MA2252 Introduction to computing

lectures 23-24

Interpolation

Matias Ruiz

November 2023

Interpolation Problem Statement

Suppose we have a data set containing n data points $(x_i, y_i), i = 1, 2, \cdots, n$.

Goal: To find an estimation function $\hat{y}(x)$ with domain $x \in [x_1, x_n]$ such that $\hat{y}(x_i) = y_i$.

The function $\hat{y}(x)$ is called **interpolation function**.

Note: The choice of interpolation function depends on other factors such as accuracy, underlying physics etc. Depending on this choice, there are different interpolation methods.

Some Interpolation methods

- Linear Interpolation
- Cubic Spline Interpolation
- Lagrange Polynomial Interpolation

Linear interpolation

Here, the interpolation function $\hat{y}(x)$ is defined piecewise by linear polynomials (or straight lines). So,

$$\hat{y}_i(x) = y_i + \frac{(y_{i+1} - y_i)(x - x_i)}{x_{i+1} - x_i}, \quad x_i \leq x \leq x_{i+1} \quad (i = 1, 2, \cdots, n-1) \tag{1}$$

MATLAB's interp1() function can be used to make life easier.

Cubic spline interpolation

A cubic spline is a function defined piecewise by cubic polynomials.

In cubic spline interpolation, the interpolating function is a cubic spline defined as

$$S_i(x) = a_i x^3 + b_i x^2 + c_i x + d_i, \quad x_i \leq x \leq x_{i+1} \quad (i = 1, 2, \cdots, n) \quad (2)$$

Again, MATLAB's interp1 function can be used by giving 'cubic' as argument.

Cubic spline interpolation (contd.)

The unknown parameters a_i, b_i, c_i and d_i are found using these conditions:

$$S_i(x_i) = y_i \quad (i = 1, 2, \cdots, n-1) \tag{3a}$$

$$S_i(x_{i+1}) = y_{i+1} \quad (i = 1, 2, \cdots, n-1) \tag{3b}$$

$$S_i'(x_{i+1}) = S_{i+1}'(x_{i+1}) \quad (i = 1, 2, \cdots, n-2) \tag{3c}$$

$$S_i''(x_{i+1}) = S_{i+1}''(x_{i+1}) \quad (i = 1, 2, \cdots, n-2) \tag{3d}$$

$$S_1''(x_1) = 0 \tag{3e}$$

$$S_{n-1}''(x_n) = 0 \tag{3f}$$

Lagrange Polynomial Interpolation

Here, the interpolation function is a **Lagrange polynomial** defined as

$$L(x) = \sum_{i=1}^{n} y_i P_i(x) \tag{4}$$

where

$$P_i(x) = \prod_{j=1, j \neq i}^{n} \frac{x - x_j}{x_i - x_j}. \tag{5}$$

The Lagrange polynomial $L(x)$ of degree $n-1$ passes through n data points i.e. it satisfies $L(x_i) = y_i \quad (i = 1, 2, \cdots, n)$.