dla liczby serii różnych długości są następujące:
   - Długość serii 1: 2343 do 2657
   - Długość serii 2: 1135 do 1365
   - Długość serii 3: 542 do 708
   - Długość serii 4: 251 do 373
   - Długość serii 5: 111 do 201
   - Długość serii 6 i więcej: 111 do 201

Różne rodzaje testów statystycznych

W literaturze można znaleźć różne grupy testów, które badają rozmaite cechy badanej populacji oraz warunki eksperymentu losowego, w tym:

- Testy losowości, które określają, czy ciąg wyników może być traktowany jako ciąg zmiennych losowych.
- Testy zgodności, które określają, czy obserwacje mają ten sam rozkład prawdopodobieństwa.
- Testy normalności, sprawdzające, czy dane pochodzą z rozkładu normalnego.
- Testy dotyczące parametrów rozkładu, potwierdzające wiarygodność estymowanych parametrów.
- Testy niezależności, które określają, czy zmienne są niezależne.

Podsumowanie - kryteria dobrego generatora

- **Okres**: Idealny generator miałby okres równy lub zbliżony do maksymalnej możliwej liczby unikatowych stanów, co oznacza, że ciąg wartości nie powtarza się przez bardzo długi czas.
- **Równomierność**: Równomierność odnosi się do równego prawdopodobieństwa każdej z możliwych wartości w zakresie generatora. Równomierność generatora można ocenić za pomocą testów statystycznych, takich jak test chi-kwadrat, które pozwalają określić, czy rozkład generowanych liczb jest bliski idealnemu rozkładowi równomiernemu.
- **Nieprzewidywalność**: Nieprzewidywalność oznacza, że nie jest możliwe efektywne przewidzenie kolejnych wartości w ciągu na podstawie jakiejkolwiek skończonej liczby wcześniejszych wartości. Nieprzewidywalność można oceniać przez analizę korelacji i innych wzorców w wygenerowanych ciągach. Testy takie jak test następnego bitu są przykładami metod oceny tego kryterium.