Interpolacja funkcji metodą Lagrange'a

Michał Saturczak
Informatyka Stosowana, Akademia Górniczo-Hutnicza
Metody Numeryczne

12 kwietnia 2024

1 Wstęp teoretyczny

Interpolacja jest fundamentalnym narzędziem w analizie numerycznej, umożliwiającym przybliżanie funkcji na podstawie skończonego zestawu jej punktów. Jednym z najprostszych i najbardziej powszechnych sposobów interpolacji jest metoda Lagrange'a.

Metoda interpolacji Lagrange'a polega na znalezieniu wielomianu, który przechodzi przez daną serię punktów (x_i, y_i), gdzie $i = 0, 1, \ldots, n$. Wielomian interpolacyjny Lagrange'a dla danego zestawu punktów jest dany wzorem:

$$L(x) = \sum_{i=0}^{n} y_i \cdot l_i(x)$$

gdzie $l_i(x)$ to bazowe wielomiany Lagrange'a, zdefiniowane jako:

$$l_i(x) = \prod_{j=0, j\neq i}^{n} \frac{x - x_j}{x_i - x_j}$$

Każdy bazowy wielomian $l_i(x)$ jest taki, że $l_i(x) = 1$ dla $x = x_i$ i $l_i(x) = 0$ dla $x = x_j$, gdzie $j \neq i$. To gwarantuje, że $L(x_i) = y_i$ dla każdego i, więc wielomian $L(x)$ przechodzi przez wszystkie dane punkty.

Choć metoda Lagrange'a jest prosta i elegancka, ma pewne wady. W szczególności, dodanie nowego punktu do zestawu wymaga przeliczenia całego wielomianu, co może być kosztowne obliczeniowo dla dużych zestawów punktów. Ponadto, metoda ta może cierpieć na efekt Rungego, gdzie błędy interpolacji rosną na końcach przedziału interpolacji.

Wielomiany Czebyszewa są szczególnym rodzajem wielomianów ortogonalnych, które są szeroko stosowane w wielu dziedzinach matematyki i nauk ścisłych. Jednym z zastosowań wielomianów Czebyszewa jest optymalizacja funkcji.

W kontekście optymalizacji, wielomiany Czebyszewa mogą być wykorzystane do znalezienia najlepszego przybliżenia funkcji w określonym przedziale. Wielomiany Czebyszewa są wyjątkowe, ponieważ minimalizują maksymalny błąd pomiędzy funkcją a jej wielomianowym przybliżeniem w danym przedziale, co jest znane jako zasada równomierności.

Optymalne położenia węzłów obliczamy jako kolejne zera wielomianu Czebyszewa. W tym przypadku można to sprowadzić do rozwiązania równania:

$$x_m = \frac{1}{2}\left[(x_{\max} - x_{\min})\cos\left(\frac{\pi(2m+1)}{2n'+2}\right) + (x_{\max} + x_{\min})\right], \quad m = 0, 1, 2, \ldots, n'$$

gdzie $x_{\max}$ i $x_{\min}$ to granice przedziału optymalizacji, a $n' + 1 = n$.

Podczas optymalizacji, funkcja jest aproksymowana przez wielomian Czebyszewa o odpowiednim stopniu. Współczynniki wielomianu są następnie dostosowywane tak, aby zminimalizować błąd pomiędzy funkcją a jej wielomianowym przybliżeniem.

Wielomiany Czebyszewa są szczególnie przydatne w optymalizacji, ponieważ umożliwiają szybkie obliczanie wartości funkcji w punktach Czebyszewa, które są punktami, w których błąd aproksymacji wielomianowej jest minimalny.

2 Problem

2.1 Rozwiązanie problemu

- Napisano program w języku C++ do rozwiązania problemów związanych z Interpolacją
- Zapisano do trzech skoroszytów dane odpowiednio dla dwóch funkcji z krokiem co 0.1 dla wartości $x \in [-5, 5]$, oraz dla wartości $n = 5, 10, 15, 20$.
- Zrobiono wykresy aktualne funkcji oraz ich interpolacje.
- Powtórzono kroki poprzednie, tylko że z wykorzystaniem optymalizacji za pomocą wielomianów Czebyszewa.

2.2 Funkcje matematyczne dla których obliczono interpolacje Lagrange'a

- $f(x) = e^{-x^2}$
- $f(x) = -1$ dla $x < 0, f(x) =$ dla $x \geq 0$
- $f(x) = \cos 2x$

2.3 Funkcje w języku C++ użyte do generowania węzłów interpolacji

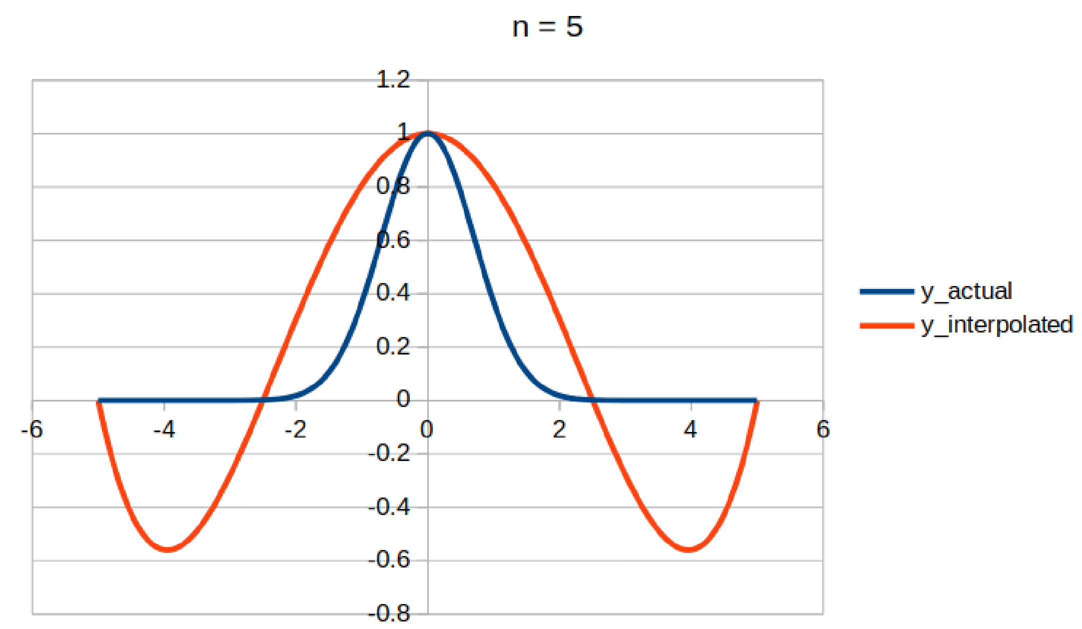
```
// Generowanie węzłów interpolacji z wykorzystaniem wielomianu Czebyszewa
std::vector<double> generateChebyshevNodes(double xmin, double xmax, int n)
{
  std::vector<double> nodes(n);
  for (int i = 0; i < n; i++)
  {
    nodes[i] = 0.5 * ((xmax - xmin) * cos(M_PI * (2.0 * i + 1) / (2.0 * n + 2)) + (xmax + xmin));
  }
  return nodes;
}

// Generowanie węzłów interpolacji
std::vector<double> generateNodes(double xmin, double xmax, int n)
{
  std::vector<double> nodes(n);
  double step = (xmax - xmin) / (n - 1);
  for (int i = 0; i < n; i++)
    nodes[i] = xmin + i * step;
  return nodes;
```

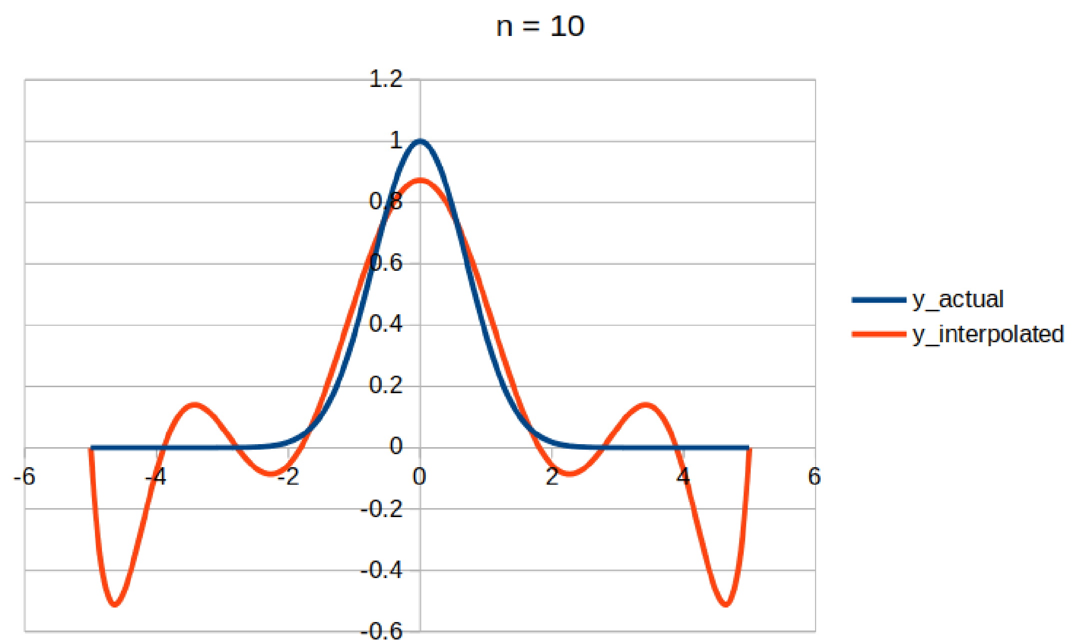
3 Wyniki

3.1 Wykresy interpolacji Lagrange'a dla funkcji $f(x) = e^{-x^2}$

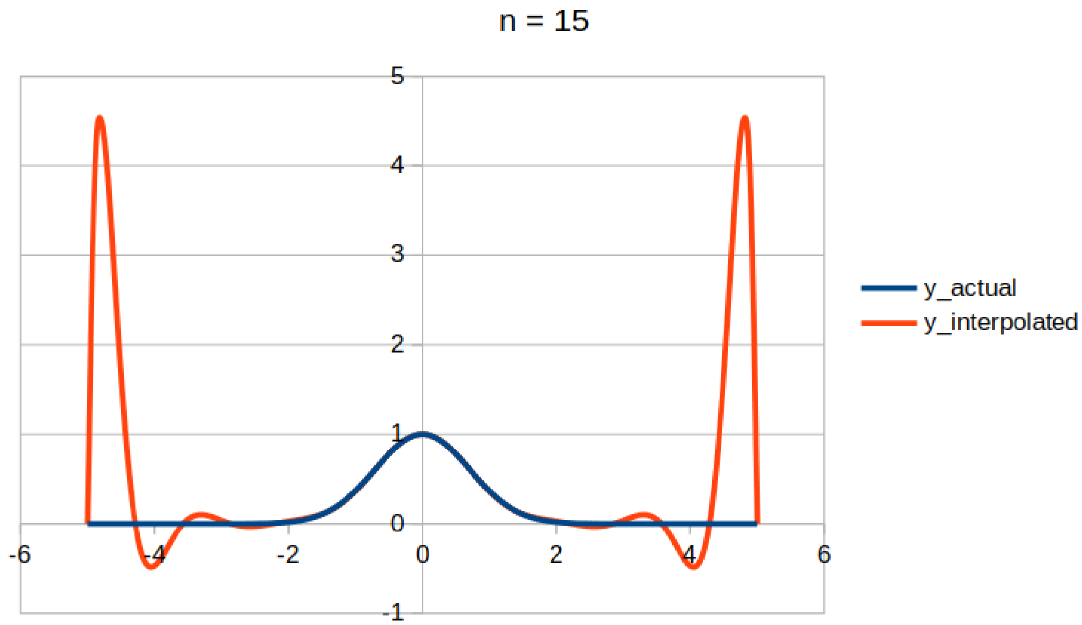
3.1.1 $n = 5$



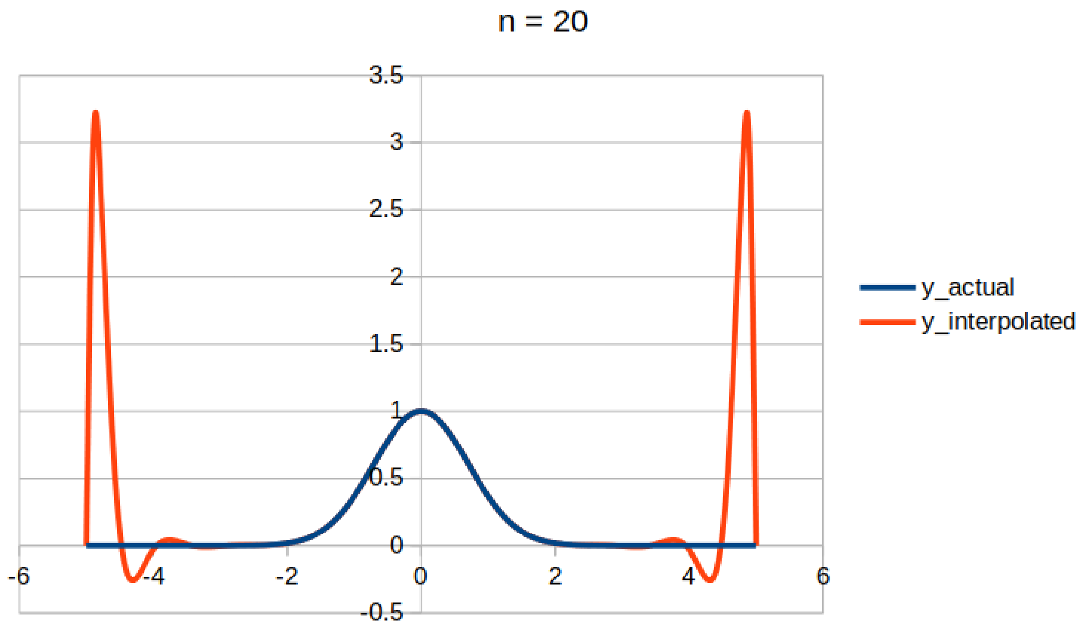
3.1.2 $n = 10$



3.1.3 $n = 15$

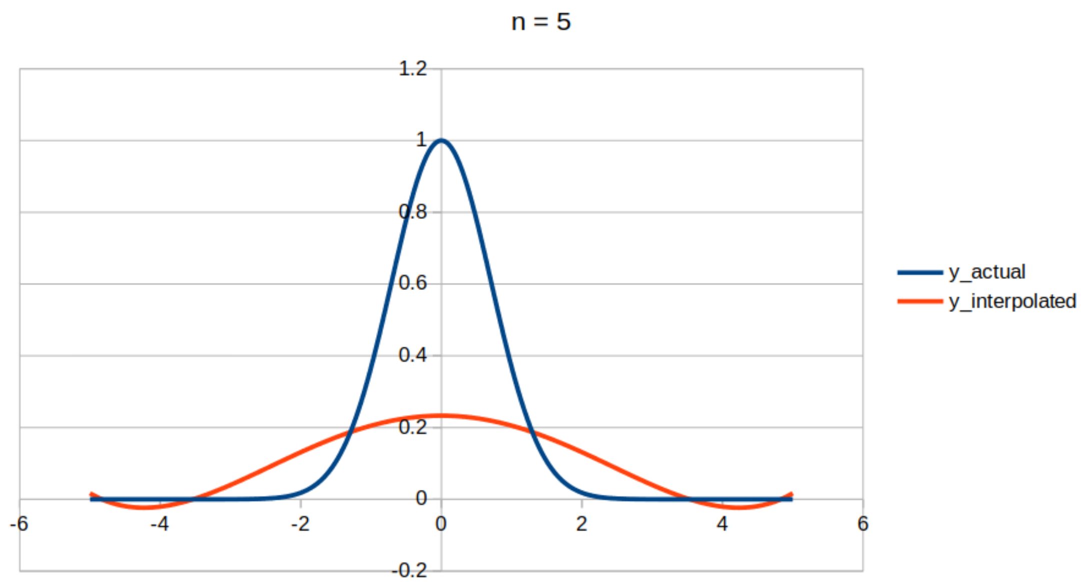


3.1.4 $n = 20$

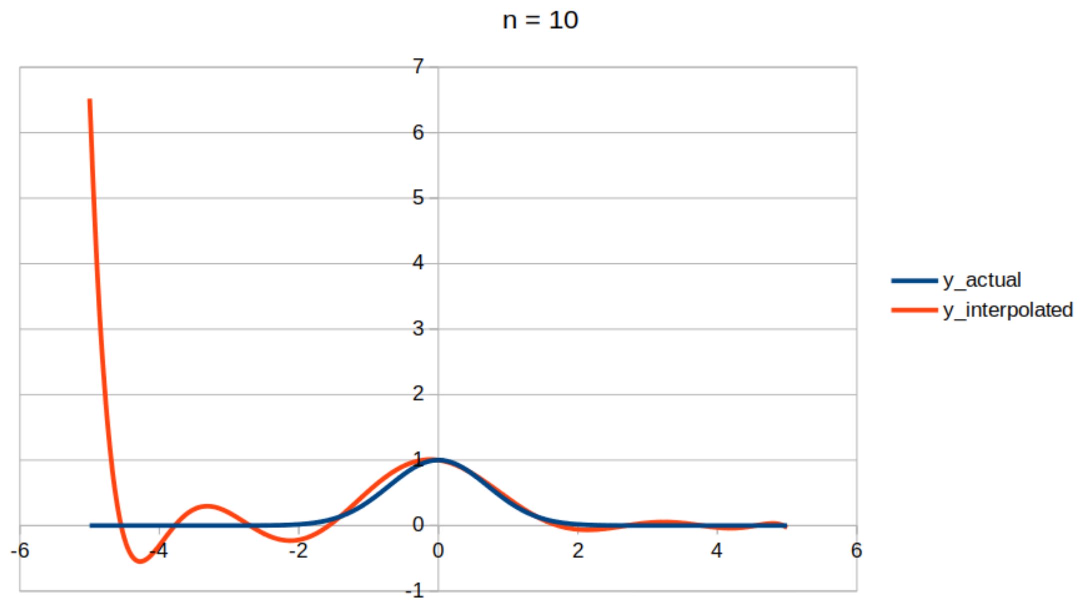


3.2 Wykresy interpolacji Lagrange'a dla funkcji $f(x) = e^{-x^2}$ z wykorzystaniem wielomianu Czybyszewa

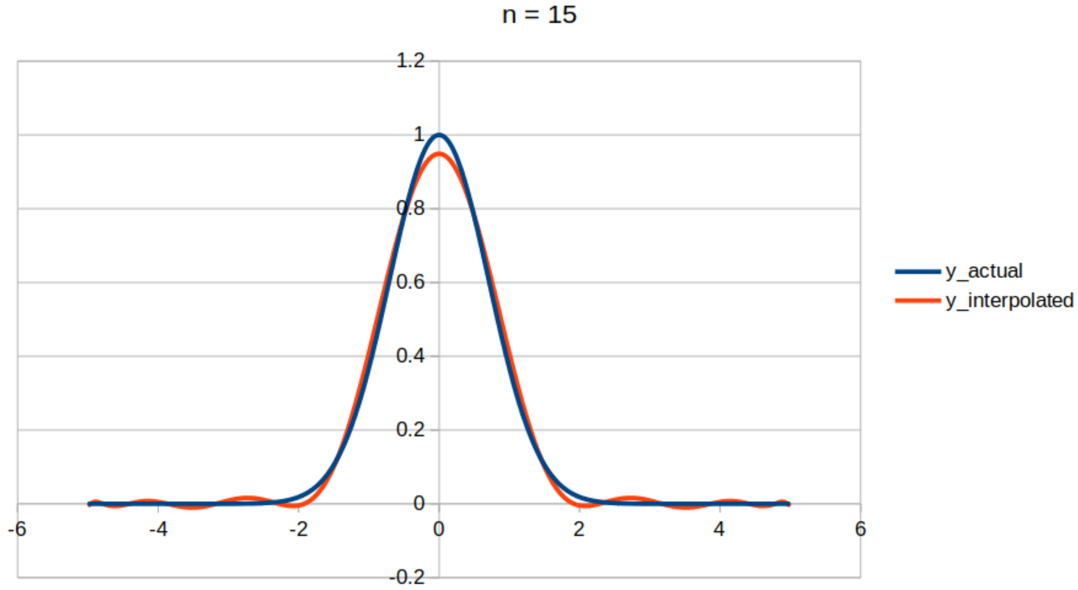
3.2.1 $n = 5$



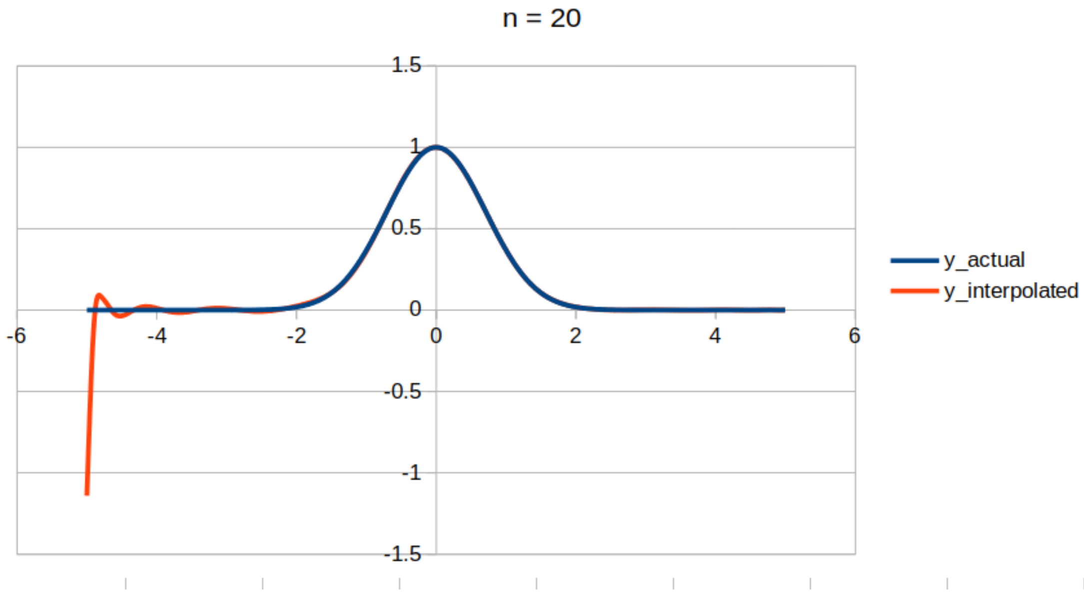
3.2.2 $n = 10$



3.2.3 $n = 15$

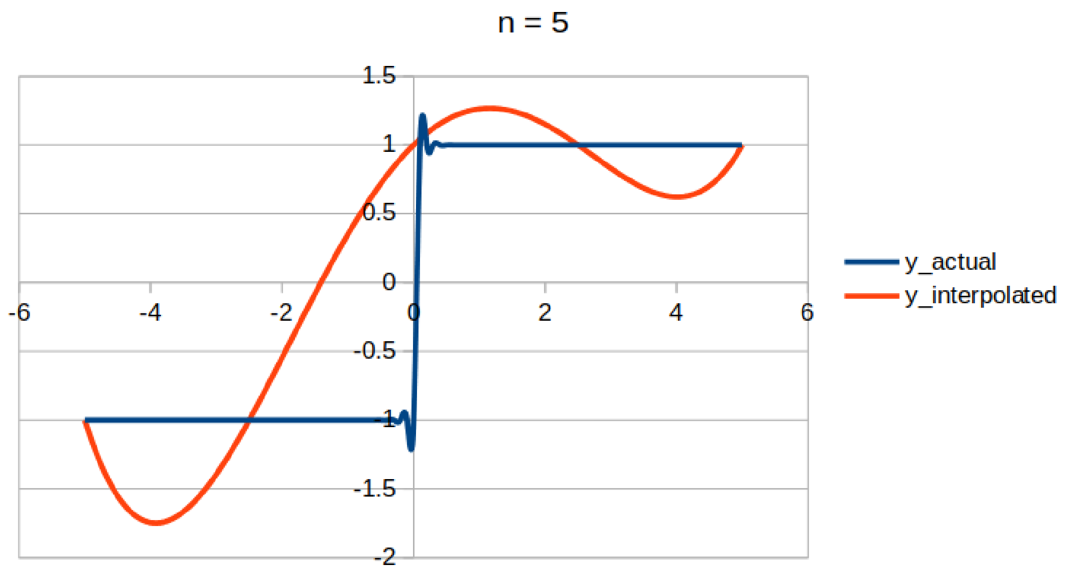


3.2.4 $n = 20$

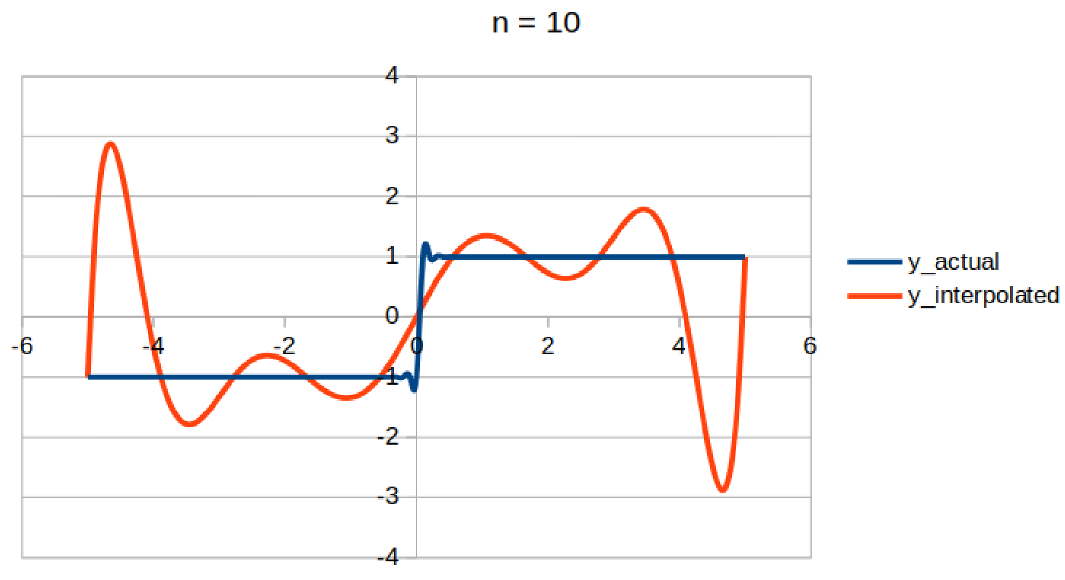


3.3 Wykresy interpolacji Lagrange'a dla funkcji $f(x) = -1$ dla $x < 0, f(x) =$ dla $x \geq 0$

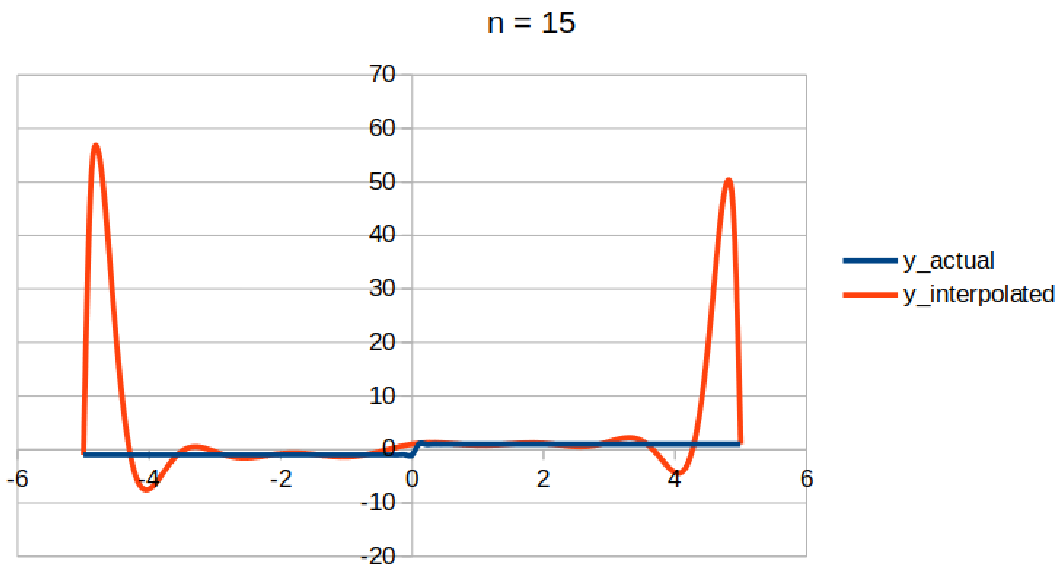
3.3.1 $n = 5$



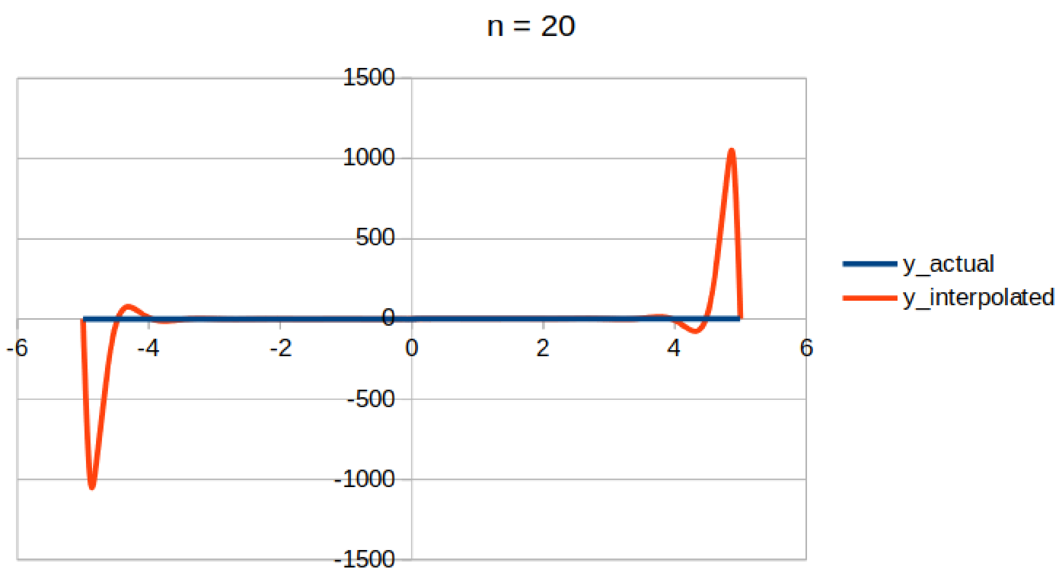
3.3.2 $n = 10$



3.3.3 $n = 15$

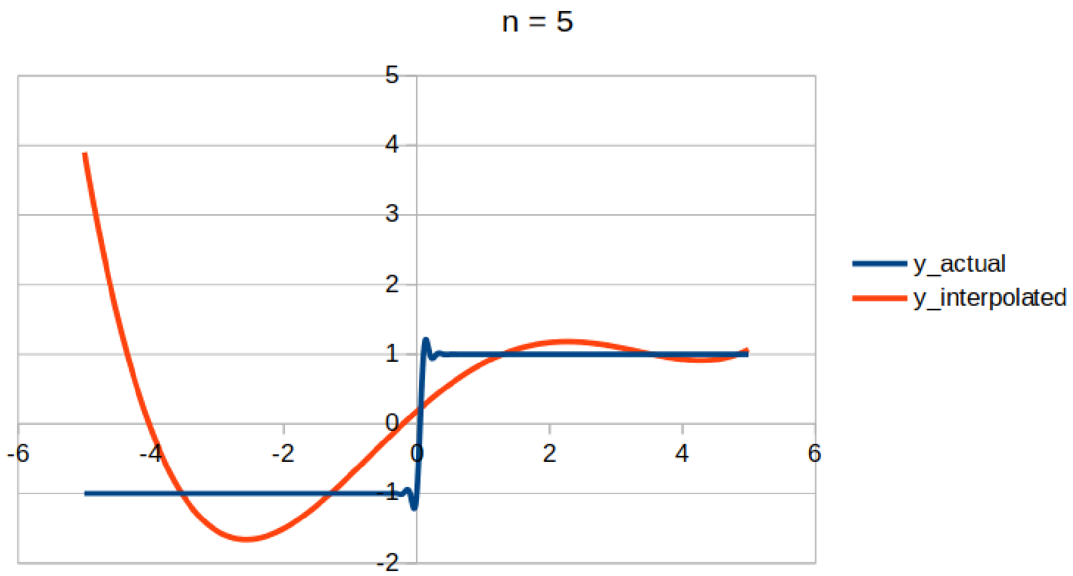


3.3.4 $n = 20$

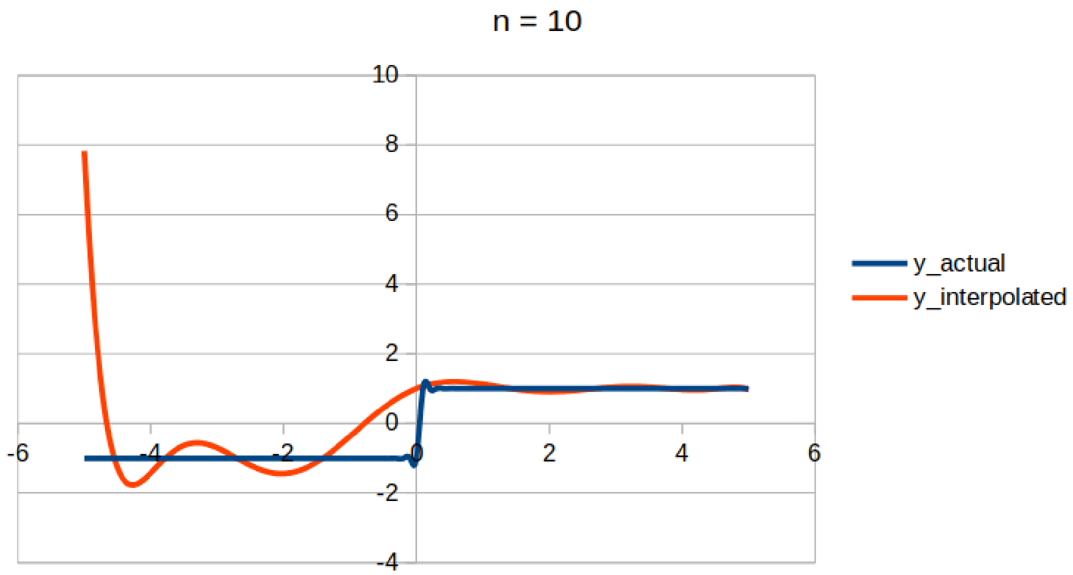


3.4 Wykresy interpolacji Lagrange'a dla funkcji $f(x) = -1$ dla $x < 0, f(x) =$ dla $x \geq 0$ z wykorzystaniem wielomianu Czybyszewa

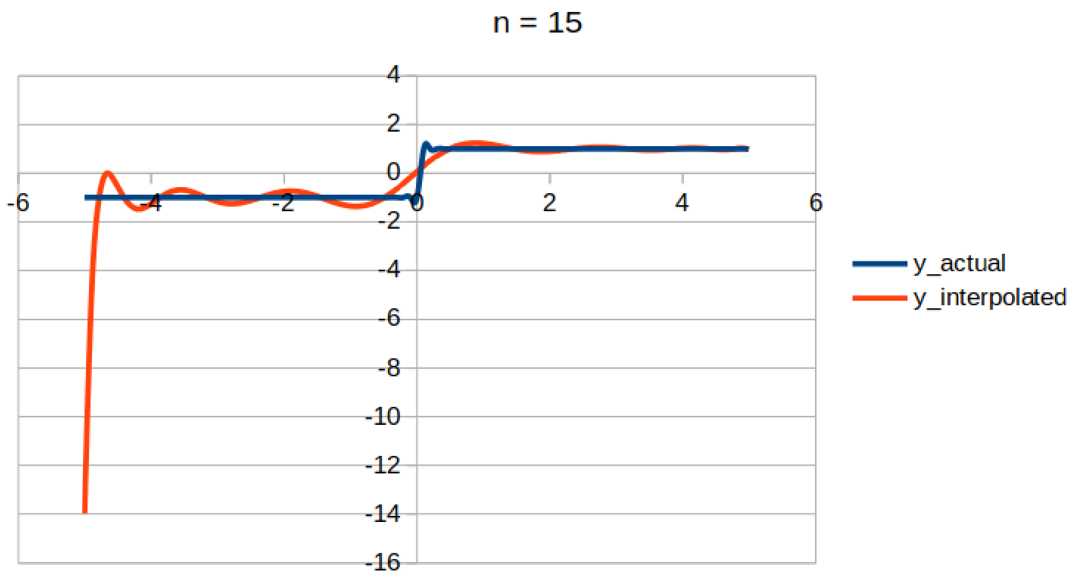
3.4.1 $n = 5$



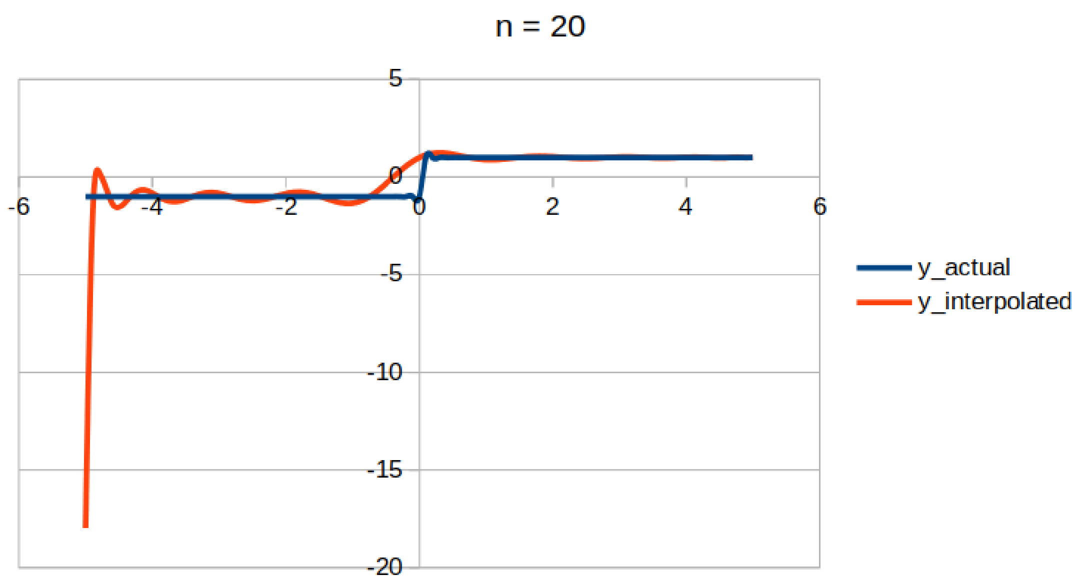
3.4.2 $n = 10$



3.4.3 $n = 15$

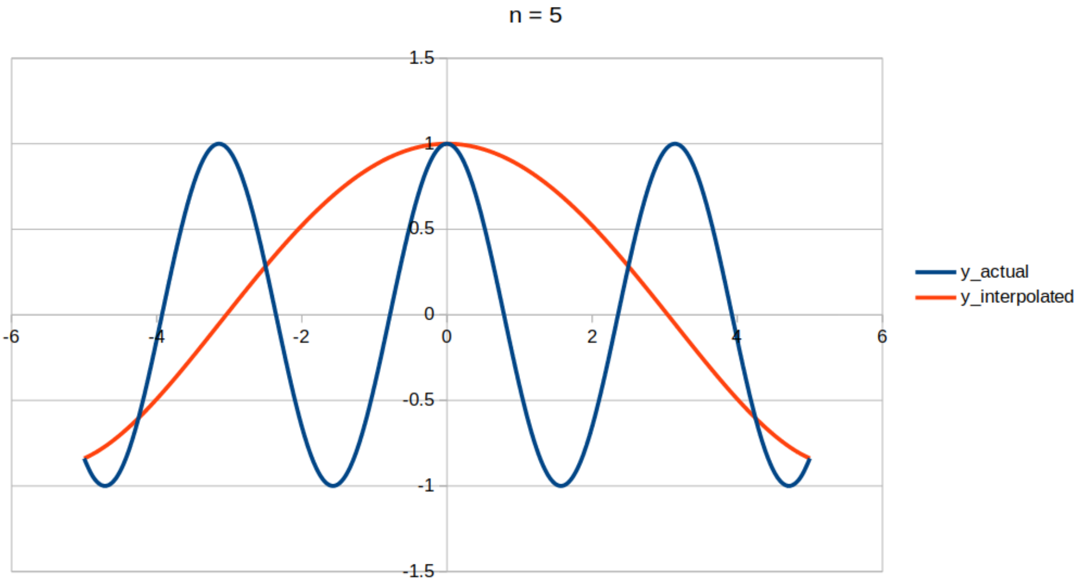


3.4.4 $n = 20$

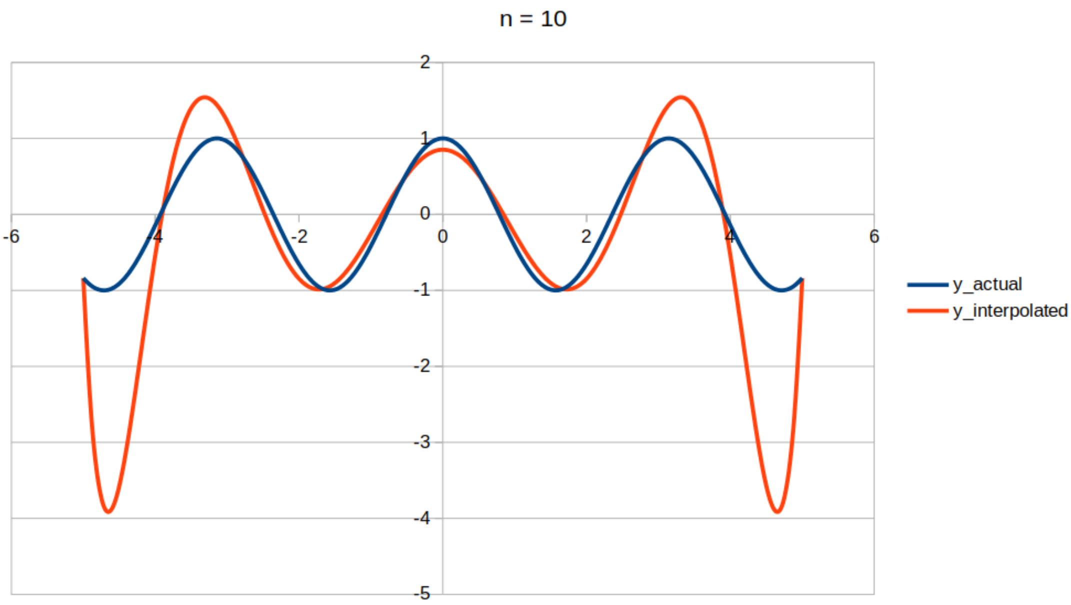


3.5 Wykresy interpolacji Lagrange'a dla funkcji $f(x) = \cos 2x$

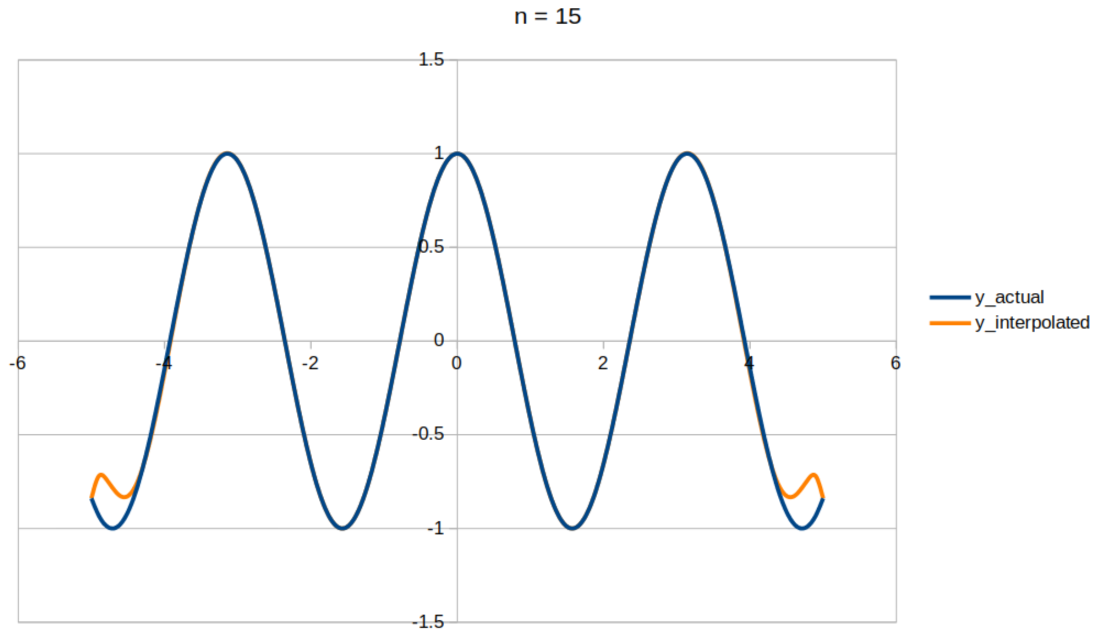
3.5.1 $n = 5$



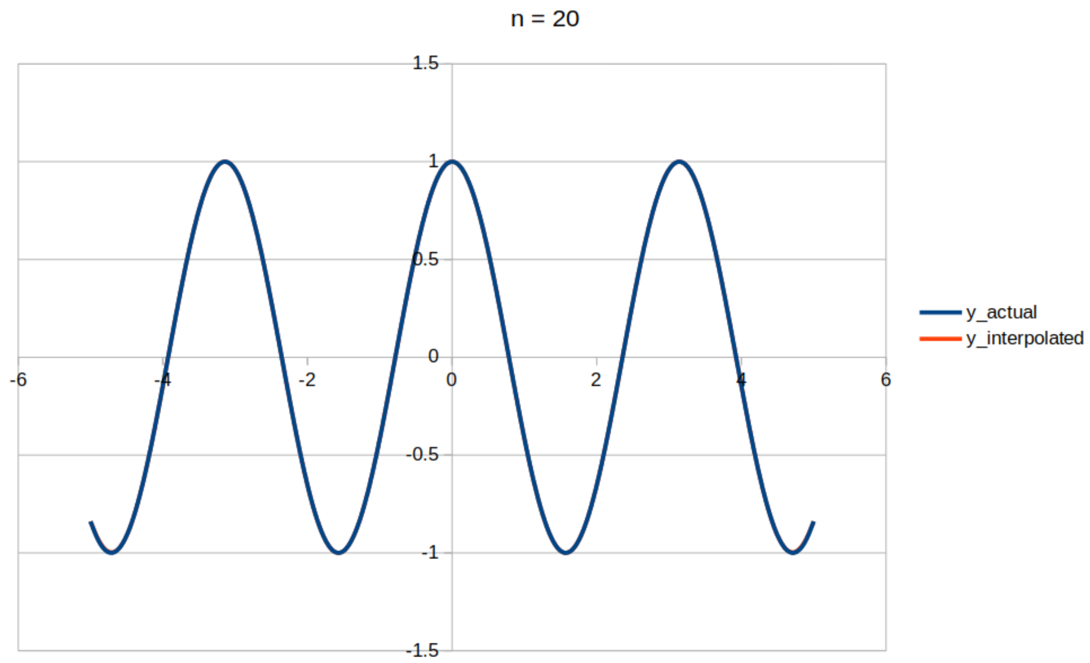
3.5.2 $n = 10$



3.5.3 $n = 15$

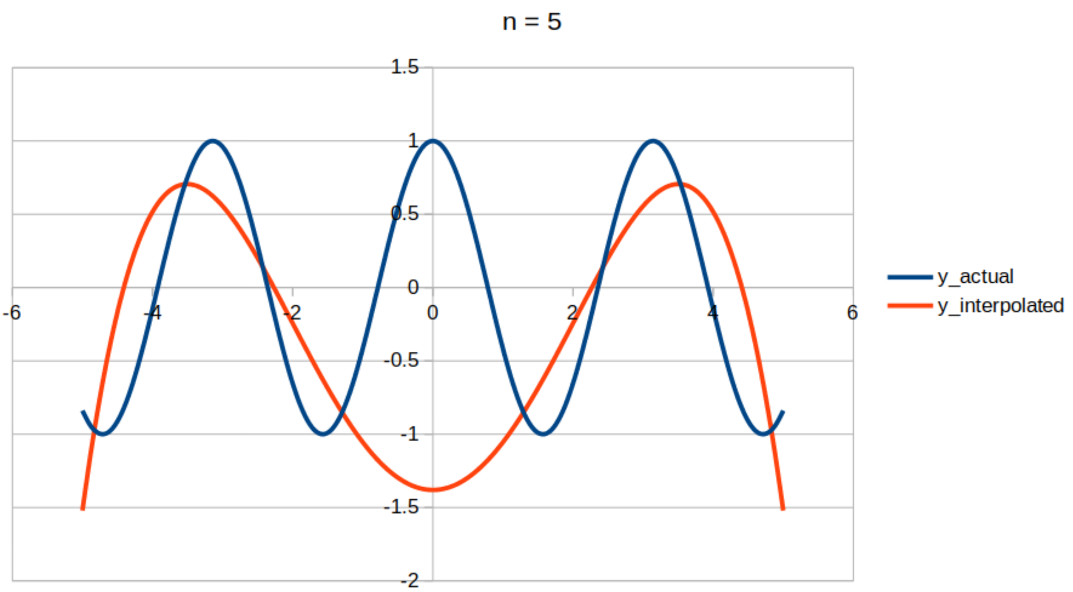


3.5.4 $n = 20$

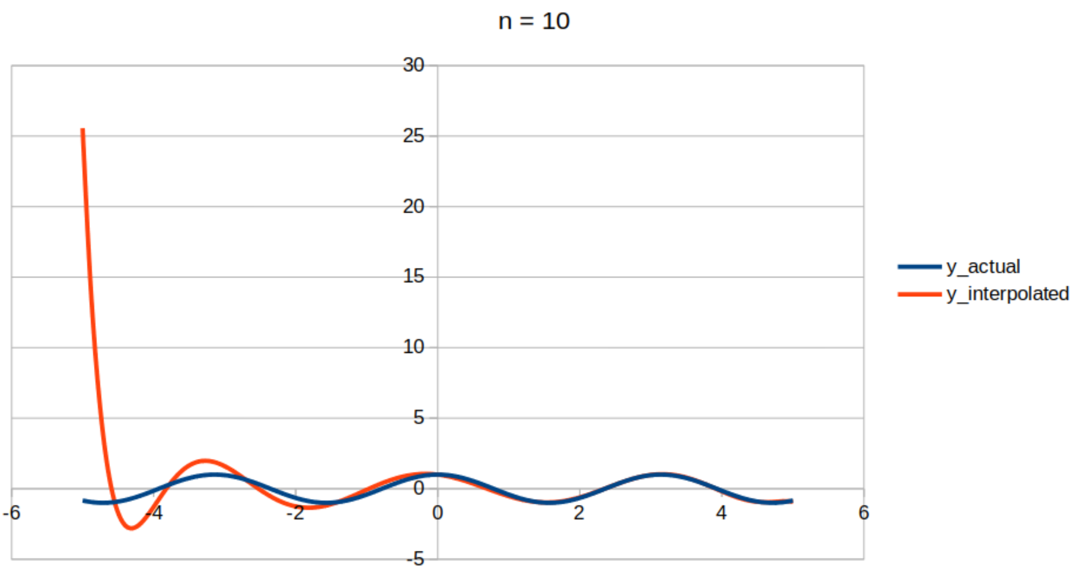


3.6 Wykresy interpolacji Lagrange'a dla funkcji $f(x) = \cos 2x$ z wykorzystaniem wielomianu Czybyszewa

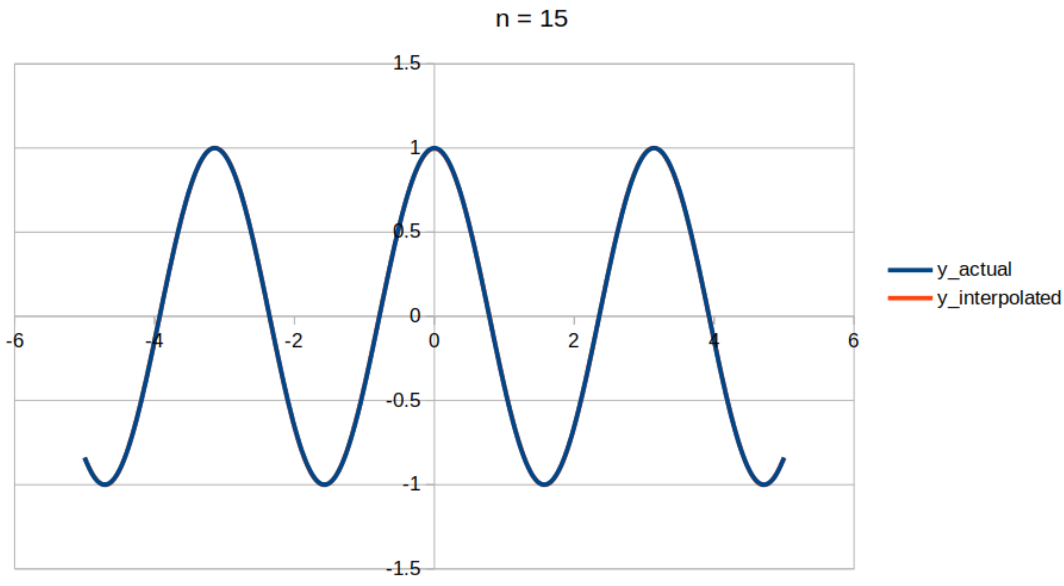
3.6.1 $n = 5$



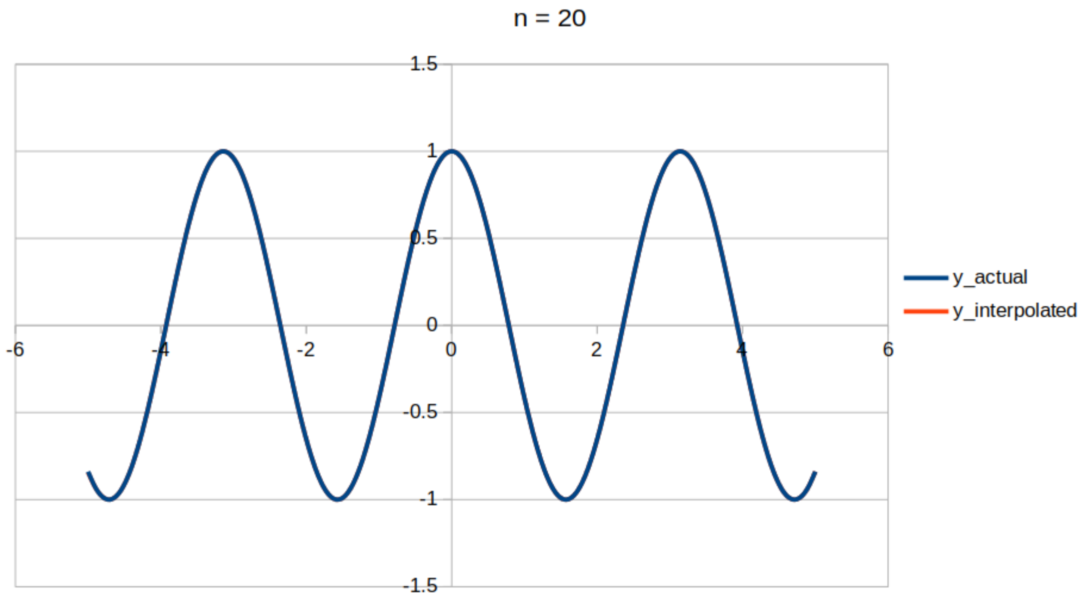
3.6.2 $n = 10$



3.6.3 $n = 15$



3.6.4 $n = 20$



4 Wnioski

- Interpolacja funkcji, której przebieg znacznie różni się od przebiegu wielomianu interpolacyjnego, może nie dawać satysfakcjonujących wyników, zwłaszcza przy dużej liczbie węzłów. Wpływ na to mają pojawiające się ekstrema w funkcji interpolującej.
- Po optymalizacji procesu interpolacji, wartości funkcji interpolującej praktycznie pokrywają się z wartościami dokładnymi. To sugeruje, że optymalizacja może znacznie poprawić jakość interpolacji.
- Interpolacja, np. za pomocą metody LaGrange'a, pozwala na wyznaczenie przybliżonych wartości funkcji w punktach nie będących węzłami. Jest to szczególnie użyteczne w sytuacjach, kiedy postać funkcyjna może nawet nie być znana.