BARYCENTRIC LAGRANGE INTERPOLATION

Brian Caravantes

Course: Math 4401 Numerical Methods

Faculty: Barry McQuarrie

University of Minnesota, Morris

Fall 2016

OUTLINE

I. Overview
   I. Interest
   II. The Problem
II. Recall
   I. Lagrange Interpolation
   II. Newton Interpolation
   III. Shortcomings
III. Calculations
   I. Derivation
   II. Variation of Nodes
   III. Examples and Comparisons
IV. Conclusion
   I. Applications
   II. In the end

OUTLINE

OVERVIEW: INTEREST

. "Lagrangian interpolation is praised for analytic utility and beauty but deplored for numerical practice."

OVERVIEW: THE PROBLEM

The Problem is that many people think that Lagrange Interpolation is at most a decent method for approximation

Many just think of it as a theoretical method

How do we show that it can be used as trustable approximating tool?

OUTLINE

I. Overview
   I. Interest
   II. The Problem
II. Recall
   I. Lagrange Interpolation
   II. Newton Interpolation
   III. Shortcomings
III. Calculations
   I. Derivation
   II. Variation of Nodes
   III. Examples and Comparisons
IV. Conclusion
   I. Applications
   II. In the end

DEF:

Flops: or Operations such as multiplication or division plus an addition or subtraction

- Will be described in $O(n)$ or $O(n^2)$ in terms of speed

(Big O Notation)

RECALL: LAGRANGE INTERPOLATION

Lagrange Form

$$\text{p}(x) = \sum_{j=0}^{n} f_j l_j(x), \quad (1)$$

where

$$l_j(x) = \frac{\prod_{k=0,k\neq j}^{n}(x - x_k)}{\prod_{k=0,k\neq j}^{n}(x_j - x_k)}$$

RECALL: NEWTON INTERPOLATION (DIVIDED DIFFERENCES)

$$f\left[x_j, x_{j+1}, \ldots x_{k-1,} x_k\right] = \frac{f\left[\, x_{j+1}, \ldots x_{k,}\right] - f\left[\, x_j, \ldots x_{k-1,}\right]}{x_k - x_j},$$

- $p(x) = f[x_0] + f[x_0, x_1](x - x_0) + f[x_0, x_1, x_2](x - x_0)(x - x_1) + \cdots + f[x_0, x_1, \ldots, x_n](x - x_0)(x - x_1)\ldots(x - x_{n-1})$

(2)

SHORTCOMINGS

- When computing p(x) Lagrangian Interpolation takes $O(n^2)$ flops
  - Computing (x_{n+1}, f_{n+1}) requires new computation every time
- Goal: To find an equation that is both quick and a trustable approximating tool?

OUTLINE

I. Overview
   I. Interest
   II. The Problem
II. Recall
   I. Lagrange Interpolation
   II. Newton Interpolation
   III. Shortcomings
III. Calculations
   I. Derivation
   II. Variation of Nodes
   III. Examples and Comparisons
IV. Conclusion
   I. Applications
   II. In the end

CALCULATIONS: DERIVATION

- Let

$$l(x) = (x - x_0)(x - x_1)\dots(x - x_n),$$

Where $l(x)$ isnt a Lagrange polynomial but rather the numerator

We then can define the Barycentric Weights as

$$w_j = \frac{1}{\prod_{k \neq j}(x_j - x_k)}, \qquad j = 0, \dots n,$$

DERIVATION(CONT.)

Where $w_j = \frac{1}{l\prime(x_j)}$, now manipulating $l(x)$ by dividing by $(x - x_j)$ and solving for $l_j(x)$, we obtain (a)

$$l_j(x) = l(x)\frac{w_j}{x - x_j}.$$

Using this equation we can solve for $p(x)$

$$p(x) = l(x)\sum_{j=0}^{n}\frac{w_j}{x - x_j}f_j\,,$$

(Modified Lagrange Formula) (3)

SHORTCOMINGS

- When computing p(x) Lagrangian Interpolation takes $O(n^2)$ flops ✓
- Computing (x_{n+1}, f_{n+1}) requires new computation every time

- Goal: To find an equation that is both quick and a trustable approximating tool?

DERIVATION (CONT.)

Suppose we interpolate on some data f_j, that is the constant function 1

$$1 = \sum_{j=0}^{n} l_j(x) = l(x) \sum_{j=0}^{n} \frac{w_j}{x - x_j}$$

Now dividing $p(x) = l(x) \sum_{j=0}^{n} \frac{w_j}{x-x_j} f_j$ by $l(x) \sum_{j=0}^{n} \frac{w_j}{x-x_j}$ to obtain

$$p(x) = \frac{\sum_{j=0}^{n} \frac{w_j}{x - x_j} f_j}{\sum_{j=0}^{n} \frac{w_j}{x - x_j}}$$

(Barycentric Formula)(4)

SHORTCOMINGS

- When computing p(x) Lagrangian Interpolation takes $O(n^2)$ flops✓
- Computing (x_{n+1}, f_{n+1}) requires new computation every time✓

- Goal: To find an equation that is both quick and a trustable approximating tool?

CALCULATIONS: VARIATION OF NODES

Equidistant Points: where the spacing $h = \frac{2}{n}$

$w_j = \frac{(-1)^{n-j}\binom{n}{j}}{(h^n n!)}$ for the interval [-1,1]

Where $\binom{n}{j}$ *is the binomial coefficient* $\frac{n!}{(n-j)!\, j!}$

$w_j = (-1)^j \binom{n}{j}$ for any interval [a,b]

DEF:

- Runge Phenomenon: Describes extreme polynomial wiggle associated with high degree interpolation at evenly spaced nodes

VARIATION OF NODES

- Chebyshev Points

$$w_j = (-1)^j \delta_j \quad \delta_j = \begin{cases} \frac{1}{2} & j = 0 \ or \ j = n \\ 1 & otherwise \end{cases}$$

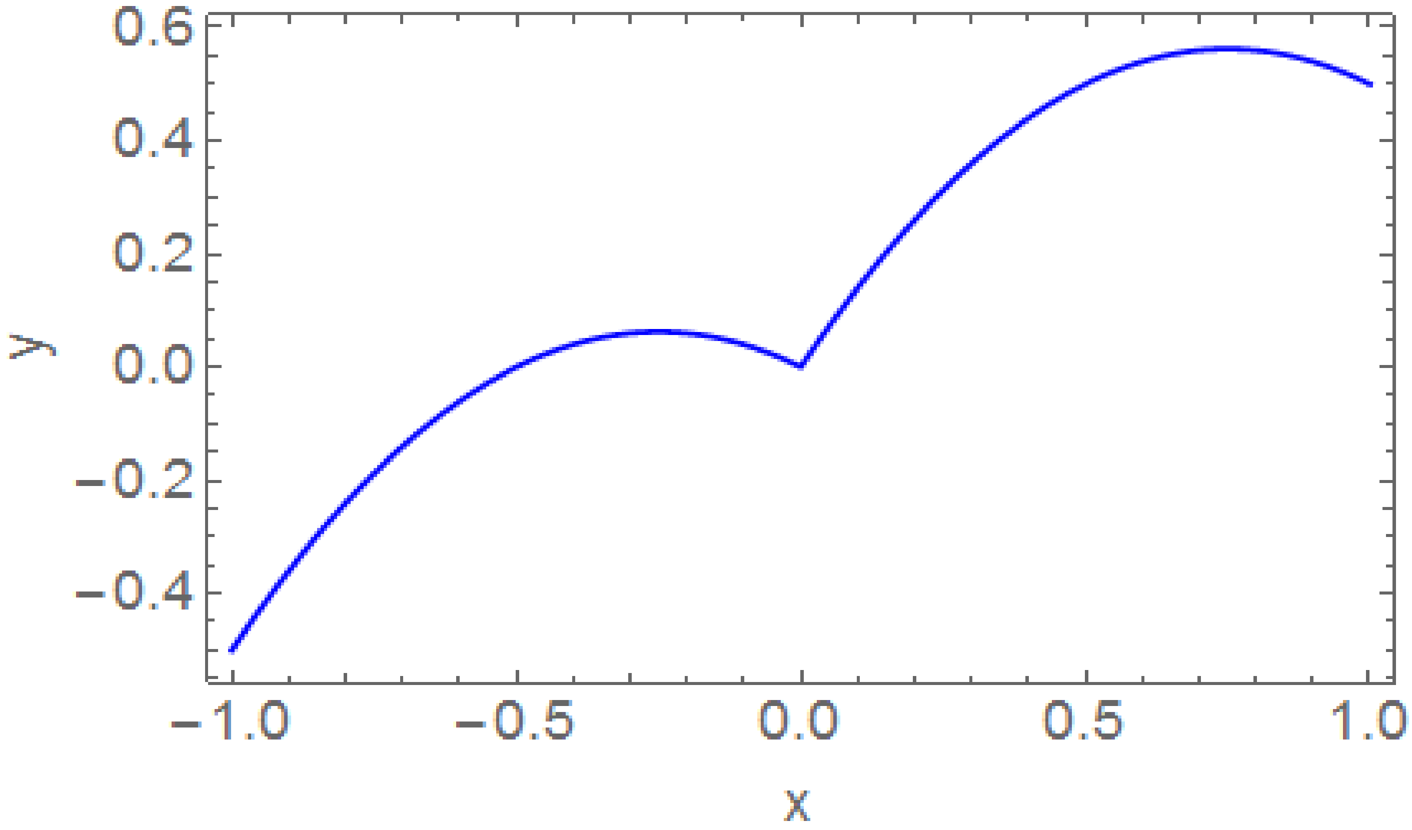
$$p(x) = \frac{\sum_{j=0}^{n} \frac{w_j}{x - x_j} f_j}{\sum_{j=0}^{n} \frac{w_j}{x - x_j}}$$

(Barycentric Formula)(4)

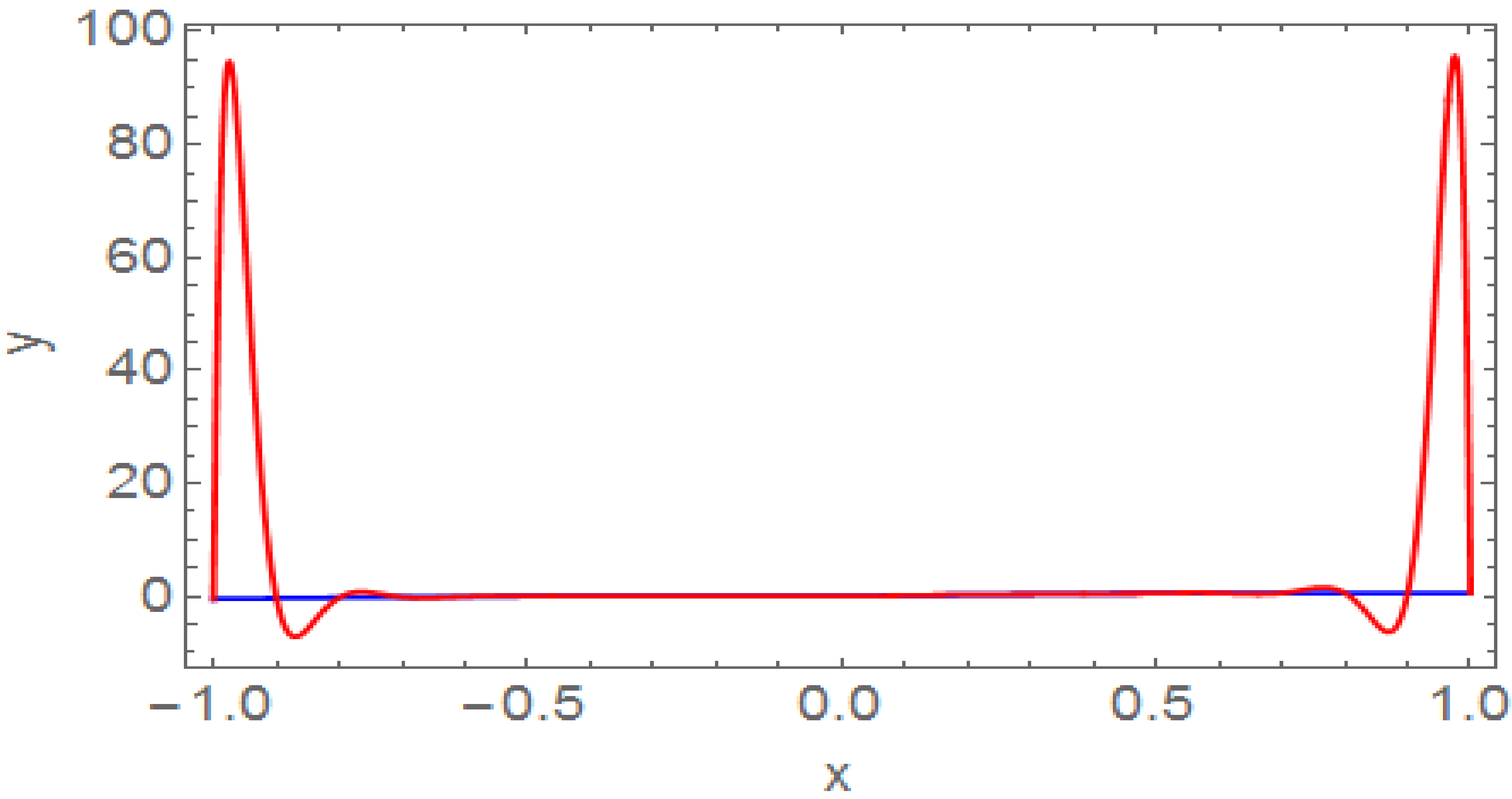
EXAMPLE 1: EQUIDISTANT NODES

- Consider the function $f(x)$ on the interval [-1,1]

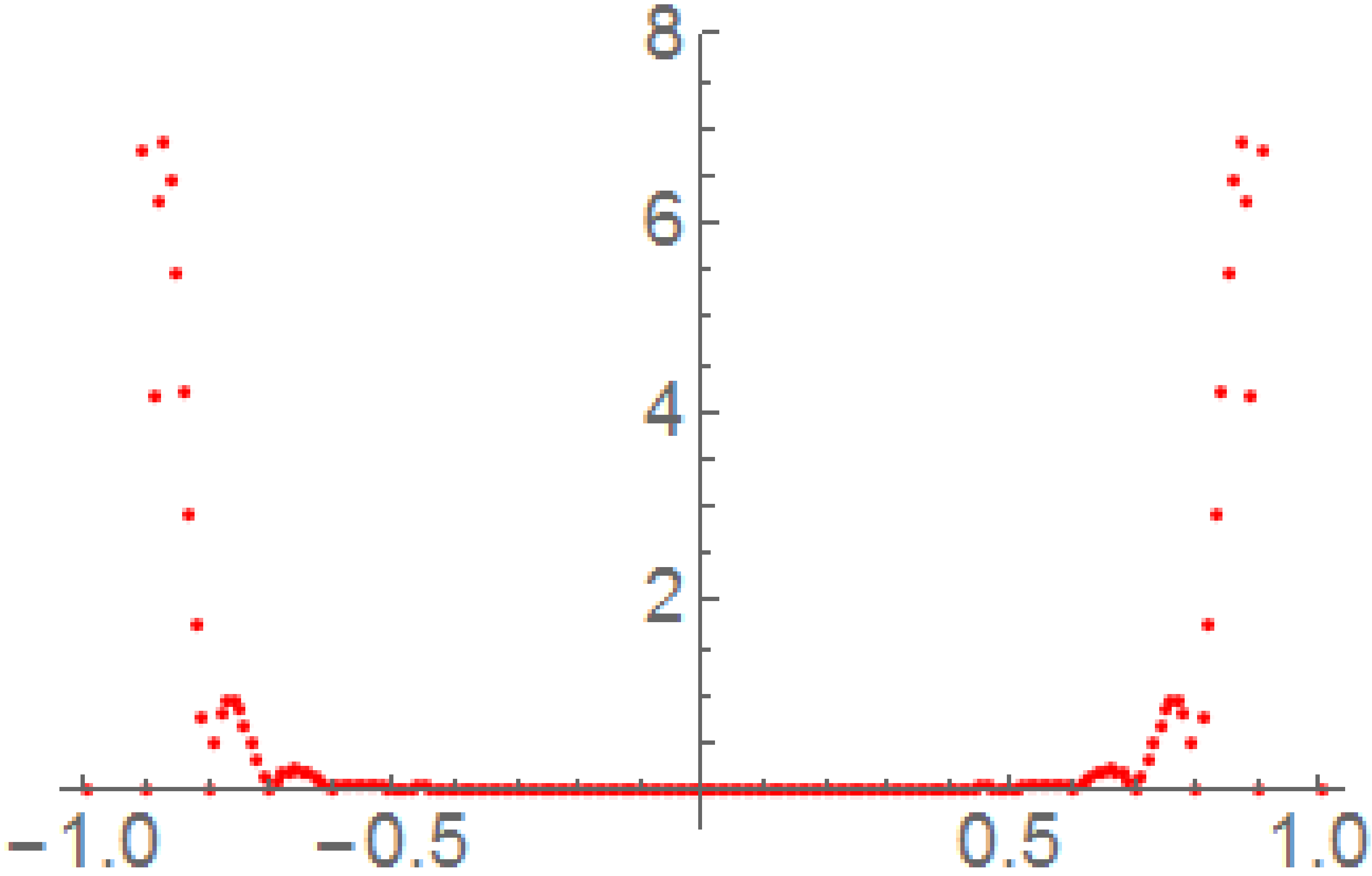
$$f(x) = |x| + \frac{x}{2} - x^2$$



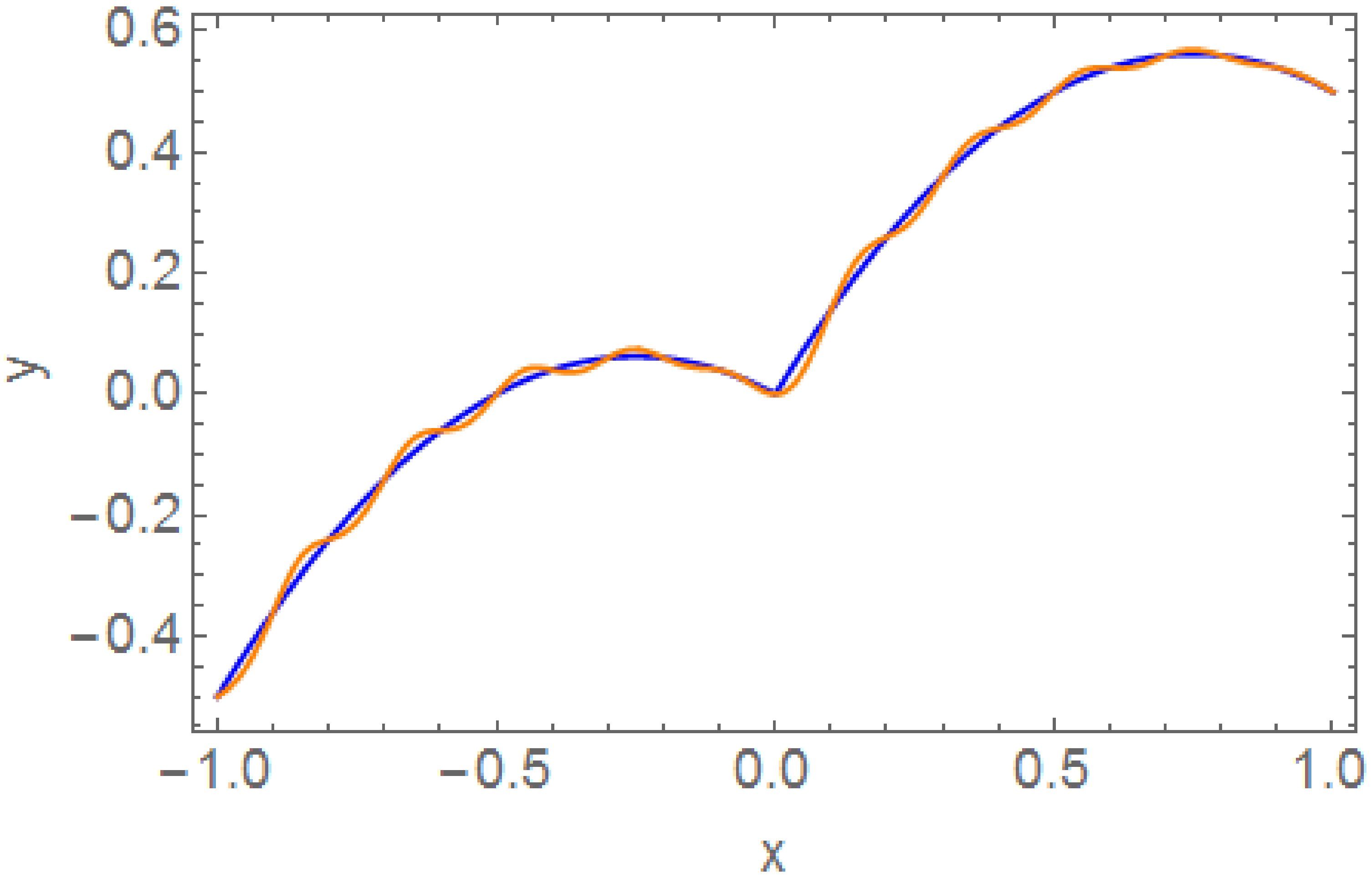
LAGRANGE INTERPOLATION



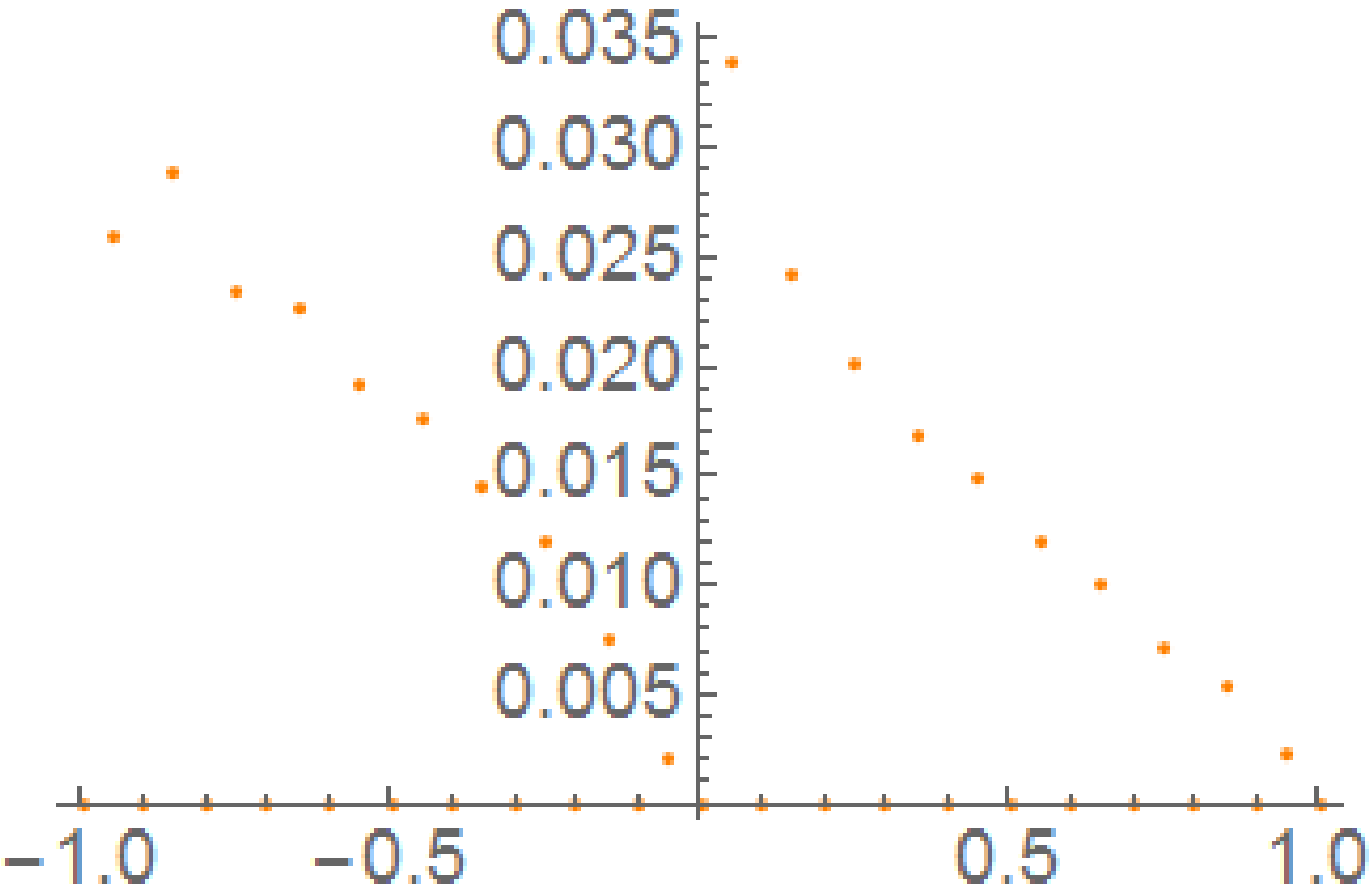
ERROR



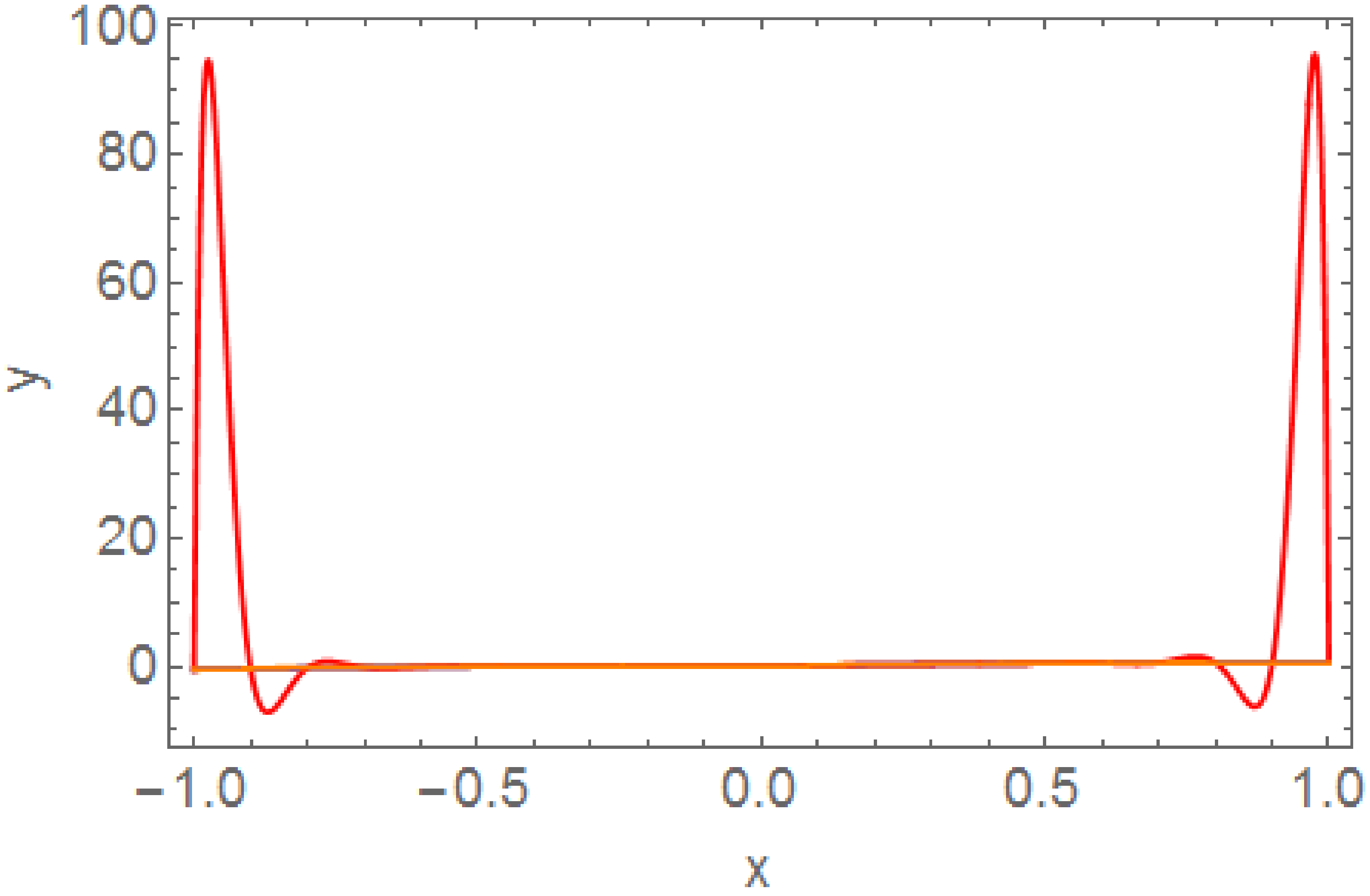
BARYCENTRIC INTERPOLATION



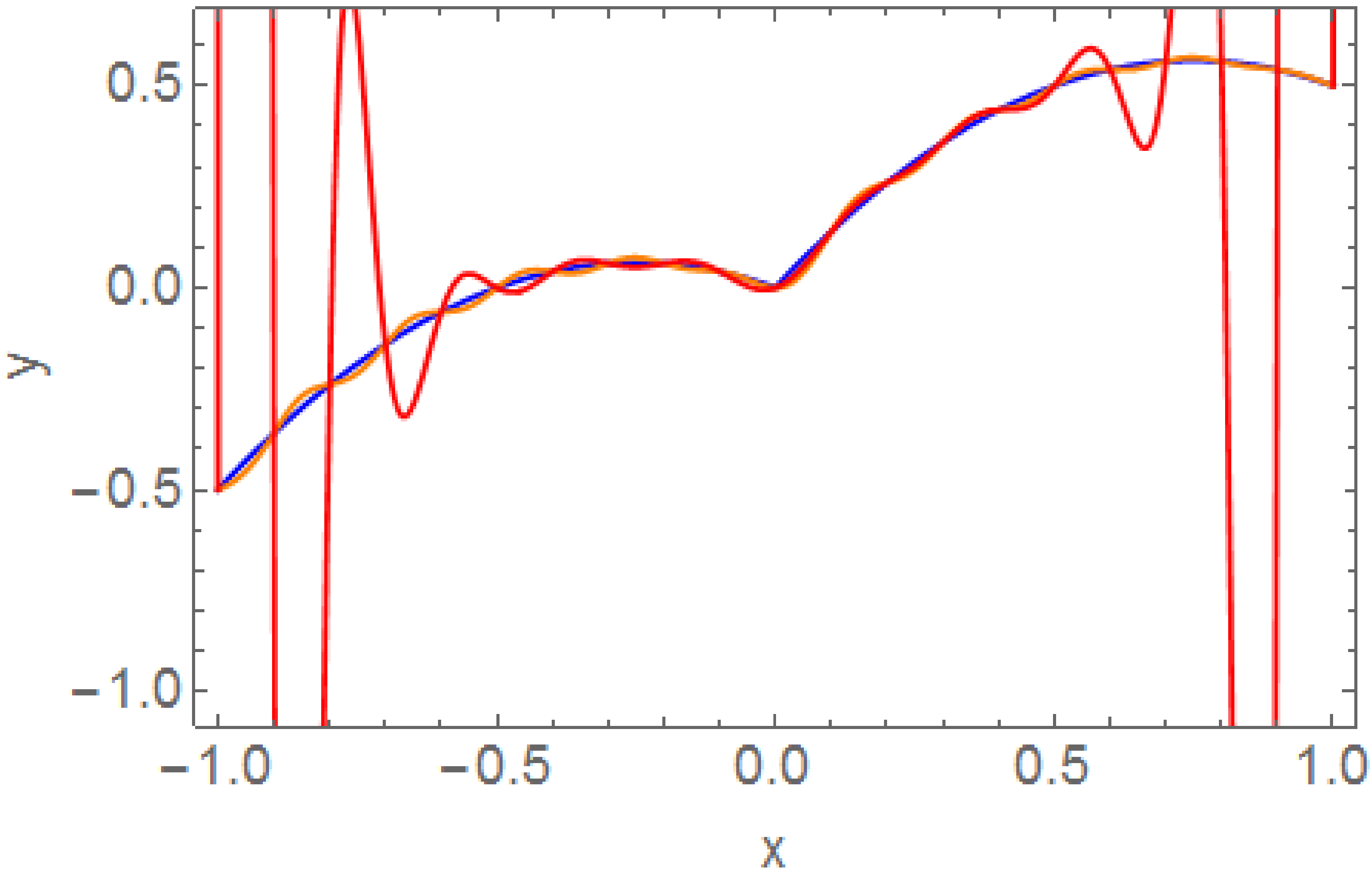
ERROR



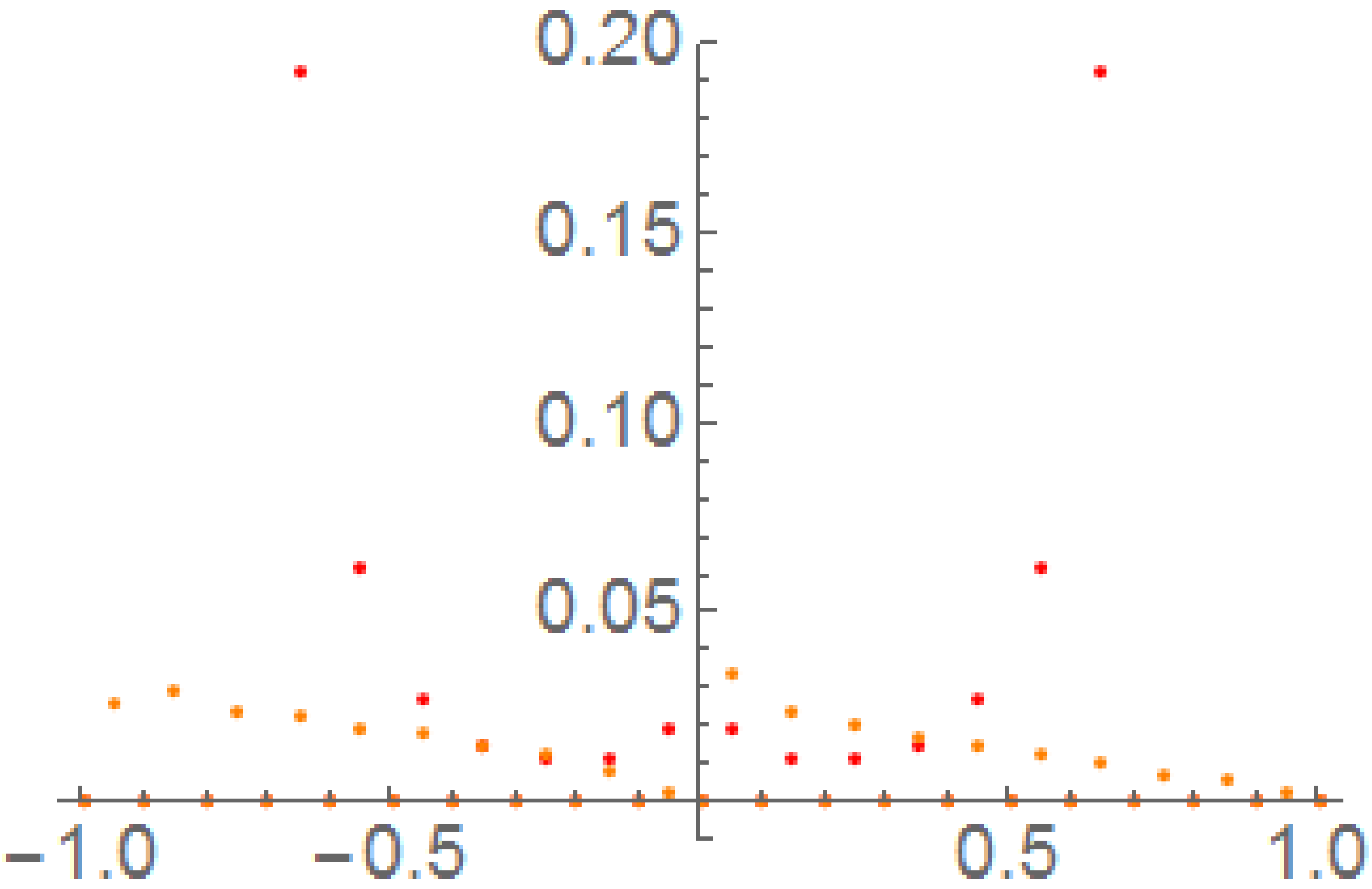
CALCULATIONS: COMPARISON



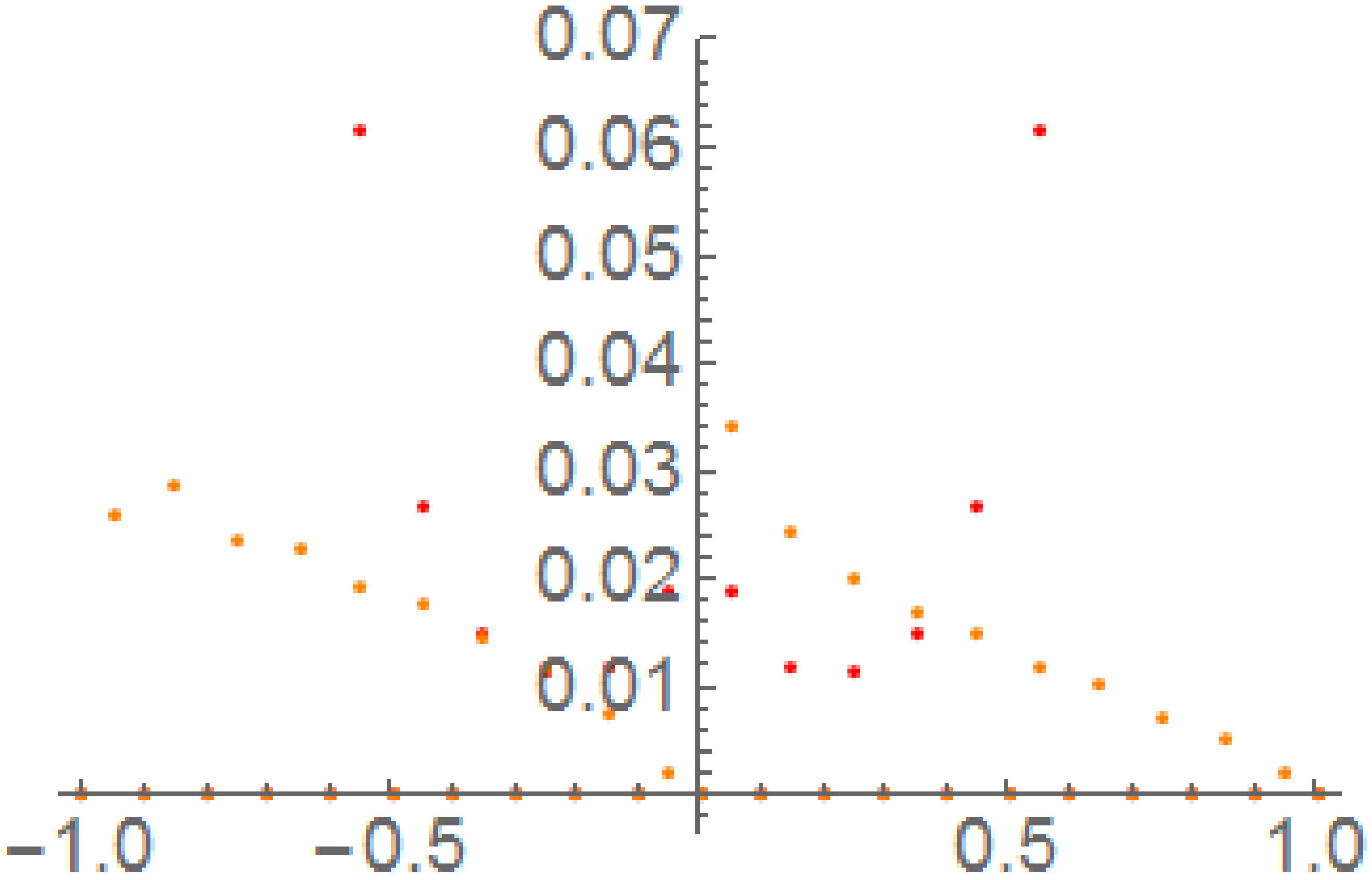
CLOSER LOOK



ERROR COMPARISON



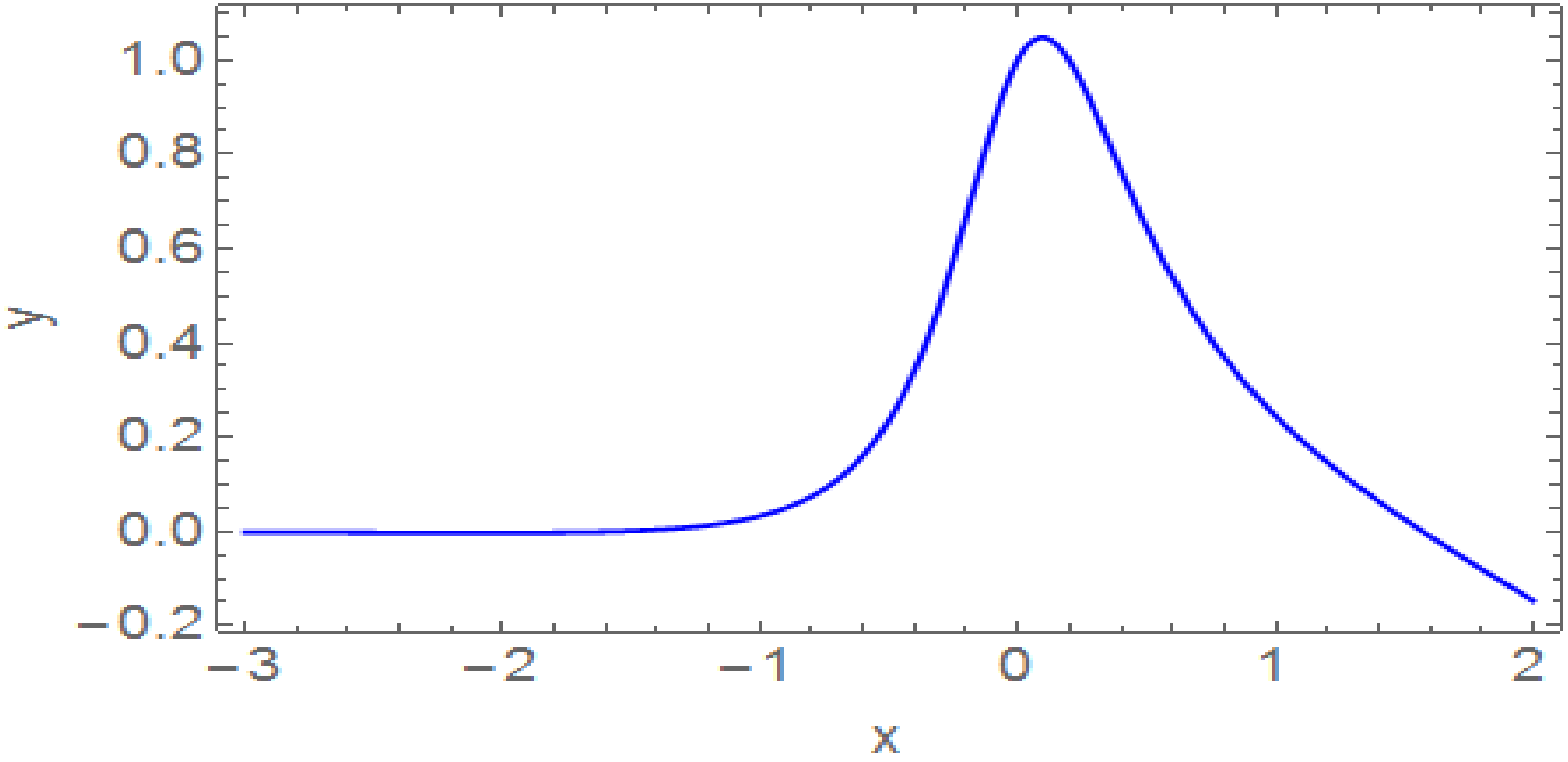
CLOSER LOOK



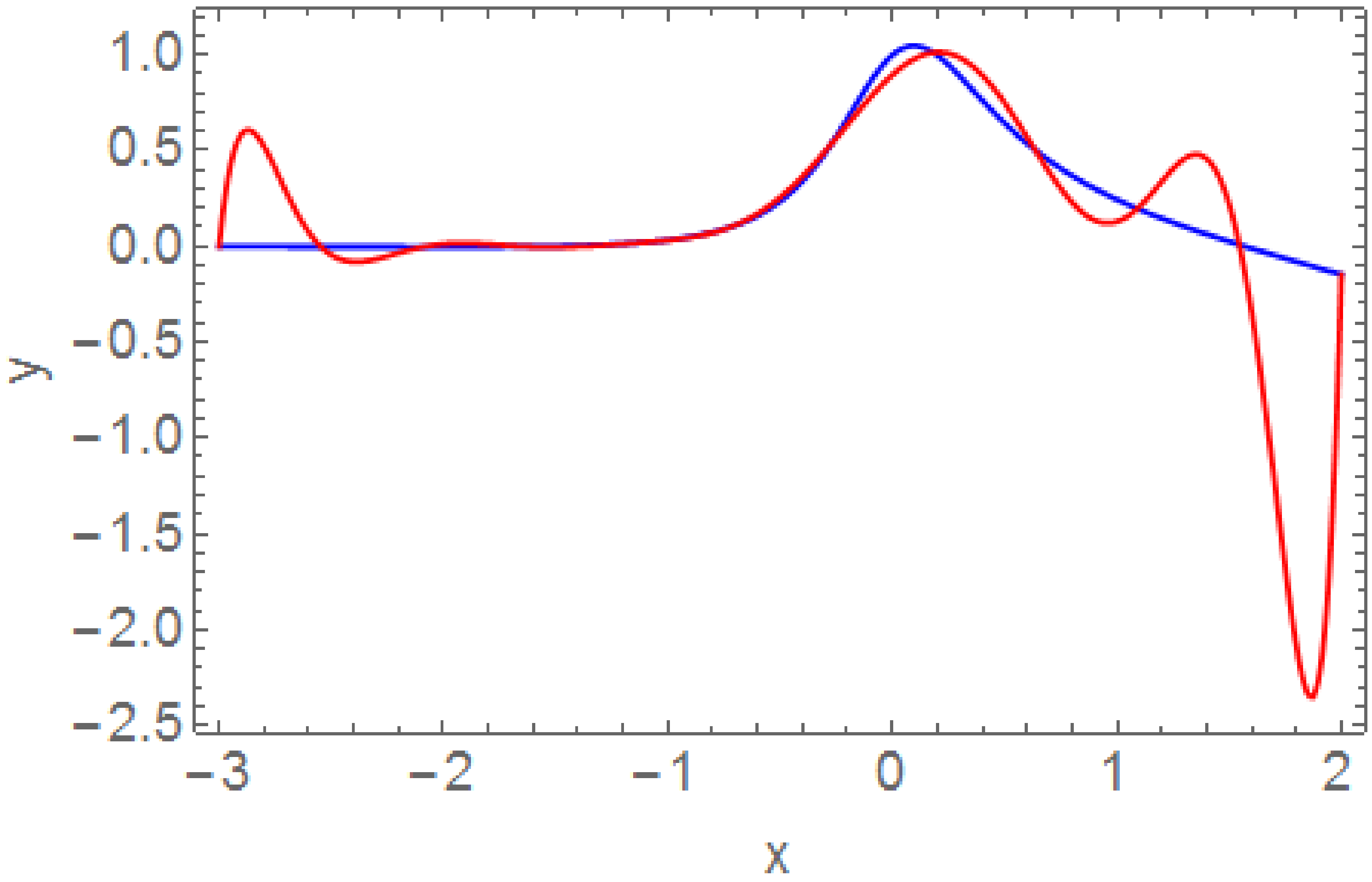
EXAMPLE 2: CHEBYSHEV POINTS

- Consider the function $f(x)$ on the interval [-3,2]

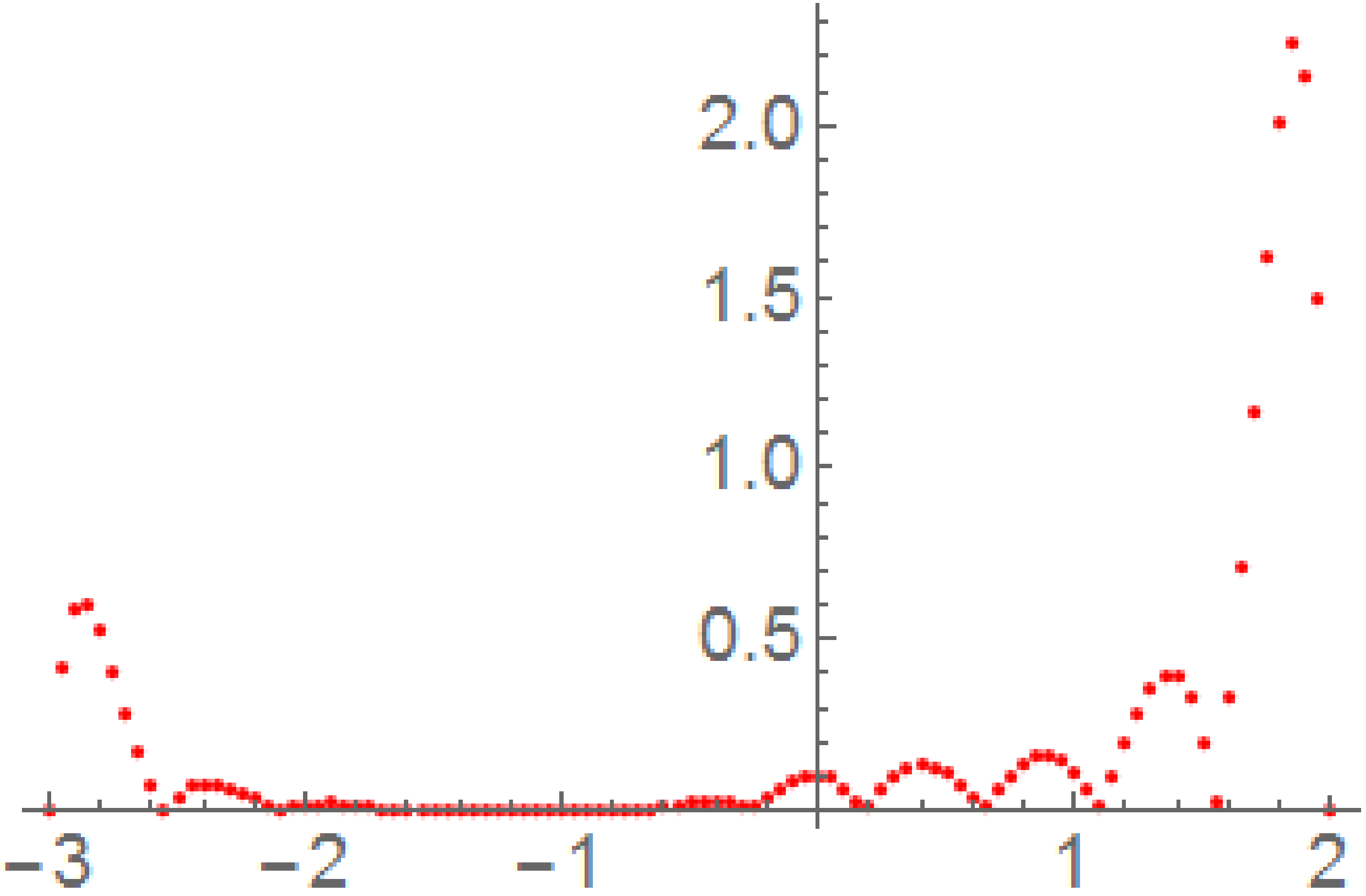
$$f(x) = \frac{Cos(x) * e^{x}}{1+5x^{2}}$$



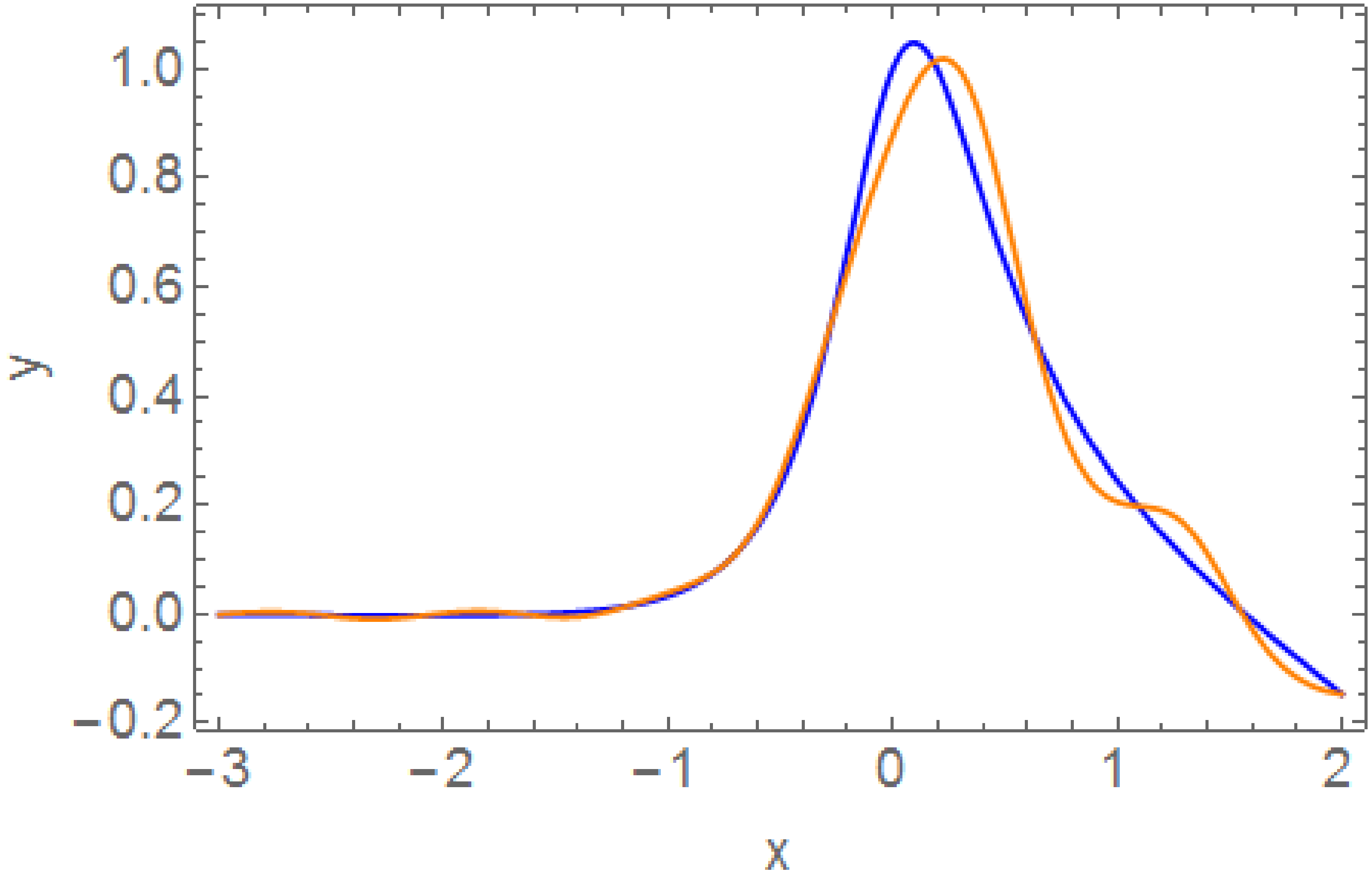
LAGRANGE INTERPOLATION



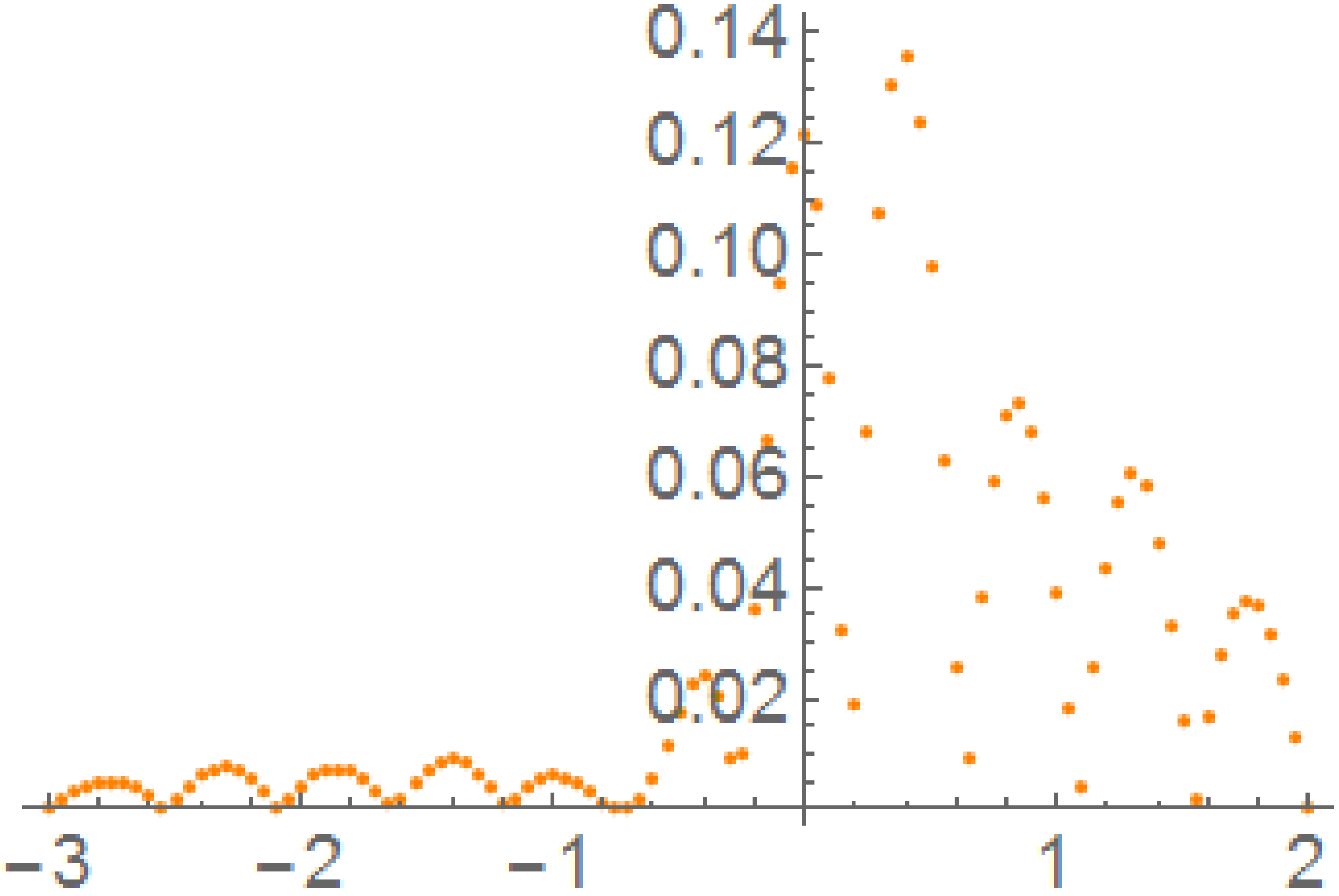
ERROR



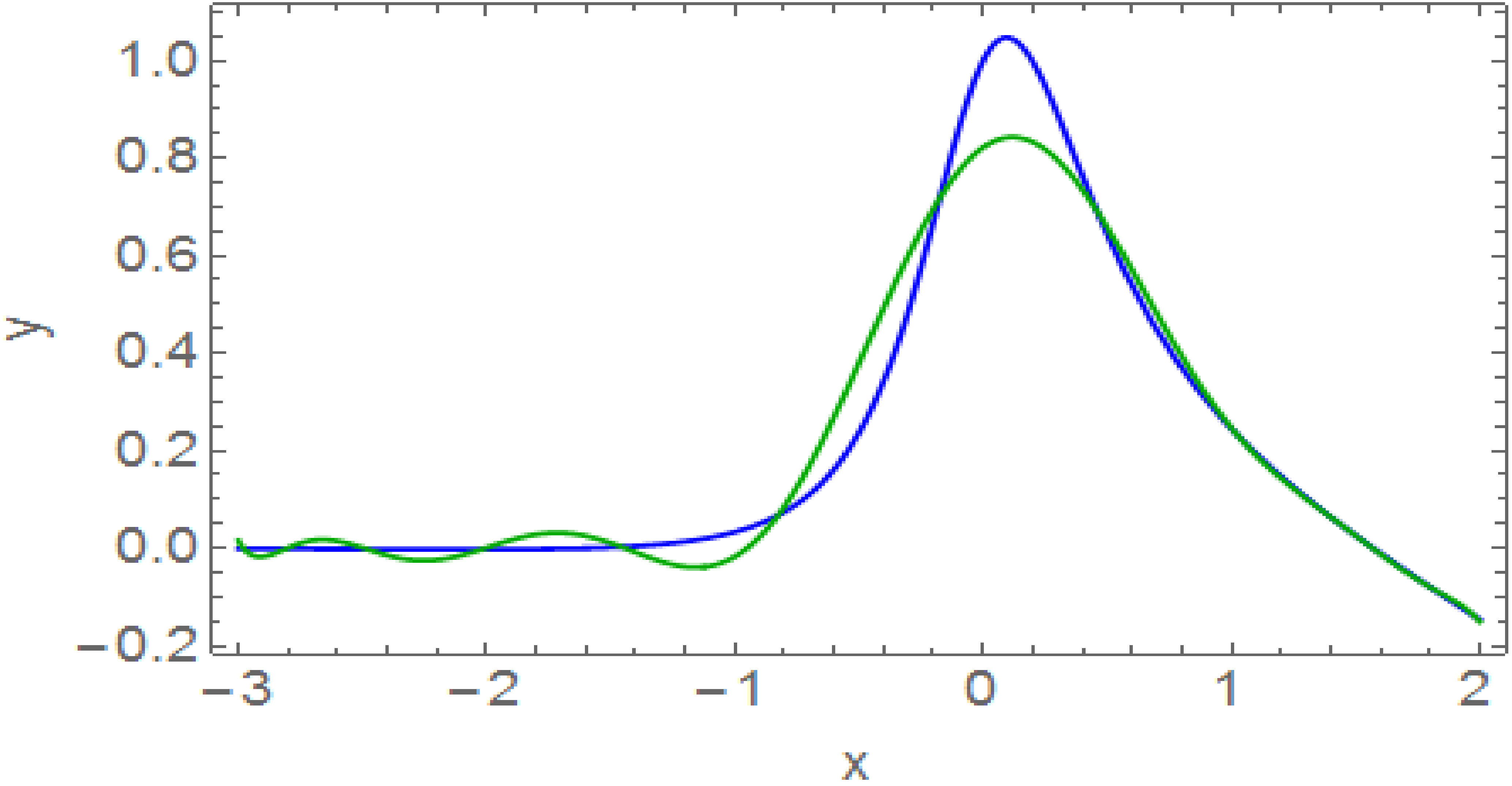
BARYCENTRIC LAGRANGE INTERPOLATION



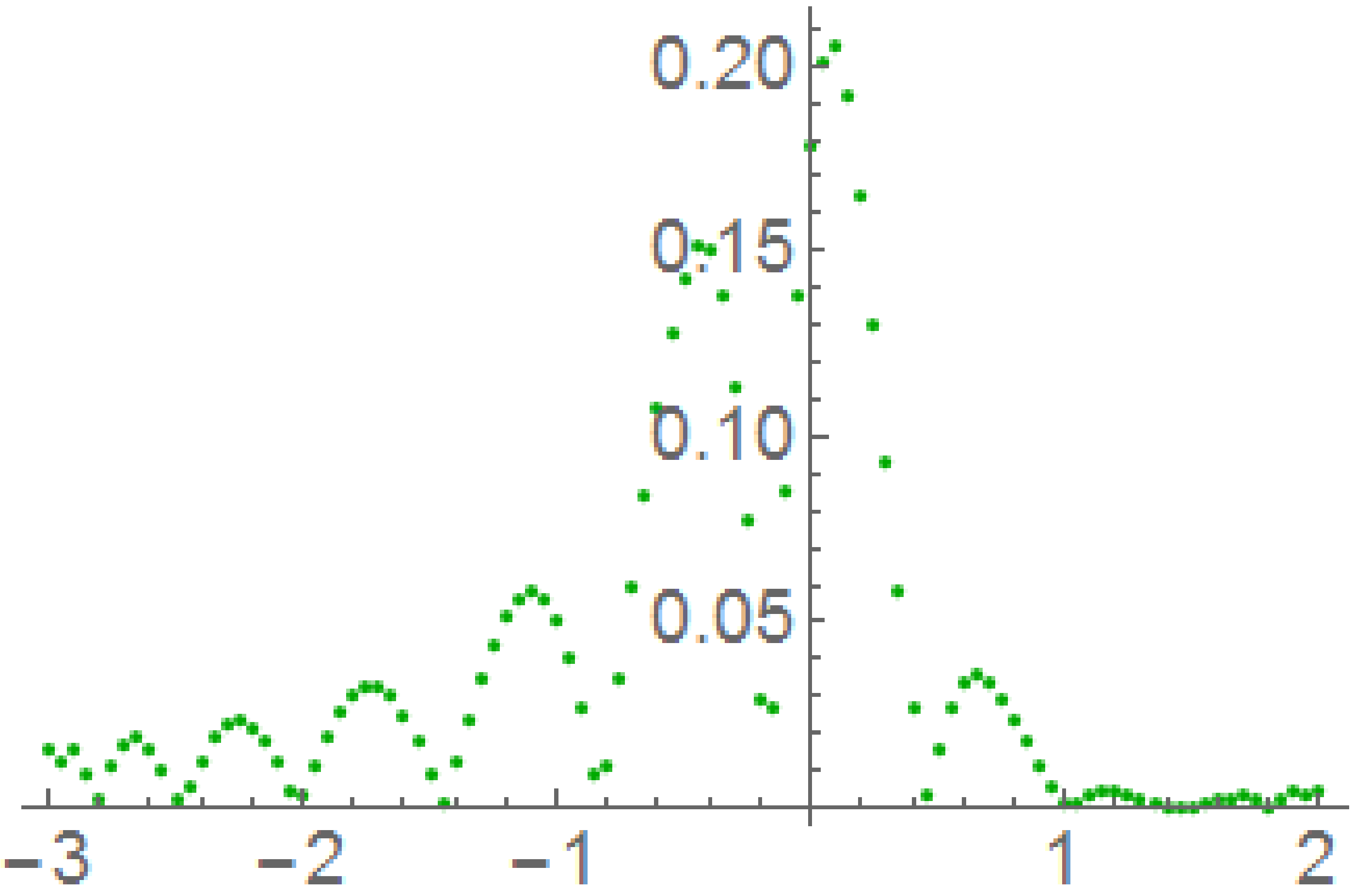
ERROR



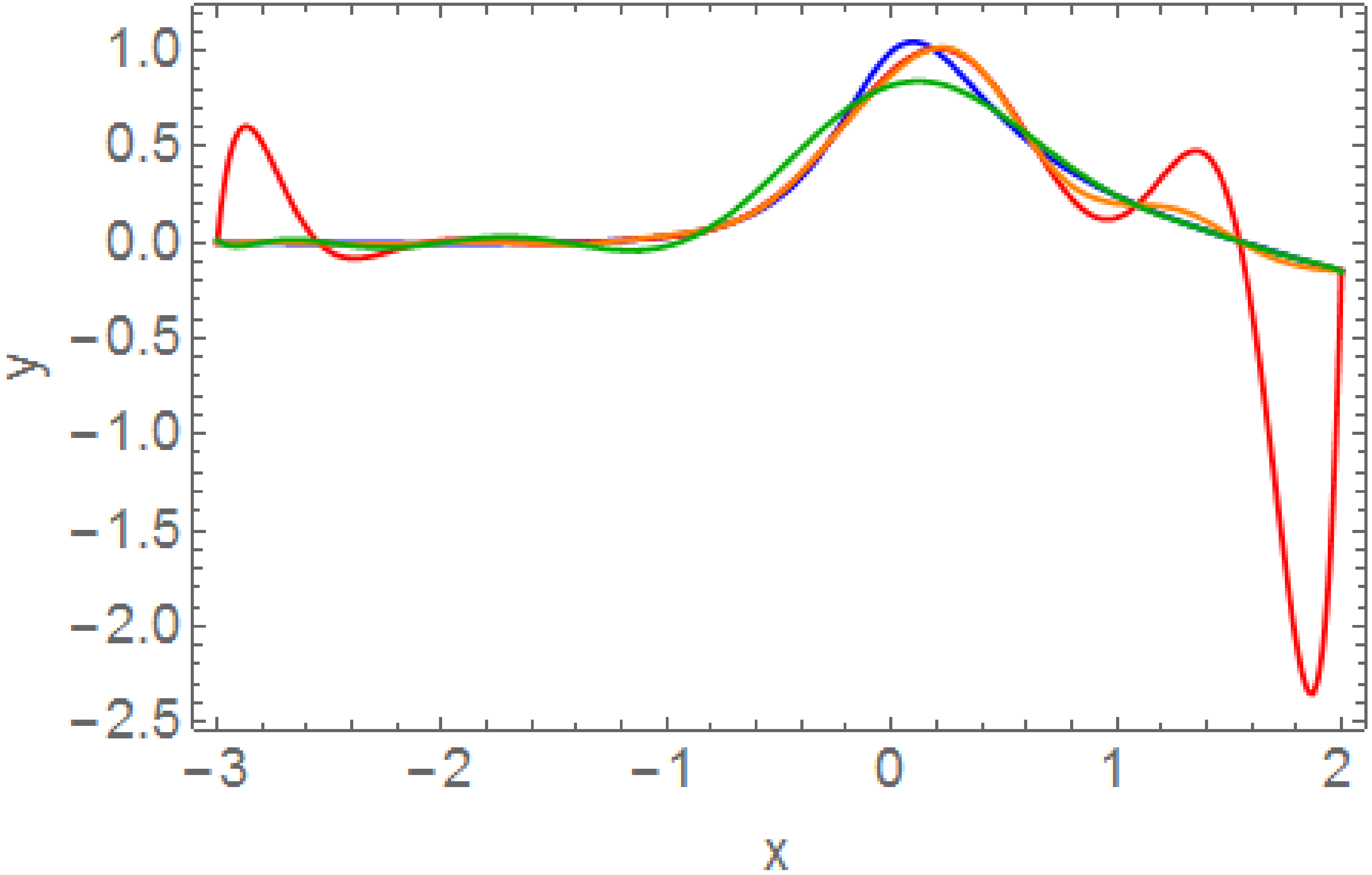
CHEBYSHEV POLYNOMIAL INTERPOLATION



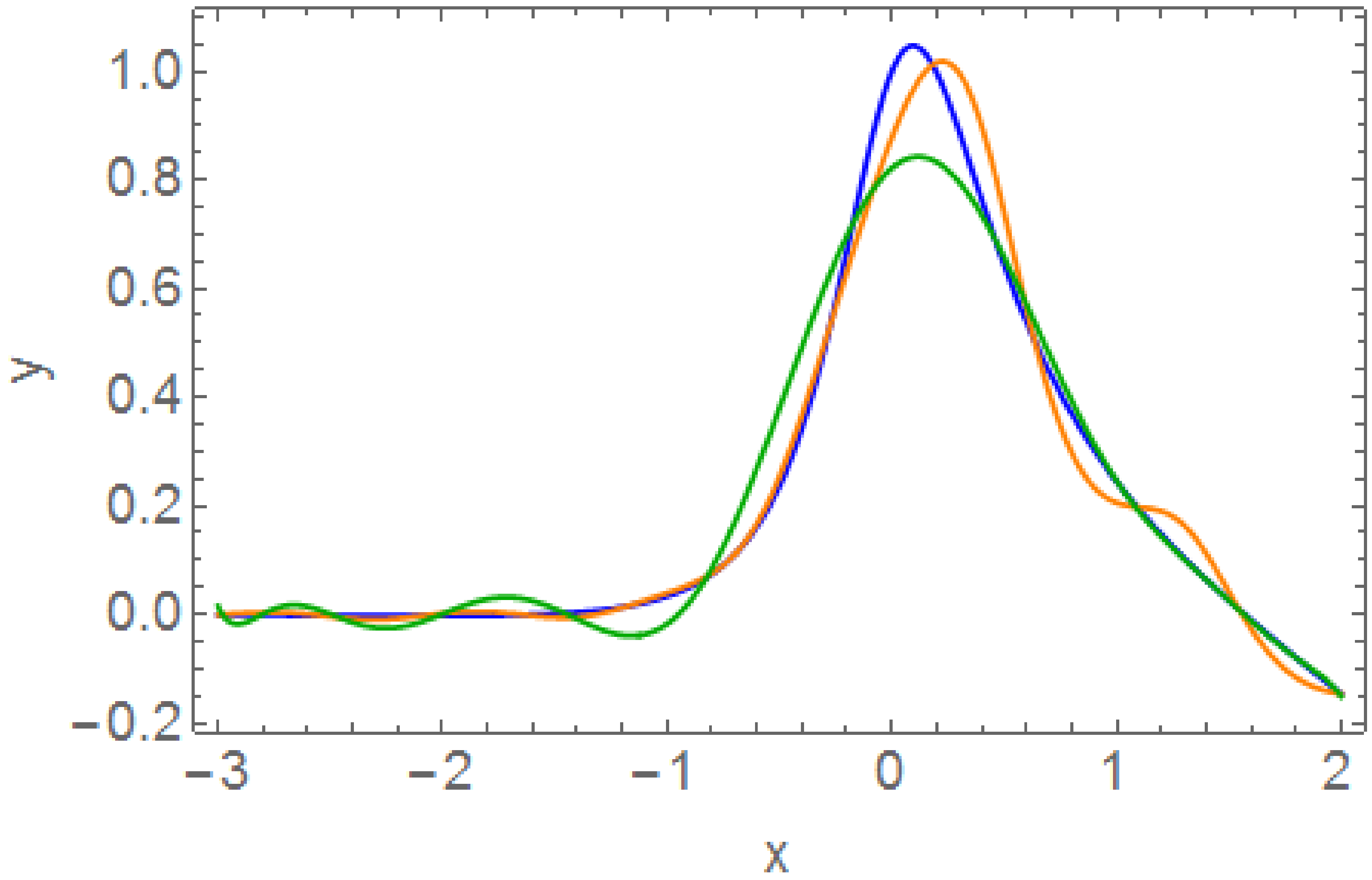
ERROR



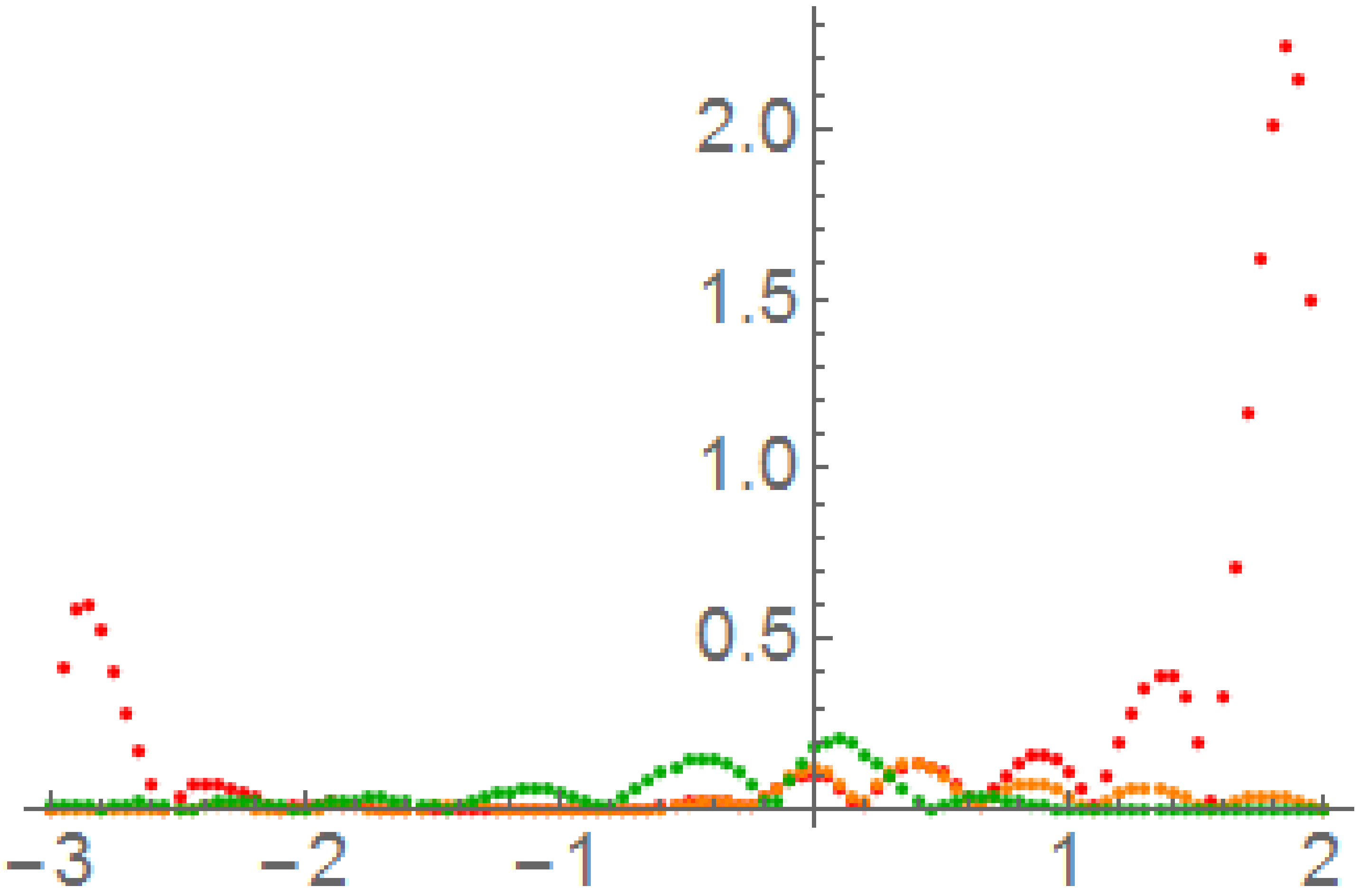
COMPARISON



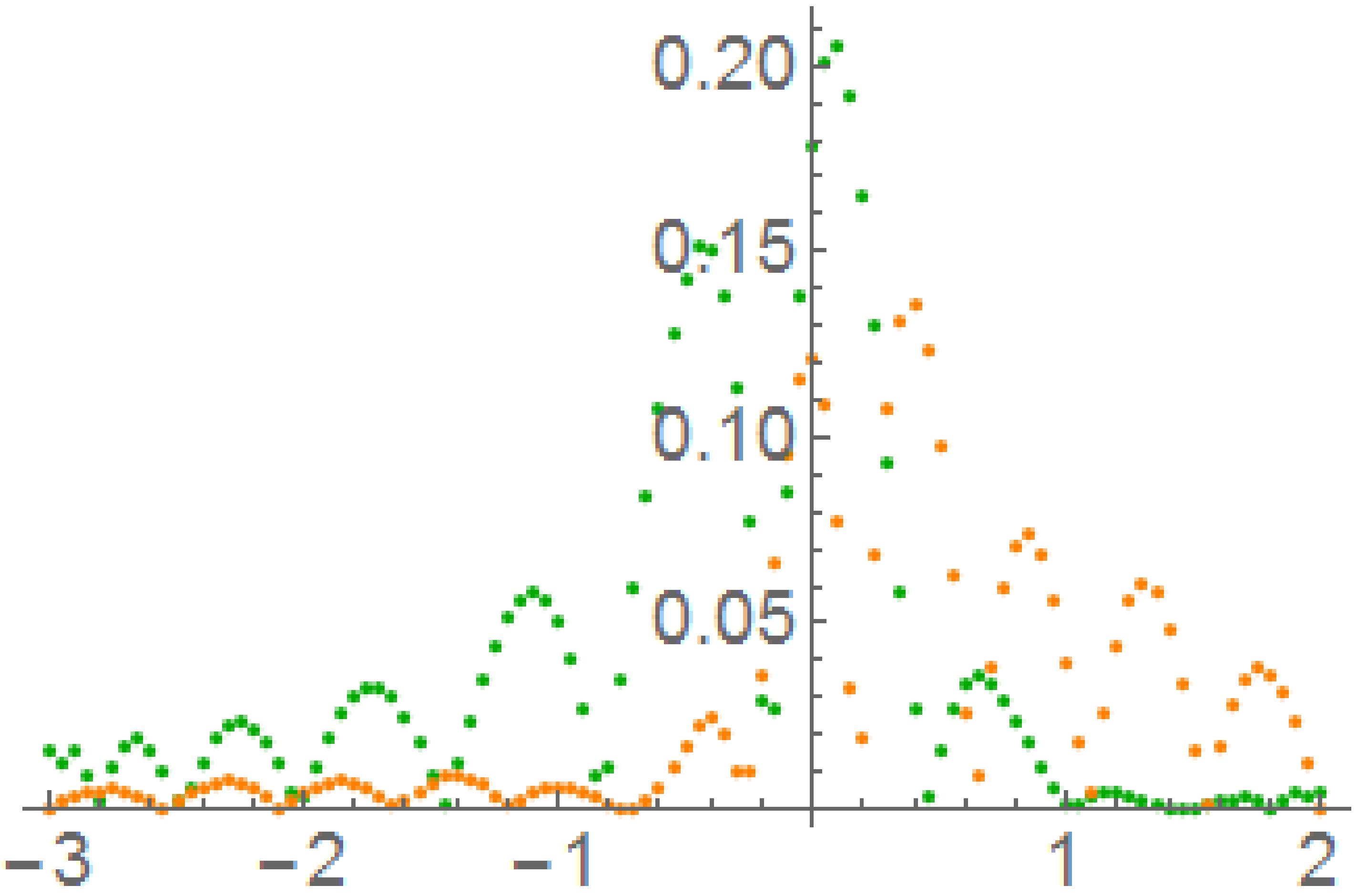
CLOSER LOOK



ERROR COMPARISON



CLOSER LOOK



OUTLINE

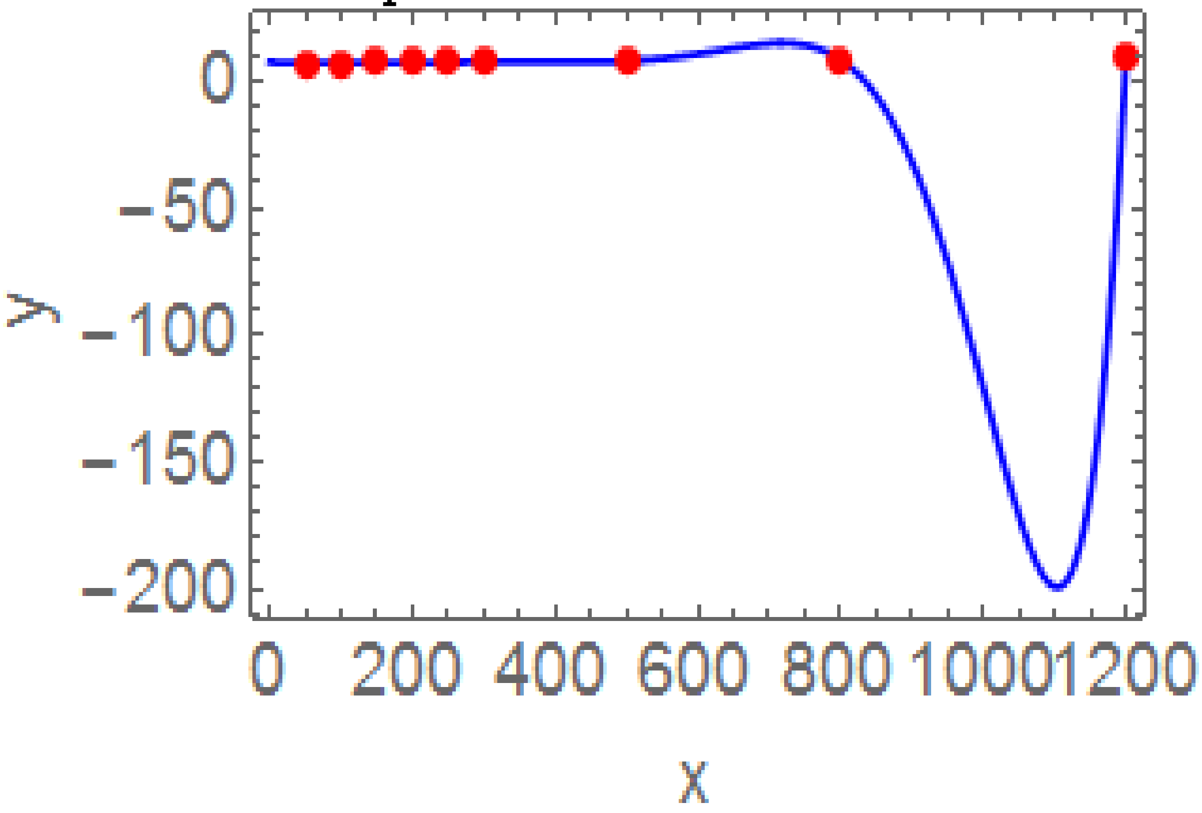
I. Overview
   I. Interest
   II. The Problem
II. Recall
   I. Lagrange Interpolation
   II. Newton Interpolation
   III. Shortcomings
III. Calculations
   I. Derivation
   II. Variation of Nodes
   III. Examples and Comparisons
IV. Conclusion
   I. Applications
   II. In the end

OTHER APPLICATIONS

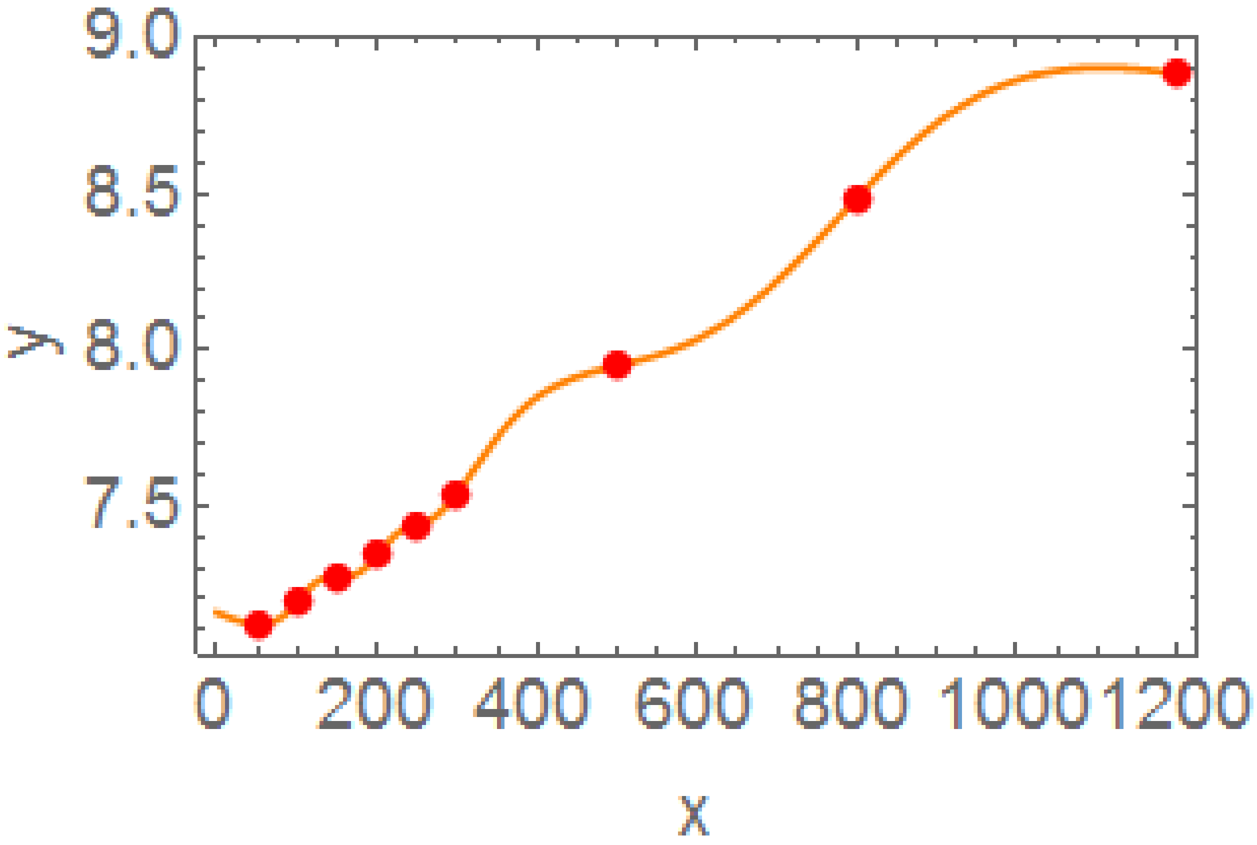
- Where f are Data Sets
- Differentiation of Polynomial Interpolants
- Fast Multipole Methods

DATA SETS PREVIEW

Without Barycentric Lagrange interpolation

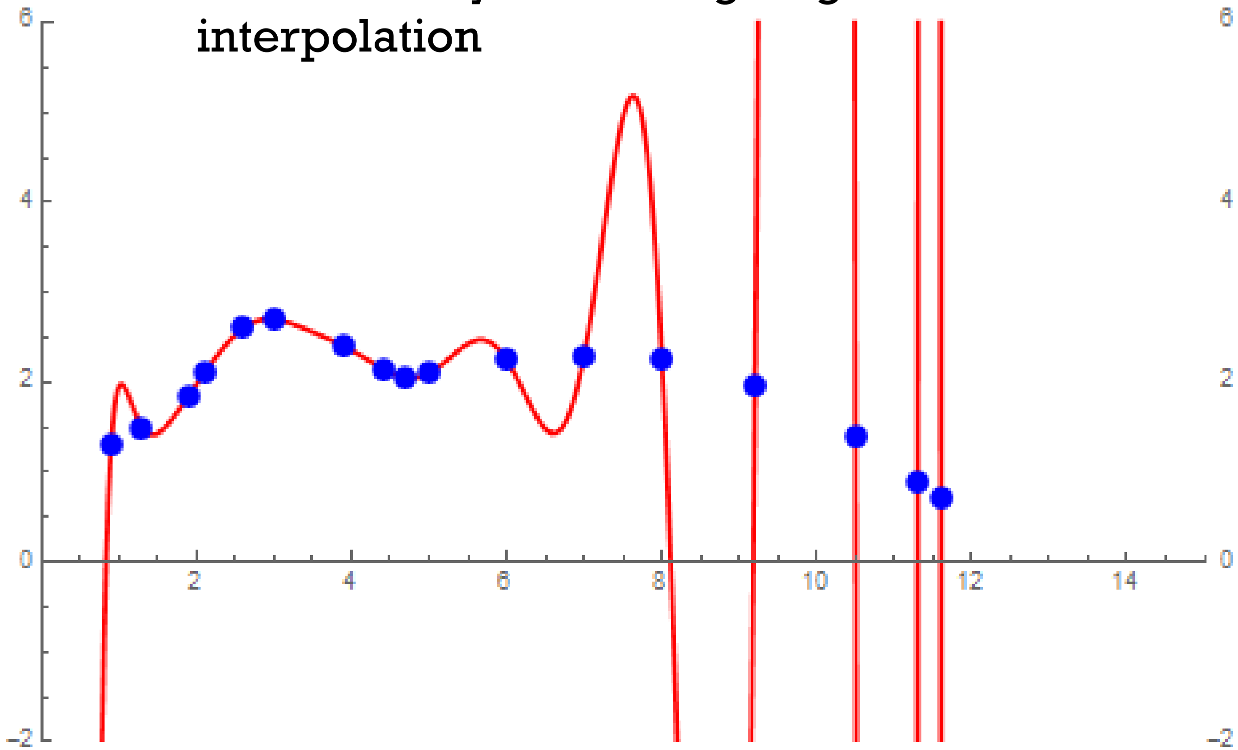


With Barycentric Lagrange Interpolation

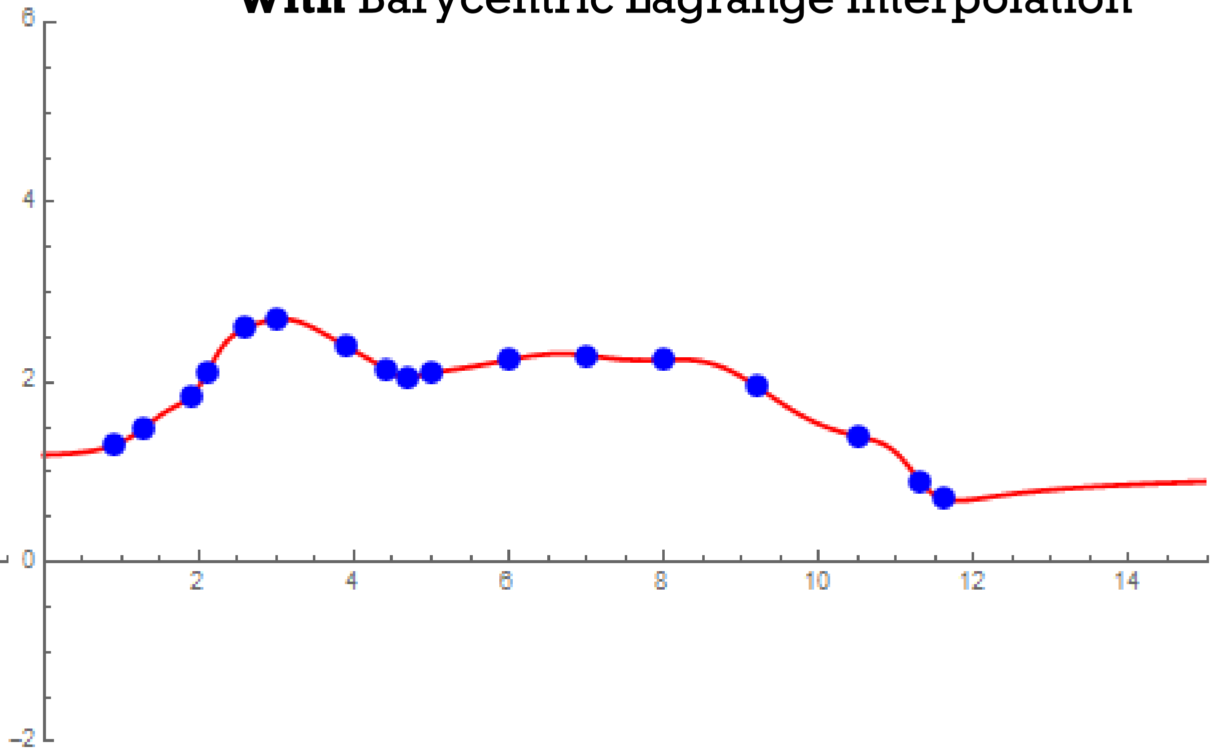


DATA SETS PREVIEW(CONT.)

Without Barycentric Lagrange interpolation



With Barycentric Lagrange Interpolation



CONCLUSION: IN THE END

- Barycentric Lagrange Interpolation is quicker in terms of flops and computational speed
- Barycentric Lagrange Interpolation is a more reliable tool of computing polynomial interpolants
- Barycentric Lagrange Interpolation should be the method of choice when dealing with interpolation

- Goal: To find an equation that is both quick and a trustable approximating tool? ✓

- "If you look in the index of a book of numerical analysis, you probably won't find 'barycentric.' Let us hope it will be different a generation from now."

QUESTIONS??

THANK YOU

REFERENCES

- Higham, Nicholas. J. "The Numerical Stability of Barycentric Lagrange Interpolation." *IMA Journal of Numerical Analysis (2004) 24, 547–556* (2004): n. pag. *Http://www.maths.manchester.ac.uk/~higham/narep/narep440.pdf*. IMA Journal of Numerical Analysis. Web. 11 Nov. 2016.
- Sauer, Timothy. "Chapter 3:Data and Interpolating Functions." *Numerical Analysis*. 2nd ed. Boston: Pearson, 2012. 139-66. Print.
- Trefethen, Lloyd N. *PDF*. N.p.: n.p., n.d. Approximation Theory and Approximation Practice. Web. 11 Nov. 2016.
- Trefethen, Lloyd N., and Jean-Paul Berrut. *PDF*. N.p.: 2004 Society for Industrial and Applied Mathematics, 2004. Web. 11 Nov. 2016.