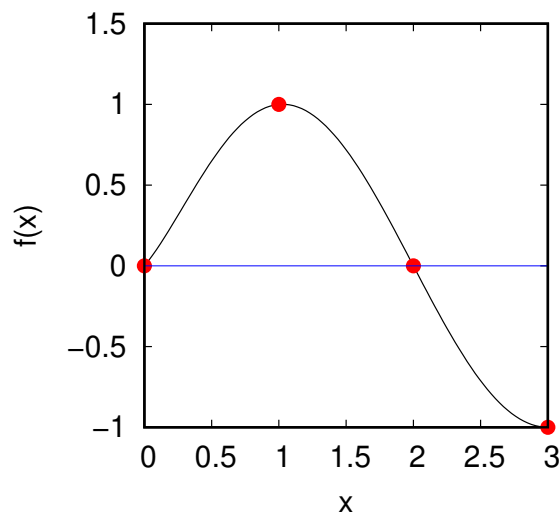


UARL: modelowanie wielomianu - rozkład LU

Tomasz Chwiej

4 grudnia 2024

1 Wstęp: modelowanie wielomianu przy użyciu układu równań



Rysunek 1: Wielomian z zaznaczonymi punktami, w których zdefiniowane są własności krzywej (wartość i/lub pochodna).

Jednym z potencjalnych zastosowań układów równań liniowych jest modelowanie kształtu wielomianów tak, aby wybranych punktach spełniały one określone warunki. Zdefiniujmy zestaw warunków dla krzywej pokazanej na rysunku 1:

$$f(x_0) = 0 \quad (1)$$

$$\left. \frac{df(x)}{dx} \right|_{x=x_0} = 1 \quad (2)$$

$$f(x_1) = 1 \quad (3)$$

$$f(x_2) = 0 \quad (4)$$

$$f(x_3) = -1 \quad (5)$$

$$\left. \frac{df(x)}{dx} \right|_{x=x_3} = 0 \quad (6)$$

Jeśli chcemy aby krzywą tą był wielomian to musi mieć on tyle współczynników a_i ile jest warunków (dokładnie 6), będzie to wielomian stopnia 5:

$$f(x) = a_0 + a_1x + a_2x^2 + a_3x^3 + a_4x^4 + a_5x^5 \quad (7)$$

Teraz zapiszmy kolejno te warunki korzystając z jawnej postaci wielomianu:

$$1 \cdot a_0 + x_0 \cdot a_1 + x_0^2 \cdot a_2 + x_0^3 \cdot a_3 + x_0^4 \cdot a_4 + x_0^5 \cdot a_5 = 0 \quad (8)$$

$$0 \cdot a_0 + 1 \cdot a_1 + 2x_0^1 \cdot a_2 + 3x_0^2 \cdot a_3 + 4x_0^3 \cdot a_4 + 5x_0^4 \cdot a_5 = 1 \quad (9)$$

$$1 \cdot a_0 + x_1 \cdot a_1 + x_1^2 \cdot a_2 + x_1^3 \cdot a_3 + x_1^4 \cdot a_4 + x_1^5 \cdot a_5 = 1 \quad (10)$$

$$1 \cdot a_0 + x_2 \cdot a_1 + x_2^2 \cdot a_2 + x_2^3 \cdot a_3 + x_2^4 \cdot a_4 + x_2^5 \cdot a_5 = 0 \quad (11)$$

$$1 \cdot a_0 + x_3 \cdot a_1 + x_3^2 \cdot a_2 + x_3^3 \cdot a_3 + x_3^4 \cdot a_4 + x_3^5 \cdot a_5 = -1 \quad (12)$$

$$0 \cdot a_0 + 1 \cdot a_1 + 2x_3^1 \cdot a_2 + 3x_3^2 \cdot a_3 + 4x_3^3 \cdot a_4 + 5x_3^4 \cdot a_5 = 0 \quad (13)$$

Układ równań zapiszmy w postaci macierzowej

$$A\vec{a} = \vec{b} \quad (14)$$

$$\begin{bmatrix} 1 & x_0^1 & x_0^2 & x_0^3 & x_0^4 & x_0^5 \\ 0 & x_1 & 2x_1^1 & 3x_1^2 & 4x_1^3 & 5x_1^4 \\ 1 & x_1^1 & x_1^2 & x_1^3 & x_1^4 & x_1^5 \\ 1 & x_2^1 & x_2^2 & x_2^3 & x_2^4 & x_2^5 \\ 1 & x_3^1 & x_3^2 & x_3^3 & x_3^4 & x_3^5 \\ 0 & x_3 & 2x_3^1 & 3x_3^2 & 4x_3^3 & 5x_3^4 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 1 \\ 0 \\ -1 \\ 0 \end{bmatrix} \quad (15)$$

Układ ten rozwiążemy numerycznie przy użyciu procedur z biblioteki numerycznej GSL, a znalezione współczynniki wykorzystamy do sporządzenia wykresu szukanego wielomianu.

Uwagi:

- ponieważ liczba kolumn i wierszy jest identyczna, rozwiązanie które uzyskamy będzie jednoznaczne
- gdybyśmy zażądali dla naszego wielomianu stopnia 5 spełnienia dodatkowego warunku, to liczba wierszy byłaby większa od liczby kolumn, taki układ równań byłby **nadokreślony** i zazwyczaj nie posiada on ścisłego rozwiązania (tj. wielomianu spełniającego wszystkie warunki) ale można poszukiwać rozwiązania przybliżonego w sensie **średniokwadratowym**
- w przypadku gdybyśmy dla wielomianu stopnia 5 zdefiniowali liczbę warunków $k < 6$ wówczas każda kolumna macierzy A wypełniona zerami redukuje stopień wielomianu o 1 - nie ma sensu poszukiwać wielomianu wyższego stopnia, bo taki nie istnieje.

2 Zadania do wykonania

1. Przyjmujemy położenia węzłów: $x_0 = 0$, $x_1 = 1$, $x_2 = 2$, $x_3 = 3$.
2. Utworzyć macierz A oraz wektor wyrazów wolnych $\vec{b}$ i wypełnić ich elementy jak we wzorze 15.
3. Znaleźć rozkład LU macierzy A przy użyciu procedury `gsl_linalg_LU_decomp(...)`.
4. Obliczyć wyznacznik macierzy A jako iloczyn elementów diagonalnych U korzystając z relacji

$$\det(A) = \det(LU) = \det(L)\det(U) = 1 \cdot \det(U) = \prod_{i=0}^{n-1} u_{ii} \quad (16)$$

Czyli po znalezieniu rozkładu macierzy A, która jest nadpisana przez macierz dolną trójkątną L (poniżej diagonal) oraz górną trójkątną (od diagonal w górę) należy policzyć iloczyn jej elementów diagonalnych.

5. Rozwiązać układ równań przy użyciu procedury `gsl_linalg_LU_solve(...)`, współczynniki wektora $\vec{a}$ zapisać do pliku.

6. Wykorzystać znalezione współczynniki a_i w celu sporządzenia wykresu wielomianu $f(x)$ (wzór 7).
7. W raporcie przeanalizować rozwiązywalność układu równań w kontekście wyznaczonej wartości wyznacznika macierzy A . Sprawdzić czy znaleziony wielomian spełnia warunki (wzory 1-6), obliczając wartość i pochodną w wybranych punktach.

3 Obsługa biblioteki GSL

- Aby wykorzystać GSL-a musimy zlokalizować bibliotekę, np. przy użyciu polecenia *locate* (LINUX)

```
locate gsl |grep gsl_linalg.h
locate gsl |grep libgsl.a
```

Pierwsza instrukcja podaje ścieżkę do katalogu z plikami nagłówkowymi (np. /usr/include), a druga do katalogu zawierającego skompilowaną bibliotekę (np. /usr/lib/x86_64-linux-gnu/). Kompilacja kodu C (C++) na Taurusie z uwzględnieniem ścieżek oraz bibliotek

```
gcc -L/usr/lib/x86_64-linux-gnu -I/usr/include kod.c -lm -lgsl -lgslcblas
g++ -L/usr/lib/x86_64-linux-gnu -I/usr/include kod.cpp -lm -lgsl -lgslcblas
```

Uwaga: podczas kompilacji ważna jest kolejność w jakiej podajemy używane biblioteki - te są odczytywane od prawej do lewej, zmiana kolejności może spowodować błąd

- W programie należy dołączyć plik nagłówkowy zawierający definicje procedur GSL-a z algorytmu liniowej

```
#include<gsl/gsl_linalg.h>
```

- Pakiet wymaga użycia macierzy i wektorów własnego typu, które tworzymy następująco

```
gsl_vector *x=gsl_vector_alloc(int n)
gsl_permutation *x=gsl_permutation_alloc(int n)
gsl_matrix *A=gsl_matrix_alloc(int n, int n)
```

gdzie: n jest liczbą elementów w wektorze oraz liczbą kolumn/wierszy w macierzy. Zapis wartości liczbowych w komórkach oraz ich odczyt wygląda następująco

```
gsl_vector_set(gsl_vector *a,int i,double value)
gsl_matrix_set(gsl_matrix *A,int i,int j,double value)
double b=gsl_vector_get(gsl_vector *a,int i)
double b=gsl_matrix_get(gsl_matrix *A,int i,int j)
```

- Rozkład LU wyznaczymy przy użyciu procedury

```
gsl_linalg_LU_decomp(gsl_matrix *A, gsl_permutation *p, int *signum)
```

po jej wykonaniu dolna część macierzy A zostaje nadpisana przez macierze L i U . Po znalezieniu rozkładu (macierz A zawiera rozkład LU) możemy rozwiązać układ równań

```
gsl_linalg_LU_solve(gsl_matrix *A, gsl_permutation *p, gsl_vector *b, gsl_vector *a);
```

Po wywołaniu procedury wektor $\vec{a}$ zawiera współczynniki szukanego wielomianu.