Q1. Discuss, compare and contrast various curve fitting and interpolation methods

Curve Fitting

- Problem statement: Given a set of $(n+1)$ point-pairs $\{x_i, y_i\}$, $i = 0, 1, \ldots n$, find an analytic, smooth curve in the interval $[x_0, x_n]$.
- Why we perform curve fitting?
  - To get estimates at some intermediate points
  - To produce a simplified version of a more complicated function
- Methods:
  - Interpolation for clean data:
    * Lagrange
    * Newton's divided-difference
    * Splines
  - Regression for noisy data:
    * Linear, Polynomial ...

Interpolation: Direct Approach

- To fit exactly $(n+1)$ data points, use the polynomial of degree n:

$$P_n = c_0 + c_1 x \ldots + c_n x^n$$

- Find c_i by solving the linear system of equations:

$$\begin{pmatrix} 1 & x_0 & \ldots & x_0^n \\ 1 & x_1 & \ldots & x_1^n \\ \vdots & \vdots & \vdots & \vdots \\ 1 & x_n & \ldots & x_n^n \end{pmatrix} \begin{pmatrix} c_0 \\ c_1 \\ \vdots \\ c_n \end{pmatrix} = \begin{pmatrix} y_0 \\ y_1 \\ \vdots \\ y_n \end{pmatrix}$$

- Caution: Not advisable to solve this system owing to the matrix-inversion!

- Do we have any inversion-free methods ?

Lagrange Interpolation

- An n-th degree Lagrange basis polynomial:

$$\phi_i(x) = \frac{\prod_{j=0,j\neq i}^{n}(x - x_j)}{\prod_{j=0,j\neq i}^{n}(x_i - x_j)} \quad i = 0, 1, \ldots n.$$

- Hence the Lagrange's interpolating polynomial is

$$P_n(x) = \sum_{i=0}^{n} c_i \phi_i(x)$$

- $\phi_i(x)$ has the property:

$$\phi_i(x_k) = \begin{cases} 1, & i = k; \\ 0, & i \neq k. \end{cases}$$

Lagrange (Cont'd)

- Using the above property, we get the coefficients

$$c_i = y_i$$

  hence much simpler to find coefficients!

- Limitations:
  - Redo the whole procedure when adding/deleting a point $\Rightarrow$ works bad with unknown order.
  - Divisions present in computing the Lagrange polynomial are expensive

Newton's Divided Difference Polynomial

- An i-th order Newton basis polynomial:

$$\phi_i(x) = \prod_{j=0}^{i-1}(x - x_j)$$

- The interpolating polynomial in terms of Newton's basis:

$$P_n(x) = \sum_{i=0}^{n} c_i \phi_i(x)$$

- Get the Coefficients:

$$c_i = [y_0, \ldots y_i]$$

where $[y_0, \ldots y_i]$ is the notation for an $(i+1)$-th order divided difference.

Newton (Cont'd)

- Virtues:
  - For equally spaced data points, replace the divided differences with functional differences.
  - Less arithmetic operations in writing the polynomial than that of Lagrangian.
  - Easy to add/delete a point $\Rightarrow$ works well for an unknown order
- All the above methods yield the same results for a given set of points. However, for larger n,they all suffer from the Runge's phenomenon.

Runge's Phenomenon

- Is the error is always guaranteed to diminish with increasing polynomial order? No!
- Runge observed an increasing oscillatory behavior when using polynomial interpolation with polynomials of high degree.
- Why?
  - The error between the generating function and the interpolating polynomial of order n is bounded by the n-th derivative of the generating function. For Runge-type functions (e.g., $f(x) = \frac{1}{1+25x^2}$), the magnitude of the derivative increases.

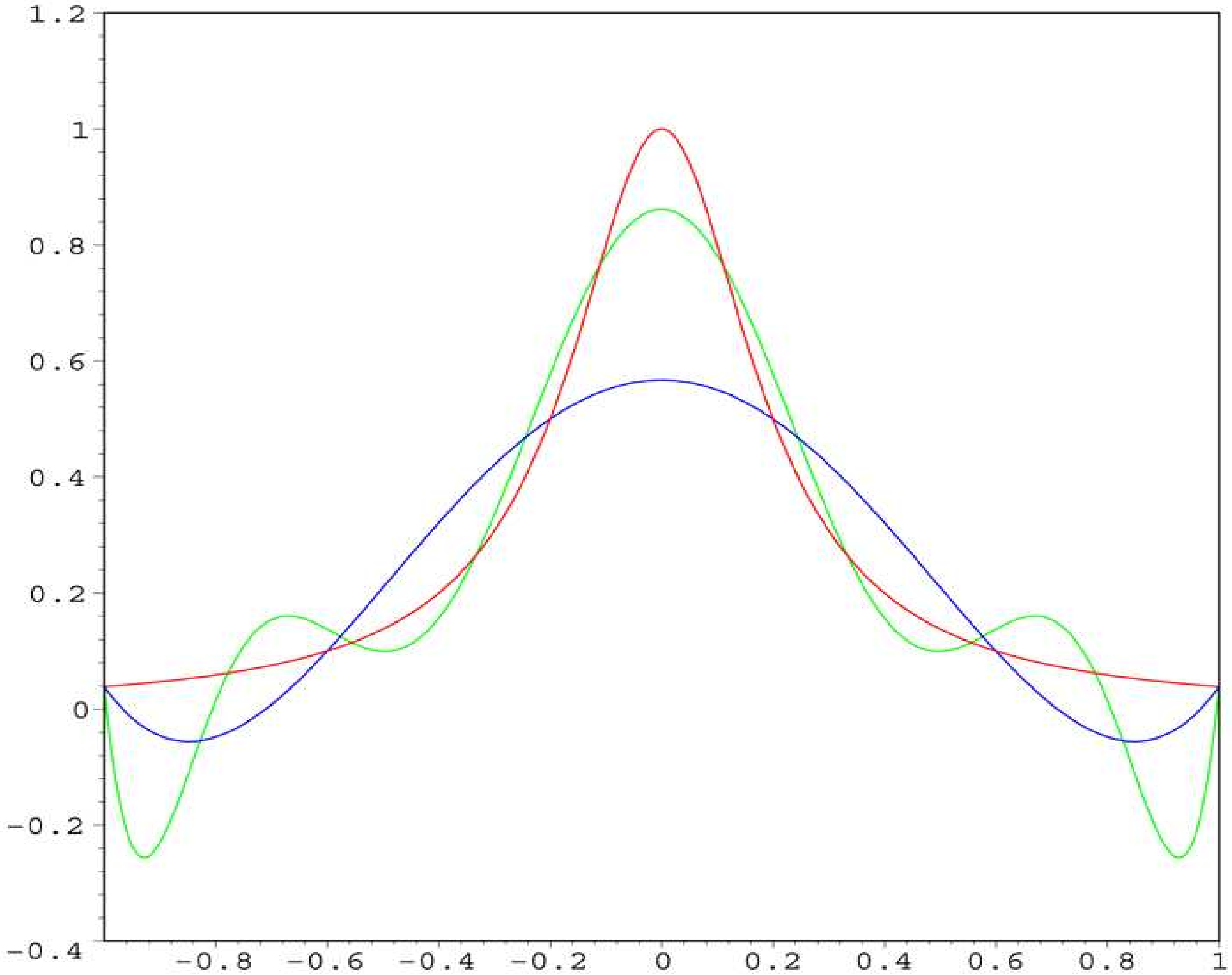
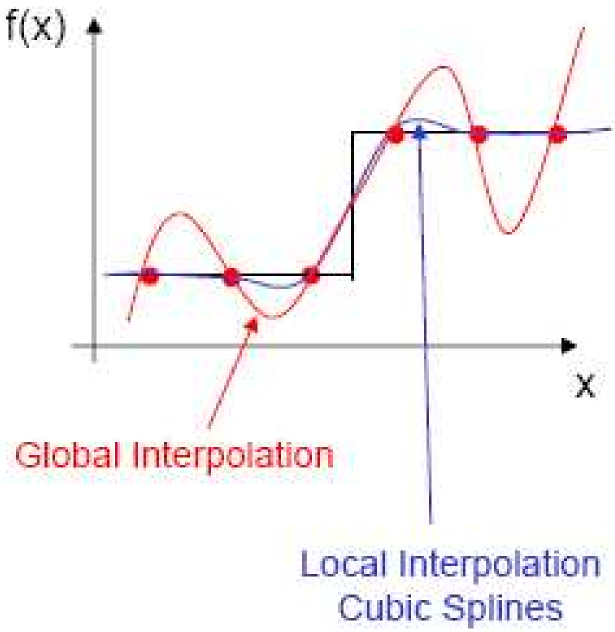


Figure 1: Runge Phenomenon in a nutshell (Runge function-red, 5th-order polynomial-blue, 9th-order polynomial-green)

Splines

- Local approach dividing into sub-intervals and fit to a low-order polynomial while preserving the following properties:
  - Continuity at the boundary
  - Slope continuity at the boundary
  - Curvature continuity at the boundary ...
- Spline candidates:
  - Linear
  - Quadratic
  - Cubic
- Useful for functions with local abrupt changes

Linear Splines



Linear Splines

$$
\begin{aligned}
f(x) &= f(x_0) + m_0(x - x_0)\,, \quad x_0 \le x \le x_1 \\
f(x) &= f(x_1) + m_1(x - x_1)\,, \quad x_1 \le x \le x_2 \\
&\vdots \\
f(x) &= f(x_{n-1}) + m_{n-1}(x - x_{n-1})\,, \quad x_{n-1} \le x \le x_n
\end{aligned}
$$

$$m_i = \frac{f(x_{i+1}) - f(x_i)}{x_{i+1}) - x_i}$$

Quadratic Splines

Quadratic Splines

$$f_i(x) = a_i x^2 + b_i x + c_i \ , \ \ x_{i-1} \le x \le x_i$$

3n coefficients

Continuity

$$\begin{aligned} a_{i-1}x_{i-1}^2 + b_{i-1}x_{i-1} + c_{i-1} &= f(x_{i-1}) \\ a_i x_{i-1}^2 + b_i x_{i-1} + c_i &= f(x_{i-1}) \end{aligned}$$

2(n-1) conditions

End Conditions

$$\begin{aligned} a_1 x_0^2 + b_1 x_0 + c_1 &= f(x_0) \\ a_n x_n^2 + b_n x_n + c_n &= f(x_n) \end{aligned}$$

2 conditions – 2n total

Derivatives

$$f_i'(x) = 2a_i x + b_i$$

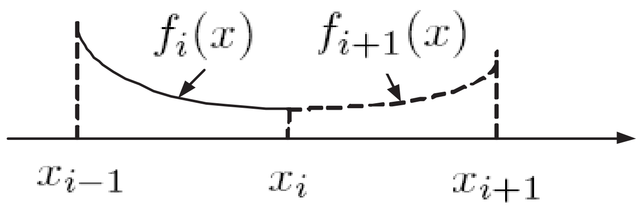
$$2a_{i-1}x_{i-1} + b_{i-1} = 2a_i x_{i-1} + b_i$$

n-1 conditions – 3n-1 total

$$a_1 = 0$$

1 condition – 3n total

Cubic Splines



- Cubic spline of the form:

$$f_i(x) = a_i x^3 + b_i x^2 + c_i x + d_i \qquad x_{i-1} \le x \le x_i$$

- Find $4n$ unknowns from the following conditions:
  1. Continuity: $2(n-1)$ conditions
  2. End: 2 conditions
  3. Slope continuity: $(n-1)$ conditions
  4. Curvature continuity: $(n-1)$ conditions
  5. Curvature at end points: 2 conditions

Linear Regression

- Assumptions:
  - We look for a general trend of the data set
  - Noisy data are available in large scale
- Fitted model:

$$f(x) = a_0 + a_1 x.$$

- Objective function: Sum of error-squared:

$$J(a_0, a_1) = \sum_{i=0}^{n} (y_i - a_0 - a_1 x_i)^2$$

Linear Regression (Cont'd)

- To find the unknown coefficients, set the partial derivatives to be zero:

$$\begin{aligned}\frac{\partial J}{\partial a_0} &= -2\sum_i (y_i - a_0 - a_1 x_i) = 0 \\ \frac{\partial J}{\partial a_1} &= -2\sum_i [y_i - a_0 - a_1 x_i] x_i = 0\end{aligned}$$

- Rearrange the above to get

$$\begin{pmatrix} n+1 & \sum x_i \\ \sum x_i & \sum x_i^2 \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \end{pmatrix} = \begin{pmatrix} \sum y_i \\ \sum x_i y_i \end{pmatrix}$$

- For the above case, the coefficient matrix $A = BB^T \Rightarrow BB^T[a] = By$. Solve the above system using the SVD or Cholesky.

References

- C. F. Gerald & P. O. Wheatley, *Applied numerical analysis*, Pearson, 2004.
- S. Chapra & R. Cannale, *Numerical methods for engineers*, 2006.

Thank you!