Spline Interpolation

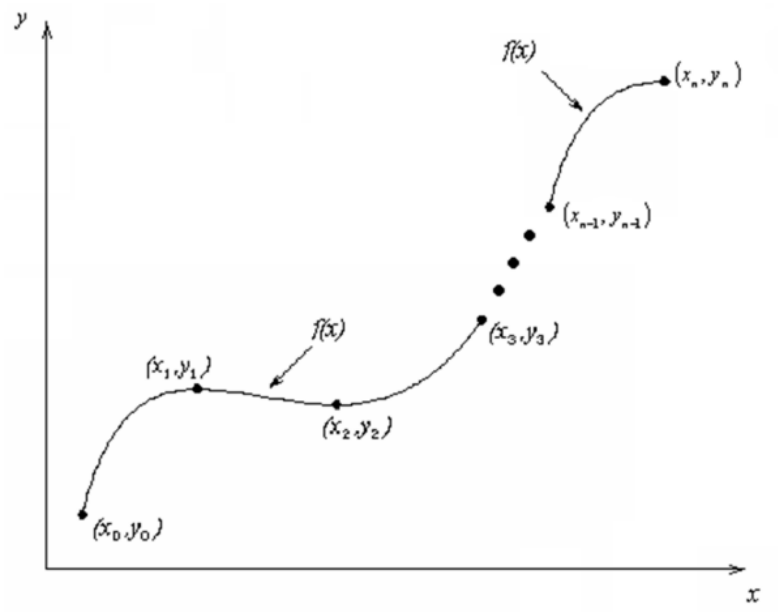
FOUNDATION OF NUMERICAL ANALYSIS

DR MOSTAFA SHAMSI

KIARASH ESMAEILBEIGI

What is Interpolation ?

Given (x0,y0), (x1,y1), …… (xn,yn), find the value of ‘y’ at a value of ‘x’ that is not given.



Most Common Interpolant

Polynomial Interpolation is the way of fitting the curve by creating a higher degree polynomial to join those points.

Polynomials are the most common choice of interpolants because they are easy to:

- Evaluate
- Differentiate
- Integrate

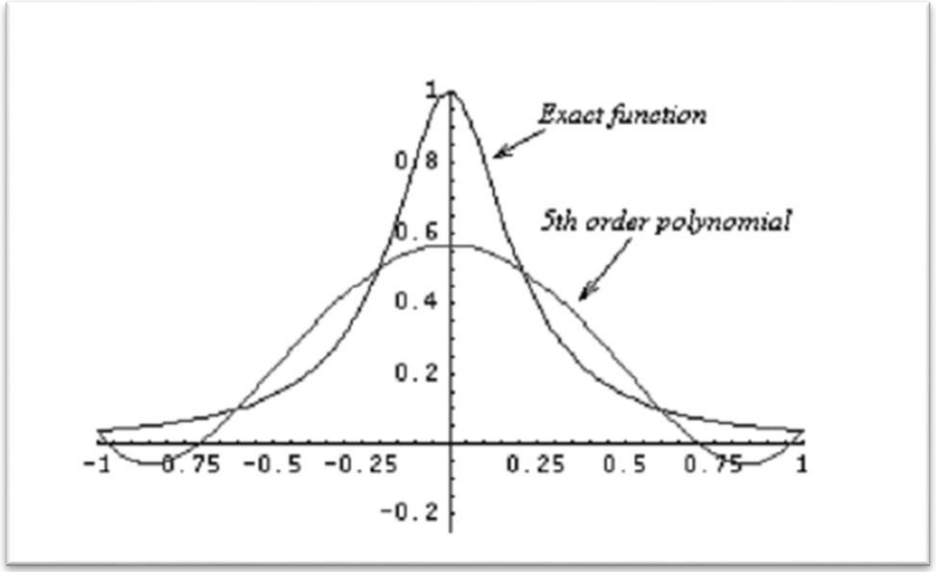
Why Splines ?

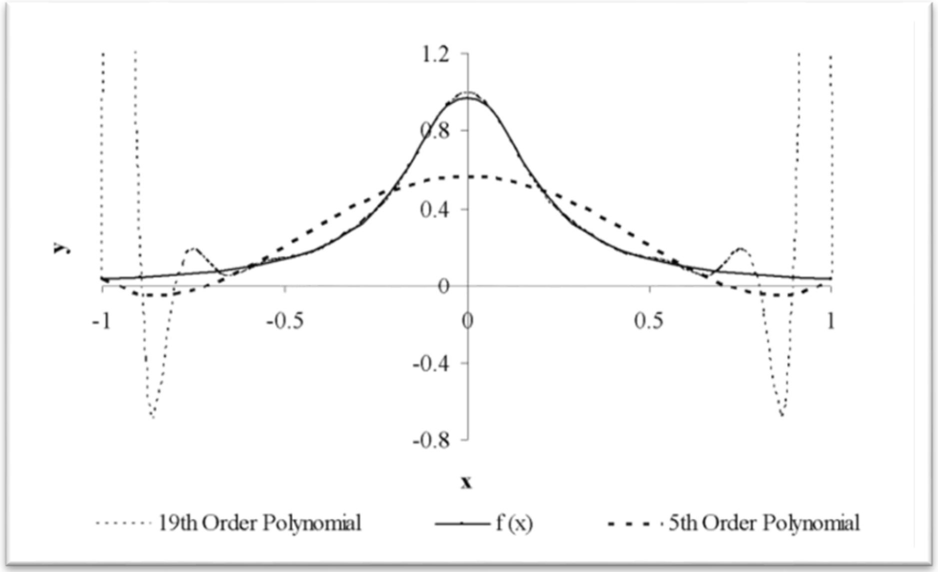
$$f(x) = \frac{1}{1+25x^2}$$

x	$y = \frac{1}{1+25x^2}$
-1.0	0.038461
-0.6	0.1
-0.2	0.5
0.2	0.5
0.6	0.1
1.0	0.038461

Six equidistantly spaced points in [-1, 1]

5^{th} and 19^{th} order polynomials vs. exact function





Spline Interpolation:

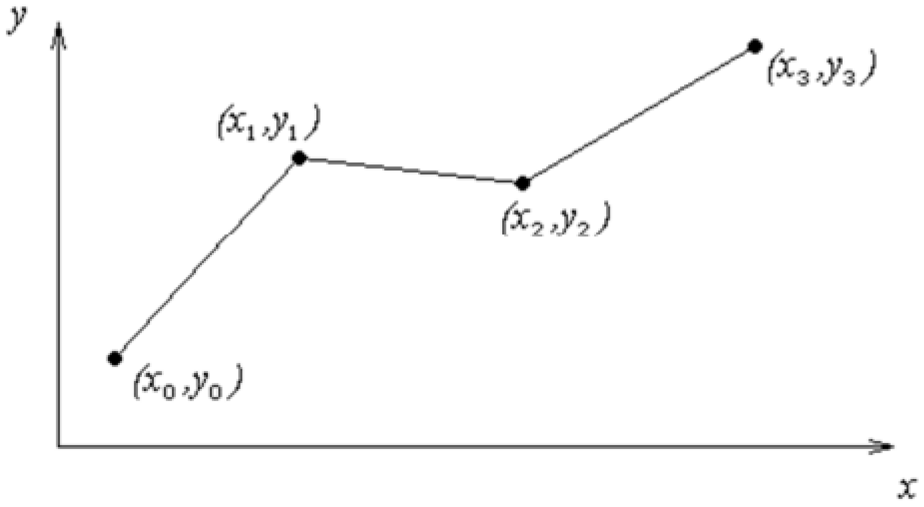
Spline interpolation is like the Polynomial interpolation

The interpolation uses low-degree polynomials in each of the intervals and chooses the polynomial pieces such that they fit smoothly together.

The resulting function is called a spline.

First order spline Interpolation:

Given (x_0, y_0), (x_1, y_1), … , (x_{n-1}, y_{n-1}), (x_n, y_n) fit linear splines to the data. This simply involves forming the consecutive data through straight lines.



First order spline formula:

$$
\begin{aligned}
f(x) &= f(x_0) + \frac{f(x_1) - f(x_0)}{x_1 - x_0}(x - x_0), && x_0 \le x \le x_1 \\
&= f(x_1) + \frac{f(x_2) - f(x_1)}{x_2 - x_1}(x - x_1), && x_1 \le x \le x_2 \\
&\;\;\vdots \\
&= f(x_{n-1}) + \frac{f(x_n) - f(x_{n-1})}{x_n - x_{n-1}}(x - x_{n-1}), && x_{n-1} \le x \le x_n
\end{aligned}
$$

Cubic Spline Interpolation

Cubic spline interpolation is a way of finding a curve that connects data points with a degree of three or less. Splines are polynomial that are:

smooth and continuous across a given plot.

continuous first and second derivatives where they join.

Problem Setup

We aim to construct a cubic spline interpolation for the given data points:

The function $y = g(x)$ is defined $(x_0, y_0), (x_1, y_1), \ldots, (x_n, y_n)$ $, x_{i+1}]$.

Each segment g_i(x) of the cubic spline is defined as:

Unknowns: a_i, b_i, c_i, d_i (4 per segment, leading to $4n$ unknowns)

$$g_i(x) = a_i(x - x_i)^3 + b_i(x - x_i)^2 + c_i(x - x_i) + d_i$$

Constraints

Interpolation Condition($2n$ constraints):

$$g_i(x_i) = y_i \quad \text{and} \quad g_i(x_{i+1}) = y_{i+1}$$

Continuity of First Derivative(n-1 constraints):

$$g_i'(x_{i+1}) = g_{i+1}'(x_{i+1})$$

Continuity of Second Derivative(n-1 constraints):

$$g_i''(x_{i+1}) = g_{i+1}''(x_{i+1})$$

Boundary Conditions(additional constraints):

These depend on the type of spline used (e.g., natural spline, clamped spline, etc.).

Total Constraints and Unknowns

- Unknowns:

  $4n$

- Constraints:

  $$2n + 2(n - 1) = 4n - 2$$

- Additional Constraints Required:

  2 extra constraints (e.g., boundary conditions).

Definitions

$h_i = x_{i+1} - x_i$ (interval length)

$\eta_i = y_{i+1} - y_i$ (difference in y-values)

System of Equations

For each segment $g_i(x)$, the coefficients are solved using:

1. $d_i = y_i$
2. $a_i h_i^3 + b_i h_i^2 + c_i h_i + d_i = \eta_i$
3. $3a_i h_i^2 + 2b_i h_i + c_i = c_{i+1}$
4. $6a_i h_i + 2b_i = 2b_{i+1}$

These equations are systematically solved to construct the cubic spline.

Steps for Solving

Solve equation (3) for a_i in terms of b_i
Solve equation (2) for c_i in terms of b_i
Substitute into equation (4).

Equation (4) becomes:

$$\frac{1}{3}h_i b_i + \frac{2}{3}(h_i + h_{i+1})b_{i+1} + \frac{1}{3}h_{i+1}b_{i+2} = \frac{\eta_{i+1}}{h_{i+1}} - \frac{\eta_i}{h_i}$$

Resulting Equation system

$$\begin{pmatrix} \dots & \dots & \dots & \dots & \text{missing} & \dots & \dots \\ \frac{1}{3}h_0 & \frac{2}{3}(h_0+h_1) & \frac{1}{3}h_1 & \dots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & \frac{1}{3}h_{n-2} & \frac{2}{3}(h_{n-2}+h_{n-1}) & \frac{1}{3}h_{n-1} \\ \dots & \dots & \dots & \dots & \text{missing} & \dots & \dots \end{pmatrix} \begin{pmatrix} b_0 \\ b_1 \\ \vdots \\ b_{n-1} \\ b_n \end{pmatrix}$$

$$= \begin{pmatrix} \text{missing} \\ \frac{\eta_1}{h_1} - \frac{\eta_0}{h_0} \\ \vdots \\ \frac{\eta_{n-1}}{h_{n-1}} - \frac{\eta_{n-2}}{h_{n-2}} \\ \text{missing} \end{pmatrix},$$

Boundary Conditions (Not-a-Knot Condition):

$g_0(x) = g_1(x)$ equivalent to $g_0'''(x) = g_1'''(x)$ which results in

$a_0 = a_1$ And is written in terms of b as following expression:

$$h_1 b_0 - (h_0 + h_1) b_1 + h_0 b_2 = 0,$$

$g_{n-2}(x) = g_{n-1}(x)$ equivalent to $g_{n-2}'''(x) = g_{n-1}'''(x)$ which results in $a_{n-2} = a_{n-1}$ And is written in terms of b as following expression:

$$h_{n-1} b_{n-2} - (h_{n-2} + h_{n-1}) b_{n-1} + h_{n-2} b_n = 0.$$

Final step

In the next step we insert the two new experession into the missing rows of the equation system to determine the value of b

Since we have a_i and c_i in terms of b_i; we can calculate the value of all unknowns After determining the values of b_i

References

https://www.math.hkust.edu.hk/~machas/numerical-methods-for-engineers.pdf

https://www.geeksforgeeks.org/cubic-spline-interpolation/

https://mathforcollege.com/nm/mws/gen/05inp/mws_gen_inp_ppt_spline.pdf

You can visit my GitHub page to view the Jupiter notebook:

- https://github.com/KiyaOfAUT/spline-interpolation

Thanks for your attention!