PROJECT

submitted for the course

Numerical Algorithms

Author: Klaudia Woch

Note: This is an educational project developed as part of coursework at Częstochowa University of Technology. The university logo is used for informational and non-commercial purposes only.

Table of Contents

Introduction

Numerical algorithms are methods used to solve mathematical problems through computational techniques. They are particularly useful in cases where obtaining an analytical solution is difficult or impossible.

In the context of this project, numerical algorithms are applied for tasks such as interpolation (both polynomial and spline-based), data approximation, numerical integration, partial derivative computation, as well as statistical analysis.

The use of these methods enables the processing and interpretation of data stored in the format (x, y, F(x, y)), and facilitates the evaluation of how well mathematical models fit real-world measurement data.

Applied Methods and Result Analysis

Data Visualization in the (x, F(x, y)) Plane for Each Line y = const

Selected Method: Line Plot.

Method Description:

The visualization process begins with importing the input data from the file data.txt. The file contains values of variables in a three-column format, where:

- the first column (x) represents the horizontal coordinate,
- the second column (y) represents the vertical coordinate,
- and the third column (F(x, y)) denotes the function value.

Example row:

```
-10.00000  -10.00000  -293.73141
```

The data is then filtered to extract unique y-values. Following this step, a line plot is generated for each unique y-value, illustrating the relationship between x and F(x, y).

Results:
As a result, a set of line plots was obtained, each showing the dependency of F(x, y) for a specific y = const line.

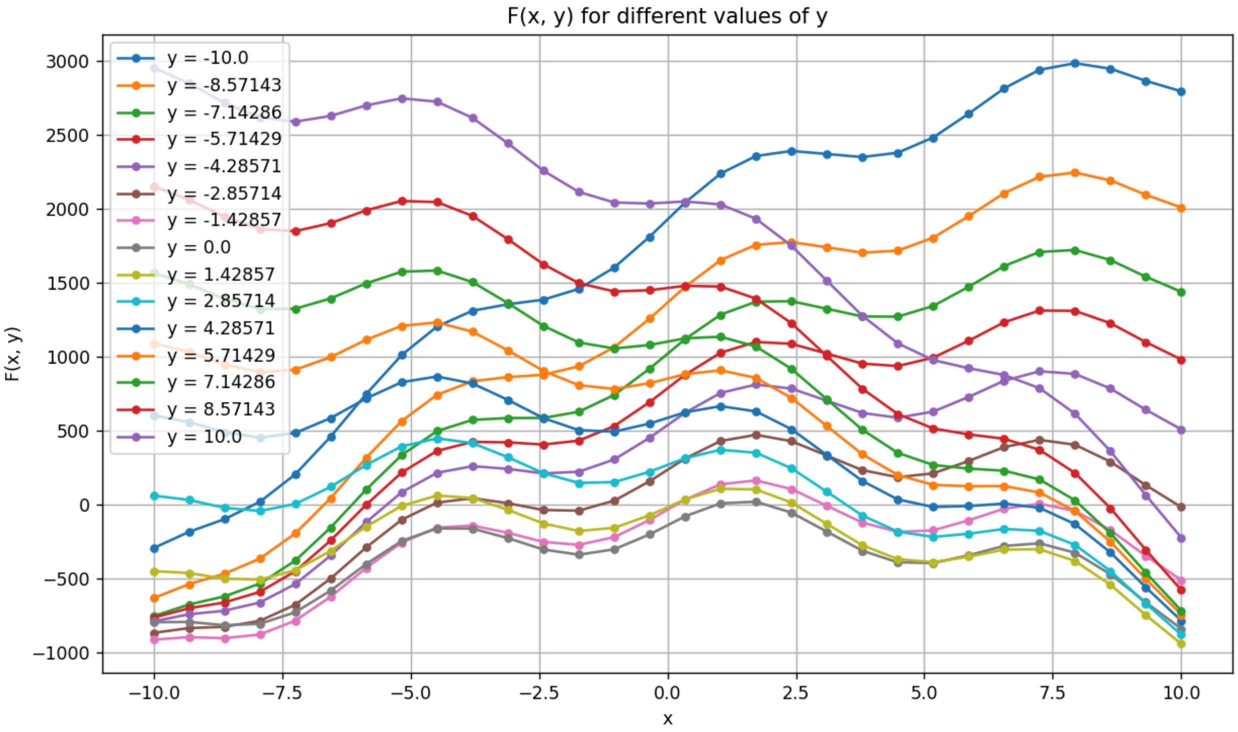

Figure 1. Line plot for each line y = const.

Computation of Statistics

Selected Methods: Mean, median, standard deviation; bar chart visualization.

Method Description:

The listed statistical methods were applied to compute the mean, median, and standard deviation of function values across defined y-intervals. To facilitate interpretation of the results, a bar chart was used. The horizontal axis represents the y values, while the vertical axis displays the corresponding computed statistics.

Results:

The bar chart enables a direct visual comparison of statistical values across different y-intervals. Based on the chart, the interval at y = 0.0 was selected for further analysis. In this region, both the mean and median are close to zero - these are the lowest values among all y intervals - indicating a highly symmetric distribution centered around zero. Additionally, the standard deviation is relatively low, suggesting minimal data variability within this group.

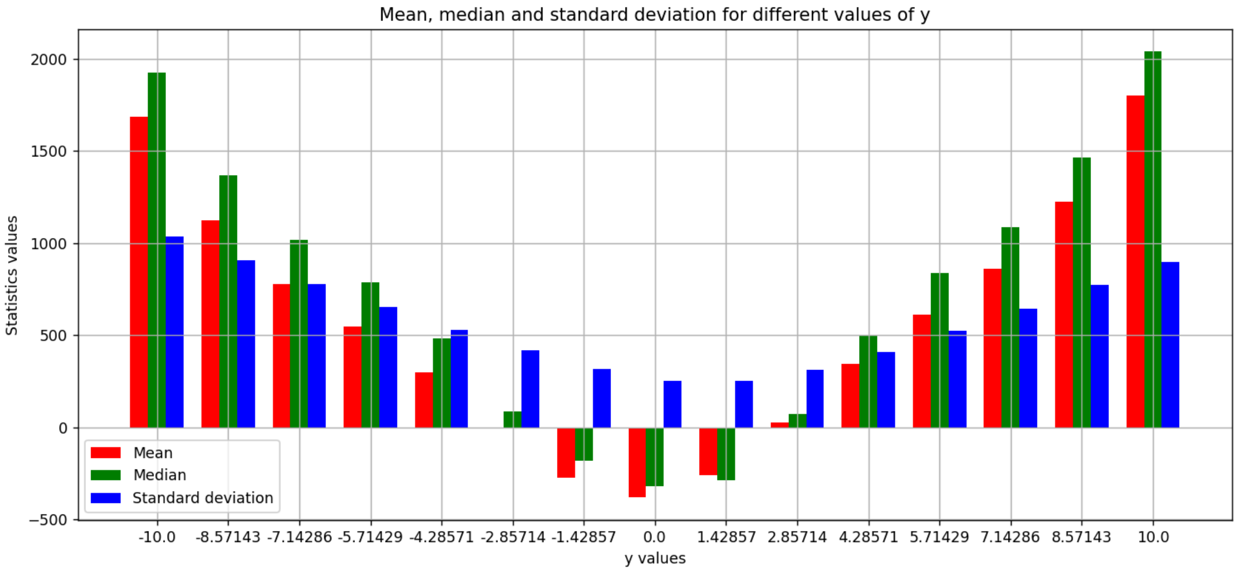

Figure 2. Visualization of the calculated statistical measures.

Polynomial Interpolation

Selected Methods: Mean, Median, Standard Deviation; Bar Plot.

Method Description:
The selected statistical methods compute the mean, median, and standard deviation of the function values across different y-intervals. To visualize the results, a bar plot is used, where the horizontal axis represents y-values, and the vertical axis shows the corresponding computed statistics.

Results:
The bar plot provides a clear comparison of the statistical values. Based on the visualized results, the value y = 0.0 was selected for further analysis. For this specific y, both the mean and the median are very close to zero - these are the lowest among all y-values, indicating a highly

symmetric distribution centered around zero. Furthermore, the standard deviation is relatively low, suggesting the smallest variability within this data group.

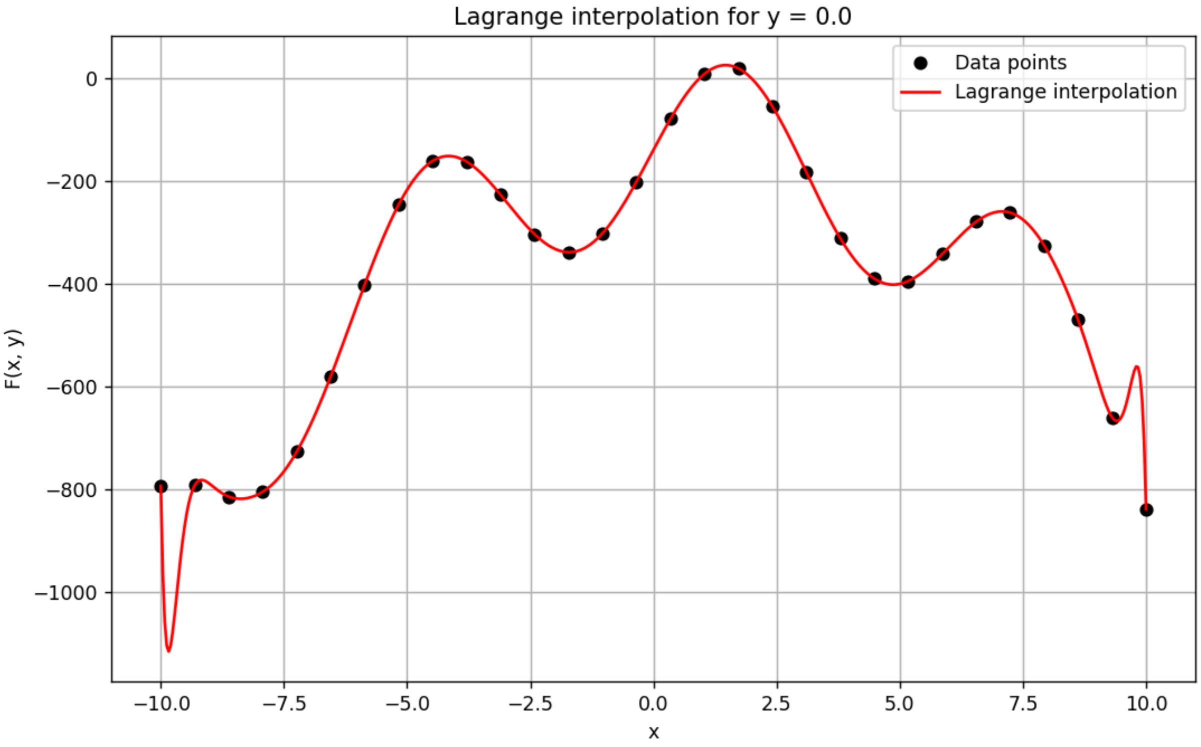

Figure 3. Plot of Lagrange polynomial interpolation.

Spline Interpolation

Selected Methods: Cubic Spline Interpolation, Line Plot

Description of Methods:

Cubic spline interpolation involves fitting a piecewise function composed of third-degree polynomials between successive data points. Each segment is represented by its own cubic polynomial, and the algorithm ensures the continuity of the function as well as its first and second derivatives across the entire domain. The coefficients of these polynomials are computed by solving a system of linear equations derived from smoothness constraints and interpolation conditions, along with chosen boundary conditions (e.g., natural or clamped splines). This method provides smooth transitions between points while maintaining high accuracy in representing the underlying signal.

Results:

The line plot demonstrates how the cubic spline produces a smooth, continuous curve that passes through all data points, effectively capturing the signal's variability without introducing sharp discontinuities or artifacts.

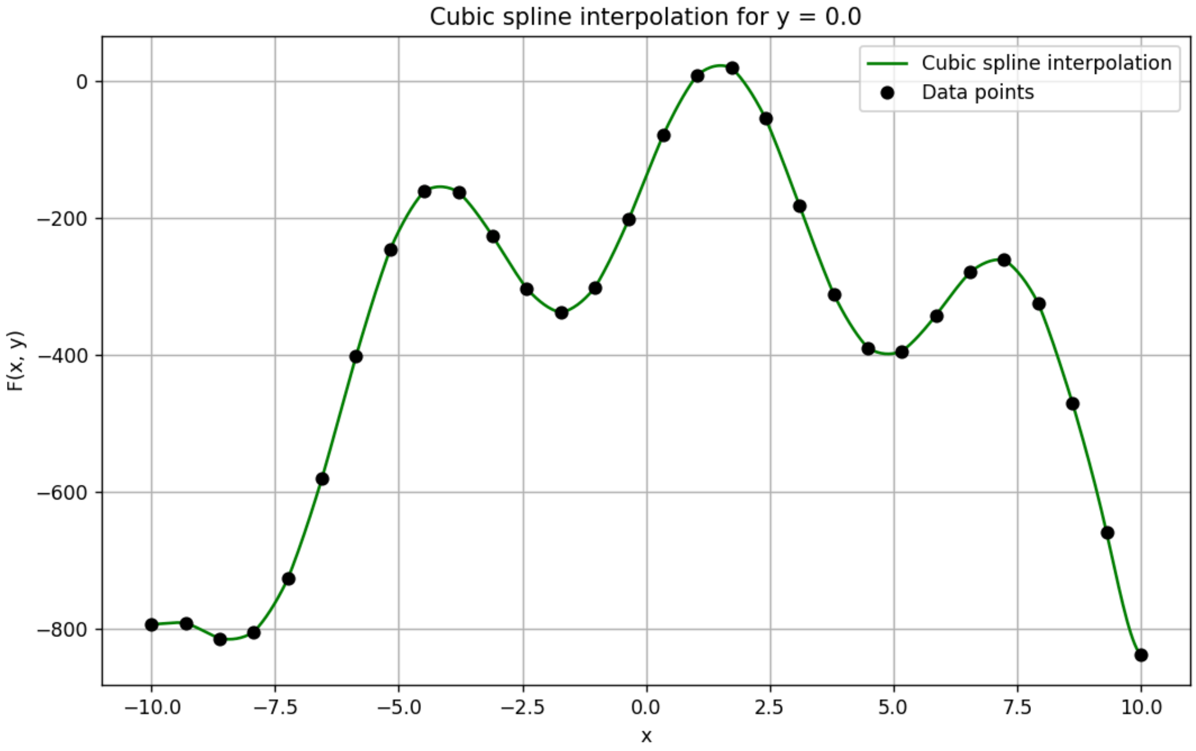

Figure 4. Plot of cubic spline interpolation.

Comparison of Interpolation Functions

Results:

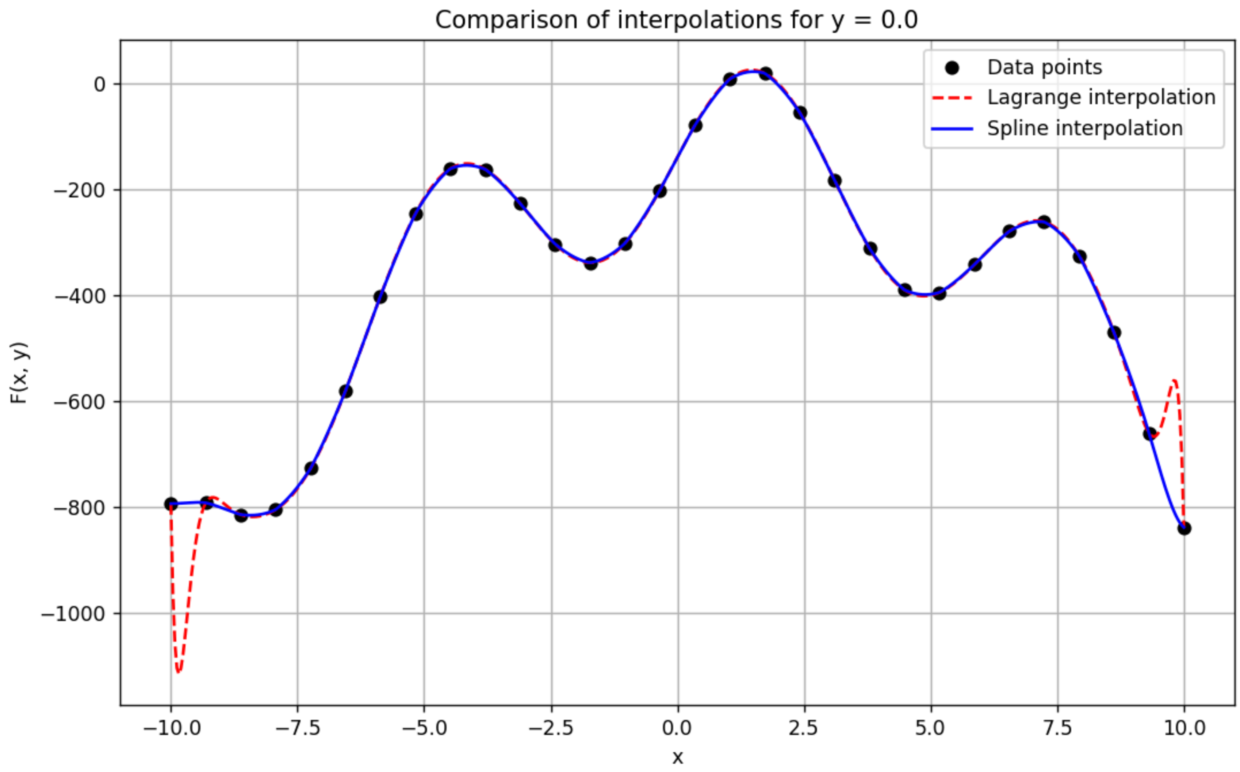

Figure 5. Comparison of Lagrange polynomial interpolation and cubic spline interpolation.

Selected Methods: Error Metrics Calculation (MAE, MSE, MAX) for Both Interpolation Types, Line Plot, Bar Chart

Description of Methods:

Error metrics are computed by comparing the interpolated values with the actual values of the function on a point-by-point basis. The calculated metrics include:

- MAE (Mean Absolute Error): the average of absolute differences,
- MSE (Mean Squared Error): the average of squared differences,
- MAX Error: the largest single difference between interpolated and actual values.

These metrics assess the accuracy and stability of interpolation methods. They allow for quick identification of which technique (e.g., Lagrange or cubic spline) more accurately reconstructs the original dataset.

Conclusions:

Polynomial interpolation using the Lagrange method may lead to significant errors, especially when the number of equally spaced nodes increases - this is due to the Runge phenomenon, which causes oscillations at the interval boundaries. Cubic splines, by contrast, avoid this issue due to their local nature and segmented structure. Increasing the number of nodes in spline interpolation generally improves accuracy without distorting the shape of the curve. Overall, cubic splines provide a more stable and accurate representation of the function, particularly over larger intervals and for denser datasets. This is confirmed by lower MAE, MSE, and MAX error values for spline interpolation.

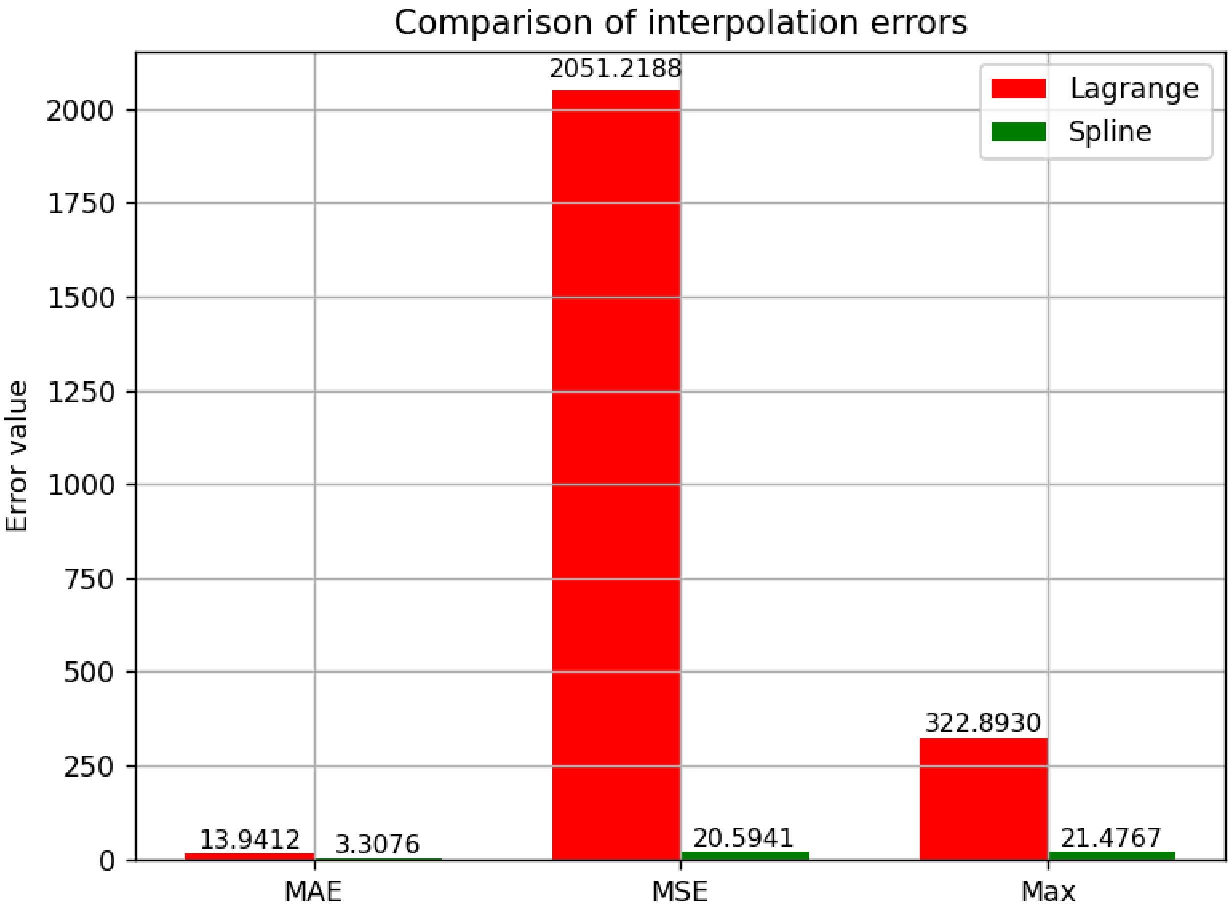

Figure 5. Comparison of Lagrange polynomial interpolation and cubic spline interpolation.

Function Approximation

Selected Methods: Linear and Quadratic Least Squares Approximation, Line Plot, Bar Chart

Description of Methods:

Linear approximation fits a line of the form $f(x) = ax + b$ by minimizing the sum of squared differences between the data points and the approximation. The coefficients a and b are obtained using formulas derived from the normal equations.

Quadratic approximation extends this approach to a second-degree polynomial $f(x) = ax^2 + bx + c$, fitting a parabola to the data. The coefficients a, b, and c are determined by solving a system of three equations based on sums of powers and products of the data. The goal of both methods is to approximate the dataset as accurately as possible using a function with a limited number of parameters.

Results:

The line plot for the linear approximation shows a best-fit line that captures the general trend of the data but fails to reflect local variations in the function’s behavior.

The quadratic approximation plot illustrates how a second-degree polynomial provides a better fit for nonlinear relationships within the data, resulting in a smoothly curved line. The additional parameters allow for a more precise representation of the function’s shape within the analyzed domain.

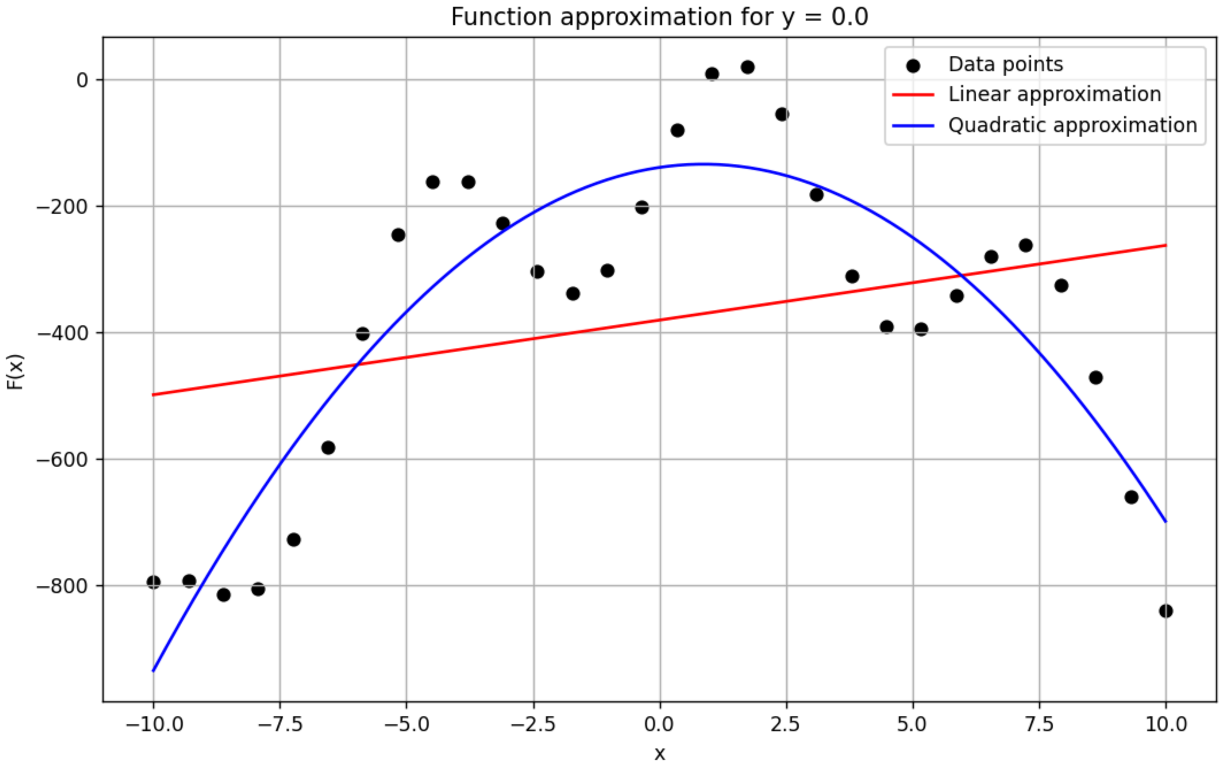

Figure 7. Comparison of linear and quadratic approximation with original data points.

Error Analysis:

The computed error metrics confirm that quadratic approximation yields a more accurate fit. This is evidenced by a lower RMSE (Root Mean Square Error) value and an R^2 (coefficient of determination) value closer to 1, indicating a higher degree of accuracy in representing the underlying data.

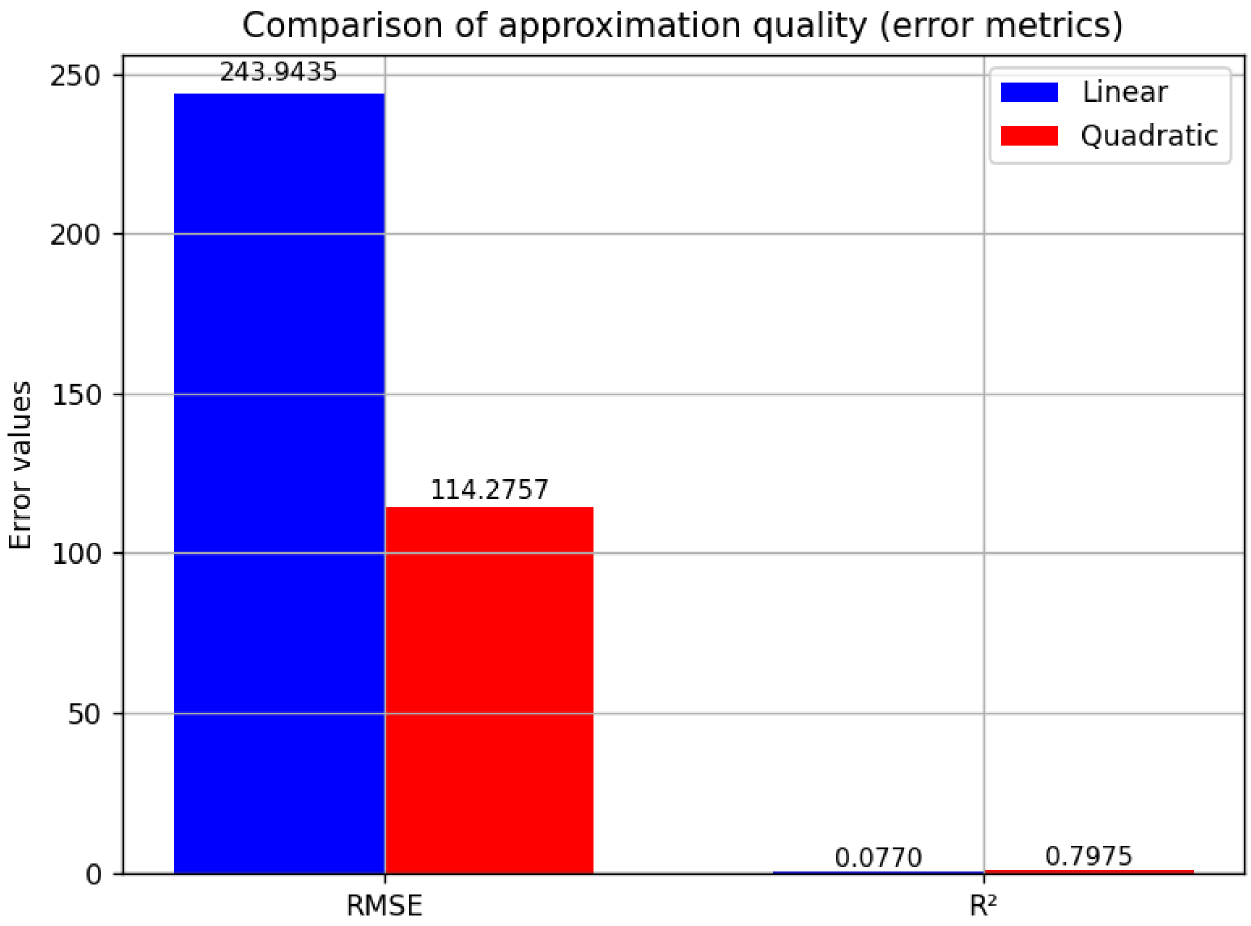

Figure 8. Quality comparison of linear and quadratic approximation.

Numerical Integration

Selected Methods: Trapezoidal Rule, Simpson’s Rule, Bar Chart, Line Plot.

Description of Methods:

The trapezoidal rule is a basic numerical integration technique that approximates the area under a function’s curve by dividing it into small intervals and treating each segment as a trapezoid. The area of each trapezoid is calculated and summed to estimate the integral. This method performs best when the function is smooth and the sample points are evenly spaced.

Simpson’s rule was applied to compare the accuracy of different integration methods. Unlike the trapezoidal rule, Simpson’s method approximates the curve using parabolic arcs rather than straight-line segments, which better capture the curvature of the function. In practice, the domain is divided into an even number of intervals, and for every three consecutive points, a quadratic polynomial is fitted. This leads to better approximation, especially for smooth functions that can be well represented by polynomials. As a result, Simpson's rule typically yields more accurate integration with reduced error.

Results:

The bar chart displays the computed integral values for various approximations: Lagrange interpolation, cubic spline interpolation, linear approximation, and quadratic approximation.

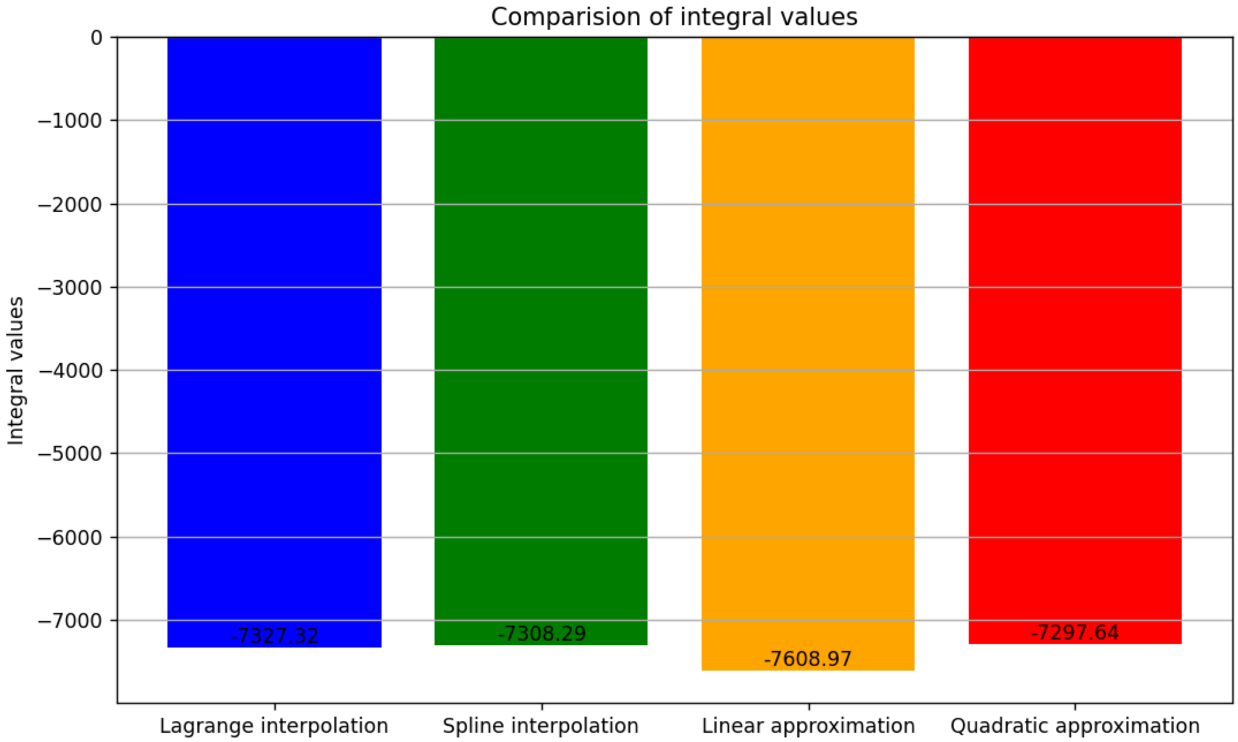

Figure 9. Comparison of integral values (Lagrange and spline interpolation, linear and quadratic approximation).

Error Analysis:

The accuracy of the trapezoidal rule and Simpson's rule was compared for the quadratic approximation of a function over the interval [a, b], using varying numbers of evaluation points N, corresponding to different step sizes. The results show that both methods converge toward the true value of the integral as the number of points increases (i.e., as the step size decreases). However, Simpson's rule exhibits significantly faster convergence and yields lower error even for small N, due to its higher order of accuracy.

The plotted results indicate that the integration step size has a considerable impact on accuracy - smaller steps lead to more precise results. Notably, Simpson's method achieves stable integral values with fewer points, making it more efficient for smooth functions.

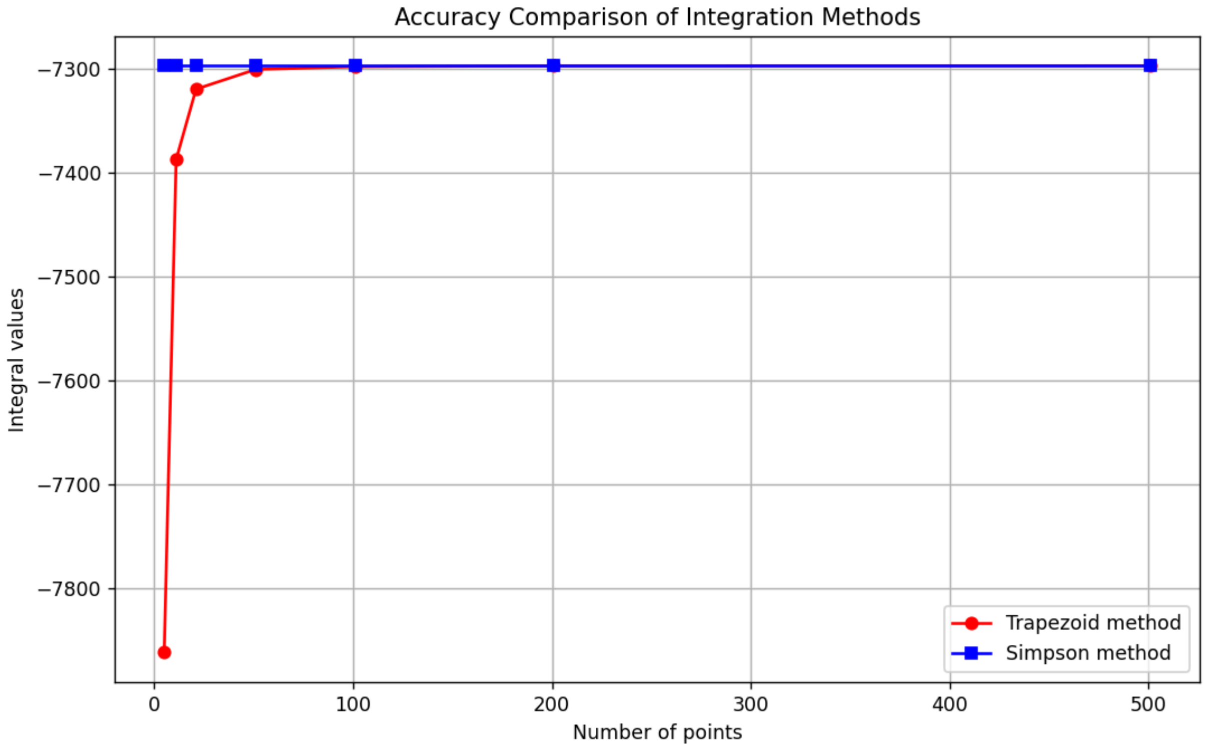

Figure 10. Accuracy comparison between the trapezoidal rule and Simpson's rule.

Calculation of Partial Derivatives

Selected Methods: Central Difference Method, Line Plot, Bar Chart

Description of Methods:

The central difference method is used for approximating the derivative of a function based on a finite set of sampled points. For interior points, the derivative at x_i is approximated using the following formula:

$\frac{f(x_{i+1})-f(x_{i-1})}{x_{i+1}-x_{i-1}}$. This approach offers higher accuracy than one-sided methods (forward or backward differences). At the boundaries of the interval, where adjacent points on one side are missing, forward or backward difference formulas are applied instead. The method assumes that the data points are approximately evenly spaced.

Results:

The line plot illustrates the variation of the function F(x, y) with respect to x, for a fixed value of y. A positive derivative indicates increasing behavior, while a negative derivative suggests a decreasing trend. For y = 0.0, noticeable sign changes in the derivative indicate the presence of local extrema or inflection points in the function.

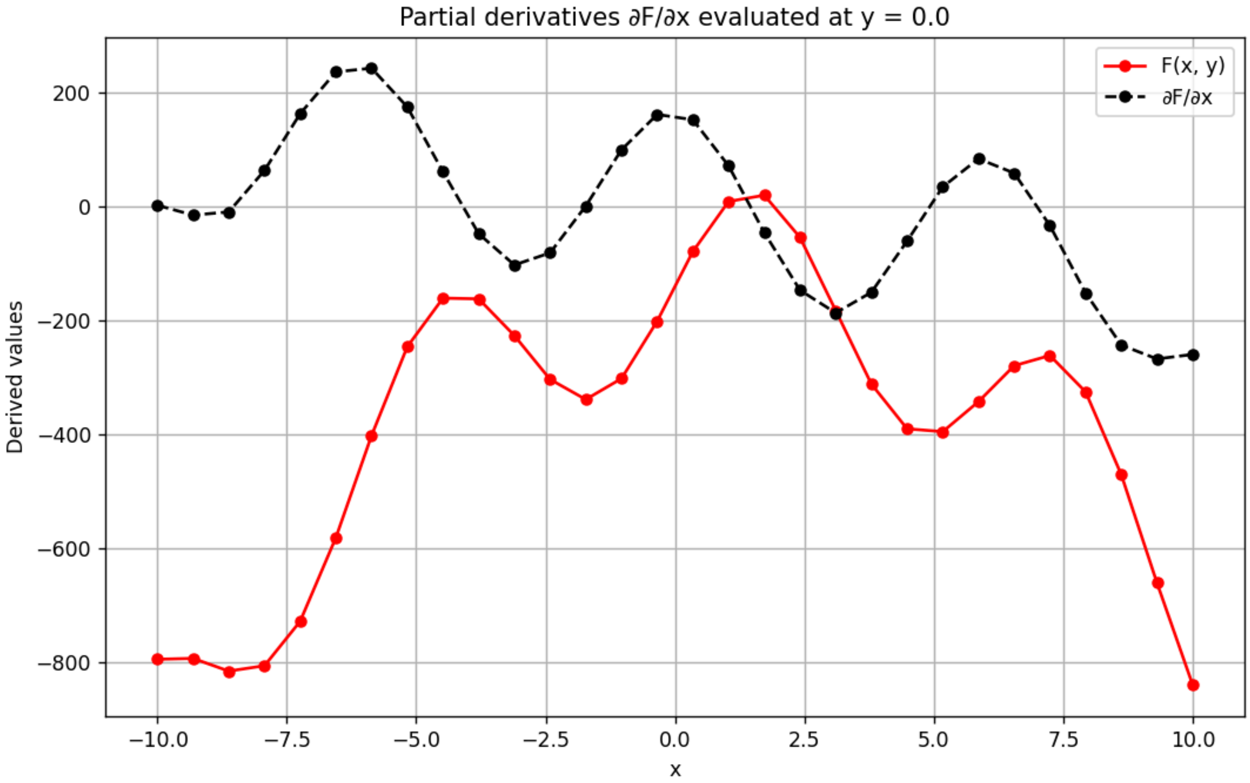

Figure 11. Plot of partial derivatives.

Error Analysis:

Choosing an appropriate step size h is a trade-off between discretization accuracy and numerical stability. The central difference method is effective for uniformly spaced data and a moderate number of points. As the data becomes denser (i.e., smaller h), the approximation error typically decreases - until numerical errors due to floating-point limitations begin to dominate. This behavior is illustrated in the accompanying bar chart, which highlights how error evolves with respect to step size.

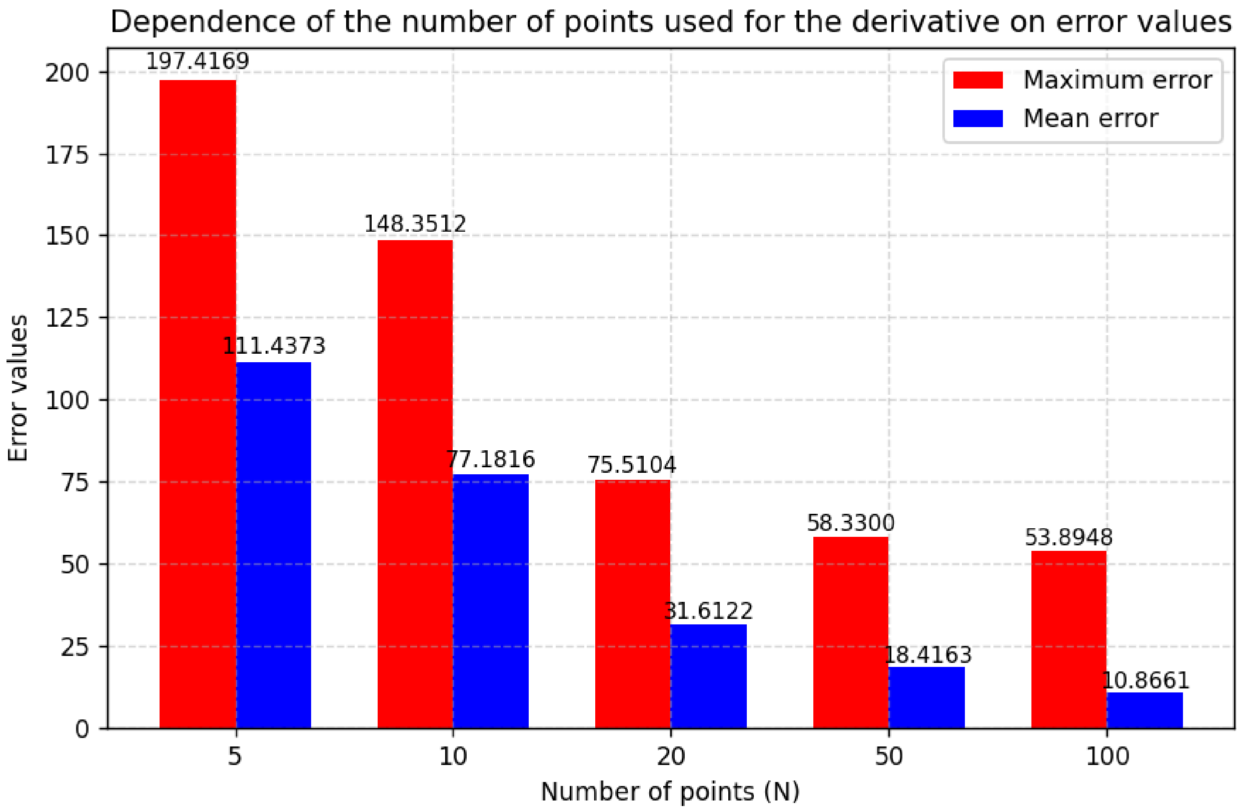

Figure 12. Plot illustrating the monotonicity of the function.

Determining Monotonicity

Selected Method: Line Plot

Description of the Method:

The line plot visualizes the monotonicity of the function. Pink segments indicate intervals where the function is increasing, blue segments correspond to decreasing intervals, and gray segments represent intervals where the function remains constant.

Results:

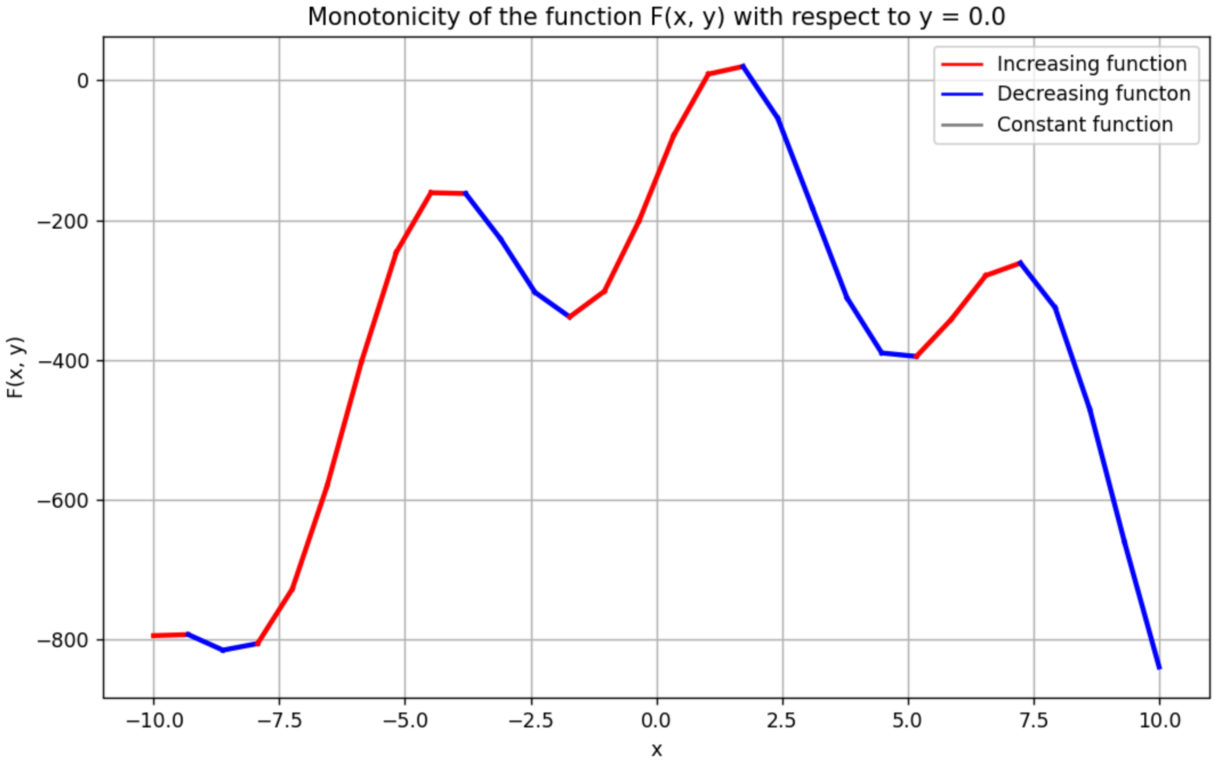

Figure 13. Plot illustrating the monotonicity of the function.

Final Conclusions

A range of classical numerical methods was implemented in this project, and their performance was evaluated in terms of both accuracy and computational stability. The key observations are summarized below:

- **Lagrange Polynomial Interpolation**
  Lagrange interpolation performs well with a small number of data points; however, as the number of nodes increases, significant errors emerge due to Runge's phenomenon - particularly near the endpoints of the interval. This method exhibits a global character and is prone to oscillations, which limits its applicability for larger datasets.
- **Cubic Spline Interpolation**
  Cubic spline interpolation offers superior stability and accuracy, especially for larger

datasets. Due to its local nature, it avoids overfitting and produces a smooth representation of the function. The MAE, MSE, and MAX error metrics were consistently lower compared to Lagrange interpolation, confirming its higher effectiveness in practical scenarios.

- **Least Squares Approximation**
  Least squares fitting - both linear and quadratic - was employed to approximate the overall trend of the function. The quadratic model provided a better representation of the data's nonlinear behavior, as indicated by lower RMSE values and a higher coefficient of determination (R^2).
- **Numerical Integration: Trapezoidal vs. Simpson’s Rule**
  The trapezoidal rule provides an approximate solution for definite integrals but has limited accuracy, particularly for functions with variable curvature. In contrast, Simpson’s rule - based on parabolic approximation- more accurately captures the curvature of the function, resulting in lower numerical integration errors and overall improved precision.
- **Central Difference Method for Partial Derivatives**
  The central difference method was effective and accurate for computing partial derivatives, assuming an appropriate choice of step size (h). Error analysis confirmed the method’s convergence as the number of points increased, validating its reliability for derivative estimation.
- **Monotonicity Analysis**
  Monotonicity analysis enabled an intuitive visualization of directional changes in the function. A predefined threshold criterion was applied to prevent insignificant fluctuations from being misclassified as increases or decreases. As a result, the plots more accurately reflect meaningful behavioral changes in the function, rather than momentary or negligible variations.

References

1. **Ewa Majchrzak, Bohdan Mochnacki**
   Numerical Methods: Theoretical Foundations, Practical Aspects, and Algorithms
   Wydawnictwo Politechniki Śląskiej, Gliwice, 2004.
   (Original Polish title: Metody numeryczne. Podstawy teoretyczne, aspekty praktyczne i algorytmy)

2. **Richard L. Burden, J. Douglas Faires**
   Numerical Analysis, 10th Edition
   Cengage Learning, 2015.

(One of the most widely used textbooks on numerical methods in academia.)

3. **Steven C. Chapra, Raymond P. Canale**
   Numerical Methods for Engineers, 7th Edition
   McGraw-Hill Education, 2015.
   (Great practical emphasis; useful for applied problems in interpolation, approximation, and integration.)

Online resources

- NumPy Documentation: numpy.org
- Matplotlib Documentation: matplotlib.org