Aproksymacja wielomianowa

Michał Saturczak
Informatyka Stosowana, Akademia Górniczo-Hutnicza
Metody Numeryczne

9 maja 2024

1 Wstęp teoretyczny

Aproksymacja wielomianowa to kluczowy temat w dziedzinie analizy numerycznej i ma szerokie zastosowanie w wielu dziedzinach nauki i inżynierii. Jest to proces reprezentowania skomplikowanej funkcji lub zestawu danych za pomocą wielomianu, który jest prostszy do analizy i manipulacji.

Podstawowym założeniem aproksymacji wielomianowej jest to, że dowolną funkcję można przybliżyć wielomianem w odpowiednim przedziale. Jest to wynik fundamentalnej twierdzenia algebry, które mówi, że dowolną funkcję ciągłą można przybliżyć wielomianem dowolnie blisko na dowolnym zamkniętym przedziale.

Aproksymacje wielomianowe są wykorzystywane w wielu różnych kontekstach. W analizie numerycznej są one używane do przybliżania funkcji, które są trudne do obliczenia bezpośrednio. W statystyce, są one często używane do modelowania zależności między zmiennymi. W inżynierii, są one używane do modelowania odpowiedzi systemu na różne wejścia.

Mimo że aproksymacja wielomianowa jest potężnym narzędziem, ma swoje ograniczenia. Na przykład, nie wszystkie funkcje można skutecznie przybliżyć za pomocą wielomianów, szczególnie jeśli funkcja jest skomplikowana lub ma duże oscylacje. Ponadto, wielomiany o wysokich stopniach mogą prowadzić do problemów numerycznych, takich jak niestabilność i duże błędy zaokrąglenia.

Pomimo tych wyzwań, aproksymacja wielomianowa pozostaje kluczowym narzędziem w dziedzinie analizy numerycznej i jest niezbędna dla wielu zastosowań naukowych i inżynieryjnych. Jest to temat, który jest ciągle rozwijany i udoskonalany, a jego zrozumienie jest kluczowe dla każdego, kto pracuje w dziedzinie nauki i inżynierii.

2 Problem

2.1 Rozwiązanie problemu

- Napisano program w języku C++ z pomocą biblioteki GSL do rozwiązania problemów związanych z aproksymacją wielomianową.
- Zapisano do trzech skoroszytów dane odpowiednio dla każdego $n = 11, 51, 101$.
- Zrobiono wykresy przedstawiające wartości właściwe funkcji oraz wartości aproksymowane.

Funkcja, na której wykonano aproksymację:

$$f(x) = -0.25x^3 - 0.5x^2 + 5x + 5 + \delta$$

gdzie

$$\delta = 5(\mathrm{rand}() - 0.5)$$

a rand to funkcja, która losuje wartości z przedziału [0,1].

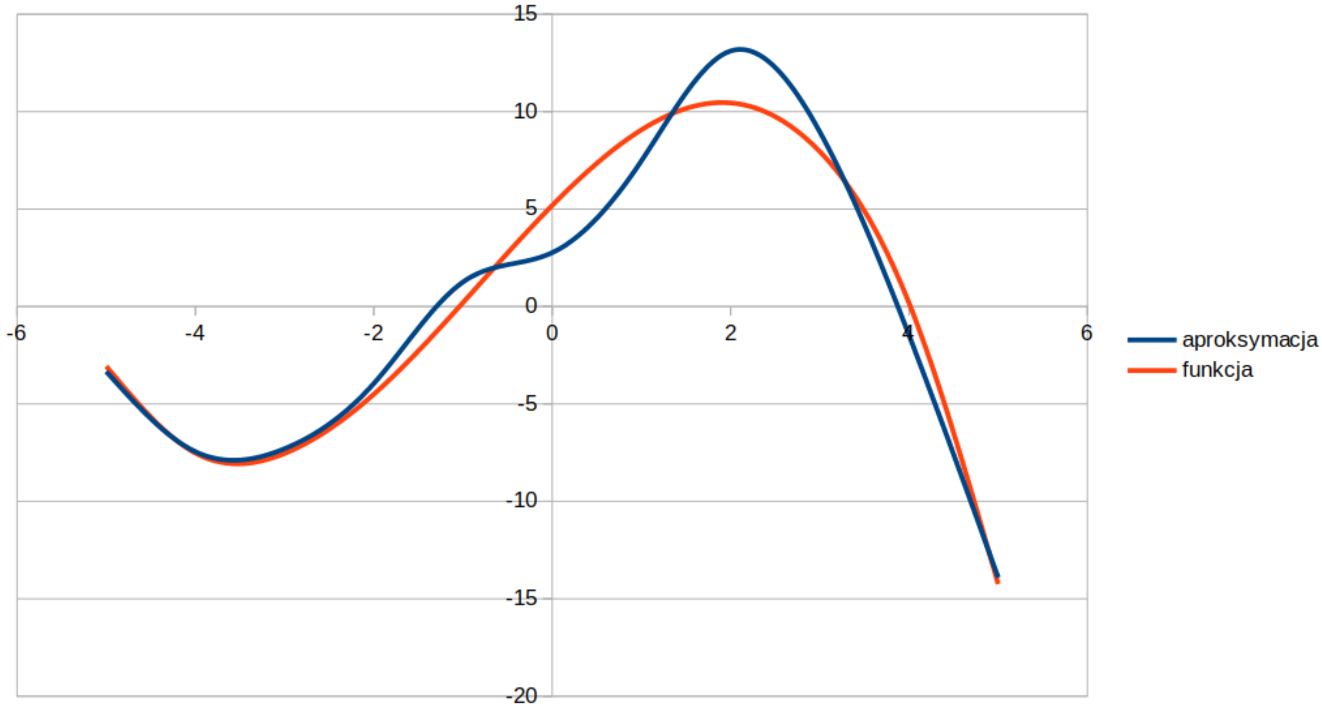
2.2 Funkcje oraz kod w języku C++ użyty do rozwiązania problemu

```
double fun(double x)
{
  std::default_random_engine generator(std::chrono::system_clock::now().time_since_epoch().count());
  std::uniform_real_distribution<double> distribution(-0.5, 0.5);
  double delta = 5 * distribution(generator);
  return -0.25 * std::pow(x, 3) - 0.5 * std::pow(x, 2) + 5 * x + 5 + delta;
}

int main()
{
```

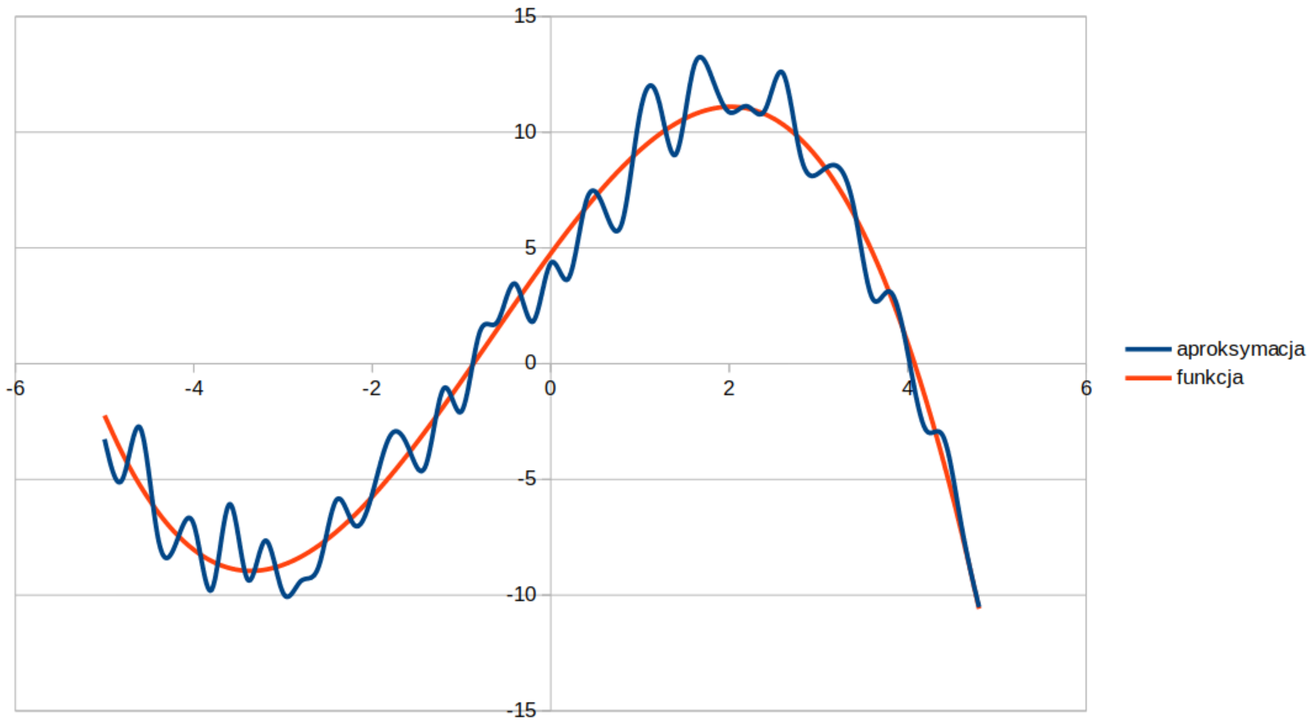
```
  for (int n : {11, 51, 101})
  {
    double x0 = -5, x1 = 5;
    gsl_matrix *X = gsl_matrix_calloc(n, 4);
    gsl_vector *Y = gsl_vector_calloc(n);
    int licznik = 0;
    for (double i = x0; i <= x1; i += (x1 - x0) / (n - 1))
    {
      for (int j = 0; j < 4; j++)
      {
        gsl_matrix_set(X, licznik, j, std::pow(i, j));
      }
      gsl_vector_set(Y, licznik, fun(i));
      licznik++;
    }

    gsl_matrix *XT = gsl_matrix_calloc(4, n);
    gsl_matrix_transpose_memcpy(XT, X);

    gsl_matrix *D = gsl_matrix_calloc(4, 4);
    gsl_blas_dgemm(CblasNoTrans, CblasNoTrans, 1.0, XT, X, 0.0, D);

    gsl_vector *r = gsl_vector_calloc(4);
    gsl_blas_dgemv(CblasNoTrans, 1.0, XT, Y, 0.0, r);

    gsl_vector *b = gsl_vector_calloc(4);
    gsl_permutation *p = gsl_permutation_calloc(4);
    int signum;
    gsl_linalg_LU_decomp(D, p, &signum);
    gsl_linalg_LU_solve(D, p, r, b);

    gsl_vector *Y_approx = gsl_vector_calloc(n);
    gsl_blas_dgemv(CblasNoTrans, 1.0, X, b, 0.0, Y_approx);

    std::ofstream file;
    file.open("data" + std::to_string(n) + ".csv");
    for (int i = 0; i < n; i++)
    {
      file << gsl_matrix_get(X, i, 1) << "\t" << gsl_vector_get(Y, i) << "\t" << gsl_vector_get(Y_ap
    }
    file.close();

    gsl_matrix_free(X);
    gsl_matrix_free(XT);
    gsl_matrix_free(D);
    gsl_vector_free(Y);
    gsl_vector_free(r);
    gsl_vector_free(b);
    gsl_vector_free(Y_approx);
    gsl_permutation_free(p);
  }
  return 0;
}
```

3 Wyniki

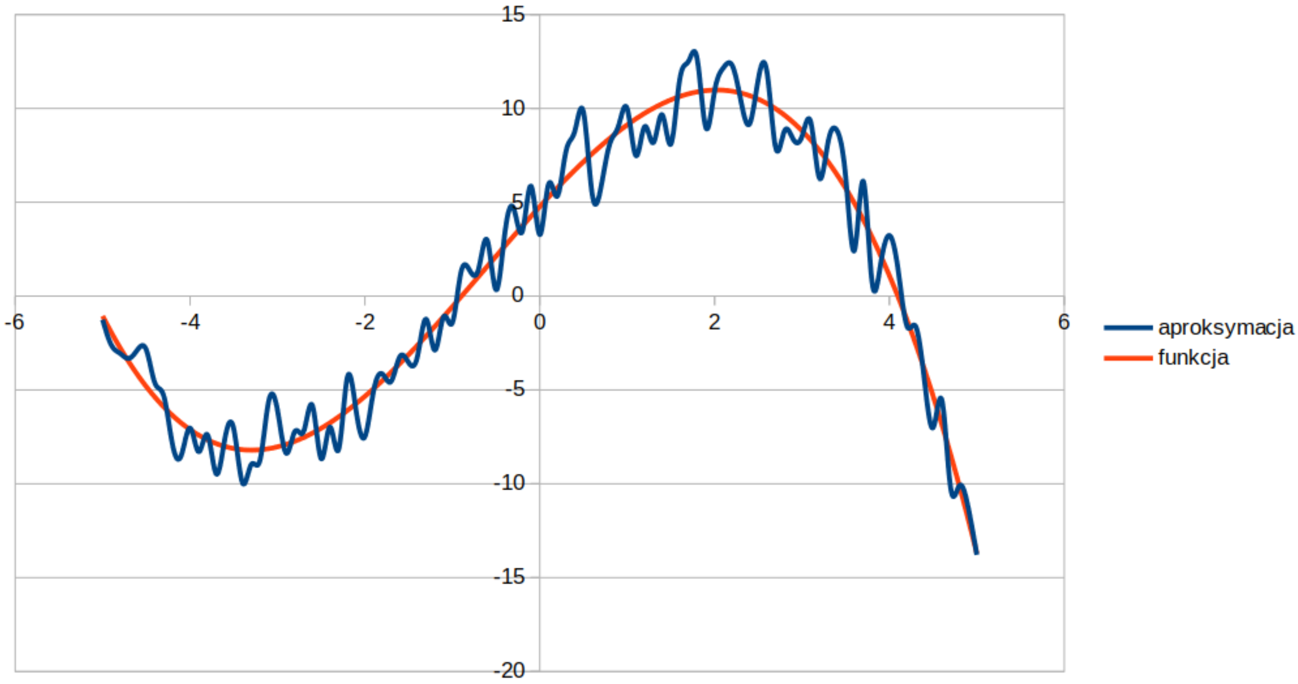
3.1 Wykres funkcji oraz jej aproksymacji dla $n = 11$



3.2 Wykres funkcji oraz jej aproksymacji dla $n = 51$



3.3 Wykres funkcji oraz jej aproksymacji dla $n = 101$



4 Wnioski

Z analizy wynika, że skuteczność metody aproksymacji wielomianowej wzrasta wraz ze wzrostem liczby punktów. Mimo że przy ograniczonej liczbie punktów wynik końcowy może nie spełniać wszystkich oczekiwań, nie jest on tak niezadowalający, jak mogłoby się to wydawać na pierwszy rzut oka.

Można zaobserwować, że wartości wielomianów w niektórych punktach znacznie odbiegają od wartości aproksymowanych. Taka sytuacja może być spowodowana faktem, że zmienna δ przyjmuje różne wartości, co wprowadza dodatkową zmienność do procesu aproksymacji.

Wartości δ mogą wpływać na kształt wielomianu, zwłaszcza jeśli są losowane z szerokiego zakresu. Ta dodatkowa zmienność może prowadzić do większych różnic między wartościami wielomianu a aproksymowanymi wartościami, szczególnie w punktach, w których wielomian jest szczególnie wrażliwy na zmiany wartości δ.

To pokazuje, jak ważne jest uwzględnienie wszystkich zmiennych wpływających na proces aproksymacji wielomianowej, a także jak ważna jest odpowiednia interpretacja wyników. Mimo pewnych wyzwań, aproksymacja wielomianowa nadal jest ważnym narzędziem w analizie numerycznej, które pozwala na efektywne modelowanie i analizę skomplikowanych zestawów danych.